DynaVect: Context-Aware Modulation of Global Edit Directions for Controllable GAN Editing

Anonymous authors
Paper under double-blind review

Abstract

Text-guided editing of generative models like StyleGAN has become a popular method for image manipulation. Current approaches face a trade-off. Optimization-based methods produce edits that are too subtle. This fails to meet the user's intent for changes. On the flip side, methods that use a single global edit vector often cause unwanted attribute entanglement and identity loss. In this work, we propose DynaVect, a hybrid approach that attempts to resolve this trade-off. Our approach is a lightweight Dynamic Contextual Modulator. The DCM is a network trained to predict a personalized correction (or delta), based on the source image's features. At inference time, this learned delta is used to steer the global edit direction. This results in changes that are visually different while attempting to preserve the original identity. We train our modulator using an optimization-distillation technique. This technique involves creating a fast feed-forward model that approximates the quality of slow, per-image optimization. We demonstrate that our method produces qualitatively superior results that better align with users expectations as compared to traditional metrics.

1 Introduction

Generative Adversarial Networks (GANs), particularly StyleGAN2 Karras et al. (2020), have achieved impressive performance in generating photorealistic images. A key area of research, is developing controls for editing these images. The rise of vision-language models (VLMs) like CLIP Radford et al. (2021) has enabled text-guided manipulation of the GAN latent space.

However, existing methods present a dilemma for users, as illustrated in Fig. 1. Optimization-based techniques Patashnik et al. (2021) excel at preserving the subject's identity, resulting in low perceptual error. However, this often leads to overly cautious edits that fail to make significant changes requested by the user. Meanwhile, methods using a single global edit vector can make strong changes but often suffer from attribute entanglement, and in some cases, completely altering identity and background features.

In an attempt to solve this, we introduce DynaVect, a framework that finds the middle ground. Our contribution is a hybrid system that combines the strengths of both approaches. We start with a pre-computed global direction. This starting point provides the power for significant edits. If we started with something "lighter", tuning down the line will be challenging. Next, we use a lightweight, trained Dynamic Contextual Modulator to predict a correction vector based on the specific features of the source image. This allows the edit to be strong and personalized.

Furthermore, we deviated from the traditional method of training a modulator. Instead of training with a simple CLIP loss, we use a form of optimization-distillation. We generate pseudo ground-truth edits via a slow optimization process with fixed loss weights. Following that, we will train our fast, feed-forward modulator to replicate these high-quality results. Our qualitative results and a user study (n=58) show that DynaVect produces edits that are judged by humans as being more successful than traditional baselines. In this process, we also aim to highlight the current limitations of traditional metrics, especially in this area of research.

Figure 1: DynaVect successfully performs a complex, multi-attribute edit ("smile + age + hair"). **StyleCLIP-O (Optimizer)** produces an overly subtle edit, failing to make the requested changes. **StyleCLIP-G (Global Direction)** applies the changes but loses the subject's identity. **DynaVect (Ours)** achieves the best balance, implementing all attributes while preserving identity.

2 Related Work

Our work is situated at the intersection of generative image models and text-guided image manipulation, building upon recent advancements in VLMs.

Generative Adversarial Networks. Generative Adversarial Networks (GANs), introduced by Goodfellow et al. Goodfellow et al. (2014), have become the standard for high-fidelity image synthesis. Our work utilizes StyleGAN2 Karras et al. (2020), which refines the architecture of StyleGAN Karras et al. (2019). A key design in the StyleGAN family is the intermediate latent space $\mathcal{W}$, obtained by mapping a noise vector $z \in \mathcal{Z}$ through a learned network. Many editing methods operate in $\mathcal{W}^+$, the per-layer extension of $\mathcal{W}$ that supplies a separate w code to each synthesis layer (e.g., 18 layers for the 1024px generator), enabling fine, layer-wise control over semantics and making the model particularly responsive to editing.

Text-Guided Image Editing. The introduction of large-scale VLMs, most notably CLIP Radford et al. (2021), revolutionized text-guided manipulation. CLIP's ability to embed both images and text into a semantically rich space allows for the measurement of text and image alignment via cosine similarity. Changes can now be described in natural language, rather than using a manual process.

Improving on Context-Agnostic Mappers. The work most closely related to ours is StyleCLIP Patashnik et al. (2021). This paper demonstrated three distinct methods for using CLIP to guide StyleGAN edits. Our work is an extension of their third and most efficient approach: the latent mapper.

While the original StyleCLIP mapper learns a single global edit vector for a given text prompt, a "one-size-fits-all" approach can struggle. This is because the edit often depends on the source image's unique features. A global vector that successfully adds a smile to one face may fail or cause unwanted artifacts on another. Our work directly addresses this limitation. Instead of applying a static global vector, we introduce a **Dynamic Contextual Modulator** that predicts a personalized correction based on the source image's features. This allows our method to steer the global direction, making the final edit more robust while retaining the subject's identity.

3 Method

Our goal is to perform text-guided edits that follow the prompts, are visually significant, and preserve the identity of the source image. DynaVect combines (i) a Global Direction Bank for strong baseline edits and

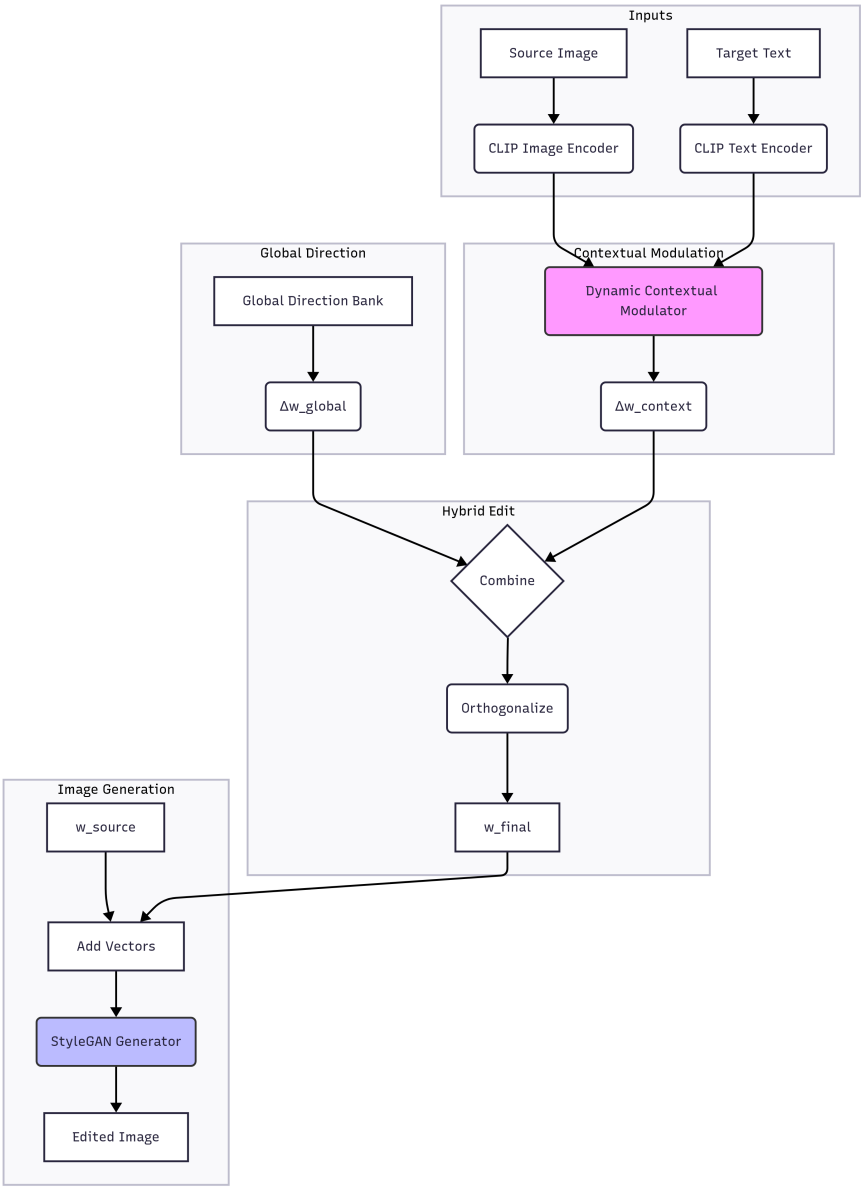

Figure 2: The DynaVect architecture. A source image and target text are encoded by CLIP. These features are fed into our trained **Dynamic Contextual Modulator (DCM)** to predict a context-aware correction, $\Delta w_{\text{context}}$. This is added to a pre-computed **Global Direction**, Δw_{global}. The combined vector undergoes **Orthogonalization** (enabled in main results) to reduce drift in protected attributes (identity by default) before being applied to the source latent code, w_{source}, to generate the final edited image.

(ii) a lightweight Dynamic Contextual Modulator for personalized correction. An orthogonalization step is enabled by default in the main results to reduce drift in protected attributes (identity by default). Additional protected attributes (e.g., gender/age) are off unless stated. We ablate its effect in Appendix C.

3.1 Global Direction Baseline

For a given text pair (T_S, T_T), we first compute a strong, generalizable edit direction. We sample a large batch of latent codes $\{w_i\}$ from the StyleGAN W+ space and score each corresponding image $G(w_i)$ against the source and target prompts using CLIP. The global direction Δw_{global} is then calculated as the difference between the means of the top-scoring latents for each prompt. This generates a baseline vector for the edit, although it may contain entangled features.

3.2 Dynamic Contextual Modulator

To personalize the edit, we introduce our modulator network, M. This is a MLP that takes two inputs, namely (1) CLIP image features of the source image $E_I(G(w_{\text{source}}))$ and (2) the CLIP text features of the target prompt $E_T(T_T)$. Its purpose is to predict a contextual correction vector, or "delta":

$$\Delta w_{\text{context}} = M(E_I(G(w_{\text{source}})), E_T(T_T)) \tag{1}$$

This allows the correction to be sensitive to the unique characteristics of the input face.

3.3 Disentanglement via Orthogonalization (Enabled in main results)

To keep key attributes stable (identity by default), we remove any part of the edit that points along a small set of protected directions$\mathcal{B}$ (identity by default). Concretely,

$$\Delta w_{\text{ortho}} = \Delta w_{\text{final}} - \text{Proj}_{\mathcal{B}}(\Delta w_{\text{final}}).$$

In code we use a simple sequential version: starting from $v = \Delta w_{\text{final}}$, for each $b_j \in \mathcal{B}$ we update

$$v \leftarrow v - \langle v, \hat{b}_j \rangle \hat{b}_j, \quad \text{with } \hat{b}_j = b_j / \|b_j\|.$$

This matches the subspace projection when the protected directions are orthogonal. Otherwise, it depends on the order. In our reported results, we protect a single direction(identity) to prevent order issues. We estimate the identity direction $\mathcal{B} = \{b_{\text{id}}\}$ with the same sampling used for global directions, using the neutral text pair("a face" vs "another face").

We evaluate two variants in ablations: (i) disabling identity orthogonalization, and (ii) a "Global+Ortho" variant that handles multi-attribute edits (e.g., "Smile + Age") by first applying the initial global direction, then orthogonalizes each subsequent global direction against all previously applied directions within that same call. Edit history is also not maintained across separate calls, and this variant does not use identity projection. Orthogonalization can slightly change the ΔCLIP/identity trade-off. Therefore, we report both settings and use identity orthogonalization in the main tables.

3.4 Hybrid Edit Application

At inference time, the final edit direction is a combination of the global baseline and the contextual delta, controlled by a modulation strength parameter α:

$$\Delta w_{\text{final}} = \Delta w_{\text{global}} + \alpha \cdot \Delta w_{\text{context}} \tag{2}$$

This hybrid vector is then applied to the source latent code w_{source} to produce the final edit.

3.5 Training via Optimization-Distillation

Training the modulator presents a challenge, as there is no "ground truth" for the ideal $\Delta w_{\text{context}}$. Therefore, we propose a self-supervised training scheme. For a given source latent w_{source} and target text T_T, we run an iterative optimization process to find an "ideal" edited latent w'_{ideal} that minimizes a joint CLIP and LPIPS identity loss.

We define the ground-truth contextual delta as $\Delta w_{\text{gt}} = w'_{\text{ideal}} - w_{\text{source}}$. The training objective for our modulator M is to minimize the L2 distance between its prediction and this "ground-truth":

$$\mathcal{L} = \|M(\dots) - \Delta w_{\text{gt}}\|_2 \tag{3}$$

This distills the knowledge from the slow optimization into our feed-forward modulator. Additionally, we also use a reconstruction loss in image space to ensure visual fidelity.

4 Optional Adaptive Loss Weighting (Future Exploration)

This section describes an exploratory module; an Adaptive Loss Balancer (ALB) that predicts an identity-loss weight λ_{id} from text embeddings to modulate the optimizer used to generate pseudo ground truth. The ALB was not enabled for the results reported in this paper. All reported experiments use fixed loss weights during optimization. We include ALB as a potential for future exploration.

If enabled, ALB would replace the fixed identity weight with a predicted value:

$$\mathcal{L}_{\text{opt}} = \mathcal{L}_{\text{CLIP}} + \lambda_{id} \cdot \mathcal{L}_{\text{LPIPS}} + \mathcal{L}_{\text{L2}} \tag{4}$$

In all reported results, λ_{id} is a constant hyperparameter.

5 Experiments

We evaluate two questions: (1) does **DynaVect** better align with user intent than standard baselines, and (2) are common automated metrics (LPIPS/L2) adequate for more significant semantic edits? Unless otherwise noted, the ALB is disabled; optimization uses fixed weights (CLIP: 1.0, LPIPS: 0.8, latent-L2: 0.5).

5.1 Experimental Setup

Implementation details. We use StyleGAN2-FFHQ Karras et al. (2020) and CLIP ViT-B/32 Radford et al. (2021). The Dynamic Contextual Modulator is trained for 2000 steps with Adam (1×10^{-4}).

Baselines. We compare DynaVect against two representative baselines from the StyleCLIP framework Patashnik et al. (2021):

- **StyleCLIP (Optimizer):** A slower and iterative optimization of the latent code to minimize a joint CLIP and LPIPS loss.
- **StyleCLIP (Global Dir):** A fast and direct method that applies a single pre-computed vector to the latent code.

Metrics and protocol. We report: (i) Directional CLIP (ΔCLIP) with neutral text “a face”; (ii) identity via ArcFace cosine (InsightFace “antelopev2”) Deng et al. (2019); Guo et al. (2025); Contributors (2023) on *non-occluding* edits only; and (iii) LPIPS (alex) Zhang et al. (2018) and L2. We evaluate 10 random StyleGAN2 seeds in $\mathcal{W}^+$ with truncation ψ=0.7. For occluding edits (e.g., sunglasses), face recognition is unreliable, so we omit ArcFace and report ΔCLIP and qualitative/user preference instead. ArcFace detection succeeded on >90% of our non-occluding test images.

5.2 Quantitative Comparison

We compare **DynaVect** with StyleCLIP's global-direction (**StyleCLIP-G**) and optimizer (**StyleCLIP-O**). Table 1 covers non-occluding edits where ArcFace is reliable; Table 2 covers an occluding edit (sunglasses) where identity is omitted.

Reproducibility. Global-direction sampling uses 2,000 latents, top-k=20 per prompt, batch size 20, truncation ψ=0.7. StyleCLIP-O uses 100 optimization steps (Adam, lr=10^{-2}) with loss weights CLIP:1.0, LPIPS:0.8, latent-L2:0.5. DCM is trained for 2,000 steps (Adam, lr=10^{-4}). For DynaVect main runs we enable identity-orthogonalization as described in Sec. 3. Evaluation uses generator outputs without external face preprocessing. The Adaptive Loss Balancer is disabled for all reported numbers. Code and checkpoints will be released.

DynaVect improves identity over StyleCLIP-G on *Smile* and *Age+*, and attains higher ΔCLIP on multi-attribute (non-occluding) edits with a slight identity trade-off. StyleCLIP-O achieves the best identity/LPIPS by minimally editing, inflating similarity metrics.

Method	Task	ΔCLIP ↑	Identity ↑	LPIPS ↓	L2 ↓
DynaVect	Smile	0.0247 ± 0.0052	**0.672** ± 0.067	0.303 ± 0.048	0.0743 ± 0.0276
StyleCLIP-G	Smile	**0.0316** ± 0.0025	0.519 ± 0.106	0.367 ± 0.066	0.0991 ± 0.0289
StyleCLIP-O	Smile	0.0380 ± 0.0074	<u>0.892</u> ± 0.038	<u>0.020</u> ± 0.004	<u>0.0019</u> ± 0.0006
DynaVect	Age+	0.0105 ± 0.0076	**0.372** ± 0.105	0.402 ± 0.050	0.1059 ± 0.0234
StyleCLIP-G	Age+	0.0103 ± 0.0114	0.277 ± 0.062	0.495 ± 0.037	0.1690 ± 0.0708
StyleCLIP-O	Age+	**0.0568** ± 0.0264	<u>0.836</u> ± 0.044	<u>0.026</u> ± 0.006	<u>0.0023</u> ± 0.0008
DynaVect	Blonde	0.0545 ± 0.0188	0.461 ± 0.144	0.437 ± 0.071	0.2010 ± 0.0597
StyleCLIP-G	Blonde	0.0534 ± 0.0219	**0.596** ± 0.113	**0.305** ± 0.052	**0.0919** ± 0.0337
StyleCLIP-O	Blonde	**0.1225** ± 0.0230	<u>0.824</u> ± 0.056	<u>0.031</u> ± 0.007	<u>0.0063</u> ± 0.0030
DynaVect	Combo-NonOcc	**0.0659** ± 0.0114	0.250 ± 0.084	0.507 ± 0.048	0.2106 ± 0.0466
StyleCLIP-G	Combo-NonOcc	0.0596 ± 0.0192	**0.297** ± 0.082	**0.480** ± 0.048	**0.2196** ± 0.0594
StyleCLIP-O	Combo-NonOcc	<u>0.1066</u> ± 0.0332	<u>0.811</u> ± 0.047	<u>0.032</u> ± 0.005	<u>0.0047</u> ± 0.0012

Table 1: Non-occluding edits. **Bold** = best among feed-forward/global methods (ours vs. StyleCLIP-G); <u>underline</u> = optimizer (best overall but under-edits). DynaVect improves identity vs. StyleCLIP-G on Smile and Age+, and achieves higher ΔCLIP on Combo-NonOcc with a modest identity trade-off.

Method	Task	ΔCLIP ↑	LPIPS ↓	L2 ↓
DynaVect	Combo-Occ	0.1050 ± 0.0179	0.558 ± 0.042	0.2036 ± 0.0730
StyleCLIP-G	Combo-Occ	0.1020 ± 0.0176	0.609 ± 0.046	0.2850 ± 0.0956
StyleCLIP-O	Combo-Occ	0.1211 ± 0.0321	<u>0.033</u> ± 0.003	<u>0.0037</u> ± 0.0009

Table 2: Occluding edit (sunglasses). ArcFace identity is unreliable due to occlusion and is omitted.

5.2.1 Results Summary

- **Versus StyleCLIP-G (feed-forward baseline).** On non-occluding edits, DynaVect improves identity on *Smile* and *Age+* while remaining competitive on ΔCLIP; on the multi-attribute *Combo-NonOcc* task, DynaVect attains the highest ΔCLIP among feed-forward methods again with a modest identity trade-off. For *Blonde*, StyleCLIP-G retains higher identity/LPIPS.
- **Versus StyleCLIP-O (optimizer baseline).** StyleCLIP-O achieves the strongest LPIPS/L2 (minimal change) and often high ΔCLIP, yet our user study (n=58) shows participants prefer DynaVect in 89.7% (vs. Optimizer) and 86.2% (vs. Global) of trials, indicating better alignment with the real world user intent despite having less favorable "similarity" metrics.

- **Occluding edit (sunglasses).** When identity is unreliable and omitted, DynaVect slightly improves ΔCLIP over StyleCLIP-G and reduces LPIPS/L2, suggesting stronger, cleaner edits in the occluding case.
- **Orthogonalization ablation.** In the main tables, we enable *identity-only* orthogonalization *for DynaVect outputs* (StyleCLIP baselines are unchanged). In ablations we (i) disable orthogonalization for DynaVect and (ii) evaluate a "Global+Ortho" variant that orthogonalizes each newly added *global* direction against the directions already applied *within the same multi-attribute call*. Identity orthogonalization can modestly shift the ΔCLIP/identity trade-off (App. C).
- **Overall.** Automated scores are inconsistent proxies for edit success. LPIPS/L2 reward minimal changes, and ΔCLIP can be high even for outputs with almost no visible changes. The results support DynaVect as a better balance of edit strength and identity preservation, compared to a static global direction.

Takeaways. Trends match our qualitative grids and user study. No single automated metric captures edit success fully. LPIPS/L2 reward minimal change, while ΔCLIP can be high even for edits that appear subtle to humans. Therefore, we report both ΔCLIP and ArcFace (when applicable) but rely mainly on user preferences as the primary measure of success.

Ablation (summary). "Global+Ortho" slightly improves ΔCLIP but hurts identity, while DynaVect yields the best identity at competitive ΔCLIP. Full results are in Appendix C.

As shown in Tables 1 and 2, a somewhat naive interpretation of the metrics would suggest the Optimizer baseline is the best. In the studies, it achieves the lowest perceptual and pixel-level change (LPIPS, L2). However, this quantitative success is misleading and reveals a fundamental paradox in using such metrics for semantic editing. LPIPS and L2 are designed to measure similarity to a source image, operating on the assumption that the source is the ground truth. While this is valid for tasks like image restoration, it contradicts the goal of editing, which is to create a meaningful difference. For significant semantic edits like "add age," a large perceptual change is not only expected but *required*. The Optimizer, when heavily penalized by a metric that punishes change, "cheats" the edit's intent by making almost no visible changes at all. This allows it to achieve an almost perfect score on the metric while completely failing the actual goal. This highlights a limitation in standard metrics and underscores the necessity of the qualitative and user-based evaluation. We also observe cases where StyleCLIP-O attains high ΔCLIP despite minimal perceived change. This indicates that ΔCLIP alone can overestimate semantic success when visual salience is low. Therefore, this motivates our use of user studies.

5.3 Qualitative Results

Qualitative evaluation reveals the true performance of each method. As shown in the examples in Fig. 3, the Optimizer baseline barely makes any changes, resulting in an image that is nearly identical to the original. The Global Direction baseline successfully changes attributes but at times degrade the subject's identity. Our method, DynaVect, successfully implements changes while maintaining a higher degree of fidelity to the original subject's core features. This demonstrates a better balance that is not captured by traditional metrics.

5.4 User Study

Given the failure of automated metrics to align with human judgment, we conducted a user study to evaluate which method produces more successful edits. We recruited 58 anonymous participants on Amazon Mechanical Turk. Each participant completed 58 two-alternative forced-choice (2AFC) trials. In each trial, we displayed the text prompt and two images (our method vs. a baseline (StyleCLIP-O or StyleCLIP-G)) The participants were asked "Which image better matches the text prompt?".

The results show a strong user preference for our method.When compared against the Optimizer baseline, our method was preferred in 89.7% of cases. Against the Global Direction baseline, our method was preferred

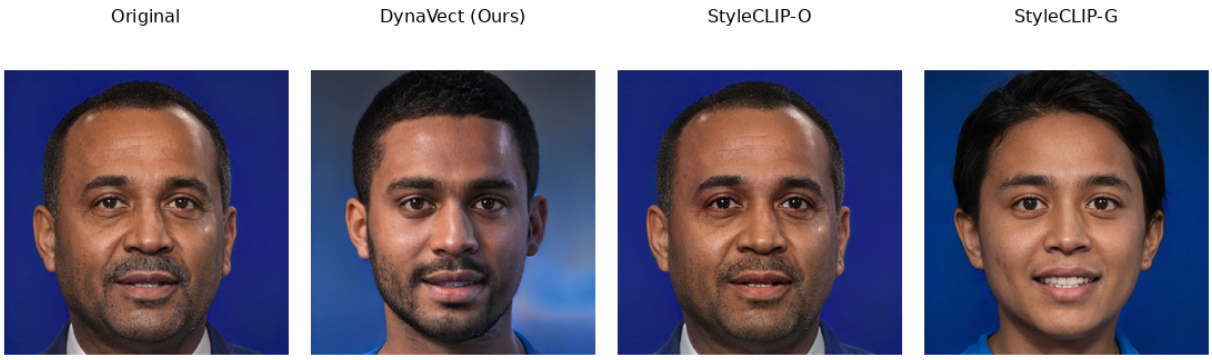

Figure 3: Qualitative comparison on two different subjects. In both cases, the Optimizer baseline (StyleCLIP-O) produces minimal changes for complex edits like aging or changing hair color. The Global Direction baseline (StyleCLIP-G) applies the edits but damages the subject's identity. Our method, DynaVect, successfully implements the attributes while preserving the original identity.

in 86.2% of cases.This confirms that DynaVect's edits better align with user intent, achieving a superior balance between edit strength and identity preservation.

5.5 Training Stability

The training of our Dynamic Contextual Modulator was observed to converge well. Figure 4 shows the progression of the key loss components during our optimization-distillation process. The decrease in both the direction and reconstruction losses indicate that the modulator successfully learned to approximate the target edits.

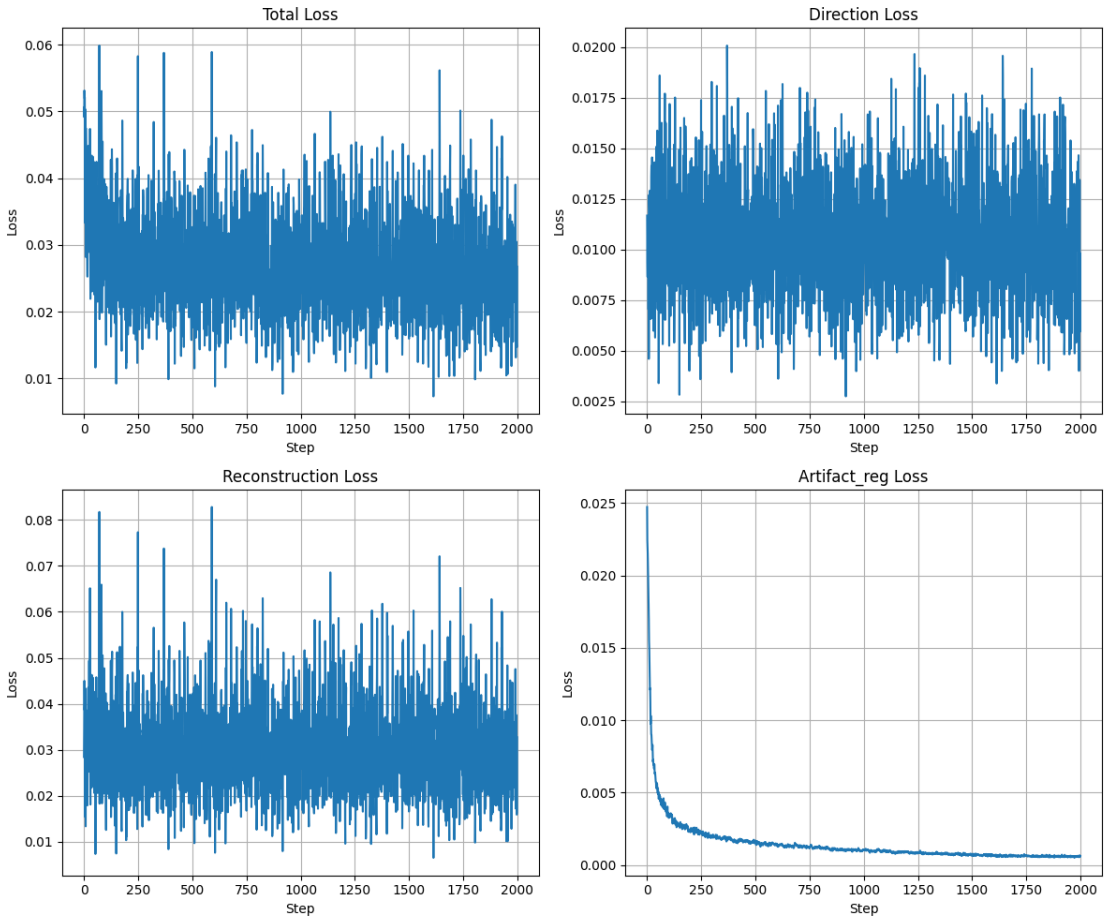

Figure 4: Training loss curves for the Dynamic Contextual Modulator. The model shows stable convergence, with the regularization loss (bottom right) decreasing as expected.

6 Conclusion

We have presented DynaVect, a hybrid framework that improves the controllability of text-guided GAN editing. Our key contribution is a lightweight Dynamic Contextual Modulator, which personalizes a global edit vector based on the features of the source image. By training this modulator with an optimization-distillation technique, our method produces significant semantic changes that better align with user intent, while preserving subject identity. We enable identity-orthogonalization in the main results and ablate its effect (including a no-orthogonalization variant) in Appendix C.

References

InsightFace Contributors. Insightface model pack: antelopev2. `https://pypi.org/project/insightface/0.7.3/`, 2023. Model zoo pack with SCRFD-10GF detector and ResNet100@Glint360K recognizer (ArcFace). Accessed: 2025-08-17.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. URL `https://arxiv.org/abs/`

1406.2661.

Jia Guo, Jiankang Deng, and InsightFace Contributors. Insightface: 2d and 3d face analysis project. https://github.com/deepinsight/insightface, 2025. Open-source toolbox used for detection/embedding runtime. Accessed: 2025-08-17.

Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, pp. 4401–4410, 2019.

Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8107–8116, 2020. doi: 10.1109/CVPR42600.2020.00813.

Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2065–2074, 2021. doi: 10.1109/ICCV48922.2021.00209.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 586–595, 2018. doi: 10.1109/CVPR.2018.00068.

Appendix

A Modulator Architecture

The detailed architecture of our Dynamic Contextual Modulator is provided in Table 3. The network is a standard Multi-Layer Perceptron (MLP) that processes concatenated CLIP features.

Layer Type	**Parameters**
Input	2 x CLIP Features (Total: 1024-dim)
Linear	1024 → 2048
LayerNorm, LeakyReLU, Dropout(0.1)	-
Linear	2048 → 2048
LayerNorm, LeakyReLU, Dropout(0.1)	-
Linear	2048 → 1024
LayerNorm, LeakyReLU	-
Layer Heads (x18)	1024 → 512 (w_dim)
Attention Head	1024 → 18 (num_layers) + Softmax

Table 3: Architecture of the Dynamic Contextual Modulator.

B Adaptive Loss Balancer Details

The Adaptive Loss Balancer is a simple MLP trained to predict the ideal identity loss weight for our optimization-distillation process. The architecture is detailed in Table 4.

Layer Type	**Parameters**
Input	2 x CLIP Features (Total: 1024-dim)
Linear	1024 → 128
ReLU	-
Linear	128 → 1
Sigmoid	-

Table 4: Architecture of the Adaptive Loss Balancer.

We train the balancer on a small dataset where we manually assign an ideal weight based on the edit's expected magnitude. A higher weight signifies stronger identity preservation. Example pairs from our training set include:

- `("a face", "a smiling face")` → 0.9
- `("a young person", "an old person")` → 0.6
- `("a face", "a face with sunglasses")` → 0.2

C Ablation Study

D Additional Qualitative Results

This section provides the remaining qualitative results grids to demonstrate the consistency of our method across various subjects.

Method	ΔCLIP ↑	Identity ↑	LPIPS ↓
Global Only	0.0378 ± 0.0105	0.5951 ± 0.0466	0.4049 ± 0.0466
Global + Ortho	**0.0436** ± 0.0179	0.5529 ± 0.0508	0.4471 ± 0.0508
DynaVect (Ours)	0.0278 ± 0.0147	**0.6573** ± 0.0501	**0.3427** ± 0.0501

Table 5: Ablation across method families. "Global+Ortho" orthogonalizes each new global direction to the directions already applied *within the same multi-attribute call*; DynaVect uses identity orthogonalization in the main tables. Orthogonalization changes the ΔCLIP/identity trade-off; DynaVect achieves the best identity at competitive ΔCLIP.

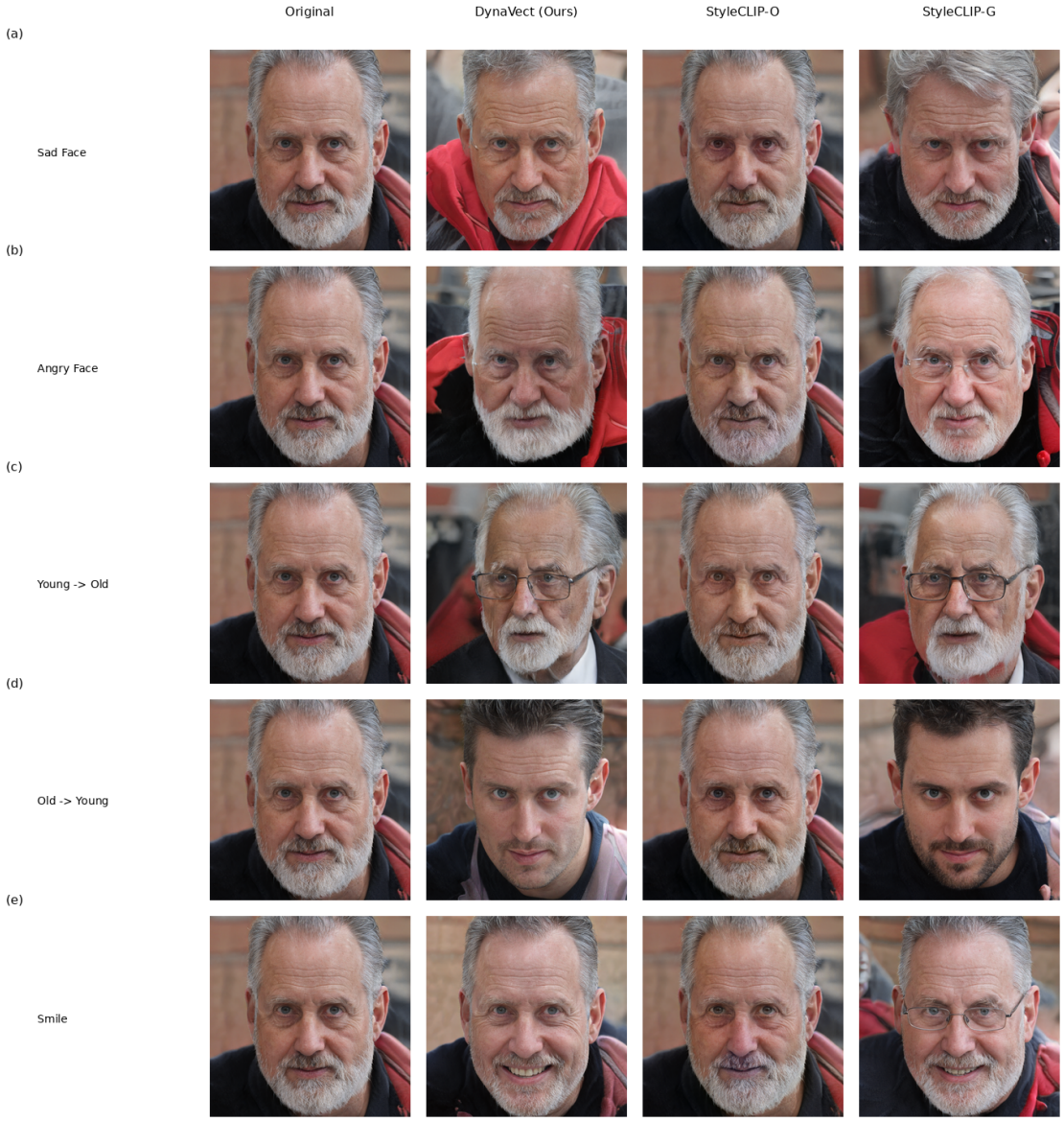

Figure 5: Additional qualitative results for Seed 494.

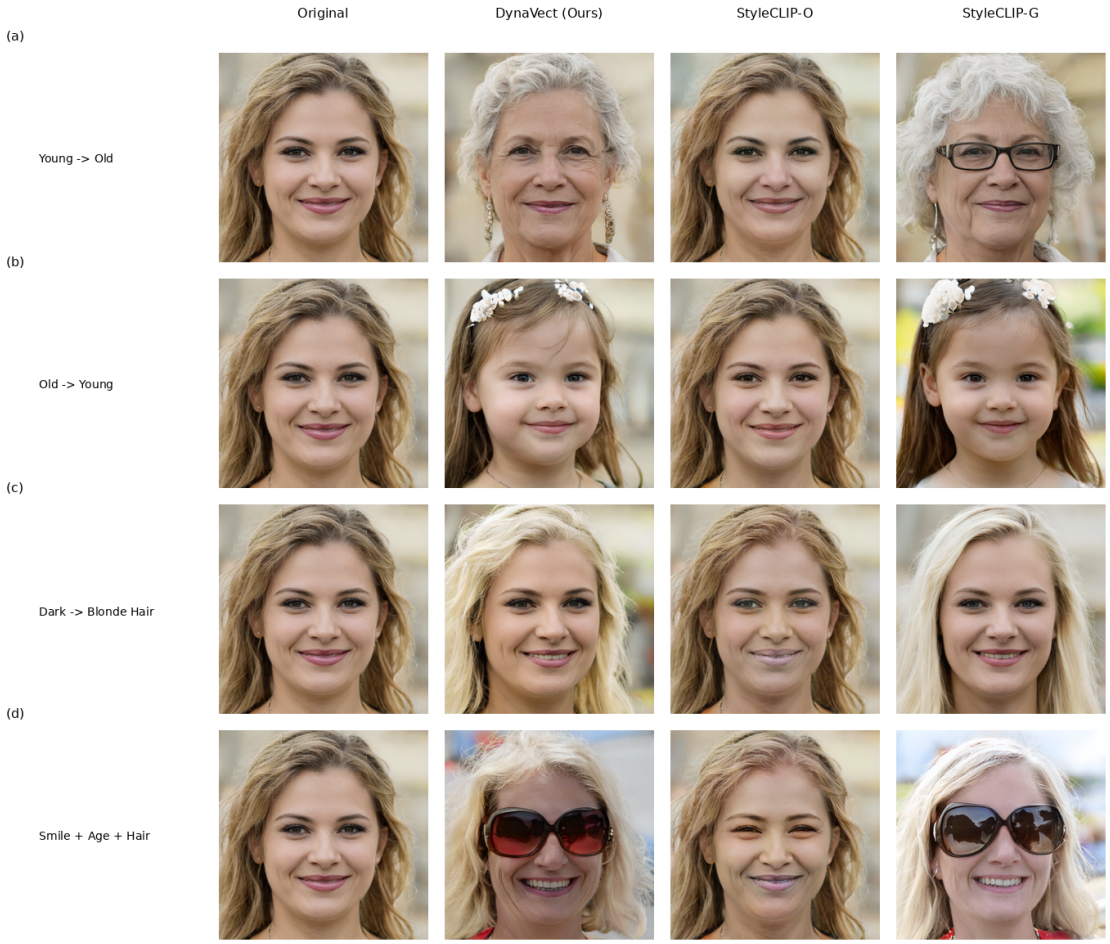

Figure 6: Additional qualitative results for Seed 95.

Figure 7: Additional qualitative results for Seed 888.

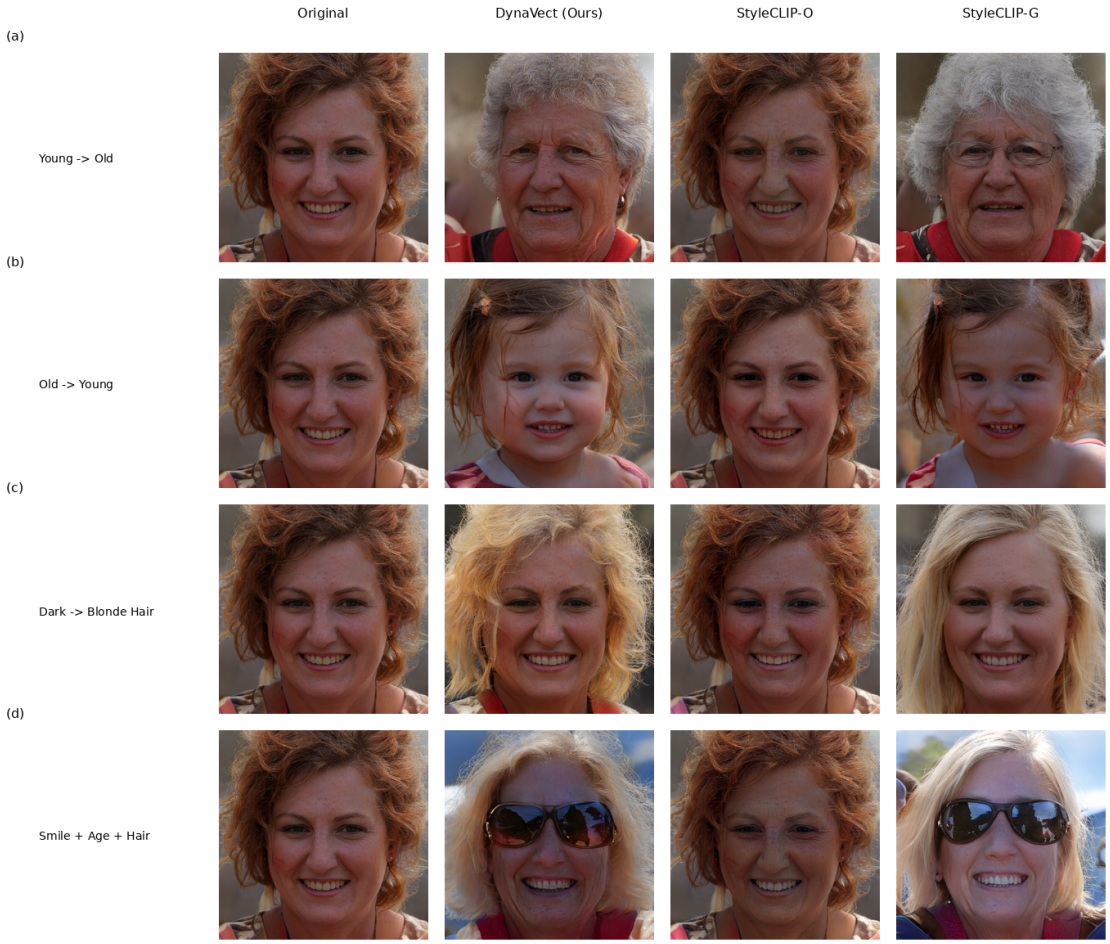

Figure 8: Qualitative results for Seed 9999.

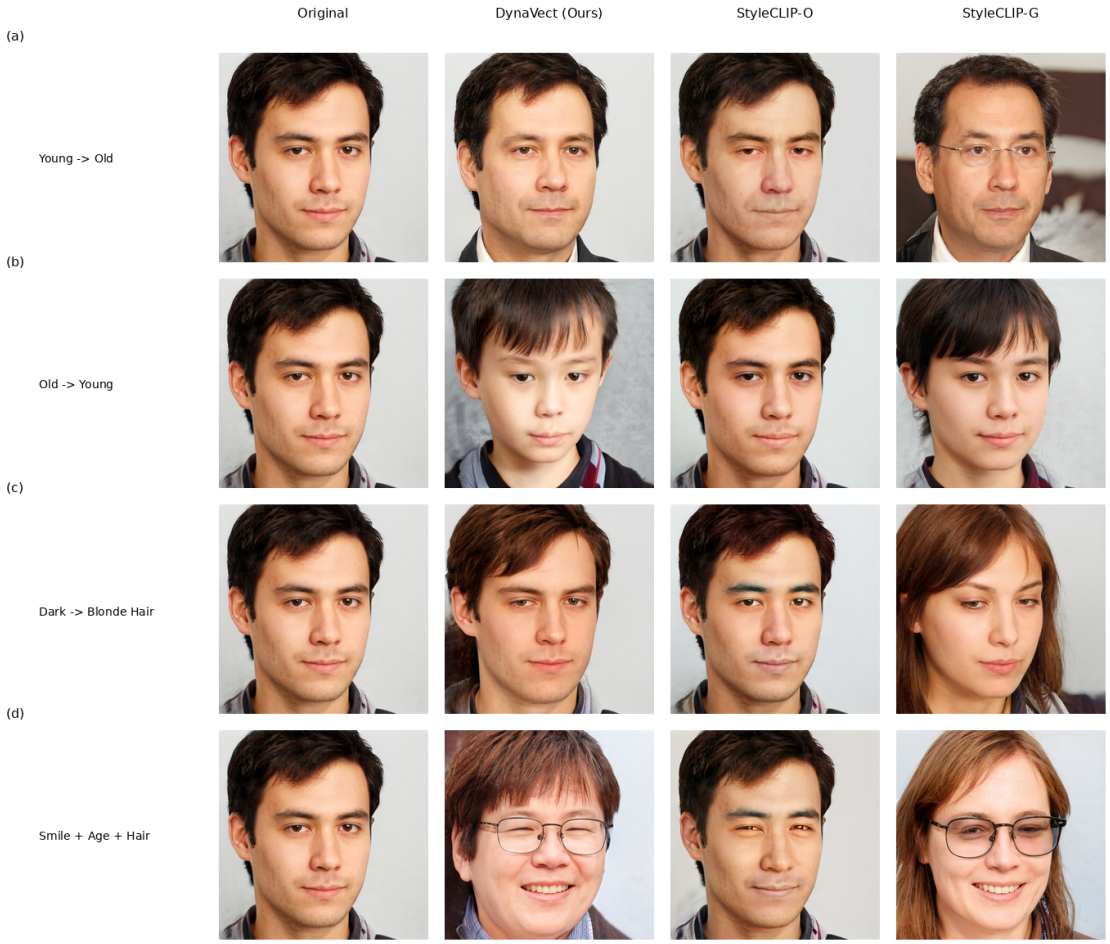

Figure 9: Failure case due to concept entanglement. The gender-neutral prompt "a person with dyed blonde hair" incorrectly feminizes the male subject.

E Limitations and Failure Cases

While DynaVect demonstrates robust performance on these edits, it is still subject to the limitations of the underlying StyleGAN and CLIP models. In this section, we present a qualitative analysis of common failure modes.

E.1 Dataset Bias and Concept Entanglement

We observe failures that stem from strong dataset biases within the base models, which leads to concept entanglement. For example, the concept of "blonde hair" is heavily correlated with the concept of "woman" in the training data. As a result, applying a gender-neutral prompt like "a person with dyed blonde hair" to a male subject can cause an unwanted shift in gender towards female, as shown in Fig. 9.

E.2 Large Identity Transformations

The framework is designed for attribute editing, not identity transformation. Attempts to perform large-scale changes like altering ethnicity, may result in some non-photorealistic outputs. Figure 10 and Figure 11 show failed attempts to transform Caucasian faces into Asian faces. **Ethical considerations.** Editing facial

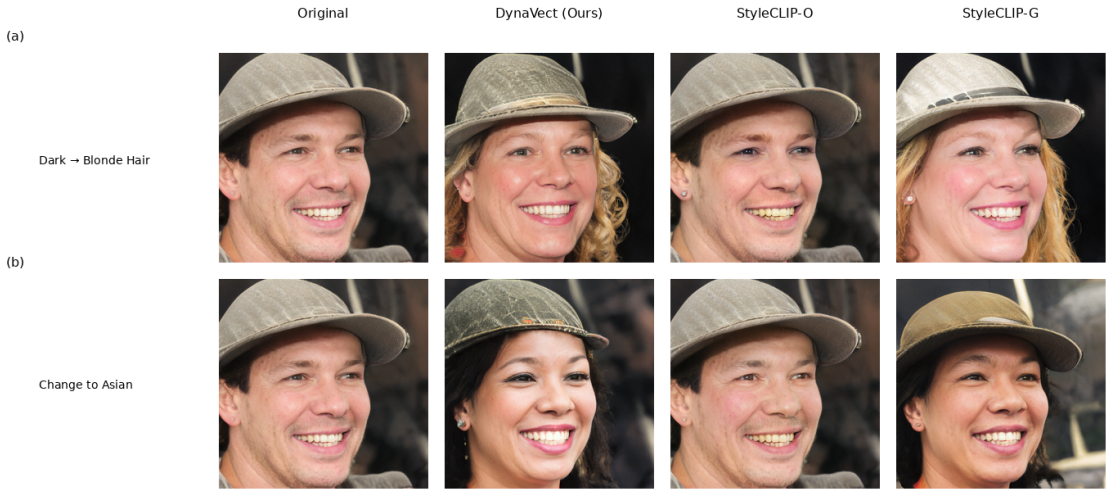

Figure 10: Failure case for a large identity transformation (Subject 1). The prompt was "an Asian face."

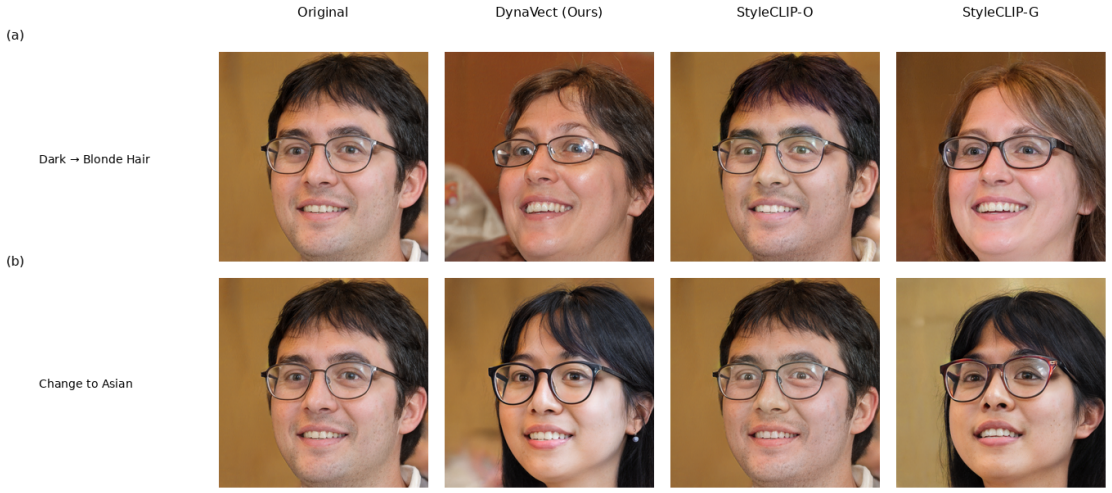

Figure 11: Failure case for a large identity transformation (Subject 2). The prompt was "an Asian face."

attributes can amplify dataset bias and may be misused. We limit evaluation to synthetic FFHQ-like faces, avoid identity conversion tasks, and do not deploy for biometric applications.