A Comprehensive Framework for Frame Detection Leveraging SIFT and Visual Feature Characterization

Adwaith Rajesh[1]; Akash V V[2]; Jyothish M[3]; Sankeerth O T[4]; Aswathy T S[5]

Students[1,2,3,4]; Assistant Professor[5]

[1;2;3;4;5]Department of Computer Science and Engineering Vimal Jyothi Engineering College,
Kannur, Kerala, India

Publication Date: 2025/05/07

Abstract: This project focuses on developing a system that can identify videos using individual frames or short sequences. This is a complex task, but it has the potential to revolutionize how we interact with video content in many industries, from entertainment to security. The ability to identify videos from just a still frame or short video segment is a complex yet highly demanded task in industries ranging from entertainment to security. The system will use visual feature extraction and a comprehensive database to match frames to videos. The methodology involves using a combination of SIFT, YOLOv5, and ResNet-50 to process and analyze the frames. ChromaDB, a vector database for AI applications, is used to store and search for matches. The system will then use a modified ensemble ranking system that considers factors like frequency, consistency, and tag coverage to calculate a confidence score for each match. This score will be displayed to the user along with the matched videos. The project aims to provide a user-friendly interface that allows users to upload images and view the predicted videos, as well as the calculations performed during the matching process. Future improvements include refining the algorithm for finding unique frames, enhancing the user interface with history tracking, and improving the confidence calculation algorithm.

Keywords: *Predictions, Recommendations, Machine Learn- ING, Collaborative Farming, Streamline Trading.*

How to Cite: Adwaith Rajesh; Akash V V; Jyothish M; Sankeerth O T; Aswathy T S (2025) A Comprehensive Framework for Frame Detection Leveraging SIFT and Visual Feature Characterization. *International Journal of Innovative Science and Research Technology*, 10(4), 2549-2553. https://doi.org/10.38124/ijisrt/25apr1659

I. INTRODUCTION

This project aims to develop a system that can identify videos from single frames or short clips, a feature with valuable applications in industries like entertainment and security. The system will analyze visual features from the input frames and match them against a database of video frames. To achieve this, it uses SIFT, YOLOv5, and ResNet-50 for feature extrac- tion, while ChromaDB, an AI-focused vector database, handles storage and search. A refined ranking method will calculate confidence scores for each match by considering factors like frequency, consistency, and tag coverage.

Users will see these scores and matched videos in an easy-to-use interface, which also shows details of the matching process. Ultimately, this system will make video search, copyright enforcement, and real-time content recognition more accessible and efficient across various applications.

A. General Background

This project develops a system to identify videos from single frames or short clips, with applications in entertainment and security. Using SIFT, YOLOv5, ResNet-50 for feature extraction and ChromaDB for search, the system matches input frames to a video database. A refined ranking method calculates confidence scores, displayed alongside matched videos in a user-friendly interface. This innovation streamlines video search, copyright enforcement, and real-time content recognition.

B. Problem Statement

There is a growing need for a system that can identify videos based on individual frames or short sequences, as current solutions rely heavily on text-based metadata and lack the ability to recognize visual cues in real-time. This system will address this gap by using visual feature extraction and a comprehensive database to accurately match frames to videos, enabling more precise and responsive video recognition.

C. Scope of the System

This system revolutionizes video identification by analyzing visual cues from frames or short sequences, enabling real- time applications like content search, copyright enforcement, and security. It combines SIFT, YOLOv5, and ResNet-50 for feature extraction, ChromaDB for efficient frame embedding storage and retrieval, and an advanced ranking method for accurate confidence scoring. A user-friendly interface ensures ease of use, while its scalable design supports future enhance- ments in accuracy and usability..

D. Objective

This system identifies videos from frames using SIFT, YOLOv5, and ResNet-50, enabling real-time recognition for media search, security, and copyright, with scalable design, confidence scoring, and user-friendly interface.

II. LITERATURE SURVEY

Studies highlight the role of machine learning in enhancing video frame identification. Techniques such as SIFT, YOLOv5, and ResNet-50 are employed for frame matching, feature extraction, and real-time video recognition. These approaches emphasize the importance of robust algorithms and scalable databases to optimize video search and retrieval.

Key references include Bose *et al.* [1], who proposed a method for visual scene recognition in movies using advanced machine learning models for accurate video identification based on individual frames. Deng *et al.* [2] addressed the challenge of detecting multiple salient objects by integrating long-range dependencies in object detection, contributing to improved frame matching accuracy.

Flores *et al.* [3] explored saliency techniques for fine-grained object recognition, focusing on domains with limited training data to enhance video frame recognition capabilities. Kaur *et al.* [4] reviewed object detection advancements using deep learning models, providing insights into improving the accuracy and efficiency of video frame identification.

Krizhevsky *et al.* [5] pioneered the use of deep convolu- tional neural networks (CNNs) for image classification, estab- lishing a foundational technique for extracting meaningful fea- tures in video frame recognition. Lowe *et al.* [6] developed the Scale-Invariant Feature Transform (SIFT) algorithm for robust keypoint detection, enabling distinctive frame identification in various video scenes.

Tsourounis *et al.* [7] integrated CNNs with dense SIFT descriptors for sequence classification, enhancing the system's ability to match video frames with precision. Vidhyalakshmi *et al.* [8] proposed a similarity metric learning technique using deep learning and SIFT for improved person re-identification, applicable to frame matching in video recognition.

Wu *et al.* [9] implemented joint learning of foreground, background, and edge features to improve salient object detec- tion, optimizing frame extraction for video identification tasks. Finally, Zhou *et al.* [10] combined SIFT and CNN features to detect partial-duplicate images, enhancing the system's ability to match similar video frames effectively.

III. REQUIREMENT SPECIFICATIONS

A. Functional Requirements

➢ *User Interface (UI):*

- Users should have a straightforward way to upload images or video frames, either by dragging files or selecting them from their device.
- Once an image or frame is uploaded, the UI should display the identified or matched video, along with a list of potential matches.
- Each match should have a confidence score displayed next to it, helping users understand the accuracy of each result.
- The interface could include a history tracking feature to let users review past searches, providing continuity and ease of access to prior matches

B. Hardware Requirements

- The project requires a powerful GPU for real-time model inference, enabling fast feature extraction and matching with YOLO and ResNet-50.
- The system needs ample storage for frame embeddings and videos, with ChromaDB scaling storage based on data volume.
- Sufficient RAM is essential for smooth performance, especially during video frame extraction and database searches.
- A high-performance CPU is required for I/O processing, database interactions, and UI responsiveness, ensuring an interactive user experience.

C. Non-Functional Requirements

- The system should identify and display results within seconds, ensuring real-time or near-real-time processing for a smooth user experience.
- The system should handle multiple simultaneous queries without significant performance drops, even in high-usage environments.
- The architecture should be scalable to handle larger datasets in ChromaDB, ensuring minimal impact on response time as new videos and frames are added.
- The UI should be intuitive, with features like drag-and-drop upload, clear result displays, and accessible confidence scores for easy navigation and improved usability.
- The system must ensure reliable data storage and retrieval, especially in ChromaDB, to prevent data corruption of frame embeddings.
- The project code should be modular, enabling easy updates and modifications to components like the UI,

detection models, and database.

- The application should be compatible across different operating systems (e.g., Windows, macOS, Linux), making it accessible to a broader audience

IV. PROPOSED SYSTEM AND DESIGN

The system leverages machine learning for video frame identification. Key modules include frame matching, feature extraction, video recognition, and confidence scoring for ac- curate and efficient video retrieval.

A. Architecture

The architecture outlines several key components such as feature extraction, object detection, embedding, and matching. The process is broken down into two key stages.

B. Use Case Diagram

We have a simple use case diagram illustrating the inter- actions between users and the system. In this system, users have the ability to upload images, which are then processed to identify and match frames from the video database. Once the process completes, users can view the result, displaying matched videos and confidence scores. Admins, on the other hand, have additional privileges allowing them to manage the content in the database by adding new videos. This feature ensures that the database stays updated with relevant content, enhancing the accuracy and relevance of the matching results provided to the users.

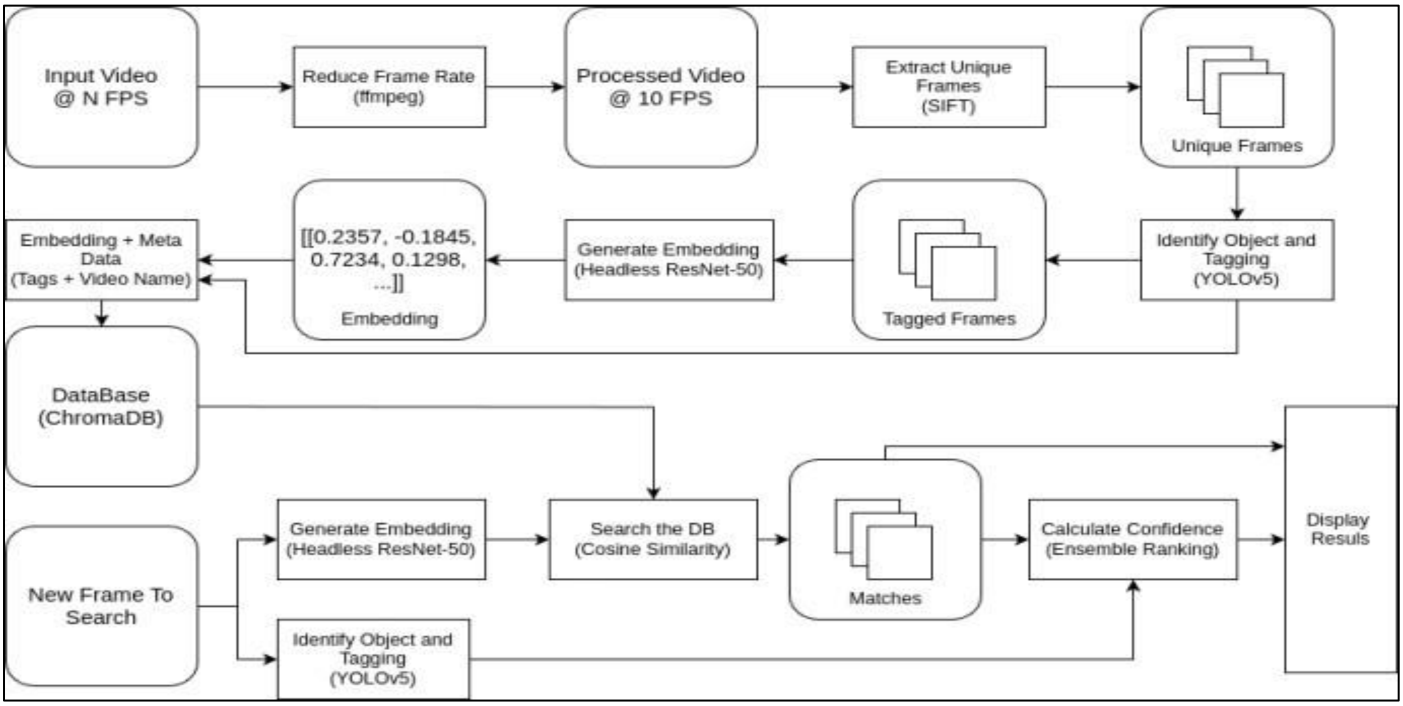
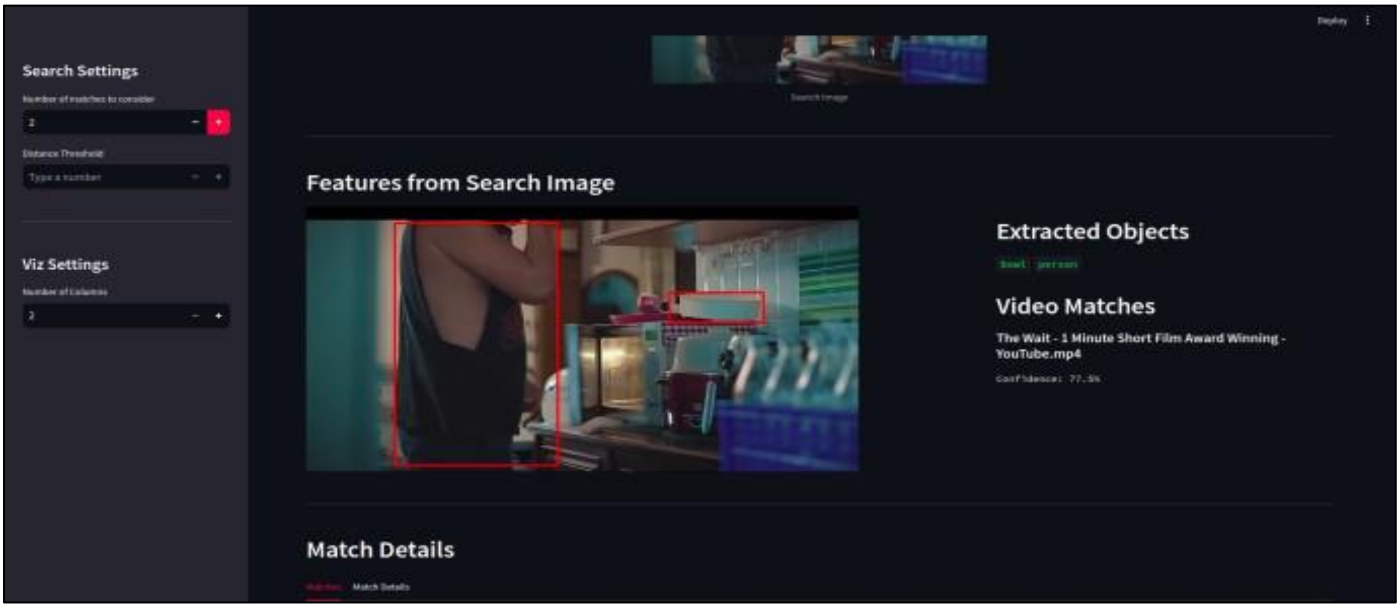

Input Video @ N FPS
Reduce Frame Rate (ffmpeg)
Processed Video @ 10 FPS
Extract Unique Frames (SIFT)
Unique Frames
Identify Object and Tagging (YOLOv5)
Tagged Frames
Generate Embedding (Headless ResNet-50)
[[0.2357, -0.1845, 0.7234, 0.1298, ...]]
Embedding
Embedding + Meta Data (Tags + Video Name)
DataBase (ChromaDB)
New Frame To Search
Generate Embedding (Headless ResNet-50)
Identify Object and Tagging (YOLOv5)
Search the DB (Cosine Similarity)
Matches
Calculate Confidence (Ensemble Ranking)
Display Resuls

Fig 1 Architecture Diagram



Search Settings

Number of matches to consider: 2

Distance Threshold: Type a number

Viz Settings

Number of Columns: 2

Features from Search Image

Extracted Objects

bowl person

Video Matches

The Wait - 1 Minute Short Film Award Winning - YouTube.mp4

Confidence: 77.5%

Match Details

Fig 2 The Results from the Search

V. IMPLEMENTATION

The implementation of the video identification system in- volves several core components designed to provide seamless and efficient video recognition. This section details the key aspects of the implementation.

➢ *Frontend Development*

The frontend is responsible for the user interface and interaction with the system. The main software requirements are as follows:

- Framework: Streamlit is used to build an interactive, web-based user interface for uploading images and viewing matched results.
- Programming Language: Python is used, enabling seamless integration with Streamlit for real-time result display and backend functionality.
- Feature Extraction: SIFT, YOLOv5, and ResNet-50 are employed for robust feature extraction from video frames.
- Database: ChromaDB is utilized to store and search frame embeddings, ensuring fast and scalable retrieval.
- Confidence Scoring: A modified ensemble ranking method calculates confidence scores based on frequency, consistency, and tag coverage.
- User Interface: The UI allows easy upload of images, displays matched videos, and shows confidence scores and calculations for clarity.

➢ *User Interface Screenshots*

The user interface is designed to be simple and easy to navigate. Users can quickly upload images, and the system displays matching video results along with detailed confidence scores. Additionally, the values used in calculating the con- fidence are presented clearly, giving users insights into the decision-making process.

➢ *Backend Development*

The backend is responsible for processing uploaded images, performing frame detection, and managing the database. Key software components include:

- Language: Python is used for core process- ing, model integration, and server-side logic.
- Frameworks and Libraries:
- YOLOv5: For object detection within video frames.
- ResNet-50: Used for generating feature embeddings from frames.
- FFmpeg: Tool for video frame rate reduction and frame extraction.
- SIFT (Scale-Invariant Feature Transform): For extracting unique frames and identifying distinctive features in frames.
- Database: ChromaDB is a vector database designed to store and manage embeddings for fast similarity searches.
- Similarity Calculation: Cosine Similarity is used for comparing embeddings and determining matches in the database.
- Real-Time Processing: The backend handles real-time image uploads and video frame extraction with minimal latency.
- Scalability: The system is designed to scale as the database grows, ensuring efficient searches and processing even with large datasets.
- Confidence Scoring: A modified ensemble ranking system calculates confidence scores based on multiple factors such as frequency, consistency, and tag coverage.
- Data Storage: Efficient data storage and retrieval mechanisms are implemented to handle large video datasets and embeddings without performance degradation.
- Programming Security: Secure APIs and encrypted data transmission ensure the safety of user data and video content.

VI. EVALUATION AND RESULTS

The prototype was tested with users, demonstrating im- proved video identification accuracy and user satisfaction. Frame matching and confidence scoring were found to be precise, supporting efficient and data-driven video retrieval.

VII. CONCLUSION AND FUTURE WORK

In conclusion, the framework efficiently identifies videos from individual frames, offering a seamless user experience with accurate matching and confidence scores. It leverages models like YOLOv5 and ResNet-50 for reliable frame recog- nition. Future improvements include refining unique frame se- lection, enhancing the UI with history tracking and recommen- dations, and optimizing the confidence calculation algorithm for even greater precision and reliability. These enhancements will increase the framework’s versatility and applicability across various video recognition tasks.

REFERENCES

[1]. D. Bose, R. Hebbar, K. Somandepalli, H. Zhang, Y. Cui, K. Cole- McLaughlin, H. Wang, and S. Narayanan, ”Movieclip: Visual scene recognition in movies,” in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2083–2092, 2023.

[2]. B. Deng, A. P. French, and M. P. Pound, ”Addressing multiple salient object detection via dual-space long-range dependencies,” *Computer Vision and Image Understanding*, vol. 235, p. 103776, 2023.

[3]. C. F. Flores, A. Gonzalez-Garcia, J. van de Weijer, and B. Raducanu, ”Saliency for fine-grained object recognition in domains with scarce training data,” *Pattern Recognition*, vol. 94, pp. 62–73, 2019.

[4]. R. Kaur and S. Singh, ”A comprehensive review of object detection with deep learning,” *Digital Signal Processing*, vol. 132, p. 103812, 2023.

[5]. A. Krizhevsky, I. Sutskever, and G. E. Hinton, ”Imagenet classification with deep convolutional neural networks,” *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[6]. D. G. Lowe, ”Distinctive image features from scale-invariant keypoints,” *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.

[7]. D. Tsourounis, D. Kastaniotis, C. Theoharatos, A. Kazantzidis, and G. Economou, ”Sift-cnn: When convolutional neural networks meet dense sift descriptors for image and sequence classification,” *Journal of Imaging*, vol. 8, no. 10, p. 256, 2022.

[8]. M. K. Vidhyalakshmi, E. Poovammal, V. Bhaskar, and J. Sathya- narayanan, ”Novel similarity metric learning using deep learning and root sift for person re-identification,” *Wireless Personal Communications*, vol. 117, no. 3, pp. 1835–1851, 2021.

[9]. Q. Wu, P. Zhu, Z. Chai, and G. Guo, ”Joint learning of foreground, background and edge for salient object detection,” *Computer Vision and Image Understanding*, vol. 240, p. 103915, 2024.

[10]. Z. Zhou, Q. M. J. Wu, S. Wan, W. Sun, and X. Sun, ”Integrating sift and cnn feature matching for partial-duplicate image detection,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 5, pp. 593–604, 2020.