# Dressing in Order: Recurrent Person Image Generation for Pose Transfer, Virtual Try-on and Outfit Editing

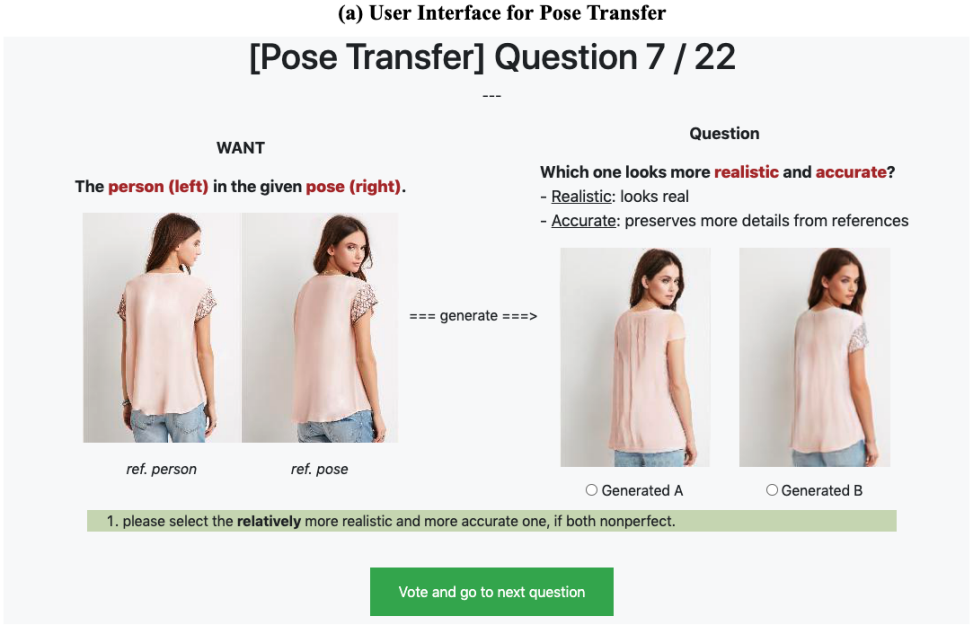
Aiyu Cui Daniel McKee Svetlana Lazebnik
University of Illinois at Urbana-Champaign
{aiyucui2,dbmckee2,slazebni}@illinois.edu

In this supplemental material, we first show the user interface that we used for our user study. Next, we provide a number of examples for each application that our DiOr system supports. Then, we present an investigation of how well the DiOr system can control garment transparency by altering the soft shape mask $M_g$. Finally, we visualize the segmentations used in the sIoU metrics, which we proposed for pose transfer evaluation. The contents will be demonstrated as following:

## Contents

## 1. User Study Interface

Here we show the user interface from our user study. For both pose transfer and virtual try-on, 22 questions are presented to each user. Only one question is displayed at a time. When the users click the "next question" button, they proceed to the next question and cannot go back. As shown in Figure 1(a), for pose transfer, the users see a person in both source and target pose. Users are asked to choose the more realistic and accurate result from two generated images. The two options are randomly sorted, with one output coming from our large model and the other from one of the compared models. In Figure 1(b), for virtual try-on, the users are provided with the reference person and target garment (upper-clothes). They are then once again asked to choose the better result from two randomly sorted generated images in terms of realism and accuracy.



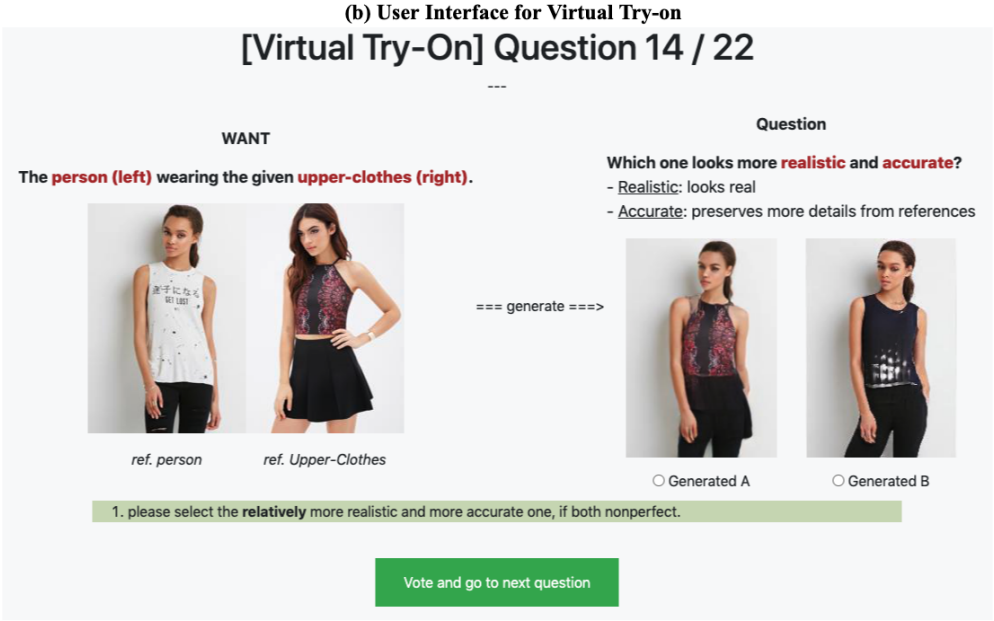




Figure 1. **User Study Interface.** (a) User Interface for pose transfer. (b) User interface for virtual try-on.

## 2. More Examples for Applications

More examples are reported for each application as follows.

### 2.1. Pose Transfer

Figure 2 shows a random batch of pose transfer outputs from the test set. We include the ground truth, output of ADGAN [3], GFLA [4], our small model and our large model.

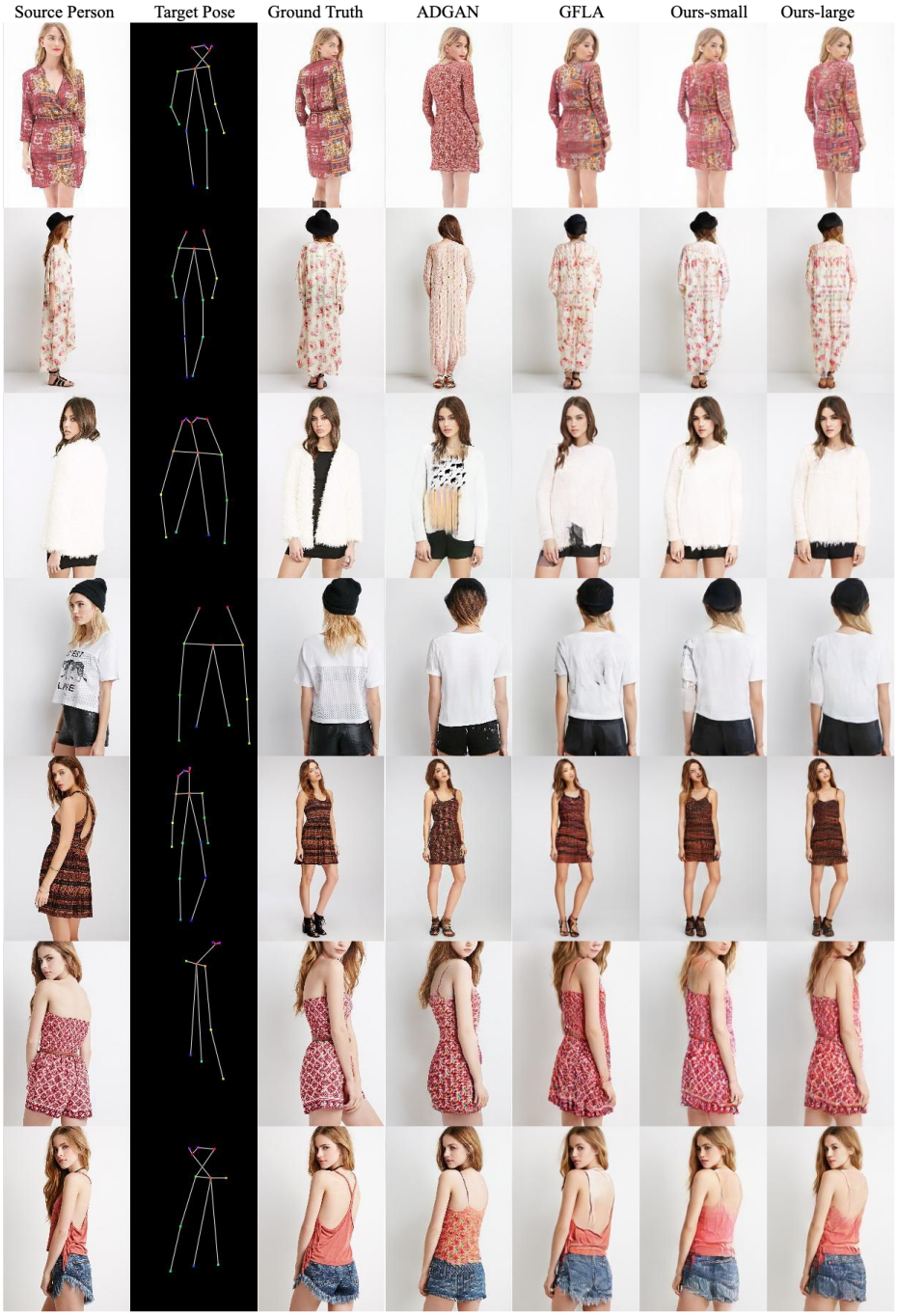



Figure 2. **Pose Transfer.**

## 2.2. Virtual Try-on

For every person, we show try-on for two garments. As we have already shown the results of upper-clothes try-on, here we present dress try-on in Figure 3(a), pants try-on in Figure 3(b) and hair try-on in Figure 3(c). Note that we treat hair as a flexible component of a person and group it as a garment, so that we can freely change the hair style of a person.

In Figure 3, the first column is the target person, the next four columns include the first selected try-on garment with output results on the target person, and the last four columns include the second selected try-on garment with output results for the same target person. We provide generation results from ADGAN [3], our small model, and our large model.

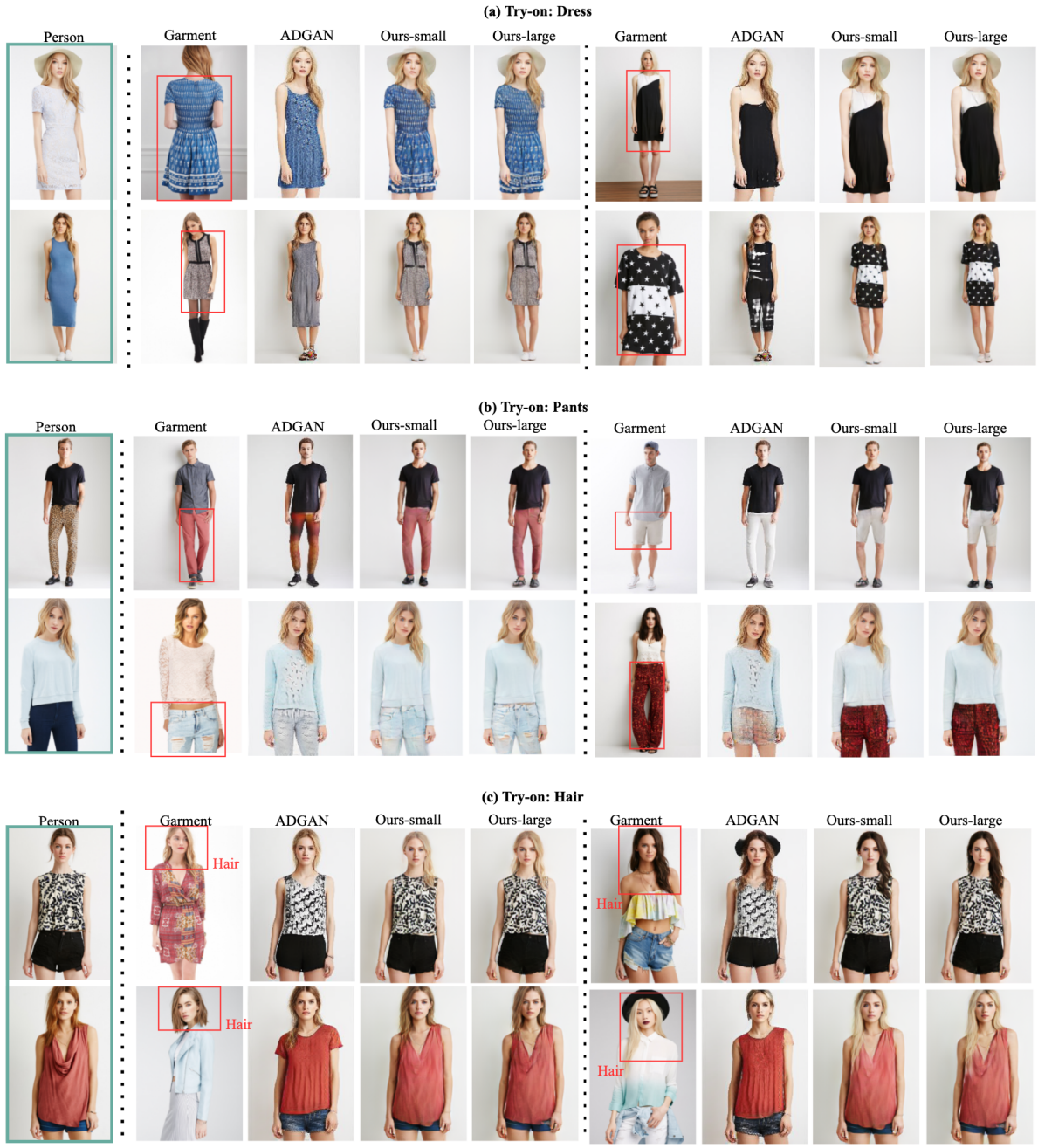



Figure 3. **Virtual Try-on.**

## 2.3. Dressing Order Effects

We can achieve different looks from the same set of garments with different orders of dressing (e.g., tucking in or not). Figure 4 demonstrates results from our large model for a person (first column) trying on a particular garment (the second column) with a different dressing order. Figure 4(a) shows the effect of dressing order for tucking or not, while Figure 4(b) demonstrates wearing a dress above or beneath a shirt.

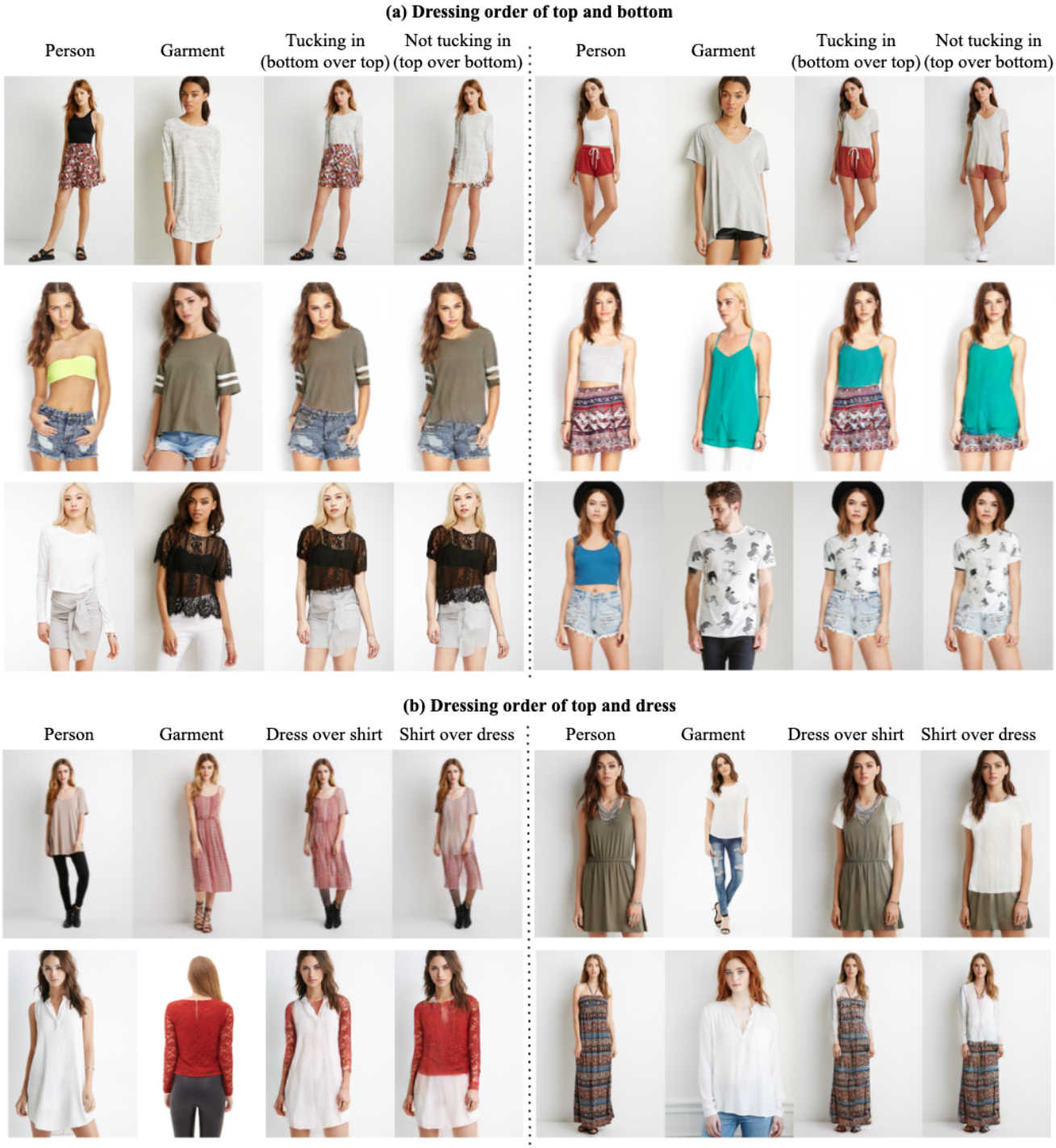



Figure 4. **Dressing in order.**

## 2.4. Layering

Here we include additional examples to demonstrate layering a single garment type. In Figure 5(a), layering a new garment outside the existing garment is demonstrated on the left and layering a garment inside the existing garment is shown on the right. Figure 5 (b) shows more examples of layering two garments on top of the original garments.

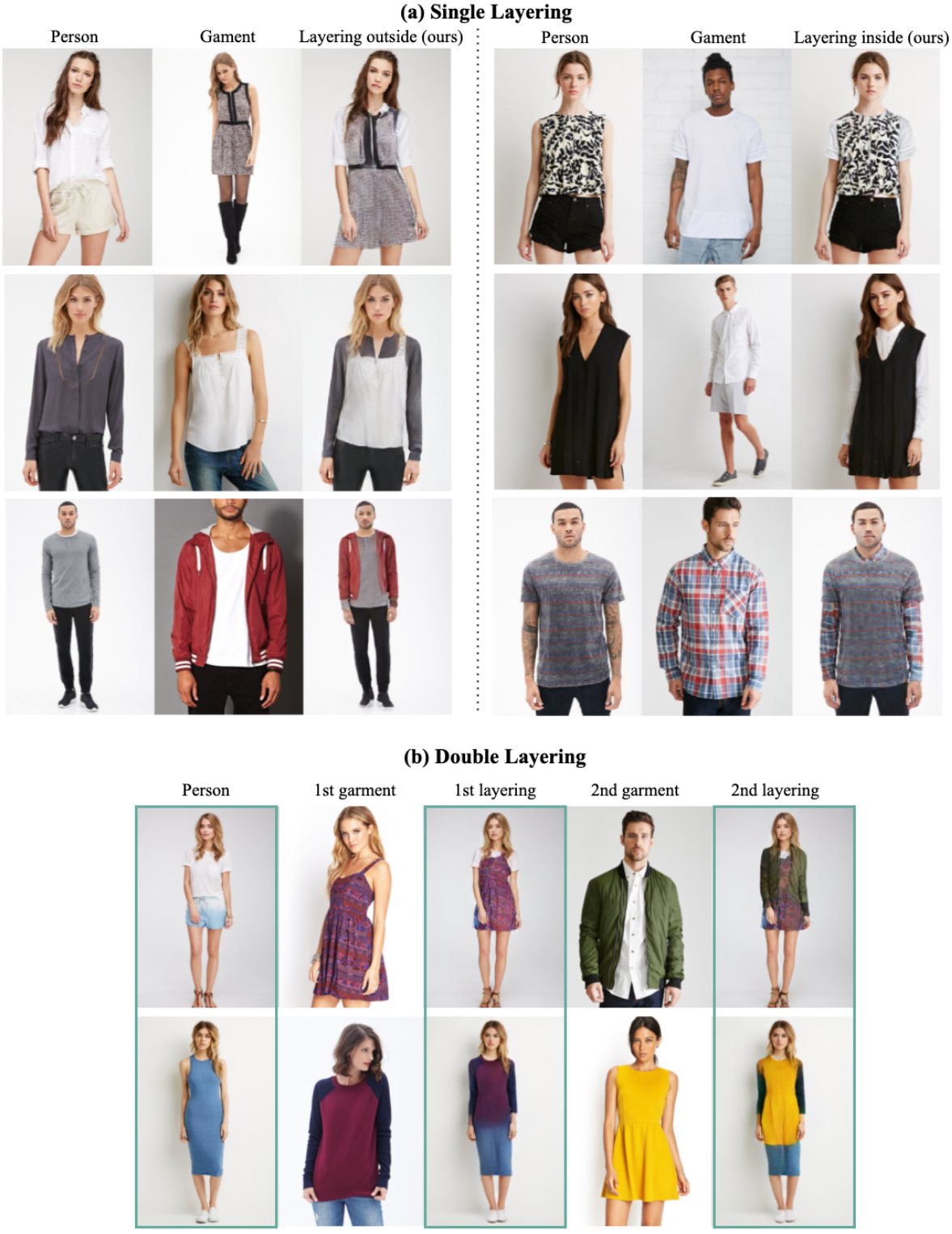



Figure 5. **Dressing in order.**

## 2.5. Content Removal

To achieve content removal, we can mask out an unwanted region in the associated texture feature map for a garment. Results are shown in Figure 6. In the bottom left example, although the girl's hair is partially masked out, we can remove only the pattern from the dress while keeping the hair, unlike the traditional inpainting methods. This is because the hair and dress are considered different garments and processed at different stages. This shows that our proposed person generation pipeline can better handle the relationships between garments.


Figure 6. **Content Removal**

## 2.6. Print Insertion

More results for print insertion are presented in Figure 7. From the left example, our model can warp a pattern onto existing garments, which is challenging for conventional harmonization methods. Additionally in the right example, although the print was placed partially on top of the hair, our novel pipeline can still render the hair in front of the inserted print. This is done by setting our model processing order to first generate the print and then generate the hair.


Figure 7. **Print Insertion.**

## 2.7. Texture Transfer

In Figure 8(a), we can achieve texture transfer by switching the texture feature map $T_g$ for a garment. In Figure 8(b) we also transfer texture from external patches. We crop the texture from the patch using the soft mask $M_g$ (resized to image size) and encode the masked patch as the new texture feature map. We show results of naive crop and paste along with results produced by our large model.

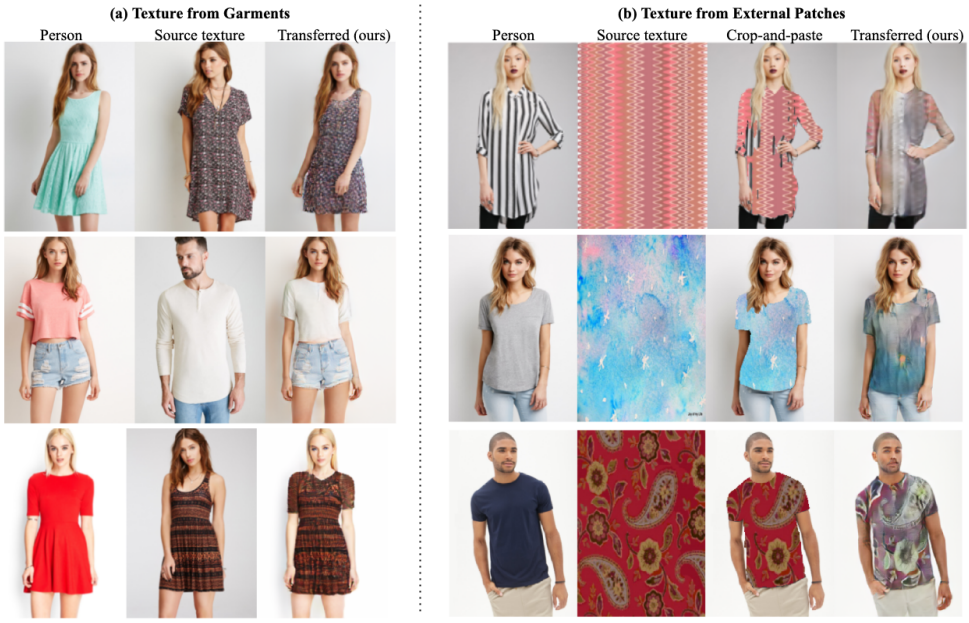



Figure 8. **Texture Transfer.**

## 2.8. Reshaping

We can reshape a garment by replacing its associated soft shape mask $M_g$ with the desired shape. In Figure 9, the left column shows shortening of long garments, and the right column shows lengthening of short garments.

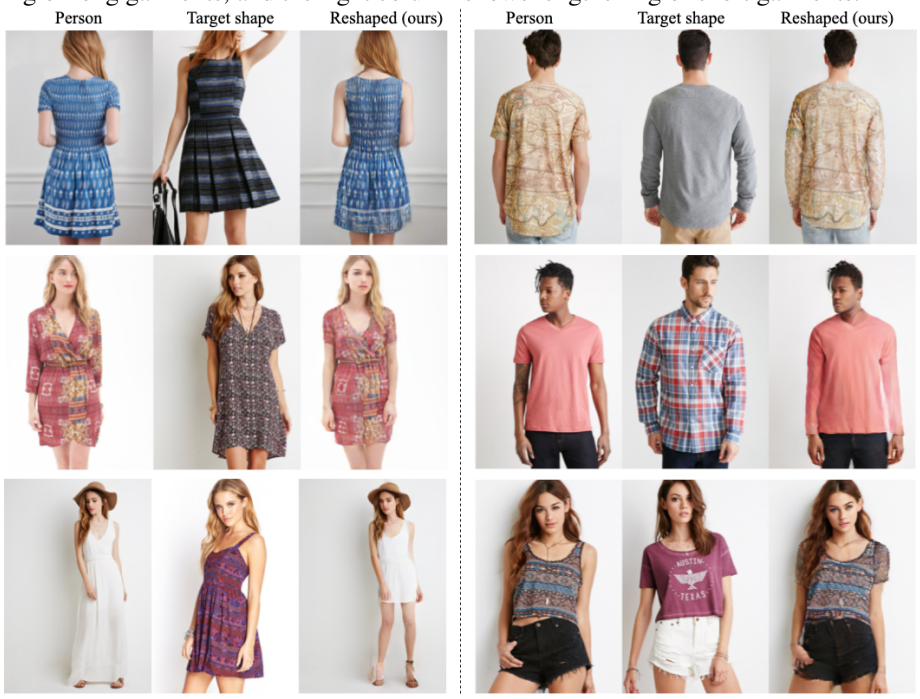



Figure 9. **Reshaping.**

## 3. Garment Transparency

We also provide an investigation of how well the soft shape mask $M_g$ can control the transparency of garments in our DiOr system. In Figure 10, we show examples of a person trying on a new garment and then reapplying the original garment on top. When layering the original garment, we control its transparency by altering the soft shape mask $M_g$ with a transparency factor $a$ setting as $M_g[M_g > a] = a$.

For the first three examples, our method can control the transparency of the outermost garments well. However, in the fourth and the fifth examples, the lace parts on the garments become a solid peach color with increasing $a$. Ideally, these lace parts should show the color of the underlying garment in this case rather than the color of skin.


Figure 10. **Transparency.**

## 4. Metrics: sIoU

In the evaluation tables in the main paper, we introduce sIoU (mean IoU of segmentations) as a metric to measure the layout consistency between the ground truth and the generated images for the pose transfer task. The segmentations are detected by an off-the-shelf human parser [1]. The human parser is trained on ATR labels [2], and will detect 18 classes: Background, Hat, Hair, Sunglasses, Upper-clothes, Skirt, Pants, Dress, Belt, Left-shoe, Right-shoe, Face, Left-leg, Right-leg, Left-arm, Right-arm, Bag and Scarf.

To further demonstrate the effectiveness of sIoU as an evaluation metric, we show a batch of test images with their detected segmentations in Figure 11. From Figure 11, we can see that the segmentation will change according to the quality of generation. Therefore, the IoU between the ground truth and generated images can effectively measure the quality of generation in terms of whether or not the layout matches.

In Figure 11, we report the mean IoU value for each method right below the method label. sIoU for GFLA is computed with ground truth using 256×256 resolution. All other methods are computed with ground truth using 256×176 resolution. Note that in the figure, each reported sIoU value corresponds to a single pair of (ground truth image, generated image). Additionally, note that not all classes are necessarily present for a single pair of images. Therefore, for this figure only, we modify the mean IoU to be the average over IoUs of those classes which either exist in the ground truth segmentations or exist in the generated images' segmentations.


Figure 11. **Visualization of detected human parsing for sIoU.**

## References


[1] Peike Li, Yunqiu Xu, Yunchao Wei, and Yi Yang. Self-correction for human parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2020.

[2] Xiaodan Liang, Si Liu, Xiaohui Shen, Jianchao Yang, Luoqi Liu, Jian Dong, Liang Lin, and Shuicheng Yan. Deep human parsing with active template regression. *IEEE transactions on pattern analysis and machine intelligence*, 37(12):2402–2414, 2015.

[3] Yifang Men, Yiming Mao, Yuning Jiang, Wei-Ying Ma, and Zhouhui Lian. Controllable person image synthesis with attribute-decomposed gan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5084–5093, 2020.

[4] Yurui Ren, Xiaoming Yu, Junming Chen, Thomas H Li, and Ge Li. Deep image spatial transformation for person image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7690–7699, 2020.