

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

RUTE HENRIQUE DA SILVA FERREIRA

**UMA METODOLOGIA PARA A DETECÇÃO DE MUDANÇAS EM IMAGENS  
MULTITEMPORAIS DE SENSORIAMENTO REMOTO EMPREGANDO SUPPORT  
VECTOR MACHINES**

PORTO ALEGRE

2014

RUTE HENRIQUE DA SILVA FERREIRA

**UMA METODOLOGIA PARA A DETECÇÃO DE MUDANÇAS EM IMAGENS  
MULTITEMPORAIS DE SENSORIAMENTO REMOTO EMPREGANDO SUPPORT  
VECTOR MACHINES**

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

Área de concentração: Sensoriamento Remoto e Geoprocessamento.

Orientador: Prof. Dr. Vitor Haertel

PORTO ALEGRE

2014

### CIP - Catalogação na Publicação

Ferreira, Rute Henrique da Silva  
DETECÇÃO DE MUDANÇAS EM IMAGENS MULTITEMPORAIS DE  
SENSORIAMENTO REMOTO EMPREGANDO SUPPORT VECTOR  
MACHINES / Rute Henrique da Silva Ferreira. -- 2014.  
89f.

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. Processamento de  
Imagens. 3. Detecção de Mudanças. 4. Métodos Baseados  
em Kernel. I. Haertel, Vitor, orient. II. Título.

**UMA METODOLOGIA PARA A DETECÇÃO DE MUDANÇAS EM IMAGENS  
MULTITEMPORAIS DE SENSORIAMENTO REMOTO EMPREGANDO SUPPORT  
VECTOR MACHINES**

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

---

Prof. Dr. Daniel Capella Zanotta, IFRS – Rio Grande

---

Prof. Dr. Claudio Rosito Jung, UFRGS

---

Dra. Dejanira Luderitz Saldanha, UFRGS

---

Prof. Dra. Denise Cybis Fontana, UFRGS (Presidente da Banca)

*Dedico esta tese ao meu marido, Leandro, aos meus filhos, Davi e Lucas, e aos meus pais, Luiz Carlos e Gercina, pelo amor, paciência e constante incentivo.*

## **AGRADECIMENTOS**

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

Primeiramente a Deus, pela minha vida e por ter me concedido infinitamente mais do que tudo aquilo que eu pudesse imaginar.

Ao meu orientador, professor Vitor Haertel, in memoriam, pela competência e sabedoria com que me conduziu na realização deste trabalho, dedicando-se à ciência até seus últimos dias. Certamente influenciou positivamente aqueles que tiveram o privilégio de conviver com ele.

Ao professor Roberto Ribeiro Baldino, meu orientador no mestrado, que me “apresentou” ao processamento de imagens.

Ao meu amigo e colega, Neide Angelo, pelas críticas e contribuições para o trabalho e por ter sempre uma palavra de incentivo quando preciso.

Aos professores e funcionários do Programa de Pós-Graduação em Sensoriamento Remoto e do Centro Estadual de Pesquisa em Sensoriamento Remoto e Meteorologia da Universidade Federal do Rio Grande do Sul pela constante disponibilidade em me auxiliar, especialmente a secretária Magdalena Assaf e as professoras Dejanira Saldanha e Denise Fontana.

Aos meus colegas do UNILASALLE, especialmente o Carlos Flores e o Paulo Vargas.

Aos meus pais, Luiz Carlos e Gercina, pelo carinho, apoio e exemplo de vida.

Ao meu esposo Leandro e aos meus filhos, Davi e Lucas, pelo amor incondicional e porque além de compreenderem minhas ausências, me deram alegria e motivos para seguir em frente.

À minha grande família: meus irmãos Ester e Daniel, minha cunhada Jaqueline, meus sobrinhos Misael e Ezequiel e à minha sogra Helena, pelo companheirismo e pelo cuidado com as crianças em todos os momentos em que precisei.

*Se enxerguei mais longe, foi porque me apoiei sobre ombros de gigantes.*

*Isaac Newton*

## RESUMO

Esta tese investiga uma abordagem supervisionada para o problema da detecção de mudanças em imagens multitemporais de sensoriamento remoto empregando *Support Vector Machines* (SVM) com o uso dos *kernels* polinomial e gaussiano (RBF). A proposta metodológica está baseada na diferença das imagens-fração produzidas para cada data. Em imagens de cenas naturais a diferença nas frações de solo e vegetação tendem a apresentar uma distribuição simétrica em torno da origem. Esse fato pode ser usado para modelar duas distribuições normais multivariadas: mudança e não-mudança. O algoritmo *Expectation-Maximization* (EM) é implementado para estimar os parâmetros (vetor de médias, matriz de covariância e probabilidade *a priori*) associados a essas duas distribuições. Amostras aleatórias são extraídas dessas distribuições e usadas para treinar o classificador SVM nesta abordagem supervisionada. A metodologia proposta realiza testes com o uso de conjuntos de dados multitemporais de imagens multiespectrais *TM-Landsat*, que cobrem a mesma cena em duas datas diferentes. Os resultados são comparados com outros procedimentos, incluindo trabalhos anteriores, um conjunto de dados sintéticos e o classificador SVM *One-Class*.

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



## ABSTRACT

In this thesis, we investigate a supervised approach to change detection in remote sensing multi-temporal image data by applying Support Vector Machines (SVM) technique using polynomial kernel and Gaussian kernel (RBF). The methodology is based on the difference-fraction images produced for two dates. In natural scenes, the difference in the fractions such as vegetation and bare soil occurring in two different dates tend to present a distribution symmetric around the origin of the coordinate system. This fact can be used to model two normal multivariate distributions: class change and no-change. The Expectation-Maximization algorithm (EM) is implemented to estimate the parameters (mean vector, covariance matrix and *a priori* probability) associated with these two distributions. Random samples are drawn from these distributions and used to train the SVM classifier in this supervised approach. The proposed methodology performs tests using multi-temporal TM-Landsat multispectral image data covering the same scene in two different dates. The results are compared to other procedures including previous work, a synthetic data set and SVM One-Class.

Keywords: change detection, kernel methods, fraction-images, EM algorithm.

## LISTA DE ILUSTRAÇÕES

Figura 1	Fenômeno de Hughes	18
Figura 2	Superfície de decisão linear separando duas classes	25
Figura 3	Superfície de decisão linear que não passa pela origem	26
Figura 4	Classificação de $\mathbf{a}$ utilizando a superfície de decisão linear que não passa pela origem	26
Figura 5	Hiperplano ótimo separando os dados com a máxima margem	28
Figura 6	Margem $m^*$ da superfície de decisão ótima	29
Figura 7	Cálculo da margem $m^*$ da superfície de decisão ótima	30
Figura 8	Hiperplano separando os dados com a máxima margem no SVM de margem suave	34
Figura 9	Mapeamento de um conjunto de dados não-linear	37
Figura 10	O hiperplano $\mathbf{w} \cdot \mathbf{g}(\mathbf{x}_i) + b = 0$ (com o normal vetor $\mathbf{w}$ e deslocamento $b$ ) separa todos os pontos a partir da origem. Devido aos <i>outliers</i> é associada a uma variável folga $\xi$ . A distância do <i>outlier</i> ao hiperplano é de $\xi/  \mathbf{w}  $ e a distância entre o hiperplano à origem é $b/  \mathbf{w}  $ . Esta última implica que um pequeno $  \mathbf{w}  $ corresponde a uma larga margem de separação a partir da origem	41
Figura 11	Para kernels RBF, que dependem apenas $x_i - x_j$ , $k(x_i, x_j)$ é constante desde que $x_i = x_j$ e os pontos de dados mapeados caem sobre uma hipersfera no espaço característico	43
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	47
Figura 13	Esquema geral de análise dos dados: as imagens dos dois períodos fracionadas são comparadas por subtração pixel a pixel gerando as imagens de diferença para cada fração; os parâmetros estatísticos que caracterizam as funções densidade de probabilidade são estimados e usados tanto numa classificação por SVM com amostras randômicas extraídas a	

	partir de uma distribuição normal multivariada quanto numa classificação com SVM <i>One-Class</i>	48
Figura 14	Gráfico do espalhamento dos dados da imagem-diferença.	49
Figura 15	Distribuições das classes mudança e não-mudança	50
Figura 16	(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)	57
Figura 17	Imagens-fração da imagem de 1991 para (a) vegetação, (b) solo e (c) sombra. Variação contínua de preto (0%) a branco (100%)	58
Figura 18	Mudanças sintéticas produzidas da imagem. Em cada par, a figura à esquerda mostra a região copiada e à direita o local onde foi colada	59
Figura 19	Imagem de diferença com dados reais. Composição em falsa cor 3 (R) 2(G) 3(B)	60
Figura 20	Histogramas da imagem de diferenças. (a) Diferenças entre as frações de vegetação. (b) Diferenças entre as frações de solo	61
Figura 21	Espalhamento dos dados da imagem-diferença no experimento com dados reais	62
Figura 22	Comparação da acurácia do usuário no SVM Polinomial nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 200 amostras de teste	65
Figura 23	Comparação da acurácia do usuário no SVM Polinomial nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 300 amostras de teste	66
Figura 24	Comparação da acurácia do usuário no SVM Polinomial nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 400 amostras de teste	66
Figura 25	Comparação da acurácia do usuário no SVM RBF nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 200 amostras de teste	67
Figura 26	Comparação da acurácia do usuário no SVM RBF nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 300 amostras de teste	68
Figura 27	Comparação da acurácia do usuário no SVM RBF nos testes	

	com a imagem sintética com 200 e 300 amostras de treinamento e 400 amostras de teste	68
Figura 28	Mapa de mudança para o SVM polinomial, com 200 amostras de treinamento nos testes com a imagem real	70
Figura 29	Mapa de mudança para o SVM polinomial, com 300 amostras de treinamento nos testes com a imagem real	71
Figura 30	Mapa de mudança para o SVM RBF, com 200 amostras de treinamento nos testes com a imagem real	72
Figura 31	Mapa de mudança para o SVM RBF, com 300 amostras de treinamento nos testes com a imagem real	73
Figura 32	Acurácia do usuário no SVM <i>One-Class</i> nos testes com a imagem sintética com 200 amostras de teste	76
Figura 33	Acurácia do usuário no SVM <i>One-Class</i> nos testes com a imagem sintética com 300 amostras de teste	76
Figura 34	Acurácia do usuário no SVM <i>One-Class</i> nos testes com a imagem sintética com 400 amostras de teste	76
Figura 35	Mapa de mudança para o SVM <i>One-Class</i> , com 200 amostras de treinamento nos testes com a imagem real	77
Figura 36	Mapa de mudança para o SVM <i>One-Class</i> , com 200 amostras de treinamento nos testes com a imagem real	78
Figura 37	Comparação SVM Polinomial, RBF e SVM <i>One-Class</i> nos testes com a imagem sintética com $C = 10$	80
Figura 38	Comparação dos mapas de mudança produzidos pelo SVM Polinomial, RBF e SVM <i>One-Class</i> nos testes com a imagem real	81

## LISTA DE TABELAS

Tabela 1	Cálculo do produto interno entre $\mathbf{w}$ e $\mathbf{z}$	27
Tabela 2	Cálculo da margem	30
Tabela 3	Transformação do problema de maximização em minimização	31
Tabela 4	Cálculo de $\mathbf{w}^* \cdot \mathbf{g}(\mathbf{x}) = b^*$	38
Tabela 5	Cálculo do produto $\mathbf{g}(\mathbf{x}) \cdot \mathbf{g}(\mathbf{y})$	38
Tabela 6	Alguns <i>kernels</i> conhecidos e seus parâmetros livres	39
Tabela 7	Matriz de confusão	54
Tabela 8	Acurácia do SVM polinomial utilizando a imagem sintética e 200 amostras de teste	64
Tabela 9	Acurácia do SVM polinomial utilizando a imagem sintética e 300 amostras de teste	65
Tabela 10	Acurácia do SVM polinomial utilizando a imagem sintética e 400 amostras de teste	65
Tabela 11	Acurácia do SVM RBF utilizando a imagem sintética e 200 amostras de teste	66
Tabela 12	Acurácia do SVM RBF utilizando a imagem sintética e 300 amostras de teste	67
Tabela 13	Acurácia do SVM RBF utilizando a imagem sintética e 400 amostras de teste	67
Tabela 14	Acurácia do SVM <i>One-Class</i> utilizando a imagem sintética e 200 amostras de teste	75
Tabela 15	Acurácia do SVM <i>One-Class</i> utilizando a imagem sintética e 300 amostras de teste.	75
Tabela 16	Acurácia do SVM <i>One-Class</i> utilizando a imagem sintética e 400 amostras de teste	75
Tabela 17	Comparação SVM Polinomial, RBF e SVM <i>One-Class</i> nos testes com a imagem sintética com o conjunto de 300 amostras de teste para cada classe	80

## LISTA DE ABREVIATURAS E SIGLAS

MVG	Método da Máxima Verossimilhança Gaussiana
SVM	<i>Support Vector Machines</i>
SVDD	<i>Support Vector Data Domain</i>
CVA	<i>Change Vector Analysis</i>
EM	<i>Expectation-Maximization</i>
KKT	<i>Karush–Kuhn–Tucker</i>
RBF	<i>Radial Basis Function</i>
MLME	Modelo Linear de Mistura Espectral
AU	Acurácia do Usuário para a Classe Mudança
FA	Taxa de Falsos Alarmes

## 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 $\mathbf{x}$
$\theta$	ângulo entre dois vetores
$G(\mathbf{x})$	superfície de separação
$f(\mathbf{x})$	função de decisão
$r$	margem dos dois hiperplanos de suporte
$\alpha_i, \beta_i$	multiplicadores de Lagrange
$M$	número de amostras de treinamento
$\xi_i$	variável de folga
$g(\mathbf{x})$	função de mapeamento
$m$	dimensão do espaço de entrada
$l$	dimensão do espaço característico
$S$	conjunto dos <i>support vectors</i>
$k(x,y)$	função de <i>kernel</i>
$d$	parâmetro do <i>kernel</i> polinomial
$\gamma$	parâmetro do <i>kernel</i> RBF
$C$	constante de custo
$b$	parâmetro de ajuste
$R$	raio da esfera
$c$	centro da esfera
$M_1$ 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 a priori das classes mudança e não-mudança
$\lambda$	autovalor

## SUMÁRIO

1	INTRODUÇÃO.....	17
2	REVISÃO BIBLIOGRÁFICA.....	21
2.1	Support Vector Machines e Métodos Baseados em Kernels.....	24
2.1.1	<i>SVM não Linear</i> .....	35
2.1.2	<i>SVM One-Class</i> .....	39
2.1.2.1	<i>O Hiperplano de Separação à Origem</i> .....	40
2.1.2.2	<i>A Hiperesfera de Volume Mínimo</i> .....	42
2.2	Contribuição deste Trabalho.....	45
3	METODOLOGIA.....	46
3.1	Pré-Processamento.....	46
3.2	Análise dos Dados.....	48
3.2.1	<i>Produção das Imagens de Diferença</i> .....	49
3.2.2	<i>Análise das Imagens de Diferença</i> .....	49
3.2.2.1	<i>O Algoritmo EM</i> .....	50
3.2.2.2	<i>Classificação com SVM e SVM One-Class</i> .....	52
3.2.3	<i>Medidas de Avaliação da Acurácia</i> .....	53
4	TESTES E EXPERIMENTOS.....	56
4.1	Introdução.....	56
4.2	Descrição das Imagens.....	56
4.3	Produção das Imagens de Fração.....	58
4.4	Produção da Imagem Sintética.....	59
4.5	Produção das Imagens de Diferença.....	60
4.6	Estimação dos parâmetros das distribuições das classes mudança e não-mudança.....	61
4.7	Testes utilizando SVM com <i>Kernels</i> Polinomial e RBF.....	63
4.7.1	<i>Testes com a Imagem Sintética</i> .....	63
4.7.2	<i>Testes com a Imagem Real</i> .....	69
4.8	Testes utilizando SVM <i>One-Class</i> .....	74
4.8.1	<i>Testes com a Imagem Sintética</i> .....	74
4.8.2	<i>Testes com a Imagem Real</i> .....	77
4.9	Comparação de SVM com SVM <i>One-Class</i> .....	79



<b>5</b>	<b>CONCLUSÕES.....</b>	<b>83</b>
<b>5.1</b>	<b>Sugestões.....</b>	<b>84</b>
	<b>REFERÊNCIAS.....</b>	<b>86</b>

## 1 INTRODUÇÃO

Na literatura disponível na área geral de reconhecimento de padrões, constam classificadores com características distintas. De um modo geral, os classificadores propostos podem ser agrupados em duas categorias: paramétricos e não-paramétricos. Em alguns casos, sabe-se que as classes presentes na cena podem ser modeladas por um conjunto de parâmetros. Em situações como essas é possível aplicar uma abordagem paramétrica, que consiste em definir funções discriminantes com base em funções de densidade de probabilidade, as quais podem ser caracterizadas por um número relativamente pequeno de parâmetros. Ocorrem outros processos de classificação para os quais não é feita nenhuma suposição com relação à caracterização das classes por parâmetros. Classificadores não-paramétricos são empregados nessas condições. Deve-se observar aqui que funções discriminantes do tipo parametrizado (como, por exemplo, os coeficientes em um polinômio), são, por vezes, empregadas em classificadores não-paramétricos. Nesses casos, entretanto, não é empregada nenhuma forma de distribuição (Bow, 1992).

Dentro de uma abordagem paramétrica, o método mais utilizado em sensoriamento remoto é o Método da Máxima Verossimilhança Gaussiana (MVG), que é um caso particular do classificador de Bayes. Já redes neurais e *Support Vector Machines* (SVM) são exemplos de classificadores não-paramétricos. Para o treinamento desses classificadores usam-se amostras representativas de cada classe envolvida na cena.

O processo de classificação de dados em alta dimensionalidade pode se tornar um problema para classificadores paramétricos, pois na medida em que a dimensionalidade dos dados aumenta (aumento no número de bandas espectrais), aumenta também o número de parâmetros a serem estimados a partir de um número geralmente limitado de amostras de treinamento disponíveis para esse fim. Essa limitação no número de amostras de treinamento causa o fenômeno de *Hughes*, que consiste na gradual degradação dos resultados da classificação, com o aumento da dimensionalidade dos dados (Landgrebe, 1997), conforme mostra a figura 1. Esse fato é especialmente negativo, pois tende a limitar na prática o uso de dados em alta dimensionalidade e de seu alto poder discriminante.

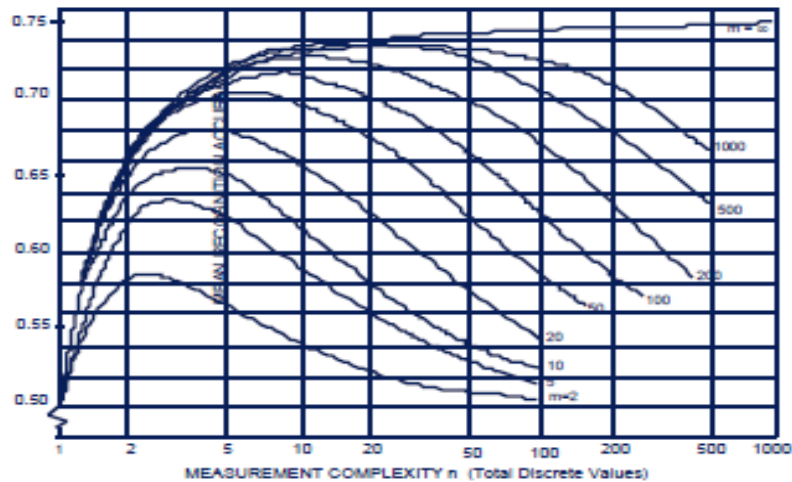


Figura 1 – Fenômeno de Hughes. Fonte: Landgrebe, 1997.

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 semirotnuladas;
- 4) utilização de classificadores não-paramétricos, como é o caso de SVM.

Nos últimos anos, SVM têm sido usados com sucesso em muitos domínios de aplicação. Eles também constituem um método bem adequado para lidar com o problema da classificação de imagens hiperespectrais, uma vez que eles podem lidar com espaços de dimensões grandes de forma eficiente, lidam com amostras ruidosas de uma forma robusta e produzem soluções esparsas, ou seja, o modelo que define a fronteira de decisão é expresso por uma função de um subconjunto de amostras de treinamento (Camps-Valls e Bruzzone, 2005).

Também é possível observar na literatura que nos últimos anos vem crescendo o 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 monitoramento de desastres ambientais, queimadas, desmatamentos, estudos urbanos, monitoramento florestal etc. (Zanotta, 2010).

Em observação da Terra, a definição formal de detecção de mudança envolve o uso de dados multitemporais para discriminar áreas de cobertura da terra que mudaram entre as datas. Portanto, a classificação multitemporal pode ser vista como uma tarefa mais geral, e isso inclui o problema de detecção de mudança como um caso particular. Classificação multitemporal e técnicas de detecção de mudança tornaram-se mais úteis nas últimas décadas, com o aumento de dados multitemporais e multifonte disponíveis a partir de plataformas de sensoriamento remoto (Muñoz-Marí et al., 2009).

Em geral, a detecção de mudanças envolve a análise de duas imagens multiespectrais cobrindo a mesma área, mas adquiridas em datas distintas. Essa análise permite a detecção de alterações ocorridas na cobertura do solo durante o período considerado.

Considerando o crescimento do interesse por estudos envolvendo classificadores não paramétricos e por estudos que envolvam técnicas automáticas para detecção de mudanças, este trabalho aborda o problema da detecção de mudanças com o uso de *Support Vector Machines*.

Entretanto, duas principais abordagens para o problema da detecção de mudanças têm sido propostas na literatura na área de sensoriamento remoto: o método supervisionado e o método não supervisionado (Singh, 1989; Bruzzone e Pietro, 2000; Lu et al., 2003; Bruzzone, Cossu e Vernazza, 2004; Catellana, D'Addabbo e Pasquariello, 2007; Filippone e Sanguinetti, 2010). O objetivo desta tese consiste em investigar uma nova abordagem para o problema: uma técnica supervisionada para o problema de detecção de mudanças com o uso do classificador *SVM*, mas com uma nova abordagem para a extração das amostras de treinamento.

Como objetivos específicos tem-se a implementação da função de decisão *SVM* em um par de imagens da mesma região, mas adquiridas em datas distintas. Dois *kernels* e seus respectivos parâmetros são testados com a finalidade de investigar sua influência na acurácia produzida. Para validação da metodologia proposta, os resultados obtidos pelo classificador *SVM* em questão serão confrontados com os obtidos pelo classificador *SVM One-Class*, conforme proposto na literatura recente (Schölkopf, Burges e Vapnik, 1995; Burges, 1999; Tax e Duin, 1999; Bovolo, Camps-Valls, Bruzzone, 2010; Muñoz-Marí e Bovolo, 2009; Campbell e Bennett, 2001).

O presente estudo 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 de contextualizar o trabalho, mostrar a relevância do tema e a contribuição que a pesquisa propõe. Também são descritos os classificadores SVM e SVM *One-Class* de forma detalhada, visando à compreensão de todas as variáveis e equações envolvidas no processo de classificação. No terceiro capítulo é descrita a metodologia, destacando as etapas que foram seguidas durante a realização do trabalho. No capítulo 4, os testes e experimentos são apresentados de acordo com as etapas descritas no capítulo 3. Por fim, o capítulo 5 apresenta as conclusões, análise final e as sugestões para futuras investigações.

## 2 REVISÃO BIBLIOGRÁFICA

O problema da detecção de mudanças em um conjunto de imagens da mesma região, mas adquiridos em datas distintas, tem sido alvo de interesse em áreas como sensoriamento remoto, diagnósticos médicos, planejamento urbano e vigilância por vídeo. No sensoriamento remoto, técnicas de detecção de mudanças em imagens multitemporais têm sido aplicadas em monitoramento agrícola, florestal, de áreas urbanas, entre outras aplicações (Zanotta e Haertel, 2012).

Duas principais abordagens para o problema da detecção de mudanças têm sido propostas na literatura na área de sensoriamento remoto: o método supervisionado e o método não-supervisionado (Singh, 1989; Bruzzone e Pietro, 2000; Lu et al., 2003; Bruzzone, Cossu e Vernazza, 2004; Catellana, D'Addabbo e Pasquariello, 2007; Filippone e Sanguinetti, 2010).

A primeira baseia-se na análise comparativa de imagens temáticas, cobrindo a mesma cena e obtidas a partir da classificação de imagens multiespectrais obtidas em datas distintas. A desvantagem desse procedimento é que ele requer informação de verdade terrestre (amostras de treinamento).

Já na segunda abordagem, a detecção é feita pela comparação direta entre duas imagens multitemporais sem fazer uso de qualquer outro dado adicional. Nesse caso, o processo se dá pela aplicação de operações matemáticas e métodos estatísticos.

Embora o método supervisionado tenha algumas vantagens em comparação ao método não-supervisionado, como a definição da natureza do tipo de mudança ocorrida, a disponibilização de dados multitemporais de verdade terrestre é geralmente uma tarefa de difícil execução e de custo elevado. Conseqüentemente, o uso de métodos não-supervisionados na detecção de mudanças é mais atraente a nível operacional e tem sido largamente explorado pelos pesquisadores no desenvolvimento de seus trabalhos (Zanotta e Haertel, 2012).

As abordagens não-supervisionadas para a detecção de mudanças são geralmente baseadas em imagens diferença. Essas imagens são produzidas pela subtração, *pixel por pixel*, das imagens adquiridas em duas datas diferentes (Radke et al., 2005). As diferenças podem ser calculadas a partir dos dados originais ou a partir de dados extraídos dos originais, tais como componentes principais ou índices de vegetação. Em qualquer caso, uma imagem de diferenças é produzida e um

limiar é aplicado para distinguir os *pixels* onde a mudança tenha ocorrido daqueles *pixels* que permaneceram inalterados.

Muitas abordagens propostas na literatura buscam modelar as distribuições para as classes mudança e não-mudança, a fim de estimar um valor adequado para o limiar que separa as duas classes (Bruzzone e Pietro, 2000; Bazi, Bruzzone e Melgani, 2005; Bovolo, Camps-Valls e Bruzzone, 2010). O processo de detecção de mudança pode ser realizado utilizando uma única banda espectral, que capta as mudanças de interesse na cena, ou utilizando um conjunto de  $p$  bandas espectrais.

Dentre os vários métodos de detecção de mudança não-supervisionados propostos na literatura, o mais amplamente utilizado é aquele conhecido como *análise do vetor de mudanças* (CVA – *change vector analysis*) (Bovolo, Camps-Valls e Bruzzone, 2010). CVA é normalmente aplicado às imagens multiespectrais adquiridas por sensores passivos, usando todos os canais espectrais que contêm informações úteis relativas ao tipo considerado de mudança.

A técnica de CVA é baseada em três etapas (Bovolo, Camps-Valls, Bruzzone, 2010):

(i) Comparação da imagem por subtração de vetores (Imagem diferença). As diferenças são calculadas para cada banda espectral e a alteração total é representada por um vetor  $\mathbf{x}$  no espaço  $p$ -dimensional. Assim, cada *pixel* na imagem está associado a um vetor multidimensional chamado vetor de mudança espectral, ou seja:

$$\mathbf{x} = \sum_{j=1}^p (\mathbf{y}_{2j} - \mathbf{y}_{1j}), \quad (1)$$

onde  $\mathbf{y}_{1j}$  e  $\mathbf{y}_{2j}$  são os vetores de cada pixel, contendo a resposta espectral para cada banda nas duas datas e  $p$  é o número de bandas espectrais.

(ii) Cálculo da norma dos vetores de mudança espectral (por vezes também a direção do vetor de mudança espectral é calculada). Essa operação resulta na imagem de uma dimensão geralmente referida como imagem de magnitude.

(iii) Limiarização aplicada à imagem de magnitude, a fim de obter o mapa de detecção de mudança. Na segunda etapa, devido ao comportamento estatístico de imagens multiespectrais e propriedades do operador de norma é possível afirmar que os *pixels*, mostrando uma magnitude superior a um valor determinado limiar,

são alterados, enquanto *pixels* mostrando uma magnitude menor do que o limiar mantêm-se inalterados.

Outra abordagem para esse problema consiste em definir graus de pertinência à classe mudança (Wang,1990; Foody,1999; Bardossy e Samaniego, 2006; Tso e Mather, 2009). Uma função de pertinência baseada na função de decisão Máxima Verossimilhança foi proposta em Wang (1990). Um Classificador de Decisão em Árvore, utilizando a abordagem fuzzy é empregado em Colditz, Smitdt e Dech (2008). Outra abordagem consiste na aplicação do Modelo Linear de Mistura Espectral para classificar *pixels* da imagem em termos de proporções dos componentes das classes (DeFries, Hansen e Townshend, 2000).

Haertel, Shimabukuro e Almeida Filho (2004) utilizam o conceito de *pixel* mistura no problema da detecção de mudanças. O fenômeno de mistura em *pixels* ocorre quando um mesmo *pixel* compreende duas ou mais classes distintas, denominadas, neste contexto, de componentes (*endmembers*). Então, é possível derivar um novo conjunto de dados com as chamadas imagens-fração, as quais informam em cada *pixel* a fração ocupada por cada uma das componentes do terreno, como por exemplo, vegetação, solo, sombra/água. Essa abordagem apresenta duas vantagens: permite uma análise em nível de subpixel e elimina a necessidade de normalização radiométrica das imagens adquiridas em datas distintas. Dessa forma, as imagens de diferenças são produzidas a partir da subtração de componentes gerados a partir das imagens multiespectrais.

Zanotta e Haertel (2012) utilizam o conceito de *pixel* mistura para estimar com mais precisão os graus de mudanças, isto é, os graus de pertinência à classe mudança. Neste trabalho foi investigada uma abordagem bayesiana para estimar graus de pertinência para as classes mudança e não-mudança com base nas diferenças entre as imagens-fração estimadas para duas datas diferentes.

O presente trabalho diferencia-se da abordagem de Zanota e Haertel (2012), pois embora utilize o conceito de *pixel* mistura, propõe uma abordagem supervisionada para o problema com o uso do classificador *Support Vector Machines* (SVM).

Observa-se que nos últimos anos, classificadores baseados em *kernels* têm sido usados com sucesso em diversas áreas, bem como o reconhecimento de faces, categorização de texto, previsão de séries temporais e reconhecimento de dígitos manuscritos. Dentre eles, pode-se citar SVM, *Kernel* Análise de Componentes



Principais (KPCA), *Kernel Fisher Discriminante* (KFD), *Kernel k-Médias*, entre outros (Shawe-Taylor e Cristianini, 2004).

SVM constitui um método bem adequado para lidar com o problema da classificação de imagens hiperespectrais, uma vez que eles podem lidar com espaços de dimensões grandes de forma eficiente; lidam com amostras ruidosas de uma forma robusta e produzem soluções esparsas, ou seja, o modelo que define a fronteira de decisão é expresso como a função de um subconjunto de amostras de treinamento (Camps-Valls e Bruzzone, 2005).

Nesta tese o problema da detecção de mudanças é abordado considerando, inicialmente, que duas distribuições gaussianas podem caracterizar as classes mudança e não-mudança. Então, o algoritmo *Expectation-Maximization* (EM) (Duda et al., (2001) é utilizado para estimar os parâmetros estatísticos que caracterizam as classes mudança e não-mudança. Esses parâmetros serão utilizados para selecionar amostras de treinamento na metodologia proposta no capítulo 3, o que caracteriza esta proposta como um método de classificação supervisionado, mas com uma nova abordagem para a extração de amostras de treinamento. Após essa etapa, é feita uma classificação com o uso de SVM, com amostragem normal multivariada. As subseções a seguir abordam os classificadores que foram utilizados: o SVM e o SVM *one-class*, utilizado para fins de comparação.

## 2.1 *Support Vector Machines* e Métodos Baseados em Kernels

Esta seção aborda o conceito de SVM, que será utilizado como classificador no problema de detecção de mudanças. Como o SVM é um classificador linear, antes de defini-lo será feita uma breve abordagem das superfícies de decisão linear.

Primeiramente, considera-se que os dados pertencem ao espaço bidimensional  $R^2$ , munido de um produto interno. Também é preciso assumir dados linearmente separáveis, ou seja, existe uma superfície de decisão linear (neste caso de  $R^2$ , uma reta) separando-os integralmente. Assume-se também, que essa superfície passa pela origem. Assim, a superfície de separação  $G(x)$  é dada por:

$$G(\mathbf{x}) = \mathbf{w} \cdot \mathbf{x} = 0, \tag{2}$$

onde:

$$\mathbf{x}, \mathbf{w} \in R^2$$

$w$  é o vetor normal à superfície de decisão.

$w \cdot x$  é o produto interno entre os vetores  $w$  e  $x$

Amostras que estiverem acima da superfície de decisão serão rotuladas +1 e as que estiverem abaixo, -1.

A figura 2 a seguir ilustra tal fato:

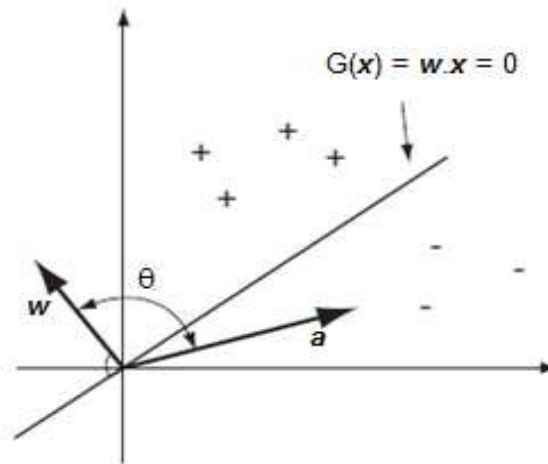


Figura 2 – Superfície de decisão linear separando duas classes. Fonte: Hamel, 2009.

Para rotular uma amostra  $a \in R^2$ , substitui-se  $a$  na expressão de  $G(x)$ :

$$G(a) = w \cdot a = |w| |a| \cos(\theta), \quad (3)$$

onde  $\theta$  é o ângulo entre os vetores  $a$  e  $w$ .

Assim, se o resultado for negativo (ângulo maior que  $90^\circ$ ),  $a$  está abaixo da superfície e se for positivo (ângulo menor que  $90^\circ$ ), está acima. Caso fosse zero, estaria sobre a superfície.

Neste contexto pode-se construir a função de decisão:

$$f(x) = \begin{cases} +1 & \text{se } G(x) \geq 0 \\ -1 & \text{se } G(x) < 0 \end{cases}, \quad \forall x \in R^2 \quad (4)$$

Quando a superfície de decisão não passa pela origem, existe um termo independente  $b$ , ou seja, o produto interno entre os vetores  $a$  e  $w$  é igual a  $b$ . Então, a expressão assume a forma:

$$G(x) = w \cdot x = b \quad (5)$$

Ainda considera-se um problema de classificação binário, com os rótulos +1 para amostras acima da superfície e -1 para amostras abaixo da superfície, como demonstra a figura 3.

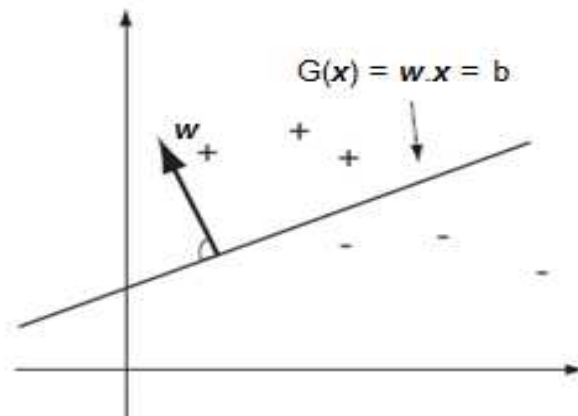


Figura 3 – Superfície de decisão linear que não passa pela origem. Fonte: Hamel, 2009.

Para rotular um vetor  $\mathbf{a} \in \mathbb{R}^2$ , considera-se um vetor  $\mathbf{c} \in \mathbb{R}^2$ , sobre a superfície de decisão. Ou seja:

$$G(\mathbf{c}) = \mathbf{w} \cdot \mathbf{c} = b \quad (6)$$

E toma-se o vetor  $\mathbf{z}$  como a diferença entre  $\mathbf{a}$  e  $\mathbf{c}$ , como ilustra a figura 4 a seguir:

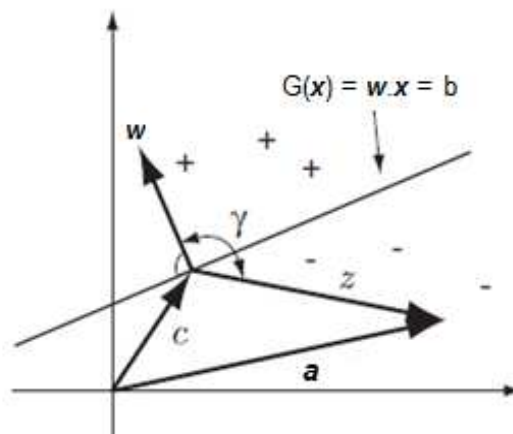


Figura 4 – Classificação de  $\mathbf{a}$  utilizando a superfície de decisão linear que não passa pela origem. Fonte: Hamel, 2009.

Calculando-se o produto interno entre  $\mathbf{w}$  e  $\mathbf{z}$  tem-se:

Tabela 1 – Cálculo do produto interno entre  $\mathbf{w}$  e  $\mathbf{z}$ 

---


$$\begin{aligned}\mathbf{w} \cdot \mathbf{z} &= \mathbf{w} \cdot (\mathbf{a} - \mathbf{c}) \\ &= \mathbf{w} \cdot \mathbf{a} - \mathbf{w} \cdot \mathbf{c} \\ &= \mathbf{w} \cdot \mathbf{a} - b \\ &= G(\mathbf{a}) - b\end{aligned}$$


---

Fonte: Produzido pela autora, 2014.

Dessa forma, se o ângulo entre os vetores  $\mathbf{w}$  e  $\mathbf{z}$  for maior que  $90^\circ$  o vetor  $\mathbf{a}$  estará abaixo da superfície de decisão; se for menor que  $90^\circ$  estará acima.

Assim, a função de decisão fica:

$$f(\mathbf{x}) = \begin{cases} +1 & \text{se } G(\mathbf{x}) - b \geq 0 \\ -1 & \text{se } G(\mathbf{x}) - b < 0 \end{cases} \quad (7)$$

De modo geral,  $k$  é o número real encontrado para  $\mathbf{w} \cdot \mathbf{x} - b$ . Assim pode-se generalizar para o espaço  $R^m$ , munido de produto interno:

$$\begin{aligned}f(\mathbf{x}) &= \text{sgn}(\mathbf{w} \cdot \mathbf{x} - b) \\ \text{sgn}(k) &= \begin{cases} +1 & \text{se } k \geq 0 \\ -1 & \text{se } k < 0 \end{cases} \\ \text{com } \mathbf{x}, \mathbf{w} &\in R^m \text{ e } b \in R \text{ e } k \in R \end{aligned} \quad (8)$$

Para desenvolver a ideia de SVM considera-se um problema que consiste de duas classes linearmente separáveis e um conjunto de treinamento  $D = \{(\mathbf{x}_1, \mathbf{y}_1), (\mathbf{x}_2, \mathbf{y}_2), \dots, (\mathbf{x}_M, \mathbf{y}_M)\} \in R^m \times \{+1, -1\}$  com  $M$  amostras associadas aos rótulos  $+1$  e  $-1$ .

Um hiperplano será chamado *hiperplano de suporte* se for paralelo com uma superfície de decisão linear e todos os pontos de sua respectiva classe estiverem acima ou abaixo dele. Os pontos que definem esse hiperplano são chamados de *support vectors*. Em um problema de classificação binária, a distância entre dois *hiperplanos de suporte* é chamada *margem*. Neste contexto, uma superfície de decisão é considerada ótima se é equidistante de dois hiperplanos de suporte e maximiza a margem, como é possível verificar na figura 5.

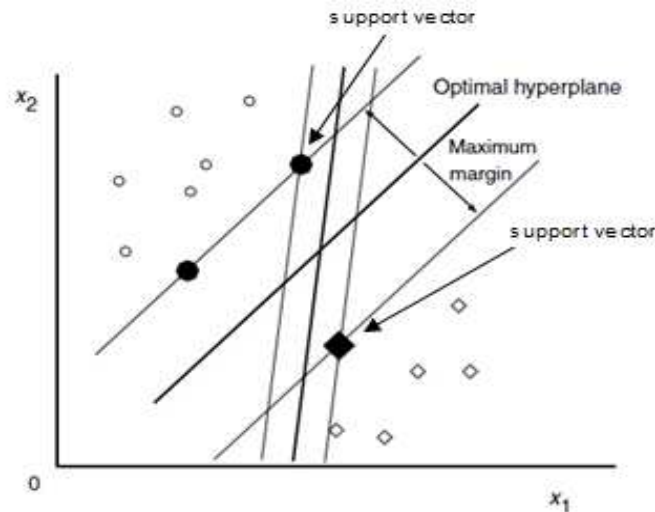


Figura 5 – Hiperplano ótimo separando os dados com a máxima margem. Fonte: Adaptada de Abe (2005).

Encontrar uma superfície de decisão que maximiza a margem entre os dois hiperplanos de suporte é um problema de otimização, onde a função objetivo calcula o tamanho da margem  $r$  para cada superfície de decisão. Então, maximiza-se a função objetivo para encontrar a margem máxima. As restrições neste caso são as posições dos hiperplanos de suporte que não cruzam as fronteiras das respectivas classes. Asumindo-se que se tem a função de decisão ótima, o símbolo  $*$  será utilizado para se referir aos parâmetros relacionados a ela. Matematicamente tem-se:

$$r^* = \max(\phi(\mathbf{w}, b)) \quad (9)$$

Aqui a função  $\phi(\mathbf{w}, b)$  calcula a margem de uma superfície de decisão  $\mathbf{w} \cdot \mathbf{x} = b$ . A margem máxima  $r^*$  é obtida para alguma superfície de decisão, que será chamada:

$$\mathbf{w}^* \cdot \mathbf{x} = b^* \quad (10)$$

Considerando a superfície de decisão é ótima, tem-se:

$$r^* = \phi(\mathbf{w}^*, b^*) = \max(\phi(\mathbf{w}, b)) \quad (11)$$

A margem máxima é calculada na função objetivo substituindo os valores de  $\mathbf{w}^*$  e  $b^*$  na superfície de decisão ótima. Uma vez que a superfície de decisão na equação (10) produz margem máxima, têm-se dois hiperplanos de suporte equidistantes desta superfície:

$$\mathbf{w}^* \cdot \mathbf{x}^* = b^* + k \quad (12)$$

$$\mathbf{w}^* \cdot \mathbf{x}^* = b^* - k, \quad (13)$$

onde  $k \in \mathbb{R}$ .

O primeiro é o hiperplano de suporte para a classe 1 e está acima da superfície de decisão; o segundo é o hiperplano de suporte para a classe -1 e está abaixo da superfície de decisão, conforme mostra a figura 6.

Ao considerar-se ponto  $(x_p, +1) \in D$  um *support vector* para a classe 1 e o ponto  $(x_q, -1) \in D$  um *support vector* para a classe -1 tem-se:

$$\mathbf{w}^* \cdot \mathbf{x}_p = b^* + k \quad (14)$$

$$\mathbf{w}^* \cdot \mathbf{x}_q = b^* - k \quad (15)$$

Ou seja, o *support vector* situa-se em seus respectivos hiperplanos de suporte, como é possível verificar na figura 7:

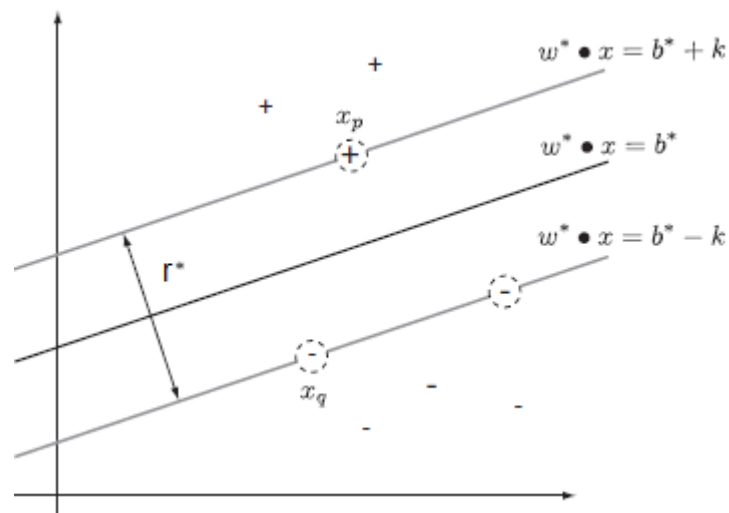


Figura 6 – Margem  $r^*$  da superfície de decisão ótima. Fonte: Hamel, 2009.

A distância entre os dois planos de suporte é a margem  $r^*$  e pode ser calculada como a projeção do vetor  $\mathbf{x}_p - \mathbf{x}_q$  na direção de  $\mathbf{w}^*$ . Isto é, pode-se calcular a margem como a projeção da diferença entre os dois vetores de suporte na direção do vetor normal da superfície de decisão, como mostra a figura 7:

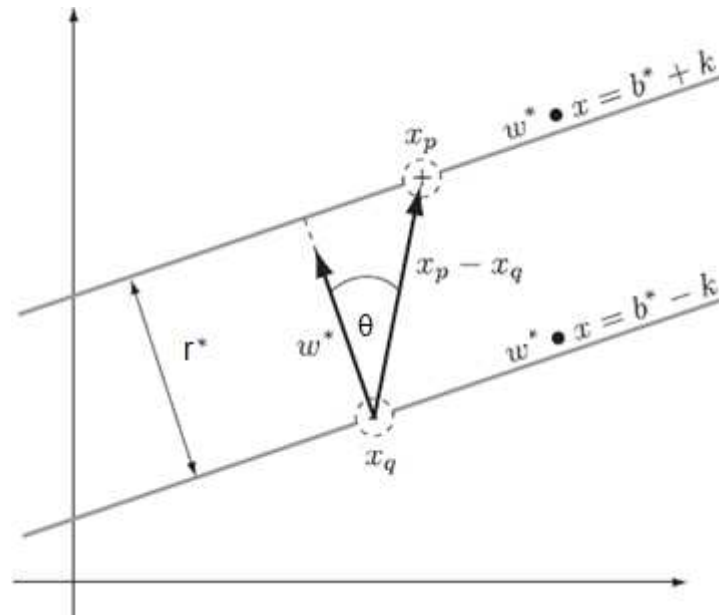


Figura 7 – Cálculo da margem  $r^*$  da superfície de decisão ótima. Fonte: Hamel, 2009.

A tabela 2 mostra como esse cálculo é feito algebricamente.

Tabela 2 – Cálculo da margem

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

Fonte: Produzido pela autora, 2014.

Aqui  $\theta$  é o ângulo entre os vetores  $\mathbf{w}^*$  e  $\mathbf{x}_p - \mathbf{x}_q$ . Logo, a otimização é expressa por:

$$r^* = \max \frac{2k}{|\mathbf{w}|} \quad (16)$$

A tabela 3 mostra que esse problema também pode ser expresso em forma de minimização como:

Tabela 3 – Transformação do problema de maximização em minimização

$$\begin{aligned}
 r^* &= \min \frac{|w|}{2k} \\
 &= \min \frac{|w|^2}{2k} \\
 &= \min \frac{1}{2k} w \cdot w \\
 &= \min \frac{1}{2} w \cdot w
 \end{aligned}$$

Fonte: Produzido pela autora, 2014.

Isso é possível porque o problema da maximização pode ser visto como um problema de minimização da recíproca da função objetivo original. O segundo passo é justificado, já que a função quadrática é uma função monótona para  $x \geq 0$ . O último passo é justificado, pois a otimização é invariante para uma constante. Isso significa que se pode escolher um valor conveniente para  $k$ , aqui  $k = 1$ .

Observa-se que na função objetivo o termo  $b$  não aparece. Entretanto, ele aparecerá nas restrições. Se for tomado  $k = 1$  as equações  $w^* \cdot x_p = b^* + k$  e  $w^* \cdot x_q = b^* - k$  podem ser escritas como  $w^* \cdot x_p = b^* + 1$  e  $w^* \cdot x_q = b^* - 1$ .

Para as restrições do problema de otimização é importante lembrar que os hiperplanos de suporte não podem cruzar as fronteiras das respectivas classes. Logo, tem-se as seguintes desigualdades:

$$w \cdot x \geq b^* + 1 \quad (17)$$

$$w \cdot x \leq b^* - 1 \quad (18)$$

Que podem ser escritas de forma mais elegante como:

$$y_i(w \cdot x_i + b) - 1 \geq 0 \quad (19)$$

para  $y_i \in \{-1, 1\}$  e  $i = 1, 2, \dots, M$

De forma mais resumida, pode-se formular matematicamente o problema desenvolvido acima da seguinte forma:

$$\text{Minimizar: } \left( \frac{1}{2} \|w\|^2 \right) = \frac{1}{2} (w \cdot w) = \frac{1}{2} (w^T w) \quad (20)$$

$$\begin{aligned} \text{Sujeito a: } & y_i(w^T x_i + b) - 1 \geq 0 \quad \text{para } i = 1, 2, \dots, M \\ & \text{e } y_i \in \{-1, 1\} \end{aligned} \quad (21)$$



Uma maneira eficiente para resolver problemas de otimização convexa, na forma apresentada aqui é via programação quadrática. A inclusão das restrições na equação 21 e no problema de minimização da equação 20 pode ser resolvida introduzindo o uso da técnica dos multiplicadores de Lagrange ( $\alpha$ ). Nessa técnica existe exatamente um multiplicador de Lagrange  $\alpha_i$  para cada restrição  $r_i(\mathbf{x}) = y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1$  e a variável  $\mathbf{x}$  é chamada de variável primal e  $\alpha$  variável dual.

Essa abordagem permite reescrever o problema anterior no seguinte formato:

$$\text{Minimizar: } \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^M \alpha_i \{y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1\} \quad (22)$$

Onde  $\alpha = (\alpha_1, \dots, \alpha_M)$  é um vetor de dimensão  $M$ , formado pelos multiplicadores de Lagrange, com  $\alpha_i \geq 0$ .

Considerando a função a minimizar  $L(\alpha, \mathbf{w}, b)$  tem-se:

$$L(\alpha, \mathbf{w}, b) = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^M \alpha_i \{y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1\} = \frac{1}{2} \mathbf{w}^T \mathbf{w} - \sum_{i=1}^M \alpha_i y_i \mathbf{w}^T \mathbf{x}_i - \sum_{i=1}^M \alpha_i y_i b + \sum_{i=1}^M \alpha_i \quad (23)$$

As condições de minimização que serão utilizadas nesse processo são chamadas condições de *Karush–Kuhn–Tucker* (KKT):

$$\frac{\partial L(\alpha, \mathbf{w}^*, b^*)}{\partial \mathbf{w}} = 0 \therefore \mathbf{w}^* = \sum_{i=1}^M \alpha_i y_i \mathbf{x}_i \quad (24)$$

$$\frac{\partial L(\alpha, \mathbf{w}^*, b^*)}{\partial b} = 0 \therefore \sum_{i=1}^M \alpha_i y_i = 0$$

$$\alpha_i y_i (\mathbf{w}^T \mathbf{x}_i + b) - 1 = 0$$

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

$$\alpha_i \geq 0$$

A partir desses resultados pode-se reescrever a equação 23 como:

$$L(\alpha, \mathbf{w}, b) = \sum_{i=1}^M \alpha_i - \frac{1}{2} \left( \sum_{i=1}^M \sum_{j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \right) \quad (25)$$

Dessa forma, o problema definido pela equação 20 pode ser reescrito como:

$$\text{Maximizar: } \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \quad (26)$$

$$\text{Sujeito a: } \sum_{i=1}^M \alpha_i y_i = 0 \quad \text{e} \quad \alpha_i \geq 0 \quad \text{para } i = 1, \dots, M \quad (27)$$

Essa última formulação é denominada de forma dual de margem rígida, e o problema original é conhecido como forma primal de margem rígida. A forma dual é bastante atraente por apresentar restrições mais simples e permitir lidar com o problema de otimização em termos de produtos internos entre dados, o que facilita o tratamento de dados não-lineares (ABE, 2005).

Para reduzir o efeito de *outliers*, pode-se introduzir ao problema de otimização dado anteriormente nas equações 20 e 21, uma variável de folga representada por  $\xi_i$ , sendo  $\xi_i \geq 0$ . Essa variável tem a finalidade de relaxar as restrições do problema aproveitando a sua estruturação, ou seja, ao aplicar esse procedimento, é possível suavizar as margens do classificador linear, permitindo que alguns dados permaneçam entre os hiperplanos formado pelos vetores de suporte. Nessa formulação também fica permitida a ocorrência de alguns erros de classificação e o problema de minimização apresentado na equação 20, nesse caso, pode ser reescrito como:

$$\text{Minimizar: } \frac{1}{2}(\mathbf{w}^T \mathbf{w}) + C \sum_{i=1}^M \xi_i \quad (28)$$

$$\text{Sujeito a: } y_i(\mathbf{w}^T \mathbf{x}_i + b) - 1 + \xi_i \geq 0 \quad \text{para } i=1, 2, \dots, M \quad (29)$$

A constante  $C$ , que aparece na equação 28, é conhecida como 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  $\left( \sum_{i=1}^M \xi_i \right)$ , haverá a minimização do erro de classificação.

Essa formulação é conhecida como SVM de margem suave (*soft-margin*) (ABE, 2005). A figura 8 ilustra a utilização das variáveis de folga  $\xi_i$ . Quando  $0 < \xi_i < 1$ , os dados não têm margem máxima, mas ainda estão corretamente classificados. Contudo, se  $\xi_i \geq 1$  os dados são classificados erroneamente pelo hiperplano de

margem máxima. No caso de  $\xi_j = 0$ , o ponto pertence ao hiperplano de suporte e esse ponto em questão é um vetor de suporte (ABE, 2005).

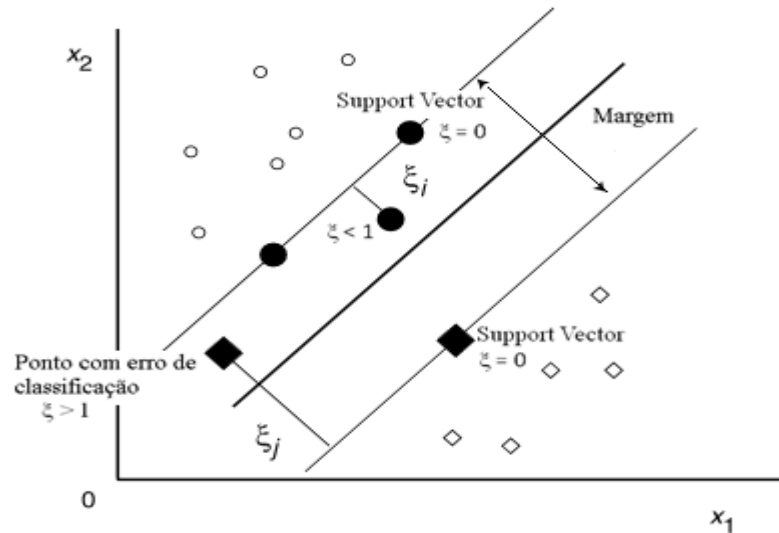


Figura 8 – Hiperplano separando os dados com a máxima margem no SVM de margem suave. Fonte: Adaptado ABE (2005).

A formulação apresentada nas equações 28 e 29 está sujeita à aplicação das condições de KKT, semelhantemente ao que foi feito anteriormente, gerando uma formulação dual apresentada na equação 30 e sujeita às restrições dadas na equação 31:

$$\text{Maximizar: } \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \quad (30)$$

$$\text{Sujeito a: } \sum_{i=1}^M \alpha_i y_i = 0 \quad \text{e} \quad C \geq \alpha_i \geq 0 \quad \text{para } i = 1, \dots, M \quad (31)$$

Em geral, as classes presentes em cenas de sensoriamento remoto, não são linearmente separáveis, o que permite usar a formulação acima. Contudo, isso pode ocasionar a perda da habilidade de generalização existente na técnica. Uma alternativa para essa perda será abordada no subitem a seguir.

Durante a implementação da proposta, as considerações feitas até aqui foram utilizadas para adaptar o problema ao uso da função *quadprog* do Matlab.

### 2.1.1 SVM não Linear

Conforme mencionado anteriormente, poucos conjuntos de dados no mundo real são linearmente separáveis. O que torna o SVM notável é que a estrutura básica linear é facilmente estendida para o caso em que o conjunto de dados não é linearmente separável. A ideia fundamental por trás dessa extensão é transformar o espaço de entrada, onde o conjunto de dados não é linearmente separável em um espaço de dimensão superior chamado espaço das variáveis (*feature space*) ou espaço característico, onde os dados são linearmente separáveis.

Se essas transformações forem escolhidas com cuidado, todos os cálculos associados ao espaço das variáveis podem ser realizados no espaço de entrada. Isto é, mesmo transformando o espaço de entrada para que os dados sejam linearmente separáveis, não é preciso arcar com o custo computacional para essas transformações.

Seja uma função não linear  $g(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \dots, g_l(\mathbf{x}))^T$  que mapeia o espaço de entrada de dimensão  $m$  para o espaço característico de dimensão  $l$ , a função de decisão linear no espaço característico é dada por:

$$G(\mathbf{x}) = \mathbf{w} \cdot g(\mathbf{x}) + b, \quad (32)$$

onde,  $\mathbf{w}$  é dado por:

$$\mathbf{w} = \sum_{i=1}^M \alpha_i y_i g(\mathbf{x}_i) \quad (33)$$

e substituindo-o na equação 32, a expressão para a função de decisão neste novo espaço pode ser escrita:

$$G(\mathbf{x}) = \sum_{i \in S} \alpha_i y_i g(\mathbf{x}_i)^T g(\mathbf{x}) + b, \quad (34)$$

onde  $S$  é o conjunto dos *support vectors*.

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, de dimensão  $M \times M$ .

$$k_{ij} = g(\mathbf{x}_i) \cdot g(\mathbf{x}_j) \text{ para } i, j = 1, \dots, M \quad (35)$$

Nessa expressão, as amostras nunca aparecem isoladamente, mas sempre em pares, em um produto interno. *Kernel* pode, portanto, ser definido como o produto interno das variáveis no espaço característico. A condição necessária para

que uma função  $k$  seja um *kernel* é conhecida como condição de Mercer (ABE, 2005):

$$\sum_{i,j=1}^M h_i h_j k(\mathbf{x}_i, \mathbf{x}_j) \geq 0 \quad (2.36)$$

Para todo  $M$ ,  $\mathbf{x}_i$  e  $h_i$ , onde  $M$  é um número natural (número de amostras disponíveis) e  $h_i$  é um número real, para os quais existe uma função de mapeamento  $g(\mathbf{x})$ , que mapeia  $\mathbf{x}$  no espaço característico.

Assim, têm-se duas possíveis abordagens ao problema do mapeamento de dados em espaços de dimensão mais elevada para fins de classificação empregando funções de decisão lineares  $g(\mathbf{x})$ :

1- Selecionando explicitamente uma função  $g(\mathbf{x})$  para mapeamento dos dados em um espaço de dimensão mais alta.

2- Selecionando diretamente um *kernel* que satisfaça as condições de Mercer. Esse *kernel* definirá de uma forma implícita a função de mapeamento  $g(\mathbf{x})$ .

Do ponto de vista matemático, as duas possíveis abordagens citadas acima são equivalentes, mas a segunda abordagem apresenta a vantagem de ser mais fácil de ser implementada. Outra vantagem oferecida por essa abordagem consiste em não se necessitar operar diretamente no espaço de dimensão mais alta, no qual os dados estão sendo mapeados. Tanto na fase de treinamento do classificador quanto na fase de classificação dos dados utiliza-se diretamente  $k(\mathbf{x}_i, \mathbf{x})$  em lugar da função de mapeamento  $g(\mathbf{x})$ .

O exemplo a seguir, adaptado de Hamel (2009), ilustra a ideia de *kernel*, utilizando um mapeamento de  $R^2$  para  $R^3$ . Aqui se utiliza  $g(\mathbf{x})$  para a função de mapeamento e  $k(\mathbf{x}, \mathbf{y})$  para as funções de *kernel*.

Na figura 9, em (a)  $\mathbf{x} = (x_1, x_2) \in R^2$  e em (b)  $\mathbf{z} = (z_1, z_2, z_3) \in R^3$  usando a transformação  $g(\mathbf{x}): R^2 \rightarrow R^3$  que será definida a seguir:

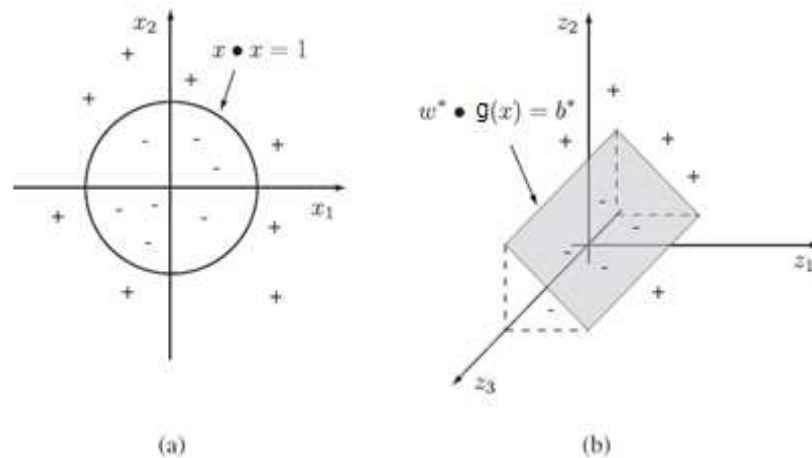


Figura 9 – Mapeamento de um conjunto de dados não-linear. Fonte: Hamel, 2009.

Pode-se observar que na figura 10(a) não há superfície de decisão linear da forma  $\mathbf{w} \cdot \mathbf{x} = b$ . Por outro lado, a superfície de decisão não linear  $\mathbf{x} \cdot \mathbf{x} = 1$  é capaz de separar os dados, conforme mostra a figura acima.

Ao invés da construção de uma função de decisão que baseia-se em uma superfície de decisão no espaço de entrada, considera-se uma função de decisão que primeiro mapeia os pontos que originalmente pertencem a  $\mathbb{R}^2$  para um espaço de dimensão maior, como o  $\mathbb{R}^3$ , para depois utilizar, neste novo espaço, a superfície de decisão linear.

Neste caso tem-se o mapeamento  $g(\mathbf{x}): \mathbb{R}^2 \rightarrow \mathbb{R}^3$  é definido como:

$$g(\mathbf{x}) = g(x_1, x_2) = (x_1^2, x_2^2, \sqrt{2}x_1x_2) = (z_1, z_2, z_3) = \mathbf{z} \quad (37)$$

Nota-se que  $g(\mathbf{x})$  mapeia os pontos de  $\mathbb{R}^2$  para  $\mathbb{R}^3$ . Com esse mapeamento qualquer ponto na superfície de decisão não-linear é mapeado para o plano  $\mathbf{w}^* \cdot g(\mathbf{x}) = b^*$  no espaço característico, com  $\mathbf{w}^* = (w_1^*, w_2^*, w_3^*) = (1, 1, 0)$  e  $b^* = 1$ .

Todo o ponto do espaço de entrada rotulado +1 será mapeado acima do plano no espaço característico. Todo o ponto do espaço de entrada rotulado -1 será mapeado abaixo do plano no espaço característico. De fato, o plano  $\mathbf{w}^* \cdot g(\mathbf{x}) = b^*$  separa linearmente as classes no espaço característico, conforme ilustrado na figura 10(b). Ou seja,  $g(\mathbf{x})$  transforma um problema não-linear em um problema linear.

Considere o ponto  $q = (1,0) \in \mathbb{R}^2$ . Esse ponto está na superfície de decisão não-linear, pois  $q \cdot q = 1$ . Esse ponto também pertence ao plano do espaço característico, pois conforme tabela 4:

Tabela 4 – Cálculo de  $w^* \cdot g(x) = b^*$

$$w^* \cdot g(q) = (1,1,0) \cdot (1^2, 0^2, \sqrt{2} \cdot 1 \cdot 0) = (1,1,0) \cdot (1,0,0) = 1 = b^*$$

Fonte: Produzido pela autora, 2014.

Com o mapeamento definido na equação 33 o valor do produto  $g(x) \cdot g(y)$ , com  $x, y \in \mathbb{R}^2$  pode ser calculado no espaço de entrada  $\mathbb{R}^2$ , ou seja, como descrito na tabela 5:

Tabela 5 – Cálculo do produto  $g(x) \cdot g(y)$

$$\begin{aligned} g(x) \cdot g(y) &= (x_1^2, x_2^2, \sqrt{2}x_1x_2) \cdot (y_1^2, y_2^2, \sqrt{2}y_1y_2) \\ &= x_1^2y_1^2 + x_2^2y_2^2 + 2x_1x_2y_1y_2 \\ &= (x_1y_1 + x_2y_2)(x_1y_1 + x_2y_2) \\ &= (x \cdot y)(x \cdot y) \\ &= (x \cdot y)^2 \end{aligned}$$

Fonte: Produzido pela autora, 2014.

Ou seja, dado um mapeamento adequado  $g: \mathbb{R}^m \rightarrow \mathbb{R}^l$ , com  $l \geq m$ , os *kernels* ou funções de *kernel* calculam um produto interno no espaço característico, mas o valor desse produto é calculado no espaço de entrada. A escrita da função de decisão em termos de funções de *kernel* é conhecida como *kernel trick*, ou seja, usando uma função de *kernel* apropriada pode-se tirar vantagem de mapear no espaço característico sem ter que pagar o preço de realmente computar o mapeamento explícito.

A seguir, a tabela 6 fornece uma lista de algumas funções de *kernel* mais usuais, considerando  $x, y \in \mathbb{R}^m$ :

Tabela 6 – Alguns *kernels* conhecidos e seus parâmetros livres

Nome	Expressão da Função de <i>Kernel</i>	Parâmetros
<i>Kernel</i> Linear	$k(\mathbf{x}, \mathbf{y}) = \mathbf{x} \bullet \mathbf{y}$	Não há
<i>Kernel</i> Polinomial Homogêneo	$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \bullet \mathbf{y})^d$	$d \geq 2$
<i>Kernel</i> Polinomial Não-Homogêneo	$k(\mathbf{x}, \mathbf{y}) = (\mathbf{x} \bullet \mathbf{y} + c)^d$	$d \geq 2, c > 0$
<i>Kernel</i> Gaussiano ou <i>Radial Basis Function</i> (RBF)	$k(\mathbf{x}, \mathbf{y}) = e^{-\gamma \mathbf{x}-\mathbf{y} ^2}$ ou $k(\mathbf{x}, \mathbf{y}) = e^{- \mathbf{x}-\mathbf{y} ^2/2\sigma^2}$	$\gamma = \frac{1}{2\sigma^2}, \gamma > 0$

Fonte: Adaptado de Hamel, 2009.

É importante salientar que embora as funções tenham parâmetros, não é empregada nenhuma forma de distribuição (Bow, 1992).

Fazendo as substituições pelas funções de *kernel*, o problema de maximização proposto nas equações 30 e 31 adquire a seguinte forma:

$$\text{Maximizar: } \sum_{i=1}^M \alpha_i - \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j y_i y_j k(\mathbf{x}_i, \mathbf{x}_j) \quad (38)$$

$$\text{Sujeito a: } \sum_{i=1}^M y_i \alpha_i = 0 \quad \text{e} \quad 0 \leq \alpha_i \leq C \quad \text{para } i=1, \dots, M \quad (39)$$

### 2.1.2 SVM One-Class

Uma abordagem para o problema da detecção de mudanças presente na literatura é a classificação *one-class*. Nessa abordagem a tarefa é modelar a distribuição de dados, ou seja, o objetivo é criar uma função binária, que é de valor positivo nas regiões do espaço de entrada que contém a maioria dos dados e de valor negativo nas demais. Neste trabalho, a razão desta escolha é que em cenas naturais o número de *pixels* que sofreram mudanças é muito menor do que o número daqueles que permaneceram inalterados.

Uma estratégia é encontrar uma hipersfera com um raio mínimo  $R$  e um centro  $c$ , que contém a maior parte dos dados. Assim, amostras de teste que se encontram fora dos limites da hipersfera são classificadas como mudança. Essa técnica foi originalmente sugerida por Schölkopf, Burges e Vapnik (1995) e Burges (1999) e adaptada para resolver o problema de detecção de mudanças por Tax e



Duin (1999). Atualmente, é utilizada em diversos problemas envolvendo detecção de mudanças (Bovolo, Camps-Valls e Bruzzone, 2010; Muñoz-Marí e Bovolo, 2009). O efeito de *outliers* é reduzido utilizando variáveis de folga para incluir dados fora da esfera.

Uma abordagem alternativa em que o problema da hiperesfera é reduzido a um problema que utiliza um hiperplano de separação, foi proposta por Schölkopf et al (1999) e pode ser vista em Campbell (2002) e Campbell e Bennett (2001). Nesse subitem, primeiramente descreve-se o problema do hiperplano, seguindo-se o da hiperesfera para concluir que os dois são similares.

### 2.1.2.1 O Hiperplano de Separação à Origem

No desenvolvimento a seguir, supõe-se um conjunto de dados  $\mathbf{x}_i$  (com  $i = 1, \dots, M$ ), a função de mapeamento  $\mathbf{x}_i \rightarrow g(\mathbf{x}_i)$ , que mapeia o espaço de entrada de dimensão  $m$  para o espaço característico de dimensão  $l$  e um *kernel*  $k(\mathbf{x}_i, \mathbf{x}_j) = g(\mathbf{x}_i) \cdot g(\mathbf{x}_j)$ .

O objetivo é encontrar uma função de decisão  $f(\mathbf{x})$  que assume o valor 1, numa pequena região que contém a maior parte dos dados e  $-1$  em qualquer outro local. A estratégia é mapear os dados para o espaço característico e separá-los da origem com margem máxima. Para um novo ponto  $\mathbf{x}$ , o valor de  $f(\mathbf{x})$  é obtido avaliando de qual lado do hiperplano o ponto está situado. Para separar o conjunto de dados da origem, é preciso resolver o problema quadrático a seguir:

$$\text{Minimizar : } \left[ \frac{1}{2} \|\mathbf{w}\|^2 \right] + C \sum_{i=1}^M \xi_i + b \quad (40)$$

$$\text{Sujeito a: } \begin{aligned} \mathbf{w} \cdot g(\mathbf{x}_i) + b &\geq -\xi_i \\ \xi_i &\geq 0 \end{aligned} \quad (41)$$

Onde:

$\mathbf{w}$  = vetor normal do hiperplano

$\xi_i$  = variável de folga

$C$  = parâmetro de custo

Nesse caso, a função de decisão é dada por:

$$f(\mathbf{x}) = \text{sgn}(\mathbf{w} \cdot g(\mathbf{x}_i) + b) \quad (42)$$

Como pode ser visto na ilustração da figura 10:

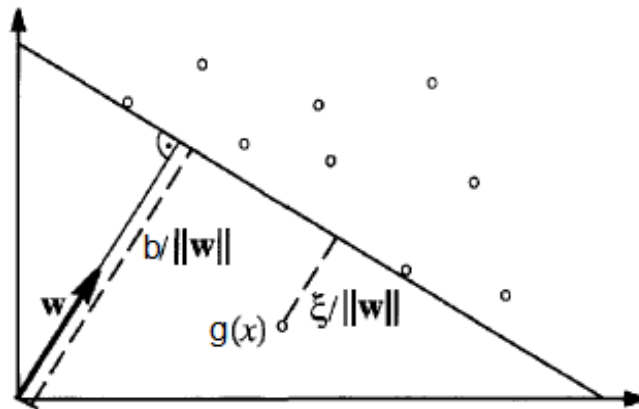


Figura 10 – O hiperplano  $\mathbf{w} \cdot \mathbf{g}(\mathbf{x}_i) + b = 0$  (com o normal vetor  $\mathbf{w}$  e deslocamento  $b$ ) separa todos os pontos a partir da origem. Devido aos *outliers* é associada a uma variável folga  $\xi$ . A distância do *outlier* ao hiperplano é de  $\xi/||\mathbf{w}||$  e a distância entre o hiperplano à origem é  $b/||\mathbf{w}||$ . Esta última implica que um pequeno  $||\mathbf{w}||$  corresponde a uma larga margem de separação a partir da origem. Fonte: Adaptado de Schölkopf e Smola, 2002.

Utilizando multiplicadores de Lagrange  $\alpha$  e  $\beta$  é possível reescrever o problema:

$$L(\mathbf{w}, \xi, \alpha, \beta) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^M \xi_i + b - \sum_{i=1}^M \alpha_i (\mathbf{w} \cdot \mathbf{g}(\mathbf{x}_i) + b + \xi_i) - \sum_{i=1}^M \beta_i \xi_i \quad (43)$$

Considerando as condições de KKT, obtém-se:

$$\mathbf{w} = \sum_{i=1}^M \alpha_i \mathbf{g}(\mathbf{x}_i) \quad (44)$$

$$\alpha_i = C - \beta_i \leq C \quad (45)$$

$$\sum_{i=1}^M \alpha_i = 1 \quad (46)$$

Assim, a função de decisão fica:

$$f(\mathbf{x}) = \text{sgn} \left( \sum_{i=1}^M \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b \right) \quad (47)$$

Após a substituição do *kernel*, a formulação dual dá origem ao problema:

$$\text{Minimizar : } \left[ \frac{1}{2} \sum_{i,j=1}^M \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \right] \quad (48)$$

Sujeito a:

$$\sum_{i=1}^M \alpha_i = 1 \quad (49)$$

$$0 \leq \alpha_i \leq C \quad (50)$$

Na solução ótima, as desigualdades na equação 41 tornam-se igualdades se  $\alpha_i$  e  $\beta_i$  são não-nulos, o que implica  $0 < \alpha_i < C$  (condições de KKT). Para encontrar  $b$  é preciso considerar que para qualquer  $\alpha_i$  o correspondente padrão  $\mathbf{x}_i$  satisfaz:

$$b = \mathbf{w} \cdot \mathbf{g}(\mathbf{x}_i) = \sum_{j=1}^M \alpha_j k(\mathbf{x}_j, \mathbf{x}_i) \quad (51)$$

### 2.1.2.2 A Hiperesfera de Volume Mínimo

Para minimizar o volume da esfera de raio  $R$  e centro  $\mathbf{c}$  a tarefa é:

$$\text{Minimizar : } \left[ R^2 + C \sum_{i=1}^M \xi_i \right] \quad (52)$$

$$\text{Sujeito a:} \quad (53)$$

$$(\mathbf{x}_i - \mathbf{c})^T (\mathbf{x}_i - \mathbf{c}) \leq R^2 + \xi_i$$

$$\xi_i \geq 0$$

Utilizando os multiplicadores de Lagrange, a expressão fica:

$$L(R, \mathbf{c}, \alpha_i, \xi_i) = R^2 + C \sum_{i=1}^M \xi_i - \sum_{i=1}^M \gamma_i \xi_i - \sum_{i=1}^M \alpha_i (R^2 + \xi_i - (\mathbf{x}_i \cdot \mathbf{x}_i - 2\mathbf{c} \cdot \mathbf{x}_i + \mathbf{c} \cdot \mathbf{c})) \quad (54)$$

$$\text{Com:} \quad (55)$$

$$\alpha_i \geq 0 \text{ e } \gamma_i \geq 0$$

Após a substituição do kernel, a formulação dual dá origem a um problema de programação quadrática:

$$\text{Minimizar : } \left[ \sum_{i,j=1}^M \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) - \sum_{i=1}^M \alpha_i k(\mathbf{x}_i, \mathbf{x}_i) \right] \quad (56)$$

$$\text{Sujeito a:} \quad (57)$$

$$\sum_{i=1}^M \alpha_i = 1$$

$$0 \leq \alpha_i \leq C$$

Após o processo de treinamento, uma amostra de teste  $\mathbf{x}$  é classificada como mudança se:

$$k(\mathbf{x}, \mathbf{x}) - 2 \sum_{i=1}^M \alpha_i k(\mathbf{x}, \mathbf{x}_i) + \sum_{i,j=1}^M \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) - R^2 \geq 0 \quad (58)$$

A abordagem proposta por Schölkopf et al. (1999) consiste em restringir o problema ao uso do *kernel* RBF. Nesse caso, os pontos encontram-se na superfície de uma hiperesfera de raio unitário, uma vez que  $g(\mathbf{x}) \cdot g(\mathbf{x}) = k(\mathbf{x}, \mathbf{x}) = e^0 = 1$ . Assim, o problema da hiperesfera é equivalente ao problema do hiperplano, uma vez que o termo contendo  $k(\mathbf{x}_i, \mathbf{x}_j)$  no problema da hiperesfera e o  $\frac{1}{2}$  no problema do hiperplano são constantes. Portanto, encontrar a menor hiperesfera contendo os dados na verdade corresponde a encontrar uma secção dessa hiperesfera. Fazendo a intersecção da hiperesfera contendo os dados com o hiperplano de margem máxima de separação à origem, obtém-se essa secção.

Na figura 11 pode-se verificar esta situação:

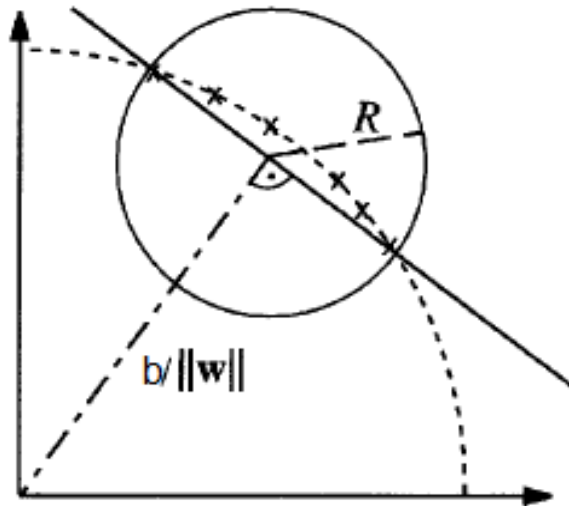


Figura 11 – Para *kernels* RBF, que dependem apenas  $x_i - x_j$ ,  $k(x_i, x_j)$  é constante desde que  $x_i = x_j$  e os pontos de dados mapeados caem sobre uma hiperesfera no espaço característico. Fonte: Adaptado de Schölkopf, 2002.

Assim, o que se procura é construir um hiperplano que maximiza a distância, a partir da origem, com todos os pontos situados do lado oposto da origem de modo que:

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq 0 \quad (59)$$

Essa abordagem conduz a um problema de programação quadrática que Schölkopf et al. (1999) relatam funcionar bem em conjuntos de dados reais e, após a substituição da formulação dual, fica:

$$\text{Minimizar : } \left( \sum_{i,j=1}^M \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) + b \right) \quad (60)$$

$$\text{Sujeito a:} \quad (61)$$

$$\sum_{j=1}^M \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) + b \geq -\xi_i$$

$$\xi_i \geq 0$$

$$\sum_{i=1}^M \alpha_i = 1$$

$$\alpha_i \geq 0$$

Nesse caso,  $b$  e a função de decisão para uma amostra de teste  $\mathbf{x}$  podem ser expressos como:

$$b = \sum_{j=1}^M \alpha_j k(\mathbf{x}_j, \mathbf{x}_i) \quad (62)$$

$$f(\mathbf{x}) = \text{sgn} \left( \sum_{i=1}^M \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b \right) \quad (63)$$

Campbell e Bennett (2001) propõem uma técnica que ao invés de afastar o hiperplano da origem o atrai para o local onde os dados se encontram, mantendo a condição da equação 59. Isso leva a um algoritmo de detecção de mudança baseado em programação linear.

Considerando a função  $f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{g}(\mathbf{x}) + b = 0$  é possível definir um hiperplano no espaço característico. Substituindo  $\mathbf{w}$  pela expressão da equação 44, tem-se:

$$f(\mathbf{x}) = \mathbf{w} \cdot \mathbf{g}(\mathbf{x}) + b = \sum_{i=1}^M \alpha_i k(\mathbf{x}, \mathbf{x}_i) + b = 0 \quad (64)$$

Assim, atrair o hiperplano para o local onde os dados se encontram é possível minimizando  $\sum_{i=1}^M f(\mathbf{x}_i)$ . Isso resulta no problema:

$$\text{Minimizar : } \left[ \sum_{i=1}^M \left( \sum_{j=1}^M \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) + b \right) + C \sum_{i=1}^M \xi_i \right] \quad (65)$$

Sujeito a:

$$\sum_{j=1}^M \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) + b \geq -\xi_i, \forall i \quad (66)$$

$$\begin{aligned}\xi_i &\geq 0 \\ \sum_{i=1}^M \alpha_i &= 1 \\ \alpha_i &\geq 0\end{aligned}$$

Nesse caso, a função de decisão pode ser expressa como:

$$f(\mathbf{x}) = \text{sgn}\left(\sum_{i=1}^M \alpha_i k(\mathbf{x}_i, \mathbf{x}) + b\right) \quad (67)$$

Campbell e Bennett (2001) afirmam que  $b$  é considerado apenas como um parâmetro de ajuste.

## 2.2 Contribuição deste Trabalho

Conforme mencionado no início deste capítulo, o problema da detecção de mudanças em um conjunto de imagens da mesma cena, mas adquiridos em datas distintas tem sido alvo de interesse em diversas áreas, incluindo o sensoriamento remoto e duas principais abordagens têm sido utilizadas para o problema: o método supervisionado e o método não-supervisionado. Também foi destacado que nos últimos anos, classificadores baseados em *kernels* têm sido amplamente utilizados.

Esta pesquisa propõe uma abordagem supervisionada para o problema com o uso do classificador SVM e o SVM *one-class* utilizado para fins de comparação, mas com uma nova abordagem para a extração das amostras de treinamento.

Os capítulos seguintes descreverão a metodologia utilizada e os experimentos realizados a fim de testar o método proposto.

### 3 METODOLOGIA

O processo de detecção de mudanças investigado neste trabalho consiste em duas etapas:

- 1) Pré-processamento – onde as imagens são registradas e as imagens de fração são produzidas.
- 2) Análise dos dados – produção das imagens de diferenças por subtração, estimação dos parâmetros da função densidade de probabilidade, produção das amostras de treinamento, classificação com SVM (empregando os *kernels* polinomial e RBF) e classificação com SVM *One-Class*.

As imagens de diferenças são produzidas a partir da subtração de componentes das imagens-fração geradas a partir das imagens multiespectrais (Haertel *et al.* 2004). A etapa de pré-processamento está baseada no trabalho de Zanotta (2010), entretanto, na classificação com SVM e SVM *One-Class*, uma nova abordagem é proposta.

As subseções a seguir apresentam os conceitos envolvidos em cada etapa do processo.

#### 3.1 Pré-Processamento

Nesta etapa, as imagens são registradas e o Modelo Linear de Mistura Espectral (MLME) é empregado para produzir imagens de fração representando componentes adequadas à cena. Conforme ilustra a figura 12 a seguir:

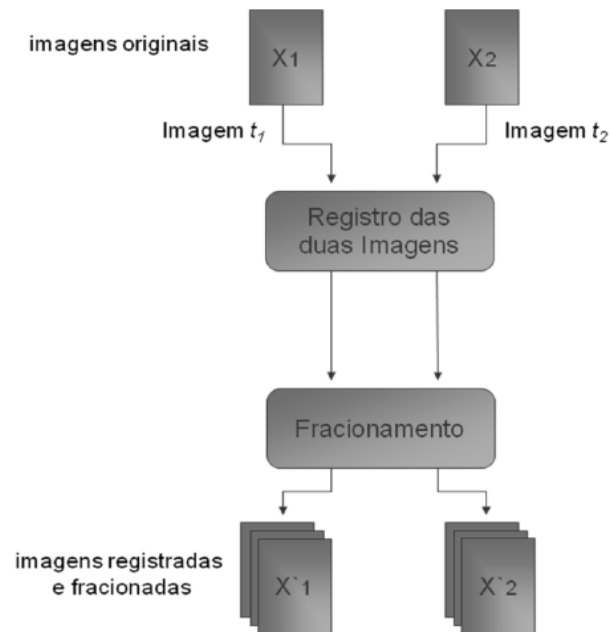


Figura 12 – Esquema geral do pré-processamento: as imagens  $X_1$  e  $X_2$  são registradas, e após, fracionadas nas suas componentes pelo MLME, gerando as imagens  $X'_1$  e  $X'_2$  registradas e fracionadas. Fonte: Zanotta (2010).

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* (Shimabukuro e Smith, 1991).

O MLME é representado por  $p$  equações lineares e  $m$  frações desconhecidas:

$$R = SF + V \quad (68)$$

Onde  $p$  é o número de bandas,  $m$  é o número de componentes,  $R$  é um vetor de dimensão  $p$  que representa a resposta espectral dos *pixels* para cada banda,  $S$  é uma matriz de ordem  $p$  por  $m$ , cujos termos  $s_{ij}$  correspondem à resposta espectral da componente  $j$  na banda  $i$ ,  $F$  é um vetor de dimensão  $m$  que corresponde à fração do *pixel* ocupada pela componente  $j$ , e  $V$  é um vetor de dimensão  $p$  que corresponde ao erro residual do modelo.

Para  $p > m$  o sistema é sobredeterminado, podendo ser resolvido para as frações ( $f_j$ ) utilizando o método dos mínimos quadrados, sujeito às seguintes restrições:



$$0 \leq f_j \leq 1 \quad \forall j \quad \sum_{j=1}^m f_j = 1 \quad (69)$$

Os valores da equação devem ser selecionados de forma a representar, da melhor forma possível, as componentes (*endmembers*) para a imagem em questão. O resultado do fracionamento contará com um número de imagens igual ao número de componentes escolhidas para representar a região.

### 3.2 Análise dos Dados

A análise dos dados pode ser esquematizada pela figura 13. Os detalhes do processo estão descritos em cada subitem a seguir:

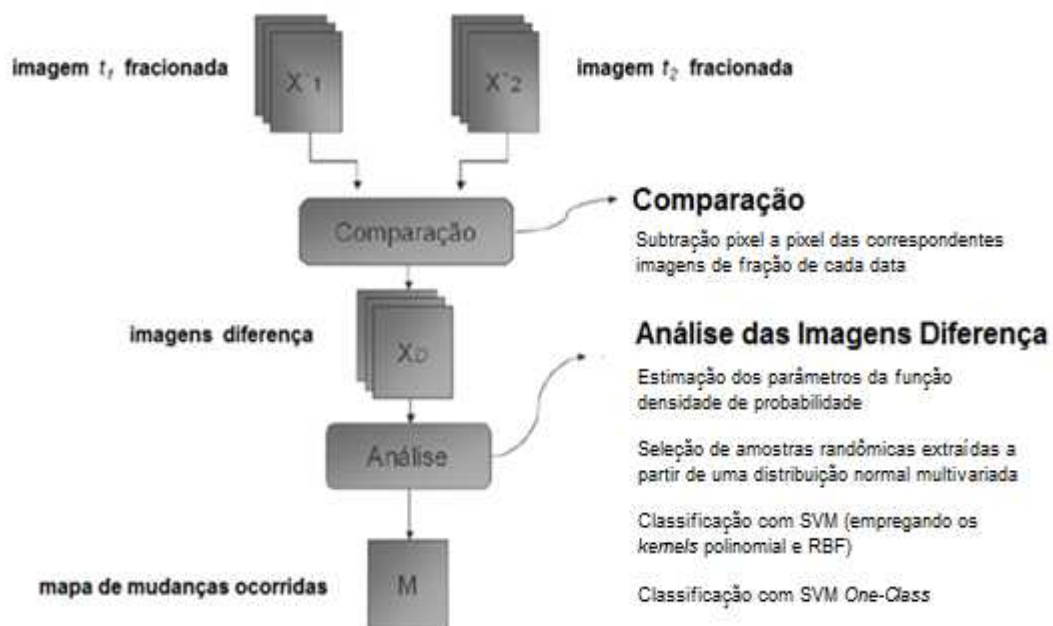


Figura 13 – Esquema geral de análise dos dados: as imagens dos dois períodos fracionados são comparadas por subtração *pixel a pixel*, gerando as imagens de diferença para cada fração; os parâmetros estatísticos que caracterizam as funções densidade de probabilidade são estimados e usados tanto numa classificação por SVM, com amostras randômicas extraídas a partir de uma distribuição normal multivariada, quanto numa classificação com SVM *One-Class*. Fonte: Adaptado de Zanotta (2010).

### 3.2.1 Produção das Imagens de Diferença

As imagens de diferenças são produzidas a partir da subtração de componentes geradas a partir das imagens multiespectrais. Ao empregar imagens-fração na geração de imagens de diferença, a abordagem proposta dispensa a etapa de normalização radiométrica, pois o MLME é aplicado independentemente para cada data.

Em cenas naturais é frequente o uso dos componentes: vegetação, solo e sombra/água. Baseado na proposta de Zanotta e Haertel (2012), aqui serão utilizados os componentes vegetação e solo. Assim, para cada data serão geradas as imagens de fração de vegetação e solo. Dessa forma, as variáveis aqui consideradas são bidimensionais: diferenças na fração vegetação e na fração solo.

### 3.2.2 Análise das Imagens de Diferença

A figura 14 demonstra o espalhamento dos dados da imagem diferença, considerando no eixo vertical as diferenças na fração de solo e no horizontal as diferenças na fração de vegetação.

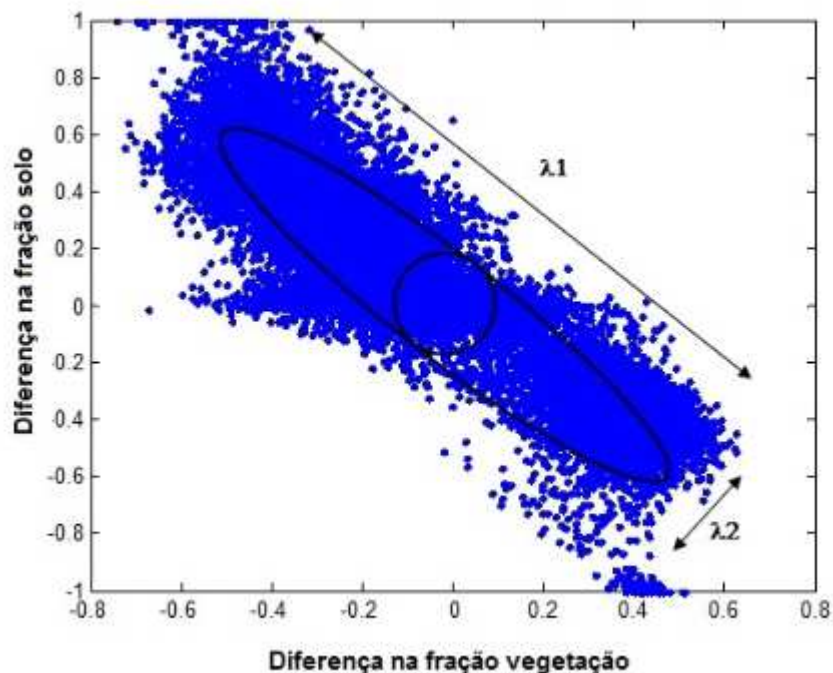


Figura 14 – Gráfico do espalhamento dos dados da imagem-diferença. Fonte: Adaptado de Zanotta, 2010.

Analisando os dados, pode-se constatar a existência de uma correlação negativa entre a distribuição das diferenças das frações de vegetação e solo. Essa correlação é explicada naturalmente pelo consequente aumento na fração de solo resultante da diminuição da fração vegetação e vice-versa.

Ainda é possível verificar que a figura 14 representa a mistura de duas distribuições: mudança e não-mudança. Supõe-se aqui que ambas apresentam uma distribuição normal, bivariada (mudança na fração vegetação e mudança na fração solo). Supõe-se ainda que ambas as distribuições tenham médias na origem (ou próxima a esta) devido à simetria na distribuição das duas variáveis.

A diferença entre as duas distribuições vai se concentrar nas variâncias, com a classe mudança apresentando uma dispersão grande e a classe não-mudança uma dispersão pequena, conforme mostra a figura 15:

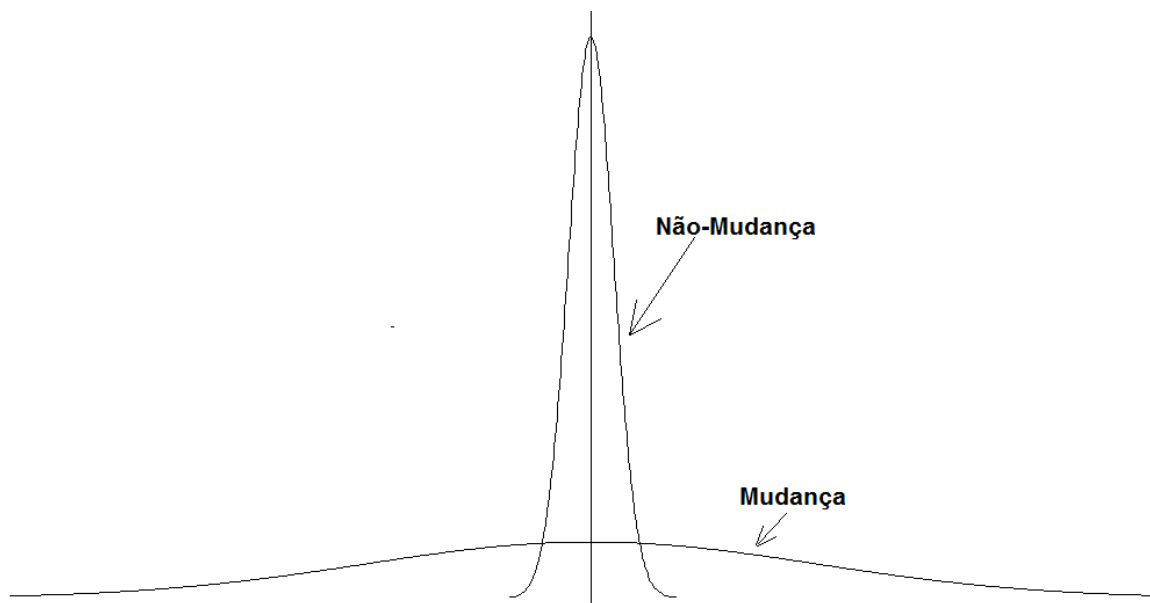


Figura 15 – Distribuições das classes mudança e não-mudança. Fonte: Adaptado de Zanotta, 2010.

### 3.2.2.1 O Algoritmo EM

Como os dados são assumidos como normalmente distribuídos, o algoritmo EM pode ser usado para fins de estimação dos parâmetros (vetor de médias, matriz de covariâncias e probabilidade *a priori*) dessas duas distribuições (Duda et al., 2001).

As estimações de máxima verossimilhança geradas pelo algoritmo são calculadas iterativamente a partir de duas etapas: *expectation* e *maximization*. A etapa 1 calcula uma expectativa do logaritmo da verossimilhança para as estimações consideradas e a etapa 2 calcula os parâmetros que maximizam as expectativas geradas pelo logaritmo da verossimilhança calculado na etapa 1. O algoritmo atualiza as estimações até a convergência.

Etapa 1:

$$\begin{aligned} \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) &= \frac{p(\mathbf{X}_k | \omega_i, \hat{\theta}_i) P(\omega_i)}{\sum_{j=1}^c p(\mathbf{X}_k | \omega_j, \hat{\theta}_j) P(\omega_j)} \\ &= \frac{|\hat{\Sigma}_i|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{X}_k - \hat{\mu}_i)' \hat{\Sigma}_i^{-1} (\mathbf{X}_k - \hat{\mu}_i)\right] \hat{P}(\omega_i)}{\sum_{j=1}^c |\hat{\Sigma}_j|^{-\frac{1}{2}} \exp\left[-\frac{1}{2}(\mathbf{X}_k - \hat{\mu}_j)' \hat{\Sigma}_j^{-1} (\mathbf{X}_k - \hat{\mu}_j)\right] \hat{P}(\omega_j)} \end{aligned} \quad (70)$$

Etapa 2:

$$\hat{P}(\omega_i) = \frac{1}{n} \sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) \quad (71)$$

$$\hat{\mu}_i = \frac{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) \mathbf{X}_k}{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta})} \quad (72)$$

$$\hat{\Sigma}_i = \frac{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta}) (\mathbf{X}_k - \hat{\mu}_i) (\mathbf{X}_k - \hat{\mu}_i)'}{\sum_{k=1}^n \hat{P}(\omega_i | \mathbf{X}_k, \hat{\theta})} \quad (73)$$

Onde: o vetor  $\theta_i$  representa os parâmetros da classe  $\omega_i$  e  $c$  representa o número de classes. No caso do problema de detecção de mudanças deste trabalho,  $c = 2$ . O  $\mu_i$  é a média da classe  $i$ . Neste trabalho será utilizada a notação  $M_1$  e  $M_2$  para as médias de cada classe e para  $\Sigma_i$ , que é a matriz de covariância da classe  $i$ , será utilizada a notação  $S_1$  e  $S_2$  para as matrizes de covariância de cada classe.

Para a escolha dos valores iniciais dos parâmetros no algoritmo EM, algumas considerações são necessárias e podem ser ilustradas por meio da figura 15.

A classe mudança apresenta uma distribuição bivariada, alongada na direção de maior dispersão, que é estimada pelo primeiro autovalor ( $\lambda_1$ ). Para valor inicial no EM propõe-se utilizar a matriz de covariância para a totalidade das amostras de diferenças nas frações. Já as amostras da classe não-mudança estão concentradas em torno da origem, apresentando uma pequena dispersão, causada pelos inevitáveis ruídos nos dados. Essa dispersão será estimada pela variância na direção ortogonal à dispersão de maior variação, ou seja, na direção do segundo autovetor. Essa variância é estimada pelo segundo autovalor ( $\lambda_2$ ). A distribuição será, portanto, circular em torno da origem com um raio proporcional a  $\lambda_2$  (Zanotta e Haertel, 2012).

Como geralmente o número de *pixels* que apresentam pequena mudança ou mudança nula é muito maior do que o número de *pixels* que apresentam sinais claros de mudança, os valores iniciais das probabilidades *a priori* podem ser estimados inicialmente como 0,1 para mudança e 0,9 para não-mudança.

Consideram-se aqui as classes  $\omega_1$  e  $\omega_2$ , respectivamente, como as classes mudança e não-mudança. Assim,  $M_1$ ,  $S_1$ ,  $P(\omega_1)$ ,  $M_2$ ,  $S_2$ ,  $P(\omega_2)$  representam os vetores de médias, matrizes de covariâncias e probabilidades *a priori* de cada classe. Assim, as estimativas iniciais para cada classe ficam:

$$M_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_1 = \begin{bmatrix} \text{matriz de} \\ \text{covariância} \\ \text{da amostra total} \end{bmatrix}, P(\omega_1) = 0,1, M_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_2 = \begin{bmatrix} \lambda_2 & 0 \\ 0 & \lambda_2 \end{bmatrix} \text{ e } P(\omega_2) = 0,9 \quad (74)$$

Após a convergência do EM, os parâmetros obtidos foram utilizados para a obtenção de amostras para cada classe, que posteriormente foram utilizadas para treinar o classificador SVM.

### 3.2.2.2 Classificação com SVM e SVM One-Class

SVM e SVM *One Class* constituem métodos supervisionados de classificação, que utilizam de amostras de treinamento. As amostras de treinamento para cada classe podem ser estabelecidas com visitas ao local, mapas, fotografias aéreas (Richards e Jia, 2006). O que muda nesta proposta é a forma de extração das amostras de treinamento.

A classificação com SVM utilizada neste trabalho está baseada em Abe (2005), cujo problema pode ser expresso pelas equações (38) e (39). Foram utilizados os *kernels* polinomial e RBF, descritos na tabela 6 do capítulo 2. Os algoritmos utilizados para a implementação do SVM foram desenvolvidos com o uso do *software Matlab*.

Como se assumiu dados normalmente distribuídos, após a estimação dos parâmetros, para fins de treinamento do classificador SVM foram utilizadas amostras randômicas extraídas a partir de uma distribuição normal multivariada, obtidas por meio da função *mvnrnd.m*, do *Matlab*, usando em cada caso os parâmetros da respectiva distribuição estimados via EM.

Para a classificação com SVM *One-Class*, os algoritmos utilizados para a implementação também foram desenvolvidos em *Matlab*. Para fins de treinamento do classificador também foram utilizadas amostras randômicas extraídas a partir de uma distribuição normal multivariada, obtidas por meio da função *mvnrnd.m*, do *Matlab*, mas usando apenas os parâmetros da distribuição não-mudança estimados via EM. Isso porque, conforme descrito no capítulo 2, a abordagem de SVM *One-Class* utilizada neste trabalho busca atrair o hiperplano de separação para o local, onde se encontram a maior parte dos dados (não-mudança). Foram utilizadas as equações (65), (66) e (67).

Como as amostras foram selecionadas de forma aleatória, foram realizados diversos testes a fim de verificar os parâmetros que estavam gerando melhores resultados. A escolha dos parâmetros que seriam utilizados foi baseada em estudos anteriores envolvendo SVM e SVM *One-Class* (Campbell e Bennett, 2000; Andreola, 2009; Bonesso, 2013). Escolhidos os parâmetros que produziram melhores acurácias, foram feitos novos testes com parâmetros fixos e foi calculada a acurácia média obtida e a variância (Gualtieri, 2009; Muñoz-Marí et al, 2009).

### **3.2.3 Medidas de Avaliação da Acurácia**

Para a avaliação da acurácia dos resultados obtidos neste trabalho foram utilizados dois procedimentos:

- a) Análise qualitativa, realizada por meio da produção do mapa de mudança após cada classificação.
- b) Análise quantitativa, realizada por meio da construção da matriz de confusão.

Como a avaliação quantitativa necessita de verdade terrestre para a construção da matriz de confusão, foram realizados dois tipos de experimentos: um com imagem sintética e outro com imagem real.

A maior utilidade da construção de uma imagem sintética reside na possibilidade do conhecimento das alterações ocorridas. A partir da construção desta imagem foram selecionadas amostras de testes para cada uma das classes.

No caso do problema da detecção de mudanças, a matriz de confusão apresenta o formato abaixo, descrito na tabela 7, onde as colunas representam a verdade terrestre e as linhas, a classificação obtida.

Tabela 7– Matriz de confusão

	Verdade Terrestre	
Classificação	$a_{11}$	$a_{12}$
	$a_{21}$	$a_{22}$

Fonte: Produzido pela autora, 2014.

Considerando a classe 1 como mudança e a classe 2 como não-mudança, os termos da matriz podem ser interpretados da seguinte maneira:

$a_{11}$  = quantidade de amostras da classe mudança que foram classificadas como mudança.

$a_{12}$  = quantidade de amostras da classe não-mudança que foram classificadas como mudança.

$a_{21}$  = quantidade de amostras da classe mudança que foram classificadas como não-mudança.

$a_{22}$  = quantidade de amostras da classe não-mudança que foram classificadas como não-mudança.

Radke et al (2005) define algumas medidas que podem ser extraídas da matriz de confusão. A seguir estão descritas as que foram utilizadas neste trabalho e que também aparecem em Richards e Jia (2006) e Bittencourt (2011). Aqui, considera-se:

Taxa de Falsos Alarmes – representa a proporção de *pixels* que não sofreram mudanças, mas foram classificados na classe mudança e pode ser estimada pela equação (76).

$$FA = \frac{a_{12}}{a_{11} + a_{12}} \quad (75)$$

Taxa de Detecção ou Acurácia do Usuário para a Classe Mudança – representa a proporção de *pixels* classificados como mudança dentre aqueles que são realmente *pixels* de mudança e pode ser estimada pela equação 76.

$$AU = \frac{a_{11}}{a_{11} + a_{21}} \quad (76)$$

Essa acurácia estima a confiabilidade de que um *pixel* classificado como mudança efetivamente pertence a essa classe.

Dessa forma, são propostas neste trabalho duas metodologias utilizando as imagens de fração: SVM, com *kernels* polinomial e RBF, e SVM *One-Class* na detecção de alterações em imagens multitemporais. No capítulo 4 serão descritos os experimentos realizados para fins de teste das metodologias propostas.



## 4 TESTES E EXPERIMENTOS

### 4.1 Introdução

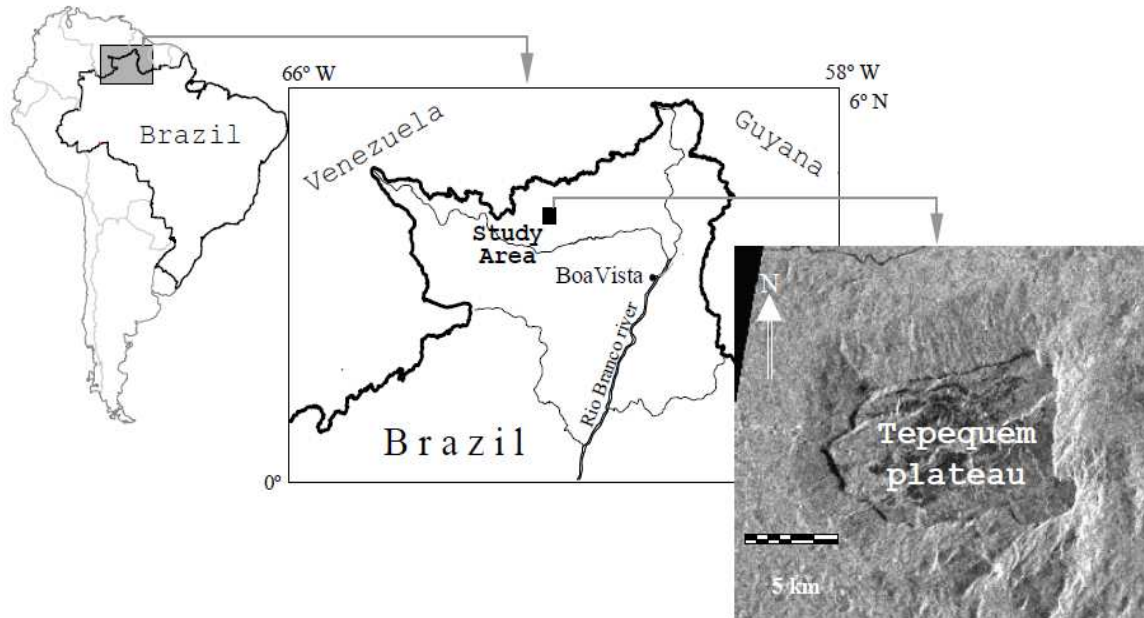
Neste capítulo são apresentados e discutidos os experimentos realizados para fins de teste da metodologia proposta. Os algoritmos utilizados para a implementação foram desenvolvidos em ambiente *Matlab*. As imagens e mapas de mudança foram visualizados utilizando o *software MultiSpec* (Landgrebe e Biehl, 2002). São apresentados dois tipos de experimentos: um com imagens sintéticas e outro com imagens reais. Tanto no primeiro tipo de experimento quanto no segundo foram realizados testes com SVM utilizando *kernels* polinomial e RBF e com SVM *One-Class*. O experimento com dados sintéticos teve como objetivo testar o processo com um par de imagens, onde se tinha um conhecimento das alterações existentes. Os testes empregando imagens reais foram executados utilizando duas imagens *Landsat 5-TM* da mesma região, mas adquiridas em datas distintas.

### 4.2 Descrição das Imagens

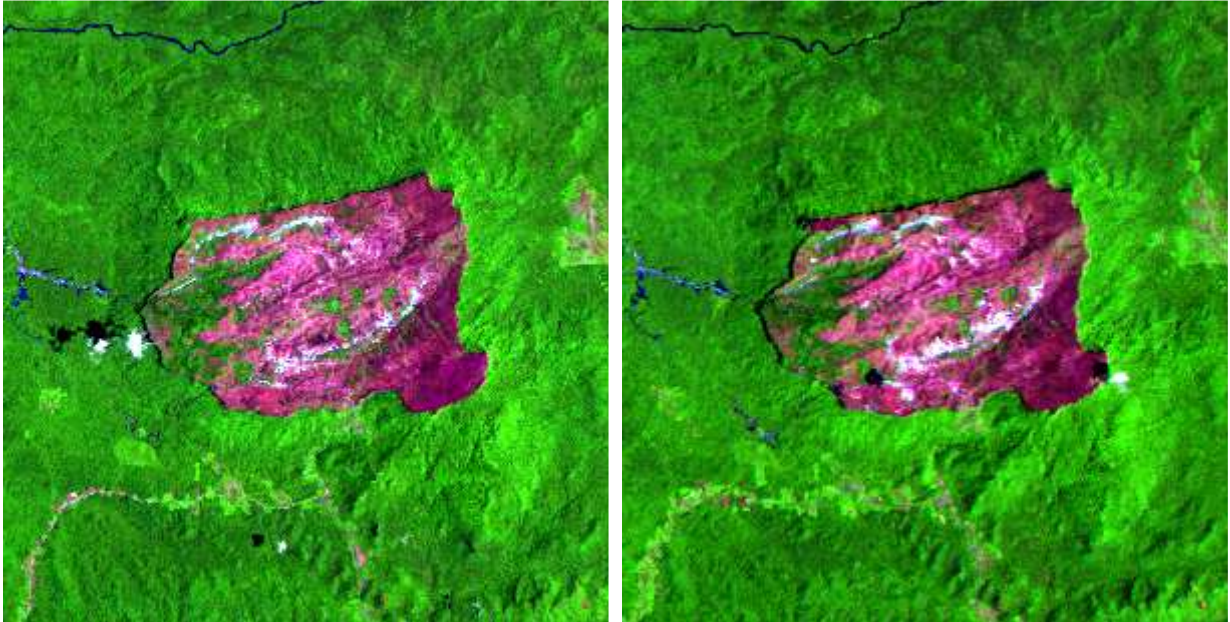
Para os experimentos foram consideradas duas imagens *Landsat 5-TM* cobrindo a mesma área, localizada no estado de Roraima, definida pelas coordenadas 61°37'W–61°49'W de longitude e 3°40'N–3°52'N de latitude. A razão dessa escolha deve-se à experiência adquirida em trabalhos anteriores (Haertel et al., 2004; Zanotta, 2010). Essa área é caracterizada por duas planícies distintas, uma localizada a uma altitude média de 200 metros e outra localizada numa alta elevação, formando um platô: o Platô do Tepequém. Uma floresta densa cobre a região de baixa elevação, enquanto a região do platô é dominada por cerrado. O clima nessa região é tropical, com uma temporada de seca bem definida (de outubro a abril) e uma estação chuvosa (de maio a setembro). A precipitação média anual varia dentro da faixa de 1750 milímetros a 2000 milímetros. A ocorrência de ouro e diamantes nessa região tem sido relatada na literatura (Haertel et al., 2004), ocasionando garimpos clandestinos.

A imagem da primeira data foi adquirida em outubro de 1991 e a imagem da segunda data em abril de 1994.

As imagens utilizadas possuem 390.625 *pixels* distribuídos em um quadrado de 625 por 625 *pixels*, conforme mostra a figura 16:



(a)



(b)

(c)

Figura 16 – (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.

### 4.3 Produção das Imagens de Fração

Conforme mencionado no capítulo 3, neste trabalho as imagens de diferenças são produzidas a partir da subtração de componentes das imagens de fração geradas a partir das imagens multiespectrais (Haertel *et al.*, 2004). 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).

Com base no conhecimento prévio da área, selecionaram-se os componentes: vegetação, solo e sombra/água como componentes na cena. Para o fracionamento, foram utilizadas as seis bandas reflectivas disponibilizadas pelo sistema sensor *Landsat 5-TM* e, a partir do MLME, três imagens de fração foram produzidas. No entanto, conforme mencionado no capítulo 3, nesta tese só foram consideradas as frações solo e vegetação, que apresentam boa correlação negativa. A figura 17 mostra as imagens-fração geradas para a imagem de 1991.

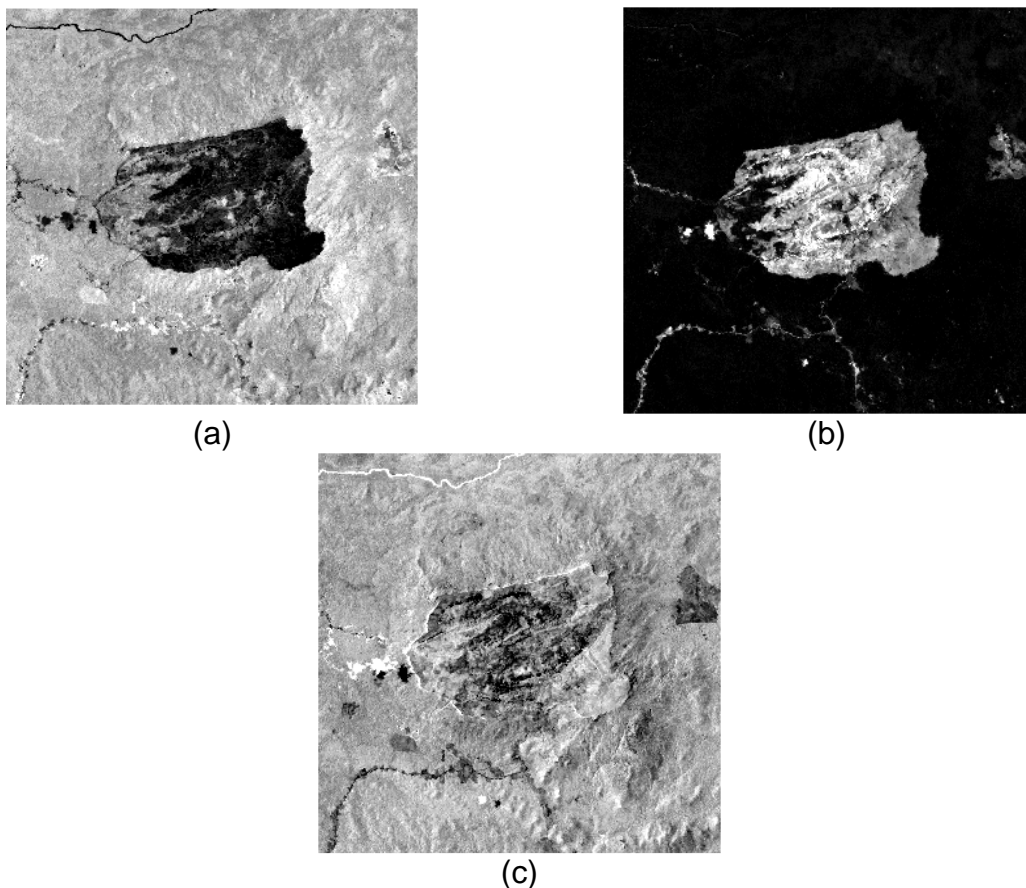


Figura 17 – Imagens-fração da imagem de 1991 para (a) vegetação, (b) solo e (c) sombra. Variação contínua de preto (0%) a branco (100%). Fonte: Haertel *et al.*, 2004.

#### 4.4 Produção da Imagem Sintética

Conforme mencionado na seção 4.1, o primeiro tipo de experimento utilizou uma imagem sintética. A maior utilidade nesta abordagem reside na possibilidade do conhecimento das alterações ocorridas. A imagem sintética foi produzida por meio da inclusão de alterações controladas na imagem *Landsat-TM* empregada como imagem da data 1.

As mudanças aplicadas sinteticamente correspondem ao simples transporte de algumas áreas para outras regiões da imagem. O objetivo da construção desta imagem foi selecionar amostras de teste para as classes mudança e não-mudança.

Utilizando a imagem de data 1 selecionaram-se duas regiões. A primeira, de vegetação, foi transportada para um local próximo, também de vegetação, para que posteriormente fossem selecionadas amostras de teste para a classe não-mudança. A segunda, de solo, foi transportada para um local de vegetação para que posteriormente fossem selecionadas amostras de teste para a classe mudança, conforme mostra a figura 18.

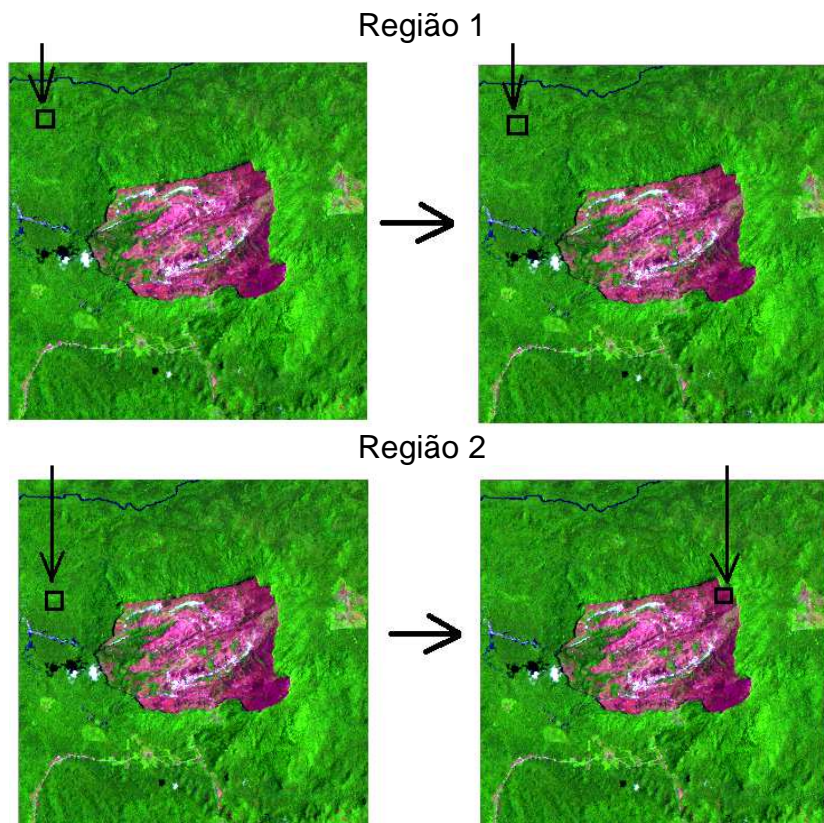


Figura 18 – Mudanças sintéticas produzidas da imagem. Em cada par, a figura à esquerda mostra a região copiada e à direita o local onde foi colada. Fonte: Produzido pela autora, 2014.

As regiões utilizadas na construção da imagem sintética foram quadrangulares, contendo cada uma um total de 900 *pixels*. Posteriormente, foram selecionados, de cada uma das regiões, conjuntos com 200, 300 e 400 amostras de teste para cada classe, sendo que da região 1 foram selecionadas as amostras para a classe não-mudança e da região 2 amostras da classe mudança.

#### 4.5 Produção das Imagens de Diferença

Nos testes com dados reais, as imagens de diferença foram obtidas pela diferença *pixel a pixel* entre a imagem fração da data 2 (1994) e a imagem fração da data 1 (1991).

A figura 19 mostra a imagem de diferença produzida nos testes com a imagem real e a figura 20 dos histogramas para as frações de vegetação e solo.

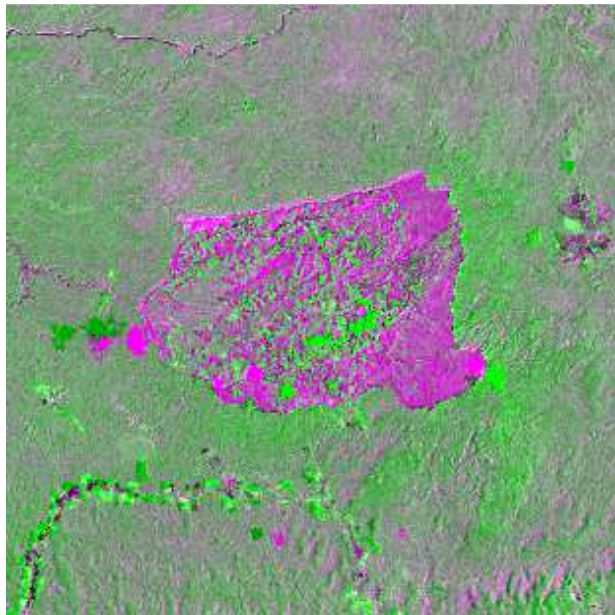


Figura 19 – Imagem de diferença com dados reais. Composição em falsa cor 3 (R) 2(G) 3(B). Fonte: Produzido pela autora, 2014.

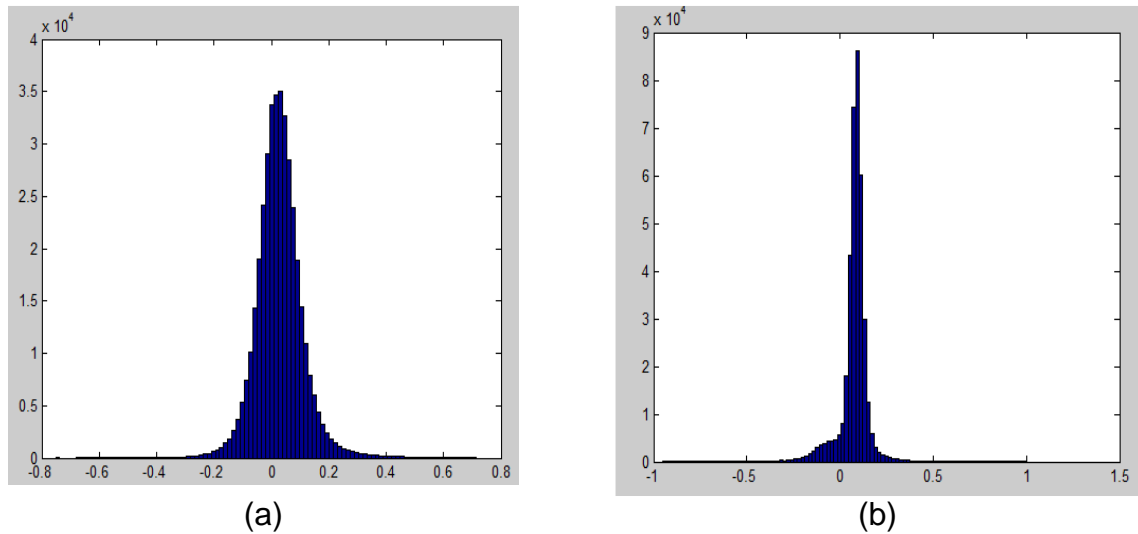


Figura 20 – Histogramas da imagem de diferenças. (a) Diferenças entre as frações de vegetação. (b) Diferenças entre as frações de solo. Fonte: Produzido pela autora, 2014.

Nos testes com dados sintéticos, as imagens de diferença foram obtidas pela diferença *pixel a pixel* entre a imagem fração alterada sinteticamente e a imagem fração original da data 1.

A imagem sintética foi utilizada apenas para a seleção de amostras de teste. Todo o processo de seleção de amostras de treinamento foi realizado com as imagens reais.

#### 4.6 Estimação dos parâmetros das distribuições das classes mudança e não-mudança

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

A figura 21 a seguir, mostra o espalhamento dos dados da imagem diferença obtida no experimento. O primeiro autovalor é a variância estimada inicialmente para a classe mudança, enquanto o segundo é a variância estimada inicialmente para a classe não-mudança.

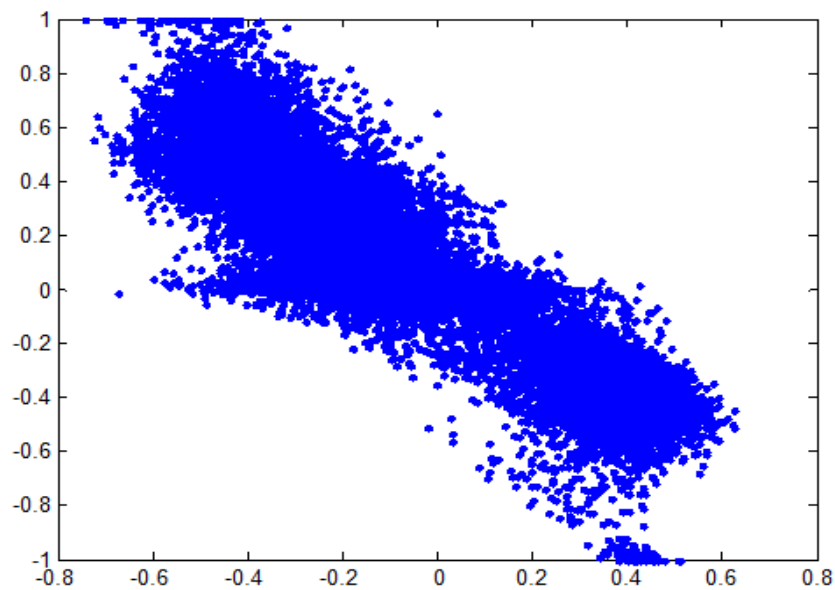


Figura 21 – Espalhamento dos dados da imagem-diferença no experimento com dados reais. Fonte: Produzido pela autora, 2014.

Consideram-se aqui as classes  $\omega_1$  e  $\omega_2$ , respectivamente, como as classes mudança e não-mudança. Assim,  $M_1$ ,  $S_1$ ,  $P(\omega_1)$ ,  $M_2$ ,  $S_2$ ,  $P(\omega_2)$  representam os vetores de médias, matrizes de covariâncias e probabilidades *a priori* de cada classe.

Os parâmetros estatísticos das distribuições de probabilidade envolvidas na mistura de densidades foram estimados da maneira descrita no capítulo 3.

As estimativas iniciais para cada classe foram:

$$M_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_1 = \begin{bmatrix} 0,0068 & -0,0028 \\ -0,0028 & 0,0099 \end{bmatrix}, M_2 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, S_2 = \begin{bmatrix} 0,0042 & 0 \\ 0 & 0,0042 \end{bmatrix}, P(\omega_1) = 0,1 \text{ e}$$

$$P(\omega_2) = 0,9.$$

Após a convergência, os valores obtidos foram:

$$M_1 = \begin{bmatrix} 0,0537 \\ 0,0297 \end{bmatrix}, S_1 = \begin{bmatrix} 0,0174 & -0,0099 \\ -0,0099 & 0,0267 \end{bmatrix}, M_2 = \begin{bmatrix} 0,0214 \\ 0,0890 \end{bmatrix}, S_2 = \begin{bmatrix} 0,0036 & -0,0004 \\ -0,0004 & 0,0009 \end{bmatrix},$$

$$P(\omega_1) = 0,22 \text{ e } P(\omega_2) = 0,78.$$

Esses valores foram utilizados como parâmetros de entrada no SVM e no SVM *One-Class* para a seleção de amostras de treinamento por meio da função *mvnrnd.m*, de *Matlab*, tanto nos testes com a imagem real, quanto com a imagem sintética.

## 4.7 Testes utilizando SVM com *Kernels* Polinomial e RBF

Os algoritmos utilizados para a implementação do SVM foram desenvolvidos em ambiente *Matlab*. Conforme já mencionado, para fins de treinamento do classificador SVM foram utilizadas amostras randômicas extraídas a partir de uma distribuição normal multivariada, obtidas por meio da função *mvnrnd.m*, de *Matlab*, usando em cada caso os parâmetros da respectiva distribuição, estimados via EM.

Os experimentos com SVM foram feitos com os *kernels* polinomial e gaussiano (RBF) com 200 e 300 amostras de treinamento para cada classe, utilizando as equações descritas no capítulo 3. Os parâmetros utilizados estão descritos nos subitens a seguir.

Nos testes com a imagem sintética, foram realizados experimentos com 200, 300 e 400 amostras de teste para cada classe e realizada uma avaliação quantitativa baseada em indicadores extraídos da matriz de confusão. Nos testes com a imagem real foi feita uma análise qualitativa através do mapa de mudança produzido após a classificação.

O tempo de processamento nos testes com a imagem real foi em média 45 minutos, uma vez que a imagem utilizada possuía um total de 390.625 *pixels*. Já no experimento com dados sintéticos, como foram utilizadas 200, 300 e 400 amostras de teste para cada classe, o tempo de processamento por teste foi, em média, 3 minutos.

### 4.7.1 Testes com a Imagem Sintética

Em todos os experimentos foram realizados testes com 200 e com 300 amostras de treinamento para cada classe e 200, 300 e 400 amostras de teste para cada classe. As amostras de testes foram selecionadas conforme descrito no item 4.4.

No caso do *kernel* polinomial, para o parâmetro  $d$  (grau do polinômio), foram realizados testes com  $d$  variando de 2 a 5. O valor utilizado para a variável  $C$  foi 10, por ter apresentado bons resultados em trabalhos anteriores (Andreola, 2009).

No caso do *kernel* RBF para o parâmetro  $\gamma$  foram utilizados 1, 2, 5 e 10 e parâmetro  $C$  também igual a 10.



As tabelas 8 a 13 mostram a acurácia obtida com o uso de cada *kernel* para 200 e 300 amostras de treinamento, com destaque para os parâmetros em que se obteve a melhor acurácia. As tabelas mostram a taxa de falsos alarmes e a acurácia do usuário (ou taxa de detecção).

As figuras 22 a 27 mostram o gráfico comparativo dos resultados com 200 e 300 amostras de treinamento para cada tipo de *kernel*.

Como as acurácias do usuário foram semelhantes, para escolher os parâmetros que obtiveram os melhores resultados, observaram-se os que produziram as menores taxas de falsos alarmes. Embora os parâmetros com melhor acurácia tenham sido distintos em cada teste, é possível verificar que no caso do *kernel* polinomial o grau 2 foi melhor em três dos testes e no *kernel* RBF  $\gamma$  igual a 5 foi melhor em dois, e no outro teste os resultados foram semelhantes. Dessa forma, foram considerados os melhores parâmetros.

Tabela 8 – Acurácia do SVM polinomial utilizando a imagem sintética e 200 amostras de teste

Número de Amostras de Treinamento	Grau	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
<b>200</b>	<b>2</b>	<b>94,5%</b>	<b>0%</b>
300	2	94,5%	8%
200	3	94,5%	3,5%
300	3	94,5%	2,5%
200	4	94%	17%
300	4	94,5%	13,5%
200	5	97%	8%
300	5	96,5%	11%

Fonte: Produzido pela autora, 2014.

Tabela 9 – Acurácia do SVM polinomial utilizando a imagem sintética e 300 amostras de teste

Número de Amostras de Treinamento	Grau	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
200	2	96,3%	1,3%
300	2	96,3%	2%
200	3	96%	1,3%
300	3	96%	3%
200	4	95,7%	1,7%
300	4	96%	20%
200	5	94,7%	2,3%
300	5	96%	15,3%

Fonte: Produzido pela autora, 2014.

Tabela 10 – Acurácia do SVM polinomial utilizando a imagem sintética e 400 amostras de teste

Número de Amostras de Treinamento	Grau	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
200	2	96,3%	0,3%
300	2	97,8%	8%
200	3	96,3%	1,5%
300	3	96,8%	10,5%
200	4	95,8%	0,8%
300	4	96,3%	19%
200	5	97,3%	3%
300	5	96,5%	30,5%

Fonte: Produzido pela autora, 2014.

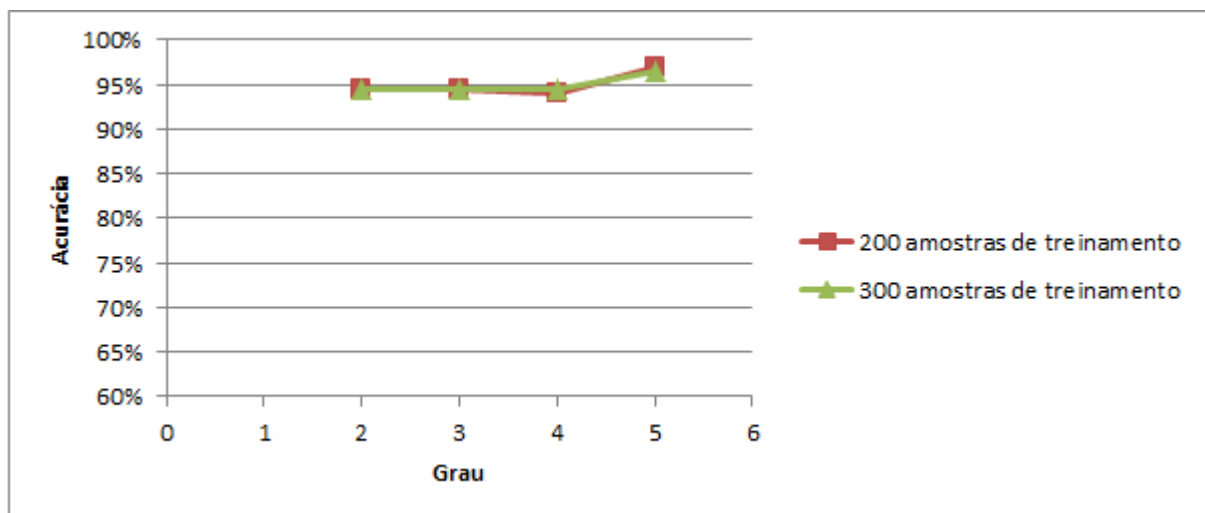


Figura 22 – Comparação da acurácia do usuário no SVM Polinomial nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 200 amostras de teste. Fonte: Produzido pela autora, 2014.

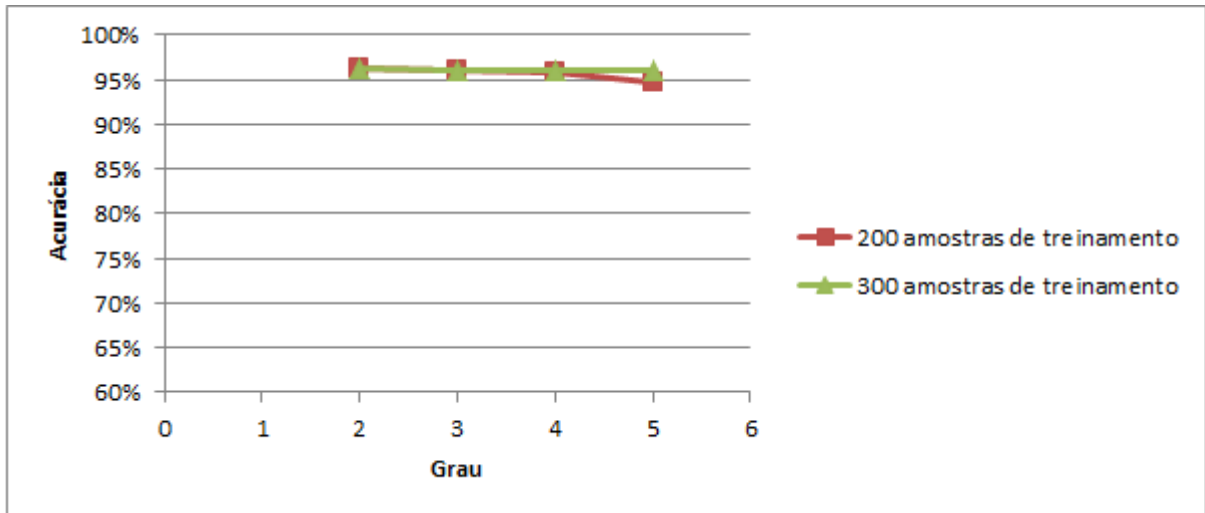


Figura 23 – Comparação da acurácia do usuário no SVM Polinomial nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 300 amostras de teste. Fonte: Produzido pela autora, 2014.

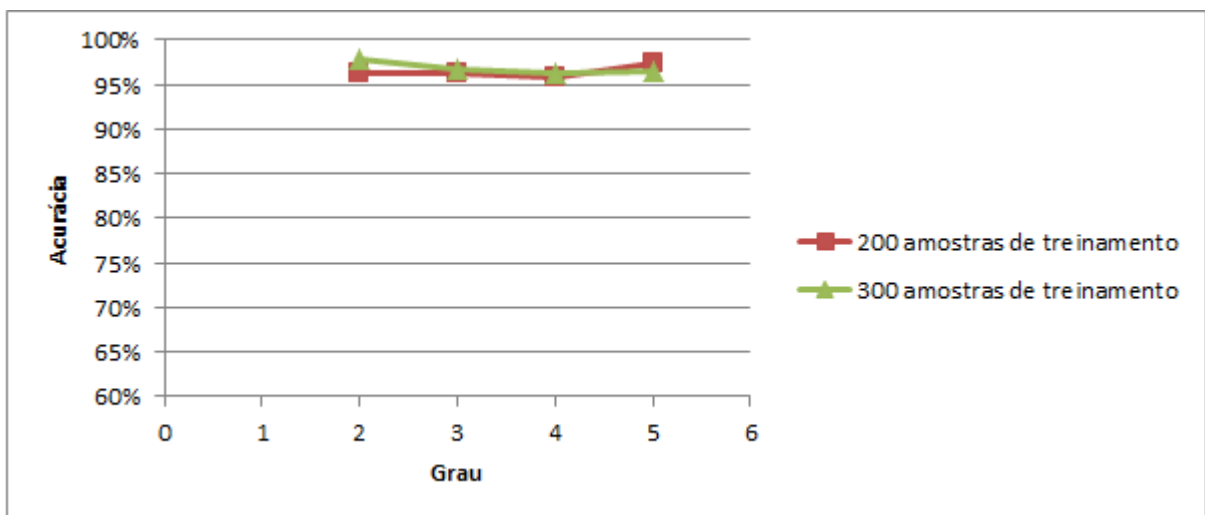


Figura 24 – Comparação da acurácia do usuário no usuário no SVM Polinomial nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 400 amostras de teste. Fonte: Produzido pela autora, 2014.

Tabela 11 – Acurácia do SVM RBF utilizando a imagem sintética e 200 amostras de teste

Número de Amostras de Treinamento	$\gamma$	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
200	1	97%	3%
300	1	92,5%	3,5%
200	2	94%	5,5%
300	2	97,5%	8%
200	5	97%	8,5
<b>300</b>	<b>5</b>	<b>98%</b>	<b>3,5%</b>
200	10	98,5%	14%
300	10	98%	4%

Fonte: Produzido pela autora, 2014.

Tabela 12 – Acurácia do SVM RBF utilizando a imagem sintética e 300 amostras de teste

Número de Amostras de Treinamento	$\gamma$	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
200	1	97%	5,3%
300	1	94,7%	2,3%
200	2	96%	2,7%
300	2	97%	5,3%
<b>200</b>	<b>5</b>	<b>96%</b>	<b>1,7%</b>
300	5	97%	4%
200	10	97,7%	28,7%
300	10	97,3%	24%

Fonte: Produzido pela autora, 2014.

Tabela 13 – Acurácia do SVM RBF utilizando a imagem sintética e 400 amostras de teste

Número de Amostras de Treinamento	$\gamma$	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
<b>200</b>	<b>1</b>	<b>96,8%</b>	<b>1,8%</b>
300	1	96,5%	10,3%
200	2	96%	2,5%
300	2	97%	14%
<b>200</b>	<b>5</b>	<b>96,8%</b>	<b>1,8%</b>
300	5	97%	5,8%
200	10	97,8%	15,5%
300	10	98,3%	20,8%

Fonte: Produzido pela autora, 2014.

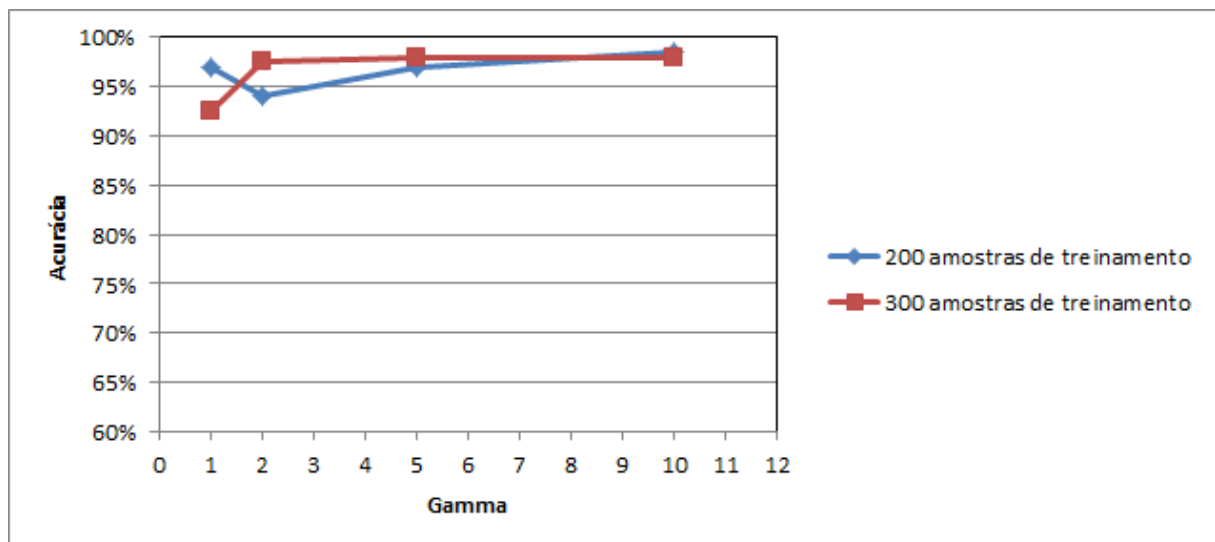


Figura 25 – Comparação da acurácia do usuário no SVM RBF nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 200 amostras de teste. Fonte: Produzido pela autora, 2014.

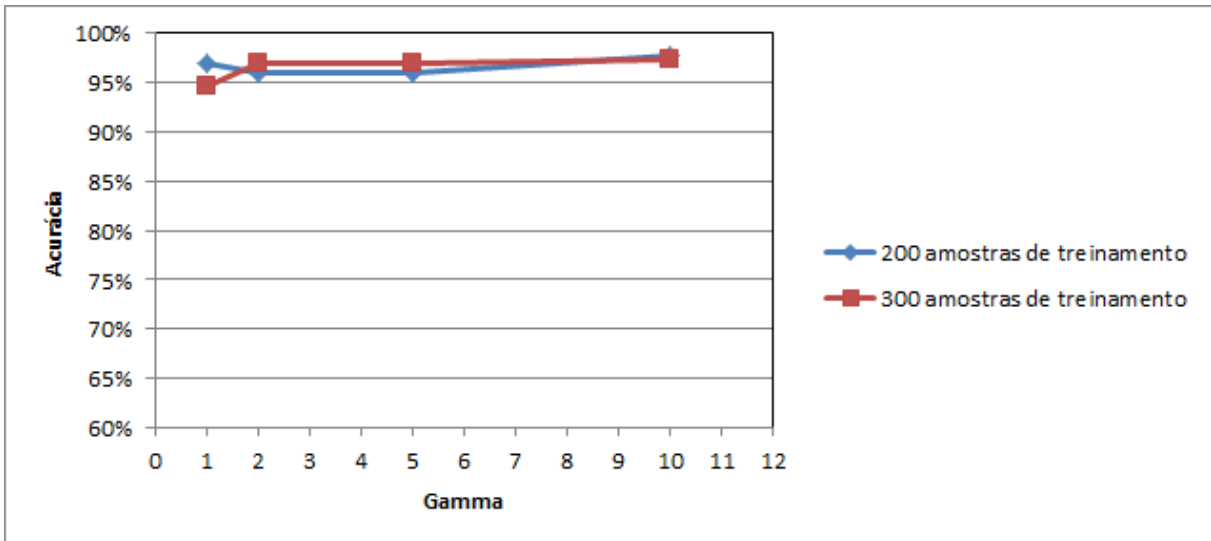


Figura 26 – Comparação da acurácia do usuário no SVM RBF nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 300 amostras de teste. Fonte: Produzido pela autora, 2014.

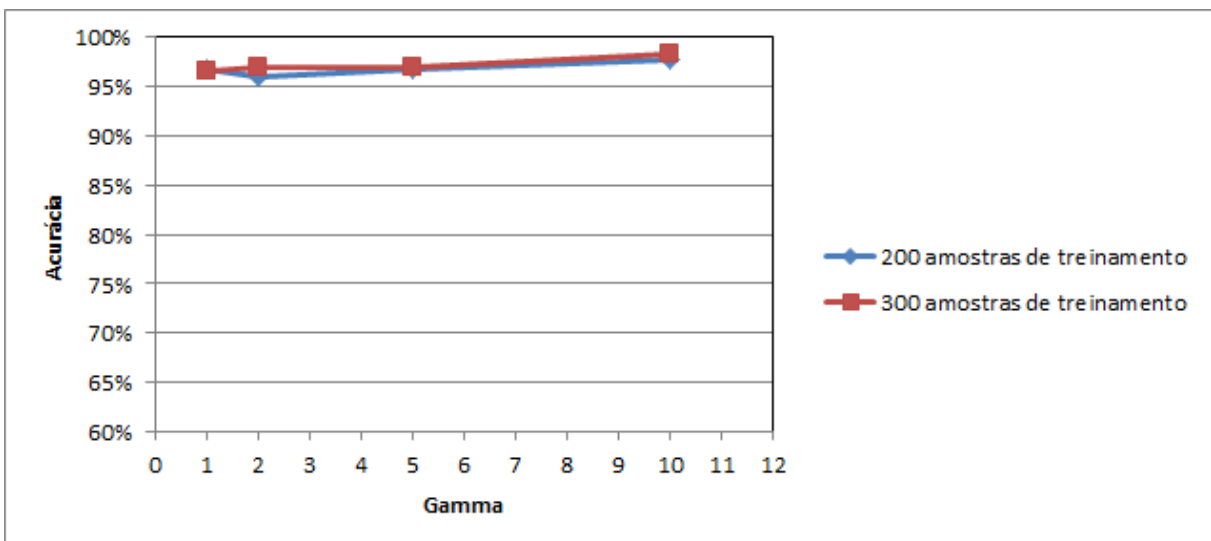


Figura 27 – Comparação da acurácia do usuário no SVM RBF nos testes com a imagem sintética com 200 e 300 amostras de treinamento e 400 amostras de teste. Fonte: Produzido pela autora, 2014.

#### 4.7.2 Testes com a Imagem Real

Em todos os experimentos foram realizados testes com 200 e 300 amostras de treinamento para cada classe. Os parâmetros utilizados aqui foram os mesmos dos testes com a imagem sintética, para que os resultados pudessem ser comparados. Ou seja, em ambos os casos o valor utilizado para a variável  $C$  foi 10, no caso do *kernel* polinomial, para o parâmetro  $d$ , foram realizados testes variando de 2 a 5 e no caso do *kernel* RBF para o parâmetro  $\gamma$  foram utilizados os valores 1, 2, 5 e 10.

As figuras 28 a 31 mostram os mapas de mudança para os *kernels* polinomial e RBF, comparando os testes com 200 e 300 amostras de treinamento, onde os pontos pretos representam os *pixels* classificados como mudança.

Uma inspeção visual dos resultados obtidos mostra uma boa concordância com os obtidos em Zanotta (2010), onde os *pixels* que apresentam maiores graus de pertinência à classe mudança correspondem aos classificados como mudança. Também é possível verificar que no caso do *kernel* polinomial os resultados foram semelhantes nos testes com 200 e 300 amostras de treinamento, exceto nos graus 4 e 5, o que também ocorreu nos testes com a imagem sintética.

Observando-se os mapas referentes aos parâmetros que obtiveram as melhores acurácias nos testes com a imagem sintética é possível constatar que nos testes com a imagem real os mapas produzidos para o *kernel* RBF são semelhantes para os casos de 200 e 300 amostras de treinamento, o que não ocorre com o polinomial, em que há uma diferença nos graus 4 e 5.

O mapa obtido no teste com o *kernel* polinomial de grau 2 e 200 amostras de treinamento e o obtido no teste com o *kernel* RBF com  $\gamma = 1$  e 200 amostras de treinamento são semelhantes.

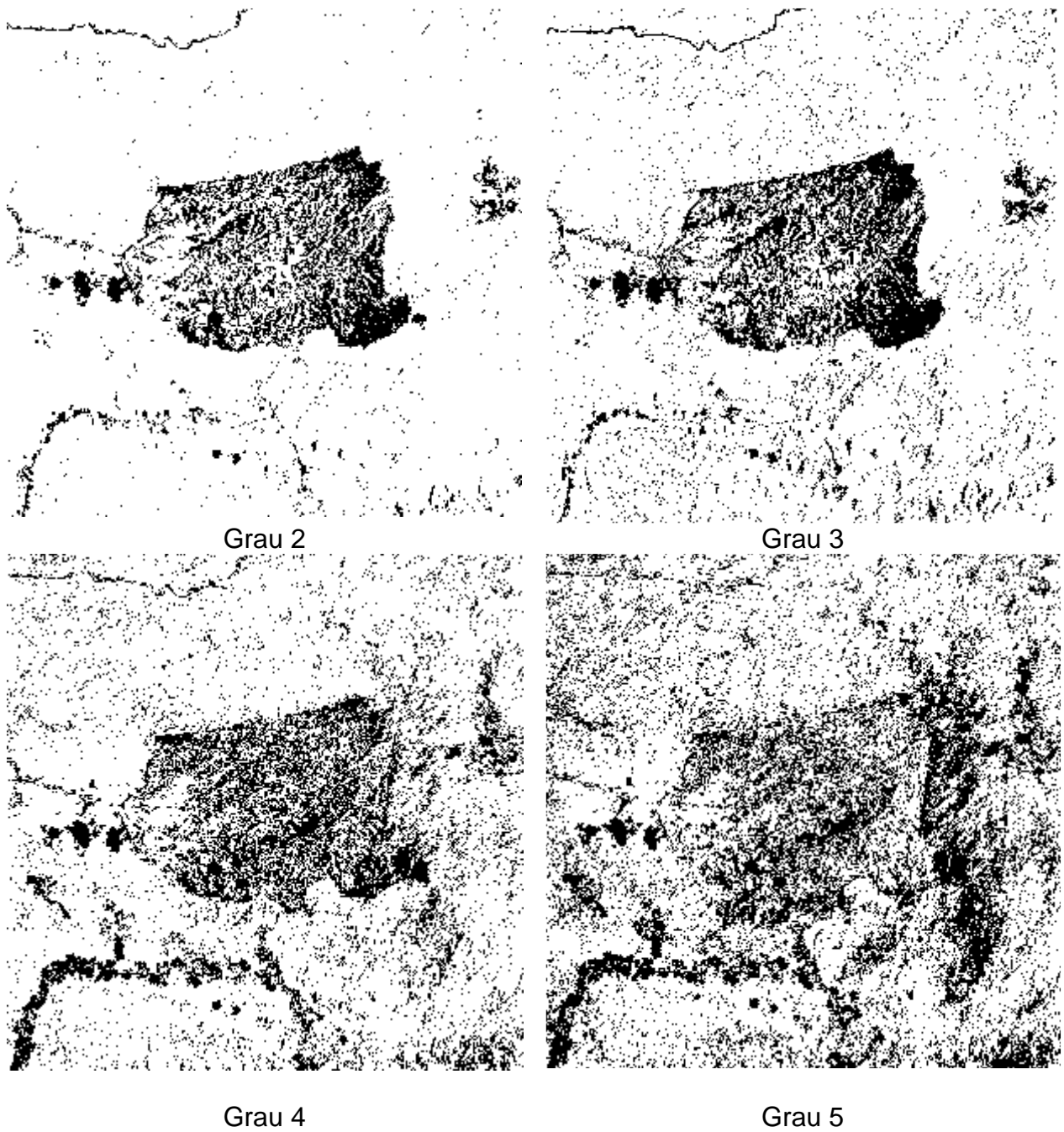


Figura 28 – Mapa de mudança para o SVM polinomial, com 200 amostras de treinamento nos testes com a imagem real. Fonte: Produzido pela autora, 2014.

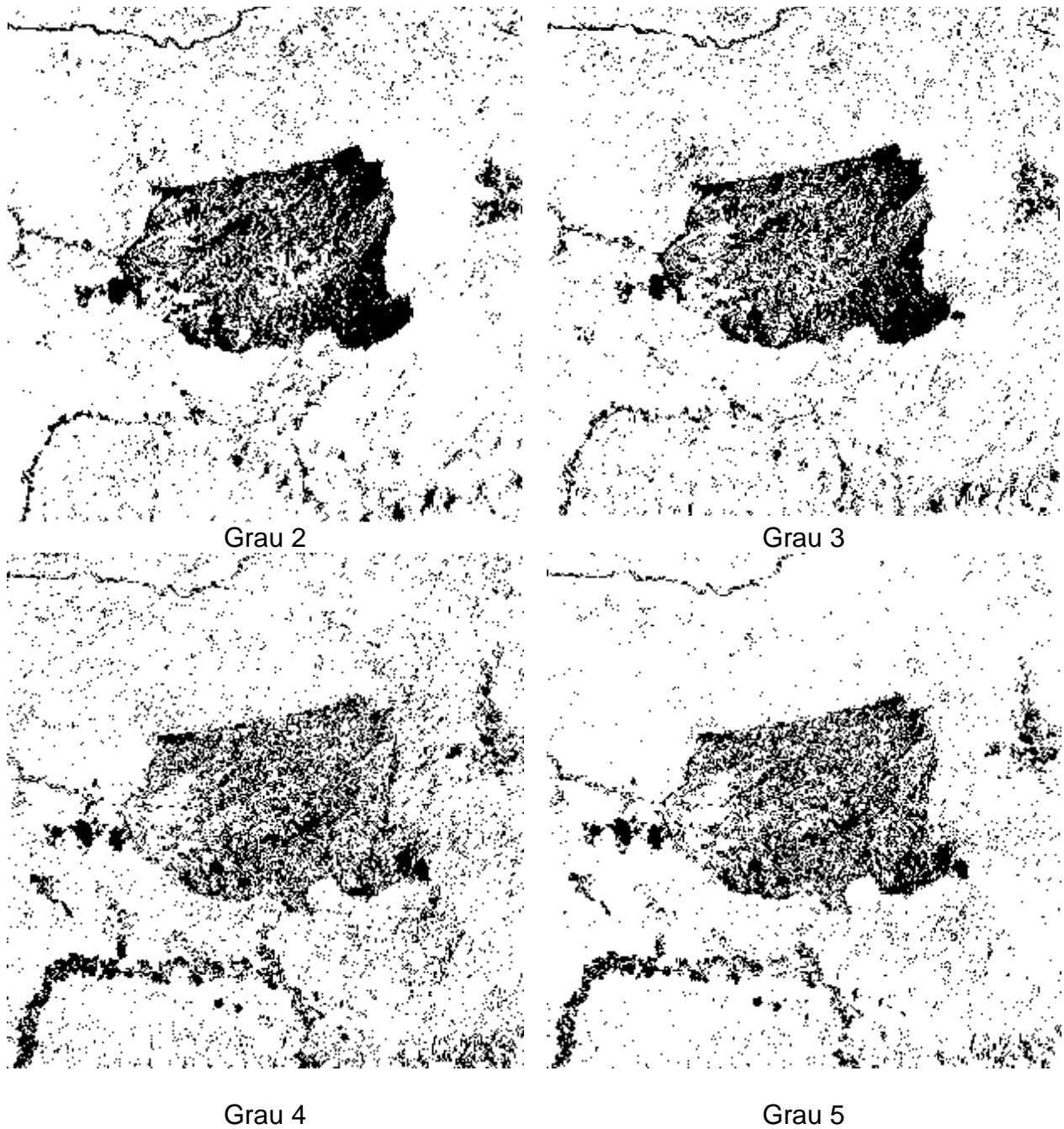


Figura 29 – Mapa de mudança para o SVM polinomial, com 300 amostras de treinamento nos testes com a imagem real. Fonte: Produzido pela autora, 2014.



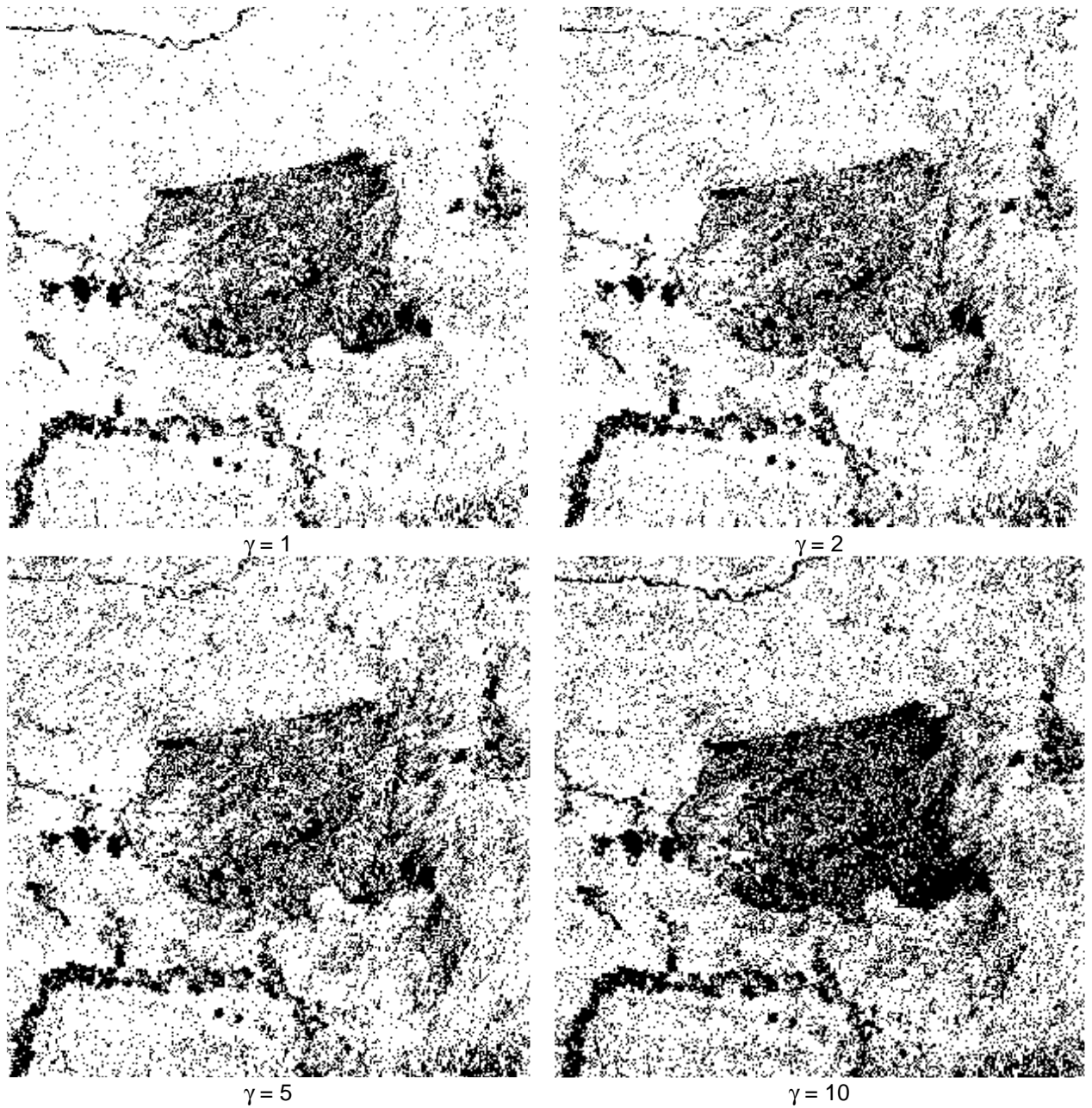


Figura 30 – Mapa de mudança para o SVM RBF, com 200 amostras de treinamento nos testes com a imagem real. Fonte: Produzido pela autora, 2014.

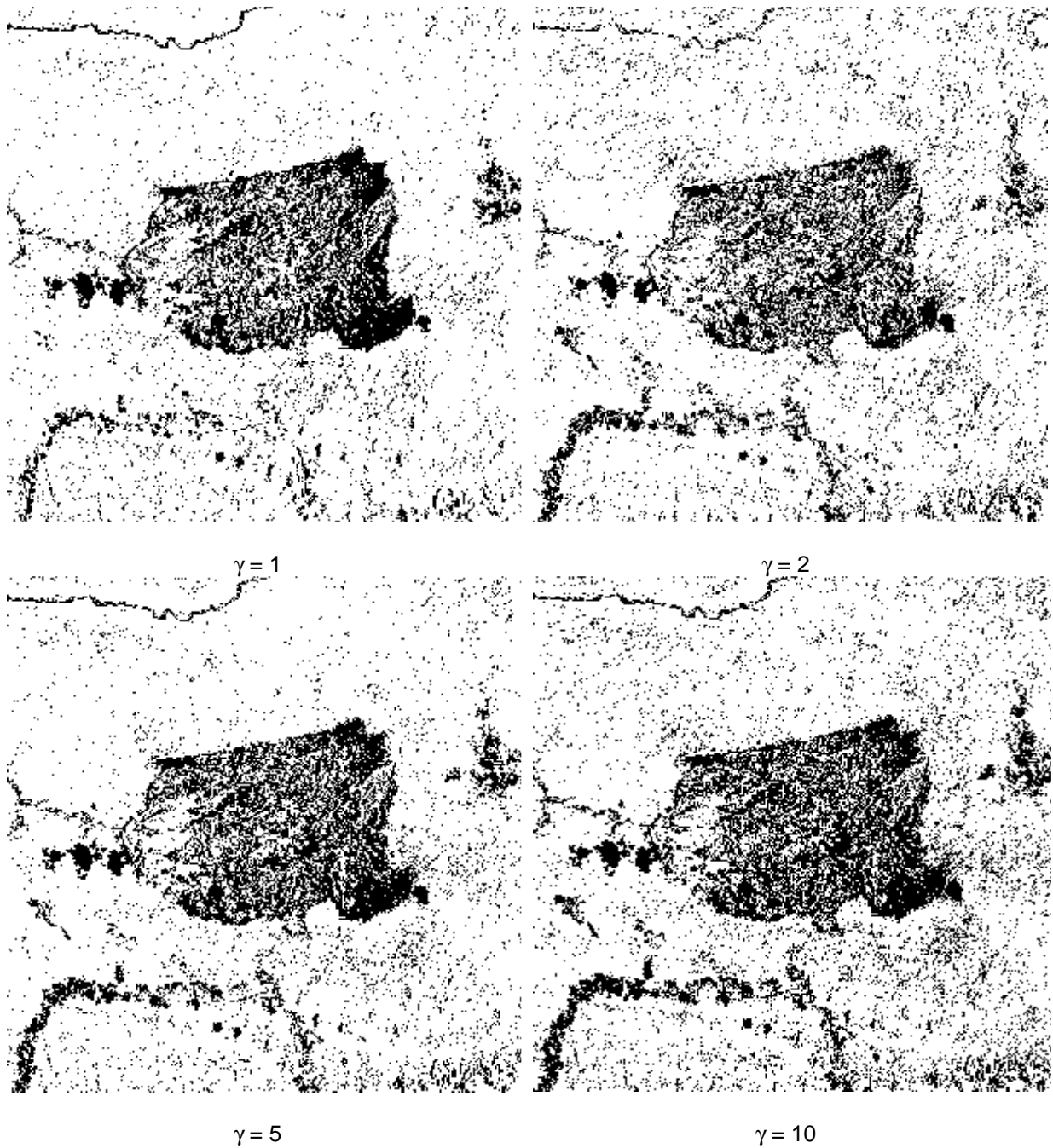


Figura 31 – Mapa de mudança para o SVM RBF, com 300 amostras de treinamento nos testes com a imagem real. Fonte: Produzido pela autora, 2014.

## 4.8 Testes utilizando SVM *One-Class*

Os algoritmos utilizados para a implementação do SVM *One-Class* também foram desenvolvidos em *Matlab*. Para fins de treinamento do classificador também foram utilizadas amostras randômicas extraídas a partir de uma distribuição normal multivariada, obtidas por meio da função *mvnrnd.m*, do *Matlab*, mas usando apenas os parâmetros da distribuição não-mudança estimados via EM. Isso porque, conforme descrito no capítulo 2, a abordagem de SVM *One-Class* utilizada neste trabalho busca atrair o hiperplano de separação para o local onde se encontram a maior parte dos dados (não-mudança). É importante ressaltar que dentre as abordagens de SVM *One-Class* presentes na literatura, a utilizada nesta tese foi a baseada em programação linear, abordada em Campbell e Bennett (2001).

Os parâmetros escolhidos foram baseados em estudos anteriores envolvendo SVM *One-Class*. Bovolo, Camps-Valls e Bruzzone (2010) obtiveram bons resultados com  $\sigma \in [0,01, 1]$ , o que é equivalente a  $\gamma \in [0,5, 5000]$ . Assim, os valores de  $\gamma$  utilizados estão dentro desse intervalo. O parâmetro C foi o mesmo utilizado nos testes com SVM.

O tempo de processamento para cada teste foi, em média, de 5 minutos, tanto nos testes com a imagem sintética quanto nos testes com dados reais.

### 4.8.1 Testes com a Imagem Sintética

Em todos os experimentos foram realizados testes com 200 e com 300 amostras de treinamento para cada classe e 200, 300 e 400 amostras de teste para cada classe. Para o parâmetro  $\gamma$  foram utilizados os valores 20, 30, 40 e 50. O parâmetro de ajuste b foi considerado -0,5.

As tabelas 15 a 17 mostram as acurácias obtidas nos testes com 200 e 300 amostras de treinamento, com destaque para os parâmetros em que se obteve a melhor acurácia. Como as acurácias do usuário foram semelhantes, para a escolha das melhores, foram levados em consideração os resultados obtidos para a taxa de falsos alarmes. A tabela mostra a taxa de falsos alarmes e a acurácia do usuário ou taxa de detecção.

As figuras 32 a 34 mostram os gráficos comparativos para os resultados com 200 e 300 amostras de treinamento.

Tabela 14 – Acurácia do SVM *One-Class* utilizando a imagem sintética e 200 amostras de teste

Número de Amostras de Treinamento	$\gamma$	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
<b>200</b>	<b>20</b>	<b>93%</b>	<b>1,5%</b>
300	20	95,5%	21%
200	30	94%	1,8%
300	30	95,5%	7,5%
200	40	95%	4,8%
300	40	95%	6%
200	50	96%	13%
300	50	96%	10%

Fonte: Produzido pela autora, 2014.

Tabela 15 – Acurácia do SVM *One-Class* utilizando a imagem sintética e 300 amostras de teste

Número de Amostras de Treinamento	$\gamma$	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
200	20	94,7%	1,7%
300	20	95,3%	13,3%
200	30	96%	3,3%
<b>300</b>	<b>30</b>	<b>97,3%</b>	<b>0,7%</b>
200	40	96,3%	7,3%
300	40	97,7%	1,3%
200	50	96,3%	16%
300	50	97,7%	4,3%

Fonte: Produzido pela autora, 2014.

Tabela13 – Acurácia do SVM *One-Class* utilizando a imagem sintética e 400 amostras de teste

Número de Amostras de Treinamento	$\gamma$	Acurácia do Usuário para a Classe Mudança	Taxa de Falsos Alarmes
200	20	95,5%	0,5%
300	20	95,5%	2%
200	30	96,5%	1,5%
<b>300</b>	<b>30</b>	<b>96,8%</b>	<b>0,8%</b>
200	40	96,5%	2%
300	40	97,8%	2,3%
200	50	96,8%	5%
300	50	98%	4,5%

Fonte: Produzido pela autora, 2014.

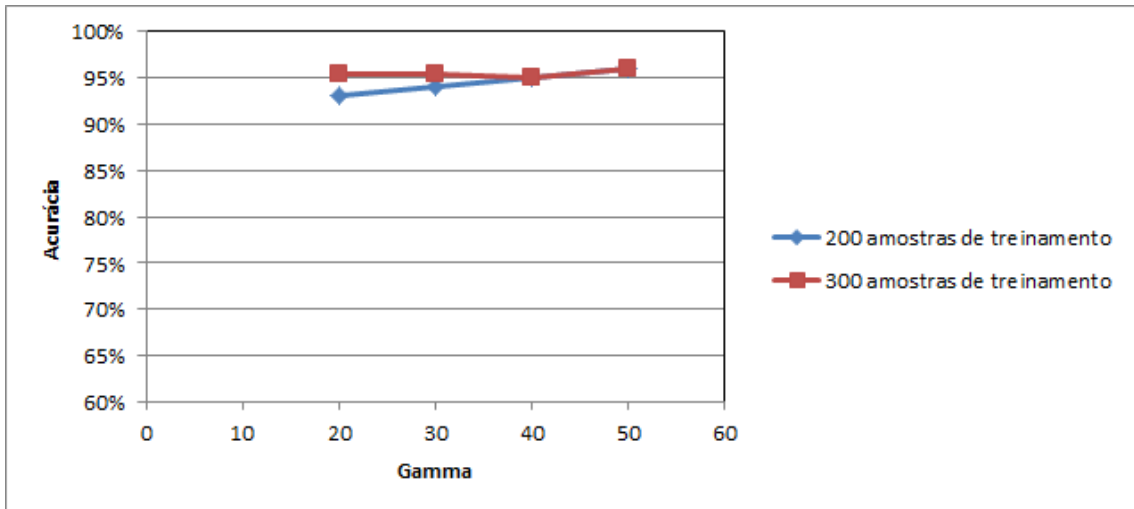


Figura 32 – Acurácia do usuário no SVM *One-Class* nos testes com a imagem sintética com 200 amostras de teste. Fonte: Produzido pela autora, 2014.

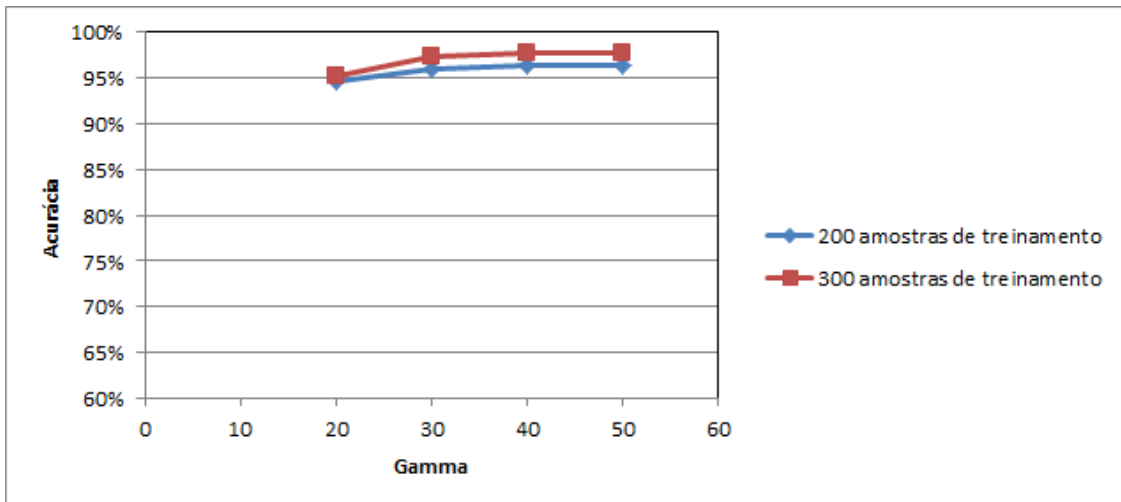


Figura 33 – Acurácia do usuário no SVM *One-Class* nos testes com a imagem sintética com 300 amostras de teste. Fonte: Produzido pela autora, 2014.

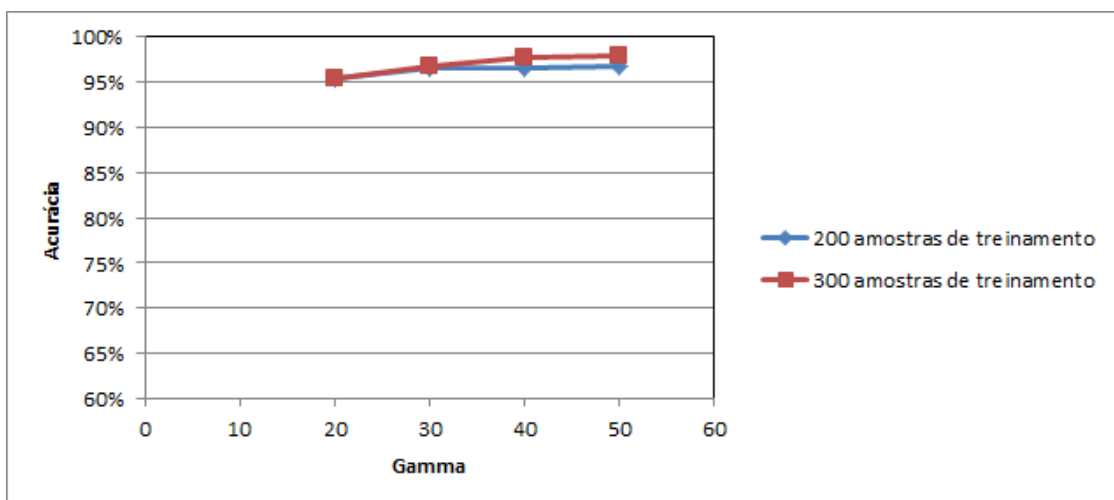


Figura 34 – Acurácia do usuário no SVM *One-Class* nos testes com a imagem sintética com 400 amostras de teste. Fonte: Produzido pela autora, 2014.

#### 4.8.2 Testes com a Imagem Real

Foram realizados testes com 200 e 300 amostras de treinamento. Para que os resultados pudessem ser comparados com os testes utilizando a imagem sintética, os parâmetros utilizados foram os mesmos. As figuras 35 e 36 mostram os mapas de mudança produzidos para cada caso.

O tempo de processamento para cada teste foi, em média, de 5 minutos, bem abaixo do SVM.

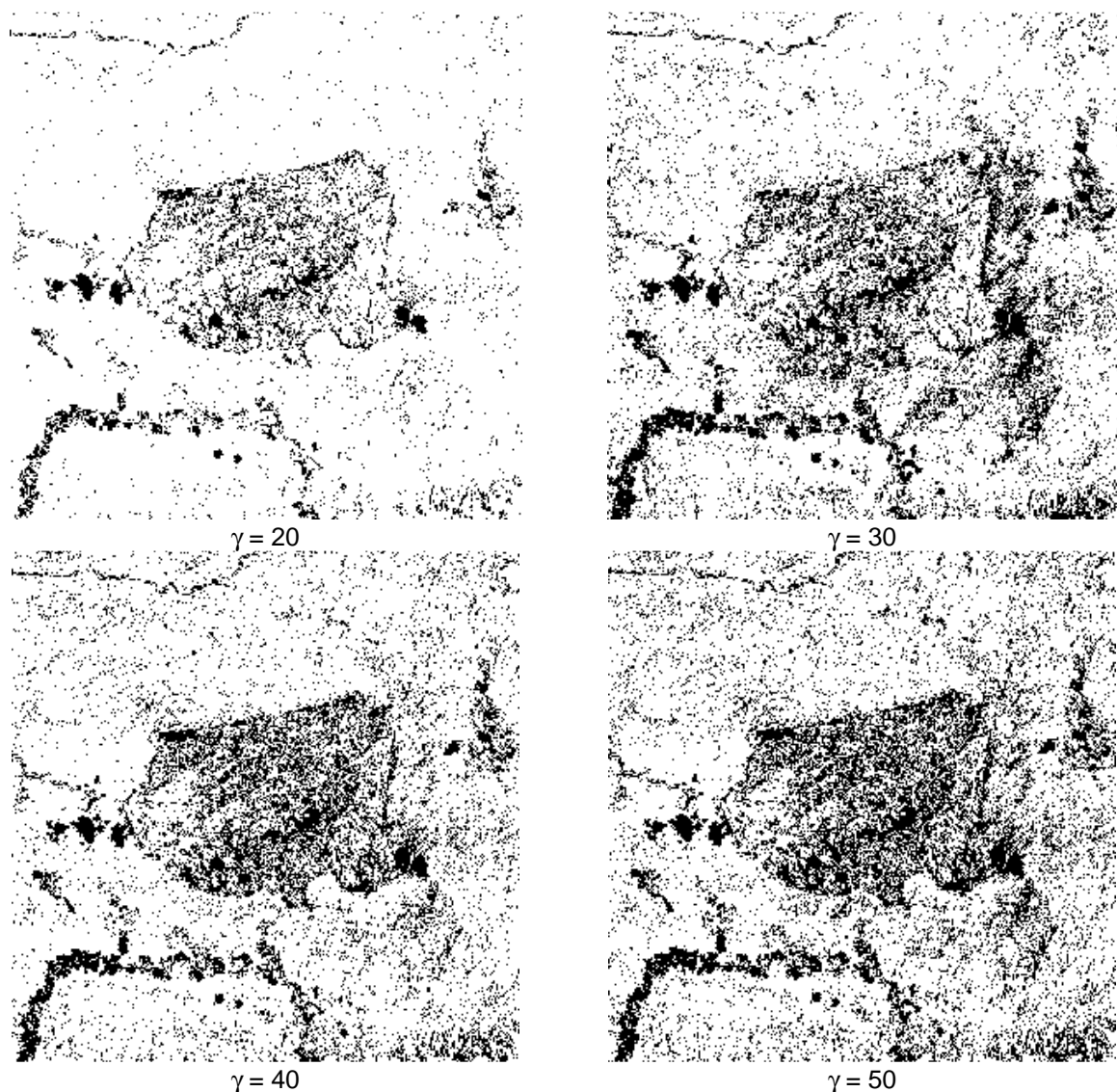


Figura 35 – Mapa de mudança para o SVM *One-Class*, com 200 amostras de treinamento nos testes com a imagem sintética. Fonte: Produzido pela autora, 2014.

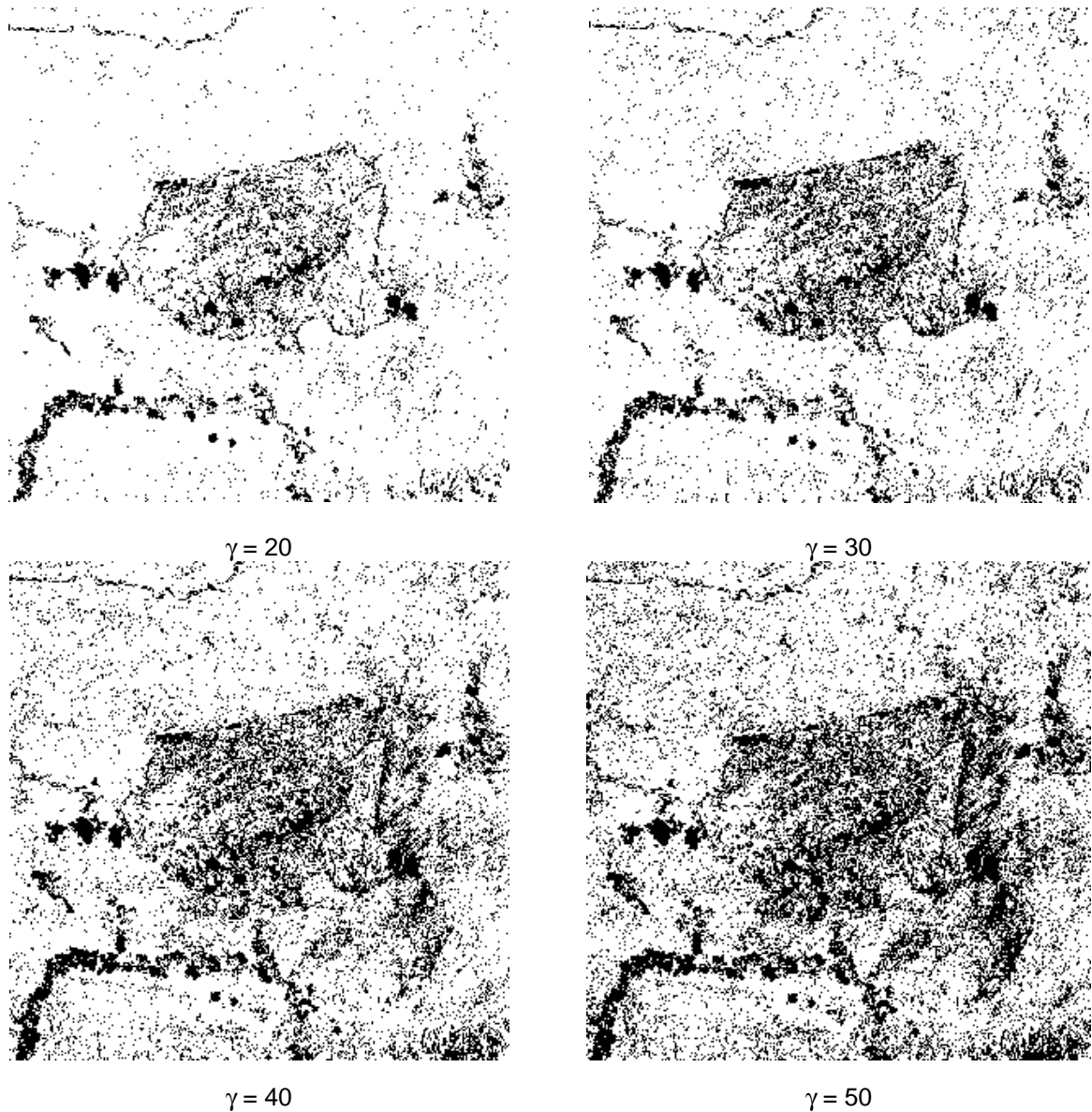


Figura 36 – Mapa de mudança para o SVM *One-Class*, com 300 amostras de treinamento nos testes com a imagem sintética. Fonte: Produzido pela autora, 2014.

Aqui também há uma boa concordância com os obtidos em Zanotta (2010) e com os obtidos no SVM polinomial e RBF. Também é possível observar que os resultados são semelhantes para os casos de 200 e 300 amostras de treinamento, exceto no caso  $\gamma = 50$ , o que ocorreu apenas no conjunto de 300 amostras de teste com a imagem sintética. Embora na imagem sintética, nos experimentos com 200 e 300 amostras de treinamento,  $\gamma = 20$  tenha produzido acurácias diferentes para os casos de 300 e 400 amostras de teste, aqui os mapas foram semelhantes. Acredita-se que o conjunto de amostras de treinamento possa ter influenciado os resultados.

#### 4.9 Comparação de SVM com SVM *One-Class*

Em relação ao tempo de processamento, a abordagem SVM *One-Class* utilizada neste trabalho é mais vantajosa por utilizar programação linear, enquanto o SVM polinomial e RBF utiliza programação quadrática. Cabe salientar que, conforme mencionado no Capítulo 2, existem abordagens SVM *One-Class* envolvendo programação quadrática. No entanto, essas não foram testadas aqui, pois o trabalho de Campbell e Bennett (2001) já demonstrava que a versão envolvendo programação linear produzia bons resultados, mas com a implementação muito mais rápida que aquela que envolvia a programação quadrática.

Quanto às acurácias obtidas em cada tipo de abordagem, a tabela 17 mostra os melhores resultados obtidos no caso do conjunto de 300 amostras de teste de cada classe. Para a construção desta tabela, observaram-se em cada abordagem, quais parâmetros produziram os melhores resultados. Para esses parâmetros foram realizados mais 9 experimentos. A tabela foi construída com base nos 10 experimentos de cada tipo (Muñoz-Marí et al, 2009). A tabela também apresenta a média, a variância e o desvio padrão da acurácia do usuário e da taxa de falsos alarmes. Para a média foi utilizada a aproximação de uma cada decimal e para variância e desvio padrão, cinco casas decimais.

A figura 37 mostra a comparação dos melhores resultados obtidos nos testes com a imagem sintética e a figura 38 apresenta os mapas de mudança correspondentes a esses parâmetros nos testes com dados reais.



Tabela 17 – Comparação SVM Polinomial, RBF e SVM *One-Class* nos testes com a imagem sintética com o conjunto de 300 amostras de teste para cada classe

Tipo	Parâmetros	Acurácia do Usuário para a Classe Mudança			Taxa de Falsos Alarmes		
		Média	Variância	Desvio Padrão	Média	Variância	Desvio Padrão
SVM polinomial	Grau 2 200 amostras de treinamento C =10	96%	$6,8 \cdot 10^{-5}$	0,00822	5,1%	0,00647	0,08041
SVM RBF	$\gamma = 5$ 200 amostras de treinamento C =10	96,3%	$6,2 \cdot 10^{-6}$	0,00249	3,9%	0,00020	0,01415
SVM <i>One-Class</i>	$\gamma = 30$ 300 amostras de treinamento C =10	96,7%	$2,9 \cdot 10^{-5}$	0,00537	16,3%	0,04510	0,21237

Fonte: Produzido pela autora, 2014.

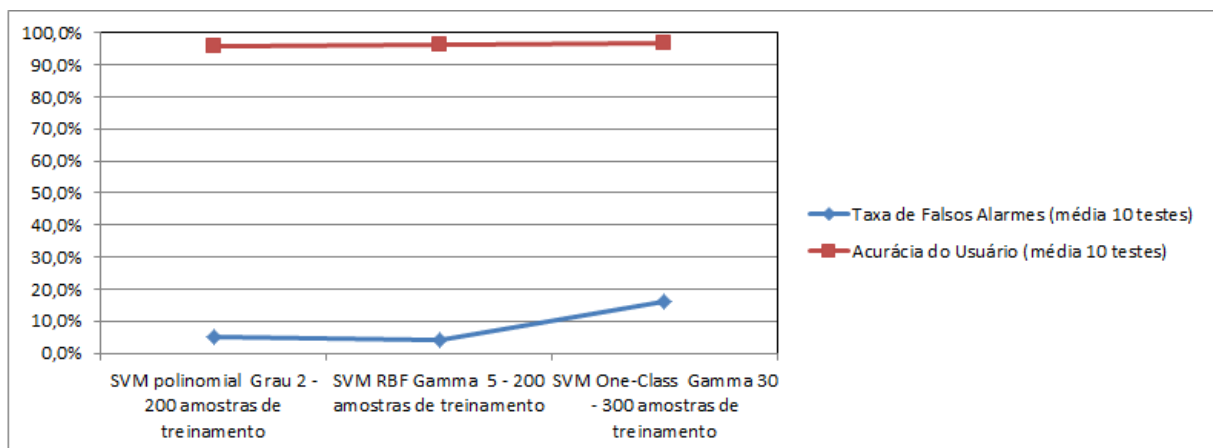


Figura 37 – Comparação SVM Polinomial, RBF e SVM *One-Class* nos testes com a imagem sintética com C = 10. Fonte: Produzido pela autora, 2014.

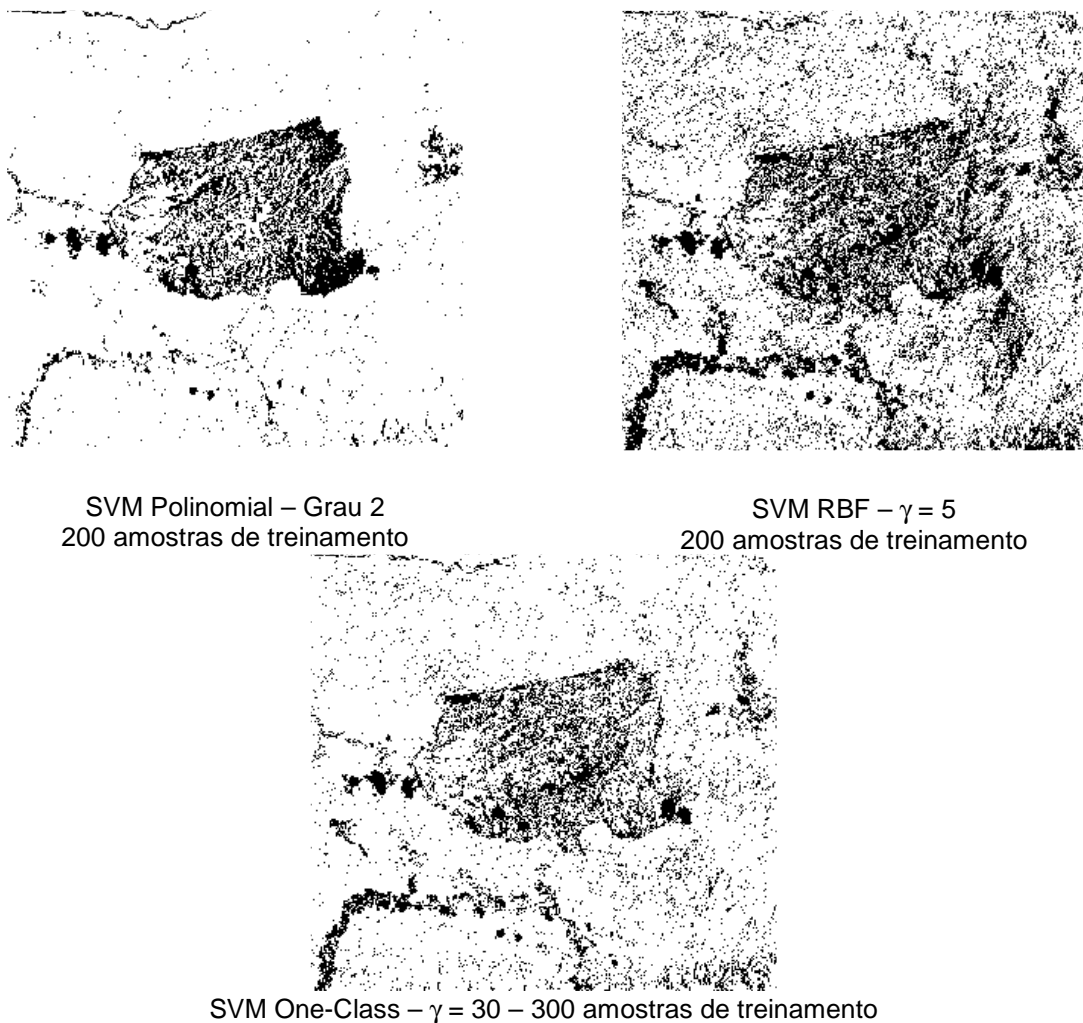


Figura 38 – Comparação dos mapas de mudança produzidos pelo SVM Polinomial, RBF e SVM *One-Class* nos testes com a imagem real. Fonte: Produzido pela autora, 2014.

Quanto às acurácias do usuário, é possível observar que tanto as médias quanto as variâncias foram muito próximas em todos os tipos de experimentos, sendo que o SVM *One-Class* apresentou uma média um pouco maior que os demais. O mesmo não ocorreu com as taxas de falsos alarmes, onde o SVM com *kernel* RBF apresentou a menor média e o SVM *One-Class* a mais alta, o mesmo ocorrendo com as variâncias. Conforme Abe (2005) *kernels* globais como o polinomial não são adequados para alguns tipos de imagens e *kernels* locais tem sido estudados. O SVM *One-Class*, embora tenha tido o melhor resultado dentre as três abordagens no primeiro teste realizado, no conjunto de 10 testes foi o que apresentou a maior variância, mostrando ser mais sensível à escolha das amostras de treinamento. A pesquisa de Muñoz-Marí et al (2009) também apresenta resultados semelhantes.

No capítulo seguinte serão apontadas algumas reflexões sobre os experimentos realizados, conclusões e sugestões para próximos trabalhos.

## 5 CONCLUSÕES

Nesta tese foi proposta uma metodologia para o problema de detecção de mudanças ocorridas em cenas naturais, com base na análise de duas imagens de sensoriamento remoto cobrindo a mesma área, mas adquiridas em duas datas distintas. Para esse fim foi utilizado o conceito de *pixel* mistura e a análise foi desenvolvida a partir das imagens-fração, estimadas para cada uma das duas datas. Conforme mencionado no capítulo 2, essa abordagem apresenta duas vantagens: permite uma análise em nível de subpixel e elimina a necessidade de normalização radiométrica das imagens adquiridas em datas distintas. A classificação foi feita para cada *pixel* da imagem a partir das mudanças ocorridas nas correspondentes frações entre as duas datas.

Baseado na proposta de Zanotta e Haertel (2012) foram utilizados os componentes vegetação e solo, sendo gerados para cada data as imagens-fração desses componentes. Dessa forma, as variáveis aqui consideradas são bidimensionais: diferenças na fração vegetação e na fração solo. Analisando os dados, foi possível constatar a existência de uma correlação negativa entre a distribuição das diferenças das frações de vegetação e solo, explicada naturalmente pelo conseqüente aumento na fração de solo resultante da diminuição da fração vegetação e vice-versa. Assim, foi possível considerar, inicialmente, que duas distribuições gaussianas podem caracterizar as classes mudança e não-mudança. Então o algoritmo EM (Duda et al., 2001) foi utilizado para estimar os parâmetros estatísticos que caracterizam as classes mudança e não-mudança. Esses parâmetros foram utilizados para selecionar amostras de treinamento, o que caracteriza esta proposta como um método de classificação supervisionado, mas com uma nova abordagem para a seleção de amostras de treinamento. Após esta etapa, foi feita uma classificação com o uso de SVM, com amostragem normal multivariada e o SVM *One-Class* foi utilizado para fins de comparação.

É importante salientar que, assim como em Zanotta (2010), o método proposto aqui visa detectar mudanças em ambientes onde há prevaletimento de vegetação e solo, ou seja, possui algumas restrições quanto ao uso em outros ambientes com diferentes componentes, principalmente pelas suposições utilizadas para a determinação dos valores iniciais do algoritmo EM. Sobre o EM, é importante acrescentar que o processo converge para valores confiáveis desde que haja uma

proporção razoável entre as duas distribuições, de forma que o método não alcançará a convergência se em uma das duas classes existir um número muito superior de representantes.

Foram apresentados dois tipos de experimentos: um com imagens sintéticas e outro com imagens reais. Tanto no primeiro tipo de experimento quanto no segundo foram realizados testes com SVM utilizando *kernels* polinomial e RBF e com SVM *One-Class*. O experimento com dados sintéticos teve como objetivo testar o processo com um par de imagens onde se tinha um conhecimento das alterações existentes.

Para a avaliação da acurácia dos resultados obtidos neste trabalho foram utilizados dois procedimentos: análise qualitativa, realizada por meio da produção do mapa de mudança após cada classificação no experimento com dados reais e análise quantitativa, realizada por meio da análise da matriz de confusão no experimento com dados sintéticos.

Os experimentos desenvolvidos nesta tese mostram a adequação da metodologia proposta, produzindo resultados bastante aceitáveis na detecção de alterações na cobertura do solo.

Os resultados foram semelhantes para todos os conjuntos de amostras de testes utilizados, mas o SVM com *kernel* RBF foi o que pareceu ser mais estável, com o comportamento das curvas de acurácias do usuário para a classe mudança e a taxa de falsos alarmes nos testes com 200 e 300 amostras de treinamento muito semelhantes aos experimentos com conjuntos de 200, 300 e 400 amostras de teste, o que não ocorreu com SVM com *kernel* polinomial e SVM *One-Class*. O SVM com *kernel* RBF também foi o que apresentou a menor taxa de falsos alarmes.

Quanto ao SVM *One-class*, é importante salientar o tempo de processamento bem inferior aos demais classificadores. No entanto, quando foram realizados mais três testes a fim de calcular a média das acurácias globais, foi o que obteve a maior variância para as taxas de falsos alarmes e as acurácias do usuário.

## 5.1 Sugestões

Embora a metodologia proposta aqui tenha mostrado resultados aceitáveis, cabe salientar que em cenas naturais, as alterações ocorrem de uma forma gradual e contínua, o que torna difícil a definição objetiva de um limiar separando

rigidamente a classe mudança da classe não-mudança. Dessa forma, sugere-se a investigação de abordagens que aliem as vantagens do classificador SVM, mostradas neste trabalho, a uma classificação em termos de graus de pertinência em cada classe, de forma que permitam que *pixels* pertençam, em parte, a mais de uma classe.

Conforme mencionado anteriormente, este trabalho investigou uma abordagem para detectar mudanças em ambientes onde há prevaecimento de vegetação e solo. Assim, sugere-se que o método seja investigado em cenas com outro tipo de componente, como por exemplo, cenas urbanas.

## 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.
- BARDOSSY, A.; SAMANIEGO, L. Fuzzy Rule-based Classification of Remotely Sensed Imagery. **IEEE Transactions on Geoscience and Remote Sensing**. Vol. 40, Nº. 2, p. 362–374, 2006.
- BAZI, Y.; BRUZZONE, L.; MELGANI, F. **An Unsupervised Approach based on the Generalized Gaussian Model to Automatic Change Detection in Multitemporal SAR Images**. *IEEE Transactions on Geoscience and Remote Sensing*. VOL 43, NO. 4, p. 874–886, 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.
- BOVOLO, F.; CAMPS-VALLS, G.; BRUZZONE, L. A Support Vector Domain Method for Change Detection in Multitemporal Images. **Pattern Recognition Letters**. Vol. 31, Nº. 10, p. 1148–1154, 2010.
- BOW, S. T. **Pattern Recognition and Image Preprocessing**. Marcel Dekker: New York, 1992.
- BRUZZONE, L.; COSSU, R.; VERNAZZA, G. Detection of Land-cover Transitions by Combining Multidate Classifiers. **Pattern Recognition Letters**. Vol. 25, Nº. 13, p. 1491–1500, 2004.
- BRUZZONE, L.; PRIETO, D. F. Automatic Analysis of the Difference Image for Unsupervised Change Detection. **IEEE Transactions on Geoscience and Remote Sensing**. Vol. 38. Nº. 3, p. 1171–1182, 2000.
- BURGES, C. **A tutorial on support vector machines for pattern recognition**. *Data Mining and Knowledge Discovery* 2 (1999) 121–167.
- CAMPBELL, C.; BENNETT, K. P. A Linear Programming Approach to Novelty Detection. In: **Advances in Neural Information Processing Systems**, pages 395-401. MIT Press, 2001.

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.; BRUZZONE, L. Kernel-Based Methods for Hyperspectral Image Classification. **IEEE Transactions on Geoscience and Remote Sensing**, Vol. 43, N<sup>o</sup>. 6, JUNE 2005.

CASTELLANA, L.; D'ADDABBO, A.; PASQUARIELLO, G. A Composed Supervised/Unsupervised Approach to Improve Change Detection from Remote Sensing. **Pattern Recognition Letters**. Vol 28, N<sup>o</sup>. 4, p. 405–413, 2007.

COLDITZ, R. R.; SCHMIDT, M.; DECH, S. A Methodology for Advanced Change Detection with Fuzzy Image Classification. In: **Proceedings of the Fourth International Conference on Electrical Engineering, Computing Science and Automatic Control**, p. 520–524, 2008.

DEFRIES, R. S.; HANSEN, M. C.; TOWNSHEND, J. R. G. Global Continuous Fields of Vegetation Characteristics: a linear mixture model applied to multi-year 8 km AVHRR data. **International Journal of Remote Sensing**. Vol. 21, N<sup>o</sup>. 6 & 7, p. 1389–1414, 2000.

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

FILIPPONE, M.; SANGUINETTI, G. Information Theoretic Novelty Detection. **Pattern Recognition**. Vol 43, N<sup>o</sup>. 3, p. 805–814, 2010.

FOODY, G. M. The Continuum of Classification Fuzziness in Thematic Mapping. **Photogrammetric Engineering and Remote Sensing**, v. 65, N<sup>o</sup>. 4, p. 443–451, 1999.

GUALTIERI, J. A. The Support Vector Machine (SVM) algorithm for supervised classification of hyperspectral remote sensing data. In CAMPS-VALLS, G. BRUZZONE, L. **Kernel Methods for Remote Sensing Data Analysis**. New York: John Wiley & Sons, 2009.

HAASDONK, B.; PEKALSKA, E. Classification with Kernel Mahalanobis Distance Classifiers. In: **Advances in Data Analysis, Data Handling and Business Intelligence: German Classification Society Annual Conference (Annual Conference of the Gesellschaft für Klassifikation)**; Springer Berlin Heilderberg; 2008. p. 315-324.

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

HAMEL, L. **Knowledge Discovery with Support Vector Machines**. A John Wiley & Sons, Inc., Publication, 2009.

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



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

LU, D.; MAUSEL, P.; BRONDIZIO, E.; MORAN, E. Change Detection Techniques. **International Journal of Remote Sensing**. Vol. 25, Nº. 12, p. 2365–2407, 2003.

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.

RADKE, R. J.; ANDRA, S.; AL-KOFAHI, O.; ROYSAM, B. Image Change Detection Algorithms: a Systematic Survey. **IEEE Transactions On Image Processing**. Vol. 14, Nº. 3, p. 294–307, 2005.

RICHARDS, J. A.; JIA, X.; **Remote Sensing Digital Image Analysis**. New York: Springer, 2006. Third Edition.

SCHÖLKOPF, B.; BURGESS, C.; VAPNIK, D. V. Extracting support data for a given task. In: U. M. Fayyad and R. Uthurusamy, editors, Proceedings, **First International Conference on Knowledge Discovery & Data Mining**, Menlo Park, 1995.

SCHÖLKOPF, B.; PLATT, J. C.; SHAWE-TAYLOR, J.; SMOLA, A. J. WILLIAMSON, R. C. Estimating the support of a high-dimensional distribution. **Microsoft Research Corporation Technical Report MSR-TR - 99-87**, 1999.

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.

SINGH, A. Digital Change Detection Techniques using Remotely-Sensed data. **International Journal of Remote Sensing**. v. 43, n. 10, p. 989–1003, 1989.

SHIMABUKURO, Y. E.; SMITH, J. A. The least-squares mixing models to generate fraction images derived from remote sensing multispectral data. **IEEE Transactions on Geoscience and Remote Sensing**, v.29, n.1, 16–20, 1991.

TAX, D., AND DUIN, R. Data domain description by support vectors. In: **In Proceedings of ESANN99** (1999), Ed. M Verleysen, D. Facto Press, Brussels, pp. 251–256.

TSO, B.; MATHER, P. M. **Classification Methods for Remote Sensed Data**. Second ed., CRC Press, Boca Raton, 2009.

WANG, F. Fuzzy Supervised Classification of Remote Sensing Images. **IEEE Transactions on Geoscience and Remote Sensing**. Vol. 28, Nº. 2, p. 194–201, 1990.

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.