A Appendix

A.1 Implementation Details

The implementation of CoLa-DCE is based on the LDCE Farid et al. (2023) implementation, which is available on GitHub `https://github.com/lmb-freiburg/ldce`. Adaptations have mainly been made to the scoring function, deriving the gradient-based guidance for the diffusion model. For computing concept patches to visualize the concepts, the CRP Achtibat et al. (2023) implementation from `https://github.com/rachtibat/zennit-crp` has been used. For optimization, the concept conditioning is relaxed in the last 50 steps of the diffusion generation to use the complete gradient for image refinement. To our knowledge, no semantic change in the image can be perceived, while mainly low-level features such as edges are refined. The parametrization in our experiments is not model-specific. It is based on the proposed parametrization in LDCE Farid et al. (2023) with only the `lp-dist` parameter changed to 0.01, as a high value might result in significant features being removed again during the diffusion process. Optimizing the parameters based on the used model is expected to affect the generated counterfactuals positively. Our implementation for CoLa-DCE is accessible at `github.com/continental/concept-counterfactuals`.

On ImageNet, one run of the CoLa-DCE code for a single set of parameters and 1000 images on an NVIDIA RTX A5000 takes approximately 16 hours with a batch size of 4. One generation step takes slightly less than 3 minutes on the same hardware. The code should be similarly efficient as the LDCE code from their GitHub.

A.2 Models and Datasets

This paper uses the following datasets and models. The images in the main paper originate from the ImageNet dataset.

Dataset	License	URL
ImageNet Deng et al. (2009)	Custom	`https://www.image-net.org/index.php`
Oxford Flowers 102 Nilsback & Zisserman (2008)	GNU	`https://www.robots.ox.ac.uk/vgg/data/flowers/102/`
Oxford-IIIT Pet Parkhi et al. (2012)	CC BY-SA 4.0	`https://www.robots.ox.ac.uk/vgg/data/pets/`

Table 2: Dataset Specification

Model	License	URL
VGG16	BSD 3	`https://pytorch.org/vision/stable/models/vgg.html`
VGG16bn	BSD 3	`https://pytorch.org/vision/stable/models/vgg.html`
ResNet18	BSD 3	`https://pytorch.org/vision/stable/models/resnet.html`
ViT-B-16	BSD 3	`https://pytorch.org/vision/stable/models/vision_transformer.html`
class-conditional LDM Rombach et al. (2022)	MIT	`https://github.com/CompVis/latent-diffusion`
miniSD (Pinkney, 2023)	Open RAIL-M	`https://huggingface.co/justinpinkney/miniSD`

Table 3: Model Specification

A.3 Further CoLa-DCE Examples

For the Flowers and Pets datasets, a VGG16bn has been finetuned on a few epochs until decent accuracy of over 85%.

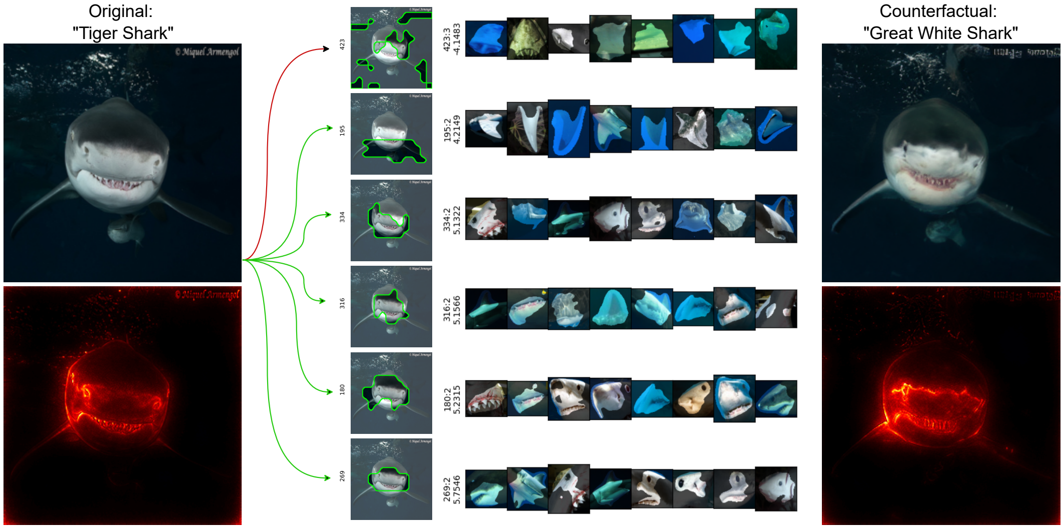

Figure 8: CoLa-DCE example for an ImageNet sample and the VGG16bn model. The counterfactual with class “Great White Shark” is modified in the head structure with more forward-facing eyes and a sharper, pointed nose. Also, the mouth section is adapted to the counterfactual class.

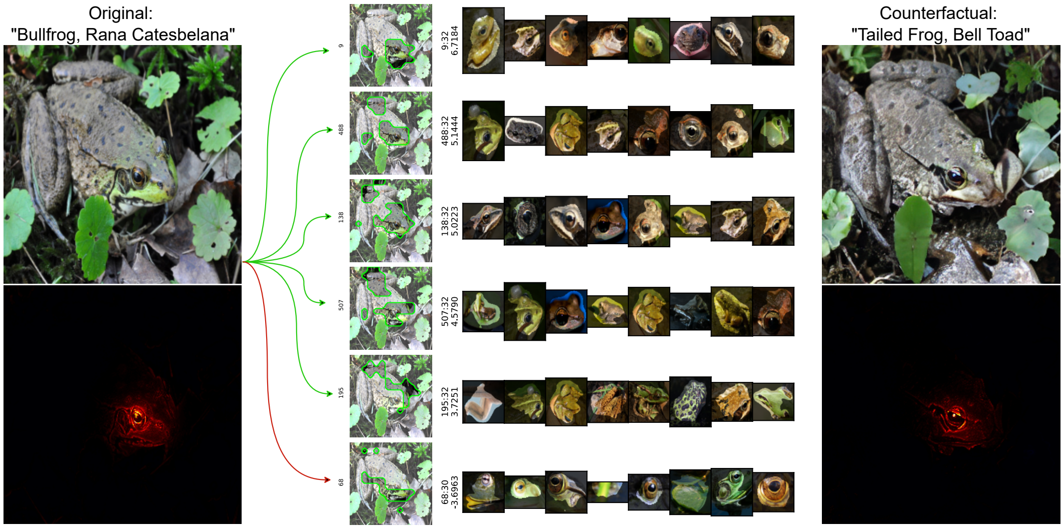

Figure 9: CoLa-DCE example on the ImageNet dataset from “Bullfrog, Rana Catesbelana” to “Tailed Frog, Bell Toad”.

A.3.1 Oxford Pets dataset:

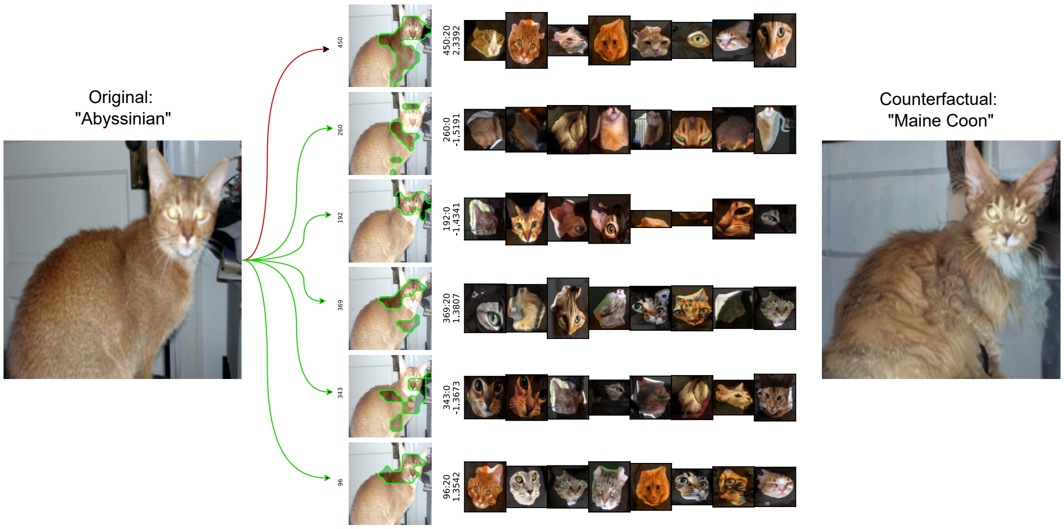

Figure 10: CoLa-DCE example on the Pets dataset from “Abyssinian” to “Maine Coon”.

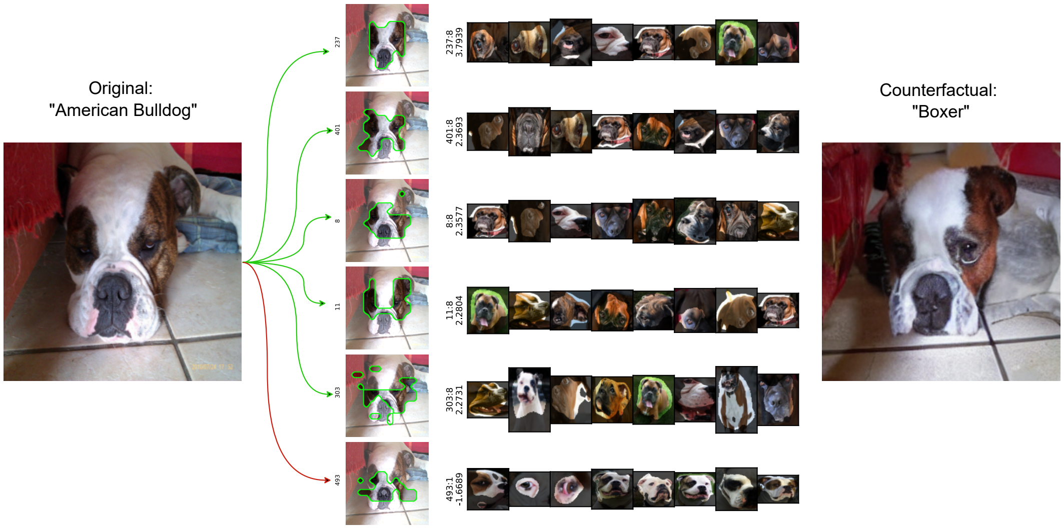

Figure 11: CoLa-DCE example on the Pets dataset from “American Bulldog” to “Boxer”.

A.3.2 Oxford Flowers dataset:

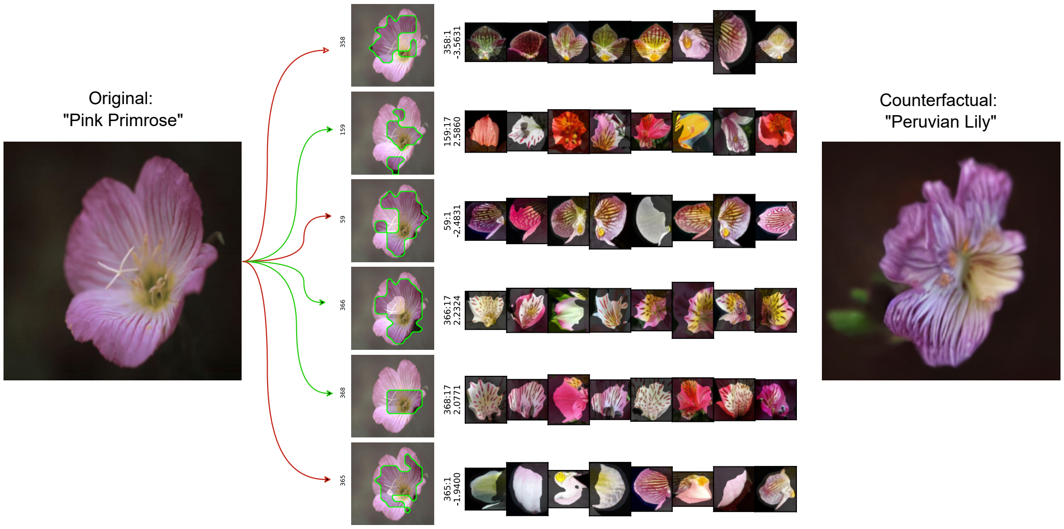

Figure 12: CoLa-DCE example on the Flowers dataset from “Pink Primrose” to “Peruvian Lily”.

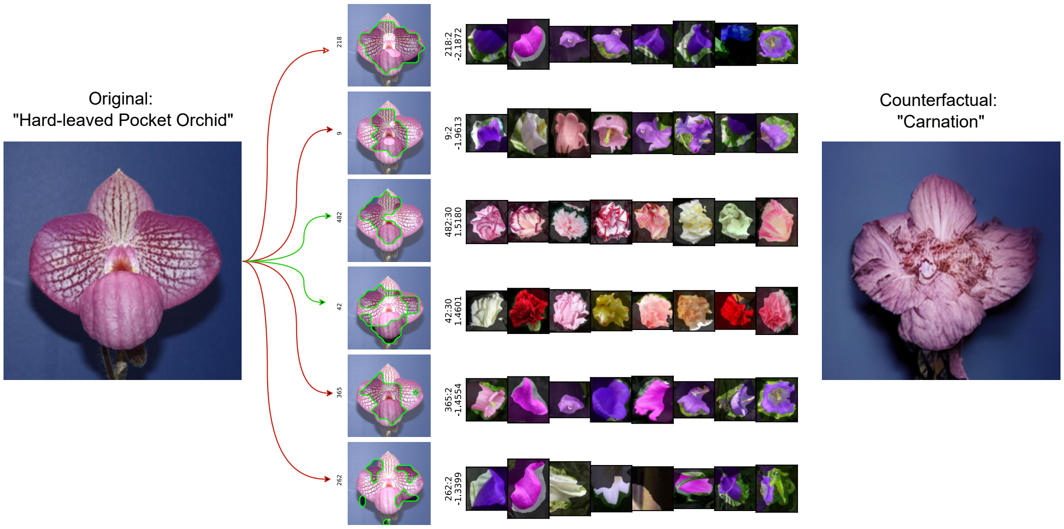

Figure 13: CoLa-DCE example on the Flowers dataset from “Hard-leaved Pocket Orchid” to “Carnation”.

A.4 Further quantitative experiments

Further quantitative experiments have been run on the test set of the Oxford PETS dataset. Instead of providing class conditioning, a prompt is used, stating the class and that it is a kind of pet.

Table 4: CoLa-DCE results for the Oxford PETS dataset using a ViT model.

Model	Num Concepts	FID	Flip Ratio	Confidence
ViT	LDCE	84.5	0.94	0.917
ViT	10	110.3	0.79	0.816
ViT	20	76.5	0.81	0.83
ViT	50	78.3	0.79	0.833
ViT	100	80.8	0.99	0.976
ViT	200	77.1	0.79	0.848

Table 5: CoLa-DCE results on the CUB-200-2011 dataset using a ViT model.

Model	Num Concepts	FID	Flip Ratio	Confidence
ViT	LDCE	40.35	0.91	0.867
ViT	10	31.66	0.51	0.616
ViT	20	31.96	0.5	0.622
ViT	50	41.3	0.92	0.87
ViT	100	31.5	0.5	0.611
ViT	200	31.96	0.52	0.62

A.5 A Discussion on Adversarial Examples

While counterfactuals are supposed to be semantic changes in an input image, there is always the possibility that single pixel changes in an image trigger the classifier to predict a different targeted class. These changes are named adversarial examples. While there is no guarantee that a generated image does not include adversarial pixel changes, we highlight the functionality of CoLa-DCE and LDCE as an ensemble of models that makes the appearance of adversarials unlikely. For the counterfactual generation on ImageNet data, the class-conditioned diffusion model and the external classifier are trained on the same data so that similar shortcuts can potentially be learned. However, the classifier is trained to discriminate between classes, while the diffusion model is trained to generate semantic class features and to represent the data distribution in a semantic encoding. A potential adversarial signal would need to be encoded in the gradients of both models to be included into the gradient alignment, which is used for guiding the diffusion process. As additionally latent diffusion is used, the encoded representation needs to be decoded to a human-observable image in input space by the trained decoder, which would be required to preserve the adversarial signal and reconstruct it into the respective image pixels. We argue that the probability of such a signal fitting a possible adversarial trigger in the external classifier is relatively low. With the usage of concept-based conditioning, the concept-gradient of the external classifier is used, directly pointing out which features should be changed in which areas of the input image. This level of control and semantic guidance is another factor diminishing the probability of adversarial patterns. While the dataset might induce semantically wrong class patterns in all related models, we argue that these patterns represent valid dataset features requiring a dataset adaptation. They can be easily found by inspecting the concept patches given in our CoLa-DCE explanations.

A.6 COMPARISON OF LDCE AND COLA-DCE COUNTERFACTUALS

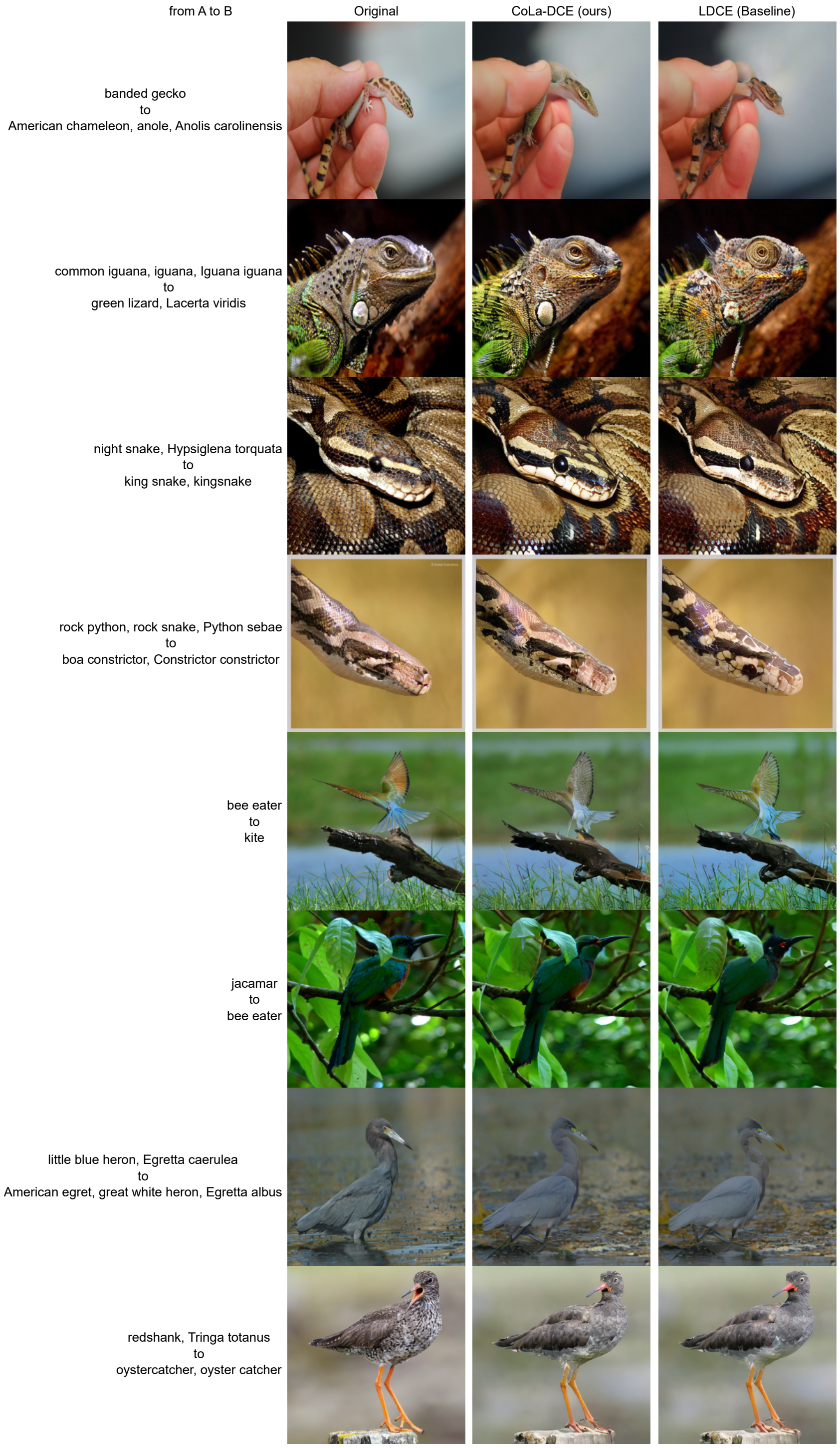
from A to B
Original
CoLa-DCE (ours)
LDCE (Baseline)
banded gecko
to
American chameleon, anole, Anolis carolinensis
common iguana, iguana, Iguana iguana
to
green lizard, Lacerta viridis
night snake, Hypsiglena torquata
to
king snake, kingsnake
rock python, rock snake, Python sebae
to
boa constrictor, Constrictor constrictor
bee eater
to
kite
jacamar
to
bee eater
little blue heron, Egretta caerulea
to
American egret, great white heron, Egretta albus
redshank, Tringa totanus
to
oystercatcher, oyster catcher

Figure 14: Comparison of generated samples on the ImageNet dataset between LDCE and our CoLa-DCE using a VGG16bn. The CoLa-DCE samples include fewer feature changes and even look more realistic for some examples.

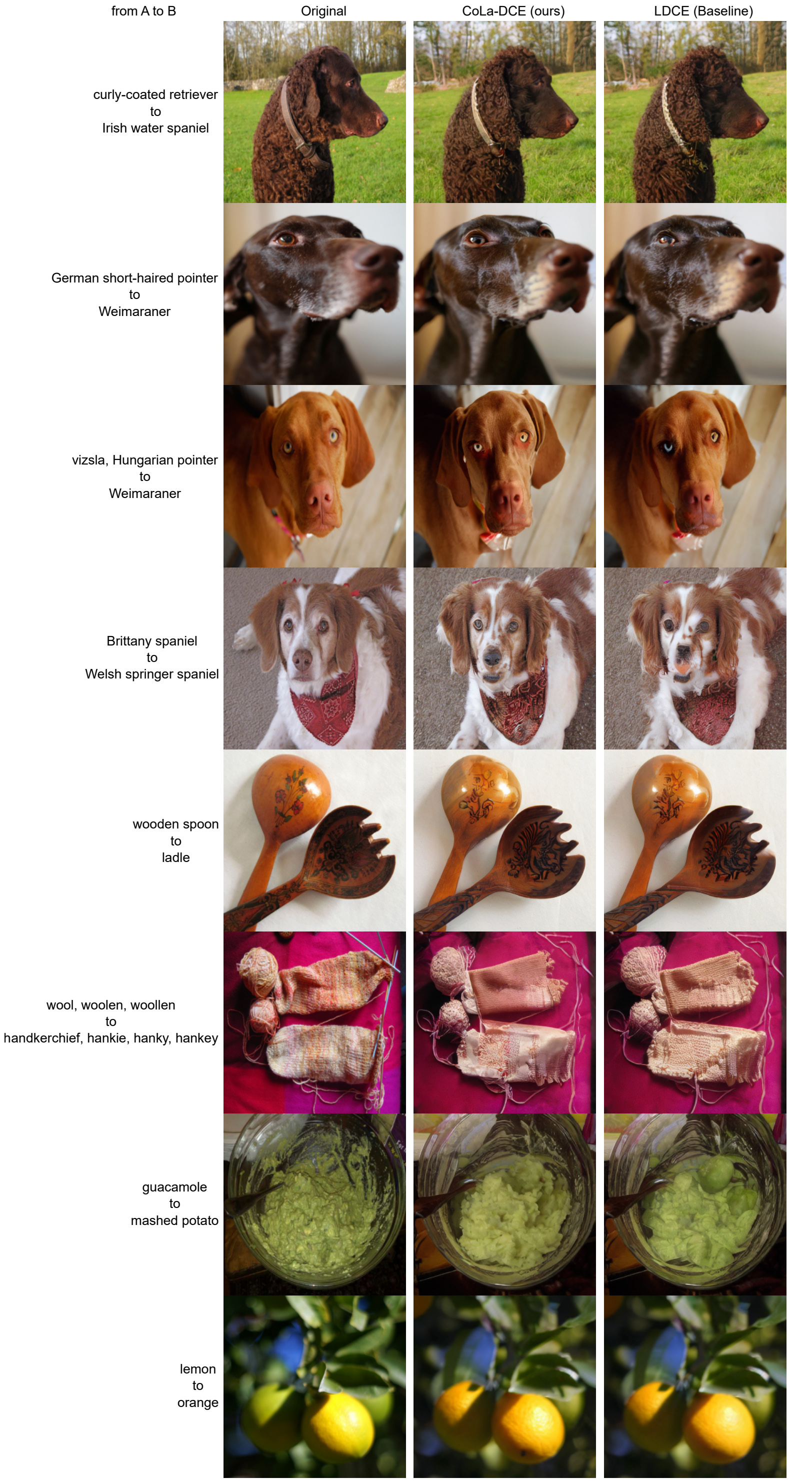
from A to B
Original
CoLa-DCE (ours)
LDCE (Baseline)
curly-coated retriever to Irish water spaniel
German short-haired pointer to Weimaraner
vizsla, Hungarian pointer to Weimaraner
Brittany spaniel to Welsh springer spaniel
wooden spoon to ladle
wool, woolen, woollen to handkerchief, hankie, hanky, hankey
guacamole to mashed potato
lemon to orange

Figure 15: Comparison of generated samples between LDCE and our CoLa-DCE.

A.7 Comparing concept guidance on LDCE and DVCE

We implemented our concept-based extension also for the DVCE Augustin et al. (2022) implementation and compare both versions in a small experiment using $c = 20$ concepts. In Table 6, a clear advantage of the CoLa-DCE method is visible, while the visual inspection in Figure 16 shows the same result. Based on the DVCE method, less feature changes are visible, while the generated images do not look as realistic as for the CoLa-DCE version. We refer these results to the additional model gradients, which have to be aligned in order to derive the conditioning signal. The gradient alignment can hereby be seen as a filter so that the additionally applied filtering removes too much information from the target-directed gradient, leaving a rather weak and unstable conditioning signal.

Table 6: Comparing CoLa-DCE results for LDCE and DVCE on ImageNet and a VGG16bn model.

Model	**Num Concepts**	**FID**	**Flip Ratio**	**Confidence**
LDCE	20	45.40	**0.84**	**0.83**
DVCE	20	**44.04**	0.246	0.453

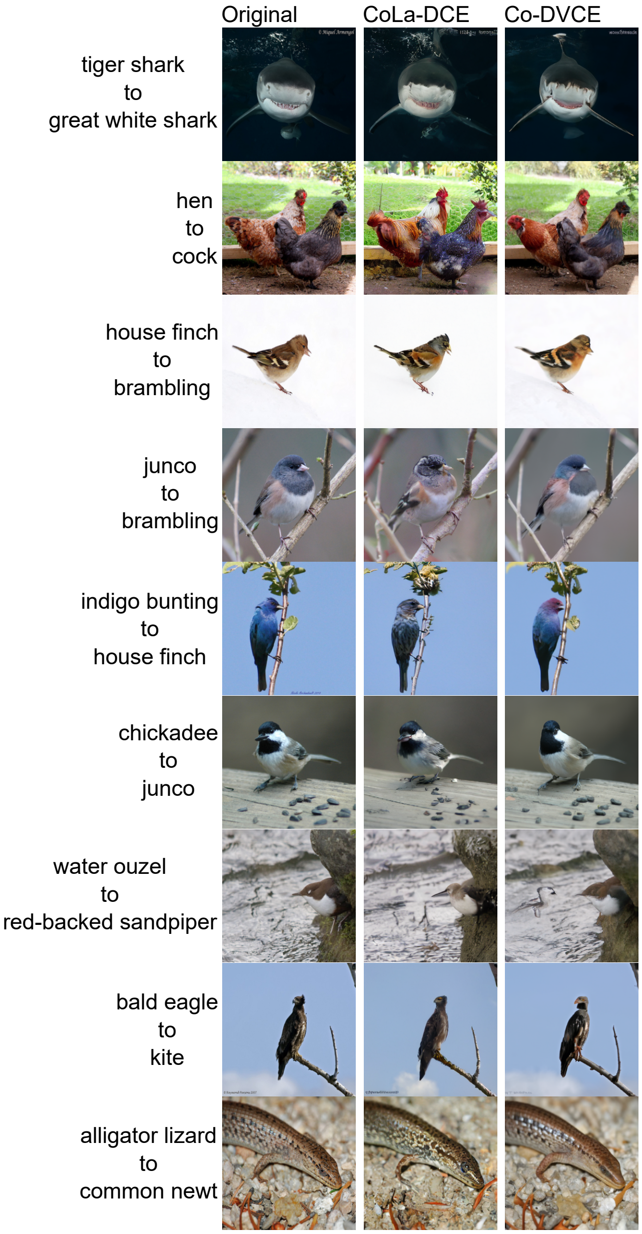
Original
CoLa-DCE
Co-DVCE
tiger shark to great white shark
hen to cock
house finch to brambling
junco to brambling
indigo bunting to house finch
chickadee to junco
water ouzel to red-backed sandpiper
bald eagle to kite
alligator lizard to common newt

Figure 16: Comparison of the generated samples on ImageNet using CoLa-DCE and the concept extension of the DVCE method. More clear feature changes towards the target class and more realistic images can be seen for our CoLa-DCE method.

A.8 How does the choice of diffusion model influence the results?

In an extension of the current framework, we updated the code base to incorporate the prompt-based image-to-image pipelines in the diffusers library von Platen et al. (2022). Multiple trained models and versions of stable diffusion can easily be tested on the integrated datasets. Additionally, a modified pipeline has been implemented to include the Kandinsky diffusion model Arkhipkin et al. (2023). The original guidance function has thereby been rearranged to include the implicit classifier score, which can be modified likewise to the LDCE and CoLa-DCE methods.

While we observe that a higher guidance scale is generally needed to obtain visually convincing results, remarkably lower quantitative performance scores are already measured in the baseline setting. The diffusion models included in the diffusers library are all trained on the LAION dataset Schuh-

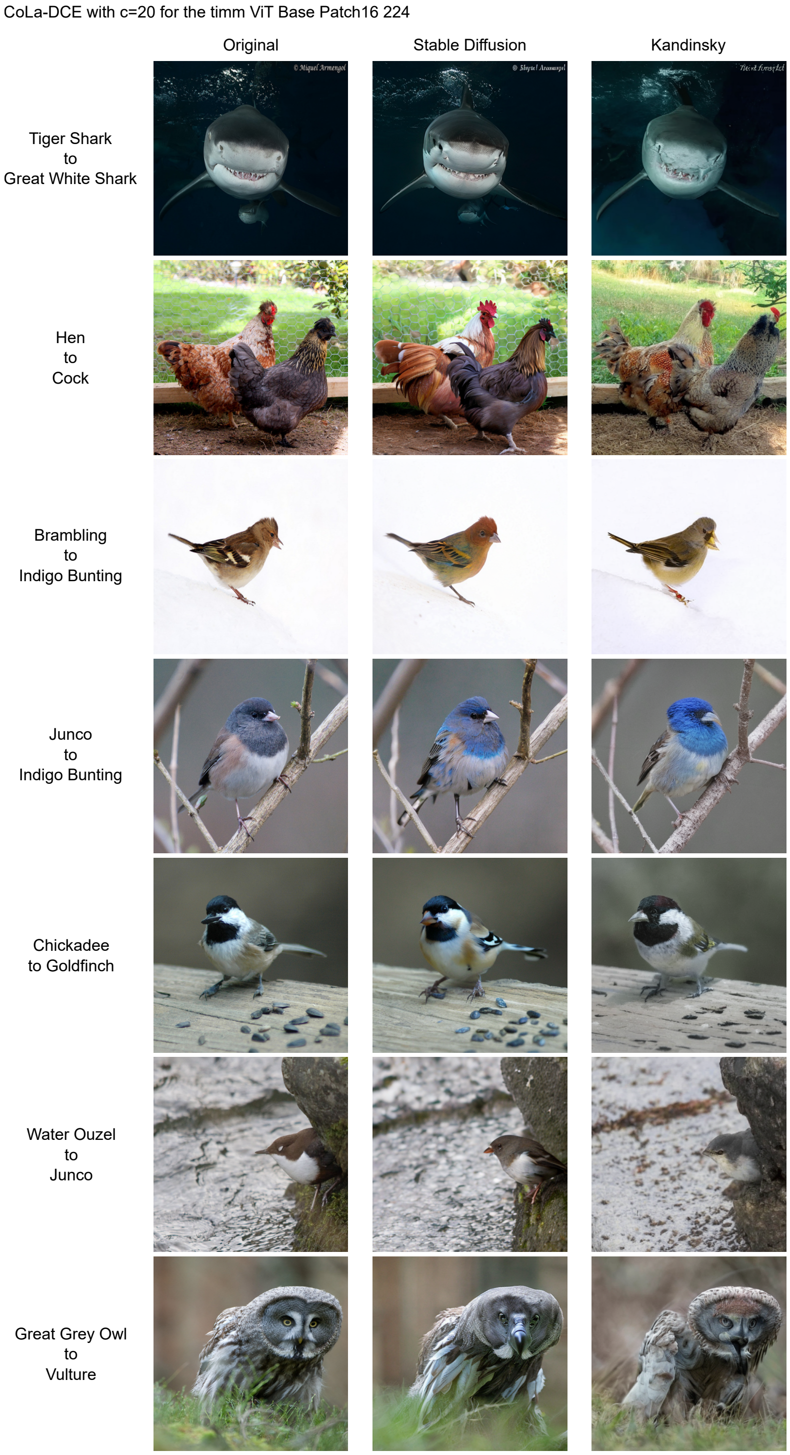
CoLa-DCE with c=20 for the timm ViT Base Patch16 224
Original
Stable Diffusion
Kandinsky
Tiger Shark to Great White Shark
Hen to Cock
Brambling to Indigo Bunting
Junco to Indigo Bunting
Chickadee to Goldfinch
Water Ouzel to Junco
Great Grey Owl to Vulture

Figure 17: Comparison of the generated samples using different diffusion models from the diffusers library, trained on the LAION dataset. We applied CoLa-DCE with the Stable Diffusion version of StabilityAI and the Kandinsky3 model from the diffusers library. As a classifier, the ViT base model from timm was used.

mann et al. (2022) instead of specifically modeling the ImageNet data, such that fine-grained class features might not be well encoded in the trained diffusion models. While zero-shot CLIP-based classifiers have shown moderate performance in cross-dataset testing, the fine-grained class-based image generation is more affected by the level of accurately encoded features. Thus, the models trained on LAION do not perform well quantitatively in generating counterfactuals with a high flip ratio. For a concept number of $c = 20$, the diffusers stable diffusion model (trained on LAION) with CoLa-DCE on ImageNet has the following performance values: FID=47.199, Flip ratio=0.144, Confidence=0.629. The diffusers Kandinsky model has with the same settings a performance of: FID=105.5, Flip ratio=0.037, Confidence=0.661.

A.9 Testing on the CUB dataset

We additionally test our CoLa-DCE method on the CUB-200-2011 dataset to additionally visualize the most important concepts used between two classes. The results show clearly recognizable concepts, which can be detected in the counterfactual image and show the learned difference between the two classes.

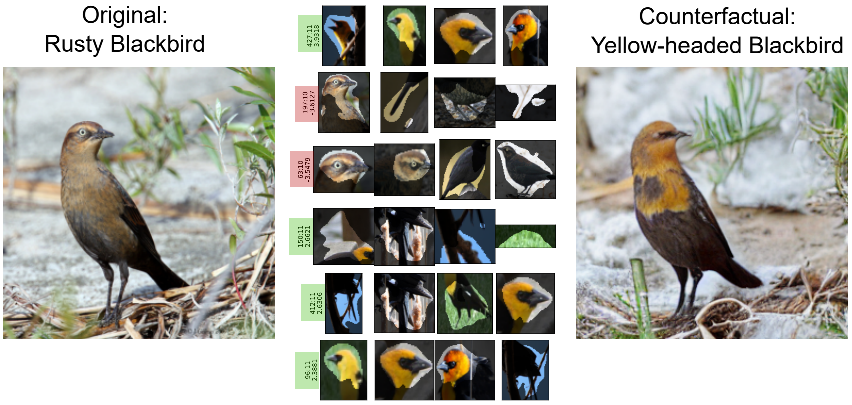

Figure 18: A CoLa-DCE example on the CUB-200-2011 dataset using a stable diffusion model together with a finetuned VGG16bn model. The yellow head can clearly be derived from the shown concept and is visible in the counterfactual. Additionally, the Google Image Search confirmed the counterfactual class.

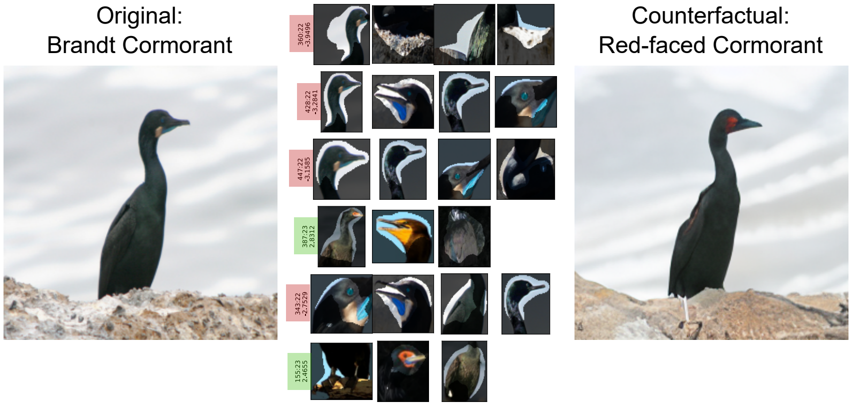

Figure 19: A CoLa-DCE example on the CUB-200-2011 dataset using a stable diffusion model together with a finetuned VGG16bn model. While the blue features from the original class are reduced, red features are added to the face of the cormorant.