SWAG-V: Explanations for Video using Superpixels Weighted by Average Gradients

Thomas Hartley
Cardiff University
hartleytw@cardiff.ac.uk

Kirill Sidorov
Cardiff University
sidorovk@cardiff.ac.uk

Christopher Willis
BAE Systems Applied Intelligence
chris.willis@baesystems.com

David Marshall
Cardiff University
marshallad@cardiff.ac.uk

Abstract

CNN architectures that take videos as an input are often overlooked when it comes to the development of explanation techniques. This is despite their use in often critical domains such as surveillance and healthcare. Explanation techniques developed for these networks must take into account the additional temporal domain if they are to be successful. In this paper we introduce SWAG-V, an extension of SWAG for use with networks that take video as an input. In addition we show how these explanations can be created in such a way that they are balanced between fine and coarse explanations. By creating superpixels that incorporate the frames of the input video we are able to create explanations that better locate regions of the input that are important to the networks prediction. We compare SWAG-V against a number of similar techniques using metrics such as insertion and deletion, and weak localisation. We compute these using Kinetics-400 with both the C3D and R(2+1)D network architectures and find that SWAG-V is able to outperform multiple techniques.

1. Introduction

In recent years a number of explanation techniques have been introduced that aim to offer explanations for the predictions made by Convolutional Neural Networks (CNNs). However, the majority of these methods are aimed at CNNs which take an image as an input, with little thought for those which use video as an input. Network architectures which are designed for a video input typically have substantially different designs, in particular, the inclusion of a temporal element to facilitate learning across the multiple frames of a video. Despite this lack of explanation techniques for video based CNNs, they are beginning to be used in such critical applications as security [23, 25] and healthcare [2, 9, 16, 31]. This introduces an extra impetus to develop accurate and interpretable explanation methods for these networks.

The explanation techniques that do exist typically use similar (or in some cases identical) techniques as explanations for image based networks with little alteration to adapt for the temporal domain [5, 24]. Often explanations can be grouped into 3 categories: activation-based, perturbation-based, and gradient-based. Activation-based methods use the activation produced by the final convolution which can result in explanations that are very coarse. The coarseness is due to importance scores being assigned to very large regions of an image in both temporal and spatial dimensions. An example of this can be seen in Figure 1. Here we show an activation based method (Saliency Tubes [24]) compared to our SWAG-V method. Note, the Saliency Tubes explanation is coarse and cannot locate the skier correctly.

Conversely, black box techniques which perturb the input space (such as LIME [18] or RISE [17]) can be effective as the temporal domain can also be perturbed. However, techniques which perturb the input space of an action recognition network have the potential to be vastly inefficient as the models often contain more parameters compared to image based networks, and the input volume is larger.

Finally, gradient based techniques are able to assign a score to every pixel in both the spatial and temporal dimensions. However, assigning a score for individual pixels often has the appearance of noise and has been deemed to be less interpretable than methods which score larger regions [13, 29, 30]. Recently techniques have been introduced that pool these individual pixel scores into more interpretable regions [10, 13]. Of these, Superpixels Weighted by Average Gradient (SWAG) [10], is able to produce explanations efficiently using only a single forward and backward pass. We believe that SWAG can serve as a basis to

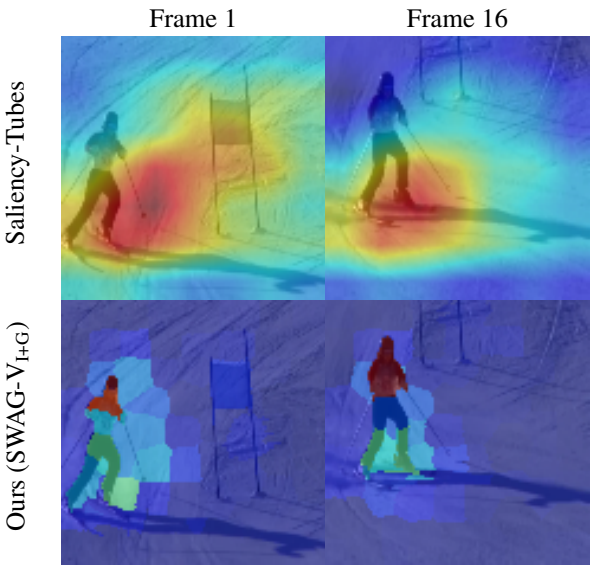

Figure 1. Example explanations for the ski slalom class using Saliency Tubes vs our proposed SWAG-V$_{I+G}$ method.

create accurate and interpretable explanations for networks that take videos as an input.

In this paper we introduce SWAG for Video (SWAG-V) by extending SWAG to produce explanations that incorporate the temporal domain. However, the original SWAG technique was optimised to perform well in deletion metrics, rather than both insertion and deletion metrics. To rectify this we will show how we can optimise the explanations created to perform well in both deletion and insertion metrics. We then conduct a number of experiments to show how SWAG-V performs compared to a range of similar methods.

2. Related Work

Whilst explanation methods designed specifically for networks which take a video as input are less common than their image based counterparts, a number have been proposed. Activation based techniques such as CAM [33], Grad-CAM [19], and Grad-CAM++ [5], rely on the activations from the final convolution layer to form the core of the visualisation. From here, it is simply a case of weighting the activations and rescaling back to the original input size. However, in action recognition networks that often use 3D convolutions, there is a temporal element to the activation maps that must also be considered. As the input passes through the network, not only is the spatial component reduced, so is the temporal component. This means that when the activation maps are resized back to the input size, the explanation created from the activation maps has to be stretched across frames, causing coarseness in the temporal dimension. A number of approaches have been taken to aligning the original 2D intent of these techniques with an additional temporal domain. In the simplest case, the CAM can be generated as it would in a 2D network. This produces a CAM with the same dimensions as the final activation layer, for example in C3D [26] this is a $14\times14\times2$ (height$\times$width$\times$depth), while in I3D [4], R(2+1)D [27] or ResNet3D [27], this is $7\times7\times2$. A number of techniques have attempted to build on top of this simple method. In the Saliency Tube work by Stergiou *et al.* [24], this is modified so that the activation maps themselves from the final convolution layer, following batch normalisation, are used to weight the activation maps rather than the gradients. In Grad-CAM++ [5], the authors discuss the application of their technique for use in action recognition networks. They propose a technique similar to the input$\odot$gradient technique proposed by Shrikumar *et al.* [20] whereby the resized CAM is combined with the spatio-temporal volume via point-wise multiplication. None of the above techniques, to our knowledge, have been subject to an empirically sound analysis using techniques that measure the accuracy of the generated heatmaps. Without first performing these experiments it is difficult to compare them against each other.

Extremal Perturbation 3D (EP-3D), a recently introduced technique by Li *et al.* [15] takes the methods proposed by Fong and Vedaldi [7, 8] and reconfigures it to work with action recognition networks. This is done through the introduction of a loss function that helps create masks that are smooth in both the spatial and temporal dimensions. These perturbation methods require multiple passes through the network (the authors suggest $2{,}000$ per image) to generate a single explanation, therefore these quickly become inefficient when working with spatio-temporal volumes.

Gradient based methods, which create explanations by backpropagation, still perform admirably for networks that use a spatio-temporal volume. The gradients are backpropagated back to the original input space which means that the inherent problem faced by activation based methods is avoided as their is no spatial or temporal resizing required. However, it is still a more difficult task to understand saliency maps based on individual pixel scores compared to more coherent heatmaps such as CAMs or those produced using perturbation techniques [13, 29, 30]. This is exasperated as often there is no cohesion between frames leading to explanations produced this way to appear like noise. Hiley *et al.* [11, 12] have subsequently proposed a modification to these gradient-based techniques to make them more suitable to action recognition networks.

Recently, a set of techniques have been introduced that aim to pool the individual pixel scores given by gradient-based techniques into more interpretable regions. These are XRAI [13] and SWAG [10]. Both have a similar approach in that they generate superpixels using the input image, and then assign a score to these regions based on the gradient values that lie within them. Of the two, XRAI is computational inefficient, requiring multiple sets of superpixels to be

created whereas a SWAG explanation can be computed in a single forward and backward pass. In addition, the original SWAG technique incorporates the gradient scores into the superpixel creation process. This allows superpixels to be created which better align to the regions deemed important to the networks prediction.

3. SWAG for Video: SWAG-V

We begin with a brief recap of SWAG and the two techniques that were introduced in the paper. The first was that an explanation for a networks prediction E_i could be created by weighting superpixel regions R_i (where R_i is the set of pixels belonging to the i^{th} superpixel) with the mean values M of the gradients found within that superpixel:

$$E_i = \frac{1}{|R_i|} \sum M \cap R_i. \tag{1}$$

The second method introduced was a way of incorporating gradients into the superpixel creation process. SWAG uses Simple Linear Iterative Clustering (SLIC)[1] as its method of generating superpixels. SLIC generates an initial grid, and then forms the superpixels based on the pixels distance (D') to the cell. The distance is given as:

$$D' = \sqrt{\left(\frac{d_c}{w_c}\right)^2 + \left(\frac{d_s}{w_s}\right)^2} \tag{2}$$

where d_c is the colour distance, and d_s is the spatial distance. To ensure these values are scaled correctly a scaling component is used for each, given by w_c and w_s respectively. SWAG modifies this to also include the distance to the gradient values as well:

$$D' = \sqrt{\left(\frac{d_c}{w_c}\right)^2 + \left(\frac{d_s}{w_s}\right)^2 + \left(\frac{d_g}{w_g}\right)^2} \tag{3}$$

where d_g is the gradient distance to a pixel and w_g is the associated scaling value. Superpixels can then created that use only the image (I), the combined image and gradient ($I + G$), or the gradient by itself (G). As with SWAG, we will show how superpixels can be generated for video inputs using both the image and the gradients individually as well as combining them. We will refer to these as SWAG-$V_{I,\ G}$, and $_{I+G}$ respectively.

We propose that SWAG is extremely well suited for use in networks that take a video as an input. These networks are often used for action recognition, and in this paper we will evaluate explanation methods on this task. However, a number of alterations must be made to the original SWAG implementation in order to allow it to produce appropriate explanations for networks using video as an input. The first is with regards to how superpixels are used. Whilst SWAG creates a single set of superpixels per images, we generate 3D superpixels that posses both a spatial and temporal element. This allows us to use Equation 1 to weight a superpixel through time as opposed to only representing a spatial area. We suggest that this will allow SWAG-V explanations to better follow motion throughout the video. We must then ascertain how many superpixels to generate, as the sizes of inputs used for action recognition networks are typically smaller than those used in image-based networks. SWAG uses 300 superpixels per image, so it may be that this is simply to many to produce a useful explanation. Indeed, the original SWAG technique optimised the number of superpixels to perform well at a metric called deletion, with no regard to the corresponding insertion metric. In the following section we propose a method for finding the optimal number of superpixels that allow us to balance these metrics and produce an explanation that is neither too coarse, or too fine.

4. Optimisation

While SWAG-V is a natural extension of SWAG, a number of changes must be considered to adapt it to use for action recognition networks. The primary considerations are the attribution method (i.e. which gradient method to use), the number of superpixels to create, and how best to combine superpixels and gradients to create SWAG-V $_{I+G}$. In this section we present the rationale behind the choices for the above considerations. Throughout this section we show results using Kinetics 400 [14] with the R(2+1)D [27] architecture. We start with an initial value of 100 superpixels, and then update this when we determine the optimal value.

To determine which choices lead to a good explanation, we use the local accuracy metric consisting of insertion and deletions scores as introduced by Petsiuk *et al.* [17]. We outline this further in Section 5.2. Briefly, it is a measure of how well an explanation scores regions of an image. For the deletion score, the pixels are ranked according to their importance to the prediction and then iteratively removed and the models prediction score observed. The area under the curve (AUC) is then used as the score. A low AUC score is desirable, suggesting the most important pixels were able to be located and removed first. Conversely, for insertion, the image pixels are again scored and ranked, but starting from a blank image the most important pixels are added back in. Again, we observe how the models predictions changes. Here a high AUC score is desirable.

4.1. Attribution Method

The attribution method is key to the success of SWAG-V. Determining which gradients produce the best initial results is important to discover early on, before further decisions are made. In this section we show results for two baselines (random noise and Sobel edges), and 3 methods for gener-

Table 1. Determining the optimal method of attribution. Results are shown for both insertion and deletion local accuracy metrics. For deletion, lower is better. For insertion, higher is better.

Method	Deletion R(2+1)D	Insertion R(2+1)D
Random Noise	0.194	0.192
Sobel	0.162	0.298
Vanilla Backprop	0.188	0.302
Guided Backprop	**0.113**	**0.371**
Input⊙Gradients	0.156	0.185

ating gradients (vanilla backpropagation [21], guided backpropagation [22], and Input⊙Gradients [20]). These results are shown in Table 1. From these results we observe that the baselines perform poorly for both deletion and insertion metrics. Vanilla backpropagation performs almost as poorly as random noise for the deletion metric, but performs much better for the insertion metric. With Input⊙Gradients this is reversed with the deletion metric performing much better than the insertion metric, which performs worse than random noise. Of all the techniques tested though, it is clear to see that, as with SWAG for 2D images, guided backpropagation performs the best. Easily outperforming all the other techniques in both deletion and insertion results. We will use guided backpropagation going forward.

4.2. Choice of Weights for SWAG-V_{I+G}

It is not initially obvious that when creating superpixels using a combination of the image and gradients, what is the best way to combine them. In this section we determine the optimal way of combining the images and gradients in the superpixel creation process. To do this we modify the weights for each input. The weights are labelled w_c and w_g for the image and gradients respectively. Each weight modifies how important the input is in the generation of the superpixel. A low weight score means a high importance, while a high weight score gives lower importance.

For both deletion and insertion metrics, we begin by performing a coarse grid search using w_c and w_g values of $[5, 10, 20]$. Here, 10 is the default value used by SLIC, while 5 and 20 represent a doubling and halving of an inputs influence respectively. To ensure that some aspect of the image is always taken into account, we only increase the image weight to a maximum of 20. From these initial results, we observed that both deletion and insertion results tend to favour a high w_c value and a low w_g value. This suggests that both metrics find the addition of the gradients to be beneficial to creating superpixels that better align with the explainable regions.

Based on this, we narrow in on the best performing values and subsequently produce a finer grid search using the combined scores of insertion and deletion to find the optimal value. We combined the scores as so: (deletion + (1−insertion)). Subtracting the insertion score from 1 means that as both scores approach 0, the better they are. We compute the combined scores for $w_c = [18, 20]$ and $w_g = [8, 9, 10, 11, 12]$. These results are shown in Figure 2. From these results we see that the lowest score occurs at $w_g = 9$ and $w_c = 20$. We will use these values for SWAG-V_{I+G} going forward.

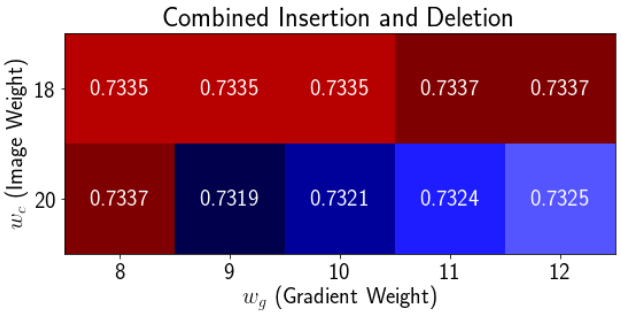

Figure 2. Fine search for optimal values for w_c and w_g. Here we show the combined insertion and deletion score. Lower (dark blue) is better.

4.3. Initial Superpixel Count

With SWAG, the authors determined that an initial count of 300 was appropriate for creating explanations for image based networks. These typically take a 224×224 image as an input. However, action recognition networks typically take an input with a spatial dimension of 112×112. Simply using the same number of superpixels as SWAG may result in explanations that are therefore too fine grained. In this section we explore the optimal number of superpixels to generate. We again use both the insertion and deletion metrics. We sweep through a range of superpixel counts and store the results. To find the optimal value, we add the deletion scores to 1−insertion scores. In this way both sets of scores are better as they approach 0. The results from the sweep of superpixels counts, and this combined result is shown in Figure 3. From the deletion and insertion results we see that they are the opposite of each other. Deletion (where a low score is better) improves the more superpixels there are in the image. The insertion metric (where a high score is better) performs much better with a very low number of superpixels and degrades as more are added. Again, looking at the combined values (deletion + (1−insertion)), where a low value is better, we see that after an initial rapid improvement in performance, the scores plateau. When we analyse this we found that the lowest score occurs at a superpixel count of 120. This is marked with the red cross in the figure. Going forward we will use a superpixel count of 120. This technique could be used for other explanation

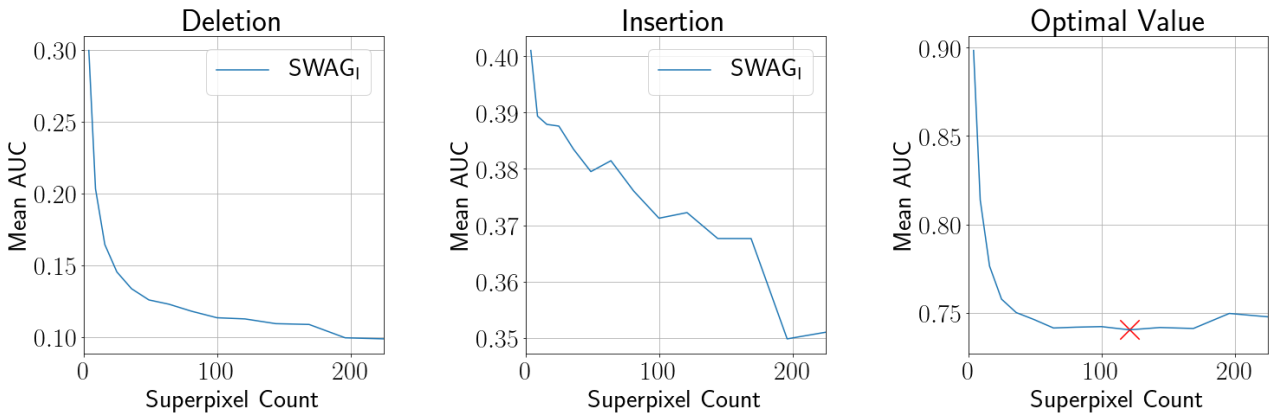

Figure 3. Top: Showing how the superpixel count affects the insertion and deletion scores. Bottom: Showing the optimal superpixel size.

techniques that offer control over the size of the explanation regions. For example both LIME and RISE could use this to optimise the number of superpixels or grid size respectively.

4.4. Final Parameter Choices

For clarity we present the final SWAG-V parameters:

- **Attribution Method**: We chose *guided backpropagation* as our attribution method. As with SWAG, we found it to outperform the other methods tests using the local accuracy metric.
- **Initial Superpixel Count**: Using the combined deletion and insertion results, we found that *120* was the optimal superpixel count to start the explanation with.
- **SWAG-V$_{\text{I+G}}$ weights**: We showed, using the combined deletions and insertion results, that $w_g = 9$ and $w_c = 20$ were the optimal values for use with SWAG-V$_{\text{I+G}}$.

5. Experiments

In this section, we present our results based on the application of SWAG-V to action recognition networks. We find our technique well suited to explaining networks that use spatio-temporal volumes. Here, we show this both quantitatively, through the use of experiments that measure saliency map accuracy, and qualitatively.

To generate local accuracy results, we use two architectures (C3D [26] and R (2+1)D [27]) and the Kinetics 400 dataset. Weights for the models were imported from those released by Tran *et al.* as part of the R(2+1)D paper [27]. We perform experiments on the spatial stream only. We do not perform any experiments on motion streams. From each of the Kinetics 400 validation videos, we extract the centre 16 frames and use these to obtain our results. This gives us 18,362 clips.

We compare SWAG-V against a number of explanation methods, namely Grad-CAM, Grad-CAM++, Saliency Tubes and EP-3D. Saliency Tubes requires a network architecture where there is only one linear layer in order to generate the weights for the activations. As such, we do not produce results for C3D using Saliency Tubes. In addition, we also compare against both guided backpropagation and the original implementation of SWAG$_{\text{I}}$. As SWAG $_{\text{I}}$ is designed for single images, we produce an explanation for each frame separately.

Finally we introduce two baselines based on the Euclidean distance from a specific pixel in each frame. We use both a centre point Euclidean distance map (referred to as centre), and the Euclidean distance to a uniformly randomly chosen pixel (referred to as random). For each clip, the random pixel is only assigned once meaning the same random explanation is produced for each frame in a 16 frame clip.

5.1. Qualitative Inspection of Results

We begin by presenting example explanations created using Kinetics 400 and R(2+1)D. Examples for the classes capoeira (a Brazilian martial art) and sheep shearing are shown in Figure 4. In the figure we show select frames from a 16-frame clip. We show our results alongside the initial input frames as well as comparable methods. From these examples we can see how much more precise all variants of SWAG-V are when compared to the activation based methods and EP-3D. For example, note that the activation map based methods are unable to precisely follow the motion of the person in the capoeira action class. Instead the explanations produced by the activation based methods seem to highlight the centre of the frame and highlight all of the action in that region. EP-3D [15], despite being trained for 2,000 epochs, produces coarse explanations which also lack definition. By this we mean that large areas of the image are assigned the highest importance value (shown by the large

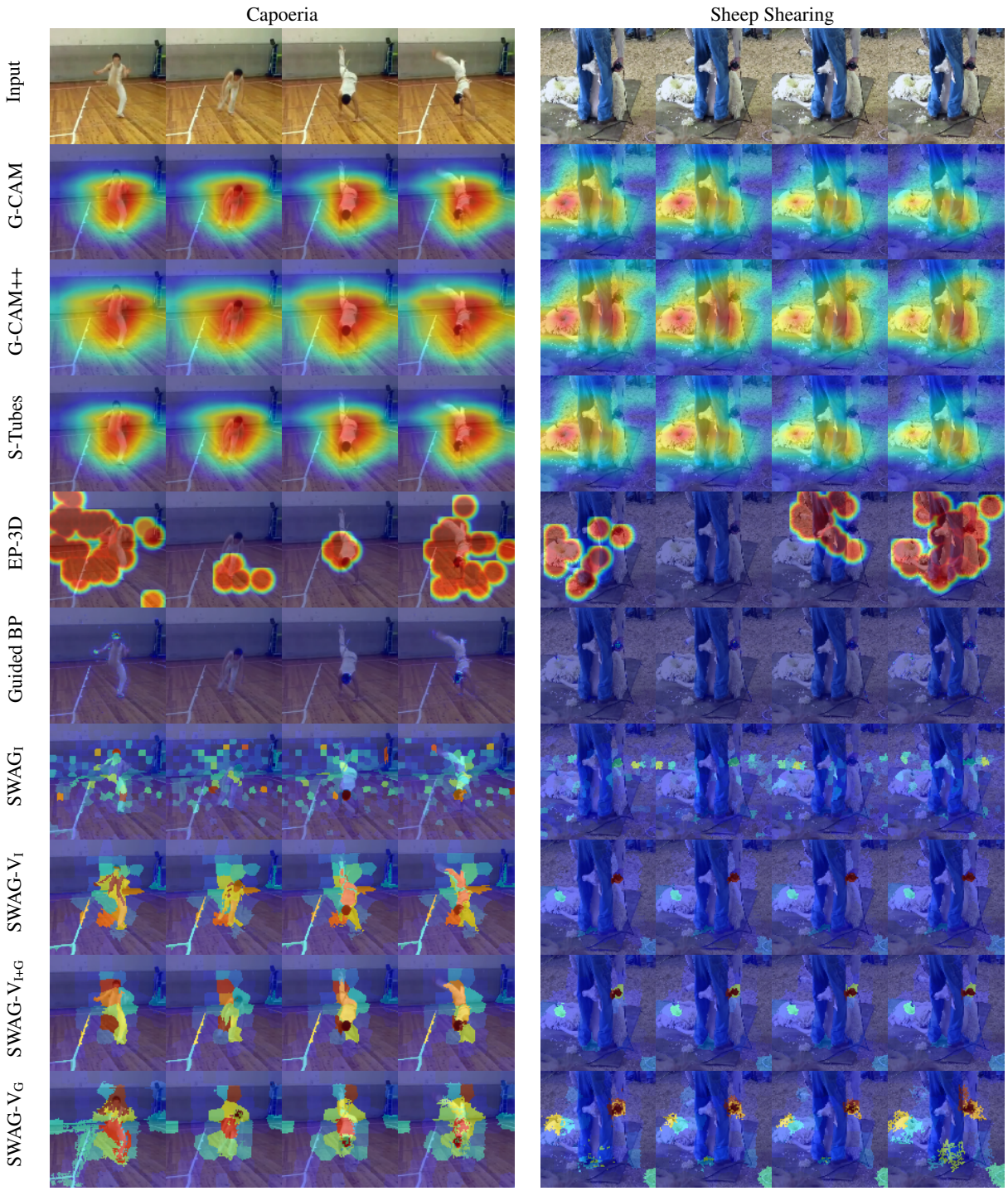

Figure 4. An explanation for the Kinetics 400 classes of capoeira and sheep shearing using R(2+1)D. We show frames 1, 6, 11 and 16 from a 16 frame clip. SWAG-V finds a middle ground between fine and coarse explanations. Best viewed on a PDF reader with zoom capability.

areas of red). In contrast to this, the original SWAG$_I$ and guided backprop methods produce fine explanations that are overly detailed, making them difficult to interpret.

In comparison, we see that our proposed methods are much able to better track smaller regions within both examples. For example, note how the movement in the persons legs are tracked for the capoeria class, or the precision with which the shears are highlighted in the sheep shearing class. Interestingly, SWAG-V$_G$ appears to be much noisier, with superpixels mixing into each other, compared to the

other SWAG-V methods. SWAG-V_I and SWAG-V_{I+G} seem to have smoother superpixels that follow the motion and object boundaries more accurately. Additional examples can be found in the supplemental material.

5.2. Local Accuracy

Now that we have qualitatively shown that SWAG-V produces explanations that are better able to highlight an action both spatially and temporally, we now perform quantitative experiments. We begin by exploring the accuracy of the explanation. For this we use the deletion and insertion metrics introduced by Petsiuk *et al.* [17]. Both deletion and insertion metrics measure how well an explanation aligns to the regions of the image used by the network to inform the predicted class. Deletion measures how well an explanation can locate the pixels that are very important to the network. By removing these pixels, in order from most important to least, we would expect to see that the more accurately an explanation can locate the important pixels, the faster the network is unable to predict the class. The insertion metric works slightly differently by reversing the order in which the pixels are altered. Starting from a blank image (all pixels set to 0), the pixels are iteratively reintroduced most important first. This insertion metric is similar to a concept introduced by Dabkowski and Gal [6], called the smallest sufficient region (SSR). They used this to help build explanations that were compact and cohesive. In the qualitative examples from Dabkowski and Gal, the SSR is cohesive, rather than a scatter of pixels the network finds important.

This then sets up a conflict between the insertion metric, and the deletion metric. While the deletion metric favours the precision of an explanation to accurately locate individual pixels that are important, the insertion metric rewards finding cohesive regions that are important. It therefore becomes a balance between how well a method performs at the insertion vs deletion metric. It is likely that as a technique becomes more precise it will give better deletion results, while a technique that becomes more cohesive should give better insertion scores.

To generate our insertion and deletion scores we first create an explanation for an input using one of our chosen methods. Using these explanations, we rank every pixel based on its importance to the network. For deletion, we begin with the video and remove pixels from most important to least. For insertion, we begin with a video containing only zeros and reintroduce the pixels from the original input, most important first. We introduce or remove pixels over 28 iterations. This equates to $7{,}168$ pixels being introduced or removed at a time. While this is more pixels than introduced or deleted for the original image experiments ($1{,}792$ pixels at each iteration), it is necessary to ensure that the experiments are able to be run efficiently.

We show the results for the insertion and deletion metrics in Table 2. From these results we can begin to discern a number of important points. The first is that a number of methods are beaten by the baselines. In particular, whilst guided backpropagation, EP-3D and the original $SWAG_I$ perform well in the deletion metric, none are able to outperform the baselines in the insertion metric. In contrast to this, the activation based methods perform the best at the insertion metric and poorly at the deletion metric. Indeed, when using the C3D network, the centre baseline outperforms Grad-CAM. This suggests that the methods based on activation maps sacrifice precision for a more cohesive explanation, whilst the methods that did well in the deletion metric do so by sacrificing cohesion.

Table 2. Deletion and insertion scores for C3D and R(2+1)D

	Deletion		Insertion	
Method	C3D	R(2+1)D	C3D	R(2+1)D
Centre	0.153	0.167	0.264	0.304
Random	0.222	0.265	0.272	0.322
Guided Bp	**0.031**	**0.035**	0.184	0.287
G-CAM	0.157	0.118	0.313	**0.447**
G-CAM++	0.136	0.125	**0.315**	0.422
S-Tubes	–	0.118	–	**0.447**
EP-3D	0.082	0.074	0.233	0.266
$SWAG_I$	0.068	0.069	0.190	0.222
SWAG-V_I	0.091	0.113	0.312	0.372
SWAG-V_{I+G}	0.087	0.111	0.314	0.381
SWAG-V_G	0.068	0.080	0.262	0.343

When we look at the results for our proposed methods, we see that SWAG-V_G performs the best in the deletion metric. However, SWAG-V_G suffers in the insertion metric, being unable to outperform the insertion baseline for C3D. Both SWAG-V_I and SWAG-V_{I+G} outperform the baselines. Alongside Grad-CAM++ and Saliency-Tubes, these are the only techniques to do so for both models tested. Of the two, SWAG-V_{I+G} outperforms SWAG-V_I is the majority of experiments, and outperforms the activation based methods in the deletion metric. For the insertion metric, SWAG-V_{I+G} is only marginally outperformed by the activation based methods. Of all methods, SWAG-V $_{I+G}$ seems to be the optimal method, balancing both insertion and deletion metrics well.

5.3. Weak-Localisation

It has become common when introducing a novel interpretability technique to have a section of experiments that discuss how well the generated saliency map locates the given object within an image. In networks that deal with image classification this primarily takes the form of extracting some portion of an explanation and seeing how it aligns spatially with a bounding box. However, for video inputs we need to localise in both the spatial and temporal dimensions. To do this, we extend the weak-localisation method established in [3, 8, 32]) for use with video inputs. We

Table 3. Localisation results as error %, where a low score is more desirable. For the 3 thresholds tested, we present the mean score.

	C3D			R(2+1)D		
	Val	Mea	Ene	Val	Mea	Ene
Centre	87.91	88.26	87.66	87.91	88.26	87.66
Random	90.51	90.63	90.34	90.51	90.63	90.40
Guided Bp	90.62	90.69	100.00	90.68	87.58	85.64
G-CAM	90.57	90.46	90.53	85.40	87.08	85.32
G-CAM++	90.21	89.91	89.92	85.40	87.08	85.32
S-Tube	–	–	–	85.58	85.86	85.64
EP-3D	90.70	90.70	88.34	90.70	90.70	87.53
$SWAG_I$	90.00	89.84	89.81	90.33	90.14	89.96
$SWAG\text{-}V_I$	74.00	74.46	74.26	71.76	72.61	72.58
$SWAG\text{-}V_{I+G}$	**73.00**	**72.52**	**72.99**	70.02	**69.94**	71.25
$SWAG\text{-}V_G$	75.22	74.25	74.09	**69.88**	70.80	**70.59**

threshold each explanation in 3 ways as: using the pixel value scaled between 0 and 1 (Val), thresholding based on the mean value (Mean), and thresholding based on the energy (Ene). A comprehensive explanation of these thresholds can be found in the work by Fong and Vedaldi [8].

To begin to adapt this experiment for use with spatio-temporal inputs, a suitable dataset with corresponding bounding boxes is required. For this experiment, we chose UCF101. This is both a commonly available dataset, and crucially contains localisation annotations for 24 of the classes. It is these classes that we will use to measure the localisation abilities of the interpret ability techniques. This is also in-line with techniques for specifically generating action localisations in spatio-temporal inputs (i.e. bounding boxes through time) [28]. For simplicity, we filter out any videos from the validation set (containing 914 videos) that have more than one action to localise per frame, and any clips that have fewer than 16 contiguous bounding boxes present. This reduces the validation set to 697 videos. We crop videos into 16 frame clips, starting a new clip every 8 frames. If the 16 frame clip does not contain a ground truth bounding box for every frame, we ignore it. This generates a total of 10,704 clips. To evaluate, we generate a bounding box for each frame in the clip, and get an IOU score for each frame. We average these over the 16 frame clip. If the average IOU is greater than 0.5 we classify it as correct.

We show the results for this experiment in Table 3. The first striking result is that Guided-Backpropagation, EP-3D and the original $SWAG_I$ struggle to offer any improvements over the centre baseline (i.e. simply pointing at the centre of each frame better locates the action). The second is that despite having a cohesive explanation that scored well in the insertion metric, the activation-based methods also perform poorly. Indeed when creating explanations for C3D they are unable to outperform the centre baseline. This is likely due to the temporal stretching that occurs when the explanation is resized, making it difficult to follow movement.

Given these results, our SWAG-V methods outperform all others tested by a large margin, often by around 15% to 20%. Of the 3 SWAG-V methods, $SWAG\text{-}V_I$ performs the worst, with $SWAG\text{-}V_{I+G}$ performing best with C3D, and $SWAG\text{-}V_G$ the best with R(2+1)D. Compared to the difference with the other methods though, there is little difference between the SWAG-V methods, with around 2% variation.

Table 4. Mean computation time in seconds

Method	C3D	R(2+1)D
Grad-CAM	**0.08**	0.15
G-CAM++	**0.08**	0.15
Saliency Tubes	–	**0.05**
EP-3D	75.45	123.89
$SWAG\text{-}V_I$	0.41	0.68
$SWAG\text{-}V_{I+G}$	0.46	0.74
$SWAG\text{-}V_G$	0.26	0.40

5.4. Efficiency

We show the mean time taken to generate an explanation in Table 4. We can see that SWAG-V adds an additional computational overhead when compared to the Guided-Backpropagation and the activation-based methods. However, compared to EP-3D, SWAG-V is much more efficient, with EP-3D requiring over 2 minutes to create a single explanation for R(2+1)D. This is due to the multiple passes required to compute a single EP-3D explanation when compared to the single pass required by SWAG-V.

$SWAG_G$ is the fastest of the SWAG-V methods due to the simplicity of making the superpixels. $SWAG_{I+G}$ is the slowest due to the overhead involved in combining the gradients and images. Despite the additional computational cost, we believe the improvement in the other metrics justifies the increase.

6. Conclusion

In this paper, we have proposed SWAG-V, an extension of SWAG that allows for the creation of explanations for action recognition networks. SWAG-V posses a number of alterations that make it more suitable for use in these networks than the original SWAG. This includes the use of a technique for finding the optimal number of superpixels to allow a trade-off between the creation of fine and coarse explanations. We have shown through experimentation that SWAG-V is the only method used to consistently beat all the baselines. With the original SWAG and EP-3D failing to outperform the baseline for insertion and weak-localisation, and activation based methods failing to outperform the baseline consistently for weak-localisation. Of the 3 proposed SWAG-V techniques, we suggest $SWAG\text{-}V_{I+G}$ is the optimal variant. We showed qualitatively that it produced cleaner explanations that $SWAG\text{-}V_G$ whilst outperforming $SWAG\text{-}V_I$ in the local accuracy and weak-localisation metrics.

References

[1] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282, 2012.

[2] Md Atiqur Rahman Ahad, Anindya Das Antar, and Omar Shahid. Vision-based action understanding for assistive healthcare: A short review. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[3] Chunshui Cao, Xianming Liu, Yi Yang, Yinan Yu, Jiang Wang, Zilei Wang, Yongzhen Huang, Liang Wang, Chang Huang, Wei Xu, Deva Ramanan, and Thomas S. Huang. Look and think twice: Capturing top-down visual attention with feedback convolutional neural networks. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 2956–2964, December 2015.

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

[5] Aditya Chattopadhyay, 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*, pages 839–847, 2018.

[6] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 69706979, Red Hook, NY, USA, 2017. Curran Associates Inc.

[7] Ruth Fong, Mandela Patrick, and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[8] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 3449–3457, Oct 2017.

[9] Yongbin Gao, Xuehao Xiang, Naixue Xiong, Bo Huang, Hyo Jong Lee, Rad Alrifai, Xiaoyan Jiang, and Zhijun Fang. Human action monitoring for healthcare based on deep learning. *IEEE Access*, 6:52277–52285, 2018.

[10] Thomas Hartley, Kirill Sidorov, Christopher Willis, and David Marshall. SWAG: Superpixels weighted by average gradients for explanations of CNNs. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 423–432, January 2021.

[11] Liam Hiley, Alun Preece, Yulia Hicks, Supriyo Chakraborty, Prudhvi Gurram, and Richard Tomsett. Explaining motion relevance for activity recognition in video deep learning models. *arXiv preprint arXiv:2003.14285*, 2020.

[12] Liam Hiley, Alun Preece, Yulia Hicks, David Marshall, and Harrison Taylor. Discriminating spatial and temporal relevance in deep Taylor decompositions for explainable activity recognition. *arXiv preprint arXiv:1908.01536*, 2019.

[13] Andrei Kapishnikov, Tolga Bolukbasi, Fernanda Viegas, and Michael Terry. XRAI: Better attributions through regions. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019.

[14] Will Kay, Joao Carreira, Karen Simonyan, Brian Zhang, Chloe Hillier, Sudheendra Vijayanarasimhan, Fabio Viola, Tim Green, Trevor Back, Paul Natsev, et al. The Kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*, 2017.

[15] Zhenqiang Li, Weimin Wang, Zuoyue Li, Yifei Huang, and Yoichi Sato. Towards visually explaining video understanding networks with perturbation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 1120–1129, January 2021.

[16] Na Lu, Yidan Wu, Li Feng, and Jinbo Song. Deep learning for fall detection: Three-dimensional cnn combined with lstm on video kinematic data. *IEEE Journal of Biomedical and Health Informatics*, 23(1):314–323, 2019.

[17] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: randomized input sampling for explanation of black-box models. In *British Machine Vision Conference 2018, BMVC*, 2018.

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

[19] 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 *The IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, Oct 2017.

[20] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pages 3145–3153. PMLR, 06–11 Aug 2017.

[21] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *2nd International Conference on Learning Representations, ICLR 2014, Workshop Track Proceedings*, 2014.

[22] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. In *3rd International Conference on Learning Representations, ICLR 2015, Workshop Track Proceedings*, 2015.

[23] G Sreenu and MA Saleem Durai. Intelligent video surveillance: a review through deep learning techniques for crowd analysis. *Journal of Big Data*, 6(1):1–27, 2019.

[24] Alexandros Stergiou, Georgios Kapidis, Grigorios Kalliatakis, Christos Chrysoulas, Remco Veltkamp, and Ronald Poppe. Saliency tubes: Visual explanations for spatio-temporal convolutions. *arXiv preprint arXiv:1902.01078*, 2019.

[25] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern*

Recognition (CVPR), June 2018.

[26] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3D convolutional networks. In *The IEEE International Conference on Computer Vision (ICCV)*, ICCV '15, pages 4489–4497, Washington, DC, USA, 2015. IEEE Computer Society.

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

[28] Jan C. van Gemert, Mihir Jain, Ella Gati, and Cees G. M. Snoek. Apt: Action localization proposals from dense trajectories. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 177.1–177.12. BMVA Press, September 2015.

[29] 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 *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9089–9099, June 2019.

[30] Mengjiao Yang and Been Kim. Benchmarking Attribution Methods with Relative Feature Importance. *CoRR*, abs/1907.09701, 2019.

[31] Jun Yin, Jun Han, Ruiqi Xie, Chenghao Wang, Xuyang Duan, Yitong Rong, Xiaoyang Zeng, and Jun Tao. Mc-lstm: Real-time 3d human action detection system for intelligent healthcare applications. *IEEE Transactions on Biomedical Circuits and Systems*, 15(2):259–269, 2021.

[32] 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.

[33] 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*, pages 2921–2929, 2016.