

画像システム特論発表用レジュメ

是松 優作, 最上 伸一

1. 概要

本稿では,「4. アプリケーションを作ってみた」に基づき,与えられた画像からその色境界を識別してクラスタリングすることで,塗り絵の問題を自動生成するアルゴリズムを提案する. また,実際に人間などの手によって作成された塗り絵を採点する機能を提案する.

2. 提案手法

塗り絵の問題を自動生成するために,入力画像に2種類の画像処理を施す.

1. 画像をクラスタリングして色の種類を減らし,塗り絵の模範解答の画像を生成する**クラスタリング処理**.
2. 塗り絵の模範解答画像から,塗り絵の問題を作成する**境界識別処理**.

また,得られた塗り絵の問題を人間などが解いた後に,その結果がどの程度元画像と合致しているかどうかを示す尺度を導入する.

2.1 クラスタリング処理

塗り絵を行いやすい画像を得るためには,色の類似度・画像の空間的な距離を同時に考慮に入れつつ,塗り絵に用いる色の種類をある閾値以下に抑えることが望ましい.そこで本稿では,mean shift 法を用いていくらかの画像クラスタリングを行った後の画像に,さらにk-means 法をかける手法を提案する.

まず,与えられた画像に mean shift 法による画像処理を施すことにより,

2.2 境界識別処理

上記手法によって得られた正解画像から塗り絵の問題を自動生成するために,色の一様な部分を白色に,色の境界線を黒色にすることで対象画像を線画化する.

2.3 評価関数

評価関数は,規格化のために以下のものを導入する.

$$100 - 100 \cdot \frac{D - D_{\min}}{D_{\max} - D_{\min}} \quad (1)$$

2.4 ソースコード

本手法のソースコードは,

<https://github.com/diesekiefer/ImagingSystem>より取得可能である.

3. 実験

代表的な画像について,実験には,画素数(いくつ?)のパブリックドメインの画像を使用した.計算機のスペックは以下である.



図1 元画像.



図2 クラスタリング処理を行った後.

MacBook (Retina, 12-inch, Early 2016)

1.1 GHz Intel Core m3

8 GB 1867 MHz LPDDR3

実験にかかった時間は色のクラスタリング処理に19.39[sec] 塗り絵画像生成に7.014[sec]であった.

4. 結論

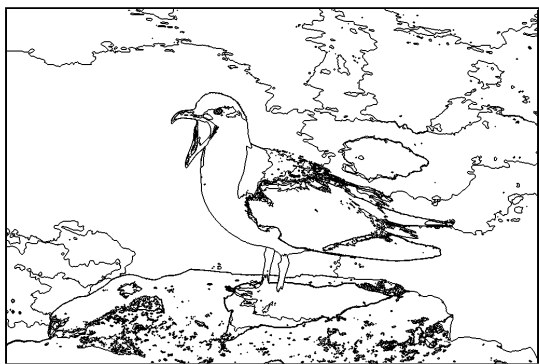


図3 得られた塗り絵の問題（外側の黒枠線は境界を明確にするために追加したものであり，元画像データにはない）.