

非反復主要色決定アルゴリズムに基づくピクセル アートの生成

高橋, 龍ノ介
九州大学大学院芸術工学府

井上, 光平
九州大学大学院芸術工学研究院 : 准教授

原, 健二
九州大学大学院芸術工学研究院 : 准教授

平岡, 透
長崎県立大学情報システム学部 情報システム学科 : 教授

<https://hdl.handle.net/2324/2928835>

出版情報 : The journal of the Institute of Image Information and Television Engineers. 74 (3), pp.597-600, 2020-04-27. The Institute of Image Information and Television Engineers

バージョン :

権利関係 : 著作権は映像情報メディア学会に帰属しています。

非反復主要色決定アルゴリズムによるピクセルアートの生成

Generating Pixel Art by Noniterative Dominant Color Decision Algorithm

高橋 龍ノ介[†], 正会員 井上 光平[†], 原 健二[†], 正会員 平岡 透^{††}

Ryunosuke Takahashi[†], Kohei Inoue[†], Kenji Hara[†] and Toru Hiraoka^{††}

あらまし 高解像度の画像から低解像度のピクセルアートを生成する方法を提案する。まず、高解像度画像上に正方形のブロックをすき間や重なりなく敷き詰めるようにして、高解像度画像をブロック分割する。次に、各ブロックをピクセルアートのピクセルとみなし、そこに単一の色を割り当てることによってピクセルアートを構成する。各ブロックの色を決定するために、計算効率のよい非反復アルゴリズムを提案する。提案法によって、従来法よりも対象の形状の崩れやぼけが少ないピクセルアートが得られることを実験で確認する。

キーワード：ピクセルアート、非反復アルゴリズム、平均絶対誤差

1. ま え が き

ピクセルアートは、作家が画素単位で色を配置して作る画像であり、ドット絵¹⁾とも呼ばれて主に初期のコンピュータゲームで用いられた。現在でもオンラインゲームやスマートフォンアプリなど、限られたリソースを有効活用する環境では需要があり、ドット絵のレトロな雰囲気を好むファンも多く、シブヤピクセルアート 2019²⁾などのイベントも開催されている。また、ドット絵を集めたドットアート集³⁾やピクセルアート作家の作品集⁴⁾なども出版されており、ピクセルアートは芸術表現の一形態として認知されてきている。

ピクセルアートに関する研究も行われており、図 1 に示すような低解像度のピクセルアートに関する方法がいくつか提案されている。Han ら⁵⁾は、深層学習を用いて高解像度画像から低解像度のピクセルアートを生成する方法を提案した。ここでは、学習データとして入出力画像の対を用意することが困難であることに着目して、入出力画像が対

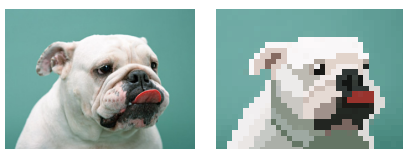
応付けられている必要のない教師なし学習法が提案されているが、学習に 900 枚のピクセルアートとイラストが使われており、学習データの収集に難点がある。鈴木ら⁶⁾は、単一オブジェクト写真からドット絵風キャラクタ画像を生成する手法を提案した。この手法は、輪郭線を途切れなく描画できるが、非等方性拡散フィルタ、 k -means 法、拡張 Canny edge detector、細線を優先した画像縮小法を利用しており、処理が複雑であるという難点がある。そこで本論文では、これらの難点を克服することを目的として、学習データが不要で、1 枚の入力画像のみから、簡単な処理でピクセルアートを生成する方法を提案する。

本論文では、画像のブロック分割と各ブロックの色の決定という 2 段階処理によるピクセルアート生成法において、色決定処理を非反復で効率的に行う方法を提案する。色決定処理には、鈴木らの手法⁶⁾でも使われている k -means 法をはじめとするクラスタリングアルゴリズムが用いられることが多いが、反復処理が必要であるため、計算に時間がかかるという問題がある。ここで提案する色決定アルゴリズムは、反復処理が不要であり、計算量が少ないという利点がある。イラストを用いた実験で、画像内容の形状の崩れや輪郭のぼけが抑えられることを確認し、平均絶対誤差による画質評価と、アンケート調査による主観評価を行う。

2. ピクセル色決定の非反復アルゴリズム

高解像度の画像を $F = [f_{ij}]$ とする。ここで f_{ij} は F の画素 (i, j) の値であり、 i と j の取り得る値は、 F の縦と横の画素数をそれぞれ m と n とし、 $i = 1, 2, \dots, m$ 及び $j = 1, 2, \dots, n$ とする。

まず、画像 F を $h \times h$ 画素からなるブロックに分割して、縦のブロック数を $M = \lfloor \frac{m}{h} \rfloor$ 、横のブロック数を $N = \lfloor \frac{n}{h} \rfloor$



(a) 犬の写真 (b) ピクセルアート

図 1: 写真 (a) を参考に手作業で作ったピクセルアート (b)

2019 年 8 月, 映像情報メディア学会年次大会で発表

2019 年 10 月 25 日受付, 2020 年 2 月 25 日再受付, 2020 年 4 月 7 日採録

[†]九州大学 大学院芸術工学研究院

(〒 815-8540 福岡市南区塩原 4-9-1, TEL 092-553-4512)

^{††}長崎県立大学 情報システム学部 情報システム学科

(〒 851-2195 西彼杵郡長与町まなび野 1-1-1, TEL 095-813-5160)

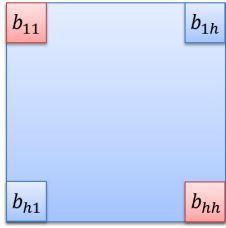


図 2: $h \times h$ 画素からなるブロック B_{kl}

とする．ここで $[\cdot]$ は床関数を表す．すなわち，ブロックを画像の左上から敷き詰めていき，右端や下端の端数は切り捨てることにする．画像の中心を切り出したい場合には，左上の画素 $(1, 1)$ から $(\lfloor \frac{m-Mh}{2} \rfloor, \lfloor \frac{n-Nh}{2} \rfloor)$ だけ平行移動した画素を新たな左上画素とする画像に対して同様の処理を行えばよい．

次に，各ブロックをひとつの画素に置き換えることによって低解像度のピクセルアートを生成する．ここでは各画素の色を決めることが主要な問題になる．求めるピクセルアートを $P = [p_{kl}]$ とする．ここで p_{kl} は P の画素 (k, l) の値であり，添え字の k と l は $k = 1, 2, \dots, M$ 及び $l = 1, 2, \dots, N$ の値を取る．ピクセルアート P の画素 (k, l) に対応する，画像 F 上のブロックを

$$B_{kl} = \begin{bmatrix} f_{h(k-1)+1, h(l-1)+1} & \cdots & f_{h(k-1)+1, h(l-1)+h} \\ \vdots & \ddots & \vdots \\ f_{h(k-1)+h, h(l-1)+1} & \cdots & f_{h(k-1)+h, h(l-1)+h} \end{bmatrix} \quad (1)$$

とし，これを

$$B_{kl} = \begin{bmatrix} b_{11} & \cdots & b_{1h} \\ \vdots & \ddots & \vdots \\ b_{h1} & \cdots & b_{hh} \end{bmatrix} = [b_{i_k, j_l}] \quad (2)$$

と略記する (図 2)．ここで b_{i_k, j_l} はブロック B_{kl} の画素 (i_k, j_l) の値であり，添え字の i_k と j_l は $i_k = 1, 2, \dots, h$ 及び $j_l = 1, 2, \dots, h$ の値を取る．このブロック上で，空間的に最も離れている画素対 (b_{11}, b_{hh}) (図 2 のピンクの画素) と (b_{1h}, b_{h1}) (同図のブルーの画素) の色差を比較し，色差が大きいほうの画素対を選び，それを (u, v) とする．次に，ブロック内の各画素 b_{i_k, j_l} を u と v のうち，色差が小さいほうに分類する．すると，ブロック内の全画素は u または v のグループに二分される． u のグループに属す画素の集合を S_u とし， v のグループに属す画素の集合を S_v とすると，次式のように，要素数が大きいほうの集合の平均色をピクセルアートの画素値とする．

$$p_{kl} = \begin{cases} \frac{1}{|S_u|} \sum_{p \in S_u} p, & \text{if } |S_u| \geq |S_v| \\ \frac{1}{|S_v|} \sum_{p \in S_v} p, & \text{otherwise} \end{cases} \quad (3)$$

ここで $|S_u|$ と $|S_v|$ はそれぞれ集合 S_u と S_v の要素数を

Algorithm 1 Noniterative Dominant Color Decision

Require: A block B_{kl}

Ensure: A color p_{kl}

```

1:  $[h, h] = \text{size}(B_{kl})$ 
2: if  $D(b_{11}, b_{hh}) > D(b_{1h}, b_{h1})$  then
3:    $u \leftarrow b_{11}$ 
4:    $v \leftarrow b_{hh}$ 
5: else
6:    $u \leftarrow b_{1h}$ 
7:    $v \leftarrow b_{h1}$ 
8: end if
9:  $S_u = S_v = \emptyset$ 
10: for  $k \leftarrow 1$  to  $h$  do
11:   for  $l \leftarrow 1$  to  $h$  do
12:     if  $D(b_{kl}, u) < D(b_{kl}, v)$  then
13:        $S_u.append(b_{kl})$ 
14:     else
15:        $S_v.append(b_{kl})$ 
16:     end if
17:   end for
18: end for
19: if  $|S_u| \geq |S_v|$  then
20:    $p_{kl} \leftarrow \frac{1}{|S_u|} \sum_{p \in S_u} p$ 
21: else
22:    $p_{kl} \leftarrow \frac{1}{|S_v|} \sum_{p \in S_v} p$ 
23: end if

```

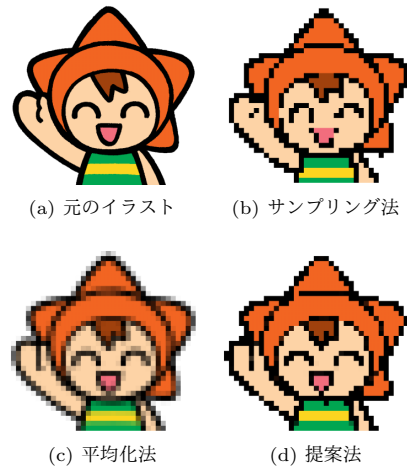


図 3: ピクセルアートの生成例 1

表す．

この非反復主要色決定アルゴリズムの擬似コードを **Algorithm 1** に示す．その中で， $\text{size}(B_{kl})$ はブロック B_{kl} のサイズを返す関数であり， $D(\cdot, \cdot)$ は二つの色の色差を計算する関数であり， $S_u.append(b_{kl})$ は色 b_{kl} を集合 S_u に追加することを表す．この擬似コードが示すように，本アルゴリズムは，ブロックのサイズが同じであれば，同じ計算量でピクセル色を決定できる．

以上のような処理をすべてのブロック $\{B_{kl}\}$ に対して行い，ピクセルアートのすべての画素の色 $\{p_{kl}\}$ を決定する．

3. 実験例

高解像度のイラストからピクセルアートを生成した例を図 3 に示す．図 3(a) は『キャラクターイラストいいね』⁷⁾ で公開されている 656×696 画素 ($m = 656, n = 696$) の

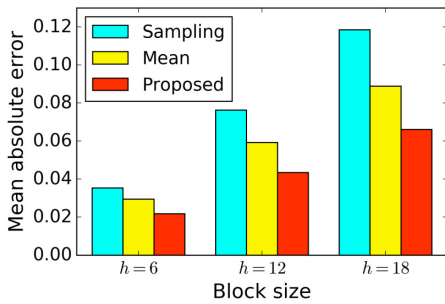


図 4: 平均絶対誤差の比較例 1

画像である。それを 18×18 画素のブロック ($h = 18$) に分割し、各ブロックを一つの画素に置き換えることによってピクセルアートを生成した。生成したピクセルアートの画素数は 36×38 になる ($M = 36, N = 38$)。図 3(b) は各ブロックの左上の画素をサンプリングした結果であり、左右の目の形の違いや右手の変形などが目立つ。図 3(c) は各ブロックの平均色を出力した結果であり、異なる色が混ざりあって輪郭がぼけている。図 3(d) は提案法による結果であり、形状の崩れや輪郭のぼけが抑えられている。

次に、平均絶対誤差 (mean absolute error: MAE) を用いて画質を評価した。元の高解像度画像 F とそのピクセルアート P は大きさが異なるため、そのままでは MAE を計算できない。そこで、 P の各画素の値を、それに対応するブロックの全画素にコピーして、 P の拡大版 \tilde{P} を作る。 F をブロック分割した際に、右端や下端に端数が生じる場合、 \tilde{P} は F よりも小さくなるので、 F の端数を切り捨てて \tilde{P} と同じ大きさの \tilde{F} を作る。同サイズの画像 $\tilde{F} = [\tilde{f}_{ij}]$ と $\tilde{P} = [\tilde{p}_{ij}]$ から、次式の MAE を計算する。

$$\text{MAE}(\tilde{F}, \tilde{P}) = \frac{1}{MNh^2} \sum_{i=1}^{Mh} \sum_{j=1}^{Nh} \|\tilde{f}_{ij} - \tilde{p}_{ij}\|_1 \quad (4)$$

図 4 に MAE の計算結果を示す。グラフの縦軸は MAE の値であり、横軸にはブロックサイズを決めるパラメータ h の値を、 $h = 6, 12, 18$ と並べている。シアン色の棒はサンプリング法を表しており、MAE 値が最も大きい。黄色は平均化法であり、サンプリング法よりも MAE 値が小さい。赤色は提案法であり、MAE 値が最も小さい。

他のイラストからピクセルアートを生成した結果を図 5 に示す。左の列から順に、元のイラスト、サンプリング法、平均化法、提案法による結果を示す。これらの例でも、図 3 と同様の結果が得られた。

図 5 の画像で MAE を計算した結果を図 6 に示す。この図の横軸は、図 5 の 5 枚のイラストの名前である。この例でも、提案法の MAE 値が最小となった。

画質評価に MAE を用いた理由を説明するために、平均 2 乗誤差 (MSE) との比較を行った。図 3 に示した画像で、MSE と MAE を計算した結果を図 7 に示す。縦軸に評価値を、横軸に 3 種の方法を表示している。シアン色の棒は MSE



図 5: ピクセルアートの生成例 2

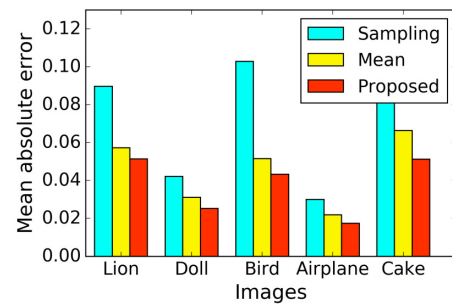


図 6: 平均絶対誤差の比較例 2

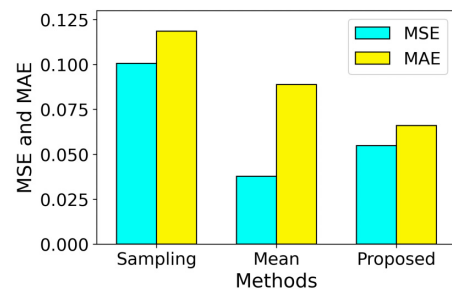


図 7: 平均 2 乗誤差と平均絶対誤差の比較

を表し、黄色の棒は MAE を表す。MSE 値は平均化法が最小になり、MAE 値は提案法が最小になった。後述の主観評価では、平均化法よりも提案法のほうが好ましいという結果が得られており、ピクセルアートの評価には、MSE よりも MAE のほうが適していると考えられる。また、Hanら⁵⁾ が用いた目的関数の中で MAE が使われていることから、MAE の有用性が示唆される。

図 5 に示したピクセルアートを、順位法を用いて主観評価した結果を図 8 に示す。各イラストのピクセルアートを 20

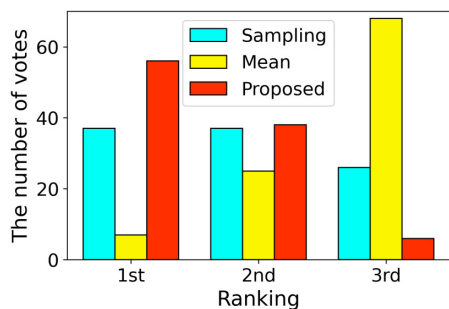


図 8: 主観評価の結果

名の被験者に見せ、ピクセルアートとして好ましいと思う順に、1~3の順位を付けてもらった。各画像はコンピュータ画面上に一辺5cm程度の大きさで提示し、視距離等は特に指定しなかった。図の縦軸は票数、横軸は順位である。提案法(赤)によるピクセルアートを1位とした人が56票と最も多く、サンプリング法(シアン)と平均化法(黄)はそれぞれ37票、7票であった。提案法によるピクセルアートが多くの人から好評価を得ていることが示された。また、提案法では、評価順位が上がると票数も増えるという傾向も見られた。一方、サンプリング法は順位間の差が小さく、評価が分かれた。その理由は、サンプリング法によるピクセルアートの成否が、元のイラストに依存するためであると推測される。平均法では、3位の票数が68票と最も多く、平均化によって生じるぼけがピクセルアートには適さないことが示唆される。

従来のピクセルアート生成方式では、例えば、Hanらの深層学習法⁵⁾のように数百枚の学習データが必要であったり、鈴木らのドット絵風キャラクタ画像生成法⁶⁾のように多段階の処理が必要であったり、実装が困難な方法が多い。それに対して提案法は、Algorithm 1に示したように簡単に実装でき、反復処理も含まないため、計算時間も短いという利点がある。欠点(克服できていない点)としては、実際のピクセルアートのように色数が制限されていない、対象の形状をデフォルメすることはできない、などが挙げられる。

提案法によるピクセルアートは、入力画像から自動的に生成されるものであり、そこにユーザの意図は反映されない。そのため、アートとしての完成度は充分であるとはいえないが、例えば、観賞に堪えるアート作品の下絵として利用することは可能であろう。

4. む す び

非反復の色決定アルゴリズムにより高解像度画像から低解像度のピクセルアートを求める方法を提案した。形状の崩れや輪郭のぼけを抑えたピクセルアートが得られることを実験で示し、平均絶対誤差による画質評価を行った。次の開発ステップとして、画像内容に応じたブロック分割法の検討、色数の制限による初期のゲーム表現、動画への拡

張などを計画している。

また、本色決定アルゴリズムでは、ブロック内の色を二つのグループに分けて多数派の平均色を選択するため、例えば、ブロック内に3色がほぼ同数あった場合、そのうちの2色の平均色が選択され、元の画像にない色が現れるおそれがあるが、メドイドやモードの色を選択するように変更することで、この問題は回避できると考えられる。さらに、文献6)のように対象の輪郭線を描画する場合、元画像に輪郭線が描かれているか否かによって処理方法が変わるなど、実装の複雑化が予想される。

謝辞 貴重なご意見を頂いた査読者に感謝します。また、主観評価にご協力頂いた皆様に感謝します。本研究はJSPS科
研費 JP19K12664, JP16H03019 の助成を受けたものです。

【文 献】

- 1) 中川悠京: ドット絵教室一描けば描くほどうまくなる!, エムディエヌコーポレーション(2018)
- 2) シンヤピクセルアート 2019. <https://pixel-art.jp/>
- 3) スクウェア・エニックス(編集): FF DOT. -The Pixel Art of FINAL FANTASY-, スクウェア・エニックス(2018)
- 4) グラフィック社編集部(編集): ピクセル百景 現代ピクセルアートの世界, グラフィック社(2019)
- 5) C. Han, Q. Wen, S. He, Q. Zhu, Y. Tan, G. Han, T.-T. Wong: "Deep Unsupervised Pixelization", ACM Trans. Graph., **37**, 6 (2018)
- 6) 鈴木和明, 齋藤豪, 張英夏, 近藤邦雄, 中嶋正之: "単一オブジェクト写真からドット絵風のキャラクタ画像を生成する手法の提案", 電子情報通信学会論文誌, **J91-D**, 12, pp. 2961-2972 (2008)
- 7) キャラクターイラストいいね(2015) <https://www.3scom.com/erasuto/>



たかはしりゅうのすけ
高橋 龍ノ介 1919年, 九州大学芸術工学部画像設計学科卒業。現在, 同大学大学院芸術工学府修士課程在籍中。デジタルアートに関する研究に従事。



いのうえ こうへい
井上 光平 1996年, 九州芸術工科大学画像設計学科卒業。2000年, 同大学大学院博士後期課程修了。同年, 九州芸術工科大学助手。2003年, 九州大学助教。現在, 同大学准教授。パターン認識, 画像処理, コンピュータビジョンに関する研究に従事。正会員。



はら けんじ
原 健二 1989年, 京都大学大学院工学研究科修士課程修了。同年, 武田薬品工業(株)入社。1991年, 福岡工業技術センター入所。2001年, 東京大学生産技術研究所協力研究員。現在, 九州大学大学院芸術工学研究科准教授。博士(工学)。コンピュータビジョン, 画像処理に関する研究に従事。



ひらおか とる
平岡 透 1995年, 九州芸術工科大学芸術工学部画像設計学科卒業。1997年, 同大学大学院芸術工学研究科博士前期課程修了。2005年, 同大学大学院芸術工学研究科博士後期課程修了。2011年, 大分工業高等専門学校情報工学科准教授。現在, 長崎県立大学情報システム学部教授。博士(工学)。画像処理, 地理情報処理に関する研究に従事。正会員。