統計的パターン認識とニューラルネット

汎化性能の高い非線形識別器の学習と画像認識への応用

産業技術総合研究所脳神経情報研究部門副研究部門長
筑波大学大学院システム情報工学研究科教授（連携）
栗田　多喜夫
takio-kurita@aist.go.jp

講演内容

- パターン認識とベイズ識別
  - パターン認識とは、ベイズ決定理論、密度関数の推定
- 線形識別関数の学習
  - 線形識別関数の性質、単純パーセプトロン、最小2乗判別関数の学習、ロジスティック回帰
- 統計的特徴抽出
  - 線形判別分析、非線形判別分析、非線形判別分析の線形近似、一般化線形判別分析
- 汎化性
  - 交差確認法、ブートストラップ、情報量基準、Shrinkage法、変数選択法、人工的なノイズの付加
- カーネル学習法
  - サポートベクターマシン、カーネルサポートベクターマシン、カーネル判別分析
- 非線形識別器の画像認識への応用

参考書・資料

- 参考書
  - R.O.Duda, P.E.Hart, and D.G.Stork, (尾上守夫監訳)、「パターン識別」、新技術コミュニケーションズ
  - 大津展之、栗田多喜夫、関田巌、「パターン認識—理論と応用—」、朝倉書店
  - C.M.Bishop, Pattern Recognition and Machine Learning,Springer, 2006.
  - S.Theodoridis, K.Koutroumbas, Pattern Recognition, Academic Press, 1999.
  - T.Hastie, R.Tibshirani, and SJ.Friedman, The Elements of Statistical Learning – Data Mining, Inference, and Prediction --
- 参考資料
  - 「パターン認識とニューラルネットワーク」
  - 「サポートベクターマシン入門」

    栗田のホームページ

    http://staff.aist.go.jp/takio-kurita/index-j.html

    からダウンロード可能

質問等

- 電子メール

  takio-kurita@aist.go.jp

- 連絡先

  〒305-8568 茨城県つくば市梅園1-1-1 つくば中央第2

  産業技術総合研究所 脳神経情報研究部門

  栗田 多喜夫

- 電話・FAX

  電話 029-861-5838 FAX 029-861-5841

パターン認識とベイズ識別

統計的パターン認識の基礎

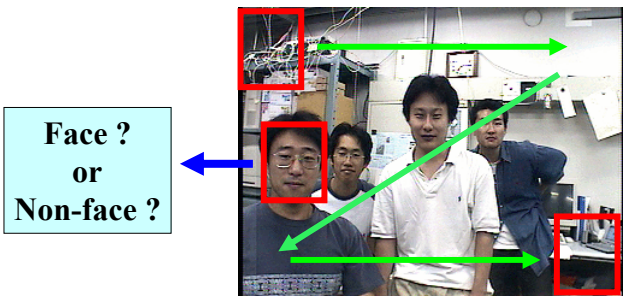
パターン認識の歴史

- パターン認識と人工知能
  - 認識や知能などの人間（生体）の脳の情報処理機能（知的情報処理機能）を解明し、それを機械（コンピュータ）で実現する試み
  - 情報処理技術に新たな概念を提供してきた
- 歴史
  - コンピュータ出現の初期
    - コンピュータは“万能機械”として、人間のあらゆる知的活動を代行してくれると期待 （チェスなどのゲーム、作曲、自動翻訳、定理証明などへの応用）
    - ニューロンモデル（McCulloch & Pitts, 1943)、パーセプトロン（Rosenblatt, 1957)
  - 1960年代～
    - コンピュータへの入力装置として、文字・図形・音声などの機械による認識（パターン認識）の試み ＝＞ まだまだ人間の能力には及ばない。
  - 1970年代～
    - 人工知能研究、第5世代コンピュータ（1982年～1992年）
  - 1980年代後半～
    - 誤差逆伝播学習法（Rumelhart, Hinton & Williams, 1986)、第2次ニューロブーム
    - リアルワールドコンピューティング（1992年～2002年）

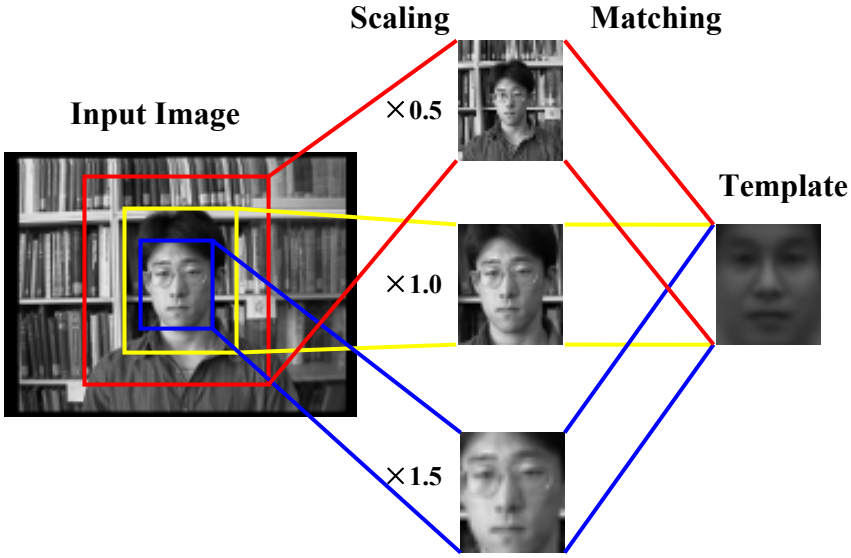
パターン認識問題の例

- スパムメイルを検出して、自動削除する
  - 特徴抽出
    - メイル本文やヘッダにどのような単語が現れているかの頻度を計測し、それらをまとめて特徴ベクトルとする
  - 訓練用のサンプルの作成
    - 過去のメイルのデータベースから特徴ベクトルを計測し、そのメイルがスパムかどうかを記録し、そのペアを訓練用サンプルデータとする
  - 識別器の学習
    - 訓練用のサンプルを用いて識別器のパラメータを学習する
  - 運用
    - 新たなメイルから特徴ベクトルを計測し、それを識別器に入力し、その結果がスパムであれば、そのメイルをスパムフォールダに移動する

画像中の顔の検出



大きさの変化への対応

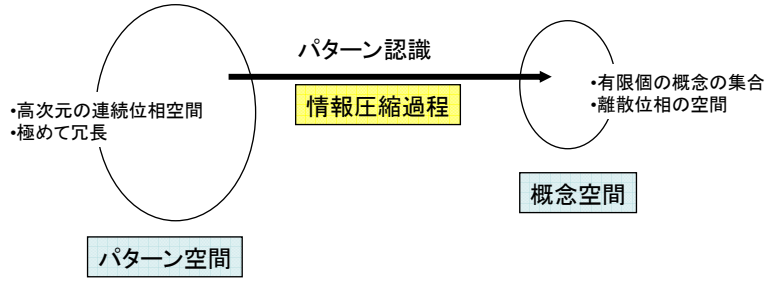


パターン認識問題の例

- ロボット
  - 顔、声から誰かを識別、音声から何を喋っているかを認識、手で触って、状態(柔らかい、硬い)を判定
- 車
  - 対向車や人の検出、運転者の状態(眠い、テンションがあがっている、、、)
- 医療
  - 検査結果から病気を推定(肺がん)
- 軍事
  - ソナーデータから潜水艦かどうかを識別
- ワイン
  - 成分からワインの種類を識別

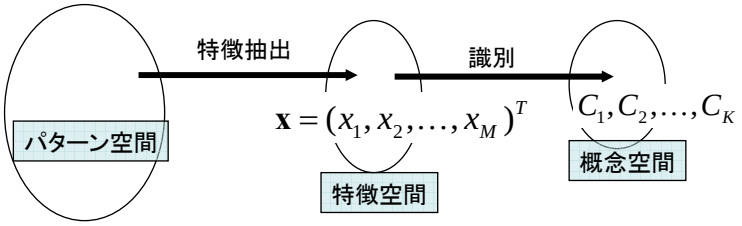
パターン認識とは

- パターン認識
  - 認識対象がいくつかの概念に分類出来るとき、観測されたパターンをそれらの概念（クラスあるいは類）のうちのひとつに対応させる処理
    - スパムメイルの検出： メイルをスパムメイルと通常のメイルに分類
    - 顔検出： 部分画像を顔か顔でないかに分類
    - 数字の認識： 入力パターンを10種類の数字のいずれかに対応させる
    - 顔画像の識別： 顔画像から誰であるかを推定する



パターン認識過程

- 特徴抽出
  - 認識対象から何らかの特徴量を計測（抽出）する必要がある
  - 認識に有効な情報（特徴）を抽出し、次元を縮小した効率の良い空間を構成する過程
    - 文字認識： スキャナ等で取り込んだ画像から文字の識別に必要な本質的な特徴のみを抽出（例、文字線の傾き、曲率、面積など）
- 識別
  - 与えられた未知の対象を、特徴とクラスの関係に関する知識に基づいて、どのクラスに属するかを決定（判定）する過程



パターン認識の基本課題

- 識別方式の開発
  - 未知の認識対象を観測して得られる特徴ベクトルからその対象がどのクラスに属するかを判定する方法
- 一般的なアプローチ
  - 教師あり学習
    - クラスの帰属が既知の学習用のサンプル集合から特徴ベクトルとクラスとの確率的な対応関係を知識として学習
  - 識別
    - 学習された特徴ベクトルとクラスとの対応関係に関する確率的知識を利用して、与えられた未知の認識対象を特徴ベクトルからその認識対象がどのクラスに属していたかを推定(決定)

ベイズ決定理論

- ベイズ識別方式
  - 特徴ベクトルとクラスとの確率的な対応関係が完全にわかっている理想的な場合の理論
  - 未知の認識対象を誤って他のクラスに識別する確率(誤識別率)を出来るだけ小さくするような識別方式
  - 誤識別率の意味で理論的に最適な識別方式
- 例:身長から男か女かを当てる

事前確率・条件付き確率

- 事前確率（先見確率）
  - クラス C_k の確率

$$P(C_k) \qquad \sum_{k=1}^{K} P(C_k) = 1$$

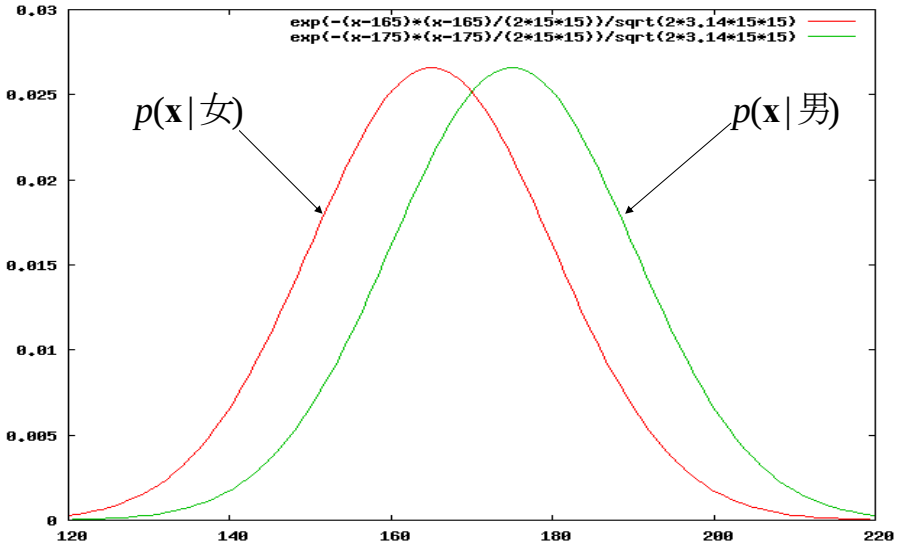
◆ 特徴ベクトルの条件付き確率

■ あるクラスに属する対象を観測したとき、その特徴ベクトルが観測される確率密度分布

$$p(\mathbf{x}|C_k) \qquad \int p(\mathbf{x}|C_k)d\mathbf{x} = 1$$

◆ これらの確率がわかれば、特徴ベクトルとクラスとの確率的な関係は全て計算できる。

身長に関する条件付密度分布



事後確率

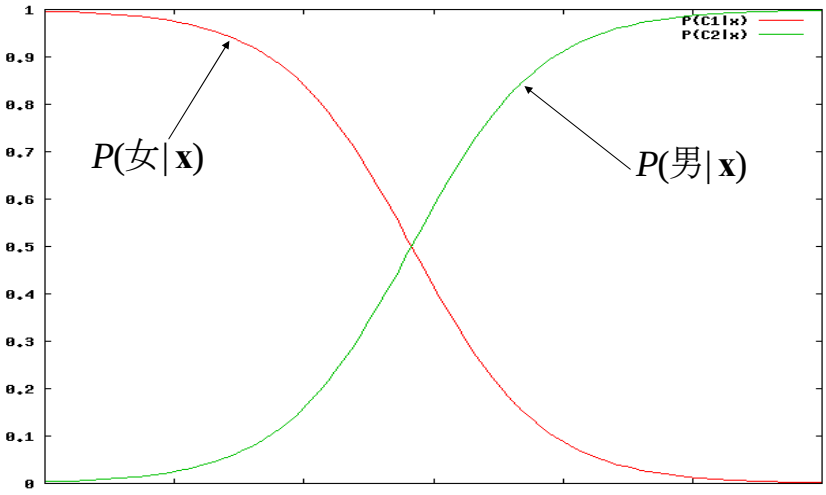
- 事後確率
  - ある対象から特徴ベクトルが観測されたとき、その対象がクラス C_k に属している確率

$$P(C_k|\mathbf{x}) = \frac{P(C_k)p(\mathbf{x}|C_k)}{p(\mathbf{x})} \qquad \sum_{k=1}^{K} P(C_k|\mathbf{x}) = 1$$

ここで、特徴ベクトルの確率密度分布は、

$$p(\mathbf{x}) = \sum_{k=1}^{K} P(C_k)p(\mathbf{x}|C_k) \qquad \int p(\mathbf{x}) = 1$$

身長に関する事後確率



期待損失

- 決定関数
  - 特徴ベクトルに基づき対象がどのクラスに属するかを決定する関数

$$d(\mathbf{x})$$

  - 損失関数
    - クラスC_kの対象をクラスC_jに決定したときの損失

$$r(C_j \mid C_k)$$

  - 期待損失（平均損失）

$$R[d] = \sum_{k=1}^{K} \int r(d(\mathbf{x}) \mid C_k) P(C_k \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{x}$$

これを最小とする決定関数を求めるのがベイズ決定理論

0－1損失の場合

- 0-1損失
  - 誤った識別に対して均等な損失を与える

$$r(C_j \mid C_k) = 1 - \delta_{jk}$$

  - 最適な識別関数（ベイズ識別方式）
    - 期待損失を最小とする最適な識別関数

$$d(\mathbf{x}) = C_k \quad \text{if} \quad P(C_k \mid \mathbf{x}) = \max_j P(C_j \mid \mathbf{x})$$

これは、事後確率が最大となるクラスに決定する識別方式

  - 最小誤識別率
    - ベイズ識別方式により達成される最小誤識別率

$$P_e^* = 1 - \int \max_j P(C_j \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{x}$$

2クラス（0－1損失）の場合

- 最適な識別方式
  - 事後確率の大小を比較すればよい

$$d(\mathbf{x}) = C_1 \quad \text{if} \quad P(C_1 \mid \mathbf{x}) \geq P(C_2 \mid \mathbf{x})$$
$$d(\mathbf{x}) = C_2 \quad \text{otherwise}$$

◆ 尤度比検定

$$d(\mathbf{x}) = C_1 \quad \text{if} \quad \frac{p(\mathbf{x} \mid C_1)}{p(\mathbf{x} \mid C_2)} \geq \theta$$
$$d(\mathbf{x}) = C_2 \quad \text{otherwise}$$

ここで、閾値は、$\theta = \frac{P(C_2)}{P(C_1)}$

正規分布の場合

- 確率密度分布

$$p(\mathbf{x} \mid C_k) = \frac{1}{(\sqrt{2\pi})^M \mid \Sigma_k \mid^{1/2}} \exp\left\{-\frac{1}{2}(\mathbf{x} - \mu_k)^T \Sigma_k^{-1} (\mathbf{x} - \mu_k)\right\}$$

◆ 2次の識別関数
  ◆ 事後確率の対数

$$g_k(\mathbf{x}) = \log P(C_k) - \frac{1}{2}(\mathbf{x} - \mu_k)^T \Sigma_k^{-1} (\mathbf{x} - \mu_k) - \frac{1}{2}\log \mid \Sigma_k \mid$$

◆ 線形識別関数
  ◆ 各クラスの共分散行列が等しい場合

$$g_k(\mathbf{x}) = \mu_k^T \Sigma^{-1} \mathbf{x} - \frac{1}{2}\mu_k^T \Sigma^{-1} \mu_k + \log P(C_k) = \mathbf{w}_k^T \mathbf{x} - h_k$$

等方的な正規分布の場合

- クラスが2つで、各クラスの共分散行列が等しい場合

$$\phi(\mathbf{x}) = g_1(\mathbf{x}) - g_2(\mathbf{x}) = (\mu_1 - \mu_2)^T \Sigma^{-1}\mathbf{x} - \frac{1}{2}(\mu_1^T \Sigma^{-1}\mu_1 - \mu_2^T \Sigma^{-1}\mu_2) + \log\frac{P(C_1)}{P(C_2)} = \mathbf{w}^T\mathbf{x} - h$$

- クラスが2つで、各クラスの共分散行列が等しく、等方的な場合

$$g_k(\mathbf{x}) = \log P(C_k) - \frac{\|\mathbf{x} - \mu_k\|^2}{2\sigma^2}$$

これは、先見確率が等しい場合には、特徴ベクトルと各クラスの平均ベクトルとの距離が最も近いクラスに決定する識別方式
つまり、各クラスの平均ベクトルをテンプレートと考えると、特徴ベクトルと各クラスのテンプレートとのマッチングによる識別

Fisherのアヤメのデータの識別課題

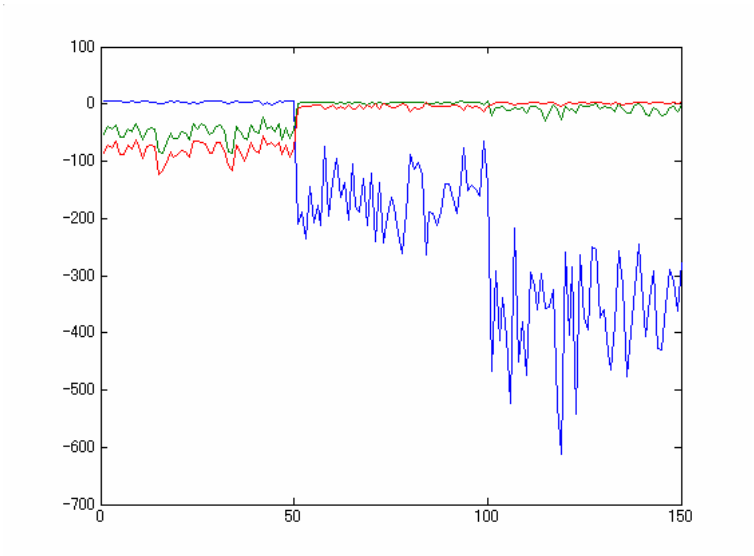
- 3種類のアヤメ
  - Setosa, Versicolor,Virginica
- 計測した特長
  - ガクの長さ、ガクの幅、花びらの長さ、花びらの幅
- 訓練用サンプル
  - 各アヤメそれぞれ50サンプルを収集
  - 合計150サンプル（50x3）
- 問題
  - ガクの長さ、ガクの幅、花びらの長さ、花びらの幅を計測して、どのアヤメかを推測する識別装置を設計すること

ベイズ決定則によるアヤメの識別

- データの表示
  - プログラム（testpca.m）
- ベイズ識別のための準備
  - 損失関数： 0－1識別の場合を考える
- 確率分布の推定
  - 各クラスの事前確率は、等確率（1／3）とする
  - 各アヤメから特徴ベクトルが得られる確率は正規分布と仮定
  - 正規分布のパラメータは、サンプル平均、サンプル分散共分散行列として推定
  - 識別関数の設計

$$g_k(\mathbf{x}) = \log P(C_k) - \frac{1}{2}(\mathbf{x}-\mu_k)^T \Sigma_k^{-1}(\mathbf{x}-\mu_k) - \frac{1}{2}\log|\Sigma_k|$$

アヤメのデータの識別結果



確率密度分布の推定

- ベイズ決定理論
  - 期待損失最小の意味で最適な識別方式

    しかし、、、
  - 各クラスと特徴ベクトルとの確率的な関係が完全にわかっていないと使えない！！！

  ＝＞ 訓練用のデータからデータの背後の確率的な関係を推定（確率密度分布の推定）

- 確率密度分布の推定法
  - パラメトリックモデルを用いる方法
    - 比較的少数のパラメータをもつモデル（パラメトリックモデル）を用いて確率分布を表現し、そのモデルをデータに当てはめ、データと尤も良く合うパラメータを推定
  - ノンパラメトリックモデルを用いる方法
    - 特定の関数型を仮定しないで、データに依存して分布の形を決める方法
  - セミパラメトリックな手法
    - 複雑な分布を表現するためにパラメータの数を系統的に増やせるようにすることで、パラメトリックモデルよりも一般的な関数型を表現できるようにする手法

パラメトリックモデル

- パラメトリックモデルによる確率密度分布の推定
  - モデル化
    - 確率密度分布をいくつかのパラメータを用いて表現
    - 正規分布：最も簡単で、最も広く用いられているパラメトリックモデル

$$p(\mathbf{x}\,|\,C_k)=\frac{1}{(\sqrt{2\pi})^M\,|\,\Sigma_k\,|^{1/2}}\exp\left\{-\frac{1}{2}(\mathbf{x}-\mu_k)^T\Sigma_k^{-1}(\mathbf{x}-\mu_k)\right\}$$

  - パラメータの推定法
    - 最尤推定法（maximum likelihood method）
      - パラメータを未知の固定値だとみなし、実際に観測された訓練データが得られる確率を最大化するようにパラメータを推定
    - ベイズ推定（Bayesian inference）
      - パラメータを既知の事前分布を持った確率変数だとみなし、パラメータの値の確信度をデータを観測した後の確率密度分布（事後確率密度分布）として表現

最尤推定

- パラメータを用いて表現された確率密度分布

$$p(\mathbf{x},\boldsymbol{\theta}) \qquad \boldsymbol{\theta}=(\theta_1,\ldots,\theta_P)^{\mathrm{T}}$$

- N個の独立なデータが与えられた時、そのデータがこの確率分布の独立なサンプルである尤もらしさ（尤度）

$$L(\boldsymbol{\theta})=\prod_{i=1}^{N} p(\mathbf{x}_i,\boldsymbol{\theta})$$

- 対数尤度（尤度の対数）

$$l(\boldsymbol{\theta})=\sum_{i=1}^{N}\log p(\mathbf{x}_i,\boldsymbol{\theta})$$

対数尤度を最大とするパラメータ（最尤解）に決定

最尤法（多変量正規分布の場合）

- 最尤解
  - 解析的に求めることが可能

$$\hat{\boldsymbol{\mu}}=\frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

$$\hat{\Sigma}=\frac{1}{\mathrm{N}}\sum_{i=1}^{N}(\mathbf{x}_i-\hat{\boldsymbol{\mu}})(\mathbf{x}_i-\hat{\boldsymbol{\mu}})^T$$

  - 平均ベクトルの最尤推定は、サンプル平均ベクトル
  - 分散共分散行列の最尤推定は、分散共分散行列のサンプル推定

ベイズ推定

- 最尤推定とベイズ推定
  - 最尤推定
    - パラメータを未知定数として、データから尤もらしいパラメータを推定
  - ベイズ推定
    - パラメータを仮に確率変数とみなして、パラメータの値の確信度を確率密度分布を用いて表現する。そして、データを観測する前にパラメータが取るであろう値の確率密度分布を事前確率として表現し、データが観測された後にパラメータが取るであろう値の確率密度分布（事後確率密度分布）を推定
    - データを観測する前： $p(\boldsymbol{\theta})$
      - データがどんな値を取るかに関する情報が無い ＝＞ 広がった分布
    - データを観測した後： $p(\boldsymbol{\theta}|X)$
      - データと整合性の良いパラメータほど大きな値を持つ ＝＞ 狭い分布

ベイズ学習：データを観測することによる確率分布の先鋭化

ベイズ推定（事後確率密度分布の計算）

- 学習データと同じ分布から特徴ベクトルxが得られる確率密度分布

$$p(\mathbf{x}|X)=\int p(\mathbf{x},\boldsymbol{\theta}|X)d\boldsymbol{\theta}=\int p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}|X)d\boldsymbol{\theta}$$

パラメトリックモデル

ただし、

$$p(\mathbf{x},\boldsymbol{\theta}|X)=p(\mathbf{x}|\boldsymbol{\theta},X)p(\boldsymbol{\theta}|X)=p(\mathbf{x}|\boldsymbol{\theta})p(\boldsymbol{\theta}|X)$$

つまり、パラメータの特定の値を決める代わりに、すべての可能な値を考えその重みつき平均により特徴ベクトルの確率密度分布を推定

- N個のデータが与えられた時のパラメータの事後確率密度分布

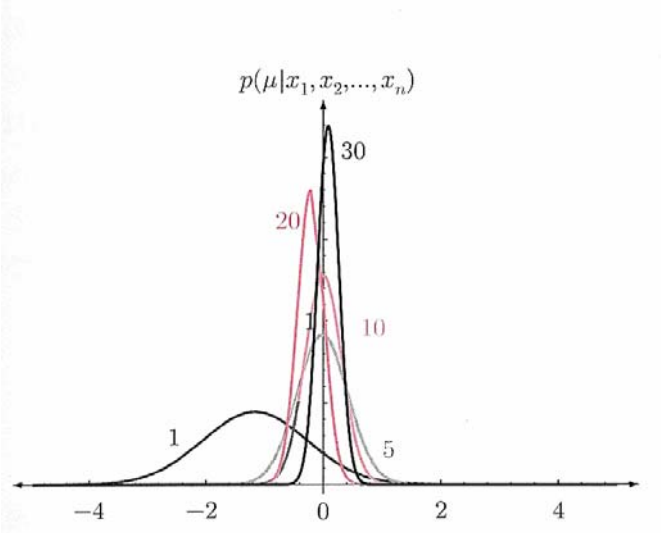
$$p(\boldsymbol{\theta}|X)=\frac{p(\boldsymbol{\theta})p(X|\boldsymbol{\theta})}{P(X)}=\frac{p(\boldsymbol{\theta})}{P(X)}\prod_{i=1}^{N}p(\mathbf{x}_i;\boldsymbol{\theta})$$

ただし、

$$p(X|\boldsymbol{\theta})=\prod_{i=1}^{N}p(\mathbf{x}_i;\boldsymbol{\theta}) \quad <= \text{ データの独立性より}$$

$$p(X)=\int p(\boldsymbol{\theta})\prod_{i=1}^{N}p(\mathbf{x}_i;\boldsymbol{\theta})d\boldsymbol{\theta}$$

ベイズ推定によるパラメータの推定



ノンパラメトリックな方法

- 特徴
  - 任意の密度関数の推定に適用できる
  - 密度関数の形が未知でも良い
    - ＝＞確率密度関数の形が訓練データに依存して決まる。
- 最も簡単なノンパラメトリックな手法の例
  - ヒストグラム
    ただし、推定された密度関数が滑らかではない
    高次元への拡張が難しい
- 代表的な方法
  - 核関数に基づく方法（kernel-based methods)
  - K-NN法（K-nearest-neighbors methods)

ノンパラメトリックな確率密度関数の推定法

- ベクトルxがある領域Rの内側に入る確率

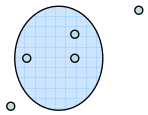
$$P = \int_R p(x')dx' \approx p(x)V$$

密度関数p(x)が連続で、領域R内でほとんど変化しない場合

- 独立なN個のサンプルが与えられた場合、N個のうちK個が領域Rに入る確率

$$\Pr(K) = \binom{N}{K} P^K (1-P)^{N-K}$$

  - Kの期待値は、E[K]=NP



- 確率密度関数は、

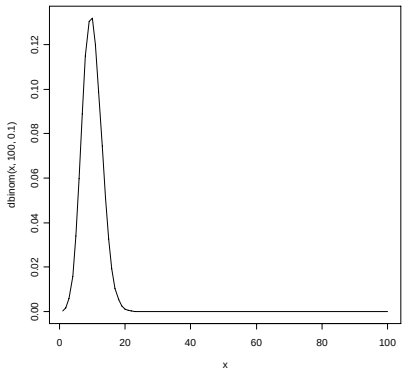
$$p(x) \approx \frac{K}{NV}$$

二項分布は平均付近で鋭いピークを持つので、比 K/N はPのよい近似

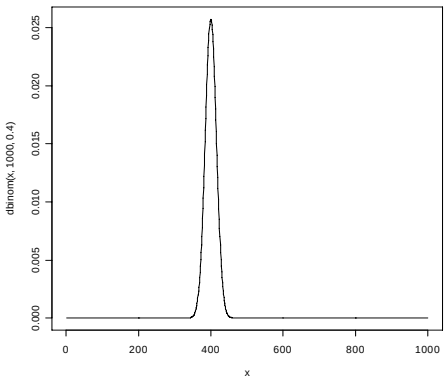
- 近似の成立の条件
  - 領域R内で確率密度関数があまり変化しないためには、領域は十分小さい
  - 二項分布がピークを持つためには、領域に入るサンプルはなるべく多くなければならず、領域はある程度大きい

二項分布とその期待値

$$\Pr(K) = \binom{N}{K} P^K (1-P)^{N-K}$$



N=100, P=0.1、E(K)=10



N=1000, P=0.4、E(K)=400

核関数に基づく方法

- 領域Rの体積Vを固定して、データからKを決定する
  - 点xを中心とする辺の長さがhの超立方体の体積： $V = h^M$
- 核関数
  - 原点を中心とする辺の長さが1の超立方体

$$\varphi(u) = \begin{cases} 1 & |u_j| < 1/2, \quad j = 1, \ldots, M \\ 0 & \text{otherwise} \end{cases}$$

  - 点uが点xを中心とする一辺hの超立方体の内部なら1： $\varphi\left(\frac{(x-u)}{h}\right)$
  - N個のデータのうち領域R内に入るデータの個数

$$K = \sum_{i=1}^{N} H(x_i) = \sum_{i=1}^{N} \varphi\left(\frac{(x-x_i)}{h}\right)$$

  - 確率密度分布

$$\hat{p}(x) \approx \frac{K}{NV} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{h^M} H\left(\frac{(x-x_i)}{h}\right)$$

核関数に基づく方法（多変量正規分布）

- 超立方体以外の核関数は？
  - 核関数の条件1

$$\varphi(\mathbf{u}) \geq 0$$

  - 核関数の条件1

$$\int \varphi(\mathbf{u}) d\mathbf{u} = 1$$

- 滑らかな核関数（多変量正規分布）を用いた場合

$$\hat{p}(x) \approx \frac{K}{NV} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{(2\pi h^2)^{M/2}} \exp\left(-\frac{\| x - x_i \|^2}{2h^2}\right)$$

滑らかさの制御

- 領域の大きさを変更することで、推定される密度関数の滑らかさが制御可能
  - 滑らかさを大きくしすぎる ＝＞ バイアスが大きくなる
  - 滑らかさが不十分 ＝＞ 個々の学習データに強く依存
  - 滑らかさのパラメータを適切に設定することが必要
- 滑らかさのパラメータの決定
  - 尤度：滑らかさの値が小さいほど尤度の値が大きくなる ＝＞ 使えない
  - Kullback-Leiblerの距離尺度

$$L = -\int p(x) \log \frac{\hat{p}(x)}{p(x)} dx$$

K-NN法

- Kを固定して、領域の大きさVを決定することで密度分布を推定
  - 点xを中心とする超球を考え、超球の半径をしだいに大きくして行き、その超球内に含まれるデータ点の数がちょうどK個になった時の超球の体積をV(x)とする

$$\hat{p}(x) \approx \frac{K}{NV(x)}$$

- 滑らかさの制御
  - データ点の個数Kを変更することで、推定される密度関数の滑らかさを制御可能
    - 滑らかさを大きくしすぎる ＝＞ バイアスが大きくなる
    - 滑らかさが不十分 ＝＞ ここの学習データに強く依存
    - 滑らかさのパラメータを適切に設定することが必要

K-NN（識別器の構成）

- K-NN法による条件付確率密度分布の推定
  - 学習データ
    - クラスCkからNk個の特徴ベクトルが得られているとする。全データ数は、N
    - 点xを中心とする超球を考え、その中にちょうどK個の学習データを含むまで超球の半径を大きくしていった時の超球の体積をV(x)とする。
    - 確率密度分布 $\hat{p}(x) \approx \frac{K}{NV(x)}$
    - その超球内、クラスCkのデータがKk個含まれているとすると、クラスCkの条件付確率密度分布

$$\hat{p}(x \mid C_k) \approx \frac{K_k}{N_k V(x)}$$

    - 事後確率

$$\hat{P}(C_k \mid x) = \frac{\hat{P}(C_k)\hat{p}(x \mid C_k)}{\hat{p}(x)} = \frac{\frac{N_k}{N}\frac{K_k}{N_k V(x)}}{\frac{K}{NV(x)}} = \frac{K_k}{K}$$

最近傍則（NN-則、Nearest Neighbor Rule)

- NN-則
  - 訓練サンプル集合の中で、xに最も近いサンプルを見つけ、そのサンプルのラベルのクラス（属していたクラス）に識別
  - 最近傍則の誤り率
    - 訓練サンプルが無数にあれば、達成可能な最小の誤り率（ベイズ誤り率）の2倍以下

$$P^* \le P \le 2P^* - \frac{K}{K-1}(P^*)^2$$

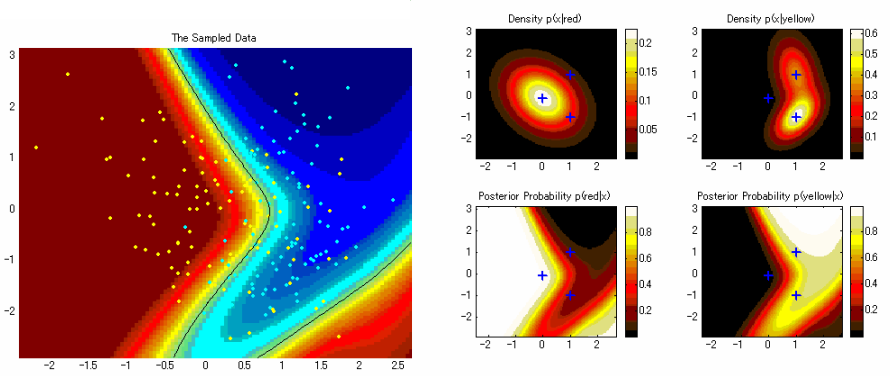
- K－NN則
  - 入力ベクトルx に近いK個のサンプルの中で、最も頻度の高いラベルのクラスに識別

＝＞ xに近いK個のサンプルを用いた多数決

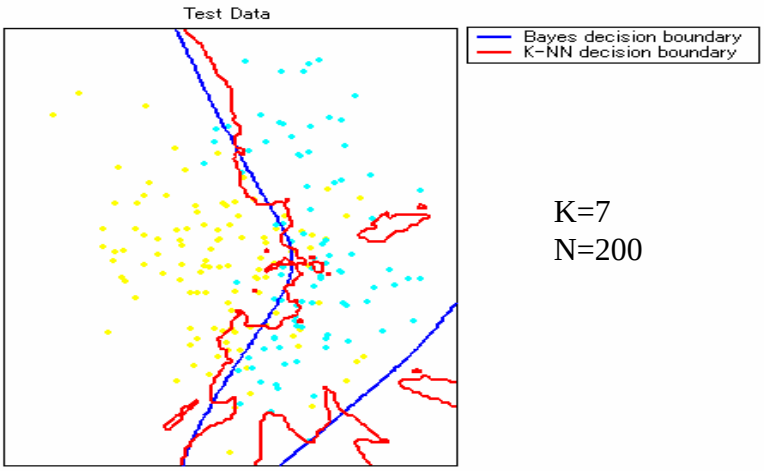
K-NN識別器によるパターン識別の例

- データ
  - Class 1: 2次元正規分布
  - Class 2: 2つの正規分布の混合分布

N1=100
N2=100



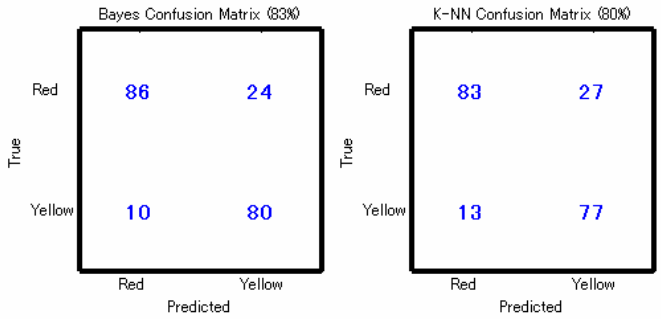
K-NN識別器による識別境界



K=7
N=200

K-NN識別器によるテストサンプルの識別結果

新たに生成したテストサンプル（N=200)の識別



セミパラメトリックな手法

- パラメトリックモデルに基づく方法とノンパラメトリックな方法の中間的手法
  - パラメトリックモデルに基づく方法
    - 利点： 新しいデータに対する確率密度の計算が比較的簡単
    - 欠点： 真の分布と仮定したモデルが異なる場合には必ずしも良い推定結果が得られない
  - ノンパラメトリックな手法
    - 利点： 真の分布がどんな関数系であっても推定できる
    - 欠点： 新しいデータに対して確率密度を評価するための計算量が学習用のデータが増えるとどんどん増加してしまう
  - 両方の良い点を取り入れ、欠点を改善するような手法

- 代表例
  - 混合分布モデル(Mixture models)に基づく方法
  - ニューラルネットワーク

混合分布モデル

- 混合分布

$$p(x) = \sum_{j=1}^{O} \omega_j p(x \mid j)$$

- 混合パラメータの条件

$$\sum_{j=1}^{O} \omega_j = 1, \qquad 0 \le \omega_j \le 1$$

- 各確率密度分布の条件

$$\int p(x \mid j) dx = 1$$

- 各確率密度分布が正規分布の場合（混合正規分布モデル）

$$p(x \mid j) = \frac{1}{(2\pi\sigma_j^2)^{d/2}} \exp\left\{ -\frac{\| x - \mu_j \|^2}{2\sigma_j^2} \right\}$$

混合正規分布の最尤推定

- N個の学習データに対する対数尤度

$$l = \log L = \log \prod_{n=1}^{N} p(x_n) = \sum_{n=1}^{N} \log p(x_n) = \sum_{n=1}^{N} \log \left\{ \sum_{j=1}^{O} \omega_j p(x_n \mid j) \right\}$$

- 各確率密度分布のパラメータ推定（正規分布の場合）
  - 非線形最適化手法を利用

$$\frac{\partial l}{\partial \mu_j} = \sum_{n=1}^{N} \frac{\omega_j p(x_n \mid j)}{p(x_n)} \frac{(x_n - \mu_j)}{\sigma_j^2} = \sum_{n=1}^{N} P(j \mid x_n) \frac{(x_n - \mu_j)}{\sigma_j^2}$$

$$\frac{\partial l}{\partial \sigma_j} = \sum_{n=1}^{N} \frac{\omega_j p(x_n \mid j)}{p(x_n)} \left\{ -\frac{d}{\sigma_j} + \frac{\| x_n - \mu_j \|^2}{\sigma_j^3} \right\} = \sum_{n=1}^{N} P(j \mid x_n) \left\{ -\frac{d}{\sigma_j} + \frac{\| x_n - \mu_j \|^2}{\sigma_j^3} \right\}$$

ただし、

$$P(j \mid x) = \frac{\omega_j p(x \mid j)}{\sum_{k=1}^{O} \omega_k p(x \mid k)}$$

混合正規分布の最尤推定（つづき）

- 混合パラメータの推定
  - 補助パラメータを利用（softmax関数）

$$\omega_j = \frac{\exp(\gamma_j)}{\sum_{k=1}^{O} \exp(\gamma_k)}$$

  - 対数尤度の補助パラメータに関する微分

$$\frac{\partial l}{\partial \gamma_j} = \sum_{k=1}^{O} \frac{\partial l}{\partial \omega_j} \frac{\partial \omega_j}{\partial \gamma_j} = \sum_{n=1}^{N} \{P(j \mid x_n) - \omega_j\}$$

混合正規分布の最尤推定（つづき）

- 最尤解の性質
  - 対数尤度の微分＝0とおくと

$$\hat{\omega}_j = \frac{1}{N} \sum_{n=1}^{N} P(j \mid x_n)$$

$$\hat{\mu}_j = \frac{\sum_{n=1}^{N} P(j \mid x_n) x_n}{\sum_{n=1}^{N} P(j \mid x_n)}$$

$$\hat{\sigma}_j^2 = \frac{1}{d} \frac{\sum_{n=1}^{N} P(j \mid x_n) \| x_n - \hat{\mu}_j \|^2}{\sum_{n=1}^{N} P(j \mid x_n)}$$

$$P(j \mid x) = \frac{\omega_j p(x \mid j)}{\sum_{k=1}^{O} \omega_k p(x \mid k)}$$

  - 各要素への帰属度を表す事後確率P(j|x)を重みとして計算される

EMアルゴリズム

- EMアルゴリズム
  - 不完全データからの学習アルゴリズム
    - 混合分布モデルのパラメータの推定に利用可能
    - 最急降下法と同様に解を逐次改良して、次第に最適な解に近づける
    - 一般的な定式化は、Dempster等による(1977)
- EMアルゴリズムの実際
  - 各確率密度分布が正規分布の場合

$$p(x \mid j) = \frac{1}{(2\pi\sigma_j^2)^{d/2}} \exp\left\{-\frac{\| x - \mu_j \|^2}{2\sigma_j^2}\right\}$$

  - 方針
    - データxがどの正規分布から生成されたかの番号zを含めたもの(x,z)を完全データとみなし、xを不完全データとみなしてEMアルゴリズムを適用

EMアルゴリズム（つづき）

- 完全データの分布

$$f(x, z) = \omega_z \, p(x \mid z)$$

- N個の完全データに対する対数尤度

$$\hat{l} = \sum_{n=1}^{N} \log f(x_n, z_n) = \sum_{n=1}^{N} \log \left\{\omega_{z_n} \, p(x_n \mid z_n)\right\}$$

- EMアルゴリズム
  - パラメータの適当な初期値からはじめて、EステップとMステップと呼ばれる二つの手続きを繰り返す

EMアルゴリズム（メタアルゴリズム）

- Eステップ
  - 完全データの対数尤度のデータとパラメータに関する条件付き期待値の計算

$$Q(\theta \mid \theta^{(t)}) = E[\, f(x_n, z_n) \mid x, \theta^{(t)})]$$

- Mステップ
  - Qを最大とするパラメータを求めて新しい推定値とする

EステップとMステップを繰り返して得られるパラメータは、尤度を単調に増加させることが知られている

EMアルゴリズム（具体例）

- 正規分布の混合分布の場合
  - Qを最大とするパラメータは陽に求まる

$$\hat{\omega}_j^{(t+1)} = \frac{1}{N}\sum_{n=1}^{N} P(j \mid x_n, \theta^{(t)})$$

$$\hat{\mu}_j^{(t+1)} = \frac{\sum_{n=1}^{N} P(j \mid x_n, \theta^{(t)}) x_n}{\sum_{n=1}^{N} P(j \mid x_n, \theta^{(t)})}$$

$$\hat{\sigma}_j^{2(t+1)} = \frac{1}{d}\frac{\sum_{n=1}^{N} P(j \mid x_n, \theta^{(t)}) \| x_n - \hat{\mu}_j^{(t)} \|^2}{\sum_{n=1}^{N} P(j \mid x_n, \theta^{(t)})}$$

$$P(j \mid x) = \frac{\omega_j p(x \mid j)}{\sum_{k=1}^{O} \omega_k p(x \mid k)}$$

  - 各要素への帰属度を表す事後確率の現時点での推定値を重みとして、パラメータを推定することを繰り返す

EMアルゴリズム（利点と欠点）

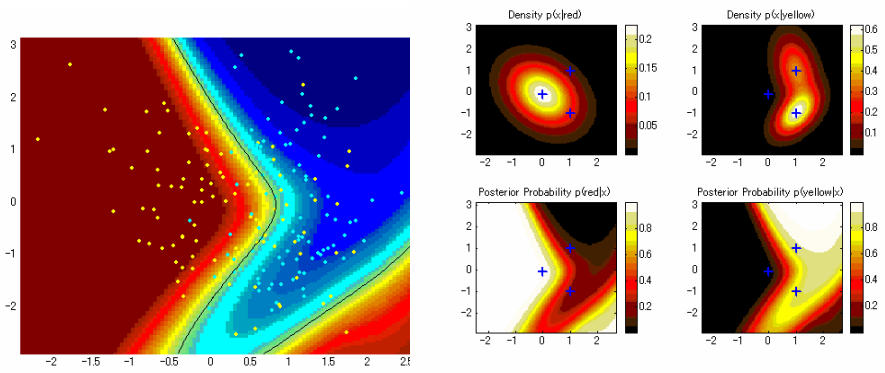
- 利点
  - 各繰り返しのステップで尤度が単調に増加
    - 他の方法（最急降下法等）と比べて数値計算的に安定
  - 逆行列の計算が必要ない
    - Newton法等の非線形最適化手法に比べて簡単
  - 多くの実例では他の手法に比べて良い解に収束する
  - 繰り返しの初期の段階ではNewton法と同程度に速い

- 欠点
  - 解の近くでは収束が遅くなるので、工夫が必要
  - 大域的な収束は保証されていないので、初期値の選び方の工夫が必要

混合正規分布モデルを用いた識別の例

- データ
  - Class 1: 2次元正規分布
  - Class 2: 2つの正規分布の混合分布

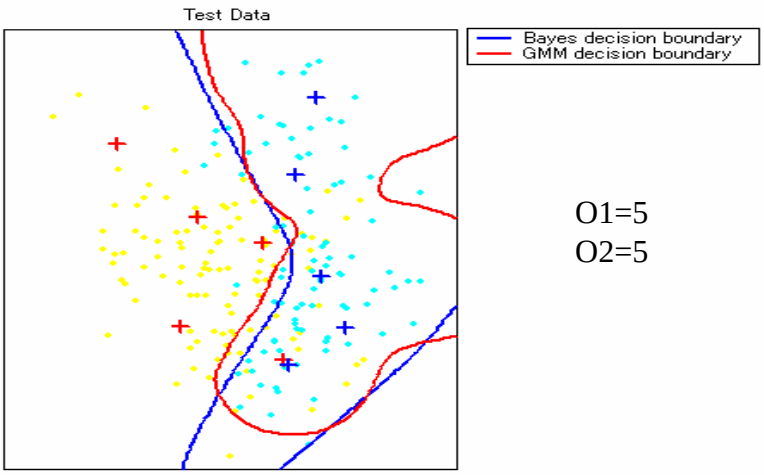
N1=100
N2=100



識別器の構成と学習

- 各クラスの分布を正規混合分布により推定
  - Class 1: O=5個の正規混合分布
  - Class 2: O=5個の正規混合分布
- 訓練サンプル
  - N=200サンプル（各クラス100サンプル）
- パラメータの学習法
  - EMアルゴリズムを利用

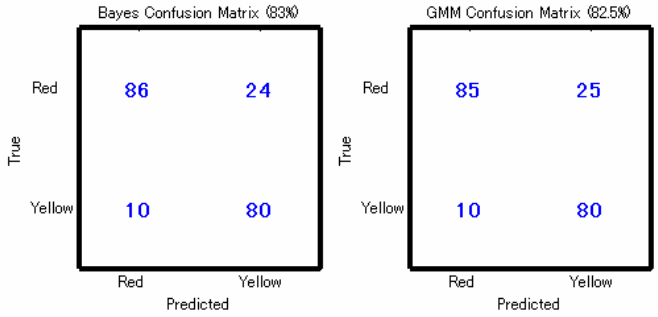
混合正規分布推定による識別境界



O1=5
O2=5

混合正規分布推定によるテストサンプルの識別結果

新たに生成したテストサンプル（N=200)の識別

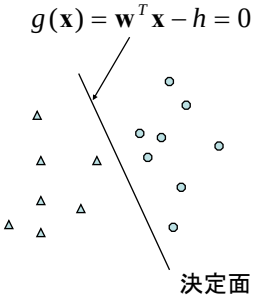


線形識別関数の学習

線形判別関数

- 線形判別関数
  - 特徴ベクトルからクラスの識別に有効な特徴を取り出す関数
  - 重みベクトルとバイアス(しきい値重み)をパラメータとするモデル

$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} - w_0$$



線形判別関数の性質(その1)

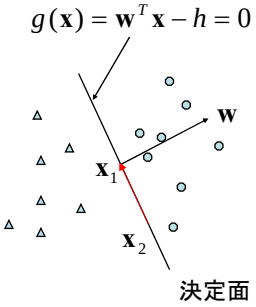
- 重みは決定面上の任意のベクトルと直交する

決定面上の2点を考える

$$g(\mathbf{x}_1) = \mathbf{w}^T\mathbf{x}_1 - w_0 = 0$$
$$g(\mathbf{x}_2) = \mathbf{w}^T\mathbf{x}_2 - w_0 = 0$$

これらの差を取ると

$$\mathbf{w}^T(\mathbf{x}_1 - \mathbf{x}_2) = 0$$



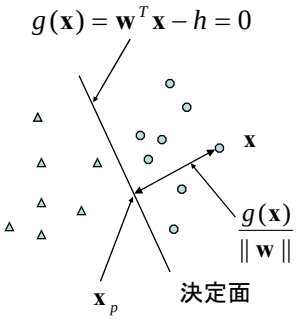
線形判別関数の性質（その2）

- 線形判別関数の値g(x)は決定面からの距離と密接に関係する

任意の点xと決定面との距離

$$
\begin{aligned}
g(\mathbf{x}) &= g(\mathbf{x}_p + r\frac{\mathbf{w}}{\|\mathbf{w}\|}) \\
&= \mathbf{w}^T(\mathbf{x}_p + r\frac{\mathbf{w}}{\|\mathbf{w}\|}) - w_0 \\
&= \mathbf{w}^T\mathbf{x}_p - w_0 + r\frac{\mathbf{w}^T\mathbf{w}}{\|\mathbf{w}\|} \\
&= g(\mathbf{x}_p) + r\frac{\|\mathbf{w}\|^2}{\|\mathbf{w}\|} \\
&= r\|\mathbf{w}\|
\end{aligned}
$$

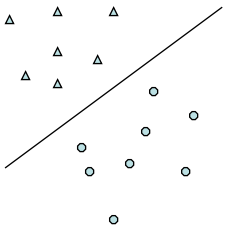
$$r = \frac{g(\mathbf{x})}{\|\mathbf{w}\|}$$



線形分離可能

- 2つのクラスC1およびC2からのN個のサンプルがあるとき、線形判別関数を用いて、N個のサンプルをすべて正しく識別できるようなパラメータが存在する

線形分離可能



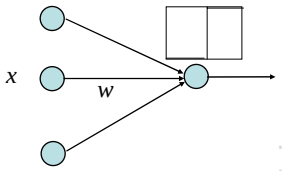
$$g(\mathbf{x}) = \mathbf{w}^T\mathbf{x} - h$$

ニューロン（神経細胞）

- 脳
  - 多数のニューロン（神経細胞）から構成される情報処理装置
    - 大脳には数百億個のニューロンが存在
    - 小脳には千億個のニューロンが存在
- ニューロン（神経細胞）
  - 電気信号を発して、情報をやり取りする特殊な細胞
    - 軸索：長い
    - 樹状突起：木の枝のように複雑に分岐したもの
- シナプス
  - 軸索の末端
  - 電気信号を化学物質の信号に変えて、次の神経細胞に情報を伝達

ニューロンのモデル

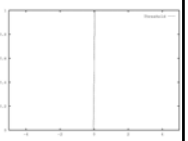
- Maculloch＆Pittsのモデル



$$y = f(\eta),$$

$$\eta = \sum_{j=1}^{M} w_j x_j - h = \mathbf{w}^T \mathbf{x} - h$$

$$f(\eta) = \begin{cases} 1 & \text{if } \eta \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

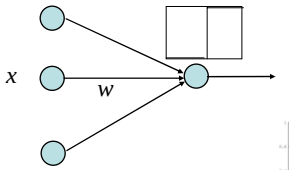


  - Y=1 は、ニューロンが興奮・発火している状態
  - Y=0 は、ニューロンが興奮していない状態
  - 他からの入力が重みつきで加算され、それがしきい値を超えたら発火する

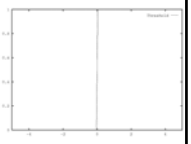
単純パーセプトロンの学習

- 計算

$$y = f(\eta),$$
$$\eta = \sum_{j=1}^{M} w_j x_j - h = \mathbf{w}^T \mathbf{x} - h$$



$$f(\eta) = \begin{cases} 1 & \text{if } \eta \geq 0 \\ 0 & \text{otherwise} \end{cases}$$



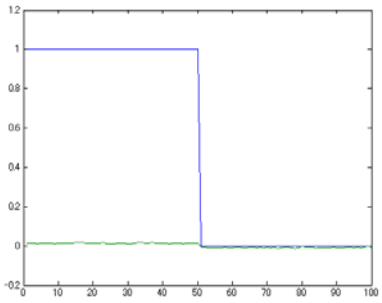
- 学習（誤り訂正学習）
  - ネットワークにパターンを分類させてみて間違っていたら結合を修正
  - 訓練サンプル
  - 学習則

$$\{< \mathbf{x}_i, t_i > | \, i = 1, \ldots, N\} \quad t_i \in \{1,0\}$$

$$w_j \Leftarrow w_j + \alpha(t_i - y_i)x_{ij}$$
$$h \Leftarrow h - \alpha(t_i - y_i)$$

単純パーセプトロンによるアヤメのデータの識別

- 問題
  - 2種類のアヤメを識別
- 手法
  - 単純パーセプトロン
- プログラム
  - （perceptron.m）



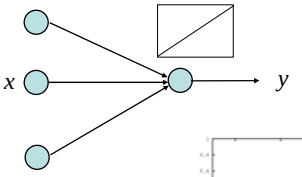
単純パーセプトロンの問題点

- 収束性の問題
  - 線形分離可能でない場合には、学習が収束しないことがある。
- 解の一意性の問題
  - 線形分離可能な場合には、たすうの可能な解が存在するが、どの解が得られるかわからない（初期値に依存する）
- 学習速度の問題
  - 収束までに必要なパラメータの更新回数が非常に多くなる場合がある。
  - クラスとクラスとの間のギャップ（間隔）が狭いと、より多くの更新が必要

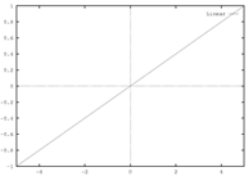
最小2乗線形判別関数

- モデル

$$y = \sum_{j=1}^{M} w_j x_j - h$$



教師信号 t



- 訓練サンプル

$$\{< \mathbf{x}_i, t_i > | \, i = 1, \ldots, N\} \quad t_i \in \{1, 0\}$$

- 評価関数（2乗誤差最小）

$$\varepsilon_{emp}^2 = \sum_{i=1}^{N} \| t_i - y_i \|^2$$

最小2乗線形判別関数の学習

- 逐次学習（最急降下法）
  - 偏微分

$$\frac{\partial \varepsilon_{emp}^2}{\partial w_j} = \sum_{i=1}^{N} -2(t_i - y_i)x_{ij}$$

$$\frac{\partial \varepsilon_{emp}^2}{\partial h} = \sum_{i=1}^{N} -2(t_i - y_i)(-1)$$

  - Widrow-Hoffの学習則（デルタルール）

$$w_j \Leftarrow w_j + \alpha \sum_{i=1}^{N} (t_i - y_i)x_{ij}$$

$$h \Leftarrow h + \alpha \sum_{i=1}^{N} (t_i - y_i)(-1)$$

確率的最急降下法 (Stochastic Gradient Descent)

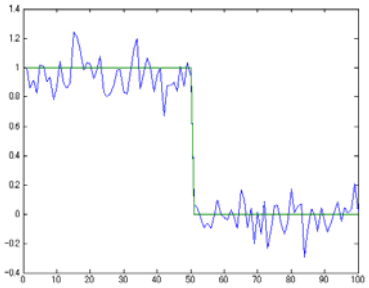
- 各訓練サンプル毎にパラメータを更新

$$w_j \Leftarrow w_j + \alpha(t_i - y_i)x_{ij}$$

$$h \Leftarrow h + \alpha(t_i - y_i)(-1)$$

Adalinによるアヤメのデータの識別

- 問題
  - 2種類のアヤメを識別
- 手法
  - 最小2乗線形判別関数の学習
- プログラム
  - （adalin.m）



最小2乗線形判別関数の学習（最適解）

- 解析解（重回帰分析）
  - 2乗誤差の行列表現

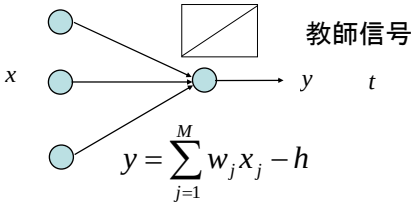
$$\varepsilon_{emp}^2 = \sum_{i=1}^{N} \| t_i - y_i \|^2 = \| \mathbf{t} - \mathbf{X}\tilde{\mathbf{w}} \|^2$$

  - 偏微分

$$\frac{\partial \varepsilon_{emp}^2}{\partial \tilde{\mathbf{w}}} = X^T(\mathbf{t} - X\tilde{\mathbf{w}}) = 0$$

  - 最適解

$$\tilde{\mathbf{w}}^* = (X^T X)^{-1} X^T \mathbf{t}$$



訓練サンプル

$$X = (\tilde{\mathbf{x}}_1, \ldots, \tilde{\mathbf{x}}_N)$$
$$\mathbf{t} = (t_1, \ldots, t_N)$$

最尤推定としての定式化

- モデル
  - 教師信号とネットワークの出力との誤差が平均0、分散σの正規分布に従うと仮定
- 誤差の尤度

$$L=\prod_{i=1}^{N} p(\varepsilon_i;\sigma^2,\mathbf{w},h)=\prod_{i=1}^{N}\frac{1}{\sqrt{2\pi\sigma^2}}\exp\{-\frac{\varepsilon_i^2}{2\sigma^2}\}$$

- 対数尤度

$$l=\sum_{i=1}^{N}\{-\frac{1}{2}\log 2\pi\sigma^2-\frac{\varepsilon_i^2}{2\sigma^2}\}=-\frac{N}{2}\log 2\pi\sigma^2-\frac{1}{2\sigma^2}\sum_{i=1}^{N}\varepsilon_i^2$$

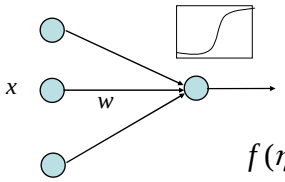
これを最大とすることは、2乗誤差の和を最大とすることとおなじ

ロジスティック回帰

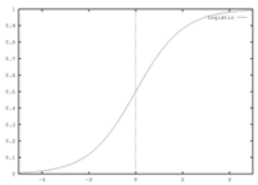
- 計算

$$y=f(\eta),$$

$$\eta=\sum_{j=1}^{M} w_j x_j-h=\mathbf{w}^T\mathbf{x}-h$$



$$f(\eta)=\frac{\exp(\eta)}{1+\exp(\eta)}$$



- 尤度

$$L=\prod_{i=1}^{N} y_i^{u_i}(1-y_i)^{(1-u_i)}$$

- 対数尤度

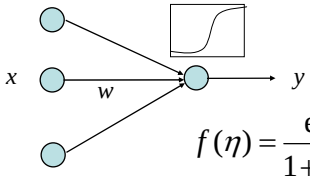
$$l=\sum_{i=1}^{N}\{u_i\log y_i+(1-u_i)\log(1-y_i)\}=\sum_{i=1}^{N}\{u_i\eta_i-\log(1+\exp(\eta_i))\}$$

ロジスティック回帰の学習

- 偏微分

$$\frac{\partial l}{\partial w_j} = \sum_{i=1}^{N} (u_i - y_i) x_{ij}$$

$$\frac{\partial l}{\partial h} = \sum_{i=1}^{N} (u_i - y_i)(-1)$$



$$f(\eta) = \frac{\exp(\eta)}{1+\exp(\eta)}$$

$$y = \sum_{j=1}^{M} w_j x_j - h$$

- パラメータ更新式（学習則）

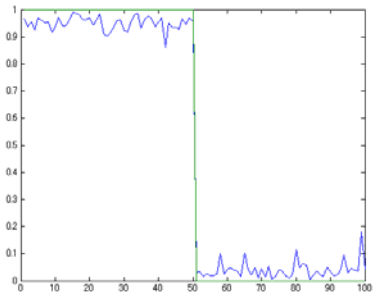
$$w_j \Leftarrow w_j + \alpha \sum_{i=1}^{N} (u_i - y_i) x_{ij}$$

$$h \Leftarrow h + \alpha \sum_{i=1}^{N} (u_i - y_i)(-1)$$

•Widrow-Hoffの学習則と全く同じ形。
•出力の計算法が異なるので、結果は同じはない。

ロジスティック回帰によるアヤメのデータの識別

- 問題
  - 2種類のアヤメを識別
- 手法
  - ロジスティック回帰
- プログラム
  - （logit.m）



もう少し凝った学習法

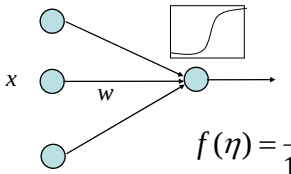
- Fisherのスコアリングアルゴリズム
  - Fisher情報行列を利用したニュートン法

$$F = X^T W X$$

  - 重み付最小2乗法の繰り返し

$$\mathbf{w} \Leftarrow (X^T W X)^{-1} X^T W (\eta + W^{-1}\delta)$$

$$W = diag(\omega_1, \ldots, \omega_N)$$
$$\omega_i = y_i(1 - y_i)$$
$$\delta = (\delta_1, \ldots, \delta_N)$$
$$\delta_i = u_i - y_i$$
$$\eta = (\eta_1, \ldots, \eta_N)$$



教師信号 u

$$f(\eta) = \frac{\exp(\eta)}{1 + \exp(\eta)}$$

$$y = \sum_{j=1}^{M} w_j x_j - h$$

多クラスの場合

- 単純パーセプトロン
  - 複数のパーセプトロンを使う？
- 最小2乗線形判別
  - 多クラスへの拡張は容易
  - 多クラスの場合のパラメータ

$$W = (X^T X)^{-1} X^T T$$

- ロジスティック回帰
  - 多クラスへの拡張は容易

多層パーセプトロン

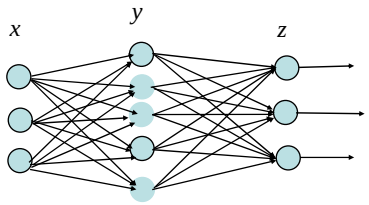
- モデル

$$\varsigma_j = \sum_{i=1}^{H} a_{ij} x_i - a_{0j}$$

$$y_j = f_{hidden}(\varsigma_j)$$

$$\eta_k = \sum_{j=1}^{J} b_{jk} y_j - b_{0k}$$

$$z_k = f_{out}(\eta_k)$$



- 入出力関数
  - 中間層:ロジスティック関数
  - 出力層:関数近似の場合は線形関数、パターン認識課題では、ロジスティック関数やsoftmax関数

多層パーセプトロンの能力[G.Cynenko,1989]

- 中間層のユニットの入出力関数が

$$\sigma(t) = \begin{cases} 1 \text{ as } t \to +\infty \\ 0 \text{ as t} \to \text{-}\infty \end{cases}$$

のような性質をもつ非線形の連続な単調増加関数であり、出力層の入出力関数が線形関数のとき、中間層が1層の多層パーセプトロンによって任意の連続関数が近似可能

ただし、任意の連続関数を近似するためには、中間層のユニット数は非常に多くする必要があるかもしれない。

誤差逆伝播学習法(Back-propagation)

- 中間層の入出力関数がロジスティック関数で、出力層のユニットの入出力関数が線形の中間層が1層のみのネットワークの場合

- 評価関数

$$\varepsilon_{emp}^2 = \sum_{p=1}^{N} \| \mathbf{t}_p - \mathbf{z}_p \|^2$$

- 学習則

$$a_{ij} \Leftarrow a_{ij} + \alpha \sum_{p=1}^{N} \gamma_{pj} \nu_{pj} x_{pi}$$

$$b_{jk} \Leftarrow b_{jk} + \alpha \sum_{i=1}^{N} \delta_{pk} y_{pj}$$

$$\nu_{pj} = y_{pj}(1 - y_{pj})$$

$$\gamma_{pj} = \sum_{k=1}^{K} \delta_{pk} b_{jk}$$

$$\delta_{pk} = t_{pk} - z_{pk}$$

その他の話題

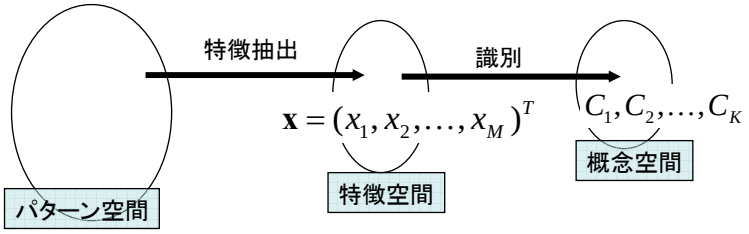
- 最尤推定としての定式化
  - 最小2乗判別関数の学習の場合と同様
  - 教師信号とネットワークの出力との誤差を、互いに独立な平均0、分散σの等方的な正規分布と仮定

- より複雑なアルゴリズム
  - 2次微分も利用　＜＝　ロジスティック回帰と同様に、IRLS法的な方法も導出可能

統計的特徴抽出

パターン認識過程

- 特徴抽出
  - 認識対象から何らかの特徴量を計測(抽出)する必要がある
  - 認識に有効な情報(特徴)を抽出し、次元を縮小した効率の良い空間を構成する過程
    - 文字認識： スキャナ等で取り込んだ画像から文字の識別に必要な本質的な特徴のみを抽出(例、文字線の傾き、曲率、面積など)
- 識別
  - 与えられた未知の対象を、特徴とクラスの関係に関する知識に基づいて、どのクラスに属するかを決定(判定)する過程



識別に有効な特徴の抽出

- 特徴空間
  - パターンを計測して得られる特徴は、必ずしも識別に有効とは限らない。

    ＝＞ 識別に有効な特徴を取り出すには？

- 有効な特徴を抽出する方法

  方法1： 統計的特徴抽出法
  - 重回帰分析
  - 主成分分析
  - 判別分析

  方法2： 特徴選択法

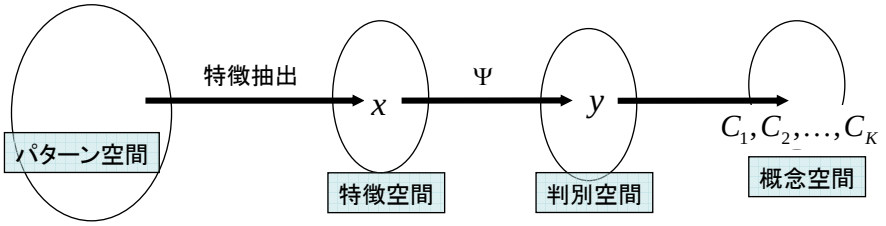
統計的特徴抽出

- パターンの変形
  - 実際のパターンは不規則な変形を伴っている、また、観測にノイズが混入することもある

    ＝＞ 特徴空間内の理想的な点の回りの確率的な散らばり（分布）となる
- 統計的特徴抽出
  - 特徴空間で特徴ベクトルの確率統計的な構造を利用して、パターンを識別するのに有効な特徴を抽出する過程

$$y = \Psi(x)$$

特徴空間から認識に有効なより低次元の判別特徴空間への最適な写像は、yでの良さを表す評価基準と特徴空間でのパターンの確率統計的構造に依存して決まる



線形多変量データ解析手法

- 線形特徴抽出

$$y = \Psi(x) = A^T x - b$$

- 多変量データ解析手法
  - 線形判別分析、線形重回帰分析、主成分分析など
  - 多変量を線形結合した新変量に関する評価基準として、平均2乗誤差最小、分散最大などの2次の統計量に基づく評価基準を考える
    - 特徴空間（データの空間）の確率統計的構造が、2次までの統計量（平均ベクトル、相関行列、共分散行列など）に要約され、線形代数の範囲で最適解が陽に求まる

線形回帰による直線の当てはめ

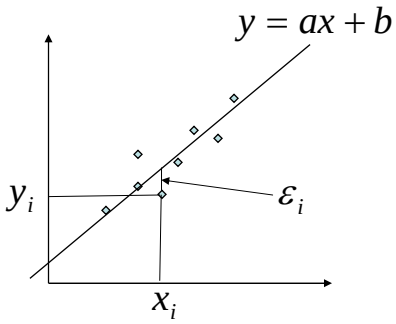
- N個のデータ

$$(x_1, y_1), \ldots, (x_N, y_N)$$

- モデル

$$y = ax + b$$

- 評価基準
  - 平均2乗誤差最小



$$\varepsilon^2 = \frac{1}{N}\sum_{i=1}^{N} \varepsilon_i^{\,2} = \frac{1}{N}\sum_{i=1}^{N} (y_i - ax_i - b)^2$$

最適解（直線の当てはめ）

- 最適なパラメータ

$$a^* = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i-\bar{x})(y_i-\bar{y})}{\frac{1}{N}\sum_{i=1}^{N}(x_i-\bar{x})^2} = \frac{r_{xy}}{\sigma_x^2}$$

$$b^* = \bar{y} - \frac{r_{xy}}{\sigma_x^2}\bar{x}$$

- 最適な直線

$$y = \frac{r_{xy}}{\sigma_x^2}(x-\bar{x}) + \bar{y}$$

達成される平均2乗誤差

- この時、達成される平均2乗誤差

$$\varepsilon^2 = \frac{1}{N}\sum_{i=1}^{N}\left\{(y_i-\bar{y}) - \frac{r_{xy}}{\sigma_x^2}(x_i-\bar{x})\right\}^2$$

$$= \sigma_y^2\left\{(1-\left(\frac{r_{xy}}{\sigma_x^2}\right)^2\right\}$$

$$= \sigma_y^2(1-\rho^2)$$

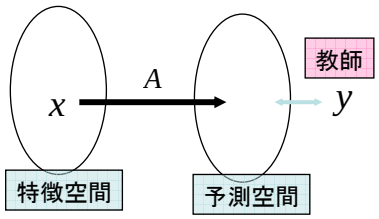
線形重回帰分析

- 訓練データ

$$(\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_N, \mathbf{y}_N)$$

- 線形写像

$$y = \Psi(x) = A^T x$$



- 平均2乗誤差基準
  - 入力と望みの出力の対が学習データとして与えられている時、線形モデルの出力と望みの出力との平均2乗誤差が最小となるような係数行列を求める

$$\varepsilon^2(A) = \frac{1}{N}\sum_{i=1}^{N} \| \mathbf{y}_i - A^T \mathbf{x}_i \|^2$$

線形重回帰分析の最適解

- 最適解

$$A = R_{XX}^{-1} R_{XY}$$

$$R_{XX} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i \mathbf{x}_i^T$$

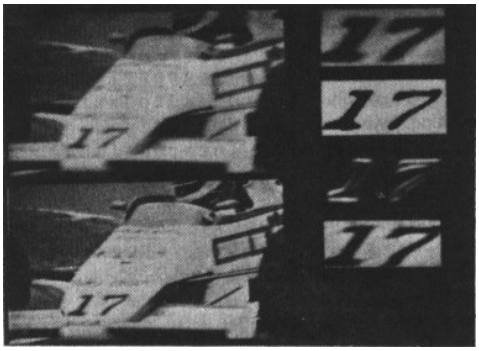
$$R_{XY} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{x}_i \mathbf{y}_i^T$$

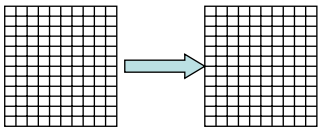
- 達成される平均2乗誤差

$$\varepsilon^2(A) = \frac{1}{N}\sum_{i=1}^{N} \| \mathbf{y}_i - A^T \mathbf{x}_i \|^2$$
$$= tr(R_{YY}) - tr(R_{XY}^T R_{XX}^{-1} R_{XY})$$

重回帰分析による画像修復

- モデル推定としての画像処理
  - 画像の修復、鮮鋭化、平滑化、エッジ抽出など
  - 与えられた画像から望みの画像を出力するような写像を推定





最小2乗線形判別写像

- 理想出力を各クラスの代表ベクトルとする
  - 平均2乗誤差

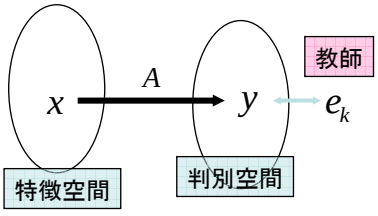
$$\varepsilon^2(A) = \frac{1}{N}\sum_{i=1}^{N} \| t_i - A^T x_i \|^2 = \sum_{k=1}^{K} \omega_k \frac{1}{N_k} \sum_{x_j \in C_k}^{N} \| e_k - A^T x_j \|^2$$

  - 最適な係数行列

$$A = R_{XX}^{-1} \sum_{k=1}^{K} \omega_k \mu_k e_k^T$$

  - 最適写像（最小2乗線形判別写像）

$$y = \sum_{k=1}^{K} \omega_k (\mu_k^T R_{XX}^{-1} x) e_k$$



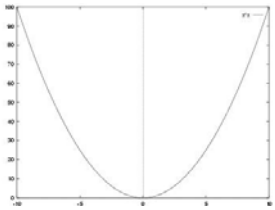
ロバスト統計手法

- 最小2乗法
  - データからモデルを推定するための基本的な道具
    - モデルとデータとの誤差が平均0の正規分布なら、推定されたモデルは最適
    - データに例外値が含まれているような場合には、得られた結果は信頼できない
- ロバスト統計
  - 例外値をある程度含むようなデータからでも比較的安定にモデルのパラメータを推定可能
    - 代表例：メディアンフィルタ（画像の平滑化、ノイズ除去）
    - データに例外値が含まれていることを前提にしてデータからモデルを推定出来ると便利
      - 複数の動きを含んだデータから主な動きを推定
      - 不連続を含むデータの滑らかさの評価

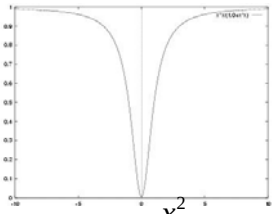
M-Estimator

- 最小2乗(LMS)基準 $LMS = \min \sum_i r_i^2$
  - すべての誤差を均等な重みで扱う
  - 大きな例外値により大きな影響を受ける
- M-Estimatorの基準 $M = \min \sum_i \rho(r_i)$
  - ρ (x)は、x=0で最小値を持つ対称な正定値関数
  - ρ (x)=x*x なら、最小2乗誤差基準と同じ（最小2乗法の拡張）
  - Influence function
    - 関数 ρ (x)により、モデルからずれたデータに対してどれくらいの重みが与えられるかの評価（ρ (x)のxに関する偏微分）
    - Geman & McClure の ρ では、データがモデルからある程度離れるとその影響はほとんどなくなる
  - 推定アルゴリズム
    - M基準を最小化する最適化問題 ＝＞ 重み付き最小2乗法
    - 初期値の選びかたに依存

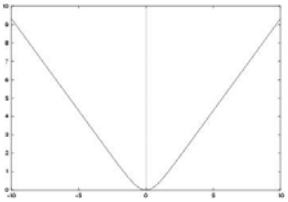
Influence Function



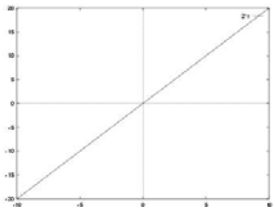
$$\rho(x) = x^2$$



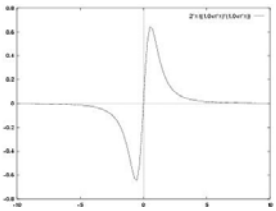
$$\rho(x) = \frac{x^2}{\sigma + x^2}$$



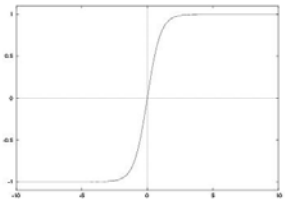
$$\rho(x) = \log(\cosh(x))$$



$$\Psi(x) = 2x$$



$$\Psi(x) = \frac{2x\sigma}{\sigma + x^2}$$



$$\Psi(x) = \tanh(x)$$

LMedS推定

- LMedS(Least Median of Squares)基準

$$LMedS = \min \text{ med } r_i^2$$

  - breakdown point（例外値に対するロバストネスの評価指標）
    - 例外値がない場合の結果と例外値を含む場合の結果が非常に大きくずれることなく、何割までのデータを非常に大きな例外値に置き換えることができるか
    - 最小2乗誤差基準は、0 ＝＞ ひとつでも大きくずれる
    - LMedS 基準は、0.5 ＝＞ 50%までの例外値でも頑健
  - 推定アルゴリズム
    - 多次元の場合には、最適解を見つけるのは難しい ＝＞ ランダムサンプリングによる方法
      - 1. 全データから p 個のデータをランダムに選ぶ
      - 2. p 個のデータを用いてモデルのパラメータを推定
      - 3. LMedS基準により、そのパラメータのモデルを評価
    - アルゴリズムの繰り返し回数
      - m回のランダムサンプリングで少なくとも1個のサンプルには例外値が含まれ無い確率

$$P = 1 - (1 - (1 - \varepsilon)^p)^m$$

例外値の検出

- モデルとデータとの誤差の標準偏差のロバストな推定

$$\hat{\sigma} = C\{1+\frac{5}{N-F}\}\,\mathrm{med}\,\sqrt{\varepsilon_{\mathrm{i}}^2}$$

- 例外値の検出
  - 例えば、誤差の標準偏差の2．5倍よりも大きな誤差を持つデータを例外値と判定

ロバストテンプレートマッチング

- テンプレートマッチング
  - 最も簡単で基本的なパターン認識手法
  - 文字認識、対象の追跡、ステレオなどに応用
- マッチング対象
  - テンプレートや画像中には、マッチングさせたい対象部分とそれ以外の部分とが含まれている
  - マッチングさせたい対象の部分（テンプレート中の大きな面積をしめる対象の部分）のみを自動的にマッチングさせるには？
- ロバストテンプレートマッチング
  - ロバスト統計の手法を用いて、マッチングさせたくない部分を自動的に除外（例外値検出）
  - 残りの部分のマッチングを行う
- 応用例
  - 顔画像のマッチング
  - ビデオ映像のカット変わりの検出

相関係数

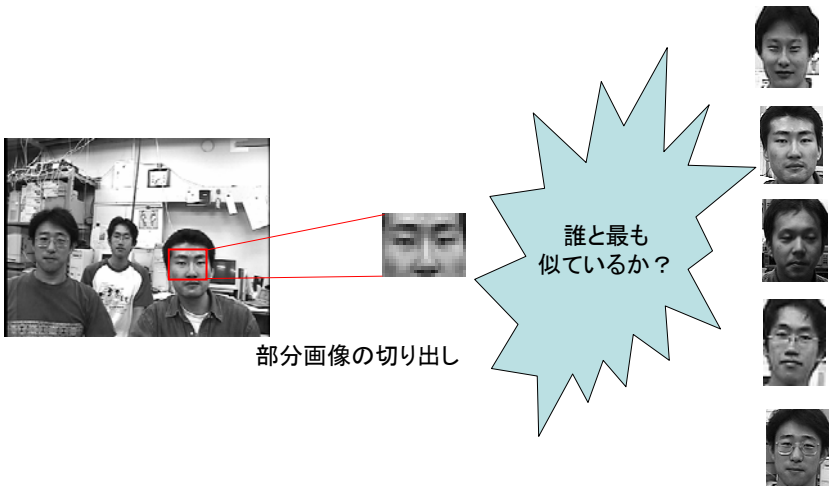
- 相関係数
  - テンプレート画像とマッチングさせたい画像との類似度

$$\rho = \frac{\frac{1}{N}\sum_{i=1}^{M} \mathbf{x}_i^T \mathbf{y}_i}{\sigma_x \sigma_y}$$

  - テンプレート画像の画素数 M
  - $\mathbf{x}_i$ テンプレート画像の画素の値（例えば、色の赤、緑、青成分）
  - $\mathbf{y}_i$ マッチングさせたい画像の画素の値

ロバスト相関を用いた部分顔画像のマッチング

- 部分的な顔画像と証明写真との照合



ロバスト相関

- テンプレートと画像のマッチング
  - テンプレート画像とマッチングさせたい画像で、マッチングさせたい部分の画素の値はほぼ等しい
  - マッチングさせたい部分の面積はそれ以外の部分の面積よりも大きい
- ロバスト相関
  - 標準偏差の推定値を計算

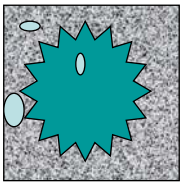
$$\bar{\sigma} = 1.4826(1+\frac{5}{M-1})\sqrt{\mathrm{med}\,|\mathbf{x}_i - \mathbf{y}_i|^2}$$

  - 例外値の検出

$$\sqrt{|\mathbf{x}_i - \mathbf{y}_i|^2} \geq 2.5\bar{\sigma}$$

  - 例外値を除いたデータに対して相関係数を計算

テンプレート



入力画像

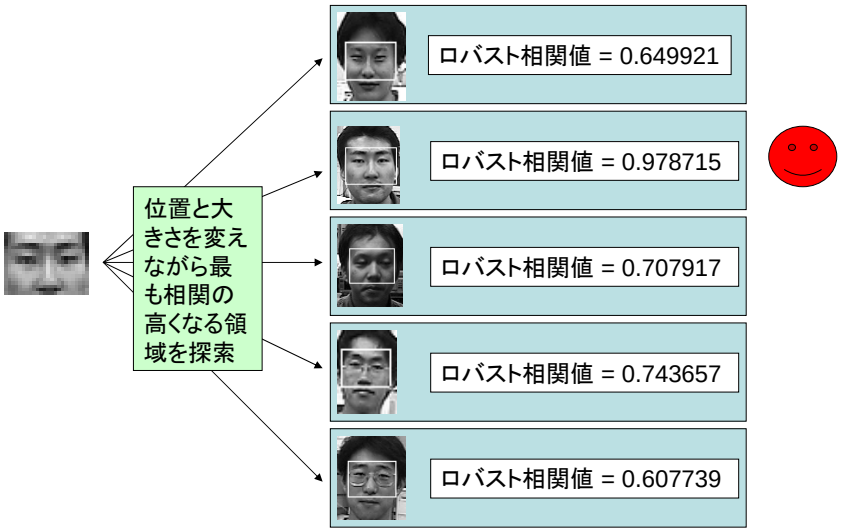
顔画像のマッチング

テンプレート

入力画像

検出された例外値（白）

ロバストテンプレートマッチングの結果



カット変わりの検出への応用

- カット変わりの検出
  - 映像データベースの構築の自動化のための基本的な手段
  - 連続する2枚の画像間のロバストテンプレートマッチングにより得られた相関係数の大きさによりカット変わりを検出
- 実験
  - 映画のビデオ映像から連続する2枚の画像を200組
    - 画像のサイズ:53x40、相関計算には色の3成分を利用

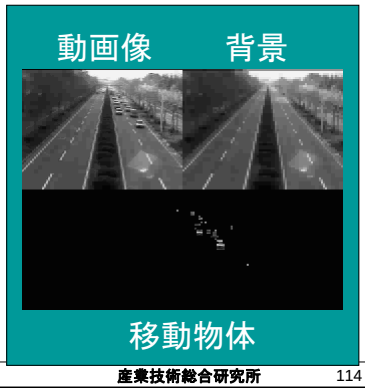
	平均	標準偏差
連続する画像	0.996367	0.009374
連続しない画像	0.388177	0.260962

    - 相関係数は、連続する画像に対して安定に大きな値
  - しきい値＝平均－2.5x標準偏差＝0.972932 より大きな値を持つ画像対を連続する画像と判定
    - 誤り確率(連続＝＞不連続 3%、不連続＝＞連続0%)

適応的背景推定による移動物体の検出

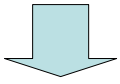
①移動物体のある動画像から、背景を獲得

②入力画像と背景画像との差分により、移動物体が検出



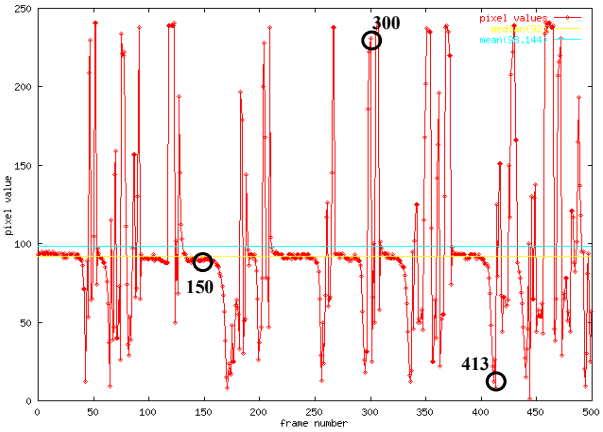
背景画像の獲得

- 従来法
  - 移動物体が含まれていない画像を背景画像として撮影
  - ノイズに強くするためには、移動物体の含まれない画像を複数枚撮影してその平均を背景画像とする
- しかし、実際の応用場面では、
  - 照明条件が変化したり、カメラが動くなどして背景が変化
  - 道路や人通りの多い場所などのように移動物体を含まない画像を取ることが難しい場合も多い



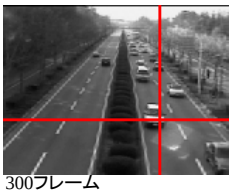
- 移動物体を含んだ動画像から適応的に背景モデルを推定

動画像中での輝度値の変化



画素110×80における輝度値の推移


150フレーム


300フレーム


413フレーム

適応的逐次M-推定

適応的逐次M−推定 （M−推定＋忘却）

- 適応的に推定（指数的に忘却）
- 逐次的に推定（最急降下法）

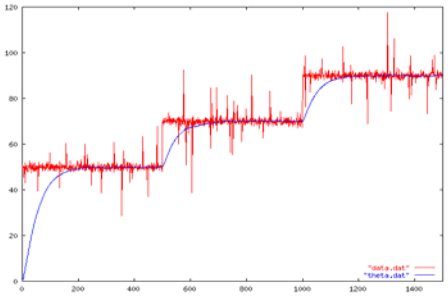
$$\rho(x) = \log(\cosh(x))$$

$$E_t = \sum_l \alpha^{t-l} \rho(\varepsilon_{t-l})$$

$0 \le a \le 1$: 忘却率

$$\theta_t = \theta_{t-1} - \eta \frac{\partial E_t}{\partial \theta}$$

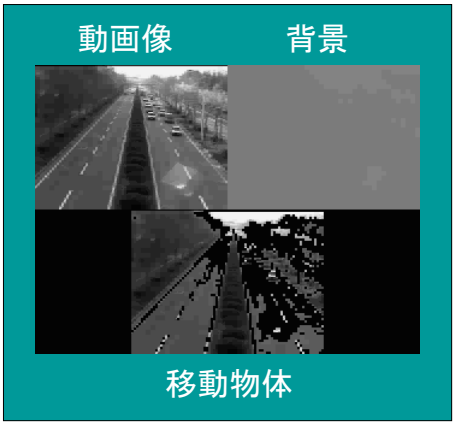
$\eta \ge 0$: 学習係数



逐次M－推定の結果例

背景画像の推定例

- 移動物体を含む動画像からの、背景画像の獲得
- 移動物体の検出



動画像：160×120画素
背景画像：80×60画素

グレースケール256階調

オンラインで
30[フレーム／秒] 達成

$$\rho(x) = \log(\cosh(\frac{x}{50}))$$
（Logistic関数）

$a = 0.8$

$\eta = 0.05$

推定速度（学習係数）の自動設定

- カメラが移動 ― 背景を高速に推定したい
- 移動物体 ― 背景として取り込みたくない

入力画像と背景画像の類似度が、
- 低い ― 学習を速く（η を大きく）
- 高い ― 学習を遅く（η を小さく）

⇨ しかし、普通の相関では、
移動物体が現れると、類似度が低下

移動物体の領域を無視して、入力画像と背景画像の類似度を計算できる手法が必要

⇨ ロバストテンプレートマッチング

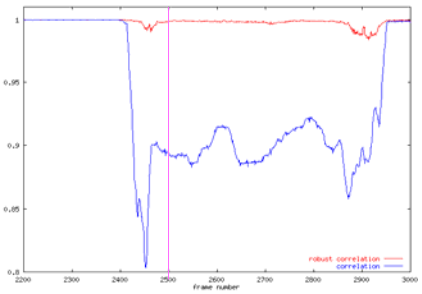

背景画像


入力画像1


入力画像2

ロバストテンプレートマッチング

移動物体の領域を例外値として、残りの部分で相関係数を求める（ロバスト相関）



相関係数とロバスト相関係数の時間変化

➡ 移動物体に影響を受けていない

背景画像

入力画像

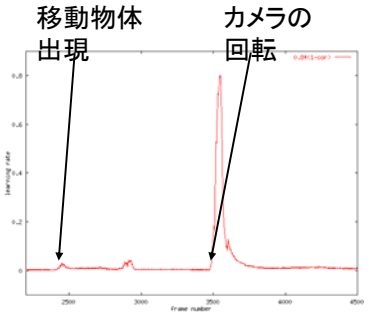
例外値

（2500フレーム）

学習速度の自動設定の例

学習係数

$$\eta = 0.8 \times (1.0 - r_{\mathrm{rob}})$$



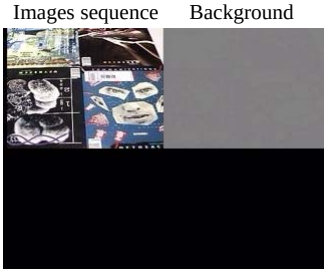
学習係数の時間変化

➡ シーンに応じて学習速度が変化している。

学習速度の適応的な調節

学習速度固定

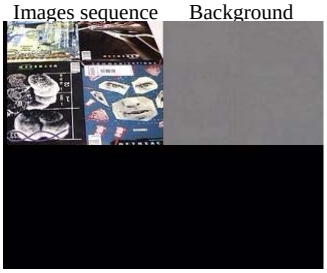
Images sequence Background



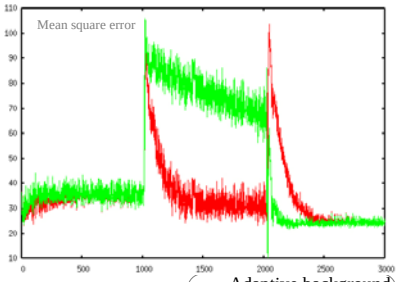
Moving Objects

適応的学習速度

Images sequence Background



Moving Objects

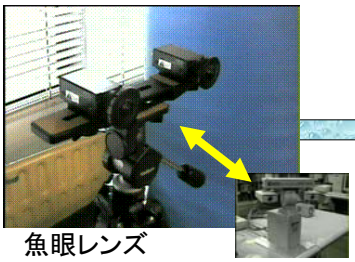


Learning rate

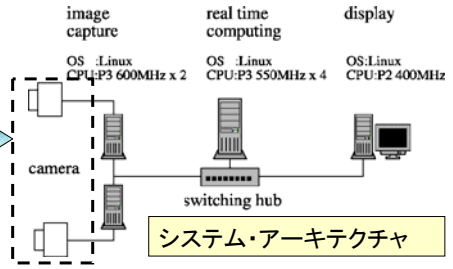
1

0

魚眼レンズを用いたバーチャルアクティブカメラ

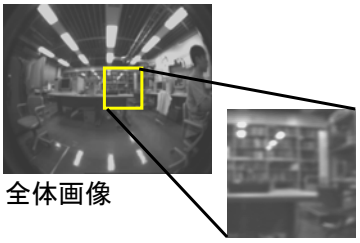


魚眼レンズ



システム・アーキテクチャ

全体画像

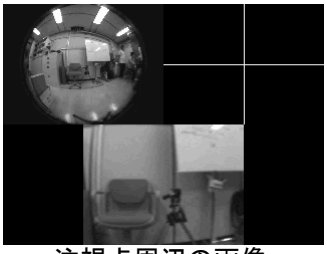


全体画像

注視点周辺の画像

移動物体を注視

全体画像 移動物体と重心

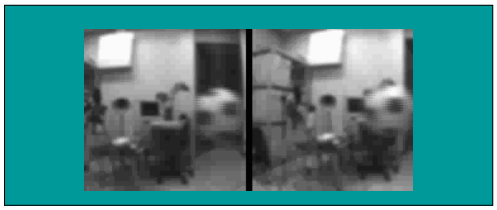


注視点周辺の画像

移動物体の追跡

両眼によるボールの追従運動

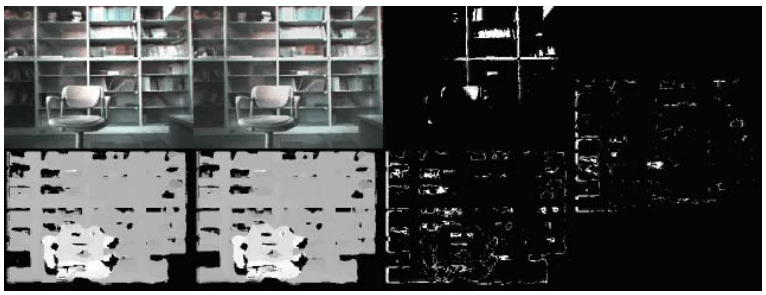
サッケード



⇨ 生体模倣型ステレオアクティブビジョンの基礎システム

ステレオカメラで撮影した動画像からの背景推定

カラー画像　背景画像　移動物体



距離画像　背景画像　移動物体

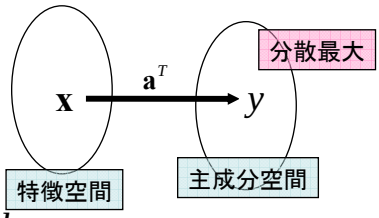
主成分分析

- 訓練データ

$$X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$$

- 与えられたデータの変動を最もよく表す新たな特徴量を求める

$$y_i = \sum_{j=1}^{M} a_j x_{ij} + b = \mathbf{a}^T \mathbf{x}_i + b$$



- 新特徴の統計量

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i = \frac{1}{N}\sum_{i=1}^{N} (\mathbf{a}^T \mathbf{x}_i + b) = \mathbf{a}^T \bar{\mathbf{x}} + b$$

$$\sigma_y^2 = \frac{1}{N}\sum_{i=1}^{N} (y_i - \bar{y})^2 = \mathbf{a}^T \left[\frac{1}{N}\sum_{i=1}^{N} (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T \right] \mathbf{a} = \mathbf{a}^T \Sigma_X \mathbf{a}$$

主成分分析（導出1）

- 評価基準
  - 新特徴の分散最大

$$\sigma_y^2 = \mathbf{a}^T \Sigma_X \mathbf{a}$$

- 制約条件

$$\sum_{j=1}^{M} a_j^2 = \mathbf{a}^T \mathbf{a} = 1$$

- 最適化問題（Lagrange乗数）

$$Q(\mathbf{a}) = \sigma_y^2 - \lambda(\mathbf{a}^T \mathbf{a} - 1) = \mathbf{a}^T \Sigma_X \mathbf{a} - \lambda(\mathbf{a}^T \mathbf{a} - 1)$$

主成分分析（導出2）

- Qのパラメータに関する偏微分

$$\frac{\partial Q(\mathbf{a})}{\partial \mathbf{a}} = 2\Sigma_X \mathbf{a} - 2\lambda \mathbf{a} = 0$$

- これから、Xの分散共分散行列の固有値問題が得られる

$$\boxed{\Sigma_X \mathbf{a} = \lambda \mathbf{a}}$$

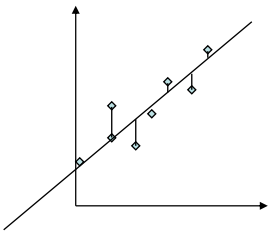
- 最適なパラメータは、Xの分散共分散行列の最大固有値として求まる。ただし、その大きさについては、制約条件を満たす必要がある。

$$\sum_{j=1}^{M} a_j^2 = \mathbf{a}^T \mathbf{a} = 1$$

直線の当てはめ

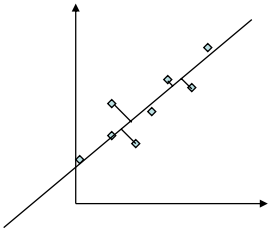
- 重回帰分析

$$\varepsilon^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - ax_i - b)^2$$



- 主成分分析

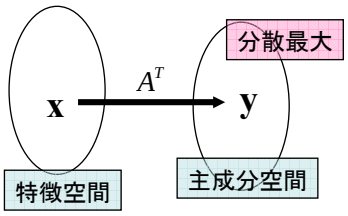
$$\varepsilon^2 = \frac{1}{N}\sum_{i=1}^{N} d_i^2(\mathbf{a}, \mathbf{r}_0)$$



主成分分析（多次元の場合）

- 主成分分析(Principal Component Analysis)
  - 多変量の計測値から変量間の相関を無くし、しかも、より低次元の変量によって元の計測値の特性を記述

$$\mathbf{y}=A^T(\mathbf{x}-\bar{\mathbf{x}})=A^T\tilde{\mathbf{x}}$$



  - 最適な係数行列

$$\Sigma_X A=A\Lambda,\quad (A^TA=I)$$

  - 最小二乗近似

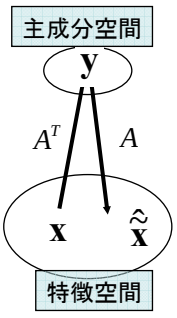
$$\varepsilon^2(A)=\frac{1}{N}\sum_{i=1}^{N}\|\tilde{\mathbf{x}}_i-\hat{\tilde{\mathbf{x}}}_i\|^2,\quad (\hat{\tilde{\mathbf{x}}}_i=AA^T\tilde{\mathbf{x}}_i)$$

主成分分析と最小2乗近似

- 最小2乗近似

$$\hat{\tilde{\mathbf{x}}}_i=A\mathbf{y}_i=AA^T(\mathbf{x}_i-\bar{\mathbf{x}})=AA^T\tilde{\mathbf{x}}_i$$

$$\varepsilon^2(A)=\frac{1}{N}\sum_{i=1}^{N}\|\tilde{\mathbf{x}}_i-\hat{\tilde{\mathbf{x}}}_i\|^2$$



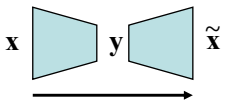
固有顔による顔画像の認識

- 主成分分析(Principal Component Analysis)
  - ◆ 多変量の計測値から変量間の相関を無くし、しかも、より低次元の変量によって元の計測値の特性を記述

$$\mathbf{y}=A^T(\mathbf{x}-\bar{\mathbf{x}})=A^T\tilde{\mathbf{x}} \qquad \Sigma_X A=A\Lambda, \quad (A^TA=I)$$

  - – 最小二乗近似

$$\varepsilon^2(A)=\frac{1}{N}\sum_{i=1}^{N}|\tilde{\mathbf{x}}_i-\hat{\tilde{\mathbf{x}}}_i|^2, \quad (\hat{\tilde{\mathbf{x}}}_i=AA^T\tilde{\mathbf{x}}_i)$$



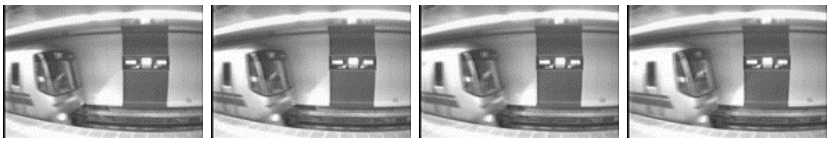
- 固有顔(Eigen Face)
  - – 各画像を画素の値をならべたベクトルとして表現し、画像集合を主成分分析して得られる固有ベクトル
  - – 主成分スコア間の距離

$$|\mathbf{y}_1-\mathbf{y}_2|^2=|A^T(\tilde{\mathbf{x}}_1-\tilde{\mathbf{x}}_2)|^2=|\hat{\tilde{\mathbf{x}}}_1-\hat{\tilde{\mathbf{x}}}_2|^2$$

カメラの回転に伴うフロー成分の推定

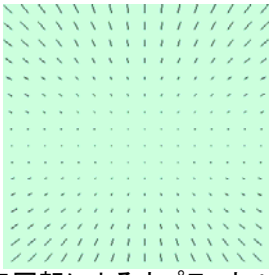
■ 視覚障害者:
列車の進入・停止の判断が困難

■ ヘッドマウントカメラを使用し、
得られる画像を処理する
→ 列車の動きを検出する



カメラから得られる画像

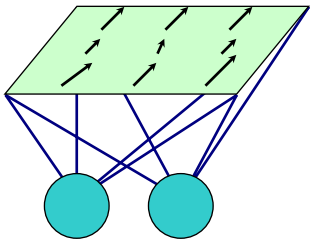
カメラ回転によって生じるオプティカルフロー



カメラ回転によるオプティカルフロー

$$\xi = \left[\mathbf{v}_1^T, \mathbf{v}_2^T, \ldots\right]^T$$

線形写像

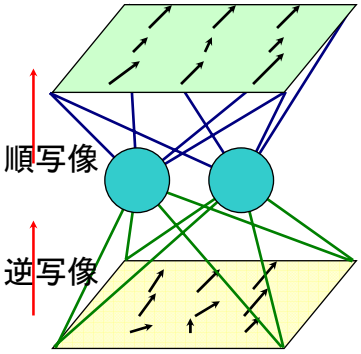


$$\Omega = \begin{bmatrix}\Omega_x & \Omega_y\end{bmatrix}^T$$

$$\mathbf{V}(x,y) = \begin{bmatrix}u(x,y)\\ v(x,y)\end{bmatrix} = \begin{bmatrix}\frac{xy}{f} & -\left(f+\frac{x^2}{f}\right)\\ f+\frac{y^2}{f} & -\frac{xy}{f}\end{bmatrix}\begin{bmatrix}\Omega x\\ \Omega y\end{bmatrix}$$

恒等写像学習を用い順逆モデルの同時推定

$$\tilde{\xi} = \left[\tilde{\mathbf{v}}_1^T, \tilde{\mathbf{v}}_2^T, \ldots\right]^T$$



$$\xi = \left[\mathbf{v}_1^T, \mathbf{v}_2^T, \ldots\right]^T$$

■ 恒等写像学習
→ 次元圧縮してノイズを除去

◆ 単純な自乗誤差に基づく学習
(主成分分析と等価)
→ 例外値に弱い

- ロバストな学習
  （重み付き自乗誤差を最小化）
- 例外値の除去

確からしさを重みとする評価

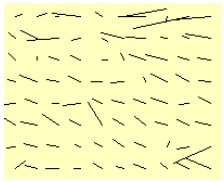
重み付き自乗誤差

$$E_W = C\sum_{t}^{T}\sum_{p}^{P} c_{tp}\left\|\mathbf{v}_{tp} - \tilde{\mathbf{v}}_{tp}\right\|^2$$ ⇒ 最小化

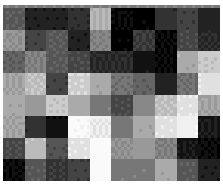
$$\left(C = \frac{1}{P}\sum_{t,p} c_{tp}\right)$$ （c_{tp}: フローごとの確からしさ）

→ [Simoncelli91]の手法を使用

元画像



得られたフロー



フローに対する確からしさ

確からしさを考慮することによってノイズによる影響を抑える

例外値の除去

■ 例外値の判定

しきい値: $\theta = 2.5\sigma$

$$\sigma = 1.4826\left(1 + \frac{5}{TP-1}\right)\sqrt{\mathrm{med}\,\varepsilon_{tp}}$$

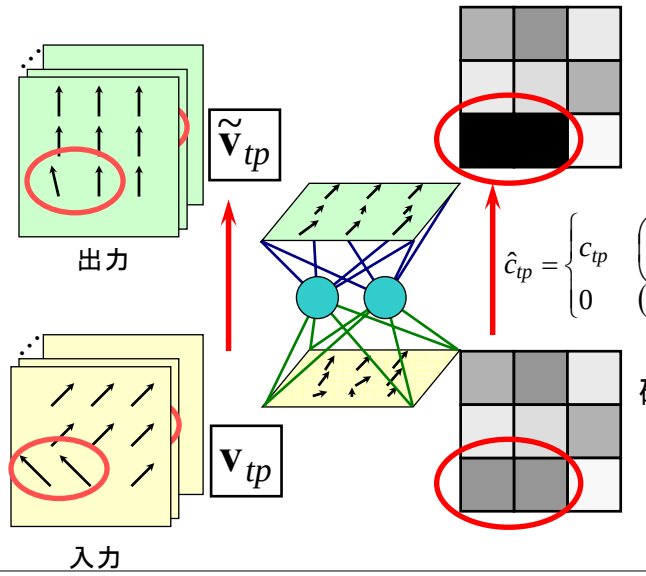
($\mathrm{med}\,\varepsilon_{tp}$: $\{\varepsilon_{tp}\}$のメディアン)

$$\varepsilon_{tp} = \sqrt{c_{tp}\left\|\mathbf{v}_{tp} - \tilde{\mathbf{v}}_{tp}\right\|^2}$$

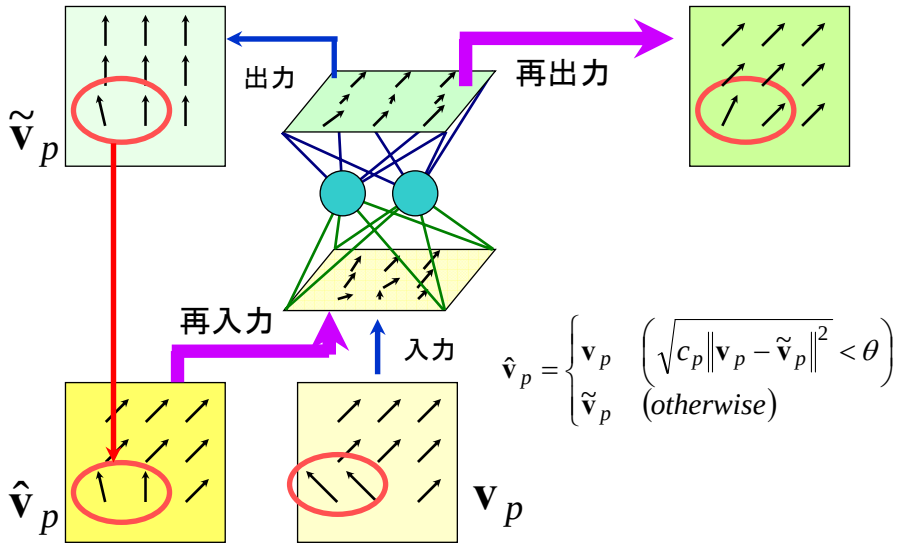
$\varepsilon_{tp} \geq \theta$ なるデータを例外値とする

- 学習時: ノイズによる影響の低減
- 学習後: 移動物体を含むデータへの対応

学習時の例外値の除去

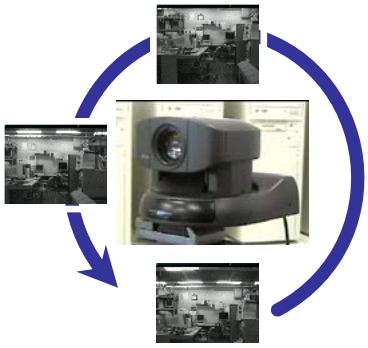


学習後の例外値の除去



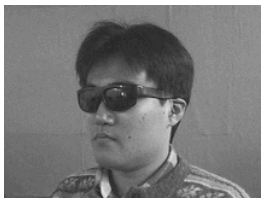
実験

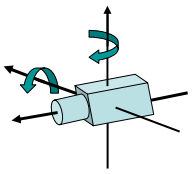
- SONY EVI-D30,
  SightLine Tech. EyeView:
  三脚に固定し一定周期で回転
- 77x60画素 30フレーム/秒
- オプティカルフロー :
  ・勾配法([Simoncelli91]の手法を応用)
  ・成分数 :10x8
- ニューラルネットワーク:
  ・オプティカルフロー
  ・線形3層(素子数 :160 - 2 - 160)



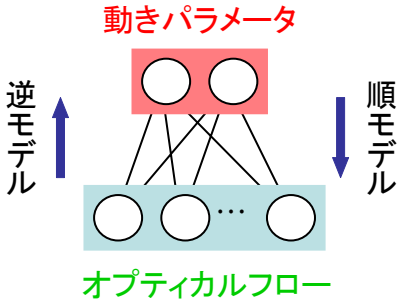
EVI-D30

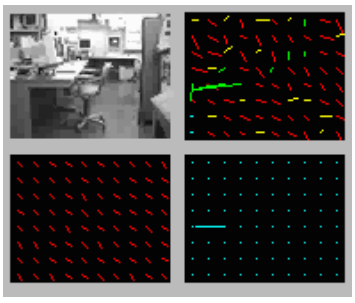
順逆モデルの学習によるカメラ動きの推定





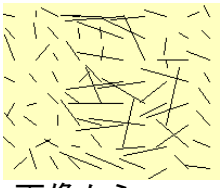
- 恒等写像学習による順逆モデルの獲得
- 背景のフローをロバストに推定



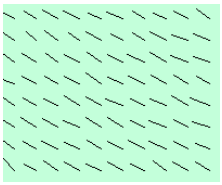


移動物体の検出例

入力画像
(EVI-D30)



画像から
得られたフロー



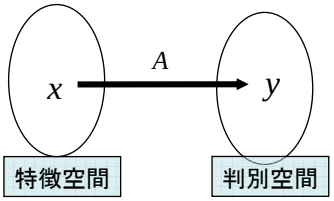
推定されたカメラの
動きによるフロー

例外値の領域

線形判別分析

- 歴史
  - 英国の統計学者フィシャーが、多くの変量に基づく2クラスの判別問題に対して、線形モデルによる解析的な手法を提案（1936年）
  - 2次の統計量に基づく判別基準を最大化（フィシャーの線形判別分析（Linear Discriminant Analysis （LDA））
  - 確率分布を仮定しないノンパラメトリックな統計手法としての多変量データ解析の誕生
- 線形判別写像

$$y = \Psi(x) = A^T x$$

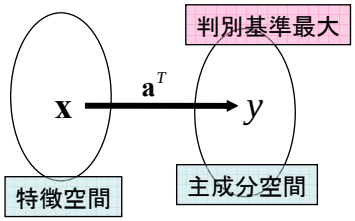


線形判別分析（1次元の場合）

- 訓練データ

$$\{< \mathbf{x}_1, l_i > | \, i = 1, \ldots, N\}$$

- 各クラスの分離度（判別基準）が最大となる新たな特徴量を求める



$$y_i = \mathbf{a}^T(\mathbf{x}_i - \bar{\mathbf{x}}_T) \qquad \bar{\mathbf{x}}_T = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i$$

- 新特徴の統計量

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i = \frac{1}{N}\sum_{i=1}^{N}\mathbf{a}^T(\mathbf{x}_i - \bar{\mathbf{x}}_T) = \mathbf{a}^T(\bar{\mathbf{x}}_T - \bar{\mathbf{x}}_T) = 0$$

$$\bar{y}_k = \frac{1}{N_k}\sum_{l_i = C_k} y_i = \frac{1}{N_k}\sum_{l_i = C_k}\mathbf{a}^T(\mathbf{x}_i - \bar{\mathbf{x}}_T) = \mathbf{a}^T(\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_T)$$

線形判別分析（1次元の場合）

- 新特徴の統計量

$$\sigma_T^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y}_T)^2 = \mathbf{a}^T\left[\frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T\right]\mathbf{a} = \mathbf{a}^T\Sigma_T\mathbf{a}$$

$$\sigma_k^2 = \frac{1}{N_k}\sum_{l_i = C_k}(y_i - \bar{y}_T)^2 = \mathbf{a}^T\left[\frac{1}{N_k}\sum_{l_i = C_k}(\mathbf{x}_i - \bar{\mathbf{x}}_k)(\mathbf{x}_i - \bar{\mathbf{x}}_k)^T\right]\mathbf{a} = \mathbf{a}^T\Sigma_k\mathbf{a}$$

- 平均クラス間分散、平均クラス内分散

$$\sigma_B^2 = \frac{1}{N}\sum_{k=1}^{K} N_k(\bar{y}_k - \bar{y}_T)^2 = \mathbf{a}^T\left[\frac{1}{N}\sum_{k=1}^{K} N_k(\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_T)(\bar{\mathbf{x}}_k - \bar{\mathbf{x}}_T)^T\right]\mathbf{a} = \mathbf{a}^T\Sigma_B\mathbf{a}$$

$$\sigma_W^2 = \frac{1}{N}\sum_{k=1}^{K} N_k\sigma_k^2 = \mathbf{a}^T\left[\frac{1}{N}\sum_{k=1}^{K} N_k\Sigma_k\right]\mathbf{a} = \mathbf{a}^T\Sigma_W\mathbf{a}$$

線形判別分析（導出１）

- 判別基準最大化

$$\eta = \frac{\sigma_B^2}{\sigma_W^2} = \frac{\mathbf{a}^T \Sigma_B \mathbf{a}}{\mathbf{a}^T \Sigma_W \mathbf{a}}$$

- 等価な問題
  - 制約条件 $\sigma_W^2 = \mathbf{a}^T \Sigma_W \mathbf{a} = 1$
  - 最大化

$$\sigma_B^2 = \mathbf{a}^T \Sigma_B \mathbf{a} = 1$$

- 最適化問題（Lagrange乗数）

$$Q(\mathbf{a}) = \sigma_B^2 - \lambda(\sigma_W^2 - 1) = \mathbf{a}^T \Sigma_B \mathbf{a} - \lambda(\mathbf{a}^T \Sigma_W \mathbf{a} - 1)$$

線形判別分析（導出２）

- Qのパラメータに関する偏微分

$$\frac{\partial Q(\mathbf{a})}{\partial \mathbf{a}} = 2\Sigma_B \mathbf{a} - 2\lambda \Sigma_W \mathbf{a} = 0$$

- これから、一般化固有値問題が得られる

$$\boxed{\Sigma_B \mathbf{a} = \lambda \Sigma_W \mathbf{a}}$$

- 最適なパラメータは、Xの分散共分散行列の最大固有値として求まる。ただし、その大きさについては、制約条件を満たす必要がある。

$$\sigma_W^2 = \mathbf{a}^T \Sigma_W \mathbf{a} = 1$$

線形判別分析（多次元の場合）

- 判別基準
  - 同じクラスに属す点はなるべく近く、異なるクラスに属す点は離れる

ただし $J[\Psi] = tr(W_Y^{-1} B_Y)$

W:平均クラス内共分散行列
B:平均クラス間共分散行列

$$W_Y = A^T \Sigma_W A, \qquad B_Y = A^T \Sigma_B A$$

$$\Sigma_W = \sum_{k=1}^{K} \frac{N_k}{N} \Sigma_k, \qquad \Sigma_B = \sum_{k=1}^{K} \frac{N_k}{N} (\overline{\mathbf{x}}_k - \overline{\mathbf{x}}_T)(\overline{\mathbf{x}}_k - \overline{\mathbf{x}}_T)^T$$

- 最適解
  - 最適な係数行列は、固有値問題

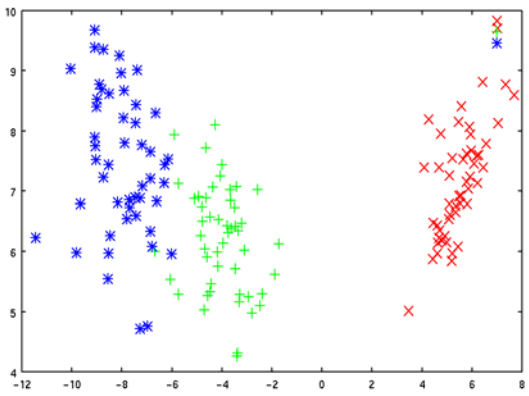
$$\Sigma_B A = \Sigma_W A \Lambda, \qquad A^T \Sigma_W A = I$$

の最大n個の固有値に対応する固有ベクトルを列とする行列として求められる。ただし、Yの次元nは行列のランクの関係から

$$n \le \min(K - 1, m)$$

線形判別分析の例（アヤメのデータの場合）

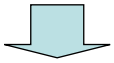
- Fisherのアヤメのデータ
  - 3種類のアヤメの花から4種類の特徴を測定（4次元の特徴ベクトル）
  - 各種類50個のサンプル



非線形判別特徴の抽出

非線形判別特徴の抽出

- 多変量解析手法
  - 一般に線形モデルを仮定
  - データの背後の確率的構造との関係が不明確
    - 解に必要な知識としての2次までの統計量の本質的な意味も明確でない
  - 最適な識別手法としてのベイズ識別との関係も不明確



  - 背後の本質的な構造を明らかにするには、線形写像という制約を取り払って一般の非線形写像を考える必要がある。

非線形判別特徴抽出

非線形重回帰分析

- データ
  - 学習サンプル{x、t}は確率的であり、確率密度分布p(x、t)で表される母集団からの標本と考える
- 非線形回帰式(モデル)

$$y = \Psi(x)$$

- 平均2乗誤差

$$\varepsilon^2[\Psi] = \int\int \| t - \Psi(x) \|^2 \; p(x,t)dxdt$$

非線形重回帰分析(最適解)

- 最適解
  - 変分法を用いて陽に求めることができる

$$y = \Psi_N(x) = \int t \; p(t \mid x)dt$$

  - これは、入力xのもとでのyの条件付平均
- 最小2乗誤差

$$\varepsilon^2_{opt} = \int\int \| t - \Psi_N(x) \| \; p(x,t)dxdt = \sigma_t^2(1-\rho^2)$$

ただし、

$$\rho^2 = \frac{\sigma_{yt}^2}{\sqrt{\sigma_y^2 \sigma_t^2}}$$

  - これは、線形重回帰の場合と同様な関係

線形手法と非線形手法

- 非線形多変量解析の理論
  - データの背後の確率的構造が既知として多変量解析手法を非線形に拡張した
  - 線形の手法と最適な非線形手法との関係は？
- 線形重回帰分析
  - 最適線形写像

$$y = \Psi_{lin}(x) = A^T x + b$$

ただし、

$$A = \Sigma_X^{-1}\Sigma_{XY}$$

$$b = \bar{t} - \Sigma_X^{-1}\Sigma_{XY}\bar{x}$$

関係？

- 非線形重回帰
  - 最適非線形写像

$$y = \Psi_{opt}(x) = \int tp(t \mid x)dt$$

条件付き確率の線形近似

- 線形近似 $L(t \mid x)$
- 評価基準：2乗誤差最小化

$$\varepsilon_{cnd}^2 = \int \| p(t \mid x) - L(t \mid x) \|^2 \; p(x)dx$$

- 条件付き確率の最適線形近似

$$L(t \mid x) = p(t)\{(\bar{x}(t) - \bar{x})^T \Sigma_x^{-1}(x - \bar{x}) + 1\}$$

- 性質

$$\int L(t \mid x)dt = 1$$

$$\int L(t \mid x)\, p(x)dx = p(t)$$

$$\int L(t \mid x)\, p(x)dt = p(x)$$

線形近似としての線形重回帰分析

- 最適非線形写像の条件付き確率をその線形近似で置き換えてみる

$$\int tL(t \mid x)dt = \Sigma_{XT}^{T}\Sigma_{X}^{-1}(x-\bar{x})+\bar{t}$$

これは、まさに、線形重回帰の最適線形写像と同じ

- 非線形最適写像の線形近似
  - 非線形最適写像をxの線形写像で最小2乗近似
  - 評価基準:2乗誤差最小化

$$\varepsilon_A^2 = \int \| \Psi_{opt}(x) - (A^T x + b) \|^2 \; p(x)dx$$

  - 最適な係数:

$$A = \Sigma_X^{-1}\Sigma_{XT}$$

$$b = \bar{t} - \Sigma_X^{-1}\Sigma_{XT}\bar{x}$$

これも、線形重回帰分析の最適線形写像と同じ

誤差の関係

- 線形重回帰で達成される最小2乗誤差

$$\varepsilon_L^2 = \int \| t - \Psi_{lin}(x) \|^2 \; p(x)dx$$

- 非線形回帰で達成される最小2乗誤差

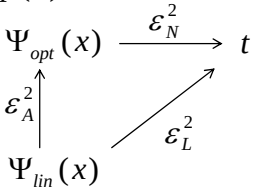
$$\varepsilon_N^2 = \int \| t - \Psi_{opt}(x) \|^2 \; p(x)dx$$

- 非線形最適写像の線形近似で達成される最小2乗誤差

$$\varepsilon_A^2 = \int \| \Psi_{opt}(x) - (A^T x + b) \|^2 \; p(x)dx$$

- 誤差の間の関係

$$\varepsilon_L^2 = \varepsilon_N^2 + \varepsilon_A^2$$



非線形最小2乗判別写像

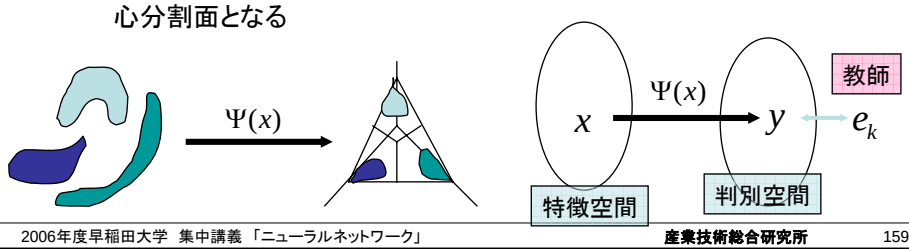
- 各クラスの代表ベクトルを理想出力
  - 平均2乗誤差

$$\varepsilon^2[\Psi] = \sum_{k=1}^{K} P(C_k) \int \| e_k - \Psi(x) \|^2 \; p(x \mid C_k) dx$$

  - 最適解

$$y = \Psi_N(x) = \sum_{k=1}^{K} P(C_k \mid x) e_k$$

  - これは、ベイズ識別（事後確率）と密接な関係がある
  - ベイズ識別境界は、各クラスの代表ベクトルを頂点とする単体の重心分割面となる



非線形最小2乗判別写像（最小2乗誤差）

- 最適解で達成される最小2乗誤差（正規直交系の場合）

$$\varepsilon^2[\Psi_N] = \sum_{k=1}^{K} P(C_k) \int \| e_k - \sum_{l=1}^{K} P(C_l \mid x) e_l \|^2 \; p(x \mid C_k) dx$$

$$= 1 - \sum_{k=1}^{K} \int P(C_k \mid x) P(C_k \mid x) \, p(x) dx = 1 - \sum_{k=1}^{K} \gamma_{kk}$$

ここで、

$$\gamma_{kl} = \int P(C_k \mid x) P(C_l \mid x) \, p(x) dx$$

  - 事後確率の積の期待値で
    - クラス間の確率的関係を要約した、確率の上の統計量

非線形最小2乗判別写像（正規直交系の場合）

- クラス代表ベクトル（教師信号）
  - クラスCkに対して、k番目の要素のみが1で残りの要素がすべて0の2値ベクトル
  - 最適写像

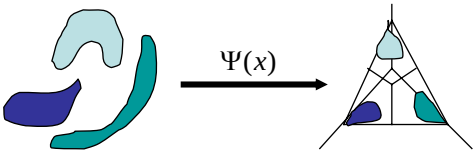
$$y = \Psi_{opt}(x) = \begin{pmatrix} P(C_k \mid x) \\ \vdots \\ P(C_K \mid x) \end{pmatrix}$$

  - ベイズ事後確率を要素とするベクトル
    - つまり、事後確率が最大のクラスに識別すればよいことになる。
    - これは、0－1リスクに対する最適なベイズの識別と同じ結果

非線形判別分析

- 非線形写像

$$y = \Psi(x)$$



- 判別基準

$$J[\Psi] = tr(W_Y^{-1} B_Y)$$

- 最適な非線形写像

$$y = \Psi_N(x) = \sum_{k=1}^{K} P(C_k \mid x)\mathbf{u}_k$$

  - 識別境界は、クラスの代表ベクトルckを頂点とする単体の重心分割面
  - ただし、クラスの代表ベクトルukは、クラス間の確率的な関係を要約した推移行列

$$S = [s_{ij}], \qquad s_{ij} = \int P(C_j \mid x)\, p(x \mid C_i)\, dx$$

の固有値問題から求まる

非線形判別分析

- クラス間の関係を要約する確率行列

$$s_{ij} = \int P(C_j \mid x)\, p(x \mid C_i) dx = \int P(C_j \mid x) \frac{P(C_i \mid x)\, p(x)}{P(C_i)} dx$$

$$= \frac{1}{P(C_i)} \int P(C_j \mid x) P(C_i \mid x)\, p(x) dx = \frac{\gamma_{ij}}{P(C_i)}$$

したがって、

$$\Gamma = [\gamma_{ij}] = \begin{bmatrix} P(C_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & P(C_K) \end{bmatrix} S = PS$$

非線形判別分析（固有値問題）

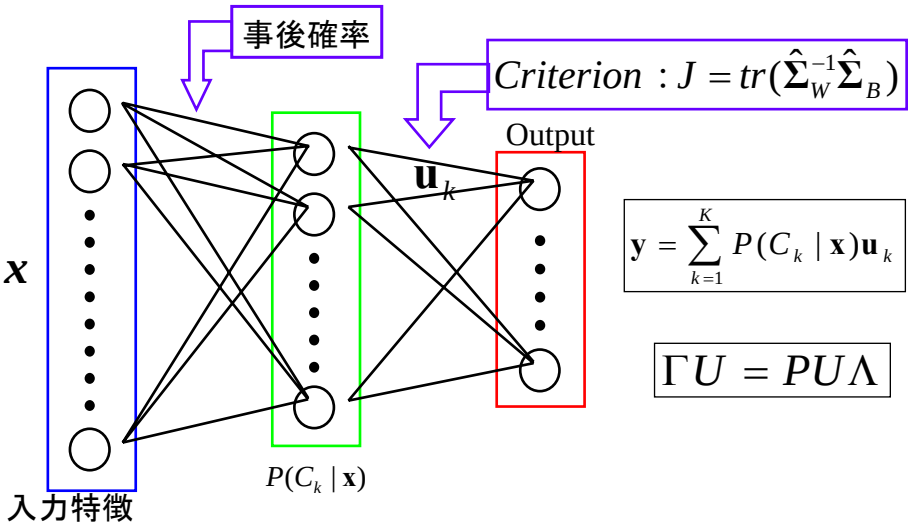
- 固有値問題

$$S \begin{bmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_K^T \end{bmatrix} = \begin{bmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_K^T \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_K \end{bmatrix}$$

$$\Gamma \begin{bmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_K^T \end{bmatrix} = \begin{bmatrix} P(C_1) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & P(C_K) \end{bmatrix} \begin{bmatrix} \mathbf{u}_1^T \\ \vdots \\ \mathbf{u}_K^T \end{bmatrix} \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \lambda_K \end{bmatrix}$$

$$\Gamma U = PU\Lambda$$

非線形判別分析



$$\mathbf{y} = \sum_{k=1}^{K} P(C_k \mid \mathbf{x})\mathbf{u}_k$$

$$\Gamma U = PU\Lambda$$

非線形判別分析の線形近似としての線形判別分析の解釈

- 事後確率の線形近似

$$L(C_k \mid x)$$

- 評価基準:2乗誤差最小化

$$\varepsilon_{cnd}^2 = \int \| P(C_k \mid x) - L(C_k \mid x) \|^2 \; p(x)dx$$

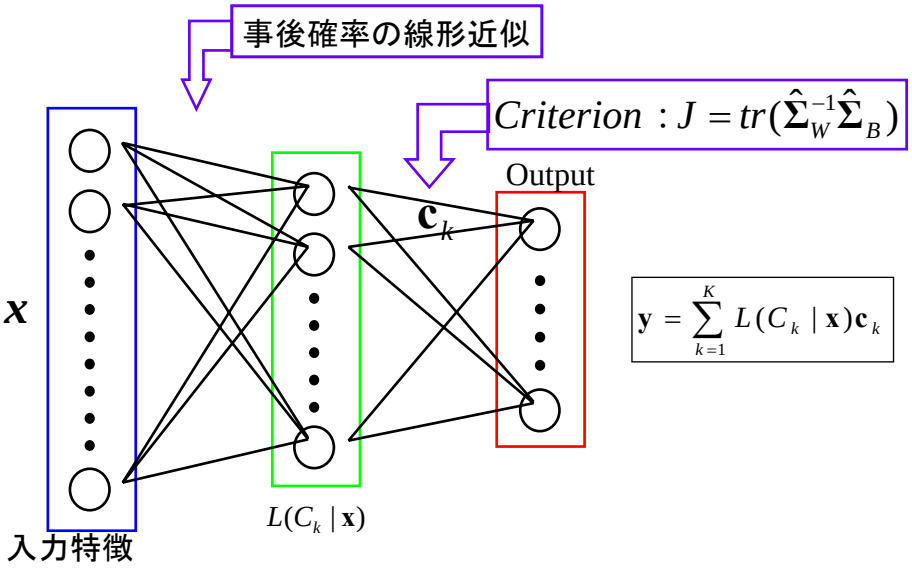
- 最適線形写像

$$L(C_k \mid x) = P(C_k)\{(\mu_k - \mu_T)^T \Sigma_X^{-1}(x - \mu_T) + 1\}$$

- 最適線形判別写像

$$y = \Psi_L(x) = \sum_{k=1}^{K} L(C_k \mid x)c_k$$

非線形判別分析の線形近似としての線形判別分析



一般化線形判別分析

- 多項ロジットモデルにより事後確率の近似
  - 多項ロジットモデルを用いてパターン認識課題を学習させたネットワークの出力

$$\mathbf{p}(\mathbf{x}) = (p_1(\mathbf{x}), \cdots, p_K(\mathbf{x}))^T$$

  - 事前確率

$$\tilde{P}(C_i) = \mathrm{E}\{p_i(\mathbf{x})\} \qquad (i = 1, \cdots, K)$$

  - 固有値問題

$$\tilde{\Gamma} A = \tilde{P}\tilde{U}\tilde{\Lambda}$$

$$\tilde{\Gamma} = \mathrm{E}\{(\mathbf{p}(\mathbf{x}) - \mathrm{E}\{\mathbf{p}(\mathbf{x})\})(\mathbf{p}(\mathbf{x}) - \mathrm{E}\{\mathbf{p}(\mathbf{x})\})^T\}$$

  - 非線形判別写像

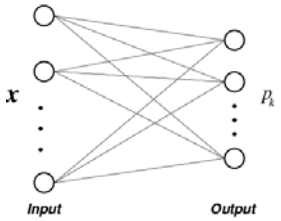
$$\tilde{\mathbf{y}} = \sum_{k=1}^{K} p_k(\mathbf{x})\tilde{\mathbf{u}}_k$$

多項ロジットモデル

Multinomial Logit model

$$p_k = \frac{\exp(\eta_k)}{1+\sum_{m=1}^{K-1}\exp(\eta_m)}, \quad k = 1,\cdots,K-1$$

$$p_K = \frac{1}{1+\sum_{m=1}^{K-1}\exp(\eta_m)} \qquad \eta_k = \mathbf{a}_k^T\mathbf{x}$$



尤度・対数尤度

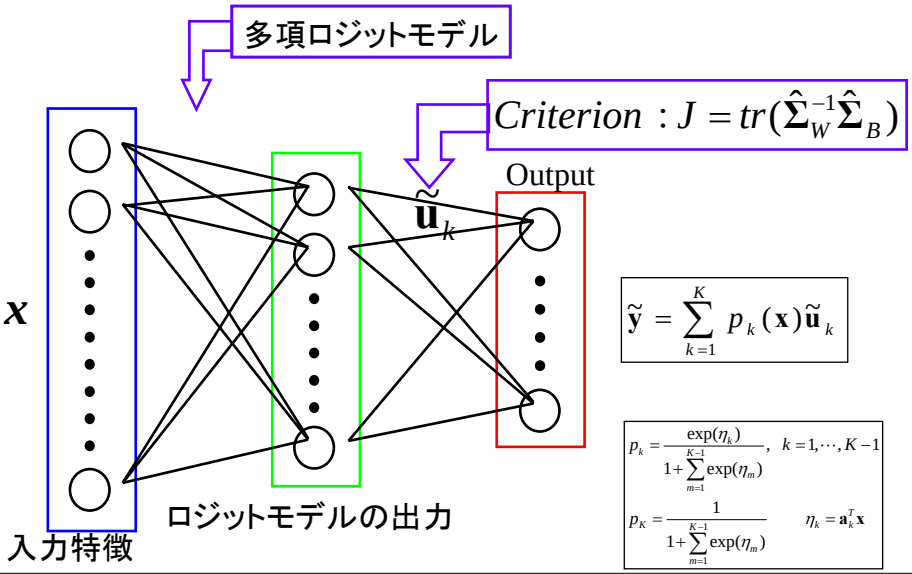
$$P(\mathbf{t}\,|\,\mathbf{x};\mathbf{A}) = \prod_{k=1}^{K} p_k^{t_k}$$

$$l = \log P(\mathbf{t}\,|\,\mathbf{x};\mathbf{A}) = \sum_{k=1}^{K-1} t_k\eta_k - \log\{1+\sum_{m=1}^{K-1}\exp(\eta_k)\}$$

学習則

$$\mathbf{a}_k \Leftarrow \mathbf{a}_k + \alpha\frac{\partial l}{\partial \mathbf{a}_k}, \qquad \frac{\partial l}{\partial \mathbf{a}_k} = (t_k - p_k)\mathbf{x}$$

一般化線形判別分析

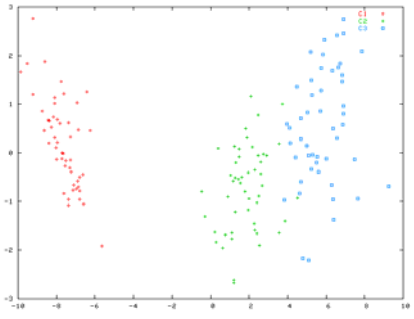


$$\tilde{\mathbf{y}} = \sum_{k=1}^{K} p_k(\mathbf{x})\tilde{\mathbf{u}}_k$$

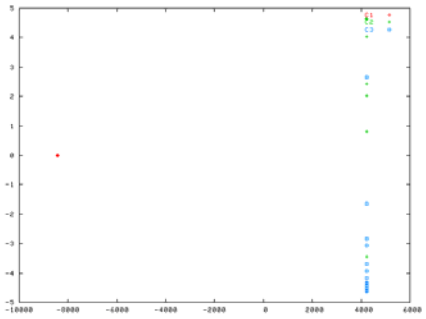
$$p_k = \frac{\exp(\eta_k)}{1+\sum_{m=1}^{K-1}\exp(\eta_m)}, \quad k = 1,\cdots,K-1$$

$$p_K = \frac{1}{1+\sum_{m=1}^{K-1}\exp(\eta_m)} \qquad \eta_k = \mathbf{a}_k^T\mathbf{x}$$

構成された判別空間（アヤメのデータ）

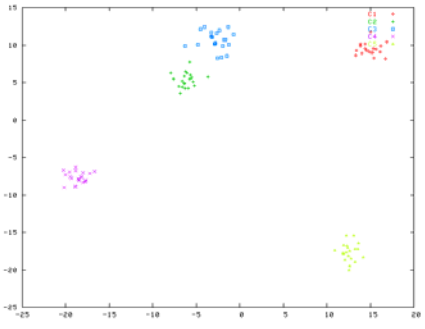


線形判別分析の結果

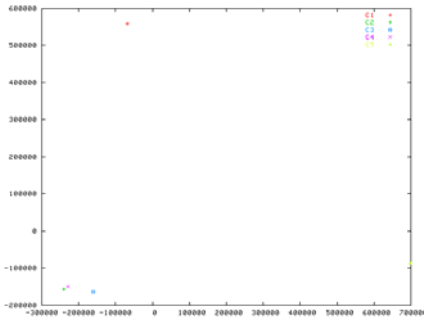


多項ロジットモデルの出力の判別分析の結果

構成された判別空間（話者識別）

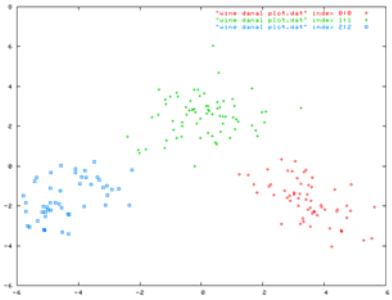


線形判別分析の結果

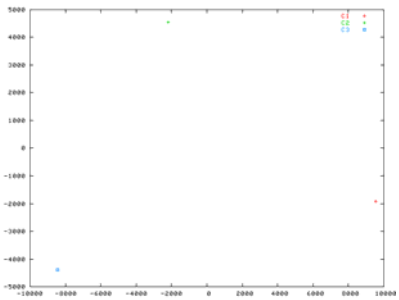


多項ロジットモデルの出力の判別分析の結果

構成された判別空間（ワインの識別）



線形判別分析の結果

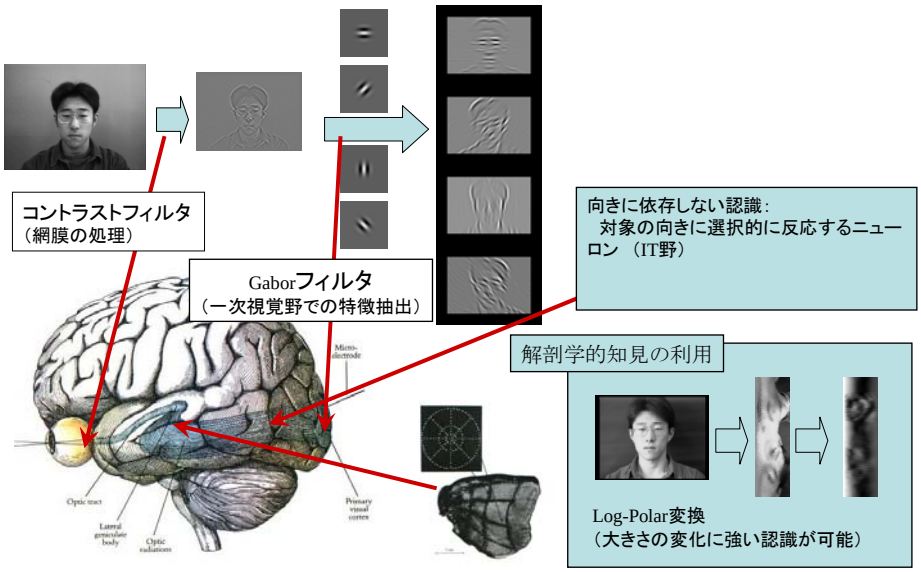


多項ロジットモデルの出力の判別分析の結果

特徴選択

- どの特徴が有効かを組み合わせ的に探索
  - 前向き探索（有効な特徴を順次追加する方法）
  - 後ろ向き探索（すべての特徴を含む特徴ベクトルから不要な特徴を削除）
  - 探索的方法（特徴の組み合わせを探索、遺伝的アルゴリズムなど）
- 特徴選択の基準が重要
  - Cross-Validation
  - 情報量基準(AIC)
  - Minimum Description Length （MDL)法

選択的注意の機構の利用



認識に最適な特徴点の選択

特徴点選択：特徴点の中からある基準に適した特徴の組を選択

- 全ての組み合わせを調べるのは難しい
  ⇒ 準最適な探索法を利用

$$n_p = \frac{d!}{(d-p)!\,p!}$$

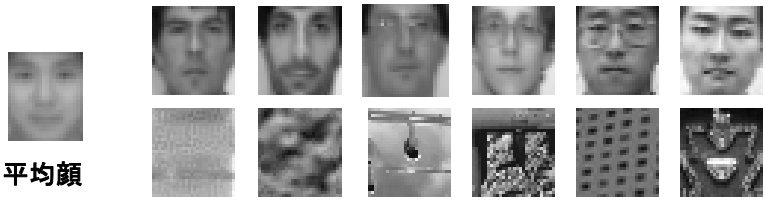
  - SFS： 0点からスタートし，1点づつ特徴点を選択，追加
  - Plus-L, take away-R Selection(L-R)： L点追加，R点削減

特徴点，選択基準，特徴点の選択方法を決定

- 特徴点：**画像中の各点に貼りついた特徴ベクトル**
- 選択基準：**未学習の顔と顔以外の画像に対する識別率**
- 選択の方法：**SFS, L-R**

特徴点の選択実験に用いた画像セット

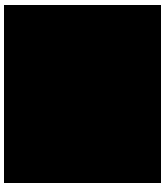
- 実験に用いた画像 （30x28画素）
  - 顔画像： 大きさと位置を正規化した顔画像（Web, MIT）
  - 顔以外の画像： 顔検出に失敗した画像のクラスタリング
- 顔と顔以外の画像を3つのセットに分割
  - 学習用セット： 顔（100枚） → 平均特徴をモデルとした
  - 変数選択用セット： 顔（300枚），顔以外（1,000枚）
  - 評価用セット： 顔（325枚），顔以外（1,000枚）

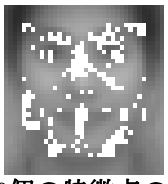


平均顔

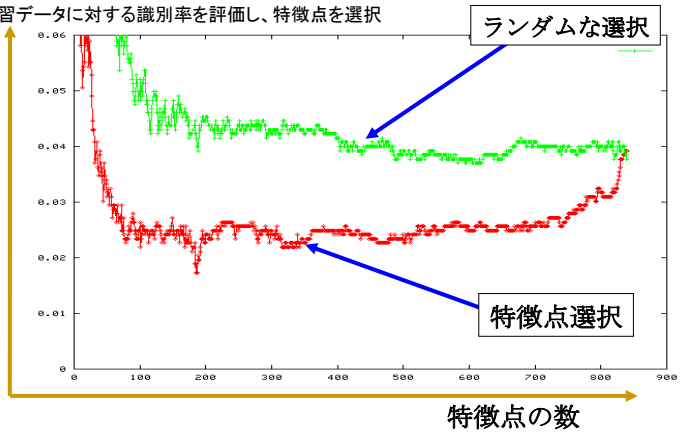
顔検出に有効な特徴点の選択

- 未学習データに対する識別率を評価し、特徴点を選択



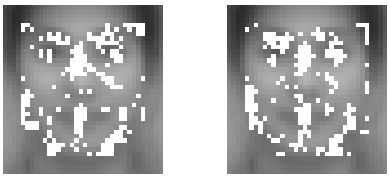


200個の特徴点の分布



認識の高速化

- 選択した初めの200点までを認識に利用



200/840 = 0.238

⇩

認識の高速化

選択された特徴点の順番に従ってマッチングを行う
200点まで見なくても識別可能 ➡ 更なる高速化

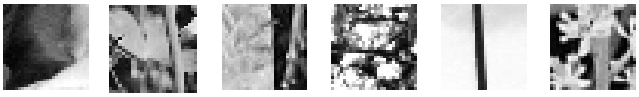
探索の打ち切りによる高速化

モデルからの距離: 少ない特徴点で顔以外を識別可能

$$\text{Distance} = \sum_{i=1}^{k} \text{Dist}\ (i) > \theta$$ ➡ Non-Face

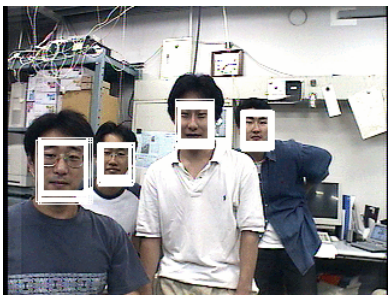
Face $0 \le \text{Dist}\ (i) \le 1$ Non-Face

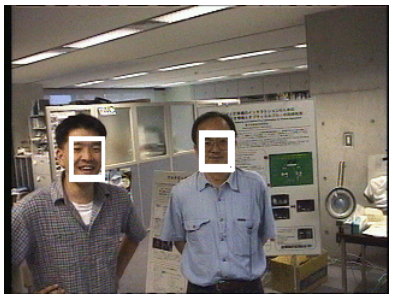
一般に顔よりも顔以外の方が面積が広い ➡ 高速化



- ランダムに選択した1,000枚の顔以外の画像に打ち切りを適用
  ➡ 平均95.5個の特徴点だけで顔以外であると識別できた

顔検出結果の例





選択した初めの200個の特徴点を用いた場合

160x120画素，大きさを5段階変化（0.1倍づつ）
0.45 sec./frame（Pentium III 800MHz Dual）
（探索打ち切り，並列計算，使用する方向を半分）

顔検出結果

	Detection Rate	False Negative	False Positive
All points	81.0%	148 /780	13 /260,682,715
Stepwise Feature Selection	93.2%	53 /780	35 /260,682,715
Plus-L, take away-R Selection (L10-R9)	94.2%	45 /780	53 /260,682,715

◆ 特徴点を選択することにより，汎化能力が飛躍的に向上

➡ 顔の本質的特徴を抽出できた

顔検出の例

歩行者の検出

- コンピュータによる対象の自動認識
  - 対象検出： 特定の対象とその他の識別 汎化能力が必要
  - 検出対象の識別： クラス間の識別
- 歩行者検出
  - 服装，手足の動き，体型，荷物のあるなし ＝＞ 変動が大きい
  - 応用例： 監視，運転者へのサポート，ビデオデータからのサーチ（index）
- 実環境下での応用 ＝＞ 明るさの変化への対応
  - 顔検出で実績のあるコントラスト特徴やガボール特徴の利用
  - 従来法： Oren CVPR97 Harr wavelet + SVM

歩行者検出に有効な特徴点の選択

- 歩行者の画像： 変動が大きい
  背景等の不必要な情報が多く含まれる
  汎化能力を低下させる可能性

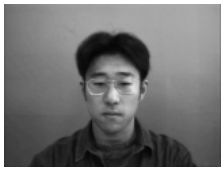
識別タスクに無関係な情報を取り除きたい

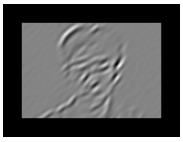
- 従来法： Mohan（PAMI2001）
  - 意図的に選択した4領域（頭，左，右腕，足）を利用して識別能力を改善

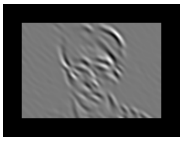
▶ 人間が意図的に決めるべきでない
▶ コンピュータがデータに基づいて（経験的に）決めるべき

変数選択（評価基準：未学習サンプルに対する識別率）
変数選択した場合としない場合の比較

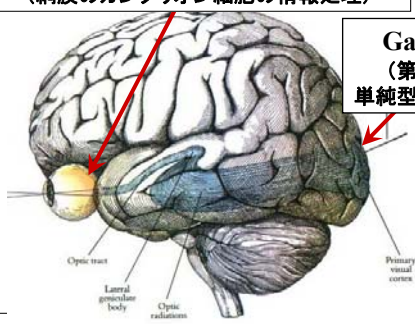
初期視覚の特徴抽出を模倣







コントラストフィルタ
（網膜のガングリオン細胞の情報処理）



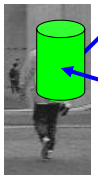
Gabor フィルタ
（第一次視覚野の
単純型細胞の特徴抽出）

実験では8方向

変数選択を用いた不要な情報の削減

変数選択: 変数の組の中からある基準に適した変数を選択
- **全ての組み合わせを調べるのは難しい → 準最適な探索法**

Sequential Backward Selection
- 全てを利用する場合からスタート
- 評価基準に適さない変数を1個づつ削減



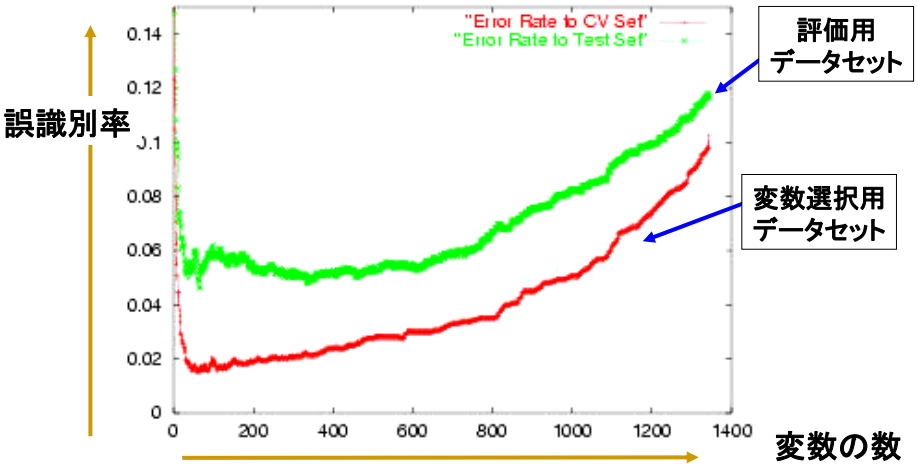
- **変数: 画像中の各点に貼りついたコントラストガボール特徴**
- **選択基準: 未学習の人と人以外の画像に対する識別率**

明らかに不要な情報の削減 → 計算コストの低いマッチング

実験に用いた画像セット

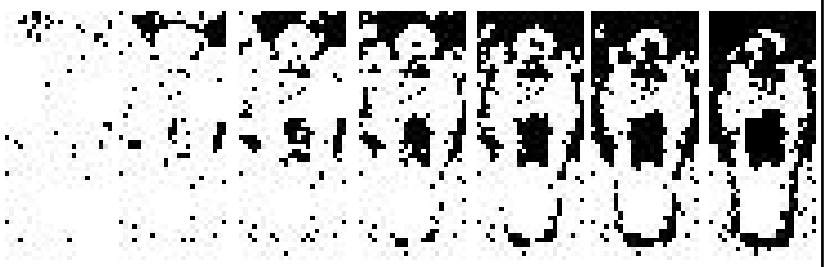
- 実験に用いた画像 （128x64画素）
  - 人画像: MIT CBCL人画像データベース 924枚
  - 人以外の画像: ランダムに選択した画像 2,700枚
- 人と人以外の画像を3つのセットに分割
  - 学習用セット: 人（100枚）, 人以外（300枚）
  - 変数選択用セット: 人（400枚）, 人以外（1,200枚）
  - 評価用セット: 人（424枚）, 人以外（1,200枚）

変数選択の結果（マッチングによる評価）



エラーの多くは人画像 ➡ マッチングでは難しい

変数選択の結果



変数: 1,300 1,200 1,100 1,000 900 800 700

黒: 取り除かれた場所

目的: 明らかに不要な情報の削減
➡ 700個の変数までで評価

変数選択の効果

評価用セットに対する識別結果（人：424，人以外1200）

変数の数	誤識別数	人画像 誤識別数	人以外の誤識 別数	対数尤度
1,344	26	12	14	-88.15
1,300	22	14	8	-85.28
1,200	21	14	7	-82.17
1,100	21	11	10	-77.49
1,000	20	9	11	-77.30
900	21	10	11	-79.34
800	22	11	11	-99.00
700	24	12	12	-93.51

False Positive Rateの計算

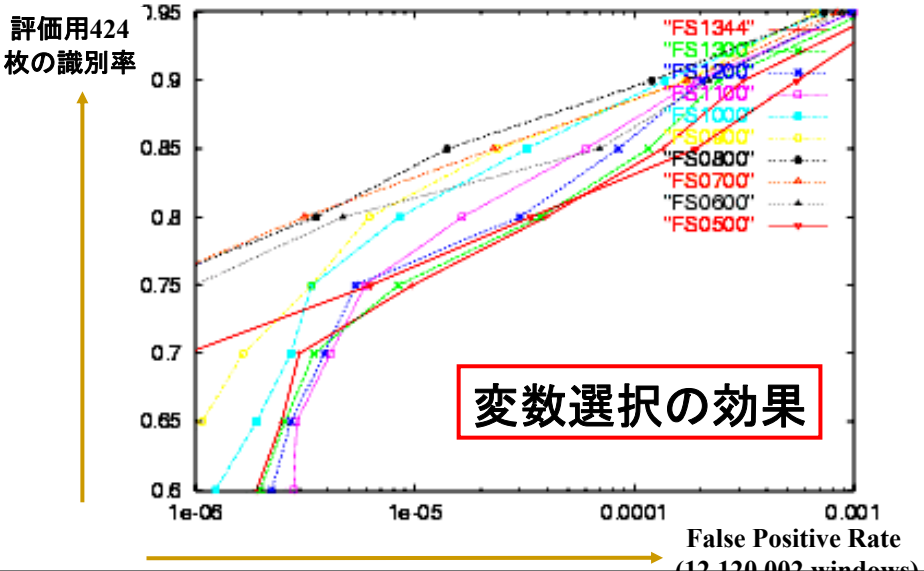
人 or 人以外？

人と識別 → 誤検出

画像中の全ての領域で大きさを変えながらマッチング
全領域に対する閾値以上となった領域の割合を計算

False Positive Rate = 誤検出数 / 100枚の画像中の全候補数

ROC curve

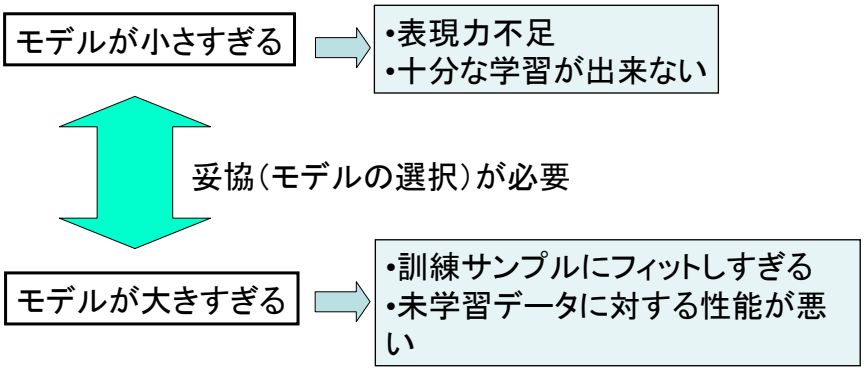


汎化性

汎化性

- 学習の目的
  - 学習データに対して良い結果を与えることでは無く、未学習のデータに対して性能が良いこと
    - 特に、パターン認識に用いる場合には、学習データでいくらうまく識別出来ても、未知のデータに対してうまく識別出来なければ意味が無い
  - 汎化性
    - 未知データに対する性能

汎化性



汎化性能の評価

- 汎化性能の高い識別器の設計
  - データの背後にある確率的な関係を表現するのにちょうど良い複雑さのモデルを見つけて、識別器を設計する必要がある
- モデル選択
  - いくつかのモデルの中から汎化性能の最も高い識別器を選択すること
- モデル選択のための評価基準
  - 未知サンプルに対する識別性能を直接評価することは出来ない！
  - 学習に利用した訓練サンプルに対する識別率は、モデルの複雑度を増せばどんどん小さくなる ＝＞ モデルの選択には利用できない

汎化性の評価（その1）

- 非常に多くのサンプルを用意できる場合

1. サンプルを訓練用サンプルとモデル選択用サンプルに分割
2. 訓練サンプルを用いて各モデルのパラメータを決定
3. モデル選択用サンプルを用いて汎化性能を評価

  - 利用可能なサンプル数が多い場合には、もっとも実際的で有効な方法

交差確認法（Cross Validation）

1. サンプル集合をK個の部分集合に分割
2. 評価用に1個の部分集合を残して、残りのK-1個の部分集合に含まれるすべてのサンプルを用いてパラメータを学習
3. 評価用の部分集合の取り出し方はK種類あるので、それらの平均で汎化性能を評価

- leave-one-out法
  - N個のサンプルがある場合、K＝Nとして、各サンプルをひとつの部分集合とする方法

$$CV = \frac{1}{N}\sum_{i=1}^{N} L[\mathbf{t}_i, f^{-i}(\mathbf{x}_i)]$$

交差確認法（ハイパーパラメータの決定)

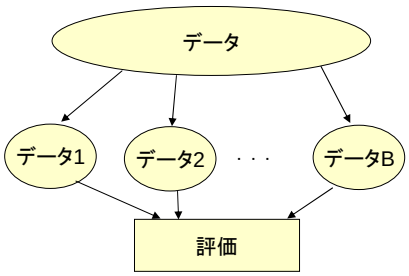
識別器に学習では決定できないようなパラメータ（例えば、正則化パラメータ）が含まれている場合にも、最適なパラメータを決定するためにも利用可能

$$CV(\alpha) = \frac{1}{N}\sum_{i=1}^{N} L[\mathbf{t}_i, f^{-i}(\mathbf{x}_i, \alpha)]$$

ブートストラップ（Bootstrap)

1. 利用可能なサンプル集合から重複を許して無作為に訓練サンプルを抽出
2. このようなデータセットをB個用意し、各データセットを訓練サンプルとして識別器を学習
3. 各識別器の識別性能の平均で汎化性能を評価

$$BS = \frac{1}{B}\sum_{b=1}^{B}\frac{1}{N}\sum_{i=1}^{N} L[\mathbf{t}_i, f^b(\mathbf{x}_i)]$$



- 欠点
  - 訓練用サンプルと評価用サンプルに重なりが生じる
    - 過適応しやすく、必ずしも良い汎化性能の推定値が得られない

ブートストラップ（改良）

- leave-one-out法の考え方を取り入れて、訓練用のブートストラップサンプルに含まれないサンプルのみを評価に利用

$$BS_{loo} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{|C^{-i}|}\sum_{b\in C^{-i}} L[\mathbf{t}_i, f^b(\mathbf{x}_i)]$$

- ここで、C^{-i} は、i番目のサンプルが含まれていないブートストラップデータセットの集合

bagging

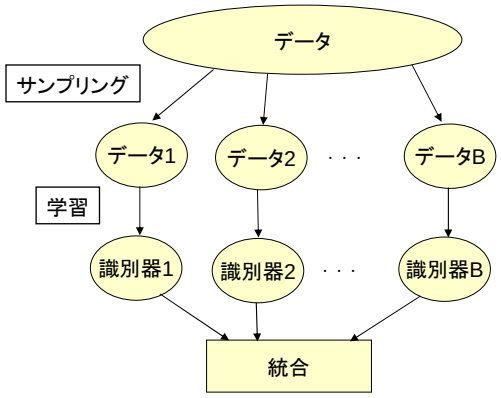
- 複数の識別器を利用して汎化性能の高い識別器を構成するアンサンプル学習のひとつ

1. ブートストラップデータセットと同じ数の識別器を用意
2. それぞれの識別器のパラメータを各ブートストラップサンプルに基づいて学習
3. それらの識別器を統合した識別器を構成

$$f_{bagg} = \frac{1}{B}\sum_{b=1}^{B} f^{b}(\mathbf{x}_i)$$

バッギング（bagging）

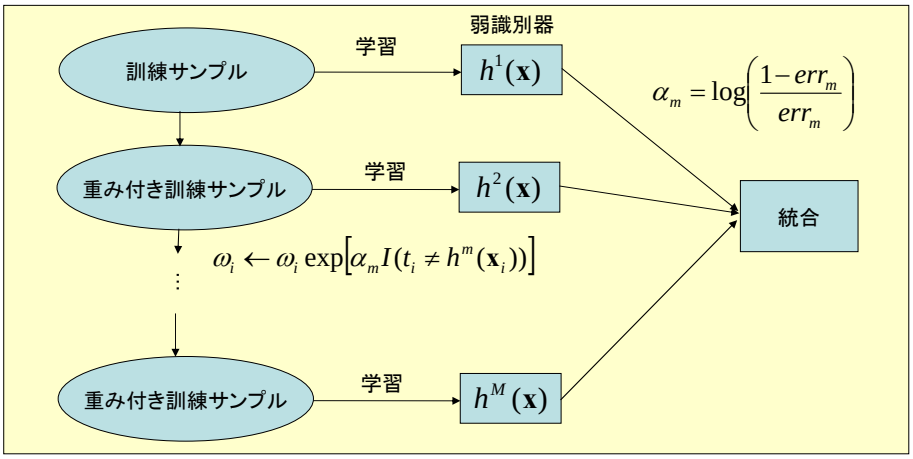
- 異なる訓練データで学習した複数の識別器を利用して汎化性能の高い識別器を構成



$$f_{bagg} = \frac{1}{B}\sum_{b=1}^{B} f^{b}(\mathbf{x}_i)$$

ブースティング(AdaBoost)

$$H(\mathbf{x}) = \mathrm{sign}\left(\sum_{m=1}^{M} \alpha_m h^m(\mathbf{x})\right)$$



情報量基準

- AIC (Akaike Information Criterion)
  - 赤池が最大対数尤度と期待平均対数尤度の間の偏りの解析的評価から導出

$$AIC = -2(\text{対数尤度}) + 2d$$

  - 学習データに対する当てはまりが悪いと第1項大きくなる
  - 第1項に大きな差が無い場合には、第2項(d:自由度)により、自由度の小さなモデルが選択される。
- BIC (Bayesian Information Criterion)、MDL (Minimum Description Length)

$$BIC = MDL = -2(\text{対数尤度}) + (\log N)d$$

学習における汎化性能の向上の工夫

- 学習の際に識別に関係しないようなパラメータを自動的に無視することで実質的な複雑さを抑制する
- Shrinkage Method
  - 識別に貢献しないパラメータを無視するように、学習のための評価基準に、パラメータの絶対値が大きくなり過ぎないようなペナルティ項を追加

$$Q(\boldsymbol{\theta}) = L(\boldsymbol{\theta}) + \lambda \sum_{j=1}^{P} \theta_j^2$$

  - 例
    - 最小2乗識別関数（重回帰分析） ＝＞ リッジ回帰
    - パーセプトロン ＝＞ サポートベクターマシン
    - ニューラルネットワーク ＝＞ Weight Decay

リッジ回帰

- ペナルティ
  - パラメータが大きくなり過ぎないようにする

$$\sum_{j=1}^{M} w_j^2 \rightarrow \min$$

- 評価基準

$$Q(\mathbf{w}, h) = \sum_{i=1}^{N} \varepsilon_{emp}^2 + \lambda \sum_{j=1}^{M} w_j^2 \rightarrow \min$$

学習則（リッジ回帰）

- パラメータの更新式

$$w_j \Leftarrow w_j + \alpha \sum_{i=1}^{N} (t_i - y_i) x_{ij} - 2\alpha\lambda w_j$$

$$h \Leftarrow h + \alpha \sum_{i=1}^{N} (t_i - y_i)(-1)$$

– 第3項は、結合荷重の絶対値が小さくなる方向に作用
＝＞ 予測に不必要な無駄なパラメータを0にする効果

最適解（リッジ回帰）

- 最適なパラメータ

$$\mathbf{w}^* = (\tilde{X}^T \tilde{X} + \lambda I)^{-1} \tilde{X}^T \tilde{\mathbf{t}}$$

– 相関行列の対角要素にλを加えてから逆行列を計算
＝＞ 逆行列が不定になることを防げる

ロジスティク回帰（Weight Decey）

- ペナルティ
  - パラメータが大きくなり過ぎないようにする

$$\sum_{j=1}^{M} w_j^2 \rightarrow \min$$

- 評価基準

$$Q(\mathbf{w}, h) = \sum_{i=1}^{N} (\log(1+\exp(\eta_i)) - u_i \eta_i) + \sum_{j=1}^{M} w_j^2 \rightarrow \min$$

学習則（ロジスティック回帰）

- パラメータの更新式

$$w_j \Leftarrow w_j + \alpha \sum_{i=1}^{N} (u_i - y_i) x_{ij} - 2\alpha\lambda w_j$$

$$h \Leftarrow h + \alpha \sum_{i=1}^{N} (u_i - y_i)(-1)$$

  - 第3項は、結合荷重の絶対値が小さくなる方向に作用
    ＝＞ 予測に不必要な無駄なパラメータを0にする効果

特徴選択

- どの特徴が有効かを組み合わせ的に探索
  - 前向き探索（有効な特徴を順次追加する方法）
  - 後ろ向き探索（すべての特徴を含む特徴ベクトルから不要な特徴を削除）
  - 探索的方法（特徴の組み合わせを探索、遺伝的アルゴリズムなど）
- 特徴選択の基準が重要
  - Cross-Validation
  - 情報量基準(AIC)
  - Minimum Description Length （MDL)法

EEGを利用したブレインコンピュータインタフェースのための特徴選択

背景

・ALS（筋萎縮性側索硬化症）患者
・脊椎損傷患者の一部

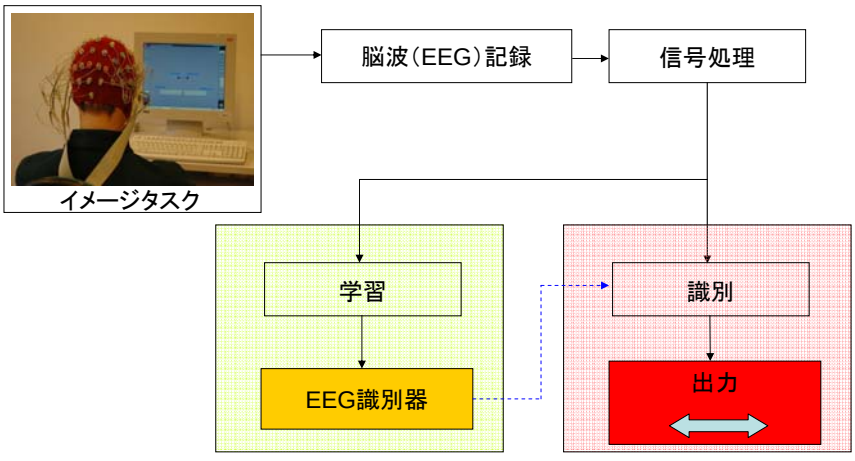
⇒ 発話や手足によるコミュニケーションが困難

脳は正常に活動している

脳波（EEG）を読み取り思考・意思の出力の支援
ブレインコンピュータインタフェース (BCI)
脳 ⇔ コンピュータ

十分な精度や応答速度が得られていない

BCIシステムの概要



データクレンジング ～特徴選択～

- EEGから計算された特徴
  冗長な情報や不必要な情報を含む

汎化性能 低下
計算時間 増加 → 特徴選択

汎化性 向上
応答時間短縮

- Millan et al. 2002
  決定木を識別器とした特徴選択
- Lal et al. 2004
  線形SVMのマージンの距離を評価基準とし逐次的にEEGの測定電極を選択

手法

- 特徴選択の手法
  Backward Stepwise Selection
  - 識別器
    カーネルSVM
  - 識別器の評価方法
    5-fold Cross Validation

より高性能な識別器　より精密な評価方法

得られた特徴 ⇒ SVM ⇒ EEG識別

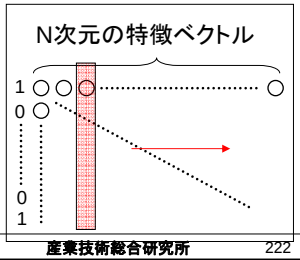
Backward Stepwise Selection による特徴選択

- **特徴選択**
  すべての特徴の組み合わせを用い識別器を構成
  それぞれを評価　組み合わせを探索

膨大な組み合わせ

全ての特徴を含むモデルから特徴を1 個づつ取り除き評価
最も良い特徴の組を選び出す

Backward Stepwise Selection

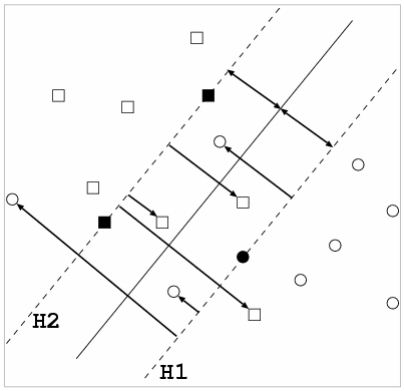


サポートベクターマシン（SVM）

線形識別素子を拡張しサンプルを2クラスに分類する学習・識別手法

- マージン最大化
- ソフトマージン
- カーネルトリック

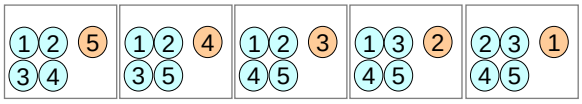
カーネルSVM



識別器の評価基準

- 5 fold Cross Validation

学習サンプルを5つに分け、4つで学習、1つで評価
5通り試行、その平均で識別器の性能を評価する

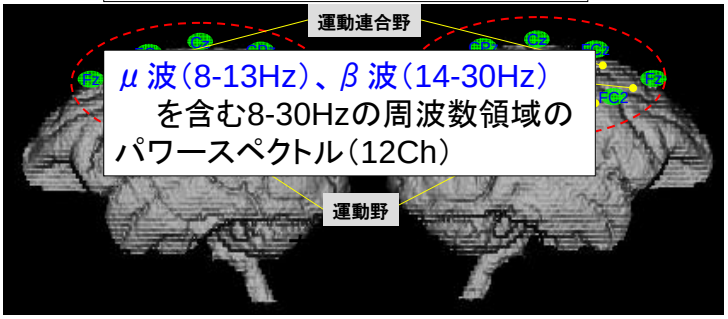


識別率の平均 → Cross Validation rate

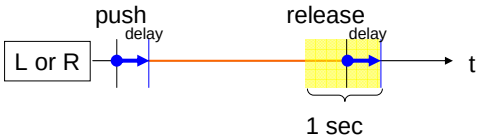
EEGデータ(1)

- 実際の右手、左手の指の動きに関連したEEGを利用

大脳中心部(運動野、感覚野)
13ヶ所から計測される脳波



EEGデータ（2）



画面上に視覚刺激が現れると
刺激に従い左右の指でボタンを押す

・健康な右利きの男性 1名
・50回計測ごとに2,3分の休憩
・一日で合計700回

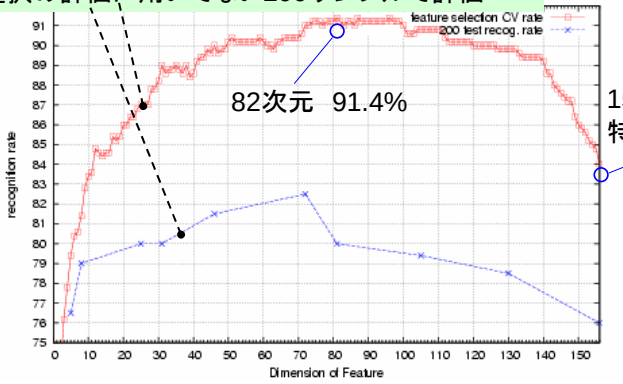
EEGデータ（3）

EEG → 13電極 × 12チャンネル（8-30Hz）
156次元特徴の教師信号（左右）つきのサンプル
2クラスに識別

700個のサンプルのうち
ランダムに選択した500個で評価

実験Ⅰ（1） 提案する特徴選択

特徴選択のそれぞれのフェーズで得られた特徴を用い
500サンプルで識別器を構成
特徴選択の評価に用いてない200サンプルで評価



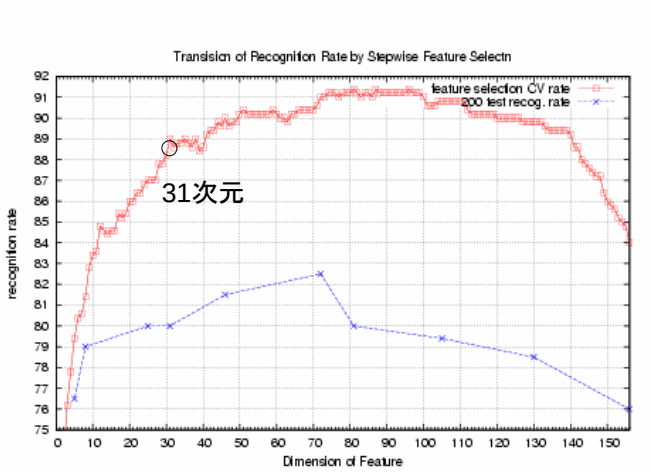
実験Ⅰ（2） 結果と考察

- 特徴選択により
  - 識別器の性能が向上した 84% → 91.4%
  - 特徴の次元を削減することができた
    156次元→82次元
  - 特徴選択に用いていないサンプルに対しても識別性能が向上する

どのような特徴が有効か？
測定部位 周波数帯

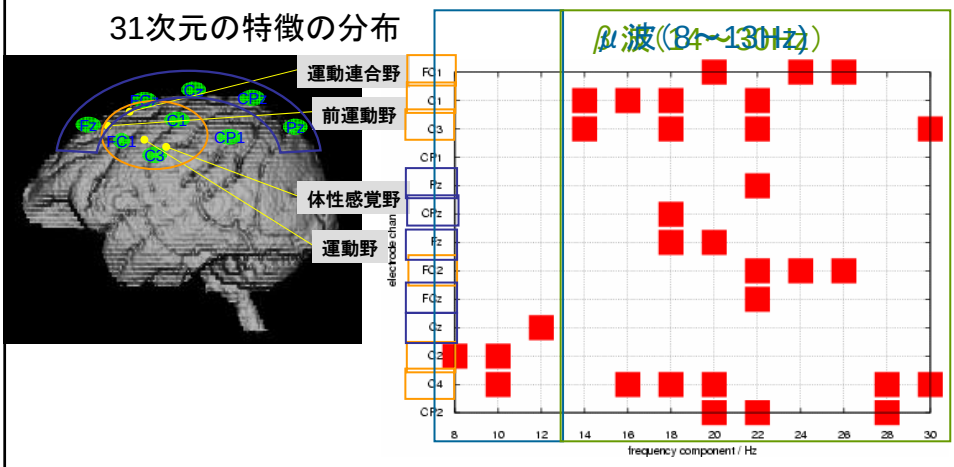
・識別に不必要なランダムな特徴
・必要な特徴と相関の高い特徴 除去

実験Ⅰ（3）



実験Ⅰ（4）得られた特徴の分布

31次元の特徴の分布



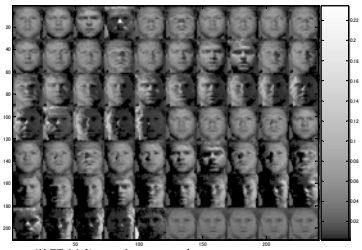
学習における汎化性能の向上の工夫

- 人工的な変動の付加
  - 入力特徴ベクトルや識別器のパラメータに人工的な変動を付加
  - なぜ旨く働くか？
    - もし識別に貢献しないパラメータがあるとすると、付加した変動の影響が出力にまで伝えられて、識別性能が劣化する
    - 学習アルゴリズムは、そうした性能劣化をなるべく抑制しようとするため、結果的に、不必要なパラメータをゼロにする効果がある
  - 変動付加の例
    - 正則化の観点からデータを補間するような多数の学習用データを生成（赤穂1992）
    - 多層パーセプトロンの中間層の各ニューロンの入力に、平均0、分散σの正規ノイズを付加（栗田1993)
    - 多層パーセプトロンの結合荷重にノイズを付加（Murray1999）

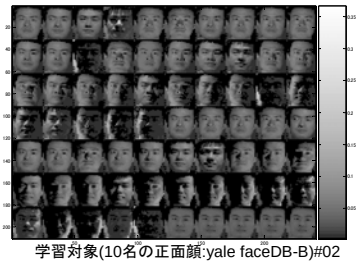
照明条件の変動を学習するには？

変動の学習による汎化性能の向上

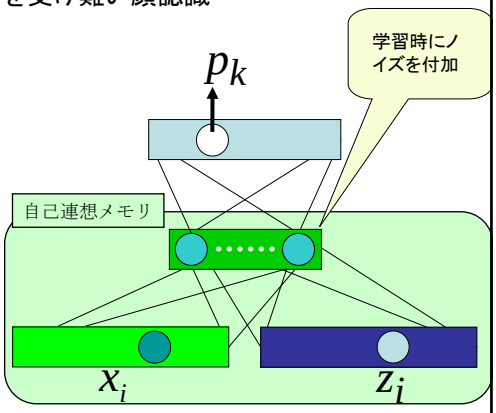
自己連想制約付3層NNによる照明条件の影響を受け難い顔認識



学習対象(10名の正面顔:yale faceDB-B)#01

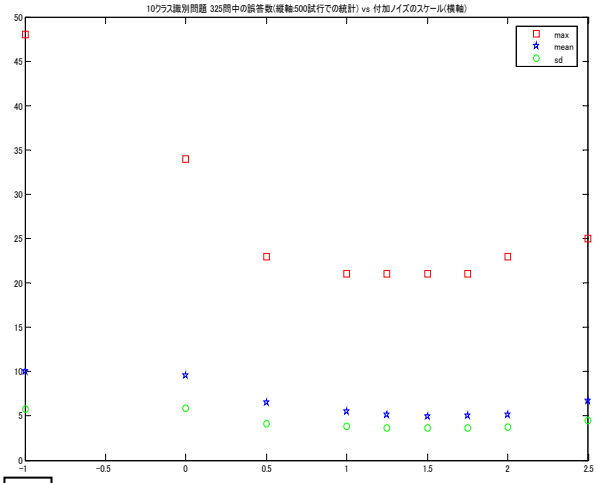


学習対象(10名の正面顔:yale faceDB-B)#02



自己連想メモリによる変動をモデル化し、学習時に中間層にノイズを付加することで対象に依存した変動を自動的に生成しその変動を吸収するような学習を行わせることで汎化性能を向上させる

実験結果 [1] (325-train), 325-tst



従来法

提案手法(ノイズの分散0〜2.5)

カーネル学習法

サポートベクターマシン（SVM）

- 単純パーセプトロン（線形しきい素子）
  - 基本的な構造は、ニューロンモデルとして最も単純な線形しきい素子（McCulloch & Pittsモデル）
  - 2クラスの識別問題に対して有効
  - Vapnik等が、単純パーセプトロンのよい性質を保ちつつ、数理計画法や関数解析に関わるいくつかの工夫を加えてSVMを実現
- 汎化性能向上の工夫（マージン最大化）
  - 未学習データに対して高い識別性能（汎化性能）を得るための工夫（マージン最大化） <= Shrinkage法
  - 正則化やBayes推定、スパース表現とも関連
- 高次元化（カーネルトリック）
  - カーネルトリックで非線形に拡張したSVMは、パターン認識の能力に関して、現在知られている中で最も優れた学習モデルのひとつ

SVMの問題設定

- 識別関数による2クラスの識別
  - 外界からd次元の入力パターンxが与えられたとき、これを2つのクラスのどちらかに識別。
  - クラスのラベルを1と-1に数値化
  - 識別関数：入力パターンからクラスラベルへの関数

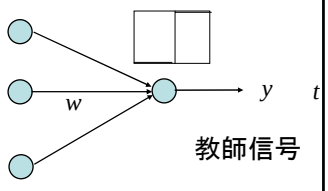
- 学習
  - N個の特徴ベクトルとそれぞれに対する正解のクラスラベルを訓練サンプルとして、それらが正しく識別されるような識別関数を求める
  - 訓練サンプルに含まれない入力パターンに対しても出力の誤りをできるだけ小さくしたい（汎化性能）

線形しきい素子

- 線形しきい素子（単純パーセプトロン）

$$y = \mathrm{sgn}\left[\sum_{i=1}^{M} \omega_i x_i - h\right] = \mathrm{sgn}\left[\mathbf{w}^T \mathbf{x} - h\right]$$

$$\mathrm{sgn}(\eta) = \begin{cases} 1 & \text{if } \eta \geq 0 \\ -1 & \text{otherwise} \end{cases}$$

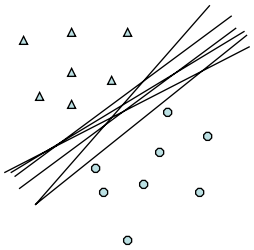


  - 入力xがシナプス荷重wに比例して内部ポテンシャルに加算され、しきい値hを超えたところで出力1を出力する
    - 幾何学的には、入力空間をしきい値hで決まる超平面で二つにわけ、一方に1を、もう一方に－1を割り当てる
  - 線形分離可能
    - すべてのサンプルに対して正しい出力を出すようにパラメータを調節可能

マージン最大化

- よりよい超平面とは？
  - 学習用のサンプル集合を線形分離可能でも、それを実現する超平面は一意でない
  - 訓練サンプルすれすれを通る超平面よりも、多少余裕をもった超平面の方が良い

  ＝＞余裕をどうやってはかる？

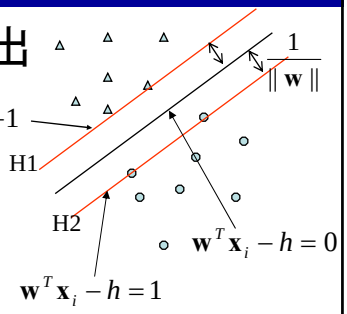


- マージン
  - 超平面と訓練サンプルとの距離の最小値

評価関数の導出

- マージンの大きさ
  - 線形分離可能
    - すべてのサンプルが制約条件

$$t_i(\mathbf{w}^T\mathbf{x}_i - h) \geq 1$$



を満たすようにできる

つまり、2枚の超平面H1とH2をはさんでラベル1のサンプルとラベル－1のサンプルが分離されており、2枚の超平面の間には1つもサンプルがない

  - マージンの大きさ

$$\frac{1}{\|\mathbf{w}\|}$$

SVMの最適化問題

- 制約条件付最適化問題
  - 目的関数：マージン最大化

$$L(\mathbf{w}) = \| \mathbf{w} \|^2 \rightarrow \min$$

  - 」制約条件：線形分離可能

$$t_i(\mathbf{w}^T \mathbf{x}_i - h) \geq 1 \quad \text{for} \quad i = 1, \ldots, n$$

制約条件付き最適化問題の解法

- Lagrange乗数を用いて変形

$$L(\mathbf{w}, h, \alpha) = \frac{\| \mathbf{w} \|^2}{2} - \sum_{i=1}^{N} \alpha_i \left\{ t_i(\mathbf{w}^T \mathbf{x}_i - h) - 1 \right\}$$

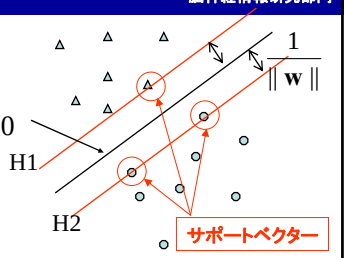
  - 停留点での条件

$$\frac{\partial L}{\partial h} = \sum_{i=1}^{N} \alpha_i t_i = 0$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^{N} \alpha_i t_i \mathbf{x}_i = 0$$

これをもとの式に代入 ＝＞ 双対問題

双対問題



- 双対問題
  - 目的関数:

$$L_D(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j \mathbf{x}_i^T \mathbf{x}_j \rightarrow \max$$

  - 制約条件:

$$\alpha_i \geq 0 \quad \text{for} \quad i = 1, \ldots, N, \qquad \sum_{i=1}^{N} \alpha_i t_i = 0$$

この解で、αが正となるデータ点を「サポートベクター」と呼ぶ。
これは、超平面H1あるいはH2の上にのる

最適識別関数

- 最適なパラメータ

$$\mathbf{w}^* = \sum_{i \in S} \alpha_i^* t_i \mathbf{x}_i \qquad h^* = \mathbf{w}^{*T} \mathbf{x}_s - t_s$$

ここで、Sはサポートベクターに対応するデータの添え字の集合

- 識別関数

$$y = \text{sgn}\left[\sum_{i \in S} \alpha_i^* t_i \mathbf{x}_i^T \mathbf{x} - h^* \right]$$

＝＞ サポートベクターのみで識別関数が構成される

SVMのまとめ

- 「マージン最大化」という基準から自動的に識別平面付近の少数の訓練サンプルのみが選択された

- その結果として、未学習データに対してもある程度良い識別性能が維持できる
- マージン最大化基準による、訓練サンプルの選択による、モデルの自由度の抑制

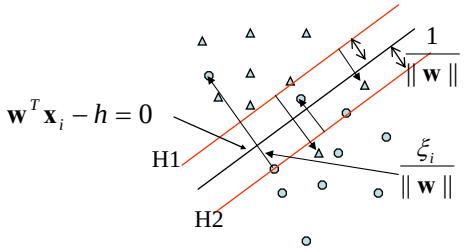
ソフトマージン

- サポートベクターマシン
  - 訓練サンプルが線形分離可能な場合の議論
- 線形分離可能で無い場合は？
  - 実際のパターン認識問題では、線形分離可能な場合は稀
  - 多少の誤識別を許すように制約を緩める
    ＝＞「ソフトマージン」

ソフトマージン

- ソフトマージン
  - マージンを最大としながら、幾つかのサンプルが超平面を越えて反対側に入ってしまうことを許す
- ペナルティ
  - 反対側にどれくらい入り込んだのかの距離の和

$$\sum_{i=1}^{N} \frac{\xi_i}{\| \mathbf{w} \|}$$



最適化問題(ソフトマージン)

- 目的関数:

$$L(\mathbf{w}, \xi) = \frac{1}{2} \| \mathbf{w} \|^2 + \gamma \sum_{i=1}^{N} \xi_i$$

- 制約条件:

$$\xi_i \geq 0,$$
$$t_i(\mathbf{w}^T\mathbf{x} - h) \geq 1 - \xi_i \quad \text{for} \quad i = 1, \ldots, N$$

双対問題（ソフトマージン）

- 目的関数：

$$L_D(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum \alpha_i \alpha_j t_i t_j \mathbf{x}_i^T \mathbf{x}_j$$

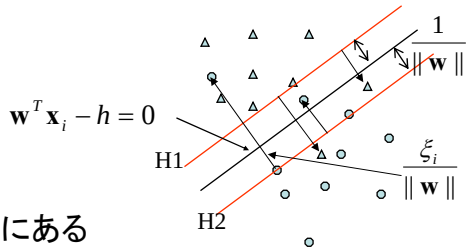
- 制約条件：

$$\sum_{i=1}^{N} \alpha_i t_i = 0,$$

$$0 \le \alpha_i \le \gamma \quad \text{for} \quad i = 1, \ldots, N$$

最適解（ソフトマージン）

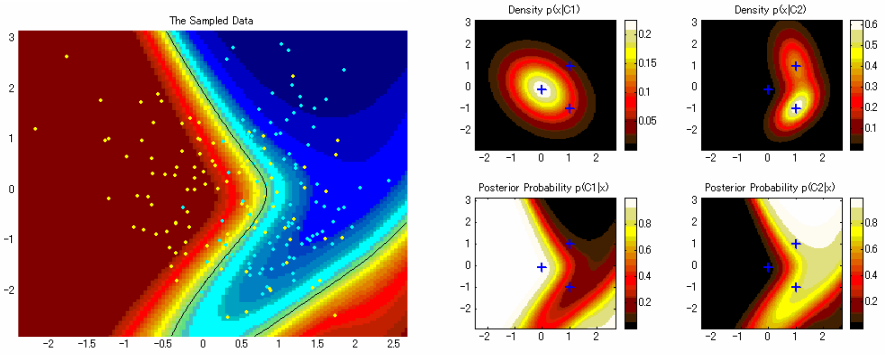
- ケース1：$\alpha_i^* = 0$
  - 正しく識別される
- ケース2：$0 < \alpha_i^* < \gamma$
  - ちょうど超平面H1かH2上にある
  - サポートベクター
- ケース3：$\alpha_i^* = \gamma$
  - 正しく識別できない
  - サポートベクター



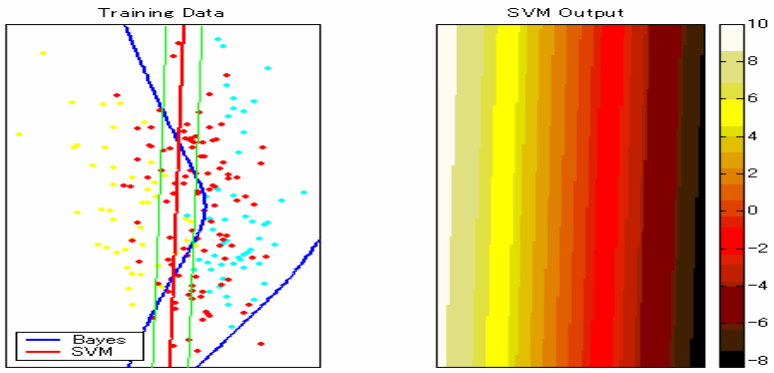
SVMによるパターン識別の例

- データ
  - Class 1: 2次元正規分布
  - Class 2: 2つの正規分布の混合分布

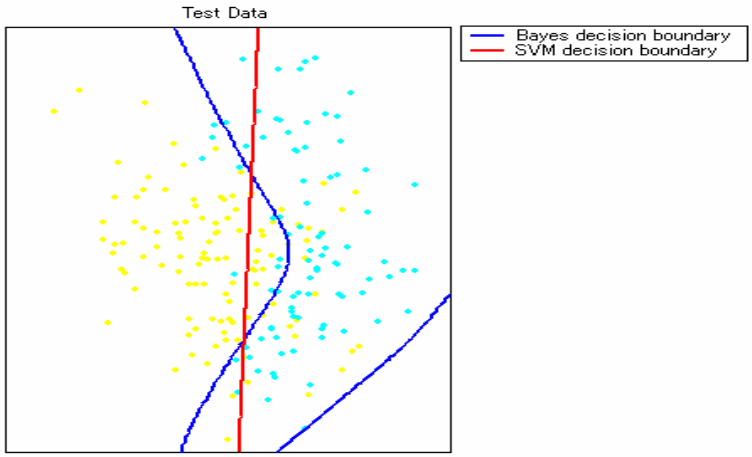
N1=100
N2=100



SVMの出力とサポートベクター

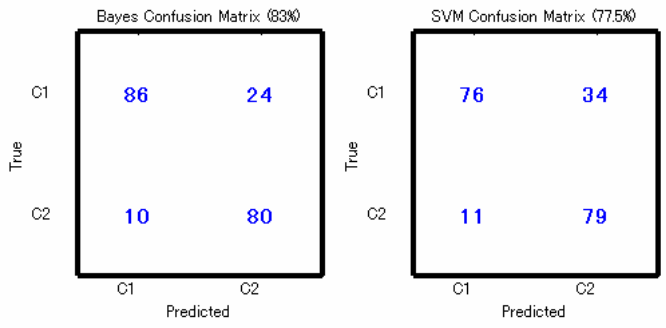


SVMによる識別境界



SVMによるテストサンプルの識別結果

新たに生成したテストサンプル(N=200)の識別



識別のための線形手法と汎化性

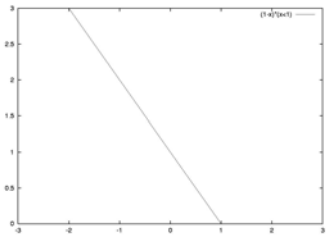
- 線形しきい素子を用いた識別器
  - 単純パーセプトロン
    - しきい値関数
    - 汎化性向上の工夫：マージン最大化⇔SVM
  - 重回帰
    - 線形関数
    - 汎化性向上の工夫：正則化ペナルティー⇔リッジ回帰
    - その他の汎化性向上の工夫：変数選択（Cross Validation, Resampling手法、情報量基準）
  - ロジスティック回帰
    - ロジット関数
    - 汎化性向上の工夫：正則化ペナルティー⇔Weight Decay

正則化法としてのSVM

- SVMの評価関数（ソフトマージン）

$$L(\mathbf{w},\xi)=\sum_{i=1}^{N}\xi_i+\lambda\sum_{j=1}^{M}w_j^2=\sum_{i=1}^{N}[1-t_i\eta_i]_+ +\lambda\sum_{j=1}^{M}w_j^2$$

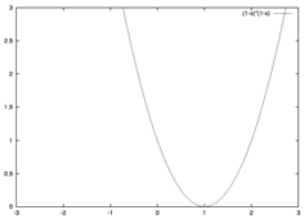
  - 第1項
    - モデルとデータの差異
      - 1より大きい場合は、0
      - 1より小さいと次第に大きな値
  - 第2項
    - パラメータの大きさに対するペナルティ



リッジ回帰

- リッジ回帰の評価関数

$$Q = \sum_{i=1}^{N} (1 - t_i \eta_i)^2 + \lambda \sum_{j=1}^{M} w_j^2$$

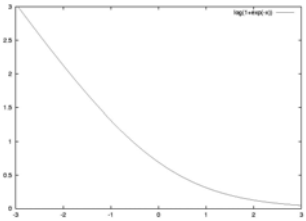


  - 第1項
    - モデルとデータの差異
      - 1からのズレで評価
      - 1以上になるような場合（正しく識別される）場合も大きなペナルティを与えてしまう
  - 第2項
    - パラメータの大きさに対するペナルティ

ロジスティック回帰

- ロジスティック回帰の評価関数（Weight Decay）

$$Q = \sum_{i=1}^{N} \log(1 + \exp(t_i \eta_i)) + \lambda \sum_{j=1}^{M} w_j^2$$



  - 第1項
    - モデルとデータとの差異
      - SVMの第1項と似ている
      - 1で不連続でなく、連続
      - 1より大きくなる（正しく識別される）サンプルには小さいペナルティ
  - 第2項
    - パラメータの大きさに対するペナルティ

評価関数の比較

- SVM（ソフトマージン）

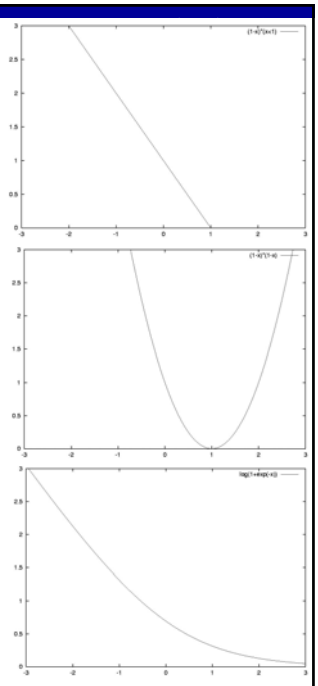
$$L(\mathbf{w},\xi) = \sum_{i=1}^{N} [1 - t_i \eta_i]_+ + \lambda \sum_{j=1}^{M} w_j^2$$

- リッジ回帰

$$Q = \sum_{i=1}^{N} (1 - t_i \eta_i)^2 + \lambda \sum_{j=1}^{M} w_j^2$$

- ロジスティック回帰（Weight Decay）

$$Q = \sum_{i=1}^{N} \log(1 + \exp(t_i \eta_i)) + \lambda \sum_{j=1}^{M} w_j^2$$



マージン最大化、リッジ回帰、Weight Decay

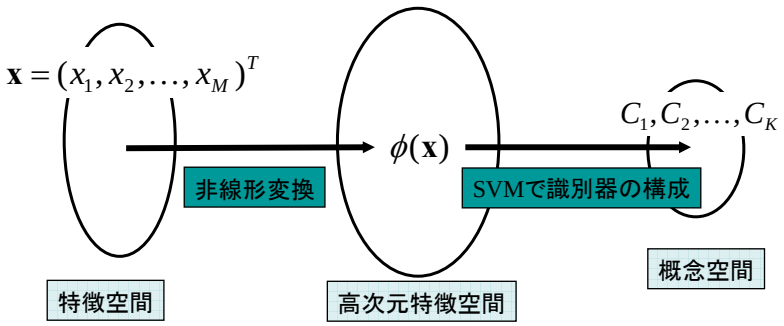
- 単純パーセプトロンタイプの識別器の学習における汎化性能の向上のための工夫
  - 重みが大きくなり過ぎない（不要な重みをなくす）
- SVM（ソフトマージン）、リッジ回帰、ロジット回帰（Weight Decay)の比較
  - 汎化性能の向上の工夫は同じ
  - モデルとデータとの差異の評価関数が異なる

特徴の高次元化

- 特徴の高次元化
  - 線形分離可能でない場合に対応するため、xを非線形変換により高次元の空間に写像して、その空間で線形の識別をする
    - 線形分離可能性は、訓練サンプル数が大きくなるほど難しく、次元が大きいほどやさしくなる。
    - 次元がサンプル数＋１以上であり、パターンが一般の位置にあれば、どんなラベル付けに対しても線形分離可能
- 高次元化の課題
  - 次元の呪い
    - 次元の増加とともに汎化能力が落ちてしまう
  - 計算量
    - 難しい問題を線形分離可能にするためには、訓練サンプル数と同程度の次元に射影する必要がある

非線形変換による高次元化

- 高次元特徴空間
  - 非線形写像により、高次元の特徴へ変換
    - 例：入力特徴を2次の多項式に変換



カーネルトリック

- 高次元特徴を用いたSVM
  - 目的関数や識別関数が入力パターンの内積のみに依存

  ＝＞内積が計算できれば、最適な識別関数を構成できる
- 内積

$$\phi(\mathbf{x}_1)^T \phi(\mathbf{x}_2) = K(\mathbf{x}_1, \mathbf{x}_2)$$

のように、入力特徴だけから簡単に計算できるなら、SVMの最適化問題や識別関数における内積をKで置き換え、線形分離可能な識別関数を得ることができる
- カーネルトリック
  - 高次元に写像しながら、実際には写像された空間での特徴の計算を避けて、カーネルの計算のみで最適な識別関数を構成するテクニック

カーネルSVM

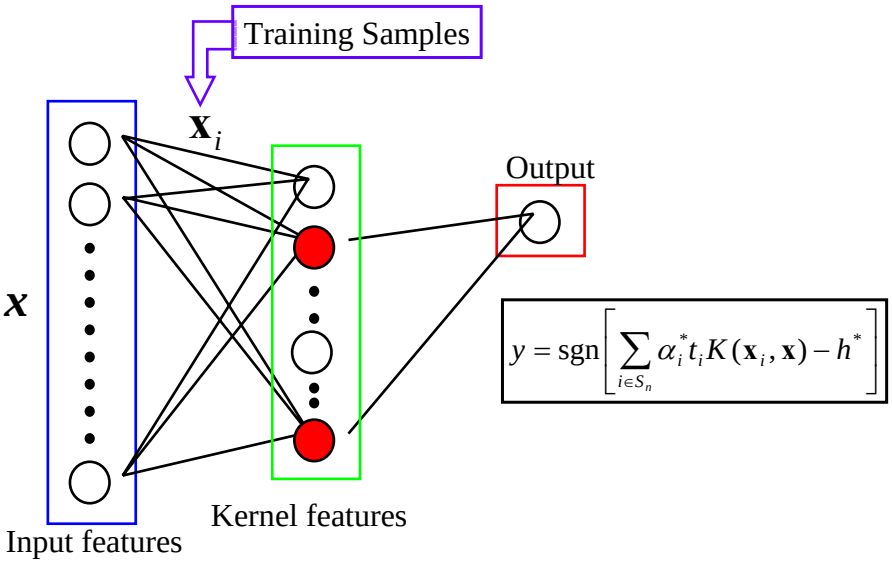
- 目的関数

$$L_D(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{N} \alpha_i \alpha_j t_i t_j K(\mathbf{x}_i, \mathbf{x}_j)$$

- 識別関数

$$y = \mathrm{sgn}\left[\sum_{i \in S_n} \alpha_i^* t_i K(\mathbf{x}_i, \mathbf{x}) - h^* \right]$$

線形SVMによるカーネルSVMの実現



標準的なカーネルの例

$$K(x, y) = \tanh(a < x, y > -b),$$

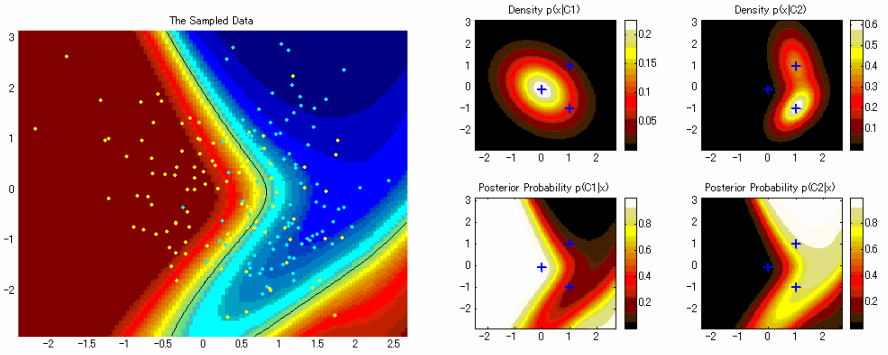
$$K(x, y) = \exp\left(-\frac{\| x - y \|^2}{2\sigma^2}\right),$$

$$K(x, y) = (< x, y > +1)^p$$

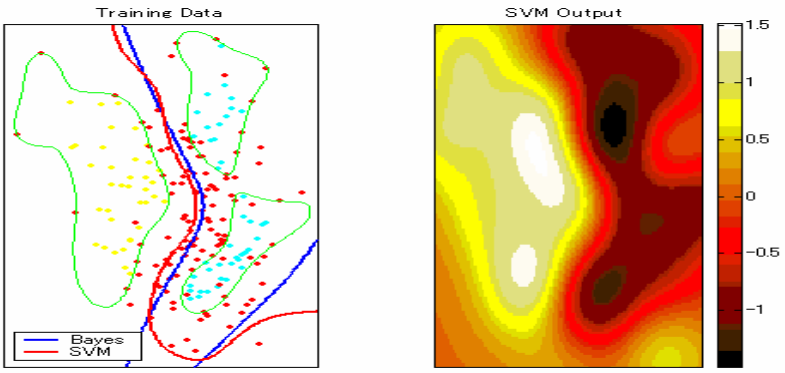
SVMによるパターン識別の例

- データ
  - Class 1: 2次元正規分布
  - Class 2: 2つの正規分布の混合分布

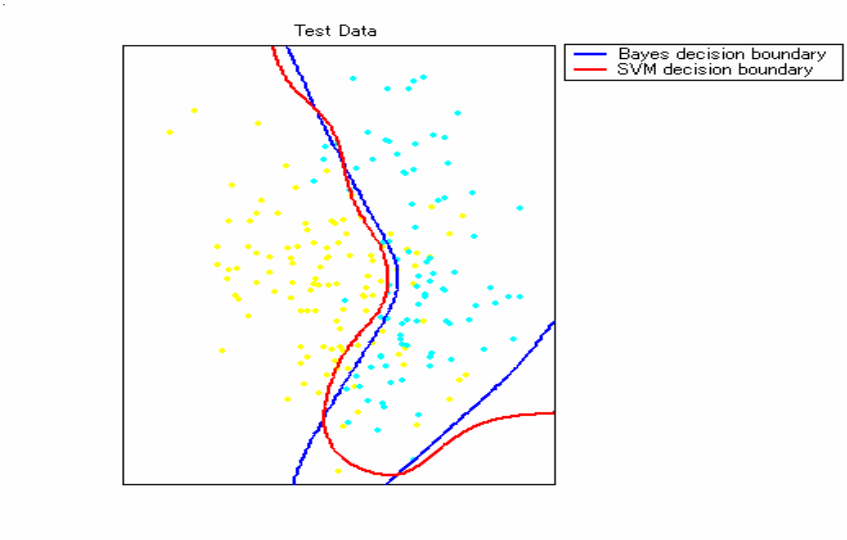
N1=100
N2=100



SVMの出力とサポートベクター

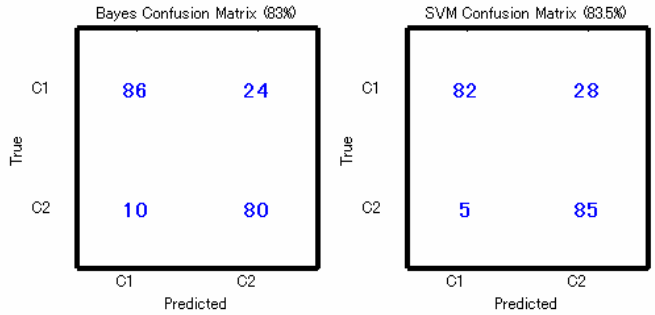


SVMによる識別境界

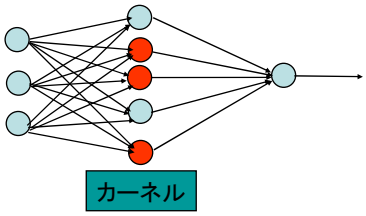


SVMによるテストサンプルの識別結果

新たに生成したテストサンプル(N=200)の識別



多層パーセプトロンとの関係



- 構造
  - シグモイドカーネル　＝＞　3層の多層パーセプトロン
  - ガウスカーネル　＝＞　RBFネットワーク
- 違い
  - 前段の入力層から中間層への結合荷重は固定
  - 中間層のユニット数が非常に多い（訓練サンプル数と同じ）
    ＜＝　マージン最大化により、ユニット数を削減

Chamfer Distanceに基づく カーネルを用いた歩行者検出

サポートベクターマシンでの
非標準カーネルの利用

画像間の距離(Chamfer Distance)

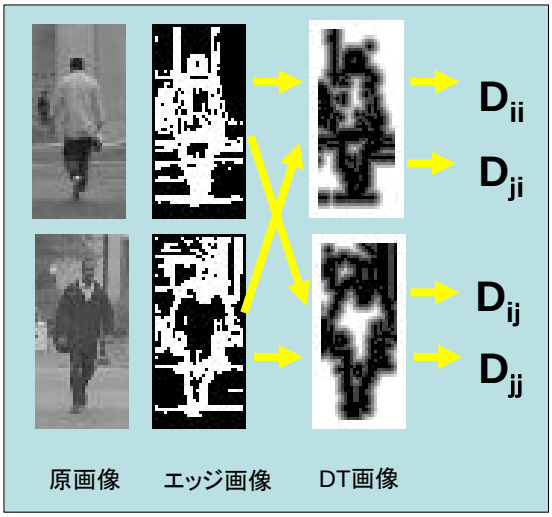
Chamfer Distance

$$D_{chamfer}(\mathbf{T},\mathbf{I}) = \frac{1}{|\mathbf{T}|}\sum_{t\in\mathbf{T}} d_{\mathbf{I}}(t)$$

Distance Transform (DT) 画像とのマッチングにより、Hausdorff distanceの近似が得られる

カーネル

$$K_{ij} = \exp\left\{-\frac{D_{chamfer}(i,j)}{2\sigma^2}\right\}$$



非標準カーネルを用いるSVMの線形SVMによる実現法

線形SVM

$$y = \mathrm{sgn}\left[\sum_{i\in S_n} \alpha_i^* t_i \mathbf{x}_i^T \mathbf{x} - h^*\right]$$

カーネルSVM

$$y = \mathrm{sgn}\left[\sum_{i\in S_n} \alpha_i^* t_i K(\mathbf{x}_i,\mathbf{x}) - h^*\right]$$

カーネル行列

$$K = \left[K(\mathbf{x}_i,\mathbf{x}_j)\right]$$

カーネルPCA写像

$$\mathbf{g}(\mathbf{x}) = K^{1/2}\mathbf{k}(\mathbf{x})$$

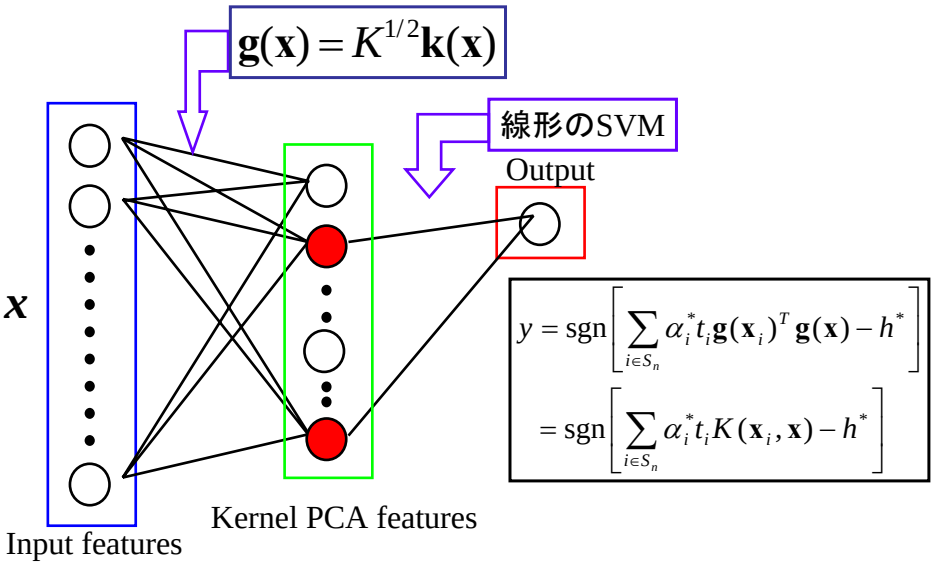
おなじ

既存の線形SVMのプログラムを利用して、任意のカーネルを用いたカーネルSVMを実現可能

g(x) を新特徴ベクトルとした線形SVM

$$y = \mathrm{sgn}\left[\sum_{i\in S_n} \alpha_i^* t_i \mathbf{g}(\mathbf{x}_i)^T \mathbf{g}(\mathbf{x}) - h^*\right] = \mathrm{sgn}\left[\sum_{i\in S_n} \alpha_i^* t_i K(\mathbf{x}_i,\mathbf{x}) - h^*\right]$$

線形SVMによるカーネルSVMの実現



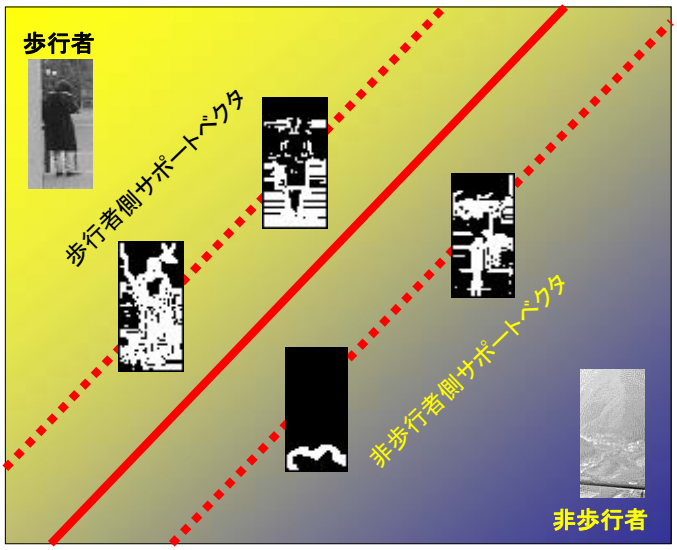
実験に用いた歩行者画像

歩行者画像 **非歩行者画像**

- 歩行者：MIT CBCL画像データベース 924枚
- 非歩行者： ランダムに選択した画像 2700枚
- Kernel化のためには画像間の距離（類似性）を定義する必要がある

Chamfer Kernel SVMによる歩行者検出

- 総認識率 90.5%
- False Positive率 10.8%



カーネル判別分析

- カーネル特徴ベクトル

$$\mathbf{k}(\mathbf{x}) = (K(\mathbf{x}_1, \mathbf{x}), \ldots, K(\mathbf{x}_N, \mathbf{x}))^T$$

- 判別写像

$$\mathbf{y} = A^T \mathbf{k}(\mathbf{x})$$

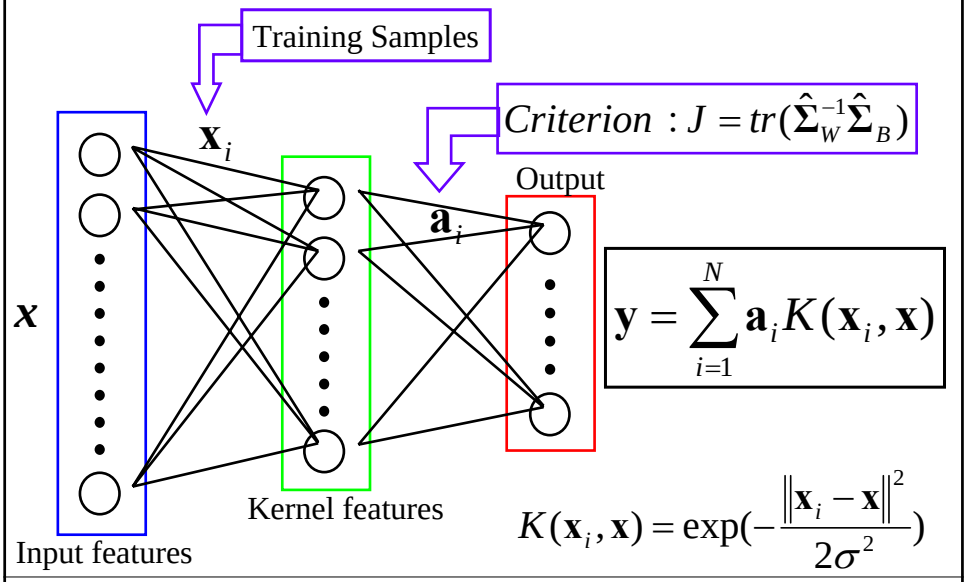
- 固有値問題

$$\Sigma_B^{(K)} A = \Sigma_W^{(K)} A^T \Lambda \quad (A^T \Sigma_W^{(K)} A = I)$$

- 汎化性能向上のための工夫（正則化）

$$\tilde{\Sigma}_W^{(K)} = \Sigma_W^{(K)} + \beta I$$

カーネル判別分析



カーネル判別分析による顔検出

顔と顔以外の対象との識別のために
判別基準を工夫

顔検出のための判別基準の工夫

顔検出＝顔と顔以外の2クラスの識別問題

- 顔以外のサンプルは様々な特性の画像が含まれるので、1つのクラスとして扱うのは難しい
- 2クラスの判別分析で構成される判別空間は1次元となる

顔と顔以外の対象の判別基準

- 顔クラスの共分散を最小
- 顔クラスの中心と顔以外の各サンプルの共分散を最大

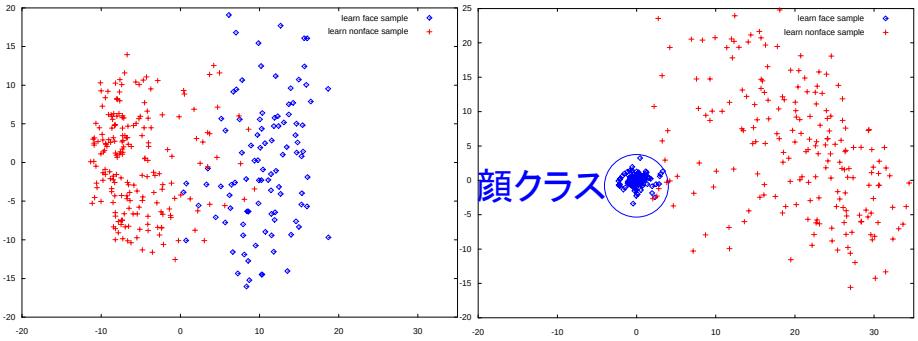
- 顔：1つのクラス
- 顔以外：各々独立したクラス

高次元の判別空間が構成できる

顔検出のための判別空間

元の特徴空間

顔検出のために構成された判別空間



顔と顔以外の画像データベース

- Webから集めた多数の顔と顔以外の画像
- MIT、CMU顔画像データベース

- 学習用データセット
  - カーネル判別分析の学習に使用
- パラメータ決定用データセット
  - 閾値、カーネルの幅、正則化パラメータの決定に使用
- 評価用データセット
  - 手法の評価に使用

評価実験結果

	全ての画像	顔画像	顔以外の画像
2クラスのカーネル判別分析 ($\alpha=0.0, \sigma=1.08$)	98.3% (1303/1325)	98.8% (321/325)	98.2% (982/1000)
提案手法 ($\alpha=0.0, \sigma=1.08$)	98.7% (1308/1325)	96.0% (312/325)	99.6% (996/1000)
提案手法 ($\alpha=0.0002, \sigma=1.08$)	99.2% (1314/1325)	98.2% (319/325)	99.5% (995/1000)
サポートベクターマシン ($\sigma=1.08$)	98.3% (1302/1325)	99.4% (323/325)	97.9% (979/1000)

指文字認識への応用

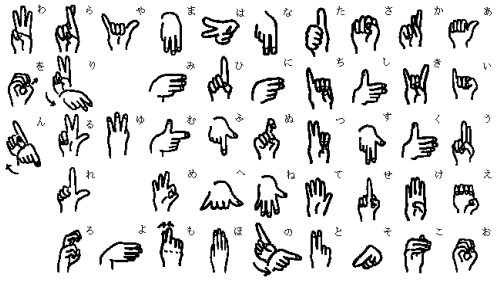
訓練サンプルのクラスタリングによる
カーネル特徴ベクトルの次元圧縮と汎化性

指文字

- 指文字とは片手の5本の指の曲げ伸ばしにより「あいうえお‥」を表現したもの。
- 手話中に固有名詞を伝えるときなどに使われる。

静文字: 41文字

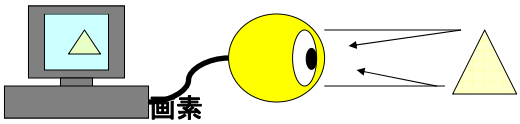
動文字: 「の、も、り、を、ん」 5文字



指文字表
Manual Japanese Syllabary "YUBI-MOJI"

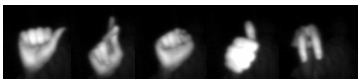
モーションプロセッサ

- 赤外線の反射光により対象物を画像として取り出すモーションプロセッサにより、指文字を画像として取り込む。



- 各画素が256階調のグレースケール

指文字の例



あ　か　さ　た　な

データ

静文字の41文字を入力データとする。
「あ」がクラス1、「い」がクラス2というように41クラスから構成。

→ 0　10　50　0　100　255　0　・・・・

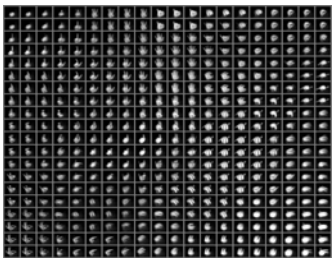
32×32＝1024個

- 学習用データ
  4人分の指文字画像492枚。
- 評価用データ
  学習用と同じ4人の指文字画像328枚。

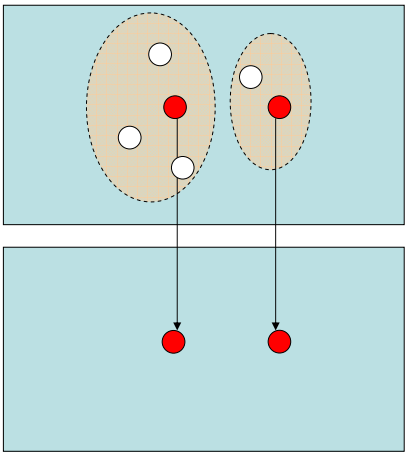
訓練サンプルのクラスタリングによるカーネル特徴の次元の圧縮

- K-means法や自己組織化マップを用いてデータをクラスタリングし、カーネル特徴の次元数を減らす。

類似度の大きいデータ同士を集めて、それぞれのクラスターの代表ベクトルを得る。

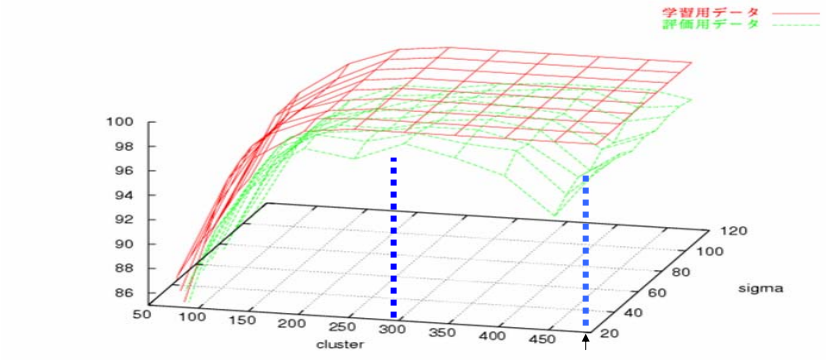


自己組織化マップの例



カーネル判別分析の結果

- K-means法におけるクラスタ数とカーネル特徴生成時のパラメタσを変化させ、判別分析を行う。



492 ＝クラスタリングなし

認識性能の比較

クラスタ数300

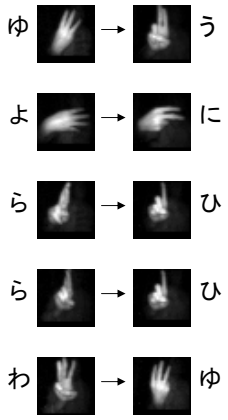
σ	学習データ	評価データ
25	100%	99.09%
45	100%	98.78%
65	100%	98.48%
85	100%	97.87%
105	100%	97.26%

クラスタリングなし

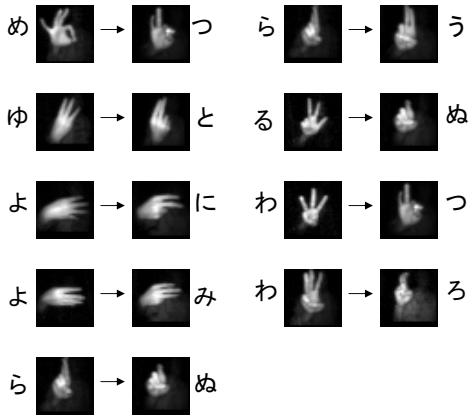
σ	学習データ	評価データ
25	100%	97.87%
45	100%	97.56%
65	100%	97.26%
85	100%	97.26%
105	100%	96.95%

誤識別

クラスタ数300（エラー5つ）



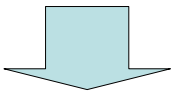
クラスタリングなし（エラー9つ）



ロジスティック回帰で推定した確率空間でのK-NN法による文字認識

K-最近傍法をベースとした汎化性能の高い識別器

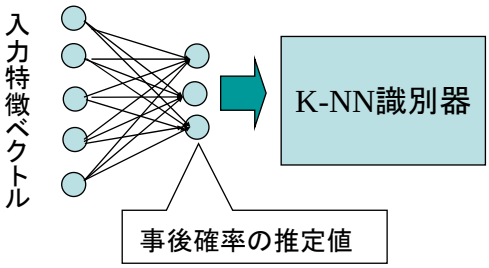
- K-最近傍法
  - 充分な訓練サンプルが与えられれば、未学習データに対する識別誤差がベイズ誤識別率の2倍を超えない
  - カーネル特徴のように入力特徴ベクトルの次元が高い場合にはこの性能は保証されないし、識別のために膨大な計算量が必要



高次元の特徴ベクトルから識別のための本質的な特徴を抽出し、それをK-NN法の入力とする

多項ロジットモデルを用いた次元圧縮

- 多項ロジットモデル
  - 多クラスパターンの識別のための最も簡単なニューラルネットモデルのひとつ
  - 入力特徴から事後確率を推定
  - 汎化性能を向上させるためには、工夫が必要
  - 2クラスの場合（ロジスティック回帰）は、単純パーセプトロンやサポートベクタマシンとも関連

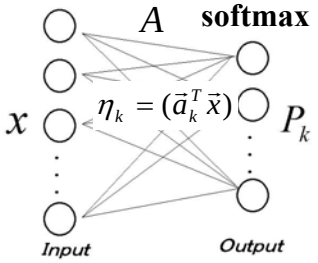


多項ロジスティック回帰モデル

$\vec{x}$ ： 入力ベクトル

$\vec{a}$ ： パラメータベクトル

$\eta_k = (\vec{a}_k^T \vec{x})$ の “softmax”



$$p_k = \frac{\exp(\eta_k)}{1+\sum_{m=1}^{K-1}\exp(\eta_k)} \quad , k=1,\ldots,K\text{-}1$$

$$p_K = \frac{1}{1+\sum_{m=1}^{K-1}\exp(\eta_k)} \quad , \quad k=K$$

多項ロジスティック回帰での学習

- 尤度

$$P(\vec{t} \mid \vec{x}; A) = \prod_{k=1}^{K} p_k^{t_k}$$

- 対数尤度

$$l(\vec{t} \mid \vec{x}; A) = \sum_{k=1}^{K-1} t_k \eta_k - \log(1 + \sum_{m=1}^{K-1} \exp(\eta_m))$$

- 学習則

$$\vec{a}_k \Leftarrow \vec{a}_k + \alpha(t_k - p_k)\vec{x}$$

α は学習係数

多項ロジスティック回帰＋K-NN法の特徴

- Kernel特徴ベクトル + 多項ロジスティック回帰
  - 多クラスに対応
  - SVMと異なり境界面の内側のデータも評価に加える
  - ベクトルの次元圧縮:「次元の呪い」からの解放
- Logit出力（確率空間）への重みつきK-NN法の適用
  - 各クラスの確率密度分布の識別境界面を明確に定めるのではなく、データの分布密度に応じて判定。
- “Kernel特徴複合ベクトル” の導入
  - 識別に有用そうな次元はどんどん加えられる
  - 多項ロジスティック回帰により識別に有用な特徴次元に重みをかけつつ、次元圧縮が達成される

汎化性向上のための3つの工夫

- 工夫1：Weight Decay
  - 学習の評価基準に照らして寄与の少ない結合加重が0　に近づくような項を更新式に加える
- 工夫2：人工的な変動の付加
  - 　　に一様乱数を付加

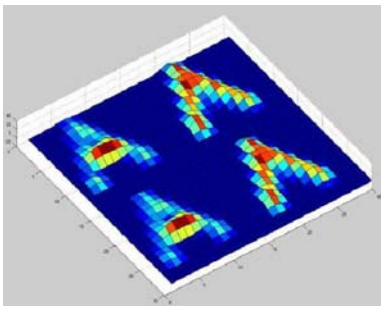
$$\eta_k = (\vec{a}_k^T \vec{x})$$

- 工夫3：エントロピーに基づく重み付き学習
  - 識別クラスが不明瞭なデータを優先して学習

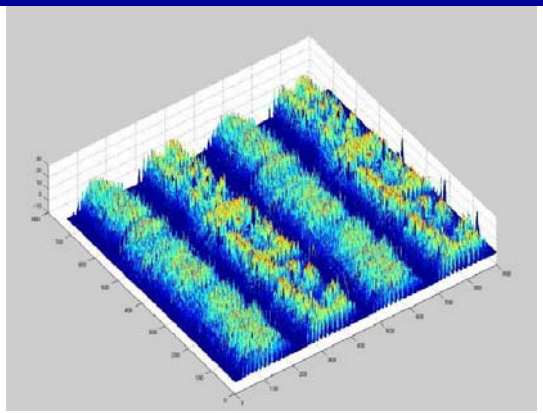
＝＞ 識別境界面付近のデータの識別クラスがより明確になるように学習

学習・評価用サンプル (ETL6を利用)

学習に用いたデータ例“A”



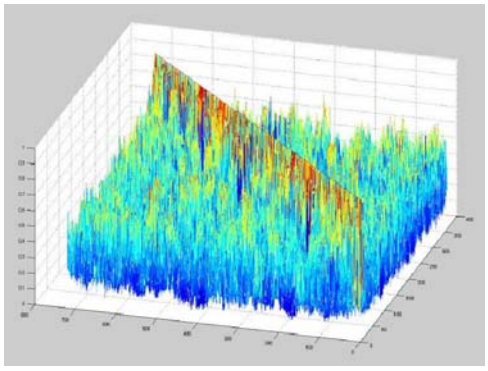
4方向特徴



基本特徴ベクトル（900次元特徴ベクトル）

4方向特徴の画像（30x30ピクセル）を900x1サイズに変換
⇒ 900次元(=4x15x15)のベクトル

サンプル数：学習用、評価用にそれぞれ
（1）36クラス　7200個，（2）82クラス　16400個



$$\vec{y}_i = (y_{i1}, \ldots, y_{ik})^T,$$

$$y_{ik} = \exp\left(\frac{-\|x_k - x_i\|^2}{2 \times \sigma}\right)$$

Kernel関数 ⇒ Gauss関数

Kernel特徴ベクトル（36クラスのみ）

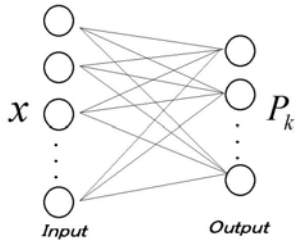
36クラスの各クラスから任意に100個のデータを抽出。
これを「基準」にKernel特徴ベクトルを構成。

Kernel特徴ベクトルと元のベクトルを結合

Kernel特徴複合ベクトル

各文字4500次元 (= 900 + 3600)

MLM 出力Pへの 重みつきK−最近傍法の適用



確率出力Pをベクトルと見なし
これにK−最近傍法を適用

MLMにより入力ベクトルの次元数は

36クラス: 4500次元 ⇒ 36次元

と大幅に削減される。（「次元の呪い」からの解放）

ベクトル間の距離・重みの定義
（他の定義も考え得る）

$$d_{i,j} = \frac{1}{\left\| \vec{p}_i - \vec{p}_j \right\|}$$

文字認識実験結果

- 基本特徴ベクトル + 標準的MLM
  (1) 36クラス：94.86%　　(2) 82クラス：92.97%
- Kernel特徴複合ベクトル + K-NN
  (1) 36クラス：96.25%
- Kernel特徴複合ベクトル + 標準的MLM
  (1) 36クラス：97.89%
- Kernel特徴複合ベクトル + 標準的MLM + K-NN
  (1) 36クラス：98.93%
- Kernel特徴複合ベクトル + 汎化性向上の工夫ありMLM + K-NN
- 基本特徴ベクトル（900次元）
  (1) 36クラス：未学習7200個 => 99.99%　(2) 82クラス：未学習16400個 => 99.90%
- Kernel特徴複合ベクトル（4500次元）
  (1) 36クラス：未学習3600個 => 100.0%

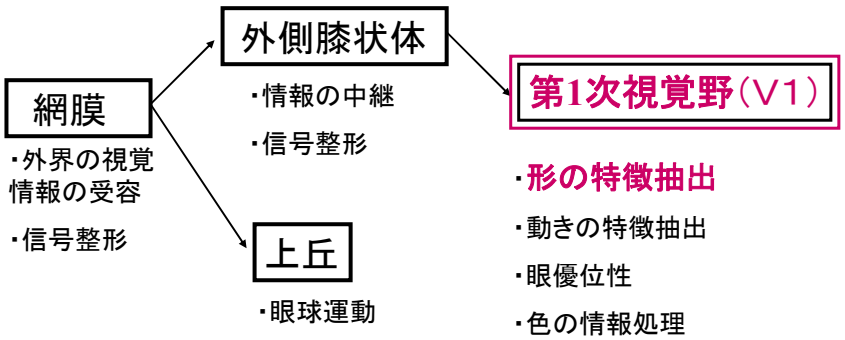
安田ら：「12方向の相補的特徴場＋摂動」相関法
（特徴ベクトルの構成に工夫、識別器はシンプル）
同じETL6の36クラス ⇒ 99%台半ば（安田2001）
こちらに比べ判定時の処理量は大幅に少ない

画像認識への応用

脳科学の進展

- 脳の視覚情報処理に関する知見
  - 網膜レベルからすでに情報が分化
    - 空間的な位置関係や動きに関する知覚---大脳皮質の視覚野から上に向かい頭頂連合野に至る経路
    - 視野内の物体が何かのパターン認識---視覚野から下の側頭連合野に至る経路
  - 視覚情報処理のための多くの専門分化された領野が存在
  - コラム構造
    - 眼優位性コラム（第1次視覚野V1）---左右どちらの芽からの情報を受け取るかでコラム構造を形成
    - 方位選択性コラム（第1次視覚野V1） --- 線分の方向に選択的に反応する細胞がコラム構造を形成
    - 三次元物体回転（TE野）---似た図形特徴に反応する細胞が三次元物体回転に対する見えの変化と整合性を持つような順序でコラム構造を形成
    - 運動方向性コラム（MT野）---視野内の刺激の方向に選択的に反応する細胞がコラム構造を形成

初期視覚情報処理

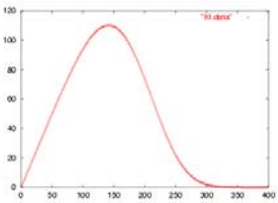


網膜

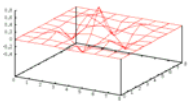
- 眼底に貼り付いている透明な神経組織
- 外界の視覚情報を受け取り、局所的な情報処理の結果を神経パルス列に符号化して、視覚中枢に送り込む
- 視細胞(photoreceptor)、水平細胞(horizontal cell)、アマクリン細胞(amacrine cell)、神経節細胞(ganglion cell)が整然と並んだ層構造

網膜での情報処理

- 自然画の局所的な自己相関
  - 風景や顔などの人工物を含まない自然画像の局所的な自己相関のパワースペクトルは空間周波数の2乗に反比例する(Field 1987)



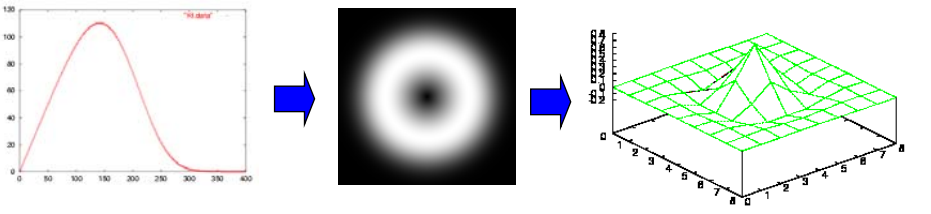
- 神経節細胞の出力のパワースペクトル
  - 低周波では、平坦(コンスタント)(Atick等 1992)
    - 自己相関を空間的に無相関化していることに対応
    - 入力情報から空間的な冗長性を取り除く処理(whitening)
  - 高周波では、高周波成分を抑制
    - Whiteningによりノイズが増幅されることを防ぐ働き

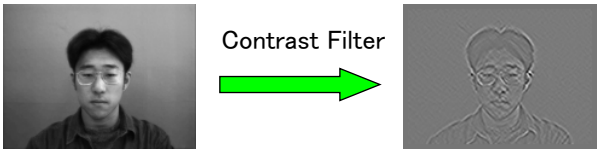


コントラストフィルタ

- 網膜のガングリオン細胞の受容野に類似 [Atick92,Olshausen97]

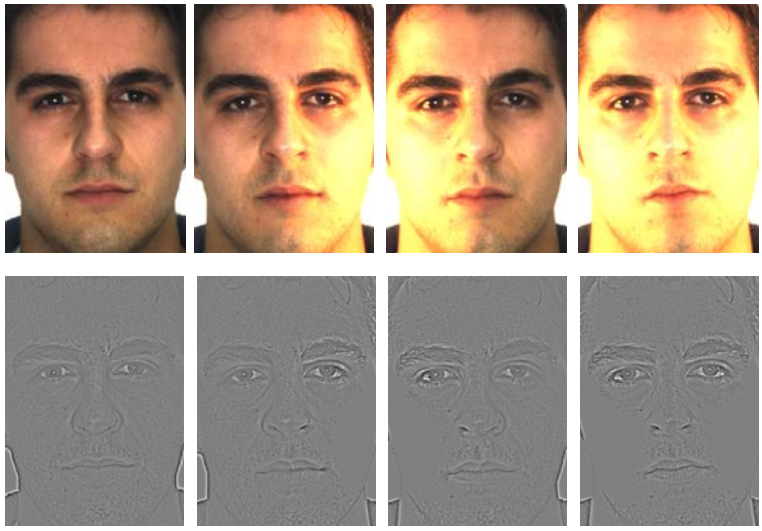
$$K(f) = W(f)L(f) = f \exp\left\{-\left(f/f_0\right)^4\right\}$$



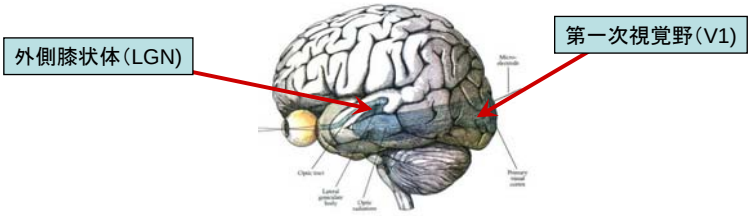


入力画像　　コントラスト画像

コントラストフィルタの明るさの変化に対する頑健性

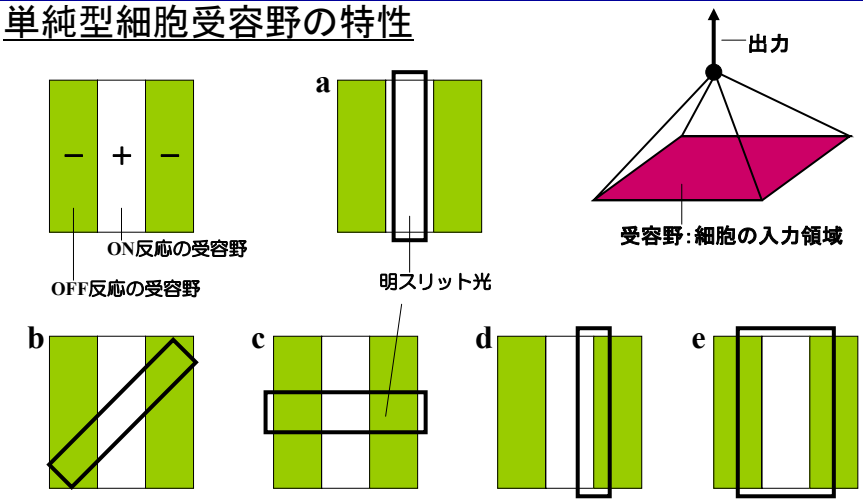


第一次視覚野(V1)



- 6層構造をした後頭部にある大脳皮質の一部で、外側膝状体(LGN)からの入力は4C層に入る
- 各ニューロンは受容野により規定される方向を持った直線状のコントラストに対して強い反応を示す(単純型細胞)
- 光刺激の位置が方位に垂直方向に多少ずれても反応の強さが変化しないニューロンも存在する(複雑型細胞)

単純型細胞受容野の特性



受容野の3特性

方位選択性（b、c）、局所性（d）、幅選択性（e）

第一次視覚野での情報処理

- 情報抽出とスパース符号化
  - いくつかの基底ベクトルの線形結合により入力をなるべく近似し、しかも、その結合係数がなるべくスパースになるような基準で基底ベクトルを求めると、第一次視覚野の単純型細胞の特性と似た特徴が得られる(Olshausen & Field, 1996)
- 独立成分の抽出
  - 独立成分分析(ICA)を用いて、Olshausen & Fieldの結果と同様な結果が得られる(Bell & Sejnowski, 1997)

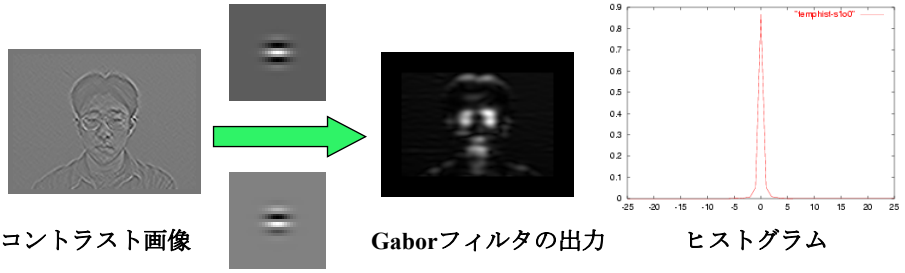
なるべく多くの情報を取り込み、しかも取り込んだ情報に含まれる冗長性をなるべく取り除くような情報処理を実現

これは、入力情報を取り込む機能を実現するための最も自然な動作原理

Gaborフィルタ

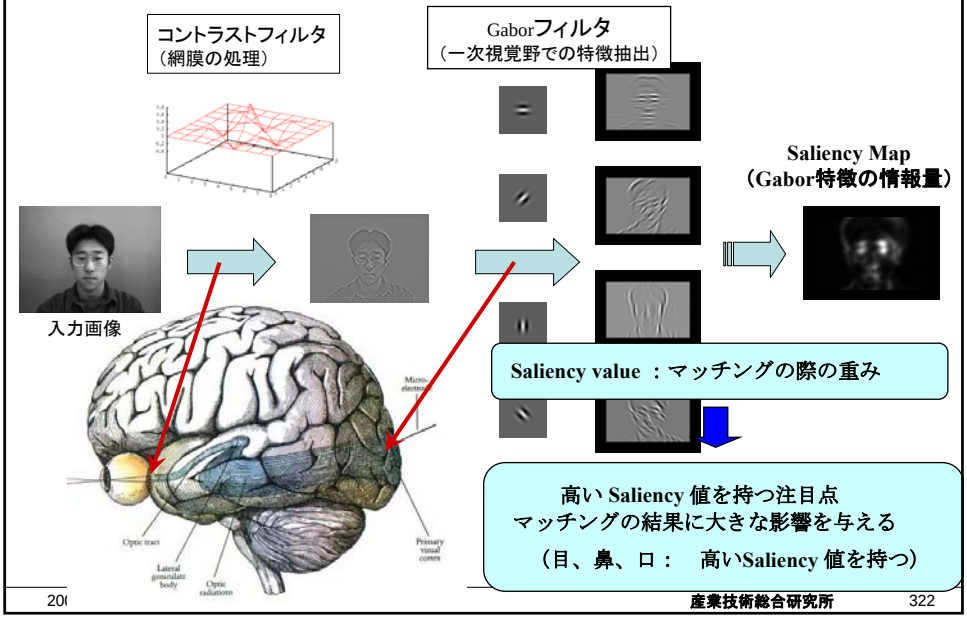
- V1野の単純型細胞の受容野特性に類似 [Jones87]
- 顔（対象）認識への有効性が報告されている [Malsburg93]
  - Sparse coding：鋭い選択性を持つ細胞集団の発火により情報を表現 [Olshausen96]
  - 自然画像のICA [Bell96] → Gabor-likeフィルタ
    - 各方位のGaborフィルタ：確率的独立性が高い

- 実験： 8方向のGaborフィルタ（9x9画素）を利用



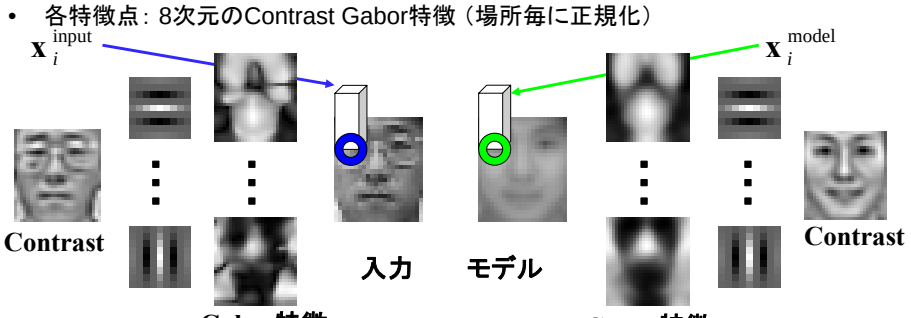
コントラスト画像　Gaborフィルタの出力　ヒストグラム

コントラスト + Gaborフィルタ



識別器

- 識別器　：モデルとのマッチング
- 各特徴点：8次元のContrast Gabor特徴（場所毎に正規化）

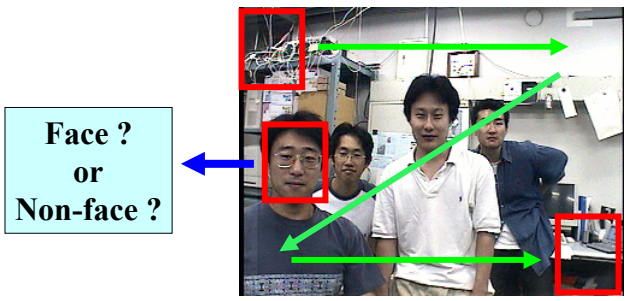


$$\text{Distance} = \sum_{i=1}^{HW} \text{Dist}(i) = \sum_{i=1}^{HW} \left\| \mathbf{x}_i^{model} - \mathbf{x}_i^{input} \right\|^2$$

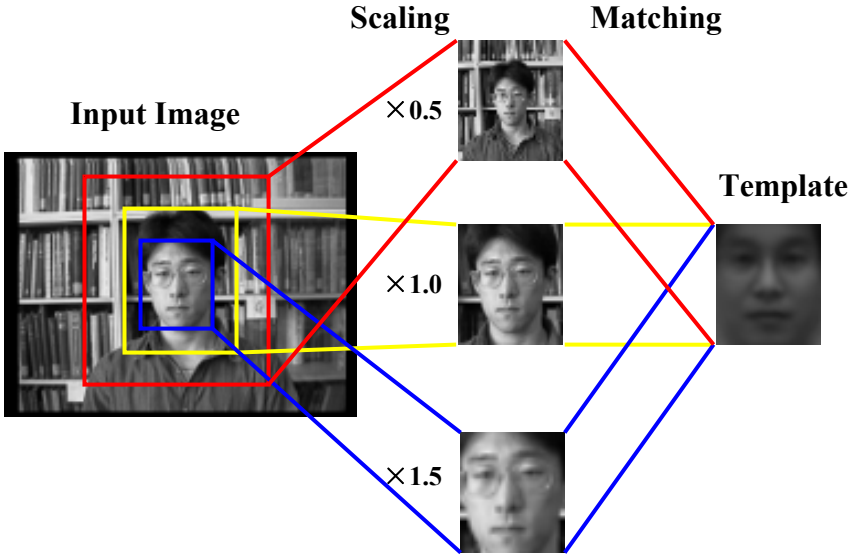
Distance $\leq \theta$ ➡ **Face**

$> \theta$ ➡ **Non-Face**

画像中の顔の検出



大きさの変化への対応



Examples of Face Detection 1

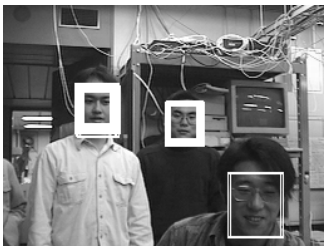
The kernel size of Gabor filter : 9x9 pixels

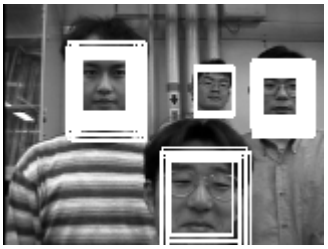
The size of model face : 31x26 pixels

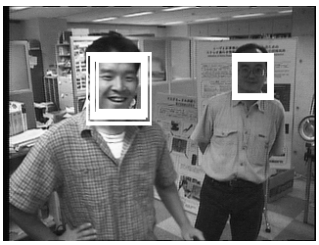
Model face : mean face of 20 persons

Model face

Saliency Map

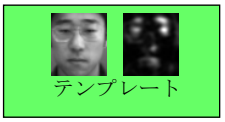






特定の人の顔をテンプレートとした顔検出


1996年に撮影


テンプレート

約200枚の顔画像に対して正しく検出できた

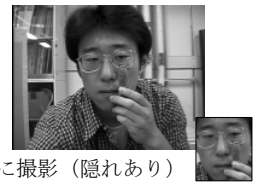

平均顔（検出された約200枚の顔画像から作成した）


1997年に撮影


1998年に撮影

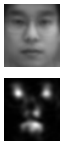

平均顔（相関マッチングで検出した顔画像で作成。顔の検出率31.7%）


1999年に撮影（暗い）


1999年に撮影（隠れあり）

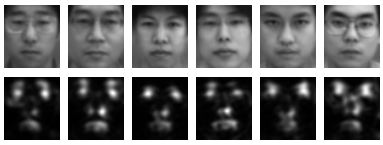
顔検出 ＋ 個人識別

顔検出
（多数の人の平均顔）



162枚の顔画像に対して、99.4%の検出＋個人識別率

個人識別
（個人の平均顔）



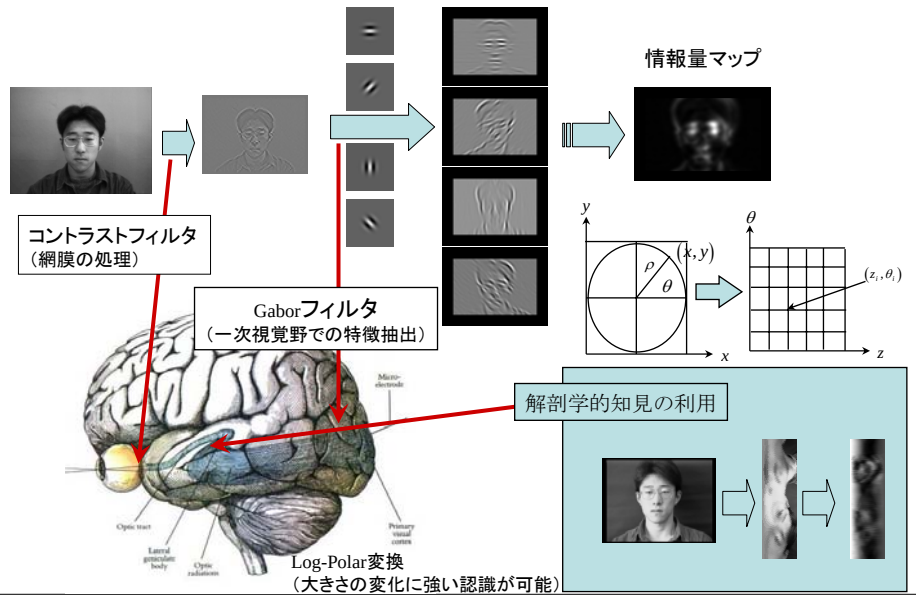

1996年に撮影


かなり暗い


めがねをかけた

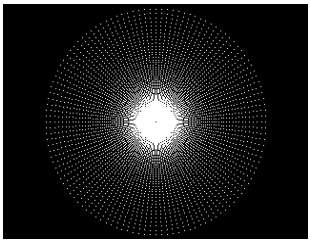
大きさの変化に強い顔認識

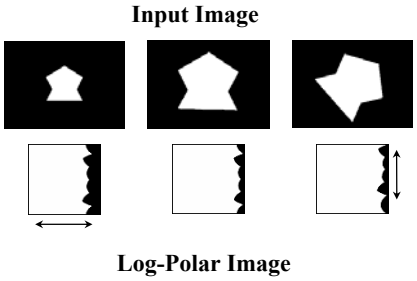
解剖学的知見の利用

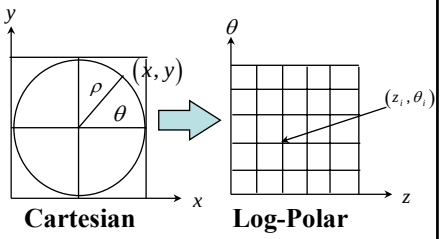


大きさの変化に影響を受けにくい顔認識

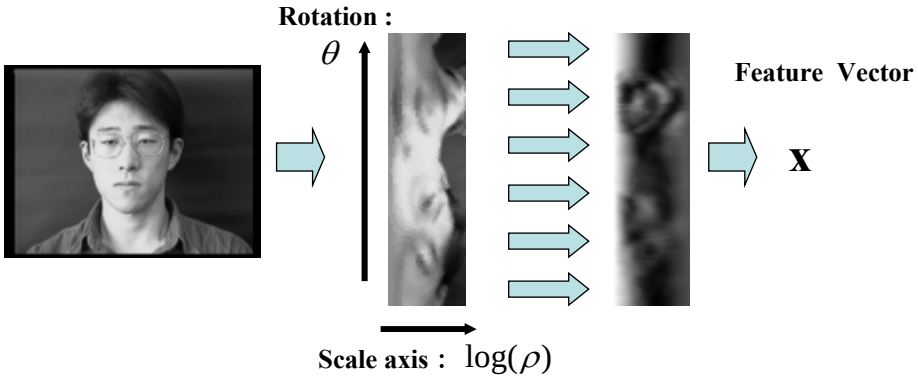
- 網膜の視細胞の密度は不均質
  - Log-Polarサンプリング
- Log-Polarサンプリングの性質
  - 中心が解像度が高く、周辺は低い
- Log-Polar画像







大きさ不変特徴



- Log-Polar画像の横軸方向の位置不変特徴（スペクトル特徴）を抽出
  - **Autocorrelation, Fourier power spectrum, and PARCOR features**

スペクトル特徴

- 自己相関特徴
  - $x(n)$ と $x(n+m)$ の自己相関

$$R(m) = \frac{1}{N}\sum_{n=0}^{N} x(n)x(n+m) \qquad \rho(m) = \frac{R(m)}{R(0)}$$

- フーリエパワースペクトル特徴

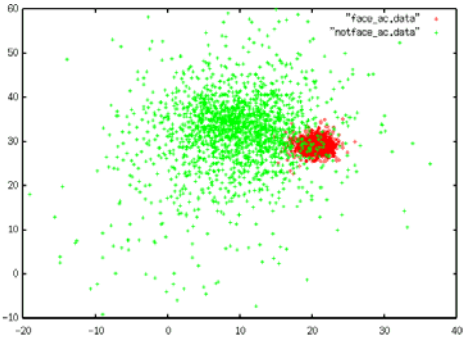
$$FP(k) = \sqrt{\{\frac{1}{N}\sum_{n=0}^{N} x(n)\exp(-2\pi\, j\frac{nk}{N})\}^2}$$

- PARCOR 特徴
  - 順方向の自己回帰モデルの予測誤差と逆方向の自己回帰モデルの予測誤差との間の相関係数

$$k_\tau = \frac{E[\varepsilon_f\ \varepsilon_b]}{\sqrt{E[\varepsilon_f]E\ [\varepsilon_b]}}$$

顔(face)と顔以外(not face)の識別

- 顔検出 : “face” and “not face” classification
- 識別空間の構成 :
  - the covariance of “face” class ⇨ Min
  - the covariance between “face” class and each “not face” samples ⇨ Max

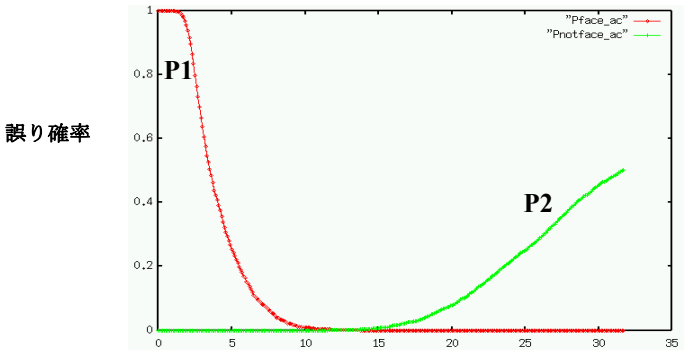


$tr(\Sigma_F^{-1}\Sigma_B)$ ⇨ Max

顔検出のためのしきい値の設定

– P1とP2の和が最小となるしきい値を選定

- P1: “face” を誤って顔でないと判定する確率
- P2: “not face” を誤って顔と判定する確率



顔検出実験

学習データ：７０名以上の人の3000枚以上の顔画像と1000枚以上の顔以外の画像

テストデータ：学習に含まれていない200枚の顔を含んだ画像

評価：顔の中心から５画素以内に顔があると検出できたものを正解とする

	認識率(%)
自己相関	95．0
パワースペクトル	97．5
PARCOR	84．0
HLAC	42．0


顔画像


顔以外の画像

顔識別実験の結果

学習データ：400枚（5人 x 20枚 x 2 scales x 2背景）

テストデータ：1200枚（5人 x 20枚 x 7 slcales x 2 背景）

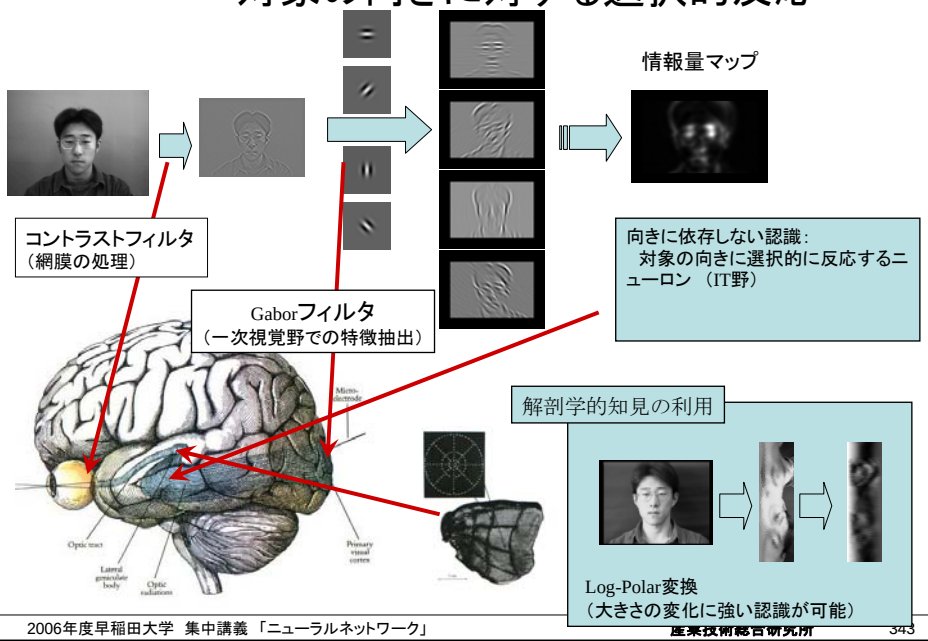
評価：顔の中心から５画素以内に顔があると検出できたものを正解とする

Log-Polar画像のサイズを変化させた場合の認識率

画像サイズ	30x30	60x30	90x30	120x30
自己相関	97.64	97.79	97.29	96.64
パワースペクトル	98.93	99.50	99.14	98.29
PARCOR	91.79	93.93	89.07	95.93
HLAC	82.21	77.36	82.79	85.93

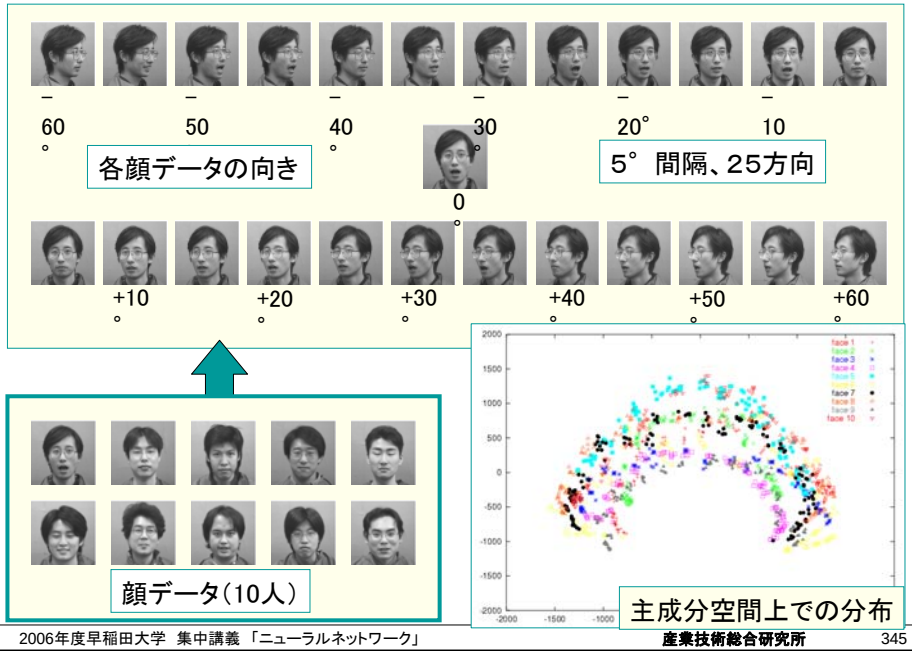
向きの変化に影響を受けない顔認識

対象の向きに対する選択的反応



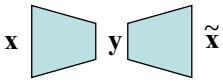
顔の向きに依存しない顔認識

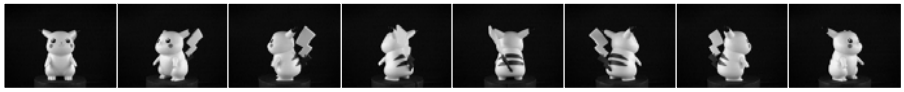
- 向きにより入力画像が大きく変化
  - ◆ 正面からの顔画像は、その人の横顔よりも他の人の正面顔に近い
  - ◆ 我々人間の視覚では異なる向きの対象を容易に認識可能
- 生体の視覚系
  - ◆ 3次元の対象を識別するように学習したサルのIT野では、対象の向きに選択的に反応するニューロンがあり、その選択性は系統的[Pauls96]
  - ◆ 顔認識タスクでも、IT野で顔の向きに選択的に反応するニューロンがある[Perrett89,Hasselmo89]
- 工学的模倣
  - ◆ RBFネットワークを用いて、少数の代表的な見えの補間で任意の向きからの見えが表現可能[Poggio90]
  - ◆ 複数の非線形のautoencodersを統合して任意の見えの顔画像が表現できる[Ando99]
- 提案手法
  - ◆ 向きに選択的に反応する複数の識別器(Classifiers)を gating ネットワークにより入力画像の向きに応じて適切に選択



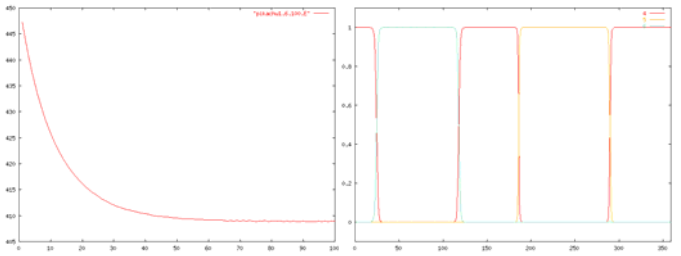
SoftMax競合の砂時計型ニューラルネット

- 代表的な見えの自己組織化
  - 中間層: SoftMax





入力画像の例（360度を1度刻みで撮影）

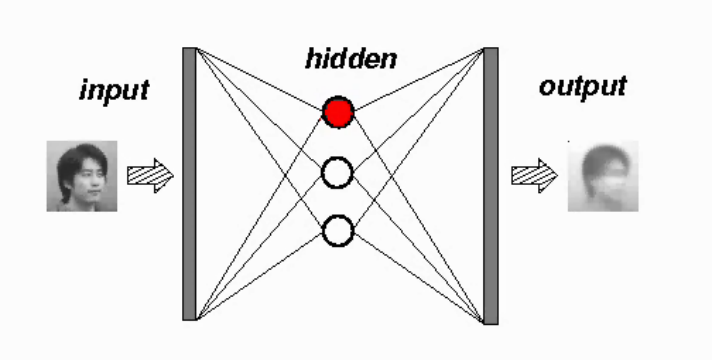


学習曲線

Gating Network
(6個の中間層で3個のみactive)

自己組織化で得られた
代表的な見え

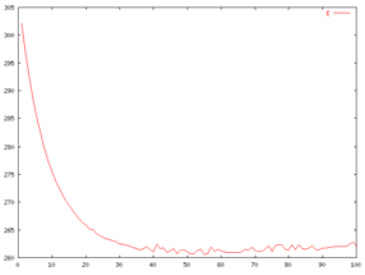
向きに選択的に反応するネットワーク



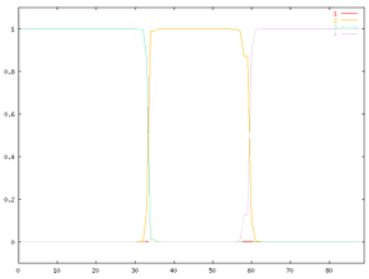
顔画像に対する代表的な見えの自己組織化

入力画像の例

自己組織化で得られた代表的な見え



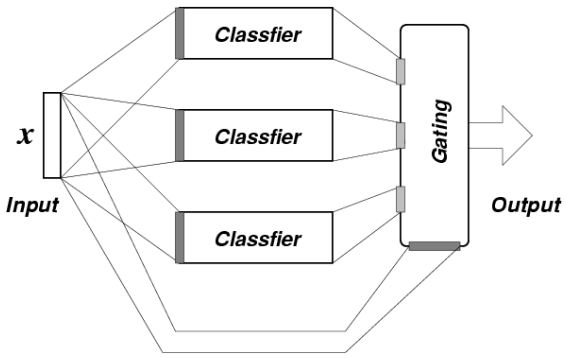
学習曲線



Gating Network
4個の中間層で3個のみActive

Mixture of Experts

- Mixture of Experts
  - Jordan等が提案した、全学習データの部分集合のみを扱うようにした複数の部分ネットワーク(Experts)を結合したネットワークアーキテクチャ（1991）



Mixture of Classifiers の学習

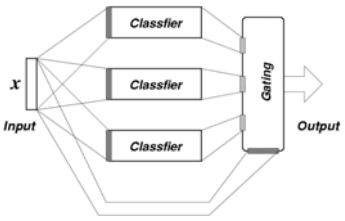
尤度・対数尤度

$$P(\mathbf{t}\,|\,\mathbf{x}) = \sum_{m=1}^{M} g_m P^{(m)}(\mathbf{t}\,|\,\mathbf{x};\mathbf{A}^{(m)}) = \sum_{m=1}^{M} g_m \prod_{k=1}^{K} p_k^{(m)t_k}$$

$$l = \log P(\mathbf{t}\,|\,\mathbf{x}) = \log\left[\sum_{m=1}^{M} g_m P^{(m)}(\mathbf{t}\,|\,\mathbf{x};\mathbf{A}^{(m)})\right]$$

学習アルゴリズム

$$\mathbf{a}_k^{(n)} \Leftarrow \mathbf{a}_k^{(n)} + \alpha h_n (t_k - p_k^{(n)})\mathbf{x}$$
$$\mathbf{b}_n \Leftarrow \mathbf{b}_n + \alpha(h_n - g_n)\mathbf{x}$$

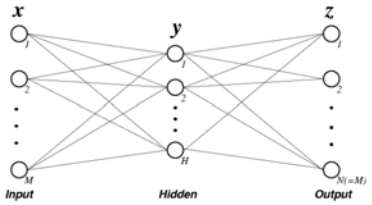


$$h_n = \frac{g_n P^{(n)}(\mathbf{t}\,|\,\mathbf{x};\mathbf{A}^{(n)})}{\sum_{m=1}^{M} g_m P^{(m)}(\mathbf{t}\,|\,\mathbf{x};\mathbf{A}^{(m)})}$$

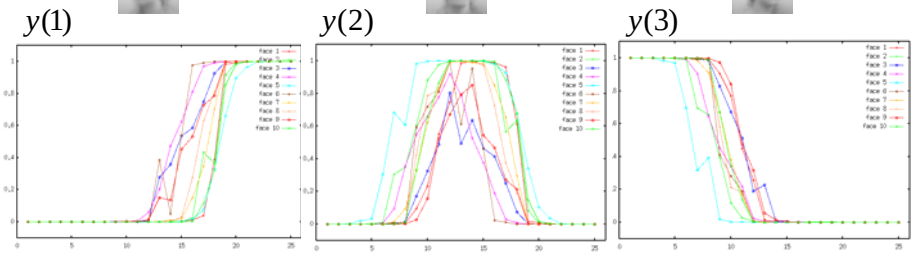
入力 x に対するn番目の識別器の事後確率

顔の向きの表現の自己組織化

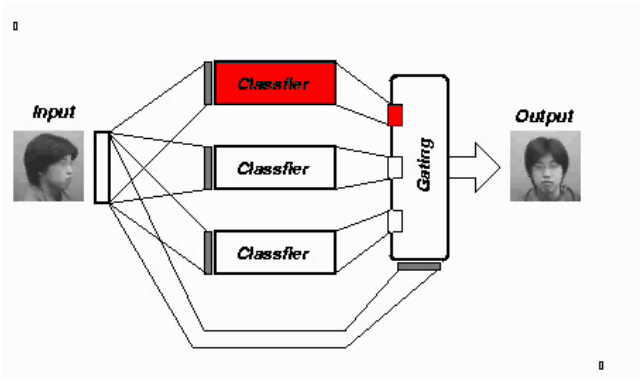
中間層にSoftmax型素子（競合学習）を持つニューラルネットワークを用いた恒等写像学習



W(1) W(2) W(3)



向きに依存しない顔認識



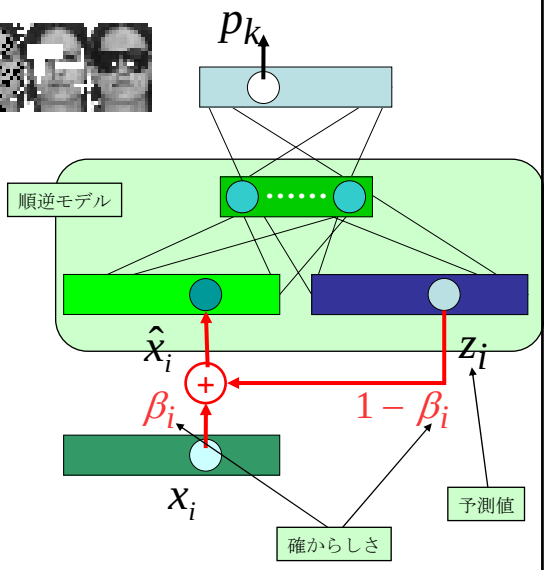
部分的な隠れに影響されにくい顔認識

部分的に隠れた画像の想起と認識

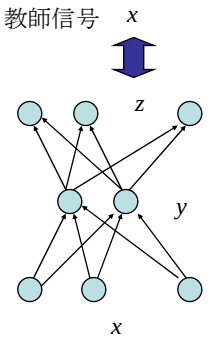
- 隠れや認識対象以外の部分を自動的に除去する機構を持つ認識器は、どのように作ればよいか？（ヒントは？）
  - 脳での視覚情報処理では、網膜から脳の高次中枢へのボトムアップな情報の流れだけでなく、トップダウンの情報の流れが第1次視覚野にも存在している。＝＞ 順逆モデル
  - ロバストテンプレートマッチング（栗田1997）＝＞ 例外地除去
  - 自己連想メモリ（Kohonen1989)
  - 主成分分析や恒等写像を学習する階層型ニューラルネット（順逆モデル）を用いて、自己連想メモリを実現可能

部分的に隠れた画像の想起と認識

- 自己連想メモリ
  - 順逆モデル(Autoencoder)として実現
  - 入力画素値と想起された画素の値との差により確からしさを求め、
  - 入力情報を修正することで元の画像を推定する
- 識別器
  - Multinomial Logit Model
  - 順逆モデルとの情報の共有



恒等写像学習



- 多層パーセプトロン

$$z_k = b_k^T y = \sum_{j=1}^{J} b_{jk} y_j \qquad y_j = a_j^T x = \sum_{i=1}^{I} a_{ij} x_i$$

- 評価基準(2乗誤差最小)

$$E = \frac{1}{2}\sum_{p=1}^{P} \| x_p - z_p \|^2$$

- 学習則

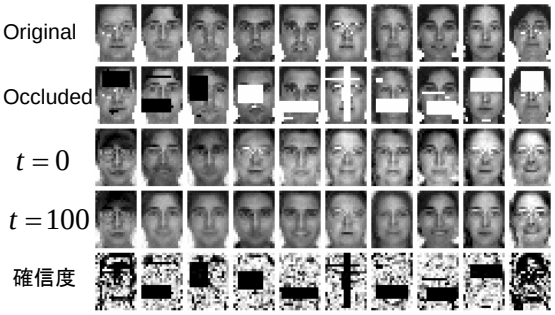
$$\Delta a_{ij} = -\frac{\partial E}{\partial a_{ij}} \qquad \Delta b_{jk} = -\frac{\partial E}{\partial b_{jk}}$$

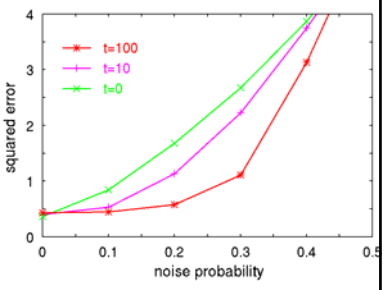
$$\frac{\partial E}{\partial a_{ij}} = -\sum_{p=1}^{P}\sum_{l=1}^{I} (x_{lp} - z_{lp}) b_{jl} x_{ip}$$

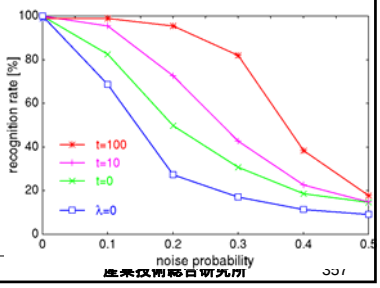
$$\frac{\partial E}{\partial b_{jk}} = -\sum_{p=1}^{P} (x_{jt} - z_{jt}) y_k$$

Recall from the occluded images (rectangular occlusions)

20% of occlusions







Recall from the occluded images (occlusions by sunglasses)

Occlusions by sunglasses

	t=0	t=10	t=100	λ=0
Recognition Rates [%]	77.4	87.1	87.1	64.5

非線形への拡張

- Linear net: Linear MLP + Classifier
- Kernel PCA: Kernel PCA + Classifier
- Classifier: Multinomial Logit Model
- Test data: sunglass
- #iteration: 100(Linear) and 20(Kernel)

Recognition Rate [%]

	initial	aft. iteration
Linear Base	77.4	87.1
Kernel Base	87.1	96.8

矩形特徴を用いた顔検出器から得られる識別スコアの最大化による顔追跡

アプローチ

- 対象追跡に対する2つのアプローチ
  - 力学系ベース（particle filter, mean shift）
    - 対象に何らかのダイナミクス（動きのモデル）を仮定し、それを頼りに対象の移動先を予想する
  - 「見え」ベース
    - 対象の画像としての現れ方（曲線、エッジ、色、形状など）の固有性を頼りに追跡を行う
- 我々のアプローチ
  - 対象のダイナミクスを仮定しない方法（見えベース）
  - 検出器ベースの追跡器（静止画から顔を検出するための手法をより高速化する手法）

ラフ追跡＋精密探索

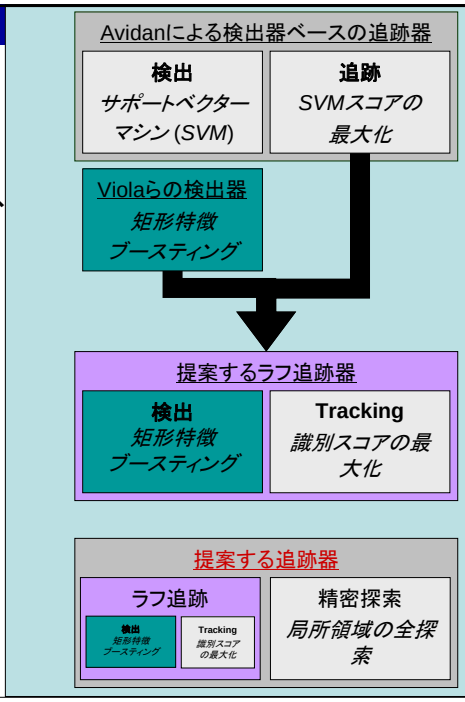
- 人間の視覚系における物体追跡：
  - サッケード運動と追従眼球運動の組合わせ
  ⇒「ラフな探索」と「精密な探索」の組合わせによって
  高速かつ正確な物体追跡を実現

⇒ 提案手法にもこの仕組みを取り入れる

ラフ追跡　＋　精密探索

提案手法

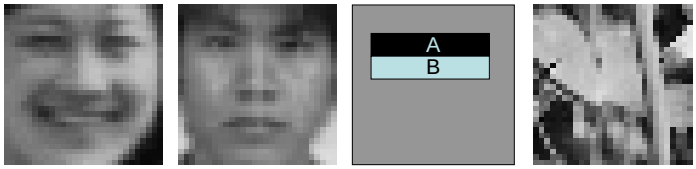
- ラフ追跡＋精密探索
- ラフ追跡器＝検出器ベースの追跡手法
  - Viola-Jonesの検出器
  - Avidanの検出器ベースの追跡器
- 精密探索
  - 局所領域の全探索



矩形特徴による顔検出

- 矩形特徴：特定領域の明るさの違いに基づいて識別を行うフィルタ

人の顔画像とそうでない画像をたくさん用意
⇒ 人の顔に特有な明るさの違い方を学習

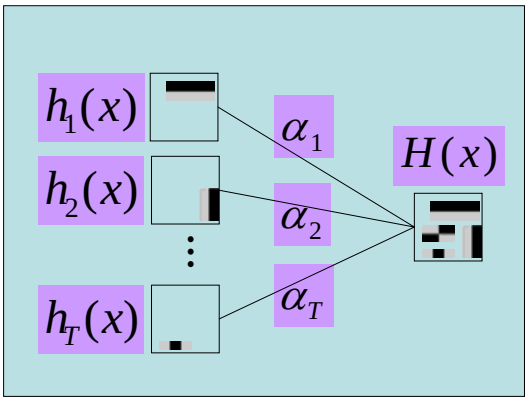


例）顔画像ではAの部分よりBの部分の方が明るいが、背景画像ではそうでないことが多い

未知の画像にこの法則を当てはめて識別を行う

Boostingにより弱識別器を組み合わせた顔検出

- Boosting：弱識別器を多数組み合わせてより強力な識別器を構築



最終的な識別結果

t番目の矩形特徴の識別結果

t番目の矩形特徴の信頼度（重み）

できあがる識別器：弱識別器の重みつき多数決

スコア関数の最大化による顔追跡

- E(I):対象の顔らしさの度合いを表すスコア関数

顔の追跡＝画像平面の中でスコアE(I)が一番大きい領域を追跡すること

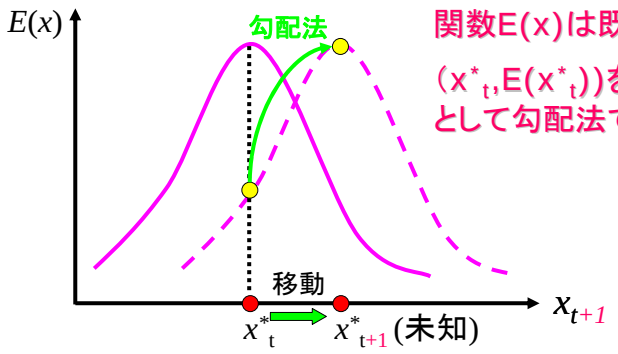
提案手法のスコア関数

$$E(I)=\sum_{t=1}^{T}\frac{\alpha_t}{1+e^{p_t(f_t(I)-\theta_t)}}$$

⇒ 位置に関して微分可能
⇒ 勾配法による関数の最大化

スコア関数の最大化による顔追跡

- 勾配法によるスコア関数E(x)のピーク追跡の様子



関数E(x)は既知
$(x^*_t, E(x^*_t))$を初期点として勾配法で最大化

到達点を対象の現在地とみなす

※図は1変数の場合
※実際は画像平面(x,y)上にE(x,y)が分布した空間で追跡

ラフ追跡と精密探索の組み合わせ

- 局所全探索
  - 移動前の対象位置を中心としたN×N個の近傍領域を顔検出器でサーチ
  - 対象が近傍領域内にいなければ原理的に追跡不可能
  - 処理速度は静的な検出器と同等
- 勾配法による最大化＋局所全探索
  - ぼかし画像:スコア関数を勾配法で最大化
  - 原画像:スコア関数を局所全探索で最大化

実験

- 実験に用いた動画
  - 320x240 pixel, 1286フレーム
- 検出器
  - 正面向きの顔画像725枚＋非顔画像2200枚(24×24ピクセルを用いて反復200回のBoostingで学習
- 追跡器
  - ぼかし画像:原画像＋2レベルのぼかし画像(計3レベル)
  - 近傍サイズ:前回の位置を中心とする5×5マス
  - 比較する追跡アルゴリズム
    - SDM(最急降下法)のみ
    - 局所全探索のみ
    - SDM＋局所全探索

実験結果

	SDM＋局所探索	SDMのみ	局所探索のみ
追跡失敗回数	3回	15回	40回以上
追跡結果	対象の動きが速いシーンや大きさ・向きが変わるシーン以外では追跡成功位置ウィンドウの振動も抑制	勾配法の収束が不十分で、追跡成功時でも位置ウィンドウが乱雑に振動してしまう	ほとんど追跡できない（対象が近傍外に動くと原理的に追跡不能）

- SDM＋局所全探索
  - 局所全探索のみより大きな移動量に対応
  - SDMのみより追跡精度が向上/追跡結果が安定
- 計算時間
  - 約40fps（Pentium4 2GHzマシン）
  - （矩形特徴200個の検出器による全探索では約0.1～0.2fps）

顔追跡の例（局所探索のみの場合）

顔追跡の例(ラフ追跡＋局所探索の場合)

歩行者検出のための部分特徴のブースティングによる統合

歩行者の検出

任意の画像中の任意の大きさの歩行者を検出する

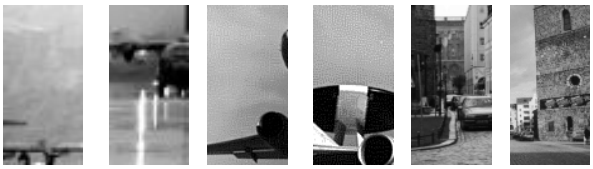
↓ そのためには

定まった大きさの画像中に

- 定まった大きさの歩行者が含まれているものと
- 歩行者の含まれていないものを区別する

歩行者の検出
歩行者画像と非歩行者画像の識別

歩行者画像

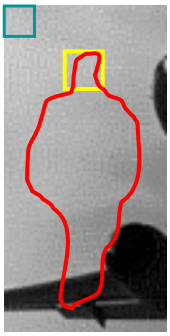


非歩行者画像

- 歩行者：MIT CBCL画像データベース 924枚
- 非歩行者： ランダムに選択した画像 2700枚

歩行者の識別(特徴選択)

この二つの画像を区別するのにどの部分を使うのが良いか？



歩行者検出の課題

- 姿形のバリエーションが豊富
- 色，模様のバリエーションが豊富

部品ごとの検出器を統合する

頭，腕，足などの識別器を作って，それらを統合する

歩行者検出の先行研究

姿形のバリエーションが豊富：
単一のモデルでは対応しにくい

- Gavrila 階層的テンプレートマッチング
  - **色々なテンプレートを用意する**
  - **テンプレートは人間が作成する**
- Mohan, Papageorgiou, Poggio
  - **部品ベースの識別器(SVM)＋上位の識別器（SVM）**
  - **部品の選択は人間が行う**
- Viola, Jones, Snow
  - **動き情報の統合、AdaBoost**
  - **高速、高性能実現**
  - **カメラ自体の動きには対応し難い**

紹介する手法の概要

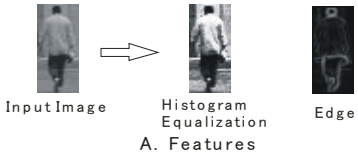
- 外形ベースの識別
  - 動き情報は用いない(カメラが動く場合を考慮)
- 部品の自動抽出
  - テンプレートや部品に相当する情報を切り出す
  - 部品ごとの情報を自動的に統合

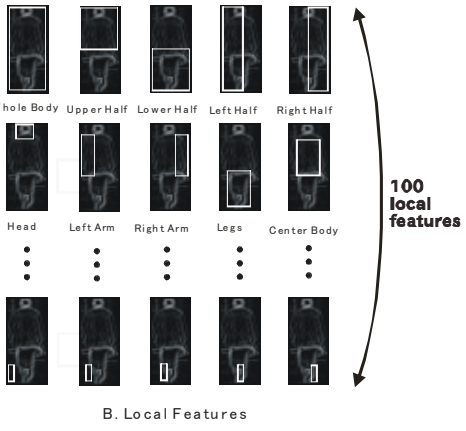
•部品ごとの識別器をBoostingで統合する

•識別器としてSVMを用いる

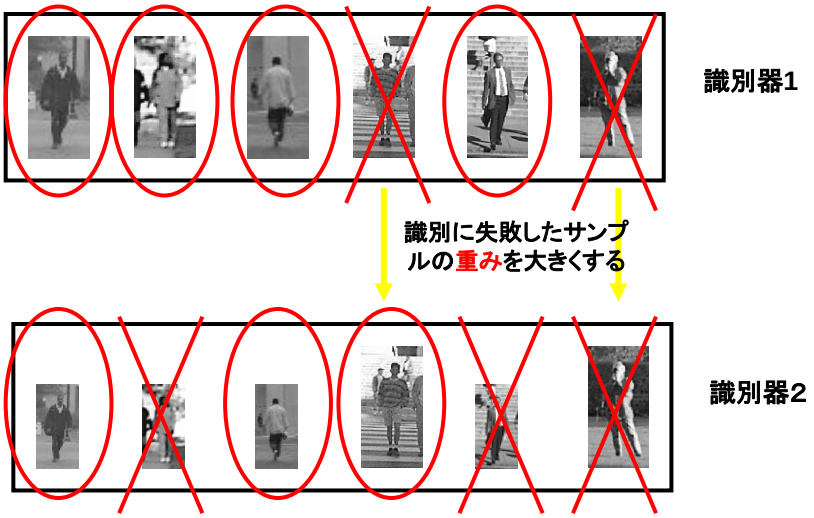
特徴と局所特徴

- 特徴
  - ヒストグラム均一化
  - エッジ
- 局所特徴(領域)
  - 100 部分領域



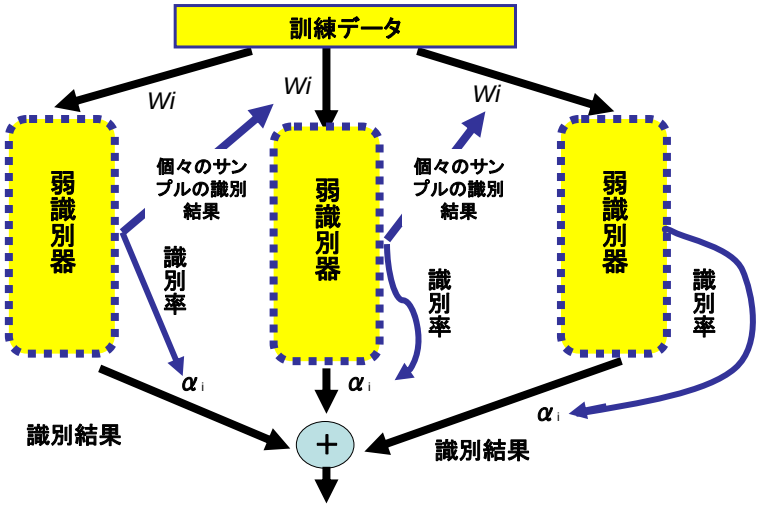


Boosting



Boosting(全体)

複数の識別器を組み合わせて高性能な識別器を構成する手法



SVMのソフトマージン化

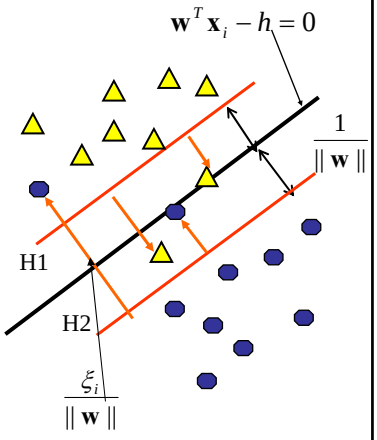
制約条件付き最適化問題
（ソフトマージン）

- **目的関数:**

$$L(\mathbf{w},\xi)=\frac{1}{2}\|\mathbf{w}\|^2+C\sum_{i=1}^{N}\xi_i$$

- **制約条件:**

$$\xi_i \geq 0,$$
$$t_i(\mathbf{w}^T\mathbf{x}-h)\geq 1_i-\xi_i \quad \text{for } i=1,\ldots,N$$



- **ソフトマージン**
  - **マージンを最大としながら、幾つかのサンプルが超平面を越えて反対側に入ってしまうことを許す**
- **ペナルティ**
  - **反対側にどれくらい入り込んだのかの距離の和**

$$\sum_{i=1}^{N}\frac{\xi_i}{\|\mathbf{w}\|}$$

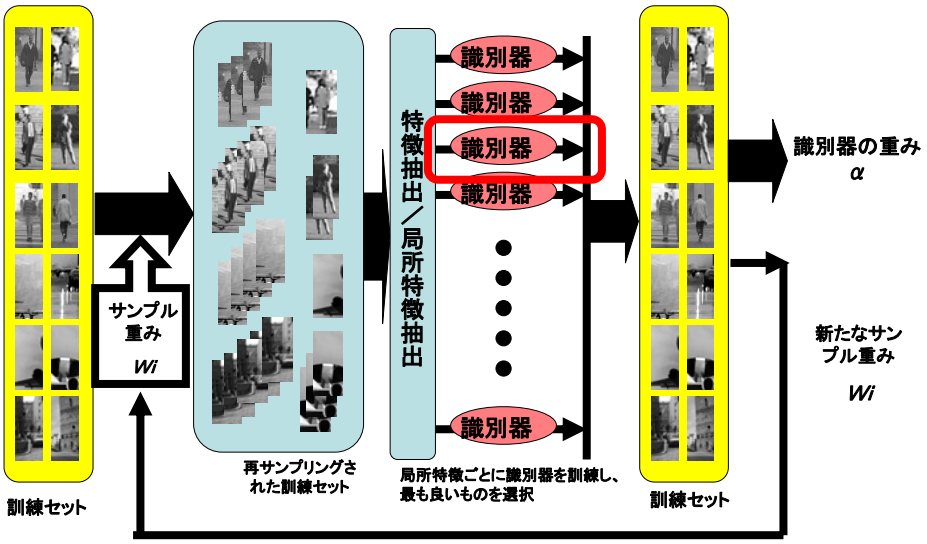
SVMをBoostingに用いる際の課題

- **弱識別器化**
  - **Soft-Margin SVMを用いる**
- **サンプルの重み付け**
  - **Marginのコストを組み込んだSVMを作る**

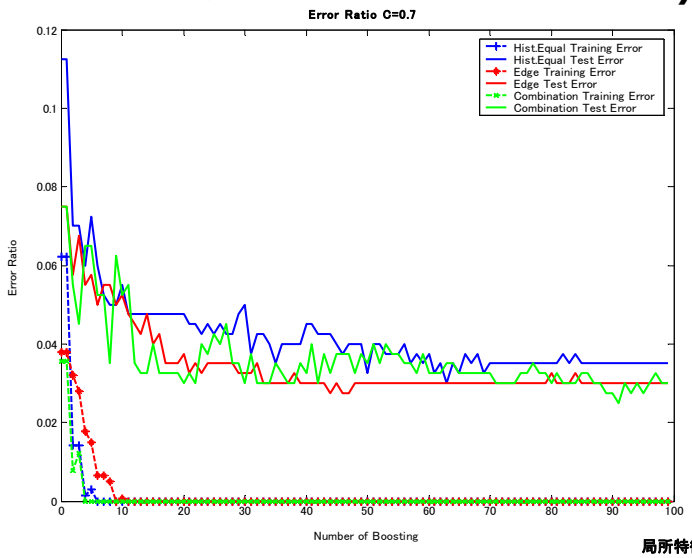
あるいは

- **サンプル重みにしたがって訓練データを再サンプリングする**

再サンプリングを用いたBoostingアルゴリズム

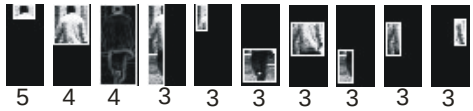


実験結果3（二特徴、100局所特徴)

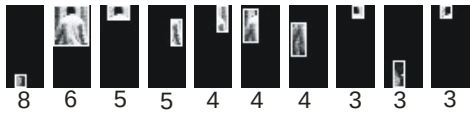


局所特徴100パターン

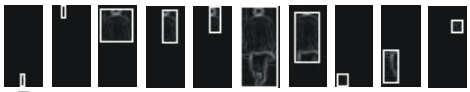
二特徴選択の効果



Combination (Histogram-Equalization, Edge)



Histogram-Equalization



Edge

Numbers denote the Selection frequency of each local feature

選択された特徴領域とその頻度

先行研究との比較

- Gavrila
  - 1段目（テンプレートマッチング）の識別率60－90%
  - 2段目でFPをリジェクト
- Mohan, Papageorgiou, Poggio
  - 識別率98－99%、低FP率（0.1%）
- Viola, Jones, Snow
  - 高速（4 f/s)
  - 識別率90%,低FP率（0.1%）
- 提案手法
  - 識別率97－98%

証明写真１枚しかないときに任意の向きの顔画像から識別するには？

多方向顔画像の主成分分析による任意方向顔画像の生成と認識

課題

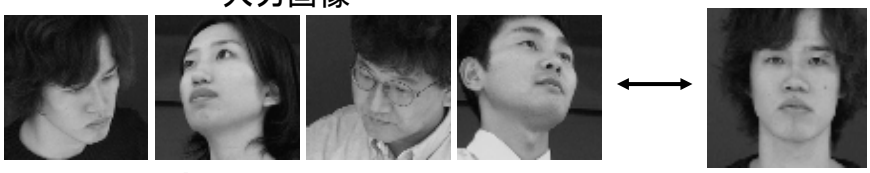
・なんらかの理由で人物Aの捜索が必要となった。

・付近一帯にある全ての防犯カメラをチェックして人物Aの捜索をしたい。

・人物Aの外見を表すものは、免許証一枚だけであるとする。

提案手法のアプローチ

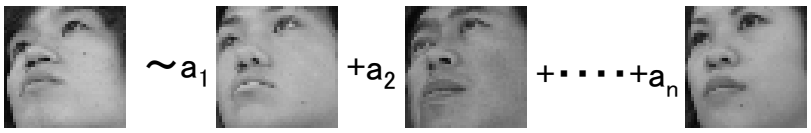
入力画像



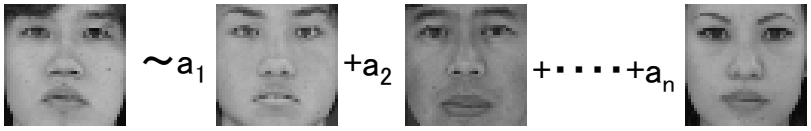
捜索人物

正面方向からの見え方を推定

Linear classes [Vetter1993]



同じ向きの複数人の顔画像を基底として、未知の顔画像を最小二乗近似する



基底に用いた人物の正面顔を基底として、未知の人物の正面顔画像を推定

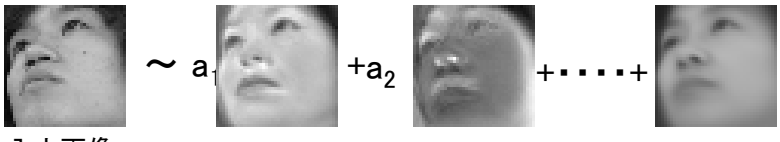
T.Vetter and T.Poggio, "Linear object classes and image synthesis from a single exmaple image, " A.I.Memo No.1531,Artificial Intelligence Laboratory,Massachusetts Institute of Technology, 1995.

Linear Classes の問題点

- 基底に用いる顔画像の枚数
  - Linear Classesにおいて、精度よい近似をするためには、充分多くの基底画像を準備する必要がある。
- 基底ベクトルの直交性
  - 基底として用いた実際の顔画像は互いが画像ベクトルとして似ており、直交しない。そのため近似性能が落ちる。

提案手法による正面顔の推定

基底を「実際の顔画像」から、主成分分析を利用して「固有顔」へ



入力画像　　平均顔

平均顔

固有顔と平均顔

画素数Mのサンプル顔画像（N枚）

平均顔（1枚）　固有顔（N－1枚）

平均顔　N人の顔画像を同じ画素ごとに平均をとった画像

固有顔　顔画像ベクトルの集合を主成分分析したもの。

少数の基底での復元

u_1　u_2　u_L

第1主成分　第2主成分　第L主成分

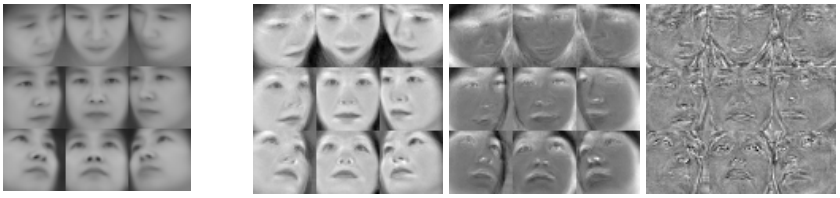
固有値の大きい固有ベクトル
→N枚の画像を区別するのにより重要な特徴を含んだベクトル

100人の顔画像を固有ベクトル50次元を使って表現した時、95%の復元率を得ている。

$$\text{寄与率} = \frac{\text{用いる固有ベクトルの固有値の和}}{\text{全ての固有ベクトルの固有値の和}}$$

多方向固有顔と多方向平均顔

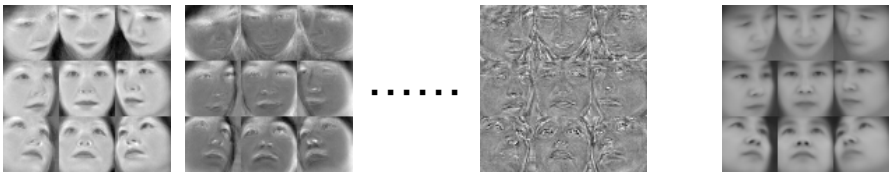
多方向顔画像



多方向平均顔　　多方向固有顔

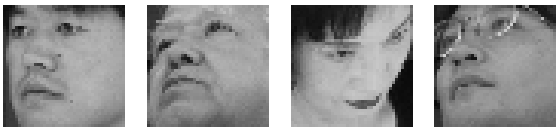
多方向固有顔の作成実験

280人分の多方向顔画像から
279個の多方向固有顔と
1枚の多方向平均顔を生成

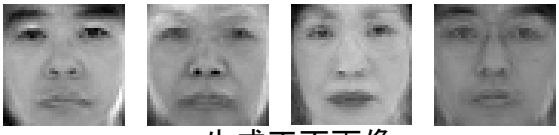


279枚の多方向固有顔　　多方向平均顔

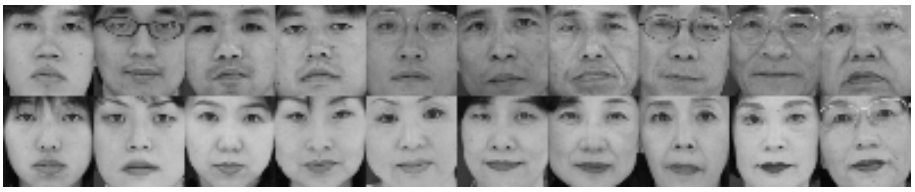
多方向固有顔から切り出した固有顔を用いた、
未知の正面顔画像の推定結果



入力画像



生成正面画像



照合用正面画像

未知の正面顔画像の推定

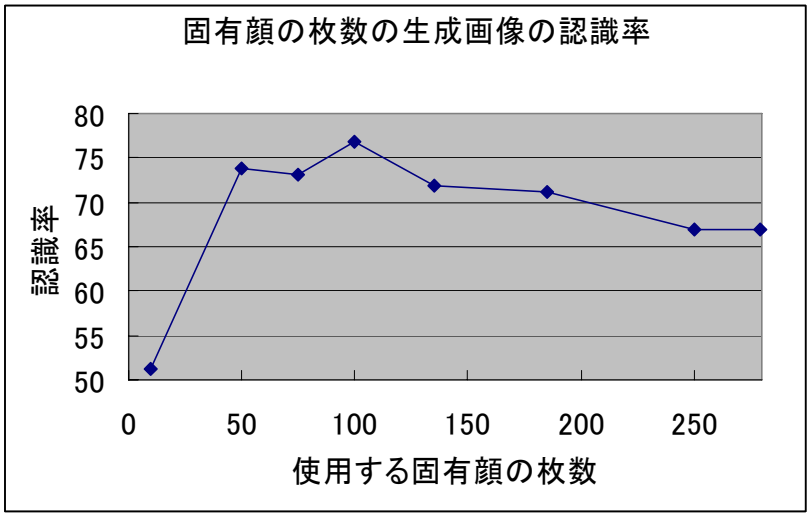
入力画像

男女20人8方向（正面画像を含まない）
計160枚

推定された正面顔

100枚の固有顔を使用

使用する固有顔の枚数と、認識率



固有顔100個を元に画像復元した結果　76．9%

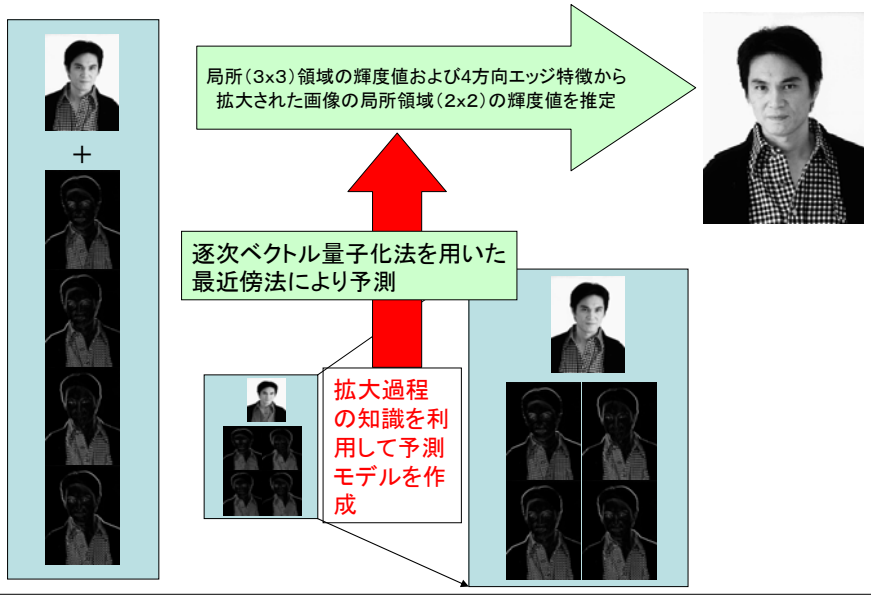
顔が小さくしか写っていなかったらどうする？

超解像度

低解像度画像の画質改善のための統計的手法

- 拡大してもボケの少ない画像をえるには？

4方向エッジ特徴量を利用した予測モデル



拡大画像の生成結果

提案手法

単純拡大

顔検出の高速化

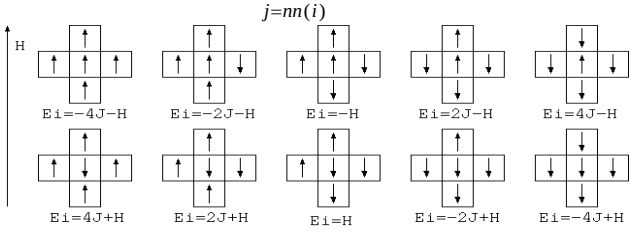
顔探索の高速化手法

- 平均探索時間の短縮のための手法
  - ランダム探索
  - Ising Modelを用いた探索 [SPIE98,ICPR98]
  - 位置に関する事前確率＋Ising Search [ICPR2000]

Ising モデル

- Ising モデル
  - **2つの状態： “up” spin と “down” spin**
  - **ある点のスピンの状態は、周辺の点のスピンの状態と外部磁場に依存して決まる**
  - Ising dynamics ：
    - **エネルギー関数を最小化するように確率的に動く**

$$E_i = -J \sum_{j=nn(i)} s_i s_j - H s_i$$



Dynamic Attention Map

- 顔検出の高速化にIsingモデルを利用
  - 顔である状態：“down” spin (-1)， 顔で無い状態：“up” spin (+1)
  - 外部磁場：調べた点での顔らしさ
  - 初期状態：すべての点は顔である状態（顔の候補）
  - 探索点の周辺のspinの状態を

$\exp(-\beta \Delta E_i)$

に比例した確率で更新

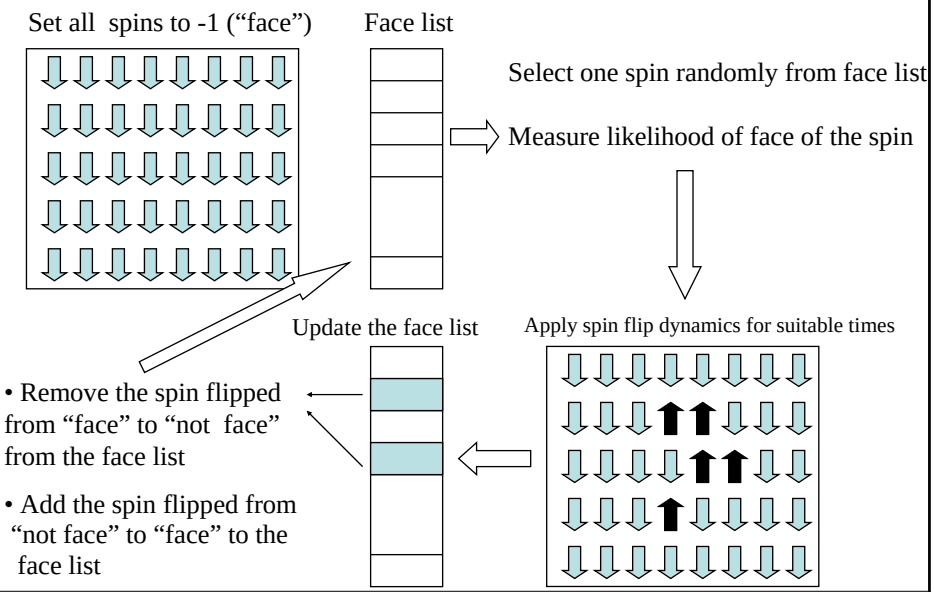
ただし、

$$\Delta E_i = 2J \sum_{j=nn(i)} s_i s_j + 2H_d (m_d(a) - \theta_d) s_i$$

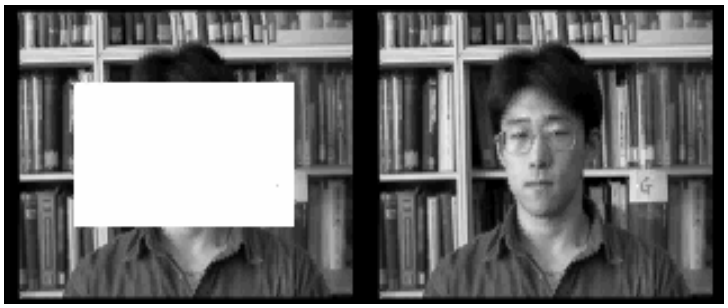
$$\Delta E_i = 2J \sum_{j=nn(i)} s_i s_j + 2H s_i$$

Dynamic Attention Map

Ising 探索アルゴリズム

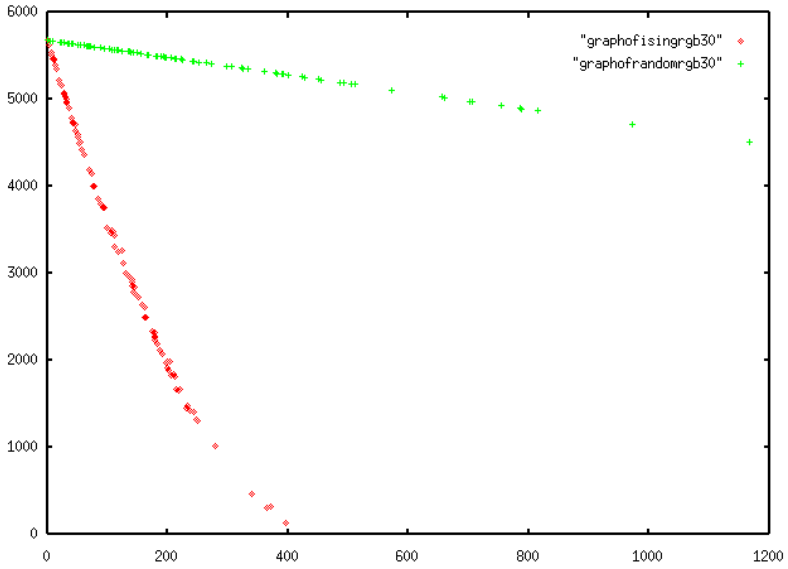


Ising探索の例



Dynamic Attention Map　　探索点Map

顔候補点の削減の様子

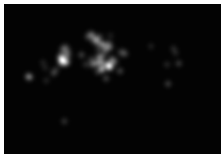


Ising探索での探索点

- 顔の候補点がIsing dynamicsにより大幅に削減される

推定された事前確率

After 10 images

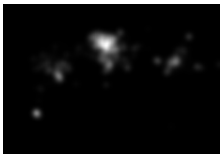


After 100 images

After 500 images

After 1000 images

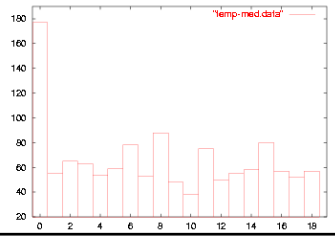
After 1500 images



After 1849 images

Number of search points needed to detect face:

- Whole region search : 28420
- Normal Ising search : 663(median)
- Ising search using priori probability : 60



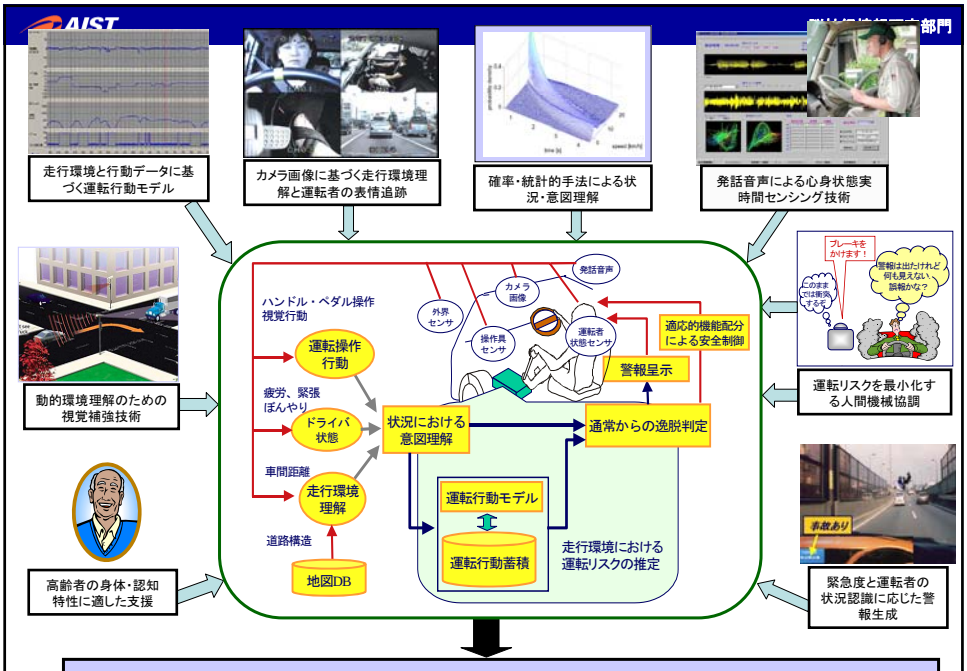
交通安全支援のための状況・意図理解

科学技術振興調整費

重要課題解決型研究 交通事故対策技術の研究開発

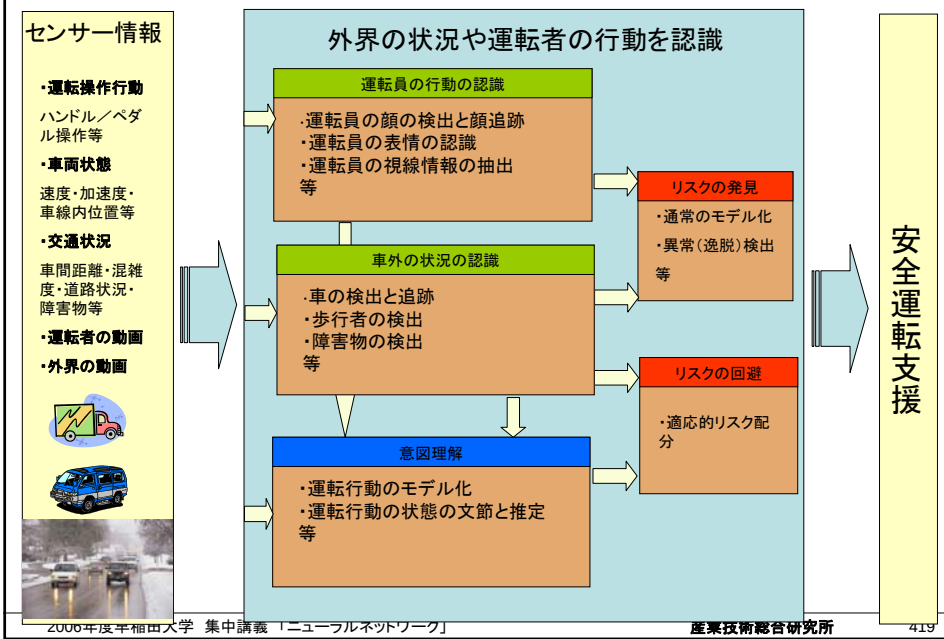
「状況・意図理解によるリスクの発見と回避」

平成16年度～平成18年度



動的環境と人の状態・意図に応じた支援により自動車の安心・安全を確保する予防安全型技術の実現

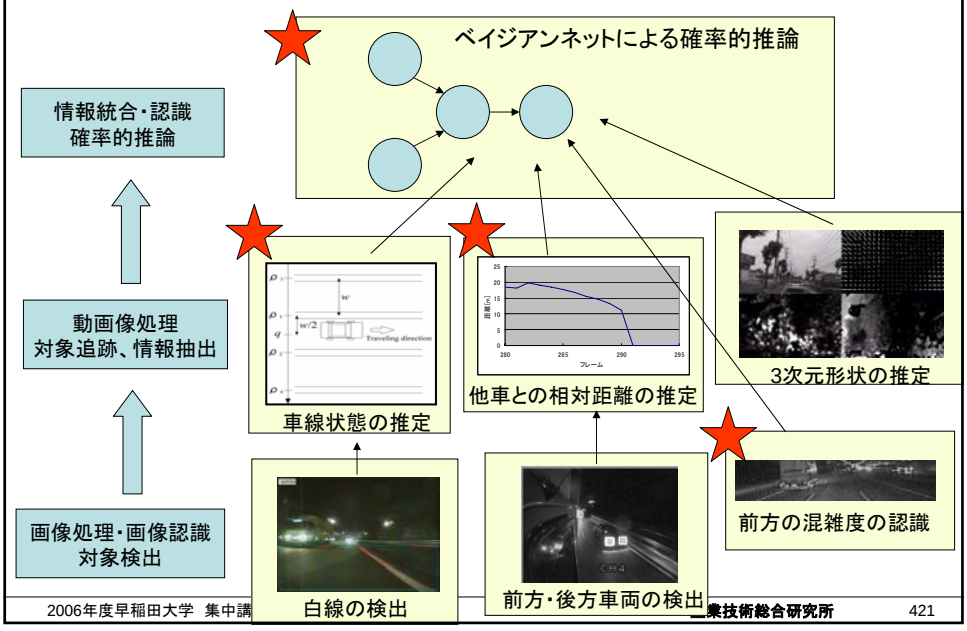
状況・意図理解によるリスクの発見と回避



外界センシング技術の確立

- アプローチ
  - 確率統計的手法の利用
- 簡単な状況の認識
  - 道路の混雑度、天候、一般道／高速道／田舎道
  - 統計的パターン認識手法
    - 特徴抽出（高次局所自己相関特徴）＋識別器の学習
- より複雑な状況の認識
  - 対象の検出 —> 対象の追跡 —> 統計的推論

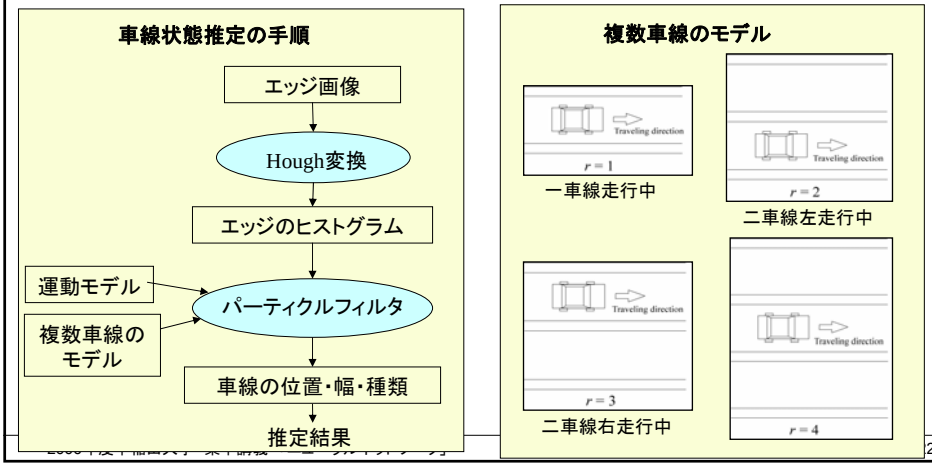
車外の状況（追い越し可能かどうか）の認識



複数車線モデルを用いた走行状態の推定

走行中の車線の状態（何車線の道路の何番目の車線を走行しているか）を推定することは、自車の走行状態を把握するための基礎的な情報として重要である。

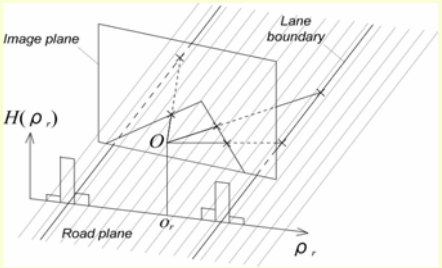
複数の車線モデルを画像から得られるエッジ点群に当てはめることにより、車線の位置・幅・種類を推定する。モデルの当てはめには多重モデルパーティクルフィルタを用いる。



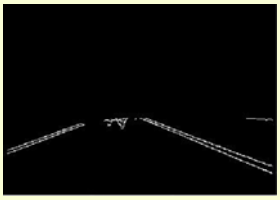
車線検出のための1次元投票（一次元ハフ変換）

仮定
- カメラと路面の相対的位置関係一定
- 車線の方向一定

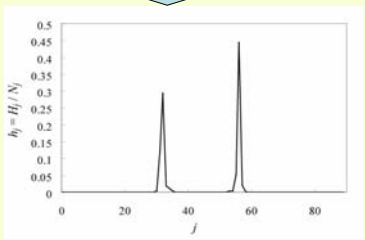
車線の方向と一致しない方向を持つエッジは無視する。



カメラと道路面との幾何学的関係

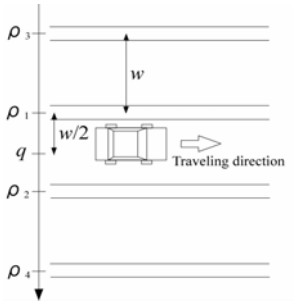


エッジ画像



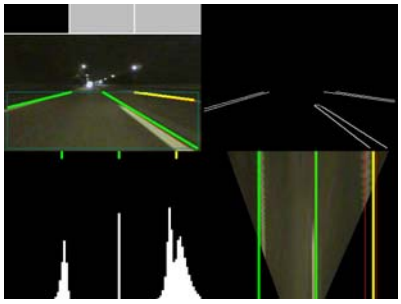
投票結果（エッジヒストグラム）

車線状態の推定



車線の状態 $\mathbf{x} = (q, w, r)$
- 車線の位置 q
- 車線の幅 w
- 車線の配置 1)〜4) r

多重モデルパーティクルフィルタにより推定

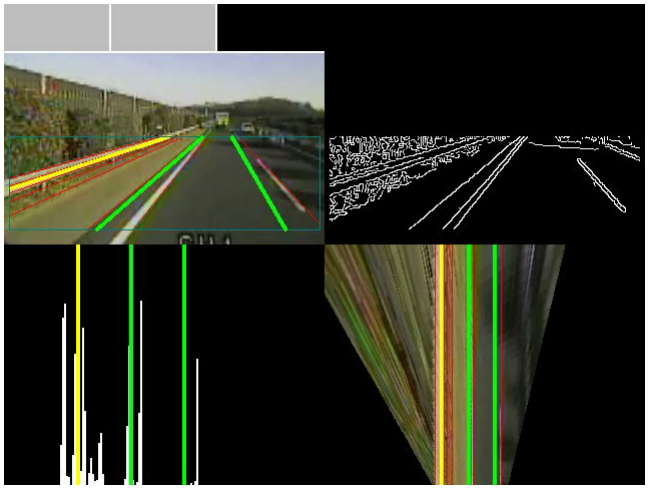


- 左上：原画像
- 右上：エッジ画像
- 左下：エッジヒストグラム
- 右下：路面を真上から見た画像

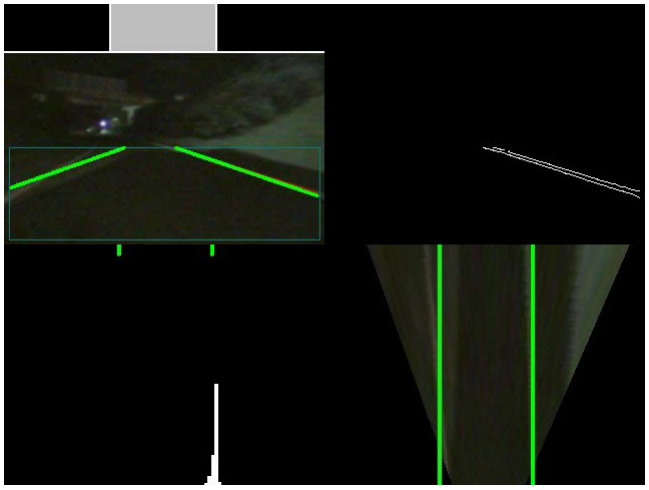
緑の線：ρ_1、ρ_2

黄色の線：ρ_3、ρ_4

走行状況の認識例1（前方画像）

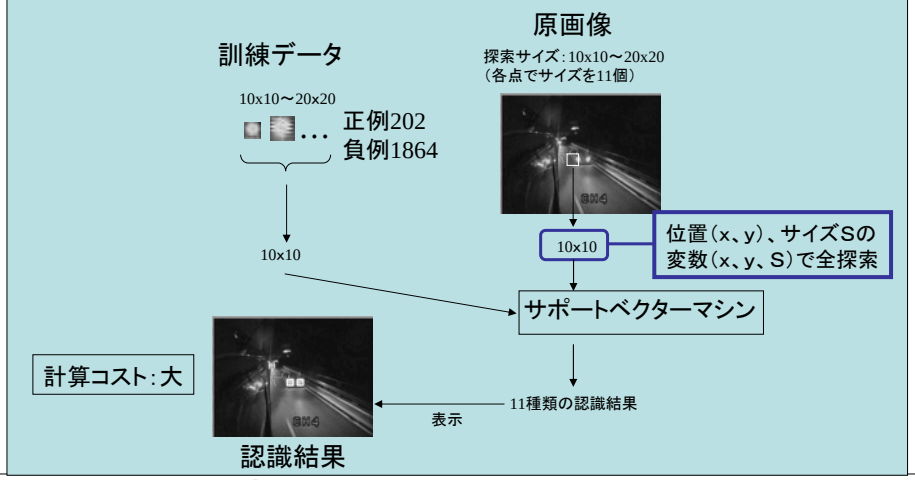


走行状況の認識例1（後方画像）



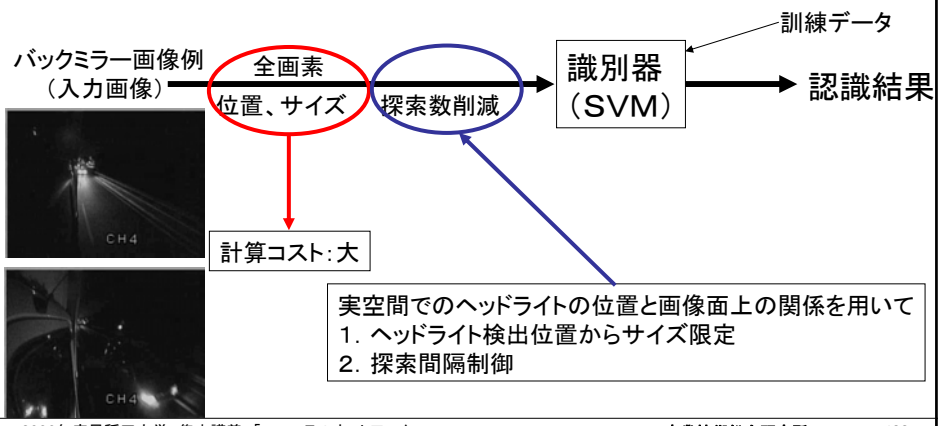
後方車両の状態の認識

サポートベクターマシンによるヘッドライト検出



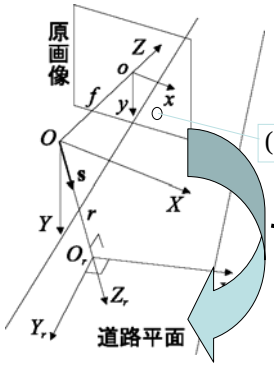
道路平面拘束を用いた夜間車両検出の高速化

後続車両のヘッドライトをSVMに訓練させ、夜間のバックミラー画像からヘッドライトのみを認識させる手法。



後方車両の検出結果の例

後続車両との距離の推定

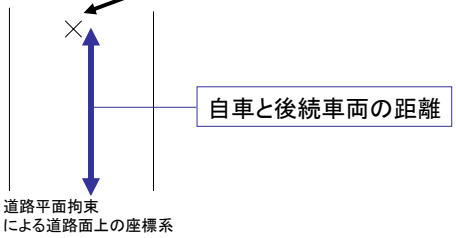


カメラ座標系でのヘッドライトの三次元位置：$\mathbf{X}_h$

$$\mathbf{X}_h = \frac{r}{\mathbf{s}^T \mathbf{x}_h} \mathbf{x}_h$$

検出されたヘッドライトの
道路面上における座標

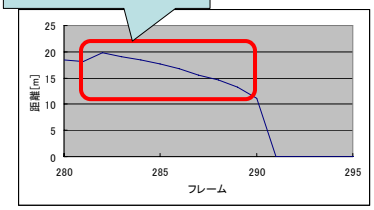
$$\mathbf{X}_r^h = R\mathbf{X}_h - [0,0,r]^T$$



後続車両との距離の推定

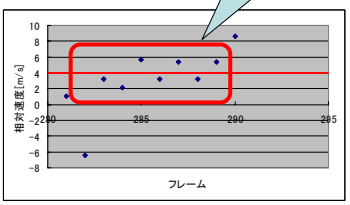
◎レーン(高速道路)の車線幅:3.5[m]
◎1秒間:4フレーム

近づいてくる



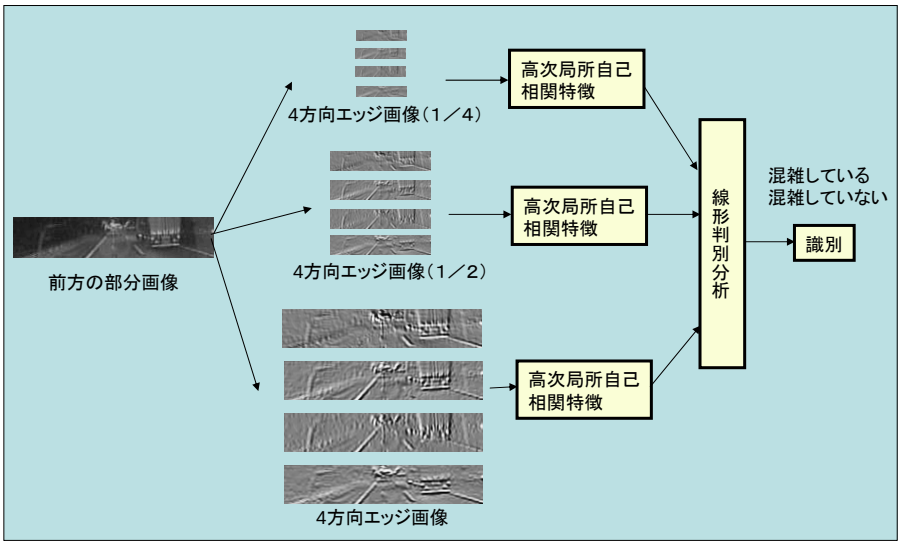
自車と後続車両の距離

14.4[km/h]



自車と後続車両の相対速度

前方の混雑度の認識



混雑度の識別結果の例

•学習サンプル
9706画像
（混雑6241枚、
それ以外3465枚）

・学習サンプルの識別結果
約92.93%

混雑と判定された画像

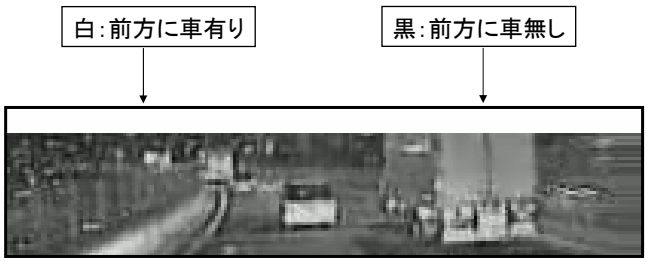
混雑していないと
判定された画像

混雑度の推定例（高速道）

白：前方に車有り

黒：前方に車無し

混雑度の推定例(首都高)



独立行政法人産業技術総合研究所

脳神経情報研究部門

- 産業技術総合研究所
  - URL http://www.aist.go.jp/
- 脳神経情報研究部門
  - URL http://unit.aist.go.jp/neurosci/
- 栗田多喜夫
  - URL http://staff.aist.go.jp/takio-kurita/
  - Email takio-kurita@aist.go.jp
  - 筑波大学連携大学院