CARTOON EXPLANATIONS OF IMAGE CLASSIFIERS

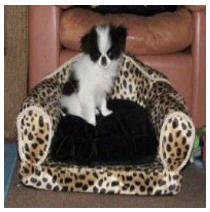
Anonymous authors
Paper under double-blind review

ABSTRACT

We present *CartoonX* (Cartoon Explanation), a novel model-agnostic explanation method tailored towards image classifiers and based on the rate-distortion explanation (RDE) framework. Natural images are roughly piece-wise smooth signals—also called cartoon images—and tend to be sparse in the wavelet domain. CartoonX is the first explanation method to exploit this by requiring its explanations to be sparse in the wavelet domain, thus extracting the *relevant piece-wise smooth* part of an image instead of relevant pixel-sparse regions. We demonstrate experimentally that CartoonX is not only highly interpretable due to its piece-wise smooth nature but also particularly apt at explaining misclassifications.

1 INTRODUCTION

Powerful machine learning models such as deep neural networks are inherently opaque, which has motivated numerous explanation methods over the last decade (see for example the survey by Das & Rad (2020)). A significant fraction of the research literature has focused on explaining image classifications due to both the practical relevance of computer vision tasks and the ease at which heatmaps can communicate explanatory information. Despite the great variety in methods and explanation philosophies, all current methods share the following characteristic: they operate in pixel space. Roughly speaking, existing explanation methods for image classifiers either allocate additive attribution scores to each pixel or optimize a deletion mask on the pixel coefficients to mark a relevant set of pixels. The result is typically a pixel-sparse and jittery explanation. We challenge the conventional approach to explain in pixel space by successfully applying the rate-distortion explanation (RDE) framework (Macdonald et al., 2019; Heiß et al., 2020) in the wavelet domain of images. Our novel explanation method, *CartoonX*, extracts the relevant piece-wise smooth part of an image (see Figure 1). Instead of demanding sparsity in pixel space, as in (Macdonald et al., 2019; Chang et al., 2019), CartoonX demands sparsity in the wavelet domain, which produces piece-wise smooth explanations (cartoon-like images). Our work makes the following contributions:



Dog classified as Egyptian cat

CartoonX of misclassification

Slam dunk classified as basketball

CartoonX of classification

Figure 1: Examples of CartoonX explanations.

Reformulation and reinterpretation of the RDE framework: We reformulate the RDE framework in a more general manner with enhanced flexibility in the input representation to accommodate complex interpretation queries such as "What is the piece-wise smooth part of the input signal that leads to its model decision?". Thereby, we reinterpret RDE as a simplification of the input signal, which is interpretable to humans and adheres to a meaningful interpretation query. The simplification is achieved by demanding sparsity in a suitable representation system, which sparsely represents the class of explanations that are desirable for the interpretation query.

CartoonX, a novel explanation method tailored to image classifiers: CartoonX is the first explanation method to extract the relevant piece-wise smooth part of an image instead of relevant pixel sparse regions. This is achieved by demanding sparsity in the wavelet domain of images, where

sparsity translates into piece-wise smooth images. We demonstrate that our piece-wise smooth explanations are more interpretable than jittery pixel-sparse explanations and that they can reveal relevant piece-wise smooth patterns that are not easily visible with existing pixel-based methods. Surprisingly, we find that our method is particularly well-equipped to explain misclassifications, often showing "what the neural network actually saw" (see Figure 1).

2 Related Work

The Rate-Distortion Explanation (RDE) framework was first introduced in (Macdonald et al., 2019), and extended in (Heiß et al., 2020), as a mathematically well-founded and intuitive explanation framework. RDEs are model-agnostic explanations and inspired by rate-distortion theory, which studies lossy-data compression. An explanation in RDE consists of a relatively sparse mask over the input features, highlighting the relevant set of features. The mask is optimized to produce low distortion in the model output after applying perturbations to the unselected features in the input while remaining relatively sparse. Heiß et al. (2020) also applied RDE to non-canonical input representations to explain model decisions in challenging domains such as audio classification (Engel et al., 2017) and radio-map estimation (Levie et al., 2021; 2020).

The explanation principle of optimizing a mask $s \in [0,1]^n$ was first proposed by Fong & Vedaldi (2017) who explained image classification decisions by considering one of the two "deletion games": (1) optimizing for the smallest deletion mask that causes the class score to drop significantly or (2) optimizing for the largest deletion mask that has no significant effect on the class score. The original RDE approach (Macdonald et al., 2019) is based on the second deletion game.

Other explanation methods developed by the research community are typically either (1) gradient-based such as Smoothgrad (Smilkov et al., 2017), Integrated Gradients (Sundararajan et al., 2017), Image-Specific Class Saliency (Simonyan et al., 2014), and Guided Backpropagation (Springenberg et al., 2015), (2) surrogate models such as LIME (Ribeiro et al., 2016), (3) based on propagation of activations in neurons such as LRP (Bach et al., 2015; Shrikumar et al., 2017), and DeepLIFT (Shrikumar et al., 2017), (4) based on Shapely values from game-theory (Lundberg & Lee, 2017), (6) concept-based such as Concept Activation Vectors (Kim et al., 2018), or (7) based on generative causal explanations (O' Shaughnessy et al., 2020). Also related are methods that were developed to explain individual neurons such as in (Nguyen et al., 2016; Dhamdhere et al., 2019). To our knowledge, all existing explainability methods operate in pixel space and all methods looking for sparse explanations demand sparsity in pixel space (Macdonald et al., 2019; Fong & Vedaldi, 2017; Chang et al., 2019).

3 Background: Rate-Distortion Explanation Framework

In this section, we review the rate-distortion explanation (RDE) framework, which was introduced by Macdonald et al. (2019) and later extended by Heiß et al. (2020) by applying RDE to non-canonical input representations. Suppose $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ is a pre-trained model, e.g., a classifier (with m class labels) or a regression model (with m-dimensional output), where n denotes the dimension of the model input. RDE produces an explanation for a model decision $\Phi(x)$ with $x \in \mathbb{R}^n$ as a relatively sparse mask $s \in \{0,1\}$ marking the relevant input features in x. More precisely, RDE aims to solve the following optimization problem over a mask $s \in \{0,1\}^n$:

$$\min_{s \in \{0,1\}^n} \quad \mathop{\mathbb{E}}_{v \sim \mathcal{V}} \left[d\Big(\Phi(x), \Phi(x \odot s + (1-s) \odot v)\Big) \right] \quad \text{s.t.} \quad \|s\|_0 \leq \ell, \tag{1}$$

where $\odot$ denotes the Hadamard product (element-wise multiplication), $d(\Phi(x), \cdot)$ is a measure of distortion (e.g. $d(\Phi(x), \cdot) = \|\Phi(x) - \cdot\|_2$), $\mathcal{V}$ is a distribution over input perturbations $v \in \mathbb{R}^n$, and $\ell \in \{1, ..., n\}$ is a given sparsity level for the explanation mask s. A solution s^* to the optimization problem in (1) marks relatively few components in the model input x that suffice to approximately retain the model output $\Phi(x)$. This approach is in the spirit of rate-distortion theory, which deals with lossy compression of data. Therefore, Macdonald et al. (2019) coined such explanations *rate-distortion explanations* (RDEs).

In practice, the optimization problem in (1) is relaxed to continuous masks $s \in [0,1]$ solving

$$\min_{s\in\{0,1\}^n} \quad \mathbb{E}_{v\sim\mathcal{V}}\left[d\Big(\Phi(x), \Phi(x\odot s + (1-s)\odot v)\Big)\right] + \lambda \left\|s\right\|_1, \tag{2}$$

where $\lambda > 0$ determines the sparsity level of the mask. The relaxed optimization problem can be solved with stochastic gradient descent in $s \in [0,1]$ if Φ is differentiable—as is the case for deep neural networks. Macdonald et al. (2019) applied the RDE method as described above to image classifiers in the pixel domain of images, where each mask entry $s_i \in [0,1]$ corresponds to the i-th pixel values. We refer to this method as *Pixel RDE* throughout this work.

4 RDE Reformulated and Reinterpreted

Instead of applying RDE to the standard input representation $x = [x_1 \dots x_n]^T$, we can apply RDE to a different representation of x to answer a particular interpretation query. For example, consider a 1D-signal $x \in \mathbb{R}^n$: if we ask "What is the smooth part in the signal x that leads to the model decision $\Phi(x)$?", then we can apply RDE in the Fourier basis of x. Since frequency-sparse signals are smooth, applying RDE in the Fourier basis of x extracts the relevant smooth part of the signal. To accommodate such interpretation queries, we reformulate RDE in Section 4.1. Finally, based on the reformulation, we reinterpret RDE in Section 4.2. Later in Section 5, we use our reformulation and reinterpretation of RDE to derive and motivate CartoonX as a special case and novel explanation method tailored towards image classifiers.

4.1 General Formulation

An input signal $x = [x_1, \dots, x_n]^T$ is represented in a basis $\{b_1, \dots, b_n\}$ as a linear combination $\sum_{i=1}^n h_i b_i$ with coefficients $[h_i]_{i=1}^n$. As we argued above and demonstrate later on, some choices for a basis may be more suitable than others to explain a model decision $\Phi(x)$. Therefore, we define the RDE mask not only on the canonical input representation $[x_i]_{i=1}^n$ but also on a different representation $[h_i]_{i=1}^n$ with respect to a choice of basis $\{b_1, \dots, b_n\}$. Examples of non-canonical choices for a basis include the Fourier basis and the wavelet basis. This work is centered around CartoonX, which applies RDE in the wavelet basis, i.e., a linear data representation since x is represented as a linear combination of basis vectors. Nevertheless, there also exist other domains and interpretation queries where applying RDE to a non-linear data representation can make sense (see the interpretation query "Is phase or magnitude more important for an audio classifier?" in (Heiß et al., 2020)). Therefore, we formulate RDE in terms of a data representation function $f : \prod_{i=1}^k \mathbb{R}^c \to \mathbb{R}^n,\ f(h_1, \dots, h_k) = x$, which does not need to be linear and allows to mask c channels in the input at once. In the important linear case and $c = 1$, we have $f(h_1, \dots, h_k) = \sum_{i=1}^k h_i b_i$, where $\{b_i, \dots, b_k\} \subset \mathbb{R}^n$ are k fixed vectors that constitute a basis. The case $c > 1$ is useful when one wants to mask out several input channels at once, e.g., all color channels of an image, to reduce the number of entries in the mask that will operate on $[h_i]_{i=1}^k$. In the following, we introduce the important definitions of *obfuscations*, *expected distortion*, *the RDE mask*, and *RDE's ℓ_1-relaxation*, which generalize the RDE framework of (Macdonald et al., 2019) to abstract input representations.

4.1.1 Definitions

The first two key concepts in RDE are *obfuscations* and *expected distortion*, which are defined below.

Definition 1 (Obfuscations and expected distortion) *Let $\Phi : \mathbb{R}^n \to \mathbb{R}^m$ be a model and $x \in \mathbb{R}^n$ a data point with a data representation $x = f(h_1, ..., h_k)$ as discussed above. For every mask $s \in [0,1]^k$, let $\mathcal{V}_s$ be a probability distribution over $\prod_{i=1}^k \mathbb{R}^c$. Then the* obfuscation *of x with respect to s and $\mathcal{V}_s$ is defined as the random vector $y := f(s \odot h + (1-s) \odot v)$, where $v \sim \mathcal{V}_s$, $(s \odot h)_i = s_i h_i \in \mathbb{R}^c$ and $((1-s)\odot v)_i = (1-s_i)v_i \in \mathbb{R}^c$, for $i \in \{1, \dots, k\}$. A choice for the distribution $\mathcal{V}_s$ is called* obfuscation strategy. *Furthermore, the* expected distortion *of x with respect to the mask s and the perturbation distribution $\mathcal{V}_s$ is defined as*

$$D(x, s, \mathcal{V}_s, \Phi) := \mathbb{E}_{v\sim\mathcal{V}_s}\left[d\Big(\Phi(x), \Phi(y)\Big)\right],$$

where $d : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}_+$ *is a measure of distortion between two model outputs.*

In the RDE framework, the explanation is given by a mask that minimizes distortion while remaining relatively sparse. The rate-distortion explanation mask is defined as follows.

Definition 2 (The RDE mask) *In the setting of Definition 1 we define the* RDE mask *as a solution* $s^*(\ell)$ *to the minimization problem*

$$\min_{s \in \{0,1\}^k} \quad D(x, s, \mathcal{V}_s, \Phi) \quad \textit{s.t.} \quad \|s\|_0 \leq \ell, \tag{3}$$

where $\ell \in \{1, \ldots, k\}$ *is the desired level of sparsity.*

Geometrically, the RDE mask s is associated with a particular subspace. The complement mask $(1 - s)$ can be seen as selecting a large stable subspace of Φ, with each point representing a possible perturbation in unselected coefficients in h. The RDE mask minimizes the expected distortion along its associated subspace, which requires non-local information of Φ. We illustrate this geometric view of RDE in Figure 2 with a toy example for a hypothetical classifier $\Phi : \mathbb{R}^2 \to \mathbb{R}^m$ and two distinct input representations: (1) Euclidean coordinates, i.e., f is the identity in $x = f(h)$, and (2) polar coordinates, i.e. $f(h) = (h_2 \cos h_1, h_2 \sin h_1) = x$. In the example, we assume $\mathcal{V}_s$ to be a uniform distribution on $[-1, 1]^2$ in the Euclidean representation and a uniform distribution on $[-\pi, \pi] \times [0, 1]$ in the polar representation. The expected distortion associated with the masks $s = (1, 0)$ and $s = (0, 1)$ is given by the red and green shaded area, respectively. The RDE mask aims for low expected distortion, and hence, in polar coordinates, the RDE mask would be the green subspace, i.e., $s = (0, 1)$. On the other hand, in Euclidean coordinates, neither $s = (1, 0)$ nor $s = (0, 1)$ produces a particularly low expected distortion, making the Euclidean explanation less meaningful than the polar explanation. The example illustrates why certain input representations

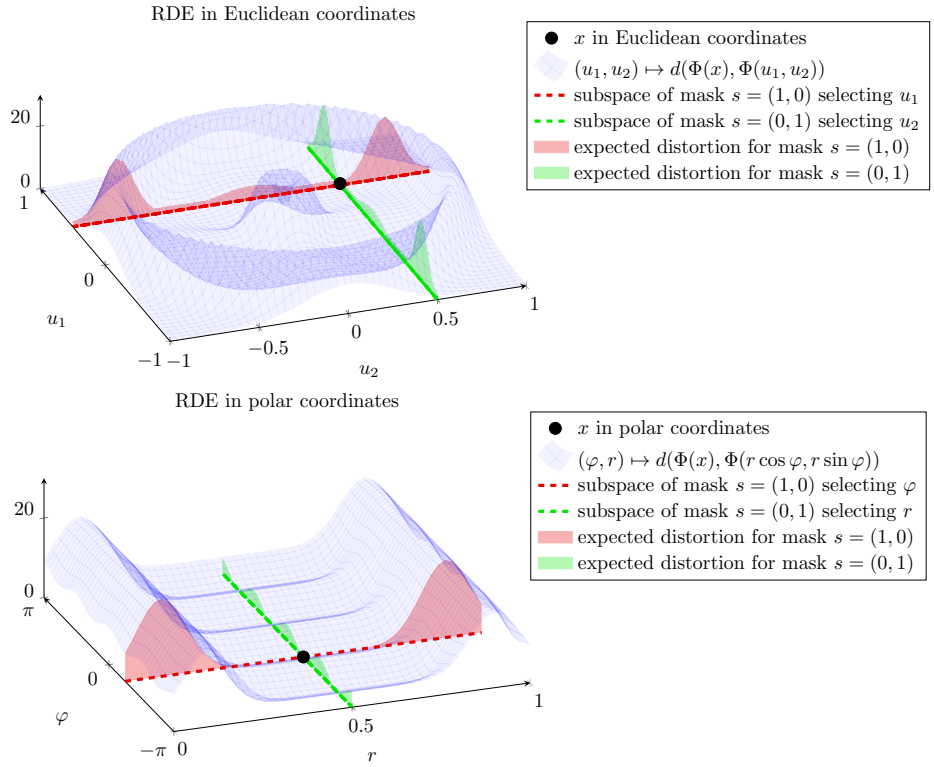


Figure 2: The RDE mask can find low expected distortion in polar coordinates but not in Euclidean coordinates. Therefore, in this example, polar coordinates are more appropriate to explain $\Phi(x)$, and RDE would determine that the angle φ, not the magnitude r, is relevant for $\Phi(x)$.

can yield more meaningful explanatory insight for a given classifier than others—an insight that

underpins our novel CartoonX method. Moreover, the plot in polar coordinates illustrates why the RDE mask cannot be simply chosen with local distortion information, e.g., with the lowest eigenvalue of the Hessian of $h \mapsto d(\Phi(x), \Phi(f(h)))$: the lowest eigenvalue in polar coordinates belongs to the red subspace and does not see the large distortion on the tails.

As was shown by Macdonald et al. (2019), the RDE mask from Definition 2 cannot be computed efficiently for non-trivial input sizes. Nevertheless, one can find an approximate solution by considering continuous masks $s \in [0,1]^k$ and encouraging sparsity through the ℓ_1-norm.

Definition 3 (RDE's ℓ_1-relaxation with Lagrange multipliers) *In the setting of Definition 1, we define* RDE's ℓ_1-relaxation with Lagrange multipliers *as a solution $s^*(\lambda)$ to the minimization problem*

$$\min_{s\in[0,1]^k} \quad D(x, s, \mathcal{V}_s, \Phi) + \lambda\|s\|_1, \tag{$\mathcal{P}_1$}$$

where $\lambda > 0$ is a hyperparameter for the sparsity level.

The ℓ_1-relaxation above can be solved with stochastic gradient descent (SGD) over the mask s while approximating $D(x, s, \mathcal{V}_s, \Phi)$ with i.i.d. samples from $v \sim \mathcal{V}_s$.

4.1.2 Obfuscation Strategies

An obfuscation strategy is defined by the choice of the perturbation distribution $\mathcal{V}_s$. Common choices are Gaussian noise (Macdonald et al., 2019; Fong & Vedaldi, 2017), blurring (Fong & Vedaldi, 2017), constants (Fong & Vedaldi, 2017), and inpainting GANs (Heiß et al., 2020; Chang et al., 2019). Inpainting GANs train a generator $G(s, z, h)$ (z denotes random latent factors) such that for samples $v \sim G(s, z, h)$ the obfuscation $f(s \odot h + (1-s) \odot v)$ remains in the data manifold. In our work, we refrain from using an inpainting GAN due to the following reason: it is hard to tell whether a GAN-based mask did not select coefficients because they are unimportant or because the GAN can easily inpaint them from a biased context. Instead, we choose a simple and well-understood obfuscation strategy, which we call *Gaussian adaptive noise*, making the explanation as transparent as possible.

Gaussian adaptive noise works as follows: Let $A_1, ..., A_j$ be a pre-defined choice of a disjoint partition of $\{1, \dots, k\}$ (recall $s \in [0,1]^k$). For $i = 1, ..., j$, we compute the empirical mean and empirical standard deviation for each partition across all partition instances:

$$\mu_i \coloneqq \frac{1}{\sum_{a\in A_i} d_a} \sum_{a\in A_i, t=1,\dots,d_a} h_{at}, \quad \sigma_i \coloneqq \sqrt{\frac{1}{\sum_{a\in A_i} d_a} \sum_{a\in A_i, t=1,\dots,d_a} (\mu_i - h_{at})^2}$$

The adaptive Gaussian noise strategy then samples $v_{at} \sim \mathcal{N}(\mu_i, \sigma_i^2)$ for all partition members $a \in A_i$ and channels $t = 1, ..., d_a$. We write $v \sim \mathcal{N}(\mu, \sigma^2)$ for the resulting Gaussian random vector $v \in \prod_{i=1}^k \mathbb{R}^c$. Note that the distribution $\mathcal{V}_s$ chosen as Gaussian adaptive noise does depend on s (unlike with an inpainting GAN). For Pixel RDE, we only use one set $A_1 = \{1, ..., k\}$ for all k pixels. In CartoonX, which represents input signals in the discrete wavelet domain, we will partition $\{1, ..., k\}$ along the scales of the discrete wavelet transform.

4.1.3 Measures of distortion

There are various choices for the measure of distortion $d(\Phi(x), \Phi(y))$. For example, one can take the squared distance in the post-softmax probability of the predicted label for x, i.e.,

$$d\big(\Phi(x), \Phi(y)\big) \coloneqq \big(\Phi_{j^*}(x) - \Phi_{j^*}(y)\big)^2,$$

where $j^* \coloneqq \arg\max_{i=1,\dots,m} \Phi_i(x)$ and $\Phi(x)$ is assumed to be the post-softmax probabilities of a neural net. Alternatively, one could also choose $d(\Phi(x), \Phi(y))$ as the ℓ_2-distance or the KL-Divergence in the post-softmax layer of Φ. In our experiments for CartoonX, we found that these choices had no significant effect on the explanation (see Appendix A.3.3).

4.2 INTERPRETATION

The philosophy of the generalized RDE framework is that an explanation for a decision $\Phi(x)$ on a generic input signal $x = f(h)$ should be some simplified version of the signal, which is interpretable to humans. The simplification is achieved by demanding sparsity in a suitable representation system h, *which sparsely represents the class of explanations that are desirable for the interpretation query*. This philosophy is the fundamental premise of CartoonX, which aims to answer the interpretation query *"What is the relevant piece-wise smooth part of the image for a given image classifier?"*. CartoonX first employs RDE on a representation system $x = f(h)$ that sparsely represents piece-wise smooth images and finally visualizes the relevant piece-wise smooth part as an image back in pixel space. In the following section, we explain why wavelets provide an appropriate representation system in CartoonX, present the CartoonX implementation, and finally provide experiments on ImageNet to demonstrate the capability of CartoonX.

5 CARTOONX

The focus of this paper is *CartoonX*, a novel explanation method—tailored to image classifications—that we obtain as a special case of our generalized RDE framework formulated in Section 4. CartoonX first performs RDE in the discrete wavelet position-scale domain of an image x, and finally, visualizes the wavelet mask s as a piece-wise smooth image in pixel space. Wavelets provide optimal representations for piece-wise smooth 1D functions (DeVore, 1998), and represent 2D piece-wise smooth images, also called *cartoon-like images* (Kutyniok & Lim, 2011), efficiently as well (Romberg et al., 2006). In particular, sparse vectors in the wavelet coefficient space encode cartoon-like images reasonably well (Stéphane, 2009a)—certainly better than sparse pixel representations. Moreover, wavelets constitute an established tool in signal processing (Stéphane, 2009c).

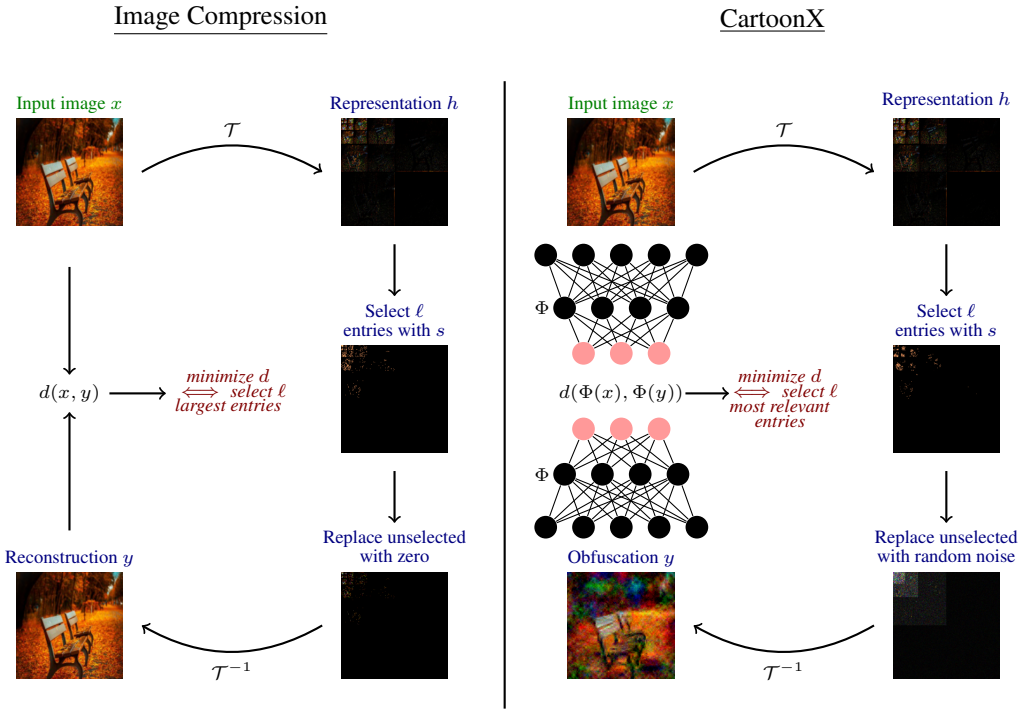


Figure 3: CartoonX has many interesting parallels to wavelet-based image compression. Distortion is denoted as d, Φ is an image classifier, h denotes the discrete wavelet coefficients, $\mathcal{T}$ is the discrete wavelet transform, and ℓ is the coefficient budget.

The optimization process underlying CartoonX produces sparse vectors in the discrete wavelet coefficient space, which results in cartoon-like images as explanations. This is the fundamental dif-

ference to Pixel RDE, which produces rough, jittery, and pixel-sparse explanations. Cartoon-like images are more interpretable and provide a natural model of simplified images. Since the goal of the RDE framework is to generate an easy to interpret simplified version of the input signal, we argue that CartoonX explanations are more appropriate for image classification than Pixel RDEs.

CartoonX exhibits interesting parallels to wavelet-based image compression. In image compression, distortion is minimized in the data domain, which is equivalent to selecting the ℓ largest entries in the discrete wavelet transform (DWT) coefficients. In comparison, CartoonX minimizes distortion in the model output of Φ, which translates to selecting the ℓ most relevant entries in the DWT coefficients. The objective in image compression is efficient data representation, i.e., producing minimal data distortion with a budget of ℓ entries in the DWT coefficients. Conversely, in CartoonX, the objective is extracting the relevant piece-wise smooth part, i.e., producing minimal model distortion with a budget of ℓ entries in the DWT coefficients. We illustrate this connection in Figure 3—highlighting once more the *rate-distortion* spirit of the RDE framework.

5.1 Implementation

An image $x \in [0,1]^{c \times w \times t}$ with $c \in \{1,3\}$ channels, width $w \in \mathbb{N}$, height $t \in \mathbb{N}$, and a total of $p = wt$ pixels can be represented in a wavelet basis by computing its discrete wavelet transform (DWT). The DWT of an image is defined by the number of scales $J \in \{1, \dots, \lfloor \log_2 p \rfloor\}$, the padding mode, and a choice of the wavelet family (such as the Haar or Daubechies family). For images, the DWT computes four types of coefficients: details in (1) horizontal, (2) vertical, and (3) diagonal orientation at scale $j \in \{1, \dots, J\}$, and (4) coefficients of the image at the very coarsest resolution. We briefly illustrate the DWT for an example image in Figure 4.

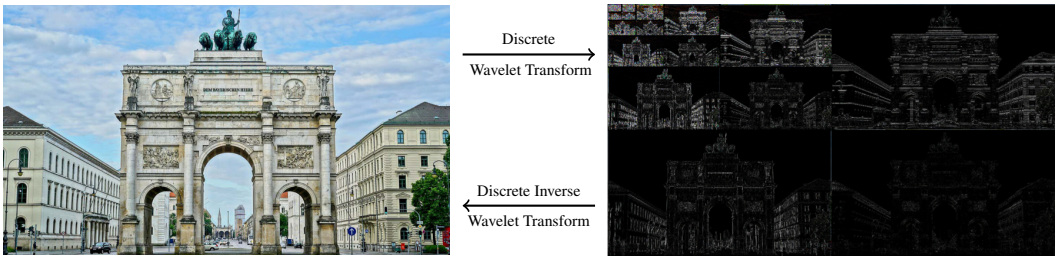


Figure 4: Left side: an image of a memorial arch dedicated to peace. Right side: visualization of the DWT coefficients for five scales. Three L-shaped sub-images describe coefficients for details in vertical, horizontal, and diagonal orientation at a particular scale. The largest sub-images (the outer L-shape) belong to the lowest scale, i.e., the highest resolution. The smaller L-shaped sub-images gradually build up to higher scales, i.e., lower resolution features.

CartoonX, as described in Algorithm 1 in Appendix A.1, computes the RDE mask in the wavelet domain of images. More precisely, for the data representation $x = f(h)$, we choose h as the concatenation of all the DWT coefficients along the channels, i.e., $h_i \in \mathbb{R}^c$. The representation function f is then the discrete inverse wavelet transform, i.e., the summation of the DWT coefficients times the DWT basis vectors. We optimize the mask $s \in [0,1]^k$ on the DWT coefficients $[h_1, \dots, h_k]^T$ to minimize RDE's ℓ_1-relaxation from Definition 3. For the obfuscation strategy $\mathcal{V}_s$, we use adaptive Gaussian noise with a partition by the DWT scale (see Section 4.1.2), i.e., we compute the empirical mean and standard deviation per scale. We measure distortion as the squared difference in the post-softmax score of the predicted label for x (see Section 4.1.3). To visualize the final DWT mask s as a piece-wise smooth image in pixel space, we multiply the mask with the DWT coefficients of the greyscale image $\hat{x} := (1/c \sum_{l=1}^{c} x_{lai})_{ai}$ before inverting the product back to pixel space with the discrete inverse wavelet transform. The inversion is finally clipped into $[0,1]^{w \times t}$ as are obfuscations during the RDE optimization to avoid overflow (we assume here the pixel values in x are normalized into $[0,1]$). The clipped inversion in pixel space is the final explanation, which we call CartoonX.

5.2 EXPERIMENTS AND ANALYSIS

We compare CartoonX to the closely related Pixel RDE (Macdonald et al., 2019) and several other state-of-the-art explanation methods , that is, Integrated Gradients (Sundararajan et al., 2017), Smoothgrad (Smilkov et al., 2017), Guided Backprop (Springenberg et al., 2015), and LRP (Bach et al., 2015). Our experiments show that CartoonX carries the following strengths: Cartoon X is (1) highly interpretable due to its cartoon-like nature and (2) remarkably apt at explaining misclassifications, and highlighting meaningful patterns that are otherwise hard to see. Due to the fast implementation of the DWT, Cartoon RDE is not significantly slower than Pixel RDE. For the ImageNet classifier MobileNetV3-Small, an image of 256 times 256 pixels, and 2001 optimization steps, we reported a runtime of 81.56 seconds for CartoonX and 70.53 seconds for Pixel RDE on the NVIDIA Titan RTX GPU. However, like other perturbation-based methods, CartoonX is significantly slower than gradient or propagation-based methods, which only compute a single or few forward and backward passes and are very fast (Integrated Gradients computes an explanation in 0.48 seconds for the same image, model, and hardware).

Our experiments use the pre-trained ImageNet classifiers MobileNetV3-Small (Howard et al., 2019) (top-1 accuracy of 67.668%) and VGG16 (Simonyan & Zisserman, 2015) (top-1 accuracy of 71.592%). We note that the open-source implementation of LRP did not implement propagation rules for certain layers in MobileNetV3-Small, therefore we compare CartoonX to LRP only for VGG16. Images were preprocessed to have 256 times 256 pixel values in $[0, 1]$. We provide further details about the choice of hyperparameters in the experiments in Appendix A.2. The three main

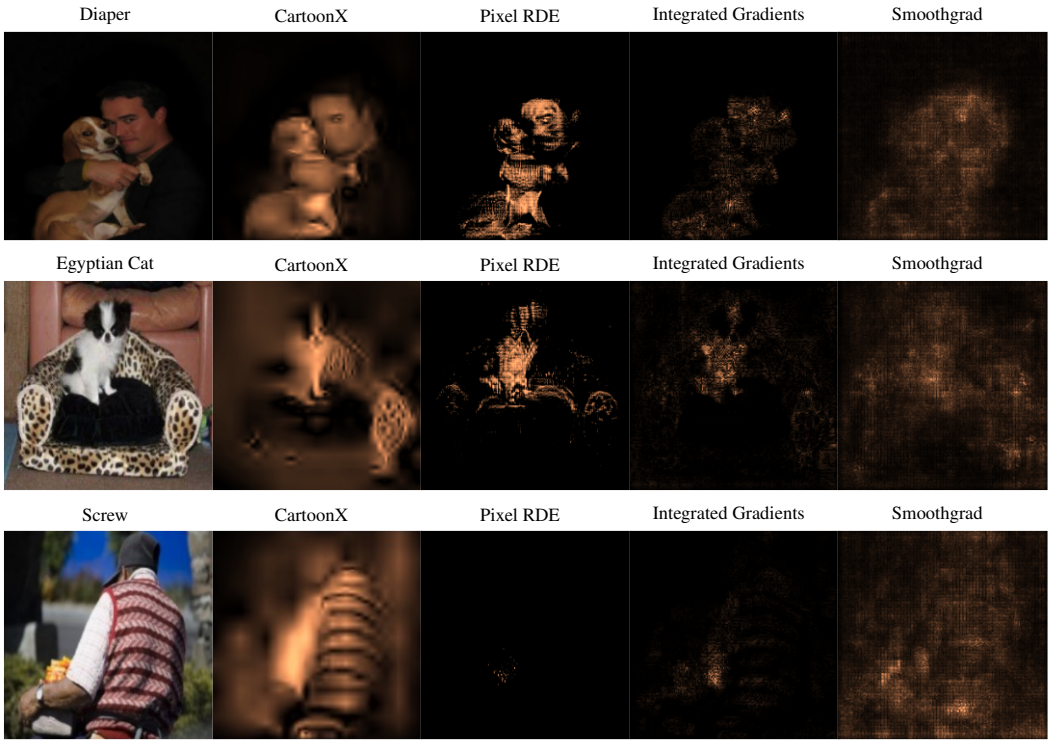


Figure 5: Each row compares CartoonX explanations of misclassifications by MobileNetV3-Smal to Pixel RDE (Macdonald et al., 2019), Integrated Gradients (Sundararajan et al., 2017), and Smoothgrad (Smilkov et al., 2017). The predicted label is depicted above each misclassified image.

hyperparameters for CartoonX are: (1) the sparsity level $\lambda > 0$, (2) the measure of distortion d, and (3) the obfuscation strategy (perturbation distribution) $\mathcal{V}_s$. We discuss the sensitivity of CartoonX to these hyperparameters in Appendix A.3. In Appendix A.5, we also shed light on the evolution of ImageNet classifiers from an explanation angle by comparing CartoonX explanations for classi-

fiers of varying generalization power, i.e., AlexNet (Krizhevsky et al., 2012), VGG16 (Simonyan & Zisserman, 2015), InceptionV3 (Szegedy et al., 2016), ResNeXt50 (Xie et al., 2017). Moreover, in Appendix A.4, we argue experimentally why CartoonX is less susceptible than Pixel RDE to so-called *explanation artifacts*—an unwanted phenomenon that we observed empirically.

In practice, explaining misclassifications is particularly relevant since good explanations can pinpoint model biases and causes for model failures. We observe that CartoonX is particularly good at explaining certain misclassifications, which we illustrate for three examples in Figure 5 and many more in Appendix A.6. In the first row in Figure 5, the input image shows a man holding a dog that was classified as a "diaper". CartoonX shows the man not holding a dog but a baby, revealing that the neural net associated diapers with babies and babies with the pose with which the man is holding the dog. In the second row, the input image shows a dog sitting on an armchair with leopard patterns. The dog was classified as an "Egyptian cat", which can exhibit leopard-like patterns. CartoonX exposes the Egyptian cat by connecting the dog's head to parts of the armchair forming the cat's torso and legs. In the last row, the input image displays the backside of a man wearing a striped sweater that was classified as a "screw". CartoonX reveals how the stripe patterns look like a screw to the neural net.

Figure 6 further compares Cartoon explanations of correct classifications by VGG16. We also compare CartoonX on random ImageNet samples in Appendix A.7 and also show failures for CartoonX in Appendix A.8 to provide maximal transparency and fair comparison.

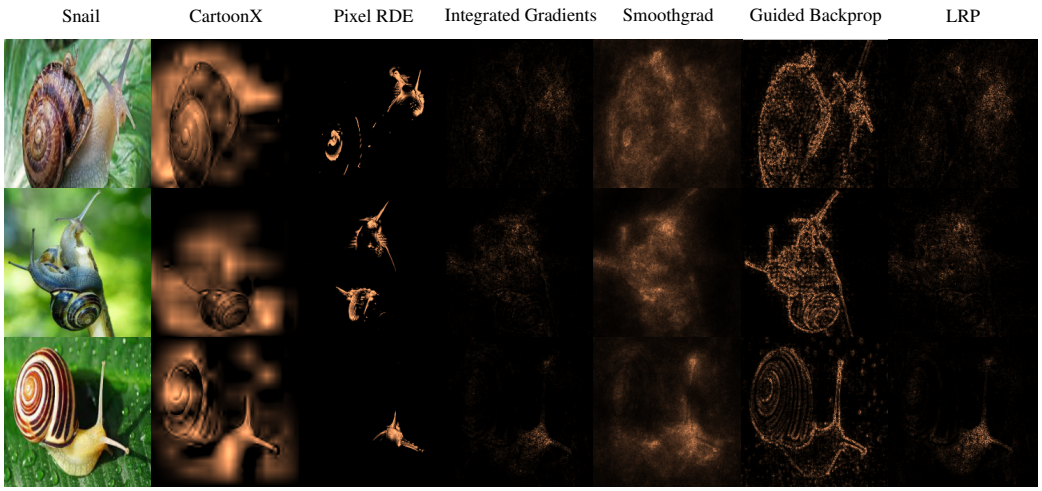


Figure 6: CartoonX explanations for VGG16 compared to state-of-the-art methods, that is, Pixel RDE (Macdonald et al., 2019), Integrated Gradients (Sundararajan et al., 2017), Smoothgrad (Smilkov et al., 2017), Guided Backprop (Springenberg et al., 2015), and LRP (Bach et al., 2015).

6 Conclusion

CartoonX is the first explainability method to extract the relevant piece-wise smooth part of an image and is based on our novel formulation of the RDE framework. We corroborated experimentally that CartoonX explanations are highly interpretable due to their cartoon-like nature and surprisingly well-suited to explain misclassifications. Nonetheless, Cartoon RDE is still computationally quite expensive, like other perturbation-based explanation methods. In the future, we hope to devise new techniques to speed up the runtime for CartoonX. Moreover, we are pursuing applications of CartoonX beyond explanation tasks, such as detecting adversarial examples. We believe CartoonX is a valuable new explanation method for practitioners and potentially a great source of inspiration for future explanation methods aiming to tailor their explanations to other data domains. Our reformulation and reinterpretation of the RDE framework provide a blueprint for such future work: First, formulate an interpretation query related to the underlying model task, then find a representation system that sparsely represents the class of desirable explanations for the interpretation query.

REFERENCES

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7):e0130140, 2015.

Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. Explaining image classifiers by counterfactual generation. In *Proceedings of the 7th International Conference on Learning Representations, ICLR*, 2019.

Arun Das and Paul Rad. Opportunities and challenges in explainable artificial intelligence (xai): A survey. *ArXiv*, abs/2006.11371, 2020.

Ronald A. DeVore. Nonlinear approximation. *Acta Numerica*, 7:51–150, 1998.

Kedar Dhamdhere, Mukund Sundararajan, and Qiqi Yan. How important is a neuron. In *International Conference on Learning Representations*, 2019.

Jesse Engel, Cinjon Resnick, Adam Roberts, Sander Dieleman, Mohammad Norouzi, Douglas Eck, and Karen Simonyan. Neural audio synthesis of musical notes with wavenet autoencoders. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70, pp. 1068–1077, 2017.

R. C. Fong and A. Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3449–3457, 2017.

Cosmas Heiß, Ron Levie, Cinjon Resnick, Gitta Kutyniok, and Joan Bruna. In-distribution interpretability for challenging modalities. *Preprint arXiv:2007.00758*, 2020.

Andrew Howard, Mark Sandler, Bo Chen, Weijun Wang, Liang-Chieh Chen, Mingxing Tan, Grace Chu, Vijay Vasudevan, Yukun Zhu, Ruoming Pang, Hartwig Adam, and Quoc Le. Searching for MobileNetV3. In *Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1314–1324, 2019.

Been Kim, M. Wattenberg, J. Gilmer, Carrie J. Cai, James Wexler, F. Viégas, and R. Sayres. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *ICML*, 2018.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

Gitta Kutyniok and Wang-Q Lim. Compactly supported shearlets are optimally sparse. *Journal of Approximation Theory*, 163(11):1564–1589, 2011. ISSN 0021-9045.

Ron Levie, Cagkan Yapar, Gitta Kutyniok, and Giuseppe Caire. Pathloss prediction using deep learning with applications to cellular optimization and efficient d2d link scheduling. In *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 8678–8682, 2020. doi: 10.1109/ICASSP40776.2020.9053347.

Ron Levie, Cagkan Yapar, Gitta Kutyniok, and Giuseppe Caire. RadioUNet: Fast radio map estimation with convolutional neural networks. *IEEE Transactions on Wireless Communications*, 20 (6):4001–4015, 2021.

Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NeurIPS*, pp. 4768–4777, 2017.

Jan Macdonald, Stephan Wäldchen, Sascha Hauch, and Gitta Kutyniok. A rate-distortion framework for explaining neural network decisions. *Preprint arXiv:1905.11092*, 2019.

A. Nguyen, A. Dosovitskiy, J. Yosinski, T. Brox, and J. Clune. Synthesizing the preferred inputs for neurons in neural networks via deep generator networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

Matthew O' Shaughnessy, Gregory Canal, Marissa Connor, Christopher Rozell, and Mark Davenport. Generative causal explanations of black-box classifiers. In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5453–5467. Curran Associates, Inc., 2020.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd International Conference on Knowledge Discovery and Data Mining, ACM SIGKDD*, pp. 1135–1144. Association for Computing Machinery, 2016. ISBN 9781450342322.

Justin K. Romberg, Michael B. Wakin, and Richard G. Baraniuk. Wavelet-domain approximation and compression of piecewise smooth images. *IEEE Trans. Image Processing*, 15:1071–1087, 2006.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70, pp. 3145–3153, 2017.

K. Simonyan, A. Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034, 2014.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*, 2015.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. In *Workshop on Visualization for Deep Learning, ICML*, 2017.

J.T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller. Striving for simplicity: The all convolutional net. In *ICLR (workshop track)*, 2015.

Mallat Stéphane. Chapter 11.3. In Mallat Stéphane (ed.), *A Wavelet Tour of Signal Processing (Third Edition)*, pp. 535–610. Academic Press, Boston, third edition edition, 2009a. ISBN 978-0-12-374370-1.

Mallat Stéphane. Chapter 6. In Mallat Stéphane (ed.), *A Wavelet Tour of Signal Processing (Third Edition)*, pp. 232. Academic Press, Boston, third edition edition, 2009b. ISBN 978-0-12-374370-1.

Mallat Stéphane. In Mallat Stéphane (ed.), *A Wavelet Tour of Signal Processing (Third Edition)*, pp. 232. Academic Press, Boston, third edition edition, 2009c. ISBN 978-0-12-374370-1.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML*, volume 70, pp. 3319–3328, 2017.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016.

Saining Xie, Ross B. Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5987–5995, 2017.

A Appendix

A.1 CartoonX Algorithm

The final CartoonX algorithm is depicted in Algorithm 1.

Algorithm 1: CartoonX

Data: Image $x \in [0,1]^{c \times w \times t}$ with c channels and wt pixels, classifier Φ.
Result: CartoonX explanation $\mathcal{E} \in [0,1]^{w \times t}$ for decision $\Phi(x)$.
Hyperparameters: Sparsity level $\lambda > 0$, number of steps N, number of noise samples L.
Initialize mask $s \coloneqq [1, ..., 1]^T \in [0,1]^k$ on DWT coefficients $h = [h_1, ..., h_k]$ with $x = f(h)$, where f is the discrete inverse wavelet transform;
Compute predicted label $j^* \coloneqq \arg\max_i \Phi_i(x)$;
for $i \leftarrow 1$ **to** N **do**
  Sample L adaptive Gaussian noise samples $v^{(1)}, ..., v^{(L)} \sim \mathcal{N}(\mu, \sigma^2)$;
  Compute obfuscations $y^{(1)}, ..., y^{(L)}$ with $y^{(i)} \coloneqq f(h \odot s + (1-s) \odot v^{(i)})$;
  Clip obfuscations into $[0,1]^{c \times w \times t}$;
  Approximate expected distortion $\hat{D}(x, s, \Phi) \coloneqq \sum_{i=1}^{L} (\Phi_{j^*}(x) - \Phi_{j^*}(y^{(i)}))^2 / L$;
  Compute loss for the mask $\ell(s) \coloneqq \hat{D}(x, s, \Phi) + \lambda \|s\|_1$ and gradient $\nabla_s \ell(s)$;
  Update mask s with gradient descent step and clip s back to $[0,1]^k$;
end
Compute wavelet coefficients $\hat{h}$ for greyscale image $\hat{x}$ of x;
Invert wavelet mask s back to pixel space as $\tilde{\mathcal{E}} \coloneqq f(\hat{h} \odot s)$;
Clip the explanation $\tilde{\mathcal{E}}$ into $[0,1]^{w \times t}$ to obtain $\mathcal{E}$. Visualize $\mathcal{E}$;

A.2 Experiment Details

Throughout our experiments with CartoonX and Pixel RDE, we used a learning rate of $\epsilon = 0.001$, a sample size of $L = 64$ for the adaptive Gaussian noise, and $N = 2000$ steps. Several different sparsity levels were used. We recommend specifying the sparsity level in terms of the number of mask entries k, i.e., choosing the product λk. Pixel RDE typically requires a smaller sparsity level than CartoonX. We chose $\lambda k \in [20, 80]$ for CartoonX and $\lambda k \in [3, 20]$ for Pixel RDE. The obfuscation strategy for Pixel RDE was chosen as Gaussian adaptive noise with mean and standard deviation computed for all pixel values (see Section 4.1.2). In Appendix 8, we show that Gaussian adaptive noise produces much more interpretable explanations than using a zero baseline perturbation. We implemented the DWT for CartoonX with the Pytorch Wavelets package, which is compatible with PyTorch gradient computations, and chose the Daubechies wavelet system with $J = 5$ scales and zero-padding. For the Integrated Gradients method, we used 100 steps, and for the Smoothgrad method, we used 10 samples and a standard deviation of 0.1.

A.3 Sensitivity to Hyperparameters

We compare CartoonX's sensitivity to its main hyperparameters, i.e., the sparsity level λ, the perturbation distribution $\mathcal{V}_s$, and the distortion measure $d(\Phi(x), \Phi(y))$. For each experiment, we fix all but one of the three parameters.

A.3.1 Sensitvity to the Sparsity Level λ

Figure 7 plots CartoonX explanations and Pixel RDEs for increasing λ—the hyperparameter determining the explanation's sparsity in the respective representation system. We find that CartoonX is less sensitive than Pixel RDE to λ. In practice, this means one can find a suitable λ faster for CartoonX than for Pixel RDE.

A.3.2 Sensitvity to the Distribution $\mathcal{V}_s$

Figure 8 plots CartoonX explanations for two choices of $\mathcal{V}_s$: (1) Gaussian adaptive noise (see Section 4.1.2) and (2) constant zero perturbations (i.e. $v = 0$ with probability one under $\mathcal{V}_s$). We observe that the Gaussian adaptive noise gives much more meaningful explanations than the simple zero baseline perturbations.

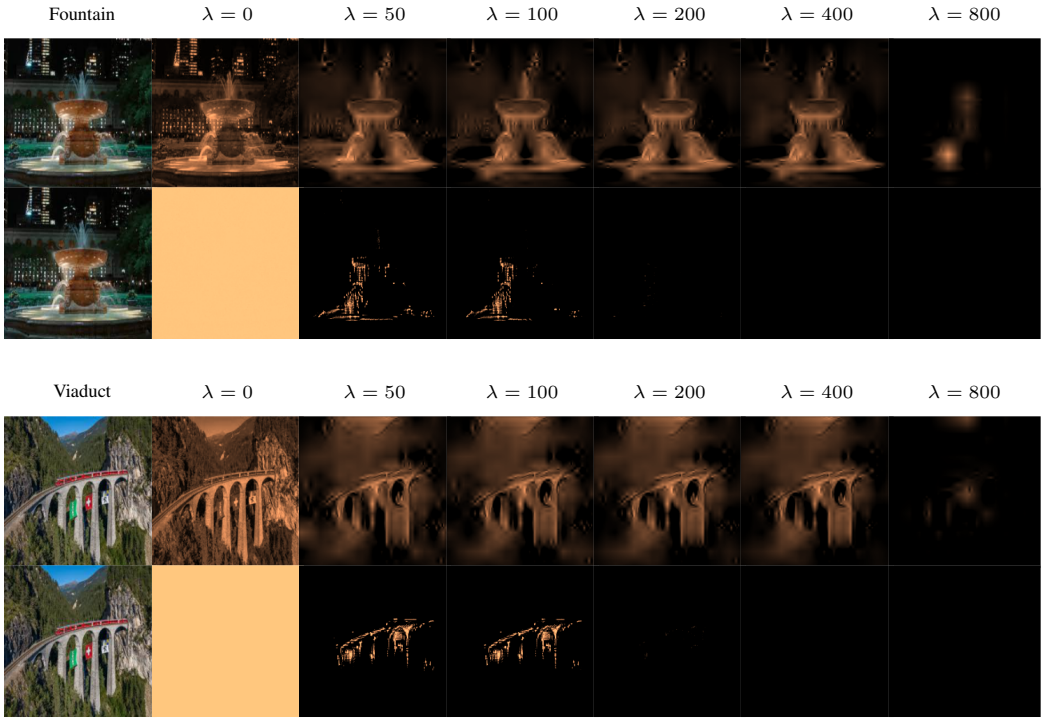


Figure 7: We compare the sensitivity of CartoonX and Pixel RDE to the sparsity level λ. The top row depicts CartoonX, and the bottom row depicts Pixel RDE, for increasing values of λ. Note that for $\lambda = 0$, Pixel RDE is entirely yellow because the mask is initialized as $s = [1 \dots 1]^T$ and $\lambda = 0$ provides no incentive to make s sparser. For the same reason, CatoonX is simply the greyscale image for $\lambda = 0$.

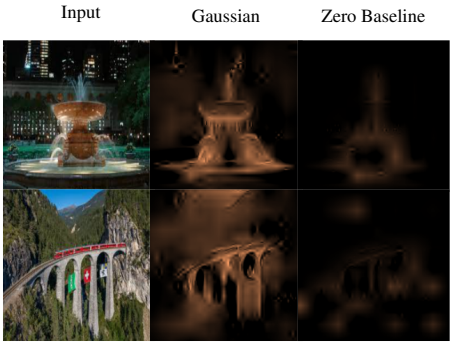


Figure 8: We compare the sensitivity of CartoonX to the perturbation distribution $\mathcal{V}_s$. The top image was classified as a fountain and the bottom image as a viaduct. The second column depicts CartoonX with $\mathcal{V}_s$ as Gaussian adaptive noise, and the third column depicts CartoonX with $\mathcal{V}_s$ as constant zero perturbations (zero baseline). We observe that Gaussian adaptive noise is much more interpretable than the zero baseline.

A.3.3 Sensitvity to the Distortion Measure $d(\Phi(x), \Phi(y))$

Figure 9 plots CartoonX explanations for the following four choices of $d(\Phi(x), \Phi(y))$, where x is the original input, y is the RDE obfuscation, and Φ outputs post-softmax probabilities:

1. $d(\Phi(x), \Phi(y)) = (\Phi_{j^*}(x) - \Phi_{j^*}(y))^2$, where $j^* := \arg\max_j \Phi_j(x)$ (squared ℓ_2 in label)

2. $d(\Phi(x), \Phi(y)) = (\Phi_{j^*}(x) - 1)^2$, where $j^* := \arg\max_j \Phi_j(x)$ (maximize label)
3. $d(\Phi(x), \Phi(y)) = \|\Phi(x) - \Phi(y)\|_2$ (ℓ_2 probabilities)
4. $d(\Phi(x), \Phi(y)) = KL(\Phi(y), \Phi(x))$ (KL-Divergence)

The explanations for $d(\Phi(x), \Phi(y))$ as "squared ℓ_2 in label", "maximize label", and "ℓ_2 probabilities" look indistinguishable. For $d(\Phi(x), \Phi(y))$ as KL-Divergence, we see a slightly less smooth explanation, which may be due to the fact that the KL-Divergence is unbounded unlike the other measures of distortion.

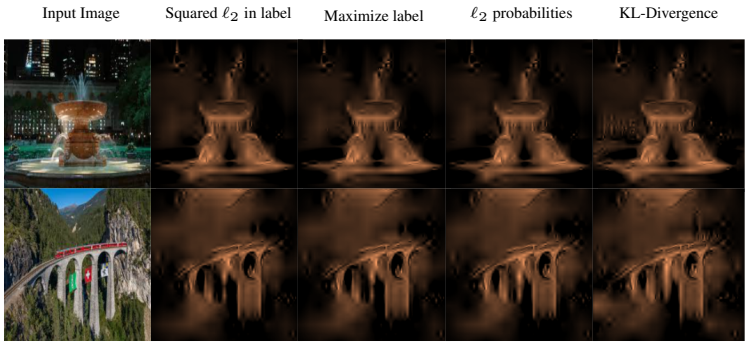


Figure 9: We compare the sensitivity of CartoonX to four measures of distortion $d(\Phi(x), \Phi(y))$. Each of the measures of distortion is marked at the top of each column. We observe almost no difference in the CartoonX explanations for the four distortion measures.

A.4 Reliability and Explanation Artifacts

We argue experimentally why CartoonX is more reliable than Pixel RDE for image data. More precisely, we show that CartoonX is less susceptible to so-called *explanation artifacts* than Pixel RDE. An explanation artifact is an unwanted phenomenon that we observed for Pixel RDE: instead of marking the relevant entries in x, the mask s creates artificial edges that end up making up an artificial class prototype. Explanation artifacts are problematic because they highlight not actual substructures but artificial structures that trigger the classification. Examples for explanation artifacts in Pixel RDE are given in Figure 11.

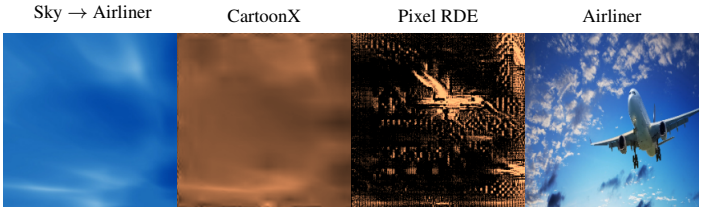


Figure 10: CartoonX and Pixel RDE are both performed on the image of the blue sky. However, both methods are adjusted here to find evidence for the output probabilities of the image of the airplane instead of the blue sky. Pixel RDE, unlike CartoonX, can create an artificial airplane as evidence for an airplane in the smooth blue sky.

Pixel RDE can produce artificial edges in smooth regions for the following reason: When s has a curve-like structure in some region, unselected points near s are replaced with perturbations that tend to differ from the values of the curve-like structure in s. Thus, the curve-like structure also appears in the obfuscation and can produce low distortion if the structure makes up a prototypical class feature (see, for example, the airplane in Figure 10).

We suspect CartoonX is inherently less susceptible to explanation artifacts for the following reason: Natural images tend to be piece-wise smooth, and piece-wise smooth images have sparse

high-frequency DWT coefficients that cluster about the edges Stéphane (2009b) (see for example Figure 4). For a DWT mask to create artificial edges, it has to select a curve-like structure in the high-frequency coefficients (low-frequency coefficients cannot create edges) and replace surrounding unselected values with different values. However, in CartoonX, perturbations of high-frequency coefficients are Gaussian with low variance centered close to zero (see adaptive Gaussian noise in Section 4.1.2), which are not very different from the values along the selected curve due to the sparsity of the coefficients.

We illustrate our previous reasoning about explanation artifacts in a controlled example (see Figure 10). We take an image $x^{(\text{sky})}$ of a blue sky that is very smooth and an image $x^{(\text{plane})}$ of a airplane. The goal is to show that Pixel RDE, unlike CartoonX, can create artificial evidence for the class airplane on the image of the smooth blue sky. We perform CartoonX and Pixel RDE on the blue sky image with the distortion function

$$\forall y \in \mathbb{R}^n : \; d(\Phi(x^{(\text{sky})}), \Phi(y)) = 10^6 \|\Phi(x^{(\text{plane})}) - \Phi(y)\|_2,$$

and a sparsity level of $\lambda = 80000$. As expected, we observe that Pixel RDE, unlike CartoonX, can create an artificial plane in the smooth blue sky(see Figure 10).

A.5 Explaining Through ImageNet History: From AlexNet to ResNetXT50

In the deep learning community, it is well-known that AlexNet (Krizhevsky et al., 2012) provided a major breakthrough in deep learning, improving the top-5 error on ImageNet from 25% to 16%. Since then, deep learning based ImageNet classifiers have continued to drastically improve on ImageNet—achieving less than 6% top-5 error in 2016. In Figure 12, we compare CartoonX for four ImageNet classifiers with increasing performance, starting with AlexNet (top-1 accuracy 56.55%, AlexNet), VGG16 (top-1 accuracy 71.59%, Simonyan & Zisserman (2015)), InceptionV3 (top-1 accuracy 77.29%, Szegedy et al. (2016)), and ResNeXt50 (top-1 accuracy 77.62%, Xie et al. (2017).) Throughout the experiment, the CartoonX hyperparameters for a given image are not changed for any of the four classifiers.

A.6 Explaining Misclassifications with CartoonX

In Figure 13, 14, 15, and 16, we provide further examples where CartoonX provides insightful explanations for misclassified images.

A.7 CartoonX Compared on Random ImageNet Samples

Figure 17, 18, 19, and 20 compares CartoonX to Pixel RDE (Macdonald et al., 2019), Integrated Gradients (Sundararajan et al., 2017), Smoothgrad (Smilkov et al., 2017), Guided Backprop (Springenberg et al., 2015), and (Bach et al., 2015) on random Imagenet samples classified by VGG16.

A.8 CartoonX Failures

We also show failures of CartoonX in Figure 21. These are examples of explanations that are not interpretable and seem to fail at explaining the model prediction. Notably, most failure examples are also not particularly well explained by other state-of-the-art methods. It is challenging to state the underlying reason for the CatoonX failures with certainty (there is always the possibility that the neural net bases its decision on non-interpretable grounds). We intentionally also showed uninterpretable CartoonX explanations that were not too sparse (all or almost black explanations) since one can typically fix these explanations by decreasing λ.

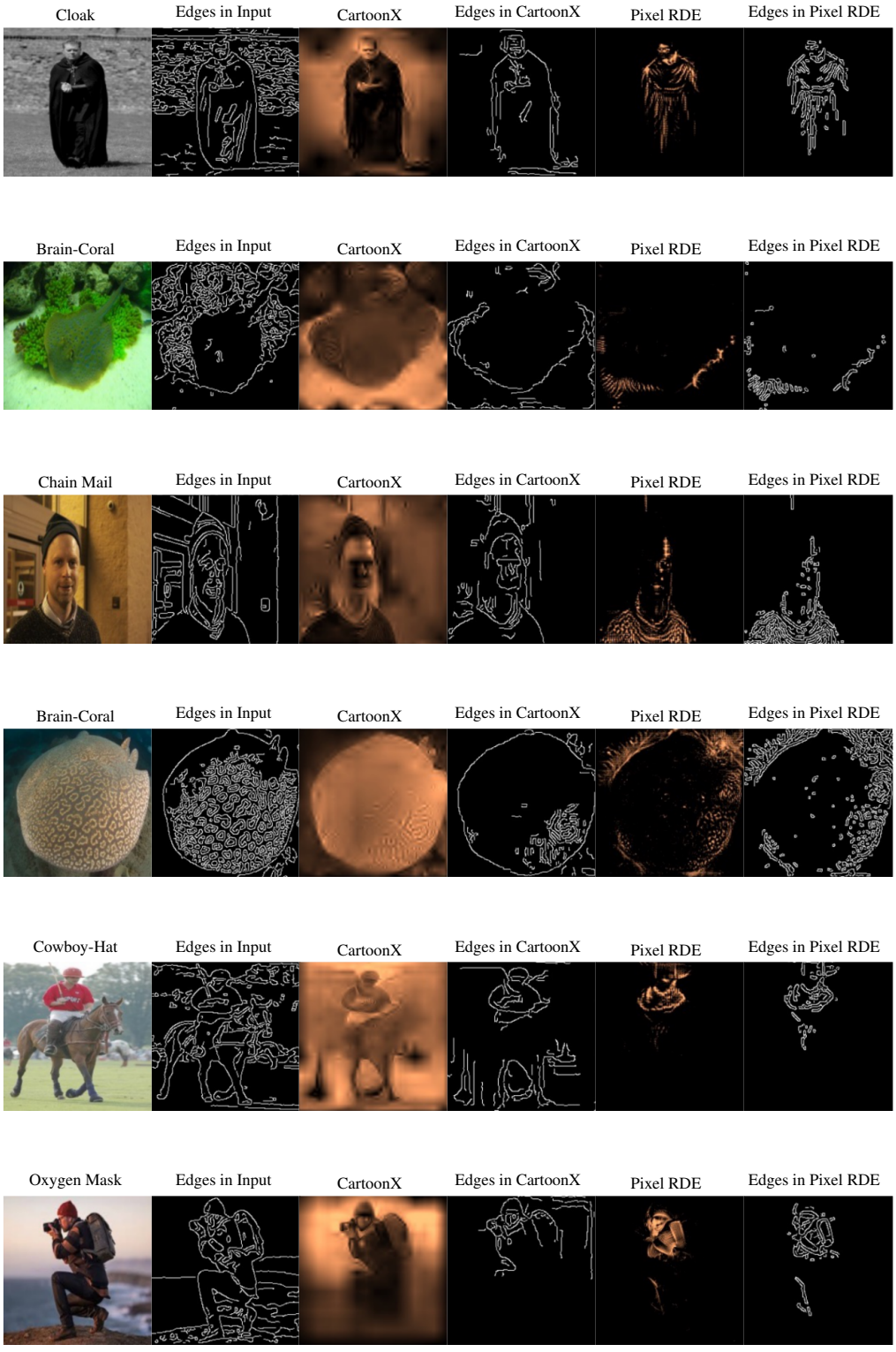


Figure 11: Explanation artifacts in Pixel RDE. We observe that Pixel RDE tends to create edges that are not a subset of the edges in the original input image. These edges can make prototypical artifact patterns such as wrinkles in the cloak (first row), coral tentacles (second row), or chain mail (third row).

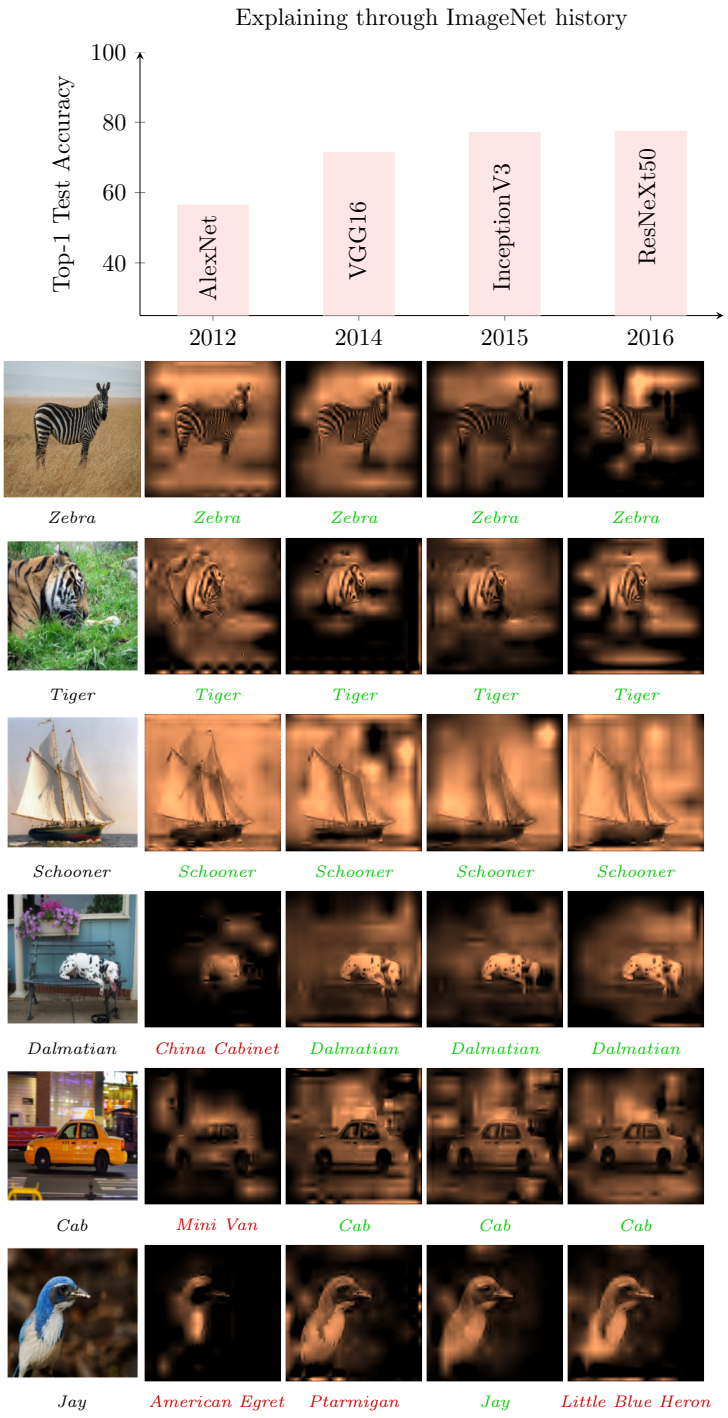


Figure 12: We compare CatoonX explanations for classifications by AlexNet (Krizhevsky et al., 2012), VGG16 (Simonyan & Zisserman, 2015), InceptionV3 (Szegedy et al., 2016), and ResNeXt50 (Xie et al., 2017). Green labels mark correct classifications and red labels mark wrong classifactions.

Figure 13: Explaining misclassifications with CartoonX on Imagenet and MobileNetV3-Small.

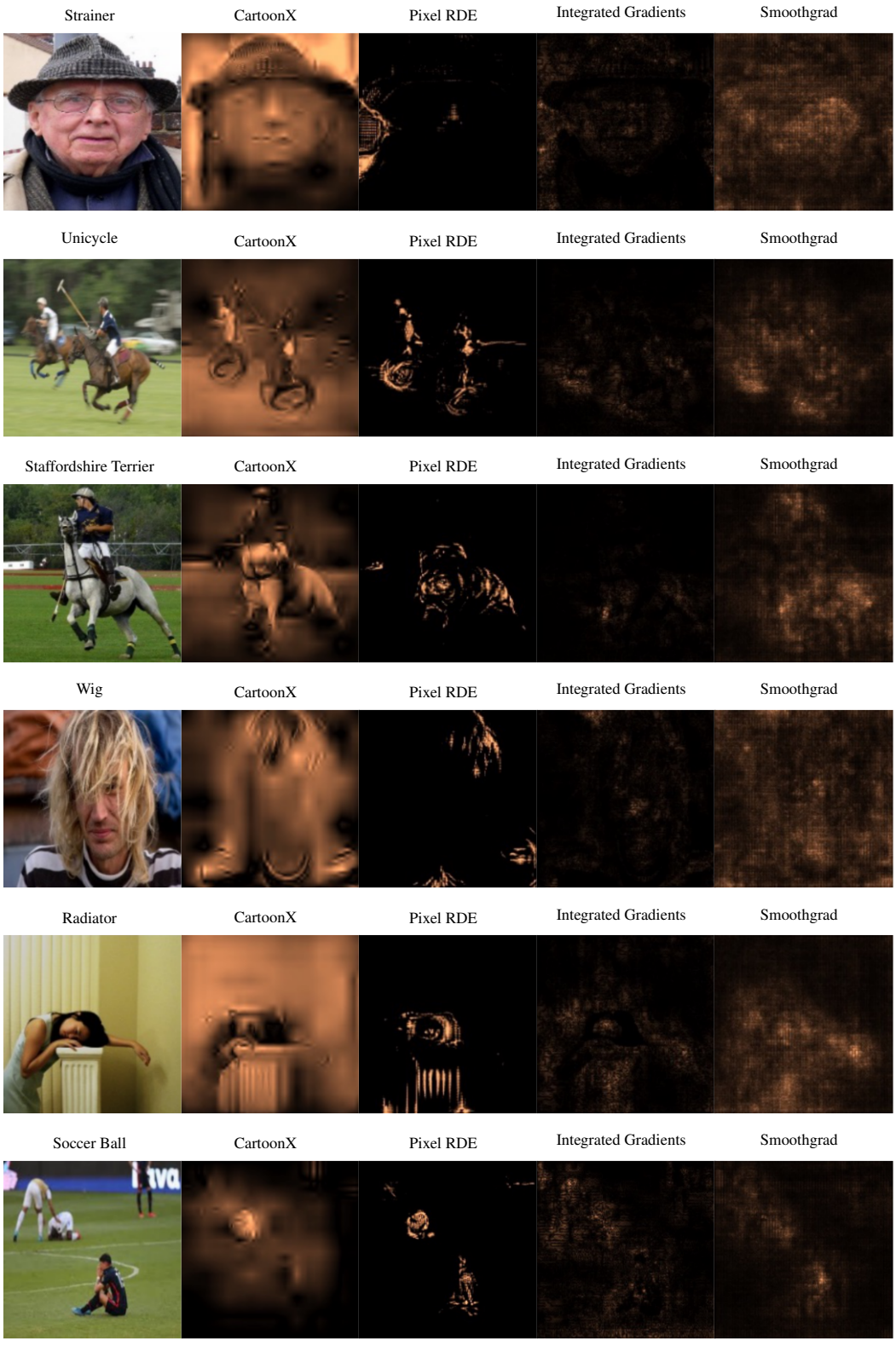


Figure 14: Explaining misclassifications with CartoonX on Imagenet and MobileNetV3-Small.

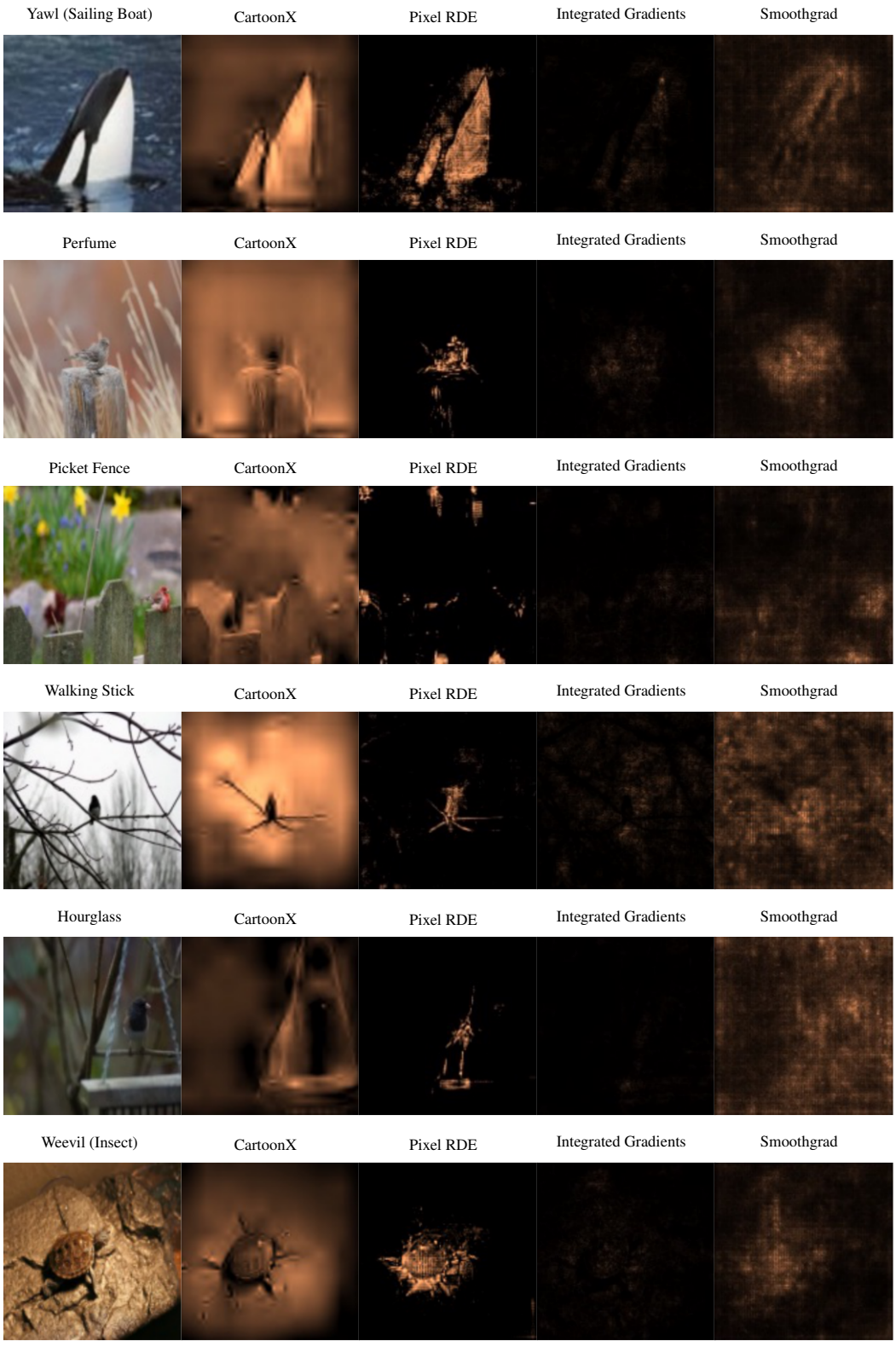


Figure 15: Explaining misclassifications with CartoonX on Imagenet and MobileNetV3-Small.

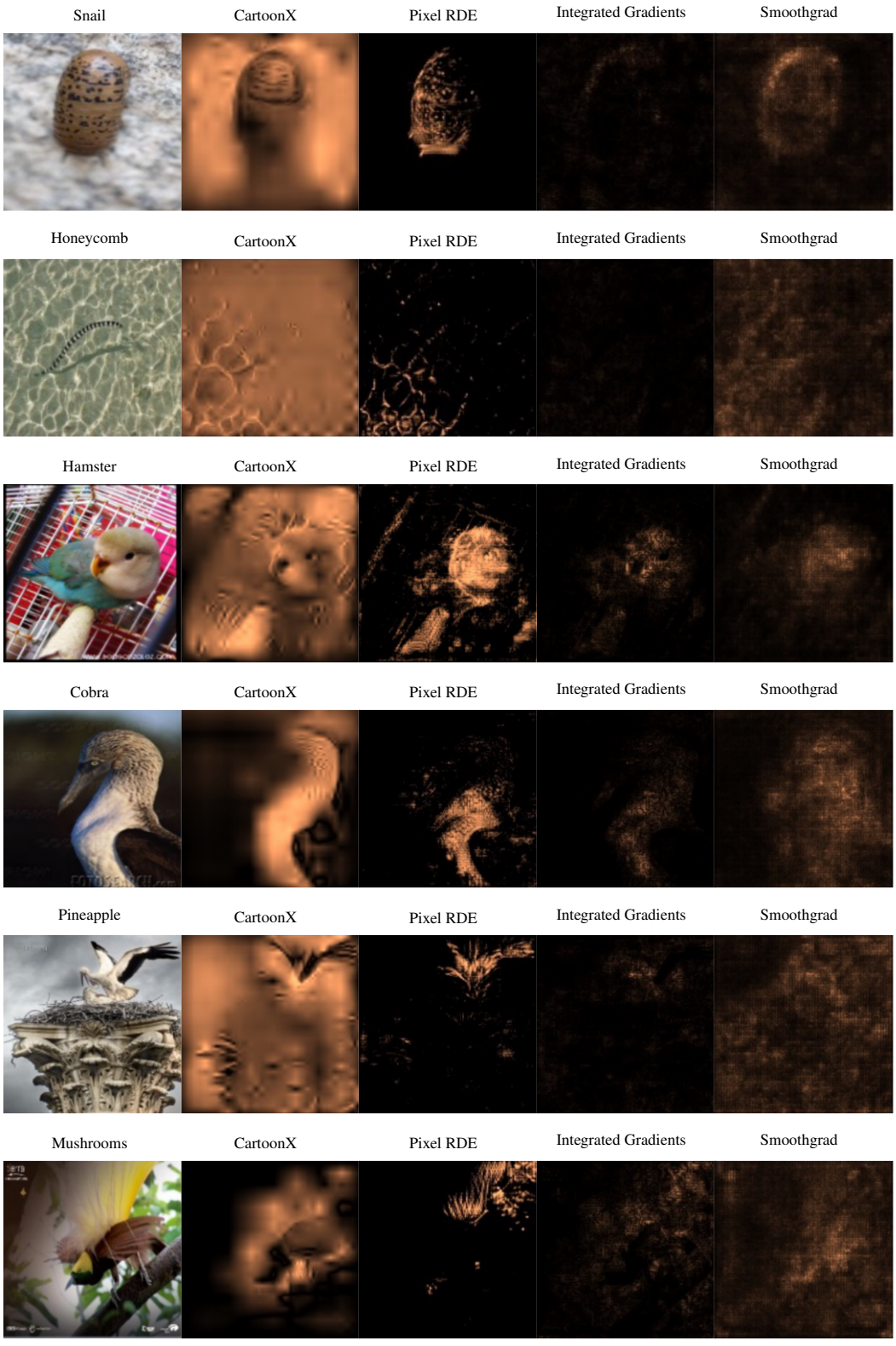


Figure 16: Explaining misclassifications with CartoonX on Imagenet and MobileNetV3-Small.

Figure 17: Comparing CartoonX on random ImageNet samples and VGG16.

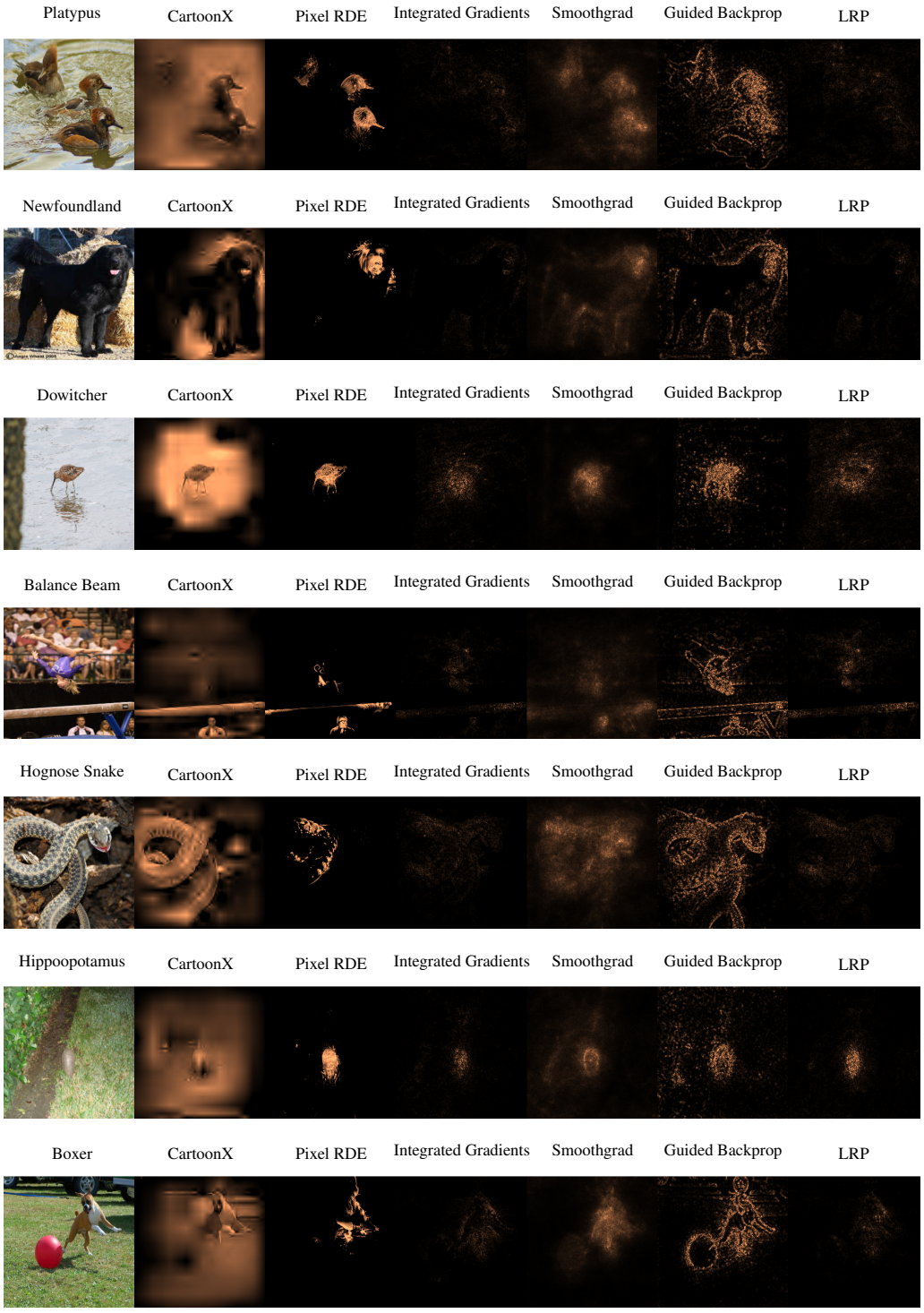


Figure 18: Comparing CartoonX on random ImageNet samples and VGG16.

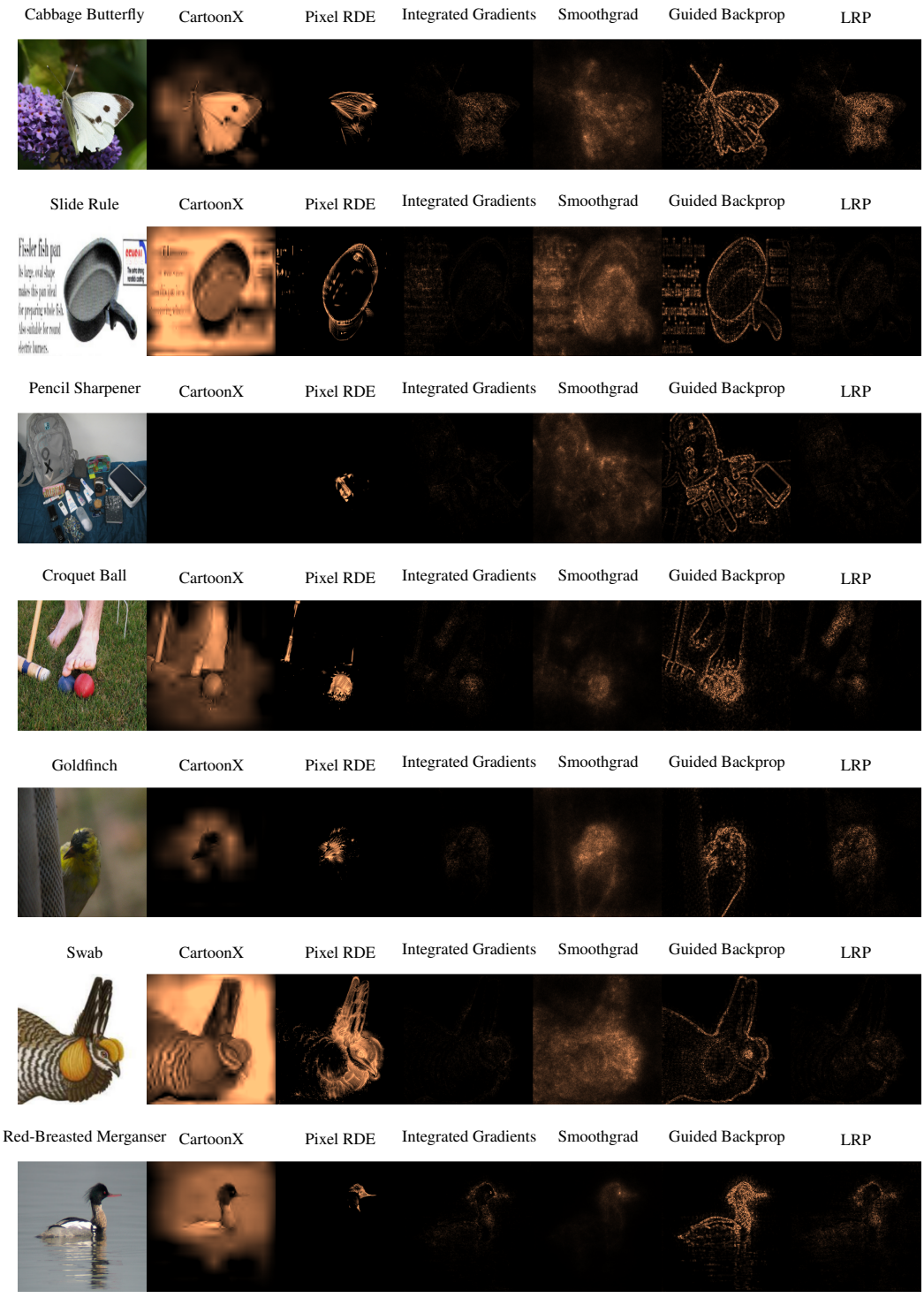


Figure 19: Comparing CartoonX on random ImageNet samples and VGG16.

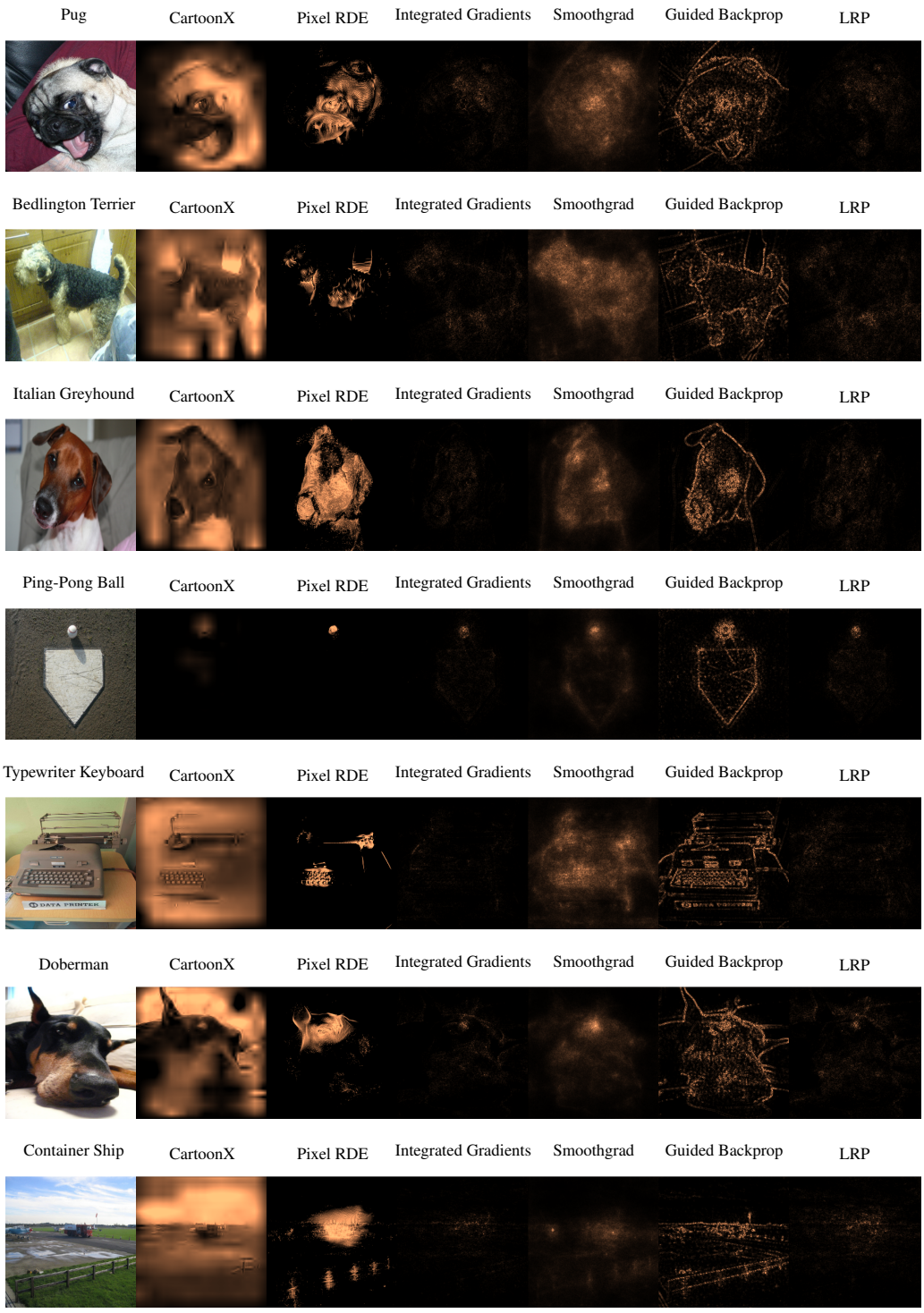


Figure 20: Comparing CartoonX on random ImageNet samples and VGG16.

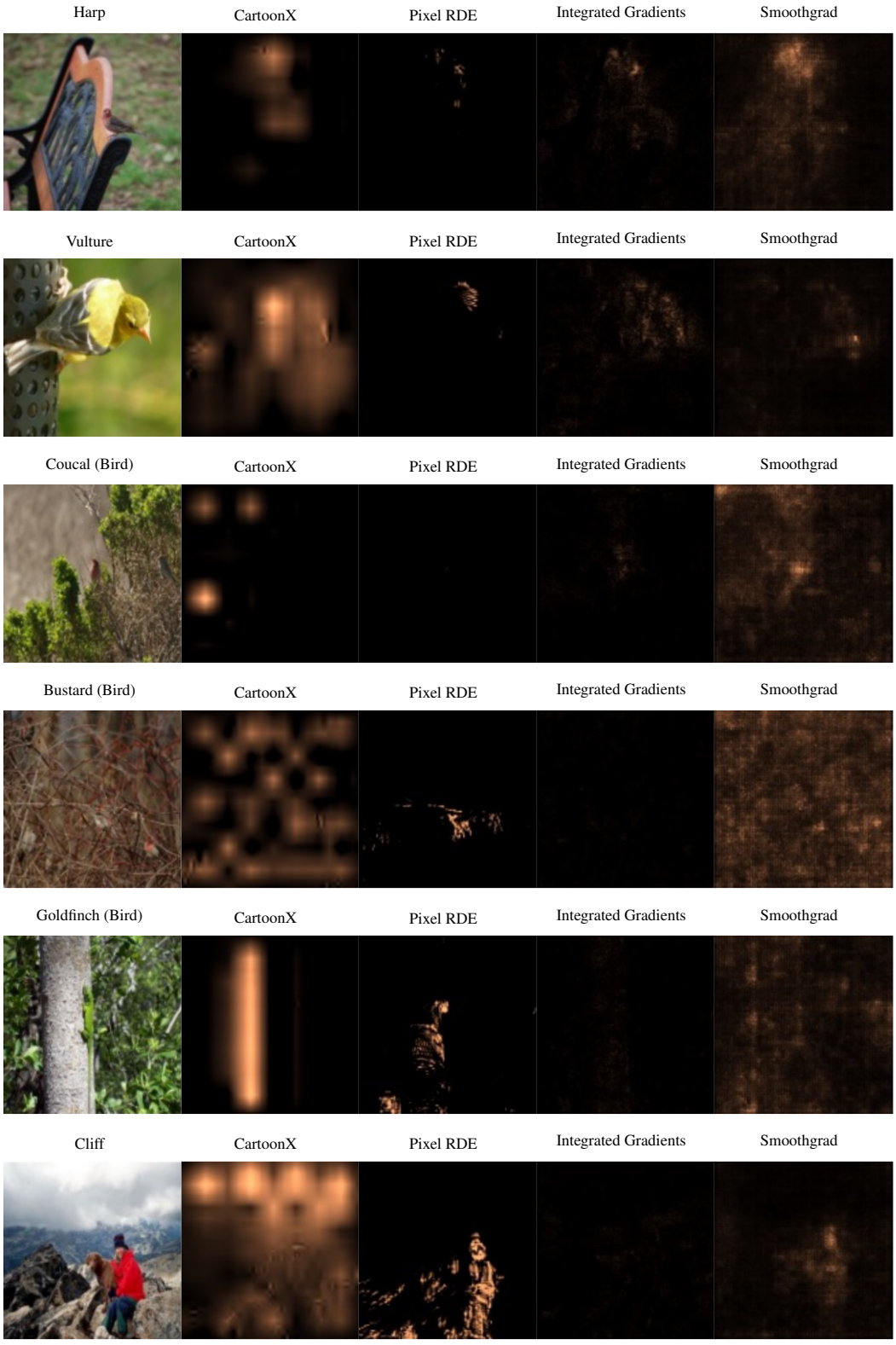


Figure 21: Failures of CartoonX.