Learning Video-Story Composition via Recurrent Neural Network

Guangyu Zhong[1,2] Yi-Hsuan Tsai[3] Sifei Liu[4] Zhixun Su[1] Ming-Hsuan Yang[2]
[1]School of Mathematical Sciences, Dalian University of Technology
[2]Electrical Engineering and Computer Science, University of California, Merced
[3]NEC Labs America [4]NVIDIA Research

Abstract

In this paper, we propose a learning-based method to compose a video-story from a group of video clips that describe an activity or experience. We learn the coherence between video clips from real videos via the Recurrent Neural Network (RNN) that jointly incorporates the spatial-temporal semantics and motion dynamics to generate smooth and relevant compositions. We further rearrange the results generated by the RNN to make the overall video-story compatible with the storyline structure via a submodular ranking optimization process. Experimental results on the video-story dataset show that the proposed algorithm outperforms the state-of-the-art approach.

1. Introduction

Nowadays people are able to capture and store more and more personal experiences and memories in videos with the decreasing cost of cameras and storages. To organize these captured videos, they are usually edited and processed to be a concise format at a later time. Since the manual post-processing is time-consuming and labor-intensive, automatic algorithms are developed to process these unorganized videos, e.g., generation of a "short story" from a collection of videos [6]. In this work, we adopt this problem setting and aim to composite a smooth and meaningful video-story from video clips.

Specifically, we describe our task as: given a set of clips taken by a person during an activity or experience, we find out an order of the clips which composes a story containing smooth transitions in terms of semantics, motions, and activity dynamics that match the storyline structures [6] (see Figure 1). Note that, different from the video summarization task that aims to select keyframes out of a long video [32, 34, 21, 17], the "story composition" problem described in this paper considers transitions between selected sub-shots and produces the consistent story in the temporal domain.

Recently, numerous methods address the temporal consistency problem by identifying temporal alignments [1, 15], storyline graph [15] or learning temporal relations [24] from images, in order to make the story more meaningful. However, most temporal alignment based methods suffer from two difficulties in practice. First, the results may look incoherent when the story is extracted from multiple video clips taken at different times. Second, the ambiguous scene transition of shots also affects the overall quality of the composition.

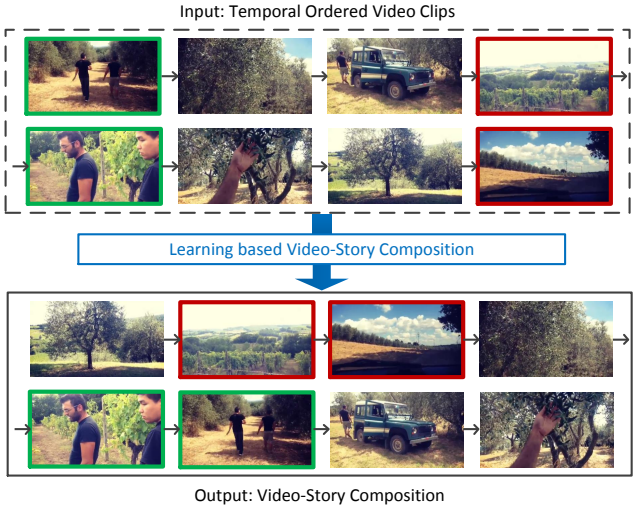


Figure 1: Learning video-story compositions. Given a set of temporally ordered video clips, our method learns to reorder and compose the clips into a coherent video-story that matches the storyline structures. For example, given the video clips taken by a person while walking in the garden, the proposed method reorders the temporally ordered one and generates the coherent video-story, where the video clips with similar scenes (red rectangles) or contents (green rectangles) are connected together.

In order to solve the above challenge, hand-crafted features can be used to represent the relations between video clips. State-of-the-art method [6] uses the color based

bidirectional similarity to describe the relations between video clips and the dense optical flow to generate dynamics scores for each clip. The video sequence order is then formulated and generated via a branch-and-bound algorithm [12]. However, the coherence between clips is built directly through feature matching, which is likely to fail in the cases when there are ambiguous appearances or interrupted motions.

In contrast, we address this problem by modeling the coherence of adjacent clips through a learning-based recurrent network. Our network learns how to select the next connected clip from the remaining set of video clips based on previous selections in the temporal domain. Specifically, we train the two-stream RNN, including a semantic RNN that uses the spatial-temporal features, and a motion RNN that exploits the motion dynamics in each video clip. To train this network, a generated initial clip is fed into the streams, and two output probabilities are jointly fused as the coherence scores between video clips to predict the next clip.

We further rearrange the probabilities from the two-stream RNN by a submodular ranking process to align with the storyline structure, which consists of the exposition, rising, action, climax, and resolution. Generally, the storyline structure ensures that video-story contains rising dynamics and has an ending with more activity than its beginning to attract the viewers [6]. Finally, we compose the video-story by solving this submodular ranking optimization.

We demonstrate the effectiveness of the proposed learning based video-story composition algorithm on the benchmark dataset [6]. We conduct a user study via Amazon Mechanical Turk to evaluate the overall video-story quality and quantitatively verify the composition quality based on pairwise annotations. Overall, our experimental results show that the proposed learning based algorithm performs favorably against the state-of-the-art methods in terms of visually quality and accuracy.

The main contributions of this work are summarized as follows. First, we propose a novel learning-based framework via the two-stream RNN for video-story composition. Second, we show that the proposed model explicitly learns better representations to model the coherence between video clips. Third, we develop a submodular ranking algorithm to improve the video-story composition results that better match the storyline structure.

2. Related Work

Video Summarization. As introduced in Section 1, although having different goals, the technical aspects of video summarization are quit similar and can be sufficiently utilized by video composition. Many video summarization approaches have been proposed via different image-based feature representations and optimization methods, either through low-level feature such as optical flow [32] and image differences [34], or high-level representations, including object trackers [21] and importance scores [17]. On the other hand, subshot-based methods represent summarizations via spatio-temporal features [19]. Numerous supervised approaches select the subshots to represent the videos based on submodular function [10] and exemplas [35]. All these methods require ground truths for training.

However, the labeling of the ground truth for either video summarization or video caption is too subjective and difficult as a consistent limitation to the above methods. In contrast, our model is learned in an unsupervised manner, making the framework more flexible to utilize large amount of data to improve the performance.

Story Composition. The story composition methods typically focus on identifying the temporal alignment of the image sets (photo albums). Basha et al. [1] use static and dynamic features to find the temporal order of the image sequence. Kim et al. [15] learn the pairwise transition to construct the storyline graphs. Recently, an unsupervised method proposed by Sigurdsson et al. [24] use a skipping Recurrent Neural Network to learn long-term correlations.

In contrast, our approach focuses on compositing video clips rather than images. The video clips contain significant dynamics and ambiguity in terms of semantics and motions, thereby resulting in more challenging scenarios. In this work, we aim to rank all the video clips and compose a coherent story rather than selecting a subset of images or videos. We note that our approach is closest related to the plot analysis based method [6]. Instead of solving this problem via hand-crafted features, we learn the coherence between clips from real videos.

Learning Temporal Representations. Temporal representations have been used in many tasks in language analysis [22, 26] and computer vision [27, 24]. Recurrent neural networks are used in language modeling [22] and text generation tasks [26] to analyze the temporal information across time steps and generate future contents. Sigurdsson et al. [24] extend this idea by modeling long-term memories to represent each story topic. On the other hand, spatial-temporal information such as C3D features [27] are used in video analysis tasks [33, 23]. These features describe the temporal representation for activities through a set of video frames. In this paper, we utilize these representations but focus on analyzing relations between video clips that contain various topics (e.g., different scenes or objects).

3. Learning Video-Story Composition

3.1. Overview

Given a set of individual video clips, our goal is to compose the clips as a video-story which meets two criteria: (1) the semantic and motion transitions of the connected clips

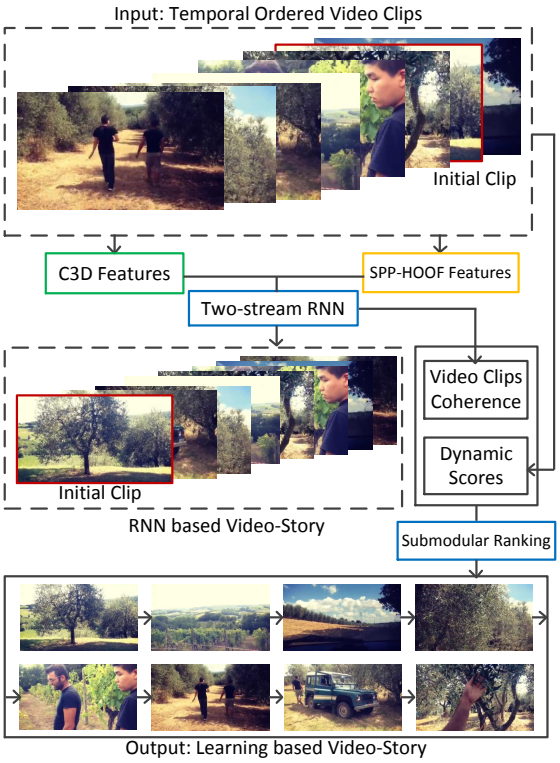


Figure 2: Overview of the proposed algorithm. We first feed the initial clip (in red rectangles) into the two-stream RNN. The output probabilities are then used as coherence scores between video clips to predict the next clip and generate the video-story. To further refine the results and match the storyline structure, we rearrange the composition order by solving a submodular ranking optimization via the learned coherence and activity dynamics of video clips.

are coherent and smooth; (2) the composed video follows the storyline structure. To achieve this, we first learn the coherence between video clips by training RNNs in an unsupervised manner. We train a two-steam RNN with clip representations of the C3D features [27] and optical flow. Then the probabilities generated in each RNN are jointly fused and learned to output the coherence score between clips. To make the video-story match the storyline structure, we further model the video composition task as a ranking problem via a submodular optimization function guided by the learned coherence and activity dynamics of video clips. Figure 2 shows the main steps of the proposed algorithm.

3.2. Learning Video Coherence via RNNs

Recurrent Neural Networks. The RNN [8] can be used to process sequential data of a input video, which meets the need of the formulated task that aims to sequentially predict the next clip given previous contents. Based on a series of T items $c_{1:T} = \{c_1, ..., c_T\}$, where each c represents a clip, the network is trained to predict the next clip by maximizing the log-likelihood:

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \sum_{t}^{T-1} \log P(c_{t+1}|c_{1:t}; \theta), \tag{1}$$

where θ indicates all the parameters in the model. We use the back propagation through time method [31, 30] to optimize the model.

The RNN model consists of the input, recurrent, and hidden layers. At the t-th time step, the output feature y_t is computed as follows:

$$\begin{aligned} h_t &= \sigma_h(W_I c_t + W_H h_{t-1}), \\ y_t &= \sigma_y(W_O h_t). \end{aligned} \tag{2}$$

The input c_t is used to update the hidden recurrent layer h_t with the weights W_I, and the hidden layer updates itself using the weights W_H. Then the output y_t is generated via weights W_O and non-linear activation functions σ_h and σ_y.

Loss Function. Considering that different topics may appear in video clips, it is not trivial to obtain video composition ground truths, we formulate an unsupervised learning task using the temporal order in real videos. We first split the entire video into several video clips, and each clip contains fixed length of frames. To avoid the vanishing gradient problem [2] introduced by training the long-term data, we randomly select a continuous subset of clips for training rather than directly using the entire set. Note that the length of input video clips $c_{1:T}$ is fixed.

Based on the previous chosen t clips $c_{1:t}$, we wish to choose the next clip c_{t+1} from the remaining clips $\mathcal{C}_t = c_{t+1:T}$ using maximum likelihood. We define the probability of an unselected clip $c_\tau \in \mathcal{C}_t$ being the next selected clip c_{t+1} using the soft-max function over the inner product of the network output y_t and the feature vector of c_τ:

$$P(c_\tau = c_{t+1}|c_{1:t}; \theta) = \frac{\exp(y_t^\top c_\tau)}{\sum_{c \in \mathcal{C}_t} \exp(y_t^\top c)}. \tag{3}$$

Different from the standard language model [22] that directly generates representations of the next clip, this formulation selects the best matching item among the remaining ones. We illustrate the architecture of our model in Figure 3.

Representation for Video Clips. To describe the spatial-temporal information within each clip, the C3D features [13, 27] have been effectively used to represent the clip. Specifically, we utilize a C3D model pre-trained on the Sports-1M video dataset [14], and extract features form the fc7 fully-connected layer as the representation f for an input clip c.

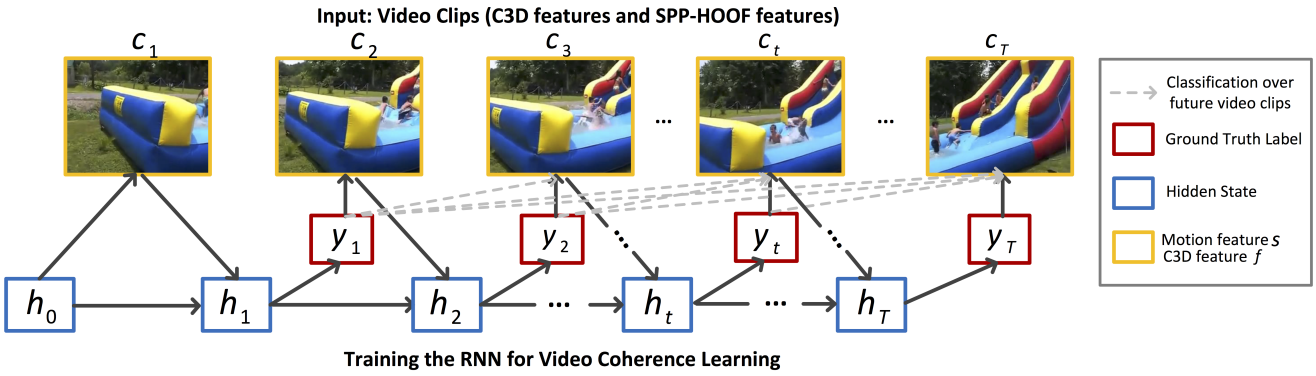


Figure 3: Illustration of the proposed RNN for learning the coherence between clips, where the hidden layer preserves the information from previous states. In the RNN model, we fix the length of training data as T and treat the problem as a classification task with the soft-max function to predict the next frame from the remaining clips. Note that, we use the same architecture for our two-stream framework, where the inputs are C3D and SPP-HOOF features, respectively.

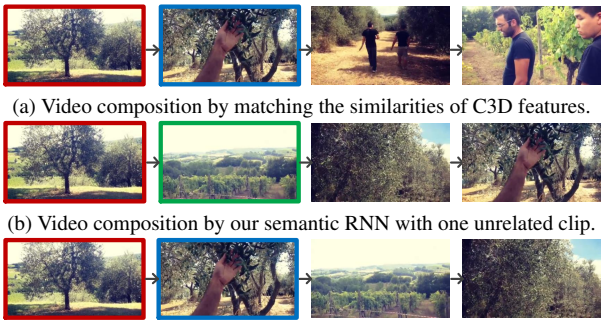
To describe the motion dynamics in each video clip, we first extract the dense optical flow [20] from each frame, which is shown to provide effective representations for action recognition [7]. Then we compute the histogram of dense optical flow (HOOF) [5] to generate a feature vector. Since the motions in each frame may vary significantly at different locations (e.g., the background scenes usually contain fewer actions compared with the foreground objects), we further adopt the spatial pyramid pooling (SPP) [11] on the optical flow to generate an SPP-HOOF feature for each frame. Given a video clip c with l frames and the pyramid level $\{M \times M\}$, the SPP-HOOF feature s^k in the k-th frame is defined as: $s^k = [h^{k_1}, ..., h^{k_{(M\times M)}}, h^k]$. We then normalize our SPP-HOOF motion features in clip c as $s = \frac{1}{l}\sum_{k=1}^{l} s^k$.

Learning Coherence between Clips. Motivated by [25], we train the two-steam RNN that accounts for semantics and motions using the above-mentioned representations for clips (i.e., C3D features f and SPP-HOOF features s). To train this network, an initial clip is required. Since a good video story that matches the storyline structure usually starts with a clip that contains fewer motions [6], we compute a dynamics score ϕ as the average magnitude of the optical flow in each clip and select the one with the smallest score as the initial clip.

Given N video clips, at each training step t, we fuse the output probabilities from the two streams to describe the coherence between previously ordered clips $c_{1:t}$ and the remaining ones $\mathcal{C}_t$. Thus the corresponding coherence vector $d(c_t, c)$ is defined as:

$$d(c_t, c) = \{\lambda P(f|f_{1:t}; \theta_f) + (1-\lambda)P(s|s_{1:t}; \theta_s), c \in \mathcal{C}_t\}, \quad (4)$$

where θ_f and θ_s are the parameters in the semantic and motion streams respectively, and λ is set to 0.5 for averaging the probabilities. We consider this process as our baseline method, in which the next video clip is the one with the highest coherence score.



(a) Video composition by matching the similarities of C3D features.

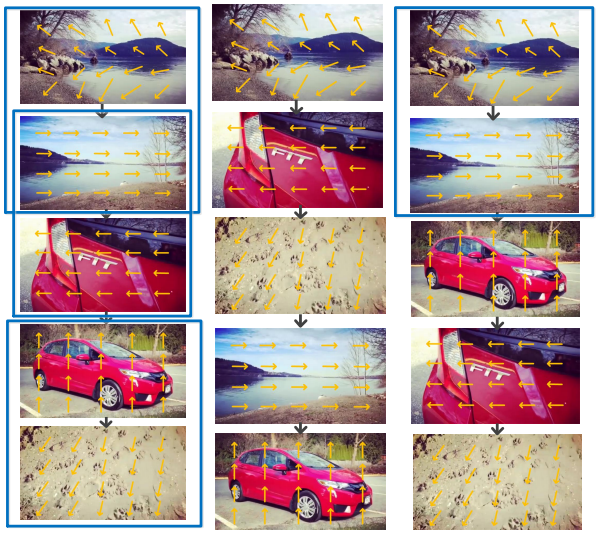
(b) Video composition by our semantic RNN with one unrelated clip.

(c) Video composition by our semantic RNN.

Figure 4: Effects of the coherence learned by RNN. Given the initial video clip (marked with red rectangle), we show the selected video clips in (a) using direct feature matching and (b) using our semantic RNN. It shows our RNN emphasizes holistic rather than local coherence in the temporal space, and results in a more consistent composition.

To validate the effectiveness of our learned coherence in terms of the semantics, we analyze the results generated by our semantic RNN in Figure 4. Figure 4 (c) shows that with our semantic RNN, the scene of forests (the red rectangle) is followed by similar scenes (e.g., the green rectangle in (c)) rather than unrelated scenes or activities (e.g, the blue rectangle in (a)), which provides smoother transitions. Furthermore, our transitions are robust. Even with one unrelated clip inserted (the blue rectangle in (b)), the story of following clips are not heavily affected by this clip due to the accumulated information learned by our semantic RNN, and thus the consistency of the whole story is kept.

In addition, without considering the motion stream, the



(a) Semantic RNN (b) Motion RNN (c) Two-stream RNN

Figure 5: Video composition by different RNNs. Given the same initializations, (a) and (b) are generated via the semantic RNN and motion RNN, respectively. (c) is generated by our baseline. The contents in (a) are consistent while the motion transitions change a lot, (e.g., the major motions of clips in the blue rectangles change about 180 degrees). The motion RNN provides better motion consistency in (b) while the adjacent contents are less related. Our two-stream RNN in (c) contains more consistency in terms of both semantics and motions. The major motion directions of each clip are shown by orange arrows.

results of the single semantic RNN may contain significant motion change across the adjacent clips that could potentially cause motion sickness. For example, the major motions of the composition by the semantic RNN in Figure 5 (a) flip (i.e., the motion direction of the camera changes almost 180 degrees) three times (marked in the blue rectangles) while the ones in Figure 5 (c) only flip once via merging the motion consistency. The quantitative results in Figure 8 further validate the effectiveness of our coherency.

3.3. Submodular Ranking

To ensure the video-story composition meets the storyline structure, we formulate a submodular optimization problem to select and rearrange video clips from the ordered set generated by the two-stream RNN. We first construct a graph where video clips are considered as nodes. We design a submodular objective function using the coherence and activity dynamics of video clips to describe the ideal video-story. The video-story result is then extracted by solving this proposed submodular function.

Graph Construction. Given a set of ordered video clips $\tilde{c}_{1:N}$ generated by our the proposed two-stream RNN, we construct a fully connected graph $G = (\mathcal{V}, \mathcal{E})$. Each element $v \in \mathcal{V}$ is a video clip from $\tilde{c}_{1:N}$ and the edge $e \in \mathcal{E}$ represents the pairwise relation between two clips. We aim to select all the nodes from $\mathcal{V}$ to $\mathcal{A}$. Then the selection ranks are considered as the composition order of video clips $\mathcal{A}$.

Submodular Function. We aim to select the video clip that meets two criteria: (1) sharing high coherence with other clips; (2) providing rising activity dynamics. The objective function is formulated with two terms, i.e., the facility location (FL) term to describe the coherence between candidate clips, and an activity dynamics (AD) term to represent the dynamics within each clip. We define the FL term as follows:

$$\mathcal{F}(\mathcal{A}) = \frac{1}{\mathcal{N}_{\mathcal{A}}} \sum_{v_i \in \mathcal{A}} \sum_{v_j \in \mathcal{V}} d(v_i, v_j), \tag{5}$$

where $\mathcal{N}_{\mathcal{A}}$ indicates the number of the selected facilities. In this function, $d(v_i, v_j)$ is defined as (4), which represents the pairwise relation between the candidate facility v_i and the previous selected element v_j. In addition, we formulate the AD term as:

$$\mathcal{U}(\mathcal{A}) = \sum_{v_i \in \mathcal{A}} \exp(-\phi_i), \tag{6}$$

where ϕ_i is the dynamics score of v_i defined in Section 3.2.

Optimization for Video Clips Ranking. We combine the FL and AD terms to formulate the submodular problem:

$$\begin{aligned} \max_{\mathcal{A}} \mathcal{L}(\mathcal{A}) &= \max_{\mathcal{A}} \mathcal{F}(\mathcal{A}) + \gamma \mathcal{U}(\mathcal{A}), \\ &\text{s.t. } \mathcal{A} \subseteq \mathcal{V}, \mathcal{N}_{\mathcal{A}} = N, \end{aligned} \tag{7}$$

where γ is the parameter to balance the contribution of two terms. The proposed submodular function ensures that the selected facilities share high coherence and maintain rising activity dynamics.

As the proposed objective function in (7) is the non-negative linear combination of two submodular terms, we solve it using a greedy algorithm similar to [37, 28, 36]. Since the video-story starts from the exposition with low activities, the facility set $\mathcal{A}$ is first initialized as the node v_1 that contains the lowest dynamics score. Then at the i-th iteration, we add the element $a \in \mathcal{V} \setminus \mathcal{A}^{i-1}$ which leads to the maximum energy gain $\mathcal{J}(A^i)$ into $\mathcal{A}$, where the energy gain is defined as: $\mathcal{J}(A^i) = \mathcal{L}(A^i) - \mathcal{L}(A^{i-1})$. We iteratively select the remaining elements until all the nodes in $\mathcal{V}$ have been selected. In addition, we use an evaluation form to speed up the optimization process as proposed in [18]. The process of submodular ranking is presented in Algorithm 1. Figure 6 shows the efficiency of our submodular ranking process. By rearranging the baseline results, the video composition contains rising dynamics scores and matches the storyline structure.

Algorithm 1 Optimization for Video Clips Ranking

Input: $G = (\mathcal{V}, \mathcal{E}), \mathcal{N}, \gamma$
Initialization: $\mathcal{A}^0 \leftarrow \{v_1\}, i \leftarrow 1$
loop
  $a^* = \arg\max_{\{\mathcal{A}^i \subseteq \mathcal{V}\}} \mathcal{J}(\mathcal{A}^i)$, where $\mathcal{A}^i = \mathcal{A}^{i-1} \cup a$
  if $\mathcal{N}_{\mathcal{A}} = \mathcal{N}$ **then**
    break
  end if
  $\mathcal{A}^i \leftarrow \mathcal{A}^{i-1} \cup a^*$,
  $i = i + 1$
end loop
Output: $\mathcal{A} \leftarrow \mathcal{A}^i$

4. Experimental Results

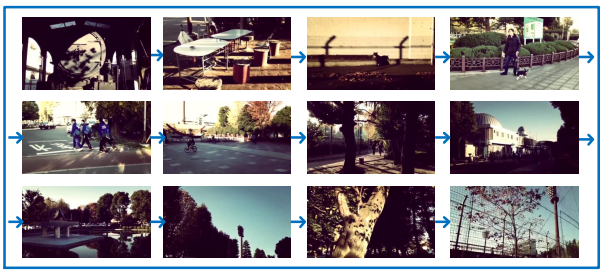
We evaluate the video-story composition results in this section. We first introduce the dataset and experimental details in Section 4.1 and then analyze the quantitative and qualitative results in Section 4.2.

4.1. Experimental Details

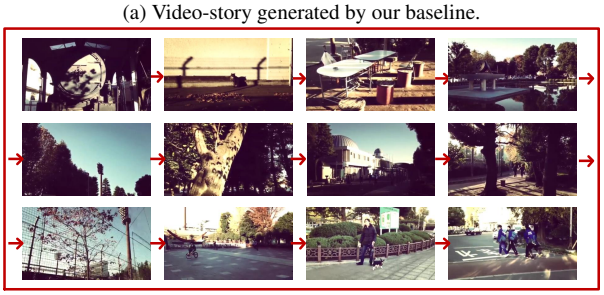
Datasets. We evaluate the proposed method on the video composition dataset [6] which consists of 23 video sets collected from YouTube. Each video set contains 8-12 video clips which last for 2-3 seconds, and the whole dataset has 236 video clips. The dataset contains rich activity contents (e.g., sightseeing, skateboarding, walking, surfing, shopping, driving, and swimming) in various scenes (e.g., river, park, ocean, streets, mall, landmarks, museum, marketplace, garden, and beach).

We train the models on the SumMe [9, 10] and TVSum [4] datasets that consist of 25 videos (with the average length of 160 seconds) and 50 videos (with the average length of 252 seconds) respectively. The training sets cover the activity contents of holidays, events and sports. The videos in the datasets have good consistency and are suitable for learning video coherence as formulated in the proposed RNN.

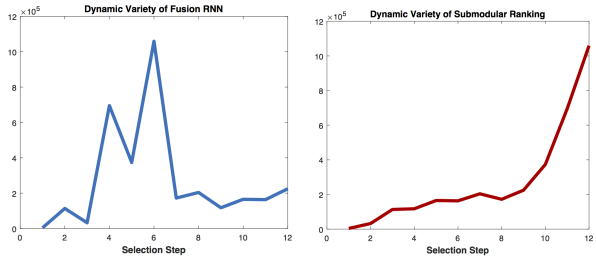
Experimental Settings. In the process of learning video coherence, considering the content varieties and lack of ground truths, we train the RNNs in an unsupervised manner. We use a fixed number (i.e., $T = 10$ in this work) of temporally continuous clips as a training sequence. Each item in the sequence is a video clip with 16 frames. The input of the semantic stream is a 4096-dimensional $fc7$ feature from the C3D model. To describe motion contents, we set the bin number of the HOOF feature as 10 and set the pyramid level as $\{3 \times 3\}$, resulting in the input size as 100. The hidden recurrent layer size is set to 100. Both the C3D and SPP-HOOF features are directly fed into the models.



(a) Video-story generated by our baseline.



(b) Video-story generated by the proposed method.



(c) Dynamics scores of the composed video-story for different methods.

Figure 6: (a) and (b) are video-stories generated by the baseline (i.e., two-stream RNN) and proposed method (i.e., two-stream RNN + submodular ranking) where the composition orders are shown by arrows. (c) Dynamics scores of (a) (left) and the proposed method (b) (right). After rearranging the results from (a), our video-story composition result (b) maintains rising dynamics.

We set σ_h and σ_y as the activation function of the rectified linear unit [16] and set the momentum of the gradient ascent as 0.9. We start the training process with the learning rate as 0.05, and gradually reduce it till the likelihood no longer increases with the weight decay $\lambda = 10^{-7}$. During the test phase, we initialize the RNN with the first video clip that has the lowest dynamics score as mentioned in Section 3.2 and set $\gamma = 0.3$ in (7).

4.2. Evaluation Results

We evaluate the video-story composition results in global and local aspects, i.e., overall video-story quality and component coherency. We first analyze the efforts of the two streams (i.e., the semantic RNN and the motion RNN) in our framework. We then compare the proposed method (i.e, two-stream RNN + submodular ranking) and

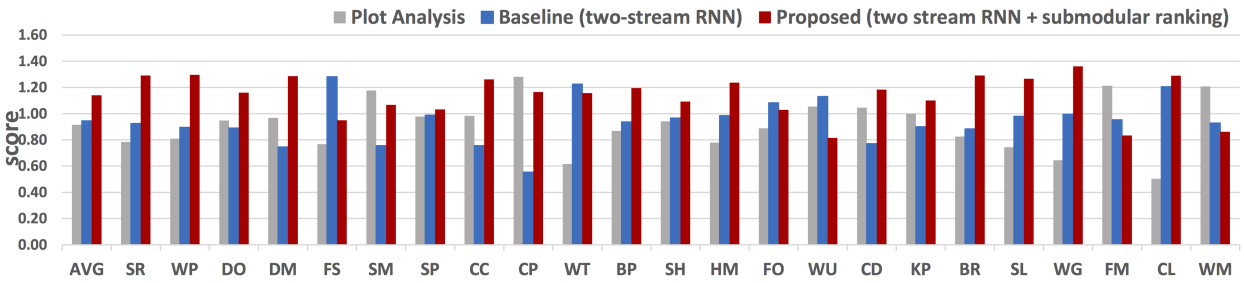
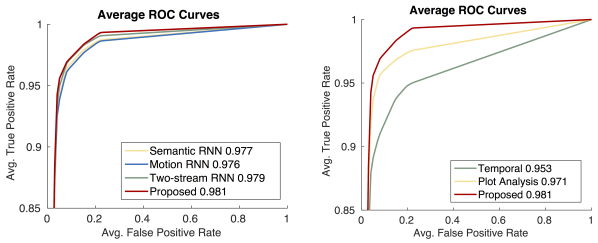


Figure 7: Pairwise preference scores. The scores are generated in a similar way to [6]. The proposed method (i.e., two-stream RNN + submodular ranking) receives higher average pairwise score (i.e., 1.14), compared with our baseline (i.e., two-stream RNN) and the PA [6] method with the average pairwise scores of 0.94 and 0.91, respectively. The abbreviations indicate the contents of each video set: SR (surfing + river), WP (walk + park), DO (drive + ocean), DM (drive + morning), FS (family + swim), SM (shopping + mall), SP (sightseeing + park), CC (chatting + cafe), CP (couple + park), WT (walk + trees), SP (skateboarding + park), SH (sightseeing + hill), HM (hiking + mountain), FO (friend + ocean), WU (walk + urban), CD (cat + dog), KP (kid + park), BR (skateboard + road), SL (sightseeing + lake), WG (walk + garden), FM (family + market), CL(car + lake), WM (walk + museum).



(a) (b)

Figure 8: Component-wise comparison results. The corresponding AUC scores are also provided in the legend. (a) average ROC curves for our methods with different design options. (b) average ROC curves for the proposed method and the comparisons. Our method achieves higher ROC curves and AUC scores compared to several baseline methods and the state-of-the-art algorithm.

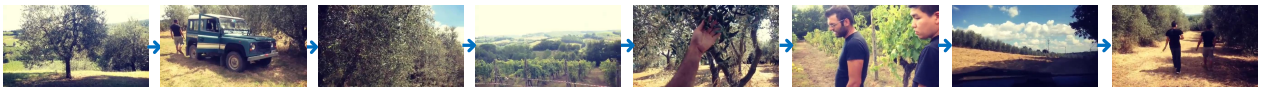
our baseline (i.e., two-stream RNN) against the state-of-the-art method, i.e., Plot Analysis (PA) [6].

More experimental results can be found in the supplementary material, and the MATLAB codes will be made available to the public for reproducible research.

Overall Video-Story Quality. Since evaluating the quality of the video-story composition is complex and subjective, we conduct a user study on the Amazon Mechanical Turk following the settings used in [6]. We show each subject to choose the one with the better video-story from a pair of composed results. Each pair consists of video-stories composed by two different methods while containing the same contents. All the video-story results from the dataset are shown in random orders. Our evaluation involves 134 subjects, resulting in a total of 3,105 pairwise results. After obtaining all the pairwise results, we use the Bradley-Terry (B-T) model [3, 29] to obtain the global ranking scores. The B-T scores of the proposed algorithm, our baseline and the Plot Analysis [6] method are 1.22, 0.88 and 0.85, which demonstrates the effectiveness of the proposed model.

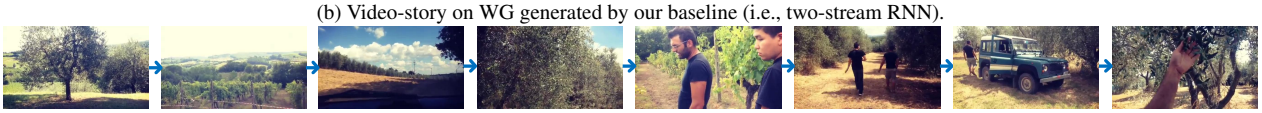
Similar to [6], Figure 7 shows the results of the pairwise preference test of our method and the comparisons. We find that the proposed method performs better when the given clips contain various scenes and activities, e.g., BR and WG video sets. In our method, the learned video coherence can help to handle such challenges and generate results with smooth and consistent view transitions. Figure 9 shows another example where the WG video set contains walking and garden such that the appearances (e.g., color distribution) in the relevant clips are similar. As a result, the PA method does not perform well due to ambiguous coherence. In contrast, the proposed RNN models the relations of scenes and activities, and thus generate video-stories with more coherent contents. In addition, we demonstrate the proposed submodular ranking process can further improve results in a dynamic scene environment. In Figure 10, the BR video set consists of a series of skateboarding actions. The scenes between video clips are similar (e.g., the road) and the contents change dynamically. The results show that our submodular ranking process incorporates both the dynamics and coherence to generate a smoother video-story.

Component Composition Quality. In this task, we evaluate the coherence quality between two adjacent components in the composed video-story. We evaluate the results using the component-wise ground truths provided from [6]. For each video set, we obtain the average ROC curves generated by the proposed algorithm and evaluated methods. We first evaluate different components of our framework, i.e., semantic RNN, motion RNN, two-stream RNN (our



(a) Video-story on WG generated by PA [6].

(b) Video-story on WG generated by our baseline (i.e., two-stream RNN).

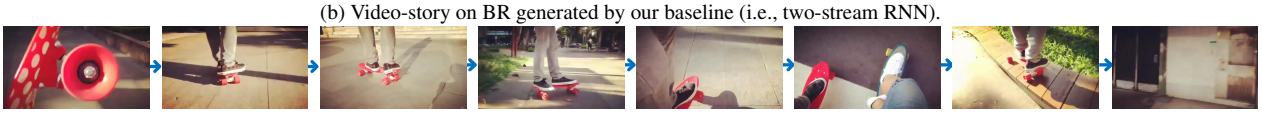


(c) Video-story on WG generated by the proposed method (i.e., two-stream RNN + submodular ranking).

Figure 9: Video-story results on the WG video set. The video clips in this video set contain various scenes and human actions. Our baseline and the proposed method order the similar scenes and activities together while the results generated by PA [6] show less consistency.

(a) Video-story on BR generated by PA [6].

(b) Video-story on BR generated by our baseline (i.e., two-stream RNN).



(c) Video-story on BR generated by the proposed method (i.e., two-stream RNN + submodular ranking).

Figure 10: Video-story results on the BR video set. The video clips in this video set contain similar activities and various dynamics. The proposed baseline method produces smooth transitions, while our submodular ranking process further improves the baseline results.

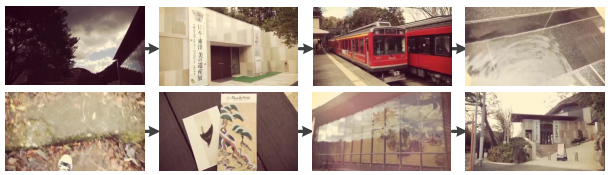


Figure 11: Failure cases by the proposed method.

baseline), and the submodular ranking process. Figure 8(a) shows that our baseline, i.e., two-stream RNN, incorporates both semantic and motion information, and generate better component-wise results. In addition, our submodular ranking process further improves the performance, in which the dynamics are considered to better match the video-story structure. Then we compare the proposed method with the state-of-the-art method, i.e., PA [6] and the temporal results in Figure 8(b). Our method achieves higher ROC curve and AUC scores, which shows the effectiveness of our learned coherence and submodular ranking process.

Failure Cases. As shown in Figure 11, in the cases with uneventful or irrelevant scenes and activities (e.g., walk and museum), our method shows less effectiveness since the contents of clips do not affect the whole story.

5. Concluding Remarks

In this paper, we focus on the video-story composition task via a learning based approach. Since the video contents may change significantly through the time, we exploit the coherence between video clips to predict connections for compositing video-stories. In the proposed framework, we train the two-stream RNN in terms of spatial-temporal semantics and motion dynamics. The probabilities generated by the two-stream RNN are fused as the coherence scores of video clips to generate smooth and relevant compositions. To further match the video-story structure, we formulate a submodular ranking problem to rearrange the video-story composition. Experimental results on the video-story dataset show that the proposed algorithm performs favorably against the state-of-the-art approach.

Acknowledgements: This work is supported in part by NSFC (No. 61572099 and 61522203), NSF CAREER (No. 1149783), 973 Program (No. 2014CB347600), NSF of Jiangsu Province (No. BK20140058), the National Key R&D Program of China (No. 2016YFB1001001), and gifts from Adobe and Nvidia.

References

[1] T. Basha, Y. Moses, and S. Avidan. Photo sequencing. In *ECCV*, 2012.

[2] Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *TNN*, 5(2):157–166, 1994.

[3] R. A. Bradley and M. E. Terry. Rank analysis of incomplete block designs the method of paired comparisons. *Biometrika*, 39(3-4):324–345, 1952.

[4] F. Caba Heilbron, V. Escorcia, B. Ghanem, and J. Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *CVPR*, 2015.

[5] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *CVPR*, 2009.

[6] J. Choi, T.-H. Oh, and I. So Kweon. Video-story composition via plot analysis. In *CVPR*, 2016.

[7] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015.

[8] J. L. Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[9] M. Gygli, H. Grabner, H. Riemenschneider, and L. Van Gool. Creating summaries from user videos. In *ECCV*, 2014.

[10] M. Gygli, H. Grabner, and L. Van Gool. Video summarization by learning submodular mixtures of objectives. In *CVPR*, 2015.

[11] K. He, X. Zhang, S. Ren, and J. Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *ECCV*, 2014.

[12] R. Horst and H. Tuy. *Global optimization: Deterministic approaches*. Springer Science & Business Media, 2013.

[13] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *PAMI*, 35(1):221–231, 2013.

[14] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. Large-scale video classification with convolutional neural networks. In *CVPR*, 2014.

[15] G. Kim and E. P. Xing. Reconstructing storyline graphs for image recommendation from web community photos. In *CVPR*, 2014.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[17] Y. J. Lee, J. Ghosh, and K. Grauman. Discovering important people and objects for egocentric video summarization. In *CVPR*, 2012.

[18] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance. Cost-effective outbreak detection in networks. In *SIGKDD*, 2007.

[19] C. Liu, W. T. Freeman, E. H. Adelson, and Y. Weiss. Human-assisted motion annotation. In *CVPR*, 2008.

[20] C. Liu, J. Yuen, and A. Torralba. Sift flow: Dense correspondence across scenes and its applications. *PAMI*, 33(5):978–994, 2011.

[21] D. Liu, G. Hua, and T. Chen. A hierarchical visual model for video object summarization. *PAMI*, 32(12):2178–2190, 2010.

[22] T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur. Recurrent neural network based language model. In *Interspeech*, 2010.

[23] Y. Pan, T. Mei, T. Yao, H. Li, and Y. Rui. Jointly modeling embedding and translation to bridge video and language. *CVPR*, 2016.

[24] G. A. Sigurdsson, X. Chen, and A. Gupta. Learning visual storylines with skipping recurrent neural networks. In *ECCV*, 2016.

[25] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. In *NIPS*, 2014.

[26] I. Sutskever, J. Martens, and G. E. Hinton. Generating text with recurrent neural networks. In *ICML*, 2011.

[27] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, 2015.

[28] Y.-H. Tsai, G. Zhong, and M.-H. Yang. Semantic co-segmentation in videos. In *ECCV*, 2016.

[29] L. Wei-Sheng, H. Jia-Bin, H. Zhe, A. Narendra, and Y. Ming-Hsuan. A comparative study for single image blind deblurring. In *CVPR*, 2016.

[30] P. J. Werbos. Generalization of backpropagation with application to a recurrent gas market model. *Neural Networks*, 1(4):339–356, 1988.

[31] R. J. Williams and D. Zipser. Gradient-based learning algorithms for recurrent networks and their computational complexity. *Back-propagation: Theory, architectures and applications*, pages 433–486, 1995.

[32] W. Wolf. Key frame selection by motion analysis. In *ICASSP*, 1996.

[33] T. Yao, T. Mei, and Y. Rui. Highlight detection with pairwise deep ranking for first-person video summarization. In *CVPR*, 2016.

[34] H. J. Zhang, J. Wu, D. Zhong, and S. W. Smoliar. An integrated system for content-based video retrieval and browsing. *Pattern recognition*, 30(4):643–658, 1997.

[35] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman. Summary transfer: Exemplar-based subset selection for video summarizatio. In *CVPR*, 2016.

[36] G. Zhong, Y.-H. Tsai, and M.-H. Yang. Weakly-supervised video scene co-parsing. In *ACCV*, 2016.

[37] F. Zhu, Z. Jiang, and L. Shao. Submodular object recognition. In *CVPR*, 2014.