

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その2

第12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/briect/Yoshizawa/Lectures/index.html](http://www.riken.jp/briect/Yoshizawa/Lectures/index.html)  
[www.riken.jp/briect/Yoshizawa/Lectures/Lec25.pdf](http://www.riken.jp/briect/Yoshizawa/Lectures/Lec25.pdf)

- Artistic Stylization ⇒ Video Stylization
- 演習:
  - 量子化画像 **1/25補講5限(金曜日)**
  - 量子化ビデオ **6218実習室**
  - Artistic Video Stylization

レポート06(来週: 1/16 〆切).

今日もプログラミングがメイン.

Shin Yoshizawa: shin@riken.jp

## 復習: 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.

© New Line Productions, Inc.

Shin Yoshizawa: shin@riken.jp

## 復習: 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 → 出力Stylized画像

エッジ抽出 (DoG) → エッジ画像

Shin Yoshizawa: shin@riken.jp

## スタイル化ビデオ

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

エッジ抽出 (DoG) → エッジ動画

Shin Yoshizawa: shin@riken.jp

## 時空間Bilateralフィルタ

✓ 単純に時間方向のガウス関数を追加するだけでOK.

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

$$Z(x,y) = g_r(|I(x) - I(y)|)g_s(|x - y|)g_t(|\alpha - \beta|)$$

$$g_a(r) = e^{-\frac{r^2}{a^2}}$$

Intensity Kernel    Spatial Kernel    Temporal Kernel

Shin Yoshizawa: shin@riken.jp

## 復習:ポスター化

✓ HSV空間の色相(H)で多値化し明度(V)を強調その後RGB毎に多値化を行う.

- 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

RGB量子化 → HSV量子化

Shin Yoshizawa: shin@riken.jp

## RGB⇔HSV変換

✓ RGB⇒HSV: 0~1に正規化されたRGBである画素値のRGBの最大をMAX、最小をMINとすると:

$$H = \begin{cases} 60 \times \frac{G-B}{MAX-MIN} + 0, & \text{if } MAX = R \\ 60 \times \frac{B-R}{MAX-MIN} + 120, & \text{if } MAX = G \\ 60 \times \frac{R-G}{MAX-MIN} + 240, & \text{if } MAX = B \end{cases} \quad V = \frac{MAX - MIN}{MAX}$$

- ただし、Hが負ならH = H + 360.

✓ HSV⇒RGB: mod XはXで割った余り、[X]は整数化.

$$H_i = \lfloor \frac{H}{60} \rfloor \pmod{6}$$

$$f = \frac{H}{60} - H_i$$

$$\begin{aligned} p &= V(1 - S) \\ q &= V(1 - fS) \\ t &= V(1 - (1 - f)S) \end{aligned}$$

if  $H_i = 0 \rightarrow R = V, G = t, B = p$   
 if  $H_i = 1 \rightarrow R = q, G = V, B = p$   
 if  $H_i = 2 \rightarrow R = p, G = V, B = t$   
 if  $H_i = 3 \rightarrow R = p, G = q, B = V$   
 if  $H_i = 4 \rightarrow R = t, G = p, B = V$   
 if  $H_i = 5 \rightarrow R = V, G = p, B = q$

Shin Yoshizawa: shin@riken.jp

## RGB毎、色相Hの量子化と明度Vの強調

✓ RGB毎、色相(H)をN個の値に線形変換する:

- N/最大値を乗算.
- 整数にする(四捨五入).
- 最大値/Nを乗算.

✓ 明度(V)の強調: alphaはパラメータ,

- 0~1に正規化する.
- $V \leftarrow 20V - 10$ .
- $V \leftarrow 255 / (1 + \exp(-\alpha V))$

16色、強調なし

$\alpha$ : 0.1 0.2 0.3 0.4 0.5

Shin Yoshizawa: shin@riken.jp

## 色相Hの量子化

入力: 256<sup>3</sup>色

2色 3色 4色

5色 6色 7色 8色

V強調なし、明度・彩度は256段階.

Shin Yoshizawa: shin@riken.jp

## 色相Hの量子化

2色 3色

4色 6色

7色 8色

Shin Yoshizawa: shin@riken.jp

## H量子化後にRGBの量子化

入力: 256<sup>3</sup>色

2<sup>3</sup>色 3<sup>3</sup>色 4<sup>3</sup>色

5<sup>3</sup>色 6<sup>3</sup>色 7<sup>3</sup>色 8<sup>3</sup>色

色相Hは16段階、明度V強調0.5、彩度は256段階.

Shin Yoshizawa: shin@riken.jp

### H量子化後にRGBの量子化

2<sup>3</sup>色      3<sup>3</sup>色  
4<sup>3</sup>色      6<sup>3</sup>色  
7<sup>3</sup>色      8<sup>3</sup>色

色相Hは16段階、  
明度V強調0.2

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];
```

3D画像の配列表現

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

Shin Yoshizawa: shin@riken.jp

### 量子化の3D拡張

✓ そのままの拡張は時間変化に弱いので、時間方向の半径を考えて、その半径内(部分画像毎)に量子化を実行する:  
✓ 例えばHSV量子化では...

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

単純な3D化

```
for(i = 0; i < st; i++){
  for(t = i - r; t <= i + r; t++){
    for(j = 0; j < sy; j++){
      for(k = 0; k < sx; k++){
        I[t][j][k] = ...
      }
    }
  }
}
```

RGB⇒HSV  
Hの多値化+Vの強調  
HSV⇒RGB

Shin Yoshizawa: shin@riken.jp

### 量子化の3D拡張2

入力: 256<sup>3</sup>色

色相Hは16段階、  
明度V強調0.5  
時間半径16

Shin Yoshizawa: shin@riken.jp

### 量子化の3D拡張3

入力: 256<sup>3</sup>色

色相Hは16段階、  
明度V強調0.5  
RGB各4段階  
時間半径16

Shin Yoshizawa: shin@riken.jp

### 量子化の3D拡張4

入力: 256<sup>3</sup>色

時間方向の半径4

色相H1色

色相H4色V強調0.2

色相H4色V強調0.2RGB各4段階

Shin Yoshizawa: shin@riken.jp

## 量子化の3D拡張5



色相Hは16段階、明度V強調0.5  
RGB各4段階  
時間半径16

### Bilateralフィルタ 3回適用後を入力

Shin Yoshizawa: shin@riken.jp

## 演習:量子化ビデオ、スタイルビデオ

[www.riken.jp/brict/Yoshizawa/Lectures/index.html](http://www.riken.jp/brict/Yoshizawa/Lectures/index.html)  
[www.riken.jp/brict/Yoshizawa/Lectures/Lec25.pdf](http://www.riken.jp/brict/Yoshizawa/Lectures/Lec25.pdf)  
[www.riken.jp/brict/Yoshizawa/Lectures/Ex16.zip](http://www.riken.jp/brict/Yoshizawa/Lectures/Ex16.zip)

1. スタイルビデオプログラムの作成.
2. Lec23.pdfとLec24.pdfの演習が出来れば簡単.

✓ 実行例:  
 ./VideoStyle ../LV3\_5 ./Test output 10 0.5 3.0 25.0 0.1 3 1 1.0 16 0.4 4

Shin Yoshizawa: shin@riken.jp

## 演習:Ex25-2

✓ VideoStyle.cxxを編集し、連番画像のスタイル動画を作成するプログラムを完成せよ.

**ヒント1:** ファイル内のコメントとStyle.cxxをよく見てみてください。Bilateralフィルタの3次元化はBilateral.hに入っているの編集しなくてもOK。量子化も答えが関数として書いてあるので、Lec24.pdfのDoGと同じ事をするだけ。

✓ パラメータを色々工夫してスタイリッシュな動画を作成してみましょう。

✓ ヒント2: makeDoGImage()とConv()の中を書いたら、makeでコンパイルし実行。引数は14個。

Shin Yoshizawa: shin@riken.jp

## 演習:Ex25-2

1. 入力フォルダ一名
2. 出力フォルダ一名
3. 出力ファイル名(.bmpなし)
4. 畳み込みの半径(int)
5. DoGの半径(double)
6. DoGのバンド幅(double)
7. Bilateralフィルタの空間標準偏差(double)
8. Bilateralフィルタの輝度値標準偏差(double)
9. Bilateralフィルタの繰り返し回数(int)
10. 時間方向畳み込み半径(int)
11. Bilateral&DoGの時間標準偏差(double)
12. HSVのHを多値化する数(int)
13. HSV量子化のV強調パラメータ(double)
14. RGBを多値化する数(int)

Shin Yoshizawa: shin@riken.jp

## 来週(1月16日)の予定

**内容(14-15):**  
**エッジ・形状・特徴抽出とパターン認識の基礎**  
**微分幾何学の基礎、形状検出、特徴量、判別・識別、学習等。**

1回	画像フォーマット
2回	周波数分解
3回	
4回	
5回	フィルタ処理・エッジ強調
6回	
7回	
8回	計算Photography・Artistic Stylization
9回	
10回	動画像処理
11回	
12回	
13回	
14回	エッジ・形状・特徴抽出とパターン認識の基礎
15回	

