Scene Separation & Data Selection: Temporal Segmentation Algorithm for Real-time Video Stream Analysis

Yuelin Xin[1,2] Zihan Zhou[1] Yuxuan Xia[1]

[1]SWJTU-Leeds Joint School, CS
Southwest Jiaotong University

[2]School of Computing
University of Leeds

Spatio-Temporal Reasoning and Learning, 2022

Miracle Factory

Outline

Introduction

- The problem (Background & What we want to achieve)
- Our motivation (Why not neural networks?)

Remark

Scene separation(Temporal segmentation) is a problem in which we want to separate a video stream into different scenes. **A scene** is defined as a group of similar-looking frames that are temporally adjacent to each other.

Miracle Factory

The problem

- **Background**: real-time video stream interpretation, including video semantics / video accessibility / surveillance footage auto-interpretation, etc.
- **Difficulties**: algorithms do not see video as a continuous stream of images, but as discrete frames.

Figure 1: Video semantics.

Miracle Factory

The problem

- **The traditional approach**: 3D CNNs (CNN models with the additional temporal dimension)
- **What's missing**: hard to control when the video is very long or it is of indefinite length (like live streaming).

Example

It would be hard to pick up sudden moves in long videos because the longer the video, the worse the temporal resolution. (like a very tiny object in a very massive picture in 2D CNNs)

Our motivation

Why not neural networks?

- Neural networks are relatively slow, the inference time of a lot of NNs makes them difficult to be used in real-time video analysis.
- And the 2SDS algorithm is fully capable of handling simple scene separation tasks.

Algorithm	FPS(higher better)	Avg. Inference time
YOLOv5s	11	92.2ms
2SDS	227	4.4ms

Table 1: Comparison of inference speed under same hardware.[1]

[1] Apple M1 (CPU)

Method: 2SDS

- Related work: SlowFast Networks architecture
- Our work: 2SDS architecture
- Scene separation
- Data selection

Related work

SlowFast Networks [Feichtenhofer *et al.*, 2019]

- **Slow pathway**:CNN with high spatial resolution(low FPS).
- **Fast pathway**:CNN with high temporal resolution(high FPS).

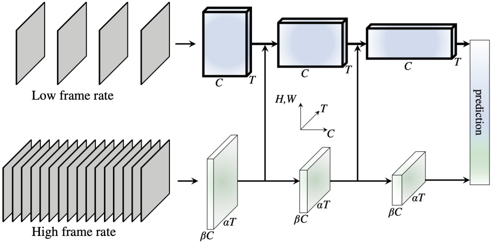

Figure 2: SlowFast Networks Architecture.

Our work

Similar with the SlowFast Networks architecture, but we replace the fast pathway with 2SDS.
This architecture has an even **finer temporal resolution** because we replaced the CNN with a faster algorithm.

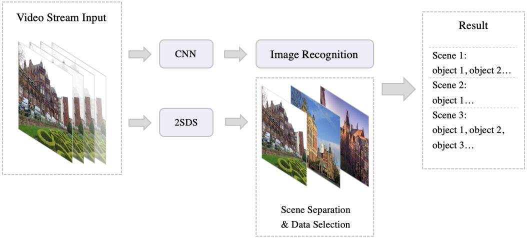

Figure 3: 2SDS Architecture.

2SDS: a two step method

- Step 1: **Scene separation**(Temporal segmentation)
- Step 2: **Data selection**

Scene separation

- **Down sample**: Downsample the frames to 8 by 9 (simplify calculation / make the algorithm less sensitive to small changes in the video).
- **Gray scale**: Convert RGB into grayscale to reduce calculation complexity.

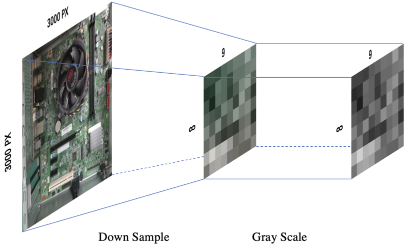

Figure 4: Temporal segmentation.

Miracle Factory

Scene separation

- **Calculate Hash value**: Convert the grayscale graph into a 16-bit hash value, using the following rules:
  (a): One binary value stands for the grayscale difference between two adjacent pixels.
  (b): If the gray scale value of the pixel on the left is greater than the pixel on the right, the value is 1, otherwise it is 0.

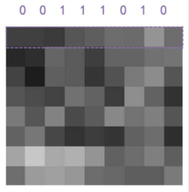

Figure 5: Binary sequence convertion.

Scene separation

- **Calculate Hamming distance**: Calculate the Hamming distance between two adjacent frames.
  Hamming distance is what determines the similarity between two frames, the higher the distance, the less similar the two frames are.

Example

The Hamming distance between the hash values $h_1 = c4e0d8988c989898$ and $h_2 = eee6989c8c989898$ is:
$c4e0d8988c989898 \bigoplus eee6989c8c989898 = 7$

Data selection

- **Data smoothing**: Filter out the data noise by using a weighted average pooling filter.

$$\psi = \frac{\sum_{i=1}^{i \leqslant \varphi} (L_i \times \omega_i)}{\sum_{i=1}^{i \leqslant \varphi} \omega_i}$$

$$\begin{cases} f_i(D) = \min_{i=0} |card(D_i) - \psi| \\ C_I = [c \in D | card(c) = f_i(D)] \end{cases}$$

- **Data selection**: Merge the selected frames into a single frame(used as the output).

Experiment

We have chosen 3 types of testing videos:

- **Interview**
- **Vibrant**
- **Hybrid**

Preliminary results

Type No.	Output - Truth	Accuracy
Interview 1	25 - 25	100.00%
Interview 2	35 - 29	82.86%
Interview 3	31 - 28	90.32%
Interview Avg.	91 - 82	90.10%
Vibrant 1	9 - 13	69.23%
Vibrant 2	19 - 38	50.00%
Vibrant Avg.	28 - 51	54.90%
Hybrid 1	105 - 106	99.06%

Table 2: Preliminary experiment results.

Future improvements

Conclusion