Text-to-Image Models for Counterfactual Explanations: a Black-Box Approach

Guillaume Jeanneret, Loïc Simon, Frédéric Jurie

Text-to-Image Models for Counterfactual Explanations: a Black-Box Approach

Guillaume Jeanneret, Loïc Simon, Frédéric Jurie
Normandy University, ENSICAEN, UNICAEN, CNRS, GREYC, France
guillaume.jeanneret-sanmiguel@unicaen.fr

Abstract

*This paper addresses the challenge of generating Counterfactual Explanations (CEs), involving the identification and modification of the fewest necessary features to alter a classifier's prediction for a given image. Our proposed method, **T**ext-to-**I**mage **M**odels for Counterfactual **E**xplanations (TIME), is a black-box counterfactual technique based on distillation. Unlike previous methods, this approach requires solely the image and its prediction, omitting the need for the classifier's structure, parameters, or gradients. Before generating the counterfactuals, TIME introduces two distinct biases into Stable Diffusion in the form of textual embeddings: the context bias, associated with the image's structure, and the class bias, linked to class-specific features learned by the target classifier. After learning these biases, we find the optimal latent code applying the classifier's predicted class token and regenerate the image using the target embedding as conditioning, producing the counterfactual explanation. Extensive empirical studies validate that TIME can generate explanations of comparable effectiveness even when operating within a black-box setting.*

1. Introduction

Recently, deep neural networks (DNN) have seen increased attention for their impressive forecasting abilities. The use of deep learning in critical applications, such as driving automation, made the scientific community increasingly involved in what a model is learning and how it makes its predictions. These concerns shed light on the field of Explainable Artificial Intelligence (XAI) in an attempt to "open the black-box" and decipher its induced biases.

Counterfactual explanations (CEs) are an attempt to find an answer to this previous problem. They try answering the following question: *What do we need to change in X to change the prediction from Y to Z*? Because CEs give intuitive feedback about what to change to get the desired result, two applications use these explanations: feedback recommendation systems and debugging tools. Take an automated loan approval system as an example. From a user's point of view, if it gets a negative prediction, the user would be more interested in knowing what plausible changes can be made to get a positive result, rather than having an exhaustive list of explanations for why the result is unfavorable. From the debugger's point of view, it can look for biases that were considered in the decision when they should not have been, thus revealing the classifier's weaknesses.

Method	Model	Training	Specificity	Optim.
DiVE [39]	VAE	Days	Only DNN	Yes
STEEX [23]	GAN	Days	Only DNN	Yes
DiME [24]	DDPM	Days	Only DNN	Yes
ACE [25]	DDPM	Days	Only DNN	Yes
TIME (Ours)	T2I	Hours	Black-Box	No

Table 1. **Advantages of the proposed methodology.** TIME uses a pre-trained T2I model and trains only a few textual embeddings, requiring hours of training instead of days. It does not require access to the target model (completely black-box) and does not involve any optimization during counterfactual generation.

While there are multiple ways to address this question for visual systems, *e.g.* by adding adversarial noise [16], the modifications must be sparse and comprehensive to provide insight into which variables the model is using. To this end, most studies for CEs use generative models, such as GANs [15], Denoising Diffusion Probabilistic Models (DDPMs) [19], or VAEs [30], as they provide an intuitive interface to approximate the image manifold and constrain the generation in an appropriate space. Although they have several advantages, training these generative models is cumbersome and may not yield adequate results, especially when the data is limited [27]. To this end, we expect that the use of large generative models trained on colossal datasets, such as LAION-5B [43], can provide a sufficient tool to generate CEs. On the one hand, these generative models have shown remarkable qualitative performance, an attractive feature to exploit. Second, since the generative model is already optimized, it can be used to capture data set specific concepts - *e.g.* textual inversion [12] captures the main aspects of a target object when subject to only three to five images.

In this paper, we explore how to take advantage of Text-

to-Image (T2I) generative models for CEs - specifically, using Stable Diffusion [10]. To do so, we take a distillation approach to transfer the learned information from the model into new text embeddings to align the concept class in text space. Second, we use inversion techniques [49] to find the optimal noise to recover the original instance. Finally, with our distilled knowledge, we denoise this optimal point to recover the final instance using the target label, thus generating the CE. This is advantageous because we can tackle the challenging scenario of explaining a black-box model, *i.e*. having access only to its predictions.

Our proposed approach has three main advantages over previous literature, as shown in Table 1. First, we only train some textual embeddings, making the training efficient, while previous methods require training a generative model from scratch. Second, we do not require an optimization loop when generating the final counterfactual, which reduces the generation time. Finally, our explainability tool works in a completely black-box environment. While most modern approaches [23–25, 39, 51] are DNN-specific, because they rely on gradients, our approach, which uses only the output and input as cues, can be used to diagnose any model regardless of its internal functioning. This setting is crucial for privacy-preserving applications, such as medical data analysis, since eliminating access to the gradients could prevent data leakage [54], as it helps protect personal or confidential information.

We summarize our contributions as follows[1]:

- We propose TIME: Text-to-Image Models for Counterfactual Explanations, using Stable Diffusion [40] T2I generative model to generate CEs.
- Our proposed approach is completely black-box.
- Our counterfactual explanation method based on a distillation approach does not require any optimization during inference, unlike most methods.
- From a quantitative perspective, we achieve similar performance to the previous state-of-the-art, while having access only to the input and the prediction of the target classifier.

2. Related Work

2.1. Explainable Artificial Intelligence

The research branch of XAI broads multiple ways to provide insights into what a model is learning. As a bird's view analysis, there are two main distinctions between methods: *Interpretable by-design* architectures, and *Post-Hoc* explainability methods. The former searches to create algorithms that directly expose why a decision was made [2, 3, 5, 8, 22, 35, 53]. Our research study is based on the latter. *Post-hoc* explainability methods study pretrained models and try to decipher the variables used for forecasting. Along these lines, there are saliency maps [4,26,37,44], concept attribution [11, 14, 29], or distillation approaches into interpretable by-design models [13]. In this paper, we study the on-growing branch of CEs [48]. In contrast to previous methods, these explanations are simpler and more aligned with human understanding, making them appealing to comprehend machine learning models.

2.2. Counterfactual Explanations

The seminal work of Watcher *et al*. [48] defined what a counterfactual explanation is and proposed to find them as a minimization problem between a classification loss and a distance loss. In the image domain, optimizing the image's raw pixels produces adversarial noises [16]. So, many studies based their work on Watcher *et al*. [48]'s optimization procedure with a generative model to regularize the CE production, such as variational autoencoders [39], generative adversarial networks [23, 28, 32, 45, 51], and diffusion models [1, 24, 25, 42]. In contrast to these works, our proposed approach, TIME, is a distillation approach for counterfactuals. Our method does not require any optimization loop when building the explanation, since we transfer the learning into the T2I model. Furthermore, we do not require access to the gradients of the target model but only the input and output, making it black-box, unlike previous methods.

Co-occurent works analyze dataset biases using T2I models to create distributional shifts in data [38, 47]. Although a valid approach to debug datasets, we argue that these approaches do not search what a model learned but instead a general strategy for the biases in datasets under distributional shifts (*e.g*. it is normal to misclassify a dog with glasses since the model was not trained to classify dog with glasses). Further, their proposed approaches are computationally heavy, since they require fine-tuning Large Language Models or optimizing each inversion step on top of Stable Diffusion. Instead, ours requires training a word embedding, and the inference merely requires Stable Diffusion without computing any gradients, which fits into a single small GPU.

2.3. Customization with Text-to-Image Models

Due to the interest in creating unimaginable scenarios with personalized objects, customizing T2I diffusion models has gained attention in recent literature. Textual Inversion [12] and following works [7, 17, 34, 41, 52] are popular approaches to learn to generate specific objects or styles by fine-tuning all or some part of the T2I model. Thus, the new concept can be used in a phrase such that the T2I model will synthesize it.

[1]Code is available at https://github.com/guillaumejs2403/TIME

One of the most difficult problems is editing real-world images with T2I models. The pioneer work of Song *et al*. [46] proposed a non-stochastic variant of DDPMs, called Denoising Diffusion Implicit Models (DDIM). Hence, a single noise seed yields the same image. So, to find an approximate noise, DDIM Inversion noises the image using the diffusion model. Yet, some problems arise with this approximation. So, novel works [33, 36] modify the inversion process by including an inner gradient-based optimization at each noising step, making it unfeasible when analyzing a bundle of images. Finally, Wallace *et al*. [49] proposed to modify the DDIM algorithm into a two-stream diffusion process, reaching a "perfect" inversion. We take advantage of these works and distill the learned information from a classifier to generate counterfactual explanations of real images, a step to interpret the target classifier.

3. Methodology

This section explains the proposed methodology for generating counterfactuals using T2I generative models. In section 3.1, we briefly introduce some useful preliminary concepts of diffusion models. Then we describe our proposed method in a three-step procedure. First, we explain how to transfer what the classifier has learned into the generative model as a set of new text tokens (Section 3.2). Second, using recent advances in DDIM Inversion, we revert the image to its noise representation using the original prediction of the classifier. Finally, we denoise the noisy latent instance using the target label (Section 3.3).

3.1. DDPM Preliminaries

Diffusion models [19] are generative architectures that create images by iteratively *removing* noise. DDPMs are based on two inverse Markov chains. The forward chain *adds* noise, while the reverse chain *removes* it. Thus, the generation process is reverse denoising, starting from a random Gaussian variable and removing small amounts of noise until a plausible image is returned.

Formally, given a diffusion model ϵ_θ and a fixed set of steps T, ϵ_θ takes as input a noisy image x_t, the current step t to compute a residual shift, and a textual conditioning C, in our case. For the generation, ϵ_θ updates x_t following:

$$x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t, C)\right) + \sigma_t\epsilon, \quad (1)$$

where σ_t, α_t and $\bar{\alpha}_t$ are some predefined constants, and ϵ and x_T are extracted from a Gaussian distribution. This process is repeated until $t = 0$. To train a DDPM, for a given an image-text pair (x, C), each optimization step minimizes the loss:

$$L(x, \epsilon, t, C) = \|\epsilon - \epsilon_\theta(x_t(x, t, \epsilon),\ t, C)\|^2, \quad (2)$$

with

$$x_t(x, t, \epsilon) = \sqrt{\bar{\alpha}_t}\, x + \sqrt{1-\bar{\alpha}_t}\, \epsilon. \quad (3)$$

The pioneering work of Ho *et al*. [19] focused on training and evaluating these models in the pixel space, making them computationally heavy. Latent Diffusion Models [40] proposed to reduce this burden by performing the diffusion process in the latent space of a Quantized Autoencoder [10]. Further, they augment the generation by using textual conditioning C at its core to steer the diffusion process, as well as increasing the quality of the generation using Classifier-Free Guidance [20] (CFG).

The CFG [20]'s core modifies the sampling strategy in Eq. 1 by replacing ϵ_θ with ϵ^f_θ, a shifted version defined as follows:

$$\epsilon^f_\theta(x_t, t, C) := (1+w)\,\epsilon_\theta(x_t, t, C) - w\,\epsilon_\theta(x_t, t, \varnothing), \quad (4)$$

where $\varnothing$ is the empty conditioning and w is a weighting constant, resulting in a qualitative improvement.

3.2. Distilling Knowledge into Stable Diffusion

To use large generative models, and in particular Stable Diffusion [40], we chose to distill the learned biases of the target classifier into the generative model to avoid any gradient-based optimization during the CE formation.

A model is subject to several biases as it learns, of which we distinguish two. The first is a *context bias*. This bias refers to the way images are formed. For example, ImageNet images [6] tend to have the object (*e.g*., animals, cars, bridges) in the center, while CelebA HQ images [31] are human faces. The second bias is class-specific, and it relates to the semantic cues extracted by the classifier to make its decision, *e.g*. white and black stripes for a zebra.

So, we take a textual inversion approach to distill the context bias and the knowledge of the target classifier into the textual embedding space of Stable Diffusion. In a nutshell, textual inversion [12] links a new text-code c^* and an object (or style) such that when this new code is used, the generative model will generate this new concept. To achieve this, Gal *et al*. [12] proposed to instantiate a new text embedding e^*, associate it to the new text-code c^*, and then train e^* by minimizing the loss

$$\mathbb{E}_{(x,C)\sim D, t\sim U[1,T], \epsilon\sim\mathcal{N}(0,I)}\left[L(x, t, \epsilon, C)\right]. \quad (5)$$

Here, D is the set of images containing the concept to be learned, U is the uniform distribution of natural numbers between 1 and T, and C is a text prompt containing the new text code c^*.

Accordingly, to distill the context bias into Stable Diffusion, we follow [12] practices and learn a new textual embedding $e^*_{context}$ minimizing Eq. 5 using as the conditioning the phrase `A` $c^*_{context}$ `picture`. Here, $c^*_{context}$ is

the textual code related to textual embedding $e^*_{context}$. In our setup, we used the complete training set of images with no labels where the model was trained.

So far, we have not been required to use the classifier. To transfer the knowledge learned by the classifier to the T2I generation pipeline, we follow a similar approach. In this case, we train a new textual embedding e^*_i for each class i and represent its text token with c^*_i. However, instead of using the full training dataset D, we used only those images that the classifier predicted to be the source class i. As for the conditioning sentence, we take the previously learned context token and add the new class token to the sentence. Thus, we optimize Eq. 5 with the new phrase `A` $c^*_{context}$ `image with a` c^*_i and the filtered dataset. For the rest of the text, we will refer to this prompt as C_i.

3.3. Counterfactual Explanations Generation

Now we want to use the learned embeddings to generate explanations. Current research on diffusion models has attempted to recover input images by retrieving the best noise, such that when the DDIM sampling strategy is used, it generates the initial instance. This is advantageous for our goal, since we can use current technological advances to generate this optimal latent noise and then inpaint the changes necessary to flip the classifier.

Since we need to perform perfect recovery to avoid most changes in the input image, we use EDICT [49]'s perfect inversion technique. In fact, they showed that inverting an image with a caption (Eqs 8) and then denoising it (Eqs. 7) with a modified version of the original caption will produce semantic changes in the image. In short, EDICT modifies the DDIM [46] sampling strategy for diffusion models into a two-flow invertible sequence. By introducing a new hyperparameter $0 < p < 1$, setting x_0 and y_0 as the target image, and new variables:

$$\begin{aligned} a_t &= \sqrt{\bar{\alpha}_{t-1}/\bar{\alpha}_t} \\ b_t &= \sqrt{1-\bar{\alpha}_{t-1}} - \sqrt{\bar{\alpha}_{t-1}(1-\bar{\alpha}_t)/\bar{\alpha}_t}, \end{aligned} \tag{6}$$

the denoising phase becomes:

$$\begin{aligned} x^{inter}_t &= a_t\, x_t + b_t\, \epsilon^f_\theta(y_t, t, C) \\ y^{inter}_t &= a_t\, y_t + b_t\, \epsilon^f_\theta(x^{inter}_t, t, C) \\ x_{t-1} &= p\, x^{inter}_t + (1-p)\, y^{inter}_t \\ y_{t-1} &= p\, y^{inter}_t + (1-p)\, x_{t-1}. \end{aligned} \tag{7}$$

In a similar vein, the inversion phase is the inverse of Eqs. 7:

$$\begin{aligned} y^{inter}_{t+1} &= (y_t - (1-p)\, x_t) \;/\; p \\ x^{inter}_{t+1} &= (x_t - (1-p)\, y^{inter}_{t+1}) \;/\; p \\ y_{t+1} &= \frac{1}{a_{t+1}}(y^{inter}_{t+1} - b_{t+1}\, \epsilon^f_\theta(x^{inter}_{t+1}, t+1, C)) \\ x_{t+1} &= \frac{1}{a_{t+1}}(x^{inter}_{t+1} - b_{t+1}\, \epsilon^f_\theta(y^{inter}_{t+1}, t+1, C)). \end{aligned} \tag{8}$$

We can see a clear connection between Wallace *et al*. [49]'s work and our main objective. If we invert an image using the caption with our context and source class tokens and then denoise it by changing the prompt to include the target token (learned in Section 3.2), we can hope to generate the necessary changes to flip the classifier's decision.

However, while adapting the EDICT method, we noticed a major problem with this approach. Although the chosen algorithm recovers the input instance, many images were difficult to modify. To circumvent this issue, we had to adjust the scores of the CFG in Eq. 4. As diffusion models are seen as score-matching models, the term

$$w(\epsilon_\theta(x_t, t, C_i) - \epsilon_\theta(x_t, t, \varnothing)) \tag{9}$$

in Eq. 4 are gradients pointing to the target distribution conditioned on C_i. We call this the positive drift. Thus, by including a negative drift term,

$$-w(\epsilon_\theta(x_t, t, C_j) - \epsilon_\theta(x_t, t, \varnothing)), \tag{10}$$

we can lead the generation process *away* from the source distribution conditioned in C_j. Therefore, we reformulate the CFG scores ϵ^f_θ, and rename it to ϵ^c_θ, as follows:

$$\begin{aligned} \epsilon^c_\theta(x_t, t, C_i, C_j) = (1+w)\, \epsilon_\theta(x_t, t, C_i) \\ - w\, \epsilon_\theta(x_t, t, C_j). \end{aligned} \tag{11}$$

As a result, and given the previously introduced notions, we propose **T**ext-to-**I**mage **M**odels for counterfactual **E**xplanations (TIME), illustrated in Figure 1. To leverage these big generative models, we first distill the context bias into the pipeline's text embedding space by training a text embedding with the complete dataset. Then, we transfer the knowledge of the classifier by training a new embedding but using solely the instances with the same predictions. Finally, given an input image classified as i and the target j, we invert the image (Eqs. 8) using ϵ^c_θ as the score network (Eq. 11) using as the positive and negative drift C_i and C_j, respectively. Then, we denoise the noisy state using Eqs. 7 but switching textual conditionings.

Practical considerations. To avoid large changes in the image, the inversion stops at an intermediate step τ instead

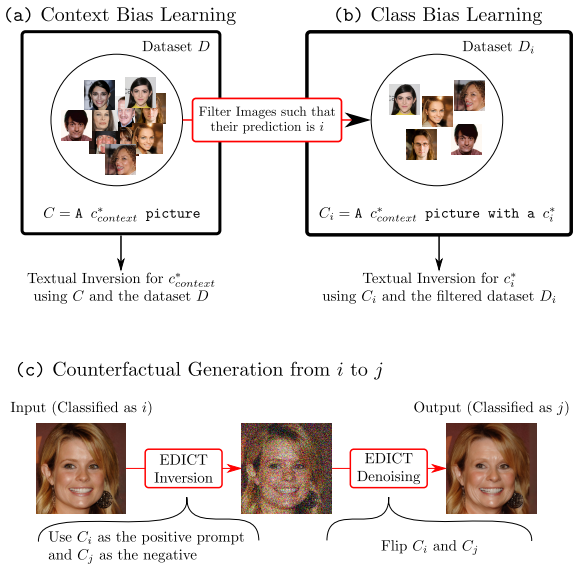

Figure 1. **TIME Overview.** Our proposed method consists of three steps: (a) We learn a context token for the whole dataset using textual inversion. (b) We filter out the images that the classifier predicts as source class i and learn a new embedding. (c) Finally, to generate the counterfactual explanation, we invert the input image using a prompt containing the source embedding and then denoise it using the target embedding.

of T. In addition, we have found that using more than a single embedding for the context and class biases yield further expressiveness. Also, if we fail to find a valid counterfactual, we choose a new τ and w to rerun the algorithm. We will give the implementation details later in Section 4.

4. Experimental Validation

Datasets and Models. We evaluate our counterfactual method in the popular dataset CelebAHQ [31]. The task at hand is classifying smile and age attributes from face instances, computed with a DenseNet121 [21] with an image resolution of 256×256 as in [23, 25]. The evaluation is performed on the test set. To make the assessment fair with previous methods, we used the publicly available classifiers for CelebA HQ dataset from previous studies [23].

Implementation Details. We based our approach on Stable Diffusion V1.4 [10]. For all dataset, we trained three textual embeddings for the context and class biases for 800 iterations with a learning rate of 0.01, a weight decay of 1e-4, and a batch size of 64. For the inference, we used the default EDICT's hyperparameter $p = 0.93$ and a total of 50 steps. For the smiling attribute, we begin the CE generation with $(\tau, w) = (25, 3)$. In case of failure, we increased the tuple to $(30, 4), (35, 4)$ or $(35, 6)$. For the age attribute, we used $(\tau, w) \in \{(30, 4), (30, 6), (35, 4), (35, 6)\}$. We performed all training and inference in a Nvidia GTX 1080.

4.1. Quantitative Assessment

Assessing counterfactuals presents inherent challenges. Despite this, several metrics approximate the core objectives of counterfactual analysis. We will now provide a concise overview of each objective and its frequent evaluation protocol, reserving an in-depth exploration of these metrics for the supplementary material.

Validity. First, we need to quantify the ability of the counterfactual explanation method to flip the classifier. This is measured by the Success Ratio (SR aka Flip Rate).

Sparsity and Proximity. A counterfactual must have sparse and proximal editions. Several metrics have been proposed to evaluate this aspect, depending on the data type. For face images [24, 25, 39, 45], there are the face verification accuracy (FVA), face similarity (FS), mean number of attributes changed (MNAC), and Correlation Difference (CD). For general-purpose images, like BDD100k [50], the quantitative assessment is done via the SimSiam Similarity (S^3) [25] and the COUT metric [28].

Realism. The CE research adapts its evaluation metrics from the generation field. Hence, the realism of CEs is commonly measured with the FID [18] and sFID [25] metrics but only in the correctly classified images.

Efficiency. An efficiency analysis is often omitted by many methods. A crucial criterion for counterfactual generation techniques is to minimize computation time for generating explanations in "real time". We evaluate this by contrasting efficiency using floating point operations (FLOPs) per explanation - lower values signify faster inference - and by measuring the average time taken to generate an explanation, specifically within our cluster environment.

4.1.1 Main Results.

Table 2 shows the results of TIME and compares them to the previous literature. Although we do not outperform the state-of-the-art in any metric, we found that our results are similar even when our proposed method is restricted to be black-box. Further, it does not require training of a completely new generative model and does not rely on any optimization for CE generation. For the realism metric, we expected to get a low FID [18] and sFID [25] due to the use of Stable Diffusion and beat ACE [25]. However, ACE uses an inpainting strategy to post-process their counterfactuals. This reduces this metric because they keep most of the original pixels in their output. If we remove the post-processing,

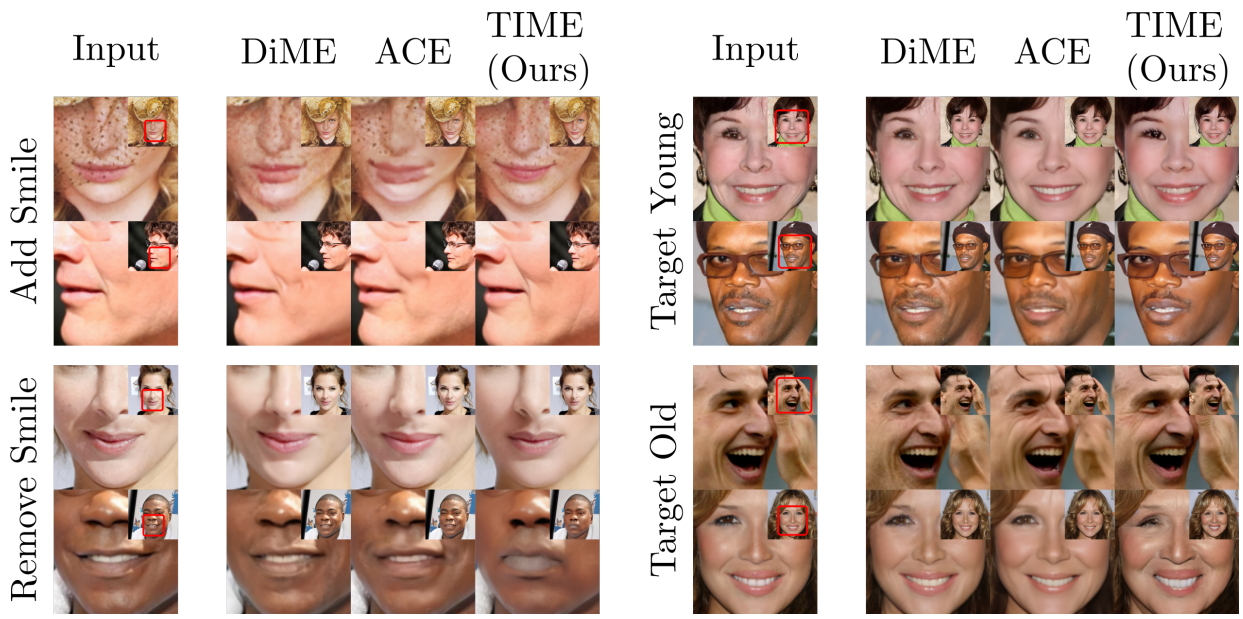

Figure 2. **Qualitative Results.** We present qualitative examples and compare them to the previous state of the art. DiME generates some out-of-distribution noise, while ACE creates blurry image sections. In contrast, TIME produces more realistic changes by harnessing the generative power of the T2I model.

Method	Smile								Age							
	FID (↓)	sFID (↓)	FVA (↑)	FS (↑)	MNAC (↓)	CD (↓)	COUT (↑)	SR (↑)	FID (↓)	sFID (↓)	FVA (↑)	FS (↑)	MNAC (↓)	CD (↓)	COUT (↑)	SR (↑)
DiVE [39]	107.0	-	35.7	-	7.41	-	-	-	107.5	-	32.3	-	6.76	-	-	-
STEEX [23]	21.9	-	97.6	-	5.27	-	-	-	26.8	-	96.0	-	5.63	-	-	-
DiME [24]	18.1	27.7	96.7	0.6729	2.63	1.82	0.6495	97.0	18.7	27.8	95.0	0.6597	2.10	4.29	0.5615	97.0
ACE* ℓ_1 [25]	26.1	36.8	99.9	0.8020	2.33	2.49	0.4716	95.7	24.6	38.0	99.6	0.7680	1.95	4.61	0.4550	98.7
ACE ℓ_1 [25]	3.21	20.2	100.0	0.8941	1.56	2.61	0.5496	95.0	5.31	21.7	99.6	0.8085	1.53	5.4	0.3984	95.0
ACE* ℓ_2 [25]	26.0	35.2	99.9	0.8010	2.39	2.40	0.5048	97.9	24.2	34.9	99.4	0.7690	2.02	4.29	0.5332	99.7
ACE ℓ_2 [25]	6.93	22.0	100.0	0.8440	1.87	2.21	0.5946	95.0	16.4	28.2	99.6	0.7743	1.92	4.21	0.5303	95.0
TIME (Ours)	10.98	23.8	96.6	0.7896	2.97	2.32	0.6303	97.1	20.9	32.9	79.3	0.6282	4.19	4.29	0.3124	89.9

Table 2. **CelebAHQ Evaluation.** While TIME does not outperform the state-of-the-art metrics, our proposed method provides competitive performance while being completely black-box, *i.e.* having access only to the input and output of the model. ACE* is [25]'s method without their post-processing method.

the FID increases dramatically. With these results, we confirm that T2I generative models are a good tool to explain classifiers counterfactually in a black-box environment.

4.1.2 Qualitative Results

We show some qualitative results in Figure 2 and added more instances in the supplementary material. First, we see that DiME [24], ACE [25], and TIME generate very realistic counterfactuals, and the differences are mostly in the details. However, the most notable changes are between ACE and our method. When we check the regions where ACE made the changes, they are blurred. This is due to their over-respacing to create the counterfactual. For DiME, we checked and found that some of their modifications seem out-of-distribution, for many cases. However, TIME produces realistic changes most of the time. Finally, in our opinion, TIME alterations can be spotted with more ease.

4.1.3 Efficiency Analysis

We continue our analysis and study the efficiency of TIME when creating the CE with respect to previous state-of-the-art methods, DiME [24] and ACE [25]. We estimated that TIME uses 98 TFLOPs and 45 seconds to create a single counterfactual, using $\tau = 35$ as the worst case scenario. In contrast, ACE took 279 TFLOPs and 62 second per CE while DiME took 1004 TFLOPs and 163 seconds.

4.2. Ablations

To show the effectiveness of each component, we realized thorough ablation experiments. To this end, we first show the hyperparameter exploration between the depth of the chain of noise τ and the guidance scale w. Additionally, we will show the effect of including multiple textual tokens, the context tokens, and, finally, the effect of adding our negative drift – please refer to the practical consideration in section 3.3 for the variable τ. Unless explicitly

Steps	GS	SR (↑)	FID (↓)	FS (↑)	CD (↓)
25	3	30.1	35.26	0.8957	2.82
	4	41.0	30.23	0.8570	2.61
	5	50.1	27.39	0.8231	2.33
30	3	62.1	23.15	0.8147	2.34
	4	74.0	22.51	0.7710	2.66
	5	80.8	23.51	0.7300	2.85
35	3	87.1	21.69	0.7227	2.63
	4	92.9	24.37	0.6731	3.03
	5	95.0	27.53	0.6306	3.54

Table 3. **Steps-Scale trade-off.** We analyze the trade-off between our hyperparameters τ and w. Our results show that increasing τ gives a strong boost in SR while impacting the other metrics and increasing the generation time. In contrast, w has a similar effect but is less potent without any effect on the generation time.

Context	SR (↑)	FID (↓)	FS (↑)	CD (↓)
Without	73.9	23.47	0.7480	2.41
With	92.9	24.37	0.6731	3.03

Table 4. **Context token ablation**. Here, we check the effect of including the context embeddings into our pipeline. The main advantage is increasing the success ratio. This result suggests that we can reduce τ to reach similar results while being more efficient - less number of EDICT iterations.

told, we set $\tau = 35$ and $w = 4$ for all the ablations. For the dataset, we did the ablation using 1000 instances of the CelebA HQ validation dataset for the smiling attribute. As the quantitative metrics, we used the SR, the FID, the FS, and the CD.

Regarding the FID metric, please note that this metric is very sensible to the number of images. When using fewer images, the FID becomes less reliable to compare two methods, and hardly becomes intelligible if the two approaches are evaluated on different number of images. Since we use the FID to compare counterfactual on only those instances that flipped the classifier, comparing FIDs where the SR varies significantly does not give any cues.

Steps and scale trade-off. To begin with, we investigate the effect of the number inversion steps and the scale of the guidance. We jointly explore both variables to check the best trade-off, as shown in Table 3. At first glance, we notice that adding a higher guidance scale or more noise inversion steps produces more successful counterfactuals, assessed with the SR. Yet, it comes with a trade-off in other compartments: namely, the quality of the CE, and the amount of editions into the image. Generally, increasing τ or w reflects a decrease in the quality of the image and the increasing numbers of editions.

Guidance	SR (↑)	FID (↓)	FS (↑)	CD (↓)
CFG	75.9	21.58	0.7749	2.34
NG	92.9	24.37	0.6731	3.03

Table 5. **Negative Guidance**. Here, we check the effect of performing the negative guidance (NG) instead of the classifier-free guidance (CFG). The main advantage is increasing the success ratio. This result suggests that we can reduce τ to reach similar results while being more efficient.

Tokens	SR (↑)	FID (↓)	FS (↑)	CD (↓)
Single	88.1	22.02	0.7177	3.02
Multiple	92.9	24.37	0.6731	3.03

Table 6. **Multiple-tokens Ablation.** We test if using multiple tokens in our pipeline provides any advantage. The results show an increase in SR.

Learning the Context Token. Continuing with our study, we analyze the inclusion of our novel context token into our counterfactual generation pipeline. To ablate this component, we test whether using our learned context tokens has any advantage in contrast to giving a generic description. The results are in Table 4. As we can see, including our tokens provides the best performance gains in terms of SR. Qualitatively, the images are similar, yet, the images without context present some artifacts in some cases. Furthermore, we see that removing the context provides a boost in the CD and FS metrics. Although it seems counterintuitive to include this component, we can easily reach these values by decreasing τ or w (*e.g.* setting $\tau = 30$ and $w = 4$, check Table 3), and reducing the inference time.

Effect of the guidance. We further explore the inclusion of the negative drift term in Eq. 10 and show the results in Table 5. From the quantitative assessment, we initially observed that using the classifier-free guidance (CFG in the Table) decreases the SR. When denoising the current stage x_t at time t, the CFG in Eq. 4 estimates gradients of the log-likelihood conditioned on C_j, $-\nabla_{x_t} \log(p(x_t|C_j))$, [20] thus, pushing the generation *toward* the distribution of C_j. In contrast, incorporating the negative guidance (NG) helps steer the generation *away* from the distribution conditioned on C_i. Therefore, the combined effect results in moving the instance from the boundary decision. From a qualitative perspective, we did not see major differences.Nonetheless, as noted in the context of ablation, this can be easily mitigated by reducing w and τ.

Multi-token Inclusion. Finally, we explore using multiple tokens instead of a single one for both the context and class embeddings, shown in Table 6. Without any sur-

Figure 3. **BDD100k, a limit for TIME.** TIME changes the entire scene when generating the counterfactuals. Nevertheless, it still gives some insight into what the models have learned, as illustrated by the features inside the red boxes.

prise, we noticed that using a single token reduces the SR by a small factor. This aligns with the observations given by [12], a token catches enough information of an object or style - or in this case, inductive biases. Like in previous analyses, including multiple tokens will increase the efficiency of the model, since we can reach similar performances by tuning τ or w. Qualitatively, the most notable change between the images is sharpness.

Recommendations. Given the previous results, we propose several recommendations for the user and the model debugger, as explained in the introduction. Recall that the counterfactual explanations are used as well to recommend changes to the user to get a positive outcome. So, for the user, we recommend using the lower amount of iterations τ and guidance scale w. This results in a similarity increase and fewer edited characteristics (as evidenced by the CD and FS metrics). If the algorithm fails, it is preferable to adjust the guidance scale rather than the number of steps. For the debugger, always use the context, the negative guidance, and multiple tokens. When building the counterfactuals, follow the same recommendations for the user.

4.3. Limitations.

To test TIME in more complex scenarios, we generate CEs in the BDD100k [50] dataset using a DenseNet121 [21] trained in a *move-forward/stop* binary classification, as in [23]. We show the quantitative evaluation in Table 7. When generating the explanations, we noticed that TIME modifies most parts of the image, unfortunately, as shown by the S^3 metric. This is expected, as this task is challenging since it requires multiple factors to decide if to stop or to move forward. Nevertheless, we believe that these explanations still give some useful insights as a debugging tool. For example, Figure 3 shows that removing the red lights and adding motion blur will change the classification from *stop* to *move*, as evidenced in [25], or adding objects in front will flip the prediction to *stop*.

Method	FID (↓)	sFID (↓)	S^3 (↑)	COUT (↑)	SR (↑)
STEEX	58.8	-	-	-	99.5
DiME	7.94	11.40	0.9463	0.2435	90.5
ACE ℓ_1	1.02	6.25	0.9970	0.7451	99.9
ACE ℓ_2	1.56	6.53	0.9946	0.7875	99.9
TIME (Ours)	51.5	76.18	0.7651	0.1490	81.8

Table 7. **BDD Assessement.** We evaluate the performance of TIME on the complex BDD100k benchmark. On this dataset, there is still room for improvement for black-box counterfactual methods.

We believe that counterfactual methods for tasks dependent on complex scenes, where the decision is impacted by large objects or co-occurrences of several stimuli, require specific architectures. In fact, we noticed that ACE [25] mainly adds some small modifications (*e.g.* changing the red lights), which is not inaccurate but is too constrained and cannot explore more insights about the learned features. Indeed, the work of Zemni *et al.* [51] focuses only on the object aspect of counterfactuals, in this case using an object-centric generator, BlobGAN [9]. This suggests that general-purpose counterfactual methods are not adapted for these tasks.

5. Conclusion

In this work, we present TIME, a counterfactual generation method to analyze classifiers disregarding their architecture and weights, only by looking at their inputs and outputs. By leveraging T2I generative models and a distillation approach, our method is capable of producing CEs for black-box models, a complex scenario not tackled before. Further, we show the advantages and limitations of TIME and shed light on possible future works. We believe that our approach opens the door to research focus on counterfactual methods in the challenging scenario of the black-box models.

Acknowledgements Research reported in this publication was supported by the Agence Nationale pour la Recherche (ANR) under award number ANR-19-CHIA-0017.

References

[1] Maximilian Augustin, Valentyn Boreiko, Francesco Croce, and Matthias Hein. Diffusion visual counterfactual explanations. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022. 2

[2] Moritz Bohle, Mario Fritz, and Bernt Schiele. Convolutional dynamic alignment networks for interpretable classifications. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10029–10038, June 2021. 2

[3] Moritz Böhle, Mario Fritz, and Bernt Schiele. B-cos networks: Alignment is all we need for interpretability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10329–10338, June 2022. 2

[4] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 839–847, 2018. 2

[5] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: Deep learning for interpretable image recognition. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[7] Wenkai Dong, Song Xue, Xiaoyue Duan, and Shumin Han. Prompt tuning inversion for text-driven image editing using diffusion models. *arXiv preprint arXiv:2305.04441*, 2023. 2

[8] Jon Donnelly, Alina Jade Barnett, and Chaofan Chen. Deformable ProtoPNet: An Interpretable Image Classifier Using Deformable Prototypes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10265–10275, 2022. 2

[9] Dave Epstein, Taesung Park, Richard Zhang, Eli Shechtman, and Alexei A. Efros. Blobgan: Spatially disentangled scene representations. *European Conference on Computer Vision (ECCV)*, 2022. 8

[10] Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12873–12883, June 2021. 2, 3, 5

[11] Thomas Fel, Agustin Picard, Louis Bethune, Thibaut Boissin, David Vigouroux, Julien Colin, Rémi Cadène, and Thomas Serre. Craft: Concept recursive activation factorization for explainability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2711–2721, 2023. 2

[12] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 2, 3, 8

[13] Yunhao Ge, Yao Xiao, Zhi Xu, Meng Zheng, Srikrishna Karanam, Terrence Chen, Laurent Itti, and Ziyan Wu. A peek into the reasoning of neural networks: Interpreting with structural visual concepts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2195–2204, 2021. 2

[14] Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. Towards automatic concept-based explanations. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. 1

[16] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. 1, 2, 12

[17] Jing Gu, Yilin Wang, Nanxuan Zhao, Tsu-Jui Fu, Wei Xiong, Qing Liu, Zhifei Zhang, He Zhang, Jianming Zhang, HyunJoon Jung, et al. Photoswap: Personalized subject swapping in images. *arXiv preprint arXiv:2305.18286*, 2023. 2

[18] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 5, 12

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 3

[20] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 3, 7

[21] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017. 5, 8

[22] Zixuan Huang and Yin Li. Interpretable and accurate fine-grained recognition via region grouping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[23] Paul Jacob, Éloi Zablocki, Hédi Ben-Younes, Mickaël Chen, Patrick Pérez, and Matthieu Cord. Steex: steering counterfactual explanations with semantics. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XII*, pages 387–403. Springer, 2022. 1, 2, 5, 6, 8, 12

[24] Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. Diffusion models for counterfactual explanations. In *Proceedings*

of the Asian Conference on Computer Vision, pages 858–876, 2022. 1, 2, 5, 6, 13

[25] Guillaume Jeanneret, Loïc Simon, and Frédéric Jurie. Adversarial counterfactual visual explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16425–16435, 2023. 1, 2, 5, 6, 8, 12, 13

[26] Hyungsik Jung and Youngrock Oh. Towards better explanations of class activation mapping. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1336–1344, October 2021. 2

[27] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. *Advances in neural information processing systems*, 33:12104–12114, 2020. 1

[28] Saeed Khorram and Li Fuxin. Cycle-consistent counterfactuals by latent transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10203–10212, 2022. 2, 5, 13

[29] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International conference on machine learning*, pages 2668–2677. PMLR, 2018. 2

[30] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. 1

[31] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 3, 5, 13

[32] Jinqi Luo, Zhaoning Wang, Chen Henry Wu, Dong Huang, and Fernando De la Torre. Zero-shot model diagnosis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11631–11640, 2023. 2

[33] Ron Mokady, Amir Hertz, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Null-text inversion for editing real images using guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6038–6047, 2023. 3

[34] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 2

[35] Meike Nauta, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. Pip-net: Patch-based intuitive prototypes for interpretable image classification. 2023. 2

[36] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 3

[37] Vitali Petsiuk, Abir Das, and Kate Saenko. Rise: Randomized input sampling for explanation of black-box models. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018. 2

[38] Viraj Prabhu, Sriram Yenamandra, Prithvijit Chattopadhyay, and Judy Hoffman. Bridging the sim2real gap with care: Supervised detection adaptation with conditional alignment and reweighting. 2023. 2

[39] Pau Rodriguez, Massimo Caccia, Alexandre Lacoste, Lee Zamparo, Issam Laradji, Laurent Charlin, and David Vazquez. Beyond trivial counterfactual explanations with diverse valuable explanations. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1056–1065, 2021. 1, 2, 5, 6

[40] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, June 2022. 2, 3

[41] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22500–22510, June 2023. 2

[42] Pedro Sanchez and Sotirios A. Tsaftaris. Diffusion causal models for counterfactual estimation. In *CLEaR*, 2022. 2

[43] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade W Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa R Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. LAION-5b: An open large-scale dataset for training next generation image-text models. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. 1

[44] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 618–626, 2017. 2

[45] Sumedha Singla, Brian Pollack, Junxiang Chen, and Kayhan Batmanghelich. Explanation by progressive exaggeration. In *International Conference on Learning Representations*, 2020. 2, 5, 13

[46] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations*, 2021. 3, 4

[47] Joshua Vendrow, Saachi Jain, Logan Engstrom, and Aleksander Madry. Dataset interfaces: Diagnosing model failures using controllable counterfactual generation. In *ArXiv preprint arXiv:2302.07865*, 2023. 2

[48] Sandra Wachter, Brent Mittelstadt, and Chris Russell. Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. *arvard Journal of Law and Technology,*, 31(2):841–887, 2018. 2

[49] Bram Wallace, Akash Gokul, and Nikhil Naik. Edict: Exact diffusion inversion via coupled transformations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22532–22541, 2023. 2, 3, 4

[50] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous

multitask learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2633–2642, 2020. 5, 8

[51] Mehdi Zemni, Mickaël Chen, Éloi Zablocki, Hédi Ben-Younes, Patrick Pérez, and Matthieu Cord. Octet: Object-aware counterfactual explanations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15062–15071, 2023. 2, 8

[52] Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*, 2023. 2

[53] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8827–8836, 2018. 2

[54] Ligeng Zhu, Zhijian Liu, and Song Han. Deep leakage from gradients. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

Supplementary Material
Text-to-Image Models for Counterfactual Explanations: a Black-Box Approach

A. Evaluation Criteria

Before describing each metric and its formulation, we will thoroughly describe the goals of counterfactual explanations. As we stated in the main manuscript, counterfactual explanations seek to change an instance prediction by modifying the input instance. However, these modifications must be small but perceptually coherent. From the previous statement, we can extract many goals of CEs:

1. CEs must flip the decision of the classifier. In the literature, this feature is called *validity*.

2. The counterfactual changes should be plausible and realistic - simply referred to as *realism*. Visual automated systems are generally brittle to adversarial noise [16]. This noise is designed to fool the classifier, but with the restriction that it is hidden from visual inspection. Since this noise cannot be perceived, it cannot be analyzed to find spurious correlations. Therefore, only realistic and plausible changes are allowed.

3. The algorithm must generate *proximal and sparse* counterfactuals. One could create a valid and realistic explanation by simply replacing the target instance with a new one. This still obeys the realistic and valid goals. However, it does not give any information about the variables. Thus, the modifications must be sparse and close to the image to visually observe which variables have changed.

4. Finally, the algorithm must generate the explanation *efficiently*. This property is required to avoid delays for the user.

Now we will proceed to describe each evaluation metric and link it to its corresponding objective. As for notations, let $M(x, y)$ be the counterfactual algorithm applied to an image $x \in D$ targeting the class y, where D is a dataset. Additionally, let C be the classifier, $\mathbb{1}(condition)$ a function that is one if the condition is true or zero otherwise. Finally, let $a \in A$ be an attribute in a set A, then O^a is an attribute oracle classifier for a. This network predicts if its input has the attribute a. Similarly, let $\mathcal{O}$ be an identity verification network This DNN is trained to give a similarity measure between two images, often computed with the cosine similarity CS.

Success Rate. The success rate (or flip rate) measures the ratio at which counterfactuals have successfully reversed the original classifier's decision. This metric correlates with the validity goal. To measure it, we simply compute the proportion of valid counterfactuals to the size of the dataset, as in

$$SR = \frac{1}{|D|} \sum_{x \in D} \mathbb{1}(C(M(x, y)) = y). \tag{12}$$

Realism. To approximate the realism of the counterfactuals, the literature adopts the FID [18] metric from generation research. Furthermore, [25] extended the metric by computing the FID between the half of the dataset and the counterfactuals of the complement set. This was motivated to reduce the inherent bias in computing the FID, given that the difference between the original images and their CE is a few pixels in the image.

Proximity and Sparsity. To evaluate this goal, previous methods proposed several metrics to quantify the degree of dissimilarity between an instance and its explanation. Initially, most metrics were proposed for face images. Initially, [23] suggested using the mean number of attributes changed (MNAC), computed as follows:

$$MNAC = \frac{1}{|D|} \sum_{x \in D} \sum_{a \in A} \mathbb{1}(O^a(M(x, y)) \neq O^a(x)). \tag{13}$$

However, [24] noted that counterfactual methods will change some attributes if they are correlated. Thus, based on the MNAC, the Correlation Difference (CD) [24] measures the correlations produced by M. To further assess the proximity and sparsity in face counterfactuals, [45] suggested using the Face Verification Accuracy (FVA) to compute whether M cannot modify the identity of the person. This metric is calculated as

$$FVA = \frac{1}{|D|} \sum_{x \in D} \mathbb{1}(CS(\mathcal{O}(x), \mathcal{O}(M(x, y))) > 0.5). \tag{14}$$

[25] noted that this metric was already saturated. To measure a more fine-grained metric, they proposed taking the continuous CS and calling the metric face similarity (FS):

$$FS = \frac{1}{|D|} \sum_{x \in D} CS(\mathcal{O}(x), \mathcal{O}(M(x, y)). \tag{15}$$

Finally, the same authors extended this metric for general-purpose images by computing Eq. 15 using a self-supervised trained model as $\mathcal{O}$. They called this metric S^3. Finally, [28] proposed to compute COUT. This metric computes the probability of the class y using multiple linear interpolations between x and $M(x, y)$.

Efficiency. The literature generally ignores computing an *efficiency* metric. To compute the efficiency of counterfactual models, the widely accepted metric is floating point operations (FLOPs). In addition, it is also recommended to compute the average time per counterfactual. However, this metric is only comparable if all measurements are computed on the under the same circumstances.

B. Qualitative Results

In this section, we provide additional qualitative results. For the CelebA HQ [31] dataset, we provide our and ACE [25] counterfactuals to show the differences.

Figure 4. Counterfactual Explanations targeting the Non-Smile attribute.

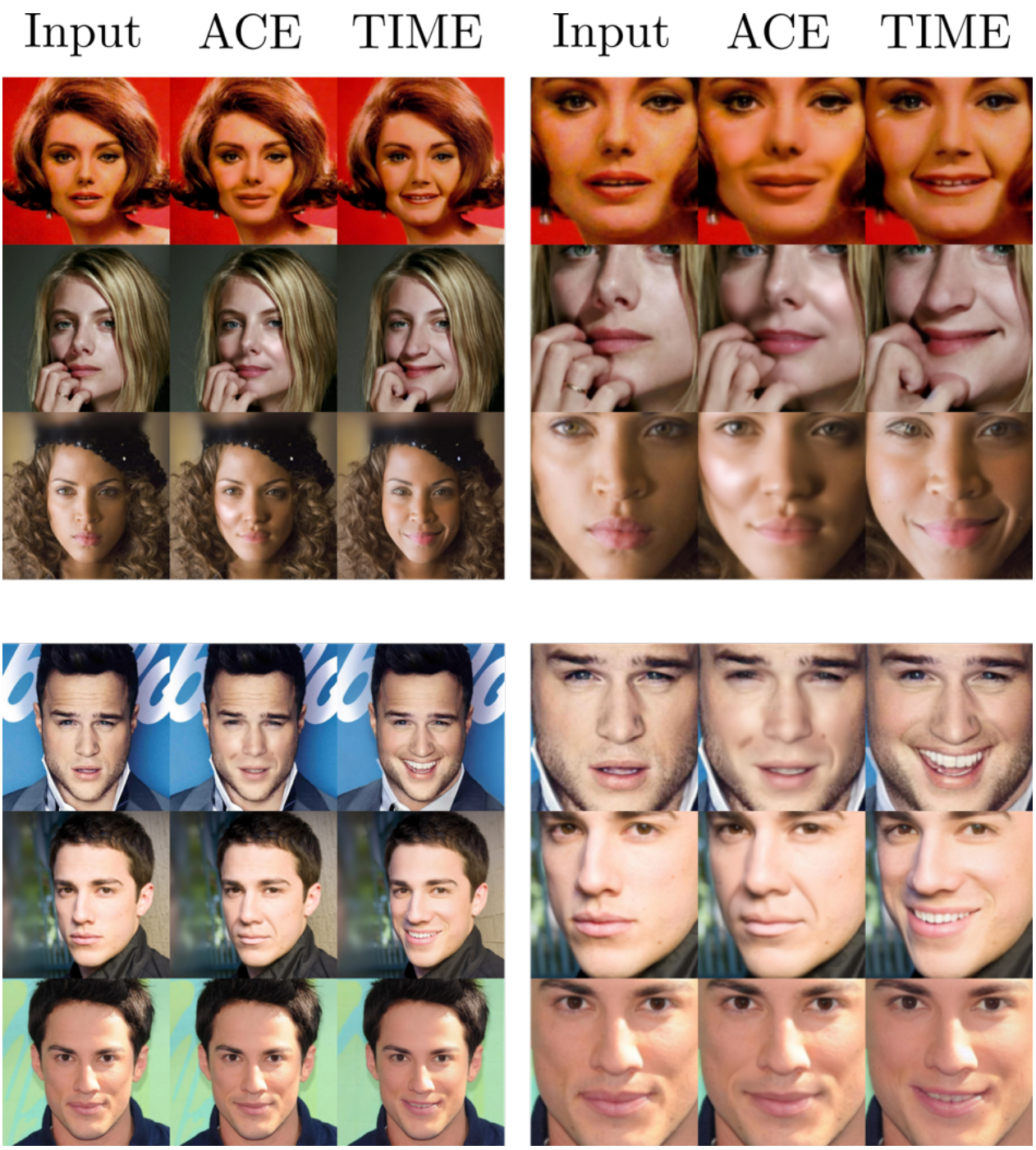

Figure 5. Counterfactual Explanations targeting the Smile attribute.

Figure 6. Counterfactual Explanations targeting the Young attribute.

Figure 7. Counterfactual Explanations targeting the Old attribute.

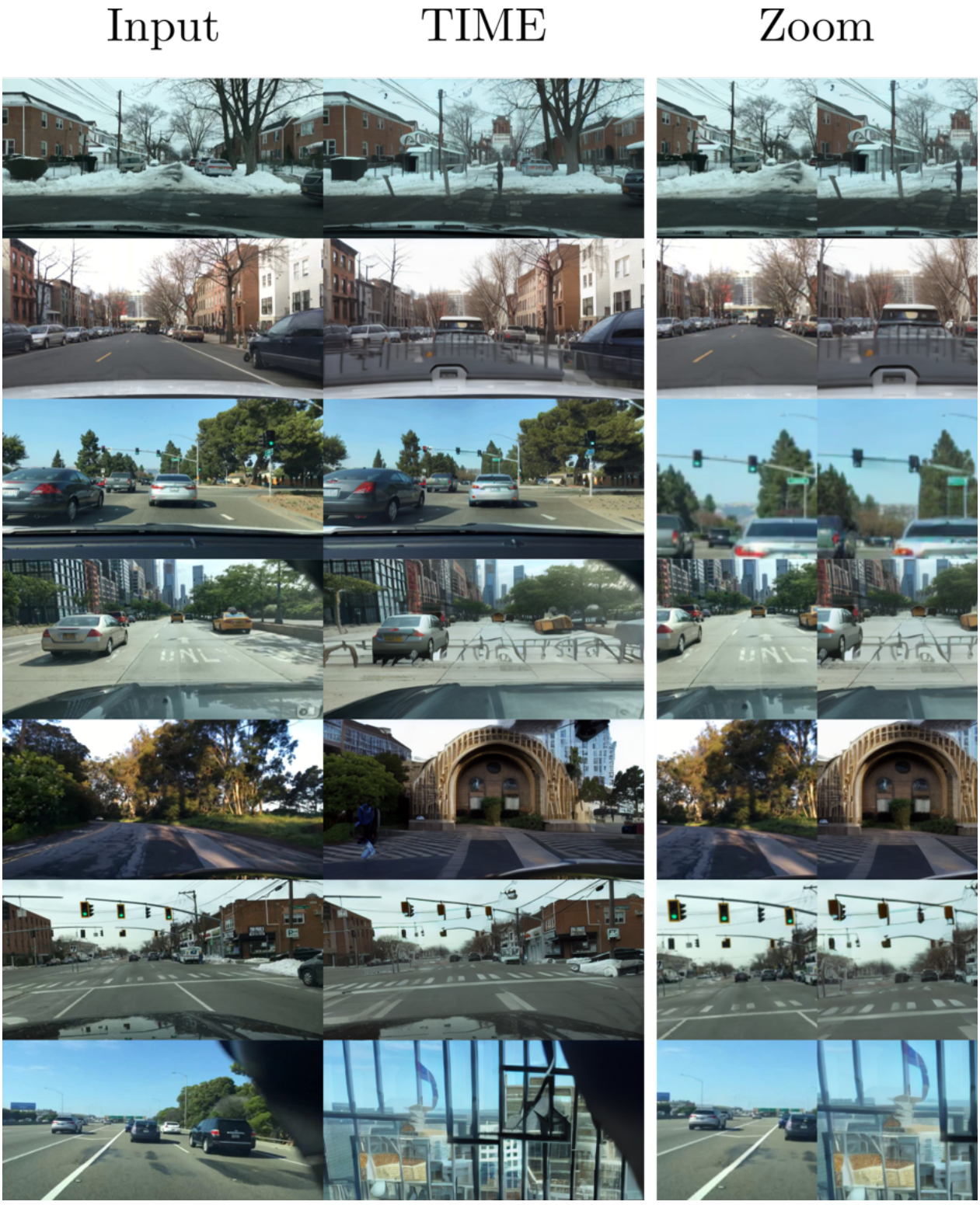

Figure 8. Counterfactual Explanations targeting the Stop action.

Figure 9. Counterfactual Explanations targeting the Forward action.