Eye-for-an-eye: Appearance Transfer with Dense Semantic Correspondence in Diffusion Models

Sooyeon Go Kyungmook Choi Minjung Shin Youngjung Uh*
Yonsei University, Seoul, South Korea
{sooyeon8658, kyungmook.choi, smj139052, yj.uh}@yonsei.ac.kr

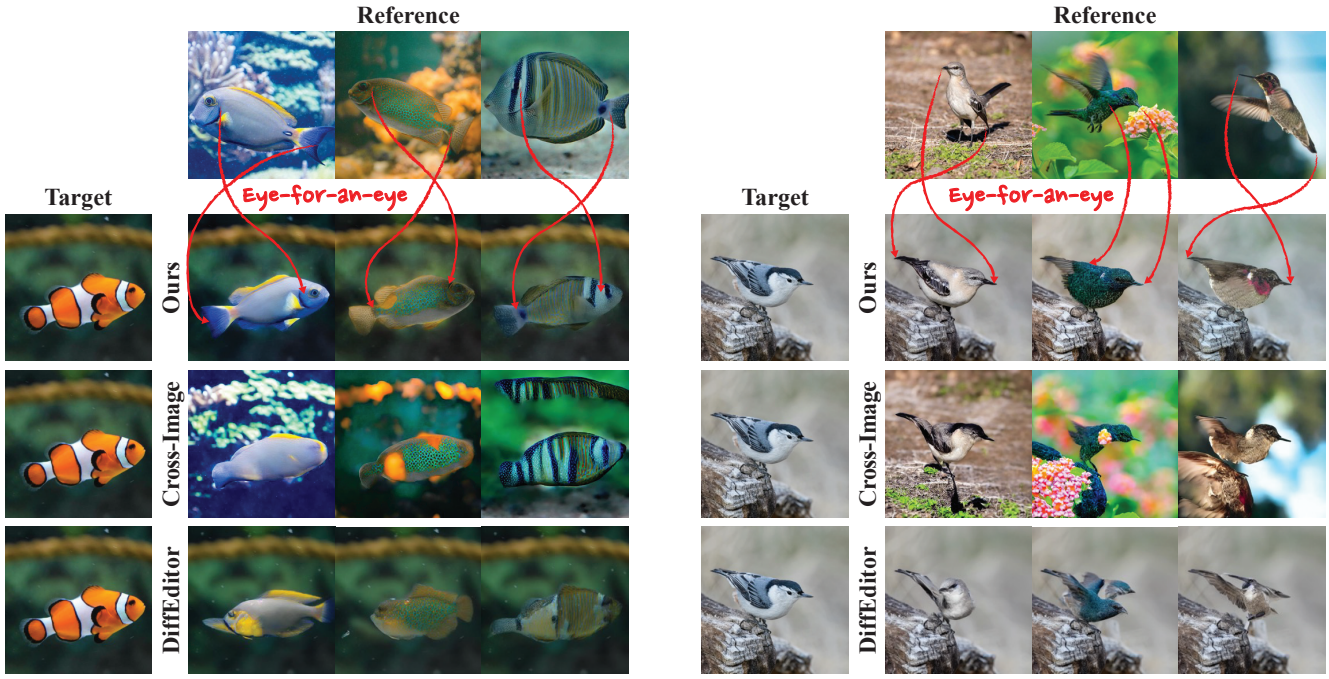

Figure 1. Our method transfers semantically corresponding appearances from reference images to target images. In contrast to other methods such as DiffEditor [29] and Cross-Image [1], our method preserves the structure of the target images successfully transfers the colors and patterns considering the semantic meanings from the references.

Abstract

*As pre-trained text-to-image diffusion models have become a useful tool for image synthesis, people want to specify the results in various ways. This paper tackles training-free appearance transfer, which produces an image with the structure of a target image from the appearance of a reference image. Existing methods usually do not reflect semantic correspondence, as they rely on query-key similarity within the self-attention layer to establish correspondences between images. To this end, we propose explicitly rearranging the features according to the **dense** semantic correspondences. Extensive experiments show the superiority of our method in various aspects: preserving the structure of the target and reflecting the correct color from the reference, even when the two images are not aligned.*

1. Introduction

Text-to-image diffusion models [35] generate high-quality, realistic images from textual inputs. Although it allows users to easily describe the desired results, it falls short in more specific controls that are difficult to be described in texts. Alternatively, it is easier for the users to provide reference images and carrying their specific elements to the results.

Such elements include shapes [49], main subject [4, 18], and most of the images for partial editing [3, 10, 43]. We tackle a scenario with two input images, where the result has the shape of one image and the color pattern of the other. It is often called appearance transfer from a reference image to a target image.

Although previous methods for appearance transfer [1, 12, 28, 29] are promising, they struggle when the poses are not aligned. Fig. 1 shows that they often transfer eyes to tails and tails to heads. Hence, we hypothesize that the solution lies in establishing correspondences between the target and reference. A straightforward solution for the above problem would be a two-stage procedure: finding semantic correspondences [37, 47, 48] and following them to transfer the reference to the target. However, most semantic matches produce sparse key-point correspondences and dense correspondences are not accurate enough. Moreover, the two-stage approach is inherently cumbersome and costly.

In this paper, we analyze the limitations of previous methods with a self-attention injection or two-stage approach and propose *Eye-for-an-eye*. It consists of three parts: finding correspondences, transferring features, and recursively running them through a generative process. As a whole, it considers *dense semantic correspondences* to transfer the appearance of a reference image to the target image. Our method has non-trivial design choices as follows. We find that the cosine similarity between features of reference and target features before the self-attention layer allows for more semantically meaningful matching than the attention mechanism between the reference key and target query within self-attention. Then we replace the target features with the reference features rearranged according to our correspondence. It accurately keeps the target structure in the result. We recursively run this operation along the generative process.

Our method accurately transfers the appearance from precise locations in a reference image even in challenging scenarios involving complex colors and patterns, or diverse views and poses. In addition, our results maintain the structure of the target image. We demonstrate the superiority of our method compared to previous methods with extensive qualitative and quantitative evaluation. Beyond intra-domain appearance transfer, our method generalizes to cross-domain appearance transfer and supports applying different appearances to multiple objects. Ours not only achieves superior appearance transfer results but also shows the best dense correspondence performance compared to existing semantic matching methods.

2. Related Work

Appearance transfer Appearance transfer produces an image that combines the shape and color patterns of two different images. This is accomplished by training on each target domain [7, 14, 30] and using either input image pairs [39] or using external models to guide diffusion model [23]. While these methods maintain the structure of the target image, they tend to struggle with unaligned images or those from different domains. Recent approaches [1, 12, 28, 29] excel with images from different domains without requiring fine-tuning. However, self-guidance [12] leads to discrepancies in color distribution between the output and reference images because they make the *average* features of the output similar to the reference. The methods with key-value injection [1, 28] expect the attention mechanism to find the semantic similarity for the transfer. The attention mechanism often produces wrong semantic correspondences such as beaks to wings and tails to heads as shown in Fig. 1. In contrast, our method transfers appearance following correct semantic correspondence, even in a training-free manner.

Manipulating features for image editing Recent approaches [1, 5, 11, 15, 18, 24, 26, 31, 32, 40] explore the manipulation of attention layers of pretrained diffusion models for image editing. In this context, PnP-diffusion [40] leverages the semantic information in self-attention layers, demonstrating that modifying attention features can be used for editing tasks without requiring fine-tuning. MasaCtrl [5] and Cross-Image [1] replace the key and value features in the self-attention layer to achieve text-guided translation of reference images. However, we observe that query-key attention maps and the weighted summation in the self-attention are insufficient for transferring semantically matched appearances. Therefore, instead of directly injecting entire key-value pairs or whole features, we propose injecting features after rearranging them based on their semantic correspondence.

Semantic correspondence Leveraging the diffusion features of models, unsupervised semantic correspondence methods [17, 37] outperform other weakly-supervised methods. SD-DINO [48] further enhances this performance by incorporating DINO ViT [2] as an additional feature extractor. Recent approaches [5, 40, 46] observe that semantic understanding in diffusion models is distributed across timesteps and U-Net layers. Consequently, Diffusion Hyperfeatures [27] leverages these distributed representations by integrating feature maps across timesteps, demonstrating their effectiveness in keypoint correspondence tasks. To incorporate semantic information distributed across timesteps into appearance transfer, we rearrange the feature maps according to the correspondences found at each timestep.

3. Method

We aim to transfer the appearance of objects from a reference image I^{ref} to a target image I^{target} based on semantic correspondences between them. The appearance includes attributes such as the color and pattern of the object. As

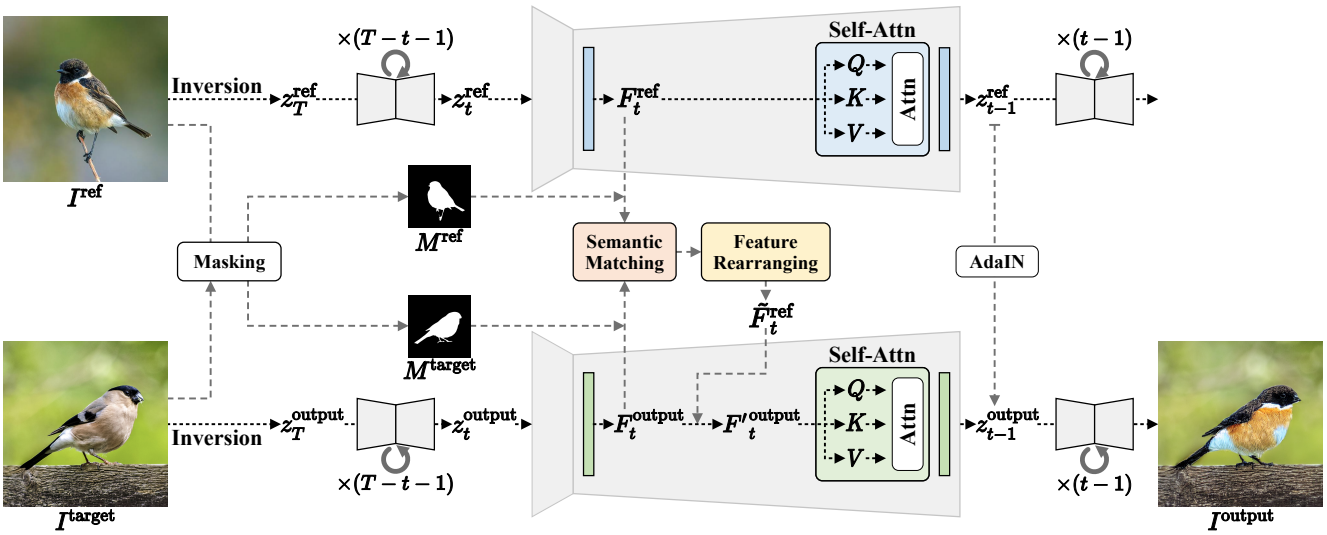

Figure 2. **Pipeline of our method.** We transfer the semantically corresponding appearance of objects from a reference image to a target image. Given I^{ref}, I^{target}, and their masks M^{ref} and M^{target}, we find semantic correspondences between their features before the self-attention layers F_t^{ref} and F_t^{output}. Then, we inject the rearranged features based on these correspondences.

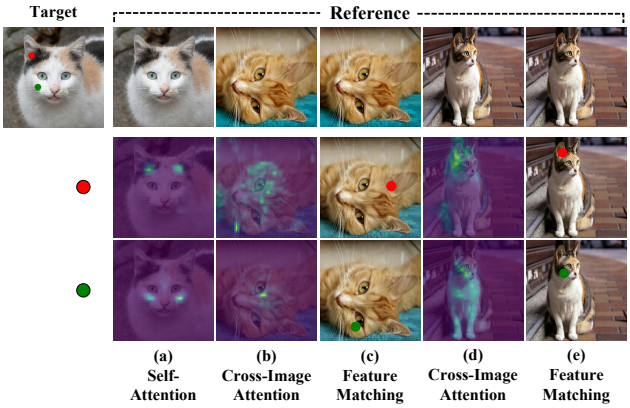

Figure 3. **Query-key attention maps vs. our feature matching.** For each query pixel $\mathbf{q}$ denoted by colored markers in the target image, we show the attention maps based on the QK attention score. (b) and (d) include other regions in the attention map where matching is incorrect. In contrast, the feature matching in (c) and (e) presents a single point with the correct semantic meaning.

shown in Fig. 2, our method produces an image from an output denoising process (the target process being injected with reference features) starting from an inversion [20] of I^{target} (bottom) with modifications from another reference-denoising process starting from an inversion of I^{ref} (top). The modification includes finding semantic correspondences between the two denoising processes and injecting features with rearrangement.

3.1. Revisiting of self-attention

In the self-attention layer of the U-Net in Stable Diffusion [35], an attention map is generated by applying a dot product and softmax to the query Q and key K, which indicates positional similarities. The weighted sum of this attention map and value V allows each position in Q to aggregate relevant information from similar positions in the K-V pairs and Q .

Recent appearance transfer methods [1, 28] introduces KV injection, which integrates K_{ref}-V_{ref} pairs from a reference denoising process into the target denoising process. During this process, the KV injection aggregates V_{ref} based on the attention map between Q_{target} and K_{ref}. Therefore, while the attention map of Q_{target} and K_{ref} indicates the similarity with Q_{target} for determining the location from which to aggregate V_{ref}, it does not represent correspondence matching between the target image and the reference image, as shown in Fig. 3. As a result, this can lead to transfers with mismatched semantic meanings.

Moreover, the self-attention with the KV injection aggregates features from *multiple locations* of the reference rather than borrowing a feature from a single point. Although it might be a good way to transfer global style, it prevents the results from having clear local appearances.

In the following subsection, we propose our method that resolves the above flaws. To ensure that the transferred appearance aligns with the semantic meaning, we rearrange the features before the self-attention layer of the reference denoising process and inject it into the target process, which is intended to rearrange Q_{ref}. Rearranging the reference features before the self-attention layer through precise corre-

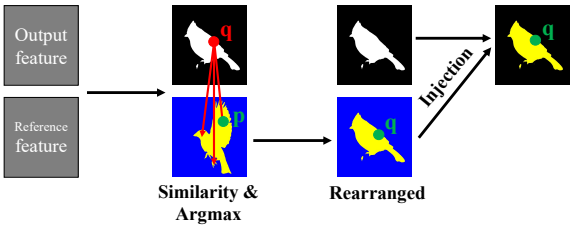

Figure 4. **Feature rearrangement and injection.** The reference feature, rearranged based on similarity to the output feature, is injected into the output denoising process.

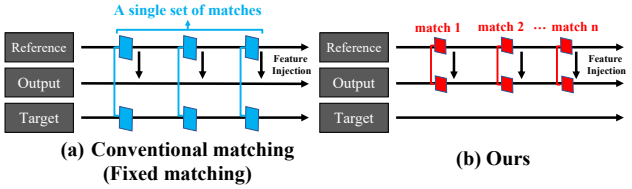

Figure 5. **Comparison between conventional matching methods and ours.** (a) Conventional methods aggregate features from multiple time steps of the reference and target into a single set and perform matching only once. (b) Ours matches the reference features with the output features and performs multiple matches across individual steps.

spondence matching and injecting them into the output denoising process yields better semantic alignment than KV injection.

3.2. Semantic matching-based feature rearrangement

As described in Sec. 3.1, previous appearance transfer methods with KV injection do not always reflect *semantic* correspondences between the reference and the target objects. On the other hand, we explicitly rearrange the reference feature map to match the spatial arrangement of semantics with the target feature map.

To find the semantic correspondence of a pixel $\mathbf{q}$ among the reference image at pixel $\mathbf{p}$, we take the $\arg\max$ of the cosine similarity in the feature map before l-th self-attention layer at denoising timestep t. Additionally, to preserve the background of the target image, we apply object masks M^{target} and M^{ref} to the target and reference features:

$$F^{\text{output}}, F^{\text{ref}} = F^{\text{output}}[M^{\text{target}}], F^{\text{ref}}[M^{\text{ref}}], \tag{1}$$

$$\mathbf{p} = \underset{\mathbf{p}\in[0,h)\times[0,w)}{\arg\max} \ \text{sim}\left(F^{\text{output}}(\mathbf{q}), F^{\text{ref}}(\mathbf{p})\right), \tag{2}$$

where $F^{\text{out}} \in R^{hw\times c}$ and $F^{\text{ref}} \in R^{hw\times c}$ are the feature maps of the target and reference; sim computes cosine similarity. l and t are omitted from $F^{*}_{l,t}$ for brevity. $[\cdot]$ denotes slicing by the object mask.

Then, we rearrange the reference feature map to reflect semantic correspondence, defining it as $\tilde{F}^{\text{ref}}(\mathbf{q}) = F^{\text{ref}}(\mathbf{p})$.

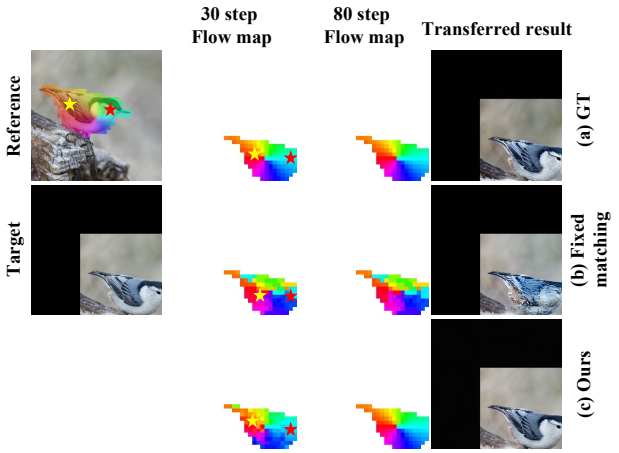

Figure 6. **Fixed matching vs. Our matching.** (a) Feature injection with Ground Truth (GT). GT represents the feature transferred from the reference. (b) Feature rearrangement and injection with fixed matching from conventional matching method [37]. (c) Feature rearrangement and injection at each step (ours). Our method performs dense matching closely aligned with the ground truth in later steps. Key-point matching at early time steps is represented as star markers.

This rearrangement is equivalent to modifying Q in self-attention. Since Q represents the object's structure, the rearrangement aligns the structure of the reference object to match the target object based on semantic matching.

Finally, we inject the rearranged reference features into the output denoising process. Reference features are aligned to the structure of the target object, enabling an effective transfer of the reference object's appearance.

$$F'^{\text{output}} = \tilde{F}^{\text{ref}}_{m} \odot M^{\text{target}} + F^{\text{output}} \odot (1 - M^{\text{target}}), \tag{3}$$

where $\odot$ represents the Element-wise product. Fig. 4 provides these processes of feature rearrangement and injection. Then the self-attention becomes

$$\text{softmax}\left(\frac{Q'^{\text{output}}(K'^{\text{output}})^{T}}{\sqrt{d}}\right) V'^{\text{output}}, \tag{4}$$

where $Q'^{\text{output}} = F'^{\text{output}}W^{\text{query}}$, $K'^{\text{output}} = F'^{\text{output}}W^{\text{key}}$, and $V'^{\text{output}} = F'^{\text{output}}W^{\text{value}}$.

Additionally, the transferred output often has different color brightness and contrast when compared to the reference. To address this issue, we apply AdaIN[19] used in Cross-Image [1] to masked noise, thereby reducing the color discrepancy between the reference and the output.

We next compare the matching processes of conventional methods and our method to highlight their differences. Conventional matching methods [27, 37, 47, 48] and ours find semantic correspondence by computing cosine similarity between two sets of feature pairs. As illustrated in Fig. 5

(b), ours matches the reference features with *the recursively transferred output features* and produces *multiple matches* at multiple individual time steps. It enables sparse key-point matching in the early step and dense matching in the later step, meaning that it can effectively capture both key-point and dense correspondence. As shown in Fig. 6 (c), while the early step flow map demonstrating dense matching is noisy compared to the ground truth (a), the sparse key point correspondences are accurate, and the flow map in the later step closely resembles the ground truth. This later step's noise-free dense matching leads to a clean transferred result. In contrast, conventional matching methods find matches between two fixed sets from reference and target features (Fig. 5 (a)). Each fixed set forms *a single set of matches*, either by aggregating features from multiple time steps [27] or by using features from an early time step [37, 48]. Semantic correspondence found with a single set of matches is suitable for finding sparse key-point matching at the RGB level, but inadequate for finding dense matching. In Fig. 6 (b), the sparse key point correspondence in the early step is accurate, whereas the dense correspondence contains a lot of noise compared to the ground truth. This noisy dense correspondence leads to a noisy transferred result. With our improved dense matching, our transferred results are more realistic and have fewer artifacts than the ones from the transferred results using conventional methods, i.e., SD-DINO [48], DIFT [37] and TLFR [47].

4. Experiments

Competitors We compare our results with recent training-free diffusion-based methods, including Cross-Image [1], which uses KV injection, DiffEditor [29], which shows the best result among methods with score-based editing guidance [12, 28, 29], and DiffuseIT [23], which leverages external models for guidance. In addition, we compare our method with the optimization-based approach Splice ViT [39], the domain-specific trained Swapping Auto-Encoder (Swapping AE) [30], and also with IP-Adapter [44] and ZeST [8], which adjust the appearance of the reference image based on a depth map input through Controlnet [49]. In Appendix D, we provide more details on implementation and hyperparameters for each method.

Evaluation Metric In appearance transfer, the key evaluation factors are: 1) whether the appearance information from the reference is transferred to the correct location, and 2) whether the structure of the target object is well-preserved. For appearance evaluation, we assess the preservation of the overall color distribution by comparing the color histograms (A_{hist}) and evaluatethe semantic consistency by comparing the CLIP embeddings (A_{clip}) between the reference and transferred objects using the object masks. To evaluate structure preservation, we assess how much the structure in the generated result deviates from the target image using the following metrics. First, we evaluate semantic consistency by comparing key points (S_{key}) detected by ViT-Pose [42] in the result image with the ground truth key points in the target. Also, to assess depth accuracy and object shape consistency, we calculate the RMSE of the depth maps (S_{depth}) and the mean intersection over union (S_{miou}) of the object masks [21]. Finally, we measure dense correspondence using the method from a previous study [48], calculating the L1 distance of the flow map (D_{flow}). Additional details on evaluation metrics are provided in Appendix B, and explanations of the datasets used for evaluation can be found in Appendix C.

Method	A_{hist} ↓		A_{clip} ↑	
Dataset	Building	AFHQ	Building	AFHQ
Ours	0.469	0.577	95.30	97.03
Cross-Image	0.491	0.608	94.05	96.75
DiffEditor	0.478	0.614	91.35	96.13
DiffuseIT	0.477	0.607	90.74	96.21
Splice ViT	0.472	0.580	94.64	96.30
Swapping AE	0.481	0.629	85.90	92.35
IP-Adapter	0.487	0.616	93.46	96.76
ZeST	0.497	0.602	89.59	96.04

Table 1. **Quantitative evaluation for appearance similarity.** We mark the best score in red and the second-best score in yellow.

4.1. Appearance similarity

As shown in Fig. 7, our method successfully transfers the correct appearance from the reference to the target, even when the target and reference images are not aligned. For instance, in the bird examples, our results capture and reflect the complex patterns of the reference image, preserving the color arrangement of the blue head, green wings, and red-and-green body, while competitors fail to retain this arrangement. Furthermore, due to our method of rearranging features according to semantic meaning, the car and cat examples demonstrate our method's robustness in cases where the reference and target are either unaligned or differ in size. Also, our results accurately reflect the reference object's color, while IP-Adapter [44] and ZeST [8] generate unrealistic colors. Notably, as shown in Tab. 1, ours achieves the lowest A_{hist} score and the highest A_{clip}, highlighting its superior performance in preserving complex appearance patterns.

Additionally, as shown in Fig. 8, our method successfully transfers the reference's appearance across diverse domains, despite substantial differences domain between the reference and target images. Even when the reference and target belong to different domains, it successfully transfers the appearance of similar semantic meanings, such as a bird's wing to an airplane's wing.

4.2. Structure preservation

As shown in Tab. 2, our method achieves high performance in S_{depth}, S_{miou}, and S_{key}, which evaluate structure preservation. Ours excel in both complex domains (e.g., buildings)

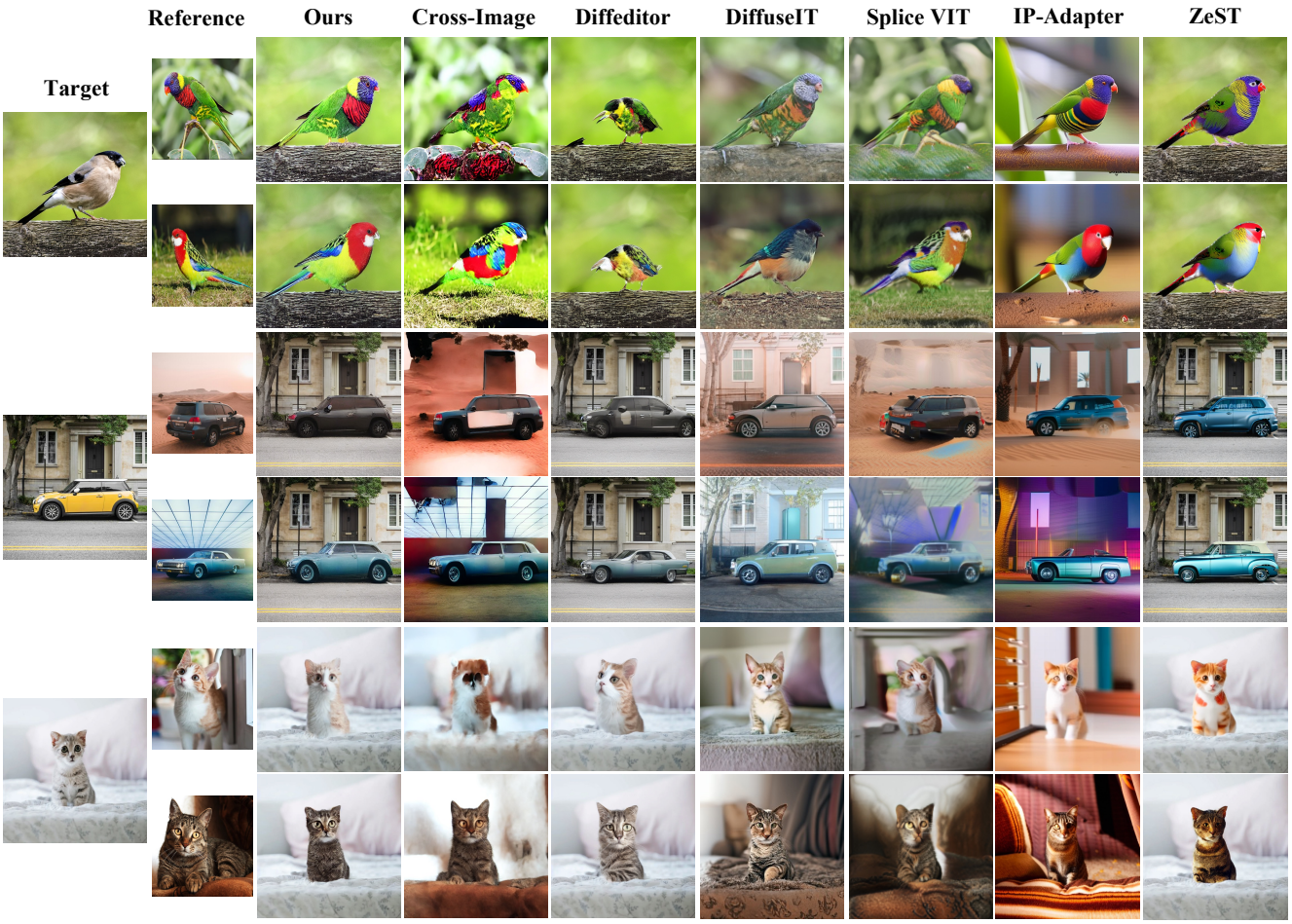

Figure 7. **Qualitative comparison.** We compare our results with the competitors on samples where the target and reference objects are unaligned, have complex patterns, or differ in size. The competitors struggle in various ways.

Figure 8. **Our results of various domain.** Our approach can transfer appearance across diverse domains.

Method	$S_{\text{depth}} \downarrow$		$S_{\text{miou}} \uparrow$		$S_{\text{key}} \uparrow$
Dataset	Building	AFHQ	Building	AFHQ	AP-10K
Ours	0.197	0.114	0.939	0.972	82.99
Cross-Image	0.287	0.139	0.758	0.915	64.49
DiffEditor	0.266	0.124	0.863	0.943	46.26
DiffuseIT	0.263	0.123	0.855	0.951	65.07
Splice ViT	0.319	0.120	0.842	0.943	47.54
Swapping AE	0.295	0.128	0.821	0.942	N/A
IP-Adapter	0.374	0.130	0.642	0.950	84.26
ZeST	0.242	0.119	0.925	0.980	78.25

Table 2. **Quantitative evaluation for structure preservation.** We mark the best score in red and the second-best score in yellow.

and simpler ones (e.g., AFHQ), where the reference and target have similar sizes and poses. While IP-Adapter [44] and ZeST [8] may appear sufficient, their high scores result from using depth as an additional condition. However, relying on such additional conditions to interpret the target structure can lead to incorrect estimations, especially when objects overlap or exhibit complex spatial arrangements, degrading object appearance transfer. In contrast, our method achieves competitive or superior results without requiring extra depth information. These quantitative results validate the structure-preserving capabilities observed in the qualitative examples in Fig. 7. Cross-Image [1], DiffEdit [10], Splice VIT [39], and IP-Adapter [44] produce results with altered target structures. The results highlight that ours notably outperforms in structure preservation and semantic consistency.

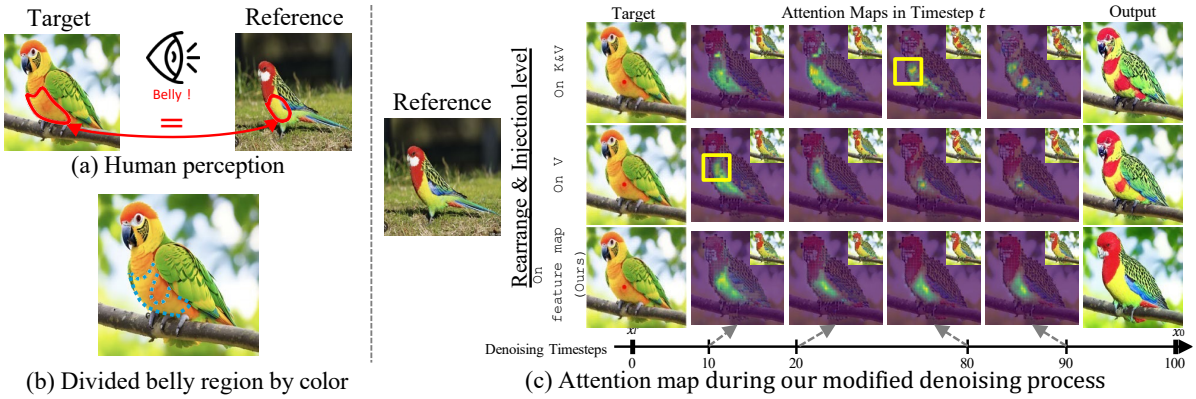

Figure 9. **Comparison of attention maps.** (a) shows the corresponding region between the target and the reference from human perception. (b) illustrates dividing regions based on color, not semantic meaning. (c) provides the attention maps for the target image's query pixel (red dot) at different timesteps during appearance transfer. The K&V modification (first row) and V modification (second row) perform semantic matching in the same manner as our method but apply the rearrangement and injection processes to K&V and V instead of the feature map, respectively. The image at the top right of each attention map represents the result of feature rearrangement, which is indirectly shown by rearranging the reference RGB image with semantic matching calculated from U-Net's 2nd up-block.

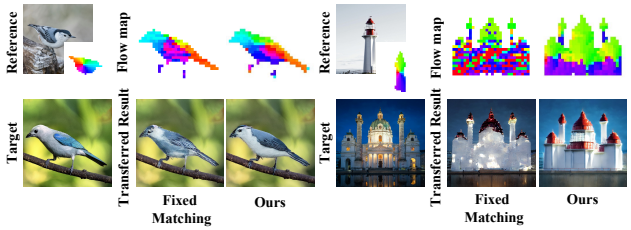

Figure 10. **Qualitative comparison between the result of fixed matching and ours.** The first row displays the reference color map and the flow map. The flow map shows the result of mapping the colors of reference pixels to matched target pixels. The second row compares the transfer result using fixed matching from an existing semantic matching method [48] with our transferred result.

In appearance transfer, both appearance similarity and structure preservation are crucial. As shown in Fig. 7, Tab. 1 and Tab. 2, compared to competitors that excel in only one aspect, ours achieves strong performance in both. It is possible due to the design of our method, which semantically rearranges the reference features to correspond with the target structure and injects them into object regions.

4.3. Dense correspondence evaluation

In this section, we demonstrate, through dense correspondence evaluation, why it is essential to perform matching at each generation step in our method.

Fig. 10 visualizes the correspondence between the reference feature map and the transferred feature map as a flow map. As a baseline, we adopt Fixed Matching, where the matching rule is determined by a single set of matches and applied across all generation steps. Our flow maps are smooth and free from noise, accurately reflecting the tendency of spatially adjacent pixels in an image to exhibit similar correspondences. In contrast, the Fixed Matching's flow maps contain more noise and lack smoothness. As a result, the transferred outputs from Fixed Matching exhibit significant noise and unnatural mismatched regions.

Method	$D_{\text{flow}} \downarrow$		
Dataset	FG3D CAR	JODS	PASCAL
Ours	9.43	28.75	21.83
TLFR[47] *	41.11	65.28	106.53
TLFR[47]	30.75	59.85	102.14
SD-DINO[48]	26.87	47.54	63.27
DIFT[37]	77.53	91.32	135.37

Table 3. **Quantitative evaluation for dense matching.** * indicates a fine-tuned backbone. We mark the best score in red and the second-best score in yellow.

For quantitative evaluation, Tab. 3 compares the conventional semantic matching method with ours. This is done using the optical flow smoothness metric D_{flow}, as employed in the dense matching evaluation protocol [48]. Our method achieves a significantly lower flow map distance compared to other semantic correspondence methods. Since the flow map distance increases when mismatching occurs or matching lacks smooth continuity, it indicates that our method demonstrates the highest dense correspondence performance among existing matching methods based on diffusion features. Please refer to Appendix B for the evaluation details.

4.4. Analysis of rearrange and injection component

In this section, we demonstrate that our feature injection aligns with human intuition. Fig. 9 (a) shows an example of appearance transfer that aligns with human expectations by considering semantic meaning; for instance, transferring the appearance of the reference belly to the target belly. Fur-

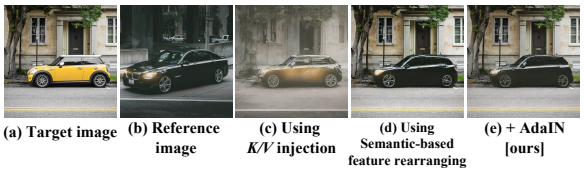

Figure 11. We perform an ablation study to validate our method.

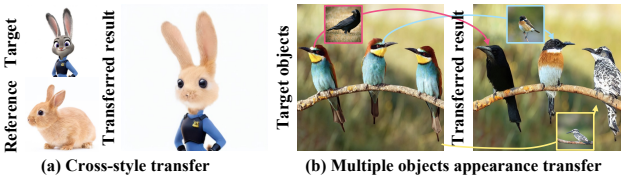

Figure 12. Results of various applications

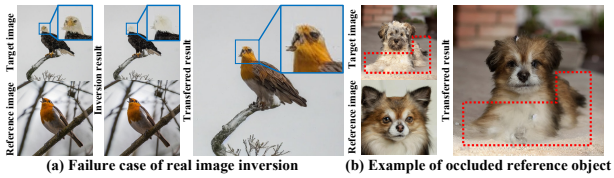

Figure 13. Results with limitations

thermore, humans tend to interpret even within the same region by segmenting colors, as in (b), rather than allowing arbitrary matches within a semantic region. Fig. 9 (c) demonstrates that our feature injection `Ours` process aligns with human intuition, unlike key-value injection (`on KV`) and value injection (`on V`).

`Ours` produces a clear, lime-like belly that accurately reflects the reference, while `on KV` and `on V` result in red smudges in the belly. We demonstrate this effect in Fig. 9 (c) by visualizing attention maps across denoising timesteps, where each attention map corresponds to the activation map for the red dot in the target image. `Ours` aggregates visual elements from the lime-colored reference belly according to its semantics. In contrast, `on KV` and `on V` focus on the red neck and head, disregarding the belly's semantics.

We suggest the reason as follows. Our method rearranges the reference F according to the semantic correspondence to the target and replaces the target F with it. Hence, our results have the visual elements of the reference arranged in the semantic structure of the target. In contrast, the Q in `on KV` and `on V` assigns high attention to the red color on the reference belly (yellow boxes in Fig. 9 (c)) and transfers it to the orange-ish belly of the target bird. It occurs because, unlike in `ours`, where the target Q is replaced with the rearranged reference Q, `on KV` and `on V` retain the original target Q, causing the target bird's belly color to be interpreted as a different structure (Fig. 9 (b)).

4.5. Ablation study

In this section, we perform ablation experiments regarding different components of our method and show its contribution in Fig. 11. Compared to KV injection (c), our semantic matching-based feature rearranging (d) transfers appearance to regions where the semantic meaning of objects aligns. For instance, unlike (c), where the reference car's headlights are transferred to the side of the target car, (d) correctly transfers the side of the reference car to the side of the target car, resulting in a properly transferred black car. In (e), the AdaIN on masked noise matches the global color distribution of the object, thereby maintaining the color brightness and contrast of the appearance object. We provide quantitative ablation results and more various ablation studies in Appendix E to Appendix J.

4.6. Application

Cross-style appearance transfer Our method enables semantic matching-based transfer even in challenging samples where the target object and the reference object belong to exhibit different styles. In Fig. 12, (a) depicts the appearance of a real rabbit applied to a Disney-style rabbit.

Multi-objects appearance transfer Our method can individually transfer the appearance of multiple objects in the target image, each from a different reference image. Objects in the target image are matched and rearranged one by one with the reference images. Each process is executed simultaneously within a single generation process, rather than sequentially. In Fig. 12, (b) presents the results with three birds in the target image, each with distinct appearances from three different images.

5. Conclusion

In this paper, we have introduced a semantic-based appearance transfer method using a pretrained text-to-image diffusion model. Our method faithfully reflects the reference image to the target image according to semantic correspondences, e.g., fin-to-fin and wing-to-wing, while previous methods often ignore semantics. Our key arguments for replacing features in the target denoising process with the reference denoising process are 1) reflecting dense semantic correspondences 2) found during the modified denoising process 3) on the input features of self-attention. Experiments demonstrate that our method achieves faithful appearance transfer between the semantically corresponding parts of the result and the reference and better preserves the structure of the target in the result compared to existing methods. Furthermore, we achieve significantly superior dense semantic correspondence results compared to existing semantic matching methods.

Limitation In order to use a real image as a reference, our method relies on inversion. If the inversion malfunctions, our method struggles as shown in Fig. 13 (a). Also, Fig. 13 (b) shows that the reference image does not have the semantically corresponding parts from the target image and our matching finds the most similar parts instead of semantic correspondence. Still, the results tend to be realistic in such cases.

References

[1] Yuval Alaluf, Daniel Garibi, Or Patashnik, Hadar Averbuch-Elor, and Daniel Cohen-Or. Cross-image attention for zero-shot appearance transfer. *arXiv preprint arXiv:2311.03335*, 2023. 1, 2, 3, 4, 5, 6, 12, 13

[2] Shir Amir, Yossi Gandelsman, Shai Bagon, and Tali Dekel. Deep vit features as dense visual descriptors. *arXiv preprint arXiv:2112.05814*, 2(3):4, 2021. 2

[3] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18208–18218, 2022. 2

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 2

[5] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. Masactrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22560–22570, 2023. 2

[6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021. 14

[7] Xi Chen, Lianghua Huang, Yu Liu, Yujun Shen, Deli Zhao, and Hengshuang Zhao. Anydoor: Zero-shot object-level image customization. *arXiv preprint arXiv:2307.09481*, 2023. 2

[8] Ta-Ying Cheng, Prafull Sharma, Andrew Markham, Niki Trigoni, and Varun Jampani. Zest: Zero-shot material transfer from a single image. *arXiv preprint arXiv:2404.06425*, 2024. 5, 6, 12

[9] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Stargan v2: Diverse image synthesis for multiple domains. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8188–8197, 2020. 13

[10] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. *arXiv preprint arXiv:2210.11427*, 2022. 2, 6

[11] Yilun Du, Joshua Meier, Jerry Ma, Rob Fergus, and Alexander Rives. Energy-based models for atomic-resolution protein conformations. *arXiv preprint arXiv:2004.13167*, 2020. 2

[12] Dave Epstein, Allan Jabri, Ben Poole, Alexei Efros, and Aleksander Holynski. Diffusion self-guidance for controllable image generation. *Advances in Neural Information Processing Systems*, 36:16222–16239, 2023. 2, 5

[13] Daniel Gatis. Rembg: A tool to remove image backgrounds, 2024. Accessed: 2024-11-21. 14, 15

[14] Vidit Goel, Elia Peruzzo, Yifan Jiang, Dejia Xu, Nicu Sebe, Trevor Darrell, Zhangyang Wang, and Humphrey Shi. Pair-diffusion: Object-level image editing with structure-and-appearance paired diffusion models. *arXiv preprint arXiv:2303.17546*, 2023. 2

[15] Jing Gu, Yilin Wang, Nanxuan Zhao, Tsu-Jui Fu, Wei Xiong, Qing Liu, Zhifei Zhang, He Zhang, Jianming Zhang, Hyun-Joon Jung, et al. Photoswap: Personalized subject swapping in images. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[16] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *2011 international conference on computer vision*, pages 991–998. IEEE, 2011. 13

[17] Eric Hedlin, Gopal Sharma, Shweta Mahajan, Hossam Isack, Abhishek Kar, Andrea Tagliasacchi, and Kwang Moo Yi. Unsupervised semantic correspondence using stable diffusion. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[18] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 2

[19] Xun Huang and Serge Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. In *Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017. 4

[20] Inbar Huberman-Spiegelglas, Vladimir Kulikov, and Tomer Michaeli. An edit friendly ddpm noise space: Inversion and manipulations. *arXiv preprint arXiv:2304.06140*, 2023. 3, 12, 13

[21] Lei Ke, Mingqiao Ye, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, Fisher Yu, et al. Segment anything in high quality. *Advances in Neural Information Processing Systems*, 36, 2024. 5, 12

[22] Lei Ke, Mingqiao Ye, Martin Danelljan, Yu-Wing Tai, Chi-Keung Tang, Fisher Yu, et al. Segment anything in high quality. *Advances in Neural Information Processing Systems*, 36, 2024. 15

[23] Gihyun Kwon and Jong Chul Ye. Diffusion-based image translation using disentangled style and content representation. *arXiv preprint arXiv:2209.15264*, 2022. 2, 5, 12, 13

[24] Hyunsoo Lee, Minsoo Kang, and Bohyung Han. Conditional score guidance for text-driven image-to-image translation. *Advances in Neural Information Processing Systems*, 36, 2024. 2

[25] Yen-Liang Lin, Vlad I Morariu, Winston Hsu, and Larry S Davis. Jointly optimizing 3d model fitting and fine-grained classification. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV 13*, pages 466–480. Springer, 2014. 13

[26] Shilin Lu, Yanzhu Liu, and Adams Wai-Kin Kong. Tf-icon: Diffusion-based training-free cross-domain image composition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2294–2305, 2023. 2

[27] Grace Luo, Lisa Dunlap, Dong Huk Park, Aleksander Holynski, and Trevor Darrell. Diffusion hyperfeatures: Searching through time and space for semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 4, 5

[28] Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. Dragondiffusion: Enabling drag-style manipula-

tion on diffusion models. *arXiv preprint arXiv:2307.02421*, 2023. 2, 3, 5

[29] Chong Mou, Xintao Wang, Jiechong Song, Ying Shan, and Jian Zhang. Diffeditor: Boosting accuracy and flexibility on diffusion-based image editing. *arXiv preprint arXiv:2402.02583*, 2024. 1, 2, 5, 12

[30] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. *Advances in Neural Information Processing Systems*, 33:7198–7211, 2020. 2, 5, 12, 13

[31] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 2

[32] Or Patashnik, Daniel Garibi, Idan Azuri, Hadar Averbuch-Elor, and Daniel Cohen-Or. Localizing object-level shape variations with text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23051–23061, 2023. 2

[33] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020. 12, 14

[34] René Ranftl, Alexey Bochkovskiy, and Vladlen Koltun. Vision transformers for dense prediction. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12179–12188, 2021. 14

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 3, 12

[36] Michael Rubinstein, Armand Joulin, Johannes Kopf, and Ce Liu. Unsupervised joint object discovery and segmentation in internet images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1939–1946, 2013. 13

[37] Luming Tang, Menglin Jia, Qianqian Wang, Cheng Perng Phoo, and Bharath Hariharan. Emergent correspondence from image diffusion. *Advances in Neural Information Processing Systems*, 36:1363–1389, 2023. 2, 4, 5, 7

[38] Tatsunori Taniai, Sudipta N Sinha, and Yoichi Sato. Joint recovery of dense correspondence and cosegmentation in two images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4246–4255, 2016. 13

[39] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10748–10757, 2022. 2, 5, 6, 12, 13

[40] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1921–1930, 2023. 2

[41] Yunyang Xiong, Bala Varadarajan, Lemeng Wu, Xiaoyu Xiang, Fanyi Xiao, Chenchen Zhu, Xiaoliang Dai, Dilin Wang, Fei Sun, Forrest Iandola, et al. Efficientsam: Leveraged masked image pretraining for efficient segment anything. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16111–16121, 2024. 15

[42] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation. *Advances in Neural Information Processing Systems*, 35:38571–38584, 2022. 5, 12, 13

[43] Binxin Yang, Shuyang Gu, Bo Zhang, Ting Zhang, Xuejin Chen, Xiaoyan Sun, Dong Chen, and Fang Wen. Paint by example: Exemplar-based image editing with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18381–18391, 2023. 2

[44] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 5, 6, 12

[45] Hang Yu, Yufei Xu, Jing Zhang, Wei Zhao, Ziyu Guan, and Dacheng Tao. Ap-10k: A benchmark for animal pose estimation in the wild. *arXiv preprint arXiv:2108.12617*, 2021. 13

[46] Jiwen Yu, Yinhuai Wang, Chen Zhao, Bernard Ghanem, and Jian Zhang. Freedom: Training-free energy-guided conditional diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23174–23184, 2023. 2

[47] Junyi Zhang, Charles Herrmann, Junhwa Hur, Eric Chen, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. Telling left from right: Identifying geometry-aware semantic correspondence. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3076–3085, 2024. 2, 4, 5, 7, 13, 14

[48] Junyi Zhang, Charles Herrmann, Junhwa Hur, Luisa Polania Cabrera, Varun Jampani, Deqing Sun, and Ming-Hsuan Yang. A tale of two features: Stable diffusion complements dino for zero-shot semantic correspondence. *Advances in Neural Information Processing Systems*, 36, 2024. 2, 4, 5, 7, 13, 14

[49] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 2, 5

A. Implementation details

We apply the proposed methods to the text-to-image diffusion model, Stable Diffusion [35] using checkpoint v1.5. We begin by inverting real images with the edit-friendly DDPM inversion [20], sampling images with 100 denoising timesteps. To find semantic correspondence during transfer, we use the feature maps input to the self-attention layer. We set the denoising step t $\in$ [42, 100] and layer l $\in$ [2, 3] from the up-blocks of U-net to find correspondences and rearrange features. Additionally, we apply AdaIN at denoising step t $\in$ [82, 100] and use the off-the-shelf model SAM [21] to obtain object masks. And we measure dense correspondence at timestep 92 and layer 2. All of the experiments are conducted on an NVIDIA A6000 GPU and during the transfer experiments, the GPU memory usage amounted to about 15.17 GB.

B. Evaluation method details

B.1. Appearance similarity

To evaluate the success of transferring the appearance of the reference image, we conduct an experiment comparing the color histograms (A_{hist}) of the result image and the ground truth (GT) image. The comparison region is set by segmenting the object using SAM [21] for both the GT and result images. For the comparison of color histograms, we measure the Bhattacharyya distance as:

$$A_{\text{hist}}(H_G, H_O) = D_B(H_G, H_O) \tag{5}$$

where $D_B(H_G, H_O)$ is the Bhattacharyya distance between the color histograms of the masked GT image (H_G) and the masked transferred output image (H_O).

Additionally, we measure semantic similarity using CLIP score:

$$A_{\text{clip}}(G, O) = \frac{1}{N}\sum_{i=1}^{N} \text{CLIP}(G_i, O_i) \tag{6}$$

where G_i and O_i are the masked GT and masked transferred output images, respectively, and N is the total number of images.

The dataset used in the experiments is described in Appendix C.

B.2. Structure preservation

To evaluate the preservation of the target image's structure, we conduct a depth evaluation (I_{depth}), a mean Intersection over Union (mIoU, S_{miou}) and a key point evaluation (S_{key}).

For S_{depth}, we use an off-the-shelf depth estimation model [33]. We extract depth from the target image and the transferred results of each model, then measure the root mean square error (RMSE) at the object level:

$$S_{\text{depth}}(D_T, D_O) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(D_{T,i} - D_{O,i})^2} \tag{7}$$

where D_T and D_O are the depth maps of the masked target image and the transferred output image, respectively, and N is the total number of pixels.

For S_{miou}, we use SAM to obtain the masks of the ground truth (GT) and the transferred result objects. The mIoU is then measured at the object level as:

$$S_{\text{miou}}(T, O) = \frac{1}{N}\sum_{i=1}^{N}\frac{|M_{T,i} \cap M_{O,i}|}{|M_{T,i} \cup M_{O,i}|} \tag{8}$$

where T and O denote the target and output images, M represents the object mask obtained from SAM-HQ, and N is the total number of objects.

To follow the default settings of the models, ours, Cross-Image [1], DiffEditor [29], Splice VIT [39], IP-Adapter [44], and ZeST [8] are tested at an image resolution of 512^2. Swapping AE [30] and DiffuseIT [23] are tested at a resolution of 256^2.

For S_{key}, we assess structural preservation through pose estimation with ViTPose++ [42]. Following its approach, we evaluate AP-10K samples [AP-10K] from the training set and compute Average Precision (AP) using Object Keypoint Similarity (OKS) over thresholds $\tau \in [0.5, 0.95]$ with target keypoints as ground truth. Our method achieves higher AP than competitors, demonstrating superior structural retention. OKS is defined as:

$$\text{OKS} = \frac{\sum_i \exp\left(-\frac{d_i^2}{2s^2\kappa_i^2}\right)\delta(v_i > 0)}{\sum_i \delta(v_i > 0)}, \tag{9}$$

where d_i is the Euclidean distance between the detected and ground truth keypoints, s is the object scale, κ_i is a keypoint-specific constant, and v_i is the keypoint visibility.

Using OKS, the Average Precision (AP) score is computed as:

$$\text{AP} = \frac{1}{|\tau|}\sum_{\tau}\text{Precision}(\tau). \tag{10}$$

The precision at each threshold τ is given by:

$$\text{Precision}(\tau) = \frac{|\{\text{detected keypoints} \mid \text{OKS} \geq \tau\}|}{|\{\text{all detected keypoints}\}|}. \tag{11}$$

The dataset used in the experiments is described in Appendix C.

Figure S1. Examples of building and AFHQ for I_{hist}.

B.3. Dense correspondence

We evaluate dense correspondence using flow maps, which represent pixel displacements derived from the correspondences estimated by each method. These flow maps are computed by subtracting the difference between the target pixel coordinates from their corresponding matches. To measure deviations from the GT flow map, we calculate the L1 distance at the resolution of 512^2 as,

$$D_{\text{flow}}(F_{\text{pred}}, F_{\text{GT}}) = \frac{1}{N}\sum_{i=1}^{N}\frac{\sum|F_{\text{pred},i} - F_{\text{GT},i}|}{|\mathcal{M}_i|} \quad (12)$$

where N is the total number of images, $F_{\text{pred},i}$ and $F_{\text{GT},i}$ are the predicted and ground truth optical flow for image i, respectively, and $\mathcal{M}_i$ is the validity mask indicating the valid pixels in the flow.

SD-DINO [48] and Telling-Left-from-Right [47] employ both 960^2 and 840^2 image resolutions to extract feature descriptors across two distinct models, and ours utilizes a resolution of 512^2. Source and target images of varying sizes are resized to the input resolution required by each method, following the padding strategy detailed in the official implementation of SD-DINO [48]. Both ours and SD-DINO [48] compute dense correspondence by upsampling feature maps to 512^2. Telling-Left-from-Right [47] derives dense correspondence with feature maps at their original resolution (60^2), using a window-soft-argmax operation, and subsequently upsamples the correspondence map to 512^2. The dataset used in the experiments is described in Appendix C.

C. Evaluation dataset

For the quantitative evaluation, we used the AFHQ [9], AP-10K [45], and TSS [38] datasets, and a Building dataset collected from the Pexels[1]. This dataset will be publicly available. Especially, as shown in Fig. S1, to evaluate appearance transfer performance, we created datasets for A_{hist} and A_{clip} with the following setup: (1) Reference: original image (2) Target: shape and color-augmented image derived from the original image (3) Ground-Truth (GT): shape augmented image derived from the original image. We perform appearance transfer on 1) Reference to (2) Target, and measure the score by comparing the result object with (3) GT object. To align with the training domain of the pre-trained Swapping AE [30], we applied flip and weak warping as augmentations. Additionally, to evaluate structure preservation, we use a building dataset comprising 30 pairs of structure and target images, as well as an AFHQ dataset with 42 pairs. We evaluate dense correspondence on the TSS dataset [38], which includes dense correspondence flows and semantic masks for 400 image pairs sampled from the FG3DCAR [25], JODS [36], and PASCAL [16] datasets.

[1]https://www.pexels.com/

D. Baseline settings

All experiments were conducted at a resolution of 512^2, except for the Swapping AE and DiffuseIT, which were trained at a resolution of 256^2.

D.1. For appearance transfer comparison

Cross-Image. Cross-Image [1] employs edit-friendly DDPM inversion [20] for image inversion. Images are sampled with 100 denoising timesteps. And Cross-Image does not use an object mask during transfer, so the background of the target is not preserved after the transfer. The KV injection in self-attention occurs at t $\in$ [42, 100] and layer l $\in$ [2, 3] from the up-blocks. The contrast strength is set to 1.65, and the swap guidance scale is set to 3.5. Additionally, for consistency with our model, experiments were conducted using Stable Diffusion v1.5.

DiffEditor. DiffEditor is experimented with under Stable Diffusion v1.5. We use the standard DDIM scheduler for 50 denoising steps. The classifier-free guidance scale was set to the default value of 5. And Diffeditor uses an object mask during transfer, so the background of the target is preserved.

DiffuseIT. DiffuseIT [23] utilizes external models to guide the denoising process. We set the denoising timestep to 200, skipping the initial 80 timesteps, and use a resampling step of N=10 (resulting in a total of 130 iterations). Images are resized to a resolution of 224^2 to compute the ViT and CLIP losses, as these models only accept this resolution. These settings are the default configuration for image-guided manipulation as specified by the authors. Additionally, other configurations, including hyperparameters, follow the default settings provided by the authors. Since the provided checkpoint is trained at a resolution of 256^2, we also conducted experiments at this resolution.

Splice ViT. Splice ViT [39] employs a pre-trained DINO ViT model [42] as a feature extractor for optimizing the model on a single image pair. We use the 12-layer pre-trained ViT-B/8 model provided in the official DINO ViT implementation. For the ViT loss, images are resized to a resolution of 224^2. Keys are derived from the deepest attention module for self-similarity, and the output of the deepest layer is used to

extract the appearance from the target appearance image. We optimize using an input image pair with a resolution of 512^2 for 2000 iterations. These settings follow the default settings provided by the authors, and other configurations, including hyperparameters, also follow the provided configurations.

Swapping AE. We use the pretrained checkpoints provided on the official GitHub repository. We evaluate the AFHQ dataset and the LSUN Church pretrained models, treating the Building dataset as in-domain for LSUN Church model. In all evaluations, the target image is treated as the structure image, and the reference image is treated as the texture image. Additionally, we set the texture mixing alpha to 1.0, i.e,. simple texture swapping.

IP-Adapter. To account for target depth, we adopt the IP-Adapter + ControlNet model, using SDXL as the base model. The target image's depth map is extracted using off-the-shelf depth estimator [33], normalized, and then used as a condition for ControlNet. The reference image is provided as the input image. The ControlNet conditioning scale is set to 0.7, and the DDIM step is set to 30, following the inference settings from the official repository.

ZeST. ZeST utilizes Dense Prediction Transformers [34] for depth estimation and Rembg [13] for foreground extraction. It also employs Stable Diffusion XL Inpainting in conjunction with the corresponding version of depth-based ControlNet and IP-Adapter. Additionally, all other configurations, including hyperparameters, follow the default settings provided by the authors.

D.2. For semantic correspondence comparison

SD-DINO. SD-DINO [48] employs Stable Diffusion v1.5 with a diffusion model timestep of $t = 100$ as the visual descriptor, while integrating DINOv2 [6] as an auxiliary descriptor. Stable Diffusion features are extracted from the 2nd, 5th, and 8th layers of the U-Net decoder at timestep $t = 50$, while DINOv2 descriptors are derived from the token facet of its 11th layer. The input resolutions are 960^2 for Stable Diffusion and 840^2 for DINOv2, resulting in a feature map with a resolution of 60^2. Then, we use 512^2 upsampled feature map to find semantic correspondence.

Telling-Left-from-Right. Telling-Left-from-Right [47] adopts Stable Diffusion and DINOv2 features in a manner similar with SD-DINO. Furthermore, it incorporates the instance matching distance (IMD) to compare the target image with the horizontally flipped source image, thereby mitigating pose variation in paired images. Semantic correspondence is computed on the 60^2 resolution map using window soft argmax with a window size of 10, followed by upsampling to 512^2 for evaluation.

Figure S2. **Qualitative comparison between the results transferred using features after the self-attention layer and ours.** (a) Results transferred using features after the self-attention layer. (b) Results transferred using features before the self-attention layer (ours). (a) shows mismatched semantic correspondence, while (b) demonstrates accurate semantic correspondence.

Metrics	$A_{\text{hist}} \downarrow$	
Dataset	Building	AFHQ
After SA	0.478	0.581
Before SA(Ours)	**0.469**	**0.577**

Table S1. **Comparison of appearance similarity on different feature positions.** For all datasets, the appearance similarity of transferred results using features before the self-attention layer shows a lower I_{hist} compared to those transferred using features after the self-attention layer. We mark the best score in bold.

E. Ablation study for feature positions

We use the input features of the self-attention layer for correspondence measurement and feature injection. However, the output of the self-attention layer can also be used for correspondence measurement. Through experiments, we confirm that the input features to self-attention yield better performance. Fig. S2 (a), which uses the self-attention output features, shows less accurate matching compared to Fig. S2 (b), which uses the self-attention input features. And as shown in Tab. S1, the transferred results using input features better preserve the reference appearance compared to those using output features.

F. Ablation study for time steps

Our method measures dense correspondence at timestep 92. Because our method performs sparse key-point matching in the early steps and dense matching in the later steps. As shown in Tab. S2, the flow map distance is lower in the later steps compared to the early steps. It demonstrates that dense correspondence is more effectively measured in the later steps than in the early ones.

Time Step \ Dataset	$D_{\text{flow}} \downarrow$ FG3D CAR	JODS	PASCAL
62	10.75	32.86	28.37
77	9.71	30.16	24.72
92(Ours)	**9.43**	**28.75**	**21.83**

Table S2. **Comparison of dense correspondence on different time steps.** For all datasets, the dense correspondence measured at later time step shows a lower flow map distance compared to that measured at mid time step. We mark the best score in bold.

Method	$S_{\text{miou}} \uparrow$		$A_{\text{hist}} \downarrow$	
Component \Dataset	Building	AFHQ	Building	AFHQ
Baseline(KV injection)	0.833	0.926	0.495	0.603
+Feature rearrange	0.942 **(+0.109)**	0.968 **(+0.038)**	0.484 (-0.009)	0.582 **(-0.021)**
+ AdaIN(Ours)	0.939 (-0.003)	0.972 (+0.004)	0.469 **(-0.015)**	0.577 (-0.005)

Table S3. **Quantitative ablation results.** We mark the greatest difference in scores between the components in bold.

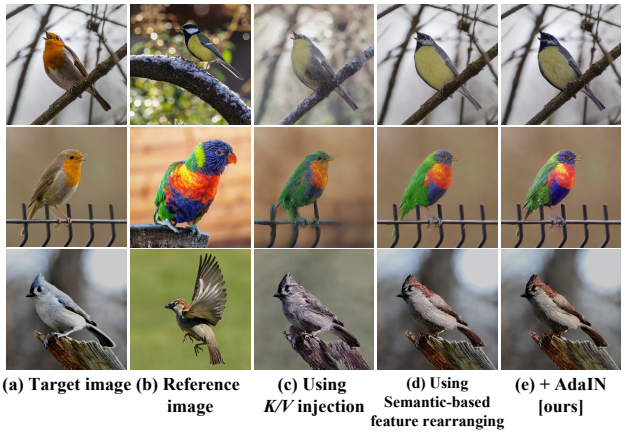

Figure S3. Additional samples of ablation study.

G. Quantitative ablation results for each component

We add the below table to provide quantitative ablation for Fig. H. Feature rearrange is our core component and mainly helps structure S_{miou}). AdaIN adjusts color distribution (A_{hist}).

H. Additional examples on ablation study

We present additional samples from the ablation study analyzing the effects of each component of our model in Fig. S3.

I. Ablation study for object mask

This section evaluates the role and effectiveness of object masks in appearance transfer tasks. Tab. S4 summarizes the approaches for obtaining object masks adopted by Ours and each baseline. Ours, DiffEditor, and ZeST utilize object masks during the transfer process, while other competitors do not incorporate object masks in their transfer processes.

	Ours	DiffEditor	ZeST	Others
For mask	SAM-HQ [22]	EfficientSAM [41]	Rembg [13]	X

Table S4. **Approaches for obtaining object masks.** The table compares the approaches used to obtain object masks in Ours, DiffEditor, and ZeST. Others refer to other baselines, including Cross-Image, DiffEditor, DiffuseIT, SpliceViT, Swapping AE, and IP-Adapter. These baselines do not utilize object masks.

Metrics	$A_{\text{hist}} \downarrow$		$S_{\text{miou}} \uparrow$	
Dataset	Building	AFHQ	Building	AFHQ
Ours *	0.469	0.577	0.939	0.972
Ours *w/o mask*	0.467	0.579	0.858	0.943
Cross-Image	0.491	0.608	0.758	0.915
DiffEditor *	0.478	0.608	0.863	0.943
DiffuseIT	0.477	0.607	0.855	0.951
Splice ViT	0.472	0.580	0.842	0.943
Swapping AE	0.481	0.629	0.821	0.942
IP-Adapter	0.487	0.616	0.642	0.950
ZeST *	0.497	0.602	0.925	0.980

Table S5. **Comparison of appearance similarity and structure preservation for our model without a mask.** Ours *w/o mask* refers to our method without using an object mask. * indicates a model using a mask. We mark the best score in red and the second-best score in yellow.

Figure S4. **Qualitative comparison between our model without an object mask and the our model.** (a), which does not apply the object mask during transfer, fails to preserve the background of the target image, whereas (b), with the mask applied, successfully retains the background.

To analyze the impact of object masks, we conduct experiments with our method without using an object mask. As shown in Tab. S5, the performance of Ours *w/o mask* decreases in structure preservation compared to ZeST and DiffEditor, which use object masks. This result demonstrates that object masks are effective in maintaining the structure of the target object. Among competitors that do not use object masks, Ours *w/o mask* achieves the best structure preservation. Regarding appearance similarity, our model maintains strong performance even without a mask, owing to its semantic matching capability during the transfer process.

Fig. S4 illustrates the appearance transfer results without using an object mask. Without an object mask, the back-

	Matching level	Matching rule	Rearrange target
Ours	Feature map	One-to-One	Feature map
Case1	Feature map	One-to-Many	Feature map
Case2	Query, Key	One-to-Many	Query

(a) Matching levels, Rules, and Targets by Case

(b) Comparison between Cases and Ours

Figure S5. Comparison of matching component combinations.

ground of the target image is not preserved after the transfer. This observation highlights that object masks ensure object-aware appearance transfer. Competitors that do not use object masks, such as Cross-Image, DiffEditor, DiffuseIT, SpliceViT, Swapping AE, and IP-Adapter, fail to preserve the background.

J. Ablation study for matching rule

Rationale: As we aim to transfer appearances according to semantic matches (e.g., beak-to-beak), it is natural to employ one-to-one winner-takes-all matches rather than softmax aggregation.

In Case 1, implicit alignments like softmax aggregation fail to preserve reference feature values. And in Case 2, the injection based on the matching between the query and key with the attention mechanism also produces similar failure results. There are no scenarios where one-to-many or many-to-one matching outperforms one-to-one. Features from similar regions inherently share similar values, so there are no cases where top-1 similarity is incorrect while top-2 to N is correct. If cosine similarity fails in one-to-one matching, cosine similarity-based attention mechanisms would also fail.

K. User study

Method	Ours	Cross-Image	Diffeditor	DiffuseIT	Splice VIT	IP-Adapter	Zest
U_{app}	**0.661**	0.059	0.033	0.026	0.096	0.062	0.064
U_{str}	**0.462**	0.014	0.042	0.121	0.030	0.062	0.276

Table S6. User study results. The bold is the best score.

We conducted a user study with 53 participants, evaluating 15 randomly selected samples for appearance similarity (U_{app}) and structure preservation (U_{str}).

L. Additional qualitative results

In Fig. S6 and Fig. S7, we provide more additional qualitative comparisons with competitors. In particular, Fig. S6 illustrates the results when the reference and target are aligned but the reference object has complex patterns, as well as when the reference and target are unaligned. And Fig. S8 and Fig. S9 showcase our transferred results across various domains. Additionally, Fig. S10 shows the results of appearance transfer from each object from two different reference images to multiple objects in a single target image. Each appearance transfer process occurs simultaneously.

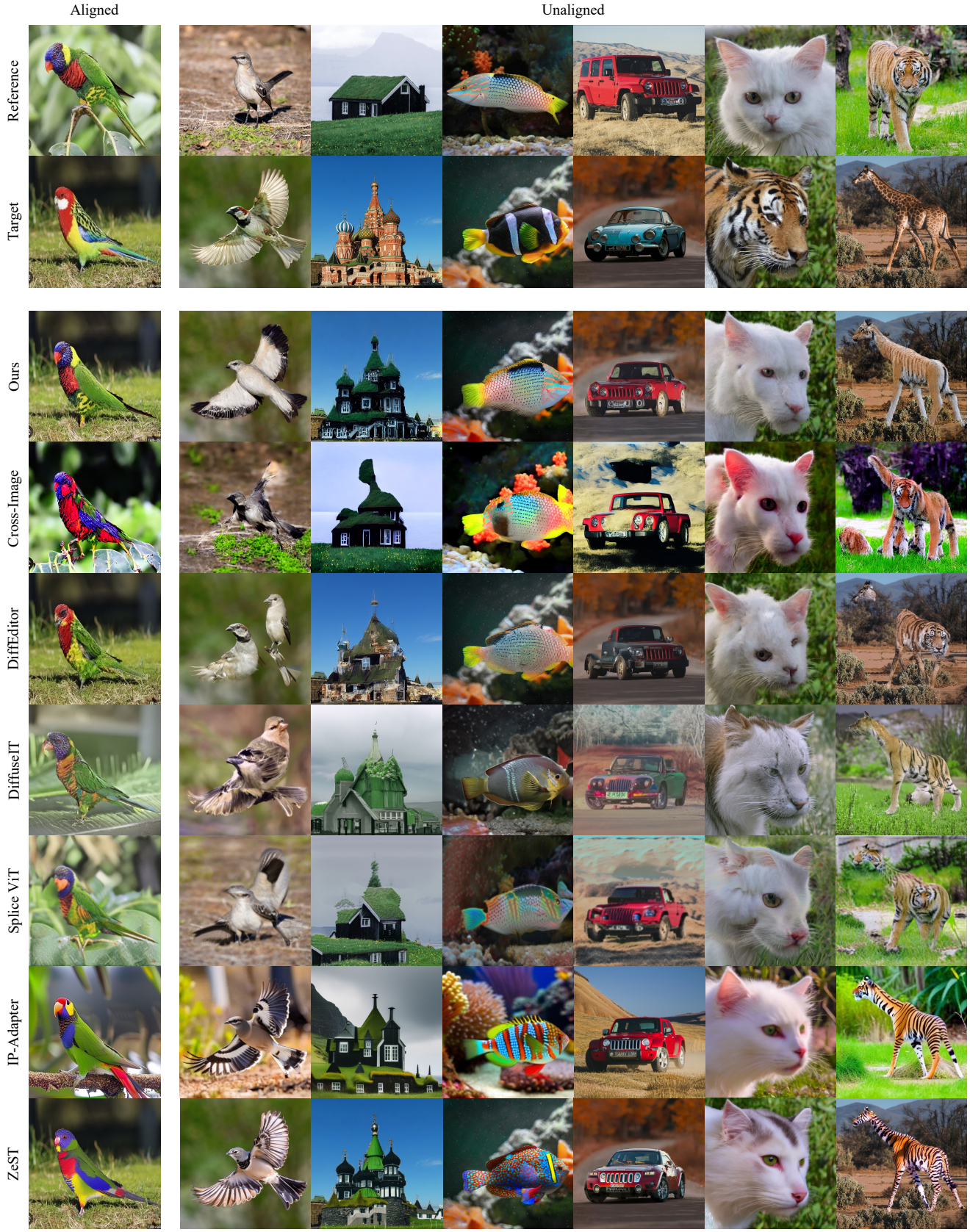

Figure S6. Our results on samples where the reference and target are aligned but the reference has complex patterns, as well as on various samples where the reference and target are misaligned.

Figure S7. Qualitative comparison of appearance transfer for bird samples.

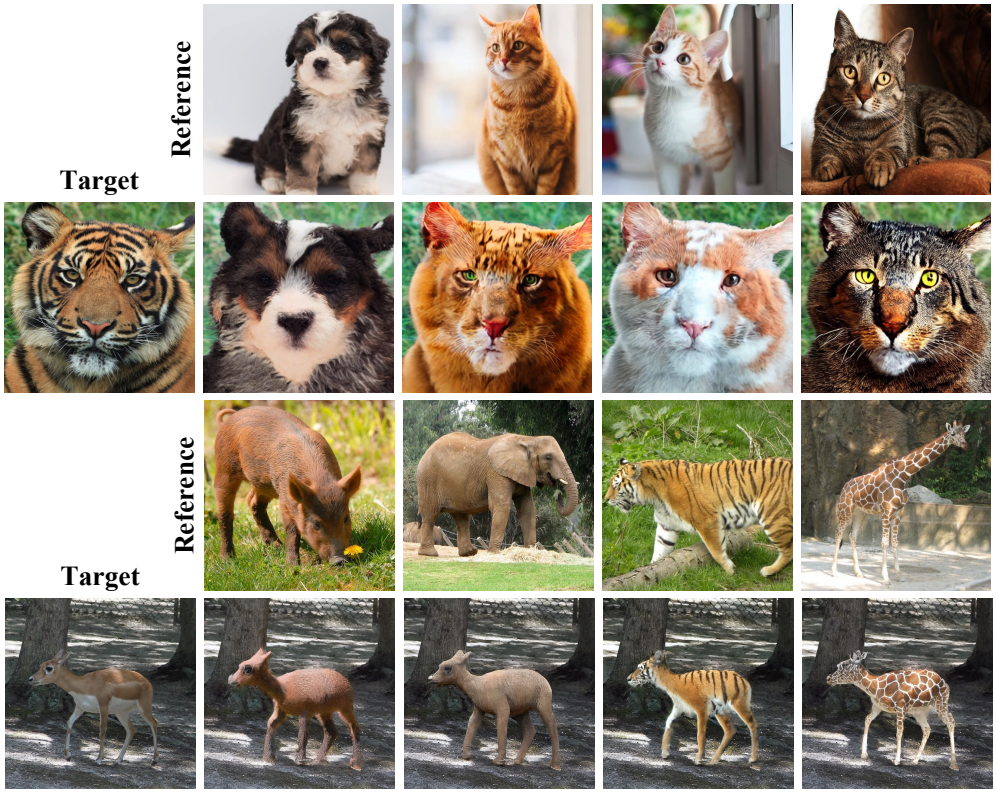

Figure S8. Our results on samples where the reference and target differ in size or are misaligned.

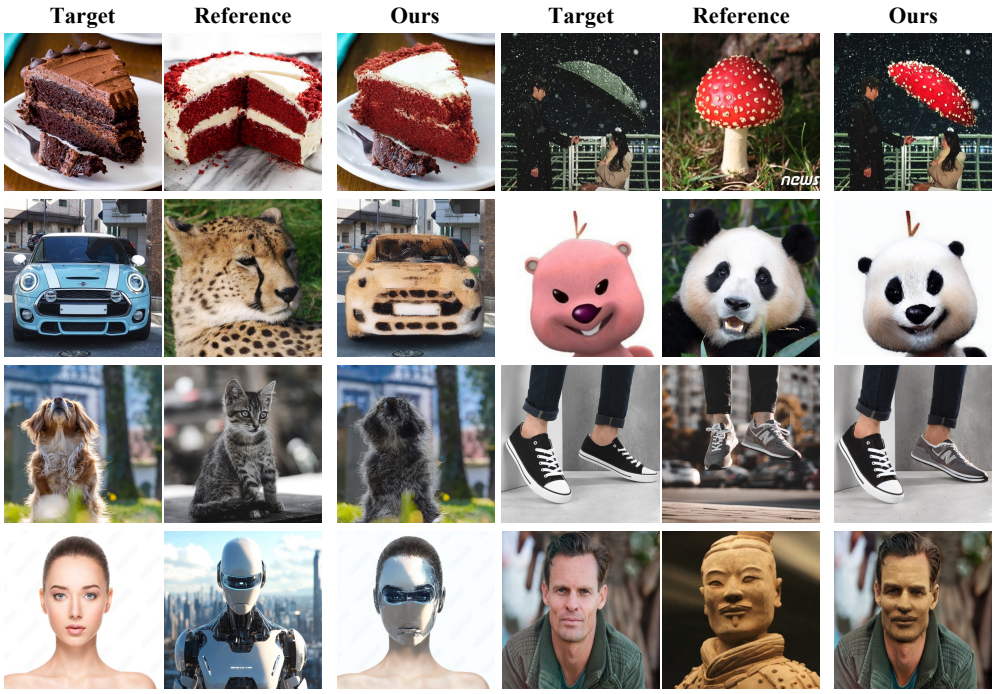

Figure S9. Our results of various domain.

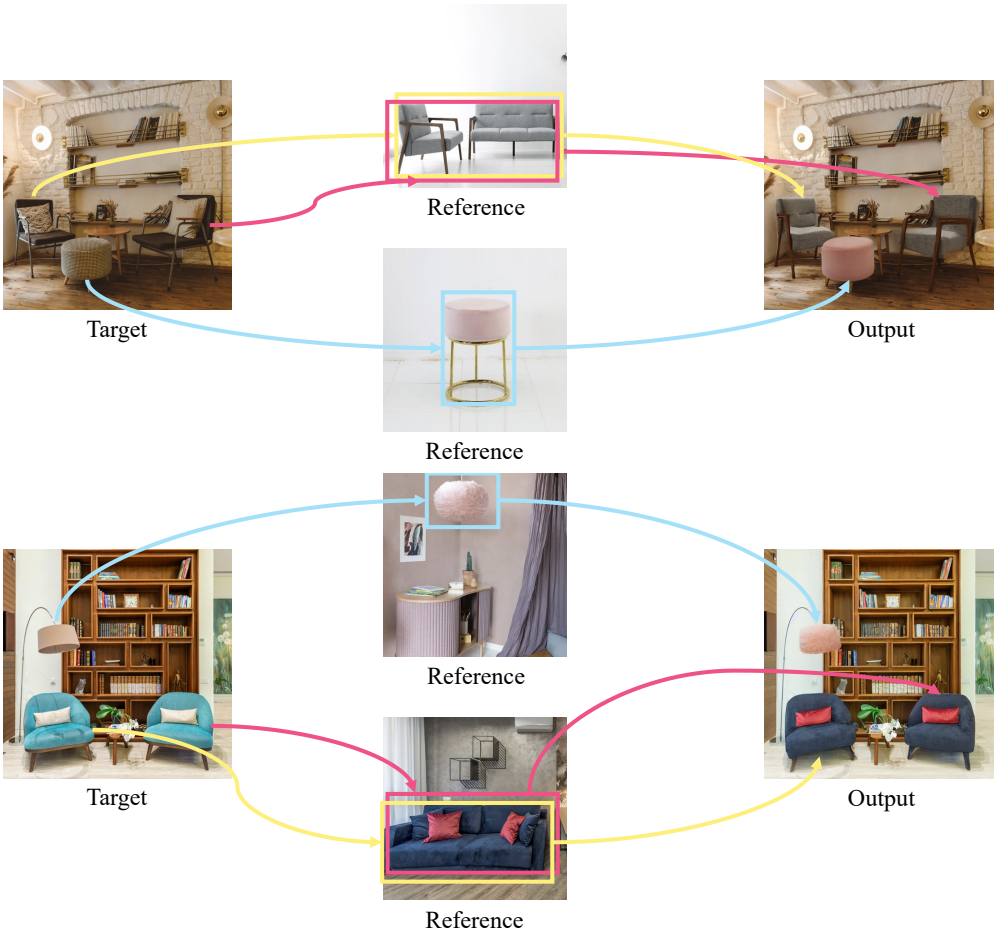

Figure S10. Results of appearance transfer between multiple objects.