Towards Visually Explaining Video Understanding Networks with Perturbation

Zhenqiang Li[1,2], Weimin Wang[2]*, Zuoyue Li[3], Yifei Huang[1], and Yoichi Sato[1]

[1]Institute of Industrial Science, The University of Tokyo

[2]National Institute of Advanced Industrial Science and Technology   [3]ETH Zürich

{lzq,hyf,ysato}@iis.u-tokyo.ac.jp

{li.zhenqiang,weimin.wang}@aist.go.jp

li.zuoyue@inf.ethz.ch

Abstract

"Making black box models explainable" is a vital problem that accompanies the development of deep learning networks. For networks taking visual information as input, one basic but challenging explanation method is to identify and visualize the input pixels/regions that dominate the network's prediction. However, most existing works focus on explaining networks taking a single image as input and do not consider the temporal relationship that exists in videos. Providing an easy-to-use visual explanation method that is applicable to diversified structures of video understanding networks still remains an open challenge. In this paper, we investigate a generic perturbation-based method for visually explaining video understanding networks. Besides, we propose a novel loss function to enhance the method by constraining the smoothness of its results in both spatial and temporal dimensions. The method enables the comparison of explanation results between different network structures to become possible and can also avoid generating the pathological adversarial explanations for video inputs. Experimental comparison results verified the effectiveness of our method.

1. Introduction

Deep neural networks have achieved remarkable performance in various tasks [26, 6, 36, 39, 17, 3, 15, 16]. Besides good results, sometimes people may also naturally concern about the interpretability of a network, *i.e.*, why a certain prediction is derived by a network for a given input. As one direction towards interpreting these networks, visual explanation (also known as *attribution*) methods, which identify and visualize the contribution of each pixel/region of a given input to the output of a trained network, have attracted much attention recently [37, 29, 32, 4, 23, 1].
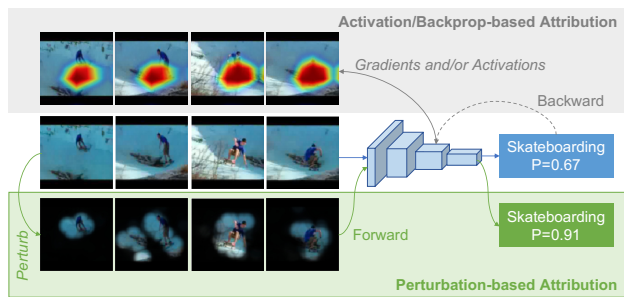
_____
*Corresponding author.

Figure 1. The visual explanation methods for deep neural networks could be generally divided into three categories. The upper block demonstrates the activation-based and backprop-based methods, which usually utilize activations or gradients extracted from the interior layers of a network to identify the significant places in the input frames. The lower block demonstrates the perturbation-based methods, who visually explain a black-box network by directly operating on the input and locating the area that affects the output most in a forward manner. In this paper, aiming at visually explaining video understanding networks via a model-agnostic method, we investigate the perturbation-based attribution method on video classification networks.

Recently, researchers begin to focus on the visual explanation methods of video understanding networks. Most existing works on visual explanation methods are concentrated on individual images [32, 27, 38, 20, 23, 11, 10]. Directly applying these methods to videos usually cannot obtain satisfactory explanation results since it is difficult for these methods to handle the complex nonlinear and even recurrent spatiotemporal dependencies in videos. Although there are a few works [1, 31] turning on the visual explanation methods for video processing networks, they are designed specifically for only a fixed type of network (*e.g.* 3D-CNNs or CNN-RNN) and cannot be generalized to other networks.

The perturbation-based approach [11, 10] is a promising direction for visually explaining video understanding networks since it is agnostic to the network structure. As il-

lustrated in Fig. 1, the perturbation-based method operates on the input and then observes changes in the model outputs. Through iteratively adjusting preserved pixels/regions in the input video and observing the effect on the output of the network, the perturbation-based methods aim to find a small subset of the input, with the preservation of which a large output value can be still retained. However, one cannot directly apply the previous perturbation-based approaches (*e.g.* [11, 10]) that were applied to individual images to video understanding networks. This is because pathological explanation results caused by the adversarial effect would be easily produced if no smoothness constraint on the temporal dimension is incorporated [18, 33, 11, 10].

In this paper, we propose a generic method for visually explaining video understanding networks by incorporating a perturbation-based method enhanced by a spatiotemporal smoothness loss function. This method can be easily applied to any video understanding networks without detailed architectural knowledge. Furthermore, the loss function exploits the spatiotemporal dependencies between frames to generate explanation results smoothed in both temporal and spatial dimensions and thus avoid the pathological adversarial explanations for video understanding networks. The contribution of this paper is three-fold:

- We are the first to introduce perturbation-based method to visually explain video understanding networks. The proposed method can be easily utilized on any diversified and complicated video understanding networks.

- We introduced a novel loss function to regularize the spatiotemporal smoothness of the visual explanation results derived by the perturbation-based method.

- Our experimental results verified that the proposed perturbation-based method equipped with our loss function could achieve competitive performances on multiple datasets.

2. Related Work

In this section, we introduce existing attribution approaches for the visual explanation, including methods mainly focused on networks for individual images (referred to as 'image attribution method' below), as well as methods especially proposed for video understanding networks (noted as 'video attribution method' below).

2.1. Image Attribution Approaches

The goal of an image attribution method is to tell us which elements of the input (*e.g.*, pixels or regions for an image input) are responsible for its output (*e.g.*, the softmax probability for a target label in the image classification task). The results are commonly expressed as an *importance map* in which each scalar quantifies the contribution of the element in the corresponding position. Differing from the attention mechanism that is commonly embedded in a deep network to enhance performance by removing redundancies in input, attribution methods are applied to a model with fixed parameters to provide explanations.

Backpropagation-based (BP-based) methods are established upon a common view that gradients (of the output with respect to the input) could highlight key regions in the input since they characterize how much variation would be triggered on the output by a tiny change on the input. [5] and [25] have shown the correlation between the pixels' importance and their gradients for a target label. However, the importance map generated by raw gradients is typically visually noisy. The way to overcome this problem could be partitioned into three branches. DeConvNets [37] and Guided Backprop [29] modify the gradient of the ReLU function by discarding negative values during the back-propagation calculation. Integrated Gradients [32] and SmoothGrad [27] resist noises by accumulating gradients. However, methods based on unmodified backpropagation tend to capture the average properties of the network, and thus are difficult to obtain class-discriminative explanation results. LRP [4], DeepLift [24] and Excitation Backprop [38] propose modified backpropagation schemes to overcome this challenge. But a modified backpropagation scheme may also limit the user-friendliness of the method, because not all operations used in networks may be compatible by the scheme.

Activation-based methods generate the importance map by linearly combining the activation maps taken from the intermediate convolutional layer of a network. Different methods vary in the choice of combining weights. CAM [40] selects parameters on the fully-connected layer as weights, while Grad-CAM [23] produces the weight by average pooling the gradients from the output to the activation. Grad-CAM++ [7] replaces the average pooling in Grad-CAM with coefficients calculated by second derivative. Since the intermediate activation maps are likely to have a lower resolution than input images on networks with pooling layers, the visual explanation results derived by these methods tend to be coarse-grained. For example, when these methods are utilized to visually explain 3D-CNNs with temporal pooling layers, adjacent input frames will be allocated with the same visual explanation result.

Perturbation-based methods start from an intuitive assumption that the change of outputs could reflect the importance of certain elements when they are removed or preserved only in the input. However, in order to find the optimal results, theoretically it is necessary to traverse the elements and their possible combinations in the input and ob-

serve their impact on the output. Due to the high time cost of this traversal process, how to obtain an approximate optimal solution faster is the research focus of this problem. Occlusion [37] and RISE [20] perturb an image by sliding a grey patch or randomly combining occlusion patches, respectively, and then use changes in the output as weights to sum different patch patterns. LIME [22] approximates networks into linear models and uses a super-pixel based occlusion strategy. Meaningful perturbation [11] converts the problem to an optimization task of finding a preservation mask that can maximize the output probability under the constraints of area ratio and smoothness. Real-time saliency [8] learns to predict a perturbation mask with a second neural network. Qi *et al.* [21] improved the optimization process by introducing integrated gradients and Wagner *et al.* [35] introduced certain restrictions in the optimization process to avoid adversarial results. Fong *et al.* [10] introduced the extremal perturbation scheme and a special smooth mask to solve the problem of imbalance between several constraining terms.

2.2. Video Attribution Approaches

The goal of the video attribution is to obtain the regions taken important by a network of the input, in both spatial and temporal dimensions. The increase of dimension means inflated searching space and time cost. [12] and [2] respectively applied pure gradients and LRP to ground the input regions taken important by a video understanding network. However, directly utilizing image attribution methods to videos is likely to obtain unsatisfactory explanation results since spatiotemporal dependencies between frames are not considered by these methods. EB-R (excitation backprop for RNNs) [1] firstly extended the Excitation Backprop attribution method to the framework for videos, to be specific, the CNN-RNN structure. Grad-CAM [23] is inherently applicable to network processing videos. [31] and [30] adapt activation-based methods for 3D convolutional networks to produce visualization results over time. However, both the EB-R and Grad-CAM family cannot treat the network totally as a real black-box, since they have to take the gradients or activations from the interior of a network, which obstructs their application on increasingly complicated and diversified video understanding networks.

3. Proposed Approach

In this section, we present the perturbation-based visual explanation method for video understanding networks. Let $\boldsymbol{X} = \{\boldsymbol{x}_t\}_{t=1}^T, \boldsymbol{x}_t \in \mathbb{R}^{H \times W \times 3}$ represents a video of T frames with width W and height H. The proposed method is investigated on a function Φ that maps the image sequence to a softmax probability $\Phi_c(\boldsymbol{x}) \in \mathbb{R}$ for a given target class with the index of c among all C classes. The goal of video attribution methods is to derive a sequence

of importance maps $\boldsymbol{M} = \{\boldsymbol{m}_t\}_{t=1}^T$ which assign to each pixel $x_{i,j,t}$ a value $m_{i,j,t} \in [0, 1]$. Here i, j refer to the spatial location of each pixel.

3.1. Perturbation-based visual explanation

The *preservation* version of the perturbation-based attribution method [11] is to find a reserving subset of the input which is as small as possible while retaining the prediction accuracy on a specified target label. The optimization target can be formulated as follows when the method is applied to a network taking an image $\boldsymbol{x}$ as input.

$$\boldsymbol{m}^* = \underset{\boldsymbol{m}}{\operatorname{argmin}}\{\lambda||\boldsymbol{m}||_1 - \Phi_c(\boldsymbol{m} \otimes \boldsymbol{x})\}, \qquad (1)$$

where $\boldsymbol{m}$ is the perturbation mask which has the same shape as the input image $\boldsymbol{x}$, $||\cdot||_1$ denotes the L_1 norm for matrices, λ controls the scope of regularization, and $\otimes$ represents the local perturbation operation on the input image according to the mask. The operation can be mathematically written as $\boldsymbol{m} \otimes \boldsymbol{x} = \boldsymbol{m} \cdot \boldsymbol{x} + (1 - \boldsymbol{m}) \cdot (k * \boldsymbol{x})$, where $\cdot$ denotes the Hadamard multiplication, $*$ represents convolution, and k denotes a kernel for Gaussian blur. The first item in Eq. 1 constrains the preservation ratio on the input image to be small while the second item encourages the model's prediction accuracy to be as high as possible.

However, the balance between the two constraint targets is difficult to control, which thus makes it difficult to obtain an optimal solution. In order to alleviate these two negative impacts, we take advantage of the extremal perturbation (**EP**) according to [10], which decomposes the optimization procedure into two steps. The first step finds a mask that maximizes the output probability under a constrained preservation ratio a, *i.e.*,

$$\boldsymbol{m}_a = \underset{\boldsymbol{m}:||\boldsymbol{m}||_1=aHW}{\operatorname{argmax}} \Phi_c(\boldsymbol{m} \otimes \boldsymbol{x}). \qquad (2)$$

The second step sets the lowest bound Φ_0 for the output probability, and searches for the smallest mask achieving this bound, *i.e.*, finds the smallest preservation ratio a^* as

$$a^* = \min\{a : \Phi_c(\boldsymbol{m}_a \otimes \boldsymbol{x}) \geq \Phi_0\}. \qquad (3)$$

The final extremal solution $\boldsymbol{m}_{a^*}$ is therefore obtained.

Eq. 2's optimization is commonly solved by stochastic gradient descent (SGD) method. In order to constrain the masks' preservation ratio to approach the setting target a, Eq. 2 is also adjusted by a loss function that regularizes the mask, which enforces values ranked in the top a to be close to 1 and the remaining values to be close to 0, *i.e.*,

$$\boldsymbol{m}_a = \underset{\boldsymbol{m}}{\operatorname{argmin}}\{\lambda||\text{vecsort}(\boldsymbol{m}) - \boldsymbol{r}_a||^2 - \Phi_c(\boldsymbol{m} \otimes \boldsymbol{x})\}, \quad (4)$$

where $\text{vecsort}(\boldsymbol{m}) \in \mathbb{R}^{HW}$ is a vector in which all the values of a mask are sorted and arranged in a descending order and $\boldsymbol{r}_a$ is a template vector consisting of aHW ones followed by $(1 - a)HW$ zeros.

3.2. Spatio-temporal perturbations for videos

The preservation ratio, which determines how many pixels will be preserved in the input, is a key constraint in the optimization procedure of extremal perturbation. In this paper, we firstly extend the original spatial constraints to the spatiotemporal dimensions, so we can constrain the overall preservation ratio on all the frames. Furthermore, in order to obtain the result with better spatiotemporal smoothness, we introduce an effective loss function, which can consider the inter-frame relationship in the optimization.

3.2.1 Extremal perturbation 3D

When adopting extremal perturbations to video cases, we constrain the overall preservation ratio in the whole spatiotemporal dimensions, *i.e.*, the distribution of all masks' values is optimized together in a 3D space. We refer to this method as Extremal Perturbation 3D (**EP-3D**). Hence, the energy function for optimizing masks under a preservation ratio constraint of v could be represented as

$$\boldsymbol{M}_v = \underset{\boldsymbol{M}:\sum_{t=1}^{T}||\mathbf{m}_t||_1=vTHW}{\operatorname{argmax}} \Phi_c(\boldsymbol{M} \otimes \boldsymbol{X}). \qquad (5)$$

Then we can transfer Eq. 4 as below for optimizing by SGD.

$$\boldsymbol{M}_v = \operatorname{argmin}\{\lambda||\operatorname{vecsort}(\boldsymbol{M}) - \boldsymbol{r}_v||^2 - \Phi_c(\boldsymbol{M} \otimes \boldsymbol{X})\}, \qquad (6)$$

where $\boldsymbol{r}_v$ is a template vector consisting of $vTHW$ ones followed by $(1-v)THW$ zeros.

3.2.2 Spatio-temporal smoothness constraint

Neural networks are vulnerable to adversarial inputs, *e.g.*, images that are unrecognizable to humans may be recognized by networks as some objects with high confidence [19], and images that are modified in a way imperceptible to humans may mislead networks to have totally wrong predictions [14]. For perturbation-based methods derived from Eq. 1, since their optimization targets are similar to that for generating adversarial inputs [18, 13, 33], they are prone to producing pathological solutions that can cause the adversarial inputs. The smoothness constraint is a common method to alleviate the generation of pathological adversarial solution. For example, meaningful perturbations [11] incorporated an extra optimization item to regularize the shape of perturbation regions in masks, and extremal perturbations [10] proposed a special up-sampling operator based on 2D transposed convolution to generate masks with spatial smoothness. However, only smoothness in the spatial dimensions are constrained by these methods. When extending the perturbation-based method to the video input with an extra temporal dimension, it is also necessary to constrain the smoothness of perturbations in the temporal dimension.

One easy way to constrain mask temporally is to use higher-order differences between frames as an energy function, *e.g.*, using second-order differences to control the smoothness. However, it tends to constrain masks to be consistent which will lose the sensitivity to features on different frames. Considering the spatiotemporal dependencies between frames, a better idea is to smooth masks in spatial and temporal dimensions jointly rather than merely in temporal. Hence we introduce a special loss function L_K that aims to gather the high-value pixels in small regions into some predefined shape by doing 3D convolution on the importance maps $\boldsymbol{M}$. The K in L_K denotes the kernel used in convolution, which has a shape of $(H_K+1)\times(W_K+1)\times(T_K+1)$. The loss function can be defined as below,

$$L_K(\boldsymbol{M}) = ||\operatorname{vecsort}(\boldsymbol{M} * K) - \boldsymbol{r}_{v'}||^2, \qquad (7)$$

where $*$ denotes 3D convolution with stride. In the experiment, we use a ellipsoid kernel which defined as $\forall t \in \{0, ..., T_K-1\}, i \in \{0, ..., H_K-1\}, j \in \{0, ..., W_K-1\}$,

$$k_{t,i,j} = \begin{cases} 0, & (\frac{2t}{T_K-1}-1)^2+(\frac{2i}{H_K-1}-1)^2+(\frac{2j}{W_K-1}-1)^2>1; \\ 1, & \text{otherwise}, \end{cases} \qquad (8)$$

$$K_{i,j,t} = Z^{-1}k_{i,j,t}. \qquad (9)$$

Here $Z = \sum_{i,j,t} k_{i,j,t}$ is the normalization factor. The loss function regularizes the first v' high values in the convoluted masks $\boldsymbol{M} * K$ to be as close as possible to 1. v' denotes the expected proportion of value-one among all values in the convolution masks, which is calculated as $vTHW/Z$ to satisfy the constraint for preservation ratio in masks $\boldsymbol{M}$. In experiment, we set the size of the kernel K to be $11\times11\times7$ and convolution stride to be 11. This will guarantee the high-value pixels of the mask to be concentrative as much as possible, as well as retain the flexibility in shape to adapt to the spatiotemporal variance in the input frame sequence.

We call this method for visually explaining video understanding networks as **Saptio-Temporal Extremal Perturbation (STEP)** considering that it could get smoothing extremal perturbation results in both spatial and temporal dimensions.

4. Experiments and Results

4.1. Experiments setting

Video classification networks are characterized by complicated and various architectures. To experimentally compare different visual explanation methods for video classification networks, we adopt two kinds of representative structures: CNN-RNN and 3D-CNNs. Specifically, we select two exemplar networks, *i.e.*, VGG16LSTM [1] and R(2+1)D [34] under the two model structures, respectively. We validate methods with the two model structures on subsets of two video datasets: UCF101-24 and EPIC-Kitchens

since their bounding box annotations are (partially) available.

UCF101-24 [28] is a subset of the UCF101 dataset, containing 3207 videos of 24 classes that are intensively labeled with spatial bounding box annotations of humans performing actions. In our experiment, we trained a VGG16LSTM model and an R(2+1)D model on the UCF101-24 dataset by the training set defined in THUMOS13. To generate importance maps for evaluating different visual explanation methods, we randomly selected 5 videos on each category to form a test set of 120 videos in total.

EPIC-Kitchens [9] is a dataset for egocentric video recognition, where 39596 video clips segmented from 432 long videos are provided, along with action and object labels. We choose the top 20 classes with the most number of clips to form the EPIC-Object and EPIC-Action sub-datasets, and randomly selected 5 clips for each class to generate two test sets and used the remaining clips to train models. Bounding boxes for the ground-truth objects in EPIC-Objects are provided in 2fps. On the EPIC-Action task, we connect a randomly selected part of each clip with its adjacent background frame sequence, to form a set for testing the temporal localization performance of attribution methods.

4.1.1 Model training

We trained a VGG16LSTM model and an R(2+1)D model on every video classification task. We use the VGG16LSTM model as it is defined in [1] and we fine-tune it on each dataset. To avert the gradient vanishing, we block the gradient propagation on hidden states and take the average of softmax probabilities on all-time steps as the final prediction. For the R(2+1)D model, we use R(2+1)D-18 structure [34] and fine-tune the network upon its pre-trained parameters on Kinetics-400. Both in training and testing phases, we sample 16 frames as the input by splitting one video clip into 16 segments and selecting one frame in each split. The classification accuracy for each network on every task's test set is shown in Tab. 1. Notably, the accuracy on the UCF101-24 test set is nearly 100%. We think this is due to the models are pre-trained on the UCF101 datasets.

Table 1. Top 1 & 5 classification acc. of test networks

Acc.	UCF101-24	EPIC-Object	EPIC-Action
R(2+1)D	1.00 / 1.00	0.57 / 0.85	0.77 / 0.97
V16L	0.97 / 1.00	0.55 / 0.84	0.81 / 1.00

4.1.2 Evaluation metrics

As most previous works did, we mainly adopt the 'Pointing Game' metric to evaluate the effectiveness of different visual explanation approaches. This metric takes advantage of manually annotated bounding boxes to evaluate whether the importance maps generated by an attribution method could locate the "key" spatial regions or temporal segments, which is called spatial pointing game (**S-PT**) and temporal pointing game (**T-PT**) respectively. We perform the S-PT evaluation on the UCF101-24 and EPIC-Object test sets. Following [1], we set a tolerance radius of 7-pixel, *i.e.*, one hit is recorded if a 7-pixel radial circle around the maximum point in an importance map intersects the ground-truth bounding box. On the EPIC-Action test set, we evaluate methods by the T-PT metric, in which a hit is recorded only when the index of the frame with the highest importance value locates in the ground-truth segment. The metric is measured by the hit rate on all test samples with bounding box annotations.

4.1.3 Implementation details

Following [10], all masks are generated and optimized based on smaller seed masks $\bar{M} = \{\bar{m}_t\} \in \mathbb{R}^{\bar{H} \times \bar{W} \times T}$ and in our experiment we set $H = 7\bar{H}$ and $W = 7\bar{W}$. The seed masks are then up-sampled by the transposed convolution operation with the 2D smooth max kernel defined in [10]. We report our explanation results under a series of preservation ratio constraints, which is $\{0.02, 0.05, 0.1, 0.2\}$ for R(2+1)D and $\{0.05, 0.1, 0.2, 0.3\}$ for VGG16LSTM. We expect the redundant information could be removed and key regions could be located via small preservation ratios. Empirically, a larger preservation ratio will not arouse a significant increase in the quantitative results.
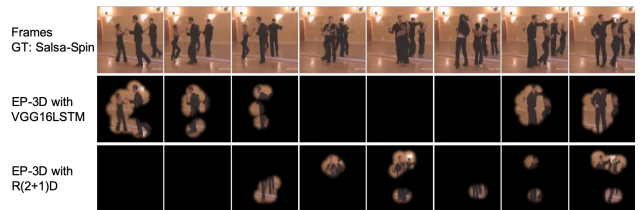
4.2. Comparison between 3D-CNNs and CNN-RNN

Figure 2. Visual explanation results comparison for two networks. The output probabilities on the two networks are guaranteed to be nearly equal when generating the results by EP-3D.

Table 2. Minimum preservation ratios without decreasing the output probabilities. For the same dataset, VGG16LSTM (V16L) need to preserve more regions than R(2+1)D.

Ratio	EPIC-Object	UCF101-24
R(2+1)D	0.064	0.170
V16L	0.157	0.280

One of the aims of investigating visual explanation methods is to understand the characteristics of networks, especially when we have multiple networks for the same task. Since we have constructed a generic visual explanation method and two networks for the task of video classification, it is natural to be curious about the difference of explanation results derived through the same method. Fig. 2 shows the visual comparison between explanation results obtained by EP-3D on R(2+1)D and VGG16LSTM. We uniformly sampled 8 frames for visualization. When obtaining the results, we ensure that perturbed videos have the same probability outputs under each network. It can be seen that VGG16LSTM relies on more input regions than R(2+1)D to derive equal output probabilities. For quantitative comparison, we calculated the minimum preservation ratio for each sample in a dataset without decreasing the output probabilities of the two networks and made an average for both datasets. The result, which can be seen in Tab. 2, is consistent with Fig. 2, where we found that VG16LSTM (V16L) requires more preservation regions to achieve a comparable output probability as R(2+1)D. We consider that the potential reason for the larger preservation in VGG16LSTM's results relies on its network architecture, where frame features are first extracted by VGG-16 and then forwarded into LSTM recurrently. If the preserved regions are too small in a certain frame, the feature vector for this frame would be close to zero and would become trivial for LSTM. As a result, VGG16LSTM tends to preserve more regions in order for higher prediction accuracy.

4.3. Comparison of existing attribution methods

We compare with three existing video attribution methods as baseline methods to validate the effectiveness of our proposed method.

- **Grad-CAM[23]**: A generic attribution method that *could be utilized on both 3D-CNNs and CNN-RNN networks*. For the R(2+1)D model, we generate the heatmaps based on the activation of the last 3d convolutional layer and upsample the maps to the shape of input images, in both spatial and temporal dimension. For the VGG16LSTM model, the heatmaps are generated based on the activation of $conv5$-layer of VGG16.

- **Saliency Tubes[31]**: A visualization method *specially designed for 3D-CNNs networks*. The activation maps of the last 3d convolutional layer are combined by weights in the final FC layer to produce heatmaps. We up-sample the maps as in Grad-CAM for visualization and evaluation.

- **EB-R[1]**: A backprop-based method *specially proposed for the CNN-RNN structure* which uses a modified back-propagation algorithm. We adopt it directly on our VGG16LSTM models and capture the

heatmaps for each frames at the $conv5$ layer of VGG16 as it was done in [1].

4.3.1 Qualitative results

Fig. 3 illustrates two groups of visualization results including the original frames and importance maps generated by different visual explanation approaches. Each group corresponds to one example video, and only 5 frames are sampled out of 16 input frames for visualization. For perturbation-based methods, we visualize the results generated under the preservation ratio constrain of 0.1. The first group shows a UCF101-24 video with the action label of Skateboarding and all visual explanation results are obtained on the R(2+1)D model. A video with the object label of Cupboard belonging to the EPIC-Object sub-dataset is showed as the second group, whose visual explanation results are all obtained on the VGG16LSTM.

There is a part of regions highly correlated with the ground-truth label in the two videos, *e.g.*, the first two frames of the left video example, and the middle three frames of the right example. STEP enhances EP-3D by a special loss function. It can be seen that STEP could generate temporally smooth mask sequences to preserve these related regions. In contrast, the regions preserved by the mask sequences generated by EP-3D perform lower consistency in the temporal dimension. They also tend to contain some regions showing no obvious relationship with the ground-truth label, such as the region of a red object in the second frame of the cupboard video. This reflects the effectiveness of L_K in improving the temporal smoothness of preservation masks. Moreover, combining with comprehensive observations of other samples' results, we discover that EP-3D tends to allocate more preserved regions on the head and tail frames for R(2+1)D and the front frames for VGG16LSTM.

On the R(2+1)D model, although stable results could be generated by Grad-CAM, they are coarse-grained in both spatial and temporal dimensions, which is the same for Saliency Tubes. This is because the two methods both generate importance maps based on the activation maps extracted from the intermediate convolutional layer, which have a lower resolution than the input frame sequence when the pooling operators are applied. On VGG16LSTM, EB-R could locate the ground-truth object while it also highlights the salient but unrelated regions.

4.3.2 Quantitative results

We then quantitatively compare the visual explanation results generated by different methods, using the spatial pointing game (S-PT) metric, which measures the percentage of importance maps whose maximum points fall into the annotation bounding boxes. When calculating the metric, only

| R(2+1)D on UCF101-24 | VGG16LSTM on EPIC-Object |

Frames

Grad-CAM

Saliency Tubes / EB-R
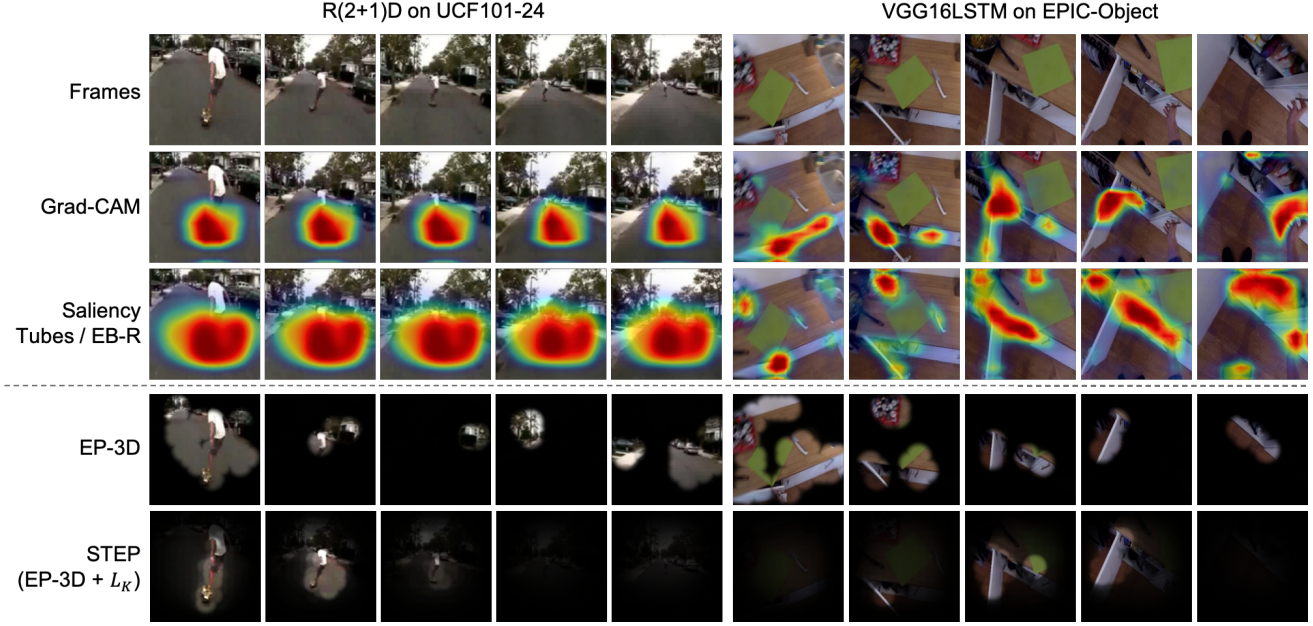
EP-3D

STEP (EP-3D + L_K)

Figure 3. Qualitative comparison of visual explanation results generated by baseline methods and the perturbation-based methods. The importance maps generated by our method could smoothly preserve the regions associated with the ground-truth label and remove areas with weaker correlations.

the frame annotated with bounding boxes are considered. The evaluation results on the two models and two datasets are shown in Tab. 3. It can be seen that in all cases, STEP is able to get the best performance and achieve obvious improvement on EP-3D. A potential reason could be that the proposed loss function L_K smooths the perturbations in both spatial and temporal dimensions so that the regions unrelated with the specified label are removed.

Table 3. Quantitative evaluation of visual explanation results generated by different method based on the spatial pointing game (S-PT) metric (in percentage). Results on this metric are measured by percentage. Here we use V16L to represent VGG16LSTM for short.

Methods	EPIC-Object		UCF101-24	
	R(2+1)D	V16L	R(2+1)D	V16L
Grad-CAM [23]	7.1	6.1	47.5	35.5
SaliencyTubes [31]	6.6	/	41.4	/
EB-R [1]	/	6.9	/	46.5
Ours (EP-3D)	8.1	6.1	47.6	47.2
Ours (STEP)	**9.0**	**9.1**	**56.7**	**52.7**

4.4. Comparison of methods for smoothness

To further evaluate the effectiveness of our proposed loss function on enhancing the EP-3D method through constraining temporal smoothness, we designed the following baseline smoothness methods for comparison. The quanti-

tative results are available in Tab. 4.

Evenly allocating preservation in temporal is a straight-forward way, *i.e.*, assigning each frame the same preservation ratio. Thus under this setting, we only optimize the distribution of mask values on the 2D image space of each frame. Formally, if we set the constraint to the overall preservation ratio as v, then the energy function Eq. 5 and 6 for EP-3D could be transcribed as follow in the video case,

$$\boldsymbol{M}_v = \operatorname*{argmax}_{\boldsymbol{m}_t : ||\boldsymbol{m}_t||_1 = vHW, \forall t} \Phi_c(\boldsymbol{M} \otimes \boldsymbol{X}), \quad (10)$$

$$\boldsymbol{M}_v = \operatorname*{argmin}_{\boldsymbol{M}}\{\lambda \sum_{t=1}^{T}||\text{vecsort}(\boldsymbol{m}_t) - \boldsymbol{r}_v||^2 - \Phi_c(\boldsymbol{M} \otimes \boldsymbol{X})\}. \quad (11)$$

which means we constrain the preservation ratio for each frame's mask uniformly as the predefined v. Here $\boldsymbol{r}_v$ is a template vector consisting of vHW ones followed by $(1-v)HW$ zeros. We refer to the variant method of EP-3D as **EP-3D-Evenly** for short below. We see this method as one straight-forward method for temporal smoothness because the distribution of preservation ratios on frames are uniform.

It can be seen from Tab. 4 that EP-3D-Evenly have comparable performance as the pure EP-3D on R(2+1)D models. On VGG16LSTM models, EP-3D-Evenly achieves obvious improvements than the pure EP-3D, which is consistent with our analysis that CNN-RNN tends to give even

focuses on each frame.

Gaussian smoothing is a generic method for blurring. As shown in equation 12, we exploit a Gaussian kernel $k \in \mathbb{R}^{2\Delta t + 1}$ to smooth the value of $\bar{m}_{i,j,t}$ on the smaller seed masks $\bar{M}$ according to its neighbours in the temporal dimension and yield the smoothed value $\bar{m}'_{i,j,t}$.

$$\bar{m}'_{i,j,t} = Z^{-1} \sum_{t'=t-\Delta t}^{t+\Delta t} k_{t'-t} \bar{m}_{i,j,t'} \qquad (12)$$

Here Z normalizes the kernel to sum to one. The kernel k is a radial basis function with profile $k_u = \exp(-u^2/(0.6\sigma))$ and set $\sigma = \Delta t$ to ensure the kernel's sharpness. Since the smoothness operator is applied on the smaller seed masks $\bar{M}$ which will be up-sampled to the masks M before perturbing frames, the operation could be viewed as smoothing the masks M both spatially and temporally. We insert this smoothness operator to the pure EP-3D method and test its effectiveness by choosing three different values of Δt. As shown in Tab. 4, the smoothness operator does not improve the performance of EP-3D obviously compared with our proposed L_K. We think this is because the smoothness method based on the loss function L_K could give the masks more searching freedom of shape in the process of optimization.

Table 4. Effectiveness evaluation of the proposed loss function L_K. We quantitatively compare it with a baseline temporal smoothness method based on Gaussian kernels. Here Δt decides the size of the applied Gaussian kernel.

Methods	EPIC-Object		UCF101-24	
	R(2+1)D	V16L	R(2+1)D	V16L
EP-3D	8.1	6.1	47.6	47.2
EP-3D-Evenly	7.9	8.7	47.4	53.2
EP-3D ($\Delta t = 1$)	8.4	7.4	48.5	51.1
EP-3D ($\Delta t = 2$)	8.1	8.1	47.9	52.6
EP-3D ($\Delta t = 3$)	7.7	7.1	50.3	**54.3**
STEP	**9.0**	**9.1**	**56.7**	52.7

4.5. Temporal pointing game

In Tab. 5, we show the results for quantitatively evaluating the ability of different visual explanation methods to locate the key action segments in the temporal dimension. The ability is measured by the metric of the temporal pointing game (T-PT) on the EPIC-Action sub-dataset. Because the temporal resolution of the importance maps generated by Grad-CAM and Saliency Tubes on 3D-CNNs is very low, that is, only two different importance maps are generated for the input sequence with 16 frames, we will not test the two methods on R(2+1)D model. The method EP-3D-Evenly is also excluded because it gives each frame the

same importance. It can be seen that EB-R performs well on the VGG16LSTM model and the pure EP-3D does not show satisfactory performance on both networks. As we analyzed, the pure EP-3D tends to attribute the network's output on the head and tail parts of videos, which lowers its performance on temporal localization. This could be effectively alleviated after adding the Gaussian smoothness kernel as the results shown in the middle three lines, which also reflects the necessity of introducing temporal smoothness. STEP could also improve the performance of EP-3D by introducing the loss function L_K. The improvement is comparable to the best of that achieved by Gaussian kernel on the R(2+1)D model but is relatively lower than the best results on the VGG16LSTM model.

Table 5. Quantitative evaluation results on the EPIC-Action test set by the temporal pointing game (T-PT) metric (in percentage). Results on this metric are measured by percentage.

Methods	R(2+1)D	V16L
EB-R [1]	/	**57.0**
EP-3D	34.0	31.0
EP-3D ($\Delta t = 1$)	46.0	39.0
EP-3D ($\Delta t = 2$)	46.0	47.0
EP-3D ($\Delta t = 3$)	41.0	45.0
STEP	**47.0**	49.0

5. Conclusion

In this paper, we shed light on the task of visually explaining video understanding networks by the perturbation-based method. The proposed method is characterized by model-agnostic and thus could be applied to diverse structures of video understanding networks in the same way even without detailed architectural knowledge. We also introduced a novel loss function to smooth the perturbation results in both spatial and temporal dimensions in order for the spatiotemporal smoothness of the explanation results. We experiment on two typical kinds of video classification network 3D-CNNs & CNN-RNN and on two datasets EPIC-Kitchens & UCF101-24, and compare smoothness with naive Gaussian blur. Both qualitative and quantitative results show that our proposed method could achieve competitive performance and the effectiveness of our method is therefore verified.

References

[1] Sarah Adel Bargal, Andrea Zunino, Donghyun Kim, Jianming Zhang, Vittorio Murino, and Stan Sclaroff. Excitation backprop for rnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1440–1449, 2018.

[2] Christopher J Anders, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Understanding patch-based learning of video data by explaining predictions. In *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, pages 297–309. Springer, 2019.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2425–2433, 2015.

[4] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7), 2015.

[5] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert MÃžller. How to explain individual classification decisions. *Journal of Machine Learning Research*, 11(Jun):1803–1831, 2010.

[6] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6299–6308, 2017.

[7] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847, 2018.

[8] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In *Advances in Neural Information Processing Systems*, pages 6967–6976, 2017.

[9] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 720–736, 2018.

[10] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 2950–2958, 2019.

[11] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 3429–3437, 2017.

[12] Chuang Gan, Naiyan Wang, Yi Yang, Dit-Yan Yeung, and Alex G Hauptmann. Devnet: A deep event network for multimedia event detection and evidence recounting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2568–2577, 2015.

[13] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[14] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[15] Yifei Huang, Minjie Cai, Zhenqiang Li, and Yoichi Sato. Predicting gaze in egocentric video by learning task-dependent attention transition. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[16] Yifei Huang, Yusuke Sugano, and Yoichi Sato. Improving action segmentation via graph-based temporal reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[17] Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 706–715, 2017.

[18] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. Adversarial examples in the physical world. *arXiv preprint arXiv:1607.02533*, 2016.

[19] Anh Nguyen, Jason Yosinski, and Jeff Clune. Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 427–436, 2015.

[20] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. *Proceedings of the British Machine Vision Conference (BMVC)*, 2018.

[21] Zhongang Qi, Saeed Khorram, and Fuxin Li. Visualizing deep networks by optimizing with integrated gradients. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1–4, 2019.

[22] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144, 2016.

[23] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017.

[24] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 3145–3153, 2017.

[25] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

[26] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in neural information processing systems*, pages 568–576, 2014.

[27] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.

[28] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

[29] J Springenberg, Alexey Dosovitskiy, Thomas Brox, and M Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015.

[30] Alexandros Stergiou, Georgios Kapidis, Grigorios Kalliatakis, Christos Chrysoulas, Ronald Poppe, and Remco Veltkamp. Class feature pyramids for video explanation. *arXiv preprint arXiv:1909.08611*, 2019.

[31] Alexandros Stergiou, Georgios Kapidis, Grigorios Kalliatakis, Christos Chrysoulas, Remco Veltkamp, and Ronald Poppe. Saliency tubes: Visual explanations for spatio-temporal convolutions. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 1830–1834, 2019.

[32] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, pages 3319–3328, 2017.

[33] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[34] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6450–6459, 2018.

[35] Jorg Wagner, Jan Mathias Kohler, Tobias Gindele, Leon Hetzel, Jakob Thaddaus Wiedemer, and Sven Behnke. Interpretable and fine-grained visual explanations for convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9097–9107, 2019.

[36] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 20–36. Springer, 2016.

[37] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision (ECCV)*, pages 818–833, 2014.

[38] Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126(10):1084–1102, 2018.

[39] Bolei Zhou, Alex Andonian, Aude Oliva, and Antonio Torralba. Temporal relational reasoning in videos. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 803–818, 2018.

[40] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016.