FICE: Text-Conditioned Fashion-Image Editing With Guided GAN Inversion

Supplementary Material

Martin Pernuš[a,*], Clinton Fookes[b], Vitomir Štruc[a], Simon Dobrišek[a]

[a]*Faculty of Electrical Engineering, University of Ljubljana, Trzaska 25, Ljubljana, 1000, Slovenia, Slovenia*
[b]*School of Electrical Engineering & Robotics, Queensland University of Technology, 2 George St, Brisbane, 4000, Queensland, Australia*

Abstract

In the main part of the paper we reported several results to highlight the capabilities of FICE. In this *Supplementary material* we present additional details and experiments to further explore the characteristics of FICE, including: (*i*) details of the hyperparameter settings for the competing techniques (pSp, E4e, ReStyle, and HyperStyle) used in the main part of the paper, (*ii*) analysis of CLIP performance in the fashion domain, *iii*) designing a stochastic FICE, (*iv*) investigations into alternative latent-code initialization schemes (with style mixing), (*iv*) additional results on the MPV image dataset, (*v*) details of the results by individual metrics, (*vi*) execution time analysis, (*vii*) provide implementation details, and (*viii*) describe the implications to other related scientific fields..

1. Hyperparameter Settings

In the main part of the paper we considered several baseline GAN-inversion techniques combined with StyleCLIP to compare with FICE. These included pSp [1], E4e [2], ReStyle [3] with the pSp and E4e backbones, and HyperStyle [4]. The optimal hyperparameter (α, β) settings that were used with these methods are listed in Table 1 for completeness.

2. Analyzing CLIP Image–Text Understanding in the Fashion Domain

In this section we explore the CLIP capabilities in understanding fashion data. To this end, we take several input text sentences from the main manuscript and search for the nearest images in LAION [5], a large, publicly available, image–text paired dataset of 400 million samples with pre-computed CLIP image

[*] Corresponding author
Email addresses: martin.pernus@fe.uni-lj.si (Martin Pernuš), c.fookes@qut.edu.au (Clinton Fookes), vitomir.struc@fe.uni-lj.si (Vitomir Štruc), simon.dobrisek@fe.uni-lj.si (Simon Dobrišek)

Table 1: **Best-performing hyperparameter settings.** The best setting for each model was determined based on the semantic-relevance score.

Model	**Magnitude** (α)	**Disentanglement** (β)
pSp [1]	4.0	0.50
e4e [2]	4.0	0.025
ReStyle-pSp [3]	7.5	0.025
ReStyle-e4e [3]	7.5	0.050
HyperStyle [4]	5.0	0.050

embeddings. The closest image matches for a given input text and the angular similarity scores are shown in Fig. 1. We observe that CLIP already incorporates a good understanding of fashion data, providing convincing matches to the text prompts.

Next, we aim to gain a deeper understanding of the CLIP similarity scores. It is important to note that the CLIP training scheme was designed to maximize the positive pair cosine similarity in comparison to negative pairs, using cross-entropy loss. This implies that the raw values are used for ranking in comparison to other pairs rather than providing straightforward information about the pair similarity. We can compare example similarity scores in Fig. 1 to the average CLIP similarities, obtained by FICE and the compared methods. The best image matches tend to have a similarity score between 0.37 and 0.40. This similarity range closely follows the average scores of the compared methods. Meanwhile, FICE achieves a similarity score of 0.446, demonstrating its ability to generate images with high semantic relevance to the text prompt.

3. Stochastic Image Editing

The proposed FICE approach to fashion-image editing, as presented in the main manuscript, is a deterministic algorithm. That is, for a given input image and text, the edited image is always the same. This approach offers certain benefits: (*i*) the model behaves consistently, (*ii*) it is easier to evaluate and compare with its competitors, and (*iii*) the results are simpler to reproduce. However, a single textual description can correspond to an infinite number of plausible visual representations. Incorporating stochastic elements can allow the algorithm to explore a more diverse solution space, enhancing the naturalness of the generated images.

In order to produce stochastic results, we follow the StyleGAN2 [6] projection protocol, which involves adding noise to the latent code during the GAN-inversion process. We apply the following mapping before the latent code is evaluated by the auxiliary models:

$$w \leftarrow w + \mathcal{N}(0, 0.05\sigma_w t^2), \tag{1}$$

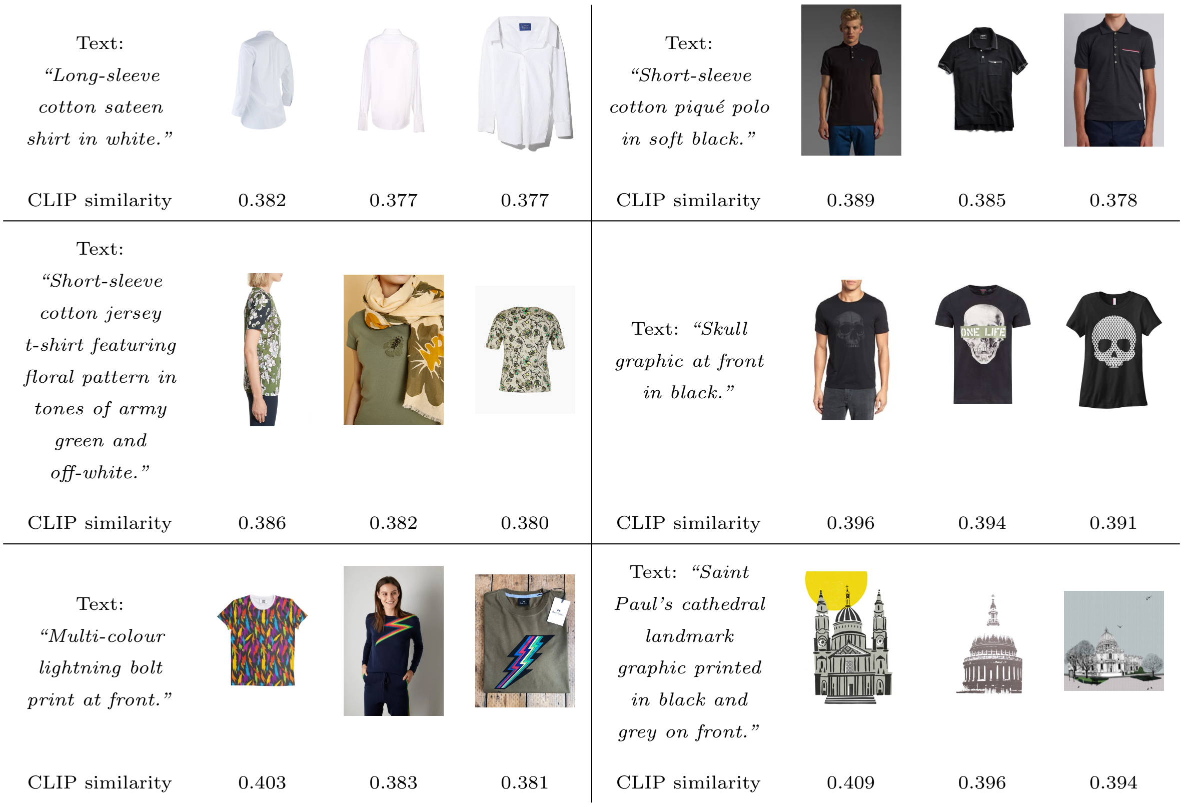

Text: “Long-sleeve cotton sateen shirt in white.”
CLIP similarity
0.382
0.377
0.377
Text: “Short-sleeve cotton piqué polo in soft black.”
CLIP similarity
0.389
0.385
0.378
Text: “Short-sleeve cotton jersey t-shirt featuring floral pattern in tones of army green and off-white.”
CLIP similarity
0.386
0.382
0.380
Text: “Skull graphic at front in black.”
CLIP similarity
0.396
0.394
0.391
Text: “Multi-colour lightning bolt print at front.”
CLIP similarity
0.403
0.383
0.381
Text: “Saint Paul’s cathedral landmark graphic printed in black and grey on front.”
CLIP similarity
0.409
0.396
0.394

Figure 1: **LAION images with the highest CLIP similarity given the text prompt.** The figure shows the three LAION images with the highest CLIP similarity score given the input text prompt. The values below denote the similarity score for each text–image pair.

where σ_w is the standard deviation of the latent codes and t goes from one to zero during the first 375 iteration steps of the GAN-inversion procedure. Example results produced with the stochastic extension are shown in Fig. 2. We observe that this extension allows us to reliably generate multiple possible results for a single image–text pair. The generated image edits are plausible given the input text description, while most of the changes stem from slight color changes as well as details in the clothing style.

4. Exploring Style Mixing for Code Initialization

As demonstrated in the experiments in the main part of the paper, FICE generates competitive, high-quality editing results when compared to state-of-the-art models from the literature. Nevertheless, we observe that in certain cases, the results generated by FICE are impacted by the characteristics of the input image. When the targeted semantics from the provided text description differ from the semantics already present in the input image (i.e., changing a plain white shirt to a dark-coloured shirt), FICE can sometimes produce unsatisfactory results. In this subsection, therefore, we explore alternative ways of latent-code

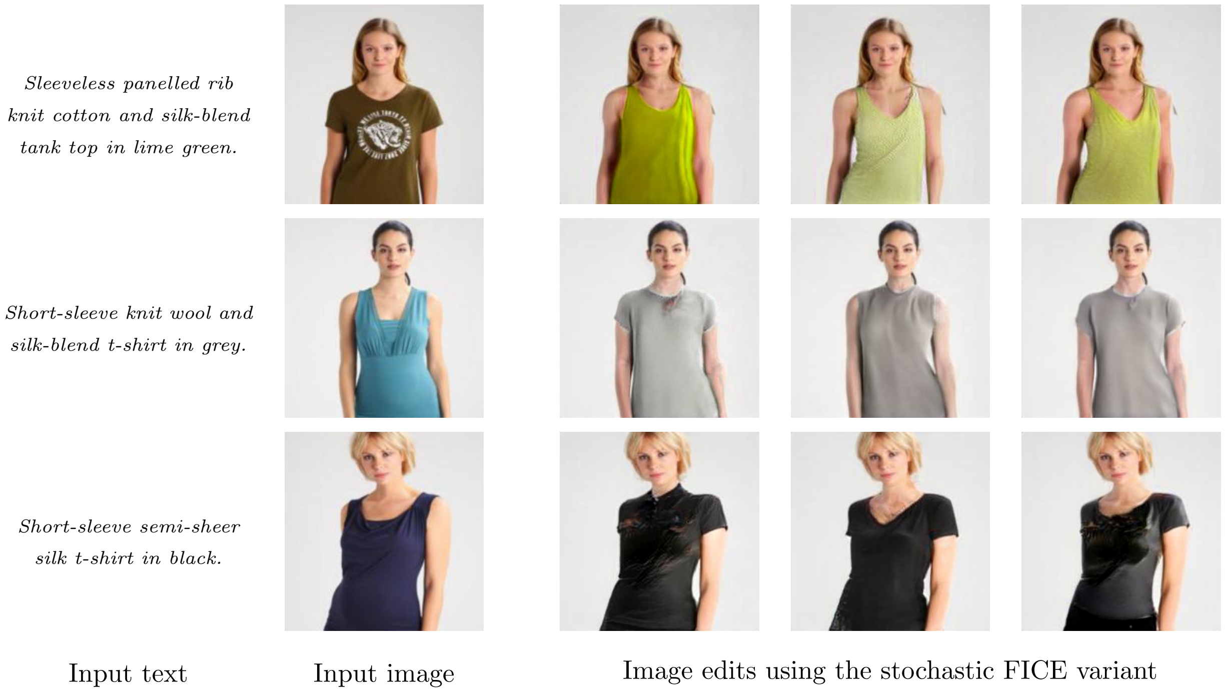

Sleeveless panelled rib knit cotton and silk-blend tank top in lime green.
Short-sleeve knit wool and silk-blend t-shirt in grey.
Short-sleeve semi-sheer silk t-shirt in black.
Input text
Input image
Image edits using the stochastic FICE variant

Figure 2: **Stochastic FICE.** By introducing stochastic mechanism in FICE, our method is capable of generating diverse image edits per single image–text pair.

initialization to mitigate this issue and synthesize images with better target semantics.

We note again that FICE operates in the extended latent vector space $\mathcal{W}^+$ of the pre-trained StyleGAN generator. The complete latent code $w \in \mathcal{W}^+$ of a given input image I, therefore, consists of several individual latent codes, each impacting an individual convolutional layer in the StyleGAN generator network G. To better understand the semantics, encoded in different subsets of the overall latent code, we conduct style-mixing experiments in this section. Style mixing refers to injecting a latent-code subset into another latent code. Similarly, as in [7], we do so for coarse, medium and fine subsets of the latent code $w = \{w_l\}_{l=1}^{L}$, (with $L = 14$ for our implementation of StyleGANv2), where the coarse subset corresponds to $l \in \{1, ..., 4\}$, medium to $l \in \{5, ..., 8\}$, and fine to $l \in \{9, ..., 14\}$ layers. A few example results of style-mixing experiments are presented in Fig. 3. We observe that copying part of the latent code that corresponds to the medium subset (layers 5 to 8) results in images with roughly the same pose as the original (destination) image, while inheriting the (approximate) clothing style of the source image.

The above observations motivated us to experiment with a different latent-code initialization procedure than used in the main part of the paper, where the coarse and fine subsets of the latent code are related to the input image, while the medium subsets exhibit visual semantics that correspond to the provided text description t. In order to obtain the latent code that best corresponds to the given text description, we use a sampling approach. Specifically, we generate N (complete) latent codes $w^{(i)}$ that serve as the

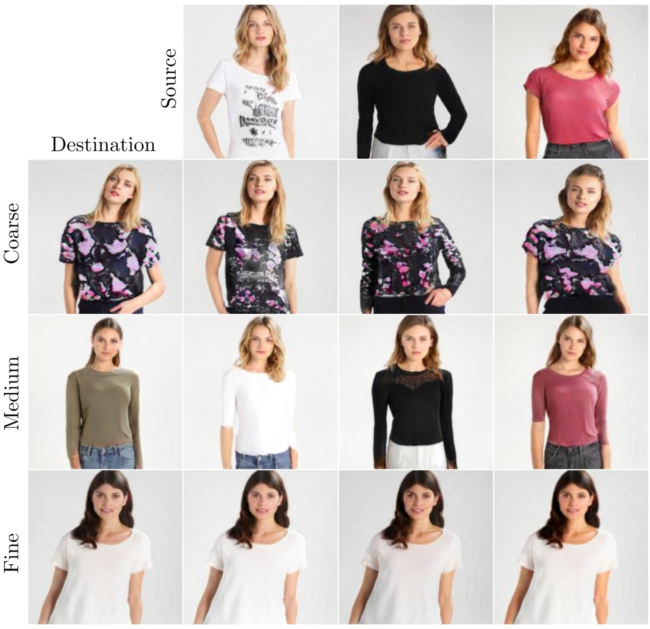


Figure 3: **Example results of style–mixing experiments.** We take part of the latent code from the Source image and use it to replace the corresponding part in the latent code of the Destination image. Only a certain subset of the original code of the Destination image is replaced, while the rest is preserved. We observe that copying the *coarse* subset (layers 1 to 4) causes the destination image to exhibit the pose of the source image. The *medium* subset (layers 5 to 8) appears most suitable for our task, as it tends to replicate the clothing style of the source image, while preserving the pose of the destination image. Finally, copying the *fine* subset (layers 9 to 14) mostly results in minor changes in the image tone without a major impact on the clothing or pose of the Destination image.

prototypes for our initialization procedure and are drawn randomly from different parts of the GAN latent space. Based on the sampled prototypes, we then generate the corresponding CLIP image embeddings $e_i^I = C^i(G(w^{(i)})) \in \mathcal{R}^{d_{clip} \times 1}$. Finally, we construct $N = 100,000$ $(w^{(i)}, e_i^I)$ pairs and store them for later processing.

When editing an image given the text description t, we process the text with the CLIP text encoder C^t to obtain the text embedding $e^T = C^t(t) \in \mathcal{R}^{d_{clip} \times 1}$ and compute all N similarities:

$$S_i(e_i^I, e^T) = \cos(e_i^I, e^T), \tag{2}$$

where $i \in \{1, \cdots, N\}$. The target prototype $w^{(i^*)}$, providing the medium latent-code subset is then selected based on the maximum similarity, i.e., $i^* = \arg\max_i \{S_i\}$. Finally, to obtain the coarse and fine latent-code subsets that best match the input image, we again leverage the E4e encoder to predict the extended latent code of the input image before and inject it with the medium latent-code subset of the selected prototype

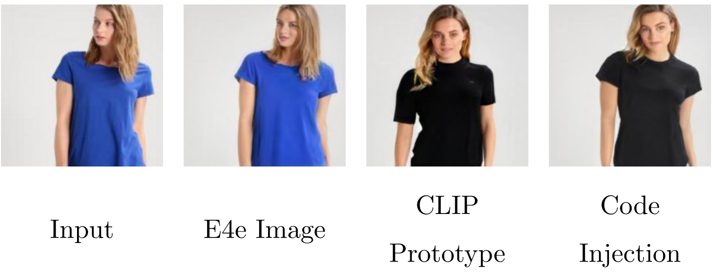


Figure 4: **Latent code initialization with style mixing (injection).** The examples show the initialization process for an input image and the following text description *"Short-sleeve antimicrobial merino wool-blend t-shirt in black"*. The input image is processed with the E4e model to obtain a latent code that corresponds to the input image (2nd column). Based on the text description, we identify a suitable CLIP prototype code (3rd column), which is injected into the computed E4e code, resulting in an image (4th column) with a similar pose to the input image and clothes resembling the identified CLIP prototype.

Table 2: **Quantitative results.** The style-mixing (code-injection) initialization procedure improves the semantic-relevance score, but degrades other performance indicators.

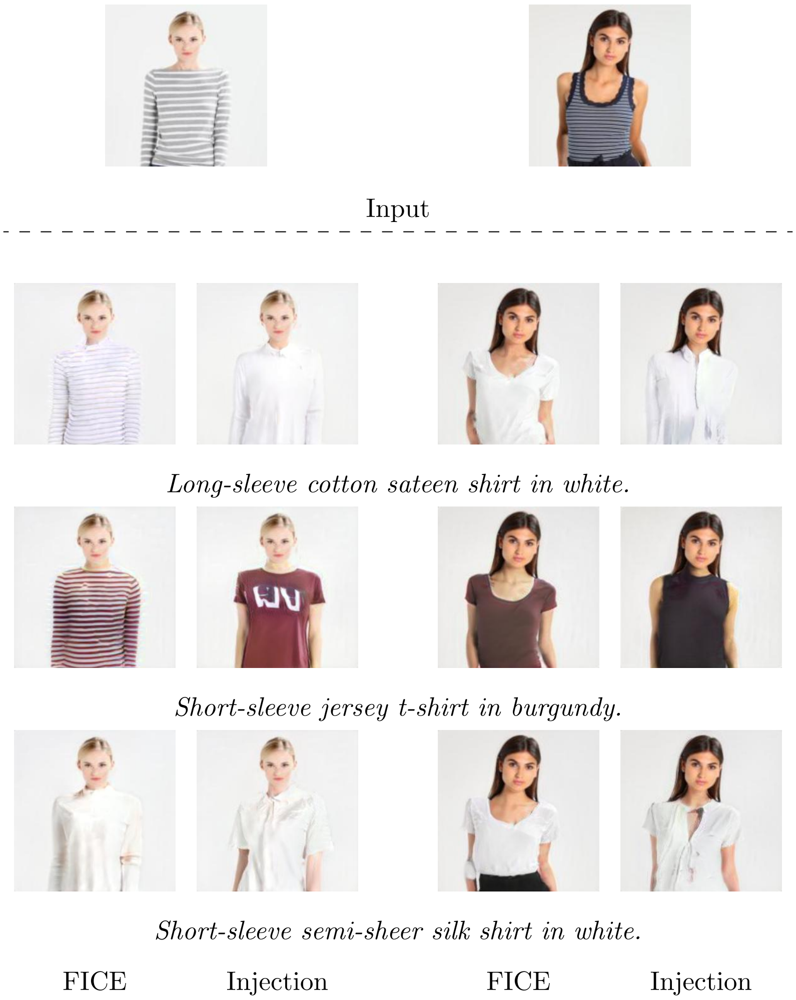
Initialization	**Semantics** (↑)	**Identity sim.** (↑)	**IoU** (↑)	**FID** (↓)
E4e init. (FICE)	0.446	**0.926**	**0.949**	**60.96**
Injection init.	**0.468**	0.912	0.931	84.03

$w^{(i^*)}$. The complete process is visualized in Fig. 4.

We evaluate the original initialization procedure and the style-mixing (with prototypes) initialization procedure quantitatively and qualitatively. In Table 2 we show the results across our performance indicators. We observe that the semantic-relevance score does increase, suggesting that the semantics, expressed in the text descriptions, are now better integrated into the edited images (on average). However, all the other performance indicators exhibit a slight degradation, most obviously, the FID score. Nevertheless, there are several positive aspects of such an initialization technique, as we show in Fig. 5. Note how the alternative initialization (marked *injection*) scheme allows us to convincingly infuse semantics that differ considerably from the original image. With the original initialization process this is not always the case.

5. Detailed Result Analysis

To visualize the VITON results across the target text prompts, we present box plots of each metric in Fig. 6. For the metrics calculated on individual images (CLIP, IoU, identity similarity), we averaged the results across the generated image set for each prompt. For the FID score, no averaging is performed, as the



Long-sleeve cotton sateen shirt in white.

Short-sleeve jersey t-shirt in burgundy.

Short-sleeve semi-sheer silk shirt in white.

Figure 5: **Comparison of latent-code initialization procedures.** The figure shows example results when initializing the latent codes needed by FICE wither with the (style-mixing-based) code injection and the vanilla E4e initialization used in the main paper. The first row shows the input images. The rows below show a comparison of the results when either initializing with E4e encoder (FICE) or when initializing with the code injection. We observe that the code-injection technique for images with certain characteristics produces better results than FICE. Specifically, the code-injection technique tends to facilitate better edits with respect to sleeve length and independence of the initial clothing characteristics – see the results corresponding to the stripe-pattern (left column).

metric is calculated based on comparisons between two image sets. Our analysis reveals the P2P model's vulnerability to pose distortion, indicated by frequent IoU outliers. P2P and other GAN-inversion methods also demonstrate occasional image-quality degradation on certain prompts, as evidenced by the FID results. In contrast, FICE consistently maintains pose preservation and identity similarity across diverse prompts.

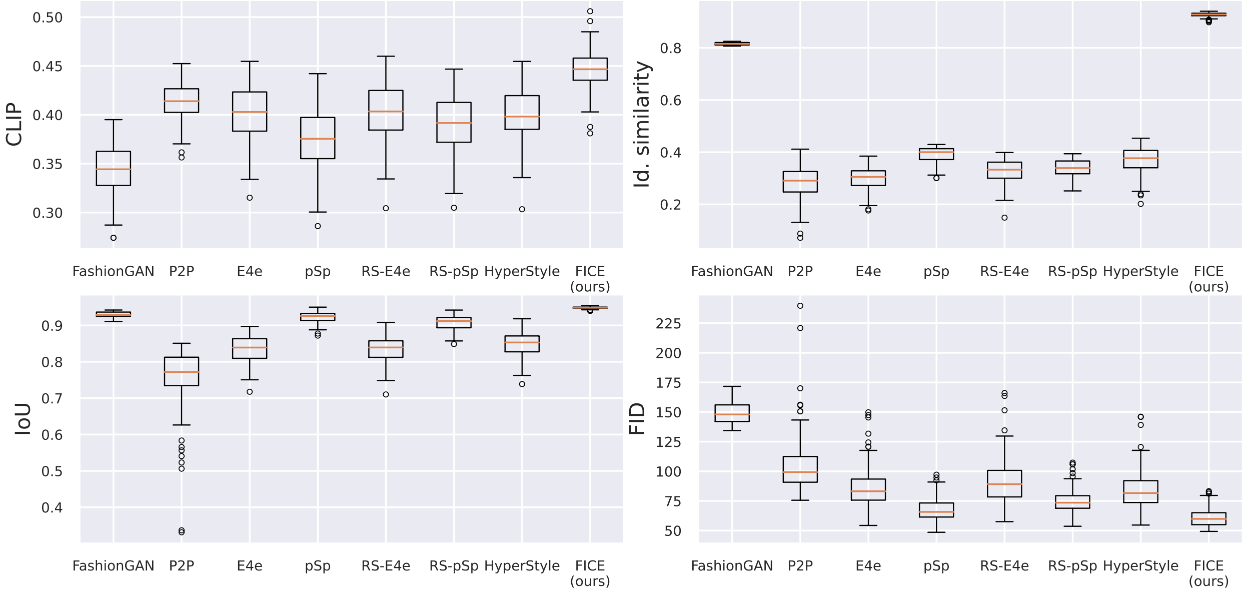


Figure 6: **Box plots of different performance indicators for the tested editing models.** The competing models were tested with hyperparameter settings that resulted in the highest semantic-relevance score to ensure a fair comparison. Results are reported in terms of variation over the text descriptions. The individual score for CLIP, identity similarity, and IoU is obtained by averaging the metric over all images for a given text description. We observe that FICE performs best across all performance indices, while ensuring the most consistent results.

6. Execution Time

To assess the computational complexity of the models, we measure the execution time required for each model to perform a single image edit based on an input image and target text prompt. For pSp, E4e, ReStyle-pSp, and ReStyle-e4e, this measurement includes the time needed to calculate the global latent direction using the StyleCLIP [8] method, considering both unconditional and target prompts. All the tests are conducted on an NVIDIA A100 GPU, with execution times averaged over five runs per model.

Table 3 summarizes the results. As expected, the encoder-decoder-based methods demonstrate shorter execution times than FICE. However, as discussed throughout the experimental section in the main manuscript, these methods fall short of delivering high-quality results.

It is interesting to note that the most time-consuming part of FICE is the latent-code optimization step, taking an average of 40.6 seconds. This step presents a clear opportunity for future research to enhance the FICE's execution time. Additionally, advancements in hardware capabilities could also further accelerate this process.

Model	FashionGAN	pSp	e4e	Restyle-pSp	Restyle-e4e	HyperStyle	P2P	FICE
Time [in s]	0.04	0.62	0.62	0.87	0.88	0.87	11.23	42.70

Table 3: **Model execution time.** Encoder-decoder methods demonstrate faster execution times. However, as discussed above, this speed comes at the cost of reduced image quality. The diffusion-based model (P2P) and FICE incur longer execution times due to their iterative processes. P2P requires numerous inversions as well as de-noising (editing) steps, while FICE relies on multiple latent-code optimization steps.

7. Implementation Details

All experiments presented in the paper were run on the Ubuntu 22.04 operating system. For experiments during the initial FICE development various different GPUs have been used, ranging from NVIDIA GeForce GTX 1080 Ti to NVIDIA RTX 3090 and NVIDIA A100. All FICE submodels as well as the baselines were implemented in the PyTorch framework. Additional implementation details are available from https://github.com/MartinPernus/FICE.

8. Implications of the Extended GAN Inversion Procedure

The extended GAN inversion procedure of FICE, introduced in the main manuscript, significantly advances the domain of text-conditioned fashion image editing. By allowing a detailed manipulation of fashion images, this procedure not only enhances the quality and precision of edited images, but also broadens the scope of potential applications in other fashion technology related methods [9–23], as well as broader computer vision methods [24–34].

Acknowledgements

Supported in parts by the Slovenian Research Agency ARRS through the Research Programme P2-0250(B) Metrology and Biometric System, the ARRS Project J2-2501(A) DeepBeauty and the ARRS junior researcher program.

References

[1] E. Richardson, Y. Alaluf, O. Patashnik, Y. Nitzan, Y. Azar, S. Shapiro, D. Cohen-Or, Encoding in style: a stylegan encoder for image-to-image translation, in: Computer Vision and Pattern Recognition (CVPR), 2021, pp. 2287–2296.

[2] O. Tov, Y. Alaluf, Y. Nitzan, O. Patashnik, D. Cohen-Or, Designing an encoder for stylegan image manipulation, ACM Transactions on Graphics (TOG) 40 (2021) 1–14.

[3] Y. Alaluf, O. Patashnik, D. Cohen-Or, Restyle: A residual-based stylegan encoder via iterative refinement, in: International Conference on Computer Vision (ICCV), 2021, pp. 6711–6720.

[4] Y. Alaluf, O. Tov, R. Mokady, R. Gal, A. H. Bermano, Hyperstyle: Stylegan inversion with hypernetworks for real image editing, in: Computer Vision and Pattern Recognition (CVPR), 2022.

[5] C. Schuhmann, R. Vencu, R. Beaumont, R. Kaczmarczyk, C. Mullis, A. Katta, T. Coombes, J. Jitsev, A. Komatsuzaki, Laion-400m: Open dataset of clip-filtered 400 million image-text pairs, arXiv preprint arXiv:2111.02114 (2021).

[6] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, T. Aila, Analyzing and improving the image quality of stylegan, in: Computer Vision and Pattern Recognition (CVPR), 2020, pp. 8110–8119.

[7] T. Karras, S. Laine, T. Aila, A style-based generator architecture for generative adversarial networks, in: Computer Vision and Pattern Recognition (CVPR), 2019, pp. 4401–4410.

[8] O. Patashnik, Z. Wu, E. Shechtman, D. Cohen-Or, D. Lischinski, Styleclip: Text-driven manipulation of stylegan imagery, in: International Conference on Computer Vision (ICCV), 2021, pp. 2085–2094.

[9] H. Zhang, S. Lin, R. Shao, Y. Zhang, Z. Zheng, H. Huang, Y. Guo, Y. Liu, Closet: Modeling clothed humans on continuous surface with explicit template decomposition, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 501–511.

[10] Y. Xiu, J. Yang, X. Cao, D. Tzionas, M. J. Black, Econ: Explicit clothed humans optimized via normal integration, in: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2023, pp. 512–523.

[11] L. Qiu, G. Chen, J. Zhou, M. Xu, J. Wang, X. Han, Rec-mv: Reconstructing 3d dynamic cloth from monocular videos, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 4637–4646.

[12] Y. Cao, K. Han, K.-Y. K. Wong, Sesdf: Self-evolved signed distance field for implicit 3d clothed human reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 4647–4657.

[13] D.-Y. Song, H. Lee, J. Seo, D. Cho, Difu: Depth-guided implicit function for clothed human reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 8738–8747.

[14] X. Zou, X. Han, W. Wong, Cloth4d: A dataset for clothed human reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 12847–12857.

[15] A. Grigorev, M. J. Black, O. Hilliges, Hood: Hierarchical graphs for generalized modelling of clothing dynamics, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 16965–16974.

[16] K. Wang, G. Zhang, S. Cong, J. Yang, Clothed human performance capture with a double-layer neural radiance fields, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 21098–21107.

[17] H. Bertiche, N. J. Mitra, K. Kulkarni, C.-H. P. Huang, T. Y. Wang, M. Madadi, S. Escalera, D. Ceylan, Blowing in the wind: Cyclenet for human cinemagraphs from still images, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 459–468.

[18] F. Zhao, Z. Li, S. Huang, J. Weng, T. Zhou, G.-S. Xie, J. Wang, Y. Shan, Learning anchor transformations for 3d garment animation, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 491–500.

[19] L. De Luigi, R. Li, B. Guillard, M. Salzmann, P. Fua, Drapenet: Garment generation and self-supervised draping, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 1451–1460.

[20] Y. Jafarian, T. Y. Wang, D. Ceylan, J. Yang, N. Carr, Y. Zhou, H. S. Park, Normal-guided garment uv prediction for human re-texturing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 4627–4636.

[21] Y. Jiao, Y. Gao, J. Meng, J. Shang, Y. Sun, Learning attribute and class-specific representation duet for fine-grained fashion analysis, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 11050–11059.

[22] Y. Han, L. Zhang, Q. Chen, Z. Chen, Z. Li, J. Yang, Z. Cao, Fashionsap: Symbols and attributes prompt for fine-grained fashion vision-language pre-training, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern

Recognition, 2023, pp. 15028–15038.

[23] R. Jain, K. K. Singh, M. Hemani, J. Lu, M. Sarkar, D. Ceylan, B. Krishnamurthy, Vgflow: Visibility guided flow network for human reposing, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2023, pp. 21088–21097.

[24] J. Zhang, C. Li, S. Kosov, M. Grzegorzek, K. Shirahama, T. Jiang, C. Sun, Z. Li, H. Li, Lcu-net: A novel low-cost u-net for environmental microorganism image segmentation, Pattern Recognition 115 (2021) 107885.

[25] J. Zhang, C. Li, Y. Yin, J. Zhang, M. Grzegorzek, Applications of artificial neural networks in microorganism image analysis: a comprehensive review from conventional multilayer perceptron to popular convolutional neural network and potential visual transformer, Artificial Intelligence Review 56 (2023) 1013–1070.

[26] H. Chen, C. Li, X. Li, M. M. Rahaman, W. Hu, Y. Li, W. Liu, C. Sun, H. Sun, X. Huang, et al., Il-mcam: An interactive learning and multi-channel attention mechanism-based weakly supervised colorectal histopathology image classification approach, Computers in Biology and Medicine 143 (2022) 105265.

[27] X. Li, C. Li, M. M. Rahaman, H. Sun, X. Li, J. Wu, Y. Yao, M. Grzegorzek, A comprehensive review of computer-aided whole-slide image analysis: from datasets to feature extraction, segmentation, classification and detection approaches, Artificial Intelligence Review 55 (2022) 4809–4878.

[28] H. Chen, C. Li, G. Wang, X. Li, M. M. Rahaman, H. Sun, W. Hu, Y. Li, W. Liu, C. Sun, et al., Gashis-transformer: A multi-scale visual transformer approach for gastric histopathological image detection, Pattern Recognition 130 (2022) 108827.

[29] F. Kulwa, C. Li, J. Zhang, K. Shirahama, S. Kosov, X. Zhao, T. Jiang, M. Grzegorzek, A new pairwise deep learning feature for environmental microorganism image analysis, Environmental Science and Pollution Research 29 (2022) 51909–51926.

[30] W. Liu, C. Li, N. Xu, T. Jiang, M. M. Rahaman, H. Sun, X. Wu, W. Hu, H. Chen, C. Sun, et al., Cvm-cervix: A hybrid cervical pap-smear image classification framework using cnn, visual transformer and multilayer perceptron, Pattern Recognition 130 (2022) 108829.

[31] M. M. Rahaman, C. Li, Y. Yao, F. Kulwa, X. Wu, X. Li, Q. Wang, Deepcervix: A deep learning-based framework for the classification of cervical cells using hybrid deep feature fusion techniques, Computers in Biology and Medicine 136 (2021) 104649.

[32] Z. Fan, X. Wu, C. Li, H. Chen, W. Liu, Y. Zheng, J. Chen, X. Li, H. Sun, T. Jiang, et al., Cam-vt: A weakly supervised cervical cancer nest image identification approach using conjugated attention mechanism and visual transformer, Computers in Biology and Medicine 162 (2023) 107070.

[33] A. Chen, C. Li, S. Zou, M. M. Rahaman, Y. Yao, H. Chen, H. Yang, P. Zhao, W. Hu, W. Liu, et al., Svia dataset: A new dataset of microscopic videos and images for computer-aided sperm analysis, Biocybernetics and Biomedical Engineering 42 (2022) 204–214.

[34] Q. Nie, C. Li, J. Yang, Y. Yao, H. Sun, T. Jiang, M. Grzegorzek, A. Chen, H. Chen, W. Hu, et al., Oii-ds: A benchmark oral implant image dataset for object detection and image classification evaluation, Computers in Biology and Medicine 167 (2023) 107620.