

実写画像に基づく画風を考慮したアニメ背景画像生成システム

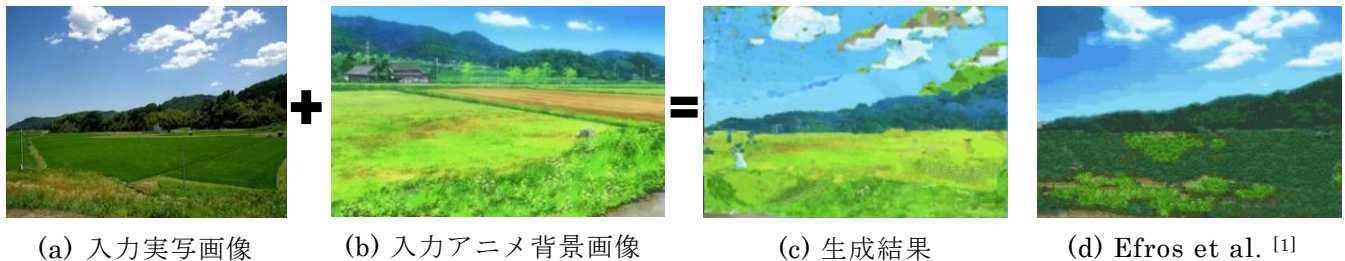
山口 周悟[†] 古澤 知英[†] 福里 司[†] 森島 繁生^{††}[†]早稲田大学, ^{††}早稲田大学理工学術院総合研究所 / JST CREST

図 1: 生成結果と既存手法の比較

1. はじめに

漫画やアニメ作品では、作品の世界観を表現するために背景を詳細に描きこむ傾向にある。しかし、アーティスト独自の色使い、塗り(ベタ塗り, 厚塗り, 水彩塗り)を表現する必要があるため、多くの技術と手間を要する。実際の風景とアニメ作品の違い(色使い, 塗り等)を分析し、転写することができれば、任意の画像をアニメ風に加工・デザインすることが可能となる。そこで本稿では、アニメ背景の画風を実写画像に転写することを研究目的とする。

絵画のような手描き画像の画風を実写画像に転写する先行研究として、Efronら^[1]は入力画像の輝度値を基に、手描き画像をパッチ単位で転写する手法を提案した。また、Hertzmannら^[2]は教師学習ペア(実写画像と手描き画像)の間に見られるテクスチャ特徴の対応関係を、任意の実写画像に転写する手法を提案した。両手法に共通する問題点として、入力手描き画像のテクスチャ情報が大きく損なわれていることがあげられる。問題の原因としては、両手法において対応パッチの探索は画像全体に対して行われるため対応先が不連続になってしまい、転写結果が乱雑になっているものと考えられる。アニメ背景はテク

スチャが鮮明である傾向を持つため、この問題を解決する必要がある。

そこで我々は、画像を領域分割し、領域ごとの対応関係を自動取得する手法を提案する。さらに各領域においてテクスチャ転写を行うことで、入力アニメ背景画像のテクスチャが保存された新たなアニメ背景画像を自動生成することが可能となる。

2. 提案手法

2.1. 領域分割と対応付け

入力された実写画像とアニメ背景画像それぞれに対してFelzenszwalb^[3]らの手法を用いて領域分割を行う(図2)。



図 2: 図 1(a)の領域分割結果

続いて、分割された実写画像と手描き画像の各領域を対応付ける。実写の各領域の $L^*a^*b^*$ 色空間におけるユークリッド距離が最小であるアニメの領域を対応先とする。しかし、異なる画像の領域を対応付けすることは困難である。例えば、全体的に明度の低い実写画像の各領域の対応先が全て、全体的に明度の高い手描き画像の最も暗い領域に集中してしまう、といった現象が発生する。

“Example-based cartoon background image creation from a photograph”

[†] Shugo YAMAGUCHI, Chie FURUSAWA, Tsukasa FUKUSATO, Waseda University

^{††} Shigeo Morishima, Waseda Research Institute for Science and Engineering/JST

そこで、対応付けを行う前に、2枚の入力画像の色味を全体的にフィッティングさせる。

Reinhardら^[4]は、2枚の画像のヒストグラムをフィッティングさせることにより、色味を転写する手法を提案した。実写とアニメの輝度値をそれぞれ L_r 、 L_a 、それぞれの分散を σ_r 、 σ_a 、変換後の実写の輝度値を L'_r とおくと、変換は以下の式に従う。

$$L'_r = \frac{\sigma_a}{\sigma_r} (L_r - \langle L_r \rangle) + \langle L_a \rangle$$

しかし、入力画像内に複数の異なる色合いの領域が存在する場合、自然な転写ができない問題がある(図 3(d))。そこで我々は、2枚の画像の輝度値を色相に基づいて6色のグループに分け、同色のグループに属する画素群に対してReinhardらの手法を適用し(図 3(c))、実写画像とアニメ背景画像の領域ごとの対応付けを行う。

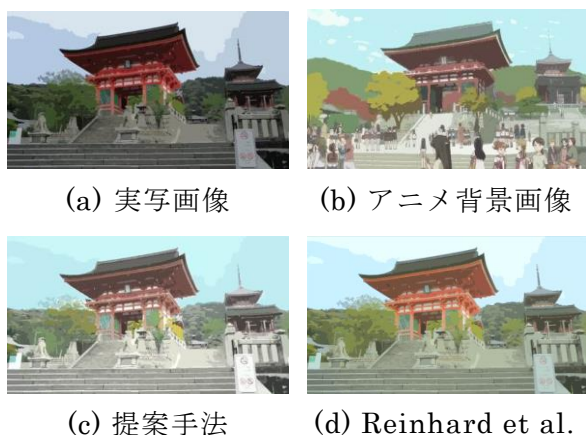
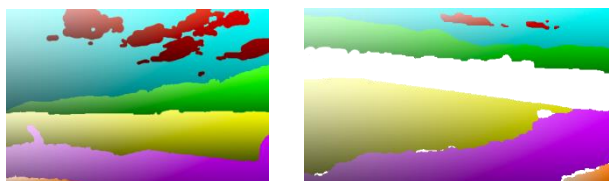


図 3: 色転写結果

2.2. テクスチャ転写

対応付けた領域に対し、Efrosらと同様に輝度値を基にパッチ単位でテクスチャ転写を行う。今回はBarnes^[5]らのPatch Matchを用いて高速なテクスチャ転写を行った。さらに、ベタ塗りのアニメ背景画像(各領域の輝度値が一定)のような各領域の輝度値を基にテクスチャ転写が困難な場合や、インタラクティブな編集を行うために、2.1章で生成したラベリング結果(図4)をユーザに提示し、各領域の輝度値を編集させることで、パッチ単位のテクス

チャ転写結果をインタラクティブに編集可能とする。



(a) ラベル画像(実写) (b) ラベル画像(アニメ)

図 4: ラベル画像

3. 結果

アニメ画像の画風を実写画像に転写した結果を図1(c)に示す。本手法の有効性を確かめるため、Efrosらの手法で転写した結果を図1(d)に示す。従来手法と比較して、よりアニメ背景の画風(色合い)を反映した結果が得られた。

4. まとめと今後の課題

本研究では、実写画像とアニメ背景画像の領域ごとの対応関係を基に、アニメ背景の画風(テクスチャ)を自動かつ高速転写する手法を提案した。

今後の課題として、各領域のテクスチャ方向やテクスチャスケールなどを考慮した領域対応付けやテクスチャ転写方法が挙げられる。さらに複数のアニメ背景画像を用いた転写手法を目指したい。

参考文献

- [1] Efros, A.A., and Freeman, W.T. "Image Quilting for Texture Synthesis and Transfer." *ACM SIGGRAPH*, 341-346, 2001.
- [2] Hertzmann, A. et al. "Image Analogies." *ACM SIGGRAPH*, 327-340, 2001.
- [3] Felzenszwalb, Pedro F. et al., "Efficient graph-based image segmentation." *International Journal of Computer Vision* 59.2,167-181, 2004.
- [4] Reinhard, E, et al. "Color transfer between images." *IEEE Computer graphics and applications*, 21.5, 34-41, 2001.
- [5] Barnes, C, et al. "PatchMatch: A Randomized Correspondence Algorithm for Structural Image Editing." *ACM Transactions on Graphics*, 28.3, 24, 2009.