Explaining Representation by Mutual Information

Lifeng Gu [1]

Abstract

As interpretability gains attention in machine learning, there is a growing need for reliable models that fully explain representation content. We propose a mutual information (MI)-based method that decomposes neural network representations into three exhaustive components: total mutual information, decision-related information, and redundant information. This theoretically complete framework captures the entire input-representation relationship, surpassing partial explanations like those from Grad-CAM. Using two lightweight modules integrated into architectures such as CNNs and Transformers,we estimate these components and demonstrate their interpretive power through visualizations on ResNet and prototype network applied to image classification and few-shot learning tasks. Our approach is distinguished by three key features:

1. Rooted in mutual information theory, it delivers a thorough and theoretically grounded interpretation, surpassing the scope of existing interpretability methods.
2. Unlike conventional methods that focus on explaining decisions, our approach centers on interpreting representations.
3. It seamlessly integrates into pre-existing network architectures, requiring only fine-tuning of the inserted modules.

1. Introduction

Representation learning has progressed rapidly (Chen et al., 2020; Grill et al., 2020), yet the essence of an effective representation remains unclear. Deep neural networks generate opaque representations, masking the connection between inputs and outputs. Methods like Grad-CAM (Selvaraju et al., 2017) identify decision-relevant features but fail to capture the complete information encoded by local inputs. We propose a mutual information (MI)-based framework that decomposes the information between local inputs and representations into three exhaustive components: total mutual information $I(Z, X_i)$, decision-related information $I(Z, X_i')$, and redundant information $R(Z, X_i) = I(Z, X_i) - I(Z, X_i')$. This approach fully accounts for $I(X_i, Z)$, providing a holistic view of representation content. Using two lightweight modules, we estimate these components and validate their effectiveness through visualizations on ResNet and a prototype network.

2. Related Works

2.1. Post-hoc Explanation Methods

Post-hoc explanation methods aim to interpret neural network behavior by analyzing global aspects of their predictions, typically falling into two broad categories: visualization techniques for convolutional networks and attribution-based approaches that link network outputs or decisions to input features. Visualization methods emphasize intuitive depictions of network activity, whereas attribution-based techniques align more closely with our objective of providing precise, actionable insights into model behavior. In this section, we focus on attribution-based methods. Gradient-based approaches, such as Gradient Maps (Baehrens et al., 2010) and Saliency Maps (Simonyan et al., 2013), compute the gradient of the output with respect to input features to identify regions driving network decisions. While straightforward and computationally efficient, their gradients are often unstable and sensitive to noise, reducing their trustworthiness. To overcome these weaknesses, Integrated Gradients (Sundararajan et al., 2017) averages gradients across multiple inputs, improving stability and mitigating noise. Other techniques, such as Layer-wise Relevance Propagation (LRP) (Bach et al., 2015a), Deep Taylor Decomposition (DTD) (Montavon et al., 2017), and Grad-CAM (Selvaraju et al., 2017), propagate relevance scores backward through intermediate layers to assign importance to features, frequently relying on gradient-derived weights. A hybrid method, Guided Grad-CAM, integrates Guided Backpropagation with Grad-CAM to enhance stability and deliver more reliable explanations. Despite these improvements, post-hoc methods often fall short of revealing the underlying structure of learned representations—a limitation our approach addresses with a more comprehensive

Affiliations [1]Tju
Correspondence Lifeng Gu - gulifeng666@163.com

and theoretically grounded framework.

2.2. Disentangled Representation Learning

Disentangled representation learning focuses on interpreting neural networks by decomposing representations into individual components, assuming that data variations arise from independent generative factors—such as image orientation, lighting conditions, or object-specific attributes (e.g., hair length in portraits). The guiding principle is that a disentangled representation isolates these factors, such that modifying one component does not impact others. This characteristic is believed to facilitate downstream tasks; for example, in gender classification, a model could rely exclusively on the "gender" component of the representation.

A leading approach in this domain is the Variational Autoencoder (VAE), which enforces constraints on the posterior approximation $z \sim q(z|x)$. The standard VAE objective balances reconstruction fidelity with regularization:

$$L_{VAE} = \mathbb{E}_{q(z|x)}[\log p(x|z)] - D_{KL}(q(z|x)\|p(z)),$$

where $p(z)$ is typically a standard Gaussian prior. To enhance disentanglement, Higgins et al. (Higgins et al., 2016) introduced β-VAE, which increases the weight of the KL divergence term:

$$L_{\beta\text{-}VAE} = \mathbb{E}_{q(z|x)}[\log p(x|z)] - \beta D_{KL}(q(z|x)\|p(z)),$$

where $\beta > 1$ encourages $q(z|x)$ to match a prior with independent dimensions. Extending this, Kim and Mnih (Kim & Mnih, 2018) proposed FactorVAE, incorporating a total correlation (TC) penalty to further minimize dependencies among latent variables:

$$L_{\text{FactorVAE}} = L_{VAE} + \lambda TC(q(z)),$$

where $TC(q(z))$ measures mutual dependencies and requires density ratio estimation for computation. While these methods excel in controlled settings, their reliance on the assumption of factor independence often proves restrictive in complex, real-world applications. In contrast, our approach harnesses mutual information to offer a more flexible and robust framework for interpreting representations, bypassing the constraints of strict independence assumptions.

3. Method

3.1. Mutual Information Estimation

Neural network representations encode information extracted from input data. To investigate this encoded information, we apply mutual information theory to quantify the mutual information between the representation $f(X)$ and the input X, denoted $I(X, f(X))$. This metric reflects how much input information is preserved in the representation. However, $I(X, f(X))$ alone offers limited insight into the specific details captured. To address this, we compute the mutual information between local input components X_i (e.g., pixels in an image or words in a sentence) and $f(X)$, pinpointing the exact input elements encoded in the representation. This is formally defined as:

$$\begin{aligned} I(X_i, f(X)) &= \mathbb{E}_{X_i}\left[KL(p(f(X) \mid X_i) \parallel p(f(X)))\right] \\ &= \mathbb{E}_{X_i}\left[KL(p(X_i \mid f(X)) \parallel p(X_i))\right], \end{aligned} \quad (1)$$

where KL denotes the Kullback-Leibler divergence. Direct computation of the marginal distributions $p(X_i)$ and $p(f(X))$, or the conditional distributions $p(X_i \mid f(X))$ and $p(f(X) \mid X_i)$ in Equation (1), is computationally infeasible due to their complexity. To tackle this, we adopt the InfoNCE method (Tschannen et al., 2019) to estimate $I(X_i, f(X))$. Letting $Z = f(X)$ represent the representation, the mutual information satisfies:

$$\begin{aligned} I(X_i, f(X)) &= I(f(X), X_i) = I(Z, X_i) \\ &\geq \mathbb{E}_{x_i \sim p(X_i), z \sim p(Z)}\left[\log \frac{e^{f(z,x_i)}}{\frac{1}{NK}V}\right], \\ V &= \sum_{n=1}^{N}\sum_{k=1}^{K} e^{f(z^{(n)}, x_k^{(n)})}, \end{aligned} \quad (2)$$

where $z^{(n)}$ is the representation of the n-th sample, and $x_k^{(n)}$ is its k-th local component. Here, N denotes the batch size, and K is the number of local components per sample (e.g., pixels or words). We optimize this lower bound by maximizing it with respect to the parameters θ of the Infomax estimator module, yielding:

$$I(Z, X_i) = \max_{\theta} \mathbb{E}_{x_i \sim p(X_i), z \sim p(Z)}\left[\log \frac{e^{f(z,x_i)}}{\frac{1}{NK}V}\right]. \quad (3)$$

The scoring function $f(z, x_i)$ is computed by the Infomax estimator module, typically a single- or two-layer multilayer perceptron (MLP), assessing similarity between the representation and local inputs. This is illustrated in Figure 1. While alternatives exist (Hjelm et al., 2018; Belghazi et al., 2018), InfoNCE is chosen for its low variance and proven efficacy in representation learning.

3.2. Information Bottleneck

Our next goal is to extract decision-related information from the representationinformation vital for preserving

model decisions. We adopt the information bottleneck principle (Tishby & Zaslavsky, 2015), where Y denotes the label. This approach maximizes the mutual information between the representation Z and Y, while minimizing the mutual information between Z and the input X, formulated as:

$$\max I(Z,Y) \quad \text{subject to} \quad I(X,Z) \leq c \tag{4}$$

Optimizing Equation (4) directly is challenging due to its complexity. Inspired by masking mechanisms, we introduce a mask layer after the input X, transforming each local component as $x_i = x_i \cdot \lambda_i$, where $\lambda_i \in [0,1]$ controls the information flow for the i-th component. The mask is generated by a simple two-layer MLP or convolutional layer, which processes the input to produce λ_i, then multiplies it with X before forwarding it to subsequent layers, as shown in Figure 1. The optimization objective becomes:

$$\max_{\phi} \left[l(X) - \beta \sum_i \lambda_i \right] \tag{5}$$

Here, $l(X)$ is the original objective function designed to maximize $I(Z,Y)$, aligning the representation with the label. The term $\beta \sum_i \lambda_i$ minimizes $I(X,Z)$, with β as a hyperparameter balancing these goals. The parameters ϕ govern the mask module, approximating the information bottleneck by retaining decision-critical information while suppressing excess details.

3.3. Information Redundancy

Finally, we aim to isolate redundant information within the representationinformation irrelevant to decision-making and removable without affecting the models output. We first calculate $I(X,Z)$, the mutual information between the input X and the representation Z, and $I(X',Z)$, the decision-related information, where X' is the masked input retaining only critical components. The redundancy for a local component X_i is:

$$R(Z,X_i) = I(Z,X_i) - I(Z,X_i') \tag{6}$$

Both $I(Z,X_i)$ and $I(Z,X_i')$ are estimated using the InfoNCE method from Equation (2). Specifically, $I(Z,X_i)$ measures the total information encoded from X_i into Z, while $I(Z,X_i')$ captures the decision-relevant portion after masking. Redundancy is derived by their difference.

To analyze the representation comprehensively using these three information typesmutual information, decision-related information, and redundant informationwe integrate them into a unified objective function. Combining Equations (3) and (5), the total objective is:

$$\begin{aligned} L(X) = \max_{\theta,\phi} \Bigg[& l(X) + \alpha \mathbb{E}_{x_i \sim p(X_i), z \sim p(Z)} \left[\log \frac{e^{f(z,x_i)}}{\frac{1}{NK} V} \right] \\ & + \alpha \mathbb{E}_{x_i' \sim p(X_i'), z \sim p(Z)} \left[\log \frac{e^{f(z,x_i')}}{\frac{1}{NK} V'} \right] \\ & - \beta \sum_i \lambda_i \Bigg], \end{aligned} \tag{7}$$

$$\begin{aligned} V &= \sum_{n=1}^{N} \sum_{k=1}^{K} e^{f(z^{(n)}, x_k^{(n)})}, \\ V' &= \sum_{n=1}^{N} \sum_{k=1}^{K} e^{f(z^{(n)}, x_k'^{(n)})} \end{aligned} \tag{8}$$

In this expression, $l(X)$ is the original objective function aimed at maximizing $I(Z,Y)$, with θ as the parameters of the Infomax estimator module and ϕ as the parameters of the mask module. The first expectation term estimates $I(Z,X_i)$, reflecting the mutual information between Z and X_i. The second term, a novel addition, estimates $I(Z,X_i')$ using the masked input X_i'. Hyperparameters α and β balance mutual information estimation and information bottleneck regularization, respectively. The terms V and V' are normalization factors for the unmasked and masked inputs. By optimizing Equation (8) via the Infomax and mask modules, we effectively estimate these three information types $I(Z,X_i)$, $I(Z,X_i')$ and $R(Z,X_i)$,without altering the original networks parameters, requiring only fine-tuning of the two lightweight modules. The architecture is depicted in Figure 1.

4. Experiments

Unlike other interpretability methods (Baehrens et al., 2010; Bach et al., 2015b; Schulz et al., 2020), which are generally developed to explain model decisions, our approach centers on analyzing and interpreting model representations. In this section, we present experiments conducted on the ImageNet-Mini (a subset of 100 classes, 50,000 images) and CUB-200-2011 (200 bird species, 11,788 images). The Infomax estimator is a two-layer MLP with 256 hidden units, and the mask module is a two-layer CNN with 3×3 kernels. Hyperparameters are set as $\alpha = 1.0$, $\beta = 0.5$

We select the output of an early layer in the network as the local representation of input samples and the output of a later layer as the global representation. By computing the mutual information between these two layers' outputs, as well as the mutual information between the masked output of the former and the global representation, we derive the

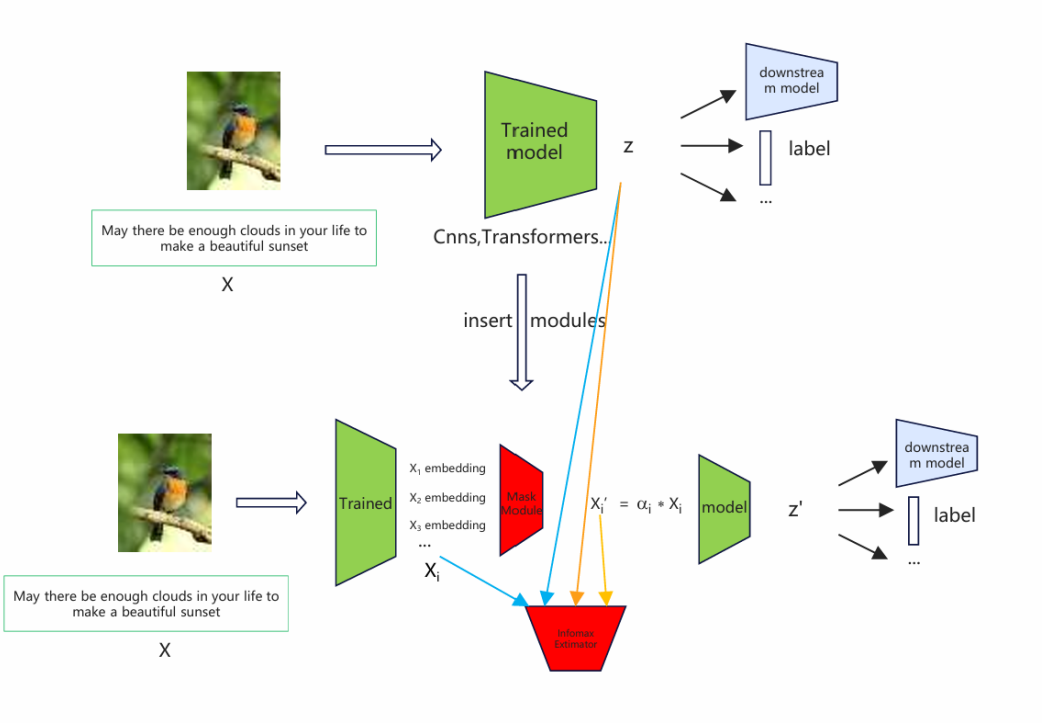


Figure 1. module architecture: the Infomax estimator processes local inputs X_i and representation Z to estimate $I(Z, X_i)$, while the mask module generates λ_i to filter X into X', enabling $I(Z, X_i')$ computation.

three types of information targeted in our study: total information, decision-related information, and redundant information.

4.1. Image Classification Visualization

Image classification visualization is a benchmark task in interpretability research. We visualize images and their corresponding information heatmaps from the ImageNet-Mini dataset and compare our results with those of Grad-CAM. Using the ResNet50 model, we designate the intermediate convolutional layer of the Layer4 block as the local representation and the output of the final AvgPool layer as the global representation. As shown in Figure2, our approach decomposes the models encoded information into three distinct types: total information, decision-related information, and redundant information. This provides a more comprehensive explanation than Grad-CAM. Specifically, in the visualization of the first image, the heatmap for total information closely resembles that of Grad-CAM. In the second image, the heatmap for decision-related information aligns similarly with Grad-CAM, highlighting the nuanced insights our method offers by separating these information components.

4.2. Prototype Network Visualization

To further distinguish between total information and decision-related information, we visualize representations from the CUB-200-2011 dataset using a prototype network, which yields compelling visualization results. We treat the output of the prototype networks Block 1 as local representations and the output of the average pooling (AvgPool) layer as global representations. Figures 3, 4, 5, and 6 illustrate the heatmaps for total information and decision-related information, revealing clear and significant differences between the two.

4.2.1. Few-Shot Learning

The prototype network is a well-established method in few-shot learning, where the goal is to classify a query image into the correct category based on a support set consisting of a few labeled examples. This task is typically formulated as an N-way K-shot problem, where N represents the number of categories and K denotes the number of samples per category. For example, in a 5-way 1-shot scenario, the support set contains 5 categories, each with a single image. The prototype network classifies query images by comparing them to class prototypes derived from the support set.

For effective visualization, we adopt a 5-way 1-shot setting: the support set includes 5 categories with one image each,

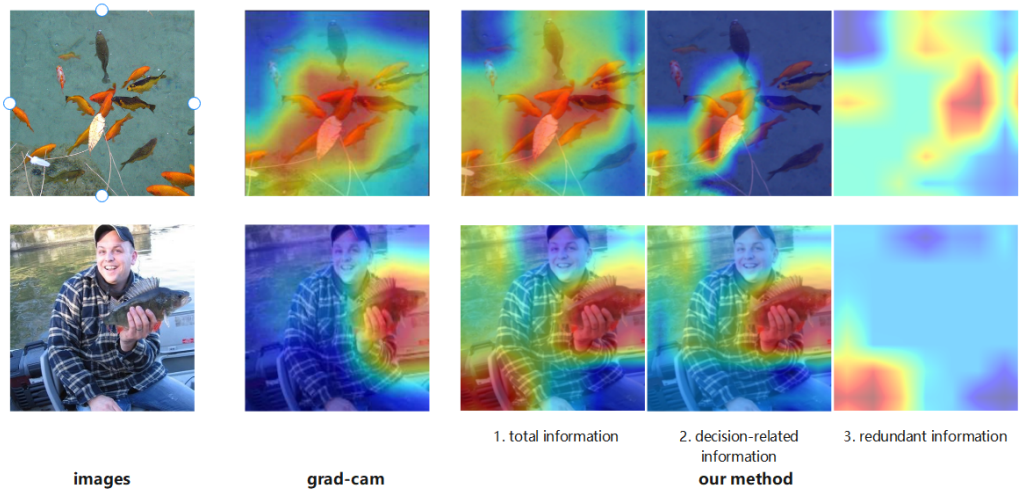


Figure 2. heatmap examples

and we select 5 query images, one from each category in the support set. Figure 3 provides an example, where each column is labeled as either support or query to distinguish whether the 5 samples belong to the support set or the query set, respectively.

5. Conclusion

We propose a mutual information (MI)-based framework that decomposes the relationship between local inputs and representations into three exhaustive components: total mutual information $I(Z, X_i)$, decision-related information $I(Z, X_i')$, and redundant information $R(Z, X_i) = I(Z, X_i) - I(Z, X_i')$. This theoretically complete framework fully explains $I(X, Z)$, surpassing partial methods like Grad-CAM by capturing all encoded content, decision drivers, and discardable noise. Using two lightweight modules, we visualize these components on ResNet and prototype network, confirming their interpretive power. This holistic approach establishes a robust tool for representation analysis.

References

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7), 2015a.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., and Samek, W. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015b.

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., and Müller, K.-R. How to explain individual classification decisions. *The Journal of Machine Learning Research*, 11:1803–1831, 2010.

Belghazi, M. I., Baratin, A., Rajeswar, S., Ozair, S., Bengio, Y., Courville, A., and Hjelm, R. D. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

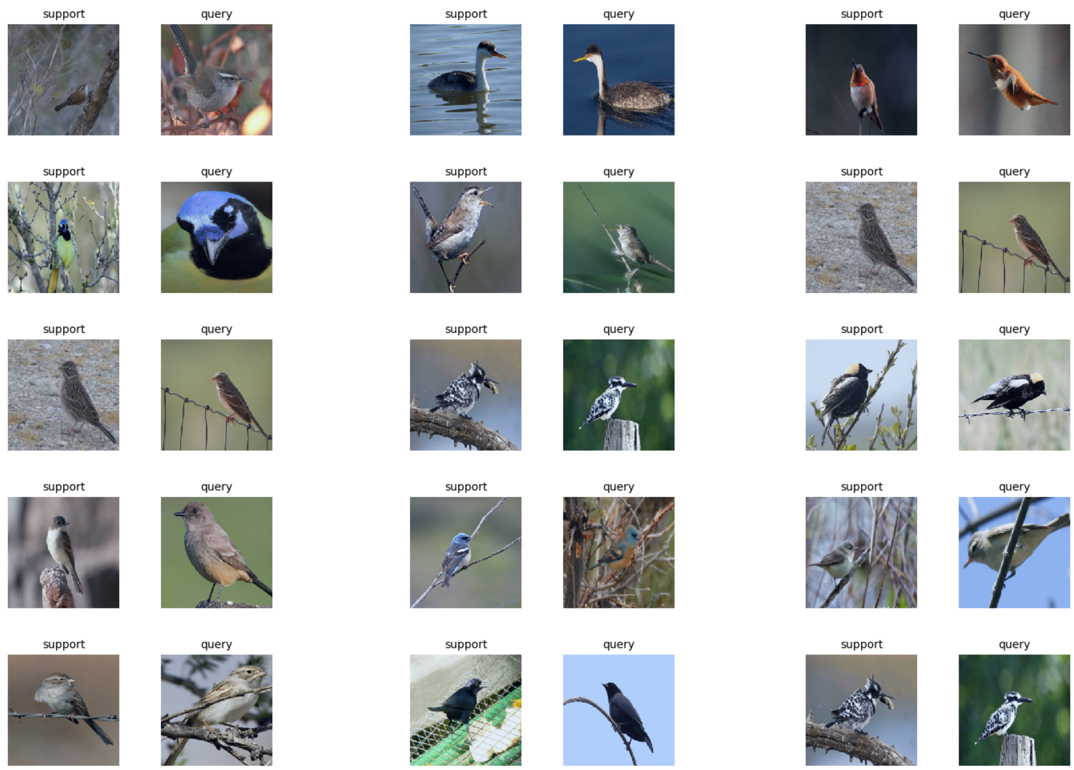
Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P. H., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z. D., Azar, M. G., et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. 2016.

Hjelm, R. D., Fedorov, A., Lavoie-Marchildon, S., Grewal, K., Bachman, P., Trischler, A., and Bengio, Y. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

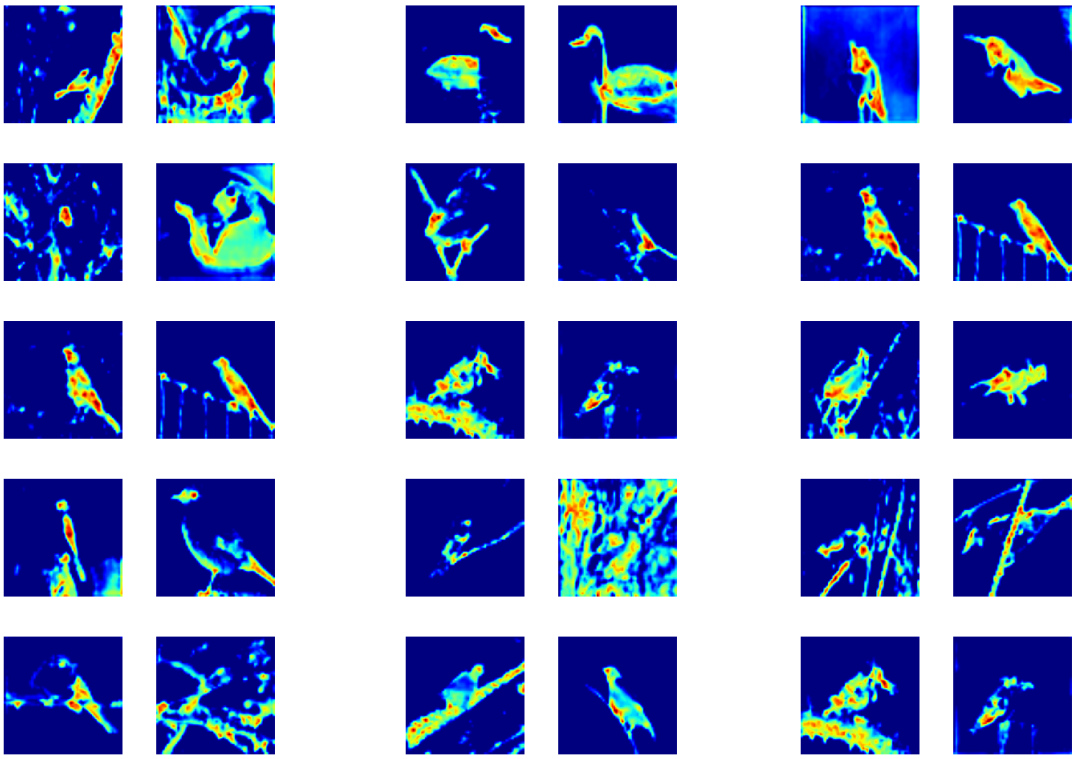
Kim, H. and Mnih, A. Disentangling by factorising. *arXiv preprint arXiv:1802.05983*, 2018.

Montavon, G., Lapuschkin, S., Binder, A., Samek, W., and Müller, K.-R. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.



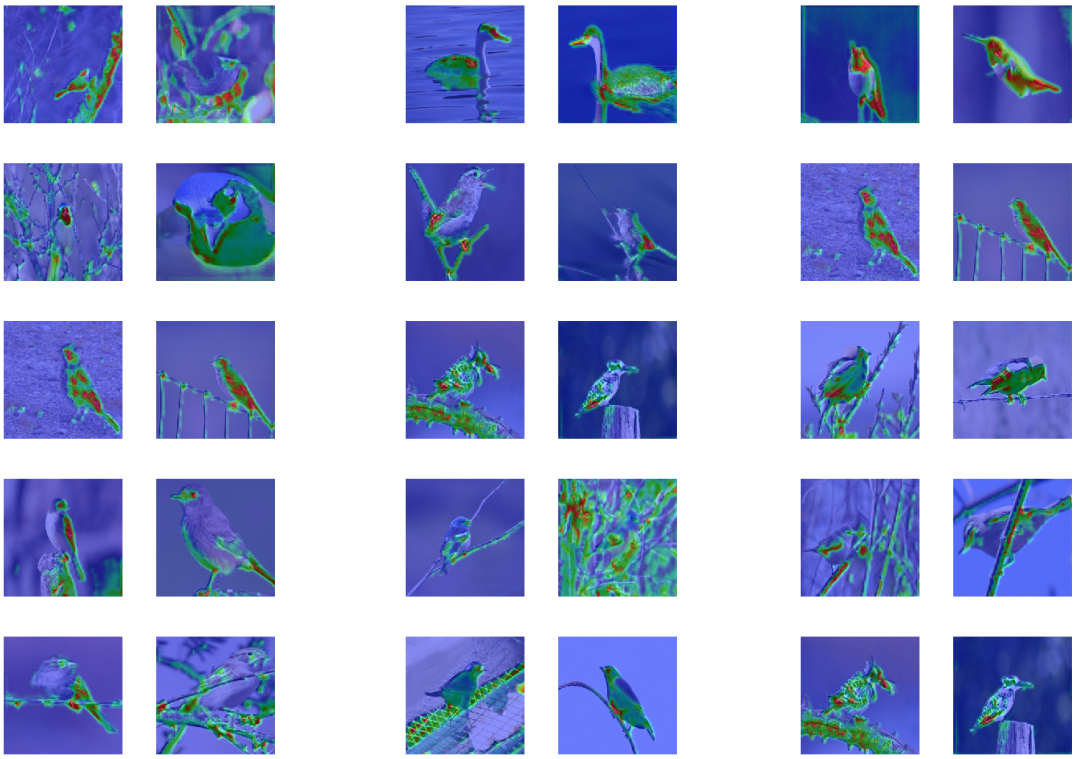
(a) (b) (c)

Figure 3. images



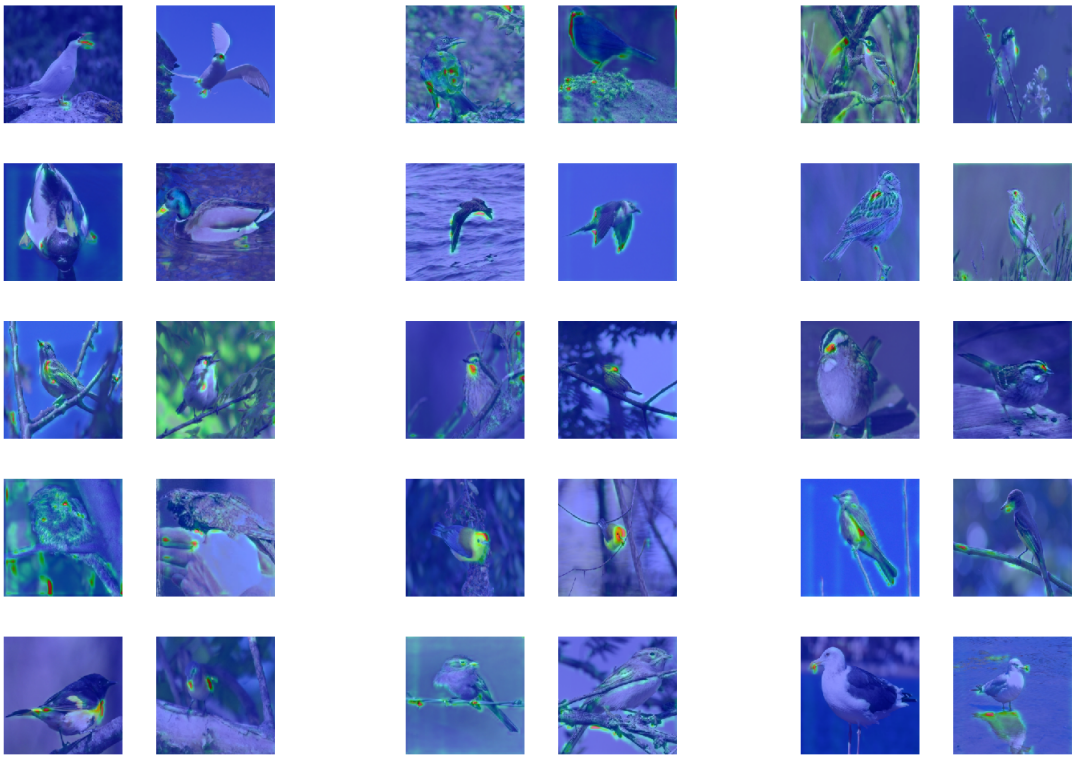
(a) (b) (c)

Figure 4. total information heat map



(a) (b) (c)

Figure 5. a mixed figure combining the total information heat map and the original image



(a) (b) (c)

Figure 6. mixed figure combining the decision-related information heat map and the original image

Schulz, K., Sixt, L., Tombari, F., and Landgraf, T. Restricting the flow: Information bottlenecks for attribution. *arXiv preprint arXiv:2001.00396*, 2020.

Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., and 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.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 3319–3328. JMLR. org, 2017.

Tishby, N. and Zaslavsky, N. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pp. 1–5. IEEE, 2015.

Tschannen, M., Djolonga, J., Rubenstein, P. K., Gelly, S., and Lucic, M. On mutual information maximization for representation learning. *arXiv preprint arXiv:1907.13625*, 2019.