ITC 4/54
Information Technology and Control
Vol. 54 / No. 4/ 2025
pp. 1122-1138
DOI 10.5755/j01.itc.54.4.41063

Submodular Pseudo-3D Network for Micro-Video Summarization

Received 2025/04/02 | Accepted after revision 2025/06/28

HOW TO CITE: Xiaowei, G., Lu, L. (2025). Submodular Pseudo-3D Network for Micro-Video Summarization. *Information Technology and Control*, 54(3), 1122-1138. https://doi.org/10.5755/j01.itc.54.4.41063

Submodular Pseudo-3D Network for Micro-Video Summarization

Xiaowei Gu
School of Software Engineering; South China University of Technology; Guangzhou Higher Education Mega Centre, Panyu District, Guangzhou, China, 510006; e-mail: amyxwgu@163.com

Lu Lu
School of Computer Science & Engineering; South China University of Technology; Guangzhou Higher Education Mega Centre, Panyu District, Guangzhou, China, 510006; e-mail: lul@scut.edu.cn

Corresponding author: amyxwgu@163.com

With the explosion of micro-videos, it is essential to develop video summarization algorithms that simultaneously capture the diverse video content and represent the original video. Many methods employ deep neural networks (DNNs). However, DNN summarization models do not directly consider summary diversity. On the other hand, submodular functions can be seen as a form of diversity. However, the shallow structure prevents the data representation at a more abstract level. This paper proposes a novel submodular pseudo-3D network (SP3D), which equips submodular functions with a multi-layered network for micro-video summarization. Unlike standard DNNs, the proposed SP3D network and the corresponding optimization method consider the interlock dependency among the selected frames, thus improving the diversity. The experimental results indicate that the proposed model and optimization method are effective in micro-video summarization.

KEYWORDS: Video Summarization, Pseudo-3D Network, Deep Diversity Layers

1. Introduction

Micro-videos are becoming increasingly popular amongst all age groups. In 2023, the number of channels uploading YouTube Shorts year-on-year grew by 50% (https://abc.xyz/2024-q1-earnings-call). Reels has driven more than 40% increase in time spent on Instagram since its launch (https://inves-

tor.fb.com/investor-events). Different from professionally produced videos, typical short videos are captured casually by handheld mobile devices [36, 40]. They have diverse content and usually lack a predefined structure [17, 59]. With this explosion of micro-video data, it is essential to develop automatic video summarization algorithms to capture the diverse video content and represent the original video.

Diversity refers to capturing different aspects of the input video. In other words, it measures how dissimilar the selected frames are and removes the redundancy from a summary. Micro-video platforms, such as YouTube Shorts, can combine multiple video clips together. Semantic discontinuities may exist within video sequences, which makes it challenging to diversify the selected summary frames [41].

Many video summarization methods employ deep neural networks (DNNs) [58, 61, 62]. Compared to shallow structures, deep networks comprise multiple processing layers that can discover the representations needed for complex data [34]. It was stated in [1] that deep-learning-based methods outperform traditional approaches that rely on weighted fusion, sparse subset selection, or data clustering algorithms. Frames or segments are selected based on an importance score computed by the summarizer network [1, 53]. However, traditional DNN video deep-learning-based summarization networks can not eliminate redundancy within the selected summaries, resulting in limited diversity performance.

The submodular function is a promising approach to improve the diversity of summaries. Unlike deep neural networks, submodular functions exploit their diminishing returns property to fit summary selection. That is, in video summarization, the incremental value of adding a video frame decreases with the growth of the summary, thus encouraging diversity [58, 25]. Moreover, submodular functions offer desirable optimization properties [27]. Generic summarization addressed by cardinality-constrained submodular maximization can be resolved in a constant factor ($\approx 63\%$) using greedy algorithm [31]. The weakness lies in the submodular part's non-deep function, resulting in no interaction among features.

The concept of deep submodular functions (DSFs) has been proposed recently [5, 16]. By utilizing additional layers of nested concave functions, DSFs have a multi-layered architecture similar to the feed-forward deep neural networks (DNNs). Submodularity follows if the data features satisfy modularity and the layers are constructed by non-negative weights. Nevertheless, the 2D structure of DSFs ignores key temporal information within the video [10]. Moreover, using only modular features restricts the capability of modeling the video summarization problems.

In this paper, we generalize DSFs and devise a submodular pseudo-3D (SP3D) network to achieve video summarization with better diversity. The main contributions of this work are summarized as follows:

1. We propose a novel submodular pseudo-3D (SP3D) network, which captures the diverse spatiotemporal content from micro-videos. The SP3D network generalizes the DSFs to accept not only modular features and handle inputs with 3D structures more efficiently.
2. A learning and optimization framework is developed to train and assess the SP3D network efficiently. Our method is more than 10 times faster than previous work.
3. We demonstrate the practical benefit of the proposed SP3D network in video summarization. Over 3,000 micro-videos are used in the experiment. The algorithm is confirmed effective according to the experiment conducted.

2. Related Work

2.1. Submodular Functions

Set functions are submodular if they fulfil the diminishing returns property [30], that is, the incremental value of adding an element decreases as the size of a set grows [4, 43]. The definitions of modular and submodular functions are in Subsection 3.1 General Definition. Submodular functions have strong theoretical support and a wide variety of applications.

Firstly, submodular functions possess attributes for efficient optimization. For monotone non-decreasing submodular functions, the classic result under uniform matroid constraint is the (1-1/e)-approximation via greedy algorithm [43, 44]. One may consider more general matroid constraints. The greedy algorithm achieves a 1/2-approximation [21, 39]. Calinescu et

al. [7, 8] leveraged a continuous relaxation (i. e. multilinear extension) and extended the (1-1/e)-optimal solution to an arbitrary matroid constraint.

Secondly, submodular functions have a wide variety of applications in the field of machine learning, especially in representing diversity. They have been utilized as diversity functions for data summarization, including document summarization [38, 56], image collection summarization [30, 51], stream data summarization [12, 42], etc. And they are useful in diversified mini-batch selection [28], web search [9, 11], interactive recommendation [15] and so on.

However, the expressivity of the traditional submodular functions is limited by their shallow structure. Bilmes et al. [4, 5] introduced deep submodular functions (DSFs). DSFs are a flexible parametric family of submodular functions that share many properties and advantages of deep neural networks (DNNs), including the many-layered hierarchical topologies, training strategies, etc. The expressivity of DSFs strictly grows with the number of layers.

Although a DSF has favorable properties for data representation and problem optimization, the 2D architecture impedes its implementation on videos. In this paper, we bridge the submodular functions with 3D architecture.

2.2. Submodular Functions and Video Summarization

In video summarization, submodular functions are exploited to improve the diversity of the selected summary. The determinantal point process (DPP), which is a non-monotone submodular function, is a powerful probabilistic model for diverse subset selection [32]. Xu et al. [55] leveraged DPP to measure the diversity of a summary with mutual information between the selected summary and the remainder of the sequence. Zhang et al. [57] utilized DPP to extract a globally optimal subset of frames when transferring the subset structures in the human-created summaries of the training videos to a new video. Zhang et al. [58] combined the LSTM network with a DPP to increase the diversity in the selected subsets. Banihashem et al. [3] proposed a dynamic algorithm for non-monotone submodular maximization and implemented it in video summarization using DPP.

Monotone submodular functions have also been explored for video summarization to reduce computation complexity. Gygli et al. [25] formulated the task of video summarization as a subset selection problem, i. e., monotone submodular functions maximization. Li et al. [37] built a general summarization framework to summarize both edited and raw videos. The problem was cast as monotone submodular functions maximization to find an optimal solution. Elhamifar et al. [17] considered the problem of online video summarization and proposed an incremental subset selection framework. The optimized solution for the framework was found via solving an unconstrained submodular objective function. Li et al. [35] provided a deterministic algorithm that achieves a 1/2-approximation for monotone submodular maximization subject to a knapsack constraint, and achieved several orders of magnitude faster than the baseline methods in video summarization.

Despite the valuable results of these methods, their weakness lies in the non-deep function of the submodular part. Compared to deep networks, shallow structures can hardly learn data representations at a more abstract level, resulting in little interaction among features. Our work bridges the gap between submodular functions and multi-layered networks.

3. Submodular Pseudo-3D Network

3.1. General Definition

Let V denotes a ground set of n elements. A set function $f: 2^V \rightarrow R_+$ is submodular if it fulfills the diminishing returns property, i.e., given arbitrary sets $A \subseteq B \subseteq V$ and an element $v \in V \setminus B$, it holds $f(A \cup v) - f(A) \geq f(B \cup v) - f(B)$. It is modular if both f and –f are submodular. Function f is said to be monotone non-decreasing if $f(A \cup v) - f(A) \geq 0$ for any $v \in V$ and $A \subseteq V$. Traditional submodular functions are submodular functions connected by weighted fusion. Submodular or modular features are feature extraction functions that satisfy submodularity or modularity, respectively [4]. In our experiment, interestingness features are utilized for the Emotion6V dataset. The interestingness feature is a kind of submodular feature [25]. The GoogLeNet feature is modular, since selecting one more frame for the summary or removing a frame from the summary does not impact the features of the other selected frames.

3.2. Continuous Extensions of Submodular Functions

A submodular function provides discrete results, but lifting the problem into the continuous domain helps to obtain optimal results. The continuous extension of a submodular function f is some function from the hypercube $[0,1]^V$ to R that agrees with f on the vertices of the hypercube [18]. We make use of the below DSF concave extension in optimizing our SP3D network.

Corollary 1. (DSF concave extension [2,5]) The DSF concave extension $F(x):[0,1]^n \rightarrow R$ is an extension of a DSF f(A) and is concave.

3.3. Matroid and Cardinality Constraints

A matroid M = (V,I) is an abstraction of linear independence structure among the columns of a matrix. Since the matroid includes subsets with independent elements, it offers the desirable property for summarization. We adopt a uniform matroid constraint in our video summarization, i.e., a cardinality constraint, which is the family of all subsets with cardinality k. Like general matroids, cardinality constraints can be generalized to the continuous domain as polytopes defined below [22]:

$$P_k = \left\{ x \in R_+^n \mid \sum_{i=1}^n x_i = k, 0 \le x_i \le 1 \right\}. \quad (1)$$

3.4. Submodular Pseudo-3D Functions

The general definition of the SP3D function is shown below:

Definition 1. Let V be a ground set of n elements, X_V with n rows be the features extracted from V, and $\forall S \subseteq V$. The submodular pseudo-3D function of S, $SP3D: 2^V \rightarrow R_+$, is defined as follows:

$$SP3D_{X_V}(S) = H\left(\ldots \omega_3 G3D\left(\omega_2 F3D\left(\omega_1 Conv3D\left(X_v \bullet S\right)\right)\right)\right), \quad (2)$$

where Conv3D(·)=Conv1D(Conv2D(·)) is the pseudo-3D function, F3D(·)=Conv1D(f(·)) and G3D(·)=-Conv1D(g(·)) are combinations of temporal convolution Conv1D(·) and the traditional submodular functions.

The F3D(·) and G3D(·) are performed as layers of submodular pseudo-3D functions. Here, the Conv1D Conv2D:$R_+ \rightarrow R_+$ are 1D and 2D discrete convolution functions, respectively. The $f(\cdot), g(\cdot), H(\cdot): R_+ \rightarrow R_+$ are non-negative, monotone non-decreasing continuous concave functions, and ω_i are trainable weights with $\omega_i \ge 0$. The $X_V \cdot S$ is an element-wise multiplication and represents that the summary set S is embedded into the feature space X_V.

Proposition 1. SP3D is a deep submodular function. The DSF concave extension $\bar{S}: [0,1]^n \rightarrow R$ is a continuous extension of SP3D and is concave.

Proof. See Appendix A.

4. Submodular Pseudo-3D Network for Video Summarization

4.1. Problem Formulation

Given a video V with n frames, we formulate the video summarization task as the maximization of the SP3D function under cardinality constraint as follows:

$$s^* = \arg\max_{s \in M} SP3D_{X_V}(s), \quad (3)$$

where $X_V \in R^{n \times f}$ is the f-dimensional feature set extracted from video V, $SP3D_{X_V}(\cdot)$ is the submodular pseudo-3D function associated with the video feature X_V, M=(V,T) is the cardinality constraint, $s \subseteq V$ is any possible summary set, and s^* is the selected summary.

Then the objective function Equation (3) is lifted to the continuous domain as below:

$$y^* = \arg\max_{y \in P_k} \bar{S}_{X_V}(y), \quad (4)$$

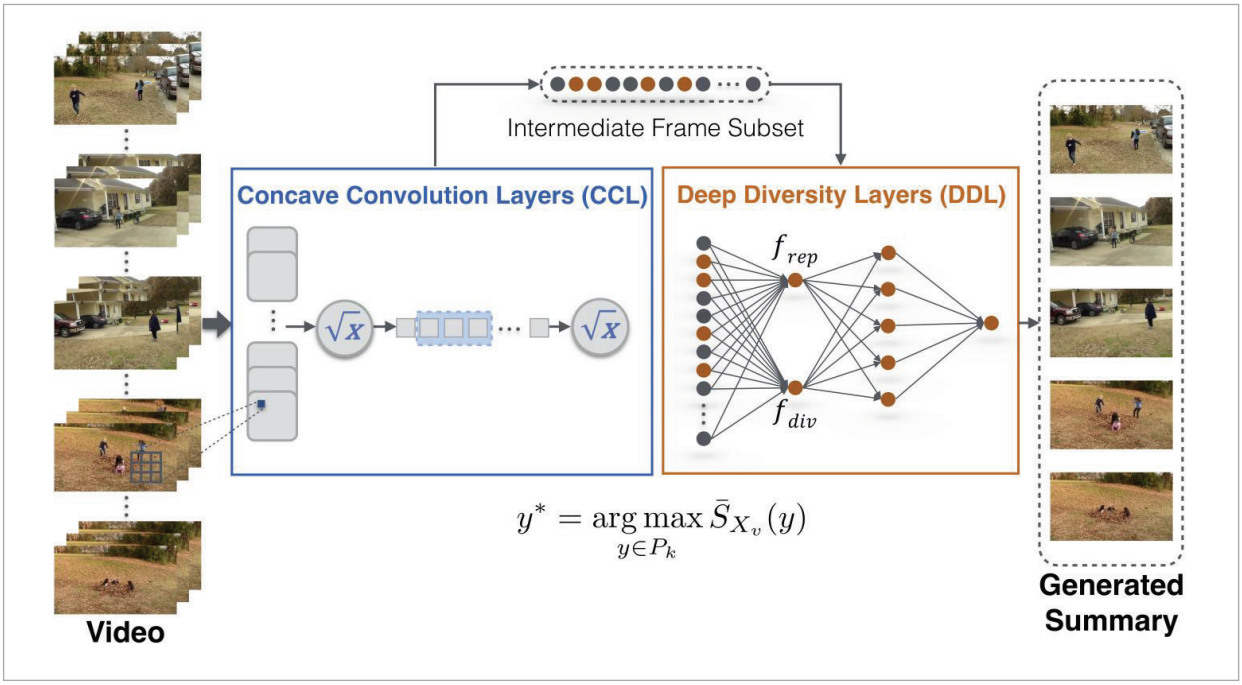
where $\bar{S}$ is the concave extension of SP3D proposed in Proposition 1, P_k is the corresponding polytope defined in Equation (1), and $y = 1_s, 1_s \in R_+^V$ is 0 for the i-th element $i \notin s$ and 1 for $i \in s$.

4.2. Network Architecture

As illustrated in Figure 1, the SP3D network contains the concave convolution layers (CCL) and deep diversity layers (DDL). The CCL converts DNNs to satisfy submodularity, capturing the intricate video

Figure 1

An overview of the submodular pseudo-3D (SP3D) network. It consists of two building blocks, the concave convolution layers (CCL) and deep diversity layers (DDL). Both blocks are submodular with multi-layer structures. The parameters in SP3D are trained via modern optimization techniques available to DNNs. The final optimized result is obtained using projected supergradient ascent and rounding.



information for further processing. The DDL leverages a multi-layered architecture to improve the diversity of the summary. Experimental results in Subsection 5.5 demonstrate their effectiveness.

Our CCL has a similar pseudo-3D architecture to the (2+1)D convolutional block proposed in [50]. Unlike the standard pseudo-3D structure, our CCL adopts a concave function, element-wise square root (element-wise sqrt), as the activation function and keeps the network parameters non-negative. A CCL block comprises several concave convolution units and one fully connected linear layer. Each unit contains one spatial convolution and one temporal convolution as pseudo-3D modeling. The concave activation is applied after the spatial or temporal convolution.

The DDL block improves the diversity of the summary subset extracted from a video. It contains submodular functions in a multi-layer structure. The output of CCL Y^*_{CCL} and the video features X_V are both taken as the input of the DDL. The Y^*_{CCL} performs as the initial summary set for DDL optimization. The next layer is submodular components f_{rep} and f_{div}, which capture a compact summary with good coverage. Before the linear output layer, concave activations are implemented over the weighted sums of the results from the previous layer, that is $f = \sqrt{\omega_{rep} f_{rep} + \omega_{div} f_{div}}$. In contrast to the traditional deep neural networks, the intermediate summary Y^*_{CCL} is optimized in DDL as a whole, i.e., as an n-dimensional variable, to achieve the final result. We will describe the details in the following subsections.

4.3. Learning

Given N_t training video clips V_c with $c = 1,2,...,N_t$, we optimize the following formulation to learn the weight vectors ω:

$$\omega^* = \arg\min_{\omega>0} \sum_{c=1}^{N_t} L_c(\omega) + \|\omega\|_1, \quad (5)$$

where $L_c(\cdot)$ is the loss function and $\|\omega\|_1$ is the l_1-norm of the parameters. The loss function could be l_1 loss, hinge loss, or squared loss. In our model, we adopt

the MSE loss. Equation (5) is the training objective function. In our experiment, the training iteration is 100. The termination condition is set to 20, which means that if there is no improvement after 20 training iterations, the training will be stopped to prevent over-fitting. In each training step, we project the ω back to the non-negative quadrant to retain the submodularity, which will be introduced in Section 4.4.

4.4. Optimization

The concave extension of SP3D can be efficiently maximized via projected supergradient ascent (PGA), where the optimized solution is projected to the constraint space. It ensures that the results meet the constraints criteria and the objective functions preserve submodularity after each optimization step. The continuous solution of PGA will be rounded to obtain an optimized set. The overall optimization algorithm is summarized in the Algorithm 1, and a visualization of the optimization procedure is in Appendix B. The method in this subsection is different from the one in Subsection 4.3. The updated object is the summary to be generated, not the network weights. We describe the details of our optimization algorithm below.

Algorithm 1 Optimization of the concave extension of SP3D

Require: SP3D concave extension $\overline{S}$, polytope P_k, learning rate η and max iteration number T.

Ensure: optimized summary set y^*

Initialize: $y^{(0)} \leftarrow$ a starting point in P_k.

For $t = 0$ **to** T **do**

$g(y^t) \in \partial\overline{S}(y^t)$

$y^{t+1/2} = y^t + \eta g(y^t)$

$y^{t+1} = \text{Projection}(y^{t+1/2})$

end for

$\bar{y} = \frac{1}{T}\sum_{t=0}^{T} y^{(t)}$

$y^* = \text{Rounding}(\bar{y})$

4.4.1. Supergradient Ascent

The g in the Algorithm 1 is the supergradient of SP3D's concave extension $\overline{S}$. For a concave function in Definition 1, the supergradient is its derivative at a specific valuation if it is differentiable.

4.4.2. Projection

Projection is a subproblem of constrained optimization. An efficient projection plays a crucial role in the optimization procedure since it is executed in every iteration step [6]. In SP3D optimization, we project the supergradient ascent result $y^{t+1/2}$ to the polytope P_k defined in Equation (1). It entails finding a point $y^{t+1} \in P_k$ such that $y^{t+1} = \arg\min_{x \in P_k} \frac{1}{2}\left\|x - y^{t+1/2}\right\|_2^2$, which is a convex optimization problem by itself. We can solve it leveraging the KKT optimality conditions and Lagrangian function in O(nlogn) time. See Appendix C for proof.

In practice, we first sort the elements in $y^{t+1/2}$ to a non-decreasing list, and use linear interpolation to find i_0, the maximum index i such that $f\left(y_{i_0}^{t+1/2}\right) \geq k$. The optimal value of λ is within the range of $\left[y_{i_0}^{t+1/2}, y_{i_0+1}^{t+1/2}\right]$. The y^{t+1} can be computed element-wise using $y^{t+1/2}$ and λ. The detailed procedure is described in Algorithm 2.

Algorithm 2 Projection on the Polytope P_k

Require: vector $y^{t+1/2} \in R^n$ and cardinality constraint $k \in N$.

Ensure: y^{t+1}

Sort elements in $y^{t+1/2}$ so that

$y_0^{t+1/2} \leq \ldots \leq y_n^{t+1/2}$

For $i=n$ **to** 0 **do**

$\lambda \leftarrow y_{i-1}^{t+1/2}$

$f \leftarrow \sum_{j=0}^{n-1} \min\left\{\max\left\{y_j^{t+1/2} - \lambda, 0\right\}, 1\right\}$

If $f \geq k$ **then**

$i_0 \leftarrow i$

break

end if

end for

$\lambda = \frac{\sum_{i=i_0}^{n-1} y_i^{t+1/2} - k}{n - i_0}$

$y_i^{t+1} = \text{median}\left(0, y_i^{t+1/2} - \lambda, 1\right)$

4.4.3. Projected Supergradient Ascent

One of our main optimization tools is the projected supergradient ascent to maximize the continuous concave extension of SP3D. Its convex analog, projected subgradient descent, has been well studied.

We modify the theory for convex functions minimization to build the optimization boundary for maximizing our SP3D network.

The exact values for R and B are problem-dependent. We will provide tighter bounds for SP3D for video summarization in the later sections.

Theorem 1. [2,6] For an SP3D concave extension $\bar{S}:[0,1]^n \to R$, let $R^2 = \sup_{y \in M} \frac{1}{2}\|y\|_2^2$ and $B^2 = \sup_{y \in M} \frac{1}{2}\|g(y)\|_2^2$, Algorithm 1 with learning rate $\eta = \frac{R}{B}\sqrt{\frac{2}{T}}$ will obtain a fractional solution $\bar{y}$ such that $\bar{S}(\bar{y}) \geq \max_{y \in M} \bar{S}(y) - RB\sqrt{\frac{2}{T}}$, where T is the total number of iterations. And for each $0 < \varepsilon < 1$, Algorithm 1 will produce the fractional solution $\bar{y}$ such that $\bar{S}(\bar{y}) \geq (1-\varepsilon)\max_{y \in M} \bar{S}(y)$ with running time $O(R^2B^2\varepsilon^{-2})$.

Proof. The $RB\sqrt{\frac{2}{T}}$ performance bound is adapted from [6] and restated for concave functions. The boundary of T is modified from [2].

4.4.4. Rounding

Rounding is a technique to obtain a discrete set from a fractional vector. Given $\bar{y}:[0,1]^n$, we say $\bar{y}$ is fractional if any $0 \leq \bar{y}_i \leq 1$. In the polytope P_k, rounding aims to move from a starting point $\bar{y}$ inside P_k to a vertex of the polytope. The rounding result y^* is the final summary set generated by the SP3D network. A visualization of the rounding step can be found in Appendix B.

The traditional randomized pipage rounding leverages a convex property of the multilinear extension [2]. Since the convex property is not required in SP3D, we adopted the deterministic pipage rounding [26] and simplified it for cardinality constraints. The detailed rounding methodology is presented in Appendix D. The time complexity is O(n), and its proof is shown in Appendix E.

4.4.5. Time Complexity

In this subsection, we present the concave functions utilized in the SP3D network, i.e., the f_{rep} and f_{div} in Subsection 4.2, and compute the running time.

The f_{rep} quantify the coverage of the selected summary S by the similarity between the S and the input video V. In Equation (6), we adopt the Euclidean norm $||V_i - S_j||_2$ as the distance function in s. The entry $1 - s_{i,j}$ is interpreted as the measure of the similarity.

$$f_{rep}(S) = \sum_{i \in V} \sum_{j \in S} 1 - s_{i,j}, \quad (6)$$

The diversity function f_{div} calculates the similarities within the summary S by summing up all the pairwise distances, as shown in Equation (7).

$$f_{div}(S) = \sum_{i \in S} \sum_{j \in S, j > i} s_{i,j}, \quad (7)$$

The following proposition shows that the running time of the above video summarization network is $O(kn\varepsilon^{-2})$, and the proof is detailed in Appendix F.

Proposition 2. Let function F be the SP3D network for video summarization with representativeness and diversity functions defined in Equations (6) and (7). The concave extension is $\bar{F}:[0,1]^n \to R$ and k is the cardinality constraint. For each $0 < \varepsilon < 1$, Algorithm 1 will produce the frational solution $\bar{y}$ such that $\bar{S}(\bar{y}) \geq (1-\varepsilon)\max_{y \in M} \bar{S}(y)$ with running time $O(kn\varepsilon^{-2})$.

Here we relate our results with the classical greedy approach. When solving a submodular function, the standard greedy is an iterative algorithm that selects the element with the maximum function value at each step. The time complexity of greedy is $O(nk^2\varepsilon^{-2})$, where O(nk) and O(k) are the time taken for element selection and a single evaluation of the submodular function, respectively. Hence, the complexity time of our optimization method has a factor k speedup than the greedy method. Since more than 10 frames are usually selected as summaries to express the original video effectively, our method is more than 10 times faster than the greedy method.

5. Experiment

5.1. Setup

5.1.1. Dataset

We conduct the experiments on SumMe [24], TVSum [48], YouTube [14] and OVP [45] datasets. The number of videos in the four datasets is 25, 50, 39, and 50, respectively. Given the small size of the datasets mentioned earlier, our experiment utilizes the Emotion6 video dataset [52], which contains 3,600 micro-videos. Each dataset contains rich and diverse content as follows: SumMe [24] consists of raw

Table 1
Comparison of F1 score (%) with state-of-the-art video summarization methods. The SumMe and TVSum datasets are employed with canonical, augmented, and transfer settings.

Method	SumMe			TVSum		
	C	A	T	C	A	T
Submodularity [12]	39.7	-	-	-	-	-
dppLSTM [7]	38.6	42.9	41.8	54.7	59.6	58.7
G-SUM [39]	43.1	-	-	52.7	-	-
DR-DSN [8]	42.1	43.9	42.6	58.1	59.8	58.9
SUM-FCN [55]	47.5	51.1	44.1	56.8	59.2	58.2
SUM-FCNuns [55]	41.5	-	-	52.7	-	-
VASNet [56]	49.7	51.1	-	61.4	62.4	-
DSNet [57]	51.2	53.3	47.6	61.9	62.2	58.0
RSGN [58]	45.0	45.7	44.0	60.1	61.1	60.0
RSGNuns [58]	42.3	43.6	41.2	58.0	59.1	59.7
AMF [59]	51.9	54.3	50.2	63.2	65.6	60.4
SP3D (Ours)	**52.9**	**56.4**	**50.4**	**64.1**	**65.8**	**58.2**

or minimally edited user videos, covering a variety of events such as holidays and sports. TVSum [48] collects videos from YouTube representing various genres. It covers a variety of topics, including news, how-to guides, documentaries, and user-generated content, such as vlogs and egocentric videos. YouTube [14] contains videos collected from websites like YouTube. These videos are distributed among several genres (cartoons, news, sports, commercials, tv-shows and home videos). OVP [14,45] is a shared video collection that spans several genres, including documentary, educational, ephemeral, historical, and lecture. Emotion6 [52] is a synthetic dataset of emotional videos using images. Since the images do not contain facial expressions or text directly associated with video summarization, the content of this dataset is highly diverse and unstructured.

5.1.2. Evaluation Metric

We adopt two popular evaluation metrics: F1 score and rank correlation coefficients [46]. The F1 score evaluates the temporal overlap between the predicted summaries and the human-annotated key frames, while rank correlation compares the ranking of the importance scores between the created and annotated summaries. Kendall's τ and Spearman's ρ correlation coefficients are employed as the rank-based metrics.

5.1.3. Implementation Details

For SumMe, TVSum, YouTube, and OVP datasets, the frame features are extracted from the Pool 5 layer of the GoogLeNet model, which leverages ImageNet for pre-training. Each frame has a feature dimension of 1024. We apply three settings, canonical, augmented and transfer, to study the performance of the proposed SP3D network. In the canonical setting (C), 80% of the given dataset is used for training, and the remaining 20% is for evaluation. In the augmented setting (A), the training set includes videos from the other three datasets. For the transfer setting (T), training is based on the other three datasets, and testing is performed on the specified SumMe or TVSum dataset. The Emotion6V dataset's feature dimension is 4225, including interestingness features [23] that are not modular but submodular.

All the parameters used in our network are learned using AdaGrad optimizer and mean squared error (MSE) loss. The learning rate is 5×10^{-5} with a weight decay of 10^{-5}. In addition, we add a dropout layer with

a rate of 0.5 to the fully connected layer of the concave convolution network.

For a testing video, we leverage the algorithms in Subsection 4.4 Optimization to assess the trained DDL model. The optimizer is AdaGrad, and the learning rate is calculated according to Proposition 2. We follow the general practice to select video shots by averaging frame-level scores within a shot. Our rounding method is based on shot-level scores.

In addition, we use 3×3×1 spatial filters and 1×1×3 temporal filters with convolutional striding of 1×1×1 in CCL. Our SP3D network is implemented under the PyTorch framework.

5.2. Quantitative Results

The methods, Submodularity [25] and G-SUM [37], are based on submodular functions. The Submodularity [25] formulated the video summarization task as a supervised subset selection problem. The G-SUM [37] built a general video summarization framework. Both methods cast the problem as monotone submodular functions maximization with a shallow network structure. The dppLSTM [58] combined the deep network LSTM with non-monotone submodular function DPP. LSTMs were leveraged to capture a video's temporal sequence, and the DPP improved the diversity of the selected summary subset. Recent studies mainly employ deep networks. The DR-DSN [62] trained a deep summarization network with a CNN encoder and a bidirectional LSTM decoder. In SUM-FCN [47], fully convolutional networks (FCNs) were adopted to indicate if the corresponding frame was selected as the summary. The VASNet [20] introduced the self-attention mechanism and replaced the LSTM network with a deep regression network. Instead of formulating video summarization as a regression problem without temporal consistency, the DSNet [63] detected temporal interest proposals to represent a video. The RSGN [60] encoded the higher shot-level dependencies using graph convolutional network in addition to the lower frame-level dependencies captured by LSTM. SUM-FCNuns and RSGNuns are the unsupervised variant of method SUM-FCN and RSGN, respectively. AMF [54] designed a visual-aesthetics encoder to extract diverse aesthetic elements and jointly integrate with visual content to create a comprehensive summary.

Table 1 shows the results in F1 score. From the table, our findings are three-fold.

1. Overall, our SP3D network outperforms other baseline methods in both SumMe and TVSum datasets, which verifies the effectiveness of our approach in capturing key information from videos.
2. Thanks to the submodular functions, our SP3D performs better on the SumMe dataset, which has more diverse content and is usually more challenging to summarize. The experimental results indicate that our SP3D narrows the performance gap between SumMe and TVSum. In particular, the accuracy of the augmented setting under the SumMe dataset is significantly improved compared to the canonical setting. It proves the learning capability of our training methodology.
3. Across the three experiment settings, we see that the performance enhancement of SP3D is the best under the canonical setting. It benefits from exacting temporal information and diverse content, alleviating the difficulties caused by the lack of annotated data.

To validate the practical usage of SP3D, we conduct experiment on the Emotion6V dataset, which includes 3,600 videos. As shown in Table 2, our SP3D network outperforms other summarization approaches for user-generated short videos, suggesting SP3D's feasibility in summarizing micro-video.

Table 2

Comparison of F1 score (%), Precision (%), and Recall (%) with state-of-the-art summarization methods using the Emotion6V dataset with a total of 3,600 videos.

Method	F1 score	Precision	Recall
Uniform sampling	52.2	55.4	49.9
RPCA [60]	50.6	53.6	48.3
BEAC [52]	62.3	63.9	61.1
SP3D (Ours)	**65.3**	**69.2**	**62.3**

Table 3 presents the results in rank-based Kendall's τ and Spearman's ρ correlation coefficients. We can clearly observe that our SP3D network consistently attains the best result against the other methods in both Kendall's τ and Spearman's ρ measures. Since the proposed optimization method up-

Figure 2
The qualitative results of SP3D video summaries. The curves denote the generated importance scores, and the grey bars depict the ground truth scores. The blue bars are selected as final summaries.

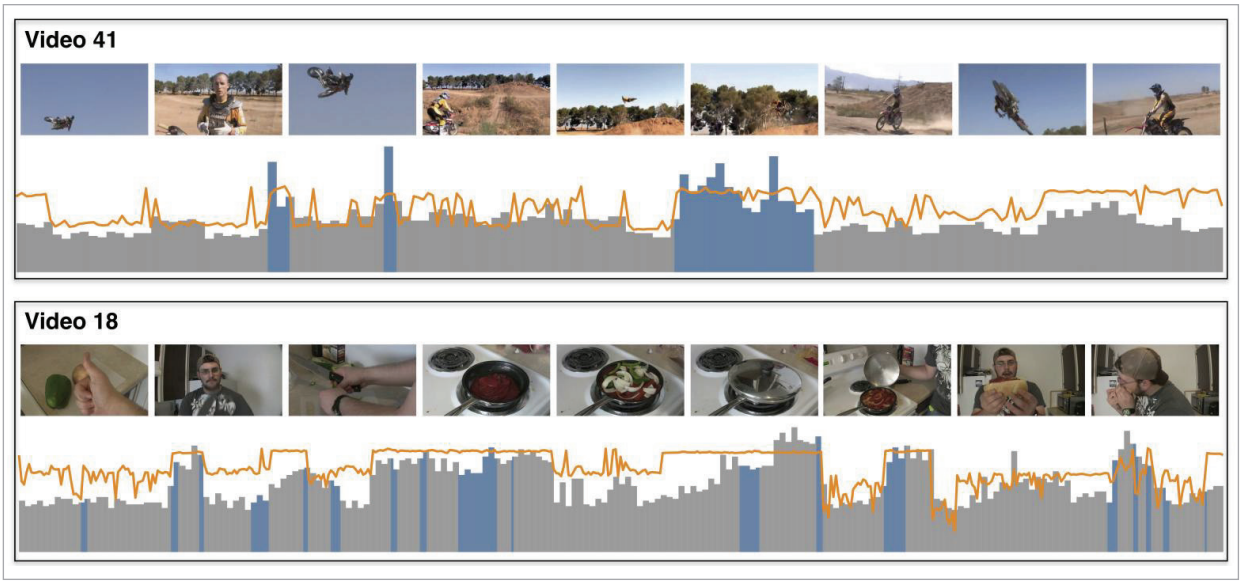


Table 3
Comparison of the rank-based correlation coefficients, Kendall's τ (K) and Spearman's ρ (S), with state-of-the-art video summarization methods. The SumMe and TVSum datasets are employed with the canonical setting.

Method	SumMe		TVSum	
	K	S	K	S
dppLSTM [7]	-	-	0.042	0.055
DR-DSN [8]	0.011	0.013	0.020	0.026
DSNet [57]	0.038	0.044	0.097	0.118
RSGN [58]	0.083	0.085	0.083	0.090
AMF [59]	0.071	-	0.063	-
SP3D (Ours)	**0.104**	**0.126**	**0.120**	**0.175**

dates the summary subset as a whole, the implicit ranking of the generated summary is correlated to the ground truth.

Are the summaries generated by our SP3D more diverse than those generated by other methods? Table 4 demonstrates the comparison results. S3 [33] is a summarization method based on shallow submodular functions. MLP [49], DR-DSN [62], and DSNet [63] are DNN-based summarization methods. MLP is based on multi-layer perceptrons (MLPs) and does not use convolutions or self-attention. The experimental results show that our SP3D outperforms both submodular with shallow architecture and DNNs, especially for the more diverse SumMe dataset.

Table 4
Comparison of the summary representativeness and diversity with state-of-the-art video summarization methods. The SumMe and TVSum datasets are employed with the canonical setting.

Method	SumMe		TVSum	
	Rep.	Div.	Rep.	Div.
S3 [61]	0.52	0.77	0.55	0.81
MLP [62]	0.64	0.51	0.57	0.75
DR-DSN [8]	0.64	0.63	0.63	0.88
DSNet [57]	0.57	0.66	0.61	0.82
SP3D (Ours)	**0.66**	**0.79**	**0.69**	**0.90**

5.3. Qualitative Results

Figure 2 displays the exemplar video summaries generated by our SP3D. Video 41 is a sharing of

the motocross tips on how to whip a motocross bike. And video 18 describes the process of making sausage sandwiches. The displayed frames are sampled from the generated summaries. We can easily imagine the main story of the videos, which demonstrates the effectiveness of our summarization method. Moreover, the predicted importance scores are highly correlated with the ground truth with similar ups and downs. The consistency illustrates the benefit of updating the summary as a whole during optimization. Specifically, in video 18, we can see that the SP3D is capable of selecting diverse content. With primary and diversified parts included, our SP3D can model user-generated videos well.

5.4. Complexity Analysis

Table 5 demonstrate that SP3D has a significantly reduced number of parameters comparing with DNNs. In Table 6, we can see that our SP3D optimization method is over 10 times faster than the traditional greedy method.

Table 5
Analysis results of network complexity. Number of network parameters are compared.

SP3D	DR-DSN	DSNet
13K	2626K	4328K

Table 6
Analysis results of time complexity. Optimization time of SP3D and greedy are compared.

Method	SumMe	TVSum
SP3D	4	9.4
Greedy	65	718.4

5.5. Ablation Study

The proposed SP3D network comprises the concave convolution layers (CCL) and the deep diversity layers (DDL). We conduct ablation studies in this subsection to understand the contribution of each component in our proposed SP3D network. Video summaries are generated for both SumMe and TVSum datasets. The following settings of SP3D are considered:

1) SP3Dccl. To understand the contribution of the CCL component, we remove the DDL part and evaluate the output of the CCL intermediate frame subset straightly. The parameters learned in the full SP3D model are utilized to ensure comparable performance. We report the evaluation results on the F1 score and Kendall's τ. 2) SP3Dddl. This ablation model retains only the trained DDL component. The CCL parameters are assigned by random. Other implementation settings are similar to the SP3Dccl with CCL replaced by DDL. 3) SP3Dwot. We keep the full SP3D model here but ignore the training stage. The parameters in the network are randomly assigned when the module is initialized. It will help us understand the benefit of the proposed learning methodology. 4) SP3Dgre. The conventional optimization procedure for submodular functions is greedy. This paper develops the optimization framework based on projected gradient ascent (PGA) to generate a summary. Here we employ the greedy algorithm to verify the effectiveness of the optimization framework. The optimization uses the trained SP3D full model.

Table 7 reveals that SP3D full model performs best, indicating the effectiveness of our algorithm. The SP3Dccl model achieves almost comparable F1 scores but negative rank-based correlation scores. It suggests that the CCL block captures the intricate 3D information for F1 score evaluation while the DDL unit optimizes the rank-based correlation. Further study SP3D2D retains only the 2D spatial convolution in CCL. The model's performance drops on both datasets, which validates the importance of the temporal dimension in video summarization. The results of SP3Dnonsub demonstrate the disadvantage of removing submodularity. Both the F1 and Kendall's τ scores decline significantly. The diminishing returns property of a submodular function helps keep the summary subset compact and representative. The results of SP3Dwot and SP3Dgre in Table 7 demonstrate the effects of our training and optimization process, respectively. Without training, the performance of SP3Dwot is not so good as the trained model, especially for the more diverse SumMe dataset. From the results of SP3Dgre, we can find that the greedy optimization approach reduces the performance, which validates that our proposed optimization framework is crucial to get an optimized summary.

Table 7
Comparison of F1 score (%) and Kendall's τ for SP3D variation models on SumMe and TVSum datasets.

Dataset	Method	F1 score	Kendall's τ
SumMe	SP3Dccl	49.9	-0.00168
	SP3Dddl	51.4	0.0979
	SP3Dwot	34.8	0.0122
	SP3Dgre	46.6	0.0746
	SP3D2D	47.8	0.0885
	SP3Dnonsub	41.7	0.0181
	SP3D	**52.9**	**0.104**
TvSum	SP3Dccl	62.8	-0.0229
	SP3Dddl	64.1	0.0866
	SP3Dwot	59.5	0.0166
	SP3Dgre	52.3	0.0514
	SP3D2D	63.1	0.0788
	SP3Dnonsub	46.6	0.0236
	SP3D	**64.1**	**0.12**

6. Conclusion

This paper proposes a novel submodular pseudo-3D network (SP3D) for micro-video summarization. It satisfies the submodular property and has a deep network structure like deep neural networks (DNNs). A learning and optimization framework is developed to train and assess the SP3D network efficiently. Extensive experiments demonstrate promising results. Currently, our model utilizes only visual features. Although the visual feature is the most important part of micro video summarization, leveraging other modalities will help improve the summarization efficiency. In the future, we would like to improve the model to accept multimodal inputs, like textual and acoustic modalities.

Appendix A

Here is the proof of Proposition 1.

Proposition 1. SP3D is a deep submodular function. The DSF concave extension $\bar{S}:[0,1]^n \to R$ is a continuous extension of SP3D and is concave.

Proof. We prove that the SP3D is a deep submodular function as the concave extension of a DSF was claimed in [5] and proved in Theorem 1 of [2].

Let us begin by understanding the convolution operation. The 2D discrete convolution is defined as [34]: $(K*I)(i,j)=\sum_m\sum_n I(i-m,i-n)K(m,n)$, where $K\in R^{m\times n}$ is a two-dimensional kernel, *I* denotes the input array, and (i,j) is the point where the convolution operation is performed. The 1D discrete convolution is a special case of 2D with m or n equal to 0. The convolution operation is a weighted average, and it can also be taken as a modular function $2^{m\times n}\to R_+$ for each output point. Since modularity follows in the composition of modular functions, the Conv3D function defined in Definition 1 is a modular function.

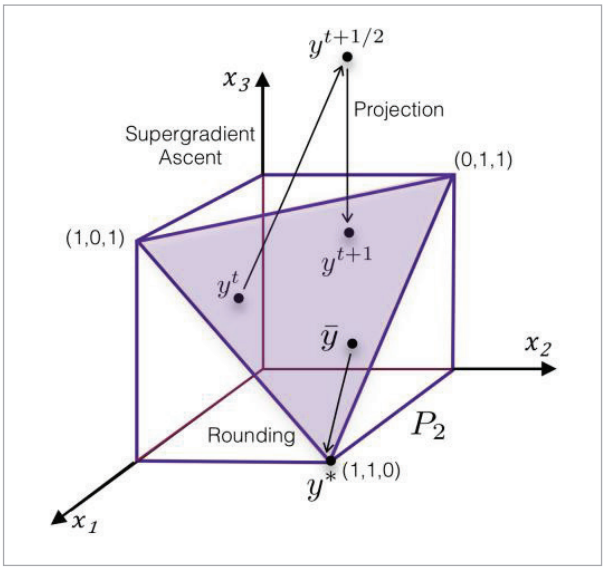
In F3D(·), the ω_1Conv1D(·) can be considered as the weighting for f(·). Then the combination of (ω_2F-3D(ω_1Conv3D(·))) is a submodular function according to the SCMM definition that SCMM is Sums of Concave over non-negative Modular plus Modular. Applying Theorem 5.4 from [5], the ω_3G3D(ω_2F-3D(ω_1Conv3D(·))) is a monotone non-decreasing submodular function. Using induction, we know that the Definition 1 is a submodular function and satisfies the DSF architecture. Therefore, it has a natural concave extension by replacing the discrete variables with real values.

Appendix B

Here is the visualization of the optimization algorithm. The general framework mentioned in Subsection 3.1 includes a continuous extension, the projected optimization procedure, and the resulting solution after rounding. As presented in Algorithm 1, the optimization procedure involves the iteration of supergradient ascent and projection for T times. In each iteration, the starting point y^t is updated to $y^{t+1/2}$ using the supergradient ascent in Equation (7). Since the updated point $y^{t+1/2}$ may not satisfy the constraints, we project it back to the constraint space P_k via finding the closest point $y^{t+1}\in P_k$, that is, $y^{t+1}=\arg\min_{x\in P_x}\frac{1}{2}\left\|x-y^{t+1/2}\right\|_2^2$. The projected point y^{t+1} lies in the polytope P_k and will be the starting point of the next iteration.

Figure A1

The illustration of the optimization procedure and rounding technique under the polytope P_2. y^t is updated to $y^{t+1/2}$ using the supergradient ascent. $y^{t+1/2}$ is projected to the polytope via finding a point $y^{t+1} \in P_k$ that $y^{t+1} = \arg\min_{x \in P_k} \frac{1}{2}\left\|x - y^{t+1/2}\right\|_2^2$. Rounding moves the fractional solution $\bar{y}$ to a vertex of the polytope P_k.



Assuming that the constraint polytope in Equation (1) has n = 3 and k = 2, then the polytope $P_2 = \left\{x \in R_+^3 \mid \sum x_i = 2, 0 \le x_i \le 1\right\}$. And the above optimization procedure is illustrated in Figure A1. The plane in purple is the polytope P_2. The optimization starts with y^t on P_2. The updated $y^{t+1/2}$ is outside of the polytope. Then we have projection to find a closest point, y^{t+1}, on P_2. And the next iteration starts from y^{t+1}.

The rounding step is also visualized in Figure A1. Unlike the optimization process, rounding performs only once after the optimization iterations. It aims to move the optimized fractional result $\bar{y}$ from inside the polytope P_2 to y^*, a vertex of the polytope. The optimized summary set is (1,1,0) in Figure A1.

Appendix C

Below is the proof of the projection time complexity.

Leveraging the Lagrange multiplier $\lambda \ge 0$, we can rewrite the expression of y^{t+1} as shown below:

$$y^{t+1} = \arg\min_{x \in [0,1]^n} \frac{1}{2}\left\|x - y^{t+1/2}\right\|_2^2 + \lambda\left(\sum x_i - k\right). \quad (8)$$

Solving the above function with regard to x gives for each $i \in [n]$:

$$x_i - y_i^{t+1/2} + \lambda = 0. \quad (9)$$

Then the optimal solution is: $y_i^{t+1} = y_i^{t+1/2} - \lambda$ and $y_i^{t+1} \in [0,1]$. That is $y_i^{t+1} = m\left(0, y_i^{t+1/2} - \lambda, 1\right)$. The function $m(\cdot)$ returns the median of inputs as shown below:

$$m\left(0, y_i^{t+1/2} - \lambda, 1\right) = \begin{cases} 0, & \text{if } y_i^{t+1/2} - \lambda < 0 \\ y_i^{t+1/2} - \lambda, & \text{if } y_i^{t+1/2} - \lambda \in [0,1]. \\ 1, & \text{if } y_i^{t+1/2} - \lambda > 1 \end{cases} \quad (10)$$

Now let us compute the value of λ. For each $\lambda \in$, define:

$$f(\lambda) := \sum_{i=0}^{n-1} \min\left\{\max\left\{y_i^{t+1/2} - \lambda, 0\right\}, 1\right\}. \quad (11)$$

Note that $f(\lambda) = 0$ if $\lambda > y_i^{t+1/2}$ if $\lambda > y_i^{t+1/2}$. Since $f(\lambda)$ decreases as λ increases, without loss of generality, we assume that $y_0^{t+1/2} \le y_1^{t+1/2} \le \cdots \le y_n^{t+1/2}$ is a non-decreasing list. By the method of linear interpolation, we know that:

$$\lambda = \frac{\sum_{i=i_0}^{n-1} y_i^{t+1/2} - k}{n - i_0}, \quad (12)$$

where i_0 is the maximum index i such that $f(y_0^{t+1/2}) \ge k$. The sorting operation can be performed within the time O(n log n), and the linear interpolation can be solved in O(n). Thus, the overall time complexity is O(n log n).

Appendix D

The detailed rounding methodology is presented here. For a given fractional $\bar{y}$ and $i,j \in N$, we define $y_{i,j}(\varepsilon)$ as the vector obtained by adding ε to y_i and subtracting ε from y_j and leaving the other values unchanged. We let $\varepsilon^- = (-1)*\min\{y_i, 1-y_j\}$ and $\varepsilon^+ = \min\{1-y_i, y_j\}$. The detailed rounding methodology is presented in Algorithm A1 below.

Algorithm A1 Deterministic Pipage Rounding for SP3D with Cardinality Constraint

Require: SP3D concave extension $\bar{S}$, fractional $\bar{y}$.

Ensure: integral y^*

While $\bar{y}$ fractional **do**

Select fractional i,j from $\bar{y}$

If $\bar{S}\left(y_{i,j}\left(\varepsilon^{+}\right)\right) > \bar{S}\left(y_{i,j}\left(\varepsilon^{-}\right)\right)$ **then**

$y^{*} = y_{i,j}\left(\varepsilon^{+}\right)$

else

$y^{*} = y_{i,j}\left(\varepsilon^{-}\right)$

end if

end while

Appendix E

We prove that our proposed rounding method can be finished in O(n) time.

Proof. The proof is similar to the one in [29], modified for the deterministic pipage rounding rather than the randomized rounding. The inequality $E\left(\bar{S}\left(y^{*}\right)\right) \geq \bar{S}\left(\bar{y}\right)$ holds as the SP3D is monotone non-decreasing and in each step we takes the more profitable direction.

Appendix F

Here is the proof of Proposition 2.

Proposition 2. Let function F be the SP3D network for video summarization with representativeness and diversity functions defined in Equations (6) and (7). The concave extension is $\bar{F}:[0,1]^{n} \rightarrow R$ and k is the cardinality constraint. For each $0 < \varepsilon < 1$, Algorithm 1 will produce the fractional solution $\bar{y}$ such that $\bar{S}\left(\bar{y}\right) \geq \left(1-\varepsilon\right)\max_{y \in M} \bar{S}\left(y\right)$ with running time $O\left(kn\varepsilon^{-2}\right)$.

Proof. We begin by bounding the value of R^2. In our case, the polytope is a family of the subsets with rank k. According to the definition of R^2 in Theorem 1, $R^2 = \sup_{y \in M} \frac{1}{2}\|y\|_2^2 = \frac{k}{2}$ as the vertex of the polytope y_v has exactly k ones, i.e., $y_v \in (0,1)^n$ and $\sum_{i=1}^{n} y_{v_i} = k$. Next, we bound the value of B^2. Since $\bar{F}$ is concave, the supergradient of $\bar{F}$ is $g(y_0)_e$, where y_0 is the smallest non-negative vector on the polytope. For arbitrary coordinates e_1, e_2, $\left\|\frac{g(y_0)_{e_1}}{g(y_0)_{e_2}}\right\| \leq \frac{\omega_{max}}{\omega_{min}}$. So $B^2 = \sup_{y \in M} \frac{1}{2}\|g(y)\|_2^2 \leq \frac{\omega_{max}}{\omega_{min}} ng(y_0)_e$ [2].

The upper bound of $g(y_0)_e$ is closely related to the input video x_v and the submodular functions used in the model. The submodular functions Equation (6), and (7) are based on Euclidean distance. For the representativeness functions $f_{rep}(S)$, the supergradient is $(-1)\Delta x_{min} y_0$ where Δx_{min} is the minimum difference between the feature values of video V. For the diversity functions $f_{div}(S)$ the supergradient is $\Delta x_{max} y_0$, where Δx_{max} is the maximum difference between the feature values.

Then in Theorem 1, $R^2 = \frac{k}{2}$ and $B^2 \leq \frac{\omega_{max}}{\omega_{min}} ng(y_0)_e$, where $g(y_0)_e = \Delta x_{max} y_0 - \Delta x_{min} y_0$, ω_{max} and ω_{min} are the largest and smallest element in the first weight layer, respectively. Since ω_{max}, ω_{min} and $g(y_0)_e$ do not change the computation time, the time complexity of B^2 is O(n). Therefore, the overall running time is $O(kn\varepsilon^{-2})$.

Data Sharing Agreement

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Declaration of Conflicting Interests

The authors declared no potential conflicts of interest with respect to the research, author-ship, and publication of this article.

Funding

The authors received no financial support for the research.

References

1. Apostolidis, E., Adamantidou, E., Metsai, A.I., Mezaris, V., Patras, I. Video Summarization Using Deep Neural Networks: A Survey. Proceedings of the IEEE, 2021, 109(11), 1838-1863. https://doi.org/10.1109/JPROC.2021.3117472
2. Bai, W., Noble, W.S., Bilmes, J.A. Submodular Maximization via Gradient Ascent: The Case of Deep Submodular Functions. Advances in Neural Information Processing Systems, 2018, 31, 7989.
3. Banihashem, K., Biabani, L., Goudarzi, S., Hajiaghayi, M., Jabbarzade, P., Monemizadeh, M. Dynamic Non-Monotone Submodular Maximization. Advances in Neural Information Processing Systems, 2023, 36, 17369-17382.
4. Bilmes, J. Submodularity in Machine Learning and Artificial Intelligence. arXiv preprint arXiv:2202.00132, 2022. https://doi.org/10.48550/arXiv.2202.00132
5. Bilmes, J., Bai,W. Deep Submodular Functions. arXiv preprint arXiv:1701.08939, 2017. https://doi.org/10.48550/arXiv.1701.08939
6. Bubeck, S. Convex Optimization: Algorithms and Complexity. Foundations and Trends® in Machine Learning, 2015, 8(3-4), 231-357. https://doi.org/10.1561/2200000050
7. Buchbinder, N., Feldman, M. Constrained Submodular Maximization via New Bounds for DR-Submodular Functions. In Proceedings of the ACM Symposium on the Theory of Computing, 2024, 1820-1831. https://doi.org/10.1145/3618260.3649630
8. Calinescu, G., Chekuri, C., Pal, M., Vondrák, J. Maximizing a Monotone Submodular Function Subject to a Matroid Constraint. SIAM Journal on Computing, 2011, 40(6), 1740-1766. https://doi.org/10.1137/080733991
9. Cevallos, A., Eisenbrand, F., Zenklusen, R. Local Search for Max-Sum Diversification. In Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, 2017, 130-142. https://doi.org/10.1137/1.9781611974782.9
10. Chen, M., Zhang, Y., Wei, J., Zhang, Y., Feng, R., Zhang, T., Gao, S. Temporal Feature Aggregation for Efficient 2D Video Grounding. In Proceedings of the IEEE International Conference on Multimedia and Expo, 2024, 1-6. https://doi.org/10.1109/ICME57554.2024.10687387
11. Chen, Q., Im, S., Moseley, B., Xu, C., Zhang, R. Min-Max Submodular Ranking for Multiple Agents. In Proceedings of the AAAI Conference on Artificial Intelligence, 2023, 37(6), 7061-7068. https://doi.org/10.1609/aaai.v37i6.25862
12. Cui, S., Han, K., Tang, S., Li, F., Luo, J. Fairness in Streaming Submodular Maximization Subject to a Knapsack Constraint. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining, 2024, 514-525. https://doi.org/10.1145/3637528.3671778
13. Dang, C., Radha, H. RPCA-KFE: Key Frame Extraction for Video Using Robust Principal Component Analysis. IEEE Transactions on Image Processing, 2015, 24(11), 3742-3753. https://doi.org/10.1109/TIP.2015.2445572
14. De Avila, S.E.F., Lopes, A.P.B., da Luz Jr, A., de Albuquerque Araújo, A. VSUMM: A Mechanism Designed to Produce Static Video Summaries and a Novel Evaluation Method. Pattern Recognition Letters, 2011, 32(1), 56-68. https://doi.org/10.1016/j.patrec.2010.08.004
15. Ding, Q., Liu, Y., Miao, C., Cheng, F., Tang, H. A Hybrid Bandit Framework for Diversified Recommendation. In Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(5), 4036-4044. https://doi.org/10.1609/aaai.v35i5.16524
16. Dolhansky, B., Bilmes, J. Deep Submodular Functions: Definitions and Learning. Advances in Neural Information Processing Systems, 2016, 29, 3404-3412.
17. Du, Q., Yu, L., Li, H., Ou, N., Gong, X., Xiang, J. M3Rec: Cross-Modal Context Enhanced Micro-Video Recommendation with Mutual Information Maximization. In Proceedings of the IEEE International Conference on Multimedia and Expo, 2022, 1-6. https://doi.org/10.1109/ICME52920.2022.9859663
18. Dughmi, Shaddin. Submodular Functions: Extensions, Distributions, and Algorithms. arXiv preprint arXiv: 0912.0322, 2009. https://doi.org/10.48550/arXiv.0912.0322
19. Elhamifar, E., Clara De Paolis Kaluza, M. Online Summarization via Submodular and Convex Optimization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2017, 1783-1791.https://doi.org/10.1109/CVPR.2017.197
20. Fajtl, J., Sokeh, H.S., Argyriou, V., Monekosso, D., Remagnino, P. Summarizing Videos with Attention. In Proceedings of the Asian Conference on Computer Vision, 2018, 11367, 39-54. https://doi.org/10.1007/978-3-030-21074-8_4
21. Fisher, M.L., Nemhauser, G.L., Wolsey, L.A. An Analysis of Approximations for Maximizing Submodular Set Functions-II. In Polyhedral combinatorics, Springer, 1978, 8, 73-87. https://doi.org/10.1007/BFb0121195

22. Fujishige, S. Submodular Functions and Optimization, Elsevier, 2005, 58.

23. Gygli, M., Grabner, H., Riemenschneider, H., Nater, F., Van Gool, L. The Interestingness of Images. In Proceedings of the IEEE International Conference on Computer Vision, 2013, 1633-1640. https://doi.org/10.1109/ICCV.2013.205

24. Gygli, M., Grabner, H., Riemenschneider, H., Van Gool, L. Creating Summaries from User Videos. In Proceedings of the European Conference on Computer Vision, 2014, 8695, 505-520. https://doi.org/10.1007/978-3-319-10584-0_33

25. Gygli, M., Grabner, H., Van Gool, L. Video Summarization by Learning Submodular Mixtures of Objectives. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2015, 3090-3098. https://doi.org/10.1109/CVPR.2015.7298928

26. Harvey, N.J., Olver, N. Pipage Rounding, Pessimistic Estimators and Matrix Concentration. In Proceedings of the ACM-SIAM Symposium on Discrete Algorithms, 2014, 926-945. https://doi.org/10.1137/1.9781611973402.69

27. Hassani, H., Soltanolkotabi, M., Karbasi, A. Gradient Methods for Submodular Maximization. Advances in Neural Information Processing Systems, 2017, 30.

28. K J, J., Vamshi Teja, R., Krishnakant, S., Vineeth, N.B. Submodular Batch Selection for Training Deep Neural Networks. In Proceedings of the International Joint Conference on Artificial Intelligence, 2019, 2677-2683. https://doi.org/10.24963/ijcai.2019/372

29. Karimi, M.R., Lucic, M., Hassani, H., Krause, A. Stochastic Submodular Maximization: The Case of Coverage Functions. Advances in Neural Information Processing Systems, 2017, 30, 6856-6866.

30. Kothawade, S., Kaushal, V., Ramakrishnan, G., Bilmes, J., Iyer, R. Prism: A Rich Class of Parameterized Submodular Information Measures for Guided Data Subset Selection. In Proceedings of the AAAI Conference on Artificial Intelligence, 558, 2022, 36(9), 10238-10246. https://doi.org/10.1609/aaai.v36i9.21264

31. Krause, A., Golovin, D. Submodular Function Maximization. Tractability, 2014, 3, 71-104. https://doi.org/10.1017/CBO9781139177801.004

32. Kulesza, A., Taskar, B. Determinantal Point Processes for Machine Learning. arXiv preprint arXiv:1207.6083, 2012. https://doi.org/10.1561/9781601986290

33. Kumari, L., Bilmes, J. Submodular Span, with Applications to Conditional Data Summarization. In Proceedings of the AAAI Conference on Artificial Intelligence, 2021, 35(14), 12344-12352. https://doi.org/10.1609/aaai.v35i14.17465

34. LeCun, Y., Bengio, Y., Hinton, G. Deep learning. Nature, 2015, 521(7553), 436-444. https://doi.org/10.1038/nature14539

35. Li, W., Feldman, M., Kazemi, E., Karbasi, A. Submodular Maximization in Clean Linear Time. Advances in Neural Information Processing Systems, 2022, 35, 17473-17487.

36. Li, X., Yuan, K., Pei, Y., Lu, Y., Sun, M., Zhou, C., Chen, Z., Timofte, R., Sun, W., Wu, H., et al. NTIRE 2024 Challenge on Short-Form UGC Video Quality Assessment: Methods and Results. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, 6415-6431. https://doi.org/10.1109/CVPRW63382.2024.00643

37. Li, X., Zhao, B., Lu, X. A General Framework for Edited Video and Raw Video Summarization. IEEE Transactions on Image Processing, 2017, 26(8), 3652-3664. https://doi.org/10.1109/TIP.2017.2695887

38. Lin, H., Bilmes, J. A Class of Submodular Functions for Document Summarization. In Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2011, 510-520.

39. Liu, K., Liu, C., Yan, G., Lee, V. C., Cao, J. Accelerating DNN Inference with Reliability Guarantee in Vehicular Edge Computing. IEEE/ACM Transactions on Networking, 2023, 31(6), 3238-3253. https://doi.org/10.1109/TNET.2023.3279512

40. Liu, Y., Wu, J., Li, L., Dong, W., Shi, G. Quality Assessment of UGC Videos Based on Decomposition and Recomposition. IEEE Transactions on Circuits and Systems for Video Technology, 2022, 33(3), 1043-1054. https://doi.org/10.1109/TCSVT.2022.3209007

41. Ma, C., Lyu, L., Lu, G., Lyu, C. Adaptive Multiview Graph Difference Analysis for Video Summarization. IEEE Transactions on Circuits and Systems for Video Technology, 2022, 32(12), 8795-8808. https://doi.org/10.1109/TCSVT.2022.3190998

42. Mirzasoleiman, B., Karbasi, A., Krause, A. Deletion-Robust Submodular Maximization: Data Summarization with "the Right to be Forgotten". In Proceedings of the International Conference on Machine Learning, 2017, 70, 2449-2458.

43. Murphy, K. P. Probabilistic Machine Learning: Advanced Topics, MIT press, 2023.

44. Nemhauser, G. L., Wolsey, L. A., Fisher, M. L. An Analysis of Approximations for Maximizing Submodular Set Functions-I. Mathematical programming, 1978, 14(1), 265-294. https://doi.org/10.1007/BF01588971

45. Open video project. http://www.open-video.org/.

46. Otani, M., Nakashima, Y., Rahtu, E., Heikkila, J. Rethinking the Evaluation of Video Summaries. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, 7596-7604. https://doi.org/10.1109/cvpr.2019.00778

47. Rochan, M., Ye, L., Wang, Y. Video Summarization Using Fully Convolutional Sequence Networks. In Proceedings of the European Conference on Computer Vision, 2018, 11216, 347-363. https://doi.org/10.1007/978-3-030-01258-8_22

48. Song, Y., Vallmitjana, J., Stent, A., Jaimes, A. TVSum: Summarizing Web Videos Using Titles. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2015, 5179-5187. https://doi.org/10.1109/CVPR.2015.7299154

49. Tolstikhin, I. O., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., Dosovitskiy, A. MLP-Mixer: An All-MLP Architecture for Vision. Advances in Neural Information Processing Systems, 2021, 34, 24261-24272.

50. Tran, D., Wang, H., Torresani, L., Ray, J., LeCun, Y., Paluri, M. A Closer Look at Spatiotemporal Convolutions for Action Recognition. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, 6450-6459. https://doi.org/10.1109/CVPR.2018.00675

51. Tschiatschek, S., Iyer, R. K., Wei, H., Bilmes, J. A. Learning Mixtures of Submodular Functions for Image Collection Summarization. Advances in Neural Information Processing Systems, 2014, 27, 1413-1421.

52. Tu, G., Fu, Y., Li, B., Gao, J., Jiang, Y. G., Xue, X. A Multi-Task Neural Approach for Emotion Attribution, Classification, and Summarization. IEEE Transactions on Multimedia, 2019, 22(1), 148-159. https://doi.org/10.1109/TMM.2019.2922129

53. Vivekraj, V., Debashis, S., Raman, B. Video Skimming: Taxonomy and Comprehensive Survey. ACM Computing Surveys, 2019, 52(5), 1-38. https://doi.org/10.1145/3347712

54. Xie, J., Chen, X., Lu, S.P. An Aesthetic-Guided Multimodal Framework for Video Summarization. In Proceedings of the IEEE International Conference on Multimedia and Expo, 2024, 1-6. https://doi.org/10.1109/ICME57554.2024.10687758

55. Xu, J., Mukherjee, L., Li, Y., Warner, J., Rehg, J. M., Singh, V. Gaze-Enabled Egocentric Video Summarization via Constrained Submodular Maximization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2015, 2235-2244. https://doi.org/10.1109/CVPR.2015.7298836

56. Zhang, H., Yu, P. S., Zhang, J. A Systematic Survey of Text Summarization: From Statistical Methods to Large Language Models. ACM Computing Surveys, 2024, 57(11), 1-41. https://doi.org/10.1145/3731445

57. Zhang, K., Chao, W. L., Sha, F., Grauman, K. Summary Transfer: Exemplar-Based Subset Selection for Video Summarization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016, 1059-1067. https://doi.org/10.1109/CVPR.2016.120

58. Zhang, K., Chao, W. L., Sha, F., Grauman, K. Video Summarization with Long Short-Term Memory. In Proceedings of the European Conference on Computer Vision, 2016, 9911, 766-782. https://doi.org/10.1007/978-3-319-46478-7_47

59. Zhang, Z., Wu, W., Sun, W., Tu, D., Lu, W., Min, X., Chen, Y., Zhai, G. MD-VQA: Multi-Dimensional Quality Assessment for UGC Live Videos. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, 1746-1755. https://doi.org/10.1109/CVPR52729.2023.00174

60. Zhao, B., Li, H., Lu, X., Li, X. Reconstructive Sequence-Graph Network for Video Summarization. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2021, 44(5), 2793-2801. https://doi.org/10.1109/TPAMI.2021.3072117

61. Zhao, D., Zhu, D., Min, X., Yue, J., Zhang, K., Zhou, Q., Zhai, G., Yang, X. Human Attention Based Movie Summarization: Dataset and Baseline Model. Neurocomputing, 2023, 534, 106-118. https://doi.org/10.1016/j.neucom.2023.03.013

62. Zhou, K., Qiao, Y., Xiang, T. Deep Reinforcement Learning for Unsupervised Video Summarization with Diversity-Representativeness Reward. In Proceedings of the AAAI Conference on Artificial Intelligence, 2018, 32(1), 7582-7589. https://doi.org/10.1609/aaai.v32i1.12255

63. Zhu,W., Lu, J., Li, J., Zhou, J. DSNet: A Flexible Detect-to-Summarize Network for Video Summarization. IEEE Transactions on Image Processing, 2020, 30, 948-962. https://doi.org/10.1109/TIP.2020.3039886