A Local-to-Global Approach to Multi-modal Movie Scene Segmentation

Anyi Rao[1], Linning Xu[2], Yu Xiong[1], Guodong Xu[1], Qingqiu Huang[1], Bolei Zhou[1], Dahua Lin[1]
[1]CUHK - SenseTime Joint Lab, The Chinese University of Hong Kong
[2]The Chinese University of Hong Kong, Shenzhen
{anyirao, xy017, xg018, hq016, bzhou, dhlin}@ie.cuhk.edu.hk, linningxu@link.cuhk.edu.cn

Abstract

Scene, as the crucial unit of storytelling in movies, contains complex activities of actors and their interactions in a physical environment. Identifying the composition of scenes serves as a critical step towards semantic understanding of movies. This is very challenging – compared to the videos studied in conventional vision problems, e.g. *action recognition, as scenes in movies usually contain much richer temporal structures and more complex semantic information. Towards this goal, we scale up the scene segmentation task by building a large-scale video dataset MovieScenes, which contains* $21K$ *annotated scene segments from* 150 *movies. We further propose a local-to-global scene segmentation framework, which integrates multi-modal information across three levels,* i.e. *clip, segment, and movie. This framework is able to distill complex semantics from hierarchical temporal structures over a long movie, providing top-down guidance for scene segmentation. Our experiments show that the proposed network is able to segment a movie into scenes with high accuracy, consistently outperforming previous methods. We also found that pretraining on our MovieScenes can bring significant improvements to the existing approaches.* [1]

1. Introduction

Imagine you are watching the movie *Mission Impossible* starred by Tom Cruise: In a fight scene, Ethan leaps onto a helicopter's landing skid and attaches an exploding gum to the windshield to destroy the enemy. Suddenly, the story jumps into an emotional scene where Ethan pulled the trigger and sacrificed his life to save his wife Julia. Such a dramatic change of scenes plays an important role in the movie's storytelling. Generally speaking, a movie is composed of a well-designed series of intriguing scenes with transitions, where the underlying storyline determines the order of the scenes being presented. Therefore recognizing the movie scenes, including the detection of scene boundaries and the understanding of the scene content, facilitates a wide-range of movie understanding tasks such as scene classification, cross movie scene retrieval, human interaction graph and human-centric storyline construction.

[1]The dataset will be published in compliance with regulations. Intermediate features, pretrained models and related codes will be released. https://anyirao.com/projects/SceneSeg.html

Figure 1. When we look at any single shot from figure (a), *e.g.* the woman in shot B, we cannot infer what the current event is. Only when we consider all the shots 1-6 in this scene, as shown in figure (b), we can recognize that "this woman is inviting a couple to dance with the band."

It is worth noting that *scenes* and *shots* are essentially different. In general, a *shot* is captured by a camera that operates for an uninterrupted period of time and thus is *visually continuous*; while a *scene* is a semantic unit at a higher level. As illustrated in Figure 1, a scene comprises a sequence of shots to present a *semantically coherent* part of the story. Therefore, whereas a movie can be readily divided into shots based on simple visual cues using existing tools [23], the task of identifying those sub-sequences of shots that constitute scenes is challenging, as it requires semantic understanding in order to discover the associations between those shots that are semantically consistent but visually dissimilar.

There has been extensive studies on video understanding. Despite the great progress in this area, most existing works focus on recognizing the categories of certain activities from short videos [28, 6, 14]. More importantly, these works assume a list of pre-defined categories that are visually distinguishable. However, for movie scene segmentation, it is impossible to have such a list of categories. Additionally, shots are grouped into scenes according to their semantical coherence rather than just visual cues. Hence, a new method needs to be developed for this purpose.

To associate visually dissimilar shots, we need semantical understanding. The key question here is *"how can we learn semantics without category label?"* Our idea to tackle this problem consists in three aspects: 1) Instead of attempting to categorize the content, we focus on scene *boundaries*. We can learn what constitute a boundary between scenes in a supervised way, and thus get the capability of differentiating between within-scene and cross-scene transitions. 2) We leverage the cues contained in multiple semantic elements, including *place*, *cast*, *action*, and *audio*, to identify the associations across shots. By integrating these aspects, we can move beyond visual observations and establish the semantic connections more effectively. 3) We also explore the top-down guidance from the overall understanding of the movie, which brings further performance gains.

Based on these ideas, we develop a local-to-global framework that performs scene segmentation through three stages: 1) extracting shot representations from multiple aspects, 2) making local predictions based on the integrated information, and finally 3) optimizing the grouping of shots by solving a global optimization problem. To facilitate this research, we construct *MovieScenes*, a large-scale dataset that contains over $21K$ scenes containing over $270K$ shots from 150 movies.

Experiments show that our method raise performance by 68% (from 28.1 to 47.1 in terms of average precision) than the existing best method [1]. Existing methods pretrained on our dataset also have a large gain in performance.

2. Related Work

Scene boundary detection and segmentation. The earliest works exploit a variety of unsupervised methods. [22] clusters shots according to shot color similarity. In [17], the author plots a shot response curve from low-level visual features and set a threshold to cut scene. [4, 3] further group shots using spectral clustering with a fast global k-means algorithm. [10, 24] predict scene boundaries with dynamic programming by optimizing a predefined optimizing objective. Researchers also resort to other modality information, *e.g*. [13] leverages scripts with HMM, [23] uses low-level visual and audio features to build scene transition graph. These unsupervised methods are not flexible and heavily rely on manually setting parameters for different videos.

Researchers move on to supervised approaches and start to build up new datasets. *IBM OVSD* [21] consists of 21 short videos with rough scenes, which may contain more than one plot. *BBC Planet Earth* [1] comes from 11 Episodes of BBC documentaries. [15] generates synthetic data from *Places205* [31]. However, the videos in these datasets lack rich plots or storylines, thus limits their real-world applications. The number of test videos is so small that cannot reflect the effectiveness of the methods considering the vast variety of scenes. Additionally, their methods take shot as the analytical unit and implement scene segmentation in the local region recursively. Due to their lack of consideration of the semantics within a scene, it is hard to learn high-level semantics and achieve an ideal result.

Scene understanding in images and short videos. Image-based scene analysis [31, 29, 9] can infer some basic knowledge about scenes, *e.g*. what is contained in this image. However, it is hard to tell the action from a single static image since it lacks contextual information around it. Dynamic scene understanding are further studied with seconds-long short videos [6, 14]. However, all these videos take single shot video without enough variations capturing the change of time and places compared to long videos.

Scene understanding in long videos. There are few datasets focusing on scene in long videos. Most available long video datasets focus on identifying casts in movies or TV series [2, 12, 16] and localizing and classifying the actions [8]. *MovieGraphs* [26] focuses on the individual scene clips in a movie and the language structures of a scene. Some transition parts between scenes are discarded, making the information incomplete.

In order to achieve more general scene analysis that could be extended to videos with long time duration, we address scene segmentation in movies with our large-scale *MovieScenes* dataset. We propose a framework considering both the relationship among shots locally and the relationship among scenes globally using multiple semantic elements, achieving much better segmentation results.

3. MovieScenes Dataset

To facilitate the scene understanding in movies, we construct *MovieScenes*, a large-scale scene segmentation dataset that contains $21K$ scenes derived by grouping over $270K$ shots from 150 movies. This dataset provides a foundation for studying the complex semantics within scenes, and facilitates plot-based long video understanding on the top of scenes.

3.1. Definition of Scenes

Following previous definition of *scene* [17, 4, 10, 24], a scene is a plot-based semantic unit, where a certain activity takes place among a certain group of characters. While a scene often happens in a fixed place, it is also possible that a scene traverses between multiple places continually, *e.g*. during a fighting scene in a movie, the characters move from indoor to outdoor. These complex entanglements in scenes cast more difficulty in the accurate detection of scenes which require high-level semantic infor-

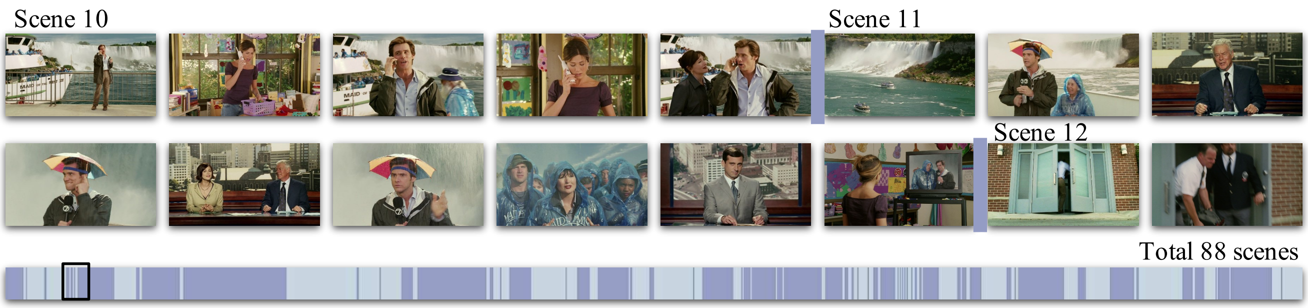

Figure 2. Example of the annotated scenes from movie *Bruce Almight (2003)*. The blue line in the bottom corresponds to the whole movie timeline where the dark blue and light blue regions represent different scenes. In Scene 10, the characters are having a phone call in two different places, thus it requires a semantic understanding of this scene to prevent it from categorizing them into different scenes. In Scene 11, the task becomes even more difficult, as this live broadcasting scene involves more than three places and groups of characters. In this case, visual cues only are likely to fail, thus the inclusion of other aspects such as the audio cues becomes critical.

Table 1. Data consistency statistics of *MovieScenes*. We divide all annotations into three categories: *high/low* consistency cases and *Unsure* cases according to annotators consistency. *unsure* cases are discarded in our experiments. More details are specified in the supplementary materials.

Consist.	High	Low	Unsure
Transit.	16,392 (76.5%)	5,036 (23.5%)	-
Non-trans.	225,836 (92.6%)	18,048 (7.4%)	-
Total	242,052 (89.5%)	23,260 (8.6%)	5,138 (1.9%)

Table 2. A comparison of existing scene datasets.

	#Shot	#Scene	#Video	Time(h)	Source
OVSD [21]	10,000	300	21	10	MiniFilm
BBC [1]	4,900	670	11	9	Docu.
MovieScenes	270,450	21,428	150	297	Movies

mation. Figure 2 illustrates some examples of annotated scenes in *MovieScenes*, demonstrating this difficulty.

The vast diversity of movie scenes makes it hard for the annotators complying with each other. To ensure the consistency of results from different annotations, during the annotation procedure, we provided a list of ambiguous examples with specific guidance to clarify how such cases should be handled. Moreover, all data are annotated by different annotators independently for multiple times. In the end, our multiple times annotation with the provided guidance leads to highly consistent results, *i.e.* 89.5% high consistency cases in total, as shown in Table 1.

3.2. Annotation Tool and Procedure

Our dataset contains 150 movies, and it would be a prohibitive amount of work if the annotators go through the movies frame by frame. We adopt an shot-based approach, based on the understanding that a *shot*[2] could always be uniquely categorized into one scene. Consequently, the scene boundaries must be a subset of all the shot boundaries. For each movie, we first divide it into shots using off-the-shelf methods [23]. This shot-based approach greatly simplifies the scene segmentation task and speeds up the annotation process. We also developed a web-based annotation tool[3] to facilitate annotation. All of the annotators went through two rounds annotation procedure to ensure the high consistency. In the first round, we dispatch each chunk of movies to three independent annotators for later consistency check. In the second round, inconsistent annotations will be re-assigned to two additional annotators for extra evaluations.

3.3. Annotation Statistics

Large-scale. Table 2 compares *MovieScenes* with existing similar video scene datasets. We show that *MovieScenes* is significantly larger than other datasets in terms of the number of shots/scenes and the total time duration. Furthermore, our dataset covers a much wider range of diverse sources of data, capturing all kinds of scenes, compared with short films or documentaries.

Diversity. Most movies in our dataset have time duration between 90 to 120 minutes, providing rich information about individual movie stories. A wide range of genres is covered, including most popular ones such as dramas, thrillers, action movies, making our dataset more comprehensive and general. The length of the annotated scenes varies from less than $10s$ to more than $120s$, where the majority last for $10 \sim 30s$. This large variability existing in both the movie level and the scene level makes movie scene segmentation task more challenging.[4]

[2] A shot is an unbroken sequence of frames recorded from the same camera.

[3] A demonstrated figure of UI is shown in supplementary materials.

[4] More statistical results are specified in the supplements.

4. Local-to-Global Scene Segmentation

As mentioned above, a scene is a series of continuous shots. Therefore, scene segmentation can be formulated as a binary classification problem, *i.e*. to determine whether a shot boundary is a scene boundary. However, this task is not easy, since segmenting scenes requires the recognition of multiple semantic aspects and usage of the complex temporal information.

To tackle this problem, we propose a Local-to-Global Scene Segmentation framework (LGSS). The overall formulation is shown in Equation 1. A movie with n shots is represented as a shot sequence $[\mathbf{s}_1, \cdots, \mathbf{s}_n]$, where each shot is represented with multiple semantic aspects. We design a three-level model to incorporate different levels of contextual information, *i.e*. clip level ($\mathcal{B}$), segment level ($\mathcal{T}$) and movie level ($\mathcal{G}$), based on the shot representation $\mathbf{s}_i$. Our model gives a sequence of predictions $[o_1, \cdots, o_{n-1}]$, where $o_i \in \{0, 1\}$ denotes whether the boundary between the i-th and $(i+1)$-th shots is a scene boundary.

$$\mathcal{G}\{\mathcal{T}[\mathcal{B}([\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_n])]\} = [o_1, o_2, \cdots, o_{n-1}] \quad (1)$$

In the following parts of this section, we will first introduce how to get $\mathbf{s}_i$, namely how to represent the shot with multiple semantic elements. Then we will illustrate the details of the three levels of our model, *i.e*. $\mathcal{B}$, $\mathcal{T}$ and $\mathcal{G}$. The overall framework is shown in Figure 3.

4.1. Shot Representation with Semantic Elements

Movie is a typical multi-modal data that contains different high-level semantic elements. A global feature extracted from a shot by a neural network, which is widely used by previous works [1, 24], is not enough to capture the complex semantic information.

A scene is where a sequence of shots sharing some common elements, *e.g*. place, cast, *etc*. Thus, it is important to take these related semantic elements into consideration for better shot representation. In our LGSS framework, a shot is represented with four elements that play important roles in the constitution of a scene, namely *place*, *cast*, *action*, and *audio*.

To obtain semantic features for each shot $\mathbf{s}_i$, we utilize 1) ResNet50 [11] pretrained on Places dataset [31] on key frame images to get *place* features, 2) Faster-RCNN [19] pretrained on CIM dataset [12] to detect cast instances and ResNet50 pretrained on PIPA dataset [30] to extract *cast* features, 3) TSN [27] pretrained on AVA dataset [8] to get *action* features, 4) NaverNet [5] pretrained on AVA-ActiveSpeaker dataset [20] to separate speech and background sound, and stft [25] to get their features respectively in a shot with 16K Hz sampling rate and 512 windowed signal length, and concatenate them to obtain *audio* features.

4.2. Shot Boundary Representation at Clip Level

As we mentioned before, scene segmentation can be formulated as a binary classification problem on shot boundaries. Therefore, how to represent a shot boundary becomes a crucial question. Here, we propose a Boundary Network (BNet) to model the shot boundary. As shown in Equation 2, BNet, denoted as $\mathcal{B}$, takes a clip of the movie with $2w_b$ shots as input and outputs a boundary representation $\mathbf{b}_i$. Motivated by the intuition that a boundary representation should capture both the *differences* and the *relations* between the shots before and after, BNet consists of two branches, namely $\mathcal{B}_d$ and $\mathcal{B}_r$. $\mathcal{B}_d$ is modeled by two temporal convolution layers, each of them embeds the shots before and after the boundary respectively, following an inner product operation to calculate their differences. $\mathcal{B}_r$ aims to capture the relations of the shots, it is implemented by a temporal convolution layer followed a max pooling.

$$\begin{aligned} \mathbf{b}_i &= \mathcal{B}([\mathbf{s}_{i-(w_b-1)}, \cdots, \mathbf{s}_{i+w_b}]) \quad \text{(window size } 2w_b) \\ &= \begin{bmatrix} \mathcal{B}_d([\mathbf{s}_{i-(w_b-1)}, \cdots, \mathbf{s}_i], [\mathbf{s}_{i+1}, \cdots, \mathbf{s}_{i+w_b}]) \\ \mathcal{B}_r([\mathbf{s}_{i-(w_b-1)}, \cdots, \mathbf{s}_i\ ,\ \mathbf{s}_{i+1}, \cdots, \mathbf{s}_{i+w_b}]) \end{bmatrix} \end{aligned} \quad (2)$$

4.3. Coarse Prediction at Segment Level

After we get the representatives of each shot boundary $\mathbf{b}_i$, the problem becomes predicting a sequence binary labels $[o_1, o_2, \cdots, o_{n-1}]$ based on the sequence of representatives $[\mathbf{b}_1, \cdots, \mathbf{b}_{n-1}]$, which can be solved by a sequence-to-sequence model [7]. However, the number of shots n is usually larger than 1000, which is hard for existing sequential models to contain such a long memory. Therefore, we design a segment-level model to predict a coarse results based on a movie segment that consists of w_t shots ($w_t \ll n$). Specifically, we use a sequential model $\mathcal{T}$, *e.g*. a Bi-LSTM [7], with stride $w_t/2$ shots to predict a sequence of coarse score $[p_1, \cdots, p_{n-1}]$, as shown in Equation 3. Here $p_i \in [0, 1]$ is the probability of a shot boundary to be a scene boundary.

$$[p_1, \cdots, p_{n-1}] = \mathcal{T}([\mathbf{b}_1, \cdots, \mathbf{b}_{n-1}]) \quad (3)$$

Then we get a coarse prediction $\bar{o}_i \in \{0, 1\}$, which indicates whether the i-th shot boundary is a scene boundary. By binarizing p_i with a threshold τ, we get

$$\bar{o}_i = \begin{cases} 1 & \text{if } p_i > \tau, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

4.4. Global Optimal Grouping at Movie Level

The segmentation result $\bar{o}_i$ obtained by the segment-level model $\mathcal{T}$ is not good enough, since it only considers the local information over w_t shots while ignoring the global contextual information over the whole movie. In order to

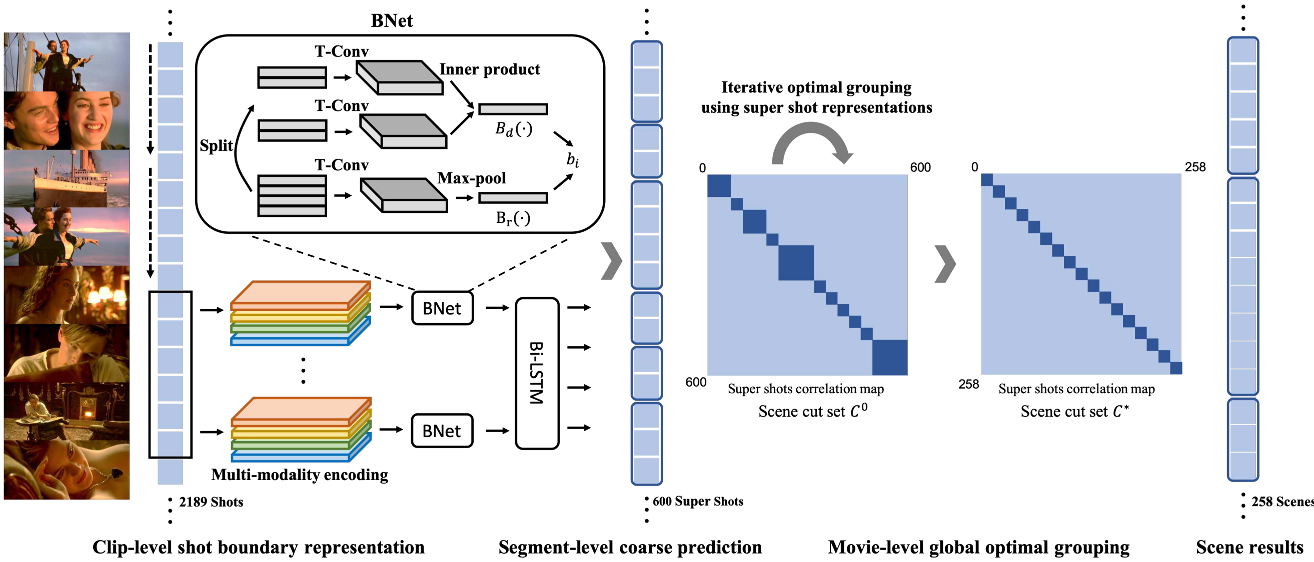

Figure 3. Local-to-Global Scene Segmentation framework (LGSS). At the clip level, we extract four encoding for each shot and take a BNet to model shot boundary. The local sequence model outputs a rough scene cut results at the segment level. Finally, at the movie level, global optimal grouping is applied to refine the scene segmentation results.

capture the global structure, we develop a global optimal model $\mathcal{G}$ to take movie-level context into consideration. It takes the shot representations $\mathbf{s}_i$ and the coarse prediction $\bar{o}_i$ as inputs and make the final decision o_i as follows,

$$[o_1, \cdots, o_{n-1}] = \mathcal{G}([\mathbf{s}_1, \cdots, \mathbf{s}_n], [\bar{o}_1, \cdots, \bar{o}_{n-1}]) \quad (5)$$

The global optimal model $\mathcal{G}$ is formulated as an optimization problem. Before introducing it, we establish the concept of super shots and objective function first.

The local segmentation gives us an initial rough scene cut set $\mathbf{C} = \{\mathcal{C}_k\}$, here we denote $\mathcal{C}_k$ as a *super shot*, *i.e.* a sequence of consecutive shots determined by the segment-level results $[\bar{o}_1, \cdots, \bar{o}_{n-1}]$. Our goal is to merge these super shots into j scenes $\mathbf{\Phi}(n = j) = \{\phi_1, \dots, \phi_j\}$, where $\mathbf{C} = \biguplus_{k=1}^{j} \phi_k$ and $|\phi_k| \geq 1$. Since j is not given, to automatically decide the target scene number j, we need to look through all the possible scene cuts, *i.e.* $F = \max_{j, j<|\mathbf{C}|} F(n = j)$. With fixed j, we want to find the optimal scene cut set $\mathbf{\Phi}^\star(n = j)$. The overall optimization problem is as follows,

$$\begin{aligned} F^\star &= \max_j F(n = j) \quad (6)\\ &= \max_j \left(\max_{\mathbf{\Phi}} \sum_{\phi_k \in \mathbf{\Phi}} g(\phi_k) \right), \\ \text{s.t.} \quad & j < |\mathbf{C}|, |\mathbf{\Phi}| = j. \end{aligned}$$

Here, $g(\phi_k)$ is the optimal scene cut score achieved by the scene ϕ_k. It formulates the relationship between a super shot $\mathcal{C}_l \in \phi_k$ and the rest super shots $\mathcal{P}_{k,l} = \phi_k \backslash \mathcal{C}_l$. $g(\phi_k)$ constitutes two terms to capture a global relationship and a local relationship, $F_s(\mathcal{C}_k, \mathcal{P}_k)$ is similarity score between $\mathcal{C}_k$ and $\mathcal{P}_k$, and $F_t(\mathcal{C}_k, \mathcal{P}_k)$ is an indicate function that whether there is a very high similarity between $\mathcal{C}_k$ and any super shot from $\mathcal{P}_k$ aiming to formulate shots thread in a scene. Specifically,

$$g(\phi_k) = \sum_{\mathcal{C}_k \in \phi_k} f(\mathcal{C}_k, \mathcal{P}_k) = \sum_{\mathcal{C}_k \in \phi_k} (F_s(\mathcal{C}_k, \mathcal{P}_k) + F_t(\mathcal{C}_k, \mathcal{P}_k)),$$

$$F_s(\mathcal{C}_k, \mathcal{P}_k) = \frac{1}{|\mathcal{P}_k|} \sum_{\hat{\mathcal{C}}_k \in \mathcal{P}_k} \cos(\mathcal{C}_k, \hat{\mathcal{C}}_k),$$

$$F_t(\mathcal{C}_k, \mathcal{P}_k) = \sigma(\max_{\hat{\mathcal{C}}_k \in \mathcal{P}_k} \cos(\mathcal{C}_k, \hat{\mathcal{C}}_k)).$$

DP. Solving the optimization problem and determining target scene number can be effectively conducted by dynamic programming (DP). The update of $F(n = j)$ is

$$\max_k \{F^\star(n = j-1 | \mathbf{C}_{1:k}) + g(\phi_j = \{\mathcal{C}_{k+1}, \dots, \mathcal{C}_{|\mathbf{C}|}\})\},$$

where $\mathbf{C}_{1:k}$ is the set containing the first k super shots.

Iterative optimization. The above DP could give us a scene cut result, but we can further take this result as a new super shot set and iteratively merge them to improve the final result. When the super shot updates, we also need to update these super shot representations. A simple summation over all the contained shots may not be an ideal representation for a super shot, as there are some shots containing less informations. Therefore, it would be better if we refine the representation of super shots in the optimal grouping. The details of this refinement on super shot representation are given in the supplements.

5. Experiments

5.1. Experimental Setup

Data. We implement all the baseline methods with our *MovieScenes* dataset. The whole annotation set is split into *Train*, *Val*, and *Test* sets with the ratio 10:2:3 on video level.

Implementation details. We take cross entropy loss for the binary classification. Since there exists unbalance in the dataset, *i.e*. non-scene-transition shot boundaries dominate in amount (approximate 9:1), we take a 1:9 weight on cross entropy loss for non-scene-transition shot boundary and scene-transition shot boundary respectively. We train these models for 30 epochs with Adam optimizer. The initial learning rate is 0.01 and the learning rate will be divided by 10 at the 15th epoch.

In the global optimal grouping, we take $j = 600$ super shots from local segmentation according to the obtained classification scores for these shot boundaries (a movie usually contains $1k \sim 2k$ shot boundaries.) The range of target scenes are from 50 to 400, i.e. $i \in [50, 400]$. These values are estimated based on the *MovieScenes* statistics.

Evaluation Metrics. We take three commonly used metrics: 1) Average Precision (AP). Specifically in our experiment, it is the mean of AP of $o_i = 1$ for each movie. 2) Miou: a weighted sum of intersection of union of a detected scene boundary with respect to its distance to the closest ground-truth scene boundary. 3) Recall@3s: recall at 3 seconds, the percentage of annotated scene boundaries which lies within 3s of the predicted boundary.

5.2. Quantitative Results

The overall results are shown in Table 3. We reproduce existing methods [18, 4, 10, 21, 24, 1] with deep place features for fair comparison. The base model applies temporal convolution on shots with the place feature, and we gradually add the following four modules to it, *i.e*., 1) multiple semantic elements (Multi-Semantics), 2) shot boundary representation at clip level (BNet), 3) coarse prediction at segment level with a local sequence model (Local Seq), and 4) global optimal grouping at movie level (Global).

Analysis of overall results. The performance of random method depends on the ratio of scene-transition/non-scene-transition shot boundary in the test set, which is approximately $1:9$. All the conventional methods [18, 4, 10, 21] outperform random guess, yet do not achieve good performance since they only consider the local contextual information and fail to capture semantic information. [24, 1] achieve better results than conventional methods [18, 4, 10, 21] by considering a large range information.

Analysis of our framework. Our base model applies temporal convolution on shots with the place feature and achieves 19.5 on AP. With the help of multiple semantic elements, our method improves from 19.5 (Base) to 24.3 (Multi-Semantics) (24.6% relatively). The framework with shot boundary modeling using BNet raises the performance from 24.3 (Multi-Semantics) to 42.2 (Multi-Semantics+BNet) (73.7% relatively) which suggests that in the scene segmentation task, modeling shot boundary directly is useful. The method with local sequence model (Multi-Semantics+BNet+Local Seq) achieves 2.7 absolute and 6.4% relative improvement than model (Multi-Semantics+BNet) from 42.2 to 44.9. The full model includes both local sequence model and global optimal grouping (Multi-Semantics+BNet+Local Seq+Global) further improves the results from 44.9 to 47.1, which shows that a movie level optimization are important to scene segmentation.

In all, with the help of multiple semantic elements, clip level shot modeling, segment level local sequence model, and movie level global optimal grouping, our best model outperforms base model and former best model [1] by a large margin, which improves 27.6 absolutely and 142% relatively on base model (Base), and improves 19.0 absolutely and 68% relatively on Siamese [1]. These verify the effectiveness of this local-to-global framework.

5.3. Ablation Studies

Multiple semantic elements. We take the pipeline with shot boundary modeling BNet, local sequence model and global optimal grouping as the base model. As shown in Table 4, gradually adding mid-level semantic elements improves the final results. Starting from the model using place only, audio improves 4.4, action improves 6.5, casts improves 4.0, and improves 8.1 with all together. This result indicates that place, cast, action and audio are all useful information to help scene segmentation.

Additionally, with the help of our multi-semantic elements, other methods [21, 24, 1] achieve $20\% \sim 30\%$ relative improvements. This result further justifies our assumption that multi-semantic elements contributing to the scene segmentation.

Influence of temporal length. We choose different window sizes in the shot boundary modeling at clip level (BNet) and different sequence lengths of Bi-LSTM at segment level

Table 3. Scene segmentation result. In our pipeline, Multi-Semantics means multiple semantic elements, BNet means shot boundary modeling boundary net, Local Seq means local sequence model, Global means global optimal grouping.

Method	AP (↑)	M_{iou} (↑)	Recall(↑)	Recall@3s (↑)
Random guess	8.2	26.8	49.8	54.2
Rasheed *et al.*, GraphCut [18]	14.1	29.7	53.7	57.2
Chasanis *et al.*, SCSA [4]	14.7	30.5	54.9	58.0
Han *et al.*, DP [10]	15.5	32.0	55.6	58.4
Rotman *et al.*, Grouping [21]	17.6	33.1	56.6	58.7
Tapaswi *et al.*, StoryGraph [24]	25.1	35.7	58.4	59.7
Baraldi *et al.*, Siamese [1]	**28.1**	**36.0**	**60.1**	**61.2**
LGSS (Base)	19.5	34.0	57.1	58.9
LGSS (Multi-Semantics)	24.3	34.8	57.6	59.4
LGSS (Multi-Semantics+BNet)	42.2	44.7	67.5	78.1
LGSS (Multi-Semantics+BNet+Local Seq)	44.9	46.5	71.4	77.5
LGSS (all, Multi-Semantics+BNet+Local Seq+Global)	**47.1**	**48.8**	**73.6**	**79.8**
Human upper-bound	81.0	91.0	94.1	99.5

Table 4. Multiple semantic elements scene segmentation ablation results, where four elements are studied including place, cast, action and audio.

Method	place	cast	act	aud	AP (↑)
Grouping [21]	✓				17.6
StoryGraph [24]	✓				25.1
Siamese [1]	✓				**28.1**
Grouping [21]	✓	✓	✓	✓	23.8
StoryGraph [24]	✓	✓	✓	✓	33.2
Siamese [1]	✓	✓	✓	✓	**34.1**
LGSS				✓	17.5
LGSS			✓		32.1
LGSS		✓			15.9
LGSS	✓				39.0
LGSS	✓			✓	43.4
LGSS	✓		✓		45.5
LGSS	✓	✓			43.0
LGSS	✓	✓	✓	✓	**47.1**

Table 5. Comparison of different temporal window size at clip and segment level. The vertical line differs on the window size of clip level shot boundary modeling (BNet), the horizontal line differs on the length of segment level sequence model (seq.).

BNet\seq	1	2	5	10	20
2	43.4	44.2	45.4	46.3	**46.5**
4	44.9	45.2	45.7	**47.1**	46.9
6	44.7	45.0	45.8	**46.7**	46.6

Table 6. Comparison of different hyper-parameters in global optimal grouping and different choices of initial super shot number.

Iter #\Init #	400	600	800	1000
2	**46.5**	46.3	45.9	45.1
4	46.5	46.9	46.4	45.9
5	46.5	**47.1**	**46.6**	**46.0**
Converged value	46.5	47.1	46.6	46.0

(Local Seq). The result is shown in Table 5. The experiments show that a longer range of information improves the performance. Interestingly, the best results come from 4 shots for a shot boundary modeling and 10 shot boundaries as the input of a local sequence model, which involves 14 shot information in total. This is approximately the length of a scene. It shows that this range of temporal information is helpful to scene segmentation.

Choice of hyper-parameters in global optimal grouping. We differ the iteration number of optimization (Iter #) and the initial super shots number (Init #) and show the results in Table 6.

We first take a look at each row and change the initial super shots number. The setting with initial number 600 achieves the best results, since it is close to the target scene number $50 \sim 400$ and meanwhile ensures enough large search space. Then, when we look at each column, we observe that the setting with initial number 400 converges in the fastest way. It achieves the best results very quickly after 2 iterations. And all the settings coverge within 5 iterations.

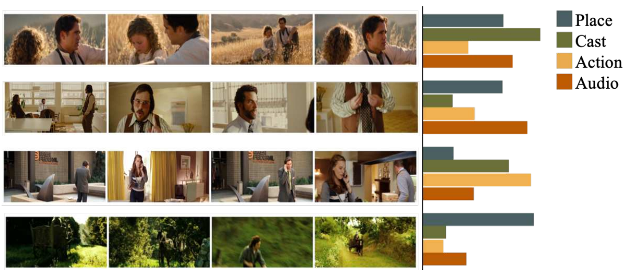

Figure 4. Multiple semantic elements interpretation, where the norm of similarity of each semantic element is represented by the corresponding bar length. These four movie clips illustrate how different elements contribute to the prediction of a scene.

Figure 5. Qualitative results of global optimal grouping in two cases. In each case, the first and second row are the results before and after the global optimal grouping respectively. The red line among two shots means there is a scene cut. The ground truth of each case is that these shots belong to the same scene.

5.4. Qualitative Results

Qualitative results showing the effectiveness of our multi-modal approach is illustrated in Figure 4, and the qualitative results of global optimal grouping are shown in Figure 5. [5]

Multiple semantic elements. To quantify the importance of multiple semantic elements, we take the norm of the cosine similarity for each modality. Figure 4 (a) shows an example where the cast is very similar in consecutive shots and help to contribute to the formation of a scene. In Figure 4 (b), the characters and their actions are hard to recognize: the first shot is a long shot where the character is very small, and the last shot only shows one part of the character without a clear face. In these cases, a scene is recognized thanks to the similar audio feature that is shared among these shots. Figure 4 (c) is a typical "phone call" scene where the action in each shot is similar. In Figure 4 (d), only place is similar and we still conclude it as one scene. From the above observations and analysis on more such cases, we come to the following empirical conclusions: multi-modal information is complementary to each other and help the scene segmentation.

[5]More results are shown in the supplementary materials.

Table 7. Scene segmentation cross dataset transfer result (AP) on existing datasets.

Method	OVSD [21]	BBC [1]
DP [10]	58.3	55.1
Siamese [1]	65.6	62.3
LGSS	**76.2**	**79.5**
DP-pretrained [10]	62.9	58.7
Siamese-pretrained [1]	76.8	71.4
LGSS-pretrained	**85.7**	**90.2**

Optimal grouping. We show two cases to demonstrate the effectiveness of optimal grouping. There are two scenes in Figure 5. Without global optimal grouping, a scene with sudden view point change is likely to predict a scene transition (red line in the figure), *e.g.* in the first case, the coarse prediction gets two scene cuts when the shot type changes from a full shot to a close shot. In the second case, the coarse prediction gets a scene cut when a extreme close up shot appears. Our global optimal grouping is able to smooth out these redundant scene cuts as we expected.

5.5. Cross Dataset Transfer

We test different methods DP [10] and Siamese [1] on existing datasets OVSD [1] and BBC [21] with pretraining on our MovieScenes dataset, and the results are shown in Table 7. With pretraining on our dataset, the performances achieve significant improvements, *i.e.* $\sim$ 10 absolute and $\sim 15\%$ relative improvements in AP. The reason is that our dataset covers much more scenes and brings a better generalization ability to the model pretrained on it.

6. Conclusion

In this work, we collect a large-scale annotation set for scene segmentation on 150 movies containing $270K$ annotations. We propose a local-to-global scene segmentation framework to cover a hierarchical temporal and semantic information. Experiments show that this framework is very effective and achieves much better performance than existing methods. A successful scene segmentation is able to support a bunch of movie understanding applications. [6] All the studies in this paper together show that scene analysis is a challenging but meaningful topic which deserves further research efforts.

Acknowledgment This work is partially supported by the General Research Fund (GRF) of Hong Kong (No. 14203518 & No. 14205719) and SenseTime Collaborative Grant on Large-scale Multi-modality Analysis.

[6]More details are shown in the supplementary materials.

References

[1] Lorenzo Baraldi, Costantino Grana, and Rita Cucchiara. A deep siamese network for scene detection in broadcast videos. In *23rd ACM International Conference on Multimedia*, pages 1199–1202. ACM, 2015. 2, 3, 4, 6, 7, 8

[2] Piotr Bojanowski, Francis Bach, Ivan Laptev, Jean Ponce, Cordelia Schmid, and Josef Sivic. Finding actors and actions in movies. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2280–2287, 2013. 2

[3] Brandon Castellano. Pyscenedetect: Intelligent scene cut detection and video splitting tool. https://pyscenedetect.readthedocs.io/en/latest/, 2018. 2

[4] Vasileios T Chasanis, Aristidis C Likas, and Nikolaos P Galatsanos. Scene detection in videos using shot clustering and sequence alignment. *IEEE transactions on multimedia*, 11(1):89–100, 2008. 2, 6, 7

[5] Joon Son Chung. Naver at activitynet challenge 2019–task b active speaker detection (ava). *arXiv preprint arXiv:1906.10555*, 2019. 4

[6] Bernard Ghanem Fabian Caba Heilbron, Victor Escorcia and Juan Carlos Niebles. Activitynet: A large-scale video benchmark for human activity understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–970, 2015. 1, 2

[7] Alex Graves and Jürgen Schmidhuber. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610, 2005. 4

[8] Chunhui Gu, Chen Sun, David A Ross, Carl Vondrick, Caroline Pantofaru, Yeqing Li, Sudheendra Vijayanarasimhan, George Toderici, Susanna Ricco, Rahul Sukthankar, et al. Ava: A video dataset of spatio-temporally localized atomic visual actions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6047–6056, 2018. 2, 4

[9] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015. 2

[10] Bo Han and Weiguo Wu. Video scene segmentation using a novel boundary evaluation criterion and dynamic programming. In *2011 IEEE International conference on multimedia and expo*, pages 1–6. IEEE, 2011. 2, 6, 7, 8

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 4

[12] Qingqiu Huang, Yu Xiong, and Dahua Lin. Unifying identification and context learning for person recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2217–2225, 2018. 2, 4

[13] Chao Liang, Yifan Zhang, Jian Cheng, Changsheng Xu, and Hanqing Lu. A novel role-based movie scene segmentation method. In *Pacific-Rim Conference on Multimedia*, pages 917–922. Springer, 2009. 2

[14] Mathew Monfort, Alex Andonian, Bolei Zhou, Kandan Ramakrishnan, Sarah Adel Bargal, Yan Yan, Lisa Brown, Quanfu Fan, Dan Gutfreund, Carl Vondrick, et al. Moments in time dataset: one million videos for event understanding. *IEEE transactions on pattern analysis and machine intelligence*, 2019. 1, 2

[15] Stanislav Protasov, Adil Mehmood Khan, Konstantin Sozykin, and Muhammad Ahmad. Using deep features for video scene detection and annotation. *Signal, Image and Video Processing*, pages 1–9, 2018. 2

[16] Vignesh Ramanathan, Armand Joulin, Percy Liang, and Li Fei-Fei. Linking people in videos with "their" names using coreference resolution. In *European conference on computer vision*, pages 95–110. Springer, 2014. 2

[17] Zeeshan Rasheed and Mubarak Shah. Scene detection in hollywood movies and tv shows. In *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, volume 2, pages II–343. IEEE, 2003. 2

[18] Zeeshan Rasheed and Mubarak Shah. Detection and representation of scenes in videos. *IEEE transactions on Multimedia*, 7(6):1097–1105, 2005. 6, 7

[19] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 91–99. Curran Associates, Inc., 2015. 4

[20] Joseph Roth, Sourish Chaudhuri, Ondrej Klejch, Radhika Marvin, Andrew Gallagher, Liat Kaver, Sharadh Ramaswamy, Arkadiusz Stopczynski, Cordelia Schmid, Zhonghua Xi, et al. Ava-activespeaker: An audio-visual dataset for active speaker detection. *arXiv preprint arXiv:1901.01342*, 2019. 4

[21] Daniel Rotman, Dror Porat, and Gal Ashour. Optimal sequential grouping for robust video scene detection using multiple modalities. *International Journal of Semantic Computing*, 11(02):193–208, 2017. 2, 3, 6, 7, 8

[22] Yong Rui, Thomas S Huang, and Sharad Mehrotra. Exploring video structure beyond the shots. In *Proceedings. IEEE International Conference on Multimedia Computing and Systems (Cat. No. 98TB100241)*, pages 237–240. IEEE, 1998. 2

[23] Panagiotis Sidiropoulos, Vasileios Mezaris, Ioannis Kompatsiaris, Hugo Meinedo, Miguel Bugalho, and Isabel Trancoso. Temporal video segmentation to scenes using high-level audiovisual features. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(8):1163–1177, 2011. 1, 2, 3

[24] Makarand Tapaswi, Martin Bauml, and Rainer Stiefelhagen. Storygraphs: visualizing character interactions as a timeline. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 827–834, 2014. 2, 4, 6, 7

[25] Srinivasan Umesh, Leon Cohen, and D Nelson. Fitting the mel scale. In *1999 IEEE International Conference on Acoustics, Speech, and Signal Processing. Proceedings. ICASSP99 (Cat. No. 99CH36258)*, volume 1, pages 217–220. IEEE, 1999. 4

[26] Paul Vicol, Makarand Tapaswi, Lluis Castrejon, and Sanja Fidler. Moviegraphs: Towards understanding human-centric situations from videos. In *Proceedings of the IEEE Con-

ference on Computer Vision and Pattern Recognition, pages 8581–8590, 2018. 2

[27] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Val Gool. Temporal segment networks: Towards good practices for deep action recognition. In *ECCV*, 2016. 4

[28] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition. In *European conference on computer vision*, pages 20–36. Springer, 2016. 1

[29] Mark Yatskar, Luke Zettlemoyer, and Ali Farhadi. Situation recognition: Visual semantic role labeling for image understanding. In *Conference on Computer Vision and Pattern Recognition*, 2016. 2

[30] Ning Zhang, Manohar Paluri, Yaniv Taigman, Rob Fergus, and Lubomir Bourdev. Beyond frontal faces: Improving person recognition using multiple cues. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4804–4813, 2015. 4

[31] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(6):1452–1464, 2018. 2, 4