

**Universidade Federal de São Carlos**  
**Centro de Ciências Exatas e de Tecnologia**  
**Departamento de Computação**  
**Programa de Pós-Graduação em Ciência da Computação**

**CLASSIFICAÇÃO DE IMAGENS  
TOMOGRÁFICAS DE CIÊNCIA DOS SOLOS  
UTILIZANDO REDES NEURAIIS E  
COMBINAÇÃO DE CLASSIFICADORES**

Fabricio Aparecido Breve

São Carlos – SP  
Fevereiro/2006

**Ficha catalográfica elaborada pelo DePT da  
Biblioteca Comunitária da UFSCar**

B846ci

Breve, Fabricio Aparecido.

Classificação de imagens tomográficas de ciência dos solos utilizando redes neurais e combinação de classificadores / Fabricio Aparecido Breve. -- São Carlos : UFSCar, 2006.

57 p.

Dissertação (Mestrado) -- Universidade Federal de São Carlos, 2006.

1. Reconhecimento de padrões. 2. Combinação de classificadores. 3. Redes neurais (Computação). 4. Tomografia computadorizada. I. Título.

CDD: 006.4 (20<sup>a</sup>)



# AGRADECIMENTOS

A Deus pela vida e por tudo que não sou merecedor, mas que por Sua graça tenho alcançado;

Aos meus pais, Dirceu e Beatriz, pelo incentivo, apoio e dedicação durante todos esses anos;

Ao Prof. Dr. Nelson Delfino D'Ávilla Mascarenhas, meu orientador, por todo o conhecimento que adquiri com ele e pela confiança depositada em mim durante o desenvolvimento deste trabalho;

Ao Dr. Paulo Estevão Cruvinel, que adquiriu e nos cedeu as imagens tomográficas utilizadas neste trabalho;

Ao Moacir pelas sugestões, dúvidas solucionadas e parceria nos artigos submetidos;

Aos professores Dr. José Hiroki Saito e Dr. Jander Moreira, pelas valiosas correções e observações feitas no Exame de Qualificação;

A todos os professores e funcionários do PPG-CC;

Aos colegas do laboratório GAPIS: Alexandre, Ana, André, Denis, Diego, Gabriel, João Fernando, João Paulo, Luís, Mairum, Marcelo, Michelle, Murilo, Murillo, Yuji, etc.

A CAPES e a FAPESP pelo apoio financeiro durante os dois anos em que este trabalho foi desenvolvido;

A todos que diretamente ou indiretamente colaboraram durante o curso e a execução deste trabalho.

# SUMÁRIO

<b>1. INTRODUÇÃO .....</b>	<b>1</b>
1.1 OBJETIVOS .....	2
1.2 ORGANIZAÇÃO DO TEXTO.....	3
<b>2. TOMOGRAFIA COMPUTADORIZADA .....</b>	<b>4</b>
2.1 PRINCÍPIOS BÁSICOS .....	4
2.2 TOMOGRAFIA DE RAIOS-X E ATENUAÇÃO .....	4
2.3 ALGORITMOS DE RECONSTRUÇÃO .....	5
2.4 IMAGENS MULTIESPECTRAIS .....	6
<b>3. CLASSIFICAÇÃO .....</b>	<b>8</b>
3.1 RECONHECIMENTO DE PADRÕES .....	8
3.2 CLASSIFICADORES E REDES NEURAIS .....	10
3.2.1 Histórico .....	10
3.2.2 O que é uma rede neural.....	11
3.2.3 Perceptron de Múltiplas Camadas .....	13
3.2.4 Redes de Função de Base Radial.....	17
3.2.5 Comparação entre Redes de Função de Base Radial e Perceptrons de Múltiplas Camadas .....	19
3.3 AVALIAÇÃO DE DESEMPENHO DOS CLASSIFICADORES .....	20
3.3.1 Método Hold-out.....	20
3.3.2 Coeficiente Kappa.....	21
<b>4. COMBINAÇÃO DE CLASSIFICADORES.....</b>	<b>23</b>
4.1 CARACTERÍSTICAS .....	23
4.2 COMBINAÇÃO DE CLASSIFICADORES COM VALORES DE SAÍDA CONTÍNUOS .....	24
4.3 COMBINADOR DECISION TEMPLATES .....	25
4.4 COMBINADOR DE DEMPSTER-SHAFER .....	26
4.5 BAGGING .....	27
<b>5. METODOLOGIA.....</b>	<b>29</b>
5.1 AQUISIÇÃO DE IMAGENS.....	29
5.1.1 O Tomógrafo.....	29
5.1.2 Corpo de Prova (Phantom).....	30
5.1.3 Imagens tomográficas obtidas e suas características.....	31
5.2 CLASSIFICAÇÃO DAS IMAGENS .....	33
5.2.1 MATLAB .....	33
5.2.2 PRTtools.....	34
5.2.3 Procedimentos .....	34
<b>6. RESULTADOS .....</b>	<b>39</b>
6.1 PERCEPTRON DE MÚLTIPLAS CAMADAS .....	40
6.2 REDES DE FUNÇÃO DE BASE RADIAL.....	46
<b>7. DISCUSSÃO .....</b>	<b>52</b>
<b>8. CONCLUSÕES.....</b>	<b>54</b>
8.1 TRABALHOS FUTUROS .....	55
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>56</b>

# LISTA DE FIGURAS

<b>FIGURA 3.1</b> – Modelo não-linear de um neurônio .....	12
<b>FIGURA 3.2</b> – Grafo arquitetural de um perceptron de múltiplas camadas com duas camadas ocultas .....	14
<b>FIGURA 3.3</b> – Arquitetura de uma rede neural com funções de base radial .....	19
<b>FIGURA 5.1</b> – Minitomógrafo do Embrapa .....	30
<b>FIGURA 5.2</b> – Diagrama de construção do corpo de prova ( <i>phantom</i> ).....	31
<b>FIGURA 5.3</b> – Curvas do coeficiente de atenuação linear em função da energia para a alumínio, fósforo, cálcio e água .....	32
<b>FIGURA 5.4</b> - Imagens adquiridas com energias de 40, 60, 85 e 662 keV. ....	33
<b>FIGURA 5.5</b> - Janelas de pixels escolhidos como amostras nas imagens com energias de 40, 60, 85 e 662 keV. ....	35
<b>FIGURA 6.1</b> – Variação do Erro Estimado médio para o Perceptron de Múltiplas Camadas.....	40
<b>FIGURA 6.2</b> – Variação do Coeficiente Kappa médio para o Perceptron de Múltiplas Camadas .....	41
<b>FIGURA 6.3</b> – Variação do Desvio Padrão para o Perceptron de Múltiplas Camadas.....	42
<b>FIGURA 6.4</b> – Imagens Temáticas para Perceptron de Múltiplas Camadas com 2 nós na camada oculta (melhores casos).....	43
<b>FIGURA 6.5</b> – Imagens Temáticas para Perceptron de Múltiplas Camadas com 2 nós na camada oculta (piores casos).....	43
<b>FIGURA 6.6</b> – Imagens Temáticas para Perceptron de Múltiplas Camadas com 13 nós na camada oculta (melhores casos).....	44
<b>FIGURA 6.7</b> – Imagens Temáticas para Perceptron de Múltiplas Camadas com 13 nós na camada oculta (piores casos).....	44
<b>FIGURA 6.8</b> – Imagens Temáticas para Perceptron de Múltiplas Camadas com 25 nós na camada oculta (melhores casos).....	45
<b>FIGURA 6.9</b> – Imagens Temáticas para Perceptron de Múltiplas Camadas com 25 nós na camada oculta (piores casos).....	45
<b>FIGURA 6.10</b> – Variação do Erro Estimado médio para Redes de Função de Base Radial.....	46
<b>FIGURA 6.11</b> – Variação do Coeficiente Kappa médio para o Perceptron de Múltiplas Camadas .....	47
<b>FIGURA 6.12</b> – Variação do Desvio Padrão para Redes de Função de Base Radial .....	48
<b>FIGURA 6.13</b> – Imagens Temáticas para Redes de Função de Base Radial com 2 nós na camada oculta (melhores casos).....	49
<b>FIGURA 6.14</b> – Imagens Temáticas para Redes de Função de Base Radial com 2 nós na camada oculta (piores casos).....	49
<b>FIGURA 6.15</b> – Imagens Temáticas para Redes de Função de Base Radial com 13 nós na camada oculta (melhores casos).....	50

<b>FIGURA 6.16</b> – Imagens Temáticas para Redes de Função de Base Radial com 13 nós na camada oculta (piores casos).....	50
<b>FIGURA 6.17</b> – Imagens Temáticas para Redes de Função de Base Radial com 25 nós na camada oculta (melhores casos).....	51
<b>FIGURA 6.18</b> – Imagens Temáticas para Redes de Função de Base Radial com 25 nós na camada oculta (piores casos).....	51

# LISTA DE TABELAS

TABELA I – Coeficientes de atenuação linear em função das energias 40, 60, 85 e 662 keV para a água, alumínio, cálcio e fósforo. ....	31
TABELA II – Cores associadas às classes nas imagens temáticas .....	35
TABELA III – Erro Estimado médio para o Perceptron de Múltiplas Camadas .....	40
TABELA IV – Coeficiente Kappa médio para o Perceptron de Múltiplas Camadas .....	41
TABELA V – Desvio Padrão do Erro Estimado e Coeficiente Kappa para o Perceptron de Múltiplas Camadas .....	42
TABELA VI – Erro Estimado médio para Redes de Função de Base Radial.....	46
TABELA VII – Coeficiente Kappa médio para Redes de Função de Base Radial.....	47
TABELA VIII – Desvio Padrão do Erro Estimado e Coeficiente Kappa para Redes de Função de Base Radial .....	48



# RESUMO

Reconhecimento de Padrões é uma área de âmbito multidisciplinar, com diversas abordagens. Uma delas é a aplicação em imagens de tomografia computadorizada, obtidas mais comumente para fins de diagnóstico médico, mas que também vêm sendo utilizadas nas mais variadas aplicações, inclusive Ciência dos Solos. O objetivo deste trabalho é estudar e discutir o desempenho de classificadores baseados em redes neurais (Perceptron de Múltiplas Camadas e Redes de Função de Base Radial) e da combinação de classificadores (Bagging, Decision Templates e Dempster-Shafer) na identificação de materiais em imagens multiespectrais de Ciência de Solos obtidas através de tomografia computadorizada. Os resultados foram avaliados através da estimativa de erro obtida pelo método *Hold-Out* e do coeficiente Kappa.

# ABSTRACT

Pattern Recognition is a subject being used in a multidisciplinary scope, with different approaches. One of them is its application in computerized tomography images, commonly acquired in order to do medical diagnosis, but they have been used in several other applications as well, including Soil Science. The objective of this work is to study and to discuss the performance of neural network-based classifiers (Multilayer Perceptron and Radial Basis Functions) and classifier combiners (Bagging, Decision Templates and Dempster-Shafer) applied to identify materials in Soil Science multispectral images, acquired using Computerized Tomography. The results were evaluated by error estimation by Hold-Out and the Kappa coefficient.

# 1. INTRODUÇÃO

---

Reconhecimento de Padrões é um processo de descobrimento, de identificação e entendimento de padrões que sejam relevantes à realização de uma tarefa que pode ser baseada em imagens [GONZALEZ; WOODS, 2000]. Uma de suas principais metas é dotar uma máquina de capacidade de classificar objetos (padrões) em categorias (classes) de maneira similar à que os seres humanos fazem. Este é um desafio muito grande, pois muitos destes problemas são resolvidos intuitivamente por um ser humano, sem um método que possa ser facilmente descrito e adaptado para ser usado em uma máquina. Entretanto, em algumas tarefas específicas as máquinas já conseguem substituir humanos com um bom grau de confiabilidade. São tarefas como separar correspondência, fazer diagnósticos médicos, verificações de assinatura, classificar imagens de sensoriamento remoto, previsão do tempo, testes de DNA e reconhecimento de impressões digitais, dentre outras.

As técnicas de Reconhecimento de Padrões podem ser aplicadas às mais diversas áreas e aos mais diversos tipos de imagens e sinais, sendo que a Tomografia Computadorizada é uma das áreas onde o Reconhecimento de Padrões vem sendo amplamente utilizado.

A EMBRAPA<sup>1</sup> Instrumentação Agropecuária da cidade de São Carlos projetou e construiu um minitomógrafo para obtenção de imagens de Ciências de Solos. Utilizando este equipamento, imagens podem ser adquiridas com o uso de diferentes fontes de energia: raios gama provenientes de fontes radioativas como Césio ( $^{137}\text{Cs}$ ) e Amerício ( $^{241}\text{Am}$ ) ou raios-X com múltiplas energias. Foi desenvolvida uma dissertação de mestrado na qual foram aplicadas técnicas de classificação multiespectral com o intuito de identificar materiais em imagens obtidas através de testes não destrutivos com tecnologia do minitomógrafo da EMBRAPA, utilizando raios- $\Gamma$  e raios-X com múltiplas energias [HOMEM,1998]. Dando

---

<sup>1</sup> Empresa Brasileira de Pesquisa Agropecuária

continuidade a este trabalho, outra dissertação de mestrado foi desenvolvida. Nela foram utilizados mais alguns tipos de classificadores, cujas saídas foram combinadas com diferentes tipos de combinadores, mostrando que com a combinação de resultados é possível obter resultados melhores que os fornecidos por cada classificador individualmente [PONTI, 2004]. Com as crescentes pesquisas em técnicas de reconhecimento de padrões baseadas em redes neurais, redes de base radial e sistemas de combinação de classificadores, abre-se espaço para propor e apresentar novos algoritmos para classificação das imagens e para combinação de classificadores.

Dentro deste contexto as Redes Neurais surgem como uma extensão das muitas técnicas convencionais que vêm sendo desenvolvidas por décadas. Em determinados tipos de problema as Redes Neurais apresentam muitas vantagens com relação aos classificadores tradicionais. Neste trabalho serão estudados e aplicados os classificadores Perceptron de Múltiplas Camadas e Redes de Base Radial, ambos desenvolvidos a partir do estudo das redes neurais.

No âmbito de Combinação de Classificadores, Kuncheva [2004] classifica a área como “um adolescente: cheio de energia, entusiasmo, espontaneidade e confusão; passando por rápidas mudanças e obstruindo as tentativas de trazer alguma ordem em sua desordenada caixa de acessórios”. Esta é uma área muito dinâmica e que vem crescendo de forma extremamente rápida. Portanto se torna bastante interessante aplicar alguns de seus métodos para avaliar seus possíveis benefícios. Neste trabalho serão utilizados as técnicas de Bagging e os combinadores Decision Templates e Dempster-Shafer. Para medir os desempenhos utilizaremos o método *Hold-out* para estimar o erro e também o Coeficiente Kappa.

## **1.1 Objetivos**

O objetivo deste trabalho é apresentar métodos para identificação de materiais em imagens tomográficas de solo utilizando técnicas de reconhecimento de padrões baseadas em

redes neurais. Em 1998, Homem [1998] propôs os classificadores de Máxima Verossimilhança, K-médias e ICM (Iterated Conditional Modes). Posteriormente, Ponti [2004] fez um trabalho de continuidade propondo os classificadores Bayesiano Linear, Logístico, Parzen e K-vizinhos mais Próximos; além dos combinadores Votação por Maioria, Máximo, Mínimo, Mediana, Soma e Produto. Pretende-se dar continuidade a estes trabalhos testando agora o desempenho de classificadores baseados em Redes Neurais nas mesmas imagens, técnicas de combinação baseadas em Bagging e os combinadores de Decision Templates e Dempster-Shafer.

Ao classificar amostras de solo e verificar o desempenho de diversos algoritmos de classificação e combinação, esta dissertação deverá contribuir para as pesquisas e trabalhos nas áreas de Ciência dos Solos, Reconhecimento de Padrões e Redes Neurais.

## **1.2 Organização do Texto**

Este trabalho de qualificação está dividido em 8 capítulos, organizados da seguinte forma: O capítulo 1 trata da introdução e do objetivo. No capítulo 2 foram abordados os principais conceitos de Tomografia Computadorizada relevantes ao desenvolvimento do trabalho de dissertação. No capítulo 3 foi feita uma revisão sobre os classificadores que serão utilizados, bem como os métodos de avaliação de desempenho dos mesmos. No capítulo 4 são apresentados os combinadores de classificadores que também serão utilizados. O capítulo 5 relata a metodologia que será usada para desenvolvimento. No capítulo 6 são apresentados os resultados obtidos, que são discutidos no Capítulo 7. Finalmente, o Capítulo 8 mostra as conclusões do trabalho através dos resultados obtidos.

## **2. TOMOGRAFIA COMPUTADORIZADA**

---

Este trabalho utiliza imagens obtidas através de Tomografia Computadorizada. Apresentaremos a seguir os princípios básicos e os principais aspectos desta tecnologia, que justificam o interesse em explorá-la na tarefa de classificação.

### **2.1 Princípios Básicos**

Tomografia é uma maneira de visualizar um objeto através de um imageamento<sup>2</sup> seccional não destrutivo. Esta técnica teve um impacto revolucionário na medicina, mas também vem sendo usada com sucesso em diversas outras áreas.

Basicamente o que a tomografia faz é reconstruir uma imagem a partir de suas projeções, que são as informações derivadas das energias transmitidas, quando o objeto é iluminado a partir de um determinado ângulo.

Do ponto de vista matemático, a solução para o problema de como reconstruir uma função através de sua projeção foi apresentada por Radon em 1917. Porém foi em 1972 que Hounsfield [1973] inventou o primeiro scanner tomográfico computadorizado, pelo qual ganhou um prêmio Nobel em 1979. Sua invenção mostrou que é possível calcular imagens seccionais de alta qualidade, com bastante precisão. Após a invenção de Hounsfield, os grandes avanços nos Raios-X computadorizados se devem às melhorias feitas nos algoritmos de reconstrução e aos avanços nas tecnologias dos sensores [KAK; SLANEY, 1988].

### **2.2 Tomografia de Raios-X e atenuação**

Os dados de projeção são resultados da interação entre a radiação usada e da composição do objeto a ser reconstruído. Um exemplo simples é a atenuação que um feixe de

---

<sup>2</sup> obtenção de imagens

raios X sofre ao atravessar um determinado objeto. Um emissor estará emitindo fótons que viajarão em linha reta através do corpo idealmente, não havendo perda por espalhamento. Porém o feixe é atenuado devido a fótons que são absorvidos pelos átomos do material ou sendo desviados de seu trajeto original.

Na faixa de energia normalmente utilizada para imageamento (de 20 a 150 keV), os mecanismos que causam a atenuação são o efeito fotoelétrico e o efeito Compton. A absorção fotoelétrica consiste em um fóton de raios-X conceder toda a sua energia para um elétron fortemente ligado (em uma das camadas mais internas) de um átomo. O elétron usa parte dessa energia para conseguir se desprender de seu átomo e o restante passa a ser energia cinética do elétron que agora é livre. O espalhamento Compton por outro lado, consiste da interação do fóton de raios-X com um elétron livre ou fracamente ligado (em uma das camadas mais externas do átomo). Como resultado dessa interação, o fóton de raios-x é desviado de sua trajetória original com alguma perda de energia, a qual é transferida para o elétron.

Ambos os efeitos Compton e fotoelétrico são dependentes de energia, ou seja, a probabilidade de um dado fóton sair de sua trajetória, por absorção ou espalhamento, é uma função da energia do próprio fóton [KAK; SLANEY, 1988].

### ***2.3 Algoritmos de Reconstrução***

A partir dos dados de projeção obtidos na tomografia é possível reconstruir a imagem do objeto. Em situações ideais, projeções são o conjunto de medidas dos valores integrais de algum parâmetro do objeto – com a integração sendo feita ao longo de linhas retas através do objeto e sendo referenciadas como integrais de linha. Um exemplo típico seria a atenuação de raios-x conforme eles se propagam pelo objeto. Neste caso o objeto é modelado como uma distribuição bidimensional (ou tridimensional) da constante de atenuação dos raios-x, e a

integral de linha representa a atenuação total sofrida por um feixe de raios-x conforme ele viaja em linha reta pelo objeto.

A chave para fazer a reconstrução é o Teorema de Corte de Fourier, o qual é obtido tomando-se a transformada de Fourier unidimensional de uma projeção paralela, notando que isto equivale a uma fatia da transformada de Fourier bidimensional do objeto original. A partir dos dados de projeção, e tendo um número suficiente de fatias, é possível então estimar o objeto apenas aplicando a transformada inversa de Fourier bidimensional.

Apesar de este modelo conceitual ser simples, implementações práticas requerem uma abordagem diferente. O algoritmo mais usado atualmente em quase todas as aplicações que usam raios em linha reta é o algoritmo de retroprojeção filtrada, que já se mostrou bastante preciso e de rápida implementação e é obtido a partir do Teorema de Corte de Fourier. Este teorema é obtido reescrevendo a transformada inversa de Fourier em coordenadas polares e rearranjando os limites de integração. A implementação no computador destes algoritmos requer que os dados de projeção sejam amostrados e então filtrados, sendo possível aplicar algoritmos de FFT (Transformada Rápida de Fourier).

Informações mais detalhadas podem ser encontradas em Kak e Slaney [1988] e Parker [1990].

## ***2.4 Imagens Multiespectrais***

As imagens tomográficas são obtidas através de um sensor que responde de acordo com a quantidade de fótons que incidem sobre ele, ou seja, os fótons que conseguiram atravessar o objeto sendo mapeados sem ser atenuados ou desviados de seu trajeto. A resposta do sensor depende não apenas da quantidade de fótons incidentes, mas também de sua frequência.

Desta forma é possível utilizar sensores com respostas espectrais diferentes, obtendo-se imagens diferentes, porém obtidas em um mesmo instante. Ao conjunto total de imagens



geradas em um mesmo momento por sensores com diferentes respostas espectrais dá-se o nome de “imagens multiespectrais”, sendo que cada imagem individual passa a ser uma “banda” ou “canal” dessa imagem multiespectral.

Se as imagens individuais forem perfeitamente alinhadas (o que normalmente é obtido na prática), cada ponto será representado não apenas por um valor, mas sim por um vetor de valores (vetor de atributos) com número de elementos igual ao número de sensores utilizados no imageamento, ou seja, um elemento para cada um dos sensores utilizados [GONZALEZ; WOODS, 2000].

## 3. CLASSIFICAÇÃO

---

Classificação de Imagens é o tema principal deste trabalho. Neste capítulo serão apresentados os principais conceitos desta técnica, bem como os classificadores utilizados nesta pesquisa.

### 3.1 Reconhecimento de Padrões

A facilidade que os humanos têm em reconhecer rostos e vozes, entender palavras faladas e escritas, reconhecer objetos pelo tato e alimentos pelo odor, dá uma idéia da complexidade do sistema que realiza essas tarefas de reconhecer padrões.

O Reconhecimento de Padrões (o ato de tomar dados crus e transformá-los em uma ação baseada na ‘categoria’ do padrão) tem sido crucial para a sobrevivência humana e com o passar dos milhares de anos os humanos foram desenvolvendo e evoluindo sistemas neurais e cognitivos sofisticados para estas tarefas [DUDA et. al., 2000].

A capacidade das máquinas reconhecerem padrões ainda está distante da dos seres humanos, entretanto bons resultados já são obtidos em algumas áreas específicas como reconhecimento de caracteres impressos, impressões digitais, reconhecimento de imagens de satélites, reconhecimento de voz, etc. Entretanto a maioria dessas aplicações é bem específica e com pouca ou nenhuma capacidade de generalização [GONZALEZ; WOODS, 2000].

Em diversos casos a maneira como as aplicações resolvem o problema de reconhecimento de padrões é diretamente influenciada pelo conhecimento de como estes problemas são resolvidos na natureza, tanto nos algoritmos empregados quanto no hardware.

O principal objetivo do reconhecimento de padrões é classificar os objetos de interesse dentro de categorias (ou classes). Estes objetos de interesse são chamados de “padrões”. Por exemplo: em um sistema para reconhecer algarismos numéricos arábicos escritos à mão, teríamos dez classes, uma para cada algarismo (de 0 a 9), e cada número seria um padrão a ser classificado como pertencente a uma das dez classes.

Para que seja possível separar objetos em diferentes classes é necessário encontrar uma ou mais características que possam ser usadas para diferenciá-los. Exemplificando: em um sistema para reconhecer faces, as características poderiam ser: formato da face, cor dos olhos, distâncias entre os olhos, formato da boca, etc. Estas características recebem o nome de *atributos*. Ao escolher atributos deve-se ter o cuidado de escolher as características que mais se diferenciam entre os objetos de uma classe e outra.

Uma vez escolhidos os atributos é hora de decidir qual modelo de classificação (ou classificador) será utilizado. Os classificadores podem ser divididos em dois grandes grupos: o dos com aprendizado supervisionado, e o dos com aprendizado não supervisionado. Neste trabalho serão utilizados apenas classificadores com aprendizado supervisionado. O aprendizado supervisionado consiste em utilizar um determinado conjunto de padrões já devidamente rotulados para treinar o classificador, para que posteriormente ele possa rotular padrões desconhecidos baseando-se na similaridade destes com os padrões apresentados durante o treinamento. Pode-se entender, então, que o treinamento do classificador consiste na divisão do espaço de atributos em diversas regiões, de modo que os objetos de uma determinada classe caiam sempre (ou o mais freqüentemente possível) dentro da mesma região.

Uma vez treinado, o classificador pode ser testado quanto à sua eficiência para classificar padrões desconhecidos por ele. Assim, pode-se tomar um conjunto de padrões já rotulados (e que preferivelmente não devem ser os mesmos utilizados no treinamento) e apresentá-los ao classificador sem o rótulo, para que o classificador possa rotulá-los. Comparando então os rótulos que já existiam com os fornecidos pelo classificador é possível estimar qual será o desempenho do classificador, ou seja, qual a taxa de acertos e erros que se espera dele.

Este é um resumo geral das principais tarefas envolvidas no reconhecimento de padrões, mas ainda há muitas outras que não foram abordadas aqui, mas que podem ser encontradas na literatura: Duda et. al. [2000], Fukunaga [1990], dentre outros.

### **3.2 Classificadores e Redes Neurais**

A maneira mais comum e natural de formular soluções para os problemas de Reconhecimento de Padrões é a Estatística, a qual reconhece a natureza probabilística tanto da informação que se procura para processar, quanto da forma pela qual se deve expressar os resultados. [BISHOP, 1995]

Os classificadores estatísticos já estão bem estabelecidos e vêm sendo usados há bastante tempo. Nos trabalhos de Homem et. al. [2000] e Ponti Jr e Mascarenhas [2004] foram apresentados e utilizados diversos classificadores estatísticos na tarefa de classificação de imagens tomográficas. Dando continuidade a estes trabalhos, serão apresentados classificadores baseados em redes neurais, como uma forma de estender as técnicas estatísticas convencionais e comparar os resultados obtidos.

O trabalho em redes neurais artificiais, usualmente denominadas apenas *redes neurais*, vem sendo motivado pelos estudos da maneira como o cérebro humano processa informações, a qual é totalmente diferente da maneira utilizada pelos computadores convencionais. O cérebro é um “computador” altamente complexo, não-linear e paralelo. Ele tem capacidade para organizar seus neurônios de forma a realizar determinados processamentos muito mais rapidamente que qualquer computador digital existente hoje [HAYKIN, 2001].

#### **3.2.1 Histórico**

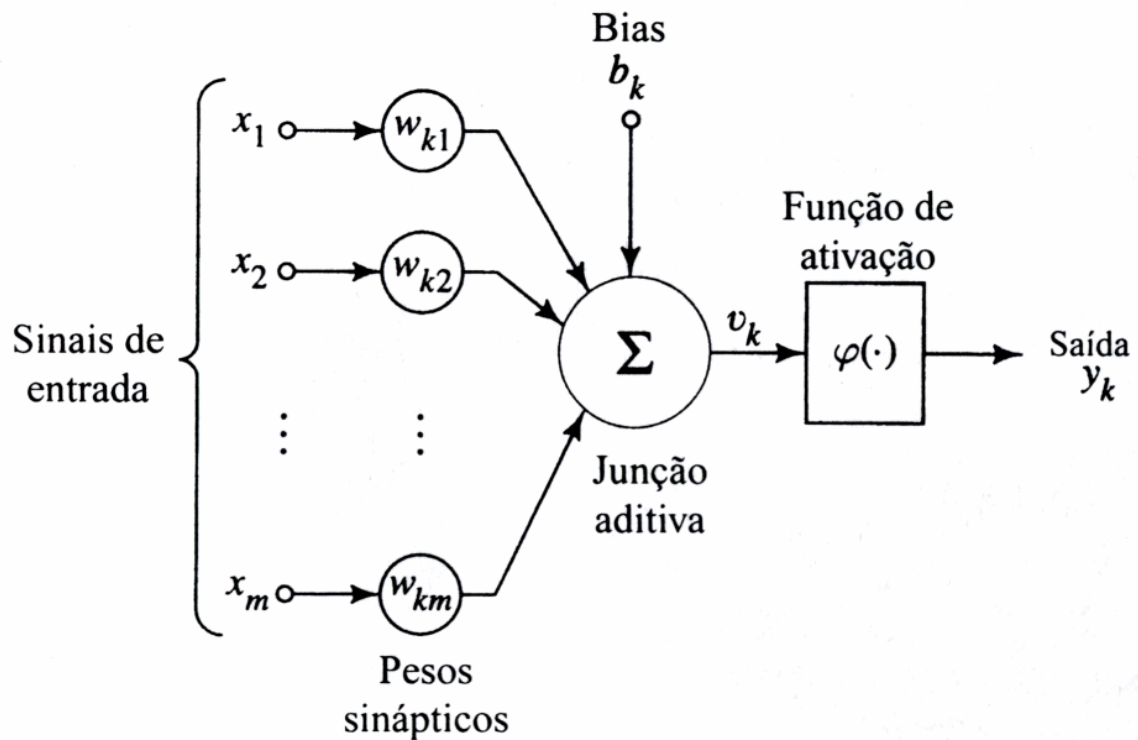
Os estudos com redes neurais começaram com os trabalhos de McCulloch e Pitts em 1942, que descreveram um cálculo lógico das redes neurais que unificava estudos de neurofisiologia e de lógica matemática. Eles demonstraram que era possível montar uma rede

de neurônios simples que seguiam uma lei “tudo ou nada” que pudesse realizar, em princípio, a computação de qualquer função computável. Em 1958, Rosenblatt criou o Perceptron, um método inovador de aprendizagem supervisionada que impulsionou os estudos sobre redes neurais por algum tempo. Mas em 1969 surgiu o livro de Minsky e Papert [1969], que demonstraram que existiam limites para aquilo que um perceptron de camada única poderia calcular e isto acabou com o entusiasmo que havia se criado em torno das redes neurais. Foi apenas na década de 1980 que foram solucionados os problemas do Perceptron, com o surgimento do algoritmo de aprendizado por retropropagação, proposto por Werbos, reinventado várias vezes e popularizado por Rumelhart [JAIN et. al., 1996; HAYKIN, 2001].

### 3.2.2 O que é uma rede neural

Uma rede neural é uma máquina projetada para modelar a maneira como o cérebro realiza uma determinada tarefa; é implementada por componentes eletrônicos ou simulada por programação em um computador digital. Para alcançar um bom resultado as redes neurais utilizam diversas células computacionais simples (denominadas *neurônios*) e altamente interligadas. Seu objetivo é armazenar conhecimento experimental e torná-lo disponível para uso. O conhecimento é adquirido pela rede através de um processo de aprendizagem. O conhecimento é armazenado nos neurônios em unidades denominadas *pesos sinápticos*. O processo de aprendizagem consiste em modificar os pesos sinápticos da rede de uma forma ordenada para alcançar o objetivo desejado.

O diagrama da Figura 3.1 mostra o *modelo* de um neurônio, que apresenta os seguintes elementos básicos [HAYKIN, 2001]:



**FIGURA 3.1** – Modelo não-linear de um neurônio  
Fonte: Haykin [2001]

1. Um conjunto de *sinapses* ou *elos de conexão*, cada um sendo constituído por um *peso*. Cada sinal  $x_j$  na entrada da sinapse  $j$  conectada ao neurônio  $k$  é multiplicado pelo peso sináptico  $w_{kj}$ .
2. Um *somador* para somar os sinais de entrada, que são ponderados pelas respectivas sinapses do neurônio, e adicionar um *bias* quando houver.
3. Uma *função de ativação* que restringe o intervalo de saída do neurônio a um valor finito, tipicamente  $[0, 1]$  ou alternativamente  $[-1, 1]$ .

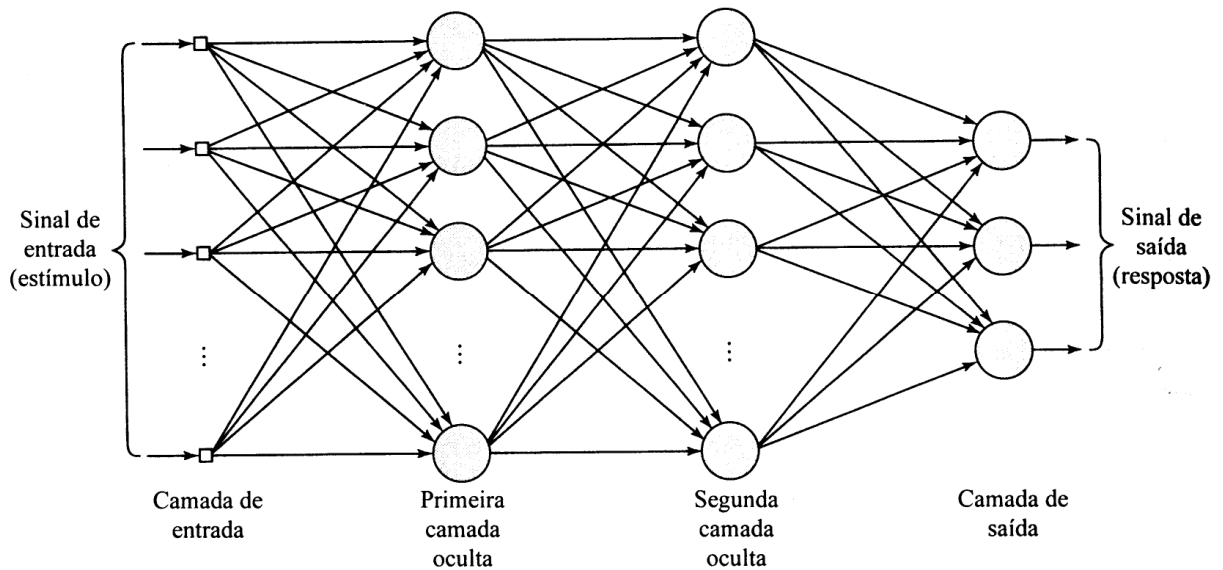
A combinação dos diversos neurônios artificiais forma a chamada *rede neural*. A saída de cada neurônio pode ser conectada à(s) entrada(s) de outro(s) neurônio(s) formando assim uma rede. Cada neurônio pode então ser definido como um “nó” dessa rede. A formação da rede pode assumir muitas estruturas diferentes

Dentre as diversas famílias de redes neurais existentes, duas são de particular interesse na tarefa de reconhecimento de padrões: o *Perceptron de Múltiplas Camadas* e as *Redes de Função de Base Radial*, que são os dois tipos de redes neurais que serão utilizados nesta dissertação e serão descritas nas seções a seguir.

### **3.2.3 Perceptron de Múltiplas Camadas**

O *Perceptron de Múltiplas Camadas* é um tipo de rede que consiste de um conjunto de unidades sensoriais, que constituem a *camada de entrada*, uma ou mais *camadas ocultas* e uma *camada de saída* de nós computacionais. O sinal que é colocado nas entradas se propaga para frente camada por camada até chegar à camada de saída.

Este tipo de rede vem sendo utilizado com sucesso para resolver problemas difíceis através de seu treinamento utilizando o *algoritmo de retropropagação de erro*, que consiste basicamente em dois passos: um passo para frente em que o sinal aplicado à entrada vai se propagando pelos nós computacionais da rede até chegar aos nós de saída; e um passo para trás, no qual todos os pesos sinápticos são ajustados de acordo com uma regra de correção de erro. A figura 3.2 mostra o grafo arquitetural de um perceptron com duas camadas ocultas e uma camada de saída. Esta é uma rede totalmente conectada, ou seja, todos os neurônios têm ligação com todos os nós da camada anterior. O fluxo da rede progride para a frente de camada em camada até a saída.



**FIGURA 3.2** – Grafo arquitetural de um perceptron de múltiplas camadas com duas camadas ocultas  
Fonte: Haykin [2001]

As características mais comuns nos perceptrons de múltiplas camadas são [Kuncheva, 2004]:

- A função de ativação na camada de entrada é uma função identidade
- Não há conexões laterais entre nós da mesma camada
- Camadas não adjacentes não são conectadas diretamente
- Todos os nós das camadas ocultas utilizam a mesma função de ativação

Há dois tipos de sinais que trafegam pela rede: os *sinais funcionais* e os *sinais de erro*. Os sinais funcionais são os que são inseridos nos terminais de entrada da rede. Recebem esse nome pois se presume que eles terão uma função útil na saída da rede. Em cada neurônio o sinal funcional que passa é calculado como uma função de suas entradas e pesos associados. Os sinais de erro se originam nos neurônios de saída da rede e se propagam para trás (camada por camada), e recebem este nome porque sua computação em cada neurônio utiliza uma função dependente do erro.



Todos os neurônios ocultos e da camada de saída são projetados para realizar dois cálculos: o dos sinais funcionais que aparecem na saída da rede e o dos sinais de erro necessários para a retropropagação através da rede. [HAYKIN, 2001]

O algoritmo de treinamento com retropropagação é descrito por Kuncheva [2004] da seguinte forma:

1. Escolher uma estrutura de Perceptron Multicamadas (o número de camadas ocultas, o número de nós em cada camada e as funções de ativação);
2. Iniciar o processo de treinamento: escolher pequenos valores aleatórios para todos os pesos (incluindo os *bias*) da rede neural. Escolher a taxa de aprendizado  $\eta > 0$ , o número máximo de épocas  $T$  e o erro desejado  $\varepsilon > 0$ ;
3. Ajustar  $E = \infty$ , o contador de épocas  $t = 1$  e o contador de objetos  $j = 1$ ;
4. Enquanto ( $E > \varepsilon$  e  $t \leq T$ ) faça
  - (a) Submeter  $\mathbf{z}_j$  como o próximo exemplo de treinamento, sendo  $\mathbf{z}$  o conjunto de amostras de treinamento
  - (b) Calcular a saída de cada nó da rede neural com os pesos atuais (propagação adiante)
  - (c) Calcular o erro em cada nó da camada de saída
  - (d) Calcular recursivamente todos os erros nos nós das camadas ocultas utilizando a retropropagação
  - (e) Atualizar os pesos de cada nó de saída e de cada nó oculto
  - (f) Calcular  $E$  usando os pesos atuais
  - (g) Se  $j = N$  (um passo completo através de  $\mathbf{Z}$  (época) é completado), então ajuste  $t = t + 1$  e  $j = 0$ . Senão, ajuste  $j = j + 1$
5. Fim % (Enquanto)

Os perceptrons de múltiplas camadas conseguem contornar a principal limitação do perceptron de única camada, que só consegue gerar bons classificadores quando as classes são linearmente separáveis. No perceptron de múltiplas camadas com uma única camada de neurônios ocultos contendo um número suficiente de neurônios é possível aproximar praticamente qualquer função com uma determinada precisão. [KUNCHEVA, 2004]

O número de unidades de entrada é igual ao número de características escolhidas para serem utilizadas com o classificador. E o número de neurônios na camada de saída é freqüentemente igual à quantidade de padrões diferenciáveis no problema, sendo que cada padrão corresponde a um neurônio da camada de saída e rotula-se a amostra de acordo com o neurônio/padrão que apresentar a maior saída. A única exceção seria o caso de apenas duas classes, onde um único neurônio de saída pode ser utilizado e se atribui a amostra a uma ou outra classe dependendo da saída do neurônio ser maior ou menor que um determinado limiar (tipicamente 0).

O número de camadas ocultas e o número de neurônios em cada camada não são limitados e normalmente são determinados de maneira experimental. Haykin [2001] sugere o critério de utilizar o menor número de neurônios ocultos que produz um desempenho próximo ao do classificador bayesiano. Duda et. al [2000] afirmam que o número de neurônios ocultos depende dos padrões a serem separados; se eles forem bem separados ou linearmente separáveis então poucas unidades são necessárias. Por outro lado se os padrões são formados por densidades complicadas, então mais unidades ocultas são necessárias e portanto não há nenhum método infalível de ajustar o número de neurônios ocultos antes do treinamento.

Os neurônios dos perceptrons de única e múltiplas camadas diferem por suas funções de ativação. Os neurônios do perceptron de única camada se utilizam de funções de ativação de limiar “abrupto”, enquanto que os do perceptron de múltiplas camadas utilizam-se tipicamente de funções logísticas sigmóides, que proporcionam uma função de ativação com

limiar mais suave. As saídas desses neurônios tipicamente ficam no intervalo [0,1] ajustados por uma função de ativação sigmoideal, dada por Bishop [1995]:

$$g(a) \equiv \frac{1}{1 + \exp(-a)}. \quad (3.1)$$

### 3.2.4 Redes de Função de Base Radial

Diferentemente das redes neurais do tipo Perceptron (que utiliza uma técnica recursiva conhecida na estatística como *aproximação estocástica*), as *Redes de Função de Base Radial* utilizam uma abordagem que vê a rede neural como um *problema de ajuste de curva* em um espaço de alta dimensionalidade. Neste contexto, aprender equivale a encontrar uma superfície num espaço multidimensional que forneça o melhor ajuste para os dados de treinamento do ponto de vista estatístico.

A construção de uma rede de função de base radial envolve três camadas com papéis totalmente diferentes. A primeira camada é constituída de nós sensoriais que recebem os dados do ambiente. A segunda camada, que é a única camada oculta da rede, faz uma transformação não-linear do espaço de entrada para um espaço oculto, normalmente de alta dimensionalidade. Finalmente a camada de saída é linear e fornece a resposta da rede ao sinal de entrada [HAYKIN, 2001].

Estes métodos têm origem nas técnicas para fazer interpolação exata de um conjunto de dados em um espaço multidimensional. Para isto é necessário que cada vetor de entrada seja mapeado para um vetor alvo correspondente. [BISHOP, 1995]

Basicamente, quando uma rede de funções de base radial é empregada para resolver uma tarefa complexa de classificação de padrões, transforma o problema em um espaço de alta dimensionalidade, de uma forma não-linear. A justificativa para este processo é encontrada no *teorema de Cover sobre separabilidade de padrões*, que pode ser formulado da seguinte forma [COVER, 1965 apud HAYKIN, 2001]:

“Um problema complexo de classificação de padrões disposto não linearmente em um espaço de alta dimensão tem maior probabilidade de ser linearmente separável do que em um espaço de baixa dimensionalidade”. E uma vez que tenhamos padrões linearmente separáveis, basta utilizar um perceptron simples para resolver o problema de classificação.

A função de interpolação exata normalmente é bastante oscilatória quando gerada a partir de dados ruidosos, portanto não é desejável. Assim uma série de modificações foram introduzidas chegando à seguinte fórmula para o mapeamento de redes neurais baseadas em funções de base radial [BISHOP, 1995]:

$$y_k(\mathbf{x}) = \sum_{j=1}^M w_{kj} \phi_j(\mathbf{x}) + w_{k0}, \quad (3.2)$$

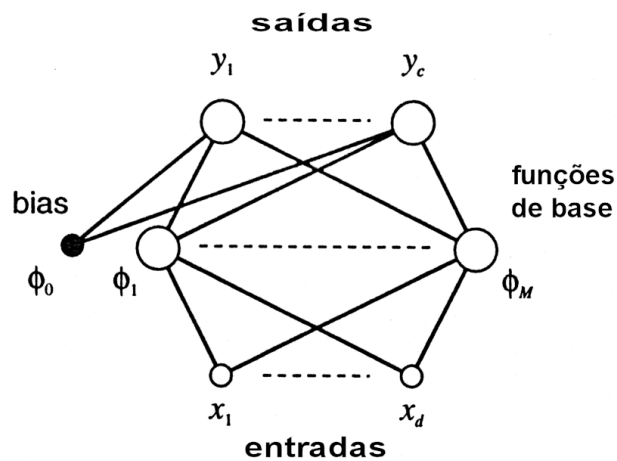
onde  $w_k$  corresponde ao vetor de pesos do neurônio,  $w_{k0}$  ao *bias* e  $\phi_j$  é uma função de ativação que normalmente é gaussiana.

Se desejado, os *bias*  $w_{k0}$  podem ser absorvidos nos somatórios incluindo-se uma função de base extra, cuja ativação é ajustada para 1. Para o caso de funções de base Gaussianas temos:

$$\phi_j(\mathbf{x}) = \exp\left(-\frac{\|\mathbf{x} - \boldsymbol{\mu}_j\|^2}{2\sigma_j^2}\right), \quad (3.3)$$

onde  $\mathbf{x}$  é o vetor de entrada  $d$ -dimensional com os elementos  $x_i$ , e  $\boldsymbol{\mu}_j$  é o vetor que determina o centro das funções de base  $\phi_j$  e tem os elementos  $\mu_{ji}$ .

Esta função de mapeamento pode ser representada como uma rede neural conforme mostrado na figura 3.3. Cada função de base funciona como uma unidade oculta. As linhas conectando as funções de base  $\phi_j$  às saídas representam os elementos  $\mu_{ji}$  correspondentes do vetor  $\boldsymbol{\mu}_j$ . Os pesos  $w_{kj}$  são mostrados como linhas partindo das funções de base para as unidades de saída, e os *bias* são mostrados como pesos de uma “função de base” extra  $\phi_0$  cujo peso está fixado em 1.



**FIGURA 3.3** – Arquitetura de uma rede neural com funções de base radial  
Fonte: [BISHOP, 1995]

### 3.2.5 Comparação entre Redes de Função de Base Radial e Perceptrons de Múltiplas Camadas

As Redes de Função de Base Radial (RBF) e os Perceptrons de Múltiplas Camadas (MLP) são exemplos de redes em camadas alimentadas adiante e não-lineares. Ambos são aproximadores universais. Portanto, pode-se constatar que sempre existe uma rede RBF capaz de imitar precisamente uma rede MLP específica e vice-versa. Porém, existem algumas diferenças entre estes dois tipos de rede que merecem destaque [HAYKIN, 2001]:

1. Uma rede RBF (em sua forma básica) tem apenas uma única camada oculta, enquanto que um MLP pode ter uma ou mais camadas ocultas.
2. Os nós computacionais de um MLP tipicamente têm todos o mesmo modelo neuronal, estejam na(s) camada(s) oculta(s) ou na camada de saída. Por outro lado os nós da camada oculta de uma RBF são bem diferentes dos nós da camada de saída.

3. Nas RBF a camada oculta é não-linear e a camada de saída é linear. Nos MLP tanto as camadas ocultas quanto as camadas de saída são normalmente não lineares.
4. A função de ativação de cada unidade oculta em uma rede RBF calcula a *norma euclidiana* entre o vetor de entrada e o centro daquela unidade. Nos MLP as unidades ocultas calculam o *produto interno* do vetor de entrada pelo vetor de peso sináptico daquela unidade.
5. Os MLPs constroem aproximações *globais* de um mapeamento de entrada-saída não-linear, enquanto que as RBF utilizam-se de não-linearidades localizadas com decaimento exponencial para construir aproximações *locais* para mapeamentos de entrada-saída não-lineares.

O MLP tipicamente requer um número menor de parâmetros que a rede RBF para o mesmo grau de precisão. As características lineares da camada de saída da rede RBF tornam esta rede mais relacionada com o Perceptron de camada única do que com o perceptron de múltiplas camadas. Porém, a RBF difere do perceptron pela capacidade de implementar transformações não-lineares arbitrárias do espaço de entrada.

### **3.3 Avaliação de Desempenho dos Classificadores**

Através de métodos de avaliação podemos obter números que indicam qual foi o desempenho dos classificadores que treinamos. Dentre os diversos métodos existentes apresentaremos aqui dois deles: *Hold-out* e *Coefficiente Kappa*.

#### **3.3.1 Método Hold-out**

Neste método, tradicionalmente se separa o conjunto de dados disponíveis (amostras pré-rotuladas) em duas metades. Usa-se uma para treinar o classificador e a outra para testá-lo, obtendo-se a taxa de erro. Separações em outras proporções também são utilizadas.

Também é possível inverter os conjuntos, obter outra taxa de erro e então tirar uma média dos dois. [KUNCHEVA, 2004]

O método *hold-out* é pessimista porque usa apenas uma porção dos dados para o treinamento. Quanto mais instâncias são deixadas para o conjunto de teste, maior é o desvio da estimativa. Entretanto, poucas instâncias de teste significam que o intervalo de precisão será maior. [KOHAVI, 1995]

### 3.3.2 Coeficiente Kappa

O *Coeficiente Kappa* foi criado por Cohen em 1960 e pode ser utilizado para medir o grau de concordância entre dois classificadores. No caso específico da avaliação de desempenho de um classificador comparam-se os rótulos das amostras de teste com as saídas do classificador.

Para calcular o Coeficiente Kappa podemos utilizar a seguinte fórmula [CARLETTA, 1996]:

$$K = \frac{P(A) - P(E)}{1 - P(E)}, \quad (3.4)$$

onde  $P(A)$  é a proporção de vezes em que o classificador acerta e  $P(E)$  é a proporção de vezes que se espera que um classificador acertasse se as classificações fossem feitas ao acaso. Por exemplo, se temos quatro classes equiprováveis o valor de  $P(E)$  será 0,25. Os valores de  $K$  podem variar entre -1 e 1. Quando não há concordância alguma além da esperada com a classificação sendo feita ao acaso  $K$  será igual a zero. E quando há uma concordância total entre os rótulos pré-estabelecidos e a saída do classificador a saída será um. Valores menores que zero indicam concordância menor do que a esperada com a classificação sendo feita ao acaso.

A interpretação do valor de  $K$  varia bastante em função do problema e do grau de precisão que se quer obter. Em geral espera-se que os classificadores apresentem valores de  $K$  maiores que 0,8 para que seja possível obter uma boa classificação.



## **4. COMBINAÇÃO DE CLASSIFICADORES**

---

A combinação de classificadores é uma forma de conseguir uma classificação mais precisa, ao custo de ter um aumento na complexidade. É dito que, ao combinar classificadores, as deficiências de um determinado classificador são compensadas pelas qualidades de outro [PRAMPERO, 1998]. Neste capítulo apresentaremos as principais características do combinador de Dempster-Shafer e a técnica de Bagging.

### **4.1 Características**

Algumas das principais motivações para se utilizar múltiplos classificadores são apresentadas por Jain et. al. [2000]:

1. Um projetista pode ter acesso a diversos classificadores, cada um deles construído em um contexto diferente, mas com o mesmo objetivo. Por exemplo: reconhecimento de pessoas por suas faces, voz, impressões digitais, etc.
2. Algumas vezes mais de um conjunto de treinamento está disponível, às vezes coletados em momentos diferentes, às vezes até com atributos diferentes.
3. Diferentes classificadores treinados nas mesmas amostras podem se diferenciar não apenas em seu desempenho global, mas também podem ter regiões do hiperespaço em que atuem melhor ou pior que os outros.
4. Alguns classificadores, como os baseados em redes neurais, mostram diferentes resultados em diferentes inicializações devido à característica aleatória de seu treinamento. Em vez de selecionar a melhor rede e descartar as outras, pode-se combiná-las.

Resumidamente: podem-se ter diferentes atributos, diferentes conjuntos de treinamento, diferentes métodos de classificação, todos resultando em um conjunto de

classificadores cujas saídas podem ser combinadas com a esperança de melhorar a confiabilidade da classificação.

Os tipos de informações que um classificador pode fornecer podem ser dispostas em 3 grupos [XU et. al, 1992]:

1. *Nível abstrato*: o classificador fornece apenas uma única saída que é o rótulo da amostra que foi submetida à classificação
2. *Nível de ranking*: o classificador retorna todos os rótulos possíveis em uma fila com o rótulo mais acima sendo a primeira opção
3. *Nível de medição*: para cada opção de rótulo é atribuído um valor que corresponde à probabilidade daquele ser o rótulo correto.

O nível 3 é o que fornece o maior número de informações sobre a classificação e o nível 1 é o que fornece menos informações. Ambos os classificadores baseados em redes neurais utilizados neste trabalho são capazes de fornecer informações no nível 3 e, portanto, foram escolhidos combinadores capazes de tirar proveito dessas informações. Neste trabalho será explorada a combinação de redes neurais com diferentes iniciações

## **4.2 Combinação de Classificadores com Valores de Saída Contínuos**

O grau de suporte para uma dada entrada  $x$  pode ser interpretado de diferentes maneiras, sendo que as mais comuns são conhecidas como *confiança* nos rótulos sugeridos e *estimação de probabilidades a posteriori* para as classes. [KUNCHEVA, 2004]

Seja  $x \in \mathfrak{R}^n$  o vetor de características e  $\Omega = \{\omega_1, \omega_2, \dots, \omega_c\}$  o conjunto de rótulos das classes. Cada classificador  $D_i$  do conjunto  $D = \{D_1, \dots, D_L\}$  fornece  $c$  graus de suporte. Sem perda de generalidade pode-se assumir que todos os  $c$  graus estão no intervalo  $[0, 1]$ , isto é,  $D_i : \mathfrak{R}^n \rightarrow [0,1]^c$ . A notação  $d_{i,j}(x)$  representa o suporte que o classificador  $D_i$  dá para a

hipótese de que  $x$  vem da classe  $\omega_j$ . As  $L$  saídas dos classificadores para uma determinada entrada  $x$  pode ser organizado em um *perfil de decisão* ( $DP(x)$ ) como a matriz:

$$DP(x) = \begin{bmatrix} d_{1,1}(x) & \cdots & d_{1,j}(x) & \cdots & d_{1,c}(x) \\ \vdots & & \vdots & & \vdots \\ d_{i,1}(x) & \cdots & d_{i,j}(x) & \cdots & d_{i,c}(x) \\ \vdots & & \vdots & & \vdots \\ d_{L,1}(x) & \cdots & d_{L,j}(x) & \cdots & d_{L,c}(x) \end{bmatrix}. \quad (4.1)$$

Vários métodos de combinação utilizam  $DP(x)$  para encontrar o suporte geral para cada classe e rotular a entrada  $x$  de acordo com a classe de maior suporte. Duas abordagens podem ser usadas. Na primeira abordagem podemos utilizar o fato de que os valores na coluna  $j$  são os valores individuais de suporte para a classe  $\omega_j$  e derivar um valor de suporte geral para esta classe. Na segunda abordagem podemos ignorar o contexto de  $DP(x)$  e tratar os valores  $d_{i,j}(x)$  como atributos em um novo espaço de atributos, chamado de *espaço de atributos intermediário*. A decisão final é feita por outro classificador que toma o espaço intermediário de atributos como entrada e dá como saída um rótulo de classe.

### 4.3 Combinador Decision Templates

Neste método os combinadores derivam  $\mu_j(x)$  usando todos os  $L \times c$  graus de suporte em  $DP(x)$ . Cada vetor no espaço de atributos intermediário é uma versão expandida das  $DP(x)$  obtidas concatenando-se suas  $L$  linhas.

São criados *padrões de decisão* (DT) que armazenam o perfil de decisão mais típico para cada classe  $\omega_j$ . Os perfis de decisão  $DP(x)$  são então comparados com os DT. O rótulo de  $x$  é dado então a partir do DT mais similar.

O treinamento de um padrão de decisão é feito calculando-se a média dos perfis de decisão  $DP(z_k)$  de todos os membros de  $\omega_j$  do conjunto de dados  $Z$  [KUNCHEVA, 2004]:

$$DT_j = \frac{1}{N_j} \sum_{\substack{z_k \in \omega_j \\ z_k \in Z}} DP(z_k), \quad (4.2)$$

onde  $N_j$  é o número de elementos de  $Z$  vindos de  $\omega_j$ .

Após o treinamento, dada uma entrada, é possível construir seu perfil de decisão  $DP(x)$  e calcular a similaridade  $S$  entre  $DP(x)$  e cada  $DT_j$ :

$$u_j(x) = S(DP(x), DT_j) \quad j = 1, \dots, c. \quad (4.3)$$

A similaridade  $S$  pode ser a distância Euclidiana ou qualquer outra medida de distância. A classe cujo padrão de decisão ( $DT$ ) for mais próximo ao perfil de decisão  $DP(x)$  irá dar rótulo à amostra.

#### **4.4 Combinador de Dempster-Shafer**

O método de Dempster-Shafer é baseado na teoria da evidência, introduzida por Glenn Shafer como uma maneira de representar conhecimento cognitivo. Neste formalismo a melhor representação de uma possibilidade é uma função de crença em vez de uma distribuição Bayesiana. Valores de probabilidade são atribuídos a conjuntos de possibilidades em vez de eventos únicos: seu apelo está no fato de que codificam evidências em vez de proposições. É fornecido um método simples de combinar evidências vindas de diferentes fontes (regra de Dempster) sem necessidade de qualquer distribuição a priori. Neste trabalho este método será utilizado para combinar itens de evidência que vêm das diferentes redes neurais com o objetivo de melhorar a classificação. [AHMADZADEH, 2000]

O treinamento do combinador de Dempster-Shafer é igual ao do Decision Templates. Porém, em vez de calcular a similaridade entre o padrão de decisão e o perfil de decisão, são calculadas a proximidade entre o modelo de decisão e a saída de cada classificador. Estas são usadas para calcular o grau de crença para cada classe. Finalmente os graus de suporte para

cada classe são calculados a partir dos graus de crença. Estes passos estão descritos abaixo:

[KUNCHEVA, 2004]:

1. Seja  $DT_j^i$  a  $i$ ésima linha do padrão de decisão  $DT_j$  e  $D_i(x)$  a saída de  $D_i$ , ou seja,  $D_i(x) = [d_{i,1}(x), \dots, d_{i,c}(x)]^T$ : a  $i$ ésima linha do perfil de decisão  $DP(x)$ . Calculamos a “proximidade”  $\phi$  entre  $DT_j^i$  e a saída do classificador  $D_i$  para a entrada  $x$ :

$$\phi_{j,i}(x) = \frac{\left(1 + \|DT_j^i - D_i(x)\|^2\right)^{-1}}{\sum_{k=1}^c \left(1 + \|DT_k^i - D_i(x)\|^2\right)^{-1}}, \quad (4.4)$$

onde  $\| \cdot \|$  é qualquer norma de matriz. Por exemplo, pode-se usar a distância euclidiana entre os dois vetores. Portanto para cada padrão de decisão existem  $L$  proximidades.

2. Usando a equação anterior calculamos todas as classes,  $j = 1, \dots, c$ ; e para cada classificador,  $i = 1, \dots, L$ , os seguintes graus de crença

$$b_j(D_i(x)) = \frac{\phi_{j,i}(x) \prod_{k \neq j} (1 - \phi_{k,i}(x))}{1 - \phi_{j,i}(x) \left[1 - \prod_{k \neq j} (1 - \phi_{k,i}(x))\right]}. \quad (4.5)$$

3. Os graus de suporte finais são

$$\mu_j(x) = K \prod_{i=1}^L b_j(D_i(x)) \quad j = 1, \dots, c. \quad (4.6)$$

onde  $K$  é uma constante de normalização.

## 4.5 Bagging

O termo Bagging foi criado por Breiman [1996] e é um acrônimo para *Bootstrap AGGregatING*. O método consiste em construir réplicas do tipo *bootstrap* do conjunto de treinamento e treiná-las individualmente. A saída dos diversos classificadores é então

combinada usando votação, ou seja, a saída apontada pela maioria dos classificadores é a escolhida.

A partir de um conjunto de treinamento pré-rotulado são gerados aleatoriamente diversos outros conjuntos utilizando substituição. Para tirar proveito deste método é necessário que o classificador base seja instável, ou seja, um classificador em que pequenas mudanças no conjunto de treinamento levam a grandes mudanças na saída do classificador. Se esta regra não for obedecida o que se obtém é somente um conjunto de classificadores quase idênticos, que dificilmente irão melhorar a performance do classificador. Exemplos de classificadores instáveis são as árvores de decisão e as redes neurais. Por outro lado o classificador de k-vizinhos mais próximos é um exemplo de classificador estável.

A evidência, teórica e experimental, conforme mostrado por Breiman [1996], é de que o Bagging funciona bem, fazendo com que classificadores bons, mas instáveis, possam ser otimizados. Por outro lado, ele pode degradar o desempenho dos classificadores estáveis.

O algoritmo do Bagging é paralelo tanto no treinamento quanto na operação, cada conjunto de treinamento pode ser treinado e mesmo executado em diferentes processadores se necessário. Portanto sua utilização é bastante indicada em sistemas com múltiplos processadores e *clusters*.

Neste trabalho os classificadores utilizados são baseados em redes neurais, portanto espera-se que o uso do método Bagging possa melhorar significativamente o desempenho dos classificadores obtidos.

## **5. METODOLOGIA**

---

Este trabalho propõe métodos para identificação de materiais em imagens multiespectrais ruidosas de tomografia de transmissão utilizando múltiplas energias. A proposta consiste em explorar técnicas de classificação de imagem e combinação de classificadores nas imagens tomográficas adquiridas de um corpo de estudo através de múltiplas energias. Através dos experimentos serão feitas comparações e corroborações teóricas sobre as técnicas utilizadas.

### **5.1 Aquisição de Imagens**

Nesta seção mostraremos como foram obtidas as imagens tomográficas, o corpo de prova e as características das imagens obtidas.

#### **5.1.1 O Tomógrafo**

O minitomógrafo utilizado para a aquisição das imagens deste trabalho (foto na figura 5.1) foi desenvolvido no EMBRAPA e seus detalhes podem ser vistos em [CRUVINEL et. al., 1990]. Ele é um equipamento de primeira geração em que a fonte de raios-X ou raios- $\Gamma$  (que pode ser proveniente de materiais com decaimento radioativo como Césio ou Amerício) e o detector (de cristal) são fixos, enquanto que a amostra em estudo sofre movimentos de rotação e translação. Todo o sistema é controlado por hardware e software desenvolvido para esta finalidade.

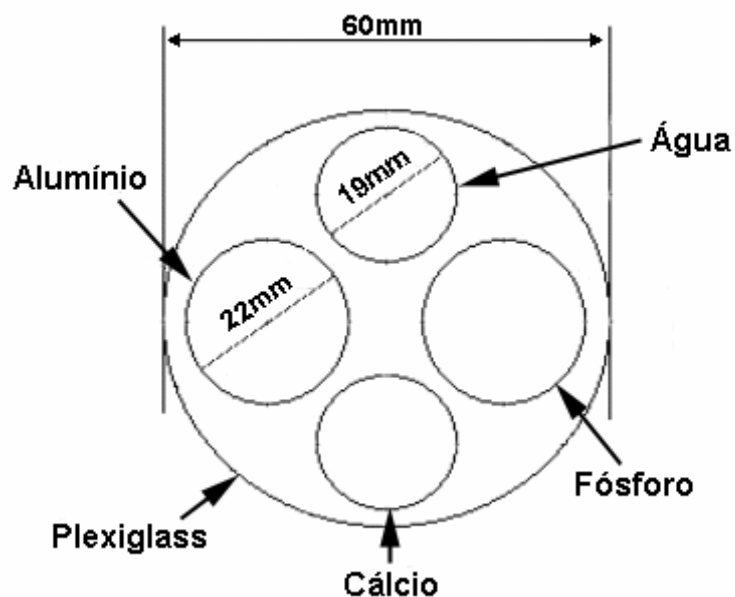


FIGURA 5.1 – Minitomógrafo do Embrapa

### 5.1.2 Corpo de Prova (Phantom)

O corpo de prova (ou *phantom*) é um objeto confeccionado especialmente para ser utilizado em experimentos. O *phantom* utilizado neste trabalho foi construído em *plexiglass* – que é um tipo de polímero – e tem um formato cilíndrico contendo quatro espaços (também de forma cilíndrica) onde são colocados os materiais a serem analisados. Nele foram inseridos 4 tipos de materiais: cálcio, fósforo, alumínio e água. O *phantom* tem 60mm de diâmetro, sendo que o espaço onde foram colocadas as amostras de cálcio e água possuem 19mm de diâmetro cada e os espaços onde foram colocadas as amostras de fósforo e alumínio têm 22mm de diâmetro cada, conforme ilustrado na Figura 5.2. Este corpo de prova foi imageado utilizando o tomógrafo descrito na seção anterior e as imagens geradas são objetos de estudo deste trabalho.





**FIGURA 5.2** – Diagrama de construção do corpo de prova (*phantom*)

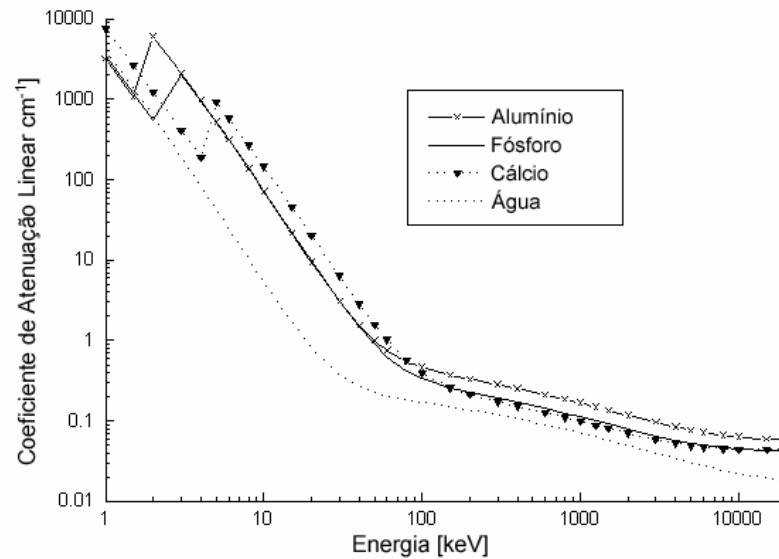
### 5.1.3 Imagens tomográficas obtidas e suas características

Para obtenção das projeções foram empregadas duas fontes de raios- $\Gamma$  (Césio e Amerício) e duas fontes de raios-X disponíveis. As energias provenientes do césio e do amerício são 662keV e 60keV, respectivamente e as duas energias provenientes de raios-X são 40keV e 85keV, sendo que as duas últimas foram escolhidas a partir da análise do comportamento dos coeficientes de atenuação (Figura 5.3) de cada material a ser identificado, de forma a obter melhor capacidade discriminatória. A Tabela I mostra os valores teóricos dos coeficientes de atenuação linear de cada material inclusive o pexiglass.

**TABELA I** – Coeficientes de atenuação linear em função das energias 40, 60, 85 e 662 keV para a água, alumínio, cálcio e fósforo.

Energia (keV)	Coeficientes de Atenuação Linear dos Energias Materiais				
	Água	Alumínio	Cálcio	Fósforo	Pexiglass
40	0,2683	1,5349	2,8365	1,4734	0,2880
60	0,2052	0,7479	1,0013	0,6315	0,2270
85	0,1750	0,6250	0,7750	0,5250	0,1000
662	0,0837	0,2019	0,1205	0,1365	0,0570

Fonte: Embrapa



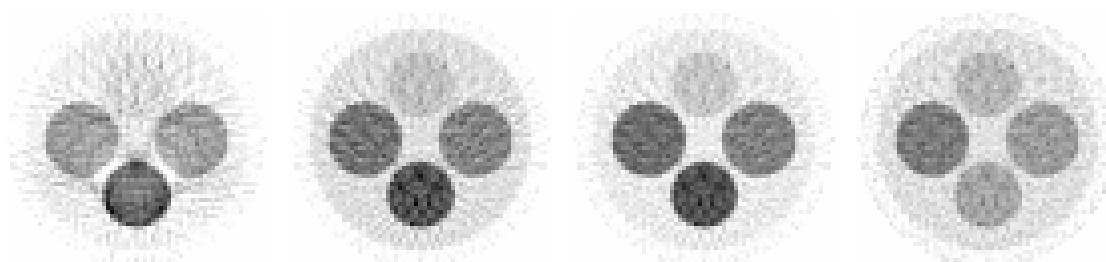
**FIGURA 5.3** – Curvas do coeficiente de atenuação linear em função da energia para a alumínio, fósforo, cálcio e água [HOMEM, 1998]

O procedimento de obtenção das imagens foi realizado de forma que fossem geradas imagens com alto nível de ruído, utilizando translação total de 7 cm, passo linear de 0,109 cm, rotação total de  $180^\circ$ , passo angular de  $2,81^\circ$  e tempo de contagem de fótons em 3 segundos. Foram utilizadas imagens com alto nível de ruído para verificar a eficiência dos classificadores propostos mesmo com imagens de baixa qualidade.

Uma vez obtidas as projeções, as imagens foram reconstruídas pelo método de retroprojeção filtrada. Assim, foram geradas quatro imagens no tamanho  $65 \times 65$  pixels, cada uma correspondendo a uma das energias utilizadas. Estas imagens foram convertidas para o formato BMP (Bitmap), para serem utilizadas no Matlab de forma que pudessem ser classificadas. Nesta conversão os valores do coeficiente de atenuação contidos em cada imagem foram normalizados para um intervalo de 256 valores (intensidades). As imagens utilizadas nesse trabalho são as mesmas utilizadas por Ponti Jr [2004] e são mostradas na Figura 5.4.

As imagens mostradas são imagens negativas das originais e foram trabalhadas dessa forma para obter uma melhor visualização, uma vez que esta inversão na escala de cinzas não

altera o resultado da classificação. Ponti Jr também normalizou estas imagens de modo que ao serem combinadas em uma única imagem multiespectral não houvesse muita perda de precisão. As bandas de 40, 60, 85 e 662 keV foram normalizadas, respectivamente, por um fator de 17, 7, 5 e 2. Estes valores foram encontrados através dos valores máximos experimentais dos coeficientes de atenuação linear encontrados em cada energia utilizada para as imagens. Os valores observados para as bandas de 40, 60, 85 e 662 keV foram, respectivamente, 4129, 1740, 1148 e 336. Estes valores estão abaixo dos valores teóricos e esta diferença é bastante expressiva em alguns casos. Ponti Jr [2004] atribui esta diferença às impurezas presentes no material em estudo e ao ruído, incluindo o ruído inerente ao próprio processo de aquisição das projeções tomográficas e o ruído decorrente do dispositivo eletrônico de aquisição (normalmente ruído gaussiano).



**FIGURA 5.4** - Imagens adquiridas com energias de 40, 60, 85 e 662 keV.

## **5.2 Classificação das Imagens**

Nesta seção mostraremos o procedimento utilizado para classificação das imagens e avaliação dos classificadores, bem como as ferramentas utilizadas.

### **5.2.1 MATLAB**

O MATLAB é um software de alto desempenho para computação técnica que pode ser aplicado em diversas áreas, incluindo o Reconhecimento de Padrões. É um sistema interativo com diversas facilidades para trabalhar com vetores e matrizes, de forma que a

implementação de um algoritmo que utiliza esses elementos se torna bem mais rápida no MATLAB do que em linguagens tradicionais como C, Fortran e Pascal.

Ele também conta com diversas *toolboxes* específicas para vários tipos de aplicações, incluindo aí uma *toolbox* específica para redes neurais, que é utilizada neste trabalho. O MATLAB também permite a criação de outras *toolboxes* e interfaces que podem ser integradas a ele para atender as necessidades do usuário.

### 5.2.2 PRTools

PRTools é uma *toolbox* para o MATLAB desenvolvida pelo grupo de Reconhecimento de Padrões da Universidade de Delft na Holanda [DUIN et. al., 2004]. A versão atual (4.0) do PRTools implementa aproximadamente 200 rotinas de reconhecimento de padrões, cobrindo uma grande parte da área de reconhecimento de padrões estatísticos. A parte de redes neurais foi parcialmente deixada de lado, pois o MATLAB já inclui uma boa *toolbox* nesta área. Entretanto algumas interfaces para as rotinas básicas foram implementadas de forma que facilitasse a comparação das técnicas de reconhecimento de padrões utilizando redes neurais com as técnicas tradicionais.

### 5.2.3 Procedimentos

Foram realizados experimentos de classificação com as imagens descritas na seção 5.1.3. No procedimento de classificação as quatro imagens BMP são combinadas como bandas de uma única imagem multiespectral. Foram consideradas 6 classes:







- Água,
- Alumínio,
- Fósforo,
- Cálcio,
- Plexiglass,

- Fundo da imagem.

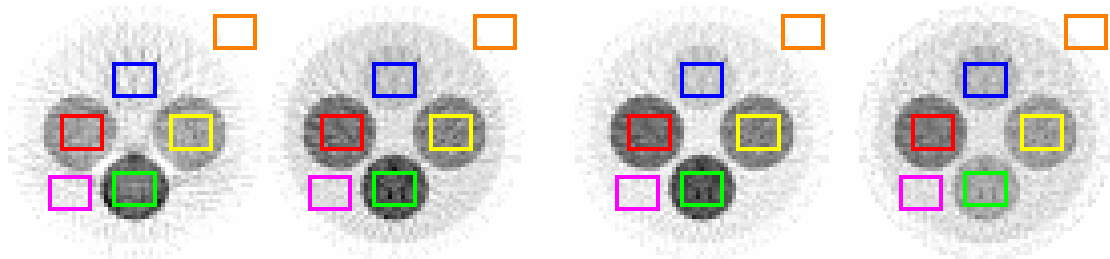
A probabilidade a priori foi considerada igual para todas as classes por não haver informações que pudessem ser utilizadas.

Para construir as imagens temáticas (imagem classificada) é necessário escolher cores diferentes para representar cada uma das classes de forma que o resultado possa ser facilmente analisado. As cores escolhidas foram as mesmas utilizadas por Ponti Jr [2004] e estão relacionadas na Tabela II.

**TABELA II** – Cores associadas às classes nas imagens temáticas

Classe	Cor	
Água	Azul	
Alumínio	Vermelho	
Fósforo	Amarelo	
Cálcio	Verde	
Plexiglass	Cinza Escuro	
Fundo	Cinza Claro	

Para o treinamento e teste foram escolhidas 80 amostras de cada classe, em janelas de 10x8 pixels que podem ser vistas na Figura 5.5, totalizando 480 amostras. Este número de amostras é reduzido devido ao tamanho das imagens geradas pelo tomógrafo, que têm apenas 65x65 pixels cada uma. O conjunto de amostras foi subdividido em dois subconjuntos de 240 amostras cada (40 de cada classe), sendo que um subconjunto foi destinado ao treinamento dos classificadores e o outro subconjunto foi destinado ao teste dos classificadores treinados, seguindo assim a técnica de *hold-out*.



**FIGURA 5.5** - Janelas de pixels escolhidos como amostras nas imagens com energias de 40, 60, 85 e 662 keV.

Os classificadores escolhidos para este trabalho são baseados em Redes Neurais. São eles: o Perceptron de Múltiplas Camadas e as Redes de Função de Base Radial, sendo que foram treinados classificadores com diferentes quantidades de neurônios na camada oculta, pois, conforme descrito por Duda et. al. [2000], a quantidade ideal de neurônios depende dos dados a serem classificados e não há nenhuma maneira precisa de determinar esta quantidade a priori. Desse modo, todos os classificadores treinados têm uma camada de entrada com 4 neurônios (nós), que correspondem a cada uma das bandas da imagem, uma camada intermediária (oculta) com número de neurônios variável e uma camada de saída com 6 neurônios, que correspondem a cada uma das classes consideradas.

Para o Perceptron de Múltiplas Camadas foram treinados classificadores contendo de 2 a 25 neurônios em uma única camada oculta. Os classificadores baseados no Perceptron de Múltiplas Camadas tendem a apresentar resultados bastante diferentes dependendo dos parâmetros de iniciação da rede, que são aleatórios, portanto para cada tipo de estrutura testada foram treinados 100 classificadores diferentes (com diferentes parâmetros de iniciação), de modo que fosse possível obter estatísticas confiáveis de seu desempenho. Assim foram obtidas as médias e desvio padrão do erro estimado e do coeficiente Kappa para cada tipo de classificador.

Para as Redes de Função de Base Radial também foram treinados classificadores contendo de 2 a 25 neurônios na camada oculta. Este tipo de rede neural não apresenta a mesma instabilidade do Perceptron de Múltiplas Camadas, pois os classificadores gerados com a mesma estrutura e as mesmas amostras tendem a ser sempre muito semelhantes. Apesar dessa observação os experimentos também foram repetidos 100 vezes com diferentes iniciações para manter um padrão.

Cada classificador treinado teve seu desempenho medido através de um teste com as amostras previamente designadas. Dessa forma foram obtidos os erros estimados pelo método *hold-out*, bem como os Coeficientes Kappa de cada classificador.

O erro estimado representa a taxa de erro que se espera do classificador e é obtido pela razão entre a quantidade de erros do classificador e o número de amostras testadas. O Coeficiente Kappa é obtido utilizando-se a equação 3.4, onde o  $P(E)$  terá sempre o valor 0,166... pois temos 6 classes equiprováveis, e  $P(A)$  será obtido facilmente obtido subtraindo-se o erro estimado de 1.

Em seguida foram executados os experimentos utilizando combinação de classificadores. Foi utilizada primeiramente a técnica de Bagging, e em seguida foram utilizados os combinadores de Decision Templates e Dempster-Shafer. Todos foram utilizados sobre os classificadores base Perceptron de Múltiplas Camadas e Redes de Função de Base Radial, com as mesmas configurações e quantidade de neurônios de seus respectivos testes individuais. Para todos os experimentos com combinadores foram treinados 10 classificadores-base diferentes, variando sempre a inicialização da rede (e também o conjunto de amostras nos experimentos com Bagging). A combinação foi feita a partir das saídas desses 10 classificadores. Todos os experimentos também foram repetidos 100 vezes para que as estatísticas fossem confiáveis. Assim nos experimentos utilizando Decision Templates e Dempster-Shafer foram variados os parâmetros de iniciação em cada uma das 100 execuções. E nos experimentos com Bagging além da variação dos parâmetros de iniciação tivemos também a variação no conjunto de treinamento, através dos diferentes conjuntos tipo *bootstrap* gerados para cada classificador. Para cada experimento com combinação também foram obtidos as estimativas de erro por Hold-Out e os Coeficientes Kappa, para que fosse possível comparar os resultados com o dos classificadores individuais.

A partir dos resultados obtidos foi observado que seria interessante estudar o que aconteceria ao combinar a técnica de Bagging com os combinadores Decision Templates e Dempster-Shafer, isto é, utilizar Decision Templates e Dempster-Shafer para fazer a combinação dos classificadores gerados para cada amostra do tipo bootstrap, e assim comparar estes resultados com os obtidos com cada técnica de combinação aplicada individualmente. Dessa forma foram conduzidos experimentos utilizando Bagging com Dempster-Shafer e Bagging com Decision Templates, ambos sobre os classificadores base Perceptron de Múltiplas Camadas e Redes de Função de Base Radial, com as mesmas configurações e quantidade de neurônios dos experimentos anteriores, e também fazendo a combinação a partir de 10 classificadores-base. Estes experimentos também foram repetidos 100 vezes, variando as iniciações e também os conjuntos do tipo bootstrap utilizados, e assim foram obtidos as estimativas de erro por Hold-Out e os coeficientes Kappa.



## 6. RESULTADOS

---

Neste capítulo são apresentados os resultados das classificações das imagens tomográficas multiespectrais segundo os classificadores descritos no Capítulo 3, seguindo a metodologia apresentada no Capítulo 5.

São apresentados os resultados obtidos com os classificadores Perceptron de Múltiplas Camadas (Seção 6.1) e Redes de Função de Base Radial (Seção 6.2), tanto para os classificadores aplicados individualmente como para a aplicação utilizando os combinadores: Bagging, Decision Templates (DT), Dempster-Shafer (DS), Bagging c/ Decision Templates (BAGDT) e Bagging c/ Dempster-Shafer (BAGDS). As tabelas apresentadas resumem o desempenho apresentado pelos classificadores com cada tipo de estrutura interna (neurônios na camada oculta), tanto individualmente quanto com o uso de combinadores. Os melhores resultados em cada coluna estão marcados em azul e negrito, e os piores estão marcados em vermelho.

Também foram incluídas imagens temáticas das redes com 2, 13 e 25 neurônios na camada oculta. Como cada experimento foi repetido 100 vezes existem 100 imagens temáticas para cada um deles e escolher qualquer uma delas sem critério poderia levar a conclusões errôneas, portanto foram escolhidas as imagens temáticas dos classificadores de melhor e pior desempenho (medidos por Erro Estimado por Hold-Out e Coeficiente Kappa) dentre os 100.

## 6.1 Perceptron de Múltiplas Camadas

TABELA III – Erro Estimado médio para o Perceptron de Múltiplas Camadas

Nós na camada oculta	Individual	Bagging	Decision Templates	Dempster-Shafer	Bagging c/ Decision Templates	Bagging c/ Dempster-Shafer
2	0,5720	0,3675	0,0349	0,0613	0,0358	0,0552
3	0,2689	0,0493	0,0163	0,0275	0,0166	0,0269
4	0,1318	0,0200	0,0141	0,0177	0,0145	0,0203
5	0,0976	0,0170	0,0123	0,0151	0,0150	0,0172
6	0,0741	0,0168	0,0127	0,0139	0,0155	0,0171
7	0,0681	0,0175	0,0129	0,0138	0,0148	0,0164
8	0,0636	0,0179	0,0130	0,0137	0,0163	0,0177
9	0,0511	0,0190	0,0134	0,0138	0,0167	0,0174
10	0,0570	0,0190	0,0135	0,0139	0,0172	0,0183
11	0,0497	0,0191	0,0136	0,0143	0,0179	0,0188
12	0,0603	0,0203	0,0136	0,0143	0,0181	0,0191
13	0,0525	0,0196	0,0137	0,0145	0,0185	0,0196
14	0,0477	0,0204	0,0140	0,0151	0,0187	0,0199
15	0,0470	0,0210	0,0143	0,0150	0,0190	0,0198
16	0,0403	0,0211	0,0145	0,0147	0,0196	0,0205
17	0,0426	0,0205	0,0148	0,0153	0,0185	0,0194
18	0,0402	0,0211	0,0146	0,0149	0,0192	0,0202
19	0,0340	0,0207	0,0145	0,0154	0,0198	0,0209
20	0,0451	0,0213	0,0151	0,0153	0,0201	0,0209
21	0,0275	0,0203	0,0155	0,0156	0,0201	0,0209
22	0,0366	0,0215	0,0150	0,0158	0,0200	0,0212
23	0,0430	0,0213	0,0150	0,0156	0,0200	0,0207
24	0,0393	0,0217	0,0151	0,0156	0,0202	0,0211
25	0,0348	0,0214	0,0148	0,0162	0,0204	0,0209

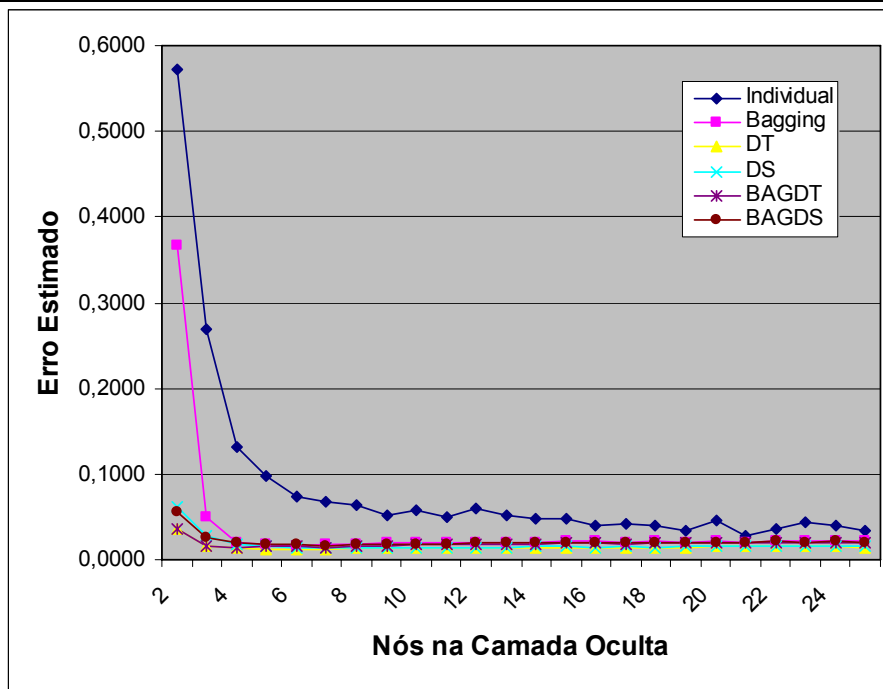


FIGURA 6.1 – Variação do Erro Estimado médio para o Perceptron de Múltiplas Camadas

TABELA IV – Coeficiente Kappa médio para o Perceptron de Múltiplas Camadas

Nós na camada oculta	Individual	Bagging	Decision Templates	Dempster-Shafer	Bagging c/ Decision Templates	Bagging c/ Dempster-Shafer
2	0,3137	0,5591	0,9581	0,9265	0,9570	0,9338
3	0,6773	0,9409	0,9805	0,9671	0,9801	0,9678
4	0,8419	0,9760	0,9831	0,9788	0,9826	0,9757
5	0,8829	0,9796	0,9853	0,9819	0,9821	0,9794
6	0,9111	0,9799	0,9848	0,9833	0,9814	0,9795
7	0,9183	0,9790	0,9845	0,9835	0,9822	0,9803
8	0,9237	0,9786	0,9844	0,9836	0,9804	0,9788
9	0,9387	0,9773	0,9840	0,9835	0,9800	0,9792
10	0,9316	0,9773	0,9838	0,9833	0,9794	0,9780
11	0,9404	0,9771	0,9837	0,9829	0,9785	0,9775
12	0,9277	0,9757	0,9837	0,9829	0,9783	0,9771
13	0,9371	0,9765	0,9836	0,9827	0,9778	0,9765
14	0,9428	0,9756	0,9832	0,9819	0,9776	0,9762
15	0,9436	0,9748	0,9828	0,9821	0,9773	0,9762
16	0,9516	0,9747	0,9827	0,9824	0,9765	0,9755
17	0,9489	0,9755	0,9823	0,9817	0,9778	0,9768
18	0,9518	0,9747	0,9825	0,9822	0,9770	0,9758
19	0,9592	0,9752	0,9826	0,9816	0,9762	0,9749
20	0,9459	0,9745	0,9819	0,9817	0,9759	0,9749
21	0,9671	0,9756	0,9815	0,9813	0,9759	0,9750
22	0,9561	0,9743	0,9820	0,9811	0,9760	0,9746
23	0,9484	0,9744	0,9821	0,9813	0,9761	0,9752
24	0,9529	0,9740	0,9819	0,9813	0,9758	0,9747
25	0,9583	0,9744	0,9822	0,9806	0,9756	0,9750

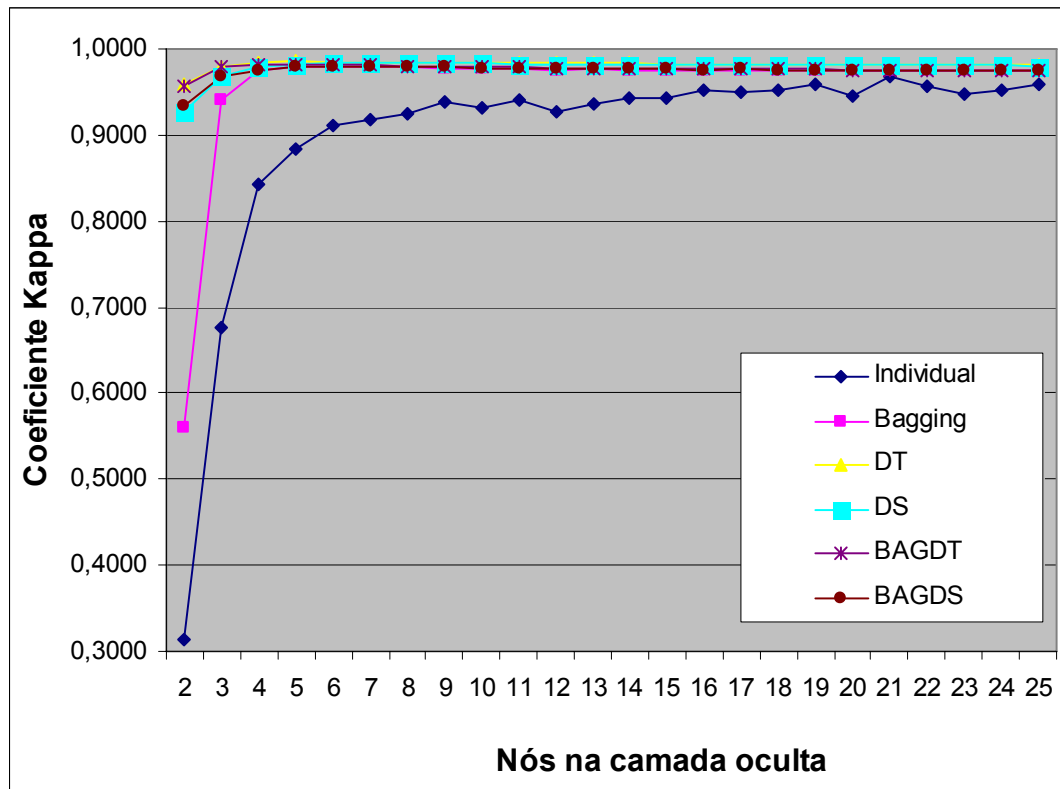
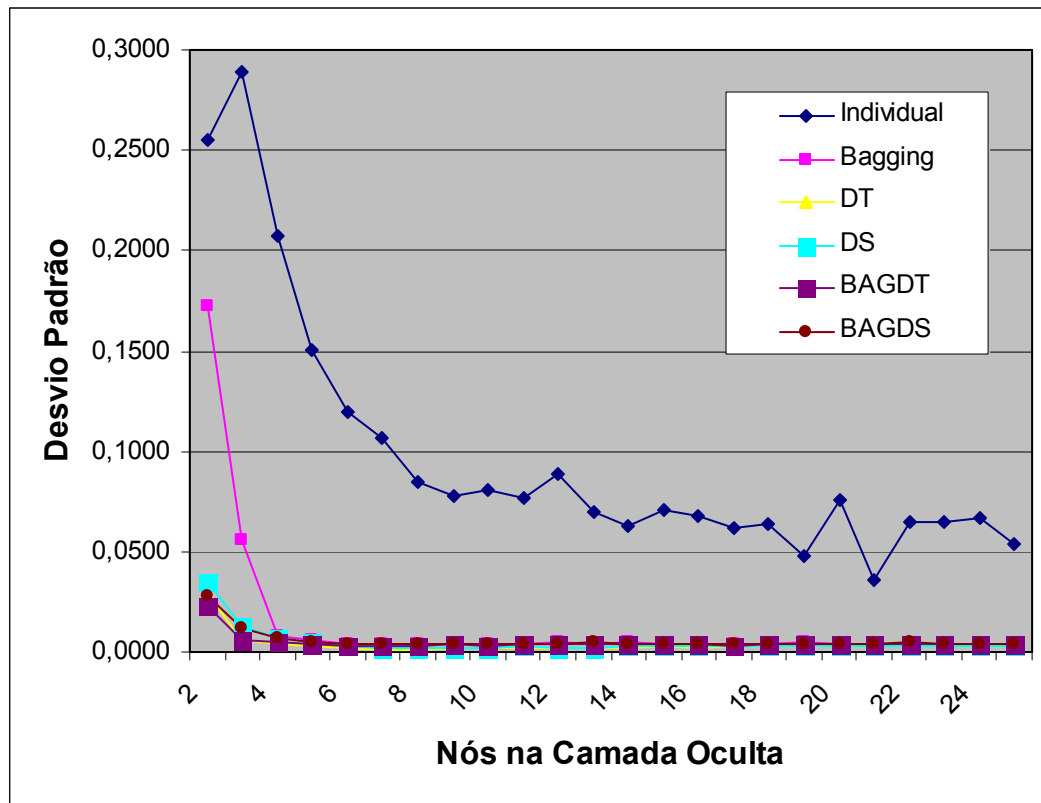


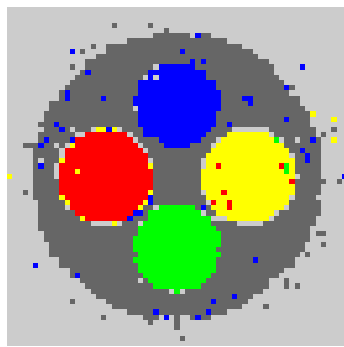
FIGURA 6.2 – Variação do Coeficiente Kappa médio para o Perceptron de Múltiplas Camadas

**TABELA V – Desvio Padrão do Erro Estimado e Coeficiente Kappa para o Perceptron de Múltiplas Camadas**

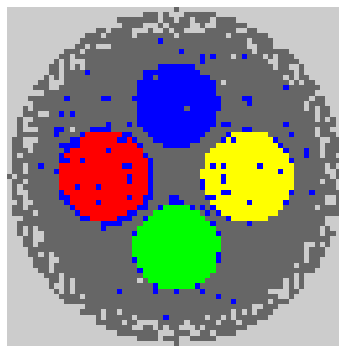
Nós na camada oculta	Individual	Bagging	Decision Templates	Dempster-Shafer	Bagging c/ Decision Templates	Bagging c/ Dempster-Shafer
2	0,2554	0,1729	0,0286	0,0348	0,0228	0,0281
3	0,2892	0,0559	0,0061	0,0133	0,0064	0,0117
4	0,2073	0,0079	0,0044	0,0065	0,0046	0,0069
5	0,1504	0,0062	0,0029	0,0046	0,0038	0,0048
6	0,1199	0,0039	0,0020	0,0028	0,0030	0,0043
7	0,1065	0,0039	0,0013	0,0024	0,0026	0,0038
8	0,0842	0,0037	0,0014	0,0020	0,0029	0,0039
9	0,0777	0,0039	0,0018	0,0021	0,0037	0,0039
10	0,0807	0,0042	0,0018	0,0023	0,0032	0,0037
11	0,0770	0,0038	0,0020	0,0026	0,0036	0,0039
12	0,0887	0,0050	0,0020	0,0024	0,0038	0,0042
13	0,0698	0,0044	0,0021	0,0025	0,0044	0,0047
14	0,0623	0,0047	0,0023	0,0026	0,0038	0,0044
15	0,0704	0,0042	0,0025	0,0027	0,0042	0,0042
16	0,0680	0,0041	0,0023	0,0027	0,0042	0,0043
17	0,0618	0,0041	0,0027	0,0030	0,0034	0,0038
18	0,0638	0,0042	0,0026	0,0025	0,0035	0,0036
19	0,0480	0,0046	0,0026	0,0035	0,0041	0,0042
20	0,0756	0,0044	0,0027	0,0034	0,0040	0,0039
21	0,0360	0,0036	0,0029	0,0031	0,0041	0,0041
22	0,0648	0,0039	0,0025	0,0031	0,0045	0,0047
23	0,0650	0,0043	0,0027	0,0027	0,0037	0,0042
24	0,0667	0,0039	0,0030	0,0033	0,0036	0,0038
25	0,0538	0,0045	0,0027	0,0033	0,0041	0,0040



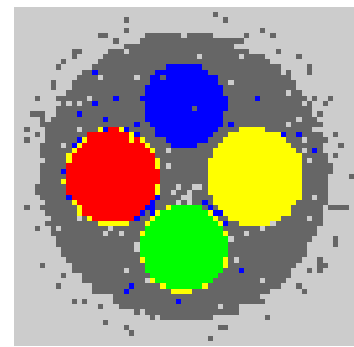
**FIGURA 6.3 – Variação do Desvio Padrão para o Perceptron de Múltiplas Camadas**



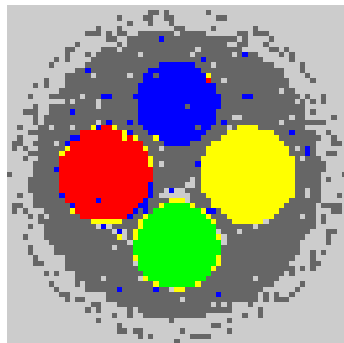
Individual  
 Erro: 0,0125 / Kappa: 0,9850



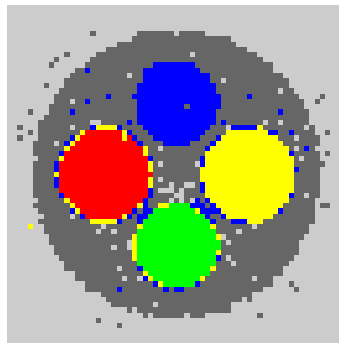
Bagging  
 Erro: 0,0250 / Kappa: 0,9700



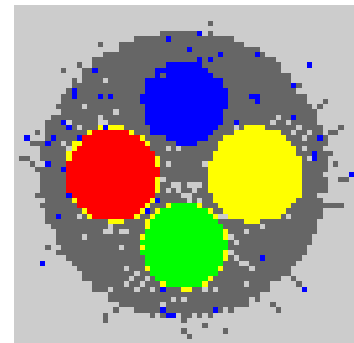
Decision Templates  
 Erro: 0,0083 / Kappa: 0,9900



Dempster-Shafer  
 Erro: 0,0125 / Kappa: 0,9850



Bagging c/ Decision Templates  
 Erro: 0,0083 / Kappa: 0,9900

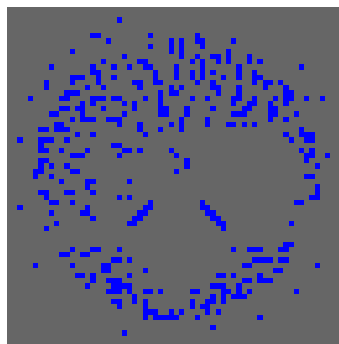


Bagging c/ Dempster-Shafer  
 Erro: 0,0083 / Kappa: 0,9900

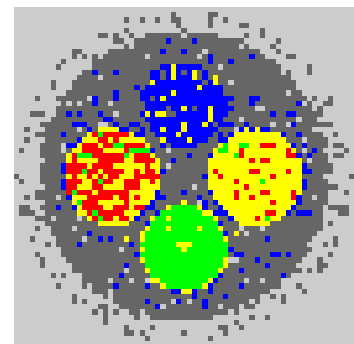
**FIGURA 6.4** – Imagens Temáticas para Perceptron de Múltiplas Camadas com 2 nós na camada oculta (melhores casos)



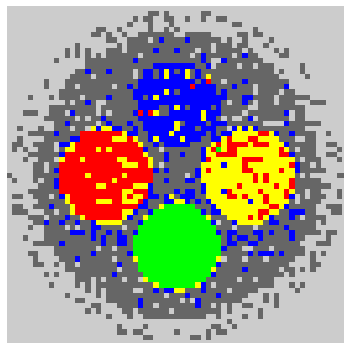
Classificador Individual  
 Erro: 0,8792 / Kappa: -0,0550



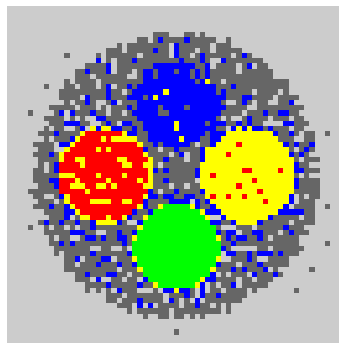
Bagging  
 Erro: 0,8500 / Kappa: -0,0200



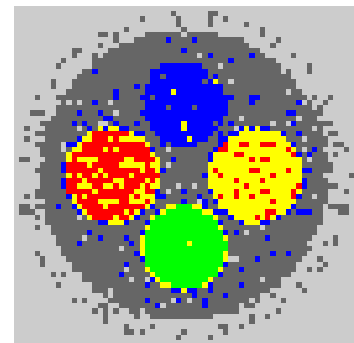
Decision Templates  
 Erro: 0,1708 / Kappa: 0,7950



Dempster-Shafer  
 Erro: 0,1500 / Kappa: 0,8200

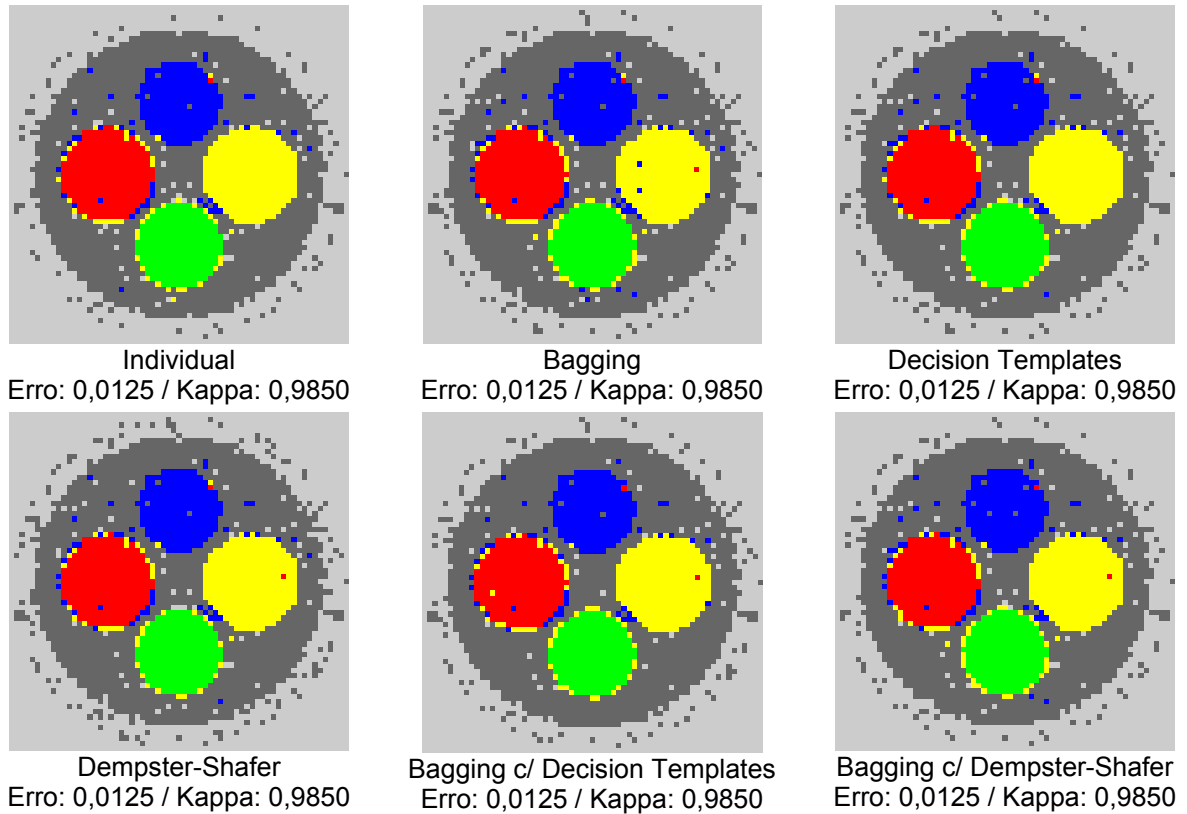


Bagging c/ Decision Templates  
 Erro: 0,1208 / Kappa: 0,8550

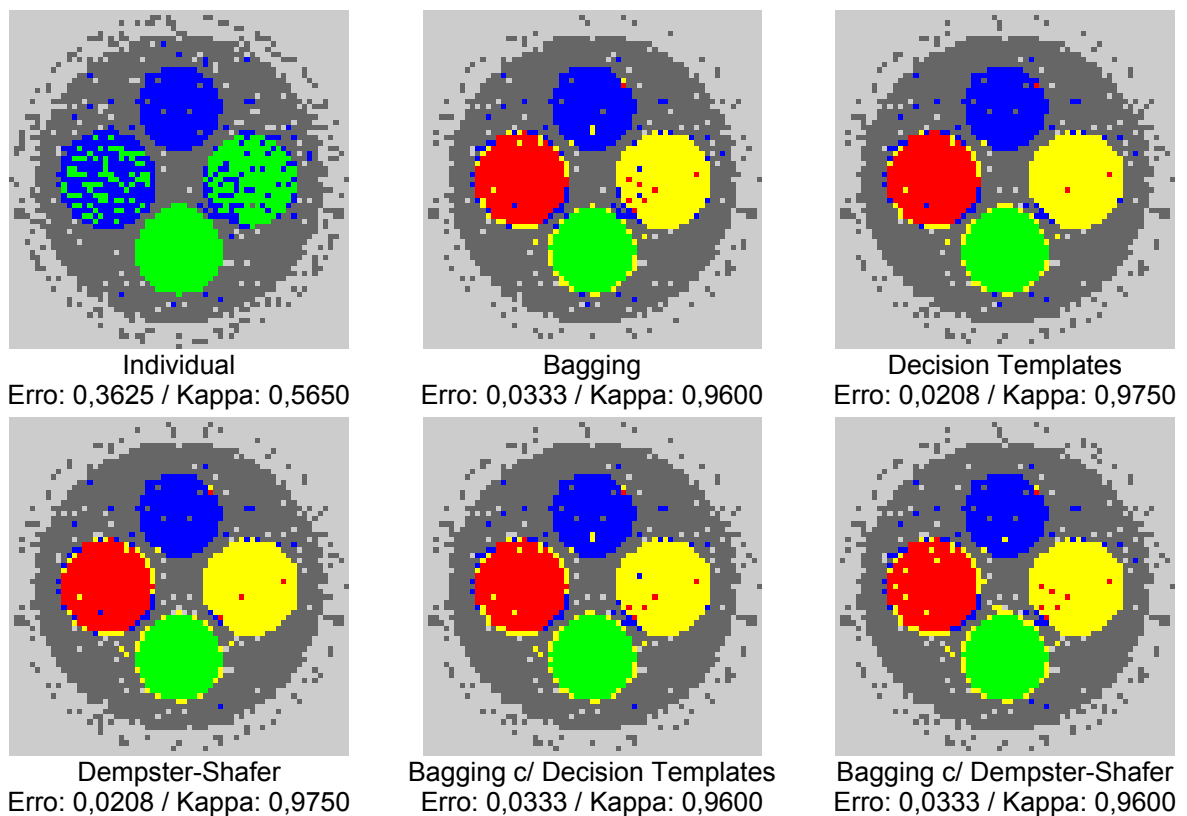


Bagging c/ Dempster-Shafer  
 Erro: 0,1333 / Kappa: 0,8400

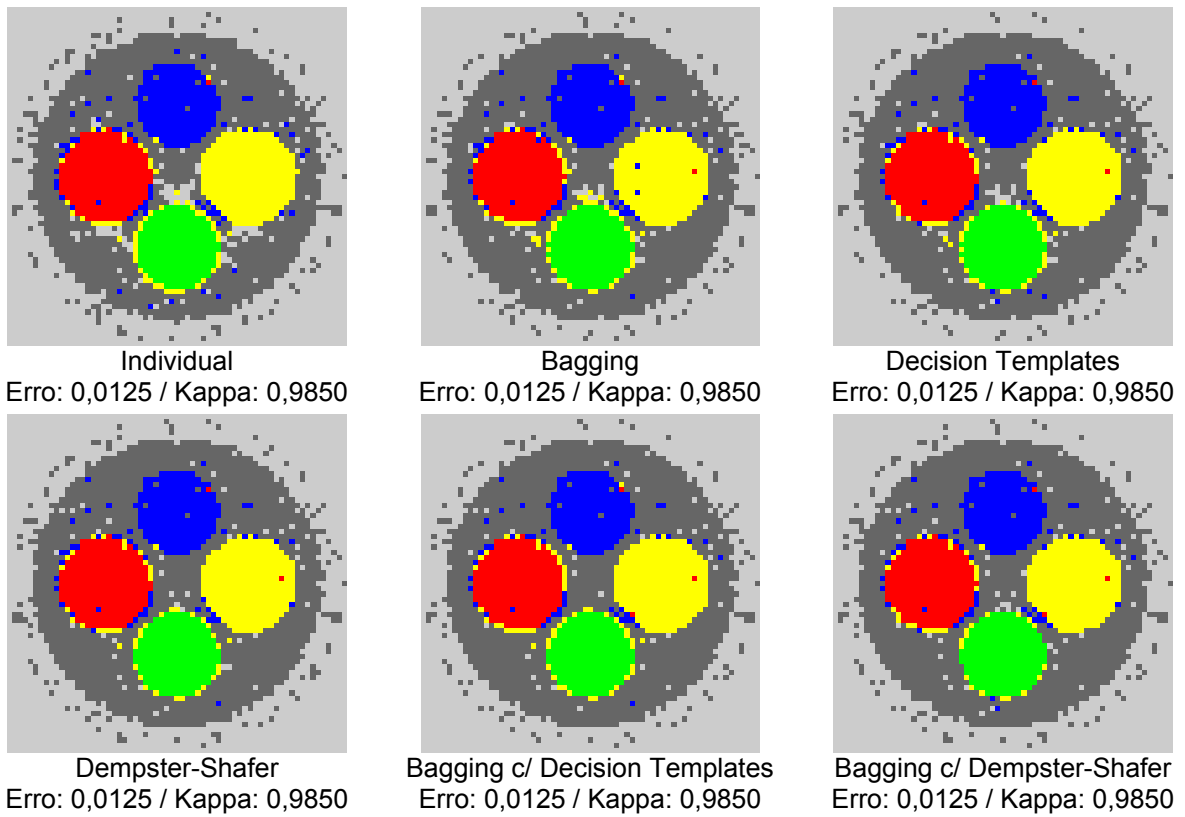
**FIGURA 6.5** – Imagens Temáticas para Perceptron de Múltiplas Camadas com 2 nós na camada oculta (piores casos)



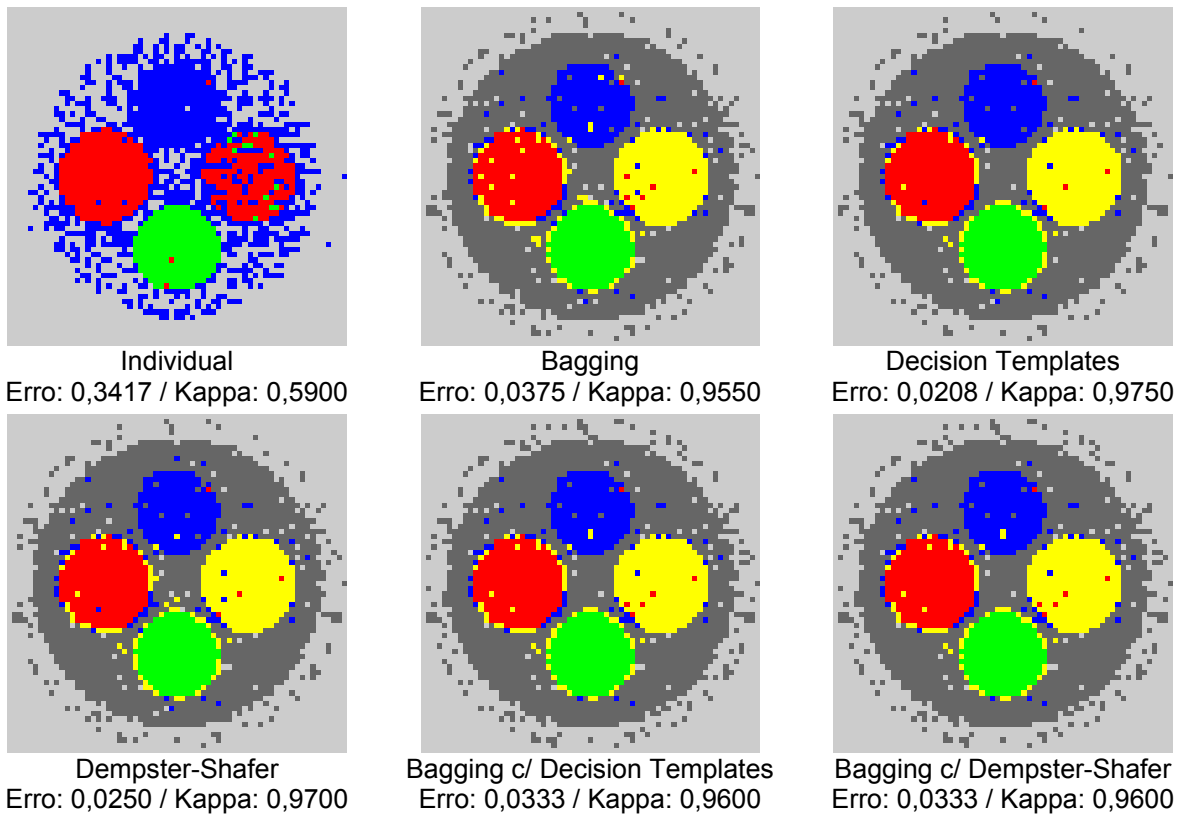
**FIGURA 6.6** – Imagens Temáticas para Perceptron de Múltiplas Camadas com 13 nós na camada oculta (melhores casos)



**FIGURA 6.7** – Imagens Temáticas para Perceptron de Múltiplas Camadas com 13 nós na camada oculta (piores casos)



**FIGURA 6.8** – Imagens Temáticas para Perceptron de Múltiplas Camadas com 25 nós na camada oculta (melhores casos)



**FIGURA 6.9** – Imagens Temáticas para Perceptron de Múltiplas Camadas com 25 nós na camada oculta (piores casos)

## 6.2 Redes de Função de Base Radial

TABELA VI – Erro Estimado médio para Redes de Função de Base Radial

Nós na camada oculta	Individual	Bagging	Decision Templates	Dempster-Shafer	Bagging c/ Decision Templates	Bagging c/ Dempster-Shafer
2	0,5125	0,3720	0,4458	0,4333	0,3391	0,4218
3	0,3958	0,2118	0,3583	0,3583	0,2334	0,3605
4	0,2958	0,1675	0,2583	0,2583	0,2094	0,1776
5	0,2292	0,1953	0,2333	0,2333	0,2100	0,2083
6	0,1333	0,1269	0,1292	0,1292	0,1332	0,1341
7	0,0750	0,0900	0,0708	0,0708	0,0853	0,0909
8	0,0833	0,0733	0,0833	0,0833	0,0717	0,0719
9	0,0750	0,0589	0,0750	0,0750	0,0596	0,0595
10	0,0833	0,0500	0,0875	0,0875	0,0497	0,0487
11	0,0708	0,0444	0,0667	0,0667	0,0432	0,0422
12	0,0500	0,0382	0,0500	0,0500	0,0365	0,0364
13	0,0390	0,0329	0,0302	0,0292	0,0321	0,0319
14	0,0334	0,0265	0,0333	0,0333	0,0252	0,0251
15	0,0252	0,0201	0,0250	0,0250	0,0197	0,0196
16	0,0232	0,0186	0,0213	0,0208	0,0181	0,0183
17	0,0219	0,0175	0,0197	0,0208	0,0175	0,0170
18	0,0177	0,0160	0,0170	0,0167	0,0161	0,0160
19	0,0173	0,0150	0,0166	0,0167	0,0145	0,0145
20	0,0145	0,0135	0,0125	0,0125	0,0137	0,0135
21	0,0130	0,0132	0,0125	0,0125	0,0130	0,0131
22	0,0126	0,0129	0,0125	0,0125	0,0127	0,0127
23	0,0126	0,0126	0,0125	0,0125	0,0127	0,0125
24	0,0125	0,0126	0,0125	0,0125	0,0126	0,0126
25	0,0125	0,0127	0,0125	0,0125	0,0126	0,0126

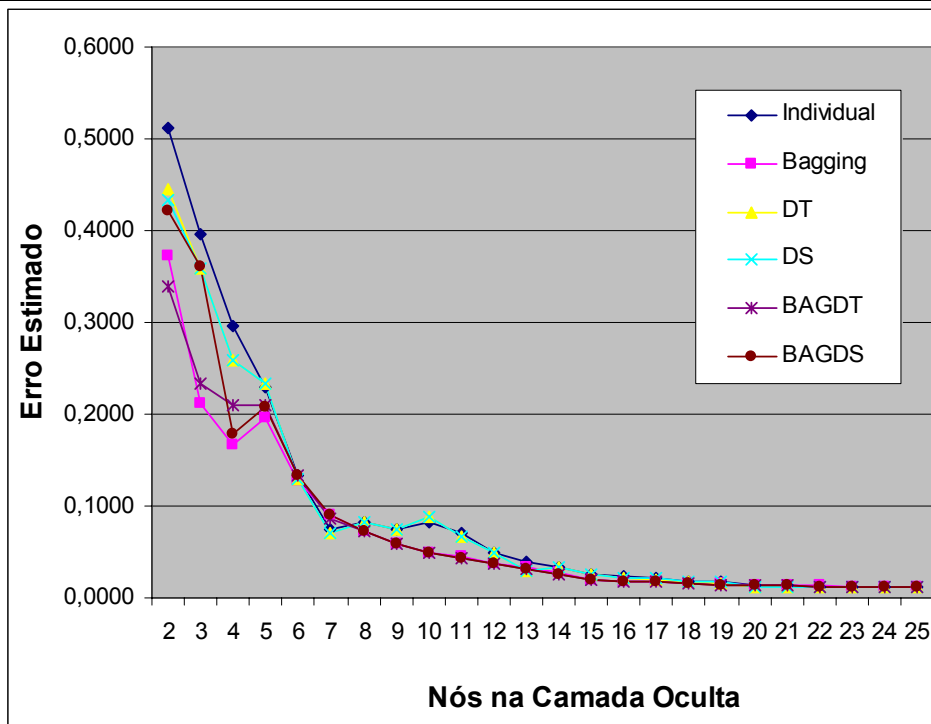


FIGURA 6.10 – Variação do Erro Estimado médio para Redes de Função de Base Radial



TABELA VII – Coeficiente Kappa médio para Redes de Função de Base Radial

Nós na camada oculta	Individual	Bagging	Decision Templates	Dempster-Shafer	Bagging c/ Decision Templates	Bagging c/ Dempster-Shafer
2	0,3850	0,5536	0,4650	0,4800	0,5931	0,4939
3	0,5250	0,7459	0,5700	0,5700	0,7199	0,5675
4	0,6450	0,7990	0,6900	0,6900	0,7488	0,7869
5	0,7250	0,7656	0,7200	0,7200	0,7480	0,7500
6	0,8400	0,8478	0,8450	0,8450	0,8402	0,8391
7	0,9100	0,8921	0,9150	0,9150	0,8976	0,8909
8	0,9000	0,9120	0,9000	0,9000	0,9140	0,9137
9	0,9100	0,9294	0,9100	0,9100	0,9285	0,9286
10	0,9000	0,9400	0,8950	0,8950	0,9404	0,9416
11	0,9150	0,9468	0,9200	0,9200	0,9482	0,9494
12	0,9400	0,9542	0,9400	0,9400	0,9562	0,9563
13	0,9532	0,9605	0,9638	0,9650	0,9615	0,9617
14	0,9600	0,9682	0,9600	0,9600	0,9698	0,9699
15	0,9698	0,9759	0,9700	0,9700	0,9764	0,9765
16	0,9722	0,9777	0,9744	0,9750	0,9783	0,9781
17	0,9738	0,9790	0,9764	0,9750	0,9790	0,9796
18	0,9788	0,9808	0,9797	0,9800	0,9807	0,9808
19	0,9793	0,9820	0,9801	0,9800	0,9826	0,9826
20	0,9826	0,9838	0,9850	0,9850	0,9836	0,9838
21	0,9844	0,9842	0,9850	0,9850	0,9844	0,9843
22	0,9849	0,9845	0,9850	0,9850	0,9848	0,9848
23	0,9849	0,9849	0,9850	0,9850	0,9848	0,9850
24	0,9850	0,9849	0,9850	0,9850	0,9849	0,9849
25	0,9850	0,9848	0,9850	0,9850	0,9849	0,9849

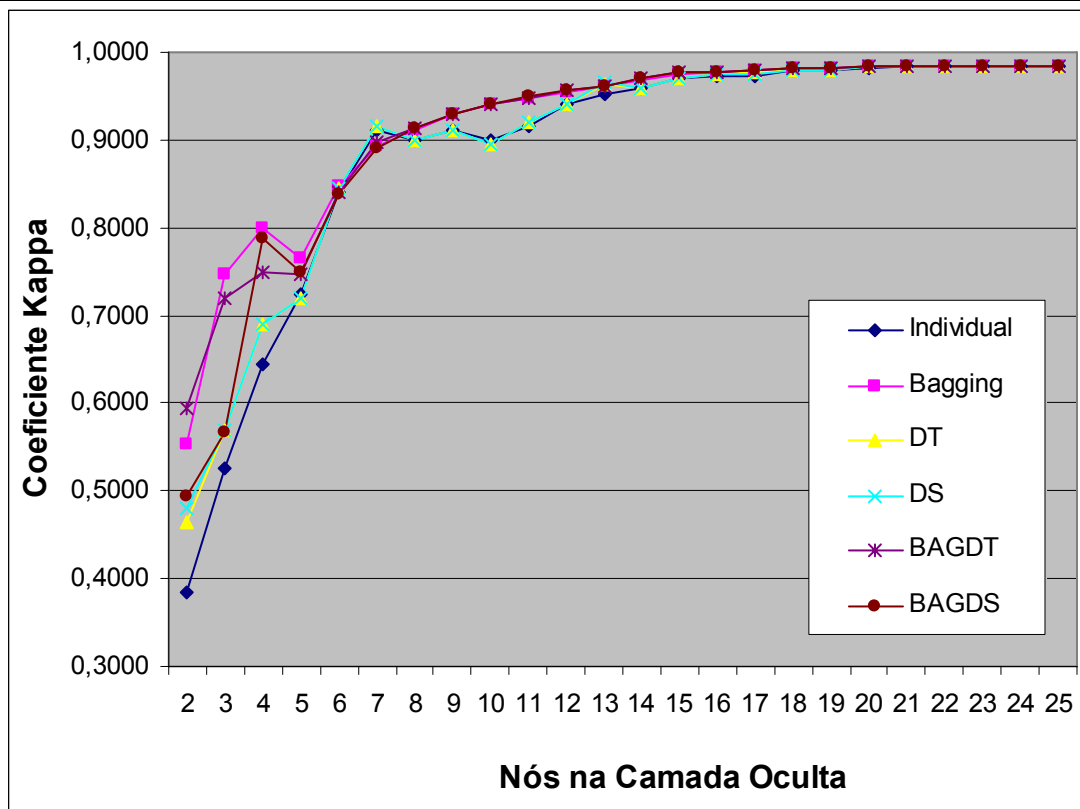


FIGURA 6.11 – Variação do Coeficiente Kappa médio para o Perceptron de Múltiplas Camadas

TABELA VIII – Desvio Padrão do Erro Estimado e Coeficiente Kappa para Redes de Função de Base Radial

Nós na camada oculta	Single	Bagging	Decision Templates	Dempster-Shafer	Bagging c/ Decision Templates	Bagging c/ Dempster-Shafer
2	0,0000	0,0000	0,0000	0,0000	0,0432	0,0171
3	0,0000	0,0000	0,0000	0,0000	0,0357	0,0123
4	0,0000	0,0000	0,0000	0,0000	0,0185	0,0287
5	0,0000	0,0000	0,0000	0,0000	0,0139	0,0119
6	0,0000	0,0000	0,0000	0,0000	0,0068	0,0067
7	0,0000	0,0000	0,0000	0,0000	0,0097	0,0097
8	0,0000	0,0000	0,0000	0,0000	0,0097	0,0100
9	0,0000	0,0000	0,0000	0,0000	0,0079	0,0097
10	0,0000	0,0000	0,0000	0,0000	0,0066	0,0073
11	0,0000	0,0000	0,0000	0,0000	0,0063	0,0057
12	0,0000	0,0000	0,0000	0,0000	0,0051	0,0056
13	0,0066	0,0021	0,0019	0,0021	0,0050	0,0041
14	0,0004	0,0000	0,0000	0,0000	0,0053	0,0050
15	0,0012	0,0000	0,0000	0,0000	0,0029	0,0033
16	0,0039	0,0015	0,0015	0,0015	0,0023	0,0022
17	0,0040	0,0019	0,0020	0,0019	0,0021	0,0020
18	0,0025	0,0014	0,0011	0,0014	0,0026	0,0021
19	0,0015	0,0008	0,0006	0,0008	0,0024	0,0022
20	0,0029	0,0000	0,0004	0,0000	0,0020	0,0020
21	0,0015	0,0000	0,0000	0,0000	0,0014	0,0015
22	0,0006	0,0000	0,0000	0,0000	0,0008	0,0009
23	0,0006	0,0000	0,0000	0,0000	0,0009	0,0004
24	0,0004	0,0000	0,0000	0,0000	0,0007	0,0006
25	0,0004	0,0000	0,0000	0,0000	0,0006	0,0006

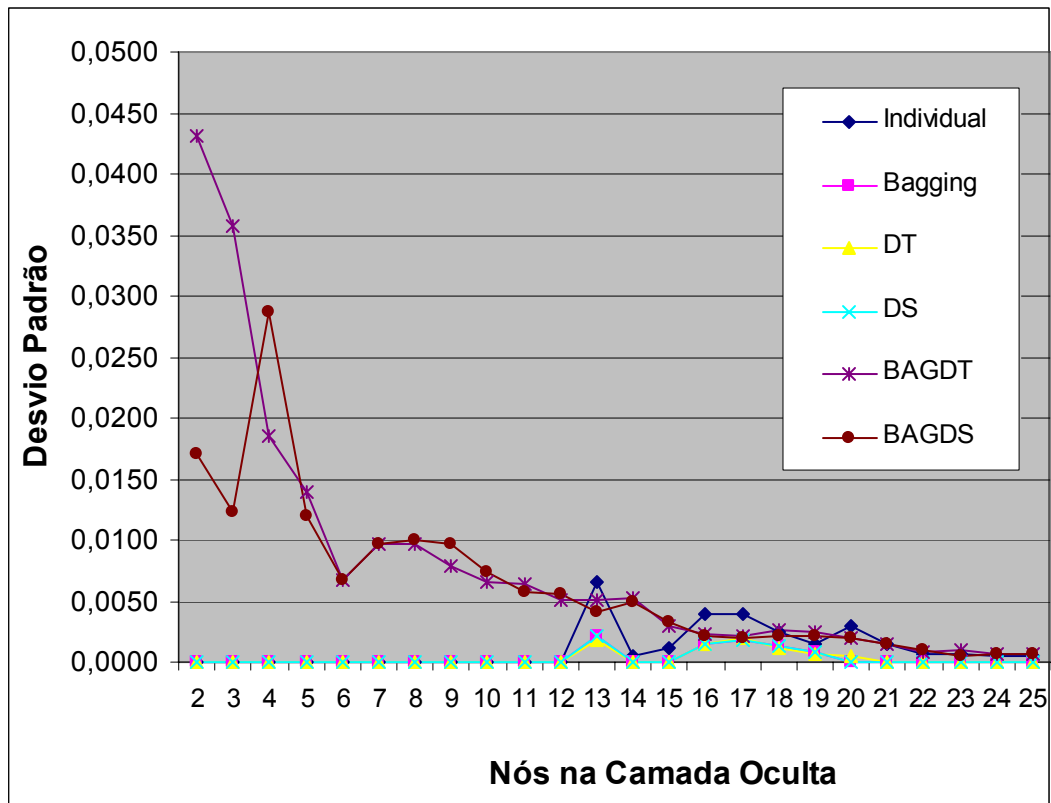
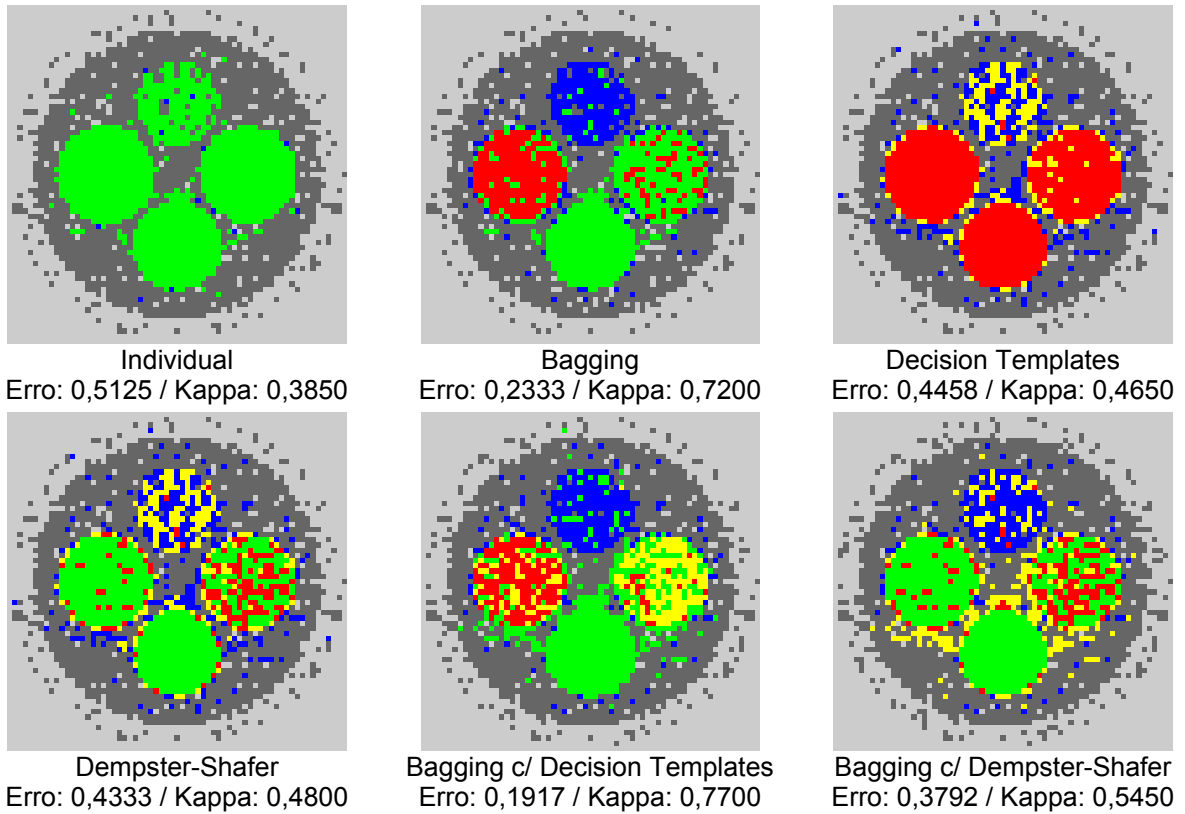
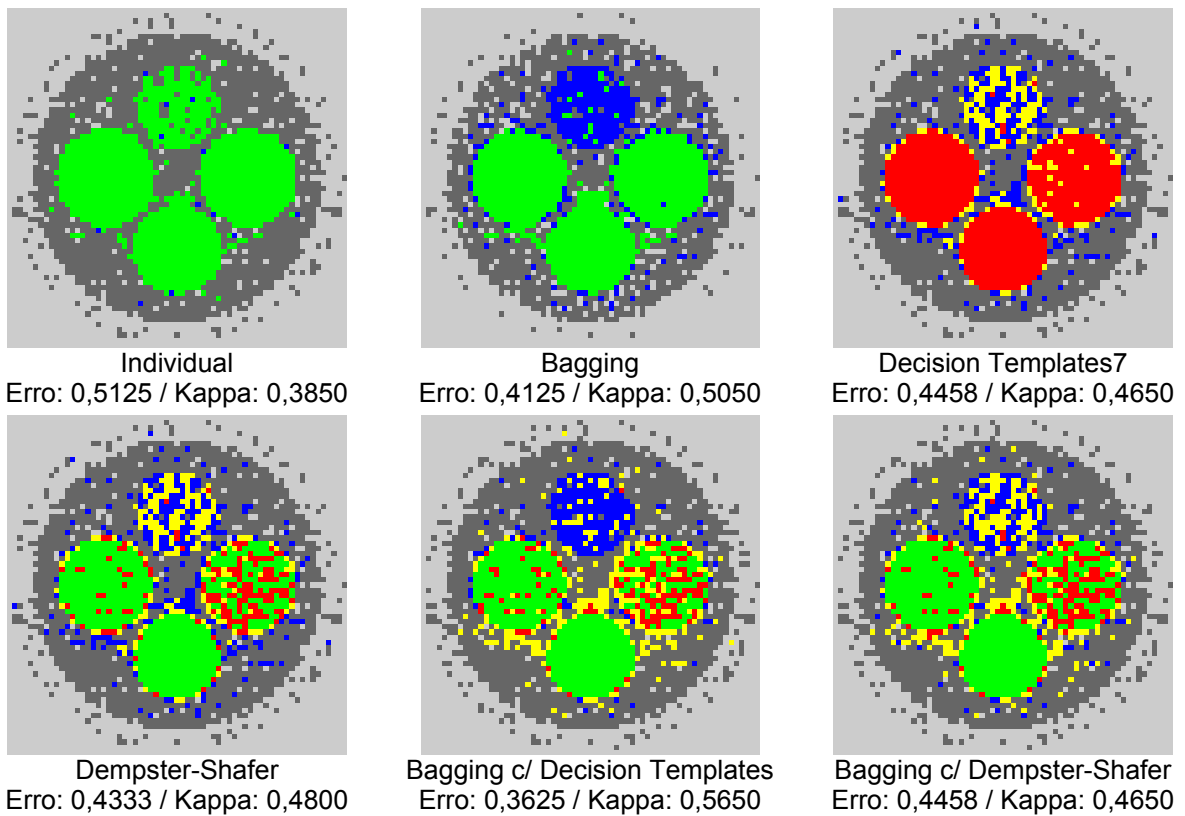


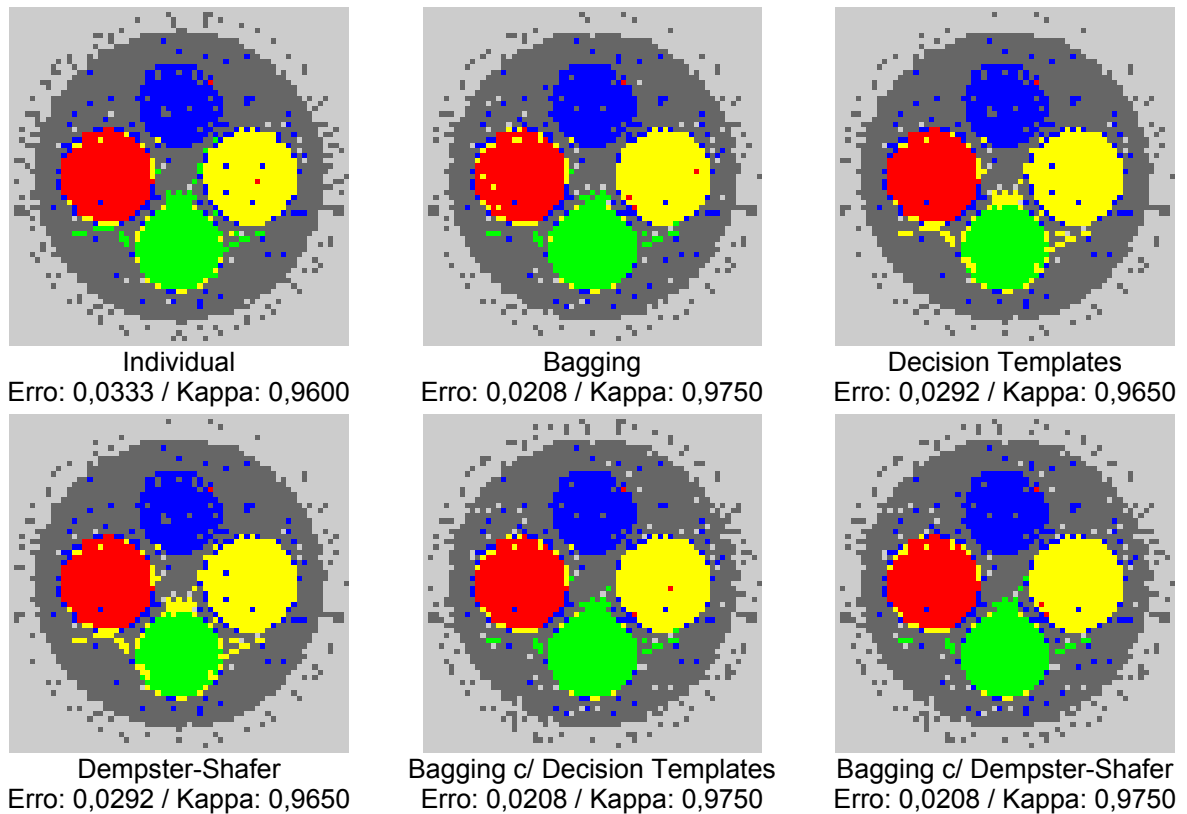
FIGURA 6.12 – Variação do Desvio Padrão para Redes de Função de Base Radial



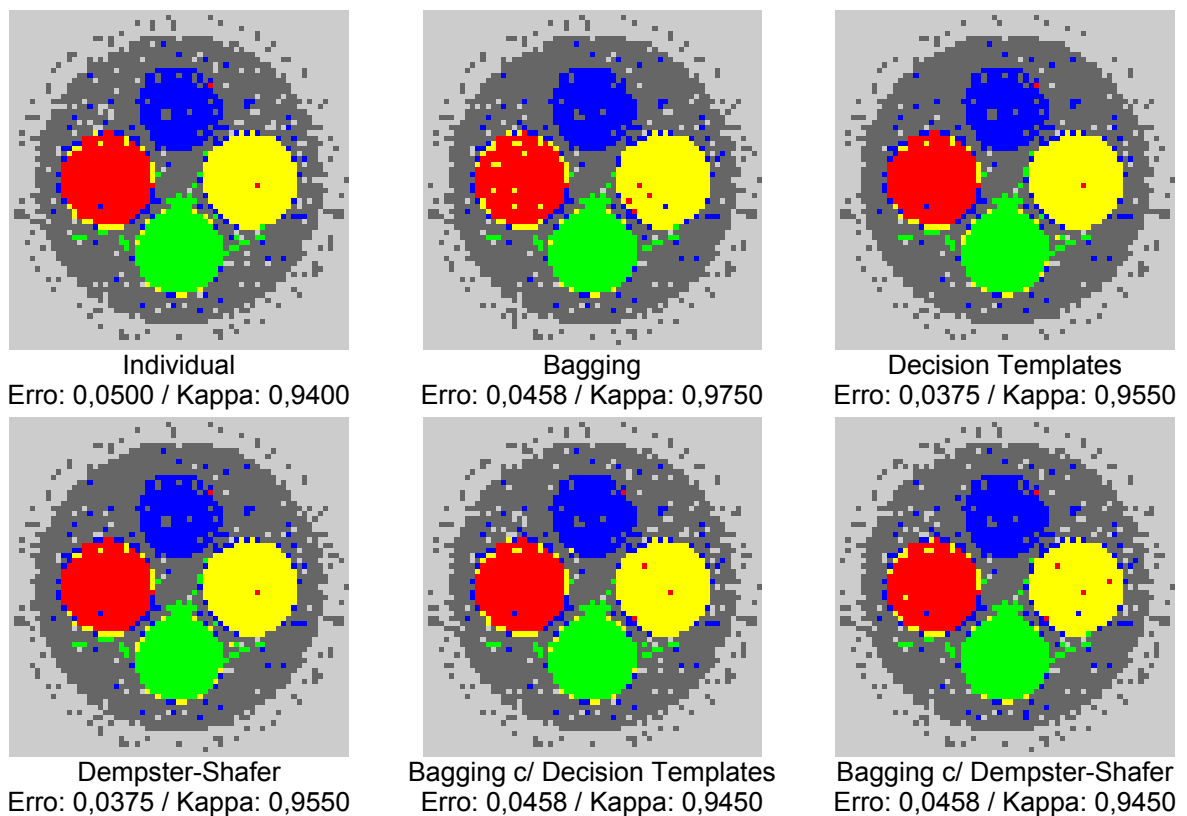
**FIGURA 6.13** – Imagens Temáticas para Redes de Função de Base Radial com 2 nós na camada oculta (melhores casos)



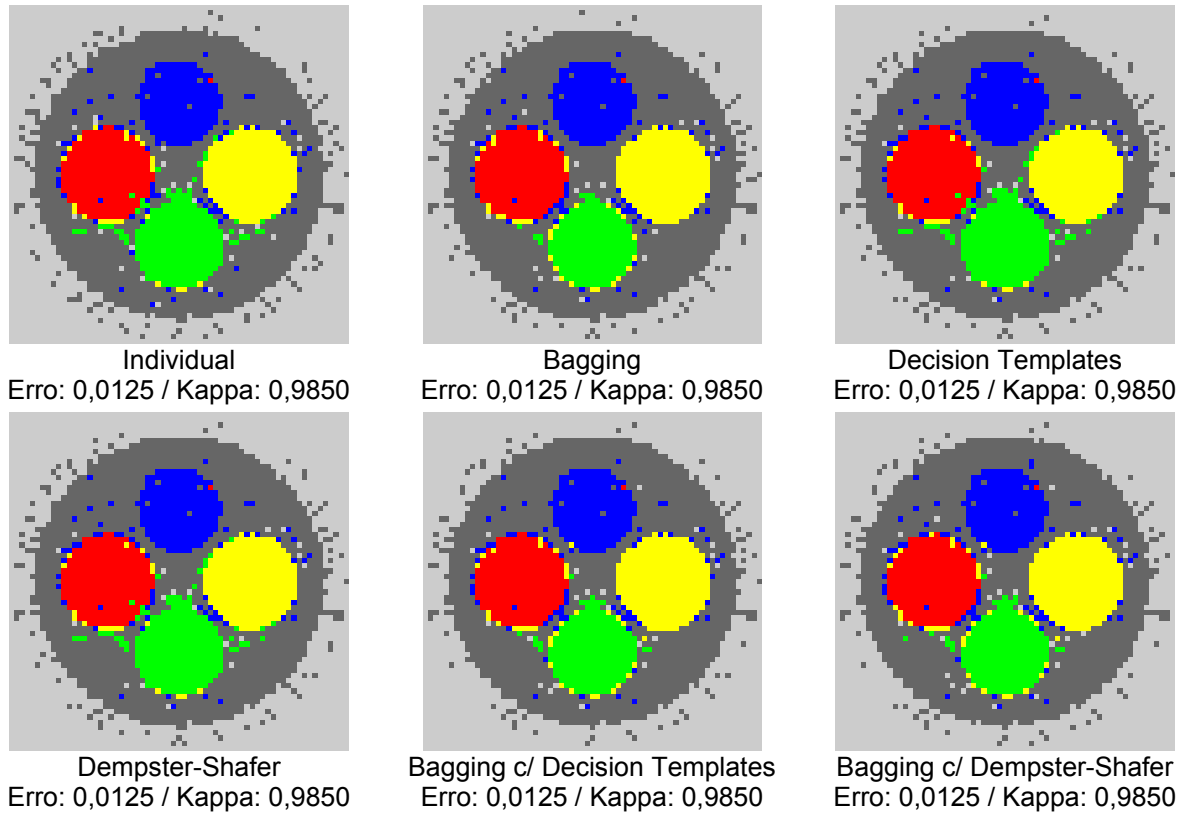
**FIGURA 6.14** – Imagens Temáticas para Redes de Função de Base Radial com 2 nós na camada oculta (piores casos)



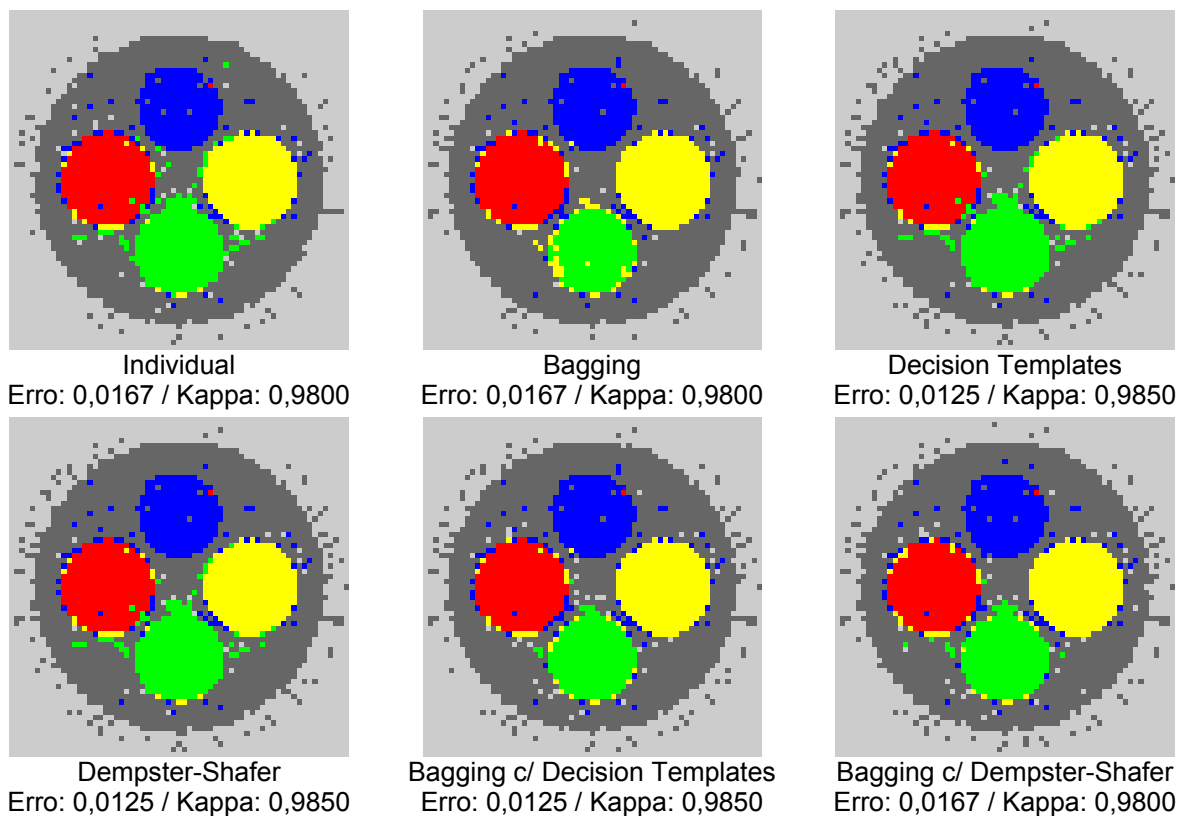
**FIGURA 6.15** – Imagens Temáticas para Redes de Função de Base Radial com 13 nós na camada oculta (melhores casos)



**FIGURA 6.16** – Imagens Temáticas para Redes de Função de Base Radial com 13 nós na camada oculta (piores casos)



**FIGURA 6.17** – Imagens Temáticas para Redes de Função de Base Radial com 25 nós na camada oculta (melhores casos)



**FIGURA 6.18** – Imagens Temáticas para Redes de Função de Base Radial com 25 nós na camada oculta (piores casos)

## 7. DISCUSSÃO

---

A partir dos resultados obtidos e apresentados na seção anterior foi possível fazer algumas análises e comparações e assim avaliar o desempenho e as diferenças entre os classificadores e combinadores testados, quando aplicados a imagens tomográficas multiespectrais ruidosas.

Uma característica das redes neurais do tipo Perceptron de Múltiplas Camadas, notada logo nos primeiros testes, foi seu desempenho depender bastante da configuração inicial dos parâmetros livres da rede (pesos sinápticos e *bias*). Treinando a mesma rede com as mesmas amostras várias vezes foi possível observar que os resultados variavam bastante e este foi o motivo de se optar por executar cada teste 100 vezes, de modo que fosse possível obter uma estatística mais confiável. Entretanto, este comportamento não se aplica as Redes de Função de Base Radial, que tendem a ser mais estáveis e apresentar resultados semelhantes quando treinadas e testadas com as mesmas amostras.

Observando as tabelas III e IV e as figuras 6.1 e 6.2 é possível observar que todas as técnicas de combinação melhoraram significativamente o desempenho dos classificadores Perceptron de Múltiplas Camadas, pois os erros estimados diminuíram e os coeficientes Kappa aumentaram. O combinador Decision Templates apresentou os melhores resultados, seguido por Dempster-Shafer, Bagging c/ Decision Templates e Bagging com Dempster-Shafer, nesta ordem e com pouca diferença entre os quatro. A técnica de Bagging apresentou o pior desempenho entre os combinadores, principalmente quando foram utilizados poucos neurônios na camada oculta.

A tabela V e a figura 6.3 indicam que o desvio padrão dentro das 100 repetições de cada experimento diminuiu significativamente com a aplicação de técnicas de combinação de classificadores, mostrando que todas as técnicas são eficientes para estabilizar classificadores Perceptron de Múltiplas Camadas. Também neste caso o combinador Decision Templates

apresentou o melhor desempenho, seguido por Bagging c/ Decision Templates, Dempster-Shafer e Bagging c/ Dempster-Shafer.

As imagens temáticas mostradas nas figuras 6.4 a 6.9 também evidenciam a estabilidade obtida com os combinadores no Perceptron de Múltiplas Camadas, pois as imagens para o pior e melhor caso são bem diferentes quando se utiliza apenas um classificador, mas tendem a ficar mais parecidas com o uso de combinadores.

Nos resultados para as Redes de Função de Base Radial, nas tabelas VI e VII e figuras 6.10 e 6.11, é possível observar que a aplicação de técnicas de combinação melhora a classificação na maioria dos casos, porém essa melhoria não é tão grande quanto no Perceptron de Múltiplas Camadas. A técnica de Bagging apresentou os melhores resultados para as Redes de Função de Base Radial, com o combinador Bagging c/ Decision Templates sendo o segundo colocado apresentando resultados bem próximos ao primeiro. Em seguida vem o combinador Bagging c/ Dempster-Shafer c/ desempenho inferior aos dois primeiros, mas ainda bastante superior a Decision Templates e Dempster-Shafer, que foram os menos eficientes.

A tabela VIII e a figura 6.12 mostram que as Redes de Função de Base Radial são bastante estáveis nas 100 repetições de cada experimento quando comparadas ao Perceptron de Múltiplas Camadas, sendo que em vários casos o desvio padrão era zero, o que não foi alterado com o uso dos combinadores Bagging, Decision Templates e Dempster-Shafer. O uso dos combinadores Bagging c/ Decision Templates e Bagging c/ Dempster-Shafer causaram uma leve instabilidade nas Redes de Função de Base Radial.

As imagens temáticas mostradas nas figuras de 6.13 a 6.20 também evidenciam a maior estabilidade dos classificadores Redes de Função de Base Radial, pois as imagens para melhor e pior caso são bem parecidas e em alguns casos são idênticas (casos em que os 100 classificadores treinados produziram um desvio padrão igual a 0).

## 8. CONCLUSÕES

---

Analisando os resultados é possível concluir que o uso de combinação de classificadores produziu classificadores mais estáveis, além de melhorar significativamente os resultados medidos com o Erro Estimado pelo método Hold-Out e o Coeficiente Kappa, principalmente nas redes com poucos neurônios na camada oculta.

Enquanto os combinadores de Decision Templates e Dempster-Shafer mostraram melhor desempenho que a técnica de Bagging no Perceptron de Múltiplas Camadas, o contrário ocorreu nos testes com Redes de Função de Base Radial. Nelas o Bagging teve um melhor desempenho, enquanto Decision Templates e Dempster-Shafer não se saíram tão bem. Isto pode ser explicado pela maior estabilidade das Redes de Função de Base Radial quando comparadas ao Perceptron de Múltiplas Camadas, o que faz com que os classificadores fiquem muito parecidos e conseqüentemente a combinação seja menos vantajosa.

O combinador Decision Templates mostrou ser o mais indicado para combinar o Perceptron de Múltiplas Camadas, apresentando o melhor desempenho e mostrando que as diferenças nas redes causadas pela inicialização aleatória dos parâmetros livres já são suficientes para produzir classificadores bastante diferentes e bons para combinação. Além disso, Decision Templates também foi o combinador que deixou o Perceptron mais estável.

Para as Redes de Função de Base Radial a técnica de Bagging mostrou ser o método mais indicado, pois foi o que apresentou o melhor desempenho. Bagging c/ Decision Templates também apresentou um bom desempenho, entretanto foi um dos métodos que trouxe uma leve instabilidade para as Redes de Função de Base Radial, de modo que seu uso deve ser feito com cautela. Decision Templates e Dempster-Shafer foram os combinadores menos eficientes, e isto provavelmente se deve ao fato das Redes de Função de Base Radial serem mais estáveis, produzindo classificadores muito parecidos, o que os torna menos



indicados para combinação. A técnica de Bagging levou vantagem por utilizar conjuntos de amostras do tipo bootstrap, produzindo classificadores diferentes para a combinação, e conseqüentemente melhores resultados.

O uso de combinação de classificadores para combinar redes neurais se mostrou bastante útil para identificar materiais em imagens de ciências dos solos. Outras áreas também podem ser exploradas utilizando esta mesma abordagem, pois as regras de combinação normalmente melhoraram a performance e aumentam a capacidade de generalização dos classificadores.

### **8.1 Trabalhos Futuros**

Diversos trabalhos podem ser desenvolvidos como extensão deste, na área de combinação de classificadores:

- Fazer uma pré-seleção dos classificadores mais relevantes para a combinação;
- Combinar redes neurais de tipos diferentes (por exemplo: Perceptron de Múltiplas Camadas com Redes de Função de Base Radial);
- Utilizar outros combinadores com a técnica de Bagging, como o Produto, Máximo e Mínimo, por exemplo;
- Combinar a saída dos combinadores, utilizando votação por maioria ou qualquer outra técnica;
- Utilizar redes neurais para combinar resultados de classificação.

## REFERÊNCIAS BIBLIOGRÁFICAS

---

- AHMADZADEH, M. R.; PETRON M.; SASIKALA, K. R. **The Dempster-Shafer Combination Rule as a Tool to Classifier Combination**. Geoscience and Remote Sensing Symposium. Proc. IGARSS 2000 IEEE International, p. 2429-2431, 2000.
- BISHOP, C. M., **Neural Networks for Pattern Recognition**, New York: Oxford, 1995.
- BREIMAN, L. **Bagging Predictors**. Machine Learning. V. 26(2) p.123-140, 1996
- CARLETTA, J. **Assessing Agreement on Classification Tasks: the Kappa Statistic**. Computational Linguistics. V. 22(2) p. 249- 254, 1996
- CRUVINEL, P. E.; CESAREO, R.; MASCARENHAS, S. **X and  $\gamma$  - Rays Computerized Minitomograph Scanner for Soil Science**. IEEE Transactions on Instrumentation and Measurements. v. 39, n. 5, p. 745-750, 1990.
- DUDA, R. O.; HART, P. E., STORK, D. G. **Pattern Classification**, 2.ed. New York: Wiley, 2000.
- DUIN, R. P. W; JUSZCZAK, P.; PACLIK, P.; PERALSKA, D.; RIDDER, D.; TAX, D. M. J. **PRTools4: A Matlab Toolbox For Pattern Recognition**. Delft University of Technology: Delft, 51p., 2004.
- FUKUNAGA, K. **Introduction to Statistical Pattern Recognition**, 2. ed. New York: Academic Press, 1990.
- GONZALEZ, R. F.; WOODS, R. E. **Processamento de Imagens Digitais**. São Paulo: Edgar Blücher LTDA, 2000.
- HAYKIN S. **Redes Neurais – Princípios e Prática**. 2 ed. Porto Alegre: Bookman, 2001.
- HOMEM, M. R. P.; MASCARENHAS, N. D. A.; CRUVINEL, P. E. **The Linear Attenuation Coefficients as Features of Multiple Energy CT Image Classification**. Nuclear Instruments and Methods in Physics Research. V. 452, p. 351-360, 2000.
- HOMEM, M. R. P. **Técnicas de Reconhecimento de Padrões Aplicadas a Imagens Tomográficas Adquiridas com Múltiplas Energias**. Dissertação de mestrado, DC-UFSCar: São Carlos, 1998, 200p.
- HOUNSFIELD, G. N. Computerized Transverse Axial Scanning (Tomography). Part I: Description of System. Part II: Clinical Applications. British Journal of Radiology. v. 46, p. 1016-1022, 1973.
- JAIN, A. K.; MAO, J.; MOHIUDDIN, K. M. **Artificial Neural Networks: A Tutorial**. IEEE Computer v. 29, p. 31-44, 1996.
- JAIN, A. K.; DUIN, R. P. W.; MAO, J. **Statistical Pattern Recognition: A Review**. IEEE Transactions on Pattern Analysis and Machine Intelligence. v. 22, n. 1, p. 4-37, 2000.

KAK, A. C.; SLANEY, M. **Principles of Computerized Tomographic Imaging**. New York: Society of Industrial and Applied Mathematics, 1988.

KOHAVI, R. **A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection**. Proc. of the 14th Int. Joint Conf. on A.I., Vol. 2, Canada, p. 1137 –1143, 1995.

KUNCHEVA, L. I. **Combining Pattern Classifiers : Methods and Algorithms**. Hoboken, NJ :Wiley-Interscience, 2004.

MINSKY, M.; PAPERT, Seymour. **Perceptrons: an Introduction to Computational geometry**. Cambridge: The MIT Press, 1969.

PARKER, J. A. **Image Reconstruction in Radiology**, New York: CRC Press, 1990.

PONTI JR., M.P.; MASCARENHAS, N.D.A, **Material Analysis on Noisy Multispectral Images Using Classifier Combination**, Proc. IEEE Southwest Symposium on Image Analysis and Interpretation, IEEE Computer Society Press, Lake Tahoe, Nevada, p. 28-30, 2004.

PONTI JR, M. P. **Combinação de Múltiplos Classificadores para Identificação de Materiais em Imagens Ruidosas**. Dissertação de mestrado, DC-UFSCar: São Carlos, 2004, 76p.

PRAMPERO, P.; CARVALHO, A. **Recognition of Vehicles Using Combination of Classifiers**. Proceedings of the IEEE World Congress on Computational Intelligence, WCCI'98, Anchorage, USA, 1998.

XU, L.; KRZYZAK, A.; SUEN, C.Y. **Methods of Combining Multiple Classifier and Their Applications to Handwriting Recognition**. IEEE Transactions on Systems, Man and Cybernetics. v. 22, n. 3, p. 418-435, 1992.