

はじめに

- 近年、微分可能レンダラー [1] が登場したことで、ベクター生成にパラメータの最適化手法が使用される
- これにより、ラスター画像を扱う生成モデルの知識をベクター画像生成に転用することが可能になった
- この流れを汲んで、拡散モデルの知識を使用したベクター形式ロゴ画像生成を行う

目的

1. 画像全体の概形
 2. 芸術的なグラフィック
- を個別に制御する **ベクター形式ロゴ画像** の生成

手法

関連研究

1. Tone Preserving Loss [2] \mathcal{L}_{tone}

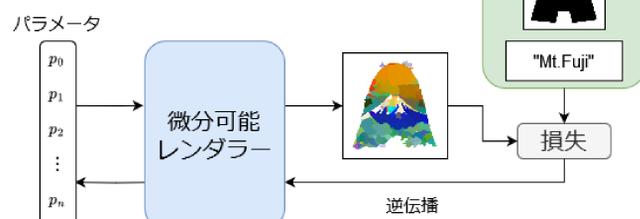
入出力間のローパスフィルタ適用後画像の最小二乗誤差
入力画像の範囲に表現の領域を制限する

2. Score Distillation Sampling Loss [3] \mathcal{L}_{LSDS}

拡散モデル (Stable Diffusion) の知識を蒸留することで
入力テキストに示されるコンテンツを反映する

手法

- 閉じたベジエ曲線により画像を表現
- 各ベジエ曲線はそれぞれパラメータとして、座標、色、不透明度を保有する [4]
- ベジエ曲線のパラメータを勾配降下法により最適化

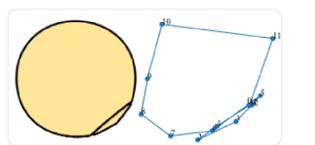


損失関数

3. Radiation Loss \mathcal{L}_{rad}

ベジエ曲線を明示的に監視して、出力の品質を保つ

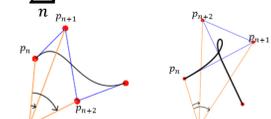
自己交差問題
最適化の過程でパスが自身と交差する
ラスター空間のみを損失に使用するため



自己交差問題の例 ([5] より引用)

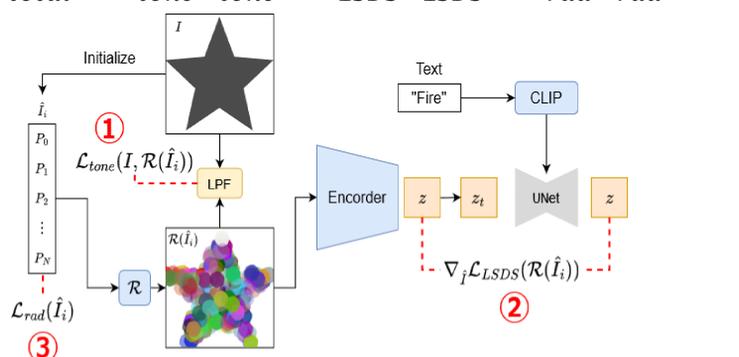
パラメータを幾何的に制御
中心点に対して一方向に
座標点を配置する

$$\mathcal{L}_{rad} = \sum_n \text{ReLU}(\angle p_n c p_{n+1} - \angle p_n c p_{n+2})$$



Total Loss

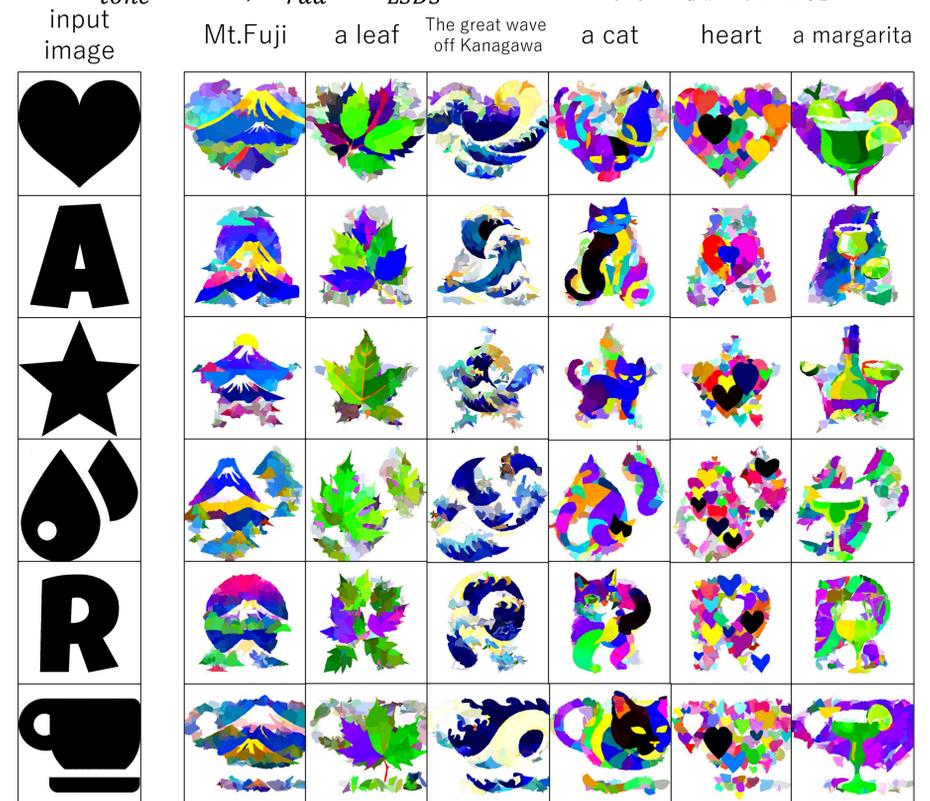
$$\mathcal{L}_{total} = \lambda_{tone} \mathcal{L}_{tone} + \lambda_{LSDS} \mathcal{L}_{LSDS} + \lambda_{rad} \mathcal{L}_{rad}$$



実験

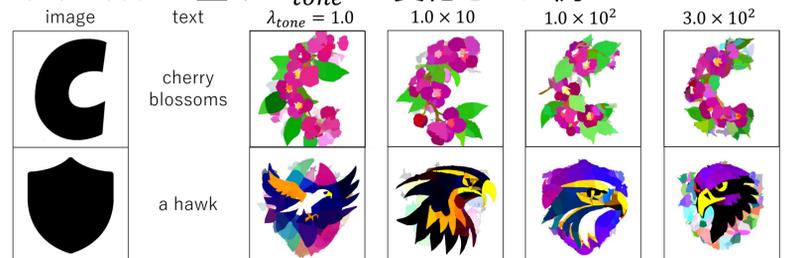
1. 生成実験

- 入力プロンプトは、" a logo of {concept}. minimal flat 2d vector. lineal color. trending on artstation."
- $\lambda_{tone} = 200$, $\lambda_{rad} = \lambda_{LSDS} = 1$ で1000回の最適化を行う

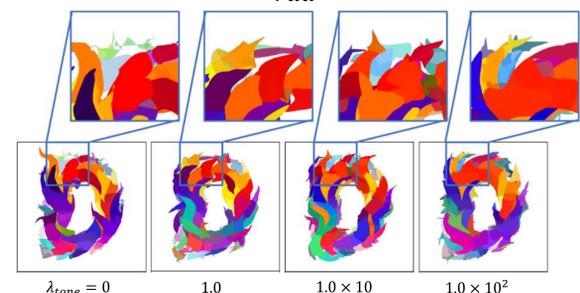


2. 重みを変化させた場合

- Tone Loss の重み λ_{tone} を変化させた例



- Radiation Loss の重み λ_{rad} を変化させた例



おわりに

- 画像とテキストを入力とした、ロゴ画像を生成するための最適化フレームワークを提案
- 目的に叶う出力が得られた一方、先行研究 [3] より出力の品質を向上させるには至らなかった



VectorFusion の出力例 ([3] より引用)

今後の展望

- 領域による形状の制限ではなく、直接拡散モデルのサンプリング結果の形状を操作する
- Radiation Loss のさらなる改善

[1] Li, T., Lukáč, M., M. G. and Ragan-Kelley, J.: Differentiable Vector Graphics Rasterization for Editing and Learning, SIGGRAPH, (2020).

[2] Iluz, S., Vinker, Y., Hertz, A., Berio, D., Cohen-Or, D. and Shamir, A.: Word-As-Image for Semantic Typography, SIIGGRAPH, (2023).

[3] Jain, A., Xie, A. and Abbeel, P.: VectorFusion: Text-to-SVG by Abstracting Pixel-Based Diffusion Models, CVPR, (2023).

[4] Frans, K., Soros, L. and Witkowski, O.: CLIPDraw: Exploring Text-to-Drawing Synthesis through Language-Image Encoders, NeurIPS, (2022).