

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
CENTRO ESTADUAL DE PESQUISAS EM SENSORIAMENTO REMOTO E
METEOROLOGIA
PROGRAMA DE PÓS-GRADUAÇÃO EM SENSORIAMENTO REMOTO

NEIDE PIZZOLATO ANGELO

**UMA ABORDAGEM PARA A DETECÇÃO DE MUDANÇAS EM IMAGENS
MULTITEMPORAIS DE SENSORIAMENTO REMOTO EMPREGANDO *SUPPORT
VECTOR MACHINES* COM UMA NOVA MÉTRICA DE PERTINÊNCIA**

PORTO ALEGRE

2014

NEIDE PIZZOLATO ANGELO

**UMA ABORDAGEM PARA A DETECÇÃO DE MUDANÇAS EM IMAGENS
MULTITEMPORAIS DE SENSORIAMENTO REMOTO EMPREGANDO *SUPPORT
VECTOR MACHINES* COM UMA NOVA MÉTRICA DE PERTINÊNCIA**

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Sensoriamento Remoto como requisito parcial para a obtenção do título de Doutor em Sensoriamento Remoto.

Área de concentração: Sensoriamento Remoto e Geoprocessamento.

Orientador: Prof. Dr. Vitor Haertel

PORTO ALEGRE

2014

CIP - Catalogação na Publicação

Pizzolato Angelo, Neide

UMA ABORDAGEM PARA A DETECÇÃO DE MUDANÇAS EM
IMAGENS MULTITEMPORAIS DE SENSORIAMENTO REMOTO
EMPREGANDO SUPPORT VECTOR MACHINES COM UMA NOVA
MÉTRICA DE PERTINÊNCIA / Neide Pizzolato Angelo. --
2014.

124 f.

Orientador: Vitor Haertel.

Tese (Doutorado) -- Universidade Federal do Rio
Grande do Sul, Centro Estadual de Pesquisas em
Sensoriamento Remoto e Meteorologia, Programa de Pós-
Graduação em Sensoriamento Remoto, Porto Alegre, BR-
RS, 2014.

1. Sensoriamento Remoto. 2. Detecção de Mudanças.
3. Support Vector Machines. 4. Métodos Baseados em
Kernels. I. Haertel, Vitor, orient. II. Título.

**UMA ABORDAGEM PARA A DETECÇÃO DE MUDANÇAS EM IMAGENS
MULTITEMPORAIS DE SENSORIAMENTO REMOTO EMPREGANDO *SUPPORT
VECTOR MACHINES* COM UMA NOVA MÉTRICA DE PERTINÊNCIA**

Tese de Doutorado aprovada para a obtenção do título de Doutor no Programa de Pós-Graduação em Sensoriamento Remoto da Universidade Federal do Rio Grande do Sul, pela banca examinadora formada por:

Prof. Dr. Daniel C. Zanotta, IFRS

Prof. Dr. Claudio R. Jung, UFRGS

Profa. Dra. Dejanira Luderitz Saldanha, UFRGS

Prof. Dr. Vitor Haertel, UFRGS (Orientador)

Dedico esta tese ao meu amigo, mentor e orientador, Prof. Vitor Haertel, in memoriam, à minha amada esposa Dinara e ao meu filho William. Aos meus pais, Ernesto e Nelly, meus sogros Ruy e Ivalina.

AGRADECIMENTOS

Gostaria de agradecer e dedicar este trabalho a todos que, de alguma forma, contribuíram para sua realização.

Primeiramente, a DEUS, por me permitir a existência e por ter me guiado até aqui de forma justa e perfeita.

Ao meu orientador, professor Vitor Haertel, pela amizade, competência e sabedoria com que me conduziu na realização deste trabalho. Certamente o seu exemplo como professor, orientador e cientista servirão de modelo em toda a minha carreira acadêmica.

À minha amiga e colega, Rute Ferreira, pelo companheirismo e pelo auxílio que sempre me ofereceu.

As minhas colegas de trabalho Patrícia Fantinel e Daniela Hoffmamm que sempre me apoiaram durante a realização deste trabalho.

Aos professores e funcionários do Programa de Pós-Graduação em Sensoriamento Remoto e ao Centro Estadual de Pesquisa em Sensoriamento Remoto e Meteorologia da Universidade Federal do Rio Grande do Sul, pela constante disponibilidade em me auxiliar.

Aos meus pais, Ernesto e Nelly pelo seu amor incondicional, apoio e exemplo de vida.

Aos meus sogros, Ruy e Ivalina pelo seu imenso amor, carinho e apoio sempre dado desde que nos conhecemos.

À minha amada esposa Dinara e ao meu filho William, pelo seu amor nos momentos em que eu mais precisei.

À minha família: irmãos Flávio e Moacir e suas respectivas famílias; minhas cunhadas: Estela, Sandra e seu marido Marcelo, pelo apoio, carinho e incentivo que sempre transmitiram, o qual serviu de alento em vários momentos ao longo desta caminhada.

RESUMO

Esta tese investiga uma abordagem não supervisionada para o problema da detecção de mudanças em imagens multiespectrais e multitemporais de sensoriamento remoto empregando *Support Vector Machines (SVM)* com o uso dos kernels polinomial e RBF e de uma nova métrica de pertinência de pixels. A proposta metodológica está baseada na diferença das imagens-fração produzidas para cada data. Em imagens de cenas naturais essa diferença nas frações de solo e vegetação tendem a apresentar uma distribuição simétrica próxima à origem. Essa característica pode ser usada para modelar as distribuições normais multivariadas das classes mudança e não-mudança. O algoritmo Expectation-Maximization (EM) é implementado com a finalidade de estimar os parâmetros (vetor de médias, matriz de covariância e probabilidade *a priori*) associados a essas duas distribuições. A seguir, amostras aleatórias e normalmente distribuídas são extraídas dessas distribuições e rotuladas segundo sua pertinência em uma das classes. Essas amostras são então usadas no treinamento do classificador SVM. A partir desta classificação é estimada uma nova métrica de pertinência de pixels. A metodologia proposta realiza testes com o uso de conjuntos de dados multitemporais de imagens multiespectrais *Landsat-TM* que cobrem a mesma cena em duas datas diferentes. A métrica de pertinência proposta é validada através de amostras de teste controladas obtidas a partir da técnica *Change Vector Analysis*, além disso, os resultados de pertinência obtidos para a imagem original com essa nova métrica são comparados aos resultados de pertinência obtidos para a mesma imagem pela métrica proposta em (Zanotta, 2010). Baseado nos resultados apresentados neste trabalho que mostram que a métrica para determinação de pertinência é válida e também apresenta resultados compatíveis com outra técnica de pertinência publicada na literatura e considerando que para obter esses resultados utilizou-se poucas amostras de treinamento, espera-se que essa métrica deva apresentar melhores resultados que os que seriam apresentados com classificadores paramétricos quando aplicado a imagens multitemporais e hiperespectrais.

Palavras-chave: detecção de mudanças, métodos baseados em *kernel*, imagens de fração, algoritmo EM, métrica de pertinência.

ABSTRACT

This thesis investigates a unsupervised approach to the problem of change detection in multispectral and multitemporal remote sensing images using Support Vector Machines (SVM) with the use of polynomial and RBF kernels and a new metric of pertinence of pixels. The methodology is based on the difference-fraction images produced for each date. In images of natural scenes. This difference in the fractions of bare soil and vegetation tend to have a symmetrical distribution close to the origin. This feature can be used to model the multivariate normal distributions of the classes change and no-change. The Expectation-Maximization algorithm (EM) is implemented in order to estimate the parameters (mean vector, covariance matrix and a priori probability) associated with these two distributions. Then random and normally distributed samples are extracted from these distributions and labeled according to their pertinence to the classes. These samples are then used in the training of SVM classifier. From this classification is estimated a new metric of pertinence of pixel. The proposed methodology performs tests using multitemporal data sets of multispectral Landsat-TM images that cover the same scene at two different dates. The proposed metric of pertinence is validated via controlled test samples obtained from Change Vector Analysis technique. In addition, the results obtained at the original image with the new metric are compared to the results obtained at the same image applying the pertinence metric proposed in (Zanotta, 2010). Based on the results presented here showing that the metric of pertinence is valid, and also provides results consistent with other published in the relevant technical literature, and considering that to obtain these results was used a few training samples, it is expected that the metric proposed should present better results than those that would be presented with parametric classifiers when applied to multitemporal and hyperspectral images.

Keywords: change detection, *kernel* methods, fraction-images, EM algorithm, metric of pertinence.

LISTA DE FIGURAS

Figura 1	Fenômeno de Hughes. Cada curva representa a acurácia obtida no processo de classificação, em função da dimensionalidade dos dados e do número de amostras de treinamento empregadas.....	21
Figura 2	Função densidade dos contadores digitais dos <i>pixels</i> da imagem de diferença com a definição de um limiar entre as classes mudança e não-mudança.....	26
Figura 3	Exemplo de representação de mudança em um <i>pixel</i> usado <i>CVA</i>	27
Figura 4	Separação entre duas classes linearmente separáveis.....	30
Figura 5	Superfície de separação linear ótima utilizando <i>SVM</i> de margem máxima.....	32
Figura 6	Determinação do hiperplano de Separação de classes com margem máxima com uso de variáveis de folga.....	33
Figura 7	Determinação da margem máxima m^* entre dois hiperplanos de suporte.....	36
Figura 8	Hiperplano ótimo separando os dados com a máxima margem.....	37
Figura 9	Hiperplano separando os dados com a máxima margem.....	41
Figura 10	(a) Duas classes não linearmente separáveis em duas dimensões. (b) mapeamento dos dados originais para um espaço de maior dimensão que o espaço original, onde é possível separar as classes de forma linear através de um hiperplano ótimo e hiperplanos contendo vetores de suporte (<i>support vectors</i>). (c) remapeamento dos hiperplanos de separação para o espaço original marcando a separação não linear das classes.....	44
Figura 11	Mapeamento de um conjunto de dados não linearmente separáveis.....	45
Figura 12	Esquema geral do pré-processamento: as imagens X_1 e X_2 são registradas, e após, fracionadas nas suas componentes pelo MLME, gerando as imagens X'_1 e X'_2 registradas e fracionadas.....	54
Figura 13	Representação do Modelo Linear de Mistura Espectral (MLME): (a) descrição real das porções existentes na área delimitada pelo <i>pixel</i> (b) resposta espectral do sensor para a área correspondida pelo <i>pixel</i> e (c)	

	resultado do fracionamento pelo MLME nas componentes de vegetação, solo e sombra/água.....	55
Figura 14	Gráfico do espalhamento dos dados da imagem-diferença.....	58
Figura 15	Distribuições das classes mudança e não-mudança.....	59
Figura 16	a) Histograma de frequência de <i>pixels</i> da imagem diferença, b) perfil do Histograma de frequência dos <i>pixels</i> sobre o eixo diferença fração vegetação e c) perfil do Histograma de frequência dos <i>pixels</i> sobre o eixo diferença fração solo.....	60
Figura 17	Visualização da métrica de pertinência Delta b	67
Figura 18	Correspondência entre os pontos mapeados distantes do hiperplano de separação e sua pertinência na classe onde está inserido.....	68
Figura 19	(a) Área de estudo; (b) Imagem de Data 1, composição em falsa cor 5(R), 4(G), 3(B) e (c) Imagem de Data 2, composição em falsa cor 5(R), 4(G), 3(B).....	73
Figura 20	Imagens-fração da imagem de 1991 para (a) vegetação (b) solo e (c) sombra. Variação contínua de preto (0%) a branco (100%).....	75
Figura 21	Imagem de diferença com dados reais. Composição em falsa cor 3 (R) 2(G) 3(B).....	76
Figura 22	Histogramas da imagem de diferenças. (a) Perfil da diferença na fração de vegetação. (b) Perfil da Diferença na fração de solo.....	76
Figura 23	Dispersão dos dados da imagem-diferença utilizada nos experimentos..	77
Figura 24	Apresentação das densidades de probabilidade conjuntas das distribuições resultantes das diferenças entre as frações de vegetação e solo para o primeiro experimento: (a) classe não-mudança (b) classe mudança (c) sobreposição das classes mudança e não-mudança.....	79
Figura 25	Gráfico de dispersão de uma amostra de treinamento com 400 <i>pixels</i> da classe mudança(verde) e 400 <i>pixels</i> da classe não-mudança (vermelho) ambas sobrepostas sobre a distribuição de <i>pixels</i> da imagem diferença (Figura 23) (azul).....	81
Figura 26	Imagem-módulo do vetor <i>CVA</i>	83
Figura 27	Imagem-módulo do vetor <i>CVA</i> com amostras da classe mudança em branco (módulo do vetor mudança maior que 0,3 e menor que 0,6) e amostras da classe não-mudança em preto (módulo do vetor mudança	

	menor que 0,1).....	84
Figura 28	Gráfico de dispersão da amostra de teste nas classes mudança (vermelho) e não-mudança (verde).....	84
Figura 29	Mapa de diferença de pertinência obtida (Zanotta, 2010).....	85
Figura 30	Classificação da amostra de testes no classificador <i>SVM</i> com 200 amostras de treinamento e utilizando <i>kernel</i> Polinomial em a) Pior pertinência usando <i>grau</i> =2 e em b) Melhor pertinência usando utilizando <i>grau</i> =5. Classe mudança representada em vermelho e não-mudança em verde.....	87
Figura 31	Classificação da amostra de testes no classificador <i>SVM</i> com 300 amostras de treinamento e utilizando <i>kernel</i> Polinomial em a) Pior pertinência usando <i>grau</i> =2 e em b) melhor pertinência usando utilizando <i>grau</i> =3. Classe mudança representada em vermelho e não-mudança em verde.....	88
Figura 32	Classificação da amostra de testes no classificador <i>SVM</i> com 400 amostras de treinamento e utilizando <i>kernel</i> Polinomial em a) Pior pertinência usando <i>grau</i> =2 e em b) melhor pertinência usando utilizando <i>grau</i> =3. Classe mudança representada em vermelho e não-mudança em verde.....	90
Figura 33	Classificação da amostra de testes no classificador <i>SVM</i> com 200 amostras de treinamento e utilizando o <i>kernel RBF</i> em a) Pior classificação por pertinência usando $\gamma=95$ e em b) melhor classificação por pertinência usando utilizando $\gamma=1$. Classe mudança representada em vermelho e não-mudança em verde.....	92
Figura 34	Classificação da amostra de testes no classificador <i>SVM</i> com 300 amostras de treinamento e utilizando o <i>kernel RBF</i> em a) Pior classificação por pertinência usando $\gamma=95$ e em b) melhor classificação por pertinência usando utilizando $\gamma=1$. Classe mudança representada em vermelho e não-mudança em verde.....	94
Figura 35	Classificação da amostra de testes no classificador <i>SVM</i> com 400 amostras de treinamento e utilizando o <i>kernel RBF</i> em a) Pior classificação por pertinência usando $\gamma=1$ e em b) melhor classificação por pertinência usando utilizando $\gamma=10$. Classe mudança representada	

	em vermelho e não-mudança em verde.....	95
Figura 36	Imagem Módulo do vetor CVA com origem transladada.....	98
Figura 37	Imagem Módulo CVA como a origem deslocada para a média da classe não-mudança. As amostras da classe mudança, em branco (módulo do vetor mudança centralizado maior que 0,3 e menor que 0,6) e as amostras da classe não-mudança em preto (módulo do vetor mudança centralizado menor que 0,1).....	99
Figura 38	Gráfico de dispersão da amostra de teste nas classes mudança (vermelho) e não-mudança (verde).....	100
Figura 39	Classificação da amostra de testes segundo a métrica Delta b usando o classificador SVM com 200, 300 e 400 amostras de treinamento e utilizando kernel Polinomial. a) Pior pertinência usando grau= 5 e 200 amostras de treinamento em b) melhor pertinência usando utilizando grau=2 e 200 amostras de treinamento c) Pior pertinência usando grau= 5 e 300 amostras de treinamento d) melhor pertinência usando utilizando grau=3 e 300 amostras de treinamento e) Pior pertinência usando grau= 2 e 400 amostras de treinamento f) melhor pertinência usando utilizando grau=5 e 400 amostras de treinamento. Em todos os gráficos a classe mudança é representada em vermelho e não-mudança em verde.....	102
Figura 40	Classificação da amostra de testes segundo a métrica Delta b usando o classificador SVM com 200, 300 e 400 amostras de treinamento e utilizando kernel RBF. a) Pior pertinência usando $\gamma=2$ e 200 amostras de treinamento em b) melhor pertinência usando utilizando $\gamma=5$ e 200 amostras de treinamento c) Pior pertinência usando $\gamma=2$ e 300 amostras de treinamento d) melhor pertinência usando utilizando $\gamma=5$ e 300 amostras de treinamento e) Pior pertinência usando $\gamma=2$ e 400 amostras de treinamento f) melhor pertinência usando utilizando $\gamma=5$ e 400 amostras de treinamento. Em todos os gráficos a classe mudança é representada em vermelho e não-mudança em verde.....	104
Figura 41	a) Mapa de Pertinência usando kernel polinomial com 200 amostras de treinamento e grau do kernel polinomial=2 b) Mapa de diferença de pertinência entre a metodologia proposta em (Zanotta, 2010) e	

	metodologia proposta neste trabalho.....	108
Figura 42	a) Mapa de Pertinência usando <i>kernel</i> polinomial com 300 amostras de treinamento e grau do <i>kernel</i> polinomial=2 b) Mapa de diferença de pertinência entre (Zanotta, 2010) e metodologia proposta.....	110
Figura 43	a) Mapa de Pertinência usando <i>kernel</i> polinomial com 400 amostras de treinamento e grau do <i>kernel</i> polinomial=4 b) Mapa de diferença de pertinência entre (Zanotta, 2010) e metodologia proposta.....	111
Figura 44	a) Mapa de Pertinência usando <i>kernel</i> RBF com 200 amostras de treinamento e $\gamma=80$ b) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta.....	112
Figura 45	a) Mapa de Pertinência usando <i>kernel</i> RBF 300 amostras de treinamento e $\gamma=90$ b) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta.....	114
Figura 46	a) Mapa de Pertinência usando <i>kernel</i> RBF com 400 amostras de treinamento e $\gamma=85$ b) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta. c) Mapa de Pertinência usando <i>kernel</i> RBF com 400 amostras de treinamento e $\gamma=100$ d) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta.....	115

LISTA DE TABELAS

Tabela 1	Avaliação da pertinência na amostra de teste usando kernel Polinomial com 200 amostras de treinamento para classe mudança.....	87
Tabela 2	Avaliação da pertinência na amostra de teste usando kernel Polinomial com 200 amostras de treinamento para classe não-mudança.....	88
Tabela 3	Avaliação da pertinência na amostra de teste usando kernel Polinomial com 300 amostras de treinamento para classe mudança.....	89
Tabela 4	Avaliação da pertinência na amostra de teste usando kernel Polinomial com 300 amostras de treinamento para classe não-mudança.....	89
Tabela 5	Avaliação da pertinência na amostra de teste usando kernel Polinomial com 400 amostras de treinamento para classe mudança.....	90
Tabela 6	Avaliação da pertinência na amostra de teste usando kernel Polinomial com 400 amostras de treinamento para classe não-mudança.....	91
Tabela 7	Avaliação da pertinência na amostra de teste usando kernel RBF com 200 amostras de treinamento para classe mudança.....	93
Tabela 8	Avaliação da pertinência na amostra de teste usando kernel RBF com 200 amostras de treinamento para classe não-mudança.....	93
Tabela 9	Avaliação da pertinência na amostra de teste usando kernel RBF com 300 amostras de treinamento para classe mudança	94
Tabela 10	Avaliação da pertinência na amostra de teste usando kernel RBF com 300 amostras de treinamento para classe não-mudança.....	95
Tabela 11	Avaliação da pertinência na amostra de teste usando kernel RBF com 400 amostras de treinamento para classe mudança.....	96
Tabela 12	Avaliação da pertinência na amostra de teste usando kernel RBF com 400 amostras de treinamento para classe não-mudança.....	96
Tabela 13	Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel Polinomial para classe mudança.....	102
Tabela 14	Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel Polinomial	

	para classe não-mudança.....	103
Tabela 15	Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel RBF para classe mudança.....	105
Tabela 16	Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel RBF para classe não-mudança.....	105
Tabela 17	Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel Polinomial e 200 amostras de treinamento.....	107
Tabela 18	Comparação entre as pertinências obtidas por (Zanotta, 2010) e as obtidas pela metodologia proposta usando kernel Polinomial e 300 amostras de treinamento.....	109
Tabela 19	Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel Polinomial e 400 amostras de treinamento.....	110
Tabela 20	Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel RBF e 200 amostras de treinamento.....	112
Tabela 21	Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel RBF e 300 amostras de treinamento.....	113
Tabela 22	Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel RBF e 400 amostras de treinamento.....	115

LISTA DE ABREVIATURAS E SIGLAS

MVG	Método da Máxima Verossimilhança Gaussiana
<i>SVM</i>	Support Vector Machines
<i>CVA</i>	Change Vector Analysis
EM	Expectation-Maximization
KKT	Karush–Kuhn–Tucker
RBF	Radial Basis Function
MLME	Modelo Linear de Mistura Espectral

LISTA DE SÍMBOLOS

\mathbf{x}	Vetor em um espaço dado
p	Número de bandas espectrais
\mathbf{w}	Vetor normal à superfície de decisão
$\mathbf{w} \cdot \mathbf{x} = \mathbf{w}^T \mathbf{x}$	Produto interno entre os vetores \mathbf{w} e \mathbf{x}
θ	Ângulo entre dois vetores
$G(\mathbf{x})$	Superfície de separação
$f(\mathbf{x})$	Função de decisão
m	Margem dos dois hiperplanos de suporte
α_i	Multiplicador de Lagrange
M	Número de amostras de treinamento
ξ_i	Variável de folga
$g(\mathbf{x})$	Função de mapeamento
m	Dimensão do espaço de entrada
l	Dimensão do espaço característico
$K(\mathbf{x}, \mathbf{y})$	Função de <i>kernel</i>
d	Parâmetro do <i>kernel</i> polinomial
γ	Parâmetro do <i>kernel</i> RBF
C	Constante de custo
b	Parâmetro de ajuste
M_1 e M_2	Médias das classes mudança e não-mudança
S_1 e S_2	Matriz de covariância das classes mudança e não-mudança
$P(\omega_1)$ e $P(\omega_2)$	Probabilidade <i>a priori</i> das classes mudança e não-mudança
λ	Autovalor
$\boldsymbol{\alpha}$	Vetor de multiplicadores de Lagrange
$P_m(\mathbf{x})$	Pertinência na classe mudança
$P_{nm}(\mathbf{x})$	Pertinência na classe não-mudança

SUMÁRIO

1	INTRODUÇÃO	20
2	REVISÃO BIBLIOGRÁFICA	25
2.1	Evolução do estudo de detecção de mudanças	25
2.2	Principais abordagens para detecção de mudanças	26
2.3	Classificadores SVM	29
2.3.1	<i>Aspectos matemáticos dos classificadores SVM</i>	34
2.4	<i>Kernel</i>	44
2.5	Estado da arte em detecção de mudanças usando SVM	49
2.6	Contribuição deste Trabalho	51
3	METODOLOGIA	53
3.1	Pré-processamento	53
3.1.1	<i>Modelo Linear de Mistura Espectral (MLME)</i>	55
3.2	Análise dos Dados	56
3.2.1	<i>Produção das Imagens de Diferença</i>	57
3.2.2	<i>Análise das Imagens de Diferença</i>	57
3.2.2.1	<i>O Algoritmo EM</i>	61
3.2.2.2	<i>Estimativa inicial dos parâmetros para uso no algoritmo EM</i>	63
3.2.3	<i>Seleção das amostras de treinamento</i>	64
3.2.4	<i>Classificador SVM</i>	65
3.2.5	<i>Determinação de métrica para estimativa de pertinência de pixels as classes</i>	65
3.3	Medidas de Avaliação da Acurácia da metodologia	69
4	TESTES E EXPERIMENTOS	71
4.1	Introdução	71
4.2	Descrição das Imagens	72
4.3	Produção das Imagens de Fração	74
4.4	Produção da Imagem de Diferença	75
4.5	Estimativa dos parâmetros das distribuições das classes mudança e não-mudança	77
4.6	Geração das amostras de treinamento	80
4.7	Geração das amostras de teste	81

4.8	Experimentos utilizando o classificador SVM.....	85
4.8.1	<i>Coeficiente e parâmetros utilizados no classificador SVM.....</i>	86
4.9	Experimentos utilizando o classificador SVM com Kernels Polinomial e RBF para amostra de teste.....	86
4.9.1	<i>Experimentos com kernel Polinomial aplicado a amostra de teste</i>	86
4.9.1.1	<i>Experimento com kernel polinomial utilizando 200 amostras de treinamento.....</i>	86
4.9.1.2	Experimento com <i>kernel</i> polinomial utilizando 300 amostras de treinamento.....	88
4.9.1.3	Experimento com <i>kernel</i> polinomial utilizando 400 amostras de treinamento.....	89
4.9.2	<i>Experimentos com kernel RBF aplicado a amostra de teste.....</i>	91
4.9.2.1	Experimento com <i>kernel</i> RBF utilizando 200 amostras de treinamento.....	92
4.9.2.2	Experimento com <i>kernel</i> RBF utilizando 300 amostras de treinamento.....	93
4.9.2.3	Experimento com <i>kernel</i> RBF utilizando 400 amostras de treinamento.....	95
4.10	Experimentos utilizando o classificador SVM para amostra de teste geradas a partir da métrica do módulo do vetor mudança deslocado...	100
4.10.1	<i>Experimentos utilizando o classificador SVM com Kernels Polinomial e RBF para amostra de teste.....</i>	100
4.10.1.1	<i>Experimentos com kernel Polinomial aplicado a amostra de teste gerada pelo Módulo CVA com origem transladada.....</i>	101
4.10.1.2	Experimentos com <i>kernel</i> RBF aplicado a amostra de teste gerada a partir do Módulo CVA com origem transladada.....	103
4.11	Experimentos utilizando o classificador SVM com Kernels Polinomial e RBF para imagem real.....	106
4.11.1	<i>Experimento com kernel polinomial.....</i>	107
4.11.1.1	<i>Experimento com kernel polinomial utilizando 200 amostras de treinamento.....</i>	107
4.11.1.2	<i>Experimento com kernel polinomial utilizando 300 amostras de treinamento.....</i>	108
4.11.1.3	<i>Experimento com kernel polinomial utilizando 400 amostras de treinamento.....</i>	110
4.11.2	<i>Experimento com kernel RBF.....</i>	111

4.11.2.1	<i>Experimento com kernel RBF com 200 amostras de treinamento.....</i>	112
4.11.2.2	<i>Experimento com kernel RBF com 300 amostras de treinamento.....</i>	113
4.11.2.3	Experimento com kernel RBF com 400 amostras de treinamento.....	114
5	CONCLUSÕES.....	117
5.1	Sugestões para trabalhos futuros.....	119
	REFERÊNCIAS.....	119

1 INTRODUÇÃO

O estudo de reconhecimento de padrões desenvolveu-se significativamente na década de 1960. Seu estudo foi interdisciplinar, abrangendo desenvolvimentos nas áreas de estatística, engenharia, inteligência artificial, ciência da computação, entre outros. Suas aplicações vão desde o reconhecimento automático de caracteres, diagnóstico médico e mineração de dados até o desenvolvimento de máquinas com performance semelhante ao cérebro humano. Recentemente, novos desafios têm surgido. Tanto no desenvolvimento de novas metodologias, como em aplicações dos recursos computacionais disponíveis, tais como: processadores mais velozes e aumento na capacidade de memória dos computadores. Esses desenvolvimentos, baseados em recursos computacionais, incluem, por exemplo, Classificadores baseados em aprendizagem de máquina e *kernels* (WEBB, 2002; ANGELO, 2000).

Em geral, existem duas abordagens para o desenvolvimento de classificadores em reconhecimento de padrões: abordagem paramétrica e não paramétrica. Na abordagem paramétrica a função de decisão do classificador (função matemática que separa as classes de dados envolvidas no problema) é definida através de uma função densidade de probabilidade caracterizada por um número relativamente pequeno de parâmetros. Na abordagem não paramétrica nenhuma suposição é feita com relação a parâmetros (BOW, 1992). Deve ser observado, neste momento, que funções de decisão do classificador do tipo parametrizado (como é o caso dos coeficientes em um polinômio), são, por vezes, empregadas em classificadores não-paramétricos. Nesses casos, entretanto, não é empregada nenhuma forma de distribuição (BOW, 1992).

Na abordagem paramétrica, o método da Máxima Verossimilhança Gaussiana (MVG), que é um caso particular do classificador de *Bayes*, é muito utilizado em sensoriamento remoto. Na abordagem não-paramétrica têm-se, comumente, os classificadores de redes neurais, árvores de decisão e os de *Support Vector Machines (SVM)*. Para o treinamento de ambos os classificadores são utilizadas amostras de cada classe envolvida na cena.

Em particular, o classificador *SVM* é um classificador baseado em aprendizagem de máquina, que lida bem com o problema da dimensionalidade, como será visto com mais detalhes a seguir.

Em função da tecnologia disponível atualmente, pode-se utilizar imagens com um número elevado de bandas e, conseqüentemente, aumentar o poder discriminante dos classificadores. Ocorre, entretanto, que o número elevado de bandas espectrais em si não

assegura uma classificação mais acurada. O processo de classificação de dados em alta dimensionalidade emprega um número muito grande de variáveis, o que resulta também em um número grande de parâmetros a serem estimados quando se utiliza classificadores paramétricos, como, por exemplo, MVG. Estimativas confiáveis de um número crescente de parâmetros exigem, por sua vez, um número também crescente de amostras de treinamento. Contudo, essa quantidade de amostras dificilmente está disponível nas situações corriqueiras de classificação de imagens em sensoriamento remoto. Assim, esse número insuficiente de amostras de treinamento resulta no chamado fenômeno de *Hughes*, o qual consiste na gradual degradação da exatidão dos resultados, na medida em que a dimensionalidade dos dados é aumentada sem um aumento proporcional no número de amostras de treinamento (LANDGREBE, 1997), conforme é ilustrado na figura a seguir:

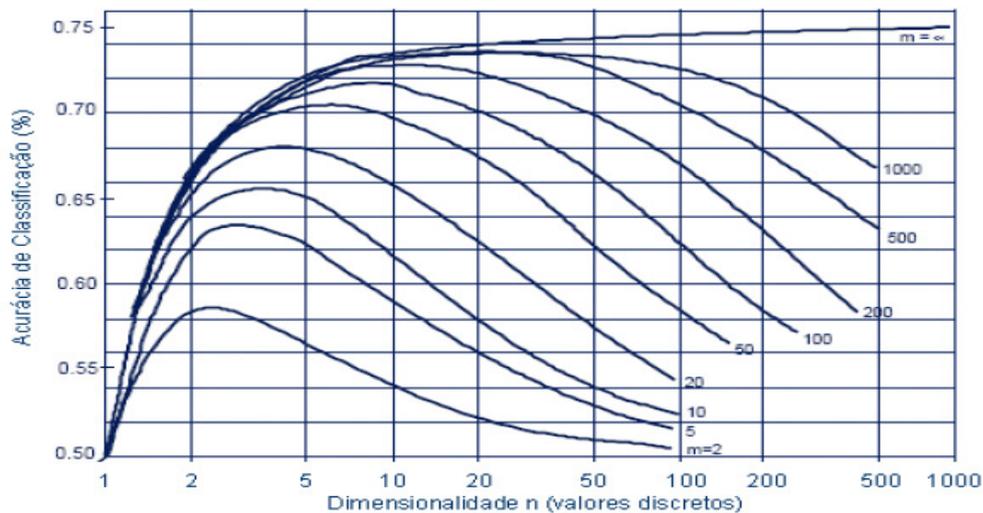


Figura 1 – Fenômeno de *Hughes*. Cada curva representa a acurácia obtida no processo de classificação, em função da dimensionalidade dos dados e do número de amostras de treinamento empregadas.

Fonte: Landgrebe, 1997.

Para minimizar esse problema, novos métodos vêm sendo investigados. Os métodos apresentados pela comunidade científica classificam-se na descrição a seguir (ANDREOLA, 2009):

- 1) Técnicas de análise discriminante regularizada;
- 2) Redução na dimensionalidade dos dados por métodos de extração/seleção de variáveis;
- 3) Uso de amostras semirodadas;
- 4) Utilização de classificadores não-paramétricos.

Neste sentido, nos últimos anos a comunidade científica mundial tem apontado para a pesquisa de classificadores baseados em aprendizado de máquina. Em particular, estes estudos tem se centrado em classificadores de *Support Vector Machines (SVMs)* devido ao seu grande potencial em diferentes áreas de aplicação, nas quais ele tem apresentando excelentes desempenhos. Especificamente, na área de sensoriamento remoto, a atenção de vários pesquisadores tem-se voltado para essa técnica (*SVM*), visando à detecção de mudanças em imagens multispectrais e multitemporais (BRUZZONE et al., 2003; BOVOLO et al., 2010; BAZI et al., 2005).

Dentre o uso de classificadores *SVM*, uma abordagem interessante está na sua aplicação para dados não linearmente separáveis. Originalmente este processo consiste em utilizar uma função de mapeamento, isto é, uma função que leva os dados do espaço original do problema para um espaço de dimensão mais elevada, no qual a separação das variáveis pode tornar-se linearmente separável (classes separadas por retas ou hiperplanos). Maiores detalhes a este respeito serão mencionados a seguir neste trabalho.

Contudo, pode-se obter o mesmo efeito através do uso de uma função de *kernel*, onde é possível promover a separação linear das classes sem a necessidade de mapear os dados do espaço original do problema para um espaço de maior dimensão. Essa possibilidade de efetuar a separação das classes sem a necessidade de executar o mapeamento dos dados, inclusive sem conhecer a função que faz esse mapeamento, procedendo da classificação dos dados diretamente no espaço original onde ele se encontra é conhecida como truque de *kernel*. Essa capacidade oferecida pela função de *kernel* permite que o classificador de *SVM* obtenha uma boa classificação sobre os dados não linearmente separáveis diretamente no espaço original. Esse bom resultado seria mais complexo de ser obtido ou até inviável sem o uso da função de *kernel* (HAMEL, 2009). Uma abordagem mais profunda a esse respeito será desenvolvida no capítulo 2 desta tese.

O estudo de *kernels* constitui-se, portanto, em um campo importante de pesquisa na área de classificação de imagens e se integra no objetivo deste trabalho, que será melhor detalhado a seguir.

Da mesma forma que tem aumentado o interesse pelo estudo de classificadores *SVM*, também é possível observar, na literatura dos últimos anos, um aumento do interesse pelo desenvolvimento de técnicas automáticas para detecção de mudanças através de imagens multitemporais de sensoriamento remoto. Esse interesse é impulsionado pelo grande número de aplicações em que os métodos de detecção de mudanças, a partir de imagens coletadas por sensores orbitais, podem ser empregados, tais como o monitoramento de desastres ambientais,

queimadas, desmatamentos, estudos urbanos, monitoramento florestal etc. (BRUZZONE, 2000).

Cabe ressaltar que a definição formal de detecção de mudança aplicada em superfícies terrestres envolve o uso de dados multitemporais para determinar na cobertura terrestre as áreas que mudaram no período de aquisição desses dados. Sob esse ponto de vista, a classificação da imagem ao longo do tempo é encarada como uma abordagem mais geral, e portanto, o problema de detecção de mudança é visto como um caso particular dessa classificação. As técnicas de detecção de mudança tornaram-se mais úteis nas últimas décadas, em função de uma maior oferta de dados multitemporais e multifontes disponíveis a partir de plataformas de sensoriamento remoto (MUÑOZ-MARÍ et al., 2009).

O processo de detecção de mudanças envolve, em geral, a análise de duas imagens multiespectrais e multitemporais, cobrindo a mesma área. A partir dessa análise pode ser realizada a detecção de alterações ocorridas na cobertura terrestre desta imagem ao longo de um determinado período.

Levando em consideração o crescimento do interesse por parte da comunidade científica a respeito de estudos envolvendo classificadores paramétricos e técnicas que envolvam a detecção automática de mudanças, no presente trabalho pretende-se abordar a problemática da detecção de mudanças utilizando o classificador de *Support Vector Machines*.

Contudo, mesmo perante à existência de duas principais abordagens para tratar esse problema especificamente na área de sensoriamento remoto, isto é, a utilização do método supervisionado ou do método não supervisionado (SINGH, 1989; BRUZZONE e PRIETO, 2000; LU et al., 2003; BRUZZONE, COSSU e VERNAZZA, 2004). Para estes métodos considera-se como classificação não supervisionada o procedimento em que não há interação entre o sistema e o analista. Contudo, durante a fase de treinamento, certos algoritmos usados para essa classificação, necessitam de alguns parâmetros que devem ser fornecidos ao sistema (MOREIRA, 2011). Por sua vez, a classificação supervisionada é o procedimento utilizado com maior frequência para análise quantitativa de dados de imagens de sensoriamento remoto. Neste processo são utilizados algoritmos para nomear os pixels em uma imagem de forma a representar tipos específicos de cobertura terrestre, ou classes que são conhecidas a priori (verdade terrestre) (LILLESAND e KIEFER, 1994).

Sendo assim, o objetivo desta tese consiste em investigar uma nova técnica para determinação de uma métrica de pertinência para as classes, através de uma abordagem não supervisionada para o problema de detecção de mudanças utilizando o classificador SVM, aliado a um novo método para a obtenção de amostras de treinamento.

O objetivo específico deste trabalho está na implementação da função de decisão *SVM* para detecção de mudanças e da métrica de pertinência de *pixels*, proposta para um par de imagens multiespectrais e multitemporais. Dois *kernels* e seus respectivos parâmetros são testados com a finalidade de investigar sua influência na acurácia produzida. Para validação da metodologia apresentada, os resultados obtidos pelo classificador *SVM* em questão serão confrontados com os obtidos pelo classificador *MVG* e abordagem difusa, apresentada em (ZANOTTA, 2010).

O presente trabalho está estruturado em cinco capítulos:

No primeiro capítulo, Introdução, faz-se uma breve apresentação da tese. No capítulo 2 é feita a revisão bibliográfica, a fim mostrar o atual estado da arte nesta área de estudo, contextualizando o presente trabalho, mostrando sua relevância ao tema e mostrando a contribuição desta pesquisa. Também é apresentado em detalhes o classificador escolhido para este estudo, visando, dessa forma, a melhor compreensão de todas as variáveis e equações envolvidas no processo de classificação e na proposição de uma nova métrica de pertinência.

No capítulo 3 é apresentada a metodologia, de forma detalhada, com a finalidade de compreender todas as nuances do trabalho proposto, bem como aprofundar os conceitos nele envolvidos.

No capítulo 4 são apresentados os experimentos utilizados para validar a metodologia proposta.

Finalmente, no capítulo 5 são apresentadas as expectativas de resultados previstos para este trabalho, bem como as sugestões para futuras investigações.

2 REVISÃO BIBLIOGRÁFICA

O desenvolvimento de metodologias para o processo de classificação de imagens digitais vem sendo objeto da atenção da comunidade científica há alguns anos. Diversas abordagens vêm sendo relatadas na literatura (*Pattern Recognition, Image and Vision Computing, Vision Research, Signal Processing, IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Image Processing, IEEE Transactions on Geoscience and Remote Sensing*, além de vários outros periódicos bem conceituados). Nesta tese foram desenvolvidos estudos relativos à detecção de mudanças em imagens multitemporais e multiespectrais de sensoriamento remoto. O objetivo deste capítulo consiste em apresentar uma síntese das principais abordagens desenvolvidas para a detecção de mudanças em imagens digitais sobre um conjunto de imagens multitemporais, cobrindo uma mesma cena, com a finalidade de inserir a presente tese de doutorado dentro de um contexto atualizado no cenário de pesquisa internacional.

Detectar mudanças em imagens da mesma cena, tomadas em momentos diferentes, é de interesse generalizado para um grande número de aplicações em várias áreas de conhecimento, tais como: vigilância por vídeo (COLLINS, et al, 2000; STAUFER e GRIMSON, 2000; WREN, et al., 1997; XIAOGANG et al., 2014), biologia e medicina (HUERTAS e NEVATIA, 2000; BOSC et. al., 2003; DUMSKYJ, 1996), sistemas de assistência à direção em automóveis (FANG et al., 2003; KAN et al., 1996), em sensoriamento remoto (BOVOLO et al., 2012; BRUZZONE e BOVOLO, 2010 e DEMIR et al., 2012), em particular, na área de sensoriamento remoto, técnicas de detecção de mudanças em imagens multitemporais têm sido aplicadas ao monitoramento agrícola, florestal, de áreas urbanas, de geleiras e oceanos, entre outras aplicações (ZANOTTA e HAERTEL, 2012; THÉAU, 2008).

2.1 Evolução do estudo de detecção de mudanças

Um dos primeiros trabalhos a respeito de detecção de mudanças foi elaborado por Stillman (1920), versando sobre a detecção de pequenas mudanças em duas fotografias, como pode ser visto pela época em que este trabalho foi apresentado, a abordagem, neste caso, é essencialmente analógica. Contudo, as primeiras publicações referentes a esse tema, utilizando imagens digitais, surgiram na década de 60, mais precisamente no trabalho desenvolvido por Rosenfeld (1961), onde foi apresentada uma discussão a respeito do uso de

detecção automática de mudanças para aplicações de uso militar. Em Shepard (1964) são apresentados os aspectos teóricos da detecção de mudança. Na década seguinte, Lillestrand (1972) enfoca a economia computacional obtida ao se representar somente as mudanças que teriam ocorrido entre duas imagens. No final da década de 80, Singh (1989) apresentou uma importante revisão sobre esse tema, na qual classificou as técnicas de detecção de mudanças desenvolvidas até aquela época, levando em conta um extenso levantamento bibliográfico disponível na literatura. Mais recentemente, quase na virada do século, outros pesquisadores como Bruzzone e Serpico (1997) também revisitaram aquela classificação feita por Singh em seu trabalho. Finalmente, na última década, os trabalhos de Radke et al. (2005) e Théau (2008) apresentaram uma análise mais geral dos diferentes algoritmos usados para detecção de mudanças. Recentemente, alguns trabalhos a respeito de detecção de mudança continuam sendo apresentados, entre eles, (BOVOLO et al., 2012; ZANOTTA e HAERTEL, 2012).

2.2. Principais abordagens para detecção de mudanças

A técnica mais simples empregada para detecção de mudanças é a **Imagem da Diferença**, a qual se baseia na subtração entre os contadores digitais de um par de imagens multitemporais, já devidamente registradas e radiometricamente normalizadas. Geralmente, essa técnica é realizada separadamente em cada banda espectral e, partindo-se do pressuposto de que existam poucas mudanças entre as imagens, o comportamento probabilístico dos contadores digitais dos *pixels* da imagem de diferença deverá revelar *pixels* de mudança nas caudas da distribuição de frequências dos *pixels* (Figura 2).

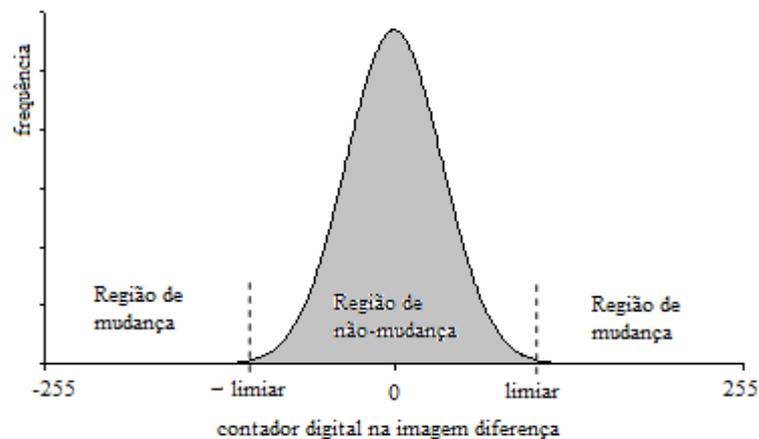


Figura 2 – Função densidade dos contadores digitais dos *pixels* da imagem de diferença com a definição de um limiar entre as classes mudança e não-mudança.

Fonte: Adaptado de BITTENCOURT, 2011.

De forma semelhante à técnica da imagem-diferença, existe a técnica da **Razão entre duas Imagens**. Nessa técnica a razão entre os contadores digitais das imagens devidamente registradas e corrigidas radiometricamente deverá ser unitária nas regiões de não-mudança e deverá apresentar valores maiores ou menores do que um (1) nas regiões de mudança. O desafio nesta técnica é a seleção de valores de fronteira entre as regiões de mudança e não-mudança. Portanto, segundo Singh (1989), os limiares entre as classes de mudança e não-mudança são, em geral, definidos de forma empírica.

Outra técnica, similar às já mencionadas, e que é amplamente utilizada, é a **Análise do Vetor de Mudanças (CVA - Change Vector Analysis)** abordada em Malila (1980), que neste trabalho propõe a utilização da magnitude do vetor mudança. Essa técnica é também trabalhada em (MICHALEK e LUCZKOVICH, 1993; BRUZZONE e BOVOLO, 2007; BOVOLO et al., 2010). Em particular, no trabalho de Bruzzone e Bovolo foi proposto, além do uso da magnitude do vetor mudança o uso de sua direção. Em geral, CVA baseia-se na representação espacial da mudança de *pixel* em um espaço espectral. Isto é, quando um *pixel* sofre uma mudança entre duas datas, essa mudança é representada por um vetor no espaço, definido por dois componentes: a direção, que fornece informações sobre a natureza da mudança e o módulo do vetor, que fornece informações sobre o nível de mudança. Essa técnica apresenta a vantagem de lidar, simultaneamente, com qualquer número de bandas espectrais. Ela também fornece informações detalhadas sobre a mudança (BRUZZONE e SERPICO, 1997). A Figura 3 ilustra a técnica de CVA.

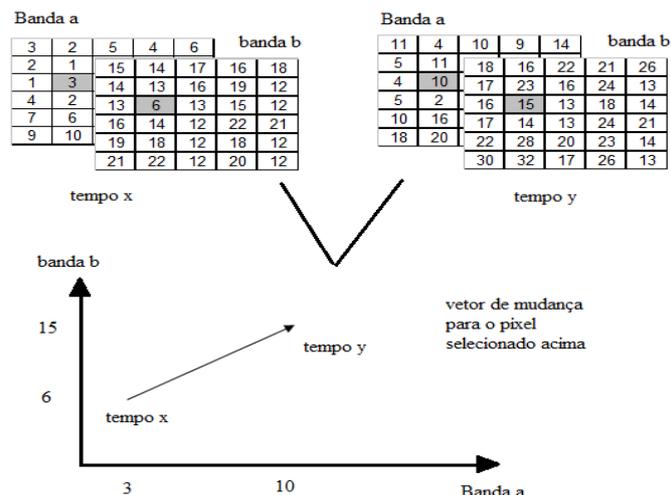


Figura 3 – Exemplo de representação de mudança em um *pixel* usado CVA

Fonte: Adaptado de THÉAU, 2008.

Para as três abordagens acima citadas, a definição do limiar de separação entre as classes mudança e não-mudança é bastante problemática. Nesse sentido, Bruzzone e Prieto (2000) apresentaram uma abordagem não-supervisionada para definição automática desse limiar.

Outra técnica para detecção de mudança é a **detecção de mudanças por regressão**. Nessa abordagem, assume-se que existe uma relação linear entre os valores de *pixel* da mesma área em dois tempos diferentes. Isso implica que a maioria dos *pixels* não apresentará alterações entre as duas datas. Sendo assim, a partir da determinação da função de regressão que melhor descreve a relação entre os valores dos *pixels* de cada banda espectral nas duas datas, podem ser determinados os *pixels* que sofreram alterações. Isto é, os *pixels* não explicados pela regressão e que, portanto estão afastados desta função de regressão são considerados *pixels* de mudanças, os demais, próximos a função de regressão são considerados *pixels* de não-mudança (THÉAU, 2008).

A técnica de **Análise de Componentes Principais** (ACP) é mais comumente empregada para reduzir a dimensionalidade dos dados espectrais. Contudo, quando aplicada a detecção de mudanças ela considera que os primeiros componentes contêm a maior parte da variância dos dados e não estão correlacionadas. Portanto, após a realização da ACP nas imagens, as áreas inalteradas são mapeadas no primeiro componente e as áreas de mudanças são mapeadas nos últimos componentes. Embora a ACP consiga separar a mudança da não-mudança ela é dependente da cena analisada e os resultados podem ser difíceis de interpretar e, além disso, é necessário um bom conhecimento da área de estudo para selecionar os limiares entre as classes mudança e não-mudança.

Outra técnica encontrada na literatura está baseada em **Índices de Vegetação**. Esses índices, normalmente, são tomados em diferentes datas e comparados com o uso de operações aritméticas de diferença ou razão entre *pixels* de diferentes bandas da imagem. É sugerido, nesse método, que o cálculo dos índices sejam realizados separadamente, um em cada data, para posterior comparação. Dessa forma, seriam minimizados os problemas radiométricos que, porventura, existam na imagem.

A **Detecção de Mudanças Pós-Classificação** é, segundo Singh (1989), um dos mais simples e fáceis procedimentos para detecção de mudanças. Nele, as duas imagens adquiridas em diferentes momentos são classificadas separadamente e posteriormente comparadas entre si. A comparação é realizada por meio de tabulação cruzada, gerando a matriz de transição de classes. As vantagens dessa técnica são a minimização de problemas de ordem radiométrica e a fácil interpretação dos resultados. Há, contudo, uma desconfiança por parte dos

pesquisadores desta área no que diz respeito à acurácia dos resultados, uma vez que ao tratar duas imagens classificadas de forma independente, ambas com acurácia de 80%, espera-se que o resultado da classificação caia para 64% (80% x 80%). Entretanto, neste trabalho ele contrariou essa suspeita ao obter os melhores resultados de classificação com esse método em comparação a outros seis métodos de detecção de mudanças aplicados à mesma imagem. Uma possível explicação para esse resultado poderia estar ligado a existência de correlação espacial entre os erros de classificação nas duas imagens, o que poderia acarretar um aumento na acurácia.

Dentre as técnicas usadas para detecção de mudança, os estudos mais recentes apontam para o uso das Máquinas de Vetores de Suporte, mais conhecidos como *Support Vector Machines (SVM)*, que são classificadores lineares e não paramétricos. A preferência por esse tipo de classificador se deve ao fato de que eles operam bem com poucas amostras de treinamento, situação muito comum em imagens estudadas em sensoriamento remoto. Essa característica do classificador *SVM* é um diferencial em relação a classificadores tradicionais paramétricos, que necessitam de um número suficientemente grande de amostras de treinamento para ter uma estimativa confiável do grande número de parâmetros envolvidos neste tipo de imagens. Em virtude das características presentes nos classificadores de *SVM*, vários trabalhos têm sido apresentados à comunidade científica envolvendo a detecção de mudança através do uso de classificadores de *SVM* (BURGES, 1998).

Na próxima seção será abordado, de forma mais detalhada, o classificador *SVM*.

2.3 Classificadores SVM

Esta seção tem a finalidade de abordar o conceito do classificador de *SVM*, que será utilizado como classificador no problema de detecção de mudanças proposto nesta tese.

Como o *SVM* é um classificador linear, antes de defini-lo, será feita uma breve abordagem de classificadores lineares.

Em geral, no processo de classificação buscam-se funções de decisão que permitam separar classes de dados. Em um problema de classificação com n classes, cada classe fica associada com uma função de decisão. Essas n funções de decisão apresentam a mesma forma geral, diferindo uma das outras apenas no valor numérico associado a cada parâmetro ou coeficiente. O processo de classificação ou rotulamento de um padrão consiste em rotulá-lo na classe para a qual o valor correspondente da função de decisão for o maior entre todos

os valores obtidos para esse padrão Bow (1992). Na Figura 4 é apresentado um exemplo de função de decisão para duas classes linearmente separáveis.

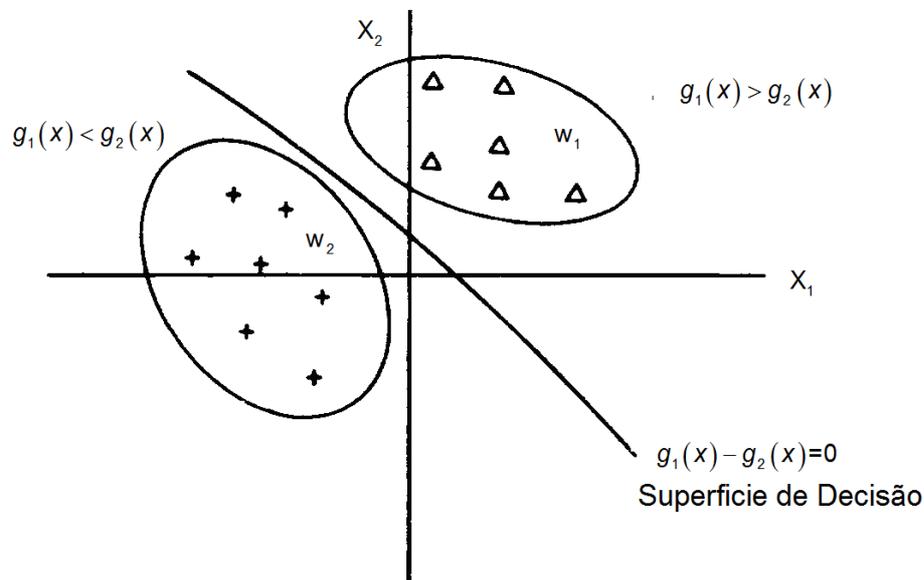


Figura 4 – Separação entre duas classes linearmente separáveis.

Fonte: Adaptado de Bow (1992).

Note que as funções de decisão podem ser qualquer tipo de função, linear ou não-linear. Inicialmente, opta-se pela formulação linear, pois ela é a mais simples de representar, computacionalmente menos onerosa, além de apresentar ótimos resultados no que diz respeito à capacidade de generalização da função, isto é, a capacidade de evitar, por exemplo, que uma superfície de decisão possa classificar corretamente todas as amostras de treinamento disponíveis, mas falhar na classificação de outras amostras que não as empregadas no treinamento do classificador, efeito conhecido como *overfitting* (DUDA et al., 2001). Nesse sentido, *SVM* é um classificador linear que busca minimizar o erro de generalização, conceito que será abordado a seguir. Ainda em relação à separação linear de classes, um problema é dito linearmente separável, como é o caso ilustrado na Figura 4, quando as amostras que representam uma classe situam-se todas de um lado da superfície linear de decisão e as da outra classe no lado oposto (BOW, 1992).

Outro aspecto em relação aos classificadores usados em reconhecimento de padrões é que, em geral, existem duas abordagens para o desenvolvimento desses classificadores em reconhecimento de padrões: abordagem paramétrica e não paramétrica. Na abordagem paramétrica a função de decisão é definida por uma função densidade de probabilidade caracterizada por um determinado número de parâmetros. Na abordagem não paramétrica nenhuma suposição a respeito de distribuição probabilística é feita (BOW, 1992). O

classificador de *SVM*, além de ser um classificador linear, situa-se na segunda abordagem (não paramétrica).

A técnica de *SVM* é oriunda da Teoria da Aprendizagem Estatística (*Statistical Learning Theory*) que se baseia na minimização do risco estrutural (*Structural Risk Minimization – SRM*), a qual consiste em minimizar o erro em relação ao conjunto de amostras de treinamento (risco empírico), assim como, o erro em relação ao conjunto de amostras de teste (risco na generalização), conforme originalmente proposto em Vapnik (1999). A técnica de *SVM* baseia-se em *SRM*, no sentido de procurar minimizar tanto o risco empírico, quanto o risco de generalização. O objetivo do *SVM* é, portanto, conseguir um equilíbrio entre esses erros, reduzindo o excesso de ajustes em relação às amostras de treinamento conhecido por *overfitting* e dessa forma aumentar a capacidade de generalização do processo (VAPNIK, 1999; DUDA et al., 2001). Essa abordagem busca, portanto, evitar que o classificador memorize os padrões de treinamento, capacitando-o a extrair características gerais que permitirão aumentar sua capacidade de generalização (SMOLA, 2000).

Assim, partindo do fato de que, via de regra, o classificador de *SVM* é usado para separar duas classes, esse classificador busca encontrar uma função de decisão linear que possa minimizar o erro de generalização. Se os dados a serem classificados, assim como os dados de treinamento do classificador forem linearmente separáveis, a separação das classes pode ser obtida através de duas funções lineares separando as duas classes presentes na cena, fazendo com que cada conjunto de dados fique exclusivamente em suas classes, ou seja, cada classe fica definida por uma região determinada, por uma desigualdade linear de forma que a função que delimita essa fronteira passe por um ponto dos dados que estejam no limite de sua respectiva classe. Essas funções lineares são paralelas entre si e têm o mesmo coeficiente angular. Dessa forma, todos os pontos pertencentes à determinada classe ficam do mesmo lado, delimitados pela respectiva função linear. Assim, garante-se que não haja sobreposição desses dados com os dados da outra classe. A função linear que separa a classe e passa no ponto mais extremo de cada classe é denominada hiperplano de suporte. Os pontos extremos das classes na sua fronteira são chamados de vetores de suporte (*Support Vectors - SV*). Assim, o hiperplano de decisão, que separa de forma linear as duas classes será um hiperplano intermediário e paralelo a esses já definidos acima (ABE, 2005).

A distância entre os hiperplanos de suporte é chamada de margem. Portanto, o princípio básico de *SVM* é encontrar o hiperplano de separação que fica entre os dois hiperplanos paralelos (hiperplanos de suporte) que separam as duas classes com a máxima

margem de separação entre elas (VAPNIK, 1999). A Figura 5 ilustra essas características do classificador *SVM*.

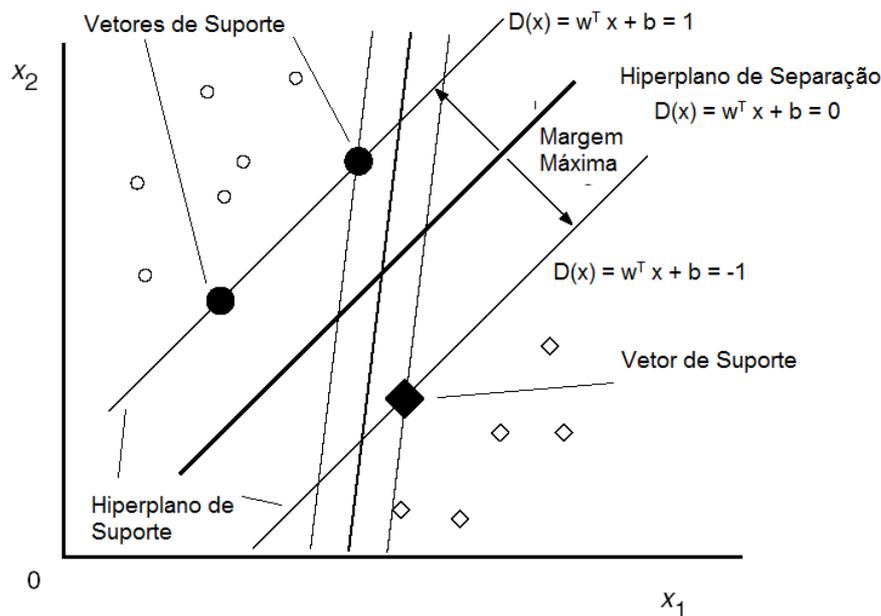


Figura 5 – Superfície de separação linear ótima utilizando *SVM* de margem máxima
Fonte: Adaptado de Abe (2005).

Neste contexto, *SVM* tem se tornado uma importante e eficaz ferramenta para resolver problemas nas áreas de reconhecimento padrão e de regressão, entre outros. Contudo, diferentemente de outros métodos de aprendizagem de máquinas, os erros de generalização das *SVMs* não estão relacionados com a dimensionalidade original dos dados do problema, mas à margem de separação entre os hiperplanos de suporte que separam os dados a serem classificados. Isso explica porque *SVM* apresenta um bom desempenho mesmo em problemas com um grande número de dimensões (PONTIL, 1998). Diferentemente do exemplo mostrado na Figura 4, é comum para dados em geral e, particularmente, em imagens de sensoriamento remoto, que as classes não sejam linearmente separáveis. Nesses casos é possível utilizar duas abordagens, utilizando *SVM*, para resolver esse problema:

Na primeira abordagem, força-se para que o problema original seja considerado de classes quase linearmente separáveis. Para tanto, inclui-se uma variável de folga (ξ_i) à formulação do problema, permitindo que alguns dados permaneçam entre os hiperplanos formados pelos vetores de suporte. Nessa abordagem, os dados que se encontram entre os hiperplanos não devem ser levados em consideração durante o processo de treinamento do classificador, mesmo que eles sejam corretamente classificados, garantindo, dessa forma, a determinação da margem máxima de separação entre os hiperplanos de suporte, fato que não

ocorreria caso esses dados fossem aproveitados no treinamento do classificador. Para determinar o quanto essas variáveis de folga devem ser consideradas no processo de treinamento, é utilizado um parâmetro de custo C inserido à formulação original do problema, que será melhor detalhado nas próximas seções. Portanto, esse parâmetro irá determinar a importância que as variáveis de folga empregadas no classificador terão no treinamento dele. A Figura 6 ilustra essa situação.

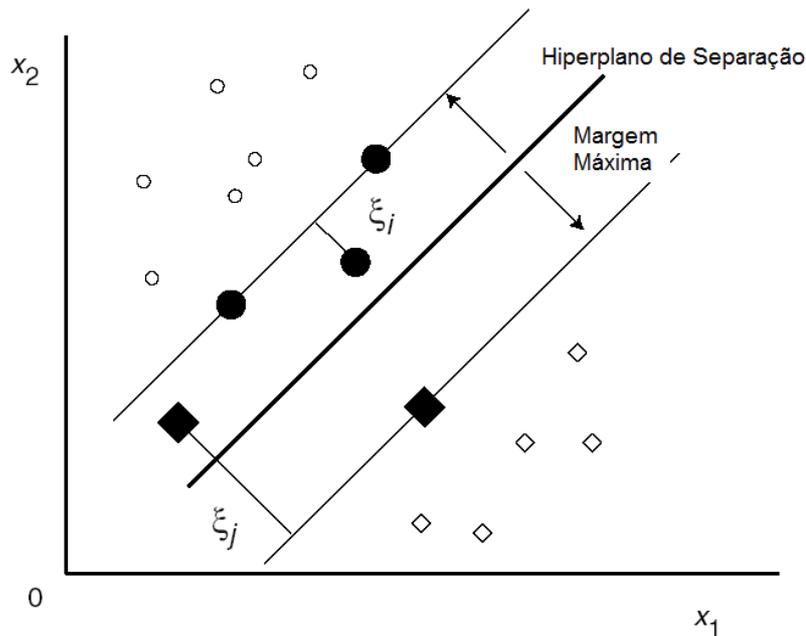


Figura 6 – Determinação do hiperplano de Separação de classes com margem máxima com uso de variáveis de folga
Fonte: Adaptado de Abe (2005).

Essa abordagem, como mencionada acima no texto, pode ser aplicada para classes quase linearmente separáveis, desde que algumas das amostras de treinamento sejam consideradas como valores ruidosos e, portanto desconsideradas no processo de treinamento do classificador, ou seja, o problema é forçado a assumir um comportamento de classes linearmente separáveis, razão pela qual essa abordagem não é utilizada na prática, pois nem sempre as classes que não são linearmente separáveis podem ser consideradas como quase linearmente separável.

Dessa forma, a segunda e última abordagem possível leva em conta o fato de que dados não linearmente separáveis podem se tornar linearmente separáveis quando mapeados para um espaço de dimensão mais alta que o original (DUDA et al., 2001). Sendo assim, pode-se adotar um vetor de funções de mapeamento $g(\mathbf{x})$, que tem a finalidade de mapear os dados do espaço original para um espaço de maior dimensão, onde as classes possam se tornar linearmente separáveis. A função de decisão linear para esse novo espaço tem características

semelhantes à função de decisão baseada no classificador *SVM*, que seria utilizada no espaço original se os dados fossem linearmente separáveis, ou seja, no espaço de maior dimensão, os dados seriam separados em classes através de dois hiperplanos de suporte e existiria uma margem máxima entre eles.

A determinação matemática dessa função de decisão pode ser obtida através de um problema de otimização que determine a máxima margem entre os hiperplanos de suporte e pode ser resolvida segundo duas abordagens: a primal e a dual. Na abordagem primal deve-se definir uma função de mapeamento para levar os dados originais para o espaço de dimensão superior para então obter a separação linear das classes. Alternativamente, a abordagem dual implica na escolha de uma função *kernel*, que expressa o produto interno dos dados originais. Essa última abordagem apresenta a vantagem de ser mais facilmente implementada e não ser necessária a classificação diretamente no espaço de maior dimensão, onde os dados seriam mapeados. Sendo assim, não é necessário conhecer a expressão analítica da função de mapeamento, nem mesmo executar o mapeamento propriamente dito desses pontos do espaço original para o espaço de característica (espaço de maior dimensionalidade), bastando calcular o valor da função de *kernel* para os dados usados no processo de treinamento do classificador diretamente no espaço original dos dados (mais detalhe desse processo serão vistos na próxima seção) (YANG et al., 2006).

O uso do *kernel* diretamente no espaço original, independentemente do conhecimento prévio da função de mapeamento, é conhecido na literatura como truque de *kernel* (ABE, 2005).

Ao se utilizar a abordagem dual para a determinação da função de decisão entre classes baseadas no classificador *SVM*, ela pode ser encontrada através da solução do problema de otimização dual para determinação da margem máxima entre os hiperplanos de suporte. Para determinar essa solução, aplica-se a técnica dos multiplicadores de Lagrange, com o uso das condições de Karush-Kuhn-Tucker (condições de KKT, a qual será vista mais detalhadamente na próxima seção, onde será formulada matematicamente uma solução para o problema de maximização da margem entre os hiperplanos de suporte).

2.3.1 Aspectos matemáticos dos classificadores SVM

A separação de dados linearmente separáveis pode ser realizada de várias maneiras, empregando diferentes superfícies lineares de separação (BOW, 1992). Contudo, conforme apresentado na seção anterior, a função de decisão linear mais indicada para a separação desse

tipo de classe é aquela determinada por *SVM*, na qual se emprega um hiperplano equidistante de dois outros hiperplanos de suporte paralelos, os quais contêm os vetores de suporte de cada classe. Esses hiperplanos paralelos limitam cada uma das classes envolvidas na classificação de maneira que todos os padrões pertencentes a cada classe ficam inteiramente de um lado do respectivo hiperplano de suporte e a margem entre eles é a maior possível. Essa configuração dos hiperplanos garante que na região entre os hiperplanos não haja sobreposição de padrões de nenhuma classe, como bem ilustra a Figura 6 acima.

A determinação dessa margem máxima entre os hiperplanos de suporte, doravante representada por (m^*) , é obtida ao se maximizar a projeção da diferença de vetores de suporte $\mathbf{x}_p - \mathbf{x}_q$ na direção do vetor \mathbf{w}^* , que é o vetor normal ao hiperplano de separação entre classes, onde os vetores de suporte \mathbf{x}_p e \mathbf{x}_q pertencem, respectivamente, a cada uma das classes e cumprem, respectivamente, as equações $\mathbf{w}^* \cdot \mathbf{x} = b^* + k$ e $\mathbf{w}^* \cdot \mathbf{x} = b^* - k$, que expressam os hiperplanos de suporte de cada uma das duas classes, onde b é o termo independente da equação do hiperplano e k uma constante qualquer. O valor θ representa o ângulo entre o vetor diferença $\mathbf{x}_p - \mathbf{x}_q$ e o vetor \mathbf{w}^* , ou seja, o valor de m (margem entre hiperplanos de suporte) pode ser determinado pela equação dada em (1)

$$\begin{aligned} m &= |\mathbf{x}_p - \mathbf{x}_q| \cos \theta = \frac{\mathbf{w}^* \cdot (\mathbf{x}_p - \mathbf{x}_q)}{|\mathbf{w}^*|} = \frac{\mathbf{w}^* \cdot \mathbf{x}_p - \mathbf{w}^* \cdot \mathbf{x}_q}{|\mathbf{w}^*|} = \\ &= \frac{(-b^* + k) - (-b^* - k)}{|\mathbf{w}^*|} = \frac{2k}{|\mathbf{w}^*|} \end{aligned} \quad (1)$$

Para a equação acima, se for tomada a constante k com valor igual a 1, as equações $\mathbf{w}^* \cdot \mathbf{x} = b^* + k$ e $\mathbf{w}^* \cdot \mathbf{x} = b^* - k$ podem ser reescritas como $\mathbf{w} \cdot \mathbf{x} = b^* + 1$ e $\mathbf{w} \cdot \mathbf{x} = b^* - 1$ e as regiões acima e abaixo desses hiperplanos de suporte podem ser expressos pelas desigualdades $\mathbf{w} \cdot \mathbf{x} \geq b^* + 1$ e $\mathbf{w} \cdot \mathbf{x} \leq b^* - 1$ ou de forma mais elegante, por $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0$, onde $y_i \in \{-1, 1\}$ é o rótulo atribuído às classes envolvidas no processo. Dessa forma, os pontos a serem classificados devem estar restritos a essas regiões. Assim, a margem máxima m^* é obtida pela maximização da expressão dada em (1) e expressa em (2) pela minimização do produto interno do vetor \mathbf{w} :

$$m^* = \max \left(\frac{2k}{|\mathbf{w}|} \right) = \min \left(\frac{|\mathbf{w}|}{2k} \right) = \min \left(\frac{|\mathbf{w}|^2}{2k} \right) = \min \left(\frac{1}{2k} \mathbf{w} \cdot \mathbf{w} \right) \underset{k=1}{=} \min \left(\frac{1}{2} \mathbf{w} \cdot \mathbf{w} \right) \quad (2)$$

A Figura 7 ilustra esse processo de determinação da margem máxima entre os hiperplanos de separação das classes a partir do módulo da projeção da diferença dos vetores de suporte na direção do vetor normal ao hiperplano de separação.

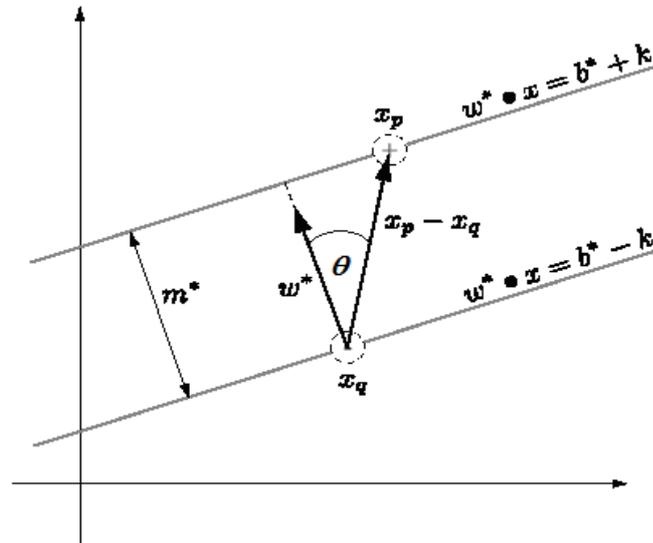


Figura 7 – Determinação da margem máxima m^* entre dois hiperplanos de suporte
Fonte: adaptado de Hamel (2009).

De maneira mais resumida e elegante, pode-se formular matematicamente o problema de otimização não linear expresso na equação (2) através do problema de minimização descrito pela função objetivo (3) e condição de restrição (4) (LUENBERGER, 1984):

$$\text{Minimizar: } \left(\frac{1}{2} \|\mathbf{w}\|^2 \right) = \frac{1}{2} (\mathbf{w} \cdot \mathbf{w}) = \frac{1}{2} (\mathbf{w}^T \mathbf{w}) \quad (3)$$

$$\text{Sujeito a: } y_i (\mathbf{w}^T \mathbf{x}_i - b) - 1 \geq 0 \text{ para } i=1, 2, \dots, M \quad (4)$$

$$\text{e } y_i \in \{-1, 1\}$$

A condição de restrição dada em (4) supõe uma classificação sem erro no processo de treinamento do SVM, o que é denominado como um problema de SVM primal de margem rígida.

A Figura 8, a seguir, ilustra a separação das duas classes linearmente separáveis através de um hiperplano intermediário entre os hiperplanos de suporte e o que é expresso pela equação $(\mathbf{w}^T \mathbf{x} + b) = 0$, onde deve ser garantida a margem máxima de separação entre as classes (distância entre os conjuntos das amostras de treinamento é maximizada). Esse

hiperplano, como já comentado, fica em igual distância de dois hiperplanos de suporte que delimitam as duas classes envolvidas na classificação e que contêm os respectivos vetores de suporte (*Support Vectors - SV*) de cada classe e são expressos pelas equações $(\mathbf{w}^T \mathbf{x} + b) = 1$ e $(\mathbf{w}^T \mathbf{x} + b) = -1$.

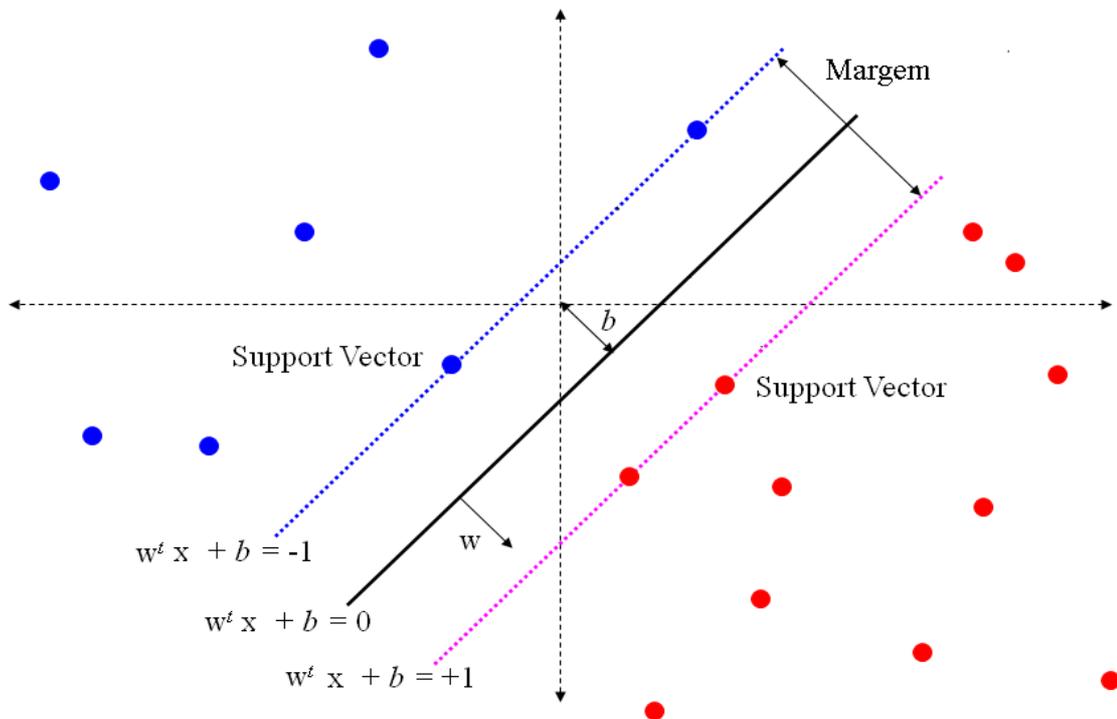


Figura 8 – Hiperplano ótimo separando os dados com a máxima margem
 Fonte: Adaptado do site: <http://research.microsoft.com/en-us/groups/vgv/vgv4.png>.

A inclusão da restrição (4) no problema de minimização de (3) pode ser resolvida através do uso da técnica dos multiplicadores de Lagrange, que serão representados doravante pelo vetor (α) . Nessa formulação existe exatamente um multiplicador de Lagrange α_i para cada restrição dada pela expressão $g_i(\mathbf{x}) = y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1$ e onde o vetor \mathbf{x} é considerado a variável primal e α é a variável dual do problema. Essa abordagem permite reescrever o problema anterior, equações (2.3) e (2.4), da seguinte forma:

$$\text{Minimizar: } \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^M \alpha_i \{y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1\}, \quad (5)$$

onde $\alpha = (\alpha_1, \dots, \alpha_M)$ é um vetor de dimensão M, onde M representa o número de amostras de treinamento usado no classificador, formado com os Multiplicadores de Lagrange

não negativos ($\alpha_i \geq 0, i = 1, \dots, M$), os quais são usados para cada uma das restrições do problema, como já mencionado anteriormente.

Desenvolvendo o problema de minimização dado em (5) através da função $L(\boldsymbol{\alpha}, \boldsymbol{w}, b)$, que deve ser minimizada, tem-se:

$$\begin{aligned} L(\boldsymbol{\alpha}, \boldsymbol{w}, b) &= \frac{1}{2} \boldsymbol{w} \bullet \boldsymbol{w} - \sum_{i=1}^M \alpha_i \left\{ y_i (\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1 \right\} = \\ &= \frac{1}{2} \boldsymbol{w} \bullet \boldsymbol{w} - \sum_{i=1}^M \alpha_i y_i \boldsymbol{w}^T \boldsymbol{x}_i - \sum_{i=1}^M \alpha_i y_i b + \sum_{i=1}^M \alpha_i. \end{aligned} \quad (6)$$

A solução para esse problema de minimização, envolvendo multiplicadores de Lagrange, deve cumprir as condições de Karush–Kuhn–Tucker (KKT), que são apresentadas nas equações (7) a (11) a seguir:

$$\frac{\partial}{\partial \boldsymbol{w}} L(\boldsymbol{\alpha}, \boldsymbol{w}, b) = 0 \quad (7)$$

$$\frac{\partial}{\partial b} L(\boldsymbol{\alpha}, \boldsymbol{w}, b) = 0 \quad (8)$$

$$\alpha_i \left\{ y_i (\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1 \right\} = 0 \quad (9)$$

$$y_i (\boldsymbol{w}^T \boldsymbol{x}_i + b) - 1 \geq 0 \quad (10)$$

$$\alpha_i \geq 0 \quad (11)$$

Assim, aplicando cada uma dessas condições no problema de minimização apresentado em (5) e reescrito na forma de função, expressão dada em (6), tem-se os resultados expressos em (12) e (13) em vermelho:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{w}} L(\boldsymbol{\alpha}, \boldsymbol{w}^*, b^*) &= \frac{\partial}{\partial \boldsymbol{w}} \left(\frac{1}{2} \boldsymbol{w} \bullet \boldsymbol{w} - \sum_{i=1}^M \alpha_i y_i \boldsymbol{w}^T \boldsymbol{x}_i - \sum_{i=1}^M \alpha_i y_i b + \sum_{i=1}^M \alpha_i \right) = \\ &= \boldsymbol{w} - \sum_{i=1}^M \alpha_i y_i \boldsymbol{x}_i = 0 \therefore \boldsymbol{w} = \sum_{i=1}^M \alpha_i y_i \boldsymbol{x}_i \end{aligned} \quad (12)$$

$$\begin{aligned}
\frac{\partial}{\partial b} L(\boldsymbol{\alpha}, \mathbf{w}^*, b^*) &= \frac{\partial}{\partial b} \left(\frac{1}{2} \mathbf{w} \bullet \mathbf{w} - \sum_{i=1}^M \alpha_i y_i \mathbf{w}^T \mathbf{x}_i - \sum_{i=1}^M \alpha_i y_i b + \sum_{i=1}^M \alpha_i \right) \\
&= - \sum_{i=1}^M \alpha_i y_i = 0 \quad \therefore \quad \sum_{i=1}^M \alpha_i y_i = 0
\end{aligned} \tag{13}$$

A partir desses resultados, pode-se reescrever a função (6) utilizando os resultados de (12) e (13). Isto é, tomando $\mathbf{w} = \sum_{i=1}^M \alpha_i y_i \mathbf{x}_i$ e $\sum_{i=1}^M \alpha_i y_i = 0$, pode-se reescrever a função dada em (6) da seguinte forma:

$$\begin{aligned}
L(\boldsymbol{\alpha}, \mathbf{w}, b) &= \frac{1}{2} \mathbf{w} \bullet \mathbf{w} - \sum_{i=1}^M \alpha_i \left\{ y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 \right\} = \\
&= \frac{1}{2} \left(\sum_{i=1}^M \alpha_i y_i \mathbf{x}_i \right) \bullet \left(\sum_{j=1}^M \alpha_j y_j \mathbf{x}_j \right) - \sum_{i=1}^M \left(\alpha_i y_i \left(\sum_{j=1}^M \alpha_j y_j \mathbf{x}_j \right)^T \mathbf{x}_i \right) - b \cdot \sum_{i=1}^M \alpha_i y_i + \sum_{i=1}^M \alpha_i = \\
&= \frac{1}{2} \left(\sum_{i=1}^M \sum_{j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i \bullet \mathbf{x}_j \right) - \sum_{i=1}^M \sum_{j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i \bullet \mathbf{x}_j + \sum_{i=1}^M \alpha_i
\end{aligned} \tag{14}$$

Portanto, essa nova formulação apresentada em (14) substitui a apresentada em (6), porém, neste caso, envolvendo apenas como variável o vetor de multiplicadores de Lagrange ($\boldsymbol{\alpha}$), dessa forma, não sendo mais necessário usar as variáveis: vetor \mathbf{w} e b , isto é, o problema pode ser expresso somente em função do vetor $\boldsymbol{\alpha}$, como apresentado em (15):

$$L(\boldsymbol{\alpha}) = \sum_{i=1}^M \alpha_i - \frac{1}{2} \left(\sum_{i=1}^M \sum_{j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i \bullet \mathbf{x}_j \right) \tag{15}$$

Dessa forma, o problema definido em (5) pode ser agora expresso pela equação (16) e restrição (17) apresentadas a seguir:

$$\text{Maximizar: } \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \tag{16}$$

$$\text{Sujeito a: } \sum_{i=1}^M \alpha_i y_i = 0 \quad \text{e} \quad \alpha_i \geq 0 \quad \text{para } i=1, \dots, M \tag{17}$$

Esta última formulação é denominada de forma dual do problema; e o problema original, dado em (3), é conhecido como forma primal. A forma dual é muito atraente, pois apresenta as restrições de forma mais simples e permite que se lide com o problema de otimização em termos de produtos internos entre os dados originais, o que facilita o tratamento de dados não-lineares (ABE, 2005).

A partir da solução desse problema de otimização pode-se agora obter a função de decisão que separa as classes com margem máxima. A qual é dada por:

$$D(\mathbf{x}) = \sum_{i \in S} \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b, \quad (18)$$

onde S corresponde ao conjunto de índices de vetores de suporte.

Dessa forma, se os dados a serem classificados não forem linearmente separáveis (situação comum encontrada em imagens de sensoriamento remoto), pode-se introduzir ao problema de otimização dado anteriormente em (3) e (4) uma variável de folga representada por ξ_i , sendo $\xi_i \geq 0$. Essa variável tem a finalidade, neste caso, de relaxar as restrições do problema aproveitando a sua estruturação, ou seja, a inserção dessa variável de folga permite que possa ser violada a restrição imposta em (4) e ao aplicar-se esse procedimento é permitido que sejam suavizadas as margens do classificador linear, permitindo que alguns dados permaneçam entre os hiperplanos de suporte, da mesma forma que fica permitido que haja a ocorrência de alguns erros de classificação e, portanto, o problema de minimização apresentado em (5), neste caso, pode ser reescrito pelas expressões (19) e (20). (ABE, 2005)

$$\text{Minimizar: } \frac{1}{2} (\mathbf{w}^T \mathbf{w}) + C \sum_{i=1}^M \xi_i \quad (19)$$

$$\text{Sujeito a : } y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 + \xi_i \geq 0 \quad \text{para } i=1, 2, \dots, M \quad (20)$$

A constante C, que aparece na equação (19), como já foi mencionado anteriormente, é o “parâmetro de margem” ou “constante de custo” e ela estabelece a importância relativa das duas parcelas presentes neste processo de minimização. Ou seja, ao minimizar o primeiro termo $\frac{1}{2} (\mathbf{w}^T \mathbf{w})$ se estará maximizando a margem do classificador, enquanto que ao minimizar a segunda parcela $(\sum_{i=1}^M \xi_i)$, haverá a minimização do erro de classificação. As SVMs obtidas neste caso são conhecidas como SVMs de margens suaves (*soft-margin*) (Abe, 2005). A Figura 9, a seguir, ilustra a separação das duas classes não linearmente separáveis

através de um hiperplano determinado pela equação $(\mathbf{w}^T \mathbf{x} + b) = 0$, o qual garante a margem máxima de separação entre as classes e, como já comentado anteriormente, esse hiperplano fica em igual distância de dois hiperplanos de suporte que contêm os vetores de suporte e são determinados pelas equações $(\mathbf{w}^T \mathbf{x} + b) = 1$ e $(\mathbf{w}^T \mathbf{x} + b) = -1$. Utilizando as variáveis de folga ξ_i , pode-se perceber que sempre existirão soluções viáveis. Contudo, se $0 < \xi_i < 1$ (Figura 8), os dados relativos a essa variável de folga não têm margem máxima, mas ainda estão corretamente classificados. Contudo, se $\xi_i \geq 1$ (Figura 9) os dados são classificados erroneamente pelo hiperplano de margem máxima. Além disso, se $\xi_i = 0$ (Figura 9) o ponto em questão pertence ao hiperplano de suporte e esse ponto é um vetor de suporte (ABE, 2005):

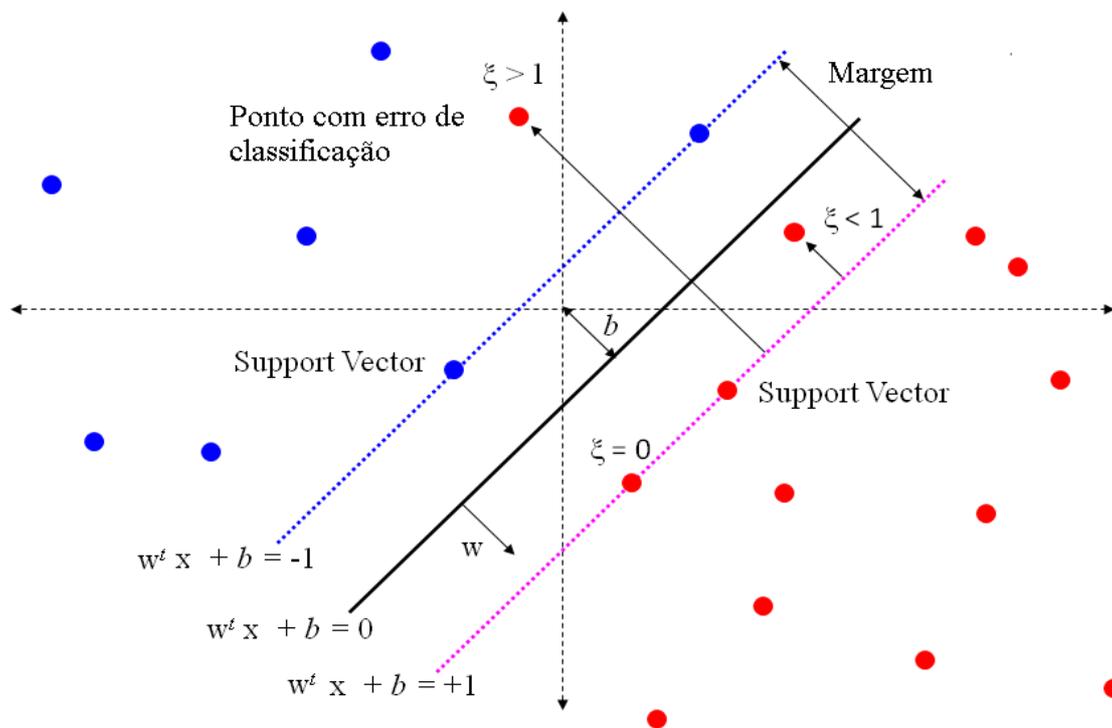


Figura 9 – Hiperplano separando os dados com a máxima margem.

Fonte: Adaptado do site: <http://research.microsoft.com/en-us/groups/vgv/vgv4.png>

A formulação apresentada em (19) e (20) acima, está sujeita à aplicação das condições de KKT, semelhantemente ao que foi feito anteriormente em (5), com a introdução de uma abordagem Lagrangiana, gerando uma formulação dual igual à encontrada em (15) e que é apresentada em (21), mas sujeita às restrições dadas em (22):

$$\text{Maximizar: } \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \quad (21)$$

$$\text{Sujeito a: } \sum_{i=1}^M \alpha_i y_i = 0 \quad \text{e} \quad C \geq \alpha_i \geq 0 \quad \text{para } i=1, \dots, M \quad (22)$$

Como já comentado anteriormente, em geral as classes presentes em cenas de sensoriamento remoto não são linearmente separáveis, o que permite que seja usada a formulação acima. Contudo, isso pode ocasionar a perda da habilidade de generalização existente na técnica, uma vez que se está tratando um problema de classes não linearmente separáveis, como se elas fossem. Uma alternativa para evitar essa perda consistiria em mapear os dados do espaço original para um novo, com dimensão mais alta que a original do problema, também conhecido como espaço de produto interno, no qual os dados passam a ser linearmente separáveis, como já comentado anteriormente. Esse novo espaço é também conhecido como espaço de característica (*feature space*). Esta transformação pode ser feita por meio de uma função de mapeamento $g(\mathbf{x})$. Assim, adaptando-se a última formulação obtida em (21) e considerando que:

$$\mathbf{w} = \sum_{i=1}^M \alpha_i y_i g(\mathbf{x}_i) \quad (23)$$

Esse valor de \mathbf{w} é obtido a partir da derivação da equação dada em (5) em relação a \mathbf{w} , agora considerando não o valor de \mathbf{x} no espaço original, mas o valor do mapeamento deste ponto no novo espaço de característica através da função de mapeamento $g(\mathbf{x})$. Ao se selecionar uma função de mapeamento $g(\mathbf{x})$, a forma geral da função de decisão dada em (18) torna-se:

$$D(\mathbf{x}) = \mathbf{w} \cdot g(\mathbf{x}) + b \quad (24)$$

Agora, usando (23) e (24) a expressão para a função de decisão neste novo espaço pode ser escrita por:

$$D(\mathbf{x}) = \sum_{i \in S} \alpha_i y_i g(\mathbf{x}_i) g(\mathbf{x}) + b \quad (25)$$

Os elementos do produto interno $g(\mathbf{x}_i) \cdot g(\mathbf{x})$ podem ser representados pela matriz \mathbf{K} , que é conhecida por matriz de *kernel*, com dimensionalidade (M, M) e onde S representa o conjunto dos índices dos *Support Vectors*:

$$\mathbf{K}_{i,j} = g(\mathbf{x}_i) \cdot g(\mathbf{x}_j) \quad \text{para } i, j = 1, \dots, M \quad (26)$$

A função de *kernel* pode, portanto, ser definida como o produto interno das variáveis (no espaço característico). E essa função está sujeita à condição de existência de Mercer (ABE, 2005; SCHÖLKOPF e SMOLA, 2002), isto é:

$$\sum_{i,j=1}^M h_i h_j \mathbf{K}_{i,j} \geq 0, \quad \text{para todo } M, x_i \text{ e } h_i \quad (27)$$

Onde M é um número natural e no caso de SVM é o número de amostras de treinamento disponíveis; h_i é um número real, para o qual existe uma função de mapeamento $g(\mathbf{x})$, que mapeia \mathbf{x} do espaço original para espaço característico.

Seguindo a metodologia anteriormente utilizada em (21) e (22), o problema de maximização adquire a seguinte forma:

$$\text{Maximizar: } \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (28)$$

$$\text{sujeito a: } \sum_{i=1}^M y_i \alpha_i = 0 \quad \text{e} \quad 0 \leq \alpha_i \leq C \quad \text{para } i=1, \dots, M \quad (29)$$

Nota-se que a escolha de uma determinada função de *kernel* $K(\mathbf{x}_i, \mathbf{x}_j)$, gera uma SVM diferente. O que sugere que uma boa escolha de função de *kernel* deve gerar um bom classificador SVM.

O processo de mapeamento, comentado anteriormente, encontra-se ilustrado pela Figura 10. Nesse exemplo, no espaço original os dados não são linearmente separáveis (Figura 10a). Ao serem mapeados para um determinado espaço de dimensão mais alta, esses dados podem se tornar linearmente separáveis (Figura 10b). Se esses dados, agora linearmente separados, forem mapeados de volta ao espaço original, o hiperplano de separação naquele espaço corresponderia a uma superfície não-linear, separando os dados no espaço original (Figura 10c). Nesse contexto, a vantagem do uso de *kernels* consiste na possibilidade de empregar superfícies lineares de decisão (que são mais simples e possuem uma capacidade de generalização mais alta, minimizando o problema de *overfitting*) na formulação do problema de SVM para separar dados que em sua forma original não são linearmente separáveis,

executando o processo de separação, descrito acima, diretamente no espaço original sem a necessidade do mapeamento de um espaço de dimensão mais alta, nem mesmo o conhecimento da função que faz esse mapeamento.

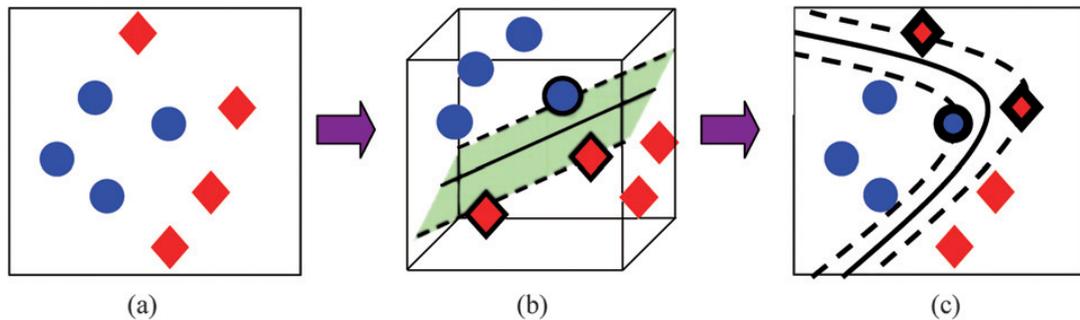


Figura 10 – (a) Duas classes não linearmente separáveis em duas dimensões. (b) mapeamento dos dados originais para um espaço de maior dimensão que o espaço original, onde é possível separar as classes de forma linear através de um hiperplano ótimo e hiperplanos contendo os *support vectors*. (c) remapeamento dos hiperplanos de separação para o espaço original marcando a separação não linear das classes.

Fonte: Adaptado de Brereton e Lloyd (2010).

Dessa forma, a função de *kernel* desempenha um papel preponderante no classificador de *SVM*, pois simplifica o processo de classificação, simulando a ocorrência da separação linear das classes no espaço de maior dimensão sem a necessidade de fazer o mapeamento propriamente dito e sem a necessidade de conhecer as funções de mapeamento, usando apenas os dados contidos no espaço original e fazendo separação entre classes diretamente na imagem original.

Na próxima subseção se fará um estudo mais detalhado desta função de *kernel*.

2.4 *kernel*

Em *SVM* o hiperplano ótimo é determinado com a finalidade de maximizar a capacidade de generalização do classificador. Contudo, se os dados de treinamento não são linearmente separáveis, o classificador obtido pode não possuir uma elevada capacidade de generalização, embora o hiperplano determinado seja obtido de forma otimizada (ABE, 2005). Assim, para aumentar a capacidade de separação entre as classes, o espaço original da cena é mapeado em um novo espaço, produto interno de alta dimensionalidade chamado de espaço de característica (*feature space*), o qual possibilita a separação linear entre as classes originais.

A função que gera essa transformação de dados da cena original para um novo espaço de característica e que possibilita a separação linear das classes é conhecida como função de mapeamento e o produto interno dessas funções de mapeamento é conhecido como função de *kernel* (ABE, 2005). A grande vantagem de usar funções *de kernel* é que não há necessidade de mapear os dados do espaço original para um espaço de dimensão mais alta através das funções de mapeamento e não é necessário conhecê-las. Sendo assim, o processo de mapeamento, que pode tornar-se computacionalmente oneroso, pode ser evitado pelo uso de funções de *kernel*, que operam diretamente no espaço original.

O exemplo abaixo serve para ilustrar essa facilidade oferecida pelo uso da função de *kernel*. Essa possibilidade de operar diretamente no espaço original é conhecida como truque de *kernel* (Figura 11). Na parte (a) desta figura pode-se perceber que não existe função de decisão linear capaz de separar as duas classes envolvidadas (+ e -). Contudo, é fácil perceber que uma equação de circunferência dada pela equação vetorial $\mathbf{x} \cdot \mathbf{x} = 1$ separa adequadamente os dados. Na Figura (10b), a partir de uma função de mapeamento $g(\mathbf{x})$ pode-se construir um hiperplano $\mathbf{w} \cdot g(\mathbf{x}) = b$ que separa linearmente dos dados:

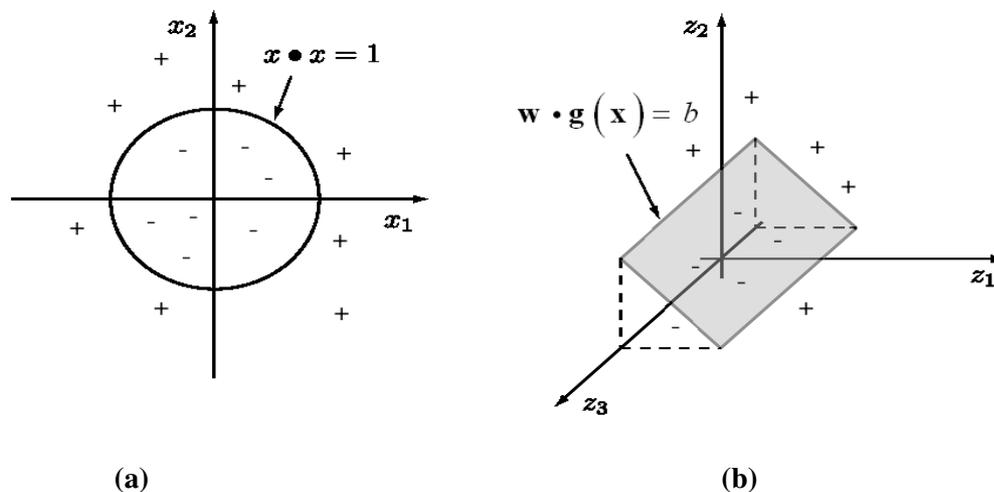


Figura 11 – Mapeamento de um conjunto de dados não linearmente separáveis.
Fonte: HAMEL (2009).

Para o problema ilustrado acima, foi feito um mapeamento através da função $g : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ definida pela seguinte expressão analítica:

$$g(\mathbf{x}) = g(x_1, x_2) = (x_1^2, x_2^2, \sqrt{2}x_1x_2) \quad (30)$$

Dessa forma, a partir desse mapeamento qualquer ponto no espaço original \mathbb{R}^2 é mapeado para um novo espaço \mathbb{R}^3 e separado linearmente pela função de decisão $\mathbf{w} \cdot \mathbf{g}(\mathbf{x}) = b$. Onde o vetor \mathbf{w} e o valor b são valores a serem determinados. Se for tomado um ponto na circunferência que separa de forma não linear os dados no espaço original, pode-se mostrar que esse ponto é mapeado para o hiperplano de separação entre classes no espaço \mathbb{R}^3 definido em (10b). Portanto, vale também o contrário, um ponto no hiperplano de separação é mapeado na circunferência. Essa afirmação pode ser mostrada tomando o ponto (x_1, x_2) que pertence à circunferência no espaço original e que tem como função de mapeamento deste ponto a função $g(x_1, x_2) = (x_1^2, x_2^2, \sqrt{2}x_1x_2)$. Agora, se for considerado como conhecido o valor do vetor normal ao hiperplano, $\mathbf{w} = (1, 1, 0)$, então $\mathbf{w} \cdot \mathbf{g}(\mathbf{x}) = (1, 1, 0) \cdot (x_1^2, x_2^2, \sqrt{2}x_1x_2) = x_1^2 + x_2^2 = b$ e, neste caso, se for considerado $b = 1$ então $x_1^2 + x_2^2 = \mathbf{x} \cdot \mathbf{x} = 1$, ou seja, qualquer ponto da circunferência no espaço original \mathbb{R}^2 é mapeado para um ponto no hiperplano de separação entre classes do espaço \mathbb{R}^3 .

De forma análoga, todo ponto acima do hiperplano em \mathbb{R}^3 corresponderá a um ponto fora da circunferência em \mathbb{R}^2 e todo ponto abaixo do hiperplano em \mathbb{R}^3 corresponderá a um ponto interno da circunferência em \mathbb{R}^2 .

Se for considerada a solução da SVM dada por $\mathbf{w} = \sum_{i=1}^M \alpha_i y_i g(\mathbf{x}_i)$ a representação da função de decisão pode ser expressa por:

$$f(x) = \text{sgn}(\mathbf{w} \cdot \mathbf{g}(\mathbf{x}) + b) = \text{sgn}\left(\sum_{i=1}^M \alpha_i y_i g(\mathbf{x}_i) \cdot \mathbf{g}(\mathbf{x}) + b\right) \quad (31)$$

Usando a função de mapeamento definida em (30) na função de decisão (31), temos a função de decisão específica para o problema acima, que é dada em (32):

$$\begin{aligned}
f(x) &= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i g(\mathbf{x}_i) \bullet g(\overline{\mathbf{x}}_j) + b \right) = \\
&= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i (x_1^2, x_2^2, \sqrt{2}x_1x_2) \bullet (x_1^2, x_2^2, \sqrt{2}x_1x_2) + b \right) = \\
&= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i (x_{i1}^2, x_{i2}^2, \sqrt{2}x_{i1}x_{i2}) \bullet (x_{j1}^2, x_{j2}^2, \sqrt{2}x_{j1}x_{j2}) + b \right) = \\
&= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i (x_{i1}^2x_{j1}^2 + x_{i2}^2x_{j2}^2 + 2x_{i1}x_{j1}x_{i2}x_{j2}) + b \right) = \\
&= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i (x_{i1}x_{j1} + x_{i2}x_{j2})(x_{i1}x_{j1} + x_{i2}x_{j2}) + b \right) = \\
&= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i (\mathbf{x}_i \bullet \mathbf{x}_j)(\mathbf{x}_i \bullet \mathbf{x}_j) + b \right) = \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i (\mathbf{x}_i \bullet \mathbf{x}_j)^2 + b \right) = \\
&= \operatorname{sgn} \left(\sum_{i=1}^M \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b \right)
\end{aligned} \tag{32}$$

Onde $K(x_i, x_j) = (x_i \bullet x_j)^2$ é a função de *kernel* definida para esse problema e essa função o resolve sem a necessidade de que seja conhecida a função de mapeamento $g(\mathbf{x})$ que leva os pontos de \mathbb{R}^2 para \mathbb{R}^3 .

Cabe ressaltar que as origens teóricas do uso e desenvolvimento dos métodos de *kernel* podem ser encontradas em Aronszajn (1950), onde *kernels* de reprodução foram desenvolvidos com base nos teoremas de Mercer (MERCER, 1909). Naquele trabalho, a função de *kernel* surge como uma medida de similaridade entre dados do espaço original, obtida a partir do produto interno das funções de mapeamento desse espaço em um espaço de características de alta dimensão. A partir desse desenvolvimento nos métodos de *kernel*, acabaram sendo criados vários tipos de *kernels* e alguns deles são muito interessantes devido às suas peculiaridades e acabam sendo bastante utilizados em várias aplicações, entre eles pode-se citar:

- *kernels* lineares: os quais são indicados quando um problema de classificação é linearmente separável no espaço original, ou seja, não há necessidade de mapear os dados do espaço original para um espaço de maior dimensão. Os *Kernels* lineares são matematicamente representados por:

$$\mathbf{K}_{i,j} = \mathbf{x}_i \bullet \mathbf{x}_j \tag{33}$$

- *kernels* polinomiais: São *kernels* lineares acrescentados da constante 1 e elevados ao expoente d , onde d é um número natural qualquer e a representação deste *kernel* é dada por:

$$\mathbf{K}_{i,j} = (\mathbf{x}_i \bullet \mathbf{x}_j + 1)^d \quad (34)$$

Quando o valor de d na equação acima é 1, há o *kernel* linear, acrescido de uma unidade. Para os demais casos, há outros *kernels* polinomiais, com características não lineares.

- *kernels* de base radial: A função *kernel* de base radial (*RBF – Radial Base Function*) é dada por:

$$\mathbf{K}_{i,j} = \exp\left(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2\right) \quad (35)$$

Onde γ é um valor positivo usado para controlar o raio de ação desse *kernel*. Essa função, por usar a distância euclidiana, não apresenta bons resultados quando aplicado a valores fora de padrão (*outliers*). Como forma de sanar essa deficiência foi apresentada uma versão robusta para o *Kernel* de base radial, usando a ideia de estatística robusta (CHEN, 2004).

- *Kernels* de Rede Neural de três camadas. Esse tipo de *Kernel* é expresso por:

$$\mathbf{K}_{i,j} = \frac{1}{1 + \exp\left(\nu(\mathbf{x}_i \bullet \mathbf{x}_j) - a\right)} \quad (36)$$

Onde ν e a são valores constantes. Essa função de *kernel* não satisfaz às condições de Mercer frequentemente, somente em alguns intervalos numéricos. Por essa razão foram feitas várias abordagens com o intuito de superar esse problema (SHAWE-TAYLOR e CRISTIANINI, 2004; ABE, 2005).

Além, dos *Kernels* apresentados acima, existem outros *kernels*, como o *kernels* de *Splines* e Malahalanobis. Existem também outros *kernels* que têm sido desenvolvidos para

aplicações específicas em processamento de imagem, classificação de textos e reconhecimento de fala (ABE, 2005; CAMPBELL, 2002).

Segundo Abe (2005), devido ao fato de que a correlação entre os *pixels* de uma imagem serem localizados, *kernels* de caráter global, como os polinomiais, não são apropriados para a classificação dessas imagens e, portanto, *kernels* de característica local têm sido estudados, dentre eles o *kernel RBF*. Portanto, segundo este autor deve-se esperar melhores resultados na classificação com *SVM* utilizando-se o *kernel RBF*.

A seguir, na próxima seção, será apresentado um estudo da arte na utilização de *SVM* para detecção de mudanças.

2.5 Estado da arte em detecção de mudanças usando *SVM*

Desde as últimas décadas, o uso de *SVM* em trabalhos de detecção de mudanças tem sido cada vez mais implementado. Os trabalhos a seguir mostram um pouco do que tem sido apresentado à comunidade científica com o uso de *SVM* nesta área.

Em Nemmour e Chibani (2006), é investigada a aplicabilidade de *SVM* para a detecção de mudanças de cobertura terrestre. A detecção de mudanças baseada em *SVM* é realizada para mapear o crescimento urbano na capital da Argélia. Indicadores de desempenho diferentes, assim como uma comparação com redes neurais artificiais, foram utilizados para mostrar a eficiência desse classificador. Em uma segunda etapa do processo, uma combinação de regras é proposta para melhorar a precisão de detecção de mudanças com a utilização de técnicas *fuzzy Integral* e *Attractor Dynamics*. Essa combinação foi implementada e avaliada em relação ao uso de *SVM*. Os resultados obtidos mostraram a eficiência do uso de *SVM* para detecção de mudanças em imagens de cobertura terrestre.

Os autores Bovolo et al. (2010), formulam o problema de separação de *pixels* representando as classes mudança e não-mudança, em imagens de sensoriamento remoto, como um problema de uma esfera mínima envolvente, tendo os *pixels* de mudança como classe alvo. A definição da esfera com volume mínimo que contem os *pixels* de não-mudança é obtida no contexto do classificador de *SVM*. Os resultados experimentais obtidos em imagens com características multitemporais e multiespectrais de sensoriamento remoto destacaram a eficácia do método proposto.

Em Habib et al. (2009) foram propostos quatro algoritmos diferentes para reduzir o número de *Support Vectors (SVs)* e, por conseguinte, o tempo de classificação. Os algoritmos foram testados para um problema de detecção de mudanças com base em um conjunto de

critérios genéricos de mudança, conjunto esse extraído de diferentes combinações de imagens de sensoriamento remoto. Os resultados mostraram que a metodologia proposta foi eficiente.

Em Steinwart (2010) foi apresentado o problema de detecção de mudança anômala como um problema de classificação binário e utilizou-se do classificador de *SVM* para construir um detector que não dependesse de suposições sobre a distribuição de dados. Para acelerar o processamento, o *SVM* foi implementado, em parte, em uma unidade de processamento gráfico. Resultados em dados contendo mudanças anômalas reais e simuladas foram utilizados para comparar o desempenho do método proposto em relação a algoritmos que efetivamente assumem uma distribuição Gaussiana dos dados.

Em Zanotta e Haertel (2012), um novo conceito de detecção de mudanças em imagens multitemporais de sensoriamento remoto é apresentado. A metodologia proposta baseia-se na estrutura *Bayesiana* e imagens-fração para classificar *pixels* de acordo com graus de pertinência para a classe mudança, segundo uma abordagem difusa. Para tanto, foram utilizadas experiências empregando imagens sintéticas, simulando alterações em imagens reais. O resultado apresentado mostrou que a metodologia proposta consegue expor adequadamente as mudanças graduais ocorridas entre duas datas analisadas.

Em Ferreira e Haertel (2013), foi realizada a investigação de uma abordagem semissupervisionada para detecção de mudança em imagens multitemporais de sensoriamento remoto aplicando *Support Vector Machines (SVM)*, com uma nova forma de obtenção de amostras de treinamento. Foi considerado que as classes mudança e não-mudança para aquele estudo apresentavam distribuições gaussianas. Dessa forma, o algoritmo de *Expectation-Maximization (EM)* foi utilizado para estimar os parâmetros estatísticos associados a essas duas distribuições. A partir desses resultados, amostras aleatórias foram extraídas daquelas distribuições e utilizadas para treinar um classificador *SVM*. A metodologia proposta foi testada e os resultados foram comparados a outros procedimentos, incluindo as observações de campo disponíveis, obtendo bons resultados.

Em Camps-Valls et al. (2008) é apresentado um quadro geral com base em métodos de *kernel* para a integração de fontes heterogêneas de informação. Com base nos princípios teóricos apresentados naquele quadro, três contribuições foram apresentadas. Primeiro, uma nova família de métodos baseados em *kernel* para a classificação multitemporal de imagens de sensoriamento remoto foi apresentada. A segunda contribuição foi o desenvolvimento de classificadores de *kernel* não-lineares para os métodos de detecção, como os de imagem-diferença e razão por estarem formulados em um espaço de característica de alta dimensionalidade adequada aos métodos segundo os autores. Por fim, a metodologia

apresentada permitiu a integração de informação e multisensores contextuais. O binário de *Support Vector (SV)* para o classificador SVM e *One-Class* foi avaliado com a utilização de funções de *kernel* linear e não linear. O Bom desempenho obtido na classificação realizada em imagens multitemporais sintéticas e reais, mostrou a capacidade de classificação dos algoritmos propostos naquele trabalho.

Em Bovolo et al. (2008) é apresentada uma nova abordagem, não supervisionada, para a detecção de mudanças em imagens multiespectrais de sensoriamento remoto. A abordagem proposta visava à extração da informação da mudança por meio da análise conjunta dos canais espectrais de imagens multitemporais no espaço característico original, sem quaisquer amostras de treinamento. Isso foi feito através da utilização de um limiar seletivo Bayesiano, para derivar um pseudo conjunto de amostras de treinamento, que foi necessário para a inicialização de um classificador SVM semissupervisionado. A partir daqueles valores iniciais, foi feita a detecção de mudanças no espaço original, considerando gradualmente padrões não marcados na definição da fronteira de decisão entre *pixels* alterados e inalterados de acordo com um algoritmo de aprendizagem semissupervisionado. Os valores dos parâmetros do classificador foram, então, definidos de acordo com uma nova técnica, baseada em um modelo não supervisionado, utilizando uma medida de similaridade entre os mapas de detecção de mudança obtidos com diferentes configurações. Os resultados experimentais obtidos em diferentes imagens multiespectrais de sensoriamento remoto confirmaram a eficácia da abordagem proposta.

2.6 Contribuição deste Trabalho

Conforme mencionado no início deste capítulo, o problema da detecção de mudanças em imagens da mesma cena, mas adquiridas em datas distintas, tem sido alvo de interesse em diversas áreas científicas, incluindo a área de sensoriamento remoto, como exposto no texto anterior. Ao longo deste capítulo, também ficou evidenciado que, nos últimos anos, classificadores baseados em SVM têm sido bastante utilizados para esse fim.

Embora alguns dos trabalhos abordem o uso de classificadores paramétricos bem conhecidos, como é o caso do classificador de Máxima Verossimilhança Gaussiana utilizado em Zanotta (2010) para a determinação da pertinência de *pixels* as classes mudança e não-mudança em imagens multitemporais, a maioria dos trabalhos fizeram uso do classificador SVM para a separação rígida dessas classes em imagens multitemporais. Entretanto, esta tese propõe uma abordagem não supervisionada para o problema de detecção de mudanças

utilizando o classificador *SVM* para, a partir de seus resultados, obter uma nova métrica de estimativa de pertinência dos *pixels* as classes, utilizando, para tanto, imagens-fração, as quais permitem que seja usado o *subpixel*, que além de eliminar a necessidade de normalização radiométrica das imagens favorece a determinação dos graus de pertinência dos *pixels*. Para validar a metodologia proposta serão utilizadas amostras de teste controladas obtidas a partir da técnica *CVA* e, além disso, os resultados de pertinência obtidos para imagem original serão comparados aos resultados de pertinência obtidos em (ZANOTTA, 2010) para a mesma imagem .

No próximo capítulo, a metodologia proposta para esse fim será apresentada de forma mais detalhada.

3 METODOLOGIA

O presente estudo visa apresentar uma métrica que permita estimar a pertinência de *pixels* a classes mudança e não-mudança em imagens multiespectrais e multitemporais de uma mesma cena. Para tanto, neste estudo serão utilizadas imagens-fração obtidas a partir destas imagens, uma vez que, essas proporcionam o uso do *subpixel*, o qual será de grande auxílio para verificar a pertinência dos *pixels* às classes já mencionadas. As imagens multitemporais de fração foram construídas a partir do uso do Modelo Linear de Mistura Espectral (MLME). Com a finalidade de classificação da imagem, será usado o classificador de *SVM* com os *kernels* Polinomial e RBF e a determinação da métrica proposta neste trabalho é obtida a partir dos resultados de classificação da imagem com esse classificador. Esses processos serão mais detalhados nas próximas subseções.

Para que os processos de classificação e determinação da métrica de pertinência acima citados sejam desenvolvidos, será seguida uma sequência de etapas com a finalidade de detectar as mudanças que porventura ocorram na cena em estudo. Assim, o processo de detecção de mudanças proposto neste estudo se desenvolve em 2 etapas:

- 1) Pré-processamento: nesta etapa as imagens em estudo são registradas e as imagens de fração são produzidas utilizando-se o MLME.
- 2) Análise dos dados: nesta etapa ocorre a produção das imagens de diferenças a partir da subtração das correspondentes frações nas imagens obtidas seguindo as indicações dadas em Zanotta e Haertel (2012). A seguir é feita a estimativa dos parâmetros da função densidade de probabilidade das classes mudança e não-mudança com o uso do algoritmo *Expectation-Maximization* (EM). Na sequência é feita a produção das amostras de treinamento das classes com base nos parâmetros estimados pelo algoritmo EM. Finalmente, será feita a classificação das imagens-fração com uso dos *kernels* polinomial e RBF aplicados ao classificador *SVM*, necessários para a obtenção da métrica, que será utilizada para estimar a pertinência dos *pixels* classificados às classes mudança e não-mudança.

As etapas mencionadas acima serão detalhadas nas próximas subseções desta tese.

3.1 Pré-Processamento

Vários trabalhos na literatura mencionam os efeitos das correções radiométricas e geométricas ao estudo da detecção de mudanças. Em particular, Zitová e Flusser (2003)

apresentam uma completa revisão de métodos para registro de imagens. Contudo, nas duas últimas décadas vários estudos abordaram questões muito importantes relativas ao processo de detecção de mudanças relativas aos erros no registro das imagens (DAI, 1998). Trabalhos como os de Bruzzone e Cossu (2003) investigaram maneiras para reduzir os efeitos de tais erros. Mais recentemente, Bovolo, Bruzzone e Marchesi (2007), estudaram o efeito dos erros de registro na classificação de imagens de alta resolução espacial.

A partir destes estudos fica determinado que para que haja um processo de detecção de mudanças eficaz é necessário que as diferenças em pares de imagens multitemporais, cobrindo a mesma cena de cobertura terrestre captadas pelos sensores remotos, sejam devidamente registradas e contenham apenas informações relativas às alterações de radiância nas imagens tomadas e não as causadas por outros fatores, tais como: diferentes condições atmosféricas e diferentes ângulos de incidência solar.

Sendo assim, torna-se necessário um pré-processamento que minimize os efeitos dessas variações indesejáveis, que podem atrapalhar o processo de detecção de mudanças. Portanto, nesta etapa, as imagens são registradas conforme preconizado na literatura e é empregado o MLME para produzir as imagens de fração que representam os componentes representativos da cena, que serão utilizados nas etapas seguintes, sem que seja necessária a correção radiométrica, uma vez que as amostras são tomadas de cada um dos conjuntos de imagens multiespectrais independentemente, como será melhor abordado na próxima subseção. A figura 12, a seguir, ilustra esta etapa.

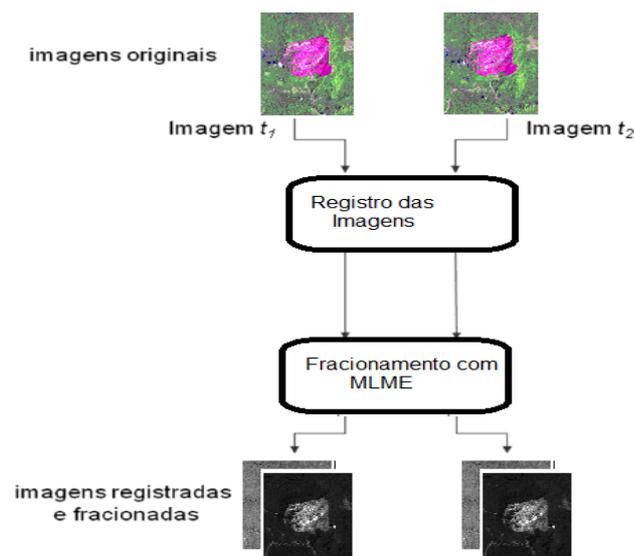


Figura 12 – Esquema geral do pré-processamento: as imagens X_1 e X_2 são registradas e, após, fracionadas nos seus componentes pelo MLME, gerando as imagens X'_1 e X'_2 registradas e fracionadas. Fonte: Adaptada de Zanotta (2010).

3.1.1 Modelo Linear de Mistura Espectral (MLME)

A maioria dos procedimentos utilizados para classificação de imagens digitais considera cada *pixel* da imagem como uma entidade “pura”, ou seja, a área correspondente a um *pixel* na superfície terrestre corresponde a uma única classe. Isso ocorre porque o que é registrado pelo sensor é o nível médio de radiância da área correspondente a um *pixel* na superfície terrestre. Entretanto, na maioria das regiões terrestres não existe uma única classe presente em cada *pixel*, mas uma combinação de classes.

A utilização do MLME, portanto, permite que seja considerada essa mistura de classes em cada *pixel*, a partir do uso de imagens-fração. O método apresenta vantagens em relação às abordagens tradicionais de detecção de mudanças. A primeira vantagem é a possibilidade de se trabalhar com alterações em nível de *subpixel*, o que permite a estimativa de graus de pertinência destes às classes em estudo. A segunda vantagem é que não há necessidade de correção radiométrica durante a fase de pré-processamento, pois diferentemente do que ocorre quando se trabalha diretamente com a resposta espectral, nesse método as imagens em cada tempo são analisadas independentemente (HAERTEL et al., 2004).

O método de fracionamento de imagens que será adotado neste trabalho emprega a abordagem descrita nos trabalhos de Haertel e Shimabukuro (2005). Naquele trabalho O MLME assume que a resposta espectral de cada *pixel* pode ser modelada por uma combinação linear das respostas espectrais de cada componente, ponderada pela fração da área que esse componente ocupa dentro do *pixel*, conforme mostrado na figura 13 abaixo.

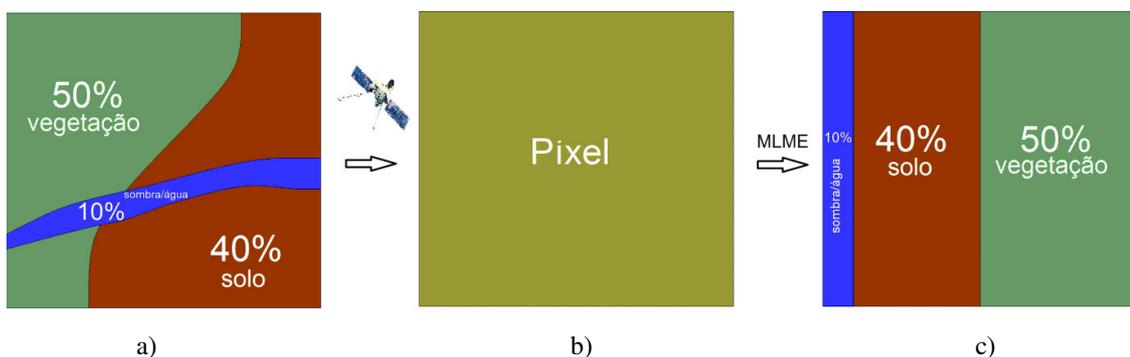


Figura 13 – Representação do Modelo Linear de Mistura Espectral (MLME): (a) descrição real das porções existentes na área delimitada pelo *pixel* (b) resposta espectral do sensor para a área correspondida pelo *pixel* e (c) resultado do fracionamento pelo MLME nos componentes de vegetação, solo e sombra/água.

Fonte: Adaptado de Zanotta (2010).

O MLME é representado por um sistema de p equações lineares e m frações desconhecidas, conforme a equação apresentada em (37):

$$R_k = \sum_{j=1}^m r_{j,k} f_j + e_k \quad (k = 1, \dots, p) \quad (37)$$

Onde:

- As entradas do vetor R_k representam as p respostas espectrais dos *pixels* para cada banda da imagem;
- As entradas $r_{j,k}$ correspondem às respostas espectrais da componente j na banda k ,
- f_j a fração do *pixel* ocupada pela componente j ;
- e_k representa o erro residual do modelo;
- p é o número de bandas espectrais em uso e m o número de componentes considerados.

Para um número de bandas maior que o número de componentes ($p > m$) tem-se um sistema do tipo sobredeterminado, o qual pode ser resolvido utilizando o Método dos Mínimos Quadrados, desde que sejam cumpridas as seguintes restrições dadas em (38) (SHIMABUKURO e SMITH, 1991):

$$0 \leq f_j \leq 1, \quad \forall j \quad \text{e} \quad \sum_{j=1}^m f_j = 1 \quad (38)$$

Os valores utilizados na equação devem ser selecionados de maneira a representar, da melhor forma possível, os componentes presentes na imagem em questão (*endmembers*). O resultado do fracionamento contará com um número de imagens igual ao número de componentes escolhidos para representar a área analisada.

3.2 Análise dos Dados

Nesta etapa serão desenvolvidos: os processos de produção das imagens de diferenças; a estimativa dos parâmetros da função densidade de probabilidade das classes mudança e não-mudança com o uso do algoritmo EM; a produção das amostras de treinamento e, finalmente, a classificação das imagens-fração com o uso dos *kernels* polinomial e RBF aplicados ao

classificador *SVM* que, a partir dos seus resultados, permitirá a obtenção da métrica de estimativa da pertinência dos *pixels* classificados às classes mudança e não-mudança.

A seguir será detalhado cada um dos passos utilizados nesta etapa.

3.2.1 Produção das Imagens de Diferença

As imagens de diferenças são produzidas a partir da subtração de componentes gerados pelo MLME das imagens multiespectrais e multitemporais.

Ao se utilizar imagens-fração para a obtenção de imagens de diferença, fica dispensada a etapa de normalização radiométrica, conforme já mencionado anteriormente, uma vez que o MLME é aplicado de forma independente para cada um dos conjuntos de imagens nas duas datas disponíveis.

Como preconizado em Zanotta e Haertel (2012), é comum que em cenas naturais sejam utilizados os componentes vegetação, solo e sombra/água. Contudo, neste trabalho serão utilizadas apenas os componentes vegetação e solo, descartando o componente sombra/água, uma vez que esse componente pode ser escrito como a combinação linear dos outros dois, exceto por uma pequena diferença devida ao erro residual do modelo presente no MLME. Dessa forma, para cada data serão utilizados apenas os conjuntos de imagens de fração de vegetação e imagens de fração de solo.

3.2.2 Análise das Imagens de Diferença

A figura 14 mostra a dispersão dos dados da imagem-diferença ao longo do eixo vertical correspondente às diferenças na fração de solo e ao eixo horizontal correspondente às diferenças na fração de vegetação.

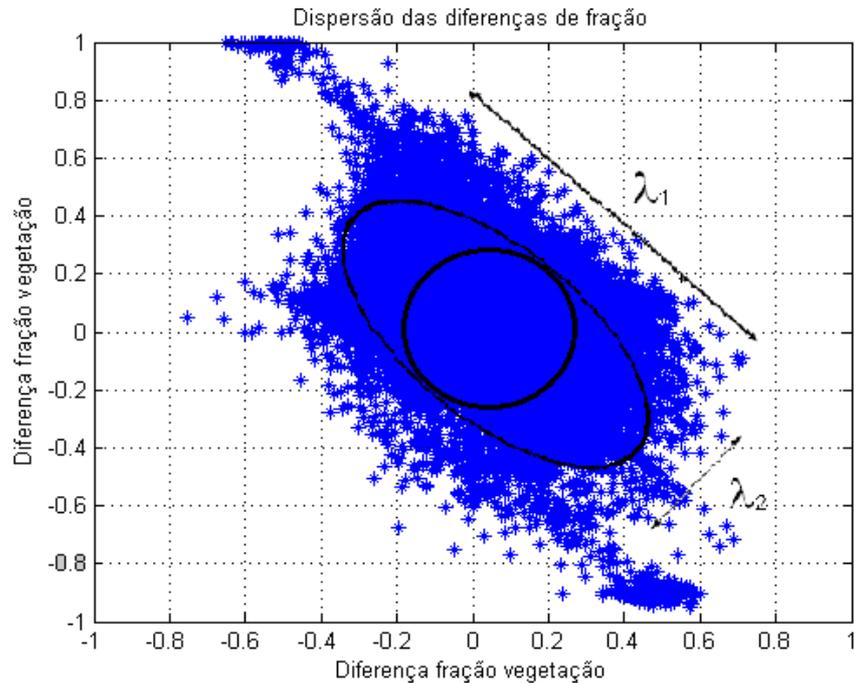


Figura 14 – Gráfico do espalhamento dos dados da imagem-diferença
 Fonte: Adaptado de Zanotta (2010).

Uma análise rápida na dispersão dos dados referentes à imagem de diferença permite verificar a existência de uma correlação negativa entre as diferenças das frações de vegetação e solo. Essa correlação pode ser explicada pelo fato de que um aumento na fração de solo implica na diminuição da fração vegetação e vice-versa, ou seja, o aumento na fração solo pressupõe que tenha ocorrido uma diminuição na fração vegetação deste *pixel* (desmatamento, por exemplo) e o contrário poderia ser explicado por reflorestamento ou crescimento de vegetação, por exemplo.

Outro fato a ser analisado e o qual, intuitivamente, deveria se esperar, é que a correlação negativa entre as duas variáveis apresentar-se-ia sob a forma de uma linha, passando pela origem dos eixos e atravessando o segundo e quarto quadrante deles (sem dispersão ortogonal a ela), pois idealmente supõe-se que as alterações na fração vegetação são causadas somente pela mudança na fração solo e vice-versa. Contudo, devido à ocorrência de ruídos (comuns a esse tipo de imagem) e também de pequenas distorções entre as duas variáveis pelo efeito do componente água/sombra, que embora tenha sido suprimido como componente da imagem-diferença, ainda acaba ocasionando uma dispersão ortogonal à linha idealizada (figura 14). Entretanto, a ocorrência dessa distorção não afeta o processo de detecção de mudança para a imagem e espera-se que os *pixels* que não sofreram alteração entre as duas datas (classe não-mudança) tendam a ocupar uma região próxima ao centro do

diagrama de dispersão (círculo da figura 14), enquanto *pixels* que sofreram alterações tendem a se deslocar para uma das duas extremidades da dispersão dos dados no diagrama, dependendo do tipo de mudança ocorrido nesses *pixels* (região de solo mudando para vegetação ou vice-versa).

Os dados de imagens diferença das frações vegetação e solo em imagens de sensoriamento remoto em geral tendem a apresentar um comportamento semelhante ao apresentado na figura 14. Nesta figura é apresentada somente a dispersão dos dados da imagem-diferença em duas dimensões (diferença na fração vegetação e diferença na fração solo). Entretanto, ao analisar o histograma dessa dispersão, esses dados apresentam um acúmulo de *pixels* nas proximidades da origem dos eixos, como pode ser ilustrado na figura 15, através de um modelo teórico do perfil da dispersão de dados, onde é apresentado um esboço dos perfis dos histogramas dos *pixels* referentes às classes mudança e não-mudança. Nesta figura pode ser vista a existência de uma sobreposição entre essas duas classes nas proximidades da origem dos eixos. Supõe-se, a partir da análise da figura 15, que ambos os perfis dos histogramas apresentam uma distribuição normal, bivariada (diferença na fração vegetação e diferença na fração solo). Além disso, o histograma da classe mudança apresenta uma dispersão grande com média próxima à origem. A classe não-mudança apresenta uma dispersão pequena, quase centrada na origem, o que corrobora a ideia intuitiva de que *pixels* sem mudança significativa devam estar próximos de zero, enquanto *pixels* representando mudança assumem poucos valores nesse intervalo.

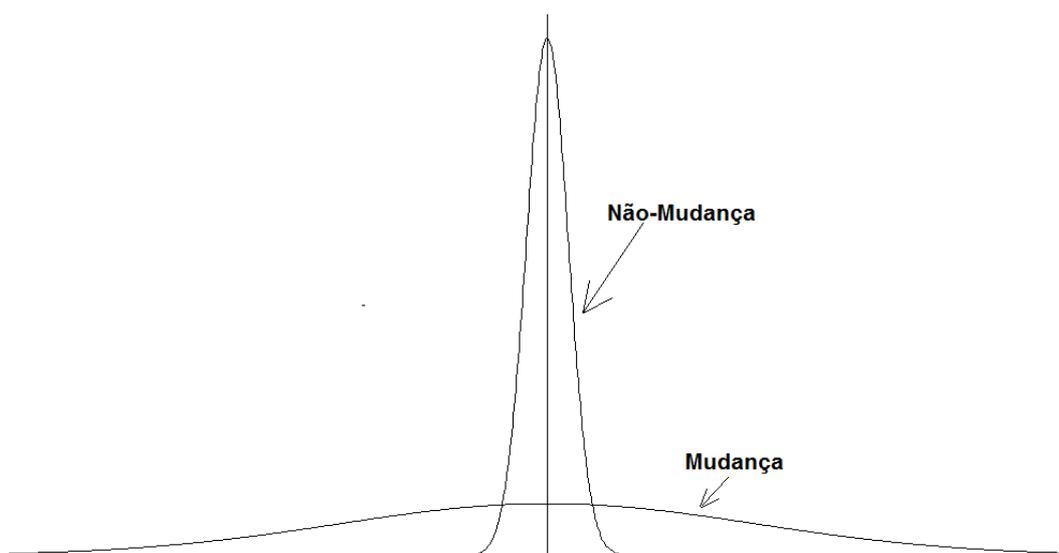
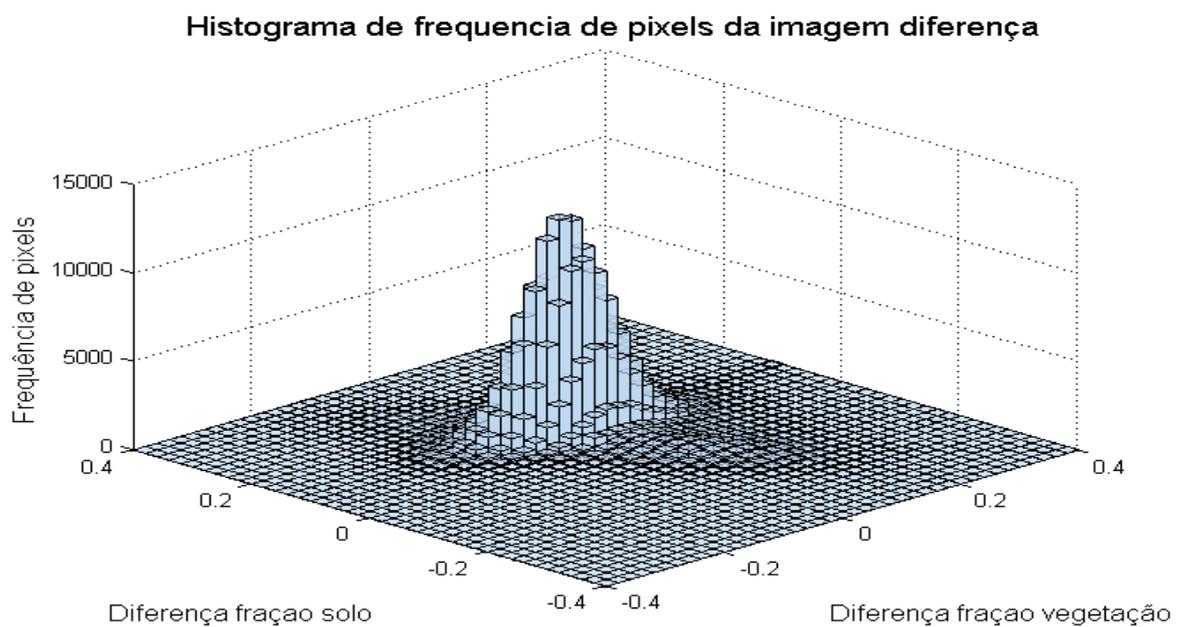
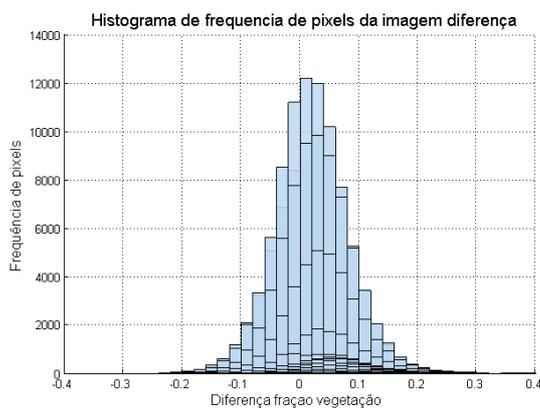


Figura 15 – Distribuições das classes mudança e não-mudança
Fonte: Adaptado de Bittencourt (2011).

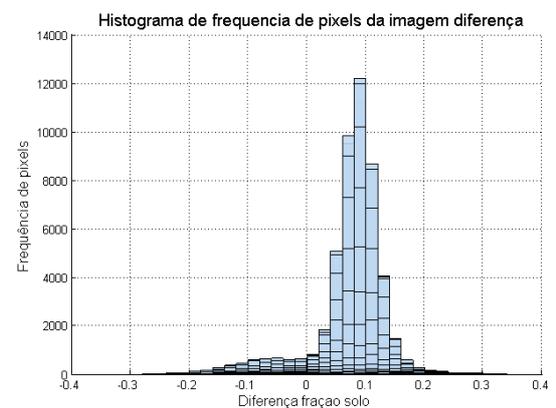
No caso em particular da imagem-diferença, que está sendo estudada neste trabalho, uma análise no histograma de frequência dos *pixels* desta imagem mostra que uma quantidade muito grande de *pixels* está próxima à origem dos eixos diferença fração vegetação e fração solo, porém, o centro deste agrupamento de *pixels* não está centrado na origem dos eixos, mas apresentando um pequeno deslocamento no sentido positivo do eixo diferença fração solo e praticamente centrada na origem em relação ao eixo diferença fração vegetação, isto é, centrado próximo ao valor 0 para o eixo diferença fração vegetação e 0,1 para o eixo diferença fração solo, como pode ser visto na figura 16, a seguir:



a)



b)



c)

Figura 16 – a) Histograma de frequência de *pixels* da imagem-diferença, b) perfil do Histograma de frequência dos *pixels* sobre o eixo diferença fração vegetação e c) perfil do Histograma de frequência dos *pixels* sobre o eixo diferença fração solo.

Fonte: Produzido pelo autor, 2014.

Portanto, da análise conjunta das figuras 14 e 15, corroborada pela figura 16, percebe-se que a classe mudança apresenta uma distribuição bivariada, alongada na direção de maior dispersão, a qual pode ser estimada pelo primeiro autovalor (λ_1) e sua direção é dada pelo autovetor correspondente. Esse autovalor pode ser interpretado como a variância associada ao conjunto das mudanças temporais, estimadas pelas diferenças entre as frações vegetação e solo, ou seja, a variância da classe mudança. Por outro lado, as amostras da classe não-mudança estão concentradas em torno da origem, apresentando uma pequena dispersão. Essa dispersão será estimada pela variância na direção ortogonal até a maior variação dos dados (direção do segundo autovetor da matriz de covariância total dos dados) e o valor dessa variância é estimado pelo autovalor dessa matriz de covariância (λ_2). Essa variância é causada pelos ruídos nos dados e por mudanças não significativas na imagem, isto é, pelas diferenças residuais causadas por influências da atmosfera, problemas no registro da imagem, iluminação, pequena variação radiométrica intrínseca da região, e pela fração água/sombra, entre outras. Portanto, a região correspondente à classe mudança tem a forma aproximada de uma elipse com eixo maior dado por λ_1 e eixo menor por λ_2 e a região na dispersão dos dados da imagem correspondente à classe não-mudança deverá estar contida numa região circular em torno da origem com um raio proporcional a λ_2 (ZANOTTA e HAERTEL, 2012).

Ainda em relação às distribuições dos dados nas classes, essas parecem apresentar um comportamento Normal (Gaussiano) e, portanto, podendo ser representadas por parâmetros estatísticos bem definidos. Assim, uma forma de estimar esses parâmetros é utilizar o algoritmo EM partindo das suposições intuitivas comentadas acima, as quais serão apresentadas com mais detalhes na próxima subseção.

3.2.2.1 O Algoritmo EM

O algoritmo EM surgiu da unificação de uma série de trabalhos, aparentemente sem relação, compilados e apresentados com esse nome por Dempster et al. (1977). Posteriormente a isso, esse algoritmo permaneceu por muito tempo esquecido. Contudo, nas últimas décadas passou a ser utilizado para determinação dos parâmetros estatísticos de uma distribuição de probabilidade, isto é, se alguma variável foi, algumas vezes, observada e outras não, nas ocasiões em que ela foi observada, essa observação pode ser utilizada para aprender a prever seus valores nas ocasiões em que não tenha sido observada. O algoritmo EM realiza essa tarefa, mas, também pode ser utilizado para variáveis cujos valores nunca

foram observados, desde que seja conhecida a forma geral da distribuição de probabilidade dessas variáveis (MITCHELL, 1997).

Dessa forma, esse algoritmo pode ser usado para fins de estimativa dos parâmetros (vetor de médias, matriz de covariâncias e probabilidade *a priori*) para as duas distribuições referentes às classes mudança e não-mudança (DUDA et al., 2001).

As estimativas de máxima verossimilhança geradas pelo algoritmo são obtidas iterativamente a partir de duas etapas: *expectation* e *maximization*. A etapa inicial do algoritmo (*expectation*) calcula uma expectativa do logaritmo da verossimilhança para as estimativas consideradas nesta etapa e, na etapa final (*maximization*), calcula os parâmetros que maximizam as expectativas geradas pelo logaritmo da verossimilhança calculado na etapa inicial. O algoritmo repete essas etapas realizando atualização nas estimativas até atingir a convergência.

Etapa 1:

$$\begin{aligned} \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) &= \frac{p(\mathbf{X}_k | \omega_i, \hat{\theta}_i) P(\omega_i)}{\sum_{j=1}^c p(\mathbf{X}_k | \omega_j, \hat{\theta}_j) P(\omega_j)} \\ &= \frac{|\hat{\Sigma}_i|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{X}_k - \hat{\mu}_i)^T \hat{\Sigma}_i^{-1} (\mathbf{X}_k - \hat{\mu}_i)\right] \hat{P}(\omega_i)}{\sum_{j=1}^c |\hat{\Sigma}_j|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{X}_k - \hat{\mu}_j)^T \hat{\Sigma}_j^{-1} (\mathbf{X}_k - \hat{\mu}_j)\right] \hat{P}(\omega_j)} \end{aligned} \quad (39)$$

Etapa 2:

$$\hat{P}(\omega_i) = \frac{1}{n} \sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) \quad (40)$$

$$\hat{\mu}_i = \frac{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) \mathbf{X}_k}{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta})} \quad (41)$$

$$\hat{\Sigma}_i = \frac{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) (\mathbf{X}_k - \hat{\mu}_i)(\mathbf{X}_k - \hat{\mu}_i)^T}{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta})} \quad (42)$$

Onde:

- o vetor θ_i representa os parâmetros da classe ω_i .
- c representa o número de classes, no caso deste estudo, $c = 2$.
- μ_i é a média da classe i . Será utilizada nesta tese a notação M_1 e M_2 para as médias de cada classe.
- Σ_i é a matriz de covariância da classe i . Será utilizada nesta tese a notação S_1 e S_2 para as matrizes de covariância de cada classe.

3.2.2.2 Estimativa inicial dos parâmetros para uso no algoritmo EM

Conforme observado anteriormente, a partir da análise dos dados em estudo, esses dados parecem ser normalmente distribuídos. Dessa forma, partindo de uma distribuição conhecida pode-se usar o algoritmo EM para estimar os parâmetros estatísticos dessas distribuições.

Assim, baseado nas características dos dados apresentados nas figuras 14 e 15 o conjunto das amostras que deve ser empregado para uma representação inicial dos parâmetros da classe não-mudança pode ser assumido como sendo formado pelas amostras em torno da origem (*pixels* com alteração nula ou insignificante), com dispersão contida em uma região quase circular centrada próxima da origem dos eixos (círculo da Figura 14). Razão pela qual a dispersão dessas amostras pode ser estimada pelo segundo autovalor da matriz de correlação dos dados da cena em estudo (λ_2) (Figura 14), ou seja, pode-se iniciar o processo de estimativa de parâmetros dessa classe considerando o vetor de médias nulo e assumindo como estimativa inicial para a matriz de covariância associada à classe da matriz diagonal com covariâncias nulas e variâncias iguais ao segundo autovalor estimado para o conjunto total dos dados. De forma semelhante, para as amostras da classe mudança podem ser utilizados como parâmetros (vetor de médias e matriz de covariância) os valores estimados a partir do conjunto total das amostras disponíveis (elipse da Figura 14).

Supondo que o número de *pixels* não apresenta mudança significativa, é muito maior do que o número de *pixels* que apresentam mudanças; os valores iniciais das probabilidades *a priori* podem ser estimados levando-se em conta esta diferença na quantidade de *pixels*. Assim, uma escolha inicial de valores para as probabilidades *a priori* podem ser 0,1 para classe mudança e 0,9 para a classe não-mudança. Essa escolha inicial de probabilidade é empírica e quanto mais próxima do valor real ela estiver, mais rapidamente o algoritmo

convergir. Portanto, o seu valor inicial, assim como dos outros parâmetros, não é preponderante (DEMPSTER et al., 1977).

Para efeito de notação, serão consideradas as classes ω_1 e ω_2 , respectivamente, como as classes mudança e não-mudança. E, portanto, os valores de M_1 , S_1 , $P(\omega_1)$, M_2 , S_2 , $P(\omega_2)$ representam os vetores de médias, matrizes de covariâncias e probabilidades *a priori* de cada classe. Assim, as estimativas iniciais para cada uma das classes é dada por:

$$M_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_1 = \begin{bmatrix} \text{matriz de} \\ \text{covariância} \\ \text{da amostra total} \end{bmatrix}, P(\omega_1) = 0,1, M_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_2 = \begin{bmatrix} \lambda_2 & 0 \\ 0 & \lambda_2 \end{bmatrix} \text{ e } P(\omega_2) = 0,9 \quad (43)$$

Partindo das estimativas iniciais apresentadas acima, o algoritmo EM estabelece os parâmetros que serão utilizados para representar a distribuição das classes mudança e não-mudança. Esses parâmetros, a seguir, serão utilizados na obtenção de amostras de treinamento a serem usadas no classificador de *SVM*, que fará a classificação da imagem-diferença e, conseqüentemente, propiciará a geração da métrica para estimativa da pertinência de *pixels em relação as classes*. Esse processo de seleção de amostras de treinamento será apresentado na próxima subseção.

3.2.3 Seleção das amostras de treinamento

Com a finalidade de treinar o classificador *SVM* devem ser utilizadas amostras randômicas extraídas a partir de uma distribuição normal multivariada, obtidas por meio da função *mvnrnd.m*, presente na biblioteca de funções do *Matlab*, usando como parâmetros da distribuição os valores estimados pelo algoritmo EM. Contudo, levando em consideração que as distribuições de probabilidade para as classes mudança e não-mudança se sobrepõem à região próxima da origem dos eixos diferença fração vegetação e solo (figura 15), pode-se utilizar os mesmos parâmetros das distribuições normais multivariadas de cada classe com a finalidade de se determinar dentre as amostras aleatórias geradas, quais delas probabilisticamente pertencem à classe indicada na geração de amostras, isto é, são comparadas as probabilidades de cada amostra gerada pela função *mvnrnd* nas duas distribuições de probabilidade obtidas a partir do algoritmo EM e, então, aquela que apresentar o maior valor em uma das distribuições é rotulada na classe correspondente àquela distribuição que apresentou o maior valor. Embora existam amostras da classe mudança nessa região de sobreposição é mais provável que uma amostra aleatoriamente gerada pertença à classe não-mudança. Sendo assim, esse procedimento visa garantir, estatisticamente, que as

amostras geradas aleatoriamente na impossibilidade de garantir sua pertinência a uma ou outra classe possam representar com pouca chance de erro, de fato, uma das classes, esse procedimento é necessário, pois embora o classificador *SVM* seja um classificador robusto e permita uma boa separação entre as classes, mesmo com a presença de alguma amostra de treinamento incorreta, no caso de classificação com *SVM* utilizando pertinência de *pixels*, o uso de amostras de treinamento mais confiáveis devem gerar resultados mais acurados.

3.2.4 Classificador SVM

A classificação da imagem-diferença prevista nesta tese pressupõe o uso do classificador não paramétrico *SVM* na sua forma dual com margem suave. A base teórica deste classificador está contida em Abe (2005) e elas foram apresentadas no capítulo 2 desta tese. Para essa classificação serão utilizados os *kernels* polinomial e RBF, também descritos naquele capítulo. As amostras de treinamento do classificador serão utilizadas como exposto na subseção anterior. Sobre o seu uso, tenta-se evitar alguns erros na classificação da imagem e, portanto, uma melhor qualidade na determinação da métrica para estimativa de pertinência dos *pixels* em relação às classes, como será mais detalhadamente trabalhado na próxima subseção.

3.2.5 Determinação de métrica para estimativa de pertinência de pixels as classes

Ao longo do processo de classificação por *SVM* os *pixels* da imagem diferença são submetidos aos *kernels* polinomial e RBF. Estes *kernels* simulam, através do truque de *kernel* tratado no capítulo 2, o mapeamento que seria feito por uma função aplicada aos dados originais transportando-os do espaço original para um espaço de dimensão superior, onde as classes mudança e não-mudança tornar-se-iam linearmente separáveis. O uso do *kernel* permite que essa separação linear existente entre as classes no espaço de dimensão mais elevada seja feita diretamente na imagem original sem a necessidade de mapeamento. Uma métrica para estimar a pertinência de *pixels* às classes mudança e não-mudança poderia ser obtida através da distância de cada *pixel* mapeado neste espaço de dimensão superior ao hiperplano de separação ótimo que separa essas classes. Dessa forma, *pixels* mais afastados ao hiperplano de separação ótimo teriam uma probabilidade maior de pertencerem a classe em que estão inseridos e *pixels* mais próximos a este hiperplano apresentariam uma probabilidade menor de pertencer a referida classe. Contudo, cabe ressaltar que a determinação desta medida

diretamente no espaço mapeado é muito complexa ou até inviável, uma vez que certos espaços mapeados apresentam um número muito elevado de dimensões (*kernel* RBF) e, também porque, a função de mapeamento equivalente ao *kernel* utilizado no classificador pode não ser conhecida (*kernel* RBF). Dessa forma, a ideia inovadora apresentada nesta tese é estimar a distância dos *pixels* ao hiperplano de separação ótimo diretamente na imagem original utilizando a potencialidade do truque de *kernel* de operar a separação das classes diretamente neste espaço.

A medida de estimação da distância do *pixel* mapeado ao hiperplano de separação ótimo, que também passa a ser considerada como uma estimativa de pertinência do pixel a uma das classes (mudança e não-mudança) é doravante denominada de Delta b (Δb). Esta medida corresponde ao vetor dado pela diferença entre o valor da função de decisão gerada pelo classificador SVM em um *pixel* qualquer da imagem e o valor da função de decisão da projeção deste ponto sobre o hiperplano de separação ótimo, segundo uma direção paralela a direção determinada pelo eixo correspondente a variável dependente da função de decisão, conforme pode ser mostrado na Figura 17, a seguir. Embora essa medida aplicada entre o *pixel* classificado e hiperplano não seja ortogonal a ele e, portanto, não a caracterize como uma distância, ela mantém sempre a mesma angulação β entre o vetor normal (\mathbf{w}) e o vetor Delta b para qualquer *pixel* da imagem, que corresponde ao mesmo ângulo entre o vetor \mathbf{w} e o eixo da variável dependente que define a função de decisão do classificador. Dessa forma, o vetor Delta b ao ser projetado sobre o vetor normal, gerará, de fato, uma medida de distância do *pixel* classificado ao hiperplano ótimo. Portanto, a magnitude do vetor Delta_b, é uma medida proporcional a distância do *pixel* mapeado ao hiperplano de separação ótimo. Sendo assim, pode-se usar o resultado da classificação por SVM, o qual corresponde a extremidade do vetor Delta b para estimar a pertinência de qualquer *pixel* a sua respectiva classe, uma vez que a magnitude deste vetor (métrica Delta b) determina a proximidade ou afastamento dos *pixels* ao hiperplano de separação e o sinal do vetor Delta b corresponde ao sinal do resultado do classificador e portanto se refere a medida referente a classe +1 (classe mudança) e -1 classe não-mudança.

Na Figura 17, a seguir, é apresentada uma visão do plano x_1x_3 do espaço \mathbb{R}^3 (espaço mapeado ou característico), onde é mostrado um *pixel* $\mathbf{x}_a=(x_1, x_2, x_3)$ que foi mapeado do espaço original (imagem diferença) para este espaço. Este ponto está inserido em um plano determinado pela função de decisão $D(\mathbf{x}_a)=b_a$ (plano em vermelho), onde b_a corresponde ao valor da função de decisão para o ponto \mathbf{x}_a . Este plano é paralelo ao plano de separação ótimo

entre as classes mudança e não-mudança, o qual é caracterizado pela função de decisão $D(\mathbf{x})=0$ (plano em cinza). A projeção do ponto a paralelamente ao eixo x_3 sobre o plano de separação ótima determina o ponto a' . A diferença entre os valores da função de decisão para estes pontos define o vetor Δb . O ângulo determinado entre este vetor e o vetor normal ao hiperplano de separação ótima (vetor \mathbf{w}) é β .

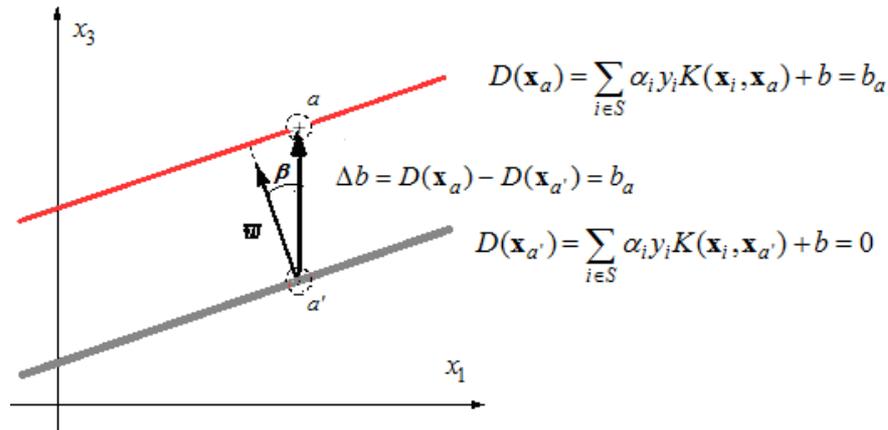


Figura 17 – Visualização da métrica de pertinência Delta b.
 Fonte: Produzido pelo autor, 2014.

Deve-se enfatizar que essa métrica não estima um valor percentual de mudança do *pixel* ao longo do tempo, nem mesmo tem um significado estatístico específico, inclusive em Platt, (2000) é afirmado que SVM produz como resultado um valor não calibrado que não é uma probabilidade. Assim, a métrica apenas atribui um valor de pertinência ao *pixel* baseado nos valores obtidos pela distância deste *pixel* mapeado ao hiperplano de separação, através da métrica Delta_b, a qual pode ser também obtida diretamente na função de decisão do classificador SVM com o uso de um *kernel* específico na imagem-diferença. Essa métrica, portanto, estima o grau de confiança com que um *pixel* pode ser rotulado em uma determinada classe. Levando-se em conta que existem apenas duas classes complementares em detecção de mudanças (mudança e não-mudança), pode-se usar apenas a pertinência sobre uma delas, no caso, aquela que se refere às classes mudança, pois é possível afirmar que um *pixel* tendo alta pertinência em uma das classes significa que ele tem baixa pertinência na outra e vice-versa. Sendo assim, essa métrica permitirá a estimativa da pertinência de um *pixel* classificado à classe mudança somente pelo valor de Delta b. O valor de pertinência representado por $P_m(\mathbf{x})$ para a classe mudança será dado pela proporção entre o maior valor de Delta e menor valor de delta b, conforme equação (44), a seguir:

$$P_m(\mathbf{x}) = \begin{cases} \frac{\Delta b}{\max(\Delta b)}, & \text{se } \Delta b > 0 \\ 0.5, & \text{se } \Delta b = 0 \\ \left(1 - \frac{\Delta b}{\min(\Delta b)}\right), & \text{se } \Delta b < 0 \end{cases} \quad (44)$$

e a pertinência da classe não-mudança $P_{nm}(\mathbf{x})$ é dada pela equação (45):

$$P_{nm}(x) = 1 - P_m(x) \quad (45)$$

A partir das equações acima, percebe-se que o menor valor de Delta b obtido após a classificação corresponderá a 0 (0% de pertinência a classe) e o maior valor de Delta b corresponderá a 1 (100% de pertinência a classe), isto é, *pixels* com valores de Delta b zero, ou próximos de zero, indicam pouca probabilidade de pertencerem à classe mudança e, portanto, possuem alta probabilidade de pertencerem à classe não-mudança, em como *pixels* com valor 1 ou próximos de 1 indicarão grande probabilidade de pertencerem à classe mudança e dessa forma, baixa probabilidade de pertencerem à classe não-mudança,

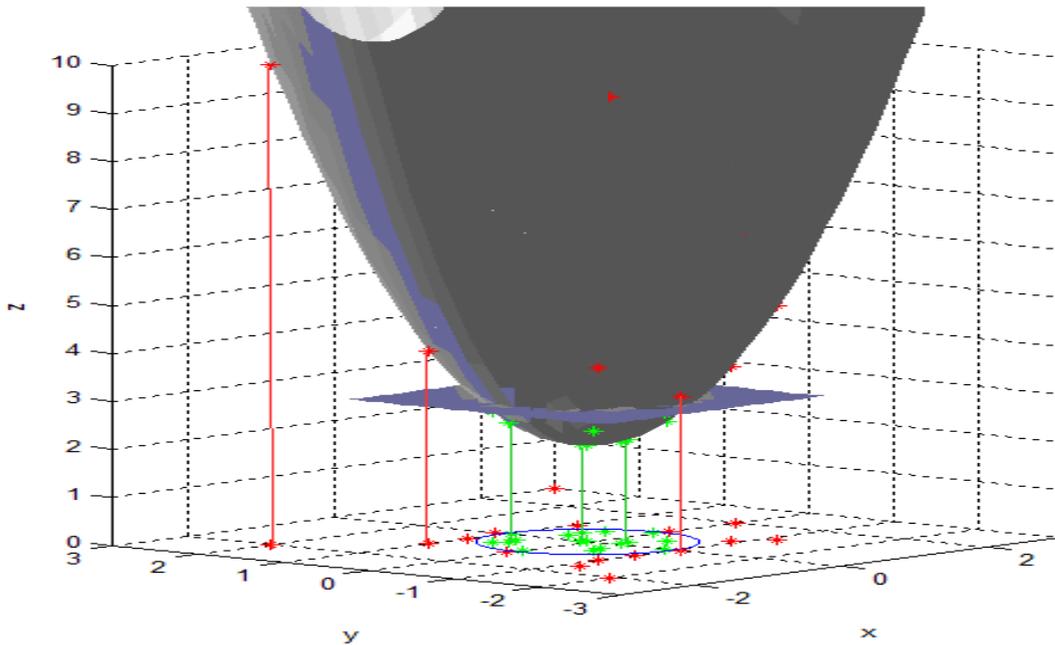


Figura 18 – Correspondência entre os pontos mapeados distantes do hiperplano de separação e sua pertinência na classe onde está inserido

Fonte: Produzido pelo autor, 2014.

Neste momento, é necessário afirmar que a determinação dessa métrica é dependente da escolha do *kernel* utilizado e dos parâmetros a ele atribuídos durante o processo de classificação com *SVM*. A figura 18, acima, ilustra, através de um exemplo simples de mapeamento de dados, a relação existente entre a distância de um *pixel* mapeado em um

espaço de maior dimensão e o hiperplano de separação entre as classes nesse espaço e a pertinência desse mesmo *pixel* no espaço original em relação à classe onde está inserido.

Note que no plano XY da figura 18 existem duas classes não linearmente separáveis: classe 1 (vermelho) e classe 2 (verde). Os dados desse plano são mapeados em um novo espaço, XYZ, de maior dimensão que o espaço original XY. Neste novo espaço percebe-se que o plano paralelo ao plano XY, passando por $z=3$ separa linearmente os pontos mapeados. Nota-se também, que os pontos mapeados mais afastados do hiperplano correspondem a pontos no espaço original mais inserido dentro da classe a qual pertencem, demonstrando por maior pertinência a essa classe, ou seja, maior chance de pertencer a ela, enquanto pontos mapeados mais próximos do hiperplano de separação correspondem a pontos no espaço original mais próximos da fronteira de separação entre classes e, portanto, de maneira semelhante ao exposto acima, com menor pertinência à sua classe, isto é, menor chance de pertencer a ela.

3.3 Medidas de Avaliação da Acurácia da metodologia

Levando em consideração que pode haver falta de informações sobre a verdade terrestre da imagem que será analisada, inclusive não havendo dados de campo para validar a classificação dela, este trabalho utilizou um conjunto de amostras de teste, gerado aleatoriamente, e distribuído uniformemente ao longo da imagem-diferença de fração, segundo critérios que serão explicados a seguir. Para garantir a confiabilidade dessas amostras, será utilizado o módulo do vetor de mudança preconizado na técnica CVA (apresentada no capítulo 2). Segundo essa técnica, será gerado, para cada *pixel* da imagem, um vetor, tendo por origem o *pixel* da imagem tomado na primeira data de aquisição (1991) e como extremidade o *pixel* correspondente a ele na segunda data de aquisição (1994). Embora essa técnica não permita separar as classes, pois não é conhecido o limiar de separação entre elas, a utilização do módulo do vetor mudança para *pixels* onde os valores de módulo sejam específicos permitirá a estimativa da mudança que, eventualmente, venha a ocorrer ou não nesses *pixels*. Ou seja, *pixels* apresentando módulo de vetor de mudança com valores superiores a 0,3 e menores que 0,6 corresponderam a *pixels* que apresentaram algum tipo de mudança significativa ao longo desse período. Portanto, pertencentes à classe mudança e *pixels* com módulo do vetor de mudança próximo de zero, ou seja, valores menores que 0,1, por exemplo, corresponderiam a *pixels* que não apresentaram mudança significativa ao longo desse período e, portanto, são pertencentes à classe não-mudança. A escolha desses intervalos

visa a garantir que os *pixels* amostrados em uma determinada classe tenham alta probabilidade de pertencer a ela. Cabe aqui comentar a limitação superior no intervalo do módulo do vetor mudança, referente à classe mudança (neste caso 0,6), porque valores elevados do módulo do vetor mudança poderiam se relacionar à presença de nuvens, as quais poderiam estar presentes na imagem em questão e que, certamente, seriam *pixels* de mudança, mas que mascarariam mudanças ou não nos *endmembers* vegetação e solo.

Com a utilização da métrica de pertinência do módulo do vetor mudança foram coletados 900 *pixels* de amostra de teste para cada classe. Selecionando as amostras de testes dessa forma, obtiveram-se amostras representativas das duas classes em estudo (classe mudança e não-mudança), proporcionando dados confiáveis para comprovar estatisticamente a validade da metodologia proposta.

No próximo capítulo serão descritos os experimentos realizados com a finalidade de testar a metodologia proposta neste capítulo.

4 TESTES E EXPERIMENTOS

4.1 Introdução

Neste capítulo serão descritos e discutidos os materiais e os experimentos que têm por finalidade a validação da metodologia proposta no capítulo anterior. Para todos os experimentos serão utilizadas imagens *Landsat 5-TM* da mesma região, mas adquiridas em datas distintas. A primeira imagem (data 1) adquirida em Outubro de 1991 e uma segunda imagem (data 2) adquirida em Novembro de 1994. Os detalhes dessas imagens serão apresentados neste capítulo. Os algoritmos utilizados na implementação dessa metodologia foram desenvolvidos em ambiente *Matlab*. As imagens e mapas de mudança foram visualizados utilizando tanto o software *Matlab*, quanto o *MultiSpec* (LANDGREBE e BIEHL, 2002). A metodologia proposta será testada em duas etapas. Na primeira etapa será utilizado um conjunto de amostras de testes que representa as classes mudança e não-mudança presentes na imagem real utilizada neste trabalho e que será, a seguir, melhor detalhada. Essas amostras foram obtidas considerando o valor do Módulo *CVA* de cada *pixel* correspondente às amostras, ou seja, *pixels* com valores entre 0,3 e 0,6 foram considerados como pertencentes à classe mudança e os com valores 0 e 0,1 como pertencentes à classe não-mudança. Essas amostras têm a finalidade de evidenciar, de forma estatística, a efetiva capacidade da metodologia proposta em estimar a pertinência de *pixels* às classes em estudo, uma vez que não é conhecida a verdade terrestre da imagem utilizada, nem estão disponíveis amostras coletadas na região. Finalmente, para validar esta métrica de estimativa de pertinência de *pixels* proposta, será realizada uma comparação dos resultados obtidos por ela com os obtidos para as mesmas imagens pela metodologia de (ZANOTTA, 2010). Naquele trabalho foi proposta a determinação de uma métrica de pertinência a partir do uso do classificador MVG, seguido de uma abordagem com lógica *fuzzy*, na qual foram estimados os graus de pertinência à classe mudança para uma imagem-diferença entre frações vegetação e solo. Na proposta daquele trabalho também foi adotada uma abordagem em nível de *subpixel* na estimativa dos graus de pertinência para cada *pixel*, usando o MLME seguido de uma etapa posterior, onde foi feita a introdução de um ajustamento adicional por meio de informações de contexto espacial aos graus de pertinência estimados para melhorar os resultados. Dessa forma, a comparação prevista é bem razoável, pois a diferença nas pertinências obtidas difere somente na metodologia aplicada.

Após a comparação entre as metodologias, espera-se que exista uma correlação significativa entre o valor da pertinência apresentada naquele trabalho e à métrica de estimativa de pertinência Δ_b , proposta nesta tese.

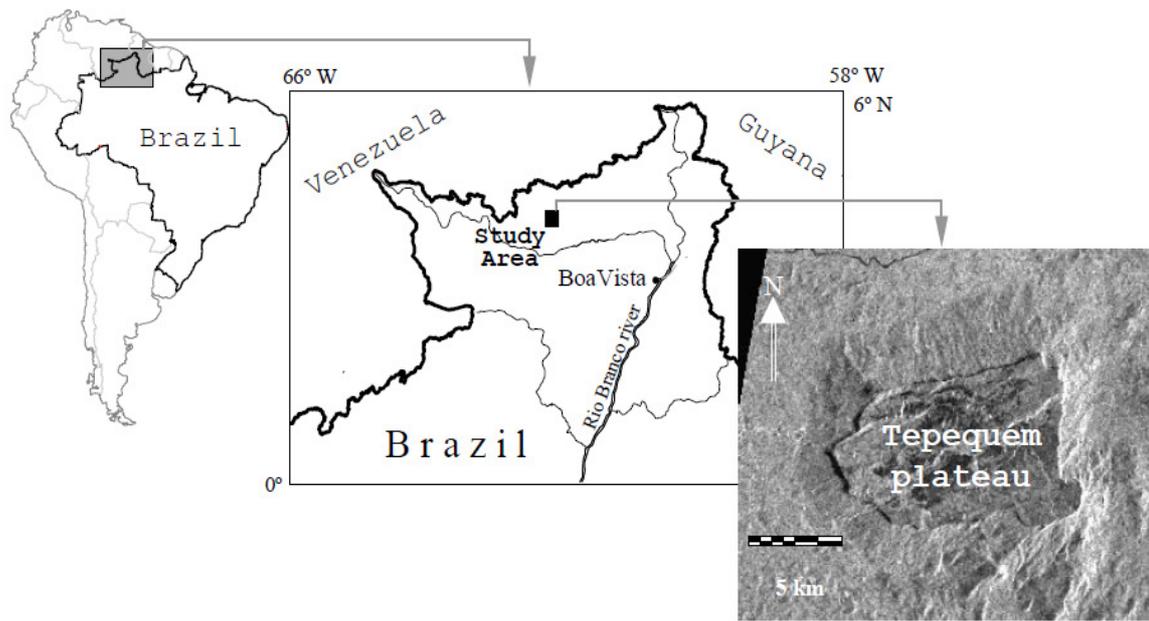
A seguir são descritas as imagens utilizadas neste capítulo.

4.2 Descrição das Imagens

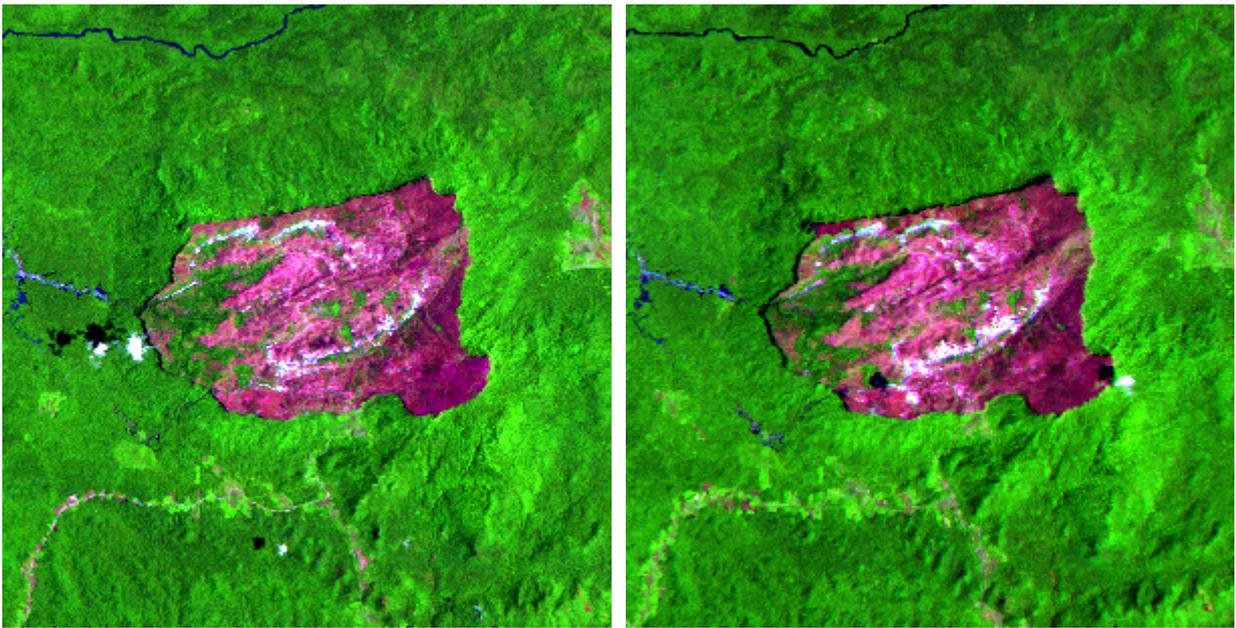
Para testar a metodologia empregada neste estudo foram utilizadas imagens reais coletadas pelo satélite *Landsat 5*, sensor TM (*Thematic Mapper*). Esse satélite, assim como seus antecessores, segue uma órbita quasi-polar e heliosíncrona, numa altitude de cerca de 710 km acima da superfície terrestre e com resolução temporal de 16 dias. O sensor presente no satélite possui sete bandas espectrais, sendo três delas na região do visível (canal 1: 0,45-0,52; canal 2: 0,52-0,60; canal 3: 0,63-0,69), uma no infravermelho próximo (canal 4: 0,76-0,90), duas na região do infravermelho médio (canal 5: 1,55-1,75; canal 7: 2,08-2,35) e uma na região do infravermelho termal (10,40-12,50). Seus canais possuem resolução espacial de 30 m, exceto o canal termal, que tem uma resolução de 120 m.

Para os experimentos propostos neste capítulo foram consideradas duas imagens *Landsat 5-TM* cobrindo a mesma cena terrestre em duas datas distintas. Essas imagens estão localizadas nas coordenadas 61°37'W–61°49'W de longitude e 3°40'N–3°52'N de latitude dentro do estado de Roraima. A razão da escolha dessas imagens se deve, em parte, por sua disponibilidade, mas também devido às suas características e, finalmente, pela possibilidade que oferece de comparar a métrica de pertinência, proposta neste trabalho, com a métrica obtida em (ZANOTTA, 2010). A imagem escolhida apresenta duas áreas distintas, uma planície localizada em uma altitude média de 200 m e outra localizada numa alta elevação, conhecida como Platô do Tepequém. A região de baixa elevação é coberta por uma floresta densa, enquanto a região do platô é dominada por uma vegetação de cerrado (ZANOTTA e HAERTEL, 2012).

As imagens utilizadas possuem um total de 390.625 *pixels* distribuídos em um quadrado de 625 por 625 *pixels*. A Figura 19, a seguir, ilustra a localização e as respectivas imagens tomadas em datas distintas.



(a)



(b)

(c)

Figura 19 – (a) Área de estudo; (b) Imagem de Data 1, composição em falsa cor 5(R), 4(G), 3(B) e (c) Imagem de Data 2, composição em falsa cor 5(R), 4(G), 3(B).

Fonte: Haertel et al. (2004).

Na próxima seção é descrito o processo de produção das imagens-fração, que serão utilizadas nos experimentos.

4.3 Produção das Imagens de Fração

Conforme mencionado na metodologia, as imagens de diferenças são produzidas a partir da subtração dos componentes das imagens de fração geradas a partir de duas imagens: multiespectrais e multitemporais da mesma cena. O método utilizado para estimar as frações dos componentes de mistura (*endmembers*) de cada *pixel* da imagem foi o Modelo Linear de Mistura Espectral (MLME).

Para o fracionamento das imagens, foram utilizadas as seis bandas reflectivas disponibilizadas pelo sistema sensor *Landsat 5-TM* e, a partir do MLME, foram geradas três imagens correspondentes às frações vegetação, solo e água/sombra. No entanto, conforme mencionado na metodologia desta tese, somente foram consideradas as frações solo e vegetação para efeitos de classificação e determinação da métrica de pertinência, uma vez que o objetivo deste estudo é analisar especificamente a alteração entre os *endmembers* fração vegetação e fração solo. Esta escolha de frações foi feita em função de ambas apresentarem boa correlação negativa entre si e por se considerar que as informações contidas no componente água/sombra não apresentam uma contribuição significativa na detecção de mudanças destes *endmembers*, podendo, inclusive, gerar informações equivocadas, uma vez que a variação anual no ângulo de elevação solar em uma mesma região, e na hora do dia em que foi adquirida, pode gerar distorções que não podem ser utilizadas para afirmar a ocorrência de alterações na cobertura do solo. A Figura 20 mostra as imagens-fração para os *endmembers* vegetação, solo e sombra/água, geradas pelo MLME para a imagem referente ao ano de 1991 (data 1), de forma semelhante à imagem-fração gerada para o ano de 1994 (data 2).

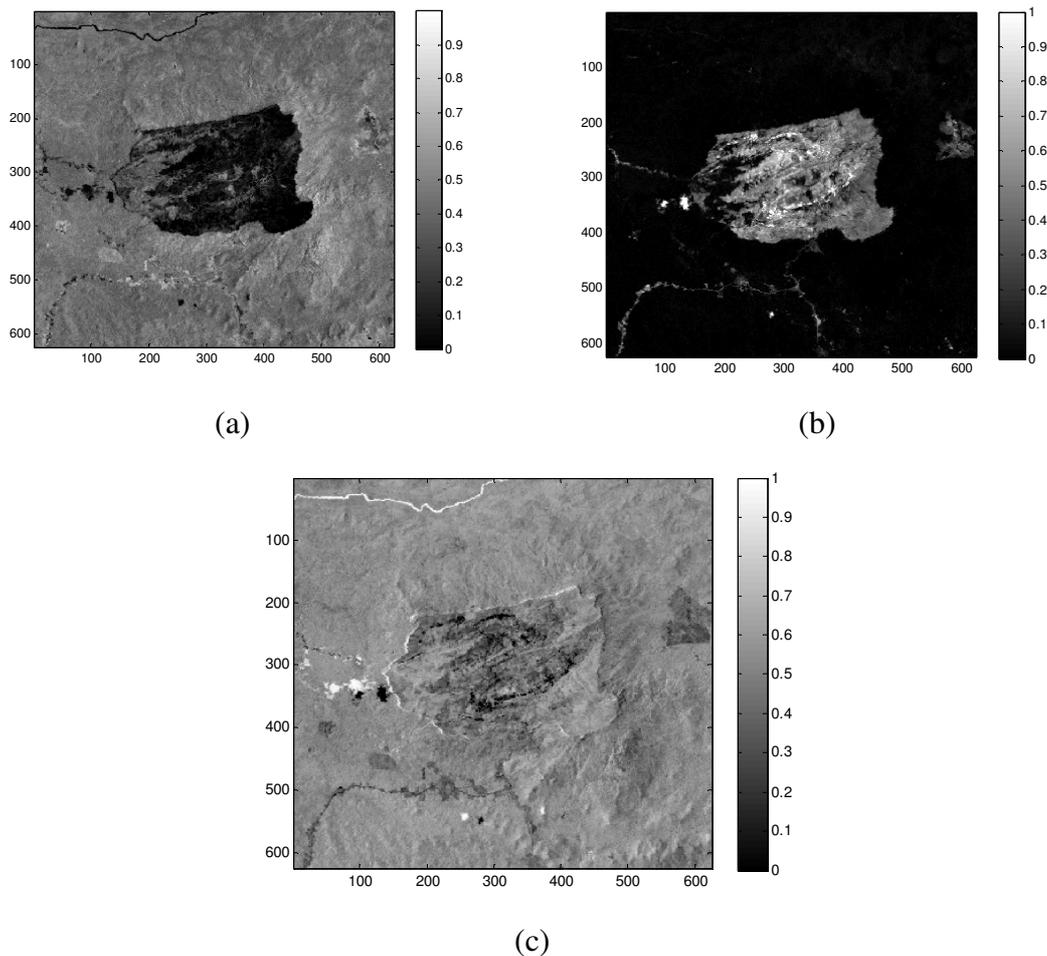


Figura 20 – Imagens-fração da imagem de 1991 para (a) vegetação (b) solo e (c) sombra. Variação contínua de preto (0%) a branco (100%).

Fonte: Adaptada de Haertel et al. (2004).

Sendo assim, a partir das imagens-fração correspondentes aos *endmembers* vegetação e solo, é construída uma imagem de diferença. A próxima seção mostrará como essa imagem-diferença fração é construída.

4.4 Produção da Imagem de Diferença

Para os testes realizados neste capítulo, a imagem de diferença foi obtida pela subtração *pixel a pixel* entre a imagem-fração da data 2 (1994) e a imagem-fração da data 1 (1991).

A Figura 21 mostra a imagem de diferença produzida para as imagens de estudo definidas na seção 4.2, a partir das imagens-fração abordadas na seção anterior. Na sequência, a Figura 22 mostra o perfil dos histogramas para as frações vegetação e fração solo da imagem-diferença.

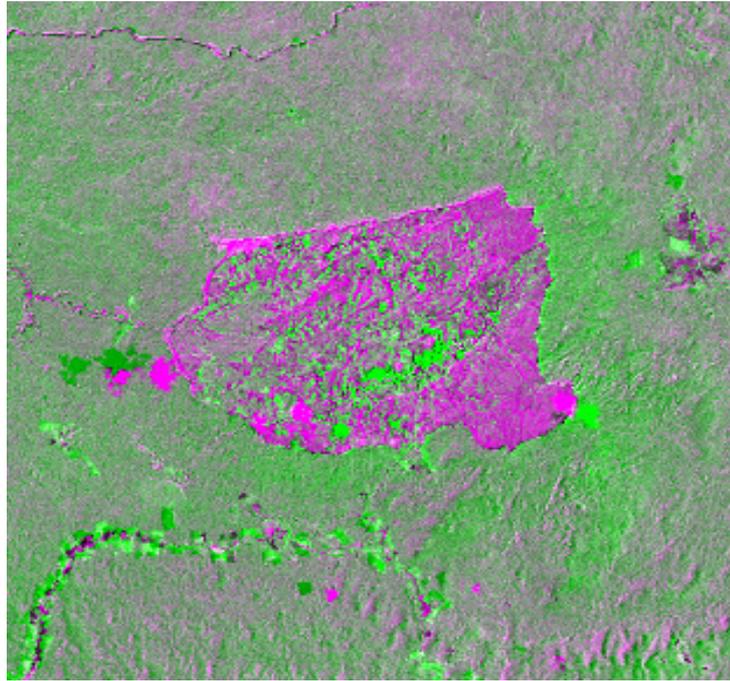
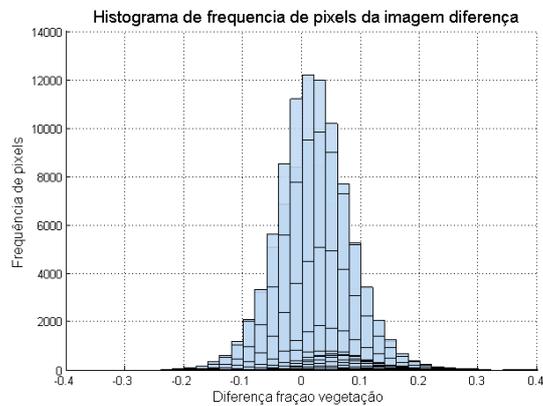
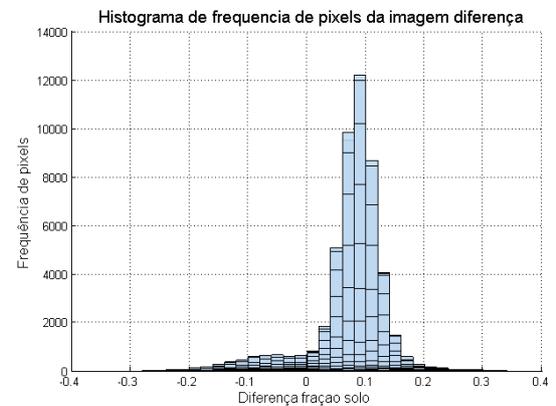


Figura 21 – Imagem de diferença com dados reais. Composição em falsa cor 3 (R) 2(G) 3(B).
Fonte: Adaptada de Haertel et al. (2004).



(a)



(b)

Figura 22 – Histogramas da imagem de diferenças. (a) Perfil da diferença na fração de vegetação. (b) Perfil da Diferença na fração de solo.

Fonte: Produzido pelo autor 2014.

4.5 Estimativa dos parâmetros das distribuições das classes mudança e não-mudança

Conforme descrito na seção 3.2, o algoritmo EM foi utilizado para fins de estimativa dos parâmetros (vetor de médias, matriz de covariância e probabilidade *a priori*) das distribuições das classes mudança e não-mudança.

A Figura 23, a seguir, mostra a dispersão dos dados da imagem-diferença para as imagens definidas na seção 4.2. Para essa dispersão, como foi mostrado na figura 14, o primeiro autovalor corresponde à variância estimada, inicialmente, para a classe mudança, enquanto o segundo corresponde à variância estimada, inicialmente, para a classe não-mudança, conforme prevê a metodologia deste estudo.

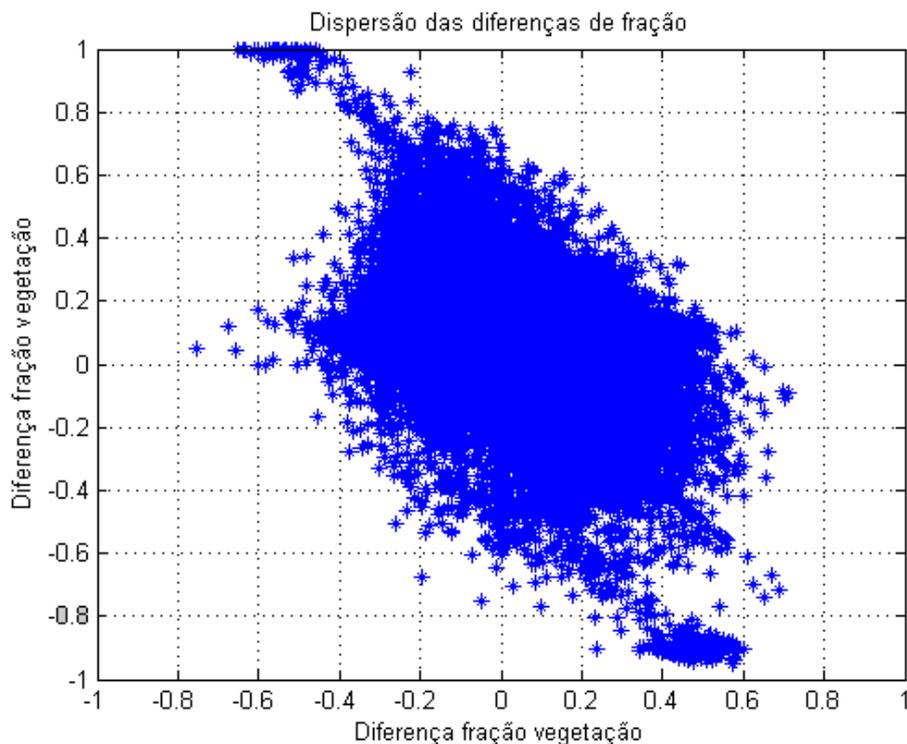


Figura 23 – Dispersão dos dados da imagem-diferença utilizada nos experimentos.
Fonte: Haertel et al. (2004).

Os parâmetros estatísticos das distribuições de probabilidade envolvidas na mistura de densidades foram estimados, conforme descrito na seção 3.2.2. Para a imagem de diferenças utilizada neste capítulo (Figura 21), os parâmetros estatísticos utilizados como entrada para o algoritmo EM foram determinados conforme descrito na seção 3.2.2.2. Os autovetores e os correspondentes autovalores para as diferenças nas frações vegetação e solo, mencionados

acima, foram estimados aplicando-se a transformação de Karhunen-Loeve. Os resultados obtidos para a matriz de autovetores (V) e matriz de autovalores (D) são:

$$V = \begin{bmatrix} -0.7304 & -0.6831 \\ -0.6831 & 0.7304 \end{bmatrix} \quad D = \begin{bmatrix} 0.0042 & 0 \\ 0 & 0.0099 \end{bmatrix}$$

Consideram-se aqui, segundo as notações definidas no capítulo de metodologia, as classes ω_1 e ω_2 , respectivamente, como as classes mudança e não-mudança, assim como, M_1 , S_1 , $P(\omega_1)$, M_2 , S_2 , $P(\omega_2)$ representam os vetores de médias, matrizes de covariâncias e probabilidades *a priori* de cada classe, respectivamente.

As estimativas iniciais propostas para cada classe são apresentadas a seguir:

$$M_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_1 = \begin{bmatrix} 0,0068 & -0,0028 \\ -0,0028 & 0,0099 \end{bmatrix}, P(\omega_1) = 0,1 \text{ e}$$

$$M_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_2 = \begin{bmatrix} 0,0042 & 0 \\ 0 & 0,0042 \end{bmatrix}, P(\omega_2) = 0,9.$$

Após a convergência do algoritmo EM para as estimativas iniciais acima, os valores dos parâmetros estimados para cada classe puderam ser obtidos e são apresentados a seguir:

A partir das equações definidas nas expressões 39 a 42, o algoritmo EM é processado e tende a convergir, segundo o exposto na seção 3.2.2.1. Foi utilizado como critério de convergência a diferença dos valores na iteração atual para a iteração imediatamente anterior. Para o experimento em questão, a convergência foi obtida para os seguintes valores de vetor de médias, matriz de covariâncias e probabilidades *a priori*:

$$M_1 = \begin{bmatrix} 0,0537 \\ 0,0297 \end{bmatrix}, S_1 = \begin{bmatrix} 0,0174 & -0,0099 \\ -0,0099 & 0,0267 \end{bmatrix}, P(\omega_1) = 0,2222 \text{ e}$$

$$M_2 = \begin{bmatrix} 0,0214 \\ 0,0890 \end{bmatrix}, S_2 = \begin{bmatrix} 0,0036 & -0,0004 \\ -0,0004 & 0,0009 \end{bmatrix}, P(\omega_2) = 0,7778.$$

Cabe mencionar aqui, que as médias ficaram próximas do vetor nulo, assim como as covariâncias das classes mudança e não-mudança apresentam uma baixa correlação entre as frações vegetação e solo e, por conseguinte, baixa correlação entre as imagens de diferença nas frações de vegetação e solo, valores próximos ao esperado na suposição teórica apresentada no capítulo de metodologia. Com relação aos valores finais estimados para as

matrizes de covariância, observa-se que, para a classe mudança, os dois componentes (vegetação e solo) apresentaram covariância negativa, como esperado. Da mesma forma que as variâncias para essa classe mantiveram valores altos. De forma muito semelhante, as matrizes de covariância para a classe não-mudança apresentaram valores muito próximos daqueles preconizados na abordagem teórica. Portanto, esses resultados confirmam as hipóteses assumidas na seção 3.2.2, quanto à estimativa dos valores iniciais para o algoritmo EM. A convergência das probabilidades *a priori*, para as duas classes, corrobora a suposição teórica de que a ocorrência de *pixels* da classe não-mudança para essa imagem é maior do que a ocorrência de *pixels* da classe mudança.

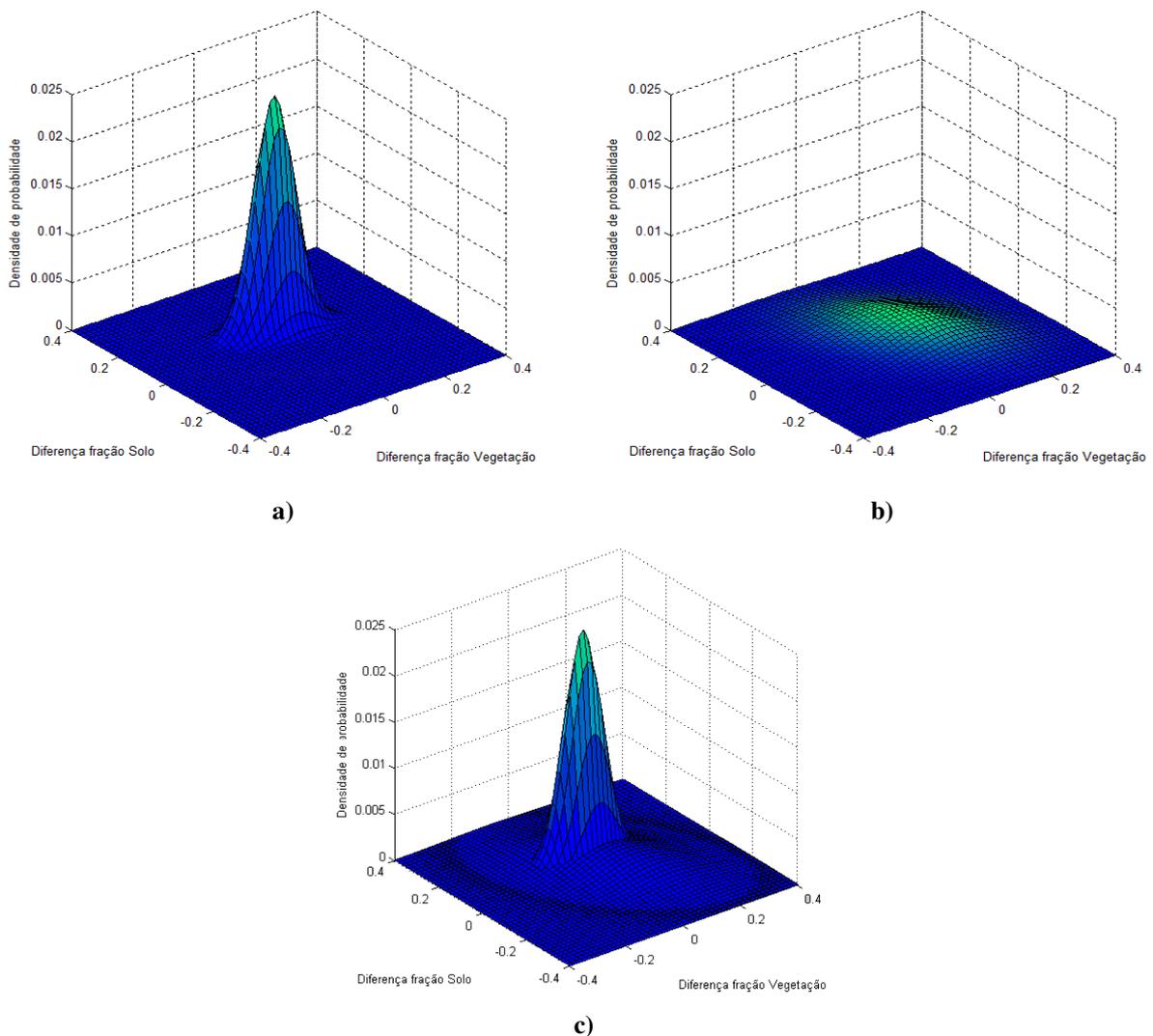


Figura 24 – Apresentação das densidades de probabilidade conjuntas das distribuições resultantes das diferenças entre as frações de vegetação e solo para o primeiro experimento: (a) classe não-mudança (b) classe mudança (c) sobreposição das classes mudança e não-mudança.

Fonte: Produzido pelo autor, 2014.

Acima, a Figura 24 mostra a distribuição das classes segundo os parâmetros estimados pelo algoritmo EM e a sobreposição das mesmas.

Os parâmetros estimados acima serão, posteriormente, utilizados como dados de entrada na função *mvnrnd.m*, do *software Matlab*, para a seleção inicial das amostras de treinamento, as quais após um teste de pertinência em relação às suas respectivas classes utilizando as distribuições estatísticas teóricas das classes mudança e não-mudança, estimadas pelos parâmetros determinados no algoritmo EM, serão devidamente rotuladas nas respectivas classes. Essas amostras serão utilizadas, posteriormente, como um dos parâmetros de entrada do classificador *SVM*, juntamente com o valor do coeficiente de custo, grau do *kernel* polinomial ou coeficiente γ do *kernel RBF* com a finalidade de obter a métrica de estimativa de pertinência dos *pixels*, conforme será visto em detalhe nas próximas seções.

Na próxima seção será abordado o processo de geração e seleção de amostras de treinamento, que serão utilizadas para treinar o classificador *SVM*.

4.6 geração das amostras de treinamento

Para fins de treinamento do classificador *SVM*, foram utilizadas amostras randômicas extraídas da imagem-diferença com a utilização da distribuição normal multivariada estimada para as classes. Essas amostras foram obtidas por meio da função *mvnrnd.m*, contida na biblioteca de funções do *Matlab*. Para usar essa função de geração aleatória de números segundo uma distribuição normal multivariada das classes, foram utilizados como valores de entrada para essa função os parâmetros da distribuição das classes mudança e não-mudança, estimados pelo algoritmo EM. As amostras geradas por essa função do *Matlab* foram rotuladas nas respectivas classes a que pertencem segundo o maior valor obtido por elas nas distribuições de classes estimadas pelo algoritmo EM. Isto é, amostras que foram geradas como pertencentes à determinada classe, mas que têm maior probabilidade de pertencerem a outra classe diferente desta, são rotuladas nessa outra classe. Dessa forma, são aceitas somente as amostras geradas, aleatoriamente, dentro da distribuição teórica daquela classe em que tenham a maior probabilidade de pertencer. Este procedimento foi executado tendo em vista a existência de uma sobreposição das classes mudança e não-mudança nas proximidades da origem dos eixos da dispersão das diferenças fração solo e vegetação (Figura 15), o que poderia fornecer amostras de treinamento não confiáveis, as quais poderiam levar a resultados de classificação imprecisos e, por consequência, à degradação da métrica de estimativa de pertinência pretendida. Para uso nos experimentos deste capítulo, os quais serão melhor

detalhados nas próximas seções, foram geradas amostras de treinamento com tamanhos de 200, 300 e 400 amostras. A Figura 25 mostra o gráfico de dispersão de uma amostra de treinamento com 400 *pixels* da classe mudança e 400 da classe não-mudança.

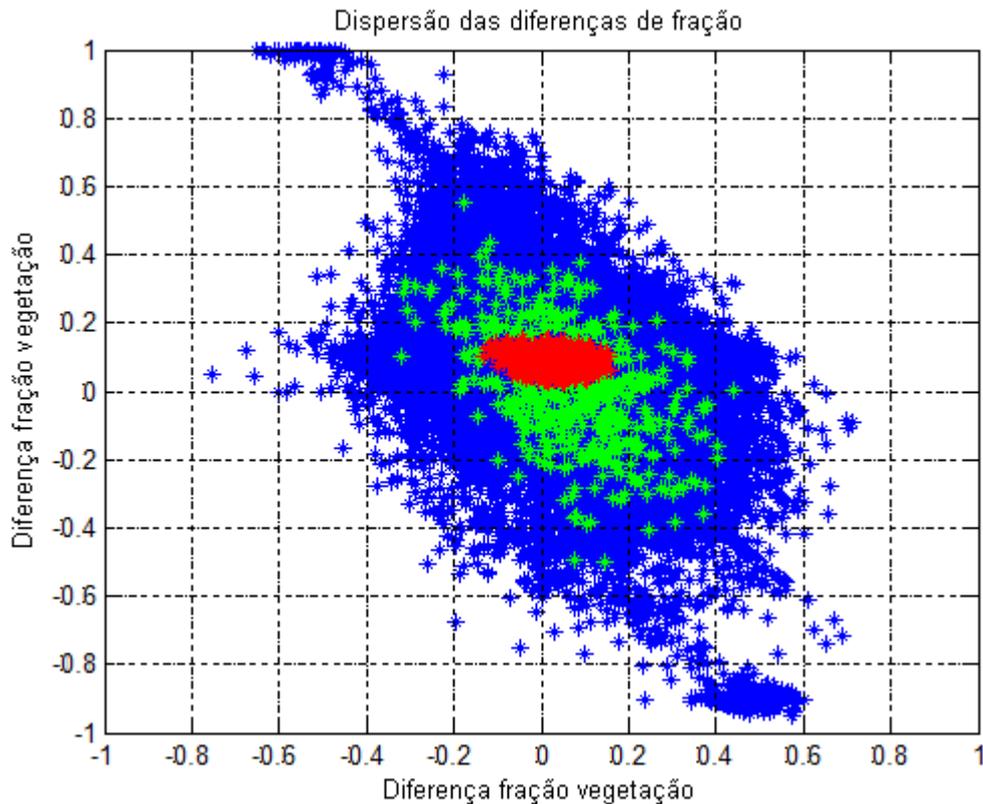


Figura 25 – Gráfico de dispersão de uma amostra de treinamento com 400 *pixels* da classe mudança (verde) e 400 *pixels* da classe não-mudança (vermelho) ambas sobrepostas sobre a distribuição de *pixels* da imagem-diferença (Figura 23) (azul)

Fonte: Produzida pelo autor, 2014.

Na próxima seção é apresentado o processo de geração de amostras de teste para verificar a capacidade de geração de estimativas da pertinência propostas nesta tese.

4.7 Geração das amostras de teste

Tendo em vista que não se dispõe de uma imagem que represente a verdade terrestre e nem se possui dados de campo para verificar a qualidade da métrica de pertinência proposta para a imagem em estudo, será utilizado um conjunto de amostras de testes aleatórios de amostras coletadas da imagem-diferença de fração. Para tanto, será utilizado o módulo do vetor de mudança preconizado na técnica CVA, que doravante será chamado de Módulo CVA, conforme apresentado no capítulo 2. Através dela será gerado, para cada *pixel* da imagem, um

vetor; tendo por origem o *pixel* da imagem tomado na data 1, e como extremidade desse vetor, o *pixel* correspondente a ele tomado na data 2. A utilização desta técnica permitirá a estimativa da mudança, que eventualmente ocorreu nos *pixels* utilizados como amostras de teste. Embora a técnica de *CVA* não seja capaz de separar as classes neste caso, pois não é possível determinar o limiar entre as classes, ela poderá ser utilizada para afirmar, para alguns *pixels*, se ocorreram ou não-mudança nas frações vegetação e solo entre as duas datas, ou seja, *pixels* apresentando Módulo *CVA* com valores superiores a 0,3 e menores que 0,6 corresponderão a *pixels* que apresentaram alguma mudança significativa ao longo do período e *pixels* com Módulo *CVA* menores que 0,1 corresponderão a *pixels* que não apresentaram mudança significativa ao longo do período. Finalmente, a escolha desses intervalos visa garantir que as amostras de teste sejam representativas de suas classes.

Dessa forma, com o uso do Módulo *CVA* serão coletados 900 *pixels* de amostra para cada classe, de forma aleatória e uniforme, ao longo da imagem de diferença, garantindo amostras com abrangência em toda a imagem e sem qualquer tendência na sua escolha. As amostras assim coletadas devem ser utilizadas para comprovar, estatisticamente, a eficiência da determinação de pertinência da proposta metodológica.

Com a finalidade de apresentar, visualmente, as amostras de teste, foi produzida uma imagem-módulo *CVA* a partir dos dados da imagem de diferença de frações, uma vez que o Módulo *CVA* utiliza a raiz quadrada das diferenças entre frações ao quadrado, não sendo necessária a utilização das imagens-fração. Portanto, essa imagem-módulo permitirá mostrar qualitativamente a pertinência das amostras de teste. As Figura 26 a 28, mostram, respectivamente: a imagem-módulo *CVA*, a localização das amostras geradas aleatoriamente segundo uma distribuição uniforme ao longo da imagem sobreposta à imagem-módulo *CVA* e o gráfico de dispersão das amostras nas classes mudança e não-mudança .

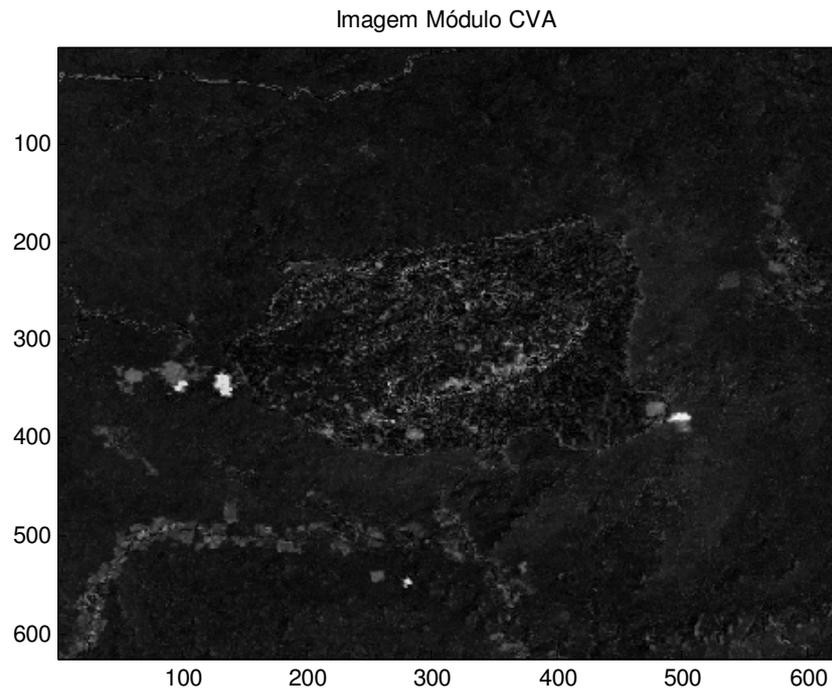


Figura 26 – Imagem-módulo do vetor *CVA*
Fonte: Produzido pelo autor, 2014.

Na figura 26, valores próximos do branco indicam valores altos de Módulo *CVA* e valores próximos do preto indicam valores baixos do Módulo *CVA* (próximos de zero), isto é, valores próximos da média da distribuição não-mudança.

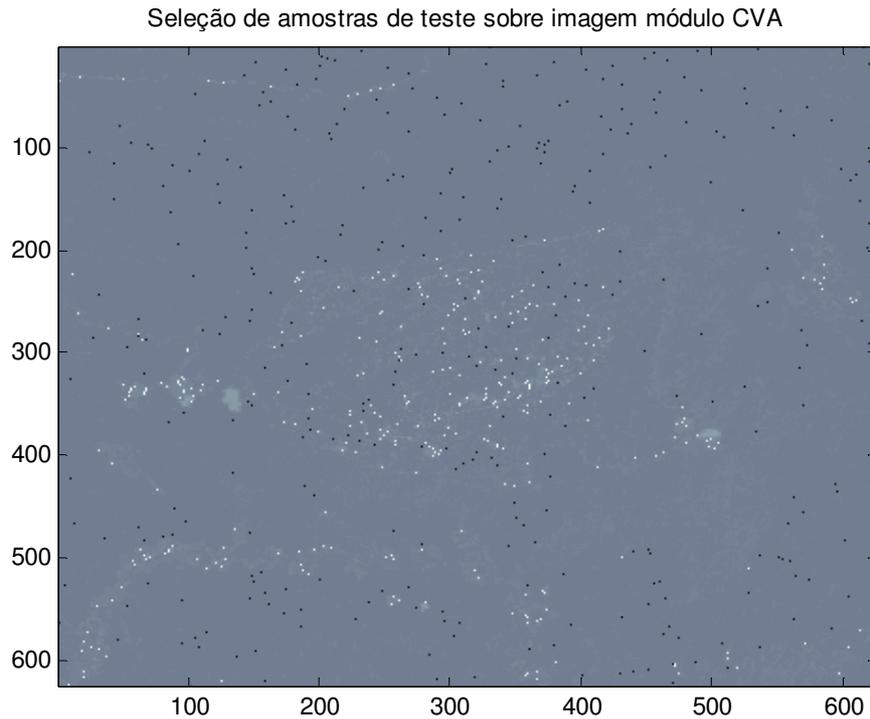


Figura 27 – Imagem-módulo do vetor *CVA* com amostras da classe mudança em branco (módulo do vetor mudança maior que 0,3 e menor que 0,6) e amostras da classe não-mudança em preto (módulo do vetor mudança menor que 0,1)

Fonte: Produzido pelo autor, 2014.

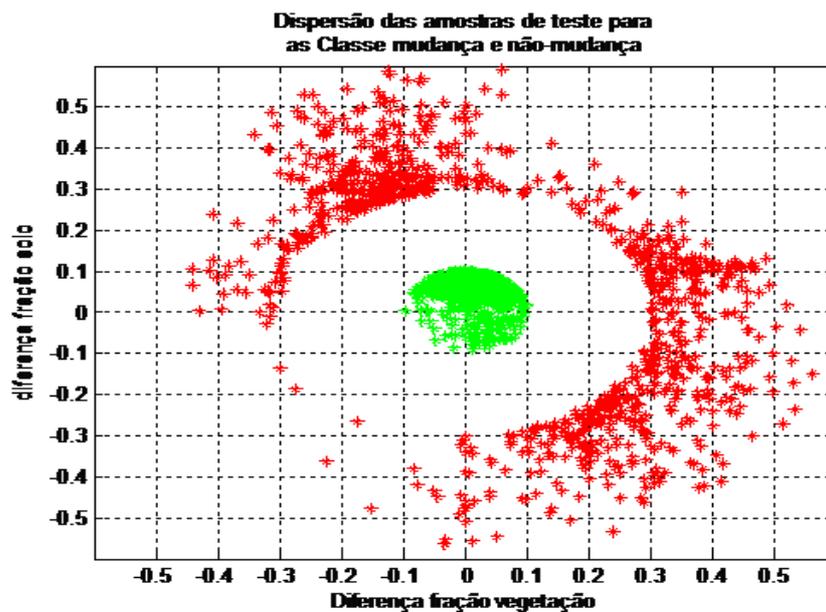


Figura 28 – Gráfico de dispersão da amostra de teste nas classes mudança (vermelho) e não-mudança (verde)

Fonte: Produzido pelo autor, 2014.

4.8 Experimentos utilizando o classificador SVM

Os experimentos deste capítulo utilizaram o classificador SVM tanto com o *kernel* polinomial, quanto o gaussiano (RBF). As amostras de treinamento utilizadas foram geradas automaticamente pelo uso de uma função do *Matlab* e rotuladas segundo uma métrica estatística de pertinência à classe que representam, usando os parâmetros estatísticos gerados pelo algoritmo EM. Os coeficientes e parâmetros utilizados no classificador SVM e nos *kernels* são detalhados a seguir.

Para os experimentos com a finalidade de verificar a eficiência da metodologia, utilizou-se a amostra de teste definida na seção 4.7 como referência, isto é, os *pixels* dessa amostra foram classificados com o classificador SVM e sua pertinência foi estimada pela métrica Delta b, sendo analisado o número de *pixels* corretamente associado à sua classe segundo essa métrica, além da média e desvio padrão para as pertinências observadas.

De forma semelhante, Para os experimentos envolvendo a comparação entre a metodologia proposta neste trabalho e a proposta de (ZANOTTA, 2010) foi utilizado o mapa de pertinência de *pixels* apresentado, naquele trabalho, como mapa de referência. A Figura 29, a seguir, apresenta esse mapa:

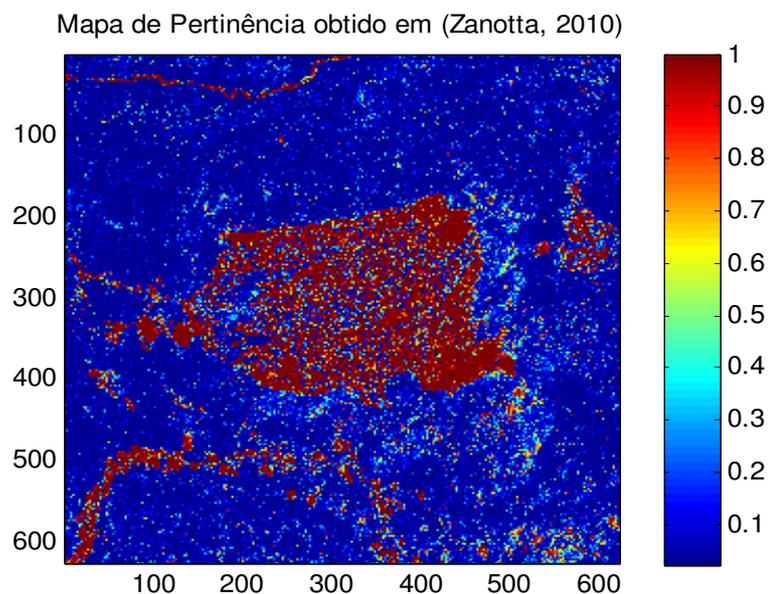


Figura 29 – Mapa de diferença de pertinência obtida em Zanotta, (2010)
Fonte: Zanotta (2010).

4.8.1 Coeficiente e parâmetros utilizados no classificador SVM

Em todos os experimentos foram realizados testes com 200, 300 e 400 amostras de treinamento para cada classe. As amostras de testes foram selecionadas conforme descrito na seção 4.7.

No caso do *kernel* polinomial, para o parâmetro d (grau do polinômio), foram realizados testes com valores de d variando de 2 a 5. O valor utilizado para a variável C foi 10, essa escolha foi feita levando-se em conta os bons resultados obtidos em (ANDREOLA, 2009), para essa escolha de valor.

No caso do *kernel RBF* para o parâmetro γ foram utilizados os valores de 1, 2, 5, 10, 80, 85, 90, 95 e 100, além disso, o parâmetro C também foi igual a 10 (BONESSO, 2013).

4.9 Experimentos utilizando o classificador SVM com Kernels Polinomial e RBF para amostra de teste

Os experimentos abordados nesta seção utilizaram o classificador SVM tanto com o *kernel* polinomial, quanto o gaussiano (RBF). Os coeficientes e parâmetros utilizados no classificador e nos *kernels* foram detalhados na seção 4.5. As amostras de treinamento utilizadas seguem o descrito na seção 4.6.

4.9.1 Experimentos com kernel Polinomial aplicado à amostra de teste

Nestes experimentos são apresentados nas Figuras 29 a 31 os gráficos de dispersão das estimativas de pertinência dos *pixels* das amostras de testes gerados a partir da metodologia proposta para os melhores e piores resultados, e as tabelas de 1 até 6, mostram, quantitativamente, as seguintes variáveis a serem analisadas: valor da menor pertinência na classe, valor médio de pertinência à classe, o desvio padrão de pertinência na classe e o maior valor de pertinência na classe, além do percentual de *pixels* classificados através da métrica de pertinência nas classes mudança e não-mudança.

4.9.1.1 Experimento com kernel polinomial utilizando 200 amostras de treinamento

Neste experimento, utilizando o *kernel* polinomial, se utilizou para o parâmetro d (grau do polinômio), valores de 2 a 5. O valor utilizado para a variável C foi 10 e foram

utilizadas 200 amostras de treinamento para cada uma das classes. A seguir, a Figura 30 mostra a dispersão da melhor e da pior classificação dos *pixels* da amostra de testes nas classes mudança e não-mudança e as tabelas 1 e 2 mostram as variáveis de análise referidas na seção 4.8.2.1.

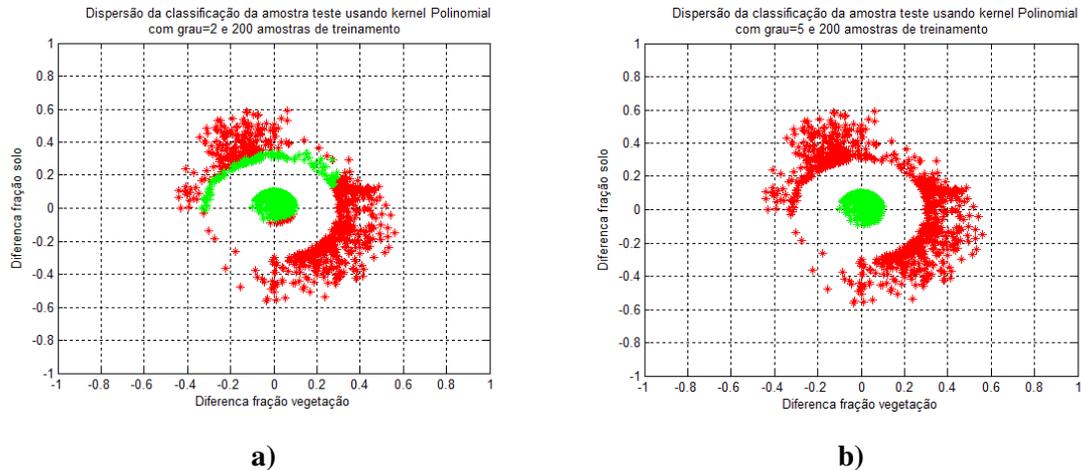


Figura 30 – Classificação da amostra de testes no classificador SVM com 200 amostras de treinamento e utilizando *kernel* Polinomial em a) Pior pertinência usando *grau*= 2 e em b) Melhor pertinência usando utilizando *grau*=5. Classe mudança representada em vermelho e não-mudança em verde

Fonte: Produzido pelo autor, 2014.

As tabelas 1 e 2, a seguir, mostram o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e, finalmente, o percentual de *pixels* classificados, segundo sua pertinência, nas classes mudança e não-mudança.

Tabela 1 – Avaliação da pertinência na amostra de teste usando kernel Polinomial com 200 amostras de treinamento para classe mudança

Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
2	23,69	59,38	12,69	100,00	84,67
3	39,05	63,59	9,80	100,00	99,00
4	51,00	63,36	9,17	100,00	100,00
5	51,00	62,92	8,88	100,00	100,00

Fonte: Produzido pelo autor, 2014.

Tabela 2 – Avaliação da pertinência na amostra de teste usando kernel Polinomial com 200 amostras de treinamento para classe não-mudança

Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
2	48,04	90,42	11,12	100,00	97,56
3	48,32	92,82	10,51	100,00	98,33
4	51,00	79,54	12,33	100,00	100,00
5	51,00	82,19	12,21	100,00	100,00

Fonte: Produzido pelo autor, 2014.

4.9.1.2 Experimento com kernel polinomial utilizando 300 amostras de treinamento

Neste experimento, utilizando o *kernel* polinomial, se utilizou para o parâmetro d (grau do polinômio), valores entre 2 e 5. O valor utilizado para a variável C foi 10 e foram utilizadas 300 amostras de treinamento para cada uma das classes. A seguir, a Figura 31 mostra a Dispersão da melhor classificação dos *pixels* da amostra de testes nas Classes mudança e não-mudança, e as tabelas 3 e 4 mostram as variáveis de análises referidas na seção 4.9.1.

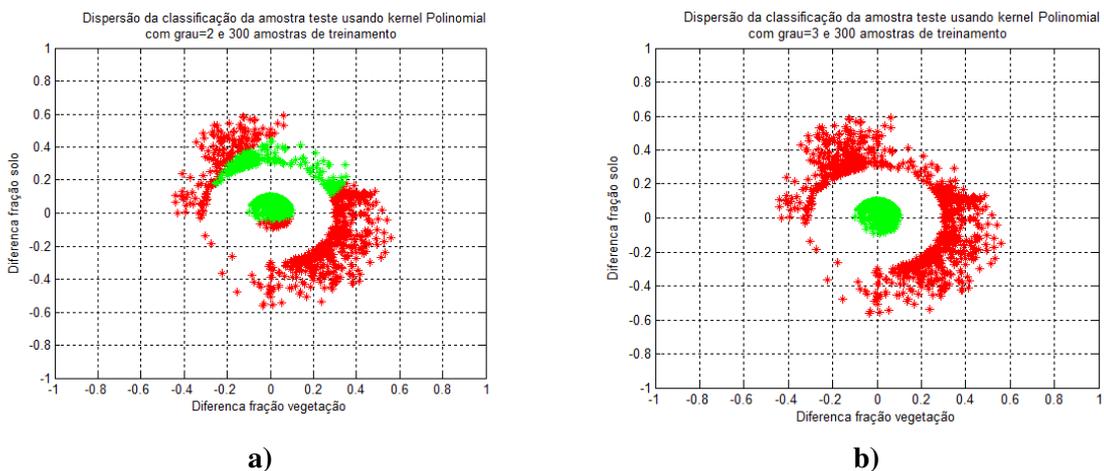


Figura 31 – Classificação da amostra de testes no classificador SVM com 300 amostras de treinamento e utilizando *kernel* Polinomial em a) Pior pertinência usando *grau*=2 e em b) melhor pertinência usando utilizando *grau*= 3. Classe mudança representada em vermelho e não-mudança em verde

Fonte: Produzido pelo autor, 2014.

As tabelas 3 e 4, a seguir, mostram o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e, finalmente, o

percentual de *pixels* classificados, segundo sua pertinência, nas classes mudança e não-mudança.

Tabela 3 – Avaliação da pertinência na amostra de teste usando kernel Polinomial com 300 amostras de treinamento para classe mudança

Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
2	0,00	54,86	19,14	100,00	74,67
3	51,00	63,75	9,74	100,00	100,00
4	17,42	63,47	12,30	100,00	93,78
5	51,00	61,72	8,70	100,00	100,00

Fonte: Produzido pelo autor, 2014.

Tabela 4 – Avaliação da pertinência na amostra de teste usando kernel Polinomial com 300 amostras de treinamento para classe não-mudança

Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
2	46,73	82,14	11,28	93,51	95,00
3	51,00	86,59	7,72	100,00	100,00
4	47,93	88,40	11,08	100,00	97,33
5	51,00	77,92	12,53	100,00	100,00

Fonte: Produzido pelo autor, 2014.

4.9.1.3 Experimento com kernel polinomial utilizando 400 amostras de treinamento

Neste experimento, utilizando o *kernel* polinomial, se utilizou para o parâmetro d (grau do polinômio) valores entre 2 e 5. O valor utilizado para a variável C foi 10 e foram utilizadas 400 amostras de treinamento para cada uma das classes. A seguir, a Figura 32 mostra a dispersão da melhor classificação dos *pixels* da amostra de testes nas Classes mudança e não-mudança, e as tabelas 5 e 6 mostram as variáveis de análises referidas na seção 4.9.1.

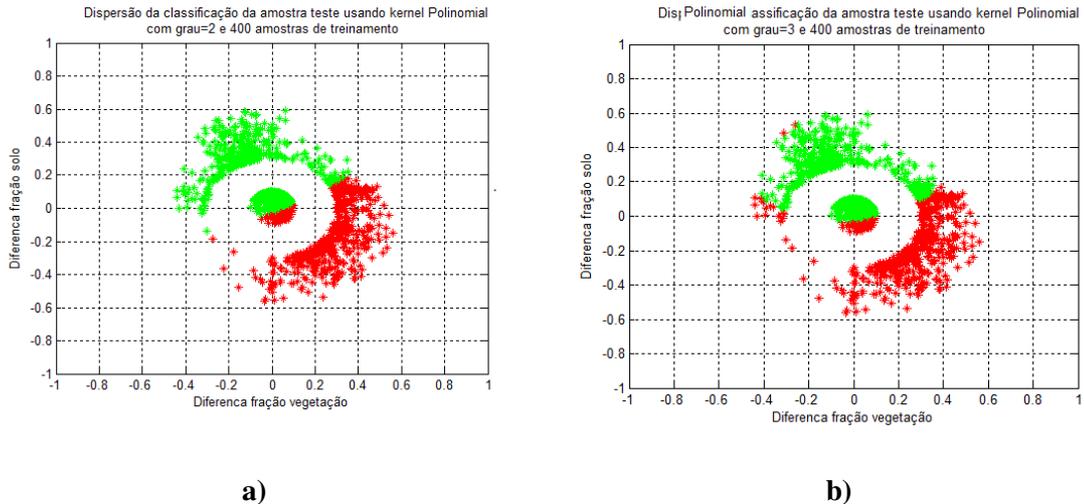


Figura 32 – Classificação da amostra de testes no classificador SVM com 400 amostras de treinamento e utilizando *kernel* Polinomial em a) Pior pertinência usando *grau*= 2 e em b) melhor pertinência usando utilizando *grau*=3. Classe mudança representada em vermelho e não-mudança em verde.

Fonte: Produzido pelo autor, 2014.

As tabelas 5 e 6, a seguir, mostram o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e, finalmente, o percentual de *pixels* classificados, segundo sua pertinência, nas classes mudança e não-mudança.

Tabela 5 - Avaliação da pertinência na amostra de teste usando kernel Polinomial com 400 amostras de treinamento para classe mudança

Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
2	0,00	48,91	28,62	100,00	55,67
3	0,00	45,16	28,22	100,00	54,11
4	0,00	47,34	30,06	100,00	56,89
5	0,00	44,37	31,30	100,00	54,56

Fonte: Produzido pelo autor, 2014.

Tabela 6 – Avaliação da pertinência na amostra de teste usando kernel Polinomial com 400 amostras de treinamento para classe não-mudança

Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
2	43,18	56,61	4,11	61,93	90,67
3	45,07	67,70	7,96	77,12	92,22
4	44,27	58,84	4,77	65,39	91,78
5	44,82	61,44	5,63	68,74	92,00

Fonte: Produzido pelo autor, 2014.

A partir da análise tanto dos gráficos, quanto às tabelas para o experimento envolvendo o *kernel* Polinomial, pode-se observar que os resultados de pertinência são compatíveis com o que foi previsto pela técnica de CVA. Embora nos piores casos tenha havido uma diminuição no número de *pixels* com pertinência na classe não-mudança em relação ao que era esperado dessa classe, o que sugere que o agrupamento de *pixels* referentes a ela está levemente deslocado na direção positiva do eixo diferença fração solo. Essa tendência fica evidente no experimento com 400 amostras de treinamento. Esses resultados apontam para o que foi teorizado no capítulo 3, referente à distribuição das classes (figura 22).

Além disso, pode ser observado que os resultados de pertinência foram melhores para a classe mudança, exceto no teste com 400 amostras de treinamento, no qual há uma perda significativa de *pixels* com pertinência na classe mudança. Esses resultados sugerem que a escolha da amostra não é compatível com a distribuição das classes e que o classificador é sensível ao aumento no número de amostras de treinamento.

4.9.2 Experimentos com *kernel* RBF aplicado à amostra de teste

Para esses experimentos as Figuras 33 a 35 apresentam os gráficos de dispersão das estimativas de pertinência dos *pixels* das amostras de testes gerados a partir da metodologia proposta para os melhores resultados obtidos, e as Tabelas de 7 até 12, mostram quantitativamente as seguintes variáveis a serem analisadas: valor da menor pertinência na classe, valor médio de pertinência na classe, o desvio padrão de pertinência na classe e o maior valor de pertinência na classe, além do percentual de *pixels* classificados através da métrica de pertinência corretamente aplicada nas classes mudança e não-mudança.

4.9.2.1 Experimento com kernel RBF utilizando 200 amostras de treinamento

Neste experimento, utilizando o *kernel* RBF, se utilizou para o parâmetro γ valores 1, 2, 5 e 10, 80, 85, 90, 95 e 100. O valor utilizado para a variável C foi 10 e foram utilizadas 200 amostras de treinamento para cada uma das classes. A seguir, a Figura 33 mostra a Dispersão da melhor e pior classificação dos *pixels* da amostra de teste nas Classes mudança e não-mudança em relação às amostras geradas a partir do módulo CVA.

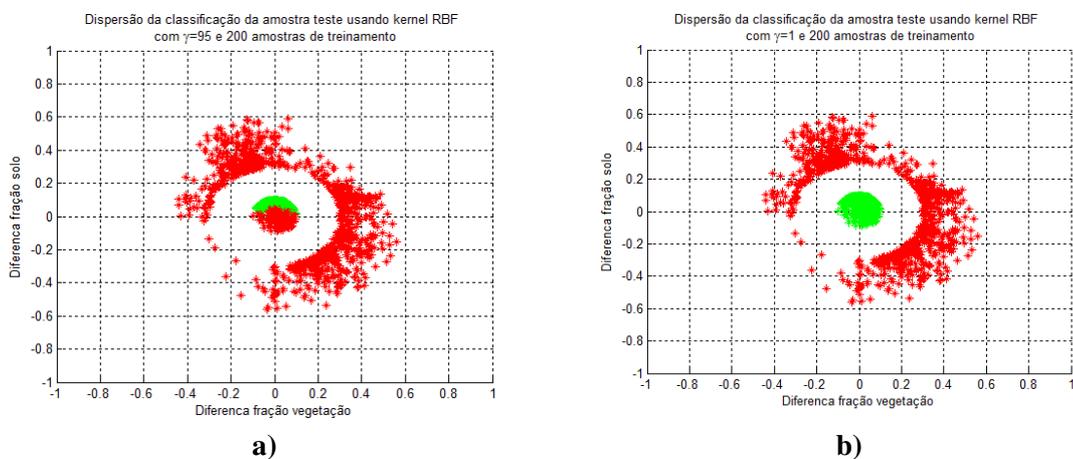


Figura 33 – Classificação da amostra de testes no classificador SVM com 200 amostras de treinamento e utilizando o *kernel* RBF em a) Pior classificação por pertinência usando $\gamma=95$ e em b) melhor classificação por pertinência usando utilizando $\gamma=1$. Classe mudança representada em vermelho e não-mudança em verde

Fonte: Produzido pelo autor, 2014.

As tabelas 7 e 8, a seguir, mostram o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências e o maior valor de pertinência e, finalmente, o percentual de *pixels* classificados, segundo sua pertinência, corretamente nas classes mudança e não-mudança.

Tabela 7 – Avaliação da pertinência na amostra de teste usando kernel RBF com 200 amostras de treinamento para classe mudança

Coeficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
1	51,00	65,27	10,14	100,00	100,00
2	51,00	66,11	9,86	100,00	100,00
5	58,15	74,17	9,01	100,00	100,00
10	62,11	80,05	8,63	100,00	100,00
80	73,02	76,12	2,97	87,52	100,00
85	71,42	74,47	3,50	86,87	100,00
90	74,03	77,30	2,82	89,54	100,00
95	71,27	73,28	1,86	82,05	100,00
100	75,33	78,92	4,02	93,25	100,00

Fonte: Produzido pelo autor, 2014.

Tabela 8 – Avaliação da pertinência na amostra de teste usando kernel RBF com 200 amostras de treinamento para classe não-mudança

Coeficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
1	51,00	94,50	8,14	100,00	100,00
2	51,00	80,80	11,84	100,00	100,00
5	48,36	87,13	10,44	100,00	99,33
10	42,46	90,69	11,89	100,00	96,56
80	0,00	78,17	25,27	100,00	86,56
85	0,00	74,60	25,61	100,00	84,67
90	0,00	80,19	23,33	100,00	88,78
95	0,00	68,57	24,33	100,00	81,56
100	0,00	74,69	24,89	100,00	86,78

Fonte: Produzido pelo autor, 2014.

4.9.2.2 Experimento com kernel RBF utilizando 300 amostras de treinamento

Neste experimento utilizando o *kernel* RBF, se utilizou para o parâmetro γ valores 1, 2, 5 e 10, 80, 85, 90, 95 e 100. O valor utilizado para a variável C foi 10 e foram utilizadas 200 amostras de treinamento para cada uma das classes. A seguir, a Figura 34 mostra a

Dispersão da melhor classificação dos *pixels* da amostra de teste nas Classes mudança e não-mudança.

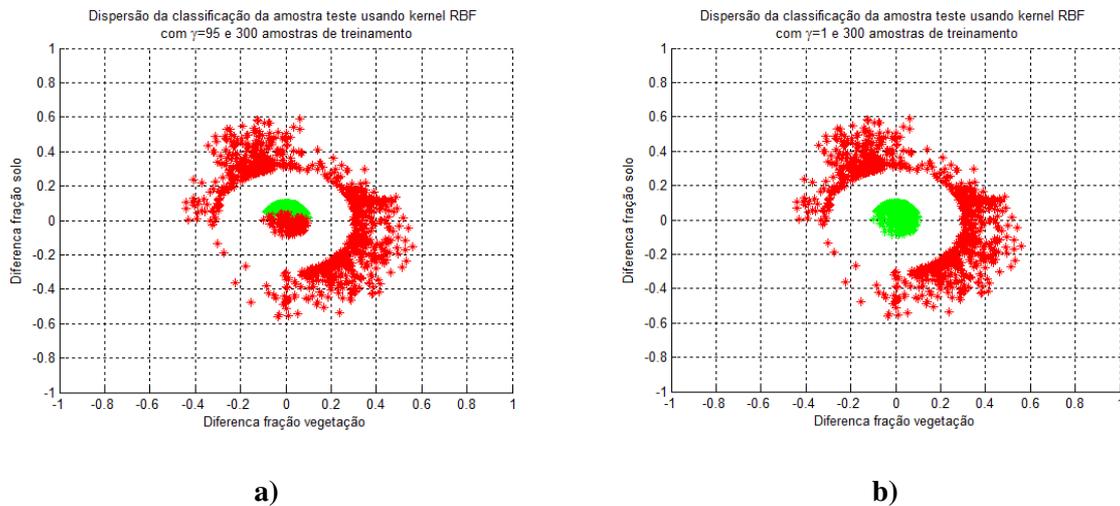


Figura 34 – Classificação da amostra de testes no classificador *SVM* com 300 amostras de treinamento e utilizando o *kernel RBF* em a) Pior classificação por pertinência usando $\gamma=95$ e em b) melhor classificação por pertinência usando utilizando $\gamma=1$. Classe mudança representada em vermelho e não-mudança em verde

Fonte: Produzido pelo autor, 2014.

As tabelas 9 e 10, a seguir, mostram o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e, finalmente, o percentual de *pixels* classificados, segundo sua pertinência, corretamente nas classes mudança e não-mudança.

Tabela 9 – Avaliação da pertinência na amostra de teste usando kernel RBF com 300 amostras de treinamento para classe mudança

Coefficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
1	51,00	67,67	10,33	100,00	100,00
2	53,12	69,43	9,63	100,00	100,00
5	55,54	79,70	11,89	100,00	100,00
10	66,04	86,50	7,28	100,00	100,00
80	78,78	82,20	2,94	97,73	100,00
85	77,46	80,87	3,16	93,61	100,00
90	80,77	84,85	3,13	97,42	100,00
95	78,07	81,80	2,75	90,67	100,00
100	80,11	84,28	2,60	93,95	100,00

Fonte: Produzido pelo autor, 2014.

Tabela 10 – Avaliação da pertinência na amostra de teste usando kernel RBF com 300 amostras de treinamento para classe não-mudança

Coeficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
1	51,00	90,00	9,24	100,00	100,00
2	48,47	84,55	11,23	100,00	99,67
5	43,26	89,66	12,83	100,00	96,00
10	38,97	89,59	13,46	100,00	95,89
80	0,00	82,53	25,61	100,00	88,44
85	0,00	79,75	26,46	100,00	87,11
90	0,00	80,80	25,73	100,00	88,22
95	0,00	75,63	25,78	100,00	85,67
100	0,00	80,39	26,25	100,00	87,44

Fonte: Produzido pelo autor, 2014.

4.9.2.3 Experimento com *kernel* RBF utilizando 400 amostras de treinamento

Neste experimento, utilizando o *kernel* RBF, se utilizou para o parâmetro γ valores 1, 2, 5 e 10, 80, 85, 90, 95 e 100. O valor utilizado para a variável C foi 10 e foram utilizadas 400 amostras de treinamento para cada uma das classes. A seguir, a Figura 35 mostra a Dispersão da melhor classificação dos *pixels* da amostra de teste nas Classes mudança e não-mudança.

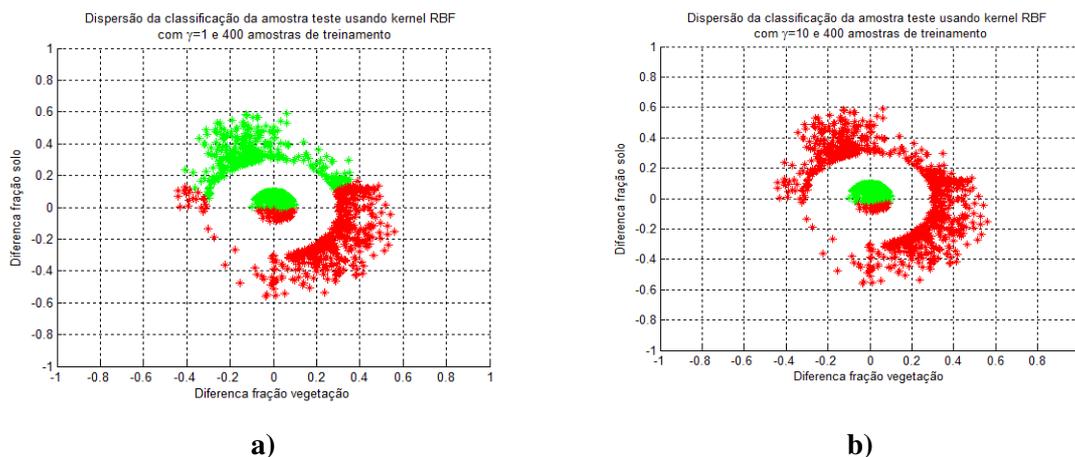


Figura 35 – Classificação da amostra de testes no classificador *SVM* com 400 amostras de treinamento e utilizando o *kernel RBF* em a) Pior classificação por pertinência usando $\gamma=1$ e em b) melhor classificação por pertinência usando utilizando $\gamma=10$. Classe mudança representada em vermelho e não-mudança em verde

Fonte: Produzido pelo autor, 2014.

As tabelas 11 e 12, a seguir, mostram o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e finalmente o percentual de *pixels* classificados, segundo sua pertinência, corretamente nas classes mudança e não-mudança.

Tabela 11 – Avaliação da pertinência na amostra de teste usando kernel RBF com 400 amostras de treinamento para classe mudança

Coefficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
1	0,00	47,17	30,35	100,00	55,11
2	0,00	58,31	23,22	100,00	72,11
5	42,32	76,05	14,71	100,00	98,89
10	66,16	88,31	8,12	100,00	100,00
80	80,49	85,02	3,09	95,68	100,00
85	80,40	85,04	3,17	96,05	100,00
90	79,09	83,13	2,64	91,88	100,00
95	80,08	84,11	2,92	94,44	100,00
100	81,74	86,20	3,00	94,52	100,00

Fonte: Produzido pelo autor, 2014.

Tabela 12 – Avaliação da pertinência na amostra de teste usando kernel RBF com 400 amostras de treinamento para classe não-mudança

Coefficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
1	42,98	66,52	7,94	75,94	91,56
2	42,84	76,13	11,03	88,40	92,33
5	41,22	87,55	13,59	100,00	94,78
10	39,85	90,54	13,21	100,00	95,89
80	0,00	81,94	24,70	100,00	88,78
85	0,00	81,56	25,88	100,00	88,33
90	0,00	81,82	25,77	100,00	88,00
95	0,00	81,24	26,36	100,00	87,44
100	0,00	80,57	26,26	100,00	87,67

Fonte: Produzido pelo autor, 2014.

A partir da análise, tanto dos gráficos quanto das tabelas para o experimento envolvendo o *kernel RBF*, pode-se observar que os resultados de pertinência são compatíveis com o que foi previsto pela técnica de *CVA*. Embora nos piores casos tenha ocorrido a diminuição no número de *pixels* com pertinência na classe não-mudança em relação ao que era esperado, o que sugere, assim como já observado para o experimento com o *kernel* polinomial, que o agrupamento de *pixels* referentes a essa classe está levemente deslocado na direção positiva do eixo diferença fração solo. Essa tendência também ficou evidenciada no experimento com 400 amostras de treinamento, como havia ocorrido no experimento com o *kernel* polinomial. Esses resultados, novamente, sugerem que o que havia sido teorizado no capítulo 3, referente à distribuição das classes (figura 22), está correto.

Ainda sobre este experimento, pode ser observado que os resultados de pertinência foram sempre melhores para a classe mudança, exceto no teste com 400 amostras de treinamento, em que há uma perda significativa de *pixels* com pertinência nessa classe. Esses resultados sugerem que a escolha das amostras de testes não foram compatíveis com a distribuição das classes e, também, que o classificador é sensível ao aumento do número de amostras de treinamento.

De forma geral, em ambos os experimentos os resultados obtidos com SVM usando os *kernels* Polinomial e *RBF*, aplicados às amostras de teste, mostraram que o classificador consegue representar bem as distribuições das classes segundo as características teorizadas no capítulo 3, as quais forneceram os parâmetros que modelaram o classificador a partir do uso do algoritmo EM. Entretanto, essa fidelidade do classificador em relação às distribuições de classe ocasionou a diminuição, para alguns testes, dos números de *pixels* com pertinência em ambas as classes em relação ao que era esperado para a amostra de teste.

Uma possível explicação para a ocorrência desses resultados é devida ao fato de que se utilizou a métrica do módulo do vetor mudança considerando a origem dos vetores de mudança no ponto (0,0), quando a partir da análise feita no capítulo 3, sobre as distribuições das classes na imagem-diferença, foi constatado que a classe não-mudança tinha a média dos seus *pixels* levemente deslocada no sentido do eixo positivo da diferença fração solo, isto é, média igual a (0,0214, 0,089) o que significa que esta pequena mudança é considerada como não significativa. Portanto, uma forma de ajustar esse efeito ao se tomar amostras de teste é considerar a métrica do Módulo *CVA* com seu centro trasladado para esse valor de média, isto é, todos os valores de módulos dos *pixels* devem ser calculados não em relação ao centro (0,0), mas em relação a esse valor de média. A utilização desta nova abordagem permitirá a estimativa da mudança que, eventualmente, ocorreu nesses *pixels* amostrados de forma mais

realista. Sendo assim, a coleta de amostras de teste deverá ser feita da mesma forma como havia sido feito na seção anterior, apenas levando em conta esse ajuste na medida do módulo CVA. Para tanto, serão utilizados os mesmos intervalos do experimento anterior, isto é, *pixels* apresentando o módulo do vetor de mudança deslocado, com valores superiores a 0,3 e menores que 0,6 correspondem a *pixels* que apresentaram alguma mudança significativa ao longo do período; e *pixels* com valores menores que 0,1 correspondem a *pixels* que não apresentaram mudança significativa ao longo do período. Assim, com o uso do Módulo CVA deslocado serão, agora, também coletados 900 *pixels* de amostra para cada classe, de forma uniforme e aleatória, garantindo amostras com abrangência em toda a imagem e sem tendência na sua escolha. As amostras assim coletadas devem ser utilizadas para comprovar estatisticamente a eficiência da metodologia proposta e representarem a real distribuição das classes.

As novas amostras de teste, selecionadas com o uso da imagem-módulo deslocada são mostrados nas Figuras 36 até a 38, que apresentam respectivamente: a imagem-módulo CVA deslocada para os *pixels* da imagem-diferença, a localização das amostras geradas aleatoriamente segundo uma distribuição uniforme na imagem-módulo CVA deslocada e o gráfico de dispersão das amostras das amostras nas classes mudança e não-mudança .

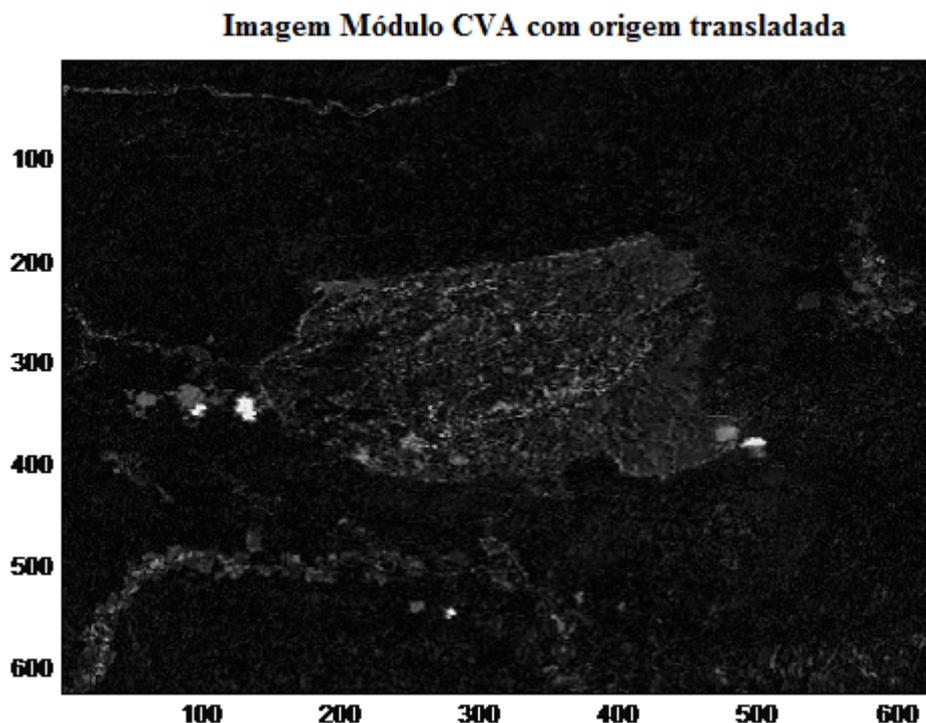


Figura 36 – Imagem-módulo do vetor CVA com origem transladada
Fonte: Produzido pelo autor, 2014.

Na figura 26 valores de *pixels* próximos do branco indicam valores altos de Módulo *CVA*, e valores próximos do preto indicam valores baixos do Módulo *CVA* (próximos de zero), isto é, próximos da média da distribuição da classe não-mudança.

A imagem a seguir mostra a distribuição das amostras de teste e como esses valores estão uniformemente distribuídos ao longo da imagem-módulo *CVA*.

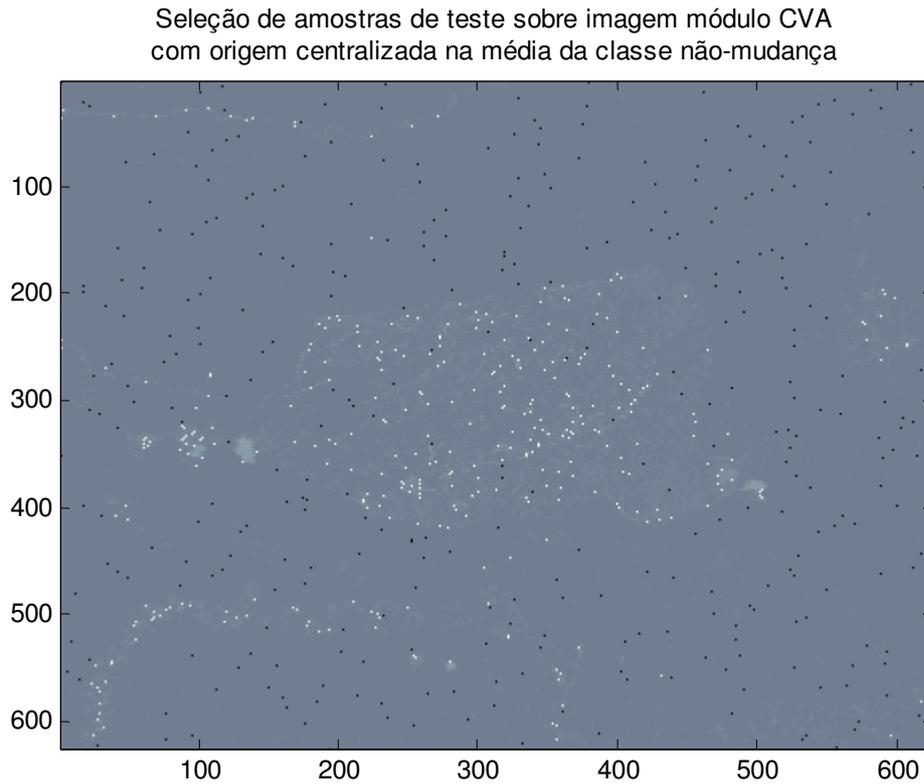


Figura 37 – Imagem-módulo *CVA* como a origem deslocada para a média da classe não-mudança. as amostras da classe mudança, em branco (módulo do vetor mudança centralizado maior que 0,3 e menor que 0,6) e as amostras da classe não-mudança em preto (módulo do vetor mudança centralizado menor que 0,1).

Fonte: Produzido pelo autor, 2014.

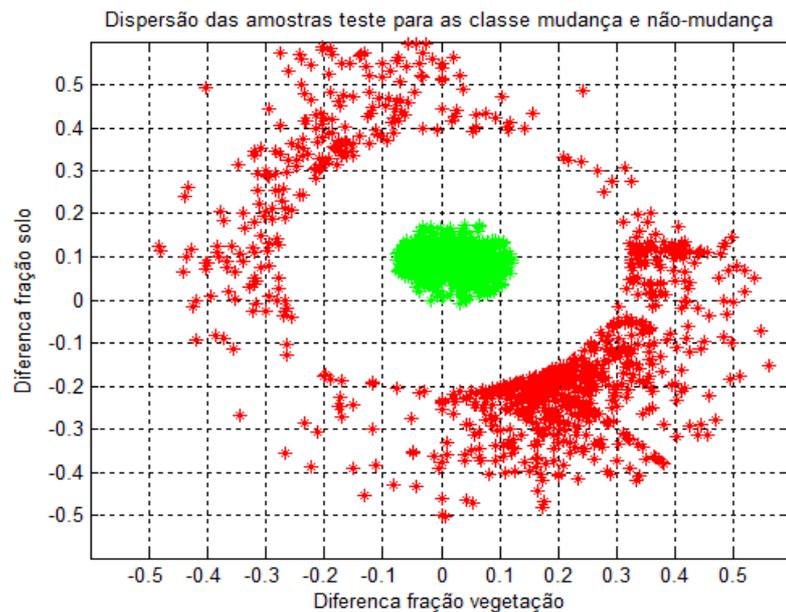


Figura 38 – Gráfico de dispersão da amostra de teste nas classes mudança (vermelho) e não-mudança (verde)

Fonte: Produzido pelo autor, 2004.

4.10 Experimentos utilizando o classificador SVM para amostra de teste gerada a partir da métrica do módulo do vetor mudança deslocado

Os novos experimentos foram desenvolvidos semelhantemente ao experimento anterior, utilizando o classificador SVM com o *kernel* polinomial e gaussiano (RBF), fazendo uso dos mesmos parâmetros, coeficientes e amostras de treinamento.

Para todos os testes deste experimento foram utilizadas as amostras de teste definida na Figura 38 e os *pixels* dessa amostra foram classificados com o classificador SVM e foi analisada a pertinência desses *pixels*, sendo considerados pertencentes à classe *pixels* com valor de pertinência maior do que 50%.

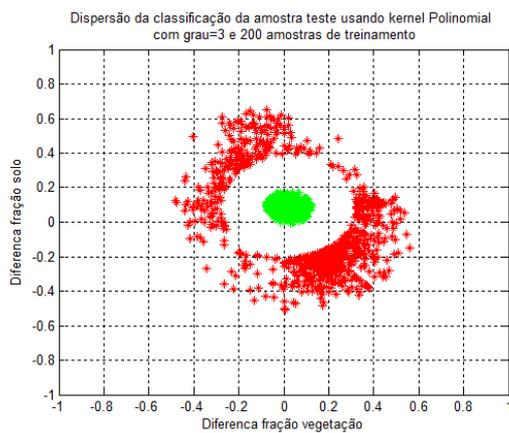
4.10.1 Experimentos utilizando o classificador SVM com Kernels Polinomial e RBF para amostra de teste

Nestes experimentos são apresentados nas Figuras 39 e 40 os melhores e piores resultados gráficos de dispersão das estimativas de pertinência dos *pixels* das amostras de testes gerados a partir da metodologia proposta em relação aos resultados esperados para essas amostras utilizando o classificador SVM, com 200, 300 e 400 amostras de treinamento. As tabelas 13 e 14 mostram, quantitativamente, os melhores e piores resultados usando o

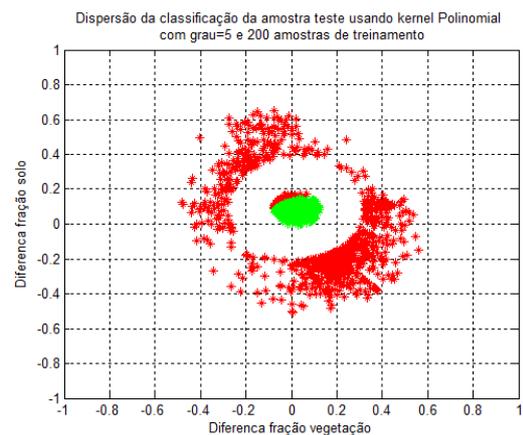
classificador SVM com 200, 300 e 400 amostras de treinamento para as seguintes variáveis: valor da menor pertinência na classe, valor médio de pertinência na classe, o desvio padrão de pertinência na classe e o maior valor de pertinência na classe, além do percentual de *pixels* classificados através da métrica de pertinência corretamente nas classes mudança e não-mudança.

4.10.1.1 Experimentos com kernel Polinomial aplicado à amostra de teste gerada pelo Módulo CVA com origem transladada

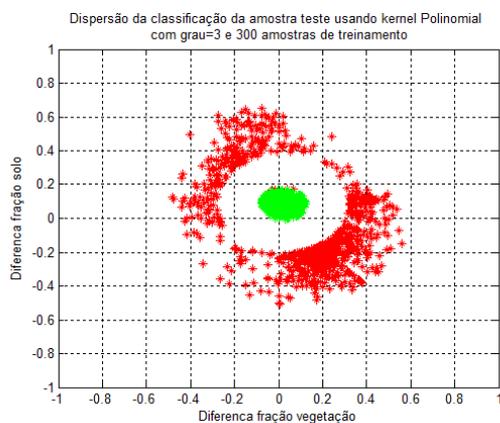
A seguir são apresentados nas Figuras 39 os melhores e piores resultados gráficos de dispersão das estimativas de pertinência dos *pixels* das amostras de testes gerados a partir da metodologia proposta em relação aos resultados esperados para essas amostras utilizando o classificador SVM com 200, 300 e 400 amostras de treinamento, respectivamente.



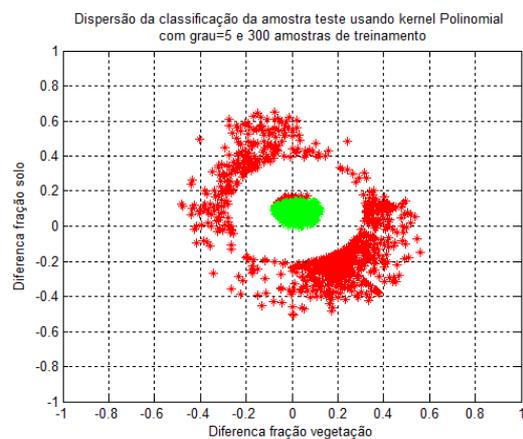
a)



b)



a)



b)

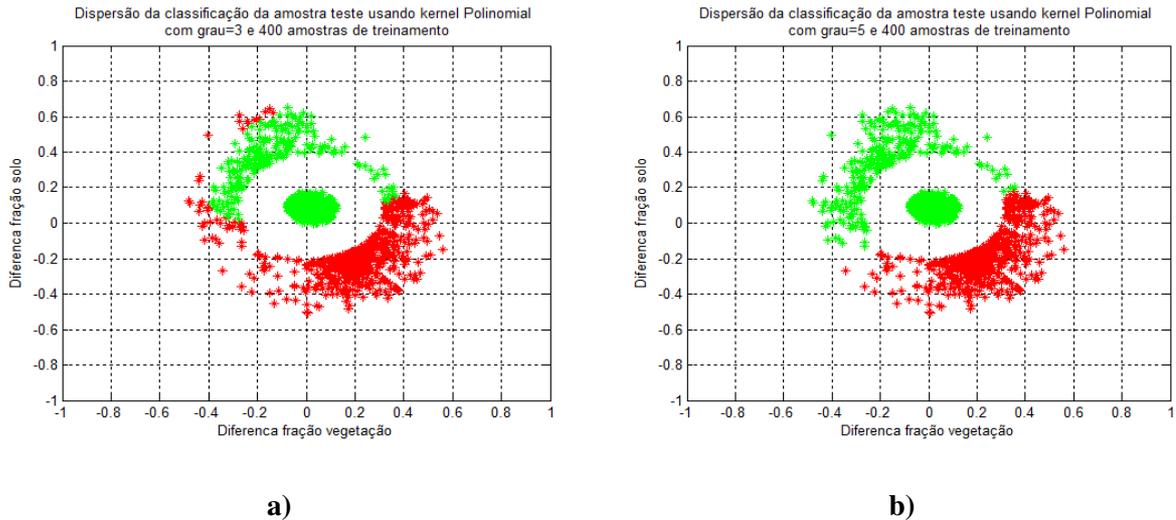


Figura 39 – Classificação da amostra de testes segundo a métrica Delta b usando o classificador SVM com 200, 300 e 400 amostras de treinamento e utilizando *kernel* Polinomial. a) Pior pertinência usando *grau*= 5 e 200 amostras de treinamento em b) melhor pertinência usando utilizando *grau*=2 e 200 amostras de treinamento c) Pior pertinência usando *grau*= 5 e 300 amostras de treinamento d) melhor pertinência usando utilizando *grau*=3 e 300 amostras de treinamento e) Pior pertinência usando *grau*= 2 e 400 amostras de treinamento f) melhor pertinência usando utilizando *grau*=5 e 400 amostras de treinamento. Em todos os gráficos a classe mudança é representada em vermelho e não-mudança em verde.

Fonte: Produzido pelo autor, 2004.

As tabelas 13 e 14 mostram, na ordem, o melhor e pior resultado para cada conjunto de amostra de treinamento utilizada no classificador SVM para as seguintes variáveis: o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e, finalmente, o percentual de *pixels* classificados corretamente nas classes mudança e não-mudança.

Tabela 13 – Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel Polinomial para classe mudança

Número de amostras de treinamento	Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
200	3	51,00	66,39	9,65	100,00	100,00
200	5	52,95	60,49	8,10	100,00	100,00
300	3	51,96	62,51	8,51	100,00	100,00
300	5	51,85	59,69	8,57	100,00	100,00
400	3	0,00	58,69	22,57	100,00	76,78
400	5	0,00	54,76	28,87	100,00	72,56

Fonte: Produzido pelo autor, 2014.

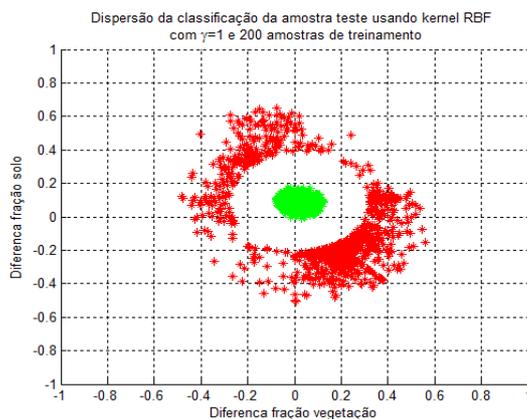
Tabela 14 – Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel Polinomial para classe não-mudança

Número de amostras de treinamento	Grau do <i>kernel</i> polinomial	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
200	3	51,00	90,40	9,08	100,00	100,00
200	5	47,68	76,93	15,86	100,00	88,11
300	3	48,75	85,27	9,77	100,00	99,33
300	5	47,93	78,65	13,30	100,00	95,11
400	3	51,46	73,57	6,60	88,39	100,00
400	5	51,00	64,78	5,06	76,76	100,00

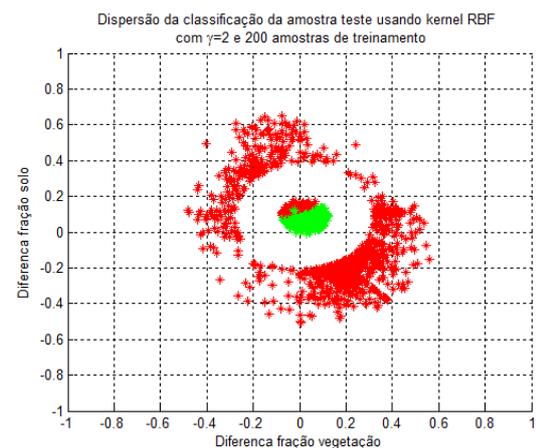
Fonte: Produzido pelo autor, 2014.

4.10.1.2 Experimentos com kernel RBF aplicado à amostra de teste gerado a partir do Módulo CVA com centro transladado

A seguir são apresentados nos gráficos da Figura 40 os melhores e piores resultados gráficos de dispersão das estimativas de pertinência dos *pixels* das amostras de testes gerados a partir da metodologia proposta em relação aos resultados esperados para essas amostras utilizando o classificador SVM com 200, 300 e 400 amostras de treinamento, respectivamente.



a)



b)

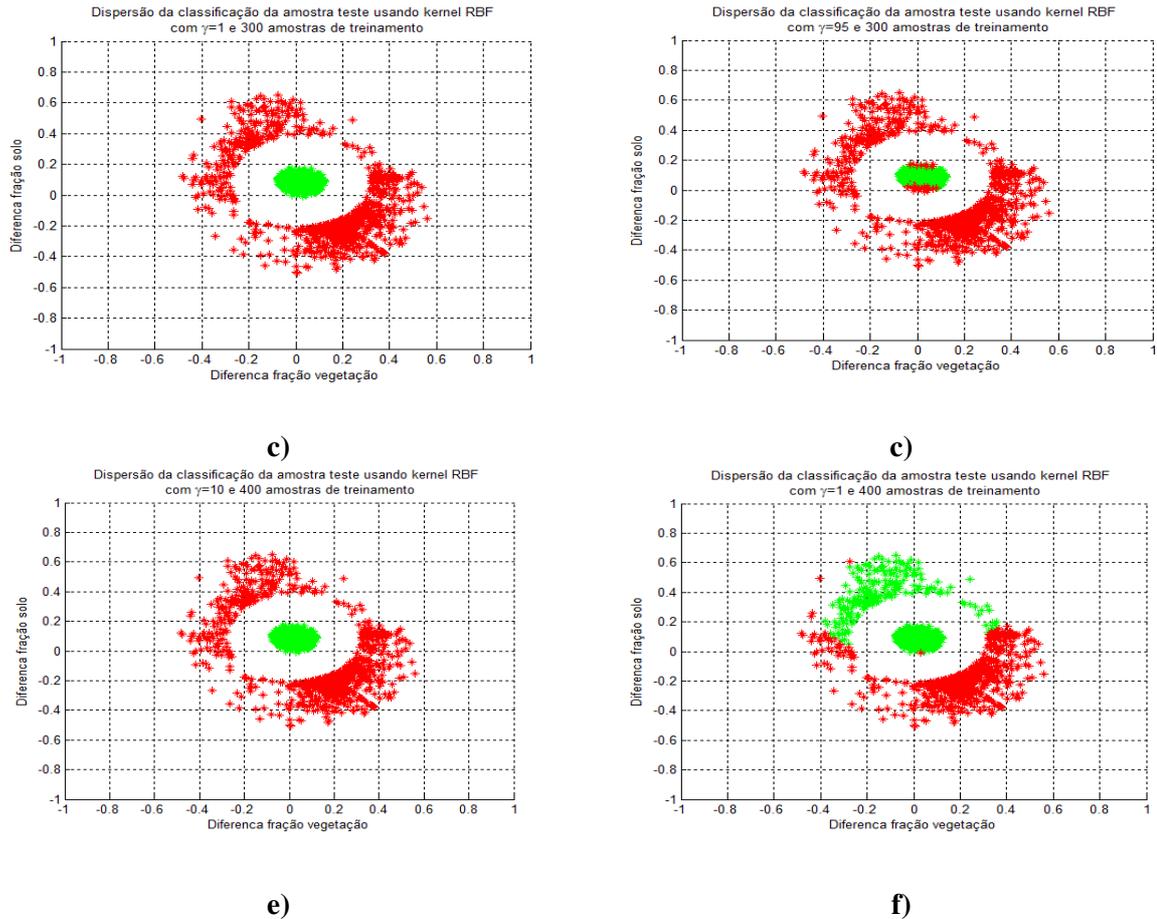


Figura 40 – Classificação da amostra de testes segundo a métrica Delta b usando o classificador SVM com 200, 300 e 400 amostras de treinamento e utilizando *kernel* RBF. a) Pior pertinência usando $\gamma=2$ e 200 amostras de treinamento b) melhor pertinência usando utilizando $\gamma=5$ e 200 amostras de treinamento c) Pior pertinência usando $\gamma=2$ e 300 amostras de treinamento d) melhor pertinência usando utilizando $\gamma=5$ e 300 amostras de treinamento e) Pior pertinência usando $\gamma=2$ e 400 amostras de treinamento f) melhor pertinência usando utilizando $\gamma=5$ e 400 amostras de treinamento. Em todos os gráficos a classe mudança é representada em vermelho e não-mudança em verde.

Fonte: Fonte: Produzido pelo autor, 2004.

As tabelas 15 e 16, a seguir, mostram, na ordem, o melhor e o pior resultado para cada conjunto de amostra de treinamento utilizado no classificador SVM para as seguintes variáveis: o menor valor de pertinência, a média do valor de pertinência, o desvio padrão das pertinências, o maior valor de pertinência e, finalmente, o percentual de *pixels* classificados corretamente nas classes mudança e não-mudança.

Tabela 15 – Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel RBF para classe mudança

Número de amostras de treinamento	Coefficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
200	1	51,00	65,96	9,36	100,00	100,00
200	2	55,06	65,58	9,50	100,00	100,00
300	1	51,00	67,67	10,33	100,00	100,00
300	95	53,12	69,43	9,63	100,00	100,00
400	10	66,16	88,31	8,12	100,00	100,00
400	1	0,00	47,17	30,35	100,00	55,11

Fonte: Produzido pelo autor, 2014.

Tabela 16 – Avaliação da pertinência na amostra de teste gerada a partir do Módulo CVA com centro transladado e usando kernel RBF para classe não-mudança

Número de amostras de treinamento	Coefficiente γ	Menor valor de pertinência na classe (%)	Valor médio de pertinência na classe (%)	Desvio padrão de pertinência na classe (%)	Maior valor de pertinência na classe (%)	Percentual de <i>pixels</i> classificados com pertinência acima de 50% na classe
200	1	51,00	87,70	9,58	100,00	100,00
200	2	46,33	74,31	15,16	100,00	87,56
300	1	51,00	90,00	9,24	100,00	100,00
300	95	48,47	84,55	11,23	100,00	99,67
400	10	39,85	90,54	13,21	100,00	95,89
400	1	42,98	66,52	7,94	75,94	91,56

Fonte: Produzido pelo autor, 2014.

A partir da análise dos gráficos, assim como das tabelas para as pertinências obtidas com os *kernels* polinomial e *RBF* percebe-se que com os parâmetros adequados, os *pixels* da amostra de teste foram corretamente classificados segundo a pertinência informada pela métrica Delta b. Embora nos experimentos com 400 amostras de treinamento tenha ocorrido uma perda de qualidade na determinação da pertinência em relação a esta amostra para ambos o *kernels* utilizados. Esse resultado sugere, novamente, a grande sensibilidade do classificador SVM para amostras com esse tamanho. Entretanto, os resultados, em geral, mostram que a métrica proposta consegue classificar adequadamente os *pixels* da amostra de teste nas classes mudança e não-mudança, segundo as distribuições teóricas das mesmas. Além disso, há

indícios de que o classificador com *kernel* RBF tem um desempenho melhor que o obtido pelo classificador com *kernel* Polinomial; e a métrica de pertinência proposta detecta, com maior acurácia a classe mudança. Finalmente, baseado nos ótimos resultados de pertinência, obtidos com a amostra de teste, pode-se afirmar que a métrica de pertinência Delta b é eficiente, propiciando que se possa seguir para o último experimento envolvendo a imagem real, sobre a qual será realizada uma comparação entre os resultados de pertinência da métrica proposta e os que foram obtidos com o uso de outra métrica de pertinência conhecida e apresentada no trabalho de Zanotta (2010).

Na próxima seção serão apresentados os experimentos envolvendo a imagem real e a comparação de seus resultados de pertinência com a metodologia desenvolvida em Zanotta (2010).

4.11 Experimentos utilizando o classificador SVM com Kernels Polinomial e RBF para imagem real

Assim como os experimentos abordados utilizando o classificador SVM, tanto com o *kernel* polinomial, quanto com o gaussiano (RBF) para as amostras de teste na imagem real foram adotadas as mesmas amostras de treinamento e os mesmos coeficientes e parâmetros utilizados no classificador e nos *kernels*. Contudo, nesses experimentos foram utilizadas como amostras de testes toda a imagem-diferença e os resultados obtidos foram comparados com os resultados apresentados em (ZANOTTA, 2010).

Para todos os experimentos com *kernel* Polinomial foram realizados testes com 200, 300 e 400 amostras de treinamento para cada classe. Os parâmetros utilizados nos experimentos foram: para a variável C foi utilizada o valor 10, para o parâmetro d (grau do *kernel* polinomial), foram realizados testes variando de 2 a 5.

As Figuras 38 até a 42 mostram os mapas de pertinência para esse *kernels* e as tabelas 17 até a 22 mostram a correlação entre as métricas de estimativa de pertinência obtidas nessas propostas, além dos resultados de pertinência apresentados pela métrica adotada em Zanotta (2010). Para todos os testes deste experimento foi utilizado como mapa de referência para comparação entre as metodologias de determinação de pertinência o mapa de pertinência gerado em Zanotta (2010), conforme ilustra a Figura 29.

Nesse mapa, os valores de pertinência são apresentados segundo uma escala de cores variando de 0 a 1, onde o valor 0, representado em azul escuro, representa 0% de pertinência à classe mudança, portanto, 100% de pertinência à classe não-mudança, e o valor 1,

representando 100% de pertinência à classe mudança, ou 0% de pertinência à classe não-mudança. A seguir, nas próximas seções são apresentados os experimentos com *kernel* Polinomial e RBF.

4.11.1 Experimento com kernel polinomial

Nestes experimentos são apresentados nas Figuras 40 até a 42 os mapas de pertinência gerados pela metodologia proposta e a diferença entre esse mapa e o mapa de referência obtido em Zanotta (2010). As tabelas 17 até a 19 mostram o grau do *kernel* polinomial e os valores de correlação entre esses mapas, assim como a média de diferença e desvio padrão entre os eles.

4.11.1.1 Experimento com kernel polinomial utilizando 200 amostras de treinamento

Neste experimento foram realizadas comparações entre os mapas de pertinência gerados pela métrica Delta_b usando o classificador SVM com *kernel* polinomial, utilizando 200 amostras de treinamento e grau do *kernel* polinomial variando de 2 a 5 com os resultados obtidos em Zanotta (2010). A seguir são apresentadas, na tabela 17, as correlações entre os mapas de pertinência gerados pelas duas metodologias e a média de erros e desvio padrão do mapa de diferença entre os dois mapas de pertinência citados.

Tabela 17 – Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel Polinomial e 200 amostras de treinamento

Grau do <i>kernel</i> polinomial	Coefficiente R de Person entre pertinências	Média das diferenças entre pertinências (%)	Desvio padrão das diferenças entre pertinências (%)
2	0,495	19,20	32,01
3	0,3916	21,45	32,66
4	0,3872	21,50	32,78
5	0,3154	21,44	33,05

Fonte: Produzido pelo autor, 2014.

Na Figura 41 são apresentados os mapas de pertinência gerados a partir da metodologia proposta nesta tese e o mapa de diferença entre essa pertinência, além do mapa

de referência (Figura 29) utilizando o classificador *SVM* com 200 amostras de treinamento para o melhor caso de pertinência gerado dentre os graus de kernel polinomial utilizados.

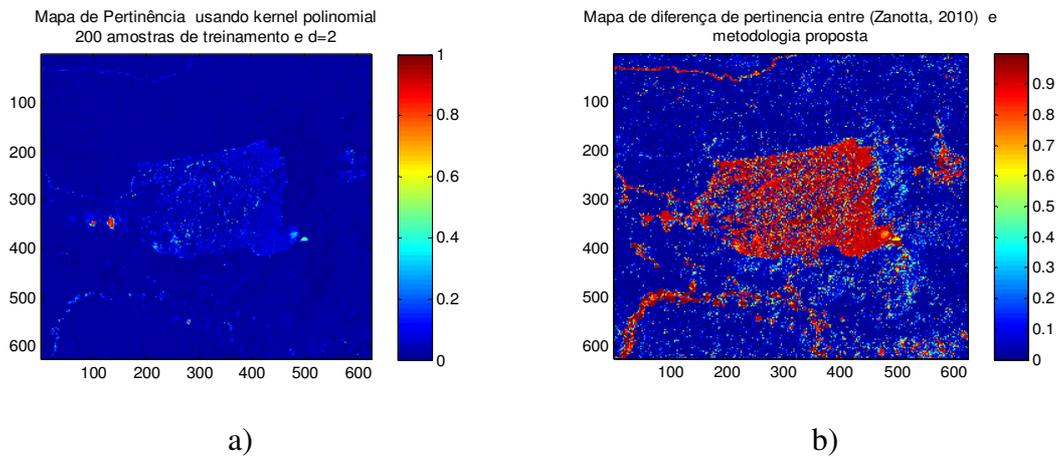


Figura 41 – a) Mapa de Pertinência usando *kernel* polinomial com 200 amostras de treinamento e grau do *kernel* polinomial=2 b) Mapa de diferença de pertinência entre a metodologia proposta em Zanotta (2010) e metodologia proposta neste trabalho.

Fonte: Produzido pelo autor, 2004.

Uma inspeção visual desses mapas de pertinência gerados pela métrica Delta b, mostra resultados inconsistentes aos obtidos em Zanotta (2010). A análise da tabela 17 mostra que o *kernel* polinomial de grau 2 apresenta a melhor correlação dentre todos os testes com 200 amostras de treinamento em relação ao mapa de pertinência gerado por Zanotta (2010). Contudo, o valor da correlação obtida para esse grau é muito baixa (0,495) e, portanto, não apresenta correlação significativa com o mapa de referência utilizado.

4.11.1.2 Experimento com kernel polinomial utilizando 300 amostras de treinamento

De forma semelhante ao que foi feito para o experimento anterior, foram realizadas comparações entre os mapas de pertinência gerados pela métrica Delta_b usando o classificador *SVM* com *kernel* polinomial, utilizando 300 amostras de treinamento e grau do *kernel* polinomial variando de 2 a 5 com os resultados obtidos em Zanotta (2010). A seguir são apresentadas, na tabela 18, as correlações entre os mapas de pertinência gerados pelas duas metodologias e a média de erro e desvio padrão do mapa de diferença entre os dois mapas de pertinência citados.

Tabela 18 – Comparação entre as pertinências obtidas por (Zanotta, 2010) e as obtidas pela metodologia proposta usando kernel Polinomial e 300 amostras de treinamento

Grau do <i>kernel</i> polinomial	Coefficiente R de Person entre pertinências	Média das diferenças entre pertinências (%)	Desvio padrão das diferenças entre pertinências (%)
2	0,4472	20,81	32,29
3	0,3791	21,40	32,72
4	0,3439	21,37	32,89
5	0,3314	21,45	32,98

Fonte: Produzido pelo autor, 2014.

Baseado na tabela acima é apresentado na Figura 42 o melhor resultado obtido da comparação entre as metodologias:

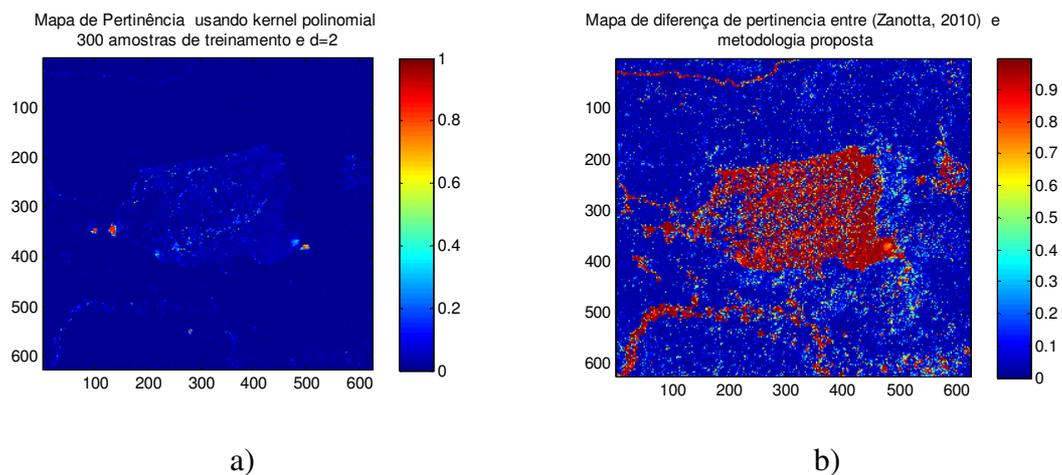


Figura 42 – a) Mapa de Pertinência usando *kernel* polinomial com 300 amostras de treinamento e grau do *kernel* polinomial=2 b) Mapa de diferença de pertinência entre (Zanotta, 2010) e metodologia proposta

Fonte: Produzido pelo autor, 2004.

Novamente, a partir de uma inspeção visual dos mapas de pertinência gerados pela métrica Delta_b, com a utilização do classificador SVM com *kernel* polinomial para 300 amostras de treinamento e grau o *kernel* polinomial variando de 2 a 5, fica claro que esses resultados de pertinência são inconsistentes com os obtidos em Zanotta (2010). A análise da tabela 18 aponta que o *kernel* polinomial de grau 2 apresenta a melhor correlação dentre todos os testes com 300 amostras em relação ao mapa de pertinência gerado por Zanotta (2010). Contudo, o valor da correlação obtido para o mapa de pertinência gerado para esse grau é ainda muito baixo ($R=0,4472$) e, portanto, não apresenta correlação significativa entre o mapa

de referência utilizado. Em geral, a média de erro em relação às medidas do mapa de referência é próxima de 21 %, com desvio padrão de 32% para esse teste.

4.11.1.3 Experimento com kernel polinomial utilizando 400 amostras de treinamento

Semelhantemente aos experimentos anteriores, foram realizadas comparações entre os mapas de pertinência gerados pela métrica Delta_b, com a utilização do classificador SVM, com kernel polinomial para 400 amostras de treinamento e grau o kernel polinomial variando de 1 a 5, com os resultados os obtidos em Zanotta (2010). A seguir são apresentadas, na tabela 19, as correlações entre os mapas de pertinência gerados pelas duas metodologias e a média de erros e desvio padrão do mapa de diferença entre os dois mapas de pertinência citados.

Tabela 19 – Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel Polinomial e 400 amostras de treinamento

Grau do <i>kernel</i> polinomial	Coefficiente R de Person entre pertinências	Média das diferenças entre pertinências (%)	Desvio padrão das diferenças entre pertinências (%)
2	0,4358	6,04	31,96
3	0,4955	16,27	31,92
4	0,5031	20,12	31,93
5	0,4606	13,72	31,83

Fonte: Produzido pelo autor, 2014.

Baseado na tabela acima é apresentado na Figura 43 o melhor resultado obtido na comparação entre as metodologias:

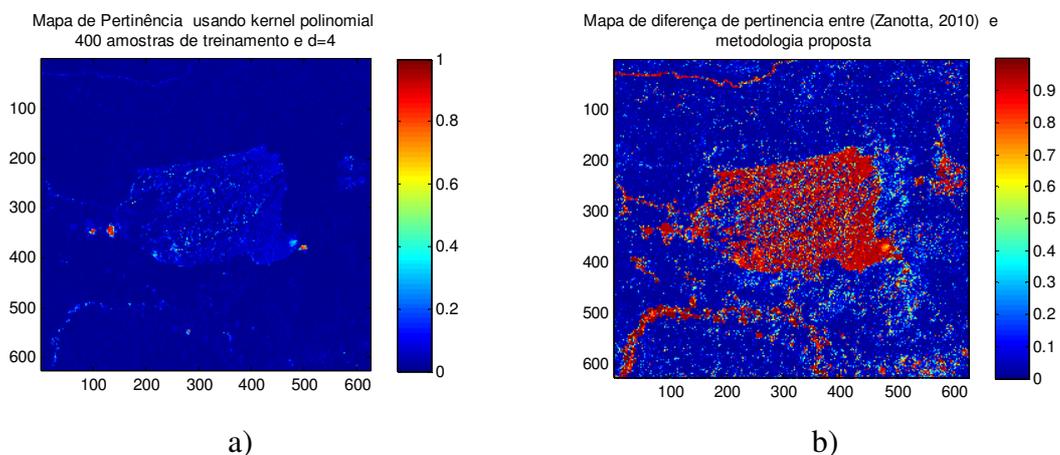


Figura 43 – a) Mapa de Pertinência usando *kernel* polinomial com 400 amostras de treinamento e grau do *kernel* polinomial=4 b) Mapa de diferença de pertinência entre Zanotta, (2010) e metodologia proposta.

Fonte: Produzido pelo autor, 2004.

De forma análoga, a análise feita nos experimentos anteriores, uma inspeção visual dos mapas de pertinência gerados pela métrica Delta_b, com a utilização do classificador SVM, com *kernel* polinomial para 400 amostras de treinamento e grau do *kernel* polinomial variando de 1 a 5, mostrou que resultados obtidos pela metodologia proposta são inconsistentes com os obtidos em Zanotta (2010). A análise da tabela 19 aponta que o *kernel* polinomial de grau 4 apresenta a melhor correlação dentre todos os testes com 400 amostras. Contudo, o valor de correlação obtido para esse grau é ainda muito baixo ($R=0,5031$) e, portanto, não apresenta correlação significativa em relação ao mapa de referência utilizado. Em geral, a média de erro em relação às medidas do mapa de referência é próxima de 20%, com desvio padrão de aproximadamente 32% para esse teste.

De forma geral, as pertinências obtidas com o *kernel* polinomial não são compatíveis com as obtidas pela metodologia proposta em Zanotta (2010).

4.11.2 Experimento com *kernel* RBF

Para todos os experimentos com *kernel* RBF foram realizados testes com 200, 300 e 400 amostras de treinamento para cada classe. Os parâmetros utilizados nos experimentos foram: para a variável C foi utilizado o valor 10, para o parâmetro γ foram utilizados os valores 1, 2, 5, 10, 80, 85, 90, 95 e 100.

A partir desses experimentos, são apresentados nas Figuras 43 até a 45 os mapas de pertinência gerados pela metodologia proposta e a diferença entre esse mapa e o mapa de referência obtido em Zanotta (2010), além das tabelas 20 até a 22, que mostram o valor γ do *kernel* RBF e os valores de correlação entre esses mapas, assim como a média de diferença e desvio padrão entre eles.

4.11.2.1 Experimento com *kernel* RBF com 200 amostras de treinamento

Para esse experimento foram realizadas comparações entre os mapas de pertinência gerados pela métrica Delta_b, proposta apresentada nesta metodologia, usando o classificador SVM com *kernel* RBF para 200 amostras de treinamento e γ com os seguintes valores: 1, 2, 5, 10, 80, 85, 90, 95 e 100, com os resultados obtidos em Zanotta (2010).

A seguir são apresentados, na tabela 20, os valores do coeficiente γ do *kernel* RBF, os valores de correlação entre esses mapas, assim como a média de diferença e desvio padrão entre eles.

Tabela 20 – Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando *kernel* RBF e 200 amostras de treinamento

Coeficiente γ do <i>kernel</i> RBF	Coeficiente R de Person entre pertinências	Média das diferenças entre pertinências (%)	Desvio padrão das diferenças entre pertinências (%)
1	0,5888	20,09	30,78
2	0,5037	17,74	30,78
5	0,6609	15,01	27,79
10	0,8038	11,87	23,59
80	0,9554	-1,85	12,11
85	0,9281	-4,49	14,21
90	0,9396	-2,95	13,52
95	0,9056	-7,57	16,02
100	0,9364	-5,07	14,24

Fonte: Produzido pelo autor, 2014.

Baseado na tabela acima, são apresentados na Figura 44 os melhores resultados obtidos na comparação entre as metodologias:

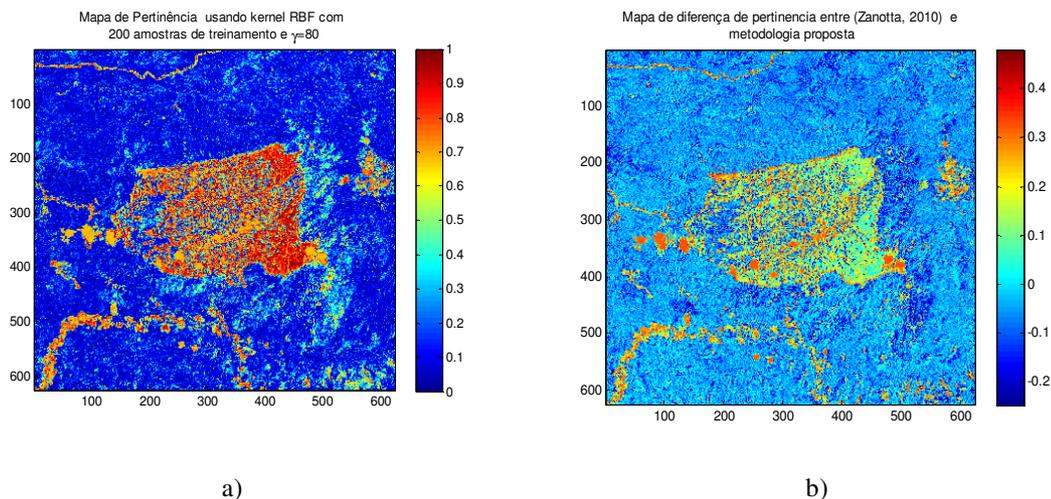


Figura 44 – a) Mapa de Pertinência usando *kernel* RBF com 200 amostras de treinamento e $\gamma=80$ b) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta.

Fonte: Produzido pelo autor, 2004.

Uma inspeção visual dos mapas de pertinência gerados pela métrica Delta_b usando o classificador SVM com *kernel* RBF com 400 amostras de treinamento mostrou que resultados

obtidos pela metodologia proposta são muito semelhantes aos obtidos em Zanotta (2010). A análise da tabela 20 aponta que o *kernel* RBF com $\gamma = 80$ apresenta a melhor correlação dentre todos os testes com 200 amostras de treinamento. Sendo o valor de correlação obtido entre as metodologias, para esse coeficiente, bem alto ($R=0,9554$), apresentando uma correlação significativa com o mapa de referência utilizado. Em geral, a média de erro em relação às medidas do mapa de referência é de cerca de 1,85%, com desvio padrão de 12%.

De forma geral, as pertinências obtidas com o *kernel* RBF, com a utilização de 200 amostras de treinamento, são compatíveis às obtidas em Zanotta (2010).

4.11.2.2 Experimento com *kernel* RBF com 300 amostras de treinamento

Para esse experimento foram realizadas comparações entre os mapas de pertinência gerados pela métrica Delta_b, com a utilização do classificador SVM, com *kernel* RBF para 300 amostras de treinamento com os resultados obtidos em Zanotta (2010). A seguir são apresentadas, na tabela 21, as correlações entre os mapas de pertinência gerados pelas duas metodologias e a média de erro e desvio padrão do mapa de diferença entre os dois mapas de pertinência citados.

Tabela 21 – Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando *kernel* RBF e 300 amostras de treinamento

Coeficiente γ do <i>kernel</i> RBF	Coeficiente R de Person entre pertinências	Média das diferenças entre pertinências (%)	Desvio padrão das diferenças entre pertinências (%)
1	0,5346	20,01	31,25
2	0,5367	17,81	30,39
5	0,7947	14,15	23,82
10	0,841	10,04	20,99
80	0,967	-0,97	10,15
85	0,9675	-1,53	9,79
90	0,9728	-1,76	9,22
95	0,961	-4,46	10,86
100	0,961	-3,56	10,38

Fonte: Produzido pelo autor, 2014.

Baseado na tabela acima é representado na Figura 45 o melhor resultado obtido na comparação entre as metodologias:

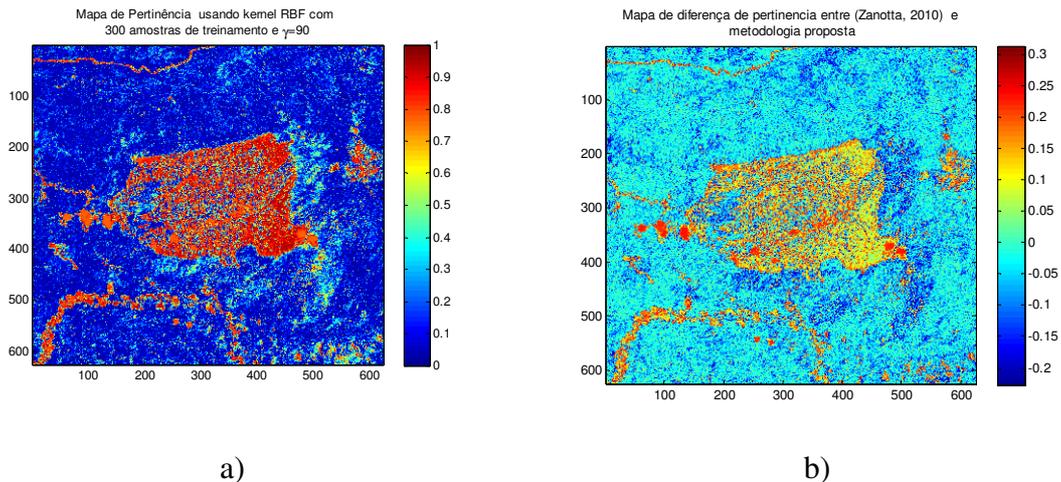


Figura 45 – a) Mapa de Pertinência usando *kernel* RBF 300 amostras de treinamento e $\gamma=90$
 b) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta.
 Fonte: Produzido pelo autor, 2004.

De forma análoga, a análise feita nos experimentos anteriores, uma inspeção visual dos mapas de pertinência gerados pela métrica Delta b, com a utilização do classificador *SVM* com *kernel* RBF, com 300 amostras de treinamento, mostrou que os resultados obtidos pela metodologia proposta são muito semelhantes aos obtidos em Zanotta (2010). A análise da tabela 21 aponta que o *kernel* RBF com $\gamma=90$ apresentou a melhor correlação dentre todos os testes com as 300 amostras de treinamento. Sendo o valor de correlação, obtido para esses coeficientes, um valor bem alto entre as metodologias ($R=0,9728$) e, portanto, apresenta correlação significativa quando comparado ao mapa de referência dado por Zanotta (2010). Em geral, a média de erro em relação às medidas do mapa de referência é próxima de 1,76 %, com desvio padrão de aproximadamente 9% para esse teste.

De forma geral, as pertinências obtidas com o *kernel* RBF são compatíveis às obtidas em Zanotta (2010).

4.11.2.3 Experimento com *kernel* RBF com 400 amostras de treinamento

Para esse experimento foram realizadas comparações entre os mapas de pertinência gerados pela métrica Delta_b, com a utilização do classificador *SVM*, com *kernel* RBF para 400 amostras de treinamento, com os resultados obtidos em Zanotta (2010). A seguir são apresentadas, na tabela 22, as correlações entre os mapas de pertinência gerados pelas duas metodologias e a média de erros e desvio padrão do mapa de diferença entre os dois mapas de pertinência citados.

Tabela 22 – Comparação entre as pertinências obtidas por Zanotta (2010) e as obtidas pela metodologia proposta usando kernel RBF e 400 amostras de treinamento

Coefficiente γ do <i>kernel</i> RBF	Coefficiente R de Person entre pertinências	Média das diferenças entre pertinências (%)	Desvio padrão das diferenças entre pertinências (%)
1	0,5769	10,55	29,85
2	0,6766	12,90	27,91
5	0,7973	13,10	23,62
10	0,8556	9,76	19,99
80	0,9685	-2,37	9,44
85	0,9739	-1,22	8,68
90	0,9699	-1,72	9,13
95	0,9714	-1,72	8,78
100	0,9739	-2,13	8,31

Fonte: Produzido pelo autor, 2014.

Baseado na tabela acima, são representados na Figura 46 os melhores resultados obtidos na comparação entre as metodologias:

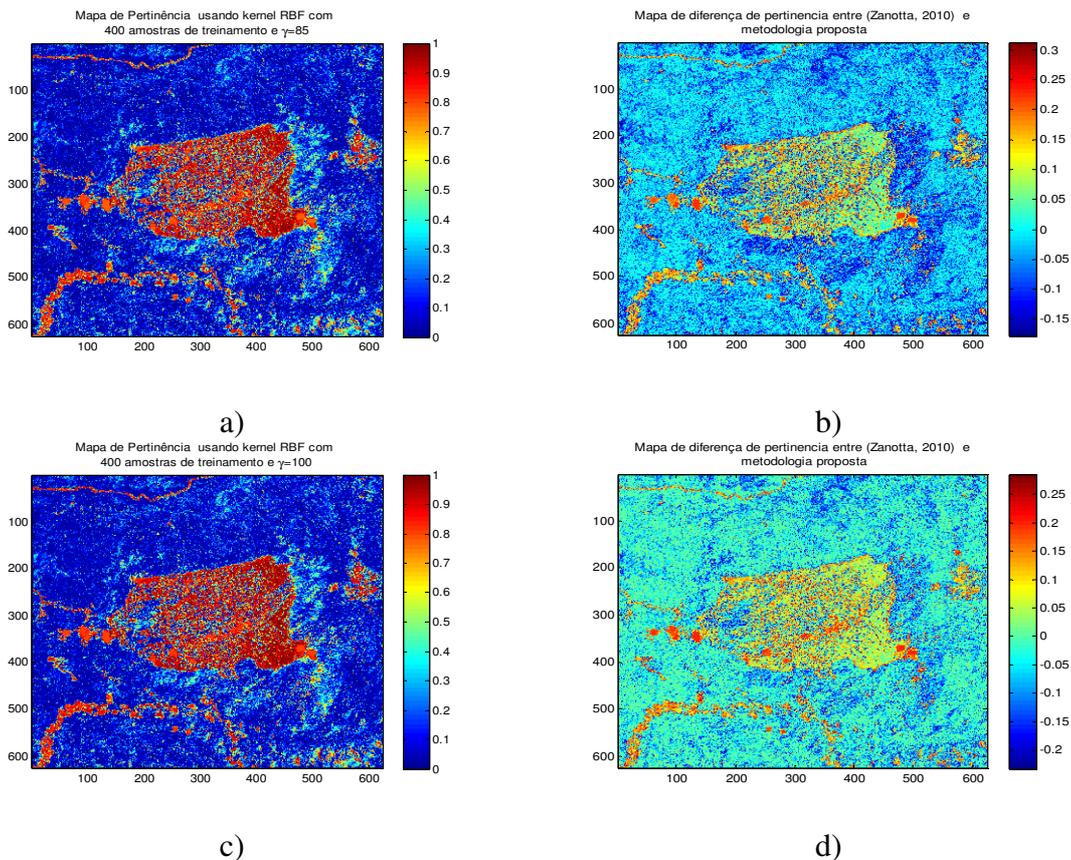


Figura 46 – a) Mapa de Pertinência usando *kernel* RBF com 400 amostras de treinamento e $\gamma=85$ b) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta. c) Mapa de Pertinência usando *kernel* RBF com 400 amostras de treinamento e $\gamma=100$ d) Mapa de diferença de pertinência entre Zanotta (2010) e metodologia proposta.

Fonte: Produzido pelo autor, 2004.

De forma análoga, a análise feita nos experimentos anteriores, uma inspeção visual dos mapas de pertinência gerados pela métrica Delta b, com a utilização do classificador *SVM*, com *kernel* RBF com 400 amostras de treinamento, mostrou que os resultados obtidos pela metodologia proposta são muito semelhantes aos obtidos em Zanotta (2010). A análise da tabela 22 aponta que os *kernels* RBF com $\gamma= 85$ e $\gamma= 100$ apresentaram a melhor correlação dentre todos os testes com as 400 amostras de treinamento. Sendo os valores de correlação, obtidos para esses coeficientes, bem altos e de mesmo valor ($R=0,9739$), apresentam, portanto, correlação significativa quando comparados ao mapa de referência utilizado. Em geral, a média de erro em relação às medidas do mapa de referência é próxima de -1,22%, com desvio padrão de aproximadamente 8,68 % para ambos os coeficientes.

De forma geral, as pertinências obtidas com o *kernel* RBF são compatíveis às obtidas em Zanotta (2010).

5 CONCLUSÕES

A partir dos resultados obtidos nesta tese, espera-se contribuir de uma forma significativa para o aperfeiçoamento da área de reconhecimento de padrões, em especial da detecção de mudanças em imagens de sensoriamento remoto. Para tanto, este trabalho investigou uma abordagem não supervisionada para o problema da detecção de mudanças em imagens multiespectrais e multitemporais de sensoriamento remoto, empregando SVM com o uso dos *kernels* polinomial e gaussiano (RBF), usado para gerar uma nova métrica de pertinência. A proposta metodológica foi baseada na diferença das imagens-fração produzidas para cada data. A característica de que em imagens de cenas naturais a diferença nas frações de solo e vegetação tendem a apresentar uma distribuição simétrica próxima à origem possibilitou a modelagem das distribuições normais multivariadas das classes mudança e não-mudança presentes na imagem; e o algoritmo EM foi implementado para estimar os parâmetros estatísticos associados a essas duas distribuições. Na sequência amostras aleatórias e normalmente distribuídas foram extraídas dessas distribuições e rotuladas segundo sua probabilidade de pertinência a cada uma das classes. Essas amostras foram então usadas no treinamento do classificador SVM e do resultado dessa classificação foi estimada uma nova métrica de pertinência de pixels chamada Delta b. Para essa metodologia foram executados dois grandes experimentos. No primeiro experimento foi utilizada uma amostragem de teste selecionada a partir da métrica consagrada do módulo do vetor de mudança (Módulo CVA) como o objetivo de validar quantitativamente a metodologia proposta. Contudo durante a execução deste experimento foi constatado que o módulo CVA poderia ser adaptado às características presentes na imagem original e, dessa forma, um outro experimento semelhante envolvendo a imagem módulo CVA foi desenvolvido, agora executando um deslocando da origem (0,0) do eixo dos vetores CVA para o valor da média da classe não-mudança. Finalmente, um último experimento foi executado, dessa vez envolvendo a classificação com SVM e aplicação da métrica de pertinência proposta neste trabalho a todos os *pixels* da imagem em estudo. Os resultados desta estimação de pertinência foram, posteriormente, comparados aos resultados de pertinência obtidos em Zanotta (2010) para a mesma imagem. Esta comparação teve a finalidade de verificar a similaridade de ambas as métricas, pois, embora os teste do experimento anterior pudessem comprovar estatisticamente que os resultados obtidos eram aceitáveis. A comparação do resultado de pertinência proposto com as obtidas por outro método apresentado na literatura e aplicada a mesma imagem em estudo corrobora a qualidade da métrica proposta.

Para a avaliação da acurácia dos resultados obtidos no primeiro e segundo experimentos foi utilizada a contagem percentual dos *pixels* amostrados com grau de pertinência superior a 50% o que caracteriza que esses *pixels* de fato pertencem à classe em que foi classificado. No último experimento foi calculada a correlação entre os mapas de pertinência obtidos por esta metodologia e o obtido em (Zanotta, 2010). Além desse valor, foram apresentados o valor médio do erro da diferenças entre as métricas e o correspondentes desvio padrão deste erro com a finalidade de comparar as métricas.

Esse último experimento mostrou que a métrica de pertinência proposta na metodologia é adequada à detecção de alterações em imagens onde prevaleçam os *endmembers* solo e vegetação e os resultados de pertinência são bem semelhantes aos produzidos com o classificador de MVG e abordagem fuzzy, desenvolvida em Zanotta (2010).

Baseado nestes resultados, os quais mostram que a métrica para determinação de pertinência proposta é válida, apresenta resultados compatíveis com outra técnica de pertinência publicada na literatura e, além disso, que os resultados obtidos pelo trabalho proposto utilizou poucas amostras de treinamento, uma das característica do classificador de SVM, espera-se que essa métrica de pertinência deva apresentar melhores resultados dos que seriam apresentados com classificadores paramétricos, como por exemplo, o MGV, quando aplicado a imagens multitemporais e hiperespectrais, pois o classificador SVM não sofre os efeitos do fenômeno de Hughes.

Contudo, é importante salientar neste momento que o método proposto visa a detectar mudanças em ambientes onde prevaleçam os *endmembers* vegetação e solo. Sendo assim, não é possível afirmar que esta metodologia, ao ser aplicada a outros ambientes com diferentes *endmembers*, possa apresentar resultados semelhantes. Principalmente em função das suposições realizadas a respeito das distribuições normais das classes mudança e não-mudança para essas componentes as quais foram determinantes para que se pudesse utilizar o algoritmo EM e, conseqüentemente, fosse feita a geração das amostras de treinamento usada no classificador SVM.

O método proposto nesse estudo visa detectar mudanças em ambientes onde há prevalência das frações vegetação e solo. Sendo assim, não há garantias quanto ao seu uso para outros ambientes com diferentes componentes, principalmente ao que se refere a determinação dos valores iniciais do algoritmo EM.

5.1 Sugestões para trabalhos futuros

Em virtude dos bons resultados obtidos por esta metodologia em imagens multitemporais e multiespectrais, com a utilização de poucas amostras de treinamento, uma característica importante do classificador de *SVM*, sugere-se que seja realizada uma investigação dessa abordagem metodológica para detecção de mudança em imagens hiperespectrais.

Além disso, conforme mencionado anteriormente, este trabalho investigou uma abordagem para detectar mudanças em ambientes onde há o prevaletimento de vegetação e solo. Assim, também sugere-se que para futuros estudos sejam investigados quais outros componentes (*endmembers*) seriam adequados para analisar mudança em imagens de sensoriamento remoto.

REFERÊNCIAS

- ABE, S. *Support Vector Machines for Pattern Classifications*. Kobe, Japão: Ed. Springer, 2005.
- ANDREOLA, R. **Support Vector Machines na Classificação de Imagens Hiperespectrais**. Dissertação de Mestrado, Programa de Pós-Graduação em Sensoriamento Remoto – UFRGS. Porto Alegre, 2009.
- ANGELO, N. P.. **Aplicação de Filtros de Gabor no Processo de Classificação de Imagens Digitais com Base em Atributos de Textura**. Dissertação de Mestrado, Programa de Pós-Graduação em Sensoriamento Remoto – UFRGS. Porto Alegre, 2000.
- ARONSZAJN, N. Theory of reproducing *kernels*. **Trans. Am. Math. Soc.** (68):337–404, 1950.
- BAZI, Y.; BRUZZONE, L.; MELGANI, F., An unsupervised approach based on the generalized Gaussian model to automatic change detection in multitemporal SAR images, *IEEE Trans. Geosci. Remote Sens.*, vol. 43, no. 4, pp. 874-887, 2005.
- BITTENCOURT, H. R. **Detecção de Mudanças a partir de Imagens de Fração**. Tese de Doutorado, Programa de Pós-Graduação em Geografia – UFRGS. Porto Alegre, 2011.
- BONESSO, D. **Estimação dos Parâmetros do Kernel em um Classificador SVM na Classificação de Imagens Hiperespectrais em uma Abordagem Multiclasse**. Dissertação de Mestrado, Programa de Pós-Graduação em Sensoriamento Remoto – UFRGS. Porto Alegre, 2013.
- BOSC, M.; HEITZ, F.; ARMSPACH, J. P.; NAMER, I.; GOUNOT, D.; RUMBACH, L., Automatic change detection in multimodal serial MRI: Application to multiple sclerosis lesion evolution, **Neuroimage**, vol. 20, pp. 643–656, 2002.

BOVOLO, F.; MARIN, C.; BRUZZONE, L.. A novel hierarchical approach to change detection with very high resolution SAR images for surveillance applications, **Geoscience and Remote Sensing Symposium (IGARSS)**, 2012, 1992 – 1995.

BOVOLO, F. E.; BRUZZONE, L. A Theoretical Framework for Unsupervised Change Detection Based on Change Vector Analysis in the Polar Domain. **IEEE Transactions on Geoscience and Remote Sensing**, v. 45, n. 1, 2007.

BOVOLO, F.; BRUZZONE, L.; MARCHESI, S. A Multiscale Technique for Reducing Registration Noise in Change Detection on Multitemporal VHR images. In: **MultiTemp 2007. Proceedings.....** Leuven: IEEE, 2007.

BOVOLO, F.; BRUZZONE, L.; MARCONTINI, M., Novel Approach to Unsupervised Change Detection Based on a Semisupervised SVM and a Similarity Measure, **IEEE Trans. Geosci. Remote Sens.**, vol. 46, no. 7, pp. 2070-2082, 2008.

BOVOLO, F.; CAMPS-VALLS, G.; BRUZZONE, L. A Support Vector Domain Method for Change Detection in Multitemporal Images. **Pattern Recognition Letters**. Vol. 31, N°. 10, p. 1148–1154, 2010.

BOW, S. T. **Pattern Recognition and Image Preprocessing**. Marcel Dekker: New York, 1992.

BRERETON, R. G.; LLOYD, G. R.. Support Vector Machines for classification and regression. *Analyst*, **Journal of the Royal Society of Chemistry**, 2010, no 135, pp 230–267.

BRUZZONE L.; SMITS, P. C.; TILTON, J. C. Foreword Special Issue on Analysis of Multitemporal Remote Sensing Images. **IEEE Transactions on Geoscience and Remote Sensing**, v. 41, n. 11, pp. 2419-2422, 2003.

BRUZZONE, L.; COSSU, R. An adaptive approach to reducing registration noise effects in unsupervised change detection. **IEEE Transactions on Geoscience and Remote Sensing**, Vol. 41, Issue 11, Part 1, Nov. 2003, pp.2455 – 2465.

BRUZZONE, L.; COSSU, R.; VERNAZZA, G. Detection of Land-cover Transitions by Combining Multidate Classifiers. **Pattern Recogniton Letters**. Vol. 25, N°. 13, p. 1491–1500, 2004.

BRUZZONE, L.; PRIETO, D. F., Automatic Analysis of the Difference Image for Unsupervised Change Detection, **IEEE Trans. Geosci. Remote Sens.**, vol. 38, no. 3, pp. 1171-1182, 2000.

BRUZZONE, L.; SERPICO S. B. An Iterative Technique for the detection of land-cover transitions in multitemporal remote sensing images, **IEEE Transaction on Geoscience and Remote Sensing**, v. 35, n. 4, p. 858-867, 1997.

BRUZZONE, L.; PRIETO, D. F., An adaptive semiparametric and context-based approach to unsupervised change detection in multitemporal remote-sensing images, **IEEE Trans. Image Processing**, vol. 11, no. 4, pp. 452–466, Apr. 2002.

BURGES, C. J. C. **A tutorial on support vector machines for pattern recognition.** Knowledge Discovery and Data Mining, pp 121-167, 1998,

CAMPBELL, C. *Kernel Methods: a survey of current techniques.* In: **Advances in Neural Information Processing Systems**, pages 395-401. MIT Press, 2002.

CAMPS-VALLS, G.; GÓMEZ-CHOVA, L.; MUÑOZ-MARÍ, J.; ROJO-ÁLVAREZ, J. L.; MARTÍNEZ-RAMÓN, M.. Kernel-Based Framework for Multitemporal and Multisource Remote Sensing Data Classification and Change Detection. **IEEE Transactions On Geoscience And Remote Sensing**, Vol. 46, No. 6, 2008.

CHEN, J.H. M- estimator based robust *Kernel* for support vector Machines. **In proceedings of 17th International conference on Pattern recognition**, Vol 1,168-171, Cambridge, UK, 2004.

COLLINS, R., LIPTON, A.; KANADE,T., Introduction to the special section on video surveillance, **IEEE Trans. Pattern Anal. Mach. Intell.**, vol. 22, no. 8, pp. 745-746, Aug. 2000.

DAI, X.; KHORRAM, S.. The effects of image misregistration on the accuracy of remotely sensed change detection. **IEEE Transactions on Geoscience and Remote Sensing**, Vol 36, Issue 5, Part 1, Sept. 1998, pp.1566 – 1577.

DEMPSTER, A. P.; LAIRD, N. M.; RUBIN, D. B. Maximum Likelihood from incomplete data via the EM algorithm, **Journal of the Royal Statistical Society**, vol. 39, n. 1, 1977, pp. 1–38.

DUDA, R. O.; HART, P. E.; STORK, D. G. **Pattern Classification**. 2nd edition, New York: John Wiley & Sons, 2001.

DUMSKYJ, M. J.; ALDINGTON, S. J.; DORE, C. J.; KOHNERE. M.. The accurate assessment of changes in retinal vessel diameter using multiple frame electrocardiograph synchronised fundus photography, **Current Eye Res.**, vol. 15, no. 6, pp. 652–632, 1996.

FERREIRA, R. H. DA S.; HAERTEL, V.. Detecção De Mudanças Em Imagens Multitemporais De Sensoriamento Remoto Empregando Support Vector Machines, **XVI Simpósio Brasileiro de Sensoriamento Remoto - SBSR**, 2013.

HAERTEL, V., SHIMABUKURO, Y. E., And ALMEIDA-FILHO, R.. Fraction Images in Multitemporal Change Detection, **International Journal of Remote Sensing**, v. 25, n. 23, pp. 5473-5489, 2004.

HABIB, T.; INGLADA, J.; MERCIER, G.; CHANUSSOT, J.. Support Vector Reduction in SVM Algorithm for Abrupt Change Detection in Remote Sensing. **Ieee Geoscience And Remote Sensing Letters**, Vol. 6, N°. 3, 2009.

HAERTEL, V.; SHIMABUKURO, Y. E.. Spectral Linear Mixing Model in Low Spatial Resolution Image Data, **IEEE Trans. Geosci. Remote Sens.**, vol. 43, no. 11, pp. 2555-2562, 2005.

HAERTEL, V.; SHIMABUKURO, Y. E.; ALMEIDA FILHO, R.. Fraction Images in Multitemporal Change Detection. **International Journal of Remote Sensing**, v. 10, n. 23, p. 5473-5489, 2004.

HAMEL, Lutz. **Knowledge Discovery With Support Vector Machines**. John Wiley & Sons Ltd, 2009.

HUERTAS, A.; NEVATIA, R.. Detecting changes in aerial views of manmade structures, **Image Vis. Comput.**, vol. 18, no. 8, pp. 583–596, 2000.

KAN, W. Y.; KROGMEIER, J. V.; DOERSCHUK, P. C.. Model-based vehicle tracking from image sequences with an application to road surveillance, **Opt. Eng.**, vol. 35, no. 6, pp. 1723–1729, 1996.

LANDGREBE, D. **On Information Extraction Principles for Hyperspectral Data**. [S.I.: s.n.] 1997.

LANDGREBE, D.; BIEHL, L., **Multi Spec, School of Electrical and Computer Engineering**, Purdue University, West Lafayette, Indiana, USA, 2002.

LILLESAND, T.; KIEFER, R. W. **Remote sensing and image interpretation**. 3rd ed. New York: John Wiley & Sons, 1994.

LILLESTRAND, R. L. Techniques for change detection. **IEEE Transactions on Computers**, 21, p. 654-659, 1972.

LU, D.; MAUSEL, P.; BRONDIZIO, E.; MORAN, E. Change Detection Techniques. **International Journal of Remote Sensing**. Vol. 25, N°. 12, p. 2365–2407, 2003.

LUENBERGER, D. **Linear and Nonlinear Programming**. Addison-Wesley, 1984.

MALILA, W. A. Change Vector Analysis: An Approach for Detecting Forest Changes with Landsat, **Proceedings of the LARS Machine Processing of Remotely Sensed Data Symposium**, pp. 326-336, 1980.

MERCER, J.. **Functions of positive and negative type and their connection with the theory of integral equations**. Philosophical Transactions of the Royal Society A 209, 1909.

MICHALEK, J. L.; LUCZKOVICH, J. J. Multispectral Change Vector Analysis for Monitoring Coastal Marine Environments, **Photogrammetric Engineering and Remote Sensing**, v. 59, pp. 381-384, 1993.

MITCHELL, T. M.; **Machine learning**. McGraw-Hill Series in computer science, 1 edition, 1997.

MOREIRA, M. A. **Fundamentos do sensoriamento remoto e metodologias de aplicação**. Viçosa: Ed. UFV, 2011.

MUÑOZ-MARÍ, J.; GÓMEZ-CHOVA, L.; RAMÓN, M.; ROJO-ÁLVAREZ, J.; CALPE-MARAVILLA, J.; CAMPS-VALLS, G. Multi-temporal image classification with kernels. In:

Camps-Valls, G.; Bruzzone, L. (org.). **Kernel Methods for Remote Sensing Data Analysis**. A John Wiley & Sons, Inc., Publication, 2009.

NEMMOUR, H.; CHIBANI, Y. Multiple support vector machines for land cover change detection: An application for mapping urban extensions. **ISPRS Journal of Photogrammetry & Remote Sensing**. 61, 125–133, 2006.

PLATT, J. Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. Em: SMOLA, A.; BARTLETT, P.; SCHÖLKOPF, B. e SCHUURMANS, D. **Advances in Large Margin Classifiers**. 2000.

PONTIL, M.; VERRI, A. Support Vector Machines for 3D Object Recognition, **IEEE Trans. Pattern Analysis and Machine Intelligence**, Vol. 20, No. 6, pp. 637-646, 1998.

RADKE, R. J.; ANDRA, S.; AL-KOFAHI, O.; ROYSAM, B. Image Change Detection Algorithms: a systematic survey. **IEEE Transactions on Image Processing**, v. 14, n. 3, p. 294–307, 2005.

RESEARCH. Disponível em: <http://research.microsoft.com/en-us/groups/vgv/vgv4.png>. Acesso em: Julho de 2013.

ROSENFELD, A. Automatic detection of changes in reconnaissance data”. **Proceedings... 5th Conv. Mil. Electron.**, 1961, p. 492.

SCHÖLKOPF, B. SMOLA, A.J. **Learning with Kernels**. MIT Press, 2002.

SHAWE-TAYLOR, J.; CRISTIANINI, N. **Kernel Methods for Pattern Analysis**. Cambridge University Press, UK, 2004.

SHEPARD, J. R. A concept of change detection. **Photogrammetric Engineering and Remote Sensing**, vol. 30, pp. 648, July 1964.

SHIMABUKURO, Y. E.; SMITH, J. A. The least-squares mixing models to generate fraction images derived from remote sensing multispectral data, **IEEE Trans. Geosci. Remote Sens.**, vol. 29, no. 1, pp.16-20, 1991.

SINGH, A. Digital Change Detection techniques using remotely-sensed data. **International Journal of Remote Sensing**, v. 10, n. 6, pp. 989-1003, 1989.

SMOLA, A.J.; BARTLETT P.L.; SCHOLKOPF B.; SCHUURMANS D.. **Advances In Large Margin Classifiers**. Massachusetts Institute of Technology. London, England: Ed. MIT Press, 2000.

STAUFFER, C.; GRIMSON, W. E. L., “Learning patterns of activity using real-time tracking”. **IEEE Trans. Pattern Anal. Mach. Intell.**, vol. 22, no.8, pp. 747–757, Aug. 2000.

STEINWART,I.; THEILER,J.; LLAMOCCA,D., Using support vector machines for anomalous change detection, **IEEE Geoscience and Remote Sensing Society and the IGARSS 2010**, 3732.

THÉAU J. **Change Detection. Encyclopedia of Geographical Sciences.** Ed: Shashi Shekhar and Hui Xiong, Springer, New-York, 77-84, 2008.

VAPNIK, V. N.; **The Nature Of Statistical Learning Theory.** USA: Springer, 2nd Ed., 1999.

WEBB, A. R.. **Statistical Pattern Recognition.** Second Edition. John Wiley & Sons Ltd: 2002.

XIAOGANG WANG; MENG WANG; WEI LI.. Scene-Specific Pedestrian Detection for Static Video Surveillance. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Vol 36, Issue: 2, pp: 361 – 374, 2014.

YANG, Chih-Cheng.; LEE, Wan-Jui.; LEE, Shie-Jue. Learning of *Kernel* Functions in Support Vector Machines. **International Joint Conference on Neural Networks**, 2006.

ZANOTTA, D. **Uma Abordagem Fuzzy na Detecção Automática de Mudanças do uso do Solo Usando Imagens de Fração e Informações de Contexto Espacial.** Dissertação de Mestrado, Programa de Pós-Graduação em Sensoriamento Remoto – UFRGS. Porto Alegre, 2010.

ZANOTTA, D.; HAERTEL, V. Gradual Land Cover Change Detection based on Multitemporal Fraction Images. **Pattern Recognition**, Vol. 45, p. 2927-2937, 2012.

ZITOVÁ, B.; FLUSSER, J. Image registration methods: a survey. **Image and Vision Computing**, 21, 2003, pp. 977-1000.