

EPIC: Explanation of Pretrained Image Classification Networks via Prototypes

Piotr Borycki^{1,2}, Magdalena Trędownicz^{1,2,5}, Szymon Janusz^{1,2,5}, Jacek Tabor^{1,5},
Przemysław Spurek^{1,3}, Arkadiusz Lewicki^{4,5}, Łukasz Struski^{1,5}

¹ Jagiellonian University, Faculty of Mathematics and Computer Science

² Jagiellonian University, Doctoral School of Exact and Natural Sciences

³ IDEAS Research Institute

⁴ University of Information Technology and Management, Faculty of Applied Computer Science, Rzeszów

⁵ Prometheus MedTech.AI

Abstract

Explainable AI (XAI) methods generally fall into two categories. Post-hoc approaches generate explanations for pre-trained models and are compatible with various neural network architectures. These methods often use feature importance visualizations, such as saliency maps, to indicate which input regions influenced the model’s prediction. Unfortunately, they typically offer a coarse understanding of the model’s decision-making process. In contrast, ante-hoc (inherently explainable) methods rely on specially designed model architectures trained from scratch. A notable subclass of these methods provides explanations through prototypes, representative patches extracted from the training data. However, prototype-based approaches require dedicated architectures, involve specialized training procedures, and perform well only on specific datasets. In this work, we propose EPIC (Explanation of Pretrained Image Classification), a novel approach that bridges the gap between these two paradigms. Like post-hoc methods, EPIC operates on pre-trained models without architectural modifications. Simultaneously, it delivers intuitive, prototype-based explanations inspired by ante-hoc techniques. To the best of our knowledge, EPIC is the first post-hoc method capable of fully replicating the core explanatory power of inherently interpretable models. We evaluate EPIC on benchmark datasets commonly used in prototype-based explanations, such as CUB-200-2011 and Stanford Cars, alongside large-scale datasets like ImageNet, typically employed by post-hoc methods. EPIC uses prototypes to explain model decisions, providing a flexible and easy-to-understand tool for creating clear, high-quality explanations.

Code — <https://github.com/piotr310100/EPIC>

Extended version — <https://arxiv.org/pdf/2505.12897>

1 Introduction

Deep neural networks (DNNs) have revolutionized predictive modeling, frequently achieving performance superior to human experts in numerous fields (He et al. 2016). However, despite their impressive results, DNNs are frequently regarded as “black boxes” due to their lack of clear interpretability (Lipton 2018). This lack of transparency has led to the fast development of explainable AI (XAI) methods,

which aim to make accurate predictions easier for people to understand (Xu et al. 2019).

Broadly, XAI methods fall into two categories: post-hoc approaches and ante-hoc (inherently interpretable) models. Post-hoc methods apply explanation techniques to pre-trained architectures without altering their internal mechanisms. Widely adopted examples include SHAP (Lundberg and Lee 2017), LIME (Ribeiro, Singh, and Guestrin 2016), LRP (Bach et al. 2015), and Grad-CAM (Selvaraju et al. 2020), all of which rely on various notions of feature importance, often visualized through saliency maps. However, while saliency maps highlight input regions contributing to predictions, they frequently fall short in providing causal or concept-level insights. As a result, they may confirm where the model is looking, but not why it arrives at a particular decision, see Fig. 1.

In contrast, ante-hoc (inherently explainable) models embed interpretability directly into their architectures, producing explanations as part of the prediction process. ProtoPNet (Chen et al. 2019), a seminal example, introduced class-specific prototypes that enable explanations by comparing input image patches with prototypical parts drawn from the training data. Building on this idea, PIPNet (Nauta, Sieb, and van Gemert 2023) introduced architectural and training innovations to explicitly disentangle feature channels, ensuring that each channel consistently encodes a distinct prototype. More recently, InfoDisent (Struski, Rymarczyk, and Tabor 2024) leveraged a pre-trained backbone but disentangled the final layer through a modified classification head, enabling interpretable outputs with only finetuning the classification head. Although ante-hoc models offer significant advantages, they encounter two fundamental challenges. First, they typically require specialized architectures and custom training regimes, demanding substantial engineering effort and computational resources. Second, they cannot be added to models that are already in use, especially if the original training data is unavailable or the model’s design cannot be changed.

In this work, we introduce Explanation of Pretrained Image Classification (EPIC), the first method that uses prototype-based reasoning without needing to retrain or change the original model’s design. Our approach maintains the model’s original accuracy while providing more precise and detailed explanations than typical saliency methods. We

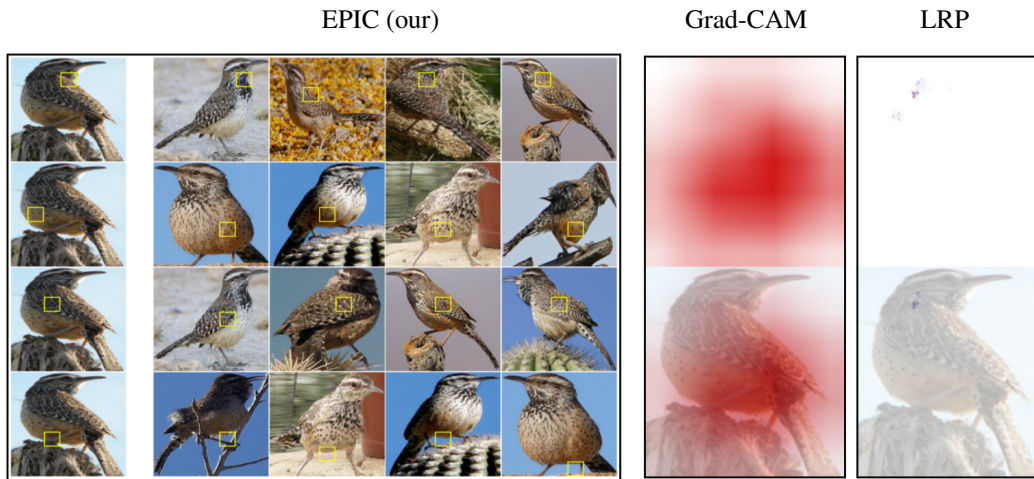


Figure 1: Comparison of explanations constructed by EPIC, and classical post-hoc models: Grad-CAM and LRP. The experiment is presented in the ResNet50 feature space on the Cactus Wren image from the CUB200-2011 dataset. Each row of EPIC (our) represents the prototypical part. The yellow boxes in each row show the activation of a given prototypical part, while in the first column, we show the activation of corresponding prototypical parts in the original image. Observe that contrary to the classical XAI post-hoc approaches (Grad-CAM and LRP), EPIC provides an explanation behind the decision of the model.

add a plug-in module to the model’s last layer that separates feature channels, as shown in Fig. 2. EPIC is the first model that uses prototypes in post-hoc XAI models, see Fig. 1. Therefore, the EPIC approach can be seamlessly applied to widely used datasets in prototype learning, such as CUB-200-2011 (Wah et al. 2011) and Stanford Cars (Krause et al. 2013), as well as general benchmarks like ImageNet (Deng et al. 2009), demonstrating broad applicability across tasks.

The core idea behind EPIC centers on defining a prototype purity measure, quantifying the degree of disentanglement of feature channels in the final layer. Naively extracting prototypes from a standard trained model typically results in low-quality explanations, as the learned channels are not aligned with coherent, interpretable concepts, see Fig. 3. To address this, EPIC introduces a lightweight sub-module attached to the final layer, which selectively reshapes the channel representations based on purity criteria. Crucially, this enhancement operates without altering the model’s predictions, focusing solely on producing disentangled, meaningful prototype channels. Our key contributions are summarized as follows:

- We propose EPIC, a principled post-hoc explanation framework that integrates prototype-based reasoning into existing deep models without retraining.
- We demonstrate that EPIC offers superior interpretability over saliency-map-based approaches by explicitly targeting prototype purity.
- We validate the versatility and generality of EPIC on both specialized fine-grained datasets (CUB-200-2011, Stanford Cars) and large-scale classification tasks (ImageNet).

2 Related Works

With the dynamic development and increasingly widespread deployment of deep learning models in key areas such as healthcare, finance, and autonomous systems, the issue of explainability has acquired the status of a fundamental research challenge. In the scholarly literature on explainable artificial intelligence (XAI), two principal paradigms can be distinguished: post-hoc explanation methods and inherently interpretable (ante-hoc) models.

Post-hoc methods focus on analyzing already trained models, providing explanations without interfering with their architecture. One example of such a method is SHAP (SHapley Additive exPlanations), which employs Shapley values to assign importance to individual features in a model’s prediction (Lundberg and Lee 2017). Similarly, the LIME (Local Interpretable Model-agnostic Explanations) method enables the creation of local linear models to interpret predictions (Ribeiro, Singh, and Guestrin 2016). Techniques such as Grad-CAM (Gradient-weighted Class Activation Mapping) generate attention maps that highlight input regions critical to the model’s decision-making process (Selvaraju et al. 2017). However, despite their popularity, these methods are often criticized for the instability and inconsistency of the explanations they generate, as well as for their limited ability to capture causal relationships (Adebayo et al. 2018).

By contrast, ante-hoc models integrate interpretability mechanisms directly into the architecture of the model itself. One such development is the ProtoPNet (Prototypical Part Network) algorithm, which introduces the concept of class prototypes, allowing the interpretation of model decisions by comparing image segments to representative prototypes (Chen et al. 2019). Extensions of this approach, such as PIP-Net (Prototype Interpretable Part Network), introduce mech-

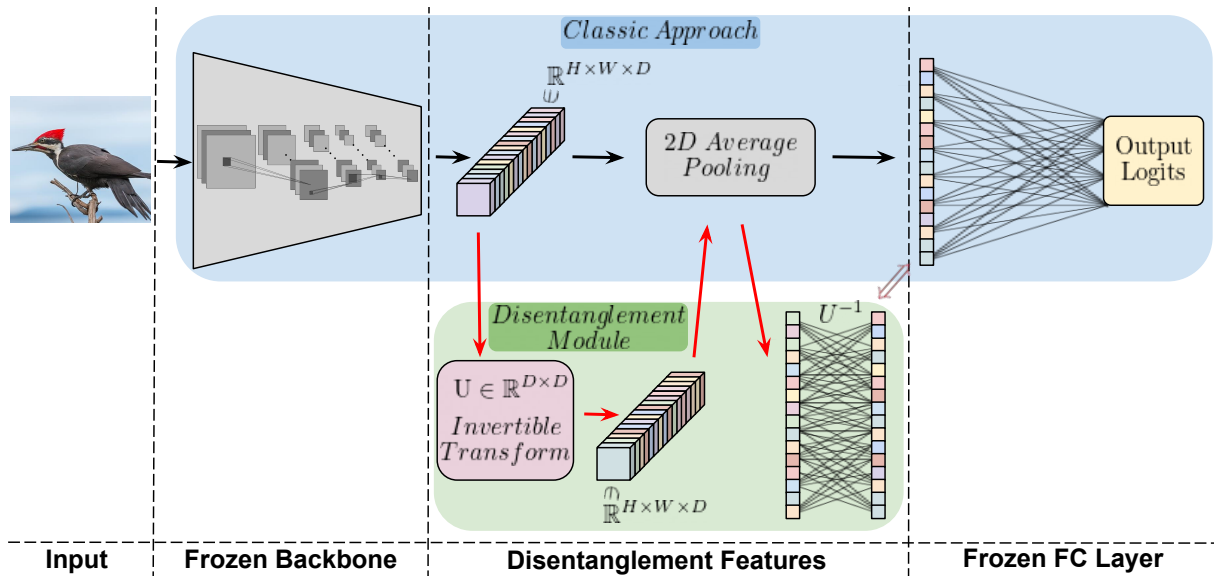


Figure 2: Our image classification interpretation model, EPIC, features three main components: a pre-trained backbone, a disentanglement layer for key features, and a fully connected layer. In contrast to the classical model, we introduce a square matrix of size equal to the number of channels, which enables the disentanglement of key features. To ensure the logits remain comparable to those of the classical model, we modify the weights in the fully connected layer by multiplying them with the inverse transformation used in the feature disentanglement step.

anisms for prototype selection and channel decomposition, thereby improving the quality of interpretations achieved (Nauta, Sieb, and van Gemert 2023). Nevertheless, ante-hoc models often require specialized architectures and retraining, which limits their applicability in existing, complex systems.

In response to the limitations of both approaches mentioned above, hybrid methods have been proposed. These combine the advantages of post-hoc and ante-hoc techniques. In this area, recent years have seen the development of solutions such as ACE (Automated Concept-based Explanations) and Concept Whitening. The ACE algorithm automatically identifies semantically coherent concepts within network layers, providing human-understandable interpretations (Ghorbani et al. 2019). Meanwhile, Concept Whitening introduces a mechanism for orthogonalizing the latent space, enabling a better understanding of the model’s internal representations (Chen et al. 2020). Although these methods offer new interpretability opportunities, they often do not provide prototype-based explanations characteristic of ante-hoc approaches and acceptable as fully correct.

Thus, there exists a clear gap between the flexibility of post-hoc methods and the deep interpretability of ante-hoc models. Our proposed method addresses this gap by enabling prototype-based explanations on top of already trained models. It combines the scalability offered by post-hoc techniques with the interpretability characteristic of ante-hoc approaches. Importantly, it achieves this without requiring any architectural modifications or retraining.

3 EPIC: Explanation of Pretrained models

In this section, we present the EPIC model, designed specifically to provide explanations for deep neural networks. Our approach involves integration of a plug-in Disentanglement Module into the network’s final layer, the classification head. EPIC disentangles the feature channels in this last layer based on a purity measure. As a post-hoc method, our model is applied to explain neural networks that have already been trained.

Our paper considers the classification networks used in PIPNet (Nauta, Sieb, and van Gemert 2023) and InfoDisent (Struski, Rymarczyk, and Tabor 2024). In the case of a classification task with N classes, we assume that we have a backbone Φ_{Θ} that transforms the input image I into the feature space $\Phi_{\Theta}(I) \in \mathbb{R}^{H \times W \times D}$ where H, W denote height and width of the map, and D denotes the number of channels (depth). Such a feature map then undergoes the pooling operation

$$v_I = \text{avg_pool}(\Phi_{\Theta}(I)) \in \mathbb{R}^D.$$

At the end of such operations, we have a linear classification layer $w_I = Av_I$, where A is a matrix of dimensions $N \times D$, where N is the number of classes. Finally, Softmax is applied to obtain the final probabilities for each class.

In this type of architecture, each channel of the final feature space in which the $\Phi_{\Theta}(I)$ resides can be interpreted as an individual prototype (Nauta, Sieb, and van Gemert 2023; Struski, Rymarczyk, and Tabor 2024). Before explaining how to ensure these channels provide coherent explanations, we first demonstrate the process of finding prototypes of a fixed channel for a traditionally trained model. Subsequently, we introduce a measure for the distribution of the

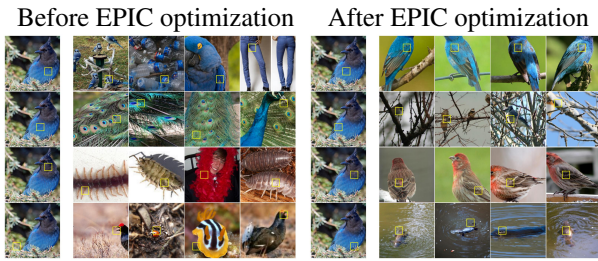


Figure 3: Explanations for a blue jay bird, before (left) and after (right) EPIC training on Resnet18. As we can see, prototypes without additional tuning correspond to random images and have limited explanatory properties. After EPIC tuning, such prototypes are consistent and correspond with input image features.

channels in a prototype, referred to as the purity measure. We then describe the approach to maximize the purity using Disentanglement Module. Finally, we outline the construction of the explanations for an input image.

Prototypes of a Feature Map Channel The main component of our approach is finding a set of images connected to each feature map channel, which will represent the information propagated by a specific channel. Consequently, we are looking for m (usually $m = 5$) prototype images from the training set for a fixed channel k . All that remains is to specify how the prototypes are selected. Provided an image I we calculate its representation in the feature space $Z_I = \Phi_{\Theta}(I) \in \mathbb{R}^{H \times W \times D}$. This can be viewed as a representation on which the model’s classification head works.

We are looking for m images that activate mainly on the k -th channel. More specifically, we define the activation of a channel $k \in \{1, 2, \dots, D\}$:

$$\text{activ}(Z; k) = \sum_{h=1}^H \sum_{w=1}^W Z[h, w, k] \text{ for } Z \in \mathbb{R}^{H \times W \times D}.$$

Activation of the channel k at height h and width w in the feature space is denoted by $Z[h, w, k]$. Additionally, let us note that we will later refer to the vector $Z[h, w] \in \mathbb{R}^D$ as a pixel in feature space interpreted as an image with D channels. This vector will later be crucial to understanding the prototype’s quality.

By using channel activation, we can select m prototype images for the k -th channel:

$$\text{Prot}_{\text{pos}}^{(k)} = \arg \text{top-}m_{I \in \text{TrainSet}} \text{activ}(Z_I; k).$$

This process can be summarized as the application of the channel activation function to all images in the training set, and finding the images for which the m largest values is obtained. The chosen images will be called positive prototypes of channel k . Similarly we can define negative prototypes as

$$\text{Prot}_{\text{neg}}^{(k)} = \arg \text{top-}m_{I \in \text{TrainSet}} - \text{activ}(Z_I; k).$$

This process can be repeated for all channels to obtain their prototypes. The results for the classically trained neural network without any modifications and the results of EPIC

are presented in Fig. 3. As we can see, without additional tuning, such prototypes are less clear than the ones obtained after the training of EPIC. To measure the quality of the prototype image we use a measure called purity introduced in the following section. In our model, we use Disentanglement Module to make the prototypes more coherent. However, we still have to find a method to evaluate the quality of a prototype.

Purity of Prototype In this paragraph, we define the purity measure employed by EPIC to disentangle channels in the feature space. Classical optimization concentrates on the prediction task and produces a mixed representation. As a result, concepts related to the model prediction are entangled between different channels. Representation is fully disentangled if only one channel is active for a given image. EPIC uses purity measure to assess the disentanglement of the feature space, see Fig. 4. In our paper, we focus on the positive prototypes. However, the process is analogous for negative prototypes. Below, we present a detailed formulation of the purity of the prototype.

For a given backbone Φ_{Θ} , input image I , and selected prototypical channel k , we define a prototypical pixel, the coordinates of it are defined as

$$\mathbb{N}^2 \ni (h, w) = \arg \max_{x, y} Z_I[x, y, k].$$

That is the coordinates of the largest activation in the k -th channel. The prototypical pixel is then given by a vector $p = Z_I[h, w] \in \mathbb{R}^D$. It spans the channels across the spatial location in which the largest activation of k -th channel is achieved. By using this vector we can define the purity by:

$$\text{purity}(I, k) = \frac{p_k}{\|p\|} \in [0, 1].$$

If the value of purity (I, k) is equal to one, we call the prototype pure. This situation occurs when all but the k -th channel activations are zeroes, which is consistent with the motivation behind this measure. In Fig. 4, we visualize such a situation. Before purity optimization, our prototype pixels were not pure since the histogram of activation was uniformly distributed. After optimization, the neural network primarily activates along a single coordinate. During optimization of Disentanglement Module the feature space is disentangled by forcing the prototypes to be pure.

Disentanglement Module The prototypes can be used to explain a neural network’s prediction, as noted the larger the purity the better the explanation. Our goal is to disentangle channels in the feature space of a pretrained model, while simultaneously preserving the original models prediction. Consequently, we propose to use a Disentanglement Module, which uses a learnable invertible matrix $U \in \mathbb{R}^{D \times D}$ to separate the channels inside the feature space. Thus, EPIC is injected into the model just before the Pooling Layer, and the final linear layer weight is multiplied by U^{-1} to preserve the original output. More precisely, for an input image I , we first transform the original image into feature space $Z = \Phi_{\Theta}(I) \in \mathbb{R}^{H \times W \times D}$. Next, we apply the matrix $U \in \mathbb{R}^{D \times D}$ to each spatial location of $Z \in \mathbb{R}^{H \times W \times D}$,

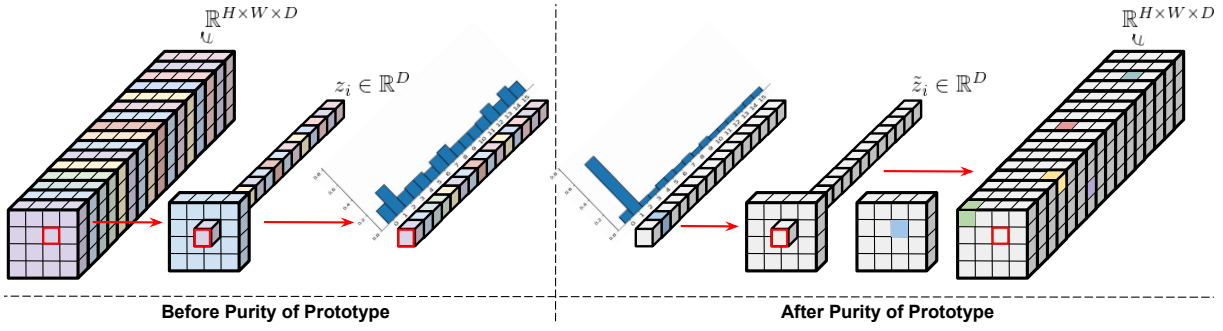


Figure 4: The illustration demonstrates the concept of the *Purity of Prototype* mechanism. For a selected channel, the vector \mathbf{z} (shown on the left) is defined by the maximum pixel value in that channel, making its values *comparable* (histogram of activation is flat). After optimizing the purity of the given prototype, only one dominant value remains in the refined vector $\tilde{\mathbf{z}}$, as seen on the right. Repeating this process for each channel results in a disentangled representation, where each channel contains only one dominant value associated with its prototype.

transforming feature space in which the channels are disentangled. More precisely, for each pixel coordinates (x, y) the feature vector $Z[x, y] \in \mathbb{R}^D$ is projected to a new space by $\mathbb{R}^D \ni \hat{Z}[x, y] = UZ[x, y]$. This operation can be summarized as the application of a linear operator U to each pixel. We will later denote this operation by $U \circledast Z$.

To preserve the original activations, we have to reverse this operation in the classification head of the model. This can be achieved by substituting the weight A of the linear classification layer, by $A' = AU^{-1}$. The final model can be summarized as

$$Z = \Phi_{\Theta}(I) \in \mathbb{R}^{H \times W \times D}, \quad (1)$$

$$\hat{Z} = U \circledast Z \in \mathbb{R}^{H \times W \times D}, \quad U \in \mathbb{R}^{D \times D}, \quad (2)$$

$$v = \text{avg_pool}(\hat{Z}) \in \mathbb{R}^D, \quad (3)$$

$$w = A'v = (AU^{-1})v, \quad (4)$$

$$\text{pred} = \text{softmax}(w). \quad (5)$$

The proposed neural network modification does not alter the final prediction of the network, which is a consequence of the simple Remark 3.1.

Remark 3.1. Let $Z \in \mathbb{R}^{H \times W \times D}$ be an image representation in feature space and $U \in \mathbb{R}^{D \times D}$ an invertible matrix, then:

$$U^{-1} \text{avg_pool}(U \circledast Z) = \text{avg_pool}(Z).$$

Proof. This follows from a distributive property of matrices. At each spatial location (x, y) , we have:

$$\begin{aligned} U^{-1} \text{avg_pool}(U \circledast Z) &= U^{-1} \left(\frac{1}{HW} \sum_{x,y} UZ(x, y) \right) \\ &= U^{-1}U \left(\frac{1}{HW} \sum_{x,y} Z(x, y) \right) = \text{avg_pool}(Z) \quad \blacksquare \end{aligned}$$

Such a simple modification allows us to disentangle channels. We train the matrix U with a restriction to either the class of invertible or orthogonal matrices. It is worth noting that if we set the matrix U to identity matrix, we get exactly the original pretrained model.

Training As mentioned in the previous section the quality of a prototype is tied to the value of purity. Consequently, the training stage focuses on the maximization of prototypes purity. But since, we want to preserve the original model output, all its weights are frozen and only the elements of matrix U in the Disentanglement Module are updated. Additionally, the optimization process is done solely on the set of prototypes. However, since each update to matrix U causes a change in the activations of channels, the new set of prototypes has to be recalculated every few epochs throughout the training. This provides the compromise between the speed, and dynamic updates to prototypes. In our experiments, the Disentanglement Module was trained for 20 epochs, with prototypes being recalculated every 2 epochs. In addition to the update of prototypes, the number of prototypes for each channel is decreased at the same time. We start with 100 images for each prototypical channel, and linearly decrease this value to 5 at the end of the training stage. We trained the Disentanglement Module using Adam optimizer with $\beta_1 = 0.9$, $\beta_2 = 0.999$, a batch size of 512 and applied a weight decay of $1e-5$, and learning rate equal to 0.001.

Explaining Model Prediction After completing the training of the Disentanglement Module and selecting the channel prototypes, the next step is to explain the model's predic-

Algorithm 1: Top- k Contributing Channels

```

1: procedure TOPKCHANNELS( $\Phi_{\Theta}, A, I, k, U$ )
2:    $Z \leftarrow \Phi_{\Theta}(I) \in \mathbb{R}^{H \times W \times D}$   $\triangleright$  Feature map
3:    $\hat{Z} \leftarrow U \circledast Z \in \mathbb{R}^{H \times W \times D}$   $\triangleright$  Disentanglement
4:    $A' \leftarrow AU^{-1}$ 
5:    $v \leftarrow \text{avg\_pool}(\hat{Z}) \in \mathbb{R}^D$   $\triangleright$  Global average pooling
6:    $w \leftarrow A'v \in \mathbb{R}^C$   $\triangleright$  Logits
7:    $\text{pred} \leftarrow \arg \max(w)$   $\triangleright$  Predicted class
8:    $w_{\text{pred}} \leftarrow A'[\text{pred}]$   $\triangleright$  Weights for predicted class
9:    $\text{scores} \leftarrow w_{\text{pred}} \odot \text{ReLU}(v)$   $\triangleright$  Hadamard product
10:   $\text{channels} \leftarrow \text{TopK}(\text{scores}, k)$ 
11:  return channels
12: end procedure

```

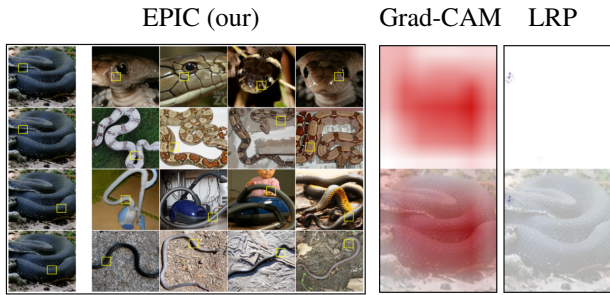


Figure 5: Explanations for the Hognose Snake from ImageNet constructed by EPIC (our), Grad-CAM and LRP. EPIC effectively capture crucial concepts, such as shapes, colors, textures, and distinctive features like the snake’s eye area. In contrast, Grad-CAM and LRP produce only saliency maps, offering less interpretability regarding specific visual attributes and concepts.

tion for a given input image. This is achieved by selecting k channels with the highest contribution to the predicted class. This can be done by examining the terms contributing to the model output in the final classification layer. More precisely, for an input image I and the model prediction of the input belonging to class y (for more details, see the algorithm in the Algorithm 1. Since we are only interested in the positive prototypes, we apply ReLU before examining the terms contributing to the sum. Example explanation is shown in Fig. 5.

4 Experiments and Results

In the experimental section, we evaluate our model across several scenarios. First, we provide a qualitative comparison, showcasing example predictions and comparing our results against post-hoc methods such as Grad-CAM, LRP, as well as the prototype-based model InfoDisent. Then, we present that our model is only a plugin to the model, and we do not change the network’s prediction. Next, we show a multidimensional analysis of the FunnyBirds datasets. Finally, we present the results of user studies.

Explanation of Model Decision This section outlines the experimental results of EPIC explanations and its comparison to other XAI methods, including both post-hoc and ante-hoc approaches. Fig. 5 illustrates the interpretability improvements of EPIC over classical post-hoc methods, Grad-CAM and LRP, on the input images from CUB200-2011 and Stanford Dogs (Khosla et al. 2011) datasets. Each row in the EPIC visualization represents the prototypical part (the corresponding channel). The yellow boxes in each row show the activation of a given prototypical part, while in the second column, we show the activation of corresponding prototypical parts in the original image. While EPIC demonstrates clear part-level interpretability, Grad-CAM and LRP produce more diffused heatmaps that highlight general areas of importance but lack the fine-grained interpretability. EPIC not only highlights critical regions, but also provides semantically rich prototypes that represent these crucial vi-

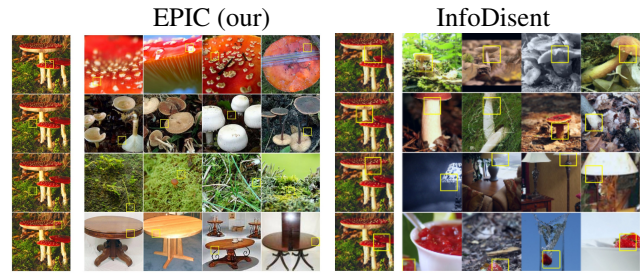


Figure 6: Comparison of explanations between EPIC (our) and prototype-based model InfoDisent. InfoDisent operates on top of the pre-trained backbone by finetuning the classification head. EPIC does not change original architecture and predictions. The comparison is conducted on a representation learned on top of pretrained ResNet50.

sual features.

Fig. 6 presents a comparison of explanations generated by EPIC and the prototype-based model InfoDisent. While InfoDisent operates on a pretrained backbone and can produce predictions on the ImageNet dataset, EPIC constructs prototypes that are more closely aligned with the input images.

Classification Performance As a consequence of 3.1 EPIC preserves the predictive ability of the pretrained model. However, since we apply additional operations, numerical errors might arise. To show that this situation does not occur, we present the numerical accuracy on ImageNet in Tab. 1.

Multi-dimensional Analysis To assess our methodology, in the last experiment, the FunnyBirds (Hesse, Schaub-Meyer, and Roth 2023) dataset was used. Semantically relevant image modifications, like deleting individual object pieces, are supported by the FunnyBirds dataset as well as by our innovative automatic evaluation algorithms. Thus, XAI methods and model architectures were developed to provide a more comprehensive evaluation of explanations on the part level. Like humans observing an image, they concentrate on distinct elements instead of individual pixels. EPIC is compared with multiple methods on classical convolutional network (Resnet50) for which it ranks among the best Fig. 7.

Model	ACC	Model	ACC
ResNet-34	73.3%	ConvNeXt-L	84.4%
EPIC	73.3%	EPIC	84.4%
InfoDisent	64.1%	InfoDisent	82.8%
ResNet-50	80.8%	Swin-S	83.7%
EPIC	80.8%	EPIC	83.7%
InfoDisent	67.8%	InfoDisent	81.4%
DenseNet-121	74.4%		
EPIC	74.4%		
InfoDisent	66.6%		

Table 1: Classification accuracy (ACC) on ImageNet dataset by competing approaches using different backbones.

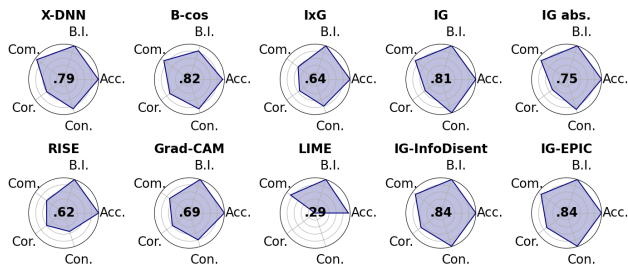


Figure 7: FunnyBirds evaluation results for various XAI methods: Input×Gradient (IxG) (2017), (absolute) Integrated Gradients (IG (abs.)) (2017), Grad-CAM (2017), RISE (2018), LIME (2016), *X*-DNN (2021), B-cos network (2022) and InfoDisent (2024). Resnet50 are used to evaluate model-agnostic techniques. The center score, which represents the mean of the completeness (Com.), correctness (Cor.), and contrastivity (Con.) dimensions, is calculated by averaging the results throughout the whole test set. Furthermore, background independence (B.I.) and accuracy (Acc.) are reported. Our approach (last from the left) equals the best result for Resnet50.

User Study Results We conducted two user studies, each involving 60 participants per dataset. Both studies utilized two datasets: CUB-200-2011 and ImageNet. During the studies, each participant answered 20 questions, with images randomly drawn from the testing datasets for each question.

The first user study aimed to evaluate user overconfidence when assessing model predictions. Participants were shown an image along with the model’s explanation and were asked to choose one of four response about the model’s prediction. Answers included information whether the model’s output was correct or incorrect along with associated confidence level—categorized as fairly confident or somewhat confident. Results from this study are reported in Tab. 2. The table reports key metrics on users’ performance including true correct accuracy (user agreement when the model was right), true incorrect accuracy (user disagreement when the model was wrong), standard deviation and p-values assessing statistical significance compared to random guessing. The findings from this study reveal that participants exposed by explanations by EPIC exhibited general statistically significant confidence in the model’s correct predictions across ImageNet and CUB200-2011 datasets. However, users encounter challenges in accurately identifying incorrect predictions made by the model based on these explanations, a pattern consistent with previous findings from other XAI techniques.

The objective of the second user study was to evaluate how effectively participants could distinguish between prototypical parts. During the study, participants were presented with an image classified by the model, along with two explanations representing the top two most activated classes. Their task was to identify which class the model had ultimately selected, using only the information provided in the explanations. The results, shown in Tab. 3, indicate that participants achieve statistically significantly higher accu-

Method	Prediction	ImageNet	CUB-200-2011
EPIC	Correct	0.637±0.480	0.611±0.487
	Incorrect	0.447±0.497	0.294±0.456
InfoDisent	Correct	0.602±0.090	0.807±0.133
	Incorrect	0.553±0.099	0.427±0.117
ProtoPNet	Correct	NA	0.732±0.249
	Incorrect	NA	0.464±0.359
GradCAM	Correct	0.708±0.266	0.724±0.215
	Incorrect	0.448±0.316	0.328±0.243

Table 2: The table reports metrics on the users’ performance in the first user study, including accuracy and standard deviation. Statistically significant values are highlighted in bold.

Method	Dataset	User Acc.	p-value
EPIC	ImageNet	0.568±0.495	$8 \cdot 10^{-4}$
	CUB	0.55±0.497	$9 \cdot 10^{-3}$
InfoDisent	ImageNet	0.593±0.149	$8 \cdot 10^{-6}$
	CUB	0.647±0.131	10^{-14}
ProtoPNet	CUB	0.515±0.052	0.288
ProtoConcepts	CUB	0.621±0.054	$3 \cdot 10^{-5}$
PIPNet	CUB	0.600±0.181	0.002
LucidPPN	CUB	0.679±0.169	$2 \cdot 10^{-6}$

Table 3: The table reports accuracy, standard deviation and p-values for users’ performance in second user study. The p-value column indicates the p-value of a test against random.

racy in identifying the correct class for both the ImageNet and CUB200-2011 datasets compared to random guessing. This demonstrates that EPIC enhances user understanding of model predictions beyond mere chance levels.

5 Conclusions

In this work, we introduced EPIC, a novel framework that unifies the strengths of post-hoc and prototype-based explanation methods for image classification. Unlike traditional prototype models that require specialized architectures and training procedures, EPIC operates directly on pretrained networks without altering their structure or predictions. At the same time, it retains the intuitive, human-interpretable explanations offered by prototype-based approaches. Our experiments across benchmark and large-scale datasets demonstrate that EPIC provides high-quality, interpretable insights into model behavior while maintaining the flexibility and applicability of post-hoc methods. EPIC is a step toward making AI systems more transparent and easier to understand without sacrificing flexibility.

Limitations EPIC can be used only for architectures with a classification head consisting of a pooling layer on top of the backbone and a single-layer classification head.

Acknowledgments

The work of P. Borycki, and P. Spurek was supported by the project *Effective Rendering of 3D Objects Using Gaussian Splatting in an Augmented Reality Environment* (FENG.02.02-IP.05-0114/23), carried out under the First Team programme of the Foundation for Polish Science and co-financed by the European Union through the European Funds for Smart Economy 2021–2027 (FENG). This research was partially funded by the National Science Centre, Poland, grants no. 2023/49/B/ST6/01137 (work of Jacek Tabor and Łukasz Struski). Some experiments were performed on servers purchased with funds from the flagship project entitled “Artificial Intelligence Computing Center Core Facility” from the DigiWorld Priority Research Area within the Excellence Initiative – Research University program at Jagiellonian University in Kraków. Magdalena Trędownicz and Arkadiusz Lewicki would also like to give their acknowledgments to the Prometheus MedTech.AI for financial support of the ongoing research work reported in this paper.

References

- Adebayo, J.; Gilmer, J.; Muelly, M.; Goodfellow, I.; Hardt, M.; and Kim, B. 2018. Sanity checks for saliency maps. *Advances in Neural Information Processing Systems*, 31.
- Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS One*, 10(7): e0130140.
- Böhle, M.; Fritz, M.; and Schiele, B. 2022. B-cos networks: Alignment is all we need for interpretability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 10329–10338.
- Chen, C.; Barnett, A.; Su, J.; Rudin, C.; and Venkatasubramanian, S. 2020. Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, 2: 772–782.
- Chen, C.; Li, O.; Tao, D.; Barnett, A.; Rudin, C.; and Su, J. K. 2019. This looks like that: Deep learning for interpretable image recognition. *Advances in Neural Information Processing Systems*, 32.
- Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 248–255.
- Ghorbani, A.; Wexler, J.; Zou, J. Y.; and Kim, B. 2019. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, volume 32.
- He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.
- Hesse, R.; Schaub-Meyer, S.; and Roth, S. 2021. Fast axiomatic attribution for neural networks. *Advances in Neural Information Processing Systems*, 34: 19513–19524.
- Hesse, R.; Schaub-Meyer, S.; and Roth, S. 2023. Funny-Birds: A Synthetic Vision Dataset for a Part-Based Analysis of Explainable AI Methods. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV), Paris, France, October 2-6, 2023*, 3981–3991. IEEE.
- Khosla, A.; Jayadevaprakash, N.; Yao, B.; and Fei-Fei, L. 2011. Novel Dataset for Fine-Grained Image Categorization. In *First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition*. Colorado Springs, CO.
- Krause, J.; Stark, M.; Deng, J.; and Fei-Fei, L. 2013. 3D Object Representations for Fine-Grained Categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, 554–561.
- Lipton, Z. C. 2018. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3): 31–57.
- Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30.
- Nauta, L.; Sieb, M. H.; and van Gemert, J. C. 2023. PIP-Net: Prototypical part network for interpretable fine-grained recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Petsiuk, V.; Das, A.; and Saenko, K. 2018. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*.
- Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144.
- Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2017. Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 618–626.
- Selvaraju, R. R.; Cogswell, M.; Das, A.; Vedantam, R.; Parikh, D.; and Batra, D. 2020. Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128: 336–359.
- Shrikumar, A.; Greenside, P.; Shcherbina, A.; and Kundaje, A. 2017. Not Just a Black Box: Learning Important Features Through Propagating Activation Differences. *arXiv:1605.01713*.
- Struski, Ł.; Rymarczyk, D.; and Tabor, J. 2024. InfoDisent: Explainability of Image Classification Models by Information Disentanglement. *arXiv preprint arXiv:2409.10329*.
- Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328. PMLR.
- Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. Caltech-UCSD Birds 200. Technical Report CNS-TR-2011-001, California Institute of Technology.
- Xu, F.; Uszkoreit, H.; Du, Y.; Fan, W.; Zhao, D.; and Zhu, J. 2019. Explainable AI: A brief survey on history, research areas, approaches and challenges. In *Natural Language Processing and Chinese Computing*, 563–574. Springer.