DOCTORAL THESIS

IMAGE FEATURE REPRESENTATION USING DEEP LEARNING FOR INSTANCE SEARCH AND SCENE RECOGNITION

Submitted by

SURAJIT SAIKIA

in fulfillment of the requirements for the Degree of

PHILOSOPHIÆDOCTOR (PH.D.)

DOCTORAL PROGRAM: PRODUCTION AND COMPUTER ENGINEERING

A dissertation supervised by

PROF. DR. ENRIQUE ALEGRE GUTIÉRREZ,

DR. LAURA FERNÁNDEZ ROBLES

León, October 2021

universidad de león

TESIS DOCTORAL

REPRESENTACIÓN DE CARACTERÍSTICAS DE LA IMAGEN UTILIZANDO EL APRENDIZAJE PROFUNDO PARA LA BÚSQUEDA DE INSTANCIAS Y EL RECONOCIMIENTO DE ESCENAS

desarrollada por

SURAJIT SAIKIA

a fin de optar al grado de

DOCTOR POR LA UNIVERSIDAD DE LEÓN

PROGRAMA DE DOCTORADO: INGENIERÍA DE PRODUCCIÓN Y COMPUTACIÓN

Tesis doctoral dirigida por

PROF. DR. ENRIQUE ALEGRE GUTIÉRREZ,

DR. LAURA FERNÁNDEZ ROBLES

León, Octubre 2021

Abstract

This thesis investigates the creation of novel algorithms for representing images to address two important areas in the field of computer vision: content-based image retrieval (CBIR) and scene recognition. CBIR can be classified into two types, instance-level retrieval and category-level retrieval, and in this thesis, we address the former. Motivated by our joint work with INCIBE, we build deep learning-based systems that can help Law Enforcement Agencies to match evidences in crime scene investigations, among a wide range of other applications. In particular, we propose two algorithms for CBIR, one based on the colour description of objects and the other one on the texture description of patches on images, and another additional algorithm for scene prediction and retrieval that relies on the combination of local and global scene content.

CBIR for instance-level retrieval aims at retrieving images from an image or video database that contain the same object or scene as the one depicted in a query image. We introduce two algorithms to address this task in order to gain robustness against colour and texture variances, respectively. On the one hand, we propose colour neural descriptors that are composed of convolutional neural networks (CNNs) features obtained by combining different colour spaces and colour channels. In contrast to previous works, which rely on fine-tuning pre-trained networks, we compute the proposed descriptors based on the activations generated from a pre-trained CNN without fine-tuning. Also, we take advantage of an object detector to optimize the proposed instance retrieval architecture to generate features at both local and global scales. In addition, we introduce a stride based query expansion technique to retrieve objects from multi-view datasets. Finally, we experimentally demonstrated that the proposed colour neural descriptors obtain state-of-the-art results on the Paris 6K, Revisiting-Paris 6k, INSTRE-M and COIL-100 datasets, with mean average precision of 81.70%, 82.02%, 78.8% and 97.9%, respectively.

On the other hand, we focus on the texture properties of images. In crime scene investigations, some clues may come from texture patches of images that do not contain much information about the object contour, like a t-shirt lying on the floor. To define the characteristics of such images, the texture patterns are the prime cues for visual descriptions. We propose a novel texture feature descriptor that is based on the combination of the spatial images and their discrete Fourier transform maps. We further present a new and efficient texture-based image retrieval framework based on a region proposal network, convolutional autoencoders and transfer learning. We extract the features from the latent space layer of the encoder as compact texture descriptors. We conducted experiments to validate the effectiveness of the proposed method and obtained average retrieval rates of 80.36%, 90.25%, and 81.02% on the Outex, USPtex, and Stex datasets. In addition, we also experimented with the TextileTube dataset, that consists of images from a real indoor real scenario. In this case, we calculated the arithmetic means of precision@k for three different intervals, where k ranges from 1 to 10, 1 to 20 and 1 to 30, and the obtained results were 99.2%, 93.2% and 67.9%, respectively. Besides, the performance achieved in these four datasets outperformed the state-of-the-art results reported in the literature.

The second area of research concerns indoor scene recognition, which is a challenging and growing task in the field of computer vision. Although CNNs can achieve outstanding results on outdoor scene recognition, their performance lacks similar robustness in the recognition of indoor scenes. This is due to the high spatial variability in semantic cues (e.g. objects), and due to the presence of similar objects throughout different scene categories. To overcome these issues, we propose DeepScenePip (DSP), a pipeline with three modules: *object-centric*, *objects-to-scene* and *scene-centric*, which independently focus on local and global scene content, respectively. The proposed pipeline has three novel components. Firstly, it produces an image caption from the recognized object labels to predict scenes using a natural language processing approach. Secondly, it relies on a weight function that combines object and scene information for an overall scene prediction. Thirdly, it includes a query expansion technique which turns out to be very beneficial in scene retrieval. We evaluated our approach for indoor scene recognition and indoor scene retrieval on three public datasets: MIT-67 Indoor, NYU-v2 and Hotels-50k. The accuracy achieved (MIT-67 Indoor = 94.5%, NYU-v2 = 74.5% and top-1 accuracy 10.1% without occlusion and 7.8% with medium occlusion on the Hotels-50k) demonstrated the effectiveness of the proposed method, which also significantly outperforms existing state-of-the-art approaches.

This thesis contributes to the development of methods for creating robust descriptors to colour, texture and view-point changes and presents frameworks to use them in CBIR and scene recognition systems.

Resumen

Esta tesis investiga la creación de algoritmos novedosos para representar imágenes con el fin de abordar dos áreas importantes en el campo de la visión por ordenador: la recuperación de imágenes basada en el contenido (CBIR, del inglés *content-based image retrieval*) y el reconocimiento de escenas. Los sistemas CBIR se pueden clasificar en dos tipos, recuperación a nivel de instancia y recuperación a nivel de categoría, y en esta tesis abordamos la primera. Motivados por nuestro trabajo conjunto con INCIBE, construimos sistemas basados en el aprendizaje profundo que pueden ayudar a las Fuerzas de Seguridad a cotejar las evidencias en las investigaciones de la escena del crimen, además de a una amplia gama de otras aplicaciones. En particular, proponemos dos algoritmos para la CBIR, uno basado en la descripción del color de los objetos y otro en la descripción de la textura de parches en imágenes. Además, proponemos un método adicional, que permite predecir y recuperar escenas, basándose en la combinación del contenido local y global de la escena.

Los sistemas CBIR para la recuperación a nivel de instancia tienen como objetivo recuperar imágenes de una base de datos de imágenes o vídeos que contengan el mismo objeto o escena que el representado en una imagen de consulta. Introducimos dos algoritmos para abordar esta tarea con el fin de ganar robustez frente a las variaciones de color y textura, respectivamente. Por un lado, proponemos descriptores neuronales de color que se componen de características de redes neuronales convolucionales (CNN, del inglés *convolutional neural networks*) obtenidas mediante la combinación de diferentes espacios de color y canales de color. A diferencia de los trabajos anteriores, que se basan en el ajuste fino de las redes preentrenadas, nosotros calculamos los descriptores propuestos basándonos en las activaciones generadas a partir de una CNN preentrenada sin ajuste fino. Además, aprovechamos un detector de objetos para optimizar la arquitectura de recuperación de instancias propuesta para generar características tanto a escala local como global. Adicionalmente, introducimos una técnica de expansión de consultas basada en zancadas (*strides* en inglés) para recuperar objetos de conjuntos de datos multivista. Finalmente, demostramos experimentalmente que los descriptores neuronales

de color propuestos obtienen resultados superiores al estado del arte en los conjuntos de datos Paris 6K, Revisiting-Paris 6k, INSTRE-M y COIL-100, con una precisión media de $81,70\%$, $82,02\%$, $78,8\%$ y $97,9\%$, respectivamente.

Posteriormente, nos centramos en describir y utilizar las propiedades de textura de las imágenes. En las investigaciones de escenas de un crimen, algunas pistas pueden provenir de parches de textura de las imágenes que no contienen mucha información sobre el contorno del objeto, como puede ser una camiseta tirada en el suelo. Para definir las características de dichas imágenes, los patrones de textura conforman los principales indicios para obtener una descripción visual. Proponemos un nuevo descriptor de características de textura que se basa en la combinación de las imágenes espaciales y sus mapas de transformada discreta de Fourier. Además, presentamos un nuevo y eficiente modelo de recuperación de imágenes basado en la textura, que se apoya en una red de propuesta de regiones, autocodificadores convolucionales y aprendizaje por transferencia. Extraemos las características de la capa de espacio latente del codificador como descriptores de textura compactos. Realizamos experimentos para validar la eficacia del método propuesto y obtuvimos tasas de recuperación medias de $80,36\%$, $90,25\%$ y $81,02\%$ en los conjuntos de datos Outex, USPtex y Stex. Además, también experimentamos con el conjunto de datos TextileTube, que consiste en imágenes en un escenario real de interior. En este caso, calculamos las medias aritméticas de la precisión@k para tres intervalos diferentes, en los que k tomaría valores en los intervalos $[1,10]$, $[1,20]$ y $[1,30]$, siendo los resultados obtenidos de $99,2\%$, $93,2\%$ y $67,9\%$, respectivamente. Además, el rendimiento obtenido en estos cuatro conjuntos de datos superó los resultados del estado del arte recogidos en la literatura.

La segunda área de investigación se refiere al reconocimiento de escenas en interiores, que es una tarea desafiante y en expansión en el campo de la visión por ordenador. Aunque las CNN pueden obtener resultados extraordinarios en el reconocimiento de escenas en exteriores, su rendimiento carece de la misma solidez en el reconocimiento de escenas en interiores. Esto se debe a la alta variabilidad espacial de las claves semánticas (por ejemplo, los objetos) y a la presencia de objetos similares en diferentes categorías de escenas. Para superar estos problemas, proponemos DeepScenePip (DSP), un *pipeline* con tres módulos: *object-centric* y *objects-to-scene*, y *scene-centric*, que se centran independientemente en el contenido local y global de la escena, respectivamente. El proceso propuesto tiene tres componentes novedosos. En primer lugar, produce una descripción de la imagen a partir de las etiquetas de los objetos reconocidos para predecir las escenas mediante un enfoque de procesamiento del lenguaje natural. En segundo lugar, utiliza una función de peso que combina la información sobre el objeto y la escena para realizar una predicción global de la misma. En tercer lugar, incluye una técnica de expansión de consultas que resulta muy beneficiosa para la recuperación de escenas. Hemos evaluado nuestro enfoque para el reconocimiento y la recuperación de escenas en interiores en tres conjuntos de datos públicos: MIT-67 Indoor, NYU-v2 y Hotels-50k. La precisión alcanzada (MIT-67 Indoor = $94,5\%$, NYU-v2 = $74,5\%$ y la precisión top-1 $10,1\%$ sin oclusión y 7,8% con oclusión media en el Hotels-50k) demostró la eficacia del método propuesto, que también supera significativamente los enfoques del estado del arte existentes.

Esta tesis contribuye al desarrollo de métodos para crear descriptores robustos a los cambios de color, textura y punto de vista y presenta marcos para utilizarlos en sistemas CBIR y de reconocimiento de escenas.

Contents

List of Figures

List of Tables

Índice general

Anexo B: Resumen de la tesis en castellano

Acknowledgements

First and foremost, I am extremely grateful to my supervisors Dr. Enrique Alegre and Dr. Laura Fernandez for their unwavering guidance and invaluable pieces of advice during my Ph.D. research. Without their continuous support, motivation, expertise, and patience, it would have been impossible to complete this thesis.

I would like to express my sincere gratitude and thanks to Dr. George Azzopardi who was my supervisor during my research stay at the University of Groningen, Netherlands.

Very special thanks to my colleagues Eduardo, Rubel, Deisy, Wesam, Pablo, Javi, Victor, Guru, Fran, Manuel, David, and Alex for their support and encouragement. Their company in the laboratory will be always appreciated and remembered.

I would like to acknowledge the University of León and INCIBE for providing us with a world-class research ambiance, facilities, and opportunities to seamlessly carry out research activities. Also, my research would have been impossible without the aid and support of Junta de Castilla y Leon for providing me with the grant.

I offer my thanks to my friends Victor, Ivan, Srabonty, Erica, Paula, Juan Carlos, Adriano, Irene, and Jose, for always being there for me throughout the research journey.

Most importantly, I express my deepest thanks to my family members Kabindra Saikia, Anjali Saikia, Pragyan Saikia, and Annanyya Kashyap, for their unfathomable love and support throughout my life.

Finally, I thank and praise God, who has been my constant support in every phase of my life, and I humbly dedicate this work to him.

Surajit Saikia
León
14th October 2021

Chapter 1

Introduction

1.1. Motivation

Image search and retrieval, object detection and scene recognition are three of the most important tasks in the domain of computer vision with a phenomenal amount of ongoing research. Modern intelligent systems are expected to perform such tasks without human intervention, and computer vision enables such systems to process and recognise objects and scenes in images and videos, reaching human-level recognition performance. Nowadays, most of the digital equipment captures high-resolution images that can surpass the vision system of humans[1], and the goal of computer vision is to understand the content unfolded in images, which typically is composed of various objects and layouts. In particular, the prime goal of a computer vision system is to simulate the capability of human vision for inferring information from image data.

After the recent breakthroughs of deep learning, computer vision has taken a great leap even surpassing human performance in the domain of image classification, image retrieval and object recognition, among other computer vision fields. The deep learning based systems have been gaining popularity in the recent years due to their revolutionary success in various domains. In fact, deep learning applied to computer vision enables machines to understand the environment around them based on visual perceptions. In Fig. 1.1, we show some of the predominant computer vision-based applications such as *image retrieval*, *scene understanding*, *2D object detection*, *3D pose estimation*, *robotic navigation* and *semantic segmentation*. As illustrated in the indicated figure, the visual perception stack is the image data representing a real-world environment, and the computer vision module, powered by deep learning, process them based on the needs of the applications. Moreover, the main underlying principle behind the performances of all the aforementioned applications relies heavily on two primary steps: (1) image representation and (2) object detection. Therefore, inspired by the impact of deep learning in the computer vision domain, we address two important applications in this thesis: (a) content-based image retrieval (CBIR) and (b) scene recognition in indoor environments, by focusing especially on image representation and object detection.

[1] https://nanonets.com/blog/ai-visual-inspection/

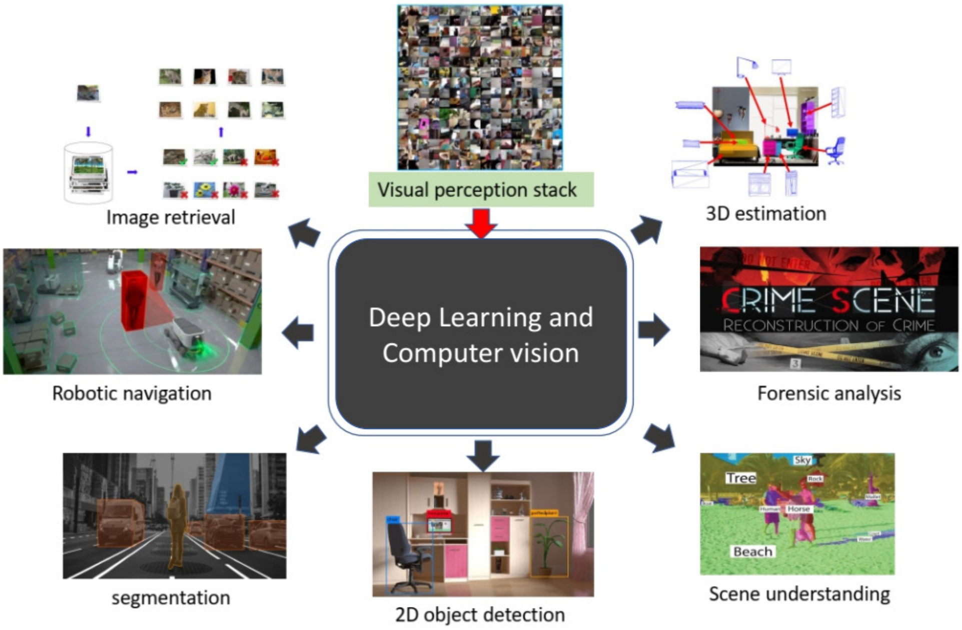
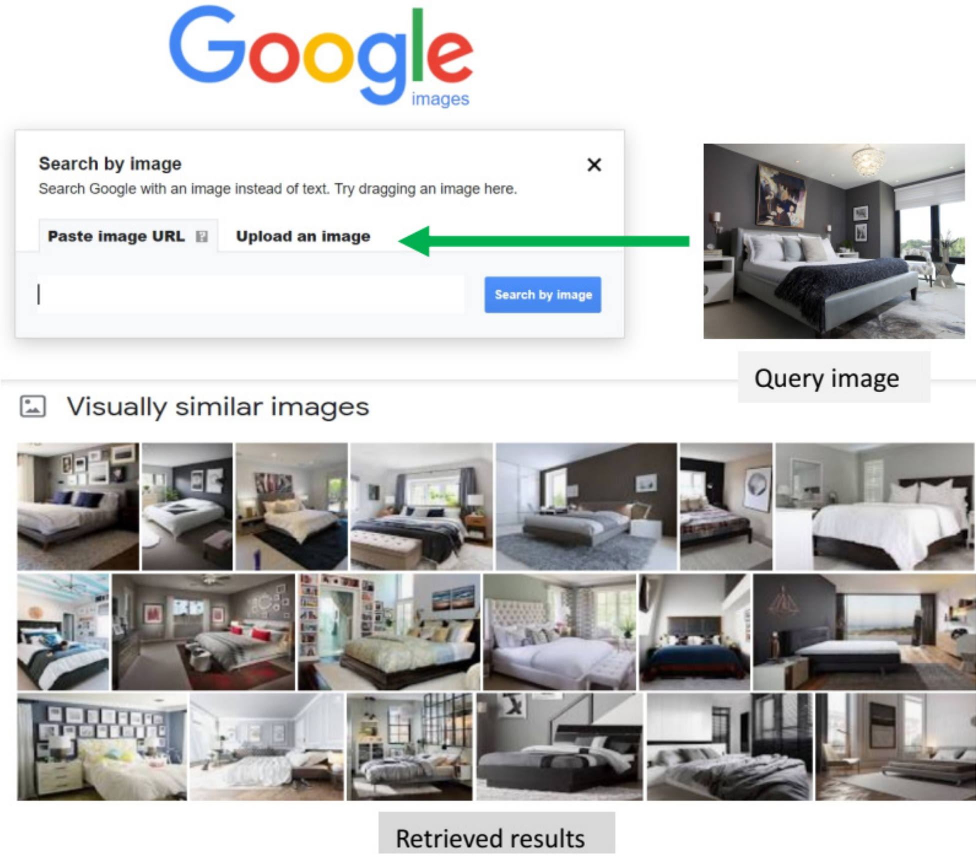


Figure 1.1: Examples of computer vision applications.

In the last few years, due to the availability of cellphones and cameras to almost everyone in the globe, there has been a rapid proliferation of digital image data in the order of billions. In fact, every day millions of images are being uploaded via many social media platforms such as *Instagram*, *Facebook*, *Twitter*, etc. Till now, *Facebook* alone has uploaded more than 250 billion photos, and people are uploading 350 million new photos every single day[2]. Recently, after the emergence of *Instagram*, there has been an unprecedented growth in the number of digital images. Due to this tremendous amount of digitally acquired data, image search is a challenging and non-trivial task as compared to one decade before. Modern image search approaches with the aid of artificial intelligence (AI) can understand the context and content in images and then return related images to a given query. Using image search, we can change the way we interact with the world around us. As we are dominated by visual perception, it is quite natural to use an image to initiate a search. In such situations, image search technology can be applied to search and retrieve desired images in a way that a textual description would have never been able to do so. Through image search, users can use their mobile phone camera to snap pictures, and then an image recognition system would find similar items based on the search results.

[2] https://blog.microfocus.com/how-much-data-is-created-on-the-internet-each-day/



Search by image

Search Google with an image instead of text. Try dragging an image here.

Figure 1.2: *Google* image search.

It is worth mentioning that technology-based enterprises such as *Google*, *Microsoft*, *Amazon*, and *Bing* have developed their own image search engines since there is an overwhelmingly large number of users that rely on images. In Fig. 1.2, we show an example of an image search using *Google* images, whereupon providing a query image of a bedroom, the *Google* image search engine returns images with beds. However, the technology is still in an infant stage, and to highlight the challenge, in Fig. 1.3, we show the estimated quantity of images that are uploaded on a daily basis through some of the prominent social platforms. Moreover, due to the digitization in the current era, the amount of data will continue to increase, as a result, it opens a great opportunity to the researchers for devising methods that can improve the efficacy of image search using CBIR systems in terms of scalability, accuracy and speed.

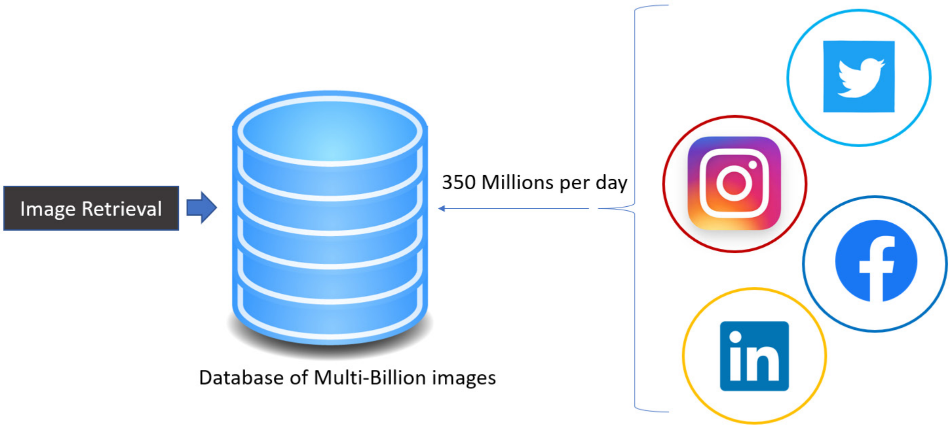


Figure 1.3: A visual illustration of the daily uploads of images using *Facebook*, *Instagram*, *LinkedIn* and *Twitter*.

In Fig. 1.4, we show a general pipeline for the CBIR process. The features could be extracted using deep learning based models, such as convolutional neural networks (CNNs) (Simonyan and Zisserman, 2014; Ren et al., 2017; Tan and Le, 2019), which have outperformed the conventional techniques (i.e. SIFT (Lowe, 2004a), SURF (Bay et al., 2006) or HOG (Dalal and Triggs, 2005)) in terms of feature representation. Not only they output end features of an image, but the different layers of a CNN can also be exploited to obtain low-level and high-level features. Furthermore, transfer learning can be applied on top of pre-trained CNN networks for cross-domain image retrieval achieving enhanced features. Similar to CNNs, convolutional autoencoders can also be applied to CBIR. Unlike CNNs, autoencoders can be trained in an unsupervised manner to learn low-level features of an input image. In general, these low-level features are called latent features, which are encoded and compact low dimensional representations of the input image. Using this compact feature vector, the original image can be reconstructed back to its original aspect. Hence, using an autoencoder, a low dimensional vector can be generated to represent different types of images. In fact, due to the compactness of the latent space, the resultant vector can be used as a feature descriptor for faster or even real-time image retrieval. In Fig. 1.5, we illustrate the architecture of a convolutional autoencoder, in which the input images are reconstructed back based on the latent space representation. Such deep learning-based methods try to perceive the high order semantic of images based on their low-level image attributes and are able to outperform handcrafted conventional methods (Gkelios et al., 2021; Nanni et al., 2017).

The CBIR task can be further categorized into two types: instance-level retrieval

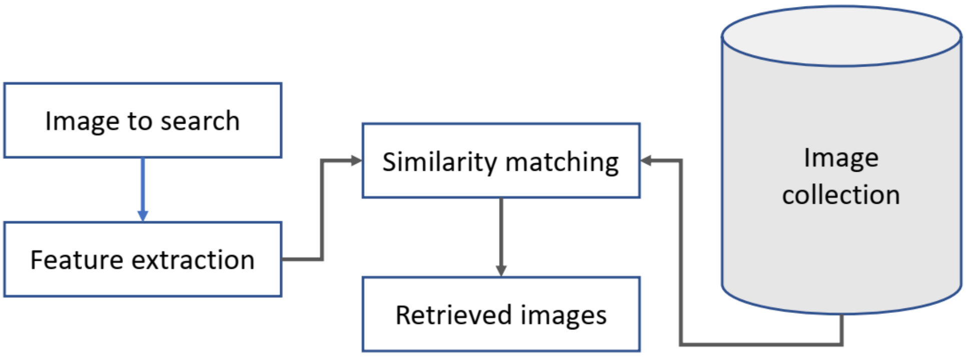


Figure 1.4: Pipeline for content-based image retrieval.

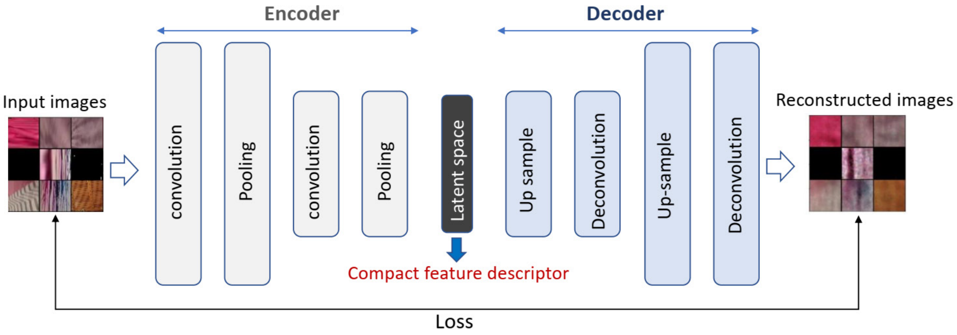


Figure 1.5: An illustration of latent space representation of an autoencoder. The compact feature descriptor can be used to represent images for addressing CBIR.

and category-level retrieval. Instance-level image retrieval or instance retrieval is the problem of retrieving images from a database representing the same object or image as the one illustrated in a query image (Tan et al., 2021). Whereas in category-level retrieval the aim is to search for similar images (or objects) that are in the same category and there is a flexible scale of relevance (as shown in Fig. 1.6). In this work, we mainly aim at instance-level image retrieval, where the goal is to create an effective method that is competitive with the current applications to retrieve images containing a particular query instance from large-scale image databases.

However, the difficulty of a CBIR system for instance-level retrieval increases as people search for specific objects, texture patches or clothes to get related recommendations. To emphasise this problem, Fig. 1.7 shows some examples that are taken from the INSTRE dataset (Wang and Jiang, 2015). The overlaid bounding boxes represent the query objects aimed to search, whereas the images in the right

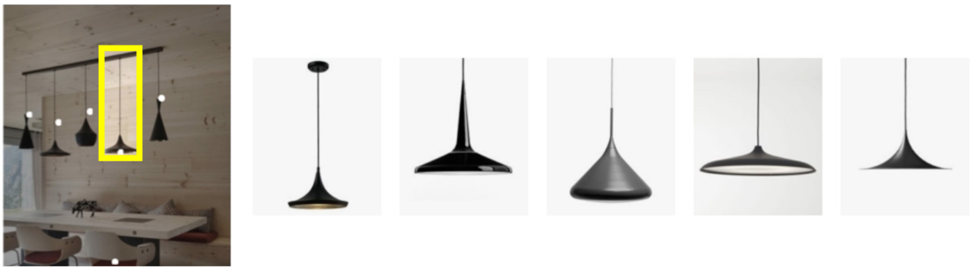


Figure 1.6: Example of the result of an image search system based on a given query instance. The query is overlaid with a yellow bounding box.

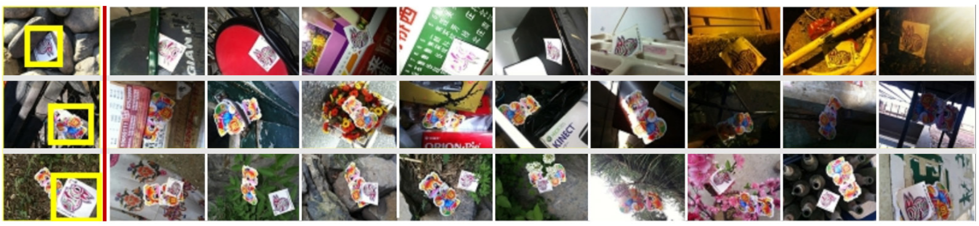


Figure 1.7: Three examples of the result of a CBIR system for instance-level image retrieval. The yellow bounding boxes represent the queried objects, and images in the right are the ground-truth images that contain the query object.

contain the found object. In these examples, the difficulty lies in the fact that the instance can be present in various shapes and sizes with different backgrounds and also partially occluded. Given this scenario, a general CBIR system would fail to retrieve the relevant images that contain the given query patch or object. However, this step can be approached by employing object detectors. For instance-level retrieval in which the instances are objects, automatic object detection is one of the most important components of a search algorithm. The object detection algorithm generates various regions where the possibility of object presence is high and attempts to identify them in the case that an object is detected. Using object detection and classification, one can search objects belonging to a detected category. For example, if a query image contains a book, then an object detection algorithm can be applied to an image database for searching images with books. However, if a user wishes to search for a query object for which the network is not trained, then the challenge manifolds. In this case, one solution would be to compute the similarity among the query descriptor and the descriptors of all the detected objects. But, the problem that hinders the performance of such an instance retrieval system is the presence of multiple instances which are visually similar to the supplied query. To

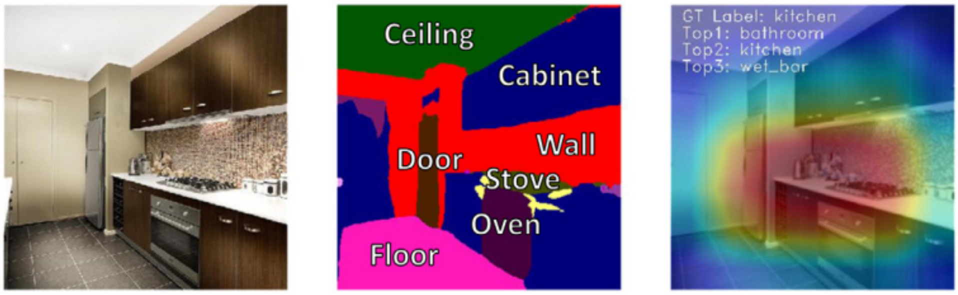


Figure 1.8: Indoor scene recognition using semantic segmentation of mid-level categories. [From López-Cifuentes et al. (2020)]

address this issue, a highly discriminative descriptor is necessary to create an effective CBIR system for instance-level retrieval, which is our main focus of motivation for this thesis.

Another important application in the field of computer vision which we address in this thesis is indoor scene recognition. It is a hot research topic whose complexity is on top of the image understanding domain (Zhou, Lapedriza, Khosla, Oliva and Torralba, 2017). Scene recognition provides a fundamental description of the image content by assigning semantic labels instead of just listing the recognised objects. It is also used in a wide range of applications, such as intelligent robotics, scene retrieval, human-computer interaction, autonomous navigation and video surveillance. Besides, it can help visually impaired persons to explore indoor and outdoor environments by detecting and avoiding obstacles. Moreover, scene recognition is regarded as a prerequisite for computer vision tasks such as scene understanding and image retrieval. In Fig. 1.8 we show an example where semantic segmentation is used with mid-level categories such as *ceiling, cabinet, door, wall, stove, oven and floor* to predict or recognise the scene category (López-Cifuentes et al., 2020). Nevertheless, such visual understanding is essentially a complex task for machines due to the presence of abstract semantic entities like objects and scenes, and thus, it is difficult to model scene categories due to a larger semantic gap.

From a real-life application point of view, the main motivation that drives our work regarding CBIR for instance-level retrieval and scene recognition is to support law enforcement agencies (LEA) to investigate crime scenes using new AI-based technologies. Recently, crimes are increasing at an incredibly fast pace, with new trends emerging constantly as criminals use new technologies against the government, business organizations and individuals. Criminals may cause serious harm and threats to people worldwide. As the world is progressing in terms of technological advancement, proportionally the crimes are becoming more agile. One

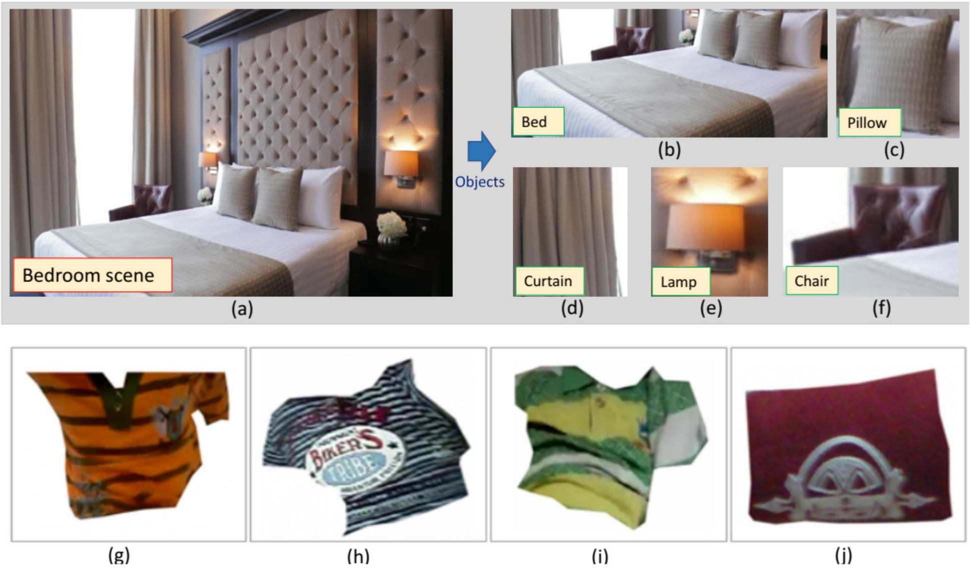


Figure 1.9: The image (a) represents a bedroom (indoor scene), and the cropped images: (b) Bed, (c) Pillow, (d) Curtain, (e) Lamp and (f) Chair are the objects present in the bedroom. Whereas, the images from (g-h) are the sample texture patches from Europol's 'stop child abuse - trace an object' activity.

crime that draws the attention of LEAs is the sexual exploitation of children. In such cases, the clues derived from images may empower the investigative work of forensic departments. Image retrieval for crime scene investigation (Liu and Wu, 2019; Liu et al., 2017) and the recognition of scenes can help to uncover various crimes by linking similar images or videos. In Fig. 1.9, we illustrate some of the images provided by Europol, the European Union's LEA, for one of their activities which aims at stopping child abuse by tracing images[3]. As illustrated, some of these images, which are related to sexual explicit material involving minors, can be objects present in an indoor scene (Fig. 1.9 (a-f)) in which colour can help to discriminate among possible objects. Other images are texture patches (Fig. 1.9 (g-j)) that do not contain much information about the object contour. One way to trace them is to compare them against dense database environments to find similarities with other suspicious images using a CBIR system. To define the characteristics of such images, the colour and texture patterns of the objects and patches are the prime cues for visual descriptions. Texture-based image retrieval was also a task in the

[3] https://www.europol.europa.eu/stopchildabuse

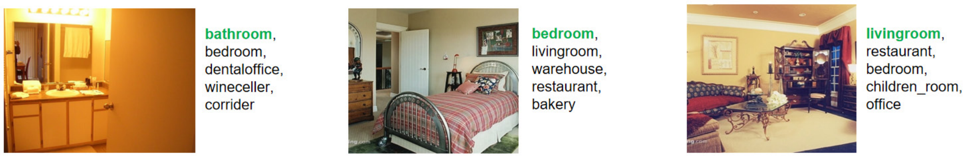


Figure 1.10: An example of indoor scene recognition problem.

European project ASASEC[4] and now in GRACE[5], where some European LEAs are or were involved, to provide solutions based on computer vision and deep learning to help LEAs to fight against Child abuse.

Indoor scene recognition can be applied to forensic evidence analysis for human trafficking investigation by identifying scenes related to various crimes (Fig. 1.9 (e)). For instance, Fig. 1.10 presents an example of three scenes being predicted as *bathroom*, *bedroom* and *livingroom*, where most of the crimes related to indoor takes place. The recognised scene images can serve as visual evidence about the place a victim has been trafficked and gives insight into trafficking operation. In this case, a scene recognition algorithm can be applied for indoor scene retrieval.

In the last few years, the number of online images related to human trafficking has grown at an alarming rate. The availability of such photographs taken in various geographical locations can serve as visual evidence during forensic analysis of trafficking. Most of such images are captured in an indoor environment, such as hotel bedrooms and bathrooms. Identifying the specific indoor place in which such images were taken would provide insights to the law enforcement agencies regarding the trafficking operations. In Fig. 1.11 we show an example of a law enforcement query, where the goal is to recognize the query image based on the retrieved images. In such images, the victims are often masked due to privacy reasons, and in this context, the algorithm should be robust to occlusions and lighting condition. Given this scenario, if an algorithm can recognize the scenes, then it would facilitate the law enforcement agencies to track crimes related to trafficking.

The most important aspect that governs the remarkable performance of deep learning algorithms is the representation of the feature descriptors. The intermediate features of images extracted from deep learning algorithms, such as CNNs, which are trained for image classification and object detection tasks can be used to generate visual representations of the images in the form of feature vectors (Gkelios et al., 2021). Those representations are the unique signatures that can be used to discriminate between images and objects which are visually unrelated. As a result, this property can be exploited to create CBIR and scene recognition systems. How-

[4] https://ec.europa.eu/home-affairs/financing/fundings/projects/HOME_2010_ISE_AG_043_en
[5] https://cordis.europa.eu/project/id/883341

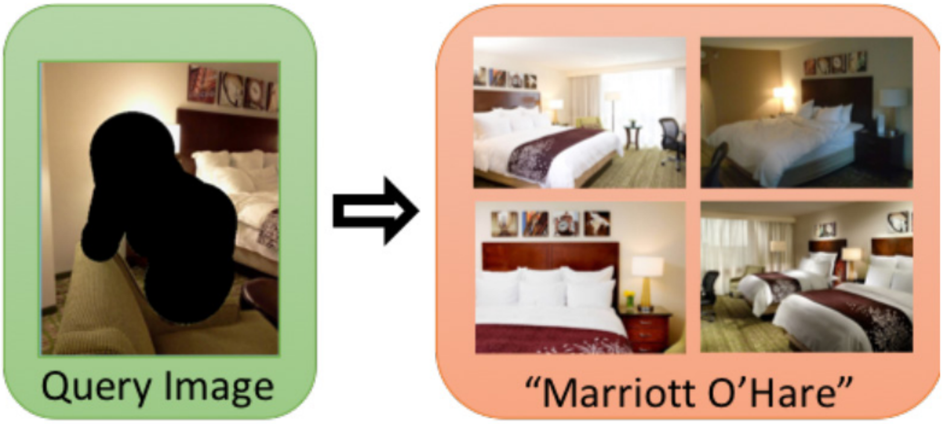


Figure 1.11: Identification of the hotel in which a picture was taken

ever, even with the aid of state-of-the-art deep learning algorithms, the CBIR and scene recognition tasks have large scopes for improvement in terms of precision of retrieval and accuracy of scene recognition. The challenge is no longer to create neural networks with different architectures and train with large scale image datasets. The bigger problem is that the present approaches are not able to generalize well with image data that is complex, multi-faceted and obscure. For such computer vision tasks, the aim is to mimic the functional attributes of the human brain using neural networks, which is a fascinating domain for research due to existing possibilities for improvement and further development. Moreover, for CBIR and scene recognition systems, the colour and texture information are the prime attributes, and by integrating them with the features obtained from a CNN more discriminative descriptors can be obtained. In the end, with these motivations and scopes for further improvement, in this work, we create solutions to address CBIR and indoor scene recognition using deep learning based approaches that focus in the following three aspects.

- **Colour neural descriptors for instance retrieval**: Colour descriptors are proposed for CBIR at instance-level retrieval using CNN features and colour models.

- **Texture based instance retrieval**: A novel texture descriptor, named as deep Fourier texture descriptor (DFTD), is proposed for CBIR at instance-level retrieval.

- **Indoor scene recognition using *object-centric* and *scene-centric* approaches**:

A novel scene recognition architecture is proposed that uses global and object features for scene prediction and retrieval.

1.1.1. Colour neural descriptors for instance retrieval

Instance retrieval requires correct detection of all objects (instances) along with precisely localizing each of them. To search for the most similar objects or instances to a given query on a dataset, we need to compare the query with all the possible instances present in the dataset. In real-world scenarios, the images might contain diverse objects and layouts. Moreover, the objects may appear partially occluded or in cluttered environments, which may lead to significant variations concerning viewpoints, scale, rotation, translation or illumination of the objects. To address those challenges, the most recent works focus on generating object proposals in images using end-to-end CNNs to learn the location of objects.

The most crucial aspect of this retrieval task is to localise the objects, which strongly depends on the object appearance. Undoubtedly, the colour provides essential cues about object resemblance. Colour is one of the most basic and straightforward visual feature that represents the spectral content of objects. Besides, colour based features should be invariant to pixel translation or rotation in images. For CBIR at instance-level retrieval, the image query and the features of an object found in an image have to be nearly or completely similar. By using colour as a feature, we can identify and discriminate among different images and the objects present in them. Since colour provides vital information on images and to increase the discriminative power of the neural features without fine-tuning, we propose to use different colour spaces and combinations of colour channels to transform the CNN features into robust descriptors.

1.1.2. Texture based instance retrieval

In addition to colour features, texture patterns also play a major role in image representation as discriminative features. In general, most CBIR systems (Zheng et al., 2017), extract features that represent various shapes and patterns present in an image. However, a query patch containing just a texture pattern without contour information presents a more challenging scenario to effectively retrieve images. Mainly, the indoor scenes images contain cluttered objects or instances which are typically textiles and cloths. Such textures of cloths and textiles present quasi repetitive patterns, and due to the presence of texture images with large intra-class variation and inter-class similarities, the retrieval task becomes challenging even with deep learning based algorithms. However, since in texture images the patterns are repeated across the whole image, each part of the image have a common frequency. As a general illustration, in Fig. 1.12, we show an example of four texture patches

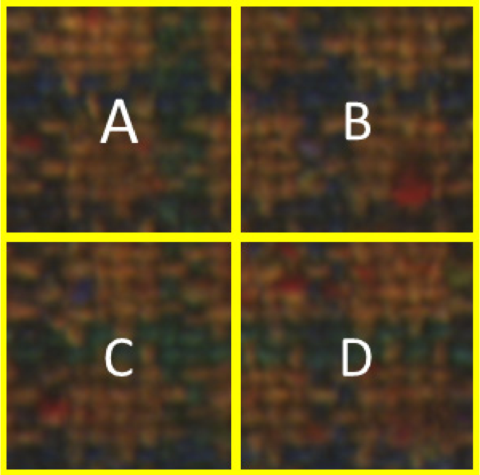


Figure 1.12: Example of four texture patches, framed in yellow, containing similar repeated patterns. The four patches were cropped from the same object.

belonging to a single image, marked by boxes A, B, C and D, containing similar repeated patterns. This information can be exploited using Fourier transform and can be used along with deep learning to make an efficient retrieval system. Driven by the challenges and to outperform other state-of-the-art methods in the literature, we were motivated to devise a new descriptor using deep learning and Fourier transform.

1.1.3. Indoor scene recognition using *object-centric* and *scene-centric* approaches

A scene is a real-world environment that contains multiple objects and surfaces that are organised in a meaningful way. By seeing the contents in a scene image, humans are capable of recognising a scene effortlessly and rapidly without observing the details of all the objects present. For example, we can identify a bathroom without noticing some of the specific objects such as a towel, sinks and soap. This human ability of the primate visual system is known as core object recognition (Cadieu et al., 2014), which is very fundamental for environment interpretation. But, automatic scene recognition is a long-standing research problem, where an algorithm needs to predict labels such as ‘bedroom’, ‘bathroom’ or ‘kitchen’ to an input image based on the overall contents present in the scene.

The complexity of the scene recognition task lies partially in the ambiguity between different scene categories with similar appearances and sets of objects which makes that a scene might be highly similar to another one. Scene recognition not only concerns the objects present but their semantic relationships and their contextual information with regards to the background. Though CNNs have been proved to automatically yield some solutions, the complexity of the problem increases with the number of categories. In addition, to train a model to classify various scenes, the training datasets need to be balanced with millions of images. Moreover, scene recognition models that perform better in outdoor scenes do not work well in indoor domains. This is because the outdoor images can be characterized by global spatial properties, and indoor images are defined by various objects contained in them. It requires learning of shared properties of indoor objects such as furniture, beds, chairs and duplicated instances, and also the global context of the image needs to be considered. Due to these inherent challenges, in this thesis, we propose a solution by focusing on the objects and global scene properties by introducing a novel hybrid deep architecture based on CNNs for indoor scene recognition. In addition, using this hybrid architecture we build a scene retrieval pipeline that can be applied to forensic evidence analysis for human trafficking investigation (Stylianou et al., 2019). In Fig. 1.13 we show some of the images on the Hotel-50k dataset where each row represents scene images of the same room.

1.2. Objectives and main contributions

There are three main objectives in this dissertation, and we summarise them along with the main contributions achieved as follows:

1. Instance retrieval by proposing colour descriptors using CNNs features obtained by combining different colour spaces and colour channels.
   - *a*) We introduce a novel method to create colour neural descriptors based on the activations generated from a pre-trained deep CNN.
   - *b*) We present a hybrid architecture that is composed of two different CNNs, which we use successfully for an instance retrieval pipeline without employing fine-tuning techniques.
   - *c*) We demonstrate experimentally that our proposed colour neural descriptors outperform the state-of-the-art in four datasets for image retrieval, COIL-100, INSTRE-M, Paris-6K and Revisiting-Paris 6k.
2. Texture based instance retrieval using latent space representation of discrete Fourier transformed (DFT) maps as image features.
   - *a*) We propose a novel descriptor, known as deep Fourier texture descriptor (DFTD). We extract the features of the descriptor from the latent space

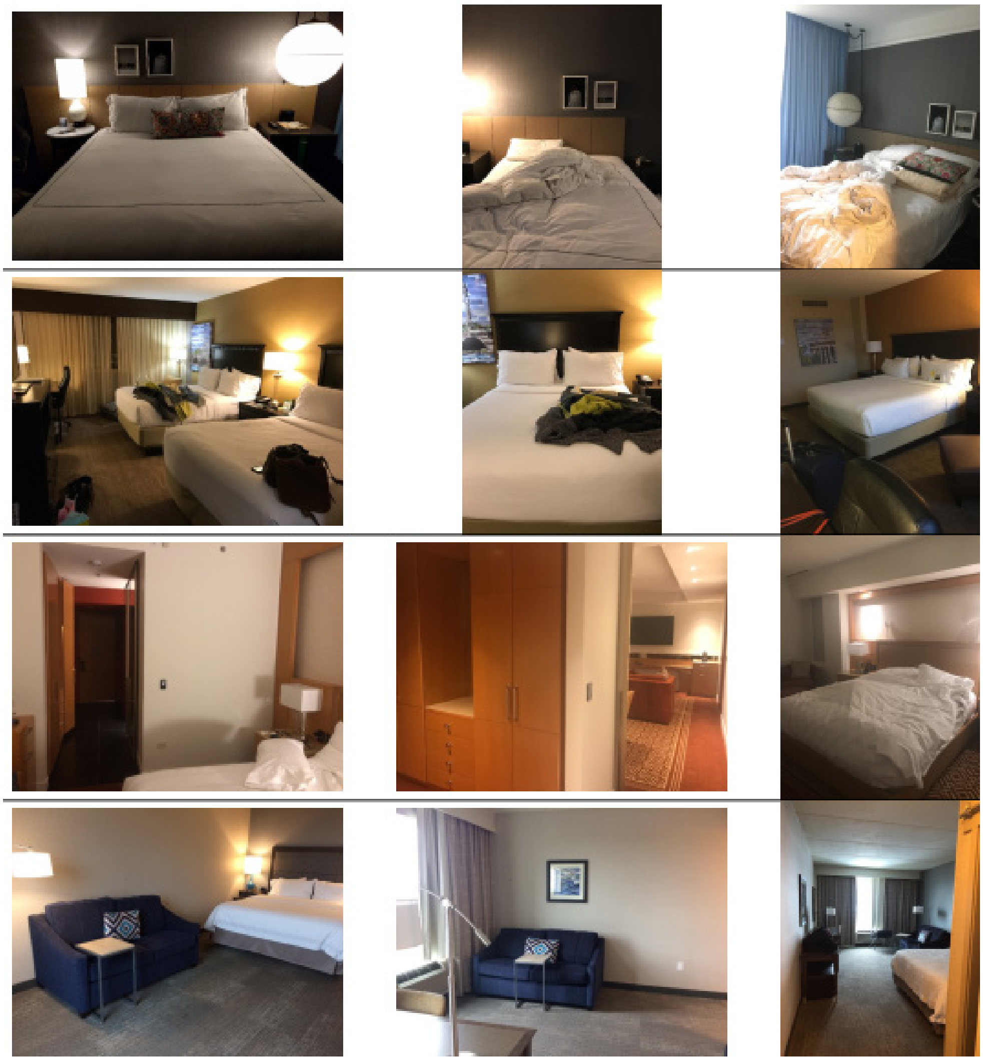


Figure 1.13: Images on the Hotel-50k dataset. Each row contains images of the same hotel room.

layer of a convolutional autoencoder whose inputs are the outcomes of the blended magnitude spectrum of a DFT and the spatial information of each image.

b) We present a CBIR framework for texture based instance retrieval which uses an object detector to propose prospective texture regions which are fed to an autoencoder.

   c) We evaluate the proposed texture-based image retrieval approach in the context of a real-world application, which can be applied for image, instance and object retrieval for crime scenes investigation during forensic analysis. We also assess the proposed texture descriptor both for texture based image retrieval and for texture classification on two widely-used public datasets outperforming state-of-the-art works.

3. Indoor scene recognition and retrieval by proposing a novel hybrid deep architecture that combines *object-centric* and *scene-centric* features.

   a) We introduce a novel architecture known as DeepScenePip, which combines both object and global features.

   b) We propose a new technique based on natural language processing to predict a scene category from captions generated from the object labels recognized in a given image.

   c) We present a weight function named as weighted combination of *object-centric* and *scene-centric* modules (WCOS) that combines object and scene information for an overall scene prediction.

   d) We introduce a retrieval approach for query-based scene retrieval of indoor images.

1.3. Thesis Organization

In this section, we describe the structure of this doctoral thesis. This first introductory chapter has been focused on motivating the work we present, its main objectives and original contributions. Now, the remaining chapters of this thesis are organised as follows.

Chapter 2 contains a detailed review of state-of-the-art methods related to image retrieval, object detection and scene recognition which were published in top research journals and conference proceedings. At first, we describe some of the widely used global and local descriptors for image representation that were applied in CBIR in the last decades. Then, we highlight some of the recent methods based on deep learning, where the feature extracted using such methods have become the new state of the art descriptors for CBIR and object detection. Finally, we review methods in the literature regarding scene recognition.

Chapter 3 introduces a colour descriptor for instance retrieval which is based on the activations generated from a pre-trained Deep CNN. In addition, it also presents a hybrid architecture composed of two different CNNs that we use successfully as the instance retrieval pipeline without employing fine-tuning techniques. In addition, this chapter presents a technique for retrieving multiview images based on query expansion.

Chapter 4 presents a descriptor for texture retrieval based on the discrete Fourier transform and the latent space representation of a VGG autoencoder. The features of the descriptor are extracted from the latent space layer of a convolutional autoencoder whose inputs are the outcomes of blending the magnitude spectrum of a DFT and the spatial information of the images. This chapter also describes our proposed CBIR framework for texture retrieval which uses an object detector to propose prospective texture regions and their corresponding descriptors.

Chapter 5 addresses the problem of scene recognition by introducing a novel architecture known as DeepScenePip. It combines both *object-centric* and *scene-centric* approaches to address scene recognition and retrieval. Also, a new technique is introduced for identifying various scenes by training a network using image captions generated from object labels of a detector. Finally, the chapter presents a framework for scene retrieval by combining *object-centric* and *scene-centric* approaches to create discriminative features.

Chapter 6 contains a summary with the conclusions of this thesis and gives an outlook of possible future work lines to extend the presented work.

1.4. Publications and research activities

This section presents the research results obtained during the completion of this doctoral thesis.

1.4.1. Publications related to the manuscript

- Surajit Saikia, Laura Fernández-Robles, Eduardo Fidalgo Fernández, and Enrique Alegre. “Colour Neural Descriptors for Instance Retrieval Using CNN Features and Colour Models.” IEEE Access 9 (2021): 23218-23234.

- Surajit Saikia, Laura Fernández-Robles, Enrique Alegre, and Eduardo Fidalgo. ”Image retrieval based on texture using latent space representation of discrete Fourier transformed maps.” Neural Computing and Applications (2021): 1-16.

- Deisy Chaves, Surajit Saikia, Laura Fernandez-Robles, Enrique Alegre, and Maria Trujillo. “A systematic review on object localisation methods in images.” Revista Iberoamericana de Automática e Informática Industrial 15, no. 3 (2018): 231-242.

- Oscar García-Olalla, Enrique Alegre, Laura Fernández-Robles, Eduardo Fidalgo, and Surajit Saikia. “Textile retrieval based on image content from

CDC and webcam cameras in indoor environments." Sensors 18, no. 5 (2018): 1329.

- Surajit Saikia, Eduardo Fidalgo, Enrique Alegre, and Laura Fernández-Robles. "Object detection for crime scene evidence analysis using deep learning." In International Conference on Image Analysis and Processing, pp. 14-24. Springer, Cham, 2017.

- Surajit Saikia, Eduardo Fidalgo, Enrique Alegre, and Laura Fernández-Robles. "Query based object retrieval using neural codes." In International Joint Conference SOCO'17-CISIS'17-ICEUTE'17 León, Spain, September 6–8, 2017, Proceeding, pp. 513-523. Springer, Cham, 2017.

1.4.2. Other Activities

Teaching Activities

- Delivered a lecture in the topic *Neural Networks* to Bachelor students.

- **Co-supervisor of final Bachelor thesis**: Andrés Fernández, *Object detection in industrial images using YOLO*, directors: Laura Fernández Robles and Surajit Saikia (Ongoing work)

International Mobility

- Research stay at University of Groningen, Netherlands, Duration: 3 months.

Participation in international conferences

- Attendance and poster presentation at the 19th International Conference on Image Analysis and Processing (ICIAP 2017).

Participation in research projects

- Addendum 22 to the Framework Agreement between INCIBE (Spanish National Cybersecurity Institute) and the University of Leon. Creation of the following deliverables:

  - *Object detection and recognition for crime scene analysis.*
  - *Face recognition using FaceNet.*
  - *Image super-resolution to enhance lower quality images.*

- Addendum 01 to the Framework Agreement between INCIBE (Spanish National Cybersecurity Institute) and the University of Leon.

- *Indoor scene recognition and retrieval for addressing human trafficking*.

Chapter 2

State-of-the-art

In the field of computer vision, the representation of digital images based on their visual content is one of the most important factors for tasks such as image classification, image retrieval, object detection and scene recognition. In order to process those images for addressing the aforementioned tasks, the common approach is to describe them using characteristic features that highlight the visual patterns. For the image classification and retrieval tasks, in the literature, various image descriptors are proposed, together with object descriptors for object detection and object retrieval. Earlier, for creating a generic computer vision system, most of these descriptors were hand-crafted, which were mainly focused on overcoming specific issues like scale, rotation, illumination and occlusions. In particular, there is a trade-off between accuracy and computational complexity in designing handcrafted features. For instance, the most well-known SIFT (Lowe, 2004b) descriptor is invariant to scale and rotation, but it has a high computational complexity since computing the features is a two-step process. To overcome the inherent limitations, many variants of the original descriptors are proposed. One such example is SURF (Bay et al., 2006), which is faster and robust against image transformations as compared to SIFT. Moreover, such descriptors are used to train traditional machine learning algorithms. However, as the scale of data increases, the scope of such algorithms becomes limited while comparing to deep learning approaches (Gkelios et al., 2021).

Recently, pioneered by the advancements of deep learning, CNNs has shown remarkable performance in the computer vision domain, with their ability to learn rich image representations as opposed to handcrafted features. The CNNs demonstrated supremacy in terms of accuracy while trained using a large amount of data. The biggest advantage of the deep learning-based algorithm is that they learn high-level feature in a hierarchical manner, which eliminates the need for domain expertise (Gkelios et al., 2021). In addition, some works exploited trained deep learning models as generic feature extractor, and are further mixed with traditional hand-designed features to obtain more discriminative descriptors (Nanni et al., 2017).

In this thesis, we mainly focus on creating discriminative descriptors to represent and identify objects and images for CBIR and scene recognition. Accordingly, we next provide a review of the state-of-the-art methods related to the research work presented in this thesis.

- Image feature representation using local and global descriptors
- Deep learning approaches for image retrieval
- Scene recognition

2.1. Image feature representation using global and local descriptors

An image can be mainly represented using global and local descriptors. The global descriptors produce a single feature vector that could describe the colour, texture and shape properties of an image based on its overall content. Whereas, the local descriptors focuses on image patches and regions. The main objective of CBIR methods is to use the descriptors for identifying and retrieving all images with similar visual content with a given query image from a large scale image database. We next present a comprehensive review of image representation using global and local features.

2.1.1. Global features

The global characteristics of an image are defined by its colour (Wang and Hua, 2011), shape (Bai et al., 2016) and texture (Wang et al., 2014). At first, the most commonly used global features were based on colour information that include the MPEG-7 features sets, which are histogram intersection (HI) descriptors, correlogram descriptors and dominant colour descriptors (DCD). The HI descriptor was proposed by Swain and Ballard (1991), which is robust to scale, rotation, and as well as to variations in image resolution. In this approach, global colour features are taken into account based on colour histograms. However, an adequate selection of the colour space and the number of bins are an important deciding factor for a good performance. The colour correlogram proposed by Huang et al. (1997) takes the global colour and the local colour distribution of each colour space in an image into account. It determines the probability of finding the colour pairs at a specified distance, and also it represents the spatial information of the pixel distributions. However, since it is computationally expensive, later, colour auto correlogram (CAC) was proposed by Chen et al. (2010). CAC captures the spatial correlation between identical colours and provides significant benefits over colour correlograms in terms of computational speed. The dominant colour descriptor (DCD) was among the most important colour-based descriptors in the MPEG-7 standard. In DCD, the overall colour information of an image is represented using dominant colours instead of using all of them (Yang et al., 2008). Extraction of dominant colours is dependent on several factors, such as the selection of the colour space, the colour quantization, the determination of the dominant colours and the calculation of the percentage of each dominant colour. In (Lantagne et al., 2003), the percentage of the extracted colours is calculated, and if it is more than five per cent, then the colour is considered to be dominant. The colour space used in MPEG-7 consists of RGB, HSV, YCrCb and HMMD, and HSV colour space is regarded to approximate the human level perception. Considering this, Shao et al. (2008) chose HSV colour space and quantized an image to obtain 72 different colours. From the quantized image, the N most representative colours $C_i, i = 1..N$, are selected as dominant colours and their percentages, $P_i, i = 1..N$ was calculated. Similarly, using HSV colour space, SCD colour descriptor was proposed. This descriptor is scalable since it uses Haar wavelet transformation (Stanković and Falkowski, 2003) for image representation at different scales.

Similar to colour, texture is another key component for global image representation. The texture can be interpreted as a visual pattern that is repeated in different scales and directions. At first, the texture descriptors were based on statistical approaches, which are grey level co-occurrence matrix (GLCM), Markov random fields (MRF) model and edge histogram descriptor (EHD), and a comparison of them was performed in (Vogel and Schiele, 2006). The texture can also be represented by MPEG-7 feature sets that aims at capturing the general characteristics of textures. Moreover, the MPEG-7 standardizes three different types of texture descriptors: homogeneous texture descriptor, text browsing descriptor and non-homogeneous texture descriptor. More details about these descriptors can be found in (Erol et al., 2005). The work proposed by Hamouchene and Aouat (2014) used random transform to extract global features of texture images, where the random transform is a projection of the 2D image into a set of 1D radial line that ensures the texture descriptor to be rotation invariant.

Apart from the colour and texture features, the shape details of an image can provide salient information for identifying real-world shapes and objects. The shape features can be represented using boundary-based and region-based descriptors. For boundary representation, the contours of the objects in an image are extracted, whereas, for region representation, region-based descriptors are used to determine salient areas in images. However, the boundary representation has a serious drawback since it is difficult to extract boundaries from natural images which contain various textured patterns. Therefore, most of the shape descriptors are region-based. Earlier popular works that used region-based techniques were discussed in (Vogel and Schiele, 2006), which are based on moment invariants, wavelet transforms, Gabor wavelets, gradient features, discrete cosine transforms and curvature scale space. Also, the shape details of images can be represented using Fourier transform. Sokic and Konjicija (2016) proposed a method for extracting Fourier descriptor for shape-based image retrieval by preserving phase, and recently, Yang and Yu (2019) introduced a novel multi-scale Fourier descriptor based on triangular features to identify shapes.

Most of these earlier CBIR works focused on using a single feature among colour, texture and shape. Later, many approaches combined the features to enhance the retrieval performance. For example, Gray and Tao (2008); Prosser et al. (2010) combined texture and colour features. Pujari et al. (2010) presented a framework that uses colour and shape features from Lab and HSV spaces to retrieve edge features, and the experiments carried out in the Corel dataset (Tao, 2009) demonstrated the efficiency of the method. (Khan et al., 2013) addressed the photometric variations by proposing a clustering technique to create colour descriptors, which are later combined with shape descriptors. Alzu'bi et al. (2015) introduced an optimized image descriptor that combines colour histogram in HSV space with the rootSIFT (Arandjelovic and Zisserman, 2012) descriptors and outperformed many state-of-the-art methods. (Cortes et al., 2016) evaluated 11 image descriptors and concluded that combinations of Gabor descriptors and dominant colour descriptors provide better performance. Lately, some works proposed to combine colour with other texture or shape descriptors. In this line, Ahmed et al. (2018) used Canny edge histograms combined with discrete wavelets of YCbCr colour images. Sotoodeh, Moosavi and Boostani (2019) presented two approaches to extract discriminative features for colour image retrieval based on Radial Mean Local Binary Pattern. In most cases, the texture features are generally combined with the colour features. In order to describe texture features, firstly, Chun et al. (2003) introduced block variation of local correlation coefficients (BVLC) and block difference of inverse probabilities (BDIP). Later, in (Chun et al., 2008), those two texture descriptors were combined with colour features for colour image retrieval. Similarly, Chun et al. (2003) combined colour histograms with the BDIP and BVLC texture features and experimented using Corel-5k (Duygulu et al., 2002), UKbench (Nister and Stewenius, 2006) and Holiday's datasets (Jegou et al., 2008). In (Liu and Yang, 2013), colour and texture features are combined with the shape features and obtained better retrieval performances as compared to other approaches that use only colour and texture features. Other works, which combine all the three features are mentioned and studied extensively in (Liu and Yang, 2013; Liu et al., 2010). Recently, Chigateri and Sonoli (2021) presented a new method for image retrieval by combining HSV colour space, RGB histogram and block contour.

Even though combining features improves the accuracy of retrieval, there are two major issues associated with it. First, combining features may reduce the retrieval speed as compared to using single features, and secondly, the retrieval speed may become slow due to the large size of the feature vector resulting after the combination of different descriptors. In fact, before the deep learning was popular, some global descriptors were developed by aggregation of hand-crafted local descriptors

(Jégou, Douze, Schmid and Pérez, 2010; Jegou et al., 2011; Jégou and Zisserman, 2014). Currently, the most high performance global descriptors are the off-the-shelf features (Gkelios et al., 2021) obtained from CNNs, which we present in Section 2.2.

2.1.2. Local features and aggregations of global features

The performance of the global descriptors is limited when the images have complex visual constitutions. Most of the global feature-based methods fail to retrieve images due to variations in occlusions, viewpoint change, illumination and image shape. The notable local feature descriptors, such as SIFT, SURF, LBP (Ojala et al., 1996) and HOG (Dalal and Triggs, 2005) emerged as popular descriptors for effective image retrieval and could resolve the issues more efficiently than the global features. The local descriptors have been widely used for patch feature extraction, where a query image can be searched locally and compared with numerous patches in an image database. Among the local descriptors, SIFT is the most notable and widely used descriptor. Up to some extent, SIFT is robust to variation in illumination, occlusions and geometric distortion. However, SIFT is computationally expensive due to its multi-stage processing, and later, many variants of the SIFT descriptor were proposed to address its limitations (Burghouts and Geusebroek, 2009; Arandjelovic and Zisserman, 2012).

Another prominent local descriptor is the LBP (Local Binary Pattern), which is widely used to extract texture features. Due to its invariance to lighting changes and low computational complexity, LBP has been the most widely used descriptor for CBIR. In the medical domain, LBP has been used as a texture feature to identify malignant breast cells and to find slices in brain magnetic resonance (Nanni et al., 2010). However, LBP was sensitive to noises present in images, and to solve this problem, local ternary patterns was proposed by Tan and Triggs (2010). Later, many variants of LBP have been proposed. In (Zhu et al., 2013), LBP has been enhanced using colour information for image retrieval. This method increases the dimensionality of the feature vector, and to reduce the dimension, uniform LBP (ULBP) and orthogonal combination of LBP (OC-LBP) was proposed. Singh et al. (2018) proposed a novel local colour descriptor known as local binary pattern to colour images (LBPC) to represent local colour texture. Recently, Garg and Dhiman (2021) proposed a computationally effective and rotational invariant descriptor based on the LBP to extract texture features.

Moreover, until the advent of deep learning, the bag-of-visual-words (BoVW) framework was one of the most popular feature representation methods (Csurka et al., 2004). In this method, the local descriptors are aggregated to develop global descriptors for effective image retrieval (Nowak et al., 2006; Van Gemert et al., 2009; Jégou, Douze and Schmid, 2010a). In particular, the local descriptors are quantified into finite visual words, and then an image is described using the frequency of occurrence of those visual words. The BoVW features are created using a clustering technique that imposes computational complexity, and in addition, it has other drawbacks, such as ambiguity of the visual words and typically very large feature vectors. However, using the approximate nearest neighbour search method, the BoVW features turned out to be effective for retrieval in medium databases. Also, to improve its efficacy, spatial pyramid matching (Lazebnik et al., 2009) was proposed as a standard component of BoVW features to address missing spatial structural information.

These local image descriptors were usually assessed using neighbourhood search approaches to find similar images from large databases. To approximate nearest neighbourhoods, there exists a fast algorithm known as FLANN (Muja and Lowe, 2014), that searches k-nearest neighbours (k-NN) of high dimensional data. Another reported effective method for approximating neighbour search is mentioned in (Jegou, Douze and Schmid, 2010b), and it provides high accuracy with a very fast retrieval speed. Using this approach, the SIFT and GIST (Oliva, 2005) descriptors demonstrated high accuracy with high retrieval speed.

Moreover, some of the most powerful local descriptors were further enhanced, for example, Burghouts and Geusebroek (2009) introduced colour-SIFT, which is more robust than the original SIFT with respect to colour and photo-metrical variations. In (Heikkilä et al., 2006), centre-symmetric local binary pattern (CS-LBP) descriptor was proposed, which is an interest region descriptor that combines SIFT with LBP. Experimental results shown that CS-LBP performs better than SIFT when images are subjected to severe illumination variations. Later, Zhu et al. (2013) intended to enhance the LBP by increasing its discriminative power and photometric invariance. The authors of (Zhu et al., 2013) proposed OC-LBP and six other descriptors based on the OC-LBP which enhanced the colour information for region description. After experimenting using OC-LBP for different applications (Zhu et al., 2013), the OC-LBP descriptors outperformed SIFT, colour-SIFT, CS-LBP, HOG and SURF descriptors. Van De Sande et al. (2010) studied the invariance properties and the distinctiveness of colour descriptors based on SIFT and histograms, and used the descriptors for CBIR. One of the major challenges in CBIR is understanding the semantic meaning of images as their local sub-regions varies drastically. To overcome this issue, Pradhan et al. (2021) proposed an approach in which colour and texture features are extracted from regions. Of late, in some works (Zheng et al., 2017; Shakarami and Tarrah, 2020; Gkelios et al., 2021) the local descriptors are aggregated with deep learning based approaches to create more robust descriptors. For instance, Shakarami and Tarrah (2020) proposed a method that combines AlexNet CNN features with the HOG and LBP features to create a descriptor for CBIR and image classification. Finally, they used Principal component analysis (PCA) to reduce the dimension of handcrafted features for faster image retrieval.

The global and local methods have been widely studied in the literature, and recently, after the advent of deep learning, the generic descriptors derived from CNNs proved to be more powerful and efficient than most of the aforementioned methods. In the next section, we present the recent state-of-the-art methods based on deep learning.

2.2. Deep learning applied to image retrieval

In the deep learning algorithms, the features extracted from intermediate layers of CNNs emerged as powerful descriptors (Sharif Razavian et al., 2014; Gkelios et al., 2021; Saikia et al., 2021). With this breakthrough of deep learning in the computer vision domain, the neural activations of a pre-trained network serve as a robust image descriptor in recenet CBIR tasks (Li et al., 2021). For example, the off-the-shelf CNN features of the OverFeat network (Sharif Razavian et al., 2014) served as a descriptor for recognition as well as for retrieval tasks. Krizhevsky et al. (2012) investigated the use of such descriptors and established that neural codes perform competitively even if a CNN is trained for unrelated tasks. Its remarkable performances in various computer vision task has drawn researchers to investigate its vast potential for image retrieval.

Several works (Salvador et al., 2016; Babenko et al., 2014; Li et al., 2017; Yang et al., 2018; Saikia et al., 2021; Zhu et al., 2019; Gkelios et al., 2021) used the neural features extracted from the intermediate layers as descriptors and achieved state-of-the-art results in instance retrieval tasks. Radenović, Tolias and Chum (2018) proposed to fine-tune CNNs for image retrieval by introducing a trainable generalized-mean pooling layer that boosts the retrieval performance. Since the features obtained from CNNs demonstrated good performance, Siméoni et al. (2019) proposed a method known as deep spatial matching for image retrieval which uses image descriptors extracted from CNN activations by global pooling. Similarly, Noh et al. (2017) introduced Deep Local Feature (DELF), also based on CNNs which are trained with image-level annotations on a landmark dataset. Gordo et al. (2017) presented a siamese architecture that produces a global representation of images that is suitable for image retrieval. Wang, Huang, Zhang, Feng, Zhang and Fan (2020) introduced a deep cascaded neural network with deep representation for establishing multi-modal relationships for image retrieval tasks. For medical image retrieval, Cai et al. (2019) proposed a framework using CNNs and hash coding, which adopts a Siamese network. In the same line, Dubey et al. (2019) proposed AlexNet descriptor for biomedical image retrieval, which is computed by max-fusing Rectified Linear Unit (ReLU) feature maps of a pretrained AlexNet, obtained from bit-plane decoded images. Furthermore, Gkelios et al. (2021) investigated the suitability of deep convolutional features of various CNNs, and demonstrated that pre-trained networks can yield results comparable to state-of-the-art approaches.

As the number of images has grown exponentially in the last few years, inefficiency of retrieval systems increased in terms of computation and storage. With the development of deep learning-based approaches, various methods have been proposed to learn hash functions to address this inefficiency. Erin Liong et al. (2015) proposed two hash functions known as deep hashing and supervised deep hashing for learning binary codes. To preserve relative similarities between images, Lai et al. (2015) presented a one-stage supervised hashing method using a deep architecture that generates pairwise hash codes. In most of the deep hashing methods, during discretization, the key category-level information may get lost. In order to address this issue, Lu et al. (2019) introduced a method known as ranking optimization discrete hashing (RODH), that directly generates discrete hash codes. For avoiding information loss, Ding et al. (2020) proposed discriminative dual-stream deep hashing (DDDH). Recently, to learn more effective binary codes, in (Lu et al., 2020), a new hashing method named as DeepFuzzy hashing network was proposed, and Chen et al. (2020) introduced deep learning supervised hashing (DLSH) that learns features and binary codes together.

Furthermore, localized instance search has benefited from the major success of deep learning in the field of object detection (Wu et al., 2020). Particularly, the performance of image retrieval systems improved after the advent of region-based networks (Sermanet et al., 2013; Girshick et al., 2014; Lei et al., 2020; Fan et al., 2020), which were originally aimed at object detection purposes. We next present the evolution of object detectors that can be used to address image retrieval systems.

2.2.1. Deep learning based object detectors

In this section, we review the state-of-the-art object detection algorithms which aid the image retrieval and scene recognition tasks. Object detection is one of the most active areas in computer vision, and these algorithms have seen a remarkable growth in the recent years after the advent of deep learning. Since 2014, new object detectors based on deep learning have been introduced, which outperformed the previous state-of-the-art algorithms. Apart from detecting and labelling objects in images, object detectors play an important role in CBIR and scene recognition. In CBIR, the object detectors can be used to predict regions and extract their corresponding ROI features in images, later a query image can be compared locally with all the regions to verify if the query is contained in the image or not. Also, for the task of scene recognition, the object detectors can be employed to differentiate among various scenes as the objects contained are the main components. We next review some of the most popular state-of-the-art object detection algorithms. In Fig 2.1, we present a road map of evolution deep learning-based object detection algorithms.

In the year 2014, Girshick et al. (2014) proposed region-based CNNs (R-CNN),

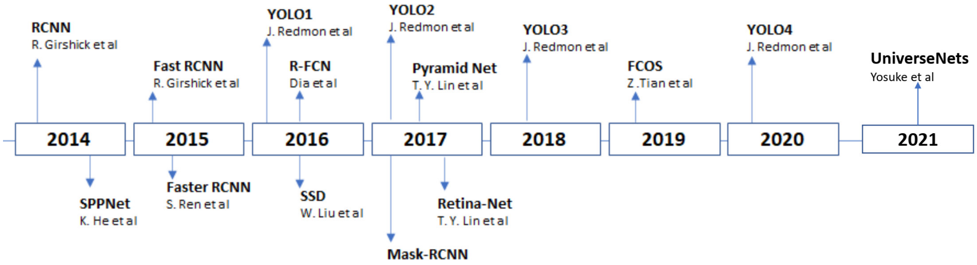


Figure 2.1: Evolution of state-of-the-art deep learning based object detectors from the year 2014 to 2021.

which uses regions with CNN features for object detection. The R-CNN relies on an external region proposal system to generate candidate regions, and uses selective search to create 2000 proposals. Finally, an SVM classifier trained with the CNN features is used to predict the presence of objects in each of the regions. R-CNN yielded a significant performance gain on the Pascal VOC-07 dataset, where it obtained a mean average precision (MAP) of 58.5% as compared to 33.7% reported by DPM. However, even though the R-CNN has made significant progress, it has certain drawbacks. The R-CNN does 2000 forward passes to the CNN, and hence it is computationally expensive, which leads to slow detection of objects. To address this issue, He et al. (2015) introduced spatial pyramid pooling networks (SPPNet), which avoids the repeated computation of the CNN features as R-CNN does. In addition, SPPNet is 20 times faster than the R-CNN and achieves a mAP of 59.2% on the same dataset, which is slightly higher than the R-CNN. Moreover, as compared to previous CNN networks that require fixed-size input images, SPPNet generates fixed-length features irrespective of the size of the region. However, the SPPNet has improved the detection speed, but it has multi-stage training, which makes the network complex, and it only fine-tunes the fully connected (FC) layers while ignoring the other layers. Another drawback with these two algorithms is that they use a selective search algorithm, and therefore no learning happens at that state, which may lead to poor region proposals.

In order to improve the proposed system of R-CNN and SPPNet for faster detection, Girshick (2015) proposed Fast R-CNN. Unlike the other two networks, in Fast R-CNN the object detector and the bounding box regressor can be trained simultaneously. In addition to reducing training time, it achieved remarkable performance gain as compared to R-CNN. The mAP on the PASCAL VOC-07 dataset increased from 58.5% (R-CNN) to 70%, and even the speed of detection is over 200 times faster than R-CNN. In 2015, Faster R-CNN was proposed by Ren et al. (2017), which is the first near real-time object detector based on deep learning with a detection speed of

0.12 seconds per image. Faster R-CNN introduced a region proposal network (RPN) to generate proposals within the network instead of using external algorithms, and also enabled end-to-end training. Moreover, it has broken the speed bottleneck of detection, and the mAP on the VOC-7 dataset increased from 70% to 73.2%. However, FasterR-CNN has computational redundancy, and later various improvements were proposed, such as R-FCN (Dai et al., 2016), feature pyramid networks (FPN) (Lin, Dollár, Girshick, He, Hariharan and Belongie, 2017) and Mask R-CNN (He et al., 2017). In this same line of research, recently, Fan et al. (2020) proposed a network known as few-shot object detection that aims at detecting unseen objects just by using few annotated examples. Similar to R-CNNs, this network also employs an RPN in its background to detect objects. Recently, Shinya (2021) designed object detectors called UniverseNets, which surpassed all baselines and achieved state-of-the-art results on existing benchmarks.

In spite of the great success of the aforementioned two-staged detectors in object detection, they are computationally expensive, hard to optimize and the performances are not real-time. To address these issues, CNNs based on one-stage detectors were proposed, such as the popular (You Only Look Once) YOLO (Redmon et al., 2016), single shot detector (SSD) (Liu et al., 2016) and RetinaNet (Lin, Goyal, Girshick, He and Dollár, 2017). YOLO-v1 was the first real-time one stage detector in the era of CNNs that could be trained end-to-end and can be easily optimized. The network can process 45 frames per second (fps), but it has localization difficulties in detecting smaller objects and makes more localization errors as compared to two-stage detectors. Later, YOLO-v2 (Redmon and Farhadi, 2017) was proposed, improving YOLO-v1 in terms of detection accuracy and attained faster detection speed. Even after attaining great improvement, YOLO-v2 has poor localization accuracy as compared to region-based CNNs. In order to overcome this drawback, (Liu et al., 2016) proposed single shot detector (SSD) with multi-reference and multi-resolution detection techniques. The SSD yields higher detection speed and accuracy as compared to YOLOs, and also it can detect smaller objects with high accuracy. The main difference between the other object detectors and SSD is that the latter can detect objects on different layers with different scales, whereas the other detectors use the top layer features for object detection. However, the inefficiency compared to two-stage object detectors remained. (Lin, Goyal, Girshick, He and Dollár, 2017) identified the reason behind the accuracy lag, which it was said to be due to the imbalance between the background-foreground class during the training process. In order to deal with this issue, the authors introduced a new loss function known as focal loss, which made the SSD achieve a comparable accuracy to the two-stage detectors while maintaining the faster detection speed. In 2018, YOLO-V3 (Redmon and Farhadi, 2018) was introduced with some incremental improvements over YOLO-v2 in terms of accuracy and speed. As compared to the other two previous versions, YOLO-v3 is a 106 layer network consisting of 75 convolutional layers, and it uses 23 residual layers to avoid vanishing gradients while training. In addition, the network is as accurate as SSD and also about three times faster. Recently, YOLO-v4 (Bochkovskiy et al., 2020) has been released which was considered as one of the best models for speed and accuracy. Afterwards, another variant of YOLO-v4 known as Scaled-YOLOv4 was proposed by Wang, Bochkovskiy and Liao (2020). In this scaling approach, the depth, width and resolution of the architecture can be modified while maintaining optimal speed and accuracy. To detect objects of different sizes, the depth and width of the network is dynamically adjusted according to the real-time inference requirements. Besides, Scaled-YOLOv4 obtains the highest accuracy on the MS-COCO dataset surpassing other recent state-of-the-art detectors such as (Tan et al., 2020; Du et al., 2020; Bochkovskiy et al., 2020).

2.3. Scene recognition

In the early years, image representation for scene recognition mostly relied on global attribute descriptors such as GIST (Oliva, 2005), Edge Straightness Analysis (Payne and Singh, 2005), CENsus TRansform hISTogram (CENTRIST) (Wu and Rehg, 2010), local difference binary pattern (LDBP) (Meng et al., 2012) and multi-channel CENTRIST (Xiao et al., 2013). However, the performance of such descriptors is limited when the spatial layout of the images are complex due to the presence of various objects and patterns.

In order to enhance the performance, researchers focused on local descriptors for describing various patches present in scene images. Some of the notable widely used local descriptors for patch feature extraction include LBP, SIFT, HOG, SURF, and BoVW (Csurka et al., 2004). Among them, BoVW was the most popular one and it integrates numerous local descriptor for image representation. Specifically, the local descriptors are quantified into visual words, and a scene image is represented using the frequency of occurrences of visual words. This process of representing an image is regarded as code-book learning, and it has a great impact on the recognition performance. However, such codebooks have certain ambiguities between the visual words. To address this issue, (Van Gemert et al., 2008) proposed kernel codebook, and later, histogram kernel (Wu and Rehg, 2009) was introduced using k-means for effective codebook learning that increases the recognition accuracy. Despite such improvement, such algorithms make large quantization errors and are sensitive to outliers. To overcome the issues, Sparse coding (Yang et al., 2009) and its derivative algorithms (Gao et al., 2010; Wang et al., 2010) were proposed to learn the codebook while reducing the quantization error. In a work proposed by Khan et al. (2016), codebook is learned from spatial patches that represent scenes, where sparse linear coding is applied to convolutional activations. The codebooks are composed of different patches and they achieved state-of-the-art performances for indoor scene recognition. Qin and Yung (2010) incorporated extra information with visual words to create powerful codebooks in which the relation between visual words are exploited for codebook learning. In Zhou et al. (2013) spatial information was combined with the local features extracted from multi-resolution images. Moreover, some deep learning architectures are being exploited for codebook learning, such as autoencoders (Xie et al., 2014) and restricted Boltzman machines (RBM) (Goh et al., 2014).

One of the most classical intuitive ways to exploit the spatial information is by partitioning the image space into a grid cell and then computing the visual features corresponding to each cell, and finally, concatenating them for global image representation. For example, spatial pyramid matching (SPM) (Lazebnik et al., 2006) method using HOG and SIFT proved to be effective in encoding spatial pyramid information which significantly improved the previous BoVW based methods of that time. Moreover, the predefined spatial layout is applied in many methods such as (Lazebnik et al., 2006; He et al., 2015; Wu and Rehg, 2010) to enhance the descriptors. Jiang et al. (2012) proposed randomized spatial partition, which characterizes the scene layout by various patterns. Using this technique, the most descriptive patterns for each category of scenes are considered for training a classifier. Subsequently, Weng et al. (2016) proposed an approach to discover class-specific spatial layouts of scene images based on the convolutional activations corresponding to the spatial partitions. This approach can determine the sparse combination of spatial layouts of different classes, which helps in boosting the recognition performance. In this line of work, He et al. (2015) proposed a randomized spatial pooling layer that incorporates spatial information with a deep CNN, so the partitioning of the features maps using randomized pooling manages to handle various image layouts. For indoor scene recognition, large scale spatial deformations and scale variations are two major challenges. To address them, Hayat et al. (2016) proposed a learnable feature descriptor called "spatial layout and scale-invariant convolutional activations", and in addition, a layer known as spatially unstructured is introduced for the CNN network making the feature robust against spatial deformations.

Region-based object centric approaches: Most of the scene images are defined by some crucial regions, which can be patches or objects. Therefore, the performance of scene recognition can drop if the fine-grained objects are neglected. Some early methods (Li et al., 2010; Pandey and Lazebnik, 2011) used handcrafted features for object detection to determine discriminative regions for scene classifications. In (Li et al., 2010), object bank was proposed, in which an image is represented using a response map of various pre-trained object detectors, and in (Pandey and Lazebnik, 2011) deformable parts model (DFM) was used for object detection to obtain the discriminative regions. Also, there are approaches that attempt to use image patches to identify important discriminative regions. For instance, Singh et al. (2012) experimentally demonstrated that the image patches as a mid-level visual representation are effective for indoor scene classification. Juneja et al. (2013) proposed a method to automatically discover distinct parts, in which they incrementally trained an Exemplar SVM to learn the most informative parts of an image. Later, to get the response of most informative regions, Lin et al. (2014) proposed a method that uses part filters, and in addition, it suppresses the noisy features. Other approaches that exploit the information provided by the distribution of object patterns concerning different scenes are (Song, Jiang and Herranz, 2017; Wu et al., 2015; Song et al., 2016). The common challenge that persists with such scene recognition algorithms is about handling inter-class and intra-class variations. To overcome this issue, (Zuo et al., 2014) introduced a method known as discriminative and shareable features learning (DSFL), which makes the learned features of the same classes closer and the learned features of different classes far away from each other. In addition to this, various methods analysed the correlation between diverse objects and scene categories, For example, inspired by the BoVW model, (Cao and Fei-Fei, 2007) presented a generative model that can simultaneously recognize and segment object and scene classes. Similarly, (Niu et al., 2012) presented a context-aware discriminative latent topic model where the global and spatial context of scene images are jointly modelled. Later, a new latent variable model was proposed. It represents scene images as a collection of regions, which are arranged in a reconfigurable pattern. Some other models that are proposed to leverage the co-occurrence patterns of objects in different scenes are (Wu et al., 2015; Song et al., 2016; Cheng et al., 2018).

Deep learning based models: Later, following the remarkable performance of the deep learning in the field of computer vision tasks, various CNN models proved to be efficient for scene recognition. Cichy et al. (2017) compared CNNs with previous computer vision-based models suggesting that CNNs are the best existing methods for representing spatial information of images. Current CNN based architectures experimented using multi-million datasets, such as Places-365 (Zhou, Lapedriza, Khosla, Oliva and Torralba, 2017), outperforming the results obtained by the handcrafted features. Besides, CNNs integrate low-level information such as colour, shape and texture with high-level information (object parts) which leads to succesful scene recognition methods. Moreover, the most popular deep learning networks such as ALexNet, GoogleNet, VGG-16, ResNet and DenseNEt are used for classifying images using the Places-365 dataset with accuracies of 53.17%, 53.63%, 55.24%, 54.74% and 56.10% respectively. Furthermore, the features extracted from such networks are fused with different methods for creating hybrid models that enhance the recognition performance. The lower layers of CNNs captures local features while the top layers generate more abstract features, and based on this property (Guo et al., 2016; Xie et al., 2015; Tang et al., 2017) fused features corresponding to multi-scale convolutional layers for scene recognition. Similarly, a multi-resolution CNN architecture was introduced by Guo et al. (2016) to capture features corresponding to different layers for scene understanding. Dixit et al. (2015) proposed a semantic Fisher vector to fuse features from convolutional and fully connected layers of CNNs. Cimpoi et al. (2015) used texture information extracted from a CNN and a BoVW model to enhance scene recognition. Yoo et al. (2014) achieved a state-of-the-art performance on the MIT Indoor dataset, Quattoni and Torralba (2009), after aggregating multi-scale CNN based activation using a Fisher kernel framework. For recognizing indoor scenes, Basu et al. (2020) trained a Capsule Neural Network, which obtained better performance as compared to other CNN-based frameworks.

Although the accuracies obtained by deep learning methods are much higher than the handcrafted features, the CNN does not lead to a linear rise in performance due to its inability to handle inter-class similarities (Cheng et al., 2018). To cope with this issue, some methods incorporate context and object information. For example, Wang et al. (2017) introduced a network called PatchNet, which aggregates both the object and holistic scene features to develop an effective visual representation of scenes. Qin and Yung (2010) used context information to enhance recognition by detecting regions based on saliency detection. Zuo et al. (2014) proposed DisNet, which generates a discriminative map of a given input image that is forwarded to a CNN for feature extraction. In such architectures, the lower layers of CNNs often capture the local features, while the top layers generate holistic features. Based on this property, Yang (2015) proposed DAG-CNN that leverages features corresponding to convolutional and fully connected layers for scene recognition, whereas CNN-DL (Liu, Chen, Chen and Wassell, 2018) used sparse coding to transform convolutional features. On the other hand, Wang et al. (2017) proposed VSAD that combines features extracted from Object-patchNet and Scene-PatchNet. Similarly, Seong et al. (2020) introduced FOS-Net that fuses both object and scene information. To obtain a more powerful image representation, a Mix-CNN (Jiang et al., 2019) was trained using objects and scene datasets for codebook learning, that can be shared by Object-CNN and Scene-CNN. Recently, López-Cifuentes et al. (2020) proposed a multi-modal CNN that combines the context information with the scene image using an attention module. In addition, to further improve the recognition accuracy, (Li et al., 2019) introduced MAPnet which fuses depth with the RGB image information. Regarding indoor scene perception for mobile robots, Ran et al. (2021) designed a shallow and efficient CNN structure that attained higher scene recognition accuracy using monocular camera images.

In particular, the indoor scene recognition task is commonly based on detecting indoor objects, and hence most of the algorithms employ object detectors to identify the inherent objects. Compared to the approaches described above, our proposal exploits the frequencies of objects of a scene by generating a caption using the class labels of the detected objects. The caption is then encoded into a vector which is fed to a small neural network for scene prediction. In addition, we emphasize on the global scene content for scene prediction, and we aggregate both the local –based on objects– and the global predictions using a weight function. As described in the literature (López-Cifuentes et al., 2020; Xie et al., 2020; Ran et al., 2021), methods focusing on increasing the network layers of CNNs do not lead to a linear performance gain. This is due to the inability of networks to handle the diversity in inter-class similarities (Cheng et al., 2018). In essence, our proposed work in this thesis enhances indoor scene recognition without the need of huge training sets and it does not need to rely on very deep networks.

Chapter 3

Colour neural descriptors for instance retrieval

This chapter [1] addresses the image instance retrieval task by creating discriminative features using CNNs and colour models. The instance retrieval aims at retrieving all the images from a corpus of a large dataset that contain the same object instance as the query. In Fig. 3.1, we show few samples where the instances are retrieved based on a query image. Unlike most of the previous works that employ fine-tuning and require new training for instance retrieval, we create new robust descriptors using several colour models without training and fine-tuning. We used the VGG-16 network pre-trained on the ImageNet dataset, Deng et al. (2009), to generate the neural activations from the last fully connected (FC8) layer. Those activations are generated concerning three colour channels, Red (R), Green (G) and Blue (B), present in an RGB image. Instead of directly generating activations of an image, in our approach, we generate neural features for each of the colour channels - R, G and B- separately and we further pass them through a colour neural descriptor generation (CDG) layer to construct the proposed colour neural descriptors. In addition, this chapter further discusses about the scalability of the proposed approach in terms of computational and space complexity.

In Fig. 3.2, we briefly outline our complete query-based instance retrieval approach. To find a query instance present in an image, it is necessary to check the entire image, part by part, to check for its presence. For this purpose, an object detector is necessary to generate rectangular proposals of various sizes. To determine if the query region is present in the image, the descriptors of each of the proposals need to be compared against the query image descriptor. In order to do that comparison, neural features of the images are extracted as descriptors and the comparison is done using a distance metric. If the computed metric is higher than a given threshold, we retrieve that particular instance assuming that the query is present in that image. When addressing datasets with multi-view or rotated objects, the retrieval task becomes even more challenging. Given that asymmetrical objects may appear rotated in images of the dataset, most methods would possibly fail to retrieve rotated views of the same object in which the appearance is remarkably

[1] This chapter is “reproduced with permission from IEEE Access”, which is adapted from S. Saikia, L. Fernández-Robles, E. F. Fernández and E. Alegre, ”Colour Neural Descriptors for Instance Retrieval Using CNN Features and Colour Models,” in IEEE Access, vol. 9, pp. 23218-23234, 2021, doi: 10.1109/ACCESS.2021.3056330.

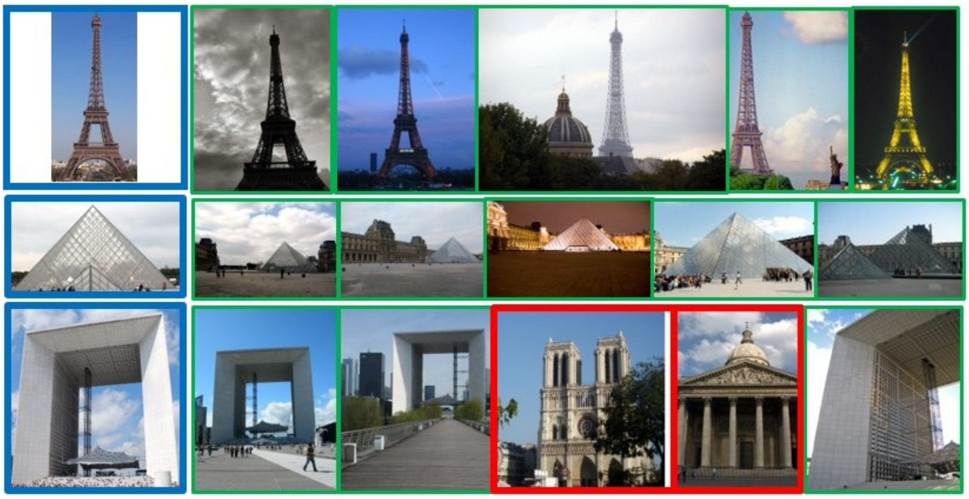


Figure 3.1: Three examples of instance retrieval. For each row, the first image on the left represents the query image to look for in a dataset, in this case Paris dataset. The rest of the images represent the hit list of retrieved images sorted according to a similarity metric to the query image. Images framed with a green rectangle correspond to correct retrievals whereas a red rectangle depicts the incorrect retrievals. These examples show the actual results of the proposed method.

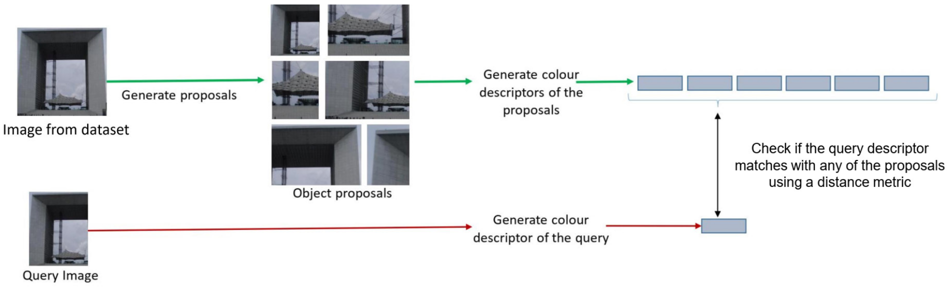


Image from dataset
Generate proposals
Object proposals
Generate colour descriptors of the proposals
Check if the query descriptor matches with any of the proposals using a distance metric
Query Image
Generate colour descriptor of the query

Figure 3.2: Overview of our proposed content-based instance retrieval framework using colour neural descriptors. The green line indicates the generation of descriptors for each of the proposals of an image from the dataset, and the red line represents the generation of the query descriptor.

different from the query. Concerning the challenges related to the same instance retrieval task, we investigate different colour models and intermediate features of CNN architectures to develop robust descriptors.

To sum up, the main contributions in this chapter are the following ones:

- We introduce new colour neural descriptors, based on the activations generated from a pre-trained Deep CNN.
- We present a hybrid architecture composed of two different CNNs, and we use it successfully as the instance retrieval pipeline without employing fine-tuning techniques.
- We demonstrate experimentally that our proposed colour neural descriptors outperform the state-of-the-art in four datasets for image retrieval, COIL-100, INSTRE-M, Paris 6K and Revisiting-Paris 6k.

3.1. Method

In this section, we present a novel approach that enhances visual search for instance retrieval using colour neural descriptors and bounding boxes predicted by an object detector. In particular, our method builds on top of the object proposals and activations generated from a pretrained CNN algorithm. We first explain a preliminary approach for instance retrieval that uses the neural activation generated by a pretrained CNN. Afterwards, we present the backbone of the architecture and the proposal for instance-based retrieval using colour neural descriptors. Finally, we describe the overall query-based instance retrieval method.

3.1.1. Preliminary method for instance retrieval

To demonstrate the efficacy of the neural features, we first proposed a method using the Faster R-CNN (Ren et al., 2017) architecture trained with MS-COCO dataset for retrieving instances (objects in this case). In this approach, the neural codes can be obtained by passing images through the Faster R-CNN network. At the top of the chosen architecture, there are three fully connected layers, and we use the neural codes generated by the last layer as a visual descriptor for detecting objects. Since the last layer has 80 output units, our descriptor is an 80-dimensional vector. Now, for each detected object, we save its patch (region cropped from the image), bounding box coordinates and confidence score along with the neural codes in a database. Finally, to retrieve similar images based on a given query, the neural codes of the query are computed and then compared against the database of neural codes using a distance metric and a confidence score. In Fig 3.3, we illustrate this preliminary method for instance retrieval. With this background, in our next proposal, we enhance the baseline architecture by replacing Faster R-CNN with the R-FCN (Dai et al., 2016) network, and we apply colour models to make the neural features robust and discriminative. Besides, the underlying architecture for the neural feature extraction step is the same as the backbone architecture which is discussed in Section

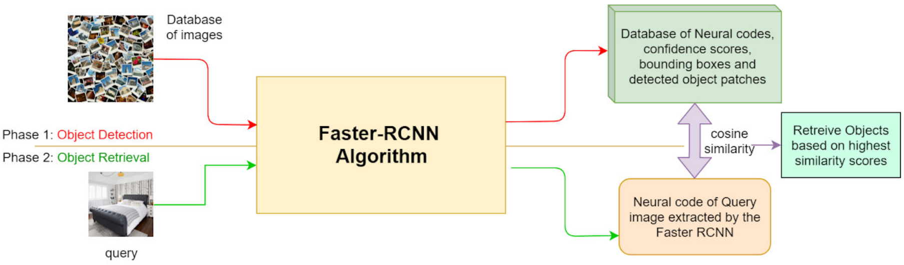

Database of images
Phase 1: Object Detection
Phase 2: Object Retrieval
query
Faster-RCNN Algorithm
Database of Neural codes, confidence scores, bounding boxes and detected object patches
cosine similarity
Retreive Objects based on highest similarity scores
Neural code of Query image extracted by the Faster RCNN

Figure 3.3: Instance retrieval pipeline using Faster R-CNN.

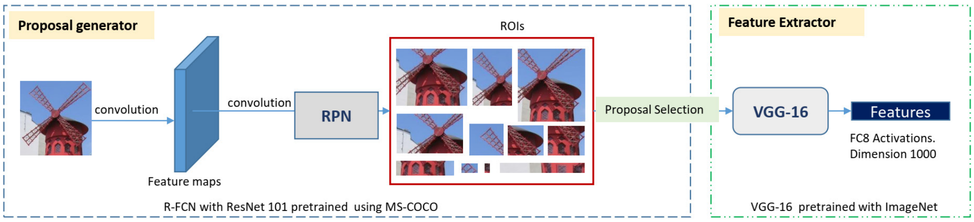

Proposal generator
convolution
Feature maps
convolution
RPN
ROIs
Proposal Selection
R-FCN with ResNet 101 pretrained using MS-COCO
Feature Extractor
VGG-16
Features
FC8 Activations. Dimension 1000
VGG-16 pretrained with ImageNet

Figure 3.4: Backbone architecture of the instance retrieval approach. It is constituted in two parts: the region proposal network (RPN) from the R-FCN for proposal generation and the VGG-16 network for feature extraction. The proposals are generated by the RPN, which are then given as an input to the VGG-16 network for region-based feature extraction.

3.1.2.

3.1.2. Backbone architecture

The backbone architecture comprises two different pre-trained networks that facilitate the local instance search and the creation of discriminative descriptors. For local instance search, we use R-FCN to generate proposals on the dataset images in order to compare the query instance against those proposals. In contrast, VGG-16 serves as a feature extractor for both the query image and the proposals. Both networks serve as a single framework, where the candidate proposals generated by the R-FCN network are given directly as an input to the VGG-16 network to compute colour descriptors. In Fig. 3.4 we illustrate the architecture of the proposed method, which consists of the following two major stages.

- Generation of object proposals for regional search using R-FCN.
- Deep feature extraction using a VGG-16 network.

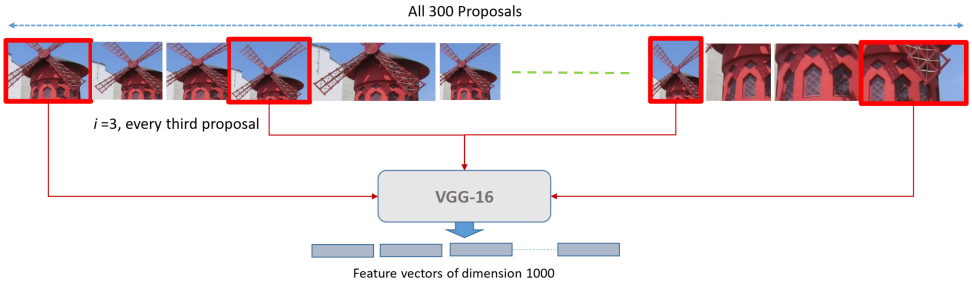


Figure 3.5: Detailed proposal selection. Every i^{th} proposals are selected out of the 300 proposals generated by the RPN to search the presence of query instance in a given image. In this case, we set $i = 3$ which results in 100 proposals per image. We next pass the proposals through the VGG-16 to obtain local features each of dimension 1000.

Generation of object proposals

To find a queried object or instance in an image, it is required to first detect and localize all possible objects for matching the query image features with each of the localized object features. The more similar the features are, the more likely the query and the proposals are the same object.

In our approach, to search a query instance locally, we use the object detector of the R-FCN network to generate object proposals. R-FCN is faster than other region-based CNNs, such as Fast or Faster R-CNN, because it derives region proposals (ROIs) from the feature maps directly. In R-FCN, the RPN generates the object proposals using convolutional features maps, but unlike Fast and Faster R-CNN, the fully connected layers after the ROI pooling are removed and hence no learnable layer is required after the ROI layer. As a result, R-FCN is up to twenty times faster than Faster R-CNN with a competitive mAP, and that is the reason why we chose this architecture to generate region proposals. The total number of proposals obtained by the object detector is around 300, with lots of overlapping boxes covering the same object. Therefore, to reduce this cost, we define a set of candidate regions per image by empirically selecting every i^{th} proposal, with i= 3 in this case, see Fig. 3.5. Moreover, we store the proposals and their corresponding descriptors in a database to be used for our image retrieval system.

Deep CNN Features (DCFs) extraction

In order to create colour neural descriptors, we extract DCFs from the VGG-16 network pre-trained on the ImageNet dataset. We use the last fully connected layer (FC8) which contains 1000 neurons, resulting in a feature vector of 1000-D. In particular, the activations from the hidden layers represent low-level features, such as edges and contours, and the higher layers produce abstract features that fully represent images. Hence, we prefer to extract the DCFs at the penultimate layer. However, to generate colour neural descriptors, we extract the features corresponding to the three different colour channels (R, G and B). We represent the DCFs obtained using R channel as R^*, G as G^* and B as B^*, which we use to obtain the colour neural descriptors.

3.1.3. Colour neural descriptors

In this section, we first introduce the intuition behind the feasibility of colour neural descriptors, later, we explain how DCFs are generated using the colour channels, and finally, we present the proposed colour neural descriptors.

Intuition behind colour neural descriptors

In some situations, the colour plays an essential role in obtaining visual information about objects present in images. Our idea is to leverage that information to create high-level discriminative colour feature vectors. An RGB image is composed of three channels, and the absence or presence of anyone would change the neural activations generated from an image.

For instance, in Fig. 3.6 we have a query representing a red box along with two other images: Image A is identical to the query and Image B differs only in the colour, which is yellow. First, for each image, we extract the DCFs of each colour channel, –R^*, G^* and B^*–, and then we concatenate them to create colour neural descriptors of the image. Next, we compute the similarity between the descriptor of the query image and the other two descriptors extracted from images A and B. Image A stands out in terms of similarity as compared to B, because the red-box on image A is similar to the query with respect to colour, and hence their respective colour neural descriptors are similar. Therefore, the channel-based activation of objects that have the same colours and textures are identical. As a result, colour neural descriptors made by deriving activations from individual channels and concatenating them are able to discriminate colour better. In this work, we present and evaluate different ways of fusing the DCFs obtained with respect to the colour channels to propose the colour neural descriptors.

Colour neural descriptor generation layer

We define a colour neural descriptor generation (CDG) layer, in which we obtain colour neural descriptors from the DCFs extracted for each specific input colour channel passed to the network. In order to obtain a robust colour neural descriptor, we evaluated different colour spaces and combinations of colour channels, inspired by the work of Van De Sande et al. (2010). Next, we present the different descriptors,

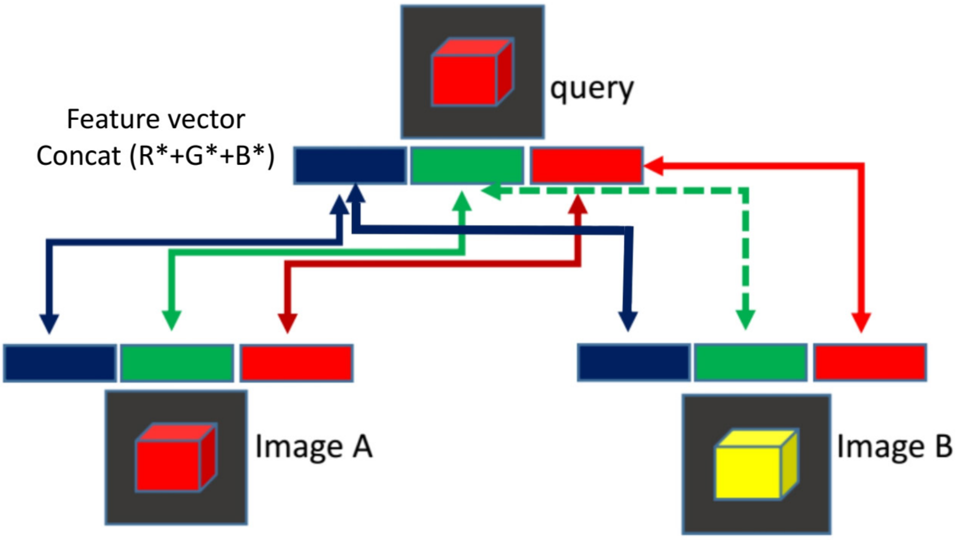


Figure 3.6: This figure visually illustrates a simple case of colour neural descriptors representation and how it affects object description. Both the query image and image A contain a red box whereas image B contains a yellow box. The query image and Image A represent similar objects, and hence they have similar colour neural descriptors with respect to every colour channel. In the case of Image B, the green descriptor differs with respect to the query image. The weak resemblance between colour neural descriptors are represented by dashed lines whereas high resemblance is marked with solid lines.

and based on our preliminary tests, we chose the one that we consider more appropriate for the retrieval problem. Consequently, the CDG layer creates the colour models, which transforms the DCFs into colour neural descriptors.

NE-Raw. The NEural Raw (*NE-Raw*) descriptor is generated by passing an image through the network without any modification of the input layer. We directly extract the activation with respect to the FC8 layer of the VGG-16 network without letting it pass through the CDG layer (we do not apply colour models). This descriptor posses no invariance to colour apart from the one conferred by the network. We use this descriptor mainly as a baseline, for comparison purposes against the other descriptors.

NE-O and NE-O3. *NE-O* represents the descriptor obtained using opponent colour space (Eq. 3.1), which is a combination of DCFs based on the channels of the opponent colour space. In the Eq. 3.1, the intensity information is represented by channel O3 and the colour information by O1 and O2. Due to the subtraction, the offsets are canceled, and hence, the descriptor is invariant to changes in light intensity. The *NE-O* descriptor is constructed as the concatenation of O1, O2 and O3. Based on our preliminary tests, in some cases, results obtained with just O3 feature vector as colour neural descriptor outperformed the combination of all the three components (O1, O2 and O3). We name this O3 feature vector as the *NE-O3* descriptor.

$$\left\{ \begin{array}{c} O1 \\ O2 \\ O3 \end{array} \right\} = \left\{ \begin{array}{c} \dfrac{R^* - G^*}{\sqrt{2}} \\ \dfrac{R^* + G^* - 2B^*}{\sqrt{6}} \\ \dfrac{R^* + G^* + B^*}{\sqrt{3}} \end{array} \right\}. \tag{3.1}$$

NE-TCD (Transformed Colour Distribution). In general, *NE-Raw* is not invariant to changes in lighting conditions. However, by normalizing the pixel value distributions (Eq. 3.2), shift invariance can be achieved with respect to changes in illumination. Since each channel is normalized independently, the descriptor is also robust to changes in colour intensity and arbitrary offsets. In (Eq. 3.2), μ_C is the mean and σ_C is the standard deviation of the colour distribution in channel C computed over the area under consideration (e.g. a patch or an image). This yields for every channel a distribution where $\mu = 0$ and $\sigma = 1$. At the CDG layer R', G' and B' are computed and concatenated to form *NE-TCD* colour neural descriptor.

$$\left\{ \begin{array}{c} R^{'} \\ G^{'} \\ B^{'} \end{array} \right\} = \left\{ \begin{array}{c} \dfrac{R^* - \mu_{R^*}}{\sigma_{R^*}} \\ \dfrac{G^* - \mu_{G^*}}{\sigma_{G^*}} \\ \dfrac{B^* - \mu_{B^*}}{\sigma_{B^*}} \end{array} \right\}. \tag{3.2}$$

NE-C. We created this descriptor by passing the DCFs R^*, G^* and B^* of the three colour channels through the CDG layer. The resultant descriptor is a concatenation of the neural features corresponding to those colour channels, i.e R^*+G^*+B^*.

3.1.4. Instance search and retrieval

In this section, we present the proposed instance retrieval method based on colour neural descriptors. Fig. 3.7 illustrates the three-stage pipeline of our approach: (1) Dataset feature extraction, (2) Query feature extraction, and (3) Retrieving and ranking the top-k instances based on a similarity score.

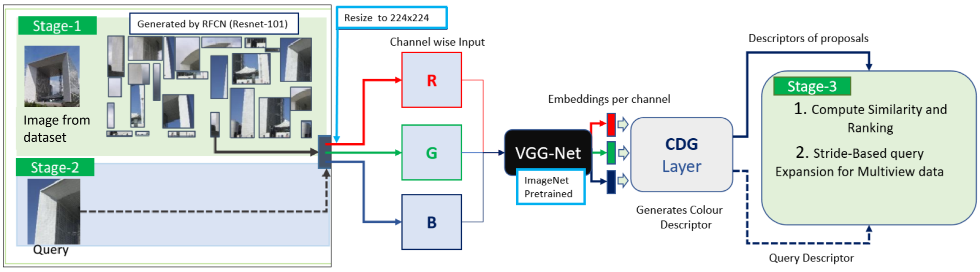


Figure 3.7: An illustration of object proposal generation for a query search. *Stage-1*: Generation of multiple object proposals using RPN. *Stage-2*: Extraction of query features to match with each of the proposals. *Stage-3*: Computing similarity and ranking instances, and for multi-view data we employ query expansion technique.

Dataset feature extraction

First, we process all the images in the dataset to calculate the descriptors, which are necessary for retrieving the images that contain objects similar to the queried one. Let $H = [H_1, H_2, ..., H_n]$ be the set of images, we process each image H_I and generate M region proposals for each of them. The number of proposals depends on the proposal selection criteria as mentioned in Section 3.1.2, where we select every i^{th} proposal to reduce the computation complexity. Then, we resize the proposals to 224×224 pixels and we extract the DCFs with respect to those regions. Next, we pass them to the CDG layer to create the colour neural descriptors as explained in Section 3.2.3. In Fig. 3.7, *stage-1* illustrates how the image proposals are extracted from the dataset.

Query feature extraction

Given a query instance H_q, the DCFs for each colour channel are extracted, and the colour neural descriptors are obtained as explained in section 3.1.3. In Fig. 3.7, *stage-2* shows the query feature extraction process.

Retrieving and ranking using cosine similarity

We aim at retrieving the images on the dataset that are the most similar to the query instance, sorting the retrieved list in descending order according to a similarity measure. First, we compute the similarity between the query instance H_q and the proposals of all images H of the dataset, where m_i is a proposal of the image H_I. Then, we create a hit list by sorting the images of the dataset in descending order, considering the similarity of every image as the highest similarity of any of

Table 3.1: Dimensions of each of the descriptors.

NE-Raw	NE-C	NE-O3	NE-O	NE-TCD
1000	3000	1000	3000	3000

its proposals and discarding images whose similarity is lower than an established threshold (Eq. 3.3).

$$S(H_q, m_i) = \begin{cases} > 0.75, & \text{retrieve } H_I. \\ else, & \text{discard.} \end{cases} \tag{3.3}$$

In order to retrieve only images with a high probability of being similar to the query, we determined experimentally the selected threshold, t. We used a sample set of images from the Outex dataset Ojala et al. (2002a), which is an image retrieval dataset containing texture patterns. We evaluated four different values, $t = [0.60, 0.75, 0.80, 0.90]$, with 10 queries and selected $t = 0.75$ because it was the value that returned consistently related images. Other values yielded a much smaller or bigger number of retrievals, what we considered less appropriate because a higher number of retrieved images increases the computational time to evaluate possible matches, and a lower value leaves out some potential candidates. We use the cosine similarity, see Eq. 3.4, to evaluate the similarity between the query image and each object proposal because it is one of the most commonly used metrics for image retrieval. If the computed score, $CosSim$, is higher than the threshold t, then we include the proposal in the hit list.

$$CosSim(H_q, m_i) = \frac{\sum_{j=1}^{d} H_{q_j} m_{i_j}}{\sqrt{\sum_{j=1}^{d} H_{q_j}^2}\sqrt{\sum_{j=1}^{d} m_{i_j}^2}}, \tag{3.4}$$

where H_q is the query image, m_i is the i_{th} proposal, and d is the dimension of the colour neural descriptors, see Table 3.1.

3.1.5. Stride-based query expansion (SBQE) for multi-view data

Multi-view datasets contain objects captured from various points of view, and hence it is difficult to retrieve all of them using a single query. We implement a query expansion technique to retrieve such multi-view objects in a cascading way. The pseudo-code is shown in Algorithm 1. As an example, let us take a query representing an object with 0-degree rotation, and a dataset of images containing the same object, but with different viewpoints produced by several degrees of rotation (see

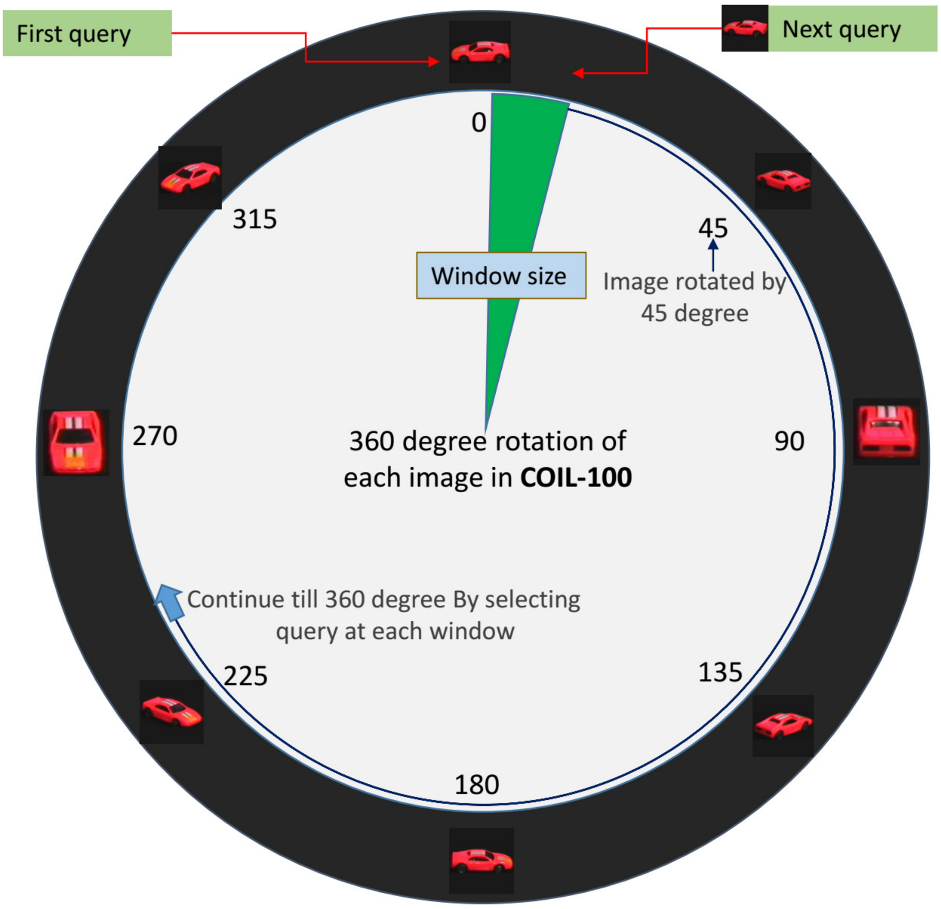

First query
Next query
0
45
90
135
180
225
270
315
Window size
Image rotated by 45 degree
360 degree rotation of each image in **COIL-100**
Continue till 360 degree By selecting query at each window

Figure 3.8: Query expansion applied to COIL dataset. The red cars are rotated from 0 to 360 degrees with an interval of 5 degrees, and the car at degree 0 is the initial selected query.

Fig. 3.8). Therefore, using the non rotated object as a query, probably we will only be capable of retrieving images with close rotations, around ± 45 degrees with respect to the original one, which correspond to rotations in the range $[315, 45]$ degrees. The rest of the images related to the query object presumably will be discarded due to high variations in their appearance caused by the rotation.

Hence, if we expand the query by considering, for example, the s^{th} image retrieved in the hit list, let us consider the one rotated by 5 degrees as the next query image, then we could retrieve images with the object rotated from 310 to 50 degrees. The maximum number of images retrieved with respect to each query is based on the size of the stride s, and we will select the s^{th} image as the next query to retrieve the next subsequent images. We realised that the s^{th} image could be of any degree or even might not belong to the same class as the query image. When selecting the s^{th} image to be the next query, a false retrieval may have a negative cascading effect and we may end up retrieving undesirable images. In order to avoid that, we decided to use a small window with a stride of $s = 3$.

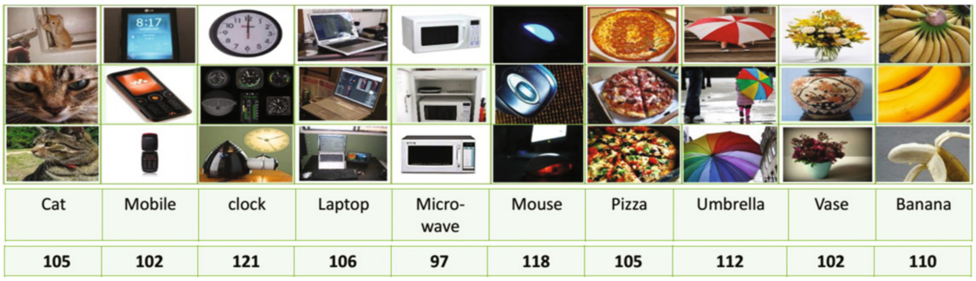
Algorithm 1 Stride-based query expansion (SBQE) to retrieve objects from multi-view datasets

```
Input: query image $H_q$, stride size $s$ and $k$ number of retrievals
Output: top-$k$ instances
while length of list (L) $<$ k do
    Extract colour neural descriptor of the query image $Hq$
    Compute CosSim score between colour neural descriptors of the query and dataset images
    Select images with CosSim score $> 0.75$ and sort them in terms of highest similarity with the query
    Append $s$ number of images to a list $L$ by removing the duplicates if present
    if lenght-of-list(L) ==$k$ then
        return top-$k$ instances;
    else
        $H_q$ = last image in the list $L$;
    end if
end while
return top-$k$ instances
```

In Fig. 3.8, we present the algorithm with a visual explanation. Since we are going to work on COIL-100 with multi-view images, we illustrate how it works using an example taken from this dataset. In COIL, the images have a viewpoint ranging from 0 to 360 degrees, which makes difficult to retrieve all the related images with a single query. In Fig. 3.8, we can see several cars belonging to the same class. Let the car at 0 degrees be the initial query, and let us consider that we want to retrieve the top-k similar images. In this case, we could select the last instance from the retrieved list, the image rotated by 10 degrees in the window, to be the next query, and we will continue doing the same until the list of retrieved instances contains k images.

3.2. Experiments and results

In this section, we present the experiments and the results obtained by evaluating our approach in four standard datasets.



Cat	Mobile	clock	Laptop	Micro- wave	Mouse	Pizza	Umbrella	Vase	Banana
105	102	121	106	97	118	105	112	102	110

Figure 3.9: Sample images from the dataset ImageNet-IndoorObjects along with the total number of images per class. From left to right: cat, mobile, clock, laptop, microwave, mouse, pizza, umbrella, vase and banana.

3.2.1. Datasets

ImageNet-IndoorObjects: To test the proposed preliminary method, we have created a dataset, which is a subset of the ImageNet dataset and comprises objects related to indoor environments. We named the dataset as ImageNet-IndoorObjects, and it is a collection of 1,078 images with 10 object categories. Fig. 3.9 shows some examples of the created dataset along with the total number of images in each class.

COIL-100: Columbia object image library (COIL-100) consists of 7,200 colour images of 100 objects class with 72 images per class. The dataset was created by placing objects in a motorized turn against a black background and were rotated from 0-360 degrees in intervals of five degrees to vary the object pose with respect to a fixed camera.

Paris 6K: This dataset consists of 6,412 still images of Paris landmarks or buildings collected from *Flickr*, which includes 55 query images of 11 buildings. Furthermore, it contains a diverse collection of class-specific images, which differ in terms of illumination, viewpoint, size and resolution.

Revisiting Paris 6K: This dataset is an updated version of the Paris 6k dataset, which is published after correcting some of the annotation mistakes that were present in the original one. There are a total of 6,332 images and 70 query images. The dataset contain the same images present in the original Paris 6k dataset, but the query images are removed.

INSTRE-M: It is an instance level object detection and retrieval dataset consisting of 5,000 images of 50 classes with 101 images per class. It presents multiple appearances of the same object in each of the 101 images with respect to the class

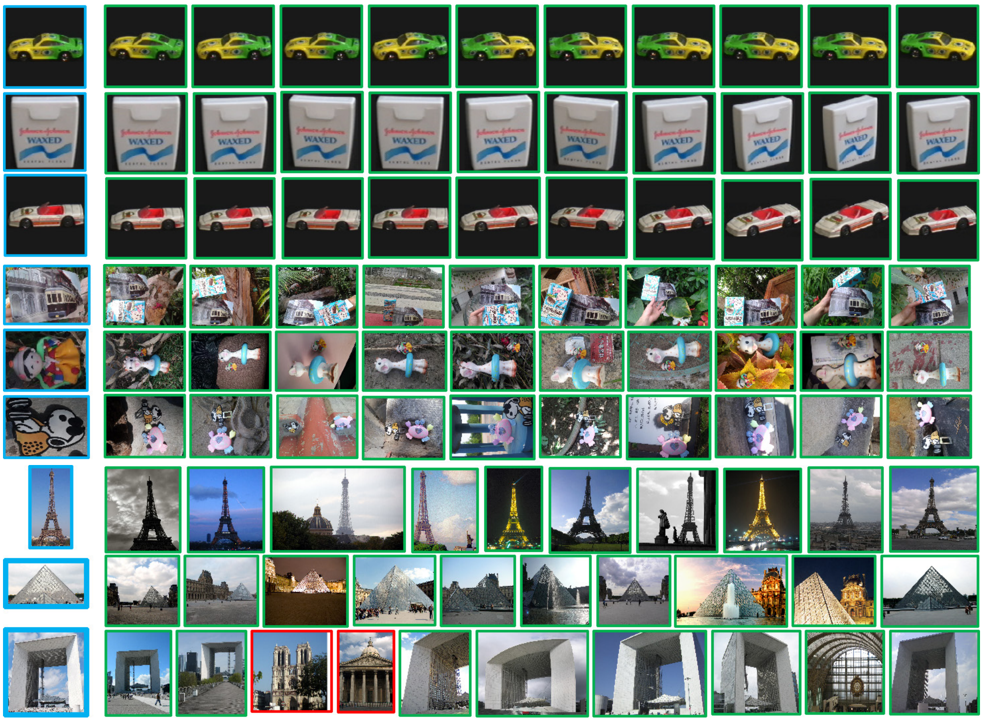


Figure 3.10: Top-10 retrieved images of COIL-100 (rows 1-3), INSTRE-M (rows 4-6) and Paris 6K (rows 7-9) datasets. The above results were obtained by using the colour neural descriptors that achieved the best results on each dataset. The queries are framed with blue rectangles, correctly retrieved images with green ones and incorrectly retrieved images with red rectangles.

category, and hence it is very suitable for instance level retrieval.

A sample of images from the three datasets is presented in Fig. 3.10, where three different queries from each dataset and the top-10 related images retrieved are shown.

3.2.2. Evaluation criteria

To evaluate the efficacy of our proposed colour descriptor we used standard evaluation protocols. We calculated the mean average precision (mAP) to measure the performance in all the experiments. First, we computed the average precision (AP), and then the APs for all the queries are averaged together to obtain the mAP. Eq. 3.5 defines AP, where $P(i)$ is the precision at the cut-off value i, k is the total number of retrieved images which are ranked according to their similarity scores, and $IsRelevant(i)$ is an indicator function which equals 1 if the retrieved image at rank i is relevant, and 0 otherwise.

$$AP = \frac{\sum_{i=1}^{k}(P(i) \times IsRelevant(i))}{k} \tag{3.5}$$

Then, we calculated the mAP given by Eq. 3.6 where Q_N is the total number of queries.

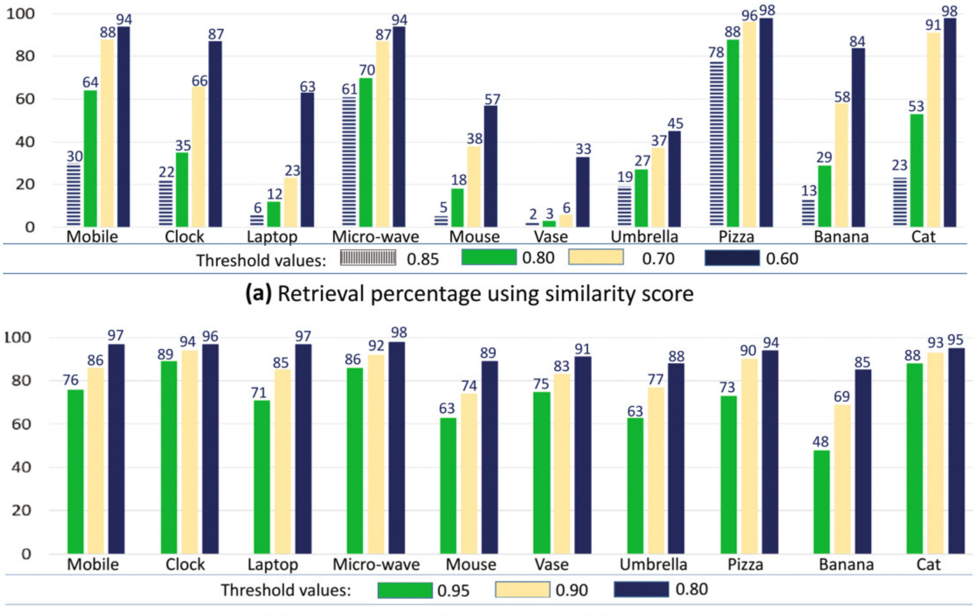
$$mAP = \frac{\sum_{q=1}^{Q_N}(AP(q))}{Q_N} \tag{3.6}$$

3.2.3. Experimental setup

Preliminary experiments

The preliminary experiment for instance retrieval was performed on the Imagenet-Indoor dataset to demonstrate the representative power of CNN deep features. In the first step, we fed the dataset images to the Faster R-CNN network and we stored the generated outputs in a database, i.e. neural codes, confidence scores, bounding box coordinates and detected object patches. While creating our database, we observed the detection accuracy for each object class in the dataset, where we define hit-rate as the percentage of true positives in each object class. We also measured the time taken to store all the information in the database i.e. cropping and storing detected objects, neural codes, confidence scores and bounding boxes coordinates. The total time taken to process all the 1078 images was 328.95 seconds. Next, the system was fed with a query image to retrieve objects from the database by using both the cosine similarity and the confidence score. On the one hand, we computed the cosine similarity between the extracted neural codes and the database of neural codes, and then we retrieved objects based on four different threshold values (0.60, 0.70, 0.80 and 0.85), which were selected empirically. Fig 3.11 (a) shows the hit-rate of objects retrieved at those threshold values. The average hit-rate was 75.3% for threshold 0.60, and the total time taken to retrieve all the 1,454 objects was 0.534 seconds. On the other hand, the objects were retrieved using the saved confidence scores where we chose three thresholds: 0.95, 0.90 and 0.85, and we retrieved objects based on these values. Fig 3.11 (b) shows the hit-rates for each object category, being the average hit-rate 93.5% with threshold 0.80.

Comparison between cosine similarity and confidence score based retrieval: We compared both the cosine similarity and the confidence score based retrieval techniques with an illustration. We selected three query images containing the object "tie", and for each query, we retrieved ten images with the highest cosine similarities and confidence scores. It is observed that, using cosine similarity the resulted



(a) Retrieval percentage using similarity score

(b) Retrieval percentage using confidence score

Figure 3.11: (a) Hit-rate for the object retrieval task in each class using cosine similarity with thresholds 0.60, 0.70, 0.80 and 0.85. (b) PHit-rate for the object retrieval task in each class using confidence score with thresholds 0.80, 0.90 and 0.95.

images are relatively similar to the query images. Fig 3.12 (a) shows the samples of the retrieved objects using the cosine similarity, in which query 2 is a cropped version of query 1. Now, if we consider the confidence score for retrieval, then the algorithm first determines the class label “tie”, and it returns images containing objects with the same class name. Fig 3.12 (b) shows the 10 objects with the highest confidence scores, and we observed that these same sets of objects were retrieved for any of the three query images considered. In Fig. 3.13, we illustrate this comparison by using two plots, for which we have selected a single image query (query 1). This illustration is to demonstrate the worst case scenario when we try to retrieve objects using confidence or probability score. Fig 3.13 (a) represents the 20 objects retrieved with the highest cosine similarity scores along with their corresponding confidence scores. Similarly, Fig. 3.13 (b) represents the 20 objects with the highest confidence scores along with their similarity scores. We can observe in Fig. 3.13 (b) that few objects which were retrieved using the confidence metric have a cosine similarity score below the minimum threshold value of 0.60. This means that using the confidence score for retrieval, we may end up retrieving objects which are not similar to the queried object. Moreover, since we were able to retrieve similar objects only by using the cosine similarity, we use this metric in our approach. In particular, these experiments demonstrated the efficacy of the neural features, and considering these findings as a baseline, we have developed colour descriptors using neural features which we employ in the next sets of experiments.

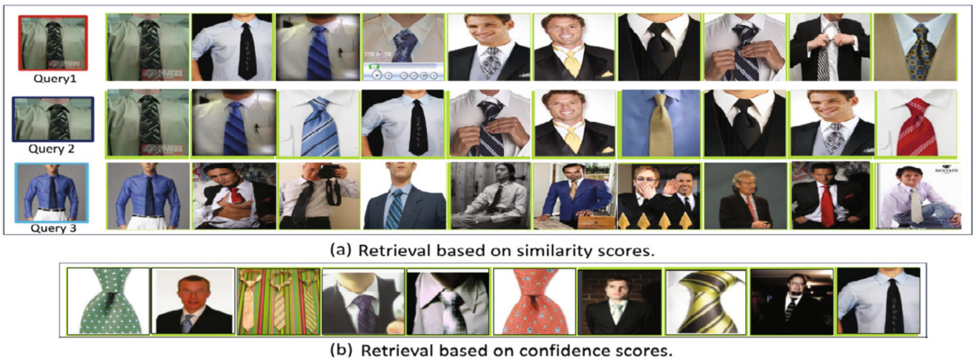

Query1
Query 2
Query 3
(a) Retrieval based on similarity scores.
(b) Retrieval based on confidence scores.

Figure 3.12: Retrieval of a query object with label "tie" using (a) cosine similarity, and (b) confidence score.

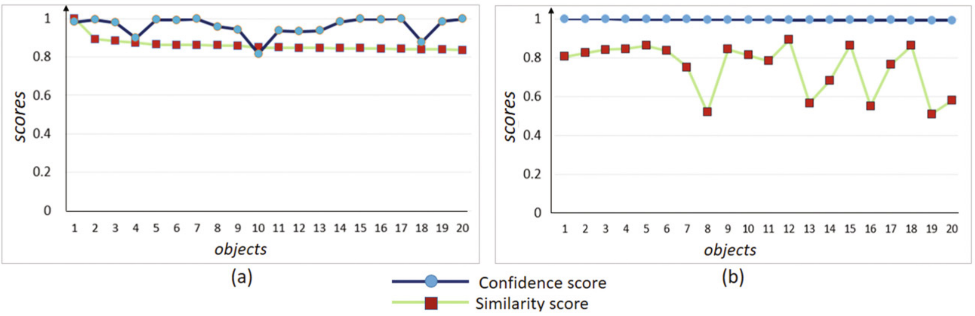


Figure 3.13: (a) Retrieval of the top 20 objects using similarity score. (b) Retrieval of the top 20 objects using confidence score.

Experimental setup for Paris 6k, Revisiting Paris 6k and INSTRE datasets with colour neural descriptors

For our experiments, we extracted 1000-D features from the FC8 layer of the VGG-16 Network and we used the RPN to generate object proposals. All the experiments were done using TensorFlow (version-1.14.0) framework in a Nvidia Ge-

Table 3.2: Comparison of various state-of-the-art methods with our architecture on the Paris 6k dataset in terms of precision@10. The highest precision (96 %) is obtained with the proposed method.

Methods	Precision@10
R-FCN	62
Faster R-CNN	67
FCOS	68
NE-Raw (20)	83
NE-Raw (100)	84
NE-C (20)	90
NE-C (100)	96

force GTX 1060 GPU machine with 16GB RAM and IntelCore processor (i7-7700HQ-2.80GHz). The programming language used for carrying out all the experiments is Python3.6 with CUDA support. For the efficient storage of the descriptors and a faster retrieval, we used the HDF5 binary file format.

For determining the effectiveness of the proposed baseline architecture, we compared the performance considering a different number of proposals with some state-of-the-art region-based CNNs: fully convolutional one-stage object detection (FCOS) Tian et al. (2019), Faster R-CNN with VGG-16 and R-FCN with ResNet. In FCOS, the proposal number varies, whereas, in FasterRCNN and RFCN, we extract features corresponding to the 300 proposals generated by them. We measured the performances in terms of *precision@10* given by Eq. 3.7, where R represents relevance, and is set to 1 if the i^{th} retrieved image contains the query image or 0 in another case. The highest precision of 96% was obtained with *NE-C* with 100 proposals as it can be seen in the Table 3.2 compared to the other approaches. This demonstrates that the proposed architecture can achieve state-of-the-art results even with a lower number of proposals per image.

$$Precision@10 = \frac{\sum_{i=1}^{10} R(i)}{10} \tag{3.7}$$

We also measured how the $precision@10$ changes depending on the different number of proposals used, to gain insight into performance versus relative time trade-off. We define relative time in a range from 0 to 100, which comprises all the steps required, from the extraction of the descriptor up-to retrieval. The value 100 represents the maximum time taken by the descriptor. When the relative time of a descriptor is 50, it means that the descriptor is $2\times$ faster. Whenever the number of proposals increased, the precision obtained was higher but it came with a cost concerning the computation time. As illustrated in Fig. 3.14, while we considered 100 proposals per image we obtained an mAP of 96%. If the proposals are reduced to 20, we obtained a precision of 90% but $5\times$ faster. This experiment was done to illustrate that with a lower number of proposals we can have a faster retrieval framework when speed is the main concern. In order to carry out the rest of the experiments, we selected 100 proposals per image to ensure a good performance at a reasonable computational cost.

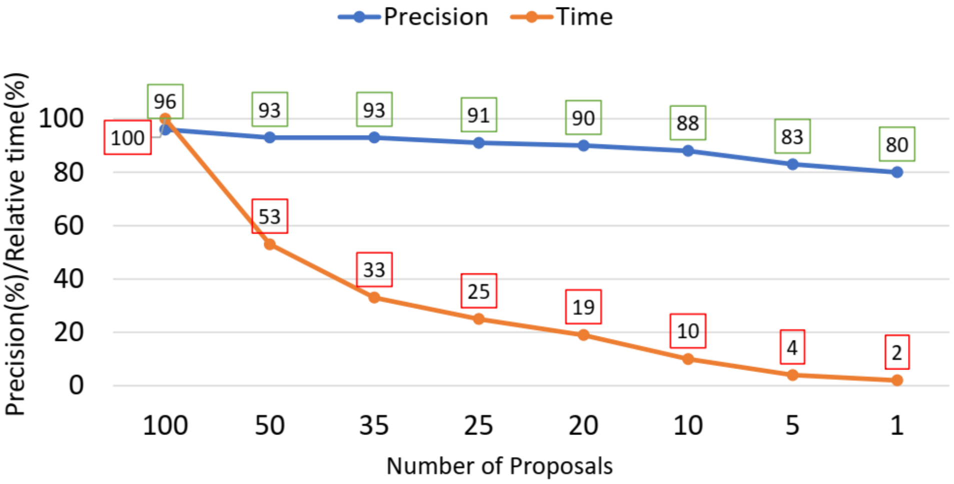


Figure 3.14: Precision@10 vs computational complexity with regard to the number of proposals considered per image on the Paris 6k dataset with NE-C colour neural descriptors. Computational cost is shown in terms of relative time from 100 to 1 proposals.

Experimental setup for COIL-100 with colour neural descriptors

COIL-100 dataset contains multi-view images with single objects on a black background. Thus, in order to address instance retrieval in such dataset, we did not generate object proposals as we did for the Paris 6k and INSTRE dataset. We directly used the VGG-16 *FC8* features to create colour neural descriptors, and we employed the presented query expansion technique to retrieve instances. We used all the 7,200 images as queries. Since the dataset consists of multi-view objects on a 360-degree turntable, we employed the query expansion technique to retrieve rotated views. Every image of the dataset contains a single object under a homogeneous background. For this reason, we directly extracted the FC8 activation without generating proposals using RPN.

Table 3.3: mAP (in percentage) for top-10 retrievals obtained with the baseline (*NE-Raw*) – shown in italics– and the proposed colour neural descriptors in Paris 6k dataset. The best result is shown in bold.

Proposed Descriptors	mAP
NE-C	**97.4**
NE-Raw	*92.2*
NE-O	89.4
NE-O3	95.02
NE-TCD	96.9

3.2.4. Experiments and results on the Paris 6k dataset

In the Paris 6k dataset, the queries are already provided with bounding boxes annotations in the dataset. Following the standard evaluation protocol for the Paris 6k dataset Philbin et al. (2008), we first cropped the 55 query images using the bounding boxes. Then, we extracted the query and the dataset features with respect to different colour neural descriptors, storing the dataset features in a database. To measure the effectiveness of the proposed descriptors, we first obtained their mAPs considering top-10 retrievals and then compared them. In Table 3.3, we present the mAPs for top-10 (k = 10 in Eq. 3.6) retrievals achieved with the different proposed colour neural descriptors. The best performance was yielded using the *NE-C* descriptor with a mAP of 97.4% followed by *NE-TCD* and *NE-O3* with mAP of 96.9% and 95.02%, respectively. In order to compare with the other approaches, we have selected our best performing descriptor NE-C. In Table 3.4, we present the mAPs reported by various state-of-the-art approaches and compare our result with them. Among the earlier works, the highest mAP reported was 79.67% by Radenović, Tolias and Chum (2018). Using our approach, we obtained a mAP of 81.70% with the NE-C descriptor and thus we outperformed state-of-the-art results.

With these experiments, we demonstrate that the proposed colour neural descriptors are very efficient for content-based instance retrieval. Furthermore, due to the low performance of the *NE-O* descriptor, we discard it for the next sets of experiments. In Fig. 3.10, we show the top-10 retrieved instances for some query image examples using *NE-C*.

3.2.5. Experiments and results on the Revisiting Paris 6k dataset

To evaluate our proposal using the revisiting Paris 6k dataset, we followed the Medium-setup and the new evaluation protocol as explained in Radenović, Iscen, Tolias, Avrithis and Chum (2018). In Table 3.5, we compared the colour neural

Table 3.4: Performance comparison with state-of-the-art methods for instance retrieval based on mAPs on the original Paris 6k dataset. We present the dimension of descriptors (dim) and mAP (in percentage) for all methods.

Method	dim	Fine-tuned	mAP
SPOC; 2015	512	No	63.52
SPOC; 2015	512	Yes	74.09
SPOC(ACK); 2020	256	yes	74.60
MAC; 2016	512	No	67.02
MAC; 2016	512	Yes	78.73
MAC(ACK); 2020	256	yes	75.69
RMAC; 2018	512	No	72.02
RMAC; 2018	512	yes	77.94
RMAC(ACK); 2020	256	yes	75.76
CROW; 2016	512	No	68.94
CROW; 2016	512	yes	77.48
CroW(ACK); 2020	256	yes	75.94
Gem; 2018	512	yes	79.67
Gem(ACK); 2020	256	yes	76.26
NE-C(ours)	3000	No	**81.70**

descriptor *NE-C* –which outperformed the rest– against some recent and relevant state-of-the-art approaches. Among the state-of-the-art methods, the highest mAP of 80.7% was obtained with DELF-GLD Teichmann et al. (2019) method, in comparison to a mAP of 82.02% using *NE-C*. We also present the mean precision at 10 (mp@10), which is the mean of the precision for the top 10 retrievals as reported in the work Teichmann et al. (2019). *NE-C* yielded a mp@10 of 97.2%, being a bit lower than some other recent methods. We can notice that for a small number of retrievals –such as 10– the mean precision is saturated since most of the approaches are able to get a high result, however, achieving a high mAP is more challenging as the number of retrievals increases.

3.2.6. Experiments and results on the INSTRE dataset

Next, we evaluated the retrieval performance on INSTRE-M, which constitutes a similar scenario to the Paris 6k dataset. To evaluate the performance, we computed the mAP following the protocol described by Iscen et al. (2017), which uses 1250 query images. In Table 3.6 we present the achieved results, and it can be observed that the concatenation of channel-specific DCFs, *NE-C*, yielded the best performance with a mAP of 78.8% followed by *NE-TCD* with mAP 77.5%. In Fig. 3.10, we show

Table 3.5: Performance comparison with state-of-the-art methods for instance retrieval based on mAPs and mean precision at 10 (mp@10) on the revisiting-Paris 6K dataset. These methods were presented by Teichmann et al. (2019). In bold, the results of the proposed method and the best results of the state-of-the-art methods.

Method	mAP	mp@10
HesAff-rSIFT-ASMK+SP ; 2016	61.4	97.9
HesAff-rSIFT-ASMK ; 2016	61.2	97.9
ResNet101-R-MAC ; 2016	78.9	96.9
DELF-ASMK+SP ; 2017; 2018	76.9	99.3
AlexNet-GeM ; 2018	58.0	91.6
HesAff-HardNet-ASMK+SP ; 2018	65.2	98.9
VGG16-GeM ; 2018	69.3	97.9
ResNet101-GeM ; 2018	77.2	98.1
ResNet101-GeM+DSM ; 2019	77.4	99.1
DELF-D2R-R-ASMK+SP ; 2019	78.2	**99.4**
DELF-GLD ; 2019	**80.7**	99.1
NE-C (ours)	**82.02**	**97.2**

Table 3.6: Performance comparison with state-of-the-art methods for instance retrieval based on mAPs in INSTRE dataset. We present the mAP for all methods in percentage.

Method	dim	mAP
CroW; 2016	512	41.6
CAM; 2017	512	32.5
R-MAC; 2016	512	47.7
R-MAC-ResNet; 2017	2048	62.6
BLCF; 2018	336	63.6
BLCF-Gaussian; 2018	336	63.6
BLCF-SalGAN; 2018	336	69.8
Lin *et el.*; 2019	1024	57.5
NE-C (ours)	3000	**78.8**
NE-TCD (ours)	3000	**77.5**
NE-O3 (ours)	1000	**70.21**

the top-10 retrieved instances for some query examples using *NE-C*.

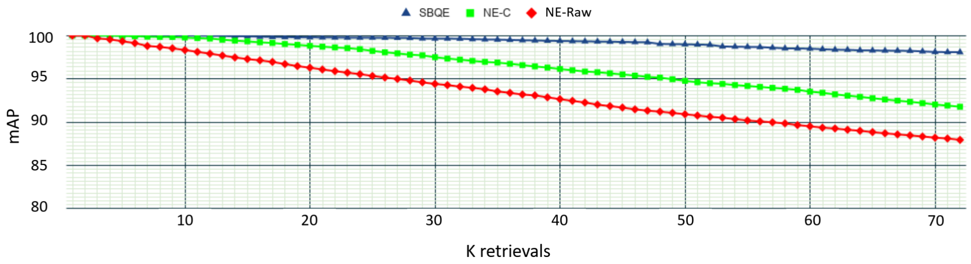


Figure 3.15: mAPs at top-k instance retrievals on COIL dataset using descriptors NE-Raw and NE-C (with and without query expansion), where SBQE represents NE-C with query expansion.

Table 3.7: mAP (in percentage) and search time per query (in seconds) for top-20 retrievals with respect to the proposed colour neural descriptors and the Stride-Based Query Expansion (SBQE) in COIL dataset.

Proposed Descriptors	mAP	Time
NE-C	98.8	0.042 secs
NE-O3	96.0	0.022 secs
NE-TCD	98.7	0.043 secs
SBQE(NE-C)	**99.8**	0.77 secs

3.2.7. Experiments and results on the COIL-100 dataset

In the first experiment, we employed the method described in Section 3.1 with the proposed colour neural descriptors for top-20 retrievals to determine the best descriptor. Since we are addressing a different dataset, we compared all the descriptors again for $k = 20$ retrievals. Table 3.7 presents the achieved results, where it can be seen that the highest mAP (99.8%) was obtained using *SBQE* (query expansion of NE-C) followed by *NE-C* and *NE-TCD* with mAPs 98.8% and 98.7%, respectively. Also, we registered and present the time required for per-query instance search with respect to each of the descriptors. Furthermore, in Table 3.8, we compare our method with some of the works that were reported in (Liu, Wu, Feng, Qiao, Liu, Luo and Wang, 2018) for top-20 retrievals based on $precision$. It can be observed that all the proposed descriptors achieved comparable mAPs, and we selected the best one (*NE-C*) to compare it with the state-of-the-art approaches.

Retrieving multi-view instances: Additionally to the previous experiments, we evaluated a top-72 retrieval system. In this way, we are allowing the retrieval of

Table 3.8: Precision (in percentage) of classical image descriptors with 20 returns in Coil-100 dataset which were reported in Liu, Wu, Feng, Qiao, Liu, Luo and Wang (2018).

Method	Precision
LBP	74.30
MSD	97.72
CDH	92.48
HSV	96.73
PUD	99.11
NE-C (ours)	**99.56**
SBQE(NE-C) (ours)	**99.99**

all the 72 objects per class in the dataset to verify if all the rotated or multi-viewed objects are retrieved correctly. We used *NE-C* descriptors with and without query expansion, and compared the results with the baseline *NE-Raw* descriptor. Fig. 3.15 shows the results obtained by plotting mAP versus k retrievals, where k goes from 1 to 72. The best results were obtained using *NE-C* with SBQE as compared to *NE-C* and *NE-Raw*. At $k = 72$, the mAP obtained using *NE-C* with SBQE is 98.3 %, whereas using *NE-C* and *NE-Raw* are 91.7 % and 87.9 %, respectively. The performance is boosted while query expansion is applied to *NE-C*, but it has high computational cost as compared to *NE-C* without query expansion.

Based on the previous experiment, we compare *NE-C* and SBQE against the state-of-the-art methods, presenting the results in Table 3.9. The mAP obtained using *NE-C* and SBQE is 97.9%, which is superior to the best mAP of 95.4% as presented in Mukherjee et al. (2020). The query expansion approach is computationally more expensive, but it is very useful when it is important to boost the performance of the colour neural descriptors for image retrieval in multi-view datasets.

In Fig. 3.16, we illustrate the retrieval of multi-view images. The initial query image corresponds to the orange mug rotated 315 degrees. Then, a window of stride 3 selects the next query, which in this case is rotated by 325 degrees. The bottom two rows show the top-11 retrieved images when the initial queries are positioned at 315 and 145 degrees, respectively. We set the size of the stride as $s = 3$ and we retrieved top-s similar images. The s_{th} retrieved instance in the hit list is set as the next query, and the process continues until k instances are retrieved. We chose the orange cup rotated 315 degrees as the initial query image. It can be observed that each of the top-s retrievals are in close vicinity to 315 degrees, and are slightly rotated (*clockwise or anticlockwise*) with respect to their queries.

In Fig. 3.10, we show the top-10 retrieved instances for some query examples of the COIL-dataset using *NE-C* descriptors.

Table 3.9: Performance comparison with state-of-the-art methods for instance retrieval based on mAPs on COIL dataset. The best results are shown in bold, and the second best value is italicised.

Proposed Descriptors	mAP
Ahmed *et al.*; 2019	93.0
BoVW ; 2020	78.5
txx ; 2007	61.5
fuzzy weights ; 2009	80.0
vwa ; 2017	86.0
BoCIDVW ; 2020	*95.4*
NE-C (ours)	**92.3**
SBQE (NE-C) (ours)	**97.9**

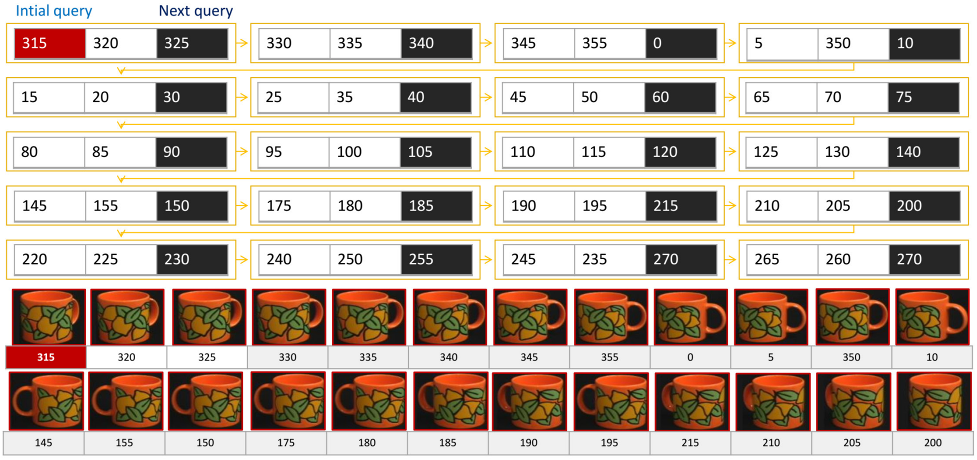


Figure 3.16: Illustration of the retrieval of an image sample in COIL dataset using the proposed query expansion approach.

3.3. Discussion

In this work, we proposed colour neural descriptors for instance retrieval, and we evaluated them in four datasets to assess its performance in terms of mAP. We wanted to provide a solution that employs colour models with deep CNNs. In situations in which it is needed to retrieve an object in datasets containing objects with very similar appearances, the colour becomes a very discriminative feature. To build our framework, we used a model previously trained with RGB images, which can detect intrinsic features such as edge, shapes and other key points as long as the provided image consists of the RGB colour channels. Besides, if the colour space changes, the model will not be able to detect discriminative features since other colour spaces would have different channel values than the RGB scale. Due to this reason, we opted for RGB colour space, and we consider experimenting with different colour spaces in the future.

To retrieve specific instances from a particular dataset, most of the proposed works based on deep learning methods to date adopt fine-tuning approaches. However, the raw neural activation obtained directly by passing an image through the CNN may not be feasible for similarity search. This is due to the fact that, under different illumination conditions, the appearance of most objects varies, and hence, we may have different neural activations for the same object. As a result, we may not be able to retrieve all instances related to the query object, and as a consequence, other approaches opt for fine-tuning. Therefore, in order to make the descriptors more discriminative, we enhanced the DCFs by using colour models, and later on, we evaluated some combinations of the DCFs to obtain more discriminative feature vectors. The hypothesis behind our proposal for creating colour neural descriptors is that, if we separate the three channels of an image and consider CNN features specific to each of them, then the vector obtained by its combination is more discriminative than the one obtained by passing through the CNN the complete RGB image.

The main advantage of our proposed approach is the usage of a hybrid architecture in which we combined two state-of-the-art networks for instance search without explicitly training or fine-tuning. Besides, we expanded our solution looking for retrieving multi-view images by introducing a stride based query expansion technique. In most of the cases, to extract such instances, fine-tuning an algorithm specific to the dataset is the cue. This is because, when an object is rotated, for example in a turntable, the appearance may significantly vary, and as result, the neural activations of the object change as well. By applying the proposed stride-based query expansion, we were able to successfully retrieve such rotated images with high mAP, but with the drawback of increasing the computational time. Furthermore, the descriptor obtained proved to be competitive for instance retrieval, outperforming state-of-the-art results in four datasets: COIL-100, INSTRE-M, Paris 6K and Revisiting-Paris 6k. However, the feature extraction is complex since they are required to be extracted from three different channels. Nevertheless, the complexity and the retrieval time can be reduced significantly by parallelizing the extraction process in a GPU machine.

3.3.1. Performance trade-offs and time consumption

We observed that the precision and mAPs obtained by our proposal are superior to the state-of-the-art approaches. However, apart from exhibiting high performance, it can be noticed that, depending on the chosen colour neural descriptor, there is a trade-off between mAP and computational cost. In Table 3.7, we can observe that *NE-O3* is approximately $1.3\times$ faster than *NE-C* and *NE-TCD*, and $35\times$ than *SBQE(NE-C)*, but it has the minimum mAP of 96% as compared to the highest mAP of 99.8% obtained by *SBQE(NE-C)*.

In Fig. 3.17, we show the *relative time* vs *mAP* trade-offs of *NE-C* descriptor with query expansion. We define relative time in a range from 0 to 100, where 100 represents the maximum time taken by the descriptor. We show relative time with respect to a stride of 3. We compute the results on COIL dataset using *NE-C* and *SBQE* approach for top-72 retrievals. As we had seen, for a stride of 3, the mAP is 99.8% and we consider it as the scenario with the maximum time, that is 100%. We can observe that as the size of the stride increases the computational complexity reduces significantly along with a slight decay in the mAP. Therefore, for instance retrieval systems where time is a prime concern, they can increase the stride size up to 10 when an mAP close to 80% suffices, or they can use *NE-O3* or *NE-Raw* descriptors to speed up the approach. Whereas, *NE-C* is the one to select when precision is of utmost importance.

In general, the computational time required to process a single image for creating the colour neural descriptor is approximately 0.25 seconds, and for generating all the 100 proposals is around 0.20 seconds. We also observed that, as compared to *NE-Raw* descriptor, the colour neural descriptors are computationally expensive since a feature extraction of three individual channels is required. But, due to the availability of recently advanced parallel computing resources with powerful GPUs, we significantly reduced the descriptor creation time by simultaneously extracting the deep features corresponding to the three colour channels. As a consequence, the time required to compute *NE-Raw* and *NE-C* descriptors are approximately equal, which is 0.08 seconds.

3.3.2. Scalability of the proposed approach

During the last few years, as the number of images has increased exponentially in the order of millions and billions, optimizing the matching and sorting tasks in databases of feature vectors in terms of time is critical for a quick instance retrieval. Besides, the storage of billions of feature vectors can result in the need for huge RAM memory. To deal with these two issues, we save the descriptors in HDF5 binary format, which allows storing vast amounts of numerical data in a single file.

In order to compute the retrieval time and to establish a time complexity order

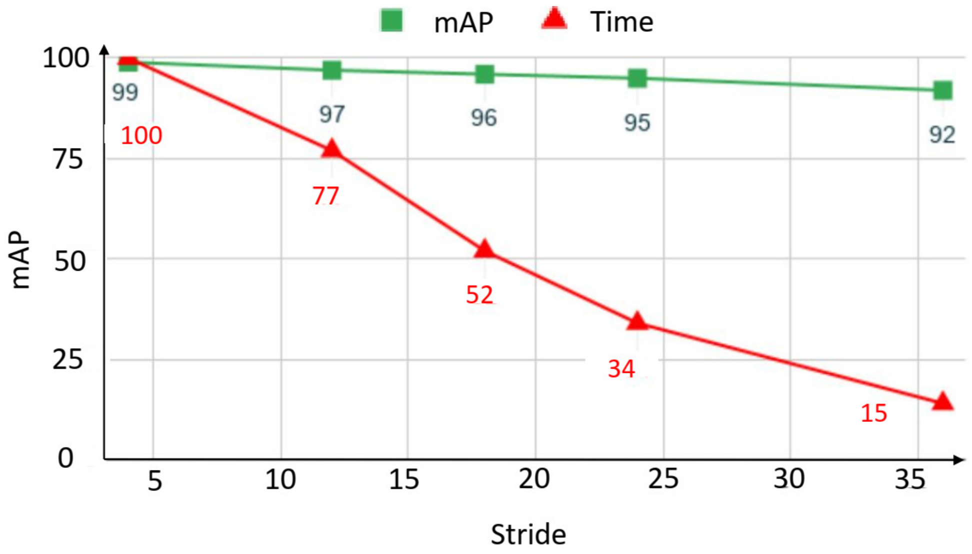


Figure 3.17: mAP vs relative time complexity of the stride base query expansion approach with respect to stride size for top-72 retrievals on the COIL dataset using *NE-C* colour neural descriptor. Time complexity is shown in terms of relative time for a stride of 3.

for big databases, we generated 100 million random vectors of dimension 3000 representing the colour descriptors *NE-C*. Since the memory of our RAM is limited with 16 GB, we created 1000 HDF5 files each one containing 10^5 vectors. To compute the similarities of a vector with respect to all 100 million vectors, we loaded an HDF5 file into the RAM, and once compared, we removed it and a newer one was loaded subsequently. In Fig. 3.18, we show the computational time with respect to the number of descriptors. The average computational time taken for loading 10^5 vectors and computing the similarity scores is approximately 2.3 seconds. Then, to compare with 10 million descriptors, the computational time was 241 seconds (about four minutes) and required 100 HDF5 files to be loaded and unloaded. At last, the computational time to compute the similarity scores of 100 million descriptors took just 37 minutes, where approximately 41000 comparisons are made per second, which is reasonably fast for many applications that do not require real-time performance. In the case of computers with different specification or feature vector dimensions, the number of HDF5 files and the vectors stored in each file could be adapted.

In addition, we can observe in Fig. 3.18 that the computation time is linearly dependent on the number of descriptors present in a database, and hence, we have a linear order of complexity $O(n)$. Once the similarity scores are obtained, we also need to sort the hit list to find the most relevant retrievals. We performed the sorting

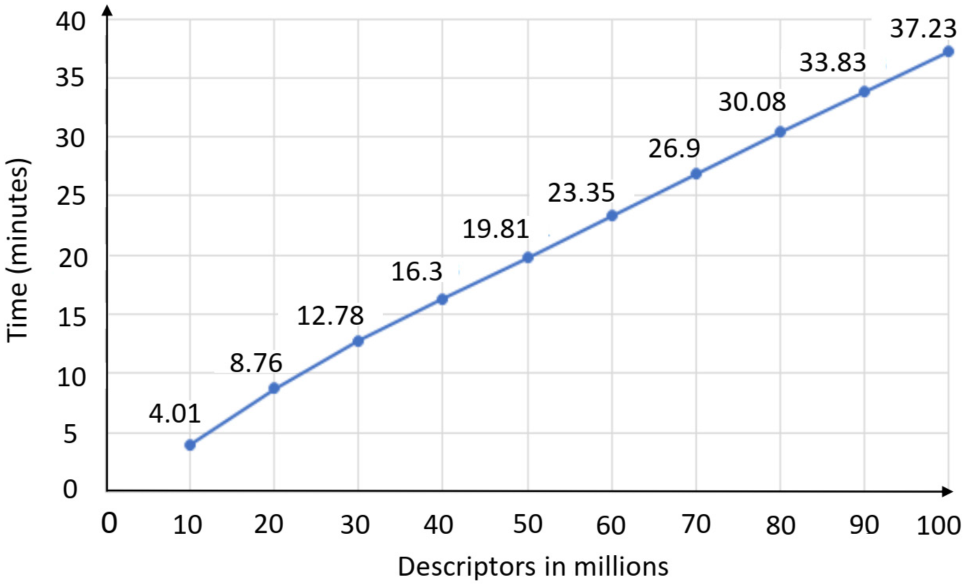


Figure 3.18: Computation time (in seconds) vs number of descriptors in millions with respect to similarity score computation between the feature vector and the database vectors.

by means of the quick-sort algorithm, which has an average complexity of $O(nlogn)$. Thus, the overall complexity of instance search is given by $O(n) + O(nlogn)$. In fact, due to this inherent linearity, we can further scale up this approach for billions of descriptors by storing them in more HDF5 binary format files. Moreover, based on the observed experimental results in which the time varies almost linearly with respect to the number of descriptors analyzed, the computation time is expected to be approximately 373 minutes (6 hours) for one billion descriptors comparisons. Typically, in a recent high-end machine, say with 128 GB RAM, 10 million colour descriptors can easily fit in. For instance, if there are one billion descriptors, then 100 HDF5 batch files can be used to store 10 million descriptors each. Moreover, the loading and unloading overhead can be further reduced in a machine with high RAM, and thus, the instance retrieval can be speeded up. In this way, the retrieval step can be scaled.

Similarly, the feature extraction step can be also scaled up while maintaining a linear order of complexity. For n images on the dataset, there are n forward passes of the network, and in each pass, we obtain M (a constant) number of object proposals at the same time (as mentioned in Section 3.2.1). As n grows more and more, the number of proposals generated for each image will become trivial as compared to n. Therefore, the order of complexity for feature extraction is $O(n)$, which is linear. However, since feature extraction using any CNN is computationally expensive while dealing with large scale datasets, it is ideal to run parallel instances of the feature extractor (CNN) in mutually exclusive image batches. In our case, for faster feature extraction, we selected a batch size of 200, and it takes approximately 1.85 seconds to extract features of 200 images and to save them as HDF5 files.

Moreover, the computation time can be further reduced if the descriptors are converted into binary codes using deep hashing, which has emerged as an important technique for image retrieval in large scale datasets. However, in this work, our prime focus was only to create colour descriptors for the instance retrieval without fine-tuning. Since we have obtained a new deep feature representation to define colour descriptor, we aimed at proving the full efficacy of this original representation without applying any techniques on top of it. Besides, we were able to do a fast retrieval where a query descriptor can be compared with 40,000 descriptors in less than just two seconds. In addition, we can efficiently store more than one million descriptors into the RAM using HDF5 binary format.

3.4. Conclusions

In this work, we have presented colour neural descriptors for instance-based retrieval using CNN feature maps and colour models. First, we modified the input part of the network to generate DCFs for the different colour channels. Next, the extracted activations were passed through the Colour neural Descriptor Generation (CDG) layer to construct the colour neural descriptors. For developing a query-based retrieval system, we first created object proposals for each of the images in a given dataset, and then we calculated the corresponding colour neural descriptor for each proposal. After that, we computed the cosine similarity between the query descriptor and each object proposal descriptor, retrieving those images where the similarity scored higher than a specified threshold. Additionally, we introduced a stride-based-query-expansion technique, especially appropriate to retrieve images from a multi-view dataset. We selected an initial query to retrieve the top-k similar instances, and then we used a stride of size s to select the s^{th} retrieved instance to be used as the next subsequent query. In contrast to prior works, which relied on fine-tuning a network, we enhance the DCFs to increase the discriminative power concerning colour variations.

We evaluated the proposed method using standard protocols. We experimentally showed that our approach significantly boosts the retrieval performance in terms of precision without fine-tuning techniques applied. Besides, to address multi-view image retrieval, we use a query expansion technique based on stride. The descriptor obtained proved to be competitive for instance retrieval, outperforming state-of-the-art results on four datasets: COIL-100, INSTRE-M, Paris 6K and Revisiting-Paris 6k.

In the future, we will train a network to directly generate colour descriptors along with the proposals in order to decrease the feature extraction time, and also we would consider multi-label retrieval using the proposed colour descriptors. In addition, other colour models such as HSL, HSV, CMYK can also be used for obtaining discriminative colour descriptors by applying the same formulation.

Chapter 4

Texture-based image retrieval

This chapter [1] addresses the problem of texture-based image retrieval using latent space representation derived from a deep CNN which is based on an autoencoder. In this thesis, texture retrieval is performed for query images that present a single texture pattern and is mainly applied to the retrieval of fabrics or textiles in complex images. Moreover, to build an effective retrieval system, we introduce a Fourier based approach, in which spatial images and their discrete Fourier transform maps are combined to derive a novel texture representation. We also present a new and efficient texture-based image retrieval framework based on an RPN, convolutional autoencoders and transfer learning, in which we extract the features from the latent space layer of the encoder as texture descriptors.

To sum up, the main contributions in this chapter are the following:

1. We introduce a novel texture descriptor known as Deep Fourier Texture Descriptor (DFTD). The features of the descriptor are extracted from the latent space layer of a convolutional autoencoder whose inputs are the outcomes of blending the magnitude spectrum of a discrete Fourier transform (DFT) and the spatial information of the images.

2. We propose a CBIR framework for texture retrieval which uses an RPN to propose prospective texture regions which are fed to an autoencoder. The model was trained with transfer learning techniques.

3. We evaluate the proposed texture-based image retrieval approach in the context of a real-world application, which can be applied for image, instance and object retrieval for crime scenes investigation during forensic analysis. We also assess the proposed texture descriptor both for texture-based image retrieval and for texture classification on four public datasets, where the proposed method outperformed some recent and relevant state-of-the-art works.

[1] This chapter is “reproduced with permission from Springer Nature”, which is adapted from Saikia, S., Fernández-Robles, L., Alegre, E. et al. Image retrieval based on texture using latent space representation of discrete Fourier transformed maps. Neural Comput & Applic (2021). DOI: 10.1007/s00521-021-05955-2.

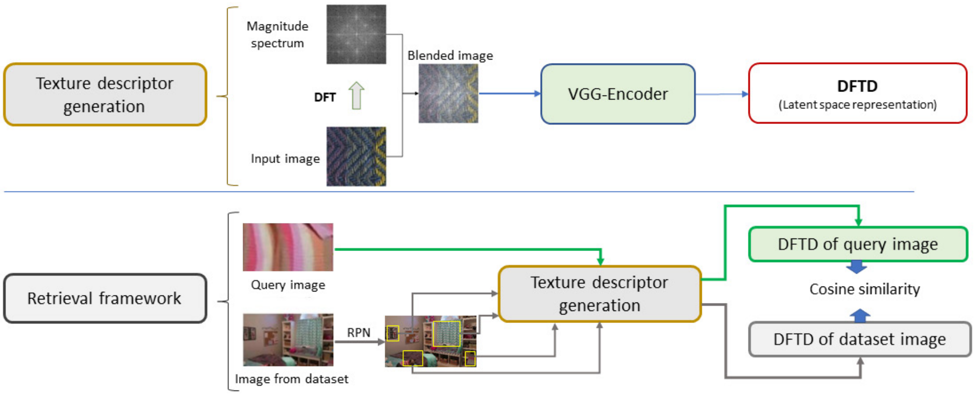


Figure 4.1: Overview of our proposed query-based retrieval method. Stage 1 (on the top) represents generation of texture descriptor DFTD, and stage 2 (down) illustrates the retrieval framework.

4.1. Method

The proposed method consists of two stages (see Fig. 4.1): (1) texture descriptor generation, and (2) instance retrieval framework. In the first stage, we compute the texture descriptors of images created by a pixel-wise combination of the magnitude spectrum DFT and the spatial domain of an image. We also propose a new architecture based on autoencoders to extract the texture descriptors, which we train with the generated images. In the second stage in Fig. 4.1, we illustrate our complete query based retrieval framework using texture descriptors. In addition, we also present a texture classification schema to validate the efficacy of the proposed texture descriptors.

4.1.1. Texture descriptor generation

Discrete Fourier transform of images

The Fourier transform is a classical image processing tool which decomposes images into sine and cosine components. The output generated by Fourier transform represents the image in the frequency or Fourier domain, while the input, which is the user-provided image, is the spatial domain equivalent. Each pixel in a Fourier domain image represents a particular frequency contained in the spatial domain image.

To generate frequency images, we use the DFT, which is the sampled Fourier transform that contains enough frequencies to fully describe an image in a spatial domain. Moreover, the image generated by the DFT is of the same size as that of the spatial image, and hence, we can train a network without modifying the input size parameters.

The DFT for an image of size $M \times N$ pixels denoted by $f(x, y)$, with x and y as its spatial coordinates is given by:

$$F(u, v) = \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} f(x, y) e^{-i2\pi(\frac{ux}{M} + \frac{vy}{N})}, \tag{4.1}$$

where u and v are in the range $[0, 1, 2, ..., M-1]$ and $[0, 1, 2, ..., N-1]$, with $i^2 = -1$ is the complex imaginary number. One of the properties of the DFT is that the original image $f(x, y)$ can be obtained by applying the inverse DFT to $F(u, v)$, which is defined as:

$$f(x, y) = \frac{1}{MN} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} F(u, v) e^{i2\pi(\frac{ux}{M} + \frac{vy}{N})}. \tag{4.2}$$

Since the DFT is a bijective function, $f(x, y) \iff F(u, v)$, the DFT of an image satisfies the following two properties:

$$f(x, y) e^{i2\pi(\frac{ux}{M} + \frac{vy}{N})} \iff F(u - u_0, v - v_0) \tag{4.3}$$

$$F(x - x_0, y - y_0) \iff F(u, v) e^{-i2\pi(\frac{ux_0}{M} + \frac{vy_0}{N})} \tag{4.4}$$

In the DFT image, the zero frequency component is present in the top left corner, and to bring it to the center, the DFT result is shifted by $M/2$ and $N/2$ in both directions u and v. Based on the properties defined in Eq. 4.3 and 4.4, the Fourier transform of an image can be shifted using the transformation:

$$f(x, y)(-1)^{(x+y)} \iff F(u - M/2, v - N/2), \tag{4.5}$$

so that $F(0, 0)$ is at the point $(u_0, v_0) = (M/2, N/2)$. Since, Fourier transform is a complex-valued function of a real-valued function, we can write it as

$$F(u, v) = R(u, v) + iI(u, v), \tag{4.6}$$

where R is the real part and I is the imaginary part, respectively. Finally, we generate the magnitude spectrum as

$$F_m(u, v) = 20 \times \log |\sqrt{R^2(u, v), iI^2(u, v)}|, \tag{4.7}$$

In this chapter, we refer to $F_m(u, v)$ as DFT magnitude spectrum, and for the shake of brevity, DFT image.

Intuition of DFT and RPN framework. In Fig. 4.2, we illustrate the intuition behind our proposed approach to show how the DFT can be used for creating distinctive features for image retrieval and classification. Let $f(x, y)$ be the image in the spatial domain and $F_m(u, v)$ be its corresponding DFT magnitude spectrum image. The task is to find out if the image $f'(x, y)$ contains the instance $f(x, y)$ or not. We compute the DFT $F'_m(u, v)$ of the image $f'(x, y)$, which represents the Fourier spectrum of the complete image, and hence the DFT image differs from $F_m(u, v)$ due to the presence of multiple textures. The image $f'(x, y)$ posses certain frequencies, and we need to search if there are regions in it that might contain frequencies similar to $f(x, y)$. The red box overlaid on image $f'(x, y)$ represents the ground truth, and that particular region contains the same frequency as that of $f(x, y)$. However, to generate possible regions in $f'(x, y)$ we need an external region proposal system. In order to address this issue, we employ an RPN to generate proposals, and then we compute the DFT magnitude spectrum images for each of the proposals. On the right part of Fig. 4.2, we present three examples of proposals generated from $f'(x, y)$ as $f'_1(x, y)$, $f'_2(x, y)$ and $f'_3(x, y)$, and their corresponding magnitude spectrums DFT images $F'_{m1}(u, v)$, $F'_{m2}(u, v)$ and $F'_{m3}(u, v)$, respectively. The proposal $f'_2(x, y)$ overlaid with the red bounding box represents a texture patch of the same class as the query $f(x, y)$, and also we can observe that they have similar magnitude spectrum DFT images, as represented by $F_m(u, v)$ and $F'_{m2}(u, v)$. Therefore, the properties of the DFT can be used with spatial images to derive representations to search similar images for retrieval.

Linear blending of DFT magnitude spectrum and spatial images

Fourier transform has the very useful property of highlighting the dominant spatial frequencies as well as the orientations of the structures contained in an image. In our problem, the texture patterns present in the images are those structures. The texture images usually contain quasi repetitive patterns, and Fourier transform can define those periodic functions present in the form of repetitive patterns. However, while dealing with a big corpus of images, the frequency information is not sufficient to distinguish correctly different texture patterns, as some images belonging to different classes might have similar DFT magnitude spectrum representations. Hence, to address this issue, we combine both frequency and spatial information of the image by doing pixel-wise weighted addition of the DFT magnitude spectrum image $F_m(u, v)$ and the spatial image $f(x, y)$ to obtain a blended image $B(x, y)$, as defined in Eq. 4.8 and shown in Fig. 4.3.

$$B(x, y) = (1 - \alpha) f(x, y) + \alpha F_m(u, v) \tag{4.8}$$

The parameter α can be varied from 0 to 1 to stress $F_m(u, v)$ or $f(x, y)$. We empirically selected α equal to 0.7.

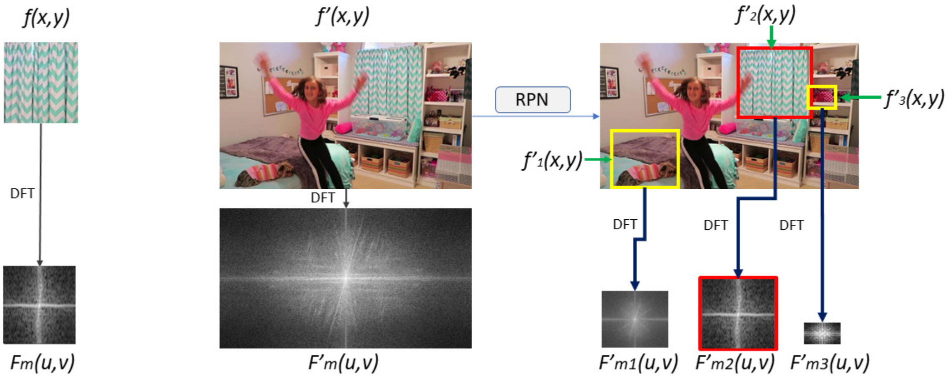


Figure 4.2: Some images and their corresponding magnitude spectrum DFT maps to illustrate the motivation behind our approach. In the left, a query image. In the middle, a dataset image. The magnitude spectrum DFT map contains the frequencies of the query image but unrecognizable due to the mix of frequencies from other textures in the image. In the right, some texture patches are detected with an RPN, they are represented with overlaid yellow and red rectangles, and the magnitude spectrum DFT maps are computed. The red rectangle identifies a texture proposal of the same class as the query.

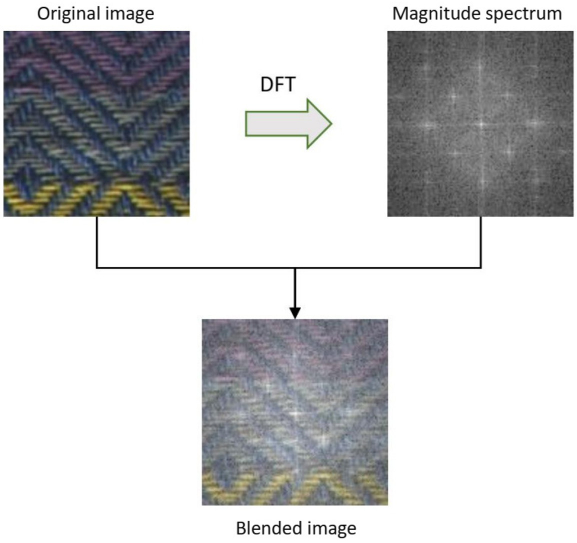


Figure 4.3: Blended image $B(x, y)$ generated by the weighted addition of the original image and the DFT magnitude spectrum image

Proposed architecture

The proposed architecture is based on convolutional autoencoders, where the encoder part consists of the convolutional layers of the VGG-16 architecture initial-

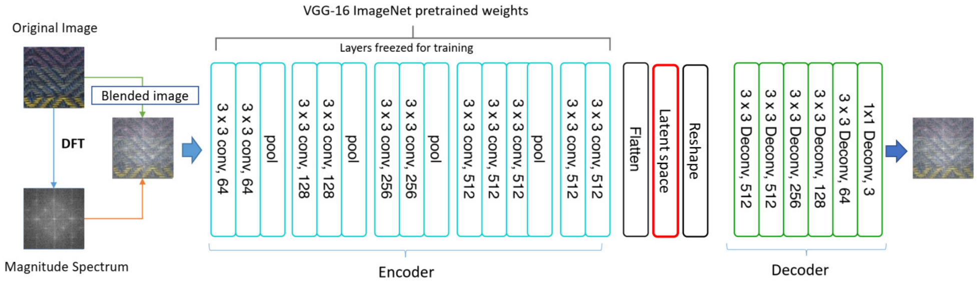


Figure 4.4: Training architecture of the convolutional autoencoder with DFT based blended images.

ised with ImageNet weights. In general, a convolutional autoencoder extends the basic structure of the simple autoencoder by changing the fully connected to convolution layers. The encoders learn to encode the input in a set of simple signals and then try to reconstruct the input from them based on the latent space representation learned by the network. However, our architecture varies from traditional autoencoders, since on the one hand, we apply transfer learning to the encoder, and on the other hand, the number of layers in the encoder and decoder are not the same. We next explain the architecture in detail as illustrated in Fig. 4.4.

VGG encoder. Creating a network and training it from scratch is expensive in terms of computational cost and availability of annotated data. Besides, it requires an optimisation of the network hyper-parameters to minimize classification errors. This complexity can be avoided by using a pre-trained network, and henceforth, we use VGG-16 model pre-trained with ImageNet dataset as an encoder. In addition, another main reason for selecting VGG-16 is that it is a sequential network, which facilitates the construction of a simplified convolutional autoencoder. It is true that other architectures such as ResNet and InceptionNet can also be used as encoders but they have some disadvantages. With such networks, we need to create a complex decoder since they are not sequential, and it will make the overall architecture computationally expensive in terms of both space and time. Hence, we choose the VGG-16 as an encoder, and we train the complete architecture with blended texture patches. In Fig. 4.4, a blended texture patch $B(x, y)$ is given as an input to the VGG encoder, which is followed by the latent space representation and the decoder.

Latent space representation as DFTD. In between the VGG encoder and the decoder we have the latent space layer (Fig. 4.4), which contains a vectorised representation of the input image of dimension 512. The 512-dimensional vector defines the proposed descriptor DFTD. The decoder takes this encoded vector and builds

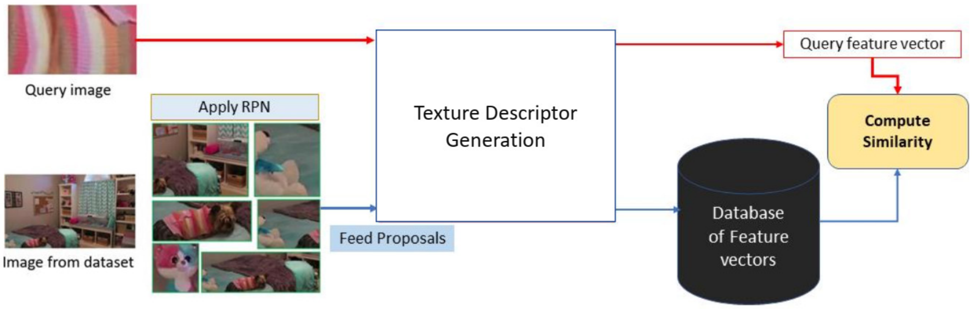


Figure 4.5: Pipeline of our texture-based instance retrieval approach. On the one hand, we compute the DFTD descriptor from the query image. On the other hand, we employ an RPN to generate proposals, and then we compute the DFTD descriptors from the proposals which are stored in a database. Finally, we compare the query and database descriptors using a similarity metric.

the output image to be as close to the input image as possible. The architecture learns to encode a DFT based blended image $B(x, y)$ in a set of simple signals and then tries to reconstruct the input from them based on the representation from the latent space layer. In Fig. 4.4, we represent the architecture of our proposal. The first convolution layer receives the DFT based blended image $B(x, y)$, and the decoder learns to convert the latent space representation back to an image as close as possible to the input image $B(x, y)$. In this way, the network learns specific texture representations on top of the ImageNet features so that it can generalize to different types of texture images for texture feature extraction.

4.1.2. Retrieval framework

In this section, we describe the complete texture retrieval framework which is composed of three steps: (1) query feature extraction (2) dataset feature extraction (3) similar texture search. In Fig. 4.5 we illustrate the proposed framework.

Query feature extraction. We first apply the DFT to the query image to obtain the DFT magnitude spectrum, and then we perform the pixel-wise weighted combination defined in Eq. 4.8 to the query and its DFT magnitude spectrum to obtain the DFT based blended query image $B(x, y)$. We next feed the resultant image to the trained convolutional autoencoder model and extract the latent space representation also named as DFTD, which provides a feature descriptor of the supplied query image.

Detection of regions of interest using an RPN and creation of a DFTD database. The identification of a particular texture in an image is the most crucial step in texture-based instance retrieval tasks. In particular, the queried texture needs to be compared with all distinctive texture patches of the image since the query can be present in different sizes, scales and orientations. In this stage, we integrate the RPN with the convolutional encoder, so that the region proposals generated by the RPN are fed to the encoder to extract the DFTD descriptors of each of the regions. The DFTD descriptors of all proposals are stored in a database together with the label of the image they belong to. However, in the case of datasets that contain a single texture pattern in each image, we can directly compute the DFTDs from the whole images without the need of an RPN.

Similar texture search. To retrieve top-k images that contain a similar texture pattern as that of the query, we compute the cosine similarity (CS) metric between all pairs of query and database DFTD descriptors as defined by Eq. 3.4 in Chapter 3.

4.1.3. Texture classification

We also present a texture classification approach based on transfer learning, in which we use the same VGG-Net architecture pre-trained with the ImageNet dataset as in Section 4.1.2. In addition, we added two dense layers after the convolutional layers of the VGG-16 network, and the end of the classification layer uses the softmax activation. Similarly to the retrieval framework, we train the network with $B(x, y)$ images using the same hyper-parameters. For testing, we extract the class labels based on the activation obtained from the classification layer of the network, and we compare the class labels with the ground truth labels. Unlike the retrieval framework, in this classification approach we have a dense layer instead of the latent space layer, and for convenience, we will also represent the dense features as DFTD.

4.2. Experimental setup and results

4.2.1. Datasets

Our proposed method is evaluated using the following datasets.

Coloured Brodatz texture (CBT) dataset: Coloured Brodatz Texture (CBT) dataset (Abdelmounaime and Dong-Chen, 2013) is an extension of the Brodatz texture dataset. It contains texture images which possess a wide variety of colour content. Further, it consists of 112 textures of size 640×640 pixels, where each one is divided into 25 non-overlapping images of size 128 × 128 pixels. As a consequence, the final dataset consists of 2800 images in total with 25 images per class. In our work, we use this dataset only to train our network with coloured texture images, so that the network can well distinguish between similar textured patterns with different colours.

Outex dataset: The Outex TC-00013 dataset (Ojala et al., 2002b) is a collection of 1360 images representing heterogeneous materials such as paper, fabric, wool and stones. It comprises 68 texture classes and each one includes 20 image samples of 128 × 128 pixels. Out of which, 10 images are for training and the other 10 are for testing in each class.

USPtex dataset: The USPtex dataset (Casanova et al., 2016) consists of 2292 images with 191 classes of both natural scenes (road, vegetation and cloud) and materials (seeds, rice and tissues). Each class consists of 12 image samples of 128 × 128 pixels, where six images are for testing and the rest for training.

Stex dataset: The Salzburg Texture Image Database (STex) (Kwitt and Meerwald, 2018) consists of 476 colour texture images which are similar to the ones present in Outex and USPtex datasets. For testing, the images are divided into 16 non-overlapping tiles of size 128 × 128 pixels. As a result, the final dataset consists of 7616 images with 16 images per class.

TextileTube dataset: This dataset (García-Olalla et al., 2018) is composed of 684 images of sizes that range between 480 × 360 and 1280 × 720 pixels obtained from 15 videos of YouTube. The videos were recorded in bedrooms with different cameras. The dataset contains 67 classes of textiles such as curtains, carpets, sofas, shirts or dresses, among others. The ground truth of this dataset comprises the class labels and the bounding boxes of the texture regions. This dataset creates a similar context for texture-based image retrieval as the one that usually appears in child sexual exploitation videos recorded in indoor environments, typically bedrooms.

To compare our methods with the state-of-the-art reports, we first use the original queries as considered in (García-Olalla et al., 2018), in which all ground truth textile regions were considered as query images. However, these queries contain several objects parts and hence present shape information. To make the queries completely texture-based, we cropped the images so that only the texture part is visible, and we named this set of queries as *New queries*. The number of queries remains the same, the task is more challenging since there is not shape information. We have made the New queries available to the research community[2]. In our work,

[2] http://gvis.unileon.es/dataset/textiltube/

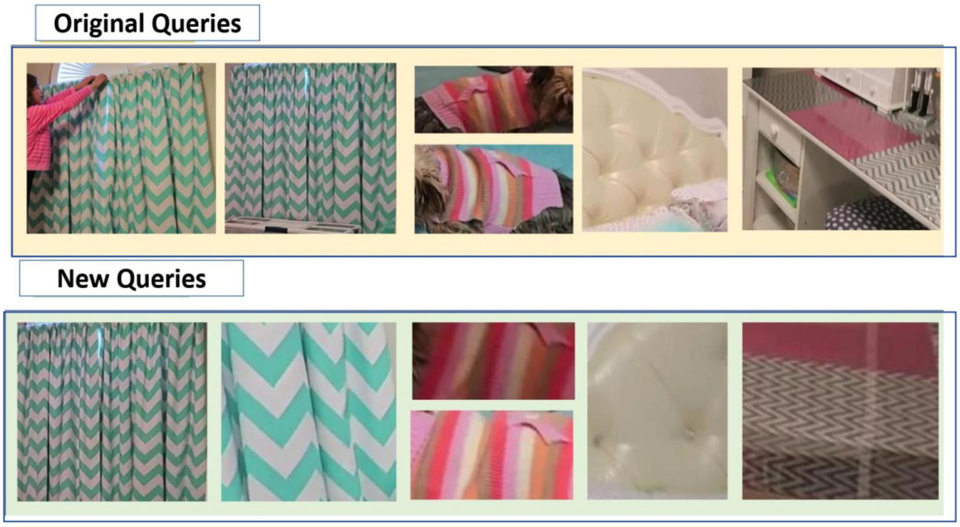


Figure 4.6: **Original queries**: the queries as in (García-Olalla et al., 2018) to compare with existing methods. **New Queries**: images with only the texture portion, which were cropped out from the original queries.

we provide the retrieval results using both types of queries, the original and the *New* ones. In Fig. 4.6, we present some original and *New* queries samples from the TextileTube dataset. Since training images are not provided in the TextileTube dataset, we randomly selected 25 classes, and randomly chose one image of each class to be in the training set. Later, we applied data augmentation to generate multiple images. Besides these few training samples, we considered all other images in the dataset to test the retrieval framework.

In Fig. 4.7, we show some sample images from the CBT, Outex and USPtex datasets. These datasets contain only a single texture pattern per image. Fig. 4.8 represents a sample from the TextileTube dataset, which contains multiple textures in a single image.

The experiments and results presented in this work aim to verify the assessment of the DFTD descriptors derived from the convolutional encoder trained with DFT based blended images. We carried out two different types of experiments with two types of datasets. The first type of experiments was performed using Outex, USPtex and Stex datasets in which images are comprised of only a single textured pattern. Whereas in the second kind of experiment, we consider TextileTube dataset consisting of multiple objects with different textures for texture-based image retrieval. The second experiment represents a real-world scenario, where the images in the

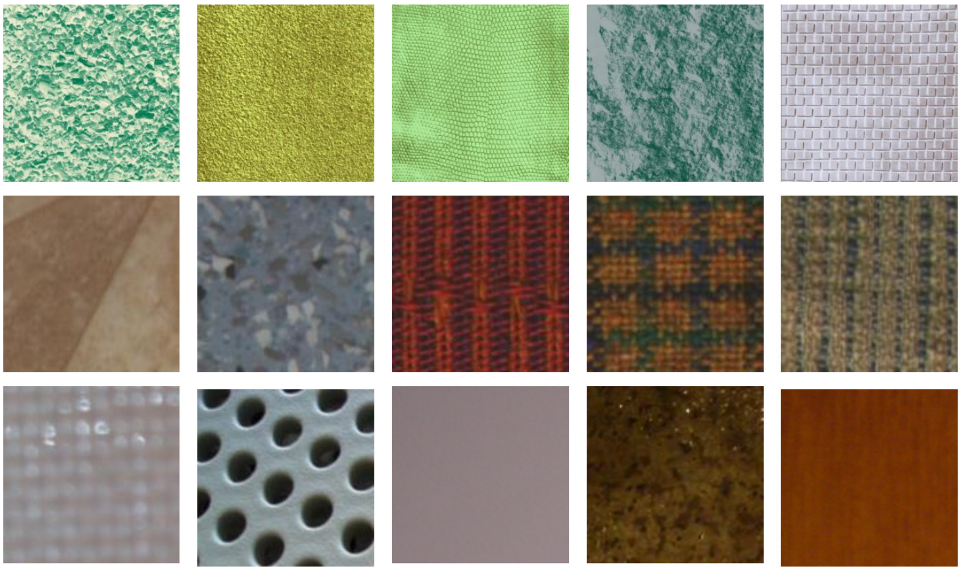


Figure 4.7: From top to bottom row: samples of CBT, Outex and USPtex datasets, respectively.

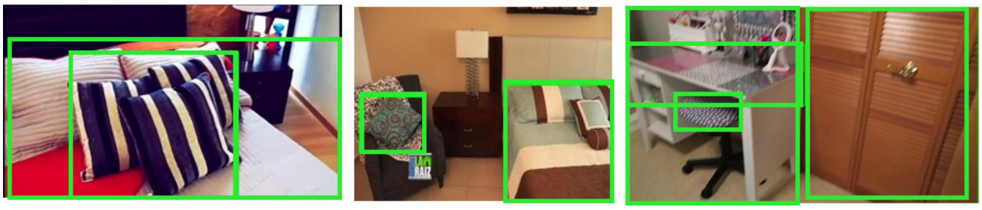


Figure 4.8: Sample images from the TextileTube dataset. Overlaid green boxes represent the bounding boxes of the groundtruth.

TextileTube dataset are taken from indoor scenes, and the queried texture pattern could be present anywhere in those images in different conditions of shape, scale, lighting, orientation, etc. We verified our descriptor using Outex, USPtex and Stex datasets for texture retrieval and classification in order to be able to compare the performance with the state-of-the-art works in the bulk of the literature. TextileTube comprises a more challenging dataset which is quite specific and, at the same time, very useful for the application of evidence search in child sexual abuse crime cases. Nonetheless, it can as well represent other applications like clothing search for textile marketing.

4.2.2. Experimental setup

Training data preparation

To train our network architecture, we prepared training images from TextileTube, CBT, Outex and USPtex datasets. The images from TextileTube consist of texture images extracted from real-world images, whereas the images from the Outex, USPtex and CBT are well defined textured patterns with various colours. Thus, the network can learn texture features along with colours. To prepare the training set, we considered the training samples provided in the Outex and USPtex datasets, and the complete CBT dataset. In addition, from the TextileTube dataset, we randomly selected 30 images taken from different classes. Besides, since there are few training examples, we created a larger set of training images by augmenting them via several random transformations such as Gaussian distortion, rotation, skewing, tilting and flipping. As a result, we generated a larger training set consisting of $60,000$ images.

Retrieval framework setup

We trained the retrieval network using DFT based blended $B(x, y)$ images by applying transfer learning to the encoder. During training, we froze the first five convolutional layers of the ImageNet initialised VGG encoder, and the decoder learns to map the latent space representation back to the input training images. We chose Adam optimizer with an initial learning rate of 0.0001, and a batch size of 16 images so that the full GPU memory could be utilized. Furthermore, we stopped our training when we observed that the images generated at the end of the decoder were visually similar to the input images. This means that the network has learnt the latent space representation of the input image in order to reconstruct the original input through the decoder. However, for Outex, USPtex and Stex datasets, the RPN was not applied since all the images in these three datasets are composed of single textured patterns.

Classification framework setup

Likewise the training of the retrieval network, we used DFT based blended $B(x, y)$ images to train the dense layers of the classification model. We trained Outex dataset with the augmented training data and evaluated the performance on the test data. For USPtex dataset, we carried out the same procedure. We measured the performance of the proposed DFTD texture descriptors on Outex and USPtex datasets for texture classification, and we evaluated the performance in terms of accuracy. We define accuracy as the percentage of test images classified as true positive.

Evaluation metrics

To compare our approach with other state-of-the-art methods, we used two different evaluation protocols. The evaluation protocol used by the Outex, Stex and USPtex is Average retrieval rate (ARR) which was suggested in (Pham, 2018). However, the state-of-the-art results concerning the TextileTube dataset are provided based on $precision@k$, as proposed in (García-Olalla et al., 2018).

Evaluation metric for Outex, USPtex and Stex: We evaluated the performance for texture-based instance retrieval using average retrieval rate (ARR). Let N be the total number of images in the dataset and R_q be the number of relevant images for a query q. Let $m_{(q,k)}$ be the number of correctly retrieved images found within the first k retrievals for a query q. ARR in terms of the number of retrieved images is given by:

$$ARR = \frac{1}{N}\sum_{q=1}^{N}\frac{m_{(q,k)}}{R_q}. \tag{4.9}$$

Evaluation metric for TextileTube dataset We evaluated the retrieved top k texture images concerning a given query image according to the precision in a ranking-based criterion. The creation of the hit list was defined in Section 4.1.2. The precision@k is defined as:

$$\textit{Precision}@k = \frac{\sum_{i=1}^{k} R(i)}{k}, \tag{4.10}$$

where $R(i)$ denotes the relevance between the i^{th} ranked image in the hit list and the query. If the bounding box of the detected texture region in the retrieved image intersects the ground truth, R is set to 1; else to 0. Fig. 4.9 illustrates this scenario. In our experiments, we consider $k = \{1, ..., 40\} | k \in \mathbb{N}$.

4.2.3. Experimentation and results

In this section, we present the experiments carried out and the results obtained in comparison with the state-of-the-art approaches.

Experiments on Outex, USPtex and Stex datasets

Results and comparison for texture classification: We tested our approach for texture classification to evaluate the performance of the proposed DFTD descriptors against state-of-the-art works since texture classification is wider explored than texture retrieval. We followed the method and experimental set-up explained above. In Table 4.1, we present the results, the best accuracy reported in the literature

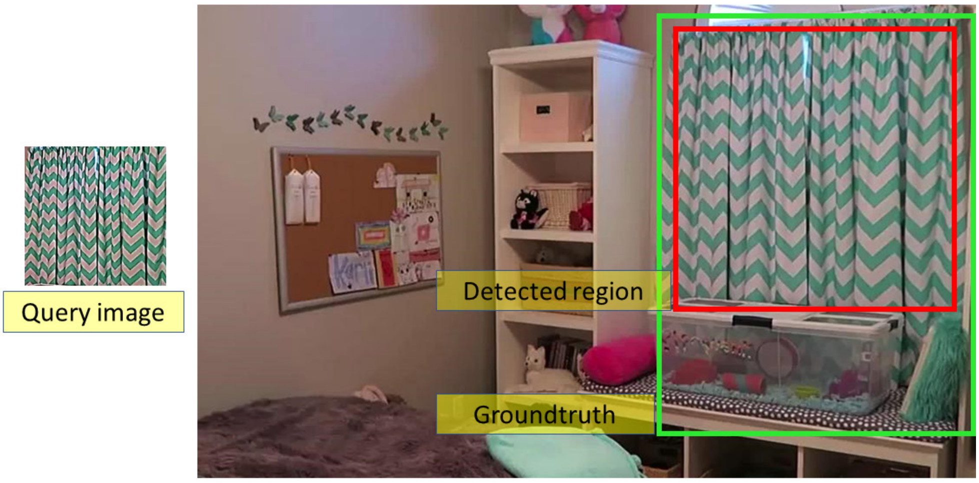


Figure 4.9: The green square shows the groundtruth, and the red one presents the detected region in relation to the query image. Since there is an overlap between both regions, $R(i)$ would be set to 1.

Table 4.1: Comparison of our proposed approach in terms of accuracy (in percentage) on Outex and USPtex datasets for texture classification.

Method	Outex	USPtex
LESTP Guo et al. (2015)	78.00	82.41
LECTP Guo et al. (2015)	79.06	83.10
PCANet Guo et al. (2015)	76.04	83.65
LQP Guo et al. (2015)	81.49	87.83
Multifractals Napoletano (2017)	75.07	68.76
Fourier Napoletano (2017)	82.21	71.16
ARCS-LBPt Napoletano (2017)	85.70	88.85
Chess-Pattern Tuncer, Dogan and Ataman (2019)	88.9	-
Tuncer *et al.* Tuncer, Dogan and Ertam (2019)	89.62	93.83
DFTD (Ours)	**90.20**	**95.62**

was 89.62% by Chess-pattern method (Tuncer, Dogan and Ataman, 2019) on Outex and 93.83% by Tuncer, Dogan and Ertam (2019) on USPtex. In contrast, we have achieved an accuracy of 90.20% and 95.62% on Outex and USPtex, respectively, outperforming all the reported results.

Table 4.2: Comparison of our proposed approach in terms of average retrieval rate (in percentage) on Outex, USPtex and Stex datasets for texture-based instance retrieval.

Method	Outex	USPtex	Stex
DDBTC Guo et al. (2015)	66.82	74.97	44.79
CNN-AlexNet Napoletano (2017)	69.87	83.57	68.84
CNN-VGG16 Napoletano (2017)	72.91	85.03	74.92
CNN-VGG19 Napoletano (2017)	73.20	84.22	73.93
LED Pham (2018)	75.14	87.50	76.71
SLED Pham (2018)	75.96	88.60	77.88
MS-SLED Pham (2018)	76.15	89.74	79.87
DFTD (Ours)	**80.36**	**90.25**	**81.02**

Results and comparison for texture-based instance retrieval: In Table 4.2, we show the performance of our approach for texture-based instance retrieval regarding ARR metric on Outex, USPtex and Stex datasets, and compare it to other state-of-the-art methods. We observed that descriptors based on learned CNN representations, i.e. CNN-VGG19 (Napoletano, 2017), yielded competitive results as compared to handcrafted features, such as DDBTC (Guo et al., 2015). It is also noticeable that our proposed approach outperformed all the methods by obtaining an ARR of 80.36% on Outex, 90.25% on USPtex and 81.02% on Stex datasets, whereas the best reported results in the literature were yielded by MS-LED method (Pham, 2018) with ARR of 76.15%, 89.74% and 79.87%, respectively.

Experiments on TextileTube dataset

Results and comparison with state-of-the-art methods: We used both types of queries, the original and the *New* queries, to evaluate our method. Fig. 4.10 summarizes the results obtained by our proposed method which outperforms the other approaches in terms of precision@k. RPN+DFTD indicates the results obtained by the original queries, whereas RPN+DFTD(new) represents the results obtained using the *New* queries. The results using ALBP+HCLOSIB, ALBP, Faster R-CNN, HOG, HOG+CLOSIB and HOG+HCLOSIB are taken from (García-Olalla et al., 2018).

Besides Faster R-CNN, the rest of the methods do not rely on modern deep learning techniques. In order to overcome this issue and establish a stronger baseline, we also considered R-FCN and fully convolutional one-stage object detection (FCOS) (Tian et al., 2019) networks. Even though both Faster R-CNN and R-FCN are region-based detectors and use ResNet-101 for feature extraction, RFCN demonstrated to be $20\times$ faster than Faster R-CNN. In contrast, FCOS can be exploited to generate multiple levels of intermediate proposals, where each level have proposals of differ-

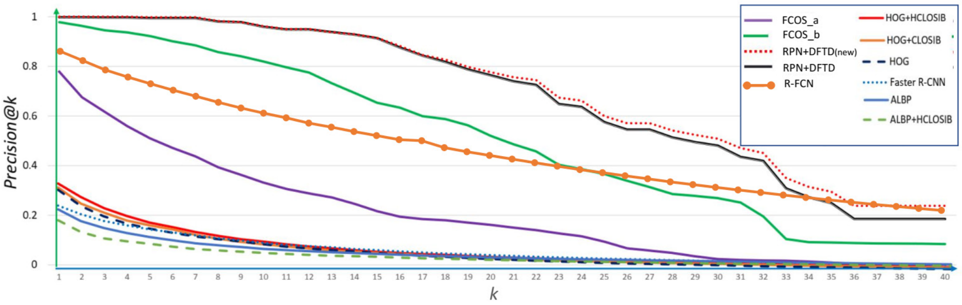


Figure 4.10: Precision@k of state-of-the-art methods, recent baselines and our method (RPN+DFTD) on TextileTube dataset. RPN+DFTD represents the results with the original queries and RPN+DFTD (New) with the New queries.

ent sizes, increasing in this way the scope of the query image search across a larger number of regions. For R-FCN, we took all 300 proposals as candidate regions to check the presence of the texture queries. Regarding FCOS, we used two approaches for the generation of proposals: (a) FCOS$_a$, where the proposals are obtained at the final classification layer, which results in around 100 proposals; and (b) FCOS$_b$, where we have considered all the raw proposals from the intermediate layers up to the final layer, which results in more than 2000 proposals. The best result for these state-of-the-art methods was yielded by FCOS$_b$ because a query image descriptor can be compared against more local descriptors, but it comes at the cost of a high computational time. To evaluate our proposed descriptors and retrieval framework, we used the RPN of R-FCN to speed up the generation of texture patches.

Fig. 4.10 illustrates the precision@k of the methods proposed in (García-Olalla et al., 2018) (ALBP+HCLOSIB, ALBP, Faster R-CNN, HOG, HOG+CLOSIB and HOG+HCLOSIB), the considered baseline methods (R-FCN, FCOS$_a$ and FCOS$_b$), and the proposed method using the original queries (RPN+SFD) and the *New* queries (RPN+SFD(new)). The baseline methods and our proposed method outperformed the results presented in (García-Olalla et al., 2018). FCOS$_b$ yielded higher precision@k than FCOS$_a$ and R-FCN. It can be seen that for a similar number of proposals, R-FCN outperformed FCOS$_a$. Our proposed method using the original queries (RPN+DFTD) achieved higher precision@k than the methods proposed in (García-Olalla et al., 2018) and the considered baseline methods, which also utilize the original queries, for all values of $k = \{1, ..., 40\} | k \in \mathbb{N}$. Specially, the proposed method shows relevant improvement with respect to the rest of the methods for larger values of k. The results for the retrieval of the *New* queries (RPN+DFTD(new)) and the original queries (RPN+DFTD) using the proposed method are quite similar. We will later comment about them concerning the numerical results.

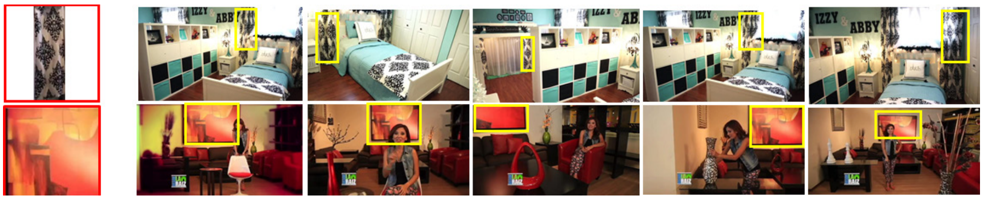


Figure 4.11: Top-5 correctly retrieved images on TextileTube dataset using RPN+DFTD(New). The queries are shown on the first column and are framed with a red box, and the proposals that lead to the retrievals are framed with a yellow box.

In Fig. 4.11, we show the top-5 retrievals with respect to two sample query images using RPN+DFTD (new) on TextileTube dataset. In each of the rows, the images framed with red bounding boxes are the query images, and the images in the left of the queries are the top-5 retrieved ones. It can be seen that the query images represent two different texture patches, which we compared with the proposals of the database images to retrieve the ones where the query patch might be present. If there is any proposal that is likely to match with the query, then it would have a high CS score, and we would retrieve the image to which that proposal corresponds. Also, it can be seen that the queried texture patterns are successfully detected in the retrieved database images, which are sorted in descending order corresponding to CS scores of the detected proposals. Moreover, the overlaid yellow boxes illustrate the detected proposals, and we can observe that the query images successfully match with them.

Table. 4.3 illustrates precision@k for top-k (1 to 7) retrievals. Using our method RPN+DFTD with the original queries, we obtained a precision@k of 99.7% at $k = 1$ in comparison to the highest precision of 37.5% in the literature with HOG+HCLOSIB. Furthermore, using our baseline FCOS$_a$ and FCOS$_b$, we achieved a precision of 77.8% and 97.7% for $k = 1$, respectively. The results obtained by our method indicate that, for every query search, the first retrieval is a hit with a probability of 99.7%. As k increases, the proposed method achieved a higher precision with respect to the rest of the methods. For example, for $k = 7$, the proposed method yielded a 99.7% of precision, whereas FCOS$_b$ obtained 88.4%, and HOG+HCLOSIB 21.7%. For the retrieval of the *New* queries, the proposed method obtained 100% precision for top-4 retrievals. For higher values of k, the precision starts slightly decreasing concerning the original queries. For large values of k, the absence of contour and surrounding information in the *New* queries, which contain only texture patterns, possibly provokes the fall in precision.

In Table 4.4, we present the arithmetic mean of precision@k for three different intervals, where k ranges from 1 to 10, 1 to 20 and 1 to 30. We achieved an arithmetic mean of 99.2%, 93.2% and 67.9% using RPN+DFTD for the three intervals,

Table 4.3: Precision@k, in percentage, for texture descriptors reported in (García-Olalla et al., 2018), the baseline and the proposed method on TextileTube dataset. Results highlighted in bold mark the best results achieved for each value of k

Descriptor	1	2	3	4	5	6	7
HOG+HCLOSIB, 2018	37.2	32.8	29.3	26.8	24.7	23.2	21.7
HOG+CLOSIB, 2018	35.9	30.8	27.9	25.4	23.8	22.5	20.8
HOG, 2018	35.2	30.0	26.7	24.4	22.8	21.5	20.2
Faster R-CNN, 2018	30.1	27.4	25.2	23.8	22.5	21.6	20.6
ALBP+HCLOSIB, 2018	28.9	25.2	23.0	21.4	20.1	19.0	18.1
ALBP+CLOSIB, 2018	25.5	21.7	19.6	18.7	17.9	17.0	16.2
R-FCN	82.1	79.0	75.9	73.5	71.3	69.2	67.2
FCOS$_a$	77.8	67.5	61.7	55.8	50.9	47.0	43.7
FCOS$_b$	97.7	96.2	94.5	93.6	92.2	90.0	88.4
RPN+DFTD	99.7	99.7	99.6	99.6	99.4	99.4	99.4
RPN+DFTD(new)	**100.0**	**100.0**	**100.0**	**100.0**	**99.7**	**99.7**	**99.7**

Table 4.4: Arithmetic mean of precision at k (M) for intervals ranging from 1 to 10, 1 to 20 and 1 to 40 on TextileTube dataset. The results with RPN+DFTD, both the original and New queries, are shown in bold.

Descriptor	M(1-10)	M(1-20)	M(1-40)
HOG+HCLOSIB	24.8	19.5	15.1
HOG+CLOSIB	23.7	18.8	14.7
HOG	23.1	18.5	14.3
Faster R-CNN	22.5	18.8	15.3
ALBP+HCLOSIB	18.1	15.7	13.5
ALBP+CLOSIB	19.6	17.0	14.6
R-FCN	70.8	62.7	49.6
FCOS$_a$	51.3	36.7	20.7
FCOS$_b$	90.4	78.0	50.8
RPN+DFTD	**99.2**	**93.2**	**67.9**
RPN+DFTD(new)	**99.5**	**94.5**	**69.8**

respectively, followed by FCOS$_b$ at 90.4%, 78.0% and 50.8%, respectively. We can notice that the improvement is significantly high as compared to the reported results in García-Olalla et al. (2018) of 24.8%, 19.5% and 15.1% with HOG+HCLOSIB. For the new queries, the results were higher in the three intervals with arithmetic mean precision of 99.5%, 94.5% and 69.8%, respectively.

All the results clearly show that, in terms of precision, our proposed method

RPN+DFTD outperformed the retrieval results obtained using HOG+HCLOSIB, from 37.2% to 99.7% for $k = 1$, and from 18.6% to 97.7% for $k = 10$. Such results prove that the activation generated by the latent space node of the VGG encoder using Fourier based images can efficiently represent the texture of a patch. Moreover, since our architecture is based on an autoencoder, we were able to train a large number of texture classes by keeping the latent space node compact with a constant dimension of 512. Furthermore, the proposals generated by the RPN cover well the regions of interest of the images on the dataset, and thus, it enables the localization of the texture query patches.

4.3. Conclusions

We presented a new deep Fourier texture descriptor DFTD based on the discrete Fourier transform and the latent space representation of a VGG autoencoder. Besides, we also proposed a framework for texture-based instance retrieval. We used a RPN to identify regions of interest in the image dataset which were given as an input to a VGG autoencoder. The VGG autoencoder was trained with images obtained from a weighted linear combination of DFT magnitude spectrum and spatial images. The RPN proved to be very useful to identify texture regions in images with several texture patterns, such as indoor scene images.

In this work, we considered two different types of datasets to test our approach. On the one hand, we experimented using Outex, Stex and USPtex, which are similar datasets containing images with only one texture pattern per image but they are broadly used for texture retrieval and classification. And, on the other hand, we considered TextileTube dataset which comprises indoor scene images with lots of different texture patterns. We selected this dataset because it represents a useful case scenario related to CSA digital content, in which apart from faces, objects, etc., textures also may represent a clue to find pieces of evidence among already known cases of CSA.

To evaluate the performance of the proposed DFTD descriptor, we also carried out experiments for texture classification since it is a problem broadly studied among computer vision researchers. We assessed the performance on Outex, STex and USPtex datasets and compared our proposal with the recent top methods for texture classification.

DFTD is a quite compact descriptor in the form of a 512-dimensional vector, which makes the matching computationally inexpensive. Furthermore, the obtained results on the four datasets demonstrate that the proposed architecture and DFTD descriptor are effective for retrieval and classification purposes, yielding state-of-the-art performance.

Chapter 5

Indoor scene recognition

This chapter has been intentionally deleted because it is subject to be patented and/or published.

Abstract: Indoor scene recognition is a challenging and growing task in the field of computer vision. Although Convolutional Neural Networks can achieve outstanding results in outdoor scene recognition, their performance lacks similar robustness in the recognition of indoor scenes. This is due to the high spatial variability in semantic cues (e.g. objects), and due to the presence of similar objects throughout different scene categories. To counteract these issues, we propose DeepScenePip (DSP), a pipeline with three modules: *object-centric*, *objects-to-scene* and *scene-centric*, which independently focus on local and global scene content, respectively. The proposed pipeline has three novel components. Firstly, it produces an image caption from the labels of the recognized object to predict scenes using a natural language processing approach. Secondly, it relies on a weight function that combines object and scene information for an overall scene prediction. Thirdly, it includes a query expansion technique for scene retrieval. We evaluated our approach for indoor scene recognition and indoor scene retrieval on three public datasets: MIT-67 Indoor, NYU-v2 and Hotels-50k. The accuracy achieved (MIT-67 Indoor = 94.5%, NYU-v2 = 74.5% and top-1 accuracy on the Hotels-50k: 10.1% without occlusion and 7.8% with medium occlusion) demonstrated the effectiveness of the proposed pipeline, which also significantly outperforms existing state-of-the-art approaches.

Chapter 6

Conclusion and outlook

6.1. Work summary

Over the last few years, with the advent of social networks and mobile phones with high-quality cameras, there has been a rapid proliferation of visual content on the internet all over the world. Due to this, image search has become a challenging problem, hindering the efficacy of CBIR applications. In such applications, search engines take a query image and try to find similar images by identifying various shapes and patterns across images. One of the most prominent applications which we addressed in this thesis is to retrieve images for forensic evidence analysis for solving crime scenes. In a crime scene, the clues derived from images can empower the investigative work of forensic departments. In these cases, the main objective is to retrieve not the same category but the same instance as the query. CBIR for instance-level retrieval applications can aid in uncovering various crimes by linking similar images or videos in a database to create concrete evidences for crime scene investigation. In addition, recognizing indoor scenes in images can also help in forensic investigation by identifying scene categories and also by retrieving the same or related scenes. Concerning the presented scenarios, this thesis establishes frameworks for addressing CBIR and indoor scene recognition tasks by employing deep learning technologies.

Particularly, we presented three main lines of work in this thesis: (1) colour neural descriptors to describe objects with colour robustness and (2) texture descriptors based on blended discrete Fourier transformed images, both for CBIR, and (3) a new pipeline that combines local and global scene content for indoor scene prediction and retrieval.

The remainder of this chapter presents a summary of the contributions and possible future work lines.

6.2. Summary of contributions

In this section, we sum up the contributions that were proposed with respect to each of the three lines of our research work.

(1) Colour neural descriptors for instance retrieval:

- *We introduced new colour neural descriptors using the activations generated from a pre-trained deep CNN.* In Chapter 3, we experimented with the deep neural features obtained from a CNN for CBIR. We demonstrated that CNN features can efficiently represent an image as a descriptor. However, the raw descriptors derived from a pre-trained CNN might not be adequate for instance retrieval if the colour of the query to retrieve is a relevant feature. Therefore, we created colour descriptors by extracting CNN features independently with respect to R, G and B colour channels, and then we constructed colour models to create robust descriptors.

- *We built a hybrid architecture composed of two different CNNs, and we used it successfully as an image retrieval pipeline without employing fine-tuning techniques.* In Chapter 3, we presented an architecture for the image retrieval pipeline which is composed of two different CNNs. The first one is a RPN from a ResNet network, which is responsible for generating object proposals in an image to facilitate localized instance search. The second network is a VGG-16 network pre-trained with the ImageNet dataset and serves as a feature extractor for both the query and the proposals. Both of the networks are unified into a single framework, where the candidate proposals generated by the RPN are given directly as an input to the VGG-16 to compute colour descriptors. In addition to these experiments, this study also provides a performance analysis of the proposed method in terms of computational and space complexity.

    - **Experimental demonstration.** *We demonstrated experimentally that the proposed colour neural descriptors outperform the state-of-the-art results on four datasets for image retrieval.* The presented architecture and the colour descriptors in Chapter 3 are evaluated using four datasets: COIL-100, INSTRE-M, Paris 6K and Revisiting-Paris 6k. Also, we compared the colour descriptors with each other to select the one that performs best in terms of accuracy. NE-C outperformed the rest, yielding mAPs of 81.70%, 82.02%, 78.8% and 97.9% in Paris 6K, Revisiting-Paris 6k, INSTRE-M and COIL-100 datasets, respectively. The obtained mAPs surpasses the results of some relevant state-of-the-art methods.

(2) Texture descriptors based on blended discrete Fourier transformed images for CBIR:

- *We introduced a novel texture descriptor named as deep Fourier texture descriptor (DFTD).* In Chapter 4, we presented a new texture descriptor with the aim of enhancing the retrieval of queries that are made of texture patches in which the shape and contour of objects are not relevant. The features were extracted from the latent space layer of a convolutional autoencoder whose inputs are the outcomes of blending the spatial information of the images with the magnitude spectrum of the DFT image. The blended images were used to train the convolutional autoencoder network, and after training, the encoder part is considered for feature extraction, where the latent space layer defines the DFTD.

- *We proposed a CBILIR framework for texture retrieval which uses an RPN to generate prospective texture regions that are fed into an autoencoder.* In Chapter 4, we presented a framework for CBIR which is comprised of three stages. At the first stage, we extracted the query features by applying the DFT to the query image to obtain the magnitude spectrum, and then we performed a pixel-wise weighted combination to the query and its DFT magnitude spectrum to obtain a blended query image. The blended image was fed to the trained convolutional encoder model to extract the latent space vector DFTD, which represents the query image. In the second stage, we integrated an RPN with the convolutional encoder to generate region proposals. The blended region proposals were fed to the encoder to generate their respective DFTD descriptors. The DFTD descriptors of all the proposals were stored in a database together with their image label and were compared with the query descriptor. Finally, in the third stage, we retrieved images by comparing the query descriptor with the database of descriptors.

    - **Experimental demonstration.** *We evaluated the proposed texture-based image retrieval approach in the context of a real-world application*: The method proposed in Chapter 4 can be applied to a variety of visual search problems such as for image, instance and object retrieval. In this thesis, we addressed an specific problem, that is crime scenes investigation for forensic analysis where the images are mainly comprised of various textured patterns. We assessed the proposed DFTD descriptor for texture-based image retrieval and obtained average retrieval rates of 80.36%, 90.25%, and 81.02% on the Outex, USPtex, and Stex datasets. In addition, we also experimented with the TextileTube dataset, that consist images from a real world scenario. In this case, we calculated arithmetic means of precision@k for three different intervals ([1-10],[1-20] and [1-40]) and obtained precision values of 99.2%, 93.2% and 67.9%. Moreover, the results obtained in these four datasets outperformed the state-of-the-art results.

(3) New pipeline that combines local and global scene content for indoor scene prediction and recognition:

- *We introduced a novel architecture known as DeepScenePip or DSP, which combines both object and global features*: In chapter 5, we introduced a novel hybrid architecture, which is a pipeline for indoor scene prediction and retrieval, that combines object and global features. The architecture is primarily composed of three parts: the *object-centric*, *object-to-scene* and the *scene-centric* modules, with a common base network. The *object-centric* module is composed of a detector and a classifier network, which we trained end-to-end for generating object labels and their corresponding features. The module is further connected to the *objects-to-scene* module, which consists of a three-layered neural network whose inputs are captions derived from the *object-centric* module. Finally, the *scene-centric* module extends the base model with two fully connected layers for the task of scene classification and global scene feature extraction. This architecture does not require an increase in the depths of the networks involved, neither an increase in the training data, and nor inference time.

- *We proposed a new technique based on natural language processing to predict a scene category from captions generated from the object labels recognized in a given image*: The technique comprehends captioning the object labels followed by vectorization using word2vec. In Chapter 5, the *objects-to-scene* module was presented to recognize scenes from the object labels derived by the *object-centric* module. To do so, we devised a technique to caption the images using the object labels, in which captions are the attributes specific to scene labels. We then converted the captions into vectors of dimension 390 using word2vec and then they were fed into a neural network for scene prediction.

- *We presented a weight function named as weighted combination of object-centric and scene-centric modules (WCOS) that combines object and scene information for an overall scene prediction*: In order to overcome the individual limitations of using only the object-level or the global features, in Chapter 5, we proposed a method to combine the prediction obtained from the *objects-to-scene* and the *scene-centric* modules using a weight function to improve the accuracy of scene prediction. The reason for using the weight function is that, on the one hand, object-level features alone may not produce the desired results when inter-class scenes have common objects. And, on the other hand, the global features are more generic due to the presence of similar layouts in different classes of images, which may result in incorrect predictions.

- *We introduced a retrieval approach for query-based scene retrieval of indoor images*: To do that, we use the DSP architecture to extract and combine features from the *object-centric* and the *scene-centric* modules to represent queries and dataset images. The object features correspond to the neural activation of the ROIs generated by the *object-centric* module, whereas the global features are generated at the last FC layer of the *scene-centric* module. Besides, we created image dictionaries where the scene and objects features along with their labels are stored in a database, which is used for matching purposes. In addition, we also developed a query object-bin expansion method for scene retrieval that smooths the issues linked to the change of viewpoint of the objects and scene images.

- **Experimental demonstration.** *We demonstrated that the combination of object features and global features yields higher performance in both scene recognition and retrieval tasks*. The proposed framework was tested using three different datasets, the indoor scene recognition framework was tested using MIT-67 and NYU datasets, and the scene retrieval one using the Hotels-50k dataset. The accuracy achieved (MIT-67 Indoor = 94.5%, NYU-v2 = 74.5% and top-1 accuracy 10.1% without occlusion and 7.8% with medium occlusion on the Hotels-50k) demonstrated the effectiveness of the proposed method, which also significantly outperforms existing state-of-the-art approaches.

6.3. Future work

In this section, we present the main research lines that remain open and that could be interesting to address in the future. The following new research lines can be added as extensions to the current work proposed in the thesis.

- *Optimization of the architecture in Chapter 3 for faster feature extraction.* The architecture presented for the colour descriptor creation is complex due to the presence of three stages at the feature extraction process, one for each colour channel. Due to which, it has a time and space complexity relatively higher than a general CNN architecture. In order to address this issue, our next step is to simplify the architecture by devising strategies for reducing the number of layers and stages.

- *Exploration of different colour models.* To increase the discriminative power of the descriptors, other colour models can be explored and adapted to create new descriptors. Furthermore, new equations adapted for different colour spaces can be formulated for generating more robust descriptors.

- *Improvement of the DFTD descriptor.* As a part of future research, we will modify the proposed DFTD descriptor to enhance rotation invariance by taking into account the Fourier coefficients. Moreover, we will also leverage various autoencoder architectures that may account for the enhancement of the DFTD descriptor such as transformers (Tan et al., 2021; El-Nouby et al., 2021).

- *Employment of deep hashing for faster instance retrieval*: Deep hashing is an effective way of processing high dimensional data due to faster query speed and low memory cost. At present, the CBIR approaches presented in the thesis store data using HDF5 format, which yields faster retrieval in database with millions of images. However, in order to scale and speed up the retrieval process for a database with billions of images, we would employ deep hashing (Lu et al., 2019; Chen et al., 2020; Lu et al., 2020).

- *Exploration of new ways to caption images in the object-centric approach for scene recognition.* The approach presented in Chapter 5 uses captions generated by an object detector to enhance the overall scene prediction. The performance depends on the quality of the generated caption and the word embeddings, which are given as an input to a neural network. To further improve the performance, we will devise new approaches to enrich the word embeddings, and also, different captioning techniques will be explored.

- *Extension of the current indoor scene recognition framework to predict outdoor scenes.* The presented scene recognition method is currently limited to indoor scene images. However, the same approach can be generalised for outdoor scene recognition by training the proposed architecture with outdoor scene images and objects. In the future, we will address the outdoor scene recognition by training and testing with outdoor scene images.

- *Examine additional object detection networks.* Object detection is the key component for both instance retrieval and scene recognition. Henceforth, performance heavily relies on the accuracy and the speed of object detection. So, for our future implementations, we will examine other object detection frameworks to be incorporated to the proposed instance retrieval and scene recognition architectures.

Chapter 7

Conclusiones y perspectiva

7.1. Resumen del trabajo

En los últimos años, con la llegada de las redes sociales y los teléfonos móviles con cámaras de alta calidad, se ha producido una rápida proliferación de contenidos visuales en Internet en todo el mundo. Debido a ello, la búsqueda de imágenes se ha convertido en un problema difícil, que dificulta la eficacia de las aplicaciones de *recuperación de imágenes basada en el contenido* (CBIR, del inglés, *content-based image retrieval*). En estas aplicaciones, los motores de búsqueda parten de una imagen de consulta e intentan encontrar imágenes similares identificando diversas formas y patrones en las imágenes. Una de las aplicaciones que abordamos en esta tesis de forma más destacable es la recuperación de imágenes para el análisis de pruebas forenses en la investigación de escenas de crímenes. En una escena del crimen, las pistas derivadas de las imágenes pueden potenciar el trabajo de investigación de los departamentos forenses. En estos casos, el objetivo principal es recuperar no la misma categoría sino la misma instancia que la consulta. La CBIR en aplicaciones que pretenden la recuperación a nivel de instancia puede ayudar a descubrir delitos ya que permite vincular imágenes o vídeos similares en una base de datos para crear evidencias concretas en la investigación de la escena del crimen. Además, el reconocimiento de escenas interiores en imágenes también puede ayudar en la investigación forense tanto al identificar las categorías de las escenas como al recuperar las mismas escenas o las relacionadas. En relación con los escenarios presentados, esta tesis establece marcos para abordar las tareas de CBIR y de reconocimiento de escenas interiores empleando tecnologías de aprendizaje profundo.

En particular, presentamos tres líneas principales de trabajo en esta tesis: (1) descriptores neuronales de color para describir objetos de forma que el color sea descrito de forma robusta y (2) descriptores de textura basados en imágenes donde se combina la imagen original con la imagen obtenida al aplicarle la transformada discreta de Fourier, ambos para CBIR, y (3) un nuevo *pipeline* que combina el contenido local y global de la escena para la predicción y recuperación de escenas de interior.

El resto del capítulo presenta un resumen de las contribuciones y las posibles líneas de trabajo futuras.

7.2. Conclusiones generales

En este apartado, resumimos las aportaciones resultantes respecto a cada una de las tres líneas del trabajo de investigación.

(1) Descriptores neuronales de color para la recuperación de instancias:

- *Introducimos nuevos descriptores neuronales de color utilizando las activaciones generadas a partir de una red neuronal convolucional (CNN, del inglés convolutional neural network) profunda preentrenada*. En el Capítulo 3, experimentamos con las características neuronales profundas obtenidas a partir de una CNN para CBIR. Demostramos que las características de la CNN pueden representar eficientemente una imagen como un descriptor. Sin embargo, los descriptores brutos derivados de una CNN preentrenada podrían no ser adecuados para la recuperación de instancias si el color de la consulta a recuperar es una característica relevante. Por lo tanto, creamos descriptores de color extrayendo las características de la CNN de forma independiente con respecto a los canales de color R, G y B, y luego construimos modelos de color para crear descriptores más robustos a esta característica.

- *Construimos una arquitectura híbrida compuesta por dos CNN diferentes, y la utilizamos con éxito como canal de recuperación de imágenes sin emplear técnicas de ajuste fino.* En el Capítulo 3, presentamos una arquitectura para recuperar imágenes compuesta por dos CNN diferentes. La primera es una red de propuesta de regiones (RPN, del inglés *region proposal network*) de una red ResNet, que se encarga de generar propuestas de objetos en una imagen para facilitar la búsqueda de instancias localizadas. La segunda red es una red VGG-16 preentrenada con el conjunto de datos ImageNet y sirve como extractor de características tanto para la consulta como para las propuestas. Ambas redes se unifican en un único marco, en el que las propuestas candidatas generadas por la RPN conforman directamente las entradas a la VGG-16 para calcular los descriptores de color. Además de estos experimentos, este estudio también proporciona un análisis del rendimiento del método propuesto en términos de complejidad computacional y espacial.

  - **Demostración experimental**. *Hemos demostrado experimentalmente que los descriptores neuronales de color propuestos superan los resultados del estado del arte en cuatro conjuntos de datos para la recuperación de imágenes.* La arquitectura presentada y los descriptores de color del Capítulo 3 se evalúan utilizando cuatro conjuntos de datos: COIL-100, INSTRE-M, París 6K y Revisiting-Paris 6k. Además, comparamos los descriptores de color entre sí para seleccionar el que mejor se comporta en términos de precisión. NE-C superó al resto, obteniendo precisiones medias promediadas (mAPs, del inglés mean average precision) de $81,70\%$, $82,02\%$, $78,8\%$

y 97, 9% en los conjuntos de datos Paris 6K, Revisiting-Paris 6k, INSTRE-M y COIL-100, respectivamente. Los mAPs obtenidos superan los resultados de algunos métodos relevantes del estado del arte.

(2) Descriptores de textura basados en imágenes combinadas con transformada discreta de Fourier para CBIR:

- *Introducimos un nuevo descriptor de textura denominado descriptor de textura de Fourier profundo (DFTD).* En el Capítulo 4, presentamos un nuevo descriptor de textura con el objetivo de mejorar la recuperación de consultas que están hechas de parches de textura en los que la forma y el contorno de los objetos no son relevantes. Las características se extrajeron de la capa de espacio latente de un autocodificador convolucional cuyas entradas son los resultados de mezclar la información espacial de las imágenes con el espectro de magnitud de la imagen DFT. Las imágenes mezcladas se utilizaron para entrenar la red del autocodificador convolucional y, tras el entrenamiento, se considera la parte del codificador para la extracción de características, donde la capa del espacio latente define la DFTD.

- *Propusimos un marco de recuperación de imágenes basadas en contenido a nivel instancia (CBILIR, del inglés content-based instance level image retrieval) para la recuperación de texturas que utiliza una RPN para generar posibles regiones de textura que conforman las entradas de un autocodificador.* En el Capítulo 4, presentamos un marco para CBIR que consta de tres etapas. En la primera etapa, extrajimos las características de la consulta aplicando la DFT a la imagen de la consulta para obtener el espectro de magnitud, y luego realizamos una combinación ponderada a nivel de píxel (*blending*) de la consulta y su espectro de magnitud DFT para obtener una imagen de consulta conteniendo la información de ambas. Dicha imagen combinada se introdujo en el modelo codificador convolucional entrenado para extraer el vector espacial latente, denominado DFTD (del inglés *discrete Fourier transform descriptor*), que representa la imagen de consulta. En la segunda etapa, integramos una RPN con el codificador convolucional para generar propuestas de región. Las propuestas de región combinadas se introducen como entradas en el codificador para generar sus respectivos descriptores DFTD. Los descriptores DFTD de todas las propuestas se almacenaron en una base de datos junto con su etiqueta de imagen y se compararon con el descriptor de consulta. Por último, en la tercera etapa, recuperamos las imágenes comparando el descriptor de la consulta con la base de datos de descriptores.

  - **Demostración experimental.** *Evaluamos el modelo de recuperación de imágenes basado en la textura propuesto en el contexto de una aplicación del* *mundo real*: El método propuesto en el Capítulo 4 puede aplicarse a una variedad de problemas de búsqueda visual, como la recuperación de imágenes, instancias y objetos. En esta tesis, abordamos un problema específico, que es la investigación de escenas de crímenes para el análisis forense, donde las imágenes se componen principalmente de varios patrones de textura. Evaluamos el descriptor DFTD propuesto para la recuperación de imágenes basadas en texturas y obtuvimos tasas de recuperación medias de $80,36\%$, $90,25\%$ y $81,02\%$ en los conjuntos de datos Outex, USPtex y Stex. Además, también experimentamos con el conjunto de datos TextileTube, que consiste en imágenes de un escenario del mundo real. En este caso, calculamos las medias aritméticas de precisión@k para tres intervalos diferentes ([1-10],[1-20] y [1-40]) y obtuvimos valores de precisión de $99,2\%$, $93,2\%$ y $67,9\%$. Además, los resultados obtenidos en estos cuatro conjuntos de datos superaron los resultados del estado del arte.

(3) Nuevo *pipeline* que combina el contenido local y global de la escena para la predicción y el reconocimiento de escenas en interiores:

- *Introducimos una novedosa arquitectura denominada DeepScenePip o DSP, que combina características de objeto y globales*: En el Capítulo 5, introducimos una novedosa arquitectura híbrida, que es un *pipeline* para la predicción y recuperación de escenas en interiores, que combina características de objeto y globales. La arquitectura se compone principalmente de tres partes: los módulos *object-centric*, *objects-to-scene* y *scene-centric*, además de una red que conforma una base común a los módulos. El módulo *object-centric* está compuesto por un detector y una red clasificadora, que entrenamos de principio a fin para generar etiquetas de objetos y sus correspondientes características. Este módulo está conectado a su vez con el módulo *objects-to-scene*, que consiste en una red neuronal de tres capas cuyas entradas son las descripciones de las imágenes derivadas del módulo *object-centric*. Por último, el módulo *scene-centric* amplía el modelo base con dos capas totalmente conectadas para la tarea de clasificación de escenas y la extracción de características globales de la misma. Esta arquitectura no requiere un aumento de la profundidad de las redes implicadas, ni un aumento de los datos de entrenamiento, ni del tiempo de inferencia.

- *Proponemos una nueva técnica basada en el procesamiento del lenguaje natural para predecir la categoría de una escena a partir de las descripciones de una imagen generadas a partir de las etiquetas de los objetos reconocidos en una imagen dada*: La técnica comprende la descripción de las etiquetas de los objetos, seguida de la vectorización mediante word2vec. En el Capítulo 5, se presentó el módulo *objects-to-scene* para reconocer escenas a partir de las etiquetas de objetos derivadas del módulo *object-centric*. Para ello, ideamos una técnica para describir las imágenes utilizando las etiquetas de los objetos, en la que las descripciones son los atributos propios de las etiquetas de las escenas. A continuación, convertimos estas descripciones en vectores de dimensión 390 utilizando word2vec y luego los introdujimos en una red neuronal para la predicción de escenas.

- *Presentamos una función de peso denominada combinación ponderada de los módulos object-centric y scene-centric (WCOS, del inglés weighted combination of object and scene) que combina la información del objeto y la escena para una predicción global de la escena*: Para superar las limitaciones individuales de utilizar sólo las características a nivel de objeto o las globales, en el Capítulo 5, propusimos un método para combinar la predicción obtenida de los módulos *objects-to-scene* y *scene-centric* utilizando una función de peso para mejorar la precisión de la predicción de la escena. La razón para utilizar la función de peso es que, por un lado, las características a nivel de objeto por sí solas pueden no producir los resultados deseados cuando las escenas entre clases tienen objetos comunes. Y, por otro lado, las características globales son más genéricas debido a la presencia de escenas similares en diferentes clases de imágenes, lo que puede dar lugar a predicciones incorrectas.

- *Introducimos un modelo de recuperación para la recuperación de escenas basada en consultas de imágenes de interior*: Para ello, utilizamos la arquitectura DSP para extraer y combinar características de los módulos *object-centric* y *scene-centric* para representar las imágenes de consulta y las imágenes del conjunto de datos. Las características de los objetos corresponden a la activación neuronal de las regiones propuestas generadas por el módulo *object-centric*, mientras que las características globales se generan en la última capa totalmente conectada del módulo *scene-centric*. Además, creamos diccionarios de imágenes en los que las características de la escena y de los objetos, junto con sus etiquetas, se almacenan en una base de datos, que se utiliza para analizar las correspondencias. Además, también desarrollamos un método de expansión de objetos de consulta para la recuperación de escenas que suaviza los problemas relacionados con el cambio de punto de vista de los objetos y las imágenes de la escena.

  - **Demostración experimental.** *Demostramos que la combinación de las características de los objetos y de las características globales produce resultados aceptables tanto en las tareas de reconocimiento como de recuperación de escenas*. El marco propuesto se probó utilizando tres conjuntos de datos diferentes: el marco de reconocimiento de escenas en interiores se testeó utilizando los conjuntos de datos MIT-67 y NYU, y el de recuperación de escenas utilizando el conjunto de datos Hotels-50k. La precisión alcanzada (MIT-67 Indoor = $94,5\%$, NYU-v2 = $74,5\%$ y la precisión top-1 $10,1\%$ sin oclusión y $7,8\%$ con oclusión media en el Hotels-50k) demostró la eficacia del método propuesto, que también supera significativamente los métodos del estado del arte existentes.

7.3. Trabajos futuros

En esta sección, presentamos las principales líneas de investigación que quedan abiertas y que podría ser interesante abordar en el futuro. Las siguientes nuevas líneas de investigación pueden añadirse como extensiones del trabajo actual propuesto en la tesis.

- *Optimización de la arquitectura del Capítulo 3 para una extracción de características más rápida*: La arquitectura presentada para la creación del descriptor de color es compleja debido a la presencia de tres etapas en el proceso de extracción de características, una para cada canal de color. Debido a ello, tiene una complejidad computacional, en términos de tiempo y memoria, relativamente mayor que una arquitectura CNN general. Para solucionar este problema, se podría simplificar la arquitectura ideando estrategias para reducir el número de capas y etapas.

- *Exploración de diferentes modelos de color*: Para aumentar el poder discriminativo de los descriptores, se pueden explorar otros modelos de color y adaptarlos para crear nuevos descriptores. Además, se pueden formular nuevas ecuaciones adaptadas a diferentes espacios de color para generar descriptores más robustos.

- *Mejora del descriptor DFTD*: Como parte de la investigación futura, se puede modificar el descriptor DFTD propuesto para mejorar la invariabilidad a la rotación teniendo en cuenta los coeficientes de Fourier. Además, también se pueden explorar diversas arquitecturas de autocodificadores que posiblemente permitirían mejorar el descriptor DFTD, como son los transformadores (Tan et al., 2021; El-Nouby et al., 2021).

- *Empleo de hashing profundo para una recuperación más rápida de instancias*: El *hashing* profundo es una forma efectiva de procesar datos de alta dimensión debido a la mayor velocidad de consulta y al bajo coste de memoria. En la actualidad, los enfoques CBIR presentados en la tesis almacenan los datos utilizando el formato HDF5, que permite una recuperación más rápida en bases de datos con millones de imágenes. Sin embargo, con el fin de escalar y acelerar el proceso de recuperación para una base de datos con miles de millones de imágenes, se puede emplear *hashing* profundo (Lu et al., 2019; Chen et al., 2020; Lu et al., 2020).

- *Exploración de nuevas formas de describir las imágenes en el enfoque object-centric para el reconocimiento de escenas*: El enfoque presentado en el Capítulo 5 utiliza descripciones generadas por un detector de objetos para mejorar la predicción global de la escena. El rendimiento depende de la calidad de las descripciones generadas y de los *word embeddings* (en español, vectores de palabras), que se dan como entrada a una red neuronal. Para mejorar aún más el rendimiento, se puede trabajar en nuevos enfoques para enriquecer los *word embeddigns*, y también en explorar diferentes técnicas de descripción.

- *Extensión del actual marco de reconocimiento de escenas de interior para predecir escenas de exterior*: El método de reconocimiento de escenas presentado se limita actualmente a imágenes de escenas interiores. Sin embargo, el mismo enfoque se puede generalizar para el reconocimiento de escenas en exteriores entrenando la arquitectura propuesta con imágenes de escenas en exteriores y objetos de exteriores. Un posible trabajo futuro podría ser abordar el reconocimiento de escenas exteriores entrenando y probando con imágenes de escenas exteriores.

- *Explorar otras redes de detección de objetos*: La detección de objetos es el componente clave tanto para la recuperación de instancias como para el reconocimiento de escenas. Por lo tanto, el rendimiento depende en gran medida de la precisión y la velocidad de la detección de objetos. Por lo tanto, para futuras implementaciones, se puede explorar otros modelos de detección de objetos para incorporarlos a las arquitecturas propuestas de recuperación de instancias y reconocimiento de escenas.

Bibliography

Abdelmounaime, S. and Dong-Chen, H.: 2013, New brodatz-based image databases for grayscale color and multiband texture analysis, *ISRN Machine Vision* **2013**.

Ahmed, K. T., Ummesafi, S. and Iqbal, A.: 2019, Content based image retrieval using image features information fusion, *Information Fusion* **51**, 76–99.

Ahmed, R., Ahmed, M., Jabbar, S., Khalid, S., Ahmad, A., Din, S. and Jeon, G.: 2018, Content based image retrieval by using color descriptor and discrete wavelet transform, *Journal of Medical Systems* pp. 42–44.

Alahi, A., Vandergheynst, P., Bierlaire, M. and Kunt, M.: 2010, Cascade of descriptors to detect and track objects across any network of cameras, *Computer Vision and Image Understanding* **114**(6), 624–640.

Alzu'bi, A., Amira, A., Ramzan, N. and Jaber, T.: 2015, Robust fusion of color and local descriptors for image retrieval and classification, *Systems, Signals and Image Processing (IWSSIP), 2015 International Conference on*, IEEE, pp. 253–256.

Alzu'bi, A., Amira, A. and Ramzan, N.: 2015, Semantic content-based image retrieval: A comprehensive study, *Journal of Visual Communication and Image Representation* **32**, 20–54.

Arandjelovic, R. and Zisserman, A.: 2012, Three things everyone should know to improve object retrieval, *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 2911–2918.

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

Babenko, A., Slesarev, A., Chigorin, A. and Lempitsky, V.: 2014, Neural codes for image retrieval, *European Conference on Computer Vision*, Springer, pp. 584–599.

Bai, S., Bai, X., Zhou, Z., Zhang, Z. and Jan Latecki, L.: 2016, Gift: A real-time and scalable 3d shape search engine, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5023–5032.

Basu, A., Petropoulakis, L., Di Caterina, G. and Soraghan, J.: 2020, Indoor home scene recognition using capsule neural networks, *Procedia Computer Science* **167**, 440–448.

Bay, H., Tuytelaars, T. and Van Gool, L.: 2006, SURF: Speeded up robust features, *European Conference on Computer Vision*, Springer, pp. 404–417.

Bochkovskiy, A., Wang, C.-Y. and Liao, H.-Y. M.: 2020, YOLOv4: Optimal speed and accuracy of object detection, *arXiv preprint arXiv:2004.10934* .

Bouachir, W., Kardouchi, M. and Belacel, N.: 2009, Improving bag of visual words image retrieval: A fuzzy weighting scheme for efficient indexation, *2009 Fifth International Conference on Signal Image Technology and Internet Based Systems*, IEEE, pp. 215–220.

Burghouts, G. J. and Geusebroek, J.-M.: 2009, Performance evaluation of local colour invariants, *Computer Vision and Image Understanding* **113**(1), 48–62.

Cadieu, C. F., Hong, H., Yamins, D. L., Pinto, N., Ardila, D., Solomon, E. A., Majaj, N. J. and DiCarlo, J. J.: 2014, Deep neural networks rival the representation of primate it cortex for core visual object recognition, *PLoS Computational Biology* **10**(12), e1003963.

Cai, Y., Huang, K. and Tan, T.: 2008, Matching tracking sequences across widely separated cameras, *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, IEEE, pp. 765–768.

Cai, Y., Li, Y., Qiu, C., Ma, J. and Gao, X.: 2019, Medical image retrieval based on convolutional neural network and supervised hashing, *IEEE Access* **7**, 51877–51885.

Cao, L. and Fei-Fei, L.: 2007, Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes, *2007 IEEE 11th International Conference on Computer Vision*, IEEE, pp. 1–8.

Casanova, D., Florindo, J. B., Falvo, M. and Bruno, O. M.: 2016, Texture analysis using fractal descriptors estimated by the mutual interference of color channels, *Information Sciences* **346**, 58–72.

Chen, X., Yang, X., Zhang, R., Liu, A. and Zheng, S.: 2010, Edge region color autocorrelogram: A new low-level feature applied in CBIR, *2010 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB)*, IEEE, pp. 1–4.

Chen, Y., Lu, X. and Li, X.: 2020, Supervised deep hashing with a joint deep network, *Pattern Recognition* **105**, 107368.

Cheng, X., Lu, J., Feng, J., Yuan, B. and Zhou, J.: 2018, Scene recognition with objectness, *Pattern Recognition* **74**, 474–487.

Chigateri, M. K. and Sonoli, S.: 2021, CBIR algorithm development using RGB histogram-based block contour method to improve the retrieval performance, *Materials Today: Proceedings* .

Chun, Y. D., Kim, N. C. and Jang, I. H.: 2008, Content-based image retrieval using multiresolution color and texture features, *IEEE Transactions on Multimedia* **10**(6), 1073–1084.

Chun, Y. D., Seo, S. Y. and Kim, N. C.: 2003, Image retrieval using BDIP and BVLC moments, *IEEE Transactions on Circuits and Systems for Video Technology* **13**(9), 951–957.

Cichy, R. M., Khosla, A., Pantazis, D. and Oliva, A.: 2017, Dynamics of scene representations in the human brain revealed by magnetoencephalography and deep neural networks, *NeuroImage* **153**, 346–358.

Cimpoi, M., Maji, S. and Vedaldi, A.: 2015, Deep filter banks for texture recognition and segmentation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3828–3836.

Cong, D.-N. T., Khoudour, L., Achard, C., Meurie, C. and Lezoray, O.: 2010, People reidentification by spectral classification of silhouettes, *Signal Processing* **90**(8), 2362–2374.

Cortes, D., Calderón, G., Arista, A., Toscano, K. and Nakano, M.: 2016, Aerial image classification using texture and color-based descriptors, *Geoespaciales (CNCG), 2016 IEEE 1er Congreso Nacional de Ciencias*, IEEE, pp. 1–4.

Csurka, G., Dance, C., Fan, L., Willamowski, J. and Bray, C.: 2004, Visual categorization with bags of keypoints, *Workshop on Statistical Learning in Computer Vision, ECCV*, Prague, pp. 1–2.

Dai, J., Li, Y., He, K. and Sun, J.: 2016, R-fcn: Object detection via region-based fully convolutional networks, *Advances in Neural Information Processing Systems*, pp. 379–387.

Dalal, N. and Triggs, B.: 2005, Histograms of oriented gradients for human detection, *2005 IEEE computer society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, Vol. 1, IEEE, pp. 886–893.

D'Amato, J. P., Mercado, M., Heiling, A. and Cifuentes, V.: 2016, A proximal optimization method to the problem of nesting irregular pieces using parallel architectures, *REVISTA IBEROAMERICANA DE AUTOMATICA E INFORMATICA INDUSTRIAL* **13**(2), 220–227.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K. and Fei-Fei, L.: 2009, Imagenet: A large-scale hierarchical image database, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 248–255.

Ding, Y., Wong, W. K., Lai, Z. and Zhang, Z.: 2020, Discriminative dual-stream deep hashing for large-scale image retrieval, *Information Processing & Management* **57**(6), 102288.

Dixit, M., Chen, S., Gao, D., Rasiwasia, N. and Vasconcelos, N.: 2015, Scene classification with semantic fisher vectors, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2974–2983.

Du, X., Lin, T.-Y., Jin, P., Ghiasi, G., Tan, M., Cui, Y., Le, Q. V. and Song, X.: 2020, Spinenet: Learning scale-permuted backbone for recognition and localization, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11592–11601.

Dubey, S. R., Roy, S. K., Chakraborty, S., Mukherjee, S. and Chaudhuri, B. B.: 2019, Local bit-plane decoded convolutional neural network features for biomedical image retrieval, *Neural Computing and Applications* pp. 1–13.

Duygulu, P., Barnard, K., de Freitas, J. F. and Forsyth, D. A.: 2002, Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary, *European Conference on Computer Vision*, Springer, pp. 97–112.

El-Nouby, A., Neverova, N., Laptev, I. and Jégou, H.: 2021, Training vision transformers for image retrieval, *arXiv preprint arXiv:2102.05644* .

Erin Liong, V., Lu, J., Wang, G., Moulin, P. and Zhou, J.: 2015, Deep hashing for compact binary codes learning, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2475–2483.

Erol, B., Dumitraş, A., Kossentini, F., Joch, A. and Sullivan, G.: 2005, 6.5 - MPEG-4, H.264/AVC, and MPEG-7: New standards for the digital video industry, *in* A. BOVIK (ed.), *Handbook of Image and Video Processing (Second Edition)*, second edition edn, Communications, Networking and Multimedia, Academic Press, Burlington, pp. 849–XXIV.

Fan, Q., Zhuo, W., Tang, C.-K. and Tai, Y.-W.: 2020, Few-shot object detection with attention-rpn and multi-relation detector, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4013–4022.

Gao, S., Tsang, I. W.-H., Chia, L.-T. and Zhao, P.: 2010, Local features are not lonely–laplacian sparse coding for image classification, *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 3555–3561.

García-Olalla, O., Alegre, E., Fernández-Robles, L., Fidalgo, E. and Saikia, S.: 2018, Textile retrieval based on image content from cdc and webcam cameras in indoor environments, *Sensors* **18**(5), 1329.

Garg, M. and Dhiman, G.: 2021, A novel content-based image retrieval approach for classification using GLCM features and texture fused LBP variants, *Neural Computing and Applications* **33**, 1311–1328.

Girshick, R.: 2015, Fast R-CNN, *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1440–1448.

Girshick, R., Donahue, J., Darrell, T. and Malik, J.: 2014, Rich feature hierarchies for accurate object detection and semantic segmentation, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587.

Gkelios, S., Sophokleous, A., Plakias, S., Boutalis, Y. and Chatzichristofis, S. A.: 2021, Deep convolutional features for image retrieval, *Expert Systems with Applications* **177**, 114940.

Goh, H., Thome, N., Cord, M. and Lim, J.-H.: 2014, Learning deep hierarchical visual feature coding, *IEEE transactions on Neural Networks and Learning Systems* **25**(12), 2212–2225.

Gordo, A., Almazán, J., Revaud, J. and Larlus, D.: 2016, Deep image retrieval: Learning global representations for image search, *European Conference on Computer Vision*, Springer, pp. 241–257.

Gordo, A., Almazan, J., Revaud, J. and Larlus, D.: 2017, End-to-end learning of deep visual representations for image retrieval, *International Journal of Computer Vision* **124**(2), 237–254.

Gray, D. and Tao, H.: 2008, Viewpoint invariant pedestrian recognition with an ensemble of localized features, *European Conference on Computer Vision*, Springer, pp. 262–275.

Guo, J.-M., Prasetyo, H. and Wang, N.-J.: 2015, Effective image retrieval system using dot-diffused block truncation coding features, *IEEE Transactions on Multimedia* **17**(9), 1576–1590.

Guo, S., Huang, W., Wang, L. and Qiao, Y.: 2016, Locally supervised deep hybrid model for scene recognition, *IEEE Transactions on Image Processing* **26**(2), 808–820.

Gupta, S., Arbelaez, P. and Malik, J.: 2013, Perceptual organization and recognition of indoor scenes from RGB-D images, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 564–571.

Hamouchene, I. and Aouat, S.: 2014, Texture matching using local and global descriptor, *2014 5th European Workshop on Visual Information Processing (EUVIP)*, IEEE, pp. 1–5.

Han, H., Li, J., Jain, A. K., s. shan and Chen, X.: 2019, Tattoo image search at scale: Joint detection and compact representation learning, *IEEE Transactions on Pattern Analysis and Machine Intelligence* pp. 1–1.

Hayat, M., Khan, S. H., Bennamoun, M. and An, S.: 2016, A spatial layout and scale invariant feature representation for indoor scene classification, *IEEE Transactions on Image Processing* **25**(10), 4829–4841.

He, K., Gkioxari, G., Dollár, P. and Girshick, R.: 2017, Mask R-CNN, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2961–2969.

He, K., Zhang, X., Ren, S. and Sun, J.: 2015, Spatial pyramid pooling in deep convolutional networks for visual recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **37**(9), 1904–1916.

Heikkilä, M., Pietikäinen, M. and Schmid, C.: 2006, Description of interest regions with center-symmetric local binary patterns, *Computer Vision, Graphics and Image Processing*, Springer, pp. 58–69.

Huang, J., Kumar, S. R., Mitra, M., Zhu, W.-J. and Zabih, R.: 1997, Image indexing using color correlograms, *Proceedings of IEEE computer society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 762–768.

Iscen, A., Tolias, G., Avrithis, Y., Furon, T. and Chum, O.: 2017, Efficient diffusion on region manifolds: Recovering small objects with compact CNN representations, *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, pp. 926–935.

Jegou, H., Douze, M. and Schmid, C.: 2008, Hamming embedding and weak geometric consistency for large scale image search, *European Conference on Computer Vision*, Springer, pp. 304–317.

Jégou, H., Douze, M. and Schmid, C.: 2010a, Improving bag-of-features for large scale image search, *International Journal of Computer Vision* **87**(3), 316–336.

Jegou, H., Douze, M. and Schmid, C.: 2010b, Product quantization for nearest neighbor search, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(1), 117–128.

Jégou, H., Douze, M., Schmid, C. and Pérez, P.: 2010, Aggregating local descriptors into a compact image representation, *2010 IEEE computer society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 3304–3311.

Jegou, H., Perronnin, F., Douze, M., Sánchez, J., Perez, P. and Schmid, C.: 2011, Aggregating local image descriptors into compact codes, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **34**(9), 1704–1716.

Jégou, H. and Zisserman, A.: 2014, Triangulation embedding and democratic aggregation for image search, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3310–3317.

Jiang, S., Chen, G., Song, X. and Liu, L.: 2019, Deep patch representations with shared codebook for scene classification, *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* **15**(1s), 1–17.

Jiang, Y., Yuan, J. and Yu, G.: 2012, Randomized spatial partition for scene recognition, *European Conference on Computer Vision*, Springer, pp. 730–743.

Jiménez, A., Álvarez, J. M. and i Nieto, X. G.: 2017, Class weighted convolutional features for visual instance search, *28th British Machine Vision Conference (BMVC)*, London, UK.

Juneja, M., Vedaldi, A., Jawahar, C. and Zisserman, A.: 2013, Blocks that shout: Distinctive parts for scene classification, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 923–930.

Kalantidis, Y., Mellina, C. and Osindero, S.: 2016, Cross-dimensional weighting for aggregated deep convolutional features, *European Conference on Computer Vision*, Springer, pp. 685–701.

Khan, R., Van de Weijer, J., Khan, F. S., Muselet, D., Ducottet, C. and Barat, C.: 2013, Discriminative color descriptors, *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, IEEE, pp. 2866–2873.

Khan, S. H., Hayat, M., Bennamoun, M., Togneri, R. and Sohel, F. A.: 2016, A discriminative representation of convolutional features for indoor scene recognition, *IEEE Transactions on Image Processing* **25**(7), 3372–3383.

King, I. and Lau, T. K.: 1996, A feature-based image retrieval database for the fashion, textile, and clothing industry in hong kong, *Proc. of International Symposium Multi-Technology Information Processing*, Vol. 96, pp. 233–240.

Krizhevsky, A., Sutskever, I. and Hinton, G. E.: 2012, Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems*, pp. 1097–1105.

Kwitt, R. and Meerwald, P.: 2018, Salzburg texture image database (stex).

Kwitt, R. and Uhl, A.: 2008, Image similarity measurement by kullback-leibler divergences between complex wavelet subband statistics for texture retrieval, *2008 15th IEEE International Conference on Image Processing*, IEEE, pp. 933–936.

Lai, H., Pan, Y., Liu, Y. and Yan, S.: 2015, Simultaneous feature learning and hash coding with deep neural networks, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3270–3278.

Lantagne, M., Parizeau, M. and Bergevin, R.: 2003, Vip: Vision tool for comparing images of people, *Proceedings of the 16th IEEE Conf. on Vision Interface*, pp. 35–42.

Lazebnik, S., Schmid, C. and Ponce, J.: 2006, Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2, IEEE, pp. 2169–2178.

Lazebnik, S., Schmid, C., Ponce, J. et al.: 2009, Spatial pyramid matching, *Object Categorization: Computer and Human Vision Perspectives* **3**(4).

Lei, J., Luo, X., Fang, L., Wang, M. and Gu, Y.: 2020, Region-enhanced convolutional neural network for object detection in remote sensing images, *IEEE Transactions on Geoscience and Remote Sensing* pp. 1–10.

Li, H., Huang, Y. and Zhang, Z.: 2017, An improved Faster R-CNN for same object retrieval, *IEEE Access* **5**, 13665–13676.

Li, L.-J., Su, H., Fei-Fei, L. and Xing, E. P.: 2010, Object bank: A high-level image representation for scene classification & semantic feature sparsification, *Advances in Neural Information Processing Systems*, pp. 1378–1386.

Li, X., Yang, J. and Ma, J.: 2021, Recent developments of content-based image retrieval CBIR, *Neurocomputing* .

Li, Y., Zhang, Z., Cheng, Y., Wang, L. and Tan, T.: 2019, MAPNet: Multi-modal attentive pooling network for RGB-D indoor scene classification, *Pattern Recognition* **90**, 436–449.

Lin, D., Lu, C., Liao, R. and Jia, J.: 2014, Learning important spatial pooling regions for scene classification, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3726–3733.

Lin, J., Zhan, Y. and Zhao, W.-L.: 2019, Instance search based on weakly supervised feature learning, *Neurocomputing* .

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B. and Belongie, S.: 2017, Feature pyramid networks for object detection, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125.

Lin, T.-Y., Goyal, P., Girshick, R., He, K. and Dollár, P.: 2017, Focal loss for dense object detection, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2980–2988.

Liu, G.-H. and Yang, J.-Y.: 2013, Content-based image retrieval using color difference histogram, *Pattern Recognition* **46**(1), 188–198.

Liu, G.-H., Zhang, L., Hou, Y.-K., Li, Z.-Y. and Yang, J.-Y.: 2010, Image retrieval based on multi-texton histogram, *Pattern Recognition* **43**(7), 2380–2389.

Liu, S., Wu, J., Feng, L., Qiao, H., Liu, Y., Luo, W. and Wang, W.: 2018, Perceptual uniform descriptor and ranking on manifold for image retrieval, *Information Sciences* **424**, 235–249.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y. and Berg, A. C.: 2016, Ssd: Single shot multibox detector, *European Conference on Computer Vision*, Springer, pp. 21–37.

Liu, W. and Wu, C. Y.: 2019, Crime scene investigation image retrieval using a hierarchical approach and rank fusion, *2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, IEEE, pp. 1974–1978.

Liu, Y., Chen, Q., Chen, W. and Wassell, I.: 2018, Dictionary learning inspired deep network for scene recognition, *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

Liu, Y., Hu, D., Fan, J., Wang, F. and Zhang, D.: 2017, Multi-feature fusion for crime scene investigation image retrieval, *2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, IEEE, pp. 1–7.

López-Cifuentes, A., Escudero-Viñolo, M., Bescós, J. and García-Martín, Á.: 2020, Semantic-aware scene recognition, *Pattern Recognition* **102**, 107256.

Lowe, D. G.: 2004a, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision* **60**(2), 91–110.

Lowe, D. G.: 2004b, Distinctive image features from scale-invariant keypoints, *International Journal of Computer Vision* **60**(2), 91–110.

Lu, H., Zhang, M., Xu, X., Li, Y. and Shen, H. T.: 2020, Deep fuzzy hashing network for efficient image retrieval, *IEEE Transactions on Fuzzy Systems* .

Lu, X., Chen, Y. and Li, X.: 2019, Discrete deep hashing with ranking optimization for image retrieval, *IEEE Transactions on Neural Networks and Learning Systems* .

Ma, A., Wan, Y., Zhong, Y., Wang, J. and Zhang, L.: 2021, Scenenet: Remote sensing scene classification deep learning network using multi-objective neural evolution architecture search, *ISPRS Journal of Photogrammetry and Remote Sensing* **172**, 171–188.

Maji, S. and Bose, S.: 2020, CBIR using features derived by deep learning, *arXiv preprint arXiv:2002.07877* .

Meng, X., Wang, Z. and Wu, L.: 2012, Building global image features for scene recognition, *Pattern Recognition* **45**(1), 373–380.

Mikolov, T., Chen, K., Corrado, G. and Dean, J.: 2013, Efficient estimation of word representations in vector space, *arXiv preprint arXiv:1301.3781* .

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

Mohedano, E., McGuinness, K., Giró-i Nieto, X. and O'Connor, N. E.: 2018, Saliency weighted convolutional features for instance search, *2018 International Conference on Content-Based Multimedia Indexing (CBMI)*, IEEE, pp. 1–6.

Muja, M. and Lowe, D. G.: 2014, Scalable nearest neighbor algorithms for high dimensional data, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **36**(11), 2227–2240.

Mukherjee, A., Chakraborty, S., Sil, J. and Chowdhury, A. S.: 2017, A novel visual word assignment model for content-based image retrieval, *Proceedings of International Conference on Computer Vision and Image Processing*, Springer, pp. 79–87.

Mukherjee, A., Sil, J., Sahu, A. and Chowdhury, A. S.: 2020, A bag of constrained informative deep visual words for image retrieval, *Pattern Recognition Letters* **129**, 158–165.

Nanni, L., Brahnam, S. and Lumini, A.: 2010, A local approach based on a local binary patterns variant texture descriptor for classifying pain states, *Expert Systems with Applications* **37**(12), 7888–7894.

Nanni, L., Ghidoni, S. and Brahnam, S.: 2017, Handcrafted vs. non-handcrafted features for computer vision classification, *Pattern Recognition* **71**, 158–172.

Napoletano, P.: 2017, Hand-crafted vs learned descriptors for color texture classification, *International Workshop on Computational Color Imaging*, Springer, pp. 259–271.

Nene, S. A., Nayar, S. K., Murase, H. et al.: 1996, Columbia object image library (coil-20), *Technical Report CUCS-005-96*, Department of Computer Science. Columbia University.

Newsam, S. and Yang, Y.: 2007, Comparing global and interest point descriptors for similarity retrieval in remote sensed imagery, *Proceedings of the 15th annual ACM International Symposium on Advances in Geographic Information Systems*, pp. 1–8.

Nister, D. and Stewenius, H.: 2006, Scalable recognition with a vocabulary tree, *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, Vol. 2, IEEE, pp. 2161–2168.

Niu, Z., Hua, G., Gao, X. and Tian, Q.: 2012, Context aware topic model for scene recognition, *2012 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 2743–2750.

Noh, H., Araujo, A., Sim, J., Weyand, T. and Han, B.: 2017, Large-scale image retrieval with attentive deep local features, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3456–3465.

Nowak, E., Jurie, F. and Triggs, B.: 2006, Sampling strategies for bag-of-features image classification, *European Conference on Computer Vision*, Springer, pp. 490–503.

Ojala, T., Pietikäinen, M. and Harwood, D.: 1996, A comparative study of texture measures with classification based on featured distributions, *Pattern Recognition* **29**(1), 51–59.

Ojala, T., Pietikainen, M. and Maenpaa, T.: 2002a, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(7), 971–987.

Ojala, T., Pietikainen, M. and Maenpaa, T.: 2002b, Multiresolution gray-scale and rotation invariant texture classification with local binary patterns, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **24**(7), 971–987.

Oliva, A.: 2005, Gist of the scene, *Neurobiology of Attention*, Elsevier, pp. 251–256.

Ouslimani, F., Ouslimani, A. and Ameur, Z.: 2019, Rotation-invariant features based on directional coding for texture classification, *Neural Computing and Applications* **31**(10), 6393–6400.

Pandey, M. and Lazebnik, S.: 2011, Scene recognition and weakly supervised object localization with deformable part-based models, *2011 International Conference on Computer Vision*, IEEE, pp. 1307–1314.

Payne, A. and Singh, S.: 2005, Indoor vs. outdoor scene classification in digital photographs, *Pattern Recognition* **38**(10), 1533–1545.

Pham, M.-T.: 2018, Efficient texture retrieval using multiscale local extrema descriptors and covariance embedding, *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 0–0.

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

Pradhan, J., Pal, A. K., Banka, H. and Dansena, P.: 2021, Fusion of region based extracted features for instance-and class-based CBIR applications, *Applied Soft Computing* **102**, 107063.

Prosser, B., Zheng, W.-S., Gong, S. and Xiang, T.: 2010, Person re-identification by support vector ranking, *Proceedings of the British Machine Vision Conference*, BMVA Press, pp. 21.1–21.11. doi:10.5244/C.24.21.

Pujari, J., Pushpalatha, S. and Padmashree, D.: 2010, Content-based image retrieval using color and shape descriptors, *Signal and Image Processing (ICSIP), 2010 International Conference on*, IEEE, pp. 239–242.

Qin, J. and Yung, N. H.: 2010, Scene categorization via contextual visual words, *Pattern Recognition* **43**(5), 1874–1888.

Quattoni, A. and Torralba, A.: 2009, Recognizing indoor scenes, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 413–420.

Radenović, F., Iscen, A., Tolias, G., Avrithis, Y. and Chum, O.: 2018, Revisiting Oxford and Paris: Large-scale image retrieval benchmarking, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5706–5715.

Radenović, F., Tolias, G. and Chum, O.: 2016, CNN image retrieval learns from bow: Unsupervised fine-tuning with hard examples, *European Conference on Computer Vision*, Springer, pp. 3–20.

Radenović, F., Tolias, G. and Chum, O.: 2018, Fine-tuning CNN image retrieval with no human annotation, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **41**(7), 1655–1668.

Rahimzadeh, M., Parvin, S., Safi, E. and Mohammadi, M. R.: 2021, Wise-SrNet: A novel architecture for enhancing image classification by learning spatial resolution of feature maps, *arXiv preprint arXiv:2104.12294* .

Ran, T., Yuan, L. and Zhang, J.: 2021, Scene perception based visual navigation of mobile robot in indoor environment, *ISA Transactions* **109**, 389–400.

Razavian, A. S., Sullivan, J., Carlsson, S. and Maki, A.: 2016, Visual instance retrieval with deep convolutional networks, *ITE Transactions on Media Technology and Applications* **4**(3), 251–258.

Redmon, J., Divvala, S., Girshick, R. and Farhadi, A.: 2016, You only look once: Unified, real-time object detection, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788.

Redmon, J. and Farhadi, A.: 2017, Yolo9000: better, faster, stronger, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271.

Redmon, J. and Farhadi, A.: 2018, Yolov3: An incremental improvement, *arXiv preprint arXiv:1804.02767* .

Ren, S., He, K., Girshick, R. and Sun, J.: 2017, Faster R-CNN: Towards real-time object detection with region proposal networks, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **39**(6), 1137–1149.

Saikia, S., Fernández-Robles, L., Fernández, E. F. and Alegre, E.: 2021, Colour neural descriptors for instance retrieval using CNN features and colour models, *IEEE Access* **9**, 23218–23234.

Salvador, A., Giró-i Nieto, X., Marqués, F. and Satoh, S.: 2016, Faster R-CNN features for instance search, *Computer Vision and Pattern Recognition Workshops (CVPRW), 2016 IEEE Conference on*, IEEE, pp. 394–401.

Seong, H., Hyun, J. and Kim, E.: 2020, Fosnet: an end-to-end trainable deep neural network for scene recognition, *IEEE Access* **8**, 82066–82077.

Sermanet, P., Eigen, D., Zhang, X., Mathieu, M., Fergus, R. and LeCun, Y.: 2013, Overfeat: Integrated recognition, localization and detection using convolutional networks, *arXiv preprint arXiv:1312.6229* .

Shakarami, A. and Tarrah, H.: 2020, An efficient image descriptor for image classification and CBIR, *Optik* **214**, 164833.

Shao, H., Wu, Y., Cui, W. and Zhang, J.: 2008, Image retrieval based on MPEG-7 dominant color descriptor, *2008 The 9th International Conference for Young Computer Scientists*, IEEE, pp. 753–757.

Shao, Z., Zhou, W., Deng, X., Zhang, M. and Cheng, Q.: 2020, Multilabel remote sensing image retrieval based on fully convolutional network, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **13**, 318–328.

Shao, Z., Zhou, W., Zhang, L. and Hou, J.: 2014, Improved color texture descriptors for remote sensing image retrieval, *Journal of applied remote sensing* **8**(1), 083584.

Sharif Razavian, A., Azizpour, H., Sullivan, J. and Carlsson, S.: 2014, CNN features off-the-shelf: an astounding baseline for recognition, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition workshops*, pp. 806–813.

Shinya, Y.: 2021, USB: Universal-scale object detection benchmark, *arXiv preprint arXiv:2103.14027* .

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

Simonyan, K. and Zisserman, A.: 2014, Very deep convolutional networks for large-scale image recognition, *arXiv preprint arXiv:1409.1556* .

Singh, C., Walia, E. and Kaur, K. P.: 2018, Color texture description with novel local binary patterns for effective image retrieval, *Pattern Recognition* **76**, 50–68.

Singh, S., Gupta, A. and Efros, A. A.: 2012, Unsupervised discovery of mid-level discriminative patches, *European Conference on Computer Vision*, Springer, pp. 73–86.

Sokic, E. and Konjicija, S.: 2016, Phase preserving fourier descriptor for shape-based image retrieval, *Signal Processing: Image Communication* **40**, 82–96.

Song, X., Chen, C. and Jiang, S.: 2017, RGB-D scene recognition with object-to-object relation, *Proceedings of the 25th ACM International Conference on Multimedia*, pp. 600–608.

Song, X., Herranz, L. and Jiang, S.: 2017, Depth CNNs for RGB-D scene recognition: Learning from scratch better than transferring from RGB-CNNs, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, AAAI Press, p. 4271–4277.

Song, X., Jiang, S. and Herranz, L.: 2017, Multi-scale multi-feature context modeling for scene recognition in the semantic manifold, *IEEE Transactions on Image Processing* **26**(6), 2721–2735.

Song, X., Jiang, S., Herranz, L., Kong, Y. and Zheng, K.: 2016, Category co-occurrence modeling for large scale scene recognition, *Pattern Recognition* **59**, 98–111.

Sotoodeh, M., a Moosavi, M. R. and Boostani, R.: 2019, A novel adaptive LBP-based descriptor for color image retrieval, *Expert Systems with Applications* **127**, 342 – 352.

Sotoodeh, M., Moosavi, M. R. and Boostani, R.: 2019, A novel adaptive LBP-based descriptor for color image retrieval, *Expert Systems with Applications* **127**, 342–352.

Stanković, R. S. and Falkowski, B. J.: 2003, The Haar wavelet transform: its status and achievements, *Computers & Electrical Engineering* **29**(1), 25–44.

Stylianou, A., Xuan, H., Shende, M., Brandt, J., Souvenir, R. and Pless, R.: 2019, Hotels-50k: A global hotel recognition dataset, *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, pp. 726–733.

Sukthankar, Rahul, Y. K.: 2004, A more distinctive representation for local image descriptors, *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattem Recognition. USA: 1EEE*, pp. 1063–6919.

Swain, M. J. and Ballard, D. H.: 1991, Color indexing, *International Journal of Computer Vision* **7**(1), 11–32.

Tan, F., Yuan, J. and Ordonez, V.: 2021, Instance-level image retrieval using reranking transformers, *arXiv preprint arXiv:2103.12236* .

Tan, M. and Le, Q.: 2019, Efficientnet: Rethinking model scaling for convolutional neural networks, *International Conference on Machine Learning*, PMLR, pp. 6105–6114.

Tan, M., Pang, R. and Le, Q. V.: 2020, Efficientdet: Scalable and efficient object detection, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10781–10790.

Tan, X. and Triggs, B.: 2010, Enhanced local texture feature sets for face recognition under difficult lighting conditions, *IEEE Transactions on Image Processing* **19**(6), 1635–1650.

Tang, P., Wang, H. and Kwong, S.: 2017, G-MS2F: Googlenet based multi-stage feature fusion of deep CNN for scene recognition, *Neurocomputing* **225**, 188–197.

Tao, D.: 2009, The corel database for content based image retrieval.

Teichmann, M., Araujo, A., Zhu, M. and Sim, J.: 2019, Detect-to-retrieve: Efficient regional aggregation for image search, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5109–5118.

Tian, Z., Shen, C., Chen, H. and He, T.: 2019, FCOS: Fully convolutional one-stage object detection, *arXiv preprint arXiv:1904.01355* .

Tolias, G., Avrithis, Y. and Jégou, H.: 2016, Image search with selective match kernels: aggregation across single and multiple images, *International Journal of Computer Vision* **116**(3), 247–261.

Tuncer, T., Dogan, S. and Ataman, V.: 2019, A novel and accurate chess pattern for automated texture classification, *Physica A: Statistical Mechanics and its Applications* **536**, 122584.

Tuncer, T., Dogan, S. and Ertam, F.: 2019, A novel neural network based image descriptor for texture classification, *Physica A: Statistical Mechanics and its Applications* **526**, 120955.

Van De Sande, K., Gevers, T. and Snoek, C.: 2010, Evaluating color descriptors for object and scene recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(9), 1582–1596.

Van Gemert, J. C., Geusebroek, J.-M., Veenman, C. J. and Smeulders, A. W.: 2008, Kernel codebooks for scene categorization, *European Conference on Computer Vision*, Springer, pp. 696–709.

Van Gemert, J. C., Veenman, C. J., Smeulders, A. W. and Geusebroek, J.-M.: 2009, Visual word ambiguity, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **32**(7), 1271–1283.

Vogel, J. and Schiele, B.: 2006, Performance evaluation and optimization for content-based image retrieval, *Pattern Recognition* **39**(5), 897–909.

Wang, A., Cai, J., Lu, J. and Cham, T.-J.: 2016, Modality and component aware feature fusion for RGB-D scene classification, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5995–6004.

Wang, C.-Y., Bochkovskiy, A. and Liao, H.-Y. M.: 2020, Scaled-yolov4: Scaling cross stage partial network, *arXiv preprint arXiv:2011.08036* .

Wang, J. and Hua, X.-S.: 2011, Interactive image search by color map, *ACM Transactions on Intelligent Systems and Technology (TIST)* **3**(1), 1–23.

Wang, J., Yang, J., Yu, K., Lv, F., Huang, T. and Gong, Y.: 2010, Locality-constrained linear coding for image classification, *2010 IEEE computer society Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 3360–3367.

Wang, Q., Lai, J., Claesen, L., Yang, Z., Lei, L. and Liu, W.: 2020, A novel feature representation: Aggregating convolution kernels for image retrieval, *Neural Networks* **130**, 1–10.

Wang, S. and Jiang, S.: 2015, INSTRE: a new benchmark for instance-level object retrieval and recognition, *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)* **11**(3), 37.

Wang, X.-Y., Zhang, B.-B. and Yang, H.-Y.: 2014, Content-based image retrieval by integrating color and texture features, *Multimedia Tools and Applications* **68**(3), 545–569.

Wang, Y., Huang, F., Zhang, Y., Feng, R., Zhang, T. and Fan, W.: 2020, Deep cascaded cross-modal correlation learning for fine-grained sketch-based image retrieval, *Pattern Recognition* **100**, 107148.

Wang, Z., Wang, L., Wang, Y., Zhang, B. and Qiao, Y.: 2017, Weakly supervised patchnets: Describing and aggregating local patches for scene recognition, *IEEE Transactions on Image Processing* **26**(4), 2028–2041.

Weng, C., Wang, H., Yuan, J. and Jiang, X.: 2016, Discovering class-specific spatial layouts for scene recognition, *IEEE Signal Processing Letters* **24**(8), 1143–1147.

Wong, C.: 2017, *Applications of computer vision in fashion and textiles*, Woodhead Publishing.

Wu, J. and Rehg, J. M.: 2009, Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel, *2009 IEEE 12th International Conference on Computer Vision*, IEEE, pp. 630–637.

Wu, J. and Rehg, J. M.: 2010, Centrist: A visual descriptor for scene categorization, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **33**(8), 1489–1501.

Wu, R., Wang, B., Wang, W. and Yu, Y.: 2015, Harvesting discriminative meta objects with deep CNN features for scene classification, *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1287–1295.

Wu, X., Sahoo, D. and Hoi, S. C.: 2020, Recent advances in deep learning for object detection, *Neurocomputing* .

Xiao, Y., Wu, J. and Yuan, J.: 2013, mcentrist: A multi-channel feature generation mechanism for scene categorization, *IEEE Transactions on Image Processing* **23**(2), 823–836.

Xie, G.-S., Zhang, X.-Y. and Liu, C.-L.: 2014, Efficient feature coding based on auto-encoder network for image classification, *Asian Conference on Computer Vision*, Springer, pp. 628–642.

Xie, G.-S., Zhang, X.-Y., Yan, S. and Liu, C.-L.: 2015, Hybrid CNN and dictionary-based models for scene recognition and domain adaptation, *IEEE Transactions on Circuits and Systems for Video Technology* **27**(6), 1263–1274.

Xie, L., Lee, F., Liu, L., Kotani, K. and Chen, Q.: 2020, Scene recognition: A comprehensive survey, *Pattern Recognition* **102**, 107205.

Yang, C. and Yu, Q.: 2019, Multiscale fourier descriptor based on triangular features for shape retrieval, *Signal Processing: Image Communication* **71**, 110–119.

Yang, J., Liang, J., Shen, H., Wang, K., Rosin, P. L. and Yang, M.: 2018, Dynamic match kernel with deep convolutional features for image retrieval, *IEEE Transactions on Image Processing* **27**(11), 5288–5302.

Yang, J., Yu, K., Gong, Y. and Huang, T.: 2009, Linear spatial pyramid matching using sparse coding for image classification, *2009 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, pp. 1794–1801.

Yang, N.-C., Chang, W.-H., Kuo, C.-M. and Li, T.-H.: 2008, A fast MPEG-7 dominant color extraction with new similarity measure for image retrieval, *Journal of Visual Communication and Image Representation* **19**(2), 92–105.

Yang, Songfan, D. R.: 2015, Multi-scale recognition with DAG-CNNs, *IEEE International Conference on Computer Vision*, IEEE, pp. 1215–1223.

Yoo, D., Park, S., Lee, J.-Y. and Kweon, I. S.: 2014, Fisher kernel for deep neural activations, *arXiv preprint arXiv:1412.1628* .

Yuan, Y., Wan, J. and Wang, Q.: 2016, Congested scene classification via efficient unsupervised feature learning and density estimation, *Pattern Recognition* **56**, 159–169.

Zheng, L., Yang, Y. and Tian, Q.: 2017, SIFT meets CNN: A decade survey of instance retrieval, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40**(5), 1224–1244.

Zhou, B., Khosla, A., Lapedriza, A., Torralba, A. and Oliva, A.: 2016, Places: An image database for deep scene understanding, *arXiv preprint arXiv:1610.02055* .

Zhou, B., Lapedriza, A., Khosla, A., Oliva, A. and Torralba, A.: 2017, Places: A 10 million image database for scene recognition, *IEEE Transactions on Pattern Analysis and Machine Intelligence* **40**(6), 1452–1464.

Zhou, L., Zhou, Z. and Hu, D.: 2013, Scene classification using a multi-resolution bag-of-features model, *Pattern Recognition* **46**(1), 424–433.

Zhou, W., Newsam, S., Li, C. and Shao, Z.: 2017, Learning low dimensional convolutional neural networks for high-resolution remote sensing image retrieval, *Remote Sensing* **9**(5), 489.

Zhou, W., Newsam, S., Li, C. and Shao, Z.: 2018, Patternnet: A benchmark dataset for performance evaluation of remote sensing image retrieval, *ISPRS journal of photogrammetry and remote sensing* **145**, 197–209.

Zhou, W., Shao, Z., Diao, C. and Cheng, Q.: 2015, High-resolution remote-sensing imagery retrieval using sparse features by auto-encoder, *Remote Sensing Letters* **6**(10), 775–783.

Zhu, C., Bichot, C.-E. and Chen, L.: 2013, Image region description using orthogonal combination of local binary patterns enhanced with color information, *Pattern Recognition* **46**(7), 1949–1963.

Zhu, J., Wang, J., Pang, S., Guan, W., Li, Z., Li, Y. and Qian, X.: 2019, Co-weighting semantic convolutional features for object retrieval, *Journal of Visual Communication and Image Representation* **62**, 368–380.

Zuo, Z., Wang, G., Shuai, B., Zhao, L., Yang, Q. and Jiang, X.: 2014, Learning discriminative and shareable features for scene classification, *European Conference on Computer Vision*, Springer, pp. 552–568.

Annex B

SUMMARY OF THE THESIS IN SPANISH

RESUMEN DE LA TESIS EN CASTELLANO

1 Introducción

1.1 Motivación

La búsqueda y recuperación de imágenes, la detección de objetos y el reconocimiento de escenas son tres tareas importantes en el ámbito de la visión por ordenador, en investigación muy activa. Se espera que los sistemas inteligentes modernos realicen estas tareas sin intervención humana, y la visión por ordenador permita a estos sistemas procesar y reconocer objetos y escenas en imágenes y vídeos.

Tras los recientes avances en aprendizaje profundo, la visión por ordenador ha dado un gran salto, superando incluso la capacidad humana en el ámbito de la clasificación de imágenes, la recuperación de imágenes, el reconocimiento de objetos, etc. El principal principio subyacente al reconocimiento de objetos se basa en gran medida en dos pasos principales: (1) la representación de la imagen y (2) la detección de objetos. De este modo, inspirados por el impacto del aprendizaje profundo en el campo de la visión por ordenador, en esta tesis, abordamos dos importantes aplicaciones: (a) la recuperación de imágenes basada en el contenido (CBIR, por sus siglas en inglés *content-based image retrieval*) y (b) el reconocimiento de escenas en entornos interiores, centrándonos especialmente en la representación de imágenes y la detección de objetos.

En los últimos años, debido a la disponibilidad de teléfonos móviles y cámaras por casi toda la población, se ha producido una rápida proliferación de imágenes digitales del orden de miles de millones. De hecho, cada día se suben millones de imágenes a través de muchas aplicaciones de medios sociales como *Instagram*, *Facebook*, *Twitter*, etc. Debido a esta enorme cantidad de datos adquiridos digitalmente, la CBIR es una tarea desafiante y no trivial en comparación con una década antes. Los algoritmos modernos de búsqueda de imágenes, con la ayuda de la inteligencia artificial (IA), consiguen comprender el contexto y el contenido de las imágenes y, a continuación, devolver las imágenes relacionadas con una dada. Además, debido a la digitalización en la era actual, la cantidad de datos continuará aumentando, y como resultado, se abre un mayor alcance a los investigadores para idear métodos que puedan mejorar la eficacia de la búsqueda de imágenes utilizando sistemas CBIR en términos de escalabilidad, precisión y velocidad.

En la figura 1, mostramos los pasos generales en un sistema CBIR. Las características se pueden extraer utilizando modelos basados en el aprendizaje profundo, como las redes neuronales convolucionales (CNN, del inglés *convolutional neural networks*) y los autocodificadores. No sólo se obtienen las características de una imagen, sino que también se pueden explotar las diferentes capas de una CNN para obtener características de bajo a alto nivel, que se pueden utilizar para aplicaciones de búsqueda de imágenes. Además, el aprendizaje por transferencia puede aplicarse sobre las redes CNN preentrenadas para la recuperación de imágenes entre

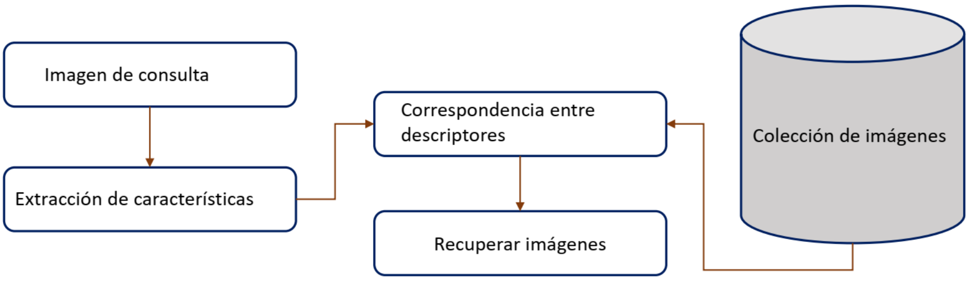


Imagen de consulta
Extracción de características
Correspondencia entre descriptores
Recuperar imágenes
Colección de imágenes

Figura 1: Pasos para la recuperación de imágenes basada en el contenido

dominios mejorando las características. Al igual que las CNN, los autocodificadores convolucionales también pueden utilizarse para la recuperación de instancias. A diferencia de las CNN, los autocodificadores pueden entrenarse de forma no supervisada y aprenden características de bajo nivel de una imagen de entrada. Estas características de bajo nivel se denominan características latentes, que son una representación codificada y compacta de baja dimensión de la imagen de entrada. De hecho, debido a lo compacto que es el espacio latente, el vector resultante puede utilizarse como un descriptor de características para una recuperación de imágenes más rápida, incluso en tiempo real. Estos métodos basados en el aprendizaje profundo tratan de percibir la semántica de alto orden de las imágenes basándose en sus atributos de imagen de bajo nivel y pueden superar a los métodos convencionales realizados a mano.

La tarea CBIR puede clasificarse en dos tipos: recuperación a nivel de instancia y recuperación a nivel de categoría. La recuperación de imágenes a nivel de instancia o recuperación de instancias es el problema de recuperar imágenes de una base de datos que representan el mismo objeto o imagen que el dado en una imagen de consulta (Tan et al., 2021). Mientras que en la recuperación a nivel de categoría el objetivo es buscar imágenes (u objetos) similares de la misma categoría y existe, por tanto, flexibilidad a la hora de fijar la relevancia. En este trabajo, nos centramos principalmente en la recuperación de imágenes a nivel de instancia, donde el objetivo es crear un método eficaz que sea competitivo con las aplicaciones actuales para recuperar imágenes que contengan una instancia de consulta concreta de bases de datos de imágenes a gran escala.

En relación a la recuperación de instancias, la detección automática de objetos es uno de los componentes más importantes de los algoritmos de búsqueda, ya que un usuario puede utilizar cualquier objeto y parche de imagen como consulta. Mediante la detección de objetos se pueden buscar objetos específicos de las categorías detectadas. Si un usuario desea buscar un objeto de consulta para el que la red no está entrenada, el reto se multiplica. En este caso, una solución sería calcular las similitudes del descriptor de la consulta con los descriptores de todos los objetos detectados. En cualquier caso, la dificultad de un sistema de recuperación de instancias de este tipo radica en la presencia de múltiples objetos que son visualmente similares a la consulta suministrada. Para resolver este problema, es necesario un descriptor altamente discriminativo que sea capaz de crear un sistema eficaz de recuperación de imágenes, y este es el principal foco de motivación de esta tesis.

Otra aplicación importante en el campo de la visión por computador que abordamos en esta tesis es el reconocimiento de escenas en interiores. Este proporciona una descripción fundamental del contenido de la imagen asignando etiquetas semánticas en lugar de limitarse a enumerar los objetos que ha detectado y reconocido. El reconocimiento de escenas se ha utilizado en una amplia gama de aplicaciones, como la robótica inteligente, la interacción persona-ordenador, la navegación autónoma y la videovigilancia. Además, el reconocimiento de escenas se considera un requisito previo para algunas tareas de visión por ordenador, como la comprensión de escenas y la recuperación de imágenes. De hecho, debido al éxito del aprendizaje profundo en los últimos años, el reconocimiento de escenas logra grandes resultados, específicamente, las CNN han mejorado significativamente los resultados.

Desde el punto de vista de la aplicación, la principal razón de realizar este trabajo sobre sistemas CBIR para la recuperación a nivel de instancia y el reconocimiento de escenas es el apoyo a los organismos encargados de la aplicación de la ley (LEA) para investigar las escenas del crimen utilizando nuevas tecnologías basadas en la IA. Últimamente, los delitos aumentan a un ritmo increíblemente rápido, con nuevas tendencias que surgen constantemente a medida que los delincuentes utilizan las nuevas tecnologías contra el gobierno, las organizaciones empresariales y los individuos. Los delincuentes pueden causar graves daños y amenazas a personas de todo el mundo. A medida que el mundo progresa en términos de avance tecnológico, proporcionalmente los delitos se vuelven más ágiles. Uno de los delitos de gran relevancia para las fuerzas de seguridad es la explotación sexual de los niños. En estos casos, las pistas derivadas de las imágenes pueden potenciar el trabajo de investigación de los departamentos forenses. La recuperación de imágenes para la investigación de la escena del crimen (Liu and Wu, 2019; Liu et al., 2017) y el reconocimiento de escenas pueden ayudar a descubrir diversos delitos mediante la vinculación de imágenes o vídeos similares.

Esta tesis se estructura en torno a tres núcleos de investigación que se detallan a continuación.

1.1.1 Descriptores de color neuronales para la recuperación de instancias

La recuperación de instancias requiere la detección correcta de todos los objetos (instancias) junto con la localización precisa de cada uno de ellos. Para buscar los objetos o instancias más similares a una consulta dada en un conjunto de datos es necesario comparar la consulta con todas las instancias posibles presentes en el conjunto de datos. Las imágenes pueden contener objetos de diseños diversos. Además, los objetos pueden aparecer parcialmente ocluidos o en entornos desordenados, lo que puede dar lugar a variaciones significativas en cuanto a puntos de vista, escala, rotación, traslación o iluminación de los objetos. Para hacer frente a estos retos, los trabajos más recientes se centran en la generación de propuestas de objetos en imágenes utilizando CNNs de extremo a extremo para aprender la ubicación de los objetos. El aspecto más crucial de esta tarea de recuperación es la localización de los objetos, que depende en gran medida de su apariencia. Sin duda, el color es una clave esencial para estimar el parecido de los objetos. El color es una de las características visuales más básicas y sencillas que representa el contenido espectral de las imágenes. Además, las características basadas en el color deben ser invariables a la traslación o rotación de los píxeles en las imágenes. Para el CBIR a nivel de instancia, la características de la imagen de consulta y de las imágenes objetivo tienen que ser casi o completamente similares. Utilizando el color como característica, podemos identificar y discriminar entre diferentes imágenes y los objetos presentes en ellas. Dado que el color proporciona información vital sobre las imágenes y para aumentar el poder de discriminación de las características neuronales sin necesidad de un ajuste fino, proponemos utilizar diferentes espacios de color y combinaciones de canales de color para transformar las características CNN en descriptores robustos.

1.1.2 Recuperación de instancias basada en la textura

Además de las características de color, los patrones de textura también desempeñan un papel importante en la representación de imágenes como característica discriminatoria para la recuperación de instancias basadas en la textura. En general, la mayoría de los sistemas CBIR extraen características que representan varias formas y patrones presentes en una imagen. Sin embargo, un parche de consulta que contiene solo un patrón de textura sin información de contorno presenta un escenario más desafiante para recuperar imágenes de manera efectiva. A veces, las imágenes de escenas de interior están formadas por objetos desordenados que suelen ser tejidos y telas, como por ejemplo un jersey tirado en el suelo o una colcha de una cama. Estas texturas de telas y tejidos contienen patrones casi repetitivos y, debido a la presencia de imágenes de textura con gran variación intraclase y similitudes interclase, la tarea de recuperación se convierte en un reto incluso con algoritmos basados en el aprendizaje profundo. Sin embargo, como en las imágenes de textura los patrones se repiten en toda la imagen, cada parte de la imagen tiene una frecuencia común. Esta información puede ser explotada mediante la transformada de Fourier y puede ser utilizada junto con el aprendizaje profundo para crear un sistema de recuperación eficiente.

1.1.3 Reconocimiento de escenas en interiores mediante un enfoque centrado en los objetos y la escena

Una escena es un entorno del mundo real que contiene múltiples objetos y superficies organizados de forma significativa. El reconocimiento automático de escenas es un problema de investigación de gran recorrido, en el que un algoritmo debe predecir etiquetas como "dormitorio", "baño" o "cocina" a una imagen de entrada basándose en el contenido general de la imagen proporcionada. La complejidad de la tarea de reconocimiento de escenas radica en parte en la ambigüedad entre diferentes categorías de escenas con apariencias similares y conjuntos de objetos que definen una escena que puede ser muy similar a otra. El reconocimiento de escenas no sólo se refiere a los objetos presentes, sino a sus relaciones semánticas y a su información contextual con respecto al fondo. Aunque se ha demostrado que las CNN aportan soluciones automáticas, la complejidad del problema aumenta con el número de categorías. Además, para entrenar un modelo que clasifique varias escenas, es necesario equilibrar los conjuntos de datos de entrenamiento con millones de imágenes. Por otra parte, los modelos de reconocimiento de escenas que funcionan correctamente en exteriores no lo hacen tan bien en interiores. Esto se debe a que las imágenes de exteriores pueden caracterizarse por propiedades espaciales globales, y las de interiores se definen por los diversos objetos que contienen. Es necesario aprender las propiedades compartidas de los objetos de interior, como los muebles, las camas, las sillas y las instancias duplicadas, y también hay que tener en cuenta el contexto global de las imágenes. Debido a estos retos inherentes, en esta tesis, proponemos proporcionar una solución mediante la creación de una red centrada en las propiedades de los objetos y de la escena global, introduciendo una novedosa arquitectura profunda híbrida basada en CNN para el reconocimiento de escenas de interior.

2 Revisión del estado del arte

2.1 Representación de imágenes para la recuperación de instancias

Una imagen está representada principalmente por características primitivas como el color, la textura y la forma, y por ello, se han diseñado muchos algoritmos para extraer esas características utilizando métodos globales y locales. Los métodos globales extraen características de toda la imagen, mientras que el método local se centra en parches y regiones de la imagen. En los primeros años, las características globales han sido el pilar de la recuperación de imágenes. El rendimiento de los descriptores globales es limitado cuando las imágenes tienen una constitución visual compleja. Los descriptores de características locales notables, como SIFT (Lowe, 2004b), PCA-SIFT (Sukthankar, 2004), SURF (Bay et al., 2006) y HOG (Dalal and

Triggs, 2005) surgieron como descriptores populares para la recuperación eficaz de imágenes. A diferencia de los descriptores globales, los descriptores locales han sido ampliamente utilizados para la extracción de características de parches, en las que una imagen de consulta puede ser buscada localmente y comparada con numerosos parches en una base de datos de imágenes.

2.1.1 Descriptores de color

En la literatura, se han propuesto varios descriptores basados en el color para la recuperación de imágenes, focalizándose en aumentar la invariabilidad de la iluminación y el poder de discriminación. Los enfoques anteriores utilizaban modelos de apariencia como el histograma de colores RGB (Cai et al., 2008), el histograma regional YCbCr (Alahi et al., 2010), el espaciograma RGB (Cong et al., 2010), y también la combinación de textura con descriptores de color (Shao et al., 2014). Desde una perspectiva general, se fundamentaron en aumentar la invariabilidad a la iluminación y el poder discriminativo de dichos descriptores. (Van De Sande et al., 2010) estudiaron las propiedades de invariancia y el poder discriminativo de los descriptores de color basados en SIFT e histogramas, en los que, además del reconocimiento de objetos, los descriptores pueden utilizarse para sistemas CBIR para buscar imágenes similares. (Pujari et al., 2010) presentaron un marco que utiliza las características de color y forma de los espacios Lab y HSV para recuperar las características de los bordes, y los experimentos realizados en el conjunto de datos Corel demostraron la eficacia del método. (Alzu'bi et al., 2015) introdujeron un descriptor de imagen optimizado que combina el histograma de color en el espacio HSV con los descriptores rootSIFT y superaron a muchos de los métodos más avanzados. (Cortes et al., 2016) evaluaron 11 descriptores de imagen y concluyeron que las combinaciones de descriptores Gabor y descriptores neuronales de color dominante proporcionan un mejor rendimiento. Últimamente, algunos trabajos proponen combinar el color con otros descriptores de textura o forma. En esta línea, (Ahmed et al., 2018) utilizaron el histograma de bordes Canny combinado con ondículas discretas en imágenes de color YCbCr o, más recientemente, (Sotoodeh, a Moosavi and Boostani, 2019) presentaron dos enfoques para extraer características discriminativas para la recuperación de imágenes en color, basadas en el patrón binario local de media radial.

2.1.2 Descriptores para la recuperación de texturas

La creación de descriptores de textura eficientes para caracterizar la imagen es esencial en los trabajos relacionados con la recuperación y clasificación de imágenes basadas en la textura (Alzu'bi et al., 2015; García-Olalla et al., 2018; Kwitt and Uhl, 2008). En la literatura, se han propuesto recientemente muchos descriptores para el análisis de la textura, por ejemplo, en (Ouslimani et al., 2019) se propuso un descriptor de textura invariante a la rotación para abordar la tarea de clasificación, y Pham (2018) introdujeron un método para la recuperación de la textura utilizando la extracción de características multiescala. Para el reconocimiento de imágenes de textura, Tuncer, Dogan and Ertam (2019) utilizaron una red neuronal para la extracción de características de textura, y más tarde, introdujeron un novedoso descriptor de imagen local (Tuncer, Dogan and Ataman, 2019) para la extracción de características de textura inspirado en el juego de ajedrez. El objetivo principal de todos estos trabajos es satisfacer las demandas de ciertas aplicaciones con respecto a CBIR. Por ejemplo, la recuperación de imágenes basada en la textura se ha utilizado en la industria textil y de la confección (King and Lau, 1996), donde la ropa y los textiles pueden representarse mediante descriptores de textura. Además, en las tiendas de textiles, la recuperación de imágenes textiles deseadas a partir de enormes bases de datos utilizando una consulta es una necesidad tanto para los clientes como para los minoristas para sugerir productos (D’Amato et al., 2016; Wong, 2017).

2.1.3 Aprendizaje profundo para la recuperación de imágenes

Desde la irrupción del aprendizaje profundo en el ámbito de la visión por ordenador, las activaciones neuronales de una red preentrenada han servido como un descriptor de imagen robusto. Varios trabajos Han et al. (2019); Zhu et al. (2019) utilizaron las activaciones neuronales extraídas de las capas intermedias y lograron resultados de vanguardia en tareas de recuperación de instancias. Radenović, Tolias and Chum (2018) propusieron reajustar las CNN para la recuperación de imágenes introduciendo una capa de agrupación generalizada entrenable que aumenta el rendimiento de la recuperación. Dado que las características obtenidas de las CNNs demostraron un buen rendimiento, Siméoni et al. (2019) propusieron un método conocido como *deep spatial matching* para la recuperación de imágenes que utiliza descriptores de imagen extraídos de las activaciones de las redes neuronales convolucionales mediante *global pooling*. Del mismo modo, Noh et al. (2017) introdujeron *deep local feature* (DELF), también basado en CNNs que se entrenan con anotaciones a nivel de imagen en un conjunto de datos de referencia. Gordo et al. (2017) presentaron una arquitectura siamesa que produce una representación global de imágenes que es adecuada para la recuperación de imágenes. Recientemente, Wang, Huang, Zhang, Feng, Zhang and Fan (2020) introdujeron una red neuronal profunda en cascada con representación profunda para establecer relaciones multimodales para tareas de recuperación de imágenes. Para la recuperación de imágenes médicas, Cai et al. (2019) propusieron un diseño utilizando CNN y *hashing* supervisado, que adopta una red siamesa. Dubey et al. (2019) propusieron un descriptor para la recuperación de imágenes biomédicas, que se computa mediante la fusión de los mapas de características RELU de una AlexNet preentrenada, obtenida a partir de imágenes decodificadas en el plano de bits. Recientemente, Maji and Bose

(2020) propusieron utilizar características derivadas de una CNN entrenada para un problema de clasificación de imágenes de gran tamaño. Además, el aprendizaje profundo también juega un papel clave en la recuperación de imágenes de teledetección. En (Zhou et al., 2015), se propuso un diseño basado en autoencoders para recuperar imágenes aéreas de teledetección utilizando la representación aprendida codificada como un descriptor de características. Zhou, Newsam, Li and Shao (2017) investigaron la extracción de características de CNN profundas para la recuperación de imágenes de teledetección de alta resolución utilizando CNN preentrenadas y reajustadas. Sin embargo, el principal desafío es la falta de disponibilidad de un conjunto de datos a gran escala, y para abordar este problema, Zhou et al. (2018) introdujeron un conjunto de datos de teledetección a gran escala conocido como PatternNet que es adecuado para el entrenamiento de redes neuronales profundas. Recientemente, Shao et al. (2020) propusieron una red neuronal totalmente convolucional para la recuperación de imágenes de teledetección de múltiples etiquetas mediante la extracción de características convolucionales de la región.

2.2 Detección de objetos

En esta sección, revisamos el estado del arte de los algoritmos de detección de objetos que ayudan a la tarea de búsqueda visual para la recuperación de instancias y a la tarea de reconocimiento de escenas. En el año 2014, Girshick et al. (2014) propuso R-CNN, que utiliza regiones con características CNN para la detección de objetos. R-CNN utiliza una ventana deslizante para generar 2000 propuestas, que luego se introducen en una CNN para generar características. La R-CNN obtuvo un rendimiento significativo, pero tiene ciertos inconvenientes. La R-CNN realiza 2.000 pases hacia delante a la CNN, por lo que es costosa desde el punto de vista computacional, lo que hace que la detección de objetos sea lenta. Para resolver este problema, He et al. (2015) introdujo las redes de agrupación de pirámides espaciales (SPPNet), que evitan el cálculo repetido de las características de la CNN como hace la R-CNN. Sin embargo, la SPPNet ha mejorado la velocidad de detección, pero tiene un entrenamiento en varias etapas, lo que hace que la red sea compleja, y sólo reajusta las capas totalmente conectadas ignorando el resto.

Para mejorar el sistema propuesto de R-CNN y SPPNet para una detección más rápida, Girshick (2015) propuso Fast R-CNN. A diferencia de las otras dos redes, en Fast RCNN el detector de objetos y el regresor de la caja delimitadora pueden ser entrenados simultáneamente. Además de reducir el tiempo de entrenamiento, se logró una notable ganancia de rendimiento en comparación con R-CNN, y con una velocidad de detección más de 200 veces más rápida que R-CNN. En 2015, Faster R-CNN fue propuesto por Ren et al. (2017), siendo el primer detector de objetos casi en tiempo real basado en aprendizaje profundo con una velocidad de detección de 0,12 segundos por imagen. Faster R-CNN introdujo la red de propuestas regionales (RPN, del inglés *region proposal network*) para generar propuestas dentro de la red en lugar de utilizar algoritmos externos, y también permitió el entrenamiento de extremo a extremo. Sin embargo, Faster R-CNN tiene redundancia computacional, y posteriormente se propusieron varias mejoras, como R-FCN (Dai et al., 2016), feature pyramid networks (FPN) (Lin, Dollár, Girshick, He, Hariharan and Belongie, 2017) y Mask R-CNN (He et al., 2017).

A pesar del gran éxito de los detectores de dos etapas antes mencionados en la detección de objetos, estos son computacionalmente costosos, difíciles de optimizar y los rendimientos no se consiguen en tiempo real. Para solucionar estos problemas, se proponen CNNs basadas en detectores de una etapa, estando YOLO (Redmon et al., 2016), SSD (Liu et al., 2016) y RetinaNet (Lin, Goyal, Girshick, He and Dollár, 2017) entre los más populares. YOLO-v1 fue el primer detector de una etapa en tiempo real en la era de las redes neuronales convolucionales que podía entrenarse de extremo a extremo y podía optimizarse fácilmente. Posteriormente, se propuso YOLO-v2, que mejoró YOLO-v1 en términos de precisión de detección y alcanzó una mayor velocidad de detección. Incluso después de lograr una gran mejora, YOLO-v2 tiene una precisión de localización pobre en comparación con las CNN basadas en regiones. Para mejorar este inconveniente, se propuso un detector de un solo disparo (SSD, del inglés *single shot detector*) con técnicas de detección multirreferencia y multirresolución. El SSD tiene una mayor velocidad de detección y precisión en comparación con YOLO, y también puede detectar objetos más pequeños con alta precisión. Sin embargo, incluso después de varias mejoras, la precisión del detector de objetos de una etapa es inferior a la alcanzada por los detectores de objetos de dos etapas. Esto se debe al desequilibrio entre la clase de fondo y la de primer plano durante el proceso de entrenamiento. Para hacer frente a este problema, los autores introducen una nueva función de pérdida conocida como pérdida focal (Lin, Goyal, Girshick, He and Dollár, 2017), debido a la cual el SSD logró una precisión comparable a la de los detectores de dos etapas, manteniendo una velocidad de detección más rápida. En 2018, se introdujo YOLO-v3 (Redmon and Farhadi, 2018) con algunas mejoras incrementales sobre YOLO-v2 en términos de precisión y velocidad. La red es tan precisa como SSD y también unas tres veces más rápida. Recientemente, YOLO-v4 (Bochkovskiy et al., 2020) ha sido la actualización en la evolución de YOLO. Además, una variante de YOLO-v4 conocida como Scaled-YOLO-v4 fue propuesta por Wang, Bochkovskiy and Liao (2020). En este enfoque de escalado, la profundidad, la anchura y la resolución de la arquitectura pueden modificarse manteniendo una velocidad y precisión óptimas. Además, Scaled-YOLO-v4 obtiene actualmente la mayor precisión en el conjunto de datos MS-COCO superando a otros detectores recientes del estado del arte como los propuestos por Tan et al. (2020); Du et al. (2020); Wang, Bochkovskiy and Liao (2020). En esta misma línea de investigación, Fan et al. (2020) propusieron una red conocida como detección de objetos de pocos disparos que tiene como objetivo detectar objetos no vistos sólo con unos pocos ejemplos anotados. Al igual que las R-CNN, esta red también emplea RPN en su fondo para detectar objetos. Recientemente, Shinya (2021) diseñó detectores de objetos llamados UniverseNets, que superaron todas las líneas de base y lograron resultados de vanguardia en los puntos de referencia existentes.

2.3 Reconocimiento de escenas

Las imágenes de escenas están definidas por algunas regiones cruciales, que pueden ser parches u objetos. Algunos métodos (Li et al., 2010; Pandey and Lazebnik, 2011) utilizan la detección de objetos para determinar dichas regiones discriminativas para clasificar las escenas. Además, algunos enfoques utilizan un gran número de parches de imagen para identificar regiones importantes (Singh et al., 2012; Juneja et al., 2013; Yuan et al., 2016). Lin et al. (2014) propusieron un método que se basa en filtros de parte para obtener las respuestas de las regiones importantes. Además, algunos enfoques (Song, Jiang and Herranz, 2017; Wu et al., 2015; Song et al., 2016) explotan la información proporcionada por la distribución de patrones de objetos para diferentes escenas. Sin embargo, el reto común con tales algoritmos de reconocimiento de escenas se focaliza en las variaciones inter-clase e intra-clase de las clases de objetos. Para superar este problema, Zuo et al. (2014) introdujeron un método conocido como aprendizaje de características discriminativas y compartibles (DSFL, del inglés *discriminative supplementary representation learning*), que hace que las características aprendidas de las mismas clases se acerquen y las características aprendidas de diferentes clases se alejen entre sí.

El aprendizaje profundo ha demostrado ser un enfoque eficaz en el reconocimiento de escenas. Las capas inferiores de las CNN capturan características locales mientras que las capas superiores generan características más abstractas, y basándose en esta propiedad, Guo et al. (2016); Xie et al. (2015); Tang et al. (2017) fusionaron características correspondientes a capas convolucionales multiescala para el reconocimiento de escenas. Del mismo modo, Guo et al. (2016) propusieron una arquitectura CNN multirresolución que puede capturar características correspondientes a diferentes capas para la comprensión de la escena. Dixit et al. (2015) introdujeron el vector semántico Fisher para fusionar las características de las capas convolucionales y totalmente conectadas de las CNN. Wang et al. (2017) presentaron una red llamada PatchNet, que agrega tanto el objeto como las características holísticas de la escena para desarrollar una representación visual efectiva de las escenas. Otro diseño que combina la información del objeto con la de la escena es la FOSNet , que es una CNN de extremo a extremo propuesta por Seong et al. (2020). Recientemente, Ma et al. (2021) han propuesto SceneNet, una arquitectura neuronal profunda que se utiliza para el reconocimiento de escenas en imágenes de teledetección.

Para reconocer escenas de interiores, Basu et al. (2020) entrenaron una red neuronal de cápsulas, que obtuvo un mejor rendimiento en comparación con otros métodos basados en CNN. Las características extraídas de las CNN también pueden utilizarse con diferentes métodos de reconocimiento de escenas. Por ejemplo, Xie et al. (2015) utilizaron características de VGG y AlexNet con la representación vectorial de Fisher por convolución para combinar las características locales con la disposición global de la escena. Yoo et al. (2014) utilizaron la activación de CNN multiescala con el marco del kernel de Fisher y obtuvieron una ganancia de rendimiento significativa en el conjunto de datos MIT-67. Cimpoi et al. (2015) mejoraron la generalización del reconocimiento de escenas extrayendo información de textura del banco de filtros CNN y de la bolsa de palabras visuales. Recientemente, López-Cifuentes et al. (2020) propusieron una CNN multimodal que combina la información de contexto con la imagen de la escena utilizando un módulo de atención. Además, para mejorar la precisión del reconocimiento, Li et al. (2019) introdujeron MAPnet que fusiona la información de profundidad y la imagen RBG. En lo que respecta a la percepción de escenas de interiores para robots móviles, los mismos autores diseñaron una estructura CNN poco profunda y eficiente que logró una mayor precisión en el reconocimiento de escenas utilizando imágenes de cámaras monoculares. En particular, la tarea de reconocimiento de escenas de interiores se basa normalmente en la detección de objetos de interior, y por lo tanto la mayoría de los algoritmos emplean detectores de objetos para identificar los objetos inherentes.

3 Descriptores de color neuronales para la recuperación de instancias

El objetivo en esta sección es presentar un enfoque para la recuperación de instancias mediante la creación de características discriminativas utilizando CNN y modelos de color. Se trata de una tarea de CBIR, que pretende recuperar todas las imágenes de un corpus de un gran conjunto de datos que contengan el mismo objeto que el de la consulta.

3.1 Método

Para encontrar si una región de consulta está presente en una imagen, es necesario revisar la imagen completa, parte por parte, para comprobar su presencia. Para ello, se utiliza un detector de objetos para generar propuestas de objetos. Para determinar si la región de la consulta está presente en la imagen, los descriptores de cada una de las propuestas se comparan con el descriptor de la imagen de la consulta. Para realizar la comparación, se extraen las características neuronales de las imágenes como descriptores y la comparación se realiza utilizando una métrica de distancia. Si la métrica calculada es superior a un umbral determinado, recuperamos esa instancia concreta suponiendo que la consulta está presente en esa imagen.

Además, en este escenario, cuando se trata de conjuntos de datos con objetos multivistas o rotados, la tarea de recuperación se vuelve aún más difícil. Dado que los objetos asimétricos pueden aparecer rotados en las imágenes del conjunto de datos, la mayoría de los métodos posiblemente fallarían a la hora de recuperar vistas rotadas del mismo objeto en las que la apariencia es notablemente diferente a la de la consulta. Con respecto a los retos relacionados con la tarea de recuperación de la misma instancia, investigamos el uso de diferentes modelos de color y características intermedias de las arquitecturas CNN para desarrollar descriptores robustos.

3.1.1 Arquitectura

La arquitectura presentada en esta tesis comprende dos redes diferentes preentrenadas que facilitan la búsqueda de instancias locales y la creación de descriptores discriminativos. Para la búsqueda de instancias locales, utilizamos R-FCN para generar propuestas sobre las imágenes del conjunto de datos con el fin de comparar la instancia de consulta con esas propuestas. Por su lado, VGG-16 sirve como extractor de características tanto para la imagen de consulta como para las propuestas. Además, ambas redes se utilizan bajo un único diseño, en el que las propuestas candidatas generadas por la red R-FCN se pasan directamente como entrada a la red VGG-16 para calcular los descriptores de color. En la Figura 2, ilustramos nuestra arquitectura.

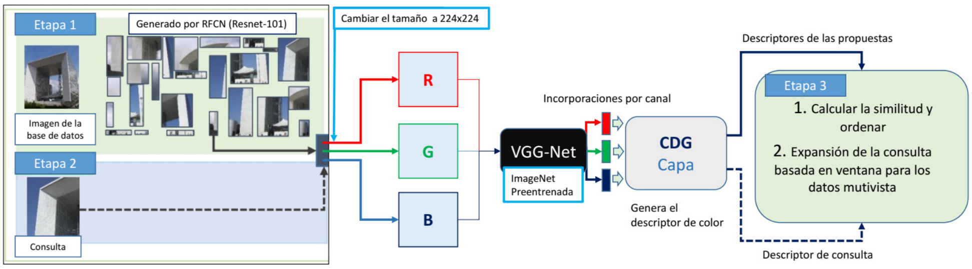

Etapa 1
Generado por RFCN (Resnet-101)
Cambiar el tamaño a 224x224
Imagen de la base de datos
Etapa 2
Consulta
R
G
B
VGG-Net
ImageNet Preentrenada
Incorporaciones por canal
CDG Capa
Genera el descriptor de color
Descriptores de las propuestas
Etapa 3
1. Calcular la similitud y ordenar
2. Expansión de la consulta basada en ventana para los datos mutivista
Descriptor de consulta

Figura 2: Arquitectura principal del enfoque de recuperación de instancias. Está constituida por dos partes: La RPN de la R-FCN y la red VGG-16 para la extracción de características. Las propuestas son generadas por la RPN, que luego se dan como entrada a la red VGG-16 para la extracción de características basadas en la región.

3.1.2 Creación de descriptores de color

El color juega un papel esencial en la obtención de la información visual sobre los objetos presentes en las imágenes. Nuestro objetivo es aprovechar esa información para crear vectores de características de color discriminatorios de alto nivel utilizando el aprendizaje profundo. Para crear descriptores neuronales de color, primero extraemos características de la red VGG-16 preentrenada de ImageNet. En concreto, extreamos las características correspondientes a los tres canales de color (R, G y B), y luego definimos una capa de generación de descriptores neurales de color (CDG, del inglés *colour descriptor generation*) para crear descriptores de color. Para obtener un descriptor neural de color robusto, evaluamos diferentes espacios de color y combinaciones de canales de color, inspirados en el trabajo de Van De Sande et al. (2010). Los descriptores de color propuestos en esta tesis son:

- *NE-Raw*: Este descriptor se genera pasando una imagen por la red sin ninguna modificación de la capa de entrada.

- *NE-O y NE-O3*: Estos descriptores se obtienen utilizando el espacio de color oponente, que es una combinación de características neuronales basadas en los canales del espacio de color oponente. El NE-O encapsula información de intensidad y de color y el NE-O3 se basa sólo en la intensidad.

- *NE-TCD*: Este descriptor es invariable a los cambios en las condiciones de iluminación y se forma normalizando independientemente las distribuciones de valores de los píxeles de cada uno de los canales.

- *NE-C*: Creamos este descriptor pasando los DCF R^*, G^* y B^* de los tres canales de color por la capa CDG. El descriptor resultante es una concatenación de las características neuronales correspondientes a esos canales de color

En la Figura 2 describimos la arquitectura propuesta para generar descriptores de color. Además, para nuestros experimentos seleccionamos el descriptor de mejor rendimiento, que es el NE-C.

3.1.3 Proceso de recuperación de instancias

El método de recuperación de instancias propuesto consta de tres fases (1) Extracción de características del conjunto de datos, (2) Extracción de características de la consulta, y (3) Recuperación y ordenación de las k instancias principales en función del valor de similitud. En primer lugar, procesamos todas las imágenes del conjunto de datos para generar los descriptores de color, que son necesarios para recuperar las imágenes con objetos similares a la consulta. A continuación, obtenemos un descriptor de color de la instancia consultada. Por último, tratamos de recuperar las imágenes del conjunto de datos que más se parecen a la instancia consultada. Para ello, calculamos la similitud entre la instancia consultada y las propuestas de todas las imágenes del conjunto de datos. A continuación, creamos una lista de aciertos clasificando las imágenes del conjunto de datos en orden descendente considerando la similitud de cada imagen como la mayor similitud de cualquiera de sus propuestas y descartando las imágenes cuya similitud es inferior a un umbral establecido. Además, también recuperamos imágenes de conjuntos de datos multivista. En general, los conjuntos de datos multivista contienen objetos capturados desde varios puntos de vista, por lo que es difícil recuperarlos todos con una sola consulta. Para solucionar este problema, en esta tesis implementamos una técnica de expansión de la consulta para recuperar dichos objetos multivista en cascada. En este enfoque, primero seleccionamos una consulta inicial para recuperar las principales k instancias similares, y luego utilizamos una zancada de tamaño s para seleccionar la s^{th} instancia recuperada que se utilizará como la siguiente consulta.

3.2 Experimentos y resultados

En primer lugar, para determinar la eficacia de la arquitectura base propuesta, comparamos el rendimiento considerando un número diferente de propuestas con las CNNs basadas en regiones del estado del arte: Fully Convolutional One-Stage object detection (FCOS) (Tian et al., 2019), Faster R-CNN con VGG-16 y R-FCN con ResNet. Durante la evaluación, descubrimos que la mayor precisión de 96 % se obtuvo con *NE-C* con 100 propuestas en comparación con otros enfoques.

Para evaluar la eficacia de nuestro descriptor de color propuesto, utilizamos protocolos de evaluación estándar para experimentar con cuatro conjuntos de datos (1) COIL-100 (Nene et al., 1996) (2) Paris-k (Philbin et al., 2008) (3) Revisiting Paris-6k (Radenović, Iscen, Tolias, Avrithis and Chum, 2018) e (4) INSTRE-M (Wang and Jiang, 2015). Calculamos la precisión media (mAP, del ingles *mean average precision*) para medir el rendimiento en todos los experimentos. En primer lugar, calculamos la precisión media (AP, del ingles *average precision*) y, a continuación, promediamos las AP de todas las consultas para obtener la mAP.

En el conjunto de datos Paris-6K, para medir la eficacia de los descriptores propuestos, primero obtuvimos sus mAPs considerando las 10 primeras imágenes recuperadas y luego las comparamos con otros métodos actuales. En la Tabla 1, presentamos los mAPs para las 10 primeras imágenes recuperadas conseguidos con los diferentes descriptores neurales de color propuestos. El mejor rendimiento se obtiene utilizando el descriptor *NE-C* con un mAP de **97,4**, seguido de *NE-TCD* y *NE-O3* con un mAP de **96,9** y **95,02**. Para comparar con los otros enfoques, hemos seleccionado nuestro descriptor de mejor rendimiento NE-C. En la Tabla 2, comparamos nuestros resultados con los mAPs reportados por varios enfoques del estado del arte. El mAP más alto reportado fue **79,67** por Gem (Radenović, Tolias and Chum, 2018). Utilizando nuestro enfoque, obtuvimos un mAP de **81,70** utilizando el descriptor NE-C y superando así los resultados del estado del arte. A continuación, aplicamos nuestro enfoque a la revisión del conjunto de datos de París basándonos

Cuadro 1: mAP para las 10 primeras imágenes recuperadas obtenidas con el método base (*NE-Raw*) -mostrada en cursiva- y los descriptores neurales de color propuestos en el conjunto de datos Paris 6k. El mejor resultado se muestra en negrita.

Descriptores propuestos	mAP
NE-C	**97.4**
NE-Raw	*92.2*
NE-O	89.4
NE-O3	95.02
NE-TCD	96.9

Cuadro 2: Comparación del rendimiento con respecto a otros métodos avanzados para la recuperación de instancias en función del mAP en el conjunto de datos original Paris 6k. Presentamos la dimensión de los descriptores (dim) y el mAP para todos los métodos.

Método	Dimensión	reajuste	mAP
SPOC; 2015	512	No	63.52
SPOC; 2015	512	Sí	74.09
SPOC(ACK); 2020	256	Sí	74.60
MAC; 2016	512	No	67.02
MAC; 2016	512	Sí	78.73
MAC(ACK); 2020	256	Sí	75.69
RMAC; 2018	512	No	72.02
RMAC; 2018	512	Sí	77.94
RMAC(ACK); 2020	256	Sí	75.76
CROW; 2016	512	No	68.94
CROW; 2016	512	Sí	77.48
CroW(ACK); 2020	256	Sí	75.94
Gem; 2018	512	Sí	79.67
Gem(ACK); 2020	256	Sí	76.26
NE-C(nuestro)	3000	No	**81.70**

en un nuevo protocolo de evaluación como se explica en (Radenović, Iscen, Tolias, Avrithis and Chum, 2018). En la Tabla 3, comparamos el descriptor neural de color *NE-C* -que superó al resto- con algunos enfoques recientes y relevantes del estado del arte. Entre los métodos del estado del arte, el mayor mAP de **80,7** se obtuvo con el método DELF-GLD (Teichmann et al., 2019), en comparación con un mAP de **82,02** utilizando *NE-C*.

A continuación, evaluamos el rendimiento de la recuperación en INSTRE-M, que

Cuadro 3: Comparación del rendimiento con respecto a los métodos del estado del arte para la recuperación de instancias basada en mAPs y precisión media para un corte de 10 (mp@10) en el conjunto de datos revisiting-Paris 6K. Estos métodos fueron presentados en (Teichmann et al., 2019). En negrita, los resultados del método propuesto y los mejores resultados de los métodos del estado del arte.

Método	mAP	mp@10
HesAff-rSIFT-ASMK+SP ; 2016	61.4	97.9
HesAff-rSIFT-ASMK ; 2016	61.2	97.9
ResNet101-R-MAC ; 2016	78.9	96.9
DELF-ASMK+SP ; 2017; 2018	76.9	99.3
AlexNet-GeM ; 2018	58.0	91.6
HesAff-HardNet-ASMK+SP ; 2018	65.2	98.9
VGG16-GeM ; 2018	69.3	97.9
ResNet101-GeM ; 2018	77.2	98.1
ResNet101-GeM+DSM ; 2019	77.4	99.1
DELF-D2R-R-ASMK+SP ; 2019	78.2	**99.4**
DELF-GLD ; 2019	**80.7**	99.1
NE-C (nuestro)	**82.02**	**97.2**

constituye un escenario similar al conjunto de datos Paris 6k. Para evaluar el rendimiento, calculamos el mAP siguiendo el protocolo descrito por Iscen et al. (2017), que utiliza 1.250 imágenes de consulta. En la Tabla 4 presentamos los resultados obtenidos, y se puede observar que la concatenación de DCFs específicos de cada canal, *NE-C*, produjo el mejor rendimiento con un mAP de **78,8** seguido de *NE-TCD* con un mAP **77,5**.

4 Recuperación de texturas

La recuperación de instancias basada en la textura se realiza normalmente en imágenes que presentan un único patrón de textura y se aplica principalmente a la recuperación de tejidos o textiles. En este trabajo, lo aplicamos a imágenes de escenas interiores que suelen presentar muchos patrones de textura diferentes, lo que constituye un problema más desafiante. Estos sistemas de recuperación, junto con la recuperación de rostros y objetos, pueden utilizarse como una valiosa herramienta para el análisis de pruebas en la investigación de escenas de crímenes. A pesar de que los recientes enfoques basados en el aprendizaje profundo han mejorado significativamente en muchas tareas de visión por ordenador, la recuperación de texturas sigue siendo un problema abierto.

Cuadro 4: Comparación del rendimiento con métodos avanzados para la recuperación de instancias en función de los mAPs en el conjunto de datos INSTRE. Presentamos el mAP para todos los métodos.

Método	dimensión	mAP
CroW; 2016	512	41.6
CAM; 2017	512	32.5
R-MAC; 2016	512	47.7
R-MAC-ResNet; 2017	2048	62.6
BLCF; 2018	336	63.6
BLCF-Gaussian; 2018	336	63.6
BLCF-SalGAN; 2018	336	69.8
Lin *et el.*; 2019	1024	57.5
NE-C (nuestro)	3000	**78.8**
NE-TCD (nuestro)	3000	**77.5**
NE-O3 (nuestro)	1000	**70.21**

4.1 Metodología

Para abordar la recuperación de texturas, proponemos un nuevo método de recuperación de texturas basado en la transformada de Fourier y en técnicas de aprendizaje profundo. Nuestro enfoque utiliza un autocodificador, y aplicamos el aprendizaje por transferencia al codificador utilizando una red VGG-16 pre-entrenada con el conjunto de datos de ImageNet (Deng et al., 2009). Combinamos la representación de la textura de las imágenes transformadas mediante la transformada discreta de Fourier (DFT, del inglés *discrete Fourier transform*) con las correspondientes imágenes espaciales. A continuación, las imágenes resultantes se utilizan para entrenar la capa de espacio latente y el decodificador con el fin de aprender su representación de la textura. Después de entrenar la red, el codificador VGG con su capa de espacio latente sirve como extractor de características para generar la representación de la textura, que denominamos descriptor de textura de Fourier profundo (DFTD, del inglés *deep Fourier transform descriptor*). Además, integramos el codificador VGG y la capa de espacio latente con la RPN de la R-FCN (Dai et al., 2016) para generar DFTD de múltiples propuestas de textura relativas a una imagen, de modo que un parche de textura consultado puede compararse localmente con varias propuestas. En la figura 3, esbozamos el método propuesto en dos etapas. La etapa 1 ilustra la generación de descriptores de textura y la etapa 2 representa el marco de recuperación. Hasta donde sabemos, este trabajo presenta un nuevo enfoque en el que se utiliza una arquitectura basada en un autoencoder con aprendizaje de transferencia y la transformada de Fourier para la clasificación y recuperación de texturas.

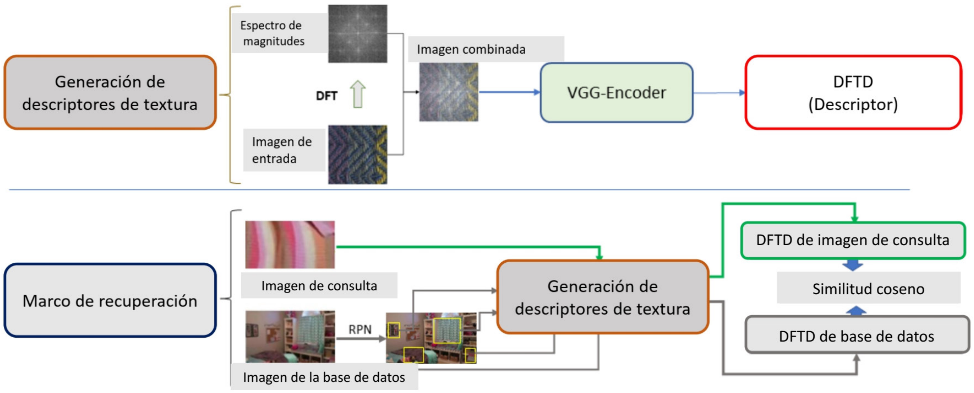


Figura 3: Descripción general del método propuesto de recuperación de texturas. La etapa 1 representa la generación del descriptor de textura DFTD, y la etapa 2 ilustra el marco de recuperación.

4.1.1 Etapa 1: Generación del descriptor de textura

Para generar el descriptor de textura empleamos la transformada de Fourier. Se trata de una herramienta clásica de procesamiento de imágenes que las descompone en las componentes seno y coseno. La salida generada por la transformada de Fourier representa la imagen en el dominio de la frecuencia o de Fourier. Utilizamos la DFT para generar imágenes en frecuencia, que son la transformada de Fourier muestreada que contiene suficientes frecuencias para describir completamente una imagen en un dominio espacial.

La combinación lineal del espectro de magnitudes DFT y las imágenes espaciales: Cuando se trata de un gran corpus de imágenes, la información de frecuencia no es suficiente para distinguir correctamente diferentes patrones de textura, ya que algunas imágenes que pertenecen a diferentes clases pueden tener representaciones de espectro de magnitud DFT similares. Por lo tanto, para resolver este problema, combinamos la información de frecuencia y espacial de la imagen mediante la multiplicación ponderada píxel a píxel de la imagen del espectro de magnitud DFT $F_m(u, v)$ y la imagen espacial $f(x, y)$ para obtener una imagen combinada $B(x, y)$, como se define en la Ec 1.

$$B(x, y) = (1 - \alpha)f(x, y) + \alpha F_m(u, v) \tag{1}$$

Arquitectura propuesta: La arquitectura propuesta se basa en un autocodificador convolucional, en el que la parte del codificador consiste en las capas convolucionales de la arquitectura VGG-16 inicializadas con los pesos de ImageNet. En general, un autocodificador convolucional amplía la estructura básica del autocodificador simple cambiando las capas convolucionales totalmente conectadas. Los codificadores aprenden a codificar la entrada en un conjunto de señales simples y luego intentan reconstruir la entrada a partir de ellas basándose en la representación del espacio latente aprendida por la red. Sin embargo, nuestra arquitectura se diferencia de los autocodificadores tradicionales, ya que, por un lado, aplicamos el aprendizaje por transferencia al codificador y, por otro, el número de capas del codificador y del decodificador no es el mismo. Entre el codificador VGG y el decodificador tenemos la capa de espacio latente, que contiene una representación vectorizada de la imagen de entrada de dimensión 512, el cual define el descriptor propuesto DFTD.

4.1.2 Etapa 2: Conjunto de pasos para la recuperación

El método de recuperación propuesto se compone de tres pasos: (1) extracción de características de la consulta (2) extracción de características del conjunto de datos (3) búsqueda de texturas similares. Primero aplicamos la DFT a la imagen de la consulta para obtener el espectro de magnitud de la DFT y, a continuación, realizamos la multiplicación ponderada píxel a píxel de la consulta y su espectro de magnitud de la DFT para obtener la imagen de la consulta mezclada basada en la DFT $B(x, y)$. A continuación, introducimos la imagen resultante en el modelo de autoencoder convolucional entrenado y extraemos la representación del espacio latente en forma de descriptor. Por otro lado, se obtienen los descriptores DFTD de un conjunto de datos y se almacenan en una base de datos para su posterior comparación con el descriptor de la consulta. En particular, la textura consultada debe compararse con todos los parches de textura distintivos de cada imagen, ya que la consulta puede estar presente en diferentes tamaños, escalas y orientaciones. En esta etapa, integramos la RPN con el codificador convolucional, de modo que las propuestas de región generadas por la RPN se introducen en el codificador para extraer los descriptores DFTD de cada una de las regiones. Los descriptores DFTD de todas las propuestas se almacenan en una base de datos junto con la etiqueta de la imagen a la que pertenecen.

Para recuperar las imágenes que contienen un patrón de textura similar al de la consulta, calculamos la métrica similitud coseno (CS, del inglés *cosine similarity*) de todos los pares de descriptores DFTD de la consulta y de la base de datos. La similitud coseno indica la similitud entre dos vectores; cuanto más alta sea, más similares son los descriptores considerados. En primer lugar, comparamos el descriptor de la consulta con todos los descriptores propuestos presentes en una imagen concreta y, a continuación, almacenamos la propuesta con la mayor puntuación de similitud en una lista de aciertos. Repetimos este proceso para todas las imágenes del conjunto de datos. Por último, ordenamos la lista de aciertos en orden descendente según las puntuaciones de CS y recuperamos las k primeras instancias.

4.1.3 Clasificación de texturas

Este trabajo también presenta un enfoque de clasificación de texturas basado en el aprendizaje por transferencia, en el que utilizamos la misma arquitectura VGGNet preentrenada con el conjunto de datos ImageNet. Añadimos dos capas densas después de las capas convolucionales de la red VGG-16, y la capa de clasificación utiliza la activación softmax. De forma similar al método de recuperación, entrenamos la red con imágenes $B(x, y)$ utilizando los mismos hiperparámetros. Para la evaluación del método, obtenemos las etiquetas de clase basándonos en la activación obtenida de la capa de clasificación de la red, y las comparamos con las etiquetas reales. A diferencia del marco de recuperación, en este enfoque de clasificación, tenemos una capa densa en lugar de la capa de espacio latente.

4.2 Experimentos y resultados

Los experimentos y resultados presentados en este trabajo tienen como objetivo verificar la evaluación de los descriptores DFTD derivados del codificador convolucional entrenado con las imágenes combinadas propuestas. Realizamos dos tipos de experimentos diferentes con dos tipos de conjuntos de datos. El primer tipo de experimentos se realizó con los conjuntos de datos Outex, USPtex y Stex, en los que las imágenes se componen de un único patrón de textura. Mientras que en el segundo tipo de experimentos, consideramos el conjunto de datos TextileTube que consta de múltiples objetos con diferentes texturas para la recuperación de imágenes basadas en la textura. Para comparar nuestro enfoque con otros métodos del estado del arte, utilizamos dos protocolos de evaluación diferentes. El protocolo de evaluación utilizado por Outex, Stex y USPtex es la tasa de recuperación media (ARR, del inglés *average return rate*) propuesta en (Pham, 2018). Sin embargo, los resultados del estado del arte relativos al conjunto de datos TextileTube se proporcionan en base a precision@k, como se propone en (García-Olalla et al., 2018).

Al principio, evaluamos nuestro enfoque para la clasificación de texturas comparando el rendimiento de los descriptores DFTD propuestos frente a los trabajos del estado del arte. En la Tabla 5 presentamos los resultados. La precisión más alta reportada en la literatura fue del 89,62 % por el método Chess-pattern (Tuncer, Dogan and Ataman, 2019) en Outex y el 93,83 % por (Tuncer, Dogan and Ertam, 2019) en USPtex. Con el método propuesto, hemos logrado una precisión del 90,20 % y 95,62 % en Outex y USPtex, respectivamente, superando los resultados reportados en la literatura.

Para la recuperación basada en la textura, la Tabla 6 muestra el rendimiento de nuestro enfoque en relación con la métrica ARR en los conjuntos de datos Outex, Cuadro 5: Comparación del enfoque propuesto con otros métodos del estado del arte en términos de precisión (en porcentaje) en los conjuntos de datos Outex y USPtex para la clasificación de texturas.

Método	Outex	USPtex
LESTP ; 2015	78.00	82.41
LECTP ; 2015	79.06	83.10
PCANet ; 2015	76.04	83.65
LQP ; 2015	81.49	87.83
Multifractals ; 2017	75.07	68.76
Fourier ; 2017	82.21	71.16
ARCS-LBPt ; 2017	85.70	88.85
Chess-Pattern ; 2019	88.9	-
Tuncer *et al.* ; 2019	89.62	93.83
DFTD (nuestro)	**90.20**	**95.62**

Cuadro 6: Comparación del enfoque propuesto con otros métodos del estado del arte en términos de tasa de recuperación promedio (en porcentaje) en los conjuntos de datos Outex, USPtex y Stex para la recuperación de instancias basada en texturas.

Método	Outex	USPtex	Stex
DDBTC ; 2015	66.82	74.97	44.79
CNN-AlexNet ; 2017	69.87	83.57	68.84
CNN-VGG16 ; 2017	72.91	85.03	74.92
CNN-VGG19 ; 2017	73.20	84.22	73.93
LED ; 2018	75.14	87.50	76.71
SLED ; 2018	75.96	88.60	77.88
MS-SLED ; 2018	76.15	89.74	79.87
DFTD (nuestro)	**80.36**	**90.25**	**81.02**

USPtex y Stex, y lo compara con otros métodos de vanguardia. Observamos que los descriptores CNN-VGG19 (Napoletano, 2017) arrojaron resultados competitivos en comparación con las características artesanales, como DDBTC (Guo et al., 2015). También se observa que el enfoque propuesto superó a todos los métodos al obtener un ARR de 80,36 % en Outex, 90,25 % en USPtex y 81,02 % en Stex, mientras que el mejor resultado reportado en la literatura lo consiguió el método MS-LED (Pham, 2018) con un ARR de 76,15 %, 89,74 % y 79,87 %.

Para experimentar con el conjunto de datos TextileTube, hemos empleado dos tipos diferentes de imágenes de consulta. En primer lugar, utilizamos las consultas originales consideradas en (García-Olalla et al., 2018), en las que se tomaron todas las regiones textiles del *ground truth* como imágenes de consulta. Sin embargo, estas consultas contienen partes de los contornos de los objetos y, por tanto, presentan información de forma. Para que las consultas se basen completamente en la textura, recortamos las imágenes de modo que sólo sea visible la parte de la textura, y denominamos a este conjunto de consultas como *Nuevas consultas*. La Figura 4 resume los resultados obtenidos por nuestro método propuesto, que supera a los otros enfoques en términos de precision@k. RPN+DFTD indica los resultados obtenidos por las imágenes de consulta originales, mientras que RPN+DFTD (new) representa los resultados obtenidos utilizando las *Nuevas consultas*. Los resultados utilizando ALBP+HCLOSIB, ALBP, Faster R-CNN, HOG, HOG+CLOSIB y HOG+HCLOSIB están tomados de (García-Olalla et al., 2018).

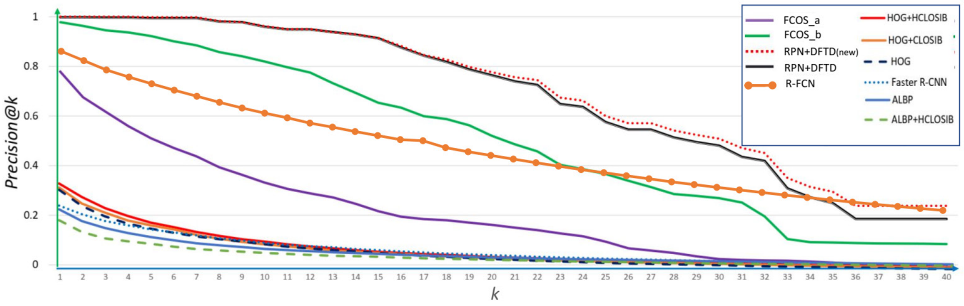


Figura 4: Precision@k de algunos métodos actuales, el método base recientes y nuestro método (RPN + DFTD) en el conjunto de datos de TextileTube. RPN + DFTD indica los resultados con las imágenes de consulta originales y RPN + DFTD (Nuevo) con las *Nuevas consultas*.

5 Reconocimiento y recuperación de escenas en interiores

En esta sección, presentamos un enfoque que aborda el problema del reconocimiento y la recuperación de escenas en interiores combinando la información de los objetos y la escena.

5.1 Metodología

Para abordar el problema del reconocimiento de escenas en interiores y la recuperación de escenas, proponemos DeepScenePip (DSP), una cadena de pasos con tres módulos: *centrado en los objetos* y *objetos a escenas*, y *centrado en la escena*, que se centran independientemente en el contenido local y global de la escena, respectivamente. El proceso propuesto tiene dos componentes novedosas. En primer lugar, produce una descripción de la imagen a partir de las etiquetas de los objetos reconocidos para predecir las escenas mediante un enfoque fundamentado en el procesamiento del lenguaje natural. En segundo lugar, se presenta una función de peso que combina la información sobre el objeto y la escena para realizar una predicción global de la misma. A continuación presentamos los detalles de la arquitectura del DSP, y luego introduciremos los métodos de reconocimiento y recuperación de escenas en interiores.

5.1.1 Arquitectura de DeepScenePip (DSP)

En la figura 5 presentamos la arquitectura de DeepScenePip. La arquitectura comienza con una red base seguida de dos módulos: *centrado en los objetos* y *centrado en la escena*. El primero consiste en un detector de objetos y una red clasificadora entrenada de extremo a extremo para generar las etiquetas de los objetos (sus clases) y sus correspondientes características. El módulo se conecta además con el módulo *objetos a escena*, que predice las categorías de la escena a partir de las descripciones de las imágenes vectorizadas determinadas a partir de las etiquetas de los objetos. El módulo *centrado en la escena* amplía la red base con dos capas totalmente conectadas para la tarea de clasificación de escenas. Por último, para obtener una predicción o recuperación global de la escena, los resultados obtenidos de los módulos *centrado en los objetos* y *objetos a escena*, y el módulo *centrado en la escena* se combinan mediante un cálculo de puntuación que difiere para el reconocimiento de la escena y para la recuperación de la misma. A continuación presentamos los distintos componentes de la arquitectura DSP.

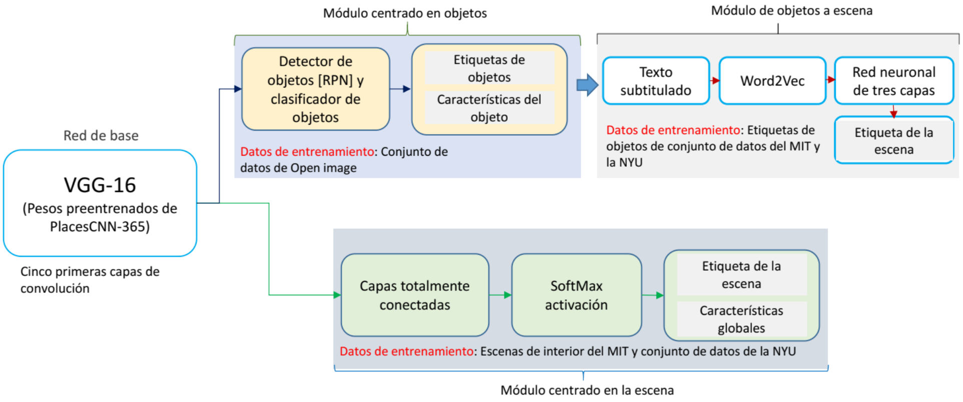

Módulo centrado en objetos
Detector de objetos [RPN] y clasificador de objetos
Etiquetas de objetos
Características del objeto
Datos de entrenamiento: Conjunto de datos de Open image
Módulo de objetos a escena
Texto subtitulado
Word2Vec
Red neuronal de tres capas
Etiqueta de la escena
Datos de entrenamiento: Etiquetas de objetos de conjunto de datos del MIT y la NYU
Red de base
VGG-16
(Pesos preentrenados de PlacesCNN-365)
Cinco primeras capas de convolución
Capas totalmente conectadas
SoftMax activación
Etiqueta de la escena
Características globales
Datos de entrenamiento: Escenas de interior del MIT y conjunto de datos de la NYU
Módulo centrado en la escena

Figura 5: Visión general de la arquitectura de DSP para reconocimiento de escenas.

Red base: La extracción de características correspondientes a los módulos *centrado en los objetos* y *centrado en la escena* implica el uso de una red base compartida, que es una red VGG-16 pre-entrenada con el conjunto de datos Places365 Zhou et al. (2016). Las características correspondientes a la última capa convolucional se utilizan como entradas a las dos redes de destino diferentes (módulos *centrado en los objetos* y *centrado en la escena*) para generar las características de los objetos y globales.

Módulo centrado en los objetos: Dado que las escenas están compuestas por objetos, su identificación es uno de los aspectos más importantes que hay que tener en cuenta durante el reconocimiento de la escena. El objetivo de este módulo es, por lo tanto, la detección de objetos discriminatorios en las imágenes y la extracción de sus características locales para la representación semántica de la escena. El módulo comprende dos redes que se usan en conjunto de extremo a extremo: una para la detección de objetos y otra para la clasificación. Además, proponemos utilizar las etiquetas de los objetos derivadas del módulo como incrustaciones de palabras para entrenar una red neuronal de predicción de escenas, y las características de los objetos para la recuperación de escenas.

Módulo de objetos a escena: El objetivo principal de este módulo es predecir las categorías de las escenas simplemente utilizando las etiquetas de clase de los objetos detectados. Las etiquetas de los objetos extraídas del módulo *centrado en los objetos* pueden tratarse como palabras visuales que describen el contenido de la imagen. Transformamos estas etiquetas en palabras visuales utilizando la incrustación de palabras word2vec (Mikolov et al., 2013). Después, para el entrenamiento, asignamos cada descripción de la imagen a su correspondiente categoría de escena real. La Tabla 7 muestra algunos ejemplos de entrenamiento creados a partir del conjunto de datos MIT-67 en relación con cuatro categorías de escenas diferentes: baño (*bathroom*), dormitorio (*bedroom*), sala de ordenadores (*computer room*) y piscina interior (*pool inside*). Por último, utilizamos las descripciones generadas para crear representaciones vectorizadas mediante el método word2vec, y luego los vectores se dan como entradas a una red neuronal de tres capas para predecir las etiquetas de las escenas.

Módulo centrado en la escena: El reconocimiento de escenas en interiores depende principalmente de la detección de objetos que se encuentran habitualmente en las escenas de interiores. Sin embargo, a veces, debido a la presencia de objetos desordenados, éstos pueden pasar desapercibidos, por lo que no se pueden generar todas las etiquetas de los objetos. Además, debido a la presencia de objetos similares que corresponden a imágenes de escenas de diferente clase, el uso de un enfoque centrado en los objetos puede reducir la precisión de la predicción a medida que aumenta

Cuadro 7: Ejemplos de descripciones generadas por el módulo (*centrado en los objetos*) extraídos del conjunto de datos MIT-67

Escena	**descripciones**	**etiqueta**
Bathroom	Sink,Toilet,Bidet,Tap,Bathroom cabinet,Countertop	1
Bathroom	Toilet,few Bidets,Tap,Bathroom cabinet,Sink,Window,Cabinetry	1
Bedroom	Bed,Nightstand,few Curtains,few Pillows,Lamp	2
Bedroom	few Beds,Table,Chair,Chest of drawers,Desk,Window blind,Window	2
Computer room	Computer monitor,Office building,Desk,Table,many Chairs	9
Computer room	Office building,Laptop,few Desks,Chair,Office supplies	9
Pool inside	Swimming pool,Chair,few Tables,Tree,Plant	30
Pool inside	Swimming pool,Boat,Swimwear,Woman,few Persons	30

el número de clases. Teniendo en cuenta esto, el contexto global de las imágenes de la escena debe ser considerado también para una predicción eficaz. Por lo tanto, creamos una red *centrada en la escena*, que consiste en dos capas totalmente conectadas añadidas a la red base. La red entrenada puede entonces predecir categorías de interiores, y su última capa se utiliza para extraer características como descriptores de la escena.

5.1.2 Método de reconocimiento de escenas

El método de reconocimiento de la escena predice la categoría de una imagen de la escena utilizando la cadena de pasos DSP. Inicializa dos flujos en paralelo. Por un lado, el módulo *centrado en los objetos* detecta y clasifica los objetos de la misma. Después, las etiquetas de los objetos se envían al módulo *objetos a escenas*, que a su vez identifica un conjunto de etiquetas de escena con sus probabilidades asociadas. Por otro lado, el otro flujo predice las etiquetas de la escena con las probabilidades asociadas, utilizando el módulo *centrado en la escena*. Sin embargo, el enfoque *centrado en los objetos* por sí solo puede no producir los resultados deseados cuando varias categorías de escenas comparten objetos comunes. En cambio, las características globales *centradas en la escena* por sí solas podrían ser demasiado genéricas debido a la presencia de diseños similares en diferentes clases de imágenes. Para superar estas limitaciones individuales de cada uno de los módulos y mejorar la precisión de la predicción y recuperación de escenas, combinamos las predicciones de los módulos *centrado en los objetos* y *centrado en la escena* utilizando una función de peso que denominamos combinación ponderada de objetos y escena (WCOS, del inglés *weighted combination of objects and scene*). Finalmente, la escena con el mayor WCOS se considera la escena predicha.

5.1.3 Método de recuperación de escenas

El marco de recuperación de escenas tiene como objetivo la búsqueda de imágenes similares a una imagen de escena interior de entrada. Utilizamos los módulos *centrado en los objetos* y *centrado en la escena* de la arquitectura DSP para extraer las características globales y del objeto de las imágenes de consulta y del conjunto de datos, respectivamente. En concreto, creamos un diccionario de imágenes relativo a cada escena, donde almacenamos las características neuronales de cada objeto detectado junto con su etiqueta predicha. Cada pareja de características de objeto y etiqueta se indexa como *Objeto-ID-número*. Posteriormente, añadimos al diccionario las características globales de la imagen, indexadas como *G-ID*. Si no se detecta ningún objeto en una imagen, el diccionario contendrá únicamente las características globales de la imagen. Para recuperar las imágenes de la escena, proponemos calcular una puntuación para cada imagen del conjunto de datos para la imagen de consulta con la similitud entre las características del objeto y las características globales. En primer lugar, creamos diccionarios para todas las imágenes de escena del conjunto de datos, que se almacenan en una base de datos, y a continuación creamos el diccionario para la imagen de consulta, que llamamos diccionario de consulta. A continuación, para cotejar las similitudes entre la consulta y el diccionario de imágenes, calculamos una función de puntuación denominada como $WCOS_{SRet}$ que combina las similitudes obtenidas entre las características del objeto y las características globales de la imagen. En esencia, nuestro objetivo es recuperar las k imágenes de escenas de interior que obtengan las puntuaciones más altas de $WCOS_{SRet}$.

Sin embargo, uno de los problemas significativos con los que se encuentra un sistema de recuperación es el cambio en el punto de vista de las imágenes dentro de la misma categoría de escena Xie et al. (2020). Esto se debe a que las imágenes de una determinada escena pueden tomarse desde distintos ángulos. En consecuencia, algunos objetos concretos de la escena pueden no aparecer en todas las imágenes. Para solucionar este problema relacionado con el punto de vista, introducimos una nueva técnica para ampliar el diccionario de consulta añadiendo objetos ya detectados en relación con una escena dada. Como resultado, aumenta la posibilidad de recuperar imágenes que corresponden a la misma escena, incluso si la consulta inicial no comparte objetos comunes con las imágenes recuperadas correctamente.

5.2 Experimentos y resultados

Los experimentos y resultados presentados en este trabajo tienen como objetivo medir la eficacia del método DeepSceneNet para dos tareas específicas: el reconocimiento de escenas interiores en los conjuntos de datos MIT-67 Indoor y NYU-V2, y para la recuperación de escenas interiores en Hotel-50K.

Para entrenar la arquitectura DeepScenePip, primero aplicamos el aprendizaje por transferencia a la red base inicializando las capas convolucionales con los pesos de Places-CNN Zhou et al. (2016). A continuación, entrenamos por separado el módulo *centrado en los objetos* con el conjunto de datos Open Image y el *centrado en la escena* con los conjuntos de datos MIT-67 y NYU. Además, para entrenar el RPN del detector de objetos, primero anotamos las imágenes de entrenamiento con coordenadas de cajas delimitadoras y luego creamos minilotes a partir de imágenes individuales.

Evaluamos nuestro enfoque para el reconocimiento de escenas en interiores a través de la métrica de precisión que mide el porcentaje de imágenes cuya etiqueta de escena con mayor puntuación coincide con la etiqueta de categoría verdadera. Para predecir la etiqueta de la escena, primero obtuvimos las 5 primeras predicciones de la escena utilizando los módulos *centrado en los objetos* y *centrado en la escena*, y luego calculamos $WCOS_{SR}$ para todas las categorías predichas. En la Tabla 8, comparamos los resultados que obtuvimos en el conjunto de datos MIT-67 con los reportados en la literatura. Obtuvimos una precisión de 94,5 %, reduciendo la tasa de error en 57,4 % en comparación con la mejor precisión de 87,10 % presentada en López-Cifuentes et al. (2020). En la Tabla 9 mostramos los resultados que alcanzamos en el conjunto de datos NYU, en comparación con los de los métodos del estado del arte. La tasa de precisión alcanzada por el método propuesto es (71,1 %) con una caída de 10.5 % en la tasa de error en comparación con la mayor precisión encontrada en la literatura (67,7 % por MAPNET Li et al. (2019)).

Para evaluar el método propuesto para la recuperación de escenas en interiores, nos centramos en la recuperación de instancias de hoteles utilizando el conjunto de datos Hotels-50k. El objetivo de esta tarea es buscar todas las imágenes del conjunto de datos que representen la misma habitación de hotel que la consulta proporcionada y, a continuación, recuperar las primeras k imágenes más similares. Seguimos el mismo protocolo de evaluación y establecimos los resultados de referencia como los reportados en Stylianou et al. (2019). Medimos el rendimiento del método propuesto calculando la métrica de precision@k con $k = 1, 10, 100$ para la recuperación de instancias de hoteles. La precision@k indica la tasa del número de veces que la etiqueta correcta de verdad está dentro de las primeras k imágenes recuperadas por el método. Además, la evaluación se realiza con dos tipos diferentes de imágenes presentes en el conjunto de datos: (1) Sin oclusión y (2) Con oclusión media. La Tabla 10 muestra los resultados de recuperación que logramos en el conjunto de datos Hotels-50k, en comparación con los reportados en Stylianou et al. (2019).

6 Conclusiones de la Tesis y posibles Trabajos Futuros

Las conclusiones y líneas de trabajo futuro de esta tesis, han sido presentadas en el capítulo 6.

Cuadro 8: Comparación de los resultados con algunos enfoques relevantes en el conjunto de datos MIT-67. Todos los resultados declarados sin citar han sido extraídos de López-Cifuentes et al. (2020).

Métodos	Red de base	Precisión
PlaceNet	Places-CNN	68.24
MOP-CNN	CaffeNet	68.90
CNNaug-SVM	OverFeat	69.00
HybridNet	places-CNN	70.80
URDL+CNNaug	AlexNet	71.90
MPP-FCR2	AlexNet	75.67
DSFL+CNN	AlexNet	76.23
MPP+DSF	AlexNet	80.78
CFV	VGG-19	81.0
CS	VGG-19	82.24
SDO	2xVGG-19	83.98
VSAD	2xVGG-19	86.20
SDO(9 scales)	2xVGG-19	86.76
RGB Branch	ResNet-18	82.68
RGB Branch	ResNet-50	84.40
Semantic Branch	4 Conv	73.43
SASR	RGB Branch + G-RGB-H	87.10
(Rahimzadeh et al., 2021)	Xception	73.60
DSN(nuestro)	VGG-16places	**94.5**

Cuadro 9: Comparación de los resultados con los de algunos métodos relevantes en el conjunto de datos NYU-V2

Métodos	Precisión
Gupta et al. (2013)	45.4
Wang et al. (2016)	63.9
Song, Herranz and Jiang (2017)	66.7
Local + OOR Song, Chen and Jiang (2017)	60.1
MAPNet Li et al. (2019)	66.8
Local+Global+OOR Song, Chen and Jiang (2017)	66.9
MAPNet+Global Li et al. (2019)	67.7
DSN(nuestro)	**74.5**

Cuadro 10: Comparación de los resultados obtenidos en el conjunto de datos Hotels-50k con respecto a los métodos propuestos en Stylianou et al. (2019) en términos de precision@k.

Métodos	**Sin oclusión**			**Con oclusión media**		
	$k = 1$	$k = 10$	$k = 100$	$k = 1$	$k = 10$	$k = 100$
FIXED-OBJECT	0.8	0.9	1.3	0.0	0.0	0.0
FIXED-SCENE	0.2	0.8	2.4	0.1	0.4	1.5
Hotel-A,-I	4.7	9.6	20.0	1.8	4.0	9.4
Hotel -A	8.1	18.4	36.0	3.5	9.2	12.8
Hotel	8.1	17.6	34.8	5.9	14.1	29.9
DSN(nuestro)	**10.1**	**20.6**	**43.8**	**7.8**	**15.6**	**32.2**