When adversarial attacks become interpretable counterfactual explanations

Mathieu Serrurier, Franck Mamalet, Thomas Fel, Louis Béthune, Thibaut Boissin

When adversarial attacks become interpretable counterfactual explanations

Mathieu Serrurier
Université Paul-Sabatier, IRIT
Toulouse, France

Franck Mamalet
IRT Saint-Exupéry
Toulouse, France

Thomas Fel
IRT Saint-Exupéry
Toulouse, France

Louis Béthune
Université Paul-Sabatier, IRIT
Toulouse, France

Thibaut Boissin,
IRT Saint-Exupéry
Toulouse, France

Abstract

We argue that, when learning a 1-Lipschitz neural network with the dual loss of an optimal transportation problem, the gradient of the model is both the direction of the transportation plan and the direction to the closest adversarial attack. Traveling along the gradient to the decision boundary is no more an adversarial attack but becomes a counterfactual explanation, explicitly transporting from one class to the other. Through extensive experiments on XAI metrics, we find that the simple saliency map method, applied on such networks, becomes a reliable explanation, and outperforms the state-of-the-art explanation approaches on unconstrained models. The proposed networks were already known to be certifiably robust, and we prove that they are also explainable with a fast and simple method.

1 Introduction

In classification, a counterfactual explanation exhibits why the decision was A and not B. When dealing with symbolic models, this modification may be significant and expresses causality between the feature values and the class [32]. Unfortunately, in the deep learning settings, a counterfactual corresponds to an adversarial attack [37]. The idea behind these attacks is that only carefully chosen small modifications, such as an imperceptible noise, are necessary to change the class of an example and to fool the network. Thus, this counterfactual usually does not provide a trustworthy explanation [56]. Since saliency maps [47] – gradient of output with respect to the input – are the basis of most adversarial attacks, they are generally unsuitable for explaining the model decision. Several methods which require more complex computations, such as SmoothGrad [49], Integrated Gradient [51] or Grad-CAM [45], have therefore been proposed to provide better explanations. Recently, the XAI community has started to investigate the link between explainability and robustness and proposed methods and metrics accordingly [28, 10, 35, 43].

In [46], authors propose to cope with the weakness with respect to adversarial attacks by training 1-Lipschitz constrained neural networks with a loss that is the dual of an optimal transport optimization problem, called hKR and noted $\mathcal{L}^{hKR}_{\lambda,m}$. The models obtained have been proven to be robust with a certifiable margin. In the following, we denote these networks as Optimal Transport Neural Networks (OTNN).

In this paper, we show that OTNNs also have very valuable properties in terms of explainability. Indeed, an optimal transport plan between two classes can be viewed as a global way to build counterfactuals [14]. These counterfactuals no longer correspond systematically to the smallest transformation for a given input sample, but to the smallest in average when pairing points of two

Preprint. Under review.

classes. OTNNs encode the dual formulation of the optimal transport problem, and we prove that their gradients on a given point x is both (i) in the direction of the closest adversarial example on the decision boundary and (ii) in the direction of images of x according to the underlying transport plan. This means that building an adversarial attack for an OTNN is equivalent to travelling along the optimal transport path. Consequently, the modification obtained is not only an adversarial attack, but a counterfactual explanation, i.e. why the classification was not another class. Fig. 1 illustrates, on an OTNN learned on MNIST dataset, the transformation of an image of the class zero with respect to the gradient of another class's output (as done in a vanilla targeted attack). We observe that it explicitly changes the zero into the target number, and gradients provide understandable explanations of why it was not this number. The consequence of this property is that the saliency map of OTNN for an image gives the importance of each pixel in the modification required to change class, and is thus a trustworthy explanation. Note that several methods based on GAN [29] or on causality penalty [30] achieve very realistic counterfactual images. In this paper, we don't try to compete with the quality of these results, but to show that OTNNs have in-built counterfactual explanations.

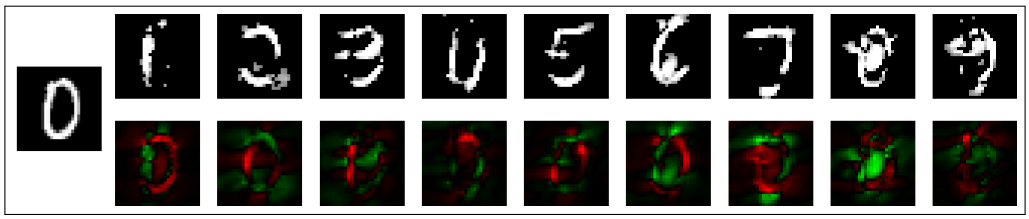


Figure 1: Counterfactual targeted samples of the form $\mathbf{x} - 10 * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$ for an OTNN multiclass classifier, learned on MNIST, on a sample x of the class 0. The second line is the targeted gradient (negative values are in red and positive values in green).

We summarize our contributions: first, after introducing the background about OTNN and XAI, we demonstrate several properties of the gradient of an OTNN with respect to adversarial attack, the decision boundary and optimal transport. Second, we link the optimal transport to counterfactual explanations, and we claim that saliency maps for OTNNs have valuable properties that make them trustworthy explanations. Third, we propose a way to automatically tune the optimal transport loss parameters and propose enhancements for the multiclass version of the hKR loss proposed in [46], leading to higher performances. We also show in the experiments that saliency maps for OTNN have top-rank scores on the state-of-the-art XAI metrics compared to more sophisticated methods, and are equivalent to the ones provided by Smoothgrad. Thus Saliency maps provide faithful, stable and trustworthy explanations for a minimal computational cost. We also find that OTNNs significantly enhance metric scores of most of the XAI methods, in comparison to their use on unconstrained neural networks. To end with we present several samples of gradient-based counterfactual obtained with OTNNs.

2 Related work

1-Lipschitz Neural network and optimal transport. For sake of simplicity, we consider, in Section 3, binary classification problems on feature vector space $X \subset \Omega$ and labels $Y = \{-1, 1\}$. We name $P_+ = \mathbb{P}(X|Y = 1)$ and $P_- = \mathbb{P}(X|Y = -1)$, the conditional distributions with respect to Y. We note $p = P(Y = 1)$ and $1 - p = P(Y = -1)$ the apriori class distribution.
A function $f : \Omega \to \mathbb{R}$ is a 1-Lipschitz functions over Ω (denoted $Lip_1(\Omega)$) if and only if $\forall \mathbf{x}, \mathbf{y} \in \Omega^2, ||f(\mathbf{x}) - f(\mathbf{y})|| \leq ||\mathbf{x} - \mathbf{y}||$. 1-Lipschitz neural networks have received a lot of attention, especially due to the link with adversarial attacks. They provide certifiable robustness guarantees [25, 39], improve the generalizations [50] and the interpretability of the model [53]. The simplest way to constrain a network to be in $Lip_1(\Omega)$ is to impose this 1-Lipschitz property to each layer. Frobenius normalization [44], or spectral normalization [38] can be used for linear layers, and can also be extended, in some situation, to orthogonalization [34, 1].

Optimal transport, 1-Lipschitz neural networks and binary classification have been first associated in Wasserstein GAN (WGAN [6]). Indeed, the discriminator of a WGAN is the solution to the Kantorovich-Rubinstein dual formulation of the 1-Wasserstein distance [55]. It could be viewed

as a binary classifier given a carefully chosen threshold. However, [46] has shown that this kind of classifier is sub-optimal, even on a toy dataset. In the same paper, the authors cope with the sub-optimality of the Wasserstein classifier by proposing the hKR loss $\mathcal{L}^{hKR}$ which adds a hinge regularization term to the Kantorovich-Rubinstein optimization goal :

$$\mathcal{L}^{hKR}_{\lambda,m}(f) = \mathop{\mathbb{E}}_{\mathbf{x}\sim P_-}[f(\mathbf{x})] - \mathop{\mathbb{E}}_{\mathbf{x}\sim P_+}[f(\mathbf{x})] + \lambda \mathop{\mathbb{E}}_{\mathbf{x}}(m - Yf(\mathbf{x}))_+ \tag{1}$$

where $m > 0$ is the margin. We note f^* the optimal minimizer of $\mathcal{L}^{hKR}_{\lambda,m}$. The classification is given by the sign of f^*. In the following, the 1-Lipschitz neural networks that minimize $\mathcal{L}^{hKR}_{\lambda,m}$ will be denoted as OTNN. Given a function f, a classifier based on $sign(f)$ and an example x, an adversarial example is defined as follows:

$$adv(f, \mathbf{x}) = \underset{\mathbf{z}\in\Omega | sign(f(\mathbf{z}))=-sign(f(\mathbf{x}))}{argmin} \| \mathbf{x} - \mathbf{z} \| . \tag{2}$$

Since f^* is a 1-Lipschitz function, $|f^*(\mathbf{x})|$ is a certifiable lower bound of the robustness of the classification of $\mathbf{x}$ (i.e. $\forall \mathbf{x}, |f^*(\mathbf{x})| \leq ||\mathbf{x} - adv(f^*, \mathbf{x})||$). The function f^* has the following properties [46] (i) if $P_+ =$ and $P_- =$ are separable with a minimal distance of $\epsilon > 0$, then for $m < 2\epsilon$, f^* achieves 100% accuracy on P_+ and P_-; (ii) minimizing $\mathcal{L}^{hKR}$ is still the dual formulation of an optimal transport problem (see appendix for more details).

Explainability and metrics. Attribution methods aim to explain the prediction of a deep neural network by pointing out input variables that support the prediction – typically pixels or image regions for images – which lead to importance maps. Saliency [47] was the first proposed white-box attribution method and consists of back-propagating the gradient from the output to the input. The resulting absolute gradient heatmap indicates which pixels affect the most the decision score. However, this family of methods suffers from problems inherent to the gradients of standard models. Methods such as Integrated Gradient [51] and SmoothGrad [49] partially address this issue by accumulating gradients, either along a straight interpolation path from a baseline state to the original image or from a set of points close to the original image obtained after adding noise but multiply the computational cost by 100. These methods were then followed by a plethora of other methods using gradients such as Grad-cam [45], Input Gradient [4], ... all relying on gradient calculation of the classification method. Finally, other methods – sometimes called black-box attribution methods – do not involve the gradient and rely on perturbations around the image to generate their explanations [41, 15].

However, it is becoming increasingly clear that current methods raise many issues [2, 26, 48] such as confirmation bias: it is not because the explanations make sense to humans that they reflect the evidence of the prediction. To address this challenge, a large number of metrics were proposed to provide objective evaluations of the quality of explanations. Deletion and Insertion methods [41] evaluate the drop in accuracy when important pixels are replaced by a baseline. μFidelity method [8] evaluates the correlation between the sum of importance scores of pixels and the drop of the score when removing these pixels. In parallel, a growing literature relies on model robustness to derive new desiderata for a good explanation [28, 10, 35, 43, 16]. The central idea is that a region is considered important if it allows to easily generate an adversarial example. In addition, [28] showed that some of these metrics also suffer from a bias due to the choice of the baseline value and proposed a new metric called Robustness-Sr. This metric assesses the ease to generate adversarial example when the attack is limited to the important variables proposed by the explanation. Finally, other metrics propose to assess other properties such as generalizability, consistency [17], or stability [60, 8] of explanation methods.

These works on explainability metrics have initiated the emergence of links between the robustness of models and the quality of their explanations [12, 58]. In particular,[17] claimed that 1-Lipschitz networks explanations have better metrics scores. But this study was not on OTNNs and was limited to their proposed metrics.

To end with, several counterfactual explanation [56] methods, providing information on "why the decision was A and B", have been proposed [23, 42, 57], but rely on complex models.

3 Optimal Transport, Robustness and explainability

In this section we extend the properties of the OTNNs to the explainability framework, all the proofs are in the appendix A.1. We note π the optimal transport plan corresponding to the minimizer of

$\mathcal{L}^{hKR}_{\lambda,m}$. Given $\mathbf{x} \in P_+$ (resp. P_-) we note $\mathbf{y} = tr_\pi(\mathbf{x}) \in P_-$ (resp. P_+) the image of $\mathbf{x}$ with respect to π. Since the π is not deterministic, we take $tr_\pi(\mathbf{x})$ as the point of maximal mass with respect to π.

Proposition 1 (Transportation plan direction) *Let f^* an optimal solution minimizing the $\mathcal{L}^{hKR}_{\lambda,m}$. Given $\boldsymbol{x} \in P_+$ (resp. P_-) and $\boldsymbol{y} = tr_\pi(\boldsymbol{x})$, then $\exists t \geq 0$ (resp. $t \leq 0$) such that $\boldsymbol{y} = \boldsymbol{x} - t.\nabla_x f^*(\boldsymbol{x})$ almost surely.*

This proposition is also true for Kantorovich-Rubinstein dual problem without hinge regularization. It proves that for most of $x \in P_+ \cup P_-$ the gradient $\nabla_x f^*(x)$ represents the direction in the transportation plan.

Proposition 2 (Decision boundary) *Let P_+ and P_- two separable distributions with minimal distance ϵ and f^* an optimal solution minimizing the $\mathcal{L}^{hKR}_{\lambda,m}$ with $m < 2\epsilon$. Given $\boldsymbol{x} \in P_+ \cup P_-$ and $\boldsymbol{y} = tr_\pi(\boldsymbol{x}) \in \{\boldsymbol{x} - t\nabla_x f^*(\boldsymbol{x})\}$, then $|t| \geq |f^*(x)|$ and $x_\delta = \boldsymbol{x} - f^*(\boldsymbol{x}).\nabla_{\boldsymbol{x}} f^*(\boldsymbol{x}) \in \delta f^*$ where $\delta f^* = \{x' \in \Omega | f^*(x') = 0\}$ is the decision boundary (i.e. the 0 level set of f^*)*

Experiments suggest this probably remains true when P_+ and P_- are not separable. Prop. 2 proves that an OTNN f learnt by minimizing the $\mathcal{L}^{hKR}_{\lambda,m}$, $|f(\mathbf{x})|$ provides a tight robustness certificate.

Corollary 1 *Let P_+ and P_- two separable distributions with minimal distance ϵ and f^* an optimal solution minimizing the $\mathcal{L}^{hKR}_{\lambda,m}$ with $m < 2\epsilon$, given $x \in P_+ \cup P_-$,*

$$adv(f^*, \boldsymbol{x}) = x_\delta$$

almost surely where $x_\delta = \boldsymbol{x} - f^*(\boldsymbol{x}).\nabla_{\boldsymbol{x}} f^*(\boldsymbol{x})$.

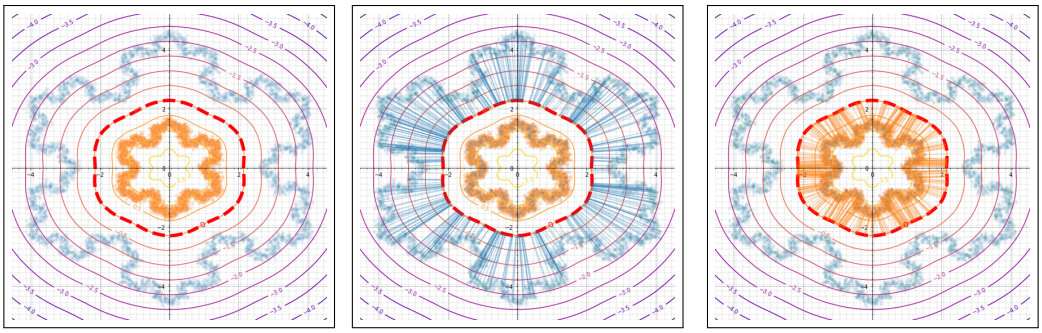
This corollary shows that adversarial examples are precisely known for the classifier based on $\mathcal{L}^{hKR}_{\lambda,m}$. In this case, optimal adversarial attacks are in the direction of the gradient (i.e. FGSM attack [22]). This corroborates the observations in [46] where all the attacks, such as PGD [37] or Carlini and Wagner [11] ones, applied on an OTNN model were equivalent to FGSM ones.

To illustrate these propositions, we learnt a dense binary classifier with $\mathcal{L}^{hKR}_{\lambda,m}$ to separate two complex distribution, following two concentric Koch snowflakes. Fig.2-a shows the two distribution (blue and orange snowflakes), the learnt boundary ($0 - levelset$) (red dashed line). Fig.2-b,c show for random samples $\mathbf{x}$ from the two distributions, the segments $[\mathbf{x}, \mathbf{x}_\delta]$ where $\mathbf{x}_\delta$ is defined in 2 . As expected by Prop. 2, $\mathbf{x}_\delta$ points fall exactly on the decision boundary. Besides, as stated in Prop. 1 each segment provides the direction of the image with respect to the transport plan.

Finally, we showed that with OTNN, adversarial attacks are formally known and simple to compute. Furthermore, since we proved that these attacks are along the transportation map, they are no more an imperceptible modification but an understandable transformation of the example. In the following, we will take advantage of these properties to show that $\nabla_x f^*(\mathbf{x})$ provides a natural counterfactual explanation with provable explainability properties.

As pointed out in the introduction, a counterfactual explanation for a given $\mathbf{x}$ of class P_+ is the closest element $\mathbf{y} \in P_-$. But we usually don't have access to P_+ and P_-, only to a classifier f. In this case, a counterfactual corresponds to an adversarial attack as defined in 2. For classical neural networks, this can be done by only adding noise which is not a valuable explanation. As it only depends on $\mathbf{x}$ and f, this definition of counterfactual explanation is **local**. On the contrary, a transport plan as the one underlying the minimizer of $\mathcal{L}^{hKR}_{\lambda,m}$describes an optimal way to go from the class P_+ to P_-. As such, the transportation plan is a **global** counterfactual explanation, and $\nabla_{\mathbf{x}} f^*(\mathbf{x})$ is the local explanation for $\mathbf{x}$. Note that, the transportation plan doesn't provide the closest example on the opposite class, but provides the closest in average on the pairing process. According to Prop. 1, the image of $\mathbf{x}$ in the optimal transport plan is $\mathbf{y} = \mathbf{x} + t\nabla_{\mathbf{x}} f^*(\mathbf{x})$. Even if t is only partially known, using $t = f^*(\mathbf{x})$, we know that $\mathbf{y}$ is on the decision boundary and is both an adversarial attack and a counterfactual explanation and $|t| \geq |f^*(\mathbf{x})|$ is on the path to the optimal transport plan.

As stated in Section 2, Saliency maps [47], given by $\phi_x(i) = |\frac{\partial f_l}{x_i}|$ often lead to blurry explanation on classical networks. In this paper, we claim that for OTNNs saliency maps lead to **trustworthy explanations**. Indeed, we have shown in the previous section that, for an OTNN, $\nabla_x f^*(x)$ indicates both the direction in the transportation plan and also to the closest point on the boundary δf^*.



(a) (b) (c)

Figure 2: Level sets of an OTNN $\hat{f}$ classifier for two concentric Koch snowflake (a). The decision boundary (0-level set) is the red dashed line. Figure (b) (resp. (c)) represents translation of the form $\mathbf{x} - \hat{f}(\mathbf{x})\nabla_x \hat{f}(\mathbf{x})$ of each point $\mathbf{x}$ of the first class (resp second class). The movement is represented by a line between the initial point and its transformation (which are all on the decision boundary).

Thus, the Saliency map $\phi_x(i) = |\frac{\partial f_l}{x_i}|$ represents the importance of each input feature along this direction. We will show in the experiments that, for the Saliency map explanation: (i) metrics scores are higher or comparable to other explanation methods (which is not the case for unconstrained networks), thus it has higher ranks; (ii) distance to other attribution methods such as Smoothgrad is unnoticeable/imperceptible; (iii) scores obtained on metrics that can be compared between networks are higher than those obtained with unconstrained networks.

4 Automatic margin and multiclass loss

In this section, we put aside the explainability to focus on hKR loss. As pointed out in [7], one drawback of working with 1-Lipschitz functions is that it depends strongly on the parameters of the loss. In the binary case, $\mathcal{L}^{hKR}_{\lambda,m}$ (equation 1) has two parameters : the margin m and the hinge weight λ. λ represents the tradeoff between robustness and accuracy. When the classes are separable and the m is small enough, the hinge part of the loss tends to zero. Since, the parameter m is hard to choose, to we propose a new formulation of the loss as follows:

$$\mathcal{L}^{hKR}_{\lambda,\alpha}(f) = \mathop{\mathbb{E}}_{\mathbf{x}\sim P_-}[f(\mathbf{x})] - \mathop{\mathbb{E}}_{\mathbf{x}\sim P_+}[f(\mathbf{x}] + \lambda\left(\mathop{\mathbb{E}}_{\mathbf{x}}(m - Yf(\mathbf{x}))_+ + \alpha m\right) \tag{3}$$

with $m > 0$ a learnable parameter, and $0 < \alpha \leq 1$ is a new parameter. It is easy to see that, according to the linear growth of αm and its opposite hinge term, if $f(\mathbf{x})$ is uniformly distributed on a bounded interval, the optimal margin m is obtained when the ratio of $\mathbf{x}$ such that $f(\mathbf{x}) \leq m$ is equal to α. The latter can be interpreted as the target proportion of data that is concerned by the hinge part of the loss. By choosing $\lambda \approx \frac{1}{\alpha}$, weight of the KR part in the loss will be approximately the same as the hinge part at the end of the optimizing process. With this approach, the only parameter to choose is α that can be interpreted as the approximated error rate targeted in the learning process.

An extension has also been proposed in [46] to the multiclass case with q classes. The idea is to learn q 1-Lipschitz functions $f_1, \ldots, f_q$, each component f_i being a *one-versus-all* binary classifier. The loss proposed was the following

$$\mathcal{L}^{hKR}_{\lambda}(f_1, \ldots, f_q) = \sum_{k=1}^{q}\left[\mathop{\mathbb{E}}_{\mathbf{x}\sim\neg P_k}[f_k(\mathbf{x})] - \mathop{\mathbb{E}}_{\mathbf{x}\sim P_k}[f_k(\mathbf{x})]\right] + \lambda \mathop{\mathbb{E}}_{\mathbf{x},\mathbf{y}\sim\bigcup_{k=1}^{q} P_k}(H\left(f_1(\mathbf{x}), \ldots, f_q(\mathbf{x}), \mathbf{y}\right) \tag{4}$$

with :

$$H\left(f_1(\mathbf{x}), \ldots, f_q(\mathbf{x}), \mathbf{y}\right) = \sum_{k=1}^{q} m - (2 * \mathbb{1}_{y=k} - 1) * f_k(\mathbf{x})$$

This formulation has three main drawbacks: (i) the optimal margin for each class may be different leading to a huge number of hyperparameters - solved by Eq.3-, (ii) for large number of classes

several outputs may have few or no positive sample within a batch leading to slow convergence, (iii) weight of $f_y(\mathbf{x})$ (the function of the true class) with respect to the other decreases when the number of classes increases. To overcome these drawbacks, we propose a softmax based hinge regularization with a learnable margin :

$$H^{\alpha}_{soft}\left(f_1(\mathbf{x}),\ldots,f_q(\mathbf{x}),\mathbf{y}\right) = m_y - \frac{1}{2}f_y(\mathbf{x}) + \frac{1}{2}\sum_{k\neq y} f_k(\mathbf{x}) * \frac{e^{f_k(\mathbf{x})}}{\sum_{j\neq y} e^{f_j(\mathbf{x})}} + \alpha m_y \tag{5}$$

with $m_y \geq 0$ and in this function, the value of $f_{\mathbf{y}}(\mathbf{x})$ for the true class always has the same weight as the value of the other functions, no matter the number of classes. At the beginning of the learning, the softmax acts like an average since all the values of f_k are close. During the learning process, the values of f_k diverge and the softmax acts like a maximum. Margins are automatically learnt as for the binary case (α a single hyperparameter).

5 Experiments

We conduct experiments networks learnt on FashionMNIST [59], and 22 binary labels of CelebA [36] datasets. Note that labels in CelebA are very unbalanced (see Table 5 in Appendix A.3, with for instance less than 5% samples for *Mustache* or *Wearing_Hat*).

A VGG-like architecture [40] is used, with equivalent linear layer sizes for OTNNs and unconstrained networks (same number of layers and neurons). Unconstrained networks use batchnorm and ReLU layers for activation, whereas OTNNs only use GroupSort2 [5, 46] activation. OTNNs are built using the *DEEL.LIP*[1] library. The loss functions are cross-entropy for unconstrained networks (categorical for multiclass, and sigmoid for multilabel settings), and hKR $\mathcal{L}^{hKR}_{\lambda,m}$ (and the proposed variants) for OTNNs. We train all networks with ADAM optimizer [31]. Details on architectures and parameters are given in Appendix A.2.

Classification performance: OTNN models achieve comparable results to unconstrained ones, confirming claims of [7]: they reach 88.5% average accuracy on FashionMNIST (Table 8), and 81% (resp. 82%) average Sensitivity (resp. Specificity) over labels on CelebA (Table 9 in Appendix A.3). We use Sensitivity and Specificity for CelebA to take into consideration the unbalanced labels.

5.1 Quantitative evaluation of explanations metrics

In this section, we present the results of quantitative evaluations of XAI metrics to compare the Saliency map method with others explanation methods on OTNN, and more generally compare XAI explanations methods on these networks and on the unconstrained counterparts. On CelebA, we only present the results for the label *Mustache*, but results for the other labels are similar. Parameters for explanation methods and metrics are given in Appendix A.4.

5.1.1 Saliency maps on OTNN become reliable explanations

Insertion and Deletion metrics: We first assess the quality of Saliency map explanations for the proposed network using Insertion and Deletion metrics [41]. Classical explanation methods, including the Saliency map, are evaluated on CelebA and FashionMNIST datasets for both types of networks. Even if the score values cannot be compared between different networks, Table 1 shows the Saliency map method becomes competitive on these metrics and matches the top-ranking methods, whereas it is not the case for unconstrained networks. In Annex A.4, we show that it is also the case for other metrics such as Robustness-SR [27].

Saliency map method on OTNN is equivalent to SmoothGrad: In the previous section, scores of Smoothgrad and Saliency were very close. We will prove that these methods are in fact equivalent, meaning that for OTNNs averaging over a large set of noisy inputs, as in SmoothGrad, is useless.

For this, we evaluated two distances in Table 2: L_2 distance indicating the per pixel difference between explanations, and $1-\rho$ where ρ is the Spearman's rank correlation coefficient (as suggested by [2, 17, 52, 21]). OTNN explanation distances are far lower than the unconstrained ones and very close to zero. Fig. 3 also illustrates this equivalence.

[1] `https://github.com/deel-ai/deel-lip` distributed under MIT License (MIT)

Table 1: Insertion and Deletion metrics evaluation; GC: GradCam, GI: Gradient $\odot$ Input, IG: Integrated Gradient, Saliency Rk : Rank (comparison by line only : in bold best score)

Dataset	Network	Deletion (uniform baseline) ($\downarrow$ is better)					
		GC	GI	IG	Rise	Saliency	SmoothGrad
CelebA	OTNN	9.41	9.38	9.36	8.89	8.32 (Rk2)	**8.28**
	Unconstrained	5.13	3.33	**3.18**	3.61	3.50 (Rk4)	3.41
Fashion-MNIST	OTNN	0.23	0.28	0.28	0.24	0.22 (Rk2)	**0.21**
	Unconstrained	0.32	0.35	0.39	0.32	0.26 (Rk2)	**0.24**
		Insertion (uniform baseline) ($\uparrow$ is better)					
CelebA	OTNN	10.25	10.44	10.50	10.46	**11.29** (Rk1)	11.28
	Unconstrained	9.43	8.45	9.24	**12.20**	6.71 (Rk6)	6.72
Fashion-MNIST	OTNN	**0.28**	0.20	0.20	0.26	0.25 (Rk4)	0.26
	Unconstrained	**0.44**	0.32	0.26	0.40	0.30 (Rk4)	0.24

Table 2: Distance between Saliency map and SmoothGrad explanations ($\downarrow$ is better)

Dataset	Network	Distance Saliency/SmoothGrad	
		L_2	$1-\rho$
CelebA	OTNN	**3.1E-04**	**4.6E-02**
	Unconstrained	1.4E-01	6.2E-01
Fashion-MNIST	OTNN	**7.5E-03**	**3.0E-01**
	Unconstrained	07.0E-02	9.1E-01

Saliency Map on OTNN are less complex: Inspired by previous works [13] highlighting a strong correlation between Kolmogorov complexity and human evaluation of complexity [20, 19]. We used a JPEG based compressor [54] as a simple approximation of visual explanation complexity [33]: OTNNs yield simpler explanations ($9.5kB$) than unconstrained networks ($16.8kB$) -see Figure 3 for a qualitative comparison-.

5.1.2 OTNNs provide better explanations

In [17], it has been shown that 1-Lipschitz neural networks, for the two proposed metrics, produce explanations with higher scores than common neural networks. In this section, we will assess this property using SoTA XAI metrics.

μFidelity metric [8] is a well-known method that measures the correlation between important variables defined by the explanation method and the model score decrease when these variables are reset to a baseline state (or replaced by uniform noise). One interesting property of this metric, as a

(a) OTNN (b) Unconstrained

Figure 3: Comparison of Saliency map and SmoothGrad explanations for (a) OTNN and (b) unconstrained network for the *Mustache* label.

Table 3: μFidelity metrics evaluation ($\uparrow$ is better); GC: GradCam, GI: Gradient.Input, IG: Integrated Gradient (in bold best model score)

Dataset	Network	μFidelity-Uniform					
		GC	GI	IG	Rise	Saliency	SmoothGrad
CelebA	OTNN	**0.028**	**0.168**	**0.149**	**0.114**	**0.244**	**0.248**
	Unconstrained	0.002	0.074	0.093	0.051	0.052	0.018
Fashion-	OTNN	**0.215**	-0.017	**-0.005**	**0.220**	**0.114**	**0.156**
MNIST	Unconstrained	0.008	**-0.009**	-0.013	0.011	-0.001	-0.001
		μFidelity-Zero					
CelebA	OTNN	**0.127**	**0.439**	**0.400**	**0.350**	**0.325**	**0.324**
	Unconstrained	0.061	0.093	0.124	0.190	0.082	0.091
Fashion-	OTNN	**0.161**	**0.479**	**0.543**	**0.182**	**0.246**	**0.332**
MNIST	Unconstrained	0.046	0.079	0.134	0.063	0.034	0.052

Table 4: Stability metrics evaluation ($\downarrow$ is better); IG: Integrated Gradient

Dataset	Network	Stability L_2		
		IG	Saliency	SmoothGrad
CelebA	OTNN	**1.7E-08**	**1.2E-07**	**7.7E-08**
	Unconstrained	1.4E-02	5.3E-02	1.4E-04
Fashion-	OTNN	**1.5E-05**	**6.3E-05**	**1.5E-05**
MNIST	Unconstrained	3.7E-03	1.6E-01	1.1E-03
		Stability Spearman rank		
CelebA	OTNN	**0.52**	**0.51**	**0.52**
	Unconstrained	0.87	0.77	0.95
Fashion-	OTNN	**0.61**	**0.60**	**0.55**
MNIST	Unconstrained	0.79	0.91	0.82

correlation score, is that it can be compared between different networks. Table. 3 clearly state that whatever the explanation method, the μFidelity score is higher when it is applied on OTNN.

Explanations on OTNN have higher stability An important property for explanations is their stability for nearby samples. In [60], the authors proposed Stability metrics based on the L_2 distance. To better evaluate this stability, one can replace the L_2 distance by $1 - \rho$, ρ being the Spearman rank correlation, as above. In Table 4 we find once more that OTNNs outperform unconstrained ones.

We conclude with all these experiments, using many types of explanation metrics, that OTNN explainability is better than the unconstrained neural networks. Besides, for OTNN the simple Saliency map method is enough. We show, in the following, qualitatively how gradient provides counterfactual explanations.

5.2 Qualitative results

CelebA: Using the learnt OTNN on multilabels, Fig. 4 presents original images, average gradients $\nabla_x f_j$ over the channels, and images in the direction of the transport plan (Prop. 1), for several negative and positive samples of different labels (other samples are given in Appendix A.5). We can see that most of the gradients are visually consistent, adding/erasing hat or mustache, opening/closing mouth, even with a very unbalanced training set.

FashionMNIST: The same illustration is given in Fig. 1 on this multiclass problem, where the gradient are targeted to explain, for instance, why the decision was a trouser and not a dress . The gradient, i.e. Saliency maps of OTNN are counterfactual explanations (See Appendix A.5 for other examples).

More generally, we observe that the gradient gives clear information about how the classifier makes its decision. For instance, for the hat, it shows that the classifier does not need to encode perfectly the concept of hat, but mainly to identify a large darker area on the top of the head.

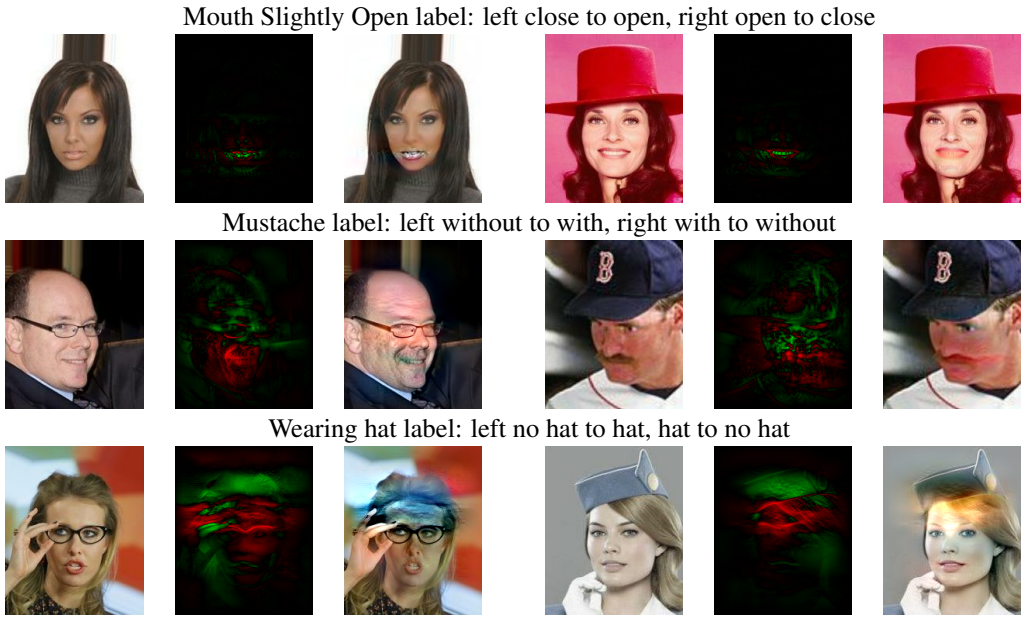


Figure 4: Samples from different labels of CelebA: (left) source image , (center) gradient image, (right) counterfactual of the form $\mathbf{x} - t * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$, for $t > 1$

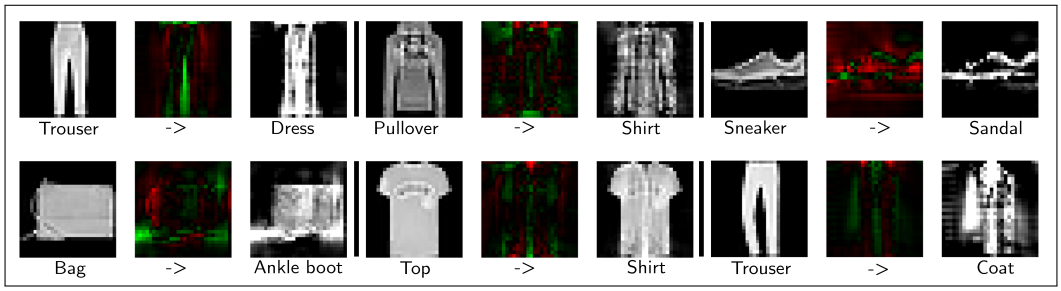


Figure 5: Samples from different classes of FashionMNIST: (left) source image , (center) targeted gradient image of an OTNN, (right) targeted counterfactual of the form $\mathbf{x} - 10 * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$

6 Conclusions and broader impact

In this paper, we study OTNN (Optimal Transport Neural Networks) that are 1-Lipschitz constrained neural networks trained with a loss that is the dual of an optimal transport optimization problem. We first provide enhancements of the $\mathcal{L}^{hKR}_{\lambda,m}$, proposed in[46], reducing the number of hyperparameters and the improving the performances at convergence. We prove that OTNNs, because of their connection with optimal transport, structurally produce counterfactual explanations. Indeed, we prove that the gradient of an OTNN at a point represents the direction of the adversarial attack but also of its image in the optimal transport plan, transforming the adversary attacks into an understandable counterfactual explanation. This is illustrated in the experiment which shows that the simple Saliency map for OTTNs has top-rank scores on state-of-the-art XAI metrics, and largely outperforms any method applied to unconstrained networks. In future works, we will investigate the link with Fairness, for instance if the explanations can point out sensitive variables.

Broader impact. This paper demonstrates the value of OTNNs for critical problems. OTNNs are certifiably robust and explainable with the simple Saliency map method and have accuracy performances comparable to unconstrained networks. On the other hand, even if the learning process of these networks is between 3 and 6 times longer than for unconstrained ones, at the inference time, OTNNs are classical networks with the same computation cost as their unconstrained counterparts. We hope that this contribution will raise a great interest for these OTNN networks.

Acknowledgments and Disclosure of Funding

This work has benefited from the AI Interdisciplinary Institute ANITI, which is funded by the French "Investing for the Future – PIA3" program under the Grant agreement ANR-19-P3IA-0004. The authors gratefully acknowledge the support of the DEEL project (`https://www.deel.ai/`)

References

[1] E. M. Achour, F. Malgouyres, and F. Mamalet. Existence, stability and scalability of orthogonal convolutional neural networks, 2021.

[2] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim. Sanity checks for saliency maps. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 9525–9536, Red Hook, NY, USA, 2018. Curran Associates Inc.

[3] L. Ambrosio and A. Pratelli. *Existence and stability results in the L1 theory of optimal transportation*, pages 123–160. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[4] M. Ancona, E. Ceolini, A. C. Öztireli, and M. H. Gross. A unified view of gradient-based attribution methods for deep neural networks. *CoRR*, 2017.

[5] C. Anil, J. Lucas, and R. Grosse. Sorting out Lipschitz function approximation. In K. Chaudhuri and R. Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 291–301, Long Beach, California, USA, June 2019. PMLR.

[6] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 214–223, International Convention Centre, Sydney, Australia, Aug. 2017. PMLR.

[7] L. Béthune, A. González-Sanz, F. Mamalet, and M. Serrurier. The many faces of 1-lipschitz neural networks. *CoRR*, abs/2104.05097, 2021.

[8] U. Bhatt, A. Weller, and J. M. F. Moura. Evaluating and aggregating feature-based model explanations. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 2020.

[9] Å. Björck and C. Bowie. An Iterative Algorithm for Computing the Best Estimate of an Orthogonal Matrix. *SIAM Journal on Numerical Analysis*, 8(2):358–364, June 1971.

[10] A. Boopathy, S. Liu, G. Zhang, C. Liu, P.-Y. Chen, S. Chang, and L. Daniel. Proper network interpretability helps adversarial robustness in classification. In *International Conference on Machine Learning*, 2020.

[11] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *2017 ieee symposium on security and privacy (sp)*, pages 39–57. IEEE, 2017.

[12] P. Chalasani, J. Chen, A. R. Chowdhury, S. Jha, and X. Wu. Concise explanations of neural networks using adversarial training. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, 2020.

[13] M. P. Da Silva, V. Courboulay, and P. Estraillier. Image complexity measure based on visual attention. In *2011 18th IEEE International Conference on Image Processing*, pages 3281–3284. IEEE, 2011.

[14] L. de Lara, A. González-Sanz, N. Asher, and J.-M. Loubes. Transport-based counterfactual models, 2021.

[15] T. Fel, R. Cadène, M. Chalvidal, M. Cord, D. Vigouroux, and T. Serre. Look at the variance! efficient black-box explanations with sobol-based sensitivity analysis. *CoRR*, abs/2111.04138, 2021.

[16] T. Fel, M. Ducoffe, D. Vigouroux, R. Cadene, M. Capelle, C. Nicodeme, and T. Serre. Don't lie to me! robust and efficient explainability with verified perturbation analysis. *arXiv preprint arXiv:2202.07728*, 2022.

[17] T. Fel, D. Vigouroux, R. Cadène, and T. Serre. How Good is your Explanation? Algorithmic Stability Measures to Assess the Quality of Explanations for Deep Neural Networks. In *2022 CVF Winter Conference on Applications of Computer Vision (WACV)*, Hawaii, United States, Jan. 2022.

[18] Fel, Thomas and Hervier, Lucas. Xplique: an neural networks explainability toolbox. 2021.

[19] A. Forsythe. Visual complexity: is that all there is? In *International Conference on Engineering Psychology and Cognitive Ergonomics*, pages 158–166. Springer, 2009.

[20] A. Forsythe, G. Mulhern, and M. Sawey. Confounds in pictorial sets: The role of complexity and familiarity in basic-level picture processing. *Behavior research methods*, 40(1):116–129, 2008.

[21] A. Ghorbani, A. Abid, and J. Zou. Interpretation of neural networks is fragile. In *Association for the Advancement of Artificial Intelligence*, 2017.

[22] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv:1412.6572 [stat.ML]*, Dec. 2014.

[23] Y. Goyal, Z. Wu, J. Ernst, D. Batra, D. Parikh, and S. Lee. Counterfactual visual explanations. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, 2019.

[24] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. C. Courville. Improved training of wasserstein gans. *CoRR*, abs/1704.00028, 2017.

[25] M. Hein and M. Andriushchenko. Formal Guarantees on the Robustness of a Classifier against Adversarial Manipulation. *arXiv:1705.08475 [cs, stat]*, May 2017.

[26] J. Heo, S. Joo, and T. Moon. Fooling neural network interpretations via adversarial model manipulation. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

[27] C. Hsieh, C. Yeh, X. Liu, P. K. Ravikumar, S. Kim, S. Kumar, and C. Hsieh. Evaluations and methods for explanation through robustness analysis. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021.

[28] C.-Y. Hsieh, C.-K. Yeh, X. Liu, P. Ravikumar, S. Kim, S. Kumar, and C.-J. Hsieh. Evaluations and methods for explanation through robustness analysis. In *International Conference on Learning Representations*, 2021.

[29] P. Jacob, É. Zablocki, H. Ben-Younes, M. Chen, P. Pérez, and M. Cord. STEEX: steering counterfactual explanations with semantics. *CoRR*, abs/2111.09094, 2021.

[30] A.-H. Karimi, B. Schölkopf, and I. Valera. Algorithmic recourse: from counterfactual explanations to interventions. In *ACM Conference on Fairness, Accountability, and Transparency*, pages 353–362, 2021.

[31] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[32] D. Lewis. Causation. *Journal of Philosophy*, 70(17):556–567, 1973.

[33] M. Li, X. Chen, X. Li, B. Ma, and P. M. Vitányi. The similarity metric. *IEEE transactions on Information Theory*, 50(12):3250–3264, 2004.

[34] Q. Li, S. Haque, C. Anil, J. Lucas, R. B. Grosse, and J. Jacobsen. Preventing gradient attenuation in lipschitz constrained convolutional networks. *arXiv:1911.00937*, Apr. 2019.

[35] Z. Q. Lin, M. J. Shafiee, S. Bochkarev, M. S. Jules, X. Y. Wang, and A. Wong. Do explanations reflect decisions? a machine-centric strategy to quantify the performance of explainability algorithms. In *NIPS*, 2019.

[36] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of International Conference on Computer Vision (ICCV)*, Dec. 2015.

[37] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. *ArXiv*, 2017.

[38] T. Miyato, T. Kataoka, M. Koyama, and Y. Yoshida. Spectral normalization for generative adversarial networks. *ArXiv*, abs/1802.05957, 2018.

[39] H. Ono, T. Takahashi, and K. Kakizaki. Lightweight Lipschitz Margin Training for Certified Defense against Adversarial Examples. *arXiv:1811.08080 [cs, stat]*, Nov. 2018.

[40] O. M. Parkhi, A. Vedaldi, and A. Zisserman. Deep face recognition. In *British Machine Vision Conference*, 2015.

[41] V. Petsiuk, A. Das, and K. Saenko. RISE: randomized input sampling for explanation of black-box models. *CoRR*, abs/1806.07421, 2018.

[42] P. Rodriguez, M. Caccia, A. Lacoste, L. Zamparo, I. Laradji, L. Charlin, and D. Vazquez. Beyond trivial counterfactual explanations with diverse valuable explanations. In *Int. Conf in Computer Vision (ICCV)*, 2021.

[43] A. Ross, H. Lakkaraju, and O. Bastani. Learning models for actionable recourse. *NeurIPS*, 2021.

[44] T. Salimans and D. P. Kingma. Weight normalization: A simple reparameterization to accelerate training of deep neural networks. *CoRR*, abs/1602.07868, 2016.

[45] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. Grad-CAM: Visual explanations from deep networks via gradient-based localization. *International Journal of Computer Vision*, 128(2):336–359, oct 2019.

[46] M. Serrurier, F. Mamalet, A. González-Sanz, T. Boissin, J.-M. Loubes, and E. Del Barrio. Achieving robustness in classification using optimal transport with hinge regularization. In *Conference on Computer Vision and Pattern Recognition (CVPR'21)*, 2021.

[47] K. Simonyan, A. Vedaldi, and A. Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In Y. Bengio and Y. LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Workshop Track Proceedings*, 2014.

[48] L. Sixt, M. Granz, and T. Landgraf. When explanations lie: Why many modified bp attributions fail. In *International Conference on Machine Learning*, 2020.

[49] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg. Smoothgrad: removing noise by adding noise, 2017.

[50] J. Sokolic, R. Giryes, G. Sapiro, and M. R. D. Rodrigues. Robust large margin deep neural networks. *IEEE Transactions on Signal Processing*, 65(16):4265–4280, Aug. 2017.

[51] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks, 2017.

[52] R. Tomsett, D. Harborne, S. Chakraborty, P. Gurram, and A. Preece. Sanity checks for saliency metrics. In *Association for the Advancement of Artificial Intelligence*, 2019.

[53] D. Tsipras, S. Santurkar, L. Engstrom, A. Turner, and A. Madry. Robustness may be at odds with accuracy, 2018.

[54] N. Vereshchagin and P. Vitányi. On lossy compression. Citeseer, 2005.

[55] C. Villani. *Optimal Transport: Old and New*. Grundlehren der mathematischen Wissenschaften. Springer Berlin Heidelberg, 2008.

[56] S. Wachter, B. Mittelstadt, and C. Russell. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841, 2017.

[57] P. Wang and N. Vasconcelos. Scout: Self-aware discriminant counterfactual explanations. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[58] Z. Wang, M. Fredrikson, and A. Datta. Robust models are more interpretable because attributions look normal, 2022.

[59] H. Xiao, K. Rasul, and R. Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *ArXiv*, abs/1708.07747, 2017.

[60] C.-K. Yeh, C.-Y. Hsieh, A. S. Suggala, D. I. Inouye, and P. Ravikumar. On the (in)fidelity and sensitivity of explanations. In *NeurIPS*, 2019.

A Appendix

A.1 Additional definition and proofs

Let us first recall the optimal transport problem associated with the minimization of $\mathcal{L}^{hKR}_{\lambda,m}$:

$$\inf_{f\in Lip_1(\Omega)} \mathcal{L}^{hKR}_{\lambda(f),m} = \inf_{\pi\in\Pi^p_\lambda(P_+,P_-)} \int_{\Omega\times\Omega} |\mathbf{x}-\mathbf{z}|d\pi + \pi_{\mathbf{x}}(\Omega) + \pi_{\mathbf{z}}(\Omega) - 1 \tag{6}$$

Where $\Pi^p_\lambda(P_+,P_-)$ is the set consisting of positive measures $\pi \in \mathcal{M}_+(\Omega\times\Omega)$ which are absolutely continuous with respect to the joint measure $dP_+ \times dP_-$ and $\frac{d\pi_{\mathbf{x}}}{dP_+} \in [p, p(m+\lambda)]$, $\frac{d\pi_{\mathbf{z}}}{dP_-} \in [1-p, (1-p)(m+\lambda)]$. We name π^* the optimal transport plan according to Eq.6 and and f^* the associated potential function.

Proof of proposition 1: According to [46], we have

$$||\nabla_x f^*(\mathbf{x})|| = 1$$

almost surely and

$$\mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\pi^*}\left(|f^*(\mathbf{x}) - f^*(\mathbf{y})| = ||\mathbf{x}-\mathbf{y}||\right) = 1$$

Following the proof of proposition 1 in [24] and [3] we have :
Given $\mathbf{x}_\alpha = \alpha * x + (1-\alpha)\mathbf{y}$, $0 \le \alpha \le 1$

$$\mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\pi^*}\left(\nabla_x f^*(\mathbf{x}_\alpha) = \frac{\mathbf{x}_\alpha - \mathbf{y}}{||\mathbf{x}_\alpha - \mathbf{y}||}\right) = 1.$$

So for for $\alpha = 1$ whe have

$$\mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\pi^*}\left(\nabla_x f^*(\mathbf{x}) = \frac{\mathbf{x}-\mathbf{y}}{||\mathbf{x}-\mathbf{y}||}\right) = 1$$

and then

$$\mathbb{P}_{(\mathbf{x},\mathbf{y})\sim\pi^*}\left(\mathbf{y} = \mathbf{x} - \nabla_x f^*(\mathbf{x}).||\mathbf{x}-\mathbf{y}||\right) = 1$$

This prove the proposition 1 by choosing $t = ||\mathbf{x}-\mathbf{y}||$.■

Proof of proposition 2: Let P_+ and P_- two separable distributions with minimal distance ϵ and f^* an optimal solution minimizing the $\mathcal{L}^{hKR}_{\lambda,m}$ with $m < 2\epsilon$. According to [46], f^* is 100% accurate. Since the classification is based on the sign of f we have : $\forall \mathbf{x} \in P_+$, $f^*(\mathbf{x}) \ge 0$ and $\forall \mathbf{y} \in P_-$, $f^*(\mathbf{y}) \le 0$. Given $\mathbf{x} \in P_+$ and $\mathbf{y} = tr_\pi(\mathbf{x}) = \mathbf{x} - t\nabla_x f^*(\mathbf{x})$ and $\mathbf{y} \in P_-$. According to the previous proposition we have :

$$\begin{aligned}
|f^*(\mathbf{x}) - f^*(\mathbf{y})| &= ||\mathbf{x}-\mathbf{y}|| \\
|f^*(\mathbf{x}) - f^*(\mathbf{y})| &= ||\mathbf{x} - (\mathbf{x} - t\nabla_x f^*(\mathbf{x}))|| \\
|f^*(\mathbf{x}) - f^*(\mathbf{y})| &= ||t\nabla_x f^*(\mathbf{x}))|| \\
|f^*(\mathbf{x}) - f^*(\mathbf{y})| &= t.||\nabla_x f^*(\mathbf{x}))|| && (t \ge 0) \\
|f^*(\mathbf{x}) - f^*(\mathbf{y})| &= t && (\nabla_x f^*(\mathbf{x}) = 1) \\
f^*(\mathbf{x}) - f^*(\mathbf{y}) &= t && (f^*(\mathbf{x}) \ge 0, f^*(\mathbf{y}) \le 0) \\
f^*(\mathbf{y}) &= f^*(\mathbf{x}) - t
\end{aligned}$$

since $f^*(\mathbf{y}) \le 0$ we obtain :

$$f^*(\mathbf{x}) \le t$$

Since f^* is continuous, $\exists t' > 0$ such that $\mathbf{x}_\delta = \mathbf{x} - t'\nabla_x f^*(\mathbf{x})$ and $f^*(\mathbf{x}_\delta) = 0$. We have :

$$\begin{aligned}
|f^*(\mathbf{x}) - f^*(\mathbf{x}_\delta|) &\le ||\mathbf{x}-\mathbf{x}_\delta|| \\
f^*(\mathbf{x}) &\le ||\mathbf{x} - (\mathbf{x} - t'\nabla_x f^*(\mathbf{x}))|| \\
f^*(\mathbf{x}) &\le t'
\end{aligned}$$

and

$$
\begin{aligned}
|f^*(\mathbf{x}_\delta) - f^*(\mathbf{y})| &\leq ||\mathbf{x}_\delta - \mathbf{y}|| \\
-f^*(\mathbf{y}) &\leq ||(\mathbf{x} - t'\nabla_x f^*(\mathbf{x})) - (\mathbf{x} - t\nabla_x f^*(\mathbf{x}))|| \\
-f^*(\mathbf{y}) &\leq t - t' \\
-f^*(\mathbf{y}) &\leq ||\mathbf{x} - \mathbf{y}|| - t' \qquad)
\end{aligned}
$$

Then, if $f^*(\mathbf{x}) < t'$ we have

$$
\begin{aligned}
f^*(\mathbf{x}) - f^*(\mathbf{y}) &< t' + ||\mathbf{x} - \mathbf{y}|| - t' \\
f^*(\mathbf{x}) - f^*(\mathbf{y}) &< ||\mathbf{x} - \mathbf{y}||
\end{aligned}
$$

which is a contradiction so $f^*(\mathbf{x}) = t'$ and

$$\mathbf{x}_\delta = \mathbf{x} - f^*(\mathbf{x})\nabla_x f^*(\mathbf{x})$$

■

A.2 Parameters and architectures

A.2.1 Datasets

FashionMNIST has 50,000 images for training and 10,000 for test of size $28 \times 28 \times 1$, with 10 classes.

CelebA contains 162,770 training samples, 19,962 samples for test of size $218 \times 178 \times 3$. We have used a subset of 22 labels: *Attractive*, *Bald*, *Big_Nose*, *Black_Hair*, *Blond_Hair*, *Blurry*, *Brown_Hair*, *Eyeglasses*, *Gray_Hair*, *Heavy_Makeup*, *Male*, *Mouth_Slightly_Open*, *Mustache*, *Receding_Hairline*, *Rosy_Cheeks*, *Sideburns*, *Smiling*, *Wearing_Earrings*, *Wearing_Hat*, *Wearing_Lipstick*, *Wearing_Necktie*, *Young*.

Note that labels in CelebA are very unbalanced (see Table 5, with less than 5% samples for *Mustache* or *Wearing_Hat* for instance). Thus we will use Sensibility and Specificity as metrics.

Table 5: CelebA label distribution: proportion of positive samples in training set (testing set) [bold: very unbalanced labels]

Attractive	*Bald*	*Big_Nose*	*Black_Hair*	*Blond_Hair*
0.51 (0.50)	**0.02 (0.02)**	**0.24 (0.21)**	**0.24 (0.27)**	**0.15 (0.13)**
Blurry	*Brown_Hair*	*Eyeglasses*	*Gray_Hair*	*Heavy_Makeup*
0.05 (0.05)	**0.20 (0.18)**	**0.06 (0.06)**	**0.04 (0.03)**	0.38 (0.40)
Male	*Mouth_Slightly_Open*	*Mustache*	*Receding_Hairline*	*Rosy_Cheeks*
0.42 (0.39)	0.48 (0.50)	**0.04 (0.04)**	**0.08 (0.08)**	**0.06 (0.07)**
Sideburns	*Smiling*	*Wearing_Earrings*	*Wearing_Hat*	*Wearing_Lipstick*
0.06 (0.05)	0.48 (0.50)	**0.19 (0.21)**	**0.05 (0.04)**	0.47 (0.52)
Wearing_Necktie	*Young*			
0.12 (0.14)	0.78 (0.76)			

preprocessing: Images are normalized between $[0, 1]$. For CelebA dataset, data augmentation is used with random crop, horizontal flip, random brightness, and random contrast. No data augmentation is used for FashionMNIST.

A.2.2 Architectures

As indicated in the paper, linear layers for OTNN and unconstrained networks are equivalent (same number of layers and neurons), but unconstrained networks use batchnorm and ReLU layer for activation, whereas OTNN only use GroupSort2 [5, 46] activation. OTNN are built using *DEEL.LIP*[2] library.

1-Lipschitz networks parametrization. Several soltutions have been proposed to set the Lipschitz constant of affine layers: Weight clipping [6] (WGAN), Frobenius normalization [44] and spectral normalization [38]. In order to avoid vanishing gradients, orthogonalization can be done using Björck algorithm [9]. DEEL.LIP implements most of these solutions, but we focus on layers called *SpectralDense* and *SpectralConv2D*, with spectral normalization [38] and Björck algorithm [9]. Most activation functions are Lipschitz, including ReLU, sigmoid, but we use GroupSort2 proposed by [5], and defined by the following equation:

$$\text{GroupSort2}(x)_{2i,2i+1} = [\min(x_{2i}, x_{2i+1}), \max(x_{2i}, x_{2i+1})]$$

Network architectures used for CelebA dataset are described in Table 6.

Network architectures used for FashionMNIST dataset are described in Table 7. The same OTNN architecture is used for MNIST expermentation presented in Fig. 1.

A.2.3 Losses and optimizer

The loss functions used for training the neural networks are: cross-entropy for unconstrained networks (categorical for multiclass, and binary for multilabel settings), and the proposed variant $\mathcal{L}^{hKR}_{\lambda,\alpha}$ of

[2] `https://github.com/deel-ai/deel-lip` distributed under MIT License (MIT)

Table 6: CelebA Neural network architectures: Sconv2D is SpectralConv2D, GS2 is GroupSort2, L2Pool is L2NormPooling, SDense is SpectralDense, BN is BatchNorm, AvgPool is AveragePooling

Dataset	OTNN	Unconstrained NN	
	Layer	Layer	Output size
CelebA	Input	Input	$218 \times 178 \times 3$
	SConv2D, GS2	Conv2D, BN, ReLU	$218 \times 178 \times 16$
	SConv2D, GS2	Conv2D, BN, ReLU	$218 \times 178 \times 16$
	L2Pool	AvgPool	$109 \times 89 \times 16$
	SConv2D, GS2	Conv2D, BN, ReLU	$109 \times 89 \times 32$
	SConv2D, GS2	Conv2D, BN, ReLU	$109 \times 89 \times 32$
	L2Pool	AvgPool	$54 \times 44 \times 32$
	SConv2D, GS2	Conv2D, BN, ReLU	$54 \times 44 \times 64$
	SConv2D, GS2	Conv2D, BN, ReLU	$54 \times 44 \times 64$
	SConv2D, GS2	Conv2D, BN, ReLU	$54 \times 44 \times 64$
	L2Pool	AvgPool	$27 \times 22 \times 64$
	SConv2D, GS2	Conv2D, BN, ReLU	$27 \times 22 \times 128$
	SConv2D, GS2	Conv2D, BN, ReLU	$27 \times 22 \times 128$
	SConv2D, GS2	Conv2D, BN, ReLU	$27 \times 22 \times 128$
	L2Pool	AvgPool	$13 \times 11 \times 128$
	SConv2D, GS2	Conv2D, BN, ReLU	$13 \times 11 \times 128$
	SConv2D, GS2	Conv2D, BN, ReLU	$13 \times 11 \times 128$
	SConv2D, GS2	Conv2D, BN, ReLU	$13 \times 11 \times 128$
	L2Pool	AvgPool	$6 \times 5 \times 128$
	Flatten, SDense, GS2	Flatten, Dense, BN, ReLU	256
	SDense, GS2	Dense,BN, ReLU	256
	SDense	Dense	22

hKR: i.e. we use Eq. 3 with the soft Hinge (Eq. 5) for CelebA, with hyperparameters λ is set to 20, and $\alpha = 0.05$. For FashionMNIST, we use Eq. 4 with the soft Hinge (Eq. 5), λ is set to 5, and $\alpha = 0.2$. As explained in the paper we set $\alpha = 1/\lambda$.

We train all networks with ADAM optimizer [31], using a batch size of 128, number of epochs 200, and a fixed learning rate $1\mathrm{e}{-2}$ for CelebA and $1\mathrm{e}{-4}$ for FashionMNIST.

Table 7: FashionMNIST Neural network architectures: Sconv2D is SpectralConv2D, GS2 is GroupSort2, SDense is SpectralDense, BN is BatchNorm, AvgPool is AveragePooling, SGAvgPool is ScaledGlobalAveragePooling (DEEL.LIP), GAvgPool is GlobalAveragePooling

Dataset	OTNN	Unconstrained NN	
	Layer	Layer	Output size
FashionMNIST	Input	Input	$28 \times 28 \times 1$
	SConv2D, GS2	Conv2D, BN, ReLU	$28 \times 28 \times 96$
	SConv2D, GS2	Conv2D, BN, ReLU	$28 \times 28 \times 96$
	SConv2D, GS2	Conv2D, BN, ReLU	$28 \times 28 \times 96$
	SConv2D (stride=2), GS2	Conv2D (stride=2), BN, ReLU	$14 \times 14 \times 96$
	SConv2D, GS2	Conv2D, BN, ReLU	$14 \times 14 \times 192$
	SConv2D, GS2	Conv2D, BN, ReLU	$14 \times 14 \times 192$
	SConv2D, GS2	Conv2D, BN, ReLU	$14 \times 14 \times 192$
	SConv2D (stride=2), GS2	Conv2D (stride=2), BN, ReLU	$7 \times 7 \times 192$
	SConv2D, GS2	Conv2D, BN, ReLU	$7 \times 7 \times 384$
	SConv2D, GS2	Conv2D, BN, ReLU	$7 \times 7 \times 384$
	SConv2D, GS2	Conv2D, BN, ReLU	$7 \times 7 \times 384$
	SGAvgPool	GAvgPool	384
	SDense	Dense	10

A.3 Complementary results

A.3.1 FashionMNIST performances and ablation study

Table 8 presents different performance resuts on FashionMNIST. First line is the reference unconstrained network. Second line shows the performances of the new version of $\mathcal{L}^{hKR}_{\lambda,\alpha}$. First, we observe that the auto-tuning of the m is as accurate as when we set it manually (line three). This is expected since the goal of the formulation of $\mathcal{L}^{hKR}_{\lambda,\alpha}$with margin penalty is only to reduce the number of hyperparameters (so to simplify the parameters' tuning). Table 8 also shows that the new version of the $\mathcal{L}^{hKR}_{\lambda,\alpha}$ in the multiclass case (Eq. 5) outperforms the $\mathcal{L}^{hKR}_{\lambda,m}$defined in [46] (Eq. 4). Obviously, the accuracy enhancement is obtained at the expense of the robustness. The main interest of this new loss is to provide a wider range in the accuracy/robustness trade-off.

Table 8: FashionMNIST accuracy comparison with the different version of multiclass $\mathcal{L}^{hKR}_{\lambda,m}$. For the fixed margin, we use the one that performs best by parameter tuning (i.e. $m = 0.5$)

Model	Accuracy
Unconstrained	88.5
OTNN $\mathcal{L}^{hKR}_{\lambda,\alpha}(\alpha = 0.1, \lambda = 10)$	**88.6**
OTNN $\mathcal{L}^{hKR}_{\lambda,m}$fixed margin ($\lambda = 10, m = 0.5$)	**88.6**
OTNN $\mathcal{L}^{hKR}_{\lambda,m}$multiclass version [46] ($\lambda = 10, m = 0.5$)	72.2

A.3.2 CelebA performances

Table 9 presents the Sensibility and Specificity for each label reached by Unconstrained network and OTNN.

As a reminder, given True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) samples, Sensitivity (true positive rate or Recall) is defined by:

$$Sens = \frac{TP}{TP + FN}$$

Specificity (true negative rate) is defined by:

$$Spec = \frac{TN}{TN + FP}$$

Table 9: CelebA performance results for unconstrained and OTNN networks

Model	Metrics: Sensibility/Specificity			
	Attractive	*Bald*	*Big_Nose*	*Black_Hair*
Unconstrained	**0.83 / 0.81**	0.64 / 1.00	**0.65 / 0.87**	**0.74 / 0.95**
OTNN	0.80 / 0.75	**0.87 / 0.83**	0.73 / 0.70	0.78 / 0.84
	Blond_Hair	*Blurry*	*Brown_Hair*	*Eyeglasses*
Unconstrained	**0.86 / 0.97**	**0.49 / 0.99**	**0.80 / 0.88**	**0.96 / 1.00**
OTNN	0.86 / 0.89	0.66 / 0.72	0.81 / 0.73	0.80 / 0.89
	Gray_Hair	*Heavy_Makeup*	*Male*	*Mouth_Slightly_Open*
Unconstrained	0.62 / 0.99	**0.84 / 0.95**	**0.98 / 0.98**	**0.93 / 0.94**
OTNN	**0.84 / 0.83**	0.89 / 0.83	0.92 / 0.89	0.80 / 0.89
	Mustache	*Receding_Hairline*	*Rosy_Cheeks*	*Sideburns*
Unconstrained	0.47 / 0.99	0.47 / 0.98	0.46 / 0.99	**0.79 / 0.98**
OTNN	**0.86 / 0.76**	**0.81 / 0.79**	**0.82 / 0.80**	0.79 / 0.82
	Smiling	*Wearing_Earrings*	*Wearing_Hat*	*Wearing_Lipstick*
Unconstrained	**0.90 / 0.95**	**0.84 / 0.90**	**0.89 / 0.99**	**0.90 / 0.96**
OTNN	0.84 / 0.88	0.78 / 0.72	0.86 / 0.90	0.90 / 0.89
	Wearing_Necktie	*Young*		
Unconstrained	0.75 / 0.98	**0.95 / 0.65**		
OTNN	**0.87 / 0.86**	0.79 / 0.69		

A.4 Complementary explanations metrics

A.4.1 Explanation attribution methods

An attribution method provides an importance score for each input variables x_i in the output $f(x)$. The library used to generate the attribution maps is Xplique [18].

For a full description of attribution methods, we advise to read [16], Appendix B. We will only remind here the equations of

- Saliency: $g(\mathbf{x}) = |\nabla_{\mathbf{x}} f(\mathbf{x})|$
- SmoothGrad: $g(\mathbf{x}) = \underset{\delta \sim \mathcal{N}(0, \mathbf{I}\sigma)}{\mathbb{E}} (\nabla f(\mathbf{x} + \delta))$

SmoothGrad is evaluated on $N = 50$ samples on a normal distribution of standard deviation $\sigma = 0.2$ around x. Integrated Gradient [51], noted IG, is also evaluated on $N = 50$ samples at regular intervals. Grad-CAM [45], noted GC, is classically applied on the last convolutional layer. And RISE black-box method [41] is evaluated on $N = 4000$ samples.

A.4.2 XAI metrics

For the experiments we use four fidelity metrics, evaluated on 1000 samples of test datasets:

- Deletion [41]: it consists in measuring the drop of the score when the important variables are set to a baseline state. Formally, at step k, with u the k most important variables according to an attribution method, the Deletion$^{(k)}$ score is given by:

  $$\text{Deletion}^{(k)} = f(\mathbf{x}_{[\mathbf{x}_u = \mathbf{x}_0]})$$

  The AUC of the Deletion scores is then measured to compare the attribution methods (↓ is better). The baseline x_0 can either be a zero value (*Deletion-zero*), or a uniform random value (*Deletion-uniform*).
- Insertion [41]: this metric is the inverse of Deletion, starting with an image in a baseline state and then progressively adding the most important variables. Formally, at step k, with u the most important variables according to an attribution method, the Insertion$^{(k)}$ score is given by:

  $$\text{Insertion}^{(k)} = f(\mathbf{x}_{[\mathbf{x}_{\overline{u}} = \mathbf{x}_0]})$$

The AUC is also measured to compare attribution methods (↑ is better). The baseline is the same as for Deletion.

- μFidelity [8]: this metric measures the correlation between the fall of the score when variables are put at a baseline state and the importance of these variables. Formally:

$$\mu\text{Fidelity} = \underset{\substack{u \subseteq \{1,\dots,d\} \\ |u|=k}}{\text{Corr}} \left(\sum_{i \in u} g(\mathbf{x})_i, f(\mathbf{x}) - f(\mathbf{x}_{[\mathbf{x}_u = \mathbf{x}_0]}) \right)$$

For all experiments, k is equal to 20% of the total number of variables, and cutting the image in a grid of 20×20. The baseline is the same as the one used by Deletion. Being a correlation score, we can either compare attribution methods, or different neural networks on the same attribution method (↑ is better).

- Robustness-Sr [27]: this metric evaluate the average adversarial distance when the attack is done only on the most relevant features. Formally, given the u most important variables:

$$\text{Robustness-Sr} = \left\{ \underset{\delta}{min} ||\delta|| s.t. f(\mathbf{x} + \delta) \neq {}_x, \delta_{\overline{u}} = 0 \right\}$$

where $\delta_{\overline{u}} = 0$ indicates that adversarial attack is authorized only on the set u. The AUC is measured to compare attribution methods (↓ is better). Note this metric cannot be used to compare different networks, since it depends on the robustness of the network.

We use also several other metrics:

- Distances between explanations: to compare two explanation $f(x)$, we use either L_2 distance, or $1 - \rho$ where ρ is the Spearman rank correlation [2, 17, 52] (↓ is better).
- Explanation complexity: we use the JPEG compression size as a proxy of the Kolmogorov complexity (↓ is better).
- Stability: As proposed in [60], the Stability is evaluated by the average distance of explanations provided for random samples drawn in a ball of radius 0.3 around x. As before, the distance can be either L_2 or $1 - \rho$ (↓ is better).

A.4.3 Supplementary metric results

In this section we present several experiments and metrics that we were not able to insert in the core of the paper.

Deletion-zero and Insertion-zero are evaluated on CelebA and FashionMNIST dataset. It is known that the baseline value can be a bias for these metrics, and we are convinced that it has a higher influence with 1-Lipschitz networks. Even if results for Deletion-zero and Insertion-zero are less obvious than for Deletion and Insertion Uniform, we can see in Table 10, that for these metrics, the rank of Saliency is most of the time higher for OTNN.

To leverage the bias of the baseline value, as proposed in [27] we evaluated the Robustness-SR metric, Saliency map on OTNN achieves top-ranking scores. One might argue that scores for unconstrained networks are lower, but this is directly linked to the higher intrinsic robustness of OTNNand thus cannot be compared.

The full results for the explanation complexity is given on Table 12. The complexity is still lower for OTNN on FashionMNIST, even if the gap with Unconstrained networks is narrower than for CelebA.

Table 10: Insertion and Deletion metrics evaluation; GC: GradCam, GI: Gradient.Input, IG: Integrated Gradient, Saliency Rk : Rank (comparison by line only : in bold best score)

Dataset	Network	Deletion-Zero (↓ is better)					
		GC	GI	IG	Rise	Saliency	SmoothGrad
		Deletion-Zero					
CelebA	OTNN	8.01	7.04	7.05	7.09	6.98 (Rk2)	**6.96**
	Unconstrained	5.77	4.56	4.38	5.07	**4.13** (Rk1)	4.51
Fashion-	OTNN	0.24	0.16	**0.15**	0.26	0.20 (Rk4)	0.19
MNIST	Unconstrained	0.33	0.28	0.23	**0.16**	0.38 (Rk5)	0.39
		Insertion-zero (↑ is better)					
CelebA	OTNN	10.26	11.63	11.58	**15.50**	10.06 (Rk6)	10.10
	Unconstrained	14.24	11.71	12.37	**15.70**	6.67 (Rk6)	7.65
Fashion-	OTNN	0.31	0.46	**0.47**	0.36	0.36 (Rk4)	0.39
MNIST	Unconstrained	0.53	0.59	0.68	**0.73**	0.45 (Rk6)	0.46

Table 11: Robustness-SR metrics evaluation; GC: GradCam, GI: Gradient.Input, IG: Integrated Gradient, Saliency Rk : Rank (comparison by line only : in bold best score)

Dataset	Network	Robustness-SR (↓ is better)					
		GC	GI	IG	Rise	Saliency	SmoothGrad
CelebA	OTNN	28.54	14.01	13.28	30.54	**11.64** (Rk1)	12.65
	Unconstrained	11.11	9.19	10.00	15.15	7.38 (Rk2)	**7.20**
Fashion-	OTNN	**1.69**	3.31	3.36	3.27	2.29 (Rk3)	2.01
MNIST	Unconstrained	1.17	1.36	1.17	**1.15**	1.21 (Rk4)	1.25

A.5 Complementary qualitative results

In this section, we provide more samples of couterfactual exlanations for OTNN, based on the gradient, i.e. $\mathbf{x} - t * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$ for $t > 1$.

Fig. 6 gives more results on FashionMNIST.

Fig. 7,8,9,10,11,12 presents more results on the labels presented in the core of the paper, *Mouth_Slightly_Open*, *Mustache*,*Wearing_Hat*.

To end with, we presents results for other labels of CelebA. For ethic concerns we have hidden labels that can be subject to misinterpretation, such as *Attractive*, *Male*, *Big_Nose*.

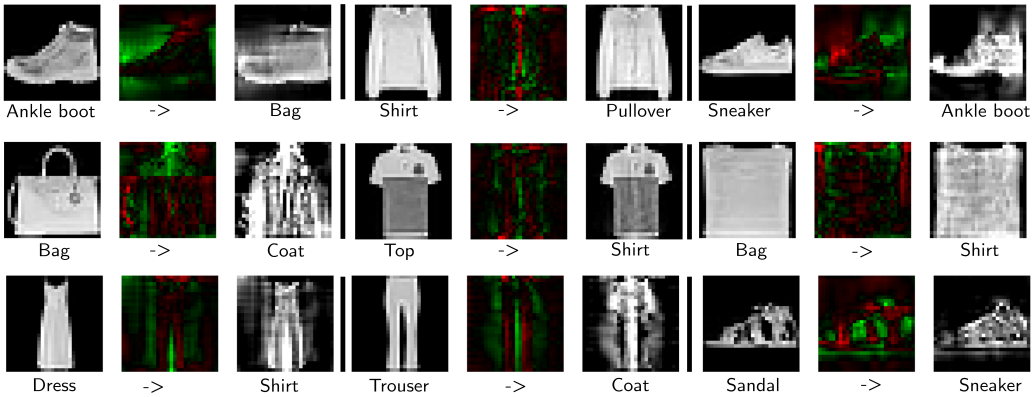


Figure 6: FashionMNIST samples

Table 12: Complexity of Saliency map by JPEG compression (kB): lower is better

	CelebA	FashionMNIST
OTNN	9.48	0.92
Unconstrained	16.84	0.94

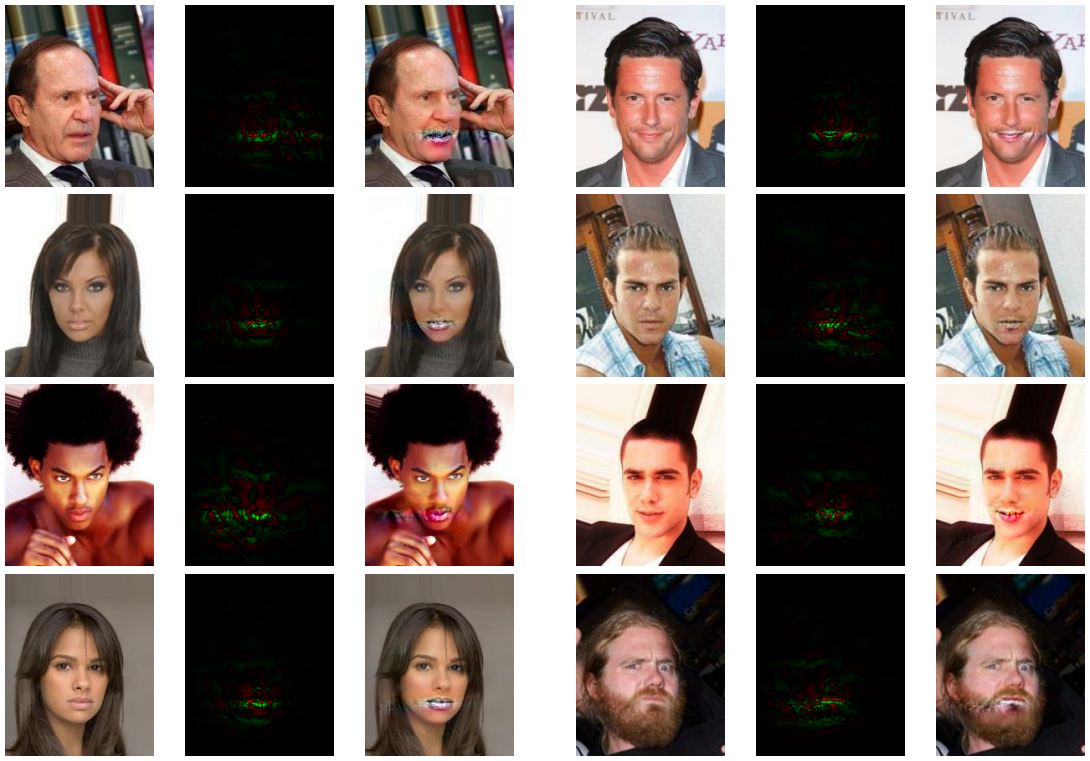


Figure 7: Samples from label Mouth_slightly_open: left source image (closed) , center difference image, right counterfactual (open) of form $\mathbf{x} - 10 * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$

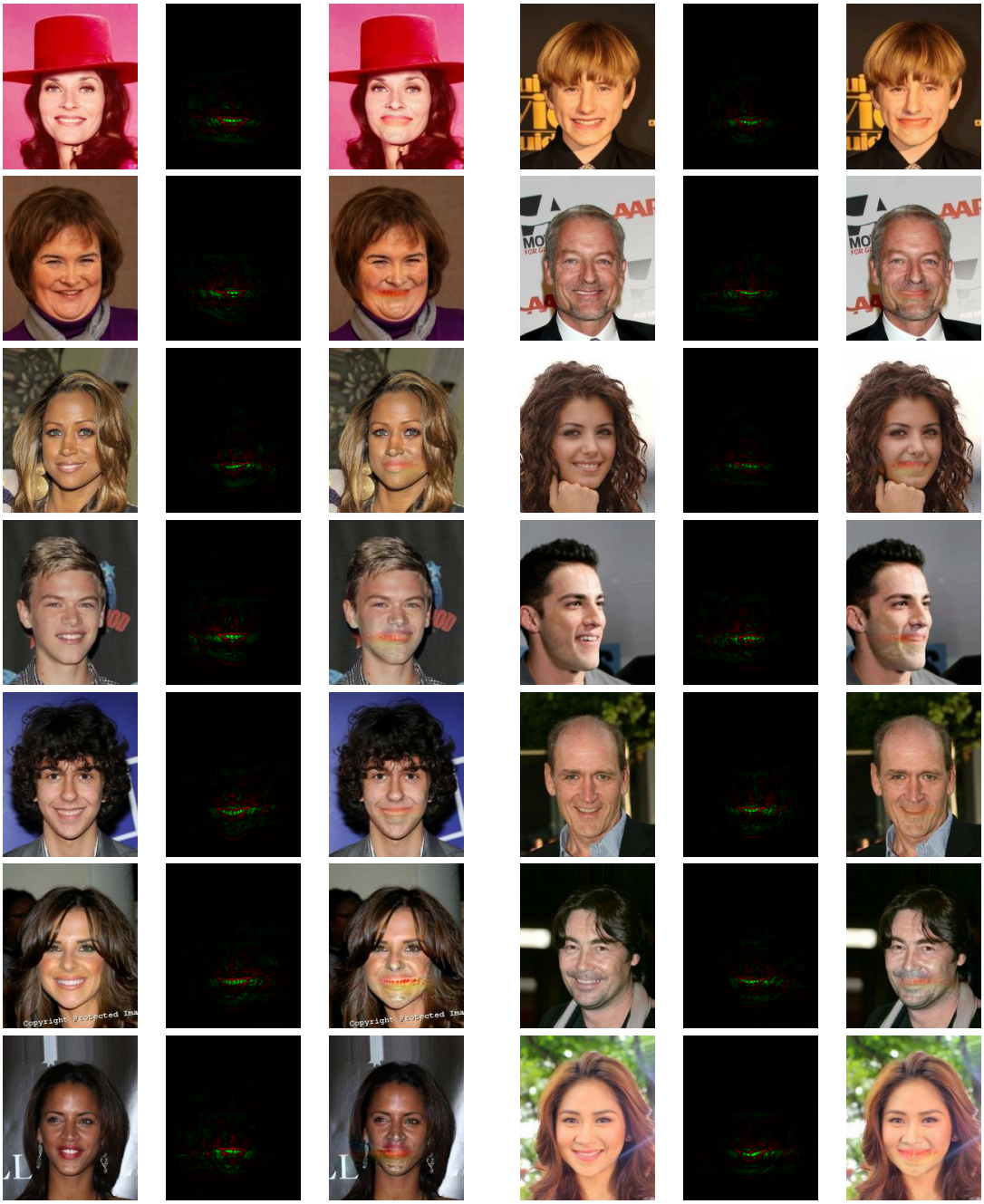


Figure 8: Samples from label Mouth_slightly_open: left source image (open) , center difference image, right counterfactual (close) of form $\mathbf{x} - 10 * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$

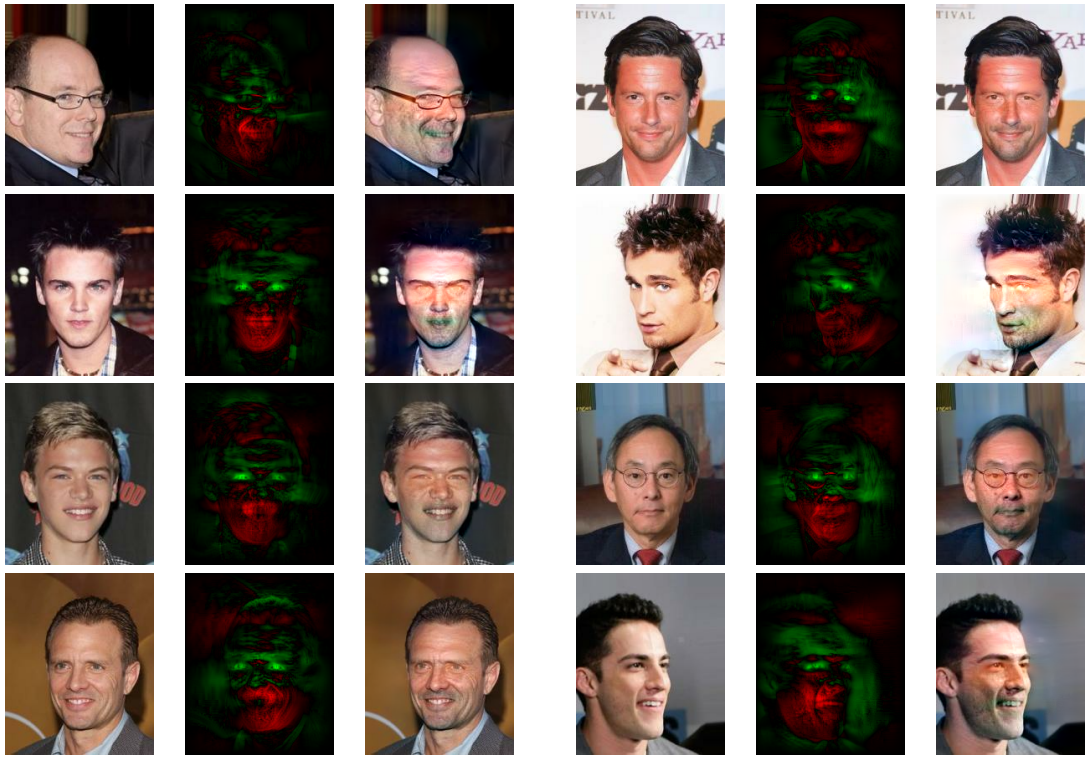


Figure 9: Samples from label Mustache: left source image (no mustache) , center difference image, right counterfactual (mustache) of form $\mathbf{x} - t * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$ with $t \in \{5, 10, 20\}$

Figure 10: Samples from label Mustache: left source image (Mustache) , center difference image, right counterfactual (Non Mustache) of form $\mathbf{x} - t * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$, $t \in 5, 10$

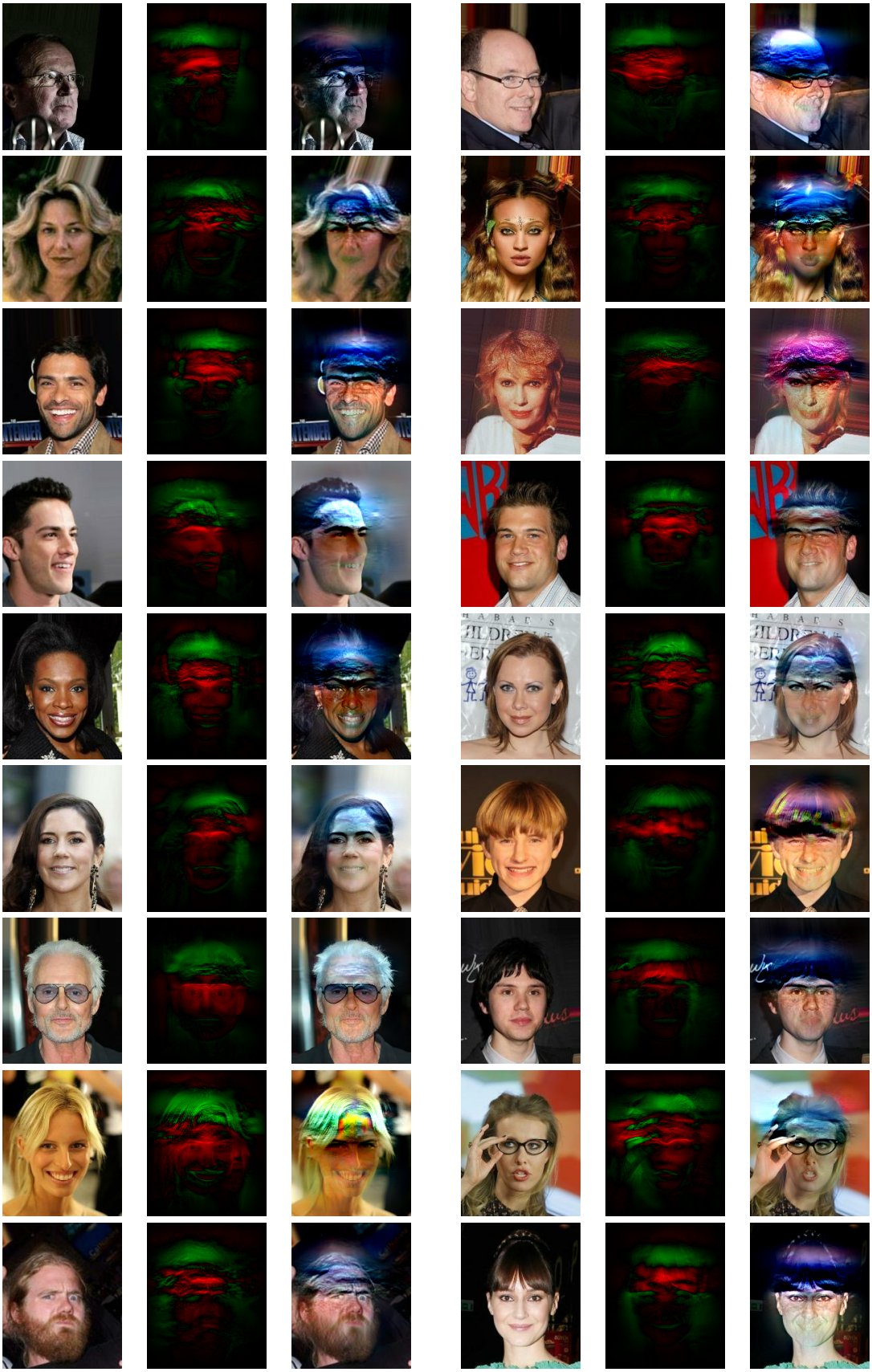


Figure 11: Samples from label Wearing Hat: left source image (No Hat) , center difference image, right counterfactual (Hat) of form $\mathbf{x} - t * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$, $t \in 5, 10$

Figure 12: Samples from label Wearing Hat: left source image (Hat) , center difference image, right counterfactual (No Hat) of form $\mathbf{x} - t * \hat{f}(\mathbf{x}) \nabla_x \hat{f}(\mathbf{x})$, $t \in 5, 10$

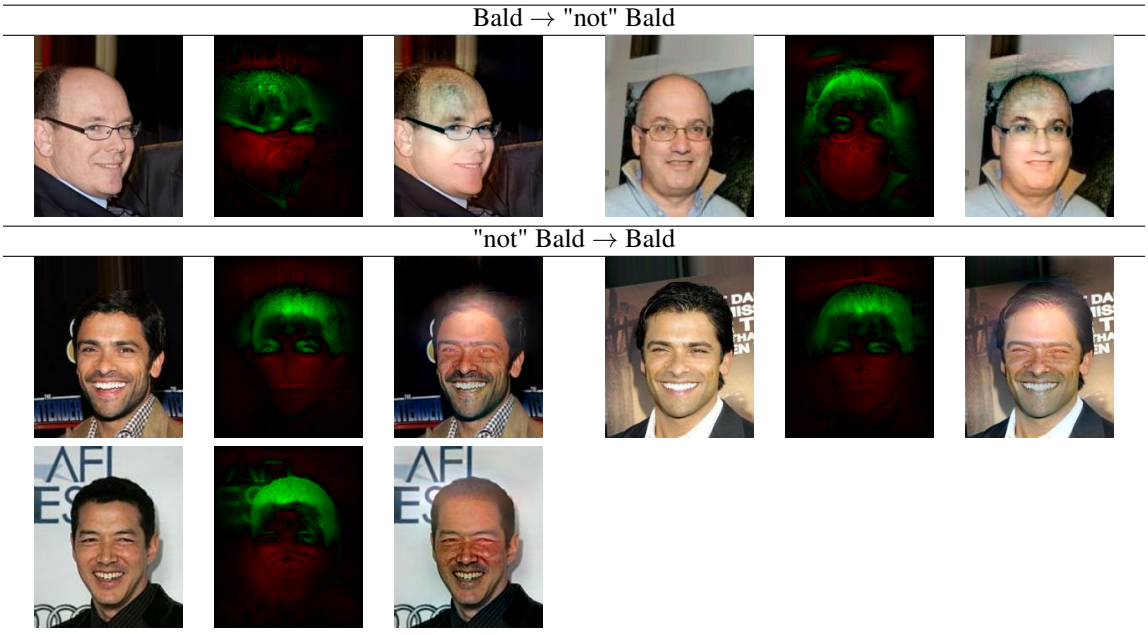
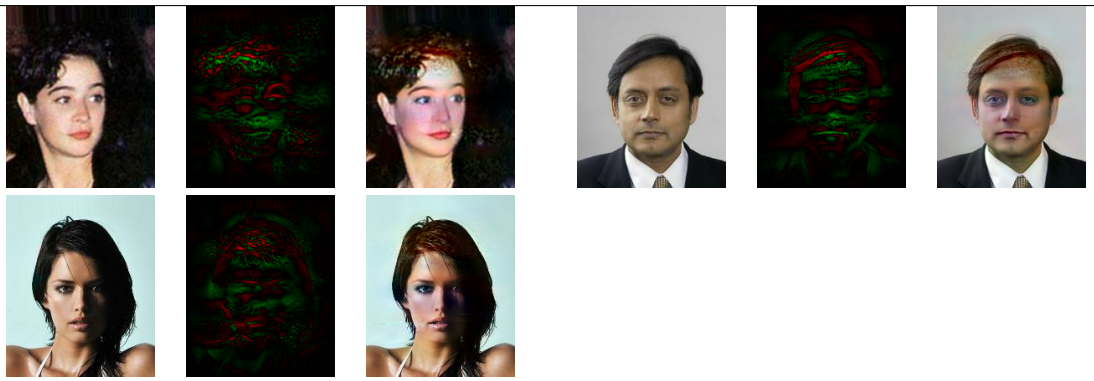


Figure 13: Samples from label Bald

Black_Hair → "not" Black_Hair



"not" Black_Hair → Black_Hair

Figure 14: Samples from label Black_Hair

Blond_Hair → "not" Blond_Hair

"not" Blond_Hair → Blond_Hair

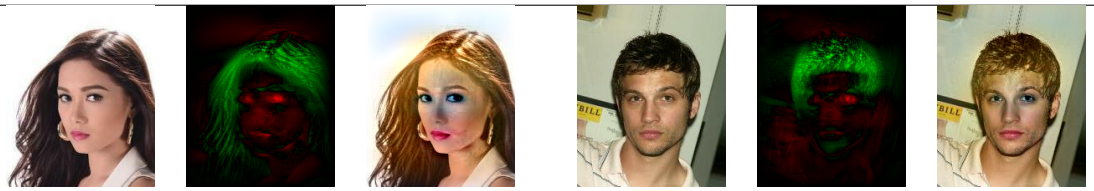
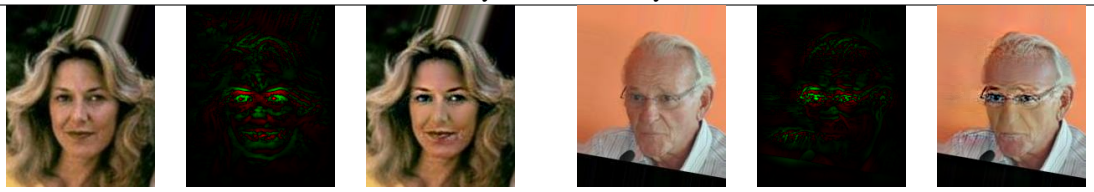


Figure 15: Samples from label Blond_Hair

Blurry → "not" Blurry



"not" Blurry → Blurry

Figure 16: Samples from label Blurry

Figure 17: Samples from label Brown_Hair

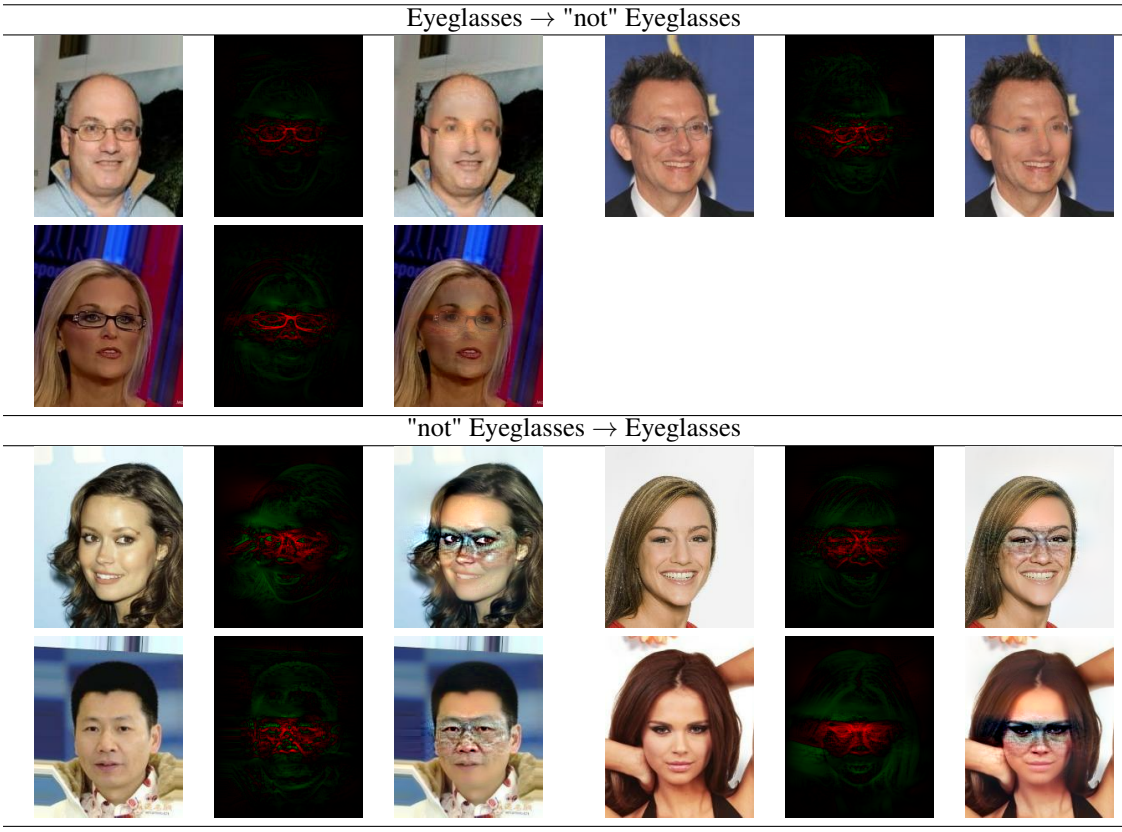


Figure 18: Samples from label Eyeglasses

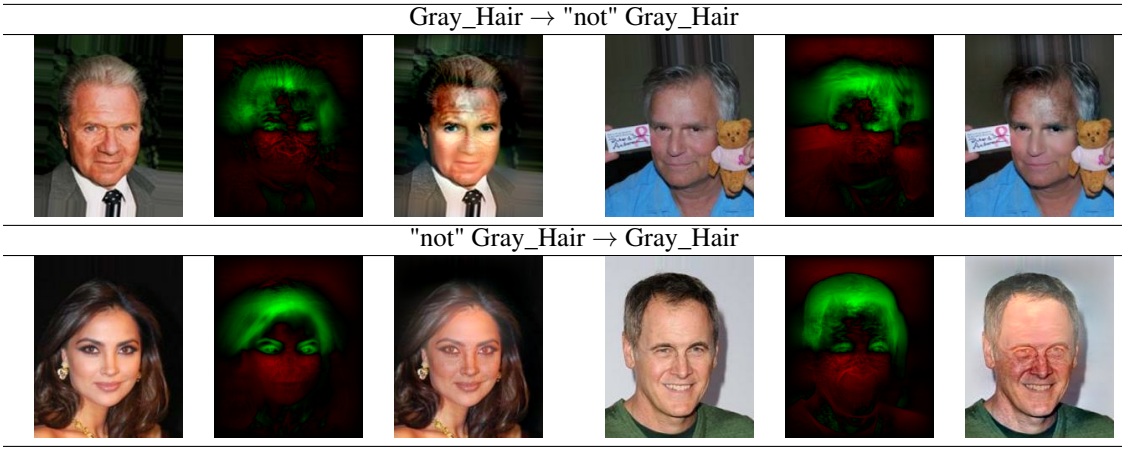


Figure 19: Samples from label Gray_Hair

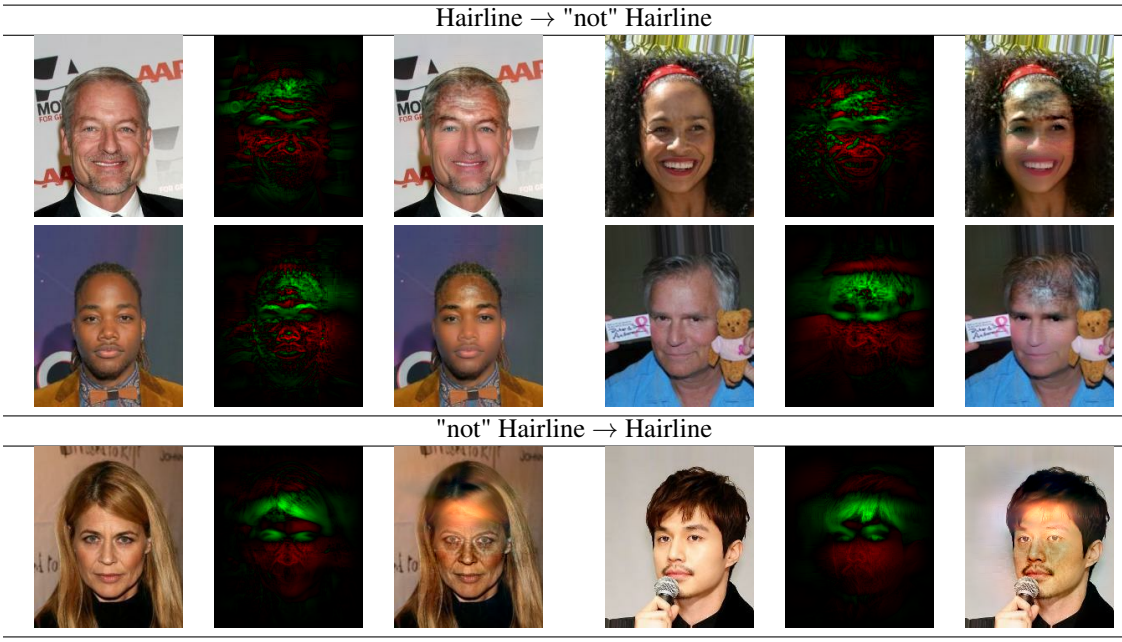


Figure 20: Samples from label Hairline

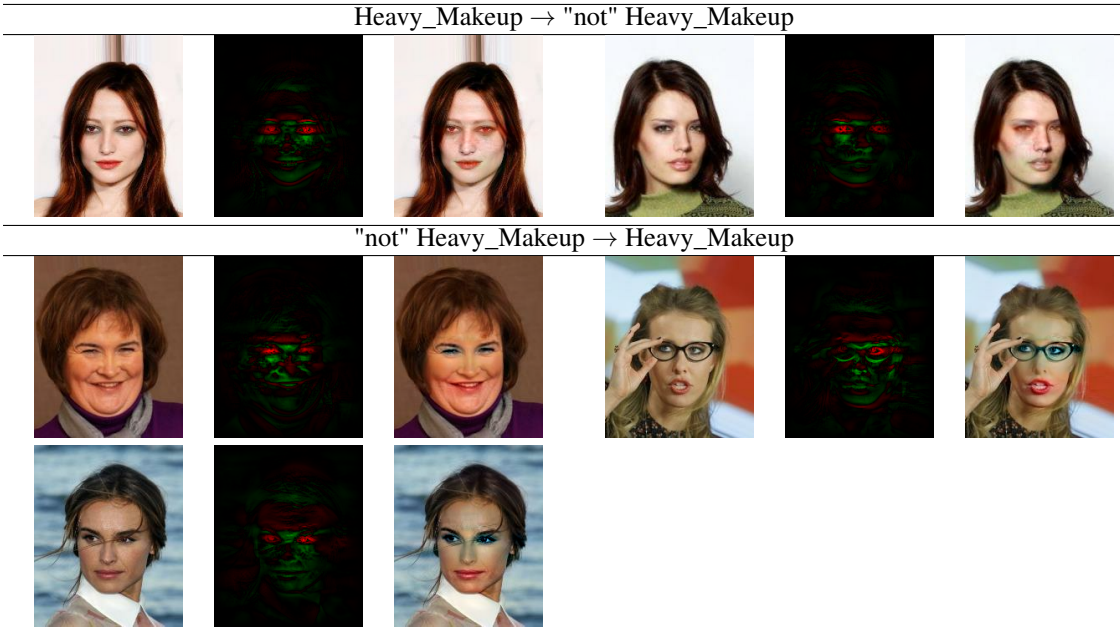


Figure 21: Samples from label Heavy_Makeup

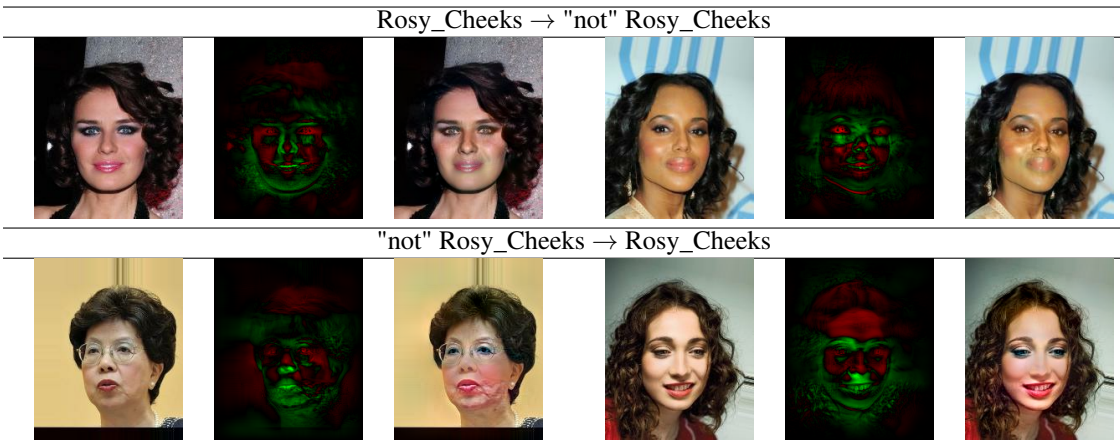


Figure 22: Samples from label Rosy_Cheeks

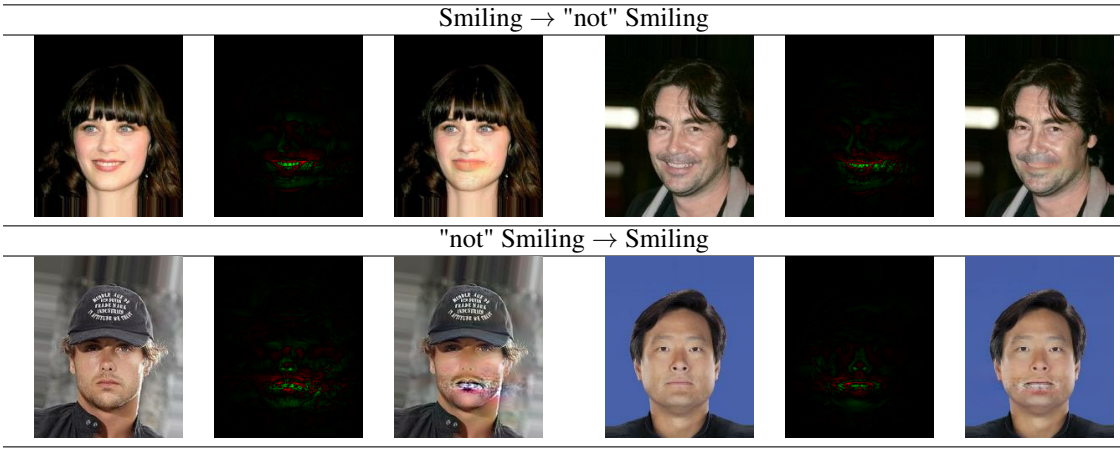


Figure 23: Samples from label Smiling

Figure 24: Samples from label Wearing_Lipstick

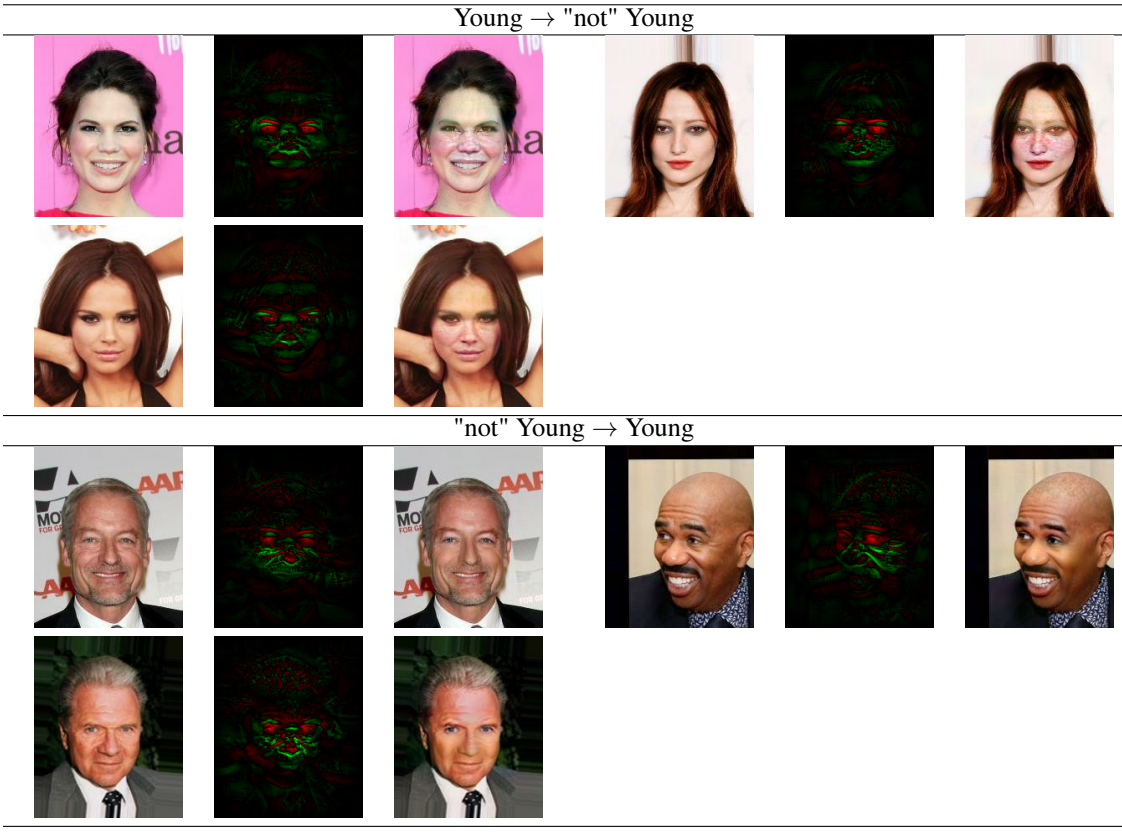


Figure 25: Samples from label Young