

## アルゴリズムによる疑似水彩画作成の一手法

5 D-8

深村 有史† 山本 強† 青木 由直†

†北海道大学工学部

‡北海道大学大型計算機センター

## 1. はじめに

写真、あるいはレンダリングの結果の画像を用いて疑似絵画を作成する方法はいくつかの例が発表されているが、現在の所、作成の過程で何らかの人間の判断が介入する場合が多い。絵画において、一様なタッチで全体を表現する場合、対象となる画像によって最も細かく表現すべき箇所の出来具合によって、絵に対する評価が決ってしまい、また、自動的に適切なタッチの細かさを選択することが非常に困難だからである。

またそうして得られた画像のほとんどが、油彩やペン画のように滲みや混色などを考慮する必要のない画材を題材にしており、特に水彩表現の自動生成は実現されていなかった。

本研究では、油彩表現、及び水墨画の様な単色水彩表現に加え、透明水彩のような複数の色彩を持つ水彩表現の自動生成を試みた。

## 2. 画像内でのタッチの方向の決定

写真画像から疑似絵画像を作成するためには、筆のタッチでその画像を再現するためにどのような方向で絵筆を動かせばよいかを決定しなくてはならない。そのために、写真画像内での色分布の方向性を検出し、それを写真の持つ情報ファイルとする。

まず写真画像をスキャナを使って取り込み、画像全体を二次元配列に対応させ、ピクセルごとにRGB値のデータとしての色の分布状態をチェックする。各ピクセルからサーチを放射状に行い、色変化の最も少ない方向を直線的に選り出す。

これを繰り返して得られた、サーチの方向、大きさ、そのピクセルでの色の強度を、筆の運びの方向、タッチの大きさ、絵の具の色として決定し、写真画像のもつ情報としてファイルに蓄える。

このデータに従って筆のタッチで色情報を画素配列に並べていくことにより、疑似絵画像が生成される。筆のパターンはあらかじめマスクとして準備され、タッチの大きさに比例して拡大・縮小、タッチの方向にあわせた回転などの変換がおこなわれて、絵の具の色でRGB強度が与えられ、再び二次元配列に格納される。

これが基本的な疑似絵画の生成法であるが、今回はタッ

チの数を疎にして得られたおおまかな画像と、もとの写真画像を色ごとにブロック分けした画像の2つを使い、水彩独特の色の滲みを表現した。

## 3. 彩色の部分決定

水彩画は油彩などと異なり、上からの塗りつぶしが効かない。また、紙に滲む、塗りむらが生じる、乾燥の早さによって、部分的に色が異なる等、物理的再現が困難である。

これらを解決するために、画像全体を色ごとにブロック分けし、その中で粗いラインをスムージングする事により、塗りむらを再現した。スムージングは、数ピクセルごとに色強度の平均値を求めて、それを新しいピクセルの強度とした。

実際の写真画像内ではフルカラーで色が使われているわけだが、使用する色を限定し色階調を落として、各色ごとに使用されている領域を抜き出し、それをマスクとして、前もって作った少ないタッチで描かれた粗い画像に軽いスムージングをかけたものからマスク部分を切り出す事によって、色の存在範囲を決定した。

そうして得られた色のブロックそれぞれについて、滲みの情報を付加し、最終的にそれらを合成して一枚の画像を得た。

## 4. 表面のにじみによる水彩表現

画像に対する滲み情報の付加は、あらかじめ作った墨の紙への滲みをスキャナで取り込み、その濃淡の強度を二次元配列に格納したものを利用する。

疑似絵画像の色強度の二次元配列と墨の濃淡の強度の二次元配列を対応させ、紙に墨が滲ん部分は、その滲みの濃さに比例して疑似絵画像の画素強度を増してやることにより、あたかも水彩色が紙ににじんだかの様な表現を行うことができる。

この様なにじみの領域を決定した後に、水分の乾燥に伴う色の濃淡を出すために、にじみ領域の外縁部分の色強度を少し強めてやる。これは、水彩画において、筆のストロークによって出来た色の塗られた部分は、縁の部分から乾燥を始めるため、その部分の色が多少他の部分より濃くなることを再現するためである。

これらの処理を行って色のにじみを再現するが、このままでは全体的に散漫な印象を与えるため、絵として認識しやすくするために写真画像から細部を抽出した画像を用いて細部の強調を行う。

写真画像の中で特に色強度の強いピクセルを始点としてペン画状の筆のストロークを行うと、ペンによるデッサン調の画像が生成される。この画像をスムージングして水彩画像に組み込んでやることにより、細部の強調が自然な形でできる。

## 5. アルゴリズム

全体の手順としては、以下の通りとなる。

1. 写真をスキャナで取り込み、二次元配列に対応させる。
2. ランダムに選んだピクセルから放射状にサーチを行い、RGB強度の変化の少ない方向にサーチを進める。サーチは、一定の長さ、一定のRGB強度変化を越えたと中止する。
3. 得られたデータを、一旦ファイルにする。
4. ファイルに蓄えられたデータに従い、定められたペンタッチで二次元配列上にRGB強度を与えていく。ペンタッチはファイルとして用意されており、任意の形状をあらかじめ選択できる。
5. 少ない数(5000程度)のタッチのストロークで、ラフデッサン程度の粗い画像を作る。
6. 写真画像のデータにスムージングをかけ、色調を粗くする。
7. 絵の具として使用する色ごとに画像内で使用されている領域を抜き出し、フィルターとする。
8. デッサン画像にスムージングをかける。
9. 紙への水の滲みをスキャナでとった画像を利用し、水の滲んだ部分にあたるピクセルを、デッサン画像に於て強度を強める。
10. 先ほどのフィルターを用いて、絵の具の色ごとにデッサン画像から領域を切り出していく。その際、領域の境界にあたる部分は色の強度を少し強める。
11. フィルターごとに切り出した領域を、一枚の画像に合成する。
12. 写真画像から特に色の濃い部分を切り出し、その画像をペンタッチのストロークの始点とした細部の強調画像を生成する。
13. 細部の強調画像にスムージングをかける。

## 14. 細部の強調画像を、11の画像に合成する。

こうして完成したデータを出力し、最終的な絵画像をえる。

以上のような方法の利点として、全体がバッチ処理の連続によるものであることが挙げられる。

個々のルーチンはフィルターとしてあつかわれ、写真画像を定められた順にフィルターを通していくことにより最終的な画像が生成される。

これは、ひとたび作られた画像がファイルとして扱われ、必要に応じて何度も利用出来ること、また、画像生成のアルゴリズムはフィルター群の組合せによって実現されるため、様々な試みがフィルター部やその組替えにより、簡単に実現できることを示している。

例えば、油彩表現などの場合に、あまり重要でない部分が必要以上に細かく描き込まれたり、逆に人物の顔を構成する主要な線が大きく歪んでしまったりした場合が考えられるが、処理の順番や利用の仕方を変えて、細かい画像と粗い画像を組み合わせたりすることによって解決することができる。

これらの方法によって、従来困難であった「人間の判断」を介させない疑似絵画生成を可能にした。

## 参考文献

1. Paul Haeberli, "Paint By Numbers: Abstract Image Representations" Computer Graphics, Volume 24, Number 4, August 1990
2. Tsuyoshi Yamamoto, "Algorithmic Painting" ACM SIGGRAPH '91 Technical Slide Set #46, July 1991