ExplainGAN: Model Explanation via Decision Boundary Crossing Transformations

Pouya Samangouei[1,2][0000−0002−4443−5175], Ardavan Saeedi[2][0000−0001−7763−7980], Liam Nakagawa[2]
Nathan Silberman[2][0000−0002−8498−5796]

[1] University of Maryland, College Park, MD, 20740
pouya@umiacs.umd.edu
[2] Butterfly Network, New York, NY, 10001
{asaeedi,nakagawaliam,nsilberman}@butterflynetinc.com

Abstract. We introduce a new method for interpreting computer vision models: visually perceptible, decision-boundary crossing transformations. Our goal is to answer a simple question: why did a model classify an image as being of class A instead of class B? Existing approaches to model interpretation, including saliency and explanation-by-nearest neighbor, fail to visually illustrate examples of transformations required for a specific input to alter a model's prediction. On the other hand, algorithms for creating decision-boundary crossing transformations (e.g., adversarial examples) produce differences that are visually imperceptible and do not enable insightful explanation. To address this we introduce ExplainGAN, a generative model that produces visually perceptible decision-boundary crossing transformations. These transformations provide high-level conceptual insights which illustrate how a model makes decisions. We validate our model using both traditional quantitative interpretation metrics and introduce a new validation scheme for our approach and generative models more generally.

Keywords: Neural Networks, Model Interpretation

1 Introduction

Given a classifier, one may ask: What high-level, semantic features of an input is the model using to discriminate between specific classes? Being able to reliably answer this question amounts to an understanding of the classifier's decision boundary at the level of concepts or attributes, rather than pixel-level statistics.

The ability to produce a conceptual understanding of a model's decision boundary would be extremely powerful. It would enable researchers to ensure that a model is extracting relevant, high-level concepts, rather than picking up on spurious features of a dataset. For example, criminal justice systems could determine whether their ethical standards were consistent with that of a model [8]. Additionally, it would provide some measure of validation to consumers (e.g., medical applications, self-driving cars) that a model is making decisions that are difficult to formalize and automatically verify.

Unfortunately, directly visualizing or interpreting decision boundaries in high dimensions is effectively impossible and existing post-hoc interpretation methods fall short of adequately solving this problem. Dimensionality reduction approaches, such as T-SNE [15], are often highly sensitive to their hyper-parameters whose values may drastically alter the visualization [27]. Saliency maps are typically designed to highlight the set of pixels that contributed highly to a particular classification. While they can be useful for explaining factors that are present; they cannot adequately describe predictions made due to objects that are missing from the input. Explanation-by-Nearest-Neighbor-Example can indeed demonstrate similar images to a particular query, but there is no guarantee that similar enough images exist to be useful and similarity itself is often ill-defined.

To overcome these limitations, we introduce a novel technique for post-hoc model explanation. Our approach visually explains a model's decisions by producing images on either side of its decision boundary whose differences are perceptually clear. Such an approach makes it possible for a practitioner to conceptualize how a model is making its decisions at the level of semantics or concepts, rather than vectors or pixels.

Our algorithm is motivated by recent successes in both pixel-wise domain adaptation [2,12,30] and style transfer [9] in which generative models are used to transform images from one domain to another. Given a pre-trained classifier, we introduce a second, post-hoc explaining network called ExplainGAN, that takes a query image that falls on one side of the decision boundary and produces a transformed version of this image that falls on the other. ExplainGAN exhibits three important properties that make it ideal for post-hoc model interpretation:

Easily Visualizable Differences: Adversarial example [26] algorithms produce decision boundary crossing images whose differences from the originals are not perceptible, by design. In contrast, our model transforms the input image in a manner that is clearly detectable by the human eye.

Localized Differences: Style transfer [5] and domain adaptation approaches typically produce low-level, global changes. If every pixel in the image changes, even slightly, it is not clear which of those changes actually influenced the classifier to produce a different prediction. In contrast, our model yields changes that are spatially localized. Such sparse changes are more easily interpretable by a viewer as fewer elements change.

Semantically Consistent: Our model must be consistent with the behavior of the pre-trained classifier to be useful: the class predicted for a transformed image must not match with the predicted class of the original image.

We evaluate our model using standard approaches as well as a new metric for evaluating this new style of model interpretation by visualizing boundary-crossing transformations. We also utilize a new medical images dataset where the concept of objectness is not well defined, making it less amenable to domain adaptation approaches that hinge on identifying an object and altering / removing it. Furthermore, this dataset represents a clear and practical use-case for model explanation. To summarize, our work makes several contributions:

1. A new approach to model interpretation: visualizing human-interpretable, decision-boundary crossing images.
2. A new model, ExplainGAN, that produces post-hoc model-explanations via such decision-boundary crossing images.
3. A new metric for evaluating the amount of information retained in decision-boundary crossing transformations.
4. A new and challenging medical image dataset.

2 Related work

Post-Hoc Model Interpretation methods typically seek to provide some kind of visualization of why a model has made a particular decision in terms of the saliency of local regions of an input image. These approaches broadly fall into two main categories: perturbation-based methods and gradient-based methods.

Perturbation-based methods [29,3], perturb the input image and evaluate the consequent change in the output of the classifier. Such perturbations remove information from specific regions of the input by applying blur or noise, among other pixel manipulations. Perturbation-based methods require multiple iterations and are computationally more costly than activation-based methods.

The perturbation of finer regions also makes these methods vulnerable to the artifacts of the classifier, potentially resulting in the assignment of high saliency to arbitrary, uninterpretable image regions. In order to combat these artifacts, current methods such as [3] are forced to perturb larger, less precise regions of the input.

Gradient-based methods such as [23,25,21,22,24] backpropagate the gradient for a given class label to the input image and estimate how moving along the gradient affects the output. Although these methods are computationally more efficient compared to perturbation-based methods, they rely on heuristics for backpropagation and may not support different network architectures.

A subset of gradient-based methods, which we call activation-based methods, also incorporate neuron activations into their explanations. Methods such as Gradient-weighted Class Activation Mapping Grad-CAM [20], layer-wise Relevance Propagation (LRP) [1] and Deep Taylor Decomposition (DTD) [16] can be considered as activation-based methods. Grad-CAM visualizes the linear combination of (typically) the last convolution layer and class specific gradients. LRP and DTD decompose the activations of each neuron in terms of contributions (i.e. relevances) from its input.

All these explanation methods are based on identifying pixels which contribute the most to the model output. In other words, these methods explain a model's decision by illustrating which pixels most affect a classifier's prediction. This takes the form of an attribution map, a heat map of the same size as the input image, in which each element of the attribution map indicates the degree to which its associated pixel contributed to the model output. In contrast, our model takes a different approach by generating a similar image on the other side of the model's decision boundary.

Adversarial Examples [26,7] are created by performing minute perturbations to image pixels to produce decision-boundary crossing transformations which are visually imperceptible to human observers. Such approaches are extremely useful for exploring ways in which a classifier might be attacked. They do not, however, provide any high-level intuition for why a model is making a particular decision.

Image-to-Image Transformation approaches, such as those used in domain adaptation [2,13,4] have shown increased success in transforming an image in one domain to appear as if drawn from another domain, such as synthetic-to-real or winter-to-summer. These approaches are clearly the most similar to our own in that we seek to transform images predicted as one class to appear to a pre-trained classifier as those from another. These approaches do not, however, constrain the types of transformations allowed and we demonstrate (Section 5.3) that significant constraints must be applied (Section 4) to ensure that the transformations produced are easily interpretable. Other image-to-image techniques such as Style Transfer [30,5,6] typically produce very low-level and comprehensive transformations to every pixel. In contrast, our own approach seeks highly localized and high-level, semantic changes.

3 Model

The goal of our model is to take a pre-trained binary classifier and a query image and generate both a new, transformed image and a binary mask. The transformed image should be similar to the query image, excepting a visually perceptible difference, such that the pre-trained classifier assigns different labels to the query and transformed image. The binary mask indicates which pixels from the query image where changed in order to produce the transformed image. In this way, our model is able to produce a decision-boundary crossing transformation of the query image and illustrate both *where*, via the binary mask, and *how*, via the transformed image, the transformation occurs.

More formally, given a binary classifier $\mathrm{F}(x) \in \{0, 1\}$ operating on an image x, we seek to learn a function which predicts a transformed image t and a mask m such that:

$$\mathrm{F}(x) \neq \mathrm{F}(t) \tag{1}$$

$$x \odot m \neq t \odot m \tag{2}$$

$$x \odot \neg m = t \odot \neg m \tag{3}$$

where Eq. (1) indicates that the model believes x and t to be of different classes, Eq. (2) indicates that the query and transformed image differ in pixels whose mask values are 1 and Eq. (3) indicates that the query and transformed image match in pixels where mask values are 0.

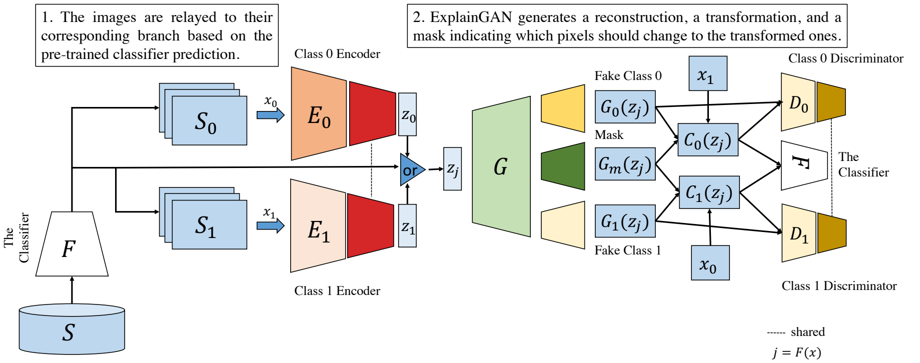

1. The images are relayed to their corresponding branch based on the pre-trained classifier prediction.
2. ExplainGAN generates a reconstruction, a transformation, and a mask indicating which pixels should change to the transformed ones.
The Classifier
F
S
S0
S1
x0
x1
Class 0 Encoder
E0
E1
Class 1 Encoder
z0
z1
or
zj
G
Fake Class 0
G0(zj)
Mask
Gm(zj)
G1(zj)
Fake Class 1
x1
x0
C0(zj)
C1(zj)
Class 0 Discriminator
D0
F
The Classifier
D1
Class 1 Discriminator
------ shared
j = F(x)

Fig. 1. Model architecture of ExplainGAN. Inference (in blue frame) consists of passing an image x of class j into the appropriate encoder E_j to produce a hidden vector z_j. The hidden vector is decoded to simultaneously create its reconstruction $G_j(z_j)$, a transformed image of the opposite class $G_{1-j}(z_j)$ and a mask showing where the changes were made $G_m(z_j)$. Composite images C_0 and C_1 merge the reconstruction and transformation with the original image x.

3.1 Prerequisites

Given a dataset of images $S = \{x_i | i \in 1 \dots N\}$, our pre-trained classifier produces a set of predictions $\{\bar{y}_i | i \in 1 \dots N\}$. Given these predictions, we now can split the dataset into two groups $S_0 = \{x_i | \bar{y}_i = 0\}$ and $S_1 = \{x_i | \bar{y}_i = 1\}$.

3.2 Inference

Given a query image and a predicted label for that image, our model maps to a reconstructed version of that image, an image of the opposite class and a mask that indicates which pixels it changed. Formally, our model is composed of several components. First, our model uses two class-specific encoders to produce hidden codes:

$$z_j = \mathrm{E}_j(x) \quad j \in \{0, 1\}, \quad x \in S_j \tag{4}$$

Next, a decoder G maps the hidden representation z_j to a reconstructed image $G_j(z_j)$, a transformed image of the opposite class $G_{1-j}(z_j)$ and a mask indicating which pixels changed $G_m(z_j)$. In this manner, images of either class can be transformed into similar looking images of the opposite class with a visually interpretable change.

We also define the concept of a composite image $\mathrm{C}_j(x)$ of class j:

$$\mathrm{C}_j(x_{1-j}) = x_{1-j} \odot (1 - \mathrm{G}_m(z_{1-j})) + \mathrm{G}_j(z_{1-j}) \odot \mathrm{G}_m(z_{1-j}) \tag{5}$$

where z_{1-j} is the code produced by encoding x_{1-j}. The composite image uses the mask to blend the original image x with either the reconstruction or the transformed image.

3.3 Training

To train the model, several auxiliary components of the network are required. First, two discriminators $\mathrm{D}_j(x) \rightarrow \{\mathrm{real}, \mathrm{fake}\}, j \in \{0, 1\}$ are trained to evaluate between real and fake images of class j.

To train the model we optimize the following objective:

$$\min_{\mathrm{G},\mathrm{E}_0,\mathrm{E}_1} \max_{\mathrm{D}_0,\mathrm{D}_1} \mathcal{L}_{\mathrm{GAN}} + \mathcal{L}_{\mathrm{classifier}} + \mathcal{L}_{\mathrm{recon}} + \mathcal{L}_{\mathrm{prior}} \tag{6}$$

where $\mathcal{L}_{\mathrm{GAN}}$ is a typical GAN loss, $\mathcal{L}_{\mathrm{classifier}}$ is a loss that encourages the generated and composite images to be likely according to the classifier, $\mathcal{L}_{\mathrm{recon}}$ ensures that the reconstructions are accurate, and $\mathcal{L}_{\mathrm{prior}}$ encodes our prior for the types of transformations we want to encourage. $\mathcal{L}_{\mathrm{GAN}}$ is a combination of the GAN losses for each class:

$$\mathcal{L}_{\mathrm{GAN}} = \mathcal{L}_{\mathrm{GAN:0}} + \mathcal{L}_{\mathrm{GAN:1}} \tag{7}$$

$\mathcal{L}_{\mathrm{GAN}:j}$ for class j discriminates between images x originally classified as class j and reconstructions of x, transformations from x and composites from x. It is defined as:

$$\begin{aligned} \mathcal{L}_{GAN:j} &= \mathbb{E}_{\mathbf{x} \sim S_j} \log(\mathrm{D}_j(x)) \\ &+ \mathbb{E}_{x \sim S_j} [\log(1 - \mathrm{D}_j(\mathrm{G}_j(\mathrm{E}_j(x))] \\ &+ \mathbb{E}_{x \sim S_{1-j}} [\log(1 - \mathrm{D}_j(\mathrm{G}_j(\mathrm{E}_{1-j}(x))] \\ &+ \mathbb{E}_{x \sim S_{1-j}} [\log(1 - \mathrm{D}_j(\mathrm{C}_j(\mathrm{E}_{1-j}(x))] \end{aligned} \tag{8}$$

Note that this formulation, in which the reconstructions of x are also penalized are part of ensuring that the auto-encoded images are accurate [10] and are included here, rather than as part of $\mathcal{L}$recon out of convenience.

Next, we encourage the composite images to produce images that the classifier correctly predicts:

$$\mathcal{L}_{\mathrm{classifier}} = \mathbb{E}_{x \in S_0} - \log(\mathrm{F}(\mathrm{C}_1(x))) \tag{9}$$

$$+ \mathbb{E}_{x \in S_1} - \log(1 - \mathrm{F}(\mathrm{C}_0(x)) \tag{10}$$

Finally, we have an auto-encoding loss for the reconstruction:

$$\mathcal{L}_{\mathrm{recon}} = \sum_{j \in 0,1} \mathbb{E}_{x \in S_j} \|G_j(E_j(x)) - x\|^2 \tag{11}$$

The mask priors are discussed in the following section.

4 Priors for Interpretable Image Transformations

There are many image transformations that will transform an image of one class to appear like an image from another class. Not all of these transformations, however, are equally useful for interpreting a model's behavior at a conceptual level. Adversarial example transformations will change the label but are not perceptible. Style transfer transformations make low-level but not semantic changes. Domain Adaptation approaches may change every pixel in the image which makes it difficult to determine which of these changes actually influenced the classifier. We want to craft set of priors that encourage transformations that are local to a particular part of the image and visually perceptible. To this end, we define our prior loss term as:

$$\mathcal{L}_{\text{prior}} = \mathcal{L}_{\text{const}} + \mathcal{L}_{\text{count}} + \mathcal{L}_{\text{smoothness}} + \mathcal{L}_{\text{entropy}} \tag{12}$$

The **consistency loss** $\mathcal{L}_{\text{const}}$ ensures that if a pixel is not masked, then the transformed image hasn't altered it.

$$\mathcal{L}_{\text{const}} = \sum_{j \in 0,1} \mathbb{E}_{x \in S_j} [\|(\mathbf{1} - G_m(z_j)) \odot x_j - (\mathbf{1} - G_m(z_j)) \odot G_{1-j}(z_j)\|^2] \tag{13}$$

where $z_j = E_j(x)$. The **count loss** $\mathcal{L}_{\text{count}}$ allows us to encode prior information regarding a coarse estimate of the number of pixels we anticipate changing. We approximate the l_0 norm via an l_1 norm:

$$\mathcal{L}_{\text{count}} = \sum_{j \in 0,1} \mathbb{E}_{x \in S_j} [\max(\frac{1}{n}|G_m(z_j)|, \kappa)] \tag{14}$$

where κ is a constant that corresponds to the ratio of number of changed pixels to the total number of the pixels. The **smoothness loss** encourages masks that are localized by penalizing transitions via a total variation [18] penalty:

$$\mathcal{L}_{\text{smoothness}} = \sum_{j \in 0,1} \mathbb{E}_{x \in S_j} |\nabla G_m(z_j)| \tag{15}$$

Finally, we want to encourage the mask to be as binary as possible:

$$\mathcal{L}_{\text{entropy}} = \sum_{j \in 0,1} \mathbb{E}_{x \in S_j} [\| \min_{elementwise} (G_m(z_j), 1 - G_m(z_j)) \|] \tag{16}$$

5 Experiments

Our goal is to provide model explainability via visualization of samples on either side of a model's decision boundary. This is an entirely new way of performing model explanation and requires a unique approach to evaluation.

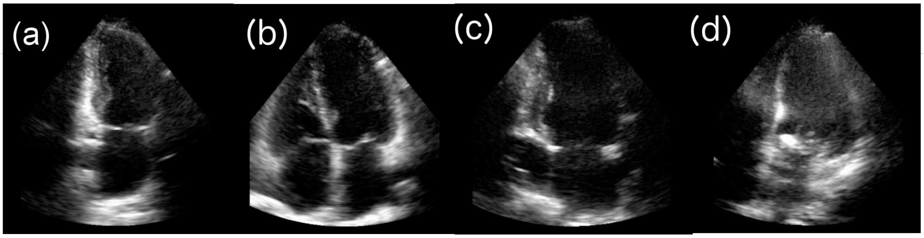


Fig. 2. An example of Ultrasound images from our Medical Ultrasound dataset. (a) A canonical Apical 2 Chamber view. (b) A canonical Apical 4 Chamber view. (c) A difficult Apical 2 Chamber view that is easily confused for a 4 Chamber view. (d) A difficult Apical 4 Chamber view that is easily confused for a 2 Chamber view.

To this end, we first demonstrate qualitative results of our approach and compare to related approaches (Section 5.3). Next, we evaluate our model using traditional criteria by demonstrating that our model's inferred masks are highly competitive as saliency maps when compared to state-of-the-art attribution approaches (Section 5.4). Next, we introduce two new metrics for evaluating the explainability of decision-boundary crossing examples (Section 5.5) and evaluate how our model performs using these quantitative methods.

5.1 Datasets

We used four datasets as part of our evaluation: MNIST [11], Fashion-MNIST [28], CelebA [14] and a new Medical Ultrasound dataset that will be released with the publication of this work. For each dataset, 4 splits were used: A classifier-training set used to train the black-box classifier, a training set used to train ExplainGAN, a validation set used to tune hyperparameters and a test set.

MNIST, Fashion-MNIST: We use the standard train/test splits in the following manner: The 60k training set is first split into 3 components: a 2k classifier-training set, a 50k training set and an 8k validation set. We used the standard test set. For MNIST, we used binary class pairs $(3, 8)$, $(4, 9)$ and $(5, 6)$. For Fashion-MNIST, we used binary class pairs (coat, shirt), (pullover, shirt) and (coat, pullover).

CelebA: We use the standard train/validation/test splits in the following manner: 2k images were used from the original validation set as the classifier-training set, all 160k images were used to train ExplainGAN, the remaining 14k validation images were used for validation. We used the standard test set. We used binary class pairs (glasses, no glasses) and (mustache, no mustache).

Medical Ultrasound: Our new medical ultrasound dataset is a collection of 72k cardiac images taken from 5 different views of the heart. Each image was labeled by several cardiac sonographers to determine the correct labels. An example of images from the dataset can be found in Fig. 2. As the Figure illustrates, the dataset is very challenging and is not as amenable to certain senses of 'objectness' found in most standard vision datasets. Of the 72k images, 2k were used as the classifier-training set, 60k were used for training ExplainGAN, 4k were used for validation and 6k were used for testing. We used the binary class pair (Apical 2-Chamber, Apical 4-Chamber).

5.2 Implementation

The model architecture implementation for E, G and D is quite similar to the DCGAN architecture [17]. We share the last few layers of E_0 and E_1 and the last few layers of D_0 and D_1. Each loss term in our objective is scaled by a coefficient whose values were obtained via cross-validation. In practice, the coefficients were quite stable across datasets (we use the same set), other than the κ hyperparameter which controls the effect of the count loss and the scaling coefficient for $\mathcal{L}_{\text{smoothness}}$, the smoothness loss.

5.3 Explanation by Qualitative Evaluation

We evaluated our model qualitatively on a number of datasets. We show results on both the Medical Ultrasound dataset and CelebA dataset in Fig. 3. The use of CelebA and a medical image dataset provides a useful contrast between images whose relationships should be quite familiar to the average reader (glasses vs no-glasses) and relationships that are likely to be foreign to the average reader (apical 2 chamber views versus apical 4 chamber views).

In each block, the ”input” column represents images $x \in S_0$, the ”transformed” column represents ExplainGAN's transformation, $G_1(z_0)$, to the opposite class. The ”mask” column illustrates the model's changes, $G_m(z_0)$, and the ”composite” column shows the composite images, $C_1(z_0)$.

The CelebA (top) results in Fig. 3 illustrates that the model's transformations for both ”glasses vs no-glasses” and ”mustache vs no-mustache” perform highly localized changes and the corresponding mask effectively produces a segmentation of the only visual feature being altered. Furthermore, the model is able to make quite minimal but perceptible changes. For example, in the first row of the ”glasses vs no-glasses” task, the mask has preserved the hair over the eyeglasses.

The Ultrasound (bottom) results in Fig. 3 illustrates that the model has both learned to model the anatomy of the heart and is able to transform from one view of the heart to the other with minimal changes. The transformations and masks clearly illustrate that the model is cuing predominantly on the presence of the right ventricle, but interestingly not the right atrium, and the shape of the pericardium.

5.4 Explanation via Pixel-Wise Attribution

Many post-hoc explanation methods that use attribution or saliency rely on visual, qualitative comparisons of attribution maps. Recently, [19] introduced a quantitative approach for comparing attribution maps in which pixels are

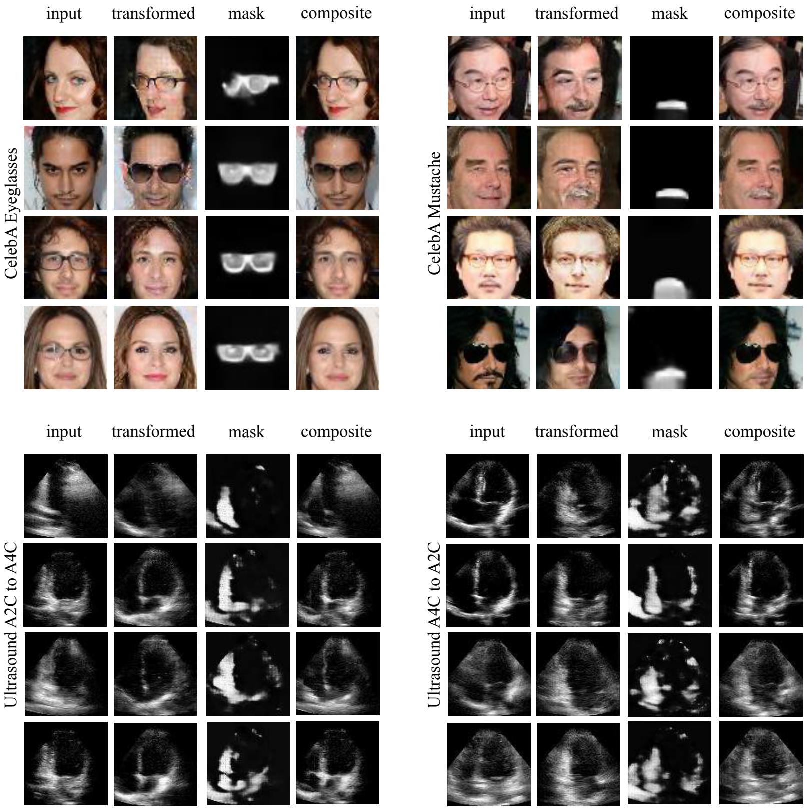


Fig. 3. Qualitative visualization of the ExplainGAN model on two datasets: CelebA and our Medical Ultrasound dataset. The "input" column represents images $x \in S_0$, the "transformed" column represents ExplainGAN's transformation, $G_1(z_0)$, to the opposite class. The "mask" column illustrates the model's changes, $G_m(z_0)$, and the "composite" column shows the composite images, $C_1(z_0)$. The results indicate that in the case of object-related transformations, such as glasses or mustaches, ExplainGAN effectively performs a weakly supervised segmentation of the object. In the ultrasound case, ExplainGAN illustrates which anatomical areas the model is cuing on: the right ventricle and pericardium.

progressively perturbed in the order of predicted saliency. Performance is judged by evaluating which methods require fewer perturbations to affect the classifier's prediction.

Our model is not designed for attribution / saliency as it produces a binary, rather than continuous mask, which is also paired to a particular transformation image. However, it is possible to loosely interpret our masks as an attribution map in which pixel priority for all pixels in the mask is not known.

While the work of [19] perturbed individual pixels, we wanted to avoid a comparison in which individual pixel changes, which are neither themselves interpretable, nor plausible as images, might alter the classification results. Consequently, we adapt the approach of [19] by perturbing the image by segments, rather than pixels. To choose the order of perturbation, we normalize the maps to the range $[0, 1]$, threshold them with $t \in [0.5, 0.7, 0.9]$ and segment the resulting binary maps. We then rank the segments based on the average map value within each segment[3]. For perturbation, we replace each pixel in each segment with uniform random noise in the range of the pixel values.

More concretely, we denote the image with k segments perturbed by $x_{\text{SP}}^{(k)}$. We compute the area over the segment perturbation curve (AOSPC) as follows:

$$\text{AOSPC} = \frac{1}{K+1} \left\langle \sum_{k=0}^{K} f(x_{\text{SP}}^{(0)}) - f(x_{\text{SP}}^{(k)}) \right\rangle_{p_x} , \tag{17}$$

where K is the number of steps, $\langle . \rangle_{p_x}$ denotes the average over all the images, and $f : \mathbb{R}^d \rightarrow \mathbb{R}$ is the classification function.

We report AOSPC after 10 steps for the explanation methods of Section 2 in Section 5.4. We choose the methods to cover the 3 main groups of methods (i.e. perturbation-based, gradient-based and activation-based). A larger AOSPC means that the sensitivity of the segments that are perturbed in 10 steps is higher. To avoid cases where the segmentation assigns all or more than half of the pixels to one segment we choose our threshold from ≥ 0.5 values. Our results demonstrate that, despite not being explicitly optimized for finding the most informative pixels, ExplainGAN performs on par with other explanation methods for classifiers. For qualitative comparison of these methods see Fig. 4.

Table 1. AOSPC value (higher is better, see Eq. (17)) after 10 steps for different segmentation thresholds. Although, ExplainGAN is not directly optimized for this metric, its performance is comparable to reasonable baselines for explanation in classifiers. A larger AOSPC means that the sensitivity of the segments that are perturbed in 10 steps is higher.

Dataset	**MNIST**			**Ultrasound**		
Threshold	**0.5**	**0.7**	**0.9**	**0.5**	**0.7**	**0.9**
Grad [22]	1474	1563	240	712	291	81
Grad-CAM [20]	17.2	8	–	–	70	**432**
Saliency [23]	817	718	126	30	63	298
Occlusion [29]	2099	1946	**1486**	**1215**	539	142
LRP [1]	1736	1478	244	700	511	71
ExplainGAN	**2622**	**2083**	1474	1167	**542**	374

[3] For ExplainGAN we take the average of the sigmoid outputs over all pixels in a segment.

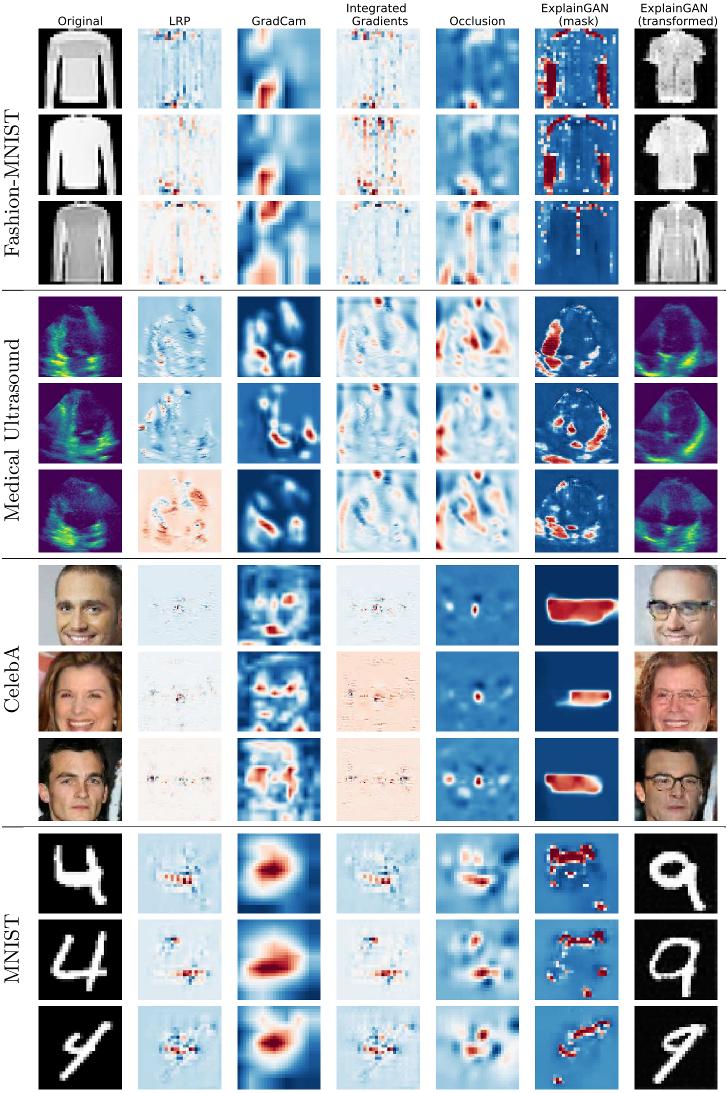


Fig. 4. Comparison of different methods for explaining the model's decision.**Fashion-MNIST**: transforming from pullover to shirt, **Ultrasound**: transforming from A2C to A4C (see Fig. 2 for examples of A2C and A4C views), **CelebA**: transforming from faces without eyeglasses to faces with eyeglasses, **MNIST**: transforming from 4 to 9.

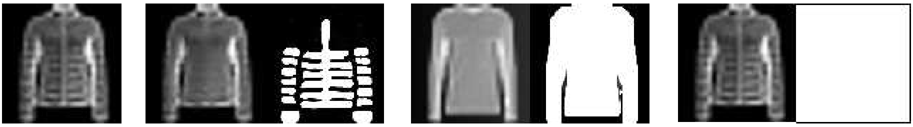
5.5 Quantitative Assessment of Explainability

Given two similar images on either side of a model's decision boundary, how can we determine quantitatively whether they provide a conceptual explanation of

Table 2. Quantitative substitutability experiments across datasets. Class 0 and Class 1 are the classes that the given classifier is trained to identify. Transformed/Composite 0/1 column shows the accuracy of the classifiers when just transformations/compositions of the images used at training time. Ceiling represents the accuracy of the base classifier on the same test set.

Dataset	Class 0	Class 1	Transformed 0	Transformed 1	Composite 0	Composite 1	Ceiling
Ultrasound	A2C	A4C	95.5	94.2	91.4	95.6	99.6
CelebA	W/O Eyeglasses	W/ Eyeglasses	93.6	96.2	96.05	96.2	96.5
CelebA	W/O Mustache	W/ Mustache	76.65	75.2	74.05	71.4	83.9
CelebA	W/O Black hair	W/ Blackhair	75.65	74.8	79.05	77.4	84.3
FMNIST	Coat	Pullover	75.8	73.7	84.8	69.1	94.1
FMNIST	Coat	Shirt	79.7	78.5	71.8	77.2	91.7
MNIST	Three	Eight	99.6	99.1	99.3	98.9	99.9
MNIST	Four	Nine	98.6	99.0	98.6	98.5	99.0
MNIST	Three	Five	98.5	99.3	98.2	98.2	99.2

why a model discriminates between them? There are several high-level criteria that must be met in order for people to find such explanatory images useful.



(a) Input

(b) Substitutable, Localized.

(c) Substitutable, Non-Localized

(d) Non-Substitutable, Non Localized

Fig. 5. Boundary-crossing images have varying explanatory power: images carry more explanatory power if they are (1) Substitutable: they can be used as substitutes in the original dataset without affecting the classifier and (2) Localized: they are different from a query image in small and easily localized ways.

Localized but not minimal: In order for the boundary-crossing image to clear demonstrate what pixels caused a label-changing event, it must deviate from the original image in a way that is localized to a clear sub-component of the image, as opposed to every pixel changing or only one or two pixels changing.

Substitutable: If we are explaining a model by comparing an original image from class A, and a boundary-crossing image is produced to appear like it came from class B, then we define *substitutability* to be the property that we can substitute our boundary-crossing image for one of the original images labeled as class B without affecting our classifier's performance.

To this end, we propose two metrics aimed at quantifying such an explanations utility. First, the degree to which changes to a query image are localized can be represented by the number of non-zero elements of the mask. Note that while other measures of locality can be used (cohesiveness, connected components), we make no such assumption as we found empirically that often such specific measures do not correlate well with conveying the set of items changing.

Second, we define the substitutability metric as follows: Let an original training set $\mathcal{D}_{\text{train}} = \{(x_i, y_i | i = 1..N\}$, a test set $\mathcal{D}_{\text{test}}$, and a classifier $\mathcal{F}(x) \rightarrow y$ whose empirical performance on the test set is some score S. Given a new set of model-generated boundary-crossing images $\mathcal{D}_{\text{trans}} = \{(x'_i, y'_i | i = 1..N\}$ we say that this set is $R\%-$substitutable if our classifier can be retrained using $\mathcal{D}_{\text{trans}}$ to achieve performance that is $R\%$ of S. For example, if our original dataset and classifier yield 90% performance, and we substitute a generated dataset for our original dataset and a re-trained classifier yields 45%, we would say the new dataset is 50% substitutable.

Table 2 illustrates the substitutability performance of our model on various datasets. These results illustrate that our model produces images that are nearly perfectly substitutable on MNIST, the Ultrasound dataset, and CelebaA for the Eyeglasses attribute. That being said, despite compelling qualitative results (Figure 4), there is still much room for improvement in terms of substitutability for the other CelebA attributes.

Table 3. Substitutability on Ultrasound Dataset. Transformed/Composite 0/1 shows the accuracy of a classifier on test set when the original samples are replaced with Transformed/Composite 0/1 at training phase. Both Transformed/Composite shows the accuracy of the classifier when all of the images are replaced with Transformed/Composite. Note that PixelDA is a oneway transformer.

	Transformed 0	Transformed 1	Both Transformed	Composite 0	Composite 1	Both composite
PixelDA	87.6	N/A	N/A	N/A	N/A	N/A
CycleGAN	94	64	84.1	N/A	N/A	N/A
ExplainGAN-norec	94.5	83.9	96.1	N/A	N/A	N/A
ExplainGAN-nomask	93.9	97.3	95.1	N/A	N/A	N/A
ExplainGAN-full	95.5	94.2	97.3	91.4	95.6	91.4
Ceiling	99.7	99.7	99.7	99.7	99.7	99.7

6 Conclusion

We introduced ExplainGAN to interpret black box classifiers by visualizing boundary-crossing transformations. These transformations are designed to be interpretable by humans and provide a high-level, conceptual intuition underlying a classifier's decisions. This style of visualization is able to overcome limitations of attribution and example-by-nearest-neighbor methods by making spatially localized changes along with visual examples. While not explicitly trained to act as a saliency map, ExplainGAN's maps are very competitive at demonstrating saliency. We also introduced a new metric, Substitutability, that evaluates how much label-capturing information is retained when performing boundary-crossing image transformations. While our method exhibits a good substitutability score, it is not perfect and we anticipate this metric being used for furthering research in this area.

References

1. Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.R., Samek, W.: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PloS one **10**(7), e0130140 (2015)
2. Bousmalis, K., Silberman, N., Dohan, D., Erhan, D., Krishnan, D.: Unsupervised pixel-level domain adaptation with generative adversarial networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). vol. 1, p. 7 (2017)
3. Fong, R.C., Vedaldi, A.: Interpretable explanations of black boxes by meaningful perturbation. arXiv preprint arXiv:1704.03296 (2017)
4. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.: Domain-adversarial training of neural networks. The Journal of Machine Learning Research **17**(1), 2096–2030 (2016)
5. Gatys, L.A., Ecker, A.S., Bethge, M.: A neural algorithm of artistic style. arXiv preprint arXiv:1508.06576 (2015)
6. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on. pp. 2414–2423. IEEE (2016)
7. Goodfellow, I.J., Shlens, J., Szegedy, C.: Explaining and harnessing adversarial examples. arXiv preprint arXiv:1412.6572 (2014)
8. Goodman, B., Flaxman, S.: European union regulations on algorithmic decision-making and a" right to explanation". arXiv preprint arXiv:1606.08813 (2016)
9. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European Conference on Computer Vision. pp. 694–711. Springer (2016)
10. Larsen, A.B.L., Sønderby, S.K., Larochelle, H., Winther, O.: Autoencoding beyond pixels using a learned similarity metric. arXiv preprint arXiv:1512.09300 (2015)
11. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE **86**(11), 2278–2324 (1998)
12. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Advances in Neural Information Processing Systems. pp. 700–708 (2017)
13. Liu, M.Y., Tuzel, O.: Coupled generative adversarial networks. In: Advances in neural information processing systems. pp. 469–477 (2016)
14. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: Proceedings of International Conference on Computer Vision (ICCV) (2015)
15. Maaten, L.v.d., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(Nov), 2579–2605 (2008)
16. Montavon, G., Lapuschkin, S., Binder, A., Samek, W., Müller, K.R.: Explaining nonlinear classification decisions with deep taylor decomposition. Pattern Recognition **65**, 211–222 (2017)
17. Radford, A., Metz, L., Chintala, S.: Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv preprint arXiv:1511.06434 (2015)
18. Rudin, L.I., Osher, S., Fatemi, E.: Nonlinear total variation based noise removal algorithms. Physica D: nonlinear phenomena **60**(1-4), 259–268 (1992)
19. Samek, W., Binder, A., Montavon, G., Lapuschkin, S., Müller, K.R.: Evaluating the visualization of what a deep neural network has learned. IEEE transactions on neural networks and learning systems **28**(11), 2660–2673 (2017)

20. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. See https://arxiv. org/abs/1610.02391 v3 **7**(8) (2016)
21. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. arXiv preprint arXiv:1704.02685 (2017)
22. Shrikumar, A., Greenside, P., Shcherbina, A., Kundaje, A.: Not just a black box: Learning important features through propagating activation differences. arXiv preprint arXiv:1605.01713 (2016)
23. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: visualising image classification models and saliency maps (2014). arXiv preprint arXiv:1312.6034 (2013)
24. Springenberg, J.T., Dosovitskiy, A., Brox, T., Riedmiller, M.: Striving for simplicity: The all convolutional net. arXiv preprint arXiv:1412.6806 (2014)
25. Sundararajan, M., Taly, A., Yan, Q.: Axiomatic attribution for deep networks. arXiv preprint arXiv:1703.01365 (2017)
26. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199 (2013)
27. Wattenberg, M., Vigas, F., Johnson, I.: How to use t-sne effectively. Distill (2016). https://doi.org/10.23915/distill.00002, `http://distill.pub/2016/misread-tsne`
28. Xiao, H., Rasul, K., Vollgraf, R.: Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms (2017)
29. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: European conference on computer vision. pp. 818–833. Springer (2014)
30. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. arXiv preprint arXiv:1703.10593 (2017)