A Stacking Ensemble Approach for Supervised Video Summarization

Yubo An, Shenghui Zhao, Guoqiang Zhang

ABSTRACT
Existing video summarization methods are classified into either shot-level or frame-level methods, which are individually used in a general way. This paper investigates the underlying complementarity between the frame-level and shot-level methods, and a stacking ensemble approach is proposed for supervised video summarization. Firstly, we build up a stacking model to predict both the key frame probabilities and the temporal interest segments simultaneously. The two components are then combined via soft decision fusion to obtain the final scores of each frame in the video. A joint loss function is proposed for the model training. The ablation experimental results show that the proposed method outperforms both the two corresponding individual methods. Furthermore, extensive experimental results on two benchmark datasets show its superior performance in comparison with the state-of-the-art methods.

CCS CONCEPTS
• Computing methodologies; • Artificial intelligence; • Computer vision; • Computer vision tasks; • Video summarization;

KEYWORDS
Video summarization, self-attention, stacking ensemble learning, shot-level, frame-level

ACM Reference Format:
Yubo An, Shenghui Zhao, and Guoqiang Zhang. 2022. A Stacking Ensemble Approach for Supervised Video Summarization. In 2022 4th International Conference on Video, Signal and Image Processing (VSIP 2022), November 25–27, 2022, Shanghai, China. ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3577164.3577183

1 INTRODUCTION
With the rapid development of the mobile networks, the rise of self-media results in massive video data. Hence the computer vision technology that can efficiently browse, watch and summarize videos, which is referred to as video summarization, has attracted more and more attention [1, 2]. Currently, the supervised learning methods for video summarization [3–5] use the training data that is composed of the ground-truth labels manually generated with human preferences. And they are usually superior to the unsupervised methods [6–9] because they can implicitly learn human preferences.

The current deep learning-based approaches for video summarization can be approximately classified into shot-level and frame-level by the partition strategies. The frame-level deep learning methods usually rely on Long Short-Term Memory (LSTM) or attention mechanism to capture long-term and short-term dependencies within a video, and use appropriate frame scoring networks to predict the probability of each frame being selected into the video summary [1, 4, 10, 11]. But the prediction scores of the video frames in the same semantic segment cannot accurately represent the importance of the corresponding segment without temporal consistency constraints. And the LSTM-based methods often suffer from the low variation problem in prediction probabilities, which would have a restricted impact while generating the final summary [12].

To solve the problem of temporal consistency, a typical shot-level method predicts the selection scores on the segmented shots rather than on each frame, which facilitates exploiting temporal similarities and dependencies within a video [13–16]. However, there are two disadvantages for the shot-level strategy: (1) the evaluation form is relatively simple and all the frames in the same segment are given the same importance score, which results in a lack of diversity while generating summaries; (2) the importance score predicted by this method is not sufficiently accurate due to the errors caused by the prediction. And the unimportant frames in the segment may be given higher scores.

Thus, the shot-level and frame-level methods are regarded as two different methods, and they are usually used individually. In order to separate the subtask of temporal consistency problem from the prediction of the frame-level importance scores and refine the importance scores of each frame under the constraints of temporal interest segments, we propose an attention-based stacking ensemble approach for supervised video summarization to learn the underlying contact between the two methods based on our previous work [17]. Inspired by the temporal interest proposals strategy proposed by [15] and the sequence labeling formulation in [3], the proposed model first encodes the features via a self-attention mechanism. Then the encoded features are fed into two predictors to predict the interest segments scores and the frame-level importance probabilities simultaneously as the intermediate features. Finally, the intermediate features are combined as the input to a soft decision fusion model to estimate the final scores of each frame in the video.
The main innovations and contributions of our video summarization method are as follows.

- A stacking ensemble approach is proposed for supervised video summarization to learn the potential complementarity between segment and frame partition.
- A joint loss function is proposed to train the stacking model with the interest proposal label, the important frame label and the important score label.
- To the best of our knowledge, this work is the first attempt to fuse the frame-level and shot-level strategies for video summarization.

2 APPROACH

The approach uses soft decision fusion to combine key frame probabilities and temporal interest segments. As shown in Figure 1, given a video \( v \) with \( T \) frames, a pre-trained backbone network is used to extract the visual features for each video frame. Following the previous methods [1, 3, 4], we uniformly down-sample the videos to 2 fps. Then we take the output of the pool5 layer in the pretrained GoogLeNet [18] as the feature descriptor for each video frame. The dimensionality of the feature descriptor is 1024.

After feature extraction, the video feature sequence is denoted as \( X \in \mathbb{R}^{d \times T} \), where \( d \) is the feature dimension. To capture the temporal long-range dependencies, we use self-attention mechanism [19] and skip connection to re-encode frame features. And the final representation is obtained. Temporal average pooling \( 1d \) is then used in time dimension with the kernel size of 4, 8, 16, 32 and the stride of 1 to avoid temporal warping or cropping, which is called multiscale temporal pooling [15]. Next, the pooled features are defined as the input, and the ensemble stacking process begins with two branches as below.

1. The pooled features are fed into the temporal interest detection module to obtain the interest segments scores denoted as \( P_S = [\rho_{1,1}, ..., \rho_{l,m}, ..., \rho_{T,N}] \in \mathbb{R}^T \), where \( \rho_{t,n} \) is the interest score of the \( t \)-th frame being divided into the \( n \)-th segment in a total of \( N \) segments. Thus, the video \( v \) that consists of a sequence of consecutive frames is temporally divided into \( N \) disjoint segments, and the frames in each segment are assigned the same interest score.

2. The combination of the pooled features and the final representation features are put into the important frame labeling model to predict the key frame probabilities of each frame in video as \( P_K = [p_{k1}, ..., p_{kt}, ..., p_{kT}] \in \mathbb{R}^T \), where \( p_{kt} \) is the \( t \)-th frame’s probability of being selected as a key frame.

Finally, \( P_S \) and \( P_K \) are integrated into a soft decision fusion mechanism to predict the final frame-level scores as \( Y = [y_1, ..., y_t, ..., y_T] \in \mathbb{R}^T \), where \( y_t \) ranging from 0 to 1, is the \( t \)-th frame-level score of a video. The higher the score a frame obtains, the higher probability the frame will be selected into the final summary with.

The detailed structure of temporal interest detection, important frame labeling and soft decision fusion in the ensemble stacking will be given in 2.1, 2.2 and 2.3 respectively. In addition, 2.4 describes the learning strategy of our method in detail, and 2.5 presents the post-processing of generating the final video summary after outputting the frame-level scores.

2.1 Temporal interest detection

Inspired by [15], we directly adopt the temporal interest proposals generation strategy to generate temporal interest segments scores by pre-defined multi-scale intervals. E.g., at the \( t \)-th temporal location, \( K \) interest proposals are appointed with the fixed range \( [t - \lambda_k/2, t + \lambda_k/2], k = 1, 2, ..., K \), where \( \lambda_k \) is the duration of the
$k$-th interest proposal. Therefore, $K \times T$ interest proposals are totally produced in a video sequence with $T$ frames. In the training stage, a proposal is positive when its temporal intersection over Union (IoU) with any ground truth proposal is higher than 0.6, or negative with $0 \leq$ IoU $< 0.3$. The proposals within $0.3 \leq$ IoU $< 0.6$ are discarded in loss calculation. $\lambda_k$ is calculated by the conclusive equations in [15] as $\{4, 8, 16, 32\}$ to cover all ground-truth proposals with the durations from 1 to 44.

The details of the model are shown in Figure 2(a). $Fc$-1 attempts to connect different scaled pooling layers and can effectively prevent the overfitting. $Fc$-1 includes $tanh$ and layer-normalization. There are two sibling output subbranches following $Fc$-2. The first outputs importance scores of proposals, and the second one outputs the associated center and the proposals length offsets.

Finally, we use the non-maximum suppression (NMS) [20] to refine the proposals. Min-max normalization is implemented on the refined proposals, and the temporal interest segments scores is obtained.

### 2.2 Important frame labeling

Just like semantic segmentation indicating the semantic label of the corresponding pixels, important frame labeling model is formulated as a sequence labeling problem to indicate the semantic label of the corresponding frame in time dimension [3]. The details of the model are shown in Figure 2(b). $Fc$-3 attempts to map the multi-layer features into single layer. $Fc$-4, including linear layer and softmax function, attempts to predict the frame-level probabilities with the output dimension of $T \times C$, where $T$ is the number of frames, and $C$, the dimension of the output channel, is 2 since we need scores corresponding to 2 classes (keyframe or non-keyframe) for each frame.

### 2.3 Soft decision fusion

Temporal interest segments scores and frame-level probabilities are integrated as intermediate features into a soft decision fusion mechanism to predict the final frame-level scores. In this model, the most common method is to average the two scores with the same weight. To introduce more nonlinearity and achieve higher fusion performance, we use a simple multilayer perceptron as the meta-learner [21] that utilizes the two intermediate features to fit final ground-truth frame-level scores.

### 2.4 Learning

Multi-task loss is proposed to train the model jointly. The objective function is obtained as

$$L = L_{cls} + L_{reg} + L_{pre} + L_{mse}$$

where $L_{cls}$ and $L_{reg}$, accounting for the temporal interest detection module, are the classification loss and the regression loss respectively (which use the interest proposal label), $L_{pre}$ is the prediction loss of the frame-level probabilities for important frame labeling (which uses the important frame label), and $L_{mse}$ is the fitting loss of the meta-learner (which uses the important score label). The four losses are considered to be equally important.

Specifically, focal loss [22] is used to obtain more accurate detection of the interesting proposals as

$$L_{cls} = -\frac{1}{N} \sum_{m=1}^{N} (1 - p_t(m))^\gamma \log(p_t(m))$$

where $M$ is the total number of predicted proposals (including positive and negative proposals), and $p_t(m)$ is the probability of the $n$-th proposal being classified into the corresponding ground-truth interest proposal labels, which is actually represented by the softmax function. $\gamma$ is a hyperparameter that is artificially set to be 1.

The regression loss $L_{reg}$ is the positioning regression for the positive interest proposal, and is obtained as

$$L_{reg} = \frac{1}{N_{pos}} \sum_{i=1}^{N_{pos}} p_t(i) \frac{1}{Q} \sum_{q=1}^{Q} \text{smooth}_{L_1}(t(q) - t_q^* (q))$$

where $N_{pos}$ is the number of positive interest proposals, $p_t(i)$ is the probability of the $i$-th positive interest proposals being classified into the corresponding ground-truth interest proposal labels, $t(q) = (\delta c_q, \delta l_q)$ is the predicted (ground-truth label) $i$-th group of positioning regression, and each group contains $Q$ parameters. The regression loss of positioning regression is calculated by the smooth$_{L_1}$ loss.

Specifically, the predicted location offset $t_q = (\delta c_q, \delta l_q)$ contains the center position and the length offsets between the generated proposals and the pre-defined proposals. The ground truth localization offset $t_q^* = (\delta c_q^*, \delta l_q^*)$ is obtained as follows.

$$\delta c_q^* = (c_q^* - c_q)/l_q, \delta l_q^* = \ln(l_q^*/l_q)$$

where $c_q^*$ and $l_q^*$ are the center location and the length of the ground truth shot respectively, $c_q$ and $l_q$ are the center location and the length of the $i$-th positive interest proposals respectively.

For the prediction loss of frame-level importance scores, the weighted focal loss is proposed because the categories of temporal important frames and non-important frames are extremely unbalanced, which is obtained as

$$L_{pre} = -\frac{1}{T} \sum_{i=1}^{T} \omega(i) (1 - p_t(i))^\gamma \log(p_t(i))$$

where $T$ is the total number of temporal frames, $p_t(i)$ is the probability of the $i$-th frame being classified into the corresponding ground-truth label as shown in Eq. 3, $\omega(i)$ is the category weight of the ground-truth classification ($\omega$) of the $i$-th frame, and $\omega = \text{median}_\text{freq}$, where $\text{freq}$ is the number of frames with ground-truth label divided by the total number of frames in the video, and $\text{median}_\text{freq}$ is the median of the computed frequencies.

The mean square error metric is adopted to be the fitting loss $L_{mse}$ of the meta-learner, and the loss function is

$$L_{mse} = \|y - \hat{y}\|_2^2$$

where $y$ is the ground-truth frame-level scores vector, and $\hat{y}$ is the meta-learner’s output frame-level scores vector.
Table 1: Comparisons of F-Score (%) and parameters (million) with state-of-art video summarization methods on the SumMe and TVSum datasets under the Canonical (C), Augmented (A) and Transfer (T) settings, respectively

| Method            | SumMe C | SumMe A | SumMe T | TVSum C | TVSum A | TVSum T | Params |
|-------------------|---------|---------|---------|---------|---------|---------|--------|
| vsLSTM [1]        | 37.6    | 41.6    | 40.7    | 54.2    | 57.9    | 56.9    | 2.63   |
| dppLSTM [1]       | 38.6    | 42.9    | 41.8    | 54.7    | 59.6    | 58.7    | 2.63   |
| FCSN [3]          | 48.8    | 50.2    | 45.0    | 58.4    | 59.1    | 57.4    | 116.49 |
| VASNet [4]        | 49.7    | 51.1    | -       | 61.4    | 62.4    | -       | 7.35   |
| SUM-GAN [6]       | 41.7    | 43.6    | -       | 56.3    | 61.2    | -       | -      |
| DR-DSN [7]        | 42.1    | 43.9    | 42.6    | 58.1    | 59.8    | 58.9    | 2.63   |
| M-AVS [10]        | 44.4    | 46.1    | -       | 61.0    | 61.8    | -       | -      |
| DSNet [15]        | 50.2    | 50.7    | 46.5    | 62.1    | 63.9    | 59.4    | 8.53   |
| SABTNet [16]      | 50.7    | -       | -       | 61.0    | -       | -       | 6.31   |
| SASUMsup [27]     | 45.3    | -       | -       | 58.2    | -       | -       | 44.07  |
| [28]              | 51.1    | 52.1    | 45.4    | 61.0    | 61.5    | 55.1    | -      |
| DHAVS [29]        | 45.6    | 46.5    | 43.5    | 60.8    | 61.2    | 57.5    | -      |
| [30]              | 51.7    | 51.0    | 44.1    | 61.5    | 61.2    | 58.9    | -      |
| [31]              | 51.7    | -       | -       | 59.6    | -       | -       | -      |
| CSNetsup [32]     | 48.6    | 48.7    | 44.1    | 58.5    | 57.1    | 57.4    | -      |
| SEVS (Ours)       | 51.8    | 51.4    | 46.7    | 62.9    | 63.3    | 59.0    | 4.33   |

Table 2: The diversity score of generated summaries on the SumMe and TVSum datasets

| Dataset | dppLSTM | DR-DSN | DSNet | SEVS (Ours) |
|---------|---------|--------|-------|-------------|
| SumMe   | 0.591   | 0.594  | 0.642 | 0.676       |
| TVSum   | 0.463   | 0.464  | 0.476 | 0.473       |

Notes: The higher the score, the better the diversity of the summary video.

2.5 Key-shot selection
The Kernel Temporal Segmentation (KTS) algorithm [23] is adopted on the video sequence to calculate the number of shots in the video and their region (i.e., the start and end). And the shot-level importance score is obtained by averaging the final scores output by meta learner in the same shot. Finally, the video summaries are produced under the constraint that the total length of selected shots is no more than 15% of the original video length for a fair comparison with the reference methods, which is implemented by Knapsack algorithm.

3 EXPERIMENTS AND ANALYSIS
3.1 Datasets and evaluation metric
We evaluate our stacking ensemble approach for video summarization (SEVS) on two public benchmark datasets, i.e., TVSum [24] and SumMe [25]. Up to 39 videos from the YouTube dataset and 50 videos from the Open Video Project (OVP) dataset [26] are used to augment the training data. Three settings as suggested in [1] are adopted to evaluate the method as following. (1) Canonical. (2) Augmented. (3) Transfer. Training and testing datasets in all the settings are randomly divided into 5 splits, and the average performance of the 5 runs is achieved. In addition, F-score are used as the metric to assess the similarity between the generated summaries and the ground truth summaries, and diversity score is used to assess the diversity performance of the generated video summaries.

3.2 Experiments
3.2.1 Implementation Details. The non-maximum suppression threshold is set as 0.5 for the final result presentation. Our model is trained over 300 epochs by using Adam optimizer with a base learning rate of $5 \times 10^{-5}$ and a weight decay of $10^{-5}$. All the experiments are conducted on a Nvidia GTX 1660Ti GPU and implemented by PyTorch.

3.2.2 Results and Comparisons. Table 1 shows the comparison results, where the performance of the other methods is obtained from the corresponding references. Among them, DSNet [15] is a classic temporal interest detection model and FCSN [3] is a model based on sequence labeling formulation, DR-DSN [7] is an unsupervised method based on reinforcement learning, and the other methods are state-of-the-art supervised methods in recent years. It can be seen that the performance of the canonical tests for our model on the two basic datasets is superior to the other methods. Specifically, the F-score on TVSum is increased by at least 1.3%. It is worth noting that the parameters of our model are fewer than those of most of the models in Table 1.

It can also be seen that our method achieves competitive performance compared with the state-of-the-art methods in augmented...
Table 3: F-score results (%) via ablation studies about interest segments, frame-level probabilities, averaging and meta-learner strategy

| segments | frame | Avg. | meta | SumMe | TVSum |
|----------|-------|------|------|-------|-------|
| ✓        | ✓     | ✓    | ✓    | 50.1  | 49.7  | 44.6  | 62.6  | 62.4  | 58.7  |
| ✓        | ✓     |       | ✓    | 47.0  | 48.6  | 45.9  | 61.6  | 61.9  | 58.5  |
| ✓        | ✓     | ✓    | ✓    | 49.8  | 50.6  | 46.5  | 62.8  | 63.0  | 58.2  |
| ✓        | ✓     | ✓    | ✓    | 51.8  | 51.4  | 46.7  | 62.9  | 63.3  | 59.0  |

Figure 3: Comparative step lines via ablation studies about interest segments, frame-level probabilities, averaging and meta-learner strategy

Figure 4: Parameter analysis of the NMS threshold on the SumMe and TVSum datasets

and transfer settings. Among all the methods, our ensemble model significantly outperforms the two typical methods, i.e., DSNet and FCSN [3], which validate the effectiveness of our model.

In addition, the diversity performance of our method is assessed on two datasets, and the results are shown in Table 2. Our method achieves the highest score on SumMe. Compared with DSNet, our method achieves an improvement of 5.5% in the diversity score.

3.2.3 Ablation studies and analysis. Ablation experiments are conducted to verify the effectiveness of our method, and the results are shown in Table 3. The experiments are tested separately for the two branches, i.e., temporal interest segments and frame-level probabilities in our model. Obviously, in the single branch test, only part of the loss function is used for model training, i.e., $L_{cls}$ and $L_{reg}$ in (1) for temporal interest segments, and $L_{pre}$ in (1) for frame-level probabilities. We also compare different soft decision fusion strategies, i.e., averaging the two scores with the same weight (the corresponding loss function in training is $L_{cls} + L_{reg} + L_{pre}$) and meta-learner strategy (ours). It can be seen that our method achieves the highest performance in three settings since the concurrent learning of segment features and frame-level features with the proposed multi-task loss function makes the model more robust.

It can also be seen from Figure 3 that our method can better fit the ground-truth important scores curve compared with other strategies in the ablation test, which means that our meta-learner strategy could refine the importance scores of each frame under the constraints of temporal interest segments. The results verify the complementarity between frame level diversity and shot level temporal consistency.

The results of the parameter analysis of the NMS threshold on the SumMe and TVSum datasets are shown in Figure 4. By comparing the F-score and balancing inference time consumption, 0.5 is specified as the final threshold.

4 CONCLUSIONS
A stacking ensemble approach for video summarization is proposed to exploit the complementarity between frame level diversity and shot level temporal consistency. Comparative experimental results demonstrate the superiority of our model, while a series of ablation experiments show that interest the segments scores and the frame-level probabilities are potentially correlated, their fusion and the corresponding multi-task loss function can result in a more accurate prediction than a single method. In the next step, ensemble learning algorithms for video summarization will be further explored.

REFERENCES
[1] K. Zhang, W.-L. Chao, F. Sha, and K. Grauman. 2016. Video summarization with long short-term memory. in Proc. ECCV. Amsterdam, Netherlands, pp. 766–782. doi: 10.1007/978-3-319-46478_747.
[2] Yi-Nung Chung, Tun-Chang Lu, Ming-Tsung Yeh, Yu-Xian Huang, and Chun-Yi Wu. June 2015. Applying the Video Summarization Algorithm to Surveillance Systems. Journal of Image and Graphics, Vol. 3, No. 1, pp. 20-24. doi: 10.18178/joig.3.1.20-24.
[3] M. Rochan, L. Ye, and Y. Wang. 2018. Video summarization using fully convolutional sequence networks. in Proc. ECCV Munich, Germany, pp. 347–363.
[4] J. Fajtl, H. S. Sokeh, V. Argyriou, D. Monekosso, and P. Remagnino. 2018. Summarizing videos with attention. in Proc. ACCV. Perth, Australia, pp. 39–54. doi: 10.1007/978-3-030-21074-9_4.

[5] S. Huang, X. Li, Z. Zhang, F. Wu, and J. Han. Jun. 2019. User-ranking video summarization with multi-stage spatio–temporal representation. IEEE Trans. Image Process. vol. 28, no. 6, pp. 2654–2664. doi: 10.1109/TIP.2018.2889265.

[6] B. Mahasseni, M. Lam, and S. Todorovic. Jul. 2017. Unsupervised video summarization with adversarial LSTM networks. in Proc. CVPR. Hawaii, USA, pp. 2982-2991. doi: 10.1109/CVPR.2017.318.

[7] K. Zhou, Y. Qiao, and T. Xiang. 2018. Deep reinforcement learning for unsupervised video summarization. in Proc. AAAI, New Orleans, Louisiana, USA, pp. 7582–7589.

[8] M. Ma, S. Mei, S. Wan, Z. Wang, D. D. Feng and M. Bennamoun. Oct. 2021. Similarity Based Block Sparse Subset Selection for Video Summarization. IEEE Transactions on Circuits and Systems for Video Technology. vol. 31, no. 10, pp. 3967-3980. doi: 10.1109/TCSVT.2020.3044608.

[9] Y. Zhang, R. Tao and Y. Wang. Jun. 2017. Motion-State-Adaptive Video Summarization via Spatiotemporal Analysis: in IEEE Transactions on Circuits and Systems for Video Technology. vol. 27, no. 6, pp. 1340-1352. doi: 10.1109/TCSVT.2016.2539606.

[10] Z. Ji, K. Xiong, Y. Pang and X. Li. Jun. 2020. Video Summarization With Attention-Based Encoder–Decoder Networks. IEEE Transactions on Circuits and Systems for Video Technology. vol. 30, no. 6, pp. 1709-1717. doi: 10.1109/TCSVT.2019.2904996.

[11] C. Huang and H. Wang. Feb. 2020. A Novel Key-Frame Selection Framework for Comprehensive Video Summarization. in IEEE Transactions on Circuits and Systems for Video Technology. vol. 30, no. 2, pp. 577-589. doi: 10.1109/TCSVT.2019.2890899.

[12] E. Apostolidis, E. Adamantidou, A. I. Metsai, V. Mezaris and I. Patras. Aug. 2021. AC-SUM-GAN: Connecting Actor-Critic and Generative Adversarial Networks for Unsupervised Video Summarization. IEEE Transactions on Circuits and Systems for Video Technology. vol. 31, no. 8, pp. 3278-3292. doi: 10.1109/TCSVT.2020.3037883.

[13] B. Zhao, X. Li, X. Lu. 2018. HSA-RNN: hierarchical structure-adaptive RNN for video summarization. in Proc. CVPR. Salt Lake City, Utah, USA, pp. 7405–7414. doi: 10.1109/CVPR.2018.00773.

[14] K. Zhang, K. Grauman, F. Sha. 2018. Category-specific video summarization. IEEE Transactions on Circuits and Systems for Video Technology. vol. 30, pp. 948–962. doi: 10.1109/TCSVT.2020.3039886.

[15] W. Zhu, J. Lu, J. Li and J. Zhou. 2021. DSNet: A Flexible Detect-to-Summarize Network for Video Summarization. IEEE Trans. Image Process. vol. 30, pp. 948-962. doi: 10.1109/TIP.2020.3039886.

[16] Hao Fu, Hongxing Wang. 2021. Self-attention binary neural tree for video summarization. Pattern Recognition Letters. vol. 143, pp. 19-26. doi: 10.1016/j.patrec.2020.12.016.

[17] Y. An and S. Zhao. A Video Summarization Method Using Temporal Interest Detection and Key Frame Prediction. arXiv preprint. doi: 10.48550/arXiv.2109.12581.

[18] C. Saeredy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich. 2015. Going deeper with convolutions. in Proc. CVPR. Boston, MA, USA, pp. 1-9. doi: 10.1109/CVPR.2015.7298594.

[19] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin. 2017. Attention is all you need. in Proc. NIPS. Long Beach, CA, USA, pp. 5999-6009.

[20] S. Ren, K. He, R. Girshick, J. Sun. 2015. Faster R-CNN: Towards real-time object detection with region proposal networks. in Proc. NIPS. Montréal, CANADA, pp. 91-99.

[21] Breiman, Leo. 1996. Stacked Regressions. Machine Learning. Vol. 24(1), pp. 49-64. doi: https://doi.org/10.1023/A:1018046112532.

[22] T. Lin, P. Goyal, R. Girshick, K. He and P. Dollar. 2017. Focal Loss for Dense Object Detection. in Proc. ICCV. Venice, Italy, pp. 2999-3007. doi: 10.1109/ICCV.2017.324.

[23] D. Potapov, M. Douze, Z. Harchaoui and C. Schmid. 2014. Category-specific video summarization. in Proc. ECCV. Zurich, Switzerland, pp. 540–555. doi: 10.1007/978-3-319-10599-4_35.

[24] Y. Song, J. Vallmitjana, A. Stent, A. James. 2015. TVSum: Summarizing web videos using titles. in Proc. CVPR. Boston, MA, USA, pp. 5179-5187. doi: 10.1109/CVPR.2015.7299154.

[25] M. Googli, H. Grabner, H. Riemenschneider, L. Van Gool. 2014. Creating summaries from user videos. in Proc. ECCV. Zurich, Switzerland, pp. 505-520. doi: 10.1007/978-3-319-10554-9_33.

[26] S. E. F. De Avila, A. P. B. Lopes, A. Da Luz Jr., A. De Albuquerque Aratijo. Jan. 2011. VSUMM: A mechanism designed to produce static video summaries and a novel evaluation method. Pattern Recognition Letters. vol. 32, no. 1, pp. 56–68. doi: 10.1016/j.patrec.2010.08.004.

[27] H. Wei, B. Ni, Y. Yan, H. Yu, X. Yang, and C. Yao. 2018. Video summarization via semantic attended networks. in Proc. AAAI, New Orleans, Louisiana, USA, pp. 216–223.

[28] Wencheng Zhu, Jwen Lu, Yucheng Han and Jie Zhou. Feb. 2022. Learning multi-scale hierarchical attention for video summarization. Pattern Recognition. vol. 122, doi: 10.1016/j.patcog.2021.108312.

[29] Y. Jung, D. Cho, D. Kim, S. Woo, I.S. Kweon. 2019. Discriminative feature learning for unsupervised video summarization. in Proc. AAAI, Hawaii, USA, pp. 8537-8544.