

Análise e Processamento de Sinal e Imagem

V - Introdução ao Reconhecimento de Padrões

António M. Gonçalves Pinheiro

Departamento de Física
Universidade da Beira Interior
Covilhã - Portugal

pinheiro@ubi.pt

Reconhecimento de Padrões

1. Caracterização de Sinais e Imagem
2. Técnicas de Classificação
3. Classificação Não-supervisionada

Reconhecimento de Padrões

Introdução

Facilidade com que reconhecemos:

- Uma cara
- Palavras ouvidas
- Lemos caracteres escritos
- Identificamos as nossas chaves nos bolsos
- Notamos comida estragada pelo cheiro

→ Tem por trás um processo de **RECONHECIMENTO DE PADRÕES**

Reconhecimento de Padrões

Bibliografia

1. S. Theodoridis, K. Koutroumbas, *Pattern Recognition*, Academic Press, 4th edition, 2009.
2. Richard O. Duda, Peter E. Hart and David G. Stork, *Pattern Classification*, 2nd Ed., Wiley, 2001, ISBN: 0-471-05669-3
3. Linda G. Shapiro and George C. Stockman, *Computer Vision*, Prentice Hall, 2001, ISBN: 0-13-030796-3

Reconhecimento de Padrões

Máquinas com Percepção

Aparelhos capazes de executar tarefas de **Reconhecimento de Padrões** são um dos grandes desafios tecnológicos:

- Reconhecimento de Fala
- Identificação de impressões digitais (ou outras características biométricas)
- Reconhecimento Óptico de Caracteres
- Identificação de Sequências de DNA

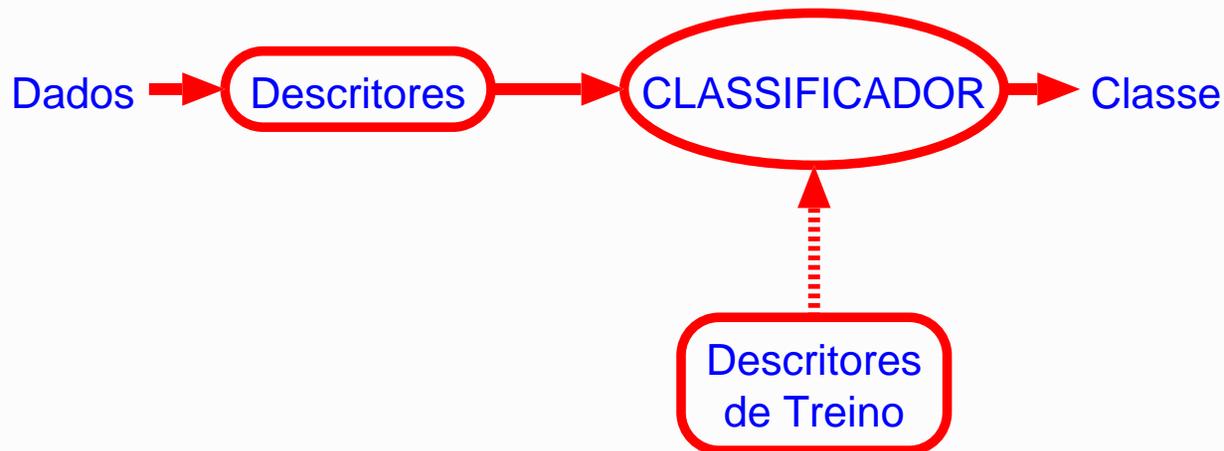
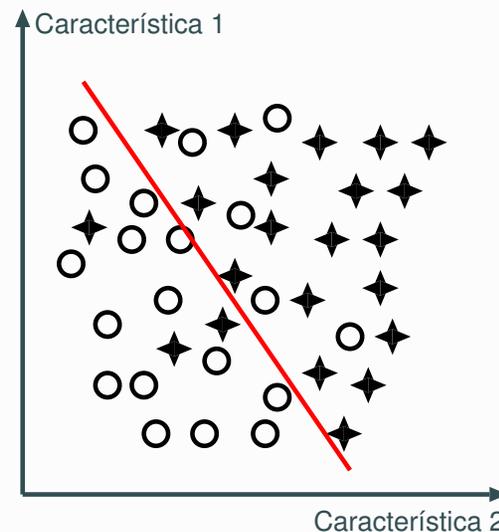
Exigência de **Precisão** e **Fiabilidade** para que sistemas de **Reconhecimento de Padrões** sejam verdadeiramente úteis.

Sistema de Reconhecimento de Padrões

Definição de Teoria da Decisão

Estabelece Regras de Decisão de forma a estabelecer Limites que minimizam o Custo.

O Reconhecimento de Padrões é um subcampo da Teoria da Decisão.

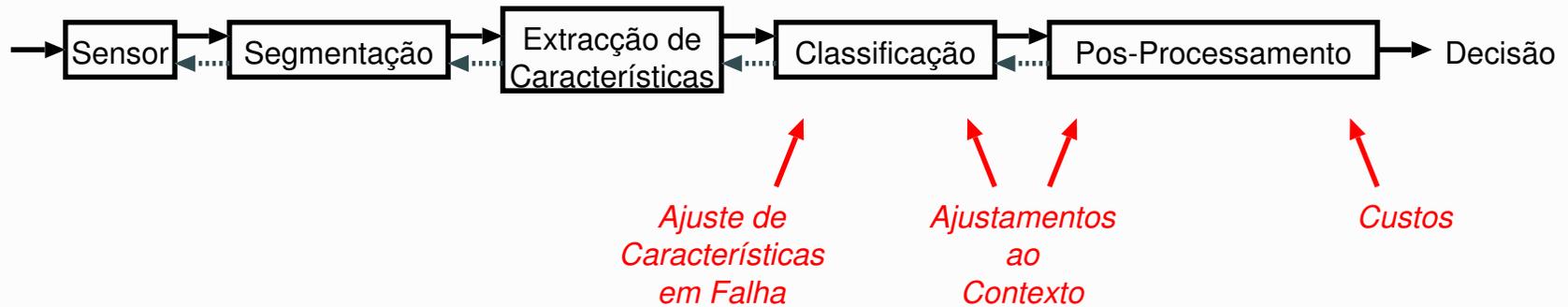


Sistema de Reconhecimento de Padrões

Áreas Relacionadas

- **Teste de Hipóteses**
Decide qual a probabilidade de uma determinada ocorrência se dar.
- **Processamento de Imagem**
Pretende obter uma nova imagem a partir de uma imagem original. Em Reconhecimento de Padrões extraem-se características de uma imagem.
- **Memória Associativa**
Estes sistemas perante um determinado padrão emitem um outro padrão que em geral é representativo de um grupo geral de padrões.
- **Regressão**
Pretende encontrar uma descrição funcional dos dados, normalmente com o objectivo de poder prever os valores de nova entrada.
- **Interpolação**
- **Estimação de Densidade**
Problema de Estimar a Densidade/Probabilidade que um membro de certa categoria ter um determinado conjunto de características.

Sistema de Reconhecimento de Padrões



Sensores

Entada do Sistema, tipicamente um Transdutor

- Camara
- Microfone(s)

As limitações dos Sensores originam muitos problemas

(largura de banda, resolução, sensibilidade, distorção, taxa sinal/ruído, latência)

Sistema de Reconhecimento de Padrões



Segmentação (e Agrupamento)

Extracção de informação relevante.

(Um dos maiores problemas do Reconhecimento de Padrões)

Sistema de Reconhecimento de Padrões

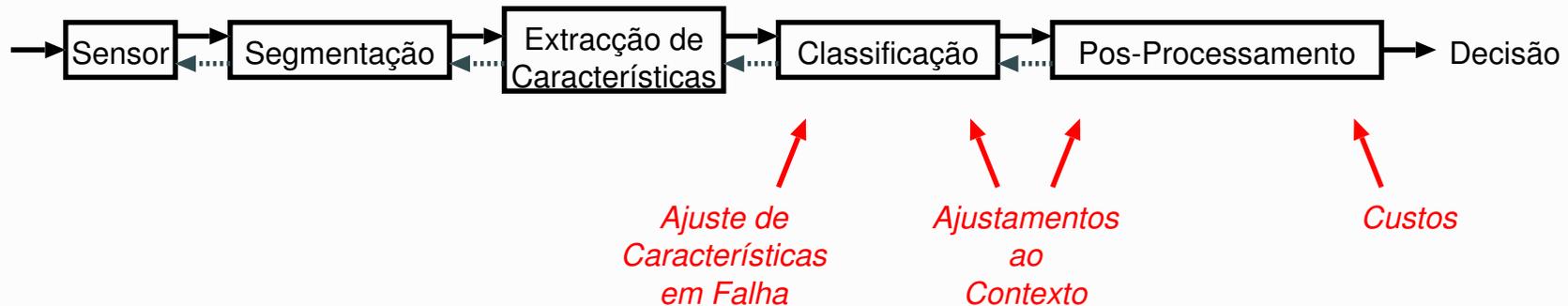


Extração de Características

Tem como objectivo caracterizar um objecto para ser reconhecido por medidas que são muito parecidas em objectos da mesma categoria e muito diferentes para objectos de outras categorias.

Pretendem-se **CARACTERÍSTICAS Distinguíveis** que sejam **INVARIANTES** a transformações irrelevantes.

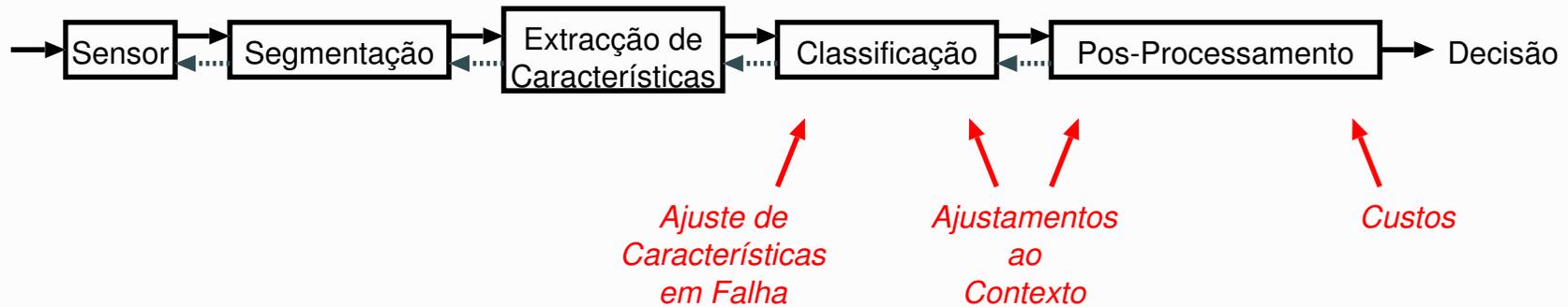
Sistema de Reconhecimento de Padrões



Classificação

- Tem como objectivo usar o vector de características para atribuir ao objecto uma categoria.
- Como uma classificação perfeita é quase sempre impossível, calcula a *Probabilidade* de um objecto pertencer a cada uma das categorias.
- Variação dos valores das características nos objectos da mesma categoria pode ser devida a:
 - Complexidade
 - Ruído/Aleatoriedade

Sistema de Reconhecimento de Padrões



Extração de Características versus Classificação

A fronteira entre estes dois blocos é de alguma forma arbitrária, dependendo do sistema:

- Extractor de Características *Ideal* → Classificação *Trivial*
- Classificação *Omnipotente* → Extractor de Características *Simple*

Sistema de Reconhecimento de Padrões



Pos-Processamento

Executa a Acção Adequada em função do Reconhecimento da Classificação.

Sistema de Reconhecimento de Padrões

Taxa de Erro

Medida mais simples da qualidade de funcionamento da Classificação.

Risco

Custo esperado total de se fazer um erro de classificação.

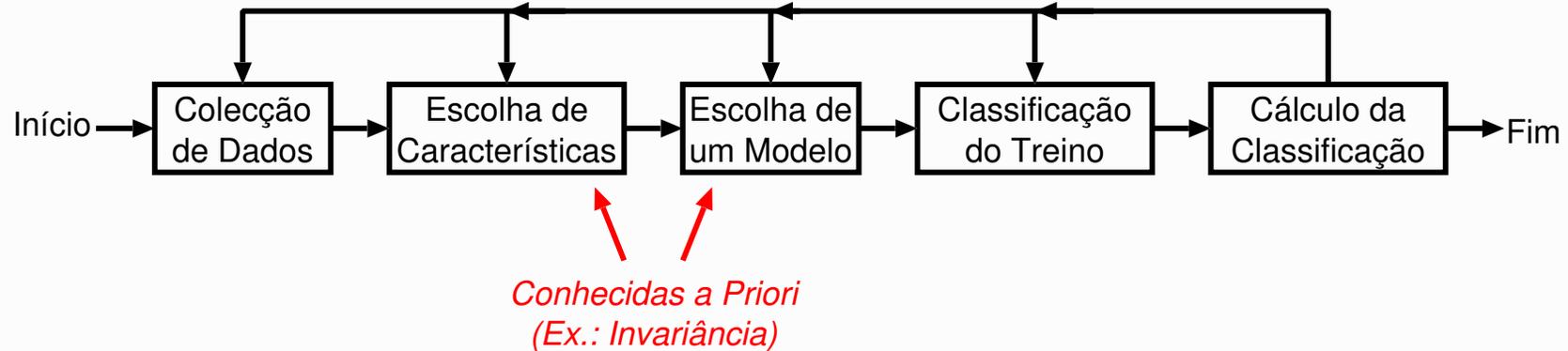
Contexto

Pode ser usado para melhorar os resultados da classificação, considerando a especificidade do modelo.

Múltiplas Classificações

Podem ser combinados de forma a melhorar o Processo de Reconhecimento.

Desenho de Sistema de Reconhecimento de Padrões



Colecção de Dados

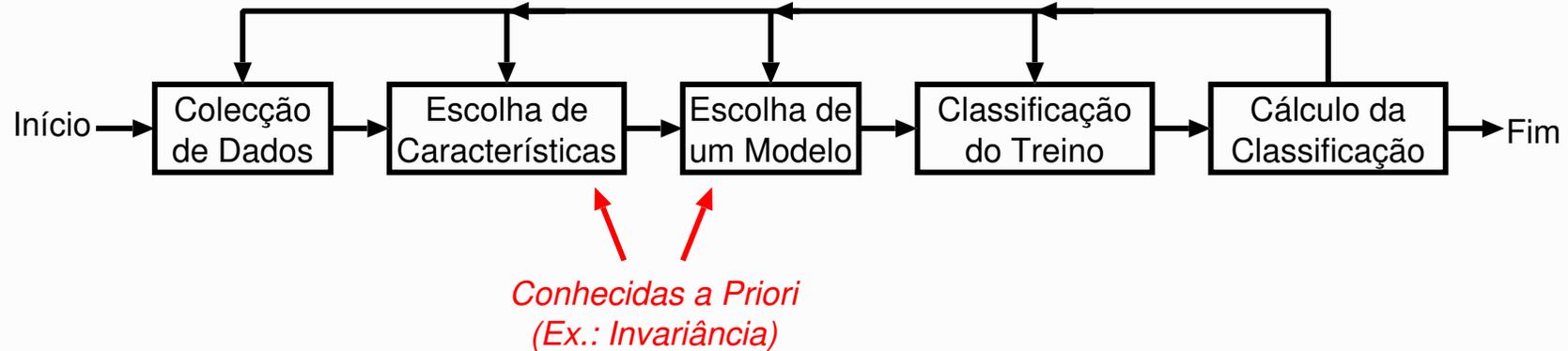
Escolha das Características

Devem ser:

- Simples de Extrair
- Invariantes a Transformações Irrelevantes
- Insensíveis ao Ruído
- Úteis para a Discriminação pretendida

O conhecimento a priori do Sistema é muito importante

Desenho de Sistema de Reconhecimento de Padrões



Escolha de um Modelo

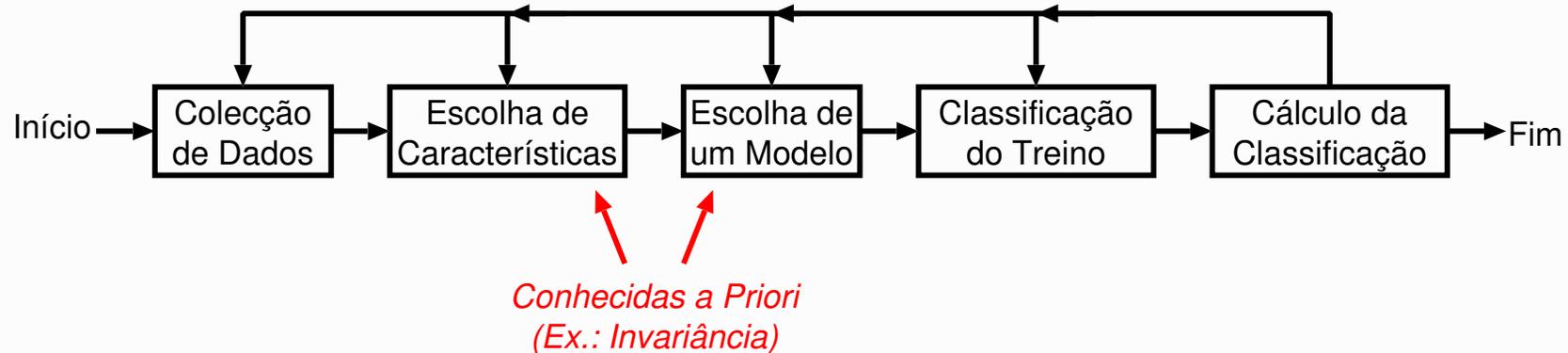
Treino

Cálculo de Classificação

Medida de qualidade do funcionamento do sistema

Permite identificar a necessidade de melhorar as suas diferentes componentes

Desenho de Sistema de Reconhecimento de Padrões

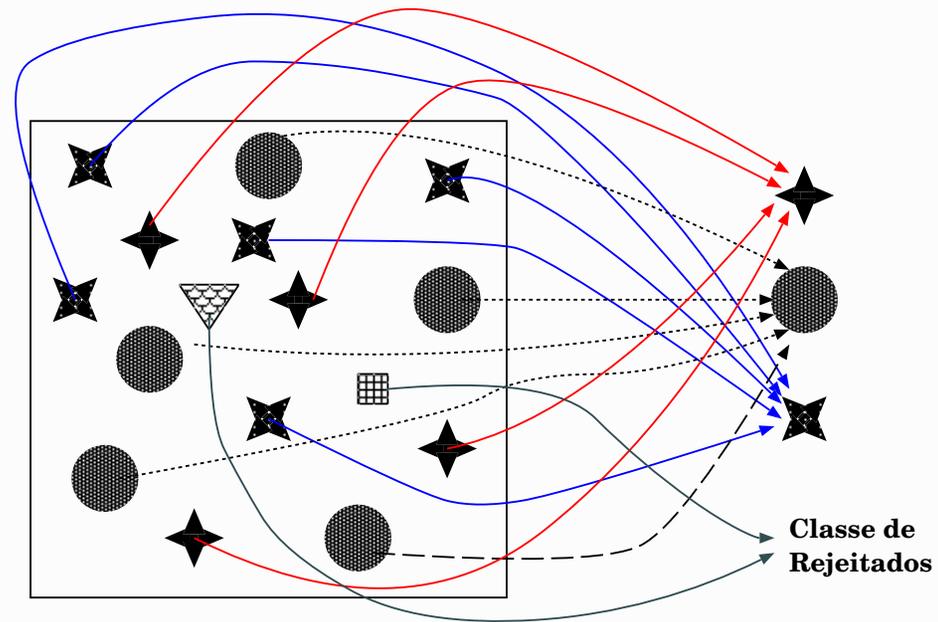


Reconhecimento e Adaptação

- Reconhecimento Supervisionado
 - Uso de *Protótipos*: A classificação é feita usando o *protótipo* mais próximo.
 - Estatístico: Usam-se funções de Densidade de Probabilidade, escolhendo-se a classificação mais provável para o vector \vec{x} .
 - Redes Neurais: Programada por um processo de aprendizagem que estabelece valores para os pesos.
- Reconhecimento não Supervisionado
- Reconhecimento Reforçado

Modelo Comum de Classificação

Classes



Modelo Comum de Classificação

Extractor de Características

Extrai a informação relevante ao processo de reconhecimento da informação lida pelo(s) sensor(s).

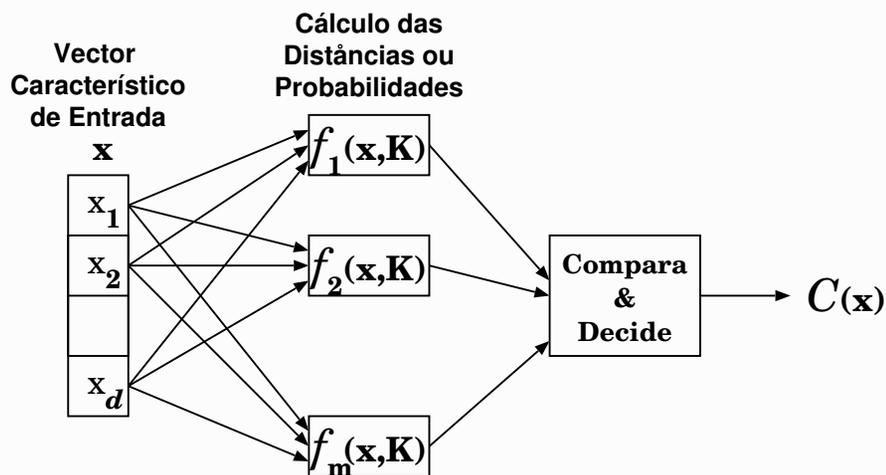
É nesta fase que algoritmos de processamento de imagem apropriados seriam usados para identificar Cores, Formas e Texturas.

Desta forma um conjunto de **CARACTERÍSTICAS** relevantes para a classificação ficam disponíveis.

Modelo Comum de Classificação

Classificador

Usa as **CARACTERÍSTICAS** extraídas para assignar ao objecto uma de m **CLASSES**, $C_1, C_2, \dots, C_{m-1}, C_m = C_r$
(C_r - Classes de Rejeitados)



- Um Vector Característico de Entrada \mathbf{x} de dimensão d representa o objecto a ser classificado.
- Cada *CLASSE* possível tem um bloco que processa o Vector Característico de Entrada \mathbf{x} e calcula uma medida do *OBJECTO* pertencer a essa *CLASSE*, usando algum conhecimento \mathbf{K} sobre a *CLASSE*.
- Um estágio final compara os m resultados e atribui uma *CLASSE* final ao *OBJECTO*.

Modelo Comum de Classificação

Cálculo do Erro do Sistema

A **TAXA DE ERRO** de um sistema de Classificação mede a qualidade do reconhecimento para que o sistema foi desenhado.

Definições:

Um **ERRO DE CLASSIFICAÇÃO** existe quando um objecto foi Classificado como pertencente à CLASSE C_i , quando a sua verdadeira CLASSE é C_j com $i \neq j$ e $C_i \neq C_r$.

Modelo Comum de Classificação

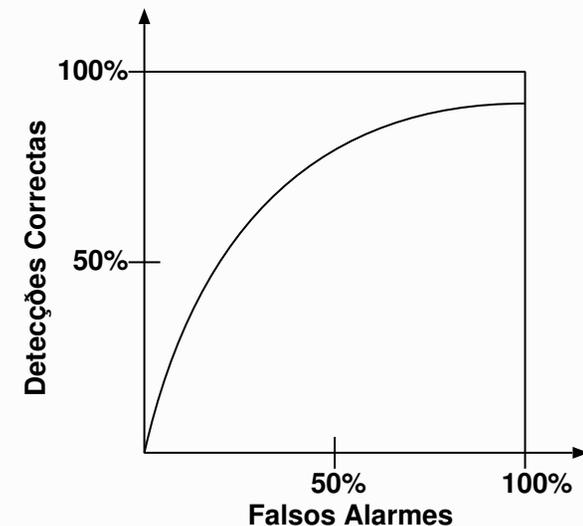
ROC

- Um **Falsos Positivo (Falso Alarme)** ocorre quando a uma CLASSE é atribuído um OBJECTO errado.
- Um **Falsos Negativo** ocorre quando a uma CLASSE não é atribuído um OBJECTO que lhe pertencia.

ROC - ("Receiver Operating Curve")

Representa graficamente a relação entre a Detecção Correcta e o Falso Alarme (Falso Positivo)

Normalmente um aumento do número de detecções correctas também leva a uma aumento do número de Falsos Alarmes.

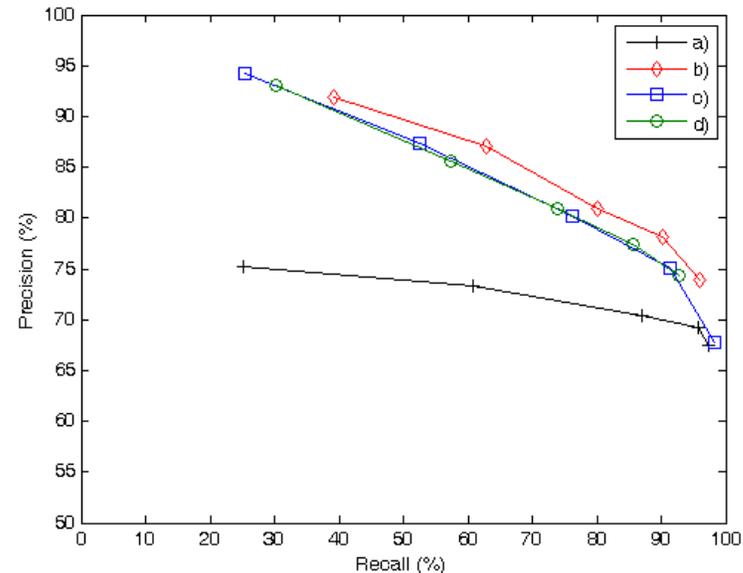


Modelo Comum de Classificação

Gráficos Precision versus Recall

$$\text{Sensibilidade: } Recall = \frac{TP}{TP + FN}$$

$$\text{Precisão: } Precision = \frac{TP}{TP + FP}$$



ROC - O objectivo de um sistema de classificação é ter um ROC com uma curva a passar no canto superior esquerdo do gráfico.

Precision versus Recall - O objectivo de um sistema de classificação é ter um gráfico "Precision versus Recall" com uma curva a passar no canto superior direito do gráfico.

Caracterização de Imagem

Sumário

- Introdução
- Caracterização de cor
- Caracterização de Texturas
- MPEG-7

Caracterização de Imagem

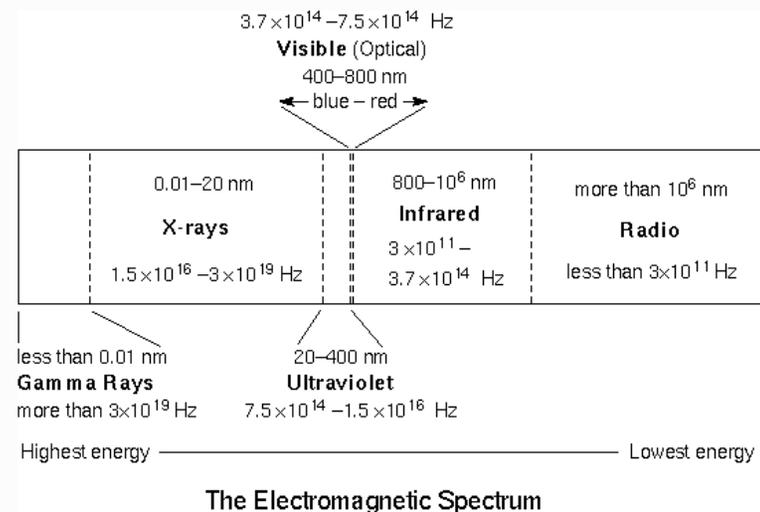
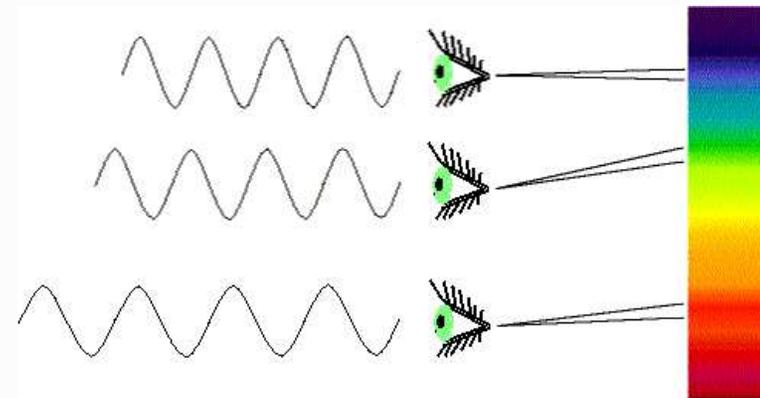
Caracterização de Cor

Física da Cor

RECEPTORES QUÍMICOS DO OLHO HUMANO SÃO SENSÍVEIS À RADIAÇÃO (LUMINOSA)

- COMPRIMENTOS DE ONDA (λ) $\in [400nm, 700nm]$ (VIOLETA AO VERMELHO);

LUZ BRANCA - COMPOSTA DE UMA ENERGIA APROXIMADAMENTE IGUAL EM TODOS OS COMPTRIMENTOS DE ONDA DO ESPECTRO VISÍVEL.



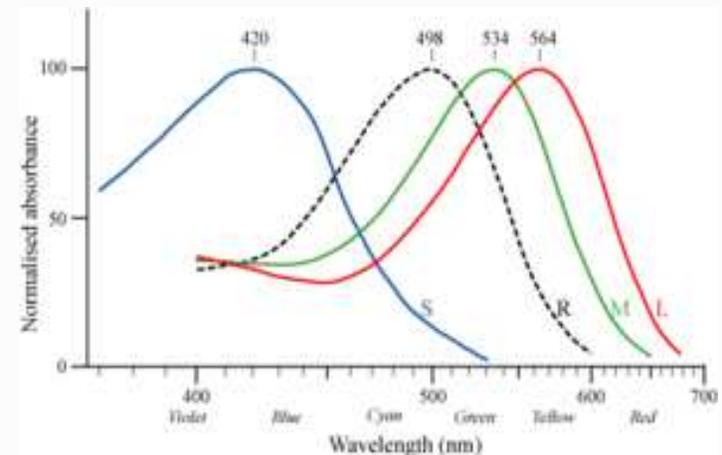
Caracterização de Imagem

Caracterização de Cor

Espaços de Cor - RGB

A percepção de cor humana resulta de três tipos de receptores (cones) sensíveis a três zonas espectrais (Vermelhos, Verdes e Azuis) - Visão de Cor Tricromática.

Tipo de Cone	Nome	Intervalo	Pico de sensibilidade
S	Azul	400..500 ηm	440 ηm
M	Verde	450..630 ηm	544 ηm
L	Vermelho	500..700 ηm	580 ηm



Uma determinada cor (comprimento de onda) estimula cada um destes cones com uma determinada intensidade.

Exemplo: Amarelo estimula fortemente os cones tipo L, moderadamente os tipo M e muito suavemente os tipo S.

Caracterização de Imagem

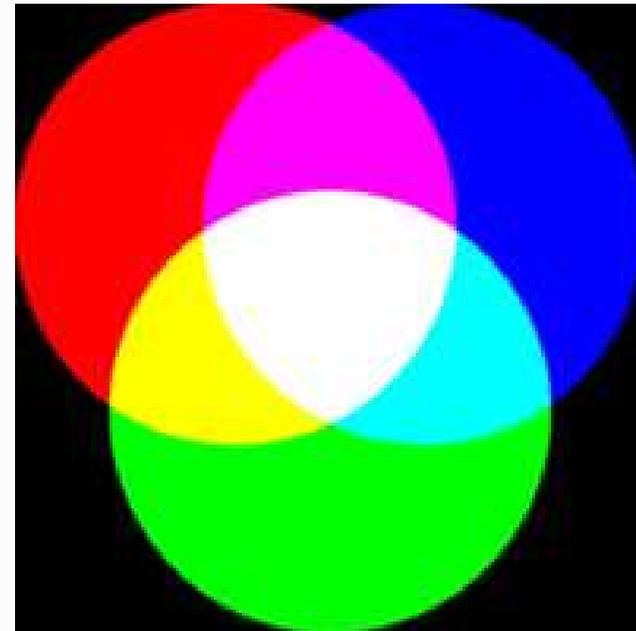
Caracterização de Cor

Espaços de Cor - RGB

Equipamentos de visualização de imagem a cores (TVs, Monitores de Computador, etc) usam um sistema de mistura de cores aditiva com cores primárias R-Vermelho, G- Verde, B-Azul.

Cada uma estimula os receptores respectivos do olho humano na quantidade pretendida.

EXEMPLO DE CORES RGB
CODIFICADAS COM 24 BITS:
◇ VERMELHO (255,0,0);
◇ AMARELO (255,255,0);
◇ BRANCO (255,255,255);
◇ PRETO (0,0,0);
◇ CINZENTO DE MÉDIA
INTENSIDADE (127,127,127);



Representação da Adição de Cores

Caracterização de Imagem

Caracterização de Cor

Espaços de Cor - RGB

RGB Normalizado:

$$r = \frac{R}{R + G + B} \quad g = \frac{G}{R + G + B} \quad b = \frac{B}{R + G + B}$$

Intensidade:

$$I = \frac{R + G + B}{3}$$

Caracterização de Imagem

Caracterização de Cor

Espaços de Cor - CIE XYZ

CIE - Comissão Internacional de Iluminação; Criado em 1931

XYZ Normalizado:

$$x = \frac{X}{X + Y + Z} \quad y = \frac{Y}{X + Y + Z} \quad z = \frac{Z}{X + Y + Z}$$

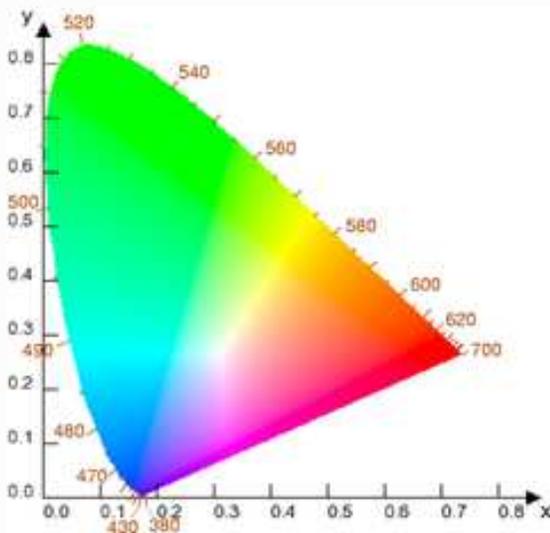
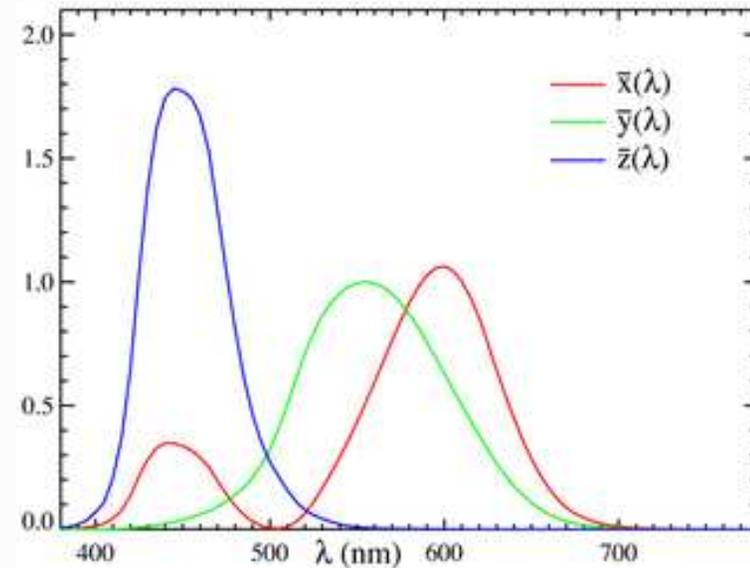


DIAGRAMA DE CROMATICIDADE CIE - XYZ



Funções Colorimétricas da Norma CIE - XYZ

Caracterização de Imagem

Caracterização de Cor

Espaços de Cor - HSV

H - Hue (tonalidade); S - Saturation; V - Value (Luminosidade)

Mais próximo da representação humana;

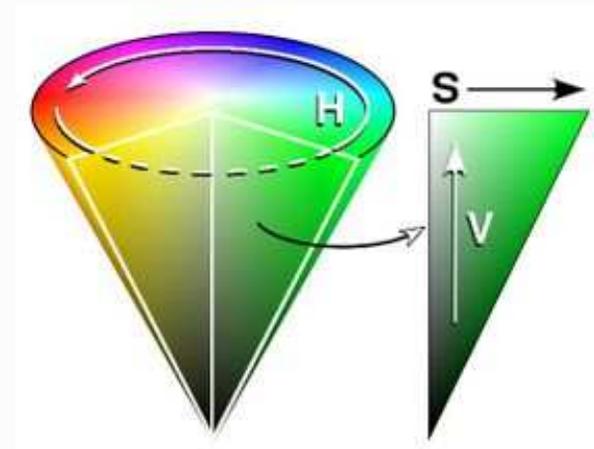
Transformação RGB para HSV

$$Max = Max\{r, g, b\} \quad Min = Min\{r, g, b\}$$

$$H = \begin{cases} 60 \frac{g-b}{Max-Min}, & \text{se } Max = r \text{ e } g \geq b \\ 60 \frac{g-b}{Max-Min} + 360, & \text{se } Max = r \text{ e } g < b \\ 60 \frac{g-b}{Max-Min} + 120, & \text{se } Max = g \\ 60 \frac{g-b}{Max-Min} + 240, & \text{se } Max = b \end{cases}$$

$$H \in \{0, 360\}^{\circ}$$

$$S = \frac{Max - Min}{Max} \in \{0, 1\} \quad V = Max \in \{0, 1\}$$



Espaço de Cores HSV

Caracterização de Imagem

Caracterização de Cor
Espaços de Cor - HSV



Efeito da Variação da Saturação numa Imagem



Efeito da Variação da Tonalidade numa Imagem

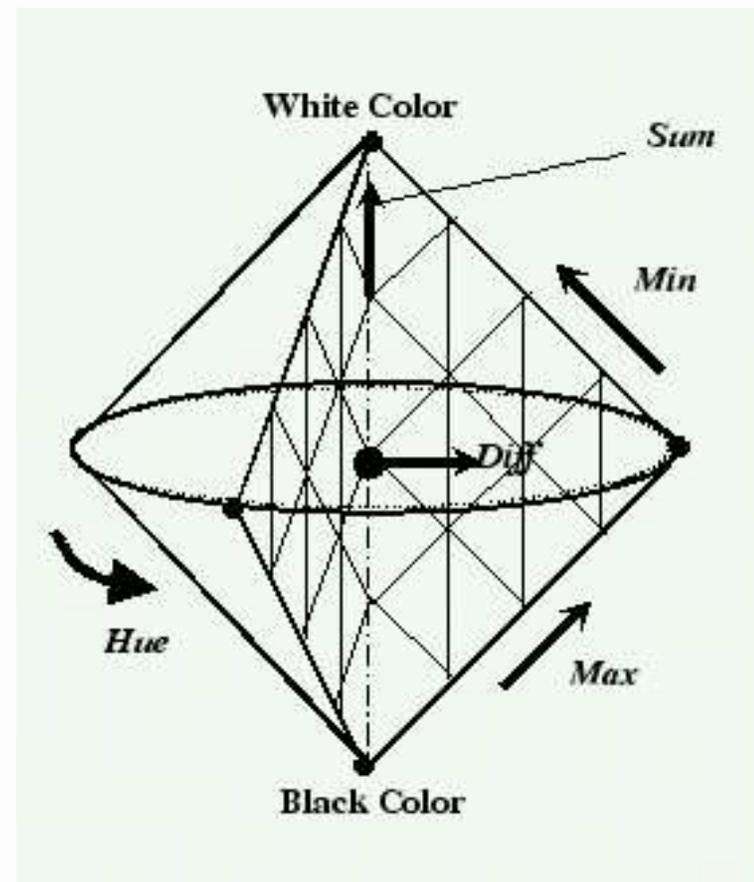
Caracterização de Imagem

Caracterização de Cor

Espaços de Cor - HMMD

HMMD: (Hue, Max, Min, Diff)

$Diff = Max - Min$



Representação do espaço HMMD

Caracterização de Imagem

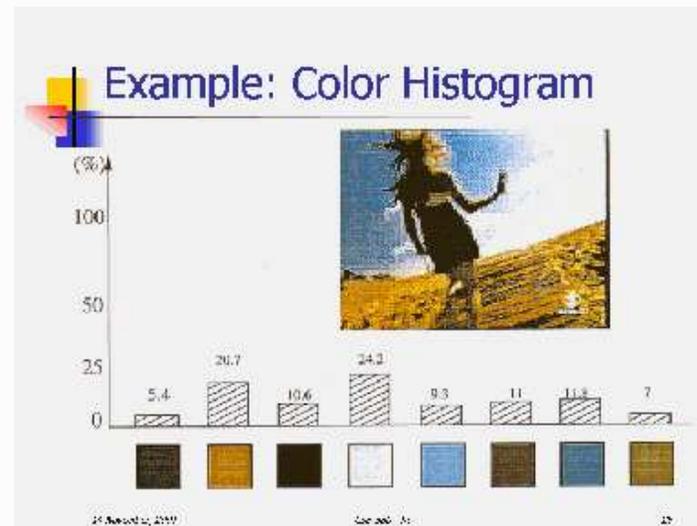
Caracterização de Cor

Histogramas de Cor

As cores são agrupadas por semelhança em “bins”.

O número de pixels com cores pertencentes a cada “bin” é contabilizado no histograma.

O espaço HSV é especialmente indicado para os histogramas de cor, pois cores semelhantes aparecem naturalmente próximas



Caracterização de Imagem

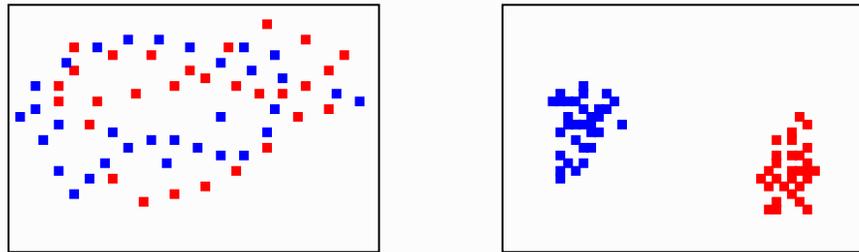
Caracterização de Cor

Descritor Estrutural de Cor

Uma janela (por exemplo de 8×8) move-se sobre a imagem.

Em vez de se aumentar o número de cores num histograma de cor, só se aumenta uma unidade cada vez que na sobreposição da janela se encontra uma ou mais vezes uma cor pertencente ao "bin" respectivo.

Este Descritor melhora o cálculo de similaridade em imagens reais.



Duas imagens com histograma de cores idêntico, mas com *Descritor Estrutural de Cor* diferentes.

Caracterização de Imagem

Caracterização de Textura

Exemplo de Texturas

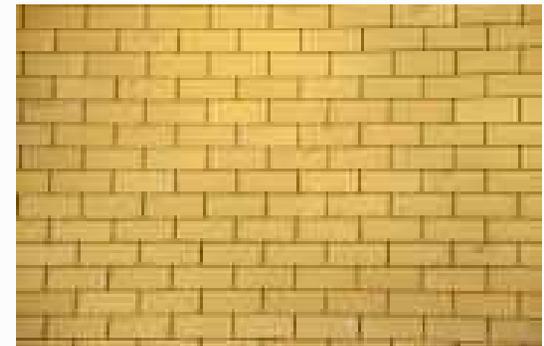


Exemplo de imagens identificáveis pela textura

Caracterização de Imagem

Caracterização de Textura

Exemplo de Texturas

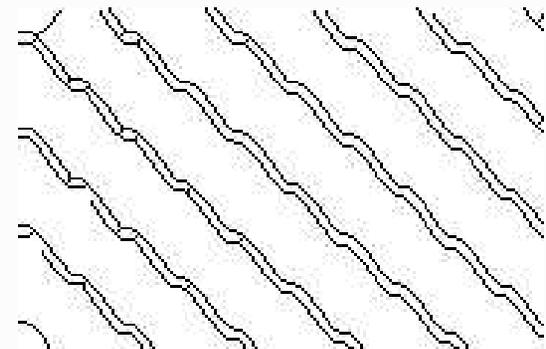
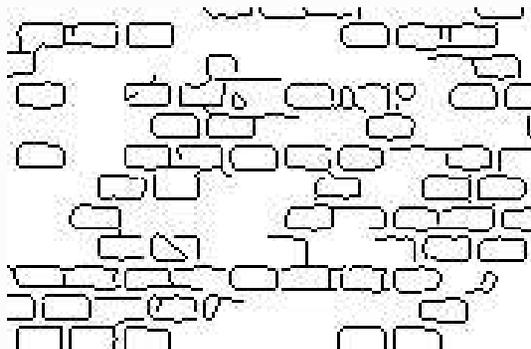
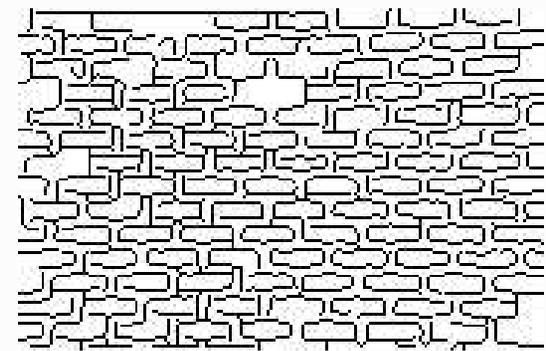
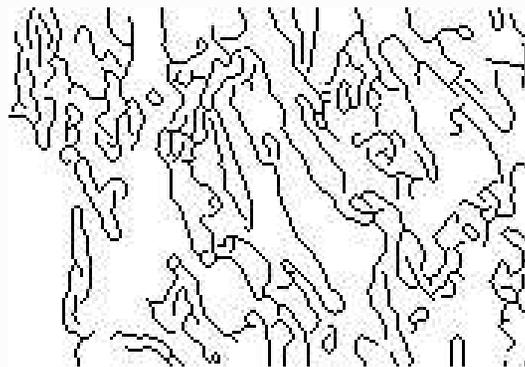


Exemplo de imagens com uma textura

Caracterização de Imagem

Caracterização de Textura

Descrição de Textura - Arestas da Imagem



Exemplo de arestas das imagens com uma textura

Caracterização de Imagem

Caracterização de Textura

Descrição de Textura - HISTOGRAMAS das ARESTAS -EHD

Existem diferentes métodos que descrevem a distribuição das arestas.

O mais relevante consiste no cálculo dos HISTOGRAMAS das ARESTAS.

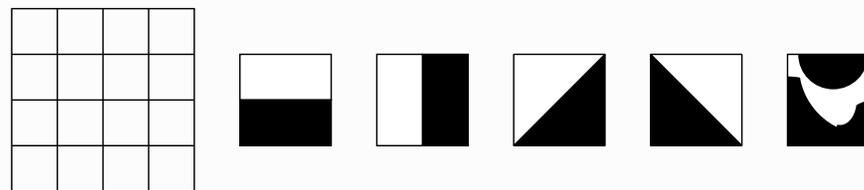
A imagem é dividida em sub-imagens de 4×4

Para cada pixel aresta é definido qual o tipo de aresta:

aresta vertical, horizontal, a 45° , a 135° ou outro.

Consoante a posição relativa na sub-imagem e o tipo de direcção da aresta podem-se definir:

$5 \text{ direcções} \times (4 \times 4) \text{ localizações de pixels na sub-imagem} = 80 \text{ situações possíveis.}$



Subimagem e Direcções possíveis das arestas

O HISTOGRAMAS das ARESTAS representa a contagem do número de vezes que acontece cada uma destas 80 situações possíveis numa imagem.

Caracterização de Imagem

Caracterização de Textura

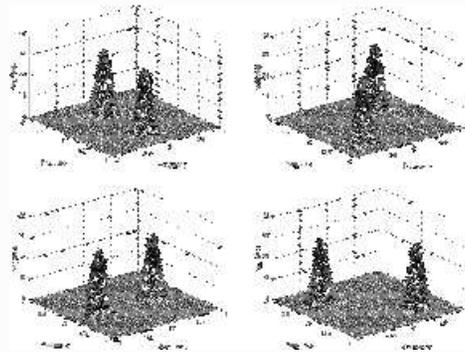
Descrição de Textura - Filtros de Gabor

Um Filtro de Gabor consiste de um filtro Gaussiano que é deslocado da frequência (0,0), para uma determinada frequência (V_c, U_c).

Consiste assim num filtro passa banda

Formulação Matemática:

$$h(y, x) = e^{-\frac{1}{2}\left(\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right)} e^{-j2\pi(u_0x + v_0y)} \quad \xleftrightarrow{TF} \quad H(v, u) = 2\pi\sigma_x\sigma_y e^{-2\pi^2[(u-u_0)^2\sigma_x^2 + (v-v_0)^2\sigma_y^2]}$$



FILTROS DE GABOR COM DIRECÇÕES 0, 45 , 90 E 135°.



Resultado da filtragem da imagem nas direcções 0, 45 , 90 e 135°.

Caracterização de Imagem

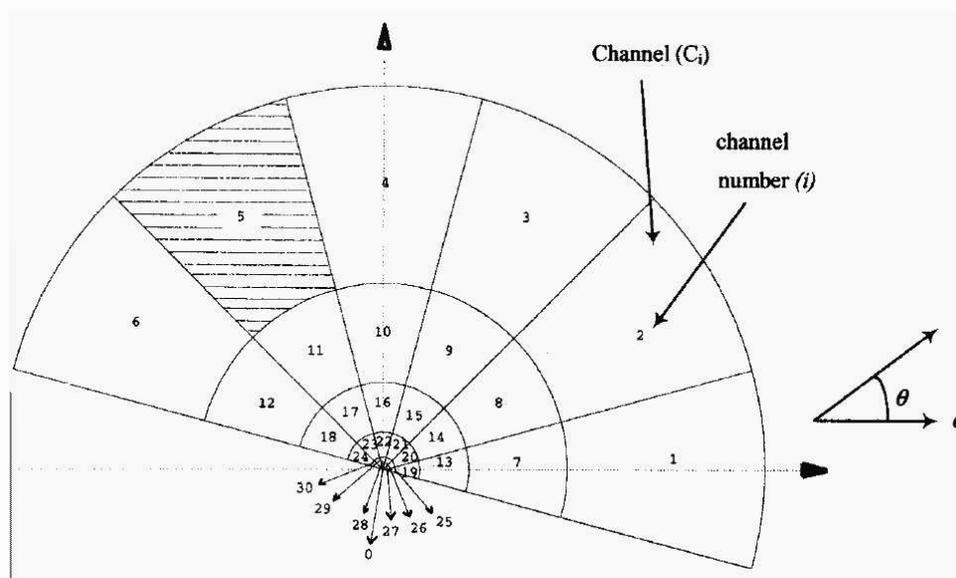
Caracterização de Textura

Descrição de Textura - Descritor de Textura Homogéneo - HTD

A imagem é filtrada por filtros de Gabor para 30 situações diferentes de (u_0, v_0) . Considerando:

$$\omega_0 = \sqrt{u_0^2 + v_0^2} \quad \text{com} \quad \omega_0 = \omega_M 2^{-s}, \quad s \in \{0, 1, 2, 3, 4\}$$

$$\theta_0 = \arctan(u_0/v_0) \quad \text{com} \quad \theta_0 \in \{0, 30^\circ, 60^\circ, 90^\circ, 120^\circ, 150^\circ\}$$



Caracterização de Imagem

Caracterização de Textura

Descrição de Textura - Descritor de Textura Homogéneo - HTD

Resultam então:

- a Imagem original;
- 30 imagens resultantes das filtragens de Gabor;

Calcula-se o valor médio μ e o desvio padrão σ de cada uma destas imagens, formando o vector característico:

$$(\mu_{orig}, \sigma_{orig}, \mu_1, \sigma_1, \mu_2, \sigma_2, \dots, \mu_{30}, \sigma_{30})$$

que é usado para comparação.

Descrição de Imagem baseada nas Características Locais

Grupo de novos Descritores de Imagem

- Descrevem a Imagem Localmente.
- Alguns baseiam-se na Detecção de pontos característicos
- Destacam-se:
 - SIFT - Scale-invariant feature transform (Lowe)
 - SURF - Speeded Up Robust Features (Bay et al)
 - HOG - Histogram of oriented gradients (Dalal & Triggs)

SIFT - Scale-Invariant Feature Transform

Resistente a:

- Rotação de Imagem;
- Deslocamento e Ampliação/Redução de Imagem;
- Variação de Iluminação.

SIFT - Scale-Invariant Feature Transform

Algoritmo

- Keypoint resultam da Detecção de extremos no espaço e na escala: (DOG - Difference of Gaussians - Aproximação do Laplaciano):
 $D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\sigma)$ onde $L(x, y, k\sigma) = G(x, y, k\sigma) \oplus I(x, y)$
 - Diferentes escalas $k\sigma$ são obtidas para cada oitava ($k_i = 2 \times k_j$);
- Interpolação com dados próximos para obtenção de localização mais precisa.
 - Cálculo da localização usando a aproximação quadrática da expansão da série de Taylor da DOG:

$$D(\mathbf{x}) = D + \frac{\partial D^T}{\partial \mathbf{x}} \mathbf{x} + \frac{1}{2} \mathbf{x}^T \frac{\partial^2 D^T}{\partial \mathbf{x}^2} \mathbf{x}$$

sendo D e as suas derivadas calculadas em cada pixel candidato a keypoint e sendo $\mathbf{x} = (x, y, k\sigma)$ o deslocamento (offset) desde esse ponto candidato.

- A localização do keypoint $\hat{\mathbf{x}}$ resulta por derivação de D e fazendo-a igual a zero em relação a \mathbf{x}
- Se $\hat{\mathbf{x}}$ for maior que 0.5 em qualquer dimensão considera-se ser uma indicação de que esse extremo fica próximo de outro candidato.
- Nesse caso o keypoint move-se para $\hat{\mathbf{x}}$, e o mesmo tipo de análise é repetido.

SIFT - Scale-Invariant Feature Transform

Algoritmo (continuação)

- Uso da Curvatura Principal para seleccionar extremos importantes
Eliminação dos que têm pouco contraste

- Calculo da matriz de Hessian:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{yx} & D_{yy} \end{bmatrix}$$

- Os valores próprios λ_1 e λ_2 , com $\lambda_1 > \lambda_2$ resultam em $r = \lambda_1/\lambda_2$
- Definindo

$$R = \frac{D_{xx} + D_{yy}}{|H|} = \frac{(r + 1)^2}{r}$$

com $|H| = D_{xx}D_{yy} - D_{xy}^2$ (Determinante de H).

- pontos com $r < r_{th}$ não são considerados (Usualmente $r_{th} = 10$)

SIFT - Scale-Invariant Feature Transform

Algoritmo (continuação)

4. • Attribuição de Orientação:
- Módulo:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) + L(x, y-1))^2}$$

- Orientação:

$$\theta(x, y) = \tan^{-1} \frac{L(x, y+1) + L(x, y-1)}{L(x+1, y) - L(x-1, y)}$$

- Cada pixel numa vizinhança do “keypoint” detectado antes contribui para o cálculo do histograma das direcções.
- São calculados 36 bins de orientação (10 graus cada)
- cada pixel contribui para o seu bin de orientação adicionando o Módulo ponderado (multiplicado) por uma janela gaussiana com $1.5 \times \sigma$, sendo σ a escala da detecção “keypoint”
 $\rightarrow L(x - x_k, y - y_k, 1.5 \times \sigma) \times m(x, y)$

SIFT - Scale-Invariant Feature Transform

Algoritmo (continuação)

5. Descritor do “Keypoint”

- É criado um conjunto de histogramas de orientação numa vizinhança de 4×4 pixels com 8 bins cada
- Estes histogramas são calculados a partir do módulo e orientação de amostras de regiões 16×16 em volta do “keypoint” tal que cada histograma contem amostras de 4×4 sub-regiões da região original
- O descritor final para cada “keypoint” tem 128 elementos = $4 \times 4 \times 8$ bins
- O vector é depois normalizado para a unidade
- depois é aplicado um threshold de 0.2 (redução de iefeito de iluminação não uniforme)
- segue-se nova normalização para a unidade

Similaridade entre dois Vectors Característicos

Para medir a Similaridade (ou Proximidade) entre dois vectores de dois objectos diferentes é muito vulgar usar a **DISTÂNCIA** entre os dois. Esta, compara cada componente do Vector Característico de um dos objectos com a componente respectiva do Vector Característico do outro objecto. (Cada componente mede uma característica; Por exemplo, a primeira componente mede a cor, a segunda componente é uma medida de uma textura específica, etc)

A **DISTÂNCIA DE MANHATTAN** entre dois Vectors Característicos de dimensão d , \mathbf{x}_1 e \mathbf{x}_2 , é dada por:

$$\|\mathbf{x}_1 - \mathbf{x}_2\| = \sum_{i=1}^d |\mathbf{x}_1[i] - \mathbf{x}_2[i]|$$

A **DISTÂNCIA EUCLIDIANA** entre dois Vectors Característicos de dimensão d , \mathbf{x}_1 e \mathbf{x}_2 , é dada por:

$$\|\mathbf{x}_1 - \mathbf{x}_2\| = \sqrt{\sum_{i=1}^d (\mathbf{x}_1[i] - \mathbf{x}_2[i])^2}$$

Similaridade entre dois Vectors Característicos

Distância Euclideana Ponderada

Alternativamente, a classificação pode usar as Distâncias ponderadas.

A **DISTÂNCIA PONDERADA** entre dois Vectors Característicos usa como factor de ponderação o desvio padrão dentro da classe em causa

$$\|\mathbf{x} - \mathbf{x}_c\| = \sqrt{\sum_{i=1}^d \left(\frac{\mathbf{x}[i] - \mathbf{x}_c[i]}{\sigma_i} \right)^2}$$

Classificadores

Surtem diferentes modelos de classificação:

- Classificação usando o VIZINHO mais PRÓXIMO.
- Classificação usando a MÉDIA da CLASSE mais PRÓXIMA.
- Classificação por KNN.
- Classificação “Bayesiana- por MAXIMIZAÇÃO da PROBABILIDADE a POSTERIORI.
- Classificação usando uma REDE NEURONAL ARTIFICIAL.
- Classificação usando Support Vector Machines (SVM).
- Classificação usando Particle Swarm Optimisation (PSO).

Classificadores

Classificadores

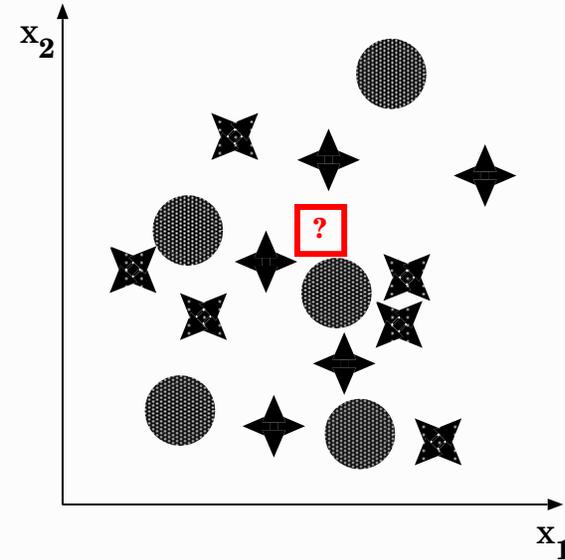
- Podem fazer classificação:
 - *Binária* \Rightarrow Classificam SIM/NÃO.
Múltiplas CLASSES usam vários classificadores binários.
 - *Múltipla*
Classificador atribui uma CLASSE de um conjunto predefinido.
Tem que se considerar sempre uma classe de rejeitados.
- Necessitam conjunto de TREINO para as diferentes CLASSES.
Num classificador BINÁRIO vão existir:
 - Conjunto de TREINO POSITIVO
 - Conjunto de TREINO NEGATIVO

Classificadores

Classificação usando o VIZINHO mais PRÓXIMO

Atribui-se a CLASSE da Amostra de Treino que origina a menor **DISTÂNCIA**.

Nota: Muitas vezes, quando as amostras de treino aparecem muito misturadas, não aparecendo regiões de classes bem definidas, torna-se conveniente usar este método de Classificação.



Classificadores

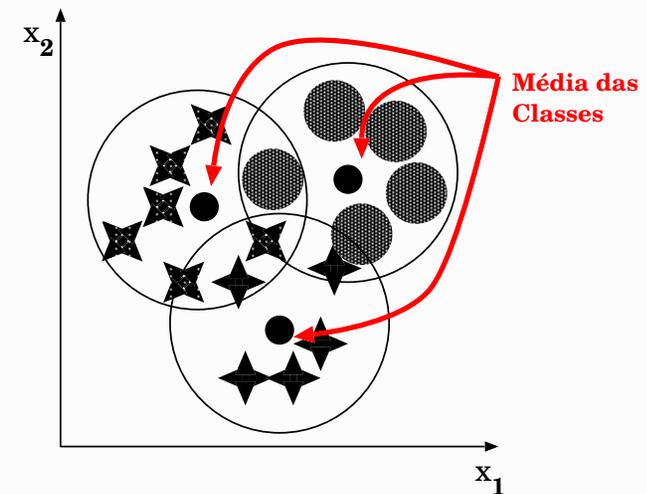
Classificação usando a MÉDIA da CLASSE mais PRÓXIMA

Usando n_i amostras de *treino* para uma dada classe i pode-se calcular a média:

$$\bar{\mathbf{x}}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \mathbf{x}_{i,j}$$

(Nota: corresponde a calcular o valor médio para cada componente do vector)

Uma amostra é classificada como pertencendo a uma dada CLASSE se a **DISTÂNCIA** ao seu centro for a menor de todas.



Classificadores

Classificação por KNN

Considera-se um conjunto de treino com N vectores característicos, representativos das diferentes classes.

Obtêm-se os K vectores característicos do conjunto de treino mais próximos do vector característico do elemento a classificar.

A classe que estiver em maioria nos K vectores de treinos é seleccionada.

Particularmente, o uso deste método em classificação binária leva a resultados bastante positivos.

Pode ser atribuído um intervalo de confiança à decisão do vector ser de uma classe C_i :

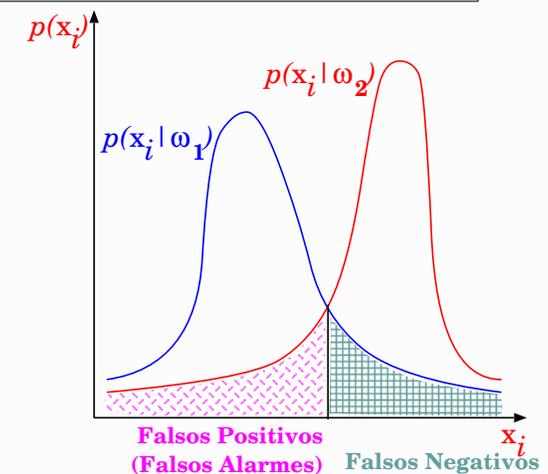
$$\frac{\text{número de vectores de treino da classe } C_i \text{ em } K}{K}$$

Classificadores

Decisão Bayesiana

Um **CLASSIFICADOR BAYESIANO** Classifica um objecto na classe a que é mais provável pertencer baseado nas Características observadas.

Classe j	ω_j
Vector Característico	\mathbf{x}
Distribuição Condicional da Classe	$p(\mathbf{x} \omega_j)$
Probabilidade a Priori	$P(\omega_j)$
Distribuição Incondicional	$p(\mathbf{x})$



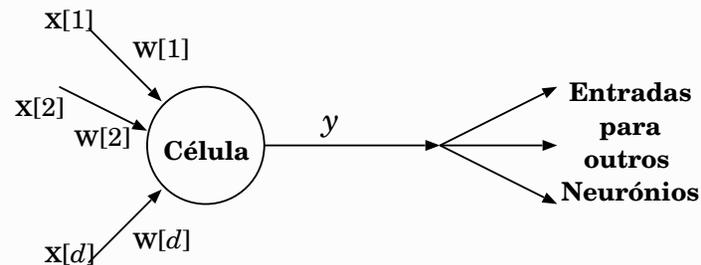
PROBABILIDADE À POSTERIORI
$$P(\omega_j | \mathbf{x}) = \frac{p(\mathbf{x} | \omega_j) P(\omega_j)}{p(\mathbf{x})}$$

onde
$$p(\mathbf{x}) = \sum_{i=1}^m p(\mathbf{x} | \omega_j) P(\omega_j)$$

Classificadores

Redes Neurais

Baseadas no modelo simplificado de um neurónio humano representado na figura.



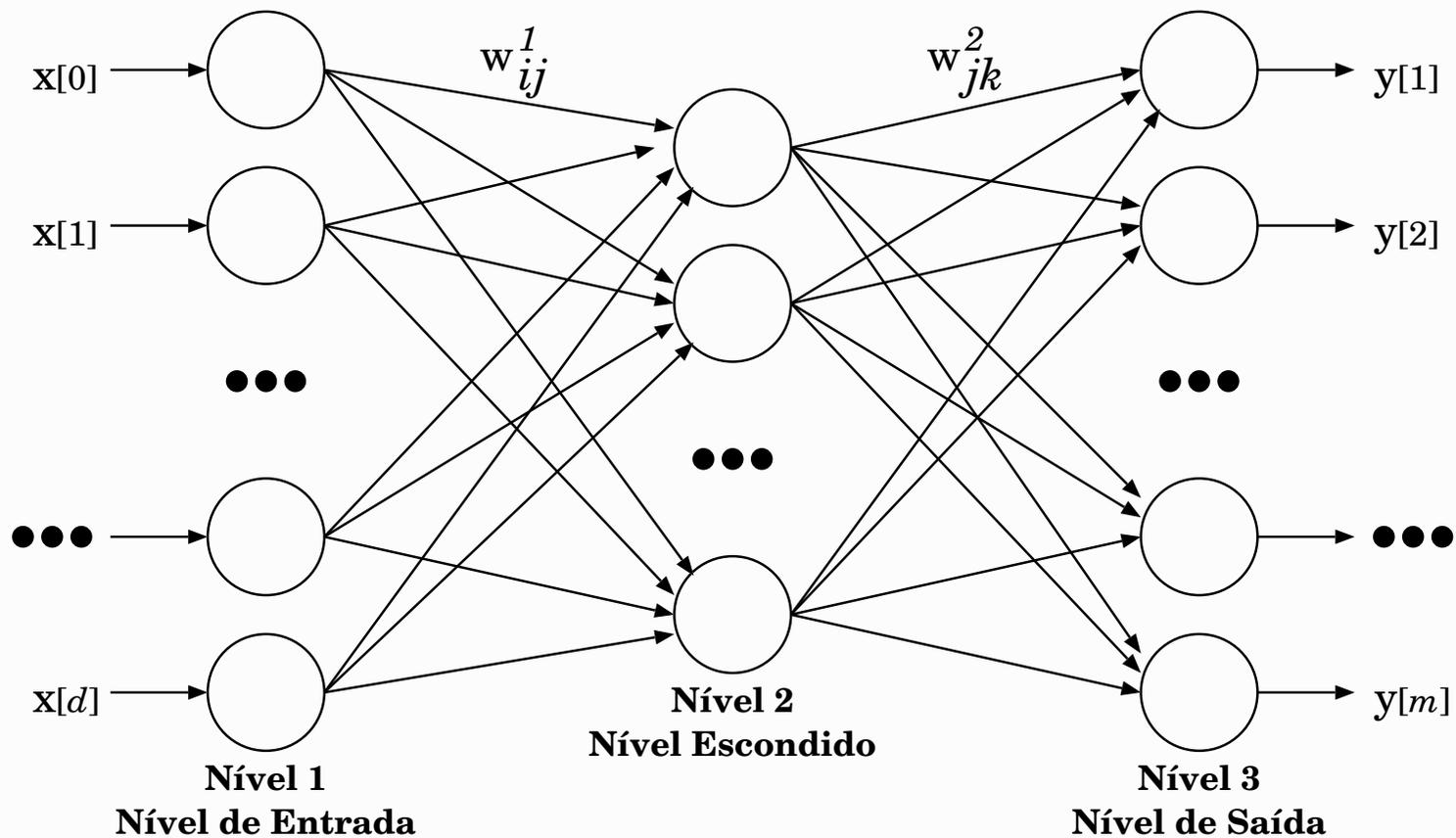
$$y = g \left(\sum_{j=1}^d w[j]x[j] \right)$$

onde $g(\alpha)$ pode ser dado por:

- $g(\alpha) = 1$ se $\alpha > t$ e 0 caso contrário
- $g(\alpha) = 1/(1 + e^{-\beta(\alpha-t)})$

Classificadores

Redes Neurais



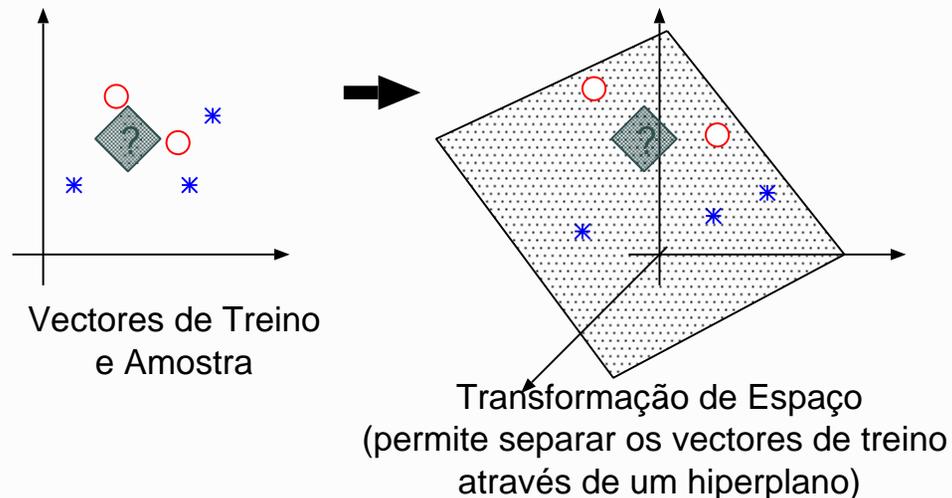
Classificadores

SVM - support Vector Machine

O classificador SVM é um classificador linear.

Mapeia os vectores descritores num espaço de maior dimensão (que até pode ter infinitas dimensões) de forma a conseguir um hiperplano de separação entre classes.

Esse mapeamento é feito a partir dos conjuntos de treino.



Classificação Não-supervisionada

“Clustering”

- Decisão é feita exclusivamente a partir dos dados usando função predefinida de f .
 - Baseado em estruturas naturais dos dados - Ex.: “*Clustering*”.
 - Algoritmo de “*Clustering K-means*”.

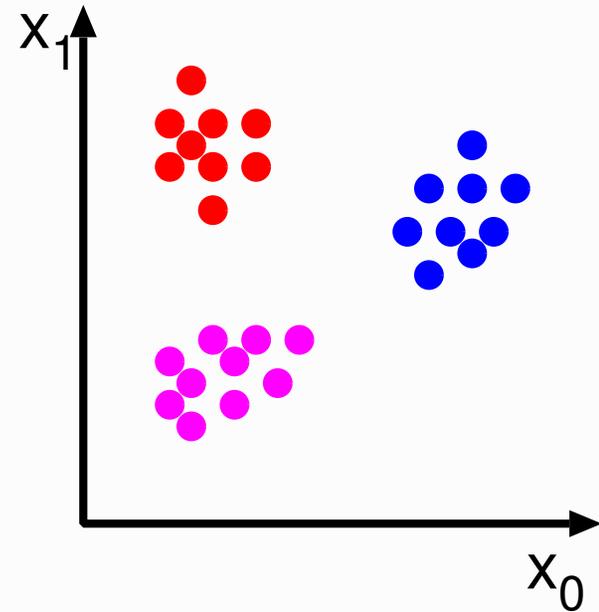
Classificação Não-supervisionada

Noção de "Clustering"

Processo de partição dos vectores característicos em subconjuntos, chamados "*Clusters*".

Uma forma normal de formar "*Clusters*" é associar pontos que estão próximos entre si no espaço euclideo considerado.

Pretende-se portanto, criar partições de um conjunto de vectores, em grupos que apresentam valores similares.



EXEMPLOS DE "*Clusters*" NUM ESPAÇO BIDIMENSIONAL.

Classificação Não-supervisionada

Algoritmos Clássicos de “Clustering”

Os vectores característicos, podem incluir componentes como:

- ◇ Valores de Intensidade
- ◇ Valores das componentes de Cor (RGB, HSV,...)
- ◇ Propriedades calculadas
- ◇ Medidas de Texturas

A escolha dos **CLUSTERS** pode ser definida

- ◇ com base no número de “Clusters”, K .
- ◇ mantendo a variância para cada “Clusters”, abaixo de um determinado valor.

Classificação Não-supervisionada

Algoritmos Clássicos de “Clustering”

Tipicamente existem K clusters, C_1, C_2, \dots, C_K , com médias m_1, m_2, \dots, m_K .

A medida do *erro quadrático mínimo* pode ser definida como:

$$D = \sum_{k=1}^K \sum_{x_i \in C_k} \|x_i - m_k\|^2,$$

que mede a proximidade dos dados aos clusters que lhe foram atribuídos.

ALGORITMO BASE

- ◇ Pixels são agrupados em “Clusters”.
- ◇ Um algoritmo de Etiquetagem permite encontrar regiões conectadas.

Classificação Não-supervisionada

Algoritmo de “Clustering” por “K-means” Iterativo

1. Fazer o número de iterações $i_C = 1$.
2. Escolher aleatoriamente um conjunto de K “Clusters” com médias $m_1(1), m_2(1), \dots, m_K(1)$.
3. Para cada vector x_i calcular $D(x_i, m_k(i_C))$, para cada $k = 1, 2, \dots, K$ e atribuir x_i ao cluster C_j com a média mais próxima.
4. Incrementar i_C , somando 1, e actualizar as médias para obter um novo conjunto $m_1(i_C), m_2(i_C), \dots, m_K(i_C)$.
5. Repetir os passos 3. e 4. até que $C_k(i_C) = C_k(i_C + 1)$ para todos os k .

NOTA: O algoritmo é convergente, embora possa não levar à solução óptima.

Usualmente para-se quando $|C_k(i_C) - C_k(i_C + 1)|$ é menor que um determinado limiar.

Classificação Não-supervisionada

Algoritmo de “Clustering” por “K-means” Iterativo



Original



Máscara

Exemplo de “Clustering” por “K-means”

“Bag of Words”

- Forma-se uma matriz em que cada linha corresponde a um descritor (Ex: SIFT, HOG, SURF)
- Calcula-se o KMEANS dessa matriz, que assim agrupa descritores por proximidade (Ex: K=300)
- Calcula-se o número de ocorrências de cada “Cluster” - “Bag of Words”
(Faz-se um histograma com o número de ocorrências)
- Usa-se este “**Bag of Words**”(histograma) como descritor de imagem
- Aplica-se os métodos de classificação a estes descritores de imagem “Bag of Words”