Edit One for All: Interactive Batch Image Editing
— Supplementary Materials —

Thao Nguyen Utkarsh Ojha Yuheng Li Haotian Liu Yong Jae Lee
University of Wisconsin–Madison
https://thaoshibe.github.io/edit-one-for-all

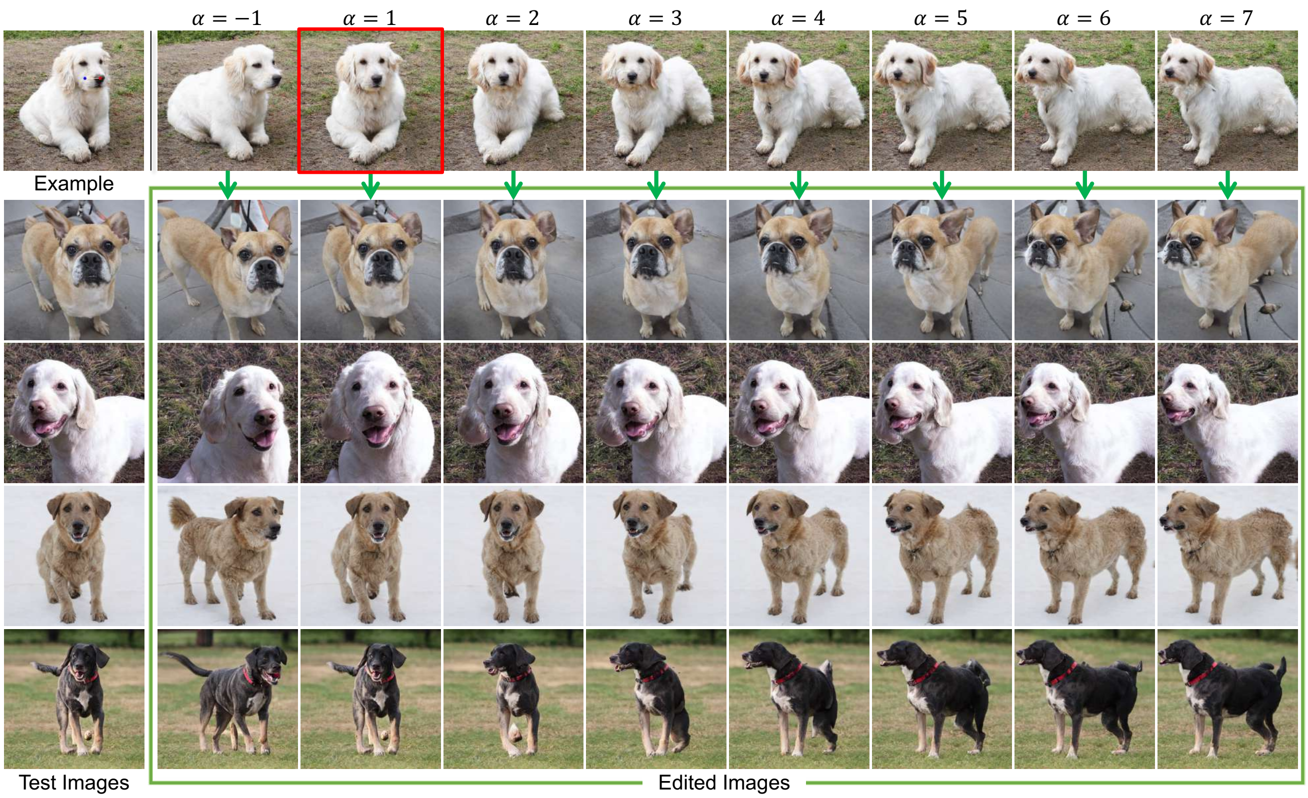

Figure 1. **Interactive Batch Image Editing**. As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (dog pose). (Red bounding box indicates the Δ_w yielded by dragging points).

This document provides additional qualitative results that could not be included in the main paper due to page limits. We present results from Interactive Batch Image Editing in Section 1, where adjustments in the example are automatically updated to test images. Subsequently, we demonstrate the effectiveness of our method when multiple edits are applied to an example before reaching the final state in Section 2. Limitations are discussed in Section 3, followed by additional qualitative results in Section 4 and implementation details in Section 5.

1. Interactive Batch Image Editing

We demonstrate the advantages of computing adaptive α values for individual example images. Consider a scenario where the editing objective is to rotate and bring into frontal view a set of n faces. User will first annotate dragging points for example image to rotate this face to frontal (Fig. 2, 1st row, $\alpha = 1$). Upon completion, each test images will also become frontal, each with unique editing strengths represented by $\alpha_1, \alpha_2, ..., \alpha_n$, computed using $\alpha_i = (w'_0 - w_i) \cdot n$ with $n = \Delta^*_w / ||\Delta^*_w||$ (Fig. 2, row 2-5, 5th column).

Now, if the user wishes to have the same faces facing slightly to the left, there is no need to re-annotate the orig-

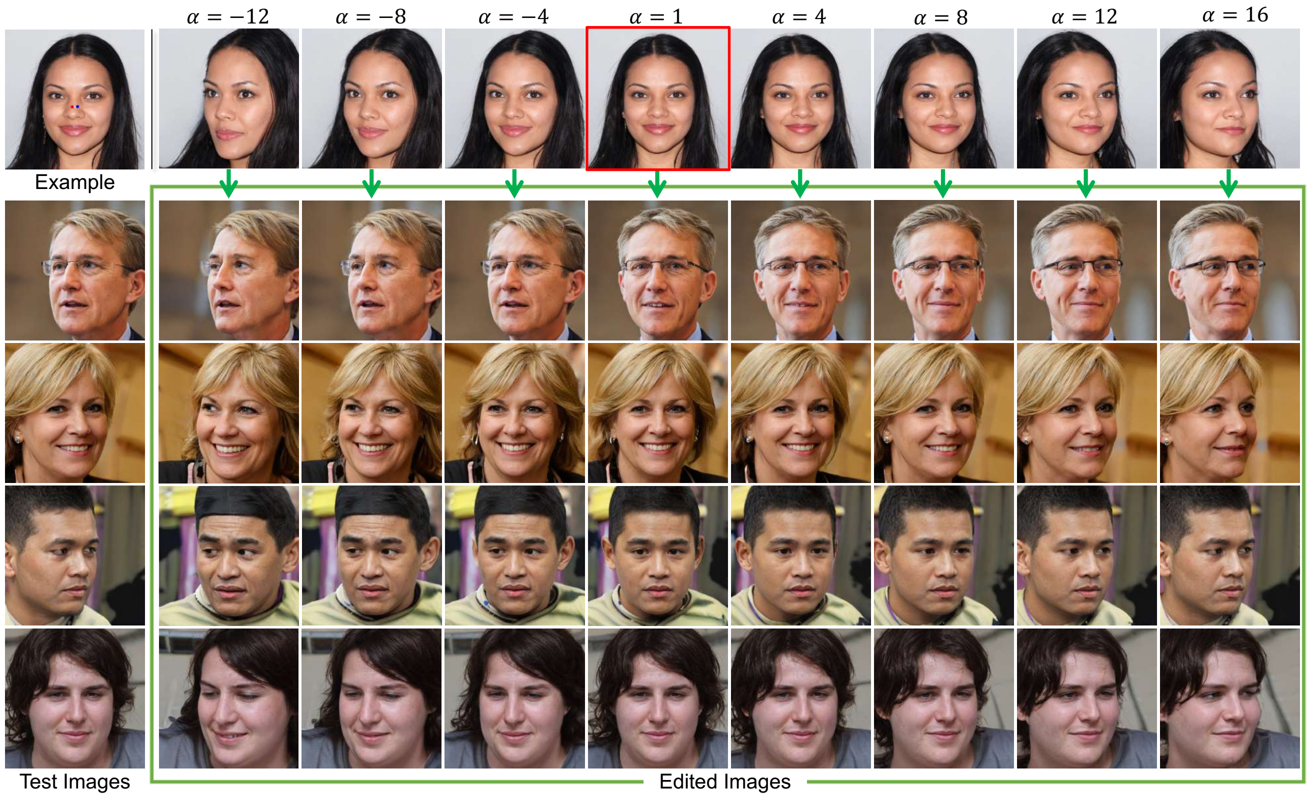

Figure 2. **Interactive Batch Image Editing**. As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (yaw degree). (Red bounding box indicates the Δ_w yielded by dragging points).

inal training examples and re-run the optimization process. In an interactive fashion, user can simply adjust the scaling of the α for the specific training example to achieve the desired edit (e.g., $\alpha = -4$), and all other α values will be automatically recalculated. Qualitative results are shown in Fig. 2. As can be seen, all poses (yaw degrees) in test images are changing accordingly to the example image (yaw degree is consistent). It is worth noting that it takes roughly 0.05s to compute a new image (about 0.03s to recompute α for each test image and 0.02s to generate a new image). Thus, it is fast enough for real-time batch image editing.

Additional qualitative results for interactive batch image editing are provided in Fig. 1 (dog pose rotation), Fig. 5 (mountain enlargement/removal), Fig. 6 (face slimming/enlargement), Fig. 7 (anime hair shortening/lengthening), Fig. 8 (tiger roaring) and Fig. 9 (model leg pose).

2. Multiple Edits

Multiple edits can be applied to the example to reach the final state, which is then transferred to the test images without intermediate steps. For example, a user may choose to initially rotate the face to the left (1st edit) and subsequently enhance the person's smile (2nd edit). The resulting final state incorporates both edits. Our method can directly transfer the final state to test images, including both edits (face rotation and smiling) (Fig. 3, columns 1-3).

Two additional examples of multiple edits are presented in Fig. 3. Columns 4-6 showcase uplifting the dog's ear (1st edit) and then opening the mouth (2nd edit). Columns 7-9 illustrate shortening the dress (1st edit) and adjusting the pose (2nd edit). Despite variations in the initial states of the test images (e.g., variations in mouth openness), our methods ensure consistency in the final states across test images (e.g., all dogs have an open mouth and uplifted ears).

3. Limitations

While our method has demonstrated effectiveness in various applications, there are some limitations/failure cases that we notice. (1) Failure to capture small details: Our method may encounter challenges in accurately transferring small details (e.g., curling elephant trunk, Fig. 4a). (2) Semantic dissimilarity in example/test images: When test images deviate significantly in semantic content from the example (e.g., vastly differing in hair lengths), the transferred results may exhibit suboptimal performance (Fig. 4b). (3) Poten-

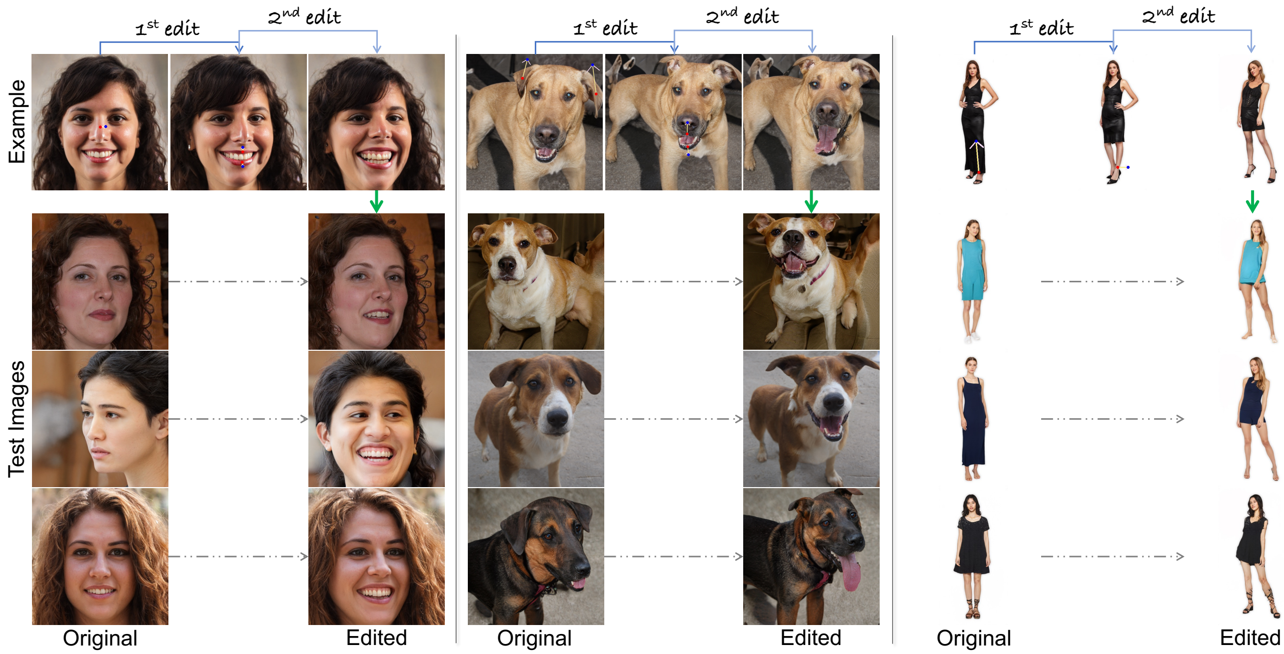

Figure 3. **Multiple edits** can be applied to example image before being transferred to test images.

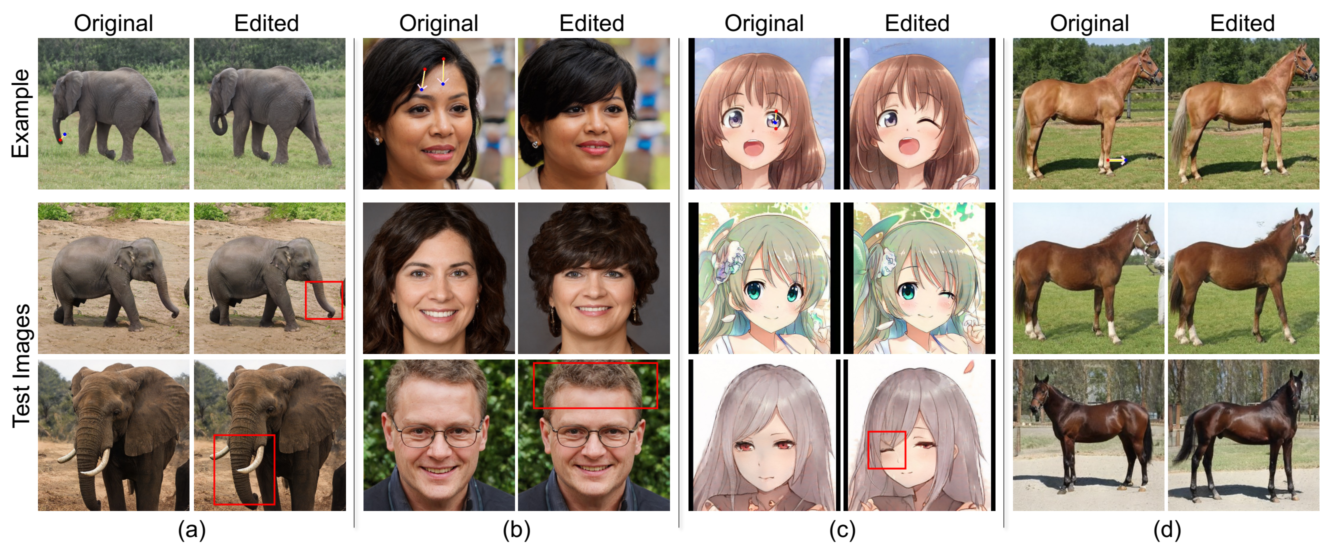

Figure 4. **Limitations.** (a) Failure Case: Our method may encounter challenges in capturing fine details (e.g., curling trunk of an elephant). (b) Example-Test Similarity: For optimal results, the example and test images should belong to the same semantic domain (e.g., both featuring long hair) to ensure correctly transferred edits. (c-d) Interesting Cases: Edits can be mistakenly interpreted, resulting in unexpected outcomes such as winking in the wrong eye (c) or unintentionally flipping the horse (d).

tial editing errors: In certain scenarios, the editing might be misinterpreted and lead to unexpected outcomes. For example, attempts to make a person wink at the left eye may occasionally lead to the wink being transferred to the right eye (Fig. 4c). Another example is pose changes, when adjusting the pose of horse legs, there are instances where the outcome unexpectedly mirrors the horse in the same position as the example image (Fig. 4d).

4. Additional Qualitative Results

Along with the domains presented in the main paper (Human faces (FFHQ) [1], Lions, Dogs [2], MetFaces [3], Human bodies [4]), we provide additional qualitative results for Cats [2], and Horses [5] domains in Fig. 11, 10 respec-

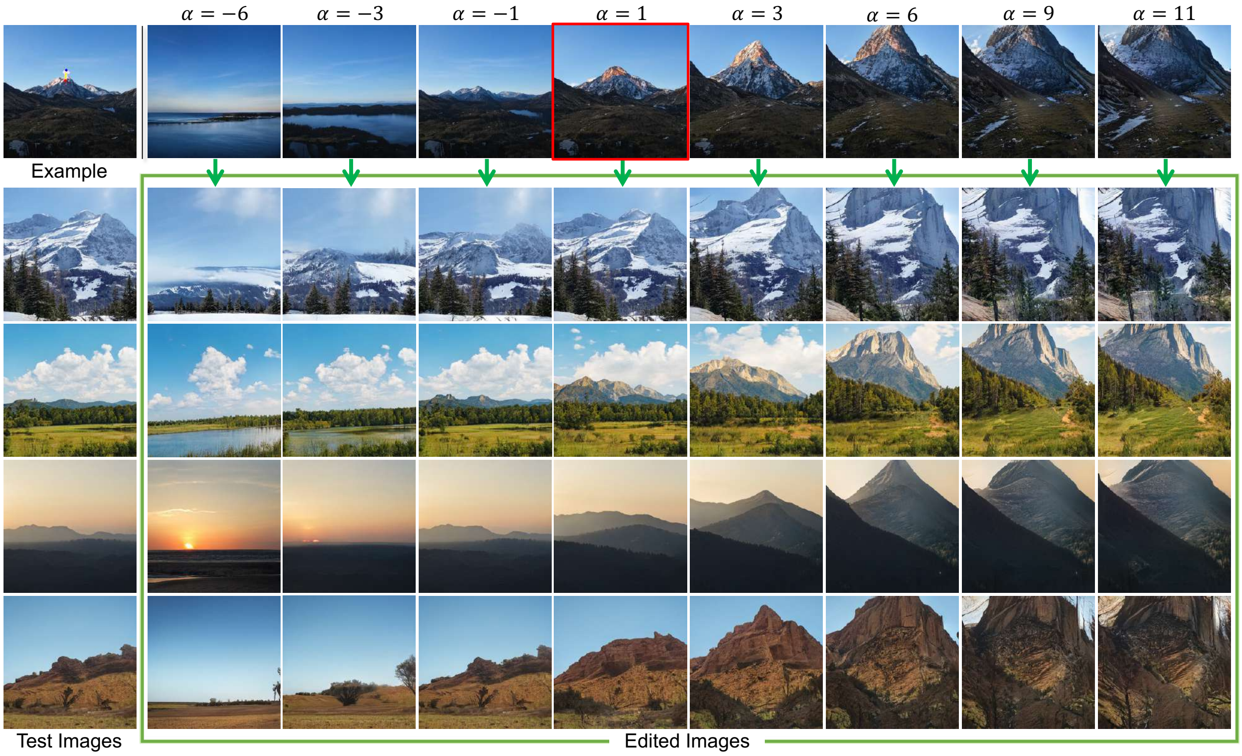

Figure 5. **Interactive Batch Image Editing.** As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (mountain height). (Red bounding box indicates the Δ_w yielded by dragging points).

tively.

5. Implementation Details

We use the AdamW optimizer [6] to optimize Δ_w^* for 1000 iterations, with a learning rate of 0.001 and a batch size of 16. All experiments are performed on a single NVIDIA RTX 3090 machine.

References

[1] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. 2019. 3

[2] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha. Starganv2: Diverse image synthesis for multiple domains. In *CVPR*, 2020. 3

[3] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data, 2020. 3

[4] Jianglin Fu, Shikai Li, Yuming Jiang, Kwan-Yee Lin, Chen Qian, Chen-Change Loy, Wayne Wu, and Ziwei Liu. Stylegan-human: A data-centric odyssey of human generation. *arXiv preprint*, arXiv:2204.11823, 2022. 3

[5] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 3

[6] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 4

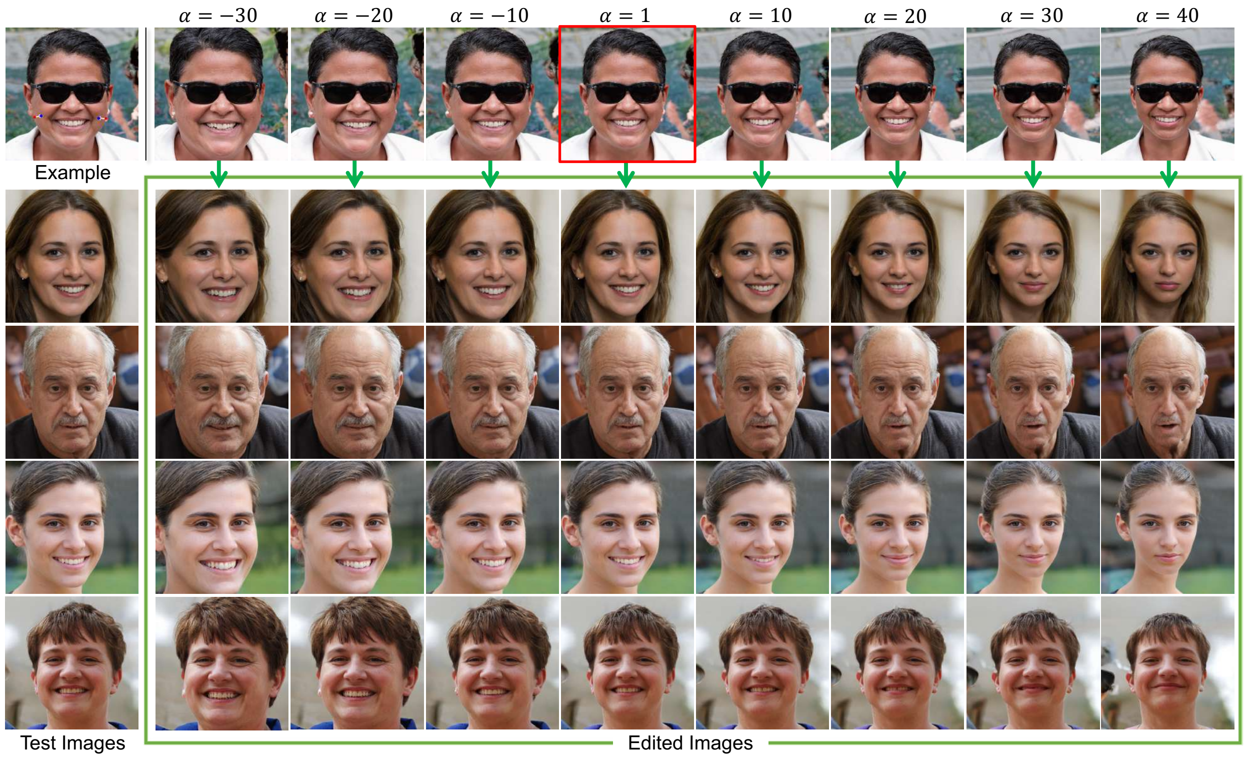

Figure 6. **Interactive Batch Image Editing.** As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (facial shape). (Red bounding box indicates the Δ_w yielded by dragging points).

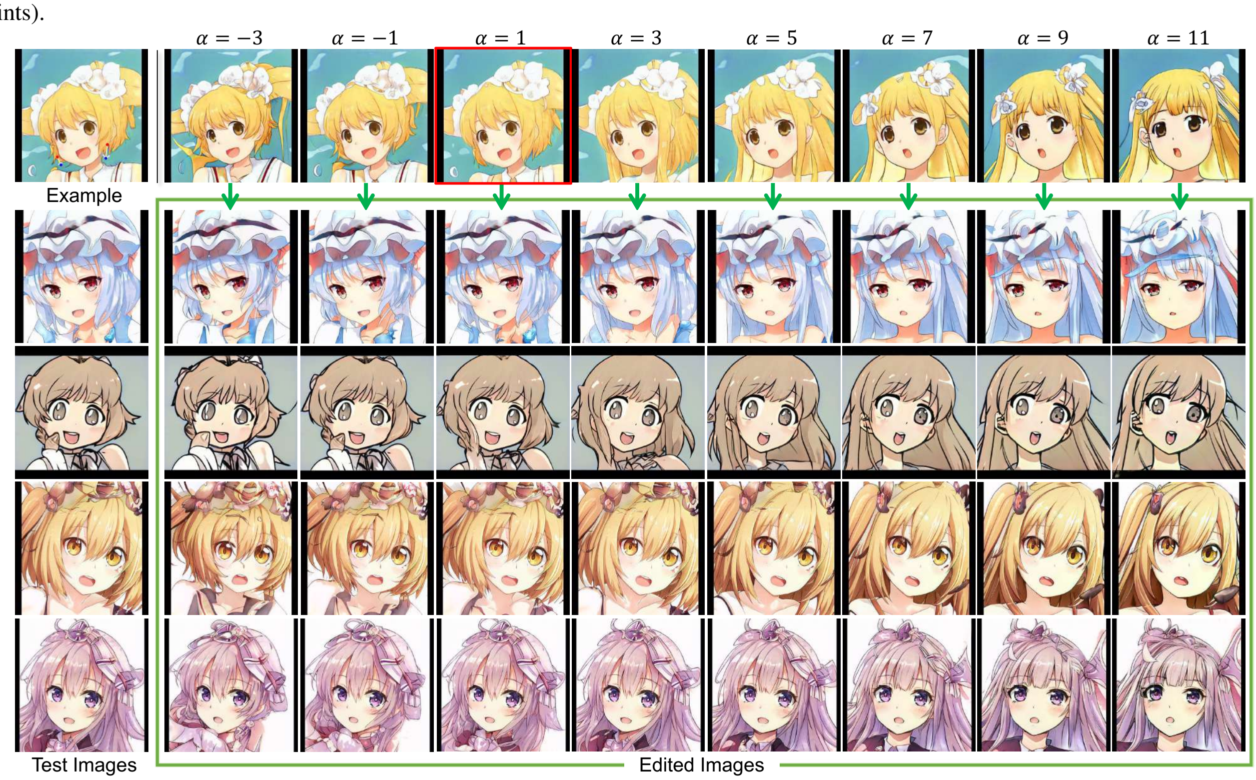

Figure 7. **Interactive Batch Image Editing**. As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (hair length). (Red bounding box indicates the Δ_w yielded by dragging points).

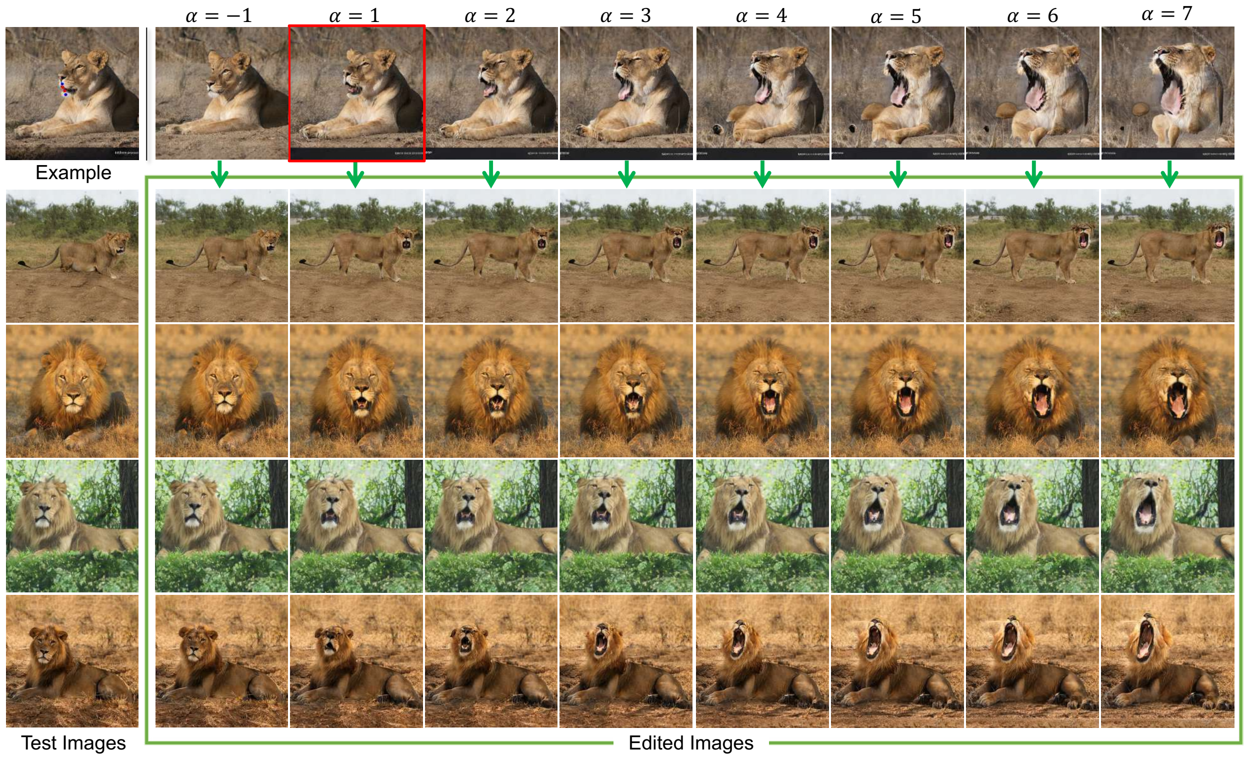

Figure 8. **Interactive Batch Image Editing**. As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (roar degree). (Red bounding box indicates the Δ_w yielded by dragging points).

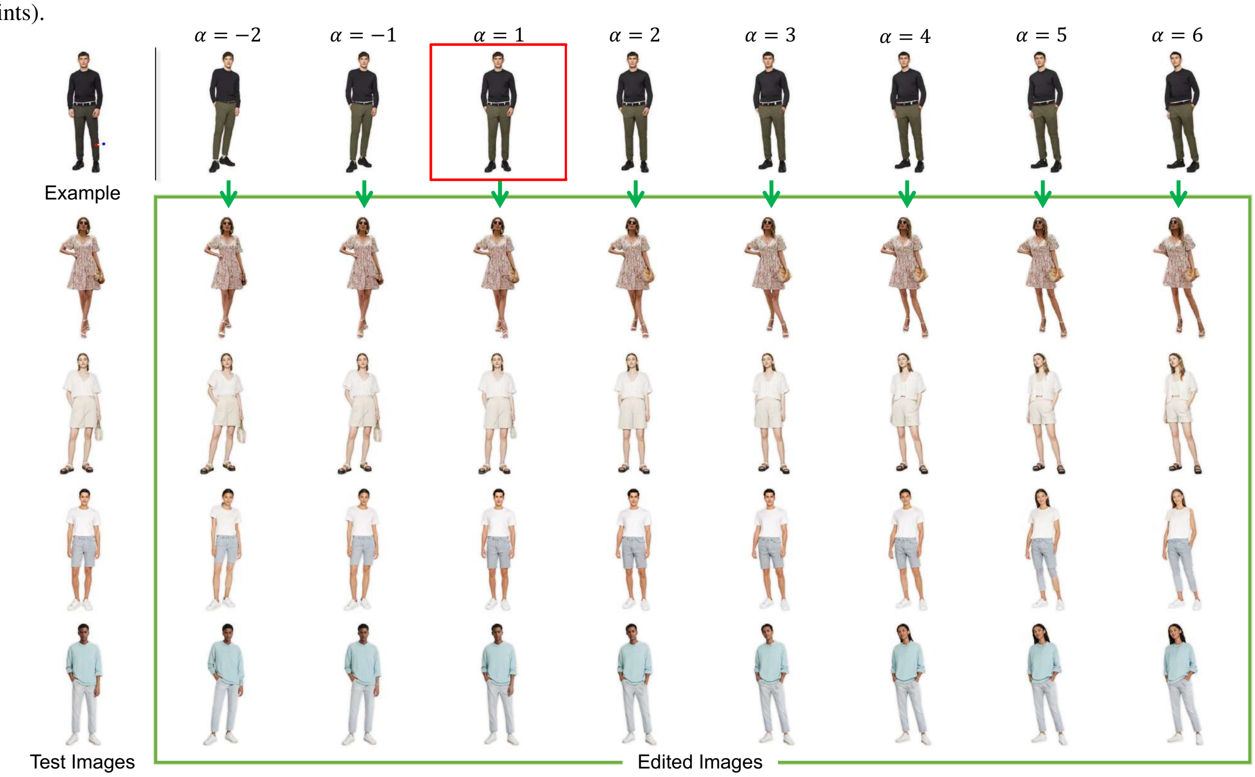

Figure 9. **Interactive Batch Image Editing**. As the user adjusts the editing strength α in the example image (first row), all test images will be automatically updated, ensuring consistency in the final state (leg position). (Red bounding box indicates the Δ_w yielded by dragging points).

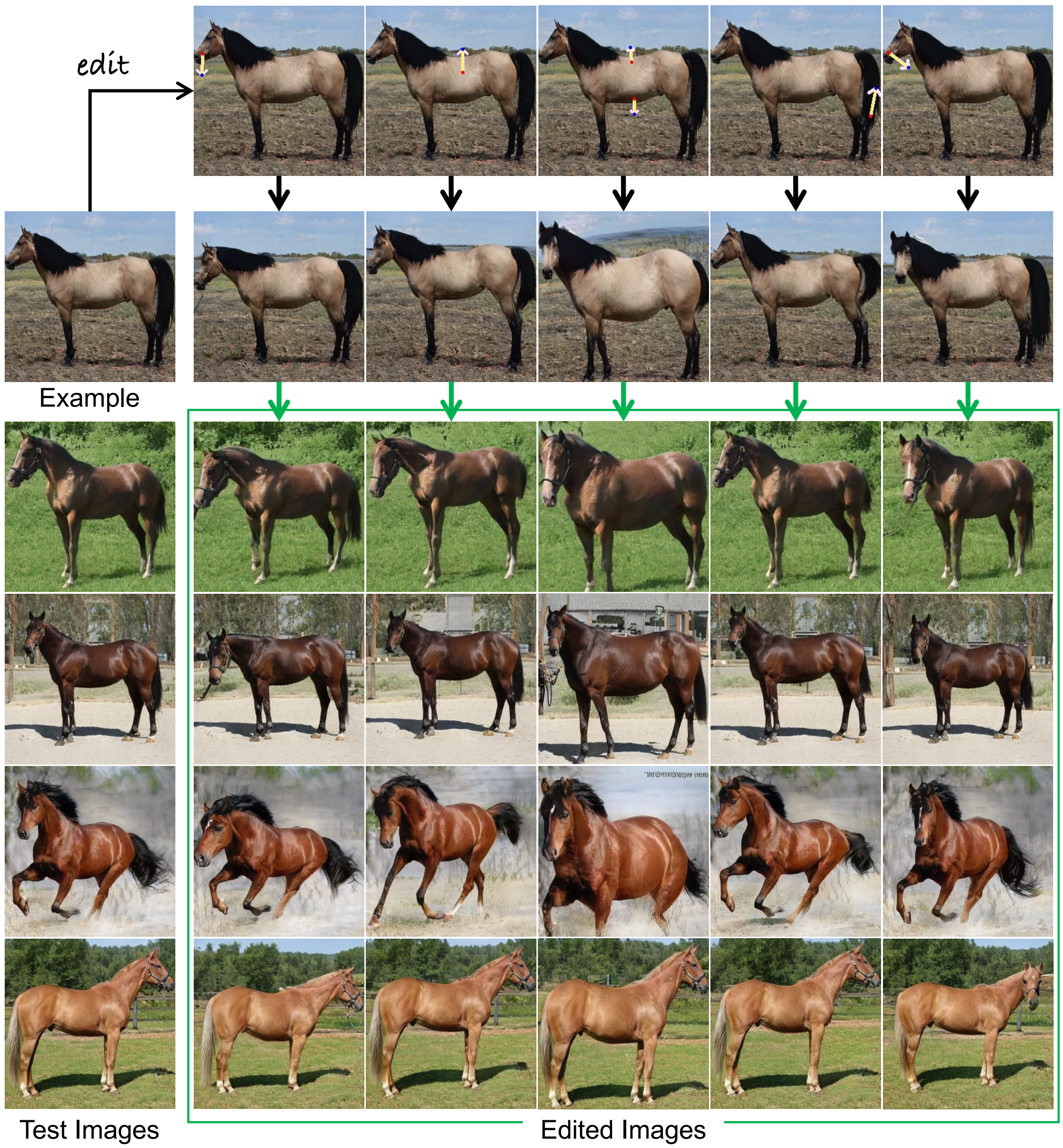

Figure 10. Additional qualitative results on Horses.

Figure 11. Additional qualitative results on Cats.