

位相限定相関法について

マルチメディア情報検索参考資料

2018 年 10 月 25 日

位相限定相関法 (1/2)

2 枚のサイズ $N_1 \times N_2$ 画像をそれぞれ $f(n_1, n_2)$, $g(n_1, n_2)$ とし, $f(n_1, n_2)$ と $g(n_1, n_2)$ の離散フーリエ変換をそれぞれ $F(k_1, k_2)$, $G(k_1, k_2)$ とする.

$$F(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} f(n_1, n_2) W_{N_1}^{k_1 n_1} W_{N_2}^{k_2 n_2}$$
$$G(k_1, k_2) = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} g(n_1, n_2) W_{N_1}^{k_1 n_1} W_{N_2}^{k_2 n_2}$$

ただし, 正整数 N に対して $W_N = e^{-j2\pi/N}$. フーリエ係数は一般に複素数であることを思い出そう.

位相限定相関法 (2/2)

複素数 $z = x + jy$ の複素共役 \bar{z} と絶対値 $|z|$ を思い出そう.

$$\bar{z} = x - jy, \quad |z| = \sqrt{x^2 + y^2}$$

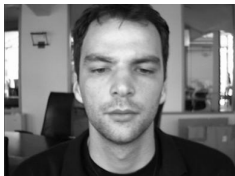
合成位相スペクトルを次式で定義する.

$$R(k_1, k_2) = \frac{F(k_1, k_2) \overline{G(k_1, k_2)}}{|F(k_1, k_2) \overline{G(k_1, k_2)}|}$$

目的の **Phase-only correlation(POC)** 関数は $R(k_1, k_2)$ の逆フーリエ変換として定義される.

$$r(n_1, n_2) = \frac{1}{N_1 N_2} \sum_{k_1=0}^{N_1-1} \sum_{k_2=0}^{N_2-1} R(k_1, k_2) W_{N_1}^{-k_1 n_1} W_{N_2}^{-k_2 n_2}$$

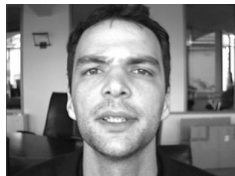
具体例



(a) DB 画像 A



(b) DB 画像 B

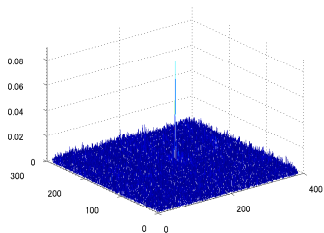


(c) クエリ画像 X

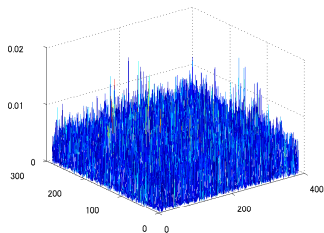
Figure : データベースとクエリの画像

POC 計算結果

入力画像 X と DB 画像 A と B の POC 関数を示す.



(a) A と X の POC 関数



(b) B と X の POC 関数

Figure : POC 関数の計算結果

プログラム例 (1/3) : POC を用いた識別

```
function poc_matching(DB, X)
```

%位相限定相関法を用いた識別アルゴリズム.

```
    dblX=double(X);
```

```
    for i=1:200
```

```
        A=DB(:, :, i);
```

```
        dblA=double(A);
```

```
        P=poc(dblA,dblX); %POC 関数本体
```

```
        distance(i)=max(max(P)); %POC 関数の最大値
```

```
    end
```

```
    [maximum, index]=max(distance); %全 POC 関数の最大値
```

```
    number=ceil(index/10);
```

```
    sprintf('X is Person #%d.', number)
```

```
end
```

プログラム例 (2/3) : POC 関数本体

```
function [P]=poc(A, B)
```

%入力 A と B の POC 関数を計算し, その絶対値を返す.

```
    A2=fft2(A); %フーリエ変換
```

```
    A3=A2./modified_abs(A2); %位相成分を計算
```

```
    B2=fft2(B); %フーリエ変換
```

```
    B3=conj(B2)./modified_abs(B2); %複素共役+位相計算
```

```
    C=ifft2(A3.*B3); %POC 関数の計算
```

```
    C2=fftshift(C); %原点シフト
```

```
    P=abs(C2); %絶対値の計算
```

```
    %figure
```

```
    %mesh(P)
```

```
end
```

プログラム例 (3/3) : 修正絶対値関数

```
function [Y]=modified_abs(X)
```

%出力 Y は入力行列 X の各要素をその絶対値で置き換えた行列.

%ただし, 絶対値が 0 の場合は 1 に補正.

```
[m,n]=size(X); %行列 X のサイズを抽出  
Y=abs(X); %行列 X の各成分の絶対値を計算  
for i=1:m  
    for j=1:n  
        if Y(i,j)==0.0  
            Y(i,j)=1.0;  
        end  
    end  
end  
end
```