Imperial College
London

Imperial College London

Department of Computing

Explainable Tissue Characterisation for Diagnosis Support in Neurosurgery via Weakly-supervised Segmentation

Author:
Jianzhong You

Supervisor:
Dr. Giannarou Stamatia

Submitted in partial fulfillment of the requirements for the MSc degree in Advanced Computing of Imperial College London

October 21, 2022

Abstract

In the recent development of Artificial Intelligence(AI), models have become much more expressive than ever, thanks to the surge of computational power in the past decades. However, these high-capacity models have a fatal drawback, which limits their application in domains with high cost of errors, such as the medical field. Thus, the area of Explainable Artificial Intelligence(XAI) has emerged to tackle this challenge and to make AI more applicable in different industry sectors.

In this project, we have a new medical dataset containing two tumour classes: Meningioma and Glioblastoma, captured by the Probe-based Confocal Laser Endomicroscopic(pCLE) technology. Our task focuses on developing a better methodology that helps explain the tissue classification decision from the model and thereby supports the medical practitioners during the diagnostic process.

In our work, we first survey various styles of explanation methods in the literature: perturbation-based, propagation-based, and activation-based. Subsequently, we present a framework that achieves state-of-the-art performance, which consists of four parts: feature representation from transfer learning, scale-invariant network architecture [32], layer-wise saliency map aggregations, and more importantly, we propose a novel post-hoc explanation method called axiom-driven Relevance-CAM(XRelevance-CAM). Against the state-of-the-art techniques, up to the ones published in 2021, XRelevance-CAM generates better explanations visually and demonstrates notable performance gains in our primary quantitative evaluation metric, the Weakly-supervised Segmentation Task. Furthermore, a comprehensive set of experiments are performed to verify the significance of individual components in the framework. A dedicated group of evaluation tasks are also completed to verify the theory behind XRelevance-CAM and the faithfulness of its explanations.

Acknowledgements

I want to give my sincere gratitude to my primary supervisor Dr.Giannarou Stamatia and Alfie Roddan for the fruitful weekly discussion, the invaluable advice, and the consistent supports during the writing of this thesis.

To my second marker, Dr. Wenjia Bai. Thank you for the helpful feedback on how to make my report better.

To Professor Patra Charalampaki. Really appreciate the effort you put in for the data collection.

Also, I am deeply grateful for the unconditional supports and encouragements that my parents, Qingxiong You and Xiujuan He, give me throughout my school journey.

Last but certainly not least, to my girlfriend Siyun Hu. Thank you for the love, keep me accompanied throughout my academic year in the U.K, and demonstrate the importance of taking time off for me.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Motivation

Since the Deep Learning(DL) era, Machine Learning(ML) algorithms have proven to have excellent performance scores in various tasks. They have been applied in a wide range of industrial sectors to boost productivity. However, stellar performance is often traded for lower interpretability due to the black-box nature of its decision. Thus the applicability of powerful machine learning algorithms is limited in sectors with low error tolerance. The medical domain is one example that has high cost of errors. Furthermore, accuracy performance of the ML algorithms alone are highly inadequate to earn the trustworthiness of medical practitioners and patients. Hence, the field of Explainable Artificial Intelligence(XAI) emerges to help tackle the challenges.

In addition to making the DL algorithms more transparent behind the final decision using XAI, this domain is also significant for general model diagnosis. Suppose a reliable explanation method capable to highlight input features that differentiate between classes and suppose the correlation score for the saliency map of tumour a and that of tumour b is high for the same input image. In that case, we know that the model is incompetent to distinguish between the two tumours and thus suggest a potential refinement direction of the model before being applied in the medical setting.

Therefore, disentangling the reasoning behind each algorithmic decision is of paramount importance, and the field of XAI plays a role in unravelling the final decision of ML algorithms by constructing human-interpretable explanations, usually a saliency map. In particular, much of the effort is placed on developing explanation techniques that help interpret the classification decision behind the black box for a disease so that it could be applied to support the decision-making process of medical practitioners.

1.2 Brain Tumour Diseases

Tumour diseases are often induced by malfunctioned tissues that proliferate uncontrollably to the surrounding tissues in a body and can generally be classified into two categories: primary and metastatic [1]. Primary brain tumours are tumours that originate from the brain tissue, whereas metastatic brain tumours are the ones that originate from other parts of the body. Brain tumours could further be classified into benign or malignant forms. If the tumour remains in its initial location, it is harmless. On the other hand, in the malignant case, it can migrate to other parts of the body. Diagnosis of tumour disease is often achieved by CT scan, MRI, MRS, PET scan, or close examination of the biopsies by pathologists. The following defines the type of brain tumours that are studied in this project.

1.2.1 Meningioma

This type of brain tumour often appears in an individual's Central Nervous System(CNS); the majority of this type of brain tumour is in a benign state and rarely found to be in cancerous form [59]. According to the statistics obtained between 2012-2016, Meningioma holds accountable for 37.6% [59] of tumour cases that appear in the CNS. Complications could develop as they slowly proliferate and exerts pressure on the surrounding brain tissues. However, it is often curable with surgical intervention or other sophisticated therapies [59].

1.2.2 Glioblastoma Multiforme (GBM)

Glioblastoma Multiforme(GBM) is a type of brain tumour that is much more severe, and aggressive [1] compares to Meningioma. It is one of the most prevalent types of malignant brain tumours. The survival rate of individuals diagnosed with this type of tumour drastically drops to several months of longevity [57] if left untreated. It is responsible for 47.7% [57] of malignant brain tumour cases, and symptoms in companies with this type of disease are often diverse. (depends on the location of the tumour, resultant symptoms differ)

1.3 Project Objectives

The primary objective of this project is to survey the existing state-of-the-art explanation methods on a chosen model that is trained with image-level labels (without voxel-level annotations) on the provided medical dataset that is captured by the pCLE imaging technology regarding the Meningioma and GBM diseases. In addition, we try to develop a new methodology that generates better explanations. Mainly, the quantitative performance is measured by some metrics and qualitative results are also included to complement the assessment of the explanation quality.

1.4 Main Contributions

We summarize the new findings and the novel contributions of the project in this section. Detail description of the findings and contributions indicate at the end of each contribution. Implementation of the project can be found in this Github repository.

1. We survey the performance of the state-of-the-art (up to the ones presented in 2021) activation-driven methods in a new medical dataset obtained from [33].

2. We have shown that Selective Kernel [32] architecture helps enhance the explanation results dramatically through qualitative and quantitative evaluations. (Section 4.2.2 and Section 5.1.2)

3. We propose a novel method **XRelevance-CAM**(Axiom-driven Relevance-CAM) that outperforms other state-of-the-art activation-driven methods in the Weakly-supervised Segmentation task. Mainly, improvement gains by a minimum of 1% in layer four and as much as 10% in layer one compared to our baseline method Relevance-CAM [30], with even greater relative improvements compared to other CAM-based methods. (Section 4.3 and Section 5.1.3)

4. We empirically show that saliency map aggregation across all layers using our XRelevance-CAM achieves significant performance gains both visually and quantitatively; with a minimum of 2.5% performance gain in the Weakly-supervised Segmentation task against Relevance-CAM [30], and as much as 6% of improvement compared to Grad-CAM [50]. (Section 4.4 and Section 5.1.4)

5. We show the theories that motivates the design of XRelevance-CAM and draw connections to the Axiom-based Gradient-CAM(XGrad-CAM) [15].(Section 4.3)

6. We propose a new quantitative evaluation method based on the layer dropout methodology [17]. The method is proposed to evaluate the model uncertainty, but we extend the application of the technique to analyze the average performance of various explanation methods. (Section 5.4)

Chapter 2

Background

2.1 Overview

In this chapter, a broad overview of the representational power of a neural network is given to demonstrate the importance of feature representation capable of driving various fields of Machine Learning into a rapid stage of development. Subsequently, a non-exhaustive literature review regarding only the visual explanation is provided (explain later). Explanation techniques are grouped in their corresponding category. The categorization of the selected methods is inspired by [15] and [58]. Each method is accompanied by a general description so that the reader will get the context of the subjects. Methods that are closely related to or inspire our novelty are provided with more detailed descriptions. In the end, an overview of different evaluation metrics used to validate explanation quality is given. To avoid confusion in the terminology, we use the importance map, saliency map, and sensitivity map interchangeably to the generated visual explanation.

2.2 Representational Learning

Data representation is an integral part of machine learning as it directly correlates with the performance of the ML algorithm [7]. For a more straightforward statistical-based model like multiple linear regression, feature selections are often used to extract the most relevant ones among the chosen features that strongly influence the decision of the model as well as for the sake of parameter efficiency. Traditionally in Machine Learning, feature engineering is a critical step during the data preprocessing stage [7] by harnessing the human domain knowledge to compensate for the fact that ML algorithms are incapable of highlighting discriminative features of data. However, it costs laborious human effort and time. Fast forward to the Deep Learning era, hand-crafted features are no longer needed as researchers have discovered that Deep Learning algorithms can learn abstract features/patterns from data during training [19]. In applied Deep Learning for Computer Vision, Convolutional Neural Network hierarchically discovers shared characteristics from training data by gradually building up high-level concept representations, using features from the lower layers [64]. In applied Deep Learning for Natural Language Processing, rapid progress is achieved when the concept of transformer [60] is invented, in which shared representations in different feature spaces (depends on how many attention heads [60]) can be learned between embedding vectors. In essence, by randomly initializing the Neural Network's weights, different representations or "views" of training data are constructed in the form of matrices that act as feature detectors for the downstream tasks.

2.3 Explanability by Mathematical Structure vs. Visualization

Explainable Artificial Intelligence(XAI) algorithms can be roughly grouped into two major categories: interpretation by visualization and mathematical formulation [58]. Interpretation derived from mathematical construction provides one perspective of interpretation of ML algorithms but requires more cognitive load of an individual to dissect the meaning. Visual explanation gives the other dimension of interpretation, and visualization is often much more intuitive to comprehend. In the health care sector, medical practitioners do not expect to equip the specialized domain knowledge to decipher the non-intuitive explanation behind mathematics. As suggested by [58], cross-disciplinary training in the technical and medical field is encouraged for medical practitioners to equip the technical skills so that mathematical explanations can be utilized to their full potential.

Methods	Category
LIME [46] RISE [44] DeconvNet [64] Prediction Difference Analysis [68]	Perturbation-driven
Guided Backpropagation [54] Integrated Gradient [56] SmoothGrad [53] DeepLIFT [51] LRP [5] CLRP [21] SGLRP [25]	Propagation-driven
CAM [67] Grad-CAM [50] Grad-CAM++ [8] Smooth Grad-CAM++ [41] Score-CAM [61] Smooth Score-CAM [40] XGradCAM [15] Relevance-CAM [30] Layer-CAM [26]	Activation-driven

Table 2.1: Explanation Methods Categorization

2.4 Explanation Methods via Perturbation

2.4.1 Overview

Interpretation via perturbation is a convenient class of explanation methods that provides a visual explanation by systematically occluding pixels in an image. However, high qualitative and quantitative results are often at the cost of increased computational resources.

2.4.2 Local Interpretable Model-agnostic Explanations (LIME)

LIME, a model-agnostic interpretation technique presented by Ribeiro *et al* [46] intends to apply to any model, even the methods yet to be released. Like some other techniques, this method provides an instance-based interpretation. The interpretation is derived by fitting a human-interpretable model g locally to a black-box ML algorithm(e.g neural network), f. Without loss of generality, the interpretable components are high-level concepts of an image in computer vision. A set of labels for g are obtained by systematically feeding f with inputs that occlude each combination of interpretable components in the input space. Subsequently, a sparse linear regression model [46] is trained to obtain the weights of the interpretable components, where the weights indicate the level of importance for the feature impact on the decision. The limitation of this approach is that human-interpretable components are required to run this algorithm. In the imaging case, high-level concept segmentation is required, but it is yet another challenging task.

2.4.3 Deconvolutional Network (DeconvNet)

Zeiler *et al* [64] propose to use the deconvolutional network [65] to examine a neuron in a feature map to identify the patch of an image that activates it [64]. However, to visualize a whole layer in the network, a perturbation approach is used: an explanation map is generated by using a grey square that systematically slides through the entire image and observes the difference in score in each occluded region. Detailed procedure and the tricks for reversing the feedforward process are provided in [64].

2.4.4 Prediction Difference Analysis

Zintgraf *et al* [68] presents a perturbation-based saliency map method named prediction difference analysis, to visualize contribution of input features. In particular, the resulting visualization highlights positive and negative features for a specific class of interest at the output layer. A higher attribution value for the positive features and a lower value for the negative features will make the class score higher. This method extends the work from [47] by proposing two novel techniques [68]: conditional sampling and multivariate analysis. Conditional sampling is employed mainly to tackle the problem of the infeasibility of marginalizing over all other pixels when occludes the input. Based on the intuition that only nearby pixels are more relevant and distant pixels are less dependent, it approximates the marginalization by sampling over the nearby pixel values (within a defined neighbourhood) around it. On the other hand, the multivariate analysis examines a patch of pixels at once instead of a singular pixel investigation by [47]. The original prediction and predictions for input that differs only in one feature (patch of pixels) are examined using the *weight of evidence*(WE) metrics [47]; the relevancy of a differing feature depends on the value of the WE score. A positive WE score indicates that the differing feature positively influences the class score of interest, and a negative WE indicate that the feature lowers the class score of interest. The final heat map is generated after examining every possible feature of a certain size in the input. However, the generation process requires high computational resources [68].

2.4.5 Section Remark

We do not use the perturbation technique to generate a visual explanation in our work. However, some metrics in the literature often apply perturbation to evaluate the explanation quality without human annotations. As mentioned in the later section, we use one of the perturbation approaches to evaluate the explanation quality.

2.5 Explanation Methods via Propagation

2.5.1 Overview

This class of explanation methods is usually computationally efficient as it only requires a single forward and backward pass.

2.5.2 Guided Backpropagation

Springenberg *et al* [53] invent a technique named Guided Backpropagation to produce a sharp saliency map. The high-resolution result is achieved by combining the gradient manipulation technique inspired by DeconvNet [64] and the vanilla backpropagation approach. Essentially, for each neuron in the network that has ReLu [3] as an activation function, the gradient successfully passes through only when the input of that neuron and the gradient coming into that neuron are positive. The shortcomings of this method are that it subjects to failure for an image that has a homogeneous background [53]. Mahendran *et al* [36] also have shown that this method is class-indiscriminative.

2.5.3 Integrated Gradient

The Integrated Gradient attribution method presented by Sundararaja *et al* [56] is invented by following the two axioms proposed: *Sensitivity* and *Implementation Invariance*. Methods ([64], [54]) that disobey the *sensitivity* rule are pruned to neglect important features and methods that subject to implementation variance ([51] [5]) are pruned to assigned non-unique set of attributions to the input features, which is undesirable. Unlike other methods, for two models that are functionally the same, Sundararaja *et al* [56] mathematically proven that the attributions of features arise from the Integrated Gradient are unique (*implementation invariant*) and satisfy the *sensitivity* axiom. In particular, the technique satisfies the *sensitivity* axiom by employing a baseline and satisfies the *implementation invariance* axiom by accumulating the gradient information from the baseline to the input dimension of interest using the first principle of integration in calculus, Riemann summation. However, the shortcoming of this approach is that the quality of the integral approximation is positively correlated with the computational cost (more terms inside the Riemann sum) and Integrated Gradient sometimes produces deceiving attributions [51] to input features.

2.5.4 Smooth Gradient (SmoothGrad)

A sensitivity map produced by vanilla gradients gives a noisy visualisation, which is impractical for a visual explanation. Smilkov *et al* [53] presents a simple modification that drastically discards the amount of noise by averaging the sensitivity maps of the same image that is perturbed with different sampled Gaussian noise. Specifically, augmenting an image with noise does not change its visual appearance to human eyes but heavily

influences the sensitivity maps. A series of experiments are also performed on different published datasets and shows that the SmoothGrad technique can be combined with other sensitivity map approaches [56][54][52] to produce higher quality results. Furthermore, adding noise to inputs during the training and evaluation phase has experimentally proven that the smoothing effect will be further enhanced [53].
Formally, the sensitivity map from the smooth gradient method is equivalent to the average of N sensitivity maps generated from the vanilla gradient method on noisy inputs. That is [53]:

$$\frac{1}{N}\sum_{i}^{N}\frac{\partial S_c(x+\epsilon_i)}{\partial(x+\epsilon_i)}, \epsilon_i \sim \mathcal{N}(0, \sigma^2) \tag{2.1}$$

Where $S_c(\cdot)$ is the logit score of class c for a particular input and σ^2 is a variance parameter usually chosen experimentally depending on the datasets.

2.5.5 Deep Learning Important Features (DeepLIFT)

Shrikumar *et al* [51] presents a novel technique known as DeepLIFT that enables the gradient information propagates back to the input space based on a chosen baseline. This approach is deliberately designed to prevent assigning misleading importance scores to inputs as well as tackle the gradient saturation and threshold problems described in [51]. Three rules are provided including *the linear rule* (applies to linear function only), *the rescale rule* (applies to nonlinear function only), and *the RevealCancel rule* (applies to nonlinear function only). When backpropagating gradient information with respect to a chosen baseline, the linear rule and one of the nonlinear rules are applied alternatingly (at least in the neural network case) until the input space is reached. The RevealCancel rule is invented to disentangle the positive and negative contributions so that the final result reveals important features in the input space that other techniques incapable of showing. However, this technique fails to satisfy one of the axioms proposed by Sundararajan *et al* [56], the implementation invariance.

2.5.6 Layer-wise Relevance Propagation (LRP)

Layer-wise Relevance Propagation (LRP) is a decomposition technique proposed by Bach *et al* [5], that assigns relevancy to each pixel of the input image by distributing the unnormalised final score backwards in the neural network under constrained manners. One of the propagation rule, $LRP - \epsilon$ is shown to be the same as the element-wise product between gradient and input value within a scale factor [4]. The relevance attribution for each neuron is demonstrated below.

Formally, define S_c as the score preceding the softmax normalisation for class c, $R_i^{l_j}$ is the relevance score for the ith neuron in layer j, then the decomposition of S_c to each neuron of a layer is [5]:

$$S_c = \sum_{n \in l_N} R_n^{l_N} = \sum_{n \in l_{N-1}} R_n^{l_{N-1}} = ... = \sum_{n \in l_0} R_n^{l_0} \tag{2.2}$$

In literature, many relevance score propagation rules are available to compute $R_i^{l_j}$. A more comprehensive list of popular propagation rules can be found in [39]. However, only two of the most commonly used rules are provided because they are used in our implementation discussed in the later chapter. To simplify the notation and without loss of generality, let $R_j^{l_k}$ be the relevance score of neuron j in layer k, a_i be a neuron activation of that layer, and w_{ij} is the weight connecting neuron a_i and neuron a_j in the next layer.
The LRP-ϵ [5] rule is defined as:

$$R_i^{l_k} = \sum_j \frac{a_i w_{ij}}{\epsilon + \sum_i a_i w_{ij}} R_j^{l_{k+1}} \tag{2.3}$$

The LRP-$\alpha\beta$ [5] rule is defined as:

$$R_i^{l_k} = \sum_j \left[\alpha \frac{max(0, a_i w_{ij})}{\epsilon + \sum_i max(0, a_i w_{ij})} - \beta \frac{min(0, a_i w_{ij})}{\epsilon + \sum_i min(0, a_i w_{ij})} \right] R_j^{l_{k+1}} \tag{2.4}$$

Where α and β are hyperparameters.

Propagation rules of the overall network can be customised by assigning different rules for different layers to achieve better results. The recommendation on the usage of each rule can reference Montavon *et al* [39].

Limitation: Applying the custom relevance propagation rule requires modification of the layer, and the implementation is less intuitive than other gradient-based activation methods. Furthermore, this technique fails to reveal the differentiating features between classes [21] and fails to highlight features with low RGB values (limitation inherited from the gradient $\odot$ input approach) for the $LRP - \epsilon$ rule.

2.5.7 Contrastive Layer-wise Relevance Propagation (CLRP)

Gu *et al* [21] has proposed a new technique called Contrastive Layer-wise Relevance Propagation to make the vanilla LRP [5] techniques become class discriminative. This is a simple technique that keeps everything the same as the vanilla LRP, except the final logit score of each class is modified by the following equation [30]:

$$R_c = \begin{cases} L_t & if\ c\ is\ the\ target\ class\ t \\ \frac{-L_t}{N-1} & else \end{cases} \tag{2.5}$$

Where R_c is the relevance score of class i, L_c is the logit score of class c, and N is the total number of classes to be classified.

Note that the score assignment in Eq 2.5 satisfies the following property to ensure that the target class's relevant features are more salient compared to other object classes combined [25]. The property becomes more notable when we have many classes to make a decision, such as the ImageNet dataset [48]

$$R_t + \sum_{i \neq t}^{N-1} R_i = 0 \tag{2.6}$$

2.5.8 Softmax Gradient Layer-wise Relevance Propagation (SGLRP)

Unlike CLRP [21], where it penalises all non-target classes by the same constant $\frac{-1}{N-1}$. Iwana *et al*[25] generalize CLRP[21] by assigning the gradient of softmax to each class so that each non-target class is penalized by a different constant, as follows [25],

$$R_c = \begin{cases} S_t(1 - S_t) & \textit{if c is the target class t} \\ -S_t S_i & \textit{else} \end{cases} \tag{2.7}$$

Where S_i is the softmax probability of class i and t denotes the target class index, note that the relevance score assignment in the final layer also complies with the property 2.6.

Connection to our Work: We use the contrasting technique in Eq 2.5 because of two reasons. First, we only have two classes to contrast with, so the comparative advantage of using SGLRP is less notable. Second, the contrastive rule 2.7 fails to apply in our dataset. We hypothesise that the color value of our data is too small, but further investigations are required to conclude.

2.6 Explanation Methods via Activation

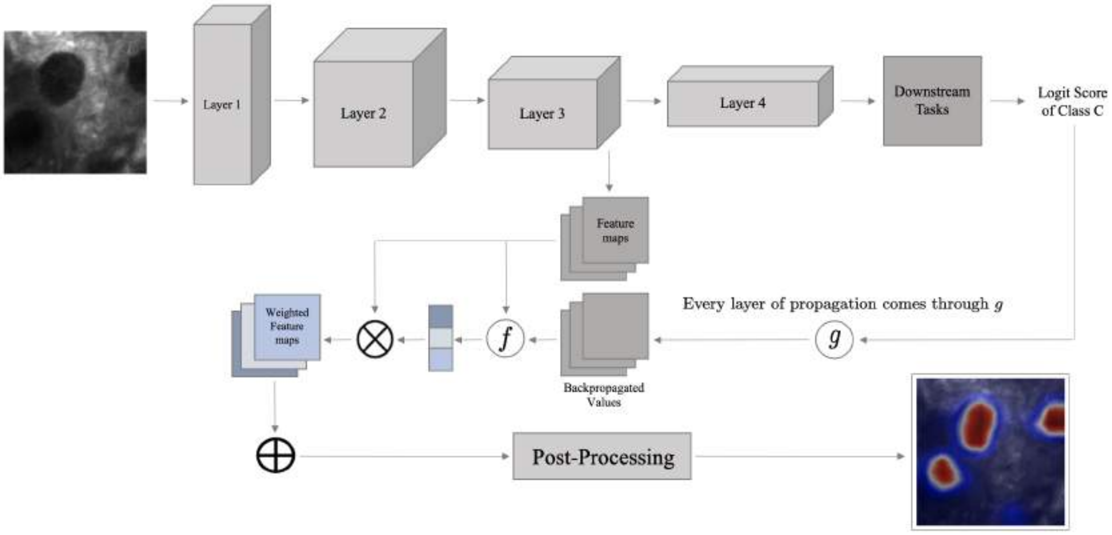


Figure 2.1: General Pipeline for most Activation-driven Methods. This diagram demonstrates the process for visualizing activation maps of layer three, but the workflow is the same for all other layers in the network. The downstream task, in our case, is the fully-connected layer. The propagation starts at the logit score of the class of interest C. g is a function of the gradient. It represents the novel propagation rule that is mentioned in the propagation-driven method, such as the L-$\alpha\beta$ rule in the LRP [5] method or even the vanilla gradient(g is the identity function). Function f is a novel weighting strategy, with backpropagated values and activation as input, defined in a activation-driven method section such as Eq 2.9, Eq 2.11, Eq 2.17. The last step of the workflow is the post-processing stage, which often consists of clipping negative values(ReLu [3]), normalization, and up-sampling.

2.6.1 Overview

Propagation-driven methods transmit the information from the output layer back to the input space. In this class of methods, activation of feature maps is used to highlight salient features in the input. One common limitation of this class of methods is that they all fail to highlight the fine-grained attributes of an input.

2.6.2 Class Activation Map (CAM)

CAM is the primal activation-based [67] method that applies only to a particular family of Convolutional Neural Network(CNN) architecture [67], CNN that uses the Global Average Pooling(GAP) layer [34]. The technique generates an explanation by weighted summation of the feature maps in the last layer and upsamples the weighted feature map to the input size. The weight for the ith feature map is the connection weight of the ith neuron in the global average pooling layer to the cth neuron in the output layer, where c is usually the class of interest.

Formally, suppose the activation map for class c is wanted and M_c is the importance map for class c, then we have the following:

$$M_c = UP\left(\sum_k w_k^c A_k\right) \tag{2.8}$$

Where w_k^c is the weight that shared across all spatial locations of the kth feature map with respects to class c, A_k is the kth activation map in the last layer, and $UP(\cdot)$ is a upsample function.

Limitation: This technique requires changes to CNN models that are not part of the GAP-based architecture family and retraining.

Remark: for the rest of this section, only the computation of w_k^c for each feature map of the desired layer is given because that is the main novelty of this class of methods, and the post-processing of the weighted feature maps is similar to Eq 2.8.

2.6.3 Gradient CAM (Grad-CAM) and its Variants

Grad-CAM

Grad-CAM [50] approach is mainly designed to resolve the limitations of the CAM technique proposed by Zhou *et al* [67] so that the technique can be applied to a much larger family of CNN architectures without modification of the model. Grad-CAM results are produced by weighted summation of feature maps, clips the negative values to zero, and upsample to the input size. The weight for the ith feature map is computed by averaging the gradient of the ith feature map propagates from class c, and the ReLu [3] function is introduced to suppress the non-relevant features. Formally [8],

$$w_{lk}^c = \frac{1}{n \cdot m} \sum_x^n \sum_y^m \frac{\partial S_c}{\partial A_{xy}^k} \tag{2.9}$$

Where n and m are the dimention of a feature map, S_c is the logic score of class c, and A_{xy}^{lk} is the activation value at location (x, y) of the kth feature map in layer l.

In particular, Eq 2.9 helps make the downstream task implementation-agnostic and thus generalize the primal CAM method [67].

HiResCAM

Draelos *et al* [14] propose another type of generalization of Grad-CAM[50], where spatial gradients are further processed by the scale and sign of the activation value before the summation operation to obtain the weight of a feature map. Mathematically, we have the following [14]:

$$w^c_{lk} = \sum_x^n \sum_y^m \frac{\partial S_c}{\partial A^{lk}_{xy}} \odot A^{lk}_{xy} \tag{2.10}$$

Notation definitions and the final saliency map generation are the same in the Grad-CAM section.

2.6.4 Grad-CAM++

Chattopadhyay *et al* [8] have shown that Grad-CAM [50] fails to localize all the objects of the same type in an image. To mitigate this issue, Chattopadhyay *et al* [8] proposed a method named Grad-CAM++ to enhance the localization ability of Grad-CAM by giving a specific importance factor for each positive gradient. In particular, the weight of each feature map is formulated as follows [8],

$$w^c_{lk} = \sum_x^n \sum_y^m \mathbb{1}\left\{\frac{\partial S_c}{\partial A^{lk}_{xy}} = 1\right\} \phi^{l,kc}_{xy} ReLU(\frac{\partial S_c}{\partial A^{lk}_{xy}}) \tag{2.11}$$

Where $\phi^{l,kc}_{xy}$ is another weighting factor for gradient with respects to class c at each spatial location of the kth feature map in layer l, defined as follows [8]:

$$\phi^{l,kc}_{xy} = \frac{(\frac{\partial^2 S_c}{\partial A^k_{xy}})^2}{2(\frac{\partial^2 S_c}{\partial A^k_{xy}})^2 + (\frac{\partial^2 S_c}{\partial A^k_{xy}})^3 \sum_n \sum_m A^{lk}_{nm}} \tag{2.12}$$

The definition of each term is the same as in the previous section, and $\mathbb{1}(\cdot)$ is an indicator function.

Intuition: The formulas look complicated, but essentially each spatial gradient of a feature map receives a different weight by multiplying an additional weighting factor $\phi^{l,kc}_{xy}$, using the second-order information.

2.6.5 Score-CAM

Common problems for activation-driven methods that use gradient propagation are that they are all subject to two issues: gradient saturation and false confidence [61]. Wang *et al* [61] introduce Score-CAM to solve these two limitations. In particular, the weight for the jth feature map is obtained using the model's score from the input mask with the jth feature map (upsampled and normalized). Mathematically, we have [31]

$$w^c_{lk} = f(n(UP(A^{lk})) \odot X) - f(X_{base}) \tag{2.13}$$

$$n(X) = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{2.14}$$

Where w^c_{lk} is the weight for channel k of layer l with respects to class c, $n(\cdot)$ is the min-max normalization function (Eq 2.14) that map the inputs to range between 0 and 1, f is a neural network, A^{lk} is the activations in channel k of layer l, X is the original input, and X_{base} is the baseline image.

Limitation: compare to other techniques that use gradients to formulate the weights, Score-CAM [61] is usually computationally more expensive.

2.6.6 Axiom-based Grad-CAM (XGrad-CAM)

XGrad-CAM [15] is proposed to satisfied two axioms that other activation-driven methods have ignored: *sensitivity* and *conservation*, defined as follow:

$$\textit{Sensitivity: } S_c(A^l) - S_c(A^l \setminus A^{lk}) = \sum_{ij} w^c_{lk} A^{lk}_{ij} \tag{2.15}$$

$$\textit{Conservation: } S_c(A^l) = \sum_{ij} \sum_{k} w^c_{lk} A^{lk}_{ij} \tag{2.16}$$

Where $S_c(A^l)$ is the logit score with A^l as input, $S_c(A^l \setminus A^{lk})$ is the residual logit score after setting $A^{lk} = 0$, other terms have the same definition stated in the previous sections.

Furthermore, Fu *et al* [15] has proven that the following weight formulates approximately satisfy the two axioms. [15]

$$w^c_{lk} = \left\langle \frac{A^{lk}}{\sum_{ij} A^{lk}_{ij}} , \frac{\partial S_c(A^{lk})}{\partial A^{lk}} \right\rangle_{\mathbf{F}} \tag{2.17}$$

Where $\langle \cdot, \cdot \rangle_F$ is the Frobenius inner product. Details derivation of Eq 2.17 refer to the appendix section of [15].

Remark: Note that the conservation axiom defined at Eq 2.16 is different from the conservation property defined at Eq 2.2. To differentiate the two definitions without ambiguity, we call the conservation property defined by Eq 2.16 as **axiom-based conservation** and the other as the **LRP-based conservation**.

2.6.7 Relevance-CAM

Lee *et al* [30] convert the CLRP backpropagation [21] into a activation-based method. The propagation starts with the relevance score defined in Eq 2.5 and decomposes that layer by layer through the rules defined in [39]. In the implementation of Relevance-cam [30], the rules defined in Eq 2.4 and 2.3 are used as the propagation rules.

Formally, let N be the spatial dimension of the kth feature map in layer l, $R^{(lk),c}_{ij}$ is the relevance score of the neuron at location (i, j) of the kth feature map in layer l decompose from the class c neuron, and the importance indicator for the kth feature map in layer l is defined as the average its relevance scores. That is [30]

$$w_{lk}^{c} = \frac{1}{N} \sum_{ij} R_{ij}^{(lk),c} \tag{2.18}$$

Connection to our Work: One observation to make is that Eq 2.18 is similar to Eq 2.8 in the primal CAM [67] method. Therefore, a better weighting strategy can be applied by borrowing ideas from other gradient-based propagation CAM methods to achieve better performance(discussed in later sections).

2.6.8 Layer-CAM

Jiang *et al* [26] propose a new technique called Layer-CAM to tackle the challenge of making the saliency map more fine-grained by taking the shallow layers of a model into account as well as a different weighting strategy. Experimentally, Jiang *et al* [26] found that the variance of gradients in a feature map is high, implicating that an equally weighted strategy for each spatial location is not ideal for revealing the small features. Therefore, a new strategy of assigning weight for each spatial location is proposed [26]:

$$w_{xy}^{lk} = max\left(0, \frac{\partial S_c}{\partial A_{xy}^{lk}}\right) \tag{2.19}$$

Where the notations are the same as the previous sections.

2.6.9 Section Remark

Two points need to be noted.

1. SmoothGrad [53] technique mentioned in the previous section can be applied to all the above methods to produce sharper visualization but at the cost of computation resources. Examples of the smooth variant of the above CAM-based methods are Smooth GradCAM++ [41] and Smoothed Score-CAM [40].

2. M_c produced from any CAM techniques is known to be coarse-grained. To produce an explanation result that reveals more details, element-wise multiplication of M_c and the result from the Guided backProp [54] will result in a class discriminative and fine-grained visualization.

2.7 Evaluation Metrics/Tasks for XAI Methods

2.7.1 Self-Evaluation Metrics

Self-evaluation metric is a class of metrics for XAI that does not require human annotations. This approach eliminates the human effort on the annotations part and is capable of selecting methods that produce a model-faithful explanation, where *model faithfulness* of an explanation is defined as whether the explanation captures the reasoning of the model. However, it might not be the ideal category of metrics to measure the human faithfulness of an explanation map where *human faithfulness* is defined as whether the saliency map looks reasonable/good to humans.

Average Drop (A.D)

Average drop is an automatic evaluation metric introduced by Chattopadhyay *et al* [8] to measure the confidence drop between the output score of the original input and the output score of the salient regions of the explanation map. Formally [8]

$$\mathit{Average\ Drop} = \sum_{i} \frac{100 \times max(0, S^c(I_i) - S^c(M_i))}{S^c(I_i)} \tag{2.20}$$

where $S^c(I_i)$ is the class c score of input image I_i, $S^c(M_i)$ is the class c score of the saliency map of image I_i.

Average Increase in Confidence (I.C)

Average increase in confidence is another self-evaluation metric proposed to complement the A.D metric [8] where $S^c(I_i) < S^c(M_i)$. Formally [8]

$$\mathit{Average\ Increase\ in\ Confidence} = 100 \times \sum_{i}^{N} \frac{\mathbb{1}(S^c(I_i) < S^c(M_i))}{N} \tag{2.21}$$

Where $\mathbb{1}(\cdot)$ is the indicator function and N is the size of the testing data.

Deletion Metric

This is a metric introduced by Petsiuk *et al* [44]. When we have a saliency map, it includes all the salient pixels of an image that has high contributions to the final decision. Gradually removing the pixels that appear in the saliency map from the original input should cause a sharp drop in the output score of the model. We plot all the scores as a deletion curve, with the x-axis being the % of pixels removed and the y-axis as the output score. The Area Under the Curve (AUC) is the final objective measurement for the quality of the saliency map; the lower, the better.

Insertion Metric

This is also a metric introduced by Petsiuk *et al* [44] to complement the deletion metric. Start with an image called inversely segmented input, with all the salient pixels that appear in the explanation map removed from the original input. Contrary to the deletion metric, pixels of the saliency map gradually add back to the inversely segmented input and observe the decision score. We plot all the metric results as an insertion curve, with the x-axis as the % of pixels added and the y-axis being the output score. Higher AUC metric score, in this case, indicates better explanation.

2.7.2 Human-centred Evaluation Tasks

Human-centred evaluation is a class of tasks that requires human annotations, but the models are trained with image-level labels only(or Weakly-supervised). It is a costly metric for evaluation because the cost of human labour is high. However, it helps identify the case where the model decision is driven by the contextual features instead of discriminative cues of the object class. In other words, this class of metrics usually select methods capable of generating human-faithful explanations. However, as [14] and [44] mention, it might not be the ideal metric to select the method that produces model-faithful explanations.

Weakly-supervised Segmentation(WSS)

With pixel-wise annotation provided, we use the Intersection Over Union(IOU) metrics defined in Eq 2.22 to measure the human faithfulness of a saliency map. If a model's output is based on some spuriously correlated features, the model would have a low probability of achieving high performance in this task. A high IOU score indicates a better explanation quality.

$$IOU = \frac{Area\ of\ the\ intersection}{Area\ of\ the\ Ground\ Truth\ +\ Area\ of\ the\ Salient\ Pixels} \tag{2.22}$$

Where the *area of the intersection* is the intersected area of the ground truth pixel annotations and the area of the salient pixels from the explanation map.

Weakly-supervised Localization

This task involves bounding box annotations and usually takes less effort for annotating than pixel-wise labelling. The same IOU metrics(Eq 2.22) is used to measure the quality of the saliency map, and a high IOU score indicates a more faithful explanation to human. In this task, *Area of the Ground Truth* in Eq 2.22 is the area of the bounding box.

2.7.3 Human Faithfulness vs. Model Faithfulness Evaluation Metrics

Researchers continuously seek an objective evaluation metric in the XAI domain to examine the explanation quality. However, as we discussed, there is a dilemma between evaluating the quality from the perspective of model faithfulness and assessing quality via the human faithfulness aspect. Some literature like [14] mentions that IOU metric in WSS should not be used to evaluate importance map because the metric does not reflect the reasoning behind the model's decision but measures how well the explanation from the model illustrates the human intuition on the object class. However, we provide a different perspective on why human-centred evaluation tasks like WSS are necessary for the evaluation pipeline of an explanation.

First of all, in some industrial sectors like the medical domain, one of the ultimate goals is using the explanation to help practitioners to make more informed decisions and gain credentials from the model classification. If an explanation demonstrates that the decision is based on some confounded features, it should be a indication that the model requires further refinement before production. On the other hand, human knowledge is used in judging whether the highlighted parts in an explanation are class relevant. Therefore, in the desired case where we are confident that the model decision is made via class discriminative cues, IOU metric in the WSS task is a sensible approach for evaluating the explanation methods.

Furthermore, features that support the model's decision could bifurcate into two categories, the true positive and the false positive features. Assume all explanation methods can highlight the true positive attributes of an input. When we examine the model faithfulness of an explanation, the metric does not penalize an explanation that includes many false positive cues of the desired class. However, the performance of the same explanation would be penalized when evaluated via human faithfulness.

In conclusion, both types of evaluations are essential to assess the quality of an explanation because they complement the limitation of the other. However, we argue that WSS task with IOU metric should be the predominant evaluation approach in the medical domain because we encourage the method highlights more true contributing features and fewer false positive cues. However, self-evaluation metrics should be used as a supportive assessment or model diagnosis.

2.7.4 Section Summary

Each evaluation metric has its limitations and supremacy. However, the ideal explanation method in the medical setting intends to help the human-in-the-loop decision process. That is, we want the chosen explanation method is based on some metrics that reflect the domain knowledge of the medical practitioners instead of surprising them with explanation methods that output confounded features as a good explanation. Therefore, in our setting, the weakly-supervised segmentation task is the dominant evaluation strategy for quantitative measurement of the explanation methods. We argue that it should be the case in the medical setting but we also include one self-evaluation experiment to make the evaluation pipeline more comprehensive.

2.8 Chapter Summary

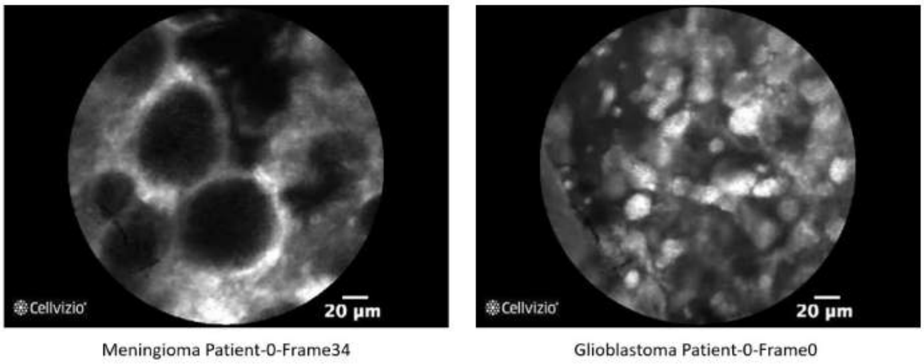
To conclude this chapter, here are a few observations to note. Firstly, most of the propagation-driven methods differ only in their propagation rules(the novelty) being applied to the layers in a network. Secondly, most activation-driven methods differ in how the weight (the novelty) of feature maps are computed. Furthermore, most of the propagation methods has its corresponding version of activation-driven method such as the vanilla backpropagation corresponds to Grad-CAM [50], CLRP [21] corresponds to Relevance-CAM [30], Integrated Gradient [56] corresponds to [49](not discussed), DeepLIFT [51] corresponds to [27](not discussed). Therefore, integrating any new propagation-based methodology into an activation-based method could be a direction of future work as a novelty. Another common property of the mentioned techniques is that they are all post-hoc methods capable of generating visual explanations without retraining. A special class of techniques like [31](not discussed) and [16](not discussed) requires manual integration of the attention module and retraining from scratch. Finally, an overview of the evaluation metrics is provided, and this chapter concludes with each metric's limitations and advantages. For the sake of completeness, a summary table for categorising different methods is presented in Table 2.1.

Chapter 3

Data Source and Data Processing

3.1 Overview

We have a dataset with images of two types of tumours: Meningioma and Glioblastoma(GBM). All the data are in video format captured by the pCLE technology, which requires pre-processing to convert to image form. We have 16 patients in the GBM data and 18 patients in the Meningioma data, and all the data are grouped in the folder of their corresponding patients. Pixel-wise segmentation and bounding box annotations for Meningioma data are done using the MakeSense.AI platform, and the output is in COCO[35] JSON format. There are no annotations for the Glioblastoma class because it is noisy and too difficult to annotate accurately. Sample original images are shown below(Fig 3.1)



Meningioma Patient-0-Frame34

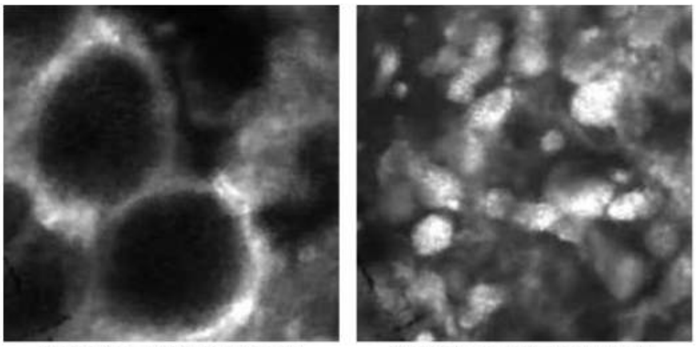
Glioblastoma Patient-0-Frame0

Figure 3.1: Sample Frames of the videos for both tumours

3.2 Data Pre-processing

Like other generic datasets, the first step of the machine learning pipeline is data pre-processing. In particular, as seen in Fig 3.1, the meaningful contents are located in the middle of the image and surrounded by the commercial logo and black pixels. To my knowledge, the popular CNN architectures do not work with circle convolution, and the surrounding black pixels inevitably confuse the models. Therefore, we centre crop each frame to the size of the largest square space within the circle in each frame (230 pixels by 230 pixels) and take every other frame in each video because the consecutive frames are the same. Fig 3.3 shows the amount of data in each patient after the pre-processing stage. The limitation of this approach is that it removes some meaningful contents surrounding the centre box in each frame. However, considering that the missed content most likely be captured in the video's future frames, we do not spend the effort to augment the missed content in the right position in the cropped image. The sample center-cropped image is illustrated in Fig 3.2. After the pre-processing step, we have a total of 12392 images, with 5862 images in the Meningioma class and 6530 images in the GBM class; the pre-processed dataset is approximately balanced.

The next step is to perform data splitting. Unlike other generic datasets, the data splitting process is performed at the patient level instead of the image level because the data of the same patient are highly correlated. In particular, random 80% of all data are used as training data(27 patients), random 10% of the data are used as validation set(three patients), and the rest serves as the test data(four patients) to examine the final performance of the model. A cross-validation procedure could be used, but considering that we have sufficient data and limited time resources, we decided the vanilla splitting strategy is adequate for this project.



Meningioma Patient-0-Frame34

Glioblastoma Patient-0-Frame0

Figure 3.2: Sample centre-cropped frames from both tumor classes. Notice the amount of information loss compare to the same frames in Fig 3.1

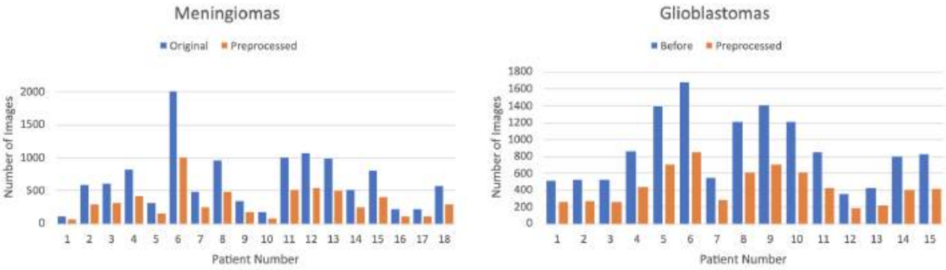


Figure 3.3: Number of images before and after pre-processing for each tumor class

3.3 Data Augmentation

In the stage of pre-processing data, every other frame of each video is used as the actual dataset, but the correlation between images in the same patient is still very high. A combination of data augmentation techniques must be utilized to prevent the model from overfitting the training data, provide a sufficient diversity of input (less correlated) to the model, and thus generalize better and optimally achieve better performance. There are two popular classes of augmentation techniques in literature: Traditional data augmentation and Deep Learning(DL) based augmentation [11].

Traditional Augmentation: most off-the-shelf augmentation techniques come from this category, including rotation, scaling, flipping, contrast, random cropping, etc. [11]. The advantage of this type of augmentation strategy is that they are easy to implement, but the diversity of the augmented data is limited.

DL-based Augmentation: This class of augmentation approach usually involves Generative Adversarial Neural Network(GAN) [20] to produce additional synthetic data. This approach is usually able to generate a more diverse set of data (assume no model collapse) but difficult to implement correctly and high cost of computation overhead and time resources. Other DL-based strategies such as the AutoAugment framework presented by Cubuk *et al*[12] is proven to work well but is dataset specific and has the same limitation as the GAN-style augmentation strategy.

At the training phase of our implementation, four traditional augmentations are used in combination to generate augmented data: random vertical flip, random horizontal flip, random rotation, and random colour contrast. The augmentations are used in a conservative manner to imitate that the augmented data still samples from the same underlying data distribution as the actual data. During the evaluation phase, no augmentation is being used. More importantly, no DL-based augmentation is being used as it is not the focus of this project and is too time-consuming. Finally, data normalisation is used for both training and testing phases so that the input features are of the same scale.

Chapter 4

Proposed Framework for Tissue Characterisation

4.1 Feature Representation From Transfer Learning

A suitable feature extractor is essential for any explanation method to generate a good importance map. In standard practice, using pre-trained weight for a new task is a fundamental step to avoid overfitting the training data, faster convergence time, and enable using a larger model for higher expressive power. The whole process is known as transfer learning. In literature, much of the effort is spent on enhancing the internal representation of complex models like Convolutional Neural Network (CNN) so that the downstream tasks, usually the non-linear classifier, can make efficient use of it to make the decision. Tuning the representation is analogous to the feature selection step of the traditional Machine Learning(ML) pipeline.

For a simple dataset like ours, a model with decent capacity should be able to reach very high accuracy. However, the confounded features and class-relevant features of a class could be the driving force of a model's decision. To emphasize the importance of using transfer learning, Fig 4.1 shows the necessity of identifying correct input features. The visual explanation are generated using Relevance-CAM [30] and the Selective Kernel ResNeXt [63] model is used as the backbone for classification. Observe that when training the model from scratch, the model's decision is based on some spuriously correlated high-level representation, the bright area in Fig 4.1. On the other hand, when training the same model with pre-trained weight obtained from Pytorch [43], the explanation only includes the tumours, which are known to be the dark region of the image. For completeness, visualizations for Glioblastoma are also included(Fig 4.2), where the white regions and the grey surrounding areas are the tumours.

For general tasks, transfer learning using the pre-trained weights that are finetuned on the ImageNet[48] is enough for initialization. In the recent development, a more state-of-the-art approach like self-supervised contrastive learning([9], [10]) and supervised-contrastive learning ([28], [66]) are proposed for representation enhancement and demonstrate even more performance gains in classification task. However, no pre-trained weights learned from the contrastive learning task are available for the backbone model we use in the framework. Therefore, all models in the subsequent experiment section are trained based on transfer learning unless explicitly specified.

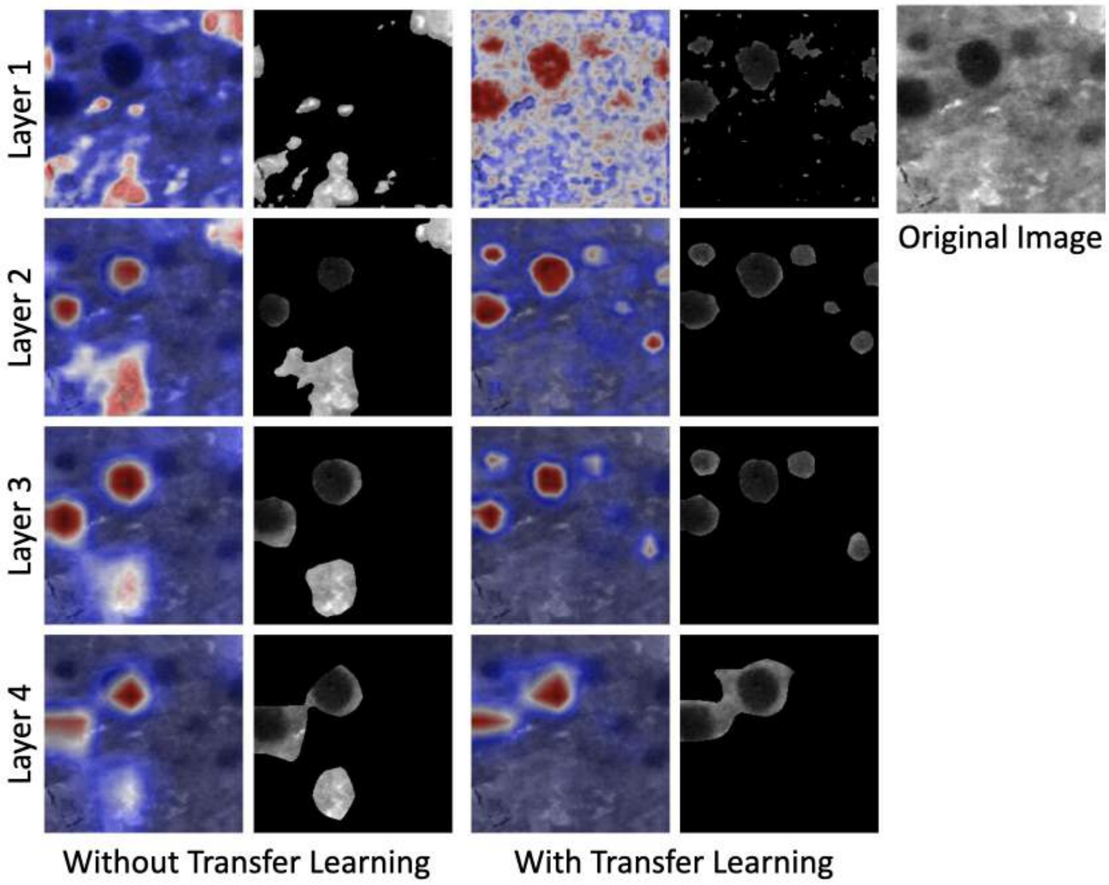


Figure 4.1: Meningiomas: Sample Saliency Maps of each Layer using Selective-Kernel ResNeXt [32] Architecture. All visualizations in this figure are generated using Relevance-CAM [30].

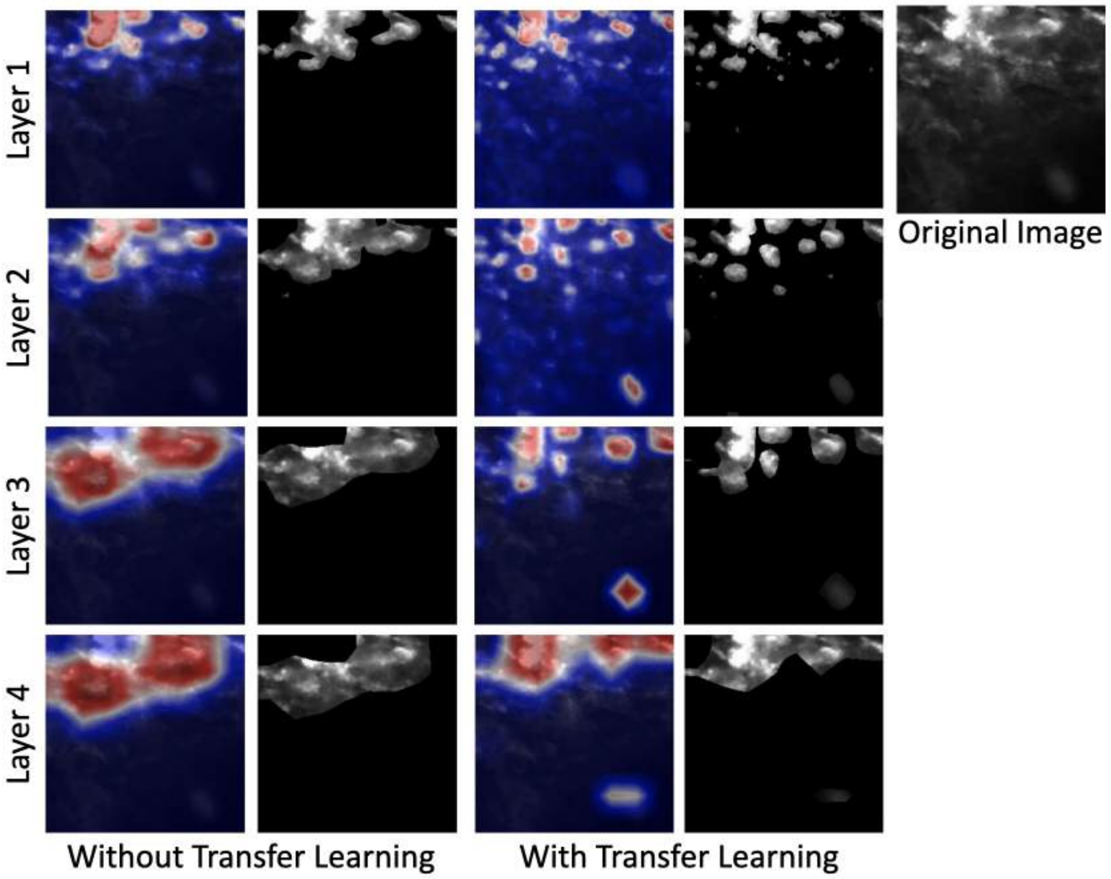


Figure 4.2: Glioblastoma: Sample Saliencys Map of each Layer using Selective-Kernel ResNeXt[32] Architecture. All visualizations in this figure are generated using Relevance-CAM [30].

4.2 Scale-Invariance Design

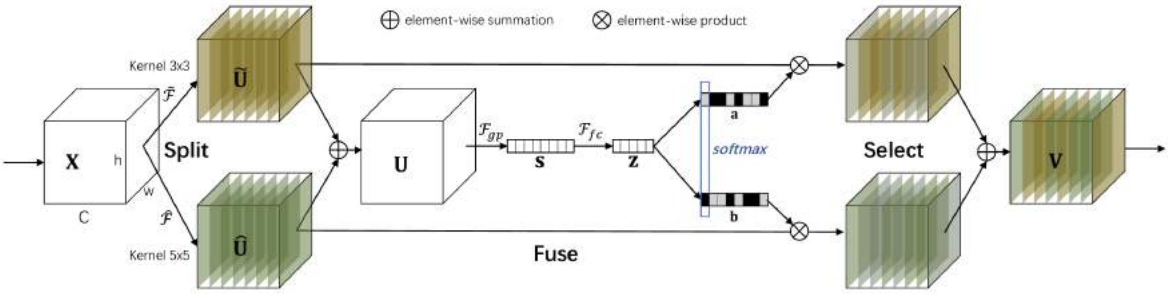


Figure 4.3: Selective Kernel Convolution, cited from [32]

4.2.1 Overview

In medical data, features of the same disease often appear in multiple sizes in the same image. However, most CNN often use the shared receptive field size to capture features even though the patterns might vary in scale. As a result, the model might fail or be less effective in recognizing patterns that appear much larger or smaller than the receptive field size. Several methodologies are proposed to allow neural networks to recognize objects at different scales that belong to the class. Qin *et al* [45] introduce a standalone module that helps capture multi-scale information of the same object through attention[6] on the output of N parallel dilated convolution. Oquab *et al* [42] introduces scale-invariance to the model by feeding the input of different scales in parallel to the model copies during the training phase. Finally, the Selective Kernel(SK) module [32] is another type of convolution designed to learn multi-scale information. We discuss SK [32] in much more detail and use that in our framework because it is the only technique to our knowledge that has experimentally proven its ability to capture multi-scale features.

4.2.2 Selective Kernel

Li *et al* [32] introduce selective kernel(SK) convolution that intends to enhance the representational power through learning multi-scale features at the expense of slight computation overhead. In particular, the multi-scale attention design is achieved through three processing stages. First, the *split* stage is introduced to perform B parallel dilated convolution(B branches) to capture local and global information of the input. Second, the output of each branch is *fused* and forwarded to a compact, fully connected network that outputs a probability tensor($\mathbf{P}$) of size $B \times C$, where B is the number of branches, and C is the number of channels per branch's output. Finally, a *select* operator achieves the adaptive scale attention step. Particularly, $\mathbf{P}$ is normalized at the branch dimension through a softmax function, the output tensor of each branch is multiplied with its corresponding vector at $\mathbf{P}$, and all "soft-attentioned" output tensors are merged by summing across the channel dimension. Details workflow of the selective kernel convolution [32] is illustrated in Fig 4.3.

4.2.3 Selective Kernel Variant of the ResNeXt-50 Backbone (SK-ResNeXt)

In our proposed framework, we use the selective kernel [32] variant of the ResNeXt-50 [63] architecture, where all bottleneck blocks within each layer is replaced with the SK module shown in Fig 4.3 is the only modification. Each SK module has two branches to capture patterns on two different scales. The use of ResNeXt-50 architecture as the backbone is for convenience only in the experiment section because SK-ResNeXt has a pre-trained copy available in the Timm open source library [62] and ResNeXt-50 has enough capacity for representation learning. Detail on the ResNeXt-50 architecture refers to [63].

4.3 A Novel Method: Axiom-driven Relevance-CAM

4.3.1 Propagation Settings and Implementation Details

Logit Scores

Akin to Relevance-CAM [30], XRelevance-CAM do not propagate using the logit scores of the softmax function because LRP [5] is known to be not class discriminative [21]. Therefore, we follow the CLRP [21] approach to backpropagate from the contrasted logit score with slight modification. Specifically, the logit score of all non-target classes is modified to be $-\frac{L_t}{N}$, where L_t is the logit score of the target class. Theoretically, the sum of all relevance scores within each layer should equal to $L_t - (N-1)\frac{L_t}{N}$ during the contrastive propagation instead of L_t.

Internal Propagations

There are many LRP rules in the literature for propagation. We follow the propagation setting of Releveance-CAM [30] by only using LRP-$\alpha\beta$ rule [5] as well as the LRP-ϵ rule [5]. In particular, the LRP-$\alpha\beta$ rule is applied to layers that have learnable weights such as the convolution layer, linear layer, DropOut layer [55], and batch normalisation layer [24]. To make XRelevance-CAM more theoretically grounded, $\alpha = 1$ and $\beta = 0$ is chosen for the LRP-$\alpha\beta$ propagation because it is the only set of hyperparameter that make the rule explainable by Deep Taylor Decomposition (DTD) [38]. The rest of the non-learnable layers are being propagated using the LRP-ϵ rule. Practical implementation of the propagation rules are illustrated in Algorithm.1 and Algorithm.2. With respect to each layer of the architecture, the level of flexibility in choosing which LRP rule to apply is huge. The recommended usage of each LRP rule can be found in [39], and the best result can be achieved using a combination of LRP rules, depending on the depth of the layer. However, a good custom combination of LRP rules depends on many variables, such as the type of layers, the datasets, and the architecture. However, the choice of appropriate layer rules depends on the subjective visual inspection of humans [39].

Implementation Details

Inspired by the official repository of RelevanceCAM [30], we provide the sample implementation in Pytorch [43] for the $L - \alpha\beta$ and $L - \epsilon$ relevance propagation rules. The connection between the implementation and its corresponding equation refers to [39].

In terms of notations, $\odot$ is the element-wise multiplication operator, x is the input of the layer, w is the weight tensor of the *layer*, and *layer* is a PyTorch [43] module. x^+ retains all positive values of x, and negative values are swapped with 0. x^- retains the negative values of x, and positive values are swapped with 0. Similar definition holds for w^+ and w^-

Algorithm 1 Pytorch Implementation for $L - \alpha\beta$ Propagation Rule

```
Require: $x$, $w$. $Layer$, $R$, $\alpha$, $\epsilon$
  $\beta = \alpha - 1$
  $z^+ = Layer(x^+, w^+) + Layer(x^-, w^-)$
  $z^- = Layer(x^+, w^-) + Layer(x^-, w^+)$
  $s^+ = R/(z^+ + \epsilon)$
  $s^- = R/(z^- + \epsilon)$
  $c^+ = x^+ \odot s^+ \odot \partial z^+/\partial x^+$
  $c^- = x^- \odot s^- \odot \partial z^-/\partial x^-$
  $R^{'} = \alpha \times c^+ + \beta \times c^-$
  Return $R^{'}$
```

Algorithm 2 Pytorch Implementation for $L - \epsilon$ Propagation Rule

```
Require: $x$, $w$, $Layer$, $R$, $\epsilon$
  $z = Layer(x, w)$
  $s = R/(z + \epsilon)$
  $R^{'} = x \odot \partial z/\partial x$
  Return $R^{'}$
```

4.3.2 XRelevance-CAM

Overview

Inspired by XGradCAM[15], we propose a new activation-driven method that fills the gap between theory and interoperability, a novel method called Axiom-driven Relevance-CAM(XRelevance-CAM) that brings theoretical backing on the Relevance-CAM [30]. We derive the optimal solution with respect to two axioms proposed by [15]: *sensitivity* axiom and *axiom-based conservation* property, and arrive at the following final weighting strategy.

$$w^c_{lk} = \frac{1}{\sum_{ij} A^{lk}_{ij}} \sum_{ij} R^{lk,c}_{ij} \tag{4.1}$$

Where R^{lk}_{ij} is the relevance score of neuron at location (i, j) of the kth feature map in layer l and x^{lk}_{ij} is the spatial activation at the location.

Remark: In practice, to prevent numerical instability caused by zero division, we define w^c_{lk} as:

$$w^c_{lk} = \frac{1}{\epsilon + \sum_{ij} A^{lk}_{ij}} \sum_{ij} R^{lk,c}_{ij} \tag{4.2}$$

Where ϵ is a very small number close to 0.

Problem Formulation and Results for the Axiom-based Conservation Property

Refers to Eq 2.16 and [15], we have the following optimization problem to find the optimal w_{lk}^c that satisify the axiom-based conservation property.

$$\underset{w_{lk}^c}{\operatorname{argmin}} \left| S_t(A^l) - \sum_{ij} \sum_k w_{lk}^c A_{ij}^{lk} \right| \tag{4.3}$$

By the LRP-based conservation property and the contrastive propagation [21], $S_t(A^l)$ is the class c score with layer activation A^l as input, and $R_{ij}^c(A^l;k)$ is the spatial relevance score as a function of A^l and the kth feature map.

$$S_t(A^l) = L_t = (N-1)\frac{L_t}{N} + \sum_k \sum_{ij} R_{ij}^c(A^l;k) \tag{4.4}$$

$$= \phi_t(A^l) \sum_k \sum_{ij} R_{ij}^c(A^l;k) \tag{4.5}$$

Where $N = 2$ in our case and

$$\phi_t(A^l) = \left((N-1)\frac{L_t}{N} + \sum_k \sum_{ij} R_{ij}^c(A^l;k) \right) \frac{1}{\sum_k \sum_{ij} R_{ij}^c(A^l;k)} \tag{4.6}$$

Combine Eq 4.5 and Eq 4.3, the final optimization problem becomes:

$$\underset{w_{lk}^c}{\operatorname{argmin}} \left| \phi_t(A^l) \sum_{k'} \sum_{ij} R_{ij}^c(A^l;k') - \sum_{ij} \sum_k w_{lk}^c A_{ij}^{lk} \right| \tag{4.7}$$

Observe that Eq 4.7 is a convex program always greater than or equal to 0, which has a unique global optimal solution. We can solve the optimization problem by minimizing the term $|\cdot|$ as follows:

$$\phi_t(A^l) \sum_{k'} \sum_{ij} R_{ij}^c(A^l;k') - \sum_{ij} \sum_k w_{lk}^c A_{ij}^{lk} = 0$$

$$\Rightarrow \phi_t(A^l) \sum_{ij} R_{ij}^c(A^l;k) = w_{lk}^c \sum_{ij} A_{ij}^{lk}$$

$$\Rightarrow w_{lk}^c = \frac{\phi_t(A^l)}{\sum_{ij} A_{ij}^{lk}} \sum_{ij} R_{ij}^c(A^l;k)$$

Therefore, the optimal solution for the axiom-conservation property is

$$w_{lk}^c = \frac{\phi_t(A^l)}{\sum_{ij} A_{ij}^{lk}} \sum_{ij} R_{ij}^c(A^l;k) \tag{4.8}$$

Experiments show that the $\phi_t(A^l)$ term only provides an infinitesimal improvement in the WSS task, and to simplify the expression, we can rewrite the optimal solution as the form of:

$$w^c_{lk} = \frac{1}{\sum_{ij} A^{lk}_{ij}} \sum_{ij} R^c_{ij}(A^l;k) \tag{4.9}$$

Problem Formulation and Results for the Sensitivity Axiom

Refers to Eq 2.15 and [15], we have the following optimization problem to find the optimal w^c_{lk} that satisfy the sensitivity axiom.

$$\underset{w^c_{lk}}{\operatorname{argmin}} \sum_{k} \left| \Big[S_c(A^l) - S_c(A^l \setminus A^{lk}) \Big] - \sum_{ij} w^c_{lk} A^{lk}_{ij} \right| \tag{4.10}$$

Likewise, Eq 4.10 is a convex program and we find the solution by combine Eq 4.5 with Eq 4.10 and equating the $|\cdot|$ term to 0:

$$\left[\phi_t(A^l) \sum_{k'} \sum_{ij} R^c_{ij}(A^l;k') - \phi_t(A^l \setminus A^{lk}) \sum_{k':k'\neq k} \sum_{ij} R^c_{ij}(A^l \setminus A^{lk};k') \right] - \sum_{ij} w^c_{lk} A^{lk}_{ij} = 0$$

$$\Rightarrow \left[\rho(A^l;k) + \phi_t(A^l) \sum_{ij} R^c_{ij}(A^l;k) \right] - \sum_{ij} w^c_{lk} A^{lk}_{ij} = 0$$

$$\Rightarrow w^c_{lk} = \frac{\rho(A^l;k) + \phi_t(A^l) \sum_{ij} R^c_{ij}(A^l;k)}{\sum_{ij} A^{lk}_{ij}}$$

$$\Rightarrow w^c_{lk} = \frac{\Psi(A^l;k)\phi_t(A^l)}{\sum_{ij} A^{lk}_{ij}} \sum_{ij} R^c_{ij}(A^l;k)$$

Where $R^c_{ij}(A^l \setminus A^{lk};k')$ is the recomputed spatial relevance score that satisfy the LRP-based conservation property when $A^{lk} = 0$ in layer l.

$$\rho(A^l;k) = \sum_{k':k'\neq k} \sum_{ij} \Big(\phi_t(A^l) R^c_{ij}(A^l;k') - \phi_t(A^l \setminus A^{lk}) R^c_{ij}(A^l \setminus A^{lk};k') \Big) \tag{4.11}$$

$$\Psi(A^l;k) = \frac{\rho(A^l;k) + \phi_t(A^l) \sum_{ij} R^c_{ij}(A^l;k)}{\phi_t(A^l) \sum_{ij} R^c_{ij}(A^l;k)} \tag{4.12}$$

Therefore, the optimal solution for the sensitivity axiom is

$$w^c_{lk} = \frac{\Psi(A^l;k)\phi_t(A^l)}{\sum_{ij} A^{lk}_{ij}} \sum_{ij} R^c_{ij}(A^l;k) \tag{4.13}$$

Note that the $\Psi(\cdot)$ term is hard to evaluate because it depends on the term $R_{ij}(A^l \setminus A^{lk};\cdot)$ in the $\rho(\cdot)$ expression, which is the redistribution of relevance scores for layer l after the activation values in its kth feature map are swapped with 0.

In practice, to find the optimal solution that satisfies the sensitivity axiom as much as possible, Ablation-CAM [13] is proposed to accomplish this by systematically forwarding the ablated layer K times, each with different feature maps swapped with 0. However, calculating the weights induce a high computational overhead compared to others.

Summary

In summary, we conclude that there is a trade-off between satisfying the axiom-based conservation and the sensitivity axiom. Notably, the solution for the axiom-based conservation can be computed efficiently but only provide a rough estimation for the sensitivity axiom with marginal error upper bounded by $\Psi(\cdot)$. The solution for the sensitivity axiom could be computed systematically proposed by Desai *et al* [13], but subject to high resource expenses. Therefore, we decide to use solution Eq 4.1 to balance the trade-off of optimizing the axiom-based conservation and estimating the optimal answer of the sensitivity axiom.

4.3.3 Connection between XRelevance-CAM and XGradCAM

The L-$\alpha\beta$ rule with $\alpha = 1$ and $\beta = 0$ in a ReLu-based CNN can be rewritten as the following form:

$$R_{ij}^{lk} = \sum_k \sum_{ij} A_{ij}^{lk} \underbrace{\frac{\mathbb{1}(w_{ij}^{lk} > 0)}{\sum_{mn} A_{mn}^{lk} \mathbb{1}(w_{ij}^{lk} > 0)} R_{ij}^{l+1,k}}_{c_{ij}^{lk}} \tag{4.14}$$

Where $\mathbb{1}(\cdot)$ is a indicator function, x_{ij}^{lk} is the activation of the neuron at location (i, j) of the kth feautre map in layer l, and c_{ij}^{lk} is the designated weight for activation x_{ij}^{lk}.

Therefore, an alternative form of Eq 4.1 can be rewritten as:

$$w_{lk}^{c_t} = \sum_{ij} \frac{A_{ij}^{lk}}{\sum_{mn} A_{mn}^{lk}} c_{ij}^{lk} \tag{4.15}$$

We can summarize the connections between XRelevance-CAM and XGradCAM [15] as follows:

1. By comparing Eq 4.15 and Eq 2.17, we see that both equations simply scale the spatial weight by its corresponding normalized activation, where the spatial weight in XRelevance-CAM is c_{ij}^{lk} and the spatial weight in XGradCAM[15] is the gradient.

2. The solution found in XGradCAM is an approximation for both axioms. In XRelevance-CAM, our solution is theoretically optimal for the axiom-based conservation and treated as the estimated solution for the sensitivity axiom by neglecting the $\Psi(\cdot)$ term in Eq 4.13.

3. Fu *et al* [15] experimentally show that the estimated solution is not sensible for visualizing the shallow layers of a DL model. In contrast, the axiom-based conservation property should hold throughout all layers for the XRelevance-CAM.

4.4 Layer-wise Saliency Map Aggregation

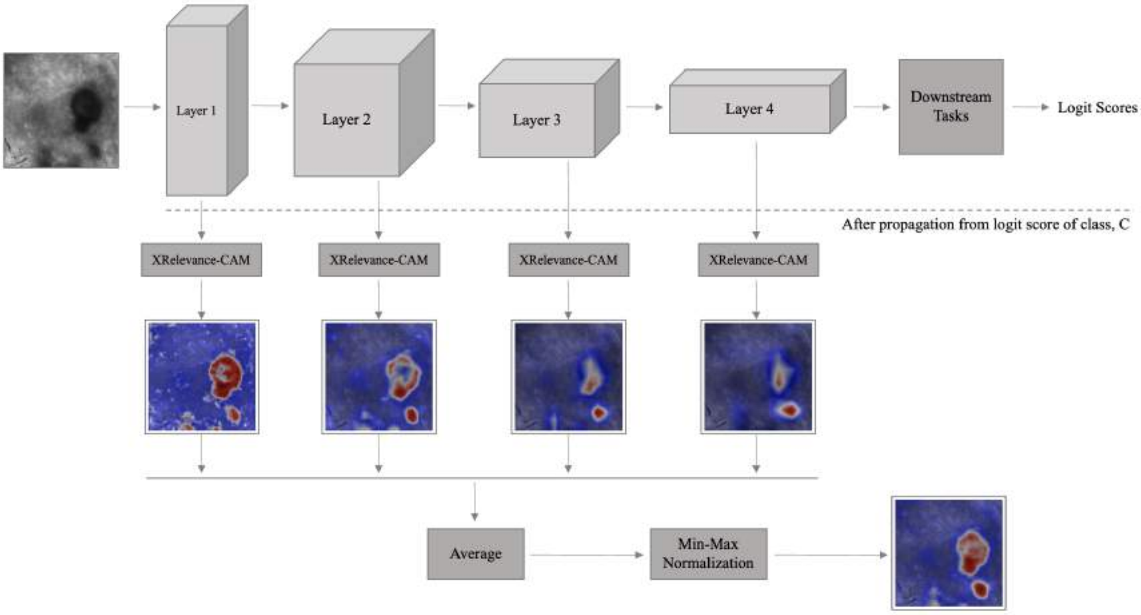


Figure 4.4: Visualization of the Workflow for Saliency Maps Aggregation. A saliency map can be generated from any activation-driven method, but XRelevance-CAM is used in this example. The workflow on generating a saliency map from a layer refers to Fig 2.1.

4.4.1 Overview

Since shallow layers of a neural network often highlight the spatial features of an object but are not class discriminative, upper layers tend to attend to the responsible features of the class of interest but miss the spatial details of the objects. Therefore, by aggregating the saliency map results from all layers, the final saliency map should, in theory, capture the spatial details as well as the discriminative features of a class and thus generate more human-faithful explanations. Note that different aggregation strategies [23], [26], and [37] are proposed, but simple average of saliency maps works best in our dataset.

4.4.2 Implementation Details

The following provides the exact list of steps to generate a final aggregated saliency map. Workflow diagram is shown in Fig 4.4.

1. Generate one saliency map from each layer using XRelevance-CAM(can be any activation-driven method).

2. Simple average across all saliency maps.

3. scale the averaged saliency map with min-max normalisation(Eq 2.14)

4. The final aggregated saliency map is obtained.

4.5 Chapter Summary

To summarize, we give a detailed description of the individual component of the proposed framework. More importantly, we propose a simple but effective CAM-based explanation method called axiom-driven Relevance-CAM and present the theories that motivates the design of the method.

Remark: We develop our novelty based on the official codebase of Relevance-CAM [30] for its propagation implementation, the Timm Open Source Library [62] for obtaining the SK-ResNeXt model implementation, Pytorch CAM library [18] for implementation of other CAM-based methods, and the Pytorch [43] platform for machine learning pipeline implementation.

Chapter 5

Results and Evaluations

In this section, we give a series of experiments to check the performance of the proposed framework. All experiments are done either in a split-out test data and the human-annotated data. Split-out test data contains both Glioblastoma and Meningioma class and the human-annotated data only cover subset of the Meningioma data. Specifically, all Weakly-supervised Segmentation tasks are done in the human-annotated data only and all the quantitative results are obtained using the correctly classified images in the annotated data.

For all the experiments, we only examine three architectures in total: ResNeXt_32x4d(ResNeXt) [63], ResNet50 [22], and the selective kernel variant of ResNeXt_32x4d(SK-ResNeXt) [32]. During the training phase, we use pre-trained weights to initialize all the models. The learning rate begins with 0.001 and adjusts with the AdaMax [29] optimizer. The training is automatically stopped after no consecutive improvement for ten epochs on the split-out validation dataset. Models and the pre-trained weights are implemented and retrieved from the Timm open source library [62]

5.1 Ablation Study of the Proposed Framework

5.1.1 Overview

In this section, we isolate the effect of individual components in the proposed framework by giving an ablation study. In particular, each part of the framework is added progressively to examine its marginal benefit. Most of the experiments are accompanied by qualitative results and metric performance. Weakly-supervised Segmentation task with IOU metric(higher value indicates better performance) is the only quantitative task in this section.

Intersection Over Union(IOU) in Weakly-Supervised Segmentation task

Pixel-wise annotations of the Meningiomas class are provided in our dataset. We utilise the labels to perform the Weakly-supervised Segmentation(WSS) task to evaluate the human faithfulness aspect of the explanation methods. Sample ground truth annotation and the corresponding segmentation of the input are shown on the left of Fig 5.1.

Before the quantitative comparison with the ground truth labels, extra steps are required to generate a segmentation mask from a saliency map. Notably, noise often appears in a saliency map, especially visualisation from the shallow layers of a network. We only keep the most relevant pixels and remove those potential false positive regions by judging their intensity value in the importance map. Concretely, we only keep the pixels of a saliency map with an intensity value at least one standard derivation away above the average. Formally, we use the following equation to extract the most relevant features in a saliency map. Sample salient region mask, as well as the segmentation of the saliency map, are demonstrated on the right side of Fig 5.1.

$$M^I = \mathbb{I}\{I > \mu(I) + \sigma(I)\} \tag{5.1}$$

Where I is the input image, M^I is the mask of I, $\mu(I)$ is the average pixel value of I, $\sigma(I)$ is the standard deviation of the pixel values in I, and $\mathbb{I}$ is a element-wise indicator function where $M^I_{xy} = 1$ if $I_{xy} > \mu(I) + \sigma(I)$, otherwise 0.

Segmentation of the most relevant regions are done by the following equation:

$$M^I \odot I \tag{5.2}$$

Finally, after we obtain the resulting mask of a saliency map, we use the IOU metrics(Eq 2.22) to quantitatively evaluate the quality of the explanation map by comparing it with the ground truth mask.

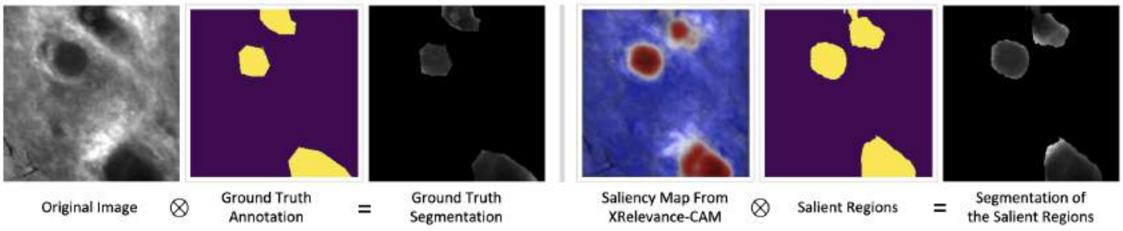

Original Image
⊗
Ground Truth Annotation
=
Ground Truth Segmentation
Saliency Map From XRelevance-CAM
⊗
Salient Regions
=
Segmentation of the Salient Regions

Figure 5.1: *Left:* Sample Ground Truth Mask and its Segmentation. *Right:* the Sample Mask for the Saliency Map and its Segmentation

5.1.2 Importance of the Scale-Invariant architecture

Overview

To examine the benefit that Selective Kernel [32] module brings, we compare the ResNeXt [63] and SK-ResNeXt model side by side to see the marginal improvement of the scale-invariance property.

Results and Analysis

Refer to Table 5.1, each of the two models is evaluated by a set of activation-driven explanation methods. Observe that the SK-ResNeXt architecture consistently exhibits much better metric performance across all examined methods. As demonstrated in Fig.5.2, the majority of the salient regions are spuriously correlated features when visualized from the ResNeXt model, whereas most of the highlighted areas done by the SK-ResNeXt model are class discriminative.

Activation-driven Methods	Use SK [32]	Layer 1	Layer 2	Layer 3	Layer 4	Average Per-layer
GradCAM [50]	False	6.18	14.49	21.04	21.07	15.70
	True	**16.38**	**18.46**	**33.14**	**31.82**	**24.95**
GradCAM++ [8]	False	3.72	6.32	17.71	21.49	12.31
	True	**8.44**	**20.23**	**35.01**	**32.15**	**23.96**
XGradCAM [15]	False	7.52	16.21	25.16	21.07	17.49
	True	**14.40**	**24.77**	**36.20**	**31.82**	**26.80**
HiResCAM [14]	False	12.50	15.45	16.47	23.18	16.9
	True	**15.06**	**21.93**	**30.56**	**31.82**	**24.84**
LayerCAM [26]	False	13.1	16.80	23.40	21.32	18.66
	True	**22.19**	**27.50**	**32.00**	**31.70**	**28.34**
RelevanceCAM [30]	False	15.38	14.64	21.32	20.80	18.04
	True	**18.00**	**30.37**	**33.42**	**31.00**	**28.19**

Table 5.1: ResNeXt [63] vs SK-ResNeXt [32] in the Weakly-supervised Segmentation Task, using IOU(%) metric

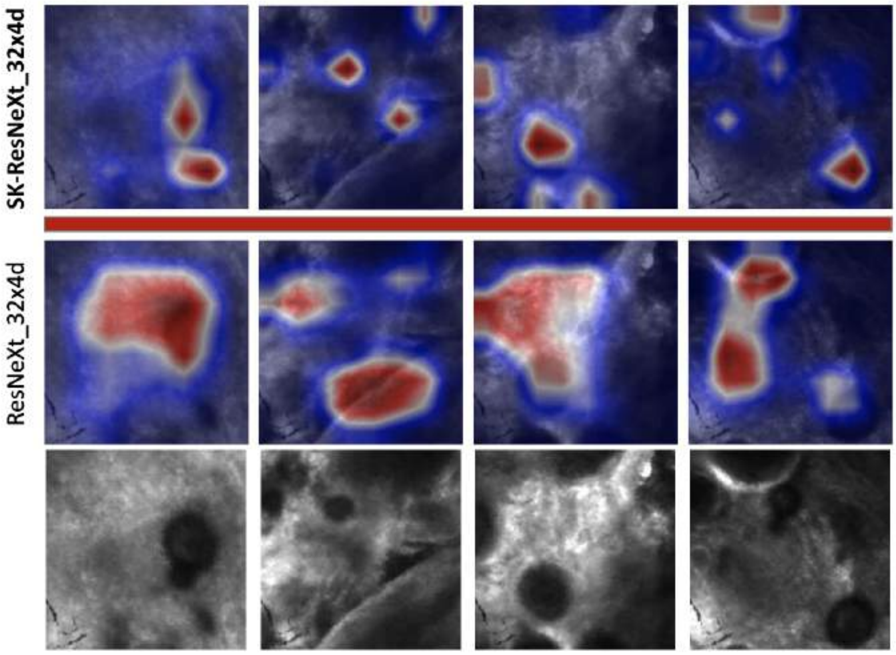


Figure 5.2: Layer 4 Saliency Maps from Both Models. *The bottom row*: this is a list of original images. *The middle row* : this row shows the saliency maps of layer 4 using ResNeXt [63]. *The top row*: this row shows the saliency maps of layer 4 using SK-ResNeXt [32].

5.1.3 Importance of the XRelevance-CAM

Overview

To demonstrate the performance of our XRelevance-CAM in a objective manner, we assess the saliency maps from both qualitative and quantitative perspective by comparing the performance relative to the most recent activation-driven methods, including Relevance-CAM [30], GradCAM [50], GradCAM++ [8], HiResCAM [14], LayerCAM [26], and XGRAD-CAM [15].

Analysis on the quantitative assessment

From Table 5.2, observe that the performance of XRelevance-CAM exceeds all other methods and surpasses by a considerable margin in the shallower layers. In particular, compared with our baseline Relevance-CAM [30], the marginal improvement reached as much as 10% in layer one, and the average per-layer performance of XRelevance-CAM exceeds around 4%. Furthermore, one impression from Table 5.2 is that the performance of all methods in the upper layers is similar; what differentiates the methods apart is the performance in the shallow layers.

Analysis on the qualitative assessment

Several findings can be made from the qualitative results shown in Fig 5.3, Fig 5.4, Fig 5.5, Fig 5.6, and Fig 5.7. Firstly, from layer 2 to layer 4, the saliency maps generated by our XRelevance-CAM are competitive or marginally better than other methods. XRelevance-CAM gives the most visually appealing result for the shallowest layer, even better than Relevance-CAM [30](baseline). Secondly, some methods even fail to give any meaningful visualisation in layer one such as GradCAM [50], GradCAM++ [8], HiResCAM [14], LayerCAM [26], and XGRAD-CAM [15]. Furthermore, observe that XRelevance-CAM has similar explanations produced in layer two to layer four compared to Relevance-CAM [30]. However, the differences between the two methods become much more noticeable from results generated from layer one, where many false positive features highlighted in the saliency map of Relevance-CAM [30] are removed from results generated by our XRelevance-CAM. Fig 5.8, Fig 5.9, Fig 5.10 and Fig 5.11 show the segmentation masks(Eq 5.1) for each layer produced by each CAM-based method to complement the findings observed in the saliency map visualizations and more importantly, the results distinctly demonstrate our XRelevance-CAM has preeminent visual performance in layer one compared to others.

Activation-driven Methods	Layer 1	Layer 2	Layer 3	Layer 4	Average Per-layer
GradCAM [50]	16.38	18.46	33.14	31.82	24.95
GradCAM++ [8]	8.44	20.23	35.01	32.15	23.96
XGradCAM [15]	14.40	24.77	**36.20**	31.82	26.80
HiResCAM [14]	15.06	21.93	30.56	31.82	24.84
LayerCAM [26]	22.19	27.50	32.0	31.70	28.34
RelavanceCAM [30]	18.0	30.37	33.42	31.0	28.19
XRelevance-CAM(ours)	**28.07**	**31.83**	35.11	**32.31**	**31.83**

Table 5.2: Per-Layer IOU(%) Performance in Weakly-supervised Segmentation Task, using SK-ResNeXt [32]. Notice that XRelevance-CAM performs the best among all tested methods in layer one, layer two, layer four, and the average per-layer in the IOU metric. Average Per-layer metric is obtained by averaging the numbers in its corresponding row.

Summary

From both perspectives of evaluation, the saliency map generated from XRelevance-CAM is on par or more aesthetically appealing qualitatively. XRelevance-CAM outperforms all other state-of-the-art(up to 2021) methods that are tested in the Weakly-supervised Segmentation task. In particular, XRelevance-CAM exceeds its closest competitor, Relevance-

CAM [30] by about 4% on the average per-layer IOU metric, and the marginal improvement is as much as 10% in layer one. Therefore, we conclude that our XRelevance-CAM can generate more human-faithful explanations than others.

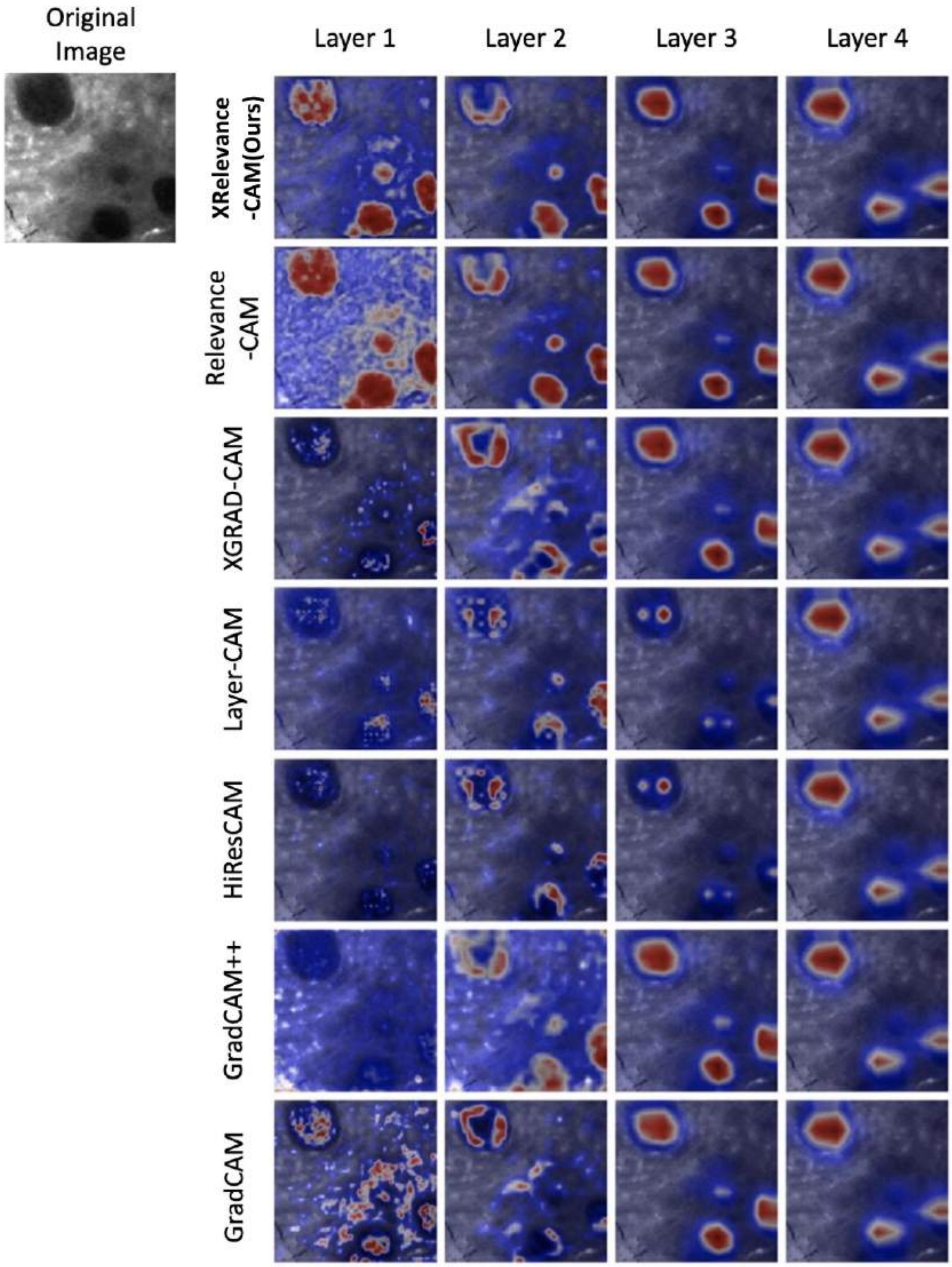


Figure 5.3: Comparison of Various Activation-driven Methods with SK-ResNeXt as the backbone - 1

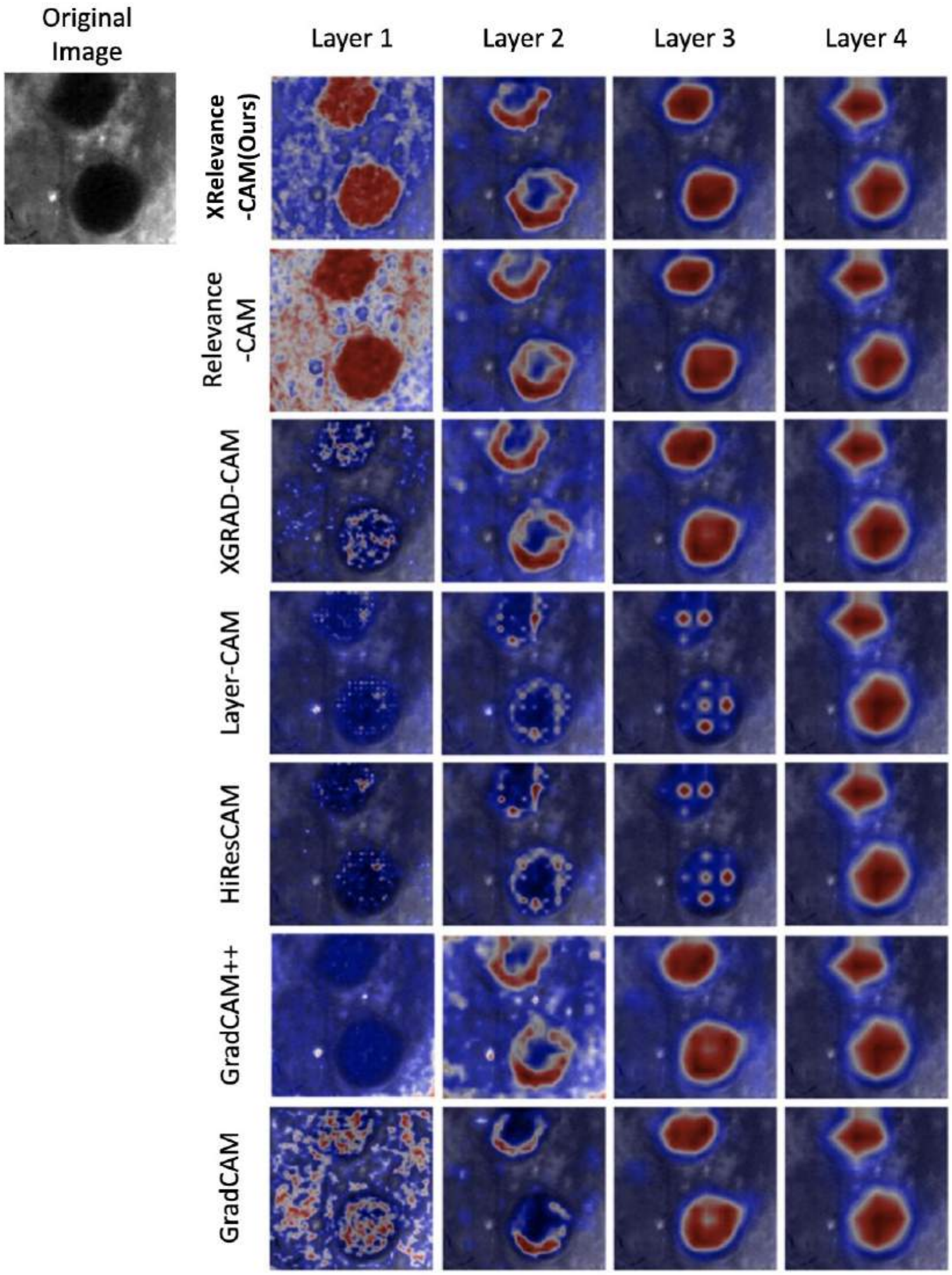


Figure 5.4: Comparison of Various Activation-driven Methods with SK-ResNeXt as the backbone - 2

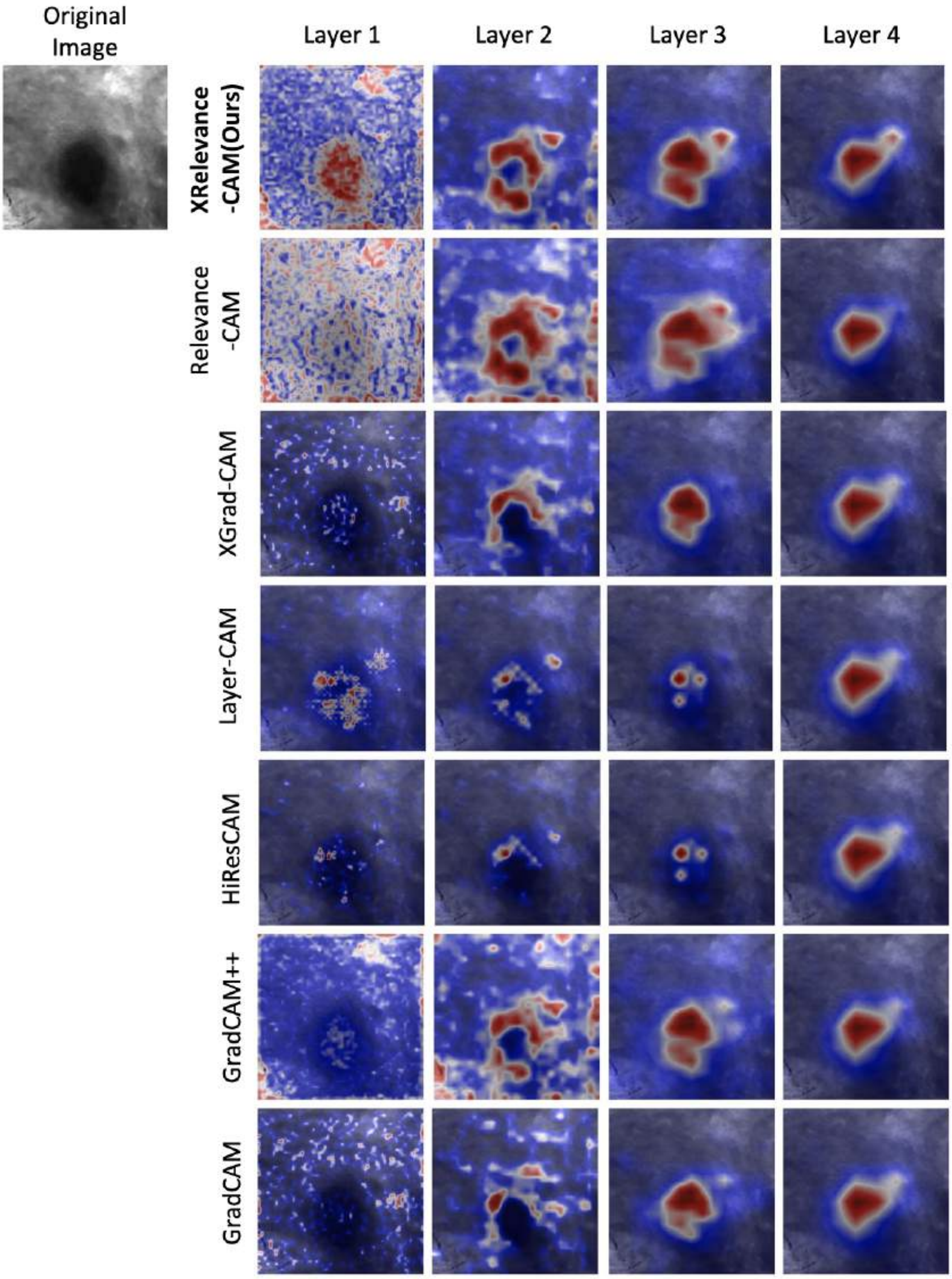


Figure 5.5: Comparison of Various Activation-driven Methods with SK-ResNeXt as the backbone - 3

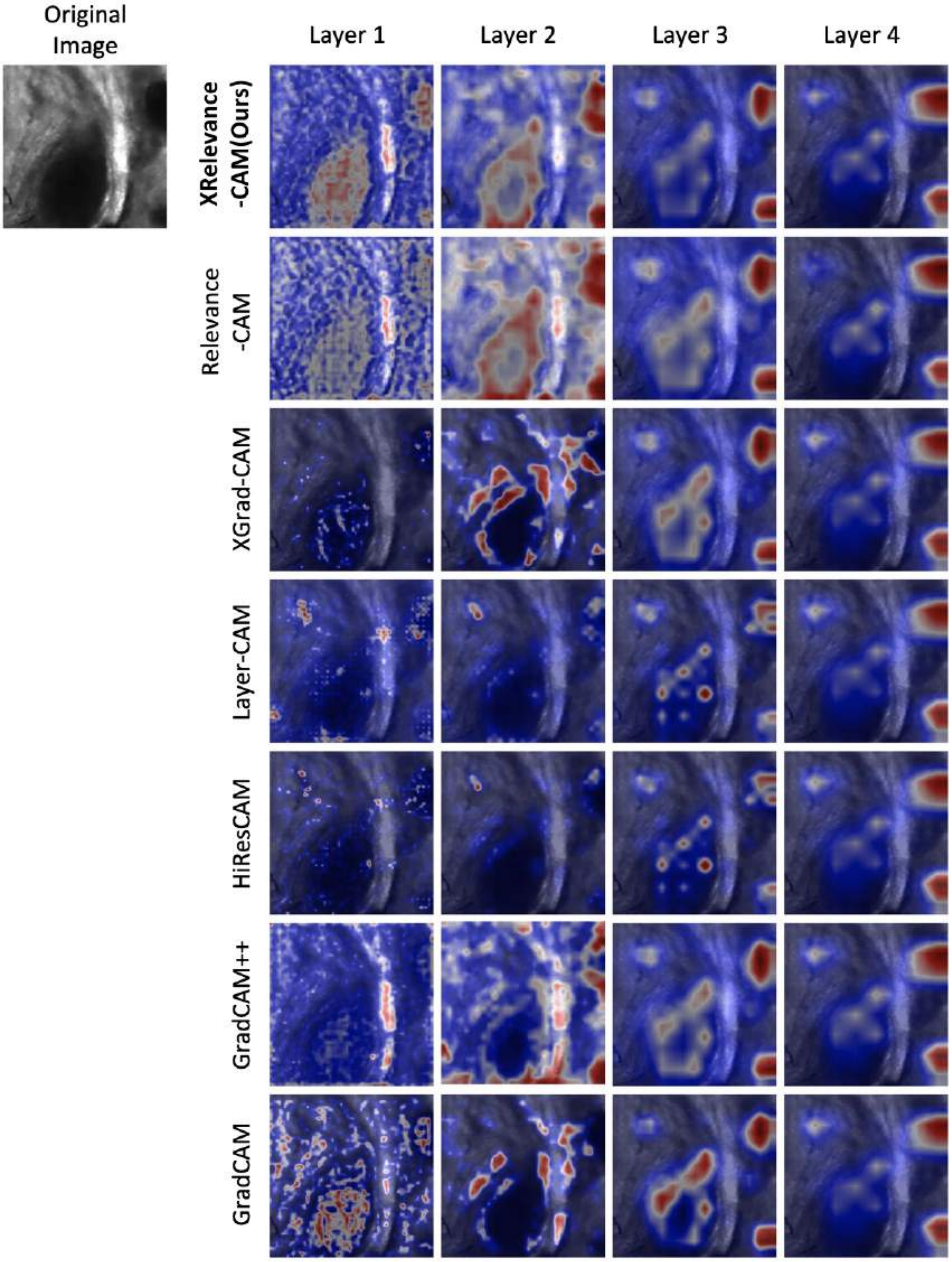


Figure 5.6: Comparison of Various Activation-driven Methods with SK-ResNeXt as the backbone - 4

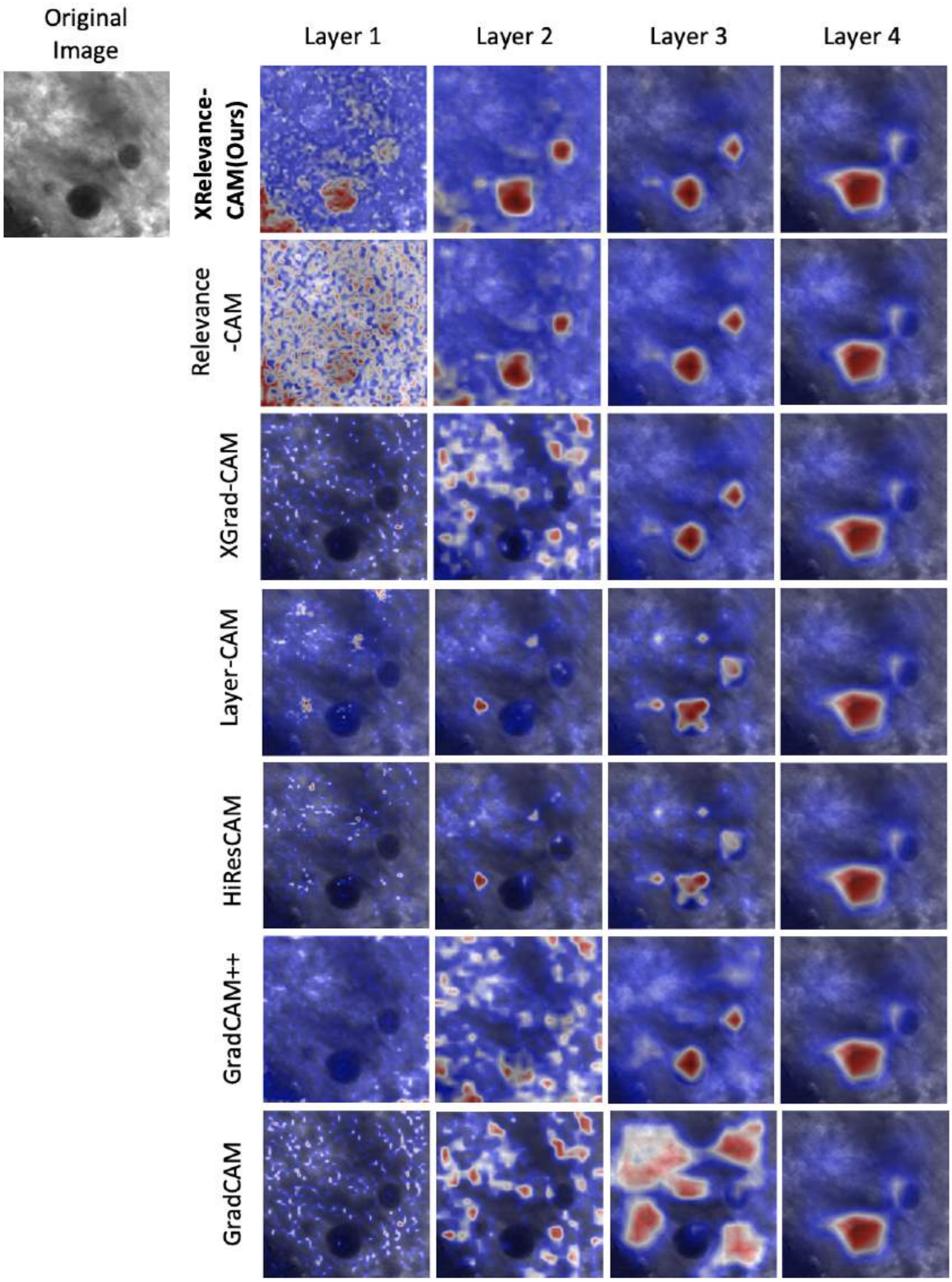


Figure 5.7: Comparison of Various Activation-driven Methods with SK-ResNeXt as the backbone - 5

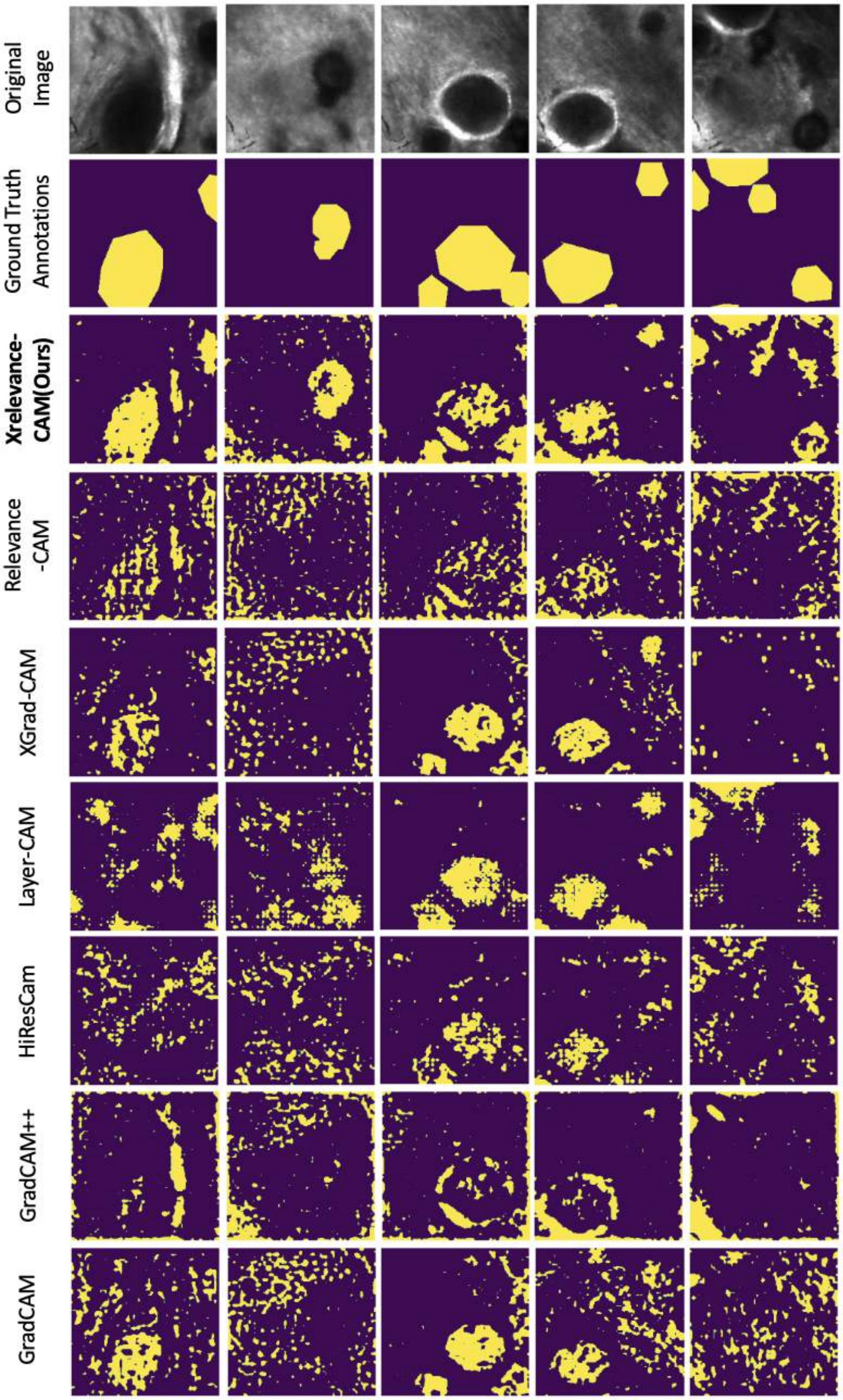


Figure 5.8: Segmentation Masks of Layer One for Various Methods

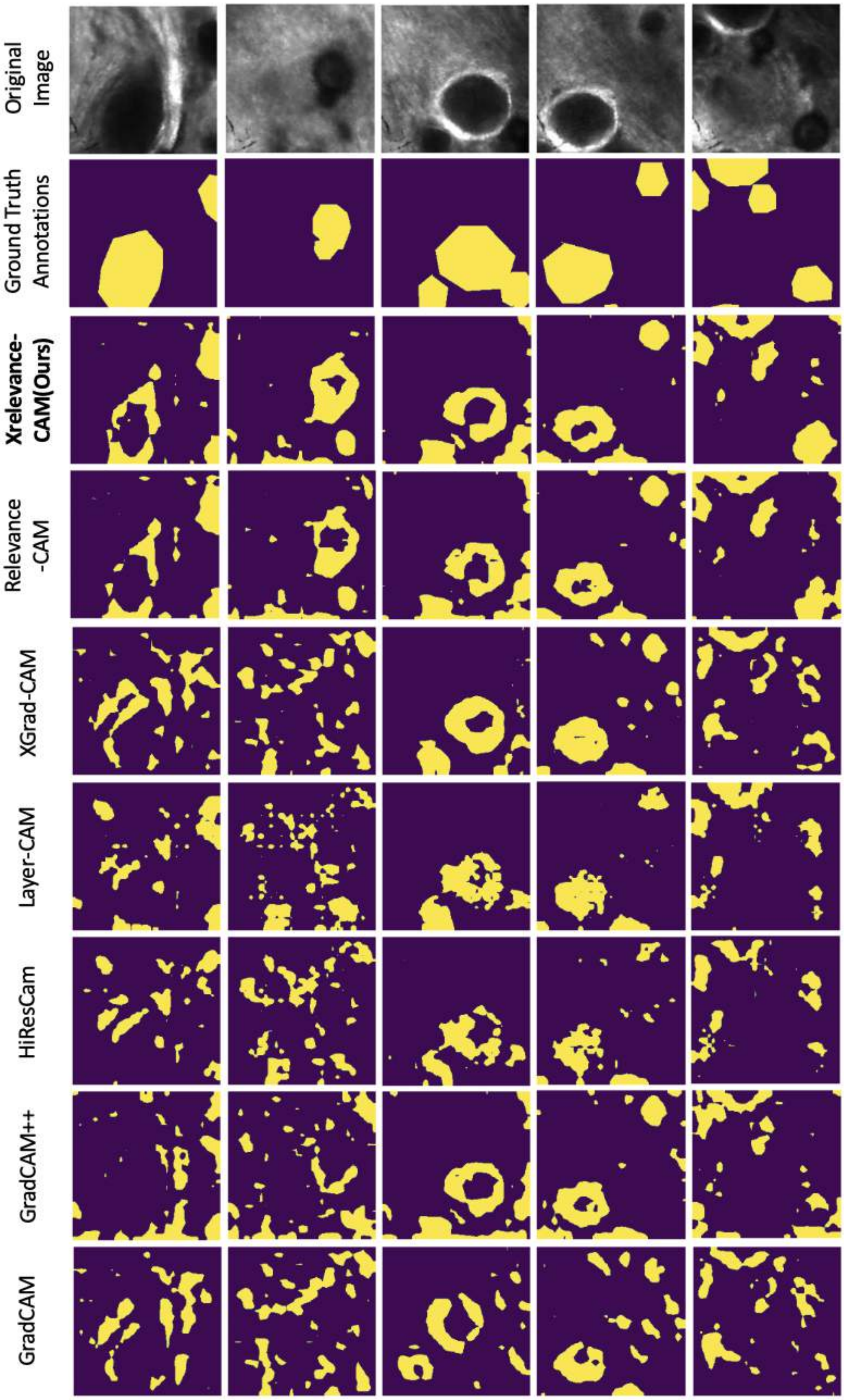


Figure 5.9: Segmentation Masks of Layer Two for Various Methods

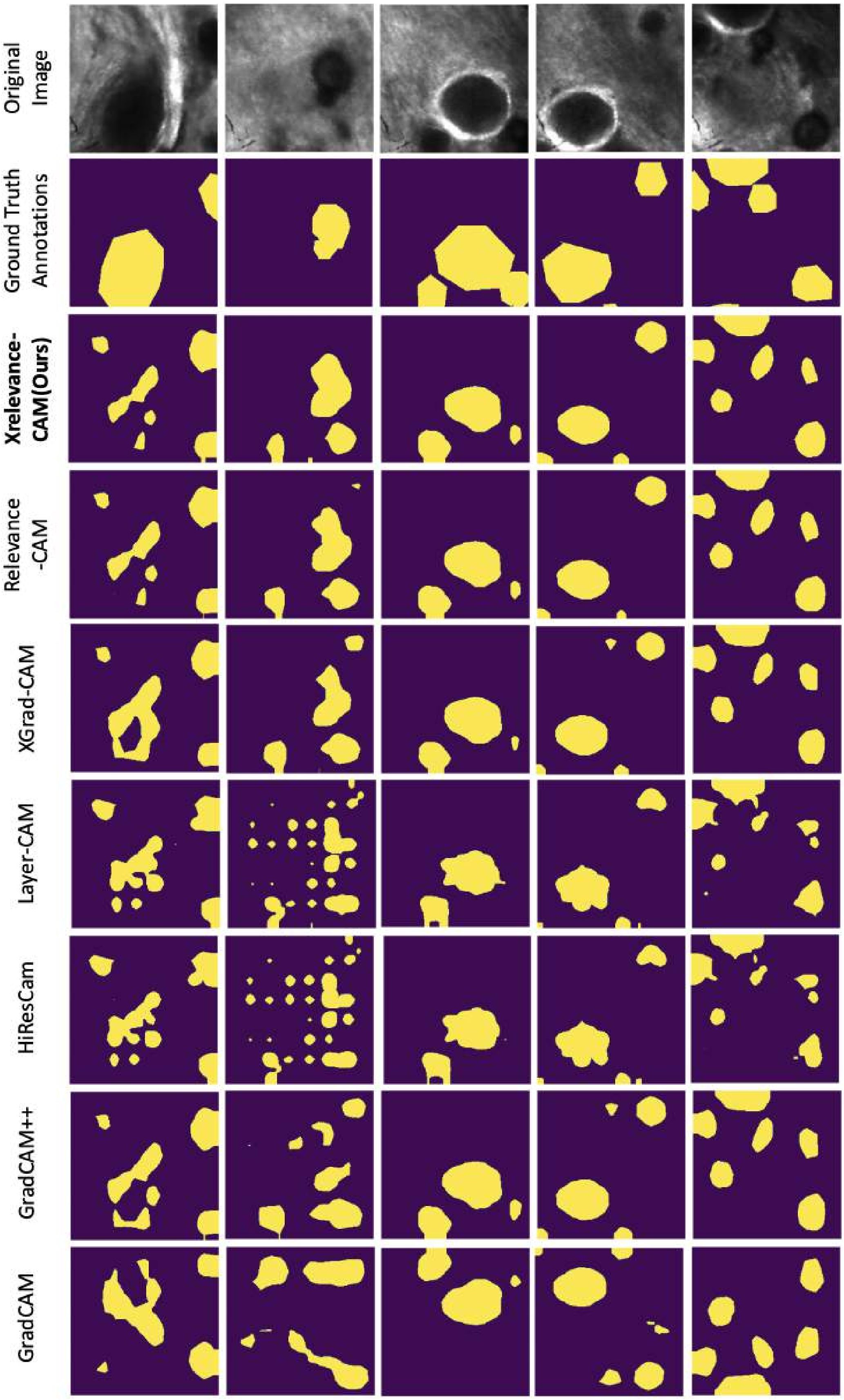


Figure 5.10: Segmentation Masks of Layer Three for Various Methods

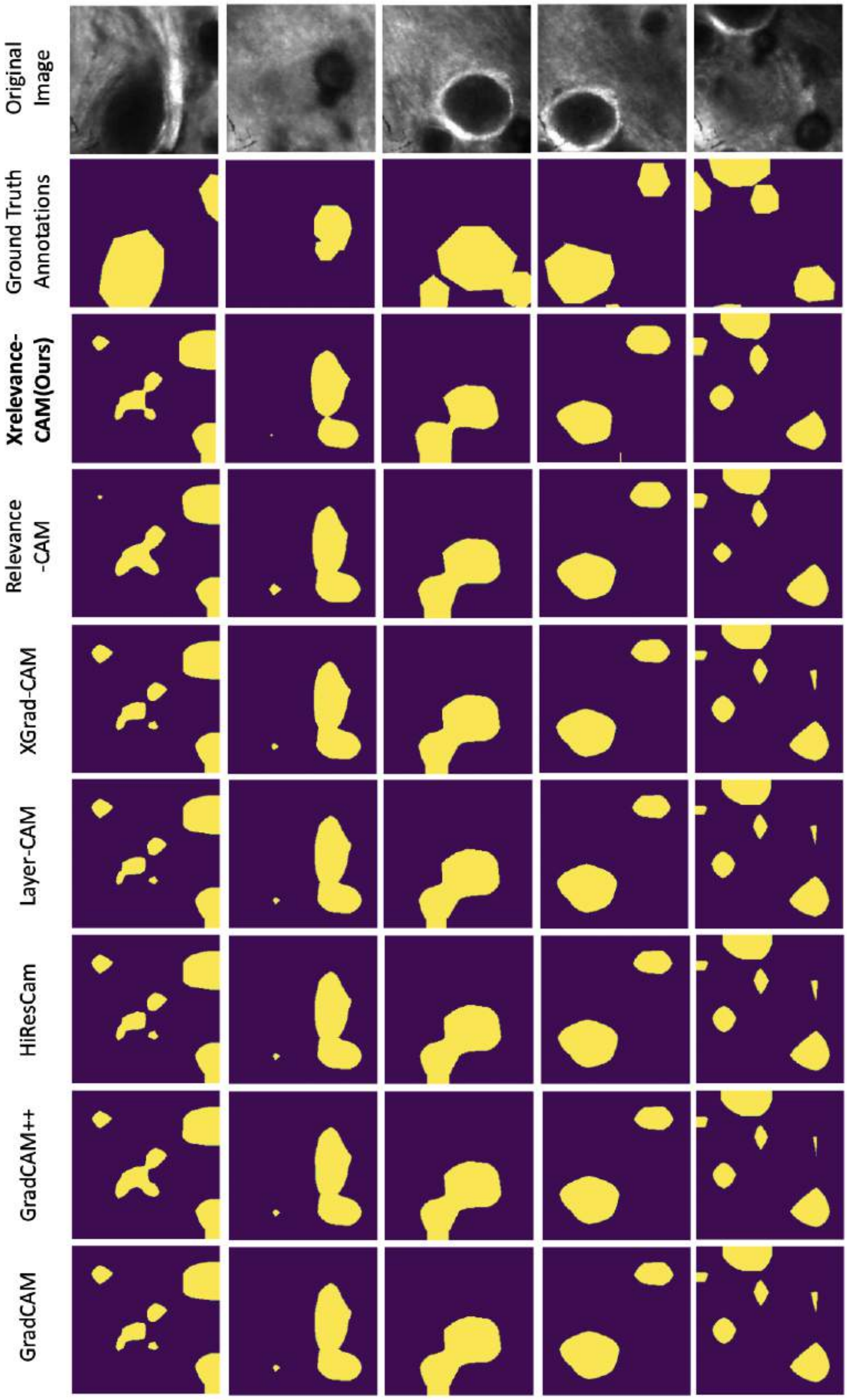


Figure 5.11: Segmentation Masks of Layer Four for Various Methods

5.1.4 Importance of the Layer-wise Saliency Map Aggregation Mechanism

Overview

This section examines the performance gain of aggregating saliency maps across all layers. The experiment is performed by starting from the top layer(layer 4) and progressively combining more shallow layers until the first layer is reached; As such, we can observe the superiority/limitations of including lower layers.

Results and Analysis

Saliency maps of various CAM-based explanation methods are shown Fig 5.13 and Fig 5.14, sample segmentation masks of all-layers aggregation(layer4+3+2+1) are shown in Fig 5.15, and more visualisations of our XRelevance-CAM are included in Fig 5.12. The visualisation results show that the salient features gradually enlarged by including the spatial details captured in the shallow layers. In addition, all tested methods do not exhibit failure cases because the aggregation of saliency maps of the upper layers compensate for the low visual performance(Fig 5.3) in the first layer. For quantitative results reported in Table 5.3, several observations can also be made. Firstly, when all layers are aggregated, we see that XRelevance-CAM(ours) exceeds the baseline, RelevanceCAM [30] by 2.27% and stands out even more, compares to the rest of the methods. Secondly, the marginal benefit of combining layer three is similar across all methods. However, results from shallow layers(layer two and layer one) are much preeminent for XRelevance-CAM(ours). Thirdly, methods such as GradCAM [50], GradCAM++ [8], XGradCAM [15], and HiResCAM [14] fall shorts by aggregating the saliency map of layer one, which might due to its low quality results demonstrated in Fig 5.3, Fig 5.4, Fig 5.5, Fig 5.6, and Fig 5.7.

Activation-driven Methods	Layer 4	Layer 4+3	Layer 4+3+2	Layer 4+3+2+1
GradCAM [50]	31.82	33.86	32.73	31.74
GradCAM++ [8]	32.15	34.87	35.32	33.79
XGradCAM [15]	31.82	**34.99**	35.51	34.95
HiResCAM [14]	31.82	32.77	33.33	32.61
LayerCAM [26]	31.70	33.16	34.70	34.38
RelavanceCAM [30]	31.0	33.13	35.79	35.93
XRelevance-CAM(ours)	**32.31**	34.45	**37.04**	**38.20**

Table 5.3: IOU(%) Performance in Weakly-supervised Segmentation Task with Saliency Maps Aggregation, with SK-ResNeXt as the backbone

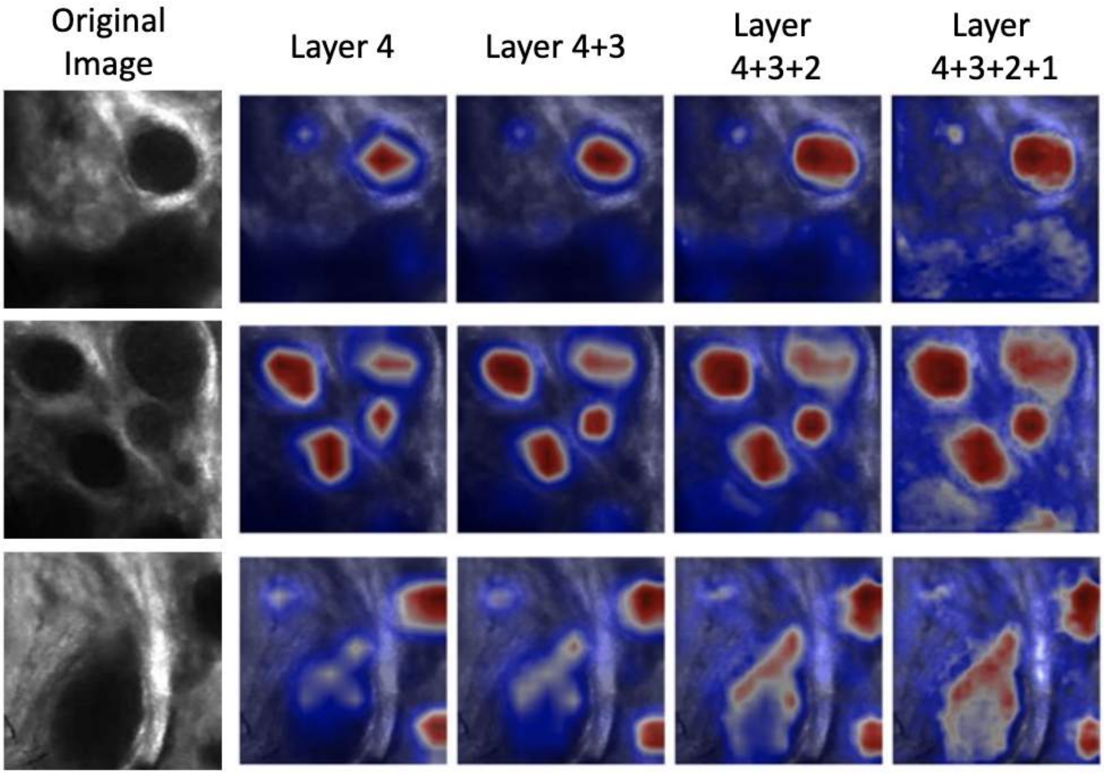


Figure 5.12: More Visualizations of XRelevance-CAM using the Layer-wise Saliency Map Aggregation Mechanism with SK-ResNeXt as the backbone. Observe the contour area of the salient regions(the tumours) gradually enlarged as more shallow layers being considered.

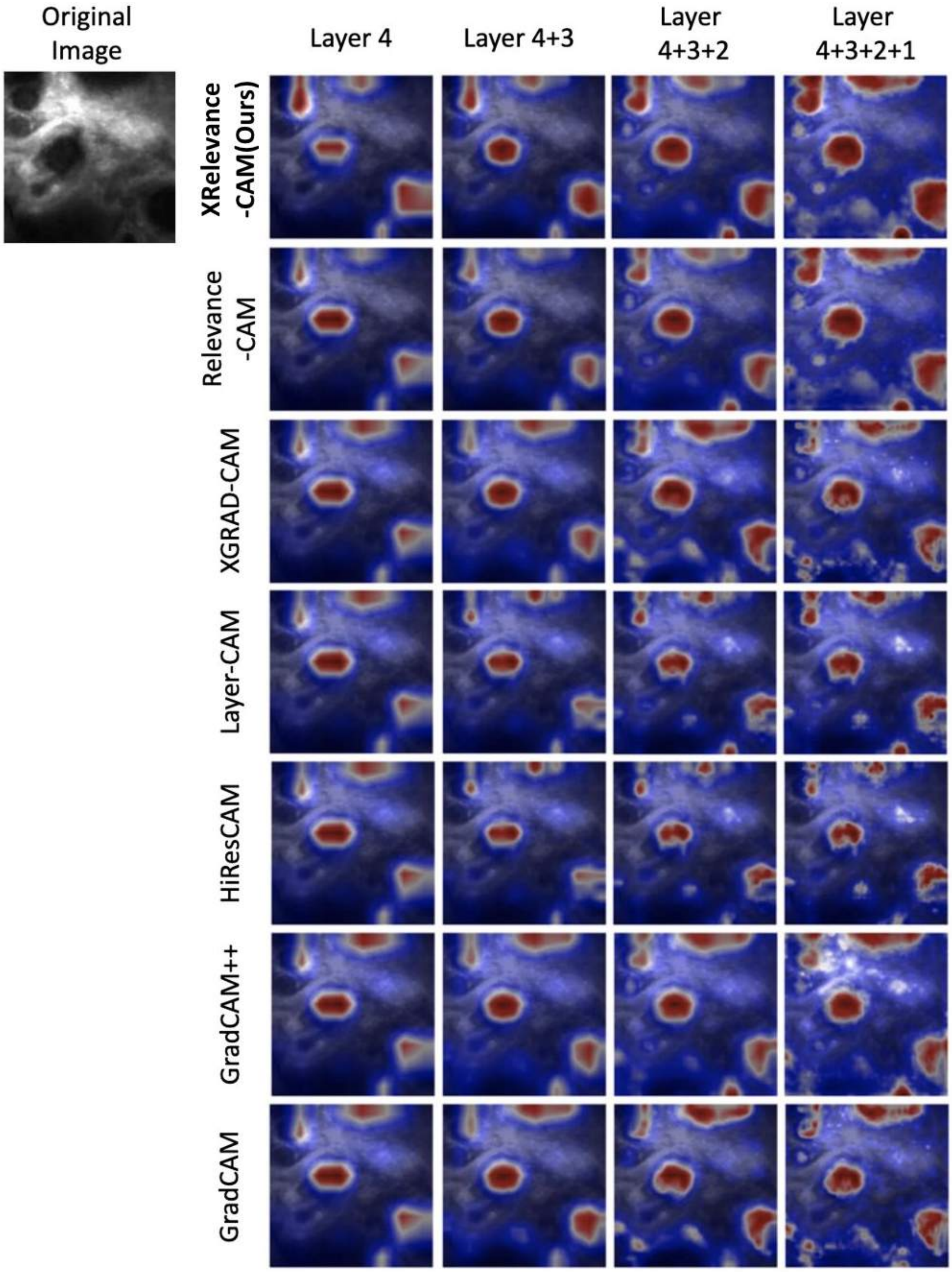


Figure 5.13: Comparison of Various Activation-driven Methods with saliency map aggregation using SK-ResNeXt as the backbone

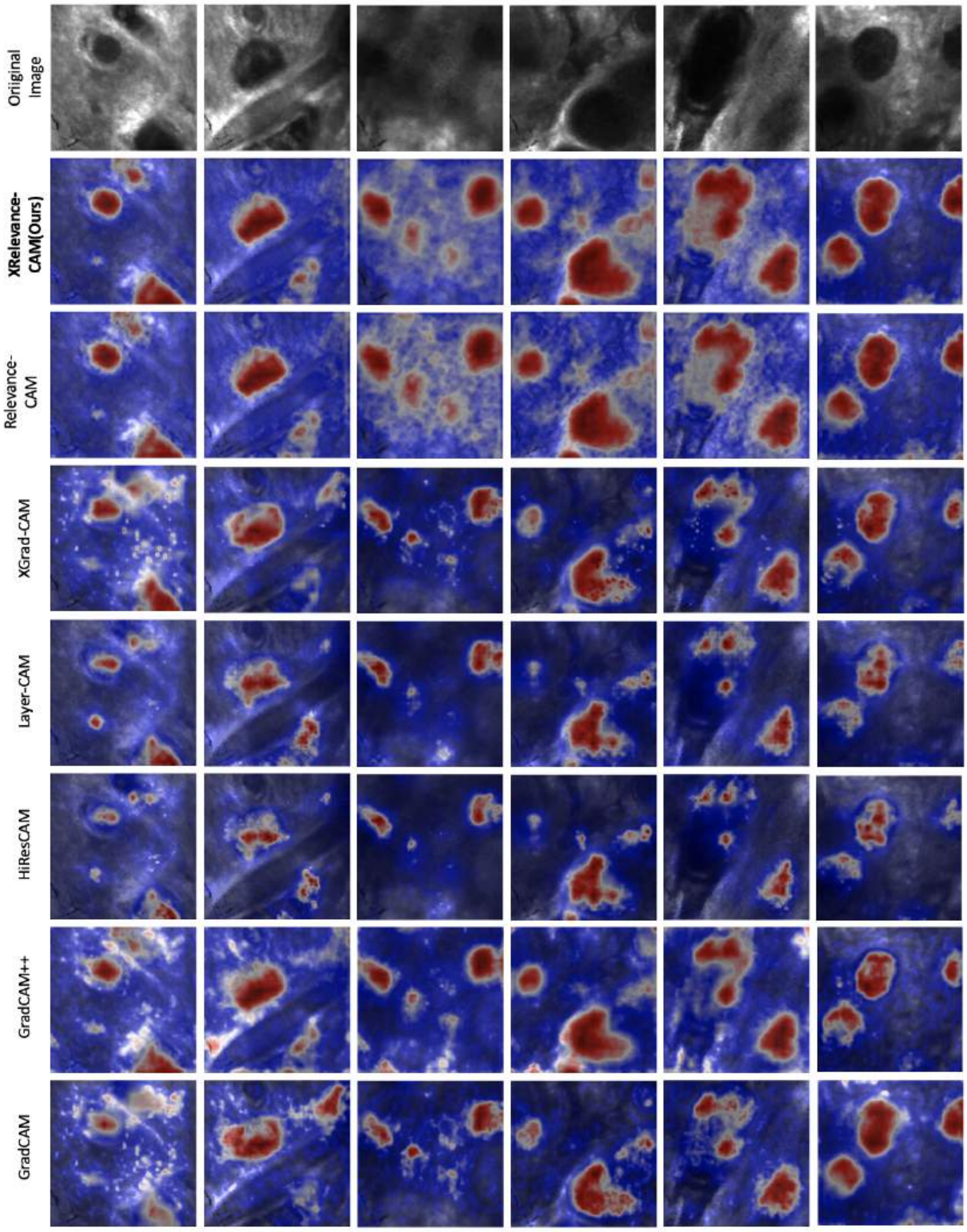


Figure 5.14: Visualization of Saliency Map Aggregation of layer 4+3+2+1 using SK-ResNeXt as the backbone.

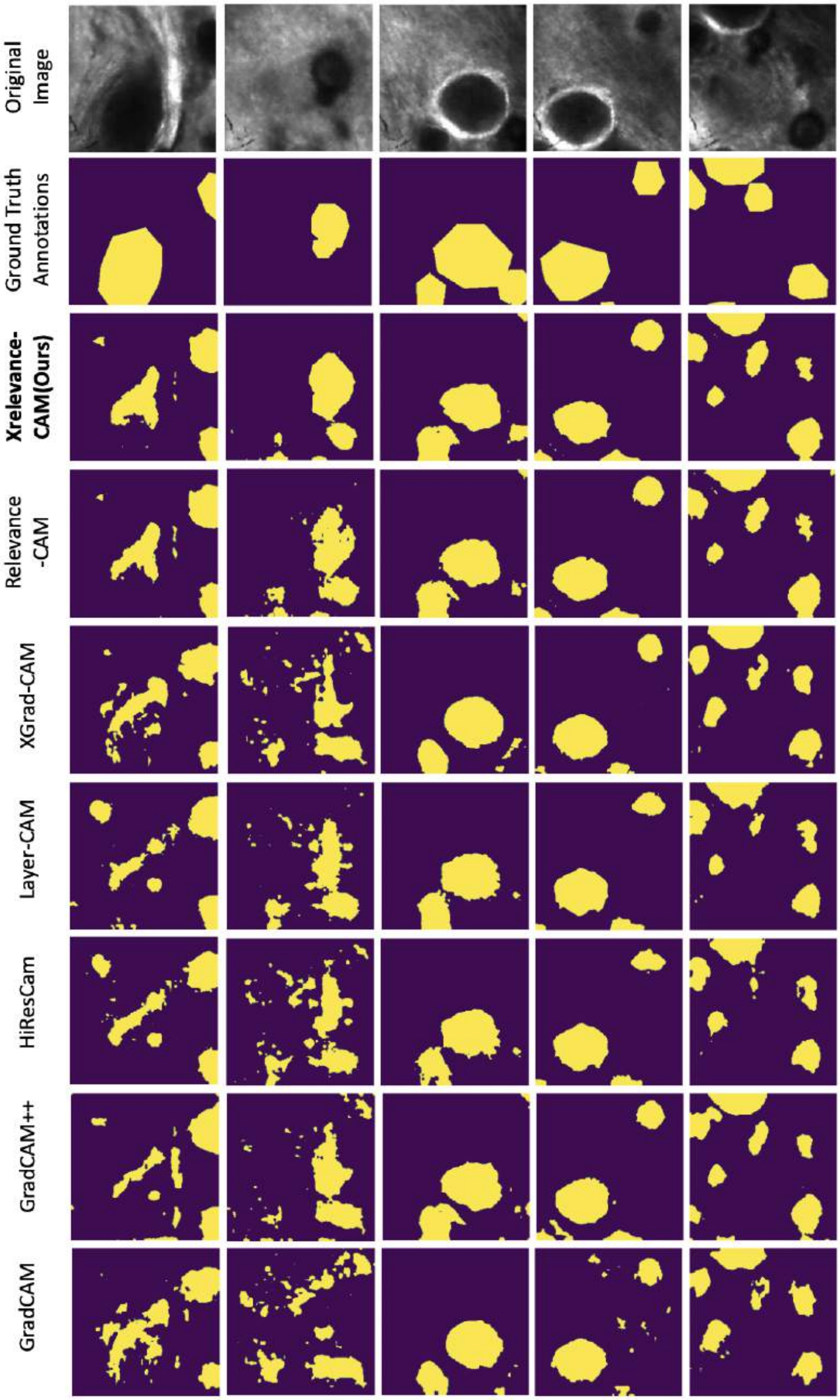


Figure 5.15: Segmentation Masks of Layer 4+3+2+1 for Various Methods

5.1.5 Section Summary

In this section, we have performed an ablation study to reveal the causality of the final performance and conclude that the selective kernel [32], our XRelevance-CAM, and saliency maps aggregation are all necessary components to maximize the performance both visually and quantitatively.

5.2 XRelevance-CAM on a Different Model

Activation-driven Methods	Layer 4	Layer 4+3	Layer 4+3+2	Layer 4+3+2+1
GradCAM [50]	23.18	25.11	23.98	21.48
GradCAM++ [8]	**24.47**	24.60	19.63	16.73
XGradCAM [15]	23.18	27.24	25.92	24.78
HiResCAM [14]	23.18	23.38	22.66	22.32
LayerCAM [26]	23.15	24.01	22.54	21.64
RelavanceCAM [30]	24.02	27.25	24.20	23.30
XRelevance-CAM(ours)	23.68	**29.20**	**27.66**	**26.10**

Table 5.4: IOU(%) Performance in Weakly-supervised Segmentation Task with Saliency Maps Aggregation, using Resnet50 [22] as the backbone.

5.2.1 Overview

To ensure the proposed attention method XRelevance-CAM generalizes to other models, we perform the same Weakly-supervised Segmentation task in ResNet50 [22]. To better reflect the overall performance of the methods, aggregated saliency map is used for evaluation.

5.2.2 Results and Analysis

All quantitative results are shown in Table 5.4. we see that XRelevance-CAM still performs the best when the saliency map of all layers are aggregated. However, one observation from the table is that including the shallowest layer does not improve the metric performance. We hypothesise that ResNet50 [22] makes final decisions by inferring some spuriously correlated features other than the discriminative cues for some inputs. In conclusion, the proposed XRelevance-CAM method works well in different models.

5.3 Sanity Check for the XRelevance-CAM

5.3.1 Overview

XRelevance-CAM is the newly introduced activation-driven method, but it requires passing the sanity check experiments to validate the legitimacy of the generated saliency maps. The sanity check experiments consist of a set of tests introduced in [2] to verify if a proposed explanation method depends on the model parameters. It is of paramount importance that the saliency maps are responsive to the model weights because the learnt weights reflect what the model discovered during the training phase. In particular, two types of weights randomization are proposed: the cascading layer-weights randomization test and the independent layer-weights randomization test [2]. If the explanation method fails to pass either of the two tests, deploying the approach in production is dangerous and not recommended.

To match the terminology in [2], we name a block of layers as a stage. Popular architectures often consist of multiple stages of the same design with different parameters. In our implementation, we randomize the network weights in a per-stage fashion. Each stage is composed of several functional layers. We only randomize the convolutional layers, the fully connected layers, and ignore the learnt weight for other types of layers, such as the batch normalisation layer [24]. All saliency maps in this section are generated using XRelevance-CAM on stage three of the resnet50 [22] architecture.

5.3.2 Cascading Layer Randomization

Experiment Settings: In this randomization test, the model's learnt parameters are progressively re-initialized with random weights stage by stage, starting from the top to the bottom (stage one). This test reveals the marginal and cumulative difference in results produced by the explanation method.

Results and Analysis: Compared to the original explanation of XRelevance-CAM to the rest in Fig 5.16, we see that after destroying the weights of stage four, more false positive features in the Meningioma input(top row) and less true positive features in the Glioblastoma input(bottom row); after successively re-initialize the parameters in stage three, no meaningful explanations are produced by XRelevance-CAM. Therefore, we can conclude that the result from XRelevance-CAM is sensitive to the model parameters.

5.3.3 Independent Layer Randomization

Experiment Settings: An alternative randomization mechanism destroys the weights one stage at a time while holding the rest of the learnt weights unchanged. The test intends to break the cross-dependence between stages by isolating the effect of each stage.

Results and Analysis: Referring to Fig 5.17, we see that the explanation is less dependent on the learnt weights in stage four compared to the rest of the stages but still reveals much more false positive cues for Meningioma input and less true positive cues for the Glioblastoma input. Furthermore, the explanations are completely broken when the parameters of the mid to shallow stages are re-initialized.

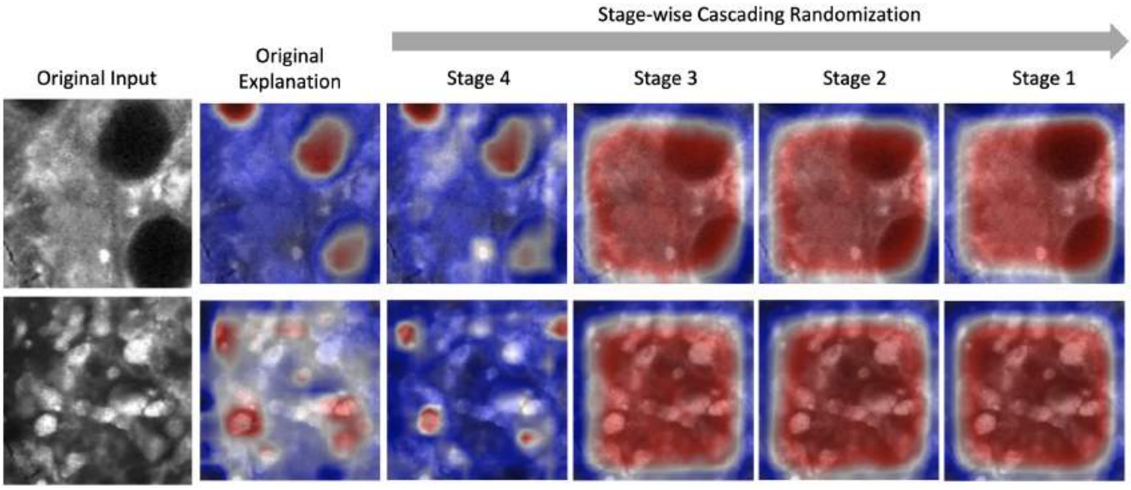


Figure 5.16: Cascading Layer-wise Randomization on resnet50 [22]: Visualizations of stage three using XRelevance-CAM, with cascading randomization from stage four to stage one. For example, the explanation under stage one is obtained from resnet50 with randomized weights in all stages. *Top:* sample Meningioma input. *Bottom:* sample Glioblastoma input. Figure style is inspired by [2].

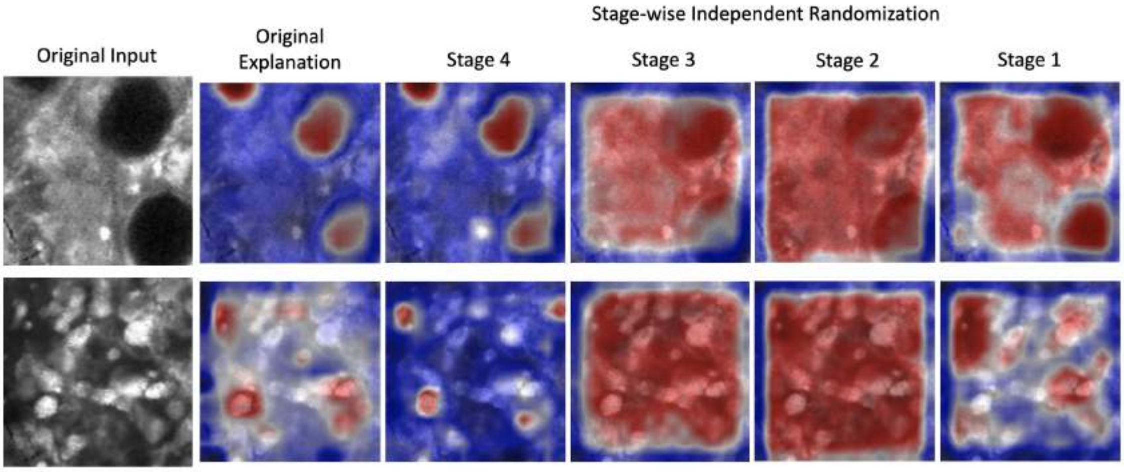


Figure 5.17: Independent Layer-wise Randomization on resnet50 [22]: Visualizations of stage three using XRelevance-CAM, with independent randomization from stage four to stage one. For instance, explanation under stage two is obtain from resnet50 with randomized weight only in stage two. *Top:* sample Meningioma input. *Bottom:* sample Glioblastoma input. Figure style is inspired by [2].

5.3.4 Section Summary

Referring to Fig 5.16 and Fig 5.17, we conclude that the saliency maps obtained from our novel XRelevance-CAM show a heavy dependence on the learnt model parameters in both cascading and independent randomization tests. In particular, the common findings in both evaluations are that the saliency map is more sensitive to the parameters in the mid-to-shallow stages and less dependent on the weights in the last stage, which has similar conclusions for Grad-CAM [50] mentioned in [2].

5.4 Layer Dropout Experiments

5.4.1 Overview

Previous experiments are evaluated based on a point estimation. However, the metric performance of explanation methods inherits the uncertainty from the stochasticity of the learnt weights for a particular model. To better account for the uncertainty, there are two ways to evaluate the average performance of an explanation method. First, we can have an ensemble of n models of the same architecture, where n is an integer. Subsequently, we can evaluate the explanation method across all ensemble members and average the final metric performance. However, this approach is inconvenient and computational expensive as it requires resources to train n networks. Second, inspired from Gal *et al* [17], we can re-train a model of interest with DropOut [55] layers and evaluate the explanation method with the DropOut [55] layers being turned on. This setup simulates the model uncertainty and uses that as a proxy to assess the average metric performance of an explanation method.

In our experiment, each layer consists of multiple blocks of the same mini neural network. We modify the SK-ResNeXt [32] architecture with one DropOut [55] layer after each block with probability of drop-out rate of 0.1. The drop-out mechanism is always turned on during the training and evaluation phase. In particular, the metric performance is obtained at the evaluation phase by passing each input ten times in the network and averaging the results. This evaluation technique is a convenient mechanism for imitating the ensemble performance of multiple models and gaining confidence in the overall quality of various explanation methods.

5.4.2 Results and Analysis

Table 5.5 shows the per-layer performance as well as the performance from aggregating all layers. With our XRelevance-CAM, we see that the average metric performance from layer one, layer two, and layer three exceeds all other methods. The average measured performance from layer four is slightly inferior compared to others. Furthermore, the metric performance by aggregation also serves as a proxy to indicate that XRelevance-CAM excels other tested methods. One remark is that as the result of applying the

DropOut[55] layer during the evaluation phase, the metric performances shown in Table 5.5 are worse than usual. Therefore, the ranking of performance is more important than the metric values. To demonstrate the drop-out effect during evaluation, we also include the metric results in Table 5.6 for the same architecture(SK-ResNeXt with DropOut layer in each block) but discard the stochasticity effect induced by the DropOut [55] layers.

Activation-driven Methods	Layer 1	Layer 2	Layer 3	Layer 4	Layer 4+3+2+1
GradCAM [50]	10.19	16.52	23.99	25.88	24.52
GradCAM++ [8]	9.97	11.11	26.42	**26.09**	25.59
XGradCAM [15]	12.15	17.55	24.81	25.87	26.58
HiResCAM [14]	9.14	15.12	21.68	25.87	25.57
LayerCAM [26]	9.22	14.85	22.41	25.99	24.49
RelavanceCAM [30]	17.87	22.08	28.08	25.98	28.57
XRelevance-CAM(ours)	**20.26**	**24.53**	**29.02**	25.96	**29.62**

Table 5.5: Per-Layer IOU(%) Performance in the Weakly-supervised Segmentation Task, using SK-ResNeXt [32] backbone. The results are obtained by running the Weakly-supervised Segmentation task ten times and averaging the IOU(%) performance. In particular, all DropOut [55] layers are turned on during the evaluation phase to resemble the stochasticity of model parameters.

Activation-driven Methods	Layer 1	Layer 2	Layer 3	Layer 4	Layer 4+3+2+1
GradCAM [50]	9.57	17.06	24.79	25.03	23.89
GradCAM++ [8]	9.23	12.14	27.03	25.29	25.54
XGradCAM [15]	11.57	18.73	25.05	25.03	25.82
HiResCAM [14]	9.51	15.28	22.41	26.00	25.21
LayerCAM [26]	9.72	16.05	22.86	25.13	24.51
RelavanceCAM [30]	12.69	26.71	35.37	31.90	35.20
XRelevance-CAM(ours)	**18.67**	**28.39**	**36.06**	**32.38**	**35.40**

Table 5.6: IOU(%) Performance in the Weakly-supervised Segmentation Task, using SK-ResNeXt [32] with DropOut [55] layers as backbone The results are obtained with no stochasticity in the model during evaluation.

5.5 Model Faithfulness Evaluation via Confidence Drop

5.5.1 Overview

In the previous sections, we evaluate the explanation quality of the human faithfulness aspect through a Weakly-supervised Segmentation task. For the sake of completeness, we use a self-evaluation metric called *confidence drop* [15] to quantitatively assess the model faithfulness perspective of the explanations on the split-out test data. That is, we want to know how well the explanation manifests the thinking process behind the decision. Formally, the confidence drop metric compares the class score of the original image and its perturbed copy defined as follows [15]

$$I_i^p = I_i \odot (1 - M_i) + \mu M_i \tag{5.3}$$

Where I_i^p is the perturbed copy of the original image I_i, M_i is the saliency map of I_i generated by an explanation method, and μ is the average intensity value of all data. Sample perturbed images are shown in Fig 5.18.

Comparable to the equation of Average Drop [8], the confidence drop metric is defined as [15]

$$\textit{Confidence Drop Metric} = \frac{1}{N}\sum_{i}^{N}\frac{S_c(I_i) - S_c(I_i^p)}{S_c(I_i)} \tag{5.4}$$

Where $S_c(\cdot)$ gives the class c score of the input and N is the number of images being examined. We choose this criterion for self-evaluation mainly because it captures the two cases that need separate evaluation through the A.D and I.C metrics shown in Section 2.7.1.

Activation-driven Methods	Layer 1	Layer 2	Layer 3	Layer 4	Layer 4+3+2+1
GradCAM [50]	3.25	29.25	**48.98**	27.32	40.81
GradCAM++ [8]	19.38	45.03	44.58	30.08	**51.19**
XGradCAM [15]	25.68	**49.46**	40.03	27.33	48.35
HiResCAM [14]	25.38	28.66	28.40	27.33	39.28
LayerCAM [26]	18.24	25.34	30.35	27.46	38.28
RelavanceCAM [30]	**42.49**	44.48	40.15	**29.71**	48.83
XRelevance-CAM(ours)	36.23	40.31	34.38	25.53	46.02

Table 5.7: Confidence Drop in %(higher the better) Analysis.

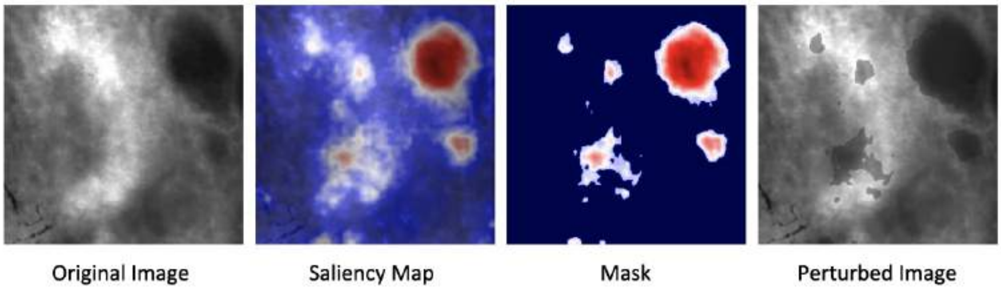


Figure 5.18: Instance of XRelevance-CAM visualisation, its mask, and the perturbed version of the original input

5.5.2 Results and Analysis

The average confidence drop results are shown in Table 5.7. We see that GradCAM++ achieves better performance in this metric. There are a few observations to be made from the table. Firstly, no methods dominate others because each CAM-based method has its comparative advantage in a particular layer. Secondly, comparing the results shown in Table 5.2 for the WSS task, we see a dilemma between the performance ranking of human faithfulness and model faithfulness, where approaches that perform well in the WSS task may perform poorly on this confidence drop metric. For instance, as demonstrated in Fig 5.3, Fig 5.4, Fig 5.5, Fig 5.6, and Fig 5.7 as well, GradCAM++ completely fail to produce meaningful explanation in layer one compare to GradCAM and thus explains its inferior result in the WSS task(8.44% vs. 16.38%), but this self-evaluation metric shows that GradCAM++ is significantly better than GradCAM(19.38% vs. 3.25%). Although our XRelevance-CAM achieves stellar performance in the WSS task, it does not stand out from the pool of tested methods in this self-evaluation metric. Our hypothesis of this contradictory outcome is based on the argument discussed in Section 2.7.3, where we believe that our XRelevance-CAM highlights fewer false positive features compared to others.

5.6 Axiom Evaluation

5.6.1 Overview

Besides the theoretical aspects behind the design of the weighting strategy (Eq 4.1), we also give an empirical study on the axiom properties for various CAM-based methods. The experiment is conducted in the split-out test data. Furthermore, analysis of the conservation and the sensitivity axioms are conducted for each layer of the SK-ResNeXt backbone, using various state-of-the-art activation-driven methods. Metric performance for each axiom evaluation is evaluated using Eq 5.5 and Eq 5.6, obtained from [15]. For fair comparisons, $S_c(\cdot)$ is the contrastive score defined in Eq 4.4 for Relevance-CAM [30] and XRelevance-CAM, and $S_c(\cdot)$ is the logit score for the rest of methods. Finally, we draw the connections between the quality of the visualisations shown in Fig 5.4, Fig 5.3, Fig 5.5, Fig 5.6, and Fig 5.7 and the axiom evaluation performance for each CAM-based method.

$$\textit{Sensitivity Axiom Metric: } \frac{1}{N}\sum_{n}^{N} \frac{\sum_k \left| S_c(A_n^l) - S_c(A_n^l \setminus A_n^{lk}) - \sum_{ij} w_{lk}^c R_{ij}(A_n^l;k))\right|}{\sum_k |S_c(A_n^l) - S_c(A_n^l \setminus A_n^{lk})|} \tag{5.5}$$

$$\textit{Conservation Axiom Metric: } \frac{1}{N}\sum_{n}^{N} \frac{\left| S_c(A_n^l) - \sum_k \sum_{ij} w_{lk}^c R_{ij}(A_n^l;k)\right|}{|S_c(A_n^l)|} \tag{5.6}$$

Where A_n^l is the activations of layer l for image n, A_n^{lk} is the kth feature map activation in layer l for image n, and definition of other terms is the same as before.

5.6.2 Results and Analysis

Activation-driven Methods	Layer 1	Layer 2	Layer 3	Layer 4	Per-Layer Average
GradCAM [50]	1.000	0.999	0.999	0.999	0.999
GradCAM++ [8]	2.912	2.09	0.925	**0.994**	1.73
XGradCAM [15]	0.998	0.994	0.994	0.999	0.996
HiResCAM [14]	0.998	0.994	0.994	0.999	0.995
LayerCAM [26]	**0.904**	0.977	0.992	0.998	**0.968**
RelavanceCAM [30]	1.01	0.93	0.82	2.40	1.29
XRelevance-CAM(ours)	0.98	**0.91**	**0.79**	1.41	1.02

Table 5.8: Metric Evaluation(Lower the better) for the Sensitivity Axiom in the split-out test data

Activation-driven Methods	Layer 1	Layer 2	Layer 3	Layer 4	Per-Layer Average
GradCAM [50]	1.261	0.884	0.6078	0.00179	0.689
GradCAM++ [8]	460.11	628.16	105.31	5.484	299.77
XGradCAM [15]	0.981	0.937	1.873	0.0018	0.948
HiResCAM [14]	0.981	0.938	1.872	0.0018	0.948
LayerCAM [26]	15.136	5.619	3.434	0.143	6.083
RelavanceCAM [30]	0.56	0.12	0.13	1.30	0.528
XRelevance-CAM(ours)	**0.02**	**0.01**	**0.0089**	$\mathbf{1.575 \times 10^{-7}}$	**0.0097**

Table 5.9: Metric Evaluation(Lower the better) for the Axiom-based Conservation in the split-out test data

Findings for the Axiom-based Conservation

Evaluation results of the axiom-based conservation property are reported in Table 5.9 with the following observations.

1. Deepest layer usually has the most promising metric performance. Among all the methods, XRelevance-CAM achieves nearly perfect results in layer four, but the performance gradually degrades as we measure the metric results in the shallow layers.

2. Compare to the other axiom-driven method XGradCAM, XRelevance-CAM surpasses its metrics performance across all layers with a minimum of 50 folds of improvement in layer one and up to $10,000$ folds of improvement in layer 4.

3. GradCAM++ and LayerCAM demonstrate catastrophic failure in complying the axiom-based conservation property, which provides another perspective on the poor qualitative results illustrated in Fig 5.3 and Fig 5.4.

4. As mentioned in [15], XGradCAM only works well in the deep layers, and the theoretical framework falls apart when applying the estimated solution in the shallow layers. Theoretically, XRelevance-CAM should have perfect results across all layers. However, due to the division steps with ϵ during the propagation, small errors accumulate to the first layer.

5. We see that the metrics results of the XRelevance-CAM outperform its close competitor RelevanceCAM throughout all layers. Furthermore, the axiomatic evaluation results exhibit a positive correlation with the qualitative results obtained in the weakly-supervised segmentation task().

Findings for the Sensitivity Axiom

Evaluation results of the sensitivity axiom are reported in Table 5.8 with the following observations.

1. Comparing the metric results between the Relevance-CAM and XRelevance-CAM for each layer, we see that XRelevance-CAM still performs marginally better even though the solution is a rough estimate.
2. Due to the estimation error, the magnitude of improvement demonstrated in the axiom-based conservation property using XRelevance-CAM does not exhibit here, which might indicate that the $\Psi(\cdot)$ term in Eq 4.12 is a significant error factor. In other words, the upper bound of the error is large.
3. In contrary to the axiom-based conservation property, the metric performance of the sensitivity axiom is less correlated with visual performance in Fig 5.4, Fig 5.3, Fig 5.5, Fig 5.6, and Fig 5.7.

5.7 Chapter Summary

We have done a series of experiments, including an ablation study of the proposed framework to demonstrate the contribution of each component; a set of sanity check evaluation tests to verify the newly proposed XRelevance-CAM is capable of generating trustworthy explanations; a layer dropout experiment is performed to show that the supreme performance of the XRelevance-CAM(ours) in the WSS task is not accidental. Finally, for the sake of completeness, an experiment for model faithfulness is also performed to demonstrate the dilemma of evaluating human faithfulness and model faithfulness. Lastly, an empirical analysis of the sensitivity axiom and the axiom-based conservation property is done to reveal the practical aspect of the theory behind our XRelevance-CAM.

Chapter 6

Conclusion and Future Work

In conclusion, we summarise the most crucial finding during our research and present the potential direction for further improvements.

6.1 Summary

This project proposes a methodology capable of generating human-faithful explanations and identifying the positive contributing cues behind the classification decision. We collect all the major discoveries below:

1. **Feature Representation by Transfer Learning**: Transfer learning is a standard practice to begin a new task because it helps mitigate the overfitting of the training data and enables using models with larger capacities. We emphasise the importance of transfer learning in this project or XAI in general because it is difficult for explanation methods to identify true positive features behind the model's decision when it learns from confounded high-level representations. Qualitative results in Section 4.1 illustrate that transfer learning helps mitigate the problem.

2. **Scale Invariance with Selective Kernel**: In medical data, disease features are often exhibited in much simpler forms compared to object classes in other generic datasets such as ImageNet [48]. However, the class discriminative features of a disease often appear at multiple scales in the same image, and ordinary architectures often apply one receptive field size of convolution to extract unique features of an object class. However, universal receptive field size cannot identify the same feature of different scales. Therefore, this limitation is resolved by the Selective Kernel architecture [32], which introduces the convolution of the same feature with multiple receptive field sizes. In our work, the Selective Kernel variant of the ResNeXt(SK-ResNeXt) [63] model is used as the main backbone for all explanation methods. However, theoretically, the Selective Kernel module can be integrated into any existing architecture for representation enhancement. In the Weakly-supervised Segmentation task(Section 5.1.2), SK-ResNeXt demonstrates significant performance

gain compared to its counterpart ResNeXt [63].

3. **Axiom-driven Relevance-CAM**: We present a novel activation-based explanation method: Axiom-driven Relevance-CAM(XRelevance-CAM). From our WSS experiment in Section 5.1.3, our XRelevance-CAM is proven to generate human-faithful explanations through qualitative assessment and demonstrates compelling metric improvement against other state-of-the-art CAM-based explanation methods. To show that the metric result is non-coincidental, we propose a convenient layer dropout experiment in Section 5.4 to show that our XRelevance-CAM maintains the same ranking among the tested methods. Furthermore, to confirm the theories behind our XRelevance-CAM, we demonstrate the axiom evaluation in Section 5.6 to show that the axiom-based conservation property indeed holds.

4. **Saliency Maps Aggregations**: Aggregation of saliency maps through simple average operation helps generate human-faithful explanations. Detail aggregation procedure is illustrated in Fig 4.4. The same aggregation procedure is also performed in different models(Section 5.2) and concludes that this technique works the best when the model's decision is based on the class-relevant features instead of the spuriously correlated cues.

6.2 Future Work

During the research of this project, we observe that all aspects of AI are significant to have high-quality explanations, including feature representations, architecture design, propagation method, and more novel weighting formulation. Furthermore, developing a metric that accounts for the two aspects of faithfulness, model faithfulness and human faithfulness, is strongly desired for researchers to have a consensus definition of the explanation quality and thereby evaluate explanation methods in a more efficient way.

6.3 Ethical Considerations

The main focus of this project is to develop a framework that explains the model's blackbox decision of tissue characterisation. The data are published by [33]. From the paper, we assure that in the data collection process, there is no conflict of interest, informed consent from the human participants, and the ethical standards declared by Helsinki is complied. Furthermore, no sensitive information is disclosed to the public during the development of this project.

Bibliography

[1] Glioblastoma multiforme gbm. *Glioblastoma multiforme - Brain Tumour Research.*

[2] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian J. Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. *CoRR*, abs/1810.03292, 2018.

[3] Abien Fred Agarap. Deep learning using rectified linear units (relu). *CoRR*, abs/1803.08375, 2018.

[4] Marco Ancona, Enea Ceolini, A. Cengiz Öztireli, and Markus H. Gross. A unified view of gradient-based attribution methods for deep neural networks. *CoRR*, abs/1711.06104, 2017.

[5] 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.

[6] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014.

[7] Yoshua Bengio, Aaron C. Courville, and Pascal Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538, 2012.

[8] Aditya Chattopadhyay, Anirban Sarkar, Prantik Howlader, and Vineeth N. Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. *CoRR*, abs/1710.11063, 2017.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *CoRR*, abs/2002.05709, 2020.

[10] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. *CoRR*, abs/2006.10029, 2020.

[11] Phillip Chlap, Hang Min, Nym Vandenberg, Jason Dowling, Lois Holloway, and Annette Haworth. A review of medical image data augmentation techniques for deep learning applications. *Journal of Medical Imaging and Radiation Oncology*, 65(5):545–563, 2021.

[12] Ekin Dogus Cubuk, Barret Zoph, Dandelion Mané, Vijay Vasudevan, and Quoc V. Le. Autoaugment: Learning augmentation policies from data. *CoRR*, abs/1805.09501, 2018.

[13] Saurabh Desai and Harish G. Ramaswamy. Ablation-cam: Visual explanations for deep convolutional network via gradient-free localization. *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2020.

[14] Rachel Lea Draelos and Lawrence Carin. Use hirescam instead of grad-cam for faithful explanations of convolutional neural networks, 2020.

[15] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. Axiom-based grad-cam: Towards accurate visualization and explanation of cnns. *CoRR*, abs/2008.02312, 2020.

[16] Hiroshi Fukui, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. Attention branch network: Learning of attention mechanism for visual explanation. *CoRR*, abs/1812.10025, 2018.

[17] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning, 2015.

[18] Jacob Gildenblat and contributors. Pytorch library for cam methods. https://github.com/jacobgil/pytorch-grad-cam, 2021.

[19] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[20] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[21] Jindong Gu, Yinchong Yang, and Volker Tresp. Understanding individual decisions of cnns via contrastive backpropagation. *CoRR*, abs/1812.02100, 2018.

[22] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[23] Kaixu Huang, Fanman Meng, Hongliang Li, Shuai Chen, Qingbo Wu, and King N. Ngan. Class activation map generation by multiple level class grouping and orthogonal constraint. In *2019 Digital Image Computing: Techniques and Applications, DICTA 2019, Perth, Australia, December 2-4, 2019*, pages 1–6. IEEE, 2019.

[24] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[25] Brian Kenji Iwana, Ryohei Kuroki, and Seiichi Uchida. Explaining convolutional neural networks using softmax gradient layer-wise relevance propagation. *CoRR*, abs/1908.04351, 2019.

[26] Peng-Tao Jiang, Chang-Bin Zhang, Qibin Hou, Ming-Ming Cheng, and Yunchao Wei. Layercam: Exploring hierarchical class activation maps for localization. *IEEE Transactions on Image Processing*, 30:5875–5888, 2021.

[27] Hyungsik Jung and Youngrock Oh. LIFT-CAM: towards better explanations for class activation mapping. *CoRR*, abs/2102.05228, 2021.

[28] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *CoRR*, abs/2004.11362, 2020.

[29] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014.

[30] Jeong Ryong Lee, Sewon Kim, Inyong Park, Taejoon Eo, and Dosik Hwang. Relevance-cam: Your model already knows where to look. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[31] Kwang Hee Lee, Chaewon Park, Junghyun Oh, and Nojun Kwak. LFI-CAM: learning feature importance for better visual explanation. *CoRR*, abs/2105.00937, 2021.

[32] Xiang Li, Wenhai Wang, Xiaolin Hu, and Jian Yang. Selective kernel networks. *CoRR*, abs/1903.06586, 2019.

[33] Yachun Li, Patra Charalampaki, Yong Liu, Guang-Zhong Yang, and Stamatia Giannarou. Context aware decision support in neurosurgical oncology based on an efficient classification of endomicroscopic data. *International Journal of Computer Assisted Radiology and Surgery*, 13(8):1187–1199, 2018.

[34] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network, 2013.

[35] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014.

[36] Aravindh Mahendran and Andrea Vedaldi. Salient deconvolutional networks. In *ECCV*, 2016.

[37] Fanman Meng, Kaixu Huang, Hongliang Li, and Qingbo Wu. Class activation map generation by representative class selection and multi-layer feature fusion. *CoRR*, abs/1901.07683, 2019.

[38] Grégoire Montavon, Sebastian Bach, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *CoRR*, abs/1512.02479, 2015.

[39] Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. Layer-wise relevance propagation: An overview. *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*, page 193–209, 2019.

[40] Rakshit Naidu and Joy Michael. SS-CAM: smoothed score-cam for sharper visual feature localization. *CoRR*, abs/2006.14255, 2020.

[41] Daniel Omeiza, Skyler Speakman, Celia Cintas, and Komminist Weldemariam. Smooth grad-cam++: An enhanced inference level visualization technique for deep convolutional neural network models. *CoRR*, abs/1908.01224, 2019.

[42] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Is object localization for free? - weakly-supervised learning with convolutional neural networks. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[43] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[44] Vitali Petsiuk, Abir Das, and Kate Saenko. RISE: randomized input sampling for explanation of black-box models. *CoRR*, abs/1806.07421, 2018.

[45] Yao Qin, Konstantinos Kamnitsas, Siddharth Ancha, Jay Nanavati, Garrison W. Cottrell, Antonio Criminisi, and Aditya V. Nori. Autofocus layer for semantic segmentation. *CoRR*, abs/1805.08403, 2018.

[46] Marco Túlio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should I trust you?": Explaining the predictions of any classifier. *CoRR*, abs/1602.04938, 2016.

[47] M. Robnik-Sikonja and I. Kononenko. Explaining classifications for individual instances. *IEEE Transactions on Knowledge and Data Engineering*, 20(5):589–600, 2008.

[48] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014.

[49] Sam Sattarzadeh, Mahesh Sudhakar, Konstantinos N. Plataniotis, Jongseong Jang, Yeonjeong Jeong, and Hyunwoo Kim. Integrated grad-cam: Sensitivity-aware visual explanation of deep convolutional networks via integrated gradient-based scoring. *CoRR*, abs/2102.07805, 2021.

[50] Ramprasaath R. Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. Grad-cam: Why did you say that? visual explanations from deep networks via gradient-based localization. *CoRR*, abs/1610.02391, 2016.

[51] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. *CoRR*, abs/1704.02685, 2017.

[52] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2013.

[53] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda B. Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *CoRR*, abs/1706.03825, 2017.

[54] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin A. Riedmiller. Striving for simplicity: The all convolutional net. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015.

[55] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[56] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. *CoRR*, abs/1703.01365, 2017.

[57] Jigisha P Thakkar, Vikram C Prabhu, and Pier Paolo Peruzzi. Glioblastoma multiforme.

[58] Erico Tjoa and Cuntai Guan. A survey on explainable artificial intelligence (XAI): towards medical XAI. *CoRR*, abs/1907.07374, 2019.

[59] Jeffrey I. Traylor and John S. Kuo. Meningiomas.

[60] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.

[61] Haofan Wang, Mengnan Du, Fan Yang, and Zijian Zhang. Score-cam: Improved visual explanations via score-weighted class activation mapping. *CoRR*, abs/1910.01279, 2019.

[62] Ross Wightman. Pytorch image models. https://github.com/rwightman/pytorch-image-models, 2019.

[63] Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *CoRR*, abs/1611.05431, 2016.

[64] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. *CoRR*, abs/1311.2901, 2013.

[65] Matthew D. Zeiler, Dilip Krishnan, Graham W. Taylor, and Rob Fergus. Deconvolutional networks. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2010.

[66] Zhibo Zhang, Jongseong Jang, Chiheb Trabelsi, Ruiwen Li, Scott Sanner, Yeonjeong Jeong, and Dongsub Shim. Excon: Explanation-driven supervised contrastive learning for image classification. *CoRR*, abs/2111.14271, 2021.

[67] Bolei Zhou, Aditya Khosla, Àgata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. *CoRR*, abs/1512.04150, 2015.

[68] Luisa M. Zintgraf, Taco S. Cohen, Tameem Adel, and Max Welling. Visualizing deep neural network decisions: Prediction difference analysis. *CoRR*, abs/1702.04595, 2017.