Jitter-CAM: Improving the Spatial Resolution of CAM-Based Explanations

Thomas Hartley[1]
hartleytw@cardiff.ac.uk
Kirill Sidorov[1]
sidorovk@cardiff.ac.uk
Christopher Willis[2]
chris.willis@baesystems.com
David Marshall[1]
marshallad@cardiff.ac.uk

[1] Cardiff University
[2] BAE Systems Applied Intelligence

Abstract

Class Activation Mappings (CAMs) are a popular group of methods for creating visual explanations of the reasons behind a network's prediction. These techniques create explanations by weighting and visualising the output of the final convolution layer. Recent CAM techniques have sought to improve these explanations by introducing methods that aim to produce weights that more accurately represent how the network informs its prediction. However, none of these methods address the low spatial resolution of the final convolutional layer, leading to coarse explanations. In this paper, we propose Jitter-CAM, a method for producing and combining multiple CAM explanations that allow us to create explanations with a higher spatial resolution than previous comparable methods. We use ImageNet and a number of well known architectures to show that our technique produces explanations that are both more accurate and better at localising the target object. Code for Jitter-CAM is available at https://github.com/HartleyTW/Jitter-CAM.

1 Introduction

The impact of Convolutional Neural Networks (CNNs) on all applications of computer vision in recent years has been profound. We now see that CNNs are becoming more common in domains where the predictions of the network can have critical real life consequences. Examples of this include healthcare [33] or security [34]. However, a common concern with the use of CNNs is their lack of interpretability. As a result, numerous techniques have been developed to create post-hoc explanations of model decisions.

A popular method for creating explanations uses weighted feature maps from the final activation layer. This is called Class Activation Mapping (CAM). Numerous explanation methods have used CAMs as the basis for their technique, with the primary difference being in how the feature maps are weighted. Despite all the advances in generating better methods of weighting the activations, a limiting factor of any explanation produced this way is the

spatial resolution of the activations. The spatial resolution of the final activation is a function of both the input image size and the architecture. The typical input size when using ImageNet is 224×224, with Inception [29] using an input of 299×299. It is therefore common for newer CNN architectures [10, 11, 12, 37], using the typical input size, to produce a final activation map with a height and width of 7. VGG16 [24] is somewhat unusual with a final activation map of 14×14 when using a 224×224 input. When these coarse activation maps are resized to the input image size using bilinear interpolation, their poor resolution will be reflected in the explanation. If there was a method for creating activation maps with a higher resolution than 7×7 then the resultant explanations would likely be more useful than before. An alternative to CAM-based methods are perturbation methods [6, 19, 21]. These typically alter the input in some way and observe the change in confidence of the network's prediction. As the input can be perturbed in multiple ways, even down to manipulation of individual pixels, there is theoretically no limit to the spatial resolution of the explanations. However, these methods are typically very inefficient as the image is required to be passed through the network multiple times to build a single explanation (e.g. 8,000 times with RISE and ResNet50). This limits the resolution of the explanation than can be realistically achieved, typically being only slightly larger than those obtained by CAM methods before the process becomes too inefficient. For example, the RISE authors propose explanations of size 8×8 (prior to resizing) compared to Grad-CAM's 7×7.

In this paper, we introduce *Jitter-CAM*, a method for creating explanations which have an increased spatial resolution compared to similar methods. This is achieved by creating multiple CAM explanations in a structured manner using a resized input image, and combining the results. Each CAM explanation locates features that are important to the networks discrimination of the given class. Combining them gives an insight into how the network represents the target class. We show that our method creates explanations with better accuracy, and better localisation abilities. We conduct experiments using ImageNet with ResNet, Inception, and DenseNet. In addition, we discuss faithfulness, a metric found in the CAM literature, and show it is a misleading metric that should be discontinued from future use.

2 Related Work

While we concentrate on comparisons with CAM-based methods in this paper, there are a number of alternative methods for generating explanations of a network. Gradient-based methods produce very fine explanations in which each pixel in an image is assigned a score. Examples of these techniques are 'vanilla' gradients [25], Guided Backpropagation [27], Layer-Wise Relevance Propagation (LRP) [1], Integrated Gradients [28], Smooth-Grad [26], and Excitation Backprop [36]. Recently, there have been techniques such as XRAI [14] and SWAG [9] that attempt to pool these gradients to produce more interpretable explanations.

Alternatively to this, there are a range of explanation techniques based on perturbation such as LIME [21], RISE [19] and meaningful perturbations [6]. Meaningful perturbations are created by learning a mask based on a novel loss function. This work is expanded upon by Qi *et al*. [20] through the use of Integrated Gradients, allowing explanations to be created more efficiently. Similar explanations that are learnt via optimising a loss function are SCOUTER [15] and the work by Schulz *et al*. [22]. The benefit of these techniques is that the spatial resolution of the explanation can be chosen at will. The caveat is that typically, an increased resolution will require an increased number of perturbations / training epochs.

CAM-based methods were first introduced by Zhou *et al*. [38]. This technique was ini-

tially confined to network architectures that possessed a global average pooling layer prior to the classification layers. Grad-CAM [23] generalised this by backpropagating the gradients from a specific class output to the activation map. Weights for the activations were produced by taking the mean value of the gradients in such a way that each activation map had a corresponding mean gradient value. The intuition here is that the mean of the gradient will be indicative as to how useful the network finds a particular activation map to the output. This concept was expanded further in Grad-CAM++ [3], which aims to improve the localisation ability of Grad-CAM, and XGrad-CAM [8], which aims to ensure that the explanation meets the axioms introduced by Montavon *et al*. [17] and Sundararajan *et al*. [28].

Recently, there has been a trend towards perturbation-based approaches. This is seen in the Ablation-CAM [4] and Score-CAM [30] techniques. In both, the network is perturbed and the output prediction score for the desired class used as the activation map's weight. With Ablation-CAM, the activation maps are extracted and iteratively 'turned off' by setting them to 0. When these activations are passed to the classification layers, it gives an indication of how useful it was to the network's prediction based on the amount the class score drops. Score-CAM is similar to RISE [19] in its approach. It extracts the activation maps and iteratively multiplies the input with each map so as to mask out regions not activated by that filter. These are then passed to the network and the prediction score used as the weight for that activation map. These methods require as many passes through the network as there are maps to weight; for VGG16 this is 512 passes. Recently, the number of filters in the final layer have increased e.g. ResNet[10] uses 2,048 filters, while ShuffleNet[37] and MobileNet[11] use 1,024 and 1,280 respectively. This subsequently increases the computational requirements needed to produce an explanation. In addition, the authors of Ablation-CAM[4], suggest that their technique does not outperform other methods when using networks that do not use fully connected layers. Fully connected layers are now uncommon in newer networks such as ResNet[10], DenseNet[12], ShuffleNet[37] and MobileNet[11].

3 Jitter-CAM

As we have highlighted, the resolution of the activation maps which CAM methods use as a base is a limiting factor. By offering a technique that allows us to increase the resolution of the CAMs, we believe that visualisations can be produced that provide more useful explanations. While other methods may achieve this through perturbations using a large number of iterations, we propose a method based on increasing the scale of the input, and then, producing explanations for patches of the new image corresponding to the original image size. These explanations are combined to produce a new explanation with an increased resolution.

Before outlining our method we adopt the following notation: Let the size of the activation map be defined as $m \times m$. Each cell of the activation map corresponds to k pixels in the input image, where $k = p/m$. Here, p is the height or width of the input image (i.e. 224 pixels). To increase the size of the CAM from $m = 7$ to $m = 10$, an increase, d, of 3, we would need to increase the size of the input image by dk pixels. We call this resized image I. A CAM is created by passing an image, X, to a CAM explanation function $E(X)$. $E(X)$ returns a CAM explanation of size $m \times m$. We use Grad-CAM as our base explanation method.

Example: suppose we would like to increase the CAM size, m, from 7 to 12, an increase value, d, of 5. For an input image of $p = 224$, this would be an increase of $5k$ where $k = 224/7$. This would give a new image size of 384×384. Patches of this enlarged image are taken corresponding to size $p \times p$, and a stride of k. For a CAM size increase of 5,

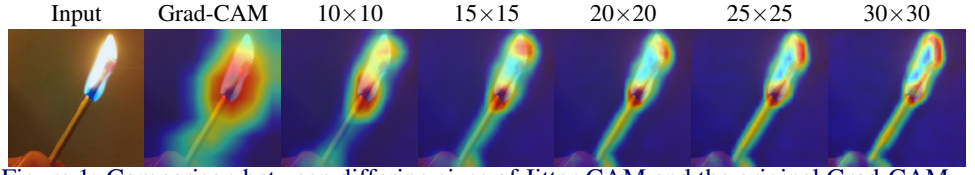

Figure 1: Comparison between differing sizes of Jitter-CAM and the original Grad-CAM.

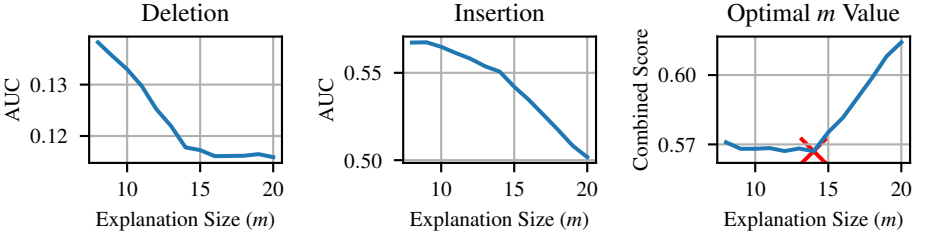

Figure 2: How the deletion and insertion scores are affected by the spatial value of m.

this would give 36 patches ($=(5+1)^2$). To create a Jitter-CAM explanation, we create and combine CAM explanations from each of the patches. We extract patches from the resized image I and creating a combined explanation J (of size $(m+d)\times(m+d)$) as so:

$$J_{i:i+m,j:j+m} = J_{i:i+m,j:j+m} + E(I_{ki:ki+p,kj:kj+p}), \tag{1}$$

where i,j is the starting location for the explanation to be added to J. Here $i,j=[1...d+1]$. Multiple explanations will be created at any given point, with more being created corresponding to centre regions. To account for this, we create a count of how many times each region of the image has had an explanation created for it. We call this C, a matrix of size $(m+d)\times(m+d)$, and define it as:

$$C_{i:i+m,j:j+m} = C_{i:i+m,j:j+m} + 1. \tag{2}$$

Again, $i,j=[1...d+1]$. We found in practice that there are few artefacts present from the resizing process, however, for transparency we show example patches and their associated pre-resized explanations in the supplemental material.

Finally, we produce the pre-resized Jitter-CAM explanation: Jitter-CAM $=\frac{J}{C}$. Given this higher resolution CAM, we can now resize it to the original input image size using bilinear interpolation to give us a familiar looking CAM explanation.

The key parameter for this technique is by how much we increase the size of the CAM? If we increase it too much, we run the risk of the explanations not being exposed to enough of the image and, therefore, unable to produce explanations for the object in the image. Too small and we may not offer any improvement over the standard Grad-CAM method. In Figure 1, we show an example of how the increase in the CAM size affects the final explanation. Here, we can see that the original Grad-CAM is fairly coarse. However, as we begin to increase the size of the CAM, we see that the explanation begins to better highlight the object for classification, rather than background regions.

How then do we find the optimal size for a Jitter-CAM explanation? To answer this we use the deletion and insertion metrics from RISE. We experiment using the ResNet50 network, and vary the size of the CAM from $m=7$ (the original Grad-CAM) to $m=20$ in steps of 1. We determine the optimal m value by combining the insertion and deletion scores together: (deletion + (1−insertion)).

The results for each of these experiments is shown in Figure 2. Here, we see the deletion score plateaus around $m = 14$ while the insertion score never improves from the original coarse size of $m = 7$. Taking the combined score m value at the lowest position determines that $m = 14$ seems to be the optimal size, double the initial size of the original CAM explanations from a ResNet50 model. For Inception [29], which has a final activation layer of 8×8, we experimented with increasing it to 14×14, and 16×16. We found better results were achieved using a 16×16 Jitter-CAM explanation. Going forward, when we refer to Jitter-CAM it will be with an explanation of double the original explanations size. This gives ResNet50 and DenseNet a 14×14 CAM prior to resizing, and Inception a 16×16 CAM.

4 Experiments

In this section, we present results for a number of common experiments found in the explainability literature. These are measures of explanation accuracy, localisation ability, and efficiency. All experiments are conducted using the ImageNet validation set (50,000 images) using the pre-trained models from PyTorch: ResNet50[10], DenseNet121[12], and Inception V3[29]. The first two networks produce CAM explanations of 7×7 prior to resizing, while the Inception network produces 8×8 CAMs. We primarily compare Jitter-CAM against other CAM methods due to the increased computational requirements of perturbation methods. However, we do include results for RISE to allow for some comparison. In addition, we test using two baselines, a heatmap radiating from the centre point, and a heatmap radiating from a random point. We label these as centre and random respectively.

4.1 Qualitative Inspection of Results

We begin with a qualitative assessment of the explanations. Examples of each of the methods tested are presented alongside Jitter-CAM examples. These are shown in Figure 3. Here, we see that using Jitter-CAM to double the size of the explanation prior to resizing, allows us to produce an explanation that is more compact around the object in the image. We see this clearly in the first image of a goldfish, where our Jitter-CAM explanation focuses less on the surrounding tank, and more on the goldfish itself. Also notice, in cases such as the canoe, how Jitter-CAM manages to highlight more of the object than previous methods. In examples where more regions of an object are highlighted, Jitter-CAM still adheres more strongly to the object boundary than previous CAM methods. Perhaps the most striking aspect of the qualitative comparison is how visually similar all the previous CAM methods are, due to the constraints of all using the same activation maps. RISE produces explanations that are finer than previous CAM methods, but still not as precise as Jitter-CAM. In addition each RISE explanation required 8,000 passes through the network. Additional examples for alternative architectures tested can be found in the supplemental material

4.2 Grad-CAM Vs 7×7 Jitter-CAM

Before we begin to discuss quantitative metrics, a question that should be asked is whether the explanation created by Jitter-CAM has the ability to recreate the original Grad-CAM explanation, if given a smaller starting image. We use an input image of size 160×160 (which produces a 5×5 Grad-CAM explanation) and use Jitter-CAM to resize it to 7×7. When compared to Grad-CAM explanations produced using an input image of size $224 \times$

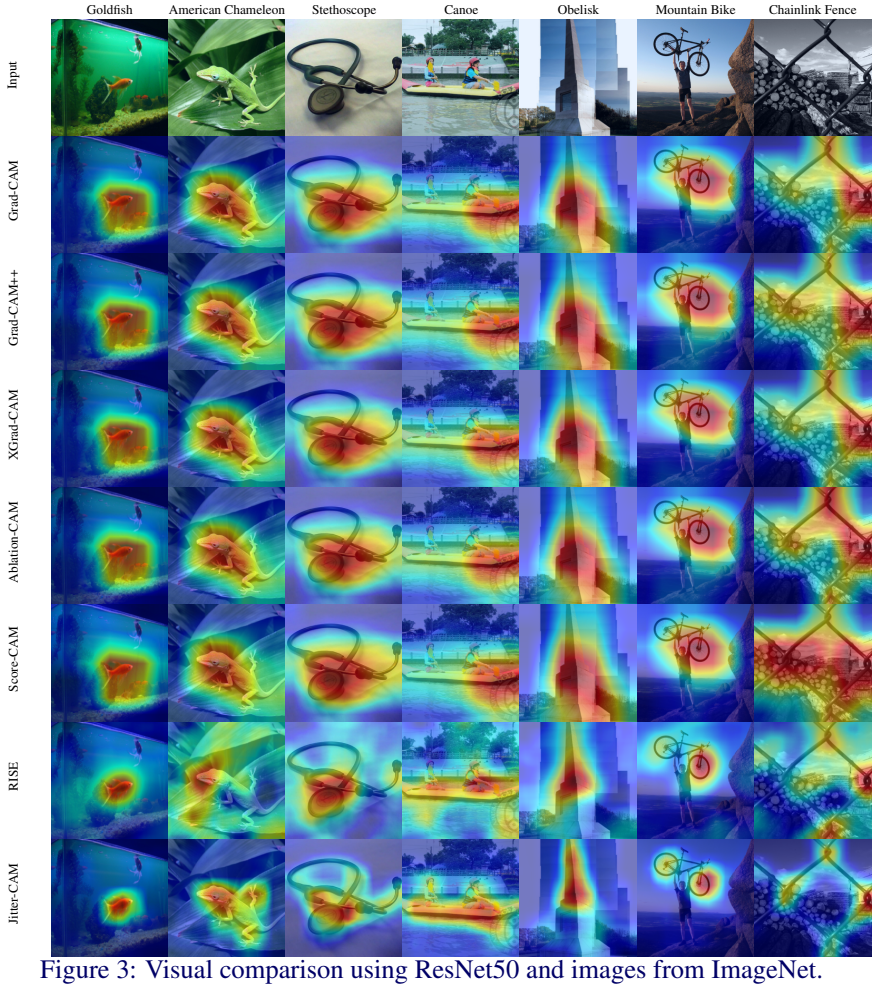

Figure 3: Visual comparison using ResNet50 and images from ImageNet.

224 we find that the mean Spearman correlation across the ImageNet validation set is 0.95 suggesting a high degree of similarity between the explanations. Visual examples of this can be found in the supplemental material.

4.3 Faithfulness

Faithfulness is a metric that is often deployed in CAM research[3, 4, 13]. In this section, we show that it is *not* an appropriate measure of an explanation, and perhaps should be discounted as a metric in future research. First, introduced by Chattopadhay *et al*. [3], the faithfulness metric was intended to be a measure of how well the regions that were deemed important by an explanation aligned to those used by the model. This is achieved by performing a point-wise multiplication of the explanation and the input image to create a masked image E^c for class c: $E^c = L^c \circ I$. Here, L^c is the original explanation for class c scaled between 0...1 and I is the input image. The masked image, E^c is passed to the model and the change in the softmax results observed. This softmax score is then used to inform two measures of faithfulness, namely the average drop (AD), and the increase in confidence (IIC).

Method	ResNet50		DenseNet		Inception	
	IIC	AD	IIC	AD	IIC	AD
Grad-CAM	40.21	14.04	39.65	13.22	40.46	16.98
Grad-CAM++	38.70	14.22	37.54	13.69	38.89	17.02
Constant	**48.22**	**2.70**	**49.14**	**1.08**	**47.36**	**1.32**

Table 1: Faithfulness scores for IIC (higher is better) and AD (lower is better).

AD is a measure of how much the models confidence drops when shown the masked image compared to the original image. The intention is that a good explanation should highlight the regions of an image important to a network, and give low scores to those that are unimportant. Therefore, E^c will be an image where useful regions are kept, and those not seen as useful are suppressed. AD is then measured as:

$$\frac{1}{N}\sum_{i=1}^{N}\frac{\max(0, Y_i^c - O_i^c)}{Y_i^c} \times 100. \tag{3}$$

Here, Y_i^c and O_i^c are the model's softmax output for class c and the i^{th} input image, and its corresponding masked image respectively. N is the number of images in the dataset. For AD, a lower value is desirable as it reportedly indicates that the masked image has only kept useful regions. IIC is the complement to this and seeks to determine if the masked image results in the model's confidence increasing. The idea behind this is that the explanation has the potential to mask out regions of the image that are detrimental to the network's prediction. IIC is given by:

$$\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}[Y_i^c < O_i^c] \times 100, \tag{4}$$

where $\mathbb{1}$ is an indicator function equal to 1 if the condition in brackets is true, 0 otherwise. For IIC a large value is desirable.

It is notable that the CAM methods that score well on this metric assign high values to a large region of the image. It transpires that this is not coincidental. We hypothesise that what these metrics are actually rewarding is having a large number of high valued pixels. To test this, we propose two experiments. The first is through the use of a baseline, consisting of an explanation which is completely made up of values of 0.9, except for two random pixels with a value of 1 and 0. These random pixels ensure that during any rescaling, the constant values are not scaled to either 0 or 1. We label this baseline as 'Constant'. In Table 1 we show this baseline compared to Grad-CAM and Grad-CAM++. Here, we can see that our baseline outperforms both CAM techniques, while also clearly being a poor explanation method. The second experiment is to create Grad-CAM explanations for each validation set image and then observe how the faithfulness metric change as the scores in the explanation are multiplied by a value from 0-10 in increments of 0.5, and clipped at the maximum value of the original explanation minus $1e{-}10^4$ (ensures rescaling is not an issue). Figure 4 shows an example of this modified explanation. In addition to using Grad-CAM, we also modify our random baseline in the same manner. Further examples shown in the supplemental material.

We show the average AD and IIC scores for this second experiment in Figure 5. In these charts we see how expanding the high value regions of an explanation (either Grad-CAM or random) outperforms a regular Grad-CAM explanation (marked by the dashed red line). Even a small multiplication value increases the performance in this metric. Again, looking at the examples in Figure 4, we see that the explanations that score well in this metric may not be useful for someone trying to understand which image regions are useful to the network.

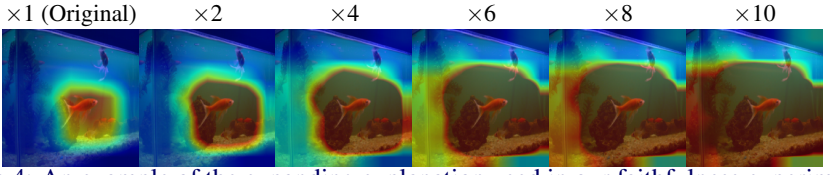

Figure 4: An example of the expanding explanation used in our faithfulness experiment.

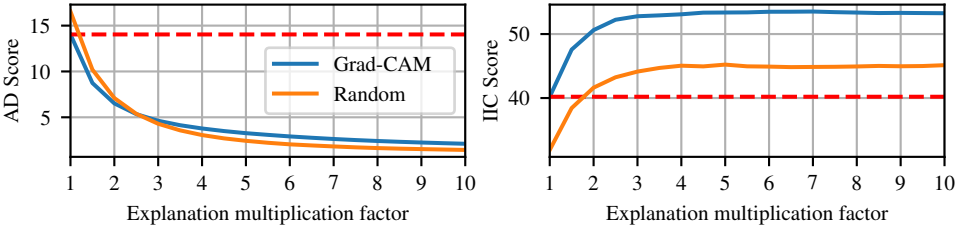

Figure 5: Showing how AD (left) and IIC (right) scores change as we expand the explanations. *Below* the red line for AD is an improvement. *Above* for IIC is an improvement.

While it makes sense that the AD metric might improve, as multiplying the input image by the constant baseline close to 1 should produce little deviation in the model's prediction. It is not immediately obvious why the IIC score should improve. We, therefore, extracted an additional measurement using the constant baseline: the mean amount of improvement $(O_i^c - Y_i^c)$. This showed the reason that the baseline was able to improve the score was that it improved more images, but with a lower mean improvement. The baseline improved the confidence of 24,094 images, with an average confidence increase of 0.022. Grad-CAM and Grad-CAM++ improved the confidence of 20,103 and 19,348 images respectively with mean increases of 0.091 and 0.081. However, none of this is represented in these faithfulness metrics. This suggests that they should be discouraged from future use as they are, at best, misleading and could lead to the development of inappropriate explanation methods.

4.4 Local Accuracy

Faithfulness is often presented alongside the local accuracy measures of deletion and insertion introduced alongside RISE[19]. However, there is a crucial difference between the two types of metric that is important when measuring explanations. Both measurements alter the input image based on the explanation, but while the faithfulness metric simply performs a point wise multiplication (which we have shown is a poor method), deletion and insertion masks the pixels iteratively according to their importance. Masking the input image in this way allows us to understand how well the explanation ranks the pixel's importance. We perform the metric over 28 iterations. The results for both insertion and deletion are shown

	ResNet50		DenseNet		Inception	
Method	Del	Ins	Del	Ins	Del	Ins
Random	0.303	0.413	0.280	0.382	0.287	0.425
Centre	0.177	0.420	0.172	0.374	0.165	0.460
Grad-CAM	0.142	**0.576**	0.137	0.536	0.128	**0.585**
Grad-CAM ++	0.147	0.564	0.141	0.524	0.132	0.572
XGrad-CAM	0.142	**0.576**	0.137	**0.540**	0.128	**0.585**
Score-CAM	0.150	0.568	0.142	0.532	0.131	0.579
Ablation-CAM	0.144	0.567	0.140	0.531	0.129	0.577
RISE	0.131	0.540	0.131	0.513	0.162	0.528
Jitter-CAM	**0.118**	0.551	**0.120**	0.522	**0.107**	0.570

Table 2: AUC for local accuracy metric. Del: Lower is better. Ins: Higher is better.

Method	ResNet50			DenseNet			Inception		
	Val	Mea	Eng	Val	Mea	Eng	Val	Mea	Eng
Random	57.43	58.96	57.39	57.43	58.96	57.39	57.74	59.10	57.66
Centre	47.58	48.18	47.68	47.58	48.18	47.68	48.56	48.34	47.84
Grad-CAM	45.94	45.89	44.35	45.44	44.99	43.48	44.84	45.29	44.60
G-CAM ++	45.76	45.83	43.85	44.89	44.88	42.88	45.05	44.94	44.87
XGrad-CAM	45.94	45.89	44.35	45.67	45.38	43.96	44.84	45.29	44.60
Score-CAM	47.53	46.86	45.32	47.29	46.35	44.74	45.72	45.59	45.19
Ablat-CAM	45.88	45.88	44.30	45.52	45.26	43.71	45.25	45.34	45.00
RISE	52.99	54.10	50.76	51.50	52.99	48.38	55.57	56.66	53.05
Jitter-CAM	**39.83**	**42.30**	**40.64**	**40.24**	**41.55**	**40.44**	**38.38**	**39.10**	**39.86**

Table 3: Weak-localisation results as % of localisation error. Lower is better.

in Table 2. From these results we can see that Jitter-CAM is much better at the deletion metric than the other CAM methods, but this is achieved via a trade-off with the insertion metric. As Qi *et al.* [20] found, as deletion scores improve, typically insertion scores fall. This suggests that Jitter-CAM is able to better locate the pixels deemed most important to the model's prediction, but is less able to determine which pixels are required when rebuilding the image from scratch. RISE outperforms CAM methods but notably struggles with the inception network, likely due to the increased activation map size.

4.5 Weak Localisation

The original CAM method [38] was primarily aimed at the localisation of objects. As such, the ability of CAM-based methods to localise an object well has been used in number of tasks such as segmentation [18, 31, 32], object recognition [7], and person re-identification [35]. Localisation metrics are therefore often presented in CAM papers. We show results for two localisation metrics: weak localisation [2, 36], and the pointing game [36].

In weak localisation, the explanation is thresholded using one of three methods, and a bounding box drawn around the explanation. This is done over a range of thresholds and the best score for each is presented. The first method of thresholding is based on scaling the explanation between 0 and 1, then sweeping through a range of thresholds in range $[0 : 0.05 : 0.95]$. This is labelled as 'Val'. The second set of thresholds is obtained by multiplying the mean value of the explanation with a value in the range $[0 : 0.5 : 10]$. This is labelled as 'Mean'. The final method is based on thresholding the heatmaps by the percentage of energy that covers a subset of the explanation in range $[0 : 0.05 : 0.95]$. This is labelled as 'Eng'. The results for these three methods are shown in Table 3. We see that Jitter-CAM offers a dramatic improvement over existing techniques, improving by around $2\% - 5\%$ depending on thresholding method. The likely reason for this improvement is that Jitter-CAM is able to better highlight more of the object used in the models predictions than previous methods. This in turn results in regions that better align to the ground-truth bounding boxes.

4.5.1 Pointing Game

The second metric used is the pointing game by Zhang *et al.* [36]. This takes a different approach to the previous weak localisation metric in two ways. The first is that the COCO [16] dataset containing multiple objects in an image is used. The second is that rather than thresholding the explanation to find regions which overlap with a bounding box, the maximum point on the explanation is used instead. This maximum point is said to be a hit if it falls (within a 15 pixel margin of error) on one of the correct annotated regions in an image. Accuracy is given as $\frac{\#\text{hits}}{\#\text{hits} + \#\text{misses}}$. The final aspect to the pointing game is that results are presented on both the entire dataset, and a difficult subset of COCO images. The

Method	All	Difficult
Centre	24.61	17.93
Random	12.33	7.99
Grad-CAM	53.47	49.40
G-CAM++	47.26	42.93
XGrad-CAM	53.46	49.39
Score-CAM	47.28	42.70
Jitter-CAM	**64.08**	**61.10**

Table 4: Pointing game results. Higher is better. Jitter-CAM outperforms all other methods.

Method	ResNet50	DenseNet	Inception
Grad-CAM	**0.03** (1)	**0.07** (1)	**0.06** (1)
G-CAM++	**0.03** (1)	**0.07** (1)	**0.06** (1)
XGrad-CAM	**0.03** (1)	**0.07** (1)	**0.06** (1)
Score-CAM	3.83 (2048)	1.92 (1024)	5.01 (2048)
Ablation-CAM	0.60 (2048)	0.30 (1024)	0.60 (2048)
RISE	17.58 (8000)	18.24 (8000)	24.27 (8000)
Jitter-CAM	0.37 (64)	0.40 (64)	0.67 (81)

Table 5: Mean computation time in seconds with number of passes required in parentheses.

difficult subset consists of images that contain objects from more than one class, that are in total smaller than 25% of the image by area. We use the pointing game implementation and pre-trained ResNet50 model from Fong and Vedaldi [5].

The results for ResNet50 can be seen in Table 4. From these results we can see that Jitter-CAM offers significantly better localisation abilities across both the regular and difficult datasets. By increasing the spatial resolution, we are able to be much more precise in where we can point. Previous methods are limited by the coarseness of the explanations.

4.6 Efficiency

Using Jitter-CAM introduces an element of inefficiency to the process of creating explanations due to the multiple CAMs required. In Table 5, we show the mean time in seconds to compute a single explanation. Methods requiring only a single pass (Grad-CAM, Grad-CAM++, and XGrad-CAM) are the most efficient. Jitter-CAM is slightly slower than single pass methods, performing similarly to Ablation-CAM, but faster than Score-CAM and RISE. We believe that the small time increase is justified by the improved results in other metrics.

4.7 Guided Jitter-CAM

In the work by Selvaraju *et al*. [23], the authors proposed an extension of Grad-CAM called Guided Grad-CAM. The authors found this technique produced higher resolution results than Grad-CAM, while being more class-discriminative than Guided Backprop. We used the same method to produce Guided Jitter-CAM explanations. When using ResNet50, we found that these outperformed both Guided Backpropagation and Guided Grad-CAM for both local accuracy and weak localisation metrics. Results are found in the supplemental material.

5 Conclusions

In this paper, we have proposed Jitter-CAM, a novel method that allows us to improve the spatial resolution of explanations created using existing CAM techniques. Rather than spend resources trying to improve the accuracy of the activation layer weights, we instead rescale the image and take multiple explanations. These are then combined into a single explanation. Through both visual inspection and and quantitative measurement we show that this technique improves local deletion accuracy, and greatly improves weak-localisation ability. In addition we have also provided evidence for why the faithfulness metric is flawed and its use should be discontinued.

References

[1] 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), 07 2015. doi: 10.1371/journal.pone.0130140. URL http://dx.doi.org/10.1371%2Fjournal.pone.0130140.

[2] 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.

[3] 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. doi: 10.1109/WACV.2018.00097. URL https://doi.org/10.1109/WACV.2018.00097.

[4] Saurabh Desai and Harish Guruprasad Ramaswamy. Ablation-CAM: Visual explanations for deep convolutional network via gradient-free localization. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[5] Ruth Fong and Andrea Vedaldi. Understanding deep networks via extremal perturbations and smooth masks. In *The IEEE International Conference on Computer Vision (ICCV)*, 2019.

[6] 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.

[7] Kun Fu, Wei Dai, Yue Zhang, Zhirui Wang, Menglong Yan, and Xian Sun. MultiCAM: Multiple class activation mapping for aircraft recognition in remote sensing images. *Remote Sensing*, 11(5), 2019. ISSN 2072-4292. doi: 10.3390/rs11050544. URL https://www.mdpi.com/2072-4292/11/5/544.

[8] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. Axiom-based Grad-CAM: Towards accurate visualization and explanation of CNNs. In *British Machine Vision Conference*, 2020.

[9] 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.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 2016.

[11] Andrew G Howard, Menglong Zhu, Bo Chen, Dmitry Kalenichenko, Weijun Wang, Tobias Weyand, Marco Andreetto, and Hartwig Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[12] Gao Huang, Zhuang Liu, Laurens van der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[13] Hyungsik Jung and Youngrock Oh. LIFT-CAM: Towards better explanations for class activation mapping. *arXiv preprint arXiv:2102.05228*, 2021.

[14] 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.

[15] Liangzhi Li, Bowen Wang, Manisha Verma, Yuta Nakashima, Ryo Kawasaki, and Hajime Nagahara. Scouter: Slot attention-based classifier for explainable image recognition. *arXiv preprint arXiv:2009.06138*, 2020.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014.

[17] GrÃl'goire Montavon, Wojciech Samek, and Klaus-Robert MÃijller. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, 73:1 – 15, 2018. ISSN 1051-2004. doi: https://doi.org/10.1016/j.dsp.2017.10.011. URL http://www.sciencedirect.com/science/article/pii/S1051200417302385.

[18] Huu-Giao Nguyen, Alessia Pica, Jan Hrbacek, Damien C. Weber, Francesco La Rosa, Ann Schalenbourg, Raphael Sznitman, and Meritxell Bach Cuadra. A novel segmentation framework for uveal melanoma in magnetic resonance imaging based on class activation maps. In M. Jorge Cardoso, Aasa Feragen, Ben Glocker, Ender Konukoglu, Ipek Oguz, Gozde Unal, and Tom Vercauteren, editors, *Proceedings of The 2nd International Conference on Medical Imaging with Deep Learning*, volume 102 of *Proceedings of Machine Learning Research*, pages 370–379, London, United Kingdom, 08–10 Jul 2019. PMLR. URL http://proceedings.mlr.press/v102/nguyen19a.html.

[19] 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. URL http://bmvc2018.org/contents/papers/1064.pdf.

[20] Zhongang Qi, Saeed Khorram, and Li Fuxin. Visualizing deep networks by optimizing with integrated gradients. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(07):11890–11898, Apr. 2020. doi: 10.1609/aaai.v34i07.6863. URL https://ojs.aaai.org/index.php/AAAI/article/view/6863.

[21] 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.

[22] Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. *arXiv preprint arXiv:2001.00396*, 2020.

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

[24] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

[25] 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.

[26] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. SmoothGrad: removing noise by adding noise. *ICML workshop on visualization for deep learning*, June 2017.

[27] 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. URL http://arxiv.org/abs/1412.6806.

[28] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319âĂŞ3328. JMLR.org, 2017.

[29] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2818–2826, 2016.

[30] Haofan Wang, Zifan Wang, Mengnan Du, Fan Yang, Zijian Zhang, Sirui Ding, Piotr Mardziel, and Xia Hu. Score-CAM: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[31] Y. Wang, F. Zhu, C. J. Boushey, and E. J. Delp. Weakly supervised food image segmentation using class activation maps. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 1277–1281, 2017. doi: 10.1109/ICIP.2017.8296487.

[32] Yunchao Wei, Jiashi Feng, Xiaodan Liang, Ming-Ming Cheng, Yao Zhao, and Shuicheng Yan. Object region mining with adversarial erasing: A simple classification to semantic segmentation approach. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[33] Nan Wu, Krzysztof J. Geras, Yiqiu Shen, Jingyi Su, S. Gene Kim, Eric Kim, Stacey Wolfson, Linda Moy, and Kyunghyun Cho. Breast density classification with deep convolutional neural networks. In *ICASSP*, pages 6682–6686, 2018.

[34] Dan Xu, Yan Yan, Elisa Ricci, and Nicu Sebe. Detecting anomalous events in videos by learning deep representations of appearance and motion. *Computer Vision and Image Understanding*, 156:117–127, 2017.

[35] Wenjie Yang, Houjing Huang, Zhang Zhang, Xiaotang Chen, Kaiqi Huang, and Shu Zhang. Towards rich feature discovery with class activation maps augmentation for person re-identification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[36] 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.

[37] Xiangyu Zhang, Xinyu Zhou, Mengxiao Lin, and Jian Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[38] 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.