Revisiting FunnyBirds evaluation framework for prototypical parts networks[⋆]

Szymon Opłatek[1], Dawid Rymarczyk[1,2][0000−0002−8543−5200], and
Bartosz Zieliński[1,3][0000−0002−3063−3621]

[1] Jagiellonian University, Faculty of Mathematics and Computer Science,
Łojasiewicza 6, 30-348, Poland
`szymon.oplatek@student.uj.edu.pl`
`{dawid.rymarczyk;bartosz.zielinski}@uj.edu.pl`
[2] Ardigen SA, Podole 76, 30-394, Kraków, Poland
[3] IDEAS NCBR, Chmielna 69, 00-801 Warsaw, Poland

Abstract. Prototypical parts networks, such as ProtoPNet, became popular due to their potential to produce more genuine explanations than post-hoc methods. However, for a long time, this potential has been strictly theoretical, and no systematic studies have existed to support it. That changed recently with the introduction of the FunnyBirds benchmark, which includes metrics for evaluating different aspects of explanations. However, this benchmark employs attribution maps visualization for all explanation techniques except for the ProtoPNet, for which the bounding boxes are used. This choice significantly influences the metric scores and questions the conclusions stated in FunnyBirds publication. In this study, we comprehensively compare metric scores obtained for two types of ProtoPNet visualizations: bounding boxes and similarity maps. Our analysis indicates that employing similarity maps aligns better with the essence of ProtoPNet, as evidenced by different metric scores obtained from FunnyBirds. Therefore, we advocate using similarity maps as a visualization technique for prototypical parts networks in explainability evaluation benchmarks.

Keywords: Prototypical Parts · Interpretability · xAI evaluation

Errata

After our paper was accepted at the XAI 2014 conference, we discovered an inaccuracy that needs clarification and correction. While this inaccuracy does not affect the interpretation or conclusions of our results, it is important for the community to be informed about it.

We submitted our paper to XAI 2014 before the authors of FunnyBirds framework [9] published their full source code. That is why we had to reimplement

the code of the ProtoPNet experiment using the description from the Funny-Birds paper. We also had to train a set of ProtoPNet models, as they were also not published. As a result, our results were obtained for the reimplemented experiment and our ProtoPNet models.

After submitting our paper to XAI 2014, the FunnyBrids authors published the code of the ProtoPNet experiment and the ProtoPNet model. However, their source code contained a critical error, significantly changing the metrics values. We reported it as an issue at the FunnyBirds GitHub repository[4]. After fixing this error, we achieved higher metrics values, as presented in Table 1.

Table 1: SD and TS metrics values for the original FunnyBirds (FB) code with and without error differ significantly.

Metric	FB code with error	FB code without error
Accuracy	0.94	0.94
BI	1.00	1.00
CSDC	0.93	0.93
PC	0.91	0.91
DC	0.92	0.93
D	0.58	0.58
SD	0.24	**0.75**
TS	0.46	**0.56**

Moreover, after incorporating our Summed Similarity Maps to FunnyBirds code without error, we obtained results presented in Table 2, which confirm the conclusions presented in our XAI 2014 conference paper.

Table 2: After fixing the error of FunnyBirds (FB) code and incorporating our Summed Similarity Maps, the main conclusions of our paper hold, i.e., values of D, SD, and TS rise, while scores for CDSC, PC, and DC drop. The reason for that is explained in the Subsection 5.1. The table below should be considered instead of the Table 3.

Metric	BB (FB code w/o error)	SSM (FB code w/o error)
Accuracy	0.94	0.94
BI	1.00	1.00
CSDC	**0.93**	0.89
PC	**0.91**	0.84
DC	**0.93**	0.89
D	0.58	**0.61**
SD	0.75	**0.83**
TS	0.56	**0.64**

---

[4] https://github.com/visinf/funnybirds/issues/5

We believe these corrections will lead to a more accurate understanding of the ProtoPNet model as well as the FunnyBirds evaluation framework. The rest of the work is as it was originally published at the XAI 2014 conference.

1 Introduction

Standard deep neural networks (DNNs) lack transparency in their decision-making process, posing challenges for human verification, especially in critical domains such as medicine [14,20]. In response, the field of eXplainable Artificial Intelligence (XAI) has emerged with post-hoc and ante-hoc methods. Post-hoc methods are commonly used because they can be applied to already-trained neural networks. However, various studies have highlighted their potential biases [1,3,26], raising concerns about the reliability of their explanations. Consequently, ante-hoc methods like ProtoPNet [6] and B-Cos [5] have gained prominence.

These intrinsically interpretable or self-explainable methods operate under the assumption that the model's design inherently allows for interpretable predictions. However, they often require more complex training to achieve comparable accuracy to standard DNNs, leading to the interpretability-accuracy trade-off phenomenon [21].

Practitioners encounter a dilemma regarding whether to choose a standard DNN coupled with a post-hoc explanation method to achieve higher accuracy or to invest in the development of a self-explanatory model to enhance interpretability. Addressing this question necessitates a reliable and trustworthy evaluation framework for model explanations. This challenge is tackled by the FunnyBirds framework [9] that introduces a synthetic dataset and a set of metrics for comparing explanation quality across different models, including post-hoc and interpretable ones.

However, a limitation of the FunnyBirds evaluation lies in how metrics are computed for the ProtoPNet model compared to other explanation methods such as GradCAM [25] and LRP [4]. The assumption made by the authors is that ProtoPNet explanations are presented as bounding boxes highlighting important image regions. However, these bounding boxes only approximate the significance of regions derived from more precise similarity maps, as illustrated in Figure 1, which can be seen as equivalent to saliency maps for post-hoc methods.

In this study, we evaluate ProtoPNet explanations based on similarity maps rather than bounding boxes within the FunnyBirds framework and comprehensively analyze the resulting changes in explanation quality. Our findings demonstrate that similarity map-based explanations better align the metrics with ProtoPNet's design intuition, yielding more accurate evaluation results. Therefore, we advocate for adopting similarity map-based activations for ProtoPNet evaluations to ensure a reliable comparison of explanations within the community.

INPUT TEST IMAGE

PROTOTYPICAL PART 1

PROTOTYPICAL PART 2

ProtoPNet's EXPLANATION

ProtoPNet's EXPLANATION

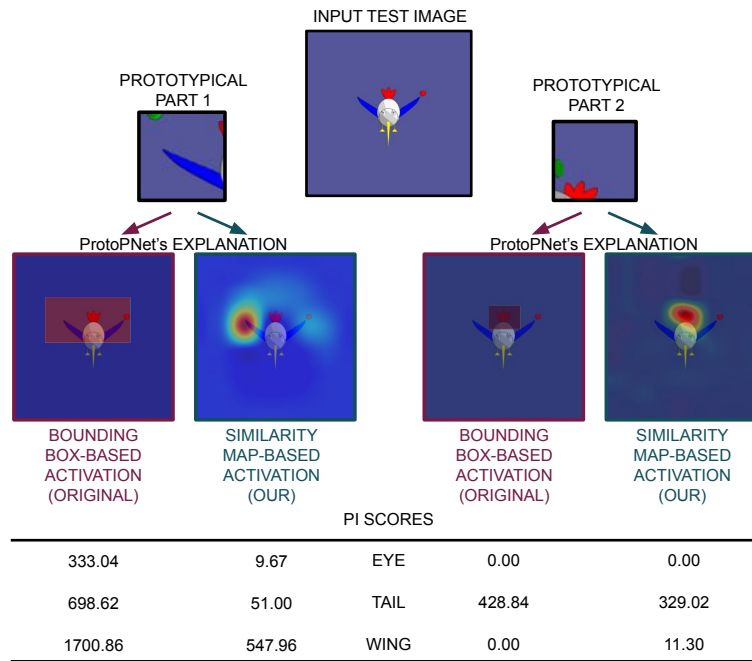| BOUNDING BOX-BASED ACTIVATION (ORIGINAL)	SIMILARITY MAP-BASED ACTIVATION (OUR)	PI SCORES	BOUNDING BOX-BASED ACTIVATION (ORIGINAL)	SIMILARITY MAP-BASED ACTIVATION (OUR)
333.04	9.67	EYE	0.00	0.00
698.62	51.00	TAIL	428.84	329.02
1700.86	547.96	WING	0.00	11.30

Fig. 1: Attribution Maps (AM) based on bounding boxes and similarity maps for two prototypical parts of ProtoPNet trained on the FunnyBirds dataset. For prototypical part 2, both AM types correctly cover the tail prototype. However, for prototypical part 1, AM based on bounding boxes incorrectly covers almost the whole area of the bird. Such discrepancy results in incorrect values of interface function PI (e.g. 333.04 instead of 0 for eyes) and inaccurate values of FunnyBirds metrics (see Section 3).

2   Related works

Evaluation of xAI. With the advancements in xAI methodologies, the need to quantify the quality of provided explanations has emerged. Benchmarking xAI approaches can be categorized into two main groups: those based on user studies and those involving the development of dedicated quantitative metrics.

Evaluation through user studies has been explored in previous research, e.g. in [11], the correctness of explanations was assessed, while [10] delved into determining the most suitable form of explanation for different data types. Additionally, in [12], the level of user overconfidence induced by explanations was measured. More specific evaluations include assessing semantic similarity for prototypical parts in [23], examining explanation saliency in [22], and evaluating the adequateness of prototypical parts for the medical domain in [16].

On the other hand, in proposing metrics and taxonomies for evaluating explanations, the Co-12 framework was introduced in [15,19,18]. This framework

provides a taxonomy for explanation evaluation and analyzes existing approaches such as Quantus [8], Ablation [7], and OpenXAI [2]. While these approaches predominantly focus on general toolkits for assessing explanation quality across multiple models and data modalities, there are also works proposing metrics specifically designed for prototypical parts, such as purity [17] and spatial misalignment [24].

However, recent work such as [9] aims to compare explanations among different methods using synthetic datasets. Nonetheless, the assumptions made for prototypical parts in this framework do not entirely align with the ProtoPNet essence. Thus, we propose a different method to derive them to ensure fair comparability with attribution-based methods.

3 Methods

3.1 FunnyBirds

Dataset. FunnyBirds dataset consists of synthetically generated bird images rendered from five human comprehensible concepts of beak, wings, feet, eyes, and tail, called parts. The dataset contains 50 bird classes, each corresponding to a unique subset of 26 predefined parts. In total, it comprises 50,000 training images and 5,000 testing images in 256×256 resolution. Furthermore, the training set incorporates augmented images with missing bird parts, simulating a data mix-up strategy.

Interface functions. The second major aspect of the framework are interface functions, $PI(\cdot)$ and $P(\cdot)$. These functions are designed to translate various explanation types (such as saliency maps or prototypical parts) into a unified format that can be used to calculate explainability metrics. Based on an explanation, the $PI(\cdot)$ function calculates a set of importance scores assigned to each part, while the $P(\cdot)$ function provides a set of important parts parameterized by the threshold t used to control the "sensitivity" of importance.

Default interface functions for prototypical parts. FunnyBirds authors introduce the default definition of interface functions for specific XAI methods. For prototypical part-based methods, they calculate $PI(\cdot)$ by summing the values of an attribution map within particular bird parts. The attribution map is obtained as follows for a training sample (x, y): the image $x \in X$ is passed to ProtoPNet; for each prototypical part corresponding to class y, we obtain a similarity map and corresponding bounding box; such a bounding box is then filled with the maximum value multiplied by the weight between the prototypical part and class y; the attribution map is obtained as a sum of such bounding boxes obtained for all prototypical parts. When it comes to $P(\cdot)$, it is defined as a set of bird parts with at least t-percent of area overlapping the union of those bounding boxes.
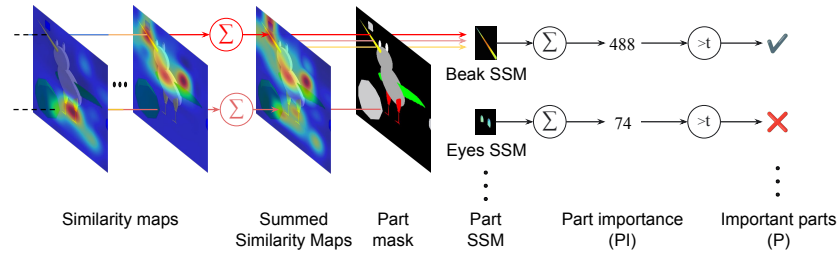
Fig. 2: Calculating Summed Similarity Maps (SSM) and interface functions $PI(\cdot)$ and $P(\cdot)$. The process starts with generating SSM by summarizing the similarity maps obtained for prototypical parts. Then, for each bird part, we multiply its mask with SMM and sum it up to obtain part importance (e.g. part importance for beak equals 488). To obtain important parts $P(\cdot)$, we analyze which of them has importance higher than the considered threshold t (e.g., eyes are not in $P(\cdot)$ because their importance 74 is smaller than the threshold). For this example, $PI = \{beak : 488, eyes : 74, legs : 371, \dots\}$ and $P = \{beak, legs, wings\}$.

Metrics. The FunnyBirds metrics design follows the Co-12 taxonomy [18], exploring categories such as Completeness, Correctness, and Contrastivity. The latter two are measured by Single Deletion (SD) and Target Sensitivity (TS) metrics, respectively. At the same time, the Completeness score is calculated as an average of Controlled Synthetic Data Check (CSDC), Preservation Check (PC), Deletion Check (DC), and Distractibility (D). Here, we recall the definition of one of the metrics, namely SD, to build an intuition on how the $PI(\cdot)$ and $P(\cdot)$ are used:

$$\text{SD} = \frac{1}{2} + \frac{1}{2|X|} \sum_{x \in X} \rho(\text{PI}(e), f(x) - \{f(x_{\backslash p})\}_p), \tag{1}$$

where e denotes the explanation received for image x, $f(x)$ is the logit of class y, and $f(x_{\backslash p})$ is the same logit obtained after removing part p of the bird. Finally, the ρ is the Spearman rank-order correlation between two sorted sets.

3.2   Summed Similarity Maps (SSM) for more precise interface functions

We propose an alternative definition of the interface functions based on the similarity maps, which are more precise than bounding boxes, as presented in Figure 1. Similarly, like in the default definition, $PI(\cdot)$ is calculated by summing the values of an attribution map within particular bird parts. However, our definition of attribution map differs as follows: the image $x \in X$ is passed to ProtoPNet; for each prototypical part corresponding to class y, we obtain a similarity map; such similarity map is then multiplied by the weight between the prototypical part and class y; the attribution map is obtained as a sum of

such similarity maps obtained for all prototypical parts. We call this approach Summed Similarity Maps (SSM).

Regarding $P(\cdot)$, we decided to reuse SSM. Therefore, for a given threshold t, a part is considered important if the sum of SSM pixels overlapping this part is larger than t-percentage of a total SSM sum. The calculation process is presented in Figure 2.

4   Experimental setup

We use the ProtoPNet model [6] with ResNet50, VGG19, and DenseNet169 backbones. We follow the training setup from FunnyBirds framework [9]. It corresponds to the multilabel classification because input images present incomplete birds fitting more than one class. We use Adam optimizer [13] with a learning rate decreasing every 10th epoch, and we apply prototype projection at the 25th epoch. Moreover, it is trained three times with different prototype sizes (128, 256, or 512) but with the same number of prototypical parts equal to 10. We do not use any augmentations.

The code is publicly available[5]. The training was conducted on four Nvidia A100 GPUs and took about 9 hours per model.

5   Results

5.1   Metrics scores for attribution maps based on bounding boxes or similarity maps

The FunnyBirds metrics design follows the Co-12 taxonomy [18], exploring categories such as Completeness, Correctness, and Contrastivity. The latter two are measured by Single Deletion (SD) and Target Sensitivity (TS) metrics, respectively. At the same time, the Completeness score is calculated as an average of Controlled Synthetic Data Check (CSDC), Preservation Check (PD), Deletion Check (DC), and Distractibility (D).

As presented in Table 3, we observe a notable enhancement in explanation correctness as the Single Deletion (SD) score increases from 0.24 to 0.73. As defined in 1, SD is computed as correlation between orders of PI(e) (GT) and $f(x) - \{f(x_{\setminus p})\}_p$ (BB or SSM). Therefore, a more precise SSM attribution map demonstrates that ProtoPNet is much more correct than reported in [9]. We explain this observation using examples in Figure 3. Moreover, a small increase is observed in its contrastivity, from 0.46 to 0.5.

Conversely, we observe a significant drop in three out of four completeness metrics. More precisely, the Controlled Synthetic Data Check (CSDC) drops from 0.93 to 0.58, the Preservation Check (PC) from 0.91 to 0.40, and the Deletion Check (DC) from 0.92 to 0.66. As we present in Figure 4, this drop is caused by the fact that the original BB approach tends to overidentify parts

---

[5] `https://github.com/hamer101/FunnyBirds_PrototypesRevisited`

Table 3: Metric scores obtained for two types of ProtoPNet with ResNet50 visualizations: bounding boxes (BB) and similarity maps (SSM). For SSM, we observe a notable enhancement in correctness and contrastivity but a drop in completeness. This is expected behavior for more precise explanations.

Co-12 category	Metric	BB (original)	SSM (ours)
	Accuracy	0.94	0.93±0.03
	BI	1.00	1.00±0.00
Completeness	CSDC	**0.93**	0.58±0.15
	PC	**0.91**	0.40±0.13
	DC	**0.92**	0.66±0.17
	D	0.58	**0.83±0.04**
Correctness	SD	0.24	**0.73±0.01**
Contrastivity	TS	0.46	**0.50±0.07**

as important, which results in an incorrectly high completeness score. In contrast, our SSM alternative generates more reliable P. Surprisingly, the remaining completeness metric, Distractibility (D), increases from 0.58 to 0.83. This phenomenon may be explained by the fact that D examines irrelevant parts while the remaining metrics concentrate on relevant ones.

These findings underscore the crucial role of visualization techniques within the FunnyBirds framework, particularly in ensuring consistency with other approaches.

5.2   Various backbones of ProtoPNet

Table 4 presents metrics scores depending on different backbone architectures (ResNet50, VGG19, and DenseNet169) used in ProtoPNet, while Figure 5 presents SSM obtained for them. Notably, ResNet50 exhibits substantially higher DC, D, and SD metrics than others, while its TS metric is the lowest. Conversely, for VGG19, CSDC and TS metrics demonstrate superiority. This discrepancy can be attributed to differences in receptive field sizes, notably smaller in the case of VGG19, and the incorporation of bottlenecks in ResNet50. However, it is important to note that while ResNet50 achieves the best metrics within the FunnyBirds framework, it is outperformed by DenseNet in terms of accuracy.

6   Conclusions

In this study, we evaluated ProtoPNet explanations based on similarity maps rather than bounding boxes within the FunnyBirds framework and comprehensively analyzed the resulting changes in explanation quality. Overall, the results indicate that the choice between bounding boxes and similarity maps significantly impacts the assessment of explanation quality, particularly for methods like ProtoPNet. While bounding boxes have been traditionally used for their

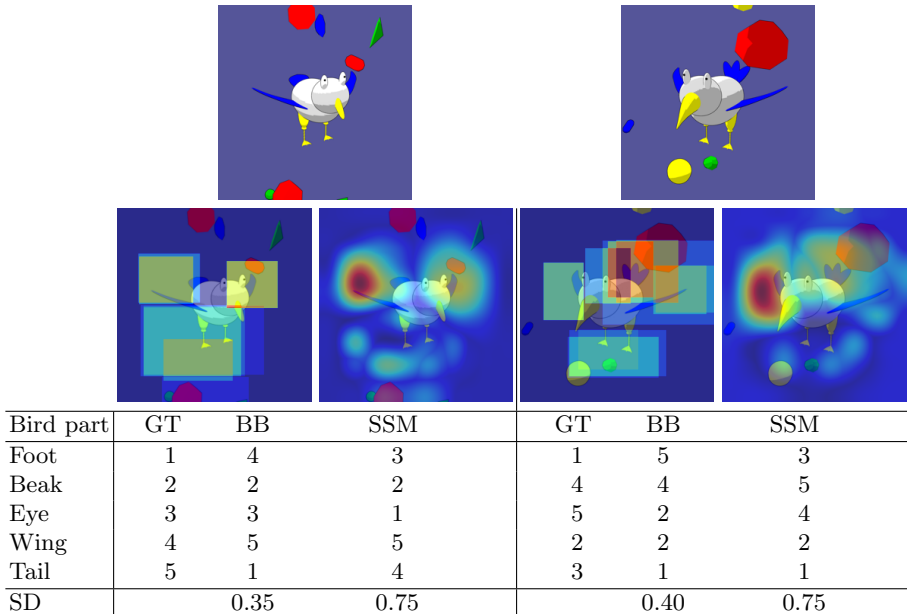| Bird part	GT	BB	SSM	GT	BB	SSM
Foot	1	4	3	1	5	3
Beak	2	2	2	4	4	5
Eye	3	3	1	5	2	4
Wing	4	5	5	2	2	2
Tail	5	1	4	3	1	1
SD		0.35	0.75		0.40	0.75

Fig. 3: Two sample images (top part), their attribution maps generated based on bounding boxes (BB) or similarity maps (SSM), and corresponding SD scores. As defined in 1, SD is computed as correlation between orders of $PI(e)$ (GT) and $f(x) - \{f(x_{\setminus p})\}_p$ (BB or SSM). We observe that a more precise SSM attribution map demonstrates that ProtoPNet is much more correct than reported in [9].

simplicity, our study demonstrates that similarity maps provide a more faithful representation of the underlying model's behavior, leading to more accurate evaluation metrics.

Furthermore, our investigation into different backbone architectures highlights the trade-off between interpretability and accuracy inherent in models like ProtoPNet. While models with higher interpretability, such as those based on ResNet50, may achieve lower accuracy, they offer more reliable explanations, as evidenced by higher metric scores. Conversely, models with higher accuracy, such as those based on DenseNet169, may sacrifice interpretability to some extent, resulting in slightly lower metric scores.

In conclusion, our study underscores the importance of carefully considering visualization techniques and model architectures in evaluating explainable AI methods like ProtoPNet. By adopting more precise visualization methods and understanding the trade-offs between interpretability and accuracy, researchers and practitioners can make more informed decisions when deploying and evaluating such models in real-world applications.

Limitations. While our study utilizes code provided by the authors of the FunnyBirds framework, it is worth noting that the experiments were conducted on
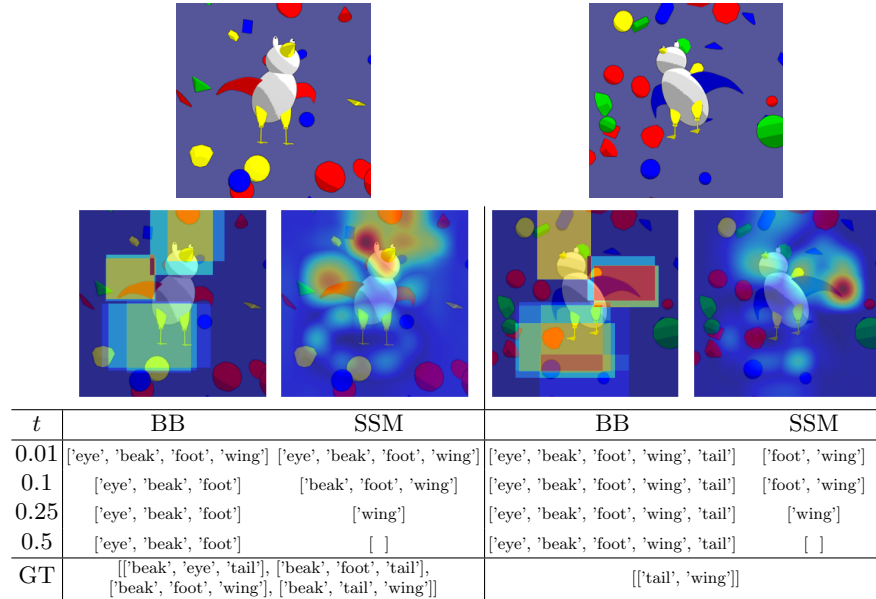
t	BB	SSM	BB	SSM
0.01	['eye', 'beak', 'foot', 'wing']	['eye', 'beak', 'foot', 'wing']	['eye', 'beak', 'foot', 'wing', 'tail']	['foot', 'wing']
0.1	['eye', 'beak', 'foot']	['beak', 'foot', 'wing']	['eye', 'beak', 'foot', 'wing', 'tail']	['foot', 'wing']
0.25	['eye', 'beak', 'foot']	['wing']	['eye', 'beak', 'foot', 'wing', 'tail']	['wing']
0.5	['eye', 'beak', 'foot']	[ ]	['eye', 'beak', 'foot', 'wing', 'tail']	[ ]
GT	[['beak', 'eye', 'tail'], ['beak', 'foot', 'tail'], ['beak', 'foot', 'wing'], ['beak', 'tail', 'wing']]		[['tail', 'wing']]	

Fig. 4: Two sample images (top part), their attribution maps generated based on bounding boxes (BB) or similarity maps (SSM), and important parts (P) obtained for various values of t. We observe that the original BB approach tends to overidentify parts as important, which results in an incorrectly high completeness score. In contrast, our SSM alternative generates more reliable P. Notice that the GT row corresponds to the sets of truly important parts, and the completeness is high if P is similar to one of those sets.

models we trained because the repository did not contain the training code while we prepared this work. This discrepancy could lead to slight variations in results due to model differences. Nevertheless, we tried to mitigate this by reporting metrics scores averaged over multiple runs.

Impact. This research addresses the challenge of evaluating different eXplainable Artificial Intelligence (XAI) methods, particularly comparing post-hoc and ante-hoc approaches. Leveraging a recently published framework, we emphasize the importance of unifying approaches for deriving metric scores and advocate for using similarity map-based explanations of prototypical parts when evaluating and comparing them with saliency-based methods.

References

1. Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., Kim, B.: Sanity checks for saliency maps. Advances in neural information processing systems **31** (2018)

Table 4: Metrics scores depending on different backbone architectures (ResNet50, VGG19, and DenseNet169) reveal notable differences. Specifically, ResNet obtains higher explanation metric scores but lower accuracy. It shows the tradeoff between interpretability and accuracy in ProtoPNet.

Co-12 category	Metric	ResNet50	VGG19	DenseNet169
	Accuracy	0.93±0.03	0.96±0.00	0.97±0.01
	BI	1.00±0.00	0.99±0.01	0.98±0.01
Completeness	CSDC	0.58±0.15	**0.63±0.06**	0.58±0.04
	PC	0.40±0.13	**0.48±0.07**	0.36±0.06
	DC	**0.66±0.17**	0.61±0.10	0.49±0.10
	D	**0.83±0.04**	0.76±0.01	0.79±0.01
Correctness	SD	**0.73±0.01**	0.48±0.09	0.49±0.03
Contrastivity	TS	0.50±0.07	**0.80±0.05**	0.67±0.04

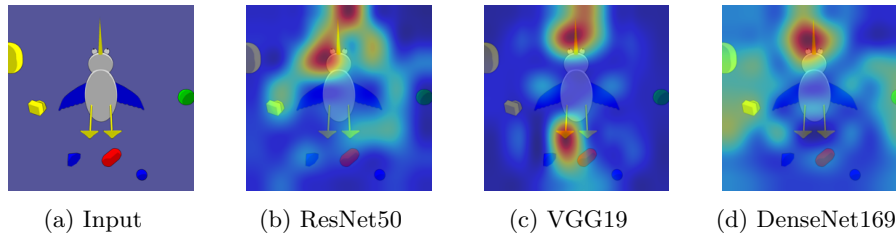(a) Input   (b) ResNet50   (c) VGG19   (d) DenseNet169

Fig. 5: Sample image (a) and SSMs obtained for ProtoPNet with various backbones (b-d).

2. Agarwal, C., Queen, O., Lakkaraju, H., Zitnik, M.: Evaluating explainability for graph neural networks. Scientific Data **10**(144) (2023), https://www.nature.com/articles/s41597-023-01974-x
3. Arras, L., Osman, A., Samek, W.: Clevr-xai: A benchmark dataset for the ground truth evaluation of neural network explanations. Information Fusion **81**, 14–40 (2022)
4. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015)
5. Böhle, M., Fritz, M., Schiele, B.: B-cos networks: Alignment is all we need for interpretability. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 10329–10338 (2022)
6. Chen, C., Li, O., Tao, D., Barnett, A., Rudin, C., Su, J.K.: This looks like that: deep learning for interpretable image recognition. Advances in neural information processing systems **32** (2019)
7. Hameed, I., Sharpe, S., Barcklow, D., Au-Yeung, J., Verma, S., Huang, J., Barr, B., Bruss, C.B.: BASED-XAI: Breaking Ablation Studies Down for Explainable Artificial Intelligence. In: Workshop on Machine Learning in Finance (2022)
8. Hedström, A., Weber, L., Krakowczyk, D., Bareeva, D., Motzkus, F., Samek, W., Lapuschkin, S., Höhne, M.M.M.: Quantus: An explainable ai toolkit for responsible evaluation of neural network explanations and beyond. Journal of Machine Learning Research **24**(34), 1–11 (2023), http://jmlr.org/papers/v24/22-0142.html

9. Hesse, R., Schaub-Meyer, S., Roth, S.: Funnybirds: A synthetic vision dataset for a part-based analysis of explainable ai methods. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 3981–3991 (2023)

10. Jeyakumar, J.V., Noor, J., Cheng, Y.H., Garcia, L., Srivastava, M.: How can i explain this to you? an empirical study of deep neural network explanation methods. Advances in Neural Information Processing Systems **33**, 4211–4222 (2020)

11. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., et al.: Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In: International conference on machine learning. pp. 2668–2677. PMLR (2018)

12. Kim, S.S., Meister, N., Ramaswamy, V.V., Fong, R., Russakovsky, O.: Hive: Evaluating the human interpretability of visual explanations. In: European Conference on Computer Vision. pp. 280–298. Springer (2022)

13. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2017)

14. Komorowski, P., Baniecki, H., Biecek, P.: Towards evaluating explanations of vision transformers for medical imaging. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3725–3731 (2023)

15. Le, P.Q., Nauta, M., Nguyen, V.B., Pathak, S., Schlötterer, J., Seifert, C.: Benchmarking explainable ai: a survey on available toolkits and open challenges. In: International Joint Conference on Artificial Intelligence (2023)

16. Nauta, M., Hegeman, J.H., Geerdink, J., Schlötterer, J., Keulen, M.v., Seifert, C.: Interpreting and correcting medical image classification with pip-net. In: European Conference on Artificial Intelligence. pp. 198–215. Springer (2023)

17. Nauta, M., Schlötterer, J., van Keulen, M., Seifert, C.: Pip-net: Patch-based intuitive prototypes for interpretable image classification. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2744–2753 (2023)

18. Nauta, M., Seifert, C.: The co-12 recipe for evaluating interpretable part-prototype image classifiers. In: World Conference on Explainable Artificial Intelligence. pp. 397–420. Springer (2023)

19. Nauta, M., Trienes, J., Pathak, S., Nguyen, E., Peters, M., Schmitt, Y., Schlötterer, J., van Keulen, M., Seifert, C.: From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable ai. ACM Computing Surveys **55**(13s), 1–42 (2023)

20. Rudin, C.: Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. Nature machine intelligence **1**(5), 206–215 (2019)

21. Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., Zhong, C.: Interpretable machine learning: Fundamental principles and 10 grand challenges. Statistic Surveys **16**, 1–85 (2022)

22. Rymarczyk, D., Struski, Ł., Górszczak, M., Lewandowska, K., Tabor, J., Zieliński, B.: Interpretable image classification with differentiable prototypes assignment. In: European Conference on Computer Vision. pp. 351–368. Springer (2022)

23. Rymarczyk, D., Struski, Ł., Tabor, J., Zieliński, B.: Protopshare: Prototypical parts sharing for similarity discovery in interpretable image classification. In: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining. pp. 1420–1430 (2021)

24. Sacha, M., Jura, B., Rymarczyk, D., Struski, Ł., Tabor, J., Zieliński, B.: Interpretability benchmark for evaluating spatial misalignment of prototypical parts explanations. arXiv preprint arXiv:2308.08162 (2023)

25. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE international conference on computer vision. pp. 618–626 (2017)
26. Tomsett, R., Harborne, D., Chakraborty, S., Gurram, P., Preece, A.: Sanity checks for saliency metrics. In: Proceedings of the AAAI conference on artificial intelligence. vol. 34, pp. 6021–6029 (2020)