Shot Boundary Detection and Video Captioning Using Neural Networks

Avantika Balaji, S. Ganesh, T. Abishek Balaji, and K. R. Sarath Chandran

1 Introduction

There has been a surge in the number of videos in cyberspace due to the availability of highly developed communication technologies and multimedia devices. This makes video segmentation and analysis the need of the hour. Shot boundary detection is a substantial process of video browsing and retrieval as it aims to detect transitions and their boundaries between consecutive shots. The factors affecting the performance of a shot boundary detection algorithm are computation complexity and the ability to accurately identify transitions in a video, and this is still very much a domain of research as even state-of-the-art transition models tend to get confused by fast abrupt changes of the visual contents caused by camera/environment/entity. This makes it important to search for more advanced approaches. The shot information can further be used for video captioning, content-based video indexing and retrieval. In this era where an exploding amount of visual data is generated every day, video captioning can have many real-life applications. For example, automatic generation of captions for videos would greatly help users to filter what's relevant to them among the huge number of videos on platforms like YouTube. Additionally, video captioning techniques will make videos accessible to the disabled. But, one of the main challenges faced by video captioning is the scarcity of video datasets that have accompanying

A. Balaji (✉) · S. Ganesh · T. Abishek Balaji · K. R. Sarath Chandran
Department of CSE,SSN College of Engineering,Chennai, India
e-mail: avantika17021@cse.ssn.edu.in

S. Ganesh
e-mail: ganesh174307@cse.ssn.edu.in

T. Abishek Balaji
e-mail: abishek17005@cse.ssn.edu.in

K. R. Sarath Chandran
e-mail: sarathchandran@ssn.edu.in

J. D. Peter et al. (eds.), *Disruptive Technologies for Big Data and Cloud Applications*,
Lecture Notes in Electrical Engineering 905,
https://doi.org/10.1007/978-981-19-2177-3_28

text descriptions, because collecting and annotating videos are significantly more arduous and expensive than images. Due to this, most video captioning models are not very accurate and are very heavy weight and complex. To optimize the system, we extract frames at a regular interval from each shot of the video as images, generate captions for the images, and combine the captions of the images belonging to the same shot using semantic textual similarity. This results in a caption/a set of captions describing each shot of the video.

2 Related Works

Shot boundary detection or techniques used to identify a transition in a sequence of frames are a highly researched topic where various methods have been proposed for the past few decades.

Pixel-based approaches are the simplest form of shot boundary detection algorithms. In the work proposed by [2], the pixel values of two consecutive frames are taken and compared pixel by pixel to evaluate the global change in the pixel values. When the resulting value is greater than a threshold, it detects that a transition has occurred. This involves major drawbacks during the detection of transitions in fast camera movements or pan/zoom shots.

Histogram difference approach, as used in [11], is similar to pixel based, but instead of comparing pixel values, it compares the histograms of two consecutive frames. Since histogram is not as sensitive to minor changes, it results in less false hits. But, one of the major problems here is any two frames of a video sequence can have same histogram values but differ in the content of the frames.

The latest approaches involve using convolutional neural networks due to their ability to extract high-level features from images and video frames. The latest work proposed in [3] employs gradient and color information to make the technique of SBD resilient to illumination effects and abrupt motions. Another approach, proposed by [12], uses a modified CNN called a dilated convolutional neural network. This approach uses multiple 3D convolutions resulting in a wider receptive field as well as fewer trainable parameters.

Describing the content of an image or a video has been a highly researched topic in artificial intelligence. This problem requires two phases to be solved, that is, a convolutional neural network for feature extraction and natural language processing to form a description of the extracted features [7]. Earlier methods consist of visual recognizers coupled with structured formal language like And-Or graphs [13]. These types of methods are extremely hand-built and are subject to errors. Another drawback is that such models are designed exclusively for a specific use case and do not work effectively for other cases beyond the scope.

Recent works on image captioning are dominated by deep neural network techniques that use CNN for feature extraction from a set of images and subsequently use a language model, typically a recurrent neural network like long short-term memory (LSTM) to generate a description for the objects (features) extracted by the CNN layer(s) [14, 15]. The utility of this technique is that the CNN layers can be trained on various different types of image data that can extract very deep features of a given frame thus resulting in a much accurate description generated by combining it with the LSTM layer.

3 Methodology

This section details the overall flow of the system, the architectural design, and the methodology of the proposed system. Figure 1 represents the architecture of the system.

The system can be broken down into three parts—shot boundary detection, video captioning, and semantic textual similarity. The detailed design of each module, its functionality, and the algorithms used are discussed in the following sections.

3.1 Shot Boundary Detection

The proposed architecture follows the standard convolutional architectures, but with the convolutions replaced by dilated convolutions. The input to the network is a series of N successive frames of the video to which a sequence of 3D convolutions is applied. The final layer returns a prediction for each frame in the given input, where the prediction represents the likelihood of a given frame being a shot boundary. Based on a fixed threshold θ, the shot boundaries can be identified. Once the frame numbers of the beginning and end of a shot are obtained, the respective timestamps are computed by dividing the frame number by the frame rate (fps) (Fig. 2).

The principal constituent of the model is the dilated CNN. The dilated CNN network comprises four 3-dimensional 3 * 3 * 3 convolutions, with each convolution

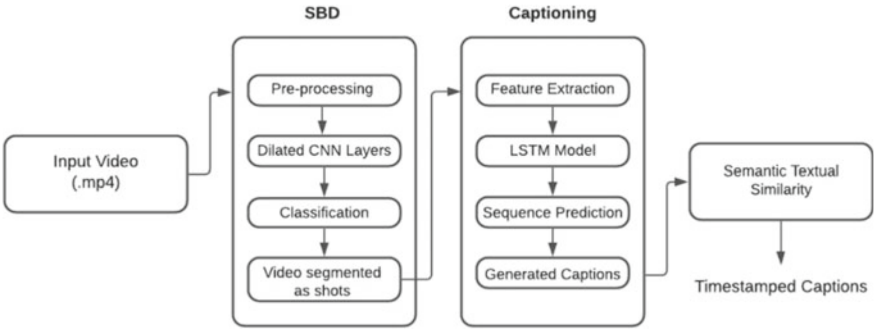


Fig. 1 Proposed system

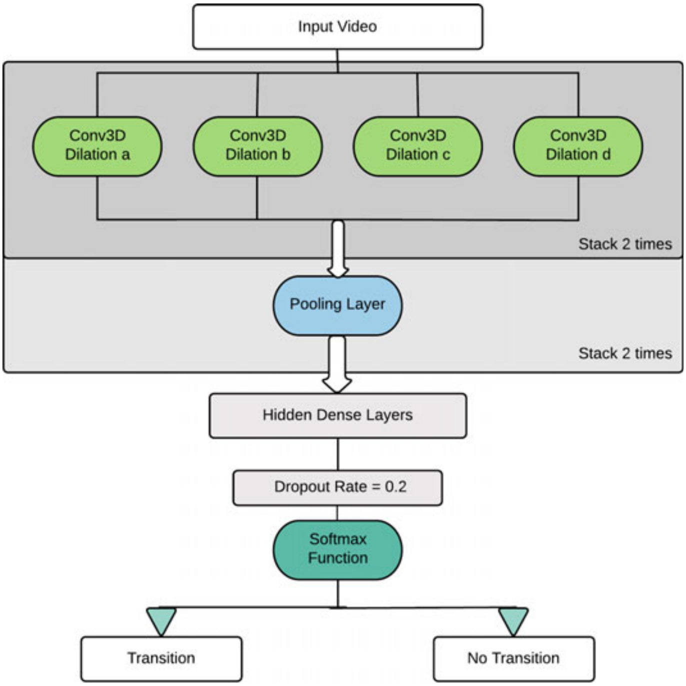


Fig. 2 Architectural design of shot boundary detection

applying a different dilation rate and the output is concatenated in the channel axis. Here, the channel is RGB. Subsequently, this output is passed to the pooling layer which employs the average pooling operation.

The dense layers refine the extracted features, and a dropout of 0.2 is used. The dataset used for training was the TRECVID IACC dataset since it is accompanied by a group of predetermined temporal segments. Training examples are generated by combining any two shots from the set with an arbitrary transition. The network was trained over 30 epochs, each with 300 batches and a batch size of 20. For testing, the BBC dataset [1] was considered.

3.2 Video Captioning

The performance and accuracy of traditional video captioning models are heavily limited by the scarcity of video datasets accompanied by text descriptions. To add to this, video captioning models are also very heavy weight and computationally intensive. To overcome this, we have used an image captioning model, which is lighter, to generate captions for the entire video. The input video is split into an array

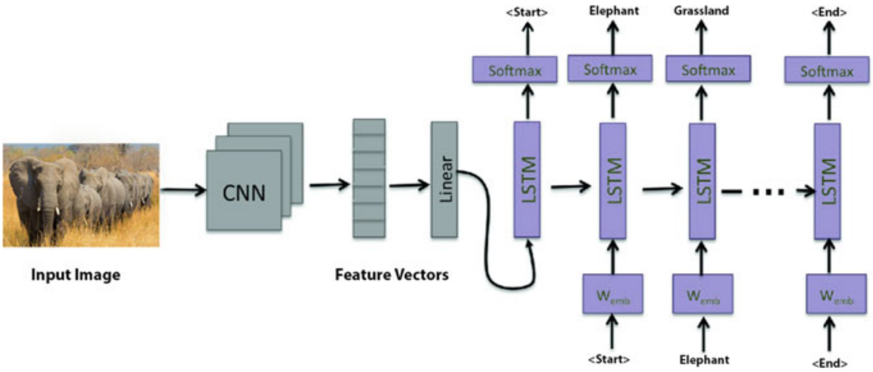


Fig. 3 Architecture of the caption generator

of frames, and at regular intervals (per second of the video), a frame is extracted as an image and given as input to the caption generator.

The architecture comprises two major components: convolutional neural network (encoder) for feature extraction and a recurrent neural network (decoder) for predicting the sequence of words/caption from the feature vectors obtained from the CNN. Here, the RNN used for predicting the sequence is long short-term memory (LSTM) (Fig. 3).

The encoder is a convolutional neural network which extracts the key features of the input image in the form of a vector. A CNN is typically a combination of feature extraction and classification. The visual feature extractor used here is the ResNet-50 [6] CNN architecture prior to feeding the images to the decoder network (LSTM). The ResNet-50 architecture is an image classification model that is 50 layers deep and can classify images into 1000 different categories. However, image classification is not relevant in this scenario, and only the feature vectors are required. Therefore, by applying transfer learning, the final classification layer of the ResNet-50 model is removed and the feature vectors are obtained.

Subsequently, the language model—LSTM—takes as input the vectors obtained from the CNN and generates the caption. During training, the inputs given to the network are the features extracted from the images and sentence embeddings generated from the ground truth, i.e., the captions. For predicting, the only input given is the extracted features. The MS-COCO [8] dataset was used for training and validation. The model was trained for 40 epochs, and a batch size of 120 was used. During the decoding phase of training, the model minimizes the negative log likelihood of the output predicted.

3.3 Semantic Textual Similarity

The output of the video captioning module is a set of captions describing each shot in the video. These captions are generated for the whole video by taking frames from each shot of the video at regular intervals and giving each of these frames to the image caption generator to generate a caption. The frames extracted from the same shot of the video may have repetitive/similar content, thereby producing the same/similar captions.

To eliminate any redundant captions, the semantic textual similarity between every pair of captions describing a shot is computed. First, the captions are encoded into sentence embeddings using a sentence transformers[1] model called "DistilRoBERTa-base". These sentence embeddings are then used for computing the similarity scores, using cosine similarity. A threshold between 0–1 is chosen, and from pairs of captions that have a similarity score higher than the threshold, a caption may be removed. From experimentation, it was found that the threshold value of 0.7 worked the best for retaining all unique captions and reducing redundancy. The outcome is a single caption/set of captions per shot of the video stored in a text file.

4 Results

In this section, we discuss the qualitative results obtained and quantitative evaluation of the results in terms of precision, recall, and $F1$-scores for shot boundary detection, and BLEU, CIDEr, and METEOR for the caption generator. Figure 4 depicts

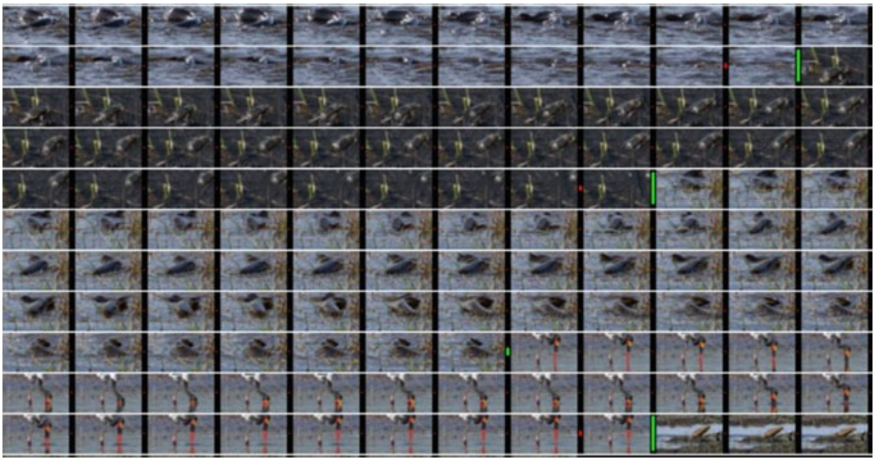


Fig. 4 Visualized results of shot boundary detection

[1] https://pypi.org/project/sentence-transformers/0.3.0/.

the visualized results of the SBD module, where the green lines indicate the shot boundaries.

Figure 5 shows the predictions generated by the captioning module. Figure 6 depicts the overall output of the system, which is in the form of a text file containing a pair of start and end frames for each shot and its description.

In Table 1, T is the total number of transitions, FN indicates false negatives, TP indicates true positives, and FP indicates false positives.

Table 2 discusses the results of the caption generation model in terms of different metrics. These metrics compare the predicted sentences with the ground truth sentences and measure the similarity between them.

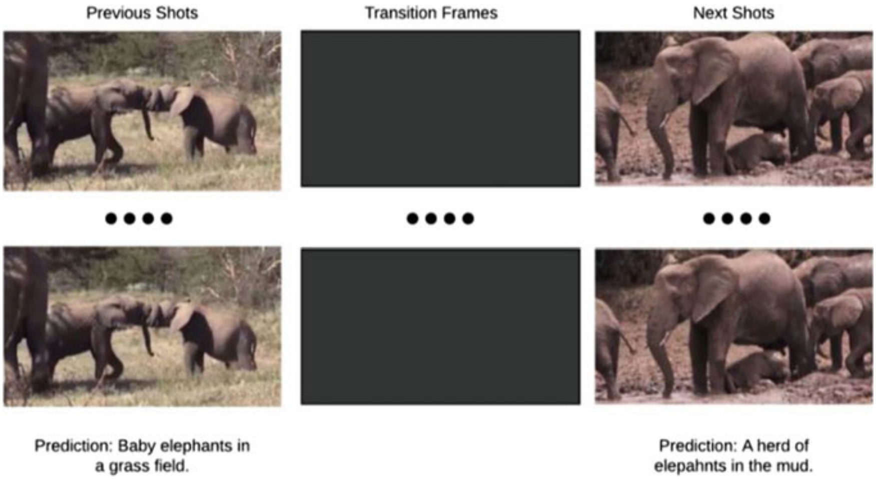
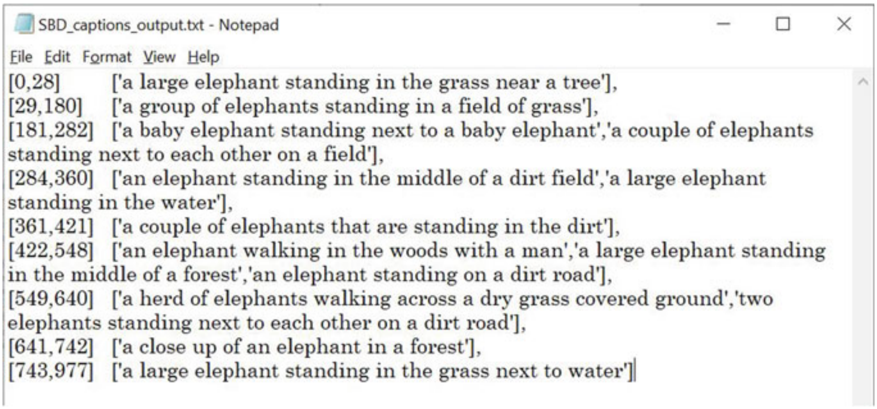

Previous Shots
Transition Frames
Next Shots
Prediction: Baby elephants in a grass field.
Prediction: A herd of elepahnts in the mud.

Fig. 5 Qualitative results of the caption generator



```
[0,28]      ['a large elephant standing in the grass near a tree'],
[29,180]    ['a group of elephants standing in a field of grass'],
[181,282]   ['a baby elephant standing next to a baby elephant','a couple of elephants
standing next to each other on a field'],
[284,360]   ['an elephant standing in the middle of a dirt field','a large elephant
standing in the water'],
[361,421]   ['a couple of elephants that are standing in the dirt'],
[422,548]   ['an elephant walking in the woods with a man','a large elephant standing
in the middle of a forest','an elephant standing on a dirt road'],
[549,640]   ['a herd of elephants walking across a dry grass covered ground','two
elephants standing next to each other on a dirt road'],
[641,742]   ['a close up of an elephant in a forest'],
[743,977]   ['a large elephant standing in the grass next to water']
```

Fig. 6 Output of the system

Table 1 Results obtained on BBC dataset for SBD

Test video	T	TP	FP	FN	Precision	Recall	$F1$-score
BBC_01	445	389	13	56	0.967	0.874	0.918
BBC_02	383	334	17	49	0.951	0.872	0.909
BBC_03	421	385	11	36	0.972	0.914	0.942
BBC_04	472	422	15	50	0.965	0.894	0.928
Overall	1721	1530	56	191	0.964	0.889	**0.924**

Bold values indicate the final score obtained by our model

Table 2 Results of the caption generation model

CIDEr	METEOR	BLEU 1	BLEU 2	BLEU 3	BLEU 4
0.692	0.213	0.657	0.471	0.327	0.228

5 Conclusion and Future Work

We have created a system that incorporates shot boundary detection with a video captioning model with the help of sentence transformers. Shot boundary detection segments the input video, and the description of these segmented videos is generated using our caption generation model. Finally, these captions are compiled into concise information that corresponds to that segment of the video using sentence transformers. This allows us to index videos or categorize different segments of a video automatically based on the content of the video.

Despite the competent performance of the caption generator, it is still far from ideal. Slight errors in the predicted captions may occur due to specific features in the images not being picked up on. In the future, we would like to experiment with attention mechanisms to overcome this and further improve the performance of our model. Currently, our system is also limited by the availability of large-scale datasets to be trained for a generalized sample space. Thus, generating datasets that focus on a specific topic or a field with captions can increase the accuracy. Our work and source code for this system can be found at our GitHub repository.[2]

References

1. L. Baraldi, C. Grana, R. Cucchiara, A deep Siamese network for scene detection in broadcast videos, in *Proceedings of the 23rd ACM International Conference on Multimedia*, Oct 2015, pp. 1199–1202
2. Y. Bendraou, *Video Shot Boundary Detection and Key-Frame Extraction Using Mathematical Models* (Université du Littoral Côte d'Opale, 2017)

[2] https://github.com/Avantika-Balaji/SBD-with-video-captioning.

3. S. Chakraborty, D.M. Thounaojam, SBD-Duo: a dual stage shot boundary detection technique robust to motion and illumination effect. Multimed. Tools Appl. **80**(2), 3071–3087 (2021)
4. Y. Chu et al., Automatic image captioning based on ResNet50 and LSTM with soft attention. Wireless Commun. Mob. Comput. **2020** (2020)
5. D.S. Guru, M. Suhil, P. Lolika, A novel approach for shot boundary detection in videos, in *Multimedia Processing, Communication and Computing Applications* (Springer, New Delhi, 2013)
6. K. He et al., Deep residual learning for image recognition, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2016)
7. N. Krishnamoorthy, G. Malkarnenkar, R. Mooney, K. Saenko, S. Guadarrama, Generating natural-language video descriptions using text-mined knowledge, in *Proceedings of the AAAI Conference on Artificial Intelligence*, June 2013, Vol. 27, No. 1
8. T.-Y. Lin et al., Microsoft coco: common objects in context, in *European Conference on Computer Vision* (Springer, Cham, 2014)
9. E. Nishani, B. Cico, Computer vision approaches based on deep learning and neural networks: deep neural networks for video analysis of human pose estimation, in *2017 6th Mediterranean Conference on Embedded Computing (MECO)* (2017), pp. 1–4
10. N. Reimers, I. Gurevych, Sentence-BERT: sentence embeddings using Siamese BERT-networks, in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing* (Association for Computational Linguistics, 2019)
11. H. Shao, Y. Qu, W. Cui, Shot boundary detection algorithm based on HSV histogram and HOG feature, in *2015 International Conference on Advanced Engineering Materials and Technology*, Aug 2015 (Atlantis Press, 2015), pp. 951–957
12. T. Souček, J. Moravec, J. Lokoč, TransNet: a deep network for fast detection of common shot transitions (2019). arXiv preprint arXiv:1906.03363
13. C.F. Tsai, Image mining by spectral features: a case study of scenery image classification. Expert Syst. Appl. **32**(1), 135–142 (2007)
14. S. Tsutsui, D. Crandall, Using artificial tokens to control languages for multilingual image caption generation (2017). arXiv preprint arXiv:1706.06275
15. S. Venugopalan, H. Xu, J. Donahue, M. Rohrbach, R. Mooney, K. Saenko, Translating videos to natural language using deep recurrent neural networks (2014). arXiv preprint arXiv:1412.4729
16. O. Vinyals, A. Toshev, S. Bengio, D. Erhan, Show and tell: a neural image caption generator, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2015), pp. 3156–3164
17. H. Wang, C. Gao, Y. Han, Sequence in sequence for video captioning. Pattern Recogn. Lett. **130**, 327–334 (2020)