INDRAPRASTHA INSTITUTE *of*
INFORMATION TECHNOLOGY **DELHI**

Optimal Transport Guided Contrastive Video Summarization

MTech Thesis Report

submitted by

ABU OSAMA SIDDIQUI
MT22006

in partial fulfillment of the requirements
for the award of the degree of

MASTER OF TECHNOLOGY
in
COMPUTER SCIENCE AND ENGINEERING
(Specialization in Artificial Intelligence)

INDRAPRASTHA INSTITUTE OF INFORMATION TECHNOLOGY DELHI
NEW DELHI - 110020

21st May 2025

THESIS CERTIFICATE

This is to certify that the thesis titled ***Optimal Transport Guided Contrastive Video Summarization,*** submitted by **Abu Osama Siddiqui**, to the Indraprastha Institute of Information Technology, Delhi for the award of the degree of Master of Technology is a bonafide record of the research work done by him under my supervision. This thesis has reached the standards of fulfilling the requirements of the regulations relating to the degree.

The contents of this thesis, whether in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. A V Subramanyam
Thesis Supervisor
Associate Professor and Institute Chair Professor
Department of CSE and HCD
Indraprastha Institute of Information Technology, Delhi

Place : New Delhi, India
Date : 21st May 2025

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to my parents for their unwavering support and encouragement throughout this journey. Their moral support, especially during the most challenging moments, has been a constant source of strength and motivation.

I am also thankful to my elder brother for his persistent encouragement, particularly during the final stages of this thesis. His assistance in creating the architecture diagrams and his belief in my abilities helped me stay focused and move forward with confidence.

I extend my appreciation to my labmates, with whom I frequently engaged in insightful discussions. These informal brainstorming sessions, often following my meetings with my advisor, played a valuable role in shaping and refining my ideas.

Most importantly, I would like to sincerely thank my supervisor, without whom this work would not have been possible. His guidance, patience, and constructive feedback throughout the course of this research were instrumental. I am especially grateful for the time and effort he invested in brainstorming sessions and for his constant support, even when results took time to materialize.

ABU OSAMA SIDDIQUI
(MT22006)

Abstract

Understanding a video from concise summaries is of great importance for various applications such as browsing, retrieval and assistive technologies. In this work, we present unsupervised summarization of videos. Video summarization is extremely challenging as it is difficult to find concise and semantic frame representations.

In order to address this problem, our contributions are twofold. First, we study different convolutional and transformer based architectures which can obtain efficient spatio-temporal representations. Second, we propose an optimal transport method to obtain representative clusters of a video. Experimental results on benchmark datasets such as TVSum and SumMe demonstrate that our approach achieves competitive performance.

Table of Contents

List of Figures

List of Tables

Introduction

1.1 Problem Statement

The volume of video data being produced daily is increasing at an exponential rate. With this rapid growth, there is an increasing demand for tools that assist users in efficiently navigating, selecting, and consuming such large-scale video content. **Video summarization** addresses this need by generating concise summary videos that capture the most significant and relevant parts of the original video. Video Summarization is the task of automatically generating a shorter, concise version of a video that preserves the most important and informative content while significantly reducing the duration or number of frames.

This technique not only provides significant evidence for video retrieval and recommendation but also helps individuals to preview videos quickly through their keyframes, which has been widely used in many application scenarios, e.g., video editing, content filtering, and semantic analysis. This capability has wide-ranging applications in domains such as video database management, consumer video analytics, and surveillance.

The output of a video summarization system can take multiple forms depending on the intended application and desired level of temporal coherence. Broadly, these can be categorized into two types:

- **Keyframes / Storyboard Summaries (Static Summaries):**
  These are collections of representative still images selected from different parts of the video. The aim is to capture the most informative or visually distinct frames that reflect the key content and scenes of the original video. Such summaries are typically presented as a storyboard or thumbnail set, allowing users to quickly browse through the essential visual highlights without watching the full video. While computationally efficient and useful for quick browsing or indexing, keyframe summaries do not preserve the temporal or narrative structure of the video.

- **Video Skims (Dynamic Summaries):**
  These are short, temporally ordered video segments extracted from the original video that collectively preserve the narrative flow and semantic continuity. Unlike static summaries, video skims maintain motion, audio, and temporal dependencies, making them more suitable for applications where storytelling or context is crucial—such as surveillance review, sports analysis, or video recommendation. Although more informative, generating dynamic summaries is generally more challenging as it requires modeling the temporal evolution and semantic structure of the video content.

The generated summary is also influenced by user perception, as different users may find different segments of the video important—an idea illustrated in Figure 1.

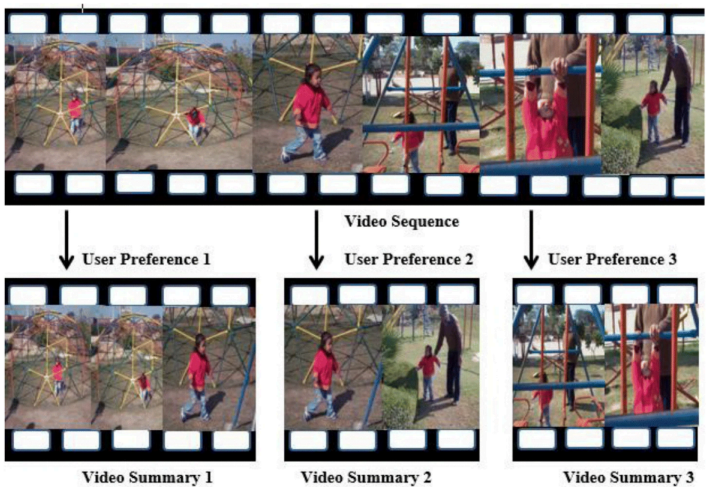

Figure 1: Illustration of user subjectivity in video summarization

1.2 Motivation

In recent years, there has been an unprecedented **explosion in video content** across various domains. The widespread availability of low-cost cameras, coupled with the popularity of **social media platforms**, **video surveillance systems**, and **online video streaming services**, has led to the generation of massive volumes of video data on a daily basis. This growth presents significant challenges in terms of both **efficient consumption** and **management** of such content.

At the same time, **user attention is limited**—viewers are often unwilling or unable to watch long videos in their entirety, especially when only a small portion contains relevant or interesting content. Similarly, the **storage and bandwidth costs** associated with archiving and transmitting full-length videos are increasingly burdensome, particularly for organizations dealing with surveillance footage or large-scale media archives.

These challenges have prompted the need for **automated video summarization techniques**, which aim to generate concise, informative summaries that retain the most meaningful parts of the original video. The potential **applications are vast**, ranging from **security and surveillance analysis**, **sports highlight generation**, and **content-based video recommendation**, to **media asset management**, **lifelogging**, and **personal video organization**.

However, most traditional approaches rely on **manual annotations** or **supervised learning**, which are not only time-consuming and expensive but also difficult to scale across diverse video domains. This has led to a growing interest in **unsupervised video summarization methods**, which can learn to generate summaries **without human-provided labels**, offering a more scalable and adaptable solution to the video content overload problem.

1.3 Objectives

In this work, we address the problem of *video summarization* by formulating it as a ***sparse subset selection*** task [1]. The goal is to select a small, informative subset of frames—commonly referred to as *keyframes*—from the original video such that the selected frames *capture the essential content, preserve semantic coherence*, and *reflect the diversity* of the full-length video. These keyframes should together convey the overarching *gist or theme* of the video while significantly reducing its temporal length.

Unlike supervised approaches, which rely on annotated summaries for training, we focus on the *unsupervised video summarization* setting. Here, the learning process is performed *without access to ground-truth summaries*, making the problem inherently more challenging. The optimization objective in this case is guided by *heuristics, self-supervised losses*, or *unsupervised learning signals* that aim to discover semantically rich and representative content *without explicit human supervision.*

1.4 Challenges in Unsupervised Video Summarization

Unsupervised video summarization poses several unique challenges due to the absence of labeled data and the inherent complexity of video content. Some of the key challenges include:

- **Lack of Ground-Truth Labels for Supervision:**
  Unlike supervised approaches, unsupervised methods do not have access to human-annotated summaries, which serve as a learning signal for selecting important frames. This absence of supervision necessitates the design of alternative objective functions—such as reconstruction, contrastive, or diversity-based losses—that can guide the model toward learning meaningful representations without explicit feedback.

- **Complex and Non-Uniform Temporal Dependencies Between Frames:**
  Videos exhibit rich temporal structures, where important content may span various time intervals and may not follow consistent or predictable patterns. Modeling such *variable-range temporal dependencies* is challenging, especially when trying to preserve the narrative flow and semantic continuity in the summary without relying on predefined annotations.

- **Visual Similarity Does Not Always Imply Semantic Redundancy:**
  Frames that appear visually similar may carry very different semantic meanings depending on their temporal context. For example, visually similar scenes occurring at different points in a narrative may be critical to understanding the story and should not be treated as redundant. This makes naive similarity-based selection strategies insufficient and calls for models that can capture higher-level semantic cues.

- **Balancing Representativeness and Summary Length Constraints:**
  A good video summary must strike a balance between *informativeness* (i.e., including

the most important and diverse content) and *sparsity* (i.e., keeping the summary short and concise). Achieving this balance in an unsupervised manner is difficult, as models must decide which frames best represent the entire video content while adhering to strict length constraints, typically without explicit criteria for what constitutes a "good" summary.

1.5 Contributions

In this work, we present a comprehensive exploration of unsupervised video summarization by investigating a variety of architectural designs and learning strategies for effective spatio-temporal feature modeling. Our primary contributions are summarized as follows:

- **Architectural Exploration for Spatio-Temporal Modeling:**
  We systematically investigate both *convolutional* and *transformer-based* architectures to model temporal dependencies and frame interactions. This includes baseline transformer encoders as well as variants with hierarchical organization and contrastive training objectives.

- **Hierarchical Transformer for Long-Range Temporal Modeling:**
  We propose a novel Hierarchical Transformer architecture, which first models *local dependencies within video segments* using a shared transformer encoder and subsequently captures *global context across segments* using a lightweight aggregation transformer. This two-stage design enables efficient and scalable modeling of long-range temporal structures.

- **Frobenius Loss for Structural Preservation and Neighbor Loss for Local Consistency:**
  To enforce structural alignment between frame representations, we introduce a ***Frobenius Loss***, which penalizes the deviation in pairwise similarity structures between the original and reconstructed features. Additionally, we propose a ***Neighbor Loss*** that ensures local consistency by measuring the distance between keyframe features and their semantically similar non-keyframe neighbors, thereby encouraging more discriminative scoring.

- **Optimal Transport-Based Frame Importance Estimation:**
  We propose a novel unsupervised scoring mechanism based on ***Optimal Transport (OT)***, wherein frame features are aligned with learnable cluster centers through an OT plan computed over pairwise distances. This method captures the *global relational structure* of video content and yields robust importance scores without supervision.

- **Unified Contrastive-Transport Framework:**
  Finally, we present a *unified training approach* that combines ***contrastive learning objectives*** (alignment and uniformity) with the ***optimal transport-based objective***,

allowing the encoder to simultaneously learn *semantic discrimination* and *structural coherence*. This joint framework enhances the quality of learned frame representations and leads to more accurate and consistent video summaries.

Together, these contributions form a versatile and effective framework for unsupervised video summarization, demonstrating competitive performance on standard benchmarks such as ***TVSum*** *and* ***SumMe***.

Following the Introduction, we take a look at the works that have been done in the past that are related to our work. Then we look at the methodology and different experiments that we have performed along with the results. Finally we conclude our work along with guidelines for future work.

Literature Review / Related Works

2.1 Why is it challenging?

Modeling *inter-frame dependencies* in video summarization is particularly challenging due to the *complex and highly non-uniform temporal relationships* present in videos. This is expected, as human viewers leverage *high-level semantic understanding* and track the *narrative progression* to judge the importance of frames for inclusion in a summary. For instance, while *temporally adjacent frames* often exhibit high visual similarity—making them candidates for redundancy removal—the *inverse does not necessarily hold*. Visually similar frames can appear far apart in time yet carry distinct narrative significance. Consider a video depicting the sequence: “leaving home in the morning, returning for lunch, and coming back again at night.” Although the frames representing the “at home” scenes may appear visually similar, the *semantic structure of the storyline* demands that all such scenes be retained in the summary. Therefore, summarization methods that rely solely on *low-level visual similarity* without accounting for *long-range semantic dependencies* risk *erroneously discarding* contextually important frames, ultimately degrading the quality of the generated summary.

2.2 Key Existing Methods

Early unsupervised video summarization techniques primarily relied on ***low-level statistical features*** such as motion descriptors, color histograms, and eigen-features to measure frame similarity or to perform clustering-based keyframe selection. These approaches often integrated ***visual, audio, and text cues*** to identify salient segments within a video. However, such heuristic-driven methods struggled to capture the *high-level semantic content* and complex temporal dependencies essential for generating coherent and informative summaries.

2.2.1 RNNs/LSTMs Based Approaches

With the advent of deep learning, a significant shift occurred toward *learning-based frameworks*, particularly those designed to model *temporal dynamics* across video frames. Early deep learning models employed ***Recurrent Neural Networks (RNNs)*** and their variants, especially ***Long Short-Term Memory (LSTM)*** networks, to address the challenge of capturing variable-range dependencies in video sequences.

A seminal work by *Zhang et al. (2016)* [1] introduced a *supervised approach* to video summarization by formulating it as a ***structured prediction problem*** over sequential data. Their method, known as ***dppLSTM***, employed LSTM networks to encode temporal dependencies and integrated ***Determinantal Point Processes (DPP)*** to promote diversity in the selected frames. Additionally, to mitigate the reliance on large-scale annotations, they incorporated ***domain adaptation*** strategies using auxiliary datasets with diverse content and styles, thereby enhancing generalization.

Building on this, *Mahasseni et al. (2017)* [2] proposed a *fully unsupervised framework* based on Generative Adversarial Networks (GANs). Their model featured an LSTM-based ***autoencoder summarizer***, which selected keyframes and attempted to reconstruct the original video, and an LSTM-based *discriminator*, which distinguished between real and reconstructed videos. The adversarial training paradigm encouraged the summarizer to learn to select *informative and representative frames* in the absence of ground-truth annotations. To enforce brevity and relevance, a *sparsity constraint* was introduced, ensuring that the generated summaries remained concise.

2.2.2 CNN-Based Approaches

To overcome the sequential bottlenecks of recurrent models, researchers have explored convolutional architectures for video summarization. *Rochan, Ye, and Wang (2018)* [3] introduced an innovative approach by formulating video summarization as a ***sequence labeling task*** and solving it using ***Fully Convolutional Sequence Networks (FCSN)***. Inspired by *semantic segmentation* in computer vision, their method draws a parallel between segmenting spatial features in images and segmenting important temporal segments in videos. By treating video frames as a 1D temporal sequence, FCSNs allow for end-to-end learning and efficient parallel computation, making them more scalable than traditional RNN-based models. Unlike recurrent networks that process frames sequentially, FCSNs operate on the entire sequence simultaneously, enabling better modeling of *long-range temporal dependencies*.

Expanding on this line of work, *Rochan and Wang (2019)* [4] tackled the problem of video summarization without requiring paired training data by introducing an *adversarial learning framework* for unpaired video summarization. Their model consists of two main components: (1) a *keyframe selector network*, implemented using FCSN, which identifies salient frames from raw input videos, and (2) a *summary discriminator*, also CNN-based, which learns to distinguish between real human-generated summaries and machine-generated ones. The training procedure combines adversarial loss to align the distributions of generated and real summaries,

reconstruction loss to ensure semantic fidelity, and a **diversity loss** to avoid redundancy among selected frames. This design allows the model to **learn summarization strategies from unpaired data**, thereby addressing the scalability issue of requiring annotated summary-video pairs.

2.2.3 Attention and Transformer-Based Methods

The introduction of *Transformer architectures* and *self-attention mechanisms* has significantly advanced the field of video summarization by enabling efficient parallel computation and improving the modeling of long-range temporal dependencies. Unlike recurrent models, which process frames sequentially, attention-based models allow each frame to directly attend to all others in the sequence, making them particularly well-suited for capturing global contextual relationships across time.

Fajtl et al. (2018) [5] proposed ***VASNet***, a supervised model that replaces traditional recurrent components with a *soft self-attention mechanism* for frame importance estimation. Instead of using BiLSTMs, VASNet processes the entire sequence in a single forward pass, dramatically improving *computational efficiency*. The model computes attention weights that quantify each frame's relevance relative to others, followed by a regression network that outputs frame-level importance scores. These scores are then used to select keyshots, ensuring that the summary captures the most significant and non-redundant content.

Building on this direction, *Apostolidis et al. (2021)* [6] introduced ***PGL-SUM***, a supervised method that further enhances attention mechanisms by combining *global and local multi-head self-attention*. This hybrid design enables the model to capture both *fine-grained local interactions* and *global context* among frames. Additionally, they incorporate *absolute positional encodings* to retain temporal order information, which is otherwise lost in standard attention operations.

More recently, Pang et al. (2023) [7] proposed a fully unsupervised contrastive learning-based framework that eliminates the need for summary bootstrapping or handcrafted objectives. Their method introduces three ***contrastive losses—local dissimilarity, global consistency***, and ***uniqueness***—to assess frame-level importance directly. By applying these losses to features extracted from pre-trained image encoders, the model learns to distinguish semantically important frames without any fine-tuning or supervision. A *lightweight contrastive projection module* is used to refine the feature space and improve generalization. Despite its simplicity, the method achieves competitive or superior performance on standard datasets, showcasing the power of contrastive objectives and self-supervised attention mechanisms in unsupervised video summarization.

2.2.4 Multimodal and Other Learning Techniques

Recent advancements in video summarization have increasingly explored *multimodal learning*, particularly the integration of *visual and textual modalities*, to better capture the high-level semantics of video content and overcome the limitations of single-modality approaches.

Wang et al. (2023) [8] introduced a *self-supervised video summarization framework* that leverages ***semantic inverse optimal transport (IOT)*** to align visual and textual modalities without requiring manual annotations. Their method first generates *textual descriptions* for video shots and learns a projection that maps these text embeddings into the visual embedding space. The alignment is formulated as an *inverse optimal transport problem*, which ensures semantically meaningful correspondences between the modalities. An *alternating optimization strategy* is employed to solve the alignment efficiently while avoiding degenerate solutions. The derived ***optimal transport plan*** is then used to compute ***pseudo-significance scores*** for video frames, which serve as *offline supervision* for training a keyframe selector in a fully unsupervised setting.

Narasimhan et al. (2021) [9] proposed ***CLIP-It***, a *unified language-guided multimodal transformer* designed for both *generic* and *query-focused video summarization*. The model utilizes Language-Guided Attention to fuse frame-level visual features with either natural language queries or automatically generated dense video captions. A dedicated *Frame-Scoring Transformer* assigns importance scores based on relevance to the textual input. These scores are aggregated into shot-level scores, and a *knapsack algorithm* selects the most relevant segments to construct the final summary. Notably, CLIP-It is versatile—it supports both *supervised and unsupervised settings* by using *reconstruction and diversity losses* in the absence of labeled data. Extensive evaluations on datasets such as TVSum, SumMe, and QFVS show that CLIP-It achieves state-of-the-art results, especially in cross-domain and transfer scenarios, highlighting its strong generalization capabilities.

Li et al. (2023) [10] (**SSPVS**) presented a self-supervised multimodal learning framework for video summarization that eliminates the need for manual annotations by jointly learning from visual and textual cues. Their approach incorporates both *coarse-grained* and *fine-grained alignment objectives* to enforce semantic consistency between video frames and associated textual descriptions. To model temporal dynamics, they introduce a *frame recovery task*, enabling the network to learn temporal structure and continuity. Furthermore, they propose a *progressive summarization strategy* that refines frame selections across multiple stages, gradually improving the quality of the summary.

2.3 Our Contributions

While unsupervised video summarization has gained increasing attention due to its scalability and reduced annotation costs, many existing methods still suffer from significant limitations. A large number of prior approaches rely on hand-crafted heuristics, such as diversity or representativeness criteria to determine frame importance. Although effective to some extent, these methods often fail to capture the high-level semantic relationships and contextual understanding that human annotators naturally consider when creating summaries.

Recently, contrastive learning-based methods [7] have emerged as promising alternatives by learning frame representations that emphasize semantic distinctions. These methods typically encourage semantic alignment of related frames and uniform distribution of frame embeddings,

enabling models to assess frame importance based on learned similarities. However, such methods may overlook structural consistency, i.e., the preservation of relational information and global context across frames.

To overcome these shortcomings, we propose a *novel framework* that **combines contrastive refinement with optimal transport-based alignment**. While the *contrastive component* guides the model to learn *semantically meaningful frame representations*, the *optimal transport* objective provides a *structure-aware loss* that aligns frame features with *learnable cluster centers* based on their global relational patterns. This joint formulation allows our model to benefit from both ***semantic discriminability and structural coherence***, leading to more informative and contextually consistent video summaries.

Methodology

3.1 Problem Definition

Unsupervised video summarization is formulated as the task of selecting a *sparse subset of frames* from a video such that the selected frames *optimally represent* the content of the original video. Since no human-annotated ground-truth summaries are available in this setting, the model must infer frame-level importance based solely on the underlying structure and content of the video. To generate a summary, we first assign ***importance scores*** to each frame. These scores are then aggregated at the shot level, and a ***0/1 knapsack algorithm*** is used to select the most informative and diverse shots that fit within a specified summary length constraint.

3.2 Proposed Method Overview

We process each video by sampling frames at *2 frames per second (fps)* and extract *1024-dimensional visual features* using an ImageNet-pretrained GoogleNet [11] model. The core challenge lies in determining the *frame-level importance scores* in the absence of supervision. To this end, we explore several models and learning strategies aimed at capturing *variable-range temporal dependencies* and *semantic relevance* among frames.

Our work explores a diverse set of approaches for frame scoring and summarization, including:

- **TCN-based scoring with reconstruction objectives**, enabling the model to learn temporal structures by reconstructing missing frames.
- **TCN-based scoring with local consistency constraints**, designed to preserve structural coherence through neighbor-aware training.
- **Contrastive learning with TCN encoders**, promoting discriminative and temporally consistent frame representations.
- **Transformer-based architectures trained with contrastive objectives**, serving as strong baselines for modeling global temporal relationships.

- **Hierarchical Transformer networks**, capturing both local and global temporal dependencies in a structured manner.
- **Clustering-based scoring using K-Means and Optimal Transport**, where frame-to-cluster alignment serves as a proxy for representativeness.
- **A unified learning framework** that jointly optimizes contrastive and Optimal Transport objectives, encouraging both semantic discrimination and structural alignment for more meaningful summarization.

3.3 Detailed Methodology

The subsequent sections provide a detailed explanation of each of these approaches, their motivations, and how they contribute to the task of unsupervised video summarization.

3.3.1 Transformer-Based Contrastive Learning Framework

As a baseline, we implemented the ***contrastive learning framework*** proposed by Pang et al. [7], which formulates unsupervised video summarization as a representation learning problem guided by the contrastive objectives. Specifically, we employed PyTorch's Transformer Encoder as the backbone network to model frame-level interdependencies. To investigate the impact of *temporal position encoding*, we conducted ablation experiments by introducing *learnable positional encodings* added to the input features before passing them to the Transformer Encoder.

Following the original work, we adopted the ***alignment*** *and* ***uniformity*** contrastive losses as our learning objectives. We excluded the *uniqueness loss* due to its dependency on an additional network for *background frame identification*, which was beyond the scope of our current implementation and significantly increased the model's complexity.

The *alignment* loss encourages similar frames to be projected close together, while the *uniformity* loss encourages a well-distributed representation space, preventing feature collapse. The formulation of these losses are given below:

<u>Local Dissimilarity: Alignment Loss</u>

To promote diversity in the summary, we consider frames as informative if they are distinct from their semantic neighbors. Given a video V, we extract deep features using an ImageNet-pretrained backbone (e.g., GoogleNet [11]), denoted as

$$F(V) = \{\mathbf{x}_t\}_{t=1}^{T}, \quad \|\mathbf{x}_t\|_2 = 1$$

where $\mathbf{x}_t$ is the L2-normalized feature of the t-th frame.

These features are passed through a learnable module G_θ to obtain

$$\mathbf{z}_t = G_\theta(\mathbf{x}_t), \quad \|\mathbf{z}_t\|_2 = 1$$

For each frame, we find a set $\mathcal{N}_t$ of top K = aT neighbors, (where a is a hyperparameter and K is rounded to the nearest integer) using cosine similarity in the original feature space $\{\mathbf{x}_t\}$. The **local alignment loss** in the learned space $\{\mathbf{z}_t\}$ is then defined as:

$$\mathcal{L}_{\text{align}}\ (\mathbf{z}_t;\theta) = \frac{1}{|\mathcal{N}_t|}\sum_{\mathbf{z}\in\mathcal{N}_t}\|\mathbf{z}_t-\mathbf{z}\|_2^2 \qquad (1)$$

<u>Global Consistency: Uniformity Loss</u>

The set $\mathcal{N}_t$ may include semantically irrelevant frames, especially when $\mathbf{x}_t$ has few meaningful neighbors within the video. To address this, and drawing inspiration from the reconstruction-based representativeness objective [2], the uniformity loss to assess how well a frame aligns with the overall theme or gist of the video, is defined as:

$$\mathcal{L}_{\text{uniform}}\ (\mathbf{z}_t;\theta) = \log\left(\frac{1}{T-1}\sum_{\mathbf{z}\neq\mathbf{z}_t,\mathbf{z}\in G_\theta(F(\mathbf{V}))} e^{-2\|\mathbf{z}_t-\mathbf{z}\|_2^2}\right) \qquad (2)$$

The final loss function used for guiding the training is then given below:

$$\mathcal{L}\left(\mathbf{z}_t;\theta\right) = \mathcal{L}_{\text{align}}\ (\mathbf{z}_t;\theta) + \lambda_1\mathcal{L}_{\text{uniform}}\ (\mathbf{z}_t;\theta) \qquad (3)$$

The architectural diagram is given below:

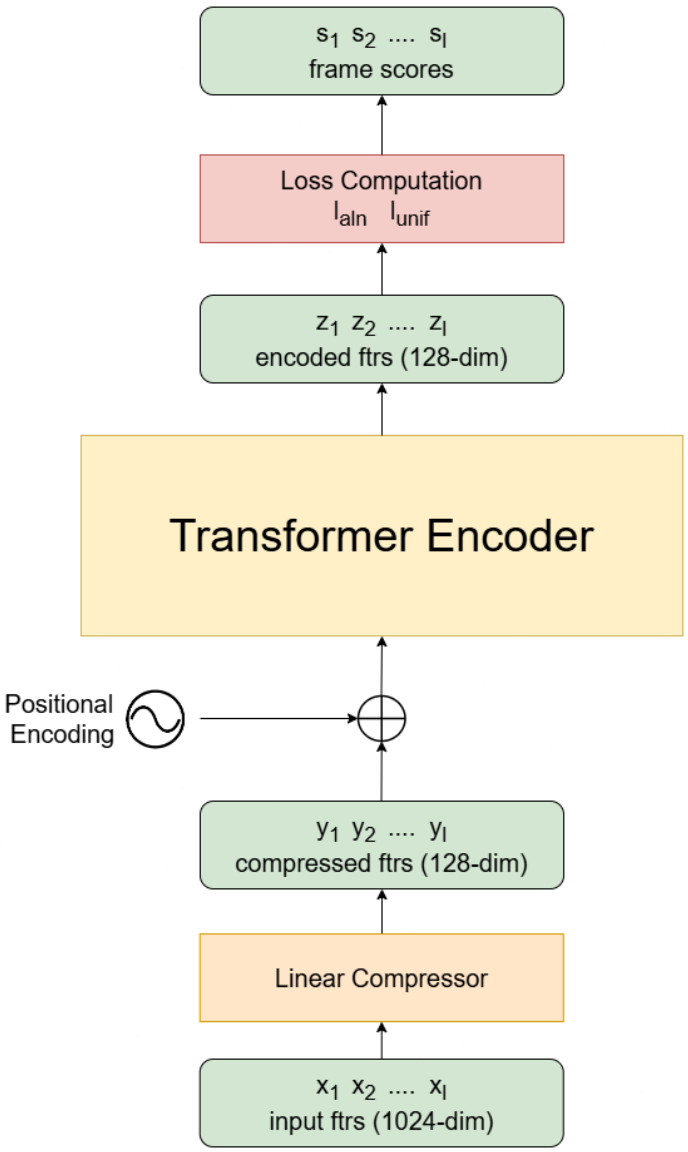

Figure 2: The architecture of the transformer-based contrastive baseline method

3.3.2 Hierarchical Transformer Network

To extend the baseline, we designed a ***Hierarchical Transformer Network*** to better capture both ***local and global temporal dependencies*** in long video sequences. Given the observation that processing very long sequences with a single Transformer can be computationally expensive and potentially ineffective, we introduced a *two-stage hierarchical attention mechanism*.

First, the input video was *divided into fixed-length segments*, allowing the model to process manageable sub-sequences. A **Transformer-based Feature Extractor**, shared across all segments, was applied to each segment independently. This component captures *local temporal dependencies* within each segment, ensuring that fine-grained frame interactions are preserved.

Subsequently, the *segment-level frame features* produced by the feature extractor were ***aggregated globally*** using a ***lightweight Transformer-based Feature Aggregator***. This aggregator applies self-attention and feedforward layers to model global temporal relationships across the entire video by attending to the aggregated segment-level features. This two-stage hierarchical design allows the network to efficiently scale to long video sequences while preserving both **local** and **global** temporal context.

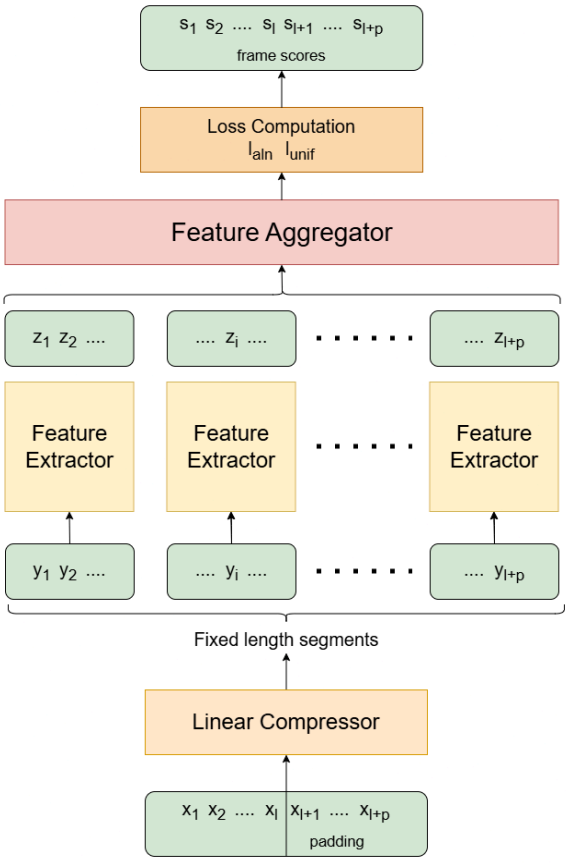

Figure 3: The architecture of hierarchical transformer based contrastive method.

3.3.3 Clustering-Based Frame Importance Scoring using K-Means

To establish a non-deep learning baseline, we implemented a *continuous K-Means clustering-based framework* for learning frame importance scores. The intuition was to treat cluster centers as proxies for *representative visual patterns* in the video. Frames that are closer to cluster centers are considered *more informative* and are assigned *higher importance scores.*

Given a set of deep frame features $\{\mathbf{x}_t\}_{t=1}^{T}$, we apply K-Means clustering to partition them into K clusters by minimizing the within-cluster variance:

$$\min_{\{\mu_k\}_{k=1}^{K}} \sum_{t=1}^{T} \min_{k \in \{1,\dots,K\}} \|\mathbf{x}_t - \mu_k\|_2^2 \qquad (4)$$

Here, μ_k denotes the centroid of the k-th cluster. After clustering, each frame is assigned an importance score based on its proximity to the nearest cluster center.

Let $d_t = \min_k \|\mathbf{x}_t - \mu_k\|_2$ be the Euclidean distance between frame $\mathbf{x}_t$ and its nearest cluster centroid. We compute the vector of distances $\mathbf{d} = [d_1, d_2, \dots, d_T]$, take the element-wise inverse to obtain $\mathbf{s} = [1/d_1, 1/d_2, \dots, 1/d_T]$, and normalize it using min-max normalization:

$$\hat{s}_t = \frac{s_t - \min(\mathbf{s})}{\max(\mathbf{s}) - \min(\mathbf{s})} \qquad (5)$$

The resulting normalized scores $\hat{s}_t \in [0, 1]$ are treated as the importance scores for each frame and are used for summary generation.

3.3.4 Optimal Transport-Based Frame Scoring

Inspired by recent advances in ***Optimal Transport (OT)*** for structured data alignment , we replaced K-Means with a **learnable Optimal Transport framework**.

Let a video be represented by frame features:

$$\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$$

where $\mathbf{x}_i \in \mathbb{R}^d$ are frame-level features.

Let $\mathbf{C} = \{\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_K\}$be **learnable cluster centers**, with $\mathbf{c}_j \in \mathbb{R}^d$.

We define the **pairwise cost matrix** $\mathbf{M} \in \mathbb{R}^{N \times K}$ as:

$$M_{ij} = \|\mathbf{x}_i - \mathbf{c}_j\|_2$$

Also, we define uniform mass distributions for both sides:

$$\mathbf{u} = \frac{1}{N}\mathbf{1}_N, \quad \mathbf{v} = \frac{1}{K}\mathbf{1}_K$$

where $\mathbf{1}_N$ and $\mathbf{1}_K$ are vectors of ones.

Then the **optimal transport plan** $\mathbf{P}$is computed by solving:

$$\mathbf{P}^* = \arg\min_{\mathbf{P}\in\Pi(\mathbf{u},\mathbf{v})} \langle \mathbf{P}, \mathbf{M} \rangle \tag{6}$$

where:

- $\langle \mathbf{P}, \mathbf{M} \rangle = \sum_{i=1}^{N}\sum_{j=1}^{K} P_{ij} M_{ij}$
- $\Pi(\mathbf{u}, \mathbf{v})$ is the set of valid transport plans with marginals $\mathbf{u}$ and $\mathbf{v}$.

And finally, the **Wasserstein loss** is then defined as:

$$\mathcal{L}_{\text{Wass}} = \sum_{i=1}^{N}\sum_{j=1}^{K} P^*_{ij} \|\mathbf{x}_i - \mathbf{c}_j\|_2 \tag{7}$$

This objective encourages the **cluster centers** to align with the **frame features** by minimizing the **transport cost**, measured here by **Euclidean distance**.

As with K-Means, we computed the frame scores based on **the distance** between the frames and their closest cluster centers. This method allowed for **more flexible and structured alignment** compared to K-Means.To compute the **frame importance scores** from the learned cluster centers:

1. For each frame x_i, compute its **Euclidean distance** to all cluster centers $\{c_j\}_{j=1}^{K}$.
2. Select the **minimum distance** (i.e., closest center):
   $$d_i = \min_j \|x_i - c_j\|_2$$

3. Compute the **inverse** of this distance:

$$s_i = \frac{1}{d_i + \epsilon}$$, where ϵ is a small constant added for numerical stability.

4. Normalize all s_i across the video using **min-max scaling**:

$$s_i^{\text{norm}} = \frac{s_i - \min(s)}{\max(s) - \min(s)} \tag{8}$$

These normalized scores s_i^{norm} are used as **frame-level importance scores** for summary generation.

In this setup, we optimized the parameters of the model (including the learnable cluster centers) as follows:

- **Optimal Transport Plan Computation**:
  First the optimal transport plan is computed using the objective in equation (6), **soft alignment** between frame features and cluster centers, enabling **probabilistic matching** rather than hard assignments.
- Gradient update:
  Once the Optimal Transport Plan is computed, then the wasserstein loss is computed as the sum of the product of individual elements of the optimal transport plan and the cost matrix which guides the training of our model.

We evaluated this OT-based method as a **direct benchmark** against the K-Means baseline.

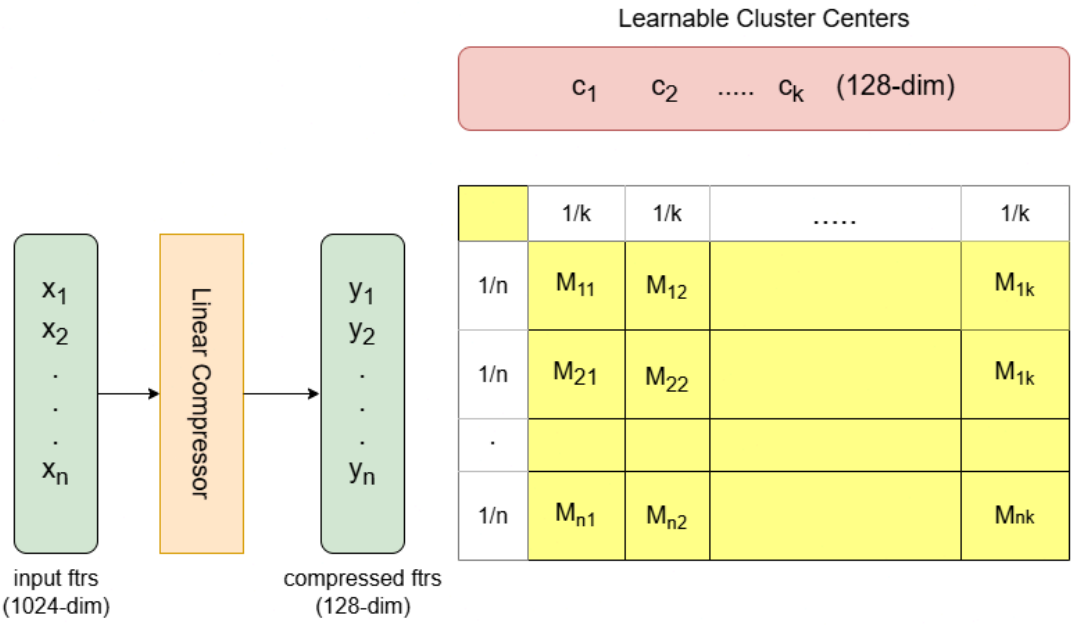

Figure 4: Framework for Optimal Transport based Frame Scoring Method

3.3.5 Combining Contrastive Refinement with Optimal Transport

Finally, we extended our investigation by **combining contrastive learning and optimal transport-based scoring**. We did a couple of experiments here. Specifically, we first **learned frame representations** using the **Transformer-based contrastive learning framework** described in Section 3.3.1. We then applied the **Optimal Transport-based clustering mechanism** on top of these **contrastively refined features**.

The motivation behind this combination was to **leverage the strengths of both methods**:

- **Contrastive refinement** ensures **semantically meaningful representations**.
- **Optimal Transport** provides **structurally aware alignment** to **estimate frame importance scores**.

We hypothesized that **contrastively pre-trained features** would result in **better transport plans and cluster centers**, ultimately leading to **improved summary quality**.

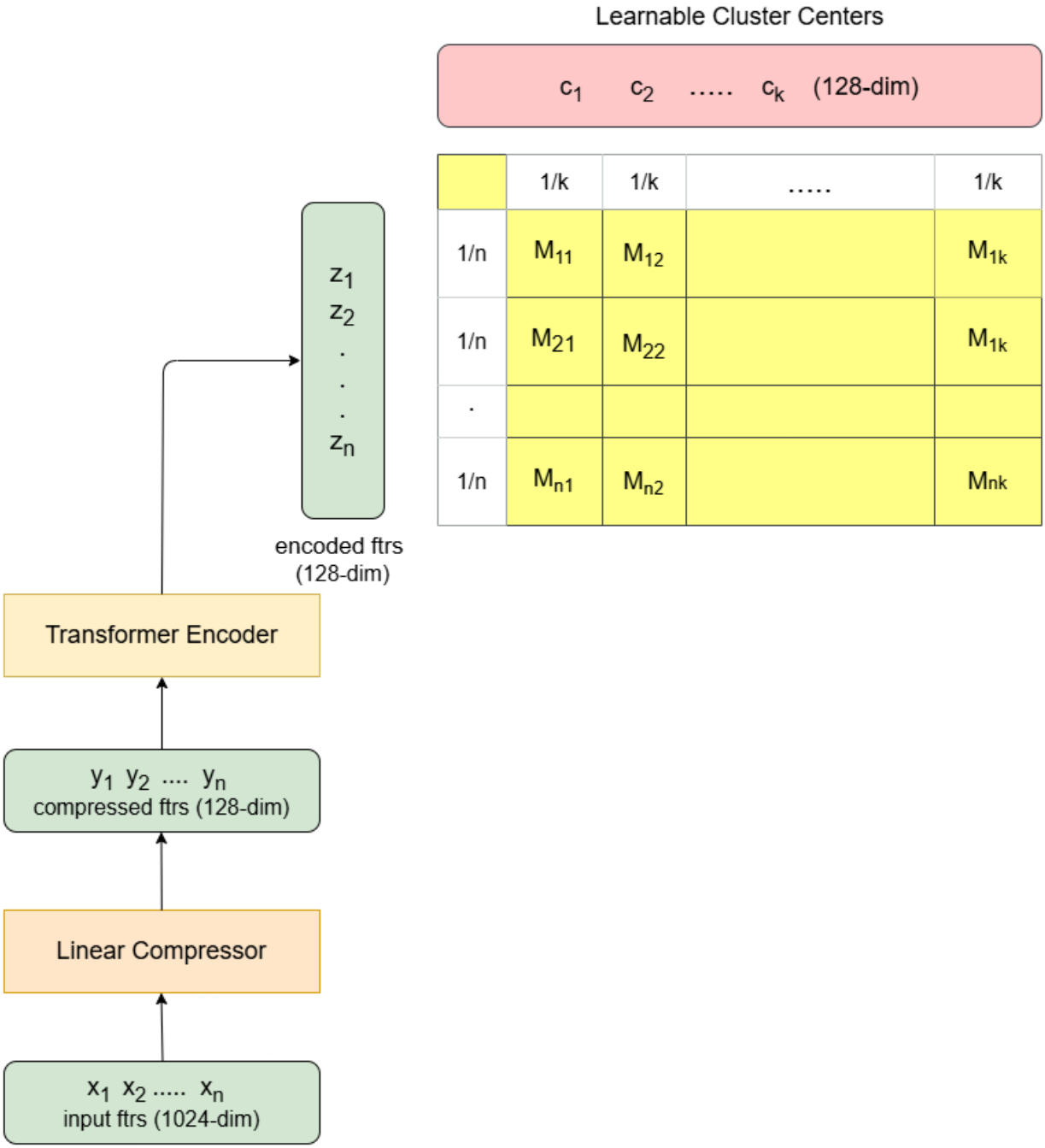

Figure 5: Framework for Optimal Transport Guided Contrastive Video Summarization Method

Bai et al (2018) [14] conducted a comprehensive empirical evaluation comparing Temporal Convolutional Networks (TCNs) with traditional recurrent architectures like LSTMs and GRUs for sequence modeling tasks. TCNs are designed with causal, dilated convolutions and residual connections, enabling them to model long-range temporal dependencies effectively. Unlike recurrent networks, TCNs allow for parallel processing and exhibit stable gradients, making

them more efficient and scalable. They show that despite the theoretical ability of recurrent architectures to capture infinitely long history, TCNs exhibit substantially longer memory, and are thus more suitable for domains where a long history is required. Therefore, inspired by their work we explored various TCN-architectures and techniques for unsupervised video summarization which we will discuss now.

3.3.6 Missing Frame Reconstruction Using TCNs

To evaluate the model's ability to **learn temporal structure from sparse inputs**, we experimented with a **sampling strategy** where **only the odd-indexed frames** were provided to a **TCN-MLP scoring and TCN-ED reconstruction pipeline**. This deterministic sampling forced the model to **infer missing frames** from **systematically reduced temporal resolution**.

Two loss functions were employed during training:

- Missing Frame Reconstruction Loss, computed as the mean squared error (MSE) between the original and reconstructed even-indexed frames.
- Frobenius Loss, which ensures the preservation of relational structure by minimizing the Frobenius norm between the similarity matrices of the original and reconstructed frame sequences.

More formally, the frobenius loss is defined below:

Given **batch input features** $\mathbf{X} \in \mathbb{R}^{B \times D \times L}$, the **pairwise similarity matrix** for each batch sample is computed as:

$$\mathbf{S}^{(b)} = \frac{1}{\sqrt{D}} \mathbf{X}^{(b)^\top} \mathbf{X}^{(b)}$$

Where:

- $\mathbf{X}^{(b)} \in \mathbb{R}^{D \times L}$ is the **feature matrix** for batch bbb,
- $\mathbf{S}^{(b)} \in \mathbb{R}^{L \times L}$ is the **similarity matrix**, scaled by $\frac{1}{\sqrt{D}}$ for normalization.

This is computed **for both**:

- Original features $\mathbf{X}_{\text{orig}}$

- Reconstructed features $\mathbf{X}_{\text{recons}}$

The **Frobenius Loss** is then the **average Frobenius norm** of the **difference** between the two similarity matrices across the batch:

$$\mathcal{L}_{\text{Fro}} = \frac{1}{B} \sum_{b=1}^{B} \left\| S_{\text{orig}}^{(b)} - S_{\text{recons}}^{(b)} \right\|_F \tag{9}$$

Where:

- $\| \cdot \|_F$ denotes the **Frobenius norm** (square root of the sum of squared matrix entries).

  This formulation was designed to evaluate the model's robustness in reconstructing missing temporal context from systematically sampled sparse frame sequences.

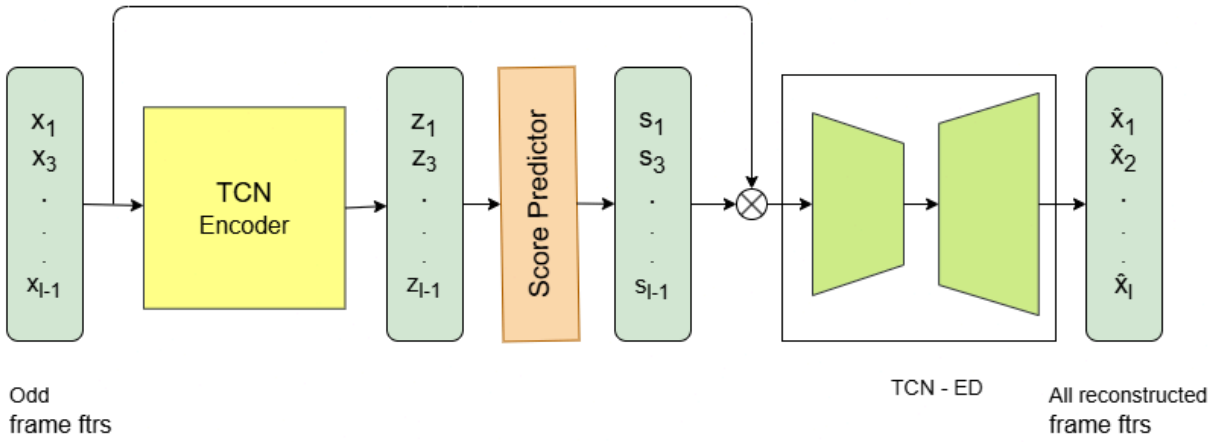

Figure 6: Architecture for Video Summarization via Missing Frame Reconstruction using TCN ED network

3.3.7 Neighbor Consistency and Structural Preservation using TCN architecture

We further explored a **TCN-based scoring framework** where a **Temporal Convolutional Network (TCN)** is used to encode frame-level features, followed by an **MLP head** that predicts frame importance scores. This design leverages the TCN's capability to model **local temporal patterns** while producing **dense importance estimates** over the input sequence.

To encourage **local consistency** among the most important frames, we define a **neighbor consistency loss** as follows. First, we select the **top-k scoring frames** based on the predicted scores. We extract the corresponding **encoded features** of these keyframes, **normalize them**, and identify their **n-nearest non-keyframe neighbors** using **cosine similarity**. We then **scale the keyframe features by their predicted scores** and compute the **Euclidean distances** between these scaled keyframes and their nearest neighbors. The distances across all keyframes and their neighbors are aggregated into a matrix, and the **Frobenius norm** of this matrix is computed as the **neighbor consistency loss**, which penalizes isolated or inconsistent keyframe selections.

Given:

- Extracted frame features $\mathbf{F} = \{\mathbf{f}_1, \ldots, \mathbf{f}_L\}$, where $\mathbf{f}_i \in \mathbb{R}^d$, and
- Predicted importance scores $\mathbf{s} = \{s_1, \ldots, s_L\}$,

The Neighbour loss is computed as given below:

Select the indices of the *top-k scoring frames*:

$$\mathcal{K} = \mathrm{TopK}(\mathbf{s})$$

Let $\mathbf{f}_{\mathrm{key},j}$ denote the *feature vector of the j-th top-scoring frame*, scaled by its score s_j:

$$\mathbf{z}_j = s_j \cdot \mathbf{f}_{\mathrm{key},j}$$

Let $\mathcal{N}$ be the set of *non-keyframes* (frames not in $\mathcal{K}$).

For each keyframe $\mathbf{f}_{\mathrm{key},j}$, retrieve the *top-n nearest neighbors* $\mathbf{f}^{(1)}_{\mathrm{neighbor},j}, \ldots, \mathbf{f}^{(n)}_{\mathrm{neighbor},j}$ from $\mathcal{N}$ *based on cosine similarity*.

Compute the *Euclidean distance* between the **score-scaled keyframe** $\mathbf{z}_j$ and its **n nearest neighbors**:

$$d_j^{(l)} = \|\mathbf{z}_j - \mathbf{f}_{\text{neighbor},j}^{(l)}\|_2$$

For $l = 1, \ldots, n$.

Construct a **distance matrix** $\mathbf{D} \in \mathbb{R}^{k \times n}$ by stacking the distances for all keyframes and their neighbors.

The **Neighbor Consistency Loss** is defined as the **Frobenius norm** of the distance matrix:

$$\mathcal{L}_{\text{neighbor}} = \|\mathbf{D}\|_F \tag{10}$$

where $\|\cdot\|_F$ is the **Frobenius norm**, computed as the square root of the sum of squared entries.

In addition to this local consistency objective, we introduce a **structural preservation loss** to maintain the **pairwise relational structure** between the **original input features** and the **TCN-encoded features**. Specifically, we compute the **pairwise cosine similarity matrices** of both the input and encoded features, and minimize the **Frobenius norm of their difference**. This ensures that the temporal and semantic relationships present in the original feature space are preserved after transformation by the TCN encoder.

Together, these objectives encourage the model to produce **locally coherent**, **structurally consistent**, and **semantically meaningful** frame importance scores, improving the quality and interpretability of the resulting video summaries.

The architecture diagram for the same is given below:

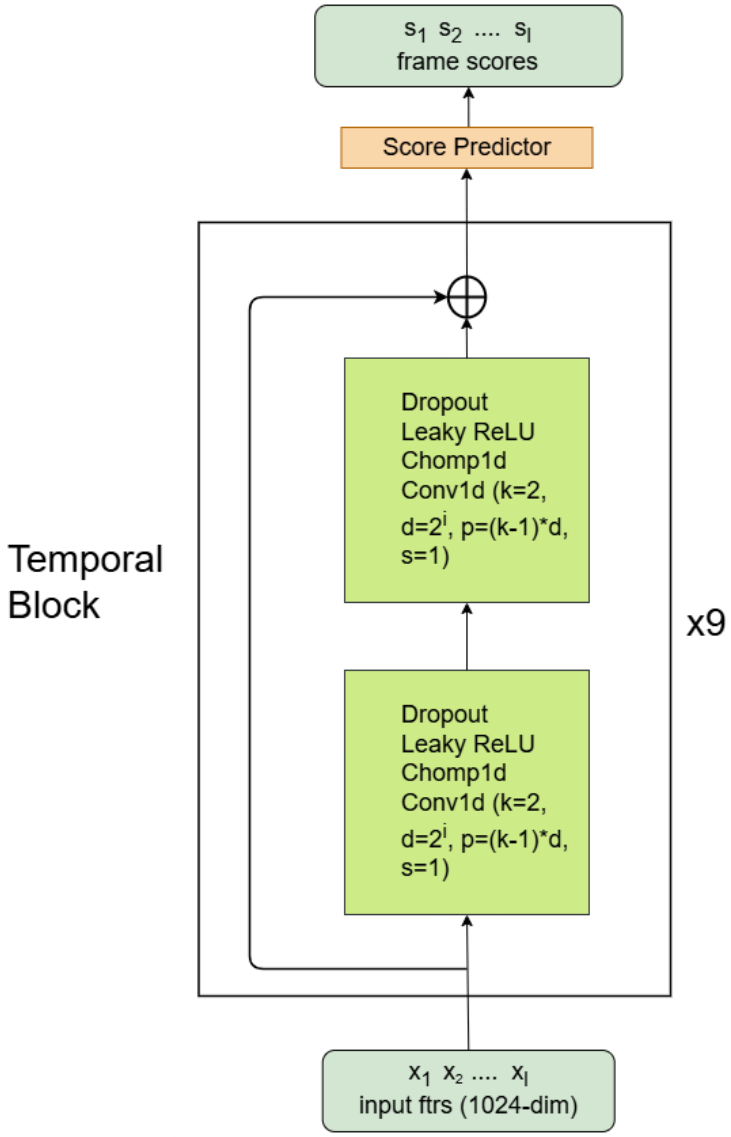

Figure 7: Architecture diagram for TCN-based Video Summarization method using Neighborhood Consistency Loss.

3.3.8 Contrastive Learning with TCN Encoder

While the baseline contrastive learning method employed a **Transformer-based encoder**, we also investigated the feasibility of replacing the Transformer with a **Temporal Convolutional Network (TCN)**. Given the reported effectiveness of TCNs over RNNs in various sequence modeling tasks, we hypothesized that TCNs might also perform well in contrastive learning settings.

In this variant, the TCN encoder processed the frame sequence to produce latent representations, and the model was trained using: the alignment loss which encourages the

semantically similar frames to have similar representations and the uniformity loss which encourages well-distributed embeddings in the feature space.

Additionally, we also tried incorporating reconstruction of the missing features along with the contrastive objectives but the results were not in our favor. The rank correlation coefficients came out to be negative.

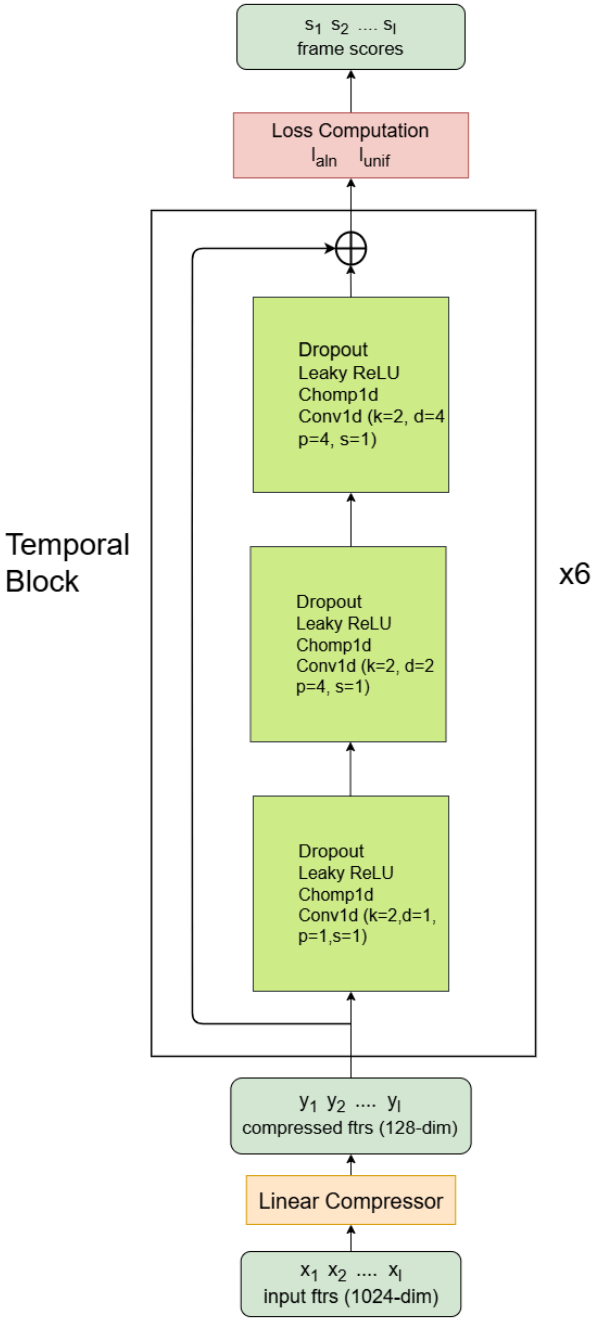

Figure 8: Architecture Diagram for TCN-based contrastive video summarization method

Experiments & Results

4.1 Dataset Description

Following previous work, we evaluate our method on two benchmarks: **TVSum** [15] and SumMe [16]. TVSum contains 50 YouTube videos, each annotated by 20 annotators in the form of importance scores for every two second-long shot. SumMe includes 25 videos, each with 15-18 reference binary summaries.

4.2 Evaluation Metrics

4.2.1 F1 Score.

Let A denote the set of frames in the ground-truth summary and B the set of frames in the generated summary.

The Precision and recall are defined as:

$$\text{Precision} = \frac{|A \cap B|}{|A|}, \quad \text{Recall} = \frac{|A \cap B|}{|B|}$$

The F1 score, which balances both precision and recall, is then computed as:

$$\text{F1} = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{11}$$

We follow the procedure outlined in [1] to handle multiple ground-truth summaries and to convert continuous importance scores into binary summaries.

4.2.2 Rank Correlation Coefficients

Recent work by Otani et al. [19] highlighted the limitations of F1 score, showing that it can yield high values even for randomly generated summaries. As a more robust alternative, they proposed using rank correlation coefficients—**Kendall's τ** [17] and **Spearman's ρ** [18]—to measure the agreement between predicted and ground-truth importance scores.

For each video, we compute the correlation coefficient between the predicted importance scores and each annotator's scores. These values are then averaged across all annotators for that video. The final metric is obtained by averaging the resulting coefficients across all videos in the dataset.

4.3 Experimental Results

4.3.1 Transformer-Based Contrastive Learning Framework

Implementation details

The input frame features—padded or trimmed to a fixed sequence length of 256—are first passed through a Transformer encoder comprising 4 layers and 8 attention heads. The resulting encoded representations are used to compute the contrastive losses, specifically alignment and uniformity which are further used to compute the frame importance scores. The model is trained for 200 epochs with a batch size of 8.

We experimented with two different input feature dimensionalities:

- **1024-dimensional features** directly extracted from the pre-trained GoogleNet backbone.

- **128-dimensional compressed features**, obtained by introducing a learnable linear projection layer before the Transformer encoder. This projection layer is trained end-to-end along with the model, enabling the network to learn compact yet informative frame-level embeddings.

Additionally, we conducted ablation experiments to assess the impact of incorporating positional embeddings into the training pipeline.

Results

Using 1024-dim features:

Table 1: Transformer contrastive baseline results using 1024-dim features

Dataset	With positional embeddings			Without positional embeddings		
	F-score	Kendall's τ	Spearman's ϱ	F-score	Kendall's τ	Spearman's ϱ
TVSum	58.062	0.09855	0.13049	**59.123**	**0.13470**	**0.17708**
Summe	42.700	0.12510	0.16306	**46.130**	**0.14103**	**0.19174**

Using 128-dim features:

Table 2: Transformer contrastive baseline results using 128-dim features

Dataset	With positional embeddings			Without positional embeddings		
	F-score	Kendall's τ	Spearman's ϱ	F-score	Kendall's τ	Spearman's ϱ
TVSum	57.534	0.12845	0.16854	**59.620**	**0.13920**	**0.18456**
Summe	44.789	**0.17003**	**0.23350**	**47.026**	0.11549	0.16327

Observations

- From the above results, we observe that the model achieves better performance overall using the 128-dimensional compressed features compared to the original 1024-dimensional features. Consequently, we adopt the 128-dimensional representation for all subsequent experiments, as it offers both improved performance and reduced computational and memory overhead.
- Additionally, incorporating positional encodings was found to degrade model performance. As a result, moving forward we won't be using the positional encoder.

4.3.2 Hierarchical Transformer Based Contrastive Learning Framework

Implementation details

The input frame features are first padded to ensure that the sequence length is a multiple of the segment length (128). These features are then passed through a Transformer-based feature extractor comprising 3 layers and 8 attention heads. The resulting encoded frame-level representations are subsequently fed into a lightweight Transformer-based feature aggregator consisting of 2 layers and 8 heads.

The final aggregated representations are used to compute the contrastive losses—alignment and uniformity which are further used to compute the frame importance scores. The model is trained for 200 epochs with a batch size of 1.

Results

Table 3: Hierarchical Transformer contrastive method results

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	56.941	0.12469	0.16339
Summe	44.890	0.11502	0.15789

Observations

The performance of the hierarchical contrastive model was found to be slightly lower than that of the single transformer-based contrastive baseline. We hypothesize that this may be due to the following factors:

- Given the limited size of the datasets, the model may have overfitted to the training videos, resulting in reduced generalization on the test set. This approach may be more effective for longer videos with richer temporal structures.
- We observed that padded frames consistently received disproportionately high scores. This could be attributed to the nature of the alignment and uniformity loss functions, which may inadvertently favor such frames.

Since the padded frames were having significantly higher scores than other frames, we thought of trimming the scores to see if it benefits the performance metrics.

Implementation details

The implementation follows the same setup as previously described, with one modification during the validation phase. Prior to computing the final scores, we truncate the laln and lunif vectors to match the original (unpadded) sequence length, thus discarding the score computation for the padded frames.

Results

Table 4: Hierarchical Transformer contrastive method results using trimmed scores

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	57.275	0.12626	0.16547
Summe	44.890	0.11756	0.16141

Observations

- From the above results we can see that trimming the loss vectors for score computations have indeed helped to improve the performance metrics although the improvement is small.
- Based on these findings, we choose to proceed with a single Transformer encoder in subsequent experiments wherever applicable, favoring simplicity and efficiency in the model design.

4.3.3 Clustering-Based Frame Importance Scoring using K-Means

Implementation details

We implemented a batch-wise K-Means clustering framework, where cluster centers are updated progressively across epochs rather than recomputed from scratch. This enables stable and continuous learning of representative centers over time.

To compute frame-level importance scores, we first calculate the Euclidean distance from each frame to its nearest cluster center. These distances are then inverted and normalized to the $[0, 1]$ range using min-max scaling. The resulting scores highlight the fine-grained, localized relevance of each frame relative to the learned cluster structure.

We used 1000 cluster centers, initialized randomly, and operated in a 1024-dimensional feature space. The model was trained for 200 epochs with a batch size of 4.

Results

The results obtained on the TVSum and SumMe datasets are given below:

Table 5: K-Means based clustering baseline method results

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	57.071	0.06698	0.08839
Summe	48.823	-0.04302	-0.06297

Observations:

- We found that the training process for K-Means clustering was notably unstable. Since the model is exposed to different sets of video frames in each training step, the cluster centers struggle to converge consistently over epochs.
- Additionally, the results show that the rank correlation coefficients for the SumMe dataset are negative. One possible explanation is that a large proportion of frames in SumMe—approximately 85%—have ground-truth importance scores of zero, reflecting

the dataset's sparsely labeled nature. In contrast, our method assigns a non-zero importance score to every frame based on its proximity to the nearest cluster center. Thus the predicted scoring may disrupt the rank order alignment with the sparse ground-truth annotations, leading to unreliable or unstable rank correlation values.

4.3.4 Optimal Transport-Based Frame Scoring

Implementation details

We make the cluster centers as learnable parameters of the model and initialized them from a standard normal distribution. We considered 1000 cluster centers. In addition to the cluster centers, we also learned a linear layer for compressing the dimensionality of the features to 128-dimensions. We ran the training for 200 epochs and used one video at a time for each step of the optimizer.

Results

Table 6: Optimal Transport based framework results on benchmark datasets

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	62.178	0.14891	0.19605
Summe	44.784	0.05644	0.07674

Observations:

- The results show that the Optimal Transport-based scoring method outperformed the contrastive baseline on the TVSum dataset. On the SumMe dataset, it demonstrated competitive performance, particularly in terms of the F-score.

4.3.5 Combining Contrastive Refinement with the Optimal Transport Framework

Implementation details

The input features, padded or trimmed to a fixed sequence length of 256, are first passed through a Transformer encoder with 4 layers and 8 attention heads. The resulting encoded representations are used to learn cluster centers via an Optimal Transport-based training objective. The Wasserstein loss not only optimizes the cluster centers but also backpropagates through the Transformer encoder, thereby reinforcing and refining the frame representations to capture richer temporal and semantic cues.

Results

Initially, the Transformer encoder is pretrained on the TVSum and SumMe datasets using contrastive objectives. This pretrained encoder is then integrated into the Optimal Transport framework and trained end-to-end using both the contrastive and Wasserstein losses. Frame importance scores, in this setting, are computed using the contrastive alignment and uniformity losses. The performance is summarized below:

Table 7: Joint Optimization framework results on benchmark datasets when using the pretrained transformer weights and computing scores using Contrastive framework

Dataset	Transformer encoder weights (pretrained) trainable (scores using contrastive losses)		
	F-score	Kendall's τ	Spearman's ϱ
TVSum	60.063	0.13220	0.17487
Summe	47.450	0.12224	0.17174

We also evaluated the performance when frame importance scores were computed using the normalized inverse distance from the closest cluster center, while still training with both contrastive and Wasserstein losses:

Table 8: Joint Optimization framework results on benchmark datasets when using the pretrained transformer weights and computing scores using OT framework

Dataset	Transformer encoder weights (pretrained) trainable (scores using ot)			Transformer encoder weights (pretrained) frozen (scores using ot)		
	F-score	Kendall's τ	Spearman's ϱ	F-score	Kendall's τ	Spearman's ϱ
TVSum	54.468	0.07256	0.09547	55.724	0.00628	0.00834
Summe	48.388	-0.01948	-0.02740	39.869	-0.02951	-0.03886

Finally, we assessed the case where only the Wasserstein loss was used during training, and the importance scores were again based on the distance from the nearest cluster center:

Table 9: Optimizing using the optimal transport objective only results on benchmark datasets when using the pretrained transformer weights and computing scores using OT framework

Dataset	Transformer encoder (non-pretrained) weights trainable (scores using ot)			Transformer encoder (pretrained) weights frozen (scores using ot)		
	F-score	Kendall's τ	Spearman's ϱ	F-score	Kendall's τ	Spearman's ϱ
TVSum	55.365	0.02812	0.03652	55.724	0.00678	0.00901
Summe	49.975	0.06939	0.09317	39.869	-0.02948	-0.03889

Observations

- The first set of results clearly shows that combining the Optimal Transport objective with contrastive learning yields improved performance over the Transformer-based contrastive baseline. Notably, we observe gains in F-score across both datasets, and improved rank correlation coefficients on SumMe, indicating better alignment with ground-truth importance scores.
- However, when frame scores are derived from the normalized inverse distance to the nearest cluster center—despite using both contrastive and Wasserstein losses—the performance deteriorates. This suggests that contrastive loss is better suited for scoring than distance-based measures in this setup.
- Furthermore, when relying solely on the Wasserstein loss (without contrastive supervision), using cluster-distance-based scores does not offer significant improvements. In fact, freezing the Transformer encoder while using this approach significantly degrades performance, especially on the SumMe dataset.

As a second approach, rather than applying contrastive learning followed by optimal transport in a sequential manner, we explored training both objectives jointly from scratch.

Implementation details

The input features, padded or trimmed to a sequence length of 256, are first passed through a Transformer encoder comprising 4 layers and 8 attention heads. The encoded representations are then used to learn cluster centers via the Optimal Transport training objective. The learning signal from the Wasserstein loss is backpropagated through the encoder, reinforcing the quality of the learned frame-level representations and encouraging structural consistency.

Results

The results obtained on the TVSum and SumMe datasets are given below:

Table 10: Joint Optimization framework jointly training from scratch results on benchmark datasets when using the pretrained transformer weights and computing scores using OT framework

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	59.474	0.13324	0.17535
Summe	48.402	0.17274	0.23106

Observations

- The results indicate that this joint training approach outperforms the contrastive-only baseline on the SumMe dataset and remains competitive on the TVSum dataset. The additional supervision provided by the Optimal Transport objective appears to enhance the model's ability to learn structurally aware and semantically richer representations.
- Moreover, this method surpasses the variant where the Transformer encoder was pretrained with contrastive learning before jointly optimizing both losses. **This suggests that training both objectives jointly from scratch leads to more effective representation learning**.

4.3.6 Missing Frame Reconstruction Using TCNs

Implementation details

The TCN-scorer was implemented using a 9-layer dilated Temporal Convolutional Network (TCN), with each layer employing a kernel size of 2 and 1024 hidden channels. A multilayer perceptron (MLP) followed the TCN to predict frame-wise importance scores. The input sequence length was capped at 512 frames, and only features corresponding to odd-indexed frames were fed into the TCN-scorer.

The predicted scores were applied to the input features via element-wise multiplication. These weighted features were then passed to a TCN-based encoder-decoder reconstruction module, configured with a hidden dimensionality of 2048. The reconstruction objective consisted of two components: a mean squared error (MSE) loss applied only to the even-indexed frames, and a Frobenius loss computed over the entire sequence to preserve global structural consistency.

The model was trained for 200 epochs with a batch size of 2.

Results

The results obtained on the TVSum and the Summe datasets are as follows:

Table 11: TCN-based missing frame reconstruction framework results on benchmark datasets

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	56.090	0.04022	0.05306
Summe	42.461	0.04537	0.06615

Observations

- While the rank correlation coefficients are relatively low, the competitive F-scores demonstrate the model's capability to reconstruct key frames and maintain temporal coherence, even when trained on sparsely sampled input sequences. This indicates the potential of TCN-based reconstruction for learning frame importance in unsupervised settings.

4.3.7 Neighbor Consistency and Structural Preservation using TCN architecture

Implementation details

We implemented a TCN-based scorer consisting of a TCN encoder with 9 dilated convolutional layers, each configured with a kernel size of 2 and 1024 channels. This encoder was followed by a multilayer perceptron (MLP) for frame-wise importance score prediction. The input sequence was limited to 256 frames.

The input features were first passed through the TCN-scorer to obtain importance scores. We then selected the top-kkk frames based on these scores (k=38, i.e., 15% of the sequence length). For each selected key frame, we identified n=7 nearest non-key-frame neighbors based on feature similarity. The scaled encoded features (scaled by their predicted importance scores) were then compared to their corresponding neighbors using Euclidean distance. These distances were assembled into a matrix, and the Frobenius norm of this matrix was used as a regularization term.

Training was further guided by an additional Frobenius loss computed between the original input features and the encoded frame features. The model was trained for 200 epochs using a batch size of 2.

Results

Table 12: TCN-based neighbor consistency and structural preservation framework results on benchmark datasets

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	59.427	0.05691	0.07507
Summe	37.882	0.08249	0.11635

Observations

- Although this method does not outperform the Transformer-based contrastive baseline, it shows promising results. The incorporation of neighbor consistency and structural preservation offers a viable alternative, especially considering the relatively strong F-score on the TVSum dataset. And this method performs better than the missing frames method.

4.3.8 Contrastive Learning with TCN Encoder

Implementation details

We implemented a TCN encoder composed of six stacked TCN blocks, each containing three layers with dilation rates of 1, 2, and 4, respectively. A kernel size of 2 was used throughout. The input video sequences were fixed to a length of 256 frames.

To reduce the feature dimensionality, a linear projection layer was used to compress the input features to 128 dimensions. The model was trained using contrastive objectives—specifically, alignment and uniformity losses—for 200 epochs.

Results

The results obtained on the TVSum and the Summe datasets are as follows:

Table 13: TCN-based contrastive framework results on benchmark datasets

Dataset	F-score	Kendall's τ	Spearman's ϱ
TVSum	57.707	0.13438	0.17782
Summe	46.373	0.22203	0.29807

Observations

- On the SumMe dataset, the method achieved rank correlation coefficients that surpassed those of the Transformer-based contrastive baseline, indicating stronger agreement with ground-truth importance rankings. On the TVSum dataset, the performance was competitive, demonstrating that the TCN encoder is also effective at modeling inter-frame temporal dependencies.

Finally we compared our methods with previous unsupervised and even supervised approaches. The results are given below:

Table 14: Performance Comparison with previous unsupervised approaches

Method	TVSum			Summe		
	F-score	τ	ϱ	F-score	τ	ϱ
DR-DSN$_{60}$ [20]	57.6	0.0169	0.0227	41.4	0.0433	0.0501
SUM-FCN$_{unsup}$[3]	52.7	0.0107	0.0142	41.5	0.0080	0.0096
SUM-GAN [2]	51.7	-0.0535	-0.0701	39.1	-0.0095	-0.0122
UnpairedVSN [4]	55.6	-	-	47.5	-	-
Contrastive transformer baseline (recreated)	59.620	0.13920	0.18456	47.026	0.11549	0.16327
OT(ours)	**62.178**	**0.14891**	**0.19605**	44.784	0.05644	0.07674
Contrastive then OT (ours)	60.063	0.13220	0.17487	47.450	0.12224	0.17174
Joint Contrastive + OT (ours)	59.474	0.13324	0.17535	**48.402**	**0.17274**	**0.23106**

Table 15: Performance Comparison with previous supervised approaches

Method	TVSum			Summe		
	F-score	τ	ϱ	F-score	τ	ϱ
VASNet [5]	61.42	0.1690	0.2221	**49.71**	0.0224	0.0255
dppLSTM [1]	54.7	0.0298	0.0385	38.6	-0.0256	-0.0311
SumGraph [21]	63.9	0.094	0.138	51.4	-0.0095	-0.0122
Multi-ranker [22]	-	**0.1758**	**0.2301**	-	0.0108	0.0137
Contrastive transformer baseline (recreated)	59.620	0.13920	0.18456	47.026	0.11549	0.16327
OT(ours)	**62.178**	0.14891	0.19605	44.784	0.05644	0.07674
Contrastive then OT (ours)	60.063	0.13220	0.17487	47.450	0.12224	0.17174
Joint Contrastive + OT (ours)	59.474	0.13324	0.17535	48.402	**0.17274**	**0.23106**

4.4 Analysis and Discussion

The results confirm that the Transformer-based contrastive learning baseline is a strong and consistent performer across both TVSum and SumMe datasets.

While the Hierarchical Transformer showed slightly lower performance, likely due to the short video lengths in these datasets, it holds promise for longer videos with more complex temporal structure.

The proposed Optimal Transport-based clustering approach significantly outperforms the K-Means baseline and even surpasses the Transformer-based contrastive model on TVSum, demonstrating its ability to capture representative visual semantics through learnable cluster centers.

Joint training with both Optimal Transport and contrastive objectives further improves performance, particularly on SumMe, where it exceeds the Transformer baseline—highlighting the benefits of combined local and global supervision.

Additionally, TCN-based methods showed competitive results, with the contrastive TCN variant achieving the highest rank correlation on SumMe, indicating strong temporal modeling capabilities.

Overall, our approaches consistently outperform previous unsupervised methods and rival supervised baselines, underscoring the effectiveness of our framework for unsupervised video summarization.

Conclusion & Future Work

5.1 Summary of Findings

In this work, we explored multiple unsupervised frameworks for video summarization, with a focus on learning frame importance without access to ground-truth annotations. Our contributions included contrastive learning-based approaches using both Transformer and TCN encoders, clustering-based scoring techniques using K-Means and Optimal Transport, and reconstruction-based methods that encouraged temporal structure and relational consistency.

Among our key findings:

- The **Transformer-based contrastive baseline** proved to be a strong and consistent performer across both TVSum and SumMe datasets, establishing a reliable reference for subsequent approaches.

- The **Hierarchical Transformer** offered a more structured representation but showed limited advantage on shorter videos; we believe it holds promise for longer, more complex video content.

- The **Optimal Transport-based clustering method** significantly outperformed the K-Means baseline and even surpassed the contrastive baseline on the TVSum dataset.

- **Joint training** using both **contrastive and Optimal Transport objectives** led to further improvements, particularly on the SumMe dataset, demonstrating the benefit of combining local alignment with global structural guidance.
- TCN-based methods showed competitive results, with the contrastive TCN variant achieving the highest rank correlation on SumMe, indicating strong temporal modeling capabilities.

- Our methods achieved **competitive performance with existing supervised methods**, despite being entirely unsupervised, and outperformed previous unsupervised baselines in most cases.

5.2 Limitations

While the proposed methods demonstrate strong performance, they are not without limitations:

- The datasets used (TVSum and SumMe) consist of relatively short and curated videos, which may not fully reflect the diversity and complexity of real-world video content. For long videos one can explore our hierarchical transformer based contrastive framework.
- Frame scoring using nearest cluster distances was found to be less effective compared to contrastive losses when jointly optimizing the model using both the contrastive and optimal transport based objectives, suggesting limitations in relying solely on distance metrics for importance estimation.

5.3 Suggestions for Future Work

This study opens several avenues for further exploration:

- **Extension to longer and more diverse videos**: Future work can explore the scalability of hierarchical and clustering-based models on longer and untrimmed videos from real-world datasets.
- **Incorporation of multimodal cues**: Future research can explore the integration of audio, text, and metadata modalities alongside visual information. This multimodal approach could provide richer context for frame scoring and enable more semantically meaningful summaries, especially in complex or narrative-driven videos.
- **Real-time summarization**: Exploring lightweight versions of the proposed models for deployment in edge environments (e.g., surveillance or mobile video summarization) could be a practical next step.

References

[1] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In ECCV, 2016.

[2] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial LSTM networks. In CVPR, 2017.

[3] Mrigank Rochan, Linwei Ye, and Yang Wang. Video summarization using fully convolutional sequence networks. In ECCV, 2018.

[4] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In CVPR, 2019.

[5] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In ACCV, 2018.

[6] E. Apostolidis, G. Balaouras, V. Mezaris, and I. Patras, “Summarizing Videos using Concentrated Attention and Considering the Uniqueness and Diversity of the Video Frames,” in Proceedings of the 2022 International Conference on Multimedia Retrieval, Newark NJ USA: ACM, Jun. 2022, pp. 407–415. doi: 10.1145/3512527.3531404.

[7] Zong shang Pang, Yuta Nakashima, Mayu Otani, and Hajime Nagahara. Contrastive Losses Are Natural Criteria for Unsupervised Video Summarization. In Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), 2023.

[8] Y. Wang, H. Xu, and D. Luo, “Self-supervised Video Summarization Guided by Semantic Inverse Optimal Transport,” in Proceedings of the 31st ACM International Conference on Multimedia, Ottawa ON Canada: ACM, Oct. 2023, pp. 6611–6622. doi: 10.1145/3581783.3612087.

[9] CLIP-It! Language-Guided Video Summarization Medhini Narasimhan Anna Rohrbach Trevor Darrell University of California, Berkeley

[10] “Progressive video summarisation.pdf.” Accessed: Apr. 01, 2023. [Online]. Available: https://arxiv.org/pdf/2201.02494.pdf

[11] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In CVPR, 2015.

[12] Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. 2022. Prompt Learning with Optimal Transport for Vision-Language Models. arXiv preprint arXiv:2210.01253 (2022).

[13] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In European conference on computer vision. Springer, 104–120.

[14] An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling Shaojie Bai 1 J. Zico Kolter 2 Vladlen Koltun 3

[15] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. TVSum: Summarizing web videos using titles. In CVPR, 2015.

[16] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos. In ECCV, 2014.

[17] Maurice G Kendall. The treatment of ties in ranking problems. Biometrika, 1945.

[18] William H Beyer. Standard Probability and Statistics: Tables and Formulae. 1991.

[19] Mayu Otani, Yuta Nakashima, Esa Rahtu, and Janne Heikkila. Rethinking the evaluation of video summaries. In CVPR, 2019.

[20] Kaiyang Zhou, Yu Qiao, and Tao Xiang. Deep reinforcement learning for unsupervised video summarization with diversity-representativeness reward. In AAAI, 2018.

[21] Jungin Park, Jiyoung Lee, Ig-Jae Kim, and Kwanghoon Sohn. SumGraph: Video summarization via recursive graph modeling. In ECCV, 2020.

[22] Yassir Saquil, Da Chen, Yuan He, Chuan Li, and Yong-Liang Yang. Multiple pairwise ranking networks for personalized video summarization. In ICCV, 2021.