Metodología de Agrupación Semántica de Objetos para la Clasificación de Escenas

SEBASTIÁN LÓPEZ FLÓREZ
ID No. 1088015984

FACULTAD DE INGENIERÍAS - UTP
MAESTRÍA EN INGENIERÍA ELÉCTRICA

Risaralda - Pereira
2020

SEBASTIÁN LÓPEZ FLÓREZ

Metodología de Agrupación Semántica de Objetos para la Clasificación de Escenas

Tesis presentada como requisito para aspirar al título de MAESTRÍA EN INGENIERÍA ELÉCTRICA DE LA FACULTAD DE INGENIERÍAS, DE LA UNIVERSIDAD TECNOLÓGICA DE PEREIRA

Director PhD.(c) Luis Hernando Ríos González

agosto
2020

UTP - FACULTAD DE INGENIERÍAS
Universidad Tecnológica de Pereira

Tesis de MAESTRÍA EN EL PROGRAMA DE MAESTRÍA EN INGENIERÍA ELÉCTRICA con el título **Metodología de Agrupación Semántica de Objetos para la Clasificación de Escenas** por Sebastián López Flórez, aprobado por la junta examinadora constituida por los siguientes profesores:

PhD.(c) Luis Hernando Ríos González
Director

PhD.Julián David Echeverry Correa
Evaluador

PhD.Maximiliano Bueno López
Evaluador

Pereira, Julio 2020

DEDICATORIA

A mi madre que siempre me acompaña durante este camino, el gran apoyo de Pedro Julio Ruíz en mi vida académica, muchos de los logros se los debo a cada unas de las personas que influyeron en mi camino profesional, como mi director, los compañeros de trabajo entre otros.

RESUMEN

Uno de los problemas más relevantes y activos en la comunidad científica es el reconocimiento de escenas. Los sistemas de visión por computador, aunque han tenido gran desarrollo en la última década, están enfocados en su mayoría en información de apariencia visual para comprender las escenas, sin tener en cuenta el potencial de información que puede aportar el análisis del contexto de los objetos relevantes en la representación de la escena. Dicha información permitirá que las máquinas perciban relaciones, coincidencias y diversos componentes de una escena de forma similar a una persona. Se propone un sistema de clasificación de escenas a partir del análisis de la semántica de los objetos en una imagen, basado en un detector SSD300 previamente entrenado que captura el estímulo visual del espectador y un módulo de representación densa que genera firmas de contenido semántico de los objetos para su posterior clasificación. Además, se evalúa una arquitectura global en la base de datos propia comparando el modelo basado en la semántica de los objetos. El modelo propio logra resultados competitivos y consistentes a través de múltiples métricas de evaluación demostramos la efectividad del enfoque sugerido en dos conjuntos de datos. En comparación con los enfoques más modernos, se presentó una perspectiva más robusta a la abstracción y generalización de escenas de interiores cuyos resultados, logran una exactitud del 99% con modelo propio, en comparación de un 87% obtenido por el modelo global del estado del arte. Por tanto se mejora la clasificación de escenas en los ambientes interiores seleccionados.

Palabras–clave: Reconocimiento de escena, codificación de características de parche, detección de región discriminativa, redes neuronales convolucionales, aprendizaje profundo.

ABSTRACT

The scene recognition is one of the most relevant and active problems in the scientific community. Computer vision systems have had great development in the last decade, but are mostly focused on information of visual appearance to understand scenes, without taking into account the information potential that the analysis of the context of objects can provide. relevant in the representation of the scene. This information will allow machines to perceive relationships, coincidences and various components of a scene in a similar way to a person. A scene classification system is proposed from the analysis of the semantics of the objects in an image, based on a previously trained SSD300 detector that captures the visual stimulus of the viewer and a dense representation module that generates signatures of semantic content of the images. objects for further classification. Furthermore, a global architecture is evaluated in its own database by comparing the model based on the semantics of the objects. The own methodology achieves competitive and consistent results through multiple evaluation metrics. We demonstrate the effectiveness of the suggested approach on two data sets. Compared with the more modern approaches, a more robust perspective was presented to the abstraction and generalization of interior scenes, the results of which achieve an accuracy of 99 % with our own model, compared to 87 % obtained by the global model of the state of the art. Therefore, the classification of scenes in the selected interior environments is improved.

Key-words: Scene Recognition, Patch Feature Encoding, Spatial Layout Pattern Learning, Discriminative Region Detection,Convolutional Neural Networks, Deep Learning.

LISTA DE FIGURAS

CONTENIDO

1 INTRODUCCIÓN

Lograr que una máquina pueda comprender el contenido y significado categórico de una escena, a partir de los objetos reconocidos en una imagen es una tarea difícil ya que estos sistemas no cuentan con la capacidad de comprensión de la información visual percibida; situación contraria sucede con un ser humano, el cual, para interpretar una escena a través de la observación, requiere de un corto periodo de tiempo para reconocerla completamente. En los humanos, esto es posible porque a través del tiempo su interacción con el entorno refuerza el proceso cognitivo de los objetos, escenas y ambientes, con los que interactúa durante su desarrollo [2][3]. La cognición no sólo permite describir las características visibles del contexto, sino también razonar sobre las correlaciones existentes entre los diferentes objetos de una escena y su importancia semántica, proporcionando sentido lógico al proceso de reconocimiento. Para lograr que esta tarea sea realizada por una máquina es indispensable empoderarla con la capacidad de comprender información visual circundante [4][5][6][7]. A medida que las máquinas comprenden la forma contextual y semántica de las escenas podrán tener mayor autonomía y proporcionar ayudas más significativas en labores de salud, educación, navegación autónoma[4].

Existen gran número de investigaciones alrededor de la comprensión general de la escena, estos enfoques se basan en características de bajo nivel que consideran detalles menores de la imagen como líneas o puntos [7][8][9][10]. En [8] y [10] combinaron una serie de características visuales de bajo nivel incluyendo los coeficientes de color y borde, alimentando un clasificador basado en máquinas de soporte vectorial (SVM) para el reconocimiento de escenas[1]. Svetlana, Cordelia Schmid y Jean [11], proponen el método de correspondencia de pirámides espaciales (SPM) para el reconocimiento de escenas basado en el popular modelo de bolsa de palabras visuales (BoVW) [12] en el que se emplean características SIFT (SIFT-Scale Invariant Feature Transform) [13]. Otros operadores de funciones de bajo nivel, como el Patrón binario local (LBP) [14], el histograma de bordes orientados (HOG) [15] y el descriptor de auto-similitud (SSIM) [16], se utilizaron con éxito como métodos de reconocimiento de escenas. Sin embargo, estas características de apariencia de bajo nivel extraídas de píxeles o parches de imágenes locales estaban lejos de ser suficientes para clasificar con éxito las escenas, ya que poseen capacidad limitada para describir una imagen debido a la alta complejidad que las rodea.

En los últimos años, con los avances en el aprendizaje profundo, las CNN (redes neuronales convolucionales)[17] por sus características han mostrado ser más efectivas, que las técnicas clásicas “hechas a mano” en el proceso de clasificación y reconocimiento de escenas. Como uno de los modelos profundos más prominentes, las redes neuronales convolucionales (CNN) en ImageNet [18], han mostrado una prometedora capacidad de

reconocimiento de objetos. Después de múltiples operaciones convolucionales y transformaciones no lineales en lo profundo de su arquitectura, las características convolucionales de salida pueden comprender más información abstracta y atributos visuales completos que las características convencionales "hechas a mano". Con el aumento de la profundidad de las CNN, el poder discriminativo puede ser mejorado [19].

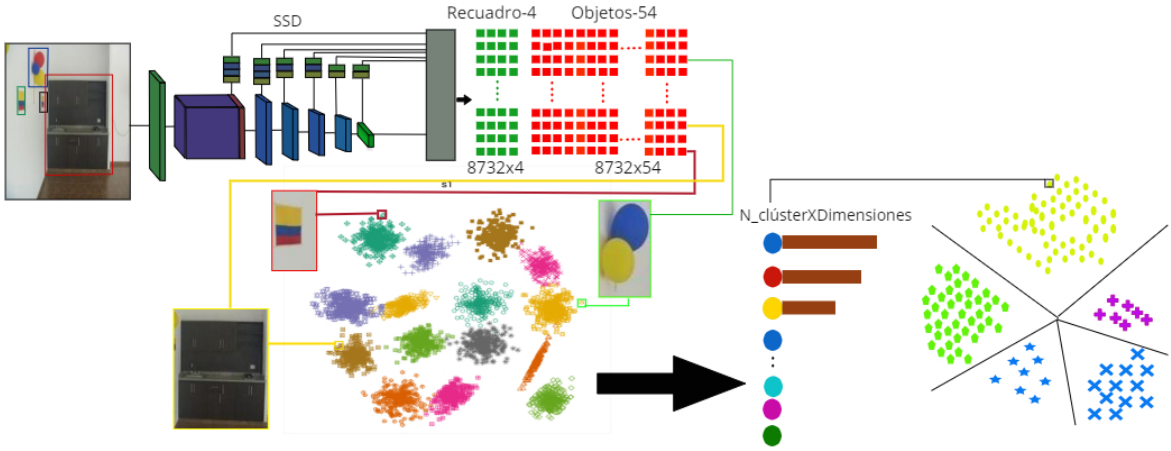
Para comprender y clasificar escenas de manera más eficiente es necesario identificar la presencia de objetos y la ocurrencia de los mismos dentro de las escenas [5] [7][20]. Por ejemplo, si una persona está en una oficina, no solo es necesario saber en qué oficina se encuentra, sino también, sería interesante conocer de quién es la oficina, además del conocimiento contextual útil que permita un alto grado de comprensión. Este es un desafío de muy alta complejidad, ya que las escenas comparten objetos comunes, vistas dinámicas entre otras. Debido a la característica de grandes variaciones intraclase y ambigüedades entre clases, es difícil representar con precisión la semántica de las imágenes [7]. Además, el sistema de visión artificial necesita inferir las propiedades de la escena usando la menor cantidad posible de imágenes de entrenamiento, aunque los sistemas actualmente exitosos se entrenan usando millones de imágenes de sescena recopiladas en distintas bases de datos [21] siendo esta la forma menos adecuada para atacar el problema, dado que en condiciones reales los escenarios a clasificar no son tan comunes . Para lograr el nivel deseado de comprensión de los atributos sutiles en las imágenes, se requiere explorar la información semántica de alto nivel de las escenas.

En este trabajo se propone un algoritmo para extraer información discriminante y atributos visuales a partir de parches (regiones de escena visualmente destacadas) que componen una escena como se muestra en la Figura 1.1. A través del modelo de detección de objetos SSD (Detector múltiples cuadros de disparo único) [22]. Las firmas globales aprendidas combinan conceptos visuales generales compartidos por diferentes clases a partir de la técnica VLAD (Vector de descriptores localmente agregados) [23] logrando una representación sólida de la imagen para su posterior clasificación.

En lugar de usar una sola característica convolucional para representar una imagen, se propone aprovechar la detección de objetos que describen y detectan los parches más relevantes, los cuales fueron detectados y etiquetados de forma manual a partir del conocimiento contextual y experticia del ser humano, para incorporar conceptos visuales más efectivos [24]. Por lo tanto, múltiples tipos de descripción de parches pueden ser capaces de complementar información crítica faltante dentro de la clase [19]. Las principales contribuciones de este documento se pueden resumir como: (1) Propuesta de un nuevo algoritmo que combina técnicas establecidas en el estado del arte para aprender una representación efectiva en el reconocimiento de escenas. (2) Extracción de parches locales aprovechando el poder de las arquitecturas de detección convolucional para hacer una descripción semántica robusta, demostrando que estos detalles son esenciales

para el reconocimiento de escenas. (3) Implementar un modelo que permite agrupar las características descritas para cada parche, aprendiendo combinaciones de grupos, basado en objetos para producir firmas globales. (4) Otra novedad es generar una serie de transformaciones a las imágenes haciendo una descripción densa de parches, aprendiendo combinaciones semánticas más ricas en el espacio semántico. (5) Realizar experimentos en 3 conjuntos de datos, dos estándares y uno propio, validando el método propuesto para la detección de escenas a partir de la semántica de los objetos.

Figura 1.1 – Modelo de representación de escenas a partir de la semántica de los objetos. Primero se etiquetan los objetos de forma manual dotando a un detector de regiones discriminantes del estímulo natural del espectador que etiqueta con base a su conocimiento innato. Luego, se extraen vectores convolucionales de características del objeto detectado, proyectándolos en el espacio semántico donde serán agrupados para obtener representaciones locales, referente a las composición de los objetos en una imagen las cuales permitirán la clasificación.



1.1 ENUNCIADO DEL PROBLEMA

¿Es posible mejorar el rendimiento de un modelo para la clasificación de imágenes a partir de la relación semántica entre escenas y objetos?

1.2 PREMISA DE HIPÓTESIS

Un sistema de clasificación de escenas, en un modelo de representación global ¿Puede ser mejorado aprovechando la información local de los objetos y la relación semántica entre escenas y objetos?

1.3 JUSTIFICACIÓN

Las aplicaciones en ámbitos cotidianos ya incluyen diferentes tecnologías cuya base es la visión por computadora y la robótica [4][7][25]. Las máquinas deben comprender conceptos espaciales abstractos y actuar en consecuencia[26][18][27] dándole una interacción humano-robot (HRI). Por ejemplo, si se le pide a una máquina reconocer un entorno específico de ejecución de tal forma que pueda interactuar con él, la comprensión de dicho lugar debe coincidir con la contextualización humana [4][24], una manera de alcanzarlo puede ser, la recolección de imágenes etiquetadas que logren capturar la semántica de las escenas de la misma forma natural, como lo hacen los seres humanos; el lograr simular esta condición natural en una máquina requiere cumplir ciertos parámetros de condicionamiento como lo son la resolución de la cámara, sus poses, las condiciones de iluminación, la proximidad entre la cámara y los elementos de la escena, el desenfoque y el etiquetado de las imágenes. Una de las técnicas que presenta resultados significativos en la detección de escenas se basa en CNN que se enfoca en construir un sistema que sea lo suficientemente profundo que pueda incluir todas las posibles variaciones que poseen las diferentes categorías de escenas. La configuración de red neuronal profunda CNN, requiere de un proceso de entrenamiento y sintonización, que le permita tener autonomía al momento de clasificar permitiendo una amplia generalización ante posibles variaciones de los datos. Por ejemplo, suponiendo que se describiera una imagen de una escena en términos de los objetos, la CNN proporciona un enfoque efectivo que mejora la capacidad de generalización de la representación, aun cuando los objetos proporcionen un alto nivel de abstracción [4][19]. Sólo es indispensable que se cumplan ciertas condiciones relacionadas con objetos únicos, dentro de las escenas y que estos permitan una descripción semántica válida.

Se propone describir una imagen de una escena a través de la recopilación sistemática de sus objetos, donde cada objeto se etiqueta de forma aislada. Se espera que conjuntamente la apariencia visual de los objetos y el contexto de alto nivel en las imágenes de la escena, permitan una comprensión más profunda de la imagen [4]. Si bien los objetos proporcionan información valiosa para describir escenas, es común encontrar estructuras en escenarios de representación compleja que comparten objetos comunes siendo este un problema para los sistemas de reconocimiento, ya que causan confusión al momento de crear una representación única, limitando la capacidad de diferenciar entre escenas, en ese caso las imágenes que comparten objetos comunes entre sí están más semánticamente relacionados que con otras imágenes. La estructura semántica en imágenes de escena con representación variable, posibilita descubrir subgrupos y sus relaciones contextuales permitiendo aprender modelos más discriminantes mapeados en un espacio semántico de parches y obteniendo un vector robusto el cual es usado para codificar la información relevante subyacente en

una imagen perteneciente a una escena.

Para la implementación de esta propuesta se parte del análisis de trabajos previos [20][28][7] basados en la detección de objetos para una descripción local de la escena utilizando arquitecturas CNN, VLAD, FISHER[29], las cuales presentan buen desempeño. Sin embargo, su principal falencia radica en poseer pocos objetos relacionados a la clase, como también una descripción basada únicamente en parches sin tener en cuenta la composición densa de los mismos para un mejor análisis en el espacio semántico. El enfoque propuesto, parte de la selección de objetos discriminantes de las escenas y su descripción semántica a partir de un detector de objetos, los cuales permiten rastrear los parches existentes en una imagen logrando obtener una representación local, que posteriormente se utilizara para formar la bolsa semántica junto con una descripción compacta utilizada para clasificar las escenas.

1.4 OBJETIVOS

1.4.1 Objetivo General

Desarrollar una metodología de agrupación semántica de objetos para clasificación de escenas.

1.4.2 Objetivos Específicos

- Seleccionar y etiquetar de forma manual los parches de los objetos cuya relación semántica permite de manera natural la identificación de las imágenes.
- Entrenar el modelo SSD300 con la base de imágenes propia.
- Obtener los puntajes de acierto de las imágenes a clasificar utilizando diferentes métodos.
- Validar el método implementado y comprobar su eficiencia.

2 CONCEPTOS GENERALES Y REVISIÓN DE LA LITERATURA

Un sistema de clasificación de imágenes se puede dividir en las siguientes etapas, extracción de características, etiquetado de la base de datos, reconocimiento de los objetos, fusión de las características convolucionales de los objetos para cada imagen, clasificación de los vectores resultantes del método de fusión implementado.

2.1 CONCEPTOS GENERALES

2.1.1 Etiquetado de la base de datos

Para el etiquetado de objetos existen en la literatura diferentes herramientas, entre las que se puede mencionar:

- LabelIMG
- VGG Image Annotation Tool
- Supervise.ly
- Labelbox

2.1.1.1 LabelImg

Es una de las herramientas más populares para el etiquetado de objetos en imágenes, esta herramienta posee gran respaldo en la comunidad GitHub, ya que el repositorio cuenta con un flujo constante de contribuciones y aportes que proporcionan confiabilidad y robustez para el uso [30]. La versatilidad del software permite usarla en cualquier sistema operativo Ubuntu, Linux o Mac, además esta herramienta proporciona licencias gratuitas que permiten el trabajo de investigación con su interfaz gráfica desarrollada en Qt. La herramienta posee dos formatos populares en el estado del arte Pascal VOC para guardar las anotaciones en forma de archivos XML y el formato YOLO (*You Only Look Once*) Detection [31][32].

2.1.1.2 VGG Image Annotation Tool

Es una herramienta de anotación manual de código abierto, fácil de usar e independiente. Se puede utilizar para anotaciones de imagen, audio y video. Siendo un software independiente, no depende de ninguna biblioteca externa y puede ejecutarse en cualquier navegador web sin ninguna instalación o configuración. Ofrece una gran cantidad de características, que comprende variedad de herramientas, líneas de soporte, puntos, círculos, polígonos y eclipses. Permite agregar objetos, introducir atributos de imagenes o etiquetas. Todas las descripciones están contenidas en un archivo de notación de objetos JavaScript (JSON) o un archivo de valores separados por comas (CSV). Las anotaciones se pueden descargar [32].

2.1.1.3 Supervise.ly

Supervise.ly es una de las mejores plataformas basadas en la web, donde no solo se puede acceder a una interfaz de anotación avanzada, sino que utiliza Python SDK para importar complementos para formatos de datos personalizados. El software ofrece una gran cantidad de opciones para la gestión de proyectos en diferentes niveles, como equipos, espacios de trabajo y conjuntos de datos. También ofrece muchas opciones para la gestión de anotadores [32].

2.1.1.4 Labelbox

Es una de las herramientas más populares de etiquetado de datos. Fue creada en 2018. Ofrece una versión comunitaria gratuita y una versión empresarial. La versión gratuita está limitada a 5000 imágenes. Labelbox también se compone de una interfaz fácil de usar. Hay varias opciones para monitorear el rendimiento, los mecanismos de control de calidad, etc. Entre sus atributos cuenta con una opción de coloración superpíxel, la cual es una característica recientemente agregada para el pincel de segmentación semántica. Todas las anotaciones se guardan en formato de archivo JSON o CSV [32].

2.1.2 Detección de escenas

En esta sección, se describe la existencia de métodos utilizados para la clasificación de escenas. En [33] se utilizaron detectores de objetos como características para formar una representación de banco de objetos para ayudar a la clasificación de escenas. En Yang et al. [34] usaron puntos de interés para la detección, creando vectores de características. Bosch et al.[35] utiliza vocabulario visual como características para entrenar una máquina de vectores de soporte (SVM). [36], [12] y [35] describen una representación de bolsa de

palabras visuales BoVW, donde las características locales se cuantifican en una sola palabra y se utiliza un histograma global para resumir el contenido visual. Las representaciones de imágenes más densas y los vectores de Fisher comprimidos [37] [38], se proponen como una representación más compacta que la BoVW. Además, Se introdujo el método de codificación de asignación suave [38] para reducir la pérdida de información durante la cuantificación. Sparse coding [39] y Locality-constrained linear coding [40], propuso explotar la sparsity and locality para el aprendizaje del diccionario y codificación de características. Métodos de codificación de alta dimensionalidad como el vector Fisher [29], VLAD y el Super Vector[39], fueron presentados para conservar información de orden superior y mejorar el reconocimiento. El enfoque que se propone se fundamenta principalmente en el método de codificación de VLAD. Al igual que diferentes tipos de CNN, arquitecturas como (AlexNet [18], VGG [41], GoogleNet [42]) fueron utilizados en clasificación de escenas recientemente. Estas CNN lograron rendimientos de vanguardia ($\approx 56\%$) con conjunto de datos como Places365 [43]. En Khan et al. [44], integraron Places-VGG con características espectrales para mejorar la clasificación de la escena. Sin embargo, los conjuntos de datos que usaron no son apropiados para aplicaciones reales, ya que las imágenes presentadas no están un contexto robótica aplicado [24].

2.1.3 Detección de objetos

La red convolucional regional (R-CNN[45]) es uno de los primeros enfoques exitosos para combinar CNN y cuadros delimitadores. El método se divide en un componente para localizar objetos y uno para clasificar cada cuadro. Desafortunadamente, R-CNN es computacionalmente costosa, ya que procesa repetidamente los mismos píxeles cada vez que aparecen en diferentes regiones superpuestas. Fast R-CNN [46] solucionó este defecto, introduciendo primero la imagen completa a través de un extractor de características, disminuyendo de esta forma el cálculo en el conjunto de cuadros delimitadores. Este conjunto de ideas termina plasmado en Faster R-CNN , donde las propuestas regionales se generan eficientemente utilizando una red totalmente convolucional. Si bien Faster R-CNN puede procesar varias imágenes por segundo, generalmente es demasiado lento para la mayoría de las aplicaciones móviles o de robótica que exigen un rendimiento en tiempo real en plataformas con restricciones de cómputo. Esto ha motivado una serie de modelos de detección de objetos, como la SSD y YOLO [31] que apuntan a una alta calidad del proceso de detección a velocidades casi en tiempo real.En el enfoque propuesto, el proceso de detección de objetos se basa en experimentos completos y en las compensaciones de velocidad / precisión de detección de objetos realizadas por Huang y col. [47], el estudio concluye que SSD tiene una de las mejores compensaciones entre velocidad y precisión. La SSD surge como una posible alternativa de base sólida en el proyecto en sus etapas iniciales.

2.1.4 Detección de escenas a partir de objetos

El análisis de correlación de objetos busca modelar las relaciones entre la distribución de diversos objetos y las categorías de las escenas. Las primeras exploraciones de la correlación de objetos se basan en modelos de temáticas en los que el reconocimiento de objetos es un requisito previo. Los modelos de temas típicos incluyen spatially coherent latent [48] y modelos sensibles al contexto [49], los cuales son modelos reconfigurables [50]. Al caso de distribución de objetos entre diferentes escenas, se proponen algunos modelos que utilizan patrones de ocurrencia de categorías, como el modelo de contexto basado en la variedad semántica [51] , MetaObject-CNN [52], Kernel Co-occurrence Noise Filter (KCNF) [53] y Semantic Descriptor with Objectness (SDO) [20].

La detección de objetos discriminantes enmarca un prospecto prometedor para seleccionar de forma autónoma algunos objetos cruciales para el reconocimiento de escenas. El banco de objetos [33] y los modelos deformables basados en piezas [54] recurren a algoritmos de detección de objetos para obtener regiones discriminatorias, mientras que otros métodos intentan identificar las regiones discriminatorias de un gran número de parches de la imagen, como la agrupación discriminatoria no supervisada [55], las curvas de rango de entropía [56] y la estimación de densidad [57]. Para suprimir características ruidosas y generar el aprendizaje de importantes regiones de agrupación espacial (ISPR), utilizan filtros parciales para retener la respuesta de regiones importantes en [58]. En [59] se utiliza aprendizaje de características discriminativas y compatibles (DSFL), para determinar la varianza interna de clases y la similitud entre clases. Al alimentar a las CNN pre-entrenadas con imágenes de la escena mejorada, las características visualmente sensibles obtenidas han demostrado ser efectivas para el reconocimiento de escenas [60], [24].

CNN-DL [61] intenta incorporar codificación dispersa en las redes personalizadas convolucionales de extremo a extremo para transformar las características convolucionales y hacer el reconocimiento de escenas. Con el predominio del aprendizaje profundo, los conceptos visuales han sido explorados por las activaciones de CNNs. La activación de clase mapas (CAM) propuesto en [62] ha revelado que las CNN identifican la etiqueta de clase por las regiones de imagen discriminativas. Recientemente, el modelo CNN visualmente sensible (VS-CNN) [63] es construido para extraer características profundas de las imágenes mejoradas por las regiones visualmente sensibles detectadas. Del mismo modo, una especie de patrón de coincidencia de todos los objetos en las escenas [20] se propuso seleccionar algunos objetos representativos para hacer en la imagen representaciones más discriminatorias. Bajo esta premisa se plantea combinar distintos métodos que describan, detecten y representen, información semántica suministrada por las características convolucionales.

3 METODOLOGÍA

En este capítulo se describirá el enfoque propuesto, el cual permite la clasificación de escenas a partir de la extracción y posterior reconocimiento de objetos en imágenes complejas con la menor información posible. La base de imágenes utilizada consiste en una serie de escenarios desafiantes, que contienen imágenes con distintos aspectos comunes y difíciles de interpretar por un sistema convencional. La fortaleza del método consiste, en el reconocimiento específico de parches, que contienen diversos objetos en primer plano, de naturalidad oscuros y de poco contexto en el fondo, los procesa y convierte en una representación espacial rica de información semántica para la clasificación de las escenas.

El desarrollo del capítulo se inicia con una descripción del etiquetado de objetos, tanto del conjunto de datos propio, como de la base de datos del estado del arte. Luego se explicará el modelo de detección de disparo único (SSD: Single Shot multibox Detector) [64], el cual es un algoritmo popular para la detección de objetos y uno de los mejores por su alta velocidad y precisión [65]. Seguidamente, se explicará el concepto y la forma de construcción de la bolsa semántica utilizando el descriptor VLAD [66]. Posteriormente, para explicar la estructura propuesta de clasificación de escenas con objetos [20], se describe la interacción de cada una de las fases que la componen, cómo el SSD, los métodos de regularización y reducción de dimensionalidad para el VLAD [23] y su posterior clasificación.

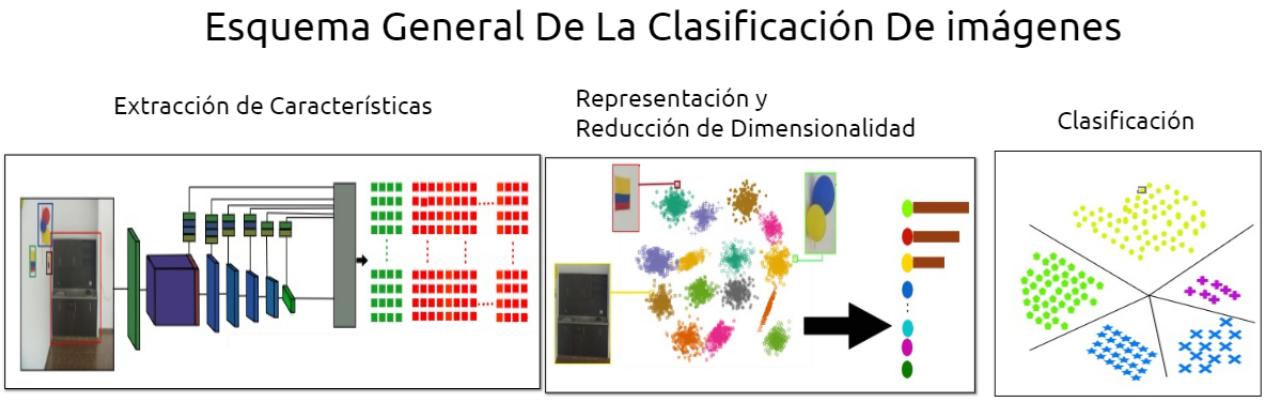
3.1 DISEÑO

En el proceso de diseño, las imágenes se describen con el mayor número de características que las componen; estas características pueden ser locales, semi-locales o globales. La descripción debe ser una representación compacta; una vez ingresada al clasificador dicha representación, se evaluarán las similitudes capturadas por el modelo, obteniendo como resultado la clasificación de la escena a la cual pertenece. La Figura 3.1 muestra el esquema general de la clasificación de imágenes.

3.1.1 Extracción de Características

Para analizar el contenido de las imágenes es indispensable la extracción de aquellas características que se encuentran inmersas en éstas, las cuales pueden ser puntos, bordes u objetos discriminantes. Este paso es útil para todas las aplicaciones, tales como la recuperación de imágenes, la clasificación, el reconocimiento de objetos o la comprensión de escenas. Las características pueden ser globales, locales o semi-locales. En esta fase del

Figura 3.1 – Esquema del modelo a implementar



proyecto, las características que se encuentran en ubicaciones específicas de las imágenes, se pueden seleccionar de tal forma que permitan definir un espacio de la imagen como un objeto discriminante, el cual se denominará parche. [67],[68],[20]. Dicho parche se selecciona utilizando la herramienta labelimg.

3.1.2 Representación de la Imagen

Consiste en obtener una representación semántica global de la imagen, a partir de la información obtenida de los parches que la componen, donde se evalúa en el espacio semántico, las similitudes de los vectores de características para formar una palabra visual que posteriormente, conformará el vector global representativo para su clasificación [69],[20].

3.1.3 Clasificación de la imagen

La clasificación de imágenes, hace referencia a la tarea de extraer información a partir de una representación semántica de los objetos como una firma única para cada imagen que permite el proceso de clasificación [69],[20].

3.2 CONJUNTOS DE DATOS

Los datos son el insumo central para entrenar, probar y validar cualquier tipo de tarea de aprendizaje automático [20], especialmente en el aprendizaje supervisado, donde es necesario que los datos etiquetados tanto de detección de objetos como de escenas, determinen el aprendizaje del algoritmo y predigan el resultado. Los datos de

entrenamiento son el factor principal que influye en el comportamiento y el rendimiento del modelo para datos nunca vistos por el sistema. La etapa de procesamiento consta de:

- La recopilación de los datos de la base propia (la cual se compone de imágenes de interiores), al igual que el conjunto del estado del arte, (la cual incluye imágenes de deportes).
- La selección, anotación y formación de los subconjuntos de entrenamiento y pruebas.

Esta sección describe los métodos utilizados para la recopilación de datos, anotaciónes y preprocesamiento para lograr el objetivo propuesto.

3.2.1 Datos Propios

Los datos utilizados en la estructura propuesta se recopilaron de forma manual. La primera etapa de la recolección de datos consistió en la conformación de la base de imágenes propia al tomar una serie de fotos en las instalaciones de la universidad, la cual incluyó escenas de pasillos y laboratorios; además se tomaron fotos de ambientes interiores domésticos que incluyeron salas y una cocina. Todas las escenas de la base propia, cuentan con imágenes de escenarios complejos, donde los objetos están sometidos a diferentes condiciones como cambios de iluminación, perspectiva y tamaño.

En total la base de imágenes propia consta de 625 imágenes, las cuales se dividen en 5 clases: Clase 1- Cocinas, Clase 2- Sala1, Clase 3- Sala2, Clase 4- Pasillo1, Clase 5- Pasillo 2. Cada clase tiene un total de 125 imágenes. Todas las imágenes están en formato jpg. Para cada clase se toman de las 125 imágenes, 100 imágenes para el conjunto de entrenamiento y 25 imágenes para validación. La Figura 3.2 muestra algunas de las imágenes de la base de imágenes propia.

3.2.1.1 Etiquetado de los objetos

Previo al proceso de detección de los objetos por la SSD, el procedimiento de etiquetado requiere anotar y recolectar todos los parches que son relevantes para la clase, este proceso se hace de forma manual donde el ser humano hace un análisis profundo a partir del concepto relación-entorno, sobre la selección de los diferentes tipos de objetos que se consideran relevantes por su peso semántico, para asociarlos a las diferentes clases de imágenes propuestas [24].

Durante el procedimiento de etiquetado de los objetos es importante tener en cuenta algunos aspectos que influirán en la detección correcta del parche:

Figura 3.2 – Escenas conjunto de datos propios

- Seleccionar aquellos objetos que se consideren relevantes por su relación con las diferentes clases a reconocer. Relación de peso semántico.
- Verificar las anotaciones de cada objeto según la clase para evitar repetir información de etiquetas ya suministradas.
- Verificar que el objeto se encuentra dentro del rectángulo propuesto.
- Organizar el formato de anotaciones estándar para que coincidan con los requisitos de las entradas al detector de objetos SSD.

De la base de imágenes propia se extrajeron 1884 cuadros del conjunto de imágenes de entrenamiento y validación. En cada cuadro capturado había uno o varios objetos. El conjunto de datos contiene 51 clases de objetos de interiores mostrados en la Figura 3.3,

3.2.2 Conjuntos estandarizados de imágenes

En el estado del arte existen muchos conjuntos de datos de imágenes disponibles [11],[70],[71],[72],[73], los cuales contienen cientos de miles de imágenes que sirven para realizar pruebas de clasificación, detección y segmentación. Por lo tanto, se pudo aplicar el modelo implementado, utilizando otros conjuntos de imágenes disponibles en la red para probar la eficiencia del modelo. De cada base de imágenes seleccionada, se eligen dos bases de datos, una se encarga de evaluar el modelo de detección de objetos y la otra del sistema de clasificación de las escenas. Es recomendable la elección de una base simple y

Figura 3.3 – Algunos objetos etiquetados

versátil para etiquetar ya que anotar manualmente todos los objetos y sus instancias con las etiquetas de clase correspondientes es un desafío de gran magnitud.

A continuación se hace una breve descripción de estas bases de imágenes.

3.2.2.1 Pascal

Los conjuntos de datos de Clasificación de objetos visuales PASCAL (PASCAL VOC) se proporcionaron como parte del desafío PASCAL Visual Object Classes de 2005 a 2012. Los conjuntos de datos VOC2007 y VOC2012 con anotaciones de recuadro delimitador se utilizaron para entrenar y evaluar la red SSD original. Por lo tanto, estos conjuntos de datos se utilizarán en la evaluación del modelo SSD-300 en pytorch. PASCAL VOC2007 y VOC2012 contienen 32,494 imágenes con anotaciones para veinte clases diferentes (aire, avión, bicicleta, pájaro, barco, botella, autobús, coche, gato, silla, vaca, mesa de comedor, perro, caballo, moto, persona, planta en maceta, oveja, sofá, tren, monitor de TV) [73].

3.2.2.2 UIUC Sports-8

El conjunto de datos UIUC Sports-8[74], contiene dos características importantes que son interesantes a tratar en el modelo implementado: primero, el tamaño de los objetos relevantes en las imágenes, permiten el etiquetado de éstas sin mayor dificultad; segundo, escenas con objetos cuya semántica es pobre de contexto, induce a la selección de parches pocos discriminantes en la clase, situación que exige al modelo a generar resultados a partir

de información reducida. Este conjunto de datos contiene 1572 imágenes en color en 8 categorías diferentes, las cuales son remo (250 imágenes), bádminton (200 imágenes), polo (182 imágenes), bochas (137 imágenes), snowboard (190 imágenes), croquet (236 imágenes), vela (190 imágenes), y escalada (194 imágenes). Estas imágenes tienen altas resoluciones (de 800 x 600 a miles de píxeles por dimensión). Siguiendo el protocolo definido en [74], se muestrean 70 imágenes al azar para entrenamiento y 60 imágenes restantes se muestrean aleatoriamente para probar en cada categoría. El muestreo múltiple también es necesario ya que la base presenta desbalanceo.

Figura 3.4 – Escenas base de deportes UIUC Sport-8

Fuente: LI, Li-Jia.[74]

Durante la etapa de selección de objetos relevantes para las clases, se tuvo que tratar con algunos desafíos, como objetos de pequeño tamaño, similitud entre clases y parches pocos discriminativos para algunas imágenes. Situaciones que predecían de manera anticipada que los resultados de clasificación podrían ser pobres.

Se extrajeron 3120 cuadros del conjunto entrenamiento y prueba de las imágenes de esta base de imágenes. En cada cuadro capturado había uno o varios objetos. El conjunto de datos contiene 18 clases de objetos mostrados en la Figura 3.5

3.2.3 Anotación de datos

El proceso de etiquetado produce anotaciones necesarias para alimentar el modelo de aprendizaje automático supervisado. Es decir, en la detección de objetos, una anotación es un proceso de localización del parche dentro de un rectángulo, donde el límite del mismo lo contendrá generando un rotulo específico para cada parche.

Figura 3.5 – Objetos deportes UIUC Sport-8

Fuente: LI, Li-Jia.[74]

3.2.3.1 LabelImg

El procedimiento de etiquetado se hace a través del programa LabelImg, el cual es una herramienta de anotación gráfica de imágenes que permite ubicar el rectángulo donde se encuentra el objeto como lo muestra la Figura 3.6. Las anotaciones se guardan como archivos XML en formato PASCAL VOC, con relación a las otras aplicaciones que existen en el mercado esta posee gran respaldo en la comunidad GitHub[30], ya que el repositorio cuenta con un flujo constante de contribuciones, aportes que proporcionan confiabilidad. Además, es una herramienta rápida que brinda una forma fácil de anotar [75].

El primer campo, <folder>, <filename>, corresponde a la ubicación de la imagen. El campo, <pach>, es la ruta del archivo. En <object> (derecha) hace referencia a un único cuadro delimitador, existiendo tantas etiquetas <object> como objetos existan en la imagen. Dentro de la etiqueta se ven dos campos principales. Uno de ellos es <name>, donde estará escrito el nombre de la clase del objeto.El otro es <bndbox>(bounding box), donde se sitúan los puntos exactos de los vértices para un rectángulo en concreto esto se pude ver en la Figura 3.7 .

3.2.4 Aumento de datos de la bolsa semántica

Dado que el conjunto inicial de la base de entrenamiento es de 500 imágenes, es importante tener una bolsa semántica rica, que permita tener parches densamente muestreados que ayuden a preservar mayor información contextual local. Es probable que múltiples escalas incorporen más pistas y relaciones entre componentes [35], para lo cual

Figura 3.6 – Etiquetado de imágenes propias

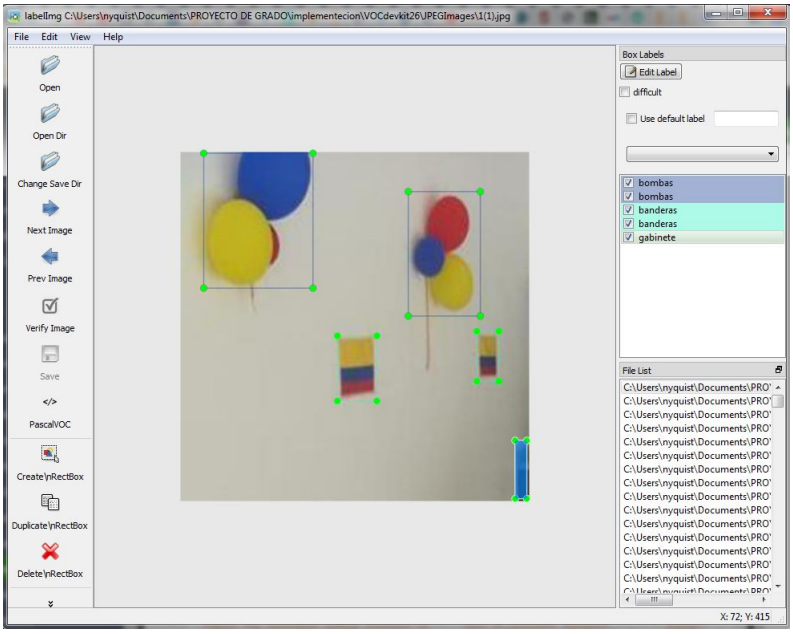


Figura 3.7 – Un ejemplo de archivo XML de detección de objetos en una imagen.

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<annotation>
   <folder>JPEGImages</folder>
   <filename>1(1).jpg</filename>
  ▼<path>
     C:\Users\lhgonza\Desktop\sebastianjunio19\sebastian2\a-PyTorch-Tutorial-to-Object-Detection\VOCdevkit26\JPEGImages\1(1).jpg
   </path>
  ▼<source>
     <database>Unknown</database>
   </source>
  ▼<size>
     <width>300</width>
     <height>300</height>
     <depth>3</depth>
   </size>
   <segmented>0</segmented>
  ▼<object>
     <name>bombas</name>
     <pose>Unspecified</pose>
     <truncated>1</truncated>
     <difficult>0</difficult>
    ▼<bndbox>
       <xmin>20</xmin>
       <ymin>1</ymin>
       <xmax>114</xmax>
       <ymax>117</ymax>
     </bndbox>
   </object>
```

es fundamental generar un aumento de la base de datos que incluya imágenes con cambios en la rotación, color y escala. Dicho aumento de datos permite obtener una nueva base de 5000 imágenes el cual conformara el nuevo espacio de características de los parches con una mayor información semántica.

3.3 MODELO DE REPRESENTACIÓN Y CLASIFICACIÓN

En esta sección, se describen los detalles de diseño e implementación del enfoque propuesto.

El enfoque propuesto no busca acertar si un objeto está o no presente en una imagen, sino capturar las sutiles propiedades discriminatorias que poseen dentro de ella, para su posterior clasificación en la clase correspondiente. Es necesario definir el grado de información que dicho objeto aporta a la escena, teniendo en cuenta que los puntajes de detección de objetos pueden variar ampliamente entre las imágenes, dependiendo de las condiciones de estos en cada una. En esta sección, se extraen descriptores semánticos que muestran la ocurrencia de los objetos discriminantes en las diferentes clases conformando la información de la bolsa semántica. Esta descripción de imagen contextual permite agrupar automáticamente estructuras semánticas relevantes aprendiendo un modelo más discriminativo no supervisado (BOS). Este modelo no depende únicamente de la agrupación semántica de sus parches, sino que aporta una descripción más compacta de la imagen para su posterior clasificación.

3.3.1 Red SSD

La red SSD de [64] es uno de los algoritmos de detección de objetos más populares[65]. Para este proyecto se propone una arquitectura SSD con una resolución de entrada de 300 x 300(SSD300). Esta se elige con base en resultados de velocidad / precisión, Huang et al. [47] y reproduciendo la metodología experimental descrita en el documento SSD [64]. Asimismo, otros documentos también usan esta red como línea de base, [76], [77], [65], [78], al igual que la propuesta de detectores de objetos para robots de interior justificando en mayor medida porque se utiliza esta arquitectura [25]. La red SSD se basa en una red convolucional de avance hacia adelante que produce una colección de cuadros de tamaño fijo, delimitadores y puntajes que denotan la presencia de instancias de clase de objeto en esos cuadros, seguido de un paso de supresión no máxima para producir las detecciones finales[64]. La red SSD combina múltiples mapas de características con diferentes tamaños para generar predicciones, que hacen que la escala sea más invariante a los objetos. Estas predicciones combinadas de mapas de características múltiples producen dos salidas: Una, compensación de cuadro delimitador y otra, confianza de clase. La red consta de tres partes:

- Una arquitectura de convoluciones base para la clasificación de imágenes que proporcionará mapas de características de nivel inferior.

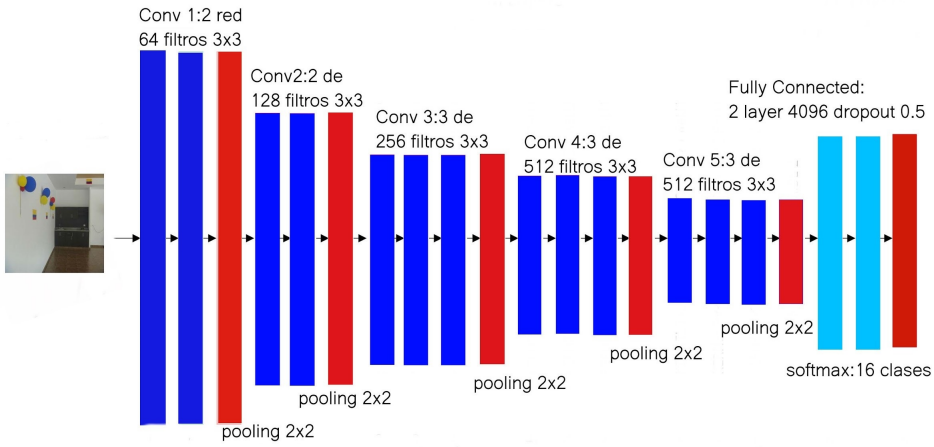
- Capas de convoluciones auxiliares agregadas en la parte superior de la red base que proporcionarán mapas de características de nivel superior.
- Convoluciones de predicción que localizan e identifican objetos en estos mapas de características.

3.3.2 Red base

Actualmente los sistemas de inteligencia artificial aprovechan arquitecturas entrenadas con millones de imágenes como estructuras de red base para sus primeras capas. La red base consiste en convoluciones apiladas en tamaño decrecientes con el propósito de generar mapas de representación que permitan detecciones en diferentes tamaños. La red base se puede ver en la Figura 3.8.

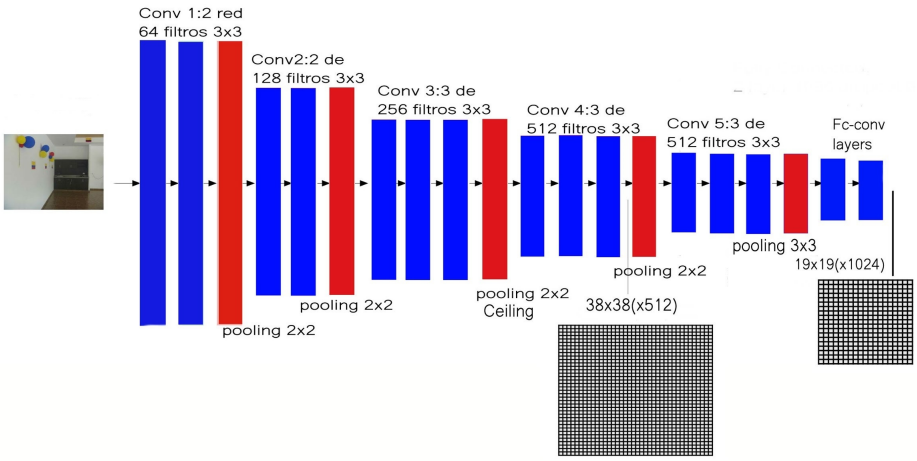
Para este caso y basado en el tutorial de Sagar Vinodababu [22], se hace la modificación de la quinta capa de agrupación de 2, 2 núcleos y 2 pasos a 3, 3 núcleos y 1 paso, y se trabaja fc6 y fc7(capas fully connected) en capas convolucionales conv6 y conv7. se eliminó la fc8 por completo, pues no se necesitan las capas completamente conectadas (es decir, clasificación) porque no tienen sentido para nuestra aplicación.

Figura 3.8 – Red base



una vez la arquitectura anterior sufre los cambios pertinentes de reducción del tamaño de cada filtro submuestreado de las capas convolucionales, se obtienen algunos mapas de convolución que permitirán codificar las respuestas subyacentes a las regiones que recorre de forma convolucional.

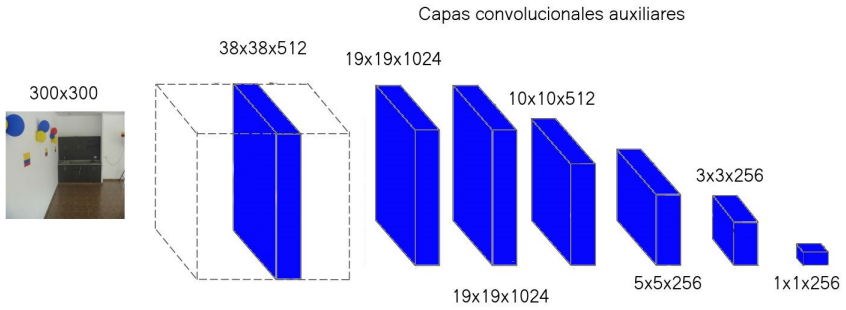
Figura 3.9 – Red base con VGG-16 modificada



3.3.3 Convoluciones auxiliares

Posteriormente a la red base, se le adicionan las capas convolucionales: conv8-2, conv9-2, conv10-2, y conv11-2 ya que es imperativo tener mapas más profundos que logren codificar regiones con objetos más pequeños; por lo tanto se apilan algunas capas convolucionales en la parte superior de la red base. Estas convoluciones proporcionan mapas de características adicionales que logra capturar información de los objetos en las diferentes escalas, cada uno progresivamente más pequeño que el anterior.

Figura 3.10 – Capas de características agregadas al final de una red base.



La Figura 3.10 muestra los diferentes mapas de características de la SSD,dichos mapas permiten predecir los desplazamientos de los cuadros predeterminados en diferentes

escalas y relaciones de aspecto.

Un concepto importante sobre el SSD es decidir cuál de los cuadros predeterminados se usará para una imagen determinada y luego predecir las compensaciones de los cuadros predeterminados elegidos para obtener la predicción final[64].

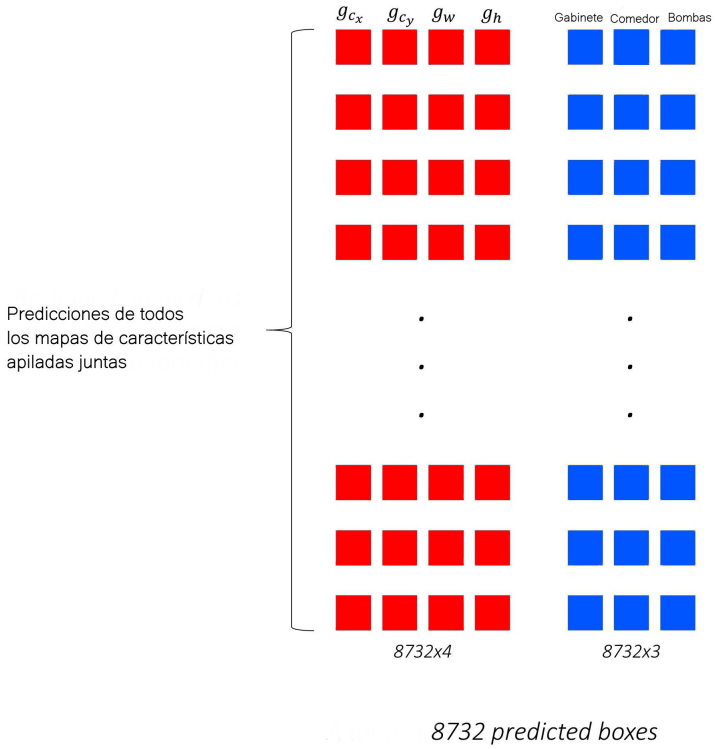
3.3.4 Capas de predicción

Las capas de predicción están unidas a la red base convolucional y capas SSD. Para una capa de características de tamaño $m \times m \times c$ donde m es el tamaño del mapa de características y c corresponde al número de canales. Se aplica un kernel convolucional de $3 \times 3 \times r \times$ (clases + coordenadas desfasadas), donde r es el número de cuadros delimitadores predeterminados y cada cuadro delimitador tendrá salidas (clases + 4) siendo r igual a 4. Por ejemplo, en Conv4-3, tiene un tamaño de $38 \times 38 \times 512$ por lo tanto, en Conv4-3, la salida es $38 \times 38 \times 4 \times (c + 4)$. Suponiendo que hay 20 clases de objetos más una clase de fondo, la salida es $38 \times 38 \times 4 \times (21 + 4) = 144,400$. En términos de número de cuadros delimitadores, hay $38 \times 38 \times 4 = 5776$ cuadros delimitadores. De manera similar se hace para las otras capas convolucional:

- Conv7: $19 \times 19 \times 6 = 2166$ cuadros (6 cuadros para cada ubicación)
- Conv8-2: $10 \times 10 \times 6 = 600$cajas (6 cajas para cada ubicación)
- Conv9-2: $5 \times 5 \times 6 = 150$cajas (6 cuadros para cada ubicación)
- Conv10-2: $3 \times 3 \times 4 = 36$ cajas (4 cuadros para cada ubicación)
- Conv11-2: $1 \times 1 \times 4 = 4$ cajas (4 cajas para cada ubicación)

Resumiendo, si se suman, $5776 + 2166 + 600 + 150 + 36 + 4 = 8732$ cuadros en total. Estas capas de predicción están unidas a múltiples puntos en la red base convolucional y capas SSD, concretamente conv4-3, conv7, conv8-2, conv9-2, conv10-2, conv11-2. Las capas inferiores capturan detalles más finos como objetos pequeños, mientras que las capas superiores capturan más información semánticamente significativa de los objetos de mayor tamaño. Por lo tanto, adjuntar varias capas de características debería ayudar a capturar los objetos de diferentes tamaños. Todas las capas de predicción son concatenadas al final de la red, lo que dará como resultado una sola capa de salida con un número fijo de predicciones de cuadro delimitador.

Figura 3.11 – Información del contenido de la capa final de predicción, para una imagen que contiene 3 objetos (gabinete, comedor y bombas), describe la información en dos etapas: El tamaño del cuadro delimitador en rojo y la predicción en azul.



3.3.5 Escalas y relaciones de aspecto de cuadros predeterminados

El modelo SSD usa diferentes proporciones para sus cuadros delimitadores predeterminados. Como se puede ver en la Figura 3.12, posee un nivel de sensibilidad en su elección, con relación a su escala y aspecto. En [64], se diseñan los cuadros predeterminados de modo que cada mapa de características corresponde a una escala específica junto con una lista de relaciones de aspecto para cada escala. Las escalas mínima y máxima, se establecen para la base de imágenes propia en $s_{min} = 0.2$ y $s_{max} = 0.9$ y para la base de imágenes UIUC Sports-8 en $s_{min} = 0.1$ y $s_{max} = 0.9$. Para cualquier mapa de características entre m mapas de características, la escala se elige como [22].

La función para calcular la escala predeterminada del cuadro delimitador s_k es.

$$s_k = s_{min} + \frac{s_{max} - s_{min}}{m - 1}(k - 1), k \in [1, m] \tag{3.1}$$

La altura y el ancho de los dos cuadros delimitadores predeterminados , h_k^a, w_k^a se

pueden calcular de la siguiente manera.

$$w_k^a = \mathrm{s}_k\sqrt{a_r} \tag{3.2}$$

$$h_k^a = \frac{\mathrm{s}_k}{\sqrt{a_r}} \tag{3.3}$$

Donde a_r es el número de relaciones de aspecto.

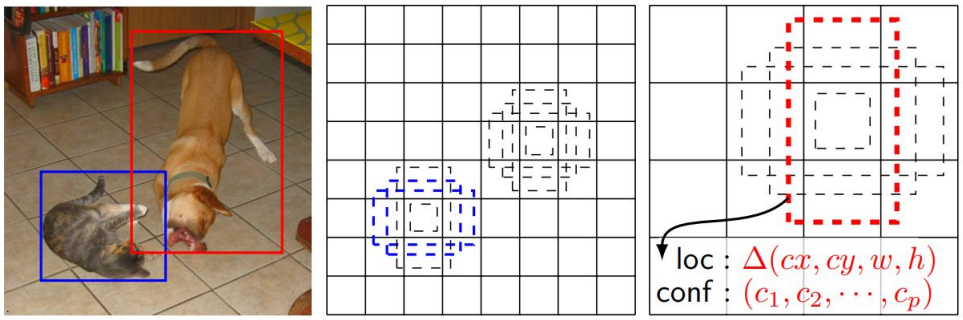
$$a_r \in \{1, 2, 3, 1/2, 1/3\}$$

Al igual que el centro del cuadro delimitador predeterminado se pude calcular de la siguiente forma.

$$x_c, y_c = \{\frac{i+0.5}{|f_k|}, \frac{j+0.5}{|f_k|}\} \tag{3.4}$$

Donde $|f_k|$ es el tamaño del mapa de características cuadradas k-ésimo $i, j \in [0, |f_k|)$, Para la relación de aspecto de 1, también se agrega un cuadro predeterminado cuya escala es $s_k = \sqrt{s_k s_{k+1}}$, resultando en 6 cuadros predeterminados por ubicación del mapa de características.

Figura 3.12 – Un ejemplo de cómo los cuadros predeterminados delimitadores se apilan.



(a) Image with GT boxes (b) 8×8 feature map (c) 4×4 feature map

Fuente: Liu et al. (2016).

3.3.5.1 Pérdida

La función utilizada es la pérdida MultiBox [79] [80], pero se extiende para manejar múltiples categorías de objetos. Consta de dos términos: la pérdida de confianza y la pérdida de localización. Sea $x_{ij}^p = \{1, 0\}$ un indicador para hacer coincidir el i-ésimo cuadro predicho con el j-ésimo cuadro de la etiqueta verdadera en la categoría p. Para el presente proyecto, la categoría de p se basa en los objetos que tenemos incluyendo fondo. La variable de emparejamiento x_{ij}^p es 1 cuando IoU (ecuación 3.25) coincide con cualquier

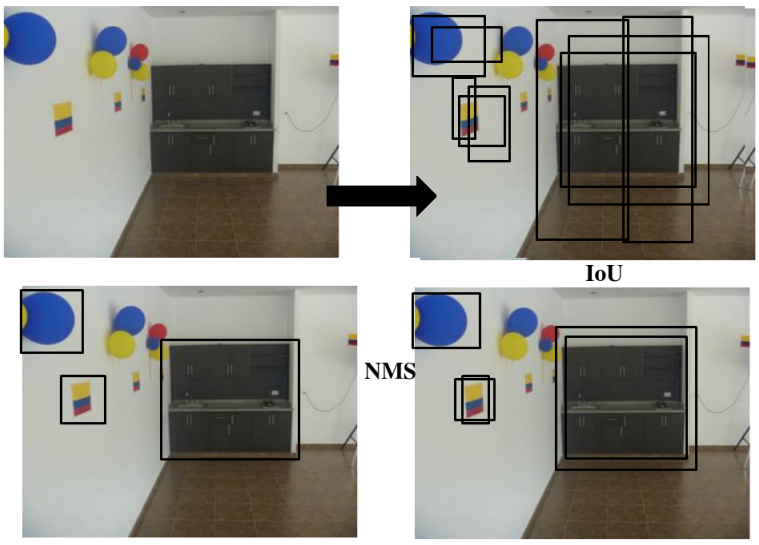
cuadro de etiqueta verdadera y el límite de la etiqueta predicha con un valor mayor a 0.5. Esto esencialmente convierte el problema en un problema de etiquetas múltiples para cada cuadro, además, para cada etiqueta verdadera el límite del cuadrado, también se hace coincidir con el cuadro predicho con la superposición de IoU más alta. los valores de x_{ij}^{p} se define así por:

$$x_j^p = \begin{cases} j & if IoU \geq 0.5 \text{ or max IoU} \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

Debido a la cantidad de cuadros delimitadores predichos se emplea la non-max supresión de IoU, como método para asegurar la detección verdadera de parches con el puntaje de confianza más grande. Para lograrlo lo primero que se debe examinar son las probabilidades asociadas con cada una de estas detecciones multi-objetos. Se ordenan sustrayendo las superposición más alta y extrayendo aquellas que cumplen con el índice IoU y una mayor probabilidad de contener el objeto suprimiendo aquellos que no cumplen las condicione estipuladas. Esto se puede resumir de la siguiente forma:

- Se extraen los puntajes de esta clase para cada uno de los 8732 recuadros.
- Se eliminan las casillas que no cumplen un cierto umbral para este puntaje.
- Los cuadros restantes (no eliminados) son candidatos para esta clase particular de objeto.

Figura 3.13 – Inferencia de los cuadros delimitadores



La función de pérdida de tareas múltiples se define como

$$L(x, c, l, g) = L_{conf}(x, c) + L_{loc}(x, l, g) \tag{3.6}$$

Donde la función de pérdida consta de dos términos, la $L_{conf}(x, c)$, que es la confianza y la pérdida de regresión del cuadro delimitador $L_{loc}(x, l, g)$. c es la confianza de clase (clase score), l es la predicción de compensación de localización, y g es la localización de la etiqueta verdadera.

3.3.5.2 Pérdida de ubicación

En la pérdida de localización, L_{loc}, se usa a partir de la pérdida de huber [81].

$$L_{\delta}(d) = \begin{cases} \frac{1}{2}d^2 & for|d| \leqq \delta, \\ \delta(|d| - \frac{1}{2}\delta) & \text{otherwise} \end{cases} \tag{3.7}$$

Donde d es la distancia entre la localización predicha y la localización de la etiqueta verdadera. Si establecemos $\delta = 1$, se obtiene la función de pérdida que se conoce como la pérdida de $L_1 - loss$.

$$L1_s(d) = \begin{cases} 0.5d^2 & if|d| \leqq 1, \\ |d| - 0.5 & \text{otherwise} \end{cases} \tag{3.8}$$

Existe una serie de razones para utilizar la pérdida de regularizar $L1_s$. En primer lugar, la función de pérdida de $L1$ no es diferenciable en 0. En segundo lugar, cuando $|d| < 1$ la función de pérdida tiene un gradiente menos empinado para optimizar mejor hacia las distancias más pequeñas.En tercer lugar, la pérdida de L1 tiene una función con menos restricción fuertes para puntos más alejados de la posición óptima, situación que no ocurre por ejemplo con el gradiente de la función de pérdida L2, que se vuelve demasiado grande cuando la distancia es grande, causando un proceso de aprendizaje inestable.La función de pérdida del cuadro predicho L_{loc} esta definida como.

$$L_{loc}(x, c^p, l_j, g_j) = \frac{1}{N^+} \sum_{i \in Pos}^{N_+} \sum_{m \in cx,cy,w,h} x_{ij}^p L1_s(l_i^m - \hat{g}_j^m) \tag{3.9}$$

donde $N^+ = \sum_{ij} x_{ij}^{p=1}$, que es un escalar para la cantidad de coincidencias positivas y l_i es la predicción de localización definida como el desplazamiento del centro, de la altura y el ancho. La d en la ecuación 3.7 se reemplaza por $l_i^m - \hat{g}_j^m$. Para $\hat{g}_j^m$ la regresión del centro

de predicción es hecho en relación con su cuadro delimitador predeterminado coincidente y definido como.

$$\hat{g}_j^{cx} = (g_j^{cx} - b_i^{cx})/b_i^w \tag{3.10}$$

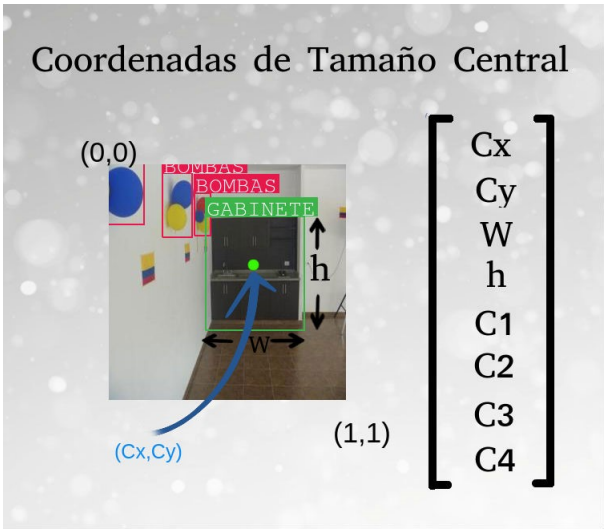
$$\hat{g}_j^{cy} = (g_j^{cy} - b_i^{cy})/b_i^h \tag{3.11}$$

$$\hat{g}_j^w = log(\frac{g_j^w}{b_i^w}) \tag{3.12}$$

$$\hat{g}_j^h = log(\frac{g_j^h}{b_i^h}) \tag{3.13}$$

Las cuatro coordenadas de los cuadros delimitadores g^{cx}, g^{cy} para el centro y g^w,g^h alto y ancho como se ve en la Figura 3.14. Los términos b_i^w,b_i^h,b_i^{cx},b_i^{cy} son las respectivas coordenadas del cuadro delimitador predicho coincidente. La división de la altura y ancho se utiliza para normalizar el ancho y la altura. La escala logarítmica se utiliza para equilibrar las diferencias de escala, esto tiene sentido porque un cierto desplazamiento sería menos significativo para un prior más grande que para un prior más pequeño. La misma operación se realiza en l_i^m.

Figura 3.14 – coordenadas



3.3.5.3 Pérdida de confianza

La pérdida de confianza es esencialmente la pérdida de entropía cruzada. Es importante que el modelo reconozca los objetos que existen en la imagen y la ausencia de

ellos. Teniendo en cuenta que generalmente solo hay un puñado de objetos en una imagen esto crea un gran desequilibrio entre la clase fondo (cuadros delimitadores negativos) y objetos (cuadros delimitadores positivos), que dificulta el proceso de optimización. Para contrarrestar este problema, se utiliza la minería negativa. En lugar de sumar todos los cuadros delimitadores negativos, se ordenan por confiaza de la clase y los cuadros de límite M negativos superiores que están seleccionados. Dónde la relación entre M y los cuadros delimitadores positivos es 3:1. La pérdida de confianza se define de la siguiente manera,

$$L_{conf}(x,c) = -\frac{1}{N^+}\sum_{i\in Positivo}^{N_+} x_{ij}^p log(\hat{c}_i^p) - \frac{1}{N^-}\sum_{i\in Negativo}^{N_-} log(\hat{c}_i^0) \tag{3.14}$$

$$\hat{c}_i^p = \frac{exp(c_i^p)}{\sum_P exp(c_i^p)} \tag{3.15}$$

donde $N^- = M$

Finalmente se tendría $l = L_{conf} + \alpha L_{loc}$ donde $\alpha = 1$

3.4 REPRESENTACIÓN Y RECONOCIMIENTO DE ESCENAS

3.4.1 Vector de Descriptores Localmente Agregados (VLAD)

Después de haber introducido la arquitectura de SSD para describir las representaciones semánticas de los parches, viéndolo como un descriptor denso, el siguiente paso es adicionar estos parches a una representación de bolsa semántica considerando su salida en la última capa, la cual posee información asignada a cada parche que contiene un objeto, es decir se tiene una cantidad de 8732 parches evaluados, pero solo un porcentaje tiene información del objeto. Según la Figura 3.11 esta información se reparte entre P x N-objetos, donde P es el número de parches que se encontró por imágenes. Por ejemplo, si se tiene un conjunto de datos con 51 objetos y para la primera imagen que se expone al detector ya entrenado se detectan 3 parches relevantes, la matriz resultante tendría un tamaño de 3x51. Se diseña a partir de esta información una bolsa semántica que representa cada imagen a partir de sus objetos a través de descriptores agregados (VLAD). Este método a diferencia de los modelos generativos no posee una parametrización tan compleja y es sensible a la inicialización que puede converger a un mínimo local [35].

Formalmente, dada la descripción local de la imagen a través del detector N D-dimensional, se tienen los objetos {Pi} como entrada, K centros de clúster ("palabras visuales"), y {ck}como parámetros VLAD. La imagen de salida VLAD representa la

posición V, la cual es K x D-dimensional, pero esta matriz se convierte en un vector y después de la normalización, se utiliza como representación de la imagen. El elemento (j, k) de V se calcula de la siguiente manera:

$$V(j,k) = \sum_{i=1}^{N} a_k(x_i)(x_i(j) - c_k(j)) \tag{3.16}$$

Donde $x_i(j)$ y $c_k(j)$ son las dimensiones j-ésima de la i-ésima descriptor y k-ésima centro del grupo, respectivamente. $a_k(x_i)$ denota la pertenencia al descriptor x_i a k-ésima palabra visual, es decir, es 1 si el grupo c_k es el grupo más cercano al descriptor x_i y 0 de lo contrario. Intuitivamente, cada D-dimensional de la columna k de V, registra la suma de los residuos $(x_i - c_k)$ de descriptores que se asignan al clúster c_k. La matriz V es luego normalizada square-rooting y convertida en un vector. Finalmente al vector se le aplica la norma L2 [66].

Norma square-rooting:

$$x_j \leftarrow sing(x_j)|x_j|^{0.5} \tag{3.17}$$

El término

$$\bar{a}_k(x_i) = \frac{e^{-\alpha\|x_i - c_k\|^2}}{\sum_{k'} e^{-\alpha\|x_i - c_k\|^2}}, \tag{3.18}$$

$\bar{a}_k(x_i)$ varía entre 0 y 1, con el peso más alto asignado al centro de clúster más cercano. $\propto$ es un parámetro (constante positivo) que controla la asignación de la respuesta con la magnitud de la distancia. Tenga en cuenta que para $\alpha \to +\infty$ esta configuración replica exactamente el VLAD original como $\bar{a}_k(x_i)$ para el grupo más cercano sera 1 o 0 en caso contrario. Al expandir los cuadrados en 3.17, es fácil ver que el término $e^{-\alpha\|x_i\|^2}$ se cancela entre el numerador y el denominador resultando en una asignación suave de la siguiente forma [28].

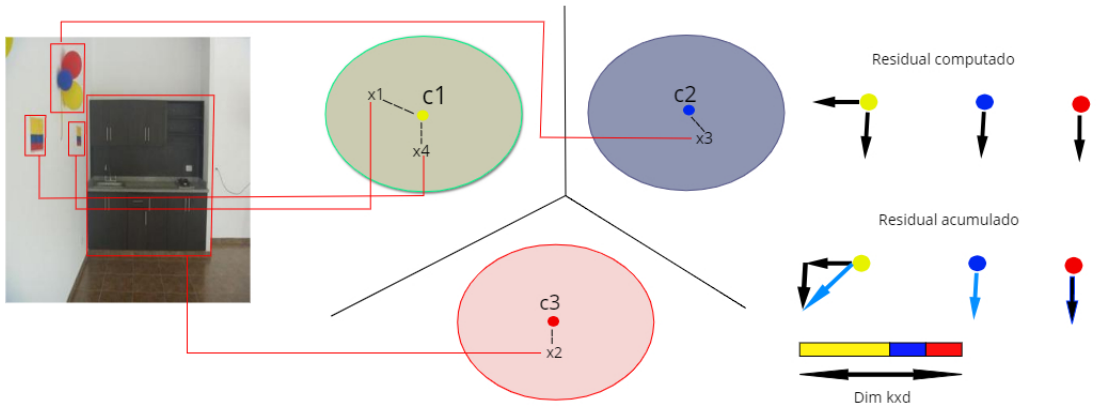
$$\bar{a}_k(x_i) = \frac{e^{W_K^T x_i + b_k}}{\sum_{k'} e^{W_{K'}^T x_i + b'_k}}, \tag{3.19}$$

Donde la salida de la convolucional para cada parche se pasa a través de la función soft-max $\sigma_k(z) = \dfrac{exp(z_k)}{\sum_{k'} exp(z'_k)}$ para obtener la asignación suave final $a_k(x_i)$, gracias a $W_K = 2\alpha c_k$ y $b_K = -\alpha\|c_k\|^2$ se logra expresar de la forma.

$$V(j,k) = \sum_{i=1}^{N} \frac{e^{W_K^T x_i + b_k}}{\sum_{k'} e^{W_{K'}^T x_i + b'_k}} (x_i(j) - c_k(j)) \tag{3.20}$$

El resultado es una representación de la imagen a partir de sus objetos, desde una descripción local, proporcionada por el detector y un aumento de datos obteniendo un conjunto de clústeres más densos. Por lo tanto, el desafío clave es hacer la agrupación de VLAD que pueda atrapar la semántica de los parches en un espacio euclidiano. La salida después de la normalización es un descriptor de dimensión $(K \times D) \times 1$.

Figura 3.15 – Proceso de generación de firmas



3.4.2 Vector de Descriptores Localmente Agregados (VLAD) y Análisis Lineal de Componentes Discriminantes(LDA)

LDA [1] permite la mayoría de las veces realizar una reducción supervisada de la dimensionalidad, proyectando los datos de entrada en un subespacio lineal que consiste en las direcciones que maximizan la separación entre clases. Suponiendo que hay K diferentes grupos, cada uno de los cuales tiene una distribución normal multivariada con vectores medios $\mu_k(k = 1, ..., k)$ y matriz de covarianza común Σ. Donde los vectores medios y las matrices de covarianza son casi siempre desconocidos, por lo tanto, para estimar estos parámetros se utiliza máxima verosimilitud.

La idea de LDA es clasificar las observaciones x_i al grupo k, que minimizan la varianza dentro del grupo, es decir,

$$k = armin_k(x_i - \mu_k)^T \Sigma^{-1}(x_i - \mu_k) \tag{3.21}$$

Bajo supuestos una distribución normal multivariada, es equivalente a encontrar el grupo que maximiza la verosimilitud de la observación. Generalmente, se puede estimar la probabilidad previa usando la proporción del número de observaciones en cada grupo con respecto al total. Por ejemplo, deje que $\pi_k = \frac{n_k}{n}$ sea la proporción del grupo k, de modo

que $\pi_1 + ... + \pi_k = 1$ Entonces, en lugar de maximizar la verosimilitud, la probabilidad posterior es maximizada; la observación de pertenecer a un grupo en particular,

$$k = armin_k[-\frac{1}{2}(x_i - \mu_k)^T \Sigma^{-1}(x_i - \mu_k) + log\pi_k] \tag{3.22}$$

Simplificando (3.21), las funciones k de LDA son,

$$d_k(x) = x^T \Sigma^{-1}\mu_k - \frac{1}{2}\mu_k^T \Sigma^{-1}\mu_K - log\pi_k \tag{3.23}$$

Dada la descripción con respecto al nuevo espacio transformado por LDA de la imagen local, la salida V esta dada por un vector de $1 \times K \times D$, donde D cambia en el caso de la base propia a 20 dimensiones y 5 para el conjunto de deportes. A su vez , se suprime el parámetro de control $a_k(x_i)$, ya que éste se ajusta únicamente cuando el número de clústers es igual al numero de objetos, por tanto la ecuación del VLAD estaría sujeta a :

$$V(j,k) = \sum_{i=1}^{N}(x_i(j) - c_k(j)) \tag{3.24}$$

3.5 K-NEAREST NEIGHBOR

kNN [1] es uno de los algoritmos de aprendizaje automático más simples. El algoritmo clasifica cada dato nuevo en el grupo que corresponda, según tenga k vecinos más cerca de un grupo o de otro. Es decir, calcula la distancia del elemento nuevo a cada uno de los existentes, y ordena dichas distancias de menor a mayor para ir seleccionando el grupo al que pertenecer. Este grupo será, por tanto, el de mayor frecuencia con menores distancias. Existen diferentes medidas para el cálculo de la distancia, como Euclidean Distance, Euclidean Squared, City-block y Chebyshev. Entre todos estos, la distancia Euclidiana es la opción más popular para medir la distancia entre los dos puntos. La distancia euclidiana [1] entre dos puntos x, y de dimensiones M está dada por:

$$d(x,y) = \sqrt{\sum_{i=1}^{M}(x_i - y_i)^2} \tag{3.25}$$

3.6 DETALLES DE IMPLEMENTACIÓN

Para la implementación de la metodología y garantizar un excelente rendimiento del método de clasificación se parte del conocimiento innato de la persona dado que en las teorías de orientación de imágenes, el foco de atención se centra en las regiones de la escena

cuyas características de imagen semánticamente son poco interpretadas [82][83][84]. En el caso de la metodología implementada, la atención del espectador se orienta a las regiones de escena visualmente destacadas, es decir aquellas características básicas de la imagen, como el contraste de luminancia, el color y la orientación de los bordes, los cuales se utilizan para formar un mapa de prominencia que proporciona la base del etiquetado. Con todo lo anterior la primera fase del método implementado es sencilla pero el defecto potencial depende de la selección apropiada del número de objetos a etiquetar para cada categoría. En la segunda fase del método se requiere dotar una arquitectura que pueda proporcionar información semántica de regiones prominentes a partir de la información etiquetada de tal forma que imite el conocimiento innato del espectador, por tanto la arquitectura idónea se establece en un marco de detección de objetos(SSD-300). Esta arquitectura requiere un entrenamiento que depende del sistema de cómputo utilizado. En cuanto a la etapa tres que consiste en una representación del espacio semántico requiere un aumento de los datos a partir de una series de transformaciones de escala, orientación y color, Dicha representación se incluye en el modelo SSD-300 que describe regiones discriminates que serán utilizadas para entrenar el sistema de representación(VLAD) creando firmas únicas para cada imagen logrando hacer distinguibles las escenas para su posterior entrenamiento del modelo de clasificación.

3.7 MÉTRICAS DE EVALUACIÓN

En esta sección describen las diferentes métricas de evaluación del modelo propuesto y se analiza el desempeño de éstas. Las métricas utilizadas son:

- Curva precision-recall
- F1-score
- Promedio micro
- Promedio macro
- Promedio ponderado
- Mean Average Precision
- Average precision.
- Precisión
- Recall

La métrica de evaluación requerida en el detector de objetos se realiza mediante la comprobación de la predicción correcta del objeto y su precisión de localización. La medida es la Intersección sobre la unión (IoU), también conocida como límite del cuadro de superposición. El desempeño del modelo de detección de objetos es cuantitativo y se utiliza para medir las predicciones relevantes.

La calidad general del modelo de detección de objetos se calcula en términos de IoU. En cuanto al detector de escenas se utilizan: Precisión, recall, curva precision-recall, promedio micro, promedio macro, promedio ponderado, f1-score y Accuracy (acc). Para un conjunto de datos multiclase de objetos a gran escala, la mean average precision (mAP) se utiliza para medir el rendimiento [85] [86].

3.7.1 Definición del detector

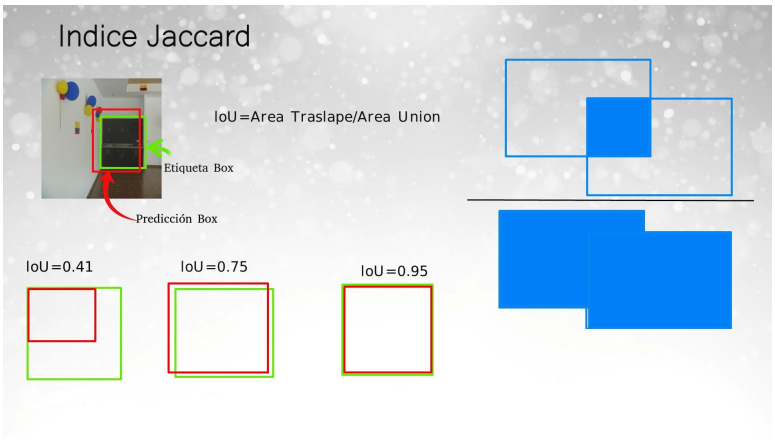
Un IoU se calcula como la relación del área de superposición y el área de la unión. El área de superposición es el área común total (la intersección) entre el cuadro delimitador predicho (Bp) y el cuadro delimitador de la etiqueta verdadera (Bgt). El área de la unión es el área total cubierta por (Bp) y (Bgt).

$$IoU = \frac{area(B_{gt} \bigcap b_p)}{area(B_{gt} \bigcup b_p)} \tag{3.26}$$

La Figura 3.16 muestra los ejemplos del cuadro delimitador predicho y el cuadro delimitador de etiqueta verdadera. El Bp dibujado en color rojo es del archivo anotado y se predice el Bgt en color verde del detector entrenado. A diferencia de los problemas de clasificación, la precisión de detección de objetos es un cálculo bastante complejo. La coincidencia exacta de las coordenadas (X, Y) del cuadro delimitador de etiqueta verdadera y el cuadro predicho son extremadamente diferentes [87]. Por esta razón, la métrica de evaluación del rendimiento de detección de objetos se define de tal manera que cuanto más se superponga el Bp con el Bgt, mejor será el rendimiento del modelo

En cuanto al modelo de detección utilizado, la supresión non-max reduce los falsos positivos a través de una serie de pasos. En primer lugar, solo se consideran cuadros con una confianza superior a un IoU de 0.5. Luego selecciona el cuadro delimitador con la mayor confianza y se suprimen todos los que tienen un IoU mayor que 0.45 comparado con el seleccionado usando éste como el límite del cuadro predicho final. Este proceso de selección de recuadro delimitador de mayor confianza y de supresión de los otros recuadros es repetido hasta que todos sean suprimidos o considerados como predicción final. Todas las predicciones positivas restantes se ordenan a partir de la confianza predicha positiva más alta y se consideran como verdaderos positivo (TP), las otras predicciones con un $IoU \leq 0.5$ de las etiquetas verdaderas y que tienen menos puntajes se consideran

Figura 3.16 – Cálculo de la intersección sobre la unión es tan simple como dividir el área de superposición entre los cuadros delimitadores por el área de la unión



falsos positivos (FP). Los cuadros delimitadores de etiquetas verdaderas, que no tienen predicciones asignadas se consideran falsos negativos (FN). Los verdaderos negativos (TN) quedan fuera de consideración porque los verdaderos negativos no influyen en la precisión y el recall.

3.7.2 Explicación Métricas de clasificación

Estas métricas se basan en una matriz de confusión que incorpora la información sobre el resultado de predicción de cada muestra de prueba ver Tabla 3.1. TP significa el número de predicciones positivas verdaderas, TN – Predicciones negativas verdaderas, FP - Predicciones falsas positivas, y FN - Predicciones falsas negativas.

Tabla 3.1 – Matrix Confusion

	Clases verdadera		
Clases actuales	Positive	Negative	Total
Positive	$TruePositive(TP)$	$FalseNegative(FN)$	$TP + FN$
Negative	$FalsePositive(FP)$	$TrueNegative(TN)$	$FP + TN$
Total	$TP + FP$	$FN + TN$	N

Al evaluar el modelo de clasificación es necesario comprender algunos conceptos. La precisión (P) se define como la capacidad del clasificador de no etiquetar como positiva una muestra que es negativa centrándose en la exactitud del modelo, mientras que el recall (R) es la capacidad del clasificador para encontrar todas las muestras positivas, también se conoce como la sensibilidad del modelo. Tanto P como R se definen como.

$$Precision = \frac{TP}{TP + FP} \tag{3.27}$$

$$Recall = \frac{TP}{TP + FN} \tag{3.28}$$

F1-score permite medir el rendimiento del modelo calculando precision-recall dando mejor percepción del desempeño del modelo. Se puede interpretar como un promedio ponderado de la precisión y el recall.

$$F1score = 2\text{x}\frac{Precision\text{x}Recall}{Precision + Recall} \tag{3.29}$$

La exactitud (accuracy) es una métrica para evaluar modelos de clasificación. Se puede interpretar como la exactitud (accuracy), que es la fracción de predicciones que el modelo realizó correctamente. De manera formal se escribe de la siguiente forma:

$$A = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.30}$$

3.7.3 PR-curve

La curva PR en una métrica, el área bajo la curva (AuC) se puede calcular. A partir de la precisión media(average precision) se puede calcular tomando los valores generales de precisión y recall entre 0 y 1 [86].

$$\int_0^1 P(k)dk \tag{3.31}$$

Se aproxima la integral a partir de la suma sobre la precisión de las precisiones en cada valor umbral posible, multiplicado por el cambio en el recall.

$$\frac{1}{p}\sum_{n=1}^{N} P(k)\Delta R(k) \tag{3.32}$$

Donde N es el número total de imágenes en el conjunto de datos, P (k) es la precisión en un corte de k imágenes y delta r(k) es el cambio en el recall que ocurrió entre el corte k-1 y el corte k. En lugar de la precisión promedio, se utiliza la precisión promedio interpolada. La precisión promedio interpolada reemplaza la precisión en el corte k por la máxima precisión observada en todos los cortes con mayor recall y se define de la siguiente forma:

$$\frac{1}{p}\sum_{n=1}^{N} max_{\hat{k}>k} P(\hat{k})\Delta R(k) \tag{3.33}$$

3.7.4 Mean average precision

Mean average precision (mAP) se ha utilizado ampliamente para comparar las precisiones generales de los modelos de detección de objetos en conjuntos de datos de múltiples clases, desde su primera introducción en el desafío Pascal VOC 2007 . Los pasos de cálculo para mAP basados en Pascal VOC son los siguientes:

- Inicialmente, la precisión y recall se calculan en función de la IoU. Un IoU mayor que el umbral (generalmente, 0.5) se considera detección real.
- AP se calcula promediando el valor de precisión máximo en diferentes niveles del recall. Esto normalmente se toma de la curva precision-recall. Los valores de precisión se recopilan de diferentes valores de recall (11 valores), que van desde (0.0, 0.1, ..., 1.0) Las ecuaciones 3.33 - 3.37 muestran el cálculo AP en Pascal VOC .

$$AP = \frac{1}{11}(AP_r(0.0) + AP_r(0.1) + ... + AP_r(1.0)) \tag{3.34}$$

$$AP = \frac{1}{11} \sum_{r \in (0.0, 0.1, ..., 1.0)}^{N} AP_r \tag{3.35}$$

$$AP = \frac{1}{11} \sum_{r \in (0.0, 0.1, ..., 1.0)}^{N} P_{interp}(r) \tag{3.36}$$

$$P_{interp}(r) = max_{r' \geq r} p(r') \tag{3.37}$$

El $P_{interp}(r)$ es la precisión óptima para recuperar valores por encima de r. La ecuación simplificada para el cálculo AP es:

$$AP = \frac{1}{11}(P_{interp}(0) + P_{interp}(0.1) + ... + P_{interp}(1.0)) \tag{3.38}$$

- Se calcula el AP para todas las clases de objetos presentes en el conjunto de datos.
- Finalmente, se calcula la mean average precision (mAP) tomando la media de AP sobre todas las clases de objetos.

4 RESULTADOS

En esta sección se muestran los resultados de varios experimentos realizados con distintos modelos propuestos que buscan representar una escena a partir de la semántica de sus objetos. Donde en primer lugar, se tiene una visualización de los datos en dos dimensiones, observando su separabilidad y el comportamiento de los diferentes clústeres formados para cada parche. De igual forma se presenta una matriz de correlación con respecto a cada parche para identificar la descripción según su clase. El enfoque es evaluar inicialmente los modelos en la base de datos propia. Luego se amplian los resultados replicando el modelo para una base genérica. Se concluye el capítulo con una comparación con el estado del arte.

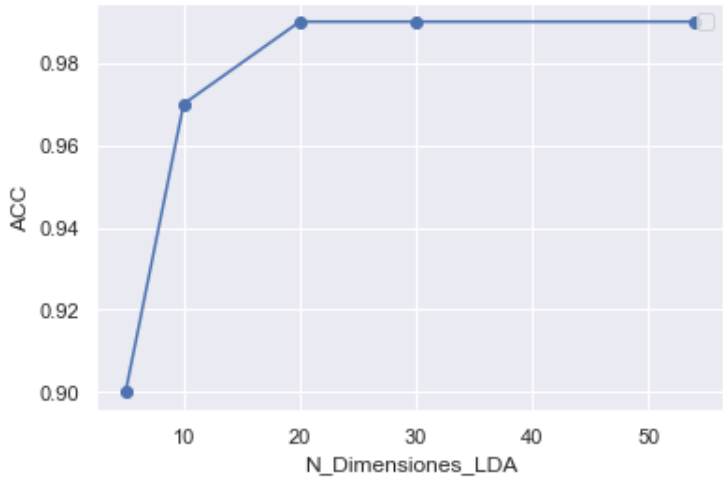
4.1 ANÁLISIS DEL MODELO

Una vez que se tiene una representación semántica única para cada objeto (vectores de gradientes para cada parche), es necesario incorporar estas abstracciones únicas del conjunto de parches de la imagen para cada clase de forma no supervisada, por tanto, se tiene un espacio de representación euclidiano de alta dimensionalidad, difícil de mapear gracias a la semántica de algunas regiones que puede ser ruidosa [88]. Por tanto, es necesario buscar una mejor representación y separabilidad, para ello se utiliza el modelo LDA que requiere un análisis adicional para alcanzar una mejor comprensión de los datos. Los parámetros del estimador que se utilizan para aplicar estos métodos se optimizan mediante la búsqueda de validación cuadrícula en una cuadrícula de parámetros como se ve en los resultados de exactitud(acc) de la Figura 4.1 donde se muestra que el mejor puntaje de clasificación es del 99% para 20 dimensiones.

Los objetos mostrados en la Figura 3.3 fueron etiquetados de forma manual de acuerdo con criterios cognitivos propios de la persona transfiriendo información representativa cotidiana, por lo tanto los objetos más representativos según el etiquetado manual fueron los de la Figura 3.3. Los parches a menudo comparten información común y conducen a similitudes que pueden ser confusas para el modelo, por consiguiente es común ver algunos clúster que comparten información entre sí.

Al representar los datos en forma bidimensional, se conoce el tamaño de la matriz 20215x51 parches para las 5000 imágenes aumentadas del conjunto inicial. Se utiliza t-SNE para realizar la reducción de dimensionalidad y obtener coordenadas bidimensionales [89], así pues, se muestra la dificultad de la tarea a realizar y como se configuran los objetos en ese espacio. La Figura 4.2 muestra una tendencia de los parches bastante lógica al

Figura 4.1 – LDA número de dimensiones óptimo



Fuente: python

organizar los objetos en subgrupos que pertenecen a la misma clase y rechazan los de las otras clases.

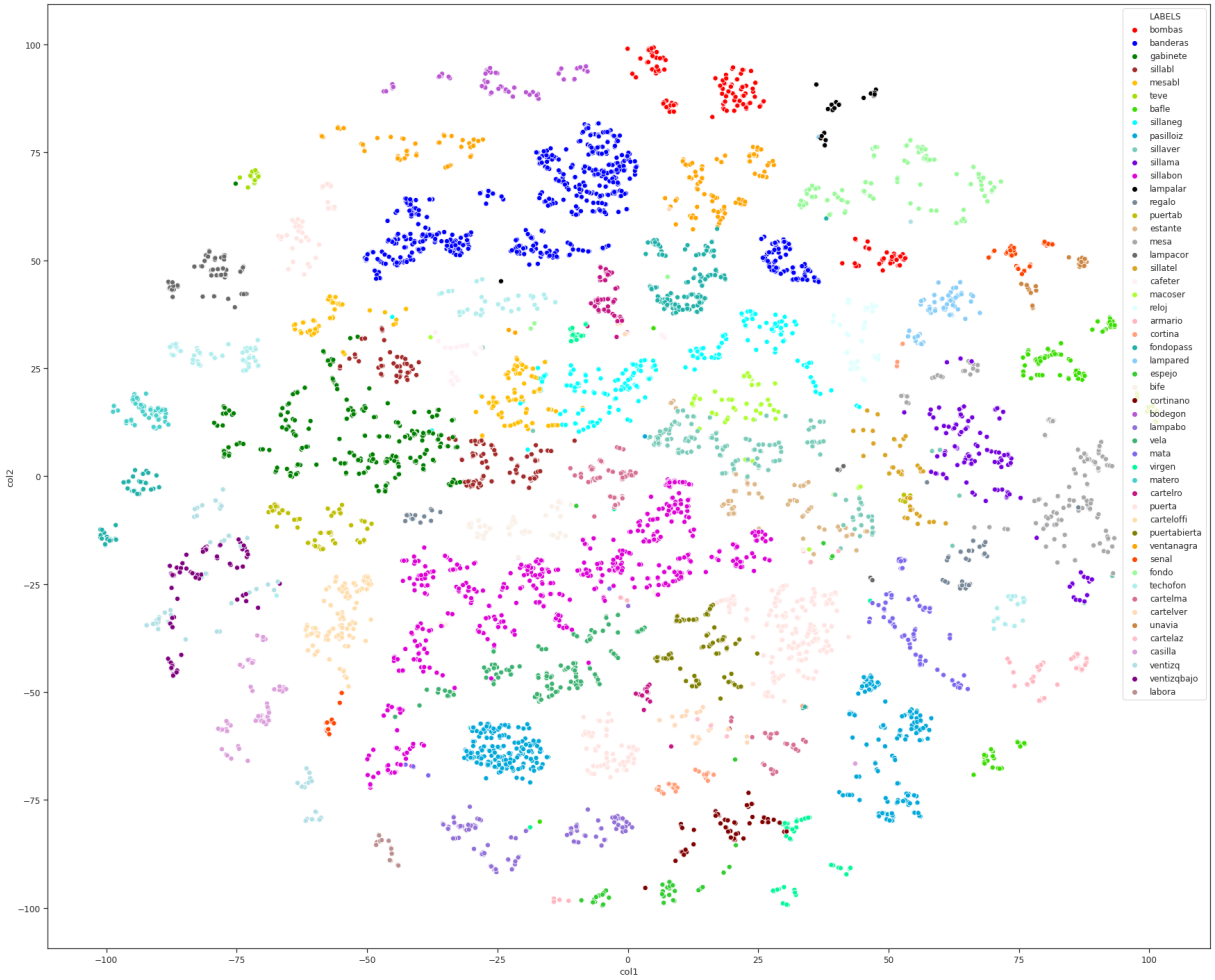
A continuación, es posible visualizar cómo los parches se relaciona directamente con las clases y a su vez estos contienen objetos únicos que describe las escenas; si bien es posible visualizar ciertas correlaciones débiles entre clases que muestran ruido semántico producto de la presencia de algunos objetos comunes y errores de descripción por parte de la SSD. Esto hace que el modelo utilizado de bolsa semántica(VLAD) tenga problemas; sin embargo, la fuerte correlación entre objetos de la misma clase ofrece una pista importante para solucionar este posible problema. Es evidente observar esta hipótesis en la Figura 4.5 con ayuda adicional de la Tabla 4.1.

Tabla 4.1 – Número de parches por clase y posición en la matriz de características convolucionales

Índice	Inicio	Final	Número de parches
Cocina	1	4924	4924
Sala 1	4924	8730	3806
Sala 2	8730	13412	4682
Pasillo 1	13412	16926	3514
Pasillo 2	16926	20314	3289

Según la gráfica es simple realizar una inspección visual que arroje una compresión más precisa, analizando propiedades únicas que posiblemente serán representativas para el espacio semántico de características; si bien las zonas con mayor iluminación coinciden entre los intervalos de las diferentes clases, esto sucede para la clases(Cocina,sala1,sala2, pasillo1) existen correlaciones con una proporción débil presente en las imágenes, por

Figura 4.2 – Representación de datos bidimensional con t-SNE



Fuente: python

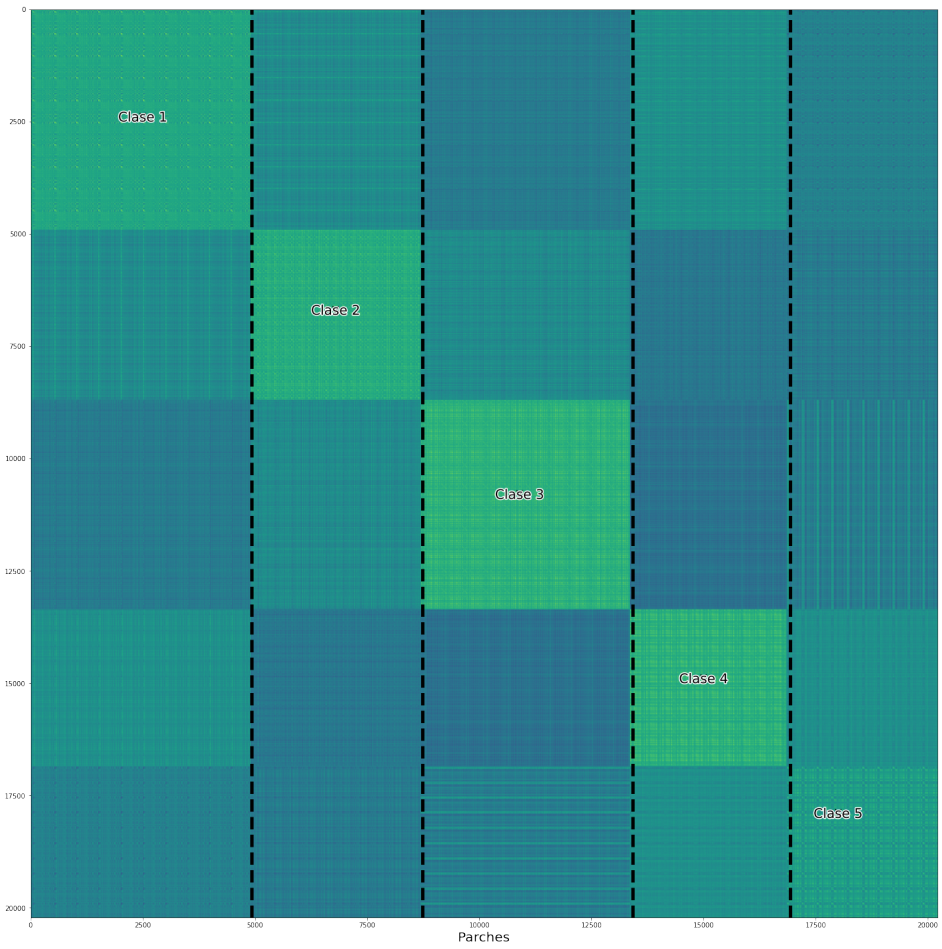
ejemplo la clase 5 presenta mayor ruido semántico ya que es muy similar a clase 4 por tanto tendrá mayor dificultad para diferenciarse de las otras.

Tabla 4.2 – Muestra el grado de correlación (C), entre los objetos de las diferentes imágenes.

Relación	Alta C	Media C	Baja C
Cocina	-	2-4	5-3
Sala 1	-	1-3	4-5
Sala 2	-	2	4-5-1
Pasillo 1	5	1	3-2
Pasillo 2	4	-	3-2-1

La Tabla 4.2 muestra las diferentes similitudes que tienen los datos en crudo, estos datos comprenden todos los parches que el sistema detectó para la base de entrenamiento aumentada, no obstante, el conjunto es caótico y requiere un sistema que logre discriminar

Figura 4.3 – Matrix de correlación de los parches



Fuente: python

las correlaciones altas y minimice la información ruidosa. La descripción mostrada en este gráfico comprende objetos detectados mas no son escenarios, por tanto, esa separabilidad natural que se observa es causada por la correlación única de los objetos de su clase, demostrando una aproximación de la hipótesis planteada, la cual describe que sí es posible representar una escena a través de sus objetos.

4.1.1 Detector SSD - Modelo Base

En esta sección, se evalúa el rendimiento del detector SSD como el modelo base utilizado para describir la semántica de los objetos de cada imagen, (ver sección 3.3 para más detalles)Logrando atrapar el estimulo natural del espectador al momento de etiquetar la base de datos. Las imágenes de entrada para este modelo deben de tener un tamaño, de 300 x 300 pixeles. En esta primera sección se prueba el modelo con tres bases de datos de imágenes: conjunto propio, compuesto con imágenes de interiores, base de imágenes de deportes UIUC Sport-8 y PascalVOC . En el estado del arte la mayoría de las bases

disponibles están constituidas por un conjunto de imágenes de entrenamiento e imágenes de pruebas y validación. Todas las imágenes fueron etiquetadas de forma manual exceptuando aquellas que no lo requeren.

Tabla 4.3 – Conjuntos de pruebas mAP para el modulo de deteccion SSD300

Bases de datos	**mAP %**
Propia	82.3
UIUC Sports-8	61.6
VOC2007	77.1
VOC2007[22]	77.2

En el proceso de verificación de la eficiencia del SSD300 se utilizó la base de datos PascalVOC replicando las mismas condiciones del estado del arte. Los valores obtenidos permiten asegurar los resultados mostrados para las dos bases propias y el UIUC Sports-8 que están diseñadas esplisitamentes para detección de escenas en términos de objetos hacen que este proceso sea confiable.Como se muestra en la Tabla 4.3.

Tabla 4.4 – Los diferentes conjuntos de prueba para la base propia y UIUC Sports-8 como también el conjunto de validación de PascalVOC

B-Propia	**mAP(%)**	**UIUC**	**mAP(%)**	**VOC2007**	**mAP(%)**
armario	100	bate	51.81	aeroplane	78.88
bafle	80.91	bote	76.24	bicycle	83.52
banderas	90.91	caballo	84.71	bird	76.23
bife	100	campo	71.43	boat	72.18
bodegon	100	gimnasio	48.26	bottle	45.98
bombas	72.73	pelota	19.80	bus	87.05
cafeter	54.55	personaboc	50.63	car	86.56
cartelaz	100	personac	78.56	cat	88.28
cartelma	67.15	personaesp	56.52	chair	59.17
carteloffi	52.35	personan	69.03	cow	82.56
cartelro	100	personap	61.01	diningtable	75.68
cartelver	100	personar	78.87	dog	87.78
casilla	72.73	personaro	58.24	horse	87.78
cortina	100	personatab	58.04	motorbike	83.17
cortinano	100	techo	52.27	person	78.84
espejo	100	velero	79.54	pottedplant	50.72
estante	80.52	wickets	60.10	sheep	79.36
fondo	81.82	wicketspalo	54.55	bate	51.81

Para este propósito, se utiliza mAP(Mean Average Precision), como métrica estándar. El modelo obtiene 77.1% mAP en la base voc2007, igual que el resultado reportado en el estado del arte [22]. En cuanto a la base propia, hay objetos fácilmente reconocibles dentro de sus clases (bafle(80.91),banderas(90.91), armario (100)), según la Tabla 4.4;por

tanto con este primer resultado es posible establecer que el modelo de detención es sensible a zonas que sobresaltan según las etiquetas del espectador basada en atención [82].

De la Tabla 4.4 basada en una primera preclasificación se podría deducir lo siguiente: Observando los resultados de la tabla anterior, en la base de imágenes de deportes UIUC Sports-8, algunos objetos pequeños con una proporción de 50 píxeles o menos, son difícilmente reconocidos, este es el caso del objeto *pelota*, el cual tiene un 0.19% de acierto y no es el único objeto afectado por esta condición; Esta dificultad posiblemente es debido a la resolución de la imagen en combinación con múltiples capas fijas de predicción, esto hace que los detalles finos que están presentes en la imagen, generen confusión al momento de detectar los objetos, debido a las debilidades del modelo base, como se menciona en [90]. Objetos con estas características pueden influir de manera negativa el resultado final de la clasificación, por lo tanto, es indispensable analizar qué tan valiosas son las características para la clase que aportan dichos objetos. Para solucionar esta situación se podría pensar en deshacerse de ellos, ya que suministran información ruidosa a la bolsa semántica, debido a su baja probabilidad de detección o información nula al no detectarlos. Sin embargo, la forma más adecuada de lidiar con ellos es a traves de una buena representación del espacio de características.

De las múltiples pruebas realizadas con las bases de imágenes disponibles se concluyo, que otro de los problemas que puede afectar el resultado final de la clasificación es el grado de percepción que sufren algunas regiones de la imagen por los cambios de perspectiva (Variación en los ángulos de visión) [24],Los cuales se solucionan considerando otros parches que posiblemente tendrán mejores condiciones de representación en el espacio semántico.

4.1.2 Bolsa semántica y reducción de dimensionalidad

Para mostrar la eficiencia del esquema de clasificación implementado, se toma la base de datos de imágenes propia y se construye una bolsa semántica representativa, utilizando el descriptor VLAD. Para cada una de las imágenes, junto con los parches que la componen, se crea una firma contextual única y posteriormente se hace la respectiva clasificación según la escena. Se analizará el rendimiento del modelo con respecto a técnicas clásicas y actuales, así como el comportamiento en aplicaciones de conjuntos de imágenes proporcionadas en el estado del arte.

Inicialmente se utiliza el detector y descriptor semántico SSD 300, el cual una vez entrenado con la base de imágenes propia, arroja un total de 8732×4 cuadros predichos y 8732×51 conjuntos de score de probabilidad de la clase objetos;en esta fase de la metodología, el modelo no contempla la ubicación y el tamaño del recuadro debido a que se quiere explorar la semántica de los objetos para describir una escena, en ese sentido el impacto de las regiones en la imagen que sufren variación en los ángulos de visión se

regularizan considerando otros parches que tienen mejores condiciones de representación en el espacio semántico. Por tanto, se utilizan los resultados de clasificación que se obtienen en las diferentes imágenes para generar un espacio semántico euclidiano, para lo cual es necesario extraer la información de la última capa convolucional de la forma $H \times W \times D$. Mapa que puede considerarse como un conjunto de parches de D-dimensiones, extraídos en ubicaciones espaciales $H \times W$ que representan el ancho y alto de la imagen [28], una vez que se identifica los vectores en el modelo base se filtran los parches con mayor score siguiendo las métricas propuestas por el modelo, para este caso se establece en 0.2% de probabilidad [22].

De esta forma, se eligen los parches N con mayor probabilidad $P = [p_1, ..., p_i, ..., p_N]$, para un parche, se obtiene un vector de 51 dimensiones donde cada elemento del vector representa un objeto en particular. Cada imagen produce un conjunto de vectores semántico $S = [s_1, ..., s_i, ..., s_N]$, donde S_i es un vector del parche p_i, en este caso dónde la base de datos aumentada contiene 5000 imágenes genera un total de 20215 x51 parches a los cuales se le aplica un reductor de dimensionalidad LDA.

Por lo tanto, la matriz resultante será de 20215×20, nuestro objetivo es encontrar los centros de ***clúster*** para ese espacio semántico a través de una descripción VLAD. En principio se tendrán 51 ***clústers***, como criterio principal donde se asume tentativamente que se generan clústers con relación al número de objetos existentes en la base de datos. Formalmente, dada la descripción de características convolucionales local de parches como entrada y K centros de ***clúster*** como parámetros VLAD, la imagen de salida VLAD representa la posición V de $51 \times 20(K \times D)$; pero esta matriz se convierte en un vector después de la normalización, se utiliza como representación de la imagen por tanto se tendrán 5000 imágenes con un tamaño de $1 \times 51 \times 20$ las cuales contienen a la información semántica de todos los objetos que se encuentran en ella;

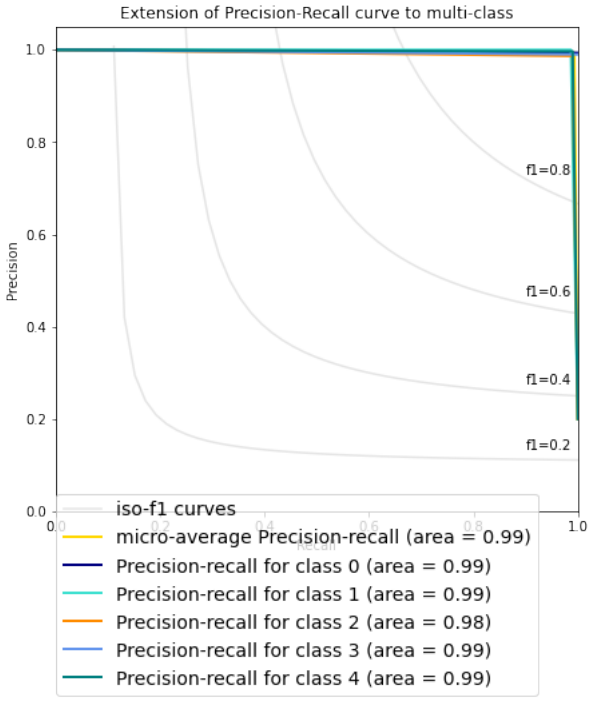
Se entrenan distintos clasificadores a partir de esta información para observar su comportamiento, juzgando en mayor medida el clasificador más simple ya que se quiere ver el comportamiento de la representación de la bolsa semántica. Antes de hacer el proceso de clasificación se parten los datos de entrenamiento(70 %) y validación(30 %), luego aplicando validación cruzada k-fold [91] se evalúan las métricas propuestas con anterioridad.Los parámetros del estimador que se utilizaron para aplicar estos métodos se optimizaron mediante la búsqueda del valor apropiado del número de cuadrículas.

Al evaluar el clasificador Bayesiano ingenuo [1] se confirman dos premisas importantes de este trabajo: (i) el enfoque propuesto puede aprender la semántica de los objetos a partir de una representación compacta más rica de la escena, y (ii) la idea de usar una arquitectura estándar que cumple un objetivo específico, ofrece resultados prometedores en el área de clasificación.

Tabla 4.5 – Métricas de evalución del clasificador Bayesiano ingenuo [1]

	Precision	**recall**	**f1-score**	**Imagnes**
Cocina	0.99	1.00	1.00	300
Sala 1	1.00	0.99	0.99	300
Sala 2	0.99	0.99	0.99	300
Pasillo 1	0.99	1.00	1.00	300
Pasillo 2	1.00	0.99	0.99	300
micro avg	0.99	0.99	0.99	1500
macro avg	0.99	0.99	0.99	1500
weighted avg	0.99	0.99	0.99	1500

Figura 4.4 – Extensión de la curva de Precision-Recall a múltiples clases para Bayesiano ingenuo [1]



Fuente: python

En la Tabla 4.5 se muestra diferentes métricas de evaluación micro avg, macro avg, weighted avg para los datos de validación de todas las clases de la base aumentada de imágenes propia obteniendo 1500 imágenes que corresponden a un 30% de la base total.

El resultado anterior se obtuvo a partir de un clasificador relativamente sencillo, a continuación se presentan los resultados obtenidos con otros tipos de clasificadores Tabla 4.6.

La Tabla 4.6 permite concluir que el clasificador KNN es el que presenta mejores resultados para la base de datos propia. El modelo final se encapsula junto con el clasificador KNN seleccionado, el cual se elige con 5 vecinos, para evaluar un total de

Tabla 4.6 – Métricas de los diferentes clasificadores

Clasificadores	**Accuracy(%)**
Gaussian Naive Bayes	0.99 (+/- 0.01)
SVM	0.99 (+/- 0.01)
SVM linear	0.67 (+/- 0.08)
Decision Tree	0.97 (+/- 0.02)
Logistic Regression	0.63 (+/- 0.08)
KNN	**1.00 (+/- 0.00)**

datos de entrenamiento de 5000 imágenes.

4.1.3 Análisis de rendimiento del método propuesto en comparación con técnicas convencionales cuando se aplican a la base de datos propia.

A continuación se compara el rendimiento del modelo propuesto con otras técnicas referenciadas en el estado del arte aplicadas a una misma base de datos. Para el modelo propuesto se utiliza el conjunto de imágenes de prueba escogido para mostrar la aplicabilidad del método implementado en escenas de interiores. Los resultados muestran que el método propuesto obtiene un mejor rendimiento, en comparación con los métodos convencionales referenciados en el estado del arte, los cuales sólo explotan la información visual de bajo nivel, obtenida de las imágenes como son la textura, color y estructura, mientras que el modelo propuesto se fundamenta en la información semántica de alto nivel construida a partir de una descripción más rica de los objetos que la componen(BOS).

Los métodos convencionales incluidas las CNN extraen principalmente las características de cada escena con independencia entre clases, además se enfocan en extraer información a nivel global sin tener en cuenta la información contextual relevante [20].

Tabla 4.7 – Métricas de evaluación de arquitecturas actuales y clásicas para la base de datos propia

	Accuracy %
Gist	52
Vlad	66
Fisher	72
BoVW	68
M-Propio(CNN)	87 (+/- 0.01)
M-Propio(BOS)	**99 (+/- 0.02)**

Los resultados del conjunto de datos propio con relación a la Tabla 4.8 y Tabla 4.7 donde muestra el desempeño ejecutado por la M-Propio(CNN) se basa en una arquitecturas

Tabla 4.8 – Métricas de evaluación de arquitecturas actuales y clásicas para la base de datos propia con relación a las clases C1 (Cocina), C2 (Sala1), C3 (Sala2), C4 (Pasillo 1), C5 (Pasillo 2)

	C1(acc%)	**C2(acc%)**	**C3(acc%)**	**C4(acc%)**	**C5(acc%)**
Gist	88	44	56	16	56
Vlad	84	76	68	8	100
Fisher	80	88	80	16	96
BoVW	100	72	48	44	76
M-Propio(CNN)	96	92	100	48	100
M-Propio(BOS)	100	100	96	100	97

ResNet- Places365 que se considera un extractor de características genéricas para imágenes de escena [20],[92].Dicho modelo se acondiciona al número de clases propia a través de 15 capas *fully connected*, ya que esta CNN se entrena en un conjunto de datos que contiene 365 clases. El conjunto de datos Places365-Challenge proporciona 8 millones de imágenes de entrenamiento, se utiliza para heredar características de una serie de escenarios de interiores que coinciden con la base de datos propia. En el experimento, se entrena a partir de un modelo ResNet para cada entorno con las escenas seleccionadas con base a los resultados mostrados en algunos artículos, [93], [94], [95], [96].

Según la Tabla 4.6 y Tabla 4.10 se hace un comparativo para un determinado conjunto de datos con clases específicas para mostrar la eficiencia del método Propio (CNN) implementado con respecto a diferentes bases de datos en el estado del arte.

Tabla 4.9 – Métricas de evaluación de escenas interiores en MIT indoor 67 [97] conjunto de datos

	DLBAISR[98](acc%)	**M-propio CNN(acc%)**
Kitchen	95.13	100
Bathroom	95.45	94
Bedroom	95.94	90
Living room	95.91	85
Mean	95.60	**92**

Los resultados experimentales logrados no superan el estado del arte en términos de exactitud.Sin embargo al comparar la arquitectura implementada con la arquitectura de detección de escenas a partir de objetos los resultados superan el estado del arte en clases seleccionadas de interiores.

En la Tabla 4.11 se muestran diferentes métricas de evaluación micro avg, macro avg, weighted avg para los datos de prueba del método propio(BOS) el cual se compone 625 imágenes, donde 125 imágenes corresponden a cada clase. De estas 100 son entrenamiento y 25 prueba.

Tabla 4.10 – Métricas de evaluación de escenas interiores en scene 15 conjunto de datos

	Shao[99](acc%)	**DLBAISR[98](acc%)**	**M-propio CNN(acc%)**
Bedroom	97	98	91.5 (+/- 0.173)
Kitchen	71	95	96.5 (+/- 0.224)
Living room	98	99	93.75 (+/- 0.433)
Mean	96.49	97	**93.75 (+/- 0.260)**

Tabla 4.11 – Métricas de evalución y pruebas base propia(BOS)

	Precision	**recall**	**f1-score**	**Imágenes**
clase 1	1.00	1.00	1.00	25
clase 2	1.00	1.00	1.00	25
clase 3	0.96	1.00	0.98	25
clase 4	1.00	1.00	1.00	25
clase 5	1.00	0.96	0.98	25
micro avg	0.99	0.99	0.99	125
macro avg	0.99	0.99	0.99	125
weighted avg	0.99	0.99	0.99	125

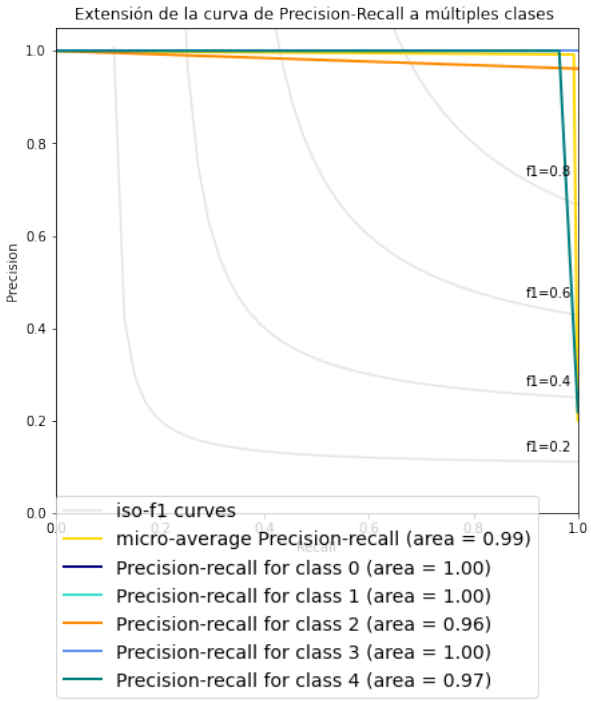
4.1.4 Comparación con el Estado del Arte

Se evalúa el modelo propuesto para el conjunto de datos UIUC Sports-8: Este conjunto de datos [74] contiene 1572 imágenes a color con 8 categorías diferentes, incluidas varias escenas de eventos deportivos. El número de imágenes para cada categoría varía de 130 a 250. Estas imágenes tienen resoluciones altas (800 x 600 píxeles). Tampoco proporciona conjuntos de entrenamiento y pruebas por separado. Siguiendo el protocolo definido en [74], se muestrean 70 imágenes al azar para entrenamiento y 60 imágenes para validación, las cuales son muestreadas al azar de las imágenes restantes de cada categoría. Esto se hace de tal forma que se pueda garantizar que todas las imágenes de la clase entran al sorteo de selección, generando 3 tipos de conjuntos de prueba según lo indica el protocolo.

En esta base se etiquetan 21 objetos que siguen el mismo procedimiento expuesto anteriormente. En este caso los híper parámetros se establecen de la siguiente forma: Para los clúster se definen en 5 y 17 dimensiones para el LDA y se elige el mismo clasificador. Las medidas de rendimiento del método propuesto (modelo-propio) y algunos métodos convencionales descritos en estado del arte se muestran en la Tabla 4.12.

Aunque el modelo propuesto no superó algunos de los métodos convencionales descritos en el estado del arte, el desempeño de éste fue muy bueno, pues la base de imágenes utilizada (UIUC Sports-8) centra su fuente de descripción en las diferentes acciones que efectúa el jugador en los diferentes deportes. Asimismo, contiene parches pequeños que afectan la bolsa semántica quitándole información descriptiva a la hora de

Figura 4.5 – Extensión de la curva de Precision-Recall a múltiples clases pruebas



Fuente: python

evaluarla, mientras que el modelo propuesto tiene un mejor desempeño para aquellas bases de imágenes donde la riqueza semántica es abúndate y descriptiva para sus objetos.

Es importante resaltar que el método propuesto centra su eficiencia en una descripción local, sin tener en cuenta la descripción global. Por consiguiente, una combinación de arquitecturas que combine la descripción global y la descripción local podría generar un sistema más compacto que mejore la eficiencia del mejor clasificador relacionado en el estado del arte.

Dado que el modelo propuesto, tiene su mejor desempeño en la descripción de entornos de interiores ricos en contexto, con alta discriminación semántica en las escenas a partir de sus objetos, se pone a prueba con esta base de datos de deportes UIUC Sports-8, en su estado más crítico sin poseer muchas de las características para las cuales fue creado y aun así tiene un desempeño competitivo en el estado del arte.

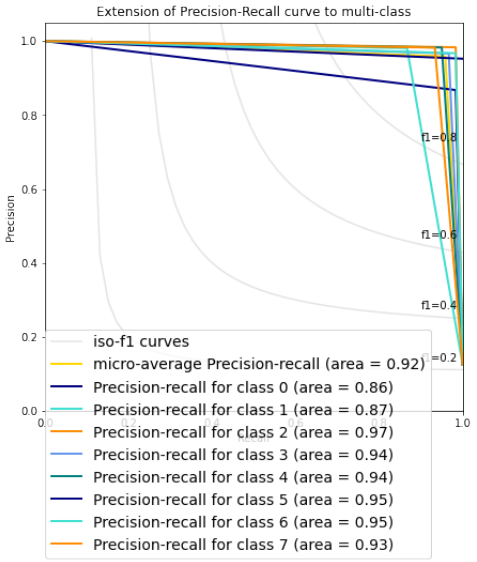
4.1.5 Bolsa semántica con parámetro de control

En esta sección se describe la arquitectura basada a partir del VLAD con un parámetro extra es decir dada la descripción de las características de los parches V y K centros de grupos ("palabras visuales") c_k. Intuitivamente, $a_k(x_i)$ es un fator de control que denota la pertenencia del descriptor xi a la k-th palabra visual permitiendo una asignación

Tabla 4.12 – Comparación del método propuesto con otros métodos

	Accuracy (%)
Object Bank [33]	76.3
Object Attributes[100]	77.9
CENTRIST [101]	78.2
RSP [102]	79.6
SPM [11]	81.8
SPMSM [103]	83.0
HIK [104]	84.2
LScSPM[105]	85.3
LCSR [106]	87.2
ISPR[58]	89.5
IFV[56]	90.8
NNSD + ICLC [19]	99.3
ImageNet-CNN [72]	94
SDCF [107]	98.5
DUCA [108]	98.7
GFG-NaïveB[109]	85.09
Modelo-propio(BOS)	94 (+/- 0.3)

Figura 4.6 – Comparación del método propuesto con otros métodos convencionales descritos en el estado del arte Conjunto 1.

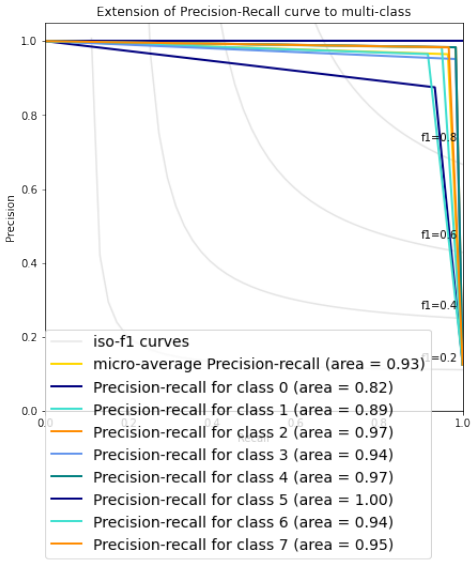


Fuente: python

suave al espacio ponderado que oscila entre 0 y 1, con el peso más alto asignado al centro de clúster más cercano(ver sección 3.4.1 para más detalles). Este vector de penalización se analiza según el resultado propuesto para el conjunto de datos propio.

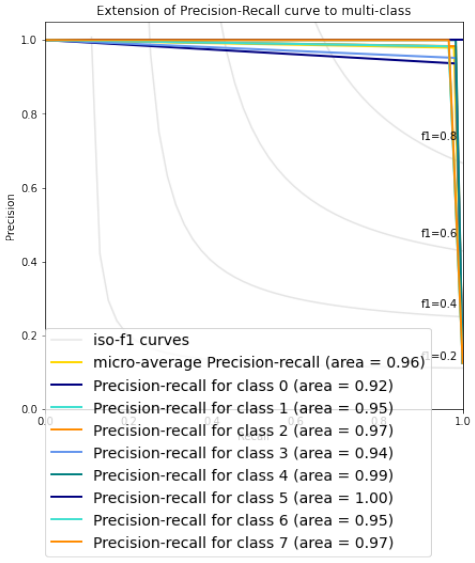
En la Tabla 4.13 se muestran diferentes métricas de evaluación micro avg, macro avg,weighted avg para los datos de prueba de todas las clases de la base de datos propia

Figura 4.7 – Comparación del método propuesto con otros métodos convencionales descritos en el estado del arte Conjunto 2.



Fuente: python

Figura 4.8 – Comparación del método propuesto con otros métodos convencionales descritos en el estado del arte Conjunto 3.



Fuente: python

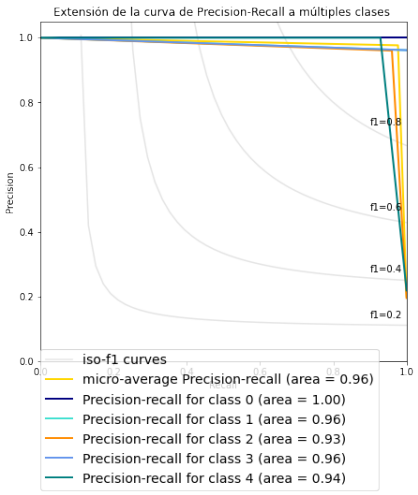
obteniendo 125 imágenes guardada para el primer conjunto evaluadas en el modelo propio(BOS).

Aunque no supera el modelo más robusto tiene potencial para transformase en una capa del modelo base haciendo mas efectivo el entrenamiento, es decir la forma en que se escribió la ecuación de bolsa semántica permite que sea derivable logrando converger con la fusión de costo [28].

Tabla 4.13 – Métricas de evalución pruebas parámetro de control

	Precision	**recall**	**f1-score**	**Imágenes**
clase 1	1.00	1.00	1.00	25
clase 2	0.96	1.00	0.98	25
clase 3	0.96	0.96	0.96	25
clase 4	0.96	1.00	0.98	25
clase 5	1.00	0.93	0.96	25
micro avg	0.98	0.98	0.98	125
macro avg	0.98	0.98	0.98	125
weighted avg	0.98	0.98	0.98	125

Figura 4.9 – Extensión de la curva de Precision-Recall a múltiples clases parámetro de control



Fuente: python

4.1.6 Análisis de caso de falla

A continuación se describen algunos ejemplos de fallas, percibidos al utilizar el modelo propuesto con la base de imágenes de datos de deportes UIUC Sports-8.

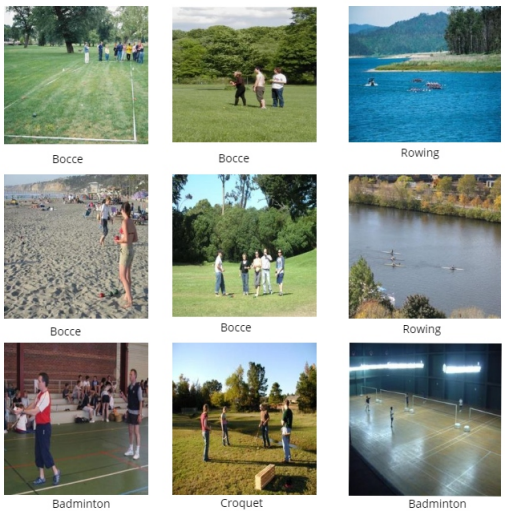
En la siguiente figura se puede notar que algunas categorías de escenas se confunden fácilmente con los demás debido a:

- Espacios similares.
- Objetos no detectados o de poco contexto.

Por ejemplo, los contenidos visuales de algunas clases proporcionan poca información ya que la acción ejecutada por las personas carece de contexto para el juego, este es el caso de sujetos que solo son espectadores. A su vez los objetos pequeños o no visibles que son ricos en semántica pero no fácilmente detectables, por tanto se hace aun más complejo

crear una descripción compacta a través de ambigüedades forzadas por la mala detección, haciendo que el reconocimiento de escenas siga siendo un problema difícil de abordar.

Figura 4.10 – Caso de fallas para la base UIUC Sports-8



5 CONCLUSIONES

En esta tesis se diseño una metodología que codifica la información del detector de objetos SSD-300 en un espacio semántico denso, obteniendo una representación única a través del VLAD, el cual representa la información contextual de los objetos de las imágenes, construyendo una firma global para el posterior reconocimiento de escenas. Los resultados obtenidos muestran la efectividad de la metodología, en el reconocimiento de escenas superando los registros del estado del arte. Usando esta metodología, los robots pueden reconocer diferentes lugares interiores con alta confianza y precisión.

Se verificó con la base de imágenes propia, que el rendimiento de la metodología en ambientes interiores es superior al registro del estado del arte.

Se verificó que la metodología propuesta mejora la cohesión semántica de los objetos en la escena, gracias a que con el aumento de la base de datos, el nivel de contexto de los objetos en las escenas mejora potencialmente el proceso de clasificación. El método propuesto se aplicó a una base de imágenes UIUC Sports-8 del estado del arte, mostrando muy buenos resultados en la clasificación de escenas.

Se concluye que el tamaño de algunos objetos menores a 50 pixeles, generan confusión en la información que codifica el espacio semántico, es decir que se requiere generar información contextual más profunda al describir parches pequeños.

Otro problema relacionado, son los objetos comunes cuya información genera confusión en el proceso de detección; una manera de abordar este problema, es combinar los parches que son extensiones de contexto de la escena, es decir objetos con información común entre parches de su misma clase. Por ejemplo, el objeto tabla comprende una acción conjunta entre el objeto tabla y el jugador que practica el snowboarding .

También existen algunos objetos que son independientes y carecen de contexto conjunto, pero que son relevantes para la clase y por ello también deben ser parte de la descripción. Lo anterior afirma que los resultados experimentales pueden mejorar potencialmente. También se observa que, el modelo de clasificación es suficiente para representar con precisión el conjunto de datos de interiores de la base de datos propia, sin modificar la composición jerárquica de las arquitecturas propuestas.

La metodología implementada tiene su campo de acción en el reconocimiento de escenarios para ambientes y entornos de trabajo en los que los sistemas robotizados puedan interactuar y cumplir funciones específicas, por ejemplo un campo de acción de dicha metodología sería el reconocimiento de entornos interiores por parte de un robot en una sala de museo, o la ubicación de herramientas para el abastecimiento de máquinas CNC

para la elaboración de equipos.

6 TRABAJO FUTURO

Como resultado del trabajo realizado, se considera que una mejora a realizar en el futuro podría orientarse al diseño de un método de detección de objetos pequeños o en su defecto mejorar las prestaciones del detector de objetos utilizado.

Generar o utilizar un método más práctico de etiquetado de los objetos en las imágenes que permita hacer que esta labor sea menos tediosa y laboriosa.

Lograr encapsular el método de tal forma que no se requiera entrenar cada sección por separado aprovechando la minimización de la función de costo para acoplarlos

Implementar una arquitectura que vincule tanto la descripción local como la descripción global de la imagen para el reconocimiento de escenas.

Comprobar el rendimiento de otras arquitecturas presentes en el estado del arte para el mismo propósito de la tesis implementada tales como detención de objetos con Redes Neuronales Generativas Adversarias junto con un espacio de agrupación de variables latentes.

REFERENCIAS

[1] Robert; FRIEDMAN Jerome HASTIE, Trevor; TIBSHIRANI. The elements of statistical learning: data mining, inference, and prediction. *Springer Science Business Media*, 2009. Citado 6 , 7, 11, 39, 40, 52 y 53.

[2] Lage-Castellanos A. Valente G. Goebel R. Valdes-Sosa M Ontivero-Ortega, M. Fast gaussian naïve bayes for searchlight classification analysis. *Neuroimage*, 163:471–479, 2017. Citado en la página 11.

[3] Ontivero-Ortega M. Iglesias-Fuster J. Lage-Castellanos A.-Gong J. Luo C. ... Yao D. Valdés-Sosa, M. Objects seen as scenes: neural circuitry for attending whole or parts. *Neuroimage*, page 116526, 2020. Citado en la página 11.

[4] George amp. *Objects in Relation for Scene Understanding*. DOCTOR OF SCIENCES of ETH ZURICH, 2016. Citado 2 , 11 y 14.

[5] P. J. LAULKAR, C. A.; KULKARNI. Semantic rules-based classification of outdoor natural scene images. *En Computing in Engineering and Technology.Springer, Singapore,*, pages 543–555, 2020. Citado 2 , 11 y 12.

[6] Dixit M. Zogg G. Vasconcelos N. George, M. Semantic clustering for robust fine-grained scene recognition. *En European Conference on Computer Vision. Springer, Cham*, pages 783–798, 2016. Citado en la página 11.

[7] Li W. Liu J. Han G. Wu-C Sun, N. Fusing object semantics and deep appearance features for scene recognition. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(6):1715–1728, 2018. Citado 4 , 11, 12, 14 y 15.

[8] Anil; ZHANG Hong Jiang VAILAYA, Aditya; JAIN. On image classification: City images vs. landscapes. *Pattern recognition*, 31(12):1921–1935, 1998. Citado en la página 11.

[9] Hanli; KWONG Sam TANG, Pengjie; WANG. G-ms2f: Googlenet based multi-stage feature fusion of deep cnn for scene recognition. *Neurocomputing*, 225:188–197, 2017. Citado en la página 11.

[10] Andreas E.; LUO Jiebo SERRANO, Navid; SAVAKIS. Improved scene classification using efficient low-level features and semantic cues. *Pattern Recognition*, 37(9):1773–1784, 2004. Citado en la página 11.

[11] Cordelia; PONCE Jean LAZEBNIK, Svetlana; SCHMID. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. *En 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06). IEEE*, pages 2169–2178, 2006. Citado 3 , 11, 23 y 58.

[12] Andrew SIVIC, Josef; ZISSERMAN. Video google: A text retrieval approach to object matching in videoss. *En null. IEEE*, pages 1470–1477, 2003. Citado 2 , 11 y 17.

[13] David G LOWE. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. Citado en la página 11.

[14] Matti; MAENPAA Topi OJALA, Timo; PIETIKAINEN. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on pattern analysis and machine intelligence*, 24(7):971–987, 2002. Citado en la página 11.

[15] Bill DALAL, Navneet; TRIGGS. Histograms of oriented gradients for human detection. *En 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). IEEE*, pages 886–893, 2005. Citado en la página 11.

[16] Michal SHECHTMAN, Eli; IRANI. Matching local self-similarities across images and videos. en 2007 ieee conference on computer vision and pattern recognition. *IEEE*, pages 1–8, 2007. Citado en la página 11.

[17] Tae; MU LEE Kyoung. NAH, Seungjun; HYUN KIM. Deep multi-scale convolutional neural network for dynamic scene deblurring. *En Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3883–3891, 2017. Citado en la página 11.

[18] Ilya; HINTON Geoffrey E KRIZHEVSKY, Alex; SUTSKEVER. Imagenet classification with deep convolutional neural networks. *En Advances in neural information processing systems*, pages 1097–1105, 2012. Citado 3 , 11, 14 y 18.

[19] Lee F. Liu L. Yin Z. Chen-Q Xie, L. Hierarchical coding of convolutional features for scene recognition. *IEEE Transactions on Multimedia*, 2019. Citado 3 , 12, 14 y 58.

[20] Lu J. Feng J. Yuan B. Zhou-J Cheng, X. Scene recognition with objectness. *Pattern Recognition*, 74:474–487, 2018. Citado 7 , 12, 15, 19, 20, 21, 54 y 55.

[21] Dong W. Socher R. Li L. J. Li-K. Fei-Fei L Deng, J. Imagenet: A large-scale hierarchical image database. *En 2009 IEEE conference on computer vision and pattern recognition*, pages 248–255, 2009. Citado en la página 12.

[22] Sagar Vinodababu. Ssd: Single shot multibox detector | a pytorch tutorial to object detectionn. `https://github.com/sgrvinod/a-PyTorch-Tutorial-to-Object-Detection`, 2020. Citado 5 , 12, 29, 32, 50 y 52.

[23] Andrew ARANDJELOVIC, Relja; ZISSERMAN. All about vlad. *En Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 1578–1585, 2013. Citado 2 , 12 y 20.

[24] Lee F. Liu L. Kotani K. Chen-Q Xie, L. Scene recognition: A comprehensive survey. *Pattern Recognition*, page 107205, 2020. Citado 6 , 12, 14, 18, 19, 22 y 51.

[25] Gita; SUKTHANKAR Rahul ALABACHI, Saif; SUKTHANKAR. Customizing object detectors for indoor robots. *En 2019 International Conference on Robotics and Automation (ICRA). IEEE*, pages 8318–8324, 2019. Citado 2 , 14 y 28.

[26] Yang Y. Mao R. Fermüller C. Aloimonos-Y. Ye, C. What can i do around here? deep functional scene understanding for cognitive robots. *En 2017 IEEE International Conference on Robotics and Automation (ICRA). IEEE*, pages 4604–4611, 2017. Citado en la página 14.

[27] E. Ricci M. Mancini, S. R. Bulo and B. Caputo. Learning deep nbnn representations for robust place categorization. *IEEE Robot. Autom. Lett.*, 2(3):1794–1801, 2017. Citado en la página 14.

[28] Gronat P. Torii A. Pajdla T. Sivic-J Arandjelovic, R. Netvlad: Cnn architecture for weakly supervised place recognition. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016. Citado 4 , 15, 38, 52 y 59.

[29] et al SÁNCHEZ, Jorge. Image classification with the fisher vector: Theory and practice. *International journal of computer vision*, 105(3):222–245, 2013. Citado 2 , 15 y 18.

[30] Tzutalin. Labelimg free software: Mit license. `https://github.com/tzutalin/labelImg`, 2015. Citado 2 , 16 y 26.

[31] et al REDMON, Joseph. You only look once: Unified, real-time object detection. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–7885, 2016. Citado 2 , 16 y 18.

[32] Suraj Venkat. Best image labeling tools for computer vision. *Medium,https://medium.com/tektorch-ai/best-image-labeling-tools-for-computer-vision-393e256be0a0*, 2019. Citado 2 , 16 y 17.

[33] E. Xing L. Li, H. Su and L. Fei-Fei. Object bank: A high-level image representation for scene classification semantic feature sparsification. *In Advances in neural information processing systems*, pages 1378–1386, 2010. Citado 3 , 17, 19 y 58.

[34] Jiang Y. G. Hauptmann A. G. Ngo-C. W. Yang, J. Evaluating bag-of-visual-words representations in scene classification. *En Proceedings of the international workshop on Workshop on multimedia information retrieval*, pages 197–206, 2007. Citado en la página 17.

[35] Wang L. Wang Y. Zhang B. Qiao-Y Wang, Z. Weakly supervised patchnets: Describing and aggregating local patches for scene recognition. *IEEE Transactions on Image Processing*, 26(4):2028–2041, 2017. Citado 3 , 17, 26 y 37.

[36] Dance C. Fan L. Willamowski J. Bray-C. Csurka, G. Visual categorization with bags of keypoints. *En Workshop on statistical learning in computer vision, ECCV*, pages 1–2, 2007. Citado en la página 17.

[37] Antonio OLIVA, Aude; TORRALBA. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International journal of computer vision*, 42(3):145–175, 2001. Citado en la página 18.

[38] Liu Y. Sánchez J. Poirier H. Perronnin, F. Large-scale image retrieval with compressed fisher vectors. *En 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE*, 42(3):3384–3391, 2010. Citado en la página 18.

[39] T. Zhang X. Zhou, K. Yu and T. S. Huang. Image classification using super-vector coding of local image descriptors,. *in ECCV*, page 141–154, 2010. Citado en la página 18.

[40] K. Yu F. Lv T. S. Huang J. Wang, J. Yang and Y. Gong. Locality- constrained linear coding for image classification,. *in CVPR*, page 3360–3367, 2010. Citado en la página 18.

[41] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition,. *arXiv preprint arXiv:1409.1556*, 2014. Citado en la página 18.

[42] Y. Jia P. Sermanet S. Reed D. Anguelov D. Erhan V. Vanhoucke C. Szegedy, W. Liu and A. Rabinovich. Going deeper with convolutions. *in Proceedings of the IEEE conference on computer vision and pattern recognition*, page 1–9, 2015. Citado en la página 18.

[43] A. Khosla A. Oliva B. Zhou, A. Lapedriza and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE Transac- tions on Pattern Analysis and Machine Intelligence*, 2017. Citado en la página 18.

[44] M. Hayat S. H. Khan and F. Porikli. Scene categorization with spectral features. *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 5638–5648, 2017. Citado en la página 18.

[45] Donahue J. Darrell T. Malik J. Girshick, R. Rich feature hierarchies for accurate object detection and semantic segmentation. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. Citado en la página 18.

[46] Ross GIRSHICK. Fast r-cnn. *En Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. Citado en la página 18.

[47] Rathod V. Sun C. Zhu M. Korattikara A. Fathi A.Murphy K Huang, J. Speed/accuracy trade-offs for modern convolutional object detectors. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7310–7311, 2017. Citado 2 , 18 y 28.

[48] L. Fei-Fei L. Cao. Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes. *in:Proceedings of the IEEE International Conference on Computer Vision*, pages 1–8, 2007. Citado en la página 19.

[49] X. Gao Z. Niu, G. Hua. Context aware topic model for scene recognition. *in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 2743–2750, 2012. Citado en la página 19.

[50] P.F. Felzenszwalb S.N. Parizi, J.G. Oberlin. Reconfigurable models for scene recognition. *in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, page 2775–2782, 2012. Citado en la página 19.

[51] L. Herranz X. Song, S. Jiang. Multi-scale multi-feature context modeling for scene recognition in the semantic manifold. *IEEE Transactions on Image Processing*, 26(6):2721–2735, 2017. Citado en la página 19.

[52] W. Wang Y. Yu R. Wu, B. Wang. Harvesting discriminative meta objects with deep cnn features for scene classification. *in:Proceedings of the IEEE International Conference on Computer Vision*, pages 1287–1295, 2017. Citado en la página 19.

[53] X. Song S. Jiang L. Herranz Y. Kong and K. Zheng category co-occurrence modeling for large scale scene recognition. *PatternRecognition 59*, pages 98–111, 2016. Citado en la página 19.

[54] S. Lazebnik M. Pandey. Scene recognition and weakly supervised object localization with deformable part-based models. *in: Proceedings of the IEEE International Conference on Computer Vision*, pages 1307–1314, 2011. Citado en la página 19.

[55] A. A. Efros S. Singh, A. Gupta. Unsupervised discovery of midlevel discriminative patches. *in: Proceedings of the European Conference on Computer Vision*, pages 73–86., 2012. Citado en la página 19.

[56] C. V. Jawahar A. Zisserman M. Juneja, A. Vedaldi. Blocks that shout: distinctive parts for scene classification. *in: Proceedings of 44 the IEEE Conference on Computer Vision and Pattern Recognition*, pages 923–930., 2013. Citado 2 , 19 y 58.

[57] Q. Wang Y. Yuan, J. Wan. Congested scene classification via efficient unsupervised feature learning and density estimation. *Pattern Recognition*, 56:159–169., 2016. Citado en la página 19.

[58] R. Liao D. Lin, C. Lu and . J. Jia. Learning important spatial pooling regions for scene classification. *En Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3726–3733, 2014. Citado 2 , 19 y 58.

[59] B. Shuai L. Zhao Q. Yang X. Jiang Z. Zuo, G. Wang. Learning discriminative and shareable features for scene classification. *in: Proceedings of the European Conference on Computer Vision*, pages pp. 552–568, 2014. Citado en la página 19.

[60] B. Shuai L. Zhao Q. Yang X. Jiang Z. Zuo, G. Wang. Learning discriminative and shareable features for scene classification. *in: Proceedings of the European Conference on Computer Vision*, pages pp. 552–568, 2014. Citado en la página 19.

[61] W. Chen I. Wassell Y. Liu, Q. Chen. Dictionary learning inspired deep network for scene recognition. *in: Proceedings of AAAI Conference on Artificial Intelligence*, pages pp. 7178–7185, 2018. Citado en la página 19.

[62] A. Lapedriza A. Oliva B. Zhou, A. Khosla and A. Torralba. Learning deep features for discriminative localization. *in Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, 2016. Citado en la página 19.

[63] S. Yu W. Wu J. Shi, H. Zhu and H. Shi. Scene categorization model using deep visually sensitive features. *IEEE Access*, 7:45230–45239, 2019. Citado en la página 19.

[64] Anguelov D. Erhan D. Szegedy C. Reed S. Fu C. Y. Berg A. C Liu, W. Ssd: Single shot multibox detector. *En European conference on computer vision. Springer, Cham*, pages 21–37, 2016. Citado 4 , 20, 28, 31 y 32.

[65] Sutreja A. Kumar A. Taneja S. Regunathan R Thakkar, Y. Efficient parking system using single-shot multibox detector. *En Data Engineering and Communication Technology. Springer, Singapore*, pages 931–939, 2020. Citado 2 , 20 y 28.

[66] Douze M. Schmid C. Pérez P. Jégou, H. Aggregating local descriptors into a compact image representation. *En 2010 IEEE computer society conference on computer vision and pattern recognition. IEEE*, pages 3304–3311, 2010. Citado 2 , 20 y 38.

[67] Leonardo Chang VILLASENOR, Miriam Mónica Duarte; FERNÁNDEZ. Clasificación de objetos en imágenes usando sift. *Proyecto del Curso Modelos Gráficos Probabilistas y sus aplicaciones Maestría en Ciencias Computacionales, INAOE*, 2015. Citado en la página 21.

[68] Giovani Gómez L. Enrique Sucar. Visión computacional. *Instituto Nacional de Astrofísica, Óptica y Electrónica Puebla, México*, 2019. Citado en la página 21.

[69] Chen S. Gao D. Rasiwasia N. Vasconcelos N. Dixit, M. Scene classification with semantic fisher vectors. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2974–2983, 2015. Citado en la página 21.

[70] A. Torralba A. Quattoni. Recognizing indoor scenes. *Proceedings of the IEEE International Conference on Computer Vision*, pages 413–420, 2009. Citado en la página 23.

[71] K.A. Ehinger A. Oliva A. Torralba J. Xiao, J. Hays. Sun database: large-scale scene recognition from abbey to zoo. *Proceedings of the IEEE International Conference on Computer Vision*, pages 3485–3492, 2010. Citado en la página 23.

[72] J. Xiao A. Torralba A. Oliva B. Zhou, A. Lapedriza. Learning deep features for scene recognition using places database. *Proceedings of the Advances in Neural Information Processing Systems*, pages 487–495, 2014. Citado 2 , 23 y 58.

[73] C. K. I. Williams J. Winn M. Everingham, L. Van Gool and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010. Citado 2 , 23 y 24.

[74] Li LI, Li-Jia; FEI-FEI. What, where and who? classifying events by scene and object recognition. *En 2007 IEEE 11th international conference on computer vision. IEEE*, pages 1–8, 2007. Citado 4 , 24, 25, 26 y 56.

[75] Lucia Ayestaran Garralda. Localización de objetos en imágenes mediante técnicas de aprendizaje profundo. *Tesis Fin de grado.Universidad De La Rioja*, 2018. Citado en la página 26.

[76] Zhu X.-Lei Z. Shi H. Wang X. Li S. Z Zhang, S. S3fd: Single shot scale-invariant face detector. *En Proceedings of the IEEE International Conference on Computer Visions*, pages 192–201, 2017. Citado en la página 28.

[77] Ying LENG, Jiaxu; LIU. An enhanced ssd with feature fusion and visual reasoning for object detection. *Neural Computing and Applications*, 31(10):6549–6558, 2019. Citado en la página 28.

[78] Li Z.-Fang L. Zhang T Zhao, H. A balanced feature fusion ssd for object detection. *Neural Processing Letters*, pages 1–18, 2020. Citado en la página 28.

[79] Szegedy C.-Toshev A. Anguelov D. Erhan, D. Scalable object detection using deep neural networks. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2147–2154, 2014. Citado en la página 33.

[80] Reed S.-Erhan D. Anguelov D Szegedy, C. Scalable, high-quality object detection. *arXiv preprint arXiv:1412.1441*, 2015. Citado en la página 33.

[81] Li Y.-Yan C. Dai H. Liu G Chen, C. A robust algorithm of multiquadric method based on an improved huber loss function for interpolating remote-sensing-derived elevation data sets. *Remote Sensing*, 7(3):3347–3371, 2015. Citado en la página 35.

[82] Taylor R HENDERSON, John M.; HAYES. Meaning-based guidance of attention in scenes as revealed by meaning maps. *Nature Human Behaviour*, 1(10):743–747, 2017. Citado 2 , 41 y 51.

[83] et al HENDERSON, John M. Meaning and attentional guidance in scenes: A review of the meaning map approach. *Vision*, 3(2):19, 2019. Citado en la página 41.

[84] Farahnaz Ahmed; POMPLUN Marc. WU, Chia-Chien; WICK. Guidance of visual attention by semantic information in real-world scenes. *Frontiers in psychology*, 5:54, 2014. Citado en la página 41.

[85] Brian; D'ARCY Aoife KELLEHER, John D.; MAC NAMEE. Fundamentals of machine learning for predictive data analytics: algorithms, worked examples, and case studies. *MIT press*, 2015. Citado en la página 42.

[86] Jason H URBANOWICZ, Ryan J.; MOORE. Exstracs 2.0: description and evaluation of a scalable learning classifier system. *Evolutionary intelligence*, 8(2-3):89–116, 2015. Citado 2 , 42 y 44.

[87] A. Rosebrock. Intersection over union (iou) for object detection. `https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/`, 2016. Citado en la página 42.

[88] Yunsheng; VASCONCELOS Nuno DIXIT, Mandar; LI. Semantic fisher scores for task transfer: Using objects to classify scenes. *IEEE transactions on pattern analysis and machine intelligence*, 2019. Citado en la página 46.

[89] Chen Z. Xu A. Wang X. Liang X. Lin L. Yan, X. Meta r-cnn: Towards general solver for instance-level low-shot learning. *En Proceedings of the IEEE International Conference on Computer Vision*, pages 9577–9586, 2019. Citado en la página 46.

[90] Deva HU, Peiyun; RAMANAN. Finding tiny faces. *En Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 951–959, 2017. Citado en la página 51.

[91] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Royal Statistical Soc. Series B*, 36:111–147, 1974. Citado en la página 52.

[92] Taghi M.; WANG DingDing. WEISS, Karl; KHOSHGOFTAAR. survey of transfer learning. journal of big data. *Journal of Big data*, 3(1):9, 2016. Citado en la página 55.

[93] et al. WANG, Shuo. Compressed holistic convolutional neural network-based descriptors for scene recognition. *En 2019 4th International Conference on Robotics and Automation Engineering (ICRAE). IEEE*, pages 135–139, 2019. Citado en la página 55.

[94] Sahdev R. Wu D. Zhao X. Papagelis M. Tsotsos J. K. Chen, B. X. Scene classification in indoor environments for robots using context based word embeddings. *arXiv preprint arXiv:1908.06422*, 2019. Citado en la página 55.

[95] et al. PEREIRA, Ricardo. Deep-learning based global and semantic feature fusion for indoor scene classification. *En 2020 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC). IEEE*, pages 67–73, 2020. Citado en la página 55.

[96] et al ZHAO, Zhengyu. Reproducible experiments on adaptive discriminative region discovery for scene recognition. *En Proceedings of the 27th ACM International Conference on Multimedia*, pages 1076–1079, 2019. Citado en la página 55.

[97] A.Torralba A. Quattoni. Recognizing indoor scenes. *Proceedings of the IEEE International Conference on Computer Vision*, pages 413–420, 2009. Citado en la página 55.

[98] et al. AFIF, Mouna. Deep learning based application for indoor scene recognition. *Neural Processing Letters*, pages 1–11, 2020. Citado 2 , 55 y 56.

[99] Guohui LIU, Shaopeng; TIAN. An indoor scene classification method for service robot based on cnn feature. *Journal of Robotics*, 40(2):188–194, 2019. Citado en la página 56.

[100] Su H. Lim Y. Fei-Fei L Li, L. J. Objects as attributes for scene classification. en european conference on computer vision. *Springer, Berlin, Heidelberg*, pages 57–69, 2010. Citado en la página 58.

[101] Jim M WU, Jianxin; REHG. Centrist: A visual descriptor for scene categorization. *IEEE transactions on pattern analysis and machine intelligence*, 33(8):1489–1501, 2010. Citado en la página 58.

[102] Junsong; YU Gang JIANG, Yuning; YUAN. Randomized spatial partition for scene recognition. en european conference on computer vision. *Springer, Berlin, Heidelberg*, pages 730–743, 2012. Citado en la página 58.

[103] Nuno; RASIWASIA Nikhil KWITT, Roland; VASCONCELOS. Scene recognition on the semantic manifold. *En European Conference on Computer Vision. Springer, Berlin, Heidelberg*, pages 359–372, 2012. Citado en la página 58.

[104] James M WU, Jianxin; REHG. Beyond the euclidean distance: Creating effective visual codebooks using the histogram intersection kernel. *En 2009 IEEE 12th International Conference on Computer Vision. IEEE*, pages 630–637, 2009. Citado en la página 58.

[105] Tsang I. W. H. Chia L. T.- Zhao P Gao, S. Local features are not lonely–laplacian sparse coding for image classification. *En 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. IEEE*, pages 3555–3561, 2010. Citado en la página 58.

[106] A. Shabou and H. LeBorgne. Locality-constrained and spatially regularized coding for scene categorization. *En 2012 IEEE Conference on Computer Vision and Pattern Recognition. IEEE*, pages 3618–3625, 2012. Citado en la página 58.

[107] Y. Yan L. Xie, F. Lee and Q. Chen. Sparse decomposition of convolutional features for scene recognition. *En 2017 2nd IEEE International Conference on Computational Intelligence and Applications (ICCIA). IEEE*, pages 345–348, 2017. Citado en la página 58.

[108] M. Bennamoun R. Togneri S.H. Khan, M. Hayat and F.A. Sohel. A discriminative representation of convolutional features for indoor scene recognition. *IEEE Transactions on Image Processing*, 25(7):3372–3383, 2016. Citado en la página 58.

[109] Ahmad; AHMED Abrar RAFIQUE, Adnan Ahmed; JALAL. Scene understanding and recognition: Statistical segmented model using geometrical features and gaussian naïve bayes. *En IEEE conference on International Conference on Applied and Engineering Mathematics*, 2019. Citado en la página 58.