COMPARING EXPLAINABLE MODELS AND BLACK-BOX MODELS FOR AUTOMATIC METASTATIC TISSUE DETECTION

Kaya ter Burg, Maarten Burger, Jesse Maas

ABSTRACT

Manual metastatic tissue detection is a time-consuming task and it would be beneficial to (partially) automate this process. This is a high-stakes environment, which means that not only should this task be done with high performance, but the decision-making process should also be interpretable. In this work we evaluate various black-box models: ResNet, ResNeSt and our own group equivariant CNN model. Furthermore, we train an inherently explainable model, which uses slot attention to generate its own explanations. We found no significant improvement from using equivariance compared to the non-equivariant baseline. The interpretable models provide useful visualisations while achieving comparable performance to the non-explainable black-box models. We argue for the importance of explainable AI in such a high-stakes environment and show that having interpretable models is a viable direction for automatic metastatic tissue detection.

1. INTRODUCTION

Manual metastatic tissue detection is a time-consuming task. Pathologists have to use valuable time to manually go through images of histopathologic scans in order to find metastatic tissue. Going through these images is not only tedious and time-consuming, but also prone to misinterpretation. With modern appliances we can create a large number of histopathologic scans in a very short time, making performing metastatic tissue detection on these tasks manually infeasible [1]. Thanks to the further digitization of pathology and advancements in computer vision, feasible solutions based on Artificial Intelligence seem to be within reach [2]. In this work, we further explore possibilities for automatic detection using various AI techniques and argue for the importance of explainable AI in tasks such as these. We aim to help answer the question of whether an AI-assisted solution is feasible for pathologists.

We first train multiple baselines in order to assess the performance of a more simple AI model, in particular a convolutional neural network (CNN), on a dataset of histopathologic scans within our computational constraints. Previous works have already shown that AI can achieve good performance on such a task [3]. These baselines can be used for us to compare our more complex models against in terms of performance. We experiment with group equivariant convolutional layers in order to determine if these can improve the performance of the model. Lastly, we train an explainable AI model that uses intrinsic explanations and compare the performance to the other models, which are all black-box models. We believe that an explainable approach can truly help in this field as transparency and interpretability are important for experts to not only determine when the model might be wrong, but also to gain trust in the model.

In summary, we propose an inherently explainable model for automatic metastatic tissue detection and compare the performance of this model against both standard and rotation equivariant models.

2. BACKGROUND & METHODOLOGY

2.1. Dataset

We used the PatchCamelyon (PCam) dataset [3]. It contains coloured images of 96 x 96 pixels. The images are taken from histopathologic scans of sentinel lymph node sections of breast cancer patients. Each image has a binary label, which indicates whether at least one pixel in the centre 32 x 32 pixel region of the image contains metastatic tissue. The training set contains 262,144 images and the validation and test set both contain 32,768 images. All splits are balanced with regard to the labels. Example images from the dataset can be found in Appendix A.

2.2. ResNet & ResNeSt

Challenging image analysis problems like automatic metastatic tissue detection with the use of AI require deep neural networks with many layers (i.e. a stack of hundreds of layers). The latter stems from the fact that the network should be able to extract relevant features and learn meaningful patterns in order to solve such issues. However, neural networks which are too deep may suffer from vanishing gradients [4, 5].

Residual networks (ResNets) are deep neural networks that overcome the problem of vanishing gradients by introducing the choice of skipping connections to enable identity mapping [6]. Various variants of the ResNet model exist, making use of the same architecture but with a different number of layers. Some of the variants are ResNet-36 and ResNet-50, where the numbers simply refer to the amount of layers in the architecture.

Even though ResNet models achieve high performance, the architecture is lacking when it comes to feature correlations. To solve the problem of lacking feature correlations, previous models such as ResNet, ResNext and SE-Net are combined to introduce a split-attention network: ResNeSt [6, 7, 8, 9]. ResNeSt makes use of cardinality as proposed in [7], i.e. features get divided into several smaller groups. Furthermore, squeeze and excitation blocks of the SE-Net [8] are used to apply gating on the channel dimensions of the feature maps.

2.3. Equivariant CNNs

The PCam dataset has the property of being invariant under some operations, specifically rotation and flipping. In other words, if we rotate or flip the images, the presence of metastatic tissue does not change. We can exploit this fact to improve the performance of the model. One way to do this could be by using data augmentation. However, it is also possible to put this prior knowledge directly into the architecture of the model. We do this by using group convolutional layers [10] as implemented in the escnn library for PyTorch [11, 12, 13]. The advantage of this approach is that the neural network is equivariant at each layer of the network (except for the final, linear layer), not just over the input-output relation. All the hidden features that the networks learns are then also equivariant. This stronger inductive bias should improve the performance more. Also, these layers are mathematically guaranteed to be equivariant, rather than hoping your neural networks learns this property through data augmentation. Note the difference between invariance and equivariance: invariance means the output does not change under certain operations, whereas with equivariance, the output changes in the same way as the input does. So for example, the output rotates the same amount as the input, but stays the same in all other respects. We will investigate whether the introduction of equivariant layers improves performance on the PCam dataset.

2.4. Explainable AI: SCOUTER

Having models that achieve a high performance should not be the only goal when developing a tool for automatic metastatic tissue detection. It is vital for the models to be interpretable for humans. Especially in the medical field, where high-stake decisions have to be made, these decisions can have huge consequences [14, 15]. It is thus important for our models to not only do their job well, but also to be transparent about how they came to their decisions. Furthermore, being able to understand the inner workings of a model also plays an important part in trusting said model [16], which is crucial for the adoption of AI models into real-world scenarios.

The field of eXplainable AI (XAI) concerns itself with the development of AI models that can be understood by humans. Various approaches and methods to achieve this goal have been developed, many belonging to the class of *post-hoc* methods. This type of method tries to explain existing (black-box) models which do not have to be inherently explainable by themselves and are not designed to be explainable either [17]. The usage of post-hoc methods for explaining models is a point of discussion and regarded by some as an unsuitable form of achieving interpretability [18, 14]. Arguably, models should ideally be inherently interpretable by themselves [19], where the model does not simply produce an output, but also an explanation on its own without using post-hoc methods.

For the explainability experiment we used a SCOUTER [20] model. SCOUTER uses xSlot attention modules inside its architecture to create an inherently explainable model. SCOUTER models thus have no need for post-hoc methods and can generate their own explanations. The SCOUTER architecture consists of a CNN backbone (e.g. ResNet-18 [6]) to extract features from the image, followed by the xSlot module. The xSlot module contains multiple slots, where each slot gives the confidence of the image belonging to a certain class. Finally, the output of the xSlot module is fed into a softmax operation to obtain the predictions. A detailed visualisation of the architecture can be found in Appendix C.

The explanations that SCOUTER generates can be both positive or negative. SCOUTER can generate explanations that tell us why an image would be classified as a particular class, but by inverting the loss function it can also generate explanations that show us why an image would not be classified as a certain class. These counter-factual explanations can give us further insights into the model and are a valuable addition together with the positive explanations. A separate model has to be trained for both versions, indicated as SCOUTER$_+$ for positive explanations and SCOUTER$_-$ for negative explanations.

The SCOUTER model has been applied on the ACRIMA dataset [21], a dataset consisting of fundus images used for binary glaucoma detection. It achieved high performance for that task [20, 22], indicating the model might be appropriate for the PCam dataset as well. With that and SCOUTER being inherently interpretable, this model seems a fitting choice for achieving explainable metastatic tissue detection.

3. EXPERIMENTS & RESULTS

Since we are using a dataset with binary classes, we evaluate the models using the following confusion matrix metrics: area under curve (AUC), accuracy (Acc), recall (Rec), precision (Prec), F1-score (F1) and Cohen's kappa (Kappa). The results for all types of models we trained can be found in Table 1. Full hyperparameter settings can be found in Appendix D. All code used is available at: `https://github.com/kayatb/AIMI_PCAM`.

Model	AUC	Acc	Rec	Prec	F1	Kappa
ResNet-18	0.91	0.81	0.76	0.86	0.80	0.62
ResNeSt-26d	0.89	0.81	0.71	0.89	0.78	0.62
ResNeSt-50d	0.91	0.82	0.75	0.88	0.81	0.65
ResNest-269e	0.88	0.80	0.68	0.90	0.77	0.61
CNN base	0.95	0.88	0.87	0.89	0.88	0.77
CNN flip eqv.	**0.96**	**0.89**	0.90	0.88	**0.89**	**0.78**
CNN flip & 180° eqv.	0.95	0.88	0.87	0.88	0.87	0.76
CNN flip & 90° eqv.	0.93	0.85	0.83	0.86	0.85	0.71
CNN flip & 45° eqv.	0.94	0.87	0.82	0.91	0.86	0.73
SCOUTER$_+\lambda = 1$	0.95	0.88	0.79	0.96	0.87	0.76
SCOUTER$_+\lambda = 3$	0.95	**0.89**	0.82	0.96	0.88	**0.78**
SCOUTER$_+\lambda = 10$	0.94	0.85	0.72	**0.97**	0.83	0.70
SCOUTER$_-\lambda = 1$	0.95	0.85	**0.98**	0.78	0.86	0.69
SCOUTER$_-\lambda = 3$	**0.96**	0.86	**0.98**	0.80	0.88	0.73
SCOUTER$_-\lambda = 10$	0.94	0.87	0.97	0.79	0.87	0.72

Table 1: Results for all trained models on the test set. Reported from left to right are the following metrics: area under curve, accuracy, recall, precision, F1-score and Cohen's kappa. All metrics were calculated on the test set.

3.1. Baselines

For the baselines we used the ResNet-18, ResNeSt-26d, ResNeSt-50d and ResNeSt-296e variants. The pre-trained weights from all models are directly used from their original sources [6, 9]. For ResNet-18 we used the implementation of PyTorch, specifically Torchvision [23] [1]. For the ResNeSt models we used the implementations of Timm [24] [2].

The hyperparameters used for all the models are the same, except the batch size of the ResNeSt-296e. A lower batch size was required due to memory constraints.

We found high training accuracies (>0.95) for all models, with ResNeSt-50d and ResNeSt-296e having training accuracies of over 0.99.

3.2. Group Equivariance

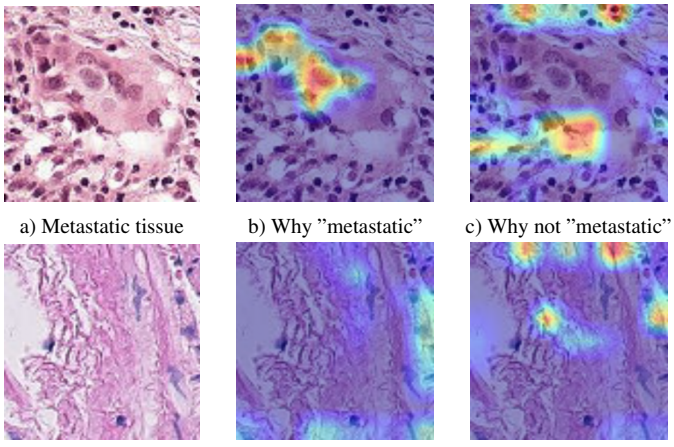
To test the effectiveness of using group equivariant layers for this task, separate models were constructed using equivariant layers from the escnn library [11, 12] for PyTorch. We evaluated models without equivariances, with flip equivariance, and flip plus rotation equivariances. For rotation equivariance, we tested 180, 90 and 45 degrees rotations. This results in a total of five different model configurations.

The base, non-equivariant model model consists of 12 convolutional layers with 7×7 kernels and zero-padding, combined with ReLU activations, average pooling layers, and a final linear layer. The full architecture can be found in Appendix B. For the equivariant models, the convolutional layers are replaced by group convolutional layers, and the number of channels is divided by the square root of the group size to maintain the same number of trainable parameters for a fair comparison. The group sizes are 2, 4, 8, and 16 for flipping, and flipping plus 180/90/45 degrees rotations, respectively. We also use the average pooling layers provided by escnn to maintain equivariance.

Each model was trained for five epochs and the model from the best epoch with the highest validation accuracy was evaluated on the test set. This process is repeated for five different runs with different random weight initializations. The full results for each run can be found in Appendix E. Based on these results, there is not a clear added value of using equivariant layers. There is no statistically significant difference between any of the architectures. A t-test on models with the biggest difference in performance (CNN flip eqv. and CNN flip & 90°) yields a p-value of $0.052 > 0.05$.

3.3. SCOUTER

In order to train the SCOUTER models we use a hyperparameter setup close to the SCOUTER model of the original authors [20]. We train both positive and negative SCOUTER models for different λ values, where we use the same λ values as tested by the original authors. In a SCOUTER model, the λ value is a hyperparameter that controls the importance of the area loss. In other words: λ defines how the model is penalized for adding explanations, so with a higher λ SCOUTER reduces the size of the attention area and with a lower λ SCOUTER increases the size of the attention area.



a) Metastatic tissue b) Why "metastatic" c) Why not "metastatic"

d) Non-metastatic tissue e) Why "metastatic" f) Why not "metastatic"

Fig. 1: Examples of SCOUTER images, these were generated using SCOUTER models with $\lambda = 10$.

We found that across all models and hyperparameter configurations, SCOUTER achieves comparable or better performance to the tested non-explainable models. Full confusion matrices for all SCOUTER model configurations can be found in Appendix F.

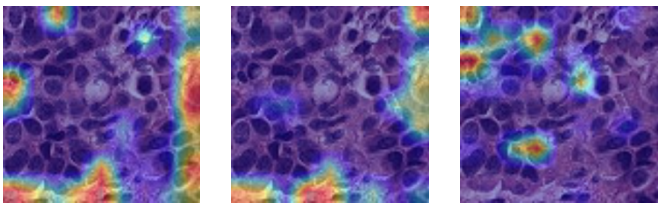
[1] https://pytorch.org/vision/main/models/generated/torchvision.models.resnet18.html

[2] https://github.com/rwightman/pytorch-image-models/blob/main/timm/models/resnest.py

We also found that all SCOUTER$_+$ models achieved very high precision scores but lower recall scores while SCOUTER$_-$ models achieved very high recall scores but lower precision scores.

In Figure 1, examples of positive and negative explanations generated by SCOUTER models are shown. The top row displays the explanations for an image that contains metastatic tissue and the bottom row for one with no metastatic tissue. We can see that in case of metastatic tissue being present, SCOUTER$_+$ focuses on the correct part of the image. SCOUTER$_-$ on the other hand, focuses on the healthy tissue that lays around the metastatic patch. For the image without metastatic tissue, SCOUTER$_+$ does not attend anywhere much. This is to be expected, since no metastatic tissue is to be found in this image. SCOUTER$_-$ focuses again on the healthy tissue. Similar behaviour can be observed across different images. More examples can be found in Appendix G.

In Figure 2, the effect of the λ hyperparameter is demonstrated. As to be expected, the explanation area size becomes smaller with increasing λ. As we increase λ, the SCOUTER models find different and smaller areas that support its decision. Regarding the task of metastatic tissue detection, one would prefer to have precise explanations that point to specific (patches of) cells, i.e. a larger λ value.



a) $\lambda = 1$ b) $\lambda = 3$ c) $\lambda = 10$

Fig. 2: Effects of the λ parameter on explanations generated by SCOUTER$_+$ models. This image contains metastatic tissue.

4. CONCLUSION & DISCUSSION

In this work, we compared the performance of various black-box models with that of an inherently interpretable model on a metastatic tissue detection task.

We found that overall the residual neural networks achieved decent accuracy scores and area under curve scores. However, these models did suffer from overfitting. Perhaps more hyperparameter tuning or architectural changes are needed to combat this. Unfortunately, we could not do a more thorough hyperparameter or architectural search due to time and computational constraints.

No significant improvement could be found from incorporating equivariance in the model architecture. This could be due to the reduced number of channels in the equivariant models. It would be interesting to see how larger networks will perform. Since we could not do proper hyperparameter optimization, it could be the case that even though the model already overfits, a larger model would still perform better. We leave this for future work to find out.

We have shown that SCOUTER models can be used to generate explainable model decisions without sacrificing performance. We believe that explainability is vital for models that are to be applied in the medical field. We showed that interpretable models achieve similar performance compared to black-box models. This demonstrates that having interpretable models is a viable way forward.

Especially interesting is very high recall score (0.98) of SCOUTER$_-$. In the world of disease classification, false negatives can be disastrous and as such a high recall score is desirable. However, this is at the cost of lower precision, which would result in more false positives. Fortunately, since the model is explainable, a pathologist could look at why the model classifies an image of tissue as metastatic and either agree or overrule the decision.

Although the explanations look convincing, we only did a qualitative evaluation of them and no quantitative analysis. This is of course necessary in order to further verify that the explanations are correct. This could be accompanied with further tuning of the λ parameter. Furthermore, our own qualitative assessment is limited, since we are not pathologists. For a further investigation into explainable metastatic tissue detection, inquiring a domain expert might be beneficial. Such an expert would be able to guide the explanation visualisation into being useful for practical applications. We believe that this is a promising direction for future research.

5. REFERENCES

[1] M. Botros, "PCam challenge," University Seminar Presentation, 2022.

[2] M. Cui and D. Y. Zhang, "Artificial intelligence and computational pathology," *Laboratory Investigation*, vol. 101, no. 4, pp. 412–422, Apr 2021. [Online]. Available: https://doi.org/10.1038/s41374-020-00514-0

[3] B. S. Veeling, J. Linmans, J. Winkens, T. Cohen, and M. Welling, "Rotation equivariant CNNs for digital pathology," Jun. 2018.

[4] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.

[5] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 249–256.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[7] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[8] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[9] H. Zhang, C. Wu, Z. Zhang, Y. Zhu, H. Lin, Z. Zhang, Y. Sun, T. He, J. Mueller, R. Manmatha *et al.*, "Resnest: Split-attention networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2736–2746.

[10] T. Cohen and M. Welling, "Group equivariant convolutional networks," in *International conference on machine learning*. PMLR, 2016, pp. 2990–2999.

[11] M. Weiler and G. Cesa, "General E(2)-Equivariant Steerable CNNs," in *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[12] G. Cesa, L. Lang, and M. Weiler, "A program to build E(N)-equivariant steerable CNNs," in *International Conference on Learning Representations*, 2022. [Online]. Available: https://openreview.net/forum?id=WE4qe9xlnQw

[13] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035. [Online]. Available: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf

[14] C. Rudin, "Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead," *Nature Machine Intelligence*, vol. 1, no. 5, pp. 206–215, 2019.

[15] B. Letham, C. Rudin, T. H. McCormick, and D. Madigan, "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model," *The Annals of Applied Statistics*, vol. 9, no. 3, pp. 1350–1371, 2015.

[16] K. Weitz, D. Schiller, R. Schlagowski, T. Huber, and E. André, "" do you trust me?" increasing user-trust by integrating virtual agents in explainable ai interaction design," in *Proceedings of the 19th ACM International Conference on Intelligent Virtual Agents*, 2019, pp. 7–9.

[17] G. Ras, N. Xie, M. van Gerven, and D. Doran, "Explainable deep learning: A field guide for the uninitiated," *Journal of Artificial Intelligence Research*, vol. 73, pp. 329–397, 2022.

[18] M. Ghassemi, L. Oakden-Rayner, and A. L. Beam, "The false hope of current approaches to explainable artificial intelligence in health care," *The Lancet Digital Health*, vol. 3, no. 11, pp. e745–e750, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2589750021002089

[19] A. Sudjianto and A. Zhang, "Designing inherently interpretable machine learning models," *CoRR*, vol. abs/2111.01743, 2021. [Online]. Available: https://arxiv.org/abs/2111.01743

[20] L. Li, B. Wang, M. Verma, Y. Nakashima, R. Kawasaki, and H. Nagahara, "Scouter: Slot attention-based classifier for explainable image recognition," in *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[21] A. Diaz-Pinto, S. Morales, V. Naranjo, T. Köhler, J. M. Mossi, and A. Navea, "Cnns for automatic glaucoma assessment using fundus images: An extensive validation," Mar 2019. [Online]. Available: https://figshare.com/articles/dataset/CNNs_for_Automatic_Glaucoma_Assessment_using_Fundus_Images_An_Extensive_Validation/7613135/1

[22] M. Burger, K. ter Burg, S. Titarsolej, and S. J. Khan, "[Re] Reproducibility Study - SCOUTER: Slot Attention-based Classifier for Explainable Image Recognition," *ReScience C*, vol. 8, no. 2, p. 8, May 2022. [Online]. Available: https://doi.org/10.5281/zenodo.6574641

[23] S. Marcel and Y. Rodriguez, "Torchvision the machine-vision package of torch," in *Proceedings of the 18th ACM International Conference on Multimedia*, ser. MM '10. New York, NY, USA: Association for Computing Machinery, 2010, p. 1485–1488. [Online]. Available: https://doi.org/10.1145/1873951.1874254

[24] R. Wightman, "Pytorch image models," https://github.com/rwightman/pytorch-image-models, 2019.

[25] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014.

A. PCAM DATA EXAMPLES

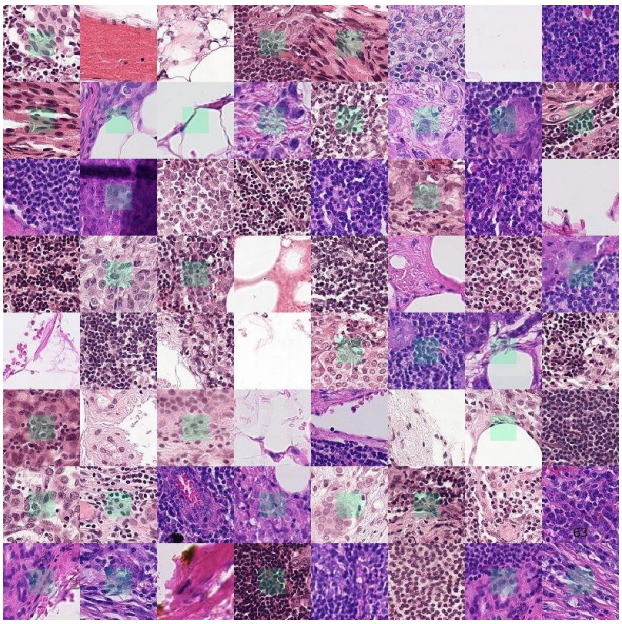


Fig. 3: Examples of images in the PCam dataset. A green square indicates the presence of at least one pixel of metastatic tissue in the centre region.

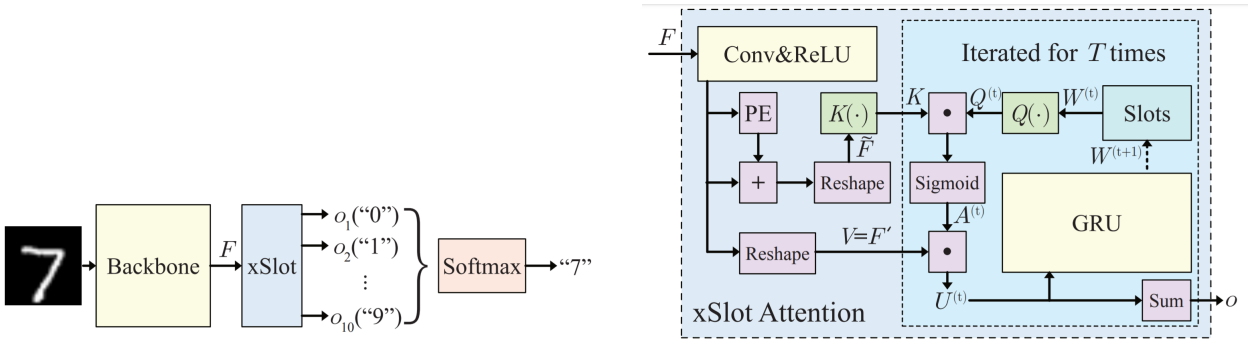
B. EQUIVARIANT CNN ARCHITECTURE

Layer	Image Size ($W \times H$)
Input (3 channels)	96×96
12-channel Conv	96×96
12-channel Conv	96×96
Average Pooling	48×48
24-channel Conv	48×48
24-channel Conv	48×48
Average Pooling	24×24
48-channel Conv	24×24
48-channel Conv	24×24
Average Pooling	12×12
96-channel Conv	12×12
96-channel Conv	12×12
Average Pooling	6×6
192-channel Conv	6×6
192-channel Conv	6×6
Linear Layer	1×1

Table 2: The architecture of the base, non-equivariant CNN model.

C. SCOUTER ARCHITECTURE

An overview of the complete SCOUTER architecture and a detailed view inside the xSlot attention module can be found in Figure 4. As can be seen, SCOUTER consists of a backbone, xSlot module and a softmax function.



a) Classification pipeline

b) xSlot attention module

Fig. 4: Overview of the SCOUTER architecture taken from Figure 2 in [20]

D. HYPERPARAMETERS

Model	Epochs	Batch Size	Learning Rate	Optimizer
ResNet18	10	128	0.0001	Adam [25]
ResNeSt-26d	10	128	0.0001	Adam
ResNeSt-50d	10	128	0.0001	Adam
ResNeSt-269e	10	64	0.0001	Adam

Table 3: Hyperparameter settings used for the residual neural network experiments.

Model	Epochs	Batch Size	Learning Rate	Optimizer	Rotations
CNN base	10	128	0.0001	Adam	1
CNN flip eqv.	10	128	0.0001	Adam	1
CNN flip & 180° eqv.	10	128	0.0001	Adam	2
CNN flip & 90° eqv.	10	128	0.0001	Adam	4
CNN flip & 45° eqv.	10	128	0.0001	Adam	8

Table 4: Hyperparameter settings used for the tested (rotation equivariant) CNN experiments.

Hyperparameter	Value	Hyperparameter	Value
Epochs	10	λ Value	$\{1, 3, 10\}$
Batch Size	64	Slots per Class	3
Number of Classes	2	Power of Slot Loss	2
Learning Rate	0.0001	Image Size	260
Learning Rate Drop	70	Channel	2048
Hidden Dimensions	64	Number of Freeze Layers	0
Hidden Layers	3	Number of Workers	4
Weight Decay	0.0001	World Size	1

Table 5: Hyperparameter settings used for the SCOUTER experiments.

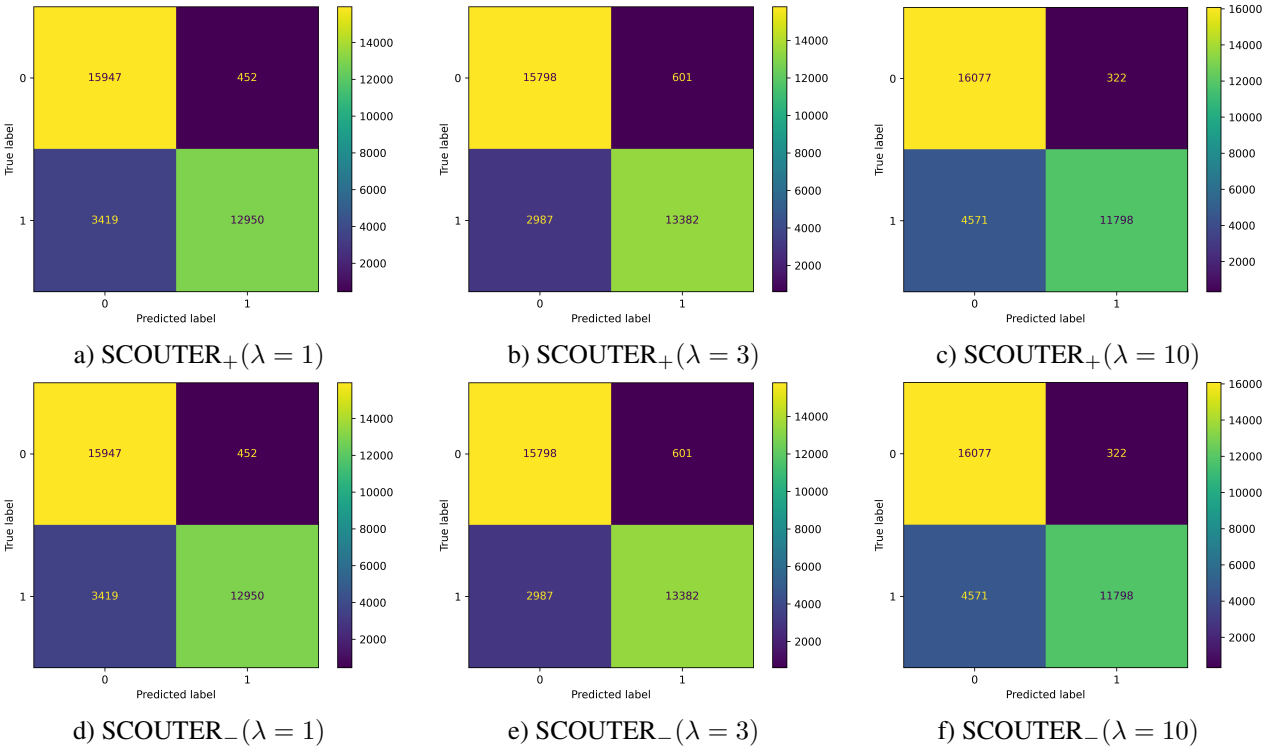
E. FULL EQUIVARIANT CNN RESULTS

Run #	No Eqv	Flip Eqv	Flip + 180°	Flip + 90°	Flip + 45°
1	0.857	0.871	0.864	0.852	0.870
2	0.861	0.868	0.861	0.846	0.853
3	0.869	0.890	0.844	0.826	0.870
4	0.883	0.887	0.879	0.854	0.847
5	0.833	0.832	0.872	0.837	0.865
Avg	0.860	0.869	0.864	0.843	0.861
Std	0.018	0.023	0.013	0.012	0.010

Table 6: Test set accuracy for all (non-)equivariant CNN models over five separate runs together with their average and standard deviation.

F. SCOUTER CONFUSION MATRICES

In Figure 5, the test set confusion matrices for all SCOUTER models we trained can be found. The raw numbers from these matrices were used for the calculations of all metrics reported in Table 1.



a) SCOUTER$_+(\lambda = 1)$ b) SCOUTER$_+(\lambda = 3)$ c) SCOUTER$_+(\lambda = 10)$

d) SCOUTER$_-(\lambda = 1)$ e) SCOUTER$_-(\lambda = 3)$ f) SCOUTER$_-(\lambda = 10)$

Fig. 5: Confusion matrices for all trained SCOUTER model configurations on the test set.

G. ADDITIONAL SCOUTER IMAGES

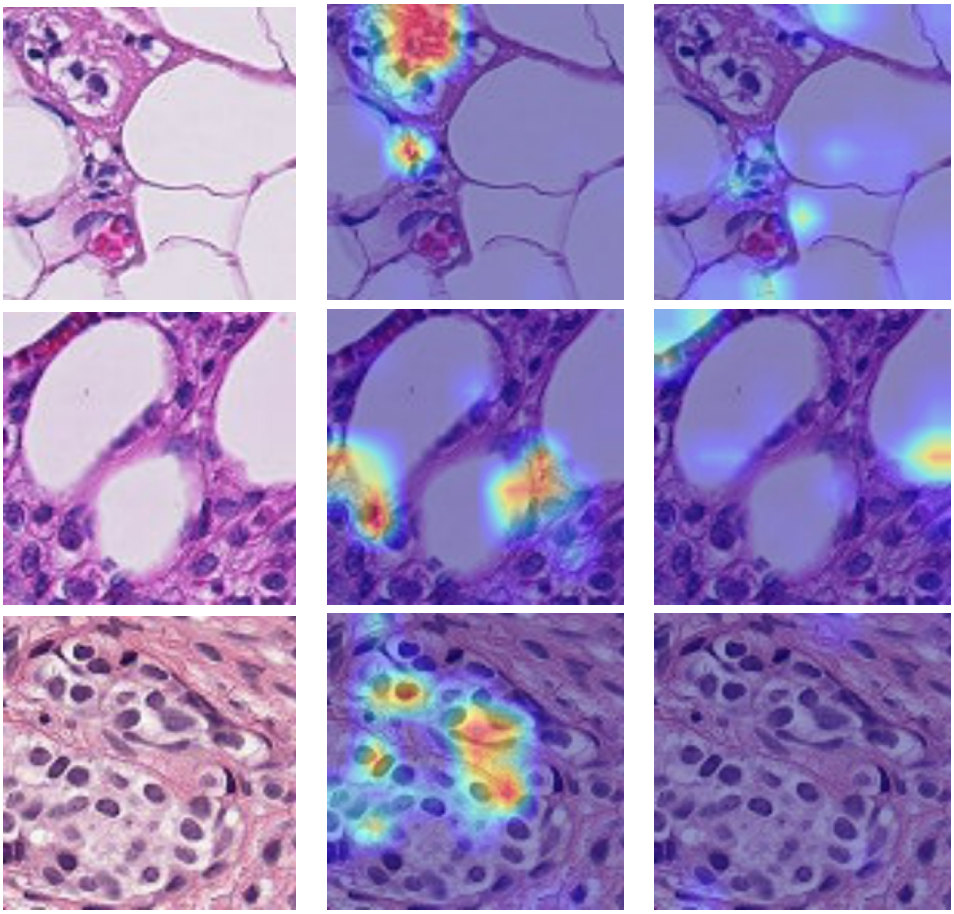
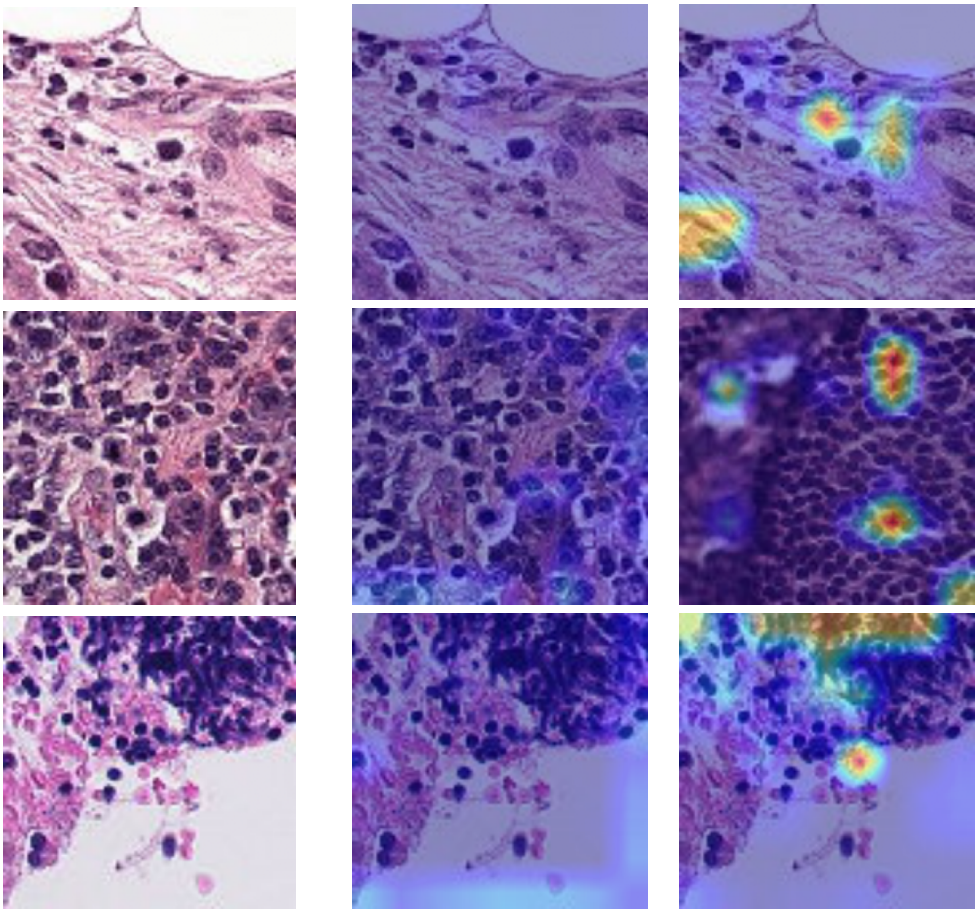


Fig. 6: Additional explanations generated by the SCOUTER model for metastatic images.



Original image (non-metastatic) Why ”metastatic” Why not ”metastatic”

Fig. 7: Additional explanations generated by the SCOUTER model for non-metastatic images.