Discovering and Mitigating Biases in CLIP-based Image Editing - Supplementary Material

Md Mehrab Tanjim, Krishna Kumar Singh, Kushal Kafle, Ritwik Sinha
Adobe Research
{tanjim,krishsin,kkafle,risinha}@adobe.com

Garrison W. Cottrell
UC San Diego
gary@ucsd.edu

1. Discovering Bias in the CLIP

1.1. Dataset Statistics

As mentioned in the main paper, we use Adobe Stock API to collect high-quality images for different professions. For each profession, we add Male/Female/White/Black or African American in front. In this way, in total, we collect 1067 images. We give the full statistics of the collected images in Table 1.

Profession	Female	Male	White	Black or African American
Administrative Assistant	17	10	21	16
Carpenter	12	26	14	12
Cleaner	16	22	17	24
Executive Manager	23	34	40	40
Farmer	21	34	32	32
Machine Operator	18	13	15	21
Military Person	18	15	13	17
Nurse	28	38	42	44
Plumber	17	20	25	15
Security Guard	4	9	5	4
Software Developers	13	17	8	16
Technical Support Person	13	20	16	4
Truck Driver	24	30	3	7
Writers	10	18	12	12

Table 1. Data statistics for different professions.

1.2. CLIP Model

We follow prompt engineering from [8] to augment the query text by using a template of 79 semantically similar meaningful sentences (e.g. 'a photo of the clean {}', 'a photo of a large {}', etc.). For discovering biases, we present results from ResNet-50 [4] pre-trained CLIP image encoder [10] in the main paper. Results from transformer-based image encoders are similar. For example, the ROC curve and percentage misranks for 'ViT-B/32' CLIP image encoder are shown in Figure 1 and 2.

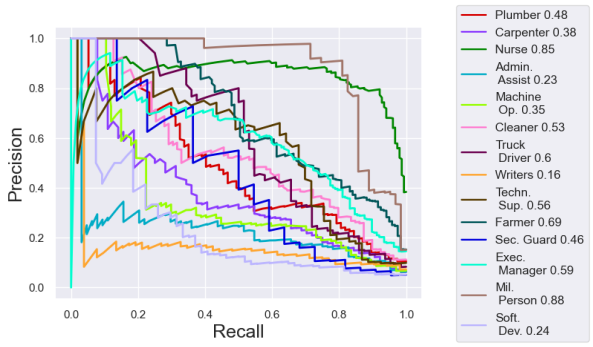
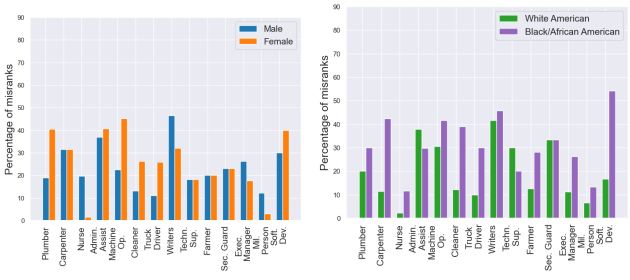


Figure 1. ROC Curve based on the CLIP scores from 'ViT-B/32' image encoder.



(a) Gender Subset (b) Race Subset

Figure 2. Percentage of misranks from 'ViT-B/32' CLIP image encoder.

1.3. Additional GradCAM Visualizations

In Figure 3, we show additional examples of GradCAM visualization. We can see, when CLIP model misranks, it is often due to facial features of the opposite gender or race, which is a clear indication of bias.

2. Debiasing Framework

Before applying our debiasing framework, we set $\alpha = 5$, which controls the magnitude of edit, to get outputs from StyleCLIP [9]. Similarly, we set the default value of $s_I = 1.5$ (image conditioning) and $s_T = 7.5$ (text conditioning)

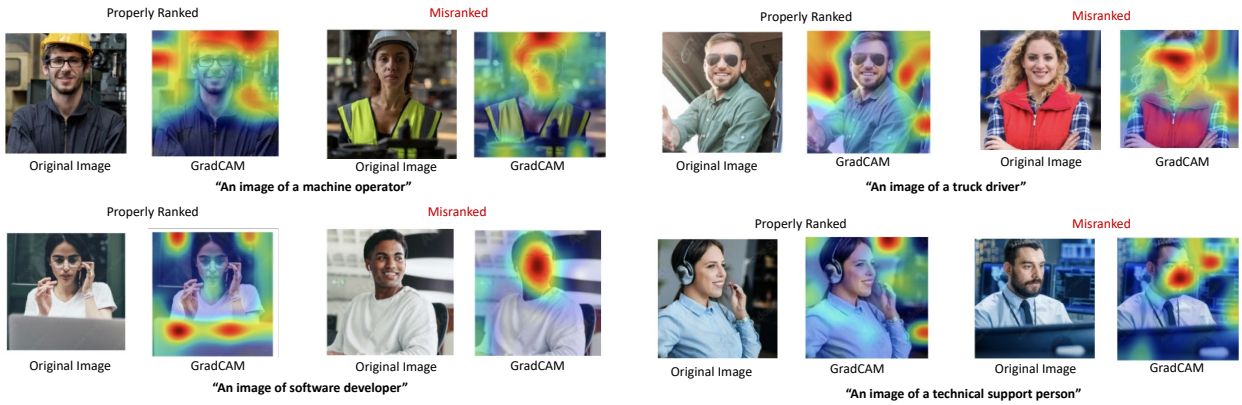
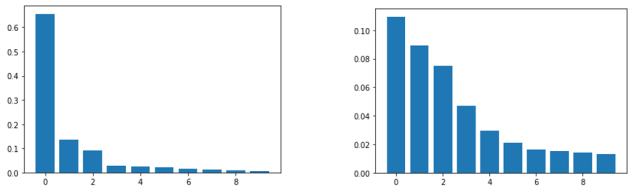


Figure 3. Additional GradCAM visualization on different professions.



(a) Gender (b) Race

Figure 4. Percentage of variance explained.

in InstructPix2Pix [2] to get outputs from it.

2.1. Text-based Debiasing

We collect gender pairs from [1]. These pairs are as follows: ('woman', 'man'), ('girl', 'boy'), ('she', 'he'), ('mother', 'father'), ('daughter', 'son'), ('gal', 'guy'), ('female', 'male'), ('her', 'his'), ('herself', 'himself'), ('Mary', 'John'). For racial words, we first use the triplets from [7]: ("black", "caucasian", "asian"), ("african", "caucasian", "asian"), ("black", "white", "asian"), ("africa", "america", "asia"), ("africa", "america", "china"), ("africa", "europe", "asia"). In addition, we utilize the most commonly occurring names from [3] for each ethnicity. Examples of these names and their respective ethnicities are as follows: "Asian Americans": Jaegwon Kim, Amartya Sen; "African Americans": Mahershala Ali, Ajiona Alexus; "European Americans": Jerome Connor, William Harnett; "Hispanic and Latino Americans": Jennifer Lopez, Jorge Garcia. To identify racial subspaces, we randomly generate 250 pairs of names by selecting two distinct ethnicities from the aforementioned list of names. The complete list of names can be found in [11].

To conduct Principal Component Analysis (PCA) on text embeddings, we start by computing the mean of word pairs or triplets. For instance, if $\overrightarrow{she}$ represents the text embedding for the word "she", and $\overrightarrow{he}$ represents the text embedding for the word "he", then we compute the mean as $\overrightarrow{c_{she-he}} = (\overrightarrow{she} + \overrightarrow{he})/2$. Next, we center each pair of text embeddings using their respective mean, i.e., $\overrightarrow{she_c} = \overrightarrow{she} - \overrightarrow{c_{she-he}}$ and $\overrightarrow{he_c} = \overrightarrow{he} - \overrightarrow{c_{she-he}}$. We then stack all these mean-centered embeddings and perform PCA. Similarly, for race triplet words, we calculate the mean as $(\overrightarrow{\text{"}black\text{"}} + \overrightarrow{\text{"}caucasian\text{"}} + \overrightarrow{\text{"}asian\text{"}})/3$ and center each triplet of word embeddings. Pairs of embeddings of different ethnic names are mean centered similarly.

For gender and race words, we conduct PCA separately. To obtain a combined subspace for debiasing, we concatenate all the principal components from gender and race. Specifically, if G represents the principal components for the Gender subspace, and R represents the principal components for the race subspace, then we can jointly perform debiasing for both Gender and Race by using the concatenation $\begin{bmatrix} G \\ R \end{bmatrix}$.

Figure 4 shows how much of the variance is explained when we perform PCA on these gender and racial words. For example, the first principal component of the gender words explains more than 60% of the total variance (Figure 4a), which suggests there is a direction that is highly correlated to gender words. Unfortunately, the same cannot be said for racial words (Figure 4b), which explains why text-based debiasing does not work well for removing racial biases.

2.2. Gradient-based Debiasing

To calculate the loss, we utilize the Adam optimizer [6] with a learning rate of 0.2 for FFHQ and SOHQ pre-trained StyleCLIP and 0.001 for InstructPix2Pix, and all the hyperparameters in Equation 5 in the main paper are set to 1, except for β_1 for CLIP, which is set to 5.0. We perform 100

Table 2. Impact of different combinations of identity preserving losses in our gradient-based approach.

Model	Attribute	Gradient-based Debiasing		
		CLIP+ID	CLIP+LPIPS	CLIP+ID+LPIPS
StyleCLIP (FFHQ)	Gender ↓	0.2329	0.1661	**0.1183**
	Race ↓	0.0882	0.0837	**0.0500**
	Age ↓	0.1018	0.0915	**0.0630**
	Profession ↑	**0.1261**	0.1212	0.1090
StyleCLIP (SOHQ)	Gender ↓	0.1434	0.0736	**0.0712**
	Race ↓	0.1177	0.0874	**0.0744**
	Age ↓	0.0935	0.0585	**0.0518**
	Profession ↑	**0.2577**	0.2215	0.2443
InstructPix2Pix	Gender ↓	0.0916	0.4332	**0.0644**
	Race ↓	**0.1152**	0.2238	0.1467
	Age ↓	**0.0732**	0.1308	0.0785
	Profession ↑	0.2156	**0.3060**	0.2192

steps for the optimization for StyleCLIP and 50 steps for InstructPix2Pix.

To extract faces during optimization to apply identity-preserving losses, we first employ dlib [5] facial recognition, which provides us with landmark and bounding box coordinates for faces. We use these coordinates to mask out everything except faces when computing the ID and LPIPS loss. This restricts the model to focus only on faces for preserving identity.

To evaluate the impact of each identity-preserving loss, we conduct an ablation study by comparing the performance of CLIP with each additional loss separately and all losses combined. We randomly generate 10 images for each of the 14 professions and apply the following combinations of losses to each image by setting the respective β values to zero: CLIP+ID, CLIP+LPIPS, and CLIP+LPIPS+ID. The results of the ablation study are shown in Table 2.

As observed, the combination of all three losses achieves the lowest scores for all protected attributes while only slightly compromising profession prediction scores. As discussed in the main paper, preserving the identity of the person is crucial, and a small change in the protected attribute score can significantly alter a person's identity. In contrast, changes in the profession prediction score still depict the profession sufficiently, even if there is a slight decrease in the score. The qualitative results from the ablation study are illustrated in Figure 8 in the main paper. Finally, when applying the CLIP loss (Equation 4 in the main paper), we have the option to use either the original text embedding from CLIP or apply our text-based debiasing steps to obtain a debiased text embedding. For this, we do similar experiments as before and randomly generate 10 images for each of the 14 professions for each pre-trained model. Our experiments have shown that using the original text embedding yields better results in most cases (see Table 3). In particular, using the original text embedding provides better performance for profession prediction scores while still minimizing gender/race/age differences in most cases.

Table 3. Comparison between using original vs. debiased text embedding in our gradient-based approach.

Model	Attribute	Gradient-based Debiasing	
		Debiased Text Emb.	Original Text Emb.
StyleCLIP (FFHQ)	Gender ↓	0.1382	**0.1254**
	Race ↓	0.0519	**0.0424**
	Age ↓	0.0724	**0.0640**
	Profession ↑	0.1079	**0.1221**
StyleCLIP (SOHQ)	Gender ↓	0.0772	**0.0657**
	Race ↓	**0.0657**	0.0671
	Age ↓	**0.0519**	0.0520
	Profession ↑	0.1583	**0.1982**
InstructPix2Pix	Gender ↓	**0.0073**	0.0644
	Race ↓	**0.0994**	0.1467
	Age ↓	**0.0507**	0.0785
	Profession ↑	0.1176	**0.2192**

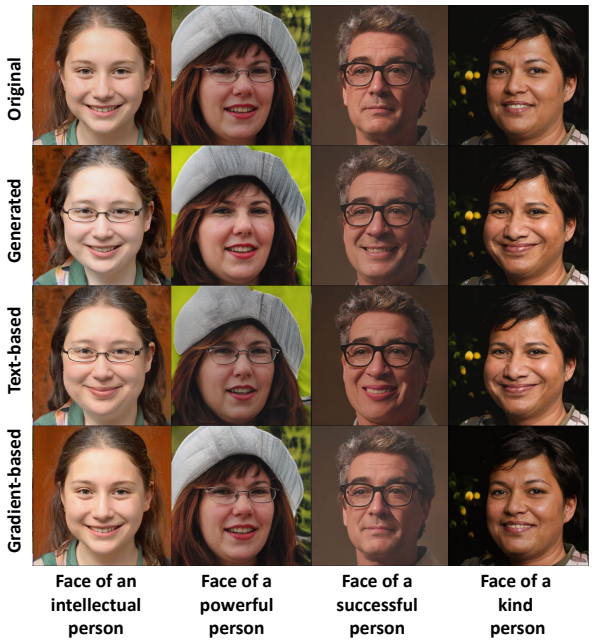


Figure 5. Different queries can impact different facial attributes, such as smiling or wearing eyeglasses. Here, the examples are shown from StyleCLIP.

2.3. Other Types of Biases

Our paper primarily focuses on exposing gender and racial biases in various occupation-related search queries. However, biases of different kinds can manifest in other types of queries as well. As an example, biases can also be indicated by certain attributes such as a smile being associated with kindness or success, eyeglasses being linked to intelligence, or the absence of eyeglasses indicating power, as shown in Figure 5. In addition, the figure demonstrates that text-based debiasing approaches are insufficient in these scenarios. This is because the text-based approach relies on the gender and race subspace, making it incapable of addressing biases beyond those categories. In contrast, our gradient-based method performs well even in these cases, as it does not rely on specific keywords or queries.

2.4. Additional Examples

We provide additional examples of the comparison among the outputs of InstructPix2Pix, StyleCLIP and our debiasing framework for each of the 14 professions in Figure 6, Figure 7 and 8 respectively.

References

[1] Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *Advances in neural information processing systems*, 29, 2016. 2

[2] Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions. In *CVPR*, 2023. 2

[3] Jwala Dhamala, Tony Sun, Varun Kumar, Satyapriya Krishna, Yada Pruksachatkun, Kai-Wei Chang, and Rahul Gupta. Bold: Dataset and metrics for measuring biases in open-ended language generation. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, page 862–872, New York, NY, USA, 2021. Association for Computing Machinery. 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[5] Davis E. King. Dlib-ml: A machine learning toolkit. *Journal of Machine Learning Research*, 10:1755–1758, 2009. 3

[6] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2015. 2

[7] Thomas Manzini, Yao Chong Lim, Yulia Tsvetkov, and Alan W Black. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. *arXiv preprint arXiv:1904.04047*, 2019. 2

[8] OpenAI. CLIP: Connecting Text and Images. https://github.com/openai/CLIP/blob/main/notebooks/Prompt_Engineering_for_ImageNet.ipynb, 2021. 1

[9] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 1

[10] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1

[11] Amazon Science. BOLD - a resource for building conversational ai. https://github.com/amazon-science/bold/blob/main/prompts/race_prompt.json, 2021. 2

Figure 6. Additional examples for InstructPix2Pix for each of the 14 professions.

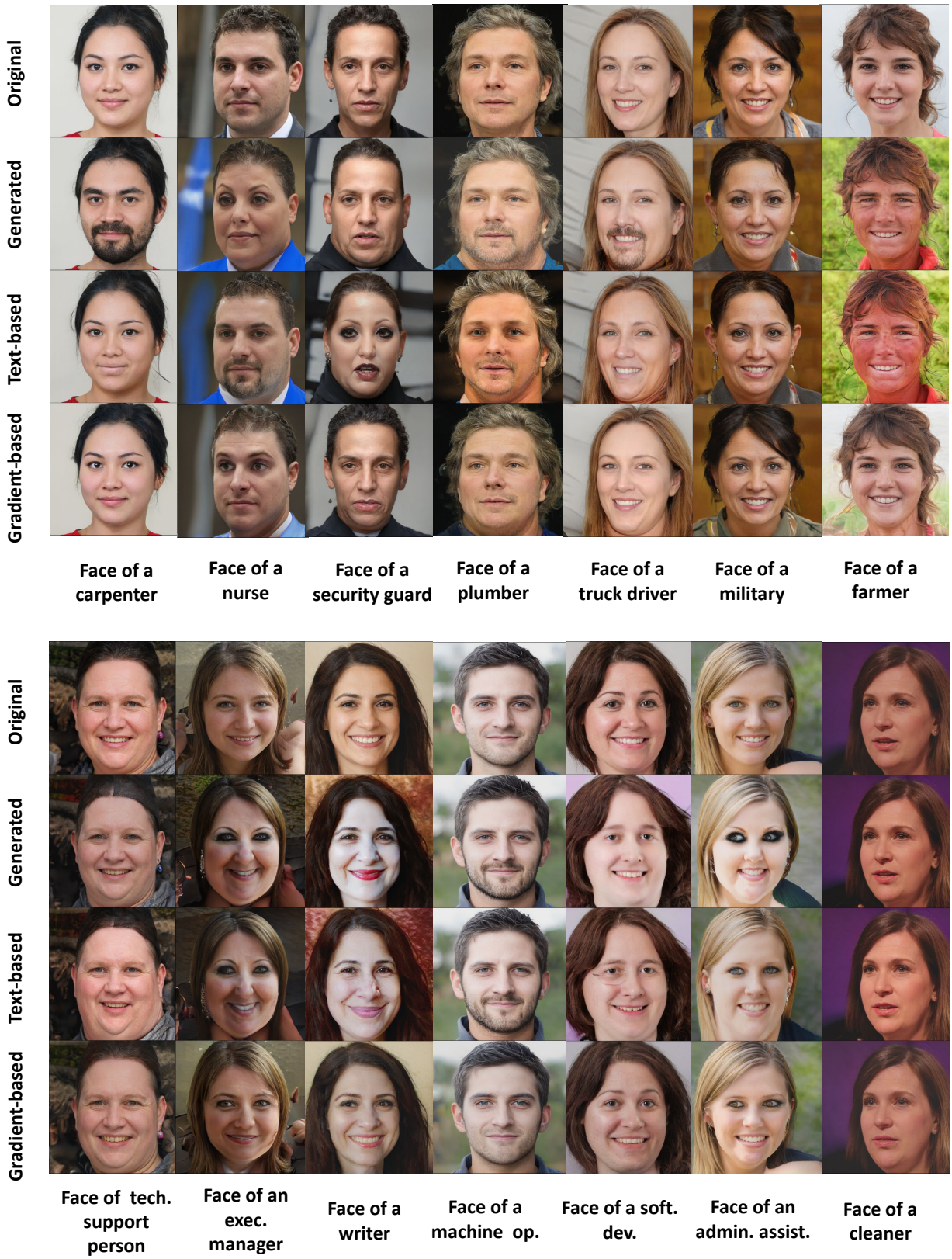


Figure 7. Additional examples for StyleCLIP (FFHQ) for each of the 14 professions.

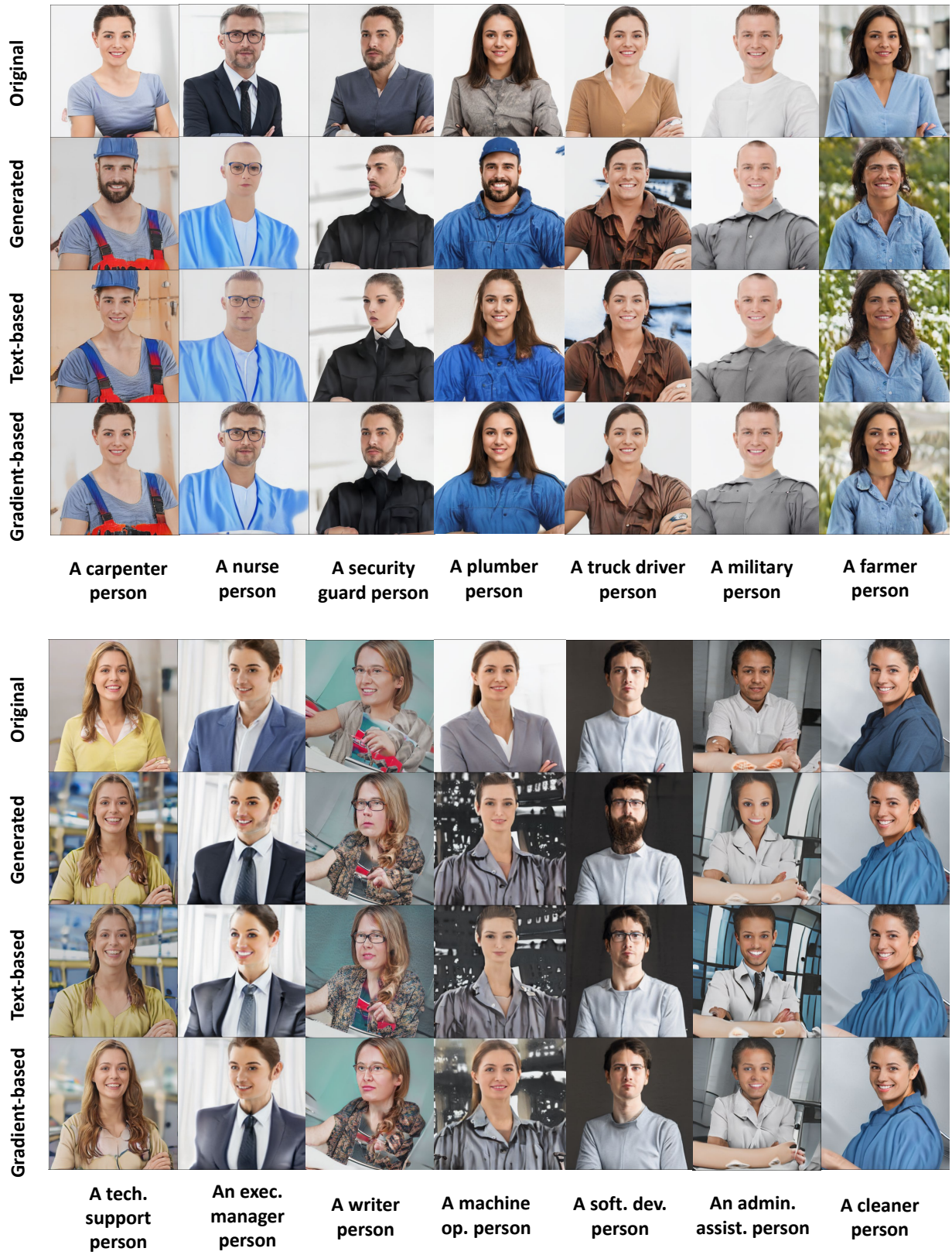


Figure 8. Additional examples for StyleCLIP (SOHQ) model for each of the 14 professions.