FunnyBirds: A Synthetic Vision Dataset for a Part-Based Analysis of Explainable AI Methods

– Supplemental Material –

Robin Hesse[1] Simone Schaub-Meyer[1,2] Stefan Roth[1,2]

[1]Department of Computer Science, TU Darmstadt [2]hessian.AI

A. Overview

This appendix provides additional information, framework results (Tab. 2), qualitative results (Figs. 7 and 8), and experimental details for reproducibility purposes, which could not be included in the main text due to space limitations.

B. Interface Functions

Our proposed interface functions have to be instantiated individually for each explanation type. In our evaluation, we examine four explanation types:

1. *Attribution maps*, *i.e.*, Input×Gradient [51], (absolute) Integrated Gradients [54], Grad-CAM [48], RISE [40], $\mathcal{X}$-DNN [25], BagNet [6], B-cos networks [8], and Chefer LRP [10].
2. *Attention from vision transformers* [16], *i.e.*, Rollout [1].
3. *Binary importance maps* that indicate if a pixel is important or not, *i.e.*, LIME [43].
4. *Prototypes* that allow for explanations of the type "this looks like that," *i.e.*, ProtoPNet [11].

In the following, we will outline how interface functions for these explanation types are instantiated:

Attribution maps:

PI$(\cdot)$ – The part importance is estimated by summing the pixelwise attribution scores within each part, where the part mask is dilated with a small square kernel of size 5×5 to also include the part's edges.

P$(\cdot)$ – A part is considered to be important if its part importance is more than $t\%$ of the total attribution sum of the explanation. For a description of how t is chosen, please refer to Sec. 4.1.2 of the main paper.

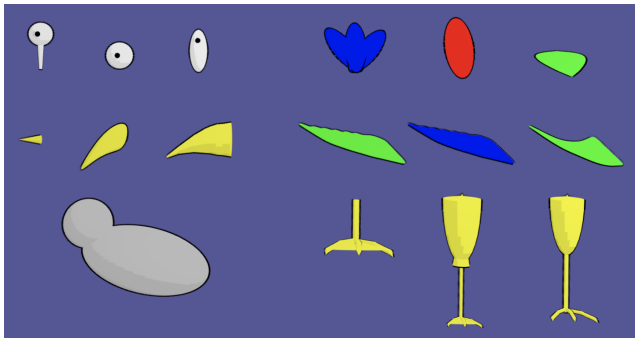

Figure 6. Example models of the neutral *body* (bottom left) and our dataset concepts (*i.e.*, parts) *beak*, *wings*, *feet*, *eyes*, and *tail*.

Attention:

PI$(\cdot)$ – The part importance is estimated by summing the pixelwise attribution scores within each part, where the part mask is dilated with a small square kernel of size 5×5 to also include the part's edges. Since attention rollout [1] cannot be computed w.r.t. a particular target class [10], the target sensitivity protocol cannot be computed.

P$(\cdot)$ – A part is considered to be important if its part importance is more than $t\%$ of the total attribution sum of the explanation.

Binary importance maps:

PI$(\cdot)$ – The part importance is not defined for this explanation type.

P$(\cdot)$ – A part is considered to be important if $t\%$ of its pixels are estimated to be important by the explanation.

Prototypes:

PI$(\cdot)$ – The importance score within the bounding box of a single prototype is the product of its similarity score

and its class connection score. The importance score outside of the bounding box is zero. The final part importance is estimated by summing the importance scores of each prototype belonging to the class of interest.

P$(\cdot)$ – A part is considered to be important if $t\%$ of its pixels overlap with the explanation's bounding boxes.

Qualitative results for P$(\cdot)$ can be seen in Fig. 9.

C. Evaluation Details

C.1. Accuracy and background independence

In the main paper, we describe the accuracy (A) and background independence (BI) metrics only textually. We here provide the accompanying formulas. The notation follows that of Sec. 4.1 in the main paper.

The accuracy A denotes the standard classification accuracy:

$$\mathrm{A} = \frac{1}{N}\sum_{n=1}^{N}\left[f(x_n) = c_n\right], \tag{1}$$

with $[\cdot]$ denoting the Iverson bracket [29].

The background independence BI denotes the ratio of background objects that, when removed, cause the model output to drop less than 5%:

$$\mathrm{BI} = \frac{1}{NB}\sum_{n=1}^{N}\left[|0.05f(x_n)| > f(x_n) - \{f(x_n'''')\}\right], \tag{2}$$

with B denoting the average number of background objects per image, and $\{f(x_n'''')\}$ the target scores of the images obtained by removing any individual background object from image x_n.

C.2. Other dimensions of evaluation

In our FunnyBirds framework, we analyze the explainability dimensions *completeness*, *correctness*, and *contrastivity*. With our custom evaluations we evaluate the *coherence* of methods. Nauta *et al*. [35] propose various additional dimensions for evaluating XAI that have been studied in related work but are not considered in our paper. We here describe the reasons for not including these dimensions; please refer to [35] for a definition of each dimension:

Consistency. Consistency received only minor attention in related work.

Continuity. Continuity received only moderate attention in related work. Additionally, we believe that the continuity of an explanation is usually strongly tied to the continuity of the model. Nevertheless, continuity could be included in our framework by measuring the part importance differences for two similar input images, *e.g*., two images with slight viewpoint or illumination changes. We leave this for future work.

Compactness. We believe that there is a discrepancy between the automatically measurable size of an explanation and the size of an explanation that is perceived by a human. For example, an attribution map contains a lot of information (in the sense of memory size, *e.g*., in MiB) that, however, is much easier for a human to parse than, *e.g*., a complex mathematical function that requires only little memory to store. For this reason, we do not believe that compactness of different explanation types can be sufficiently evaluated without a human in the loop.

Covariate complexity. Covariate complexity received only moderate attention in related work. Additionally, just as compactness, it is strongly related to a human assessment, and thus, not qualified for our fully automatic framework. However, one could develop custom evaluations that measure the covariate complexity of specific methods.

Composition. Composition received only minor attention in related work.

Confidence. Confidence received only minor attention in related work.

Context. Context received only minor attention in related work.

Controllability. Controllability received only minor attention in related work.

D. Experimental Details

All examined models are initialized with weights obtained by pre-training on ImageNet [70]. If not specified otherwise, we use the hyper-parameters from the original implementation of each respective model. To ensure that test images with removed parts are from the same distribution as the training data, we augment half of our training set by randomly removing $n \in \{0, \ldots, 5\}$ bird parts from each image. Since these images can no longer be distinctly associated with one specific class, we utilize a multi-label classification training scheme, where we compute the average cross-entropy loss for all potential targets, *i.e*., all the classes that contain all the remaining parts. For training, we use a single NVIDIA A100-SXM4-40GB GPU. We train each model twice and select the run with the higher test set accuracy for our evaluation.

ResNet-50. To train ResNet-50 [22], we use a batch size of 64 and an SGD optimizer with a weight-decay of $1e{-}4$,

Table 2. *Quantitative results of our FunnyBirds evaluation protocols.* See Sec. 4.1 of the main paper for a description of the evaluation metrics. We additionally report the final scores for each respective explainability dimension completeness (Com.), correctness (Cor.), and contrastivity (Con.). The mean explainability score (mX) denotes the mean of the final completeness, correctness, and contrastivity scores. Note that A (Acc.), BI (B.I.), Com., Cor., Con., and mX are also included in Fig. 3 of the main paper. * denotes slight architectural changes.

Backbone	XAI Method	A	BI	CSDC	PC	DC	D	SD	TS	Com.	Cor.	Con.	mX
VGG16 [53]	IG [54]	0.99	0.99	0.92	0.92	0.92	0.97	0.67	0.92	0.95	0.67	0.92	0.85
VGG16 [53]	IG abs. [54]	0.99	0.99	0.96	0.99	0.97	0.97	0.69	0.84	0.97	0.69	0.84	0.83
VGG16 [53]	RISE [40]	0.99	0.99	0.8	0.73	0.7	0.84	0.73	0.83	0.79	0.73	0.83	0.78
VGG16 [53]	LIME [43]	0.99	0.99	0.89	0.88	0.9	0.92	0	0	0.91	0	0	0.3
VGG16 [53]	IxG [51]	0.99	0.99	0.79	0.71	0.69	0.94	0.55	0.69	0.84	0.55	0.69	0.69
VGG16 [53]	Grad-CAM [48]	0.99	0.99	0.94	0.97	0.93	0.87	0.75	0.93	0.91	0.75	0.93	0.86
ResNet-50 [22]	IG [54]	1	1	0.92	0.94	0.88	0.81	0.59	0.98	0.86	0.59	0.98	0.81
ResNet-50 [22]	IG abs. [54]	1	1	0.95	0.97	0.91	0.79	0.53	0.86	0.87	0.53	0.86	0.75
ResNet-50 [22]	RISE [40]	1	1	0.82	0.75	0.74	0.63	0.56	0.61	0.7	0.56	0.61	0.62
ResNet-50 [22]	LIME [43]	1	1	0.94	0.94	0.92	0.78	0	0	0.86	0	0	0.29
ResNet-50 [22]	IxG [51]	1	1	0.74	0.61	0.53	0.54	0.54	0.8	0.58	0.54	0.8	0.64
ResNet-50 [22]	Grad-CAM [48]	1	1	0.8	0.74	0.69	0.74	0.55	0.78	0.74	0.55	0.78	0.69
ViT-B/16 [16]	IG [54]	0.98	1	0.89	0.86	0.85	0.9	0.65	0.91	0.88	0.65	0.91	0.82
ViT-B/16 [16]	IG abs. [54]	0.98	1	0.96	0.98	0.95	0.89	0.63	0.74	0.92	0.63	0.74	0.76
ViT-B/16 [16]	RISE [40]	0.98	1	0.79	0.71	0.7	0.83	0.79	0.75	0.78	0.79	0.75	0.77
ViT-B/16 [16]	LIME [43]	0.98	1	0.95	0.96	0.96	0.85	0	0	0.9	0	0	0.3
ViT-B/16 [16]	IxG [51]	0.98	1	0.74	0.59	0.6	0.43	0.51	0.67	0.54	0.51	0.67	0.57
ViT-B/16 [16]	Grad-CAM [48, 10]	0.98	1	0.75	0.67	0.68	0.91	0.7	0.48	0.81	0.7	0.48	0.66
ViT-B/16 [16]	Rollout [1]	0.98	1	0.86	0.8	0.82	0.8	0.76	0	0.81	0.76	0	0.52
ViT-B/16 [16]	Chefer LRP [10]	0.98	1	0.91	0.92	0.89	0.9	0.74	0.95	0.9	0.74	0.95	0.86
BagNet [6]	BagNet [6]	1	1	0.95	0.98	0.91	0.91	0.76	0.99	0.93	0.76	0.99	0.9
ResNet-50* [22]	B-cos network [8]	0.96	0.87	0.93	0.88	0.94	0.86	0.69	0.89	0.89	0.69	0.89	0.82
ResNet-50* [22]	$\mathcal{X}$-DNN [25]	0.99	1	0.9	0.88	0.85	0.93	0.6	0.87	0.91	0.6	0.87	0.79
ResNet-50 [22]	ProtoPNet [11]	0.94	1	0.93	0.91	0.92	0.58	0.24	0.46	0.75	0.24	0.46	0.48

a momentum of 0.9, and a learning rate of 0.1; we train for 120 epochs with a learning rate scheduler that multiplies the initial learning rate with a factor of 0.1 after 60 epochs.

VGG16. To train the VGG16 [53] model, we use the same training setup as for the ResNet-50 model with an initial learning rate of 0.001.

ViT-B/16. To train the ViT-B/16 [16] model, we use the same training setup as for the ResNet-50 model with an initial learning rate of 0.01.

$\mathcal{X}$-DNN. To train the $\mathcal{X}$-DNN [25] model, we use the same training setup as for the ResNet-50 model with an initial learning rate of 0.01.

BagNet. To train the BagNet [6] model, we use the same training setup as for the ResNet-50 model with an initial learning rate of 0.01. Our instantiation of BagNet uses a receptive field of 33×33.

B-cos network. To train the B-cos network [8], we use the same training setup as for the ResNet-50 model with an initial learning rate of 0.01. As recommended in the original paper [8], we use a binary cross-entropy loss, and we concatenate the input x' with its complement, giving us the final input $x = [x', 1 - x']$.

ProtoPNet. To train ProtoPNet [11], we use the same hyper-parameters as in the original paper, *i.e.*, a batch size of 80, a learning rate of $1e{-}4$ for the features and $3e{-}3$ for the add-on layers and prototype vectors, 100 training epochs, and a learning rate decay factor of 0.1 after every 5 epochs.

D.1. Dataset generation

Our proposed dataset consists of rendered 3D scenes, as shown in Fig. 8. The required bird parts are manually modeled using Blender.[1] To render the scenes we use Three.js, a JavaScript 3D Library.[2] For our proposed toon shading, we use *MeshToonMaterial*.[3] In order to add shadows and achieve a 3D effect, we add a point light source to the scene. We empirically validated that an image with all bird parts removed cannot be classified beyond random guessing, to ensure that the background contains *no* class-specific information.

[1] blender.org
[2] threejs.org
[3] threejs.org – MeshToonMaterial

D.2. Stability across runs

To measure the stability of our FunnyBirds framework across runs, we report the absolute difference of two runs in Tab. 3. We report results for two setups: *(1)* the absolute difference between the evaluation on two training runs (*i.e.*, trained with differing random seeds) and *(2)* the absolute difference between evaluating the respective model from the main paper on the original test set (500 samples) and a larger test set with 2 500 samples. This allows us to measure the stability across different training runs and across different test set sizes. The absolute difference between different training runs is fairly small ($\leq$ 0.039 for mX, see Tab. 3). CNN-based architectures appear to be more stable than the vision transformer. Also, the absolute difference of the explainability protocols across runs is somewhat correlated with the absolute difference of the accuracy across runs. This suggests that the fluctuation of the accuracy across different training runs is a good proxy for the stability of the explanation protocols. This may be due to models with similar accuracy learning similar functions, and thus, providing similar explanations.

The absolute difference between evaluating on the original test set and on a larger test set is even smaller ($\leq$ 0.009 for mX), indicating that the proposed dataset size (500 images) is sufficiently large. We purposely did not use the larger test set for the principal evaluation in the main paper to keep the computational expense at bay and allow for an easy adoption of our analysis framework in future work. Note that for slower explanation methods like RISE [40], evaluating 2 500 images would take $\sim$ 50h on an NVIDIA A100-SXM4-40GB GPU, which would impair the practicability of our proposed framework. Nevertheless, we will also publish the larger test set for evaluation under these conditions. To conclude, we find that our framework is quite stable under different training runs and that our test set size is sufficiently large.

References

[70] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, pages 248–255, 2009. 2

Table 3. *Stability of our evaluation protocols.* Scores indicate the absolute difference of two runs. See Sec. 4.1 of the main paper for a description of the metrics. The mean explainability score (mX) denotes the mean of the final completeness, correctness, and contrastivity scores. We report results for the stability across *(1)* two training runs and *(2)* between the original and a larger test set (see Appendix D.2).

Setup	Backbone	XAI Method	$\lvert\Delta A\rvert$	$\lvert\Delta BI\rvert$	$\lvert\Delta CSDC\rvert$	$\lvert\Delta PC\rvert$	$\lvert\Delta DC\rvert$	$\lvert\Delta D\rvert$	$\lvert\Delta SD\rvert$	$\lvert\Delta TS\rvert$	$\lvert\Delta mX\rvert$
(1)	VGG16 [53]	IG [54]	0.002	0.003	0.008	0.008	0.008	0.001	0.015	0.005	0.005
(1)	ResNet-50 [22]	IG [54]	0	0.001	0	0.02	0.006	0	0.022	0.002	0.008
(1)	ViT-B/16 [16]	IG [54]	0.01	0	0.051	0.11	0.104	0.016	0.031	0.037	0.035
(1)	VGG16 [53]	IxG [53]	0.002	0.003	0.005	0.002	0.066	0.012	0.019	0.071	0.023
(1)	ResNet-50 [22]	IxG [53]	0	0.001	0.02	0.012	0.032	0.031	0.003	0.015	0.003
(1)	ViT-B/16 [16]	IxG [53]	0.01	0	0.066	0.12	0.126	0.026	0.017	0.06	0.039
(2)	VGG16 [53]	IG [54]	0.0128	0.001	0.008	0.027	0.004	0.013	0	0.007	0.001
(2)	ResNet-50 [22]	IG [54]	0.0124	0	0.008	0.079	0.003	0.029	0.032	0.005	0.009
(2)	ViT-B/16 [16]	IG [54]	0.0232	0	0.008	0.082	0.02	0.017	0.016	0.008	0.006
(2)	VGG16 [53]	IxG [53]	0.0128	0.001	0.004	0.054	0.001	0	0.006	0.013	0.001
(2)	ResNet-50 [22]	IxG [53]	0.0124	0	0.011	0.079	0.03	0.005	0.009	0.005	0.007
(2)	ViT-B/16 [16]	IxG [53]	0.0232	0	0.039	0.106	0.053	0.058	0.004	0	0.003

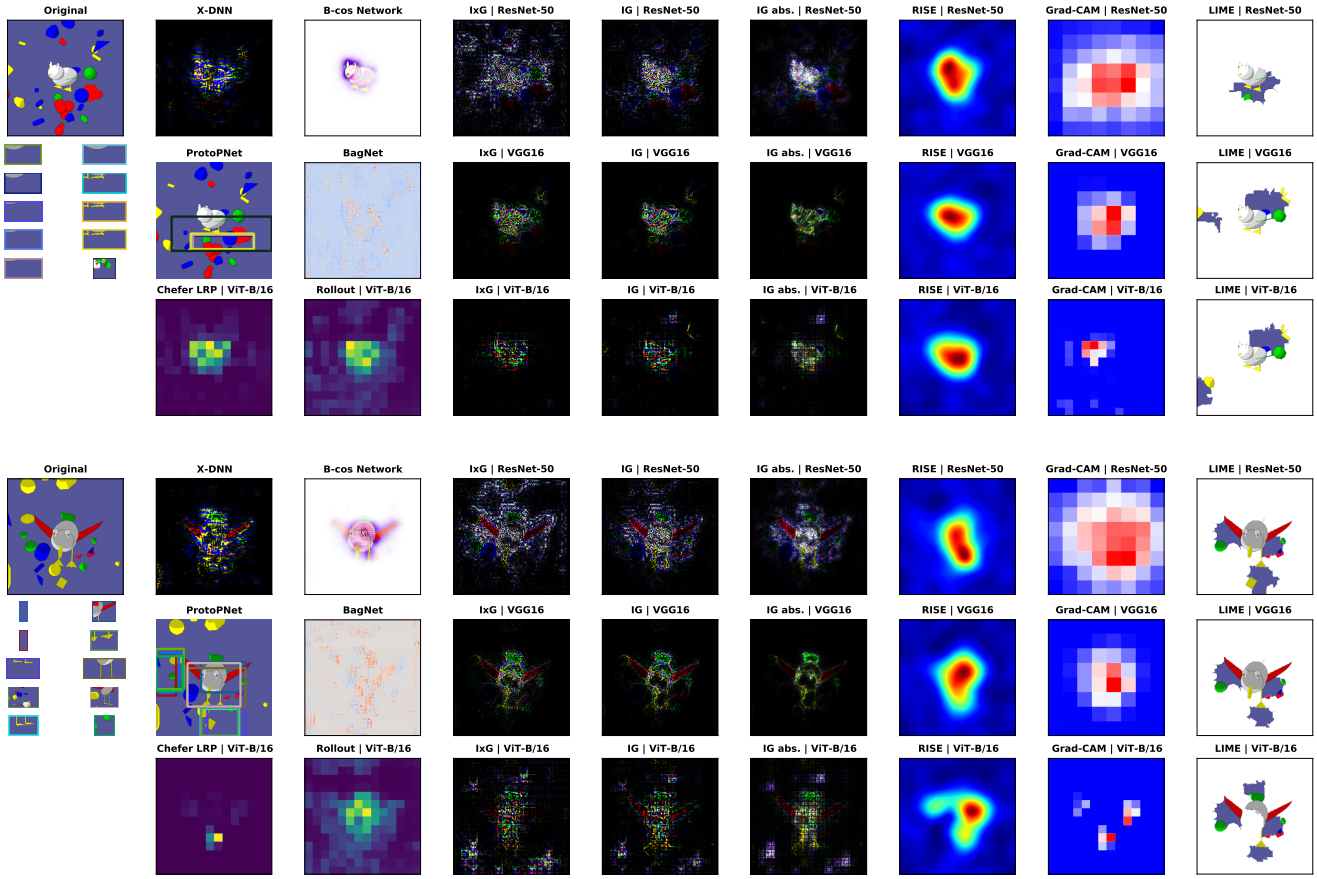

Figure 7. *Additional qualitative results for the examined explanation methods.* Each group of three rows shows results for the same input image and all respective XAI methods and backbones that have been examined in our *FunnyBirds* framework. The displayed qualitative results are consistent with the qualitative results in Fig. 3 from the main paper.

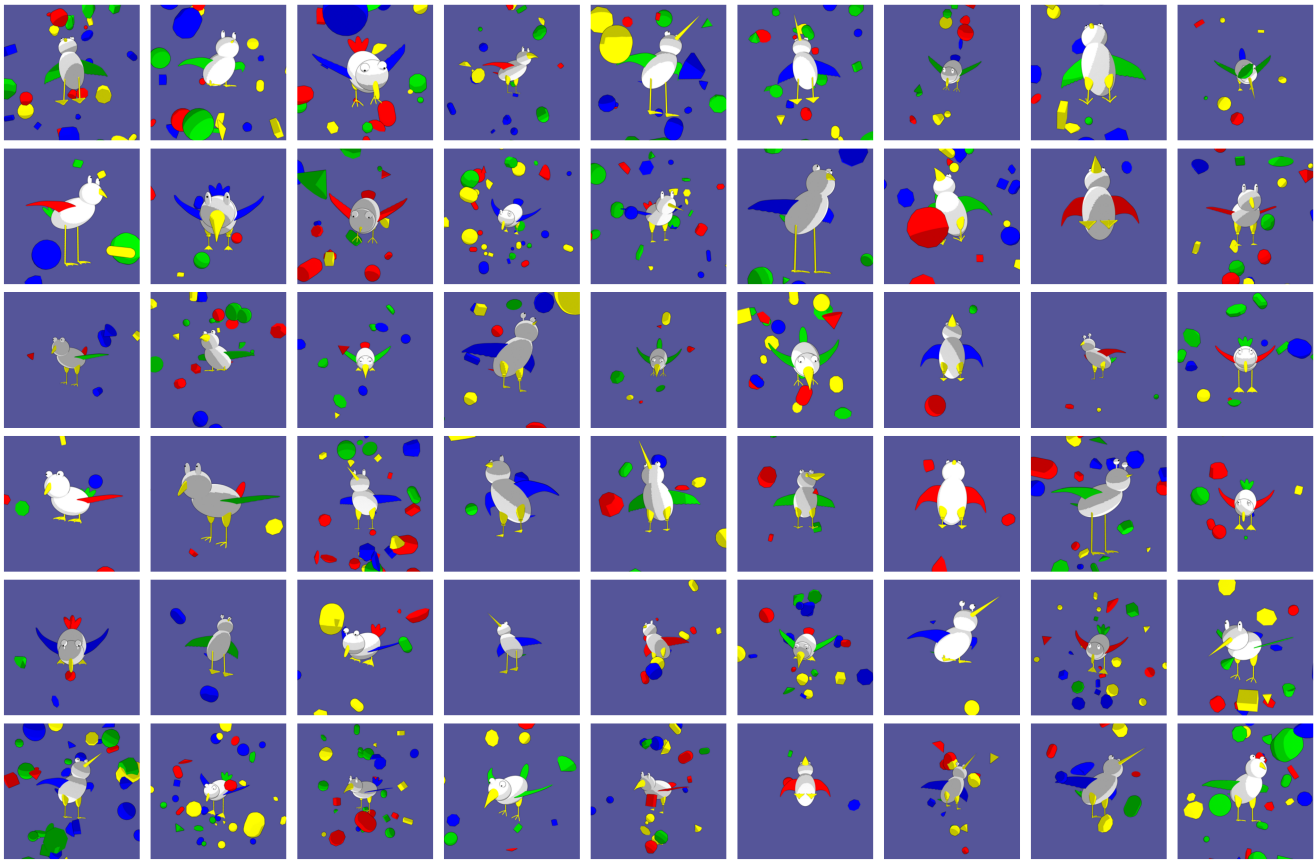

Figure 8. *Example images from our FunnyBirds dataset.*

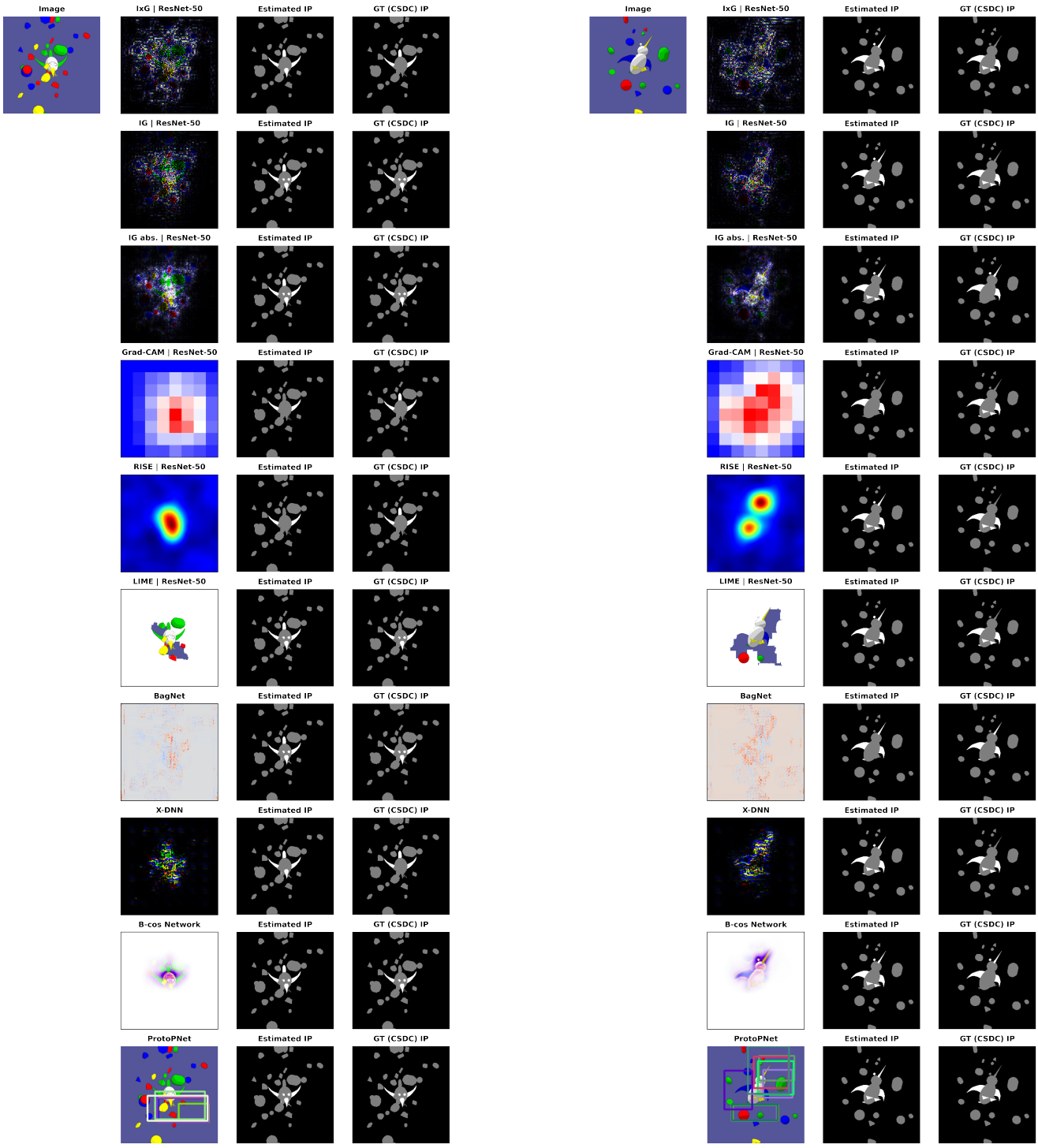

Figure 9. *Explanations and the extracted important parts.* The left column of each block shows the original input image. Next, we show the explanation from each respective XAI method. This is followed by the estimated important parts (estimated IP – highlighted in white) from the explanation using our interface function $\mathrm{P}(\cdot)$ with a threshold $t = 0.02$. In the last column, we show the ground-truth minimal important parts from the *controlled synthetic data check* protocol (GT (CSDC) IP). For example, the parts estimated to be important by Grad-CAM [48] in the left block are not fully complete, since fewer parts are highlighted than for GT (CSDC) IP.