S1. EmoPair Dataset

In response to the absence of an image dataset specifically designed for emotion-evoke image generation, we introduce EmoPair, comprising two distinct subsets: the EmoPair-Annotated Subset (EPAS), encompassing 331,595 image pairs sourced from Ip2p [1] and annotated with emotion labels; and the EmoPair-Generated Subset (EPGS), featuring 6,949 pairs generated through text instructions specifying target emotions.

For the EmoPair-Annotated Subset (EPAS), we use $\mathcal{P}$ to categorize the source and target images from the Ip2p dataset into eight different emotions. We remove samples where the emotional labels of the source and target images are consistent, then use the emotion labels to augment the Ip2p dataset, generating 331,595 image pairs.

To **amusement**	Add colorful, flying balloons in the sky.
To **awe**	Turn the background into mountains and rivers.
To **contentment**	Change the setting to a tranquil outdoor scene.
To **excitement**	Add some exciting fireworks in the background.
To **anger**	Set the background on fire.
To **disgust**	Add a dirty spider web to the picture.
To **fear**	Twist and stretch figures into grotesque forms.
To **sadness**	Add a tombstone.

Figure S1. **Instruction examples of EmoPair-Generated Subset (EPGS).** For each emotion category, we retained the top ten text instructions based on the rankings determined by human annotators according to the efficacy of each emotion category. We utilized these instructions for Ip2p in image editing to generate target images capable of evoking the desired emotions, thereby constructing our EPGS.

For the EmoPair-Generated Subset (EPGS), we formulated 50 general instructions that are agnostic to source images, prompting transitions to desired emotions across 8 categories using GPT-3 [2]. Human annotators then ranked these instructions based on efficacy within each emotion category, ultimately retaining the top ten. Figure S1 illustrates examples of these instructions.

We employ the following selection criteria to control the quality of generated image pairs of EPGS: (1) Using our emotion predictor $\mathcal{P}$, we analyze the generated images and only select those with a Top-1 classification confidence over 90% for the target emotions. (2) To ensure the preservation of similar scene structures, we utilize Structural Similarity Index (SSIM) [12] and Learned Perceptual Image Patch Similarity (LPIPS) [16] for filtering the remaining outcomes. SSIM measures structural similarity between images, while LPIPS quantifies perceptual differences. Specifically, we require the generated images $\hat{x}$ and the source image x to meet the conditions below: $0.3 < SSIM(x, \hat{x}) < 0.6$ and $LPIPS(x, \hat{x}) > 0.1$. Ultimately, EPGS retains 6,949 image pairs.

To ensure the quality of EPGS, we have human annotators re-annotate 300 images. The annotators view both the source and edited images, and we ask them to select the one that better evokes the target emotion. Since we need the target image to evoke the desired emotion more effectively than the source image, if the annotators choose the source image, we swap the source and target images in the original dataset to create a new subset. Each image is reviewed by three annotators, and the final subset is determined by majority vote. We then use this subset to fine-tune our EmoEditor.

To provide a comprehensive overview of our dataset, additional image pair examples are presented in Figure S2 and S3.

S2. Experiment

S2.1. Emotion Encoder

Figure S4 shows the network structure of our emotion encoder τ_θ. We represent the target emotion as a one-hot encoding e_{oh} and use the emotion predictor $\mathcal{P}$ to calculate the emotional state of the source image e_s. Then, we compute the emotion editing direction e_{dir} by calculating the difference between e_{oh} and e_s. Our Emotion Encoder τ_θ is a fully connected network designed for transforming emotion editing direction e_{dir} into the structured emotion embedding e. The architecture of the model is composed of a series of fully connected layers that progressively increase the dimensionality of the input.

The network begins with an input layer of size 8, which is then passed through a sequence of fully connected layers with increasing sizes: 256, 512, and 768 neurons, respectively. Each of these layers is followed by a ReLU activation function to introduce non-linearity and improve the model's learning capacity. The final linear layer projects the output to a dimension of 77×768, effectively structuring the embedding to fit the input size of our denoising network ϵ_θ.

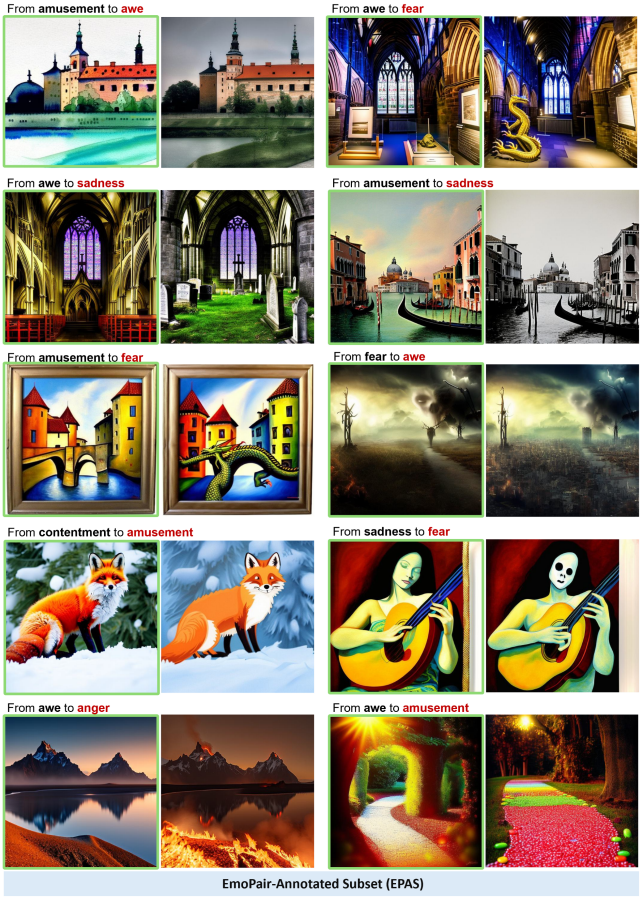


Figure S2. **Sample of EmoPair-Annotated Subset (EPAS).** On the left side of each pair of images is the source image (framed in green), while the right side shows the target image. The emotion labels for the source and target images (highlighted in red) are indicated above the images.

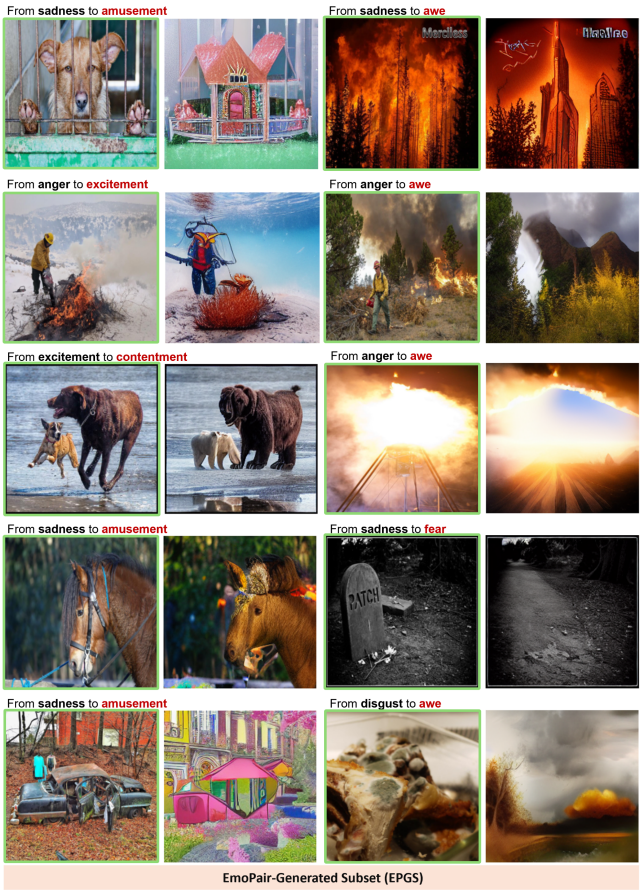


Figure S3. **Sample of EmoPair-Generated Subset (EPGS).** On the left side of each pair of images is the source image (framed in green), while the right side shows the target image. The emotion labels for the source and target images (highlighted in red) are indicated above the images.

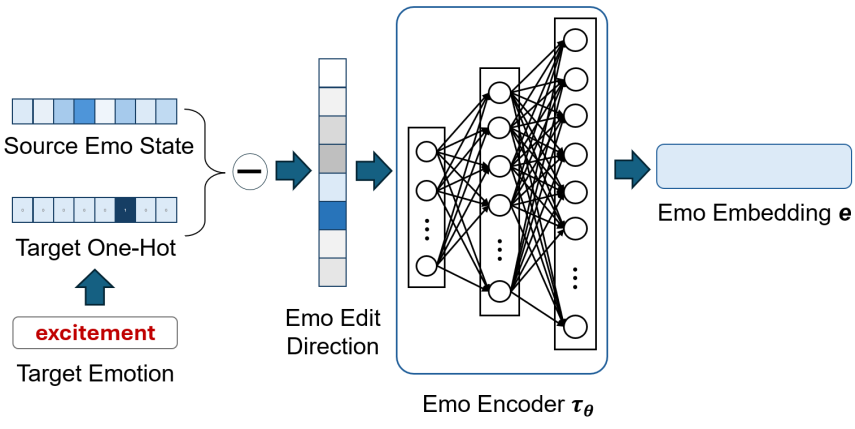


Figure S4. **The network structure of our emotion encoder τ_θ.**

S2.2. Baseline: Large Model Series

Emotion-evoked image generation involves three key steps: image understanding, instruction generation, and instruction-based image editing. To tackle these, we concatenate existing large language and vision models and introduce the baseline "Large Model Series" (LMS). This includes GPT-4o [4] for image captioning, followed by GPT-o4 [7] for generating reasoning-based text instructions, and Ip2p for image editing based on the instructions.

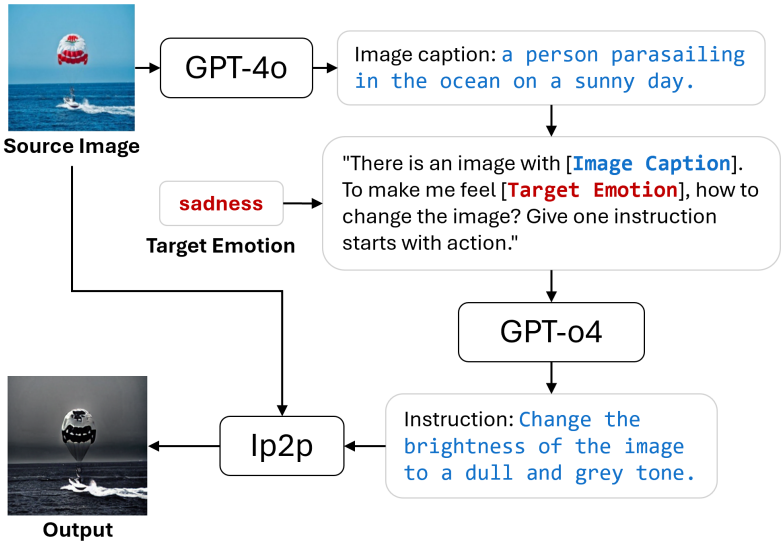


Figure S5. **Workflow of Large Model Series (LMS).** We decompose emotion-evoked image generation into three steps: employing GPT-4o for image understanding, utilizing GPT-o4 for instruction generation, and employing Ip2p for image editing. We refer to this approach of chaining multiple large models as the Large Model Series.

Figure S5 shows the workflow of LMS. First, we employ GPT-4o to comprehend the source image, generating an image caption corresponding to the source image. Subsequently, based on the generated caption, we utilize the sentence structure "There is an image with [Image caption]. To make me feel [Target Emotion], how to change the image? Give one instruction that starts with an action." to query GPT-o4. Following this, GPT-o4 generates an instruction to guide Ip2p in editing the source image, resulting in the final output.

S2.3. Human Psychophysics Experiment

Figure S6(a) shows the MTurk experiment schematic and Figure S6(b) shows the experiment instruction pages given to the participants. To control the quality of data collected, we have implemented preventive measures to screen participants:

(a) Each participant undergoes 6 randomly dispersed dummy trials within the real experiments. To assess if participants are making random selections, we use 6 image pairs with prominent emotional differences as references. Considering individual emotional response variability, participants are allowed a maximum of one incorrect choice in these dummy trials. Subjects exceeding an error rate of 1/6 are excluded, resulting in 24,480 trials. The outcomes of the 6 dummy trials are also excluded from the final result analysis. Figure S6(c) provides an example of a dummy trial.

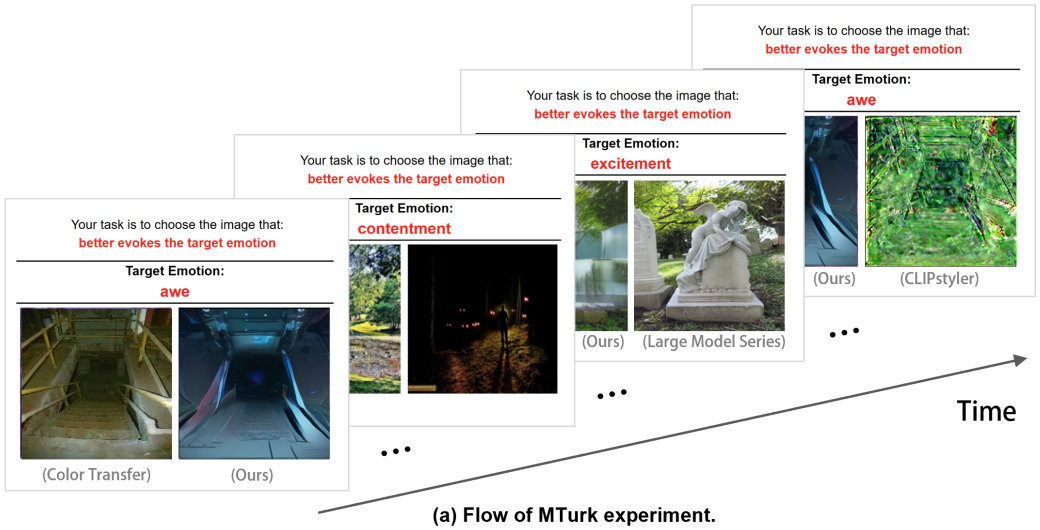
(b) Each participant can only take part in the experiment once.

(c) Image pairs used in the entire experiment are drawn from the 2,016 result pairs generated by our model and other SOTA methods.

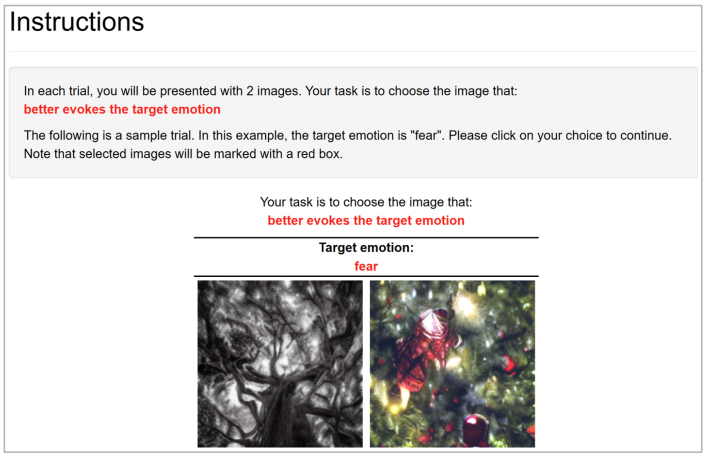
Participants view randomly sampled pairs from these 2,016 pairs, and all trials are presented in a random order. The order of the two images for selection is also randomized. An example trial in the real experiment is illustrated in Figure S6(d).

S2.4. Evaluation Metrics: Structural Score

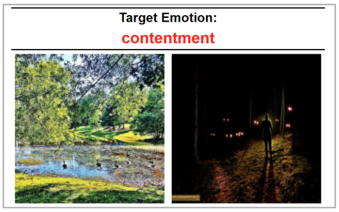
Emotion-evoked image generation needs to be evaluated from both emotional induction and structural preservation perspectives. Traditional metrics, such as SSIM, are inadequate for structural preservation evaluation because they do not consider emotional cues. The regions in the image that evoke emotions are called Emotion-Evoking Regions (R_{emo}). These areas need to be altered to change the source emotion and induce the target emotion. Conversely, some regions in the image



(a) Flow of MTurk experiment.



(b) Instructions and form of MTurk experiment.



(c) Dummy trial example.

(d) Real trial example.

Figure S6. **MTurk Experiment.** (a) Schematic of MTurk experiment. Participants are presented with a set of images and a target emotion. They must choose between two images, one generated by our model and the other by 9 state-of-the-art methods, selecting the one that more effectively evokes the target emotion. (b) Instructions of the MTurk experiment. (c) Dummy trial example. We chose six image pairs with prominent emotional distinctions as benchmarks to assess participants' comprehension of the task and their attentiveness to the experiment. (d) Real trial example.

lack significant emotional cues and are referred to as Emotion-Neutral Regions (R_{neu}). These regions should be preserved as much as possible during the editing process.

Grad-CAM [9] can visualize which parts of the image most influence the model's decision. Therefore, we use Grad-CAM with the emotion predictor $\mathcal{P}$ to identify Emotion-Evoking Regions (R_{emo}) in the source images. Figure S7 shows the visualization of the Emotion-Evoking and Emotion-Neutral Regions. Using Grad-CAM [9] with $\mathcal{P}$, we binarize Grad-CAM maps at a 0.5 threshold to identify Emotion-Evoking Regions (valued at 1) and Emotion-Neutral Regions (valued at 0) on source images.

To evaluate changes, we calculate pixel-level L1 differences between the source and generated images for these regions, denoted as L_{emo} and L_{neu}. The structural score is: $S_{str} = L_{emo}/(L_{emo} + L_{neu} + \epsilon)$ where ϵ is a small constant to avoid division by zero. This score reflects the proportion of changes in R_{emo} relative to all changed pixels, with higher values indicating better preservation of neutral areas.

Considering that Grad-CAM may be inaccurate or unstable due to the limitations of the emotion predictor $\mathcal{P}$, we invited 46 participants to annotate the emotion-evoking regions of 504 source images. Each worker annotated 50 trials, resulting in a total of 2,300 trials. Additionally, we included 6 dummy trials, which were the same as those used in our human

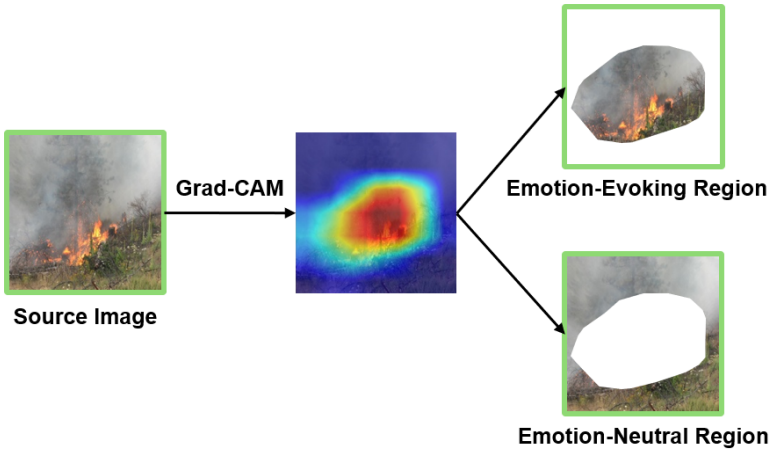


Figure S7. **Visualization Explanation of the Emotion-Evoking and Emotion-Neutral Regions.** The source image is framed in green. Using Grad-CAM [9] with $\mathcal{P}$, we binarize Grad-CAM maps at a 0.5 threshold to identify Emotion-Evoking Regions (valued at 1) and Emotion-Neutral Regions (valued at 0) on source images.

psychophysics experiment (Figure S6(c)), to ensure annotation quality. Results from workers who failed more than once on the dummy trials were discarded. Ultimately, each image was annotated by at least three individuals. We then averaged all the annotations and binarized the results to generate a human-annotated weight map. Figure S8 shows example annotations and a comparison with Grad-CAM.

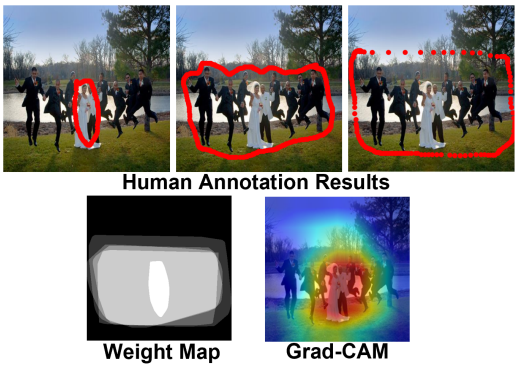


Figure S8. **Emotion-Evoking Region Annotated by Human.** Each image has been annotated by at least three human annotators who identified the emotion-evoking regions. We averaged all the annotations and binarized the result to generate a human-annotated weight map. Row 2 presents both the weight map and the heatmap computed using Grad-CAM, showing that the two are largely consistent.

S2.5. Implementation Details

Following [11], we initialize the weights of $\mathcal{E}$, $\mathcal{D}$, and ϵ_θ with the pre-trained Ip2p [1] weights. Throughout the training process, we maintain the fixed parameters of $\mathcal{E}$ and $\mathcal{D}$, focusing on training τ_θ and ϵ_θ. The frozen emotion predictor $\mathcal{P}$ is only used during inference. We conduct experiments with NVIDIA RTX A5000 GPUs, implementing our PyTorch-based framework using the ADAM optimizer. The max time step T is set to 1,000.

S3. Results

S3.1. Quantitative Evaluation in Cross-Valence Scenarios

We assess the generated outputs of all methods using Amazon Mechanical Turk (MTurk)[10] (Online). We recruit 136 participants, with each participant undergoing 180 trials, yielding a total of 24,480 trials. Figure S9 illustrates the preference

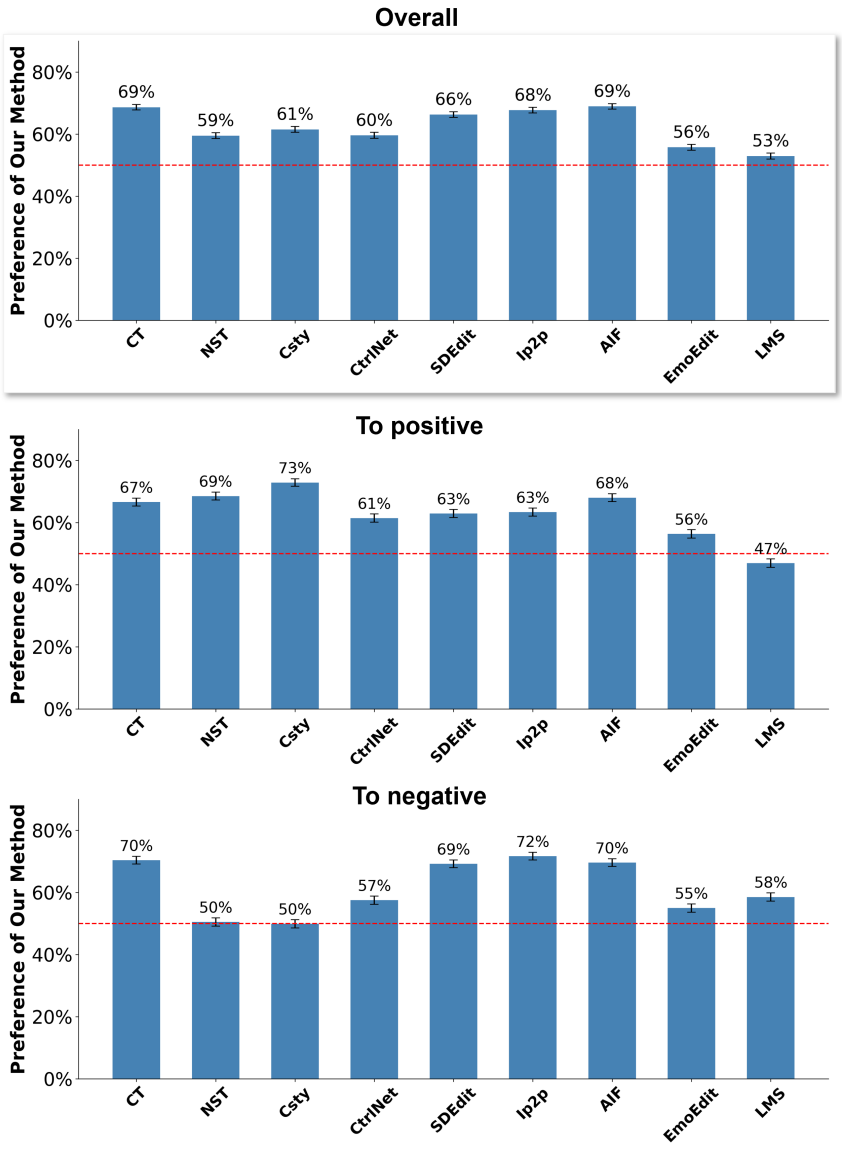


Figure S9. **Preference of Our Method Over SOTA Methods.** The first row shows the overall results, the second row only considers the trials with transitions from negative to positive emotions, and the third row focuses on the trials with transitions from positive to negative emotions. Chance is 50% in the red dotted line. Error bars are standard errors.

of our method over other state-of-the-art methods, demonstrating the superiority of our EmoEditor in producing results that evoke the target emotion compared to other state-of-the-art methods.

From the overall results (Row 1), we see that human participants consistently prefer the generated results of our EmoEditor over all competitive baselines. Results from the "To positive" (Row 2) and "To negative" (Row 3) indicate that Neural-Style-Transfer (NST) [3] and CLIP-Styler (Csty) [5] perform comparably to our method in generating images that evoke negative emotions, but struggle with positive outcomes. This indicates that negative emotions are easily influenced by factors such as chaotic textures and overall tones, whereas positive emotions are more challenging and require an understanding of the source image along with effective adjustments. Our EmoEditor significantly outperformed Color Transfer (CT) [8] and AIF [13] in all scenarios. This confirms that emotion-evoked image generation involves more than just global color and brightness adjustments. Additionally, our EmoEditor surpasses SDEdit [6] and Ip2p [1] in generating results evoking positive emotions, while the improvement in generation performance is even more significant in generating results evoking negative emotions. This highlights challenges in current diffusion models for understanding

and generating emotion-evoked images and indirectly confirms the greater challenge of generating positive emotion-evoked images compared to negative ones.

For both positive and negative emotion generation, our method is preferred over EmoEdit in human evaluations. Furthermore, the performance of our EmoEditor and the Large Model Series (LMS) is comparable. While LMS achieves slightly higher scores in positive emotion generation, the margin is narrow and largely influenced by its reliance on powerful external reasoning modules. In contrast, our method offers a compact and fully end-to-end solution that generalizes well across both positive and negative emotions, without the need for multi-stage processing or large-scale language models. Moreover, these language models are often proprietary and equipped with safety constraints, which limit their applicability in privacy-sensitive contexts.

In addition, we show the confusion matrix of different methods in Figure S10. The confusion matrix provides more detailed insights into the emotion transition effects of our EmoEditor across specific emotion categories. From the upper quadrant of the confusion matrix, it can be observed that people prefer Neural-Style-Transfer (NST) [3] and CLIP-Styler (Csty) [5] more in the manipulation of emotions from positive to negative. This preference may arise because these methods excel in inducing negative emotions through distortions and irregular textures on the source image. In contrast, our EmoEditor achieves higher human preference scores in the lower quadrant of the confusion matrix, demonstrating our method's significant superiority in generating positive-valence images. Overall, for generating positive-valence images, our EmoEditor achieves the best results in generating images that evoke awe and contentment, but it is slightly less effective in generating images that evoke excitement. This may be because excitement is a more dynamic and complex emotion, requiring a more intricate combination of visual elements.

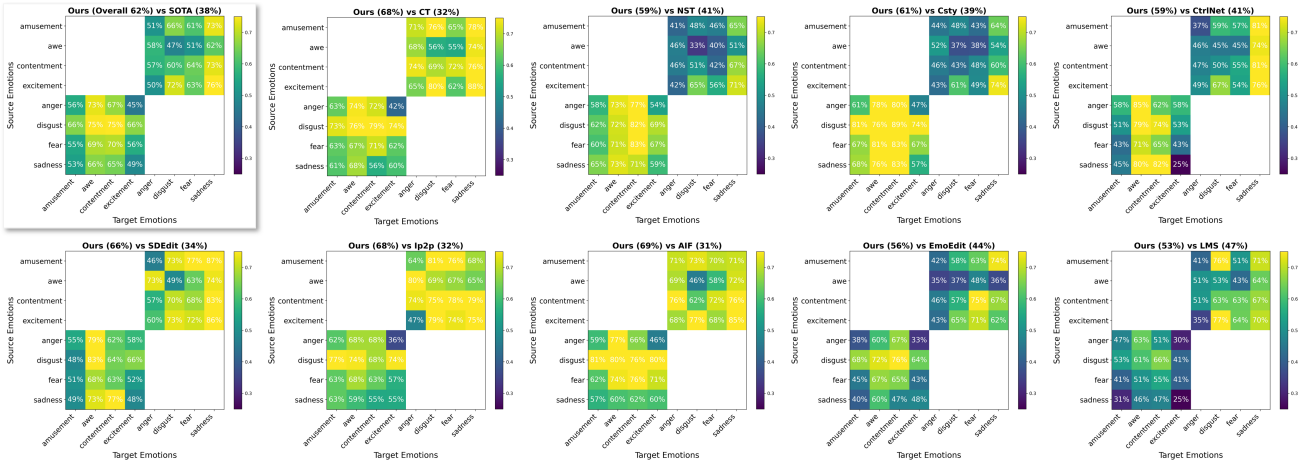


Figure S10. **Confusion matrix of different methods.** The confusion matrix reflects the human preference for our EmoEditor over other methods based on source and target emotion pairs. The top-left matrix presents the average overall results of our EmoEditor compared to the other SOTA methods. The remaining nine matrices show individual comparisons between our EmoEditor and each of the nine SOTA methods. Average preferences for our EmoEditor and other methods are indicated above each matrix. The vertical axis corresponds to the source emotion, while the horizontal axis represents the target emotion. Each cell in the matrix denotes the ratio at which our method is selected for the corresponding transition from the source to the target emotion.

Table S1 presents the quantitative evaluation of generated images for all competitive methods. Our EmoEditor achieves the best performance in both S_{emo} and ESMI. Although Ip2p scores highest on S_{str}, this is because it makes minimal changes to the original images in most cases, as reflected by its low S_{emo} scores and visual results. In contrast, our method effectively edits the emotion-evoking regions in the source images, achieving a balance between evoking the target emotion and preserving the structure, which is supported by the highest ESMI score.

S3.2. Visualization of Emotional Image Generation across Valence

Figure S11 illustrates the visualizations of images generated by all methods for the task of generating positive-valence images from negative emotions. Color-Transfer (CT) [8] proves ineffective for significant emotional enhancement because it primarily replicates the color features of the reference image. For instance, in example(b), it enhances brightness but fails to evoke awe as it does not alter elements causing fear. Neural-Style-Transfer (NST) [3] heavily relies on randomly selected reference images, lacking in preserving the fundamental semantic content of the source image. For example, in example(a),

Metric	CT	NeurST	Csty	CtrlNet	SDEdit	Ip2p	AIF	EmoEdit	LMS	Ours
$S_{emo}(\%)\uparrow$	27.70	65.50	38.54	36.34	39.16	19.92	31.42	46.88	38.77	**69.89**
$S_{str}(\%)\uparrow$	28.96	29.47	30.11	29.55	32.65	**34.82**	28.00	31.60	32.03	33.22
ESMI(%)↑	28.33	47.48	34.33	32.94	35.91	27.37	29.71	39.24	35.40	**51.56**

Table S1. **Quantitative Evaluation of Generated Images for All Competitive Methods.** Best in bold. Larger (↑) is better.

it generates multiple differently colored blocks but fails to evoke the emotion of excitement. CLIPstyler (Csty) [5], SDEdit [6], and CtrlNet [15] often introduce inexplicable textures, losing the semantics and structure of the source images.

Ip2p [1], due to its limited understanding of emotions, often produces results identical to the source image, showcasing restricted image generation capabilities. AIF [13] heavily relies on text descriptions for understanding emotions and can only perform global edits, similar to applying a filter. As a result, it fails to achieve effective local edits to evoke the target emotion. EmoEdit [14] has difficulty accurately identifying and localizing strongly emotion-evoking regions in source images, resulting in ineffective or imprecise edits. In example (a), Ip2p, AIF, and EmoEdit fail to understand that the fear in the source image is caused by a strange shadow. As a result, they do not alter the shadow and thus cannot counteract the fear. Even with a series of large models in LMS, it does not consistently generate ideal outputs to trigger the target emotion. In addition, it often greatly changes the scene of the source image to achieve the goal of changing the emotion. In example (c), it fails to recognize that the large flames in the background are the primary source of the original emotion, and instead generates an image where the person remains in front of the fire, merely adjusting the flame colors to be more vibrant. Such changes are insufficient to alter the original emotional impact.

In contrast, our EmoEditor, requiring only the target emotion and source image inputs, can generate highly creative images that evoke the target emotion while striving to maintain scene structure and semantic coherence. In example(a), our EmoEditor understands that the shadow in the source image is the key factor triggering fear. Instead of altering the shadow's shape, it generates a sunset sky background, transforming the shadow into a cloud in the sky, which successfully evokes the target emotion of excitement. In example(b), it replaces a Halloween pumpkin with a castle to elicit the target emotion of awe. In example(c), it replaces the background flames with a red sunset in an attempt to evoke contentment. Notably, the generation process involves no manual intervention, showcasing the imagination and creativity of our model.

Additionally, Figure S12 shows the generation results of different methods for transitioning from positive to negative emotions. Compared to generating positive-valence images, CT and AIF perform better in generating negative-valence images. This indicates that negative emotions are more easily evoked than positive ones and can be achieved through global edits to the source image.

In addition to its creativity in generating positive-valence images, our EmoEditor also exhibits strong understanding and editing capabilities in generating negative emotions. In example(a), EmoEditor not only darkens the image's tone to create a fearful atmosphere but also transforms the field into a desolate scene with weeds and branches, effectively evoking the emotion of fear. In example(c), it submerges the person, previously relaxing on a swim ring, beneath the pool water and alters his appearance to look frail and distressed, resembling a drowning victim, to evoke sadness. This highlights that EmoEditor can comprehend the content of the source image, align with its original emotional state, and combine it with the target emotion to perform specific edits, addressing both global and local aspects.

S3.3. CrossTest Evaluation Results

Our method employs an emotion predictor to guide emotional expression during image editing. However, using the same type of predictor for both generation and evaluation can introduce bias, as models may exploit specific predictor behaviors to inflate performance. To mitigate this, we split EmoSet into two disjoint subsets and trained two independent emotion predictors: one for generation and one for evaluation.

As shown in Table S2, NST achieves the highest ESMI score. However, as illustrated in our visualization results, its outputs often feature emotion-colored overlays that obscure the original image content. These artifacts tend to inflate S_{emo} because the emotion predictor is overly sensitive to low-level color cues rather than genuine emotional semantics. This exposes a key limitation of current evaluation practices and underscores the need for more semantically grounded emotion understanding.

In contrast, results from our human study (Figure S9) show a clear preference for our method. Participants favored images where emotion-evoking regions were meaningfully altered, rather than simply covered with ambiguous color patches. This suggests that semantic coherence in emotional edits is more impactful and preferable than superficial visual modifications.

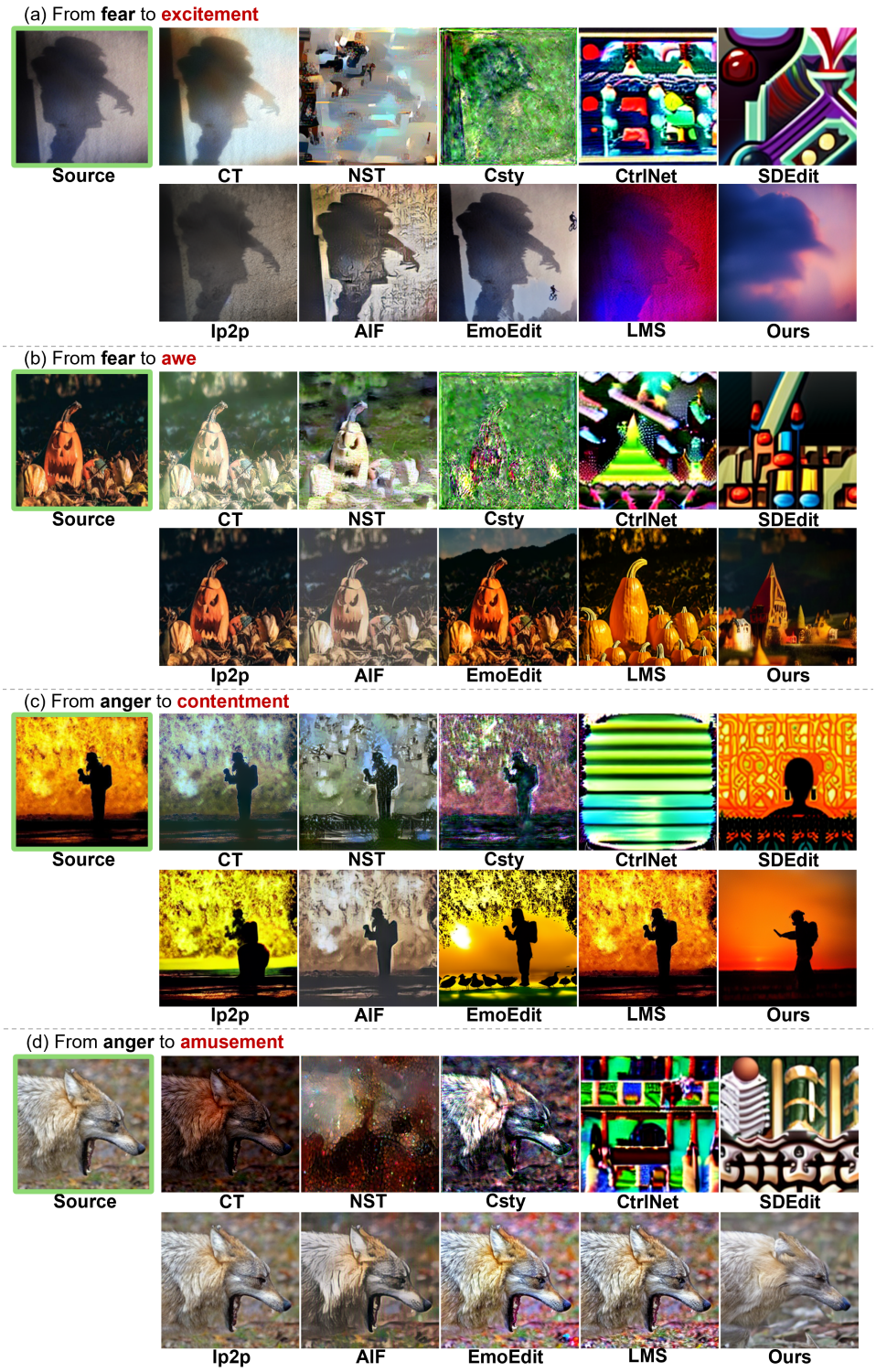


Figure S11. **Visualization of Generated Images from Different Methods (from negative to positive).** The target emotion is highlighted in red and the source image is framed in green.

Our method ranks second, only 2% below NST, while outperforming all other baselines. Unlike NST, it preserves visual semantics while achieving strong emotional alignment, demonstrating robust generalization and a more balanced trade-off

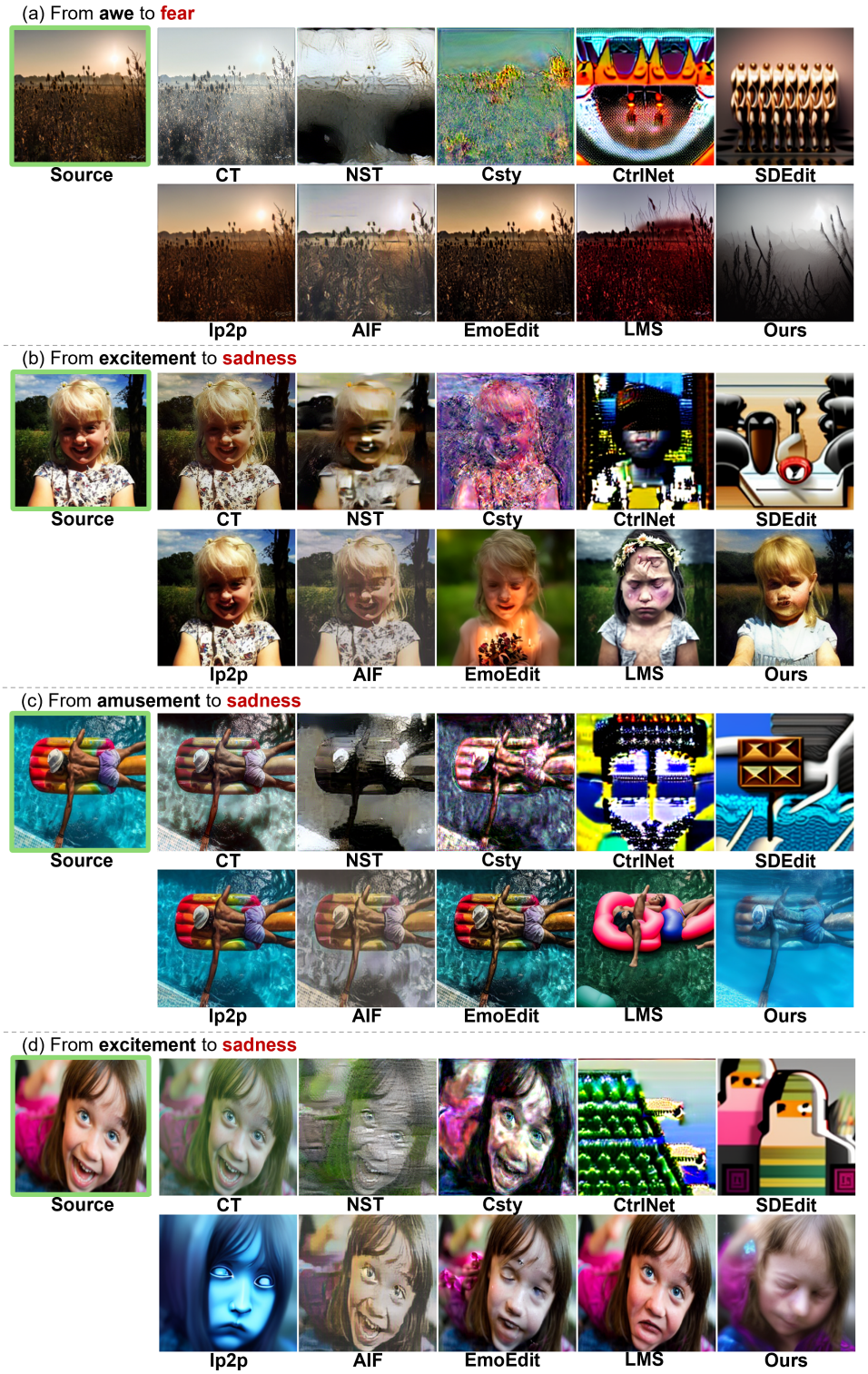


Figure S12. **Visualization of Generated Images from Different Methods (from positive to negative).** The target emotion is highlighted in red and the source image is framed in green.

between content fidelity and emotional expressiveness.

ESMI(%) ↑	CT	NST	Csty	CtrlNet	SDEdit	Ip2p	AIF	EmoEdit	LMS	Ours
CrossTest	28.39	**45.35**	32.61	39.38	38.50	26.54	28.67	38.50	34.09	<u>43.95</u>

Table S2. **Quantitative Comparison of Competing Methods under CrossTest.** CrossTest refers to the evaluation scheme where images are generated using an emotion predictor trained on the EmoSet subset 1 and tested by an emotion predictor trained on the EmoSet subset 2. Bold indicates the best performance, while the second-best is underlined. Larger (↑) is better.

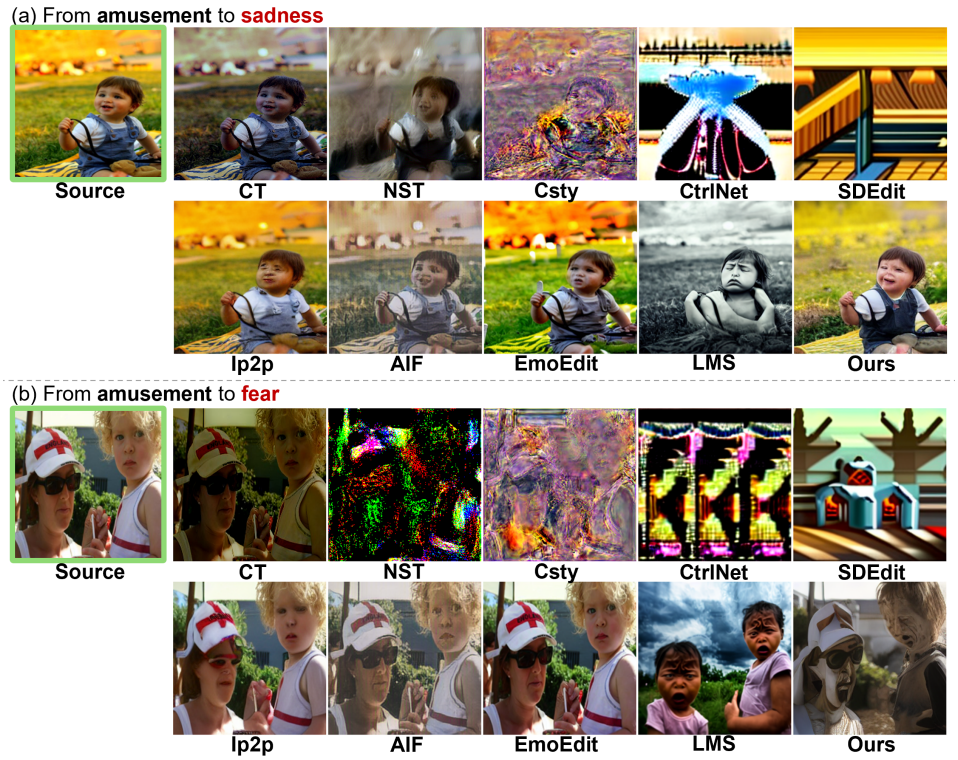


Figure S13. **Failure Case.** The target emotion is highlighted in red and the source image is framed in green.

S3.4. Limitations

While EmoEditor excels in quantitative performance and visual results, it encounters difficulties in handling fine details of individuals, especially when their faces in the images are small. For example, in Figure S13, although our EmoEditor attempts to make the boy appear sad, the boy's face seems somewhat twisted and distorted.

Moreover, although we use the 8-category Mikels model, real emotions are far more diverse and complex, and distinguishing positive emotions is often more challenging than negative ones. This is an area for future exploration.

References

[1] 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. 1, 5, 6, 8

[2] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020. 1

[3] Leon A. Gatys, Alexander S. Ecker, and Matthias Bethge. A neural algorithm of artistic style, 2015. 6, 7

[4] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 3

[5] Gihyun Kwon and Jong Chul Ye. Clipstyler: Image style transfer with a single text condition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18062–18071, 2022. 6, 7, 8

[6] Chenlin Meng, Yutong He, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Guided image synthesis and editing with stochastic differential equations. *arXiv preprint arXiv:2108.01073*, 2021. 6, 8

[7] OpenAI. Introducing openai o3 and o4-mini: Our smartest and most capable models to date with full tool access. https://openai.com/index/introducing-o3-and-o4-mini/, 2025. 3

[8] François Pitié, Anil C Kokaram, and Rozenn Dahyot. Automated colour grading using colour distribution transfer. *Computer Vision and Image Understanding*, 107(1-2):123–137, 2007. 6, 7

[9] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE international conference on computer vision*, pages 618–626, 2017. 4, 5

[10] Amazon Mechanical Turk. Amazon mechanical turk. *Retrieved August*, 17:2012, 2012. 5

[11] Tengfei Wang, Ting Zhang, Bo Zhang, Hao Ouyang, Dong Chen, Qifeng Chen, and Fang Wen. Pretraining is all you need for image-to-image translation, 2022. 5

[12] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 1

[13] Shuchen Weng, Peixuan Zhang, Zheng Chang, Xinlong Wang, Si Li, and Boxin Shi. Affective image filter: Reflecting emotions from text to images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10810–10819, 2023. 6, 8

[14] Jingyuan Yang, Jiawei Feng, Weibin Luo, Dani Lischinski, Daniel Cohen-Or, and Hui Huang. Emoedit: Evoking emotions through image manipulation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24690–24699, 2025. 8

[15] 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. 8

[16] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 1