
Improving Interpretability in Medical Imaging Diagnosis using Adversarial Training

Andrei Margelou¹, Nikola Simidjievski¹, Mateja Jamnik¹, Adrian Weller^{1,2}

¹University of Cambridge, UK

²The Alan Turing Institute, UK

[am2770, ns779, mj201, aw665]@cam.ac.uk

Abstract

We investigate the influence of adversarial training on the interpretability of convolutional neural networks (CNNs), specifically applied to diagnosing skin cancer. We show that gradient-based saliency maps of adversarially trained CNNs are significantly sharper and more visually coherent than those of standardly trained CNNs. Furthermore, we show that adversarially trained networks highlight regions with significant color variation within the lesion, a common characteristic of melanoma. We find that fine-tuning a robust network with a small learning rate further improves saliency maps' sharpness. Lastly, we provide preliminary work suggesting that robustifying the first layers to extract robust low-level features leads to visually coherent explanations.

1 Introduction

As diagnostic errors contribute to approximately 10% of patients' deaths and up to 17% of hospital adverse events [12], there is great hope for machine learning to help in medical diagnosis [11]. However, lack of transparency or inaccurate explanations for model predictions may lead to suboptimal or even harmful treatments [2].

We explore the extent to which adversarial training [10] could improve the interpretability of deep neural networks, specifically in detecting skin cancer. Adversarially trained classifiers learn robust features and can synthesize realistic images by iteratively updating the pixels in order to maximize the score of a target class [4, 8, 22]. This holds even when the classifier has low robustness to adversarial attacks [1]. Regarding adversarially trained classifiers' interpretability, Zhang and Zhu [25] showed that they provide more visually coherent SmoothGrad saliency maps [17] and more shape-based saliency maps on natural images in general.

However, these claims arise from experiments on natural images, which may differ significantly from medical images. We suspect that distinguishing medical pathologies involves a careful inspection of small changes in textures across a particular region, to a greater extent than natural images. Moreover, there are typically far fewer images in medical datasets. Hence, we want to investigate if adversarial training can also be used on smaller size medical imaging datasets to improve interpretability.

Contributions: To the best of our knowledge, we present the first study on using adversarial training to improve interpretability in a medical imaging machine learning setting. We show that adversarially trained models have sharper and more visually coherent gradient-based saliency maps, highlighting the changes in the lesions' color, a common characteristic of melanoma. Further, we present initial findings suggesting that standard fine-tuning an already adversarially trained model further sharpens the saliency maps.

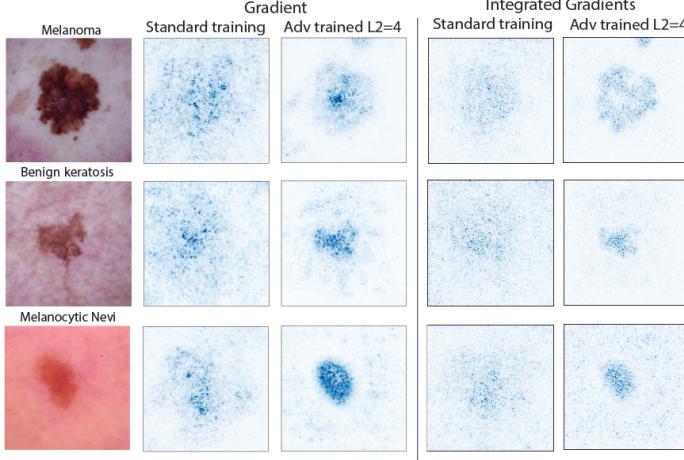


Figure 1: Saliency maps (obtained using Gradient and Integrated Gradients (IG) with Uniform baseline) on a standard and a robust model. Notice that the robust model has significantly sharper saliency maps, and it highlights the skin lesion predominantly. Furthermore, notice that the saliency map using IG of the robust model (last column of the first row) highlights the color changes in the lesion, common to melanoma.

2 Method and Experimental Setup

Adversarial training: Classifiers that achieve high accuracy on adversarially perturbed inputs [20] are called adversarially robust (or simply **robust**). A common way to obtain robust classifiers is using adversarial training [10], which approximates the solution to a min-max objective function. To apply adversarial training, at every training iteration, the training samples are augmented with an adversarial perturbation. Typically the set of allowed perturbations, Δ , is a norm ball $\Delta = \{\delta : \|\delta\| \leq \epsilon\}$ (using e.g., l_1 , l_2 , or l_∞ norm), where ϵ is the maximum size of the norm.

Proposed method: We propose using adversarial training as a replacement to standard training *after* all training hyper-parameters have been selected in a standard way. Thus, using adversarial training does not require changing the model architecture or hyper-parameters and can be added later if improved interpretability is desired.

1. *Perform hyper-parameter tuning using standard training.*
2. *Retrain the same model from scratch using adversarial training and re-use the same hyper-parameters previously found.* We suggest using a PGD adversary [10] with an l_2 norm ϵ . The adversary power ϵ should be tuned depending on the dataset, but we suggest using $\epsilon = 4$ for 50,000 image pixels (scale ϵ proportionally to the desired input size). Perform perturbation augmentation using seven steps of size $\epsilon/5$ and start from a different random point. The adversarial image is clipped in the allowed range $[0, 1]$. Note that all other training hyper-parameters are the same as found using standard training in step 1.

Experimental Setup: We perform experiments in an image classification problem, on a subset of the dermatology dataset HAM10000 [21] containing three classes: Melanoma, Melanocytic Nevi, and Benign Keratosis. We use a ResNet-18 [6] architecture. All relevant code is available at <https://github.com/margiki/Interpretability-Adversarial>. See Appendix A for training details.

Saliency maps are a common way to understand the predictions of a model by highlighting the features (pixels in case of images) deemed important to the model in making the prediction. A more important feature has a higher color intensity. All showcased saliency maps are computed on correctly-classified images. Appendix B presents in detail common saliency methods.

Evaluation: We follow previous work in interpretability [17, 19] and *qualitatively* assess the *visual coherence* of saliency maps of robust and standard models. Following Smilkov et al. [17], we use *visual coherence* to mean that salient areas primarily highlight the object of interest, rather than the background. Evaluating models’ interpretations remains an active area of research. Current quantitative metrics [7, 13, 23] seem unsuitable here since they are based on performing various input perturbations (which make less sense for adversarially trained models).

3 Results and Discussion

Robust models have sharper and more visually coherent saliency maps. Figure 1 shows that robust models provide sharper and more visually coherent gradient-based saliency maps using Gradient [19] and downstream methods such as Integrated Gradient (IG) [19]. Notice that the high saliency values are attributed predominantly inside the skin lesion, which is the object of interest. See Appendix B for more figures, including positive results using perturbation-based attribution methods.

Robust models may learn robust features from medical images. In the results on melanoma using Integrated Gradients from Figure 1 (last two columns of the first row), notice that the saliency map of the standard model does not resemble the change in variation from the lesion. However, the robust model (last column of the first row) predominantly highlights regions with significant color variation, a common characteristic of melanoma. This indicates that robust models may indeed learn to make predictions based on robust features [8].



Figure 2: Fail-cases of the saliency maps of robust models. Notice that artifacts such as dark regions (lesion 1) or hairs (lesion 2) can be highlighted as relevant, or the saliency map can be noisy (lesion 3).

Limitations: Figure 2 shows that the saliency maps of robust models can highlight image artifacts such as dark regions or hairs, as well as being noisy. On further investigation, we found that the maximum perturbation size ϵ used for training greatly influences the final saliency map. This can be caused partially because saliency methods are not stable to training noise [17] (empirical examples available in Appendix C.1 and C.2). However, our early results indicate the possibility of an image-specific optimal training perturbation ϵ for providing the sharpest saliency maps (more details about this hypothesis in Appendix C.3).

It will be interesting to explore how well our claims generalize to other types of medical images (e.g., X-rays, MRI scans), datasets, and network architectures.

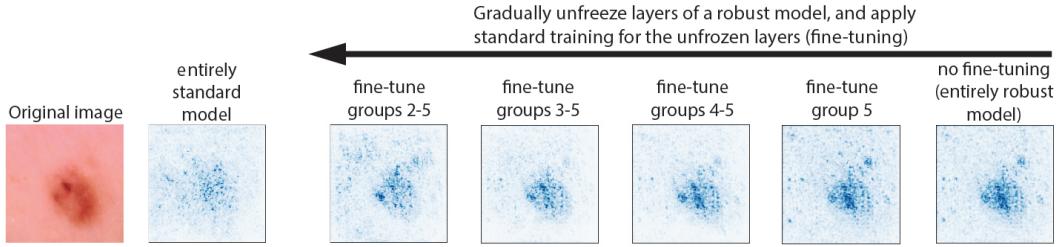


Figure 3: Gradient saliency maps when gradually unfreezing the layers of a robust network, and performing standard fine-tuning. The last column shows the saliency maps of a robust model. Columns 3-7 represent new models obtained by fine-tuning a robust model (from the last column) with a small learning rate. By fine-tuning the last layers, we essentially enable them to combine the robust features extracted by the first layers. Notice how fine-tuning the last layers reduces the noise in the saliency maps.

Conclusion and Future Work: We presented a preliminary investigation on a medical imaging dataset, suggesting that gradient-based saliency maps can become sharper and more visually coherent using adversarial training. This way, robust models attribute high saliency values predominantly to the lesion and its color change, common to melanoma.

This work suggests several directions for future research. We intend to investigate which layers are important to ‘robustify’ to obtain visually coherent explanations. Figure 3 shows early evidence that fine-tuning an already robust network further sharpens the saliency maps - suggesting that having robust first layers to extract robust low-level features leads to visually coherent explanations (see training details and more figures in Appendix D). Finally, we look forward to seeing how our findings generalize to other datasets and network architectures.

Broader Impact

Convolutional Neural Networks are applied in a number of classification tasks involving Medical Images. As diagnostic errors contribute to approximately 10% of patients' deaths and up to 17% of hospital adverse events [12], having a reliable diagnosis is critical. Our research can help mitigate misdiagnosis by enabling doctors to better individual decisions of the model and providing trustworthy explanations.

The potential risks of our method are associated with the doctors becoming overconfident in their decision, thus not seek a second opinion and lead to misdiagnosis. To address this issue, we clearly mentioned the limitations of our work and made specific that it can capture artifacts (see Figure 2 and common fail-cases in Appendix C). The proposed method is not ready to be used in a real-world scenario. We encourage research in understanding why saliency methods are brittle to training noise as well as understanding why saliency method can capture artifacts (e.g., dark regions in the skin lesion). Furthermore, the used dataset contains images of only one skin type; thus, investigating more diverse datasets is required to generalize the method's effectiveness on other skin types.

As a near-term impact, we envision researchers and practitioners using this method for "bug-fixing" during model experimentation to understand fail-cases better and improve their models.

To advance the collective understanding of the limitations of using adversarial training to improve interpretability, we encourage research in understanding the explanations of robust models in real-world scenarios in Medicine, other high-stake domains (e.g., autonomous driving). We also acknowledge the importance of gathering more diverse datasets so that progress in AI benefits all people.

Acknowledgments and Disclosure of Funding

Andrei Margeloiu acknowledges support from the Cambridge ESRC Doctoral Training Partnership. Nikola Simidjevski and Mateja Jamnik acknowledge support from The Mark Foundation for Cancer Research and Cancer Research UK Cambridge Centre [C9685/A25177]. Adrian Weller acknowledges support from the David MacKay Newton research fellowship at Darwin College, The Alan Turing Institute under EPSRC grant EP/N510129/1 and U/B/000074, and the Leverhulme Trust via CFI.

References

- [1] Gunjan Aggarwal, Abhishek Sinha, Nupur Kumari, and Mayank Singh. On the benefits of models with perceptually-aligned gradients. *International Conference on Learning Representations (ICLR) 2020 Workshop: Towards Trustworthy ML*, 2020.
- [2] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115, 2020.
- [3] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [4] Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations, 2020. URL <https://openreview.net/forum?id=rygvFyrKwH>.
- [5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.
- [6] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [7] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems*, pages 9734–9745, 2019.

- [8] Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *NeurIPS 2019 : Thirty-third Conference on Neural Information Processing Systems*, pages 125–136, 2019.
- [9] Diederik P. Kingma and Jimmy Lei Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.
- [10] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations (ICLR)*, 2018. URL <https://openreview.net/forum?id=rJzIBfZAb>.
- [11] Travis B Murdoch and Allan S Detsky. The inevitable application of big data to health care. *Jama*, 309(13):1351–1352, 2013.
- [12] Engineering National Academies of Sciences and Medicine. *Improving diagnosis in health care*. National Academies Press, 2015.
- [13] Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.
- [14] Rory Sayres, Ankur Taly, Ehsan Rahimy, Katy Blumer, David Coz, Naama Hammel, Jonathan Krause, Arunachalam Narayanaswamy, Zahra Rastegar, Derek Wu, et al. Using a deep learning algorithm and integrated gradients explanation to assist grading for diabetic retinopathy. *Ophthalmology*, 126(4):552–564, 2019.
- [15] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017.
- [16] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *Workshop at International Conference on Learning Representations (ICLR)*, 2014.
- [17] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*, 2017.
- [18] Pascal Sturmfels, Scott Lundberg, and Su-In Lee. Visualizing the impact of feature attribution baselines. *Distill*, 2020. doi: 10.23915/distill.00022. <https://distill.pub/2020/attribution-baselines>.
- [19] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR.org, 2017.
- [20] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.
- [21] Philipp Tschandl, Cliff Rosendahl, and Harald Kittler. The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Scientific data*, 5:180161, 2018.
- [22] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In *International Conference on Learning Representations (ICLR)*, 2019. URL <https://openreview.net/forum?id=SyxAb30cY7>.
- [23] Mengjiao Yang and Been Kim. Benchmarking attribution methods with relative feature importance. *arXiv preprint arXiv:1907.09701*, 2019.
- [24] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.
- [25] Tianyuan Zhang and Zhanxing Zhu. Interpreting adversarially trained convolutional neural networks. In *International Conference on Machine Learning (ICML)*, pages 7502–7511, 2019.

Appendix A Experimental Setup and Training details

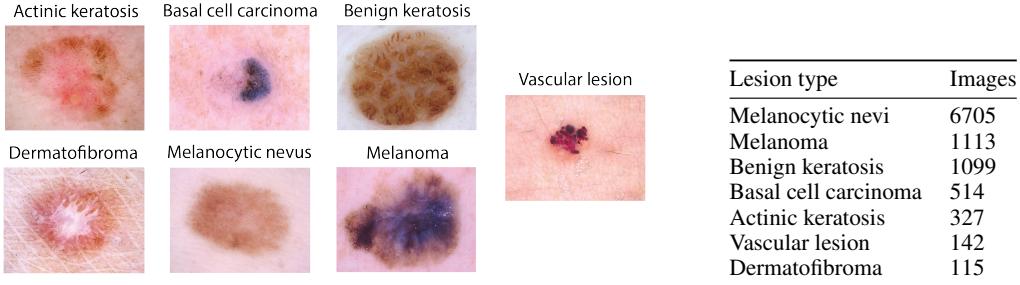


Figure 4: Overview of the dataset. Left: Images from each class of the dataset. Right: Dataset distribution.

Dataset: We use the dermatology dataset HAM10000 [21] which contains 10,015 images presenting seven lesion types as shown in Figure 4 with the number of instances of each. Adversarial training studies are not typically conducted on datasets with high-class imbalance (the majority/minority ratio is 58 : 1 for this dataset). Consequently, we perform experiments on a randomly selected balanced subset of three classes: Melanocytic Nevi, Benign Keratosis, and Melanoma. We randomly split the dataset into training (2400 images: 800 for each class) and test (897 images: 299 for each class). The training set is randomly split into five folds, which are used for cross-validation. There is no overlap between training and test sets or between the folds in cross-validation.

Architecture: We use a ResNet-18 architecture [6], with the weights initialised from a model pretrained on ImageNet [3]. The last fully connected layer is replaced with one that has three outputs (i.e., one for each target class), and whose weights are initialized using Kaiming initialization [5].

Training: We use the transfer learning methodology first to train the randomly initialized last layer, and then we fine-tune the entire network. All training is performed using Adam [9] with standard parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$) and with weight decay of 5e-4. We train only the last fully connected layer for ten epochs (to avoid large gradient updates that would collapse the pre-trained weights in the first convolutional layers during fine-tuning), with a mini-batch of size 32 and learning rate $\alpha = 0.001$. The fine-tuning is performed as follows.

We fine-tune the whole network for 25 epochs, with a mini-batch of size 16 and a learning $\alpha = 0.0003$. The learning rate is decayed by a factor of 10 after 15 epochs. The fine-tuning hyperparameters were found using five fold cross-validation on the training set, through a grid-search of learning rates of $1e - 3$, $3e - 4$, $1e - 4$ and mini-batch sizes of 8, 16, 32. We selected the hyper-parameters providing the highest accuracy and trained the final model on the entire training set.

For the fine-tuning experiment (Appendix D), we randomly take 150 samples as a balanced validation dataset for early stopping, leaving 747 samples for the test dataset. The hyper-parameters and training procedure are as presented above and have two changes to accommodate for extra fine-tuning a robust network. Firstly, we use early stopping if the validation loss does not decrease for ten epochs. Secondly, we decrease the learning rate by a factor of 10 if the validation loss does not decline for five epochs.

Adversarial training: In all experiments, we obtain robust classifiers by closely following the adversarial training methodology of Madry et al. [10] by augmenting the training data with perturbations coming from a projected gradient descent (PGD) adversary. Thus, we train against a PGD adversary in the l_2 norm using seven steps, varying power ϵ (mentioned in each experiment), step size of $\epsilon/5$ and starting from a different random point. The adversarial image is clipped in the allowed range $[0, 1]$. All other hyper-parameters are the same as during standard training.

Data augmentation: The original images have size 600x450px. We crop the centre region of 450x450px and downscale to 224x224px. We apply data augmentation in order to prevent overfitting: random horizontal/vertical flips, changing the brightness, contrast, and saturation randomly by up to 0.2, and applying random affine transformation of 10° and rotation of up to 50° . The images are normalized with the mean and standard deviation of the ImageNet training set.

Appendix B Saliency Methods and Additional Results

All showcased saliency maps are computed on correctly-classified images and are visualized as single-colored heatmaps. The saliency of one pixel is the sum of the absolute values of each RGB channel. We cap the extreme value to the 99th percentile as proposed by Smilkov et al. [17], and rescale the values in [0, 1].

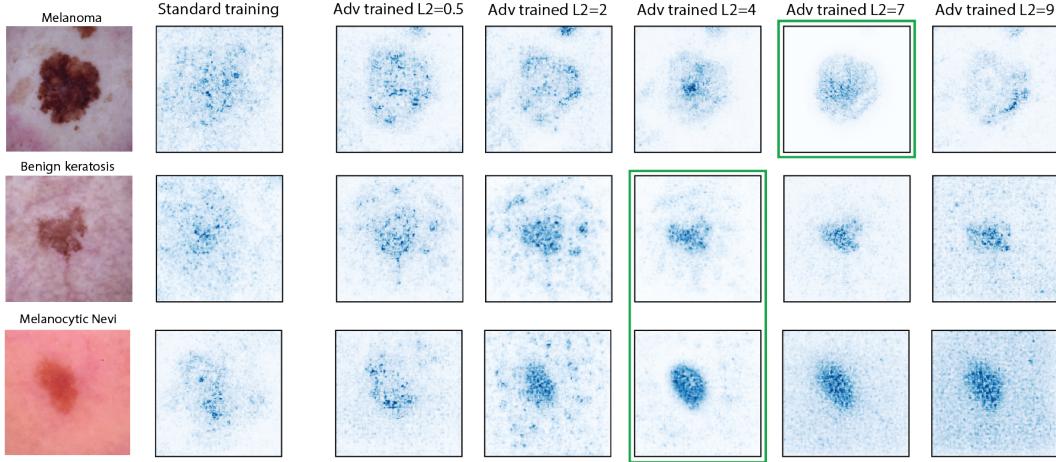


Figure 5: Gradient (also called Saliency) attribution method applied to several models. The second column shows the saliency map of a standard model. The following columns show the saliency map of robust models trained with a PGD adversary with varying l_2 norm. Notice that powerful adversaries ($\epsilon = 4, \epsilon = 7$) correlate with more perceptually-aligned gradients on the skin lesions. In green are highlighted the most coherent saliency maps to our opinion. Furthermore, notice that using a large adversary ($\epsilon = 9$) starts showing noise in the saliency map, but still retains the perceptually-aligned gradients.

B.1 Gradient

Gradient (also called Saliency) [16] is a feature attribution method that assigns importance by computing the gradient of the score of a class of interest y^c with respect to every input feature x_i . For images, a feature x_i represents one RGB channel of a pixel.

$$\frac{\delta y^c}{\delta x_i} \quad (1)$$

The saliency map is constructed by highlighting each pixel proportionally to the absolute value of its gradient (eq. 1). It is worth emphasizing that the saliency map is created for a target class c .

Figure 5 shows a visual comparison of Gradient applied on standard and robust models trained with varying l_2 norm. Notice in the second column that the saliency of a standard model is visually noisy and not very coherent with the lesion. In comparison, increasing the adversary level ($\epsilon = 0.5, \epsilon = 2, \epsilon = 4$) makes the saliency map more coherent by having perceptually-aligned gradients with the lesion. A very powerful adversary with $\epsilon = 9$ starts introducing noise.

B.2 Integrated Gradients

Integrated Gradients [19] compute the importance score by summing the gradients on images interpolated between a baseline x' and the input image x . The baseline image x' represents the absence of the input features, and we will shortly discuss several options about choosing the baseline. Thus, by integrating over the path between the baseline x' and the real input x , we obtain multiple estimates of the importance of every feature; this avoids the issue of local gradients being saturated. Formally, the importance score for feature i is computed as:

$$\phi_i^{IG}(f, x, x') = (x_i - x'_i) \times \int_{\alpha=0}^1 \frac{\delta f(x' + \alpha(x - x'))}{\delta x_i} d\alpha \quad (2)$$

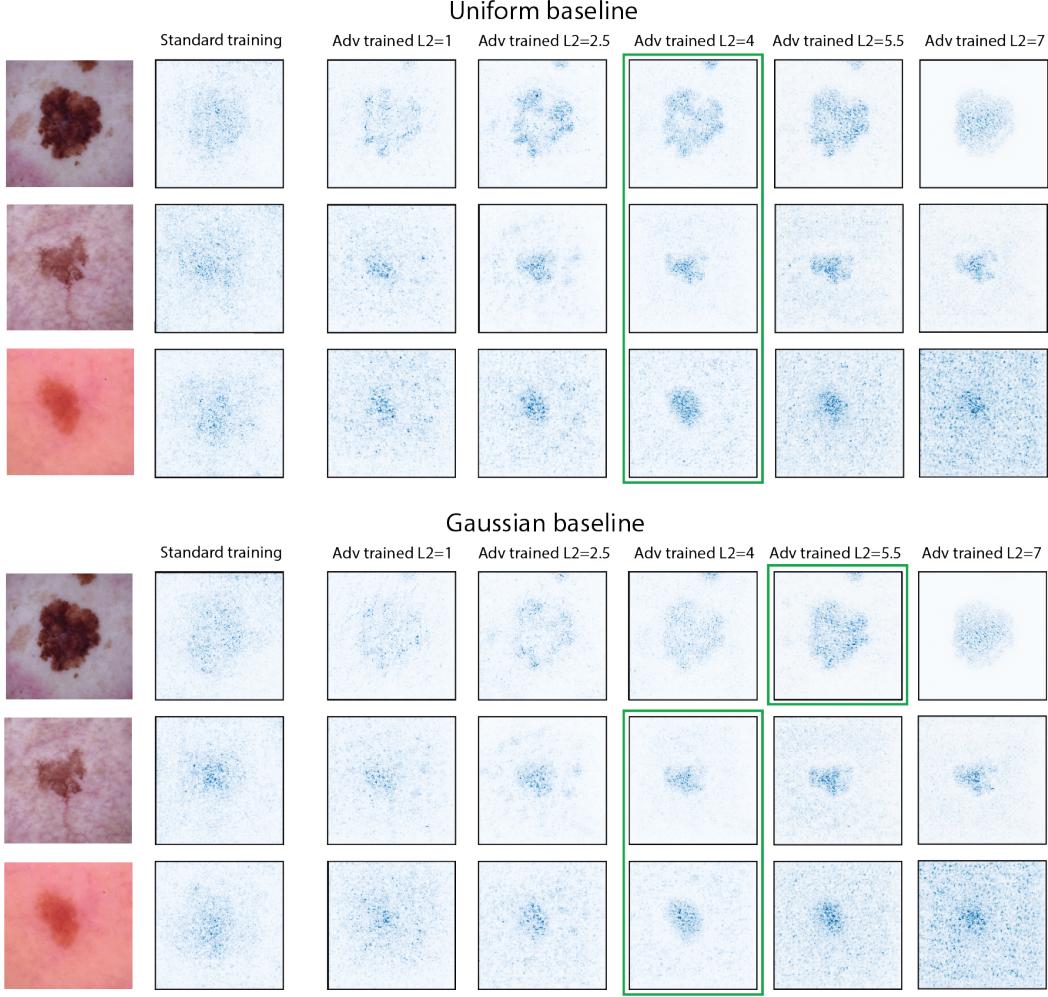


Figure 6: Integrated Gradients attribution applied on several images. The top section shows the attributions using a baseline of uniform noise, and the bottom section shows the attribution using a baseline of a Gaussian distribution centered at the original image with $\sigma = 1$. The most coherent saliency maps are highlighted in green. Notice that for both baselines, robust models have more coherent saliency maps.

where f represents the model, x_i represents one feature of the input, δ represents partial derivative, and α is part of the integral and defines the distance on the path between x and x' .

The **baseline** aims to describe the absence of the input features. Choosing the right baseline is closely related to finding a natural way of expressing the absence of the features from a dataset. With regard to images, pixels cannot be removed, and we need to define a value that represents their absence. For example, in diabetic retinopathy, a common baseline is a black image [14].

For our skin cancer dataset, a **black baseline** is not appropriate because the black color is strongly associated with melanoma. Thus the sensitivity of the lesion's pixels would be minimal because the baseline is very similar to the lesion color. With regards to the equation 2, a black baseline makes the difference between baseline and input $(x - x') \rightarrow 0$, thus attributing zero importance to the lesion.

Two alternative baselines are uniform random and Gaussian [18]:

- The **uniform baseline** represents an image where each pixel is drawn from a uniform distribution between $[0, 1]$. Intuitively, this signifies a random image that looks like noise.

- The **Gaussian baseline** is inspired by [17] and is a sample from a Gaussian distribution centered at the original image with a standard deviation of $\sigma = 1$. Intuitively, it can be thought of adding Gaussian noise to the input image.

Figure 6 shows the attribution with a uniform and a Gaussian baseline. Notice that robust models assign importance predominantly within the boundary of the lesion; this hints that robust models make the prediction mainly by looking at the lesion.

B.3 Occlusion

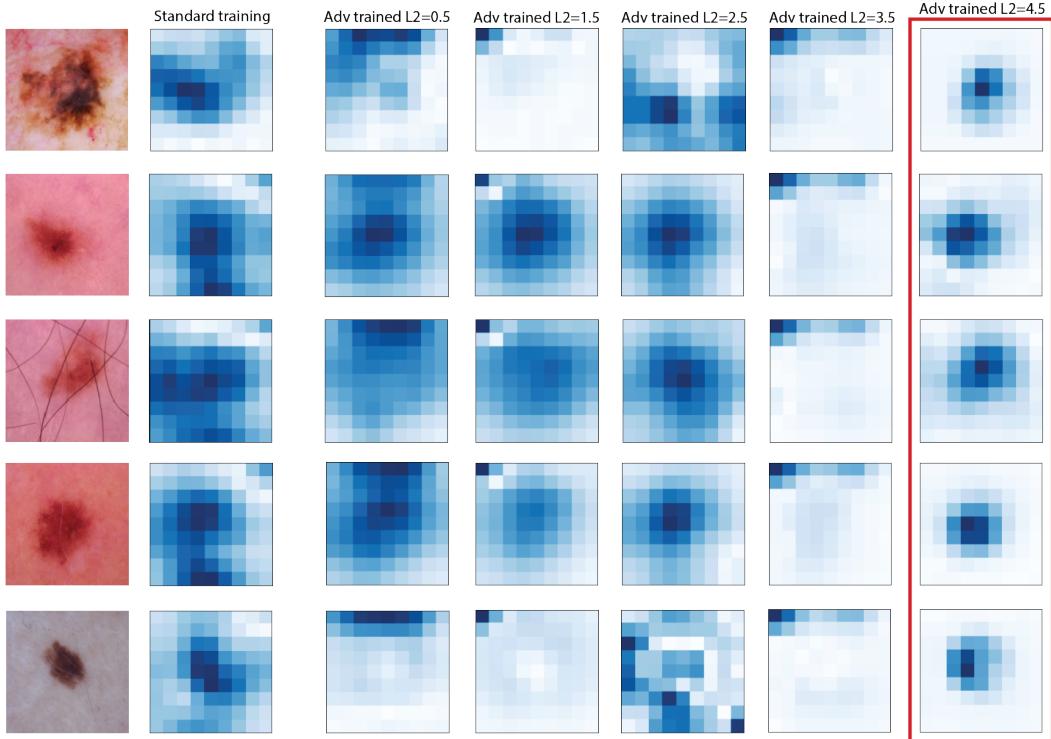


Figure 7: Occlusion attribution on a standard and several robust models. Notice that the standard model attributions are uninformative. Furthermore, it is common that robust models explanation are not coherent (columns 3-6). However, notice in the last column that the robust model trained against an adversary of $\epsilon = 4.5$ gives good localization on all images, hinting that there may be an optimal adversary.

Occlusion [24] is a perturbation-based approach for computing the attributions. It involves replacing rectangular patches of the input with a baseline and computing the decrease of the predicted probability for the new input. A larger drop indicates that the replaced region contained features important for the prediction. In our experiments, we use patches of 50x50px with a stride of 25px. If a feature is part of multiple patches, then it is assigned the average of each patches' sensitivity.

Figure 7 shows Occlusion applied to several robust models and a standard model. Firstly looking at the standard model, notice that the saliency map is noninformative because it gives non-trivial sensitivity to marginal regions of the image.

Analyzing the robust model, notice that using adversary $\epsilon = 4.5$ gives good localization across several images. This suggests that the robust model learns to predict solely based on the lesion; however, we cannot extrapolate this to all robust models, since for example, for $\epsilon = 0.5, \epsilon = 1.5, \epsilon = 2.5, \epsilon = 3.5$ the saliency maps are not coherent.

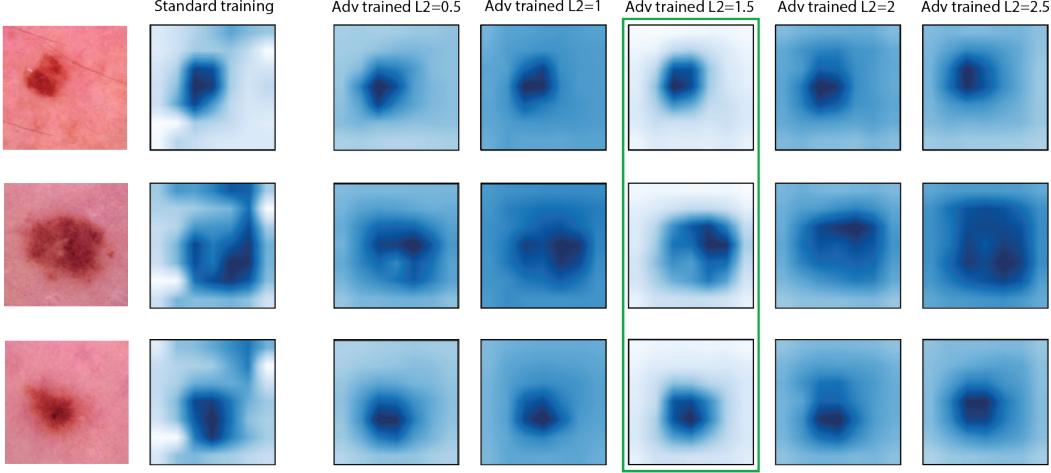


Figure 8: Grad-CAM attribution of several models; Grad-CAM highlights only the regions that positively support predicting the class of interest. Notice that the robust model with $\epsilon = 1.5$ has a more informative saliency map because it gives positive attribution *only* to the lesion. Other robust models attribute a non-trivial level of importance across the skin, making their explanations generally uninformative. This hints towards the possibility of an optimal value of the adversary.

B.4 Grad-CAM

Grad-CAM [15] is a technique for highlighting the regions which positively impact the prediction of a target class. Given a user-chosen convolutional layer A , it computes the gradient of the target class y^c with respect to every unit in the volume outputted by the convolutional layer A_{ij}^k (k represents the feature map, and i, j represent the position in the feature map). Then, it averages the gradients to create a coefficient α_k^c for the contribution of feature map k to class c :

$$\alpha_k^c = \frac{1}{Z} \sum_i \sum_j \frac{\delta y^c}{\delta A_{ij}^k} \quad (3)$$

The importance score attributed by Grad-CAM is the weighted average of the coefficients of the convolutional layer:

$$\text{Grad-CAM}^c = \text{ReLU}\left(\sum_k \alpha_k^c A^k\right) \quad (4)$$

It is important to note two nuances of Grad-CAM. Firstly, the sensitivity is not assigned for an input feature (as it is the case with Gradient and Integrated Gradients). Instead, the sensitivity is attributed to a unit within a convolutional layer, which extracts a particular feature from a given region of the image. Secondly, it shows *only positive* attributions because it applies ReLU on the attributions and all regions with a negative influence receive zero importance.

Our experiment applies Grad-CAM on the last convolutional layer of a ResNet-18 network, which has a $7 \times 7 \times 512$ output volume. Thus the output of a Grad-CAM attribution is 7×7 , where each value corresponds to a region of $224/7 = 32$ pixels. Figure 8 shows the attribution on several models. Notice that the robust model $\epsilon = 1.5$ has a coherent saliency map, as it attributes importance *only* to the lesion. Interestingly, the other robust models assign a non-trivial positive influence on the skin regions; however, the marginal regions should not carry much importance for the prediction.

Appendix C Limitations

C.1 Additional fail-cases for saliency maps of robust models

See Figure 9.

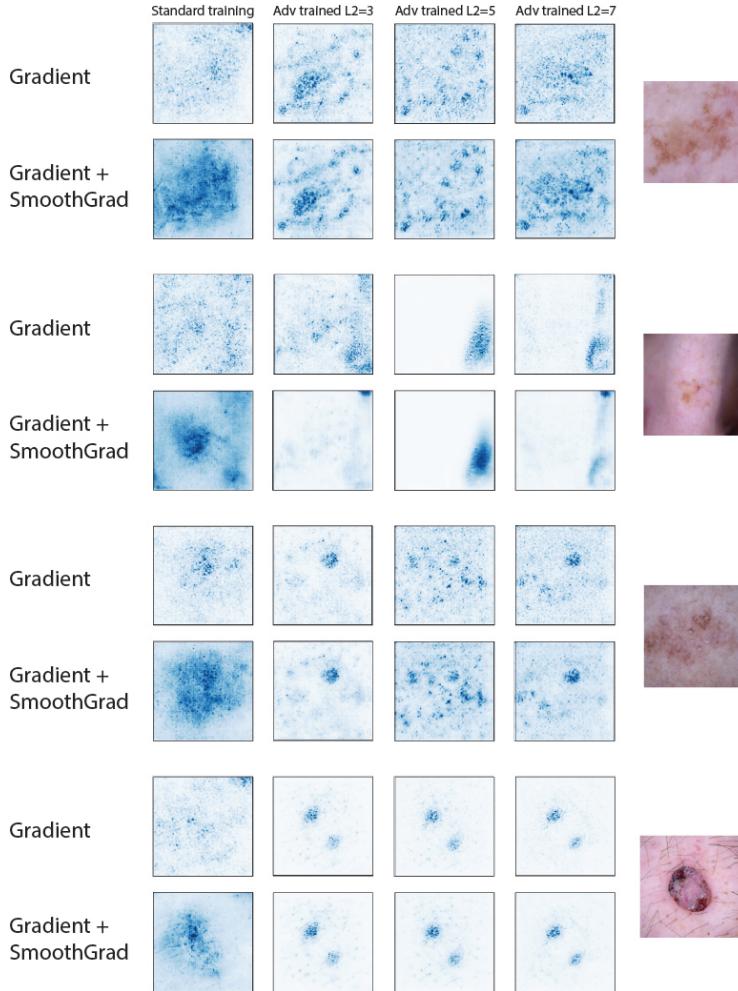


Figure 9: Inputs on which the saliency maps the robust model fail to provide meaningful explanations (e.g., highlights artifacts such as dark regions or providing noisy explanations).

C.2 Saliency maps are brittle to training noise

See Figure 10.

C.3 Varying the adversary power

The saliency map's sharpness varies significantly with the size of the norm used during adversarial training. Figure 11 shows that robust models trained with varying adversary powers ϵ give very different saliency maps on the same input.

Our results indicate the possibility of an image-specific optimal adversary power, similar to a sweet spot, for providing the sharpest saliency maps. Thus, we propose the hypothesis: "The power of the adversary is an image-specific hyper-parameter of the explanations, where the optimal value gives the clearest explanation." It is important to note that different images have different optimal values (in Figure 11, $\epsilon = 4$ seems to be optimal only for the first three images). As a rough empirical estimation,

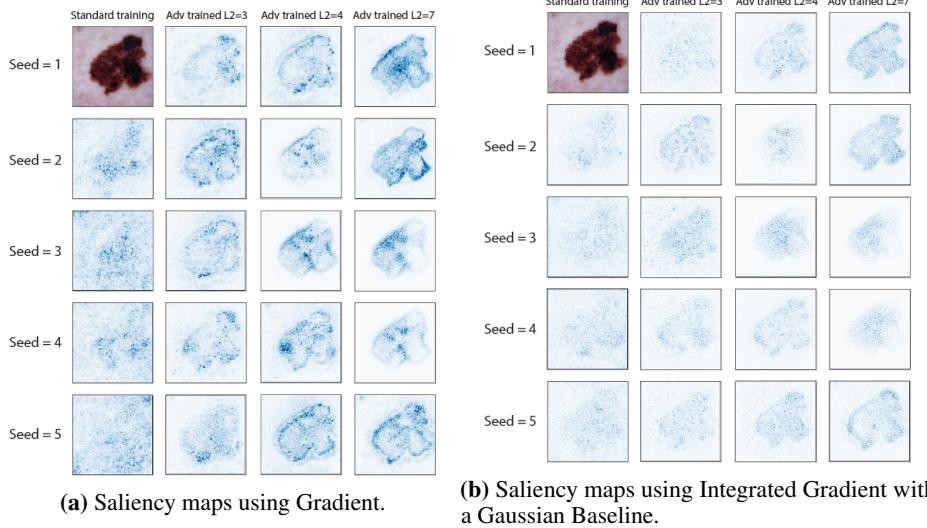


Figure 10: Saliency maps using Gradient and Integrated Gradient on multiple models trained identically except the random seed. The random seed influences the order of the mini-batches. Notice that the saliency maps differ in non-trivial ways across the runs, showing the saliency method themselves are brittle to training noise.

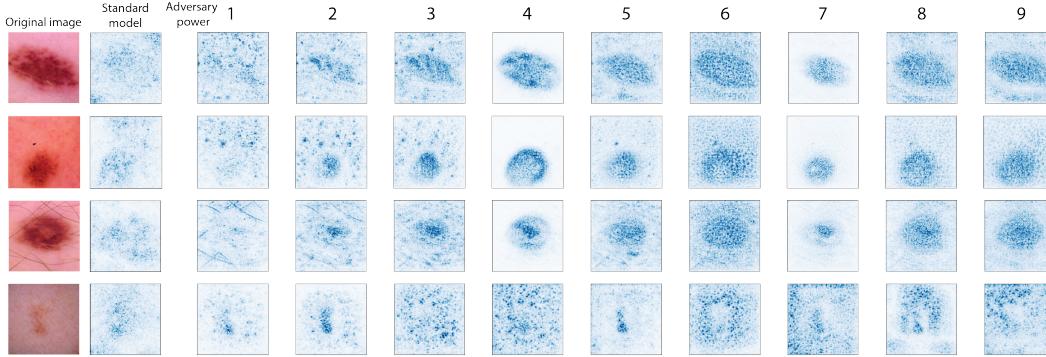


Figure 11: Each row shows the Gradient saliency maps of a standard and robust models trained with varying adversary powers in the l_2 norm. Notice how adversarially trained models with different adversary powers provide different saliency maps on the same input. Also, see that the robust model trained with $\epsilon = 4$ gives a relatively clear explanation for the first three images, but a noisy explanation for the fourth image. This hints towards the possibility of an image-specific optimal adversary power for providing the clearest saliency maps.

we observed that the optimal values tend to be between $\epsilon = 3$ and $\epsilon = 5$, while for $\epsilon \geq 8$, the saliency maps become very noisy.

An interesting observation emerges by looking at the third row from Figure 11. Notice that for $\epsilon = 1$, the saliency map highlights predominantly the hairs, which are irrelevant for the prediction. However, as ϵ increases, the model shifts its attention to the skin lesion, which is of interest. We leave the investigation on how to select the optimal adversary power for future work.

Appendix D Standard fine-tuning the last layers of a robust model

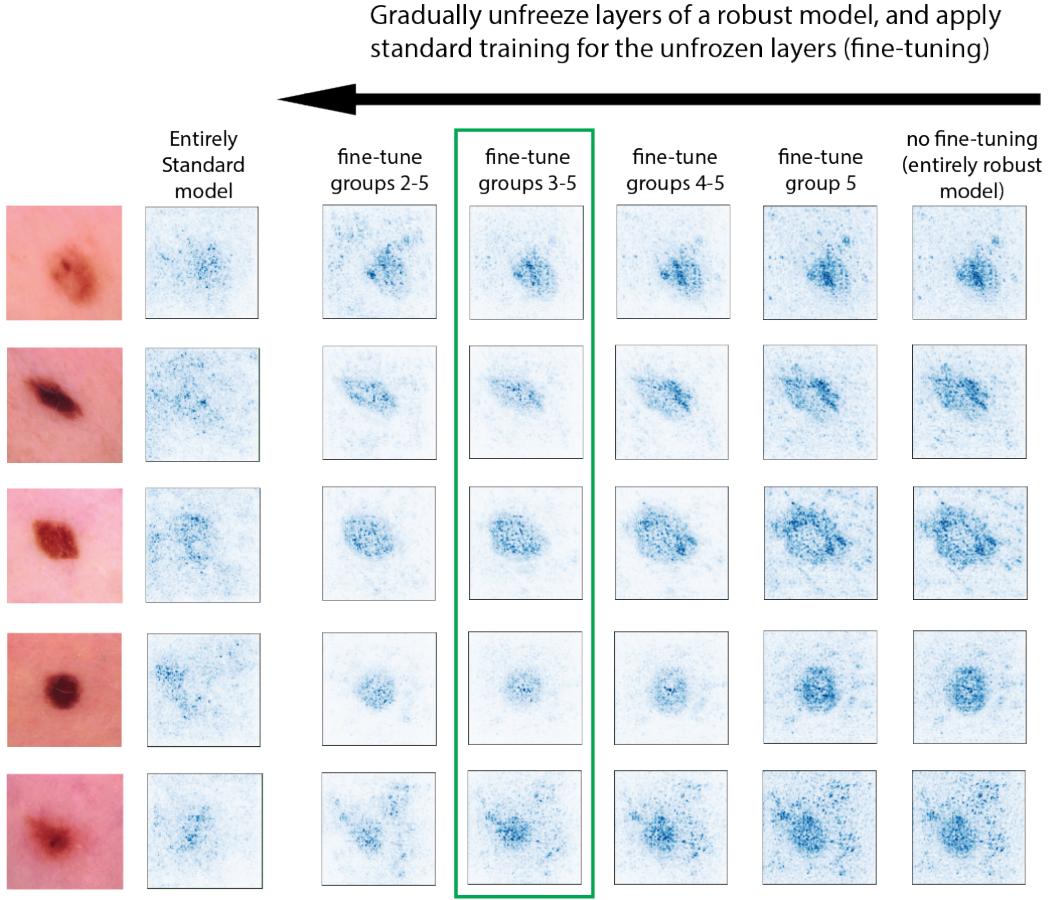


Figure 12: Saliency maps using Gradient method when gradually unfreezing the layers of a robust network, and performing standard fine-tuning. The last column shows the saliency maps of a robust model with $\epsilon = 3$. Columns 3-7 represent new models obtained by fine-tuning a robust model (from the last column) with a learning rate of $1e - 5$. The layers which are not fine-tuned continue to extract robust features. By fine-tuning the last layers, we essentially enable them to combine the robust features extracted by the first layers. Notice how fine-tuning the last layers reduces the noise in the saliency maps.

Making accurate predictions is crucial in high-stake domains, such as Medicine. However, robust models have lower predictive performance than standard models [22]. This raises the question: "Can the performance of a robust model be increased while maintaining its visually coherent explanations?". The answer is affirmative, and we shortly present a methodology for achieving this.

We hypothesize that higher predictive performance could be achieved by making only some layers learn robust features, while other layers learn standard features. This could combine both sides' benefits: higher predictive performance from standard training and sharper saliency maps from adversarial training.

Groups of layers: The ResNet-18 architecture [6] contains five groups of layers: four blocks of convolutional layers and a final fully-connected layer. We perform standard fine-tuning on some groups of layers of a robust model as follows:

1. Get a robust model (i.e., adversarially trained on the target dataset) and freeze all layers, such that the weights are not updated during training.
2. Unfreeze some groups of layers starting from the last one and keep the others frozen. This way, only the unfrozen layers are updated during training.

3. Perform fine-tuning using standard training with early stopping and a small learning rate (we use $1e - 5$).

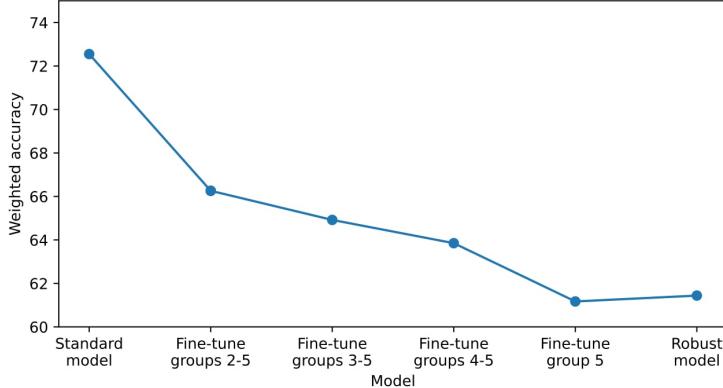


Figure 13: The accuracies of standard, robust and fine-tuned models. Notice that the accuracy increases roughly linearly with fine-tuning more layers. For example, standard fine-tuning of groups 3-5 improves the accuracy by 3% compared to the robust model, but is still lower than the standard model.

Interestingly, fine-tuning the last half of the network increases the predictive performance (Figure 13), while also reducing the noise in the saliency maps. Figure 12 shows the saliency map using Gradient after unfreezing a number of groups and performing standard fine-tuning. The second column shows the standard model, and the last column shows the robust model. The middle columns show the saliency maps for models with only some of the first layers robust and the latter layers fine-tuned in a standard way.

Notice that the model which underwent fine-tuning on groups 3-5 provides a more visually coherent explanation than the standard model and a less noisy explanation than the robust model. Furthermore, it has 3% higher accuracy than the fully robust model (Figure 13).

This experiment shows the potential of using robust first layers as a way to improve interpretability. A natural question is: "Why does fine-tuning reduce noise in the saliency map?". We conjecture that visually coherent explanations are attributed to having robust first layers. Each layer of a CNN can be thought of as a feature extractor at different abstraction levels by combining the features extracted by the previous layers [24]. In this way, the first layers extract robust low-level features, and the latter ones combine them in any way that increases the predictive performance. So, the fine-tuned layers learn to combine the robust low-level features extracted by the first layers. We leave this investigation as future work.