

情報デザイン専攻

画像情報処理論及び演習II

-動画像処理- Video Stylization

第11回講義
水曜日 1限
教室6218

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/brict/Yoshizawa/Lectures/index.html
www.riken.jp/brict/Yoshizawa/Lectures/Lec22.pdf

- Artistic Stylization ⇒ Video Stylization
- 演習:
 - DoG画像
 - DoGビデオ
 - Artistic Stylization

今日もプログラミングの話メイン。

Shin Yoshizawa: shin@riken.jp

復習: 動画像の配列表現

```

int I[st][sy][sx];
double I[st][sy][sx];
for(i = 0; i < st; i++) {
  for(j = 0; j < sy; j++) {
    for(k = 0; k < sx; k++) {
      I[i][j][k] = ...
    }
  }
}

```

3D画像の配列表現

© New Line Productions, Inc.

Shin Yoshizawa: shin@riken.jp

復習: Artistic Stylization

✓アーティストの様式を疑似的に再現して実画像を生成・編集する事: NPR/計算Photographyの分野。

© Collobert and J. Kyriazidis, ECCV'11

Shin Yoshizawa: shin@riken.jp

Artistic Video Stylization

✓2Dの基本フレームワークを3D化してみよう!
エッジ保存平滑化→エッジ抽出→ポスター化(多値化、量子化)→合成。

スタイル化

動画像へ拡張

© New Line Productions, Inc.

Shin Yoshizawa: shin@riken.jp

2Dの基本フレームワーク

入力

Bilateralフィルタの繰り返し

平滑化画像

色相Hの多値化& 明度Vの強調

ポスター化

RGBの多値化

HSV量子化画像

RGB量子化画像

エッジ抽出

DoG

エッジ画像

出力Stylized画像

Shin Yoshizawa: shin@riken.jp

復習: DoG

✓ DoG: Difference of Gaussian.

$$g_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)$$

$$DoG_{\sigma, K}(x, y) = g_{\sigma}(x, y) - g_{K\sigma}(x, y)$$

$\sigma = 0.5, K = 2, K = 3, K = 4, K = 5$

$\sigma = 1.0, K = 2, K = 3, K = 4, K = 5$

Shin Yoshizawa: shin@riken.jp

DoG2

✓ DoGと入力画像の畳み込みが負の領域=エッジ:

$\sigma = 0.5, K = 2, K = 3, K = 4, K = 5$

$\sigma = 1.0, K = 2, K = 3, K = 4, K = 5$

Shin Yoshizawa: shin@riken.jp

復習: Bilateralフィルタとは?

Gaussian Filter ← Input → **Bilateral Filter**

$$Z(x, y) = g_{\sigma}(|x - y|)$$

$$Z(x, y) = g_s(|I(x) - I(y)|)g_{\sigma}(|x - y|)$$

$$g_{\sigma}(r) = e^{-\frac{r^2}{\sigma^2}}$$

Intensity (Tonal) Kernel **Spatial Kernel**

Spatial-Tonal Normalized Convolution:

$$I^{new}(x) = \int Z(x, y)I(y)dy / \int Z(x, y)dy,$$

エッジ特徴を保存する!

Shin Yoshizawa: shin@riken.jp

Bilateralフィルタの繰り返し適用

✓ エッジ保存平滑フィルタを繰り返し適用するとエッジに沿った領域が断片化される(領域抽出効果):

```

I^{n+1} = Filtering(I^n)
for(iteration){
  tmp = Filtering(I);
  I = tmp;
}

```

$$I^{new}(x) = \int Z(x, y)I(y)dy / \int Z(x, y)dy,$$

$$Z(x, y) = g_s(|I(x) - I(y)|)g_{\sigma}(|x - y|) \quad g_{\sigma}(r) = e^{-\frac{r^2}{\sigma^2}}$$

入力 1回 2回 3回

$\sigma = 25.0, h = 0.1 \times \text{輝度値の標準偏差}$

Shin Yoshizawa: shin@riken.jp

DoG+Bilateralフィルタ

✓ Bilateralフィルタを繰り返し適用後にDoGを適用:

上: 入力画像にDoG:

下: Bilateralフィルタ3回適用後にDoG:

$\sigma = 0.5, K = 2, K = 3, K = 4, K = 5$

$\sigma = 1.0, K = 2, K = 3, K = 4, K = 5$

Shin Yoshizawa: shin@riken.jp

DoG+Bilateralフィルタ

✓ Bilateralフィルタ後の画像と合成すると...

上: 入力画像にDoG:

下: Bilateralフィルタ3回適用後にDoG:

$\sigma = 0.5, K = 2, K = 3, K = 4, K = 5$

$\sigma = 1.0, K = 2, K = 3, K = 4, K = 5$

Shin Yoshizawa: shin@riken.jp

ポスター化

- ✓ 多値化で量子化する事でポスター化:
 - RGB毎に多値化すると色が混ざる.
 - HSV空間の色相(H)で多値化し明度(V)を強調.

Bilateralフィルタ3回適用後 BGR毎に4段階の値へ量子化

Shin Yoshizawa: shin@riken.jp

ポスター化2

- ✓ HSV空間の色相(H)で多値化し明度(V)を強調.

- ✓ 色相(Hue): 色の様相の相違: 光の波長の様相.
- ✓ 彩度(Saturation/Chroma): 鮮やかさ.
- ✓ 明度(brightness/value/intensity): 明るさ.

色相を16段階の値へ量子化+明度を強調.

Shin Yoshizawa: shin@riken.jp

ポスター化3

- ✓ HSV空間の色相(H)で多値化し明度(V)を強調.

RGB毎の混色で鏡面的効果演出.

色相を16段階の値へ量子化+明度を強調+RGB毎に4段階に多値化.

Shin Yoshizawa: shin@riken.jp

今週はVideoへのDoG拡張

入力 → Bilateralフィルタの繰り返し → 平滑化画像 → 色相Hの多値化&明度Vの強調 → HSV量子化画像 → RGBの多値化 → RGB量子化画像 → 出力Stylized画像

DoG → エッジ抽出

最終的にEx15.zipのStyle.cxxのビデオへの拡張を作成.

Shin Yoshizawa: shin@riken.jp

DoGの3D拡張

- ✓ そのままの拡張は時空間エッジになるので Artistic Stylizationでは工夫が必要:
 - 注意点: 時間方向のパラメータは空間と分けなければダメ、時間方向の畳み込み半径も同様.

$$g_{\sigma,h}(x,y,t) = \frac{1}{2\pi\sigma^2} \frac{1}{\sqrt{2\pi h}} \exp\left(-\frac{x^2+y^2}{2\sigma^2} - \frac{t^2}{2h^2}\right)$$

$$DoG_{\sigma,K,h}(x,y,t) = g_{\sigma,h}(x,y,t) - g_{K\sigma,Kh}(x,y,t)$$

Shin Yoshizawa: shin@riken.jp

DoGの3D拡張2

Shin Yoshizawa: shin@riken.jp

DoGの3D拡張3



ストーリー展開の描写はOKだが単純に重ねるとあまり良くない。



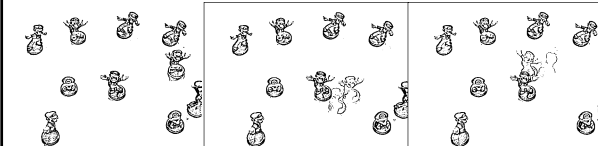
© New Line Productions, Inc.

Shin Yoshizawa: shin@riken.jp

DoGの3D拡張4

✓ 2D空間DoGを時間方向に平滑化し、残像効果:
 - 注意点: レポートでは講義で紹介した時間方向の拡張の仕方以外でもデザインしてOK.

$$g_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad g_h(t) = \frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{t^2}{2h^2}\right)$$

$$DoG_{\sigma, K, h}(x, y, t) = g_h(t)(g_{\sigma}(x, y) - g_{K\sigma}(x, y))$$


Shin Yoshizawa: shin@riken.jp

DoGの3D拡張5




© New Line Productions, Inc.




Shin Yoshizawa: shin@riken.jp

DoGの3D拡張6



© New Line Productions, Inc.



Shin Yoshizawa: shin@riken.jp

DoGの3D拡張7

✓ レポートでは時間方向拡張の仕方をデザインしてOK. ただし狙ったデザインの目的と使った数式を明記する事.
 ✓ パラメータの調節が必要.



© New Line Productions, Inc.

Shin Yoshizawa: shin@riken.jp

演習: DoG画像、DoGビデオ

www.riken.jp/briect/Yoshizawa/Lectures/index.html
www.riken.jp/briect/Yoshizawa/Lectures/Lec22.pdf
www.riken.jp/briect/Yoshizawa/Lectures/Ex15.zip

- Ex15内のプログラムを動かしてみる.
- DoGビデオプログラムの作成.



Shin Yoshizawa: shin@riken.jp

演習:Ex22-1

- ✓ Ex15.zip内でmakeでコンパイルし、testVideoIO.cxx、DoGEdge.cxx、DoGEdge2.cxx、Style.cxxを動かしてみる。
- ✓ 連番画像の入出力: VideoIO.h
void OpenVideo(char *入力フォルダ名, Image3D *R, Image3D *G, Image3D *B, int *sx, int *sy, int *st);
void SaveVideo(char *出力フォルダ名, char *出力ファイル名, Image3D *R, Image3D *G, Image3D *B);
- ✓ DoGEdge.cxx: DoGによるエッジ画像の作成: 引数3.
./DoGEdge 畳み込み半径(int) DoG標準偏差(double) DoGバンド幅(double)
[./DoGEdge lena.bmp ex22_1_1.bmp 10 0.5 2]、
[./DoGEdge lena.bmp ex22_1_2.bmp 10 0.5 3]、
[./DoGEdge lena.bmp ex22_1_3.bmp 10 0.5 4]、
[./DoGEdge lena.bmp ex22_1_4.bmp 10 0.5 5]を実行して!

Shin Yoshizawa: shin@riken.jp

演習:Ex22-1

- ✓ DoGEdge2.cxx: DoGエッジと元画像の合成(引数3, DoGEdgeと同じ): DoGEdgeと同じパラメータで出力ファイル名を変えて実行してみましょう!
- ✓ Style.cxx: Artistic Stylization画像の作成(引数11).
./Style 畳み込み半径(int) DoG標準偏差(double) DoGバンド幅(double) Bilateralフィルタ空間標準偏差(double) Bilateralフィルタ輝度標準偏差(double) Bilateralフィルタ繰り返し回数(int) HSV量子化数(int) HSV量子化V強調パラメータ(double) RGB量子化数(int)
[./Style lena.bmp ex22_st_1.bmp 0 0.5 3.0 25.0 0.1 3 16 0.7 4]と
[./Style lena.bmp ex22_st_1.bmp 10 0.5 5.0 25.0 0.1 3 16 0.7 4]で実行してみましょう!
- ✓ 自分の画像でDoGEdge.cxxとStyle.cxxをパラメータを調節してスタイリッシュな画像にしてみてください。

Shin Yoshizawa: shin@riken.jp


演習:Ex22-2

- ✓ DoGVideoEdge.cxxとDoGVideoEdge2.cxxを編集し、連番画像のDoGエッジ動画を作成するプログラムを完成せよ。
ヒント: ファイル内のコメントとDoGEdge.cxxをよく見てみてください。

$$g_{\sigma}(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad g_h(t) = \frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{t^2}{2h^2}\right)$$

$$DoG_{\sigma,K,h}(x, y, t) = g_h(t)(g_{\sigma}(x, y) - g_{K\sigma}(x, y))$$

- ✓ ↑と同じでもOK、自分独自の拡張でもOK、ただし単純拡張はダメ。
- ✓ ↑は⇒の様に残像だけなので、評価時刻でのDoGエッジも出るようにデザインすると高得点!



Shin Yoshizawa: shin@riken.jp

補足:

- ✓ 先週の演習Ex14.zip内のVideoIO.cxxでカレントフォルダの指定は、削除してください。以下の3カ所です。

✓49,67,98行目: ./%s/%s ⇒ %s/%s

現在のEx14.zipは修正してありますm(_ _)m

Shin Yoshizawa: shin@riken.jp

来週の手配

- ✓ 動画処理その3(12/21).

