ICLR

Sample Reward Soups: Query-Efficient Multi-Reward Guidance for Text-to-Image Diffusion Models

Yinghua Yao[1], Yuangang Pan[1†], Guoji Fu[2], Ivor Tsang[1,3]

[1]Agency for Science, Technology and Research, Singapore
[2]National University of Singapore, Singapore
[3]Nanyang Technological University, Singapore

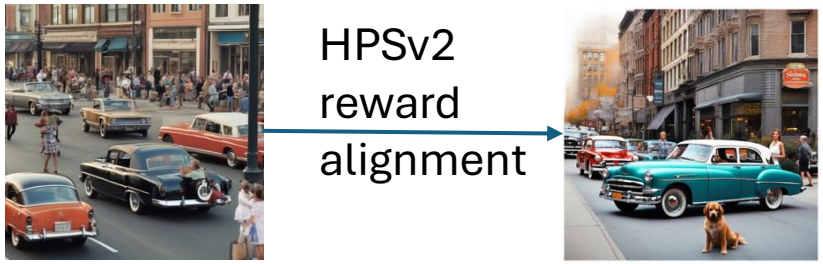
Centre for Frontier AI Research
CFAR

NUS National University of Singapore

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

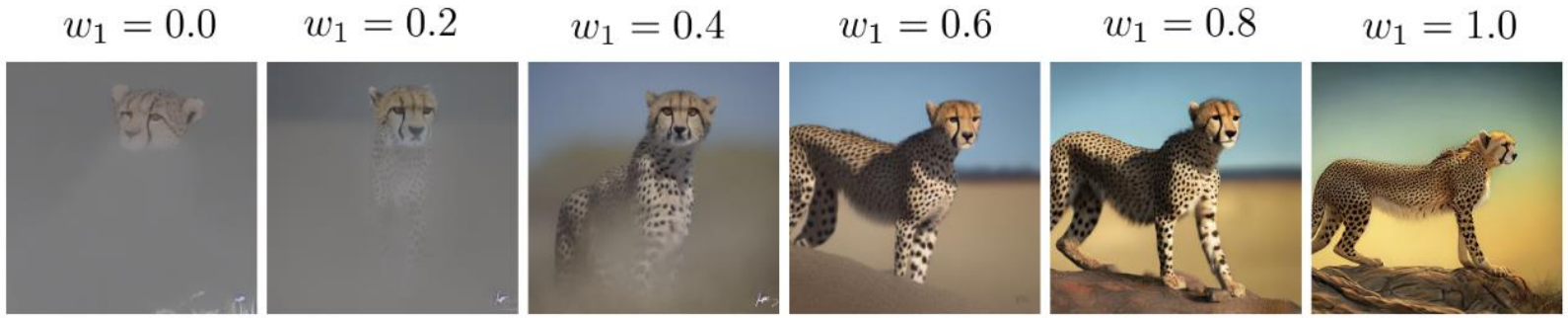
Alignment for Text-to-Image Models

- Alignment of Text-to-Image (T2I) models (diffusion/flow models)



Classic cars on a city street with people and a dog

- Multi-reward alignment: aesthetics, compressibility, etc



$w_1 f_1 + w_2 f_2$: aesthetic reward f_1 and compressibility reward f_2

Alignment Techniques

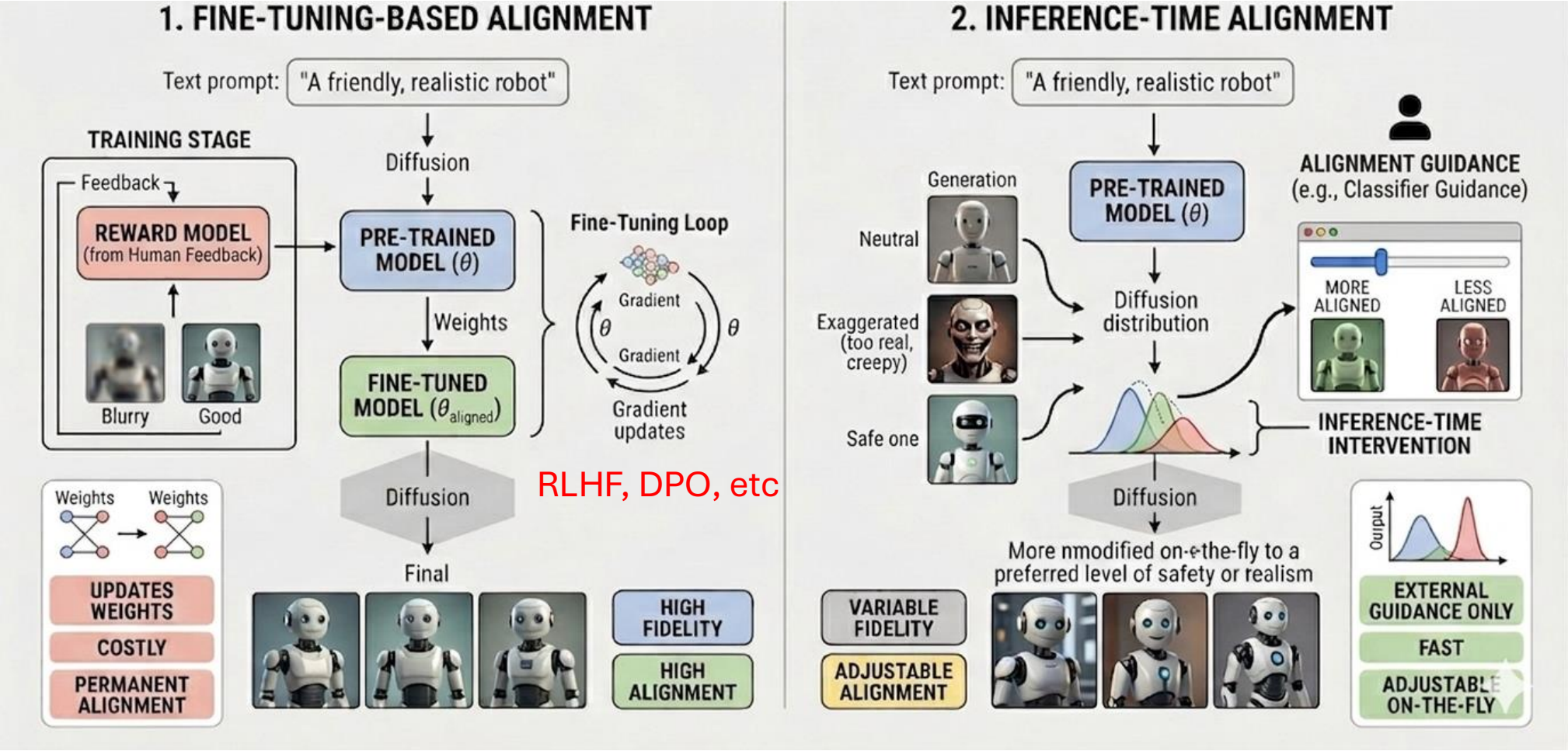
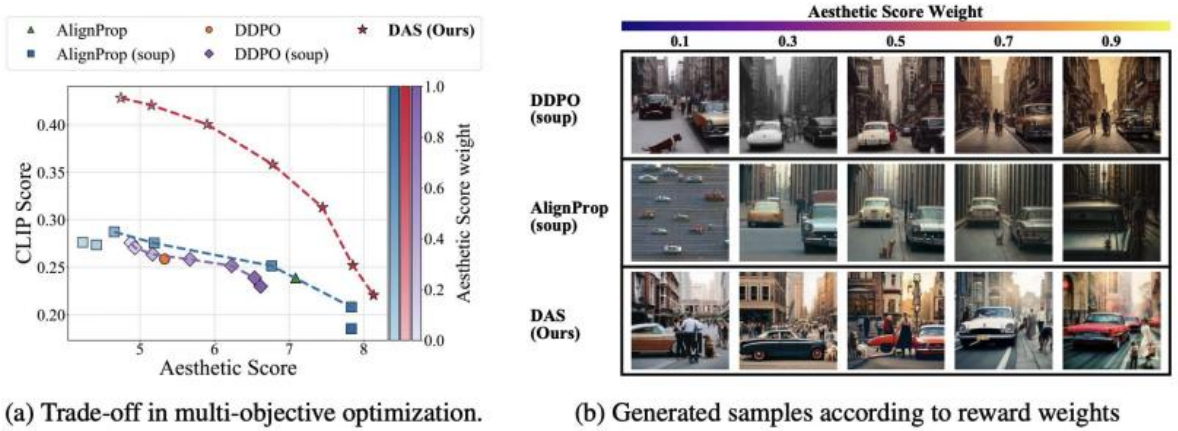


Figure is created by Gemini.

Challenges of Multi-Reward Alignment

- Weighted-Sum: $F(x, w_{1:M}) = \sum_{i=1}^{M} w_i f_i(x)$
- Reward combinations grows exponentially with the number of reward functions: $\{w_{1:M}^j\}_{j=1}^{L}, L \gg M$

$$w \cdot \text{Aesthetic Score} + (1 - w) \cdot 20 \cdot \text{CLIPScore}$$

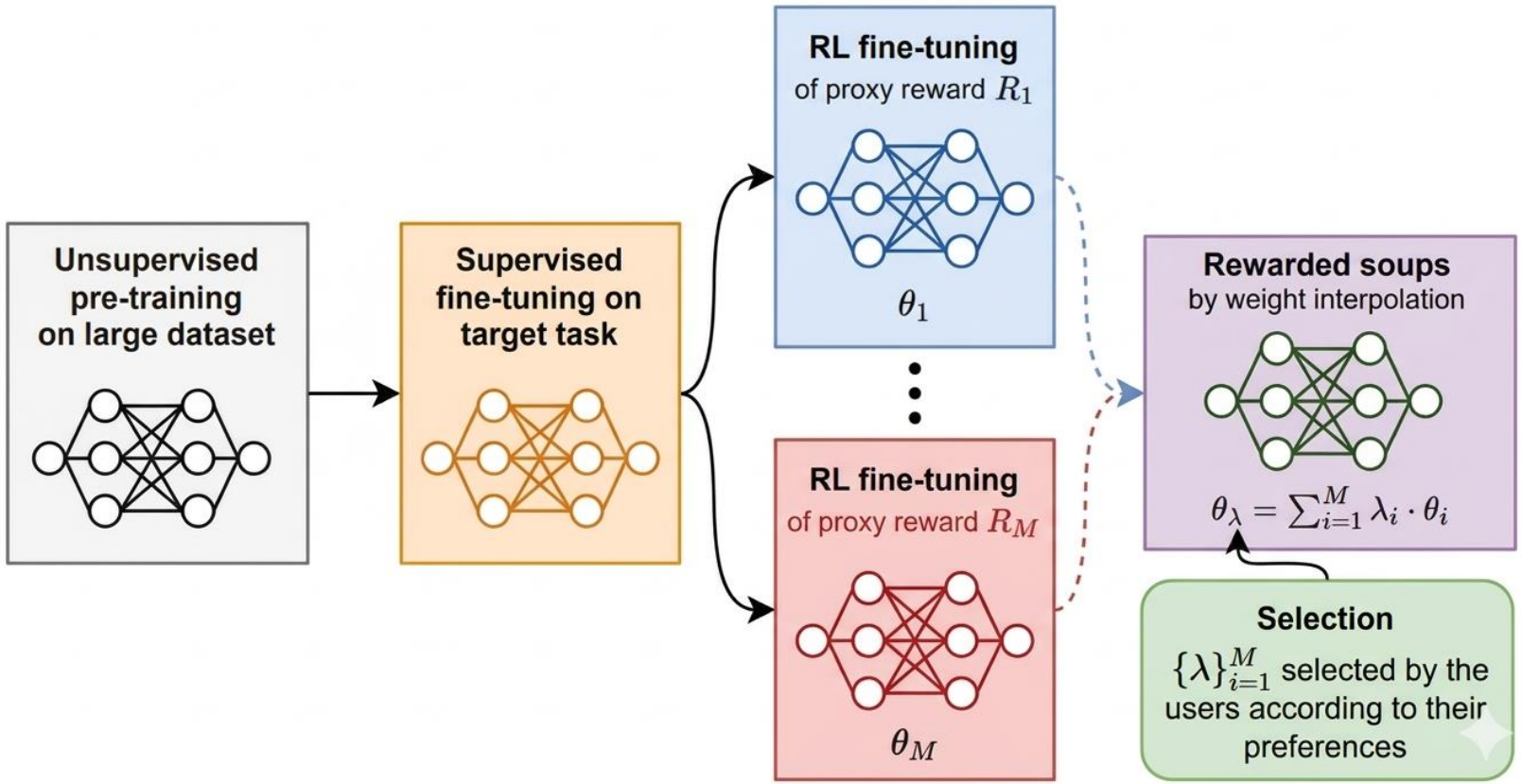


(a) Trade-off in multi-objective optimization.

(b) Generated samples according to reward weights

Figure is from "Test-time Alignment of Diffusion Models without Reward Over-optimization," ICLR 2025.

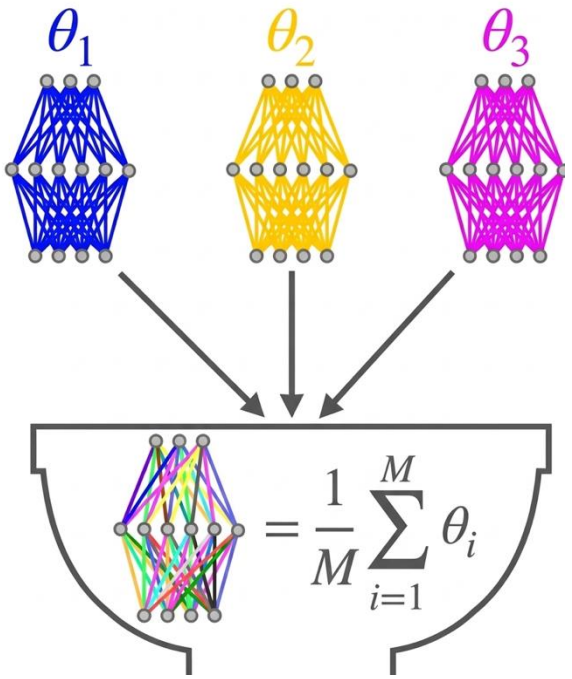
Rewarded Soups for Efficient Fine-tuning

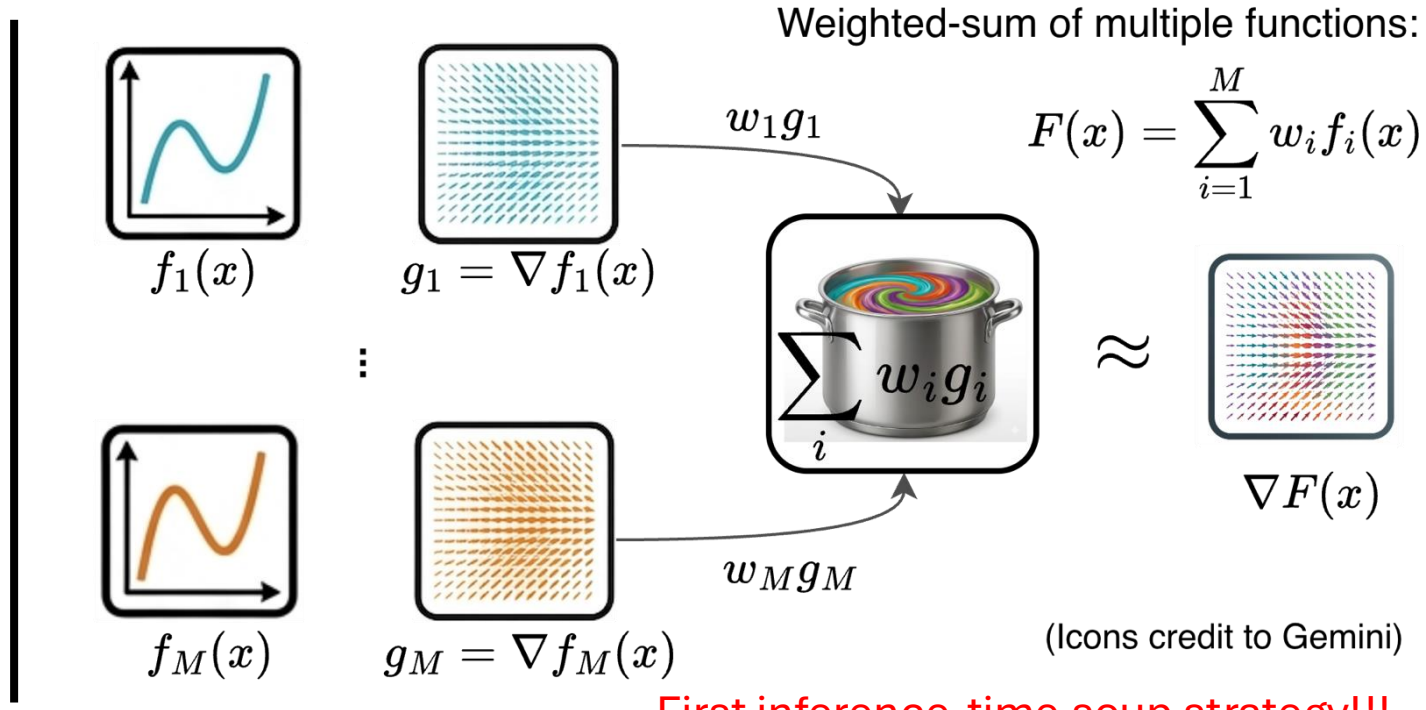


NO efficient strategy for inference-time multi-reward alignment

Figure is from “Rewarded soups: towards Pareto-optimal alignment by interpolating weights fine-tuned on diverse rewards,” NeurIPS 2023. Small edits credit to Gemini.

Weight Soups ⟶ Gradient Soups





First inference-time soup strategy!!!

- Reward guidance subject to the number of reward functions M, not the number of reward combinations L. $M \ll L$
- Gradient soups can approximate the gradient of weighted function under the black-box denoising alignment process

Left figure is from “Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time”, ICML 2022
Small edits credit to Gemini.

Black-box Reward Inference-time Alignment

- Optimize the denoising distribution at t step for high-reward generation

$$\max_{\boldsymbol{\mu}_{t-1}} \mathcal{F}(\boldsymbol{\mu}_{t-1}) = \mathbb{E}_{\mathcal{N}(\boldsymbol{x}_{t-1};\boldsymbol{\mu}_{t-1},\beta_t \boldsymbol{I})}[f(\boldsymbol{x}_{t-1})] \qquad \boldsymbol{\mu}_{t-1} = \mu_\theta(\boldsymbol{x}_t, t)$$

- Reward-guided search gradient

$$\nabla_{\boldsymbol{\mu}_{t-1}} \mathcal{F}(\boldsymbol{\mu}_{t-1}) = \frac{1}{\sqrt{\beta_t}} \mathbb{E}_{\mathcal{N}(\boldsymbol{z};\boldsymbol{0},\boldsymbol{I})} \left[f(\boldsymbol{\mu}_{t-1} + \sqrt{\beta_t}\boldsymbol{z})\boldsymbol{z} \right]$$

Reward query

Obtain rewards for N samples. Update μ_{t-1} towards high-reward regions

Sample Reward Soups

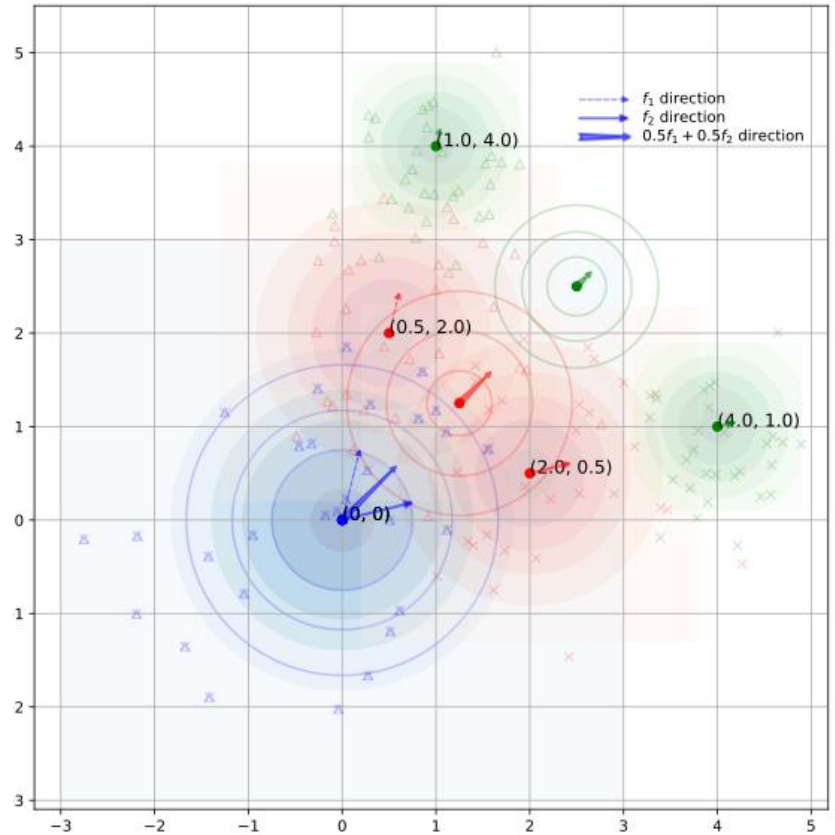
- Obtain search gradient for each reward function

$$\nabla_{\boldsymbol{c}_{t-1}^m} \mathcal{F}(\boldsymbol{c}_{t-1}^m, e_m)$$

- Interpolate search gradients

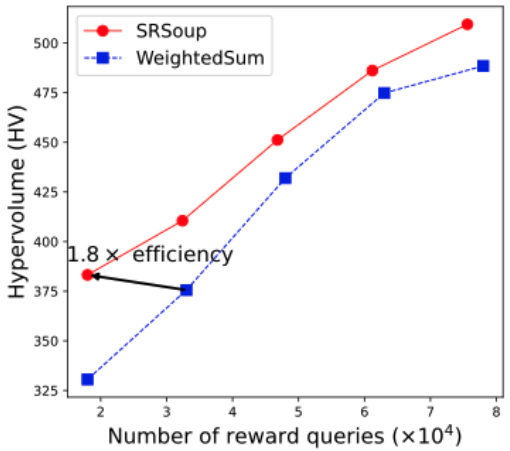
$$\nabla_{\boldsymbol{\mu}_{t-1}^l} \mathcal{F}(\boldsymbol{\mu}_{t-1}^l, w_{1:M}^l) = \sum_{m=1}^{M} w_m^l \Big(\nabla_{\boldsymbol{c}_{t-1}^m} \mathcal{F}(\boldsymbol{c}_{t-1}^m, e_m) + \frac{1}{\sqrt{\beta_t} N} \sum_{n=1}^{N} f_m(\boldsymbol{x}_{t-1}^{m,n})(\boldsymbol{c}_{t-1}^m - \boldsymbol{\mu}_{t-1}^l) \Big)$$

- Hybrid schedule: interpolate gradients at the first K steps, weighted-sum update for later steps

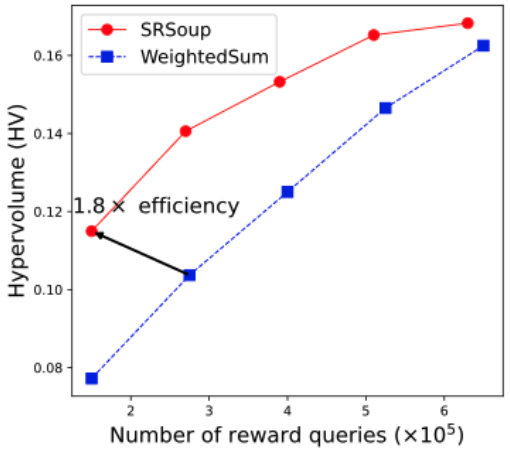


Insight: Early denoising distributions for different rewards are still close, so sample rewards can be shared

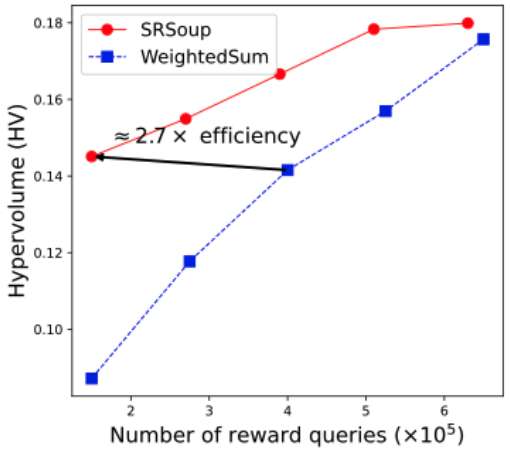
Efficiency Gains



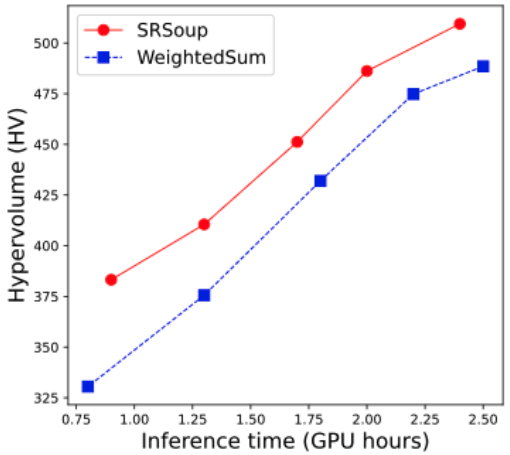
Aesthetic & Compressibility



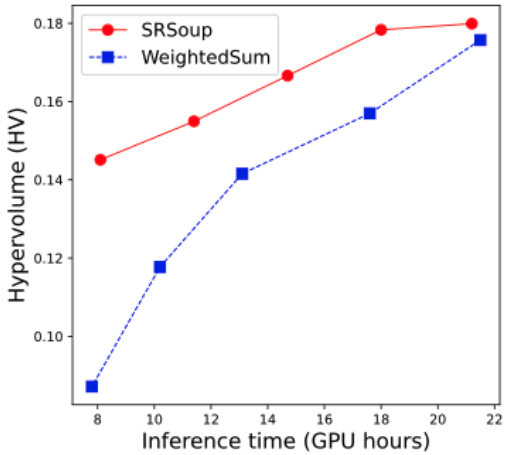
Aesthetic & PickScore



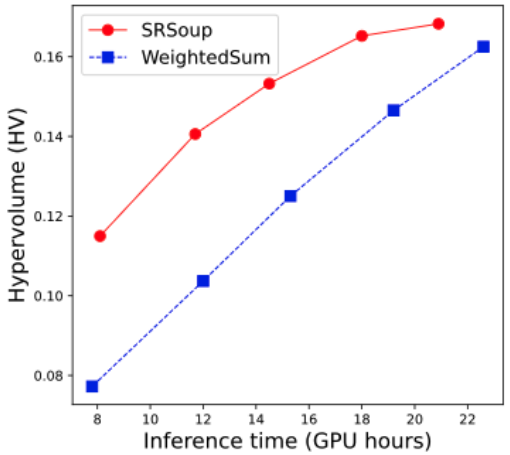
Aesthetic & HPSv2



Aesthetic & Compressibility

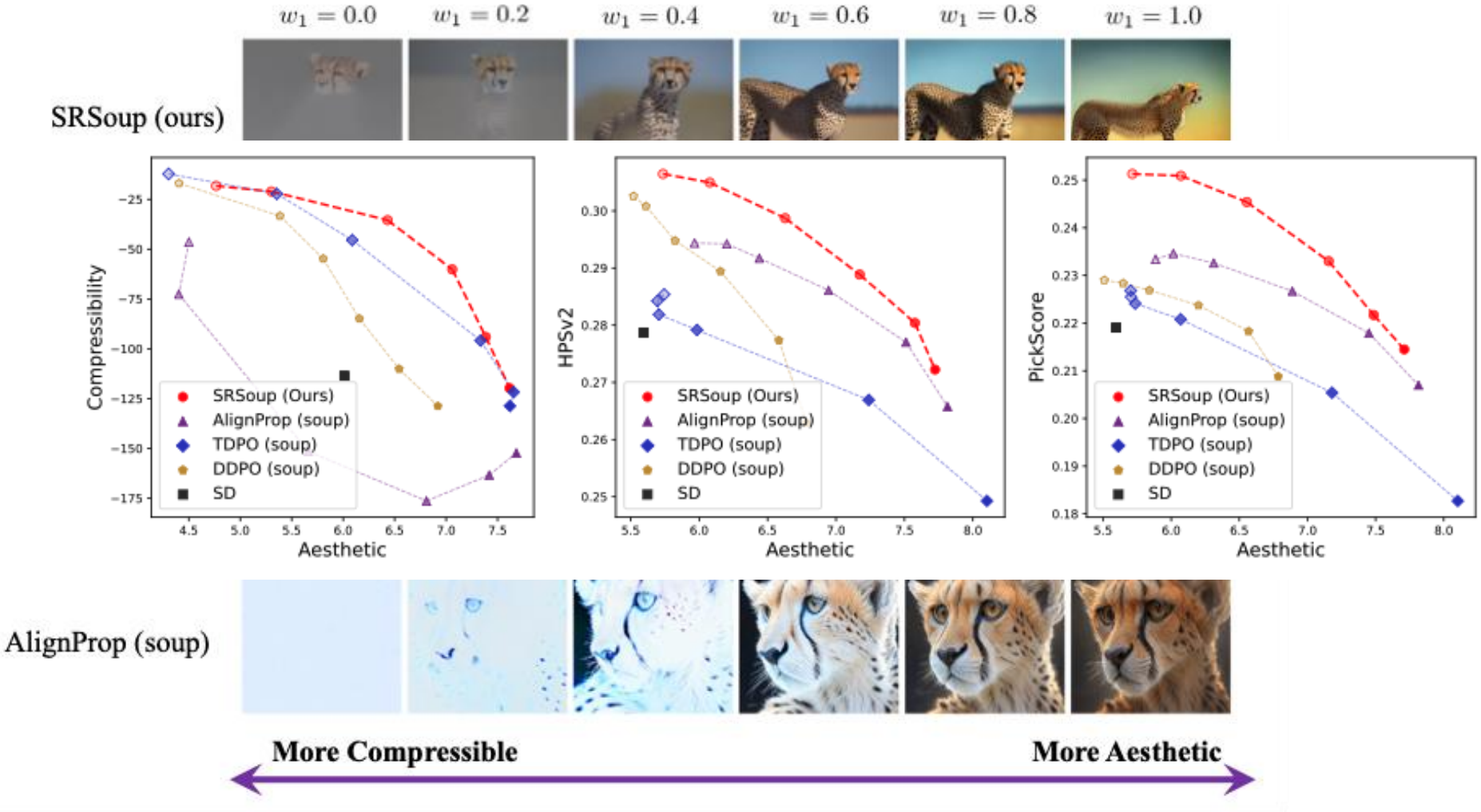


Aesthetic & PickScore



Aesthetic & HPSv2

Compared with Model Soup Baselines



Code Link:

Conclusion

- First **inference-time soup strategy (gradient soups)** for multi-reward diffusion alignment.
- Enables **training-free Pareto sampling** over the full preference space.
- Achieves **1.8X-2.7X** better query efficiency than weighted-sum guidance.

Thank you!