
知能制御システム学

画像追跡 (2)
— 領域の追跡 —

東北大学 大学院情報科学研究科

鏡 慎吾

swk(at)ic.is.tohoku.ac.jp

2009.07.14

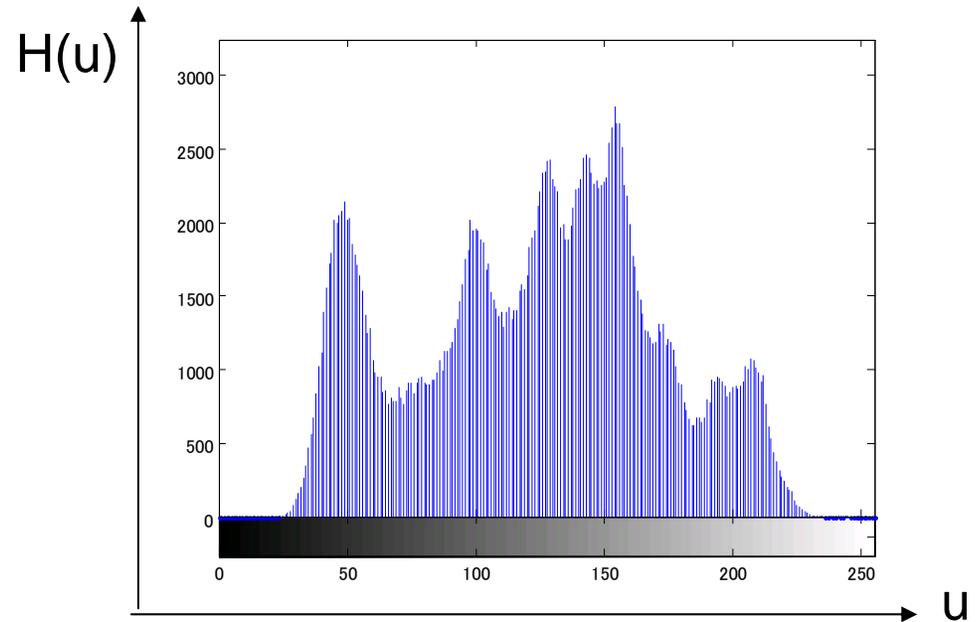
今日の内容

前回紹介した特徴点追跡では, 原理上, 自由な変形を含む領域全体を対象として追跡することは難しい(そのような領域全体を特徴点の集合として追跡することはできる).

今回は非剛体に適用可能な方法として, Mean Shift 法 (平均値シフト法) に基づく追跡アルゴリズム [Comaniciu 2003] を紹介する.

- ヒストグラムによる対象モデル
- ヒストグラムの類似度
- Mean Shift 法による類似度マップの勾配の計算と極大値探索 [Fukunaga 1975] [Comaniciu 2002]

(濃淡値)ヒストグラム



$$H = \{H_u\}_{u=1,2,\dots,m}, H_u = \sum_{x \in S(u)} 1$$

ただし画素値がビン u に含まれる画素の集合を $S(u)$ と書いた

$$p = \{p_u\}, p_u \propto H_u, \sum_{u=1}^m p_u = 1 \quad (\text{正規化ヒストグラム})$$

色ヒストグラム

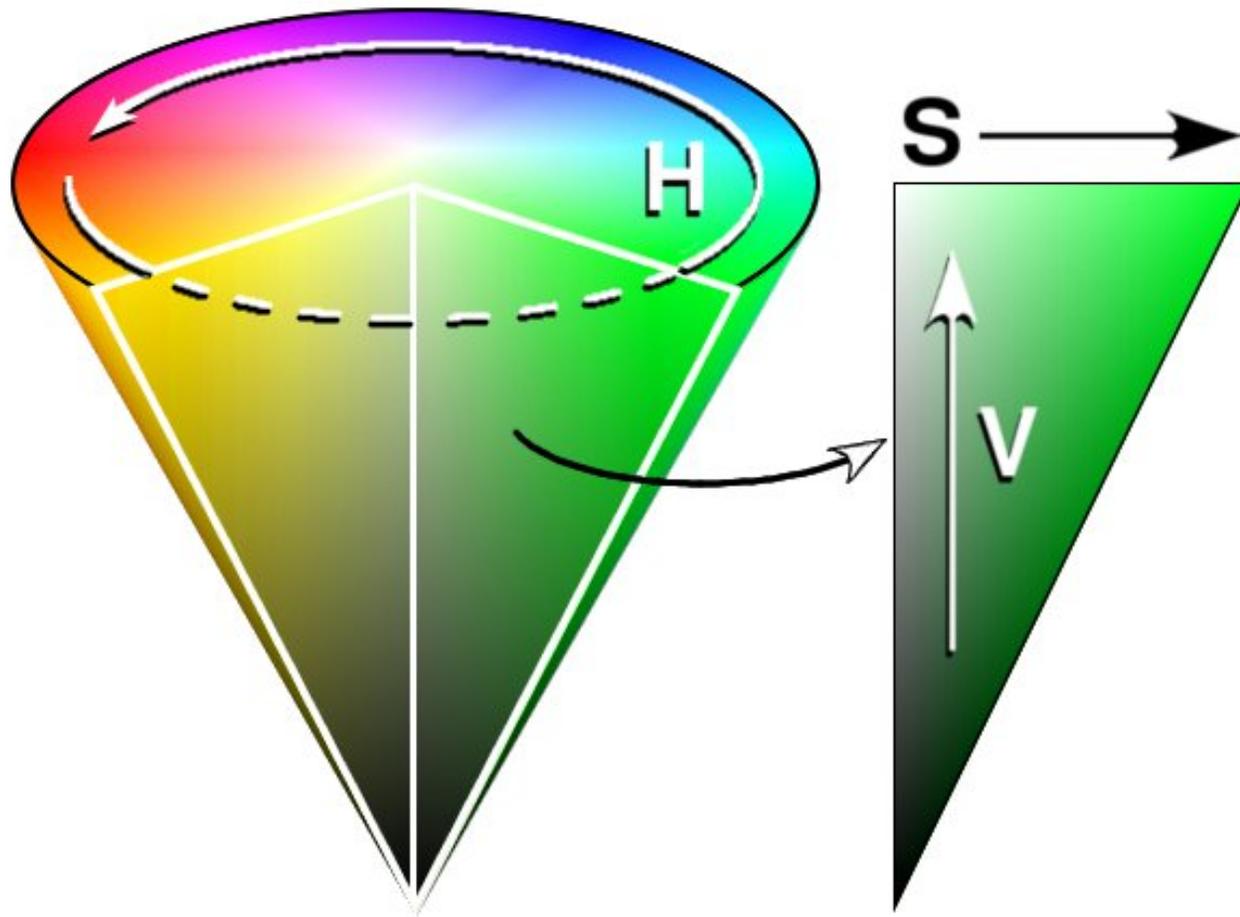
カラー画像の「画素の色」に対しても同様にヒストグラムを定義できる

例1) RGB 色空間の各成分を各 16 分割して, $16 \times 16 \times 16$ ビンのヒストグラムを作成

例2) HSV (色相(Hue), 彩度(Saturation), 明度(Value)) 色空間で, Hue と Saturation だけを各 64 分割して, 64×64 ビンのヒストグラムを作成

例3) RGB色空間の各成分「のみ」のヒストグラム(例えば64ビン)を考えて, それらを連結して 3×64 ビンのヒストグラムを作成

参考: HSV色空間



http://ja.wikipedia.org/wiki/%E7%94%BB%E5%83%8F:HSV_cone.jpg

ヒストグラム間の類似度

領域の特徴としてヒストグラムを採用して追跡を行うには、追跡対象のモデルのヒストグラムと、任意の画像領域のヒストグラムが「どのくらい似ているか」を評価する必要がある

代表的な類似度: Bhattacharyya係

正規化ヒストグラム \mathbf{p} , \mathbf{q} に対して以下の通り定義する

$$\rho(\mathbf{p}, \mathbf{q}) = \sum_{u=1}^m \sqrt{p_u q_u}$$

(つまり単位ベクトル $(\sqrt{p_1}, \sqrt{p_2}, \dots, \sqrt{p_m})^T$ と $(\sqrt{q_1}, \sqrt{q_2}, \dots, \sqrt{q_m})^T$ の内積. 完全に同一のとき、最大値 1 をとる)

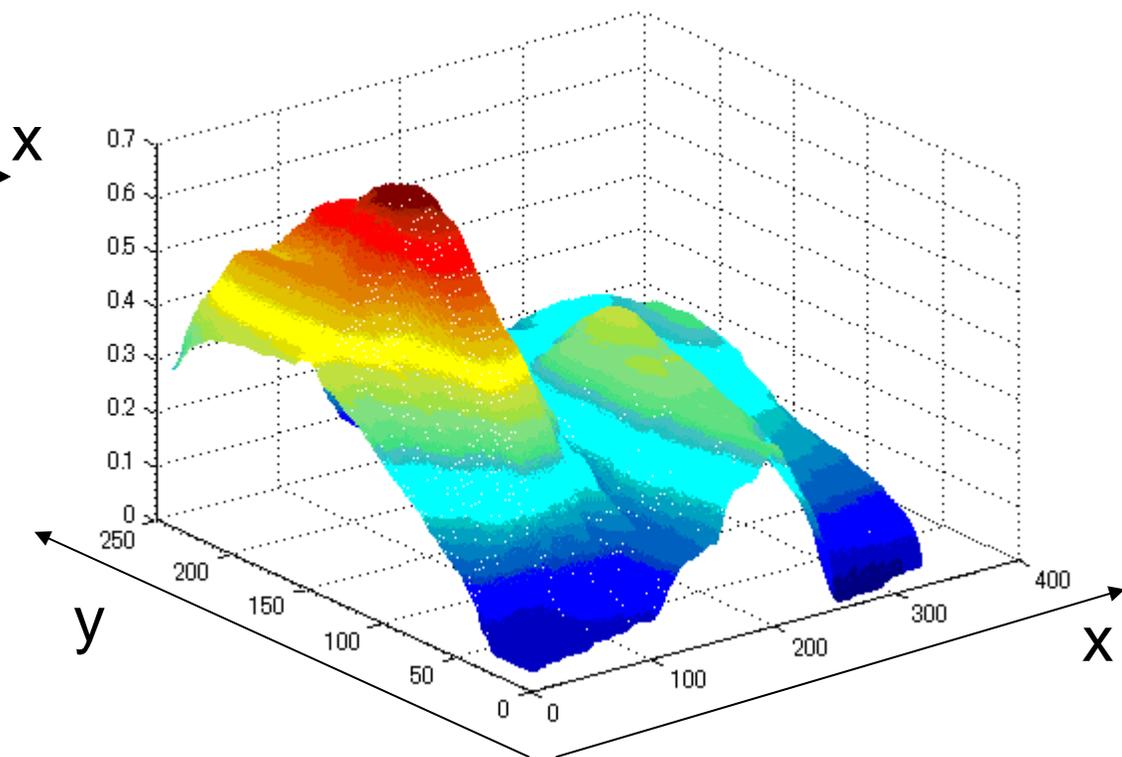
類似度マップ



現在の
入力画像



対象モデル



重みつきヒストグラムによる対象モデル

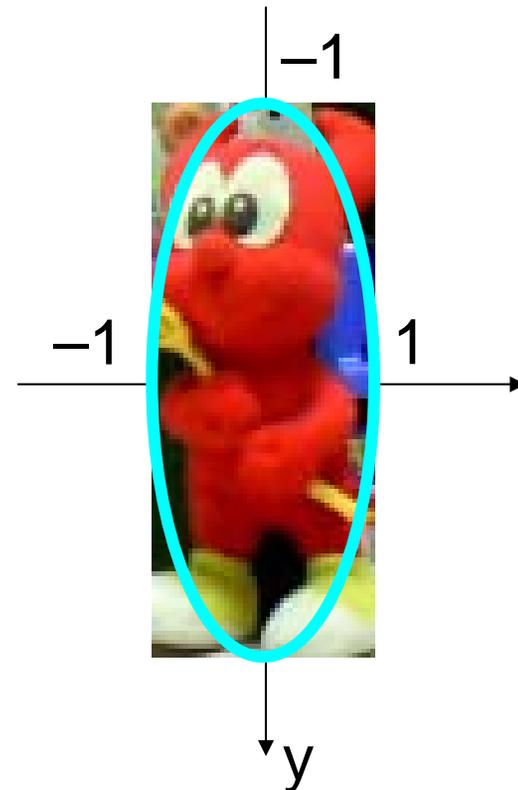
- 追跡する領域の端の方はあまり考慮したくない
- 類似度マップが不連続になると扱いにくい
→ 適当な重みをつけてヒストグラムを計算する

対象モデル(一般性を失わず, 原点を中心とする)

$$q_u \propto \sum_{x \in S_0(u)} k(\|x\|^2)$$

$S_0(u)$: 初期画像において, 画素値がビン u に属する画素の集合
 k : 重み関数(カーネル関数)

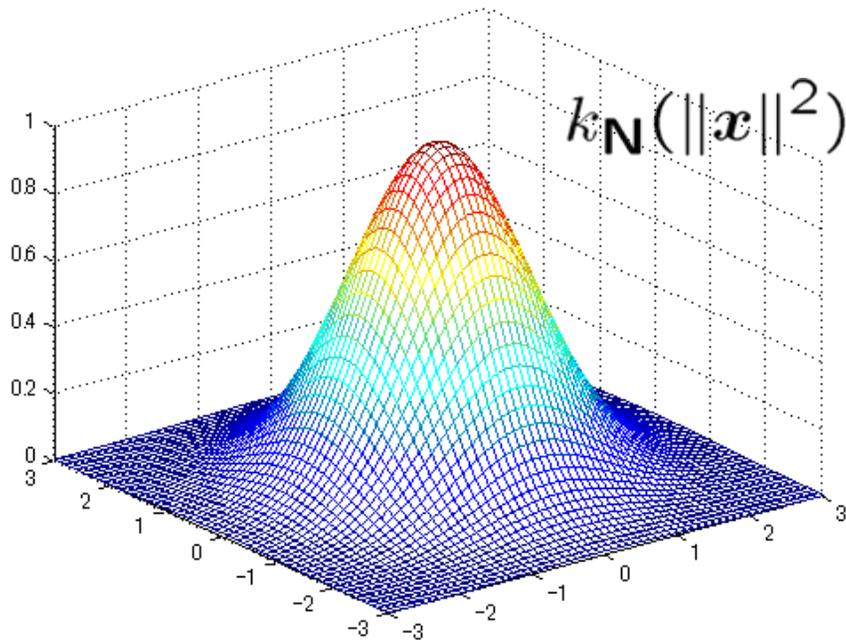
ただし画像座標 x はモデル領域が単位円に収まるように正規化する. 以下同様



重み関数(カーネル関数)の例

例1: Gauss カーネル

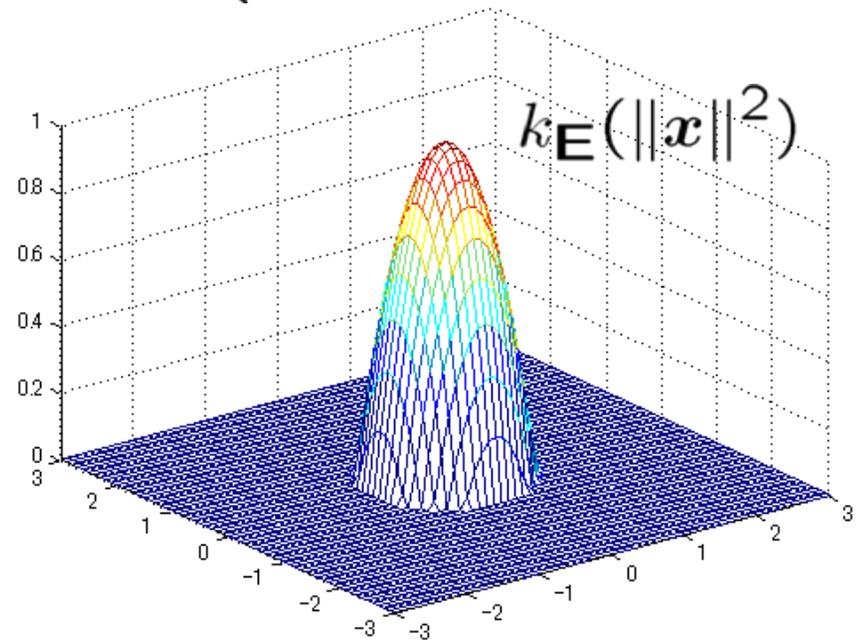
$$k_{\mathbf{N}}(x) \propto \exp\left(-\frac{x}{2}\right)$$



例2: Epanechnikov カーネル

(正確には, kernel with Epanechnikov profile)

$$k_{\mathbf{E}}(x) \propto \begin{cases} 1 - x, & x \leq 1 \\ 0, & \text{otherwise} \end{cases}$$



追跡領域候補の重みつきヒストグラム

追跡領域候補(中心座標 \mathbf{y})

$$p_u(\mathbf{y}) \propto \sum_{\mathbf{x} \in S(u)} k(\|\mathbf{y} - \mathbf{x}\|^2)$$



$S(u)$: 現画像において, 画素値がビン u に属する画素の集合

\mathbf{y} を中心とした単位円になるように正規化

画像座標 \mathbf{x} は同様に正規化されていることに注意

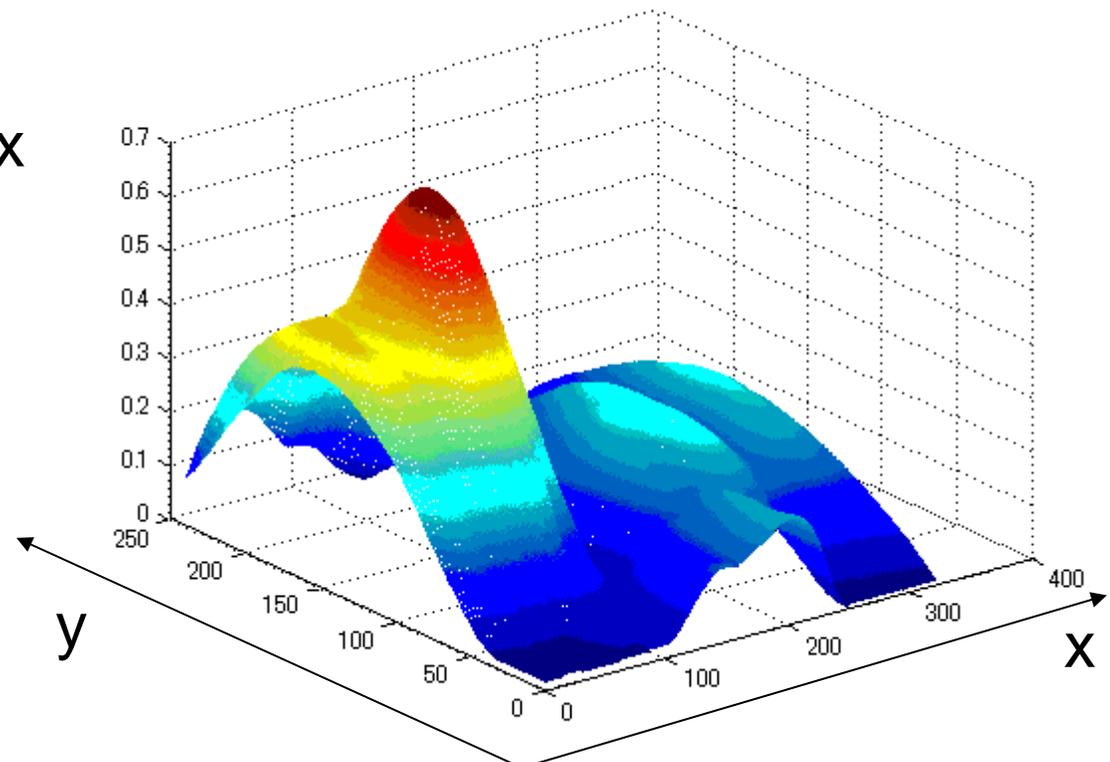
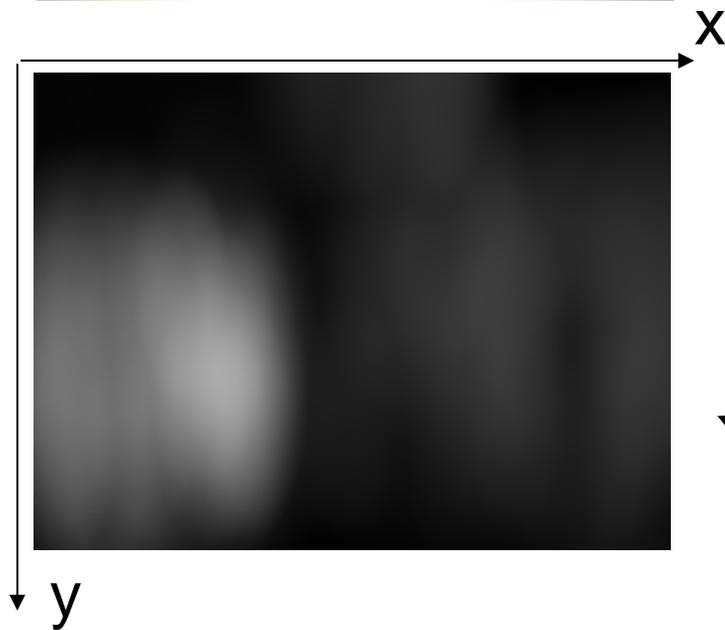
重みつきヒストグラムの場合の類似度マップ



現在の
入力画像



対象モデル



類似度極大点の探索

類似度マップをすべて計算するのは時間がかかり過ぎるので、その勾配を求めて、勾配方向に探索していくことにする

まず探索の開始位置を \mathbf{y}_0 とする(前フレームでの位置などをとればよい). \mathbf{y}_0 に近い \mathbf{y} では、ヒストグラムの値 $p_u(\mathbf{y})$ も $p_u(\mathbf{y}_0)$ に近いと考えると、Bhattacharyya 係数を Taylor 展開すると

$$\begin{aligned}\rho(\mathbf{p}(\mathbf{y}), \mathbf{q}) &= \sum_u \sqrt{p_u(\mathbf{y})} \sqrt{q_u} \\ &\approx \sum_u \sqrt{q_u} \left\{ \sqrt{p_u(\mathbf{y}_0)} + \frac{1}{2} p_u(\mathbf{y}_0)^{-1/2} (p_u(\mathbf{y}) - p_u(\mathbf{y}_0)) \right\} \\ &= \sum_u \sqrt{q_u} \left(\sqrt{p_u(\mathbf{y}_0)} + \frac{1}{2} p_u(\mathbf{y}) \frac{1}{\sqrt{p_u(\mathbf{y}_0)}} - \frac{1}{2} \sqrt{p_u(\mathbf{y}_0)} \right) \\ &= \frac{1}{2} \sum_u \sqrt{q_u} \sqrt{p_u(\mathbf{y}_0)} + \frac{1}{2} \sum_u p_u(\mathbf{y}) \frac{\sqrt{q_u}}{\sqrt{p_u(\mathbf{y}_0)}}\end{aligned}$$

第1項は \mathbf{y} に依存しないので, 類似度を最大にする \mathbf{y} を探す

問題は第2項
$$\frac{1}{2} \sum_u p_u(\mathbf{y}) \frac{\sqrt{q_u}}{\sqrt{p_u(\mathbf{y}_0)}}$$

を最大化する問題になる. ここで追跡領域候補のヒストグラムが

$$p_u(\mathbf{y}) \propto \sum_{\mathbf{x} \in S(u)} k(\|\mathbf{y} - \mathbf{x}\|^2)$$

だったことを思い出すと, 問題は以下の関数の最大化に帰着する

$$\begin{aligned} & \sum_{u \in \text{全ビン}} \sum_{\mathbf{x} \in \text{ビン } u \text{ に属する全画素}} k(\|\mathbf{y} - \mathbf{x}\|^2) \frac{\sqrt{q_u}}{\sqrt{p_u(\mathbf{y}_0)}} \\ = & \sum_{\mathbf{x} \in \text{全画素}} \sqrt{\frac{q_{b(\mathbf{x})}}{p_{b(\mathbf{x})}(\mathbf{y}_0)}} k(\|\mathbf{y} - \mathbf{x}\|^2) \quad b(\mathbf{x}) \text{ は画素 } \mathbf{x} \text{ が属するビン} \end{aligned}$$

最後の変形が納得行かない人は、以下のように考えてもよい。
 $\delta[\cdot]$ はクロネッカのデルタ記号である。(原論文ではこの表記)

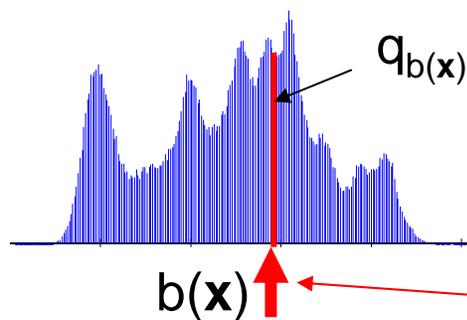
$$\begin{aligned}
 & \sum_{u \in \text{全ビン}} \sum_{x \in \text{ビン } u \text{ に属する全画素}} k(\|\mathbf{y} - \mathbf{x}\|^2) \frac{\sqrt{q_u}}{\sqrt{p_u(\mathbf{y}_0)}} \\
 = & \sum_{u \in \text{全ビン}} \sum_{x \in \text{全画素}} k(\|\mathbf{y} - \mathbf{x}\|^2) \frac{\sqrt{q_u}}{\sqrt{p_u(\mathbf{y}_0)}} \delta[b(\mathbf{x}) - u] \\
 = & \sum_{x \in \text{全画素}} \sum_{u \in \text{全ビン}} k(\|\mathbf{y} - \mathbf{x}\|^2) \frac{\sqrt{q_u}}{\sqrt{p_u(\mathbf{y}_0)}} \delta[b(\mathbf{x}) - u] \\
 = & \sum_{x \in \text{全画素}} \sqrt{\frac{q_{b(\mathbf{x})}}{p_{b(\mathbf{x})}(\mathbf{y}_0)}} k(\|\mathbf{y} - \mathbf{x}\|^2)
 \end{aligned}$$

ここまでのまとめ

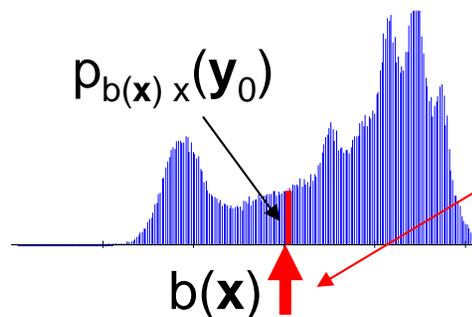
$$\sum_{x \in \text{全画素}} \sqrt{\frac{q_b(x)}{p_b(x)(y_0)}} k(\|y - x\|^2) = \sum_{x \in \text{全画素}} w(x) k(\|y - x\|^2)$$

の勾配方向に進んで行けばよい

対象モデルのヒストグラム



y_0 (探索開始位置)のヒストグラム



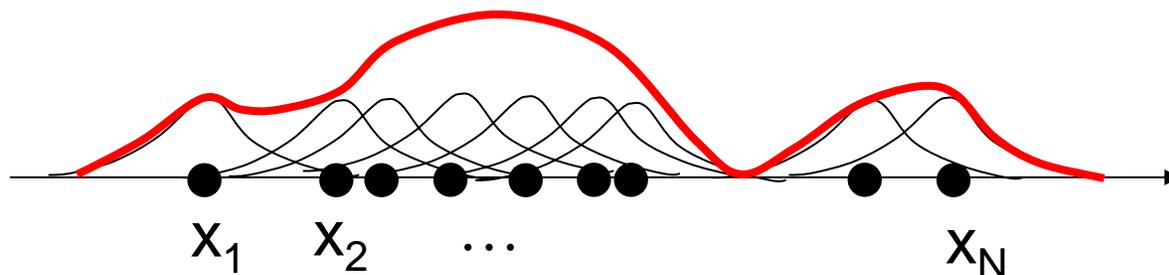
勾配をどのように求めるか?

カーネル密度推定法

ちょっと見方を変えて、 $k()$ が y のまわりではなく x のまわりに広がっていると考えると、各データ点 x を中心として、 $w(x)$ 倍されたカーネル $k()$ をたたみこんでいるとみなせる

$$\sum_{x \in \text{全画素}} w(x)k(\|y - x\|^2)$$

一般に、ある確率密度分布に従って発生した有限個のサンプルから、もとの(連続)確率密度分布を推定する際に用いられる考え方で、カーネル密度推定法、あるいは Parzen 推定法と呼ばれる



Mean Shift 法

カーネル密度推定法により求められる密度分布の勾配を計算する方法が Mean Shift 法である

$$f_k(\mathbf{y}) = \sum_x w(\mathbf{x})k(\|\mathbf{y} - \mathbf{x}\|^2)$$

重みつきサンプルからの
カーネル $k()$ による密度推定

$$\nabla f_k(\mathbf{y}) = \sum_x k'(\|\mathbf{y} - \mathbf{x}\|^2) \cdot 2(\mathbf{y} - \mathbf{x})w(\mathbf{x})$$

$$g(x) = -k'(x) \quad \text{と書いて}$$

$$\nabla f_k(\mathbf{y}) = 2 \sum_x g(\|\mathbf{y} - \mathbf{x}\|^2)(\mathbf{x} - \mathbf{y})w(\mathbf{x})$$

$$= 2 \left[\sum_x \left\{ \mathbf{x}w(\mathbf{x})g(\|\mathbf{y} - \mathbf{x}\|^2) \right\} - \mathbf{y} \sum_x \left\{ w(\mathbf{x})g(\|\mathbf{y} - \mathbf{x}\|^2) \right\} \right]$$

$$= 2 \sum_x \left\{ w(\mathbf{x})g(\|\mathbf{y} - \mathbf{x}\|^2) \right\} \left[\frac{\sum_x \left\{ \mathbf{x}w(\mathbf{x})g(\|\mathbf{y} - \mathbf{x}\|^2) \right\}}{\sum_x \left\{ w(\mathbf{x})g(\|\mathbf{y} - \mathbf{x}\|^2) \right\}} - \mathbf{y} \right]$$

$$m_g(\mathbf{y}) = \frac{\nabla f_g(\mathbf{y})}{2f_g(\mathbf{y})} \quad \text{where}$$

$$f_g(\mathbf{y}) = \sum_{\mathbf{x}} w(\mathbf{x}) g(\|\mathbf{y} - \mathbf{x}\|^2) \quad \text{カーネル } g() \text{ による密度推定 (スカラ)}$$

$$m_g(\mathbf{x}) = \left[\frac{\sum_{\mathbf{x}} \{ \mathbf{x} w(\mathbf{x}) g(\|\mathbf{y} - \mathbf{x}\|^2) \}}{\sum_{\mathbf{x}} \{ w(\mathbf{x}) g(\|\mathbf{y} - \mathbf{x}\|^2) \}} - \mathbf{y} \right] \quad \text{Mean Shift ベクトル} \\ \rightarrow \text{勾配方向}$$

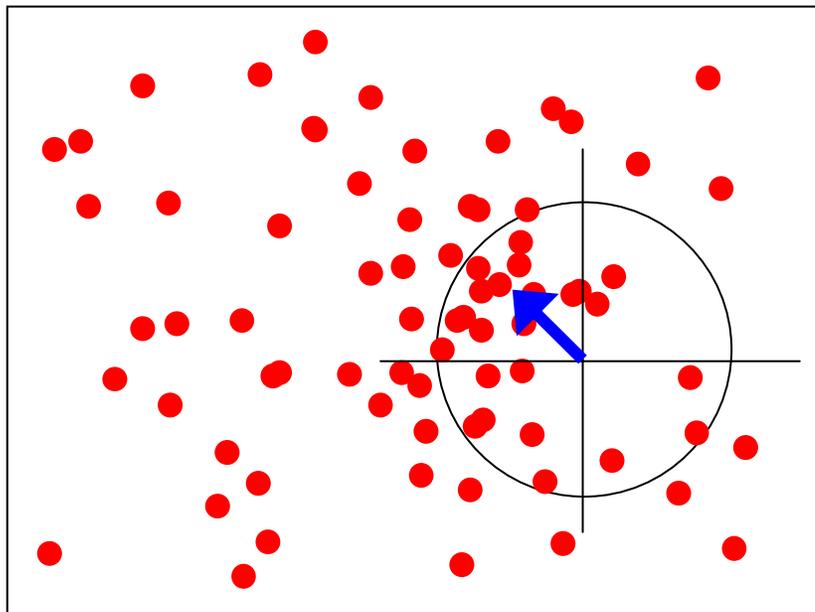
特に $k()$ が Epanechnikov カーネルの場合は $g()$ は単位円内で1, 外で0になり,

$$f_g(\mathbf{y}) = \sum_{\mathbf{x} \in \text{単位円}} w(\mathbf{x})$$

$$m_g(\mathbf{x}) = \left[\frac{\sum_{\mathbf{x} \in \text{単位円}} \mathbf{x} w(\mathbf{x})}{\sum_{\mathbf{x} \in \text{単位円}} w(\mathbf{x})} - \mathbf{y} \right]$$

Mean Shift 法の流れ

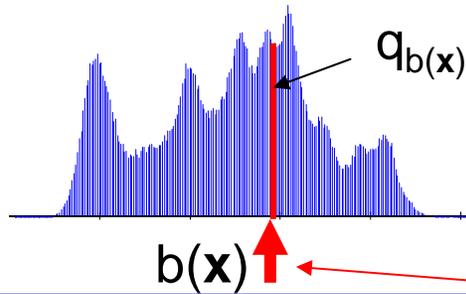
1. 現在位置の周囲でサンプルの重心を計算する
2. その重心まで移動する (Mean Shift)
3. 移動量が十分に小さくなければ, 1に戻る



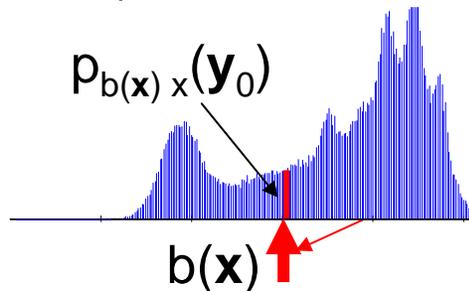
- 重心への移動 $m_g(\mathbf{y})$ は, $f_k(\mathbf{y})$ の勾配方向への移動であり, またその移動量は $f_g(\mathbf{y})$ に反比例する (推定密度の小さなところは粗く探索し, 推定密度が大きくなるに従って細かく探索するようになる)
- 適当な形状のカーネルを仮定すると, 移動先が $f_k(\mathbf{y})$ の mode に収束することが証明できる. Epanechnikov カーネルはその条件を満たす

Mean Shift 追跡法の流れ

対象モデルのヒストグラム



y_0 (現在位置)のヒストグラム



1. 現在位置の周囲の重みつきヒストグラムを計算
2. 現在位置の周囲の各画素について、 $\sqrt{q/p}$ を計算し、その重心に現在位置を移動する
3. 移動量が十分に小さくなければ、1に戻る

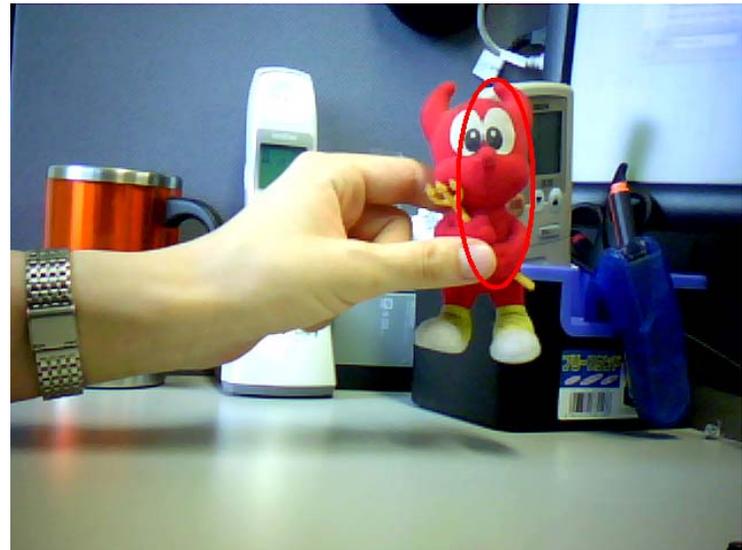
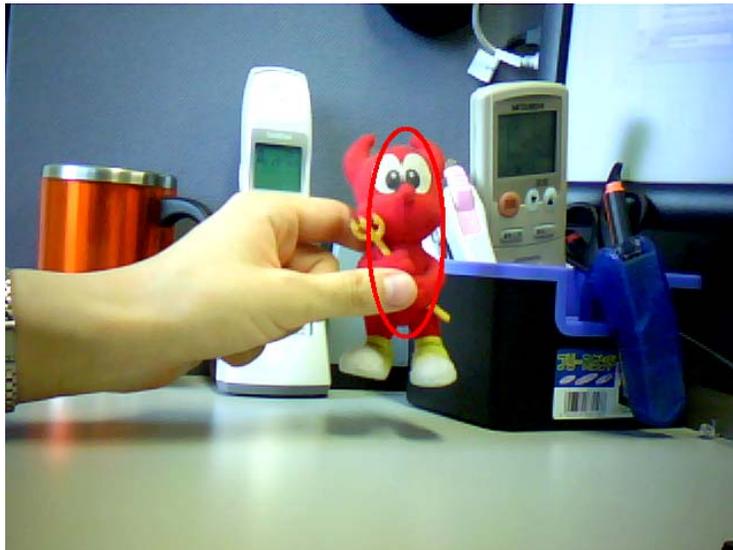
サンプルプログラム

sample program:

- mean_shift.cpp

OpenCV に関する注意:

- OpenCVには, CAMSHIFT [Bradski 1998] と呼ばれる領域追跡処理が標準で用意されている. これも Mean Shift 法に基づいているが, 今回紹介したものとは基本的に別物である(単色に近いものを追いかけるのに適しており, 例えば顔領域の追跡などに使える)
- cvMeanShift() 関数は CAMSHIFT 用に特化した仕様になっており, 今回紹介した Mean Shift 追跡法には使えないようである
- 標準のヒストグラム計算関数 cvCreateHist() では, 重みつきヒストグラムは計算できないようである



References

- D. Comaniciu, V. Ramesh and P. Meer: Kernel-Based Object Tracking, IEEE Trans. of Pattern Analysis and Machine Intelligence, vol.25, no.5, 2003.
- D. Comaniciu and P. Meer: Mean Shift: A Robust Approach Toward Feature Space Analysis, IEEE Trans. on Pattern Analysis and Machine Intelligence, vol.25, no.5, 2003.
- K. Fukunaga and L. D. Hostetler: The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition, IEEE Trans. on Information Theory, vol.IT-21, no.1, 1975.
- G. R. Bradski: Computer Vision Face Tracking for Use in a Perceptual User Interface, Intel Technical Journal, Q2, 1998.