

**Universidade de São Paulo
Escola Superior de Agricultura “Luiz de Queiroz”**

Mapeamento de quintais privados por meio de sensoriamento remoto

Caio Hamamura

Dissertação apresentada para obtenção do título de
Mestre em Ciências, Programa: Recursos Florestais.
Opção em: Conservação de Ecossistemas Florestais

**Piracicaba
2013**

Caio Hamamura
Bacharel em Ciências Biológicas

Mapeamento de quintais privados por meio de sensoriamento remoto

versão revisada de acordo com a resolução CoPGr 6018 de 2011

Orientador:
Prof. Dr. **DEMÓSTENES FERREIRA DA SILVA FILHO**

Dissertação apresentada para obtenção do título de Mestre em Ciências, Programa: Recursos Florestais. Opção em: Conservação de Ecossistemas Florestais

**Piracicaba
2013**

**Dados Internacionais de Catalogação na Publicação
DIVISÃO DE BIBLIOTECA - ESALQ/USP**

Hamamura, Caio

Mapeamento de quintais privados por meio de sensoriamento remoto / Caio Hamamura. - - versão revisada de acordo com a resolução CoPGr 6018 de 2011. - - Piracicaba, 2013.
106 p: il.

Dissertação (Mestrado) - - Escola Superior de Agricultura "Luiz de Queiroz", 2013.

1. Quintais 2. Filtro kuwahara 3. Sensoriamento remoto 4. Classificação orientada a objetos I. Título

CDD 711.4
H198m

“Permitida a cópia total ou parcial deste documento, desde que citada a fonte – O autor”

AGRADECIMENTOS

À Deus.

À minha esposa Glaucia pelo amor incondicional, carinho e apoio.

Aos meus pais e sogros pelo amor e apoio.

Aos companheiros de laboratório que propiciaram bons momentos de descontração.

Especialmente aos amigos Sabrina Mieko e Eric Gorgens que foram como co-orientadores no meu trabalho.

Ao técnico do laboratório Jefferson Lordelo Polizel, pela contribuição com o trabalho e pelos momentos de descontração.

Ao orientador do trabalho prof. Dr. Demóstenes Ferreira da Silva Filho pela contribuição essencial para a realização deste trabalho.

À Fundação de Amparo à Pesquisa (FAPESP), pelo apoio financeiro (bolsa de mestrado – processo nº 2010/14602-4) para o desenvolvimento deste trabalho.

SUMÁRIO

| | |
|---|----|
| RESUMO | 9 |
| ABSTRACT | 11 |
| LISTA DE FIGURAS | 13 |
| LISTA DE TABELAS | 15 |
| LISTA DE SIGLAS | 17 |
| 1 INTRODUÇÃO | 19 |
| 2 DESENVOLVIMENTO | 21 |
| 2.1 Revisão Bibliográfica | 21 |
| 2.1.1 Impactos da Urbanização | 21 |
| 2.1.2 Quintais Privados como Recurso da Floresta Urbana | 22 |
| 2.1.3 Mapeamento dos Quintais | 24 |
| 2.1.4 Classificação de Imagens | 25 |
| 2.1.5 Classificação Orientada a Objetos | 27 |
| 2.1.5.1 Segmentação | 27 |
| 2.1.5.1.1 Algoritmos de Segmentação | 28 |
| 2.1.5.1.2 Calibração dos Parâmetros da Segmentação | 29 |
| 2.1.5.1.3 Aferindo a Qualidade do Resultado | 29 |
| 2.1.5.1.4 Algoritmos de Otimização | 30 |
| 2.1.5.2 Extração dos Atributos | 32 |
| 2.1.5.3 Classificação | 33 |
| 2.1.6 Classificador Misto | 34 |
| 2.1.7 Validação da Classificação | 35 |
| 2.1.8 Mercado e Softwares | 36 |
| 2.2 Descrição Física do Local | 37 |
| 2.3 Material e Métodos | 37 |
| 2.3.1 Imagens Utilizadas | 37 |
| 2.3.2 Diagrama do Método Utilizado | 38 |
| 2.3.3 Tratamento das Imagens | 40 |
| 2.3.4 Softwares | 40 |
| 2.3.5 1ª Etapa | 41 |
| 2.3.5.1 Definição das Classes | 41 |

| | | |
|-----------|--|----|
| 2.3.5.2 | Algoritmo B&S | 42 |
| 2.3.5.2.1 | Definição dos Segmentos de Referência para a Segmentação | 42 |
| 2.3.5.2.2 | Calibração dos Parâmetros da Segmentação | 43 |
| 2.3.5.2.3 | Segmentação | 45 |
| 2.3.5.2.4 | Treinamento | 45 |
| 2.3.5.2.5 | Extração dos Atributos dos Segmentos | 45 |
| 2.3.5.2.6 | Classificação | 46 |
| 2.3.5.3 | Bins et al. (1996) | 46 |
| 2.3.5.4 | Filtragem Kuwahara | 46 |
| 2.3.5.4.1 | Segmentação | 47 |
| 2.3.5.4.2 | Extração de Atributos dos Segmentos | 48 |
| 2.3.5.4.3 | Treinamento | 48 |
| 2.3.5.4.4 | Classificação | 48 |
| 2.3.6 | Extração das Áreas Residenciais | 49 |
| 2.3.7 | Segunda etapa | 49 |
| 2.3.7.1 | Definição das classes | 49 |
| 2.3.7.2 | Filtragem Kuwahara | 49 |
| 2.3.7.3 | Classificação ECHO | 49 |
| 2.3.7.3.1 | Treinamento | 49 |
| 2.3.7.3.2 | Classificação | 50 |
| 2.3.7.4 | Classificação Orientada a Objetos | 50 |
| 2.3.7.5 | Análise dos resultados | 50 |
| 2.4 | Resultados e Discussão | 50 |
| 2.4.1 | Aplicativos Desenvolvidos | 50 |
| 2.4.1.1 | Filtro Kuwahara | 50 |
| 2.4.1.2 | Segmentador (C++) | 51 |
| 2.4.1.3 | Calibração por AG (APÊNDICE B) | 51 |
| 2.4.2 | 1ª Etapa | 52 |
| 2.4.2.1 | Segmentação B&S | 52 |
| 2.4.2.1.1 | Aplicativo AG para Calibração dos Parâmetros | 52 |
| 2.4.2.1.2 | Resultados das Segmentações | 57 |
| 2.4.2.1.3 | Chave de Interpretação | 59 |
| 2.4.2.1.4 | Classificação | 60 |
| 2.4.2.2 | Segmentação BINS | 63 |

| | |
|--------------------------------------|----|
| 2.4.3 2ª Etapa..... | 66 |
| 2.4.3.1 Segmentação B&S | 66 |
| 2.4.3.2 Chave de Interpretação | 67 |
| 2.4.3.3 Classificação | 68 |
| 3 CONCLUSÃO | 79 |
| REFERÊNCIAS | 81 |
| APÊNDICES | 91 |

RESUMO

Mapeamento de quintais privados por meio de sensoriamento remoto

O mapeamento da vegetação urbana tem se restringido às áreas públicas ou a área total, não deixando de forma evidente a contribuição dos quintais privados para a vegetação urbana. Alguns trabalhos indicam que essas áreas teriam potencial para mitigar os impactos causados pela urbanização. Neste trabalho testaram-se diversos métodos para tentar mapear as áreas de quintais permeáveis com o intuito de desenvolver um método que possa auxiliar o planejamento urbano e na investigação da contribuição dessas áreas para a vegetação urbana. O trabalho conseguiu resultados com acurácia bastante elevada ($Kappa = 0,9553$) e apresentou técnicas inovadoras de filtragem e classificação, destacando-se o aplicativo desenvolvido para realizar a filtragem Kuwahara que demonstrou melhorar bastante os resultados de classificações por abordagem de pixels. As classificações de abordagem por pixel levaram a resultados estatisticamente até melhores que a classificação orientada a objetos, no entanto, a apresentação visual dos resultados da classificação orientada a objetos é superior pela redução do ruído.

Palavras-chave: Quintais; Filtro kuwahara; Sensoriamento remoto; Classificação orientada a objetos

ABSTRACT

Mapping private gardens with remote sensing

Urban vegetation mapping has been restricted to the public areas or the total area, not showing the contribution of the private yards to the urban greening. Some studies indicate that these areas have potential to mitigate the impacts caused by the urbanization. In this work, many methods for mapping pervious gardens were tested aiming the development of a method that could aid urban planning and investigating the contribution of these areas to the urban greening. This work achieved highly accurate results ($\kappa = 0.9553$) and presented novel techniques for filtering and classification. We highlight the development of a simple application to perform the Kuwahara filter, which improved the results of the classification by pixel approach. The classification algorithms by pixel approach resulted in statistically more accurate products, although visually the results presented by object-oriented approach is closer to landscape features.

Keywords: Gardens; Kuwahra filter; Remote sensing; Object based image analysis

LISTA DE FIGURAS

| | |
|--|----|
| Figura 1 - Esquema ilustrativo das subjanelas utilizadas pelo filtro Kuwahara..... | 27 |
| Figura 2 - Esquema ilustrando o funcionamento do algoritmo evolutivo | 31 |
| Figura 3. Diagrama resumo do método proposto | 39 |
| Figura 4 - Amostras de referência para calibração da segmentação..... | 43 |
| Figura 5 - Exemplo de imagem sem o filtro Kuwahara (a) e com o filtro Kuwahara (b) | 47 |
| Figura 6 - Gráfico demonstrando a relação entre o RBSB dos recortes de 1000 x 1000 e o RBSB usando os mesmos parâmetros para os recortes de 2000 x 2000..... | 55 |
| Figura 7 - Gráficos demonstrando a correlação entre os parâmetros de entrada no segmentador e o índice RBSB, utilizando-se os resultados da Tabela 5 | 57 |
| Figura 8 - Resultados das segmentações com diferentes valores de RBSB. Os indicadores vermelhos demonstram locais em que se visualizam as maiores mudanças..... | 58 |
| Figura 9 - Resultados das classificações. Em amarelo estão os segmentos classificados como residenciais. As setas azuis indicam erros de inclusão e as vermelhas de omissão | 62 |
| Figura 10 - Bandas com equalização de histograma, demonstrando maior destaque do asfalto na banda do infravermelho próximo 1 | 64 |
| Figura 11 - Resultados de segmentações com o algoritmo de Bins et al. (1996) variando os parâmetros de entrada. As setas em azul representam resultados subsegmentados e as setas vermelhas os resultados subsegmentados | 65 |
| Figura 12 - Segmentação do mesmo trecho de imagem sem o filtro Kuwahara, demonstrando a supersegmentação com as setas vermelhas e subsegmentação pelas setas azuis | 66 |
| Figura 13 - Amostras de copa de árvore para a calibração do RBSB | 66 |
| Figura 14 - Exemplo de resultado da segmentação da segunda etapa..... | 67 |
| Figura 16 - Imagem de referência para analisar os resultados da classificação | 74 |
| Figura 17 - Resultado da classificação orientada a objetos com segmentador Baatz e Schäpe (2000) e classificador Random Forests..... | 74 |
| Figura 18 - Resultado da classificação ECHO | 75 |
| Figura 19 - Resultado da classificação ECHO após filtragem Kuwahara..... | 75 |
| Figura 20 - Resultado da classificação pela máxima verossimilhança..... | 76 |
| Figura 21 - Resultado da classificação pela máxima verossimilhança após a filtragem Kuwahara | 76 |

LISTA DE TABELAS

| | |
|---|----|
| Tabela 1 - Resolução espectral do satélite WorldView-2..... | 38 |
| Tabela 2 - Genes ou parâmetros dos indivíduos definidos no algoritmo genético do aplicativo desenvolvido com seus respectivos parâmetros | 52 |
| Tabela 3 - Parâmetros da população utilizados no algoritmo genético do aplicativo desenvolvido | 52 |
| Tabela 4 - Resultados da calibração com o algoritmo genético nos 10 recortes, mostrando a convergência para melhor resultado de RBSB | 53 |
| Tabela 5 - Extrapolação dos três melhores resultados de cada recorte de 1000 x 1000 pixels para um dos recortes de 2000 x 2000 pixels | 54 |
| Tabela 6 - Chave de interpretação para a primeira etapa de classificação | 59 |
| Tabela 7 - Ranking dos melhores atributos em ordem decrescente da soma dos rankings de cada algoritmo | 61 |
| Tabela 8 - Chave de interpretação usada para a segunda etapa da classificação..... | 68 |
| Tabela 9 - Melhores atributos escolhidos pelos diferentes algoritmos ordenados em ordem decrescente de importância..... | 69 |
| Tabela 10 - Matriz de erros da classificação com o algoritmo C5.0, sem a seleção de atributos. Kappa=0.8559..... | 70 |
| Tabela 11 - Matriz de erros da classificação com o algoritmo C5.0, após a seleção dos atributos (média das quatro bandas, razão das quatro bandas, máximo da banda IVP e verde, mínimo da banda vermelho e perímetro). Kappa=0.8638. | 70 |
| Tabela 12 - Matriz de erros da classificação com o algoritmo SVM, sem seleção dos atributos. Kappa=0.8005..... | 70 |
| Tabela 13 - Matriz de erros da classificação com o algoritmo SVM, com seleção dos atributos (média das quatro bandas, razão das quatro bandas e máximo da banda IVP). Kappa=0.8235..... | 71 |
| Tabela 14 - Matriz de erros da classificação com o algoritmo RF, sem seleção dos atributos. Kappa=0.8368 | 71 |
| Tabela 15 - Matriz de erros da classificação com o algoritmo RF, com seleção dos atributos (razão das quatro bandas, média das bandas IVP e vermelho, moda da banda vermelho e brilho). Kappa=0.9113 | 71 |
| Tabela 16 - Matriz de erros da classificação ECHO. Kappa=0.8311 | 72 |

| | |
|---|----|
| Tabela 17 - Matriz de erros da classificação com o algoritmo ECHO depois de filtrar a imagem com o filtro Kuwahara. Kappa=0.9553 | 72 |
| Tabela 18 - Resultado da classificação pelo algoritmo da máxima verossimilhança. Kappa 0.9380 | 73 |
| Tabela 19 - Matriz de erros da classificação pelo algoritmo da máxima verossimilhança depois de usado o filtro Kuwahara. Kappa=0.9542 | 73 |

LISTA DE SIGLAS

AD – Árvores de decisão

AG – Algoritmo genético

B&S – Algoritmo Baatz e Schäpe (2000)

BINS – Algoritmo Bins (1996)

ECHO – *Extraction and Classification of Homogeneous Objects*

FK – Filtro Kuwahara

GI – Método de seleção de atributos por ganho de informação

INPE – Instituto Nacional de Pesquisas Espaciais

IVP – Infravermelho próximo

RBSB – *Reference Bounded Segments Booster*

RF – *Random Forests*

SVM – *Support Vector Machines*

χ^2 – Qui-quadrado

1 INTRODUÇÃO

O mapeamento da vegetação urbana geralmente tem se restringido a áreas de valor de conservação conhecido ou de potencial, como parques, reservas, matas ciliares, e em alguns estudos a própria vegetação urbana total para fins de planejamento urbano (MATHIEU; FREEMAN; ARYAL, 2007).

Os quintais privados ou domésticos têm características muito distintas das áreas públicas, pois por ser de domínio privado variam muito conforme as necessidades e recursos de cada proprietário. Além disso, a área dos quintais pode representar uma grande proporção da área urbana total. Por isso, o mapeamento dessas áreas, além representar boa parte do uso do solo urbano, pode apresentar uma ferramenta importante para análise da ecologia urbana devido à grande variabilidade de hábitat. Conhecer qual a contribuição dos quintais privados para a vegetação total da área urbana pode fornecer bases para o planejamento urbano e para políticas públicas para essas áreas.

Este trabalho objetivou criar uma técnica de simples utilização para a classificação supervisionada de imagens multiespectrais de altíssima resolução obtidas pelo satélite WorldView-2 em áreas urbanas para a identificação das áreas de quintais domésticos.

2 DESENVOLVIMENTO

2.1 Revisão Bibliográfica

2.1.1 Impactos da Urbanização

Apenas recentemente os ecologistas vêm dando importância ao ambiente urbano, antes considerado desprovido de vida silvestre e não merecedor de estudos ecológicos (MATHIEU; FREEMAN; ARYAL, 2007). Porém, sabe-se que o crescimento da área urbana tem sérias implicações na biodiversidade e no funcionamento do ecossistema local, regional e global (GRIMM et al., 2008), o que justifica a relevância do estudo desse ambiente.

A urbanização provoca a fragmentação dos habitats naturais e a criação de novos pelo desmatamento e conseqüente redução das áreas naturais. Devido à intensa fragmentação antrópica, Niemelä (1999a) considera que as áreas urbanas são semelhantes às ilhas em termos ecológicos, dependendo de migração e com grande suscetibilidade à invasão de espécies exóticas. Essa fragmentação causa impactos principalmente para espécies com baixa mobilidade, pois as espécies de alta mobilidade têm maior capacidade de migrar entre os fragmentos (WOOD; PULLIN, 2002).

A urbanização leva a mudanças na paisagem, onde a vegetação é substituída pelas construções, vias e outras infraestruturas causando o aumento regional da temperatura, que caracteriza a formação das ilhas de calor – fenômeno onde as regiões urbanas presenciam temperaturas mais elevadas do que as áreas rurais adjacentes (U.S. ENVIRONMENTAL PROTECTION AGENCY, 2008). Os principais parâmetros que causam as ilhas de calor incluem: a redução da evapotranspiração, devido à substituição da vegetação, e as propriedades térmicas dos materiais usados nas construções e pavimentação, com grande capacidade de absorção da radiação e de armazenar energia térmica (MYRUP, 1969; OKE, 1973; TAHA, 2008). Ackerman (1985) mediu uma diferença de temperatura que alcançou 9,3 °C a mais na área urbana em relação à área rural durante o verão. As ilhas de calor afetam: recursos hídricos, qualidade do ar, saúde humana, biodiversidade e funcionamento do ecossistema (LOMBARDO, 1985). A ilha de calor já é um fenômeno bem documentado por diversos estudos (AKBARI; POMERANTZ; TAHA, 2001; GABOR; JOMBACH, 2009; JENERETTE et al., 2006; OKE, 1973).

A concentração de meios de transporte e indústrias representam duas principais fontes de poluição no ambiente urbano, emitindo poluentes como SO₂, óxidos de nitrogênio, ozônio, particulados, CO, compostos orgânicos voláteis e metais pesados, além do CO₂ que contribui

para o aquecimento global, causando impactos na saúde humana (U.S. ENVIRONMENTAL PROTECTION AGENCY, 2008; GRIMM et al., 2008; MOREIRA, 2010).

Nas áreas urbanas o volume de água que escoar superficialmente é maior devido à menor proporção de superfícies permeáveis (BEDIENT; HUBER; VIEUX, 2008). Essa água que escoar superficialmente costuma ter baixa qualidade e requer certas medidas de tratamento antes de ser adequadamente escoada para os corpos d'água (U.S. ENVIRONMENTAL PROTECTION AGENCY, 2008). Nas áreas de vegetação natural, grande parte da água da chuva retorna pela evapotranspiração (PAULEIT; DUHME, 2000). O estudo de Pauleit e Duhme (2000) mostrou que na área urbana, quanto maior a densidade de área construída menor as taxas de infiltração. A mesma pesquisa demonstrou que 21% da precipitação é escoada superficialmente na área urbana estudada.

O uso do solo determinado pela urbanização causa impactos sobre a biodiversidade, Godefroid e Koedam (2007) demonstram que a biodiversidade urbana está relacionada com a densidade urbana e com o tipo de construção presente. Diversos trabalhos estudam a biodiversidade urbana, pois é um fator importante para a qualidade do ambiente urbano (HOPE et al., 2003; LUCK, 2007; NIEMELÄ, 1999b; ROBERTS; AYRE; WHELAN, 2007; SAVARD; CLERGEAU; MENNECHEZ, 2000).

2.1.2 Quintais Privados como Recurso da Floresta Urbana

Neste trabalho será utilizada a definição de floresta urbana proposta por Miller (2007): “A soma de toda a vegetação arbórea e associada no entorno de estabelecimentos humanos densos, desde pequenas comunidades rurais a regiões metropolitanas” (tradução nossa); e a definição de arborização urbana (*urban forestry* em inglês) como a arte, ciências e a tecnologia de manejo da floresta urbana pelos benefícios estéticos, sociais, econômicos e fisiológicos que ela fornece, conforme discutido por Konijnendijk et al. (2006).

Uma das alternativas para diminuir os impactos causados pelo ambiente urbano é a arborização urbana, inclusive fornecendo serviços importantes para o ambiente urbano (JIM; CHEN, 2009).

A floresta urbana tem importantes funções, dentre elas: aumento da biodiversidade; aumento da área de infiltração e retenção de água da chuva, diminuindo riscos de enchentes; diminuição da amplitude térmica e dos efeitos das ilhas de calor; diminuição da poluição sonora; redução da poluição atmosférica pela retenção de partículas sólidas em suas estruturas e eventual absorção foliar; melhoria estética das cidades; absorção da radiação ultravioleta e

de dióxido de carbono (JIM; CHEN, 2009; MOREIRA, 2010; SHASHUA-BAR et al., 2010; STEWART et al., 2009; TRATALOS et al., 2007; ULRICH, 1986).

Os quintais domésticos, como parte do recurso da floresta urbana, são muito pouco estudados. A maioria dos estudos de quintais aqui considerados fazem parte de um projeto maior denominado “Biodiversity in Urban Gardens in Sheffield” (BUGS) (GASTON et al., 2007; GASTON et al., 2005, 2005a, 2005b; LORAM et al., 2007, 2008; SMITH et al., 2005, 2006a, 2006b, 2010). Os quintais privados apresentam características bem distintas das áreas verdes públicas. Ao contrário das áreas verdes públicas, que necessitam de certa homogeneidade para fins de sistematização do manejo (SILVA FILHO et al., 2002), as áreas de quintal privado são heterogêneas entre si, conforme as necessidades e disponibilidade de recursos do morador (GASTON, 2005a; LORAM; WARREN; GASTON, 2008; LORAM et al., 2007). No estudo de Silva Filho e Bortoleto (2005) foram encontradas 123 espécies na arborização viária de São Pedro, SP, enquanto que em Smith et al. (2006) foram encontradas 1166 espécies de plantas vasculares nos quintais em Sheffield no Reino Unido e em Loram et al. (2008) foram 1056 espécies em cinco cidades do Reino Unido. Deve-se verificar que nos estudos considerados os quintais avaliados se encontram em realidades distintas dos municípios brasileiros diferentes, porém isso pode representar o potencial de utilização dos quintais privados, ainda mais se levar em conta a biodiversidade brasileira.

As áreas dos quintais privados podem representar grande parte da área urbana total. No trabalho de Loram et al. (2007), verificou-se que os quintais em cinco cidades representavam uma proporção da área urbana de 21,8% a 26,8%, e em Mathieu, Freeman e Aryal (2007) a proporção foi de 36%. Mais uma vez, ambos os estudos considerados se situam em locais com realidades bem diferentes do município a ser estudado e dos municípios do estado de São Paulo em geral, o que justifica a necessidade desse tipo de estudo. A verificação da contribuição das áreas privadas em relação às áreas públicas também é um fator pouco estudado.

As áreas de quintais privados poderiam servir como corredores ecológicos e matriz de diversidade genética, podendo inclusive contribuir para a recuperação de espécies ameaçadas de extinção (GODDARD; DOUGILL; BENTON, 2010; ROBERTS; AYRE; WHELAN, 2007).

2.1.3 Mapeamento dos Quintais

Existe uma quantidade considerável de trabalhos que estudam a vegetação urbana por mapeamento através do sensoriamento remoto, no entanto existem poucos que discriminam a vegetação correspondente à área de quintais privados.

O sensoriamento remoto é uma ferramenta para estudo da paisagem através de mapas, imagens e modelos georreferenciados. As imagens são adquiridas por fotos aéreas, videografia, sensores de satélites ou a partir da superfície. Cada forma de obtenção da imagem possui vantagens e desvantagens. Segundo Wendtz (2006) as imagens obtidas por fotos aéreas possuem vantagem de ter alta extensão temporal e acurácia na classificação, porém possui baixa extensão espacial e a sua produção é considerada demorada e cara. As imagens produzidas por satélites possuem alta extensão espacial e temporal e custo inferior. Já a imagem obtida por meio terrestre possui alta acurácia, porém cobre extensões espaciais e temporais restritas e é mais utilizada como verdade terrestre para validar a classificação por outros métodos.

Nos trabalhos do projeto BUGS o mapeamento dos quintais é realizado de forma manual. Apesar de ser uma técnica acurada, consome muito tempo e pode não ser prática para o mapeamento em larga escala. Por isso o desenvolvimento de técnicas de mapeamento automatizadas é importante, pois pode fornecer resultados com boa acurácia e de forma mais rápida. Mathieu, Freeman e Aryal (2007) estimaram que o mapeamento manual da cidade de Dunedin, Nova Zelândia, levaria pelo menos oito meses para um analista experiente, enquanto que pelo método automatizado levaria no máximo dois meses.

Um dos desafios da classificação automatizada de áreas urbanas é a alta heterogeneidade de materiais com diversas propriedades e interações complexas (RIDD, 1995). Enquanto imagens de sensores com baixa resolução espacial agrupam diversos tipos de cobertura em um único pixel, dificultando o uso de técnicas que analisam pixels individuais (PHINN et al., 2002), enquanto que com uma imagem de alta resolução, numerosos pixels com diferentes reflexões espectrais quando agrupadas podem representar um único tipo de uso do solo (WENTZ et al., 2008).

Mathieu, Freeman e Aryal (2007) utilizaram uma metodologia para o mapeamento de quintais na Nova Zelândia utilizando imagens do satélite Ikonos e conseguiram resultados satisfatórios. A metodologia deste estudo foi utilizada como base para o desenvolvimento da técnica para a classificação supervisionada automática das imagens de satélite WorldView-2.

2.1.4 Classificação de Imagens

A análise e estudo das imagens para o mapeamento envolvem a interpretação e classificação dessas imagens. A interpretação é realizada através de uma combinação das propriedades da imagem (tamanho, forma, textura, padrão) e as propriedades espectrais, como a tonalidade e a cor (MATHIEU; ARYAL; CHONG, 2007). A classificação visa a simplificação da imagem em um mapa temático, que aponta somente algumas feições de interesse. A partir desse mapa, é possível analisar o tamanho, a distribuição, o percentual correspondente a cada feição, possibilitando adquirir informações da paisagem para diferentes estudos.

As feições de interesse podem ser definidas *a priori* dependendo dos objetivos do estudo. Por exemplo, se for um estudo para analisar apenas a vegetação presente nos bairros de uma cidade, as classes de interesse poderiam ser definidas como copa de árvores, relvado e o restante. No entanto, algoritmos baseados em estatísticas paramétricas apresentariam bastante problema, pois a classe restante teria uma variância muito grande e, provavelmente não apresentaria distribuição normal. Por isso, é comum que se crie mais classes definidas *a posteriori* dependendo das feições identificadas na imagem, de sua variação, e do desempenho da classificação com as classes utilizadas inicialmente.

As formas de classificação também podem ser agrupadas dependendo do uso ou não de amostras de treinamento como supervisionadas ou não-supervisionadas (LU; WENG, 2007). A classificação supervisionada utiliza amostras de treinamento que são utilizadas pelos algoritmos para criar regras para diferenciar as classes na classificação. Na classificação não-supervisionada, as classes são definidas por regras definidas nos parâmetros de entrada, ou então por análise dos dados, buscando maximizar a diferença entre as classes por alguma medida.

Para se realizar o treinamento é necessário que se verifique a que classes pertencem os diferentes objetos ou pixels identificados na imagem, pois essas amostras deverão corresponder à verdade terrestre. Por isso, as amostras podem verificadas por análise a campo, ou em casos de imagens de resolução suficiente apenas pela interpretação visual. Mais recentemente se tornou possível utilizar, se disponível, a ferramenta do Google StreetView, que permite visualizar fotos de alta resolução com perspectiva de visão a partir das ruas.

Ao se determinar a verdade terrestre, é comum, antes do treinamento se criar a chave de interpretação como em Novack (2009). A chave de interpretação consiste em uma tabela que apresenta uma amostra da imagem representativa para cada classe. Ela se presta como

apoio para o próprio intérprete criar as amostras de treinamento, e como prova para posterior crítica das classes consideradas e possível reavaliação.

Recentemente muitas abordagens para a classificação de imagens têm sido amplamente utilizadas e podem ser agrupadas em diferentes categorias dependendo da forma de se trabalhar os dados (LU; WENG, 2007). Para a interpretação de imagens urbanas dois principais fatores do classificador que influenciam no resultado final são: o tipo de informação utilizada do pixel (classificadores por pixel, subpixel, região, ou objeto orientados) e se a informação espacial (contextual) é ou não utilizada.

A paisagem urbana apresenta alta heterogeneidade e, por isso, necessita de imagens de altíssima resolução para representá-la. Além disso, essas imagens contêm alta variação intraclasse, pois um único objeto pode ser representado por diversos pixels heterogêneos. É comum, por exemplo, um telhado cerâmico conter manchas e possuírem tonalidade mais escura e esverdeada. Os classificadores por região conseguem contornar esse tipo de ruído, pois agrupam os pixels em parcelas, reduzindo a unidade de análise e a heterogeneidade (LU; WENG, 2007; THOMAS; HENDRIX; CONGALTON, 2003).

Outra forma de contornar o problema da heterogeneidade na classificação é utilizar a informação espacial ou o contexto em que os pixels estão inseridos. A inclusão desse tipo de informação não é simples e requer conhecimento para se gerar as regras de classificação e/ou reclassificação (THOMAS; HENDRIX; CONGALTON, 2003). A confusão entre o pavimento da calçada e telhas metálicas é comum, no entanto, sabemos que se o objeto estiver adjacente a uma rua provavelmente é uma calçada. No entanto, incluir todos os conhecimentos que temos a respeito de interpretação de imagens em regras lógicas não é simples e nem sempre são computacionalmente eficientes.

O uso de filtros para se reduzir a heterogeneidade também é comum, no entanto deve-se utilizá-los com muito critério e cautela, para que informações espaciais e espectrais importantes não sejam perdidas após a filtragem. Para isso, pode-se utilizar filtros que preservam a borda como o proposto por Nagao e Matsuyama (1979) e o filtro Kuwahara (HALL, 2007). O filtro Kuwahara (FK) tem a característica de suavizar a imagem e ao mesmo tempo preservar suas bordas. A abordagem deste filtro se baseia na utilização de quatro subjanelas da janela principal do filtro parcialmente sobrepostas, conforme a Figura 1. O valor do pixel central será a média da subjanela que apresentar menor variância. Esse filtro, além de acentuar a suavização, ainda previne que as bordas sejam suavizadas.

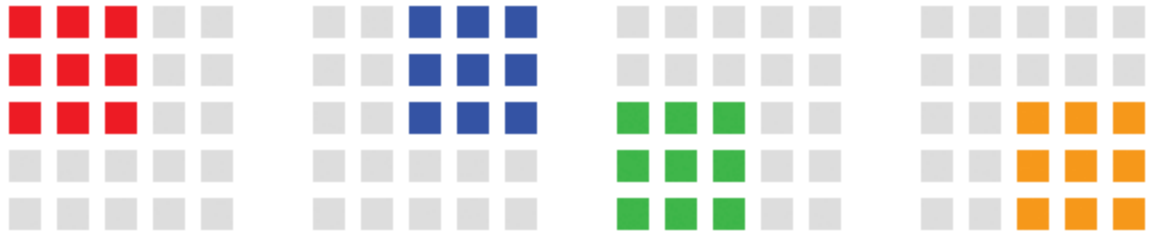


Figura 1 - Esquema ilustrativo das subjanelas utilizadas pelo filtro Kuwahara

Fonte: Hall (2007)

2.1.5 Classificação Orientada a Objetos

Neste trabalho daremos maior atenção à classificação orientada a objetos, que contorna o problema da heterogeneidade por utilizar a abordagem de classificação por regiões, e ainda é capaz de implementar condições lógicas para incluir a informação espacial.

A classificação orientada a objetos é um método que busca mimetizar a forma de interpretar as imagens de um intérprete humano, reconhecendo agrupamentos de pixels como objetos para posterior análise.

As principais vantagens da classificação orientada a objetos são: a estatística mais significativa, a possibilidade do cálculo da textura e outros atributos dos objetos, o uso de dados espaciais como o formato e a topologia dos objetos além da próxima relação entre os objetos reais e os objetos extraídos da imagem; o que amplia o valor da classificação final em relação à classificação por pixels (BENZ et al., 2004).

No entanto, a semelhança entre os objetos vetorizados da imagem e os objetos reais depende da etapa da segmentação, que consiste no agrupamento dos pixels em objetos vetorizados.

2.1.5.1 Segmentação

A segmentação divide a imagem em grupos de células contíguas, correspondendo a elementos da paisagem como casas, ruas, corpos d'água. Portanto, as unidades de análise da classificação objeto-orientada são objetos da paisagem e não células individuais.

A segmentação é uma etapa chave da classificação orientada a objetos, se os seus resultados não forem minimamente satisfatórios comprometerá a classificação final (FEITOSA et al., 2006).

Os segmentos gerados não podem agrupar pixels que pertencem a diferentes classes (subsegmentação), mas também, idealmente, não devem segmentar demais uma área contígua da mesma classe (supersegmentação). Neste caso não se justifica o uso desse método sobre a

classificação por pixels, já que o algoritmo de classificação ainda terá que lidar com o problema da heterogeneidade e o ruído dos pixels nas classes. Segundo Gao et al. (2011), entretanto, o problema da supersegmentação não é um problema significativo, pois o classificador ainda pode alocar os segmentos na classe correta.

2.1.5.1.1 Algoritmos de Segmentação

Para realizar a segmentação existem diversos algoritmos; o tipo mais conhecido é denominado crescimento de regiões, em inglês *region growing*.

Neste trabalho serão considerados e testados dois algoritmos distintos. Um proposto por Baatz e Schäpe (2000), que será denominado B&S e o outro proposto por Bins et al. (1996), que será denominado BINS.

Segundo trabalho apresentado por Meinel e Neubert (2004), os algoritmos de segmentação que apresentam os melhores resultados são os segmentadores por crescimento de regiões B&S e BINS. O primeiro é o mais popular, pois é o algoritmo implementado no sistema eCognition da Definiens, um dos softwares mais populares para este tipo de classificação. O segundo algoritmo é implementado pelo software brasileiro SPRING do Instituto Nacional de Pesquisas Espaciais (CÂMARA et al., 1996).

O procedimento geral dos algoritmos por crescimento de regiões se inicia por segmentos de apenas um pixel a partir do qual novos pixels serão agrupados de acordo com algum limiar de homogeneidade ou heterogeneidade que os segmentos teriam se fossem incluídos.

O algoritmo B&S possui parâmetros de entrada interessantes, que permitem realizar a segmentação não apenas pelo valor digital dos pixels, mas também pela forma que os segmentos possuirão.

A heurística de decisão do algoritmo pode ser definida da seguinte forma: a decisão de agrupar dois diferentes segmentos se dá por meio do cálculo do custo (perda ou ganho de homogeneidade) para mesclar os segmentos. O agrupamento será realizado se esse custo for menor que o parâmetro da escala.

Os parâmetros de entrada são:

1. Escala: limiar para a decisão de agrupamento dos segmentos;
2. Compactação: refere-se ao peso dado à compactação (menor relação entre perímetro/área) no cálculo da homogeneidade;
3. Cor: contrário à compactação, o parâmetro se refere ao peso dos valores digitais dos pixels no cálculo da homogeneidade;

4. Pesos: peso de cada banda no cálculo da homogeneidade.

O algoritmo BINS tem a vantagem de possuir menos parâmetros, portanto a seleção destes é mais fácil.

Este algoritmo funciona da seguinte maneira: dados os segmentos iniciais correspondentes aos pixels da imagem, a decisão de agrupamento é definida a partir do cálculo da distância euclidiana (módulo da diferença das médias) de dois segmentos adjacentes; a decisão de agrupar é aceita se a distância euclidiana for menor que o parâmetro de similaridade definido pelo usuário, dando preferência para o agrupamento dos vizinhos com menor distância euclidiana.

Os parâmetros de entrada do algoritmo são:

1. Similaridade: limiar de decisão para agrupamento;
2. Área mínima: se o segmento for menor que a área mínima ele será mesclado ao vizinho mais similar.

2.1.5.1.2 Calibração dos Parâmetros da Segmentação

O desempenho da segmentação não apenas depende do algoritmo utilizado para realizar a tarefa. Outro fator tão importante quanto à escolha do algoritmo é a escolha dos parâmetros de entrada adequados do algoritmo.

Para a escolha desses parâmetros apenas recentemente se desenvolveram programas e extensões para programas que realizam a escolha de forma objetiva e mais automatizada (CAZES; COSTA; FEITOSA, 2007; DRĂGUȚ; TIEDE; LEVICK, 2010; GAO et al., 2011).

Antes do desenvolvimento de tais ferramentas, a escolha se dava por métodos de tentativa e erro, seguido de análise visual da imagem. Isso torna o processo de escolha dos parâmetros de segmentação uma etapa extremamente subjetiva, demorada e difícil, ainda mais para algoritmos que requerem muitos parâmetros de entrada como é o caso do algoritmo B&S. No entanto, trabalhos recentes ainda utilizam esse método (GENOVEZ, 2010; MATHIEU; ARYAL; CHONG, 2007; MATHIEU; FREEMAN; ARYAL, 2007).

2.1.5.1.3 Aferindo a Qualidade do Resultado

As ferramentas para calibração automática dos parâmetros de calibração normalmente necessitam de uma função que verifique a qualidade das segmentações que serão geradas, para que se possa escolher objetivamente quais parâmetros resultaram em melhores produtos. Podemos identificar duas principais formas de avaliar a qualidade da segmentação:

1. Baseada na discrepância dos segmentos gerados com polígonos vetoriais de referência (CAZES; COSTA; FEITOSA, 2007; MAXWELL, 2005);
2. Pela homogeneidade interna dos segmentos gerados e heterogeneidade entre os segmentos gerados (DRĂGUȚ; TIEDE; LEVICK, 2010; ESPINDOLA et al., 2006; GAO et al., 2011).

A primeira abordagem requer que o usuário defina polígonos de treinamento, sob o qual alguma função de discrepância entre os segmentos gerados e os polígonos será calculada.

A segunda abordagem usa da premissa de que os segmentos ideais seriam internamente homogêneos, mas heterogêneos entre seus vizinhos.

Mesmo depois de validar a segmentação estatisticamente, é imprescindível ainda aferir a qualidade de forma visual. Conforme os próprios autores do algoritmo de segmentação B&S constataam: “A strong and experienced source for evaluation of segmentation techniques is the human eye. [...] No segmentation result - even if quantitatively proofed - will convince, if it does not satisfy the human eye”.

2.1.5.1.4 Algoritmos de Otimização

Após aferir a qualidade do resultado, é necessário que exista algum algoritmo de otimização que faça a inferência de quais parâmetros de entrada poderiam gerar melhores resultados. Para aplicações com baixo custo computacional, os novos parâmetros podem ser definidos por métodos estocásticos, no entanto, a segmentação requer alto custo computacional. Por isso, escolhem-se ou se desenvolvem algoritmos que fazem essa inferência de maneira direcionada.

Um desses métodos de inferência é realizado pelos algoritmos evolucionários ou evolutivos, particularmente trataremos aqui do algoritmo genético (CAZES; COSTA; FEITOSA, 2007; NOVACK, 2009). Esse algoritmo se baseia em teorias de genética de populações, onde os indivíduos com melhor *fitness* ou aptidão têm maior capacidade de sobrevivência e reprodução. Para o caso da segmentação, cada indivíduo tem uma série de genes correspondentes aos parâmetros a serem testados, e a aptidão do indivíduo é calculada por uma função que avalie a qualidade dos resultados gerados por esses parâmetros gerando um valor de aptidão. Os indivíduos com melhor aptidão têm maiores chances de reprodução e permanecem na população. O processo está sistematizado na Figura 2.

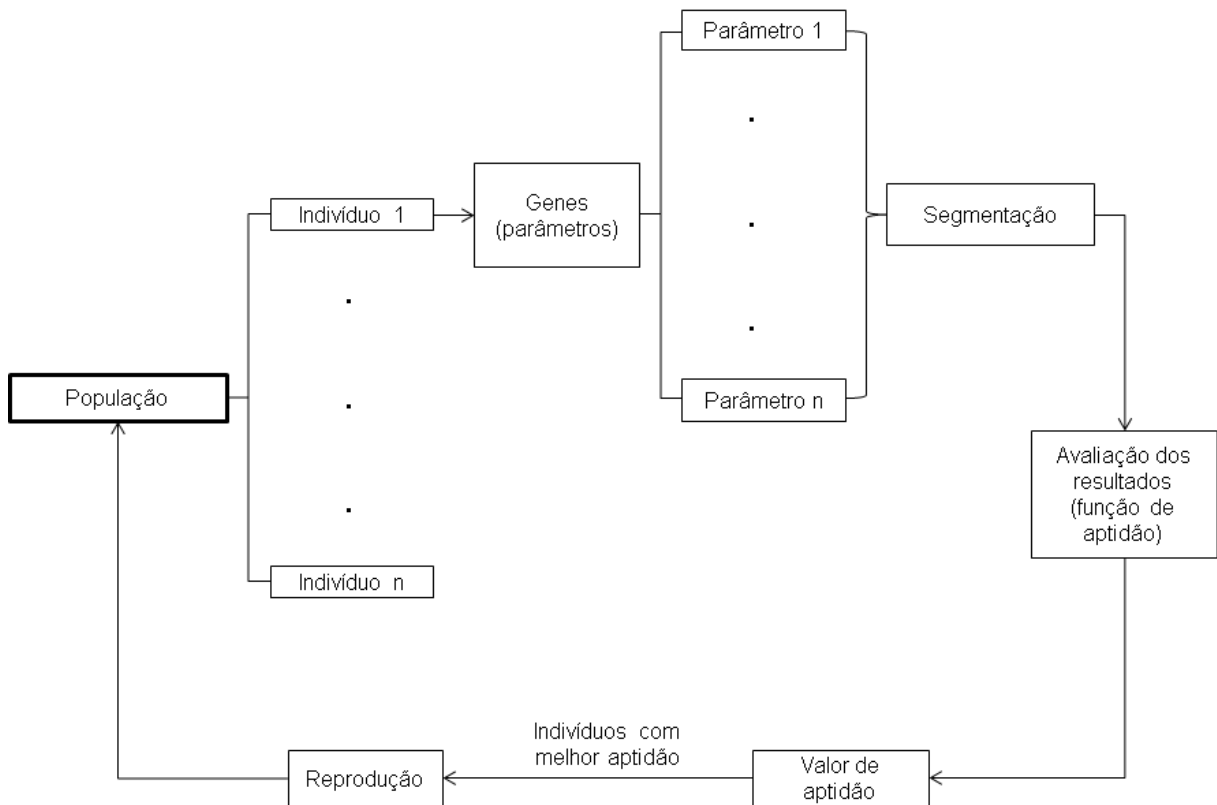


Figura 2 - Esquema ilustrando o funcionamento do algoritmo evolutivo

A premissa deste tipo de algoritmo é que uma combinação entre os parâmetros (genes) que geraram bons resultados anteriormente possam gerar resultados ainda melhores. A combinação dos genes pode ser programada de diversas formas, seja simulando a reprodução sexuada de um organismo mendeliano com os genes organizados em cromossomos e sujeitos à crossing-over e mutação aleatória, ou por simples média entre os genes dos pais.

Outra estratégia de inferência dos melhores parâmetros, utilizada por Maxwell (2005), é a utilização da inferência difusa (em inglês *fuzzy inference*). A inferência difusa não é um algoritmo de simples implementação como o genético. O método proposto define diversas regras e variáveis difusas para inferir quais os parâmetros que serão utilizados.

As regras para avaliação das variáveis difusas são baseadas no conhecimento do desenvolvedor, e a inferência pode simular as decisões que um intérprete humano faria. No caso da segmentação pelo algoritmo B&S, para um intérprete humano um resultado supersegmentado automaticamente faria com que o usuário escolhesse parâmetros de escala maior dependendo do grau de supersegmentação. Essas regras são definidas no sistema de inferência, que calcula ao final os novos parâmetros.

Uma terceira abordagem foi utilizada por Zhang, Feng e Jiang (2010), que utilizam uma técnica em que se verifica o tamanho do maior segmento gerado em relação ao valor do parâmetro de similaridade. Neste, assume-se a premissa de que o valor de similaridade ideal é aquele a partir do qual um incremento não resulte em um incremento no tamanho do maior segmento gerado. No entanto, o artigo não deixa claro que algoritmo é usado, referindo-se ao algoritmo de B&S, mas apenas se refere ao parâmetro de escala, sem comentar como foram definidos os demais parâmetros do algoritmo.

2.1.5.2 Extração dos Atributos

Para realizar a classificação dos segmentos, se faz necessária a extração de alguns atributos desses objetos (em inglês *feature extraction*), que servirão de base para o algoritmo classificador.

Na classificação orientada a objetos é possível utilizar fatores complexos como a relação espacial entre os objetos (contiguidade) e distância entre os objetos além de inúmeros atributos possíveis de se extrair por operações matemáticas de agregação dos valores dos pixels, diferentemente da classificação por pixels onde os únicos atributos para análise são os próprios valores dos pixels. Esses diversos atributos podem ser classificados como: espectrais ou estatísticos dos próprios valores digitais dos pixels, geométricos da forma dos segmentos, texturais, e contextuais (GENOVEZ, 2010).

Os atributos texturais utilizam também os valores dos pixels, mais especificamente os níveis de cinza. Para se analisar matematicamente a textura presente na imagem usam-se diversas estratégias buscando padrões de repetição, detecção de bordas, entre outras. Os valores para textura normalmente são calculados por alguma abordagem estatística, como a matriz de co-ocorrência e aplicação de filtros (HARALICK, 1979).

Os atributos contextuais podem ser definidos com base na relação dos segmentos com alguma referência, ou com seus próprios vizinhos. Por exemplo, a distância entre o segmento e um ponto de interesse, ou o número de segmentos vizinhos.

A etapa de extração de atributos é de extrema importância, pois se escolhido os atributos a se extrair incorretamente, por exemplo, por serem redundantes e fornecerem poucas informações para diferenciar os segmentos entre si, certamente o algoritmo classificador, por melhor desempenho que tiver, não conseguirá definir as classes corretamente.

2.1.5.3 Classificação

A etapa final da análise de imagem orientada a objetos (OBIA – do inglês *object based image analysis*) é a classificação ou rotulação dos segmentos dentro das classes de interesse.

Para executar esse tipo de classificação, pode-se utilizar qualquer algoritmo classificador capaz de realizar uma classificação multivariada. Em face da complexidade, na maioria dos casos, das múltiplas variáveis e do número de classes, utilizam-se os classificadores supervisionados.

Os classificadores supervisionados necessitam de amostras de treinamento que devem ser amostradas de forma criteriosa, dependendo das necessidades do estudo. Com a finalidade de facilitar a aquisição, pode-se optar por métodos de amostragem sistematizada e estratificada ao longo das imagens e direcionando as amostras dentro das classes de interesse. Conforme Congalton (2001): “[...] experience as well as the multinomial equation show that approximately 50 samples (30 as an absolute minimum) per map class are required to adequately populate an error matrix”, recomenda-se um esforço amostral de aproximadamente 50 amostras.

Outra forma de classificação que vem sendo utilizada é a classificação por árvore de decisão, onde as regras são definidas por conhecimento do usuário a respeito dos comportamentos espectrais das classes e ferramentas de análise exploratória dos dados.

O fato de a imagem estar simplificada, com os segmentos desenhados, facilita e acelera muito a classificação manual por fointerpretação que pode ser uma alternativa de classificação. Para imagens urbanas de altíssima resolução e em grandes extensões, sem a segmentação, essa forma de classificar é praticamente inviável.

Devido ao efeito sumarização dos dados após a segmentação e da complexidade e diversidade de atributos possíveis de se extrair, é improvável que possamos assumir que os dados tenham distribuição normal. Por esta razão, normalmente se utilizam classificadores não paramétricos como as Árvores de Decisão (AD) (NOVACK, 2009; THOMAS; HENDRIX; CONGALTON, 2003) e a máquina de suporte a vetores (SVM – *Support Vector Machine*) (HEUMANN, 2011; HUANG; ZHANG, 2013).

As AD correspondem a um grupo de algoritmos que particiona os dados segundo regras de decisão.

Ao final, obtém-se uma estrutura hierárquica de regras em forma de limiarização de atributos, que pode ser aplicada sobre a base de dados completa para a realização da tarefa de classificação. Após a geração da árvore, pode-se “podá-la”,

ou seja, uma ou mais regras podem ser descartadas sem que se mude o grupo de amostras nos nós-folha. (NOVACK, 2009)

Foody e Marthur (2004) em experimento comparativo de algoritmos não-paramétricos verificou que o algoritmo SVM apresenta melhores resultados.

O SVM é um algoritmo de aprendizado de máquina, baseado em aprendizagem estatística. Foi proposto por inicialmente por Cortes e Vapnik (1995). O algoritmo busca otimizar um hiperplano, que é a superfície capaz de separar da melhor forma um conjunto de dados baseado nos vetores de suporte, que são as amostras de diferentes classes que estão mais próximas uma das outras. No trabalho inicial de Cortes e Vapnik (1995) já se demonstrou que o algoritmo superava os resultados de algoritmos de Árvores de Decisão CART e C4.5 e algumas redes neurais criadas por outros autores especialmente para o conjunto de dados usados no trabalho; portanto, demonstrando o seu grande potencial.

Outro algoritmo de potencial para uso na classificação é o *Random Forests* (RF), proposto por Breiman (2001). Segundo o autor, o algoritmo se baseia na ideia de gerar um conjunto aleatório de classificadores estruturados em AD. O objeto pertencerá à classe em que este foi classificado pelo maior número de árvores do conjunto. A geração das árvores aleatórias pode ser realizada por várias estratégias. A árvore pode ser construída baseada em um subconjunto das amostras aleatoriamente selecionadas, ou então, em vez de os nós das árvores serem definidos sempre pela melhor separação, ela pode ser definida aleatoriamente dentre um conjunto dos melhores possíveis nós.

2.1.6 Classificador Misto

Esta seção foi desenvolvida em especial para tratar de um algoritmo desenvolvido em 1980, mas que até 2002 fora implementado e permanece pouco testado pela comunidade científica. Foram encontrados poucos trabalhos (ERSOY, 2007; MORAN, 2010) que utilizaram o algoritmo e, ambos parecem não ter realizado testes aprofundados com calibração dos parâmetros ou utilizou mais variáveis distintas para outros algoritmos.

O algoritmo ECHO, acrônimo de *Extraction and Classification of Homogeneous Objects* (KETTIG; LANDGREBE, 1976), foi um dos primeiros a serem propostos com a ideia de agrupar espacialmente os pixels vizinhos em objetos homogêneos (HUANG; ZHANG, 2008). A grande vantagem deste algoritmo é a implementação extremamente eficiente e de baixo custo computacional em relação aos outros algoritmos tratados anteriormente. Além de ter apresentado melhores resultados que o classificador da máxima

verossimilhança, é particularmente mais eficiente para dados com ruído por considerar a relação espacial (LANDGREBE, 1980).

O algoritmo é processado em dois níveis de teste. Inicialmente, os pixels são divididos por uma grelha em grupos de células quadradas, de tamanho definido pelo usuário. A célula é validada desde que ela atinja certo nível de homogeneidade, cujo limiar também é definido pelo usuário. Se a célula não atingir o nível de homogeneidade, então ela será classificada pela abordagem por pixel e as demais células passarão pelo segundo nível do teste. O teste de segundo nível analisa as células da vizinhança para decidir se elas serão agrupadas. A decisão de agrupamento se dá por uma abordagem denominada razão de verossimilhança, em que as funções de densidade probabilística das duas células são testadas para hipótese de igualdade. Se forem consideradas iguais elas serão agrupadas. O limiar de decisão para a hipótese também é definida pelo usuário. A descrição mais detalhada do algoritmo pode ser encontrada no trabalho de Kettig e Landgrebe (1976).

O ECHO necessita de três parâmetros principais:

1. Tamanho da célula: o tamanho das células iniciais;
2. Limiar de homogeneidade: limiar de decisão para considerar a célula homogênea:
 - a. Homogênea: passa pelo segundo teste de agrupamento com seus vizinhos;
 - b. Não homogênea: classificação da célula com abordagem por pixel;
3. Limiar de anexação: limiar para aceitar a decisão de agrupar células vizinhas.

2.1.7 Validação da Classificação

Using a single accuracy a single accuracy parameter to summarize an error matrix may not satisfy the objectives of an accuracy assessment. [...] the best course of action is to report the full error matrix along with the sampling design used to collect the reference data. (STEHMAN, 1997)

Conforme constatado por Stehman (1997), não é adequado utilizar apenas um ou poucos parâmetros para analisar a acurácia da classificação, mas se deve fornecer a própria matriz de erros e concordância.

Outra forma muito comum de se realizar a análise quantitativa do resultado é pelo cálculo dos erros de omissão e inclusão para cada classe. O erro de omissão, também denominado de acurácia do produtor, corresponde à probabilidade de uma amostra de teste ser classificada corretamente; o cálculo é efetuado dividindo-se o número de amostras

classificadas corretamente pelo número total de amostras teste da classe. Já o erro de inclusão, também denominado de acurácia do usuário, é a probabilidade de uma amostra de uma determinada classe realmente pertencer àquela classe; seu cálculo é efetuado dividindo-se o número de amostras classificadas corretamente pelo número de amostras que foram classificadas na classe. Esses indicadores permitem a análise de quais classes estão tendo maior peso na redução ou no aumento da acurácia da classificação (STORY; CONGALTON, 1986).

A análise Kappa, outra forma muito comum de se aferir o resultado estatisticamente, consiste no cálculo da concordância não aleatória entre dois classificadores em uma amostra (COHEN, 1960). Considera-se que a concordância aleatória dependeria da frequência de amostras atribuídas para cada classe. A concordância entre os classificadores então pode ser normalizada em relação somente à concordância não aleatória. Matematicamente κ pode ser definida como:

$$\kappa = \frac{f_o - f_c}{N - f_c}$$

Onde,

f_o : somatório da frequência de concordância observada (diagonal principal);

f_c : somatório da frequência de concordância esperada ao acaso;

N : Número total de amostras.

f_c pode ser calculada da seguinte maneira:

$$f_c = \sum \frac{A_i * R_i}{N}$$

Onde,

A_i : Número de amostras de treinamento na classe i ;

R_i : Número de amostras de referência na classe i .

Podem entender o κ simplesmente como a proporção de concordância depois de retirar a concordância aleatória.

2.1.8 Mercado e Softwares

A classificação orientada a objetos, diferente de muitas tecnologias, não é dominada pela língua inglesa, mas sim a alemã. O grande desenvolvedor na área de classificação orientada a objetos que detém os direitos do software comercialmente mais utilizado

(FERNANDES; NUNES; SILVA, 2012; GENOVEZ, 2010; MATHIEU; FREEMAN; ARYAL, 2007; MEINEL; NEUBERT, 2004; NOVACK, 2009; ZHANG; FENG; JIANG, 2010), o eCognition, é a alemã Definiens (BLASCHKE; KUX, 2007).

No entanto, a grande limitação do software é o fato de ser de licença comercial. E embora se destaque neste ramo pela qualidade e consistência de suas análises, sua licença não é barata.

O eCognition pode derivar um grande número de atributos geométricos particulares dos segmentos, e por isso mostra grande vantagem sobre os métodos tradicionais baseados na informação espectral, principalmente por causa de sua classificação integrada que faz uso desses atributos. No entanto, ele também tem altos custos, em termos de preço de licença, requisitos de hardware e alto conhecimento do usuário, e não é acessível para muitas instituições. Por isso pesquisas substanciais têm focado no uso de softwares de acesso gratuito, como o SPRING (GAO et al., 2011, tradução nossa).

Outro software desenvolvido especificamente para a análise orientada a objetos é o InterImage, de desenvolvimento em parceria da Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) e o Instituto Nacional de Pesquisas Espaciais (INPE). O software é de código aberto e utiliza bibliotecas do terraLib – projeto de desenvolvimento de bibliotecas de código aberto do INPE – e se baseia no sistema geoAIDA – biblioteca de código aberto de desenvolvimento pela universidade alemã Leibniz Universität Hannover.

2.2 Descrição Física do Local

De acordo com o IBGE (2012) o município de Rio Claro situa-se entre nas coordenadas geográficas de latitude 22,41° S e longitude 47,56° O e está a 613 m de altitude. O clima, pela classificação de Köppen-Geiger é cwa, ou seja, subtropical úmido com estiagem no inverno, temperatura média, do mês mais quente, superior a 22°C, e a do mês mais frio, inferior a 18°C.

2.3 Material e Métodos

2.3.1 Imagens Utilizadas

Neste trabalho foram utilizadas imagens com as quatro banda padrão (azul, verde, vermelho e infravermelho-próximo) obtidas pelo satélite da DigitalGlobe WorldView-2 em 27 de Julho de 2010, do município de Rio Claro, SP. A imagem utilizada foi adquirida em outro trabalho de pesquisa do mesmo laboratório.

O satélite WorldView-2 obtém imagens com dados até 11-bits e resoluções espaciais de 0,46 m (pancromática) e 1,84 m (multiespectral), mas são reamostradas para serem distribuídas comercialmente a resolução de 0,5 m (pancromática) e 2 m (multiespectral). Sua faixa de aquisição nominal é de 16,4 km. A resolução espectral pode ser verificada na Tabela 1 (UPDIKE; COMP, 2010).

Tabela 1 - Resolução espectral do satélite WorldView-2.

| Banda espectral | Comprimento de onda central – 50% Passa-Banda (nm) | 50% Passa-Banda (nm) | Comprimento de onda central – 5% Passa-Banda (nm) | 5% Passa-Banda (nm) |
|------------------------|---|-----------------------------|--|----------------------------|
| Pancromática | 632,2 | 463,7 - 800,6 | 627,4 | 447,2 - 807,6 |
| Costal | 427,3 | 401,4 - 453,2 | 427,0 | 396,0 - 458,0 |
| Azul | 477,9 | 447,5 - 508,3 | 478,3 | 441,6 - 515,0 |
| Verde | 546,2 | 511,3 - 581,1 | 545,8 | 505,5 - 586,0 |
| Amarelo | 607,8 | 588,5 - 627,0 | 607,7 | 583,6 - 631,7 |
| Vermelho | 658,8 | 629,2 - 688,5 | 658,8 | 624,1 - 693,5 |
| Vermelho Limítrofe | 723,7 | 703,8 - 743,6 | 724,1 | 698,7 - 749,4 |
| IVP 1 | 831,3 | 772,4 - 890,2 | 832,9 | 764,5 - 901,3 |
| IVP 2 | 908,0 | 861,7 - 954,2 | 949,3 | 856,1 - 1042,5 |

Fonte: adaptado de Updike e Comp (2010)

A resolução espacial de 0,5 m da imagem WorldView-2 permite a identificação dos quintais privados bem como a identificação do uso do solo do seu interior, e apresenta menor custo em relação ao GeoEye.

2.3.2 Diagrama do Método Utilizado

Antes de explicar como foram realizadas as análises e processamentos, elaborou-se um diagrama resumo para um panorama geral do método.

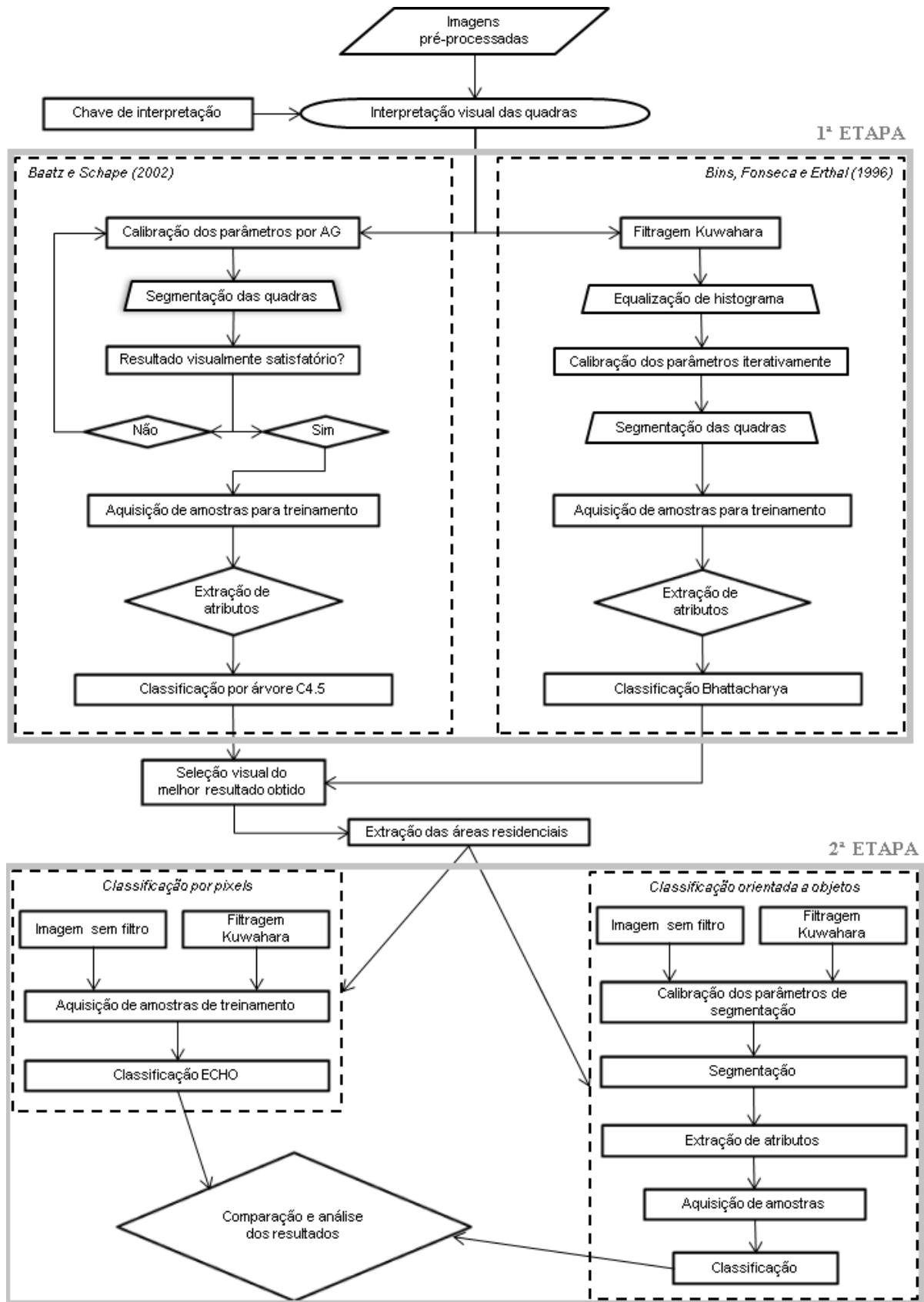


Figura 3 - Diagrama resumo do método proposto

2.3.3 Tratamento das Imagens

A imagem utilizada apresentava-se ortorretificada e fusionada. Estes procedimentos foram efetuados pela própria fornecedora da imagem.

A ortorretificação deve ser realizada para que se eliminem as distorções provocadas pelo sensor em decorrência da planificação da superfície terrestre irregular.

O procedimento da fusão das imagens tem por objetivo combinar a informação espectral das bandas multiespectrais com a informação geométrica mais detalhada da banda pancromática.

Para a classificação de uma única imagem, onde não haverá a comparação das assinaturas espectrais das classes ou utilização delas para classificar outras imagens, não é necessária a correção atmosférica, pois os parâmetros de classificação serão relativos apenas a esta imagem (SONG et al., 2001). A correção atmosférica é mais importante para trabalhos de comparação temporal ou em que dados serão extraídos diretamente das imagens, como em trabalhos de prospecção de minerais e cálculo de parâmetros de solo (SCHOWENGERDT, 1997). Levando-se em consideração estes fatores, não se processou a correção atmosférica.

2.3.4 Softwares

Para a realização deste trabalho, optou-se pelo uso apenas de softwares de licença gratuita ou de código aberto, reduzindo o ônus para a realização do método proposto.

Devido a escolha de métodos avançados, com poucas opções de softwares livres para realizar o tipo de análise requerida, e a existência de trabalhos semelhantes utilizando principalmente o software eCognition, para muitos passos foi necessário o desenvolvimento de alguns aplicativos pelo próprio autor, baseando-se em funções e algoritmos descritos em outros trabalhos.

Devido à simplicidade, objetividade e o desempenho da linguagem para se trabalhar com álgebra linear, além do amplo suporte e existência de pacotes específicos para o trabalho com SIG, utilizou-se a linguagem de programação Python.

Os softwares utilizados foram:

- InterIMAGE 1.34 (COSTA et al., 2007): tentativa de segmentação pelo algoritmo B&S, extração de atributos e classificação por C4.5;
- *Segmentation Parameters Tuner 2.0 (CAZES et al., 2012): Ajuste dos parâmetros de segmentação;
- MultiSpec 3.3 (BIEHL; LANDGREBE, 2002): classificação utilizando o algoritmo ECHO (LANDGREBE, 1980);

- Quantum GIS 1.8.0 (QUANTUM GIS DEVELOPMENT TEAM, 2012): visualização dos dados, criação e edição vetorial das amostras, aplicação do filtro Kuwahara (FK), manipulação e visualização da tabela de atributos;
- R 2.15.2 64-Bits (R DEVELOPMENT CORE TEAM, 2008): Classificação dos resultados da segmentação;
- SPRING 5.2.2 64-Bits (CÂMARA et al., 1996): para a segmentação das imagens com o algoritmo BINS;
- Python 2.7 (VAN ROSSUM, 1989): para o desenvolvimento das ferramentas não existentes gratuitamente;
- Editra 0.7.20 (PRECORD, 2013): edição do código python;
- Microsoft Visual Studio Express 2008 (MICROSOFT CORPORATN, 2007): para compilação do TerraLib (CÂMARA et al., 2007) para exportação das funções de segmentação que implementam os algoritmos B&S e BINS.

*: O software não pôde ser utilizado por apresentar erros e travamentos na análise que serão discutidos posteriormente.

Todos os códigos fonte gerados estão no repositório GitHub (PRESTON-WERNER; WANSTRATH; HYETT, 2013). O GitHub é uma ferramenta criada para compartilhamento de códigos, possibilitando a criação de repositórios para armazenamento e acesso compartilhado, além de espaços para reportar bugs, problemas e solicitar ajuda para os autores. A página do repositório deste trabalho está disponível em <<https://github.com/caiohamamura/>>.

2.3.5 1ª Etapa

Para a primeira etapa de classificação foram testados dois algoritmos de segmentação com procedimentos diferentes para cada um.

2.3.5.1 Definição das Classes

A definição das classes *a priori* foi efetuada com base em interpretação visual das imagens.

Com a finalidade de sistematizar as classes definidas *a priori* e ao mesmo tempo verificar se foram abordados todos ou a maioria dos segmentos em alguma classe, elaborou-se uma chave de interpretação que também serviria para basear o treinamento dos segmentos.

Foram considerados também os próprios resultados da segmentação, verificando de que forma o operador havia segmentado a imagem e como estes poderiam ser classificados.

Caso fosse necessário, após as classificações iniciais e avaliação das mesmas, também poderiam ser criadas ou eliminadas classes.

2.3.5.2 Algoritmo B&S

A calibração dos parâmetros de B&S, conforme já discutido anteriormente não é uma etapa simples, pois existem vários parâmetros de entrada que afetam significativamente os resultados finais, conforme se pôde verificar posteriormente ao se tentar realizar a calibração dos parâmetros de forma iterativa.

Por isso, optou-se por realizar a calibração inicial dos parâmetros utilizando o algoritmo genético (AG) para otimização, de maneira semelhante ao método descrito por Cazes, Costa e Feitosa (2007) em seu trabalho.

Depois da calibração os parâmetros seriam testados e aplicados em recortes maiores para finalmente se efetuar a classificação.

A seguir será discutida de forma mais detalhada os procedimentos adotados.

2.3.5.2.1 Definição dos Segmentos de Referência para a Segmentação

Antes de dar início à segmentação ou ao processo de classificação, é importante que se definam as classes de interesse para o trabalho segundo os objetivos do mesmo.

Para fins de interpretação da imagem considerou-se como quintal qualquer área permeável (com solo exposto, relvado ou copa de árvore) que esteja dentro de uma quadra tipicamente residencial. Para tanto, deve-se reconhecer inicialmente quais são as quadras residenciais.

Considerou-se como quadra residencial aquelas que possuem predominância de telhas cerâmicas e aqueles, em casos de casas de baixa renda, que possuem predominância de telhas cinzas (de amianto).

A segmentação inicial teria a finalidade de buscar e identificar as quadras, ou ao contrário, buscando excluir as vias de asfalto das quadras, para que na etapa de classificação se reconhecessem os segmentos que pertencessem às quadras residenciais. Para que o classificador pudesse identificar os segmentos pertencentes à classe residencial, a escala de segmentação deve ser grande, para que não se delimite objetos isolados como apenas uma construção ou apenas uma árvore, mas sim um agrupamento de árvores e construções que permita analisar se, dentro das construções, existem telhas típicas de residências.

Portanto o desenho dos vetores de referência para a segmentação foram obtidos seguindo os contornos das quadras conforme Figura 4.



Figura 4 - Amostras de referência para calibração da segmentação.

Os polígonos de referência foram desenhados para aferir a qualidade da segmentação e calibração dos parâmetros adequados. Essa vetorização pode ser feita em qualquer ambiente SIG que possua suporte para vetorização em algum formato suportado pelo pacote OGR/GDAL, descrito em Warmerdam (2008). O formato deve ser compatível porque os aplicativos de análise foram desenvolvidos utilizando a biblioteca neste trabalho.

Neste trabalho utilizou-se o software Quantum GIS 1.8.0 por ser de código aberto e de simples utilização para vetorização.

2.3.5.2.2 Calibração dos Parâmetros da Segmentação

Inicialmente realizou-se uma busca por programas que realizassem de forma automática a calibração dos parâmetros, uma vez que vários trabalhos indicavam que haviam sido desenvolvidos (CAZES; COSTA; FEITOSA, 2007; MAXWELL, 2005).

Entretanto, o único programa para tal finalidade que foi encontrado de distribuição gratuita foi o SPT 2.0. Porém, este não funcionou em três computadores disponíveis em que foi testado. Dois deles com o sistema operacional Windows 7 Home Premium 64Bits, um

AMD Phenom II X6 1055T, 2 GB de memória RAM e outro Intel Quad-Core Q8400, 4 GB de memória RAM e o terceiro com o sistema operacional Windows XP Professional Edition 32 Bits, Intel Pentium 4 2.8 GHz, 1GB de memória RAM.

O SPT teria diferentes funcionalidades, dentre elas: a seleção de feições para referência e posterior comparação com os resultados da segmentação realizada por duas opções de funções diferentes, o *Reference Bounded Segments Booster* e o *Larger Segments Booster*; a segmentação em si pelos algoritmos BINS e B&S; e a visualização do resultado para fazer uma análise exploratória dos resultados. No entanto, a única ferramenta que foi possível utilizar foi a de aquisição de amostras, mas a exportação para o formato shp só funciona, no Windows 7, se o programa for carregado como administrador. As demais ferramentas não funcionaram com nenhum dos algoritmos disponíveis.

Frente a essas dificuldades e à falta de suporte ou local para reportar o problema, a única opção para se realizar esta etapa foi o desenvolvimento de um aplicativo que realizasse esse tipo de tarefa que será mais adiante.

A calibração utilizada para os parâmetros da segmentação foram baseados no método proposto por Feitosa et al. (2006) e também utilizado em Novack (2009), onde o método está mais bem detalhado. Este método propõe o uso dos algoritmos genéticos para a otimização e definição dos parâmetros. O algoritmo requer que exista uma função para verificar a qualidade dos resultados para decidir quais vetores (grupo de parâmetros) serão reproduzidos. Utilizou-se no caso a própria função proposta pelo autor, *Reference Bounded Segments Booster* (RBSB), sendo que esta também foi criada em linguagem Python.

O RBSB é calculado comparando-se os polígonos de referência com o segmento que possui maior área de intersecção com a referência. Para esses pares, calcula-se um índice pela divisão da soma das áreas de não intersecção pela área do polígono de referência. O somatório deste índice, para todas as áreas de referência, dividido pelo número de referências é igual resultado do RBSB.

A execução do algoritmo requer alto custo computacional, levando de uma hora a várias horas para retornar os resultados, dependendo de como as populações são programadas no algoritmo, das imagens, e dos objetos de referência. Por isso, os próprios autores do método utilizam o algoritmo em recortes pequenos da imagem, no trabalho de Feitosa et al. (2006), imagens de 256 x 256 pixels.

Tendo em vista o alto custo computacional, optou-se por trabalhar com 10 recortes da imagem de 1000 x 1000 pixels para rodar o algoritmo. Dessa forma seria possível verificar se

os parâmetros otimizados para os recortes podem ser extrapolados para recortes maiores da imagem.

Ainda que existam evidências de que este tipo de método apresentem bons resultados, é também foi aferida a qualidade pela interpretação visual.

2.3.5.2.3 Segmentação

Inicialmente utilizou-se o programa InterIMAGE para realizar os testes iniciais de segmentação, no entanto, após verificar a necessidade de se trabalhar de forma mais automatizada sem o uso de interface gráfica para visualização, optou-se por utilizar o algoritmo desenvolvido no TerraLib.

O algoritmo de segmentação foi executado por meio da exportação em biblioteca dinâmica (arquivo *.dll) de uma função desenvolvida pelo autor e baseada no exemplo de segmentação do TerraLib. Posteriormente a função foi incorporada em código Python para avaliar e definir os parâmetros de entrada.

Após a calibração dos parâmetros de segmentação, os conjuntos de parâmetros que apresentaram os melhores resultados quantitativos pelo método RBSB e por análise visual foram testados em recortes de imagens maiores para verificar se seria possível realizar a extrapolação dos parâmetros para outros recortes.

A extrapolação foi realizada sobre um conjunto de três imagens de 2000 x 2000 pixels para posterior validação visual dos resultados. Não seria possível realizar a segmentação da área toda, não apenas pelo tempo elevado de processamento, mas também pela própria limitação do algoritmo (importado do TerraLib) na manipulação do uso de memória RAM, que acaba excedendo o máximo da máquina, gerando um erro.

2.3.5.2.4 Treinamento

O treinamento classes foi realizado no software Quantum GIS, por seleção dos polígonos e edição de sua tabela de atributos. Inicialmente foram obtidas 30 amostras para cada classe. Dependendo dos resultados, poderiam se obter mais amostras das classes que apresentaram maior confusão. O número de amostras foi pequeno, devido ao reduzido número de segmentos gerados por este algoritmo.

2.3.5.2.5 Extração dos Atributos dos Segmentos

A princípio se efetuou a extração dos atributos no ambiente para exportação de segmentos do InterIMAGE. Este apresenta uma ferramenta de fácil utilização e permite a

extração de um número muito grande de atributos, além de mostrar excelente velocidade de processamento. Para se tentar realizar o processo de forma que necessite ao mínimo ficar trocando de ambiente e softwares para se realizar o processo, também se programou uma ferramenta que realiza a extração dos atributos automaticamente. Além disso, o único classificador existente para o InterIMAGE é o C4.5, e se desejava testar outros classificadores.

A escolha dos atributos a serem utilizados baseou-se em outros trabalhos que já fizeram classificações semelhantes (NOVACK, 2009).

2.3.5.2.6 Classificação

Para a classificação foram testados três algoritmos não-paramétricos, o SVM, as Árvores de Decisão e o algoritmo RF, que já demonstraram resultar em boa acurácia, embora SVM não venha sendo aplicado à classificação orientada a objeto, devido à própria estrutura de análise do software da Definiens ser por rede semântica, semelhante às Árvores de Decisão.

A classificação foi realizada no software R, utilizando pacotes específicos para esse tipo de análise, o pacote C50 (KUHN et al., 2013) para realizar as Árvores de Decisão, o e1071 (MEYER et al., 2012) para realizar a análise por SVM, e o pacote randomForest (LIAW; WIENER, 2002) para RF.

A definição dos parâmetros de cada algoritmo foi baseada em trabalhos já realizados anteriormente.

2.3.5.3 Bins et al. (1996)

A classificação baseada nos segmentos gerados por este algoritmo foi realizada por método diferente do anterior, já que o conjunto de softwares para utilizar esse algoritmo é outro. Neste caso, o software utilizado foi o SPRING.

2.3.5.4 Filtragem Kuwahara

Antes da segmentação, para fins de teste utilizou-se duas fontes de imagens, uma sem o filtro e outra utilizando o FK.

A filtragem foi processada utilizando o ambiente do Quantum GIS, utilizando uma extensão desenvolvida neste trabalho. O intuito da filtragem é de diminuir o ruído dos pixels, mas preservar as bordas, facilitando o procedimento de segmentação. A seguir pode-se visualizar um exemplo do resultado deste filtro.

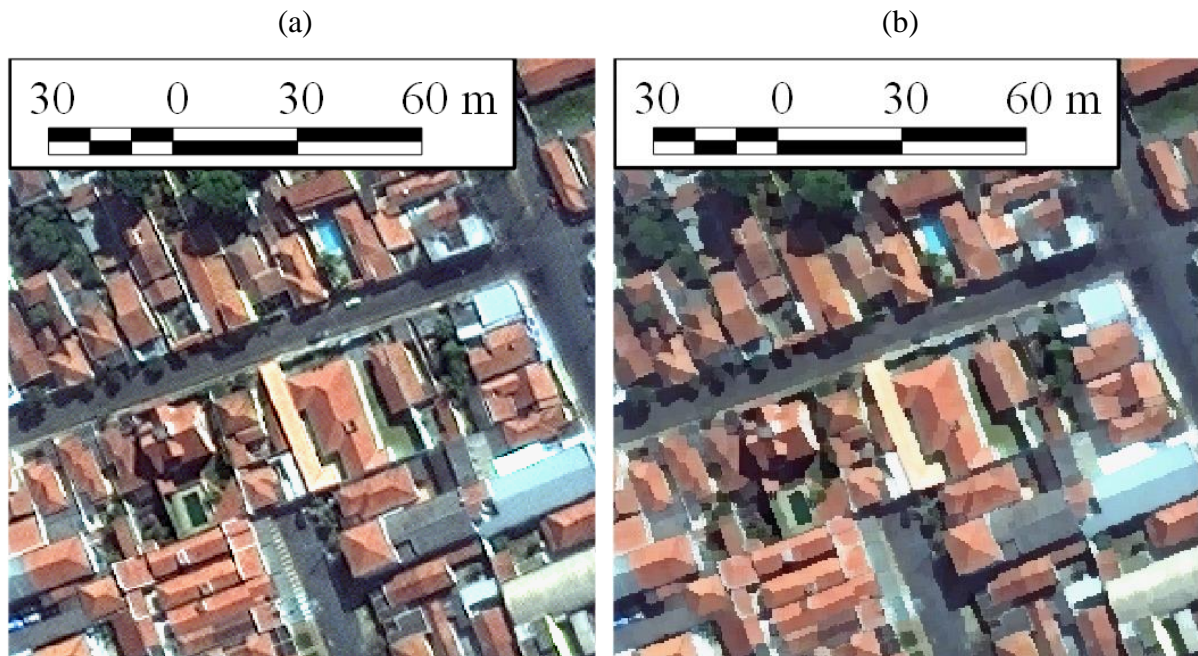


Figura 5 - Exemplo de imagem sem o filtro Kuwahara (a) e com o filtro Kuwahara (b)

2.3.5.4.1 Segmentação

Previamente à segmentação, neste momento, não foi realizada a etapa de calibração automática dos parâmetros. Pois o algoritmo disponível no TerraLib para efetuar a segmentação BINS apresenta processamento excessivamente lento, levando cerca de dez vezes mais tempo para se processar do que a implementação do algoritmo B&S. Isso inviabilizaria o uso da implementação utilizada por algoritmos genéticos, pois levaria mais de 10 horas para se processar.

Neste caso, os parâmetros foram escolhidos por um processo de busca de tentativa e erro. Como o algoritmo possui apenas dois parâmetros de entrada, isso torna a busca por tentativa e erro um processo mais simples.

O processo foi realizado inicialmente em um recorte da imagem de 1000 x 1000 pixels e em seguida testada em outros recortes de mesmo tamanho. Verificando-se que os mesmos parâmetros tiveram bons resultados, então se processou os mesmos parâmetros nas imagens de 2000 x 2000 pixels, as mesmas utilizadas com o algoritmo B&S.

De maneira geral a forma com que se escolhe os parâmetros se baseia numa análise visual entre um resultado supersegmentado, quando objetos contíguos e uniformes são divididos em vários segmentos ou subsegmentado quando os segmentos agrupam vários objetos da paisagem, sendo que o segundo caso leva a resultados finais piores. Se o resultado é supersegmentado, o parâmetro de similaridade é aumentado proporcionalmente ao nível de

supersegmentação. No caso inverso, quando o resultado é subsegmentado, deve-se diminuir o parâmetro de similaridade.

Para definir parâmetro de área mínima, basta analisar visualmente quais os menores objetos da paisagem que se deseja segmentar e, em seguida, calcular o número de pixels que possuem esses menores objetos. Esse é o valor do parâmetro de área mínima. Portanto, apenas se deve alterar por tentativa e erro o parâmetro de similaridade, o que facilita bastante o processo. Talvez, convergindo em um resultado satisfatório ainda antes de um algoritmo de otimização que não seja bem programado e direcionado.

2.3.5.4.2 Extração de Atributos dos Segmentos

No SPRING, a extração de atributos dos segmentos é feita de forma automática, sem a escolha ou intervenção do usuário para tal. Isso, por um lado, facilita o uso para usuários inexperientes, mas também retira a possibilidade de usuários mais experientes de se fazer análises multivariadas mais criteriosas para escolher os atributos mais relevantes.

2.3.5.4.3 Treinamento

A seleção de amostras de treinamento foi realizada no próprio ambiente do SPRING, utilizando a aquisição por regiões. Como a unidade de análise de uma análise orientada a objeto são os próprios objetos ou segmentos, o treinamento foi composto de um número de amostras de pelo menos 50 amostras para cada classe, quando possível.

2.3.5.4.4 Classificação

A classificação para ambos os resultados de classificação foram realizados utilizando as bandas verde, vermelho e infravermelho próximo da imagem sem filtragem, pois ao se filtrar perde-se informação espectral da imagem.

O único algoritmo para classificação supervisionada baseada em regiões é o baseado na distância Bhattacharyya, descrita pelo matemático Bhattacharyya (1943). O algoritmo calcula as funções de densidade probabilística baseadas nas amostras de treinamento, utilizando-as como base para avaliar as distâncias Bhattacharyya entre as classes (SANTOS; PELUZIO; SAITO, 2010).

2.3.6 Extração das Áreas Residenciais

Os resultados finais dos dois algoritmos foram analisados visualmente. O principal critério seria manter as quadras residenciais íntegras, retirando apenas feições que claramente não fossem residenciais como terrenos baldios, praças, construções industriais e comerciais.

Baseado no melhor resultado de classificação efetuou-se a extração das áreas residenciais para que a análise da vegetação examine apenas essas regiões, como possíveis quintais contribuindo para a vegetação urbana.

A extração foi realizada utilizando uma ferramenta do pacote binário disponível pela biblioteca GDAL (GDAL DEVELOPMENT TEAM, 2012) para se fazer a extração de partes de uma imagem baseada em uma máscara vetorizada e uma função SQL WHERE, que possibilita a extração da imagem apenas em polígonos que satisfaçam a cláusula *where*.

2.3.7 Segunda etapa

Na segunda etapa, fez-se a análise e classificação das áreas residenciais com a finalidade de se analisar a vegetação presente apenas nessas regiões.

Nesta etapa, realizou-se o teste com dois métodos diferentes. Um deles utilizando o algoritmo de segmentação que obteve melhor resultado e outro por classificação com o algoritmo ECHO.

2.3.7.1 Definição das classes

A definição das classes nesta etapa foi realizada da mesma forma como anteriormente, gerando uma nova chave de interpretação.

2.3.7.2 Filtragem Kuwahara

Diferentemente da filtragem apenas para segmentação, neste caso a filtragem foi realizada para facilitar também a decisão de classificação para comparar com os resultados da imagem sem filtro.

2.3.7.3 Classificação ECHO

2.3.7.3.1 Treinamento

O treinamento do classificador ECHO foi realizado no próprio ambiente do MultiSpec que possui uma ferramenta específica para realizar o procedimento. Este classificador requer que sejam extraídas amostras baseadas nos pixels e não em segmentos.

A aquisição de amostras por polígonos foi realizada de forma distribuída ao longo da imagem, de forma direcionada, adquirindo amostras representativas das classes. Foram adquiridos pelo menos 1000 pixels para cada classe. Esse é o tipo de recomendação dada para algoritmos paramétricos (FOODY; MATHUR, 2004).

2.3.7.3.2 Classificação

O algoritmo ECHO necessita de três parâmetros de entrada mais importantes, que foram calibrados por AG de forma automática. Devido a eficiente implementação do algoritmo ECHO, a calibração se faz de forma bem mais rápida, levando menos de 10 minutos com parâmetros de população idênticos aos usados para B&S.

No entanto, verificou-se posteriormente que os parâmetros de entrada estabelecidos por padrão no software apresentaram resultados equivalentes ou superiores, eliminando esta etapa de calibração.

2.3.7.4 Classificação Orientada a Objetos

A classificação orientada a objetos foi realizada pelo algoritmo que apresentou os melhores resultados na primeira etapa e a forma de classificar foi basicamente a mesma, com exceção do treinamento que foi realizado baseada na segunda chave de interpretação gerada.

2.3.7.5 Análise dos resultados

Levando em conta que o uso de poucos parâmetros que resumam a acurácia não é adequado para descrever os erros, as classificações foram avaliadas pela matriz de erro e concordância, pelos os erros de omissão e inclusão e pelo KAPPA.

2.4 Resultados e Discussão

2.4.1 Aplicativos Desenvolvidos

Em praticamente todas as etapas houve a necessidade de se desenvolver algum aplicativo para se completarem as tarefas necessárias. Por eles serão brevemente abordados quanto às suas abordagens e características.

2.4.1.1 Filtro Kuwahara

O FK foi inicialmente executado em ambiente SPRING, no entanto este não apresentava boa velocidade de processamento, por isso a implementação foi realizada em linguagem python, que se mostrou rápida mesmo para imagens grandes.

O código foi desenvolvido utilizando-se as bibliotecas Numpy (HUGUNIN et al., 2013), GDAL e o PyQt4 (RIVERBANK COMPUTING, 2013) para interface gráfica. Após o desenvolvimento do algoritmo, este foi escrito como uma extensão para o Quantum GIS, disponível no repositório oficial do projeto <http://plugins.qgis.org/plugins/kuw_filter/>. Observando-se a quantidade de downloads, mais de 900 até 15 de março de 2013, já se verifica que o mesmo se tornou popular entre os usuários do programa. A extensão está disponível nas versões em inglês e português.

2.4.1.2 Segmentador (C++)

O código do segmentador B&S utilizado, foi implementado pelo TerraLib, sendo que o código desenvolvido neste trabalho foi apenas chamar a função já desenvolvida. Na verdade código é basicamente um interpretador dos parâmetros para chamar a função e para exportá-la para uma biblioteca dinâmica (*.dll), permitindo que se use a função por outros programas.

Depois de exportar a função para a biblioteca dinâmica, elaborou-se um código Python para facilitar a importação em outros códigos da mesma linguagem e também para realizar a vetorização pela biblioteca GDAL, exportando os resultados para o formato ESRI Shapefile.

2.4.1.3 Calibração por AG (APÊNDICE)

O algoritmo desenvolvido para a calibração dos parâmetros foi desenvolvido utilizando a biblioteca pygene (MCNAB, 2012), uma biblioteca para programar algoritmos genéticos. A biblioteca pygene oferece uma série de parâmetros para se criar a população e os genes que farão parte do organismo. Os parâmetros para os genes são:

1. Tipo de dado: o tipo de dado são os típicos de um script python, sendo os mais comuns *FloatGene* para números com parte decimal, *IntGene* para números inteiros e *BitGene* para dados binários;
2. Valor mínimo e máximo: o intervalo de valores possíveis para os genes;
3. Probabilidade de mutação: chance porcentual de mutação do gene na reprodução;
4. Valor de mutação: caso sofra mutação, a mudança de valor.

Os parâmetros de população são:

1. População inicial: número de organismos farão parte da população inicial;
2. Seleção de Organismos: número de organismos a manter na população depois de uma nova geração;
3. Filhos: número de filhos a serem gerados;

4. Incesto: número máximo de pais a manter de uma geração a outra;
5. Imigração: número de organismos aleatórios a adicionar a cada geração.

O desenvolvimento com a biblioteca pygene pode ser dividida em quatro etapas:

1. Definição dos genes ou parâmetros que o organismo possuirá;
2. Definição do organismo e de sua função de aptidão;
3. Definição da população;
4. Geração da população.

A população e gerações geradas são exportadas para arquivo de texto ASCII para posterior análise dos parâmetros.

Na função de aptidão ainda foi utilizado outro código gerado no trabalho para o cálculo do RBSB. O código foi baseado na biblioteca OGR/GDAL para leitura dos polígonos e suas geometrias.

2.4.2 1ª Etapa

2.4.2.1 Segmentação B&S

2.4.2.1.1 Aplicativo AG para Calibração dos Parâmetros

O organismo do AG foi definido com seis genes em seu genoma referentes aos seis parâmetros do B&S, que podem ser visualizados na Tabela 2. Os parâmetros da população podem ser visualizados na Tabela 3.

Tabela 2 - Genes ou parâmetros dos indivíduos definidos no algoritmo genético do aplicativo desenvolvido com seus respectivos parâmetros

| Gene | Tipo de Dado | Valor Mínimo | Valor Máximo | Probabilidade de Mutação | Valor de mutação |
|--------------|--------------|--------------|--------------|--------------------------|------------------|
| Escala | <i>Float</i> | 80 | 200 | 0,4 | 0,05 |
| Compactação | <i>Float</i> | 0,1 | 1 | 0,4 | 0,2 |
| Cor | <i>Float</i> | 0,1 | 1 | 0,4 | 0,2 |
| Peso banda 1 | <i>Float</i> | 0,5 | 1 | 0,4 | 0,2 |
| Peso banda 2 | <i>Float</i> | 0 | 1 | 0,4 | 0,2 |
| Peso banda 3 | <i>Float</i> | 0 | 1 | 0,4 | 0,2 |

Tabela 3 - Parâmetros da população utilizados no algoritmo genético do aplicativo desenvolvido

| População inicial | Seleção de Organismos | Filhos | Incesto | Imigração |
|-------------------|-----------------------|--------|---------|-----------|
| 10 | 4 | 6 | 3 | 0 |

O algoritmo apresentou bons resultados, convergindo, à medida que se passam as gerações, para melhores resultados conforme é possível verificarmos na Tabela 4. O restante desses resultados está disponível no APÊNDICE A.

Tabela 4 - Resultados da calibração com o algoritmo genético nos 10 recortes, mostrando a convergência para melhor resultado de RBSB

| Geração | Escala | Compactação | Cor | Peso das Bandas | | | RBSB |
|-----------|--------|-------------|--------|-----------------|--------|--------|--------|
| | | | | 1 | 2 | 3 | |
| Recorte 1 | | | | | | | |
| 0 | 128 | 0,7606 | 0,4189 | 0,2716 | 0,3477 | 0,5621 | 0,7287 |
| 1 | 139 | 0,8149 | 0,4976 | 0,3748 | 0,4812 | 0,5424 | 0,6580 |
| 2 | 160 | 0,7677 | 0,4497 | 0,3338 | 0,4810 | 0,1785 | 0,6359 |
| 3 | - | - | - | - | - | - | - |
| 4 | - | - | - | - | - | - | - |
| 5 | 139 | 0,7701 | 0,5585 | 0,3037 | 0,0735 | 0,1396 | 0,5879 |
| 6 | - | - | - | - | - | - | - |
| 7 | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | 141 | 0,7772 | 0,5485 | 0,3316 | 0,0307 | 0,3258 | 0,5600 |
| Recorte 2 | | | | | | | |
| 0 | 279 | 0,3695 | 0,5926 | 0,6089 | 0,5867 | 0,3019 | 0,5358 |
| 1 | 280 | 0,3171 | 0,5957 | 0,5949 | 0,5536 | 0,1040 | 0,5320 |
| 2 | 252 | 0,3207 | 0,6039 | 0,8067 | 0,5083 | 0,0764 | 0,4904 |
| 3 | - | - | - | - | - | - | - |
| 4 | 252 | 0,3791 | 0,4580 | 0,8067 | 0,4824 | 0,1554 | 0,4151 |
| 5 | - | - | - | - | - | - | - |
| 6 | 252 | 0,4372 | 0,4835 | 0,8897 | 0,2508 | 0,1554 | 0,3524 |
| 7 | 271 | 0,3485 | 0,4766 | 0,8618 | 0,1839 | 0,0319 | 0,3476 |
| 8 | - | - | - | - | - | - | - |
| 9 | 280 | 0,3659 | 0,4643 | 0,7421 | 0,1792 | 0,1433 | 0,3073 |
| 10 | 244 | 0,4372 | 0,4534 | 0,7663 | 0,3244 | 0,0301 | 0,3006 |

Analisando-se as tabelas é possível verificar que, em quase todos os casos, houve melhora gradual do RBSB, com exceção do recorte 4, onde o melhor resultado convergiu na terceira geração. Como todo o processo é regido pelo acaso nem sempre os resultados se apresentam como o esperado.

Outro fator interessante de se notar é que, devido à convergência do AG acontecer a partir de uma população inicial gerada ao acaso, se rodarmos a calibração várias vezes com sementes diferentes obteremos parâmetros bem distintos, o que demonstra a complexidade do problema e, por isso, o uso do AG e não de algoritmos de otimização lineares.

O tempo de execução do AG foi cerca de 27 minutos para a geração de uma população inicial de oito indivíduos seguido de 10 gerações de seis filhos, num total de 68 indivíduos gerados. A parte do algoritmo que leva mais tempo é a segmentação B&S, pois para cada indivíduo deve-se segmentar a imagem e calcular o RBSB dos segmentos. A implementação RBSB, por outro lado, apresentou desempenho satisfatório, levando cerca de 5 segundos para computar os resultados nos recortes de 2000 x 2000 pixels.

Ao final da execução do programa de calibração, geram-se dois produtos em arquivo ASCII texto, um para leitura visual e outro para leitura pelo algoritmo com os dados de cada indivíduo gerado e o resultado da função de aptidão calculado. Especificando o arquivo de população para máquina é possível continuar convergindo a mesma população gerada anteriormente.

A extrapolação dos parâmetros das melhores segmentações não geraram resultados com valor de RBSB idênticos ao calculado para a imagem em que foi calibrada. Ao contrário, utilizando parâmetros três parâmetros que obtiveram melhores resultados em um dos recortes menores, é possível se verificar que, em alguns casos, os parâmetros que resultaram em pior RBSB para o recorte menor podem resultar em RBSB melhor no recorte maior, vide Tabela 5. Isso demonstra ainda mais a complexidade do comportamento dos parâmetros e também que existem variações consideráveis entre diferentes partes da imagem.

Tabela 5 - Extrapolação dos três melhores resultados de cada recorte de 1000 x 1000 pixels para um dos recortes de 2000 x 2000 pixels

| Recorte | Ordem | RBSB | RBSB Anterior | Escala | Compactação | Cor | Peso IV | Peso Verm | Peso Verde |
|----------------|--------------|-------------|----------------------|---------------|--------------------|------------|----------------|------------------|-------------------|
| 1 | 1 | 0,3739 | 0,5600 | 141 | 0,7772 | 0,5485 | 0,3316 | 0,0307 | 0,3258 |
| 1 | 2 | 0,2882 | 0,5879 | 139 | 0,7701 | 0,5585 | 0,3037 | 0,0735 | 0,1396 |
| 1 | 3 | 0,3214 | 0,6359 | 160 | 0,7677 | 0,4497 | 0,3338 | 0,481 | 0,1785 |
| 2 | 1 | 0,3437 | 0,3524 | 244 | 0,4372 | 0,4534 | 0,7663 | 0,3244 | 0,0301 |
| 2 | 2 | 0,6497 | 0,3073 | 280 | 0,3659 | 0,4643 | 0,7421 | 0,1792 | 0,1433 |
| 2 | 3 | 0,6391 | 0,3476 | 271 | 0,3485 | 0,4766 | 0,8618 | 0,1839 | 0,0319 |
| 3 | 1 | 0,3697 | 0,5522 | 153 | 0,5140 | 0,3149 | 0,8313 | 0,3615 | 0,4525 |
| 3 | 2 | 0,3737 | 0,5961 | 153 | 0,4465 | 0,3917 | 0,7842 | 0,416 | 0,4369 |
| 3 | 3 | 0,3613 | 0,5941 | 153 | 0,4250 | 0,3323 | 0,8312 | 0,3632 | 0,4685 |
| 4 | 1 | 0,3860 | 0,5913 | 204 | 0,7262 | 0,6569 | 0,7743 | 0,1467 | 0,3042 |

| Recorte | Ordem | RBSB | RBSB Anterior | Escala | Compactação | Cor | Peso IV | Peso Verm | Peso Verde |
|---------|-------|--------|---------------|--------|-------------|--------|---------|-----------|------------|
| 4 | 2 | 0,3927 | 0,5933 | 204 | 0,7617 | 0,6314 | 0,6705 | 0,5129 | 0,2953 |
| 4 | 3 | 0,3958 | 0,5972 | 204 | 0,7066 | 0,6173 | 0,8102 | 0,4909 | 0,3042 |
| 5 | 1 | 0,3968 | 0,4491 | 177 | 0,7688 | 0,2994 | 0,8127 | 0,4471 | 0,4602 |
| 5 | 2 | 0,3419 | 0,5237 | 179 | 0,8923 | 0,4033 | 0,7972 | 0,3645 | 0,2439 |
| 5 | 3 | 0,3776 | 0,6357 | 177 | 0,4552 | 0,4068 | 0,8298 | 0,4345 | 0,3664 |
| 6 | 1 | 0,4729 | 0,5281 | 245 | 0,3732 | 0,831 | 0,6226 | 0,4013 | 0,1758 |
| 6 | 2 | 0,5370 | 0,5462 | 245 | 0,7274 | 0,9499 | 0,5776 | 0,3314 | 0,5017 |
| 6 | 3 | 0,5431 | 0,5499 | 245 | 0,6567 | 0,8506 | 0,6739 | 0,2511 | 0,1012 |
| 7 | 1 | 0,7205 | 0,3619 | 280 | 0,8350 | 0,3719 | 0,89 | 0,4466 | 0,1389 |
| 7 | 2 | 0,5373 | 0,4050 | 280 | 0,8166 | 0,3719 | 0,8069 | 0,6311 | 0,3385 |
| 7 | 3 | 0,6432 | 0,4386 | 280 | 0,8058 | 0,3342 | 0,906 | 0,5358 | 0,1389 |
| 8 | 1 | 0,3789 | 0,5625 | 196 | 0,6818 | 0,5051 | 0,6281 | 0,5412 | 0,1558 |
| 8 | 2 | 0,3568 | 0,5643 | 199 | 0,3774 | 0,4998 | 0,7139 | 0,3867 | 0,3785 |
| 8 | 3 | 0,4328 | 0,7062 | 135 | 0,1684 | 0,7978 | 0,7021 | 0,4386 | 0,3349 |
| 9 | 1 | 0,4109 | 0,4592 | 192 | 0,7025 | 0,4447 | 0,8076 | 0,2164 | 0,3047 |
| 9 | 2 | 0,3966 | 0,4957 | 193 | 0,4448 | 0,6861 | 0,8484 | 0,111 | 0,4129 |
| 9 | 3 | 0,4164 | 0,5052 | 213 | 0,3165 | 0,6077 | 0,8173 | 0,7374 | 0,6172 |
| 10 | 1 | 0,4327 | 0,7666 | 196 | 0,3830 | 0,8466 | 0,9645 | 0,0189 | 0,7800 |
| 10 | 2 | 0,4129 | 0,7666 | 208 | 0,3830 | 0,8466 | 0,9645 | 0,0229 | 0,7750 |
| 10 | 3 | 0,4751 | 0,8158 | 96 | 0,6065 | 0,7011 | 0,5715 | 0,4923 | 0,5928 |

A análise gráfica da relação entre os índices RBSB calculado para os recortes e o RBSB na imagem maior pode ser visualizada na Figura 6.

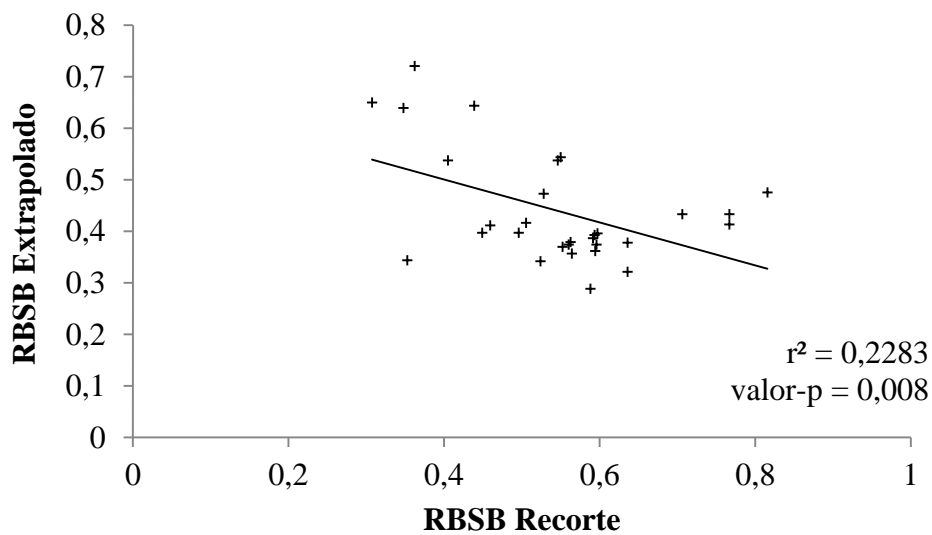
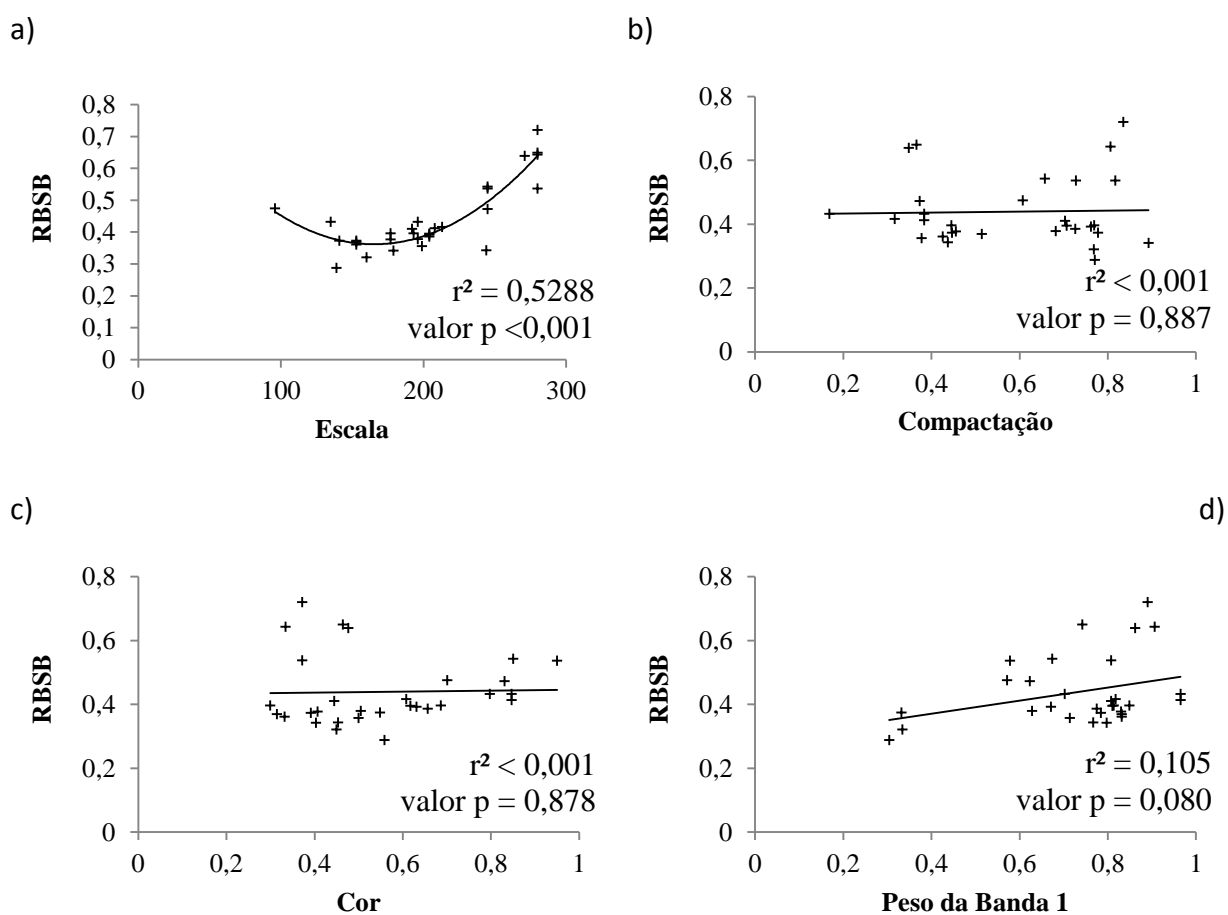


Figura 6 - Gráfico demonstrando a relação entre o RBSB dos recortes de 1000 x 1000 e o RBSB usando os mesmos parâmetros para os recortes de 2000 x 2000

É possível verificar que existe uma fraca correlação altamente significativa, no entanto apresentou baixo coeficiente de correlação linear. Entretanto a correlação foi negativa, isso significaria que os piores resultados calibrados é que retornariam os melhores resultados. Mas isso não faz sentido para esta análise e deve ter ocorrido devido às diferenças na aquisição de amostras entre os recortes e pela própria variação entre as partes da imagem.

Com base ainda na Tabela 5 é possível avaliar a relação entre os parâmetros de entrada e o valor do índice RBSB para saber se existe algum tipo de relação linear entre os parâmetros para que se faça a calibração de forma mais direcionada. Os gráficos de dispersão podem ser visualizados na Figura 7.



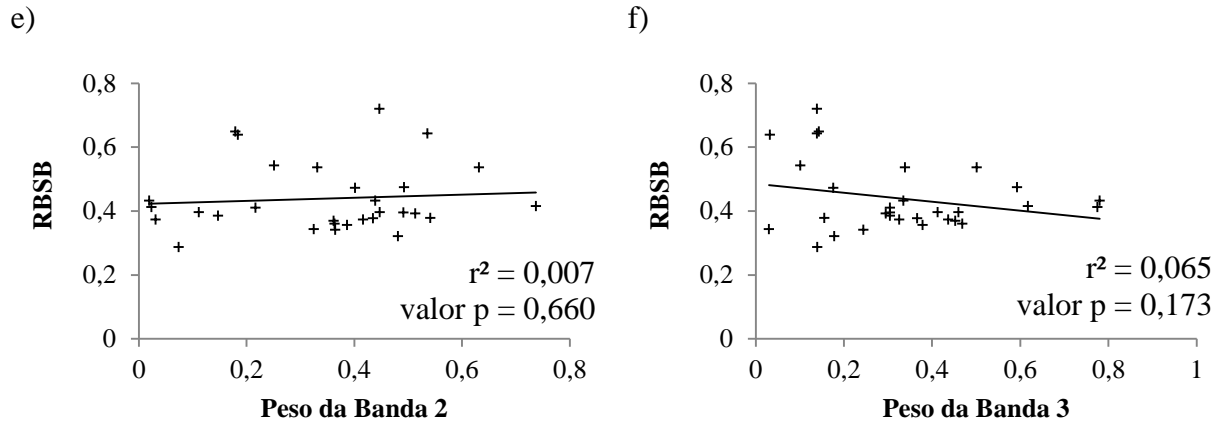


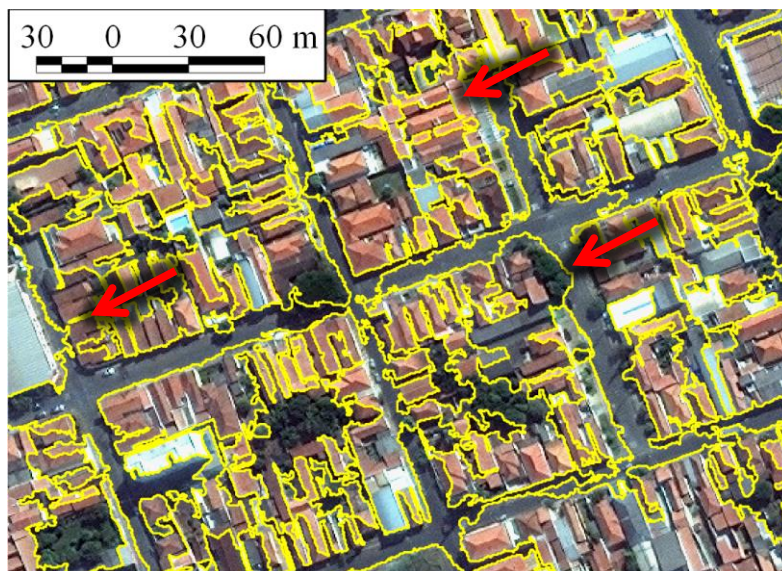
Figura 7 - Gráficos demonstrando a correlação entre os parâmetros de entrada no segmentador e o índice RBSB, utilizando-se os resultados da Tabela 5

É possível constatar que existe uma correlação altamente significativa de RBSB com o parâmetro escala. A correlação polinomial de segundo grau é esperada, pois o valor do parâmetro teria um valor ideal, abaixo e acima do qual o resultado será supersegmentado e subsegmentado respectivamente. Este resultado pode levar à conclusão de que o parâmetro da escala pode ser escolhido com base em otimizadores lineares e fixado em determinado ponto, e apenas os demais parâmetros seriam determinados com algoritmos de otimização não lineares, o que simplificaria o problema.

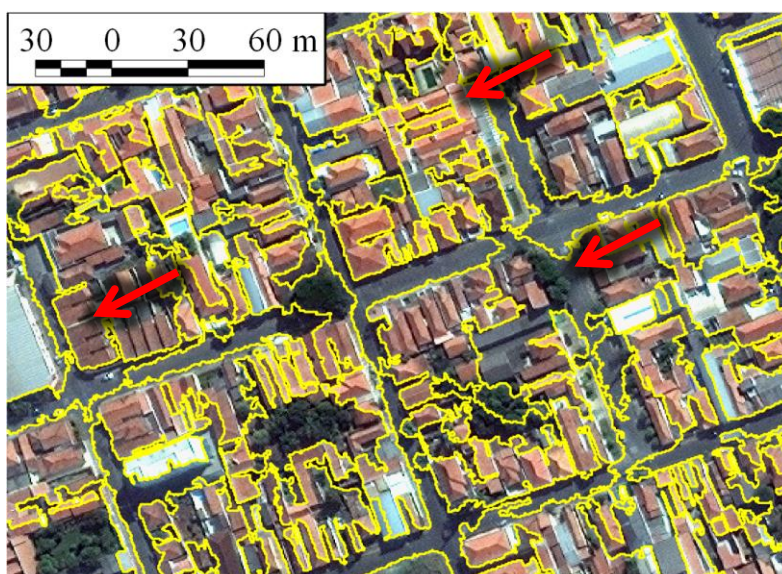
2.4.2.1.2 Resultados das Segmentações

Os resultados de algumas segmentações podem ser visualizados na Figura 8. Analisando-se os resultados é possível perceber que os menores índices de RBSB apresentam resultados mais satisfatórios visualmente, indicando que este índice é adequado para a calibração.

a) RBSB: 1,0021



b) RBSB: 0.7729



c) RBSB: 0,4395

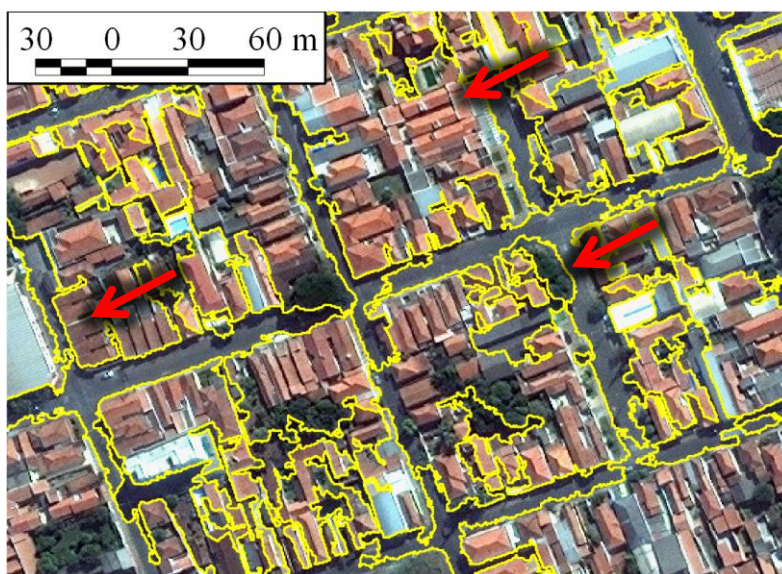


Figura 8 - Resultados das segmentações com diferentes valores de RBSB. Os indicadores vermelhos demonstram locais em que se visualizam as maiores mudanças

Nota-se que o primeiro resultado da segmentação (a), além de supersegmentar as quadras, apresenta segmentos que incluíram asfalto e partes da quadra. O resultado intermediário (b) separou algumas regiões subsegmentadas e diminuiu a supersegmentação. O terceiro (c), e melhor resultado, tanto pelo índice como por análise visual já apresenta menor grau de supersegmentação, com poucos segmentos representando uma quadra, além de separar muitas regiões subsegmentadas.

Por esses resultados é possível constatar que a relação de supersegmentação e subsegmentação nem sempre é antagônica para o algoritmo B&S, uma vez que o terceiro resultado (c) apresentou ao mesmo tempo menor grau de supersegmentação e subsegmentação do que o primeiro resultado (a). Ou seja, incluiu-se menos áreas de asfalto junto a fragmentos de quadras e ao mesmo tempo as quadras possuíam menos segmentos internos.

2.4.2.1.3 Chave de Interpretação

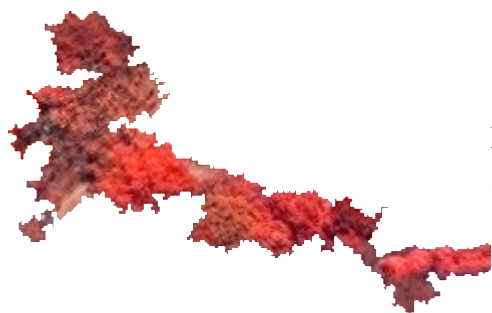
Após análise visual das imagens e dos resultados das segmentações, foi possível determinar uma chave de interpretação para auxiliar a aquisição de amostras de treinamento e teste para aferir os resultados. A chave de interpretação resultante pode ser visualizada na Tabela 6.

Tabela 6 - Chave de interpretação para a primeira etapa de classificação



Asfalto: feições predominantemente lineares e com baixa reflectância

Campo/solo exposto: áreas amplas com certa homogeneidade e tonalidade marrom



Mata: Predominância de copa de árvore, sempre presença de construções



Residencial 1: predominância de telhados cerâmicos, áreas de alta heterogeneidade



Residencial 2: característico de prédios.



Comercial/Industrial: predominância de feições azuladas, de telhas metálicas e concreto, com pouca ou nenhuma vegetação.

2.4.2.1.4 Classificação

Após a segmentação, procedeu-se com a aquisição de atributos das amostras para a classificação. Ao todo foram testados 51 atributos espectrais, de forma e de contexto.

Por análise de trabalhos semelhantes, verificou-se que os atributos mais relevantes para esse tipo de classificação são os espectrais.

A seleção de atributos foi realizada com base no próprio algoritmo para o caso dos algoritmos C5.0 e RF. Para o SVM foram utilizados uma série de filtros, dentre eles o ganho de informação (GI), o RRelirfF, oneR e o qui-quadrado. Foi realizado então um ranking e escolhidos os oito primeiros, como apontado em Novack (2009) como a quantidade média dos melhores resultados. Os atributos em ordem podem ser visualizados na Tabela 7.

Tabela 7 - Ranking dos melhores atributos em ordem decrescente dos rankings para cada algoritmo, sendo os testes respectivamente: qui-quadrado; oneR, RReliefF, ganho de informação, *random forests*, e árvore de decisão C5.0.

| χ^2 | oneR | RReliefF | GI | RF | C5.0 |
|-----------|---------|----------|---------|-----------|---------|
| MEDIA4 | DIFVIZ1 | MODA4 | RAT4 | RAT4 | DIFVIZ3 |
| MEDIA3 | RAT4 | VAR1 | RAT2 | RAT3 | MEDIA3 |
| ENTROPIA2 | DIFVIZ4 | MODA1 | RAT3 | RAT2 | MODA2 |
| VAR2 | MODA1 | RAT3 | RAT1 | RAT1 | MODA3 |
| VAR3 | DIFVIZ2 | RAT4 | DIFVIZ4 | ENTROPIA4 | RAT1 |
| ENTROPIA1 | MEDIA2 | RAT1 | MODA4 | ENTROPIA1 | RAT2 |
| RAT2 | RAT3 | VAR3 | DIFVIZ1 | ENTROPIA3 | RAT3 |
| VAR1 | MODA4 | MODA2 | DIFVIZ2 | VAR3 | RAT4 |
| MODA3 | DIFVIZ3 | VAR2 | MEDIA2 | MEDIA4 | VAR2 |

É possível notar que os atributos melhores ranqueados foram principalmente os atributos espectrais. Os que mais se destacaram foram: a razão (RAT) de contribuição de da banda para o brilho total; a média dos valores dos pixels (MEDIA); a moda dos valores digitais (MODA); a variância dos pixels (VAR) e a entropia (ENTROPIA).

Com menor destaque, mas também bem ranqueados para alguns algoritmos de seleção, enquadram-se os atributos contextuais de distância euclidiana (DIFVIZ) entre o segmento e seus vizinhos para cada banda.

Os três classificadores foram treinados utilizando as mesmas amostras.

Os parâmetros de entrada para cada classificador foram:

- C5.0: Número de iterações 100;
- SVM: gamma=0.038, C=15;
- RF: Número de árvores=500.

Como o intuito da classificação inicial é de apenas separar as áreas residenciais, não faz sentido ficar adquirindo amostras para todas as classes para se fazer uma matriz de erros, já que o interesse é em apenas uma classe, por isso, nesta etapa a qualidade foi aferida visualmente.

A comparação dos resultados dos três algoritmos pode ser visualizada nas imagens a seguir:



Figura 9 - Resultados das classificações. Em amarelo estão os segmentos classificados como residenciais. As setas azuis indicam erros de inclusão e as vermelhas de omissão

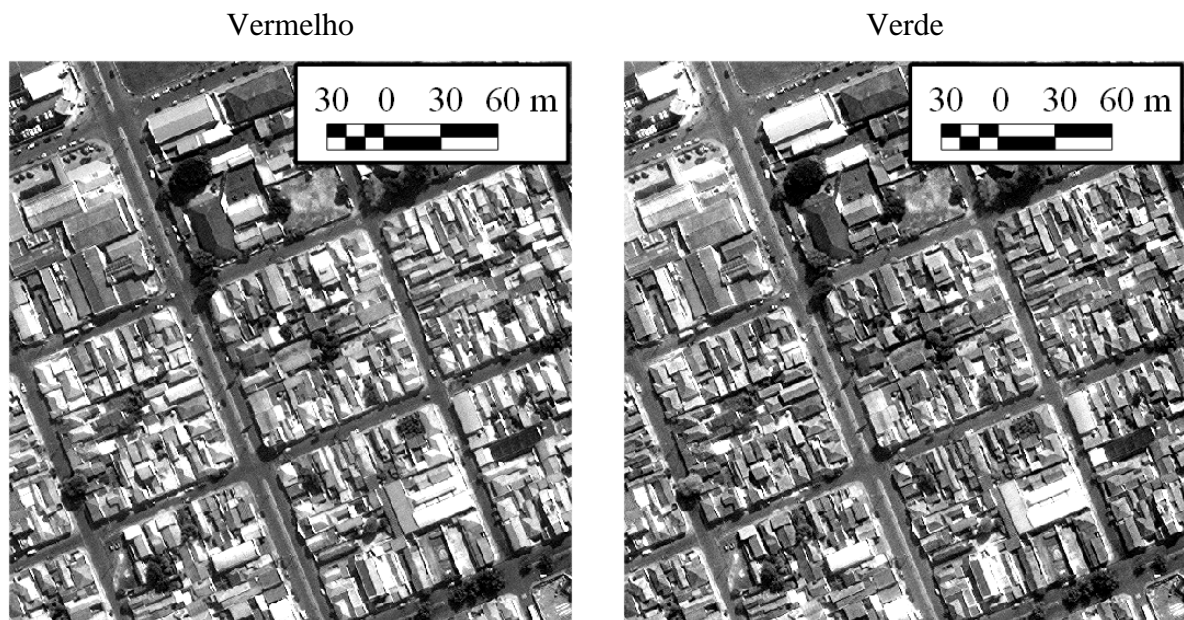
Todos os resultados foram muito parecidos e satisfatórios de maneira geral, sendo que o algoritmo C5.0 incluiu algumas áreas indevidas, confundindo áreas residenciais com solo exposto e deixou de reconhecer algumas áreas que seriam residenciais. O SVM apresentou o pior resultado dentre os três, apresentando muitas áreas de inclusão e outras de omissão. O RF apresentou resultado ligeiramente melhor.

Existem alguns erros inerentes que derivam do processo de segmentação como a inclusão de algumas regiões de asfalto juntamente com partes das quadras que levarão certo viés na análise.

2.4.2.2 Segmentação BINS

Os parâmetros da segmentação foram obtidos por um processo de tentativa e erro. Inicialmente, para que se ressaltassem as diferenças entre os pixels, aplicou-se o contraste por equalização do histograma, que busca igualar a quantidade de pixels para todos os valores, ressaltando o contraste geral.

A segmentação foi realizada apenas baseada na banda do infravermelho próximo, pois esta banda é a que apresenta maior contraste entre as áreas residenciais e as vias de asfalto, conforme se nota na Figura 10. O uso desta banda se justifica, levando-se em conta que a separação entre as ruas e as quadras é o principal intuito da primeira etapa.



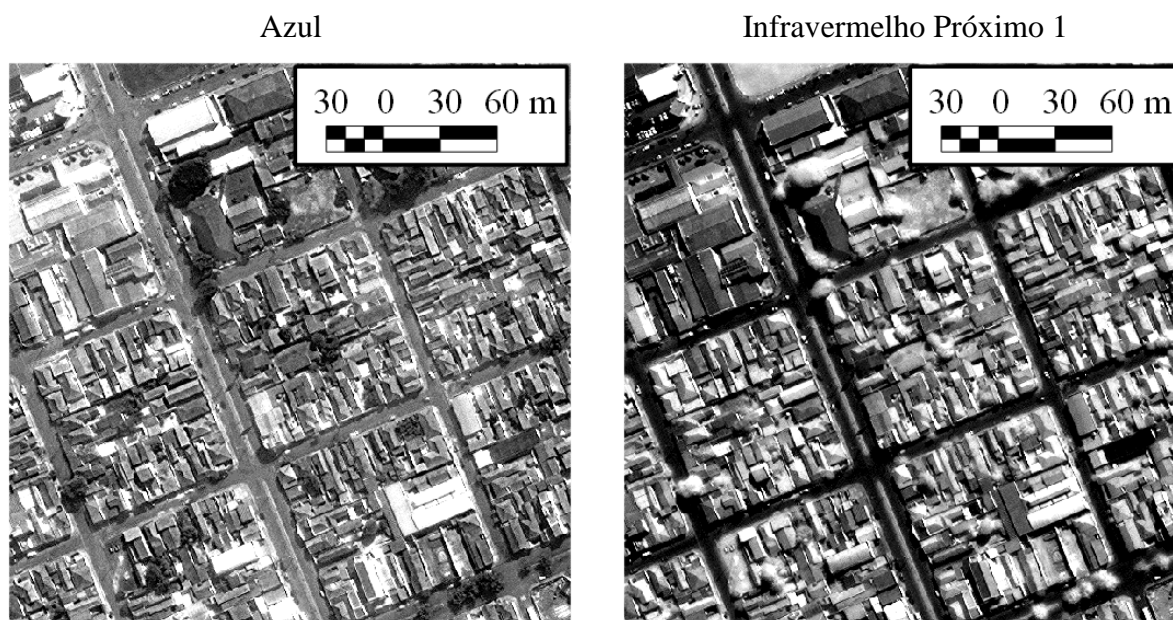


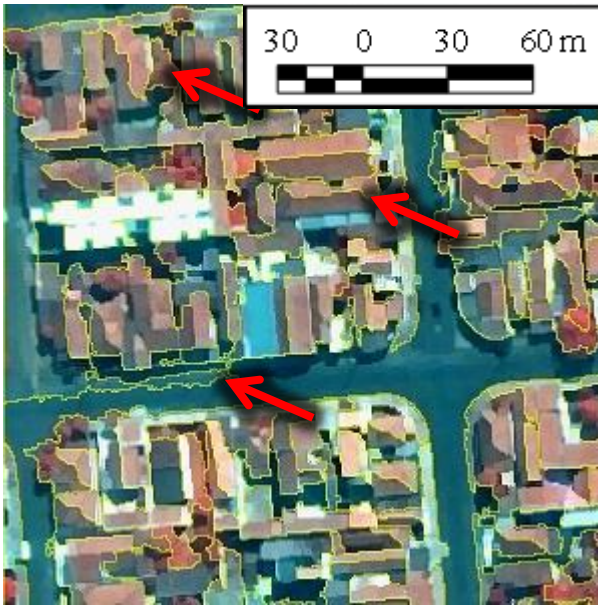
Figura 10 - Bandas com equalização de histograma, demonstrando maior destaque do asfalto na banda do infravermelho próximo 1

Como o principal objetivo desta etapa é o de separar partes residenciais das industriais, comerciais e de asfalto, verificou-se o tamanho das menores feições de interesse, e se verificou que os menores objetos seriam telhados metálicos em quadras residenciais, que provavelmente seriam galpões comerciais ou industriais e alguns terrenos baldios. As medições destes objetos demonstrou que os menores possuíam áreas de 400 pixels, ou 100 m². Portanto se definiu 400 como a área mínima.

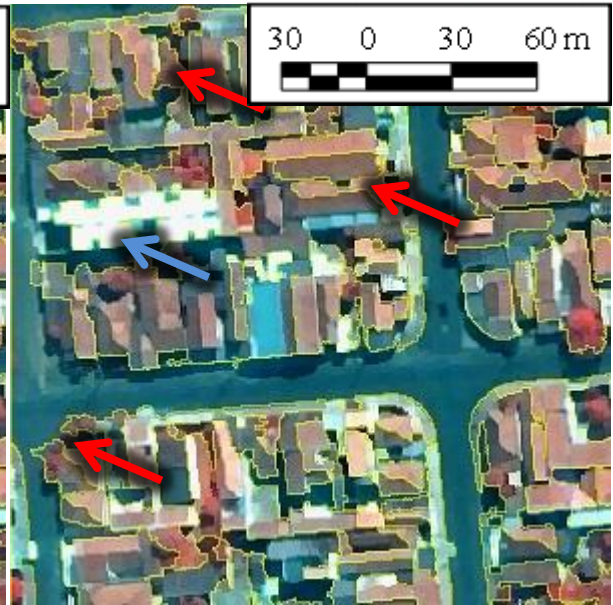
Os resultados variando-se os parâmetros de similaridade podem ser visualizados nas imagens da Figura 11. Verificando-se os resultados, é possível se notar que os valores menores que 35 resultavam em supersegmentação, separando telhas e fragmentos e copas em segmentos individuais, o que não seria interessante para a classificação da primeira etapa para identificar segmentos residenciais.

Por outro lado, valores maiores que 35 resultam em subsegmentação, incluindo muitas áreas de asfalto junto aos segmentos residenciais, o que faria com que a arborização viária fosse incluída como quintais. Portanto, se escolheu como parâmetro a similaridade 35, que apresentou algum grau de subsegmentação, mas não em regiões cruciais para se realizar a classificação.

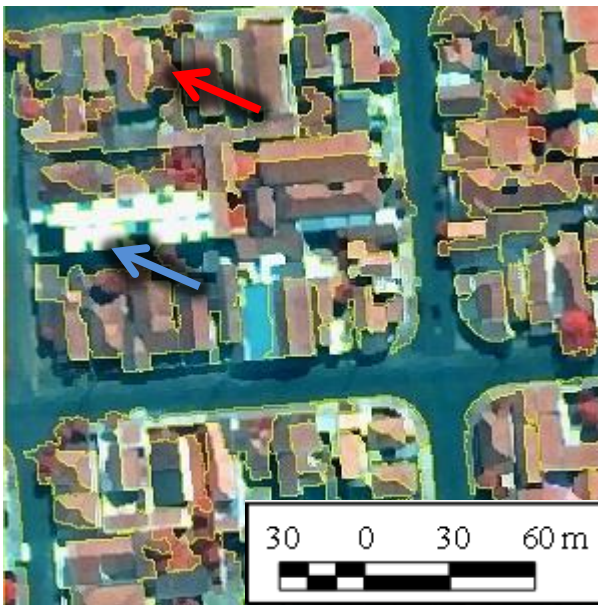
Similaridade: 25; Área mínima: 400



Similaridade: 30; Área mínima: 400



Similaridade: 35; Área mínima: 400



Similaridade: 40; Área mínima: 400

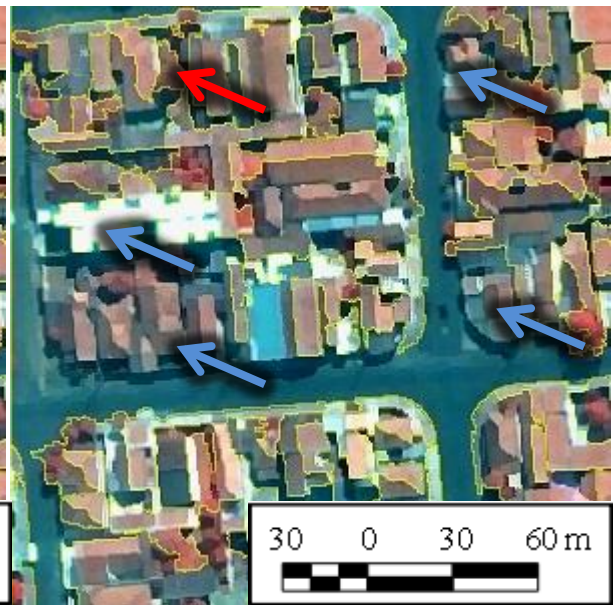


Figura 11 - Resultados de segmentações com o algoritmo de Bins et al. (1996) variando os parâmetros de entrada. As setas em azul representam resultados subsegmentados e as setas vermelhas os resultados supersegmentados

O algoritmo que produziu melhores segmentos pela análise visual qualitativa e o RBSB foi o B&S. O algoritmo BINS supersegmentou as quadras. O FK ajudou neste sentido, diminuindo bastante o efeito de supersegmentação, uma vez que apenas elevar o parâmetro de similaridade fazia com que se produzissem resultados em que muitos segmentos agrupam áreas de asfalto, o que não é desejado para a classificação. Analisando o resultado sem o filtro (Figura 12), é possível notar alto nível de supersegmentação e algum nível de subsegmentação

também, além dos segmentos seguirem linhas menos naturais, dando aspecto muito serrilhado aos objetos.

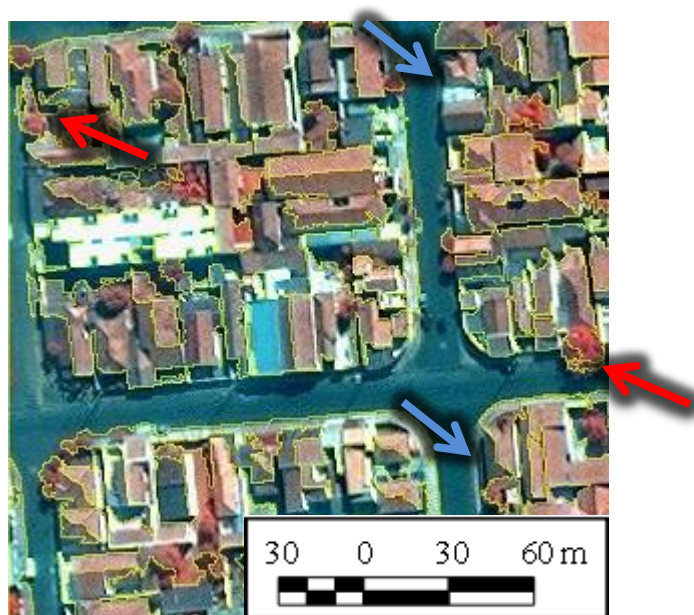


Figura 12 - Segmentação do mesmo trecho de imagem sem o filtro Kuwahara, demonstrando a supersegmentação com as setas vermelhas e subsegmentação pelas setas azuis

2.4.3 2ª Etapa

2.4.3.1 Segmentação B&S

Para a calibração da segunda etapa da segmentação, escolheram-se amostras de copas de árvore para se fazer os vetores, pois é o objeto de maior interesse para este trabalho. As amostras foram obtidas como na Figura 13.

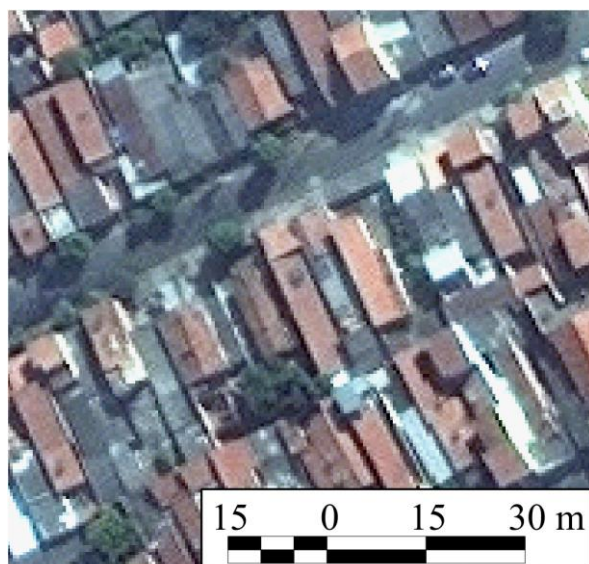


Figura 13 - Amostras de copa de árvore para a calibração do RBSB

Tendo validado a calibração na etapa anterior, desta vez, apenas se processou a segmentação, obtendo resultados de segmentação como na Figura 14.

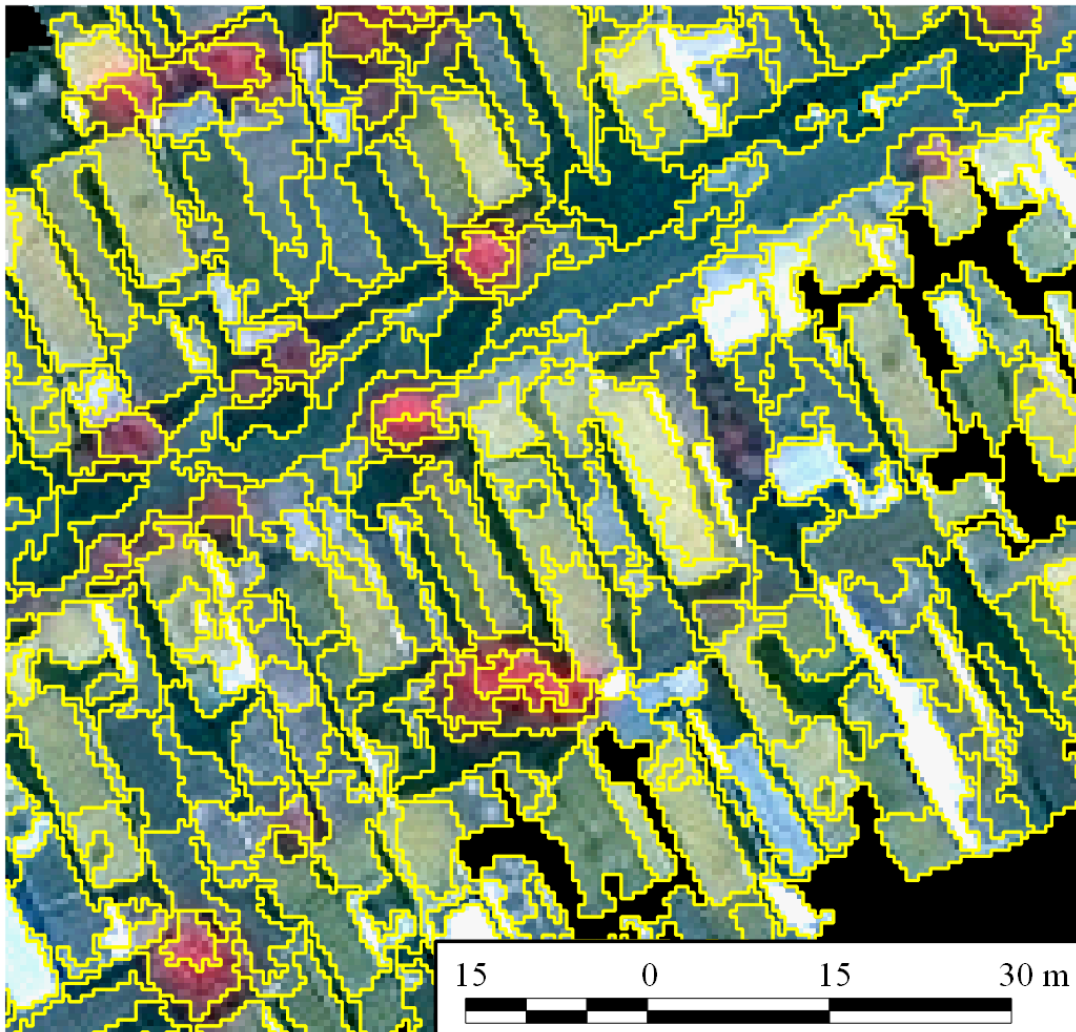


Figura 14 - Exemplo de resultado da segmentação da segunda etapa

O resultado com certeza é supersegmentado, mas isso não é um grande problema para esta etapa, pois agora o classificador ainda pode alocar os segmentos nas classes corretas, o que não seria possível na segmentação da primeira etapa. Por exemplo, se árvores de dentro das áreas residenciais fossem segmentadas individualmente o classificador não teria como saber se essa árvore faz parte de um fragmento de floresta ou se é uma árvore de quintal.

2.4.3.2 Chave de Interpretação

Para auxiliar no treinamento das amostras foi desenvolvida a seguinte chave de classificação:

Tabela 8 - Chave de interpretação usada para a segunda etapa da classificação

| | |
|---|-----------------------|
|  | Asfalto |
|  | Copa de árvore |
|  | Relvado |
|  | Solo exposto |
|  | Sombra |
|  | Telha azulada |
|  | Telha cerâmica clara |
|  | Telha cerâmica escura |
|  | Telha cinza |
|  | Telha clara |

2.4.3.3 Classificação

Após a segmentação e o treinamento das amostras, selecionaram-se os atributos mais relevantes. Os resultados foram semelhantes em relação à primeira etapa, demonstrando que atributos como a média, moda e a razão de importância da banda para o brilho total (RAT) são atributos que interferem mais na separabilidade das classes. No entanto, alguns atributos diferentes apareceram como os valores máximo (MAX) e mínimo (MIN), conforme a Tabela 9.

Tabela 9 - Melhores atributos escolhidos pelos diferentes algoritmos ordenados em ordem decrescente de importância, sendo os testes respectivamente: qui-quadrado; oneR, RReliefF, ganho de informação e *random forests*.

| χ^2 | oneR | RReliefF | GI | RF |
|-----------|--------|----------|--------|--------|
| ENTROPIA1 | RAT3 | MODA1 | RAT1 | RAT1 |
| MEDIA2 | RAT1 | MODA4 | RAT4 | RAT4 |
| DIFVIZ1 | MIN1 | MEDIA1 | MEDIA1 | MEDIA1 |
| AMPL3 | RAT4 | MEDIA4 | MODA1 | RAT2 |
| MODA2 | RAT2 | MODA3 | RAT2 | RAT3 |
| BRILHO | MAX1 | MEDIA3 | MEDIA3 | MODA1 |
| MODA1 | MEDIA4 | MIN4 | MODA3 | MEDIA4 |
| MAX3 | MODA1 | MODA2 | MEDIA4 | MEDIA2 |
| MEDIA1 | MIN2 | RAT4 | MEDIA2 | BRILHO |
| MIN3 | MODA3 | MAX1 | RAT3 | MEDIA3 |

É importante destacar que, o método escolhido para se calcular a importância do atributo apresenta resultados diferentes, assim como o atributo mais importante muda de um resultado de segmentação para outro, sendo necessário sempre que se faça a extração dos atributos, principalmente se os objetivos do estudo forem diferentes ou se as imagens forem provenientes de outros sensores.

Apesar das diferenças ocorridas, pode-se ressaltar que os atributos espectrais de média, moda e RAT, aparecem com frequência indicando um possível caminho para a escolha direta, sem a necessidade da etapa de mineração de dados.

Tabela 10 - Matriz de erros da classificação com o algoritmo C5.0, sem a seleção de atributos. Kappa=0,8559

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|-------------|--------------|-------------|-------------|---------------|-------------|--------------|-------------|--------------|---------------------|
| Telha cc | 5147 | 331 | 0 | 1 | 14 | 0 | 0 | 12 | 1 | 0 | 5506 | 93,48% |
| Telha ce | 0 | 3681 | 128 | 285 | 0 | 133 | 1 | 0 | 20 | 0 | 4248 | 86,65% |
| Asfalto | 0 | 194 | 5949 | 1 | 0 | 510 | 23 | 0 | 27 | 0 | 6704 | 88,74% |
| Solo | 449 | 1837 | 2 | 1500 | 14 | 67 | 0 | 44 | 3 | 0 | 3916 | 38,30% |
| Copa | 0 | 8 | 0 | 0 | 6884 | 0 | 0 | 0 | 46 | 107 | 7045 | 97,71% |
| Telha cz | 0 | 10 | 1752 | 0 | 0 | 5219 | 2033 | 62 | 14 | 0 | 9090 | 57,41% |
| Telha az | 0 | 0 | 57 | 0 | 0 | 35 | 4465 | 72 | 2 | 0 | 4631 | 96,42% |
| Telha cl | 162 | 4 | 0 | 0 | 0 | 1 | 2 | 3260 | 0 | 0 | 3429 | 95,07% |
| Fundo | 0 | 0 | 0 | 3 | 0 | 2 | 0 | 2 | 25747 | 0 | 25754 | 99,97% |
| Relvado | 26 | 5 | 0 | 0 | 223 | 0 | 0 | 0 | 2 | 2769 | 3025 | 91,54% |
| Totais | 5784 | 6070 | 7888 | 1790 | 7135 | 5967 | 6524 | 3452 | 25862 | 2876 | 73348 | |
| Acurácia do Produtor | 88,99% | 60,64% | 75,42% | 83,80% | 96,48% | 87,46% | 68,44% | 94,44% | 99,56% | 96,28% | | 88,10% |

Tabela 11 - Matriz de erros da classificação com o algoritmo C5.0, após a seleção dos atributos (média das quatro bandas, razão das quatro bandas, máximo da banda IVP e verde, mínimo da banda vermelho e perímetro). Kappa=0,8638

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|-------------|--------------|-------------|-------------|---------------|-------------|--------------|-------------|--------------|---------------------|
| Telha cc | 5072 | 406 | 0 | 0 | 13 | 0 | 0 | 12 | 1 | 2 | 5506 | 92,12% |
| Telha ce | 0 | 3547 | 128 | 263 | 158 | 131 | 1 | 0 | 20 | 0 | 4248 | 83,50% |
| Asfalto | 0 | 156 | 5968 | 0 | 0 | 530 | 23 | 0 | 27 | 0 | 6704 | 89,02% |
| Solo | 449 | 1834 | 0 | 1467 | 14 | 69 | 0 | 80 | 3 | 0 | 3916 | 37,46% |
| Copa | 0 | 12 | 0 | 0 | 6861 | 0 | 0 | 0 | 80 | 92 | 7045 | 97,39% |
| Telha cz | 0 | 10 | 1197 | 0 | 0 | 5774 | 2095 | 0 | 14 | 0 | 9090 | 63,52% |
| Telha az | 0 | 0 | 57 | 0 | 0 | 35 | 4521 | 16 | 2 | 0 | 4631 | 97,62% |
| Telha cl | 162 | 1 | 0 | 0 | 0 | 4 | 2 | 3260 | 0 | 0 | 3429 | 95,07% |
| Fundo | 0 | 0 | 0 | 3 | 0 | 2 | 0 | 2 | 25747 | 0 | 25754 | 99,97% |
| Relvado | 26 | 5 | 0 | 0 | 103 | 0 | 0 | 0 | 2 | 2889 | 3025 | 95,50% |
| Totais | 5709 | 5971 | 7350 | 1733 | 7149 | 6545 | 6642 | 3370 | 25896 | 2983 | 73348 | |
| Acurácia do Produtor | 88,84% | 59,40% | 81,20% | 84,65% | 95,97% | 88,22% | 68,07% | 96,74% | 99,42% | 96,85% | | 88,76% |

Tabela 12 - Matriz de erros da classificação com o algoritmo SVM, sem seleção dos atributos. Kappa=0,8005

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|-------------|--------------|-------------|-------------|---------------|-------------|--------------|-------------|--------------|---------------------|
| Telha cc | 5361 | 53 | 0 | 1 | 19 | 0 | 0 | 13 | 59 | 0 | 5506 | 97,37% |
| Telha ce | 122 | 2827 | 117 | 526 | 1 | 121 | 1 | 0 | 375 | 158 | 4248 | 66,55% |
| Asfalto | 0 | 162 | 5370 | 55 | 1 | 409 | 23 | 0 | 684 | 0 | 6704 | 80,10% |
| Solo | 138 | 1743 | 0 | 1475 | 59 | 49 | 0 | 5 | 447 | 0 | 3916 | 37,67% |
| Copa | 4 | 0 | 0 | 4 | 6075 | 0 | 0 | 0 | 585 | 377 | 7045 | 86,23% |
| Telha cz | 0 | 190 | 2713 | 4 | 0 | 5558 | 538 | 0 | 87 | 0 | 9090 | 61,14% |
| Telha az | 0 | 0 | 0 | 0 | 0 | 39 | 4335 | 16 | 241 | 0 | 4631 | 93,61% |
| Telha cl | 352 | 3 | 0 | 0 | 0 | 0 | 5 | 2944 | 125 | 0 | 3429 | 85,86% |
| Fundo | 1217 | 0 | 0 | 3 | 0 | 0 | 2 | 2 | 24530 | 0 | 25754 | 95,25% |
| Relvado | 26 | 7 | 0 | 0 | 120 | 0 | 0 | 0 | 2 | 2870 | 3025 | 94,88% |
| Totais | 7220 | 4985 | 8200 | 2068 | 6275 | 6176 | 4904 | 2980 | 27135 | 3405 | 73348 | |
| Acurácia do Produtor | 74,25% | 56,71% | 65,49% | 71,32% | 96,81% | 89,99% | 88,40% | 98,79% | 90,40% | 84,29% | | 83,64% |

Tabela 13 - Matriz de erros da classificação com o algoritmo SVM, com seleção dos atributos (média das quatro bandas, razão das quatro bandas e máximo da banda IVP).
Kappa=0,8235

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|----------------|---------------|---------------|---------------|---------------|---------------|---------------|--------------|---------------------|
| Telha cc | 5073 | 18 | 0 | 0 | 0 | 0 | 0 | 13 | 1 | 401 | 5506 | 92,14% |
| Telha ce | 14 | 3924 | 130 | 0 | 0 | 1 | 1 | 0 | 20 | 158 | 4248 | 92,37% |
| Asfalto | 0 | 174 | 6343 | 0 | 1 | 136 | 23 | 0 | 27 | 0 | 6704 | 94,62% |
| Solo | 59 | 3073 | 22 | 695 | 50 | 0 | 0 | 5 | 3 | 9 | 3916 | 17,75% |
| Copa | 0 | 4 | 0 | 0 | 6351 | 0 | 0 | 0 | 376 | 314 | 7045 | 90,15% |
| Telha cz | 0 | 35 | 3367 | 0 | 0 | 4662 | 948 | 62 | 16 | 0 | 9090 | 51,29% |
| Telha az | 0 | 0 | 105 | 0 | 0 | 18 | 4505 | 0 | 3 | 0 | 4631 | 97,28% |
| Telha cl | 1 | 4 | 0 | 0 | 0 | 1 | 6 | 3417 | 0 | 0 | 3429 | 99,65% |
| Fundo | 0 | 3 | 2 | 0 | 0 | 0 | 0 | 2 | 25747 | 0 | 25754 | 99,97% |
| Relvado | 26 | 5 | 0 | 0 | 1034 | 0 | 0 | 0 | 2 | 1958 | 3025 | 64,73% |
| Totais | 5173 | 7240 | 9969 | 695 | 7436 | 4818 | 5483 | 3499 | 26195 | 2840 | 73348 | |
| Acurácia do Produtor | 98,07% | 54,20% | 63,63% | 100,00% | 85,41% | 96,76% | 82,16% | 97,66% | 98,29% | 68,94% | | 85,45% |

Tabela 14 - Matriz de erros da classificação com o algoritmo RF, sem seleção dos atributos.
Kappa=0,8368

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|--------------|---------------------|
| Telha cc | 4806 | 320 | 0 | 0 | 14 | 0 | 0 | 12 | 1 | 353 | 5506 | 87,29% |
| Telha ce | 0 | 3679 | 134 | 285 | 0 | 125 | 1 | 4 | 20 | 0 | 4248 | 86,61% |
| Asfalto | 0 | 202 | 5954 | 1 | 0 | 497 | 23 | 0 | 27 | 0 | 6704 | 88,81% |
| Solo | 99 | 2748 | 22 | 986 | 14 | 0 | 0 | 44 | 3 | 0 | 3916 | 25,18% |
| Copa | 0 | 4 | 0 | 4 | 6965 | 0 | 0 | 0 | 72 | 0 | 7045 | 98,86% |
| Telha cz | 0 | 34 | 3094 | 0 | 0 | 4674 | 1274 | 0 | 14 | 0 | 9090 | 51,42% |
| Telha az | 0 | 0 | 14 | 0 | 0 | 39 | 4560 | 16 | 2 | 0 | 4631 | 98,47% |
| Telha cl | 20 | 4 | 0 | 0 | 0 | 1 | 3 | 3401 | 0 | 0 | 3429 | 99,18% |
| Fundo | 0 | 0 | 0 | 3 | 0 | 2 | 0 | 2 | 25747 | 0 | 25754 | 99,97% |
| Relvado | 26 | 6 | 0 | 19 | 278 | 0 | 0 | 0 | 2 | 2694 | 3025 | 89,06% |
| Totais | 4951 | 6997 | 9218 | 1298 | 7271 | 5338 | 5861 | 3479 | 25888 | 3047 | 73348 | |
| Acurácia do Produtor | 97,07% | 52,58% | 64,59% | 75,96% | 95,79% | 87,56% | 77,80% | 97,76% | 99,46% | 88,41% | | 86,53% |

Tabela 15 - Matriz de erros da classificação com o algoritmo RF, com seleção dos atributos (razão das quatro bandas, média das bandas IVP e vermelho, moda da banda vermelho e brilho). Kappa=0,9113

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|---------------|---------------|---------------|---------------|---------------|----------------|---------------|--------------|---------------------|
| Telha cc | 3867 | 860 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4727 | 81,81% |
| Telha ce | 0 | 2127 | 0 | 89 | 1 | 936 | 0 | 0 | 0 | 0 | 3153 | 67,46% |
| Asfalto | 0 | 0 | 8677 | 65 | 4 | 965 | 0 | 0 | 1 | 0 | 9712 | 89,34% |
| Solo | 0 | 8 | 0 | 3502 | 0 | 19 | 0 | 6 | 0 | 0 | 3535 | 99,07% |
| Copa | 0 | 3 | 0 | 57 | 4570 | 1 | 0 | 0 | 0 | 5 | 4636 | 98,58% |
| Telha cz | 0 | 0 | 885 | 0 | 2 | 3290 | 0 | 0 | 0 | 0 | 4177 | 78,76% |
| Telha az | 0 | 0 | 337 | 0 | 0 | 98 | 1860 | 0 | 0 | 0 | 2295 | 81,05% |
| Telha cl | 0 | 0 | 0 | 0 | 0 | 0 | 93 | 1573 | 0 | 0 | 1666 | 94,42% |
| Fundo | 0 | 0 | 22 | 0 | 0 | 13 | 0 | 0 | 58188 | 0 | 58223 | 99,94% |
| Relvado | 0 | 0 | 0 | 12 | 420 | 0 | 0 | 0 | 0 | 971 | 1403 | 69,21% |
| Totais | 3867 | 2998 | 9921 | 3725 | 4997 | 5322 | 1953 | 1579 | 58189 | 976 | 93527 | |
| Acurácia do Produtor | 100,00% | 70,95% | 87,46% | 94,01% | 91,45% | 61,82% | 95,24% | 99,62% | 100,00% | 99,49% | | 94,76% |

Os resultados apontam que o classificador RF apresentou melhor desempenho geral. No entanto, é importante notar que o algoritmo que distinguiu melhor as classes copa e relvado, que são as classes de maior interesse para o estudo da vegetação urbana, foi o classificador C5.0, mas a diferença não é tão significativa para que possamos considerar isso um fator relevante.

O classificador ECHO foi definido com os parâmetros padrões propostos pelo MultiSpec. A matriz de erro dos resultados pode ser visualizada a seguir.

Tabela 16 - Matriz de erros da classificação ECHO. Kappa=0,8311

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|--------------|---------------------|
| Telha cc | 3564 | 1154 | 0 | 0 | 0 | 0 | 0 | 9 | 0 | 0 | 4727 | 75,40% |
| Telha ce | 3 | 3102 | 0 | 43 | 0 | 0 | 4 | 0 | 1 | 0 | 3153 | 98,38% |
| Asfalto | 0 | 0 | 5322 | 2 | 1 | 1318 | 116 | 0 | 2953 | 0 | 9712 | 54,80% |
| Solo | 1 | 26 | 0 | 3478 | 6 | 10 | 3 | 5 | 0 | 6 | 3535 | 98,39% |
| Copa | 0 | 0 | 5 | 6 | 3826 | 0 | 0 | 0 | 705 | 94 | 4636 | 82,53% |
| Telha cz | 0 | 96 | 131 | 1 | 0 | 3672 | 57 | 2 | 218 | 0 | 4177 | 87,91% |
| Telha az | 0 | 0 | 7 | 0 | 0 | 80 | 1775 | 32 | 401 | 0 | 2295 | 77,34% |
| Telha cl | 0 | 0 | 1 | 0 | 0 | 0 | 113 | 1552 | 0 | 0 | 1666 | 93,16% |
| Fundo | 62 | 112 | 40 | 30 | 3 | 167 | 68 | 63 | 57678 | 0 | 58223 | 99,06% |
| Relvado | 0 | 0 | 109 | 3 | 95 | 0 | 0 | 0 | 594 | 602 | 1403 | 42,91% |
| Totais | 3630 | 4490 | 5615 | 3563 | 3931 | 5247 | 2136 | 1663 | 62550 | 702 | 93527 | |
| Acurácia do Produtor | 98,18% | 69,09% | 94,78% | 97,61% | 97,33% | 69,98% | 83,10% | 93,33% | 92,21% | 85,75% | | 90,42% |

Tabela 17 - Matriz de erros da classificação com o algoritmo ECHO depois de filtrar a imagem com o filtro Kuwahara. Kappa=0,9553

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|---------------|---------------|---------------|---------------|----------------|---------------|---------------|--------------|---------------------|
| Telha cc | 4618 | 92 | 0 | 0 | 0 | 0 | 17 | 0 | 0 | 0 | 4727 | 97,69% |
| Telha ce | 0 | 3125 | 0 | 27 | 0 | 1 | 0 | 0 | 0 | 0 | 3153 | 99,11% |
| Asfalto | 0 | 0 | 8390 | 46 | 5 | 1255 | 4 | 0 | 12 | 0 | 9712 | 86,39% |
| Solo | 1 | 0 | 0 | 3524 | 0 | 0 | 3 | 0 | 0 | 7 | 3535 | 99,69% |
| Copa | 0 | 0 | 0 | 0 | 4058 | 0 | 0 | 0 | 0 | 578 | 4636 | 87,53% |
| Telha cz | 0 | 27 | 146 | 0 | 0 | 4004 | 0 | 0 | 0 | 0 | 4177 | 95,86% |
| Telha az | 0 | 0 | 0 | 0 | 0 | 11 | 2284 | 0 | 0 | 0 | 2295 | 99,52% |
| Telha cl | 0 | 0 | 0 | 0 | 0 | 0 | 34 | 1632 | 0 | 0 | 1666 | 97,96% |
| Fundo | 0 | 0 | 0 | 0 | 54 | 10 | 0 | 0 | 58159 | 0 | 58223 | 99,89% |
| Relvado | 0 | 0 | 0 | 1 | 143 | 0 | 0 | 0 | 0 | 1259 | 1403 | 89,74% |
| Totais | 4619 | 3244 | 8536 | 3598 | 4260 | 5281 | 2342 | 1632 | 58171 | 1844 | 93527 | |
| Acurácia do Produtor | 99,98% | 96,33% | 98,29% | 97,94% | 95,26% | 75,82% | 97,52% | 100,00% | 99,98% | 68,28% | | 97,35% |

O algoritmo ECHO, tem a vantagem de ser muito mais simples e robusto para se trabalhar com imagens grandes, rodando sem apresentar travamentos. O tempo necessário para classificar pelo algoritmo ECHO foi de apenas dois segundos, enquanto que apenas a

etapa de segmentação B&S leva cerca de dois minutos, mais as etapas de extração de atributos e a classificação em si. Utilizando a imagem filtrada pelo FK obteve-se resultado ainda superior à classificação orientada a objetos tanto visualmente quanto quantitativamente, no entanto, há uma perda considerável de acurácia na classificação do relvado, que é confundido com copa de árvore.

Também se testou como referência o algoritmo da máxima verossimilhança, muito utilizado para comparação em trabalhos do mesmo gênero. O resultado foi estatisticamente acurado (Tabela 18 e 19), no entanto, o resultado visual deixa a desejar pelo efeito sal e pimenta da imagem.

Tabela 18 - Resultado da classificação pelo algoritmo da máxima verossimilhança. Kappa 0,9380

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|---------------|---------------|---------------|---------------|---------------|----------------|---------------|--------------|---------------------|
| Telha cc | 4644 | 63 | 0 | 0 | 0 | 0 | 11 | 9 | 0 | 0 | 4727 | 98,24% |
| Telha ce | 11 | 3095 | 0 | 44 | 0 | 0 | 0 | 0 | 0 | 3 | 3153 | 98,16% |
| Asfalto | 0 | 0 | 8099 | 57 | 11 | 1530 | 14 | 0 | 1 | 0 | 9712 | 83,39% |
| Solo | 1 | 29 | 0 | 3470 | 6 | 0 | 0 | 5 | 0 | 24 | 3535 | 98,16% |
| Copa | 0 | 0 | 0 | 5 | 3916 | 0 | 0 | 0 | 0 | 715 | 4636 | 84,47% |
| Telha cz | 0 | 26 | 433 | 3 | 0 | 3697 | 16 | 2 | 0 | 0 | 4177 | 88,51% |
| Telha az | 0 | 0 | 2 | 0 | 0 | 68 | 2124 | 101 | 0 | 0 | 2295 | 92,55% |
| Telha cl | 0 | 0 | 0 | 0 | 0 | 0 | 46 | 1620 | 0 | 0 | 1666 | 97,24% |
| Fundo | 0 | 0 | 8 | 0 | 15 | 11 | 1 | 0 | 58188 | 0 | 58223 | 99,94% |
| Relvado | 0 | 0 | 0 | 7 | 154 | 0 | 0 | 0 | 0 | 1242 | 1403 | 88,52% |
| Totais | 4656 | 3213 | 8542 | 3586 | 4102 | 5306 | 2212 | 1737 | 58189 | 1984 | 93527 | |
| Acurácia do Produtor | 99,74% | 96,33% | 94,81% | 96,77% | 95,47% | 69,68% | 96,02% | 93,26% | 100,00% | 62,60% | | 96,33% |

Tabela 19 - Matriz de erros da classificação pelo algoritmo da máxima verossimilhança depois de usado o filtro Kuwahara. Kappa=0,9542

| | Telha cerâmica clara | Telha cerâmica escura | Asfalto | Solo exposto | Copa | Telha cinza | Telha azulada | Telha clara | Fundo | Relvado | Totais | Acurácia do Usuário |
|-----------------------------|----------------------|-----------------------|---------------|---------------|---------------|---------------|---------------|----------------|---------------|---------------|--------------|---------------------|
| Telha cc | 4647 | 63 | 0 | 0 | 0 | 0 | 17 | 0 | 0 | 0 | 4727 | 98,31% |
| Telha ce | 0 | 3125 | 0 | 27 | 0 | 1 | 0 | 0 | 0 | 0 | 3153 | 99,11% |
| Asfalto | 0 | 0 | 8425 | 46 | 5 | 1220 | 4 | 0 | 12 | 0 | 9712 | 86,75% |
| Solo | 1 | 0 | 0 | 3511 | 0 | 0 | 3 | 0 | 0 | 20 | 3535 | 99,32% |
| Copa | 0 | 0 | 0 | 0 | 4120 | 0 | 0 | 0 | 0 | 516 | 4636 | 88,87% |
| Telha cz | 0 | 27 | 343 | 0 | 0 | 3807 | 0 | 0 | 0 | 0 | 4177 | 91,14% |
| Telha az | 0 | 0 | 0 | 0 | 0 | 12 | 2283 | 0 | 0 | 0 | 2295 | 99,48% |
| Telha cl | 0 | 0 | 0 | 0 | 0 | 0 | 34 | 1632 | 0 | 0 | 1666 | 97,96% |
| Fundo | 0 | 0 | 0 | 0 | 54 | 10 | 0 | 0 | 58159 | 0 | 58223 | 99,89% |
| Relvado | 0 | 0 | 0 | 1 | 118 | 0 | 0 | 0 | 0 | 1284 | 1403 | 91,52% |
| Totais | 4648 | 3215 | 8768 | 3585 | 4297 | 5050 | 2341 | 1632 | 58171 | 1820 | 93527 | |
| Acurácia do Produtor | 99,98% | 97,20% | 96,09% | 97,94% | 95,88% | 75,39% | 97,52% | 100,00% | 99,98% | 70,55% | | 97,29% |



Figura 15 - Imagem de referência para analisar os resultados da classificação

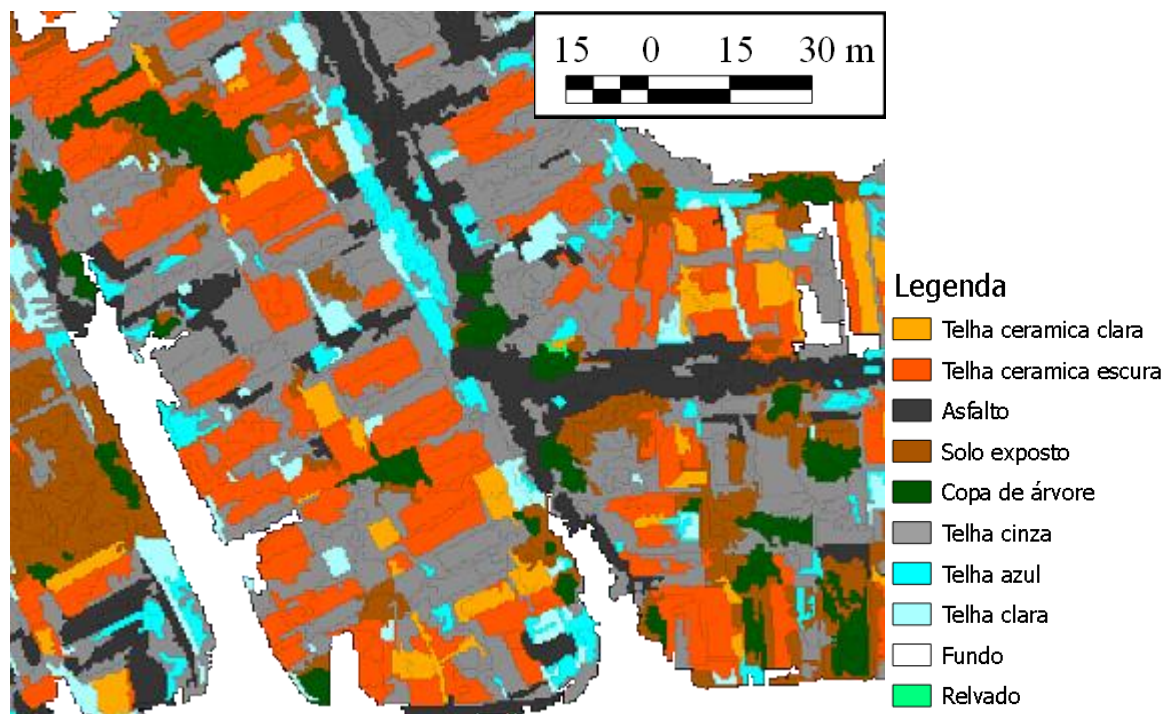


Figura 16 - Resultado da classificação orientada a objetos com segmentador Baatz e Schäpe (2000) e classificador Random Forests

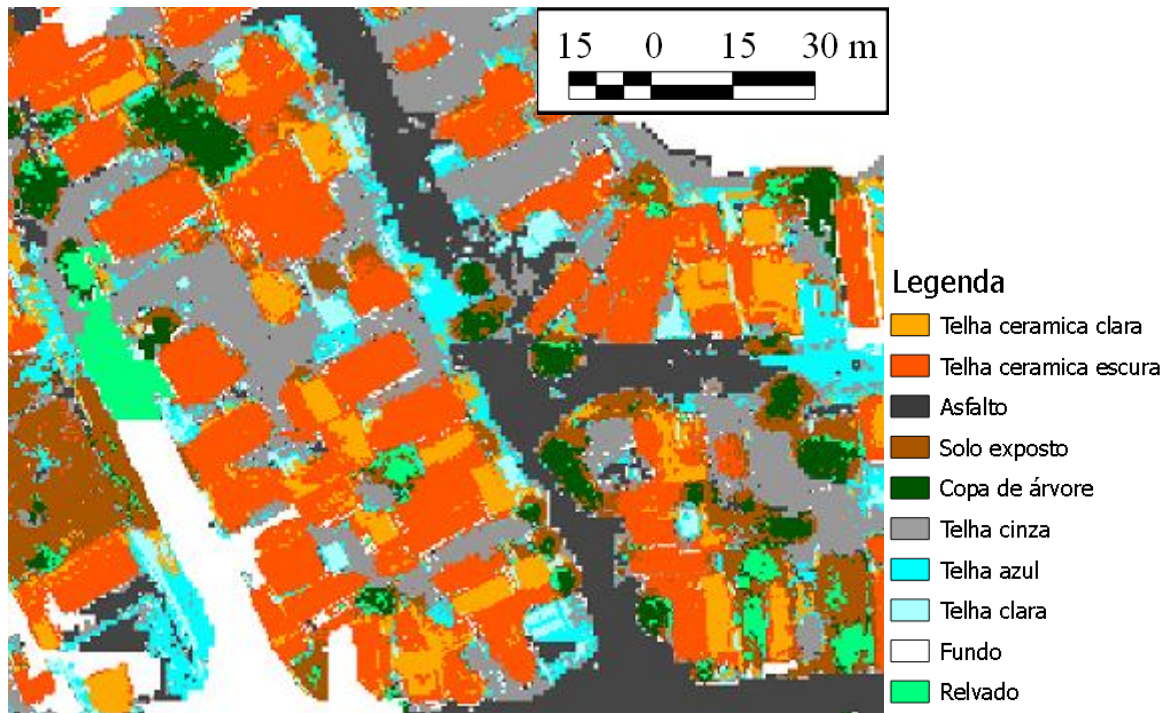


Figura 17 - Resultado da classificação ECHO

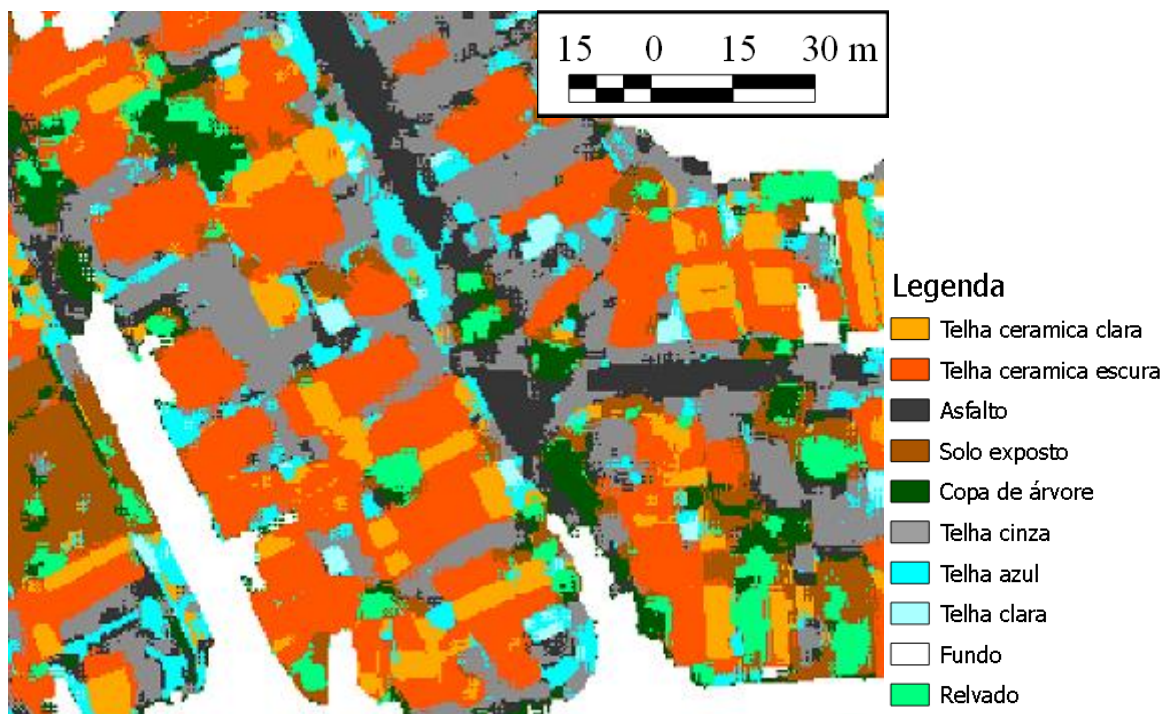


Figura 18 - Resultado da classificação ECHO após filtragem Kuwahara

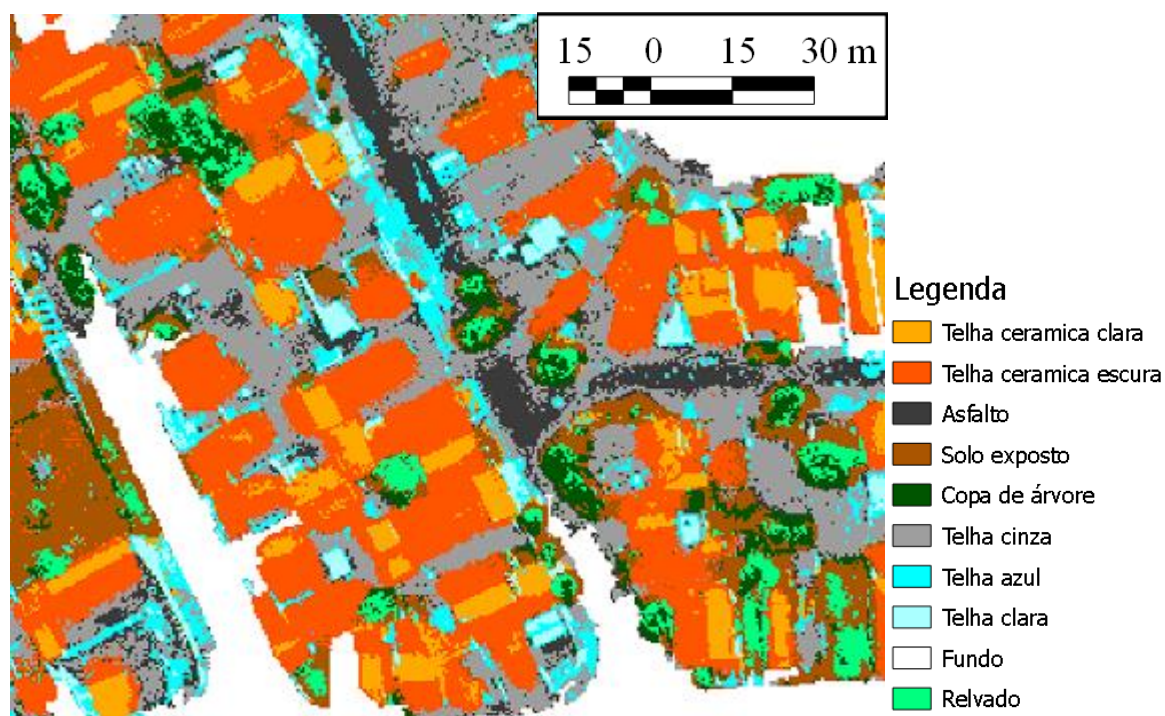


Figura 19 - Resultado da classificação pela máxima verossimilhança

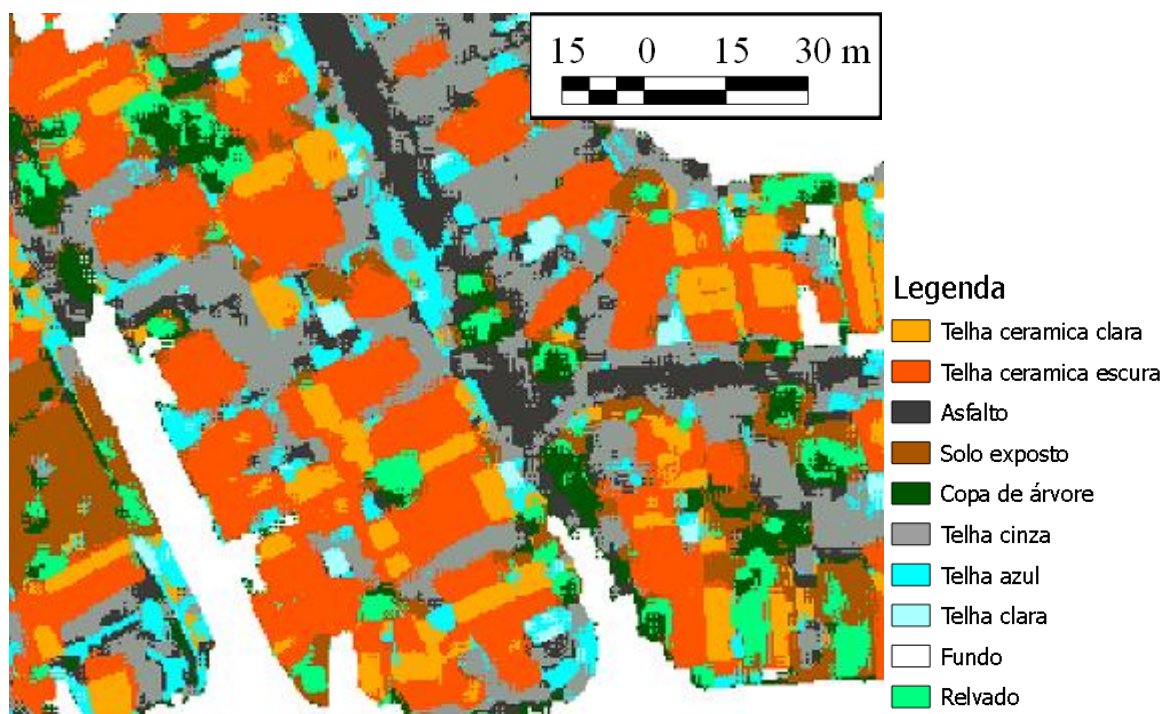


Figura 20 - Resultado da classificação pela máxima verossimilhança após a filtragem Kuwahara

A classificação orientada a objetos mostra uma clara vantagem no visual nos resultados, com os limites entre as classes bem definidos. Além disso, este foi o que apresentou a melhor distinção e detecção das copas de árvore, apesar de este superestimar sua área por conta da segmentação agrupar pixels que não são de copa.

Os classificadores por pixel, por outro lado demonstram acurácia estatisticamente superiores, mas resultados visuais com muito ruído, com vantagem para o classificador ECHO. A filtragem Kuwahara demonstra melhora substancial dos resultados visuais, sendo que os limites entre as classes ficam mais bem definidos, como é de se esperar pela natureza do filtro.

3 CONCLUSÃO

Os classificadores ECHO e da máxima verossimilhança já apresentam soluções gratuitas eficientes e de simples utilização, possibilitando a sua ampla utilização, principalmente para fins não acadêmicos. Além de serem de simples utilização, ainda apresentam a vantagem de possuírem uma implementação eficiente, sobretudo no MultiSpec que possibilita o trabalho em imagens grandes.

Por outro lado, a classificação orientada a objetos, até o momento, é de uso bastante restrito pelo alto custo das licenças comerciais e pela pouca disponibilização de ferramentas de uso gratuito com implementações consistentes e simples. Apesar disso, foi possível atingir bons níveis de acurácia, mesmo sem a utilização de regras contextuais definidas manualmente, principalmente utilizando o algoritmo *Random Forests*, que apresentou os melhores resultados. No entanto, ainda não existe implementação simples e integrada a um software GIS que permita o seu uso.

Para que seja possível a utilização extensiva da classificação orientada a objetos é necessário que se desenvolvam ferramentas que simplifiquem as tarefas de definição dos parâmetro da segmentação e a extração e seleção de atributos, etapas necessárias para se obter resultados consistentes.

Quanto à definição dos parâmetros da segmentação, mais estudos poderiam indicar uma forma mais direta e rápida para definição do parâmetro de escala para o algoritmo Baatz e Schäpe (2000), já que se verificou a existência de uma relação linear entre os melhores resultados de segmentação. Até o momento, os algoritmos genéticos apresentam resultados satisfatórios, no entanto levam muito tempo para serem processados.

Os atributos que demonstraram maior importância pelos diversos algoritmos testados foram os espectrais, principalmente os atributos de razão de contribuição de cada banda para o brilho e a média dos valores dos pixels para cada uma das bandas multiespectrais.

O algoritmo para realizar a filtragem Kuwahara desenvolvido neste trabalho apresentou bom desempenho e demonstrou aumentar a qualidade dos resultados das classificações por pixel, sobretudo visualmente. Recomenda-se o uso deste filtro para imagens que apresentam heterogeneidade e ruído, pois o filtro suaviza o ruído e ao mesmo tempo realça as diferenças entre os diferentes objetos da paisagem.

REFERÊNCIAS

- ACKERMAN, B. Temporal march of the Chicago heat island. **Journal of Applied Meteorology**, Boston, v. 24, p. 547–554, June 1985.
- AKBARI, H.; POMERANTZ, M.; TAHA, H. Cool surfaces and shade trees to reduce energy use and improve air quality in urban areas. **Solar Energy**, Kidlington v. 70, n. 3, p. 295-310, 2001.
- BAATZ, M.; SCHÄPE, A. Multiresolution segmentation: an optimization approach for high quality multi-scale image segmentation. In: STROBL, J.; BLASCHKE, T.; GRIESEBNER, G. (Ed.). **Angewandte Geographische Informations-Verarbeitung XII**. Salzburg: Beiträge zum AGIT-Symposium, 2000. p. 12-23.
- BEDIENT, P.; HUBER, W.; VIEUX, B. **Urban Hidrology**. In: Hydrology and Floodplain Analysis. 4. ed. Prentice Hall, Upper Saddle River, EUA. 2008. p. 356-443.
- BENZ, U. C.; HOFMANN, P.; WILLHAUCK, G.; LINGENFELDER, I.; HEYNEN, M. Multi-resolution, object-oriented fuzzy analysis of remote sensing data for GIS-ready information. **ISPRS Journal of Photogrammetry and Remote Sensing**, Amsterdam v. 58, n. 3/4, p. 239-258, Jan. 2004.
- BHATTACHARYYA, A. On a measure of divergence between two statistical populations defined by their probability distributions. **Bulletin of the Calcutta Mathematical Society**, Calcuta, v. 35, n. 1, p. 99-109, 1943.
- BIEHL, L.; LANDGREBE, D. MultiSpec: a tool for multispectral–hyperspectral image data analysis. **Computers & Geosciences**, Kidlington, v. 28, n. 10, p. 1153-1159, Dec. 2002.
- BINS, L.; FONSECA, L.; ERTHAL, G.; MITSUO II, F. Satellite imagery segmentation: a region growing approach. In: SIMPÓSIO BRASILEIRO DE SENSORIAMENTO REMOTO 8., 1996, Salvador, **Anais...** Disponível em: <<http://bibdigital.sid.inpe.br/rep-sid.inpe.br/deise/1999/02.05.09.30>>. Acesso em: 08 mar. 2013
- BLASCHKE, T.; KUX, H. **Sensoriamento remoto e SIG avançados: novos sistemas sensores, métodos inovadores**. 2. ed. São Paulo: Oficina de Textos, 2007. 303 p.
- BREIMAN, L. Random forests. **Machine learning**, Boston, v. 45, p. 5-32, 2001.
- CÂMARA, G.; SOUZA, R.; FREITAS, U.; GARRIDO, J. SPRING: Integrating remote sensing and GIS by object-oriented data modelling. **Computers & Graphics**, New York, v. 20, n. 3, p. 395-403, 1996.
- CÂMARA, G.; VINHAS, L.; FERREIRA, K.R.; QUEIROZ, G.R.; SOUZA, R.C.M.; MONTEIRO, A.M.V.; CARVALHO, M.T.; CASANOVA, M.A.; FREITAS, U.M. TerraLib: an open source GIS library for large-scale environmental and socio-economic applications. In: HALL, G.B. (Ed.). **Open source approaches for spatial data handling**. Berlin: Springer, 2007. p. 24.

- CAZES, T.B.; COSTA, G.A.O.P.; FEITOSA, R.Q. Automatic evaluation of segmentation parameters. In: SIMPÓSIO BRASILEIRO DE GEOMÁTICA, 2007, Presidente Prudente. **Anais...** Presidente Prudente 2007. p. 699-705. CAZES, T.; COSTA, G.A.O.P.; FEITOSA, R.Q.; FREDRICH, C.M.B.; GOBBI, H.V.S.; PALERMO, B.C. **Segmentation Parameters Tuner (SPT)**. Rio de Janeiro: LVC; PUC-Rio, 2012.
- COHEN, J. A coefficient of agreement for nominal scales. **Educational and Psychological Measurement**, New York, v. 20, n. 1, p. 37-46, 1960.
- CONGALTON, R.G. Accuracy assessment and validation of remotely sensed and other spatial information. **International Journal of Wildland Fire**, Collingwood, v. 10, n. 4, p. 321-328, 1 jan. 2001.
- CORTES, C.; VAPNIK, V. Support-vector networks. **Machine Learning**, Boston, v. 20, n. 3, p. 273-297, Sept. 1995.
- COSTA, G.; PINHO, C.; FEITOSA, R.; ALMEIDA, C.M. de; KUX, H.J.H.; FONSECA, L.M.G.; OLIVEIRA, D.A.B. INTERIMAGE: an open source platform for automatic image interpretation. In: SIMPÓSIO BRASILEIRO DE GEOMÁTICA, 2., 2007, Presidente Prudente. **Anais...** Presidente Prudente: UNESP, 2007. Disponível em: <http://docs.fct.unesp.br/departamentos/cartografia/eventos/2007_II_SBG/artigos/A_104.pdf>. Acesso em: 12 mar. 2013.
- DRĂGUȚ, L.; TIEDE, D.; LEVICK, S. ESP: a tool to estimate scale parameter for multiresolution image segmentation of remotely sensed data. **International Journal of Geographical Information Science**, London, v. 24, n. 6, p. 859-871, 2010.
- ERSOY, O. K. Consensual and hierarchical classification of remotely sensed multispectral images. **IEEE Transactions on Geoscience and Remote Sensing**, New York, v. 45, n. 9, p. 2953-2963, Sept. 2007.
- ESPINDOLA, G.M.; CAMARA, G.; REIS, I.A.; BINS, L.S.; MONTEIRO, A.M. Parameter selection for region growing image segmentation algorithms using spatial autocorrelation. **International Journal of Remote Sensing**, London, v. 27, n. 14, p. 3035-3040, July 2006.
- FEITOSA, R.; COSTA, G.; CAZES, T.; FEIJO, B. A genetic approach for the automatic adaptation of segmentation parameters. In: INTERNATIONAL CONFERENCE ON OBJECT BASED IMAGE ANALYSIS, 1., 2006, Salzbourg. **Proceedings...** Disponível em: <http://www.isprs.org/proceedings/XXXVI/4-C42/Papers/11_Adaption and further development III/OBIA2006_Feitosa_et_al.pdf>. Acesso em: 12 mar. 2013
- FERNANDES, R.; NUNES, G.; SILVA, T. Classificação orientada a objetos aplicada na caracterização da cobertura da terra no Araguaia. **Pesquisa Agropecuária Brasileira**, Brasília, v. 47, n. 9, p. 1251-1260, set. 2012.
- FOODY, G.M.; MATHUR, A. Toward intelligent training of supervised image classifications: directing training data acquisition for SVM classification. **Remote Sensing of Environment**, New York, v. 93, n. 1/2, p. 107-117, Oct. 2004.

- GABOR, P.; JOMBACH, S. The relation between the biological activity and the land surface temperature in Budapest. **Applied Ecology and Environmental Research**, Budapest, v. 7, n. 3, p. 241-251, 2009.
- GAO, Y.; MAS, J.F.; KERLE, N.; NAVARRETE PACHECO, J.A. Optimal region growing segmentation and its effect on classification accuracy. **International Journal of Remote Sensing**, London, v. 32, n. 13, p. 3747-3763, July 2011.
- GASTON, K.J.; FULLER, R.A.; LORAM, A.; MACDONALD, C.; POWER, S.; DEMPSEY, N. Urban domestic gardens (XI): variation in urban wildlife gardening in the United Kingdom. **Biodiversity and Conservation**, Dordrecht, v. 16, n. 11, p. 3227-3238, May 2007.
- GASTON, K.J.; SMITH, R.M.; THOMPSON, K.; WARREN, P.H. Urban domestic gardens (II): experimental tests of methods for increasing biodiversity. **Biodiversity and Conservation**, Dordrecht, v. 14, p. 395-413, 2005.
- GASTON, K.J.; WARREN, P.H.; THOMPSON, K.E.N.; SMITH, R.M. Urban domestic gardens (VI): environmental correlates of invertebrate species richness. **Biodiversity and Conservation**, Dordrecht, v. 15, n. 8, p. 235-253, Dec. 2005a. <http://ccn.ibict.br/busca.jsf>
- _____. Urban domestic gardens (IV): the extent of the resource and its associated features. **Biodiversity and Conservation**, Dordrecht, v. 14, n. 14, p. 3327-3349, Dec. 2005b.
- GDAL DEVELOPMENT TEAM. **GDAL - Geospatial Data Abstraction Library, Version 1.9.20**. Open Source Geospatial Foundation, 2012. Disponível em: <<http://www.gdal.org>>. Acesso em: 02 abr 2013.
- GENOVEZ, P. **Segmentação e classificação de imagens SAR aplicadas à detecção de alvos escuros em áreas oceânicas de exploração e produção de petróleo**. 2010. 253 p. Tese (Doutorado em Engenharia Civil) - Universidade Federal do Rio de Janeiro, Rio de Janeiro. 2010.
- GODDARD, M.A.; DOUGILL, A.J.; BENTON, T.G. Scaling up from gardens: biodiversity conservation in urban environments. **Trends in Ecology & Evolution**, London, v. 25, n. 2, p. 90-98, Feb. 2010.
- GODEFROID, S.; KOEDAM, N. Urban plant species patterns are highly driven by density and function of built-up areas. **Landscape Ecology**, Dordrecht, v. 22, n. 8, p. 1227-1239, Apr. 2007.
- GRIMM, N.B.; FAETH, S.H.; GOLUBIEWSKI, N.E.; REDMAN, C.L.; WU, J.; BAI, X.; BRIGGS, J. M. Global change and the ecology of cities. **Science**, New York, v. 319, n. 5864, p. 756-760, Feb. 2008.
- HALL, M. Smooth operator : smoothing seismic interpretations and attributes. **The Leading Edge**, Tulsa, v. 26, p. 16-20, 2007.
- HARALICK, R.M. Statistical and structural approaches to texture. **Proceedings of the IEEE**, Raleigh, v. 67, n. 5, p. 786-804, 1979.

- HEUMANN, B.W. An object-based classification of mangroves using a hybrid decision tree—support vector machine approach. **Remote Sensing**, Basel, v. 3, n. 12, p. 2440-2460, Nov. 2011.
- HOPE, D.; GRIES, C.; ZHU, W.; FAGAN, W.F.; REDMAN, C.L.; GRIMM, N.B.; NELSON, A.L.; MARTIN, C.; KINZIG, A. Socioeconomics drive urban plant diversity. **Proceedings of the National Academy of Sciences of the United States of America**, Washington, v. 100, n. 15, p. 8788-92, July 2003.
- HUANG, X.; ZHANG, L. An adaptive mean-shift analysis approach for object extraction and classification. **IEEE Transactions on Geoscience and Remote Sensing**, Piscataway, v. 46, n. 12, p. 4173-4185, 2008.
- _____. An SVM ensemble approach combining spectral, structural, and semantic features for the classification of high-resolution remotely sensed imagery. **IEEE Transactions on Geoscience and Remote Sensing**, Piscataway, v. 51, n. 1, p. 257-272, Jan. 2013.
- HUGUNIN, J.; FULTON, J.; ASCHER, D.; DUBOIS, P.; HINSEN, K. **Numpy 1.7.0**. 2013. Disponível em: <<http://www.numpy.org/>>. Acesso em: 02 abr. 2013.
- INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. Disponível em: <<http://www.ibge.gov.br/>>. Acesso em: 13 dez. 2012.
- JENERETTE, G.D.; HARLAN, S.L.; BRAZEL, A.; JONES, N.; LARSEN, L.; STEFANOV, W.L. Regional relationships between surface temperature, vegetation, and human settlement in a rapidly urbanizing ecosystem. **Landscape Ecology**, Dordrecht, v. 22, n. 3, p. 353-365, Dec. 2006.
- JIM, C.Y.; CHEN, W.Y. Ecosystem services and valuation of urban forests in China. **Cities**, Oxford, v. 26, n. 4, p. 187-194, Aug. 2009.
- KETTIG, R.; LANDGREBE, D. Classification of multispectral image data by extraction and classification of homogeneous objects. **IEEE Transactions on Geoscience Electronics**, Piscataway, v. 14, n. 1, p. 19-26, Jan. 1976.
- Konijnendijk, C.C., Ricard, R.M., Kenney, A. and Randrup, T.B. Defining urban forestry – A comparative perspective of North America and Europe. **Urban Forestry & Urban Greening**, v.4 n. 3-4, p. 93-103, 2006.
- KUHN, M.; WESTON, S.; COULTER, N.; C CODE FOR C5.0 BY R. QUINLAN. **C50: C5.0 decision trees and rule-based models**. R package version 0.1.0-14, 2013. Disponível em: <<http://cran.r-project.org/package=C50>>. Acesso em: 02 abr 2013.
- LANDGREBE, D.A. The development of a spectral-spatial classifier for earth observational data. **Pattern Recognition**, Oxford, v. 12, n. 3, p. 165-175, Jan. 1980.
- LIAW, A.; WIENER, M. Classification and regression by randomForest. **R News**, Ohio, v. 2, n. 3, p. 18-22, 2002.

LOMBARDO, M. **Ilha de calor nas metrópoles: o exemplo de São Paulo**. São Paulo: Hucitec, 1985. 244 p.

LORAM, A.; WARREN, P.H.; GASTON, K.J. Urban domestic gardens (XIV): the characteristics of gardens in five cities. **Environmental Management**, New York, v. 42, n. 3, p. 361-376, Sept. 2008.

LORAM, A.; THOMPSON, K.; WARRER, P.H.; GASTON, K.J. Urban domestic gardens (XII): the richness and composition of the flora in five UK cities. **Journal of Vegetation Science**, Washington, v. 19, p. 321-330, Jan. 2008.

LORAM, A.; TRATALOS, J.; WARREN, P.H.; GASTON, K.J. Urban domestic gardens (X): the extent & structure of the resource in five major cities. **Landscape Ecology**, Dordrecht, v. 22, n. 4, p. 601-615, Jan. 2007.

LU, D.; WENG, Q. A survey of image classification methods and techniques for improving classification performance. **International Journal of Remote Sensing**, London, v. 28, n. 5, p. 823-870, Mar. 2007.

LUCK, G.W. A review of the relationships between human population density and biodiversity. **Biological Reviews**, Hoboken, v. 82, n. 4, p. 607-645, Nov. 2007.

MATHIEU, R.; ARYAL, J.; CHONG, A.K. Object-based classification of ikonos imagery for mapping large-scale vegetation communities in urban areas. **Sensors**, Peterborough, v. 7, n. 11, p. 2860-2880, 2007.

MATHIEU, R.; FREEMAN, C.; ARYAL, J. Mapping private gardens in urban areas using object-oriented techniques and very high-resolution satellite imagery. **Landscape and Urban Planning**, Amsterdam, v. 81, n. 3, p. 179-192, June 2007.

MAXWELL, T. **Object-oriented classification**: classification of pan-sharpened quickbird imagery and a fuzzy approach to improving image segmentation efficiency. 2005. 157 p. Thesis (Master in Engineer) - University of New Brunswick, New Brunswick, 2005.

MCNAB, D. **Oygene**. 2012. Disponível em: <<https://github.com/blaa/PyGene>>. Acesso em: 02 abr. 2013.

MEINEL, G.; NEUBERT, M. A comparison of segmentation programs for high resolution remote sensing data. In: INTERNATIONAL SOCIETY FOR PHOTOGRAMMETRY AND REMOTE SENSING CONGRESS, 35., 2004, Istanbul. **Proceedings...** Istanbul: International Society for Photogrammetry and Remote Sensing Congress, 2004. Disponível em: <https://sharepoint.ngdc.wvu.edu/sites/digital_soils/RemoteSensing/References/comparison_of_image_segmentation_software.pdf>. Acesso em: 08 mar. 2013

MEYER, D.; DIMITRIADOU, E.; HORNIK, K.; WEINGESSEL, A.; LEISCH, F. **e1071**: Misc Functions of the Department of Statistics (e1071), TU Wien. R package version 1.6-1, 2012. Disponível em: <<http://cran.r-project.org/package=e1071>>. Acesso em: 02 abr 2013

MICROSOFT CORPORATION. **Microsoft Visual C++ 2008 Express Edition**. 2007. Disponível em: <go.microsoft.com/?linkid=7729279>. Acesso em: 02 abr 2013.

MILLER, R. **Urban forestry: planning and managing urban greenspaces**. Waveland Press, Long Grove, Ill. 2007. p. 502

MORAN, E. Land cover classification in a complex urban-rural landscape with Quickbird imagery. **Photogrammetric Engineering and Remote Sensing**, Bethesda, v. 76, n. 10, p. 1159-1168, 2010.

MOREIRA, T.C.L. **Interação da vegetação arbórea e poluição atmosférica na cidade de São Paulo**. 2010. 79 p. Dissertação (Mestrado em Ciências), Escola Superior de Agricultura "Luiz de Queiroz", Universidade de São Paulo, Piracicaba, 2010.

MYRUP, L.O. A numerical model of the urban heat island. **Journal of Applied Meteorology and Climatology**, Boston, v. 8, p. 908–918, 1969.

NAGAO, M.; MATSUYAMA, T. Edge preserving smoothing. **Computer Graphics and Image Processing**, San Diego, v. 9, n. 4, p. 394-407, Apr. 1979.

NIEMELÄ, J. Is there a need for a theory of urban ecology? **Urban Ecosystems**, Dordrecht, v. 3, n. 1, p. 57–65, 1999a.

_____. Ecology and urban planning. **Biodiversity and Conservation**, Dordrecht, v. 8, n. 1, p. 119–131, 1999b.

NOVACK, T. **Classificação da cobertura da terra e do uso do solo urbano utilizando o sistema InterIMAGE e imagens do sensor Quickbird**. 2009. 216 p. Dissertação (Mestrado em Sensoriamento Remoto) - Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2009.

OKE, T.R. City size and the urban. **Atmospheric Environment**, Oxford, v. 7, p. 769-779, 1973.

PAULEIT, S.; DUHME, F. Assessing the environmental performance of land cover types for urban planning. **Landscape and Urban Planning**, Amsterdam, v. 52, p. 1-20, 2000.

PHINN, S.; STANFORD, M.; SCARTH, P.; MURRAY, A. T.; SHYY, P. T. Monitoring the composition of urban environments based on the vegetation-impervious surface-soil (VIS) model by subpixel analysis techniques. **International Journal of Remote Sensing**, Abingdom, v. 23, n. 20, p. 4131-4153, 2002.

PRECORD, C. **Editra**. 2013. Disponível em: <<http://editra.org>>. Acesso em: 02 abr. 2013.

PRESTON-WERNER, T.; WANSTRATH, C.; HYETT, P. **GitHub**. Disponível em: <<https://github.com/>>. Acesso em: 02 abr. 2013.

QUANTUM GIS DEVELOPMENT TEAM. **Quantum GIS Geographic Information System**. Open Source Geospatial Foundation Project, 2012. Disponível em: <<http://qgis.osgeo.org>>. Acesso em: 02 abr. 2013.

R DEVELOPMENT CORE TEAM. **A language and environment for statistical computing**. Vienna: R Foundation for Statistical Computing, 2008. Disponível em: <<http://www.r-project.org>>. Acesso em: 02 abr. 2013.

RIDD, M.K. Exploring a V-I-S (vegetation-impervious surface-soil) model for urban ecosystem analysis through remote sensing: comparative anatomy for cities†. **International Journal of Remote Sensing**, Abingdom, v. 16, n. 12, p. 2165-2185, Aug. 1995.

RIVERBANK COMPUTING. **PyQt4 4.10**. 2013. Disponível em: <<http://www.riverbankcomputing.com/software/pyqt/>>. Acesso em: 02 abr. 2013.

ROBERTS, D.G.; AYRE, D.J.; WHELAN, R.J. Urban plants as genetic reservoirs or threats to the integrity of bushland plant populations. **Conservation Biology**, Hoboken, v. 21, n. 3, p. 842-852, June 2007.

SANTOS, A.R. dos; PELUZIO, T.M.O.; SAITO, N.S. **SPRING 5.1.2**: passo a passo: aplicações práticas. Alegre: CAUFES, 2010. 153 p.

SAVARD, J.; CLERGEAU, P.; MENNECHEZ, G. Biodiversity concepts and urban ecosystems. **Landscape and Urban Planning**, Amsterdam, v. 48, p. 131-142, 2000.

SCHOWENGERDT, R.A. **Remote sensing**: models and methods for image processing. 2nd ed. San Diego: Academic Press, 1997. 522 p.

SHASHUA-BAR, L.; POTCHTER, O.; BITAN, A.; YAAKOV, Y. Microclimate modelling of street tree species effects within the varied urban morphology in the Mediterranean city of Tel Aviv, Israel. **International Journal of Climatology**, Hoboken, v. 57, p. 44-57, Mar. 2010.

SILVA FILHO, D.F.; BORTOLETO, S. Uso de indicadores de diversidade na adefinição de plano de manejo de arborização viária de águas de São Pedro-SP. **Revista Árvore**, Viçosa, v. 29, p. 973-982, 2005.

SILVA FILHO, D.F.; PIZETTA, P.U.C.; ALMEIDA, J.B.S.A.; PIVETTA, K.F.L.; FERRAUDO, A.S. Banco de dados relacional para cadastro, avaliação e manejo da arborização em vias públicas. **Revista Árvore**, Viçosa, v. 26, p. 629-642, 2002.

SMITH, R.M.; GASTON, K.J.; WARREN, P.H.; THOMPSON, K. Urban domestic gardens (V): relationships between landcover composition, housing and landscape. **Landscape Ecology**, Dordrecht, v. 20, n. 2, p. 235-253, Feb. 2005.

_____. Urban domestic gardens (VIII) : environmental correlates of invertebrate abundance. **Biodiversity and Conservation**, Dordrecht, v. 15, n. 8, p. 2515-2545, July 2006a.

SMITH, R.M.; THOMPSON, K.; WARREN, P.H.; GASTON, K.J. Urban domestic gardens (XIII): Composition of the bryophyte and lichen floras, and determinants of species richness. **Biological Conservation**, Oxford, v. 143, n. 4, p. 873-882, Apr. 2010.

SMITH, R.; THOMPSON, K.; HODGSON, J.; WARREN, P.; GASTON, K. Urban domestic gardens (IX): Composition and richness of the vascular plant flora, and implications for native biodiversity. **Biological Conservation**, Oxford, v. 129, n. 3, p. 312-322, May 2006b.

SONG, C.; WOODCOCK, C.E.; SETO, K.C.; LENNEY, M.P.; MACOMBER, S.A. Classification and change detection using landsat TM data. **Remote Sensing of Environment**, New York, v. 75, n. 2, p. 230-244, Feb. 2001.

STEHMAN, S.V. Selecting and interpreting measures of thematic classification accuracy. **Remote Sensing of Environment**, New York, v. 62, n. 1, p. 77-89, Oct. 1997.

STEWART, G.H.; MEURK, C.D.; IGNATIEVA, M.E.; BUCKLEY, H.L.; MAGUEUR, A.; CASE, B.S.; HUDSON, M.; PARKER, M. Urban biotopes of Aotearoa New Zealand (URBANZ) II: Floristics, biodiversity and conservation values of urban residential and public woodlands, Christchurch. **Urban Forestry & Urban Greening**, Jena, v. 8, n. 3, p. 149-162, 2009.

STORY, M.; CONGALTON, R. Accuracy assessment-A user's perspective. **Photogrammetric Engineering and Remote Sensing**, Bethesda, v. 52, n. 3, p. 397-399, 1986.

TAHA, H. Episodic performance and sensitivity of the urbanized MM5 (uMM5) to perturbations in surface properties in Houston Texas. **Boundary-Layer Meteorology**, Dordrecht, v. 127, n. 2, p. 193-218, Jan. 2008.

THOMAS, N.; HENDRIX, C.; CONGALTON, R.G.A Comparison of urban mapping methods using high-resolution digital imagery. **Photogrammetric Engineering & Remote Sensing and Remote Sensing**, Bethesda, v. 69, n. 9, p. 963-972, 2003.

TRATALOS, J.; FULLER, R.; WARREN, P.; DAVIES, R.; GASTON, K. Urban form, biodiversity potential and ecosystem services. **Landscape and Urban Planning**, Amsterdam, v. 83, n. 4, p. 308-317, Dec. 2007.

ULRICH, R.S. Human responses to vegetation and landscapes. **Landscape and Urban Planning**, Amsterdam, v. 13, p. 29-44, 1986.

UPDIKE, T.; COMP, C. **Radiometric use of WORLDVIEW-2 imagery**. Longmont, DigitalGlobe, 2010. 17 p. (Technical Note, revision 1.0). Disponível em: <https://www.digitalglobe.com/downloads/Radiometric_Use_of_WorldView-2_Imagery.pdf>. Acesso em: 02 abr. 2013.

U.S. ENVIRONMENTAL PROTECTION AGENCY. **Reducing urban heat islands** : compendium of strategies urban heat island basics. Washington, D.C., 2008. 177 p.

VAN ROSSUM, G. **Python programming language**. National Research Institute for Mathematics and Computer Science, 1989. Disponível em: <www.python.org/>. Acesso em: 02 abr. 2013

WARMERDAM, F. The geospatial data abstraction library. In: HALL, G.B.; LEAHY, M. (Ed.). **Open source approaches in spatial data handling**. Berlin; Heidelberg: Springer, 2008. v. 2, p. 87-104.

WENTZ, E.A.; NELSON, D.; RAHMAN, A.; STEFANOV, W.L.; ROY, S. SEN. Expert system classification of urban land use/cover for Delhi, India. **International Journal of Remote Sensing**, Abingdom, v. 29, n. 15, p. 4405-4427, Aug. 2008.

WENTZ, E.; STEFANOV, W.; GRIES, C.; HOPE, D. Land use and land cover mapping from diverse data sources for an arid urban environments. **Computers Environment and Urban Systems**, Oxford, v. 30, n. 3, p. 320-346, May 2006.

WOOD, B.C.; PULLIN, A.S. Persistence of species in a fragmented urban landscape: the importance of dispersal ability and habitat availability for grassland butterflies. **Biodiversity and Conservation**, Dordrecht, v. 11, n. 8, p. 1451-1468, 2002.

ZHANG, X.; FENG, X.; JIANG, H. Object-oriented method for urban vegetation mapping using IKONOS imagery. **International Journal of Remote Sensing**, Abingdom, v. 31, n. 1, p. 177-196, Jan. 2010.

APÊNDICES

APÊNDICE A

| Geração | Escala | Compactação | Cor | Peso das Bandas | | | RBSB |
|-----------|--------|-------------|--------|-----------------|--------|--------|--------|
| | | | | 1 | 2 | 3 | |
| Recorte 1 | | | | | | | |
| 0 | 128 | 0.7606 | 0.4189 | 0.2716 | 0.3477 | 0.5621 | 0.7287 |
| 1 | 139 | 0.8149 | 0.4976 | 0.3748 | 0.4812 | 0.5424 | 0.6580 |
| 2 | 160 | 0.7677 | 0.4497 | 0.3338 | 0.4810 | 0.1785 | 0.6359 |
| 3 | - | - | - | - | - | - | - |
| 4 | - | - | - | - | - | - | - |
| 5 | 139 | 0.7701 | 0.5585 | 0.3037 | 0.0735 | 0.1396 | 0.5879 |
| 6 | - | - | - | - | - | - | - |
| 7 | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | 141 | 0.7772 | 0.5485 | 0.3316 | 0.0307 | 0.3258 | 0.5600 |
| Recorte 2 | | | | | | | |
| 0 | 279 | 0.3695 | 0.5926 | 0.6089 | 0.5867 | 0.3019 | 0.5358 |
| 1 | 280 | 0.3171 | 0.5957 | 0.5949 | 0.5536 | 0.1040 | 0.5320 |
| 2 | 252 | 0.3207 | 0.6039 | 0.8067 | 0.5083 | 0.0764 | 0.4904 |
| 3 | - | - | - | - | - | - | - |
| 4 | 252 | 0.3791 | 0.4580 | 0.8067 | 0.4824 | 0.1554 | 0.4151 |
| 5 | - | - | - | - | - | - | - |
| 6 | 252 | 0.4372 | 0.4835 | 0.8897 | 0.2508 | 0.1554 | 0.3524 |
| 7 | 271 | 0.3485 | 0.4766 | 0.8618 | 0.1839 | 0.0319 | 0.3476 |
| 8 | - | - | - | - | - | - | - |
| 9 | 280 | 0.3659 | 0.4643 | 0.7421 | 0.1792 | 0.1433 | 0.3073 |
| 10 | 244 | 0.4372 | 0.4534 | 0.7663 | 0.3244 | 0.0301 | 0.3006 |
| Recorte 3 | | | | | | | |
| 0 | 149 | 0.2641 | 0.2811 | 0.8115 | 0.5737 | 0.4247 | 0.6322 |
| 1 | - | - | - | - | - | - | - |
| 2 | 153 | 0.4250 | 0.3323 | 0.8312 | 0.3632 | 0.4685 | 0.5941 |
| 3 | - | - | - | - | - | - | - |
| 4 | - | - | - | - | - | - | - |
| 5 | - | - | - | - | - | - | - |
| 6 | - | - | - | - | - | - | - |
| 7 | - | - | - | - | - | - | - |

| Geração | Escala | Compactação | Cor | Peso das Bandas | | | RBSB |
|----------------|---------------|--------------------|------------|------------------------|--------|--------|-------------|
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | 153 | 0.5140 | 0.3149 | 0.8313 | 0.3615 | 0.4525 | 0.5522 |
| Recorte 4 | | | | | | | |
| 0 | 204 | 0.4725 | 0.6549 | 0.8616 | 0.2356 | 0.4652 | 0.7409 |
| 1 | - | - | - | - | - | - | - |
| 2 | 204 | 0.7617 | 0.6314 | 0.6705 | 0.5129 | 0.2953 | 0.5933 |
| 3 | - | - | - | - | - | - | - |
| 4 | - | - | - | - | - | - | - |
| 5 | - | - | - | - | - | - | - |
| 6 | - | - | - | - | - | - | - |
| 7 | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | 204 | 0.7262 | 0.6569 | 0.7743 | 0.1467 | 0.3042 | 0.5913 |
| 0 | 181 | 0.5391 | 0.4822 | 0.6573 | 0.6093 | 0.2806 | 0.6393 |
| Recorte 5 | | | | | | | |
| 1 | - | - | - | - | - | - | - |
| 2 | - | - | - | - | - | - | - |
| 3 | - | - | - | - | - | - | - |
| 4 | 179 | 0.5149 | 0.5771 | 0.7372 | 0.4354 | 0.3117 | 0.6056 |
| 5 | - | - | - | - | - | - | - |
| 6 | - | - | - | - | - | - | - |
| 7 | 177 | 0.7688 | 0.2994 | 0.8127 | 0.4471 | 0.4602 | 0.4491 |
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | - | - | - | - | - | - | - |
| Recorte 6 | | | | | | | |
| 0 | 245 | 0.4270 | 0.9487 | 0.6718 | 0.3550 | 0.1758 | 0.5664 |
| 1 | - | - | - | - | - | - | - |
| 2 | - | - | - | - | - | - | - |
| 3 | - | - | - | - | - | - | - |
| 4 | 245 | 0.7274 | 0.9499 | 0.5776 | 0.3314 | 0.5017 | 0.5462 |
| 5 | - | - | - | - | - | - | - |
| 6 | - | - | - | - | - | - | - |
| 7 | - | - | - | - | - | - | - |

| Geração | Escala | Compactação | Cor | Peso das Bandas | | | RBSB |
|----------------|---------------|--------------------|------------|------------------------|--------|--------|-------------|
| 8 | 245 | 0.3732 | 0.8310 | 0.6226 | 0.4013 | 0.1758 | 0.5281 |
| 9 | - | - | - | - | - | - | - |
| 10 | - | - | - | - | - | - | - |
| Recorte 7 | | | | | | | |
| 0 | 175 | 0.4474 | 0.2958 | 0.7703 | 0.5043 | 0.2163 | 0.5090 |
| 1 | 280 | 0.4159 | 0.5282 | 0.9102 | 0.5564 | 0.4313 | 0.4875 |
| 2 | - | - | - | - | - | - | - |
| 3 | - | - | - | - | - | - | - |
| 4 | 280 | 0.8058 | 0.3342 | 0.9060 | 0.5358 | 0.1389 | 0.4386 |
| 5 | - | - | - | - | - | - | - |
| 6 | - | - | - | - | - | - | - |
| 7 | - | - | - | - | - | - | - |
| 8 | 280 | 0.8166 | 0.3719 | 0.8069 | 0.6311 | 0.3385 | 0.4050 |
| 9 | 280 | 0.8350 | 0.3719 | 0.8900 | 0.4466 | 0.1389 | 0.3619 |
| 10 | - | - | - | - | - | - | - |
| Recorte 8 | | | | | | | |
| 0 | 135 | 0.1684 | 0.7978 | 0.7021 | 0.4386 | 0.3349 | 0.7062 |
| 1 | 197 | 0.3575 | 0.5428 | 0.6777 | 0.5952 | 0.1958 | 0.6928 |
| 2 | - | - | - | - | - | - | - |
| 3 | 198 | 0.2468 | 0.5189 | 0.7304 | 0.6144 | 0.0998 | 0.6649 |
| 4 | 199 | 0.6506 | 0.3879 | 0.6267 | 0.3989 | 0.4387 | 0.6507 |
| 5 | 199 | 0.3774 | 0.4998 | 0.7139 | 0.3867 | 0.3785 | 0.5643 |
| 6 | 196 | 0.6818 | 0.5051 | 0.6281 | 0.5412 | 0.1558 | 0.5625 |
| 7 | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | - | - | - | - | - | - | - |
| Recorte 9 | | | | | | | |
| 0 | 213 | 0.3165 | 0.6077 | 0.8173 | 0.7374 | 0.6172 | 0.5052 |
| 1 | - | - | - | - | - | - | - |
| 2 | - | - | - | - | - | - | - |
| 3 | - | - | - | - | - | - | - |
| 4 | - | - | - | - | - | - | - |
| 5 | 193 | 0.4448 | 0.6861 | 0.8484 | 0.1110 | 0.4129 | 0.4957 |
| 6 | - | - | - | - | - | - | - |
| 7 | 192 | 0.7025 | 0.4447 | 0.8076 | 0.2164 | 0.3047 | 0.4592 |

| Geração | Escala | Compactação | Cor | Peso das Bandas | | | RBSB |
|----------------|---------------|--------------------|------------|------------------------|--------|--------|-------------|
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | - | - | - | - | - | - | - |
| Recorte 10 | | | | | | | |
| 0 | 160 | 0.3360 | 0.9158 | 0.9821 | 0.2382 | 0.5364 | 0.7705 |
| 1 | 202 | 0.3830 | 0.8466 | 0.9645 | 0.0229 | 0.7750 | 0.7666 |
| 2 | - | - | - | - | - | - | - |
| 3 | - | - | - | - | - | - | - |
| 4 | - | - | - | - | - | - | - |
| 5 | - | - | - | - | - | - | - |
| 6 | 208 | 0.3830 | 0.8466 | 0.9645 | 0.0229 | 0.7750 | 0.7099 |
| 7 | - | - | - | - | - | - | - |
| 8 | - | - | - | - | - | - | - |
| 9 | - | - | - | - | - | - | - |
| 10 | 196 | 0.3830 | 0.8466 | 0.9645 | 0.0189 | 0.7800 | 0.6684 |

APÊNDICE B

Código para efetuar o cálculo do RBSB.

```

#Esta funcao calcula o Reference Boundary Segments Booster [1]
#A funcao avalia a qualidade de poligonos gerados com base em outros
poligonos de referencia

from PyQt4.QtGui import QDialog, QApplication, QWidget
from osgeo import ogr
from shapely.geometry import *
from shapely.wkb import loads
app = QApplication([''])
widget = QWidget()

def rbsb(path1='', path2=''):
    outdir=''
    # Primeira camada: os poligonos de referencia
    if path1 == '':path1=str(QFileDialog.getOpenFileName(widget, 'Abrir
shape 1', outdir, filter='Shapefile (*.shp)'))
    i=0
    if path1.find('/',i) != -1 :
        while path1.find('/', i) != -1:
            i = path1.find('/', i)+1
            outdir = path1[0:i]
        else:
            while path1.find('\\', i) != -1:
                i = path1.find('\\', i)+1
                outdir = path1[0:i]
    if path2 == '':path2=str(QFileDialog.getOpenFileName(widget, 'Abrir
shape 2', outdir, filter='Shapefile (*.shp)'))
    source1 = ogr.Open(path1)
    layer1 = source1.GetLayer()
    # Segunda camada: poligonos de teste
    source2 = ogr.Open(path2)
    layer2 = source2.GetLayer()
    inter = 0
    rbsb = 0
    rbsb_acc = 0
    # Buscar a geometria da referencia e comparar com o polígono com
maior interseccao
    # Retornando a soma das áreas de nao interseccao
    for element in layer1:
        geom = loads(element.GetGeometryRef().ExportToWkb())
        layer2.SetSpatialFilter(element.geometry())
        for element2 in layer2:
            geom2 = loads(element2.GetGeometryRef().ExportToWkb())
            if geom.intersection(geom2).area > inter:
                rbsb =
                (geom.difference(geom2).area+geom2.difference(geom).area)/geom.area
                inter = geom.intersection(geom2).area
                rbsb_acc += rbsb
            rbsb=0
            inter=0
            layer2.ResetReading()
    # dividir o total acumulado pelo numero de poligonos de referencia
    rbsb_acc /= layer1.GetFeatureCount()
    if rbsb_acc == 0:rbsb_acc=1000
    return rbsb_acc

if __name__ == '__main__':

```

```
from sys import argv
if len(argv)==3:
    print rbsb(argv[1], argv[2])
else:
    print rbsb()
```

#Referencia:

#1: CAZES, T. B.; COSTA, G. A. O. P.; FEITOSA, R. Q. Automatic evaluation of segmentation parameters. Simposio Brasileiro de Geomatica. Anais... Presidente Prudente, SP: 2007.

APÊNDICE C

Código do algoritmo genético

```
#####
##IMPORTAR BIBLIOTECAS
#####
from pygene.gene import FloatGene, IntGene
from pygene.organism import Organism, MendelOrganism
from pygene.population import Population
from rbsb import rbsb
from subprocess import call
from sys import exit
from sys import argv
from segbaatz import SegBaatz
from numpy import round as npround
from PyQt4 import QtGui
from gdal import Open as gdalOpen
from ogr import Open as ogrOpen
import atexit
import os
from time import clock
clock()
#
def exit_handler():
    print '\nUso: Imagem PolRef PolSaida [InicioEm] [Log] [Arq Populacao]
\n\n'
atexit.register(exit_handler)
#####
##DEFINIR VARIÁVEIS
#####
arr = [[None]]
try:
    n=int(argv[4])
except:
    n=1
try:
    img=argv[1] #imagem de referencia
    ref=argv[2] #poligonos de referencia
    seg=argv[3] #arquivos de saida
    seg2=seg[0:len(seg)-4]

###Verificar parametros de entrada e retornar erro caso nao estejam
definidos corretamente###
except:
    app = QtGui.QApplication([])
    wid = QtGui.QWidget()
    try:
        len(img)
    except:
        img=str(QtGui.QFileDialog.getOpenFileName(wid,"Imagem de
referencia", "", "tif (*.tif; *.tiff)"))
    try:
        len(ref)
    except:
        ref = str(QtGui.QFileDialog.getOpenFileName(wid,"Poligonos de
referencia", "", "Shapefile (*.shp)"))
        seg = str(QtGui.QFileDialog.getSaveFileName(wid,"Salvar resultados
como...", "", "Shapefile (*.shp)"))
if not os.access(img,os.R_OK):
    exit("\nERRO: Imagem Invalida")
if not os.access(ref,os.R_OK):
```

```

    exit("\nERRO: Poligonos invalidos")
if not os.access(os.path.dirname(seg),os.W_OK):
    exit("\nERRO: Nao e possivel acessar local de saida, verifique se o
diretorio existe")
try:
    logpath=argv[5]
except:
    logpath="D:/Caio/logbaatz.txt"
try:
    poplog=argv[6]
except:
    poplog="D:/Caio/poplog.txt"
###

###Definir variaveis da populacao###
numChild=6
imigration=0
popCull=3
initialPop=8
maxParents=2
###

#####
#FUNCOES DE LOG
#####
def writepop():
    xmlfile = open(poplog, "a")
    for j in range(0,len(pop)):
        xmlfile.write(str(pop[int(j)].genes).replace(':', ' ',
'*').replace('<', ' ').replace(':', ' ').replace('>', ' ').replace('*', ' ':
')+\n')
    xmlfile.write('\n')
    xmlfile.write(str(arr)+'\n')
    xmlfile.write('\n\n')
    xmlfile.close()

def writelog(what):
    file = open(logpath, 'a')
    file.write(what)
    file.close()

#
#####
##DEFINIR GENES (PARAMETROS)
#####
class comp(FloatGene):

    # valores maximos e minimos
    randMin = 0.0
    randMax = 1.0

    # probabilidade de mutacao
    mutProb = 0.4

    # grau de mutacao
    mutAmt = 0.2

class color(FloatGene):

    randMin = 0.2
    randMax = 1.0

```

```

mutProb = 0.4

mutAmt = 0.2

class weights(FloatGene):

    randMin = 0.0
    randMax = 1.0

    mutProb = 0.4

    mutAmt = 0.2

class weights1(FloatGene):

    randMin = 0.5
    randMax = 1.0

    mutProb = 0.4

    mutAmt = 0.2

class scale(FloatGene):

    randMin = 80
    randMax = 280

    mutProb = 0.4

    mutAmt = 0.05

#
#####
##DEFINIR ORGANISMO
#####
class Params(MendelOrganism):
    #####
    ##GENOMA
    #####
    genome = {'s':scale, 'w1':weights1, 'w2':weights, 'w3':weights,
'c':comp, 'col':color}
    #
    #####
    ##FUNCAO DE APTIDAO
    #####
    def fitness(self):
        global arr, ref, seg, img, n
        fit = None

w1,w2,w3,c,col,s=self['w1'],self['w2'],self['w3'],self['c'],self['col'],
self['s']
        s=int(round(s))
        ###Armazenar informacoes sobre o organismo em uma variável para
verificar se o RBSB já foi calculado
        for i in arr:
            if i[1:7] == [w1,w2,w3,c,col,s]:
                fit = i[0]
                break;
        if not fit:
            ###Se o RBSB ainda nao foi calculado realizar a segmentacao e
calcular o valor de fitness

```

```

        try:
            SegBaatz(img, seg2+str(n), str(s),
str(c),str(col),str(w1),str(w2),str(w3))
            fit = rbsb(ref,seg2+str(n)+".shp")
            ###Se ocorrer algum erro imprimir os dados do ultimo organismo
sendo testado e escrever a populacao no log
        except:
            print "scale: %s, compactness: %s, color: %s, weights: %s
%s %s" % (str(s), str(c), str(col), str(w1),str(w2), str(w3))
            writepop()
            exit()
            arr.insert(0,[fit, w1,w2,w3,c,col,s])
            print "%s %s %s %s %s %s %s\n" % (s,c, col,w1,w2,w3,fit)
            writelog("Individuo %s:\n%s %s %s %s %s %s %s\n" % (n, s,c,
col,w1,w2,w3,fit))
            n+=1
            print clock()
            return fit
#
#####
##FUNCAO DE REPRESENTACAO
#####
def __repr__(self):
    return "<Fitness=%f scale=%s compactness=%s color=%s
weights=%s,%s,%s>" % (
        self.fitness(), int(round(self['s'])), self['c'], self['col'],
self['w1'], self['w2'], self['w3'])
#
#####
##TENTAR CARREGAR POPULACAO DE ARQUIVO
#####
try:
    popfile = open(argv[5], 'r')
    pop = Population(species=Params, init=0, incest=maxParents)
    i = popfile.readline()[:-1]
    while len(i) > 1:
        org = Params()
        org.genes = eval(i)
        pop.add(org)
        i = popfile.readline()[:-1]
    i = popfile.readline()[:-1]
    if len(i)>1:
        arr = eval(i)
    if len(pop)==0:len(ThrowError)
    print "Populacao carregada do arquivo"
#
#####
##CRIAR NOVA POPULACAO
#####
except:
    pop = Population(species=Params, init=initialPop, incest=maxParents)
    print "Nova populacao"
#
#####
##FUNÇÃO PARA GERAR A POPULAÇÃO
#####
def main():
    global logpath, arr
    i=0
    if n != 1:
        i=((n+1-8)/popCull)

```

```

else:
    writelog("SCALE COMPACTNESS COLOR W1 W2 W3 FITNESS\n-----
-----"+str(i)+"-----\n")
    best = None
    try:
        while True:
            writelog("=====\n== Geracao: %s
==\n=====\n" % i)
            if i == 0:
                pass
            else:
                # Se houver imigracao adicionar os individuos
                for j in range(0,imigration):
                    pop.add(Params())
                    # Fazer uma geracao
                    pop.gen(popCull,numChild/2)
                # Adicionar o melhor membro da populacao a variavel
                best2 = pop.best()
                if best != None:
                    if best2.fitness() == best[0].fitness():
                        pass
                    else:
                        best = [best2,str(i)]
                else:
                    best = [best2,str(i)]
                # Apos a geracao imprimir o melhor individuo e em que geracao
                foi gerado
                print str(best2) + "\n Na geracao " + str(i)
                writelog("----MELHOR----\n" + str(best[0].fitness()) + " na
geracao " + str(best[1]) + "\n")
                i+=1
            if i > 10:
                writepop()
                writelog("\nTempo decorrido: %sm%ss" %
(int(clock()/60),int(round(clock()-int(clock()/60)*60))))
                writelog("\n\n")
                exit()

        except KeyboardInterrupt:
            writepop()
            writelog("\nTempo decorrido: %sm%ss" %
(int(clock()/60),int(round(clock()-int(clock()/60)*60))))
            writelog("\n\n")
            pass

if __name__ == '__main__':
    main()

```


APÊNDICE D

Filtro Kuwahara

```
#####
###IMPORTAR BIBLIOTECAS
#####
import gdal
import numpy
import sys
from PyQt4.QtCore import QApplication
gdal_datatypes={'binary':'Byte','uint8':'Byte','uint16':'UInt16','int32':'Int32'}
try:
    from osgeo import gdal
except ImportError:
    import gdal
from gdalconst import GA_ReadOnly, GDT_Float32

#####
###DEFINIR VARIÁVEIS
#####
def dofilter(input, refband, maxmemoryusage):
    layer = QgsLayers.getObjectFromUri(input)
    w = 5
    w2 = (w+1)/2
    memuse=int(maxmemoryusage)
    tif = gdal.Open(str(layer.source()), GA_ReadOnly )
    driver = tif.GetDriver()
    xsize = tif.RasterXSize
    ysize = tif.RasterYSize
    nbands = tif.GetRasterCount
    out = driver.Create(str(output), xsize, ysize, 4, gdal.GDT_Byte)
    out.SetGeoTransform(tif.GetGeoTransform())
    out.SetProjection(tif.GetProjection())
    band = [None]*nbands
    oband = [None]*nbands
    for i in range(0,nbands):
        band[i] = tif.GetRasterBand(i+1)
        oband[i] = out.GetRasterBand(i+1)
    nr=numpy.roll
    na=numpy.add
    band=nr(band,(1-refband))
    oband=nr(oband,(1-refband))
    refband2 = None
    #calcular numero de linhas a ler por vez
    readrows = int(((memuse-67)*48036)/xsize)/2
    #leitura inicial de quatro linhas de cada banda
    oband[0].WriteArray(numpy.repeat(0,xsize*2).reshape(2,-1),0,0)
    arr2 = band[0].ReadAsArray(0, 0, xsize, 4)
    arr3 = band[1].ReadAsArray(0, 0, xsize, 4)
    arr4 = band[2].ReadAsArray(0, 0, xsize, 4)
    arr5 = band[3].ReadAsArray(0, 0, xsize, 4)
    #ler as linhas da imagem
    for y in range(4, ysize, readrows):
        if ysize-y < readrows : readrows = ysize-y
        arr1 = band[0].ReadAsArray(0, y, xsize, readrows)
        #juntar ultimas 4 linhas lidas com as próximas (retirar efeito
da borda)
        arr1 = numpy.uint32(numpy.vstack((arr2,arr1)))
        arr2 = arr1[readrows:readrows+5,]
        #fazer o somatório dos pixels sobrepondo matrizes deslocadas
```

```

        sum=na(na(na(na(na(arr1,nr(arr1,1,1)),nr(arr1,-
1,1)),na(nr(nr(arr1,-1,1),-1,0),nr(nr(arr1,-
1,1),1,0))),na(nr(nr(arr1,1,1),-1,0),nr(nr(arr1,1,1),1,0))),na(nr(arr1,-
1,0),nr(arr1,1,0)))
        #calcular da matriz de referencia (para variancia)
        sum2=arr1**2
        #fazer o somatorio dos quadrados sobrepondo matrizes deslocadas
        sum2=na(na(na(na(na(sum2,nr(sum2,1,1)),nr(sum2,-
1,1)),na(nr(nr(sum2,-1,1),-1,0),nr(nr(sum2,-
1,1),1,0))),na(nr(nr(sum2,1,1),-1,0),nr(nr(sum2,1,1),1,0))),na(nr(sum2,-
1,0),nr(sum2,1,0)))
        #calcular a variancia
        sum2 = numpy.uint16(numpy.round((sum2-((sum**2)/9.0))/9.0))
        #calcular a media
        sum = numpy.ubyte(numpy.round((sum/9.0)+0.5))
        #deslocar as matrizes de soma quadrada
        sum23=nr(sum2,-2,1)
        #criar uma matriz de teste, cada linha sera uma matriz da
imagem linearizada

        t=numpy.vstack((sum2.flatten(),sum23.flatten(),nr(sum2,2,0).flatten()
,nr(sum23,2,0).flatten()))
        #escolher o menor valor de variancia colocando o valor 1 na
posicao da matriz de menor variancia
        t=t==numpy.min(t,0)
        #deslocar novamente a matriz
        sum23=nr(sum,-2,1)
        #multiplicar a matriz de teste pelas matrizes de medias,
retornando o unico valor diferente de zero em cada coluna do teste e
retornar a forma original da matriz

        arr1=nr(nr(numpy.ubyte(numpy.reshape(numpy.max((t)*numpy.vstack((sum.
flatten(),sum23.flatten(),nr(sum,2,0).flatten(),nr(sum23,2,0).flatten()))),0
),(readrows+4,-1))),-1,0),1,1)[2:(readrows+2),2:(xsize-2)]
        #escrever os resultados
        oband[0].WriteArray(arr1,2,y-2)
        progress.setPercentage(int((100*(y+readrows*(1/4)+4))/ysize))
        QCoreApplication.processEvents()
        #repete para a proxima banda, mas sem calcular qual a matriz de
menor variancia (sera baseada na banda de referencia apenas)
        for i in range(1,nbands):
            arr1 = band[i].ReadAsArray(0, y, xsize, readrows)
            arr1 = numpy.uint32(numpy.vstack((arr3,arr1)))
            arr3 = arr1[readrows:readrows+5,]

            sum=numpy.ubyte((na(na(na(na(na(arr1,nr(arr1,1,1)),nr(arr1,-
1,1)),na(nr(nr(arr1,-1,1),-1,0),nr(nr(arr1,-
1,1),1,0))),na(nr(nr(arr1,1,1),-1,0),nr(nr(arr1,1,1),1,0))),na(nr(arr1,-
1,0),nr(arr1,1,0))))/9.0+0.5)
            sum23=nr(sum,-2,1)

            arr1=nr(nr(numpy.ubyte(numpy.reshape(numpy.max((t)*numpy.vstack((sum.
flatten(),sum23.flatten(),nr(sum,2,0).flatten(),nr(sum23,2,0).flatten()))),0
),(readrows+4,-1))),-1,0),1,1)[2:(readrows+2),2:(xsize-2)]
            oband[1].WriteArray(arr1,2,y-2)

            progress.setPercentage(int((100*(y+readrows*((i+1)/nbands)+4))/ysize)
)
        QCoreApplication.processEvents()

#Fecha as imagens para gravar os resultados

```

```
out = None
tif = None
print "done..."
```