

情報デザイン専攻

画像情報処理論及び演習I

-画像合成・類推- Image Analogy

第9回講義
水曜日1限
教室6218情報処理実習室

吉澤 信
shin@riken.jp, 非常勤講師
大妻女子大学 社会情報学部

独立行政法人
理化学研究所

Shin Yoshizawa: shin@riken.jp

今日の授業内容

www.riken.jp/briect/Yoshizawa/Lectures/index.html
www.riken.jp/briect/Yoshizawa/Lectures/Lec09.pdf

- ① 画像類推のアルゴリズム・演習.
- ② NumberEditorの演習.
- ③ レポート3.

Shin Yoshizawa: shin@riken.jp

復習: Pixel TransferによるInpainting

✓ 画像から似ている画素・Textureを持つてくる.

- 局所Windowで類似パターンを検索: Windowサイズに依存.
- 低周波画像は補間で生成しておくことと影等の効果を反映出来る.
- 穴(マスク)を埋める順番が重要!

類似検索

©D. Hahn, Univ. Illinois.
©H. Yamuchi et al., CGJ 2005.

Shin Yoshizawa: shin@riken.jp

復習: 画像類推: Image Analogy

✓ AがA'になるならば、BはB'になる → B'の画素値(色、明るさ)を類似検索で計算: 局所テクスチャーを特徴ベクトルとし類似検索する事で、類推したピクセルを転写.

入力

出力

©A. Hartmann et al., SIGGRAPH 2001.

Shin Yoshizawa: shin@riken.jp

Image Analogyアルゴリズム

パラメータ(Windowサイズ): $r \geq 2$.

CREATEIMAGEANALOGY(A, A', B)

- 1 Compute Gaussian pyramids for (A, A', B)
- 2 Compute features for (A, A', B)
- 3 Initialize search structures
- 4 for $\ell = 0$ to L
- 5 for each pixel $q \in B'_\ell$, in scan-line order
- 6 $p \leftarrow \text{BESTMATCH}(A, A', B, B', s, \ell, q)$
- 7 $B'_\ell(q) \leftarrow A'_\ell(p)$
- 8 $s_\ell(q) \leftarrow p$
- 9 return B'_ℓ

©A. Hartmann et al., SIGGRAPH 2001.

Shin Yoshizawa: shin@riken.jp

Image Analogyアルゴリズム2

✓ 検索はANN (Approximate Nearest Neighbor)ライブラリを使う.

✓ ANNはエラー(誤差)を許して高速にn次元空間の近傍をサーチ.

パラメータ(ANNEror): $E \geq 1.0$.

CREATEIMAGEANALOGY(A, A', B)

- 1 Compute Gaussian pyramids for (A, A', B)
- 2 Compute features for (A, A', B)
- 3 Initialize search structures
- 4 for $\ell = 0$ to L
- 5 for each pixel $q \in B'_\ell$, in scan-line order
- 6 $p \leftarrow \text{BESTMATCH}(A, A', B, B', s, \ell, q)$
- 7 $B'_\ell(q) \leftarrow A'_\ell(p)$
- 8 $s_\ell(q) \leftarrow p$
- 9 return B'_ℓ

©A. Hartmann et al., SIGGRAPH 2001.

Shin Yoshizawa: shin@riken.jp

Image Analogyアルゴリズム3

✓ Best Approximate MatchはWindowの半径2のとき5次元ベクトルのガウス相関。
 - ガウス相関: 中心の画素からガウス関数で重みを付けて対応する画素を要素とするベクトルの距離。

✓ Best Coherence MatchはTextureの整合性を加味して既に合成された画素の対応する画素でサーチ。

BestMATCH(A, A', B, B', s, t, q)

```

1 papp ← BESTAPPROXMATCH(A, A', B, B', s, t, q)
2 pcoh ← BESTCOHERENCEMATCH(A, A', B, B', s, t, q)
3 dapp ← ||Fi(papp) - Fi(q)||2
4 dcoh ← ||Fi(pcoh) - Fi(q)||2
5 if dcoh ≤ dapp(1 + 2s-t-k)
6   then return pcoh
7   else return papp
    
```

パラメータ(Texture度): k>=0.

CA: Hartmann et al., SIGGRAPH 2001.

Shin Yoshizawa: shin@riken.jp

復習:演習:Texture Transfer

✓ Image Analogyを用いてTexture Transferを実行するプログラム:
引数 8

- TextureTransfer 入力画像A.ppm 入力画像A'.ppm 入力画像B.ppm 出力画像B'.ppm Texture度k(double>=0.0) ANN誤差(double>=1.0) Window半径(int>=2) Blending(1.0>=double>=0.0, 小→元画像強め)
- sh Run_TextureTransfer.shでも実行可能。

Shin Yoshizawa: shin@riken.jp

復習:演習:Texture by Numbers

✓ Image Analogyを用いてTexture by Numbersを実行するプログラム:
引数 7

- TextureByNumbers 入力画像A.ppm 入力画像A'.ppm 入力画像B.ppm 出力画像B'.ppm Texture度k(double>=0.0) ANN誤差(double>=1.0) Window半径(int>=2)
- sh Run_TextureByNumbers.shでも実行可能!実行結果。

Shin Yoshizawa: shin@riken.jp

復習:演習:Artistic Filters

✓ Image Analogyを用いて様々なArtisticフィルタを実行: **引数 7**

- ArtisticFilter 入力画像A.ppm 入力画像A'.ppm 入力画像B.ppm 出力画像B'.ppm Texture度k(double>=0.0) ANN誤差(double>=1.0) Window半径(int>=2)
- sh Run_ArtisticFilter.sh, sh Run_Etc1~3.sh, sh Run_Smoothing_Sharpning.shでも実行可能。

Shin Yoshizawa: shin@riken.jp

演習:Image Analogyを使ってみよう!

www.riken.jp/brict/Yoshizawa/Lectures/Ex05.zip
www.riken.jp/brict/Yoshizawa/Lectures/Lec09.pdf

Image Analogyでフィルタリング:

1. Ex05内に用意されたプログラム群を動かしてみる。
2. Ex05内の画像を用いてImage Analogyによる色々なフィルタリング処理を試してみる。
3. 新しいフィルタリングを考えてみよう!

この演習は第3回レポートの内容なので頑張ってくださいねーp(^)q

Shin Yoshizawa: shin@riken.jp

演習11-5: NumberEditor

Shin Yoshizawa: shin@riken.jp

重要: 演習: NumberEditor

www.riken.jp/brict/Yoshizawa/Lectures/Ex06.zip

- 展開後に端末でEx06/NumberEditorへ移動.
- 端末にて「tcsh」と打ち込んでエンターキー.
- 端末にて「setenv LANG C」と打ち込んでエンターキー.
- 「sh Run_NumberEditor.sh」で実行.

Shin Yoshizawa: shin@riken.jp

演習: NumberEditor

- Image Analogy用TextureByNumbersのお絵かきGUI (Java).
- Ex06/NumberEditor/
- コンパイル: 端末で「javac NumberEditor.java」
- 実行: 端末で「java NumberEditor」
- sh Run_NumberEditor.shでもOK!

- 画像を読み込む: File->Load ppm Image. ppm画像
- お絵かき: 左クリック(ドラッグ)
 - 色を変える: 右下のSelectボタン.
 - ブラシのサイズを変える: 右のスクロールバー or マウスホイール.
 - 表示の透明度を変える: 下のスクロールバー.
- マスク画像(ppm)をセーブ:File->Save Number Image. 拡張子.ppm要る画像ファイル名.
- Ex06/TextureByNumbersの第2引数及び第3引数(もちろん違うお絵かきの結果)へ.

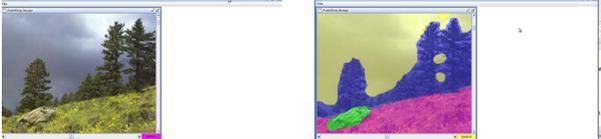


Shin Yoshizawa: shin@riken.jp

演習: NumberEditor & TextureByNumbers

- Image Analogy用TextureByNumbersのお絵かきGUI (Java).
- Ex06/NumberEditor/

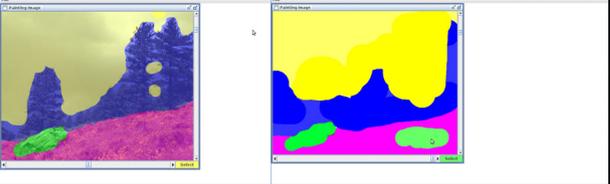
- sh Run_NumberEditor.shで立ち上げてください.
- 画像を読み込む: File->Load ppm Image. Ex05/darkclouds.ppmを開いてみてください.
- お絵かき: 左ドラッグ: 木、岩、草原、空を違う色で塗ってみてください.
 - 色を変える: 右下のSelectボタン.
 - ブラシのサイズを変える: 右のスクロールバー or マウスホイール.
 - 表示の透明度を変える: 下のスクロールバー.
- セーブ:File->Save Number Image. A.ppmという名前で保存してください.



Shin Yoshizawa: shin@riken.jp

演習: NumberEditor & TextureByNumbers

- 追加のお絵かき: 木、岩、草原、空で使った色とほぼ同じ色で書き足してみてください.
- マスク画像(ppm)をセーブ:File->Save Number Image. B.ppmという名前で保存してください.
- A.ppmとB.ppmをEx05の下に移動(コピーでもカット&ペーストでもOK)してください.
- 端末を新たに立ち上げて、Ex05にcdで移動してください. もしもEx05をデスクトップで立ち上げていたら「cd ~/Desktop/Ex05」又はファイルブラウザのパスをコピーして端末に張り付けて「cd パス」でエンターキーを押す.



Shin Yoshizawa: shin@riken.jp

演習: NumberEditor & TextureByNumbers

- emacsでRun_TextureByNumbers.shを立ち上げて、以下の様

- 第一引数oxbow-mask.ppm は A.ppm
- 第二引数oxbow.ppm は darkclouds.ppm
- 第三引数oxbow-newmask.ppm は B.ppm
- その後のppmファイル名も上のルールで変更してください.

- Run_TextureByNumbers.shをセーブ(上書き保存)してください.
- 端末にて「sh Run_TextureByNumbers.sh」で実行してみてください.



こんな感じなのが
出れば正解⇒

Shin Yoshizawa: shin@riken.jp

学生さんの良く出来ている作品



