Color Conditional Generation with Sliced Wasserstein Guidance

Alexander Lobashev[†], Maria Larchenko[⋆], Dmitry Guskov[†],

[†]Glam AI, San Francisco, US
[⋆]Magicly AI, Dubai, UAE

NeurIPS 2025, spotlight

25 October 2025

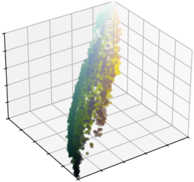
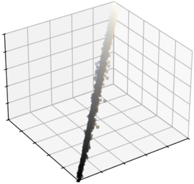
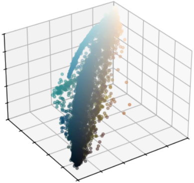
Outline

1. Color-conditional Image Generation
2. Distribution-conditional Image Generation
3. Solving Non-linear Inverse Problems using Diffusion Models
4. Measuring Distances between Probability Distributions
5. Wasserstein Distance
6. Sliced Wasserstein Distance
7. Generalized Sliced Wasserstein Distance
8. Baselines and Evaluation Metrics
9. Experimental Results
10. Conclusion and Discussion

Color-conditional Image Generation

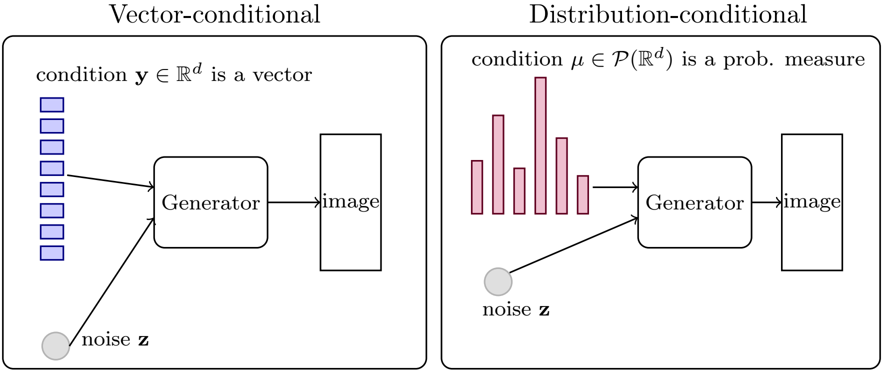
For a given image, there is an associated color distribution. One might want to generate an image based on a text description while maintaining a fixed color distribution.



Distribution-conditional Image Generation

In a more general case, there might be a distribution associated with a given image, such as the distribution of VGG activations computed for different image patches. One could also request text-based image generation where the associated distribution is fixed.



Solving Non-linear Inverse Problems using Diffusion Models

General inverse problem may be formulated as finding a vector x from a prior distribution $p(x)$ that is consistent with the observations y:

$$y = A(x) + n, \tag{1}$$

where A is an observation operator and n is a Gaussian noise.
Conditional score could be expressed as:

$$\nabla_x \log p(x|y) = \nabla_x \log p(y|x) + \nabla_x \log p(x) \tag{2}$$

The likelihood term then becomes

$$p(y|x) = e^{-\frac{1}{\sigma_n^2} d(y, A(x))} \tag{3}$$

and the gradient of log-likelihood is

$$\nabla_x \log p(y|x) = -\frac{1}{\sigma_n^2} \nabla_x d(y, A(x)), \tag{4}$$

where we need a distance $d(y, A(x))$ which compares observations $A(x)$ with the target observations y.

Solving Non-linear Inverse Problems using Diffusion Models

Consider a simplified algorithm:

Algorithm 1 Conditional Generation with Guidance

```
Initialize latent vector x_T ~ N(0, I) and target y
for t = T to 1 do
    Get prediction of x_0(x_t) ← DDIM(t, x_t)
    Compute loss L ← L(x_0(x_t), y)
    Update latent x_t^* ← x_t − ∇_{x_t} L
    Get next latent x_t ← DDIM(t, x_t^*)
end for
```

Solving Non-linear Inverse Problems using Diffusion Models

Algorithm 2 Conditional generation with guidance using control vector

Initialize latent vector $x_T \sim \mathcal{N}(0, I)$ and target y
for $t = T$ to 1 **do**
    $u \leftarrow \mathbf{0}$ ▷ Initialize control vector
    for $j = 1$ to M **do**
        $\hat{x}_t \leftarrow x_t + u$
        Get prediction of $x_0 \leftarrow \text{DDIM}(t, \hat{x}_t)$
        Compute loss $\mathcal{L} \leftarrow \mathcal{L}(x_0, y)$
    end for
    Update control vector $u \leftarrow u - \nabla_u \mathcal{L}(u)$
end for
Update latent $x_t^* \leftarrow x_t + u$
Get next latent $x_t \leftarrow \text{DDIM}(t, x_t^*)$

Measuring Distances between Probability Distributions

Given a two probability measures μ and ν on $\mathbb{R}^n$ one can introduce a mapping $\rho : \mathcal{P}(\mathbb{R}^n) \times \mathcal{P}(\mathbb{R}^n) \longrightarrow \mathbb{R}_{\geq 0}$ which satisfies the following properties

1. (symmetry): $\rho(\mu, \nu) = \rho(\nu, \mu)$
2. (identity of indiscernibles): $\rho(\mu, \nu) = 0$ if and only if $\mu = \nu$ almost everywhere.
3. (triangle inequality) $\rho(\mu, \nu) + \rho(\nu, \sigma) \geq \rho(\mu, \sigma)$
4. (weak convergence) The numerical sequence satisfies $\lim_{n\to\infty} \rho(\mu_n, \mu) = 0$ if and only if there is weak-* convergence of probability measures $\mu_n \longrightarrow \mu$.

The weak convergence property ensures that if we use ρ as our loss function, then minimizing the loss will result in the trained model capturing the true data distribution.

Wasserstein Distance

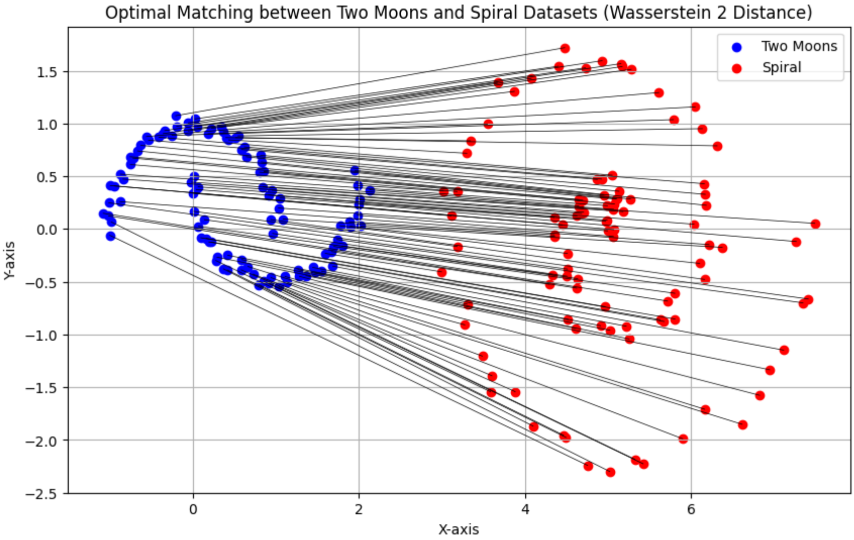
A common metric for measuring the distance between two probability distributions is the Wasserstein distance, rooted in optimal transport theory.
The Wasserstein distance of order p is defined as follows:

$$\mathrm{W_p}(\pi_0, \pi_1) = \left(\inf_{\pi \in \Pi(\pi_0, \pi_1)} \int_{\mathcal{X}_0 \times \mathcal{X}_1} d(x, y)^p \, d\pi(x, y) \right)^{1/p}, \quad (5)$$

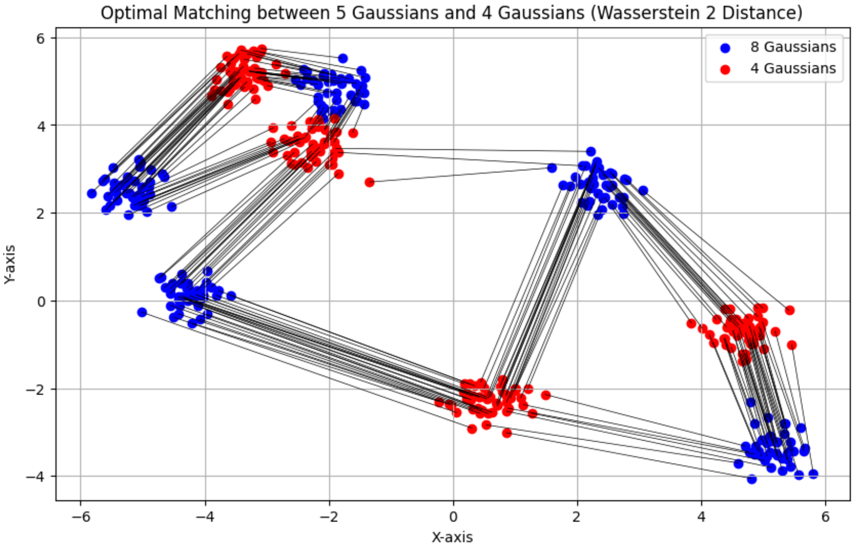
where $\Pi(\pi_0, \pi_1)$ represents the set of all possible couplings between π_0 and π_1, and d is a chosen metric.

- Theorem: $\mathrm{W_p}(\pi_0, \pi_1)$ satisfies properties (1)-(4): symmetry, identity of indiscernibles, triangle inequality, weak convergence.

Wasserstein Distance. Illustrations



Wasserstein Distance. Illustrations



Sliced Wasserstein Distance

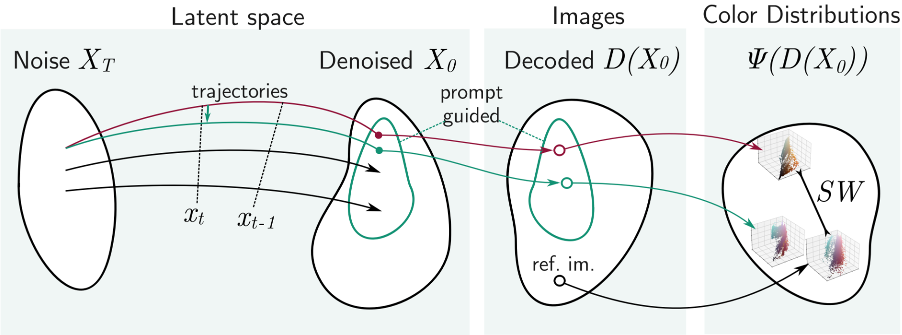
- However, the computational cost of the Wasserstein distance, particularly in high dimensions, can be prohibitive.
- To address these limitations, the sliced Wasserstein distance was introduced. It simplifies the computation by projecting high-dimensional distributions onto lower-dimensional subspaces, where the Wasserstein distance can be more easily computed.

Formally, sliced p-Wasserstein distance is defined as:

$$SW_p(\pi_0, \pi_1) = \left(\int_{\mathbb{S}^{d-1}} W_p^p(P_\theta \pi_0, P_\theta \pi_1)\, d\theta \right)^{1/p}, \tag{6}$$

where $\mathbb{S}^{d-1}$ is the unit sphere in $\mathbb{R}^d$ with $\int_{\mathbb{S}^{d-1}} d\theta = 1$ and P_θ is the linear projection onto a one-dimensional subspace defined by a vector unit θ.

Sliced Wasserstein Guidance



For two probability distributions π_0 and π_1 on $\mathbb{R}$, with respective CDFs $F_{\pi_0}(x)$ and $F_{\pi_1}(x)$, the Wasserstein distance $W_1(\pi_0, \pi_1)$ is given by:

$$W_1(\pi_0, \pi_1) = \int_{-\infty}^{\infty} |F_{\pi_0}(x) - F_{\pi_1}(x)| \, dx, \tag{7}$$

which represents the integral of the absolute difference between two CDFs over the real line.

Algorithm 3 Color-Conditional Generation with Sliced Wasserstein

Initialize latent vector $x_T \sim \mathcal{N}(0, I)$
for $t = T$ to 1 **do**
  $u \leftarrow \mathbf{0}$ ▷ Initialize control vector
  for $j = 1$ to M **do**
    $\hat{x}_t \leftarrow x_t + u$
    Get prediction of last latent $x_0 \leftarrow \mathrm{DDIM}(t, \hat{x}_t)$
    Compute $\hat{x_0} \leftarrow \mathrm{VAE}(x_0)$ ▷ Decode latent to image
    for $k = 1$ to K **do** ▷ Sliced Wasserstein
      Rotate distributions with random matrix R
      Update loss $\mathcal{L} \leftarrow \mathcal{L} + \sum |\mathrm{cdf}_x - \mathrm{cdf}_y|$
    end for
    Update control vector $u \leftarrow u - \nabla_u \mathcal{L}(u)$
  end for
  Update latent $x_t^* \leftarrow x_t + u$
  Get next latent $x_t \leftarrow \mathrm{DDIM}(t, x_t^*)$
end for

Computing the CDF and Sliced Wasserstein Distance

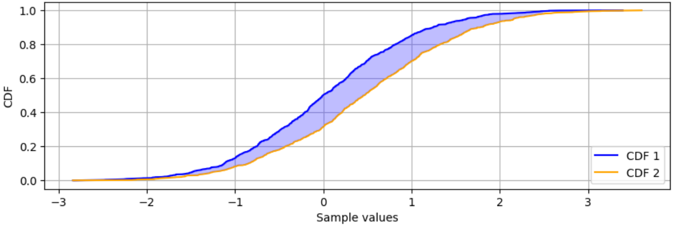
- Given a set of sorted samples $\{x_{(i)}\}_{i=1}^{n}$, the CDF value for each sample $x_{(i)}$ is computed as:

$$\mathrm{CDF}(x_{(i)}) = \frac{i}{n},$$

this approach provides a CDF which is differentiable with respect to input samples $x_{(i)}$ and can be used in optimization.

- For two probability distributions π_0 and π_1, with respective CDFs $F_{\pi_0}(x)$ and $F_{\pi_1}(x)$ the Wasserstein distance $W_1(\pi_0, \pi_1)$ is given by:

$$W_1(\pi_0, \pi_1) = \int_{-\infty}^{\infty} |F_{\pi_0}(x) - F_{\pi_1}(x)| \, dx$$



Evaluation Metrics

We compare our algorithm with other methods using three different metrics.

1. To measure a stylizing strength we calculate Wasserstein-2 distance between color distributions.
2. CLIP-T, is a cosine similarity between CLIP representations of a text prompt and an image generated from this prompt. In other words, CLIP-T score indicates whether a modified sampling process still follows an initial text prompt.
3. CLIP-IQA, a cosine similarity between a generated image and pre-selected anchor vectors, defining a "good-looking" pictures. CLIP-IQA measures an overall quality of pictures.

The experiments are conducted on a set of 1000 images generated with Dreamshaper-8, a StableDiffusion-based model, from a set of prompts taken from ContraStyles dataset.

Baselines

Generate a text conditional image and perform color transfer. Color transfer baselines

- Histogram matching
- PhotoWCT2
- Monge Kantorovich Linear (MKL)
- WCT2
- PhotoNAS
- Modulated Flows

Color conditional ControlNet

- ControlNet Colorcanny

Style Transfer baselines

- IP-Adapter
- InstantStyle
- RB-Modulation

Experimental Results: qualitative evaluation for SDXL

Experimental Results: qualitative evaluation for SD-1.5

A masterpiece in the form of a wood forest world inside a beautiful miniature	Magic, dark and moody landscape, in Gouache Style, Watercolor	retro-futurism anime castle on a mountain in clouds with lots of details	cute cat, smooth, sharp focus, cinematic lightning	Reference image

SW-Guidance is compatible with ControlNets

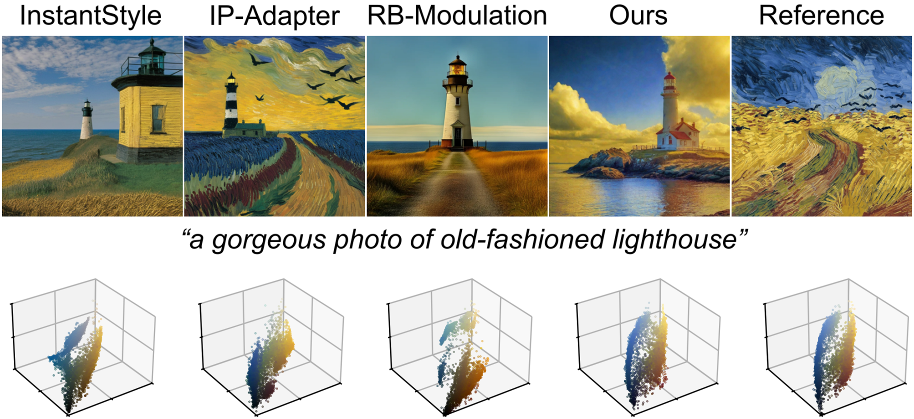
Note that the prompt is ignored and color distribution differs from the refence.

SW-Guidance is compatible with ControlNets

SW-Guidance reference could be just a palette of colors

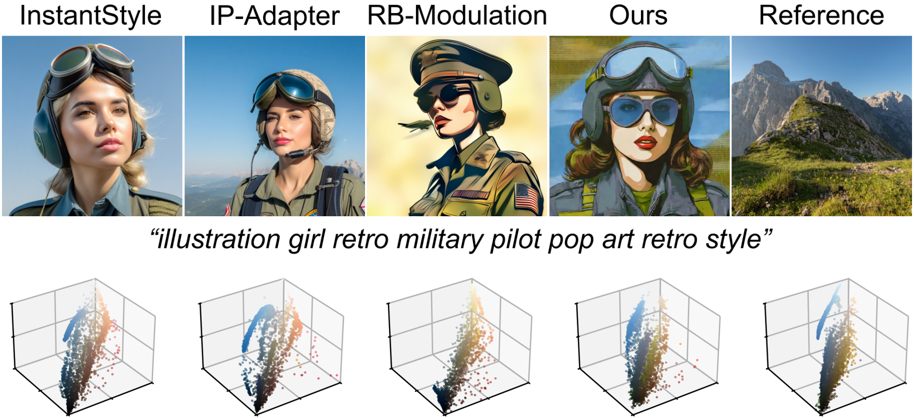
Comparison with Stylization Methods

Compared to stylizers, SW-Guidance achieves tighter palette matching without importing unwanted semantic/style patterns.



Comparison with Stylization Methods

Compared to stylizers, SW-Guidance achieves tighter palette matching without importing unwanted semantic/style patterns.



Experimental Results: quantitative evaluation

Wasserstein-2 distance↓	
Algorithm	mean ± std
SW-Guidance (ours)	0.033 ± 0.010
hm-mvgd-hm (Hahne, 2021)	0.057 ± 0.037
hm (Gonzales, 1977)	0.090 ± 0.057
PhotoWCT2 (Chiu, 2022)	0.109 ± 0.049
ModFlows (Larchenko, 2024)	0.118 ± 0.049
Colorcanny (ghoskno, 2023)	0.118 ± 0.051
MKL (Pitié, 2007)	0.127 ± 0.058
mvgd (Hahne, 2021)	0.135 ± 0.056
CT (Reinhard, 2001)	0.141 ± 0.060
WCT2 (Yoo, 2019)	0.143 ± 0.056
PhotoNAS (An, 2020)	0.172 ± 0.057
InstantStyle (Wang, 2024)	0.176 ± 0.086

Experimental Results: quantitative evaluation

Content scores ↓		
Algorithm	CLIP-IQA	CLIP-T
InstantStyle Wang et al.	0.332 ± 0.082	0.238 ± 0.056
PhotoNAS An et al. 2020	0.288 ± 0.088	0.259 ± 0.049
SW-Guidance (ours)	0.222 ± 0.089	0.262 ± 0.051
hm Gonzales and Fittes 1977	0.205 ± 0.091	0.270 ± 0.050
Colorcanny ghoskno 2023	0.195 ± 0.080	0.260 ± 0.053
ModFlows Larchenko et al. 2024	0.193 ± 0.088	0.269 ± 0.050
mvgd Hahne and Aggoun 2021	0.188 ± 0.088	0.270 ± 0.051
MKL Pitié and Kokaram 2007	0.185 ± 0.087	0.270 ± 0.051
CT Reinhard et al. 2001	0.183 ± 0.087	0.271 ± 0.051
WCT2 Yoo et al. 2019	0.182 ± 0.083	0.276 ± 0.050
PhotoWCT2 Chiu and Gurari 2022	0.180 ± 0.085	0.262 ± 0.053

Conclusion and Discussion

- We conclude that SW-Guidance achieves state-of-the-art results for color-conditional generation.
- The results show a significant improvement in color similarity to the reference palette compared to color transfer-based and stylization baselines, while maintaining semantic coherence and alignment with text prompts.
- The method also is applicable to general distribution-conditional generation tasks.

References I

An, Jie et al. (2020). "Ultrafast photorealistic style transfer via neural architecture search". In: Proceedings of the AAAI Conference on Artificial Intelligence. Vol. 34. 07, pp. 10443–10450.

Chiu, Tai-Yin and Danna Gurari (2022). "Photowct2: Compact autoencoder for photorealistic style transfer resulting from blockwise training and skip connections of high-frequency residuals". In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 2868–2877.

ghoskno (2023). Color-Canny ControlNet. https://huggingface.co/datasets/ghoskno/laion-art-en-colorcanny.

Gonzales, Rafael C and BA Fittes (1977). "Gray-level transformations for interactive image enhancement". In: Mechanism and Machine theory 12.1, pp. 111–122.

References II

- Hahne, Christopher and Amar Aggoun (2021). "PlenoptiCam v1.0: A Light-Field Imaging Framework". In: IEEE Transactions on Image Processing 30, pp. 6757–6771. DOI: 10.1109/TIP.2021.3095671.
- Larchenko, Maria et al. (2024). "Color Style Transfer with Modulated Flows". In: ICML 2024 Workshop on Structured Probabilistic Inference and Generative Modeling.
- Pitié, François and Anil Kokaram (2007). "The linear monge-kantorovitch linear colour mapping for example-based colour transfer". In: 4th European conference on visual media production. IET, pp. 1–9.
- Reinhard, Erik et al. (2001). "Color transfer between images". In: IEEE Computer graphics and applications 21.5, pp. 34–41.
- Wang, Haofan et al. (2024). "InstantStyle: Free Lunch towards Style-Preserving in Text-to-Image Generation". In: arXiv preprint arXiv:2404.02733.

References III

Yoo, Jaejun et al. (2019). "Photorealistic style transfer via wavelet transforms". In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 9036–9045.