INTERPRETABILITY THROUGH INVERTIBILITY: A DEEP CONVOLUTIONAL NETWORK WITH IDEAL COUNTERFACTUALS AND ISOSURFACES

Anonymous authors
Paper under double-blind review

ABSTRACT

Current state of the art computer vision applications rely on highly complex models. Their interpretability is mostly limited to post-hoc methods which are not guaranteed to be faithful to the model. To elucidate a model's decision, we present a novel interpretable model based on an invertible deep convolutional network. Our model generates meaningful, faithful, and ideal counterfactuals. Using PCA on the classifier's input, we can also create "isofactuals"– image interpolations with the same outcome but visually meaningful different features. Counter- and isofactuals can be used to identify positive and negative evidence in an image. This can also be visualized with heatmaps. We evaluate our approach against gradient-based attribution methods, which we find to produce meaningless adversarial perturbations. Using our method, we reveal biases in three different datasets. In a human subject experiment, we test whether non-experts find our method useful to spot spurious correlations learned by a model. Our work is a step towards more trustworthy explanations for computer vision. For code and datasets see [1].

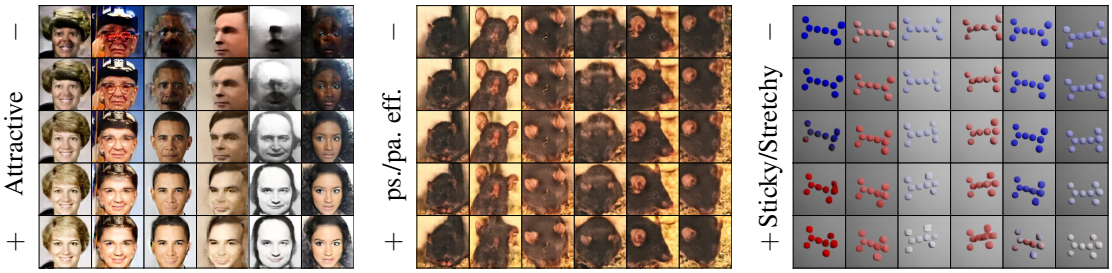
1 INTRODUCTION

Neural networks are deployed in high-stakes domains such as medical image diagnosis and autonomous driving. However, they are considered black boxes lacking transparency and interpretability. Logistic regression, although less powerful, is broadly used in production because of its algorithmic simplicity and inherent interpretability: the weight coefficients directly reflect feature relevance. We can generate counterfactual examples by moving along the weight vector $\boldsymbol{x} + \alpha \boldsymbol{w}$.

To improve the interpretability of deep convolutional networks, numerous saliency methods have been proposed that assign a relevance score to each input variable. However, there is no guarantee that these explanations faithfully reflect the network's output. Some methods have been found to even ignore the network's parameters partially; others can be manipulated easily (Dombrowski et al., 2019; Anders et al., 2020; Sixt et al., 2020; Adebayo et al., 2018; Anders et al., 2020). Instead of attributing a classification result to individual pixels, Kim et al. (2018) introduced user-defined concepts such as "striped" that e.g. the classfication "zebra" can be traced back to. Post-hoc explanations have the advantage that they work with almost any model. As an alternative interpretable components can be integrated in the network's architecture. An example is ProtoPNet (Chen et al., 2019) which learns class-specific prototypical representations of image patches. A prediction can then be explained by the similarity to the matching prototypes. However, no counterfactuals can be visualized in image space. To generate counterfactuals, the GAN framework has been adapted by (Liu et al., 2019; Baumgartner et al., 2018).

In our paper, we propose an interpretable deep model that generates ideal counterfactuals, discovers concepts and provides faithful saliency maps. The idea is to use an invertible neural network $\varphi(\boldsymbol{x})$ to learn a powerful representation and predict the logits by $y = \boldsymbol{w}^T \varphi(\boldsymbol{x})$. We can move along the weight vector as before and visualize the changes as images using the inverse: $\varphi^{-1}(\boldsymbol{z} + \alpha \boldsymbol{w})$ where $\boldsymbol{z} = \varphi(\boldsymbol{x})$. We construct a basis of the isosurface using PCA.The principal components are

[1] We plan to publish our code `https://anonymous.4open.science/r/ae263acc-aad1-42f8-a639-aec20ff31fc3/`



(a) CELEBA (b) BLACK MICE (c) TWO4TWO

Figure 1: Counterfactuals: Using $\tilde{\boldsymbol{x}} = \varphi^{-1}(\boldsymbol{z} + \alpha \boldsymbol{w})$, we can generate images that look like the original $\boldsymbol{x}$ but are classified differently. **(a) (b)** Mice classified as showing post-surgical/-anesthetic effects: Lower head, Ear position, more yellow bedding material. **(c)** Sticky (bottom, right arms moved inwards) and Strechy (top, right arms moved outwards): color and rotation also change.

orthogonal to $\boldsymbol{w}$ and correspond to meaningful concepts (gender, skin color). As they represent the entire data distribution they can explain the model globally.

We show the correctness and usefulness of our method on a broad range of evaluations. We analyzed the differences of our approach to gradient-based saliency methods and found that in contrast to our method, gradient-based counterfactuals appear meaningless. We verified our method on a synthetic dataset, CELEBA, and a real-world image dataset of laboratory mice. Our method revealed biases in all datasets. Finally, we ran a human subject experiment to evaluate the usefulness to lay users.

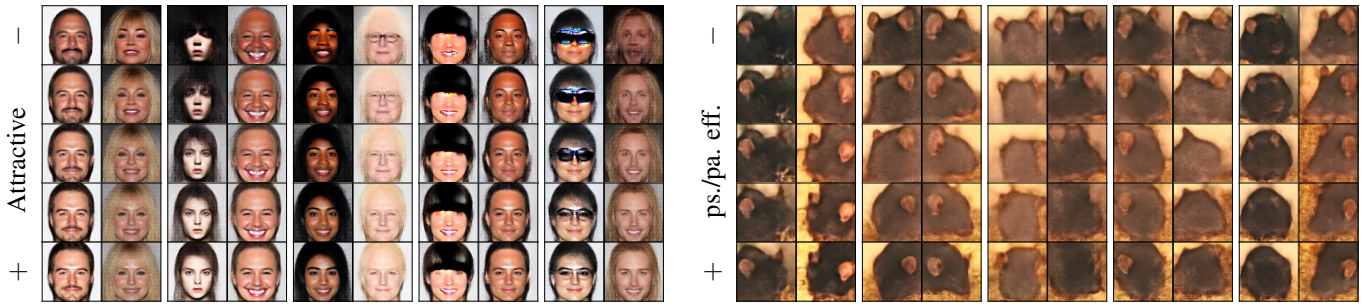
2 METHOD

Wachter et al. (2018) defines a counterfactual example $\tilde{\boldsymbol{x}}$ of a data point $\boldsymbol{x}$ to differ in some properties $\boldsymbol{y}$ while keeping unrelated properties $\boldsymbol{\xi}$ fixed. They call a counterfactual ideal if it exclusively alters the properties of interest $\boldsymbol{y}$ and if it is the closest possible counterfactual to $\boldsymbol{x}$. In the following paragraphs, we show how ideal counterfactual examples can be obtained for an invertible neural network $\varphi\colon \mathbb{R}^n \mapsto \mathbb{R}^n$. The forward function φ maps a data point $\boldsymbol{x}$ to a feature vector $\boldsymbol{z} = \varphi(\boldsymbol{x})$ and the corresponding inverse function yields $\boldsymbol{x} = \varphi^{-1}(\boldsymbol{z})$. The features $\boldsymbol{z}$ are used by a binary classifier $f(\boldsymbol{x}) = \boldsymbol{w}^T\boldsymbol{z} + b$ to predicts the label $\boldsymbol{y}$, which corresponds to the explained properties.

Counterfactuals To create counterfactual examples, we can change the logit score of the classifier by adding the weight vector to the representation and then invert it back: $\tilde{\boldsymbol{x}} = \varphi^{-1}(\boldsymbol{z} + \alpha \boldsymbol{w})$. For any values of $\boldsymbol{y}$, we can generate counterfactual samples by picking α accordingly. The change in logit score is $\alpha \boldsymbol{w}^T\boldsymbol{w}$. In figure 1, we visualize counterfactual interpolations on different datasets. In the $\boldsymbol{z}$-space, $\tilde{\boldsymbol{z}} = \boldsymbol{z} + \alpha \boldsymbol{w}$ is the closest sample under Euclidean norm.

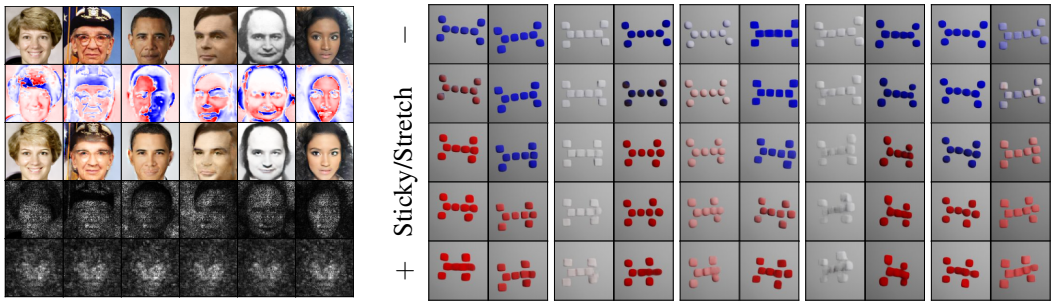
Isosurface The isosurfaces of the classifier ($\boldsymbol{w} \in \mathbb{R}^m$) are $m-1$ dimensional hyperplanes corresponding to the independent factors $\boldsymbol{\xi}$. Moving along the isosurface will not change the classifier output. To construct an orthogonal basis of the isosurface, we first project each $\boldsymbol{z} = \varphi(\boldsymbol{x})$ to become orthogonal to $\boldsymbol{w}$: $\boldsymbol{z}_\perp = \boldsymbol{z} - (\boldsymbol{w}^T\boldsymbol{z}/\boldsymbol{w}^T\boldsymbol{w})\boldsymbol{w}$. Then, a PCA is computed on the $\boldsymbol{z}_\perp$'s and the resulting principal components $\boldsymbol{e}_1 \dots \boldsymbol{e}_{m-1}$ form the basis. The last principal component $\boldsymbol{e}_m$ (smallest explained variance) is oriented towards $\boldsymbol{w}$ due to using $\boldsymbol{z}_\perp$ and discarded. PCA does not require any labels and nonetheless finds interesting directions in z-space. Depending on the use-case, supervised labels could determine a subset of the basis vectors. The principle components represent the direction of highest variance and can provide a global understanding of the model. If the changes by counterfactuals show a pattern on the principal components, it must also happen to the data. In figure 2 and 3b, we visualize the changes $\boldsymbol{w}$ has on the principal component: $\varphi(\boldsymbol{e}_i + \alpha \boldsymbol{w})$.

Saliency maps Saliency maps help draw attention to the differences between $\boldsymbol{x}$ and the counterfactuals $\tilde{\boldsymbol{x}}$. We can measure the difference either in input space or in an intermediate feature map $\boldsymbol{h}$. The saliency map of an intermediate layer can be resized to fit the resolution of the input. To determine which visual features are strengthened and which are weakened, we calculate the similarity measure $\boldsymbol{m}_{(i,j)} = \Delta\boldsymbol{h}_{ij}^T\boldsymbol{h}_{ij}/|\boldsymbol{h}_{ij}|$ per feature map location i, j. The saliency map $\boldsymbol{m}$ is positive if the change is aligned with the direction of the feature vector ($\measuredangle(\Delta\boldsymbol{h}_{ij}, \boldsymbol{h}_{ij}) > 0$) and therefore strengthened it,



(a) CELEBA "Attractive" (b) BLACK MICE

Figure 2: Isofactuals for the CELEBA and BLACK MICE datasets. The panels in (a) and (b) show generated images grouped by principal components (image pairs per column). All images in a given row have identical logit scores. By adding or subtracting the weight vector associated to the label "attractive" we can shift the isosurface, corresponding to the rows. Images were generated by inverting a latent coordinate given by the principal components.



(a) CELEBA "Attractive" (b) TWO4TWO

Figure 3: **(a)** Saliency maps for an increase of "Attractive". Row 1: original images, 2: our saliency maps, 3: Counterfactuals, 4: Int. gradient (with CFs as $\boldsymbol{x}_0$). 5: SmoothGrad. *red*: change is aligned with the current feature, *blue*: change goes into a different direction than the current feature), **(b)** Isosurface of the TWO4TWO dataset.

or negative if the feature was weakened. In figure 3a, we present saliency maps for the CELEBA "Attractive" label.

Model Our invertible network follows the Glow architecture (Kingma & Dhariwal, 2018). The network is trained to map the data distribution to a standard normal distribution. We reduce the input dimensionality of (3, 128, 128) down to (786) by fading half of the channels out per downsampling step. For counterfactual generation, we can simple fix the out-faded values as the correspond to small details. We have 7 downsampling steps and 351 layers in total. An important design decision is that the final layer's output is not input to the linear classifier because the PCA would fail to discover meaningful directions. The classifier uses the output of layer 321. The layers 322-351 are optimized using the standard unsupervised flow objective. For the first 321 layers, we also train on the supervised loss from the classifier, for details see Appendix A.1.

3 THE GRADIENT OF x AND THE DIRECTIONAL DERIVATIVE $\mathrm{d}x/\mathrm{d}w$

Conceptually, saliency maps and counterfactuals are closely related. A method generating counterfactuals should access relevant features, and saliency methods should indicate those features that affect the logit scores when changed. The gradient is the basis for an array of saliency methods. For example, SmoothGrad, LRP$_\epsilon$, LRP$_{\alpha\beta}$, γ-rule and integrated gradients (Smilkov et al., 2017; Bach et al., 2015; Montavon et al., 2019; Sundararajan et al., 2017). The gradient can be very noisy (Balduzzi et al., 2017), and previous works have proposed smoothing it either integrating along a line (int.grad.) or by discarding information of higher layers (LRP$_{\alpha\beta}$, see Sixt et al. (2020)). A fundamental question is unanswered: does the gradient provide the correct basis for feature importance?

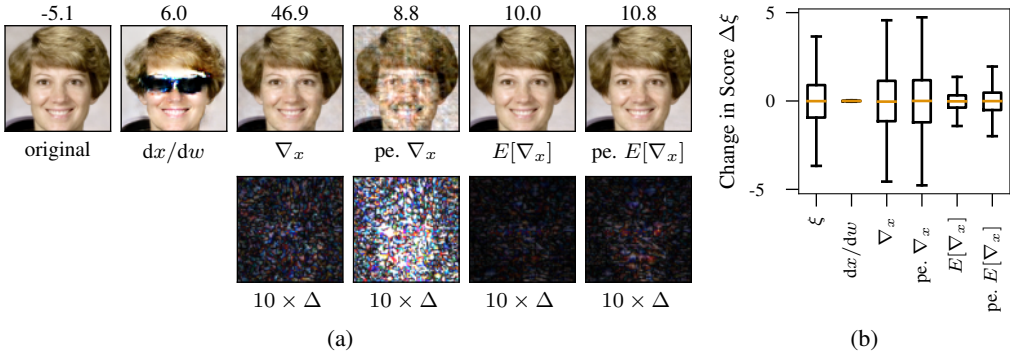


Figure 4: **(a)** Integration of the *original* image along the *directional derivative* $\mathrm{d}\boldsymbol{x}/\mathrm{d}\boldsymbol{w}$ *gradient* $\nabla_{\boldsymbol{x}} f$, *gradient+penalize* (pe. $\nabla_{\boldsymbol{x}}$), *SmoothGrad* ($E[\nabla_{\boldsymbol{x}}]$) *SmoothGrad+penalize* (pe. $E[\nabla_{\boldsymbol{x}}]$) for the class *sunglasses*. Integrating the gradient shows barely visual changes. The differences are multipled by $\times 10$ and shown below. **(b)** Changes to the unrelated factors $\boldsymbol{\xi}$ (isosurface). All gradient-base methods result in strong and there not ideal counterfactuals. Integrating the directional derivative $\nabla_{\boldsymbol{w}} \boldsymbol{x}$ results in a meaningful change (sunglasses appear) while the $\boldsymbol{\xi}$ changes little.

The gradient $\nabla_{\boldsymbol{x}} f(\boldsymbol{x})$ yields a pixel-level answer: what changes in the input results in the *steepest* change of the classifier score. By integrating those changes, we can obtain a gradient-based counterfactual. We compare these to counterfactuals obtained using the directional derivative $\nabla_{\boldsymbol{w}} \boldsymbol{x} = \mathrm{d}\varphi^{-1}(\boldsymbol{z})/\mathrm{d}\boldsymbol{w}$ which answers to what changes in $\boldsymbol{x}$, when $\boldsymbol{z}$ moves into the direction of $\boldsymbol{w}$? Their similarity is easy to seen when using the Jacobian: $\nabla_{\boldsymbol{x}} f = J_{\varphi}(\boldsymbol{x})\boldsymbol{w}$ and $\nabla_{\boldsymbol{w}} \boldsymbol{x} = J_{\varphi^{-1}}(\boldsymbol{z})\boldsymbol{w}$.

We integrated both methods (see Appendix A.2 for technical details) and show exemplary counterfactuals integrated along the "Sunglass" dimension in figure 4a. While the gradient-based counterfactual increased the logit score more than the one based on the directional derivative, the resulting image is hardly different to the original. Only noisy high-frequency patterns appear – similar to adversarial examples. We integrated the feature map of layer 40, as the gradient of the input gave in even worse samples. Integrating the directional derivative adds sunglasses to the astronaut – a meaningful counterfactual.

How much do the counterfactuals affect the independent factors $\boldsymbol{\xi}$? Using 100 random samples, we calculated the changes in principal components $\boldsymbol{\xi}$, see figure 4b. Gradient-based methods affect the independent factors strongly, while the directional derivative scores lowest. Thus, gradient-based counterfactuals do not only explain the increase of the logit score but all the other changes too, rendering them uninformative. In contrast, counterfactual based on the directional derivative keeps the independent factors almost constant. A numerical imprecision in the integration are the cause as: $\int_0^{\alpha} \mathrm{d}\boldsymbol{x}(t)/\mathrm{d}\boldsymbol{w}\mathrm{d}t = \varphi^{-1}(\varphi(\boldsymbol{x}) + \alpha\boldsymbol{w}) - \boldsymbol{x}$.

A simple fix could be to also penalize changes to the independent factors: $\nabla_{\boldsymbol{x}}[f(\boldsymbol{x}) + (\boldsymbol{\xi}(\boldsymbol{x}) - \boldsymbol{\xi}(\boldsymbol{x}_0))^2]$. Here, the problem is $\boldsymbol{x} = \boldsymbol{x}_0$ at first and a derivation is not penalized. We found it only amplifies the meaningless patterns and still does not move exclusively into the direction of $\boldsymbol{w}$. We also use SmoothGrad to dampen the gradient's noise (relative noise level=0.25, 1% of highest/lowest values clipped). SmoothGrad results in a smaller change of the image, smaller logit value than the gradient, and smaller changes to $\boldsymbol{\xi}$, still considerably high than for the directional derivative.

Furthermore, we measure the local self-correlations of the gradient and the directional derivative per 8×8 tiles (each pixel values with each other), again on 100: The raw pixels are correlated by 0.89, the gradient by 0.15, SmoothGrad by 0.21, and the directional derivative by 0.71. Thus, the changes by $\mathrm{d}\boldsymbol{x}/\mathrm{d}\boldsymbol{w}$ seem to be more structured than the gradient.

The gradient itself and also smooth variants cannot provide meaningful counterfactuals. The evidence presented suggest that the gradient and smooth version contain less structural information as the directional derivative $\mathrm{d}\boldsymbol{x}/\mathrm{d}\boldsymbol{w}$. To compute the $\mathrm{d}\boldsymbol{x}/\mathrm{d}\boldsymbol{w}$, a backward model φ^{-1} is required. The alternative to construct the Jacobian matrix $J_{\varphi}(\boldsymbol{x})$ and then compute $\mathrm{d}\boldsymbol{x}/\mathrm{d}\boldsymbol{w} = [J_{\varphi}(\boldsymbol{x})]^{-1}\boldsymbol{w}$ is not feasible for deep neural networks, see (Jacobsen et al., 2018) for a detailed discussion. This finding motivates the application of invertible neural network for interpretability further.

4 EVALUATION

Interpretability research is notoriously hard to evaluate. How can it be shown that an explanation technique works correctly if we do not understand the model it explains? Singla et al. (2020) creates counterfactuals for a black box classifier using a GAN and then measure how well their counterfactuals affect the classifier. The invertibility of our network and the way we generate the counterfactuals renders such an analysis irrelevant. Instead, our evaluation focuses on the questions: Can our method be used to find biases reliably? Does the method provide helpful insights to an end user? We answer these questions using three datasets:

Two4Two The TWO4TWO dataset (Anonymous, 2020) is a set of computer-generated images intended to evaluate interpretable ML – to test both humans and algorithms. It contains two abstract "animals", Sticky and Stretchy. For Sticky, the right arms are moved inwards and for Stretchy outwards, see figure 3b. Color, building blocks (cubes or spheres), bending, rotation, and background can be changed arbitrarily to introduce additional biases. We sampled the color such that blue is slightly predictive for Stretchy and red for Sticky. As the arms overlap sometimes, it is beneficial to use the color. We sample all other parameters independently of the animal class.

Inspecting the counterfactuals, we can confirm that the model uses the color bias. Even though we sampled rotation randomly, the counterfactuals and isosurfaces indicate that the model assigns rotated objects a higher score for Sticky. To test each parameter's influence, we generated 1000 new objects and re-sampled each parameter 8 times separately. For the arm's position, we found a correlation with the model's probability of -0.798. The color correlates with 0.352 and the absolute value of the rotation parameter with 0.136 $(0.11, 0.16)_{95\%}$. At high rotations, the model is more certain that it is a Sticky. The other parameters do not correlate, see table 2.

To investigate why the model's logit score changes with object rotation – a bias not intended when generated the dataset, we visually inspected how rotations affect other geometrical properties. We found that under high rotations, the arms of the object appear to move inwards. We calculated the relative position (λ) of the arms and found a correlation between λ and the object rotation: $r = -0.180(-0.2, -0.17)_{95\%}$. The arm position is the strongest feature, and our results indicate that the model was not able to "understand" 3D space.

CelebA CELEBA (Liu et al., 2015) is a popular face dataset, and neural networks have successfully been employed in supervised and generative tasks. For interpretable ML, it is still a challenging dataset: how can an abstract concept as attractiveness be pinned to pixels? In figure 1, we show the counterfactuals for different images on the "Attractive" class. The background, camera angle, skin color, age and eyes change, glasses disappear and faces look more feminine. The changes are well highlighted by our saliency maps (fig. 3a). For example, the model thinks Grace Hopper would be more attractive without glasses. In figure 2a, we visualize how isofactuals change when adding different amounts of the weight vector associated to the output class "Attractive". We find that meaningful visual features co-vary with one or more principal components (pc): gender (pcs 1 and 3), smiling (pcs 1, 2 ,3 and 4), background (pcs 1, 2 and 3), skin color (pcs 2, 3 and 4), age (pcs 1 and 3), eyeglasses (pcs 4 and 5) and bangs (pc 4).

Inspecting how adding the weight vector transforms the isosurface we observe the same changes as in the interpolations: eyeglasses disappearing, the background becoming lighter, and camera position moving upwards. We also observe a racial bias: in principle components 2, 3 and 4 the skin color is notably lighter with high attractiveness. Principal component 3 (middle column in Figure 2a) shows tellingly how the model perceives "attractiveness": a middle-aged, blond, white male shares the same attractiveness score with a young black woman. To analyze the racial bias, we annotated 3250 randomly selected images whether they show a person with dark skin, according to the Fitzpatrick skin type V-VI (Fitzpatrick, 1986). We annotated 249 samples as dark skin. Of the persons with dark skin 17.7% were annotated as "Attractive". For light skin, the percentage of "Attractive" is 52.0%. The same bias is in the model. The partial correlation between model probabilities and the dark skin label is $r =$-0.124 $(-0.16, -0.09)_{95\%}$, when controling for all other labels. Uncorrected, the correlation is $r =$-0.187 $(-0.22, -0.15)_{95\%}$.

Black Mice In this case study, we apply our method on the BLACK MICE dataset (Andresen et al., 2020). In contrast to CELEBA, the images vary more strongly in location, size, posture, and camera

angle. The dataset contains a total of 32576 images of 126 individual mice. Andresen et al. (2020) trained a ResNet and reported an accuracy of 88.5±2.6% using 10-fold cross-validation. Our model achieves a similar accuracy of 86.5% tested on a single fold. The images were collected for earlier works (Hohlbaum et al., 2018; 2017). The mice were divided in three groups: castration, only anesthesia, or untreated. Any signs of post-surgical/-anesthetic effects are marked by a binary label. According to (Langford et al., 2010), typical features for pain are squeezed eyes, pulled-back ears, bulged cheeks and nose, and change in whisker position.

Together with the authors of (Andresen et al., 2020), we reviewed our model's explanations. We confirmed that counterfactuals affect different image features accordingly: eyes, nose, ears, whiskers, and head position change in biologically plausible ways. The mice's eyes seem to be less relevant to the network. For humans, squeezed eyes are a good indicator for pain. However, the counterfactuals only showed slight changes: sometimes the eyes blend into the surroundings. As neural networks perform well on the task using only the eyes (Neurath, 2020), we believe changes to our network architecture could preserve these details. Some other features may co-appear with image artifacts, e.g. the ear's shape changes may also appear partially blended with the background.

Intriguingly, the counterfactuals also show contrast changes in the surroundings, see figure 1b. The authors of (Andresen et al., 2020) voiced the suspicion that this may be explained by the way the photos were taken. Since mice after anesthesia or surgery predominantly drop the head and the nose tip points downwards, the camera angle may have been adjusted to get a better view of the animal and, in effect, show more of the yellow wooden bedding material on the cage floor.

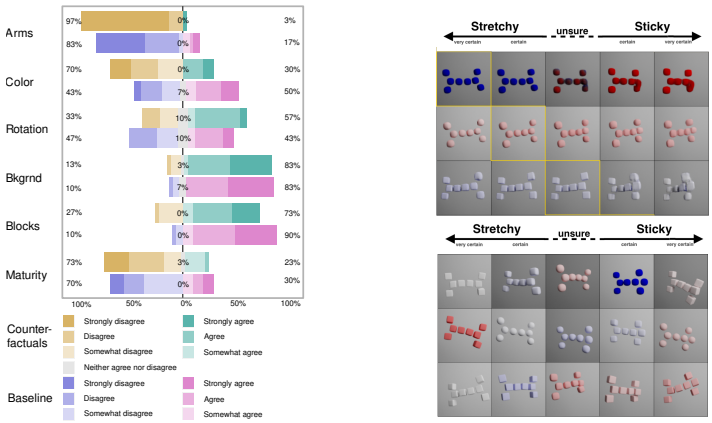
To verify if wooden bedding material is predictive, we annotated 1000 randomly selected images from our test set. Depending on the image area covered by the wooden bedding material, we assigned each sample to the classes: (0) $\leq 5\%$, (1) $\leq 20\%$, (2) $> 20\%$. if the bottom of the image showed yellow bedding material. This classification resulted in 346, 258, 396 samples per bin. Of all samples, 44.7% were marked to show post-surgical/-anesthetic effects. Per bin, the label was unevenly distributed: 19.9%, 52.3%, 61.4%. We account for the unequal distribution of labels using partial correlation (see Appendix D) and obtain the following values between the models output probabilities and the bins (95% CI): (1) -0.255 (-0.31, -0.20)$_{95\%}$, (2) 0.026 (-0.04, 0.09)$_{95\%}$, (3) 0.217 (0.16, 0.27)$_{95\%}$. These results confirm a connection between the surroundings, label, and logit score. The hint of our explanations to this bias in the data was correct.

The surroundings' changes can be explained probably by mice dropping their head if in pain and by changes to the camera angle. As we were also able to confirm many characteristic features, the network does not base its decision solely on wooden bedding material. This case study highlights the practicability of our method in a real-world scenario.

5 Human Subject Study

We conducted a preregistered between-group online study to evaluate whether counterfactual interpolations can help users identify neural network patterns and biases. We chose the Two4Two data set (Section 4) for the study because it was specifically designed for *human-grounded evaluations* (Anonymous, 2020; Doshi-Velez & Kim, 2017). The simple scenario of two abstract animals allows for the recruiting of none-experts. It also eliminates participants' prior knowledge as a cofounding factor in the experimental design. We assessed whether users could identify the two biases in color and rotation. Participants (N=60) were recruited from Prolific and needed to have an academic degree with basic math education.

Participants were randomly but equally assigned to view two either counterfactual interpolations or example-based explanations. In both conditions, we displayed a 30x5 image grid, each column representing a logit range. Ranges were chosen so that high certainty predictions for *Stretchy* were shown on the far left and high certainty predictions *Sticky* on the far right. Less certain prediction were shown in the directly adjoining columns. The remaining middle column represented borderline-cases. This visual design had prevailed throughout numerous iterations and ten pilot studies, as it allows users to quickly scan for visual similarities in columns and differences in rows. In the *example-based baseline condition*, the grid was filled with images drawn from validation set (see Figure 5b). In the *counterfactual interpolations conditions*, only the diagonal of the grid was filled with such "original" images. They were marked with a golden frame. The remaining cells were filled row-wise with



(a) (b)

Figure 5: Left: Participants agreements to statements about the patterns the network is using for its prediction. Right: The study interface in the counterfactual interpolations (Top) and baseline condition (Bottom)

counterfactual interpolations of the original images that matched the corresponding columns score range (see Figure 5b).

Upon commencing the study on the Qualtrics platform, participants were shown handcrafted video instructions. After that they studied the image grid while rating their agreement to six statements on a 7-point Likert scale. Five statements were concerned with patterns the network learned: arm position, rotation of animal, color of animal, background and block type. One statement concerned the maturity of the network for deployment. Participants also rated their agreement to four applicable statements taken from the Explanation Satisfaction Scale (Hoffman et al., 2018).

5.1 Study Results and Discussion

The significance of rating difference was assessed using a Kruskal-Wallis Test. For a detailed assessment of all preregistered hypothesis, please refer to the Appendix (Section F.1). Responses are summarized in Figure 5a.

Counterfactual interpolations allowed users to identify the model's main pattern: the position of the "arms" of Stretchy and Sticky. They did this with high certainty as 83.34 % strongly agreed with the corresponding statement. They were more certain about this pattern than with the baseline technique ($H(1) = 8.86$, $p = 0.003$), even though the baseline technique also performed well for this task. The large majority (70 %) also identified the color bias with counterfactual interpolations, while only 43% identified this bias using the baseline explanations. However, the difference in rating between conditions for the corresponding statement about color bias closely missed the threshold for significance ($H(1) = 3.21$, $p = 0.07$). Participants who had missed the color bias using our method were later asked to provide their reasoning. A participant stated: *"I would think that the color would be relevant if I saw an example where it went from certain to very certain and only the color, brightness or intensity changed."* Such rule-based rather than probabilistic cognitive models of the network may have led users to reject the presence of color bias even though we instructed them clearly that interpolation would only change relevant features.

To our surprise, few participants noticed the networks more subtle bias towards object rotation in both conditions. As Figure 5a indicates, participants were somewhat undecided about the relevance, leaning rather to conclude that the network is not sensitive to rotation. As a limitation, we note that participants may not have noticed the rotation bias due to how we had phrased the corresponding statement. When we asked them to explain their reasoning, many explained that they instead focused on the individual blocks' rotation rather than the whole animal.

Both explanation techniques allowed participants to confidently reject statements about irrelevant patterns (sensitivity to the background, sensitivity to the type of blocks). We argue this indicates a high quality of collected responses and good utility of both explanation techniques. Somewhat worrying is participants' assessment of the system's maturity. They were very confident that the network has learned the right patterns and is ready to be used for both techniques.

Explanation Satisfaction ratings were very high for both techniques, see Figure 9 in Appendix. As a limitation we note that participants may have found the introductory videos very informative as many of them reported enjoying the study. This may have led them to favorable ratings as well as the conclusion that they understand the system very well regardless of the explanation technique they had used.

6 Related Work

Counterfactuals have been generated before with the goal to explain the outcome of deep convolutional networks. (Goyal et al., 2019) finds regions in a distractor image that would change the prediciton if present. (Chang et al., 2019) evaluate heuristic and generative infilling strategies for counterfactual generation. Liu et al. (2019); Baumgartner et al. (2018) expanded the GAN framework to the generation of CFs. A limitation of GANs for CF is that forward (discriminator) and backward (generator) functions do not have to align perfectly. Another approach to interpretability uses the so-called concept vectors (Kim et al., 2018), which allow to investigate by how much a manually defined concept influences the final prediction. (Ghorbani et al., 2019) extend this framework to automatically discover concept by using super-pixels. However, all the above mentioned methods are post-hoc interpretability methods. Their main goal is to explain the decisions of a model after it has been trained, rather than providing an interpretable model directly.

The trick to interpolate in feature space and invert back is common for invertible networks (Jacobsen et al., 2018; Kingma & Dhariwal, 2018). Invertibility has been used to by (Mackowiak et al., 2020) to improve the trustworthiness but focus more on out-of-distribution and adversarial examples. (Rombach et al., 2020; Esser et al., 2020) use invertible networks to understand tradition vanilla convolutional networks better. The field of interpretable deep vision models, is not widely explored yet. One example of interpretable model is ProtoPNet Chen et al. (2019). Prototypical patches are compared in a convolutional feature map and used for the final prediction. Brendel & Bethge (2018) slices the image in patches and still reach state-of-the performance.

Our combination of PCA and invertible neural networks for interpretability is novel. Our finding, that the directional derivative corresponds to ideal counterfactuals whereas the gradient does not, was also not reported before. No prior human subject evaluation has demonstrated that counterfactual interpolations can help users identify neural network patterns and biases.

7 Discussion

Our method is based on well-established, straight-forward components, invertible networks and PCA. The combination yields a rich interpretability toolset: ideal counterfactuals, global explanations, and informative saliency maps. Using our method, we were able to reveal biases in all tested datasets.

Several undesirable properties of gradient-based methods have been described (Adebayo et al., 2018; Sixt et al., 2020). Here, we provide evidence that gradient-based methods do not explain the model, but rather provide adversarial examples that increase the logit score. Still, gradient-based methods are widely used (Eitel et al., 2020; Böhle et al., 2019; Sturm et al., 2016; Lapuschkin et al., 2017; Sipple, 2020), presumably because the gradient is a quantity readily available. We showed that the directional derivative $\mathrm{d}\boldsymbol{x}/\mathrm{d}\boldsymbol{w}$ has more desirable properties for interpretability. However, it requires an invertible network. Given the advantages of invertible neural networks for interpretability, we think the added computational costs due to the generative model can be justified. Furthermore, our technique may simply be used to check for biases in a dataset while a smaller model is deployed.

Our user study confirmed our explanations technique's utility, as it allowed users to find the major patterns present in the data and one bias while rejecting all irrelevant patterns.

We focused on convolutional networks but the general idea should work also for other network types, as it depends mostly on the existence of an inverse which is available for recurrent and graph neural networks MacKay et al. (2018); Madhawa et al. (2019).

In this work, we mainly demonstrated the potential of this method to reveal biases through human inspection. We believe that invertibility for interpretability is an underexplored topic. Future work, might be able to quantify biases or even correct them automatically.

REFERENCES

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pp. 9505–9515, 2018.

Christopher J Anders, Plamen Pasliev, Ann-Kathrin Dombrowski, Klaus-Robert Müller, and Pan Kessel. Fairwashing explanations with off-manifold detergent. *arXiv preprint arXiv:2007.09969*, 2020.

Niek Andresen, Manuel Wöllhaf, Katharina Hohlbaum, Lars Lewejohann, Olaf Hellwich, Christa Thöne-Reineke, and Vitaly Belik. Towards a fully automated surveillance of well-being status in laboratory mice using deep learning: Starting with facial expression analysis. *Plos one*, 15(4): e0228059, 2020.

Anonymous. Two4two dataset. *Under Review*, 2020.

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), 2015. doi: 10.1371/journal.pone.0130140.

David Balduzzi, Marcus Frean, Lennox Leary, JP Lewis, Kurt Wan-Duo Ma, and Brian McWilliams. The shattered gradients problem: If resnets are the answer, then what is the question? In *International Conference on Machine Learning*, pp. 342–350, 2017.

Christian F Baumgartner, Lisa M Koch, Kerem Can Tezcan, Jia Xi Ang, and Ender Konukoglu. Visual feature attribution using wasserstein gans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 8309–8319, 2018.

Moritz Böhle, Fabian Eitel, Martin Weygandt, and Kerstin Ritter. Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification. *Frontiers in aging neuroscience*, 11:194, 2019.

Wieland Brendel and Matthias Bethge. Approximating cnns with bag-of-local-features models works surprisingly well on imagenet. In *International Conference on Learning Representations*, 2018.

Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and D. Duvenaud. Explaining image classifiers by counterfactual generation. In *ICLR*, 2019.

Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. In *Advances in neural information processing systems*, pp. 8930–8941, 2019.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803*, 2016.

Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. In *Advances in Neural Information Processing Systems*, pp. 13589–13600, 2019.

Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv: 1702.08608*, 2017.

Fabian Eitel, J. P. Albrecht, M. Weygandt, F. Paul, and Kerstin Ritter. Harnessing spatial homogeneity of neuroimaging data: patch individual filter layers for cnns. *ArXiv*, abs/2007.11899, 2020.

Patrick Esser, Robin Rombach, and Bjorn Ommer. A disentangling invertible interpretation network for explaining latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9223–9232, 2020.

T. Fitzpatrick. Ultraviolet-induced pigmentary changes: benefits and hazards. *Current problems in dermatology*, 15:25–38, 1986.

Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, pp. 9277–9286, 2019.

Yash Goyal, Z. Wu, J. Ernst, Dhruv Batra, D. Parikh, and Stefan Lee. Counterfactual visual explanations. *ArXiv*, abs/1904.07451, 2019.

Robert R Hoffman, Shane T Mueller, Gary Klein, and Jordan Litman. Metrics for explainable ai: Challenges and prospects. *arXiv preprint arXiv:1812.04608*, 2018.

Katharina Hohlbaum, Bettina Bert, Silke Dietze, Rupert Palme, Heidrun Fink, and Christa Thöne-Reineke. Severity classification of repeated isoflurane anesthesia in c57bl/6jrj mice—assessing the degree of distress. *PLOS ONE*, 12:1–21, 06 2017. doi: 10.1371/journal.pone.0179588. URL https://doi.org/10.1371/journal.pone.0179588.

Katharina Hohlbaum, Bettina Bert, Silke Dietze, Rupert Palme, Heidrun Fink, and Christa Thöne-Reineke. Impact of repeated anesthesia with ketamine and xylazine on the well-being of c57bl/6jrj mice. *PLOS ONE*, 13(9):1–24, 09 2018. doi: 10.1371/journal.pone.0203559. URL https://doi.org/10.1371/journal.pone.0203559.

Jörn-Henrik Jacobsen, Arnold W.M. Smeulders, and Edouard Oyallon. i-revnet: Deep invertible networks. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=HJsjkMb0Z.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International Conference on Machine Learning*, pp. 2673–2682, 2018.

Diederik P. Kingma and P. Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. *ArXiv*, abs/1807.03039, 2018.

Dale J Langford, Andrea L Bailey, Mona Lisa Chanda, Sarah E Clarke, Tanya E Drummond, Stephanie Echols, Sarah Glick, Joelle Ingrao, Tammy Klassen-Ross, Michael L LaCroix-Fralish, Lynn Matsumiya, Robert E Sorge, Susana G Sotocinal, John M Tabaka, David Wong, Arn M J M van den Maagdenberg, Michel D Ferrari, Kenneth D Craig, and Jeffrey S Mogil. Coding of facial expressions of pain in the laboratory mouse. *Nature Methods*, 7(6):447–449, June 2010. ISSN 1548-7091, 1548-7105. doi: 10.1038/nmeth.1455. URL http://www.nature.com/articles/nmeth.1455.

Sebastian Lapuschkin, Alexander Binder, Klaus-Robert Muller, and Wojciech Samek. Understanding and comparing deep neural networks for age and gender classification. In *The IEEE International Conference on Computer Vision Workshops (ICCVW)*, Oct 2017.

Shusen Liu, Bhavya Kailkhura, Donald Loveland, and Yong Han. Generative counterfactual introspection for explainable deep learning, 2019.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, December 2015.

Matthew MacKay, Paul Vicol, Jimmy Ba, and Roger B Grosse. Reversible recurrent neural networks. In *Advances in Neural Information Processing Systems*, pp. 9029–9040, 2018.

Radek Mackowiak, Lynton Ardizzone, Ullrich Köthe, and Carsten Rother. Generative classifiers as a basis for trustworthy computer vision. *arXiv preprint arXiv:2007.15036*, 2020.

Kaushalya Madhawa, Katushiko Ishiguro, Kosuke Nakago, and Motoki Abe. Graphnvp: An invertible flow model for generating molecular graphs. *arXiv preprint arXiv:1905.11600*, 2019.

Grégoire Montavon, Alexander Binder, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. Layer-wise relevance propagation: an overview. In *Explainable AI: interpreting, explaining and visualizing deep learning*, pp. 193–209. Springer, 2019.

Daniel Neurath. Analysis of the eye as a visual indicator for the well-being status of laboratory mice. *Unpublished Bachelor Thesis*, 2020.

Robin Rombach, Patrick Esser, and Björn Ommer. Making sense of cnns: Interpreting deep representations & their invariances with inns. In *Proceedings of the European Conference on Computer Vision*, 2020.

Sumedha Singla, Brian Pollack, Junxiang Chen, and Kayhan Batmanghelich. Explanation by progressive exaggeration. In *International Conference on Learning Representations*, 2020.

John Sipple. Interpretable, multidimensional, multimodal anomaly detection with negative sampling for detection of device failure, 2020.

Leon Sixt, Maximilian Granz, and Tim Landgraf. When explanations lie: Why many modified bp attributions fail. In *Proceedings of the 37th International Conference on Machine Learning*. PMLR, 2020.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv: 1706.03825*, 2017.

Irene Sturm, Sebastian Lapuschkin, Wojciech Samek, and Klaus-Robert Müller. Interpretable deep neural networks for single-trial eeg classification. *Journal of neuroscience methods*, 274:141–145, 2016.

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

Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harvard Journal of Law & Technology*, 31(2), 2018.

Dataset	NLL	Accuracy	Supervised
BLACK MICE	3.28	86.5	88.5±2.5 (Andresen et al., 2020)
CELEBA	2.55	88.2	
TWO4TWO	0.63	84.9	

Table 1: Model Performances on the different datasets. Negative Log Likelihood in bits/pixels.

Parameter	Corr. Data	CI95%	Corr. Change	CI 95%
Color	0.329	0.27, 0.38	0.352	0.33, 0.37
Background	0.022	-0.04, 0.08	-0.037	-0.06, -0.01
Incline	0.032	-0.03, 0.09	0.003	-0.02, 0.02
Arm Position	-0.799	-0.82, -0.78	-0.798	-0.81, -0.79
Spherical	-0.053	-0.11, 0.01	-0.010	-0.03, 0.01
Abs. Rotation	0.060	-0.00, 0.12	0.136	0.11, 0.16

Table 2: TWO4TWO: Correlation between object parameters and the model's output probabilities. *Corr. Data*: Correlation over the joint distribution. *Corr. Change*: Correlation if only the parameter is changed.

A APPENDIX: TECHNICAL DETAILS

A.1 NEURAL NETWORK ARCHITECTURE

Our model follows the Glow model closely (Kingma & Dhariwal, 2018). Similarly, we use a block of *actnorm*, *invertible* 1×1 *convolution* and *affine coupling* layer. After 18 blocks, we add an reshuffle operation to reduce the spatial dimensions by a factor of 2 and half of the channels are faded out. The first layer is The classification is done before the final mapping to the prior $\mathcal{N}(0, 1)$.

As described in section 2, we trained added the classifier after layer 321 before the final layer 351. Let φ denote the first 321 layers and $\mu : \mathbb{R}^n \mapsto \mathbb{R}^n$ the last. We train φ both on a supervised loss from the classifier $f(\boldsymbol{x})$ and an unsupervised loss from matching the prior distribution $\mathcal{N}(0, I)$ and the log determinante of the Jacobian. μ is only trained on the unsupervised loss:

$$\arg \min_{\theta_\varphi, \theta_\mu, \theta_f} L_{\text{un}}(\mu \circ \varphi(\boldsymbol{x})) + \beta \, L_{\text{sup}}(\boldsymbol{w}^T \varphi(\boldsymbol{x}) + b, \, y_{\text{true}}). \tag{1}$$

For the supervised loss L_{sup}, we use the binary cross entropy although our method is not restricted to this loss function and could be extend to more complex losses easily. As unsupervised loss L_{un}, we use the commonly used standard flow loss obtained from the change of variables trick Dinh et al. (2016). The unsupervised loss ensures that inverting the function results in realistic looking images and can also be seen as a regularization.

In total, φ and μ have 158.769.600 parameters. We use the identical network architecture on all datasets.

A.2 DETAIL TO INTEGRATION: SECTION 3

In section 3, we integrated the gradient and the directional dirivative. We used the `torchdiffeq` package. For figure 4a, we integrated from t=[0, 11] using the midpoint method with 20 steps. Here the integration was done in layer 40. As this was rather slow, we used 5steps and t = [0, 4] to determine the differences in the unrelated factors $\boldsymbol{\xi}$ again in 40, shown in Figure 4b.

Table 3: Bias in the BLACK MICE dataset. The post-surgical/-anesthetic effects label is unevenly distributed across the bins (0-2) for the amount of yellow bedding material present in an image. The classifier's probabilties are correlated negatively with (0) fewer bedding material and positively with more (2). When we account for the effect of unequal label distribution using partial correlation, the output probabilities and bins (0 & 2) are still correlated.

Bin	Data	Method	Corr.	CI 95%
(0) ≤ 5	Label	full	-0.362	-0.410, -0.310
(1) ≤ 20	Label	full	0.090	0.030, 0.150
(2) > 20	Label	full	0.271	0.210, 0.330
(0) ≤ 5	Pred. Prob.	full	-0.429	-0.480, -0.380
(1) ≤ 20	Pred. Prob.	full	0.085	0.020, 0.150
(2) > 20	Pred. Prob.	full	0.341	0.290, 0.390
(1) ≤ 5	Pred. Prob.	partial	-0.255	-0.310, -0.200
(2) ≤ 20	Pred. Prob.	partial	0.026	-0.040, 0.090
(3) > 20	Pred. Prob.	partial	0.217	0.160, 0.270

B APPENDIX: PCA AND UL FLOW TRAINING

C TWO4TWO

D MICE DATASET STATISTICAL ANALYSIS

As described in section 4, the label "post-surgical/-anesthetic effects" is unequally distributed across the three bins: 346, 258, 396. This can be problematic, when we measure the correlation between sample's bin and the model logit score. The model has learned to predict lower scores for a negative label and vise versa. To account for this, we calculate the partial correlation between the model's output probability and the bin class while using the label as a confounding variable. In table 3, we report both full and partial correlations and also the correlations of the bins with the label.

E ISOSURFACE VISUALIZATION

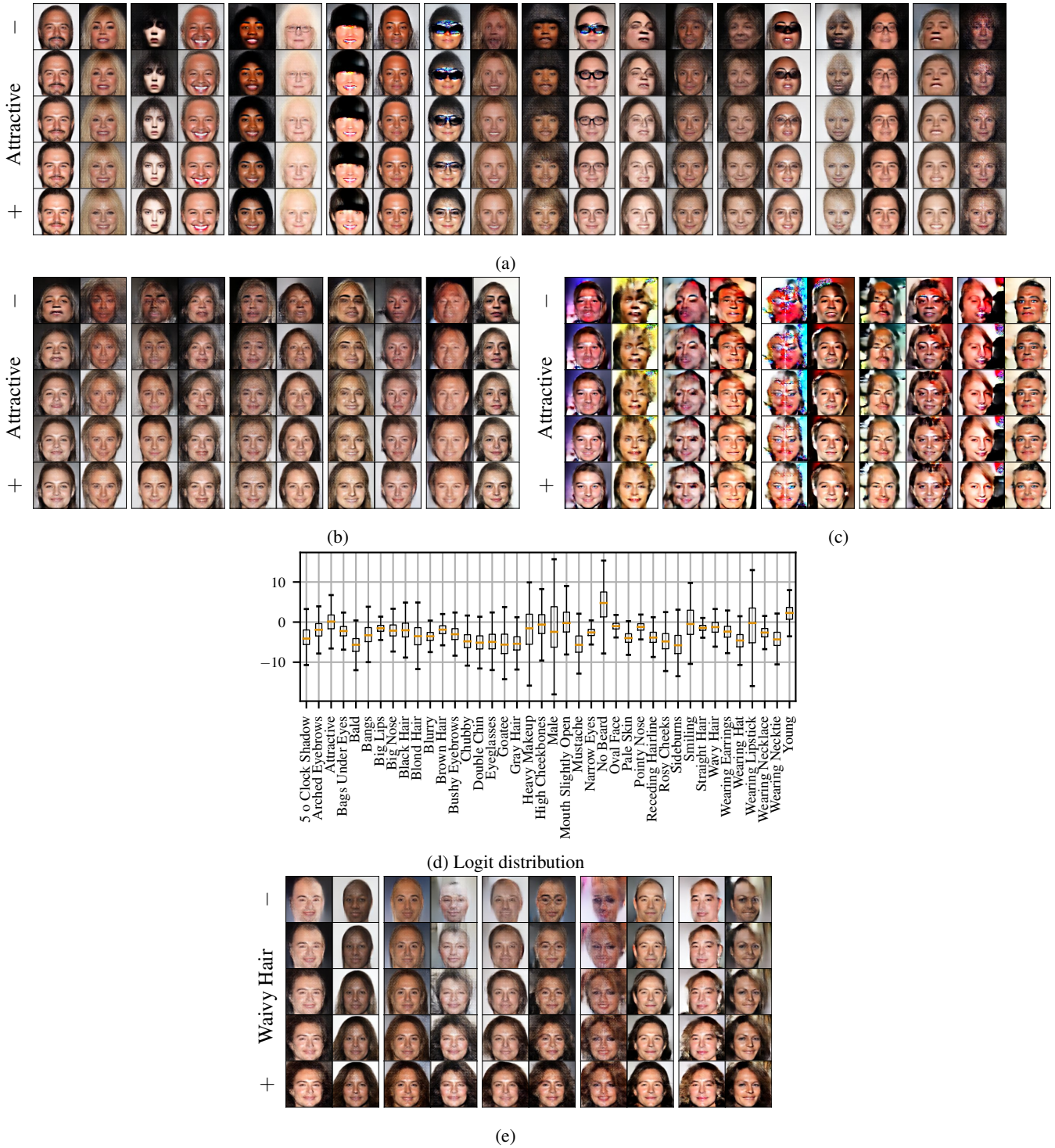
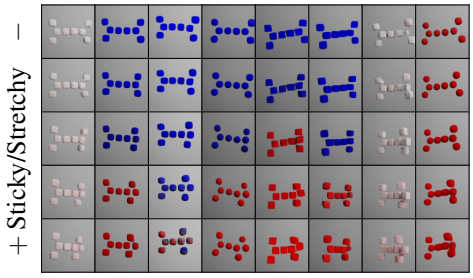
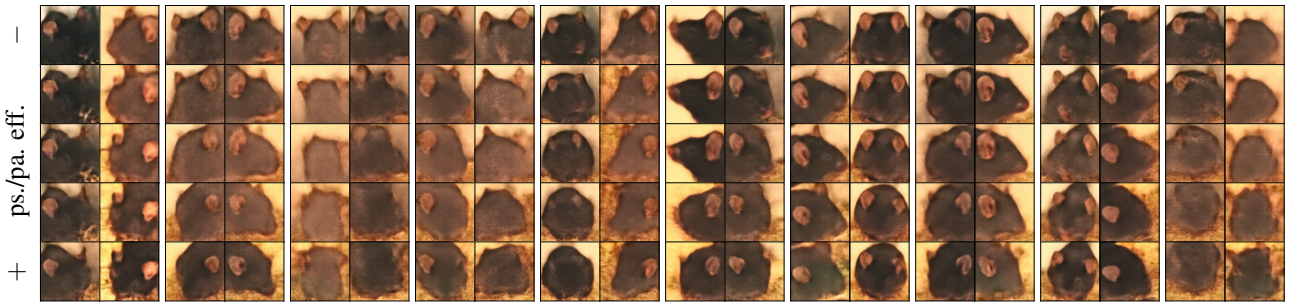


Figure 6: Eigenvectors of the isosurface for the "attractiveness" class. **(a)** Eigenvectors with the higest explained variance **(b)** Randomly selected eigenvectors: 82, 129, 131, 157, 224 **(c)** Eigenvectors with the smallest explained variance

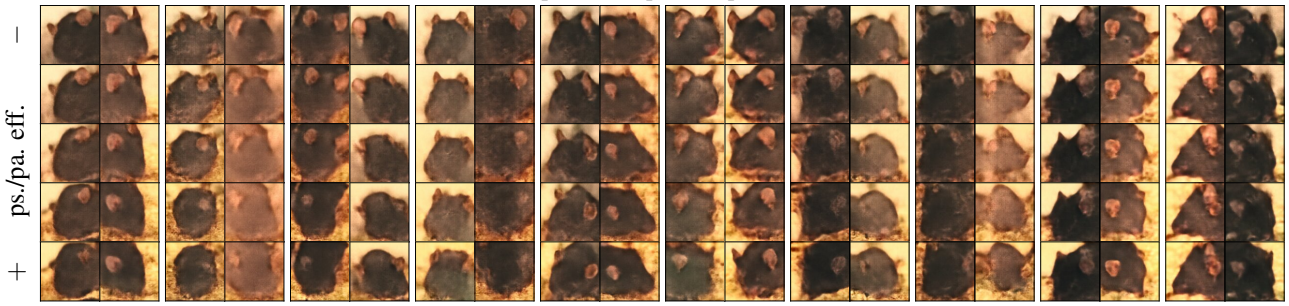


(a) Two4Two

Figure 7: Random Principal Components for the Two4Two dataset.

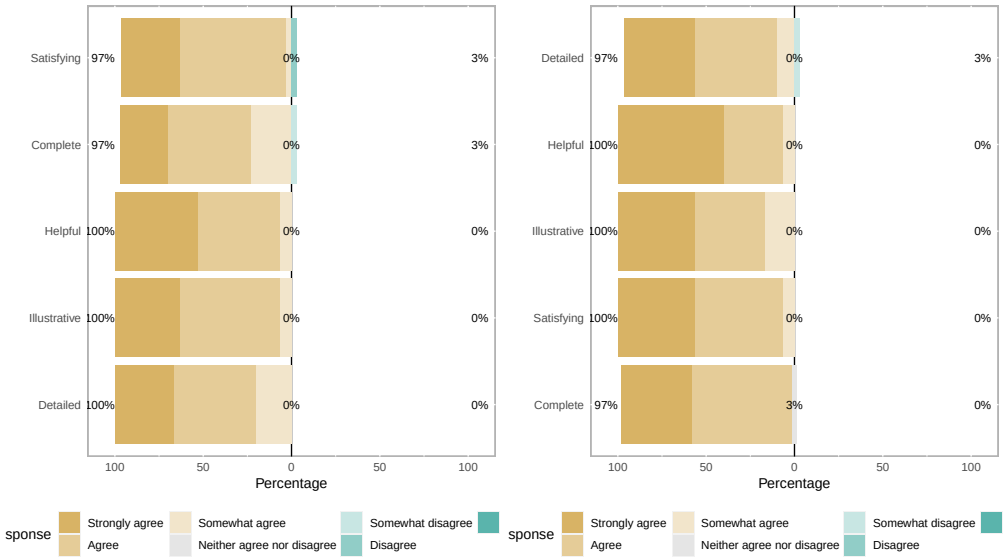


(a) Top 10 Principal Components



(b) Randomly

Figure 8: Random Principal Components: 82, 95, 100, 144, 145, 172, 195, 240, 415, 564



(a) (b)

Figure 9: Explanation Satisfaction Ratings by our study participants for counterfactual interpolations (a) and example-based explanation baseline (b)

F User-study Preregistration and Hypothesis

The Study is preregisterd at `https://aspredicted.org/` we provide and anonimyzed pdf version in supplemental material. Participants (N=60), of the study were required to fluent in English and needed to have an approval rate of at least 95. Given the demanding nature of the task and the complex of concepts used in the instructions they also needed to have an academic degree in Computer Science, Engineering, Finance, Mathematics, Medicine, Physics or Psychology. Figure 9 summarizes the subjective ratings participants gave about the two explanation technqiues used in the study.

F.1 Evaluation of preregistered Hypothesis

H1: The hypothesis *"Studying the system's predictions on the validation set (Baseline Explanation technique - referred to as Baseline) allows users to verify that the neural network (NN) is using the blocks spatial arrangement (Pattern 1) for its predictions of the abstract animals."* is confirmed as 83.33% at least somewhat agree to corresponding statement.

H2: The hypothesis *"Baseline does not allow users to detect the NN bias for colour (Pattern 2) and rotation (Pattern 3)."* is confirmed. Only 46.66 % of users at least somewhat disagree with the statement claiming that there is a rotation pattern while only 43.33% at least somewhat agree (the remaining are undecided). For the colour pattern 50% at least somewhat disagree that there is such a pattern and only 43.33% at leat somewhat agree.

H3: Duplicate of H2 (copy and paste error during preregistration)

H4: The hypothesis *"Studying the system's predictions with counterfactual interpolations as explanations (referred to as NNWI) allows users to verify that NN is using Pattern 1."* is confirmed as 96.66% at least agree with the corresponding statement.

H5: The hypothesis *"Counterfactual interpolations allows users to detect Pattern 2 and Pattern 3."* is rejected. While 70 % at least somewhat agree with the statement about Pattern 2 only 33.33% at least somewhat agree with the statement about Pattern 3.

H6: The hypothesis *"Counterfactual interpolations allows users to verify that NN is neither using the background of the image (Pattern 4) nor the surface structure of objects (Pattern 5)."* is confirmed.

The corresponding statement about Pattern 4 and 5 have been at least somewhat disagreed to by 83.33% and 73.33% respectively.

H7: The hypothesis *"Counterfactual interpolations allow users to detect Pattern 1 with higher confidence"* is confirmed. Agreement with the corresponding statement was significantly different between conditions (p = 0.003) and on average higher for Counterfactual Interpolations (2.67) compared to the baseline (1.76).

H8: The hypothesis *"Counterfactual interpolations allow users to reject Pattern 4 and Pattern 5 with higher confidence"* is rejected. There was no significant difference in the certainty for disagreeing with corresponding statements.

H9: The hypothesis *"Counterfactual interpolations allow users to detect Pattern 2 and Pattern 3 with higher confidence."* is rejected since H5 was rejected. However, it is worth pointing out that for the statement about color 70% of participants at least somewhat agreed to it if they received counterfactual interpolation while only 43.33% at least somewhat agreed to it if they received example based explanations. This difference was almost significant (p-value = 0.07).

H10: The hypothesis *"Counterfactual interpolations leads users to be more confident in the maturenessof the system."* is rejected as agreement to the corresponding statement was not significantly different across conditions. In both conditions participants where rather confident in the system.

H11: The hypothesis *"Users are more satisfied with Counterfactual interpolations as explanations."* is rejected. Explanation Satisfaction rating were very high in both conditions but not significantly different.

G IMAGE SOURCE

As the copyright of CELEBA is unclear and includes images under no free license, we decided against showing any original CELEBA images in the paper. We show the following these six images all under permissive license: Obama (CC BY 3.0): `https://commons.wikimedia.org/wiki/File:Official_portrait_of_Barack_Obama.jpg`

Commander, Eileen M. Collins (Public Domain): `https://www.flickr.com/photos/nasacommons/16504233985/`

Carl Jacobi (Public Domain): `https://de.wikipedia.org/wiki/Datei:Carl_Jacobi.jpg`

Grace Hopper (Public domain): `https://de.wikipedia.org/wiki/Datei:Grace_Hopper.jpg`

Alan Touring (CC BY 4.0): `https://commons.wikimedia.org/wiki/File:%D0%A2%D1%8C%D1%8E%D1%80%D0%B8%D0%BD%D0%B3.jpg`

Lyndsey Scott (CC BY 4.0): `https://en.wikipedia.org/wiki/File:Lyndsey_Scott_being_combed.jpg`