

## 誤差拡散法を応用した非写実的白黒画像生成に関する研究

衛藤 宗一郎<sup>†</sup> 長橋 宏<sup>‡</sup> 齋藤 豪<sup>†</sup> 張 英夏<sup>†</sup> 中嶋 正之<sup>†¶</sup>

<sup>†</sup> 東京工業大学 大学院 情報理工学研究科

<sup>‡</sup> 東京工業大学 大学院理工学研究科 像情報工学研究施設

<sup>¶</sup> 国立情報学研究所

{eto@img.cs, longb@isl, suguru@img.cs, chang@img.cs, nakajima@img.cs}.titech.ac.jp

本稿では、現実の描画技法を模倣するのではないアプローチにより、非写実的白黒画像を描画する手法を、白黒二値化画像生成で用いられる誤差拡散法を応用して実現する。画像の色空間分割法、画像上の非等方拡散法を誤差拡散法と組み合わせ、誤差拡散法で通常は点を配置する処理の代わりに線を描画する処理を行うことで、カラー画像を非写実的白黒画像へと変換する。処理の手順は、はじめに入力画像の各分割領域に対し線の描画パラメータの設定を行い、次に、擬似輪郭などの不自然な部分の発生を防ぐため、線の描画パラメータに対し非等方拡散法を適用する。そして、求めた線の描画パラメータを用いて、誤差拡散法によって線を描画する。

## A generating method for non-photorealistic black-and-white image based on the error diffusion method

Souichiro Eto<sup>†</sup> Hiroshi Nagahashi<sup>‡</sup> Suguru Saito<sup>†</sup> Youngha Chang<sup>†</sup>  
Masayuki Nakajima<sup>†¶</sup>

<sup>†</sup>Graduate School of Information Science and Engineering, Tokyo Institute of Technology.

<sup>‡</sup>Graduate School of Science and Engineering, Imaging Science and Engineering Laboratory

<sup>¶</sup>National Institute of Informatics

{eto@img.cs, longb@isl, suguru@img.cs, chang@img.cs, nakajima@img.cs}.titech.ac.jp

### Abstract

In this paper, we propose a method to create non-photorealistic black and white images based on the error diffusion method. Our goal is to create a variety of new artistic styles, rather than to imitate real-world drawing styles. To achieve this, our algorithm segments the input image, allows user to set drawing-style parameters for each segment, and then adjusts those parameters not to create any artifacts, e.g. pseudo-contours. Finally, the error diffusion method is applied with those parameters. We present a variety of results, arguing that our algorithm can generate a variety of new styles.

# 1 はじめに

グレースケール画像による表現は、色数が少ないために、写実的なカラー画像で表現される情報量よりも限られた情報しか表す事が出来ない。しかし、これにより、写実的な画像では難しい表現（非写実的な表現）が可能となる。そのような背景から、ペン画や鉛筆画による非写実的表現については、過去に様々な研究が行われてきた。だが、これらの研究は、従来の手法をコンピュータ上で模倣する事をテーマにしたものが多い。そこで、単に模倣するのではなく、従来の画材によるストローク描画とは異なるストロークの配置手法を導入すれば、従来のものとは異なる表現が可能になると考えられる。

そこで本論文では、誤差拡散法による線画生成法 [8] を拡張した白黒線画生成手法を提案する。本手法では、ストロークの配置に多様性を生じさせる為に、画像の色空間領域分割および非等方性拡散法を用いる。

# 2 関連研究

グレースケール画像による非写実的表現の代表例として、ペン画や鉛筆画が挙げられる。ペン画は、写実的な画像とは異なる印象の画像を生成するための表現手法の一つであり、線の描き方によって生成される画像の印象がどのように変化するかについては、昔から様々な考察が行われてきた。また、近年、コンピュータを用いたペン画の生成に関する研究は数多くなされている。

Winkenbach らは、3次元のポリゴンモデルを用いる事でペン画を自動生成する手法を提案している [1]。この手法では、ストロークによる明暗やテクスチャ性を表現するために、ストロークテクスチャを導入している。

また、ポリゴンモデルだけではなく、B-Spline 曲面、NURBS 曲面、回転体などのパラメトリックな自由曲面モデルから画像を生成する研究も存在する [2]。

Salisbury らは、ペン画をインタラクティブに生成するシステムについて提案している [3]。このシステムでは、画像生成の時に、ユーザがストロークテクスチャを「貼り付ける」よう指定出来るインターフェイスを提供している。

また、Salisbury らは、この手法を元にし、方向性を持ったテクスチャの表現も出来るように改良を加えている [4]。ここでは、テクスチャの方向もユーザが制御

することが出来る。これにより、物体の表面の性質を考慮したペン画を生成する事が出来、[3] よりも魅力的な表現が可能となっている。

その他にも、ペン画の生成時に解像度の違いによって印象が異なってしまうという問題を解決する手法 [5] や、アニメーションを行うための手法 [6] など、従来の多くの研究ではストロークテクスチャを用いることによってペン画を可能にしていた。

一方、Mao らは、ペン画を模した画像をあらかじめテクスチャを用意するのではなく、LIC (Line Integral Convolution) 法を用いて作り上げている [7]。

また、濃淡画像から、誤差拡散法により線を直接敷き詰めていく線画生成手法も存在する [8]。この手法では、全体の画像に対し、単一方向で長さが一定の単調なストロークを適用しているため、ごく限られた表現力しか持ち合わせていない。しかし、この手法は、画素の白黒の値をアルゴリズムによって決定しているため、あらかじめテクスチャを用意する必要が無いという利点がある。

そこで、本論文では、[8] の手法を拡張し、線のパラメータを変化させ、新たな白黒画像を出力する手法について提案する。

# 3 理論と具体的な手法

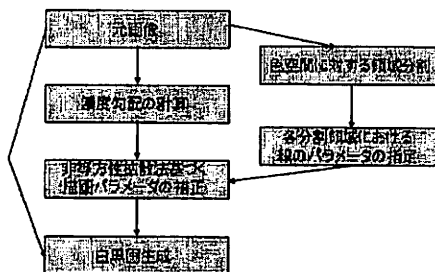


図 1: アルゴリズムの流れ

本提案アルゴリズムの流れを図 1 に示す。

### 3.1 色空間に対する領域分割

本アルゴリズムでは、描画するストロークに多様性を持たせるため、CIE LAB 色空間において領域分割を行う。ここでは、領域分割の手法として K-means 法を用いる。そして、各領域ごとに異なる線の長さ、幅、角度の指定を行う。

### 3.2 濃度勾配の計算

本提案手法では、ユーザにより指定された方向に線を描画するモードと、濃度勾配を計算し、それに直交する方向へ描画するモードがある。

画素  $D(x, y)$  における濃度勾配ベクトルの方向  $\phi(x, y)$  は、以下の様に求められる。

$$g_x(x, y) = \partial(G(x, y) * D_{mono}(x, y)) / \partial x \quad (1)$$

$$g_y(x, y) = \partial(G(x, y) * D_{mono}(x, y)) / \partial y \quad (2)$$

$$\phi(x, y) = \tan^{-1}(g_y(x, y) / g_x(x, y)) \quad (3)$$

ここで、 $D_{mono}(x, y)$  は、入力画像の画素  $(x, y)$  における輝度値を示し、 $g_x(x, y)$ 、 $g_y(x, y)$  は、それぞれ画素  $(x, y)$  における水平および垂直方向への一次微分値を表す。本研究では、微分値の計算に Sobel フィルタを用いた。ただし、ノイズによる問題を軽減するために、ガウス関数  $G$  を入力画像に畳み込んだ後に微分を行っている。

### 3.3 線のパラメータの指定

3.1 節で求めた色空間領域分割に基づき、画像上の各画素を  $k$  個のカテゴリに分類する。ユーザは、各カテゴリに対し、線の長さ  $l$  および太さ  $w$  という 2 つのパラメータを指定することが出来る。線の方向に関しては、前述の二つのモードのどちらか選択されているかにより、設定の仕方が変わる。すなわち、方向固定モードの場合はユーザの設定した方向に、濃度勾配直交方向モードの場合は 3.2 節で計算された方向に設定される。

### 3.4 非等方性拡散法に基づくパラメータの補正

色がなだらかに変化している画像領域が色空間の分割結果によっては分断されることがある。そのような

場合、線のパラメータが急変する箇所では擬似輪郭が生じてしまう可能性がある。そこで、擬似輪郭が発生しそうな領域に対しては線パラメータの変化をなだらかにし、なおかつ、エッジの情報が失われないようにする事で、違和感の無い結果画像を生成する事が出来る。そのため、本稿では非等方性拡散法 [8] を用いる。

具体的には、以下の通りである。

各画素において、線の太さ  $w$ 、線を表すベクトル  $\vec{v}$  の  $x$  成分  $v_x = l \cos \theta$ 、 $y$  成分  $v_y = l \sin \theta$  ( $\theta = 2\phi$ ) を以下のように計算する。

$$w(x, y) := \frac{\sum_{i=-1}^1 \sum_{j=-1}^1 \alpha(x+i, y+j) w(x+i, y+j)}{\sum_{i=-1}^1 \sum_{j=-1}^1 \alpha(x+i, y+j)} \quad (4)$$

$$v_x(x, y) := \frac{\sum_{i=-1}^1 \sum_{j=-1}^1 \alpha(x+i, y+j) v_x(x+i, y+j)}{\sum_{i=-1}^1 \sum_{j=-1}^1 \alpha(x+i, y+j)} \quad (5)$$

$$v_y(x, y) := \frac{\sum_{i=-1}^1 \sum_{j=-1}^1 \alpha(x+i, y+j) v_y(x+i, y+j)}{\sum_{i=-1}^1 \sum_{j=-1}^1 \alpha(x+i, y+j)} \quad (6)$$

ここで、 $\alpha(x+i, y+j)$  は、エッジストップ関数を示す。この関数は、 $D(x, y)$  と  $D(x+i, y+j)$  の色差が小さければ大きい数字を、色差が大きければ小さい数字を返す。つまり、もし  $D(x, y)$  と  $D(x+i, y+j)$  の値が似ているのであればぼかしを施し、似ていなければほとんどぼかしは行われぬ。このような関数を用いることで、擬似輪郭の発生を抑制することが出来る。

本稿では、この手続きを 10 回繰り返す。

そして、最後に以下の計算を行う事により  $l$  と  $\phi$  の計算を行い、 $\phi$  を  $\theta$  に変換し直す。

$$l(x, y) = \sqrt{v_x(x, y)^2 + v_y(x, y)^2} \quad (7)$$

$$\phi(x, y) = \arctan \frac{v_y(x, y)}{v_x(x, y)} \quad (8)$$

### 3.5 白黒画生成

3.4 節で求めた線のパラメータを元に、誤差拡散法を利用する事により白黒画を生成する。

通常、誤差拡散法は、白と黒の 2 値情報のみを用いて豊富な濃淡階調を表現する時に用いられるが、本稿

では、画素に対し白か黒のピクセルを割り当てる代わりに画素  $(x, y)$  を中点とする長さ  $l(x, y)$ 、太さ  $w(x, y)$ 、線の角度  $\theta(x, y)$  の線を描き込む事とする。

誤差拡散法として、本稿では Floyd-Steinberg アルゴリズム [8] を適用する。このアルゴリズムは、各画素における誤差拡散を次のような手順で行う。

最初に、描画する画像領域を作り、黒一色に塗り潰す。その後、最大輝度の白の線を描いていく。ここで、ある画素における濃度（ここでは0から1の値をとる。0が黒を表し、1が白を表している。） $g(x, y)$  の値が1以上になった時に線を描画し、1未満だった場合には線を描画しないようにする。

周囲に拡散させる誤差の総和  $E_{x,y}$  の値は式 (7) で決定する。

$$E_{x,y} = g(x, y) - P_{x,y} \quad (9)$$

ここで、 $P_{x,y}$  は、画素  $(x, y)$  において線を描画する時に塗り潰したピクセル数を表す。 $(x, y)$  で線を描画しない場合は、 $P_{x,y}$  の値は0となる。

そして、周囲に誤差を拡散させた後のピクセルの値は以下の様になる。

$$g(x+1, y) := g(x+1, y) + \frac{7E_{x,y}}{16} \quad (10)$$

$$g(x-1, y-1) := g(x-1, y-1) + \frac{3E_{x,y}}{16} \quad (11)$$

$$g(x, y-1) := g(x, y-1) + \frac{5E_{x,y}}{16} \quad (12)$$

$$g(x+1, y-1) := g(x+1, y-1) + \frac{E_{x,y}}{16} \quad (13)$$

以上の処理をラスタ走査しながら行っていく。

## 4 実験結果

本章では、本提案アルゴリズムの実験結果を示す。

まず、図2を元画像とした実験例を示す。

まず、3.1節で説明した色空間領域分割方法を適用する。ここでは、色空間の領域分割数を12に設定している。領域分割された結果の画像を図3に示す。

次に、分割した領域ごとに線の長さ、太さ、角度の指定を行う。本稿では、線の長さの最大値  $l_{max}$ 、最小値  $l_{min}$ 、線の幅の最大値  $w_{max}$ 、最小値  $w_{min}$ 、角度のモードはどのようになっているかについてのみ書く。ここでは  $l_{max} = 6$ 、 $l_{min} = 2$ 、 $w_{max} = 2$ 、 $w_{min} = 1$  と



図 2: 元画像



図 3: クラスタリング画像

し、角度のモードは、色領域によって濃度勾配に従い線の方向を設定するモードと、ユーザが設定した固定値を設定するモードとを使い分けるように指定した。

3.4節に説明した非等方拡散処理の効果を示すため、図4に3.4節の処理なしで結果画像を作成した例を示す。この図の空の部分を見ると、分割領域の境界付近の描画で擬似輪郭が生じている事が分かる。これは、急激な線のパラメータの変化により生じたものである。

非等方性拡散法を適用して生成された白黒画を図5に示す。この画像を見ると、擬似輪郭が発生せず、なおかつ、エッジの情報も保持されている事が分かる。

他のパラメータを与えた結果を図6、7に示す。図6では、線のパラメータは、 $l_{max} = 12$ 、 $l_{min} = 4$ 、 $w_{max} = 2$ 、 $w_{min} = 1$  とし、図6では、線のパラメータは、 $l_{max} =$



図 4: そのまま白黒画を生成した例



図 6:  $l_{max} = 12$ ,  $l_{min} = 4$ ,  $w_{max} = 2$ ,  $w_{min} = 1$



図 5: 生成された白黒画

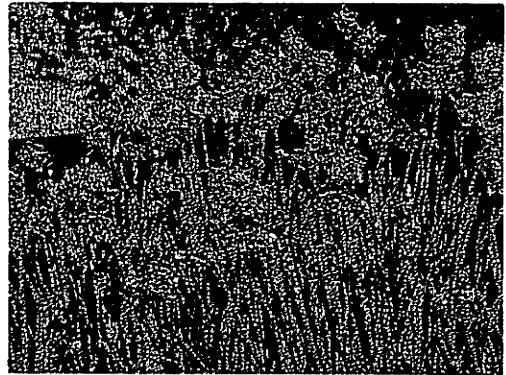


図 7:  $l_{max} = 12$ ,  $l_{min} = 2$ ,  $w_{max} = 3$ ,  $w_{min} = 1$

12,  $l_{min} = 2$ ,  $w_{max} = 3$ ,  $w_{min} = 1$ とした。角度のモードは、両方とも色領域によって濃度勾配に従い線の設定するモードと、ユーザが設定した固定値を設定するモードとを使い分けるように設定した。これらの図から分かるように、ユーザにより設定されるパラメータを変えることによってさまざまな風合いの結果画像を作り出すことができる。

他の画像から生成した例については、図 9 に示す。線のパラメータは、 $l_{max} = 6$ ,  $l_{min} = 2$ ,  $w_{max} = 2$ ,  $w_{min} = 1$ とし、角度のモードは、色領域によって濃度勾配に従い線の設定するモードと、ユーザが設定した固定値を設定するモードとを使い分けるようにした。この例でも、擬似輪郭が生じることなく、自然な

結果を生成することができていることが分かる。

## 5 結論

本稿では、誤差拡散法を応用した非写実的な白黒画生成手法を示した。本手法では、ユーザが自由に描画パラメータを調整することによって多種多様な描画法を生み出すことができた。

今後の課題として、誤差拡散の時に敷き詰めるストロークの形状の多様化、誤差拡散のさせ方によるテクスチャの変化の分析などが考えられる。また、インタラクティブな白黒画生成システムの導入や、アニメーションへの応用なども今後の課題である。



図 8: 元画像



図 9: 生成された白黒画

## 参考文献

- [1] Georges Winkenbach, David H. Salesin : "Computer-Generated Pen-and-Ink Illustration" , *Proceedings of SIGGRAPH 94*, pp.91-100, 1994
- [2] George Winkenbach, David H. Salesin : "Rendering Parametric Surfaces in Pen and Ink" , *Proceedings of SIGGRAPH 96*, pp.469-476, 1996
- [3] Michael P. Salisbury, Sean E. Anderson, Ronen Barzel, and David H. Salesin : "Interactive Pen-and-Ink Illustration" , *Proceedings of ACM SIGGRAPH 94*, pp.101-108, 1994
- [4] Michael P. Salisbury, Michael T. Wong, John F. Hughes, and David H. Salesin : "Orientable Textures for Image-Based Pen-and-Ink Illustration" , *Proceedings of SIGGRAPH 97*, pp.401-406, 1997
- [5] Salisbury M, Anderson C, Lischinski D, and Salesin, D.H. : "Scale-Dependent Reproduction of Pen-and-Ink Illustrations" , *Proceedings of SIGGRAPH 96* , pp.461-468, 1996
- [6] "Real-Time Hatching" E. Praun, H. Hoppe, M. Webb, A. Finkelstein. *Proceedings of SIGGRAPH 2001*, pp.581-586, 2001
- [7] Shigefumi Yamamoto, Xiaoyang Mao, Kenji Tanii, Atsumi Imamiya : "Enhanced LIC Pencil Filter" , *Proceedings of International Conference on Computer Graphics, Imaging and Visualization*, pp.251-256, 2004
- [8] Thomas Strothotte, Stefan Schlechtweg : "Non-Photorealistic Computer Graphics" , *Morgan Kaufmann Publishers*, 2002
- [9] P. Perona, J. Malik : "Scale-space and edge detection using anisotropic diffusion" , *PAMI 12(7)*, pp.629-639, 1990