Appendix of Full-Resolution Correspondence Learning for Image Translation

A. Additional Generation Results

1. Pose-to-body

Figure 1 to Figure 5 show more results about pose-to-body generation at the resolution 512×512 on the Deepfashion dataset. To the best of our knowledge, our approach is the first work to generate person images at the resolution 512×512 on the Deepfashion dataset. Our approach is able to well preserve the patterns, *i.e.*, logos and letters, on the clothing because of the *full-resolution* correspondences constructed between two images. The person images generated by our approach are highly authentic and vivid.

Figure 1: Pose-to-body image translation results at resolution 512×512. 1st row: exemplar images, 2nd row: generated images. (Deepfashion dataset)

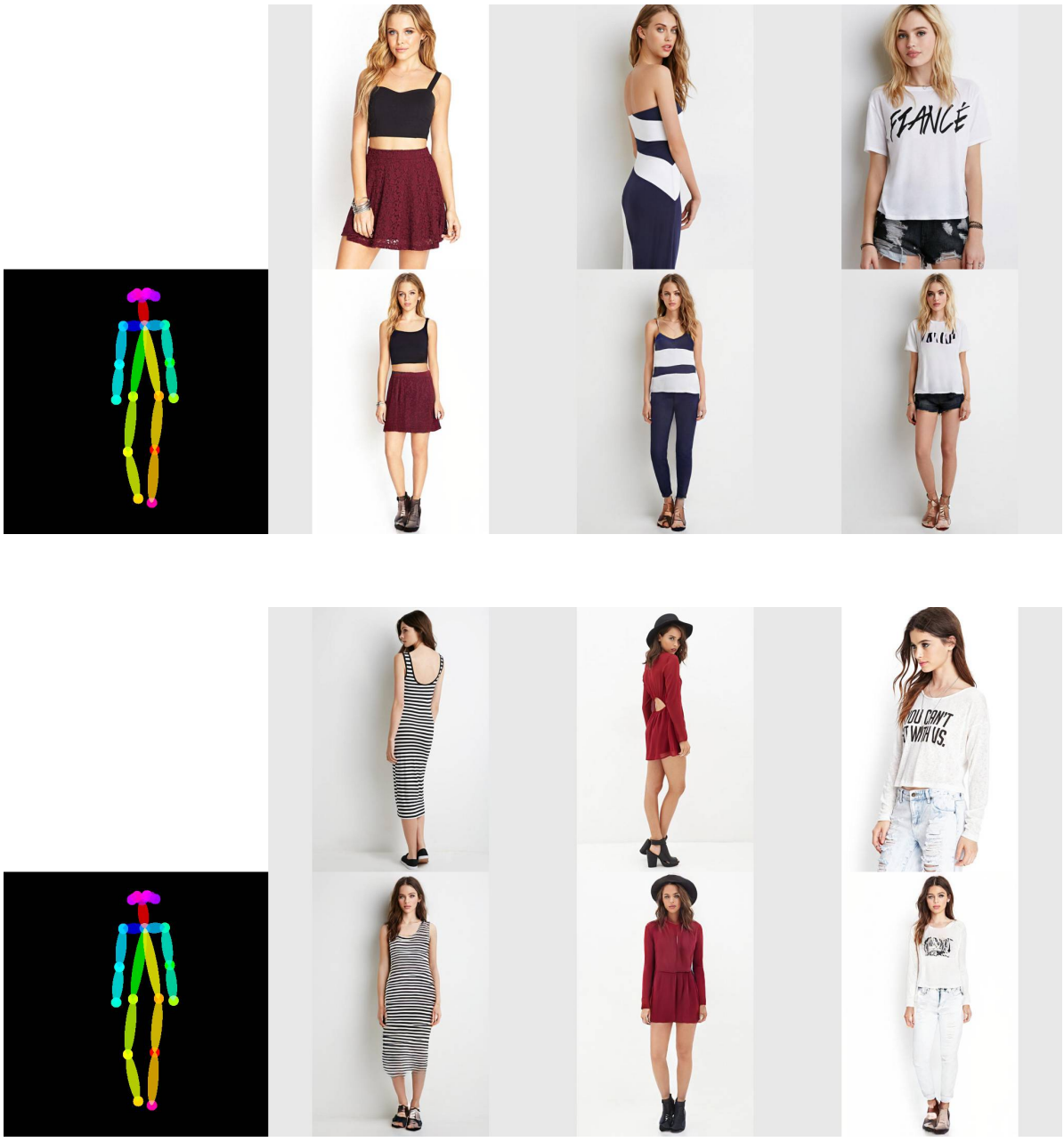

Figure 2: Pose-to-body image translation results at resolution 512×512. 1st row: exemplar images, 2nd row: generated images. (Deepfashion dataset)

Figure 3: Pose-to-body image translation results at resolution 512×512. 1st row: exemplar images, 2nd row: generated images. (Deepfashion dataset)

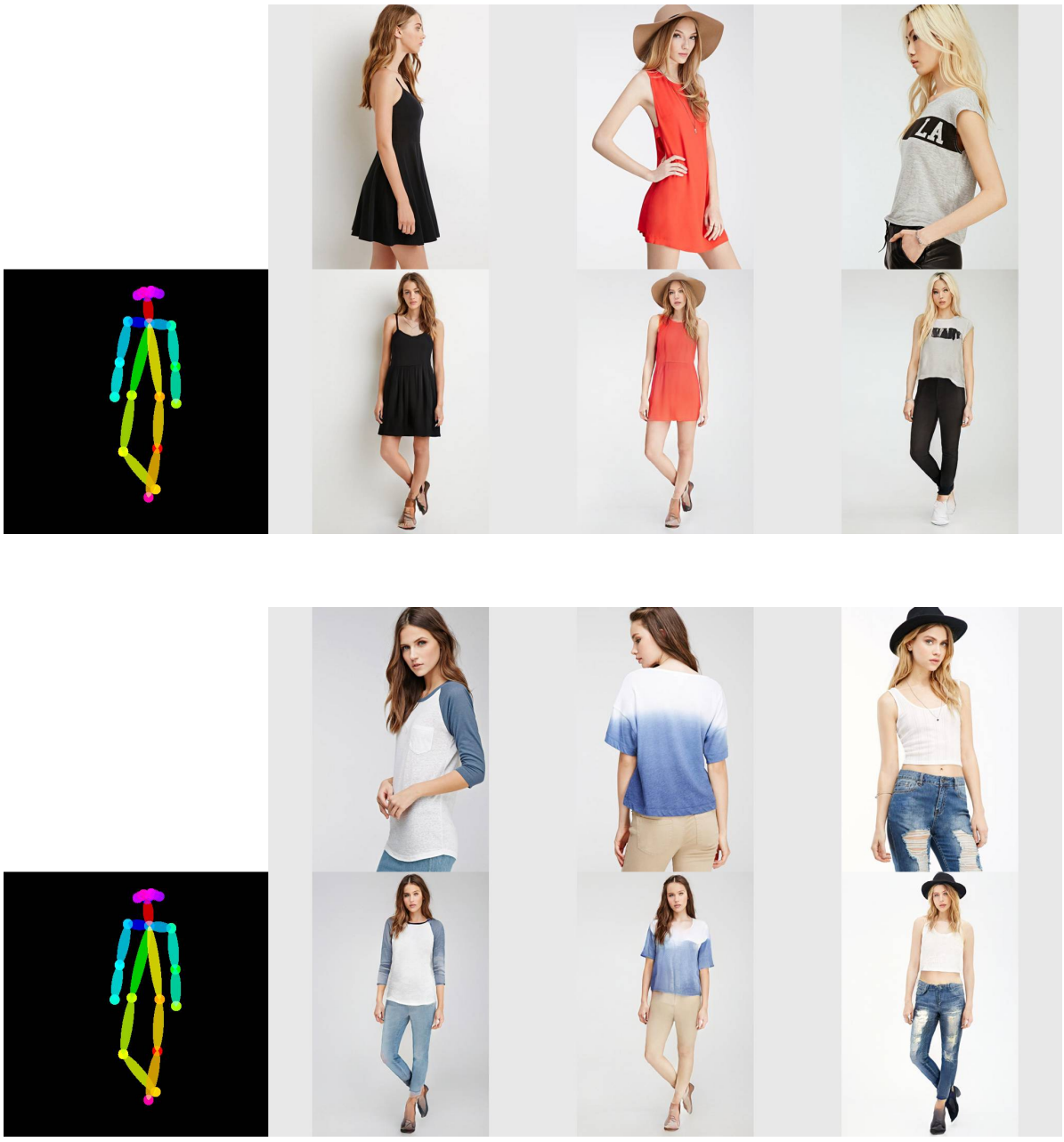

Figure 4: Pose-to-body image translation results at resolution 512×512. 1st row: exemplar images, 2nd row: generated images. (Deepfashion dataset)

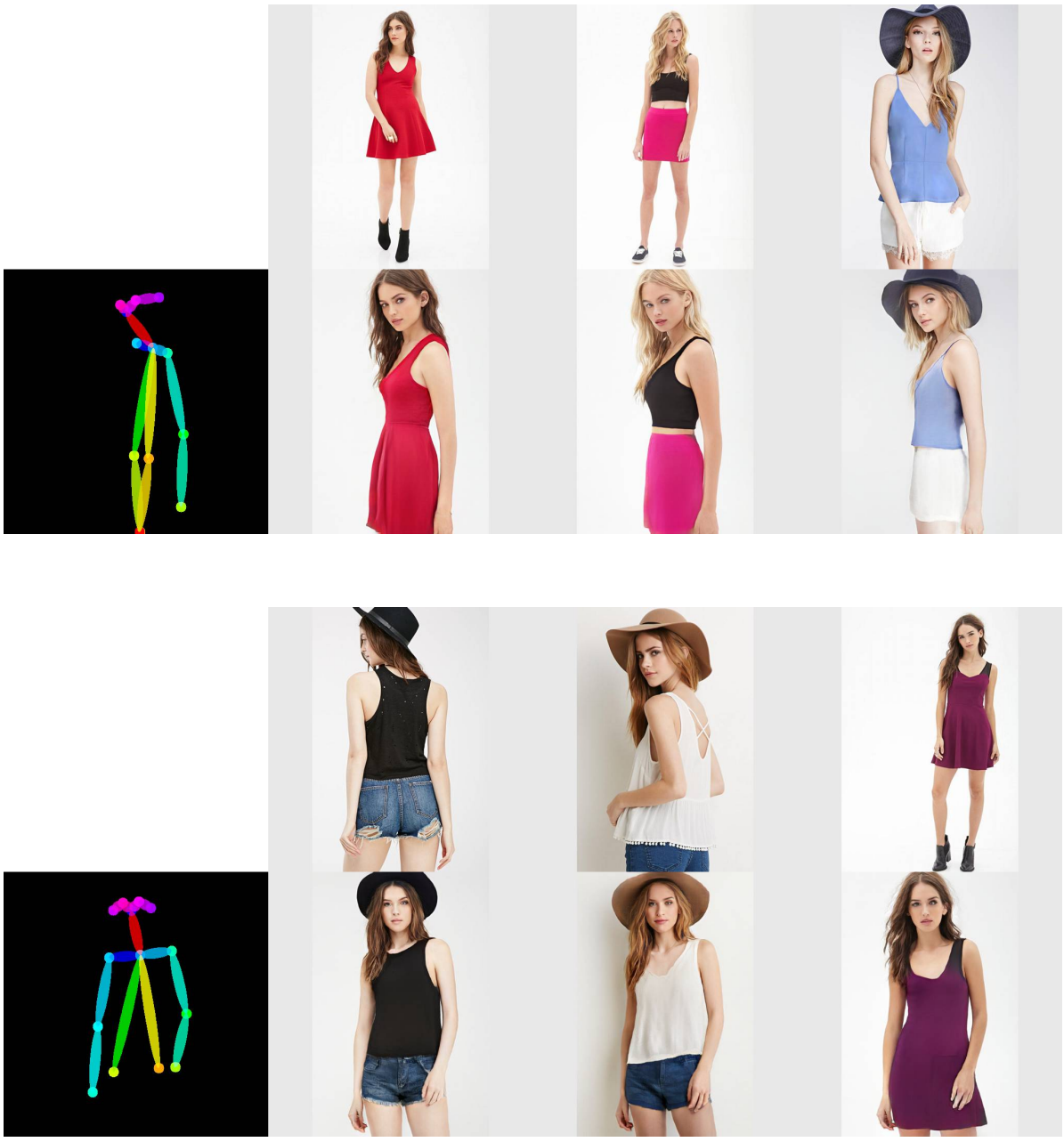

Figure 5: Pose-to-body image translation results at resolution 512×512. 1st row: exemplar images, 2nd row: generated images. (Deepfashion dataset)

2. Edge-to-face

Figure 6 to Figure 9 show more results of edge-to-face generation at the resolution 1024×1024 on the MetFaces dataset. Our approach produces visually appealing edge-to-face translation results at high-resolution.

Figure 6: Edge-to-face image translation results at resolution 1024×1024. 1st row: exemplar images, 2nd row: generated images. (MetFaces dataset)

Figure 7: Edge-to-face image translation results at resolution 1024×1024. 1st row: exemplar images, 2nd row: generated images. (MetFaces dataset)

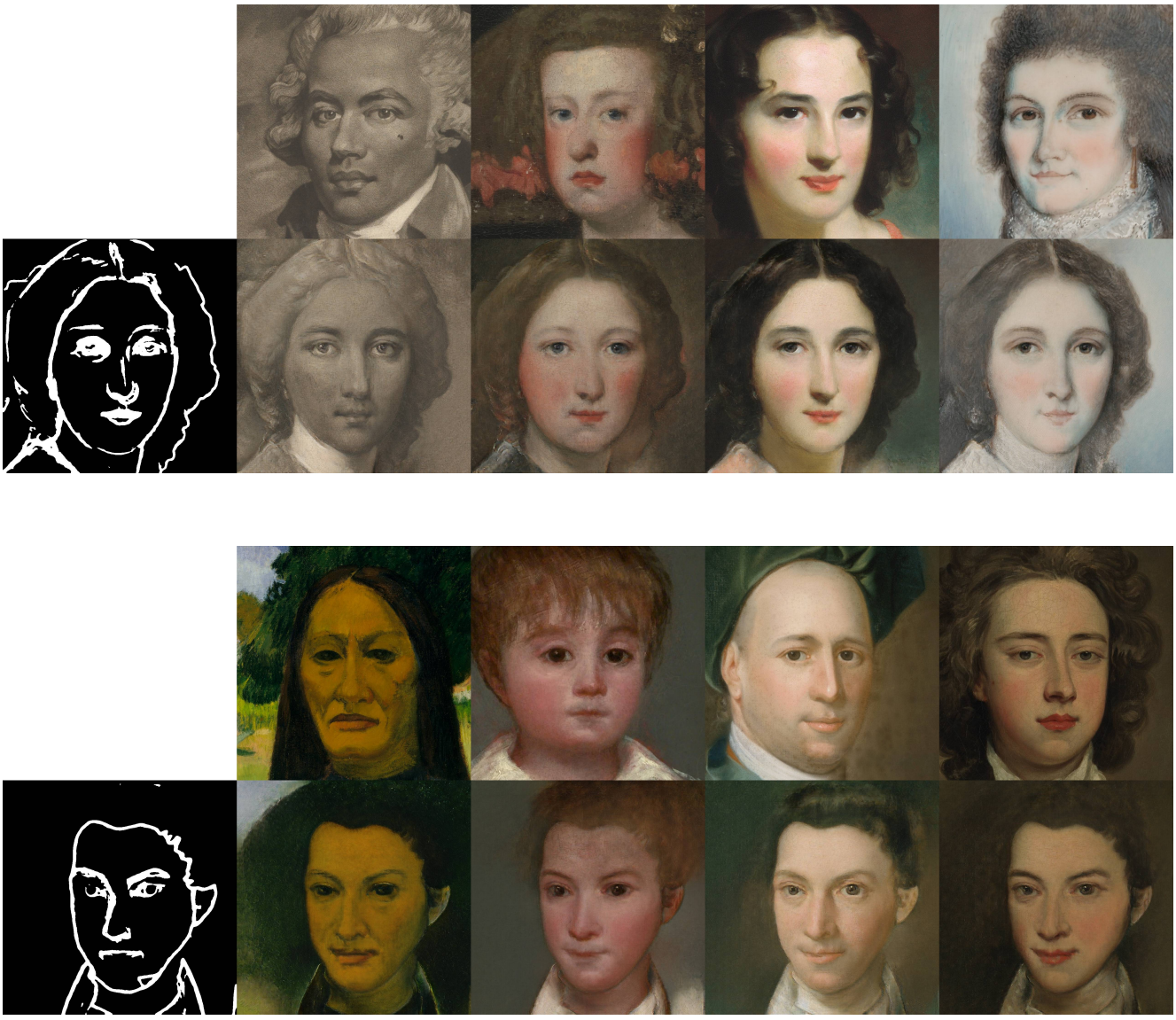

Figure 8: Edge-to-face image translation results at resolution 1024×1024. 1st row: exemplar images, 2nd row: generated images. (MetFaces dataset)

Figure 9: Edge-to-face image translation results at resolution 1024×1024. 1st row: exemplar images, 2nd row: generated images. (MetFaces dataset)

3. Mask-to-image

Figure 10 shows more results of mask-to-image generation on the ADE20K dataset. The proposed method is able to achieve state-of-the-art quality for diverse scenes on this challenging dataset.

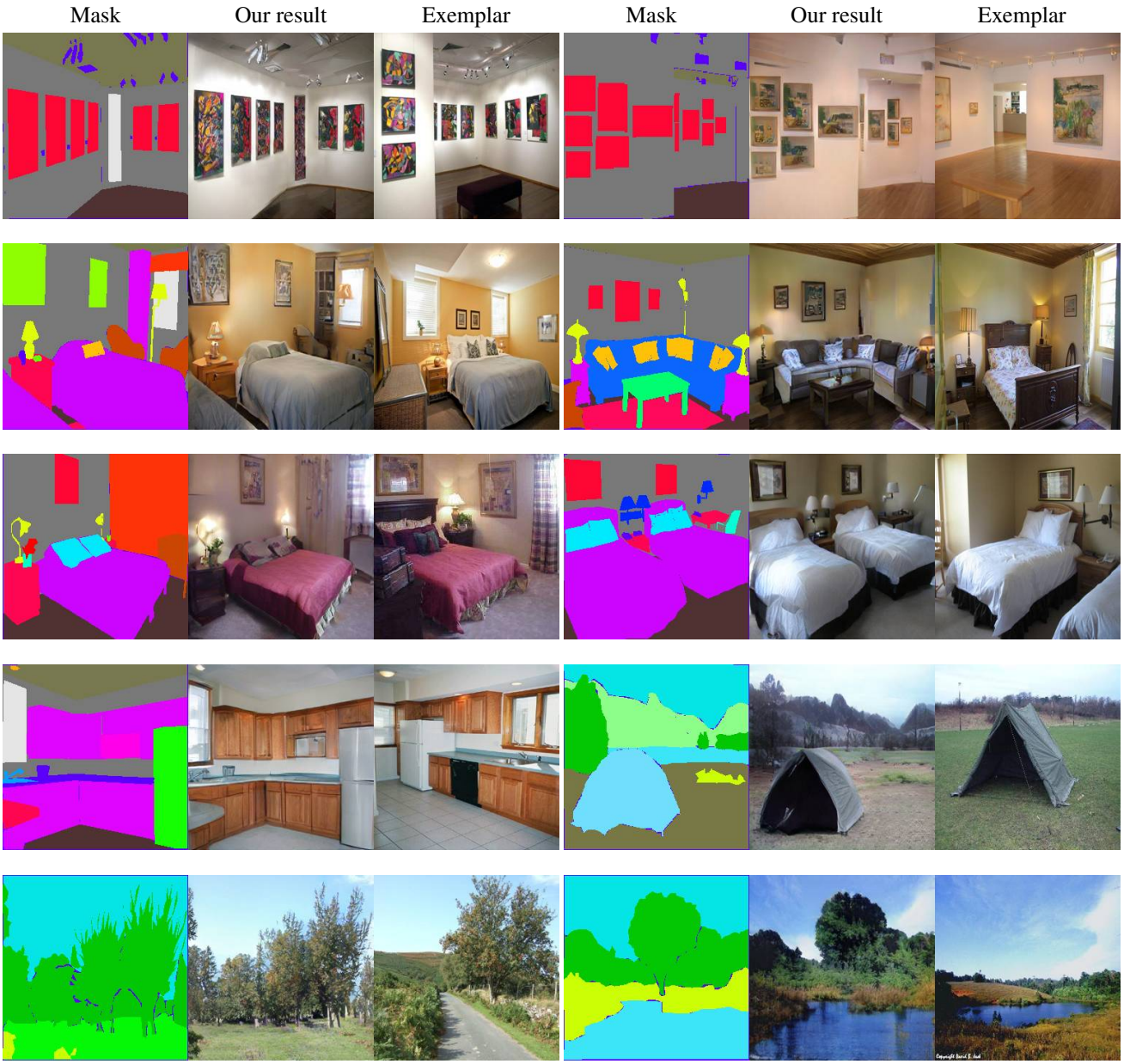

Figure 10: Mask-to-image generation results. (ADE20K dataset)

4. Oil portrait

Figure 11 to Figure 12 show more results of oil portrait. Our method takes the edge from real people (CelebA dataset) as input. The output looks like transferring the real person into the oil painting. While the model is purely trained using the paintings in the MetFaces dataset, the model could generalize well to the sketches of real faces.

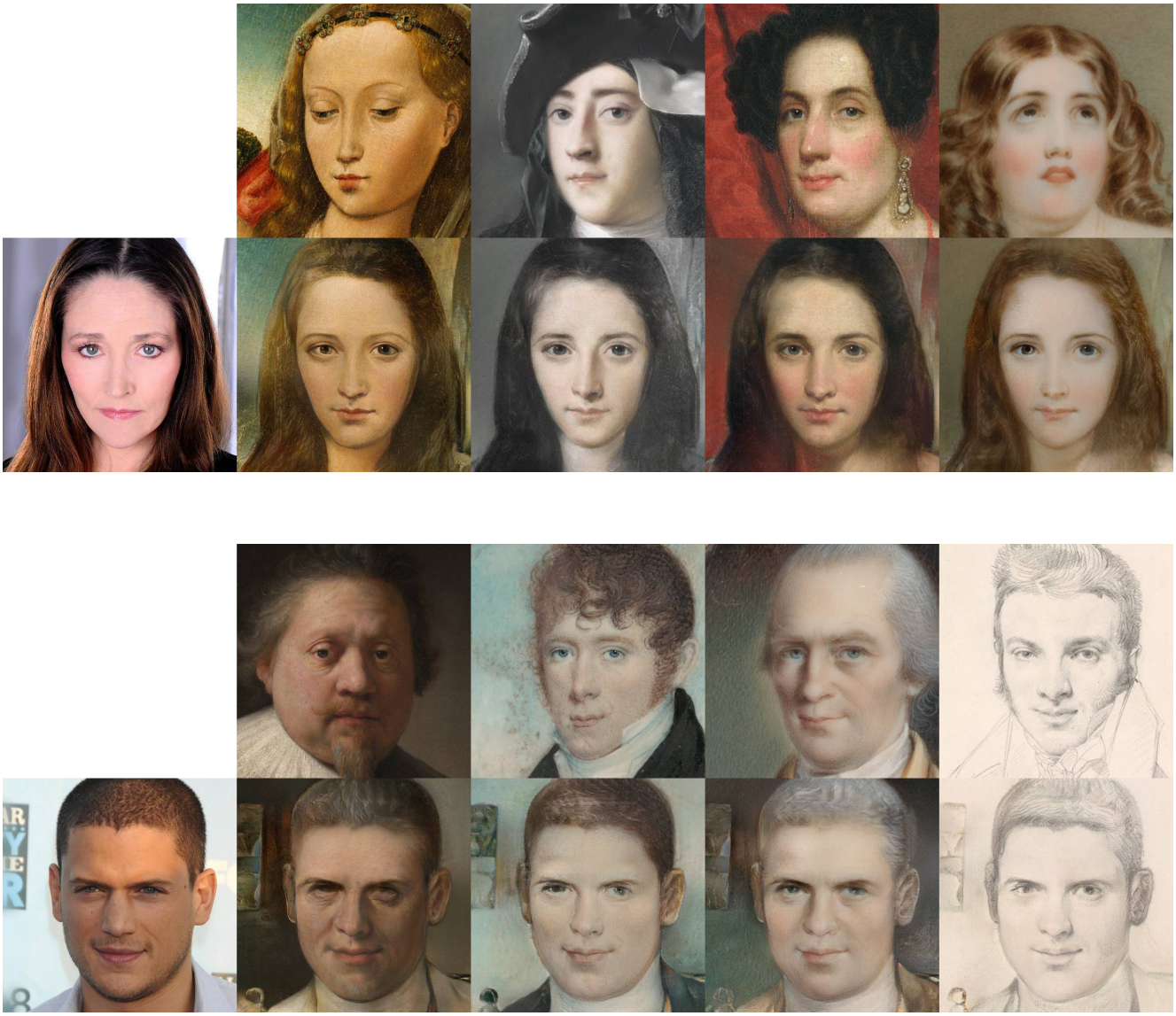

Figure 11: Oil portrait results with resolution 512×512. The edge is from the CelebA dataset while the exemplar is from the MetFaces dataset. 1st row: exemplar images, 2nd row: generated images.

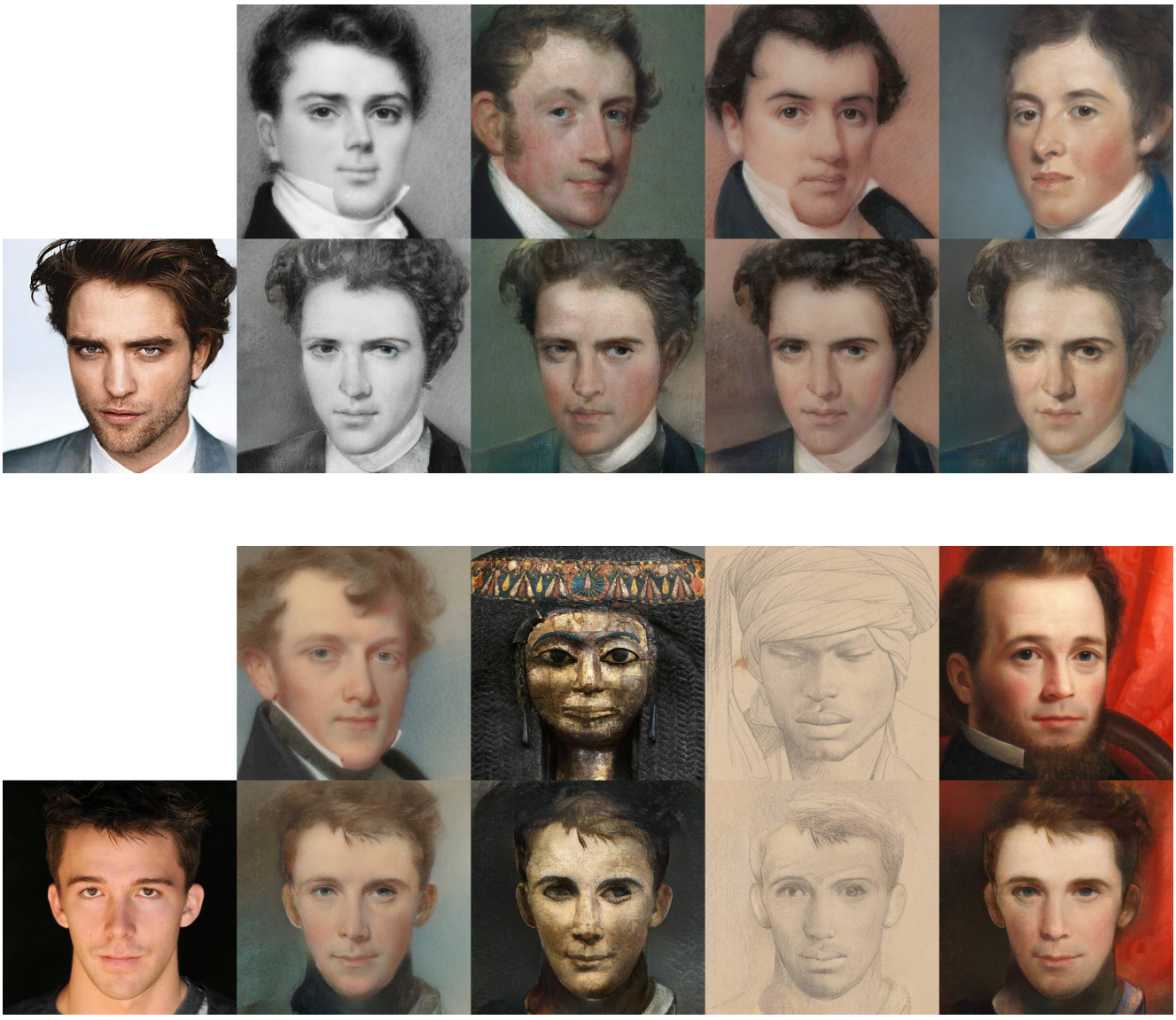

Figure 12: Oil portrait results with resolution 512×512. The edge is from the CelebA dataset while the exemplar is from the MetFaces dataset. 1st row: exemplar images, 2nd row: generated images.

B. Implementation Details

Our Hierarchical GRU-assisted PatchMatch establishes full-correspondence with multi-level features. We take the generation resolution 512×512 as an example to elaborate upon the implementation details. We choose $L = 4$ levels for the resolution 512×512 translation, so we establish correspondence on the 64×64, 128×128, 256×256, and 512×512 levels.

Hierarchical strategy. Our method establishes the correspondences via the hierarchical strategy. The smallest scale that we use in the experiments is 64×64. Please note that we calculate all the pair-wise similarities on this scale, *i.e.*, we make the two features ($\mathbf{f}_1^x$ and $\mathbf{f}_1^y$) flatten and calculate the similarity matrix on this scale. We do not rely on sparse matching and spatial propagation at this scale because the correspondence learning is guided by the warped images – in an indirect manner rather than providing the correspondence ground-truth, and it is difficult to use sparse matching to establish reliable correspondence when the features are not well-learned and appear noisy at the early training phase.

GRU-assisted PatchMatch. The GRU-assisted PatchMatch module requires the local correspondences in the subsequent higher-scale level, *i.e.*, the scale of 128×128, 256×256, and 512×512. We choose $K = 16$ nearest neighbors as candidates for each feature point and the PatchMatch is differentiable as we compute the soft matching by averaging across all these matchings and gradient can be back-forwarded to multiple locations.

Translation network. The translation network takes the warped exemplar images of multi-levels as input and synthesizes the final output according to the exemplar style. The warped exemplar images of multi-levels are first resized to the same scale (512×512 in this example) and then concatenated along the channel dimension. Two convolutional layers digest this concatenation input and produce the parameters for style modulation. We use *positional normalization* [1] within this sub-network, with the denormalization modulated by the warped exemplar.

The detailed architecture. Table 1 shows the implementation details of our method, with the naming convention as the CycleGAN [2]. Please note we take the generation at the resolution 512×512 as an example, and the network can be adapted to even higher resolutions.

Geometric data augmentation. The geometric augmentation $\mathcal{T}$ includes flip, random crop, and piecewise affine transformation, which is used to form the pseudo exemplar $\mathcal{T}(x_A)$. The training triplet thus becomes: input x_A, pseudo exemplar $\mathcal{T}(x_A)$ and the desired output x_B.

The protocol of sampling examples. 1) For training: we randomly sample images of the same person but of a different pose as exemplars for DeepFashion whereas example oil portraits are randomly sampled for MetFaces; for ADE20k, we retrieve the top 20 images in terms of pretrained VGG feature distance. 2) For figures reported in the paper: we randomly sample 10 exemplars for DeepFashion and MetFaces, and retrieve top-5 exemplars for ADE20k.

The speed. It takes around 200h to train 100 epochs using 8 NVIDIA V100 GPUs. During inference, it takes $\sim$0.6s to synthesize an 512×512 image. This information will be added in the final paper.

Sub-network	Module	Layers in the module	Output shape (H×W×C)
Multi-level Domain Alignment Network	Adaptive Domain Feature Encoder×2	Conv2d / k3s1 + Resblock / k3s1	512×512×64
		Conv2d / k4s2 + Resblock / k3s1	256×256×128
		Conv2d / k4s2 + Resblock / k3s1	128×128×256
		Conv2d / k4s2 + Resblock / k3s1	64×64×512
		Bilinear Interpolation + Resblock / k3s1	128×128×256
		Bilinear Interpolation + Resblock / k3s1	256×256×128
		Bilinear Interpolation + Resblock / k3s1	512×512×64
	Correspondence	(GRU-assisted PatchMatch & Warping) ×4	64×64×3
			128×128×3
			256×256×3
			512×512×3
Translation Network	Style Encoder×7	Bilinear Interpolation	$h^i \times w^i \times 3$
		Conv2d / k3s1	$h^i \times w^i \times 128$
		Conv2d / k3s1	$h^i \times w^i \times c^i$
	Generator	Conv2d / k3s1	8×8×1024
		Resblock×7	256×256×64
		Conv2d / k3s1	256×256×3

Table 1: The detailed architecture of our approach. k3s1 indicates the convolutional layer with kernel size 3 and stride 1.

References

[1] Boyi Li, Felix Wu, Kilian Q Weinberger, and Serge Belongie. Positional normalization. In *Advances in Neural Information Processing Systems*, pages 1622–1634, 2019. 13

[2] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232, 2017. 13