UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escola Tècnica Superior d'Enginyeria Informàtica

AViTag Automatic Video Tagger

Treball Fi de Grau

Grau en Enginyeria Informàtica

AUTOR/A: Diez Lambies, Iñaki

Tutor/a: Martínez Hinarejos, Carlos David

Cotutor/a extern: PEREZ FENOLLOSA, OLEGARIO

CURS ACADÈMIC: 2022/2023

Resum

El treball de final de grau que es proposa té com a objectiu solucionar un gran problema que afecta als tècnics d'il·luminació: la dificultat de mantindre ordenades i accessibles grans quantitats de continguts de vídeo per a la seua feina. La cerca manual entre aquests continguts cada volta que volen realitzar alguna tasca suposa una pèrdua de temps i recursos que afecta negativament el seu rendiment.

Per a solucionar aquest problema, es proposa el desenvolupament d'una aplicació d'escriptori multiplataforma que utilitza tècniques d'anàlisi estadístic, visió per ordinador i intel·ligència artificial per a la classificació de vídeos. A més a més, l'aplicació proporcionarà una interfície d'usuari amigable on l'usuari podrà cercar els seus vídeos classificats.

L'arquitectura de l'aplicació constarà d'una interfície i un motor de classificació que realitzarà les operacions en paral·lel a través de diferents tècniques de repartiment de tasques entre processos per a una millor eficiència en el tractament de les dades.

El treball inclourà una investigació a fons sobre les tècniques d'anàlisi estadístic, visió per ordinador i intel·ligència artificial per a la classificació de vídeos, així com el disseny i desenvolupament d'un primer prototip d'aquesta aplicació. Es realitzarà una avaluació exhaustiva del rendiment del prototip i es proposaran possibles millores.

Aquesta aplicació representa una solució valuosa per als tècnics d'il·luminació per a la gestió dels seus continguts de vídeo i la cerca ràpida i eficient d'aquests. A través de la búsqueda amb llenguatge natural, l'usuari podrà trobar els vídeos utilitzant frases o paraules clau sense necessitat de fer gastar termes tècnics. El treball de final de grau proporcionarà una contribució important a la millora d'aquesta eina i als seus possibles usos en altres àmbits.

Paraules clau: vídeo, imatges, classificació, intel·ligència artificial, multiplataforma, recursos de vídeo, color, reconeixement de formes, interfície d'usuari

Resumen

El trabajo de fin de grado que se propone tiene como objetivo solucionar un gran problema que afecta a los técnicos de iluminación: la dificultad de mantener ordenados y accesibles grandes cantidades de contenidos de vídeo para su trabajo. La búsqueda manual entre estos contenidos cada vez que quieren realizar alguna tarea supone una pérdida de tiempo y recursos que afecta negativamente en su rendimiento.

Para solucionar este problema, se propone el desarrollo de una aplicación de escritorio multiplataforma que utiliza técnicas de análisis estadístico, visión por computadora e inteligencia artificial para la clasificación de vídeos. Además, la aplicación proporcionará una interfaz de usuario amigable donde el usuario podrá buscar sus vídeos clasificados.

La arquitectura de la aplicación constará de una interfaz y un motor de clasificación que realizará las operaciones en paralelo a través de diferentes técnicas de reparto de tareas entre procesos para una mejor eficiencia en el tratamiento de los datos.

El trabajo incluirá una investigación a fondo sobre las técnicas de análisis estadístico, visión por computadora e inteligencia artificial para la clasificación de vídeos, así como el diseño y desarrollo de un primer prototipo de esta aplicación. Se realizará una evaluación exhaustiva del rendimiento del prototipo y se propondrán posibles mejoras.

Esta aplicación representa una solución valiosa para los técnicos de iluminación para la gestión de sus contenidos de vídeo y la búsqueda rápida y eficiente de estos. A través

de la búsqueda con lenguaje natural, el usuario podrá encontrar los vídeos utilizando frases o palabras clave sin necesidad de gastar términos técnicos. El trabajo de fin de grado proporcionará una contribución importante a la mejora de esta herramienta y a sus posibles usos en otros ámbitos.

Palabras clave: vídeo, imágenes, clasificación, inteligencia artificial, multiplataforma, recursos de vídeo, color, reconocimiento de formas, interfaz de usuario

Abstract

The proposed final degree project aims to solve a major problem affecting lighting technicians: the difficulty of keeping large amounts of video content organized and accessible for their work. The manual search among these contents every time they want to perform a task represents a waste of time and resources that negatively affects their performance.

To solve this problem, the development of a multiplatform desktop application that uses statistical analysis techniques, computer vision, and artificial intelligence for video classification is proposed. In addition, the application will provide a user-friendly interface where the user can search for their classified videos.

The architecture of the application will consist of an interface and a classification engine that will perform operations in parallel through different task distribution techniques among processes for better efficiency in data processing.

The work will include a thorough investigation of statistical analysis techniques, computer vision, and artificial intelligence for video classification, as well as the design and development of a first prototype of this application. An exhaustive evaluation of the prototype's performance will be carried out, and possible improvements will be proposed.

This application represents a valuable solution for lighting technicians for the management of their video content and the quick and efficient search of these. Through natural language search, the user will be able to find the videos using phrases or keywords without the need to spend technical terms. The final degree work will provide a significant contribution to the improvement of this tool and its possible uses in other areas.

Key words: video, images, classification, artificial intelligence, cross-platform, video resources, color, shape recognition, user interface

Índex

Índex de figures

CAPÍTOL 1
Introducció

La informàtica i, més concretament, l'enginyeria del programari [10], ha desempenyat un paper fonamental en la transformació de moltes indústries i professions. Un d'aquests camps que encara presenta reptes significatius és el dels tècnics de il·luminació. Aquests professionals utilitzen una gran quantitat de continguts de vídeo, però sovint es troben amb la dificultat de mantenir ordenats aquests materials i poder accedir-hi de manera eficient. Aquesta problemàtica no sols provoca una pèrdua de temps considerable, sinó que també pot reduir la productivitat i el rendiment general.

Aquest treball busca abordar aquest problema a través del desenvolupament d'una aplicació d'escriptori multiplataforma que utilitza tècniques d'anàlisi estadística [26], visió per ordinador [62] i intel·ligència artificial [61] per a la classificació automàtica de vídeos. Això no sols permetrà als tècnics d'il·luminació disposar de les seues col·leccions de vídeo de manera més eficient, sinó que també facilitarà la cerca de continguts específics a través d'una interfície d'usuari amigable.

L'arquitectura de l'aplicació combinarà una interfície d'usuari comprensible amb un potent motor de classificació que operarà en paral·lel, utilitzant diverses tècniques de repartiment de tasques entre processos per millorar l'eficiència en el processament de les dades.

D'altra banda, aquest treball se centrarà en la investigació de les tècniques d'anàlisi estadístic, visió per ordinador i intel·ligència artificial que seran utilitzades per a la classificació de vídeos [42], així com en el disseny i el desenvolupament d'un primer prototip de l'aplicació. Posteriorment, es realitzarà una avaluació exhaustiva del rendiment del prototip per identificar i proposar possibles millores.

Finalment, es creu que aquesta aplicació podria no solament millorar l'eficiència dels tècnics d'il·luminació en la seva gestió dels continguts de vídeo, sinó que també podria ser d'interès per a altres àmbits que requereixen la gestió i cerca eficient de grans volums de continguts de vídeo. Aquest treball, per tant, representa una contribució significativa al camp de la gestió de continguts de vídeo, amb l'objectiu d'optimitzar el rendiment i millorar la productivitat dels usuaris.

1.1 Motivació

El món de la il·luminació professional, com molts altres àmbits, ha experimentat un canvi significatiu amb l'arribada de les noves tecnologies. Tot i això, encara existeixen alguns reptes que requereixen solucions més eficients, en especial quan es tracta de gestionar grans quantitats de continguts de vídeo. La complexitat i la monotonia d'aquesta tasca

sovint provoquen pèrdua de temps, desaprofitament de recursos i una reducció general del rendiment professional.

Aquest treball naix de la motivació de contribuir a la solució d'aquest problema i oferir una eina que permeti als tècnics d'il·luminació millorar la seva eficiència. Amb l'ajuda de l'enginyeria del programari, la visió per ordinador i la intel·ligència artificial, és possible desenvolupar una aplicació d'escriptori multiplataforma que transforme completament la manera en què aquests professionals interactuen amb els seus continguts de vídeo.

La importància d'esta aplicació per a l'empresa on s'està desenvolupant [1] és incommensurable. Com a empresa que ven dispositius i programari per a instal·lacions de vídeo i so d'espectacles en directe, s'entén la necessitat de millorar l'eficiència i la gestió dels continguts de vídeo. Esta aplicació no solament millorarà la productivitat dels clients, sinó que també pot ser un gran atractiu per a nous usuaris que busquen solucions innovadores per a la gestió de continguts de vídeo. A més, esta aplicació pot ser una gran oportunitat per a l'empresa per a expandir-se cap a nous mercats i oferir servicis addicionals, com ara suport i formació en l'ús de l'aplicació. En resum, pot aportar un gran valor a l'empresa, als clients i al sector de la il·luminació professional en general.

1.2 Objectius

En aquest treball, s'han proposat una sèrie d'objectius clau per a guiar el desenvolupament i assegurar que la nostra solució siga eficaç i beneficiosa per als tècnics d'il·luminació. Aquests objectius són els següents:

1. Investigar les tècniques d'anàlisi estadístic, visió per ordinador i intel·ligència artificial que es poden aplicar a la classificació de vídeos. Aquesta investigació permetrà definir la base tècnica necessària per al desenvolupament de la nostra solució.

2. Dissenyar i desenvolupar un primer prototip d'una aplicació d'escriptori multiplataforma. Aquesta aplicació haurà de ser capaç de realitzar la classificació automàtica de vídeos i facilitar la seua cerca d'una manera eficient i intuïtiva per a l'usuari.

3. Optimitzar la gestió de continguts de vídeo dels tècnics d'il·luminació. L'aplicació haurà de proporcionar una interfície d'usuari amigable que facilite l'organització i la cerca dels continguts.

4. Realitzar una avaluació exhaustiva del rendiment de l'aplicació. L'objectiu d'aquesta avaluació és identificar oportunitats de millora, que permetin ajustar i perfeccionar la solució proposta.

5. Explorar possibles aplicacions de l'eina desenvolupada en altres àmbits. Encara que l'aplicació està pensada per als tècnics d'il·luminació, és probable que altres professionals que necessiten gestionar grans volums de continguts de vídeo també puguen beneficiar-se d'aquesta.

1.3 Estructura de la memòria

La memòria està estructurada en diverses seccions que faciliten una comprensió completa i sistemàtica del treball realitzat en aquest projecte de final de grau.

[1] Equipson S.A. - www.equipson.es

Començant amb el capítol 2, es proporciona una profunda revisió de la literatura existent, així com una anàlisi de treballs similars, incloent-hi l'examen de diversos models de classificació de vídeos i l'estudi de les eines existents que ofereixen funcionalitats similars a les que es pretén implementar en aquest projecte.

A continuació, en el capítol 3, es descriuen en detall les diferents tecnologies que s'han utilitzat per al desenvolupament de la nostra aplicació, incloent les plataformes de programació, les llibreries, els marcs de treball i els algorismes específics de classificació.

En la el capítol 4, s'exposa el procés de disseny i implementació de la aplicació, aprofundint en detalls importants com el conjunt de dades utilitzat, el rendiment de la classificació de vídeos i la implementació del sistema. Primer es descriu el conjunt de dades utilitzat per a l'entrenament i la validació dels models de classificació, explicant la seva procedència, la seva preparació i la seva representativitat en el món real. Després, es presenten els resultats del rendiment de la classificació de vídeos, utilitzant mètriques com la precisió, la sensibilitat i l'especificitat, entre d'altres. Finalment, es detalla la implementació final del sistema, describint l'arquitectura del programari, les funcionalitats implementades i la interacció amb l'usuari.

L'últim capítol, Conclusions i treballs futurs, fa un resum de tot el treball realitzat, valorant la consecució dels objectius i proposant futures línies d'investigació i desenvolupament. Aquesta part final proporciona una visió global de l'aportació d'aquest treball de final de grau i els possibles camins a seguir en el futur.

CAPÍTOL 2
Estat de l'art

En aquest capítol explorarem una sèrie de temes clau que són fonamentals per al desenvolupament de l'aplicació que es descriu en el treball. Aquesta aplicació, destinada a ajudar als tècnics d'il·luminació a gestionar grans quantitats de continguts de vídeo, utilitza una combinació de tècniques d'anàlisi estadístic, visió per ordinador i intel·ligència artificial per a la classificació de vídeos. A més, l'aplicació proporciona una interfície d'usuari de fàcil ús on l'usuari pot cercar els seus vídeos classificats.

Primer, ens centrarem en la classificació de vídeos, àrea de la informàtica que s'ocupa de l'assignació de categories o etiquetes a vídeos basant-se en el seu contingut [35] . Es tracta d'una tasca complexa que requereix l'ús de tècniques avançades de visió per ordinador i intel·ligència artificial. La visió per ordinador és una disciplina que busca replicar i superar la capacitat humana de comprendre imatges i vídeos [58], mentre que la intel·ligència artificial, per la seua banda, es refereix a l'ús de màquines per a realitzar tasques que normalment requereixen de la intel·ligència humana, com ara l'aprenentatge, la percepció i la resolució de problemes [33] .

A continuació, discutirem sobre l'ús de l'anàlisi estadística en el context de la classificació de vídeos, ja que és una eina poderosa que ens permet entendre i interpretar grans conjunts de dades, com ara aquelles que es generen quan es classifiquen vídeos.

També parlarem sobre l'arquitectura de l'aplicació, que inclou l'ús de tecnologies com Docker [82] i ZeroMQ [90] per a la comunicació entre processos i la distribució de tasques. Docker és una plataforma que permet empaquetar una aplicació i les seves dependències en un contenidor virtual que pot funcionar en qualsevol màquina, mentre que ZeroMQ és una biblioteca de xarxa de gran rendiment que proporciona patrons de comunicació entre processos.

Finalment, discutirem sobre la cerca amb llenguatge natural, una característica clau de l'aplicació que permet als usuaris cercar vídeos utilitzant frases o paraules clau sense necessitat d'utilitzar termes tècnics.

Cadascun d'aquests temes és crucial per al desenvolupament de l'aplicació i proporciona el marc necessari per a la seva implementació i ús efectiu. En les següents seccions, els explorarem en més detall.

2.1 Classificació de vídeos

La classificació de vídeos és una tasca complexa que requereix l'ús de tècniques avançades de visió per ordinador i intel·ligència artificial. En aquesta secció, explorarem algunes de les tècniques i aplicacions més rellevants en aquest camp.

En primer lloc, és important entendre que la classificació de vídeos implica l'assignació de categories o etiquetes a vídeos basant-se en el seu contingut. Aquesta tasca és crucial per a nombroses aplicacions, com ara la indexació de vídeos, la cerca, l'anotació i la vigilància. Com es destaca en l'estudi de Saddam Bekhet i Abdullah M. Alghamdi [48], hi ha tres tècniques principals que s'han adoptat generalment per a classificar vídeos: l'aparellament directe de característiques [27, 30, 28] , els mètodes basats en l'aprenentatge automàtic [19, 17, 31, 45, 29] , i els mètodes basats en l'aprenentatge profund [88, 34]. Cadascun d'aquests mètodes és adequat per a un tipus d'aplicació específic.

L'aparellament directe de característiques implica comparar les característiques d'un vídeo amb les d'un conjunt de vídeos de referència. Aquesta tècnica pot ser útil per a tasques com la cerca de vídeos, on l'objectiu és trobar vídeos que siguin similars a un vídeo de consulta donat [14].

Els mètodes basats en l'aprenentatge automàtic, per la seva banda, entrenen un model a partir d'un conjunt de vídeos etiquetats, i després utilitzen aquest model per a classificar nous vídeos. Aquests mètodes poden ser útils per a tasques com la detecció d'activitats, on l'objectiu és identificar quines activitats estan sent realitzades en un vídeo [39].

Finalment, els mètodes basats en l'aprenentatge profund utilitzen xarxes neuronals [4] per a aprendre representacions de vídeos que són útils per a la classificació. Aquests mètodes són particularment potents per a tasques que requereixen una comprensió detallada del contingut d'un vídeo, com ara la generació de descripcions de vídeos [50].

En aquest sentit, l'estudi de Sohini Roychowdhury [60] presenta dos enfocaments semi-supervisats que automatitzen el procés de selecció manual de fotogrames en fluxos de vídeo, classificant automàticament les escenes per al contingut i filtrant els fotogrames per a la millora de tasques d'enteniment de l'escena.

El primer mètode està basat en regles i comença a partir d'un detector d'objectes preentrenat [25]. Assigna el tipus d'escena, incertesa i categories d'il·luminació a cada fotograma basant-se en les distribucions de probabilitat dels objectes de primer pla. Els fotogrames amb la major incertesa i dissimilaritat estructural s'aïllen com a fotogrames clau [60].

El segon mètode es recolça en el model simCLR [38] per a la codificació de fotogrames, seguit de l'extensió d'etiquetes des del 20% de les mostres per etiquetar la resta d'aquests per a categories d'escena i il·luminació. La agrupació de les imatges del vídeo en l'espai de característiques codificades aïlla encara més els fotogrames clau en les fronteres dels grups [60].

Aquests mètodes aconsegueixen una precisió d'entre el 64% i el 93% [60] per a la categorització automàtica d'escenes per a vídeos d'imatges a l'aire lliure de conjunts de dades de domini públic de JAAD [85] i KITTI [16]. A més, menys del 10% dels fotogrames d'entrada es poden filtrar com a fotogrames clau que després es poden enviar per a anotació i ajustament fi dels algoritmes de visió de màquines.

Aquests mètodes podrien aplicar-se a la classificació de vídeos i a la selecció de fotogrames clau per a una anàlisi o processament posterior. El mètode basat en regles podria ser particularment útil per la seua robustesa a les variacions de les dades d'entrenament, mentre que el mètode basat en simCLR podria proporcionar temps de processament més ràpids [60].

D'altra banda, en l'estudi de M. Awais, et al. [47] es compara l'aprenentatge automàtic clàssic [13] i l'aprenentatge profund [22] per a la classificació d'activitats de la vida diària en adults majors. S'utilitzen tant les xarxes de memòria a llarg termini (LSTM) [54] com les Màquines de Vector Suport (SVM) [54] per a la classificació. No obstant això, es troba

que LSTM era més eficaç per a aquesta tasca, aconseguint una puntuació F global del 97,23% en comparació amb el 94,33% de SVM. Aquests resultats podrien ser útils per a informar l'elecció de tècniques d'aprenentatge automàtic o d'aprenentatge profund per a la classificació de vídeos en el context del TFG.

L'estudi de Saddam Bekhet i Abdullah M. Alghamdi [63] realitza una anàlisi comparativa de tres tècniques principals per a la classificació de vídeos (l'aparellament directe de característiques, els mètodes basats en l'aprenentatge automàtic i els mètodes basats en l'aprenentatge profund), i conclou que els mètodes basats en l'aprenentatge profund són els més eficaços per a la classificació de vídeos, amb una gran diferència de rendiment en comparació amb l'aprenentatge automàtic i l'aparellament directe de característiques. Aquesta conclusió es basa en l'ús de dues tècniques de classificació principals: la Màquina de Vector Suport (SVM) i el Naïve Bayes [23], amb SVM superant a Naïve Bayes en termes de precisió, recordatori i puntuació F1. Aquesta informació podria ser útil per a informar l'elecció de tècniques per a la classificació de vídeos.

Aquestes investigacions mostren que la classificació de vídeos és una tasca complexa que requereix l'ús de tècniques avançades de visió per ordinador i intel·ligència artificial. Tot i que s'han fet grans avanços en aquest camp, encara hi ha moltes limitacions a superar. Per exemple, la majoria de les tècniques actuals requereixen grans quantitats de dades etiquetades per a l'entrenament, cosa que pot ser un desafiament en termes de temps i recursos. A més, moltes tècniques encara tenen dificultats per a gestionar la variabilitat inherent als vídeos, com ara canvis en la il·luminació, oclusions i moviments de càmera.

Una última tècnica és l'ús de YOLO (*You Only Look Once*) [24], un mètode popular en l'anàlisi i classificació de vídeos. Aquesta pràctica és especialment útil en la detecció d'objectes en temps real, ja que processa les imatges en un sol pas, cosa que la fa més ràpida que altres mètodes que utilitzen dues fases [25] per a la detecció i classificació d'objectes [32].

Un exemple d'aplicació real de YOLO en l'anàlisi de vídeo es troba en el treball de Lee i col·laboradors [57]. En aquest estudi, els autors utilitzen YOLO-V5 per a la detecció i classificació d'objectes en entorns marítims. Els autors utilitzen el Singapore Maritime Dataset (SMD) [56], un conjunt de dades públic amb vídeos anotats, per a l'entrenament de xarxes neuronals profundes (DNN) [22] en la detecció d'objectes marítims. No obstant això, l'SMD té etiquetes sorolloses i caixes de contorn imprecises, per la qual cosa els autors corregixen aquestes anotacions i presenten una versió millorada, anomenada SMD-Plus.

Per a l'entrenament de YOLO-V5, els autors proposen tècniques d'augmentació, dissenyades especialment per a l'SMD-Plus. Apliquen una transformació en línia de les imatges d'entrenament mitjançant la tècnica de *Copy And Paste* per a resoldre el problema del desequilibri de classes en el conjunt de dades d'entrenament. A més, utilitzen la tècnica de *mix-up* a banda de les tècniques d'augmentació bàsiques per a YOLO-V5 [56]. Els resultats experimentals mostren que el rendiment de detecció i classificació del YOLO-V5 modificat amb l'SMD-Plus ha millorat en comparació amb el YOLO-V5 original.

Aquesta aplicació de YOLO en l'anàlisi de vídeo demostra la seua eficàcia en la detecció i classificació d'objectes en entorns reals. A més, l'ús de tècniques d'augmentació específiques per a l'entrenament de YOLO-V5 mostra com es poden superar els desafiaments com el desequilibri de classes en els conjunts de dades d'entrenament.

L'ús de YOLO per a la classificació de vídeos presenta una sèrie d'avantatges i desavantatges que cal considerar:

- Avantatges

  1. Velocitat
     YOLO és extremadament ràpid. El model base de YOLO pot processar imatges en temps real a 45 fotogrames per segon, i una versió més petita de la xarxa, Fast YOLO, pot processar fins a 155 fotogrames per segon. Aquesta velocitat el fa ideal per a aplicacions en temps real, com ara la detecció d'objectes en fluxos de vídeo en directe [41, 24].
  2. Eficiència
     A diferència d'altres mètodes de detecció d'objectes que utilitzen dues fases per a la detecció i classificació d'objectes, YOLO processa les imatges en un sol pas. Això el fa més eficient i ràpid [41, 46].
  3. Precisió
     Tot i que YOLO pot cometre més errors de localització que altres sistemes de detecció, és menys probable que prediga falses deteccions on no hi ha res. A més, YOLO aprèn representacions molt generals dels objectes, superant altres mètodes de detecció, incloent DPM [15] i R-CNN [20], quan es generalitza des d'imatges naturals a obres d'art [24].

- Desavantatges

  1. Errors de localització
     YOLO fa més errors de localització que altres sistemes de detecció. Això es deu al fet que YOLO prediu les caixes de contorn i les probabilitats de classe directament des de les imatges completes en una sola avaluació, cosa que pot conduir a errors de localització [24].
  2. Dificultat amb objectes petits
     YOLO pot tenir dificultats per a detectar objectes petits que apareixen en grups, com ara un bandada d'ocells o un grup de persones. Això es deu al fet que YOLO divideix la imatge en una graella i assigna cada objecte a la cel·la de la graella on es troba el centre de l'objecte. Si hi ha molts objectes petits agrupats en una sola cel·la de la graella, YOLO pot no ser capaç de detectar-los tots [44, 46].
  3. Necessitat de dades etiquetades
     Com altres mètodes d'aprenentatge profund, YOLO requereix grans quantitats de dades etiquetades per a l'entrenament. Aquesta necessitat pot ser un desafiament en termes de temps i recursos [44].

Abans de decidir sobre quina tècnica és millor per a la classificació del contingut dels vídeos cal tindre en compte el tipus de contingut que s'analitzarà. Els recursos de vídeo destinats a les pantalles LED dels espectacles solen contenir un contingut molt específic, com ara imatges de persones, objectes quotidians, textos, etc. Tanmateix, solem trobar principalment amb formats amb formes geomètriques i abstractes que es mouen per la pantalla. Aquest tipus de contingut és molt diferent del que es pot trobar en un vídeo de YouTube, per exemple. Per tant, triar la millor opció per a la classificació de contingut de vídeo depèn del tipus de contingut que es vol analitzar.

YOLO podria ser una eina útil per a aquesta aplicació, especialment si es necessita processar i classificar vídeos en temps real o a una velocitat elevada. La seua eficiència i velocitat podrien ser avantatges significatius en aquest context.

No obstant això, cal considerar com, ja hem comentat, que YOLO pot tenir dificultats amb objectes petits que apareixen en grups. Si els vídeos contenen moltes formes geomè-

(**a**) Original Image (**b**) Reduced to $4 \times 4 \times 4 = 64$ colors (**c**) Image Histogram

Figura 2.1: Exemple d'anàlisi d'histograma realitzat al treball de Gonzalez i Woods [8]

triques petites o formes extravagants agrupades, és possible que YOLO no siga capaç de detectar-les totes de manera eficient.

D'altra banda, YOLO requereix grans quantitats de dades etiquetades per a l'entrenament. Si no es disposa d'un conjunt de dades d'entrenament extens i ben etiquetat, podrien tenir-se dificultats.

Finalment, si els vídeos són principalment abstractes, la classificació pot ser més difícil que amb vídeos que contenen objectes més convencionals. YOLO, com altres mètodes d'aprenentatge profund, en molts casos necessita ser entrenat amb dades similars a les que es volen classificar. Si les dades són molt diferents de les que s'han emprat per entrenar YOLO, s'hauria de realitzar un treball considerable per a preparar les dades d'entrenament i ajustar el model.

En resum, mentre que YOLO té potencial per a ser útil en aquesta aplicació, també presenta alguns desafiaments que s'haurien de considerar. És possible que siga convenient explorar altres tècniques de classificació de vídeos, o combinar YOLO amb altres mètodes, per tant d'obtenir els millors resultats per a l'aplicació específica.

2.2 L'estadística aplicada a la classificació de vídeos. L'histograma de color

Un histograma de color és una representació de la distribució de colors en una imatge o vídeo. Cada color es representa per un punt en l'histograma, i l'alçada de cada punt indica la quantitat de píxels en la imatge o vídeo que tenen aquest color [8].

Un dels usos més comuns dels histogrames de color en la classificació de vídeos és la detecció de canvis de color. Això es pot utilitzar per detectar canvis en l'escena, com ara el pas d'un personatge o objecte d'un color a un altre. A més, este tipus d'histogrames també es poden utilitzar per a la classificació de vídeos basada en el color. Per exemple, si un vídeo conté una gran quantitat de blau, es pot classificar com a vídeo d'aigua o cel com podem observar en la figura 2.1.

Gonzalez i Woods [8] van proposar una tècnica per a la segmentació d'imatges de color utilitzant histogrames jeràrquics adaptatius. Aquest mètode utilitza un histograma de color per dividir una imatge en regions basades en el color. A continuació, cada regió es divideix en subregions utilitzant un altre histograma de color. Aquest procés es repeteix fins que les regions no es poden dividir més.

Zhang, Chi i He [21] van presentar una tècnica adaptativa no supervisada per a l'agrupació de píxels i la segmentació d'imatges de color. Aquesta pràctica utilitza un histograma de color per determinar els canvis de gradient de pics distints de la intensitat del canal RGB. A continuació, se seleccionen els valors de pic d'intensitat més alts dins dels

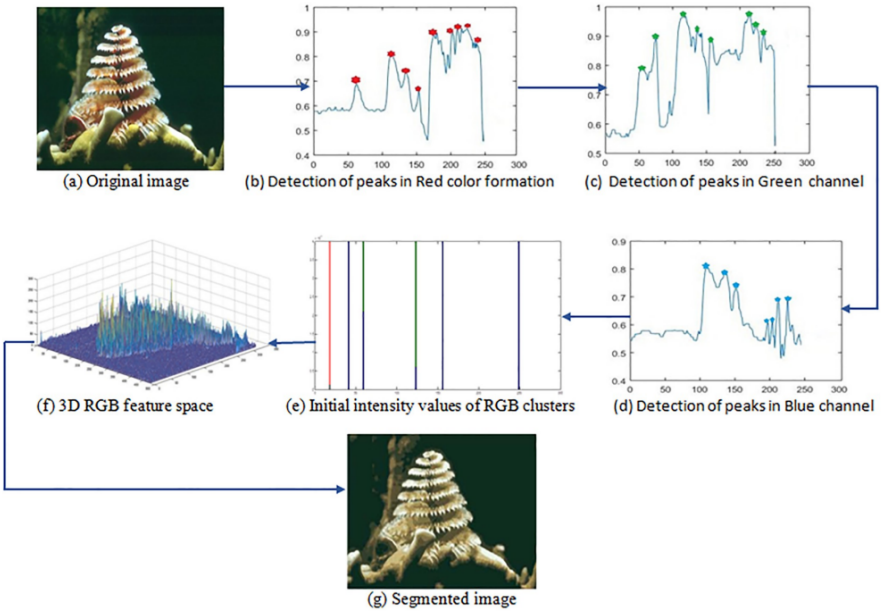

Figura 2.2: Exemple de segmentació basada en color realitzat al treball de Zhang, Chi i He [21]

rangs d'intensitat veïns per a cada canal de color. Aquesta tècnica permet una segmentació precisa de les imatges de color, que és útil per a la classificació de vídeos. Podem vore aquest procés en el diagrama il·lustrat en la figura 2.2.

Com podem veure, els histogrames de color són una eina útil en la classificació de vídeos. Permeten detectar canvis de color, classificar vídeos basats en el color i segmentar imatges basades en el color [8, 21].

D'altra banda, l'ús de l'histograma per a la classificació de contingut de vídeo és una pràctica comuna en el camp de la recuperació d'informació basada en contingut [5]. Aquesta pràctica es basa en la representació de la distribució de color en una imatge o un fotograma de vídeo [11]. Principalment, l'aproximació de l'histograma de color compta el nombre d'ocurrències de cada color únic en una imatge de mostra. Aquest mètode és molt simple, fàcil de manejar i computacionalment barat.

A més, aquesta pràctica pot ser justificada per la seva eficàcia en l'extracció de paletes de color representatives dels continguts. En l'article de Saykol et al. [11], es descriu un enfocament on es fa servir una mesura probabilística per pesar els colors dels píxels respecte a algun veïnat de píxels. Aquesta mesura millora l'eficàcia d'aquesta tècnica sense imposar cap complexitat computacional.

En resum, l'ús de l'histograma per a la classificació de contingut de vídeo i l'extracció de paletes de color representatives és una pràctica eficient i computacionalment viable que proporciona resultats útils en el camp de la recuperació d'informació basada en contingut.

2.3 Servicis distribuïts

Les aplicacions distribuïdes són una part integral de l'arquitectura de programari moderna. Aquestes aplicacions són dissenyades per a funcionar en múltiples nodes de computació que es comuniquen entre si, permetent un major rendiment, escalabilitat i resiliència en comparació amb les aplicacions monolítiques [36, 53].

Hi ha diversos avantatges associats amb l'ús d'aplicacions distribuïdes. En primer lloc, este tipus d'aplicacions poden gestionar càrregues de treball més grans que les aplicacions monolítiques. Això es deu al fet que la càrrega de treball es distribueix entre múltiples nodes de computació, permetent a cada node processar una part de la càrrega de treball total. Això pot resultar en un rendiment significativament millorat, especialment per a aplicacions que requereixen una gran quantitat de recursos de computació [36].

En segon lloc, les aplicacions distribuïdes són més escalables que les aplicacions monolítiques. Això es deu al fet que es poden afegir més nodes de computació a la xarxa a mesura que augmenta la demanda de recursos de computació. Això permet a les aplicacions distribuïdes manejar un augment en la càrrega de treball sense necessitat de canvis significatius en l'arquitectura de l'aplicació [53].

En tercer lloc, les aplicacions distribuïdes són més resilients que les aplicacions monolítiques, ja que si un node de computació falla, la càrrega de treball que estava processant pot ser redistribuïda a altres nodes de la xarxa. Això pot resultar en una menor interrupció del servici en comparació amb una aplicació monolítica, on un únic punt de fallida pot resultar en una parada completa d'aquesta [36].

S'han realitzat diversos experiments per a demostrar els avantatges de les aplicacions distribuïdes. Per exemple, en un estudi realitzat per Nadeem et al. [36], es va utilitzar una arquitectura de núvol distribuïda basada en *blockchain* per a millorar la seguretat en les xarxes de vehicles *ad hoc*. Els resultats obtinguts van demostrar que aquesta arquitectura podia prevenir eficaçment els atacs de suplantació d'identitat, garantint així la seguretat de les dades en la xarxa [36].

En un altre treball realitzat per Wu et al. [53], es van implementar diverses aplicacions distribuïdes escalables basades en HPX [52], un sistema de temps d'execució exemplar. Els resultats de la investigació van demostrar que les aplicacions distribuïdes presenten una bona escalabilitat, especialment quan es tracta de processar grans quantitats de dades.

En general, les aplicacions distribuïdes ofereixen nombrosos avantatges sobre les aplicacions monolítiques, incloent una major capacitat per a gestionar càrregues de treball grans, una major escalabilitat i una major resiliència. A més, els experiments realitzats han demostrat que les aplicacions distribuïdes poden millorar significativament el rendiment i la seguretat de les xarxes de vehicles *ad hoc*, entre altres aplicacions. Per tant, és recomanable considerar l'ús d'aplicacions distribuïdes en l'arquitectura de programari moderna.

2.4 Dockerització d'aplicacions

La dockerització d'aplicacions és un concepte que ha guanyat popularitat en els últims anys degut als seus múltiples avantatges. Docker [82] és una plataforma de programari que permet la creació, desplegament i gestió d'aplicacions dins de contenidors. Aquests contenidors permeten empaquetar una aplicació amb totes les seues dependències en una unitat estàndard per al desplegament de programari [43].

Un dels principals avantatges de la dockerització és la seua capacitat per a proporcionar un alt grau d'aïllament, el que és particularment útil quan es tracta de capturar un entorn computacional específic per a la reproducció i extensió de resultats de recerca [43]. Aquest aïllament permet als desenvolupadors assemblar *pipelines* de recerca amb versions específiques de programari per a minimitzar problemes amb canvis disruptius i facilitar la compartició de fluxos de treball.

Un altre avantatge significatiu de la dockerització és la seua capacitat per a proporcionar entorns reproduïbles, escalabilitat, eficiència i portabilitat a través d'infraestructures. Aquestes característiques són especialment útils en el context de desplegaments escalables, ja que Docker pot moure fàcilment el processament entre màquines, com ara localment, una màquina virtual d'un proveïdor de núvol, o un altre servici de contenidors com a servici [43].

En un estudi realitzat per Nüst et al. [43], es va demostrar que la dockerització pot ser utilitzada per a millorar la reproducibilitat de la recerca. En aquest estudi, es va utilitzar Docker per a crear un «compendi de recerca» que contenia totes els dades i codis necessaris per a reproduir completament un article de recerca. Aquest compendi es va empaquetar dins d'un contenidor Docker, el que va permetre a altres investigadors descarregar el compendi del repositori i executar el Dockerfile inclòs per a construir una nova imatge que recreava l'entorn computacional utilitzat per a produir els resultats originals de la recerca.

Aquesta estratègia va resultar ser extremadament eficaç per a garantir la reproducibilitat de la recerca. Si la construcció de la imatge fallava, les instruccions del Dockerfile proporcionaven una guia detallada per a resoldre qualsevol problema. A més, una vegada que la imatge es va construir amb èxit, es podia utilitzar per a reproduir exactament els resultats de la recerca, independentment de l'entorn computacional local de l'usuari.

D'altra banda, l'execució de contenidors Docker dins d'altres contenidors Docker, coneguda com a *Docker In Docker* (DinD), és una pràctica que pot ser útil en una varietat de situacions, com ara la creació d'entorns de proves aïllades, la construcció de *pipelines* d'integració i distribució continua (CI/CD), i altres tasques de desenvolupament de programari. No obstant això, aquesta pràctica pot presentar alguns problemes i desafiaments únics [76].

Una de les maneres més senzilles d'executar contenidors Docker des de dins d'un contenidor Docker és compartir el *socket* d'execució del *Docker Engine*. Això es pot fer muntant el *socket* d'execució en el contenidor Docker. Aquesta tècnica permet al contenidor accedir al *Docker Engine* de l'amfitrió, permetent així l'execució de contenidors Docker "germans" en lloc de contenidors Docker "fills" [76].

Una altra opció és utilitzar imatges Docker específicament dissenyades per a l'execució de *Docker In Docker*, com ara la imatge disponible en [65]. Aquestes imatges estan configurades per a permetre l'execució de *Docker In Docker* i poden ser una opció útil per a certes tasques. No obstant això, és important tenir en compte que l'execució de *Docker In Docker* pot presentar alguns problemes i desafiaments únics, i que potser no és la millor opció per a totes les situacions [76].

La dockerització d'aplicacions ofereix nombrosos avantatges, incloent aïllament, reproducibilitat, escalabilitat, eficiència i portabilitat. Aquests avantatges fan que Docker siga una eina valuosa per a la recerca i el desenvolupament de programari. Malgrat els desafiaments que presenta, com ara la necessitat de comprendre i gestionar les dependències de programari, els avantatges de la dockerització són evidents i la seua utilització continuarà creixent en el futur [43].

2.5 Recuperació de la informació a través de llenguatge natural

La recuperació de la informació a través del llenguatge natural és una àrea de la ciència de la computació que es dedica a processar i analitzar el llenguatge humà de manera que les màquines el puguen entendre, interpretar i respondre'n de manera significativa [12]. Aquesta àrea ha experimentat un gran creixement i desenvolupament en els últims anys,

gràcies a l'avanç de les tècniques de processament del llenguatge natural (NLP) [49] i l'aprenentatge automàtic .

Aquest tipus de tècniques tenen diversos avantatges. Primerament, permet a les màquines comprendre i respondre a les consultes dels usuaris en un llenguatge natural, millorant així la interacció entre l'usuari i la màquina. A més, aquesta tècnica pot millorar significativament la precisió de la recuperació de la informació, ja que pot comprendre el context i la semàntica de les demandes dels usuaris [2].

En segon lloc, la recuperació de la informació a través del llenguatge natural pot ser utilitzada en una àmplia gamma d'aplicacions, incloent la cerca d'informació en línia [9], l'assistència virtual [40], la traducció automàtica [18] i l'anàlisi de sentiments [59], entre d'altres. Aquesta àmplia aplicabilitat fa que la recuperació de la informació mitjançant del llenguatge natural siga una àrea de gran interès per a la recerca i el desenvolupament [1].

Diversos estudis han demostrat l'eficàcia de la recuperació de la informació per mitjà del llenguatge natural. Per exemple, un estudi va utilitzar tècniques de NLP per analitzar les consultes dels usuaris en un motor de cerca en línia. Els resultats van mostrar que l'ús de NLP va millorar significativament la precisió de la recuperació de la informació, en comparació amb les tècniques de recuperació de la informació tradicionals [50].

Una investigació va utilitzar tècniques de NLP per analitzar les consultes dels usuaris en un sistema d'assistència virtual. Els resultats van mostrar que l'ús de NLP va permetre al sistema comprendre millor les consultes dels usuaris i proporcionar respostes més precises i útils [1].

D'altra banda, en un estudi realitzat amb la col·lecció CACM-3204 [3] d'abstracts de ciències de la computació, es va observar una millora consistent en el rendiment: la precisió mitjana va augmentar del 32,8% al 37,1% (un augment del 13%), mentre que la recuperació normalitzada va passar del 74,3% al 84,5% (un augment del 14%), en comparació amb les estadístiques del sistema base NIST [6]. Aquesta millora és un efecte combinat del nou *stemmer* [51], els termes compostos, la selecció de termes en les consultes i l'expansió de consultes utilitzant relacions de similitud filtrades. L'elecció del filtre de relació de similitud s'ha trobat crítica per a millorar la precisió de la recuperació mitjançant l'expansió de consultes [1].

La recuperació de la informació a través del llenguatge natural és una àrea prometedora de la ciència de la computació que té el potencial de millorar significativament la interacció entre l'usuari i la màquina, així com la precisió de la recuperació de la informació. Malgrat els desafiaments que presenta, com ara la necessitat de comprendre el context i la semàntica de les consultes dels usuaris, els avanços recents en NLP i l'aprenentatge automàtic estan facilitant el desenvolupament de tècniques més eficaces i precises per a la recuperació de la informació per mitjà del llenguatge natural [1].

CAPÍTOL 3
Tecnologies utilitzades

En el món actual de la tecnologia, la creació d'aplicacions eficients, robustes i de qualitat professional requereix més que el domini d'una única eina o llenguatge de programació. Es necessita una combinació de diverses tecnologies, cadascuna amb les seves pròpies fortaleses, per afrontar els diferents reptes que es presenten en el procés de desenvolupament. La integració d'aquestes tecnologies permet aprofitar el millor d'elles, creant solucions més completes i eficaces.

En el context d'aquest projecte, la combinació de tecnologies com OpenCV, CVAT, Docker, ZeroMQ i Flask ha estat clau per aconseguir els objectius proposats. Cadascuna d'aquestes eines aporta una funcionalitat específica que, en conjunt, permeten el desenvolupament d'una aplicació d'escriptori multiplataforma capaç de gestionar i classificar grans quantitats de continguts de vídeo de manera eficient.

La visió per ordinador i l'aprenentatge automàtic, proporcionats per OpenCV i CVAT, permeten la classificació i l'etiquetatge de vídeos. Docker assegura la portabilitat i la consistència de l'aplicació en diferents entorns mentres que ZeroMQ facilita la comunicació entre processos i la gestió de tasques en paral·lel, millorant l'eficiència del processament de dades. D'altra banda, Flask proporciona una interfície d'usuari amigable i eficient.

Aquesta combinació de tecnologies no només permet la creació d'una aplicació funcional, sinó que també assegura que l'aplicació siga escalable, mantenible i de qualitat professional. Aquesta és la força de la combinació de tecnologies: la capacitat de crear solucions que són més grans que la suma de les seves parts.

3.1 OpenCV

OpenCV (*Open Source Computer Vision Library*) [87] és una biblioteca de programació de codi obert principalment per a la visió per ordinador en temps real. Originalment desenvolupada per Intel, més tard va ser suportada per Willow Garage, i seguidament per Itseez, que finalment va ser adquirida per Intel. La biblioteca és multiplataforma i està llicenciada com a programari lliure i de codi obert sota la Llicència Apache 2 [80]. A partir de 2011, OpenCV inclou acceleració GPU per a operacions en temps real [64].

OpenCV proporciona una infraestructura comuna per a les aplicacions de visió per ordinador i accelera l'ús de la percepció de màquina en els productes comercials. La biblioteca té més de 2500 algoritmes optimitzats, que inclouen un conjunt complet d'algoritmes clàssics i d'avantguarda de visió per ordinador i aprenentatge automàtic. Aquests algoritmes es poden utilitzar per a detectar i reconèixer cares, identificar objectes, classificar accions humanes en vídeos, seguir moviments de càmera, seguir objectes en moviment, extreure models 3D d'objectes, produir núvols de punts 3D a partir de càmeres

estereoscòpiques, cosir imatges per produir una imatge d'alta resolució de tota una escena, trobar imatges similars a partir d'una base de dades d'imatges, eliminar ulls vermells de les imatges preses amb flash, seguir moviments d'ulls, reconèixer escenaris i establir marcadors per superposar-los amb realitat augmentada, entre altres [64].

OpenCV és compatible amb diversos llenguatges de programació, incloent C++, Python, Java i MATLAB, i ofereix suport per a Windows, Linux, Android i Mac OS. OpenCV es decanta principalment cap a les aplicacions de visió en temps real i aprofita les instruccions MMX i SSE quan estan disponibles. Actualment es desenvolupen interfícies completes de CUDA i OpenCL [64].

En el context del nostre projecte, OpenCV s'utilitza per a l'anàlisi i processament del contingut de vídeo. Les seves funcions avançades de visió per ordinador permeten la detecció i el reconeixement d'objectes en els vídeos, així com altres tasques. Aquesta capacitat és crucial per a la classificació i l'organització eficaç dels vídeos en l'aplicació.

3.2 CVAT

CVAT (Computer Vision Annotation Tool) [81] és una eina interactiva d'anotació de vídeo i imatges per a la visió per ordinador. És utilitzada per desenes de milers d'usuaris i empreses arreu del món. La seva missió és ajudar els desenvolupadors, empreses i organitzacions arreu del món a resoldre problemes reals utilitzant l'enfocament de la Intel·ligència Artificial (IA) centrada en les dades [71].

CVAT ofereix una interfície d'usuari intuïtiva i potent que permet als usuaris anotar imatges i vídeos de manera eficient. A més, CVAT suporta múltiples formats d'anotació, el que permet als usuaris importar i exportar anotacions en el format que millor s'adapti a les seves necessitats. Aquesta flexibilitat fa que CVAT siga una eina ideal per a projectes que requereixen anotacions d'imatges o vídeos per a tasques de visió per ordinador [71].

Una de les característiques més importants de CVAT és que és una aplicació dockeritzada. Això significa que es pot desplegar fàcilment en qualsevol sistema que suporti Docker, sense haver de preocupar-se per les dependències del sistema. Aquesta característica fa que CVAT siga una eina molt portable i fàcil d'instal·lar i utilitzar [71].

A més, CVAT inclou funcions d'ajuda a l'etiquetat de contingut multimèdia a través de models d'intel·ligència artificial. Aquestes funcions inclouen el seguiment (*tracking*), l'etiquetatge automàtic i la segmentació. El seguiment permet a CVAT seguir objectes a través de múltiples fotogrames en un vídeo, facilitant l'etiquetatge de vídeos. L'etiquetatge automàtic permet a CVAT utilitzar models d'IA per a etiquetar automàticament objectes en imatges i vídeos. Finalment, la segmentació permet a CVAT dividir una imatge o un fotograma de vídeo en múltiples segments, els quals poden ser etiquetats individualment [71].

A aquest treball, CVAT s'utilitza per a l'anotació manual de vídeos. Les seves funcions avançades d'anotació permeten als usuaris marcar i etiquetar objectes d'interès en els vídeos, la qual cosa és crucial per a la classificació i l'organització eficaç dels vídeos en l'aplicació.

3.3 Docker

Docker [82] és una tecnologia de contenidors que permet als desenvolupadors enviar aplicacions de programari juntament amb les seves dependències en imatges Docker.

Aquestes imatges Docker són lleugeres, portàtils i autònomes, el que facilita la construcció de programari de manera eficient [72].

Docker encapsula tot el que una aplicació necessita per executar-se, permetent que les aplicacions es puguin traslladar fàcilment entre entorns. Qualsevol *host* amb el temps d'execució de Docker instal·lat pot executar un contenidor Docker, el que fa que Docker sigui ideal per a l'arquitectura de microservicis, on les aplicacions es constitueixen a partir de molts components poc acoblats [78].

A més, Docker permet un ús més eficient dels recursos del sistema, ja que les instàncies d'aplicacions contenitzades utilitzen molt menys memòria que les màquines virtuals, s'inicien i s'aturen més ràpidament, i es poden empaquetar de manera més densa en el seu maquinari *host* [77]. Això resulta en menys despeses en tecnologies de la informació. Aquesta tecnologia també permet cicles de lliurament de programari més ràpids, ja que facilita la posada en producció de noves versions de programari amb noves funcionalitats empresarials de manera ràpida, i també permet retrocedir ràpidament a una versió anterior si cal [78].

En el context d'aquest projecte, Docker és particularment útil per a la gestió de les dependències de programari necessàries per a l'anàlisi de vídeo. A més, la capacitat de Docker per contenir i aïllar aplicacions facilita la gestió de les tasques de processament de vídeo, ja que cada tasca pot ser encapsulada en un contenidor Docker independent. Això pot permetre una millor distribució de la càrrega de treball i una millor gestió dels recursos del sistema [74]. A més, l'ús de Docker ens facilita la implementació i l'escalabilitat de l'aplicació, ja que les imatges Docker es poden desplegar sense cap problema en qualsevol sistema que tingui Docker instal·lat, i es poden escalar per acomodar càrregues de treball més grans [55].

3.4 ZeroMQ

ZeroMQ [90] és una biblioteca de missatgeria de codi obert que proporciona funcions de missatgeria asincrònica, modelatge de col·lecció de dades, i altres patrons de comunicació entre processos, fils, i nodes. Aquesta biblioteca és coneguda per la seva alta velocitat i flexibilitat, i és utilitzada en una àmplia varietat d'aplicacions, des de sistemes de cua de missatges a gran escala fins a microservicis [79].

Aquesta és una biblioteca de xarxa que proporciona una abstracció de les connexions TCP, PGM i IPC. A diferència d'altres biblioteques d'aquest tipus, ZeroMQ proporciona una interfície simple i consistent que és independent del protocol de connexió subjacent. Això permet als desenvolupadors construir aplicacions escalables sense haver de preocupar-se pels detalls de baix nivell de la gestió de les connexions de xarxa [79].

La llibreria és útil per a la comunicació entre processos en una màquina, entre fils dins d'un procés, i entre nodes en una xarxa. Aquesta biblioteca proporciona una interfície de programació simple i consistent per a la comunicació entre aquests diferents contextos. A més, ZeroMQ proporciona una varietat de patrons de comunicació, incloent sol·licitud-resposta, publicació-subscripció, i envia-rebre [66].

En el context de la comunicació Màquina-a-Màquina (M2M) per a aplicacions de l'Internet Industrial de les Coses (IIoT), ZeroMQ ha estat utilitzat per a la implementació d'un mecanisme de missatgeria M2M orientat a dades. Aquesta implementació ha demostrat ser eficaç per a l'accés ubicu a dades en aplicacions industrials riques en sensors, gràcies a la flexibilitat de ZeroMQ per a tractar amb l'arquitectura de sistema jeràrquica i la heterogeneïtat de plataformes [75].

Finalment, ZeroMQ ha estat utilitzat en la creació de biblioteques de primitives de nodes (NEP) en Python, C# i JavaScript. Aquestes biblioteques permeten el desenvolupament fàcil d'arquitectures de programari de robot que són independents de la plataforma. Aquestes biblioteques NEP es basen en ZeroMQ i altres *middlewares* de robot per a proporcionar una comunicació interprocessos lleugera, senzilla, d'alt rendiment, usable i fàcil d'instal·lar [37].

En el context de la nostra aplicació, ZeroMQ ha resultat ser extremadament útil per a la gestió de la comunicació entre diferents parts del sistema. Per exemple si parem atenció, els diversos contenidors o fils que necessiten comunicar-se entre ells per a la classificació de vídeos i ZeroMQ facilita aquesta comunicació de manera eficient i fiable. A més, la capacitat de ZeroMQ per a la comunicació asincrònica ha sigut particularment útil per a tasques com l'anàlisi de vídeo, on pot ser necessari processar grans quantitats de dades de manera asincrònica.

A més, la flexibilitat de ZeroMQ en termes de patrons de comunicació és útil per a diferents aspectes. Per exemple, podem utilitzar el patró de publicació-subscripció per a notificar a diferents parts de l'aplicació quan es completen determinades tasques, o el patró de sol·licitud-resposta per a la comunicació entre el servidor i el client.

La capacitat de ZeroMQ per a la comunicació entre nodes en una xarxa resulta d'utilitat per escalar el projecte i suportar la classificació de vídeos en múltiples màquines o en un entorn de núvol. Això permet processar grans quantitats de vídeos de manera més ràpida i eficient.

3.5 Flask

Flask [83] és un marc de treball de microservicis per a Python que es fa servir per desenvolupar aplicacions web. Es basa en Werkzeug [89], una biblioteca d'utilitats WSGI [84], i Jinja2 [86], que és un motor de plantilles per a Python. Flask és conegut per la seva simplicitat, flexibilitat i fina granularitat del control que ofereix als desenvolupadors. A diferència d'altres marcs de treball com Django, Flask no té cap dependència en un model de base de dades específic o una estructura de projecte específica, la qual cosa el fa ideal per a projectes petits a mitjans que necessiten una gran flexibilitat i control sobre els components de l'aplicació [67].

En el context de la nostra aplicació, Flask s'utilitza com a motor per a la nostra interfície gràfica web. La simplicitat i flexibilitat de Flask el fan ideal per a aquesta tasca, ja que ens permet construir una interfície d'usuari personalitzada que s'ajusti a les necessitats específiques dels nostres usuaris. A més, Flask ofereix una gran varietat d'extensions que podem utilitzar per afegir funcionalitats addicionals a la nostra interfície d'usuari, com ara suport per a formularis web, autenticació d'usuaris, i més [67].

Flask també ofereix suport per a la creació d'APIs RESTful [7], que ens permeten interactuar amb la les aplicacions d'escriptori. Aquesta característica és particularment útil per aquest treball, ja que ens permet crear una interfície d'usuari que pot comunicar-se amb altres servicis de manera eficient i segura [67].

La gran comunitat de desenvolupadors de Flask és una gran font de suport i recursos. Hi ha una abundant quantitat de tutorials, exemples de codi i solucions a problemes comuns disponibles que són d'ajuda quan es tracta de dissenyar i implementar la nostra interfície d'usuari [67].

A més a més, Flask ofereix una gran varietat d'extensions que permeten afegir funcionalitats addicionals a l'aplicació. Aquestes extensions poden afegir suport per a l'enviament de correus electrònics o la connexió a una base de dades, entre altres. Algunes

extensions fins i tot afegeixen marcs de treball complets per ajudar a construir certs tipus d'aplicacions, com ara una API REST [68].

Flask també ofereix una guia per al desenvolupament d'extensions, que permet als desenvolupadors crear les seues pròpies extensions si no troben una que s'ajuste a les seues necessitats. Esta guia proporciona informació sobre com estructurar l'extensió, com inicialitzar-la, com utilitzar l'aplicació actual i com dissenyar l'API de l'extensió [69].

Per tot açò, Flask és útil. En primer lloc, la seva arquitectura lleugera i modular ens permet construir una aplicació eficient que es pot adaptar fàcilment a les nostres necessitats. En segon lloc, el suport de Flask per a APIs RESTful ens permet crear una interfície de programació d'aplicacions que pot interactuar amb la nostra aplicació d'escriptori. Finalment, la gran varietat d'extensions disponibles per a Flask ens permet afegir funcionalitats addicionals a la nostra aplicació de manera fàcil i ràpida.

CAPÍTOL 4
Desenvolupament de la solució

En aquest capítol, ens centrarem en el desenvolupament de la nostra solució per a la gestió i classificació de continguts de vídeo. La nostra solució es basa en una combinació de diverses tecnologies, incloent OpenCV per a la visió per ordinador i l'aprenentatge automàtic, CVAT per a l'etiquetatge de vídeos, Docker per a la portabilitat i consistència de l'aplicació, ZeroMQ per a la comunicació entre processos i la gestió de tasques en paral·lel, i Flask per a proporcionar una interfície d'usuari amigable i eficient.

Aquesta combinació de tecnologies no només ens permet crear una aplicació funcional, sinó que també assegura que la nostra solució siga escalable, mantenible i de qualitat professional. A través d'aquest capítol, detallarem com hem utilitzat aquestes tecnologies per a desenvolupar la nostra solució, i com aquestes interactuen entre si per a proporcionar una experiència d'usuari eficient i eficaç.

4.1 Arquitectura

Aquesta solució està basada en una arquitectura de microservicis. Per aconseguir açò s'han desenvolupat múltiples mòduls que s'encarreguen de dur a terme tasques específiques. Aquests mòduls són executats en paral·lel i es comuniquen fent gastar ZeroMQ.

Cada mòdul es tracta d'un contenidor Docker independent que es podria executar en qualsevol màquina que tinga instal·lat el motor d'execució de Docker; amb açò assolim evitar problemes de dependències així com una major portabilitat del sistema.

Un avantatge d'aquesta arquitectura és la fàcil modificació, actualització i substitució de mòduls, ja que cadascun d'aquests es defineix per la seua imatge Docker. En el cas que haguérem de portar a cap l'actualització d'un mòdul sols hauríem de crear una nova imatge amb allò que volem modificar, fer-la provar tant de forma independent com juntament amb els altres mòduls i aleshores substituir l'antiga imatge per la nova. D'aquesta forma tindríem un dels servicis actualitzats de forma senzilla i resistent a errades al desplegament.

Solucionar els problemes de dependències i portabilitat ens permet que el nostre sistema siga escalable, ja que podem executar tants contenidors com necessitem en qualsevol màquina. Ara mateix aquest servici està dissenyat per ser executat a un únic ordinador de forma local, però en el futur es podria desplegar en un clúster de màquines per a aconseguir una major potència de càlcul així com una major disponibilitat del servici.

D'altra banda, açò també dona a l'usuari la possibilitat que, en el cas que necessite una major potència o que el seu dispositiu no siga capaç d'executar el servici, puga utilitzar un servici en el núvol. Això seria ideal per a poder fer gastar aquesta aplicació en entorns

on no es disposa d'un ordinador amb una gran capacitat de processament o inclús en dispositius mòbils.

Finalment, gràcies a aquest disseny podem trobar interessant la idea d'inclús poder realitzar d'aquesta aplicació un dispositiu físic independent que s'encarregue de desplegar aquest servici per a l'usuari en la seua xarxa local. Aquesta possibilitat seria interessant per a aquells usuaris que no disposen d'un ordinador prou potent i que a més vulguen fer gastar aquesta aplicació en un entorn local. Tot el que faria falta per fer açò funcionar és un dispositiu físic que puga executar Docker i un registre d'imatges local amb aquelles que són necessàries per al funcionament del servici.

4.2 Classe Mailbox

Amb l'objectiu que la comunicació entre els diferents mòduls siga correcta, de qualitat i el més resistent a fallades possible, s'ha desenvolupat una classe de Python auxiliar anomenada *Mailbox*.

Aquesta component auxiliar es tracta d'un sistema de cues per l'emmagatzemament de peticions. Aquestes cues no soles mantenen l'ordre de les peticions, sinó que aconsegueixen evitar omplir la pila TCP/IP, com podria ocórrer en cas que no es creara cap estructura addicional.

A més, aquest objecte Python realitza la seua execució en un fil independent del programa principal quan es troba activament escoltant. D'aquesta forma, el programa principal pot executar altres tasques mentre el fil de la classe *Mailbox* està rebent peticions i emmagatzemant-les.

Quan el programa principal vol acomplir noves tasques, és aleshores quan demana a la classe el paquet més antic emmagatzemat al seu interior. Aquesta petició és bloquejant, és a dir, que el programa principal no pot continuar la seua execució ni la classe pot emmagatzemar noves peticions fins que la classe *Mailbox* no li torna el paquet que li ha demanat. Aquesta característica és fonamental, ja que allà on s'emmagatzemen les peticions és el mateix lloc d'on s'acaben llegint, i s'han d'evitar, per tant, les condicions de carrera a través de l'ús de semàfors.

Aquesta classe està present en tots els components del sistema que tenen algun tipus de comunicació en altres components, de forma que qualsevol mòdul pot estar segur que les seues comunicacions entrants estan protegides de ser sobreescrites per altres peticions.

4.3 Mòduls

Com s'ha comentat amb anterioritat, aquesta solució està composta per múltiples mòduls que s'encarreguen de dur a terme tasques específiques. En la figura 4.1 podem apreciar el conjunt dels mòduls que formen part de l'aplicació, així com una representació de les seues interaccions i jerarquies. Es pot destacar que cadascun d'aquests mòduls es tracta d'un contenidor Docker independent, il·lustrat per una caixa rectangular per mòdul.

En aquest apartat s'analitzaran un per un els diferents mòduls parlant dels seus objectius i funcionalitats, així com en les seues relacions en profunditat. Es començarà pels mòduls amb un nivell en la jerarquia més baix, és a dir, que no invoquen ni fan peticions a altres mòduls sinó que són més aviat utilitzats com a clients. Seguidament, avançarem per aquells que criden a aquests i així apujant fins a arribar al mòdul més alt i que es comunica directament amb l'usuari. Per a cadascun d'aquests es parlarà de les tecnologies

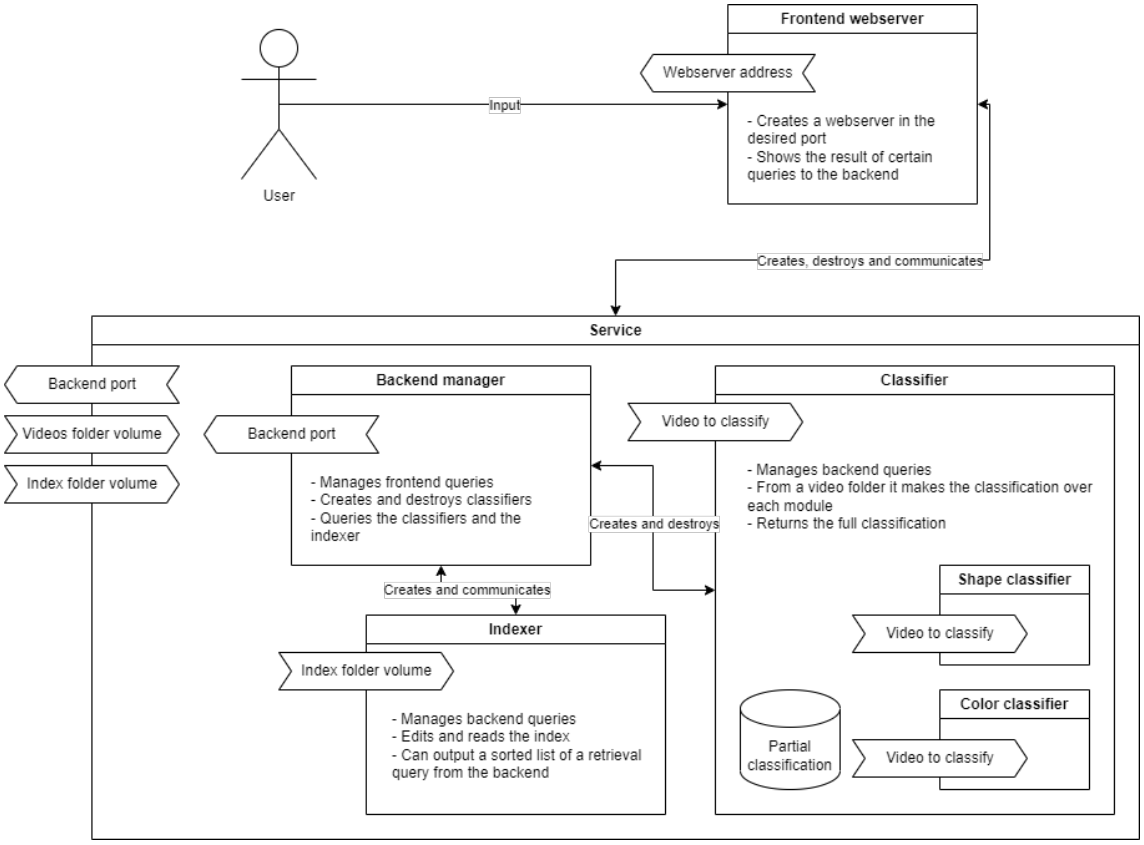

Figura 4.1: Arquitectura de la solució

que el formen, sent que tots tenen en comú que estan formats una imatge Docker, així com de les seues funcionalitats i objectius.

4.3.1. Mòduls de classificació

L'objectiu de l'aplicació és realitzar una classificació de vídeo; per a això s'han desenvolupat dos mòduls que s'encarreguen de dur a terme aquesta tasca. Aquests són els encarregats de portar a cap la classificació parcial dels diferents fotogrames dels vídeos que s'han encarregat classificar. Cadascun d'aquests mòduls té un únic objectiu, el de tornar un diccionari amb dos valors: el tipus de classificació i la llista ordenada dels valors donats a aquesta. Per a aquest treball s'han dessenvolupat dos mòduls diferents, un que s'encarrega de la classificació per color i altra per forma.

Tots dos mòduls estan compostos per dues imatges Docker que instal·len les dependències necessàries per a la seua tasca, així com l'arxiu Python que executaran al moment de la seua instanciació en forma de contenidor. L'únic que necessiten per a funcionar correctament és la disponibilitat en una carpeta muntada de l'arxiu que han d'analitzar.

Una vegada el mòdul completa el seu treball, mana la informació a través d'un missatge ZeroMQ al mòdul classificador (del qual es parla a la secció 4.3.2) i, a l'acabar la seua execució, es destrueix sense deixar rastre.

Color

El mòdul de color és l'encarregat de tornar una paleta de colors representativa del contingut del vídeo. Aquesta tasca s'ha estudiat a la secció 2.2, on s'han comentat altres estudis que feien gastar l'histograma de color per extraure informació rellevant del contingut

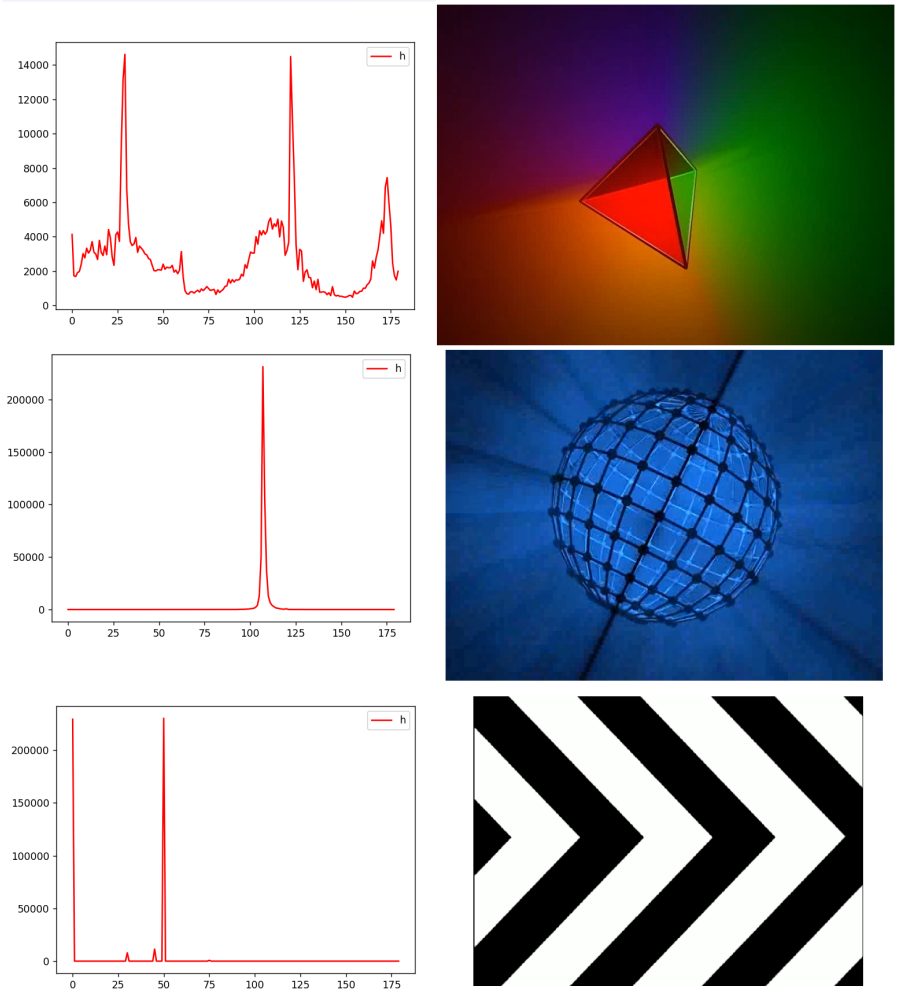

Figura 4.2: Proves amb l'histograma de tonalitat

del vídeo. Aquest mòdul pretén realitzar una funció diferent, però on la informació a extraure és un subconjunt dels colors del vídeo.

Agafant inspiració del treball de Şaykol, Güdükbay, i Ulusoy [11], presentat en 2005, on es fan servir diversos histogrames per a elaborar l'anàlisi del contingut del vídeo i la seua posterior recuperació, podem obtenir l'histograma de color a l'espai *Hue Saturation Value* (HSV). Aquest tipus d'histograma permet extraure una característica del color en concret de les tres que descriu, la tonalitat o *hue*.

Amb aquesta component podem diferenciar els colors de forma clara en l'espectre a partir d'un únic valor, ja que independentment de la seua saturació o valor, el color es manté. I, mentre que açò és cert per a un gran nombre de casos on els colors, per a aquells amb valors extrems de les altres dos components (saturació i valor), es complica la seua diferenciació a partir del seu valor de tonalitat. Com podem apreciar, la figura 4.2 il·lustra molt bé aquest problema en certs tipus de continguts, i mostra per què no hauria de ser la forma d'escollir la paleta representativa de colors a partir dels valors pic d'aquest histograma.

Darrerament, es va proposar una aproximació basada en l'espai RGB en conjunt, és a dir, realitzar un histograma que contara les aparicions d'un cert valor RGB combinat. Aquesta aproximació, tot i que més costosa computacionalment pel que fa a espai, permetia una millor diferenciació dels colors i, per tant, una millor representació del contingut

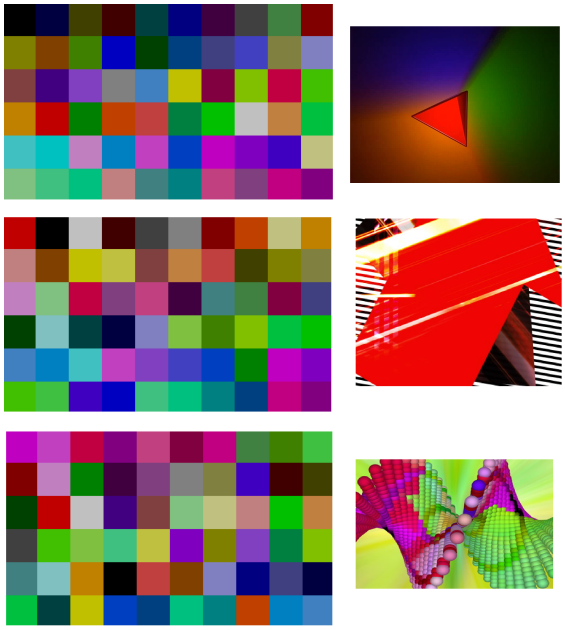

Figura 4.3: Primers resultats d'escollir els 70 colors més abundants en un espai de 4x4x4 colors

multimèdia. Així i tot, suposava un gran cost espacial per a representar la informació, ja que per a cada fotograma s'havia de guardar un histograma de 256x256x256 valors.

Ràpidament, es va veure que tanta precisió no era necessària per a representar els colors. Encara que dos colors siguen diferenciables pel seu valor RGB no vol dir que siguen diferenciables per l'ull humà. D'aquesta forma, es va suggerir una alternativa similar que reduïa l'espai RGB agrupant els colors en un nombre menor de grups. Aquest canvi va provocar no sols una millora en l'espai necessari per a guardar la informació, sinó també una millora en la diferenciació dels colors, creant paletes més significatives on els diferents colors d'aquesta són realment diferents entre si.

Com podem apreciar a la figura 4.3, el resultat d'aquesta aproximació és una paleta de colors que depenent del contingut de la suma dels fotogrames aconsegueix una representació més o menys precisa del contingut. Aquest experiment va donar un resultat satisfactori per a la majoria dels vídeos, encara que el principal repte ara era assolir extraure la quantitat de colors representativa del contingut. Per exemple, a la figura 4.3 podem veure que la primera paleta deixa de ser representativa a partir del color número 30 aproximament, mentre que a la segona ho és a partir del tercer color.

Per a solucionar aquest problema es va proposar una solució basada en la suma de les diferents paletes de colors de cada fotograma. El procés computacionalment és molt similar. En primer lloc, per a cada fotograma es calcula la paleta de colors agafant els colors que representen un alt percentatge del total de colors del fotograma. Açò es fa per a cada fotograma i, una vegada tots els fotogrames estan analitzats, se sumen les diferents paletes de colors i es torna a calcular la paleta a partir de les aparicions de cada color en les paletes dels diferents fotogrames.

Aquest procés ajudava en múltiples dimensions. La primera és que aconseguia obtindre un nombre significatiu de colors; si el vídeo compta amb multitud de colors diferents i igual de presents es mostren tots a la paleta final, sense deixar cap pel camí. D'altra banda, si el vídeo era principalment monocromàtic amb poquetes tonalitats, la paleta també passa a ser més reduïda amb els colors que conformen el contingut. Finalment, és capaç de destacar els colors primaris del contingut del vídeo a les primeres posicions de la paleta, així com assolir colors secundaris també importants i diferents dels primaris.

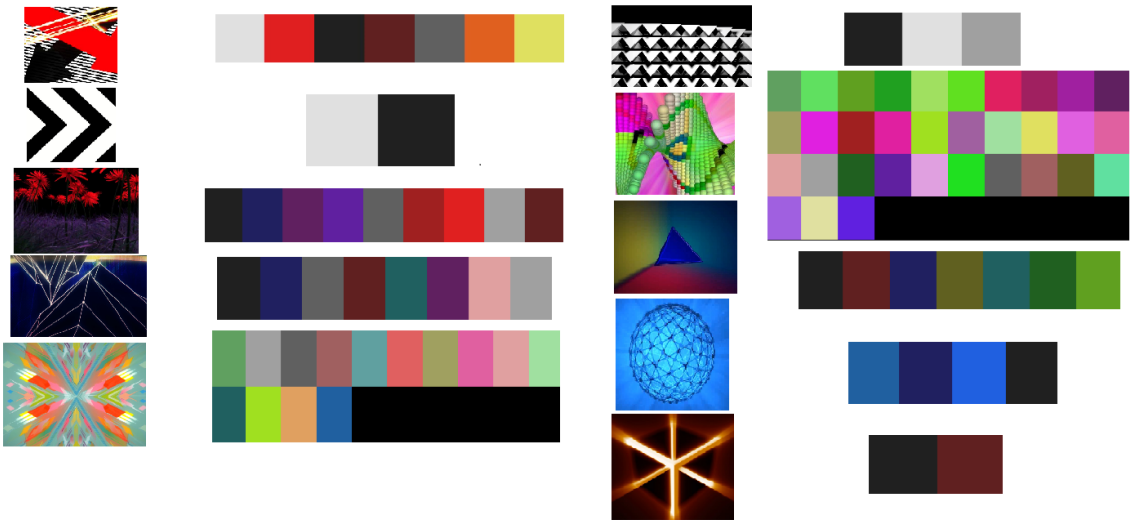

Figura 4.4: Resultats d'aplicar la tècnica de suma de paletes de colors

Tot açò fa que aquesta tècnica siga ideal per acomplir aquesta tasca. Com podem apreciar a la figura 4.4, els resultats són més que satisfactoris i s'aconsegueix l'objectiu a la perfecció. A més, el procés és molt ràpid i computacionalment poc costós, per la qual cosa es pot aplicar a tots els vídeos que es desitgen sense cap problema. Així i tot, no és necessari aplicar-ho a tots els fotogrames, ja que molta de la informació és redundant i, encara que normalitzada no afecta els resultats, sí que afecta el temps de computació. Per això, aquests experiments s'han portat a cap fent gastar 5 fotogrames per segon a l'hora d'elaborar l'anàlisi del contingut i han donat uns resultats idèntics amb un temps de computació molt menor.

Gràcies a aquesta investigació s'ha pogut obtindre una tècnica que permet obtindre una paleta de colors representativa del contingut del vídeo de forma ràpida i eficient.

Forma

El mòdul de forma és l'encarregat de tornar aquells objectes trobats al contingut. Arran de l'estudiat a la secció 2.1 s'ha decidit fer gastar YOLO com a model de detecció d'objectes. Per aquesta decisió s'han tingut en compte diversos factors. El primer és que YOLO és un model molt ràpid i eficient, per la qual cosa no suposa un problema a l'hora de fer servir-lo en temps real. A més, és un model que permet detectar múltiples objectes en una sola passada, per la qual cosa no és necessari fer servir un model per a cada objecte que es vol detectar. En últim lloc, cal destacar la seua facilitat d'ús, ja que és un model que amb un conjunt de dades i unes quantes línies de codi permet entrenar un model per a detectar els objectes que es desitgen.

El principal problema és crear un conjunt de dades d'entrenament que siga representatiu dels objectes que es volen detectar, així com ampli per tindre una bona precisió. D'aquesta forma, en un inici es va decidir confeccionar l'etiquetatge d'un conjunt de recursos de vídeo disponibles a l'empresa a partir de l'ús de CVAT. La tasca en qüestió consistia a anar fotograma a fotograma i etiquetar els objectes que apareixien a la imatge d'un conjunt de classes predefinides. Encara que els vídeos es van reduir per solament fer etiquetatge d'ells a 5 fotogrames per segon, ràpidament ens adonarem que, en cas de fer correctament aquesta tasca, es necessitaria molt de temps.

Per aquestes raons, s'ha decidit que, fins que es poguera trobar un conjunt de dades d'entrenament adequat, es faria servir un model de detecció d'objectes ja entrenat perquè servira de prova que el sistema funciona correctament. Aquest model es podria canviar posteriorment per l'adequat per a la tasca de vídeo.

La implementació del mòdul consisteix en el seu contenidor amb un *script* de Python que, fent gastar la llibreria YOLO-V8 de UltraLytics [73], aconsegueix detectar els objectes que apareixen a un fotograma i tornar la llista de deteccions que es van acumulant. Quan s'han trobat tots els objectes als fotogrames es torna la llista final a través de ZeroMQ cap al mòdul que es descriu a la secció 4.3.2 i finalitza la seua execució destruint el contenidor. D'aquesta forma, aquest mòdul és capaç, a partir d'un model preestablert al seu contenidor, de detectar els objectes que apareixen a un vídeo.

4.3.2. Classificador

Seguidament, el mòdul de classificació és l'encarregat de tornar la classificació total d'un vídeo a partir d'arreplegar les classificacions dels mòduls de classificació i tornar aquesta informació de forma total al *manager*. Aquesta component està formada per la imatge que conté les dependències necessàries per a executar el codi de Python que s'encarrega de crear el servidor de ZeroMQ i rebre les dades resultants de la classificació i tornar-les al *manager*.

Per cada vídeo a classificar s'ha de fer còrrer un contenidor de classificació. D'aquesta manera podem paral·lelitzar la classificació dels vídeos y aconseguir que el temps de classificació siga menor. Aquesta mesura és clau perquè el temps d'execució del sistema no siga molt gran i pugam aconseguir resultats de rendiment molt majors.

Aquest és un dels dos mòduls que compta amb una característica especial. Al costat del servici, la imatge d'aquesta component parteix d'una altra imatge de tipus *Docker In Docker* [65]. Mentre que les imatges dels mòduls de classificació i servici resulten ser de tipus *Docker In Docker*, els mòduls de tipus *manager* i *webserver* són de tipus on es compareix el *socket* d'execució del *Docker Engine*.

La raó d'aquesta estructura be donada per l'arquitectura de l'aplicació. Els mòduls que tenen una imatge de tipus *Docker In Docker* necessiten contindre en si mateixos, ja que la parada del contenidor pare suposa la parada dels contenidors fills. D'altra banda, aquells que comparteixen el mateix *socket* d'execució del *Docker Engine* en realitat executen altres contenidors que estan al mateix nivell que ells mateixos en l'arquitectura de l'aplicació. Tot açò es pot apreciar a la figura 4.1, on es pot veure clarament aquestes diferències.

4.3.3. Indexador

Aquest mòdul indexador és l'encarregat d'acomplir dues tasques fonamentals: arreplegar la informació de la classificació dels vídeos analitzats i tornar el resultat els resultats de la cerca per a les peticions que faça l'usuari.

La primera de les dues tasques es duu a terme a través de la creació d'un servidor de ZeroMQ que escolta les peticions que el component *manager* li envia. Aquestes peticions contenen la informació de la classificació dels vídeos analitzats i són emmagatzemades a l'arxiu d'indexació de l'usuari del servici. És important que tant la lectura com l'escriptura sobre aquest arxiu sols puga ser duta a terme per aquesta component, perquè mai obtinguem situacions de lectura i escriptura simultània. Per això aquestes operacions són crítiques i han de recaure sobre un sol component que les porte a cap de forma controlada.

D'altra banda, quan un usuari fa una petició de cerca, l'indexador s'encarrega de llegir l'arxiu de l'usuari i tornar-li els resultats de la cerca. Aquesta tasca es porta a cap a través d'una cerca al diccionari de vídeos dels termes buscats per l'usuari i una posterior puntuació dels vídeos si concedeixen amb els termes de cerca. Aquesta cerca en l'actualitat se soluciona a través d'una petició senzilla on s'especifiquen les formes que cerca l'usuari i els colors que vol aconseguir als vídeos. Així doncs, es troba al diccionari aquells vídeos que compleixen amb els termes i, en cas que diversos termes siguen trobats al mateix vídeo, aquest és puntuat més alt per cadascun. D'aquesta forma l'indexador ens pot tornar una llista de vídeos ordenats per puntuació depenent de tan propers que són al que ha demanat l'usuari.

En últim lloc, cal destacar que l'indexador quan és invocat també té l'objectiu a comunicar al *manager* quins vídeos de la carpeta encara no han sigut classificats i haurien de passar pels classificadors. Aquest procés es pot també demanar per part del *manager* de forma voluntària en cas que l'usuari, per exemple, haja introduït nous vídeos en la carpeta.

4.3.4. Manager

Seguidament, el mòdul *manager* és l'encarregat de director d'orquestra d'un servici Avitag. Aquest component és el que s'encarrega de coordinar les peticions d'usuari realitzades a través del *webserver*, les peticions de classificació dels vídeos a l'indexador així com els resultats de la classificació que el classificador li envia.

Aquesta component està formada per una imatge de Docker amb el que és necessari per a fer funcionar el seu *script* de Python. Aquest *script* és el que s'encarrega de coordinar les peticions que li arriben dels altres components i de tornar els resultats de les peticions que li arriben dels classificadors. Aquesta comunicació es duu a terme a través de ZeroMQ, on el *manager* té un *socket* per a cadascun dels diferents tipus de components que es comuniquen amb ell.

Com a component de coordinació, el seu punt crític es troba en intentar mantindre una comunicació tan fluida com siga possible i que no es perda cap petició. És aquí més important que mai l'ús de la component *mailbox* per a cada un dels tipus de peticions que rep. Així doncs, podem assegurar la correcta comunicació amb els altres components i que no es perden peticions.

4.3.5. Servici

Aquest mòdul és diferent de la resta tant en objectiu com en idiosincràsia. Mentre que la resta de mòduls tenen en comú que tenen com a finalitat executar un *script* de Python, aquest no realitza res d'això. L'única funció del servici és fer d'embolcall per a la resta de mòduls que no són el *webserver*, de forma que s'executen en un ambient controlat i aïllat del dispositiu que l'execute.

Al igual que el mòdul classificador, la imatge que forma aquest mòdul està basada en una imatge de tipus *Docker In Docker*. D'aquesta forma, l'únic que ha de fer aquesta imatge al instanciar-se en forma de contenidor és executar una instància de la component *manager* per tal que es comunique amb el *webserver*.

Podem veure com funciona d'embolcall a la resta de mòduls en la figura 4.1.

4.3.6. Web server

Finalment, el mòdul amb més visibilitat per part de l'usuari és aquest. Quan l'aplicació s'executa en un dispositiu en realitat està realitzant l'execució d'aquesta imatge que conté un *script* de Python amb una aplicació web. Gràcies a l'ús de Flask i a la interfície creada en HTML, CSS i JavaScript, l'usuari pot relacionar-se amb el servici d'etiquetatge i cercar de forma ràpida, intuïtiva i eficaç.

Aquest component executa el servidor escoltant a la IP i port designats a la imatge de Docker. D'aquesta forma, si es designa la IP de l'ordinador en el qual s'executa el contenidor, així com un port, es pot accedir a la interfície sempre que siga accessible a través de la xarxa des de qualsevol dispositiu que compte amb un navegador.

La portabilitat que suposa açò per a l'usuari és extremadament alta, així com convenient en certs entorns de treball. Així doncs, l'usuari pot executar el servici en un ordinador de sobretaula i accedir a ell des del seu ordinador portàtil, des d'un dispositiu mòbil o des d'un ordinador remot. Inclús es podria desenvolupar infraestructura perquè l'usuari puga accedir de forma remota a través d'internet amb l'ús d'alguna VPN o similar que li permetera accedir a la xarxa on s'executa el servici.

Interfície gràfica d'usuari

La interfície desenvolupada consta d'una única pantalla principal on la informació que es mostra varia dinàmicament enfront del que ocorre al servici. Aquesta pantalla està composta per tres elements: una barra superior d'informació i configuració, una barra lateral amb les opcions per realitzar una cerca i una zona central que pot mostrar els resultats de la cerca.

En la barra superior es mostra una xicoteta barra sempre que al servici està ocorrent una cosa, així com una xicoteta infografia que il·lustra l'estat d'aquest. També podem trobar un botó per a accedir a la configuració del servici i així canviar diferents opcions a aquest.

A la barra lateral podem observar les diferents categories, entre les quals poder filtrar i cercar el nostre contingut classificat. L'usuari pot trobar ací el llistat d'etiquetes del model de classificació actual, així com un apartat on afegir i llevar colors que s'aplicaran com a filtre als vídeos cercats. En últim lloc, a la zona central de la pantalla es mostra el resultat de la cerca. Ací es llistaran els vídeos resultats d'aplicar el filtre de l'usuari amb la seua ruta al sistema.

Tot açò configura una interfície amigable i eficaç perquè el client puga fer gastar l'aplicació al seu gust i aconseguir els millors resultats possibles en el menor temps possible.

4.4 Usabilitat, intercomunicació i procés

El procés de comunicació entre els diferents mòduls que formen el sistema està sustentat sobre dos pilars: la creació de contenidors i la comunicació entre ells. Per fer entendre com funciona l'aplicació al seu conjunt suposarem un escenari on l'usuari executa el servici en el seu ordinador de sobretaula i accedeix a ell per classificar una carpeta plena de contingut de vídeo i després passa a cercar en aquests emprant diferents filtres.

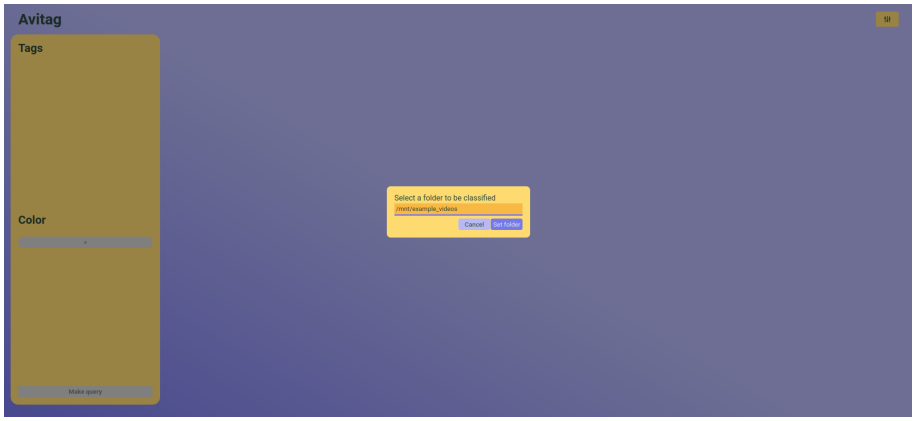

Figura 4.5: Interfície d'usuari a l'arrancada demanant la carpeta a classificar

4.4.1. Arrencada del servici

En primer lloc, l'usuari executa el contenidor *webserver* en el seu ordinador. Aquest contenidor permet connectar l'usuari a través d'un navegador web a l'aplicació web que s'executa en el contenidor. Una vegada ací el sistema demana al client que introduïsca quina carpeta serà l'escollida per realitzar la classificació com podem observar a la figura 4.5. Aquesta acció fa que el contenidor *webserver* execute altre contenidor de tipus servici compartint la carpeta escollida per l'usuari, així com la carpeta on s'emmagatzema l'índex a la carpeta per defecte del client.

El servici es crea i executa la creació dins seu d'un contenidor de tipus *manager* que s'encarrega de, al seu torn, executar un contenidor de tipus indexador. Aquest mòdul executat té com a primera tasca la d'analitzar el directori i el fitxer d'indexació (si existeix) per tal de determinar la llista de vídeos a classificar. Una vegada això s'ha dut a terme, el mòdul indexador s'encarrega de manar al *manager* aquesta llista.

4.4.2. Classificació de contingut

Una vegada el *manager* rep la llista de vídeos a classificar, aquest comença a executar contenidors de tipus classificador on li assigna a cadascun d'ells un vídeo de la llista. Aquest contenidor classificador, al seu torn, té com a objectiu executar els mòduls de classificació que s'encarreguen de classificar el contingut i manar, en finalitzar, el resultat al classificador.

Finalment, quan el mòdul classificador rep el resultat de tots els mòduls de classificació que ha invocat, implica manar el resultat final al mòdul de tipus *manager*. Aquest mòdul ara s'encarregarà de manar la informació de classificació obtinguda i manar-la al mòdul indexador perquè l'emmagatzeme al fitxer d'indexació.

Paral·lelament a aquest procés ens trobem que durant l'execució dels mòduls de classificació, aquests manen de forma continuada i repetitiva el seu progrés en el treball que se'ls ha assignat. Aquesta informació és recopilada pel seu mòdul classificador que l'encapsula en un missatge que fa arribar també al *manager*. Mentres tant, l'usuari pot observar a la barra d'estat aquest procés com podem observar a la figura 4.6.

L'objectiu d'açò és que el propi *manager* puga fer arribar de forma continuada al *webserver* aquesta informació perquè així l'usuari puga rebre alguna mena de retroalimentació sobre el procés de classificació del seu contingut. Aquesta tasca és essencial perquè el client puga observar el progrés d'una tasca que pot arribar a tardar moltes hores en el cas que comptem amb una gran quantitat de contingut a analitzar. A més, pot fer sentir a

Figura 4.6: Interfície d'usuari durant la classificació del contingut

Figura 4.7: Interfície d'usuari una vegada el contingut s'ha classificat correctament

l'usuari una sensació d'avanç i acompliment molt satisfactòria que pot millorar la relació de l'aplicació amb aquest.

4.4.3. Cerca del contingut

Una vegada l'usuari ja ha classificat el seu contingut, el sistema li permet realitzar cerques sobre aquest. Quan cap vídeo queda sense classificar, el mòdul indexador mana al *manager* l'ordre que està llest per començar a portar a cap cerques sobre el contingut. Aquest missatge ve acompanyat d'informació relacionada amb les possibilitats de cerca que, en el nostre cas, passa a ser la llista d'etiquetes disponibles per a l'usuari. Podem observar un exemple d'açò a la figura 4.7.

A partir d'aquest moment l'usuari pot fer cerques sobre el contingut a través de la pàgina web que li ofereix el *webserver*. El client pot seleccionar les etiquetes que voldria que apareguessin als vídeos, així com els colors, i, una vegada faça clic al botó de cerca, començarà el procés de recuperació de la informació desitjada.

Quan açò ocorre el *webserver* recopila la informació introduïda i fa una petició al *manager* perquè aquest la complete. Tanmateix, el *manager* s'encarrega de manar la petició l'indexador que tornarà el llistat ordenat de vídeos que respon a eixa cerca. De la mateixa forma, el *manager* s'encarregarà de recopilar-la i manar-la al *webserver* que la mostrarà de forma ordenada al navegador de l'usuari. Un exemple de resultat el podem vore a la figura 4.8.

Figura 4.8: Interfície d'usuari per a un exemple de resultat de cerca

CAPÍTOL 5
Conclusions i treballs futurs

El projecte ha estat un èxit en molts aspectes, però també ha trobat alguns reptes que encara estan per resoldre.

S'ha aconseguit desenvolupar una aplicació eficient per a la gestió i classificació de grans quantitats de continguts de vídeo, gràcies a la combinació estratègica de diverses tecnologies, incloent-hi OpenCV, CVAT, Docker, ZeroMQ i Flask. Cadascuna d'aquestes eines ha aportat una funcionalitat específica que, en conjunt, ha permés el desenvolupament d'una aplicació d'escriptori multiplataforma robusta i escalable.

No obstant això, dos objectius importants no s'han assolit completament. En primer lloc, el model de classificació no s'ha pogut enfocar específicament a recursos de vídeo per a tècnics de so. Aquesta limitació es deu principalment a la gran tasca que suposa crear un conjunt de dades vàlid per a aquesta tasca específica. En segon lloc, l'aplicació encara no suporta la cerca a través de llenguatge natural, una funcionalitat que podria millorar significativament la usabilitat de l'aplicació.

A pesar d'aquests reptes, el projecte ha demostrat una gran flexibilitat i potencial per a la millora i l'expansió. Actualment, el servici està dissenyat per ser executat en un únic ordinador de forma local, però en el futur es podria explorar la possibilitat de desplegar el servici en un clúster de màquines o en el núvol per aconseguir una major potència de càlcul i una major disponibilitat del servici.

El desenvolupament d'aquest projecte ha permés l'aplicació de diverses competències adquirides durant el grau. En primer lloc, s'han utilitzat una varietat de tecnologies, com OpenCV, CVAT, Docker, ZeroMQ i Flask, per desenvolupar una aplicació d'escriptori multiplataforma per a la gestió i classificació de grans quantitats de continguts de vídeo. Aquesta aplicació requereix un coneixement profund de la programació, la visió per ordinador, l'aprenentatge automàtic i altres àrees tècniques, demostrant així les competències tècniques adquirides en assignatures com Programació, Aprenentatge Automàtic i Enginyeria del programari, entre altres.

Aquest projecte ha requerit una gran capacitat de resolució de problemes. S'ha identificat el problema de la gestió i classificació de grans quantitats de continguts de vídeo i s'ha desenvolupat una solució eficient per a aquest problema fent servir una combinació estratègica de diverses tecnologies. Aquesta capacitat de resoldre problemes complexos és una competència clau que s'ha adquirit durant el grau.

El projecte també ha demostrat un fort pensament crític. S'han analitzat les necessitats del projecte i s'han seleccionat les tecnologies més adequades per a cada tasca. A més, s'han identificat les limitacions de la solució i s'han proposat possibles millores per a futures versions de l'aplicació. Aquesta capacitat d'analitzar i millorar les solucions desenvolupades és una part essencial de la formació com a enginyer informàtic.

Finalment, el projecte ha permés l'aplicació de diverses matèries estudiades al grau. Per exemple, els coneixements adquirits en matèries com Percepció o Sistemes Distribuïts, on la visió per ordinador i el disseny de sistemes han estat essencials per al desenvolupament de l'aplicació. Aquesta aplicació pràctica dels coneixements teòrics adquirits durant la meua formació demostra la rellevància i utilitat d'aquests estudis.

En general, aquest projecte ha sigut una oportunitat per a aplicar els coneixements adquirits durant el grau en un projecte real. A més, ha permés desenvolupar competències clau per a la formació com a enginyer informàtic, com la capacitat de resolució de problemes, el pensament crític i l'aplicació pràctica dels coneixements teòrics.

Pel que fa als èxits de l'aplicació, cal destacar la seua capacitat per a gestionar grans quantitats de contingut de vídeo, la seua robustesa i la seua escalabilitat. A més, l'aplicació és multiplataforma i pot ser executada en ordinadors amb diferents sistemes operatius. Aquesta característica és molt important per a la seua usabilitat i portabilitat, així com per a poder ser utilitzada per una gran quantitat d'usuaris i que la disponibilitat d'un dispositiu o altre que siga un limitant per a la seua utilització.

D'altra banda, l'extracció de la paleta de color dels vídeos ha estat un èxit. Aquesta funcionalitat és molt útil per a trobar vídeos que puguen contindre les tonalitats que l'usuari desitja. La investigació d'aquest tipus de funcionalitat ha sigut molt interessant i ha permés aprendre sobre la teoria del color i la seua aplicació en el camp de la classificació de vídeo.

Tot i que el projecte ha aconseguit molts dels seus objectius inicials, encara hi ha reptes rellevants per afrontar. La creació d'un conjunt de dades vàlid per a la classificació de vídeos per a tècnics de so i la implementació de la cerca a través de llenguatge natural són àrees clau per a la millora en el futur. Altres reptes crucials que podrien ser abordats en el futur són múltiples i variats.

Pel que fa a la interfície d'usuari, l'aplicació podria fer gastar un altre marc de treball més adaptat a tecnologies actuals, per exemple Electron [70]. Canviar a aquest tipus de eines permetria que l'aplicació fora més similar al que un usuari està acostumat a treballar. Eliminaria aleshores la necessitat d'utilitzar Flask com a servidor web i podria, a través de NodeJS, executar el servici de forma local a l'ordinador de l'usuari o donar l'opció d'executar aquest servici en un servidor en línia.

Seguint pel servici de classificació, seria interessant que poguérem trobar nous camps de classificació més enllà de les etiquetes de forma i els colors. Aquests camps podrien ser els sentiments que evoca el vídeo o inclús les direccions principals de moviment que podem trobar. La combinació inclús d'aquesta última amb les etiquetes podria donar lloc a cerques molt interessant com: «m'agradaría un objecte que es menejara en certa direcció», arribant així a un nivell de personalització molt alt del contingut recuperat.

En últim lloc, pensant en l'emmagatzematge de les dades, seria interessant que el sistema poguera emmagatzemar informació sobre els usuaris, les seues classificacions i les seues cerques en forma de bases de dades més complexes i escalables. Això permetria que el sistema poguera aprendre dels usuaris i oferir-los contingut més personalitzat, així com permetre la possibilitat que aquesta informació fora més fàcilment manejable i potencialment portable i sincronitzable amb altres dispositius.

Malgrat açò, la flexibilitat de l'arquitectura del projecte i la seua capacitat per adaptarse a diferents entorns de desplegament són factors clau que poden facilitar aquesta expansió. Aquest projecte suposa el primer pas per crear un servici que pot ser de gran utilitat per a l'empresa en el que s'ha desenvolupat i que pot estalviar moltes hores de treball als seus empleats i potencials clients en el futur.

Bibliografia

[1] Karen Sparck Jones. "A statistical interpretation of term specificity and its application in retrieval". A: *Journal of"'latex Documentation* 28.1 (1972), pàg. 11 - 21. DOI: 10.1108/eb026526. URL: https://dl.acm.org/doi/pdf/10.3115/981967.981981.

[2] Stephen Robertson. "The probability ranking principle in IR". A: *Journal of Documentation* 33.4 (1977), pàg. 294 - 304. DOI: 10.1108/eb026647. URL: https://academic.oup.com/comjnl/article-pdf/35/3/268/1406415/35-3-268.pdf.

[3] Edward A. Fox. *Characterization of Two New Experimental Collections in Computer and Information Science Containing Textual and Bibliographic Concepts*. Inf. tèc. Cornell University, 1983. URL: https://ecommons.cornell.edu/handle/1813/6401.

[4] James A. Anderson. *An Introduction to Neural Networks*. MIT Press, 1995.

[5] Peter Schäuble. *Content-Based Information Retrieval from Large Text and Audio Databases*. Vol. 397. Springer International Series in Engineering and Computer Science. Springer, 1997.

[6] National Institute of Standards i Technology. *Automatic Indexing: An Experimental Inquiry*. Inf. tèc. 1997. URL: https://www-nlpir.nist.gov/works/pubs/ir4873.html.

[7] Roy Thomas Fielding. *Representational state transfer (REST)*. https://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm. Accessed: 2023-06-28. 2000.

[8] R.C. Gonzalez i R.E. Woods. "Color Image Segmentation Using Adaptive Hierarchical-Histogram Thresholding". A: *Sensors* 12.9 (2002), pàg. 12489 - 12500. DOI: 10.3390/s120912489. URL: https://www.mdpi.com/1424-8220/12/9/12489/pdf?version=1403318288.

[9] Stephen E Robertson i Hugo Zaragoza. "Probabilistic models of information retrieval based on measuring the divergence from randomness". A: *ACM Transactions on Information Systems (TOIS)*. Vol. 20. 4. ACM New York, NY, USA. 2002, pàg. 357 - 389.

[10] Roger S. Pressman. *Software Engineering: A Practitioner's Approach*. Palgrave Macmillan, 2005.

[11] Ediz Şaykol, Uğur Güdükbay i Özgür Ulusoy. "A histogram-based approach for object-based query-by-shape-and-color in image and video databases". A: *Image and Vision Computing* 23.13 (2005), pàg. 1170 - 1180. URL: https://www.sciencedirect.com/science/article/pii/S026288560500123X.

[12] Christopher D Manning, Prabhakar Raghavan i Hinrich Schütze. "Introduction to Information Retrieval". A: *Natural Language Engineering* 16.1 (2008), pàg. 100 - 103. DOI: 10.1017/S1351324909990224. URL: https://link.springer.com/content/pdf/bfm:978-3-031-02155-8/1?pdf=chapter%20toc.

[13] S.J. Preece et al. "A comparison of feature extraction methods for the classification of dynamic activities from accelerometer data". A: *IEEE Trans. Biomed. Eng.* 56 (2008), pàg. 871 - 879.

[14] O. Duchenne et al. "Graph-based object recognition in video sequences". A: *Pattern Recognition Letters* 30.12 (2009), pàg. 1082 - 1086. URL: http://liris.cnrs.fr/Documents/Liris-4735.pdf.

[15] Pedro F Felzenszwalb et al. "Object Detection with Discriminatively Trained Part-Based Models". A: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 32.9 (2010), pàg. 1627 - 1645. URL: http://vision.princeton.edu/projects/2014/SlidingShapes/paper.pdf.

[16] Andreas Geiger et al. *KITTI Vision Benchmark Suite*. http://www.cvlibs.net/datasets/kitti/. 2012.

[17] K.K. Reddy i M. Shah. "Recognizing 50 human action categories of web videos". A: *Machine Vision and Applications* 24 (2013), pàg. 971 - 981.

[18] Dzmitry Bahdanau, Kyunghyun Cho i Yoshua Bengio. "Neural machine translation by jointly learning to align and translate". A: *arXiv preprint arXiv:1409.0473* (2014).

[19] V. Kantorov i I. Laptev. "Efficient feature extraction, encoding, and classification for action recognition". A: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. Columbus, Ohio: Institute of Electrical i Electronics Engineers, 2014, pàg. 2593 - 2600.

[20] Shaoqing Ren et al. "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks". A: *Advances in Neural Information Processing Systems* 28 (2015). URL: http://arxiv.org/pdf/1506.01497.

[21] Y.D. Zhang, Z. Chi i F. He. "An Adaptive Unsupervised Approach Toward Pixel Clustering and Color Image Segmentation". A: *PLOS ONE* 10.10 (2015), e0240015. DOI: 10.1371/journal.pone.0240015. URL: https://journals.plos.org/plosone/article/file?id=10.1371/journal.pone.0240015&type=printable.

[22] Ian Goodfellow, Yoshua Bengio i Aaron Courville. "Deep Learning". A: *MIT Press* (2016). URL: http://www.deeplearningbook.org/.

[23] P. Navarro et al. "A Machine Learning Approach to Pedestrian Detection for Autonomous Vehicles Using High-Definition 3D Range Data". A: *Sensors* (2016). DOI: 10.3390/s17010018. URL: https://dx.doi.org/10.3390/s17010018.

[24] Joseph Redmon et al. "You Only Look Once: Unified, Real-Time Object Detection". A: (2016). arXiv: 1506.02640 [cs.CV]. URL: https://arxiv.org/abs/1506.02640.

[25] K. Ren et al. "Faster r-cnn: towards real-time object detection with region proposal networks". A: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39.6 (2016), pàg. 1137 - 1149.

[26] Kathleen F. Weaver et al. *An Introduction to Statistical Analysis in Research: With Applications in the Biological and Life Sciences*. John Wiley & Sons, 2017.

[27] S. Bekhet i A. Ahmed. "An integrated signature-based framework for efficient visual similarity detection and measurement in video shots". A: *ACM Transactions on Information Systems* 36.4 (2018), pàg. 1 - 38.

[28] S. Bekhet i A. Ahmed. "Graph-based video sequence matching using dominant colour graph profile (DCGP)". A: *Signal, Image and Video Processing* 12.2 (2018), pàg. 291 - 298.

[29] V. Mygdalis et al. "Semi-supervised subclass support vector data description for image and video classification". A: *Neurocomputing* 278 (2018), pàg. 51 - 61.

[30] S. Bekhet i A. Ahmed. "Video similarity detection using fixed-length statistical dominant colour profile (SDCP) signatures". A: *Journal of Real-Time Image Processing* 16.6 (2019), pàg. 1 - 16.

[31] W. Dan i X. Weihua. "Short Video Classification Based on Spatio-Temporal Features and SVM". A: *Proceedings of the IEEE/ACIS 18th International Conference on Computer and Information Science*. Beijing: Institute of Electrical i Electronics Engineers, 2019, pàg. 493 - 496.

[32] Caiwen Ding et al. "REQ-YOLO: A Resource-Aware, Efficient Quantization Framework for Object Detection on FPGAs". A: *Proceedings of the 2019 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays*. ACM. 2019, pàg. 249 - 258.

[33] D. G. Harkut i K. Kasat. "Introductory Chapter: Artificial Intelligence - Challenges and Applications". A: (2019). URL: https://www.intechopen.com/citation-pdf-url/66147.

[34] W. Jianbin et al. "Coverless Information Hiding Algorithm Based on Image Classification". A: *Journal of Hunan University Natural Sciences* 46.12 (2019), pàg. 25 - 32.

[35] Muhammad Suleman Khan i Jianguo Zhang. "Content based Automatic Video Genre Identification". A: *International Journal of Advanced Computer Science and Applications* 10.6 (2019), pàg. 77 - 85. URL: http://thesai.org/Downloads/Volume10No6/Paper_77-Content_based_Automatic_Video_Genre_Identification.pdf.

[36] Ayesha Nadeem et al. "Securing Cognitive Radio Vehicular Ad Hoc Networks (CR-VANETs) Using Blockchain Technology". A: *International Journal of Advanced Computer Science and Applications* 10.1 (2019), pàg. 138 - 147. DOI: 10.14569/IJACSA.2019.0100138. URL: http://thesai.org/Downloads/Volume10No1/Paper_38-Securing_Cognitive_Radio_Vehicular_Ad_Hoc_Network.pdf.

[37] Enrique Coronado i G. Venture. "Towards IoT-Aided Human-Robot Interaction Using NEP and ROS: A Platform-Independent, Accessible and Distributed Approach". A: *Sensors* 20.5 (2020), pàg. 1500. DOI: 10.3390/s20051500. URL: https://www.mdpi.com/1424-8220/20/5/1500/pdf?version=1583947756.

[38] T. Chen et al. "A simple framework for contrastive learning of visual representations". A: *International Conference on Machine Learning*. PMLR, 2020, pàg. 1597 - 1607.

[39] A. Farhadi i P.P. Tabrizi. "Human Activity Recognition Using Neural Networks". A: *Journal of Information and Communication Technology* 19.2 (2020), pàg. 1 - 14. URL: http://e-journal.uum.edu.my/index.php/jict/article/download/jict2020.19.2.1/2011/.

[40] Tianyu Gao, Adam Fisch i Danqi Chen. "Making Pre-trained Language Models Better Few-shot Learners". A: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 2020, pàg. 1567 - 1580.

[41] Ayoosh Kathuria. *YOLO v5 is Here: Custom Object Detection in Google Colab*. 2020. URL: https://towardsdatascience.com/yolo-v5-is-here-custom-object-detection-in-google-colab-774418df4f22.

[42] M. Liu. "Video Classification Technology Based on Deep Learning". A: *2020 International Conference on Information Science, Parallel and Distributed Systems (ISPDS)*. Xi'an, China, 2020, pàg. 154 - 157.

[43] Daniel Nüst et al. "The Rockerverse: Packages and Applications for Containerisation with R". A: *The R Journal* 12.1 (2020), pàg. 437 - 461.

[44] Adrian Rosebrock. *YOLO object detection with OpenCV*. 2020. URL: https://www.pyimagesearch.com/2020/06/01/yolo-object-detection-with-opencv/.

[45] A. Sasithradevi i S.M.M. Roomi. "Video classification and retrieval through spatio-temporal Radon features". A: *Pattern Recognition* 99 (2020), pàg. 107099.

[46] Pulkit Sharma. *YOLO - You Only Look Once – Real Time Object Detection explained*. 2020. URL: https://www.analyticsvidhya.com/blog/2020/12/yolo-you-only-look-once-real-time-object-detection-explained/.

[47] Muhammad Awais et al. "Classical Machine Learning Versus Deep Learning for the Older Adults Free-Living Activity Classification". A: *Sensors* 21 (jul. de 2021), pàg. 4669. DOI: 10.3390/s21144669.

[48] Saddam Bekhet i Abdullah M Alghamdi. "A Comparative Study of Video Classification Techniques: Direct Features Matching, Machine Learning, and Deep Learning". A: *IEEE Access* 9 (2021), pàg. 10370 - 10384. URL: http://jsju.org/index.php/journal/article/download/994/985.

[49] T. Lancaster. "Academic Dishonesty or Academic Integrity? Using Natural Language Processing (NLP) Techniques to Investigate Positive Integrity in Academic Integrity Research". A: *Journal of Academic Ethics* 19 (2021), pàg. 1 - 26. URL: https://link.springer.com/content/pdf/10.1007/s10805-021-09422-4.pdf.

[50] Weijia Liu et al. "A Survey on Recent Advances in Sequence-to-Sequence Learning". A: *Information* 12.1 (2021), pàg. 38. DOI: 10.3390/info12010038. URL: https://www.mdpi.com/2078-2489/12/1/38/pdf?version=1610948749.

[51] Angelina McMillan-Major et al. "Reusable Templates and Guides For Documenting Datasets and Models for Natural Language Processing and Generation: A Case Study of the HuggingFace and GEM Data and Model Cards". A: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing* (2021), pàg. 88 - 100. DOI: 10.18653/v1/2021.gem-1.11. URL: https://aclanthology.org/2021.gem-1.11.pdf.

[52] Nanmiao Wu. "Performance Analysis and Improvement for Scalable and Distributed Applications Based on Asynchronous Many-Task Systems". Tesi doct. Louisiana State University, 2021. URL: https://digitalcommons.lsu.edu/cgi/viewcontent.cgi?article=6873&context=gradschool_dissertations.

[53] Yifan Wu et al. "A Distributed Application Framework for Public Blockchain". A: *Electronics* 10.4 (2021), pàg. 445. DOI: 10.3390/electronics10040445. URL: https://www.mdpi.com/2079-9292/10/4/445.

[54] Y. Zhang i L. Wu. "Deep Learning in Mobile and Wireless Networking: A Survey". A: *Wireless Communications and Mobile Computing* 2021 (2021), pàg. 1140611. URL: https://downloads.hindawi.com/journals/wcmc/2021/1140611.pdf.

[55] Zied Bedhief i Mohamed Jemni. "Empowering Docker for High Performance Computing and Big Data". A: *ACM Digital Library* (2022). DOI: 10.1145/3469831. URL: https://dl.acm.org/doi/10.1145/3469831.

[56] Jun-Hwa Kim et al. "Improved Singapore Maritime Dataset (SMD-Plus) and Its Application to YOLO-V5 for Maritime Object Detection". A: *Journal of Marine Science and Engineering* 10.3 (2022), pàg. 377. URL: https://www.mdpi.com/2077-1312/10/3/377.

[57] Junghwa Lee, Junghwan Kim i Junmo Kim. "An Improved Singapore Maritime Dataset (SMD-Plus) and Its Benchmark Results for Object Detection and Classification". A: *Journal of Marine Science and Engineering* 10.3 (2022), pàg. 377. URL: https://www.mdpi.com/2077-1312/10/3/377.

[58] Linjie Li et al. "Revisiting Video-Language Understanding with Atemporal Probes". A: (2022). URL: http://arxiv.org/pdf/2206.01720.

[59] Mihaela Popescu et al. "Sentiment Analysis on Romanian Tweets: An Approach Based on Machine Learning with Feature Engineering and Deep Learning with Word Embeddings". A: *Algorithms* 15.10 (2022), pàg. 357.

[60] Sohini Roychowdhury. "Semi-supervised and Deep learning Frameworks for Video Classification and Key-frame Identification". A: *2022 International Joint Conference on Neural Networks (IJCNN)*. 2022, pàg. 1 - 8. DOI: 10.1109/IJCNN55064.2022.9891884. URL: https://dx.doi.org/10.1109/IJCNN55064.2022.9891884.

[61] Stuart Russell i Peter Norvig. *Artificial Intelligence: A Modern Approach*. 4a ed. Pearson, 2022.

[62] Richard Szeliski. *Computer Vision: Algorithms and Applications*. 2a ed. Springer, 2022.

[63] J. Vaishnavi i V. Narmatha. "Multi-level Video Captioning based on Label Classification using Machine Learning Techniques". A: *International Journal of Advanced Computer Science and Applications* 13.11 (2022), pàg. 209 - 217. URL: http://thesai.org/Downloads/Volume13No11/Paper_67-Multi_level_Video_Captioning_based_on_Label_Classification.pdf.

[64] *About*. OpenCV. 2023. URL: https://opencv.org/about/ (cons. 22-06-2023).

[65] cruizba. *cruizba/ubuntu-dind*. https://hub.docker.com/r/cruizba/ubuntu-dind/tags. Docker image: cruizba/ubuntu-dind, Last updated 20 days ago. 2023.

[66] PyZMQ Documentation. *PyZMQ: Python bindings for ØMQ*. 2023. URL: https://learning-0mq-with-pyzmq.readthedocs.io/en/latest/pyzmq/pyzmq.html.

[67] Flask. *Flask (A Python Microframework)*. 2023. URL: https://flask.palletsprojects.com/en/2.0.x/.

[68] Flask. *Flask (Extensions)*. 2023. URL: https://flask.palletsprojects.com/en/2.0.x/extensions/.

[69] Flask. *Flask (Flask Extension Development)*. 2023. URL: https://flask.palletsprojects.com/en/2.0.x/extensiondev/.

[70] OpenJS Foundation. *Build cross-platform desktop apps with JavaScript, HTML, and CSS | Electron*. Accessed: 2023-06-30. 2023. URL: https://www.electronjs.org/ (cons. 30-06-2023).

[71] *GitHub - opencv/cvat: Annotate better with CVAT, the industry-leading data engine for machine learning. Used and trusted by teams at any scale, for data of any scale.* GitHub. 2023. URL: https://github.com/openvinotoolkit/cvat (cons. 22-06-2023).

[72] Red Hat. *What's a Linux container?* 2023. URL: https://www.redhat.com/en/topics/containers/whats-a-linux-container.

[73] Glenn Jocher, Ayush Chaurasia i Jing Qiu. *YOLO by Ultralytics*. Vers. 8.0.0. Gen. de 2023. URL: https://github.com/ultralytics/ultralytics.

[74] Stefan Lumpp, Stefan Wagner i Andreas Wiedemann. "Design and Implementation of a Docker Analysis Tool". A: *ACM Digital Library* (2023). DOI: 10.1145/3603111. URL: https://dl.acm.org/doi/10.1145/3603111.

[75] University of Manchester. "Machine-to-Machine Communication for the Industrial Internet of Things". A: (2023). URL: https://pure.manchester.ac.uk/ws/files/51166768/IoT2016.pdf.

[76] Jérôme Petazzoni. *Using Docker-in-Docker for your CI or testing environment? Think twice.* 2023. URL: https://jpetazzo.github.io/2015/09/03/do-not-use-docker-in-docker-for-ci/ (cons. 27-06-2023).

[77] Fernando Rosa. "What is Docker? The spark for the container revolution". A: *InfoWorld* (2023). URL: https://www.infoworld.com/article/3310941/what-is-docker-the-spark-for-the-container-revolution.html.

[78] Serdar Yegulalp. "Why you should use Docker and containers". A: *InfoWorld* (2023). URL: https://www.infoworld.com/article/3310941/why-you-should-use-docker-and-containers.html.

[79] ZeroMQ. *What is ZeroMQ?* 2023. URL: https://zeromq.org/what-is-zeromq/.

[80] *Apache License, Version 2.0*. https://www.apache.org/licenses/LICENSE-2.0. Accessed: 2023-06-28.

[81] *CVAT: An Open Source Computer Vision Annotation Tool*. https://github.com/openvinotoolkit/cvat. Accessed: 2023-06-28.

[82] *Docker: Empower Developers to Build and Share Any Application*. https://www.docker.com. Accessed: 2023-06-28.

[83] *Flask: a micro web framework written in Python*. https://flask.palletsprojects.com. Accessed: 2023-06-28.

[84] Python Software Foundation. *PEP 3333 – Python Web Server Gateway Interface v1.0.1*. https://www.python.org/dev/peps/pep-3333. Accessed: 2023-06-28.

[85] *JAAD Dataset*. http://data.vision.ee.ethz.ch/cvl/jaad/.

[86] *Jinja: A small but fast and easy to use stand-alone template engine written in pure python*. https://jinja.palletsprojects.com. Accessed: 2023-06-28.

[87] *OpenCV: Open Source Computer Vision Library*. https://opencv.org. Accessed: 2023-06-28.

[88] TensorFlow. *Image Recognition*. URL: https://www.tensorflow.org/ (cons. 22-06-2023).

[89] *Werkzeug: The Python WSGI Utility Library*. https://werkzeug.palletsprojects.com. Accessed: 2023-06-28.

[90] *ZeroMQ*. http://zeromq.org. Accessed: 2023-06-28.

APÈNDIX A
Objectius de Dessenvolupament Sostenible

Objectius de Desenvolupament Sostenible	Alt	Mitjà	Baix	No procedeix
ODS 1. **Fi de la pobresa.**				X
ODS 2. **Fam zero.**				X
ODS 3. **Salut i benestar.**				X
ODS 4. **Educació de qualitat.**		X		
ODS 5. **Igualtat de gènere.**				X
ODS 6. **Aigua neta i sanejament.**				X
ODS 7. **Energia assequible i no contaminant.**				X
ODS 8. **Treball decent i creixement econòmic.**		X		
ODS 9. **Indústria, innovació i infraestructures.**	X			
ODS 10. **Reducció de les desigualtats.**				X
ODS 11. **Ciutats i comunitats sostenibles.**				X
ODS 12. **Producció i consum responsables.**				X
ODS 13. **Acció pel clima.**				X
ODS 14. **Vida submarina.**				X
ODS 15. **Vida d'ecosistemes terrestres.**				X
ODS 16. **Pau, justícia i institucions sòlides.**				X
ODS 17. **Aliances per aconseguir objectius.**		X		

A.1 Reflexió sobre la relació del TFG amb els ODS i amb els ODS més relacionats

El meu treball està estretament relacionat amb l'ODS 9: Indústria, innovació i infraestructures. Aquest objectiu busca construir infraestructures resistents, promoure la industrialització inclusiva i sostenible i fomentar la innovació. El desenvolupament d'una aplicació d'escriptori multiplataforma que utilitza tècniques d'anàlisi estadística, visió per ordinador i intel·ligència artificial per a la classificació de vídeos és una clara manifestació d'innovació tecnològica. Aquesta aplicació no només millora l'eficiència en la gestió de continguts de vídeo, sinó que també contribueix a la creació d'infraestructures de dades més robustes i eficients.

A més, també té una relació moderada amb l'ODS 4: Educació de qualitat i l'ODS 8: Treball decent i creixement econòmic. L'aplicació que he desenvolupat pot ser emprada com una eina educativa per a l'aprenentatge de la gestió de continguts de vídeo. A més, en facilitar la cerca ràpida i eficient de continguts de vídeo, l'aplicació pot contribuir a millorar la productivitat dels tècnics d'il·luminació, la qual cosa pot tenir un impacte positiu en el seu rendiment laboral i, per tant, en el creixement econòmic.

Podem dir també, que aquest treball també està relacionat amb l'ODS 17: Aliances per assolir els objectius. El desenvolupament de l'aplicació ha requerit la integració de diverses tecnologies, demostrant així la importància de la col·laboració i la creació d'aliances per assolir objectius comuns.

En resum, tot i que el meu treball fi de grau no està directament relacionat amb tots els ODS, sí que contribueix de manera significativa a alguns d'ells, especialment a l'ODS 9. A més, el projecte demostra la importància de la innovació tecnològica per assolir els ODS i com la tecnologia pot ser utilitzada per millorar l'eficiència i la productivitat en diversos àmbits.