Diff-CA

Separating Common and Salient Factors with Diffusion Models

Michaël Soumm[1*] · Alexandre Fournier Montgieux[2*] · Yunlong He[1] · Pietro Gori[1] · Alasdair Newson[1,3]

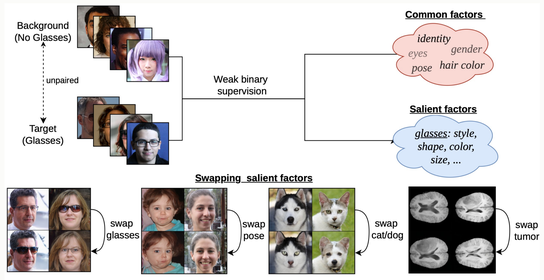
[1] Télécom Paris, Institut Polytechnique de Paris [2] CEA List, Palaiseau [3] ISIR, Sorbonne Université [*] Equal contribution

NeurIPS 2026

Spotlight

msoumm.github.io/Diff-CA

1 Contrastive Analysis



Two unpaired datasets: a **background** ($Y = 0$, e.g. faces without glasses) and a **target** ($Y = 1$, with glasses). Contrastive Analysis (CA) separates what is **common** to both (identity, pose, gender…) from what is **salient** to the target (glasses style, shape, color…), from the binary label Y alone.

Why Contrastive Analysis?

- **Discovery:** a binary label (healthy / tumor) is cheap, while a description of what differs may not exist.
- **Control:** swap, remove or interpolate the salient factor, keeping the rest fixed.
- **Gap:** VAE/GAN-based CA models are blurry, so what the salient code captures cannot be verified; text-guided editors need the factor to be named.

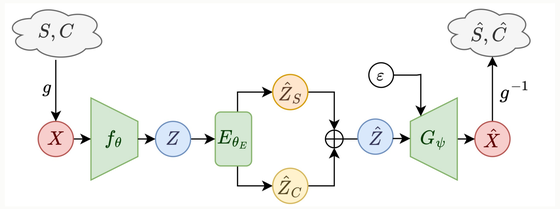
Contributions

1. **Theory:** identifiability of the additive common/salient decomposition, and a bound on control.
2. **Conditioning:** a high-fidelity, image-only conditioning space for diffusion (32 tokens).
3. **Diff-CA:** CA on diffusion conditioning tokens, with 94.5% vs. 47.0% swap accuracy over the best CA baseline.

Take-home

Contrastive Analysis on diffusion conditioning tokens gives high-fidelity images and a clean separation of **common** and **salient** factors, from a binary label alone.

2 Identifiable CA



Factors (S, C) generate X. The separator E splits the conditioning Z into $(\hat{Z}_S, \hat{Z}_C)$, whose sum conditions the generator G_ψ.

Images come from independent factors, $X = g(S, C)$ with $S \perp\!\!\!\perp C$, and $S = 0$ on the background. We assume an additive conditioning space:

$$Z = Z_S + Z_C, \text{ with } Z_S = 0 \text{ if } Y = 0$$

Theorem 1 (Identifiability)

If a learned decomposition $Z \mapsto (\hat{Z}_S, \hat{Z}_C)$ satisfies

(i) additive reconstruction: $\hat{Z}_S + \hat{Z}_C = Z$

(ii) pinning: $Y = 0 \Leftrightarrow \hat{Z}_S = 0$

(iii) independence: $\hat{Z}_S \perp\!\!\!\perp \hat{Z}_C$

then $\hat{Z}_S = Z_S$ and $\hat{Z}_C = Z_C$: the true factors are recovered.

Proposition 1 (Control)

$$I(\hat{S}; \hat{Z}_S) \geq \underbrace{I(S; \hat{S})}_{\text{fidelity}} - \underbrace{I(\hat{X}; \varepsilon \mid Z)}_{\text{noise dependence}} - \underbrace{I(S; \hat{Z}_C \mid \hat{Z}_S)}_{\text{entanglement}}$$

Controlling the salient factor through $\hat{Z}_S$ needs a faithful generator that depends little on the noise ε, and disentangled codes.

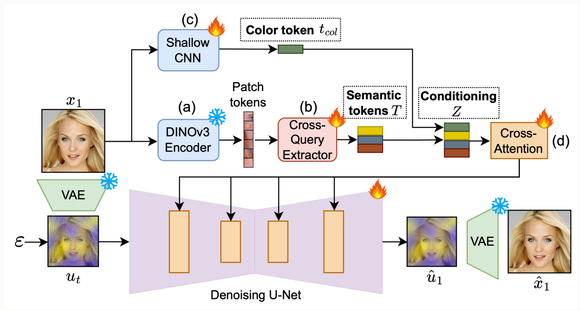
From theory to design

- **Stage 1** builds a generator with high fidelity and low noise dependence.
- **Stage 2** trains a separator towards (i)–(iii), with $\mathcal{L}_{\text{rec}}$, $\mathcal{L}_{\text{pin}}$, and $\mathcal{L}_{\text{adv}} + \mathcal{L}_{\text{cyc}}$.

3 Diff-CA in two stages

Stage 1 · Image-only conditioning

A latent flow-matching U-Net, trained from scratch and conditioned only on the image. A Cross-Query extractor compresses frozen DINOv3 features into 32 tokens, plus a color token. Deterministic sampling: the image is driven by Z, not by the noise.

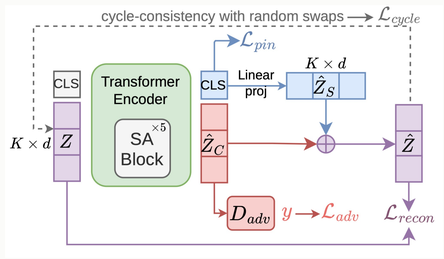


Conditioning	Tokens	SSIM ↑	LPIPS ↓	FID ↓
DiffAE	1 × 512	.530	.209	14.28
EncDiff	32 × 128	.420	.323	18.43
DINOv3	261 × 128	.645	.099	5.66
Ours (CQC)	**32 × 128**	**.611**	**.114**	**5.52**

Reconstruction on CelebA-HQ (trained on FFHQ). With 8× fewer tokens than DINOv3, our conditioning stays close, and unlike DINOv3 it supports editing.

Stage 2 · Common/salient separator

A 5-layer Transformer splits Z into a **salient** code (CLS token) and a **common** code (32 tokens), summed back to Z.



$\mathcal{L}_{\text{rec}}$
the sum $\hat{Z}_S + \hat{Z}_C$ reconstructs Z (i)

$\mathcal{L}_{\text{pin}}$
$\hat{Z}_S = 0$ on background, $\neq 0$ on target (ii)

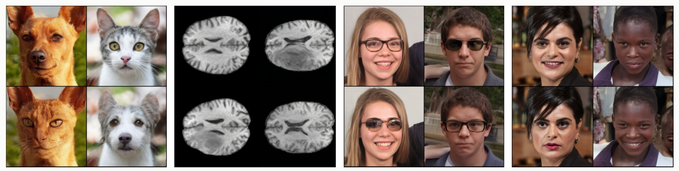
$\mathcal{L}_{\text{adv}}$
$\hat{Z}_C$ carries no information on Y (gradient reversal)

$\mathcal{L}_{\text{cyc}}$
swapped codes re-encode consistently (iii)

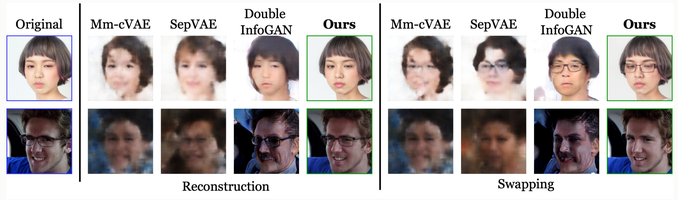
4 Results

Table 1. FFHQ, glasses. Swap accuracy (%) on fine-grained classes (none, reading, sunglasses), learned from binary labels only. ID-Sim, Gender, Smile (%) and Pose (MAE, °) measure how well common attributes are preserved by the swap.

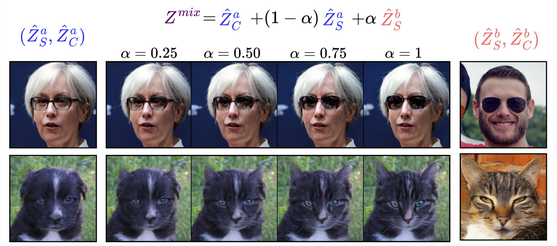
Method	SSIM ↑	LPIPS ↓	Acc ↑	ID-Sim ↑	Gender ↑	Smile ↑	Pose ↓
MM-cVAE	.370	.643	11.8	.302	62.3	57.7	4.62
SepVAE	.358	.642	19.6	.301	57.7	56.4	8.29
DoubleInfoGAN	.326	.430	47.0	.314	68.5	79.2	7.35
Diff-CA (ours)	**.610**	**.115**	**94.5**	**.474**	**92.5**	**92.9**	**1.87**



Swapping the salient factor on AFHQ (cat), BraTS (tumor) and FFHQ (glasses, smile). Top: real background and target images; bottom: after the swap.



VAE- and GAN-based CA baselines swap the attribute, but produce blurry images and lose identity.



Salient interpolation with $\hat{Z}_C$ fixed: the glasses style (top) and cat features (bottom) change progressively.

Ablations

Without $\mathcal{L}_{\text{adv}}$, no separation occurs (at most 2.1% swap accuracy); adding it gives 91.4%, and $\mathcal{L}_{\text{cyc}}$ 94.5% while removing color shifts. On EncDiff tokens of the same size, our separator reaches 58.1%, still above all CA baselines: both stages matter.