ML in PL CONFERENCE 2024

Region-constrained Visual Counterfactual Explanations

Bartłomiej Sobieski [1,2], Jakub Grzywaczewski [2], Bartłomiej Sadlej [1], Matthew Tivnan [3], Przemyslaw Biecek [1,2]

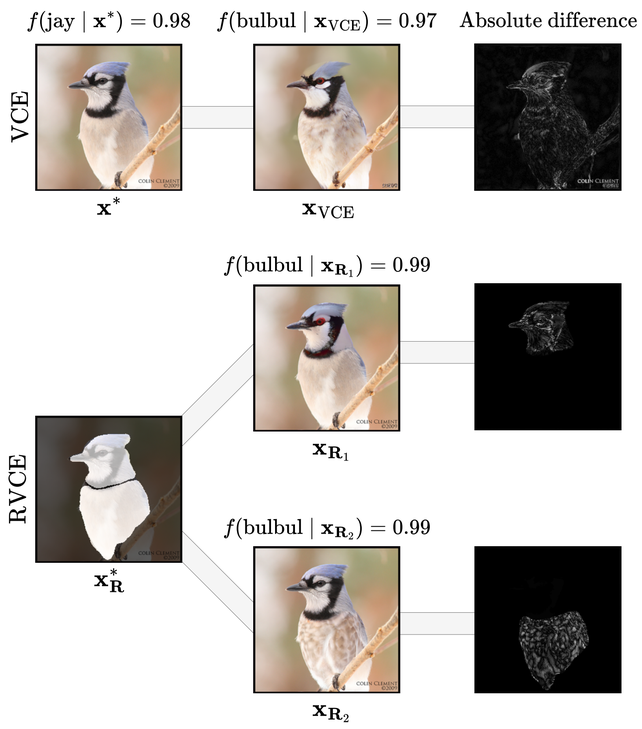
[1] University of Warsaw, [2] Warsaw University of Technology, [3] Harvard Medical School

przemyslaw.biecek@gmail.com

Let's talk about: explainable computer vision, counterfactuals, adversarial attacks, diffusion models

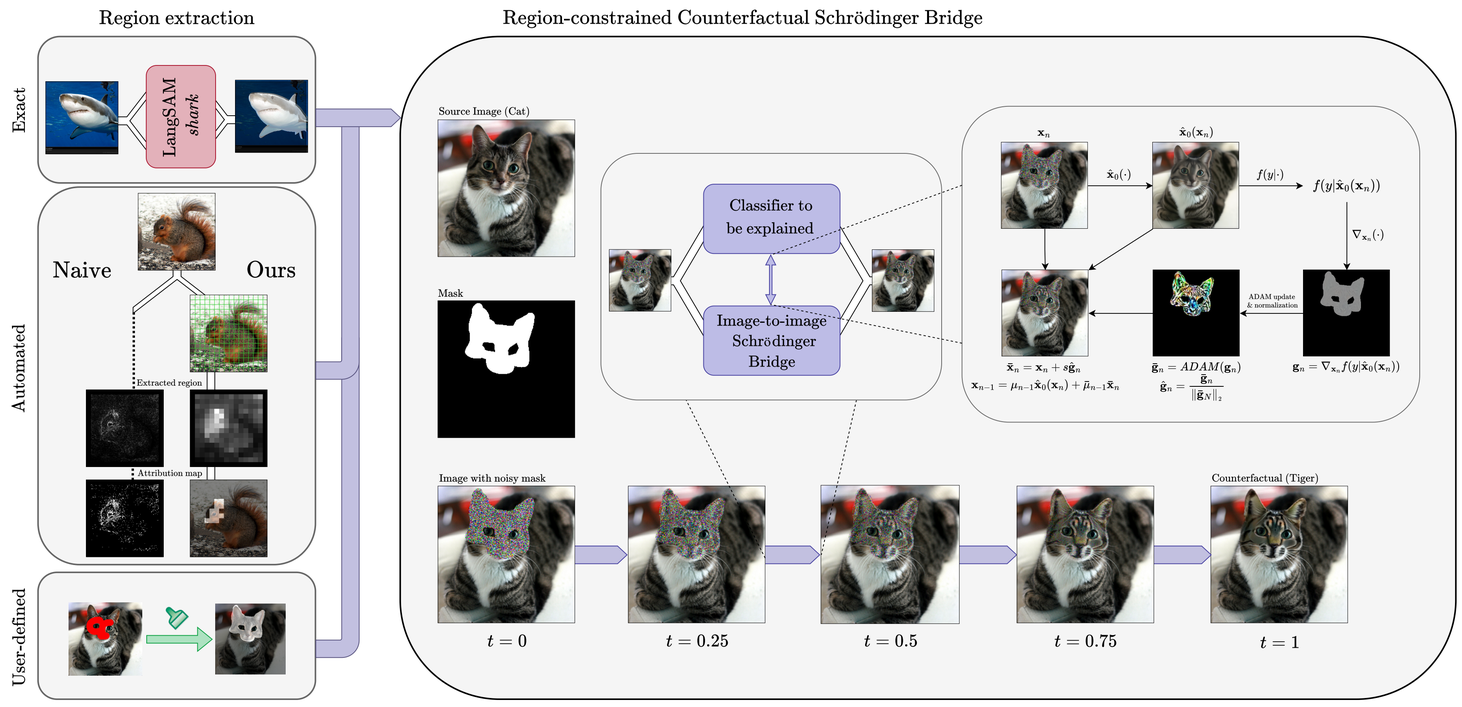
What are RVCEs?

Visual counterfactual explanations (VCEs) aim to explain the decision-making process of an image classifier by modifying the input image in a semantically meaningful and minimal way so that the classifier's decision changes. Right now, they are the cornerstone of explainable AI for vision models, yet they suffer from grave limitations. VCE methods can modify the entirety of the image at once. We propose *region-constrained* VCEs (RVCEs), which assume that one can change a predefined image region to influence the model's prediction. Our work advances the state of VCE generation by constraining the explanations to differ from the factual image exclusively within a predetermined region.



Region-Constrained Counterfactual Schrödinger Bridges (RCSB)

To effectively sample from the distribution of RVCEs based on a provided image, we propose *RCSB*, an adaptation of a tractable subclass of Schrödinger Bridges to the problem of conditional inpainting, where the conditioning signal originates from the classifier of interest. In addition to setting a new SOTA by a large margin, we extend RCSB to allow for *exact* counterfactual reasoning, where the predefined region contains *only* the factor of interest, and incorporating the user to actively interact with the RVCE by predefining the regions manually. Our method utilizes the OT-ODE version of I^{2}SB trained for inpainting. Both I^{2}SB and diffusion models are score-based methods where the generation is the backward process guided by the score of the forward SDE. One can consider the conditional score as we replace the score $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t)$ with $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t \mid \mathbf{y})$, which can be decomposed with Bayes' Theorem into $\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t) + \nabla_{\mathbf{x}_t} \log p(\mathbf{y} \mid \mathbf{x}_t, t)$ which we can model separately by a trained I^{2}SB and our classifier.



Automated region extraction

As a backbone of our automated extraction technique we decided to use Attribution Methods. To extract a region from pixel-level attributions, one can threshold them to cover a specific fraction a of the total image area. However, after binarizing the attributions with $a = 0.05$, we observe that the resulting region is highly scattered, losing focus from semantic concepts. To address this issue, we divide the image into a grid of square cells of size $c \times c$, where each cell receives the value equal to the sum of the absolute pixel attributions inside it. By thresholding it with $a = 0.05$, we observe the extracted region to focus solely on the squirrel's head (see left panel). This leads to a fully automated strategy for obtaining regions that are both aligned with semantically meaningful concepts and based on pixels that are important for the classifier. To fortify our method we have experimented with the most well established attribution methods such as GradCAM, Integrated Gradients or Occlusion.

Improving estimation

Tweedie's formula. The formula states that the *denoised estimate* of the final image at step t can be achieved by computing the posterior expectation

$$\hat{\mathbf{x}}_0(\mathbf{x}_t) := \mathbb{E}[\mathbf{x}_0 \mid \mathbf{x}_t] = \mathbf{x}_t + \sigma_t^2 \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t, t).$$

Replacing $\nabla_{\mathbf{x}_t} \log f(y \mid \mathbf{x}_t)$ with $\nabla_{\mathbf{x}_t} \log f(y \mid \hat{\mathbf{x}}_0(\mathbf{x}_t))$ as the estimator of the $\nabla_{\mathbf{x}_t} \log p(y \mid \mathbf{x}_t, t)$ brings the inputs of f much closer to what a non-robust classifier encountered during its training.

ADAM stabilization. We propose to smooth out the gradients by applying the ADAM update rule at each step.

Adaptive normalization. At the beginning of the conditional inpainting process, we register the norm of the first encountered gradient. We then use it as a normalizing constant for each subsequent gradient, meaning that the generation begins with a gradient of the unit norm.

Trajectory truncation. This intervention allows for mixing available information from the image with the information from the classifier and gives direct control over preserving the original content. It decomposes the generation process into initial phases that are responsible for the overall structure of objects and later ones that are responsible for details.