

デジタルメディア処理2

担当: 井尻 敬

○ 講義の概要:

画像処理・画像認識は、産業・自然科学・エンタテインメントなど、多種多様な分野の発展に関わる非常に重要な技術です。デジタルメディア処理では、画像データ構造・画像撮影方法・線形フィルタ・非線形フィルタ・フーリエ変換・拡大縮小・補間など、主に**画像を編集する方法**を置き紹介しました。本科目では、計算機を利用して**画像を認識する手法**について紹介します。具体的には、画像から目的部分を切り抜く領域分割、画像の局所領域の特徴を計算機が理解できる形で記述する特徴抽出、および、抽出した特徴を用いて画像を識別するパターン認識を紹介します。また、講義の後半では、深層学習を用いた画像認識についても取り扱います。

それぞれの技術に関して、コーディング可能な深さで理解できるよう、ソースコードを交えながら詳細な技術解説を行ないます。また、Pythonを用いたプログラミング演習を通して講義内で取り扱う手法のより深い理解を目指します。講義資料は井尻のweb pageに事前にアップロードします。また、講義動画は、受講者に限りscombより閲覧可能です。前期のデジタルメディア処理に比べ、比較的高度な内容を多く紹介するので、講義資料を予習に、講義動画を復習に活用してください。

○ 達成目標:

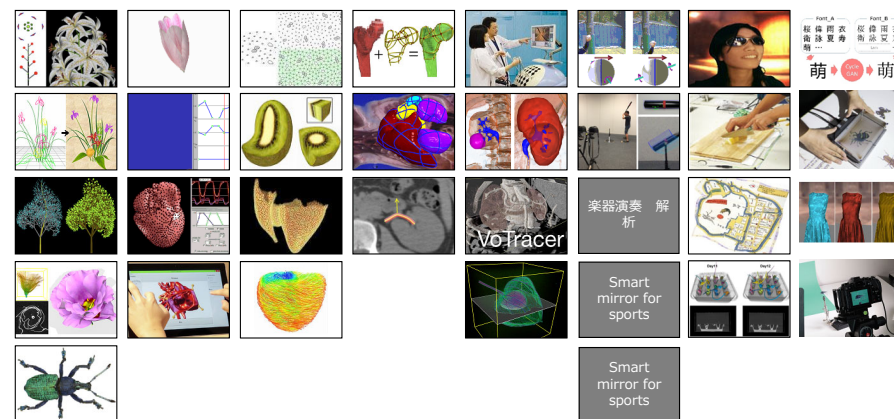
1. 領域分割 - 画像の領域分割法について主要なアルゴリズムを説明・実装できる
2. 特徴抽出 - 画像認識に必要な特徴抽出の基礎を説明・実装できる
3. パタレコ - 画像に対するパターン認識（顔認識など）の基礎とアルゴリズムを説明・実装できる

○ 成績評価:

筆記試験（45%）、プログラミング課題（45%）、ミニッツペーパー(10%)に基づき評価します。

井尻敬 - takashijiri.com

専門: Computer Graphics / 画像処理 / ユーザインタフェース



○ 講義資料:

講義で紹介した資料・ソースコードは可能な限りWeb上に公開します。以下のURLを参考にしてください。

URL: takashijiri.com/classes

○ 質問など:

講義に関する質問があれば、講義後またはメールにてご連絡ください。

オフィスアワーは金曜日2限。

takashi.ijiri80 AtMark gmail.ac.jp

○ その他:

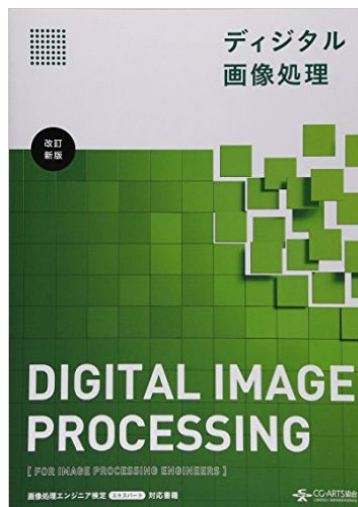
講義中・講義後の質問は歓迎します。（すぐに答えられない場合は私の宿題になります。）

講義中の飲食は、飲み物とあめガム程度まで許可します。

講義資料閲覧やLSM利用のためのラップトップ・スマホの利用を許可します。

講義中の私語は周囲の迷惑になるため控えてください。必要な場合はオンラインで行なってください。（本学講義において私語で困ったことはないのですが、一応）

この文章を書いている3月現在、COVID-19の収束が予測できない状況です。基本的に自宅で講義を受けられるよう、動画配信やscombによるミニッツペーパー・筆記試験を用意しています。自宅にpythonが動くPCを持っている方は、原則自宅にて受講してください。



教科書

- CG-Arts協会（画像情報教育進行委員会）
- デジタル画像処理[改訂新版] 大型本
- 日本語で読める画像処理の教科書です
- 画像や例が多く入門者には最適だと思います
- 網羅性が非常に高い反面，初学者にとっては説明が少ない部分もある
→ 講義中に丁寧に解説します

ある手法を『理解する』とは？

- 教科書をおぼえた：×
- 人にその手法を説明できる：△
- 例を挙げて人に説明できる：○
- プログラムとして記述できる：◎

→ コードを書こう！

※井尻の偏見に基づきます。異論は認めます。
※きちんと理解できていない手法も離散化された数式さえあれば
コードに落とせることも結構あります。。

ソースコードについて

- 本講義紹介する手法はなるべくソースコードも合わせて提供します
 - takashijiri.com/classes
 - github.com/TakashiIjiri/PythonOpenCVPractice
- Python & OpenCV または MFC & C++ 環境で書いてあります
- インストール方法・コーディングの基本に関する資料も用意します
 - ただし詳細は講義中には触れません
 - 興味のある人だけ自由に勉強を進めてください
 - Pythonについては学内環境ではインストールの必要がありません（学情の方のおかげです）

Contents（旧）

1. 序論 : インTRODakション, テクスチャ合成
2. 特徴検出1 : テンプレートマッチング, コーナー・エッジ検出
3. 特徴検出2 : DoG特徴量, SIFT特徴量, ハフ変換
4. 領域分割 : 領域分割とは, 閾値法, 領域拡張法, 動的輪郭モデル
5. 領域分割 : グラフカット, モーフォロジー処理, Marching cubes ※変更可能性あり
6. パターン認識基礎1 : パターン認識概論, サポートベクタマシン
7. パターン認識基礎2 : ニューラルネットワーク, 深層学習
8. パターン認識基礎3: 主成分分析, オートエンコーダ
9. 筆記試験 (50点満点)
10. プログラミング演習 1
11. プログラミング演習 2
12. プログラミング演習 3
13. プログラミング演習 4
14. プログラミング演習 5

Contents (新：学年暦変更後)

- 00. 序論 : インTRODakション
- 01. 特徴検出1 : テンプレートマッチング、コーナー・エッジ検出
- 02. 特徴検出2 : DoG特徴量, SIFT特徴量, ハフ変換
- 03. 領域分割 : 領域分割とは, 閾値法, 領域拡張法, グラフカット法
- 04. パターン認識基礎1 : パターン認識概論, サポートベクタマシン
- 05. パターン認識基礎2 : ニューラルネットワーク、深層学習
- 06. パターン認識基礎3 : 主成分分析, オートエンコーダ
- 07. プログラミング演習 1 : zoom実施 ※ 講義時間中zoomを開設, TAに自由に質問可
- 08. プログラミング演習 2 : zoom実施 ※ 提出済み課題について, 井尻に説明する時間を設ける
- 09. プログラミング演習 3 : zoom実施
- 10. プログラミング演習 4 : zoom実施
- 11. プログラミング演習 5 : zoom実施
- 12. 筆記試験 (Scomb実施)

復習：デジタルメディア処理

本講義はディジタルメディア処理(前年後期)の知識を前提とする

- 画像とは
- 画像の変形とアファイン変換
- フーリエ変換 (FFT)
- フィルタ処理
- 畳み込み
- 逆畳み込み

過去資料は, <http://takashiijiri.com/classes/index.html>

復習：講義を理解するために前提とする数学知識

- 高校までの数学
- 線形代数全般：ベクトル, ベクトルの内積, 行列, 行列の積, 固有値固有ベクトル
- 最適化数学（主に最急降下法を利用）
- 畳み込みとフーリエ変換

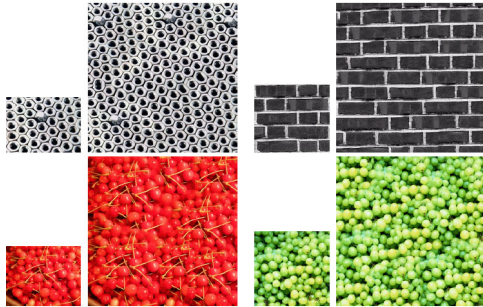
テキストチャ合成 と Seam curving

研究室配属後の論文輪講のイメージと発展課題紹介

• **テクスチャとは**, ここでは『物体表面に現れる模様』のことを指す
※分野によっては, 触感・歯ざわりなどもテクスチャと呼ばれる

• **テクスチャ合成とは**, 例となるテクスチャから新たなテクスチャを生成する技術のこと.

図は[Kwatra et al SIGGRAPH 2005]より



小さなテクスチャから大きなテクスチャを生成

画像は[Simakov et al. CVPR 2008]より



画像リサイズ

テクスチャ合成法

- 似た局所領域を検索
 - 画素毎にコピーする (←今回はここだけ紹介)
 - Alexei A. Efros and Thomas K. Leung, Texture Synthesis by Non-parametric Sampling, ICCV 1999
- 画像最適化(片方向類似度)
 - Vivek Kwatra et al. Texture Optimization for Example-based Synthesis, SIGGRAPH 2005
- 画像最適化(双方向類似度)
 - Simakov et al. Summarizing Visual Data Using Bidirectional Similarity, CVPR 08.
 - Wei et al. Inverse texture synthesis, SIGGRAPH 08.
- 高速近傍計算
- 目立たないシームを計算

Image quilting

Graph Cut textures

IEEE International Conference on Computer Vision, Corfu, Greece, September 1999

Texture Synthesis by Non-parametric Sampling

Alexei A. Efros and Thomas K. Leung
Computer Science Division
University of California, Berkeley
Berkeley, CA 94720-1776, U.S.A.
{efros,leung}@cs.berkeley.edu

• ICCV 1999

背景: 幅広い応用があるため『テクスチャ合成』は需要の高い技術.

課題: 当時既存のテクスチャ合成技術は, Markov random fieldや周波数解析に基づくものだったが, 多様なテクスチャに対してよい合成は難しかった.

提案: 中心から徐々に合成を進めるアルゴリズムを提案. 新しい画素値を埋めるときに入力画像から似た近傍領域を持つ画素をサンプリングする

アルゴリズム (簡易版)

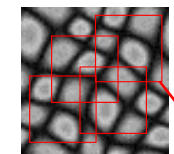
- 中央からテクスチャを“grow”させる

入力: サンプル画像 I_{smp} , 近傍サイズ k

出力: 合成画像 I

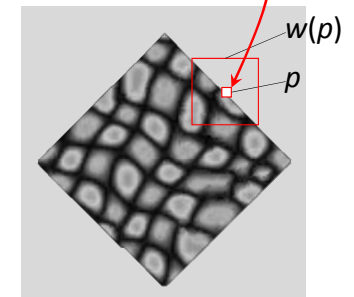
1. 画像 I の中心 3×3 画素をランダム初期化
2. 以下を繰り返す
 - 2.1 既合成部分の隣接画素 p を選択
 - 2.2 p の近傍 $w(p)$ と最も似た領域 w_{best} を I_{smp} より検索
 - 2.3 w_{best} の中央画素値を p に代入
 - 2.4 全画素の合成がなされたら終了

本当はもう少し複雑 (次頁へ)



サンプル画像

近傍が似た画素を検索



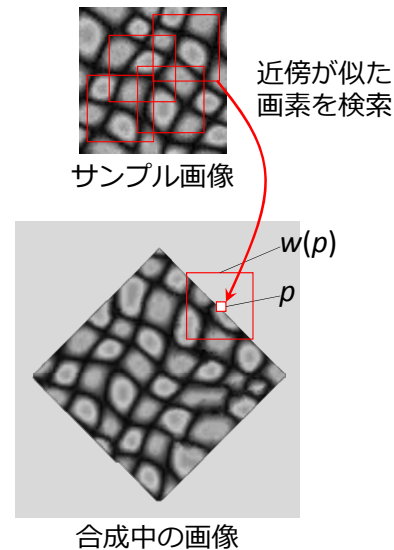
合成中の画像

アルゴリズム

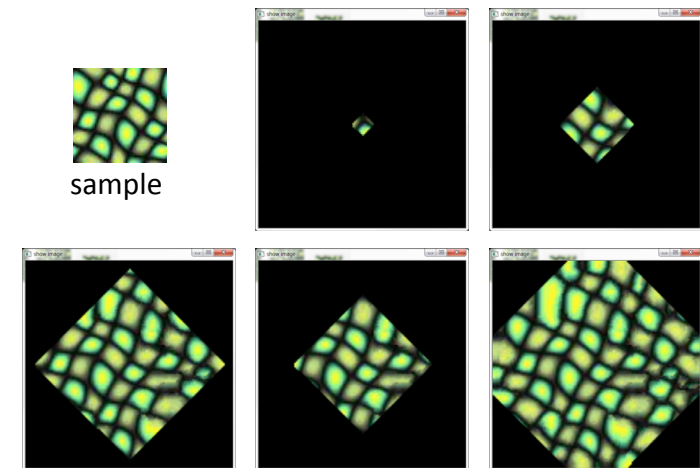
入力：サンプル画像 I_{smp} , 近傍サイズ k

出力：合成画像 I

1. 画像 I の中心3x3画素をランダム初期化
2. 以下を繰り返す
 - 2.1 既合成部分の隣接画素 p を選択
 - 2.2 p の近傍 $w(p)$ と最も似た領域 w_{best} を I_{smp} より検索
 - 2.3 $d(w(p), w') \leq 1.1 * d(w(p), w_{best})$ を満たすすべての w' を I_{smp} より検索
 - 2.4 発見した複数の w' の中央画素値からヒストグラムを作成し, 最も頻度が高いものを p に代入
 - 2.4 全画素の合成がなされたら終了



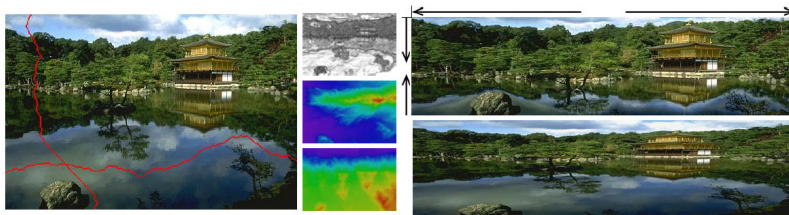
実装例 - プログラミング課題として出題する予定



Seam Carving for Content-Aware Image Resizing

Shai Avidan
Mitsubishi Electric Research Labs

Ariel Shamir
The Interdisciplinary Center & MERL



背景：コンテンツの縦横比を変化させたいことが多くある (image retargeting と呼ばれる)

問題：画像のretargetingにおいて単純に横方向に伸縮させると, 撮影されたものの縦横比も変化してしまう

提案：画像全体の縦横比を変化させながら映ったものの縦横比を変化させないretargeting手法

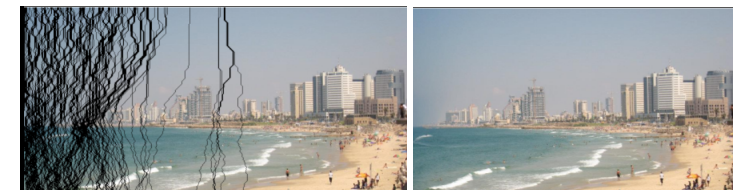
アイデア

- 各画素に重要度を定義 (例 : $I_x^2 + I_y^2$ or $|I_x| + |I_y|$)
- 横方向に縮める場合は, 以下の条件を満たすseamを計算しそれを間引く
 - 画像の上辺から下辺へ画素をひとつずつつないだもの
 - ひとつ下に行く際, 一画素分横方向に移動できる
 - 重要度の総和が最小となる

重要度マップ



[図は論文より]



実装例 - プログラミング課題として出題します

