

Processamento de Imagens

Prof. Marcelo Portes de Albuquerque
Prof. Márcio Portes de Albuquerque

Monitores: Fernanda Dutra Moraes (CBPF)
Pedro de Souza Asad (CBPF)

IX Escola do CBPF – 2012

Organização do Curso

Aulas – 23 a 27 de Julho

Seg. – Introdução a Análise de Imagens

Ter . – Segmentação e Morfologia Matemática

Qua. – Classificação e Reconhecimento

Qui. – Laboratório (ImageJ e OpenCV)

Sex. – Laboratório (ImageJ e OpenCV) – Apresentação GPU/Cuda

Site Web:

Notas de Aula: <http://mesonpi.cat.cbpf.br/e2012>

Organização do Curso

The screenshot shows the website for the IX Escola do CBPF, held in Rio de Janeiro from July 16 to 27, 2012. The main navigation menu includes: Principal, Natureza/Estrutura, Informações/Datas, Localização, and Contato. The left sidebar contains several categories: Programação Principal (Graduação, Pós-Graduação, Palestras, Horários), Eventos Satélites (PROFCEM, ENAF), Inscrições, Geral (Organizadores, Palestrantes), and Novidades (FAQ, Últimas, Novidades). The Administrative section has a login form with fields for 'Usuário:' and 'Senha:', and a 'Login' button. The main content area is titled 'G06. Processamento de Imagens' and lists professors Márcio Portes de Albuquerque and Marcelo Portes de Albuquerque, and monitors Fernanda Dutra Moraes and Pedro de Souza Asad. The course syllabus (Ementa) includes: Introduction and basic concepts, Image segmentation and mathematical morphology, Pattern recognition, Image processing software (ImageJ and OpenCV), and GPU techniques in image processing. A 'Material do Curso' section provides links to exercise lists for Aulas 1 and 2, and a PDF of a book on 'Análise de Imagens e Visão Computacional'. The footer of the website reads: 'IX Escola do CBPF - Copyright © CAT/CBPF/MCTI - versão PHP - 13 de junho de 2012'. A Facebook logo for CBPF is also visible in the bottom left corner of the website screenshot.

Site Web:

Notas de Aula: <http://mesonpi.cat.cbpf.br/e2012>

Informações IX Escola CBPF

- A Escola fornecerá certificado de **frequência** aos alunos inscritos que tiverem assistido **no mínimo 8 horas** de aula de um determinado curso de graduação, e no mínimo 12 horas, no caso dos cursos de pós-graduação.
- Além disto, a Escola **exige freqüência** integral dos estudantes que **beneficiam de auxílio financeiro** (hospedagem e alimentação), em 2 cursos no caso dos estudantes de graduação e em 3 cursos, dos estudantes de pós-graduação.
- Pelas duas razões acima, é necessário que os alunos **assinem** as listas de presença a cada aula.

Sumário Aula 03

I - Introdução ao Reconhecimento de Padrões

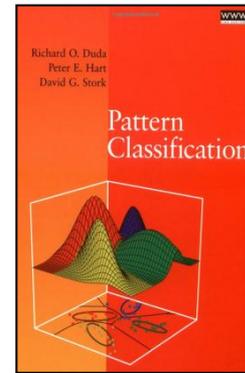
1. Definição de Padrão e Classe
2. Vetor de características
3. Exemplo
4. Projeto do classificador
5. Abordagens típicas
6. Exemplos

II – Introdução as Redes Neurais Artificiais

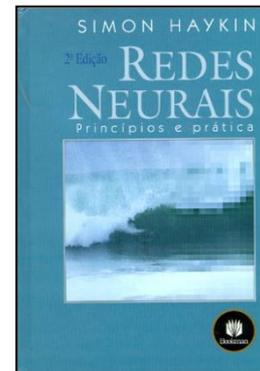
1. Definição
2. Modelo de um neurônio artificial
3. Modelo de uma rede MLP
4. Algoritmo Backpropagation
5. Exercícios

Referências Bibliográficas

Pattern Classification (2nd Edition)
**Richard O. Duda, Peter E. Hart,
David G. Stork**



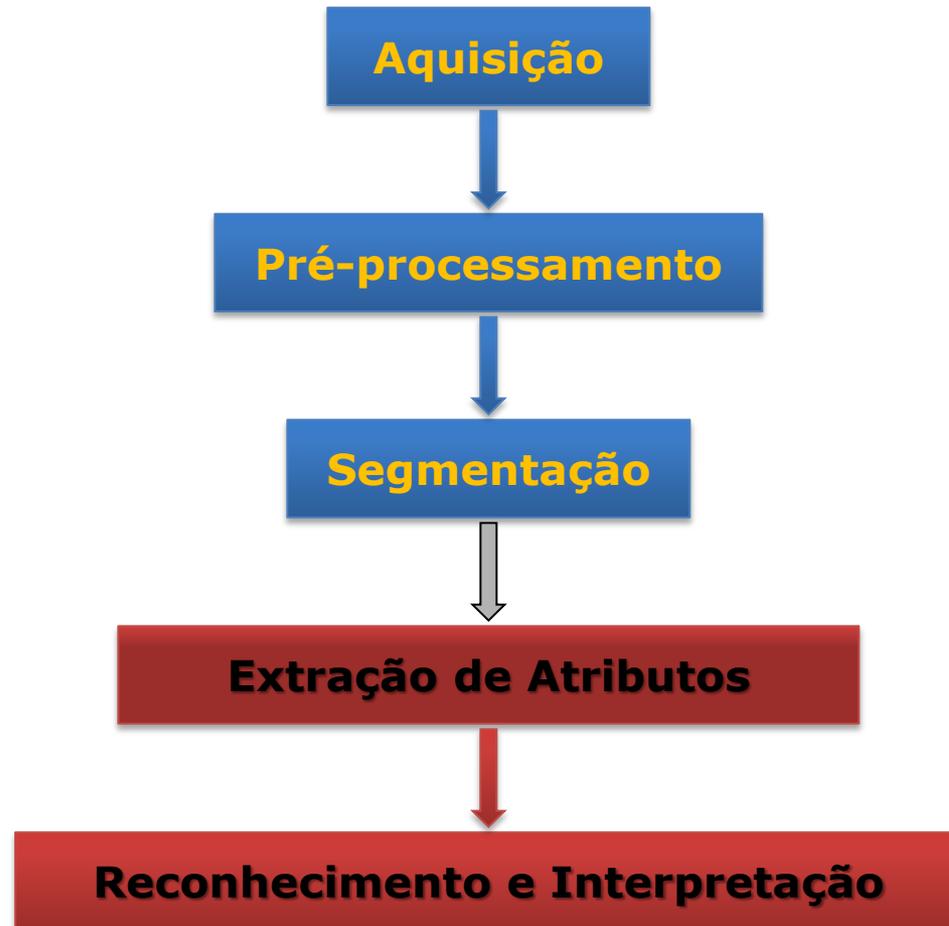
Redes Neurais: Princípios e Prática
Simon Haykin
(Traduzido para o português)



PARTE I

Introdução ao Reconhecimento de Padrões

Fases de Análise e Processamento



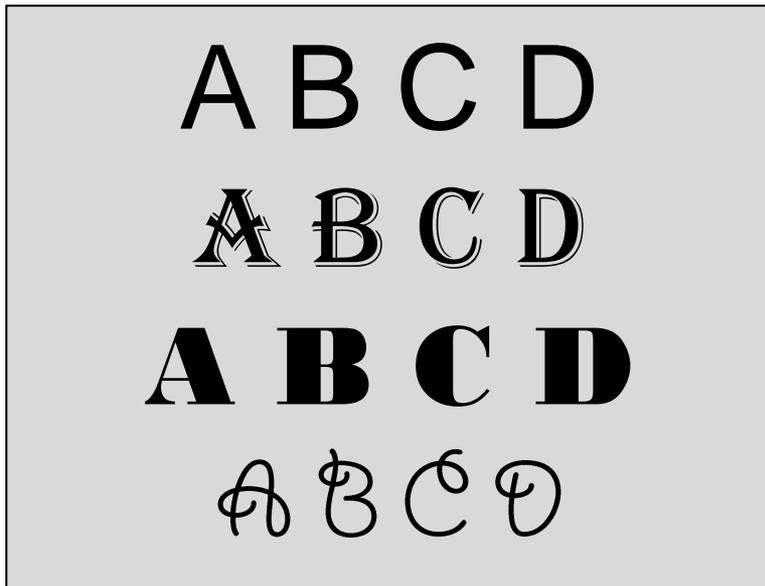
O que é um Padrão?

Entidade à qual se pode dar um nome

Ex. Objeto, Rosto Humano, Sinal de voz

Um padrão é avaliado de forma física e subjetiva

Caracteres em Imagens



Peças em Imagens



O que é uma Classe?

Classe: um modelo matemático descrito por várias características que servem para identificar um padrão

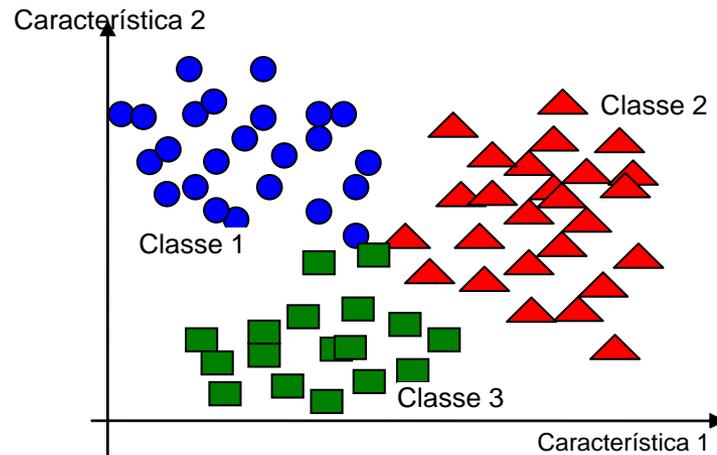
A característica é qualquer aspecto, qualitativo ou quantitativo.

A característica pode ser simbólica (ex. cor) ou numérica (ex. altura)

A combinação de d características é representada como um vetor de dimensão d chamado de vetor de características – *feature vector*,

O espaço de d -dimensões formado pelo vetor de característica é chamado de espaço de características – *feature space*

Os objetos são representados como pontos no espaço de características.



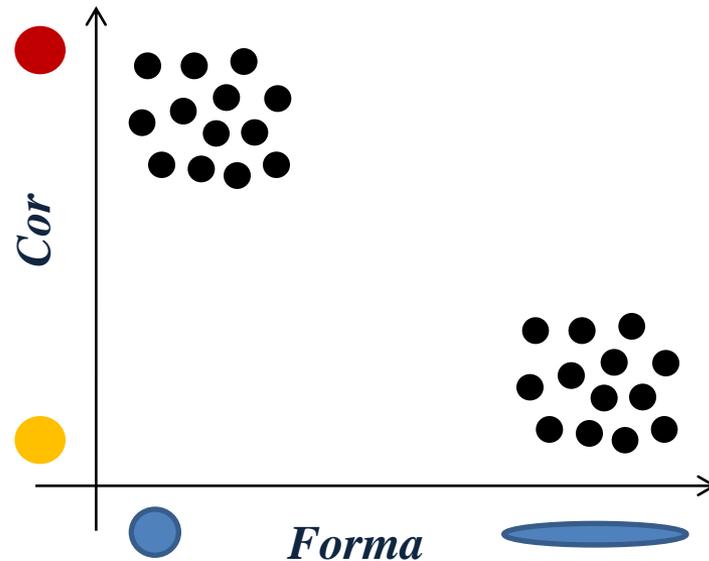
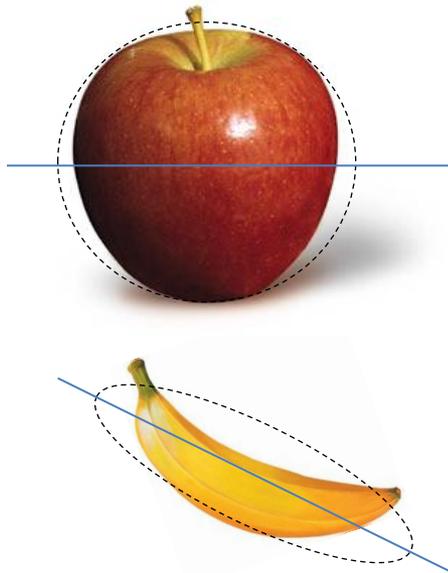
O Que é Reconhecimento de Padrões ?

- *“Atribuir a um objeto ou evento físico, uma das diversas categorias pré-especificadas”* -- Duda & Cervo.
- *“Problema de estimar funções da densidade em um espaço de dimensão elevada, dividindo o espaço em regiões de categorias ou das classes”* – Fukunaga.
- *“Dado alguns exemplos de sinais complexos e de decisões corretas para eles, fazer decisões automaticamente para uma sequência dos exemplos futuros”* – Ripley.
- *“Ciência que concerne a descrição ou a classificação das medidas”* – Schalkoff.
- *“O processo de dar nomes w às observações x ”* – Schürmann.
- *“Reconhecimento de padrões se dedica a fazer uma máquina responder à pergunta: o que é isto?”* – Morse.
- *Reconhecimento de padrões é um sub-tópico da aprendizagem de máquina cujo objetivo é classificar informações (padrões) baseado ou em conhecimento a priori ou em informações estatísticas extraídas dos padrões.* – Wikipédia, 2012.

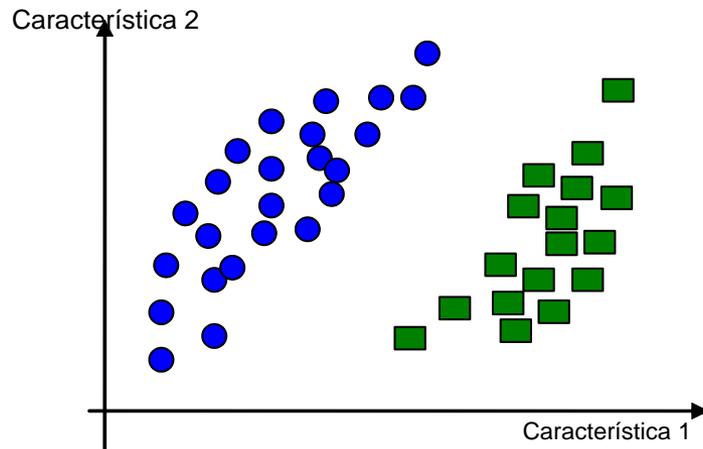
Objetivos do Reconhecimento de Padrões

Identificar um padrão como membro de um conjunto conhecido de classes -
classificação

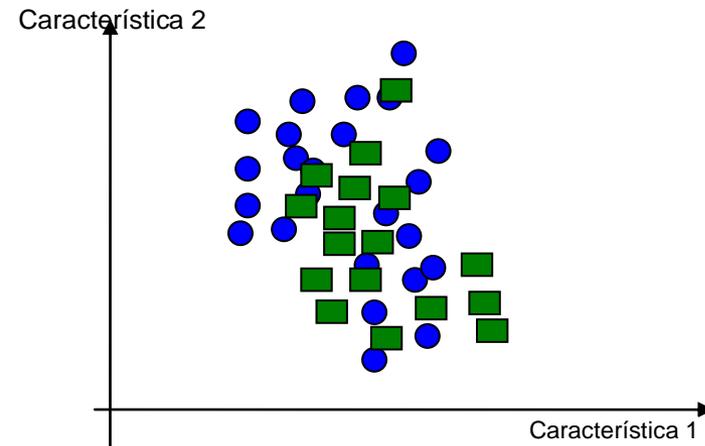
Atribuir um padrão a um conjunto de classes de padrões – clustering
(agrupamento)



Um “bom” Vetor de Características

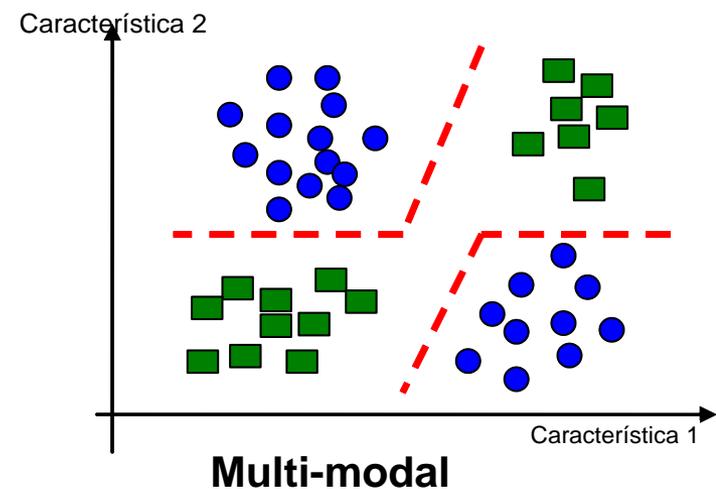
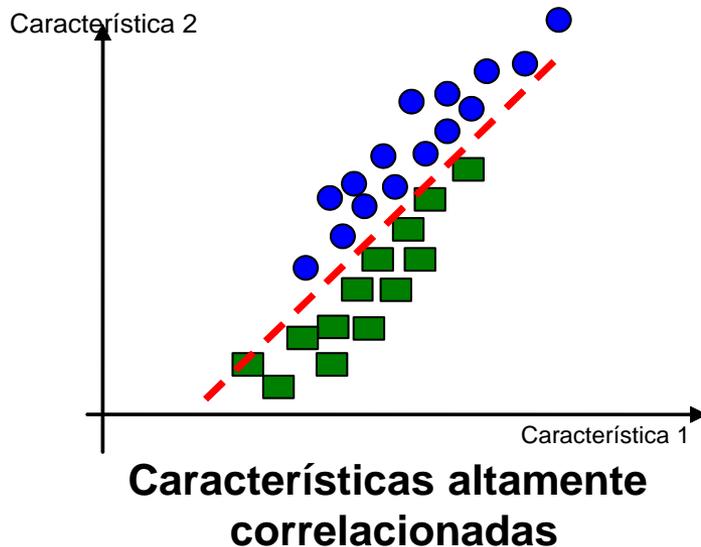
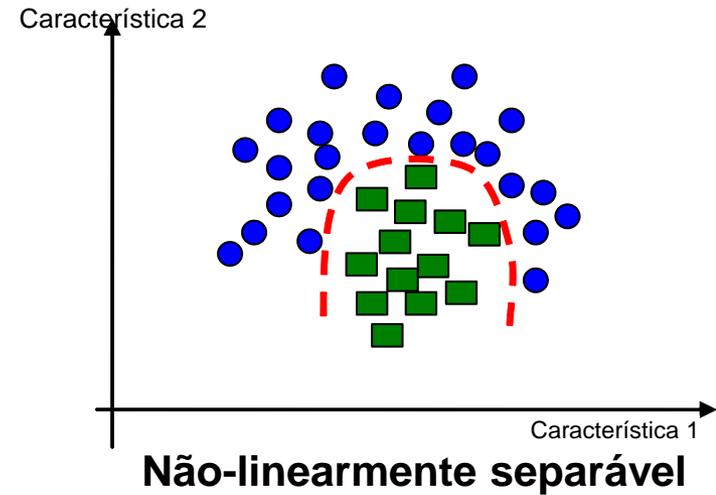
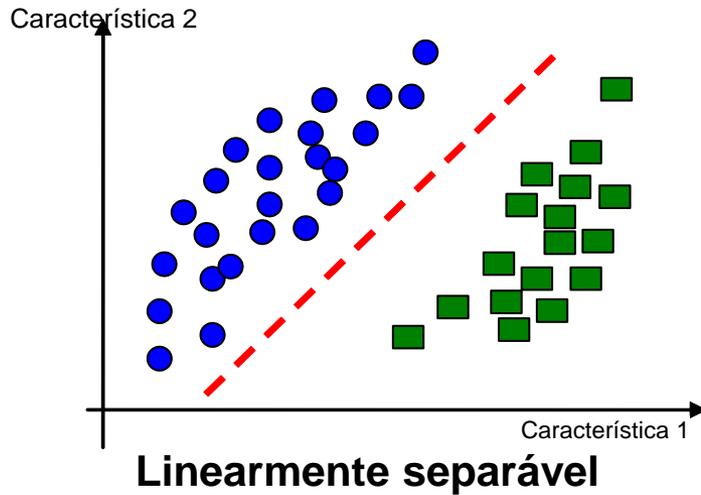


bom

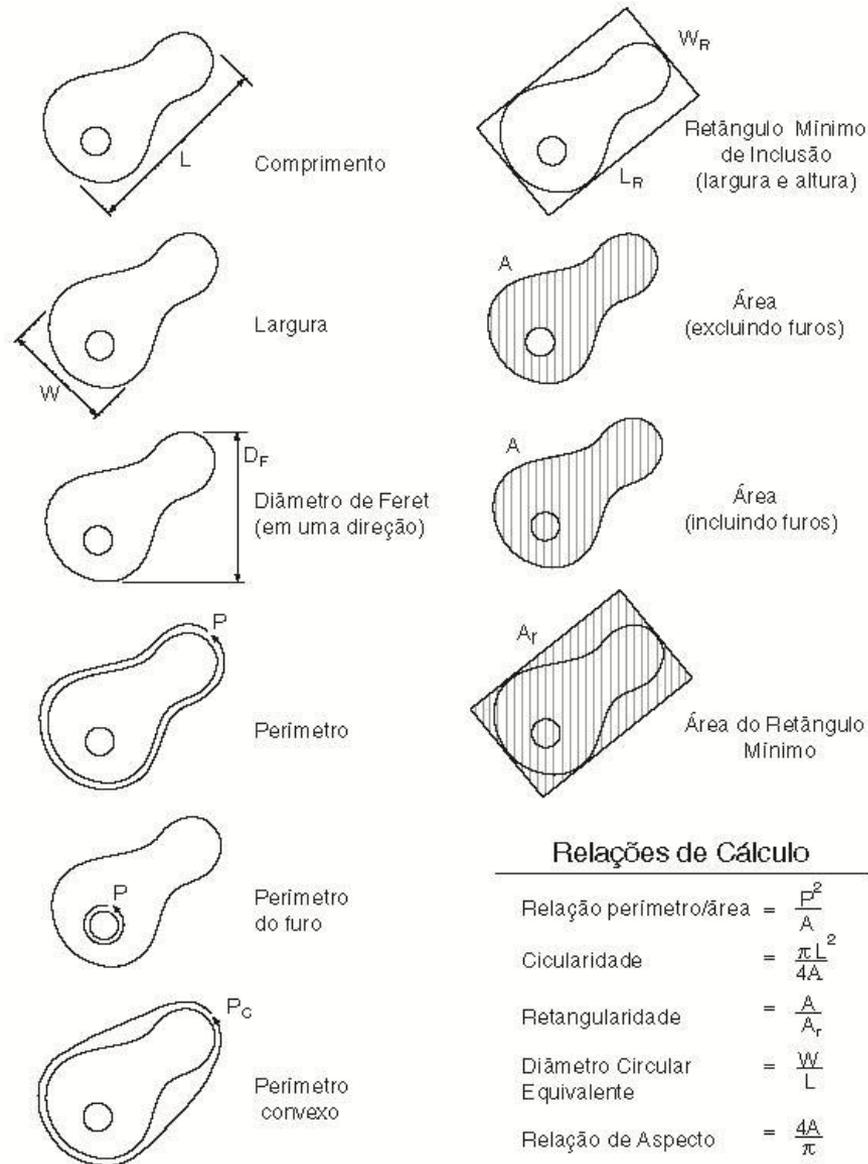


ruim

Outras Propriedades das Características



Características em Imagens

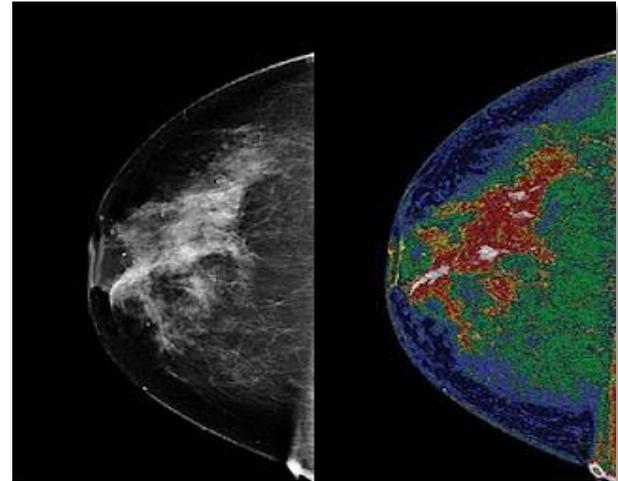


Motivação

Para que queremos um sistema de reconhecimento de padrões?

Para de forma iterativa, restringir nossa atenção a um conjunto de casos selecionado pelo sistema.

- Ex. Imagens médicas



Para automatizar completamente o processo de tomada de decisão, sem necessidade de intervenção humana.

- Ex. Identificação de placas de automóveis.



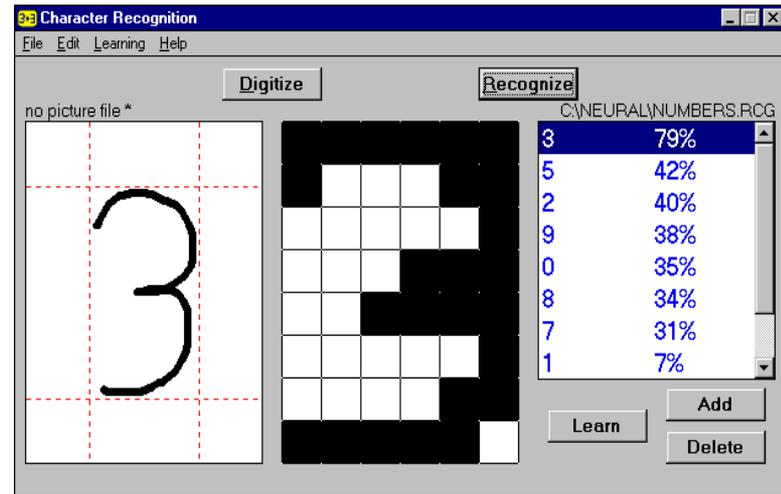
Aplicações de Reconhecimento de Padrões

Processamento de documentos: Reconhecimento de caracteres impressos ou escritos

Máquinas de leitura para cegos

Leitores de códigos de barras

Interpretação automática de textos



Automação industrial: Inspeção e montagem de objetos complexos

Inspeção de circuitos impressos

Inspeção de partes mecânicas

Visão por computador



NEMA 4X Enclosures

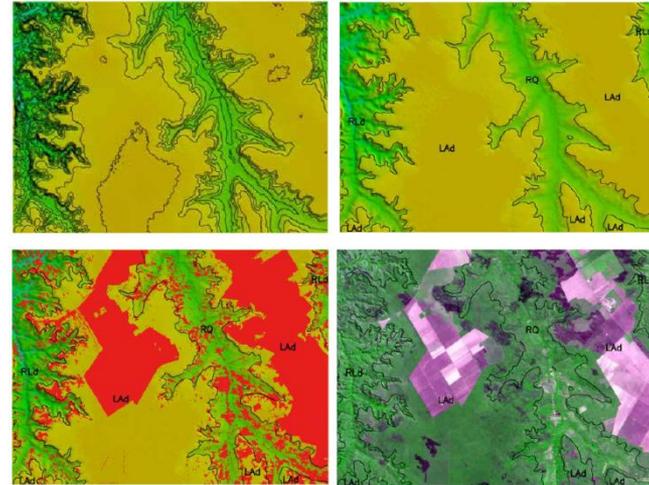
Camera View



Aplicações de Reconhecimento de Padrões

Detecção remota. Observação da terra através de sensores em satélites e aviões

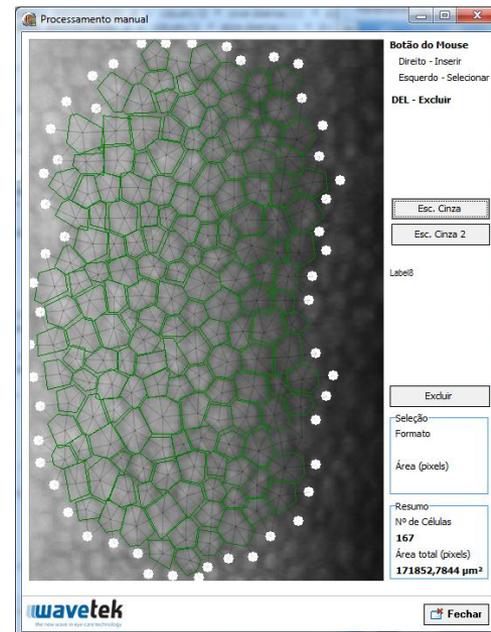
- Planejamento de uso da terra
- Monitoramento ambiental
- Meteorologia
- Exploração mineral
- Mapas topográficos



Fonte:
INPE, 2012

Medicina e biologia. Processamento de sinais e imagens médicas

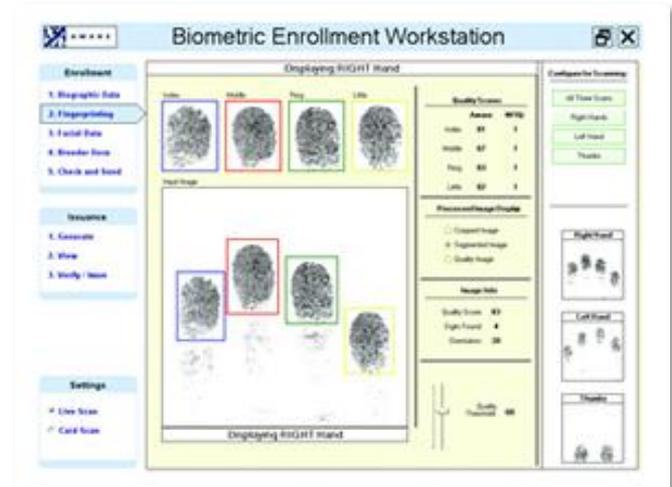
- Contagem de células no sangue
- Detecção de tumores em imagens de Raios-X
- Análise de imagens de cromossomos
- Interpretação de eletrocardiogramas
- Diagnóstico médico



Aplicações de Reconhecimento de Padrões

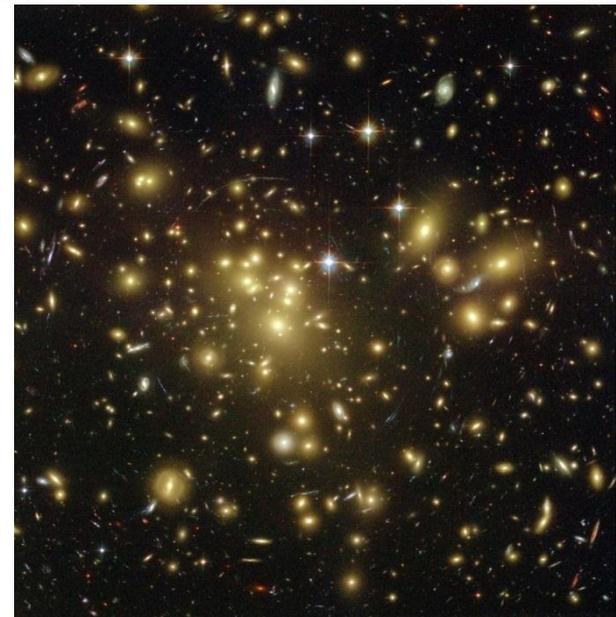
Identificação de pessoas

- Restrição de acesso em instalações de segurança
- Reconhecimento de voz
- Identificação de impressões digitais
- Reconhecimento de faces
- Reconhecimento de placas de veículos



Aplicações científicas

- Interpretação de ondas sísmicas
- Análise de imagens de microscópios
- Análise de imagens de telescópios (Hubble)



Fonte: Projeto DES
Dark Energy Survey
Colaboração do CBPF

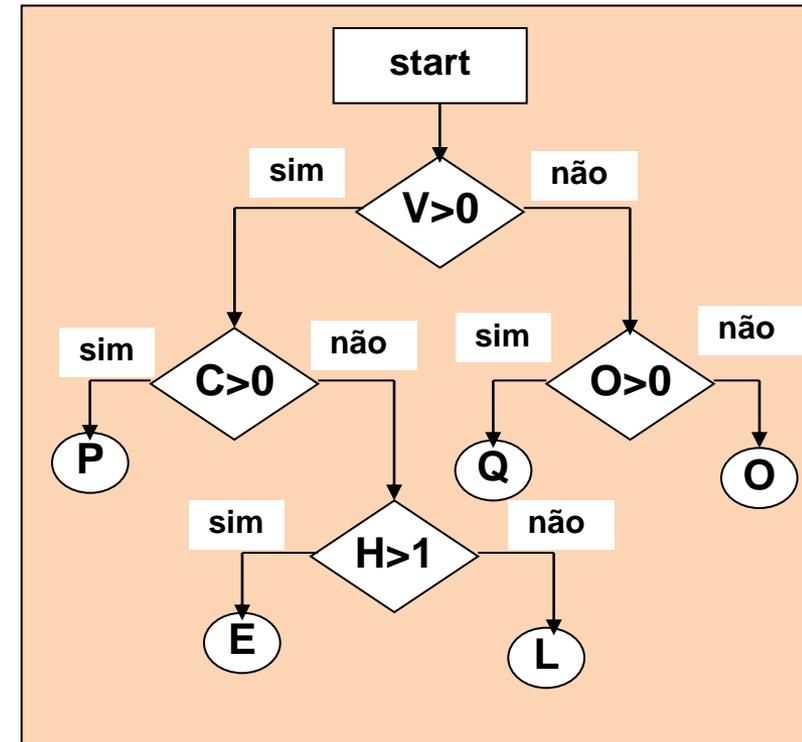
Exemplo Simples

Reconhecimento das letras L, P, O, E, Q

Determinar um conjunto suficiente de características

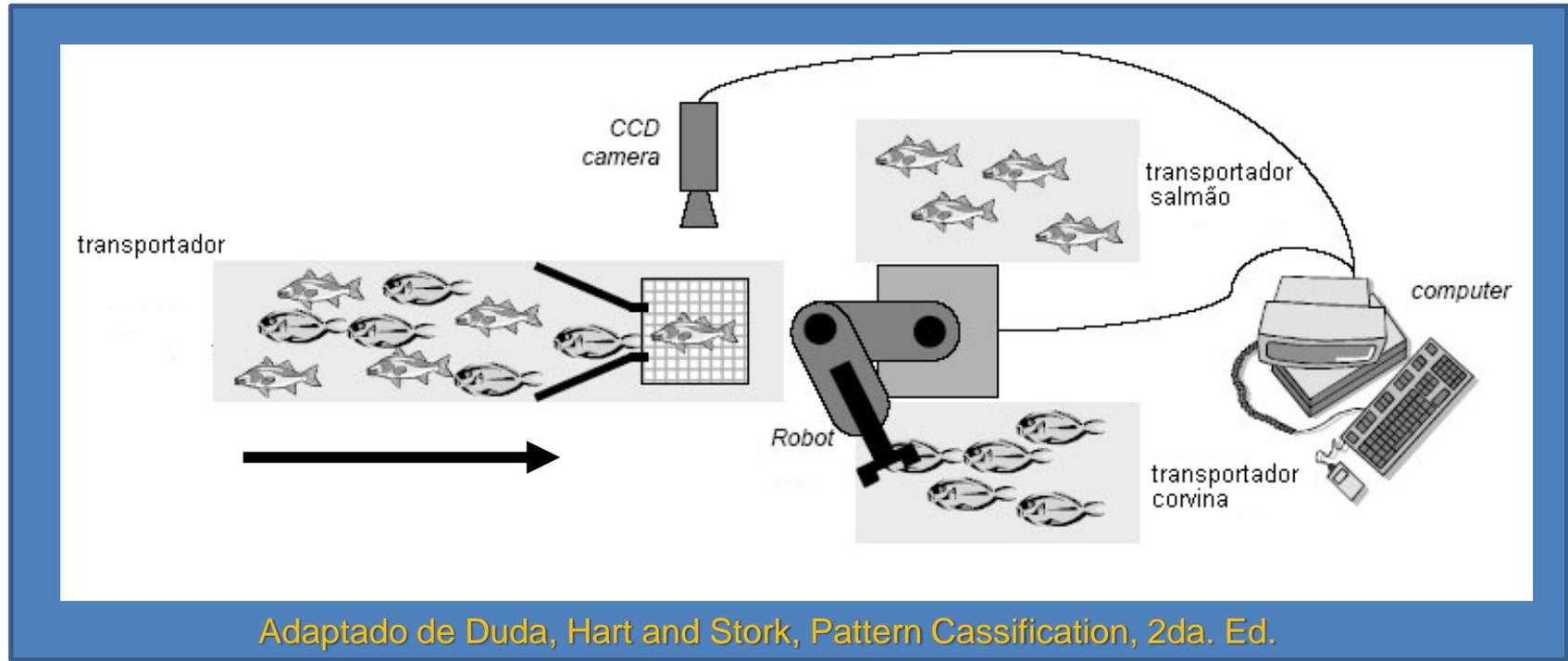
Desenhar um classificador de estrutura em árvore

Caracteres	Características			
	Linhas retas verticais (V)	Linhas retas horizontais (H)	Linhas retas obliquas (O)	Linhas curvas (C)
'L'	1	1	0	0
'P'	1	0	0	1
'O'	0	0	0	1
'E'	1	3	0	0
'Q'	0	0	1	1



Exemplo Prático

**Classificar os peixes em uma linha automatizada de um transportador de acordo com a espécie a que pertence:
Salmão ou Corvina.**



Análise do Problema

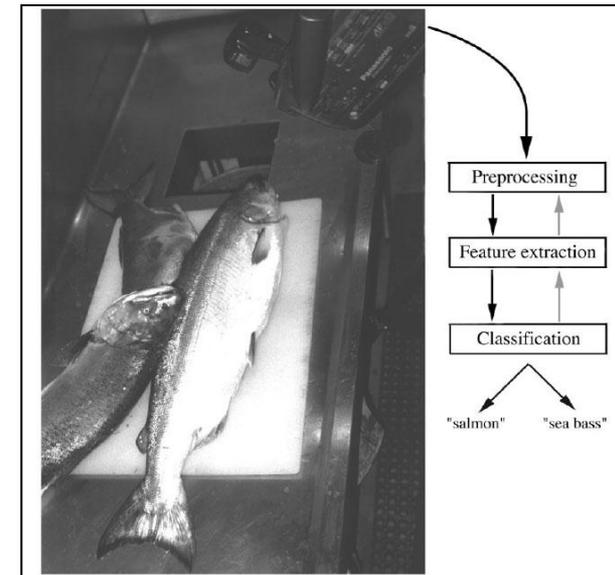
Montar um sistema de aquisição de imagens digitais para extrair algumas características:

- Luminância
- Eixos das Elipses – Largura e Comprimento
- Formato da “cauda”
- Área e perímetro do Objeto
- Etc..

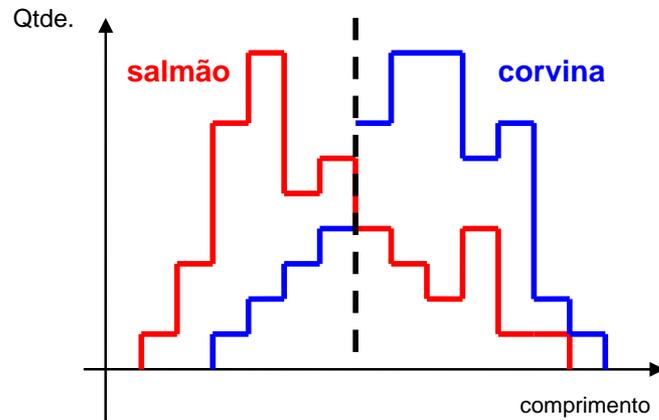
Qualquer uma destas características, ou uma combinação delas pode ser utilizada pelo classificador

Antes do Classificador

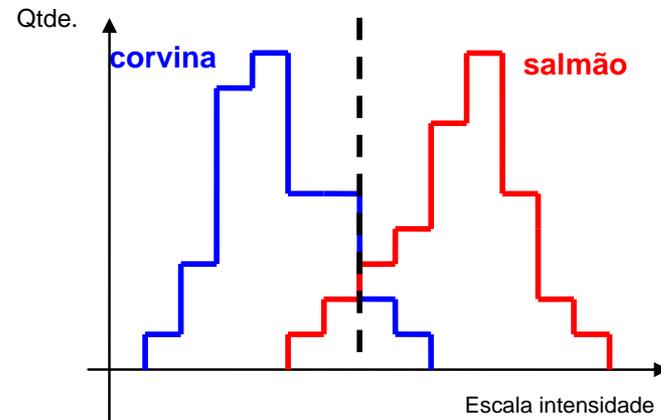
- Usar uma operação de segmentação para isolar os objetos (peixes) do fundo
- Usar técnicas de pós-processamento (morfologia), ainda na imagem, para garantir a separação dos objetos
- A informação correspondente a cada “peixe” é enviada a um extrator de características, cujo principal objetivo é reduzir os dados a um conjunto limitado de características
- As características são enviadas a um classificador



Análise do Problema - Características

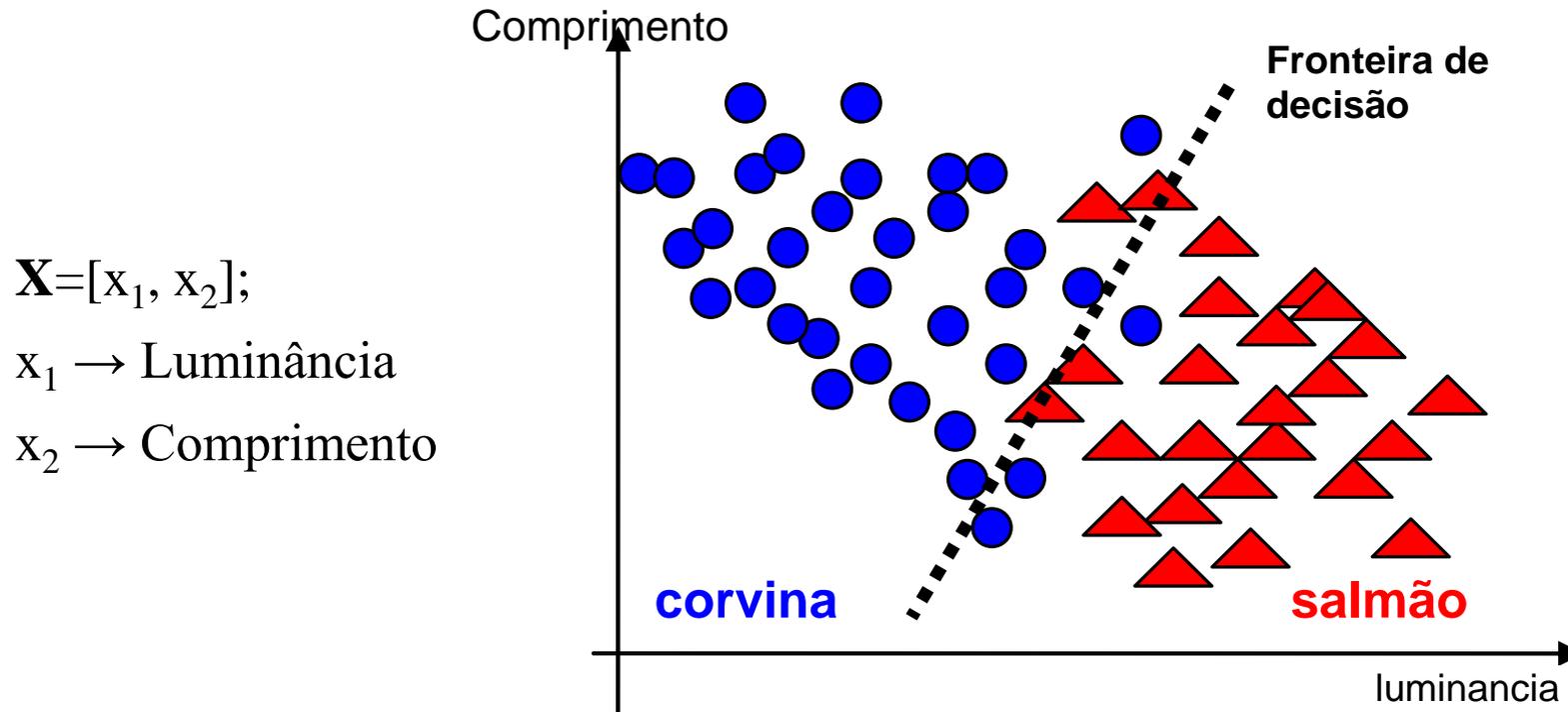


O comprimento não é uma boa característica ...



A luminância foi melhor !!!

Combinando as características



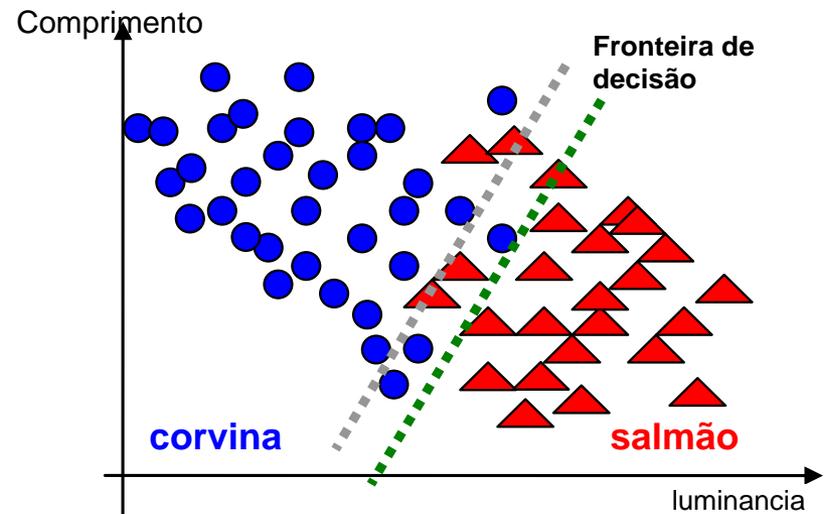
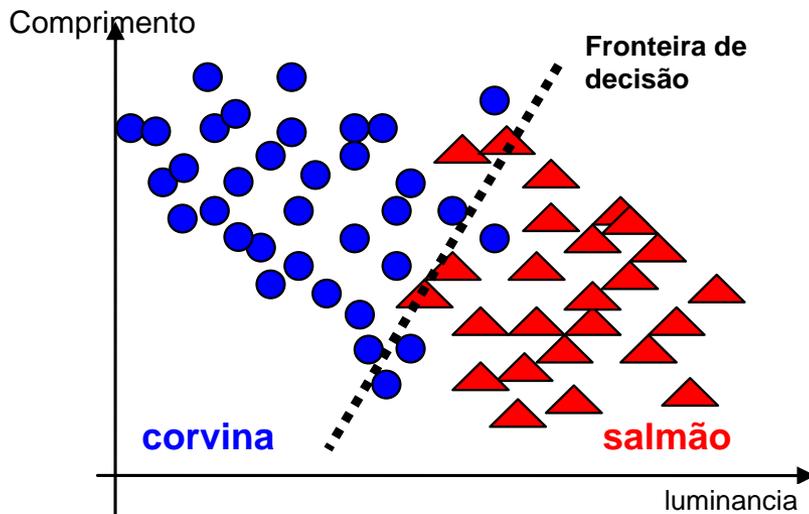
A combinação de características melhora a separação das classes
Melhora a taxa (erro) de classificação

Custo x Taxa de Classificação

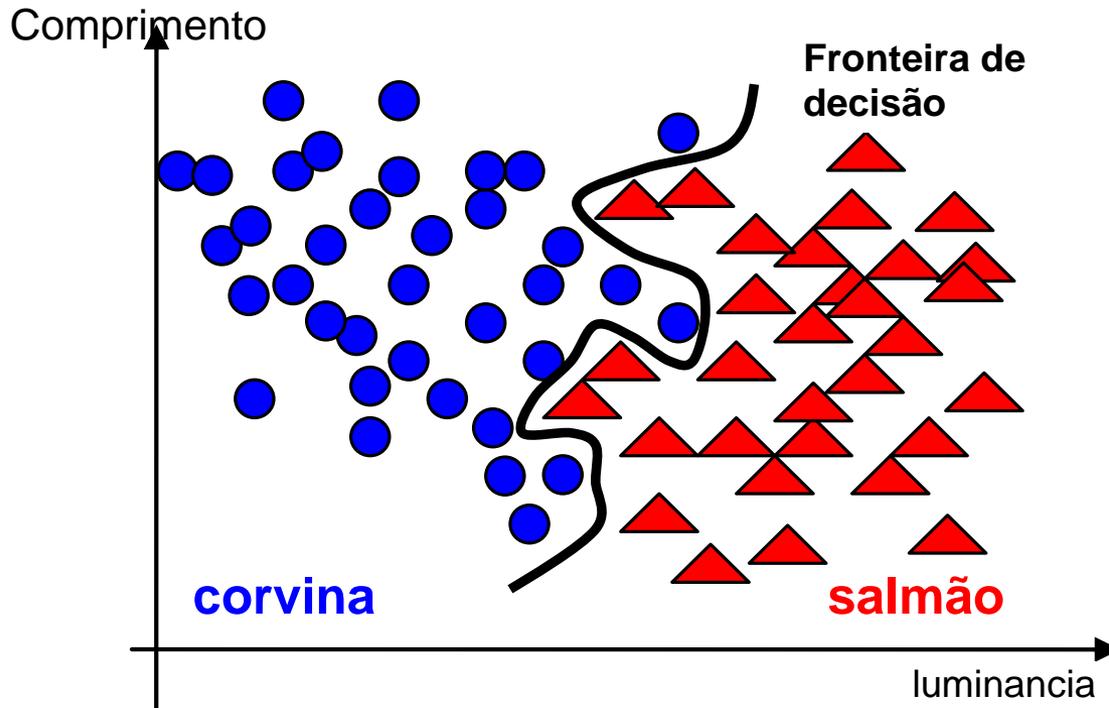
A taxa (erro) de classificação é a melhor função objetivo deste problema?

- Classificar o salmão como corvina fará com que “cliente final” encontre ocasionalmente um pedaço de corvina quando comprar salmão. E daí?..
- O custo de errar na classificação de corvina como salmão é ter um “cliente insatisfeito” quando encontrar um pedaço de corvina no salmão!

Mover a fronteira de decisão para diminuir a função de custo



Decisão Ótima?

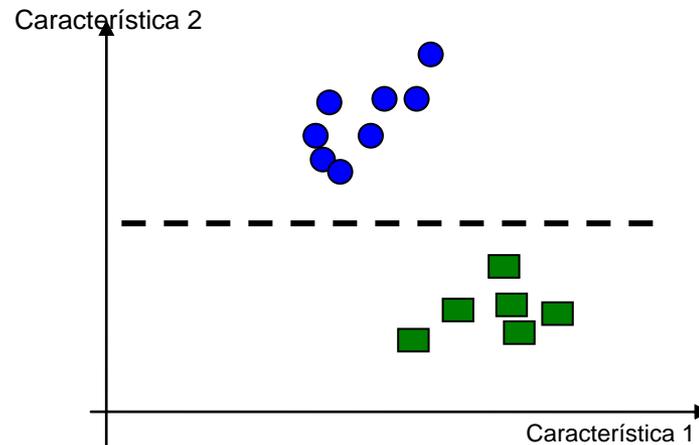
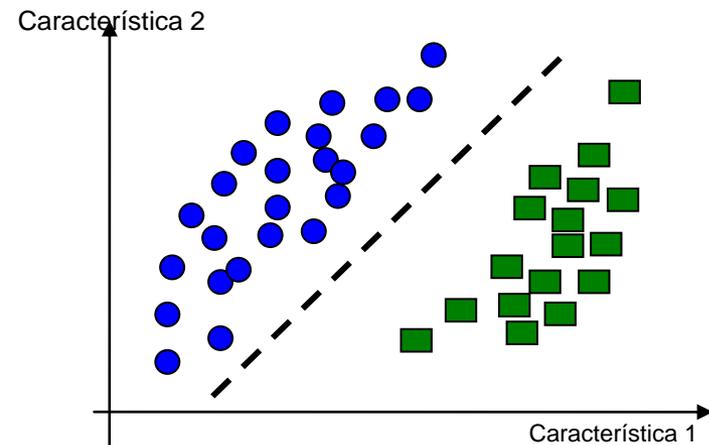
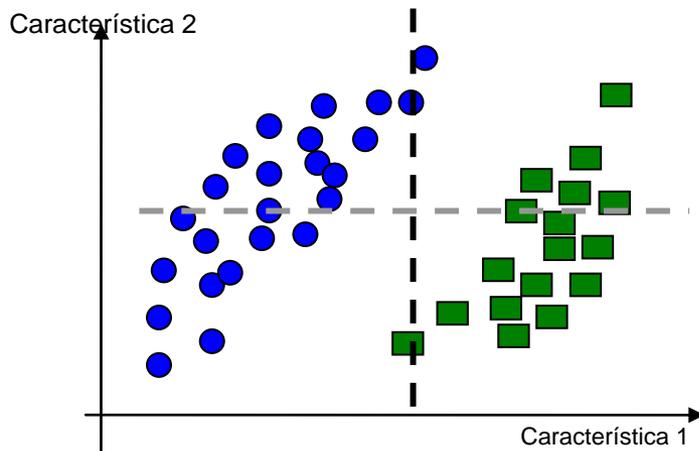


Ótimo!!! Integramos nosso sistema e desenvolvemos nossa aplicação na planta de processamento ... Depois de um tempo ligam da gerência que o erro de classificação é de 25%!!! Que aconteceu?

Generalização

O objetivo principal do classificador é classificar corretamente novas entradas – poder de generalização

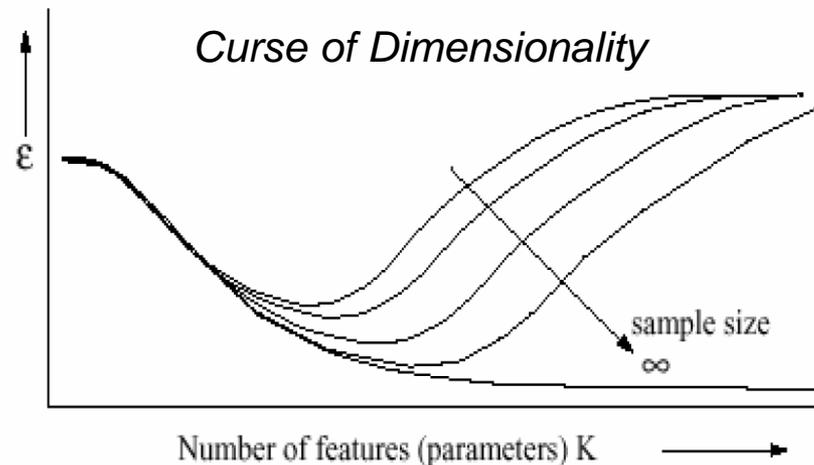
Relação entre dimensionalidade e conjuntos amostra



Quantas características devem ser usadas no classificador?

Quantas mais melhor!! – idéia errada

Na pratica o desempenho começa por melhorar mas vai se deteriorando à medida que mais características são consideradas



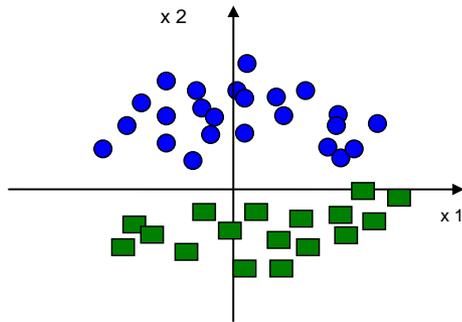
Os erros ocorrem devido ao uso não ótimo da informação adicional, que supera a vantagem da informação extra. Esparcidade

É necessário limitar o número de características para uma dada dimensão do conjunto de treino

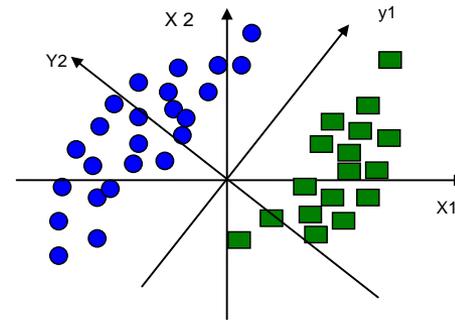
Seleção ou Extração de Características

Seleção: processo de escolha de um sub-conjunto das características originais

Extração: definição de novas características que podem ser função das características originais



Seleção: escolhe-se x_2 pois separa facilmente as classes



Extração: após rotação dos eixos de coordenadas, é evidente que é apenas necessária uma das características: y_2

Seleção de Características

Objetivo: Encontrar o melhor subconjunto de dimensão d das D características existentes

A melhor solução só pode ser encontrada através de uma procura exaustiva em todos os conjuntos possíveis de dimensão d

Pode ser computacionalmente impraticável

Escolher as d características que produzem individualmente melhores resultados → ERRADO

Técnica de seleção sequencial: Suponhamos que selecionamos k características. Então, a $(k+1)$ ésima característica é aquela que, em combinação com as k existentes, proporciona o melhor desempenho

Outras soluções: Algoritmos genéticos, Testes correlativos, etc.

Extração de Características

Objetivo: Aplicar algum tipo de transformação sobre o conjunto original de características, de forma a que as classes estejam mais separadas no novo espaço.

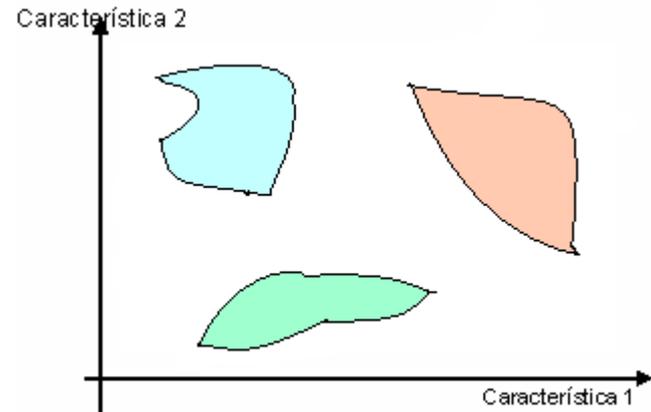
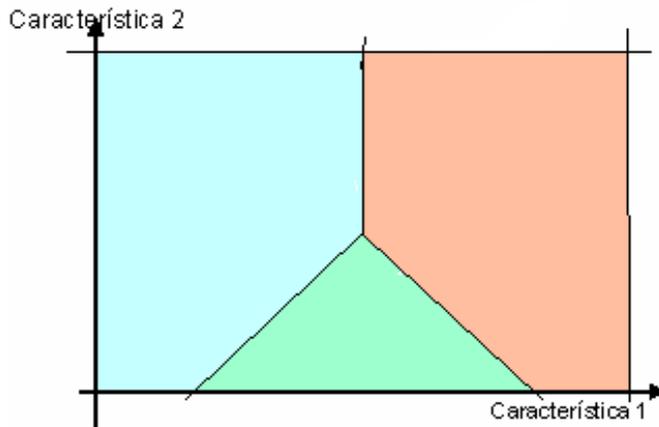
Vantagem adicional: O problema de seleção de características no novo espaço é mais simples

Exemplos de técnicas: PCA ou transformada de Karhunen-Loeve:

Usa os vetores próprios (auto-vetores) e valores próprios (auto-valores) da matriz de covariância de todos os dados de forma a reduzir a sua dimensionalidade.

Projeto do Classificador

Projetar um classificador é equivalente a escolher as regiões de decisão, que por sua vez podem ser associadas a funções discriminantes.



Funções Discriminantes

A classificação pode ser representada por um conjunto de funções de discriminação

As funções discriminantes são uma sequência de funções $g_1(\cdot); \dots; g_c(\cdot)$ associadas às regiões de decisão $S_1; \dots; S_c$.

$$g_i(x) > g_j(x), \quad \forall x \in S_i, \quad \forall j \neq i$$

$$w' = w_k : k = \arg \max \{g_i(x)\}, \quad i = 1; \dots; C$$

O conhecimento das funções discriminantes dispensa a numeração das classes para cada ponto do espaço de características.

Não existe uma única sequência de funções discriminantes para um dado classificador.

Avaliação

A qualidade de um classificador pode ser representada sinteticamente na forma de uma matriz quadrada P , designada por matriz de confusão, e pela probabilidade total do erro do classificador.

Matriz de Confusão

- A matriz de confusão P é uma matriz quadrada de $C \times C$, em que C é o número de classes

$$P = \begin{bmatrix} p_{11} & \dots & p_{1C} \\ \cdot & & \cdot \\ p_{C1} & \dots & p_{CC} \end{bmatrix}$$

$$p_{ij} = p(\omega' = \omega_j \mid \omega = \omega_i) \quad p_{ij} \geq 0; \quad i, j = 1, \dots, C \quad \sum_j^C p_{ij} = 1$$

No caso ideal P é uma matriz unitária

Matriz de Confusão

O valor da matriz de confusão geralmente estimado experimentalmente

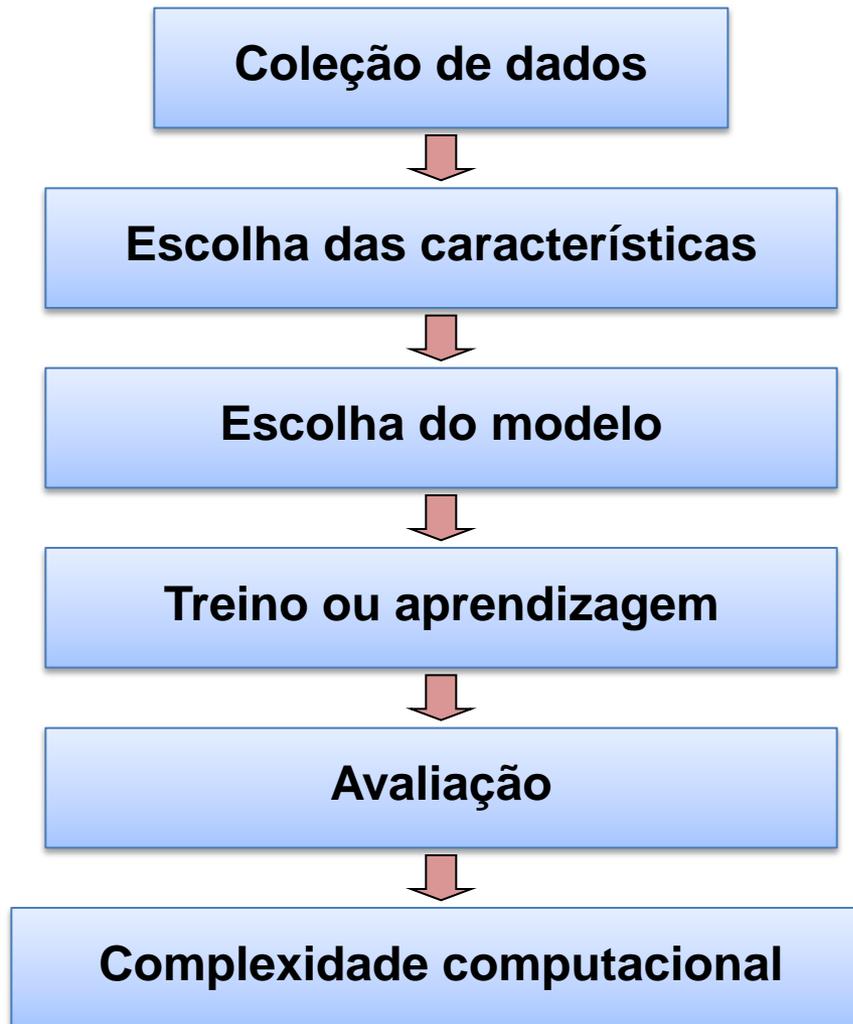
Necessário dispor de um conjunto de observações previamente classificadas, testar esse conjunto no classificador, e comparar os resultados do classificador com os valores já conhecidos.

Cada elemento p_{ij} da matriz de confusão pode ser estimado por

$$p'_{ij} = \frac{n_{ij}}{n_i}$$

n_{ij} é o número de padrões gerados pela classe w_j e classificados como pertencentes a w_i . E n_i é o número total de padrões pertencentes à classe w_i .

Projeto de um Classificador



Projeto de um classificador

Coleção de dados. Conjunto de treinamento. Conjunto grande e representativo de exemplos para treinar e testar o sistema.

Escolha das características. Propriedades estatísticas. Dependem do problema. Simples a extrair, invariantes a transformações e “insensível” ao ruído.

Escolha do modelo. Experimentar vários modelos se o desempenho não for satisfatório.

Treino ou aprendizagem. Usar os dados para determinar classificador. Existem muitos métodos.

Avaliação. Critério de desempenho. Medir a taxa de erro.

Complexidade computacional. Analisar relação entre a complexidade computacional e o desempenho. Analisar a complexidade do algoritmo em função do número de características e padrões.

Aprendizagem

Aprendizagem supervisionada

Conjunto de treino rotulado

Conhece-se a classificação correta de cada um dos padrões

Forma de aprendizagem, pois é apresentado ao classificador a sua entrada(padrão) e a sua saída (classe)

Aprendizagem não supervisionada

Não se conhece a classificação dos padrões de treino

O sistema forma clusters ou agrupamento naturais dos padrões de entrada

Abordagens Típicas

Estatística: assume um modelo de dados estatístico.
Supervisionado ou não.

Ex. Classificador de Bayes, *k-means*.

Estrutural ou sintática: é definido uma sequência de elementos básicos (primitivas) que representem as formas dos objetos. É criada uma linguagem e uma gramática

Neuronal: Imita o funcionamento do cérebro humano.
Supervisionado ou não. Ex. *Redes MLP, SOM, RBF*

EXEMPLO 1

Sistema de Reconhecimento de Padrões

Análise de Imagens e Visão Computacional

1. O Problema

Observação de microorganismos

Observação - 2 Classes distintas

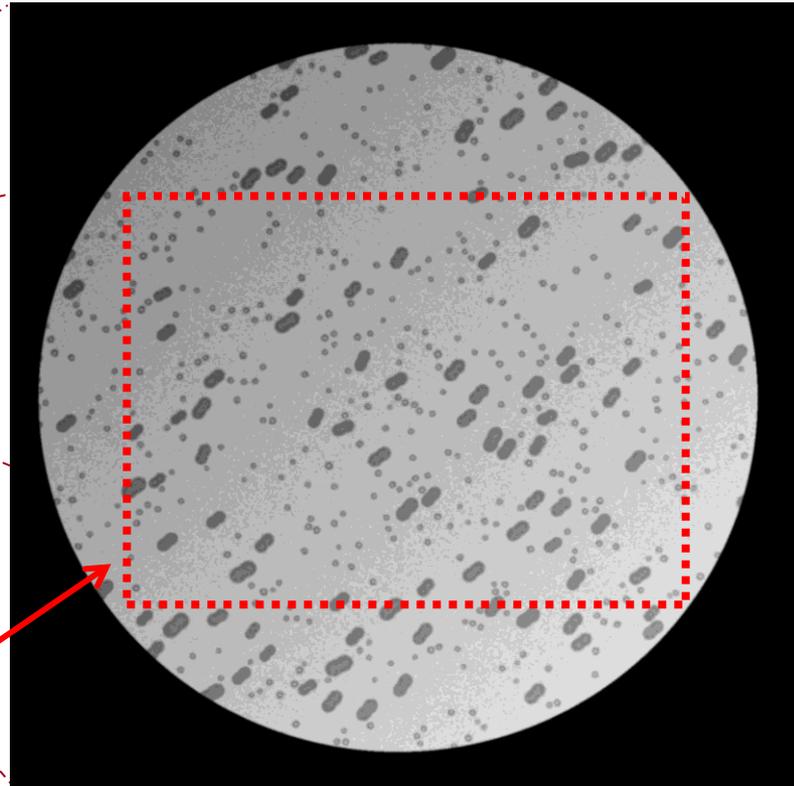
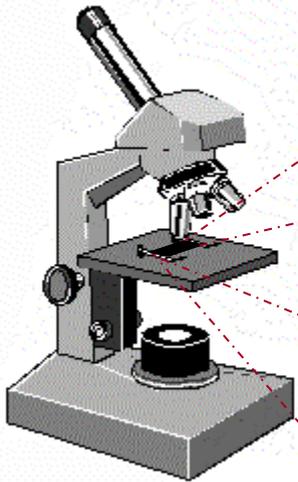
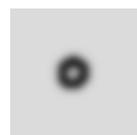


Imagem útil



Bactéria
Classe 1



Célula
Classe 2

Procura-se:

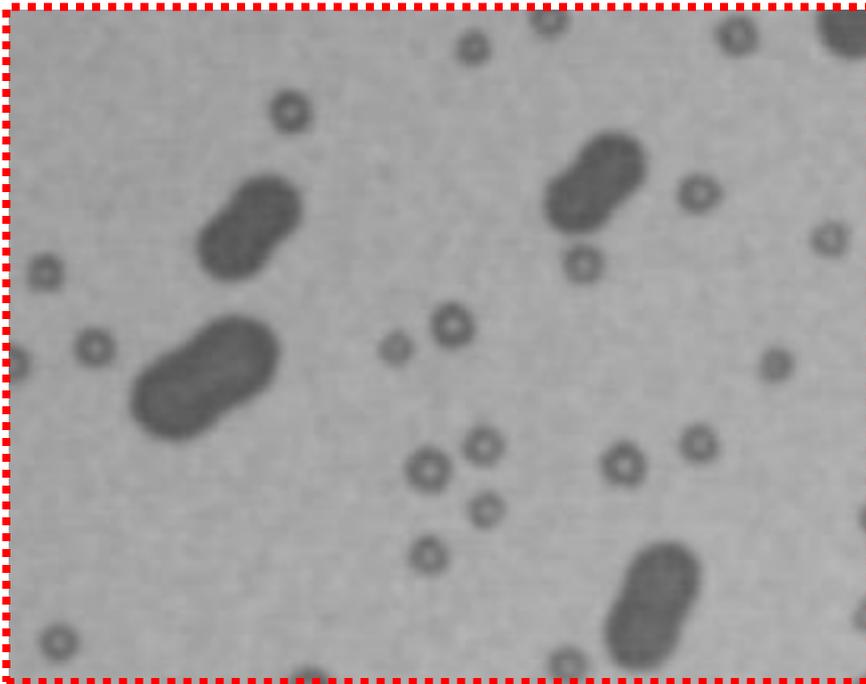
1. número de bactérias e de células observadas
2. Orientação das bactérias e células
3. Tamanho

Problemas:

1. Imagem com iluminação irregular
2. Ruídos diversos
3. Identificação dos objetos e análises independentes

2. Número de bactérias e células

Imagem útil → Imagem analisada



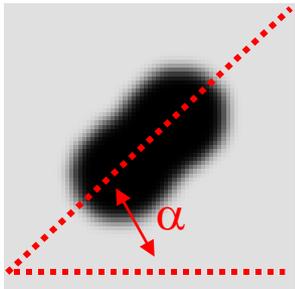
Exemplo

		Ocorrências	
	Bactéria Classe 1	4	18,2%
	Célula Classe 2	18	81,8%

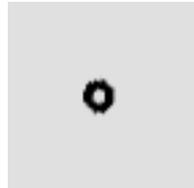
Objetos que tocam a borda da imagem devem ser descartados

3. Orientação e Tamanho

Bactéria
Classe 1



Célula
Classe 2



Bactéria
Classe 1

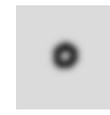
0.5



Célula
Classe 2



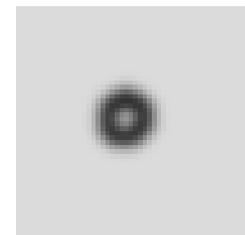
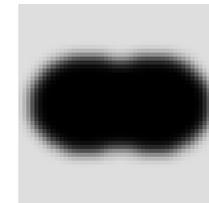
0.85



1.0



1.5



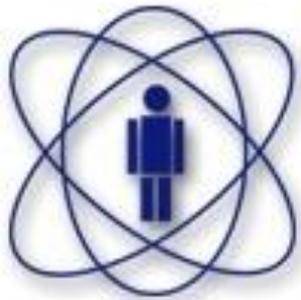
1. Tamanho é relativo

1. α é o ângulo de orientação
2. Classe 2 não tem ângulo de orientação

Obs: Estamos interessados em valores estatísticos – média e desvio padrão

EXEMPLO 2

**APLICAÇÃO DE TÉCNICAS DE
PROCESSAMENTO DIGITAL DE IMAGENS
PARA A DETECÇÃO DE MARFES NO JET**



CBPF

Ministério da
Ciência e Tecnologia



Centro Brasileiro de Pesquisas Físicas

<http://www.cbpf.br>

Coordenação de Formação Científica

<http://www.cfc.cbpf.br>

Rua Dr. Xavier Sigaud, 150 - Urca - Rio de Janeiro, Brasil

Tel:(0xx21)2141-7163/2141-7203 Fax:(0xx21)2141-7557 CEP:22290-180

Mestrado em Instrumentação

**APLICAÇÃO DE TÉCNICAS DE
PROCESSAMENTO DIGITAL DE IMAGENS PARA
A DETECÇÃO DE MARFES NO JET**

Germano Teixeira Chacon

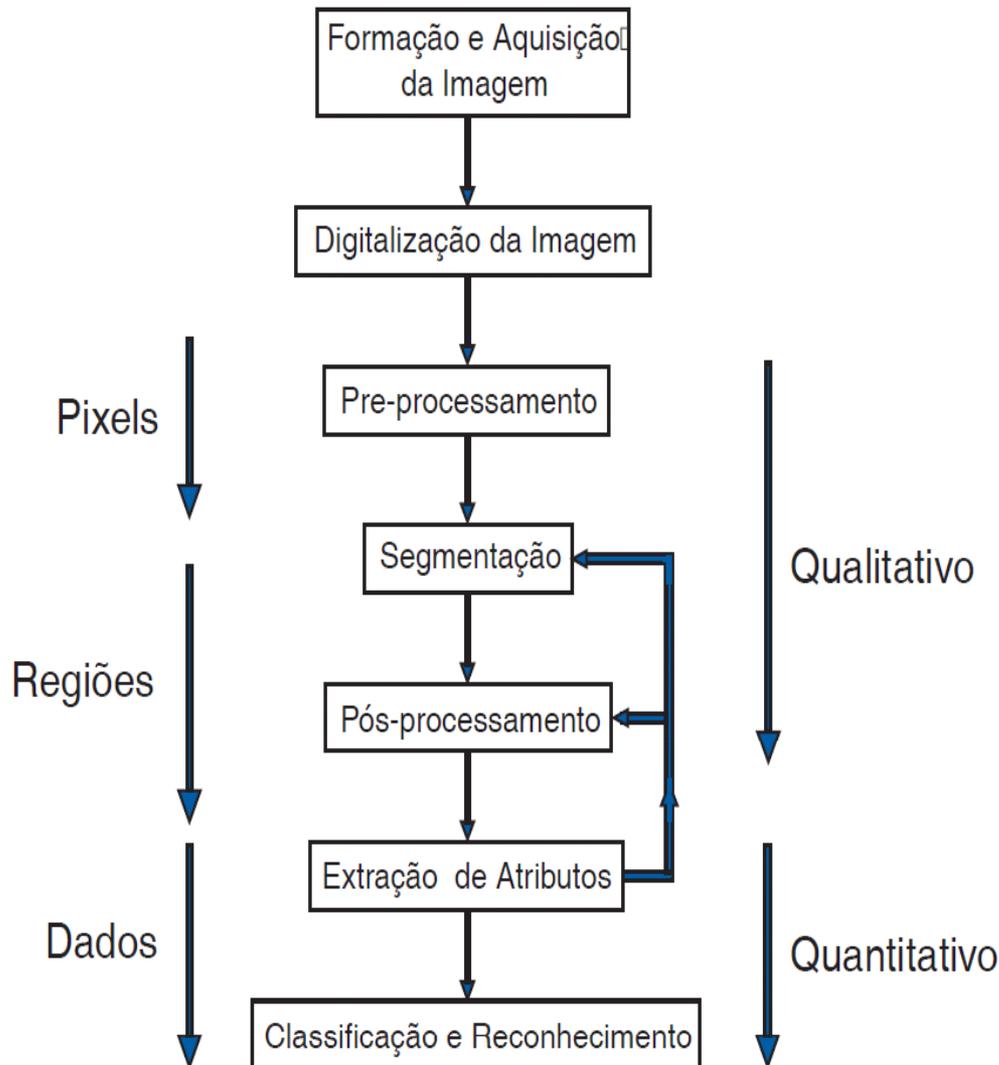
Tópicos

1. Motivação
2. Fundamentos Teóricos
 - a) Extração de Atributos
 - b) Classificação e Reconhecimento
3. Fusão Nuclear
4. Processamento de Imagens para o JET
5. Desenvolvimento e Caracterização
 - a) Em Matlab
 - b) Em C/C++
6. Conclusões e Perspectivas

1 Motivação

- A produção comercial de energia elétrica a partir da fusão de átomos leves, como uma fonte alternativa de energia de larga escala.
- Existem atualmente diversos desafios tecnológicos para a operacionalização deste tipo de equipamento
- No final de 2009 um acordo de cooperação foi assinado com a Comunidade Europeia da Energia Atômica (EURATOM), do qual o Laboratório JET é um dos principais experimentos científicos.
- Neste trabalho estudamos, desenvolvemos e apresentamos técnicas e algoritmos de processamento de imagem para processar vídeos do JET. O objetivo é detectar um fenómeno conhecido por MARFE (Multifaceted Asymmetric Radiation From The Edge).

2 Fundamentos Teóricos



Processos de baixo nível: envolvem operações primitivas de pré processamento .

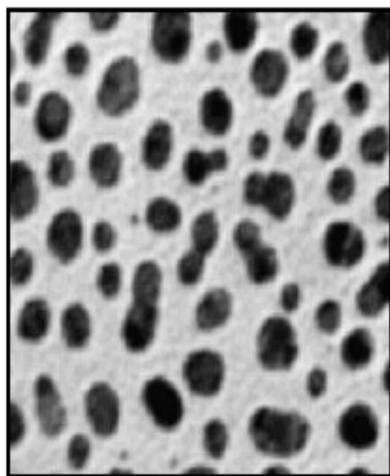
Processos de médio nível: estes processos envolvem tarefas de segmentação de objetos.

Processos de alto nível: estes processos identificam e atribuem um significado aos objetos reconhecidos.

2 Fundamentos Teóricos

Extração de Atributos

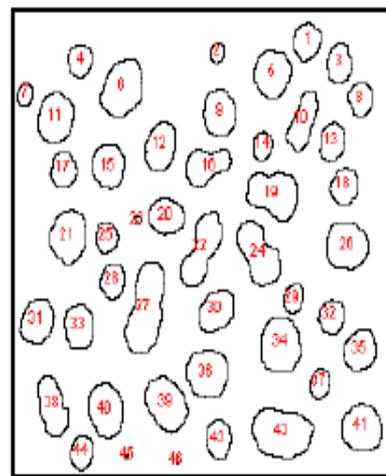
- O objetivo desta fase é extrair informações úteis, normalmente na imagem binária. Existem basicamente duas classes de medidas são elas:
 - atributos da Imagem como um todo (Field Features), ex.: número de objetos, área total de objetos, etc;
 - atributos de região (Region Features) que se referem aos objetos independentes, ex.: área, perímetro, forma, etc.



(a)



(b)



(c)

2 Fundamentos Teóricos

Momentos de Uma Imagem

- A técnica de momentos é um dos vários métodos utilizados para extração de características de uma imagem.
- Os momentos de imagem mais utilizados são os **momentos regulares** definidos a partir da seguinte fórmula:

$$m_{pq} = \sum_1^{nx} \sum_1^{ny} x^p y^q f(x, y)$$

- os momentos regulares de ordem 0 e 1 são usados para o cálculo do baricentro ou centro de massa do objeto, através das seguintes fórmulas:

$$x_c = \frac{m_{10}}{m_{00}}; y_c = \frac{m_{01}}{m_{00}}$$

2 Fundamentos Teóricos

Momentos de Uma Imagem

- Com a informação obtida dos baricentros obtemos o que chamamos de ***momentos centrais*** que são definidos para imagens digitais como

$$u_{pq} = \sum_{x=1}^{nx} \sum_{y=1}^{ny} (x - x_c)^p (y - y_c)^q f(x, y)$$

- Finalmente existem os ***momentos centrais normalizados*** representados por n_{pq} e definidos pela seguinte fórmula:

$$n_{pq} = \frac{u_{pq}}{m_{00}^{\frac{(p+q)+1}{2}}}$$

2 Fundamentos Teóricos

Momentos de Uma Imagem

- **Momentos Hu** resolvem um grande problema no reconhecimento de padrões que é a variação das características dos objetos.

$$I_1 = n_{20} + n_{02}$$

$$I_2 = (n_{20} - n_{02})^2 + (2n_{11})^2$$

$$I_3 = (n_{30} - 3n_{12})^2 + (3n_{21} - n_{03})^2$$

$$I_4 = (n_{30} + n_{12})^2 + (n_{21} + n_{03})^2$$

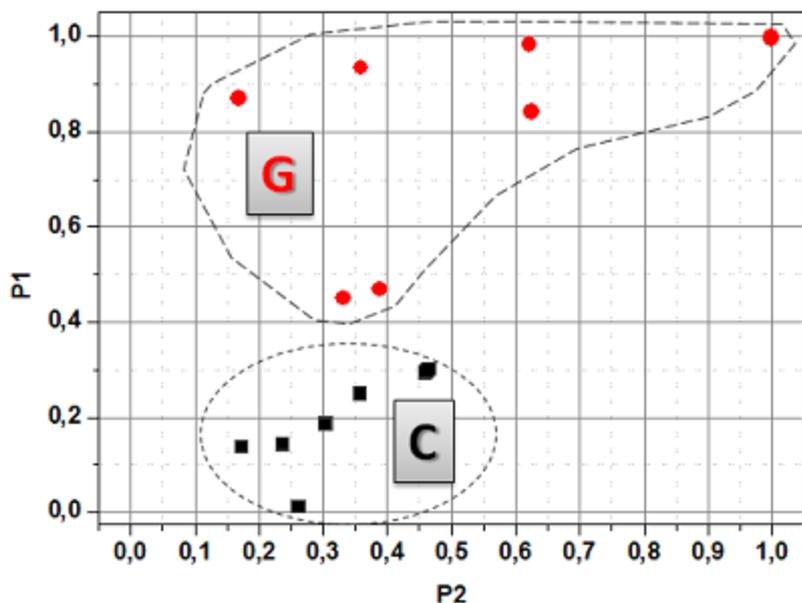
$$I_5 = (n_{30} - 3n_{12})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] \\ + (3n_{21} - n_{03})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2]$$

$$I_6 = (n_{20} - n_{02})[(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2] + 4n_{11}(n_{30} + n_{12})(n_{21} + n_{03})$$

$$I_7 = (3n_{21} - n_{03})(n_{30} + n_{12})[(n_{30} + n_{12})^2 - 3(n_{21} + n_{03})^2] \\ + (n_{30} - 3n_{12})(n_{21} + n_{03})[3(n_{30} + n_{12})^2 - (n_{21} + n_{03})^2]$$

2 Fundamentos Teóricos

Extração de Atributos por Momentos Hu



- Descrever formas em imagens binárias.
- Desenvolvimento de aplicações para reconhecimento de padrões.
 - Reconhecimento automático de caracteres e de faces,
 - Na área médica temos aplicações no reconhecimento de anomalias fisiológicas em imagens tomográficas ou radiográficas.
 - Aplicação em imagens do JET/EFDA

2 Fundamentos Teóricos

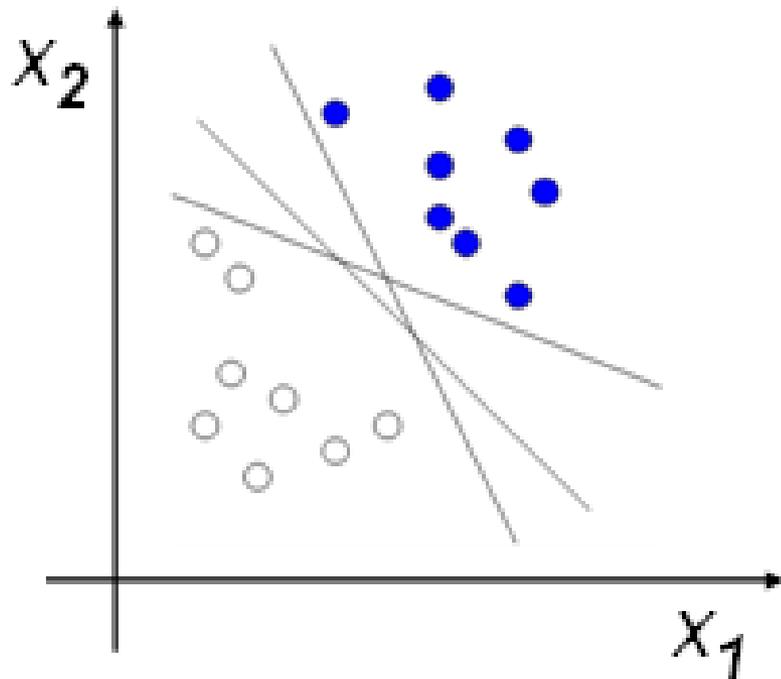
Classificação e Reconhecimento

- A necessidade do Reconhecimento de Padrões surge de problemas onde precisamos separar dados de natureza diferente em categorias.
- O objetivo do reconhecimento é realizar, de forma automática, a “identificação” dos objetos segmentados na imagem.
- Os parâmetros provenientes da etapa de extração de atributos são utilizados para construir um espaço de características à N dimensões.
- As formas são representadas por vetores contidos dentro do espaço de características.
- Podemos utilizar algoritmos de Aprendizado de Máquina (AM) a fim de obter um classificador capaz de prever a classe de novos eventos que ocorram no mesmo espaço de características.

2 Fundamentos Teóricos

Classificação e Reconhecimento

Support Vector Machines

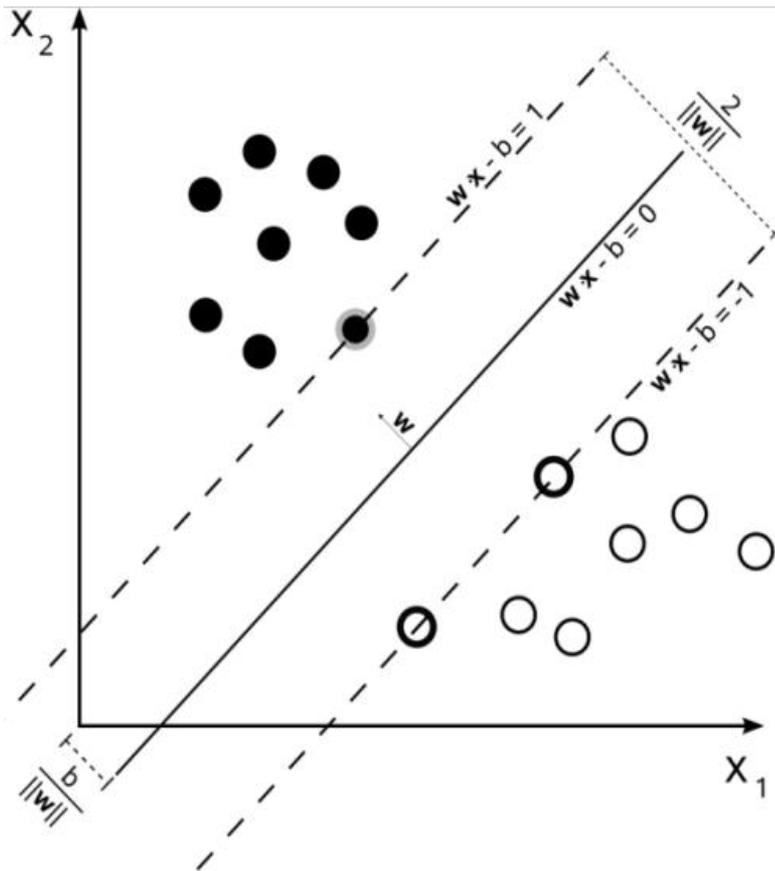


- Técnica de aprendizado supervisionado amplamente usada em reconhecimento de padrões.
- O algoritmo de treinamento recebe um conjunto de exemplos na forma (x_i, y_i) , ao término deste obtemos um classificador que é representado por uma função $f(x)=y$

2 Fundamentos Teóricos

Classificação e Reconhecimento

Support Vector Machines

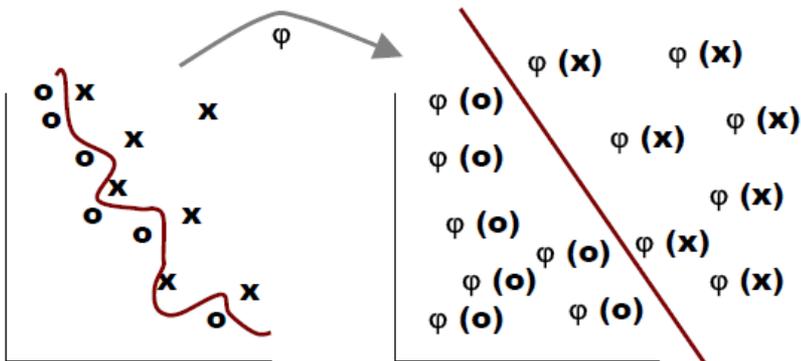


- Busca um Hiperplano ótimo maximizando margem de separação entre as classes
- Minimizar a norma de W

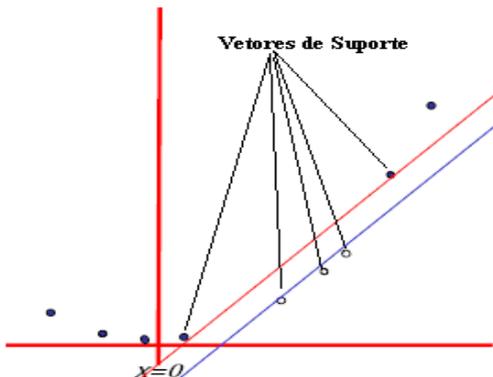
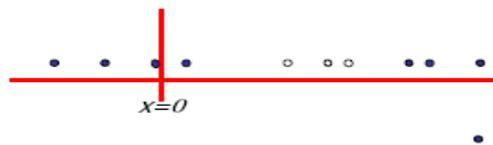
2 Fundamentos Teóricos

Classificação e Reconhecimento

Support Vector Machines



- No caso de dados não linearmente separáveis mapeamos os dados num espaço com mais dimensões que o espaço original.
- Uso de funções Kernel.



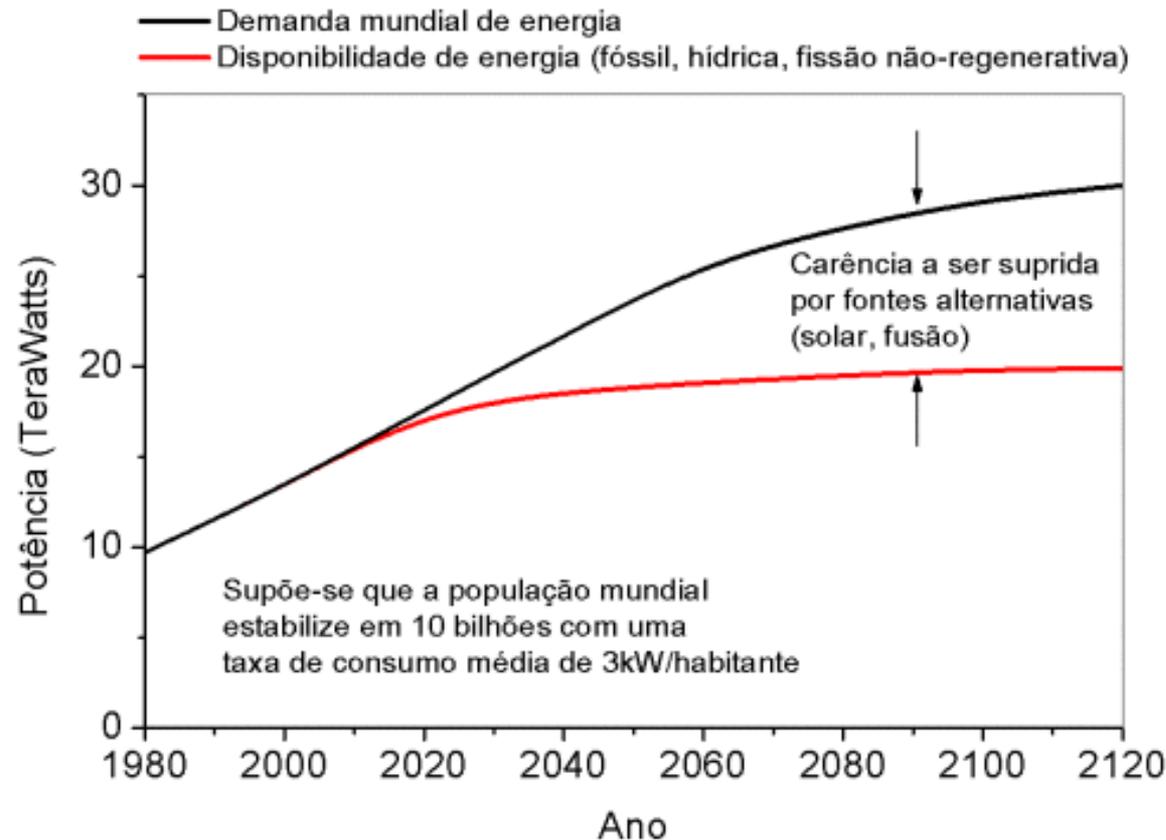
3 Fusão Nuclear

Um breve Histórico

- As pesquisas na área de fusão nuclear foram iniciadas em Los Alamos (EUA), com claras motivações militares para produzir uma bomba de fusão.
- Em 1951 Andrei Sakharov e Igor Tamm conceberam o Tokamak.
- Todos os trabalhos de investigação se mantiveram secretos até à Conferência "Átomos para a Paz" que decorreu em Genebra, em 1958.
- Em 1978, a Comunidade Europeia (mais a Suécia e a Suíça) empreenderam a construção do projeto JET ("Joint European Torus")
- O projeto ITER ("International Thermonuclear Experimental Reactor") iniciado em 1992.

3 Fusão Nuclear

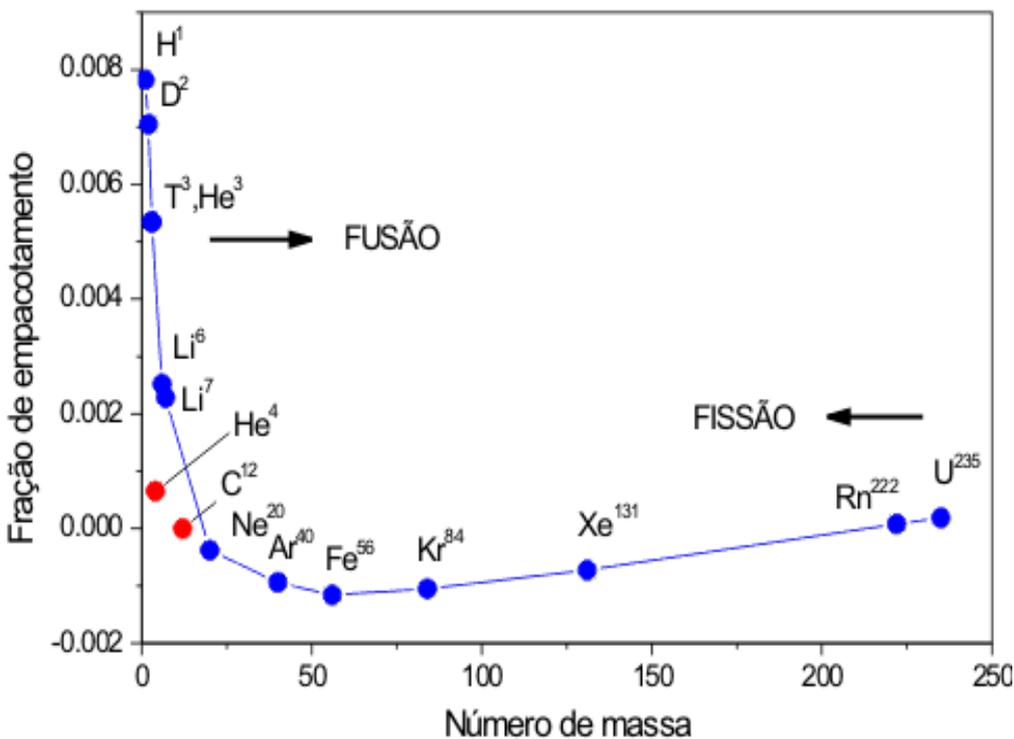
Vantagem e Necessidade



- Potência
- Segurança
- Matéria prima abundante.
- Baixa toxicidade

3 Fusão Nuclear

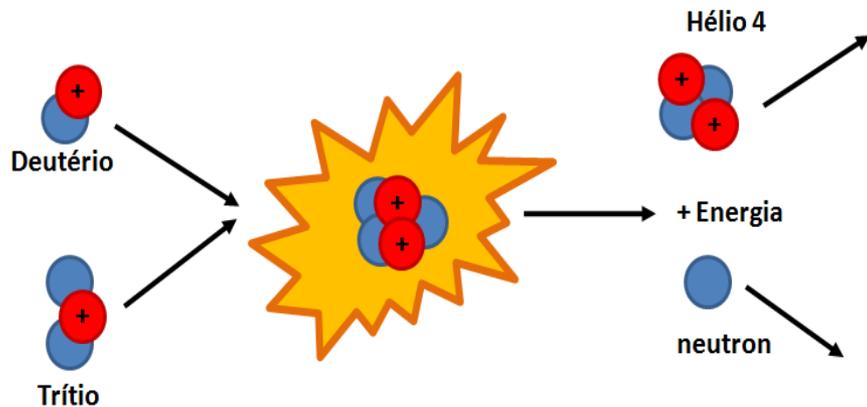
Conceitos Básicos



- A fusão nuclear é o processo de construir núcleos mais pesados a partir de núcleos mais leves.
- A energia nuclear pode ser obtida a partir da ligação entre dois núclídeos leves (fusão) ou da cisão de núcleos pesados (fissão)
- A energia disponível por núcleo é dada pela fração de empacotamento definida como $P=(M-A)/A$

3 Fusão Nuclear

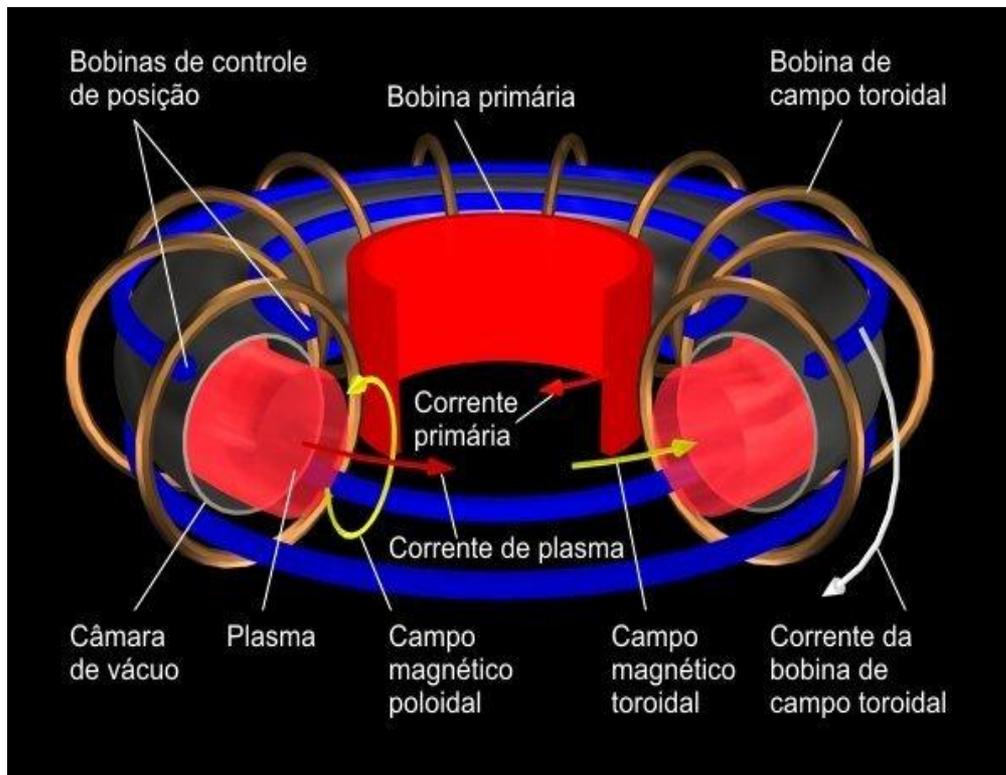
Reações de Fusão



- As reações mais importantes envolvem deutério (D^2) e trítio (T^3) na forma de um plasma dentro do reator.
- Requerem temperaturas muito elevadas.
- Utilizam uma mistura de reagentes pouco densa.
- Submeter este plasma a ação de uma força externa de forma a mantê-lo confinado.

3 Fusão Nuclear

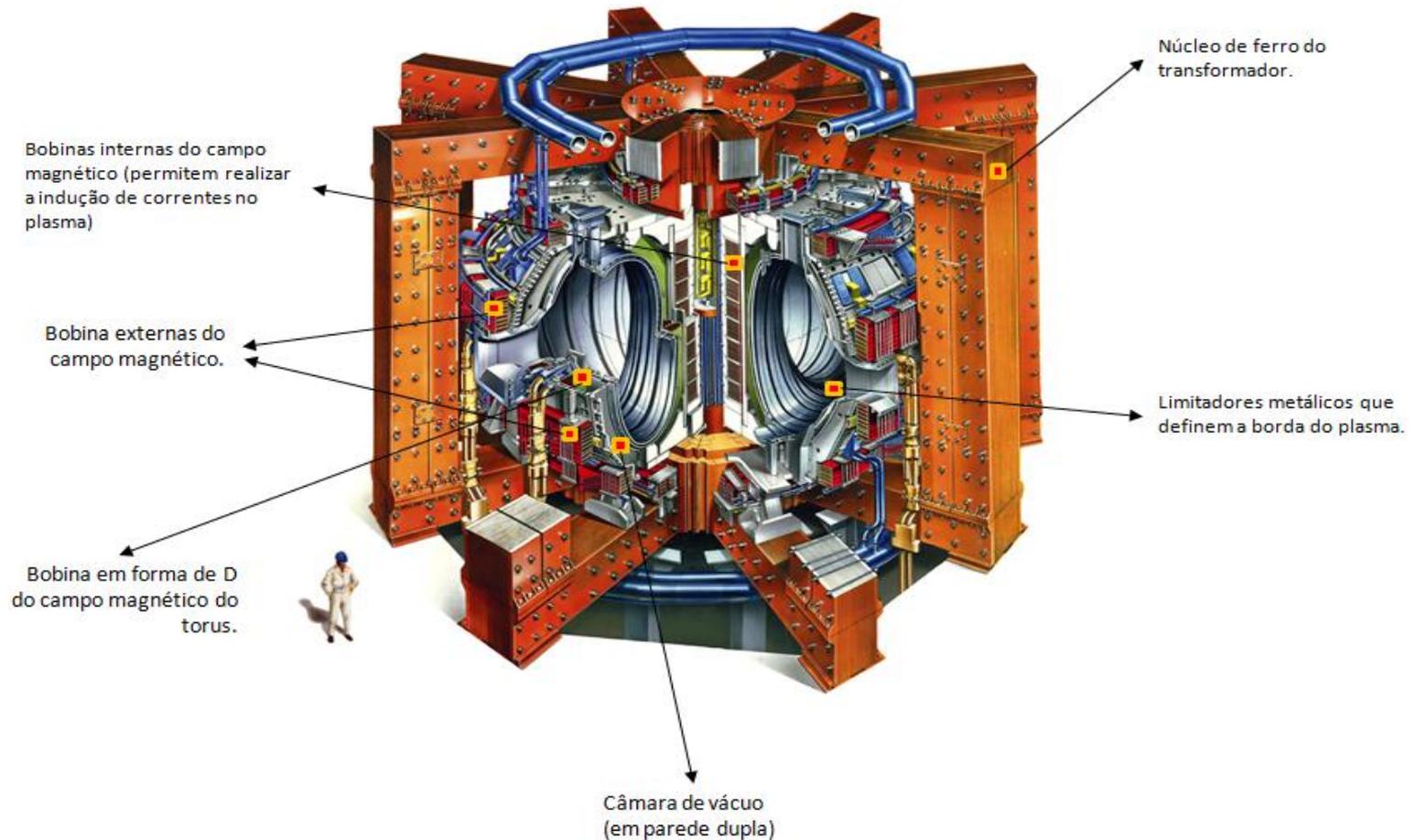
Tokamaks



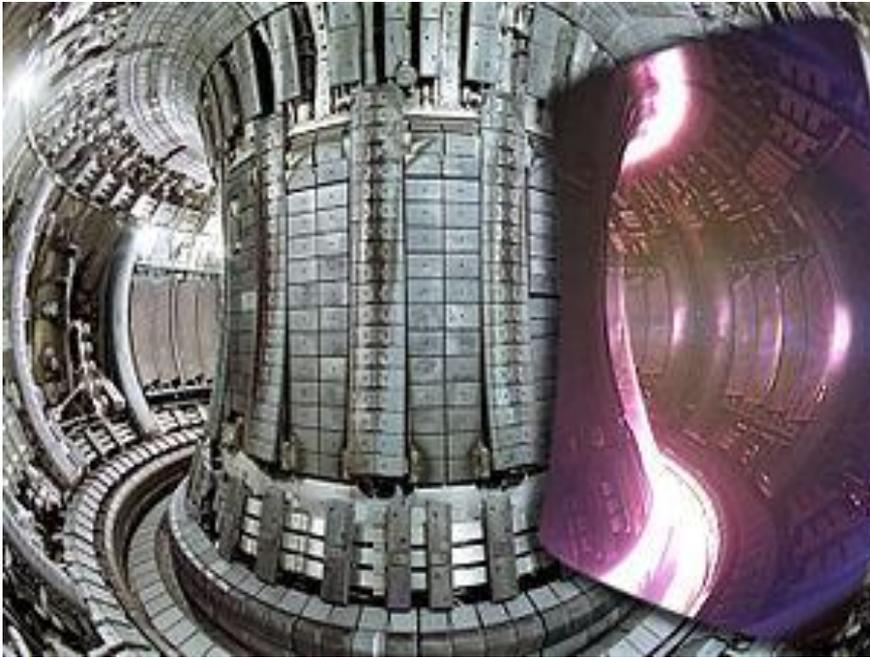
- O Plasma é confinado magneticamente.
- Tipo de Reator onde as linhas do campo magnético são fechadas em si mesmas, formando um toro.
- Busca a tornar a reação auto sustentada (Condição de Ignição)
- O ganho Q é definido como a razão entre a energia liberada e a energia consumida

3 Fusão Nuclear

O JET



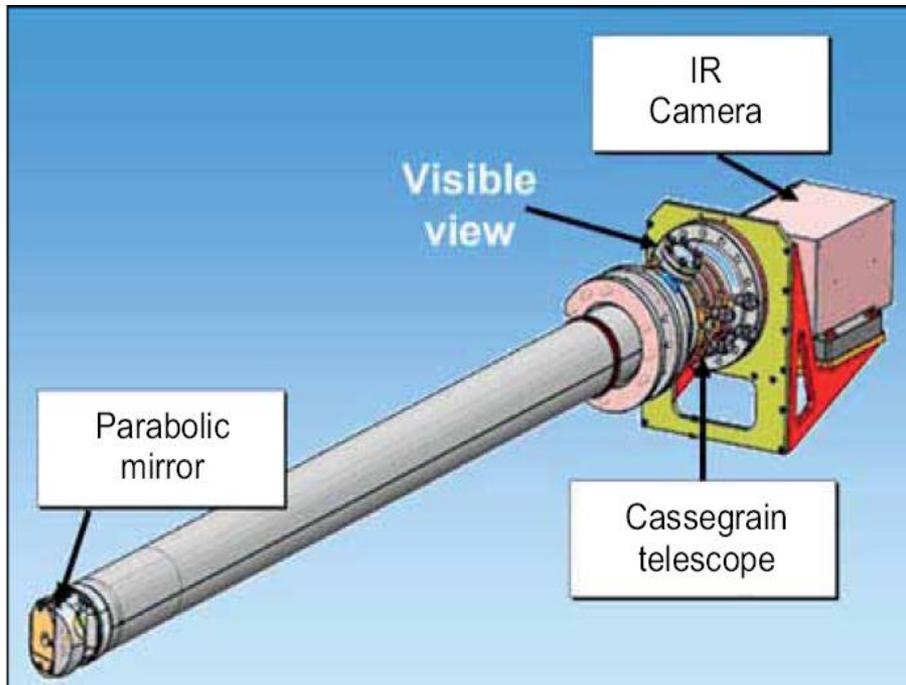
4 Processamento de Imagens para o JET



- A recuperação da informação
- Extração de informações em tempo real
- Derivação da informação física quantitativa

4 Processamento de Imagens para o JET

As Câmeras

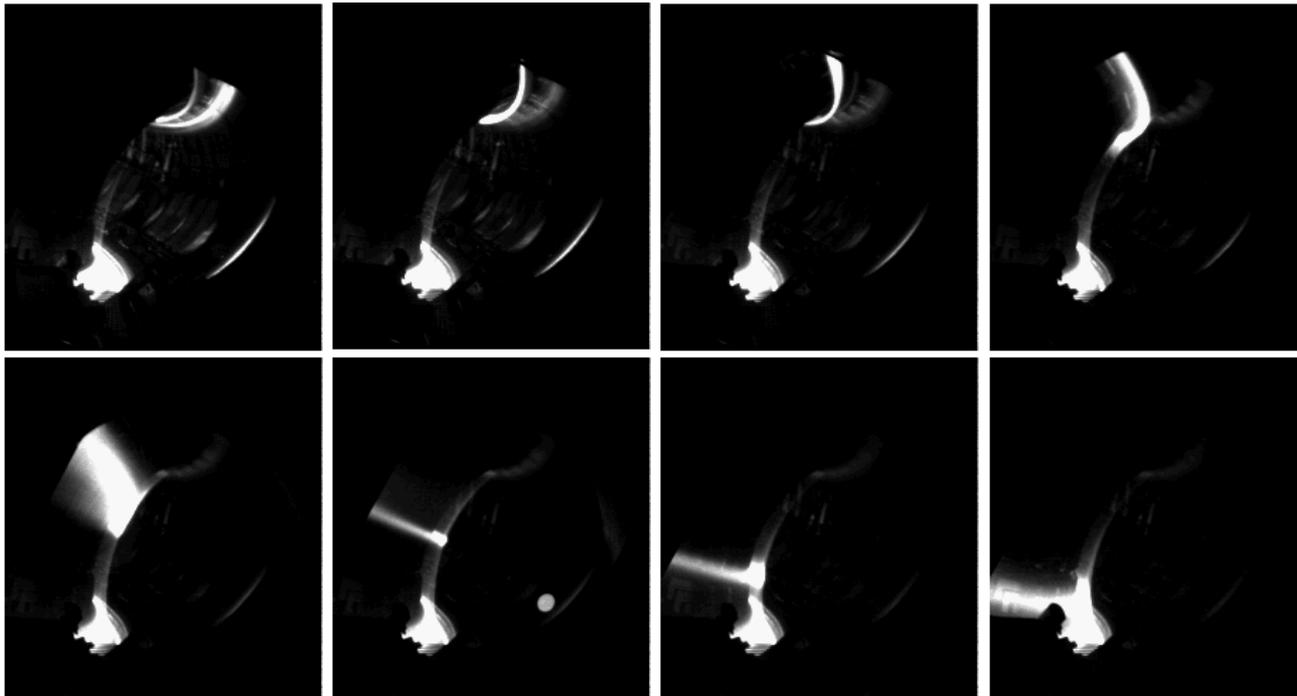


- Taxa de aquisição de 30.000 quadros por segundo
- Resolução de 256 x 256 pixels.
- A câmera possui uma saída de imagens no formato PNG.
- Montadas em um endoscópio

4 Processamento de Imagens para o JET

Os *MARFEs*

- Abreviação para o fenômeno *Multifaceted Asymmetric Radiation From the Edge*
- O MARFE é caracterizado como um anel brilhante concêntrico de radiação.

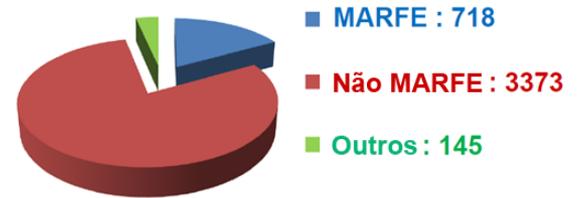


4 Processamento de Imagens para o JET

A Base de Dados

- 22 vídeos da câmera APX-RS
 - 3300 imagens
 - 4236 padrões
- Classificados em MARFE, Não-MARFE ou Outros

Estadística da Base de Dados



Coordenadas do baricentro da região

Momentos Hu e Posição da Região								ID da Região				
1,0683	0,9509	0,3114	0,0387	-0,0038	-0,0339	0,002	129,5646	68,3537	1	75	1	70050V2
0,3987	0,123	0,0004	0,0007	0	0,0002	0	57,137	194,5479	0	76	1	70050V2
0,9072	0,7635	0,055	0,0063	-0,0001	-0,0053	0	131,8879	63,8972	1	76	2	70050V2

NÃO-MARFE = 0
MARFE = 1
Outros = 2

2ª região da imagem número 76 do video 70050 (2º vídeo)

4 Processamento de Imagens para o JET

Avaliação de Desempenho

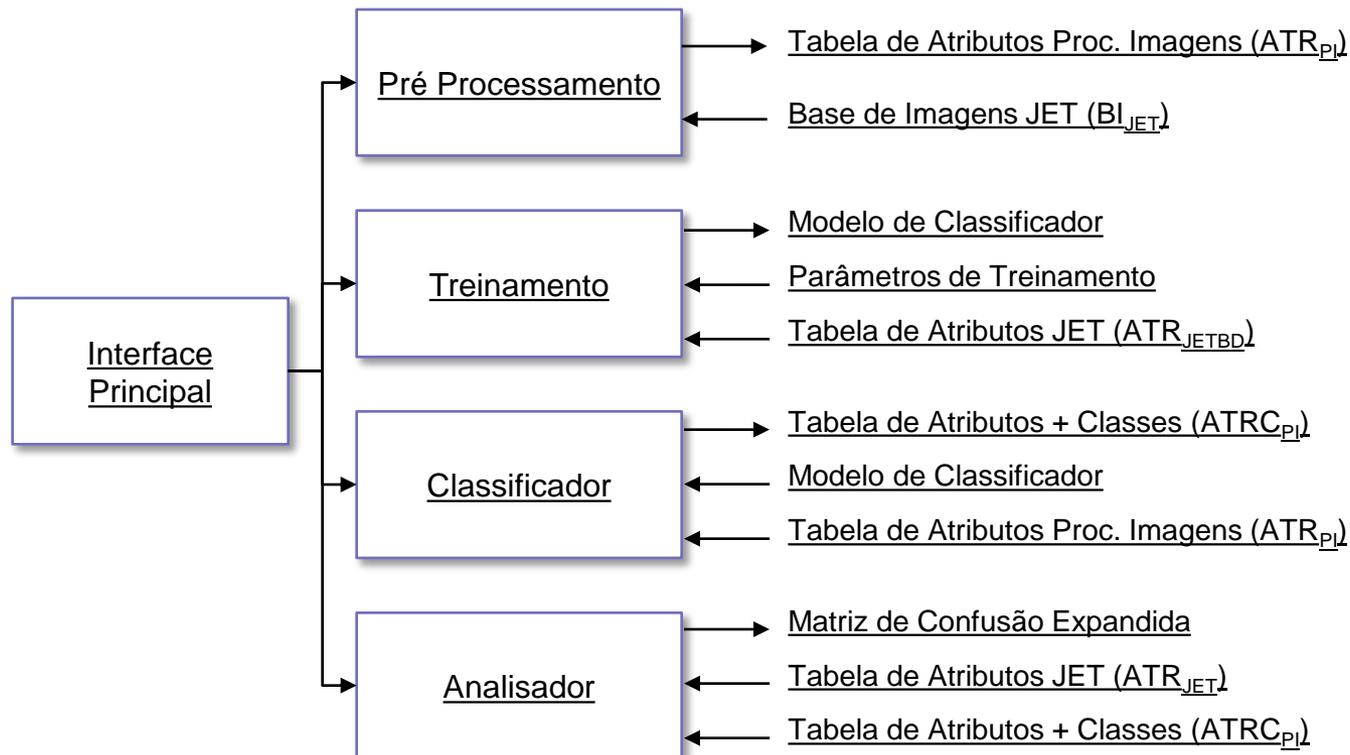
Matriz de Confusão Expandida		Image Data Base			NoBD
		0	1	2	3
Image Processing Algorithm	0	TN	FN	FN	
	1	(FP)	TP	(FP)	X _{BD}
	2	FN	FN	TN	
	3		X _{IP}		
NoIP					

- A precisão do sistema como um todo foi determinada utilizando o método de Matriz de Confusão
- O objetivo final do módulo de Análise de Desempenho (*Performance Analyzer*) é a construção da matriz de confusão para vários algoritmos de processamento de imagem.

5 Desenvolvimento e Caracterização

Desenvolvimento em MATLAB

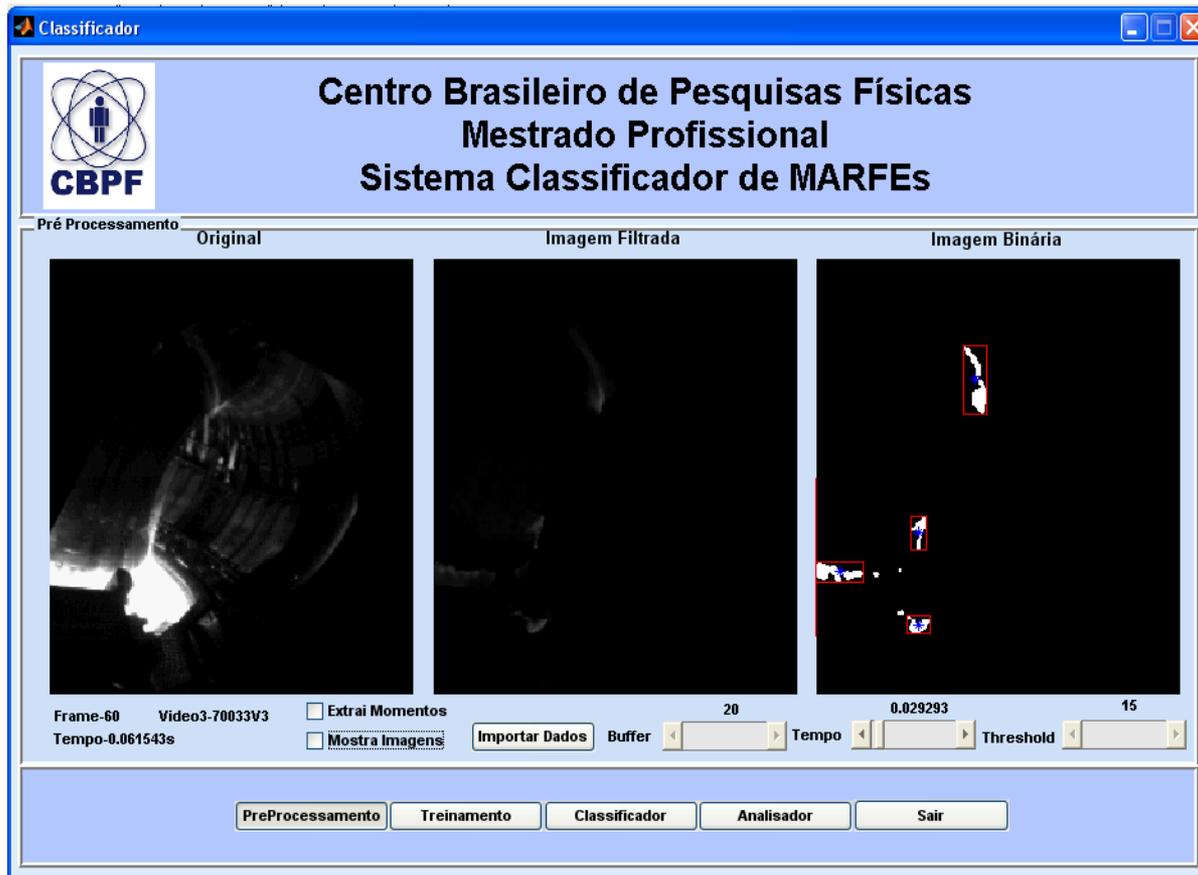
- Servir como plataforma de teste e avaliação das técnicas para a classificação de MARFES
- Ferramenta de Software - LibSVM (Biblioteca em C/C++)



5 Desenvolvimento e Caracterização

Desenvolvimento em MATLAB

PRÉ PROCESSAMENTO



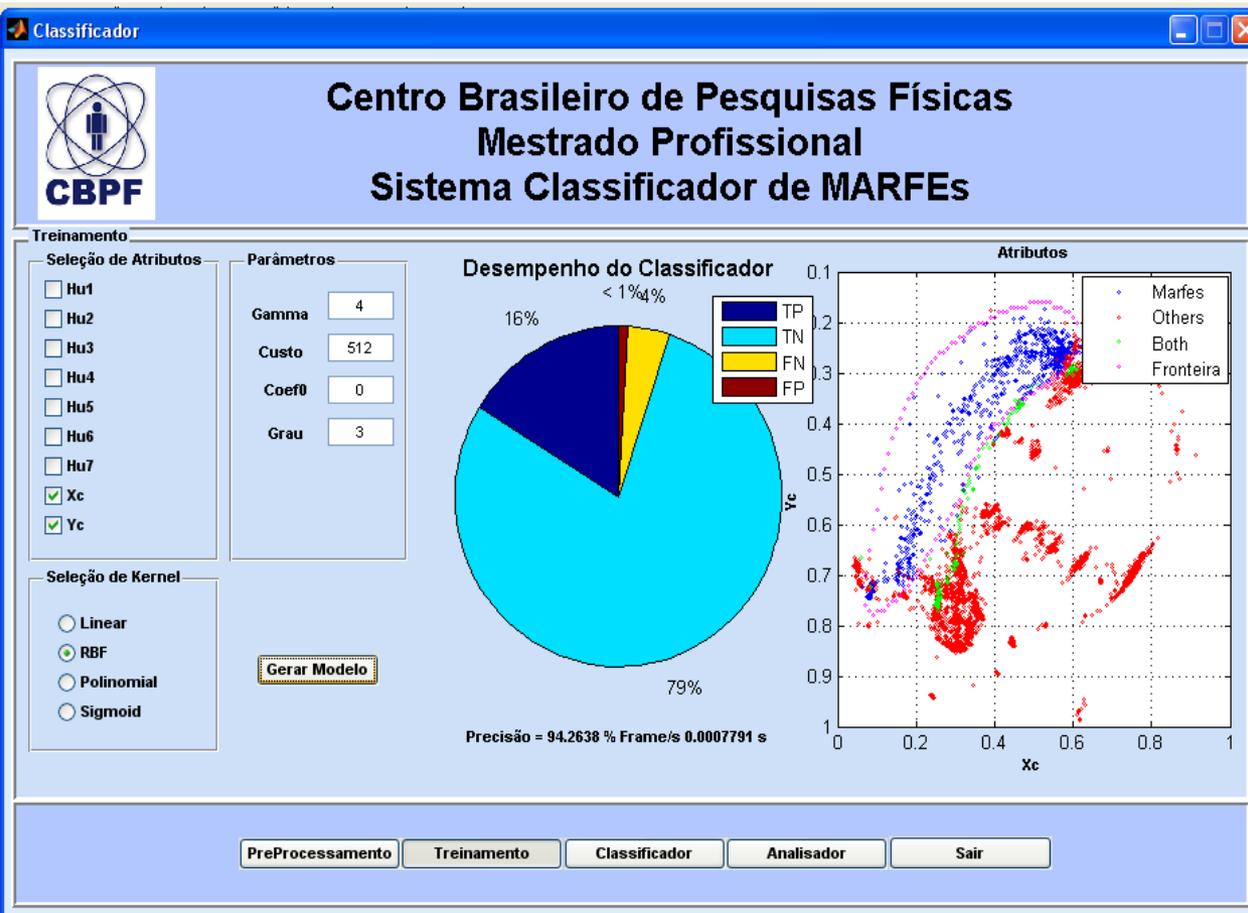
- Entrada
 - Base de Imagens(BI_{JET})
- Saída
 - Tabela ATR_{PI}
- Permite a variação do threshold e do número de imagens usadas para o cálculo da imagem de fundo

5 Desenvolvimento e Caracterização

Desenvolvimento em MATLAB

Treinamento

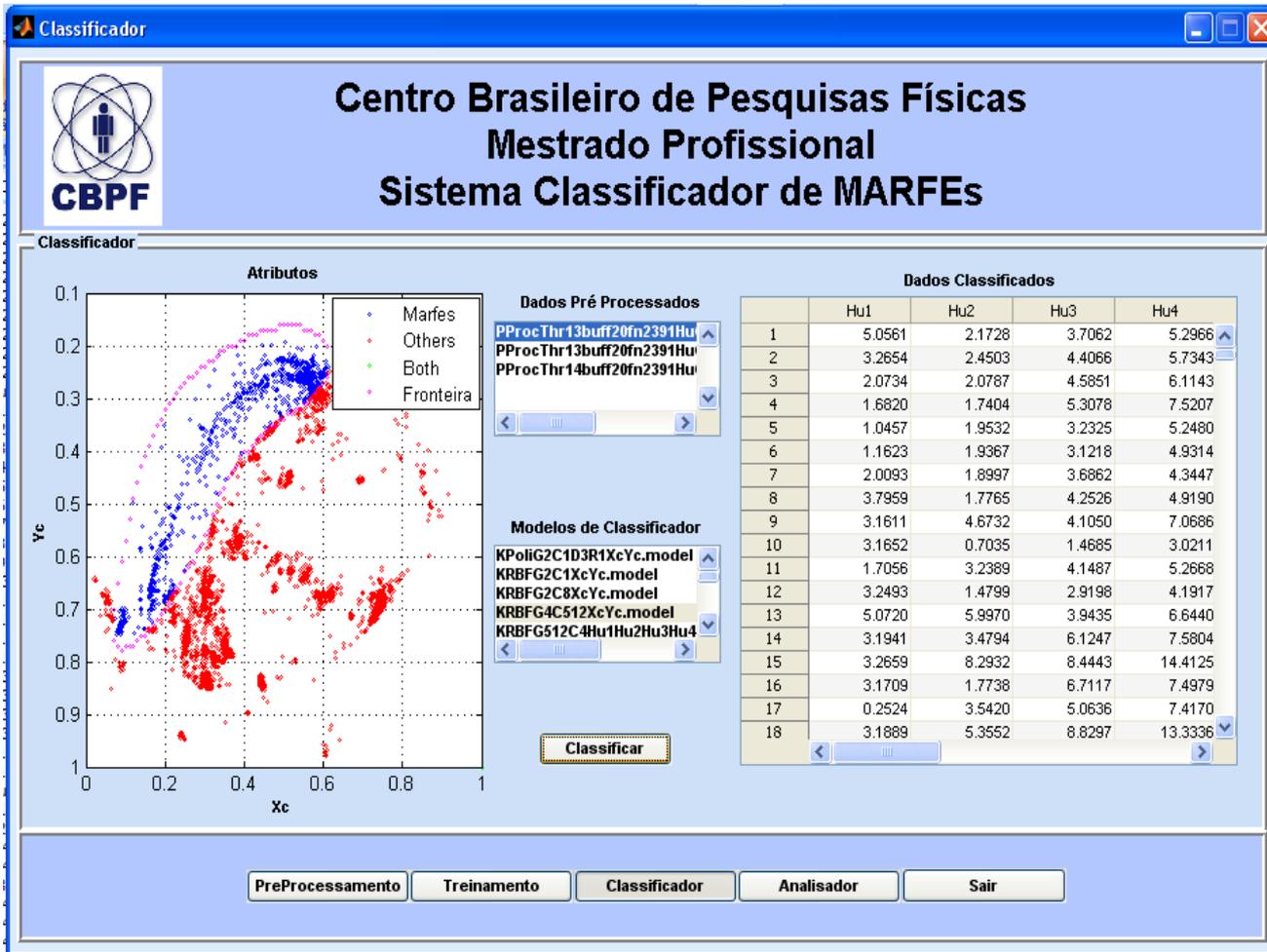
- Entradas
 - A tabela de atributos (ATR_{JETBD}).
 - Os parâmetros de treinamento.
- Saída
 - Modelo de Classificador
- 25% da base para teste do classificador e 75% para o treinamento



5 Desenvolvimento e Caracterização

Desenvolvimento em MATLAB

Classificador



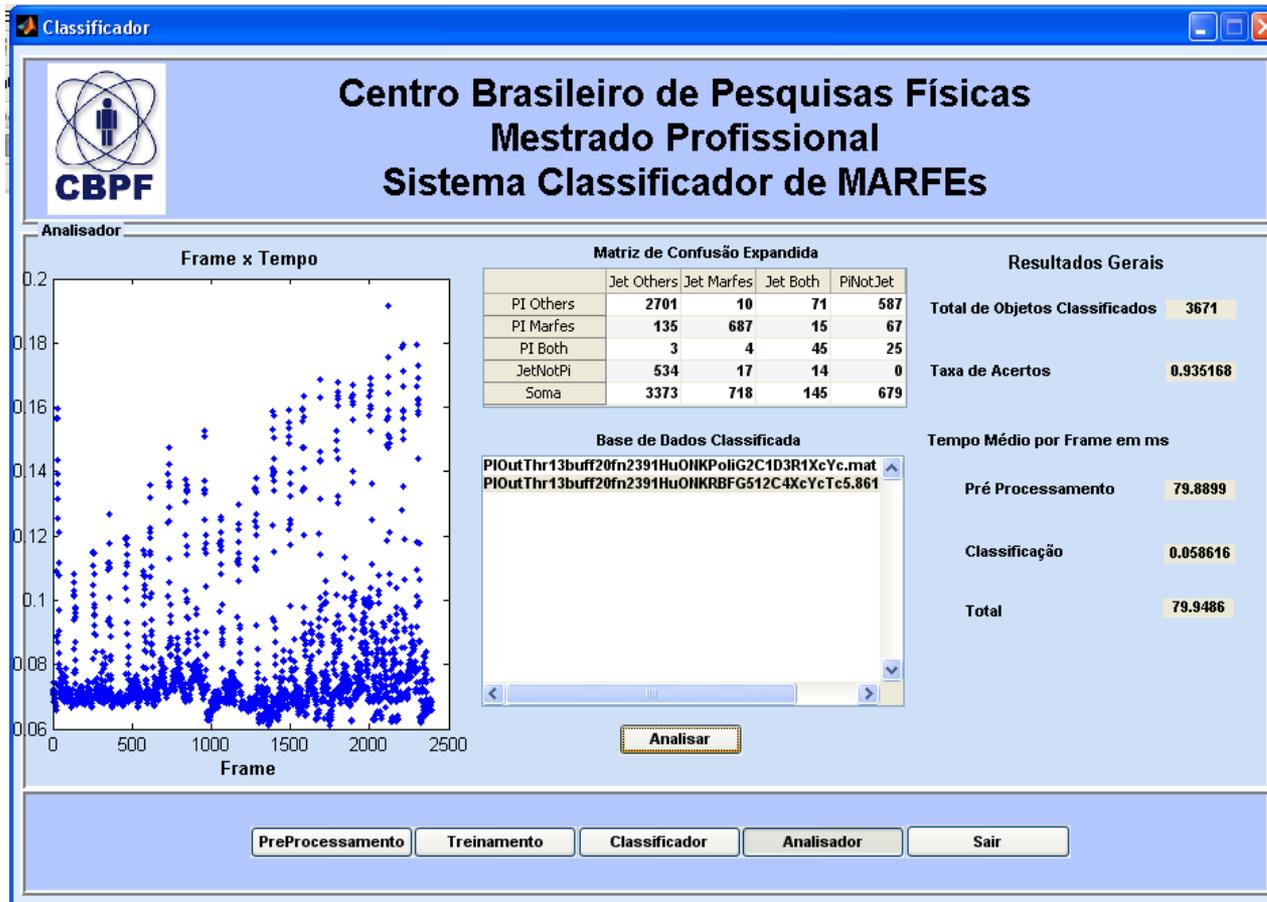
■ Entradas

- A tabela de atributos (ATR_{PI}).
- O modelo de Classificador SVM.

5 Desenvolvimento e Caracterização

Desenvolvimento em MATLAB

ANALISADOR



Entradas

- Tabela de atributos do JET (ATR_{JETBD}).
- Tabela de atributos do processamento de imagens, após a classificação ($ATRC_{PI}$).

Saída

- Matriz de Confusão expandida
- Análise Temporal
- Resultados Gerais

5 Desenvolvimento e Caracterização

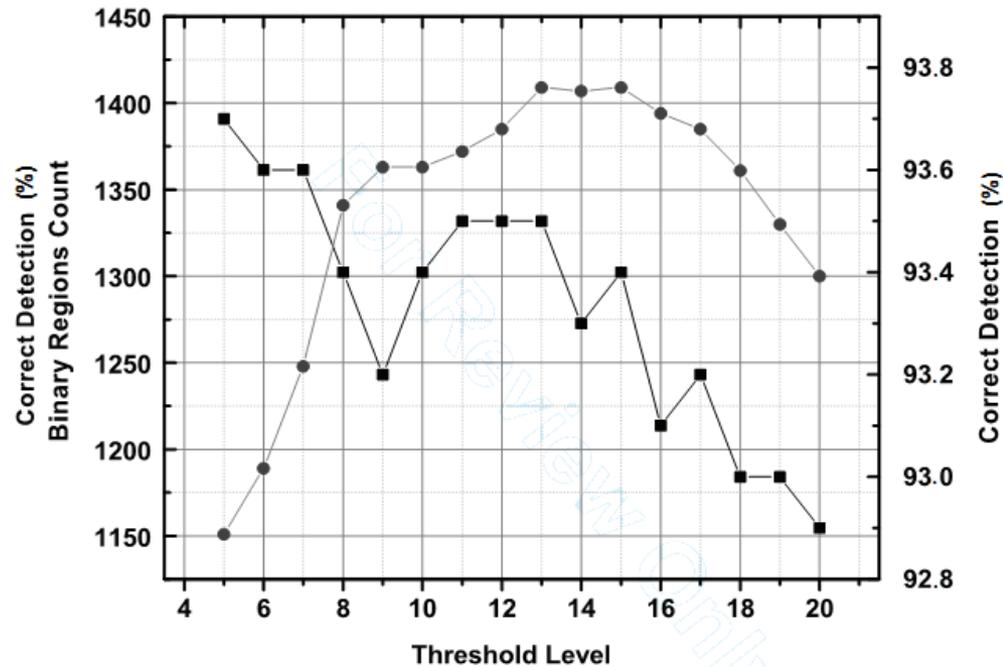
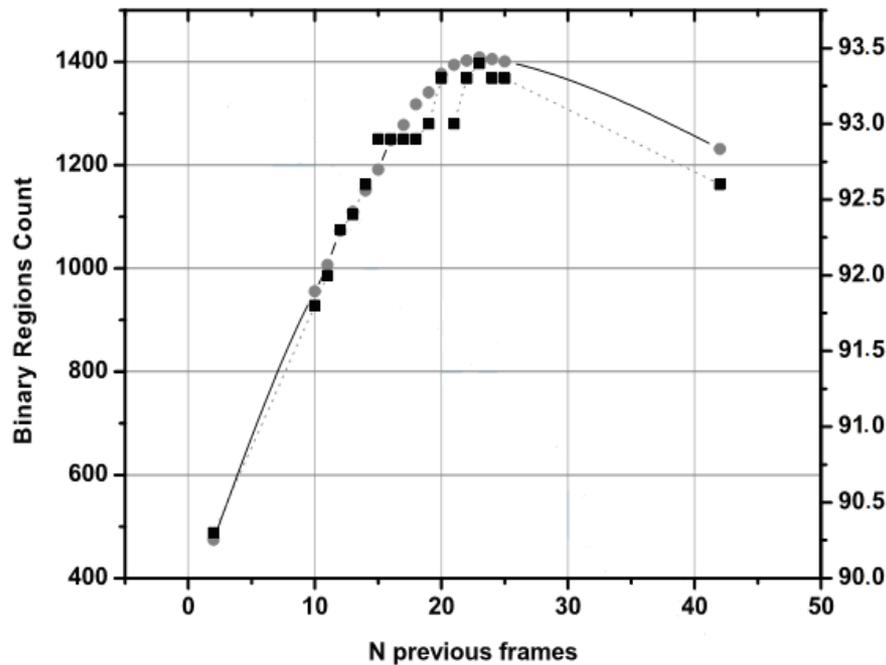
Desenvolvimento em C/C++

- ❑ O código foi testado em uma plataforma de computação Linux Intel 64 bits.
- ❑ A implementação buscou ao mesmo tempo, alta precisão e desempenho em tempo de execução.
- ❑ A versão final apresenta uma taxa de detecção correta de 93,3% e uma taxa média de processamento de imagem de 650 frames por segundo.
- ❑ Ferramentas de Software Utilizadas
 - ❑ OpenCV
 - ❑ LibSVM
- ❑ Módulo de Abertura da Imagem
 - ❑ Utiliza discos em memória com alta taxa de transferência 1300 MB/s a 3200 MB/s

5 Desenvolvimento e Caracterização

Desenvolvimento em C/C++

Módulo de Estimação da Imagem de Fundo e Binarização da Imagem



5 Desenvolvimento e Caracterização

Desenvolvimento em C/C++

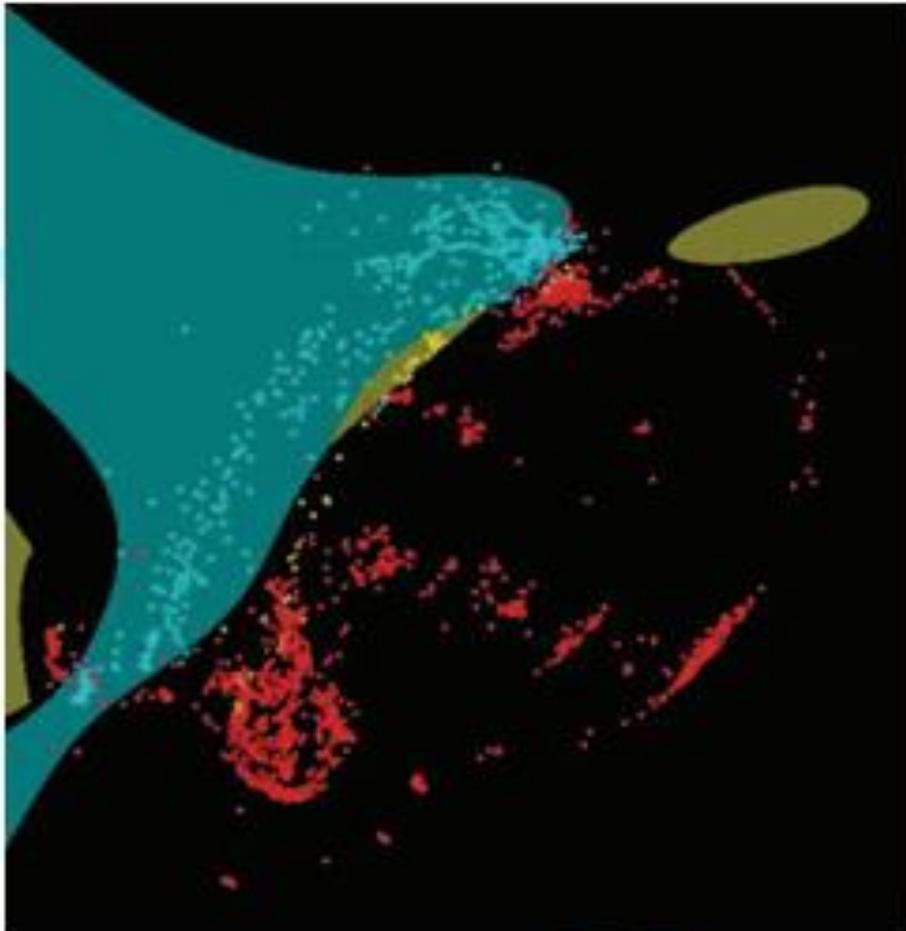
Módulo de Extração de Características

- Este módulo tem como objetivo a extração de características das regiões de pixels contíguos nas imagens binárias.
- Usamos a biblioteca OpenCV: cvBlobsLib. Esta biblioteca rotula e segmenta imagens binárias
- No a versão do algoritmo em C/C++ os momentos Hu's foram calculados porem não foram utilizados

5 Desenvolvimento e Caracterização

Desenvolvimento em C/C++

Módulo de Extração de Características



- O objetivo principal do módulo de classificação é determinar a qual classe pertence a forma encontrada
- 60% da base para Treino e 40% para Testes
- 11018 Modelos SVM foram testados
- o melhor resultado foi obtido por um modelo SVM baseado em um kernel polinomial de grau 10

5 Desenvolvimento e Caracterização

Desenvolvimento em C/C++

Análise de Desempenho

- Duas abordagens Distintas
 - Maior taxa de acertos
 - Menor Tempo de Processamento
- As medidas de tempo se referem ao método *walltime*, que é uma estimativa do tempo total consumido por todo o sistema
- A plataforma computacional de teste foi um nó Linux do Cluster computacional do CBPF
 - 2 placas mãe, cada uma com 2 processadores Intel CPU Xeon E5430 HarperTownQuad-Core.



TOp: Open Image
TSAv: Subtract Average
TBin: Binarization
TFHu: Feature Extraction
TCls: Classification
TRIs: Release Image

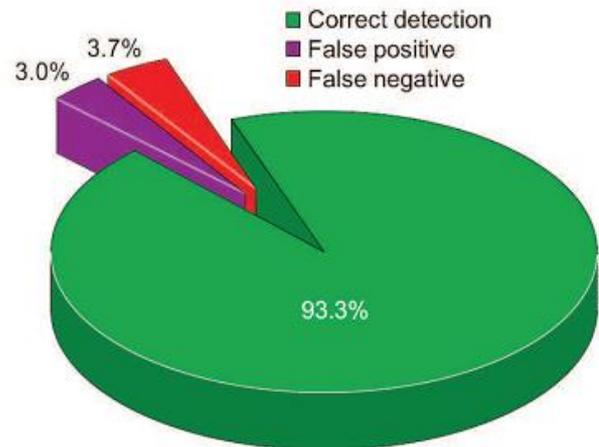
5 Desenvolvimento e Caracterização

Desenvolvimento em C/C++

Análise de Desempenho Resultados

Neste trabalho, apresentamos apenas o desempenho da classificação geral do sistema de processamento dedicados ao reconhecimento de MARFEs. O algoritmo de processamento de imagens processou 1.409 regiões

	Non MARFE	MARFE	Other	NoBD
Non MARFE	1073	2	39	401
MARFE	37	235	5	15
Other	10	1	7	2
NoIP	248	52	4	0



6 Conclusões e Perspectivas

- Neste trabalho apresentamos um conjunto de técnicas de processamento de imagens para reconhecimento e detecção de MARFEs em imagens do Laboratório JET
- A caracterização dos algoritmos abordou duas direções:
 - Em termos das diversas técnicas para o processamento da informação com um enfoque principal na análise padrões por momentos invariantes e na precisão do sistema de classificação
 - Em termos de sua velocidade total de processamento
- O trabalho se consolidou como uma efetiva contribuição as necessidades de caracterização de MARFE dentro da colaboração.
- Está em discussão neste momento, um novo sistema de aquisição tendo como base uma placa de aquisição que receba as imagens diretamente da câmera KL8.
- Um dos objetivos futuros é que parte dos algoritmos desenvolvidos possam ser migrados para dispositivos eletrônicos de programação em hardware, como as FPGAs.
- Avaliação de outras técnicas como RNA e PCA não linear.
- Avaliação dos algoritmos de execução paralela em ambientes multiprocessados
- Usar outros atributos para caracterizar os Marfes.
- Usar processamento via GPU.

Sumário Aula 03

I - Introdução ao Reconhecimento de Padrões

1. Definição de Padrão e Classe
2. Vetor de características
3. Exemplo
4. Projeto do classificador
5. Abordagens típicas
6. Exemplos

II – Introdução as Redes Neurais Artificiais

1. Definição
2. Modelo de um neurônio artificial
3. Modelo de uma rede MLP
4. Algoritmo Backpropagation
5. Exercícios

O Cérebro Humano

O cérebro humano processa as informações de uma forma completamente diferentes que o computador

O cérebro é um computador altamente complexo, não-linear e paralelo

Ele tem capacidade de organizar os neurônios de forma realizar certos processamentos muito mais rapidamente que o más rápido computador digital.

Ex:

- Reconhecimento de uma bola por uma criança
- Sonar de um morcego - distância, velocidade relativa, tamanho, azimute, etc...

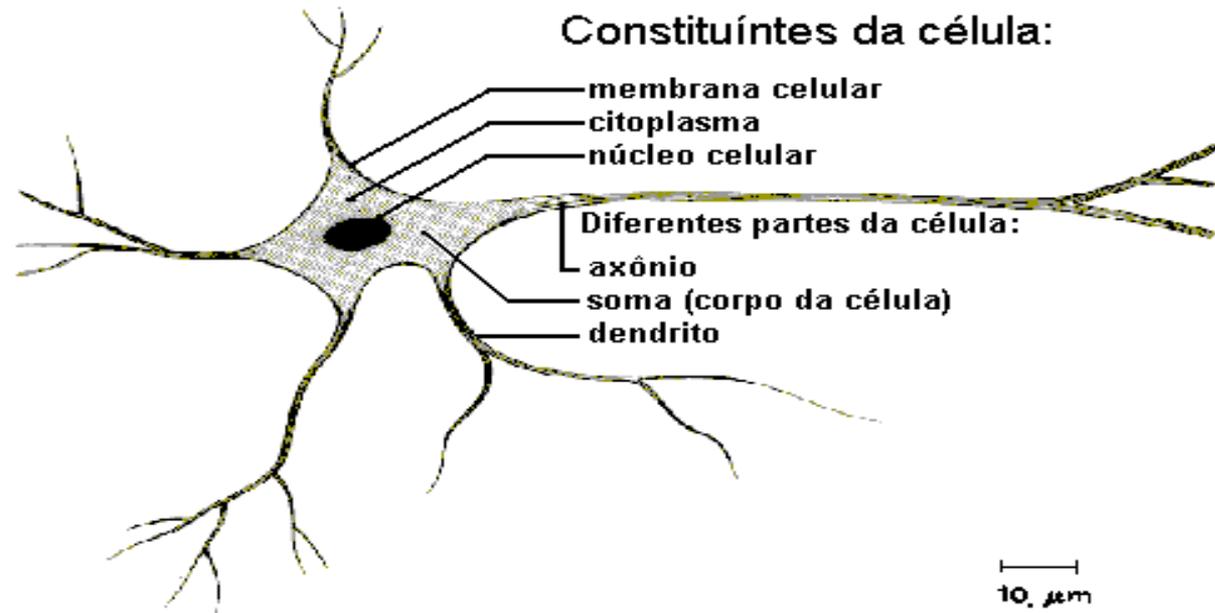
Como o cérebro humano e o morcego fazem isso?

- O cérebro tem uma grande estrutura e habilidade de desenvolver suas próprias regras através do que chamamos de experiência
- Us neurônios que formam o cérebro tem a característica de plasticidade, permitindo que o sistema nervoso se adapte ao meio ambiente

Neurônio Biológico

PARTES

- dendritos
- corpo celular
- axônio
- sinapse

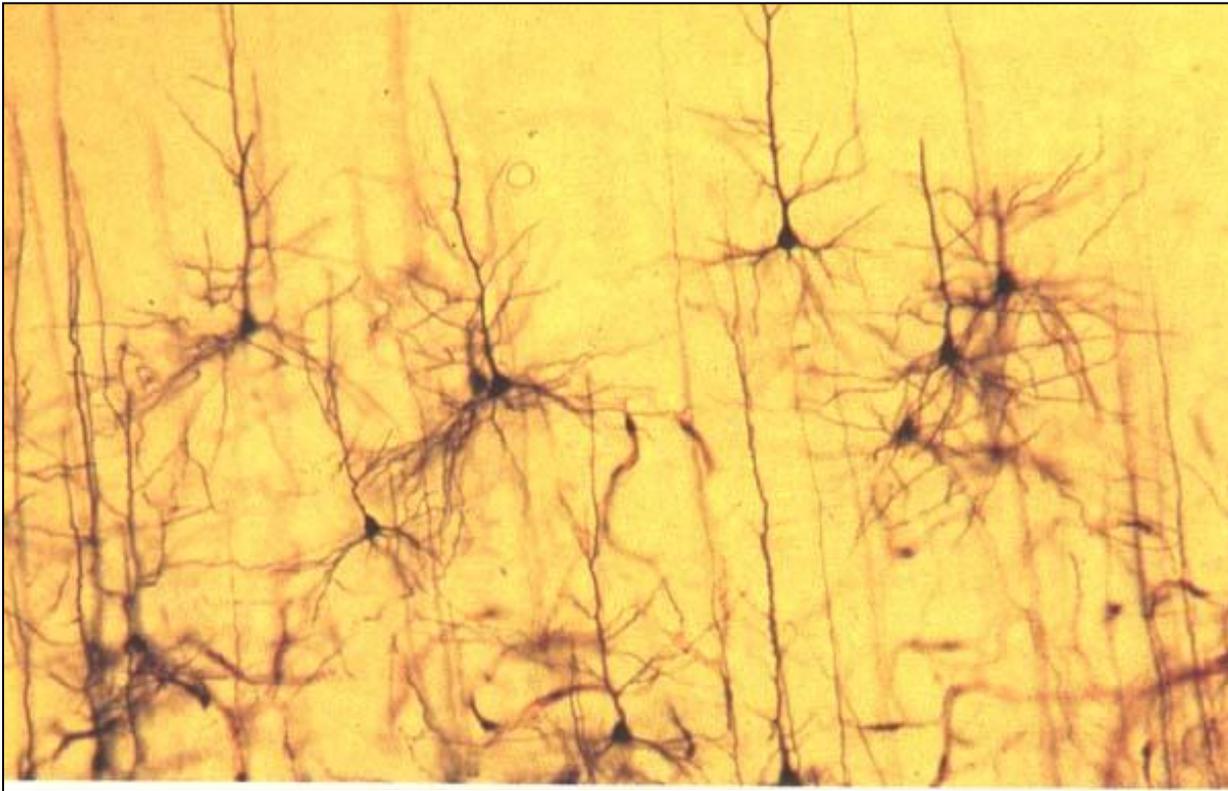


Idéia de entrada, processamento e saída

Rede Biológica

APRESENTAÇÃO

- formam malhas complexas
- comunicação entre neurônios pelas sinapses



Como reproduzir tal complexidade?

O que é uma Rede Neuronal?

“ É um processador maciçamente paralelo, constituído por unidades de processamento simples (neurônios), que tem propensão natural para armazenar conhecimentos experimental e torná-lo disponível para seu uso.

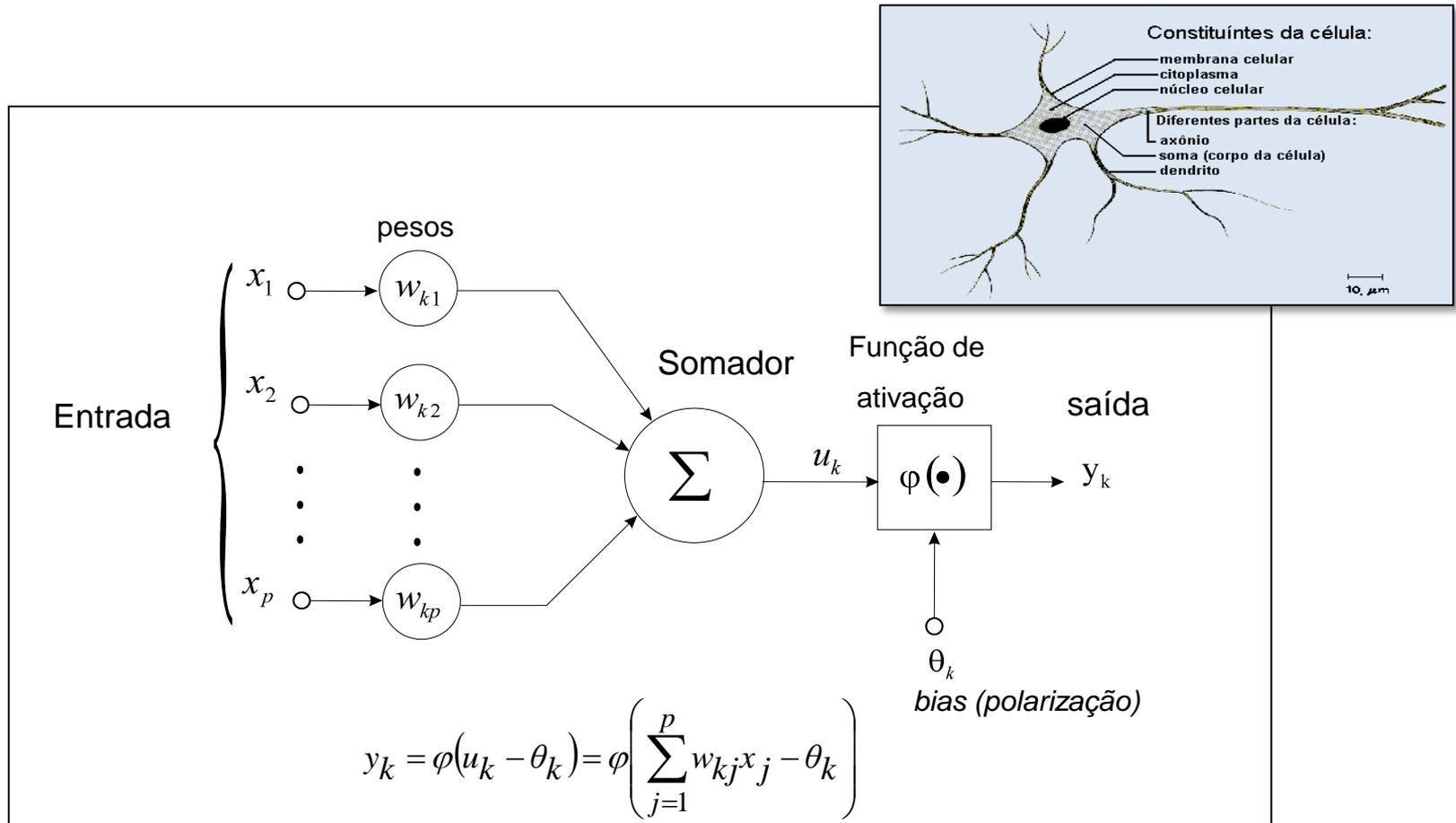
Ela se assemelha ao cérebro em dois aspectos:

O conhecimento é adquirido através da aprendizagem

Forças de conexão entre neurônios – os pesos sinápticos-, são utilizadas para armazenar o conhecimento adquirido...”

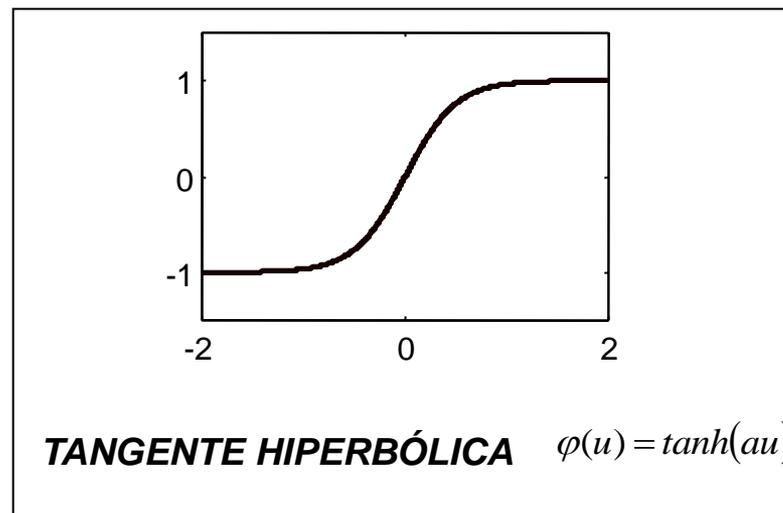
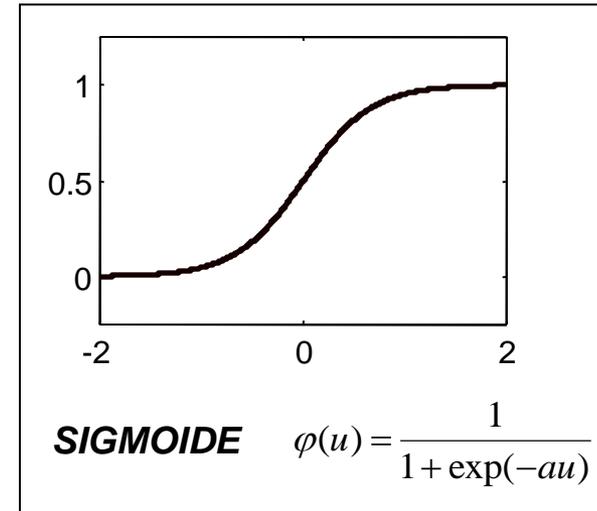
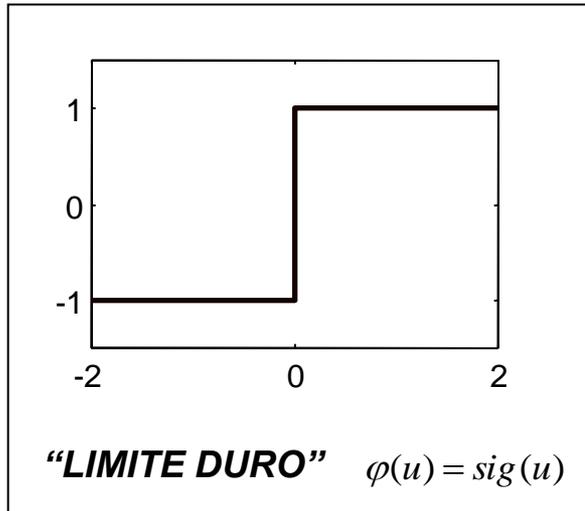
Simon Haukin

Modelo de um Neurônio



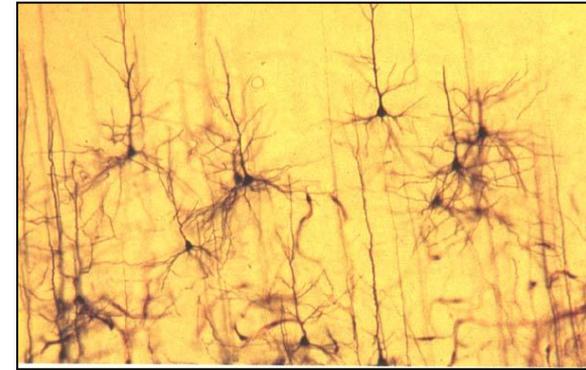
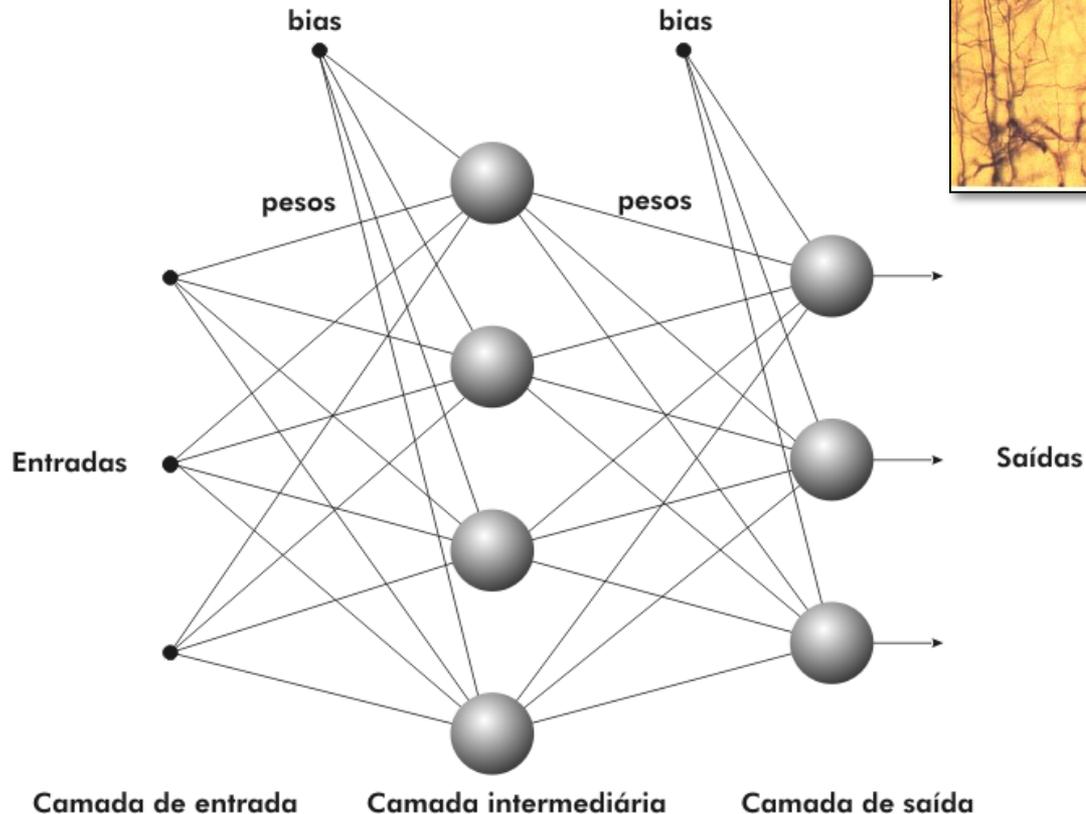
Semelhança entre os modelos

Funções de Ativação



Modelo de uma Rede

Disposição em camadas. Camadas ocultas
Comunicação entre neurônios entre camadas



Benefícios das Redes Neurais

Não linearidade

Mapeamento Entrada-Saída

Adaptabilidade

Resposta a evidências

Tolerâncias a falhas

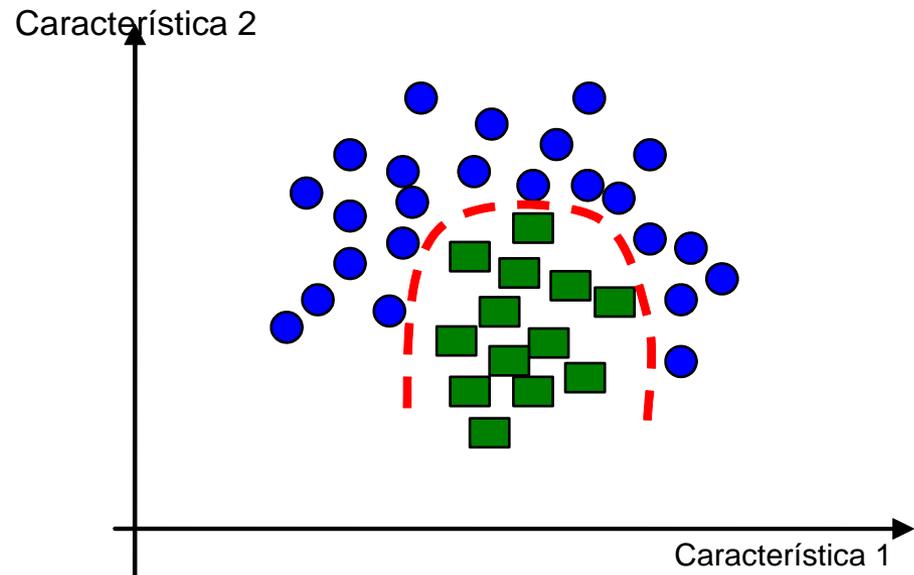
Implementação VLSI

Não-Linearidade

Um neurônio artificial pode ser linear ou não-linear

Uma rede é não-linear quando constituída por neurônios não lineares

Esta propriedade é importante porque a natureza (dados de entrada) é principalmente não-linear



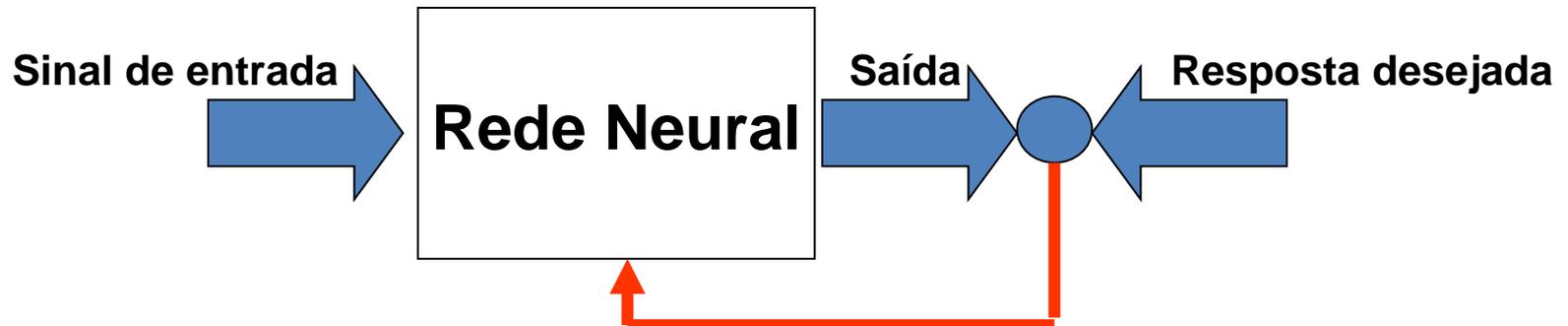
Mapeamento Entrada-Saída

Aprendizagem supervisionada. Aplicação de um conjunto de amostras de treinamento rotuladas

Cada conjunto consiste de um sinal de entrada único com a saída desejada

Apresenta-se para a rede um exemplo, e os pesos sinápticos da rede são modificados para minimizar a diferença entre a resposta desejada e a resposta da rede, de acordo com um critério estatístico apropriado

O treinamento é repetido para muitos exemplos até que a rede alcance estabilidade



Adaptabilidade

Capacidade de adaptar os pesos sinápticos a modificações

Pode ser facilmente *retreinada* para lidar com pequenas modificações nas condições operativas

Em ambientes não-estacionários (onde as estatísticas mudam com o tempo), uma rede pode ser projetada para modificar seu pesos em tempo real.

Perceptrons de Múltiplas Camadas

Treinamento supervisionado

Algoritmo de retro-propagação do erro
(*error backpropagation*)

Dois passos :

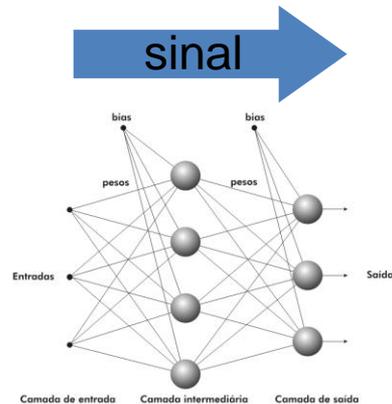
propagação (para frente - *Feedforward*)

retro-propagação (para a trás - *backpropagation*)

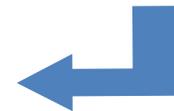
entrada saída

$$\bar{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix}$$

$$\bar{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix}$$



$$\tilde{y} = \begin{bmatrix} \tilde{y}_1 \\ \vdots \\ \tilde{y}_N \end{bmatrix} \quad y = \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} \quad e = y - \tilde{y} = \begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix}$$

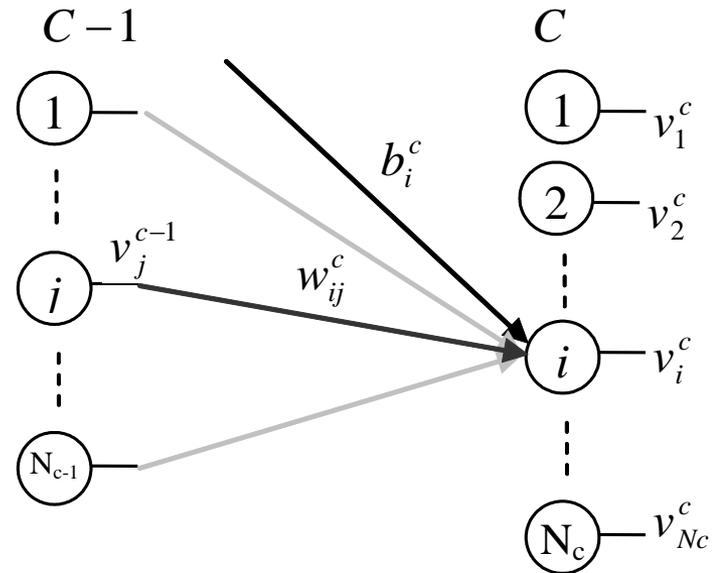


Fase de Propagação

Propagação:

Um vetor de entrada é aplicado a entrada da rede. Seu efeito se propaga camada trás camada.

Os pesos sinápticos são todos fixos
Um conjunto de saída é produzido –
resposta real da rede



$$u_i^c = \sum_{j=1}^{N_{c-1}} w_{ij}^c v_j^{c-1} + b_i^c$$

$$v_i^c = tgh(u_i^c)$$

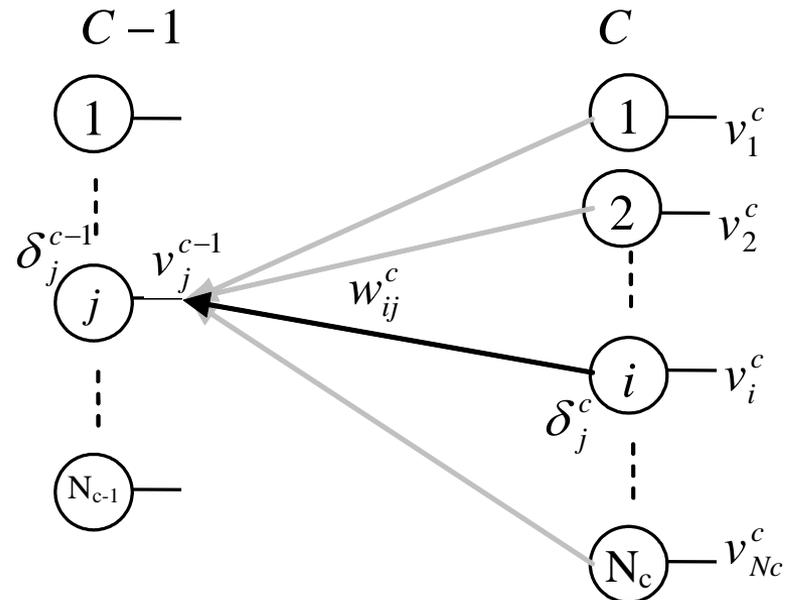
Fase de Retro-propagação

Retro-propagação:

A resposta real da rede é subtraída de uma resposta desejada (alvo) para produzir o sinal de erro

O sinal de erro é propagado através da rede contra a direção das conexões sinápticas – daí o nome

Os pesos sinápticos são ajustados para fazer que a resposta real da rede se mova para mais perto da resposta desejada (estaticamente)



$$\delta_i^{c-1} = [1 - (v_j^{c-1})^2] \cdot \sum_{j=1}^{N_c} w_{ij}^c \delta_j^c$$

$$\delta_i^c = [1 - (v_j^c)^2] \cdot e$$

Fase de Atualização

Existem muitos métodos de atualização. Regra Delta
 α – taxa de aprendizado. Quanto menor seja α , menor serão as variações dos pesos sinápticos

$$\Delta w_{ij}^c = 2\alpha v_j^{c-1} \delta_i^c$$

$$\Delta b_i^c = 2\alpha \delta_i^c$$

$$w(n+1) = w(n) + \Delta w(n)$$

$$b(n+1) = b(n) + \Delta b(n)$$

Conjunto de Dados

Geralmente são usados 3 conjuntos de dados

Treinamento. Conjunto usado para atualizar os pesos

Validação. Para validar como esta se comportando o treinamento. Determinar a parada do algoritmo

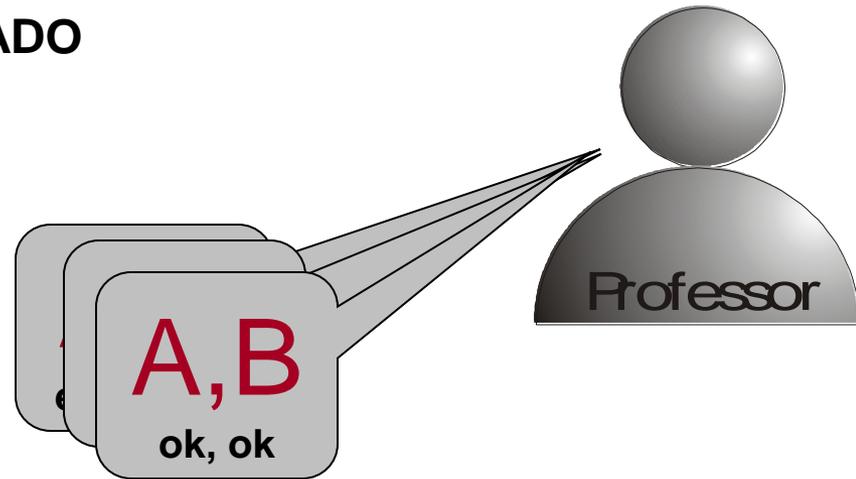
Teste. Avaliação final do modelo

Os conjuntos de treinamento e validação participam na criação do modelo

Treinamento iterativo

TREINAMENTO SUPERVISIONADO

- conjunto de treinamento



PADRÕES

A
B



1ª ITERAÇÃO

~~D~~
~~R~~

2ª ITERAÇÃO

A
~~P~~

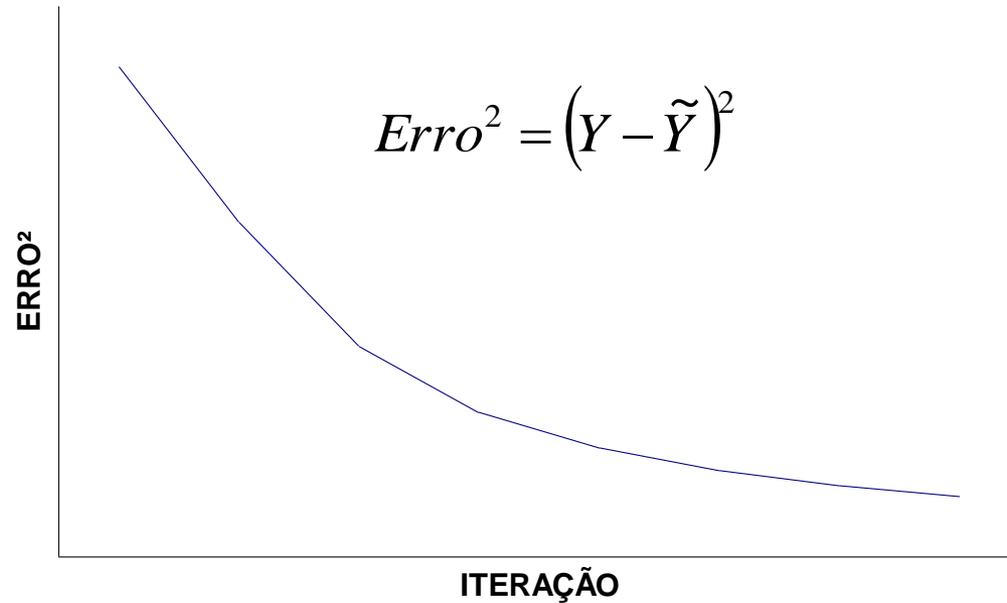
3ª ITERAÇÃO

A
B

O erro diminui !!!

$Y \rightarrow$ Valor desejado

$\tilde{Y} \rightarrow$ Valor obtido na saída da rede



Minimização do Erro

MÉTODO DO GRADIENTE DESCENDENTE

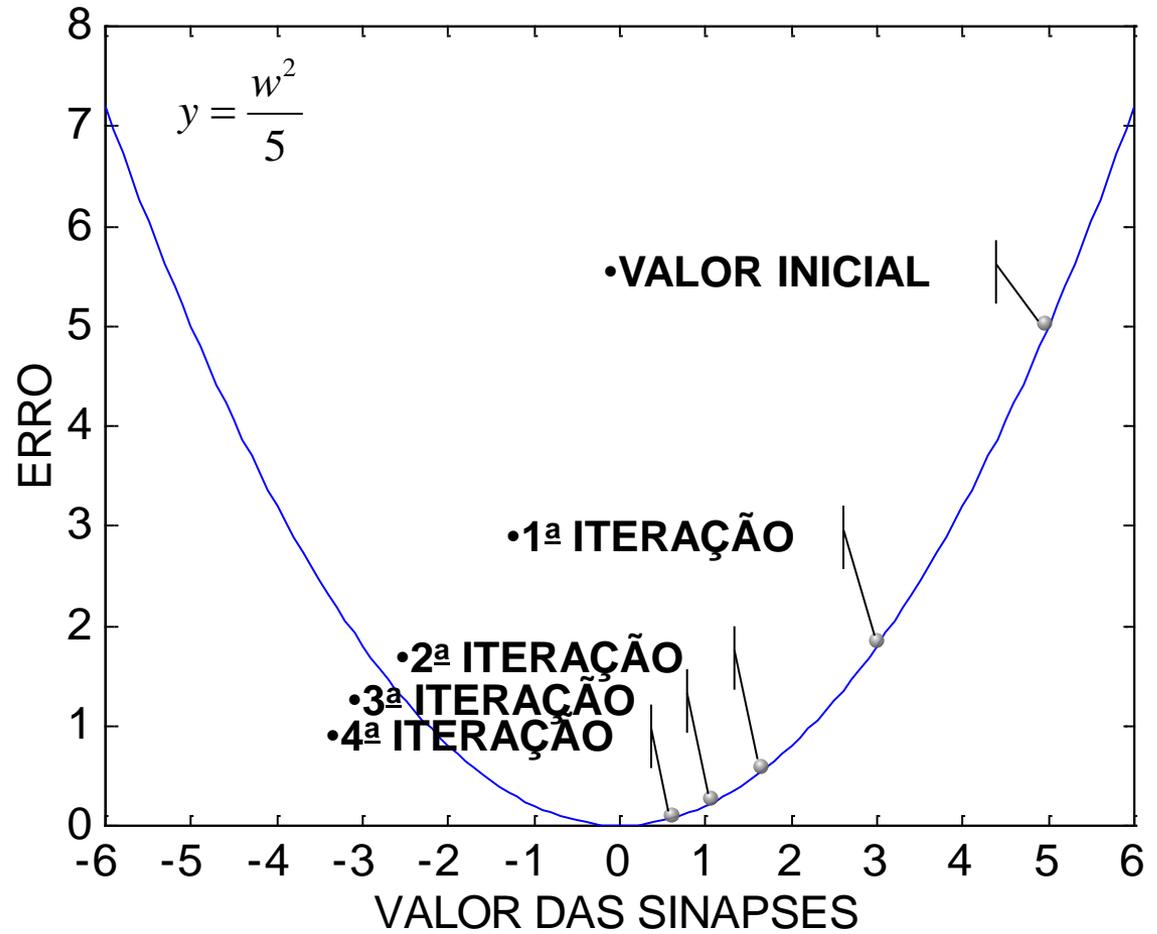
- minimização do erro
- mesma direção e sentido oposto ao gradiente

$$\Delta w(n) = -\eta \frac{\partial y(n)}{\partial w}, \eta = 1$$

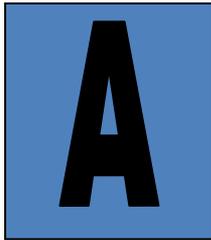
$$\Delta w(n) = \frac{\partial y(n)}{\partial w} = -\frac{2w}{5}$$

$$w(n+1) = w(n) + \Delta w(n)$$

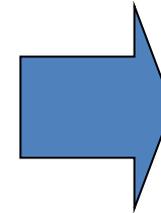
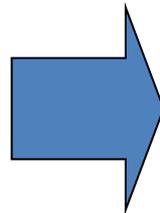
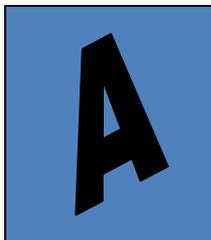
n	w(n)	$\Delta w(n)$
0	5	-2
1	3	-1,2
2	1,8	-0,72
3	1,08	-0,43
4	0,65	-0,26



Como incorporar invariâncias no projeto de RP



A própria imagem pode ser o vetor de características



‘A’



Incorporando Invariância

Invariância por estrutura

Redes de estruturas mais complexas. As sinapses são criadas de forma que versões transformadas da mesma entrada sejam forçadas a produzir a mesma saída

Invariância por treinamento

Muito usado. A rede é treinada apresentando-se um número de exemplos diferentes correspondente a diferentes transformações.

Espaço de características invariantes

Extrair *características* que caracterizem o conteúdo essencial da informação e que sejam invariantes a transformações das entradas

IX Escola do CBPF 2012

LISTA DE EXERCÍCIOS DE RECONHECIMENTO DE PADRÕES

REDES NEURAIS

Calcule os novos pesos sinápticos usando o algoritmo de *backpropagation* com a regra Delta. Todos os neurônios tem uma função de transferência tangente hiperbólica. A taxa de aprendizado é igual a 0.1.

$$u_i^c = \sum_{j=1}^{N_{c-1}} w_{ij}^c v_j^{c-1} + b_i^c$$

$$v_i^c = \text{tgh}(u_i^c)$$

$$\delta_i^c = [1 - (v_j^c)^2] \cdot e$$

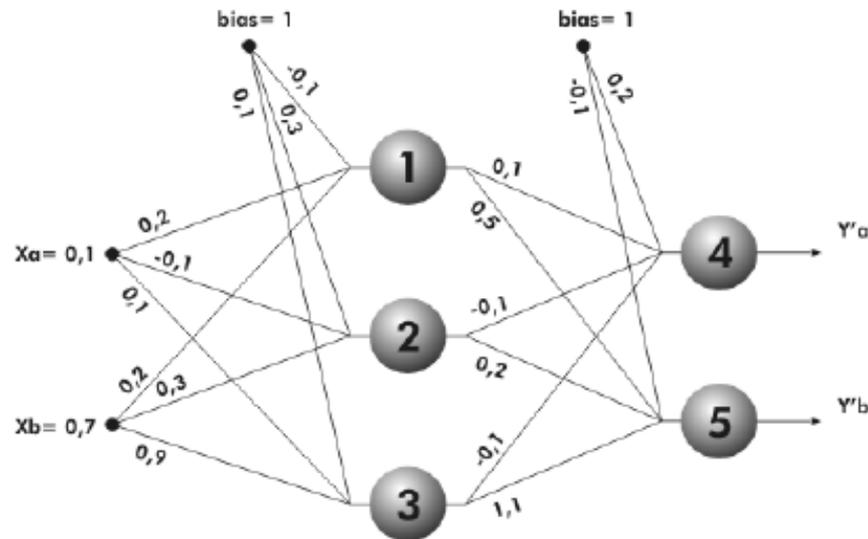
$$\delta_i^{c-1} = [1 - (v_j^{c-1})^2] \cdot \sum_{j=1}^{N_c} w_{ij}^c \delta_i^c$$

$$\Delta w_{ij}^c = 2\alpha v_j^{c-1} \delta_i^c$$

$$\Delta b_i^c = 2\alpha \delta_i^c$$

$$w(n+1) = w(n) + \Delta w(n)$$

$$b(n+1) = b(n) + \Delta b(n)$$



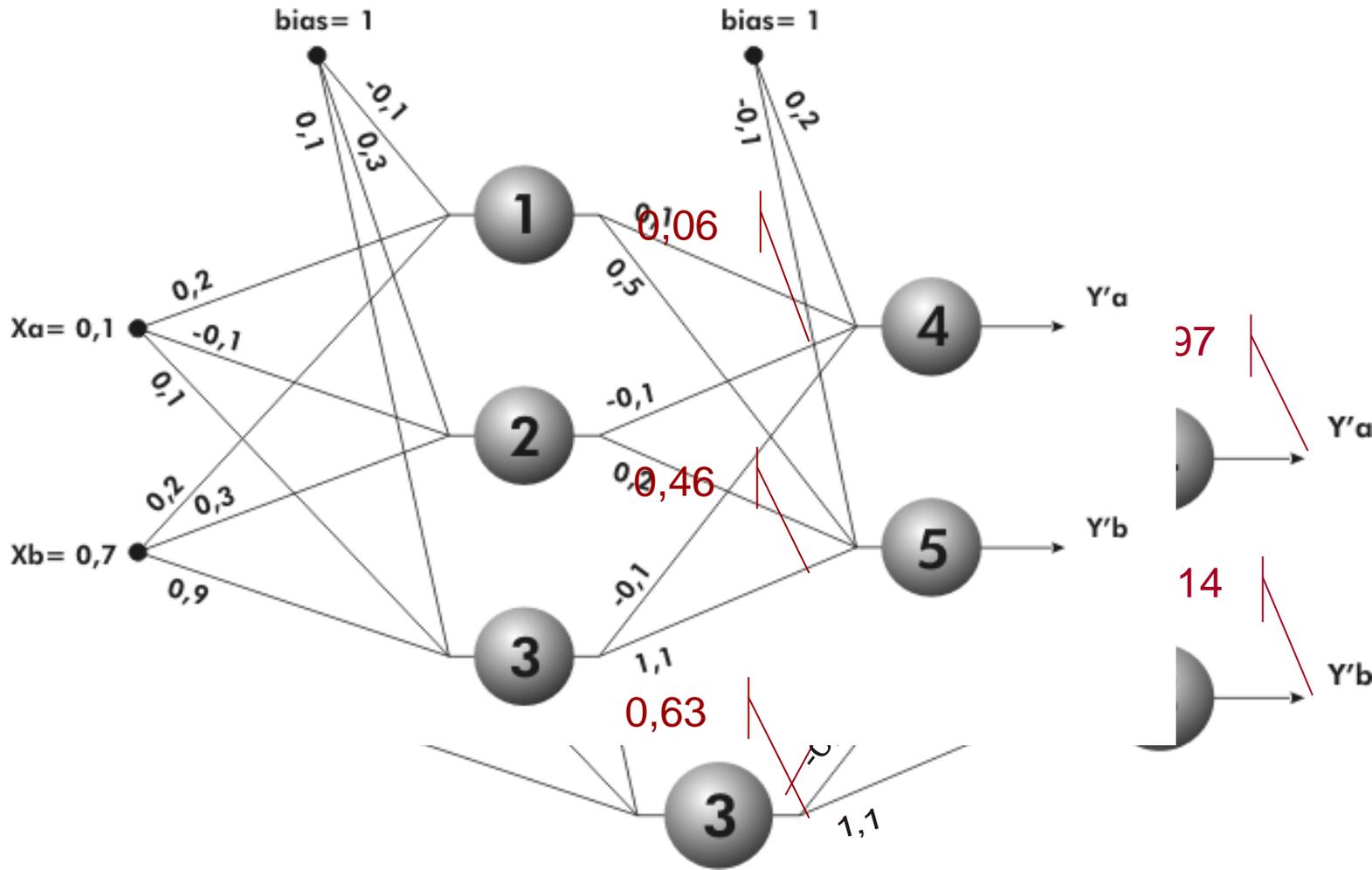
$$\bar{x} = \begin{bmatrix} 0,1 \\ 0,7 \end{bmatrix}$$

$$\bar{y} = \begin{bmatrix} 0,2 \\ 1 \end{bmatrix}$$

FEEDFORWARD

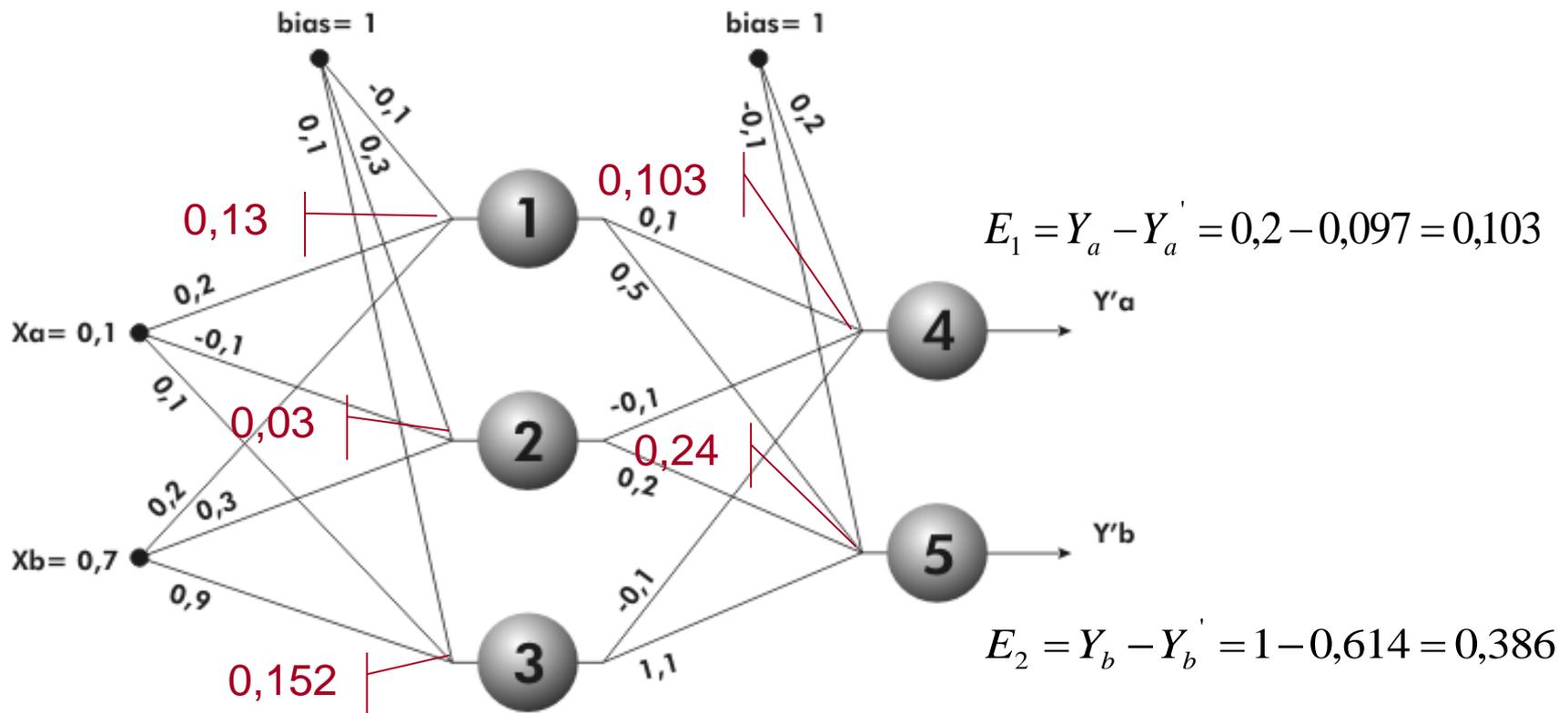
$$U_4 \equiv 1:W_{4i}$$

$$U_5 \equiv 1:W_{5i}$$



BACKPROPAGATION

- valores desejados: $Y_a=0,2$; $Y_b=1$
- Retro-propagação do erro



Processamento de Imagens

Prof. Marcelo Portes de Albuquerque
Prof. Márcio Portes de Albuquerque

Monitores: Fernanda Dutra Moraes (CBPF)
Pedro de Souza Asad (CBPF)