

画像処理

物体の検出

宮崎大学 工学部 情報システム工学科

第 12 回

ハフ変換の拡張

ハフ変換

- ▶ 直線をパラメータで表現
- ▶ エッジ点を見つけたらパラメータ空間に投票
- ▶ 投票の集積点から直線を決定

⇒ パラメータで表現できれば、直線以外でも検出できる

例：円 (3 パラメータ)

$$(x - a)^2 + (y - b)^2 = r^2$$

例：楕円 (4 パラメータ)

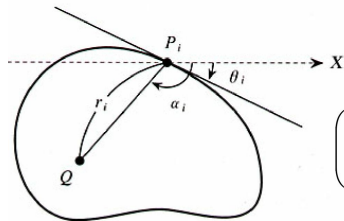
$$\left(\frac{x - a}{c}\right)^2 + \left(\frac{y - b}{d}\right)^2 = 1$$

一般化ハフ変換

任意の形状をもつ物体をパラメータ表現し、
その物体の位置・回転角・拡大率を投票により求める

形状定義

- ▶ 基準点 Q を定義
- ▶ 輪郭線上の各点 P_i から基準点 Q へ方向ベクトル $(r_i \cos \alpha_i, r_i \sin \alpha_i)$ を求める。
- ▶ 点 P_i での接線方向 θ 毎に, $(r_i(\theta), \alpha_i(\theta))$ を記録

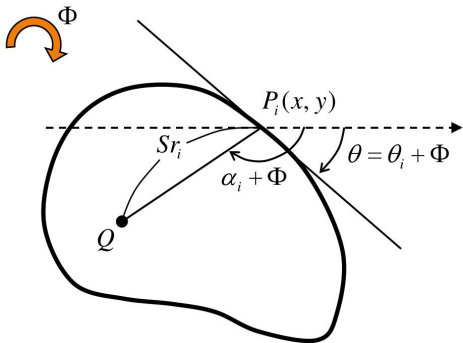


$$\begin{pmatrix} x_{P_iQ} \\ y_{P_iQ} \end{pmatrix} = \begin{pmatrix} r_i \cos \alpha_i \\ r_i \sin \alpha_i \end{pmatrix}$$

接線方向	基準点の位置
θ_1	(r_1, α_1)
θ_2	(r_2, α_2)
θ_3	$(r_3, \alpha_3), (r_{15}, \alpha_{15})$
θ_4	(r_4, α_4)
.	.
.	.

一般化ハフ変換-パラメータ空間

基準点位置 (u, v) , 拡大・縮小 s , 回転 Φ を考慮
 $\Rightarrow 4$ 次元



$$u = x + sr_i \cos(\alpha_i + \Phi)$$

$$v = y + sr_i \sin(\alpha_i + \Phi)$$

一般化ハフ変換—投票

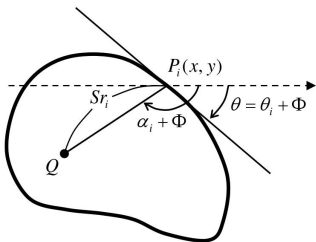
- ▶ 画像中のエッジ点 (x_j, y_j) とその勾配方向 ψ_j を求める．勾配方向 ψ_j から接線方向 θ_j を求める（勾配方向に垂直）
- ▶ 全ての回転角 ϕ , 拡大・縮小 s の組み合わせについて、次式により基準点位置 (u, v) を計算する

$$u = x_j + r(\theta_j - \phi)s \cos(\alpha(\theta_j - \phi) + \phi)$$

$$v = y_j + r(\theta_j - \phi)s \sin(\alpha(\theta_j - \phi) + \phi)$$

- ▶ (ϕ, s, u, v) に投票する
- ▶ 全てのエッジ点について投票したら、パラメータ空間の集積点を求め、物体を検出する

一般化ハフ変換-投票



$$\theta - \Phi$$

接線方向	基準点の位置
θ_1	(r_1, a_1)
θ_2	(r_2, a_2)
θ_3	$(r_3, a_3), (r_{15}, a_{15})$
θ_4	(r_4, a_4)
.	.
.	.

一般化ハフ変換-特徴

- ▶ 任意の輪郭形状を持った物体を取り扱い可能
- ▶ 輪郭が途切れていたり，一部遮蔽されていても検出可能
- ▶ 物体が拡大・縮小されていたり，回転していても検出可能
- ▶ パラメータ空間が4次元で，計算量が多い

テンプレートマッチング

入力画像とテンプレート画像を重ね合わせるにより比較照合し、両者が一致しているか判定する手法

物体の位置検出: 最も一致している場所を求める

物体の種別判定: それぞれの物体のテンプレートと比較し、最も一致するテンプレートの物体に分類

の特徴にビデオのダビン
起こると、探索精度が低
ティンクを提案する. こ

と

相違度

入力画像: $f(u, v)$ テンプレート $t(x, y)$

$$SAD(u, v) = \sum_{(x,y)} |t(x, y) - f(x + u, y + v)|$$

$$SSD(u, v) = \sum_{(x,y)} (t(x, y) - f(x + u, y + v))^2$$

- ※ SAD: Sum of Absolute Differences
(画素値の差の絶対値の和)
- ※ SSD: Sum of Squared Differences
(画素値の差の2乗の和)

入力画像とテンプレートが完全に一致する場合には SAD, SSD が 0 となる

相互相関

入力画像: $f(u, v)$ テンプレート: $t(x, y)$

$$COR(u, v) = \sum_{(x,y)} t(x, y) f(x + u, y + v)$$

※ SSD と関連

入力画像とテンプレートが完全に一致する場合には COR は最大となる

正規化相互相関

入力画像: $f(u, v)$ テンプレート: $t(x, y)$

$$NCOR(u, v) = \frac{\sum_{(x,y)} (t(x, y) - \bar{t})(f(x + u, y + v) - \bar{f})}{\sqrt{\sum_{(x,y)} (t(x, y) - \bar{t})^2 \sum_{(x,y)} (f(x + u, y + v) - \bar{f})^2}}$$

$$\bar{t} = \sum_{(x,y)} t(x, y) / K^2$$

$$\bar{f} = \sum_{(x,y)} f(u + x, v + y) / K^2$$

K^2 : テンプレートの大きさ

画像の明るさ変化の影響をそれほど受けずに処理が行える.

プログラム例

入力画像 iimg, テンプレート画像 timg が与えられたとき, 両者の相違度が最小となる位置 (min_u, min_v) を求める.

```
int tmatch(K_IMAGE *iimg, K_IMAGE *timg, int *xpos, int *ypos)
{
    int xsize =k_xsize(iimg); int ysize =k_ysize(iimg);
    int txsize=k_xsize(timg); int tysize=k_ysize(timg);
    int min_val = 255 * txsize * tysize;
    int min_u, min_v;

    for(int v=0; v<ysize-tysize; v++) {
        for(int u=0; u<xsize-txsize; u++) {
            int val = calc_diff(iimg, timg, u, v);
            if (val < min_val) {
                min_u = u; min_v = v; min_val = val;
            }
        }
    }

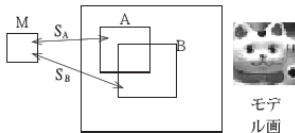
    *xpos = min_u; *ypos = min_v;
    return(min_val);
}
```

アクティブ探索

テンプレート画像全体と入力画像の対応する部分全体それぞれから、特徴量(ヒストグラム)を計算し特徴量の比較により照合する

利点: 処理時間の短縮

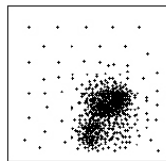
24時間のデータ中から 15秒のCMを照合
⇒ 1秒程度の処理時間



(a) アクティブ探索法の概念



モデル画



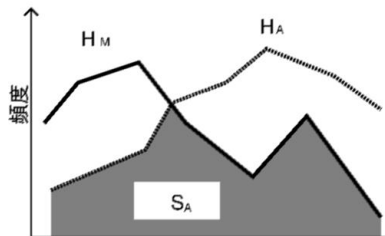
(b) アクティブ探索法により探索された箇所

類似度の計算

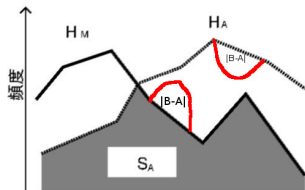
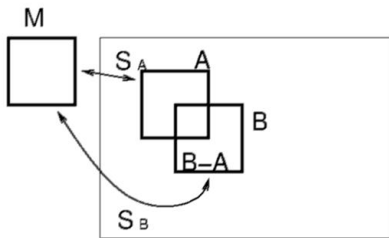
ヒストグラム: 画像中に現れる画素値の頻度

ヒストグラムを用いた類似度計算
(ヒストグラムインターセクション)

$$S_A = \sum_{l=1}^L \min(h_{Ml}, h_{Al})$$



アクティブ探索の原理



$$S_B \leq S_A + |B - A| \leq S_{max}$$

類似度は最大でも上式しか変化しないので，その間の比較を飛ばせる。