SummDiff: Leveraging Generative Diffusion to Video Summarization

Anonymous CVPR submission

Paper ID *****

Abstract

In this paper, we propose the first model that introduces a diffusion model for the task of video summarization. The goal of video summarization is to distill a video into a compact representation that preserves its essential elements and key moments, thereby reducing its overall length. Unlike previous methods, our approach draws inspiration from the human-like process of iteratively refining video content by repeatedly viewing and condensing key segments. To emulate this, we introduce SummDiff. SummDiff employs a Video Importance Score Denoiser to iteratively distinguish genuine importance scores from random noise. By progressively refining the score distribution through a generative diffusion model, our model dynamically adapts to visual contexts, enhancing the robustness and scalability of video summarization. Extensive evaluations demonstrate that SummDiff achieves state-of-the-art performance on benchmark datasets, including Mr.HiSum, TVSum, and SumMe. These results highlight the potential of diffusion-based approaches to transform the field of video summarization.

1. Introduction

In the fast-paced digital era, the exponential growth of video content across platforms such as YouTube, Instagram, and TikTok has highlighted the necessity for efficient content management and retrieval systems. Moreover, the market has seen a shift from traditional long-form videos to short-form videos, reflecting changes in viewer preferences and the rising popularity of quick, digestible content. Video summarization, the process of compressing the most significant content from a video into a concise summary, addresses this need by allowing users to grasp the essence of videos quickly without consuming the entire content. This technique not only enhances user engagement by providing tailored content highlights but also serves crucial roles in areas like surveillance and educational content curation, where quick information retrieval is important.

The process of summarizing a video by a human is as follows: Initially, one watches the video from start to finish, filtering out the segments to be included in the summary. Starting with the most important parts, subsequent segments of lesser importance are added. This involves a repetitive process of gauging the relative importance of different parts of the video and incorporating them into the summary until the given video length is filled. In essence, it is a process of continuously determining the relative importance of segments and summarizing accordingly. Previous research on video summarization [2, 10, 22, 50, 56] attempted to directly learn the relationship between a video and its summary for a single model inference when a video is presented. However, due to the limitation of this single direct inference, the performance was often found to be insufficient. It can be said that adjusting video summaries through relative comparisons of video segments over several steps, similar to human video summarization, constitutes the conditions for creating good video summaries.

A diffusion model [14, 42] is a model that learns the process of denoising from complete noise to a specific original distribution. There are similarities between the diffusion models and the process of composing a video summary, starting without any information when first viewing the video, and then repeatedly watching to gauge the relative importance. Through the process of denoising multiple times during inference, a more accurate video summary can be constructed. Moreover, when defining the relationship function between a given video and its summary as $p(\theta)$, progressively fitting the derivative function $p'(\theta)$ rather than directly modeling $p(\theta)$ surpasses the limitations of previous models and is a method suitable for the characteristics of this task. It is shown in our experiment result that ours outperform existing baselines across all metrics. Therefore, we propose a video summarization method utilizing the diffusion model.

In conclusion, we introduce an innovative approach to video summarization through the incorporation of diffusion models, inspired by their capacity for denoising and their potential to closely mimic the iterative human process of summarizing video content. By drawing parallels between the diffusion process and the methodical way humans assess and incorporate the relative importance of video segments, this work highlights the advantages of a model that iteratively refines its summaries. Our method, which progressively fits a derivative function to model the relationship between a video and its summary, represents a significant departure from traditional single-step inference approaches, offering enhanced performance and a more nuanced understanding of video content. The empirical results affirm that our approach not only surpasses existing baselines across all metrics but also captures the essence of effective video summarization, which lies in the ability to distill and convey the most pertinent information from extensive video data.

Contributions. The contributions of this work are manifold and significant, delineated as follows:

Innovative Application of Diffusion Models: We introduce a novel utilization of diffusion models in video summarization, drawing inspiration from their denoising capabilities and the iterative nature of human summarization processes.

Progressive Fitting Method: Our method, which involves progressively fitting a derivative function to model the relationship between a video and its summary, marks a significant departure from traditional approaches, offering enhanced performance and deeper insights into video content.

Superior Performance: The empirical evidence showcases that our approach not only outstrips existing benchmarks across all metrics but also encapsulates the essence of effective video summarization.

2. Related Work

Video Summarization is task where the model focuses on finding important information or sequences from the given video. Early works in video summarization use unsupervised learning as their techniques where models select video frames by predefined heuristics [6, 8, 18, 24, 27–31, 45, 51]. They mainly focus on importance or diversity of the frame from the video. More recent works are done by supervised learning where the categorized video is given by manual tagging [4, 24, 34–36].

Supervised learning models can easily learn the high-level features than unsupervised learning based models, so they have more accuracy on the task. DSNet[56] creates temporal interest proposals to identify and pinpoint the representative content within video sequences. iPTNet[22] employs cross-task sample transfer by designing an importance propagation module, enabling the conversion between summarization-guided and localization-guided importance maps. SL-module[50] applies unsupervised domain adaptation technique to video highlight-generation, which can also be applied to summarization task.

As the application of deep learning to the video summarization, RNN models such as LSTM is popular approach. Zhang et al. [52] proposed first summarization model using LSTM to model both short-range and long-range dependencies. More models were suggested using recurrent models as they are appropriate for predicting importance of video frames over time [53–55]. Moreover, using attention layer for modeling the users' interest on time has been suggested for improving the video summarization model. VASNet[10] proposed the first approach for using attention, and PGL-SUM[2] proposed combining local attention with global attention. More recent studies suggest on using transformer models by leveraging their ability to model long-range dependencies and complex patterns within video sequences [3, 20, 21]. A2Summ [12] suggests the use of aligning and attending the multi-modal transformer based on both text and video inputs, using the dual contrastive loss.

Diffusion models [41] are initially adds Gaussian noise to data over several steps, gradually transforming the data into a pure noise distribution [14, 43]. Diffusion models have emerged as a groundbreaking tool in the field of generative such as image generation [14, 32, 38, 43, 44] or image super-resolution [7, 23, 37, 39]. Due to their outstanding performance, the application of diffusion models to video tasks has risen as a recent main focus, primarily concentrating on video generation [9, 15, 16] or prediction [19]. More recent study by Li et al. [25] suggests applying diffusion model to video moment retrieval task when language description is given.

One of the most commonly used dataset for video summarization tasks is TVSum, which contains 50 videos with duration of 2 to 10 minutes [45]. SumMe is also popular dataset which is a collection of 25 videos from 1 to 6 minutes [11]. These two datasets have popularity of 20% and 19% each [40]. Other popular datasets are VSUMM and MED [8, 35]. The problem with these datasets is that the number of videos each dataset contains is too small due to the high cost of manual labeling, which makes them prone to overfitting. For these reasons, a larger dataset for video summarization has been suggested, named Mr. Hisum [46]. Mr.Hisum contains 31,892 videos labeled by 50,000+ users per video which makes video summarization task more robust.

In this study, we propose the use of a diffusion model for the task of video summarization, marking the first attempt to apply such a model in this context. Also by using the large scale dataset, Mr. Hisum for our model, we can train and test model more reliably.

3. Problem Formulation

Given a video comprising a total of N frames, the objective of video summarization is the identification and selection of $K \ll N$ frames that effectively encapsulate the essence of the video content. Let X denote the set containing all video frames, *i.e.* $X = \{X_1, \cdots X_N\}$, where X_t represents t-th frame. Each frame is associated with a binary

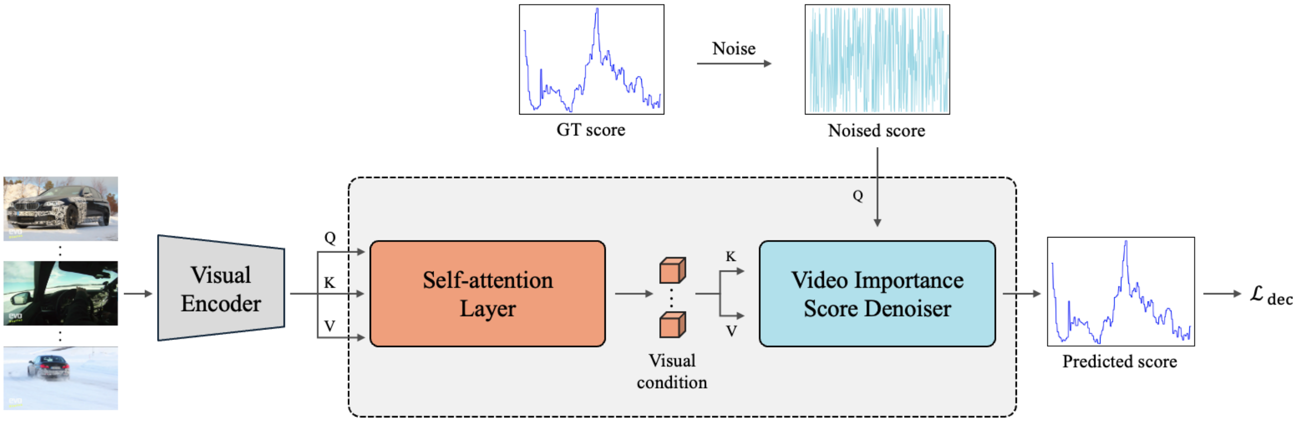

Figure 1. **Overview of SummDiff.** Given an input video, SummDiff generates importance scores conditioned on video frames.

label indicating its inclusion in the summary; thus, there exists a binary label Y, *i.e.* $Y = \{Y_1, \cdots, Y_N\}$, where $Y_t \in \{0, 1\}$. The predicted video summary can be represented as $\hat{Y} = \{\hat{Y_1}, \cdots, \hat{Y_N}\}$. To regulate the proportion of frames included in the summary, a specific threshold, $\rho = K/N$, is established for each dataset, ensuring $\sum_{t=1}^{N} Y_t \leq K$.

Most summarization datasets provide binary annotation from multiple annotators A, denoted as $Y_a \in \{0, 1\}^N$, *i.e.* $a = 1, \cdots A$. The average of these annotations is defined as the importance score, $S = \{S_1, \cdots, S_N\}$. Consequently, the majority of video summarization models, including ours, tackle this as a regression problem aiming to predict these important scores accurately.

For the evaluation process, we adopt a widely-used evaluation scheme by [52]. Specifically, we aggregate predicted frame importance scores $\hat{S} = \{\hat{S_1}, \cdots, \hat{S_N}\}$ by averaging the scores within each shot to form $\hat{S}^* = \{\hat{S}_1^*, \cdots, \hat{S}_n^*\}$ utilizing the boundary information by KTS [35]. Then, solving 0/1 knapsack problem with dynamic programming [45] to maximize the selected scores within a given budget (e.g., 15% of the original video length) constructs a video summary. F1 score is widely used to evaluate the selected video summary.

4. Method

In this section, we introduce the SummDiff model along with a comprehensive overview of the training and inference processes. Overall illustration of SummDiff is outlined in Fig 1. SummDiff is a model designed to adapt the distribution of importance scores for a given video by learning to denoise the noise distribution to ground truth importance score. Initially, we extract frame-level features and employ a transformer encoder [49] to obtain contextualized visual embeddings. These embeddings serve as visual conditions, enabling our **Video Importance Score Denoiser** to distinguish the genuine distribution of importance scores from the random noise distribution.

4.1. Learning Video Importance Score with Generative Diffusion

In this section, we first discuss the principles underlying the forward and reverse processes in diffusion models. Subsequently, we detail the construction of the diffusion generation process within the video importance score denoiser.

Forward. During training, we initially create a forward process that adds noise to ground truth importance scores $\boldsymbol{S}_0 \sim q(\boldsymbol{S}_0)$ into noisy data $\boldsymbol{S}_t$, where t represents the number of time steps. Specifically, the Gaussian noise process for any two consecutive intensities [14] is defined as: $q(\boldsymbol{S}_t \mid \boldsymbol{S}_{t-1}) = \mathcal{N}\left(\boldsymbol{S}_t; \sqrt{1-\beta_t}\boldsymbol{S}_{t-1}, \beta_t \mathrm{I}\right)$, with β being the variance schedule. Consequently, $\boldsymbol{S}_t$ is derived from $\boldsymbol{S}_0$ as follows: $q(\boldsymbol{S}_{1:t} \mid \boldsymbol{S}_0) = \prod_{i=1}^{t} q(\boldsymbol{S}_i \mid \boldsymbol{S}_{i-1})$. Leveraging the re-parameterization technique, we achieve noised importance score $\mathbf{S}_t$ with $\boldsymbol{S}_t = \sqrt{\bar{\alpha}_t}\boldsymbol{S}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}_t$, where $\boldsymbol{\epsilon}_t \sim \mathcal{N}(0, \mathrm{I})$, $\bar{\alpha}_t = \prod_{i=1}^{t}(1-\beta_i)$.

Reverse. The denoising process aims to progressively remove noise, transitioning from S_t to S_0. The conventional single-step approach to this process is captured by the equation: $p_\theta(\boldsymbol{S}_{t-1} \mid \boldsymbol{S}_t) = \mathcal{N}\left(\boldsymbol{S}_{t-1}; \boldsymbol{\mu}_\theta(\boldsymbol{S}_t, t), \sigma_t^2 \boldsymbol{I}\right)$

Here, σ_t^2 is determined in relation to β_m, and $\boldsymbol{\mu}_\theta(\boldsymbol{S}_t, t)$ represents the estimated mean. In our study, we focus on training the Video Importance Score Denoiser to invert this denoising trajectory. The key distinction lies in our approach: instead of estimating $\boldsymbol{\mu}_\theta(\boldsymbol{S}_t, t)$, we derive importance scores from the Video Importance Score Denoiser network through $f_\theta(\boldsymbol{S}_t, t, \boldsymbol{F})$.

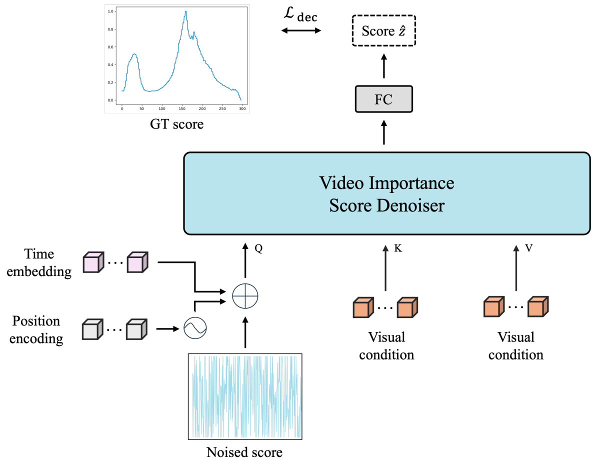

Figure 2. **Video Importance Score Denoiser.** Importance score is denoised conditioned on contextualized visual features.

4.2. Visual Condition: Importance-aware visual representation

To obtain embeddings that encapsulate importance-aware visual information, we first process all video frames $X = \{X_1, \cdots, X_N\}$ through pretrained image encoders to obtain a D-dimensional feature for each frame. These features are then passed through a Multi-Layer Perceptron (MLP) layer, resulting in a set of embeddings denoted as Z, *i.e.*, $Z = \{Z_1, \cdots, Z_N\}, Z \in \mathbb{R}^{N\times D}$. To contextualize these embeddings further, they are processed through Multi-Head Self Attention layers described in Vaswani *et al* [49]. Video embedding Z is projected as query Q_i, key K_i, and value V_i, where i stands for head index to apply self-attention as follows,

$$\begin{aligned}\text{Contextualized } Z : \hat{Z} &= \text{Concat}(H_1, \cdots, H_h)W \\ \text{where } H_i &= \text{softmax}(Q_i K_i^T / \sqrt{d_k})V\end{aligned} \tag{1}$$

4.3. Quantization of Importance Score.

To denoise the noised score via the Video Importance Score Denoiser, it is necessary to transform $S_t \in \mathbb{R}^N$ into a space of $\text{R}^{N\times D}$. This transformation is achieved through discretization, which involves dividing the score distribution into a predefined number of uniform segments. Each segment is assigned a learnable embedding of dimension D. Then, all scores are mapped according to this configuration, resulting in a dimensionality of $\mathbb{R}^{N\times D}$. Furthermore, this also allows the discretized learnable embedding to be adjusted according to visual conditions. This effect will be discussed further in the experimental results.

4.4. Video Importance Score Denoiser

Training. Initially, we map the ground-truth score values into the range $[-\lambda, \lambda]$ by applying the transformation $\boldsymbol{S}_0 = \lambda(2\boldsymbol{S}_0 - 1)$, which facilitates the incorporation of Gaussian noise. After the noise addition, we ensure that $\boldsymbol{S}_t$ remains within $[-\lambda, \lambda]$ by clamping. A reverse transformation is then utilized to scale $\boldsymbol{S}_t$ back to the original interval of $[0, 1]$, using the equation $\boldsymbol{S}_t = (\boldsymbol{S}_t/\lambda + 1)/2$, setting $\lambda = 1$. The noised scores are quantized into K intervals, with each interval linked to a distinct learnable embedding. To aid in denoising, the model is made aware of the noise addition moment, t, by integrating a sinusoidal time encoding, Emb_t, with $\boldsymbol{S}_t$. Moreover, acknowledging the sequential essence of video scores, positional embeddings, pos, are introduced via sinusoidal functions. To prevent any mixing of these two sinusoidal signals, their vectors are kept orthogonal. The comprehensive representation $Q_t = \boldsymbol{S}_t + Emb_t + pos$ is finally processed to produce the model output.

$$Q_t = \text{softmax}(Q_t K^T)V + Q_t \tag{2}$$

Ultimately, the transformer's output is converted into the predicted scores $\hat{\boldsymbol{S}}_{t-1}$ through a straightforward fully connected (FC) layer. In alignment with the findings of [5], our objective is to minimize the difference between the network's predictions and the ground-truth scores $\boldsymbol{S}_0$.

$$L_{dec}(\boldsymbol{S}_0, f_\theta(\boldsymbol{S}_t, t, F)) = ||\boldsymbol{S}_0 - \hat{\boldsymbol{S}}_{t-1}||_2^2 \tag{3}$$

Inference. After training, our SummDiff model can generate video importance scores for video summarization by initiating with randomly sampled noise $\boldsymbol{S}_t$ from a Gaussian distribution $N(0, I)$. Leveraging the principles of diffusion models [42], the model iteratively refines these scores towards cleaner estimations.

$$\begin{aligned}\hat{\boldsymbol{S}}_{t-1} = &\sqrt{1 - \bar{\alpha}_{t-1} - \sigma_t^2}\,\frac{\hat{\boldsymbol{S}}_t - \sqrt{\bar{\alpha}_m} f_\theta(\boldsymbol{S}_t, t, F))}{\sqrt{1 - \bar{\alpha}_m}} \\ &+ \sqrt{\bar{\alpha}_{t-1}} f_\theta(\boldsymbol{S}_t, t, F)) + \sigma_t \epsilon_t.\end{aligned} \tag{4}$$

The overall architecture of the importance score denoiser are detailed in Fig 2. This framework is instrumental in progressively refining the video importance scores. Notably, in the final step, $f_\theta(\hat{\boldsymbol{S}}_1, 1, F)$ is directly employed to estimate $\hat{\boldsymbol{S}}_0$.

5. Experiments

5.1. Datasets and Evaluation Metrics

Dataests. We evaluate our approach, along with the state-of-the-art video summarization models [2, 10, 12, 22, 50, 56], on established benchmarks such as Mr.HiSum [46], TVSum [45], and SumMe [11]. We use Inception-v3 [47] features PCA-ed to 1024D, following YouTube-8M [1]. To measure the zero-shot performance on TVSum [45] and SumMe [11] when pretrained on Mr.HiSum [46], idenitcal features are extracted from all of two other datasets.

Evalutaion Metrics. To measure video summarization performance, we adopt a widely-used evaluation scheme proposed by [52]. Specifically, we aggregate predicted frame importance scores by averaging the scores within each shot, utilizing the boundary information provided by the KTS algorithm [35]. Then, we solve 0/1 knapsack problem using dynamic programming [45] to maximize the selected scores within a given budget (e.g., 15% of the original video length), which constructs the video summary. The F1 score is widely used to evaluate the selected video summary.

Furthermore, in line with [46], we also evaluate our method on the video highlight detection task. First, we uniformly divide the input video into 5-second-long shots and calculated the average frame scores for each shot. The top $\rho \in \{15\%, 50\%\}$ of these shots are designated as ground truth highlights, following previous works [17, 33, 52]. Mean Average Precision (MAP) is used to measure the performance. This evaluation metric differs from the summarization f1 score in that it divides each segment uniformly and greedily includes each segment into the highlights. We demonstrate that our model consistently outperforms other baselines across all of these different metrics.

5.2. Implementation Details

For input video representation, we downsample the videos to a uniform frame rate of one frame per second (1 fps). Our model's architecture employs two transformer encoder layers for visual encoding and an additional two transformer layers focusing on denoising video importance scores. Each of these transformer layers has a hidden size of 256, 8 attention heads, and feed-forward network with a dimensionality of 1024. To optimize our model, we use the AdamW optimizer [26], incorporating a cosine annealing strategy for the learning rate [48]. This strategy gradually reduces the learning rate from an initial value of 5e-5. Our training process utilizes a batch size of 256, runs for 200 epochs and is executed on a single NVIDIA A4000 GPU.

5.3. Experimental Results

Comparison with State-of-the-Art models. We follow Mr.HiSum [46], to split train, validation and test set to train our model and other existing video summarization methods on the train set and select the best performing model on the validation set. The performance on the test set are summarized in Table 1. For A2Summ [12], which integrates text information, the text part of the model is removed for fair comparison.

Model	F1 Score	MAP$_{\rho=50\%}$	MAP$_{\rho=15\%}$
SummDiff (Ours)	**57.71**$\pm$**0.005**	**65.28**$\pm$**0.007**	**33.67**$\pm$**0.03**
PGL-SUM [2]	55.89$\pm$0.04	61.60$\pm$0.14	27.45$\pm$0.15
VASNet [10]	55.26$\pm$0.05	58.69$\pm$0.30	25.28$\pm$0.40
SL-module [50]	55.31$\pm$0.09	58.63$\pm$0.13	24.95$\pm$0.13
A2Summ [12]	51.87$\pm$0.16	59.18$\pm$0.13	30.70$\pm$0.21
DSNet [56]	50.78$\pm$0.16	57.31$\pm$0.18	24.35$\pm$0.34
iPTNet [22]	50.53$\pm$0.16	55.53$\pm$0.25	22.74$\pm$0.13

Table 1. Performance on Mr.HiSum dataset

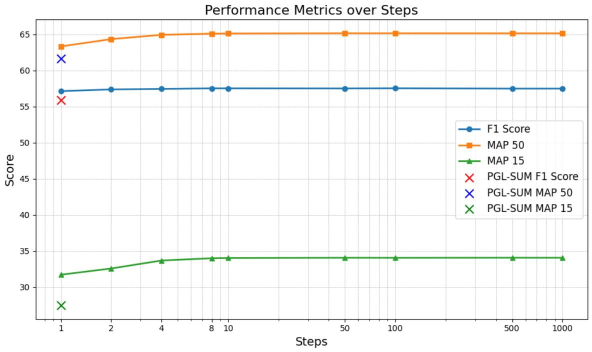

Figure 3. Performance metrics over DDIM Steps.

As you can see clearly from the table, PGL-SUM [2], VASNET [10], SL-module [50], and our model, SummDiff, consistently outshine other video summarization techniques in terms of F1-score. Particularly, SummDiff demonstrates superior performance across various metrics when compared with leading models. For example, it surpasses PGL-SUM by 1.82% in F1-score, 3.68% in MAP$_{\rho=50\%}$, and 6.22% in MAP$_{\rho=15\%}$. Unlike prior models that estimate importance scores in a straightforward manner from the video input, SummDiff adopts a novel approach by modeling the flow of the functional relationship between the video content and its importance scores, resulting in more precise video summaries. Crucially, our approach departs from conventional methods that generate importance scores for the entire video at once. Instead, SummDiff employs a generative technique that allows for conditional creation. It crafts each segment of the summary by considering the current video segment and refining based on the previously denoised score. This methodological innovation ensures that each portion of the summary is contextually coherent with the preceding content, significantly enhancing the quality of the summary, as demonstrated by the outstanding results reported in Table 1.

Performance over DDIM Steps. In figure 3 and table 2, we explore the effects of different DDIM [42] steps. 1 step refers to directly going from a complete noise distribution to the predicted importance score. Both the figure 3 and table

Model	SummDiff (Step 1)	PGL-SUM
F1 Score	**57.12**	55.89
MAP$_{\rho=50\%}$	**63.32**	61.60
MAP$_{\rho=15\%}$	**31.69**	27.45

Table 2. Performance comparison at step 1.

Embedding Type	F1 score	MAP$_{\rho=50\%}$	MAP$_{\rho=15\%}$
Fixed Uniform Rand	56.95	64.85	33.50
Fixed Fourier	57.27	65.24	33.02
Learnable	**57.71**	**65.28**	**33.65**

Table 4. Embedding Type Performance Comparison

Datasets		Models		
Training	Test	SummDiff	VASNet	PGL-SUM
TVSum	TVSum	**56.7**	54.2 ±0.9	55.5 ±0.7
Mr.HiSum	TVSum	**57.6**	57.1 ±1.0	57.1 ±0.7
SumMe	SumMe	**42.4**	41.9 ±3.3	41.7 ±3.2
Mr.HiSum	SumMe	**41.8**	42.6 ±1.3	42.3 ±2.1

Table 3. Comparison of training performance from scratch and zero-shot transfer performance when pretrained on Mr.HiSum [46], evaluated on the TVSum [45] and SumMe [11] datasets.

K	F1 Score	MAP$_{\rho=50\%}$	MAP$_{\rho=15\%}$
5	57.31	64.94	32.22
10	57.23	64.58	32.30
50	57.49	64.54	32.39
100	57.60	65.20	33.08
200	**57.83**	64.78	32.91
400	57.76	**65.34**	32.66
800	57.20	64.93	33.01
1600	57.47	65.15	**33.21**

Table 5. Performance Metrics for Different Values of K

Classifier Free Guidance [13]							
w	0.1	0.2	0.3	0.4	0.8	1.6	3.2
$p = 0$	57.51						
$p = 0.1$	57.31	57.67	57.67	57.05	57.34	56.80	57.31
$p = 0.2$	**57.71**	57.59	57.08	57.24	57.25	57.33	57.20

Table 6. CFG: Classifier Free Guidance F1 score over p and w.

2 suggest that SummDiff (Ours) outperforms PGL-SUM[2], the best performing model among pre-existing video summarization models, even with DDIM [42] step 1 across all of the metrics. This result also supports our claim that modeling the flow of the function between video input and the corresponding importance score is superior to direct modeling like other models. We can also observe that increasing step size leads to better model performance showing that the iterative generation process refines the video importance score at each step and produces a better summary.

Comparison on Traditional Benchmarks. To validate the effectiveness of our model on additional video summarization datasets, we present its performance on TVSum [45] and SumMe [11]. The first and third columns of table 3 display the performance metrics when each model is trained from scratch on the respective dataset. Due to the limited size of the dataset, we employed 5-fold cross-validation. This approach allowed us to set aside a test set while using a validation set from each training fold. The average performance across all 5 folds is reported to ensure a robust evaluation of our model. Across both datasets, SummDiff outperforms the two leading baselines, PGL-SUM [2] and VASNet [10], by a margin of 1.2% for TVSum and 0.5% for SumMe, demonstrating superior capabilities. The second and fourth columns illustrate the zero-shot prediction competency when pretrained on the Mr.HiSum [46] dataset and subsequently evaluated on the target datasets. In every evaluation setting across all datasets, SummDiff consistently showcases enhanced performance.

6. Ablation

Embedding Type. To analyze the effect of learnable vectors after quantization as described in section 4.3, we experimented with fixed uniform random vectors and fixed Fourier vectors for each quantized score. For the fixed Fourier embedding, we assigned vertical embeddings along with positional embeddings to ensure that each signal remains distinct during training. As illustrated in Table 3, learnable embeddings outperform the other two options. This superiority can be attributed to the learnable embeddings' capacity to adapt to visual conditions during the training of the importance score denoiser.

Quantization Analysis in SummDiff. The quantization process plays a pivotal role in our SummDiff model, facilitating the transformation of scalar scores into embeddings for processing by the attention layer. In table 5, we explored a range of K values, which decide the number of segments into which the score values ranging from 0 to 1 are divided. Each segment is associated with a learnable embedding. Contrary to expectations, setting $K = 5$, thereby dividing the score range into 5 uniform segments, yielded satisfactory performance. This outcome highlights a unique aspect of video summarization—its reliance on relative rather than absolute score precision. Incremental increases in the value of K led to marginal improvements in model accuracy, as finer segmentation allowed for more precise score predictions. However, the performance enhancement plateaued and subsequently declined beyond $K = 200$, suggesting that the model struggles to learn from more than 200 score segments effectively.

Classifier Free Guidance [13]. In our experiments, we explored different values for the hyperparameters p and w to optimize our model. The parameter p represents the probability of an unconditioned sample, where we replace the visual condition with a null video (a completely black video) that undergoes the same feature extraction process as other videos. The parameter w determines the extent to which unconditioned information is used during inference.

The inference update is performed using the following formula by [13]:

$$\tilde{\epsilon}_\theta(\boldsymbol{S}_t, c) = (1 + w)\epsilon_\theta(\boldsymbol{S}_t, c) - w\epsilon_\theta(\boldsymbol{S}_t),$$

where c denotes the condition, and $w\epsilon_\theta(\boldsymbol{S}_t)$ represents the unconditional score term.

Scale λ	0.5	1	2	4
F1 Score	57.28	57.71	57.02	55.73
MAP 50	64.31	65.28	64.14	62.00
MAP 15	32.02	33.67	31.52	29.10

Table 7. Performance metrics for different scale values of λ.

Scale Parameter λ. The scale parameter λ plays a crucial role in mapping the ground-truth score values into the range $[-\lambda, \lambda]$ as described in section 4.4. This parameter influences the extent of the tail information from the Gaussian distribution that is utilized. After the forward process, we apply clamping, making λ a significant factor for the model's performance. As demonstrated in Table 7, setting λ to 1 is effective for scaling the score values to the range $[-1, 1]$ during the training's forward process.

7. Conclusion

In this paper, we proposed SummDiff, the first model using the diffusion technique for the video summarization task inspired by the denoising nature of the human summarization process. Our model works by denoising the noisy score based on contextualized visual features with a generative diffusion technique. Additionally, our model incrementally fits a derivative function to model the relationship between a video and its summary, representing a substantial advancement over traditional methods. These techniques not only improve performance but also provide a more profound understanding of video content. As a result, our model outperforms state-of-the-art video summarization models, trained not only on the large-sized Mr.HiSum dataset but also on the traditional TVSum and SumMe datasets.

References

[1] Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *arXiv preprint arXiv:1609.08675*, 2016. 5

[2] Evlampios Apostolidis, Georgios Balaouras, Vasileios Mezaris, and Ioannis Patras. Combining global and local attention with positional encoding for video summarization. In *2021 IEEE international symposium on multimedia (ISM)*, pages 226–234. IEEE, 2021. 1, 2, 4, 5, 6

[3] Manjot Bilkhu, Siyang Wang, and Tushar Dobhal. Attention is all you need for videos: Self-attention based video summarization using universal transformers. *arXiv preprint arXiv:1906.02792*, 2019. 2

[4] Sijia Cai, Wangmeng Zuo, Larry S Davis, and Lei Zhang. Weakly-supervised video summarization using variational encoder-decoder and web prior. In *Proceedings of the European conference on computer vision (ECCV)*, pages 184–200, 2018. 2

[5] Ting Chen, Ruixiang Zhang, and Geoffrey Hinton. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022. 4

[6] Wen-Sheng Chu, Yale Song, and Alejandro Jaimes. Video co-summarization: Video summarization by visual co-occurrence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3584–3592, 2015. 2

[7] Hyungjin Chung, Byeongsu Sim, and Jong Chul Ye. Come-closer-diffuse-faster: Accelerating conditional diffusion models for inverse problems through stochastic contraction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12413–12422, 2022. 2

[8] Sandra Eliza Fontes De Avila, Ana Paula Brandao Lopes, Antonio da Luz Jr, and Arnaldo de Albuquerque Araújo. Vsumm: A mechanism designed to produce static video summaries and a novel evaluation method. *Pattern recognition letters*, 32(1):56–68, 2011. 2

[9] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 2

[10] Jiri Fajtl, Hajar Sadeghi Sokeh, Vasileios Argyriou, Dorothy Monekosso, and Paolo Remagnino. Summarizing videos with attention. In *Computer Vision–ACCV 2018 Workshops: 14th Asian Conference on Computer Vision, Perth, Australia, December 2–6, 2018, Revised Selected Papers 14*, pages 39–54. Springer, 2019. 1, 2, 4, 5, 6

[11] Michael Gygli, Helmut Grabner, Hayko Riemenschneider, and Luc Van Gool. Creating summaries from user videos.

In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VII 13*, pages 505–520. Springer, 2014. 2, 5, 6

[12] Bo He, Jun Wang, Jielin Qiu, Trung Bui, Abhinav Shrivastava, and Zhaowen Wang. Align and attend: Multimodal summarization with dual contrastive losses. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14867–14878, 2023. 2, 4, 5

[13] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 6, 7

[14] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 2, 3

[15] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2

[16] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in Neural Information Processing Systems*, 35:8633–8646, 2022. 2

[17] Fa-Ting Hong, Xuanteng Huang, Wei-Hong Li, and Wei-Shi Zheng. Mini-net: Multiple instance ranking network for video highlight detection. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIII 16*, pages 345–360. Springer, 2020. 5

[18] Richang Hong, Jinhui Tang, Hung-Khoon Tan, Shuicheng Yan, Chongwah Ngo, and Tat-Seng Chua. Event driven summarization for web videos. In *Proceedings of the first SIGMM workshop on Social media*, pages 43–48, 2009. 2

[19] Tobias Höppe, Arash Mehrjou, Stefan Bauer, Didrik Nielsen, and Andrea Dittadi. Diffusion models for video prediction and infilling. *arXiv preprint arXiv:2206.07696*, 2022. 2

[20] Tzu-Chun Hsu, Yi-Sheng Liao, and Chun-Rong Huang. Video summarization with spatiotemporal vision transformer. *IEEE Transactions on Image Processing*, 2023. 2

[21] Jia-Hong Huang, Luka Murn, Marta Mrak, and Marcel Worring. Gpt2mvs: Generative pre-trained transformer-2 for multi-modal video summarization. In *Proceedings of the 2021 International Conference on Multimedia Retrieval*, pages 580–589, 2021. 2

[22] Hao Jiang and Yadong Mu. Joint video summarization and moment localization by cross-task sample transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16388–16398, 2022. 1, 2, 4, 5

[23] Bahjat Kawar, Michael Elad, Stefano Ermon, and Jiaming Song. Denoising diffusion restoration models. *Advances in Neural Information Processing Systems*, 35:23593–23606, 2022. 2

[24] Aditya Khosla, Raffay Hamid, Chih-Jen Lin, and Neel Sundaresan. Large-scale video summarization using web-image priors. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2698–2705, 2013. 2

[25] Pandeng Li, Chen-Wei Xie, Hongtao Xie, Liming Zhao, Lei Zhang, Yun Zheng, Deli Zhao, and Yongdong Zhang. Momentdiff: Generative video moment retrieval from random to real. *Advances in neural information processing systems*, 36, 2024. 2

[26] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 5

[27] Zheng Lu and Kristen Grauman. Story-driven summarization for egocentric video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2714–2721, 2013. 2

[28] Yu-Fei Ma, Lie Lu, Hong-Jiang Zhang, and Mingjing Li. A user attention model for video summarization. In *Proceedings of the tenth ACM international conference on Multimedia*, pages 533–542, 2002.

[29] Behrooz Mahasseni, Michael Lam, and Sinisa Todorovic. Unsupervised video summarization with adversarial lstm networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 202–211, 2017.

[30] Padmavathi Mundur, Yong Rao, and Yelena Yesha. Keyframe-based video summarization using delaunay clustering. *International Journal on Digital Libraries*, 6:219–232, 2006.

[31] Chong-Wah Ngo, Yu-Fei Ma, and Hong-Jiang Zhang. Automatic video summarization by graph modeling. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 104–109. IEEE, 2003. 2

[32] Alex Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob McGrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. *arXiv preprint arXiv:2112.10741*, 2021. 2

[33] Rameswar Panda and Amit K Roy-Chowdhury. Collaborative summarization of topic-related videos. In *Proceedings of the IEEE Conference on computer vision and pattern recognition*, pages 7083–7092, 2017. 5

[34] Rameswar Panda, Abir Das, Ziyan Wu, Jan Ernst, and Amit K Roy-Chowdhury. Weakly supervised summarization of web videos. In *Proceedings of the IEEE international conference on computer vision*, pages 3657–3666, 2017. 2

[35] Danila Potapov, Matthijs Douze, Zaid Harchaoui, and Cordelia Schmid. Category-specific video summarization. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI 13*, pages 540–555. Springer, 2014. 2, 3, 5

[36] Mrigank Rochan and Yang Wang. Video summarization by learning from unpaired data. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7902–7911, 2019. 2

[37] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 2

[38] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans,

et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2

[39] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE transactions on pattern analysis and machine intelligence*, 45(4):4713–4726, 2022. 2

[40] Parul Saini, Krishan Kumar, Shamal Kashid, Ashray Saini, and Alok Negi. Video summarization using deep learning techniques: a detailed analysis and investigation. *Artificial Intelligence Review*, 56(11):12347–12385, 2023. 2

[41] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 2

[42] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 1, 4, 5, 6

[43] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019. 2

[44] Yang Song and Stefano Ermon. Improved techniques for training score-based generative models. *Advances in neural information processing systems*, 33:12438–12448, 2020. 2

[45] Yale Song, Jordi Vallmitjana, Amanda Stent, and Alejandro Jaimes. Tvsum: Summarizing web videos using titles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5179–5187, 2015. 2, 3, 5, 6

[46] Jinhwan Sul, Jihoon Han, and Joonseok Lee. Mr. hisum: A large-scale dataset for video highlight detection and summarization. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 4, 5, 6

[47] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016. 5

[48] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021. 5

[49] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 3, 4

[50] Minghao Xu, Hang Wang, Bingbing Ni, Riheng Zhu, Zhenbang Sun, and Changhu Wang. Cross-category video highlight detection via set-based learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7970–7979, 2021. 1, 2, 4, 5

[51] Hong Jiang Zhang, Jianhua Wu, Di Zhong, and Stephen W Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern recognition*, 30(4):643–658, 1997. 2

[52] Ke Zhang, Wei-Lun Chao, Fei Sha, and Kristen Grauman. Video summarization with long short-term memory. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, pages 766–782. Springer, 2016. 2, 3, 5

[53] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hierarchical recurrent neural network for video summarization. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 863–871, 2017. 2

[54] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Hsa-rnn: Hierarchical structure-adaptive rnn for video summarization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7405–7414, 2018.

[55] Bin Zhao, Xuelong Li, and Xiaoqiang Lu. Tth-rnn: Tensor-train hierarchical recurrent neural network for video summarization. *IEEE Transactions on Industrial Electronics*, 68(4): 3629–3637, 2020. 2

[56] Wencheng Zhu, Jiwen Lu, Jiahao Li, and Jie Zhou. Dsnet: A flexible detect-to-summarize network for video summarization. *IEEE Transactions on Image Processing*, 30:948–962, 2020. 1, 2, 4, 5