

Industrial Style Transfer with Large-scale Geometric Warping and Content Preservation: Supplementary Material

Jinchao Yang^{1*}, Fei Guo^{1*}, Shuo Chen², Jun Li^{1†}, Jian Yang¹
¹PCA Lab, Nanjing University of Science and Technology ²RIKEN

{yangjinchao, feiguo, junli, csjyang}@njjust.edu.cn shuo.chen.ya@riken.jp

*Contributes equally †Corresponding author&project lead

S. Supplementary Material

This supplementary material consists of the following five parts: an explanation of the smoothness mask generation (subsection S.1), intermediate results of different deformation degrees (subsection S.2), potential applications (subsection S.3), a description of our evaluation interface for the user study (subsection S.4), and more comparison results (subsection S.5)

S.1. Smoothness Mask Generation

Relying only on the shape-consistency loss is not enough because there exist many distinct warp fields that can make the loss minimized. Due to the lack of guidance on sampling direction, although it can achieve shape matching between masks, the post-deformation result on the image domain may be very random and chaotic, as shown in Fig. S1. Therefore, we must further restrict the sampling direction of the warping field to retain the content details of the source object, to the greatest extent. In this work, we abide by the deformation rule of an edge-to-edge sampling, which can make full use of the source content, and propose a mask smoothness regularization based on it.



Figure S1. The shape-consistency only ensures the shape matching between masks and cannot further restrict the sampling direction, which may cause chaotic deformation on the image domain.

To achieve the goal of the edge-to-edge sampling, under the shape-consistency loss, we add a smoothness constraint on both sides of the target edge (M_{edge}) to the warp field, as illustrated in Fig. S2. Firstly, the displacement on both sides of the target edge should be as close as possible. Secondly,

the shape and background areas of the target will sample from the shape (purple arrow) and background (blue arrow) areas of the source, respectively. Combined with the two constraints described above, the target edge region will tend to sample from the source edge.

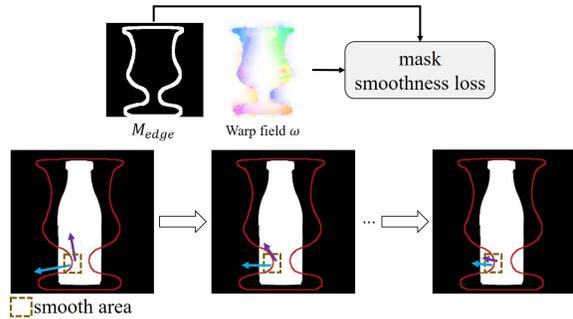


Figure S2. By adding a smoothness constraint on both sides of the target edge (M_{edge}), the target edge region (take brown square area as example) tend to sample from the source edge.

More specially, we divide the deformation into two parts, a compressed part (like the handle part of the cup in Fig. S2) and an expanded part (like the body and bottom part of the cup). When there is no constraint on the shape-consistency loss, Fig. S1 shows that the compressed part looks like an edge truncation and the extended part randomly samples from the source shape area. To overcome the truncation problem, we add the edge smoothness mask into the region for imposing the edge-to-edge sampling in the compressed part. To avoid the blur problem, we use the smoothness constraint in the extended part. As a result, we explore the smoothness mask in the form shown in Fig. S3.

S.2. Intermediate Results of LGW

Since the LGW module generates a prediction sequence, we expect to reflect the deformation process. Because the latter warp field produces a larger scale warping, we use increasing weights to balance the shape-consistency loss

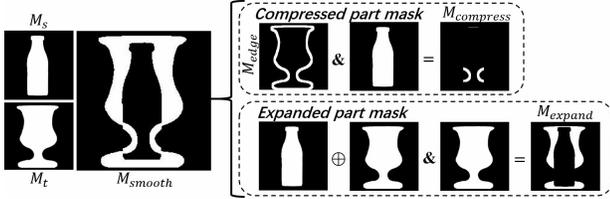


Figure S3. Given two shapes, we design different smoothness masks for the two parts of compression (upper right) and expansion (lower right). The smoothness mask in the middle is what we use in the smoothness regularization.

sequence between the warped source mask and the target mask. As the latter warp field needs to relax smoothness requirements more than the former, we employ decreasing weights to balance the smoothness loss sequence. They are described as

$$L_{shape} = \sum_{r=1}^R \alpha_r \|\omega_r(M_s) - M_t\|_1, \quad (S1)$$

$$L_{smooth} = \sum_{r=1}^R \beta_r L_{smooth}(\omega_r, \mathbf{M}), \quad (S2)$$

where R is the number of iterations, and we set $R = 3$ in our implementation. The increasing weights $\{\alpha_r\}_{r=1}^3$ are set to $\{0.1, 0.2, 1\}$, and the decreasing weights $\{\beta_r\}_{r=1}^3$ are set to $\{0.1, 0.05, 0.01\}$.

Once trained, the LGW network can be used to generate warp fields iteratively which become increasingly accurate with respect to the geometric shape of the target, as shown in Fig. S4.

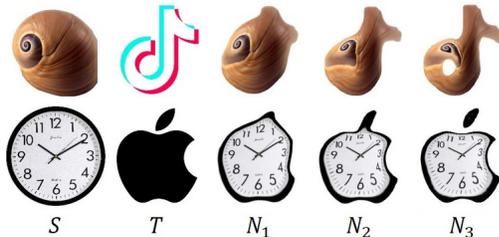


Figure S4. Intermediate results $\{N_r\}_{r=1}^3$ with three warping fields $\{\omega_r\}_{r=1}^3$.

S.3. Potential Applications

In addition to the smoothness regularization applied to the warp field, we can also restrict the sampling direction by dividing different corresponding areas into many masks. As shown in Fig. S5, we expect the specific *WARP* area in the colored M_t to be warped from the cyan circle in the plate. We achieve it by iterating a warp field based on L_{shape} and L_{smooth} between M_s and M_t . N is the finally warped result. So, depending on users' preferences, they can customize the various artistic product, for example, the design of clothing with a logo in Fig. S5.

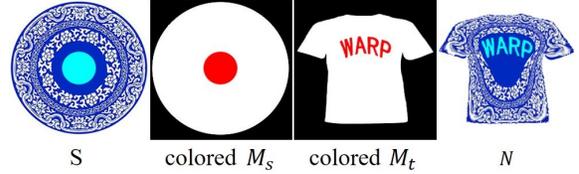


Figure S5. Restricting the sampling direction by using colored masks. Here, we expect the specific *WARP* area in the cloth to be warped from the cyan circle in the plate.

S.4. User Study

As described in subsection 4.2 of the main paper, we conduct a user study to evaluate the effect of the proposed InST algorithm against the existing methods. We divide the evaluation into three groups from the perspectives of geometric warping, content maintenance, and their combination, and each group is set to ten options. The evaluation interface is partly shown in Fig. S6.

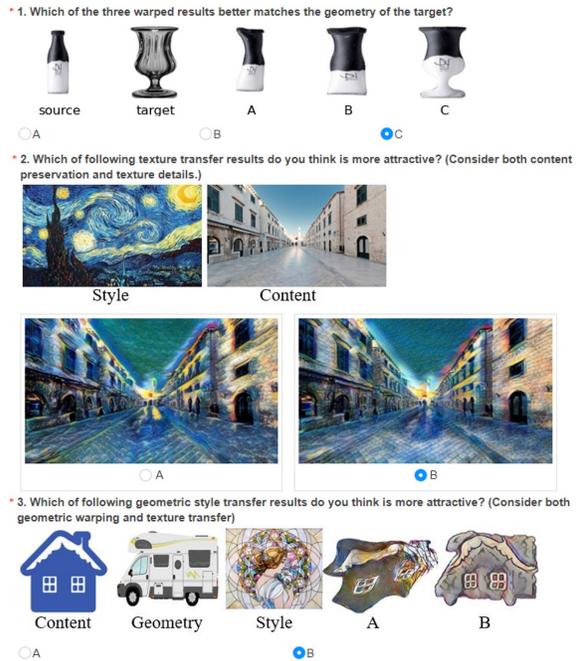


Figure S6. Evaluation Interface

S.5. More Results

Here, more qualitative results are provided to assist the readers in assessing the effectiveness of our proposed InST algorithm. We showcase the comparison results from three aspects like the main paper: (i) geometric warping (Fig. S7, Fig. S8), (ii) texture transfer (Fig. S9), (iii) their combination (Figs. S10 and S11).



Figure S7. More visual logo and product design results using the geometric transfer methods, *e.g.*, DST [3], GTST [5] and our InST.

References

- [1] Jie An, Siyu Huang, Yibing Song, Dejing Dou, Wei Liu, and Jiebo Luo. Artflow: Unbiased image style transfer via reversible neural flows. In *CVPR*, pages 862–871, 2021. 5
- [2] Xun Huang and Serge J Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*, pages 1501–1510, 2017. 5
- [3] Sunnie S. Y. Kim, Nicholas I. Kolkin, Jason Salavon, and Gregory Shakhnarovich. Deformable style transfer. In *ECCV*, pages 246–261, 2020. 3, 4
- [4] Xueting Li, Sifei Liu, Jan Kautz, and Ming-Hsuan Yang. Learning linear transformations for fast arbitrary style transfer. In *CVPR*, pages 3809–3817, 2019. 5
- [5] Xiao-Chang Liu, Yong-Liang Yang, and Peter Hall. Learning to warp for style transfer. In *CVPR*, pages 3702–3711, 2021. 3, 4, 5, 6

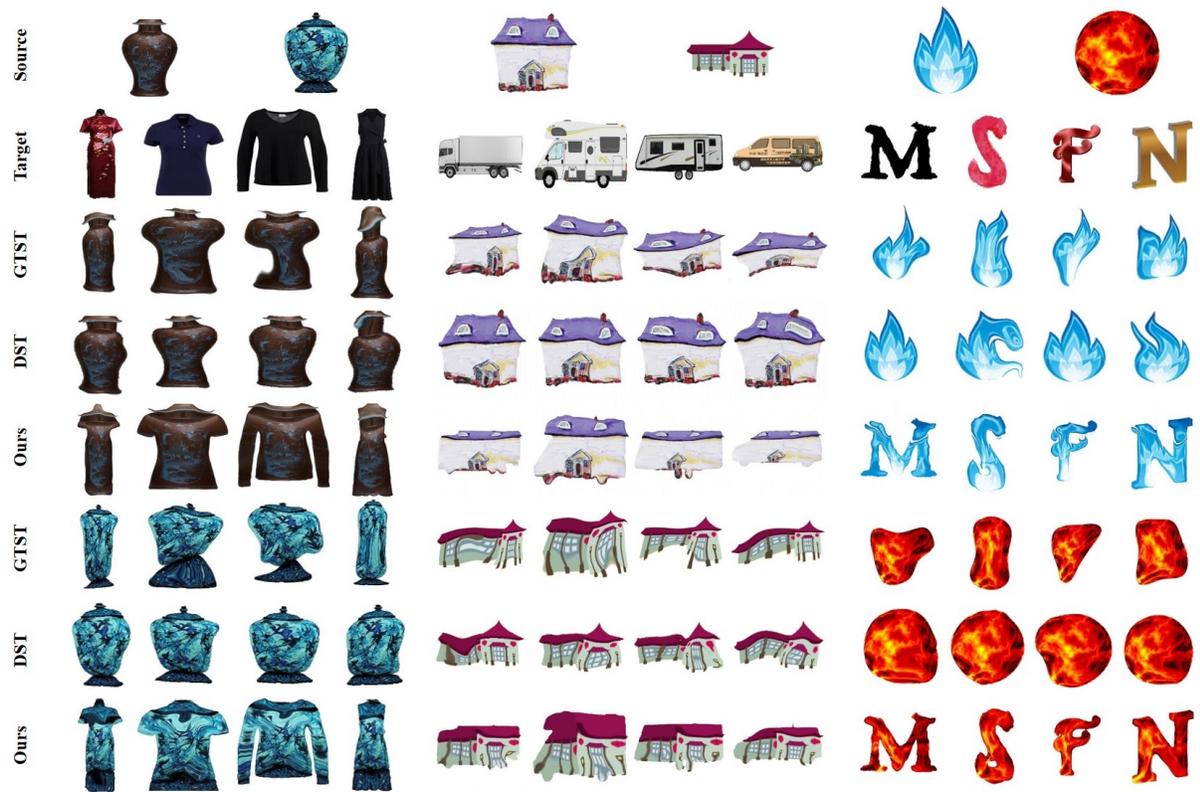


Figure S8. More visual product design results using the geometric style transfer methods, *e.g.*, DST [3], GTST [5] and our InST.



Figure S9. More content preservation results using the texture style transfer methods, *e.g.*, AdaIN [2], LinearWCT [4] and ArtFlow [1].

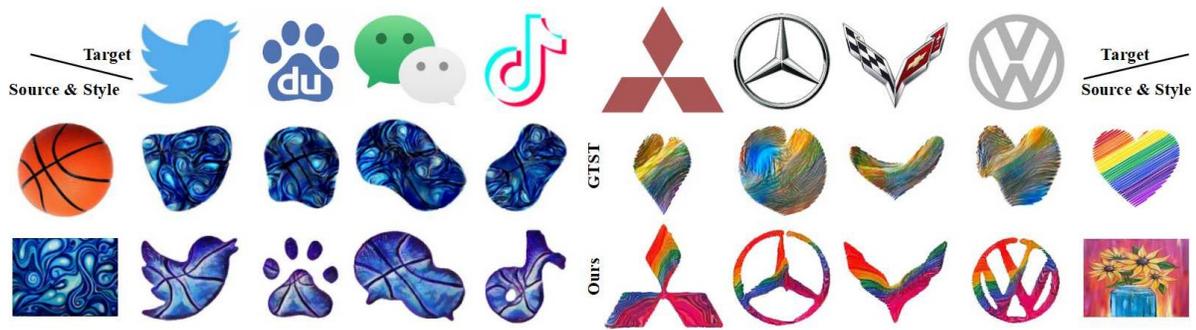


Figure S10. More visual logo design results using the geometric and texture style transfer methods, *e.g.*, GTST [5] and our InST.



Figure S11. More visual product design results using the geometric and texture style transfer methods, e.g., GTST [5] and our InST.