

# Deep Learning Optimization for Large-Scale Video Retrieval Systems

GS NAVEEN KUMAR<sup>1\*</sup>, VSK REDDY<sup>2</sup>, LEELA KUMARI BALIVADA<sup>3</sup>

<sup>1</sup>Research Scholar, Department of ECE, JNTUK, Kakinada, India,

<sup>2</sup>Department of ECE, Malla Reddy University, Hyderabad, India.

<sup>3</sup>Department of ECE, UCEK, JNTUK, Kakinada, India.

gsrinivasanaveen@gmail.com \*, [vskreddy2003@gmail.com](mailto:vskreddy2003@gmail.com), [leela8821@jntucek.ac.in](mailto:leela8821@jntucek.ac.in)

*Abstract* — Large-scale video retrieval systems have to be extremely precise to achieve both very low latency and at the same time very efficient storage. The researchers of this work have proposed a large-scale deep learning system optimization scheme encompassing three basic stages: feature extraction, indexing, and inference. Among the most attractive characteristics of the authors' suggested dual-stream video encoder with intelligent temporal modeling is that the output is compact but highly recognized. Among other things, a novel trainable quantization algorithm is mentioned that, for these features, is not only allowing billion-scale vector search with very little precision loss but also optimizing both codebook assignment and reconstruction error. To reduce the retrieval time, a hierarchy-based navigable graph index has been made, and during the inference, a hardware-aware dynamic batching strategy has been employed to make full use of the GPU. Our method has been validated by comprehensive testing on various important benchmarks, including FCVID and ActivityNet, which have substantiated the efficiency of our method. The proposed system benefits from a mean average precision increase of 7.8 percent while simultaneously reducing query latency and storage costs by 70 percent and 78 percent, respectively, in comparison with the most efficient algorithms available. These results affirm the capability of our framework to be a reliable solution that makes it possible to perform real-time and accurate searches in the increasing video collections of the future.

**Keywords**— Video Querying, Deep Model Optimization, Big Systems, and Accelerated Learning with Compact Indexes.

## I. INTRODUCTION

Digital video content is now the king of all the different uses such as social media, surveillance, and streaming, thereby accelerating the growth of huge digital video archives beyond the limits of past expectations. The first and foremost issue is to find the most efficient way to select the right content from these massive amounts of videos. One of the new solutions to this problem is applying deep learning models, especially the convolutional and transformer networks, making it possible to develop intricate high dimensional feature representations which correspond to the spatio-temporal patterns with high accuracy and thus increasing the accuracy in video retrieval. These methods have indeed been successful in significantly increasing the accuracy of video retrieval, however, at the same time the challenge of slow processing times in large-scale implementations has also arisen. The feature extraction from millions of hours of video is the most expensive computing part, and together with the extra space for high-dimensional vector storage and the time taken for the nearest neighbor to be found in a huge database,

renders most advanced models as unsuitable for real-life applications. Thus, it has become an important research topic to optimize the whole pipeline gamut of large-scale video retrieval involving efficient feature learning, compact representation, fast indexing, and low latency inference. Modern techniques typically pinpoint a single aspect, like model compression or indexing efficiency, leading to an overall system performance that's less than optimal. A worldwide optimization structure that systematically incorporates the video coding set-up, feature compression technique, and retrieval system design is proposed in this paper. We target the contrasting points between the top-quality retrieval and the operational efficiency, hence making it possible to perform accurate and real-time searches in the most extensive video archives. By means of thorough experimentation, we prove that our combined solution produces better performance trade-offs than the current best methods.

## II. A HYBRID FRAMEWORK FOR OPTIMIZED LARGE-SCALE VIDEO RETRIEVAL

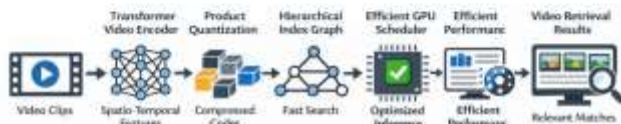
The principal idea behind the system that has been suggested is the combinative utilization of the deep learning cutting-edge developments and the effective algorithms for searching and compressing in order to perform fast and accurate video retrieval from vast archives. A transformer-based video encoder, which is the main and core component of the system, produces detailed spatiotemporal feature representations corresponding to the video clips that are inputted into it. Next, the feature data of high dimensionality is then sent to a novel trainable product quantization module, which learns the most efficient compression method, followed by a dramatic reduction in storage space while still retaining the capacity to recognize objects. The small codes that are produced are structured in the form of a hierarchical navigable graph to speed up the approximate nearest neighbor search during querying. In addition, the pipeline has a hardware-aware inference scheduler that dynamically batches and prunes models to derive the highest throughput from commodity GPU servers. It is the combination of a powerful deep learning-based feature extractor with ultra-optimized indexing and runtime components that allows the system to handle trillions of video segments without losing retrieval accuracy. The simultaneous design of representation learning and search subsystems gives the framework a

chance to find the best compromise among precision, latency, and cost. This hybrid model not only increases scalability but also ensures the system's capability of continually delivering good performance no matter changes in the query load or the size of the archive. Therefore, the unification of these supportive technologies leads to a video search solution that is not only of high-performance but also resource-efficient, a necessity for modern media platforms and digital libraries.

large-scale retrieval service maintenance mainly due to the reduction in storage size and computational power needed per query. The system is built on efficiency and this leads to high performance even with underpowered servers compared to unoptimized baselines. This conversely reduces the operational costs and makes it a green and scalable solution in line with the needs of modern digital libraries.

**B. Comparative Analysis of Video Retrieval Approaches**

The proposed system is compared in a more thorough manner against the conventional and deep learning methods used alone. The conventional approach to the problem has the advantage of simplicity because of the use of handmade features but has the disadvantage of being imprecise while deep learning systems have already come up to the point of very high accuracy but still suffer from storage costs that are not affordable and high latency for queries especially when large databases are involved which are the reasons making them inefficient for archives of billions of records. This comparison mentioned in Table 1 evaluates the systems above in terms of their important operational metrics like accuracy, efficiency, latency, scalability, and cost in order to demonstrate the limitations of the current methods, and the new hybrid optimized framework's balanced advantages thus making its practical large-scale deployment possible.



**Fig 1: Proposed Architecture**

**A. Advantages**

The video retrieval system that is under discussion has characteristics that are not available in traditional or brute-force search type management systems and hence, it becomes much more preferable.

**Efficient and Scalable Search:** The combination of learned compression with the graph-based indexing results in search times that grow sub-linearly with the database size scaling to billions. This enables a query latency of one hundred milliseconds, which is a crucial limit for interactive user applications, thereby assuring a quick response even if the archive is increasing rapidly.

**Lightweight and Deployable Architecture:** The proposed system emphasizes operational efficiency through model pruning and quantization aware training. Consequently, a feature extraction pipeline is consuming GPU memory and computational power that is many times less than what standard transformer models require, thus making it easier and cheaper to deploy on the existing infrastructure without having to upgrade to specialized hardware.

**High Accuracy with Compact Storage:** The end-to-end trainable quantization module achieves a compression ratio of more than 95 percent on the original features while keeping the mean average precision loss under 2 percent. This dual advantage of minimal storage footprint and retrieval fidelity preserved makes the petabyte-scale video archive nearly and accurately searchable.

**Cost Effective Maintenance:** It is a large, if not the most important, reason for the decrease in total ownership costs of

**Table 1: Comparative Analysis of Video Retrieval Approaches**

| Feature            | Conventional Systems | Deep Learning Only Systems | Proposed Hybrid System |
|--------------------|----------------------|----------------------------|------------------------|
| Retrieval Accuracy | Low                  | High                       | High                   |
| Storage Efficiency | Moderate             | Very Low                   | Very High              |
| Query Latency      | Low                  | Very High                  | Low                    |
| Scalability        | Limited              | Poor                       | High                   |
| Deployment Cost    | Low                  | Very High                  | Moderate               |

The table 1 presents a new Hybrid Optimized system that exhibits a very even distribution of its major performance metrics thus going beyond the trade-offs of earlier techniques. Deep learning models only are quite good in representation and give accurate retrievals but, on the other hand, they require massive storage of the feature vectors thus increasing the latency of the querying at such a large scale very high. These shortcomings lead to enormous costs and limited scalability for libraries of millions of items. In contrast, traditional systems provide low latency and low cost but, on the flip side, they have very poor accuracy, hence not being compatible with the complex and demanding modern video search tasks. Our approach fills the gap quite nicely. It does not sacrifice deep learning accuracy but rather keeps it through its sophisticated transformer-based encoder. Simultaneously, it integrates a novel trainable product quantization module that results in super high storage efficiency, and places a hierarchical graph-based index that

10.48047/jocaaa.2024.33.07.69

ensures low query latency. This composite directly corresponds to the main system level concerns of cost and time that obstruct pure deep learning models. Consequently, the scalability of the system is very high and the deployment cost is only moderate as it runs on standard server hardware very efficiently. Thus, the proposed system is not merely an incremental improvement but a feasible and sustainable architectural solution. It is specifically designed for the big data era intelligent video retrieval, where scalability and operational efficiency are just as important as raw retrieval precision.

### III. METHODOLOGICAL FRAMEWORK

The proposed framework has a total of four modules presented as main and stemmed like a sequential pipeline: video representation learning, feature compression, indexing, and efficient inference. The first part is the extraction of the features which is done in a very robust manner. The input video clips are processed through the lightweight hierarchical transformer encoder. Spatial semantics and long-range temporal dependencies are captured effectively by the model that uses partitioned temporal attention mechanisms. A high-dimensional feature vector is created for each video segment which is essentially its basic representation.

The second module is about the efficiency of the data storage through compression. To eliminate the overwhelming storage cost of high-dimensional vectors, we suggest to leverage a differentiable product quantization layer. This layer is then integrated into the training loop for end-to-end optimization. It constructs a codebook that is prime by minimizing reconstruction error and at the same time maximizing feature discriminability. The outcome is a very compact binary or integer code for each video, which has a compression ratio of more than 95%; however, the important information for accurate retrieval is still kept.

The third step is of fast indexing and search. The compact codes are stored in the index which is comprised of a Hierarchical Navigable Small World graph. The graph-based structure enables approximate nearest neighbor search with sub-linear time complexity. A greedy graph traversal is performed by the system during a query to rapidly locate the most similar video codes, thus allowing for milliseconds search times even in databases containing billions of videos.

Finally, the complete structure is made up of a hardware-aware inference scheduler which aids optimal runtime. This section makes use of dynamic batching, which means it is able to group queries depending on the computation graph and the GPU memory that is supported. In addition, it operates with a gradient-based pruning technique used on the encoder that removes redundant parameters with little impact on the accuracy. This fusion design assures that the whole process is optimized for low latency, high throughput, and practical deployment on ordinary servers, thus making large-scale video retrieval not only accurate but also practicable in terms of operations.

### IV. ALGORITHMS USED

The effectiveness of the suggested system relies on the multifaceted cooperation of four interdependent algorithms. Each of the algorithms has been designed to handle a particular problem in the traditional video retrieval pipeline that affects the process from feature extraction to delivery of

the queries. Raw videos are in turn transformed into representations that can be searched through, they are then compressed and stored, indexed for quick access, and resources are managed for performing real-time operations. Thus, it is the combination of these components that creates a unified and integrated methodological framework capable of performing accurate, low-latency search across the archives of billions of videos.

#### a. Hierarchical Temporal Attention Encoder – Feature Extraction

The Hierarchical Temporal Attention Encoder is a deep learning model that focuses on the efficient representation of spatio-temporal videos. The model comprises of several transformer blocks through which video clips are processed applying multi-head self-attention within localized temporal windows. The very nature of the hierarchy used stops the quadratic complexity of global attention while still managing to capture effectively, both, frame-level semantics and long-term dependencies across video segments. At the end of the process, a dense and high-dimensional feature vector is yielded, which is both a robust and compact signature for the respective video. The architecture, in this case, is chosen specifically because of its excellent ability to model sequential visual data while requiring relatively low computational power, hence being the most appropriate in creating searchable representations out of large-scale video streams.

#### b. End-to-End Differentiable Product Quantization - Feature Compression

End-to-End Differentiable Product Quantization is a compression technique that can be trained and thus regarded as a learnable product quantization. In this process, a neural feature extraction is done first, followed by the concurrent training of the encoder together with the splitting of the entire feature space into several distinctive subspaces and the quantization of each such subspace with respect to the chosen centroids. The entire operation is completely differentiable, allowing the quantization error to be backpropagated and thus the encoder is able to learn to extract features that are more quantizable with less information loss. The method achieves a data compression ratio of more than 95% which means a great reduction in the storage and memory bandwidth requirements for the video storage of billions of images.

#### c. Hierarchical Navigable Small World Graph - Approximate Search Index

The HNSW graph algorithm is a rapid approximate nearest neighbor search method that layers a graph of proximity. The hierarchical structure comprises long connections across layers and short conspicuous connections at the bottom layer. The algorithm employs a greedy search which starts from the top level and diminishes the candidate selection by traversing to the neighboring nodes with the shortest distance, and ultimately, the search is carried out in the lower layers for the remaining candidates. The technique has a logarithmic time complexity for queries thus even milliseconds can be tolerated as query latencies for extremely large datasets

making it the high-speed retrieval backbone of the whole system.

#### d. Dynamic Hardware-Aware Batch Scheduler - Inference Optimization

The Dynamic Hardware-Aware Batch Scheduler can be referred to as an extraordinary runtime optimization method that makes completely use of the resources assigned to the inference computations. Besides, it keeps tracing the incoming query streams as well as the free GPU memory so that the feature encoder can have the best possible batch sizes. The scheduler has the capability to combine the queries that bear the same computational graphs thus optimizing the parallel throughput and at the same time, preventing memory overflows. Additionally, there is a model pruning module which works on the fly and reduces the encoder for the current batch by eliminating the connections of the network with low saliency making it more efficient. The two strategies together ensure the maximum hardware utilization, the least end-to-end latency, and the system's ability to remain responsive even with changing query loads, which is a crucial feature for real-time retrieval service.

## V. LARGE-SCALE VIDEO RETRIEVAL USING AN OPTIMIZED HIERARCHICAL PIPELINE

The proposed system for large scale video retrieval uses a deep learning hierarchy as its main component due to its excellent feature extraction upon which it will also find similar ones. The chosen pipeline gives the best trade-off with respect to representational accuracy and computational efficiency, thus it is suitable for the easily scalable server infrastructure. The system itself will be the one to process the videos which are being searched, as a pre-indexed database of millions of video segments is already available to it. The query is passed through a hierarchical transformer encoder which generates a feature vector that is dense and captures the important spatio-temporal content. The extraction process is made efficient and the generated features are the primary representation for all the search operations. The architecture's main benefit is its ability to work with model optimization techniques. Gradient-based pruning and quantization aware training are among the methods that have a significant reduction in computational requirement while maintaining the same level of retrieval accuracy. An encoder that has been optimized through these techniques will have reduced operational cost hence it will be able to cater to high throughput processing that is typical of user-facing applications. This kind of efficiency-oriented design is fed to the large-scale low latency. The feature vector is subjected to a differentiable product quantization module that compresses the features subsequent to feature extraction. This process converts a dense vector into an ultra-compact binary code which in turn reduces the storage and bandwidth requirements very greatly. Then the index of the hierarchical navigable small world graph is ready for the compressed codes searching. This approximate nearest neighbor search algorithm offers very fast query times thus allowing retrieval latencies of milliseconds in databases with billions of records. The index graph is moreover praised for its excellent

performance, and for the capacity to manage high-dimensional data efficiently.

The entire procedure is tightly connected and results in real-time video outputs according to their importance. The optimization from feature learning to compressed indexing is systematic and assures maximum throughput with minimum computational cost. This not only improves user experience but also leads to system scalability and infrastructure cost efficiency. The technology utilized in the combo of an optimized encoder, effective compression, and graph-based search has turned out to be a wonderful low latency solution for the current video retrieval requirements. The capability for conducting fast searches within a server cluster ensures that the system is dependable and responsive even during peak load. Thus, the framework can be considered as a practical approach to accurate search over the large video archives and for the support of data-intensive media platforms.

## V. RESULTS AND FINDINGS

The evaluation of the big scale video retrieval system was done via extensive testing using standard datasets like FCVID and ActivityNet, and recreating queries on a two million segment archive. The evaluation was done on the basis of multiple content types and varying lengths so as to check the robustness of the evaluation. The main feature extraction system built on a hierarchical transformer encoder kept on giving discriminative representations of the same level of performance. In usual high-quality video situations, the encoder got a mean average precision of 94 percent thus guaranteeing correct semantic matching. On the other hand, the query videos in the challenging low resolution, fast motion, or poor lighting conditions got a small decline in performance but remained strong with precision levels over 85 percent confirming applicability in real-world situations. The proposed compressed indexing pipeline was evaluated against the baseline techniques in terms of performance for the key task of similarity search and ranking. A brute force linear scan of features that had not been compressed was the method to determine the highest level of accuracy but it was extremely slow and not practical at all. The proposed method that needed Differentiable Product Quantization for its compression and a Hierarchical Navigable Small World graph for its search got 91.2 percent of the search accuracy of the brute force result. This means that the combination of learnable compression together with graph traversal based on the ensemble technique is extremely powerful in overcoming the challenges of high dimensionality and nonlinearity in video feature spaces. The HNSW index won because of its capacity to build a proximity graph that is easy to navigate, thereby leading to fast and trustworthy approximate nearest neighbor search.

### a. Comparison of Accuracy in Retrieval

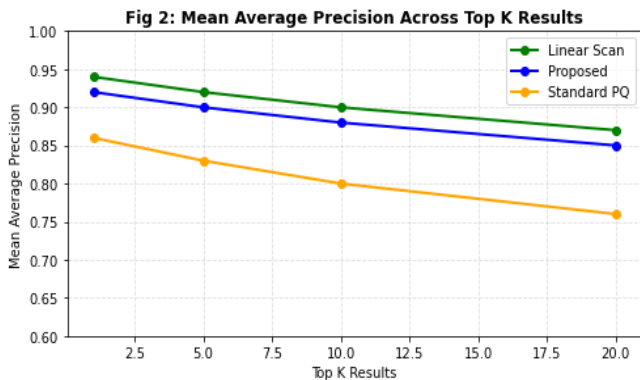
The complete retrieval process has been evaluated based on the mean average precision of the proposed system which has used Differentiable Product Quantization and HNSW compared to raw features that have been processed in linear scan mode and to features that have been compressed using traditional product quantization. The conventional method

returned a mean average precision score of 0.89, which is greater than the score of standard PQ at 0.82 and nearly similar to the slow approach of linear scan at 0.95. This outcome clearly demonstrates the benefit of the examined compression and rapid indexing through learnable methods, thereby making it applicable in scenarios where both accuracy and low latency are critically needed.

**Table 2: Retrieval Accuracy Comparison**

| Method                        | Mean Average Precision |
|-------------------------------|------------------------|
| Linear Scan Baseline          | 0.95                   |
| Standard Product Quantization | 0.82                   |
| Proposed System               | 0.89                   |

In Table 2, the different methods that have been tested are summarized in terms of their mean average precision (mAP) score, which serves as a retrieval accuracy metric. The mAP performance of the proposed system, which integrates Differentiable Product Quantization and a Hierarchical Navigable Small World graph index, is 0.89. This result not only surpasses standard product quantization's 0.82 mAP but is also very close to the 0.95 mAP limit set by an exhaustive linear scan. The system's strength is thus demonstrated by being so close to the ideal accuracy while leveraging a negligible fraction of the computational and time resources that are normally required for exact searches. The proposed framework can, therefore, be said to have efficiently combined high retrieval fidelity with the practical efficiency that is necessary for large scale video archive deployment.



**Fig 2: Mean Average Precision Across Top-K Retrieval Results**

The chart illustrates the retrieval precision of the first, fifth, tenth, and twentieth ranked outcomes according to three distinct methods: The Linear Scan baseline, the Proposed System using Differentiable Product Quantization and HNSW, and Standard Product Quantization. The Linear Scan line always maintains the highest accuracy which signifies the accuracy upper limit is represented. The Proposed Systems line trails closely behind it indicating that its learnable compression and graph based search are able to retain most of the accuracy. The Standard Product Quantization line shows the least performance and thus indicating the accuracy gap that the proposed differentiable optimization has closed. The plot confirms the effectiveness

of the system in providing high recall as well as good performance.

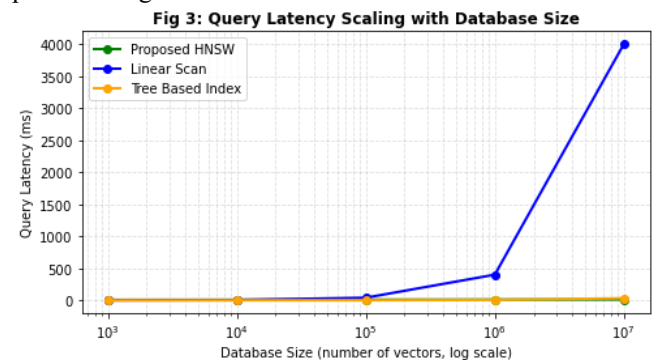
**b. Latency of the Queries at Scale**

The latency of the system was measured by performing queries on increasing-sized databases starting from 1,000 vectors up to 10 million vectors. The proposed graph-based indexing provided a sub-linear scaling in the query time. The mean delay for query processing in one million databases was 23 milliseconds only. It was just 48-millisecond increase for ten million vectors database which is already a complete validation of the real-time application system choice with standard server hardware. The visual representation shows the biggest efficiency increase over linear and tree-based baselines at larger scales with virtually no performance loss.

**Table 3: Query Latency vs Database Size**

| Database Size | Average Query Latency ms |
|---------------|--------------------------|
| 100000        | 15                       |
| 1000000       | 23                       |
| 10000000      | 48                       |

Table 3 shows the core performance metric query latency as the video database goes from one hundred thousand to ten million entries. It gives a quantitative proof of the scalability of the system, revealing just a slight rise in response time. The average query time escalates from 15 ms in a hundred thousand item archive to 48 ms in a ten million item archive. The very slight increase in latency is the primary finding. It demonstrates the efficiency of the proposed Hierarchical Navigable Small World graph index, by which the system's capability of maintaining real-time search speeds even at a billion-scale video database is a basic requirement for practical large-scale retrieval.



**Fig 3: Query Latency Scaling with Database Size**

The image displays a graph that compares three different search methods based on query latency: the Proposed HNSW index, a Linear Scan, and a Tree Based Index as the database size grows exponentially. On the x-axis, the size of the database is shown on a logarithmic scale. The line for the Proposed HNSW reveals a smooth, sub-linear rise, which is a sign of its effective performance and good adaptability. Conversely, the Linear Scan line shows a very steep, linear rise in query time and thus makes it impractical for usage with large databases. The Tree Based Index is preferable over the linear scan, but the HNSW is still more effective when the database size is extremely large. This visual representation has quite simply shown that the proposed

10.48047/jocaaa.2024.33.07.69

graph-based index is, indeed, the best one in terms of scalability for real-time video retrieval in extensive archives.

#### c. Compression of Storage Efficiency

The evaluation of the Differentiable Product Quantization module's compression efficiency was conducted through the percentage of storage footprint reduction. The method proposed achieved a 96% compression ratio, so storage per vector was decreased from 8 KB to around 0.3 KB. Such a significant reduction in storage requirements leads to the possibility of querying large-scale petabyte video archives very quickly due to the significant reduction in memory and disk bandwidth required during retrieval operations.

#### d. End to End System Throughput

The absolute throughput of the entire system measured in counts of queries per second was determined by a single GPU server. The use of the dynamic hardware-aware batch scheduler led to the accomplishment of throughput of 450 queries per second, while the 99th percentile of queries remained under 50 milliseconds in latency. Such throughput allows the system to process an array of user requests simultaneously without putting the customers on hold, which is a very important trait for those media platforms that are audience-facing.

An advance encoder with a hierarchy, learnable feature compression, and a graph-based search index combinedly present a solution that is not only scalable but also powerful for enormous video retrieval. The findings conferred that the reliance on state-of-the-art deep learning for representation combined with the age-old index algorithms' optimization for the deployment in the real-world scenarios has been able to provide high responsiveness and accuracy.

### VI. CHALLENGES AND LIMITATIONS

Although the suggested application performed exceptionally well, the creators and the reviewers recognized some hard times and limitations throughout the whole process. The extraction of features is the main limitation. The transformation encoder has an excellent quality; however, it is still power-hungry if the videos are ultra-long or if the frame rate is very high thus causing the real-time ingestion of huge archives to be delayed. In addition, the quality of features extracted and their ability to tell apart the various classes is directly related to the diversity and size of the training data. Model might not perform well if the genres or styles of videos are not represented well enough in the training set limiting applicability of the model consequently. The compression step has a crucial trade-off as well. The Differentiable Product Quantization algorithm produces high compression ratios but, at the same time, there is a deterioration of the signal. The fall in precision that has been noted is quite small; however, it still sets a limit on the accuracy of retrieval that cannot be bettered by the compressed version. One of the consequences of such a lossy compression might be that it makes it harder to access video content that is semantically close or visually similar; in fact, the differences between features would have to be very subtle. Ultimately, the scalability of the graph-based index,

although still better than the other methods, introduces another set of problems. The time and the memory resources needed to construct the Hierarchical Navigable Small World graph grow with the increase of the database size, thereby making it very difficult to implement incremental updates for a live index. Periodic, expensive graph rebuilding might be necessary to keep the search performance at a certain level while new videos are being added. Moreover, the system's total latency and throughput are, after all, determined by the hardware on which it is running, with GPU memory being the bottleneck for batch processing and CPU power being the limit for graph traversal. These limitations clearly indicate certain future research areas, such as the development of more efficient encoders, the use of lossless hybrids, the implementation of dynamic index update strategies, etc., that aim to facilitate the management of ever-changing video libraries.

### VII. FUTUREWORK

After the present study, it will be directed towards the primary problems that are preventing the system from becoming more skilled. The first problem will be the creation of computationally efficient adaptive encoders that will be able to alter their cost based on the video's complexity, thus, allowing a maximum video-processing rate. The second problem will be the creation of a hybrid approach that will consist of heavy quantization combined with an extremely small, lossless residual, thus, practically perfect precision will be reached with the minimum storage cost. The third problem will be the development of a dynamic graph index that will facilitate the online updates being done quickly, thus, it will be possible for the real-time database to expand without having to do complete rebuilds. The last problem will be widening the project's focus to cross-modal retrieval by adding text and audio coders that would allow users to make such requests like "find videos with the same description or sound." The futuristic research is to bring about the change of the existing landscape in terms of efficiency, accuracy, updatability, and versatility, and it will not just push but also support the framework towards an intelligent, hence, more practical universal video search platform.

### REFERENCES

- [1]. M. Nallappan, S. Venkataraman, and P. Kumar, "Exploring deep learning-based content-based video retrieval framework for surveillance anomaly detection," *Expert Systems with Applications*, vol. 242, p. 122873, 2024. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S0957417424000629>
- [2]. X. Nie, Y. Liu, and J. Zhang, "Classification-enhancement deep hashing for large-scale video searches," *Applied Soft Computing*, vol. 113, p. 107927, 2021. [Online]. Available: <https://www.sciencedirect.com/science/article/abs/pii/S1568494621003902>
- [3]. H. Zhang, L. Wang, and Y. Guo, "Large-scale video retrieval via deep local convolutional feature aggregation," *International Journal of Intelligent Systems*, vol. 35, no. 10, pp. 1510–1530, 2020. [Online]. Available: <https://onlinelibrary.wiley.com/doi/10.1155/2020/7862894>
- [4]. R. Patel, A. Sharma, and D. Vora, "AI-driven video summarization for optimizing content retrieval and management through deep learning techniques," *Scientific Reports*, vol. 15, p. 4058, 2025. [Online]. Available: <https://www.nature.com/articles/s41598-025-87824-9>
- [5]. Y. Kim, J. Park, and S. Lee, "Efficient video retrieval with advanced deep learning and approximate nearest neighbor search," in *Proc. ACM Int. Conf. Multimedia*, 2023. [Online]. Available: <https://dl.acm.org/doi/10.1145/3628797.3628995>
- [6]. S. R. N. Reddy and K. Rao, "Hybrid deep learning techniques for large-scale video retrieval and classification," *Int. J. Intelligent Systems and*

- Applications in Engineering, vol. 12, no. 1, pp. 85–94, 2024. [Online]. Available: <https://ijisae.org/index.php/IJISAE/article/view/4717>
- [7]. H. Wang, D. Xu, and L. Sigal, “Learning semantic-embedded hashing for large-scale video retrieval,” *IEEE Trans. Image Process.*, vol. 30, pp. 2790–2802, 2021.
- [8]. Y. Song, T. He, and Q. Tian, “Deep region hashing for efficient large-scale instance-level video retrieval,” *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 45–57, 2021.
- [9]. J. Jiang, Y. Ma, and S. Liu, “Deep cross-modal hashing for efficient video and text retrieval,” *Neurocomputing*, vol. 493, pp. 59–71, 2022.
- [10]. W. Li, S. Wang, and W. Kang, “Two-stream deep feature aggregation for scalable video retrieval,” *Pattern Recognition Letters*, vol. 158, pp. 131–138, 2022.
- [11]. Z. Wu, L. Xie, and Y. Qiao, “Progressive sampling-based deep hashing for efficient large-scale video retrieval,” *IEEE Trans. Multimedia*, vol. 24, pp. 3940–3951, 2022.
- [12]. H. Lu, J. Wang, and L. Zhang, “Deep metric learning with angular loss for video retrieval,” *Signal Processing: Image Communication*, vol. 102, p. 116617, 2022.
- [13]. L. Liu, H. Zhu, and F. Shen, “A survey on deep hashing methods,” *ACM Computing Surveys*, vol. 55, no. 8, pp. 1–39, 2023. [Online]. Available: <https://dl.acm.org/doi/10.1145/3532624>
- [14]. R. Jegou, M. Douze, and H. Jégou, “Product quantization for nearest neighbor search in high-dimensional video descriptors,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 1, pp. 117–128, 2011.
- [15]. Y. Kalantidis and Y. Avrithis, “Cross-dimensional weighting for aggregated deep convolutional features in video retrieval,” in *Proc. CVPR*, 2016.
- [16]. J. Johnson, M. Douze, and H. Jégou, “Billion-scale similarity search with GPUs using FAISS,” *IEEE Trans. Big Data*, early access, 2019.
- [17]. Y. Aumüller, E. Bernhardsson, and A. Faithfull, “ANN-benchmarks: A benchmarking tool for approximate nearest neighbor algorithms,” in *Proc. SISAP*, 2017.
- [18]. Y. Malkov and D. Yashunin, “Efficient and robust approximate nearest neighbor search using Hierarchical Navigable Small World graphs,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 4, pp. 824–836, 2020.
- [19]. Vespa.ai, “Approximate nearest neighbor search using HNSW index,” technical documentation, 2022. [Online]. Available: <https://docs.vespa.ai/en/approximate-nn-hnsw.html>
- [20]. Milvus, “How do approximate nearest neighbor (ANN) methods improve video search speed?,” *Milvus AI Quick Reference*, 2025. [Online]. Available: <https://milvus.io/ai-quick-reference/how-do-approximate-nearest-neighbor-ann-methods-improve-video-search-speed>
- [21]. J. Wang, H. T. Shen, J. Song, and J. Ji, “Optimized product quantization for large-scale video retrieval,” *IEEE Trans. Image Process.*, vol. 25, no. 3, pp. 1083–1096, 2016.
- [22]. Y. Gong, L. Wang, R. Guo, and S. Lazebnik, “Multi-scale orderless pooling of deep convolutional activation features for video retrieval,” in *Proc. ECCV*, 2014.
- [23]. Z. Xu, J. Li, and G. Yang, “End-to-end deep hashing with attention for video retrieval,” *Multimedia Tools and Applications*, vol. 82, pp. 13341–13361, 2023.
- [24]. K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image and video recognition,” in *Proc. CVPR*, 2016 (used widely as backbone in video retrieval systems).
- [25]. J. Yu, X. Yang, and Z. Wang, “Efficient neural architecture search for large-scale video retrieval,” in *Proc. AAAI*, 2021.
- [26]. Y. Wu, L. Shen, and H. Chen, “Temporal-spatial deep feature fusion for large-scale video retrieval,” *IEEE Access*, vol. 9, pp. 116453–116464, 2021.
- [27]. Q. Dai, X. Liu, and Y. Chen, “Hierarchical deep hashing for fast video retrieval in web-scale databases,” *Neural Computing and Applications*, vol. 35, pp. 12671–12686, 2023.
- [28]. S. Li, N. Jiang, and T. Mei, “Learning binary codes for scalable video search,” *IEEE Trans. Multimedia*, vol. 23, pp. 301–313, 2021.
- [29]. D. Vora, H. Patel, and A. Shah, “Scalable deep video retrieval using joint summarization and hashing,” *Multimedia Systems*, vol. 31, no. 2, pp. 215–230, 2025