



UnB

UNIVERSIDADE DE BRASÍLIA

INSTITUTO DE GEOCIÊNCIAS

**USO DE *CO-CLUSTERING* PARA ANÁLISE DE IMAGENS DE ALTÍSSIMA
RESOLUÇÃO ESPACIAL**

Dissertação de Mestrado N° 199

Fabício de Almeida Santos

Orientadores: Prof. Dr. Edilson de Souza Bias;
Prof. Dr. Ricardo Seixas Brites (*in memoriam*)



UNIVERSIDADE DE BRASÍLIA

INSTITUTO DE GEOCIÊNCIAS

**USO DE *CO-CLUSTERING* PARA ANÁLISE DE IMAGENS DE ALTÍSSIMA
RESOLUÇÃO ESPACIAL**

*Dissertação de Mestrado submetida ao
Programa de Pós-Graduação em
Geociências Aplicadas e Geodinâmica,
como parte dos requisitos necessários à
obtenção do grau de Mestre em
Geociências Aplicadas.*

Brasília, 2022

FICHA CATALOGRÁFICA

de Almeida Santos, Fabrício

dF126u Uso de co-clustering para análise de imagens de altíssima
resolução espacial / Fabricio de Almeida Santos; orientador
Edilson de Souza Bias. -- Brasília, 2022.

116 p.

Dissertação (Mestrado - Mestrado em Geociências
Aplicadas) -- Universidade de Brasília, 2022.

1. Co-clustering. 2. mineração de dados. 3. morfologia
matemática. 4. textura. 5. classificação de imagens. I. de
Souza Bias, Edilson, orient. II. Título.

DEDICATÓRIA

À minha querida e amada esposa, Thaís, por tudo.

Aos meus filhos amados Isaac, na Pátria Espiritual, e Dante, por todo amor e vibrações excelentes.

À minha família, sempre presente, pelo apoio e compreensão.

A Deus, pela oportunidade da vida.

AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

Agradeço à UnB e ao Instituto de Geociências da UnB, pela oportunidade e altíssima qualidade de estudos ofertada.

Agradeço ao saudoso Professor Ricardo Seixas Brites que passou para a Pátria Espiritual no período de desenvolvimento do trabalho, por ter me acolhido como orientador inicialmente.

Agradeço ao dedicado e compreensivo professor Edilson de Souza Bias, por ter me acolhido como orientador já ao final do tempo destinado a este trabalho, após o falecimento do professor Brites, demonstrando vasto conhecimento técnico e capacidade de adaptação.

Agradeço ao ITC, *Faculty of Geo-Information Science and Earth Observation*, da *University of Twente*, dos Países Baixos, pelo conhecimento ofertado e pela motivação inicial deste trabalho.

Agradeço à ISPRS - *The International Society for Photogrammetry and Remote Sensing*, pelos insumos utilizados no desenvolvimento deste trabalho.

Agradeço ao Exército Brasileiro, à sua Diretoria de Serviço Geográfico, ao 2º Centro de Geoinformação e ao Instituto Militar de Engenharia, IME, bem como a todo seu corpo de militares e engenheiros pela excelente oportunidade, apoio técnico e institucional.

RESUMO

No contexto de mineração de dados, ao se realizar a classificação em imagens de sensoriamento remoto, a extração de padrões é um importante passo. O uso de *co-clustering* para análise de imagens agrega novas possibilidades de identificação de padrões, no ramo do conhecimento do sensoriamento remoto. Comumente se realiza a busca de padrões em imagens considerando-se cada dimensão por vez, portanto uma única banda. A proposta do uso de técnicas de *co-clustering* é justamente considerar-se, de forma iterativa, na dimensão espectral, todas as bandas da imagem original, além de camadas criadas, por exemplo de textura e morfologia matemática, simulando novas bandas de imagem. Ao final do processo, resulta-se em *clusters* de pixels efetivamente classificados. A partir de medidas de similaridade dadas pelos Índice de Jaccard, Índice de Rand e Índice de Rand Ajustado avaliaram-se os agrupamentos resultantes da técnica de *co-clustering* quando aplicada a um cubo de imagem gerado a partir de uma imagem RGB de altíssima resolução, concatenada a resultados de morfologia matemática e de análise de textura. Utilizou-se o método tradicional de classificação não supervisionada K-médias como base de comparação para avaliação dos resultados encontrados. Concluiu-se que o método é eficiente, desenvolvido a partir de imagens e classificação prévia, disponibilizadas pela ISPRS, classificação essa tratada como verdade para o contexto deste trabalho.

Palavras-chave: *Co-clustering*, mineração de dados, morfologia matemática, textura, altíssima resolução, classificação de imagens

ABSTRACT

In the context of data mining, when performing classification on remote sensing images, pattern extraction is an important step. The use of co-clustering for image analysis adds new possibilities for pattern identification in the field of remote sensing knowledge. Commonly, the search for patterns in images is performed considering each dimension individually at a time, therefore, a single band. The proposal for the use of co-clustering techniques is precisely to consider, in an iterative way, in the spectral dimension, all bands of the original image, in addition to created layers of texture and mathematical morphology, simulating new image bands. At the end of the process, effectively classified clusters of pixels are obtained. Based on similarity measures given by the Jaccard Index, Rand Index and Adjusted Rand Index, the resulting clusters of the co-clustering technique were evaluated when applied to an image cube generated from a very high resolution RGB image, concatenated to mathematical morphology and texture analysis results. The traditional method of unsupervised classification Kmeans was used as a basis for comparison to evaluate the results. It was concluded that the method is efficient, developed from images and previous classification, made available by the ISPRS, which was treated as true for the context of this work.

Keywords: Co-clustering, data mining, morphological profiles, texture, very high resolution, image classification

SUMÁRIO

FICHA CATALOGRÁFICA.....	2
DEDICATÓRIA	3
AGRADECIMENTOS	4
RESUMO	5
ABSTRACT.....	6
SUMÁRIO	7
LISTA DE FIGURAS	9
LISTA DE QUADROS	12
LISTA DE ABREVIÇÕES	13
1 INTRODUÇÃO.....	13
1.1 JUSTIFICATIVA E IMPORTÂNCIA DO PROJETO.....	13
1.2 HIPÓTESE.....	17
1.3 OBJETIVOS.....	17
2 REVISÃO BIBLIOGRÁFICA	18
2.1 MINERAÇÃO DE DADOS	18
2.2 CLASSIFICAÇÃO NÃO SUPERVISIONADA.....	19
2.3 CO-CLUSTERING	20
2.3.1 <i>Algoritmo escolhido para estudo</i>	22
2.4 MORFOLOGIA MATEMÁTICA.....	28
2.5 ANÁLISE DE TEXTURA	32
2.6 K-MÉDIAS	33
2.7 AVALIAÇÃO DE QUALIDADE	34
2.7.1 <i>Norma da Especificação Técnica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG)</i>	35
2.7.2 <i>Amostragem</i>	35
2.7.3 <i>Índice Kappa</i>	36
2.7.4 <i>Medidas de similaridade</i>	38
3 MATERIAL E MÉTODOS	41
3.1 SOFTWARES E APLICAÇÕES	41
3.2 ÁREA DE ESTUDO	42
3.3 ORIGEM DOS DADOS E INFORMAÇÕES.....	42
3.4 MÉTODO.....	44

3.4.1	<i>Código para estudo de morfologia matemática</i>	45
3.4.2	<i>Código para análise de textura</i>	46
3.4.3	<i>Etapas e métodos de classificação</i>	47
3.4.4	<i>Etapa 1: classificação K-médias</i>	48
3.4.5	<i>Etapa 2: classificação co-clustering</i>	49
3.4.6	<i>Etapa 3: classificação co-clustering com refinamento por K-médias</i>	50
3.4.7	<i>Etapa 4: avaliação</i>	50
4	DISCUSSÃO E APLICAÇÃO DO MÉTODO	51
4.1	EXEMPLO DE APLICAÇÃO DO MODELO PROPOSTO	52
4.2	RESULTADO	58
4.2.1	<i>Comparação de resultados – Índice de Jaccard</i>	59
4.2.2	<i>Comparação de resultados – Índice de Rand</i>	61
4.2.3	<i>Comparação de resultados – Índice de Rand Ajustado</i>	62
4.2.4	<i>Análise conjunta dos resultados</i>	64
4.2.5	<i>Avaliação quanto ao cubo de imagem integrado com morfologia matemática e textura</i>	75
4.2.6	<i>Imagens classificadas pelos três métodos</i>	79
4.2.7	<i>Análise estatística dos resultados quanto à significância da melhora do resultado</i>	117
5	CONCLUSÕES	121
	REFERÊNCIAS BIBLIOGRÁFICAS	123
	APÊNDICE 1	134
	APÊNDICE 2	143
	APÊNDICE 3	148
	APÊNDICE 4	150
	APÊNDICE 5	152
	APÊNDICE 6	154
	APÊNDICE 7	156

LISTA DE FIGURAS

Figura 1 - interpretação geométrica da divergência de Bregman no R^2	24
Figura 2 - Exemplo de representação de cubo de imagem	26
Figura 3 - matriz remodelada a partir do cubo de imagem.....	27
Figura 4 - exemplo de erosão de uma imagem	30
Figura 5 - exemplo de dilatação de uma imagem	30
Figura 6 - exemplo de operador opening	31
Figura 7 - exemplo de operador closing	31
Figura 8 - exemplo imagem formada a partir de cálculo de entropia local	33
Figura 9 - mosaico de ortofotos da região de Postdam, Alemanha	43
Figura 10 - Fluxograma representando o processo de construção do cubo	44
Figura 11 - camadas de morfologia matemática.....	46
Figura 12 - fluxograma representando as etapas do trabalho	48
Figura 13 - ortofoto teste	53
Figura 14 - imagem teste	54
Figura 15 - representação de cada pixel classificado da imagem teste.....	55
Figura 16 - representação de cada classe da imagem teste	56
Figura 17 - clusters de bandas resultante do co-clustering	57
Figura 18 – figura da esquerda representa a imagem teste classificada pelo método co-clustering; figura da direita representa imagem teste "verdade" obtida por classificação supervisionada, disponibilizada pela ISPRS (ISPRS, 2022)	57
Figura 19 - Variação do k-valor para o método Co-clustering refinado.....	59
Figura 20 – distribuição da média do Índice de Jaccard para os diferentes métodos: (a) K-médias; (b) Co-clustering; (c) Co-clustering refinado	60
Figura 21 - distribuição da média do Índice de Rand para os diferentes métodos: (a) K-médias; (b) Co-clustering; (c) Co-clustering refinado	61
Figura 22 - distribuição da média do Índice de Rand Ajustado para os diferentes métodos: (a) K-médias; (b) Co-clustering; (c) Co-clustering refinado	62
Figura 23 - Distribuição em intervalos para os métodos Co-clustering e Co-clustering refinado	63
Figura 24 - Resultado dos índices, por método, para todos as ortofotos.....	64

Figura 25 - distribuição de máximo, mínimo e média dos índices considerando todas as ortofotos.....	66
Figura 26 - Comparativo Percentual do melhor método a partir das medidas de similaridade consideradas.....	68
Figura 27 - Comparativo de melhora em relação ao método K-médias - Índice de Jaccard.....	70
Figura 28 - Comparativo de melhora em relação ao método K-médias - Índice de Rand.....	72
Figura 29 - Comparativo de melhora em relação ao método K-médias - Índice de Rand Ajustado.....	74
Figura 30 - ortofoto 2_10.....	80
Figura 31 - ortofoto 2_11.....	81
Figura 32 - ortofoto 2_12.....	82
Figura 33 - ortofoto 2_13.....	83
Figura 34 - ortofoto 2_14.....	84
Figura 35 - ortofoto 3_10.....	85
Figura 36 - ortofoto 3_11.....	86
Figura 37 - ortofoto 3_12.....	87
Figura 38 - ortofoto 3_13.....	88
Figura 39 - ortofoto 3_14.....	89
Figura 40 - ortofoto 4_10.....	90
Figura 41 - ortofoto 4_11.....	91
Figura 42 - ortofoto 4_12.....	92
Figura 43 - ortofoto 4_13.....	93
Figura 44 - ortofoto 4_14.....	94
Figura 45 - ortofoto 4_15.....	95
Figura 46 - ortofoto 5_10.....	96
Figura 47 - ortofoto 5_11.....	97
Figura 48 - ortofoto 5_12.....	98
Figura 49 - ortofoto 5_13.....	99
Figura 50 - ortofoto 5_14.....	100
Figura 51 - ortofoto 5_15.....	101

Figura 52 - ortofoto 6_7.....	102
Figura 53 - ortofoto 6_8.....	103
Figura 54 - ortofoto 6_9.....	104
Figura 55 - ortofoto 6_10.....	105
Figura 56 - ortofoto 6_11.....	106
Figura 57 - ortofoto 6_12.....	107
Figura 58 - ortofoto 6_13.....	108
Figura 59 - ortofoto 6_14.....	109
Figura 60 - ortofoto 6_15.....	110
Figura 61 - ortofoto 7_7.....	111
Figura 62 - ortofoto 7_8.....	112
Figura 63 - ortofoto 7_9.....	113
Figura 64 - ortofoto 7_10.....	114
Figura 65 - ortofoto 7_11.....	115
Figura 66 - ortofoto 7_12.....	116
Figura 67 - ortofoto 7_13.....	117

LISTA DE QUADROS

Quadro 1 - Distribuição em intervalos para os métodos Co-clustering e Co-clustering refinado	63
Quadro 2 - Estatísticas dos índices considerando todas as ortofotos.....	65
Quadro 3 - Comparativo de melhora em relação ao método K-médias - Índice de Jaccard	69
Quadro 4 - Comparativo de melhora em relação ao método K-médias - Índice de Rand.....	71
Quadro 5 - Comparativo de melhora em relação ao método K-médias - Índice de Rand Ajustado	72
Quadro 6 - Comparativo de melhora em relação ao cubo de imagem original - método co-clustering sem refinamento.....	75
Quadro 7 - Comparativo de melhora em relação ao cubo de imagem original - método co-clustering com refinamento	76
Quadro 8 – Média, Desvio padrão e Variância dos índices, por método, para todas as ortofotos	119
Quadro 9 - resultados dos índices para o método K-médias.....	148
Quadro 10 – resultados dos índices para o método co-clustering sem refinamento	150
Quadro 11 – resultados dos índices para o método co-clustering com refinamento	152
Quadro 12 – resultados dos índices para o método co-clustering sem refinamento	154
Quadro 13 – resultados dos índices para o método co-clustering com refinamento	156

LISTA DE ABREVIÇÕES

- DSG – Diretoria de Serviço Geográfico
- ET-PCDG - Especificação Técnica para Produtos de Conjunto de Dados Geoespaciais
- INDE - Infraestrutura Nacional de Dados Espaciais
- ET-CQDG - Especificação Técnica para Controle de Qualidade de Dados Geoespaciais
- ISO – International Organization for Standardization
- EGB – Espaço Geográfico Brasileiro
- CONCAR - Comissão Nacional de Cartografia
- MPOG - Ministério do Planejamento, Orçamento e Gestão
- SVM – *Support-vector machine*
- OBIA – *Object-Based Image Analysis*
- ISODATA - *Iterative Self-Organizing Data Analysis Technique*
- BCC - *Bregman Co-clustering*
- ITCC - *Information-Theoretic Co-clustering*
- SCC - *Spectral Co-clustering*
- UTM - *Universal Transversa de Mercator*
- ROI – *region of interest*
- CGC – *Clustering Geo-data Cubes*
- SIG – Sistemas de Informações Geográficas
- ISPRS - *The International Society for Photogrammetry and Remote Sensing*
- TIF – *Tagged Image File*
- GSD - *ground sampling distance*

1 INTRODUÇÃO

1.1 JUSTIFICATIVA E IMPORTÂNCIA DO PROJETO

No âmbito da análise e do processamento digital de imagens, a identificação de alvos e feições se mostra com importância destacada, principalmente em sistemas de defesa e segurança. Nesse contexto, o conhecimento geoespacial é fundamental no apoio tático e estratégico às operações dos sistemas de defesa, oferecendo, sobretudo, informações espaciais confiáveis, para tomada de decisão das autoridades competentes. No Brasil, o Exército Brasileiro, por meio de sua Diretoria de Serviço Geográfico, a DSG, tem o encargo de elaborar Normas Técnicas para o Sistema Cartográfico Nacional “no que concerne às séries de cartas gerais, das escalas de 1:250.000 e maiores”, conforme Decreto-Lei 243, de 28/02/1967, Art. 15, §1º, item 2 (BRASIL, 1967).

Com o avançar permanente da tecnologia de aquisição de imagens, a resolução espacial segue, cada vez mais, atingindo níveis mais avançados de detalhamento. Como consequência desse avanço, tornam-se mais disponíveis ao usuário as imagens de altíssima resolução, aquelas que serviriam de insumo para escalas cadastrais, ou Grandes Escalas, conforme apresentado na Norma para Especificação Técnica para Produtos de Conjunto de Dados Geoespaciais (ET-PCDG) (DSG, 2016a). Cabe destacar que a ET-PCDG segue definições de normas internacionais, especialmente as padronizações ISO (ISO, *[s. d.]*), mais especificamente a norma ISO 19131 (ISO, 2007).

No campo do Sensoriamento Remoto, mais especificamente na fase operacional de aquisição de imagens, com o avançar tecnológico, percebe-se o surgimento de opções cada vez mais rápidas, de custo acessível e logística facilitada para imageamento em altíssima resolução, como por exemplo o uso de Sistemas Remotamente Pilotados para o referido fim.

De forma geral, no âmbito da pesquisa e desenvolvimento de processos para o estudo dessa categoria de insumo geoespacial, o foco central encontra-se no modelo escolhido para classificação da imagem. A fim de se desenvolverem métodos de processamento de imagens, a decisão pelos insumos passa principalmente pela definição da resolução espacial da imagem, não dependendo para determinados processos, necessariamente qual seja o sistema imageador.

De forma específica, no âmbito do Exército Brasileiro, há uma demanda por tecnologias e processos de análise voltados ao estudo de imagens compatíveis às adquiridas por câmeras embarcadas em sistemas imageadores de uso operacional, como DRONES, de forma geral considerados como os referidos Sistemas Remotamente Pilotados. De forma ainda mais específica, um dos desafios atuais do Exército Brasileiro se refere ao processamento de imagens de altíssima resolução espacial, de forma rápida, de fácil operação, com custo acessível, geralmente de áreas específicas e relativamente pequenas. Neste contexto, o processamento das imagens RGB de altíssima resolução espacial, para geração de produtos na escala cadastral, é foco deste trabalho.

No âmbito da geração sistemática de produtos geoespaciais, considerando-se a sequência tradicional de etapas para mapeamento e extração de informação geoespacial de referência a partir das citadas imagens de altíssima resolução, a etapa da classificação de imagens mostra-se como parte essencial do processo. Destaca-se, como resultado direto da etapa de classificação, a geração de mapas de uso e ocupação do solo.

No momento atual, encontramos diversas técnicas de classificação de imagens, como o uso de SVM, do inglês *support-vector machine*, *Random Forest*, *Deep Learning* e uma grande aplicação de estudos que seguem abordagem de Análise de Imagem Baseada em Objeto, do inglês *Object-Based Image Analysis*, OBIA.

Tais métodos têm sido utilizados em diversas áreas do conhecimento, com aplicação de sensoriamento remoto, considerando-se, na construção do algoritmo, a vizinhança espacial pixel a pixel, na formação dos agrupamentos no processo de classificação da imagem.

No caso da abordagem SVM, há na literatura trabalhos que apresentam sucesso a partir da sua aplicação para classificação de imagens, com destaque para aplicação em casos em que há sobreposição de classes por confusão espectral (BROWN; LEWIS; GUNN, 2000). Sua principal limitação é se tratar de técnica supervisionada baseada na análise estatística de regressão (NASCIMENTO *et al.*, 2009). Desta forma, faz-se necessária intervenção direta do operador no processo de classificação da imagem.

Em se tratando da abordagem *Random Forest*, amplamente utilizado em Sensoriamento Remoto, foi apresentada como similarmente efetiva em comparação ao método SVM, nos quesitos acurácia e tempo de treinamento (PAL, 2005). De forma

similar ao apresentado para SVM, trata-se de método de classificação supervisionado, que exige intervenção do operador no processo.

Métodos que tem por base inteligência artificial, por intermédio de processos de aprendizado de máquina e aprendizagem profunda, são dependentes na sequência de trabalho de etapa que envolve considerável volume de amostras de treinamento.

No caso da abordagem OBIA, a segmentação de imagens é importante passo para posterior classificação pelo referido método. Com o avançar da tecnologia e advento das imagens de alta resolução espacial, a identificação de alvos ou objetos surge como novo objetivo (HOSSAIN; CHEN, 2019).

A abordagem OBIA, no entanto, apresenta suas peculiaridades, dentre elas destaca-se a necessidade de intervenção manual para determinação de feições ou classes de uma imagem, característica comum em processos de classificação supervisionada de imagens. Como consequência da referida necessidade de intervenção manual, tem-se potencialmente um alto custo de pessoal para realização do processo, bem como a dependência da realização de uma boa segmentação por parte do operador. Ainda no campo de estudo de métodos OBIA, há trabalhos que propõem integração da referida técnica baseada em objetos com a tecnologia *deep learning*, a fim de buscar se superar tais limitações (WANG *et al.*, 2021).

No âmbito das diversas abordagens tradicionais de agrupamento, ou *clustering*, em inglês, nota-se constante limitação de processamento pois em sua maioria as abordagens consideram cada dimensão separadamente, seja ela espacial, temporal ou espectral. Um exemplo disso seria classificar os pixels de uma determinada banda pré-selecionada, não considerando a dimensão espectral como um todo, ou seja, todas as bandas da imagem original. Seguindo-se as etapas de processamento para classificação, tem-se a seleção prévia de bandas espectrais. Percebe-se a necessidade da referida seleção seja em abordagens tradicionais, seja em modelos que usam a abordagem OBIA.

A referida seleção prévia de bandas espectrais se mostra com maior importância à medida em que se aumenta a dimensão z , das bandas originais, por exemplo para o estudo de imagens multiespectrais, mais ainda para o estudo de imagens hiperespectrais. Tradicionalmente, uma primeira fase de trabalho consiste em explorar os grupos de bandas da imagem, conforme critério previamente bem definido. Desta forma tem-se um facilitador da etapa de seleção de bandas no processo de classificação de imagens. Tal

estudo pode ser realizado, por exemplo, a partir do estudo da análise por componentes principais, que permite avaliação quanto a correlação entre as diversas bandas de uma imagem, justamente com o objetivo de diminuir a redundância, portanto a dimensão espectral da imagem (WATRIN; FILHO; SANTOS, 1996).

Ao se superar a necessidade de seleção prévia de bandas espectrais para classificação de imagens, amplia-se o escopo dos estudos, gerando-se informação a partir de um maior volume de dados, uma vez que se consideraria mais de uma banda no referido processo de classificação. Dessa forma, a partir da análise multivariada, portanto considerando-se mais de uma dimensão simultaneamente, estuda-se maior quantidade de variáveis, simultaneamente.

Para avaliação de acurácia da classificação de produtos de sensoriamento remoto, pode se dividir tal acurácia em posicional e temática. Diversos são os fatores que podem influenciar em seu resultado e causar erro, seja de omissão ou de comissão (CONGALTON; GREEN, 2008).

Estudos recentes apontam que a análise e processamento de imagens de altíssima resolução espacial podem potencialmente sofrer impacto negativo no que se refere a acurácia temática da classificação devido a resolução espacial ser muitas vezes mais detalhada que o próprio objeto ou feição a ser identificada na imagem (HU *et al.*, 2020). Desse modo, em se tratando de imagens de altíssima resolução não se deve considerar apenas as informações espectrais, mas sim considerar, em conjunto, informações como textura e o contexto espacial (HU *et al.*, 2020), como verificado no uso de morfologia matemática.

Do ponto de vista científico, a automação para reconhecimento de padrões nos dados de sensoriamento remoto é um desafio da atualidade, a fim de se facilitar o processamento dos referidos dados. O desenvolvimento de processos automáticos, não supervisionados, no campo do Sensoriamento Remoto, mais especificamente na fase classificação de imagem, potencialmente fará aumentar a capacidade dos modelos em volume, velocidade e precisão, por exemplo.

O uso de *co-clustering* para análise de imagens agrega novas possibilidades de identificação de padrões no ramo do conhecimento do sensoriamento remoto. A referida técnica permite a análise de grande volume de dados e se mostra como possibilidade para

análise de imagens sem necessidade de treinamento prévio nem mesmo de seleção prévia de bandas espectrais.

A partir da motivação apresentada, este trabalho se insere no âmbito do desenvolvimento de conhecimento sobre a exploração de imagens de altíssima resolução espacial. Para tal, busca-se o incremento da eficácia da classificação de pixels por intermédio de estratégia integrada de análise por *co-clustering*, textura e morfologia matemática.

Este trabalho inova ao propor levar em consideração a dimensão *z* para a aplicação da técnica de *co-clustering*, em vez da abordada na literatura, em que se considera especificamente a dimensão temporal (ZURITA-MILLA *et al.*, 2020).

Neste contexto, a fim de se incrementar a dimensão *z* do cubo de imagem original RGB, explora-se o uso de análise de morfologia matemática e análise de textura a partir da imagem original, de forma a se gerar novas camadas de informações.

Especificamente, dentro do âmbito do Serviço Geográfico do Exército Brasileiro, cujos serviços relacionados à Geoinformação são responsabilidade da DSG, há um constante desafio de mapeamento para produtos de representação de uso do solo, na fase de classificação de imagens e posterior fase de vetorização automática de feições, devido a necessidade de retrabalho da equipe de revisão.

A fim de se obter um objetivo claro, dentro das possibilidades das imagens de altíssima resolução disponíveis e dentro do previsto e estabelecido nas normas, este trabalho tem por escopo o mapeamento cadastral, mais especificamente a escala 1:1000, compatível com a resolução espacial das imagens utilizadas.

1.2 HIPÓTESE

A hipótese científica proposta é: A aplicação da técnica de *co-clustering* pode melhorar os resultados de classificação não supervisionada, considerando a dimensão espectral, na busca por padrões com o uso de cubos de imagens.

1.3 OBJETIVOS

O objetivo geral desta pesquisa é:

Propor, testar e aplicar uma técnica de *co-clustering*, considerando a dimensão z , das bandas originais, com o incremento de camadas de morfologia matemática e análise de textura para aplicação em imagens de altíssima resolução espacial.

Para tal, seguir-se-á os seguintes objetivos específicos:

1. Adaptar, para o contexto específico das imagens RGB de altíssima resolução espacial, as abordagens existentes de *co-clustering*, com o incremento de camadas de morfologia matemática e análise de textura;
2. Integrar em abordagem única, por intermédio de código integrado, técnica de *co-clustering*, aplicada a cubo de imagem proveniente de resultados de extração de atributos de morfologia matemática e análise de textura, quando aplicados a imagens RGB de altíssima resolução espacial;
3. Avaliar matematicamente, por intermédio de medidas de similaridade, os agrupamentos resultantes da aplicação da técnica integrada de *co-clustering*;
4. Avaliar matematicamente o impacto quanto as medidas de similaridade ao se utilizar do incremento de camadas de morfologia matemática e textura para aplicação da técnica de *co-clustering*;
5. Identificar padrões de similaridade dentre os agrupamentos resultantes da aplicação da técnica de *co-clustering*, por intermédio da técnica de refinamento K-médias;
6. Aplicar método de classificação não supervisionada K-médias e avaliar a diferença de resultado em comparação com o método *co-clustering* proposto.

2 REVISÃO BIBLIOGRÁFICA

2.1 MINERAÇÃO DE DADOS

O estudo e desenvolvimento da mineração de dados remonta inicialmente à década de 1960 e segue se desenvolvendo ainda nos dias de hoje, culminando naturalmente em processos mais avançado de análise de dados tais como mineração de dados (HAN; KAMBER; PEI, 2011).

Processos de mineração de dados buscam, a partir do extenso volume de dados, organizar e extrair padrões que sejam pertinentes à tomada de decisão almejada. Face a grande quantidade de dados e informações que se apresenta no campo das geociências, a busca de soluções no campo do geoprocessamento se insere no conceito de mineração de dados. Destaca-se o permanente crescimento do volume dos referidos dados e consequente necessidade e desenvolvimento de novas técnicas e abordagens para tratamento dos mesmos.

No campo do sensoriamento remoto são diversas as aplicações da mineração de dados que buscam melhorar os processos de agrupamento de informações, mais especificamente de pixels, seja para problemas de classificação de imagens ou ainda para detecção de objetos.

É o caso de estudos que abordam *Object-Based Image Analysis*, OBIA, análise de imagens orientada a objeto, em português, no contexto integrado de mineração de dados aplicado a mapeamento urbano a partir de imagens de satélite de alta resolução espacial (HAMEDIANFAR; SHAFRI, 2016), por exemplo.

Nota-se ampla gama de estudos na área envolvendo a abordagem OBIA, que evidencia sua eficiência quando se estuda, tradicionalmente, bandas previamente selecionadas de um cubo de imagem. No entanto, para tal abordagem, faz-se necessária prévia seleção de banda para aplicação do processo e posterior análise do resultado, uma vez que a referida abordagem não considera todas as bandas originais da referida imagem.

2.2 CLASSIFICAÇÃO NÃO SUPERVISIONADA

A classificação não supervisionada organiza as classes a partir de agrupamentos, denominados como *clusters*, obtidos matematicamente, e possui caráter exploratório dos dados, uma vez que os agrupamentos obtidos podem não coincidir com as classes da verdade de campo. Processos de classificação não supervisionada tem cada vez mais sido tema de trabalhos acadêmicos na área de Sensoriamento Remoto, uma vez que permite a classificação, pixel a pixel, da imagem sem a necessidade da participação direta do operador no processo de classificação, mais especificamente na etapa de treinamento do método, necessária nos processos tradicionais de classificação supervisionada.

A técnica de *co-clustering* se insere no contexto geral de mineração de dados, em processos de classificação não supervisionada, no campo do sensoriamento remoto, a fim

de, potencialmente, permitir a análise multivariada de dados, segundo diferentes critérios, ou dimensões, seja espacial, temporal ou espectral, por exemplo. Dessa forma supera-se a necessidade já apresentada de prévia seleção de bandas e prévio treinamento de amostras para processos de classificação de imagens.

Destaca-se como método tradicional de classificação não supervisionada, o método K-médias, que realiza a referida classificação sem a necessidade de treinamento prévio, o que torna os resultados coerentes aos encontrados a partir da técnica *co-clustering*.

Um outro método de classificação não supervisionada de amplo uso é o método *Iterative Self-Organizing Data Analysis Technique*, ISODATA, aplicado para classificação de imagem de cobertura de solo (WASEEM *et al.*, 2016). Devido o referido método não considerar a variabilidade interna do agrupamento resultante da classificação, limitou-se no escopo deste trabalho ao uso do método K-médias, como método base de comparação para posterior avaliação dos resultados. (MENESES *et al.*, 2012)

2.3 CO-CLUSTERING

Co-clustering trata de simultaneamente agrupar linhas e colunas de uma matriz de dados. O *co-clustering* frequentemente se confunde com *bi-clustering* (CHENG; CHURCH, 2000), por vezes apresentado na literatura seja para agrupar simultaneamente linhas e colunas de uma matrizes, semelhantemente ao *co-clustering*, no entanto por vezes apresentado com relação a técnicas de agrupamento aplicadas em um subconjunto de dados (DIAZ; PERES, 2019).

De forma a acompanhar o estado da arte, é utilizado, no contexto deste estudo, o termo *co-clustering*, em inglês, em vez do termo co-agrupamento, traduzido para o português, uma vez que o termo em português não é amplamente difundido, principalmente no campo do sensoriamento remoto.

Abordagens que utilizam *co-clustering* surgem como alternativa a métodos tradicionais de agrupamento de dados, os quais consideram cada dimensão de informação separadamente, como é o caso de abordagem k-médias (PAPALEXAKIS; SIDIROPOULOS; BRO, 2013). Isso se dá pois aplicações de mineração de dados com base em *co-clustering* têm elevado potencial de agrupar dados segundo diferentes critérios, ou dimensões, simultaneamente, como se pode concluir de diversos estudos

especializados na área de estudo, tais como (IZQUIERDO-VERDIGUIER; ZURITAMILLA, 2015) e (CHENG; CHURCH, 2000).

A proposta de se buscar padrões considerando-se mais de uma dimensão de informações iniciou-se, na década de 1970, com estudos acerca da distribuição de votos por estado, por ano, para presidente dos Estados Unidos (HARTIGAN, 1972).

Desde o surgimento dos estudos de abordagens de busca e análise multivariada, mais especificamente estudos com base em *co-clustering* são amplamente aplicados em computação, como no desenvolvimento de técnicas de fatoração matricial (SHANG; JIAO; WANG, 2012) que, por fim, particionam o dado de entrada em partes menores, considerando diferentes dimensões de informação. Abordagens de *co-clustering* tem sido amplamente investigadas tais como em estudos na busca por padrões em genes, ou ainda na busca por padrões de palavras em textos e documentos, conforme apresentado em (BANERJEE *et al.*, 2007), ou ainda busca por padrões de texto, imagem e usuários em redes sociais a fim de eliminar possíveis informações irrelevantes (QIAN *et al.*, 2019).

No campo computacional, em estudos que utilizam redes neurais para agrupamento de informações, surgem abordagens que utilizam técnicas de *co-clustering* a fim de melhorar a performance dos processos tradicionais (DU *et al.*, 2021). A abordagem *co-clustering* tem sido utilizada como estado da arte para rastrear múltiplos objetos em problemas de segmentação de movimento, ao se explorar o movimento e trajetória de objetos ao longo do tempo, a partir de múltiplas imagens, seja para agrupar pixels, seja para detectar objetos (KEUPER *et al.*, 2020).

Co-clustering foi também utilizado no campo da matemática teórica, como estratégia para integrar informações complementares no que diz respeito a escolha automática de pesos matemáticos ao se se integrar diferentes métodos de agrupamento de dados e seus pesos conhecidos (HU; YAN; YE, 2020).

Por vezes, no campo do processamento digital de imagens, quando dispõe de uma imensa variedade de imagens relativas à mesma área ou ao mesmo objeto, o analista depara-se com múltiplas observações, por vezes rotacionadas do mesmo objeto. Neste contexto, técnicas de *co-clustering* tem sido também utilizadas para realizar a detecção de objetos, com necessidade reduzida de intervenção humana no processo de separação das imagens (YAO *et al.*, 2017).

Em linhas gerais, no contexto de mineração de dados, ao se trabalhar com enorme quantidade de dados, o processo de *co-clustering* apresenta-se como especialmente útil, uma vez que resulta em informações agrupadas e interpretáveis, preservando as informações do dado original (BANERJEE *et al.*, 2007), entre tantas outras aplicações, tais como busca por padrões em séries temporais de imagens (WU *et al.*, 2020).

A teoria baseada em co-clusterização foi apresentada como boa ferramenta para selecionar bandas para classificação em imagens hiperespectrais (IZQUIERDO-VERDIGUIER; ZURITA-MILLA, 2015), trabalho esse que se apresenta como a primeira aplicação da técnica no campo do sensoriamento remoto. A referida teoria mostrou-se especialmente útil quando se lida com grande quantidade de dados multivariados (WU; ZURITA-MILLA; KRAAK, 2015).

De forma simplificada, considerando a notação apresentada por (DIAZ; PERES, 2019), adaptada de (PENSA *et al.*, 2010), tem-se a seguinte notação matemática para o entendimento da técnica de *co-clustering*:

Seja uma matriz X de dimensão $m \times n$ ($X \in \mathbb{R}^{m \times n}$), seja x_{ij} o elemento correspondente à linha i e coluna j . Sejam \vec{x}_i e \vec{y}_j os vetores associados respectivamente às linhas i e colunas j . Um resultado de *co-clustering* $C^{k \times l}$ aplicado na matriz X produz simultaneamente um conjunto de $k \times l$ blocos, ou seja, uma partição em k grupos de linhas associada a uma partição em l grupos de colunas, que otimizam uma determinada função objetivo.

2.3.1 Algoritmo escolhido para estudo

A solução buscada em algoritmos de *co-clustering* é comumente abordada em diversas aplicações de mineração de dados, ou mineração de dados, e diversos são os estudos que compõem o estado da arte relativo a *co-clustering*.

Conforme apresentado em (GOVAERT; NADIF, 2014), há diferentes critérios estatísticos na composição de algoritmos baseados em *co-clustering*, sendo os algoritmos mais comuns os chamados *Bregma Co-clustering* (BCC), *Information-Theoretic Co-clustering* (ITCC) e *Spectral Co-clustering* (SCC). O referido trabalho de (HUANG *et al.*, 2015) propõe um abordagem agregando-se os três algoritmos numa estratégia única e apresenta uma análise estatística dos resultados.

Para o contexto deste trabalho, seguiu-se o algoritmo eleito por (IZQUIERDO-VERDIGUIER; ZURITA-MILLA, 2015), sendo o algoritmo *Bregman block-average clustering* (BBAC) (BANERJEE *et al.*, 2007), devido a sua capacidade potencial de ser aplicado em matrizes de grandes dimensões, sendo esse o caso quando se trabalha com imagens. O referido algoritmo BBAC é uma generalização do Information Theory Co-clustering algorithm (ITCC).

O algoritmo BBAC parte do cálculo da divergência de Bregman, que determina a similaridade entre dois pares de números, vetores ou matrizes (BANERJEE *et al.*, 2007). Define-se divergência de Bregman, conforme notação apresentada em (FERREIRA, 2017):

Dada uma função convexa $\varphi: \mathbb{R}^d \rightarrow \mathbb{R}$, a divergência de Bregman entre x e $y \in \text{dom}(\varphi)$ é tal que:

$$d_{\varphi}(x, y) = \varphi(x) - \varphi(y) - [\nabla\varphi(y), x - y] \tag{1}$$

em que $\nabla\varphi(y)$ é o vetor gradiente de φ em y .

A interpretação geométrica da divergência de Bregman no \mathbb{R}^2 , dada pela equação (1), conforme apresentada por (FERREIRA, 2017), é dada pela Figura 1. Nota-se, em azul, a divergência de Bregman $d_{\varphi}(x, y)$.

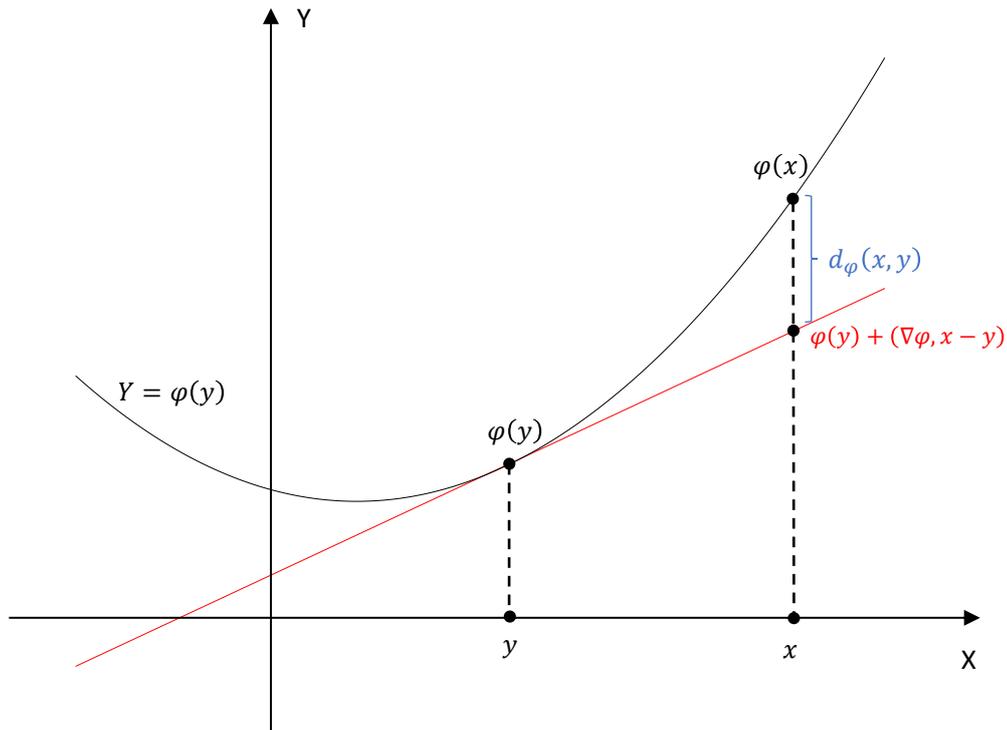


Figura 1 - interpretação geométrica da divergência de Bregman no \mathbb{R}^2

O algoritmo BBAC busca o resultado ótimo de *co-clustering*, com o objetivo de minimizar a distância entre a matriz aproximação, equação (2), e a matriz original, a partir da divergência de Bregman (BANERJEE *et al.*, 2007).

De acordo com o algoritmo *Bregman block-average co-clustering* (BBAC) (BANERJEE *et al.*, 2007), a análise com base em *co-clustering* se dá pela busca de blocos, de forma que se inicia com valores arbitrários para os agrupamentos em linhas e colunas da matriz de entrada, sendo que a cada iteração o agrupamento de linhas ou de colunas é atualizado, de forma se alcance a convergência desejada. Tal convergência é calculada a partir da divergência de Bregman, equação (1), tendo por entrada a matriz aproximação, equação (2), atualizada a cada iteração, conforme equações (3), (4) e (5).

Conforme notação apresentada por (IZQUIERDO-VERDIGUIER; ZURITA-MILLA, 2015), a fim de se minimizar a referida distância entre a matriz aproximação ($\hat{X} \in \mathbb{R}^{m \times n}$) e a matriz original ($X \in \mathbb{R}^{m \times n}$), obtém-se a matriz aproximação a seguir:

$$\hat{X} = \frac{E[X|\hat{U}, \hat{V}] \times E[X|U] \times E[X|V]}{E[X|\hat{U}] \times E[X|\hat{V}]} \quad (2)$$

em que $E[]$ é o valor esperado, U e V são os grupos de iniciação aleatórios e \hat{U} e \hat{V} são os grupos após as iterações do método *co-clustering*.

Para se construir a matriz de aproximação \hat{X} , deve-se atribuir os grupos de linhas e os grupos de colunas a partir do produto de matrizes a seguir:

$$\hat{X} = \hat{X}_{constante} \times \hat{X}_{variável} \quad (3)$$

em que $\hat{X}_{constante}$ é um fator constante de atualização seja de linhas ou colunas, e $\hat{X}_{variável}$ depende se está relacionado a linhas ou colunas, de forma que

- no caso de agrupamento de linhas, tem-se:

$$\begin{aligned} \hat{X}_{constante} &= \frac{E[X|U] \times E[X|V]}{E[X|\hat{U}]} \\ \hat{X}_{variável} &= \frac{E[X|\hat{U}V]}{E[X|\hat{V}]} \end{aligned} \quad (4)$$

- no caso de agrupamento de colunas, tem-se:

$$\begin{aligned} \hat{X}_{constante} &= \frac{E[X|U] \times E[X|V]}{E[X|\hat{U}]} \\ \hat{X}_{variável} &= \frac{E[X|U\hat{V}]}{E[X|\hat{U}]} \end{aligned} \quad (5)$$

Para aplicação do método no campo do sensoriamento remoto, deve-se considerar a matriz em questão como uma imagem, de forma que as linhas representem pixels e as colunas representem as bandas (IZQUIERDO-VERDIGUIER; ZURITA-MILLA, 2015). No escopo deste trabalho, a matriz será dada por linhas representando pixel e colunas representando camadas, uma vez que não se trata apenas de bandas espectrais, mas agregada de atributos calculados a partir de morfologia matemática e textura, conforme Figura 2.

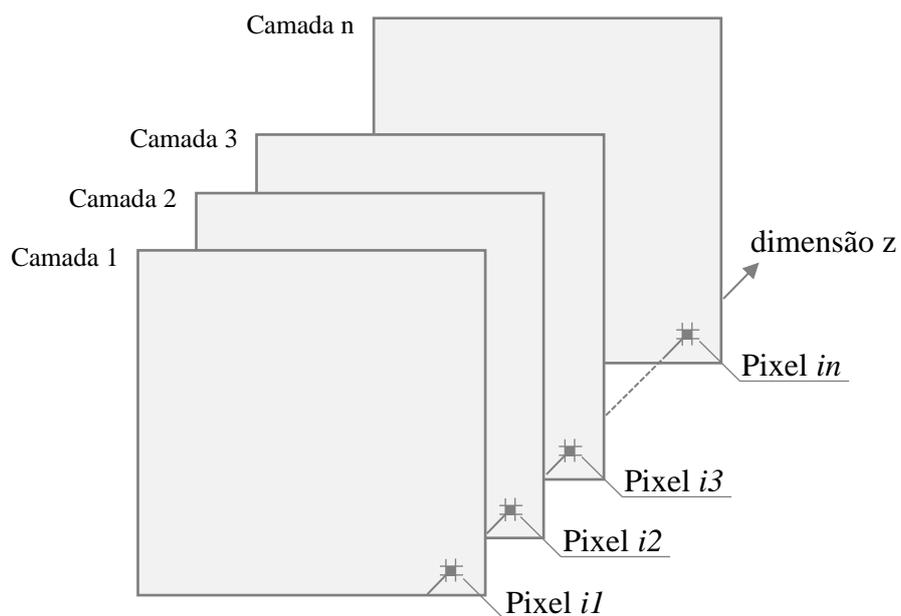


Figura 2 - Exemplo de representação de cubo de imagem

A partir do cubo de imagem representado pela Figura 2, constrói-se a matriz remodelada, em que cada linha representa um pixel da imagem original, e cada coluna representa as camadas do cubo em questão, conforme Figura 3.

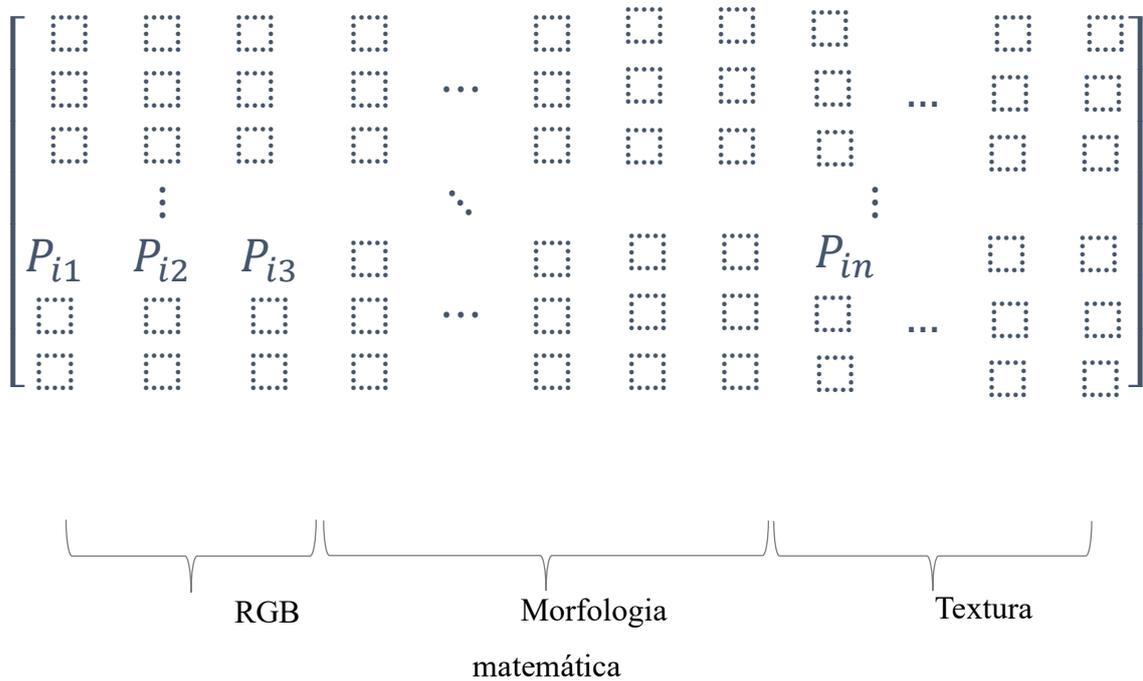


Figura 3 - matriz remodelada a partir do cubo de imagem

A técnica de análise de cubos de imagens mostra-se útil quando aplicada em análise de diferentes resoluções, sejam elas espacial, temporal ou espectral. Tal técnica é tipicamente apresentada como útil para casos de variação das dimensões espaciais e temporais (ZURITA-MILLA *et al.*, 2020). Dessa forma os agrupamentos resultantes representariam padrões que caracterizam espaço e tempo, simultaneamente.

Para se seguir tal proposta de estudo de imagens RGB de altíssima resolução espacial, a fim de se explorar o potencial da técnica *co-clustering*, mostra-se necessário o incremento de informação na dimensão z do cubo de imagem original, no caso, composto originalmente por 3 bandas RGB.

A partir do cubo de imagem original, a fim de se incrementar a gama de informações que se tem sobre a área a ser classificada na imagem, a proposta é produzir informação nova, de forma que as novas camadas geradas não sejam linearmente correlacionadas às demais já existentes, no caso as bandas RGB originais. Tal informação

deve ser concatenada como nova camada na dimensão z do cubo de imagem original, que passa a conter maior volume de dados. A partir desse novo cubo de imagem, com diversas camadas, simulando novas bandas, busca se trabalhar no contexto da análise de dados multivariados, conforme explorados na literatura, como apresentado, por exemplo em (WU; ZURITA-MILLA; KRAAK, 2015).

A proposta geral da técnica de *co-clustering* para classificação de imagens multiespectrais é a busca por padrões considerando, simultaneamente, características espaciais e espectrais, de forma que ao final do processo se tenha agrupados padrões no espaço e relativos à informação espectral, ou seja, de bandas. Dessa forma, geram-se grupos de bandas espectrais além de grupos de *pixels* similares. Um exemplo de sequência de trabalho do método *co-clustering* é apresentado na Seção 3.4.5, com entradas e saídas.

2.4 MORFOLOGIA MATEMÁTICA

Os estudos em morfologia matemática surgiram em 1965, com base na teoria dos conjuntos, apresentando desde então importantes contribuições, sendo sua principal aplicação a análise e processamento de imagens, em que mostrou-se como ferramenta eficaz para solução de problemas não lineares conforme apresentado por (BANON; BARRERA; BRAGA-NETO, 2007).

Mais especificamente, no contexto do Sensoriamento Remoto, morfologia matemática foi apresentado como eficaz para a extração automática de estradas, usando *Path Openings* e *Path Closing* para executar Perfis Morfológicos (VALERO *et al.*, 2009). Morfologia matemática pode ser encontrado na literatura a partir dos termos, em inglês, ‘*Mathematical Morphology*’, como, por exemplo, em (ANDALÓ *et al.*, 2007; BANON; BARRERA; BRAGA-NETO, 2007; KUPIDURA, 2019; RETORNAZ; MARCOTEGUI, 2007; VALERO *et al.*, 2009) e ‘*morphological profile*’ como, por exemplo, em (FAUVEL *et al.*, 2008; IZQUIERDO-VERDIGUIER; ZURITA-MILLA, 2015; VALERO *et al.*, 2009).

Estudos recentes apresentam desde introdução de novos formatos de elementos estruturantes que promovem ganhos em custo computacional (ANDALÓ *et al.*, 2007), passando por estudos que combinam morfologia matemática ao estudo de análise de textura para melhor resultado da classificação de imagens (APTOULA; LEFÈVRE,

2007), até estudos que objetivam utilizar morfologia matemática na busca automática de elementos textuais em uma imagem (RETORNAZ; MARCOTEGUI, 2007).

Resultados consistentes foram encontrados em alguns experimentos adotando metodologia que inclui simultaneamente informações espaciais e espectrais no processo de classificação por um esquema de fusão de dados das informações morfológicas e dos dados originais (FAUVEL *et al.*, 2008).

Para uso de técnicas relativas à filtragem morfológica deve-se aplicar à imagem o chamado elemento estrutural, que se trata de uma pequena imagem binária, ou seja, uma pequena matriz com pixels de valor 0 (zero) ou 1 (um). O padrão de posição dos valores 0 (zero) ou 1 (um) do elemento estrutural determina sua forma, como por exemplo, um disco, que será posicionado em todas possíveis posições na imagem (AUCKLAND UNIVERSITY, [s. d.]).

Há uma coleção de operações não lineares relacionadas a morfologia de imagens, em especial duas operações básica, sendo elas a erosão e a dilatação (PEDREGOSA *et al.*, 2011). Diversas operações combinadas podem ser feitas a partir da combinação dessas duas operações principais.

Tem-se, na literatura, diferentes abordagens e notações para descrever matematicamente os operadores morfológicos. No contexto deste trabalho, para o estudo de filtragem morfológica, segue-se as seguintes definições matemáticas, conforme apresentado por (NIXON; AGUADO, 2013):

A aplicação do elemento estruturante é dada pela translação de um conjunto A por x , denotado por $A + x$ ou A_x .

Sendo dois conjuntos A e B , sendo B o elemento estruturante, a Erosão de A por B é definida como

$$A \ominus B = \{x: B + x \subseteq A\} = \{x: B_x \subseteq A\} \quad (6)$$

Sendo dois conjuntos A e B , a Dilatação de A por B , sendo B o elemento estruturante, é definida como

$$A \oplus B = [(A)' \ominus (\check{B})]'$$

(7)

Em que $(A)'$ é o complementar de (A) e $\check{B} = \{-b: b \in B\}$, ou seja \check{B} é a reflexão de B .

Como resultados dos operadores apresentados, tem-se que erosão, conforme Figura 4, reduz os objetos removendo extrusões e dilatação, conforme Figura 5, aumenta os objetos e pode reparar quebras.

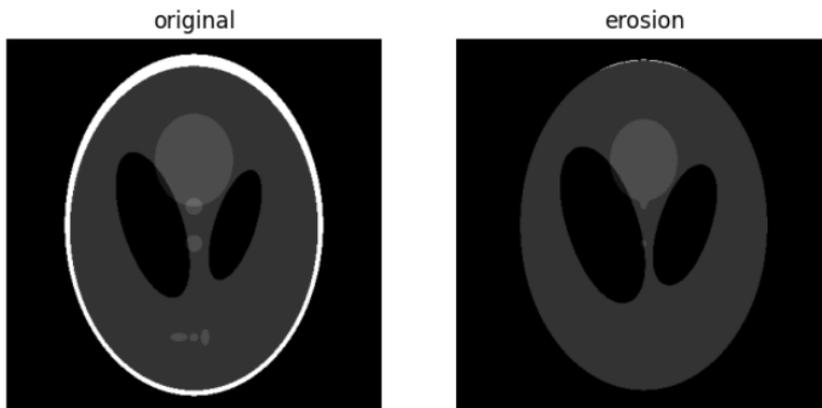


Figura 4 - exemplo de erosão de uma imagem

Fonte: (PEDREGOSA *et al.*, 2011)

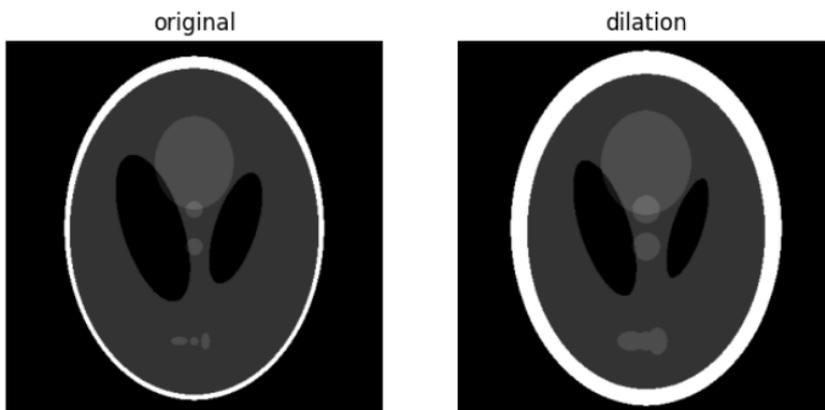


Figura 5 - exemplo de dilatação de uma imagem

Fonte: (PEDREGOSA *et al.*, 2011)

A operação *Opening* é dada por uma erosão seguida por uma dilatação:

$$A \circ B = (A \ominus B) \oplus B$$

(8)

A operação *Closing* é dada por uma dilatação seguida por uma erosão:

$$A \bullet B = (A \oplus B) \ominus B$$

(9)

Como resultados dos operadores apresentados, tem-se que o operador *Opening* remove pequenos pontos brilhantes e liga pequenos pontos escuros, e, por outro lado, o operador *Closing* pode remover pequenos pontos escuros e ligar pequenos pontos brilhantes.

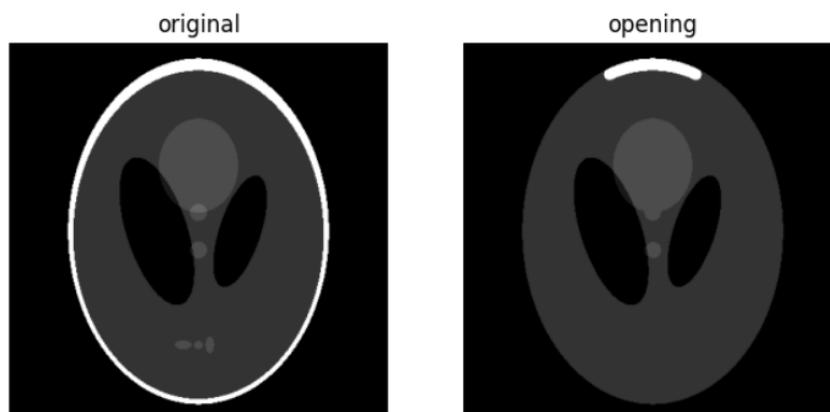


Figura 6 - exemplo de operador opening

Fonte: (PEDREGOSA et al., 2011)

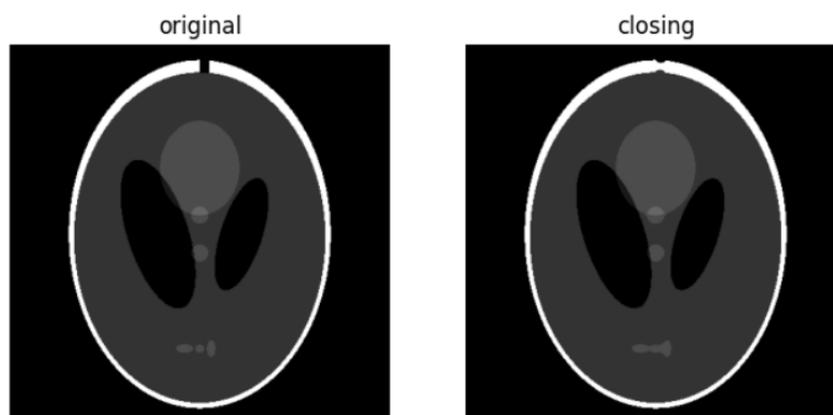


Figura 7 - exemplo de operador closing

Fonte: (PEDREGOSA et al., 2011)

Até então, os usuários devem executar filtragem morfológica e *co-clustering* separadamente, pois há códigos separados para as duas abordagens. Este trabalho explora a elaboração de um código integrado que cubra as duas abordagens, simultaneamente. Além disso, a análise de textura foi incluída como um terceiro elemento a ser investigado.

Há código desenvolvido em Python para filtragem morfológica, ou morfologia matemática (PEDREGOSA et al., 2011). Há, separadamente, um código desenvolvido considerando a abordagem de *co-clustering* para o agrupamento e análise de imagens

hiperespectrais (IZQUIERDO-VERDIGUIER; ZURITA-MILLA, 2015). Há, ainda, trabalho apresentando ferramentas de *co-clustering* em Python que realizam agrupamento explicitamente considerando a dimensionalidade do cubo de imagem, em especial dados com múltipla resolução temporal (ZURITA-MILLA *et al.*, 2020).

Este trabalho se concentra na integração de ambos os códigos, *co-clustering* e morfologia matemática, e na adição de análise de textura em um código único, a fim de permitir a análise de imagens multiespectrais e reconhecimento de padrões.

2.5 ANÁLISE DE TEXTURA

Textura é uma das características mais comuns na análise de imagens de Sensoriamento Remoto, especialmente simples por não exigir segmentação prévia (KUPIDURA, 2019). A análise de textura é uma técnica amplamente difundida e utilizada em Sensoriamento Remoto especialmente devido à facilidade computacional e ampla aplicabilidade no campo do sensoriamento remoto (HARALICK; DINSTEIN; SHANMUGAM, 1973).

A abordagem de textura apresenta, de forma geral, uma medida da variação de informação espacial local de uma imagem, a partir da definição dos parâmetros a serem analisados, tais como ordem de aplicação, tamanho do filtro e seleção prévia de bandas espectrais (WARNER, 2011). Filtros de textura são desenvolvidos a partir do estudo de entropia local de uma imagem, considerando a vizinhança de uma determinada região, avaliada a partir de um elemento estruturante definido. Ao final do processo, tem-se detectado variações sutis em valores digitais de cada pixel (PEDREGOSA *et al.*, 2011).

Análise de textura em imagens se mostrou eficaz em processos de classificação em variados métodos, modelos, filtros, resoluções e objetos observados (ZHU; YANG, 2010). Mais especificamente, em estudos de imagens de alta resolução espacial a análise de textura se mostra como fonte complementar de informação para processos de classificação de imagens de sensoriamento remoto, sendo que abordagens combinadas de análise de textura e informação espectral melhoram o resultado da classificação (RUIZ; FDEZ-SARRÍA; RECIO, 2004).

Para o escopo deste trabalho, utilizou-se a abordagem computacional, escrita em Python, para análise de entropia local e posterior detecção de textura desenvolvida por

(PEDREGOSA *et al.*, 2011), da qual tem-se como resultado imagem formada a partir de cálculo de entropia local, conforme Figura 6.

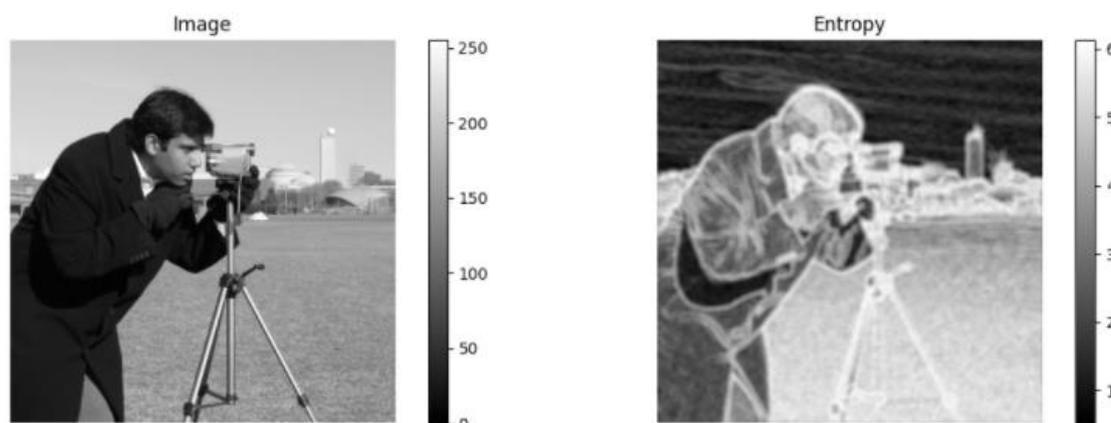


Figura 8 - exemplo imagem formada a partir de cálculo de entropia local

Fonte: (PEDREGOSA *et al.*, 2011)

2.6 K-MÉDIAS

No contexto geral de Sensoriamento Remoto, o índice K-médias tem sido amplamente utilizado (MENESES *et al.*, 2012), especialmente na redução da dimensionalidade espectral para fins de classificação de imagens hiperespectrais (R. KIRAN; SAICHANDANA; SRINIVAS, 2016) e (SAICHANDANA *et al.*, 2016). O referido índice também é explorado em processos de classificação não-supervisionada, servindo como objeto de comparação para diferentes métodos com base em algoritmos genéticos (SAICHANDANA; SRINIVAS, 2015).

No contexto deste trabalho, o referido método é utilizado tanto de forma independente como método de classificação não supervisionada clássico, quanto como uma fase de pós processamento, realizando um refinamento no agrupamento posterior às classes encontradas pelo método proposto de *co-clustering*. Desta forma, encontra-se similaridades já estabelecidas pelo código de *co-clustering*.

O método K-médias considerado como método de classificação não supervisionada tradicional utilizado no âmbito deste trabalho é realizado a partir da ferramenta “*scikit*”, conforme apresentado em (PEDREGOSA *et al.*, 2011). Destaca-se que o método k-médias foi aplicado ao cubo de imagem original, para fins de comparação com os demais métodos propostos.

O refinamento K-médias utilizado no âmbito deste trabalho segue o apresentado e utilizado em (NATTINO *et al.*, 2021). Sendo “k” o número de *clusters* de entrada do método de *co-clustering*, testa-se diversos valores “k”, a fim de se definir o valor ótimo, conforme coeficiente de Silhouette, que mede matematicamente a validade de um *cluster*, a partir da similaridade de um objeto ao seu próprio *cluster*, em comparação aos demais *clusters* (ROUSSEEUW, 1987).

Para o cálculo do coeficiente de Silhouette, deve-se definir a distância média de um ponto $a(i)$ para todos os demais pontos dentro de seu agrupamento, conforme notação de (MAROTTA, 2019):

$$s(i) = \frac{(a(i) - b(i))}{\max(b(i), a(i))} \quad (10)$$

2.7 AVALIAÇÃO DE QUALIDADE

O sucesso nos processos de classificação de imagens serve de insumo para tomada de decisões nos mais diversos processos da atividade humana, seja nos âmbitos público ou privado.

São diversas as normas nacionais e internacionais que abordam o conceito de avaliação da qualidade. No Brasil, a Infraestrutura Nacional de Dados Espaciais do Brasil (INDE) foi instituída pelo Decreto Presidencial nº 6.666, de 27 de novembro de 2008 (BRASIL, 2008). Coube, por sua vez, à Comissão Nacional de Cartografia (CONCAR) a responsabilidade de criar o Plano de Ação para Implantação da INDE (CINDE, 2010). O referido plano “indica as normas e padrões de dados geoespaciais de referência” (DSG, 2016b). A CONCAR encontra-se atualmente desativada, com a extinção do Ministério do Planejamento, Orçamento e Gestão, MPOG.

Conforme previsto no art. 15, do Cap. VIII, do Decreto-Lei nº 243, de 28 de fevereiro de 1967 (BRASIL, 1967), cabe à Diretoria de Serviço Geográfico (DSG) “elaborar a norma cartográfica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG) utilizados como referência para o Espaço Geográfico Brasileiro (EGB)” (DSG, 2016b).

A ET-CQDG (DSG, 2016b) adota o previsto na ISO 19157:2013 (ISO, 2013), que define parâmetros de descrição, registro e avaliação da qualidade de dados geoespaciais e acrescenta conhecimentos pertinentes ao contexto nacional.

A avaliação da qualidade, entendida como correspondência a um determinado padrão ou norma, é um importante passo no processo de produção da geoinformação. A ISO 19157 (ISO, 2013) e, por sua vez, a ET-CQDG (DSG, 2016b), define os passos a serem seguidos para o processo de avaliação da qualidade de dados espaciais.

2.7.1 Norma da Especificação Técnica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG)

A ET-CQDG apresenta os princípios da qualidade conforme definições da ISO 19157 (ISO, 2013). São eles Completude, Consistência lógica, Acurácia posicional, Acurácia temporal e Acurácia temática (DSG, 2016b). Para o escopo desse trabalho, o estudo foi feito na dimensão da Acurácia temática, que “diz respeito à correta interpretação das feições e atributos, bem como ao seu enquadramento nas classes previstas no modelo conceitual” (DSG, 2016b). Mais especificamente, este trabalho visa explorar a Acurácia da Classificação que, por sua vez, “obtem-se pela comparação das classes ou atributos encontrados no conjunto de dados geoespaciais com o modelo de dados adotado” (DSG, 2016b).

2.7.2 Amostragem

A ET-CQDG estabelece conceitos de amostragem, conforme os previstos nas normas ISO 2859-1 (ISO, 1999) e ISO 2859-2 (ISO, 1985). A ET-CQDG (DSG, 2016b) define, ainda, os procedimentos e métricas para amostragem dos tipos de produtos definidos na ET-PCDG (DSG, 2016b).

Em conformidade com a ISO 19157 (ISO, 2013), a ET-CQDG apresenta diferentes métodos de avaliação, sendo eles: inspeção completa, inspeção por amostragem, agregação/derivação, ou avaliação indireta (DSG, 2016b). Devido à natureza dos dados geoespaciais, para o escopo deste trabalho em conformidade com norma em questão, optou-se pelo método inspeção por amostragem, detalhado a seguir.

Em conformidade com a ISO 19157 (ISO, 2013), a ET-CQDG apresenta as seguintes orientações para realizar inspeção por amostragem:

“No procedimento de amostragem espacial, o produto a ser avaliado é particionado em células de 4 x 4 cm na escala do produto usando valores inteiros, ou seja, algo como o grid UTM. Por exemplo, em um produto 1:10.000 a célula teria 400 x 400 m, enquanto que num produto 1:25.000 a célula possui 1000 x 1000 m. Considerando que os produtos da cartografia sistemática no Brasil seguem a articulação UTM (DCT/DSG, 2016), cada produto terá um valor distinto de células. Algumas células não terão a área máxima, sobretudo nos limites do produto, enquanto que outras células ocuparão toda a área”. (DSG, 2016b)

Aplicando-se os conceitos apresentados, este trabalho estuda a escala cadastral 1:1000, a partir de imagens de resolução espacial de 5cm, de forma que cada Ortofoto objeto de estudo deste trabalho, após particionamento em células de 4 x 4 cm na escala do produto, resulta em células objeto de avaliação com 40m x 40m de extensão, para fins de avaliação.

2.7.3 Índice Kappa

O Índice Kappa surge como uma medida de confiabilidade proposta por (COHEN, 1960), inicialmente aplicada a casos de análise estatística que dava a proporção de concordância de decisões de dois diferentes juízes sobre classificar diferentes casos em diferentes categorias, de forma a demonstrar a correlação do resultado com as diferentes variáveis envolvidas. A estimativa do valor de Kappa é apresentado pela estatística Khat, conforme (CONGALTON, 2001) podendo variar entre +1 a -1, em que valores positivos são esperados em análises de sensoriamento remoto, de forma que o valor +1 indica concordância total, e que valor negativos indicam que não há concordância.

Dada a matriz de confusão, conforme apresentada por (NAKAO, 2020)

$$\begin{bmatrix} m_{11} & \cdots & m_{1L} \\ \vdots & \ddots & \vdots \\ m_{L1} & \cdots & m_{LL} \end{bmatrix}$$

tem-se que o cálculo do coeficiente Kappa é expressa, conforme (CONGALTON, 1991), por

$$Kappa = \frac{n \sum_{i=1}^L m_{ii} - \sum_{i=1}^L m_{i+} m_{+j}}{n^2 - \sum_{i=1}^L m_{i+} m_{+j}}$$

em que:

L: quantidade de classes

m_{ij} : quantidade de elementos da classe i pertencente ao grupo j

m_{i+} : soma dos elementos da linha i

m_{j+} : soma dos elementos da coluna j

n: número total de observações

Uma metodologia estatística de análise de dados multivariados relacionada à concordância de diferentes resultados obtidos por diferentes observadores é apresentada por (LANDIS; KOCH, 1977), de forma que os possíveis valores de Kappa são divididos em três grupos, sendo que valores superiores a 0,8 representam forte concordância, valores entre 0,40 e 0,80 representam concordância moderada e valores abaixo de 0,4 representam fraca concordância.

O referido índice é amplamente utilizado em estudos de imagens de alta resolução espacial, especificamente para avaliação da acurácia da classificação supervisionada, como por exemplo em estudo referente a classificação orientada a objeto, OBIA, com utilização do software livre InterIMAGE (INTERIMAGE, [s. d.]), proposto por (PAHL *et al.*, 2003), em que se verificaram resultados satisfatórios de classificação (ABOUD NETA; BIAS; DOS SANTOS, 2018). O referido software livre foi utilizado como ferramenta de avaliação de classificação OBIA, em áreas urbanas (PASSO *et al.*, 2013). Em complemento, a partir do referido índice, foi avaliado como satisfatório o resultado da classificação realizada por intermédio da integração de diferentes técnicas de mineração de dados utilizando software livre, aplicadas a imagens orbitais de alta resolução espacial (ANTUNES *et al.*, 2018a) e (ANTUNES *et al.*, 2018b).

Apesar da sua importância destacada, e ampla utilização em aplicações de Sensoriamento Remoto, o Índice Kappa não se aplica ao escopo deste trabalho por este estudo se tratar de classificação não-supervisionada, sem atuação do operador em qualquer das etapas, inclusive sem uma posterior tematização das feições identificadas. Logo, ao final da classificação, não se tem, de forma óbvia, uma classe da verdade de campo, no caso, uma classe da classificação prévia, representando uma única classe resultante do classificador *co-clustering*, objeto de estudo deste trabalho. Disso, tem-se como alternativa o estudo de medidas de similaridade, que consideram o agrupamento resultante e não associam o mesmo a uma única classe da classificação prévia.

2.7.4 Medidas de similaridade

Dado o resultado do agrupamento de pixels de uma imagem de sensoriamento remoto, em comparação à verdade de campo, ou classificação previamente realizada, é possível estabelecer medidas de similaridade entre os mesmos. São apresentados a seguir diferentes medidas de similaridade comumente utilizadas em Sensoriamento Remoto e exemplos de aplicação.

Conforme apresentado por (WAGNER; WAGNER, 2007), adaptado para o cálculo dos índices de similaridade comumente utilizados e apresentados na literatura especializada, seguem-se as seguintes definições e notações matemáticas:

Seja X um conjunto finito com cardinalidade $|X| = n$. Um agrupamento C é um conjunto $\{C_1, \dots, C_k\}$ de subconjuntos disjuntos não vazios de X de modo que sua união seja igual X . O conjunto de todos os agrupamentos de X é denotado por $P(X)$. Para um agrupamento $C = \{C_1, \dots, C_k\}$ assumimos $|C_i| > 0$ para todo $i = 1, \dots, k$.

Seja $C' = \{C'_1, \dots, C'_l\} \in P(X)$ um segundo agrupamento de X . A matriz de confusão, ou tabela de contingência, $M = (m_{ij})$ do par (C, C') é uma matriz $k \times l$ cujas entradas ij – ésima é igual ao número de elementos na interseção dos agrupamentos C_i e C'_j :

$$m_{ij} = |C_i \cap C'_j|, 1 \leq i \leq k, 1 \leq j \leq l \quad (11)$$

Para fins de se desenvolver as medidas de similaridade, comumente se definem pares de objetos, no caso *pixels*, relacionando a classificação obtida e a classificação prévia, considerada verdade para fins de comparação. Sejam as possíveis interpretações da relação entre tais pares de objetos, descritos a seguir, adaptado de (WAGNER; WAGNER, 2007):

$S_{11} = \{\text{pares que estão no mesmo } cluster \text{ e mesma classe em } C \text{ e } C'\}$

$S_{00} = \{\text{pares que estão em diferentes } cluster \text{ e mesma classe em } C \text{ e } C'\}$

$S_{10} = \{\text{pares que estão no mesmo } cluster \text{ e diferentes classes em } C \text{ e } C'\}$

$S_{01} = \{\text{pares que estão em diferentes } cluster \text{ e diferentes classes em } C \text{ e } C'\}$

Tem-se que a representação da quantidade de pares relacionados, tal que:

$$n_{ab} := |S_{ab}|, a, b \in \{0,1\} \quad (12)$$

$$n_{11} + n_{00} + n_{10} + n_{01} = \binom{n}{2} \quad (13)$$

2.7.4.1 Índice de Rand

O Índice de Rand, proposto por (RAND, 1971), apresenta uma medida de similaridade entre dois conjuntos, considerando também os chamados valores verdadeiros negativos, ou seja, os casos em que um par de pontos resulta como pertencente a grupos distintos e classes distintas.

Os valores do Índice Rand variam entre 0 e 1, de forma que o valor 0 representa os casos em que o agrupamento gerado e a classe com que se compara não concordam em nenhum par de pontos e, por outro lado, o valor 1 representa os casos em que o agrupamento gerado e a classe com que se compara são idênticos.

O Índice de Rand, conforme apresentado por (WAGNER; WAGNER, 2007), é dado por:

$$R(C, C') = \frac{2(n_{11} + n_{00})}{n(n-1)} \quad (14)$$

Sendo n dado pela equação (11).

O método proposto pelo índice de Rand para avaliação da similaridade entre dois conjuntos em complementação à análise visual foi utilizada como objeto de comparação para método de avaliação de resultados de segmentação de imagens (ZHANG; XIAO; FENG, 2012). O referido índice foi utilizado para avaliar e concluir quanto a uniformidade dos resultados de *clustering* em segmentação de imagens de alta resolução espacial (DI *et al.*, 2017). De forma mais específica, o Índice de Rand foi também utilizado para avaliação do resultado da segmentação de imagens urbanas (DENG; ZHANG; TIAN, 2015).

2.7.4.2 Índice de Rand Ajustado

A partir dos resultados de estudos utilizando-se o Índice de Rand, foi proposto o Índice Rand Ajustado, por (HUBERT; ARABIE, 1985), de forma que os limites dos valores fiquem entre -1 e 1, sendo que o valor zero representa total independência entre os métodos de agrupamento e classificação, o valor máximo de 1 representa classificações idênticas e, por fim, valores negativos representam comportamento aleatório (WAGNER;

WAGNER, 2007). Com isso, o índice de Rand Ajustado avalia a similaridade entre os conjuntos de forma que o intervalo de possíveis valores resultantes é mais amplo, conferindo mais sensibilidade do método quanto ao resultado obtido.

O Índice Rand Ajustado, conforme apresentado por (WAGNER; WAGNER, 2007), é dado por:

$$R_{adj}(C, C') = \frac{\sum_{i=1}^k \sum_{j=1}^l \binom{m_{ij}}{2} - t_3}{\frac{1}{2}(t_1 + t_2) - t_3} \quad (15)$$

$$\text{Em que } t_1 = \sum_{i=1}^k \binom{|c_i|}{2}, t_2 = \sum_{j=1}^l \binom{|c'_j|}{2}, t_3 = \frac{2t_1 t_2}{n(n-1)} \quad (16)$$

Sendo m dado pela equação (10) e n dado pela equação (11).

O referido índice foi considerado em estudo de busca de similaridade espectral em imagens multiespectrais, para fins de monitoramento ambiental em agricultura (TASDEMIR; MOAZZEN; YILDIRIM, 2015). O índice de Rand Ajustado foi explorado na fase de comparação à verdade de campo considerada, em processos de extração de regiões de interesse em imagens, ROI, *region of interest*, a fim de se ter redução de complexidade computacional para aplicação de métodos automáticos de segmentação de imagens (LI; WEI; ZHAO, 2017). O referido índice foi, ainda, utilizado na avaliação de resultados de segmentação para imagens hiperespectrais (GHOLIZADEH; ZOEJ; MOJARADI, 2012).

A partir do exposto, tem-se que o Índice de Rand Ajustado mostra-se como o índice mais adequado para avaliação de classificação não supervisionada, dentre os considerados no contexto deste trabalho, uma vez que o mesmo considera em seu cálculo os valores verdadeiros negativos, ou seja, os casos em que um par de pontos resulta como pertencente a grupos distintos e classes distintas, e em complemento expande os limites para valores entre -1 e 1, de forma que explicita possível comportamento aleatório dos resultados.

2.7.4.3 Índice de Jaccard

O Índice de Jaccard, também conhecido por representar “intersecção sobre união” entre dois conjuntos, se coloca no ramo da estatística como mais uma medida de similaridade entre diferentes amostras. Tal índice foi proposto por (JACCARD, 1912),

tendo sido amplamente utilizado na análise do resultado de classificação de imagens de sensoriamento remoto, como visto em (NAKAO, 2020), em especial para avaliação de processos de classificação e segmentação de imagens (SILLERO *et al.*, 2009; SOMODI *et al.*, 2012; WU *et al.*, 2018).

O Índice de Jaccard, conforme apresentado por (WAGNER; WAGNER, 2007), é dado por:

$$J(C, C') = \frac{n_{11}}{n_{11} + n_{10} + n_{01}} \quad (17)$$

Sendo n dado pela equação (11).

Dada sua importância, estudos na área de segmentação semântica buscam métodos de se melhorar o resultado do referido índice, tais como aplicação de modelo com base em redes neurais, para análise de imagem de alta resolução espacial (WU *et al.*, 2018) e (MOHAJERANI; KRAMMER; SAEEDI, 2018).

3 MATERIAL E MÉTODOS

3.1 SOFTWARES E APLICAÇÕES

- Python, versão 3.9, como linguagem de programação (PYTHON SOFTWARE FOUNDATION, 2020).
- Pacote de software Python “*Clustering geo-Data Cubes (CGC)*”, que realiza análise de agrupamento para dados geoespaciais multidimensionais (NATTINO *et al.*, 2021).
- Pacote de software Python “*scikit*”, que realiza diversas ferramentas de processamento de imagens, com destaque para classificação não supervisionada K-médias (PEDREGOSA *et al.*, 2011).
- Software livre QGIS (QGIS, 2018), como plataforma SIG para apresentação e manipulação das imagens.

Para a aplicação e verificação da metodologia, foram desenvolvidos códigos Python para análise das imagens a partir dos conceitos agregados de *co-clustering*, textura

e morfologia matemática, classificação K-médias e posterior avaliação, conforme Apêndice 1 e Apêndice 2.

3.2 ÁREA DE ESTUDO

Para fins de comparação de resultados de classificação, utilizaram-se imagens referentes à cidade Potsdam, na Alemanha, disponibilizados pela ISPRS – *The International Society for Photogrammetry and Remote Sensing* (ISPRS, 2013), obtidas por aerolevanteamento.

A proposta inicial deste trabalho envolvia uso de imagens adquiridas por Veículos Aéreos Não Tripulados da DSG, os quais demandariam extenso trabalho adicional para realizar a “verdade de campo”, ou seja, classificação manual que seria base de comparação ao método proposto. A partir deste contexto, decidiu-se pela escolha de imagens de altíssima resolução disponibilizadas pela ISPRS.

Justifica-se o uso das referidas imagens, fora dos limites territoriais do Brasil devido sua ampla utilização em estudos de sensoriamento remoto, os quais tem por insumos os dados da consagrada ISPRS, para fins de processamento e posterior comparação com a “verdade de campo” divulgada e exaustivamente testada (ISPRS, 2013), a fim de se avaliar o método proposto.

As imagens utilizadas, referentes à cidade Potsdam, na Alemanha, apresentam grandes quadras de prédios e construções, ruas estreitas e densa estrutura urbana. Os dados de classificação referentes à área considerada, utilizados como verdade de campo para o contexto deste trabalho, foram obtidos manualmente (ISPRS, 2022).

3.3 ORIGEM DOS DADOS E INFORMAÇÕES

As soluções tecnológicas propostas são baseadas em softwares livres, em especial o QGIS (QGIS, 2018) e a linguagem de programação Python, em sua versão 3.9 (PYTHON SOFTWARE FOUNDATION, 2020).

A fim de se realizar o estudo proposto, integraram-se os códigos relativos às análises de *co-clustering* apresentado por (NATTINO *et al.*, 2021), aos códigos para análise de morfologia matemática e de textura apresentado por (PEDREGOSA *et al.*, 2011) para análise e processamento de imagens de alta resolução espacial.

O estudo foi realizado a partir de imagem de altíssima resolução espacial disponibilizadas gratuitamente pela ISPRS (ISPRS, 2013). Tais imagens são disponibilizadas em formato “TIF”, abreviação de *Tagged Image File*. Além das imagens, são disponibilizadas ainda pela ISPRS imagem classificada, também em formato “TIF”, que compõe a verdade de campo para nossa análise, que possibilita a comparação com o método apresentado nesse trabalho.

Os dados são relativos a um banco de dados representado a partir de um mosaico, Figura 9, com 38 ortofotos. O GSD, *ground sampling distance*, da imagem é 5cm. A fim de evitar áreas sem dados, “buracos”, as imagens disponibilizadas na página da ISPRS, foram selecionadas na parte central do mosaico (ISPRS, 2013).



Figura 9 - mosaico de ortofotos da região de Postdam, Alemanha

Fonte: (ISPRS, 2013)

As imagens disponibilizadas pelo ISPRS são fornecidas em arquivos TIFF, em diferentes composições de canais, em que cada canal possui uma resolução radiométrica de 8 bits:

IRRG: 3 canais (IR, infra vermelho -R, vermelho -G, verde)

RGB: 3 canais (R, vermelho – G, verde – B, azul)

RGBIR: 4 canais (R, vermelho – G, verde – B, azul – IR, infra vermelho)

Para o escopo desse trabalho decidiu-se por utilizar a composição RGB, por ser a composição de imagens mais comum de se obter, inclusive para dados provenientes de sistemas imageadores de veículos aéreos não tripulados. Conforme acentuado na introdução deste trabalho, tem-se as imagens RGB como insumo para aplicação deste método. As referidas imagens são compreendidas como cubo de imagem. O referido cubo foi posteriormente modificado, concatenando-se novas camadas de textura e de morfologia matemática, a fim de explorar todo o potencial da técnica de *co-clustering* para análise de dados multivariados.

3.4 MÉTODO

Para aplicação da técnica de *co-clustering*, construiu-se um cubo de imagem, a partir da concatenação de camadas de morfologia matemática, camadas geradas de análise de textura e camadas dadas pelas bandas da imagem RGB original, conforme Figura 10.

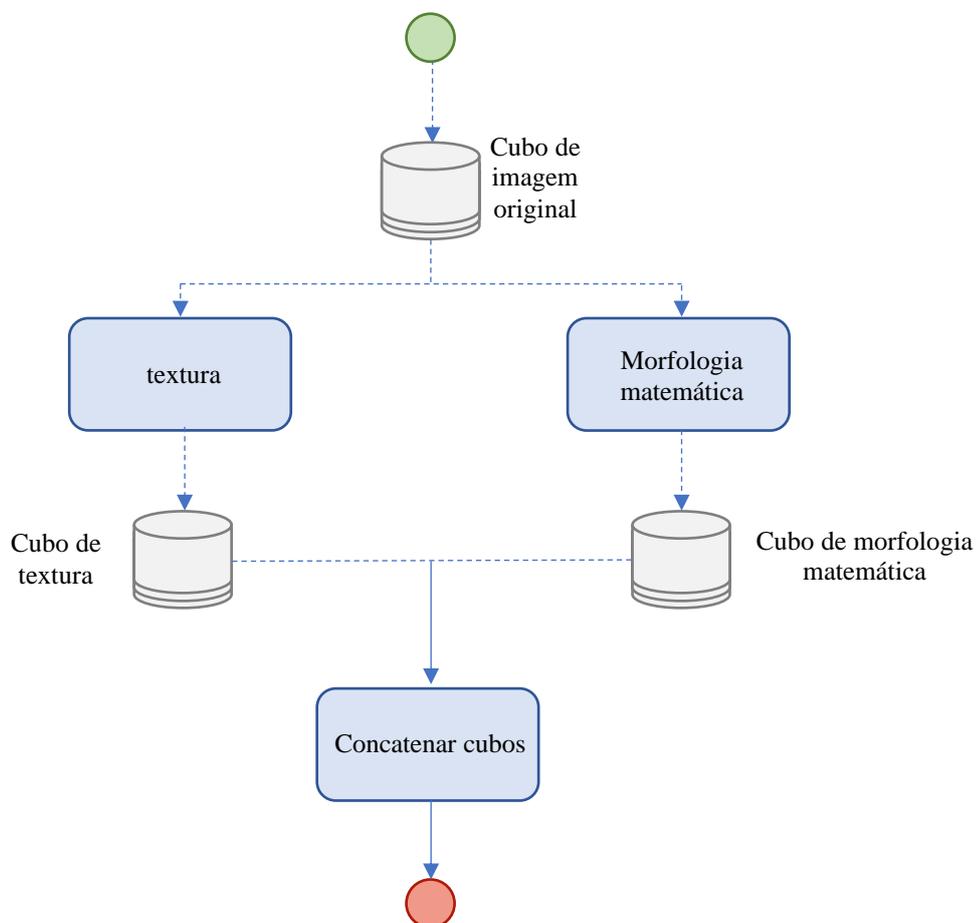


Figura 10 - Fluxograma representando o processo de construção do cubo

O círculo verde indica o ponto inicial com a imagem referida como entrada e o círculo vermelho indica o ponto final com os resultados da técnica de *co-clustering*, como saída. A imagem RGB original incrementada com camadas de morfologia matemática e camadas de textura é representada como um cubo de imagem. Nesse cubo de imagem, duas tarefas são executadas em paralelo: análise de morfologia matemática e análise de textura. De cada uma dessas duas tarefas, um novo cubo de imagens é produzido como uma saída local, conforme Figura 10. Os detalhes para cada uma dessas duas tarefas, entradas e saídas, são apresentados nas seções 3.4.1 e 3.4.2. A fim de agrupar os cubos resultantes, reapresentam-se os cubos de imagem como matrizes e, em seguida, concatenam-se ambas as matrizes, reorganizadas para se ter uma matriz única. Tal matriz concatenada agrupa informações de valor de cada pixel para todas as bandas produzidas, que são as bandas da imagem original, as bandas produzidas a partir do código da morfologia matemática e as bandas produzidas a partir de a análise de textura. Finalmente, tal matriz concatenada inclui informação da dimensão z, representada nas colunas, para cada pixel da imagem original, representados pelas linhas.

3.4.1 Código para estudo de morfologia matemática

Para este trabalho, a análise de morfologia matemática foi realizada usando a função Python ‘morphology’, do pacote ‘Scikit’ (PEDREGOSA *et al.*, 2011). Esse código fornece a morfologia matemática de uma imagem e foi desenvolvido em Python, sendo parte do código apresentado no Apêndice 1. Conforme apresentado na Seção 2.4, a entrada é o cubo de imagem original, a partir do qual, para cada banda da imagem, fez-se a análise de morfologia matemática, segundo conceitos de *opening* e *closing*.

Após processamento, banda a banda, concatena-se ao cubo da imagem original a imagem resultante da referida análise de morfologia matemática, a partir da dimensão banda. A partir da referida concatenação de camadas, o cubo da imagem original passa a ter mais camadas, na dimensão banda.

Para desenvolver-se a referida análise de morfologia matemática, definiu-se o elemento estruturante disco, de diferentes raios, sendo eles raios de 20 pixels, de 8 pixels e de 2 pixels, de forma a se ter diferentes tamanhos de elementos estruturantes aplicados a cada banda da imagem original, a fim de que sejam processadas e geradas informações

de diferente teor estatístico. Dessa forma produziram-se 3 perfis para cada banda do cubo original, no caso 3 bandas.

Por exemplo, tendo como entrada uma imagem RGB, com 3 bandas e número de perfis igual a 3, a função calcularia para cada banda da imagem original as imagens de fechamento e abertura, considerando o disco do elemento de estrutura dos raios 20, 8 e 2 pixels. Neste exemplo, a saída seria um cubo com 3 bandas da imagem original, concatenadas a mais 6 camadas para cada banda da imagem original; a saída seria, portanto, um cubo de imagem com 21 camadas, conforme Figura 11.

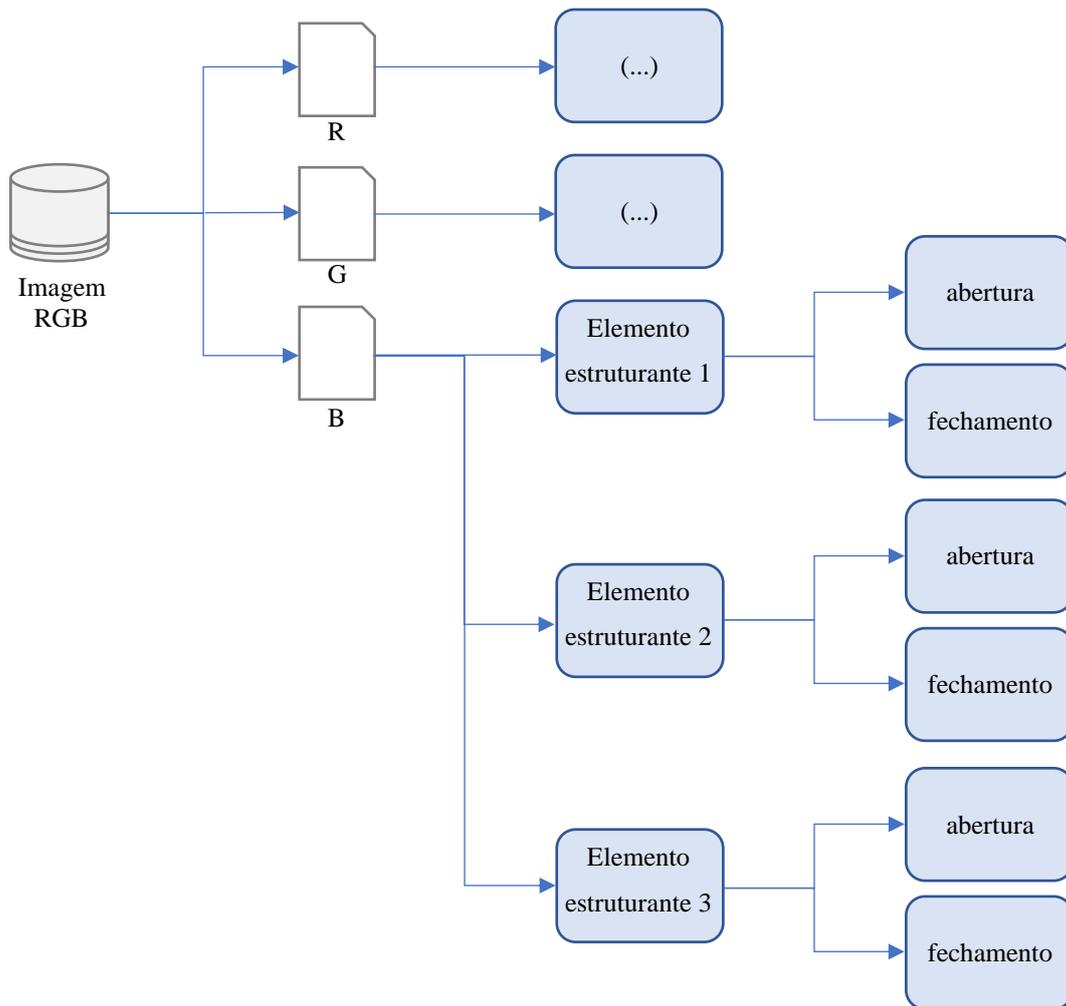


Figura 11 - camadas de morfologia matemática

3.4.2 Código para análise de textura

Conforme apresentado na Seção 3.4.1, tratando de morfologia matemática, a fim de se manter a coerência das decisões, decidiu-se, no processo de análise de textura, por

usar o elemento estruturante disco de raios de 20 pixels, de 8 pixels e de 2 pixels, de forma que sejam processadas informações na imagem de diferentes tamanhos.

A entropia caracteriza a textura de uma imagem como uma medida de aleatoriedade. Para este trabalho, a análise de textura foi realizada usando a função Python 'entropy', do pacote 'Scikit' (PEDREGOSA *et al.*, 2011). Esta função calcula a entropia local da imagem de intensidade. A entrada é o cubo da imagem original. A saída é um cubo do mesmo tamanho da imagem de entrada, em que cada camada corresponde a cada banda da imagem de entrada, de forma que é uma matriz em que cada pixel contém o valor de entropia da vizinhança 9 (*pixels*) por 9 (*pixels*) ao redor do pixel correspondente na banda de entrada da imagem.

Por exemplo, tendo-se como entrada um cubo de imagem RGB inicial, com 3 (três) bandas, a saída foi um cubo de tamanho com 9 (nove) camadas, conforme propriedades das Ortofotos utilizada no âmbito deste trabalho, representadas pela Figura 9.

3.4.3 Etapas e métodos de classificação

A metodologia foi dividida em etapas, com relação aos métodos de classificação, conforme apresentado no fluxograma da Figura 12.

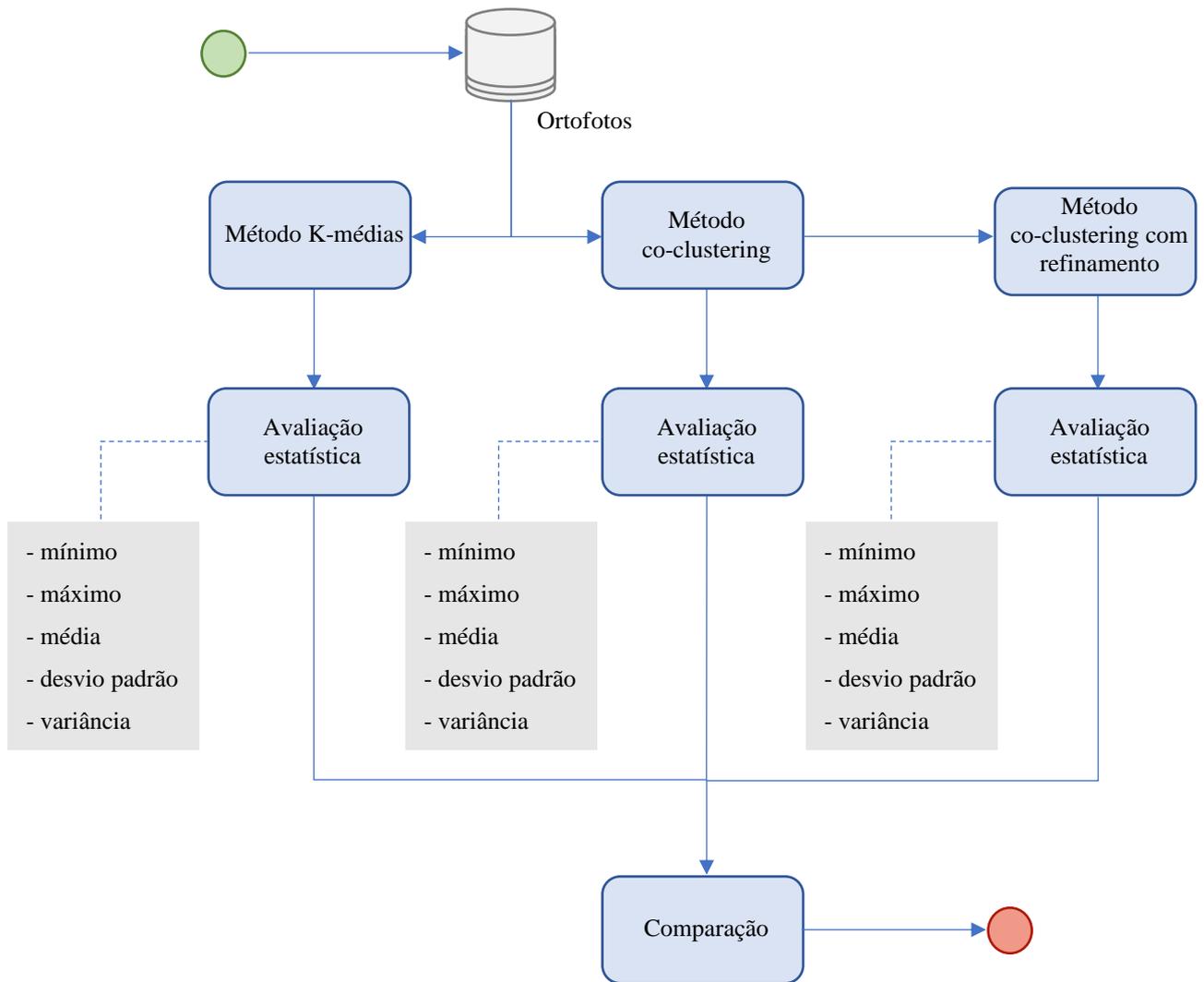


Figura 12 - fluxograma representando as etapas do trabalho

3.4.4 Etapa 1: classificação K-médias

A primeira etapa consiste em realizar a classificação não supervisionada pelo método tradicional K-médias, das imagens geradas a partir das ortofotos disponibilizadas pela ISPRS, conforme Figura 9. Para tal utilizou-se a ferramenta “*scikit*”, conforme apresentada por (PEDREGOSA *et al.*, 2011). Como parâmetros de entrada do método é necessário determinar o número máximo de classes, o número máximo de iterações e tolerância relativa a iterações, os quais foram mantidos conforme o padrão da ferramenta. Para o estudo em questão, definiu-se número máximo de classes igual a 10, uma vez que não há mais de 10 classes diferentes na verdade de campo considerada. Definiu-se, ainda, o número máximo de iterações igual a 100 e a tolerância relativa a iterações consecutivas igual a 0,2.

3.4.5 Etapa 2: classificação co-clustering

A segunda etapa apresenta a abordagem utilizada na aplicação da técnica de *co-clustering* ao cubo resultante da concatenação da imagem original aos estudos de perfis morfológicos e textura, apresentada como um fluxograma na Figura 10. Essa matriz remodelada foi usada como entrada para executar a técnica de *co-clustering*. Finalmente, como saída desta última tarefa, a matriz classificada é produzida como saída final. Após a referida classificação *co-clustering*, cada pixel correspondente da imagem resultante é associado a uma das classes resultantes.

Conforme apresentado na Seção 2.3, a referida função codificada em Python (NATTINO *et al.*, 2021) originalmente aplicava o método *co-clustering* considerando a dimensão “tempo”, sendo que a mesma foi adaptada para este projeto a fim de se considerar a dimensão *z*, das bandas originais da imagem em questão. A partir de tal adaptação, a referida função fornece o resultado do *co-clustering* do cubo de imagem RGB, concatenado a camadas de morfologia matemática e camadas de textura, em linhas, colunas e bandas espectrais. Os parâmetros de entrada são a matriz na qual a técnica foi processada, o número de grupos de linhas (*num_clusters_espaciais*), o número de grupos de colunas (*num_clusters_banda*), o tamanho da matriz de entrada em pixels (*sampleSize*), o número máximo de iterações (*max_iterations*), o limite do erro de convergência (*conv_threshold*) e o número máximo de inicializações randômicas (*num_runs*).

A matriz de entrada possui um número de linhas igual ao número de pixels na imagem original e um número de colunas igual ao número de bandas/camadas após concatenar os cubos de resultado da análise de textura e do código da morfologia matemática. O número de grupos de linhas significa o número de grupos final para se agrupar em linhas e o número de grupos de colunas significa o número de grupos final para se agrupar em colunas. O tamanho da matriz representa o valor em pixels da matriz quadrada que se deseja trabalhar. A partir desse número o código é capaz de reconhecer a imagem original e dividir a mesma em células de tamanhos iguais conforme o valor apresentado na entrada do código.

Virtualmente, o código de *co-clustering* gera duas matrizes, ‘R’ e ‘C’, que são respectivamente grupos em linha final e grupos em coluna final. A matriz ‘R’ possui um número de linhas igual ao número de pixels na imagem original e um número de colunas

igual ao número de grupos de linhas escolhidos como entrada. Na matriz ‘R’, cada linha representa cada pixel na matriz de entrada e cada coluna representa cada grupo de clusters. O valor dos pixels é 0 (Falso) ou 1 (Verdadeiro), indicando se o pixel na imagem original representada pela linha pertence ao grupo representado pela coluna. A matriz ‘C’ possui um número de linhas igual ao número de bandas da matriz de entrada e número de colunas igual ao número de clusters de colunas escolhidos como entrada. Na matriz ‘C’, cada linha representa cada camada gerada a partir da imagem original, ou seja, as bandas da imagem original, os perfis morfológicos e a análise de textura.

Por exemplo, tendo por base cada ortofoto, dividindo-a em 49 imagens, tendo cada uma delas 800x800 *pixels*, resulta-se total de pixels igual a 640000, total de bandas ou camadas do cubo de imagem igual a 30. Definindo-se o número de *clusters* de linhas igual a 10 e o número de *clusters* de “bandas” igual a 3, virtualmente resulta-se na matriz R, matriz de grupos de linha final, com 640000 linhas e 10 colunas e a matriz C, matriz de grupos de coluna final, com 30 linhas e 3 colunas.

Para o escopo deste trabalho, considerou-se a matriz ‘R’ para conclusão da classificação, ou seja, o agrupamento em linhas resultante do código *co-clustering*. Após processamento, resultou-se em imagem classificada, com dimensões iguais às dimensões da imagem original em questão.

3.4.6 Etapa 3: classificação co-clustering com refinamento por K-médias

A terceira etapa consiste em realizar a classificação não supervisionada pelo método *co-clustering* integrado com estudos de morfologia matemática e análise de textura, apresentado pelo item anterior, com posterior refinamento pelo método K-médias, conforme proposto por (NATTINO *et al.*, 2021). Para tal, aplica-se o método K-médias sobre a imagem previamente classificada pelo método *co-clustering*, a fim de se refinar o resultado do agrupamento resultante.

3.4.7 Etapa 4: avaliação

Após a conclusão das etapas apresentadas, referentes à classificação tradicional K-médias, ao método *co-clustering* sem refinamento e ao método *co-clustering* com refinamento, a etapa seguinte trata da avaliação da qualidade da classificação final das imagens, tendo por crivo medidas matemáticas de confiabilidade, dadas por medidas de

similaridade representadas pelos Índice de Rand, Índice de Rand Ajustado e Índice de Jaccard, conforme apresentadas respectivamente nas Seções 2.7.3 e 2.7.4.

Para avaliação do resultado da classificação das imagens disponibilizadas pela ISPRS, referentes à área de Potsdam, na Alemanha, após aplicação da técnica de *co-clustering*, seguiu-se os seguintes passos:

- divisão por amostragem das imagens;
- utilização dos seguintes métodos de qualidade/acurácia da classificação:
 - Índice de Jaccard (*intersection over union*);
 - Índice Rand;
 - Índice de Rand Ajustado.
- comparação dos resultados de classificação, a partir do cálculo de máximo, mínimo, média, desvio padrão e variância dos valores medidos para os índice de Rand, índice de Rand Ajustado e índice de Jaccard, para as diversas imagens processadas;
- por fim, é realizada uma avaliação estatística da hipótese nula de os resultados de classificação serem estatisticamente iguais, a partir dos dados resultantes do cálculo do Índice de Rand Ajustado, considerando o resultado do método proposto em comparação ao método K-médias.

4 DISCUSSÃO E APLICAÇÃO DO MÉTODO

A fim de se verificar a aplicação do método e posterior análise da qualidade, desenvolveu-se um código capaz de integrar abordagens de análise de imagens por meio de filtragem morfológica e *co-clustering*, adicionando-se ao estudo bandas de textura, conforme apresentado em detalhes no Apêndice 1.

Como complemento, após realizado o agrupamento de pixels pelo método *co-clustering*, é realizado o refinamento como método *K-médias*. As entradas são o cubo de imagem, os raios dos discos que serão considerados na análise de textura e na análise de perfis morfológicos, o número de clusters em bandas, o número de clusters de pixels, o número máximo de iterações, o limite de convergência, o número máximo de iterações. As saídas são matrizes referentes a grupos de linha final e grupos de coluna final.

Seguindo-se os objetivos específicos deste trabalho, o referido código apresentado no Apêndice 1 avalia as medidas de similaridade a partir do cálculo dos índices de Jaccard, de Rand e de Rand Ajustado.

4.1 EXEMPLO DE APLICAÇÃO DO MODELO PROPOSTO

Conforme apresentado em detalhes no Apêndice 1, o módulo *co-clustering* realiza a análise de *co-clustering* em uma matriz de entrada. Há trabalho realizando a análise variando-se a dimensão tempo, em que as linhas da matriz de entrada representam a informação espacial e as colunas da referida matriz representam a dimensão tempo (NATTINO *et al.*, 2021). Para o escopo deste trabalho, considerou-se igualmente as linhas representando a informação espacial da imagem, no entanto com as colunas representando a dimensão bandas, de forma que as colunas representam tanto a informação espectral original, quanto as camadas de morfologia matemática e de textura concatenadas ao cubo de imagem de entrada analisado.

O resultado do processamento da matriz de entrada representa a busca por padrões que consideram características espaciais, espectrais, de morfologia matemática e de textura, simultaneamente, gerando-se *co-clusters*.

A fim de se ilustrar a sequência do procedimento em questão, desenvolveu-se um código Python, adaptado do manual de usuário do código de referência de *co-clustering* (NATTINO *et al.*, 2022):

- Inicialmente, tem-se a importação de bibliotecas e configuração geral;
- Em seguida, tem-se a leitura e preparação da Ortofoto;
- Após a entrada da imagem, divide-se a mesma conforme regra de amostragem apresentada na Seção 2.7.2;
- Em continuidade, realiza-se a análise de morfologia matemática, conforme apresentado na Seção 3.4.1;
- Paralelamente, tem-se a análise de textura, conforme apresentado na Seção 3.4.2;
- Depois, faz-se a concatenação das camadas resultantes, conforme Seção ;
- Em seguida, define-se os parâmetros de entrada do código *co-clustering*, conforme apresentado na Seção 2.3;
- por fim, realiza-se a investigação dos resultados, conforme apresentado a seguir.

A fim de se verificar o método aplicado, conforme os procedimentos metodológicos apresentados, aplicado a todas as ortofotos, a partir do mosaico de ortofotos, representado pela Figura 9, da ISPRS (ISPRS, 2013), as imagens classificadas pelos métodos k-médias, *co-clustering* e *co-clustering* com refinamento são apresentados na Seção 4.2.6.

Para melhor entendimento do método *co-clustering*, selecionou-se como exemplo, a ortofoto 7_11, representada na Figura 13. A referida ortofoto foi dividida segundo regras de amostragem apresentadas na Seção 2.7.2, e em seguida selecionou-se a imagem exemplo correspondente, conforme Figura 14. A referida imagem particionada foi selecionada como exemplo de aplicação para teste, tendo em vista a presença de diversas feições, como telhados, asfalto, vegetação rasteira, árvores e automóveis.

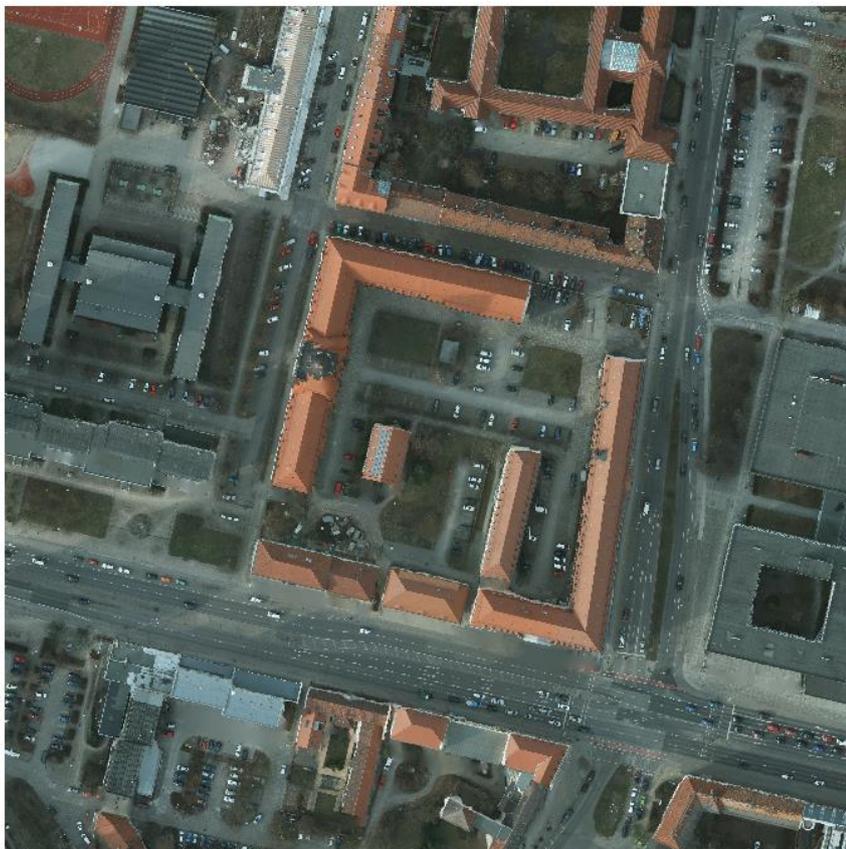


Figura 13 - ortofoto teste

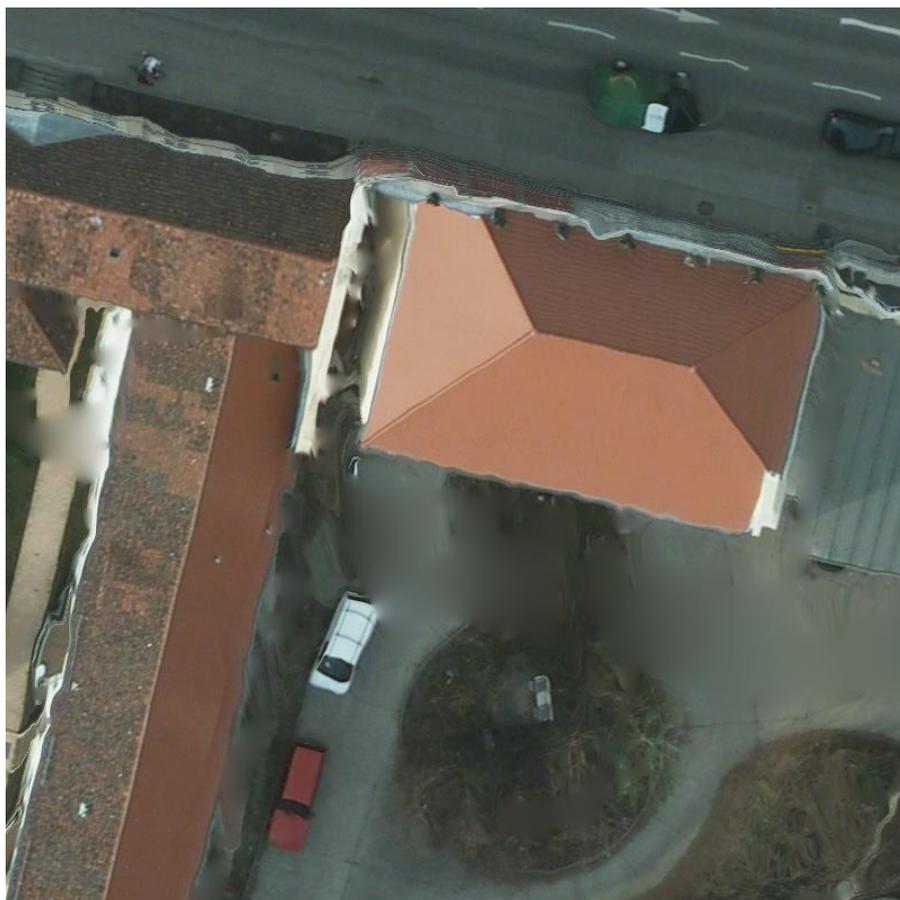


Figura 14 - imagem teste

Após seleção da imagem, realizou-se análise de *co-clustering* proposta, considerando a extração de atributos e incremento do cubo de imagem, conforme Seções 3.4.1 e 3.4.2. Para tal, estabeleceu-se os seguintes parâmetros de teste:

- número de *clusters* de bandas = 3;
- número de *clusters* espaciais = 20;
- máximo de iterações = 100;
- limite de convergência = 0,00001;
- número de diferentes inicializações = 10;
- disco de análise de textura e morfologia matemática = [2, 8, 20].

Após a referida classificação *co-clustering*, cada pixel correspondente da imagem resultante é associado a uma das 5 classes resultantes, conforme Figura 15, sendo cada pixel representado por uma coluna da Figura 15. No eixo horizontal tem-se representado cada um dos pixels, de forma que se tem 640.000 pixels classificados, portanto 640.000

colunas representadas. No eixo vertical, tem-se representado cada uma das “bandas” resultantes, melhor referidas como “camadas”, uma vez que são mais um elemento na dimensão das bandas. Resulta-se em um total de 30 camadas, sendo 3 camadas representadas pelas bandas RGB originais, concatenadas a 9 camadas adicionais de textura e 18 camadas adicionais de morfologia matemática, por haver sido utilizados 3 tamanhos de elemento estruturante para análise de textura e igualmente 3 tamanhos de elemento estruturante para análise de morfologia matemática.

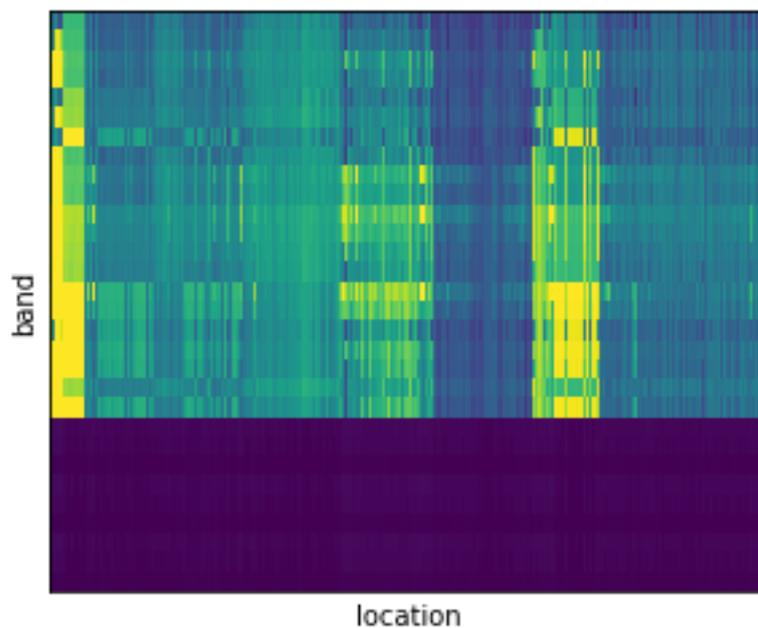


Figura 15 - representação de cada pixel classificado da imagem teste

Os pixels classificados e apresentados pela Figura 15, após serem organizados de forma que pixels semelhantes sejam representados de forma agrupada aos demais da sua classe são apresentados pela Figura 16.

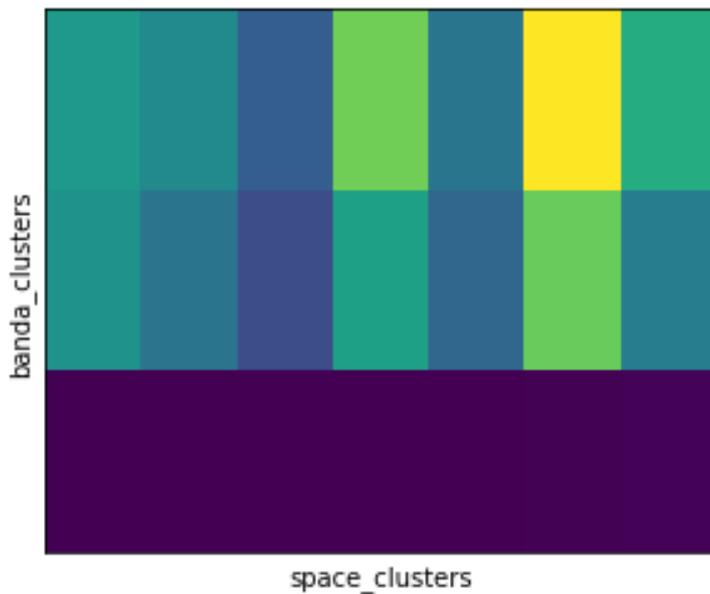


Figura 16 - representação de cada classe da imagem teste

Assim, como os pixels são classificados por intermédio do método *co-clustering*, tem-se também um agrupamento em bandas, ou camadas. No caso da imagem teste, definiu-se um número de *clusters* em bandas iguais a 3, de forma que cada uma das referidas 30 camadas existentes é classificada em uma das três classes de bandas, ou camadas. A representação gráfica dessa distribuição das 3 classes resultantes da classificação em bandas, ou camadas, é dada pela Figura 17. Tem-se, ainda, representado um histograma em que se apresenta o número de bandas em cada um dos três *clusters*. Nota-se, a partir do histograma da Figura 17, a distribuição do agrupamento em bandas, resultando em 9 bandas no agrupamento 1, 13 bandas no agrupamento 2 e 8 bandas no agrupamento 3.

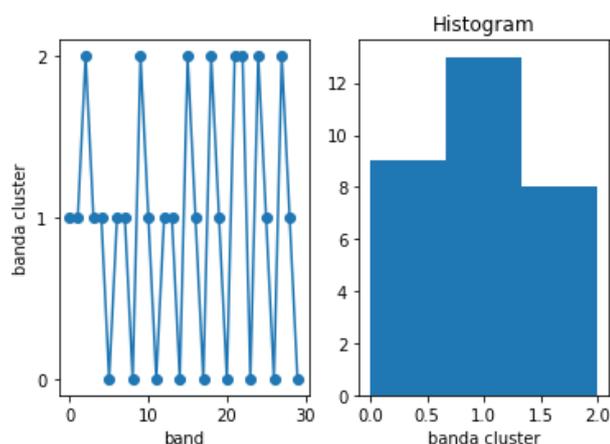


Figura 17 - clusters de bandas resultante do co-clustering

A fim de se verificar visualmente o resultado da classificação, a partir da imagem representada pela Figura 15, remodela-se a apresentação de cada um dos 640.000 pixels, de forma a se representar cada pixel segundo sua posição correta na geometria da imagem original, conforme Figura 18, à esquerda. Para tal, utiliza-se o resultado do método *co-clustering*, relativo ao agrupamento de linhas, no caso em questão, trata-se justamente do agrupamento na dimensão dos pixels, conforme Seção 3.4.5.

Para fins de comparação visual, a verdade de campo considerada é dada pela imagem classificada manualmente disponibilizada pela ISPRS (ISPRS, 2022), conforme Figura 18, à direita, correspondente à imagem de teste considerada. A partir da referida verdade de campo é realizado o cálculo dos índices de similaridade propostos no contexto deste trabalho. Os resultados são apresentados na Seção 0.

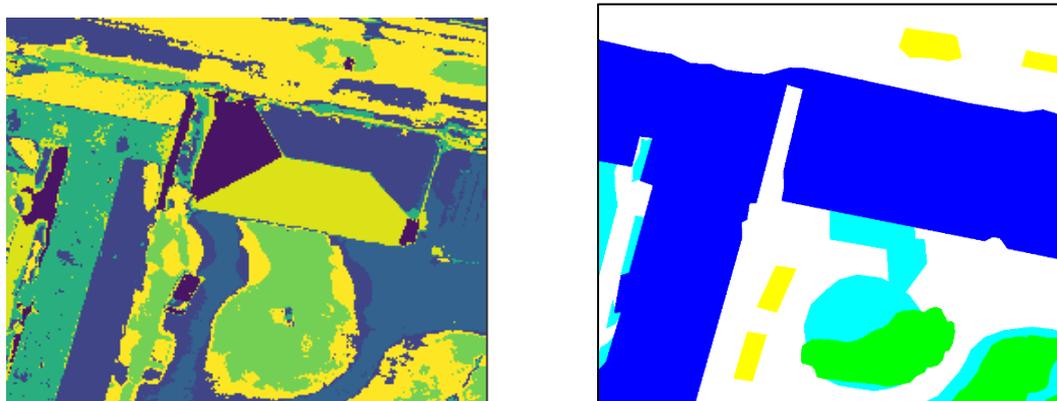


Figura 18 – figura da esquerda representa a imagem teste classificada pelo método co-

clustering; figura da direita representa imagem teste "verdade" obtida por classificação supervisionada, disponibilizada pela ISPRS (ISPRS, 2022)

4.2 RESULTADO

Para realizar os testes, foram processadas todas as imagens referentes a cidade de Potsdam, na Alemanha, disponibilizadas pela ISPRS, de 5cm de resolução espacial, representadas e divididas em ortofotos, conforme Figura 9.

Para tal, dividiu-se cada ortofoto, conforme regra de amostragem prevista na norma ET-CQDG, para Mapeamento Sistemático Topográfico em Grandes Escalas do Sistema Cartográfico Nacional, SCN, conforme Seção 2.7.2.

Após dividida cada uma das referidas 38 ortofotos em 49 imagens menores, conforme regra de amostragem apresentada na Seção 2.7.2, conferiu-se um total de 1862 imagens de altíssima resolução, de 800 x 800 pixels, portanto um total de 2,979,200 m² de área estudada.

Para cada uma das 1862 imagens geradas, realizou-se o processamento proposto, dividindo-se o resultado e correspondentes estatísticas em três etapas:

1. Processamento por método clássico de classificação não supervisionada K-médias;
2. Processamento por método *co-clustering*;
3. Processamento por método integrado com análises de *co-clustering*, textura e morfologia matemática, e posterior refinamento pelo método K-médias.

Em seguida, para cada uma das três etapas apresentadas, avalia-se a qualidade do resultado a partir dos índices Jaccard, Rand e Rand Ajustado. As séries de resultados percentuais que variam entre 0 e 1 são apresentadas em distribuição definidas para os seguintes intervalos:

- entre 0 e 0,2;
- entre 0,2 e 0,4;
- entre 0,4 e 0,6;
- entre 0,6 e 0,8;
- entre 0,8 e 1.

Seguindo-se essa lógica, são apresentadas a seguir as distribuições dos índices de similaridade propostos e calculados para a série de ortofotos para cada método considerado.

Para o desenvolvimento do método *co-clustering* refinado, o número de classes foi calculado automaticamente para o refinamento pelo método K-médias, conforme solução apresentada por (NATTINO *et al.*, 2021). Nota-se variação do número de classes final para o refinamento K-médias, dado pelo *k-valor*, com valores médios concentrados entre 2,5 e 4, conforme gráfico da Figura 19. Disso, nota-se que o método de refinamento K-médias evidenciou separação espectral considerável com número de classes variando entre 2 e 4, em valores inteiros, após resultado da classificação pelo método *co-clustering*.

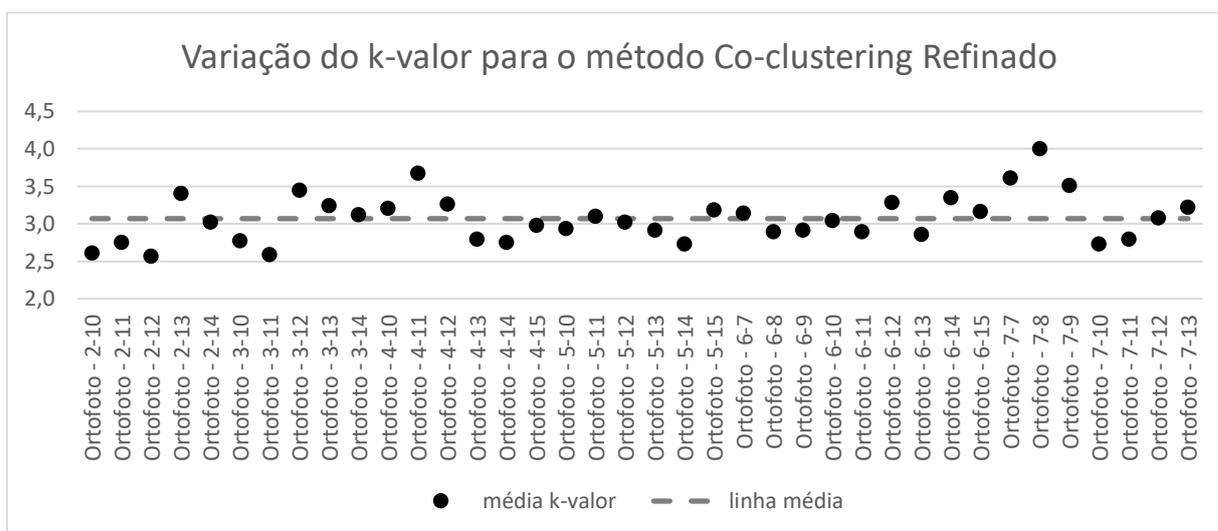


Figura 19 - Variação do *k-valor* para o método *Co-clustering* refinado

Aplicando-se os métodos de classificação não supervisionada “K-médias”, “*co-clustering*” e “*co-clustering* refinado” a todas as 1862 imagens geradas, tem-se por resultado as medidas de similaridade dadas pelos Índice de Jaccard, Índice de Rand e Índice de Rand Ajustado, conforme apresentado por completo nos Apêndice 3, 4 e 5. Destaca-se que o método K-médias foi aplicado ao cubo RGB de imagem original, cujos resultados seguem apresentado no Apêndice 3..

4.2.1 Comparação de resultados – Índice de Jaccard

Nota-se, a partir dos valores apresentados na Figura 20, imagem (a), que o pior valor médio encontrado para o Índice de Jaccard foi relativo ao método de classificação não supervisionada K-médias. Nota-se uma melhora nos resultados da distribuição do

índice relativo ao método *co-clustering*, imagem (b) em comparação com a imagem (a), e uma melhora mais evidente relativa ao método *co-clustering* com refinamento, imagem (c), em comparação com as demais.

A distribuição da média do índice para os referidos métodos pode ser visualmente verificada pelo gráfico, da Figura 20.

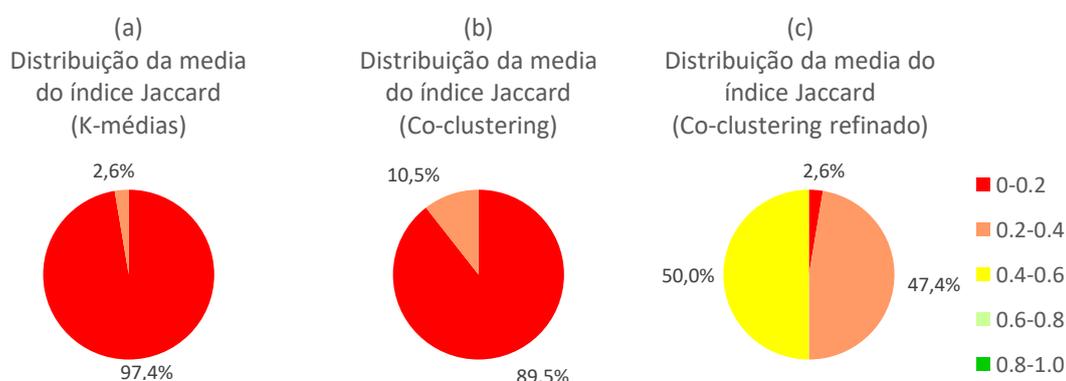


Figura 20 – distribuição da média do Índice de Jaccard para os diferentes métodos: (a) K-médias; (b) Co-clustering; (c) Co-clustering refinado

De forma mais detalhada, avaliando-se os resultados, nota-se para a imagem (a) que 97,4% dos valores relativos ao Índice de Jaccard encontra-se no intervalo $[0 - 0,2]$ e 2,6% dos resultados encontra-se no intervalo $[0,2 - 0,4]$. Para o referido índice, valores próximos a zero representam maior independência entre os resultados, o que significa uma classificação de pior resultado.

Nota-se para a imagem (b) que 89,5% dos valores relativos ao Índice de Jaccard encontra-se no intervalo $[0 - 0,2]$ e 10,5% dos referidos resultados encontra-se no intervalo $[0,2 - 0,4]$.

Por fim, nota-se para a imagem (c) que 50% dos valores relativos ao Índice de Jaccard encontra-se no intervalo $[0,4 - 0,6]$, 47,4% no intervalo $[0,2 - 0,4]$ e 2,6% no intervalo $[0 - 0,2]$. Percebe-se, a partir dos resultados representados pela Figura 20, uma melhora considerável em relação à distribuição do índice de Jaccard, para o método *co-clustering* com refinamento K-médias, em comparação aos demais resultados apresentados, de forma que se tem resultados para o intervalo definido mais próximos de 1.

Destaca-se, a partir da definição do índice de Jaccard, que o mesmo considera apenas os valores verdadeiros positivos, ou seja, os pares de pixels que estão no mesmo *cluster* e na mesma classe, quando em comparação com a verdade considerada.

4.2.2 Comparação de resultados – Índice de Rand

Nota-se, a partir dos valores apresentados na Figura 21, que o pior valor médio encontrado para o Índice de Rand foi relativo ao método de classificação não supervisionada K-médias, imagem (a), com a presença de 76% dos valores relativos ao referido Índice encontra-se no intervalo $[0,6 - 0,8]$, 21% no intervalo $[0,4 - 0,6]$ e 3% no intervalo $[0,2 - 0,4]$.

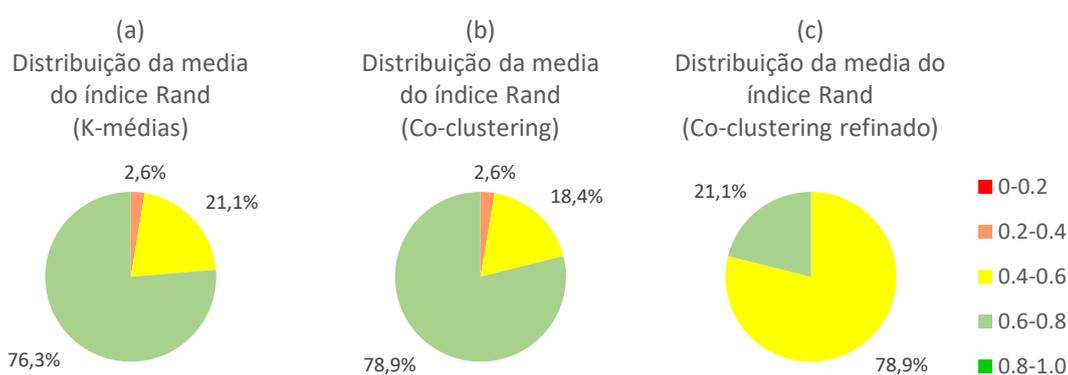


Figura 21 - distribuição da média do Índice de Rand para os diferentes métodos: (a) K-médias; (b) Co-clustering; (c) Co-clustering refinado

Na sequência, nota-se uma melhora sutil nos resultados da distribuição do índice relativo ao método *co-clustering*, imagem (b) em comparação com a imagem (a), e uma distribuição mais concentrada nos valores médios próximos de 0,5, relativa ao método *co-clustering* com refinamento, imagem (c), em comparação com as demais.

Destaca-se, a partir da definição do índice de Rand, melhora, método a método, nos resultados para o referido índice, Figura 20, quando comparadas aos resultados obtidos para o índice de Jaccard, Figura 21. Tal fato é coerente com a definição do Índice de Rand, uma vez que se considera os valores verdadeiros negativos para o seu cálculo, conferindo assim maior precisão no resultado relativo a agrupamento. Destaca-se que o valor ideal 1 (um) para o referido índice representa resultados idênticos para as classificações consideradas.

4.2.3 Comparação de resultados – Índice de Rand Ajustado

Por fim, em relação ao Índice de Rand Ajustado, nota-se que 97,4% dos valores relativos ao referido Índice encontra-se concentrado no intervalo $[0,2 - 0,4]$ e 2,6% no intervalo $[0 - 0,2]$, conforme Figura 22, imagens (b) e (c).

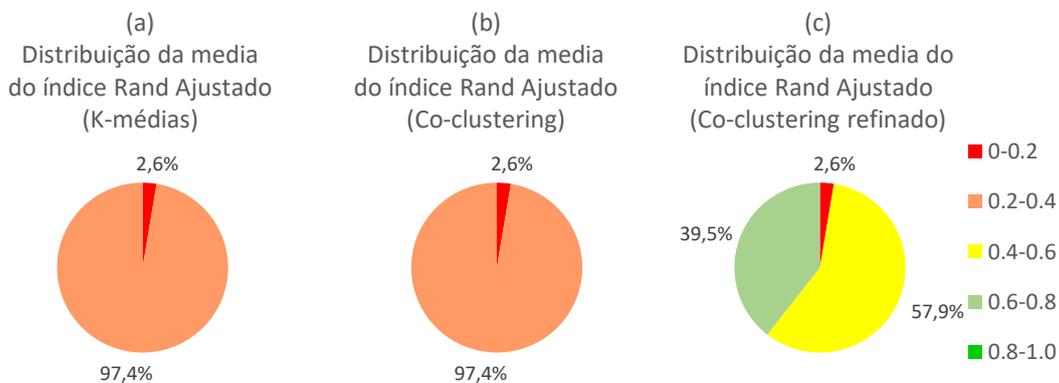


Figura 22 - distribuição da média do Índice de Rand Ajustado para os diferentes métodos: (a) K-médias; (b) Co-clustering; (c) Co-clustering refinado

Nota-se, pela imagem (c), que 39,5% dos valores relativos ao referido Índice encontra-se no intervalo $[0,6 - 0,8]$, 57,9% no intervalo $[0,4 - 0,6]$ e 2,6% no intervalo $[0 - 0,2]$. Nota-se melhora considerável dos resultados, quando em comparação com as imagens (a) e (b), que representam a aplicação dos métodos k-médias e *co-clustering* sem refinamento.

Destaca-se o fato de o Índice de Rand Ajustado ser o mais adequado para avaliação de métodos de classificação não supervisionada, uma vez que é uma evolução do Índice de Rand, pois o índice de Rand Ajustado se mostra mais sensível ao resultado, uma vez que o intervalo de possíveis valores é estendido de -1 a 1. Com isso, o índice em questão evidencia em seu resultado possíveis valores aleatórios, dados por resultados negativos.

Corroborar-se o uso do refinamento K-médias para o método *co-clustering* a partir dos dados apresentados no Quadro 1, visualmente verificados pela Figura 23. Nota-se, considerável melhora na distribuição dos índices, com maior distribuição dos resultados entre os intervalos considerados, em que se verifica maior incidência de valores de distribuição de índices em intervalos mais próximos de 1. O valor de um índice de medida de similaridade igual a 1 (um) representaria a classificação idealmente perfeita, ou seja,

idêntica à verdade de campo. No caso deste trabalho, a verdade de campo considerada é a classificação disponibilizada pela ISPRS.

Quadro 1 - Distribuição em intervalos para os métodos Co-clustering e Co-clustering refinado

Distribuição em intervalos para os métodos Co-clustering e Co-clustering refinado						
Distribuição do índice	Jaccard (Co-clustering)	Jaccard (Co-clustering refinado)	Rand (Co-clustering)	Rand (Co-clustering refinado)	Rand Ajustado (Co-clustering)	Rand Ajustado (Co-clustering refinado)
0 - 0,2	79,1%	8,5%	0,4%	0,5%	3,1%	2,6%
0,2 - 0,4	21,3%	52,3%	4,1%	12,9%	92,3%	4,7%
0,4 - 0,6	0,2%	31,1%	20,3%	50,7%	4,5%	57,3%
0,6 - 0,8	0,0%	6,1%	72,4%	31,8%	0,1%	28,5%
0,8 - 1,0	0,0%	2,0%	2,7%	4,1%	0,0%	6,9%

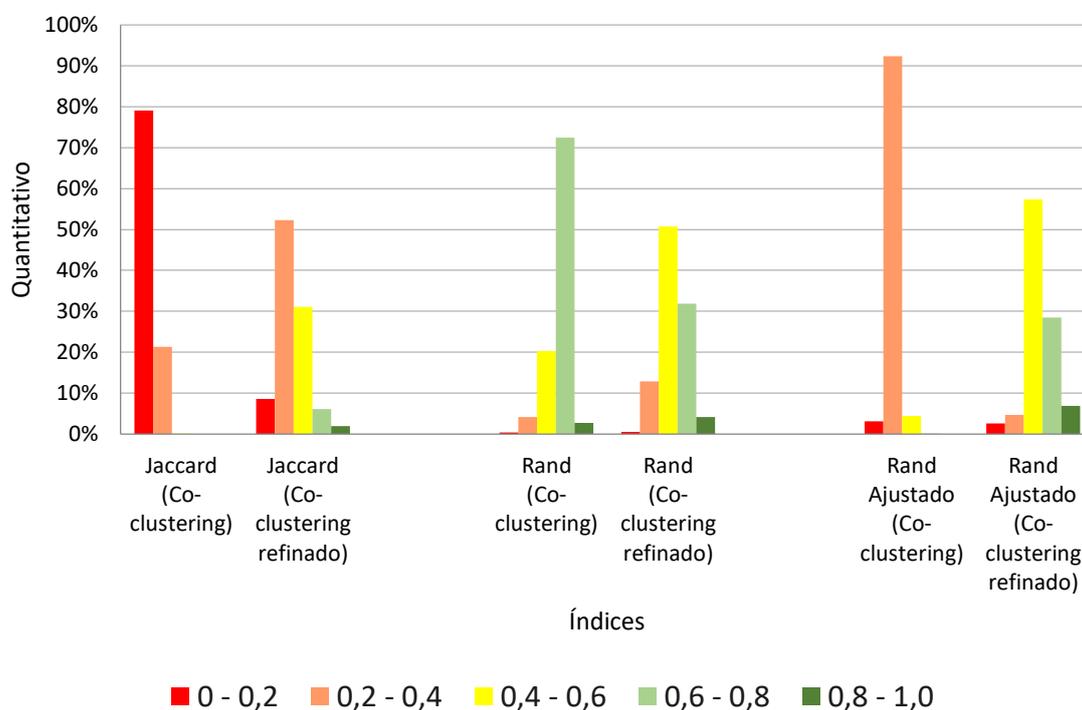


Figura 23 - Distribuição em intervalos para os métodos Co-clustering e Co-clustering refinado

4.2.4 Análise conjunta dos resultados

Considerando-se, de forma geral, as medidas de similaridade dadas pelos Índices considerados neste trabalho, pode-se avaliar de forma conjunta os resultados encontrados e apresentados nos Apêndice 3, Apêndice 4 e Apêndice 5.

Nota-se, inicialmente de forma agrupada, a partir do gráfico da Figura 24, considerável melhora para os Índice de Jaccard e Índice de Rand Ajustado, ao se aplicar o método *co-clustering* integrado com estudos de morfologia matemática e análise de textura, com posterior refinamento com técnica K-médias. Nota-se, também, que visualmente não há ganho considerável de melhora dos Índice de Jaccard e do Índice de Rand Ajustado, comparando-se puramente o método K-médias com o método *co-clustering* sem refinamento. Nota-se, ainda a partir do gráfico da Figura 24, que para o Índice de Rand não se observa variação considerável de resultado entre os métodos de classificação considerados.

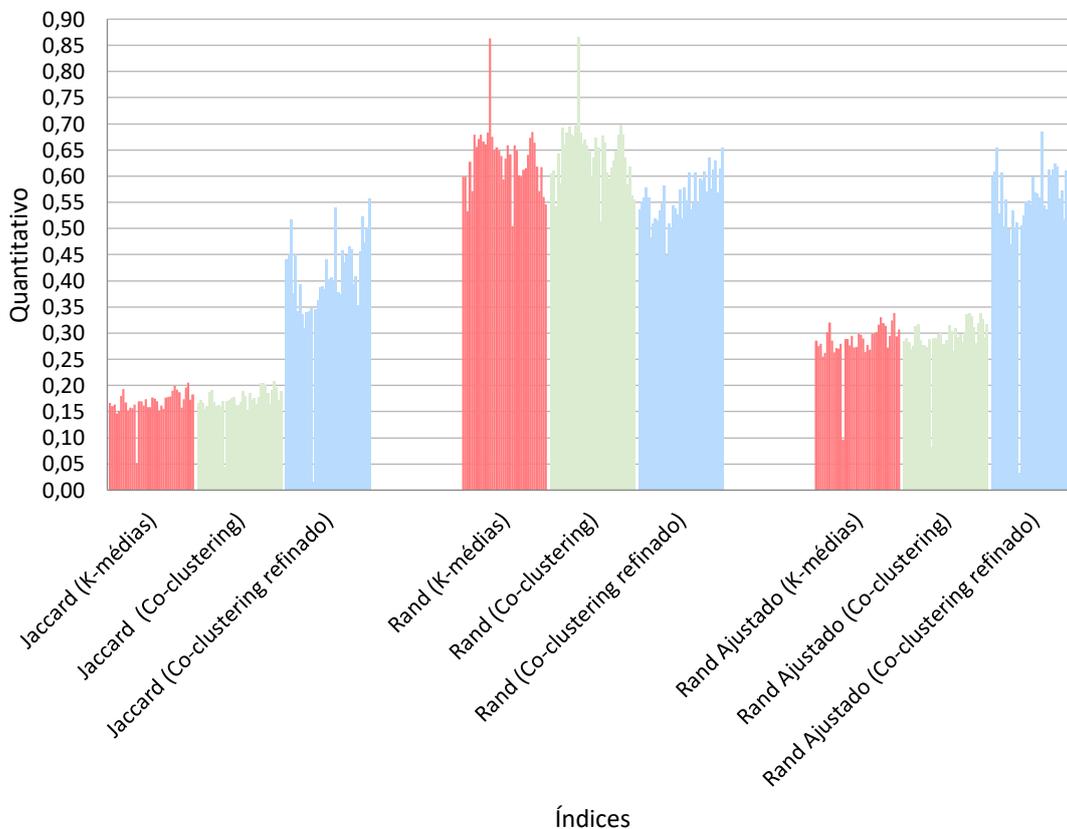


Figura 24 - Resultado dos índices, por método, para todos as ortofotos

A partir das medidas de similaridade dadas pelos Índice de Jaccard, Índice de Rand e Índice de Rand Ajustado, considerados neste trabalho, e calculados a partir de classificação não supervisionadas para os métodos K-médias aplicado ao cubo de imagem RGB original, *co-clustering* sem refinamento e *co-clustering* com refinamento, apresentados respectivamente nos Apêndice 3, Apêndice 4 e Apêndice 5, pode-se calcular as estatísticas dos resultados para o total das 1862 imagens conforme resultado condensado no Quadro 2.

Quadro 2 - Estatísticas dos índices considerando todas as ortofotos

Estatísticas dos índices considerando todas as ortofotos					
Índice	Máximo	Mínimo	Média	Desvio padrão	Variância
Jaccard (K-médias)	0,205	0,052	0,168	0,024	0,001
Jaccard (Co-clustering)	0,208	0,044	0,173	0,026	0,001
Jaccard (Co-clustering refinado)	0,557	0,017	0,405	0,088	0,008
Rand (K-médias)	0,863	0,504	0,634	0,058	0,003
Rand (Co-clustering)	0,867	0,513	0,644	0,059	0,003
Rand (Co-clustering refinado)	0,655	0,448	0,560	0,044	0,002
Rand Ajustado (K-médias)	0,338	0,097	0,285	0,037	0,001
Rand Ajustado (Co-clustering)	0,338	0,083	0,292	0,040	0,002
Rand Ajustado (Co-clustering refinado)	0,699	0,032	0,561	0,103	0,011

Nota-se, a partir do Quadro 2, que o pior valor médio encontrado para os índices estudados foi o Índice de Jaccard. Tal fato é esperado por ser coerente com a própria definição dos índices, uma vez que o índice de Jaccard considera apenas os valores verdadeiros positivos, ou seja, os pares de pixels que estão no mesmo *cluster* e na mesma classe, quando em comparação com a verdade considerada.

Avaliando-se os resultados apresentados no Quadro 2, nota-se que as medidas dos Índice de Jaccard e Índice de Rand Ajustado para o método de *co-clustering* refinado apresentam maiores valores calculados de variância, o que evidencia maior dispersão de valores em torno do valor médio, bem como de desvio padrão, que representa valores bem distribuídos em torno da média. Tais resultados evidenciam maior sensibilidade dos Índice de Jaccard e Índice de Rand Ajustado em relação a heterogeneidade das amostras

após classificação não supervisionada pelo método *co-clustering* refinado, dada pela formação do agrupamento resultante.

Os valores apresentados no Quadro 2 podem ser visualmente verificados pelo gráfico da Figura 25.

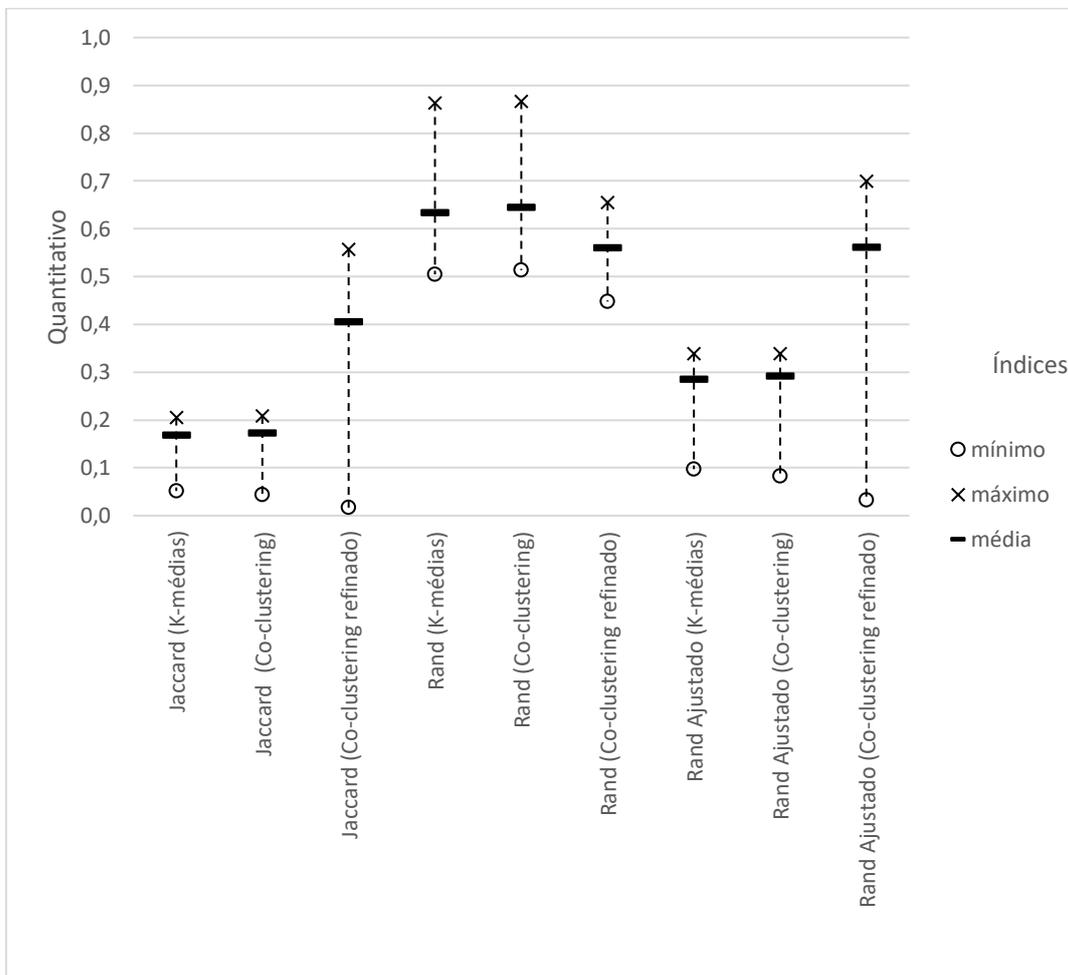


Figura 25 - distribuição de máximo, mínimo e média dos índices considerando todas as ortofotos

Por outro lado, espera-se que o índice Rand seja, de fato, superior ao índice de Jaccard uma vez que considera, em seu cálculo, os valores verdadeiros negativos, ou seja, os pares de pixels que estão em diferentes *cluster* e diferentes classes.

Por sua vez, o Índice de Rand Ajustado também considera em seu cálculo os valores verdadeiros negativos, com a diferença de matematicamente alterar o seu *range* de variação para valores entre -1 e 1, sendo esperados valores positivos, uma vez que valores negativos indicam comportamento aleatório, conforme apresentado na Seção

2.7.4.2. Há grande aceitação da comunidade acadêmica atualmente para uso desse índice, conforme apresentado com mais detalhes na Seção 2.7.4.

Pode-se notar, ainda, a partir do Quadro 2, e a partir do gráfico, da Figura 25, o comportamento dos valores máximos, mínimos e médios para cada um dos índices de similaridade calculados.

Nota-se, a partir da Figura 26, em relação ao Índice de Jaccard, que se encontrou melhor resultado em 74% das imagens classificadas pelo método *co-clustering* sem refinamento, quando em comparação ao método K-médias; melhor resultado em 97% das imagens classificadas pelo método *co-clustering* refinado quando em comparação ao método K-médias; e igualmente melhor resultado em 97% das imagens classificadas pelo método *co-clustering* refinado quando em comparação ao método *co-clustering* sem refinamento. Em relação ao Índice de Rand, nota-se que se encontrou melhor resultado em 97% das imagens classificadas pelo método *co-clustering* sem refinamento, quando em comparação ao método K-médias; melhor resultado em 84% das imagens classificadas pelo método K-médias quando em comparação ao método *co-clustering* refinado; e igualmente melhor resultado em 87% das imagens classificadas pelo método *co-clustering* sem refinamento quando em comparação ao método *co-clustering* refinado. Em relação ao Índice de Rand Ajustado, igualmente ao encontrado para o Índice de Jaccard nota-se que se encontrou melhor resultado em 74% das imagens classificadas pelo método *co-clustering* sem refinamento, quando em comparação ao método K-médias; melhor resultado em 97% das imagens classificadas pelo método *co-clustering* refinado quando em comparação ao método K-médias; e igualmente melhor resultado em 97% das imagens classificadas pelo método *co-clustering* refinado quando em comparação ao método *co-clustering* sem refinamento.

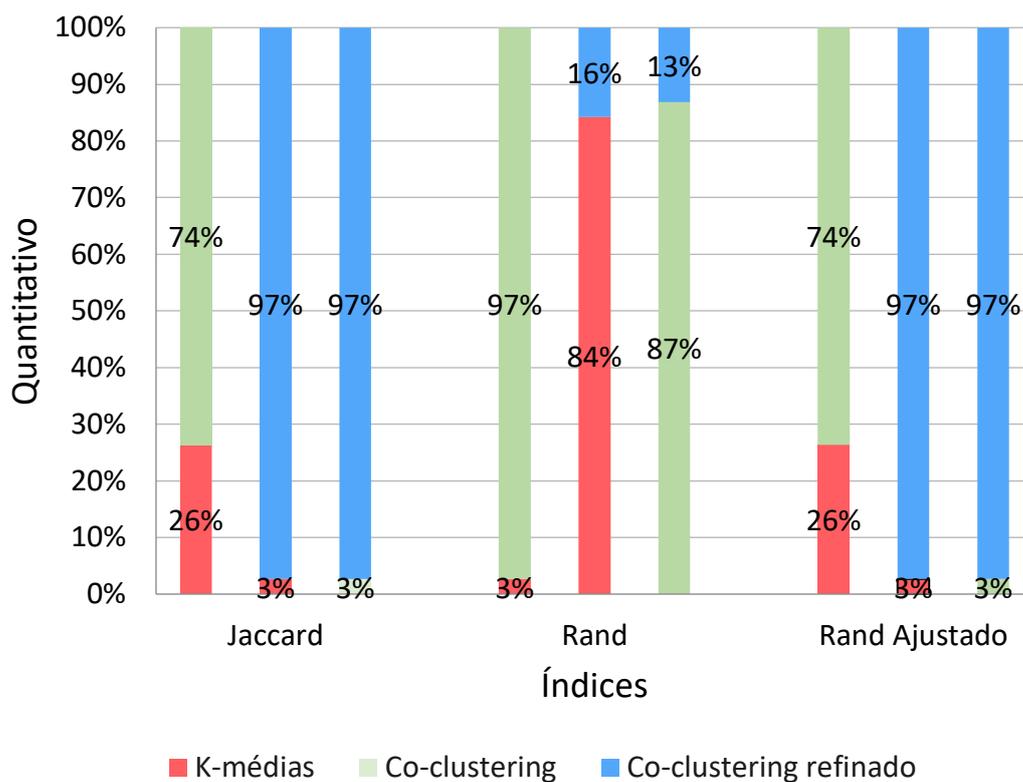


Figura 26 - Comparativo Percentual do melhor método a partir das medidas de similaridade consideradas

Em relação ao Índice de Jaccard, a partir dos valores apresentados no Quadro 3, representados pela Figura 27, nota-se que, em média, os valores encontrados para o referido índice calculado a partir do método *co-clustering* refinado são 2,38 vezes melhores quando comparados com o método de classificação não supervisionada K-médias. Em complemento, o referido índice, em média, calculado a partir do método *co-clustering* sem refinamento possui resultado menos satisfatório, sendo 1,026 vezes melhor quando comparado com o método de classificação não supervisionada K-médias.

Avaliando-se os resultados apresentados no Quadro 3, nota-se que as medidas de melhora, em relação ao método K-médias, do Índice de Jaccard calculado a partir do método de *co-clustering* refinado apresentam maior valor calculado de variância, o que evidencia maior dispersão de valores em torno do valor médio, bem como de desvio padrão, que representa valores bem distribuídos em torno da média. Tais resultados evidenciam maior sensibilidade na formação dos agrupamentos do método *co-clustering*

refinado pelo método K-médias, em relação a heterogeneidade das amostras, em comparação ao método *co-clustering* sem refinamento.

Quadro 3 - Comparativo de melhora em relação ao método K-médias - Índice de Jaccard

Comparativo de melhora em relação ao método K-médias - Índice de Jaccard		
Ortofoto	Comparativo (Co-clustering)	Comparativo (Co-clustering refinado)
Ortofoto - 2-10	1,000	2,638
Ortofoto - 2-11	1,068	2,809
Ortofoto - 2-12	1,023	3,164
Ortofoto - 2-13	1,061	2,567
Ortofoto - 2-14	1,064	2,979
Ortofoto - 3-10	1,042	1,904
Ortofoto - 3-11	0,988	2,031
Ortofoto - 3-12	1,006	2,005
Ortofoto - 3-13	1,057	2,036
Ortofoto - 3-14	1,026	2,156
Ortofoto - 4-10	1,017	2,184
Ortofoto - 4-11	1,039	2,131
Ortofoto - 4-12	0,846	0,323
Ortofoto - 4-13	1,004	2,034
Ortofoto - 4-14	1,009	2,132
Ortofoto - 4-15	1,095	2,408
Ortofoto - 5-10	1,019	2,239
Ortofoto - 5-11	1,028	2,421
Ortofoto - 5-12	1,027	2,777
Ortofoto - 5-13	0,949	2,284
Ortofoto - 5-14	1,081	2,318
Ortofoto - 5-15	1,055	2,342
Ortofoto - 6-7	1,011	3,549
Ortofoto - 6-8	1,144	2,343
Ortofoto - 6-9	1,105	2,405
Ortofoto - 6-10	0,991	2,587
Ortofoto - 6-11	0,928	2,454
Ortofoto - 6-12	0,991	2,506
Ortofoto - 6-13	1,079	2,454
Ortofoto - 6-14	1,031	2,311
Ortofoto - 6-15	1,045	2,054
Ortofoto - 7-7	0,989	2,183
Ortofoto - 7-8	1,037	2,241

Ortofoto - 7-9	1,096	2,621
Ortofoto - 7-10	1,059	2,662
Ortofoto - 7-11	0,957	2,309
Ortofoto - 7-12	1,000	2,896
Ortofoto - 7-13	1,041	3,051
Média	1,026	2,382
Desvio Padrão	0,053	0,489
Variância	0,003	0,239

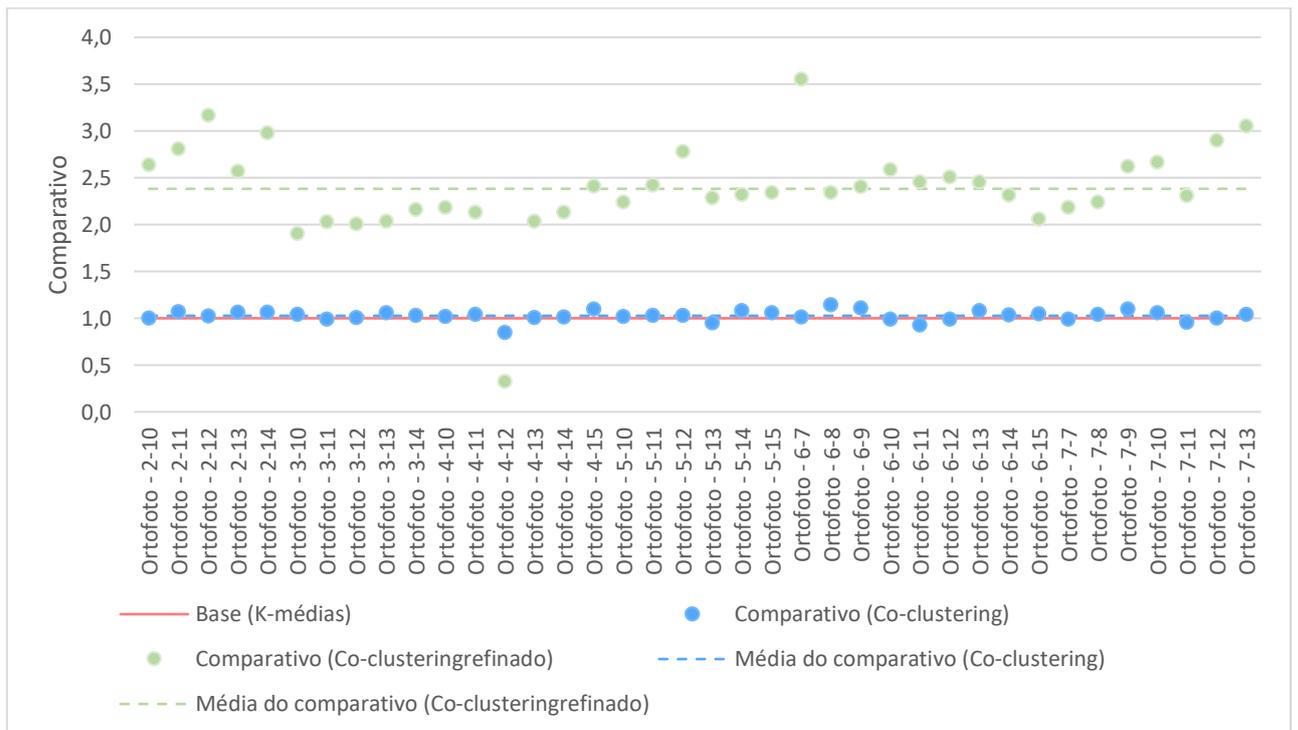


Figura 27 - Comparativo de melhora em relação ao método K-médias - Índice de Jaccard

Em relação ao Índice de Rand, a partir dos valores apresentados no Quadro 4, representados pela Figura 28, nota-se que, em média, os valores encontrados para o referido índice calculado a partir do método *co-clustering* refinado são piores quando comparados com o método de classificação não supervisionada K-médias, com razão igual a 0,89. Percebe-se que a referida razão é próxima de 1, portanto o resultado médio da classificação *co-clustering* sem refinamento é muito próximo ao encontrado para a classificação não supervisionada K-médias. Esse resultado em si evidencia a potencialidade do método de classificação não supervisionado *co-clustering* em que os

resultados apontam qualidade semelhante àquela obtida a partir do método clássico K-médias, amplamente considerado em análise de qualidade de produtos semelhantes.

Quadro 4 - Comparativo de melhora em relação ao método K-médias - Índice de Rand

Comparativo de melhora em relação ao método K-médias - Índice de Rand		
Ortofoto	Comparativo (Co-clustering)	Comparativo (Co-clustering refinado)
Ortofoto - 2-10	1,010	0,897
Ortofoto - 2-11	1,021	0,922
Ortofoto - 2-12	1,017	1,049
Ortofoto - 2-13	1,026	0,922
Ortofoto - 2-14	1,026	0,979
Ortofoto - 3-10	1,019	0,709
Ortofoto - 3-11	1,007	0,776
Ortofoto - 3-12	1,018	0,774
Ortofoto - 3-13	1,025	0,759
Ortofoto - 3-14	1,020	0,802
Ortofoto - 4-10	1,021	0,828
Ortofoto - 4-11	1,021	0,853
Ortofoto - 4-12	1,004	0,519
Ortofoto - 4-13	1,011	0,754
Ortofoto - 4-14	1,017	0,772
Ortofoto - 4-15	1,023	0,830
Ortofoto - 5-10	1,014	0,831
Ortofoto - 5-11	1,013	0,826
Ortofoto - 5-12	1,014	0,968
Ortofoto - 5-13	1,003	0,819
Ortofoto - 5-14	1,022	0,878
Ortofoto - 5-15	1,019	0,863
Ortofoto - 6-7	1,017	1,205
Ortofoto - 6-8	1,027	0,814
Ortofoto - 6-9	1,023	0,845
Ortofoto - 6-10	1,010	1,009
Ortofoto - 6-11	0,994	0,919
Ortofoto - 6-12	1,006	0,973
Ortofoto - 6-13	1,023	0,962
Ortofoto - 6-14	1,012	0,949
Ortofoto - 6-15	1,009	0,848
Ortofoto - 7-7	1,019	0,928
Ortofoto - 7-8	1,022	0,869
Ortofoto - 7-9	1,027	0,990

Ortofoto - 7-10	1,022	1,103
Ortofoto - 7-11	1,002	0,921
Ortofoto - 7-12	1,006	1,097
Ortofoto - 7-13	1,016	1,199
Média	1,016	0,894
Desvio Padrão	0,008	0,131
Variância	0,000	0,017

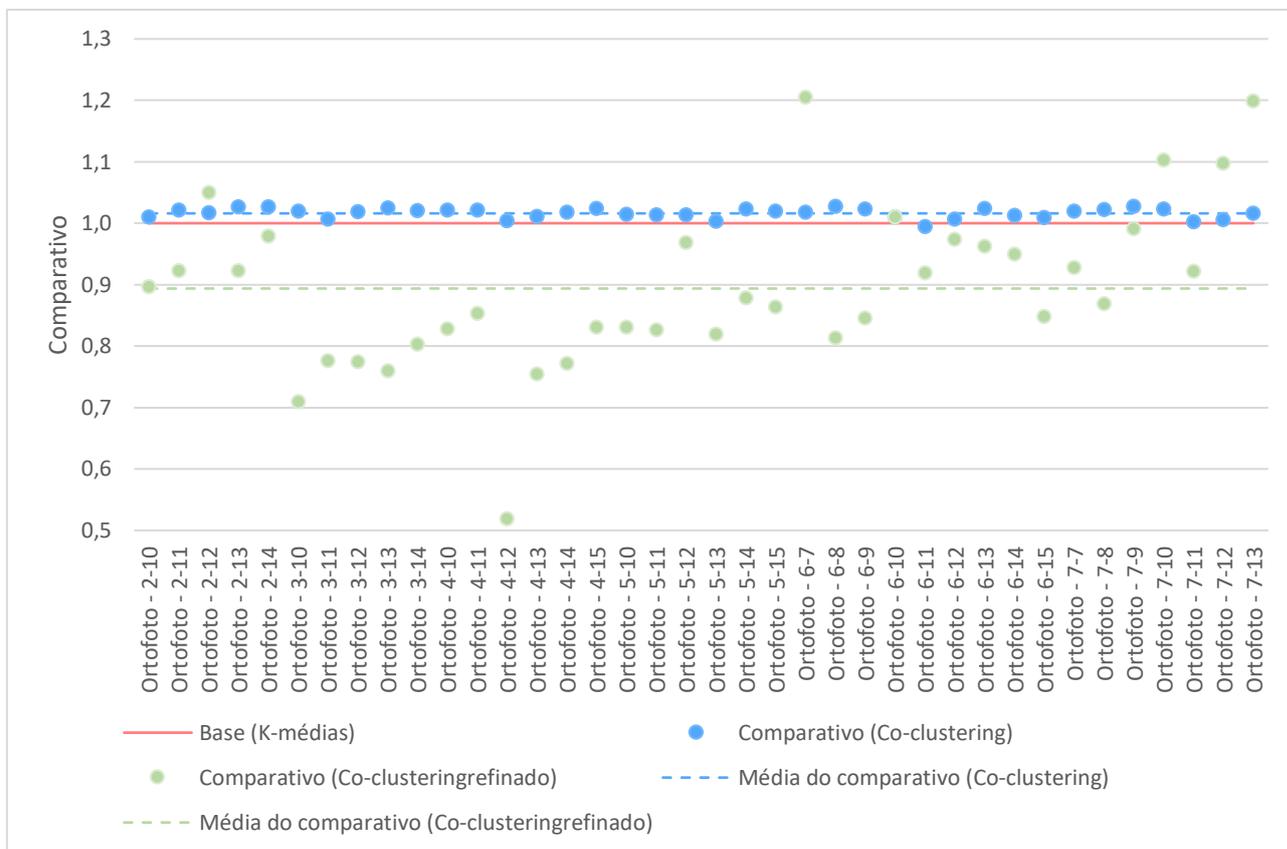


Figura 28 - Comparativo de melhora em relação ao método K-médias - Índice de Rand

Por fim, em relação ao Índice de Rand Ajustado, referência de qualidade mais adequada para avaliação de classificação não supervisionada, a partir dos valores apresentados no Quadro 5, visualmente representados pela Figura 29, nota-se que, em média, os valores encontrados para o referido índice calculado a partir do método *co-clustering* refinado são 1,94 vezes melhores quando comparados com o método de classificação não supervisionada K-médias.

Quadro 5 - Comparativo de melhora em relação ao método K-médias - Índice de Rand

Ajustado

Comparativo de melhora em relação ao método K-médias - Índice de Rand Ajustado		
Ortofoto	Comparativo (Co-clustering)	Comparativo (Co-clustering refinado)
Ortofoto - 2-10	1,000	2,102
Ortofoto - 2-11	1,056	2,212
Ortofoto - 2-12	1,013	2,339
Ortofoto - 2-13	1,052	2,069
Ortofoto - 2-14	1,052	2,318
Ortofoto - 3-10	1,034	1,668
Ortofoto - 3-11	0,992	1,732
Ortofoto - 3-12	1,003	1,743
Ortofoto - 3-13	1,050	1,783
Ortofoto - 3-14	1,022	1,969
Ortofoto - 4-10	1,012	1,852
Ortofoto - 4-11	1,033	1,830
Ortofoto - 4-12	0,856	0,335
Ortofoto - 4-13	1,004	1,754
Ortofoto - 4-14	1,007	1,814
Ortofoto - 4-15	1,079	1,992
Ortofoto - 5-10	1,016	1,883
Ortofoto - 5-11	1,024	2,001
Ortofoto - 5-12	1,022	2,193
Ortofoto - 5-13	0,956	1,902
Ortofoto - 5-14	1,065	1,912
Ortofoto - 5-15	1,048	1,930
Ortofoto - 6-7	1,008	2,597
Ortofoto - 6-8	1,117	1,960
Ortofoto - 6-9	1,086	1,995
Ortofoto - 6-10	0,994	2,046
Ortofoto - 6-11	0,940	1,999
Ortofoto - 6-12	0,994	2,026
Ortofoto - 6-13	1,062	1,978
Ortofoto - 6-14	1,024	1,873
Ortofoto - 6-15	1,037	1,747
Ortofoto - 7-7	0,991	1,824
Ortofoto - 7-8	1,031	1,902
Ortofoto - 7-9	1,079	2,071
Ortofoto - 7-10	1,044	2,038
Ortofoto - 7-11	0,963	1,874
Ortofoto - 7-12	0,999	2,237

Ortofoto - 7-13	1,030	2,276
Média	1,021	1,942
Desvio Padrão	0,045	0,327
Variância	0,002	0,107

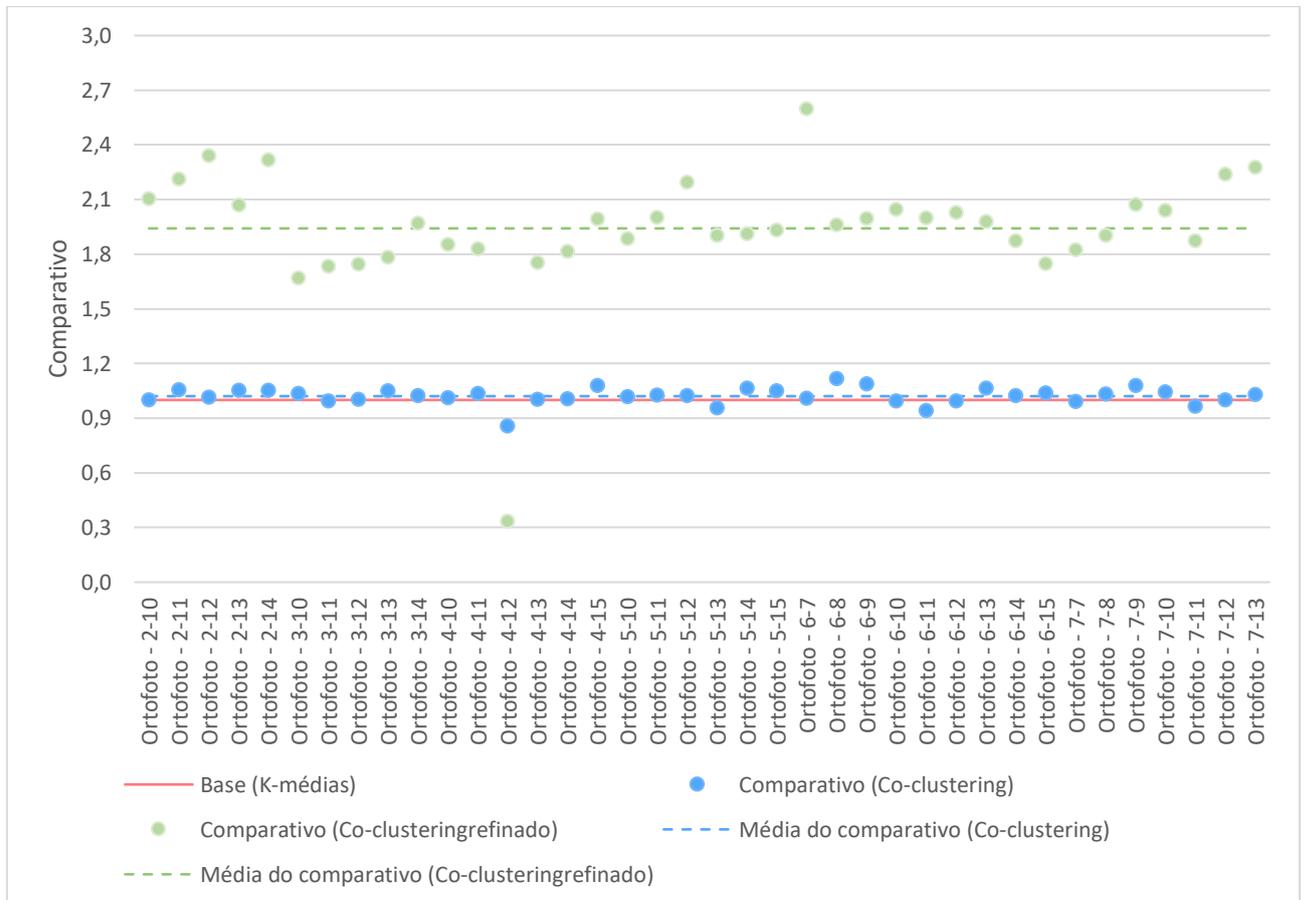


Figura 29 - Comparativo de melhora em relação ao método K-médias - Índice de Rand Ajustado

Avaliando-se os resultados apresentados nos Quadro 3, Quadro 4 e Quadro 5, nota-se que o método *co-clustering* refinado apresenta maiores valores de desvio padrão e variância, maior dispersão em torno da média, quando comparado ao método *co-clustering* sem refinamento, ambos em comparação ao método K-médias. Tais resultados evidenciam, de forma geral, maior sensibilidade do método *co-clustering* refinado em relação a heterogeneidade das amostras e consequente heterogeneidade do resultado da classificação não supervisionada, dada pela formação do agrupamento resultante.

4.2.5 Avaliação quanto ao cubo de imagem integrado com morfologia matemática e textura

Foram processadas todas as imagens referentes a cidade de Potsdam, na Alemanha, disponibilizadas pela ISPRS, representadas e divididas em ortofotos, conforme Figura 9. A fim de se avaliar o impacto de se ter ou não o cubo de imagem original integrado com camadas de informações de morfologia matemática e textura, as referidas imagens foram classificadas pelo método *co-clustering* proposto com e sem prévia integração do cubo de imagem original a informações de morfologia matemática e textura.

O Quadro 6, a seguir, foi construído a partir da razão entre os valores apresentados nos Apêndices 6 e 4. Por sua vez, o Quadro 7 foi construído a partir da razão entre os valores apresentados nos Apêndices 7 e 5.

Quadro 6 - Comparativo de melhora em relação ao cubo de imagem original - método co-clustering sem refinamento

Ortofoto	Jaccard	Rand	Rand Ajustado
Ortofoto - 2-10	1,012	1,006	1,010
Ortofoto - 2-11	1,066	1,011	1,054
Ortofoto - 2-12	1,031	1,005	1,021
Ortofoto - 2-13	1,084	1,012	1,072
Ortofoto - 2-14	1,082	1,019	1,068
Ortofoto - 3-10	1,058	1,010	1,047
Ortofoto - 3-11	0,970	0,995	0,977
Ortofoto - 3-12	1,025	1,007	1,021
Ortofoto - 3-13	1,081	1,012	1,069
Ortofoto - 3-14	1,027	1,009	1,023
Ortofoto - 4-10	1,036	1,010	1,029
Ortofoto - 4-11	1,074	1,010	1,062
Ortofoto - 4-12	0,814	0,994	0,823
Ortofoto - 4-13	1,024	1,000	1,020
Ortofoto - 4-14	1,038	1,010	1,031
Ortofoto - 4-15	1,096	1,014	1,080
Ortofoto - 5-10	1,028	1,006	1,025
Ortofoto - 5-11	1,040	1,006	1,034
Ortofoto - 5-12	1,028	1,006	1,024
Ortofoto - 5-13	0,965	0,996	0,971
Ortofoto - 5-14	1,082	1,015	1,067

Ortofoto - 5-15	1,063	1,014	1,018
Ortofoto - 6-7	1,049	1,016	1,041
Ortofoto - 6-8	1,154	1,021	1,127
Ortofoto - 6-9	1,096	1,014	1,079
Ortofoto - 6-10	1,002	1,004	1,004
Ortofoto - 6-11	0,934	0,986	0,947
Ortofoto - 6-12	0,994	0,999	0,998
Ortofoto - 6-13	1,063	1,014	1,047
Ortofoto - 6-14	1,043	1,004	1,035
Ortofoto - 6-15	1,021	1,001	1,022
Ortofoto - 7-7	1,035	1,010	1,030
Ortofoto - 7-8	1,091	1,007	1,077
Ortofoto - 7-9	1,112	1,020	1,092
Ortofoto - 7-10	1,119	1,029	1,089
Ortofoto - 7-11	0,971	0,996	0,975
Ortofoto - 7-12	0,993	1,000	0,994
Ortofoto - 7-13	1,039	1,008	1,030
Média	1,038	1,008	1,030
Desvio padrão	0,058	0,008	0,050
Variância	0,003	0,000	0,003

Quadro 7 - Comparativo de melhora em relação ao cubo de imagem original - método co-clustering com refinamento

Ortofoto	Jaccard	Rand	Rand Ajustado
Ortofoto - 2-10	0,997	0,932	0,999
Ortofoto - 2-11	1,009	1,023	1,007
Ortofoto - 2-12	0,976	0,926	0,982
Ortofoto - 2-13	1,032	1,223	1,025
Ortofoto - 2-14	0,996	1,023	0,999
Ortofoto - 3-10	0,945	0,943	0,964
Ortofoto - 3-11	0,952	0,913	0,969
Ortofoto - 3-12	1,047	1,215	1,034
Ortofoto - 3-13	1,027	1,186	1,021
Ortofoto - 3-14	1,057	1,222	1,111
Ortofoto - 4-10	1,023	1,162	1,014
Ortofoto - 4-11	1,090	1,311	1,064
Ortofoto - 4-12	1,121	1,104	1,117
Ortofoto - 4-13	1,073	1,184	1,051
Ortofoto - 4-14	0,987	1,042	0,989
Ortofoto - 4-15	1,114	1,138	1,078

Ortofoto - 5-10	1,028	1,074	1,022
Ortofoto - 5-11	1,075	1,155	1,051
Ortofoto - 5-12	1,070	1,212	1,047
Ortofoto - 5-13	1,016	1,060	1,013
Ortofoto - 5-14	1,087	1,182	1,061
Ortofoto - 5-15	1,050	1,157	1,034
Ortofoto - 6-7	0,998	1,002	1,000
Ortofoto - 6-8	1,031	1,062	1,024
Ortofoto - 6-9	1,029	1,082	1,022
Ortofoto - 6-10	1,048	1,154	1,036
Ortofoto - 6-11	1,018	1,054	1,012
Ortofoto - 6-12	1,064	1,136	1,049
Ortofoto - 6-13	1,046	1,076	1,031
Ortofoto - 6-14	1,151	1,248	1,103
Ortofoto - 6-15	1,012	1,071	1,010
Ortofoto - 7-7	1,168	1,267	1,116
Ortofoto - 7-8	1,108	1,372	1,076
Ortofoto - 7-9	1,117	1,229	1,081
Ortofoto - 7-10	1,027	1,116	1,024
Ortofoto - 7-11	1,054	1,032	1,043
Ortofoto - 7-12	1,026	1,102	1,020
Ortofoto - 7-13	1,060	1,137	1,039
Média	1,045	1,119	1,035
Desvio padrão	0,049	0,105	0,038
Variância	0,002	0,011	0,001

Em relação ao Índice de Jaccard, tendo por base os valores apresentados no Quadro 6, nota-se que, em média, os valores encontrados para o referido índice calculado a partir do método *co-clustering* sem refinamento aplicado ao cubo de imagem integrado com camadas de morfologia matemática e textura são 1,038 vezes melhores quando comparados com o método *co-clustering* sem refinamento aplicado ao cubo de imagem original. Em complemento, tendo por base os valores apresentados no Quadro 7, o referido índice, em média, calculado a partir do método *co-clustering* com refinamento pelo método K-médias aplicado ao cubo de imagem integrado com camadas de morfologia matemática e textura possui resultado semelhante, sendo 1,045 vezes melhor quando comparado com o com o método *co-clustering* com refinamento aplicado ao cubo de imagem original.

Em relação ao Índice de Rand, tendo por base os valores apresentados no Quadro 6, nota-se que, em média, os valores encontrados para o referido índice calculado a partir do método *co-clustering* sem refinamento aplicado ao cubo de imagem integrado com camadas de morfologia matemática e textura são 1,008 vezes melhores quando comparados com o método *co-clustering* sem refinamento aplicado ao cubo de imagem original. Em complemento, tendo por base os valores apresentados no Quadro 7, o referido índice, em média, calculado a partir do método *co-clustering* com refinamento pelo método K-médias aplicado ao cubo de imagem integrado com camadas de morfologia matemática e textura possui resultado semelhante, sendo 1,119 vezes melhor quando comparado com o com o método *co-clustering* com refinamento aplicado ao cubo de imagem original.

Em relação ao Índice de Rand Ajustado, referência de qualidade mais adequada para avaliação de classificação não supervisionada, tendo por base os valores apresentados no Quadro 6, nota-se que, em média, os valores encontrados para o referido índice calculado a partir do método *co-clustering* sem refinamento aplicado ao cubo de imagem integrado com camadas de morfologia matemática e textura são 1,030 vezes melhores quando comparados com o método *co-clustering* sem refinamento aplicado ao cubo de imagem original. Em complemento, tendo por base os valores apresentados no Quadro 7, o referido índice, em média, calculado a partir do método *co-clustering* com refinamento pelo método K-médias aplicado ao cubo de imagem integrado com camadas de morfologia matemática e textura possui resultado semelhante, sendo 1,035 vezes melhor quando comparado com o com o método *co-clustering* com refinamento aplicado ao cubo de imagem original.

Avaliando-se os resultados apresentados nos Quadro 6 e Quadro 7, para os índices calculados a partir do método *co-clustering* sem refinamento e calculados a partir do método *co-clustering* com refinamento, nota-se que as medidas de melhora apresentam valor calculado de variância próximo de zero, o que evidencia pequena dispersão de valores em torno do valor médio, bem como valor calculado de desvio padrão próximo de zero, que representa valores bem distribuídos em torno da média. Tais resultados evidenciam, de forma geral, pequena sensibilidade na formação dos agrupamentos ao se incluir camadas de morfologia matemática e de textura ao cubo de imagem original, nos

parâmetros definidos, em relação a heterogeneidade das amostras, em comparação ao método de classificação aplicado ao cubo de imagem original.

Destaca-se a aplicabilidade do cubo de imagem integrado com informações de contexto espacial, de forma a se explorar melhor o potencial de análise multivariada de dados da técnica de classificação *co-clustering*, quando aplicada a imagem RGB.

4.2.6 Imagens classificadas pelos três métodos

Foram processadas todas as imagens referentes a cidade de Potsdam, na Alemanha, disponibilizadas pela ISPRS, de 5cm de resolução espacial, representadas e divididas em ortofotos, conforme Figura 9.

A totalidade das ortofotos foram classificadas pelos métodos propostos e seguem apresentadas para as Figuras a seguir, sendo a ortofoto RGB original representada pela imagem A; a ortofoto classificada pelo método K-médias, aplicada ao cubo de imagem RGB original, representada pela imagem B; a ortofoto classificada pelo método *co-clustering* sem refinamento representada pela imagem C; e a ortofoto classificada pelo método *co-clustering* com refinamento pelo método K-médias representada pela imagem D.

Os índices relativos às medidas de similaridade para cada método são apresentados nos Apêndice 3, Apêndice 4 e Apêndice 5.

Em relação ao resultado apresentado pela Figura 30 - ortofoto 2_10, tem-se que o Índice de Rand Ajustado calculado para o método K-médias aplicado ao cubo de imagem original é de 0,2851, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2850, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5994, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

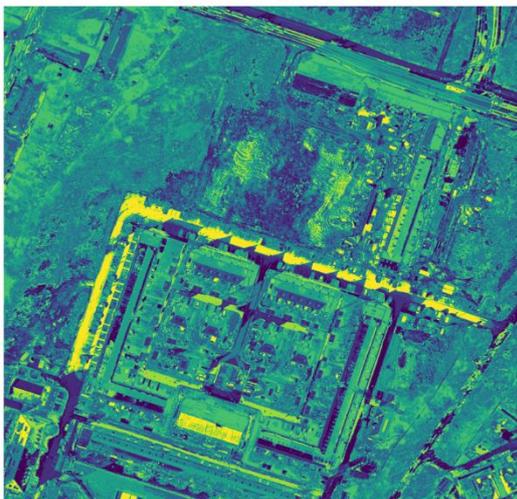
A



B



C



D

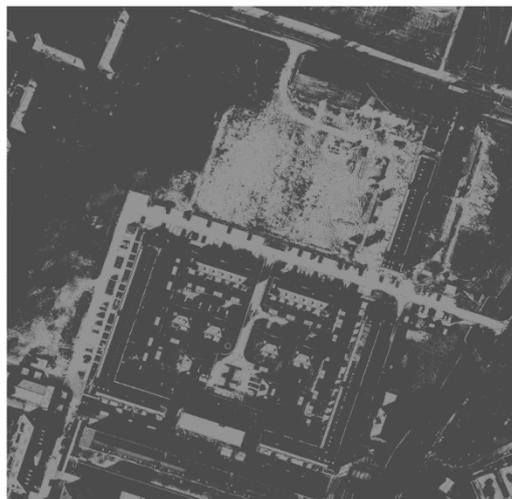


Figura 30 - ortofoto 2_10

Em relação ao resultado apresentado pela Figura 31 - ortofoto 2_11, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2751, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2905, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento

K-médias é de 0,6084, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

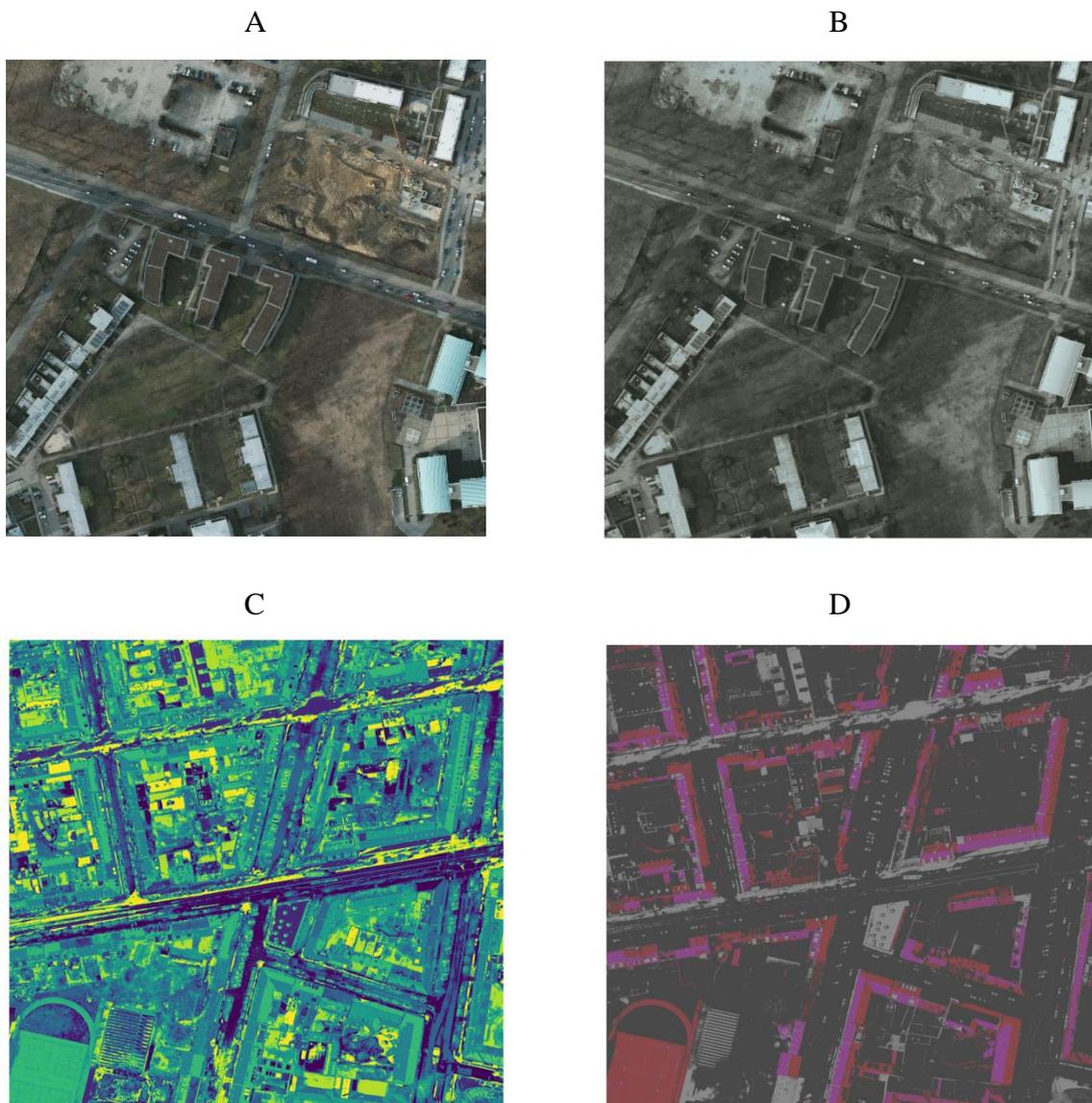


Figura 31 - ortofoto 2_11

Em relação ao resultado apresentado pela Figura 32 - ortofoto 2_12, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2799, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2836, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6548, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

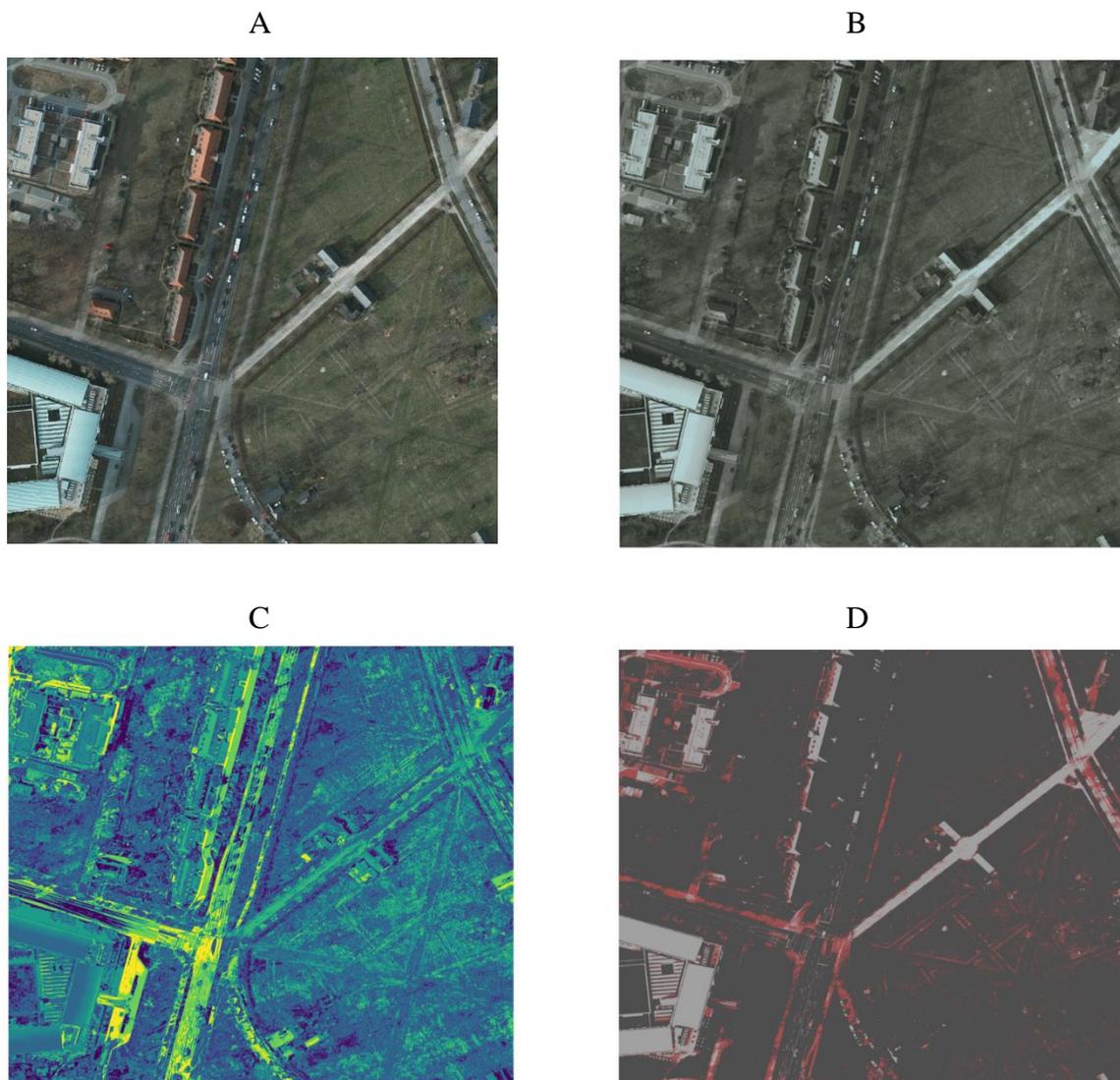


Figura 32 - ortofoto 2_12

Em relação ao resultado apresentado pela Figura 33 - ortofoto 2_13 , tem se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2553, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2684, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5281, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

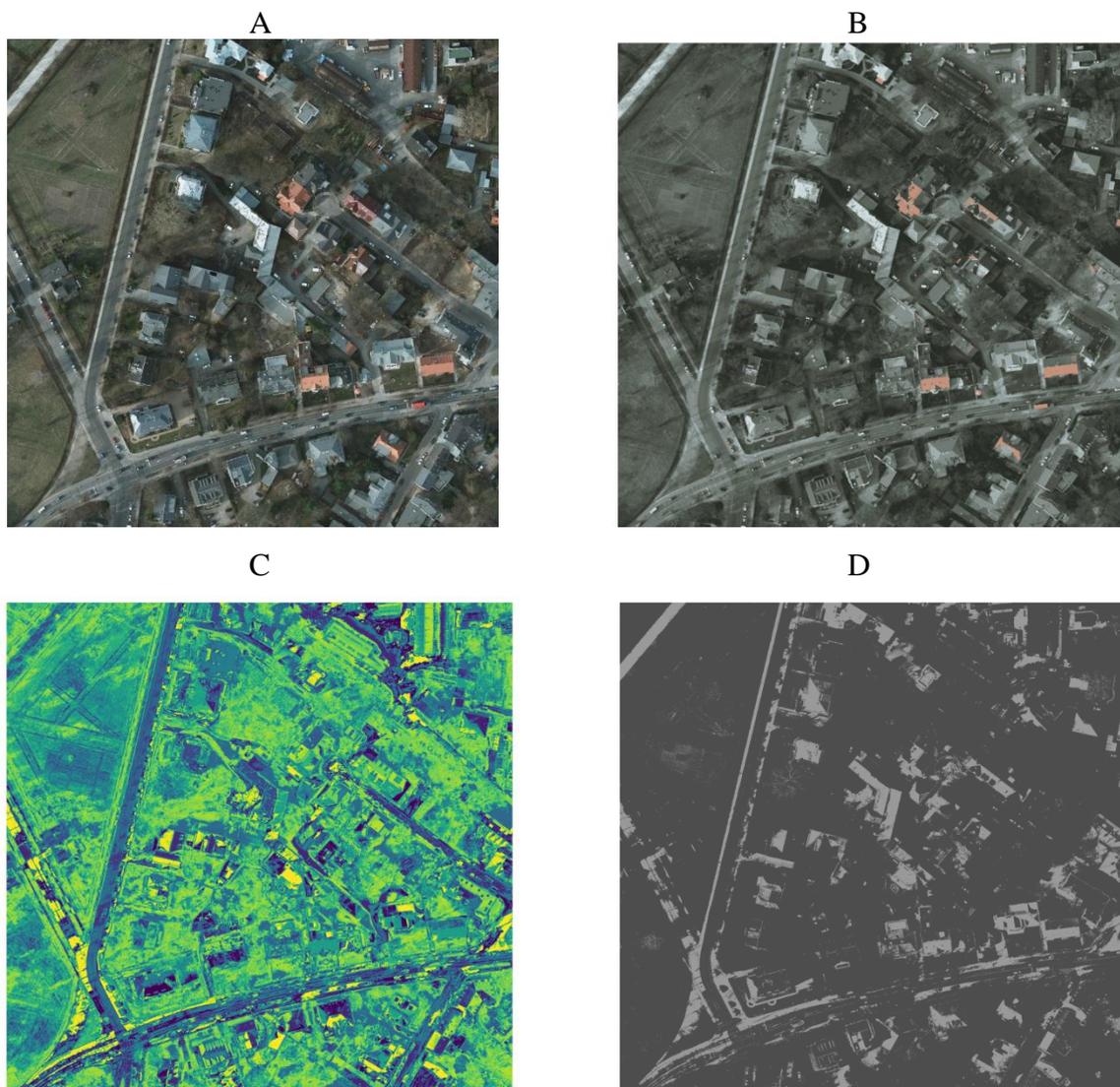


Figura 33 - ortofoto 2_13

Em relação ao resultado apresentado pela Figura 34 - ortofoto 2_14, tem se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2619, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2755,

imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,0670, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

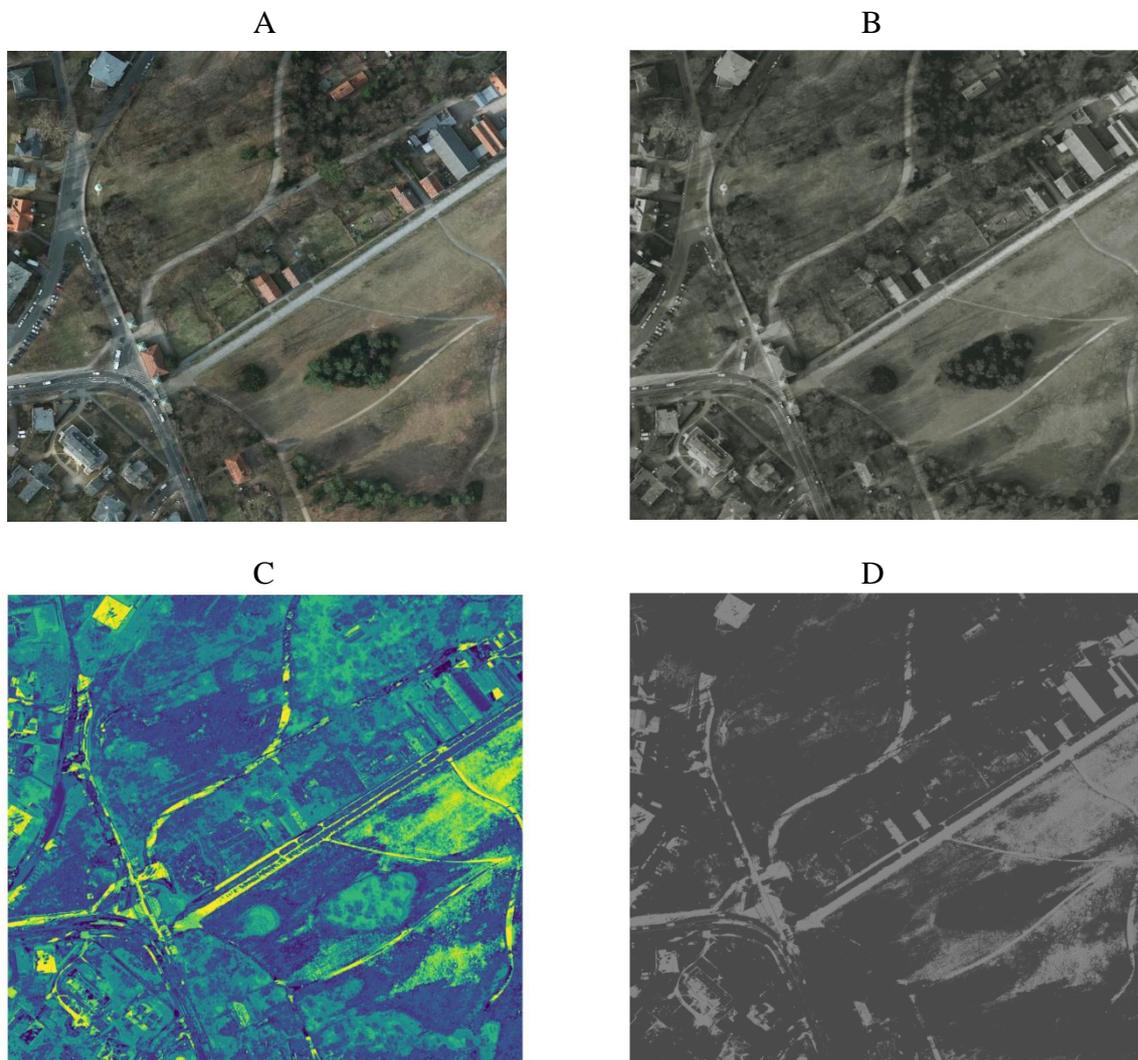


Figura 34 - ortofoto 2_14

Em relação ao resultado apresentado pela Figura 35 - ortofoto 3_10, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3022, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3126, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5041, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

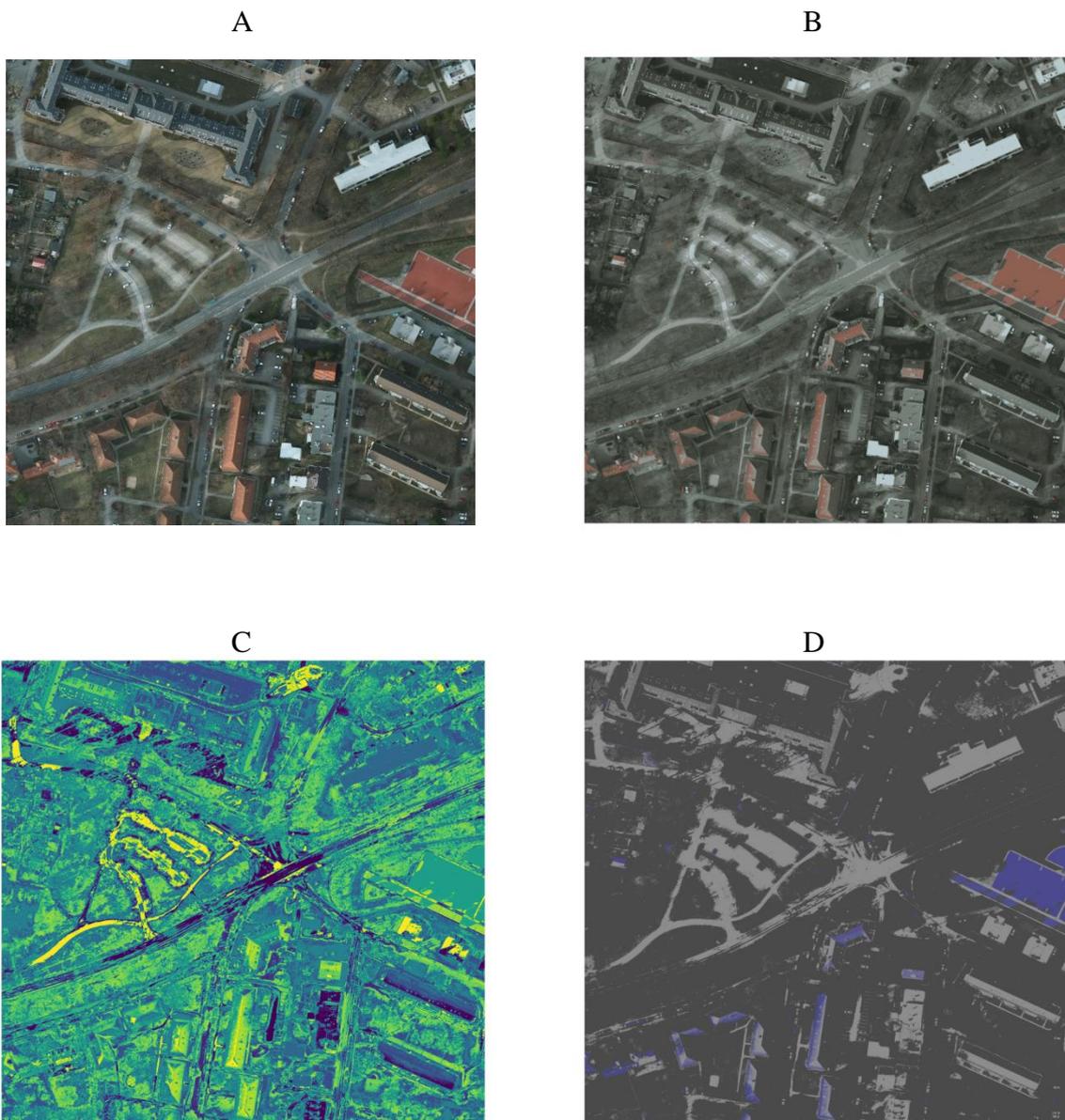


Figura 35 - ortofoto 3_10

Em relação ao resultado apresentado pela Figura 36 - ortofoto 3_11, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3204, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3179, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5550, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

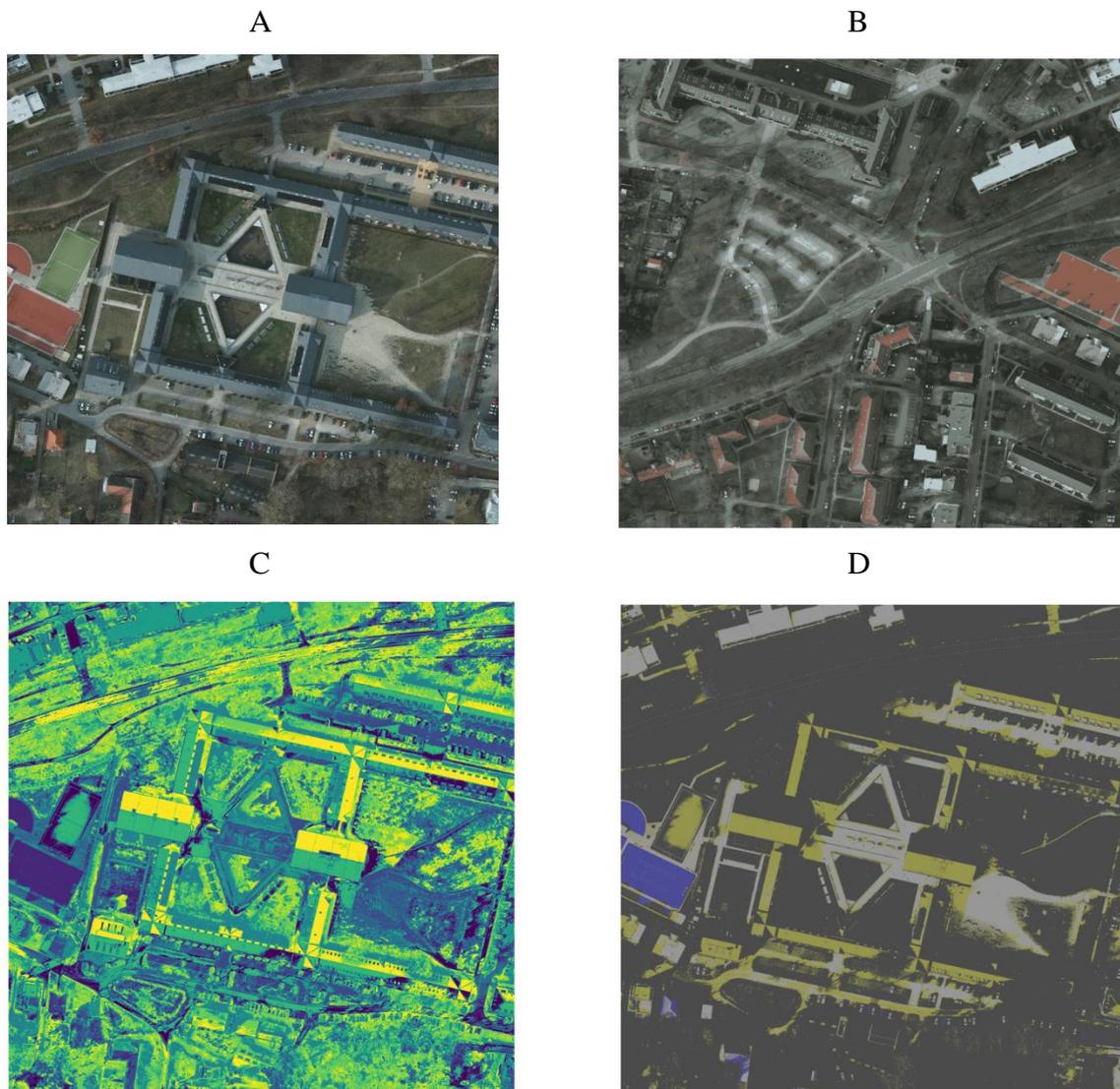


Figura 36 - ortofoto 3_11

Em relação ao resultado apresentado pela Figura 37 - ortofoto 3_12, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2857, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2867, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,4980, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

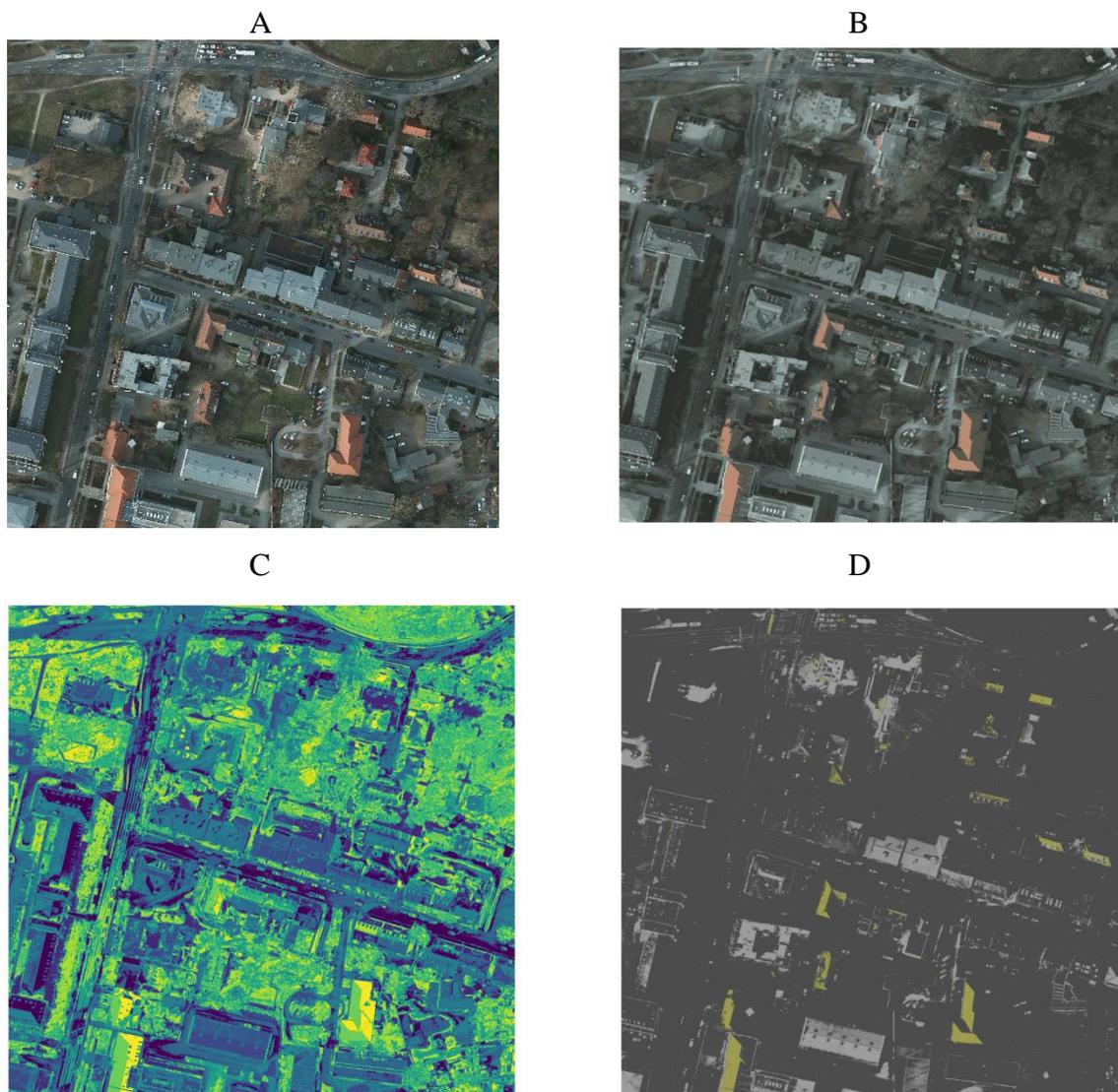


Figura 37 - ortofoto 3_12

Em relação ao resultado apresentado pela

Figura 38 - ortofoto 3_13, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2634, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2765, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,4697, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

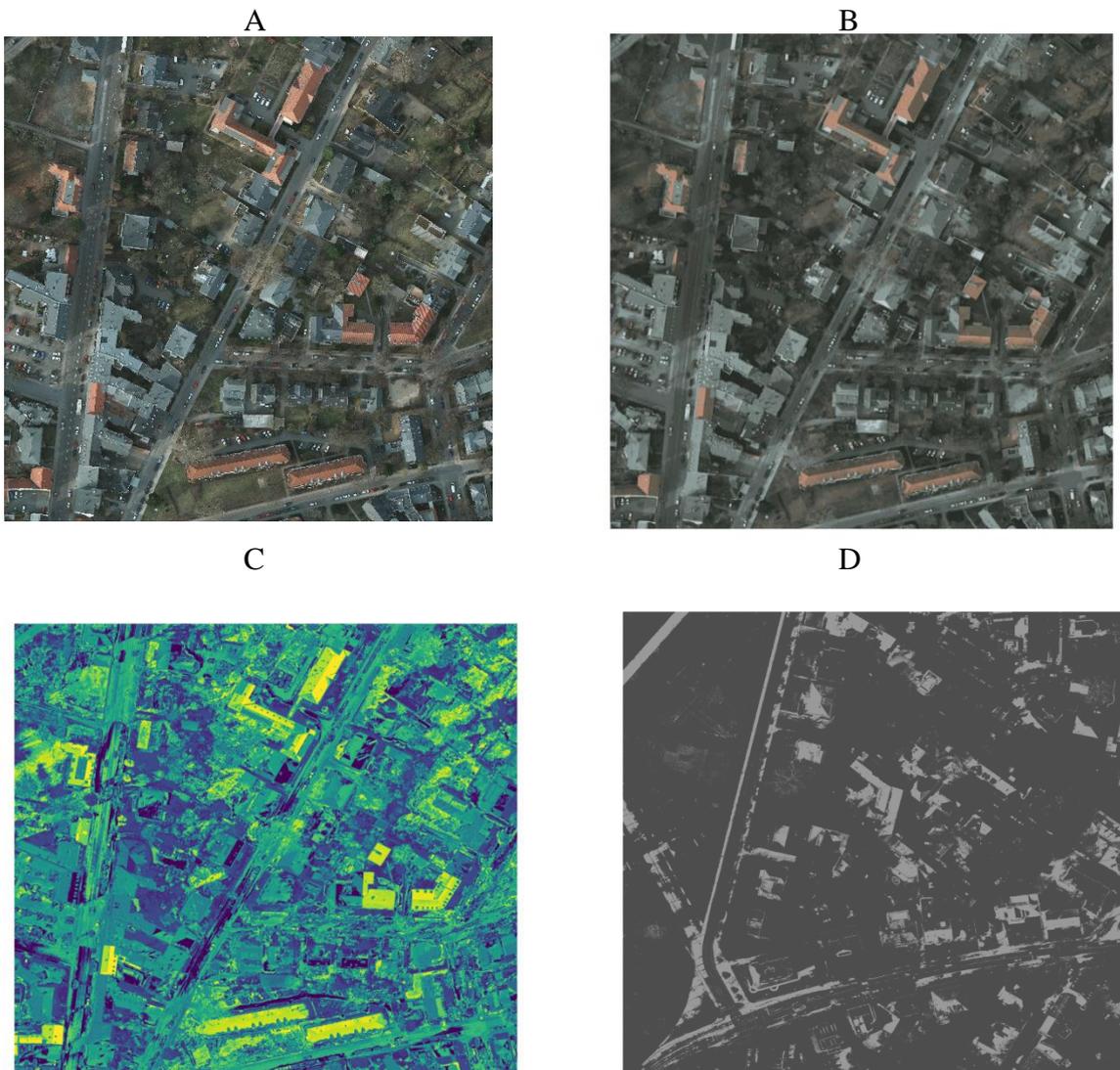


Figura 38 - ortofoto 3_13

Em relação ao resultado apresentado pela Figura 39 - ortofoto 3_14, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2714, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2772, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5344, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

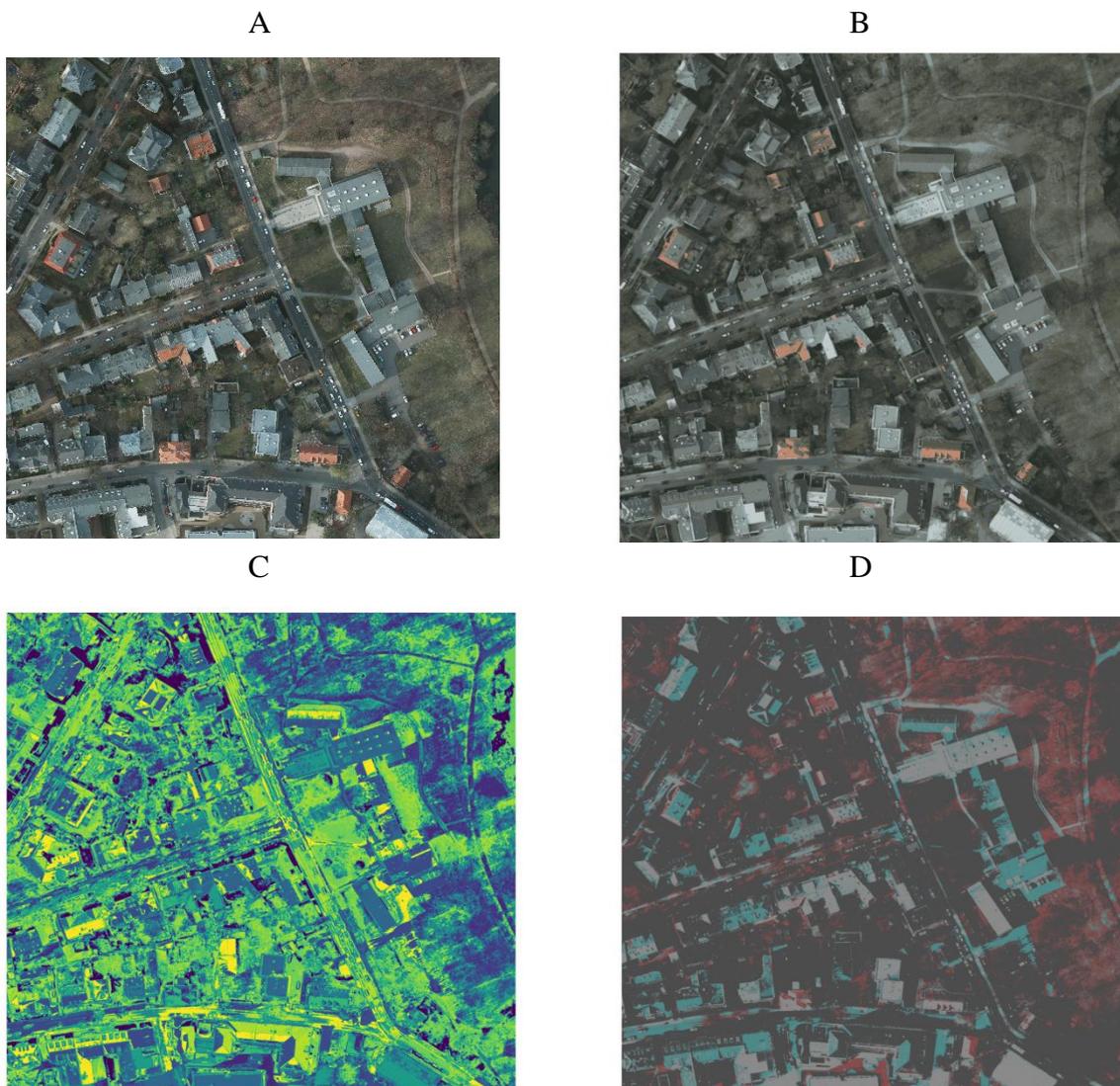


Figura 39 - ortofoto 3_14

Em relação ao resultado apresentado pela Figura 40 - ortofoto 4_10, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2700, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2733, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,4999, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

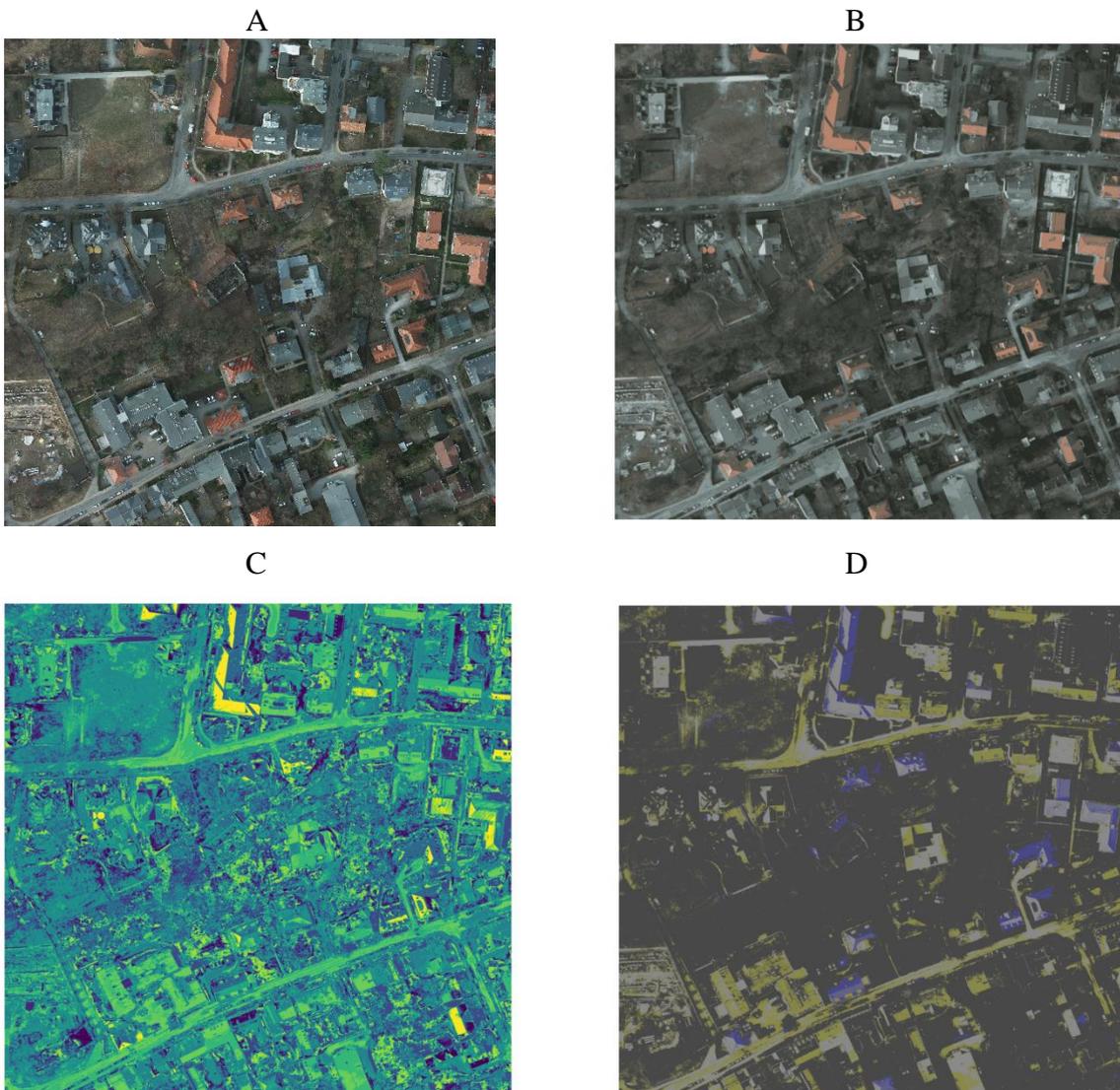


Figura 40 - ortofoto 4_10

Em relação ao resultado apresentado pela Figura 41 - ortofoto 4_11, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2797, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2891, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5119, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

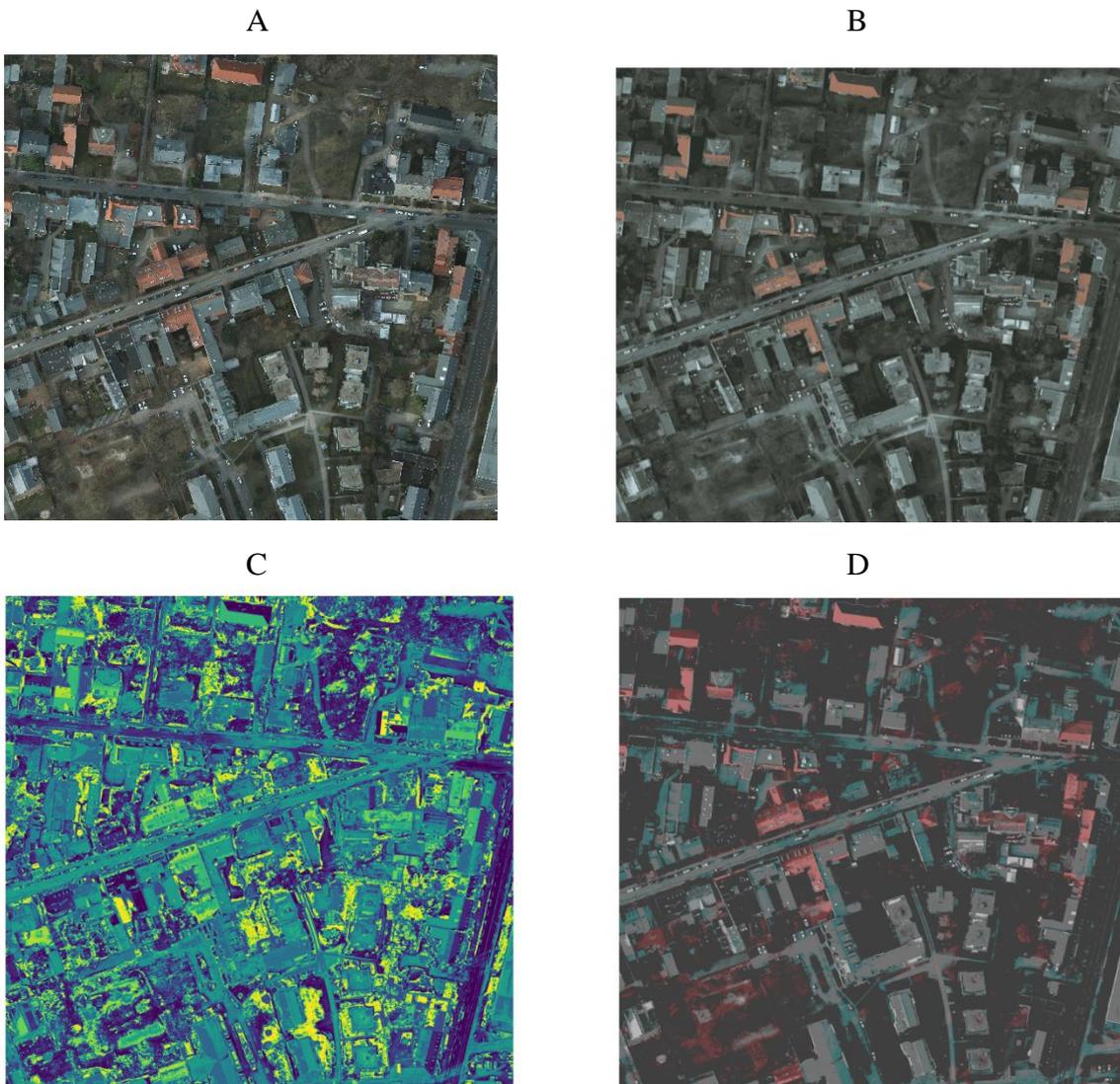


Figura 41 - ortofoto 4_11

Em relação ao resultado apresentado pela Figura 42 - ortofoto 4_12, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,0969, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,0830, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,0325, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

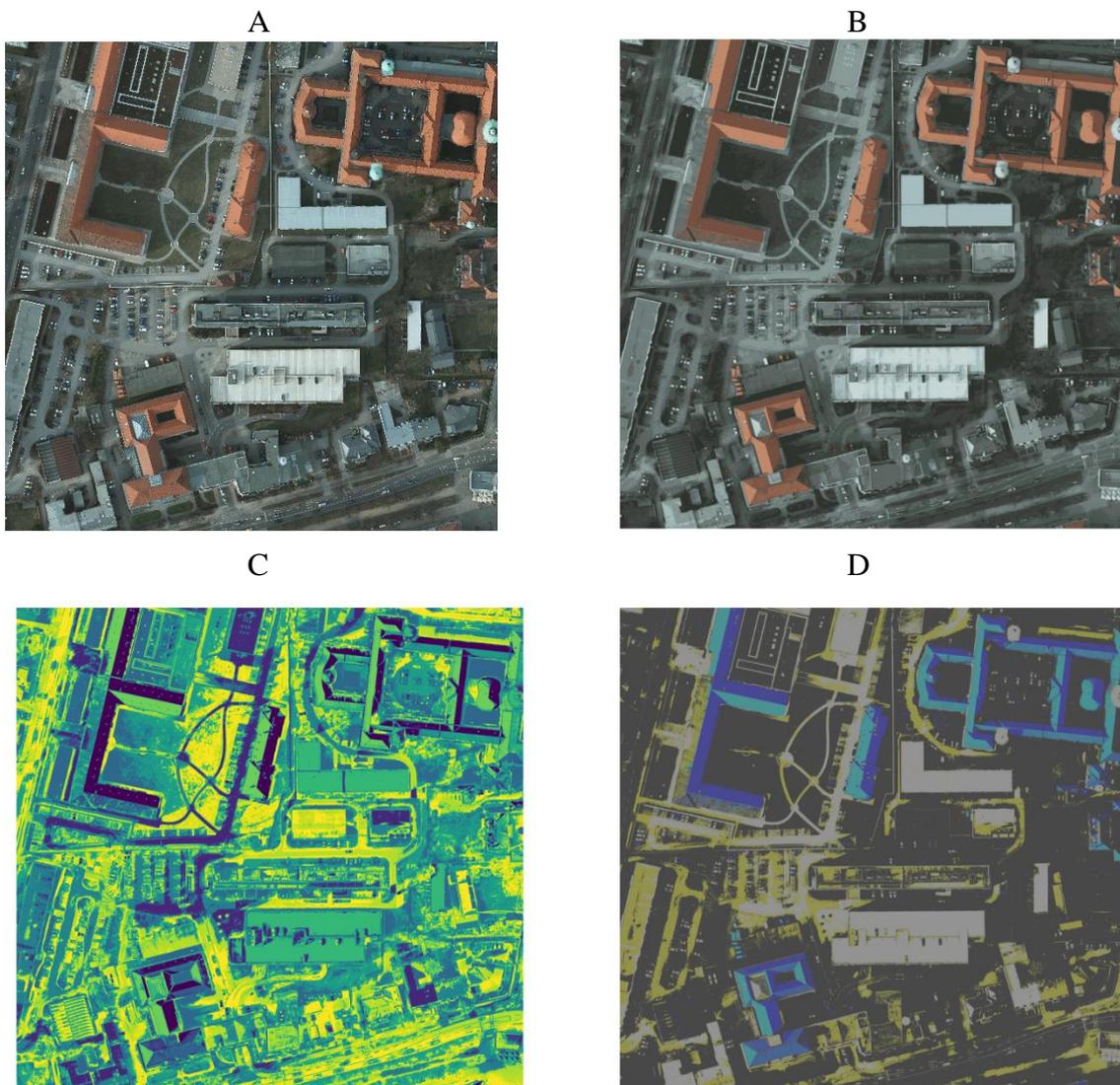


Figura 42 - ortofoto 4_12

Em relação ao resultado apresentado pela Figura 43 - ortofoto 4_13, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2888, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2899, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5067, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

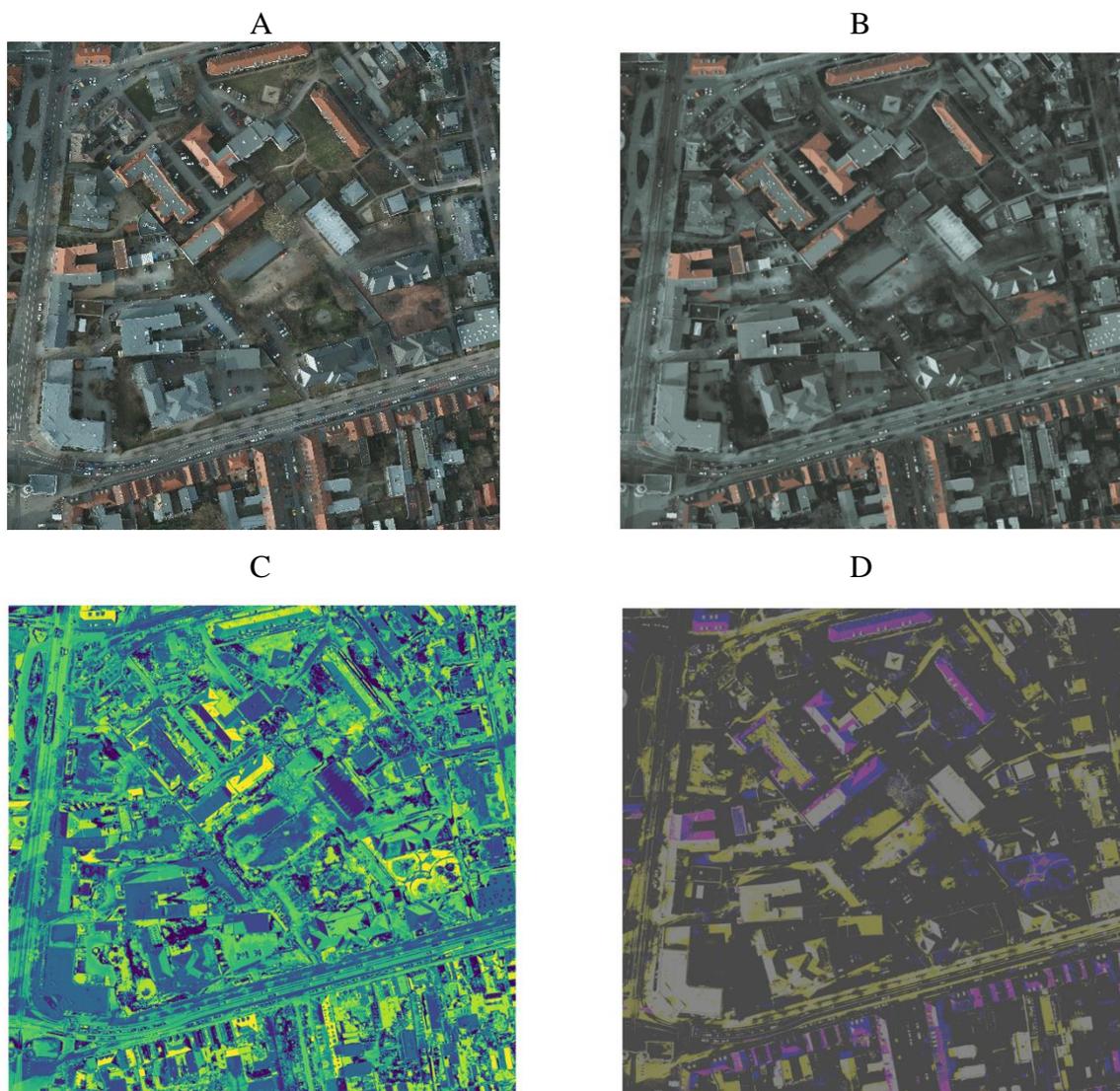


Figura 43 - ortofoto 4_13

Em relação ao resultado apresentado pela Figura 44 - ortofoto 4_14, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2891, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2910, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5245, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

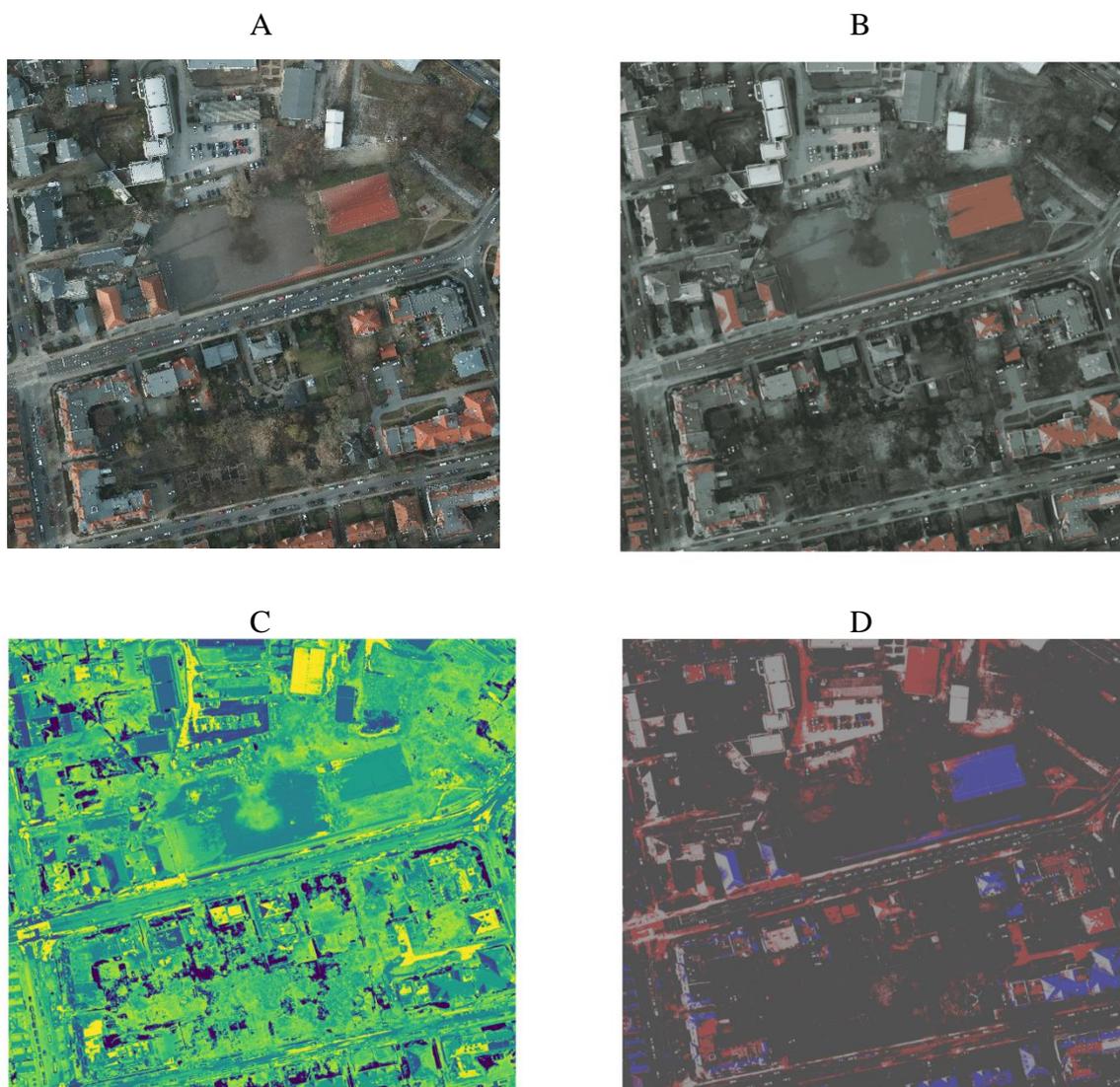


Figura 44 - ortofoto 4_14

Em relação ao resultado apresentado pela Figura 45 - ortofoto 4_15, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2768, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2988, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5514, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

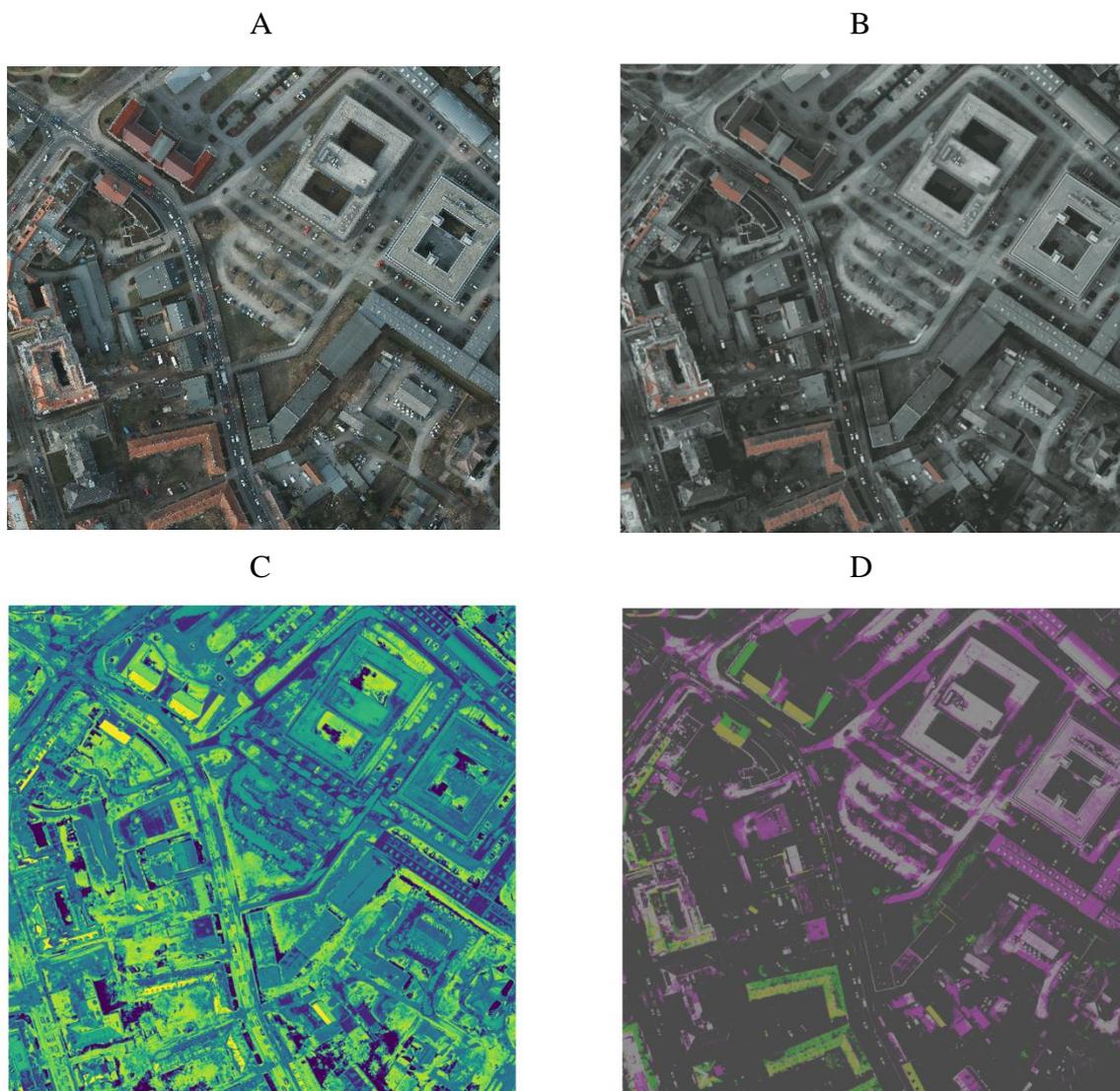


Figura 45 - ortofoto 4_15

Em relação ao resultado apresentado pela Figura 46 - ortofoto 5_10, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2944, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2990, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5543, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

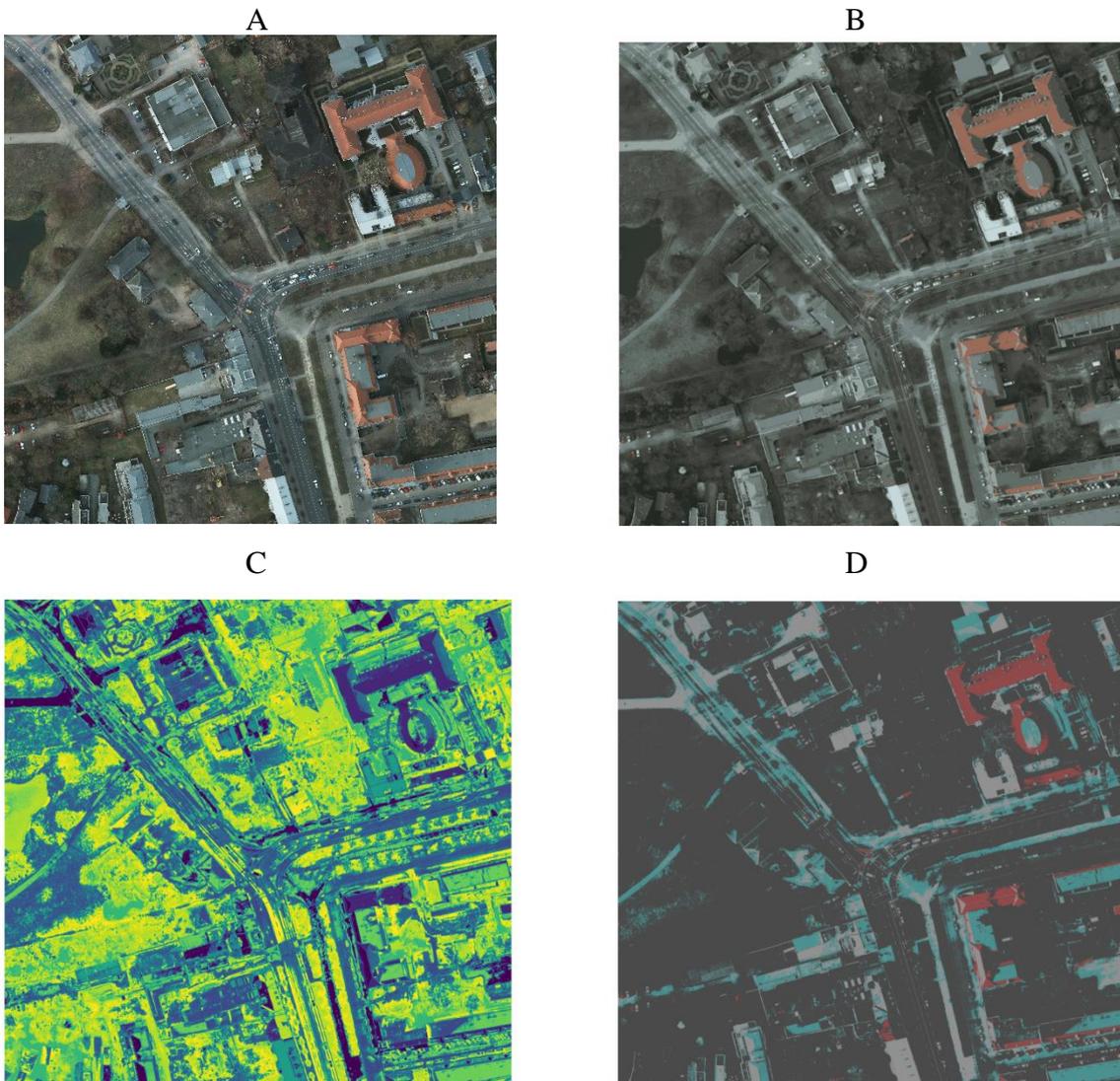


Figura 46 - ortofoto 5_10

Em relação ao resultado apresentado pela Figura 47 - ortofoto 5_11, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2727, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2793, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5458, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

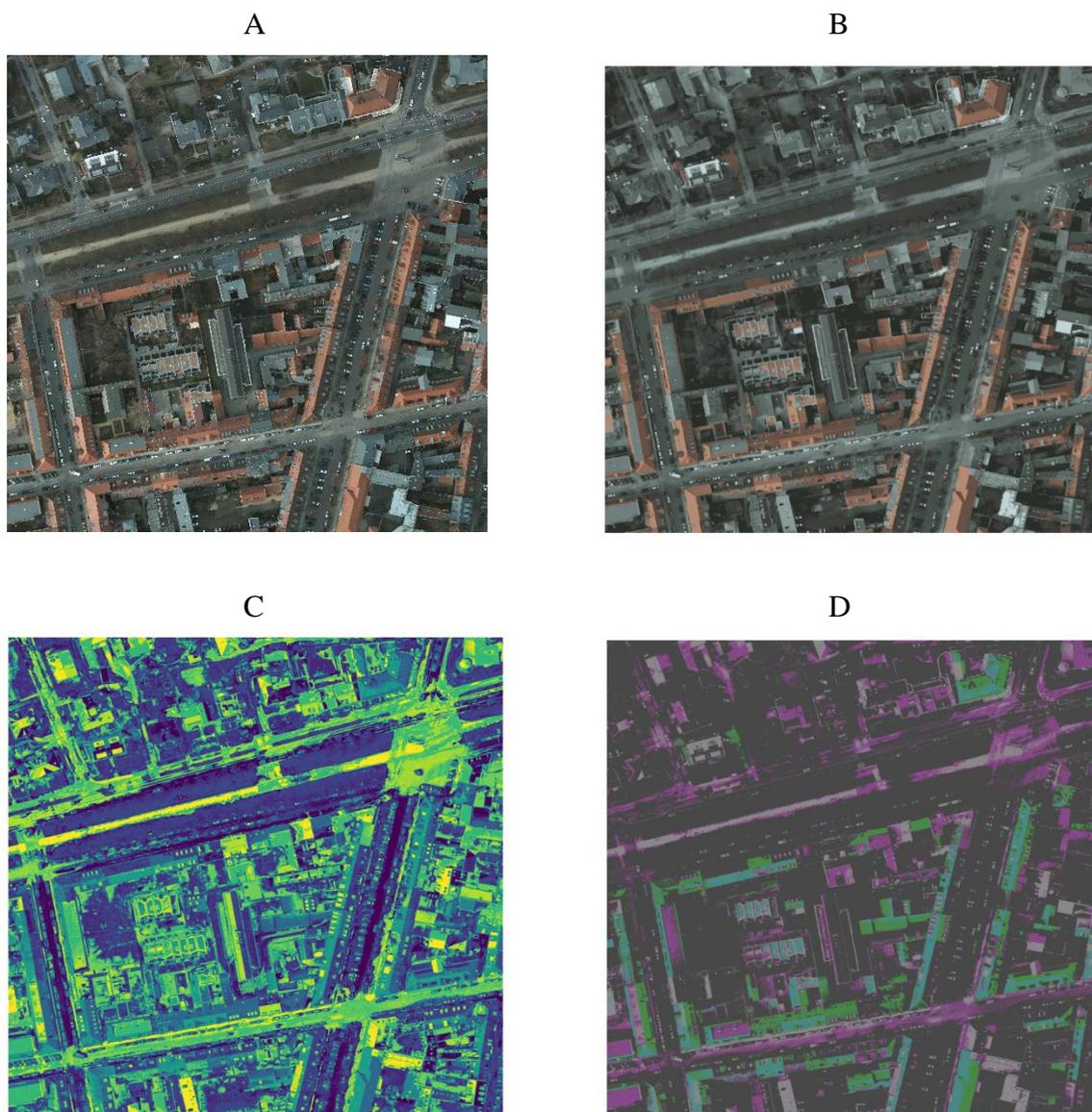


Figura 47 - ortofoto 5_11

Em relação ao resultado apresentado pela Figura 48 - ortofoto 5_12, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2731, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2792, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5989, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

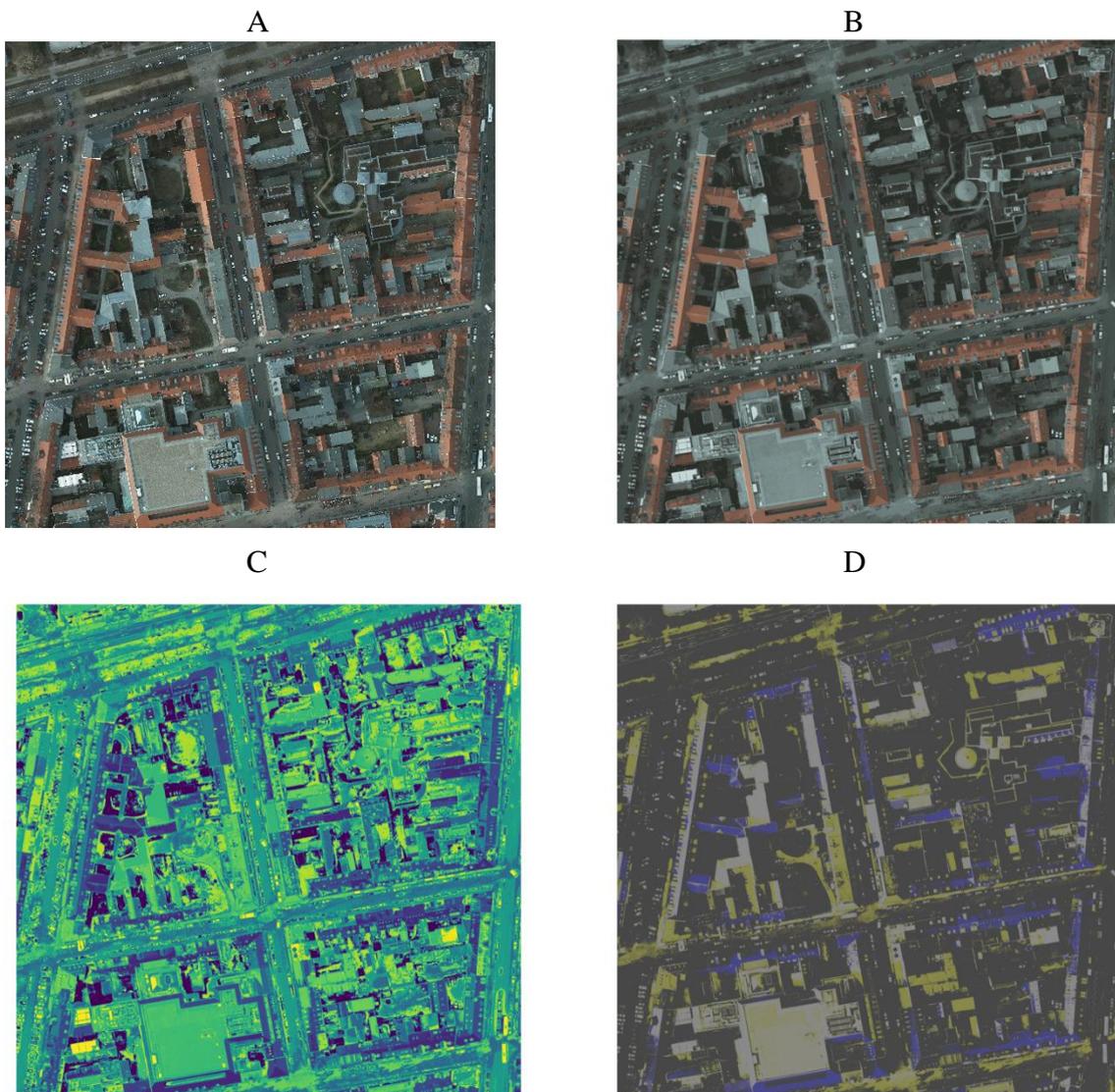


Figura 48 - ortofoto 5_12

Em relação ao resultado apresentado pela Figura 49 - ortofoto 5_13, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2987, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2857, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5682, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

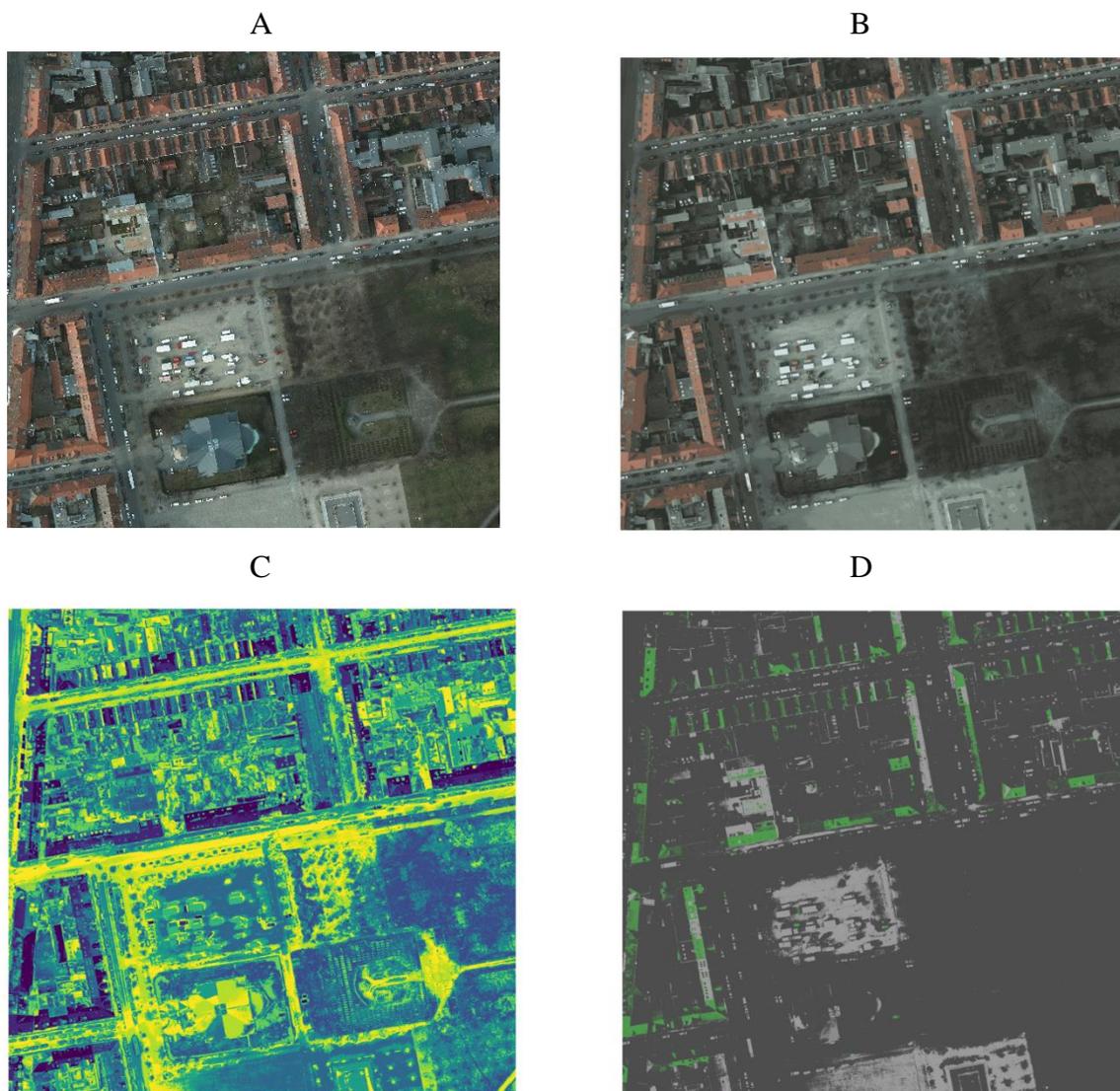


Figura 49 - ortofoto 5_13

Em relação ao resultado apresentado pela

Figura 50 - ortofoto 5_14, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2967, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3160, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5673, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

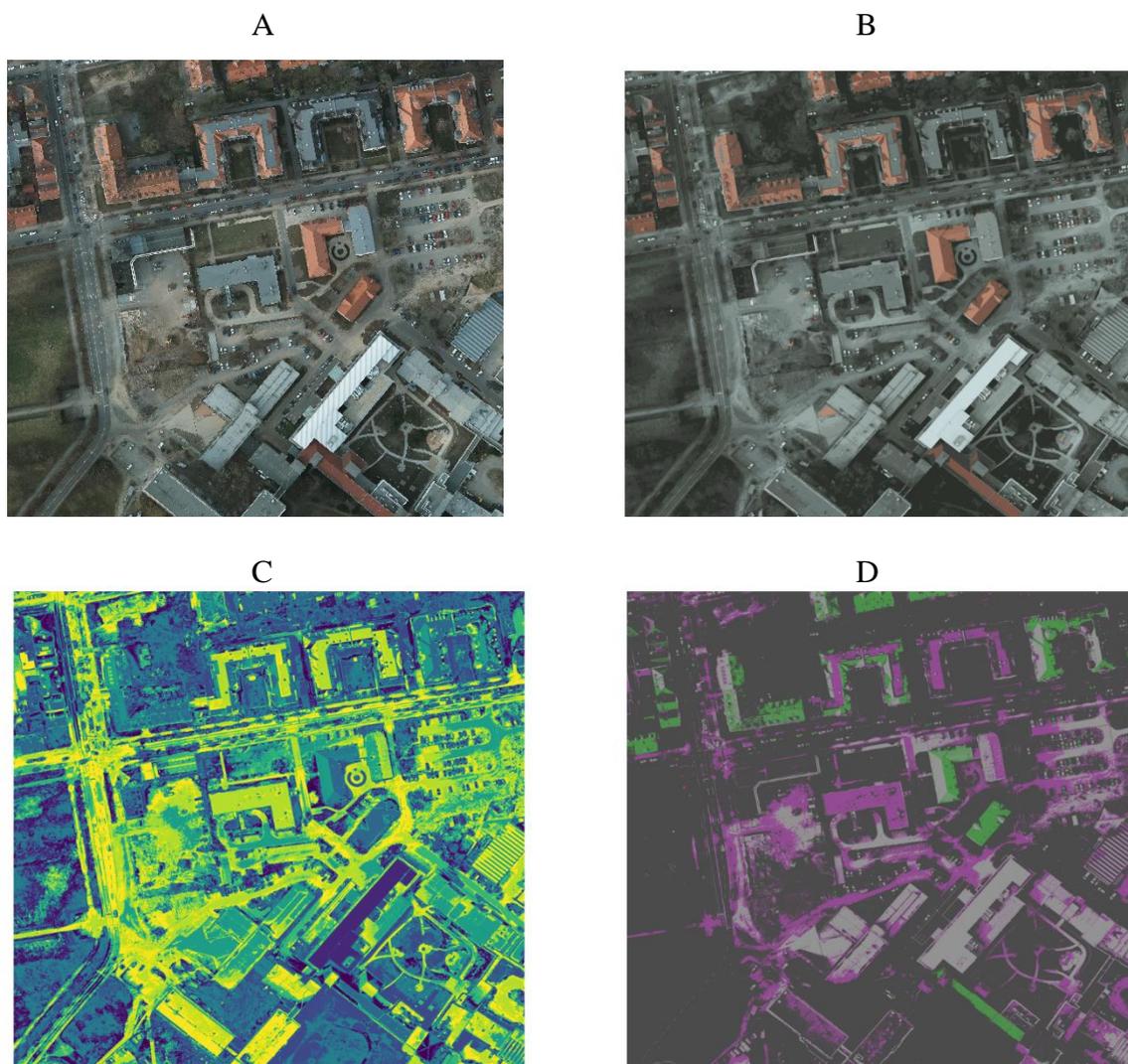


Figura 50 - ortofoto 5_14

Em relação ao resultado apresentado pela Figura 51 - ortofoto 5_15, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2893, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3031, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5584, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

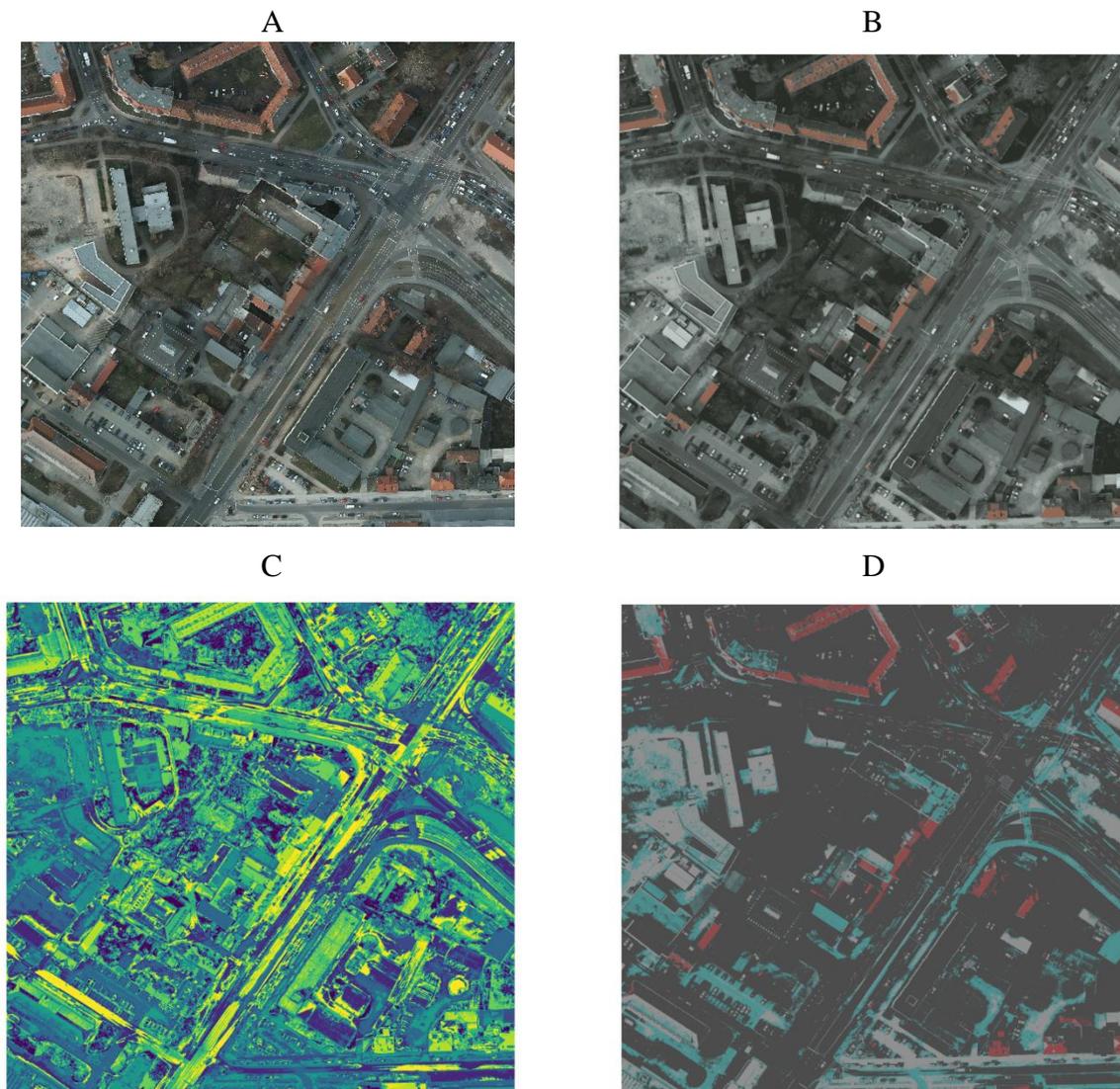


Figura 51 - ortofoto 5_15

Em relação ao resultado apresentado pela Figura 52 - ortofoto 6_7, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2639, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2660, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6852, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

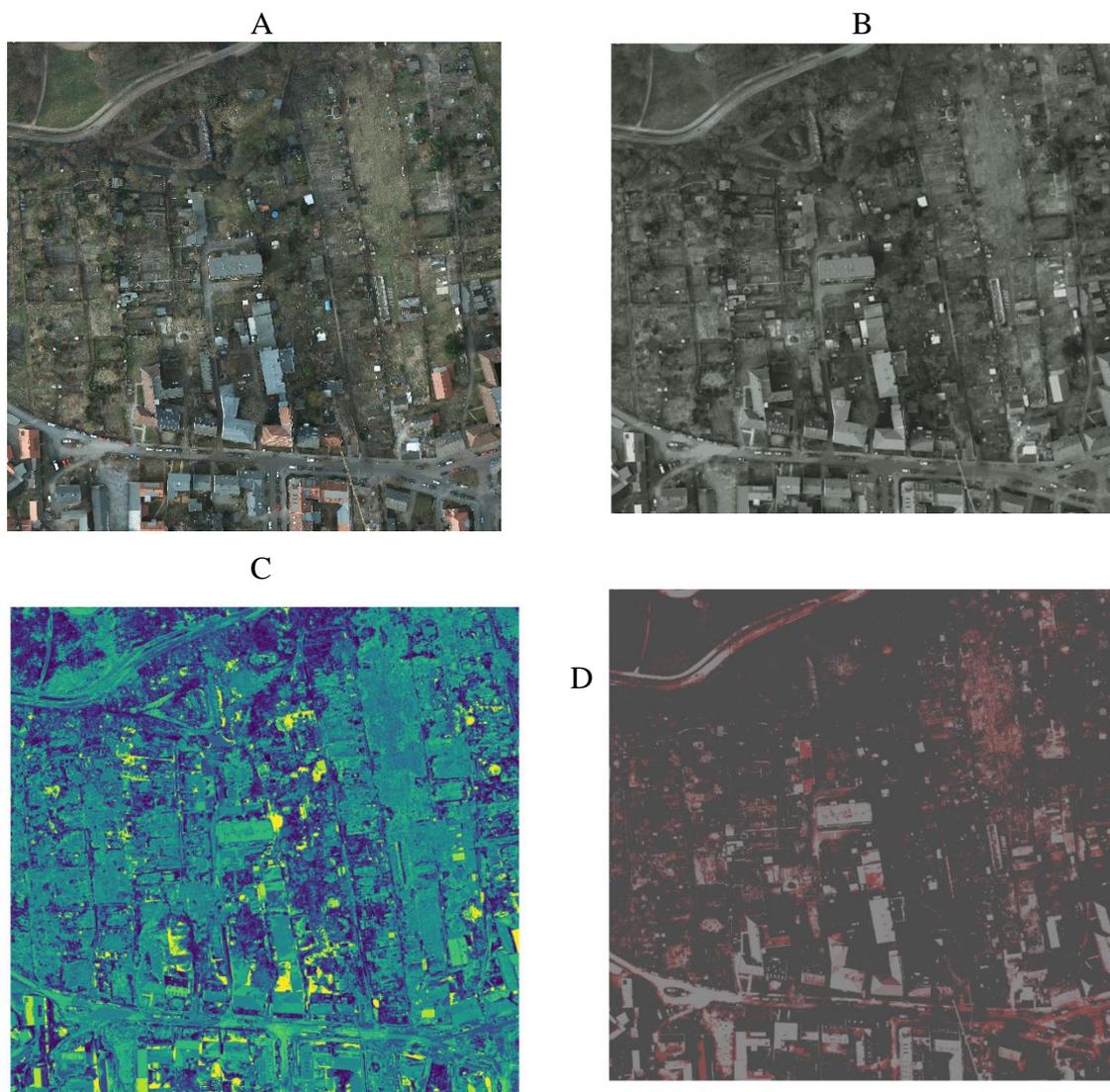


Figura 52 - ortofoto 6_7

Em relação ao resultado apresentado pela Figura 53 - ortofoto 6_8, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2772, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3095, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5431, sendo este o melhor resultado.

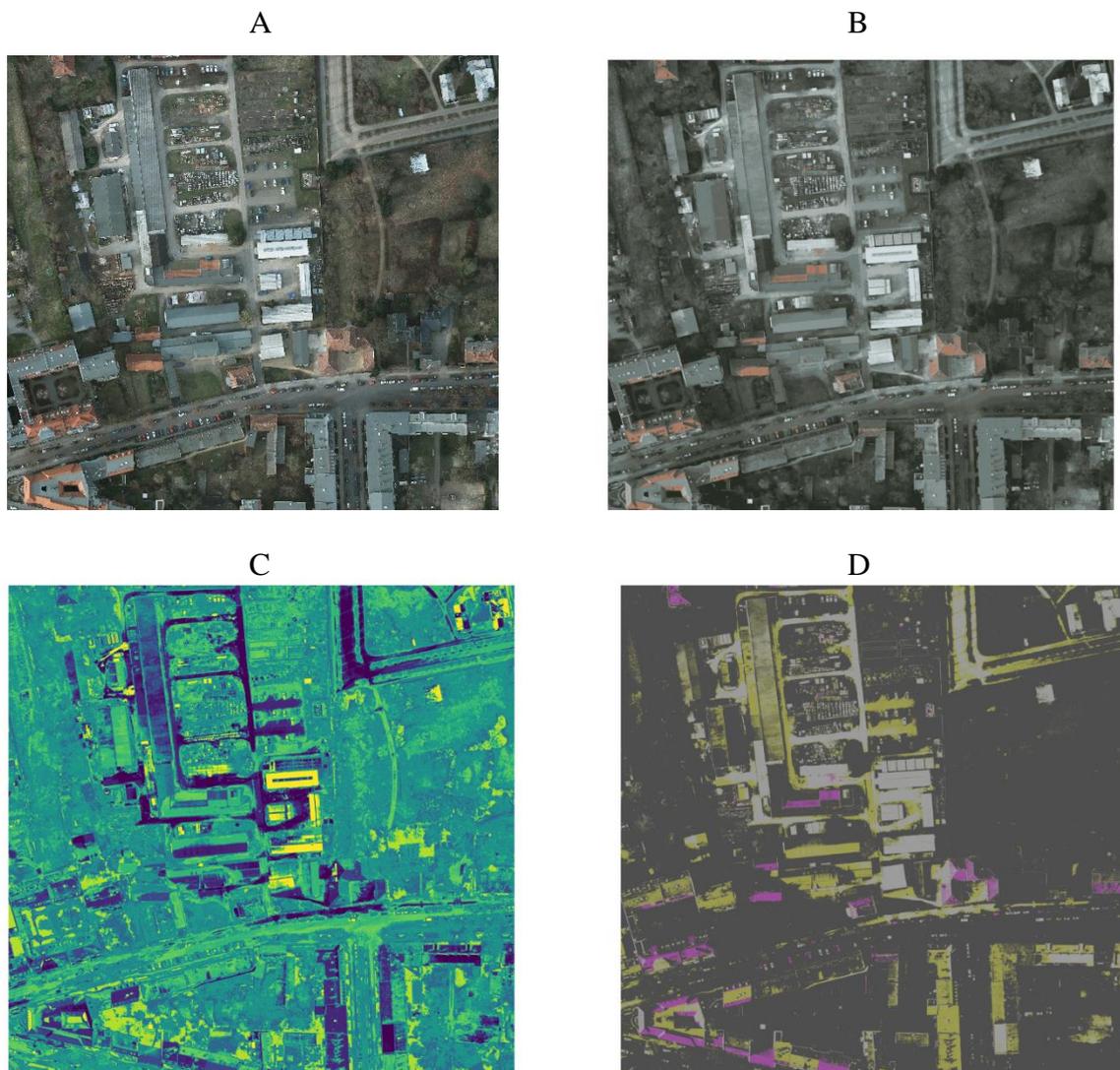


Figura 53 - ortofoto 6_8

Em relação ao resultado apresentado pela Figura 54 - ortofoto 6_9, tem se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2688, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2919, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5362, sendo este o melhor resultado.

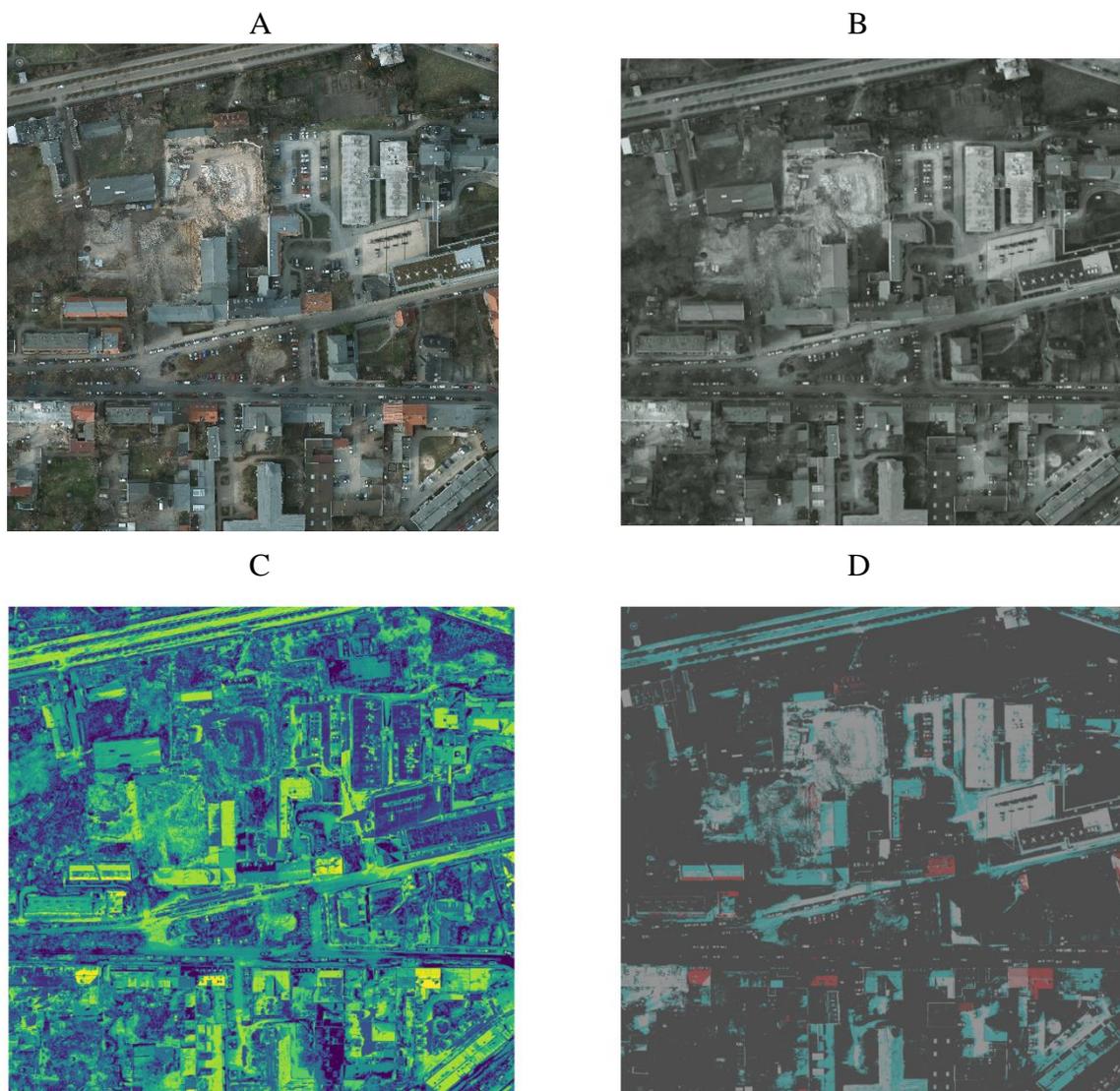


Figura 54 - ortofoto 6_9

Em relação ao resultado apresentado pela Figura 55 - ortofoto 6_10, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2990, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2971, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6118, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

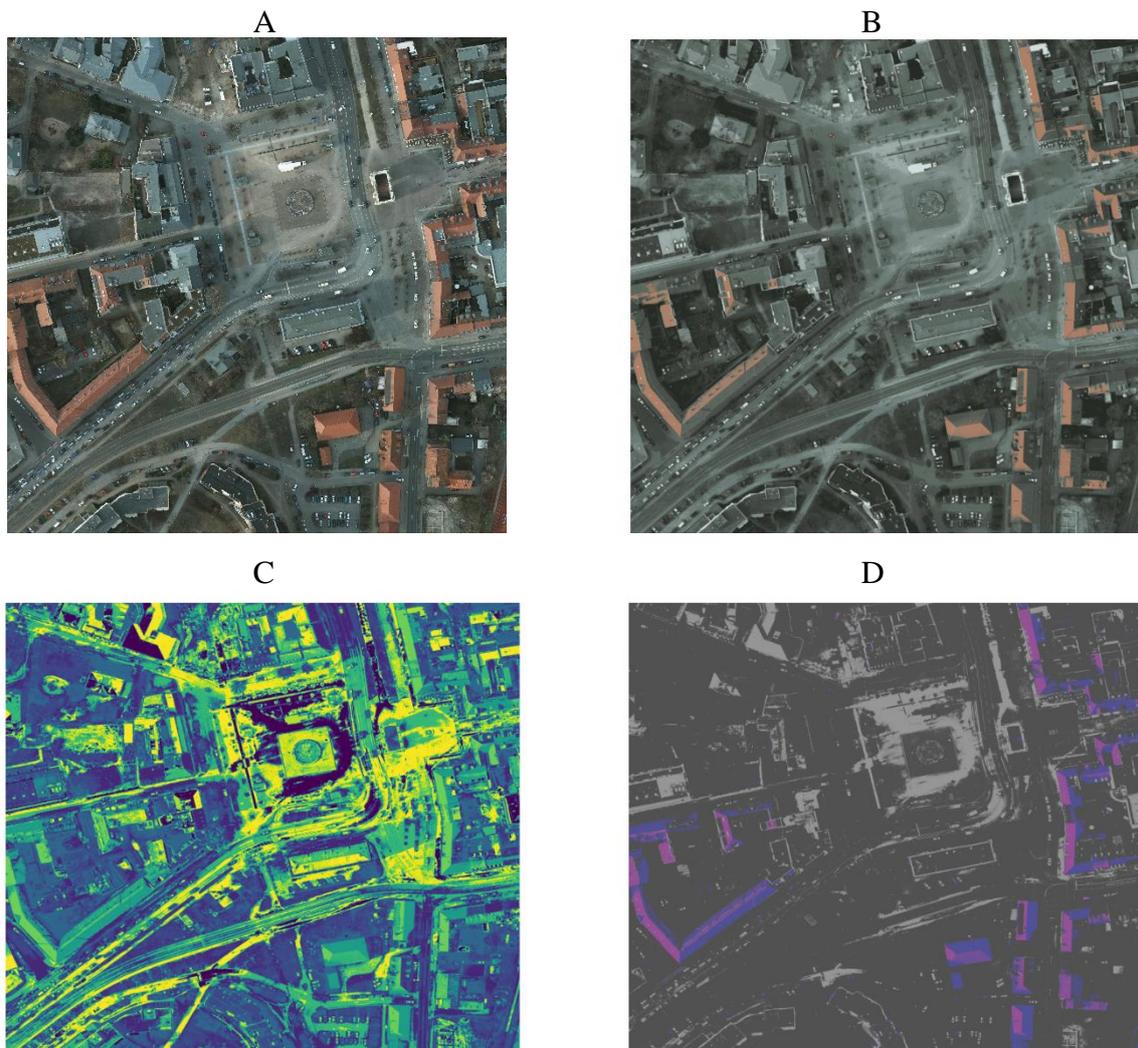


Figura 55 - ortofoto 6_10

Em relação ao resultado apresentado pela Figura 56 - ortofoto 6_11, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3001, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2820, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5999, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

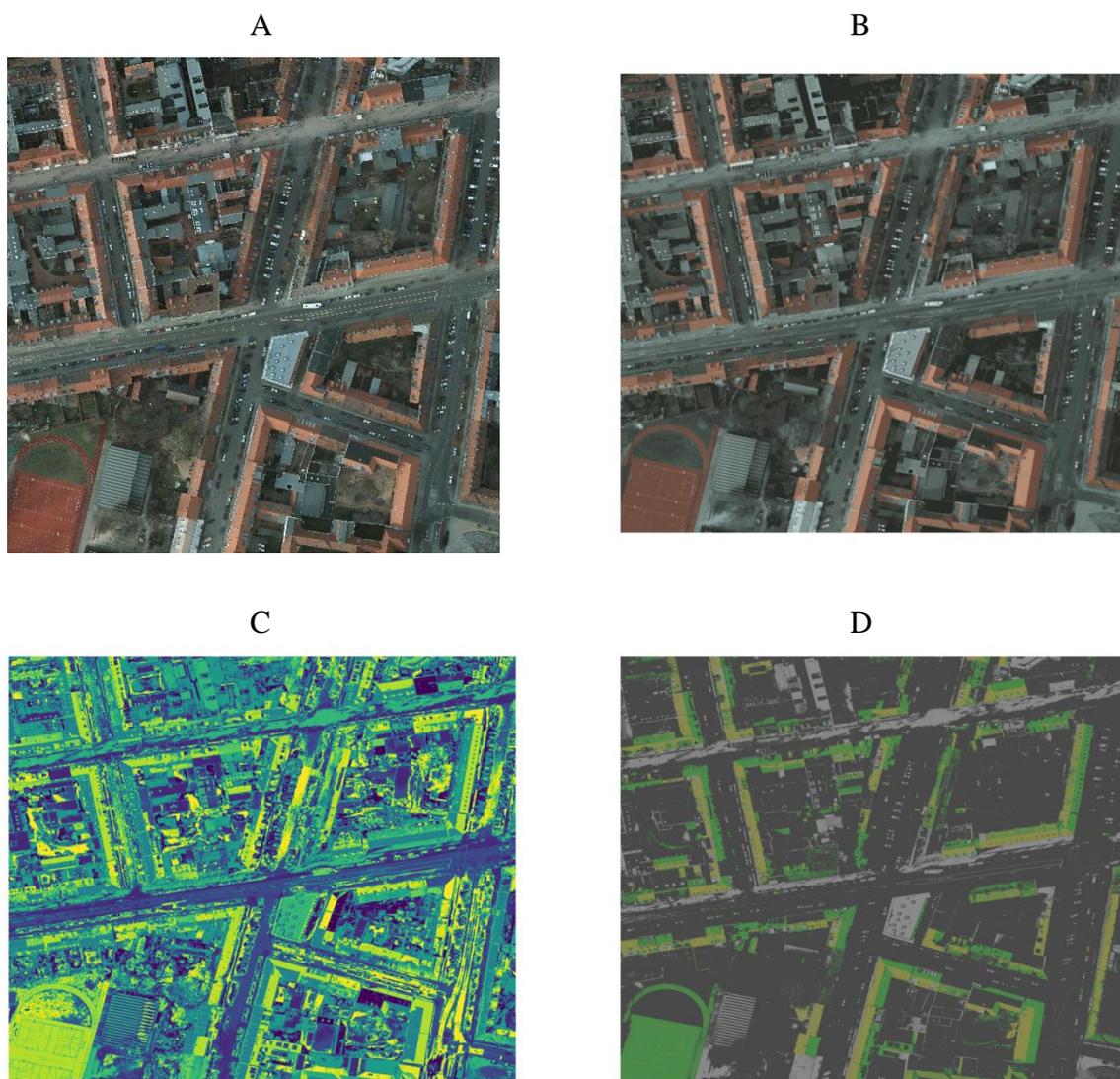


Figura 56 - ortofoto 6_11

Em relação ao resultado apresentado pela Figura 57 - ortofoto 6_12, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3019, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3000, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6118, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

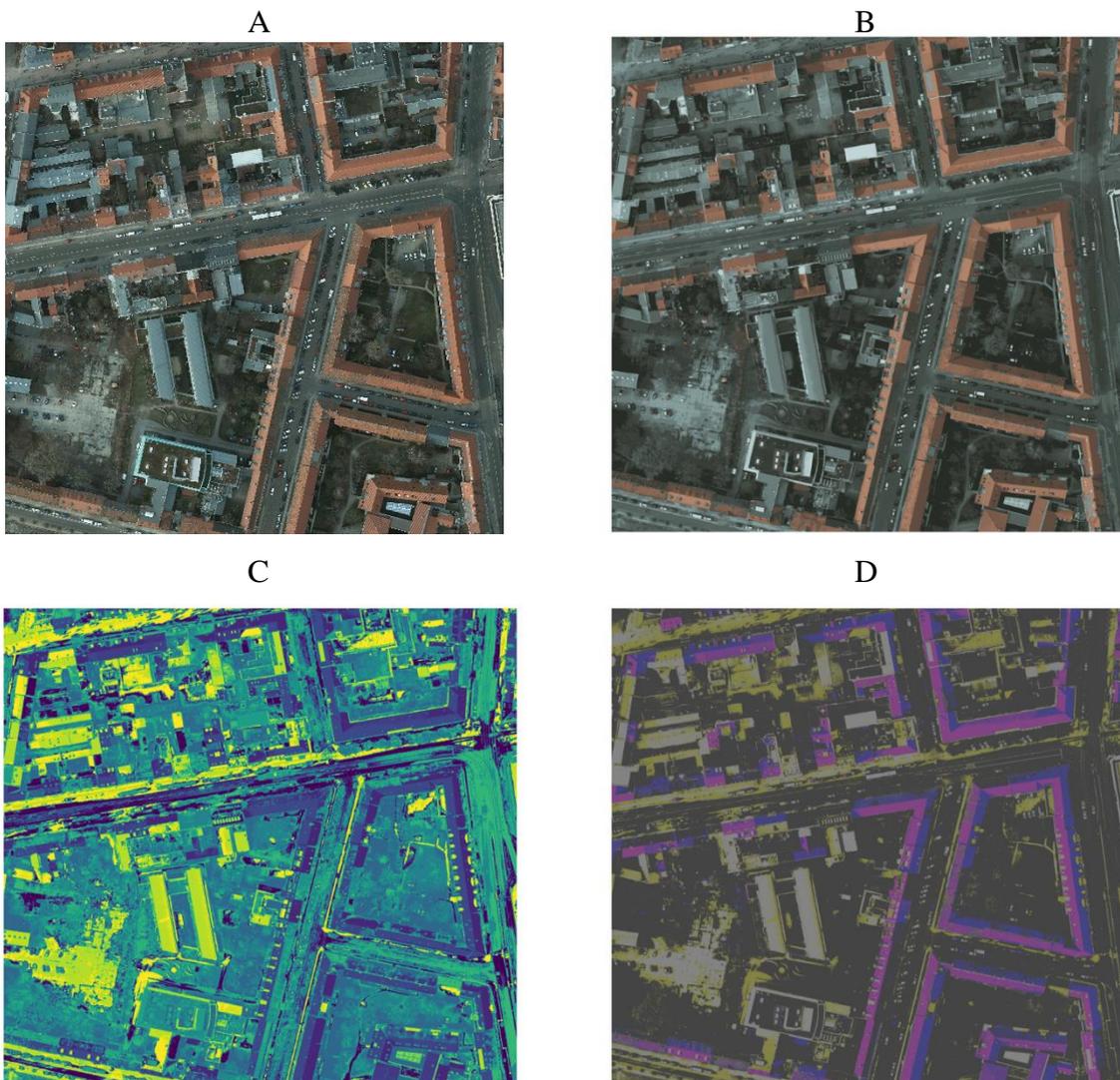


Figura 57 - ortofoto 6_12

Em relação ao resultado apresentado pela

Figura 58 - ortofoto 6_13, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3158, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3354, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6247, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

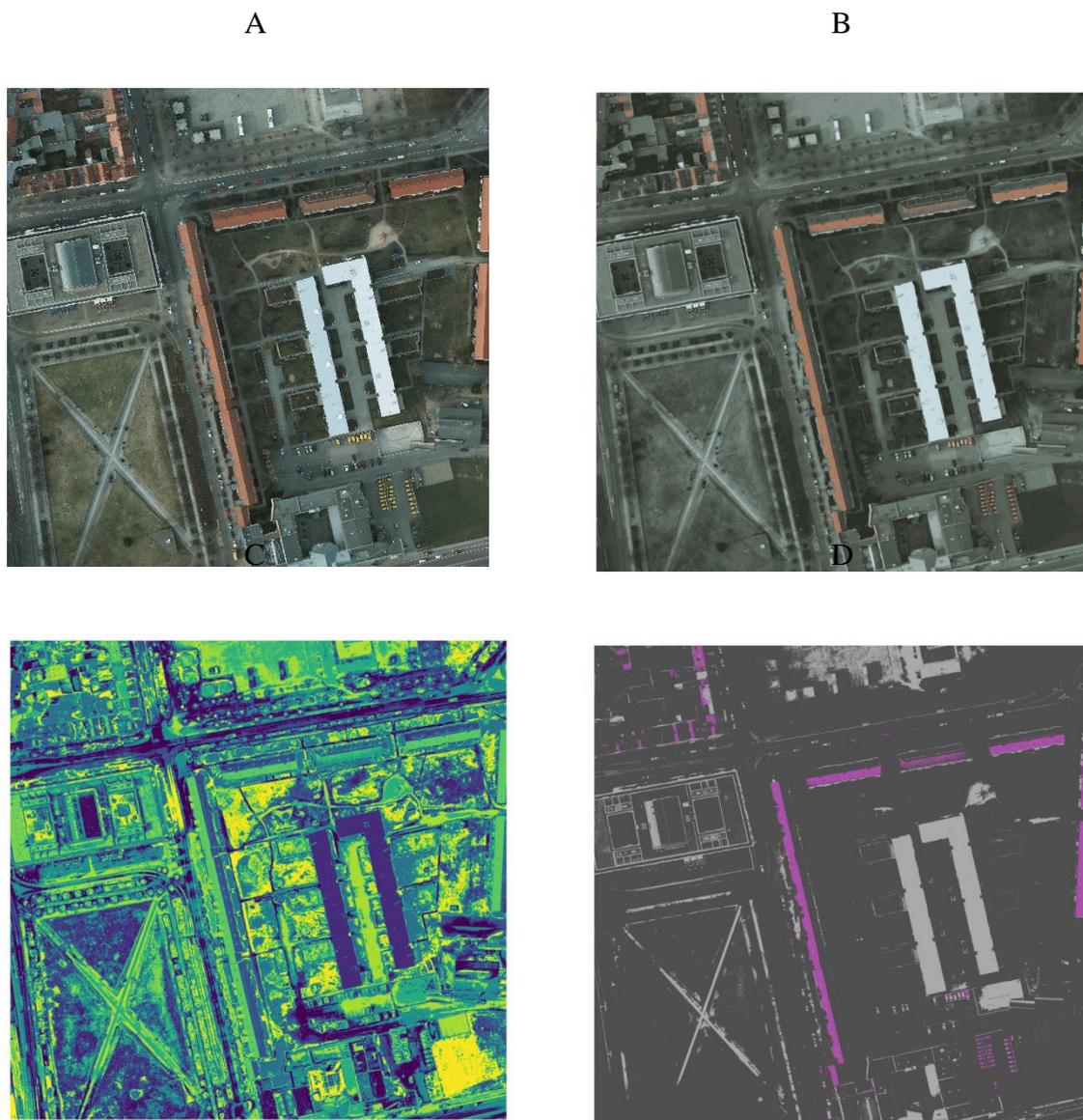


Figura 58 - ortofoto 6_13

Em relação ao resultado apresentado pela Figura 59 - ortofoto 6_14, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3299, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3378, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6180, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

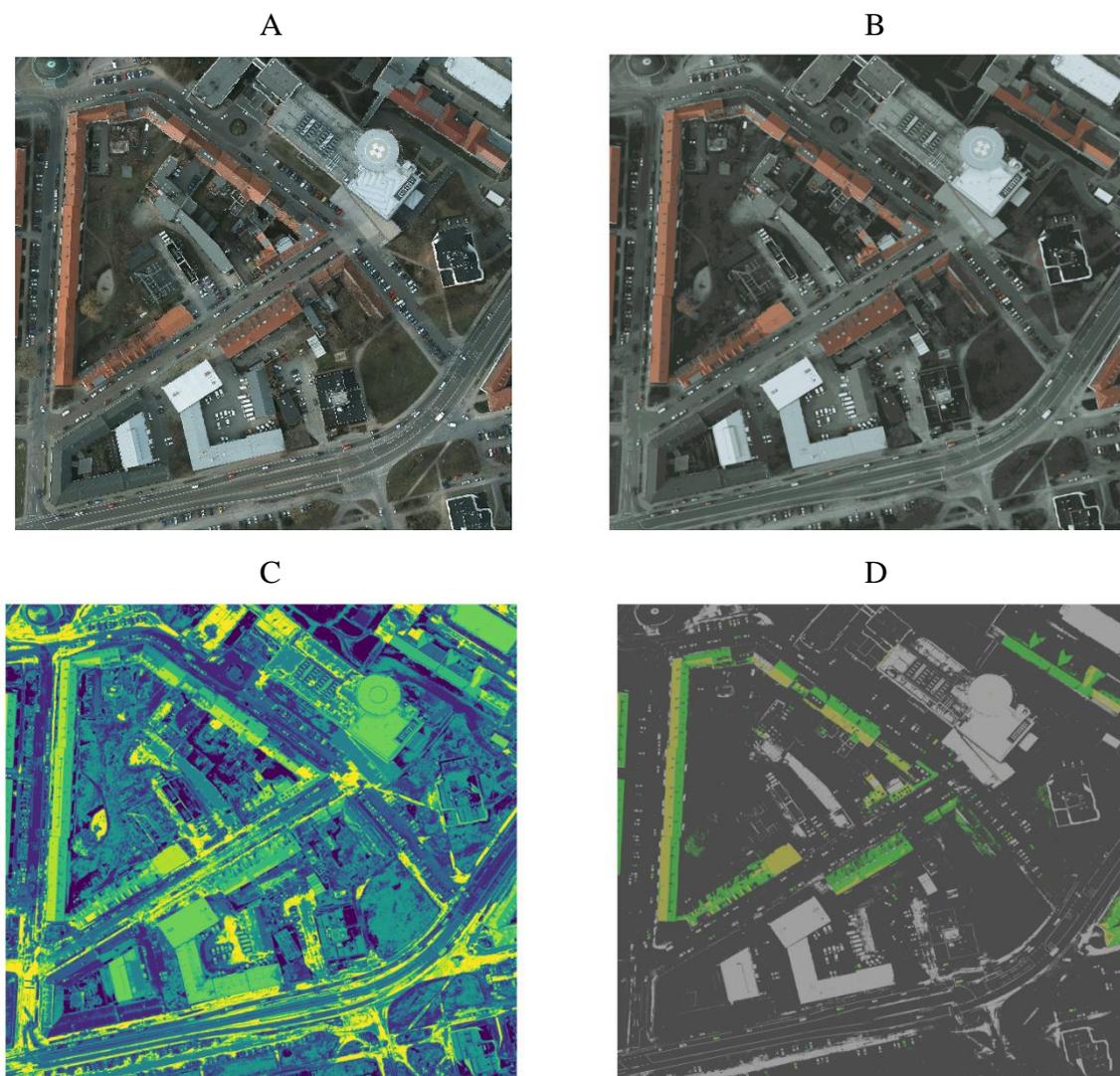


Figura 59 - ortofoto 6_14

Em relação ao resultado apresentado pela Figura 60 - ortofoto 6_15, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3195, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3313, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5581, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

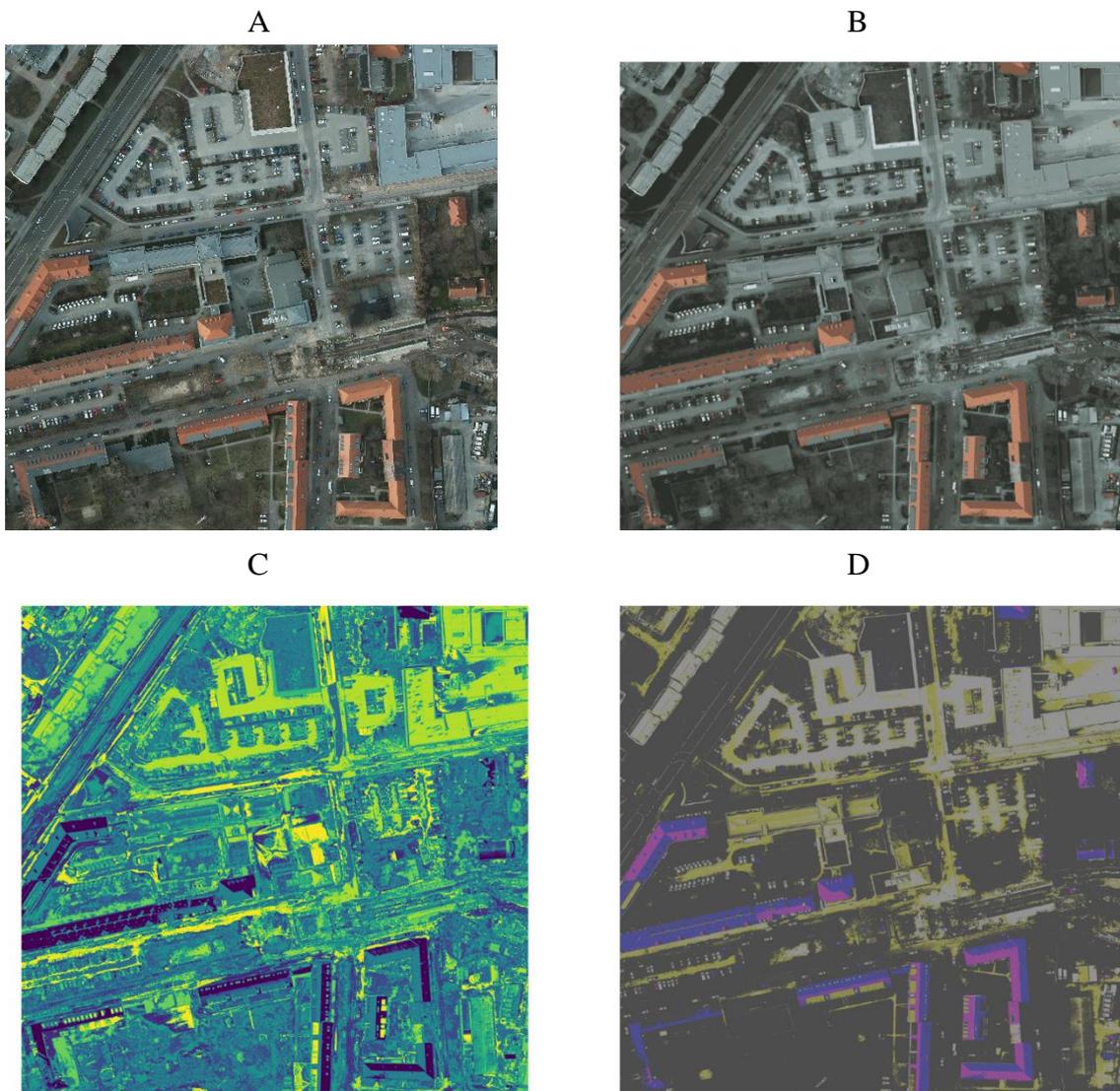


Figura 60 - ortofoto 6_15

Em relação ao resultado apresentado pela Figura 61 - ortofoto 7_7, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3143, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3115, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5734, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

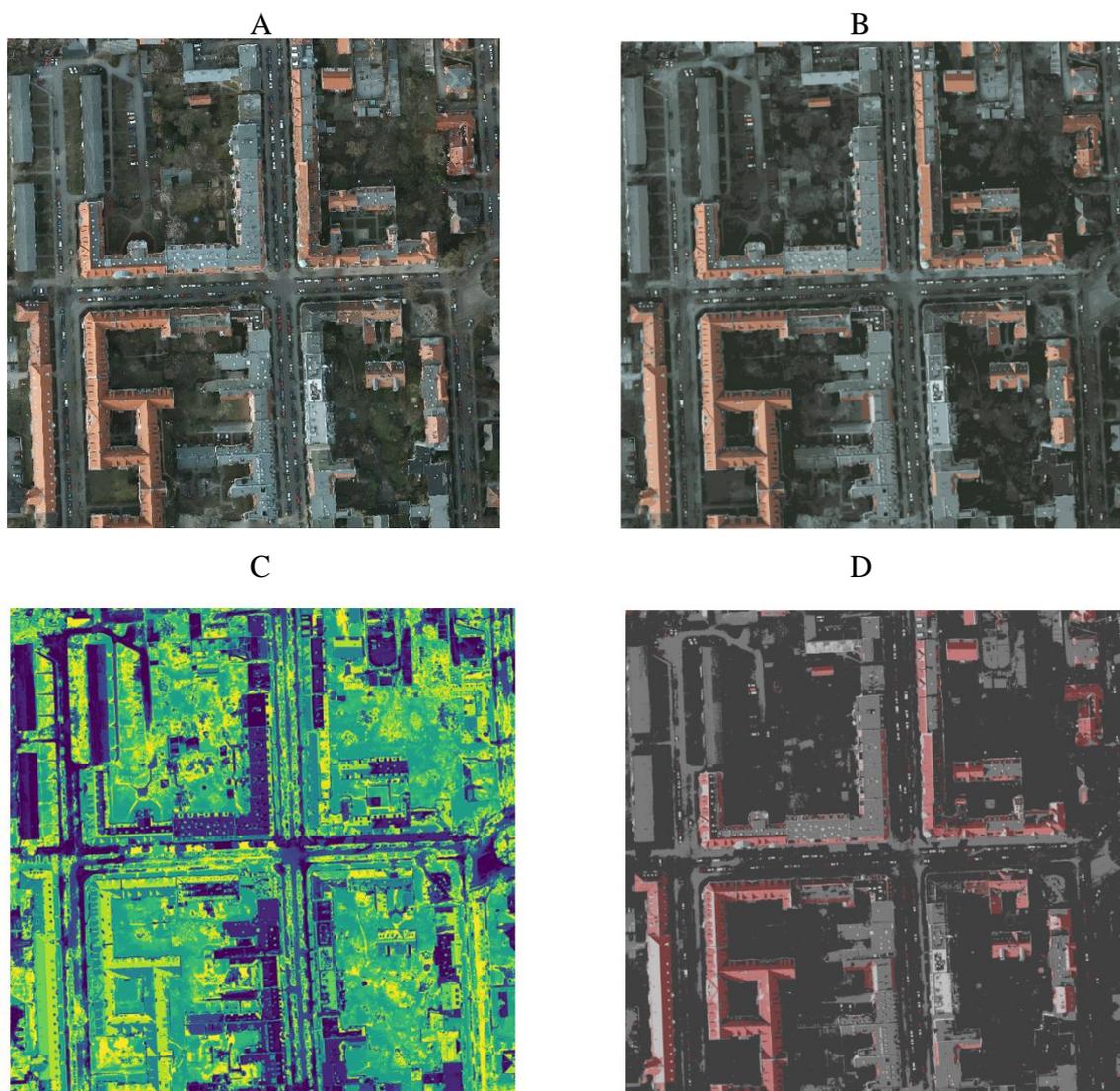


Figura 61 - ortofoto 7_7

Em relação ao resultado apresentado pela Figura 62 - ortofoto 7_8, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2719, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2804, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,5172, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

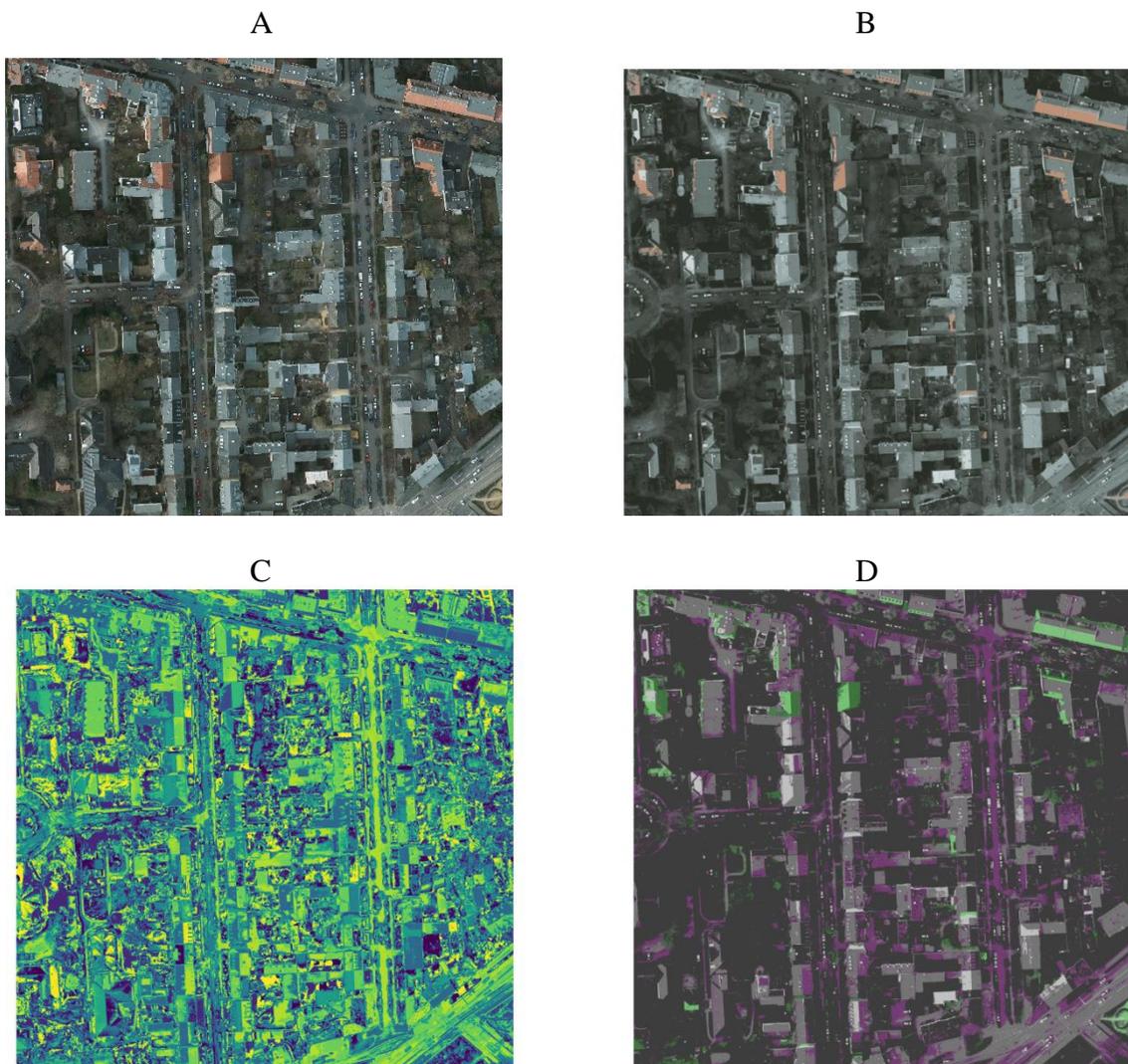


Figura 62 - ortofoto 7_8

Em relação ao resultado apresentado pela Figura 63 - ortofoto 7_9, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2952, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3186, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6115, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

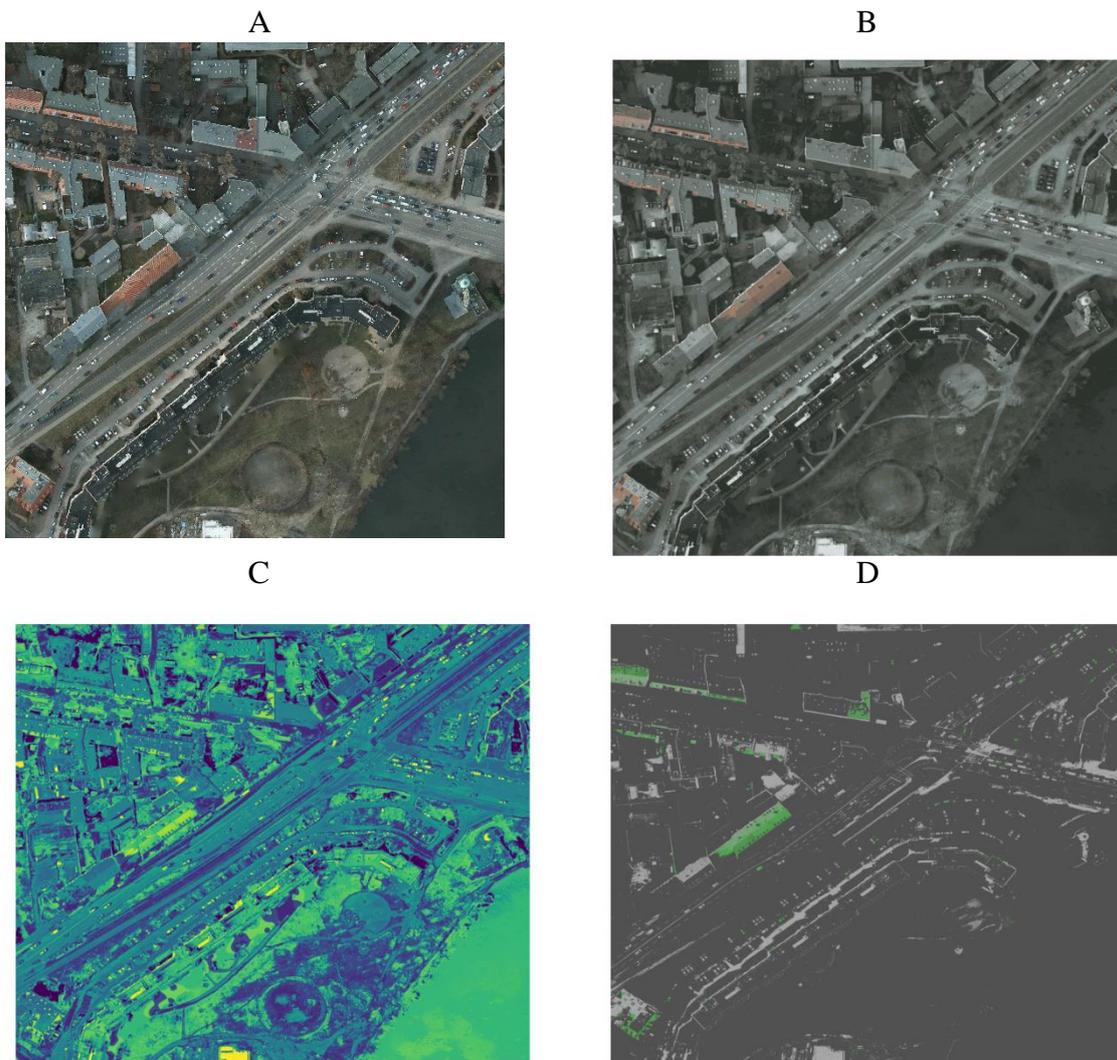


Figura 63 - ortofoto 7_9

Em relação ao resultado apresentado pela Figura 64 - ortofoto 7_10, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3243, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3385, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6610, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

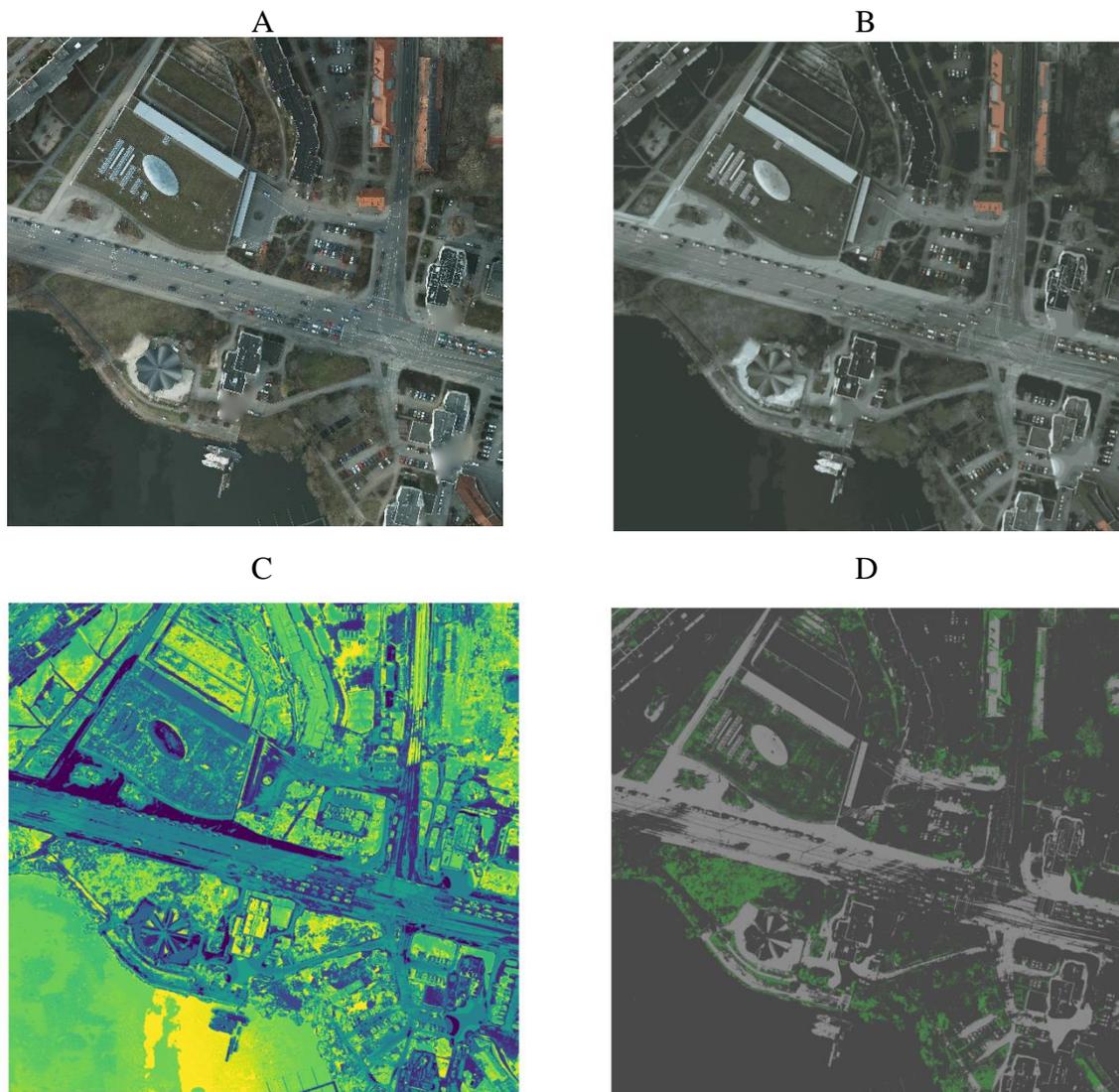


Figura 64 - ortofoto 7_10

Em relação ao resultado apresentado pela Figura 65 - ortofoto 7_11, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3383, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3260, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6340, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

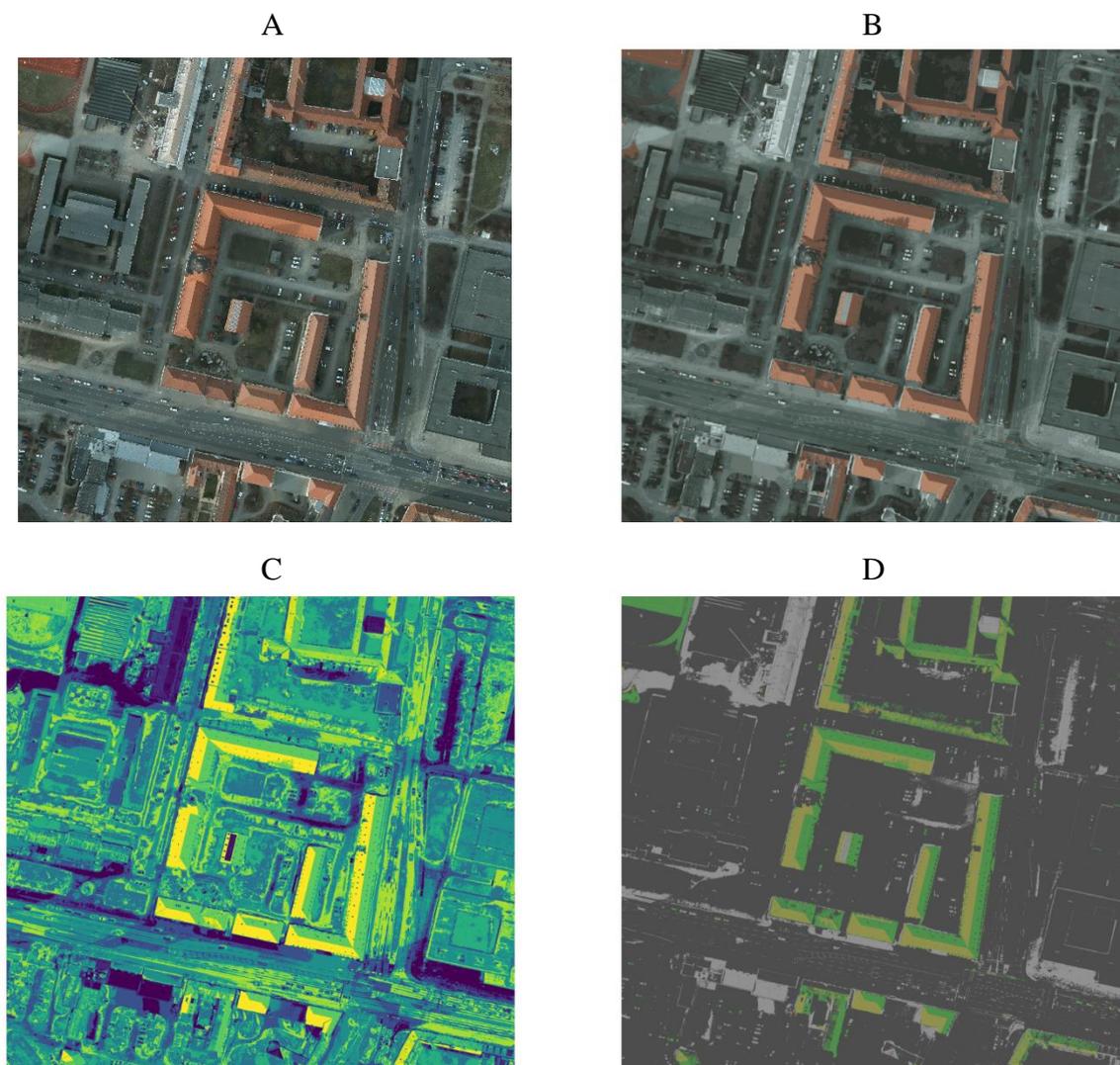


Figura 65 - ortofoto 7_11

Em relação ao resultado apresentado pela Figura 66 - ortofoto 7_12, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,2933, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,2929, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6562, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados.

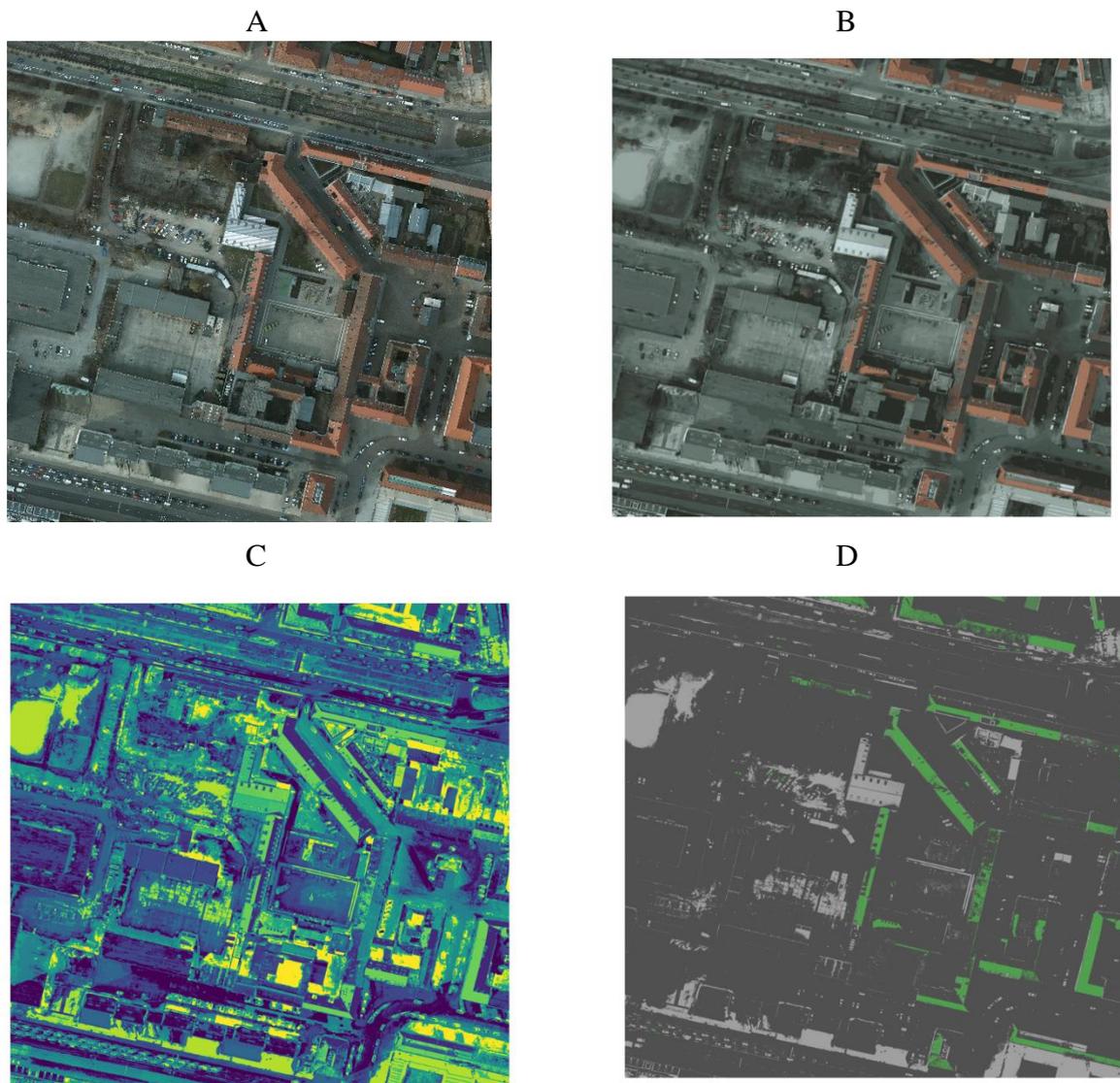


Figura 66 - ortofoto 7_12

Em relação ao resultado apresentado pela Figura 67 - ortofoto 7_13, tem-se que o Índice de Rand Ajustado calculado para o método K-médias é de 0,3071, imagem B; o referido índice calculado para o método *co-clustering* sem refinamento é de 0,3162, imagem C; e o referido índice calculado para o método *co-clustering* com refinamento K-médias é de 0,6989, sendo este o melhor resultado. Visualmente nota-se semelhança de resultado entre as imagens B e C, sendo coerente aos valores calculados

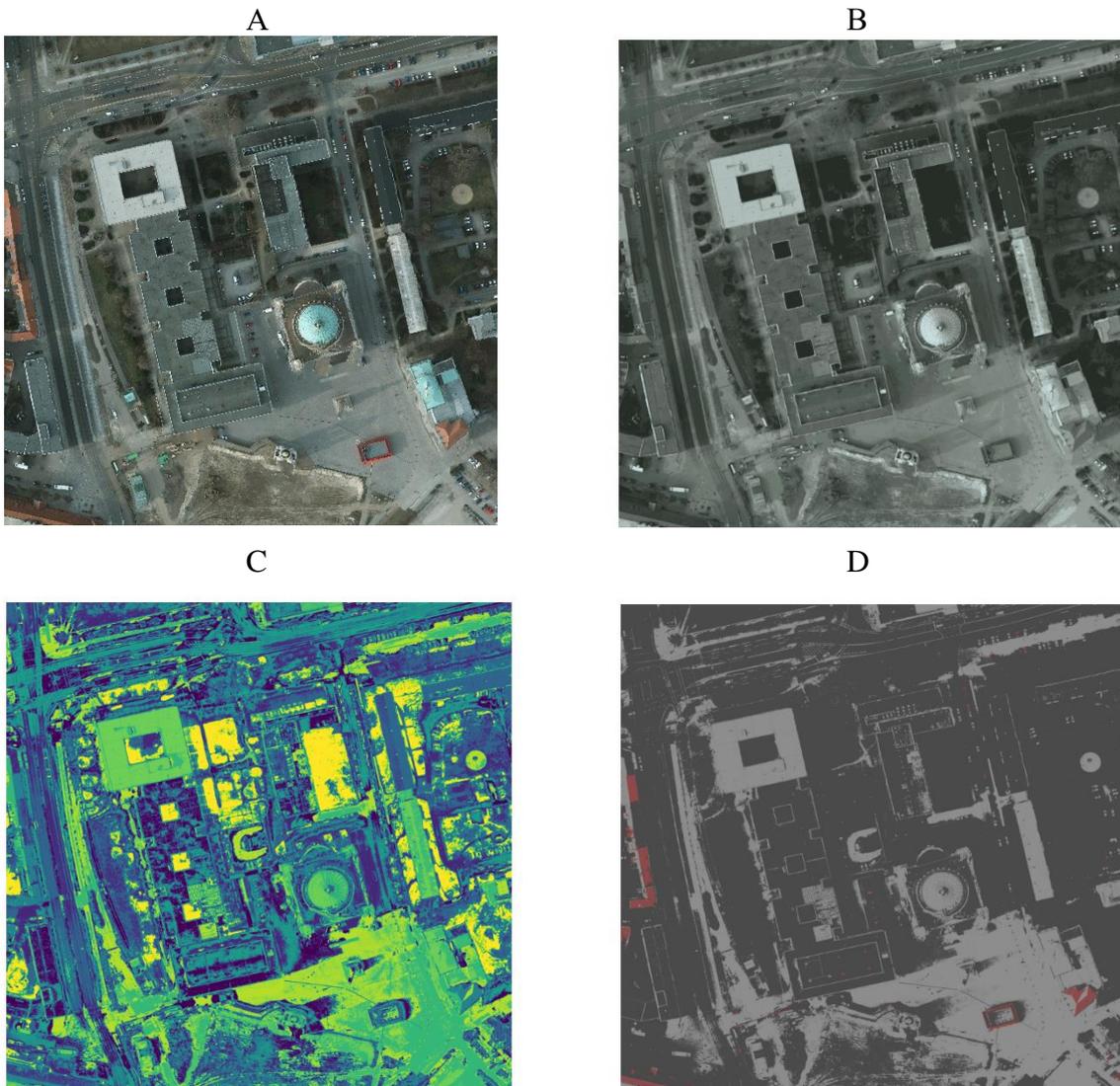


Figura 67 - ortofoto 7_13

4.2.7 Análise estatística dos resultados quanto à significância da melhora do resultado

A fim de se concluir pela melhora nos resultados das medidas de similaridade do método proposto neste trabalho de *co-clustering* com refinamento, quando em comparação ao método tradicional de classificação não supervisionada K-médias, considerando todos os valores calculados para os índices, realizou-se análise estatística dos resultados quanto à significância da igualdade das médias calculadas. Para tal, aplicou-se teste de igualdade entre duas amostras (*two-sample t-test*). Devido ao tamanho

da amostra $n=38$ ortofotos, maior que 30, a distribuição das médias amostrais pode ser aproximada satisfatoriamente por uma distribuição normal, logo, pode-se utilizar o valor de “z” no lugar de “t” para determinar se os resultados dos índices são significativamente iguais ou diferentes (DAVIS; WILEY, 2002).

Para realização do teste estatístico, selecionou-se a medida de similaridade dada pelo Índice de Rand Ajustado, sendo o referido índice o mais adequado para avaliar agrupamento dentre os apresentados neste trabalho, conforme apresentado na Seção 2.7.4.2.

Uma vez que a média e a variância dos valores calculados dos índices são conhecidos, adaptado de (DAVIS; WILEY, 2002), tem-se:

$$z = \frac{\bar{x}_1 - \bar{x}_2}{S_e} \tag{18}$$

$$S_e = S_p \sqrt{\frac{1}{n_1} + \frac{1}{n_2}} \tag{19}$$

$$S_p^2 = \frac{(n_1 - 1)S_1^2 + (n_2 - 1)S_2^2}{n_1 + n_2 - 2} \tag{20}$$

Em que:

\bar{x}_1 : média do índice 1

\bar{x}_2 : média do índice 2

$n_1 = n_2 = n$: tamanho da amostra

Quadro 8 – Média, Desvio padrão e Variância dos índices, por método, para todas as ortofotos

Média, Desvio padrão e Variância dos índices, por método, para todas as ortofotos			
Índice	Média	Desvio padrão	Variância
Jaccard (K-médias)	0,1680	0,0239	0,0006
Jaccard (Co-clustering)	0,1728	0,0256	0,0007
Jaccard (Co-clustering refinado)	0,4055	0,0881	0,0078
Rand (K-médias)	0,6342	0,0580	0,0034
Rand (Co-clustering)	0,6443	0,0587	0,0034
Rand (Co-clustering refinado)	0,5600	0,0438	0,0019
Rand Ajustado (K-médias)	0,2855	0,0370	0,0014
Rand Ajustado (Co-clustering)	0,2920	0,0396	0,0016
Rand Ajustado (Co-clustering refinado)	0,5610	0,1029	0,0106

A partir das equações (18), (19) e (20), substituindo-se pelos valores conhecidos das variáveis, e apresentados no Quadro 8, tem-se que:

Índice 1: índice Rand Ajustado para o método K-médias

Índice 2: Índice Rand Ajustado para o método *co-clustering* com refinamento

$$S_p^2 = 0.005973782$$

$$S_p = 0.077290247$$

$$S_e = 0.017731599$$

$$z = 15.56543225$$

Definição das hipóteses:

$$H_0: \bar{x}_1 = \bar{x}_2 \tag{21}$$

$$H_1: \bar{x}_1 \neq \bar{x}_2 \tag{22}$$

Definição do nível de confiança:

$$\alpha = 5\%$$

$$\therefore Z_{critico} = \pm 1,96$$

Como o valor obtido de $z = 15,56$ é maior que $1,96$, conclui-se que cai dentro da área de rejeição da hipótese nula H_0 . Portanto, conclui-se que o teste Z é significativo para um nível de significância de 5%, ou seja, as médias calculadas dos índices para os

métodos de classificação considerados não são iguais, logo o resultado da classificação *co-clustering* com refinamento é significativamente melhor que o resultado encontrado aplicando-se o método tradicional K-médias.

5 CONCLUSÕES

A partir do exposto, pode-se concluir que a técnica proposta de *co-clustering* com refinamento mostra-se como ferramenta eficaz para análise de imagens RGB de altíssima resolução, sendo em média, 1,942 vezes melhor, estatisticamente diferente, para um nível de significância de 5%, quando comparado ao resultado da classificação não supervisionada K-médias.

O objetivo proposto foi alcançado, uma vez que se produziu abordagem eficiente de técnica de *co-clustering* para análise de imagens multiespectrais de altíssima resolução espacial. Para tal, foi verificada a qualidade seguindo-se padrão de verificação do resultado para métodos de classificação não supervisionada a partir de medidas de similaridade, sendo elas índice de Jaccard, índice de Rand e índice de Rand Ajustado.

Pode-se concluir que o índice mais apropriado para a avaliação de resultado de *co-clustering* é o índice Rand Ajustado, uma vez que considera os chamados valores verdadeiros negativos, ou seja, os pares de pixels que estão em diferentes *cluster* e diferentes classes, quando em comparação com a classificação prévia e, adicionalmente, evidencia possíveis comportamentos aleatórios quando o resultado der negativo para o referido índice.

Avaliando-se os resultados das medidas de similaridade calculadas a partir do método *co-clustering* sem refinamento e calculados a partir do método *co-clustering* com refinamento, nota-se, de forma geral, pequena sensibilidade na formação dos agrupamentos finais ao se incluir camadas de morfologia matemática e de textura ao cubo de imagem original, nos parâmetros definidos, em comparação ao método aplicado ao cubo de imagem original. Portanto a inclusão dessas informações não altera de forma significativa os resultados de classificação não supervisionada pelo método *co-clustering*. Apesar da pequena sensibilidade do método aplicado ao cubo de imagem integrado com informações de contexto espacial em comparação ao cubo de imagem RGB original, destaca-se a aplicabilidade do referido cubo de imagem integrado, de forma a se explorar melhor o potencial de análise multivariada de dados da técnica de classificação *co-clustering*, quando aplicada a imagem RGB.

A partir do estudo desenvolvido por intermédio da investigação de dados multivariados e adaptação de técnicas de *co-clustering* para análise de imagens de

altíssima resolução espacial, foram aprofundados os conhecimentos no contexto amplo de mineração de dados.

A partir dos resultados encontrados e desta conclusão, sugere-se a investigação posterior e sugestão de trabalhos futuros a seguir:

- realizar análise geoestatística do agrupamento de bandas resultantes, uma vez que o escopo deste trabalho se limitou ao estudo dos *clusters* de pixels, ou seja, na dimensão espacial;

- utilizar o resultado de segmentação semântica encontrado neste projeto como insumo para rotinas de redes neurais;

- utilizar o resultado de segmentação semântica encontrado neste projeto como insumo para diversas rotinas de vetorização automática;

- testar diferentes elementos estruturais no código de morfologia matemática, como por exemplo um retângulo em vez do disco;

- testar imagens em diferentes cenários, além daqueles em áreas urbanas.

Exemplo: grandes coberturas vegetais como na região amazônica;

- testar imagens cobrindo áreas maiores, portanto com maior variação espectral e causa de possíveis confusões espectrais;

- testar imagens em diferentes escalas e resoluções, especialmente aquelas no nível de escala cadastral;

- testar imagens multiespectrais orbitais como insumo do processo apresentado;

- desenvolver metodologias para comparação com métodos de classificação supervisionada;

- a partir do código de *co-clustering*, subdividir o resultado virtualmente encontrado para agrupamento em bandas, apresentado pela Figura 17, não explorado no escopo deste trabalho, de forma a se obter um tipo de *co-clustering* irregular, a partir dos diferentes grupos de colunas obtidos.

REFERÊNCIAS BIBLIOGRÁFICAS

ABOUD NETA, Sumaia; BIAS, Edilson; DOS SANTOS, Celso Aparecido. Classificação Baseada em Objetos e Mineração de Dados: Aplicação de Ferramentas Open Source no Mapeamento de Área Urbana. **Revista Brasileira de Cartografia**, [S. l.], v. 70, n. 3, p. 932–966, 2018. Disponível em: <https://doi.org/10.14393/RBCV70N3-45707>

ANDALÓ, F. A. *et al.* A new shape descriptor based on Tensor Scale. **Mathematical Morphology and its Applications to Signal and Image Processing International Symposium on Mathematical Morphology**, [S. l.], p. 141–152, 2007.

ANTUNES, Rodrigo *et al.* Análise de Integração de Mineradores de Dados com a Plataforma InterIMAGE – Qual a Melhor Solução? **Revista Brasileira de Cartografia**, [S. l.], v. 70, n. 4, p. 1470–1509, 2018 a. Disponível em: <https://doi.org/10.14393/RBCV70N4-46512>

ANTUNES, Rodrigo Rodrigues *et al.* OBJECT-BASED ANALYSIS FOR URBAN LAND COVER MAPPING USING THE INTERIMAGE AND THE SIPINA FREE SOFTWARE PACKAGES. **Boletim de Ciências Geodésicas**, [S. l.], v. 24, n. 1, p. 01–17, 2018 b. Disponível em: <https://doi.org/10.1590/S1982-21702018000100001>. Acesso em: 1 mar. 2022.

APTOULA, Erchan; LEFÈVRE, Sébastien. On morphological color texture characterization. [S. l.], n. May 2014, p. 153–164, 2007.

AUCKLAND UNIVERSITY. **Morphological Image Processing**. [s. l.], [s. d.]. Disponível em: <https://www.cs.auckland.ac.nz/courses/compsci773s1c/lectures/ImageProcessing-html/topic4.htm>. Acesso em: 24 ago. 2021.

BANERJEE, Arindam *et al.* A generalized maximum entropy approach to Bregman Co-clustering and matrix approximation. **Journal of Machine Learning Research**, [S. l.], v. 8, p. 1919–1986, 2007. Disponível em: <https://www.scopus.com/record/display.uri?eid=2-s2.0-34548691246&origin=inward&txGid=6f09858c5805b23a5f1ccc9fbc5d70>

BANON, G.; BARRERA, J.; BRAGA-NETO, U. Mathematical Morphology and Its Applications to Signal and Image Processing. *In*: 2007, São José dos Campos. **8th International Symposium on Mathematical Morphology**. São José dos Campos:

INPE, 2007.

BRASIL. **DECRETO-LEI Nº 243, DE 28 DE FEVEREIRO DE 1967. Fixa as Diretrizes e Bases da Cartografia Brasileira e dá outras providências.** [S. l.: s. n.] Disponível em: http://www.planalto.gov.br/ccivil_03/Decreto-Lei/1965-1988/De10243.htm

BRASIL. **6.666 - DECRETO Nº 6.666, DE 27 DE NOVEMBRO DE 2008 - Institui, no âmbito do Poder Executivo federal, a Infra-Estrutura Nacional de Dados Espaciais - INDE, e dá outras providências.** [S. l.], 2008. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2007-2010/2008/decreto/d6666.htm. Acesso em: 9 out. 2020.

BROWN, Martin; LEWIS, Hugh G.; GUNN, Steve R. Linear Spectral Mixture Models and Support Vector Machines for Remote Sensing. **IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING**, [S. l.], v. 38, n. 5, 2000.

CHENG, Y.; CHURCH, G. M. Biclustering of expression data. **Proceedings / ... International Conference on Intelligent Systems for Molecular Biology ; ISMB. International Conference on Intelligent Systems for Molecular Biology**, [S. l.], v. 8, p. 93–103, 2000.

CINDE. **Plano de Ação para Implantação da Infraestrutura Nacional de Dados Espaciais.** [S. l.: s. n.]. Disponível em: <https://www.concar.gov.br/pdf/PlanoDeAcaoINDE.pdf>. Acesso em: 9 out. 2020.

COHEN, Jacob. A Coefficient of Agreement for Nominal Scales. **Educational and Psychological Measurement**, [S. l.], v. 20, n. 1, p. 37–46, 1960. Disponível em: <https://doi.org/10.1177/001316446002000104>. Acesso em: 28 jul. 2021.

CONGALTON, Russell G. A review of assessing the accuracy of classifications of remotely sensed data. **Remote Sensing of Environment**, [S. l.], v. 37, n. 1, p. 35–46, 1991. Disponível em: [https://doi.org/10.1016/0034-4257\(91\)90048-B](https://doi.org/10.1016/0034-4257(91)90048-B)

CONGALTON, Russell G. Accuracy assessment and validation of remotely sensed and other spatial information. **International Journal of Wildland Fire**, [S. l.], v. 10, n. 3–4, p. 321–328, 2001. Disponível em: <https://doi.org/10.1071/wf01031>

CONGALTON, Russell G.; GREEN, Kass. **Assessing the Accuracy of Remotely Sensed Data.** [S. l.]: CRC Press, 2008. *E-book*. Disponível em: <https://doi.org/10.1201/9781420055139>

DAVIS, John C.; WILEY, John. **Statistics Data Analysis Geology**. 3rd. ed. [S. l.]: John Wiley and Sons, 2002. *E-book*.

DENG, Shupe; ZHANG, Ye; TIAN, Shu. Building surface texture segmentation in urban remote sensing image using improved ORTSEG algorithm. **International Geoscience and Remote Sensing Symposium (IGARSS)**, [S. l.], v. 2015- November, p. 4348–4351, 2015. Disponível em: <https://doi.org/10.1109/IGARSS.2015.7326789>

DI, Yanan *et al.* Multi-scale segmentation of high resolution remote sensing images by integrating multiple features. *In*: 2017, **International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives**. [S. l.: s. n.] p. 247–255. Disponível em: <https://doi.org/10.5194/isprs-archives-XLII-1-W1-247-2017>

DIAZ, Alexandra Katiuska Ramos; PERES, Sarajane Marques. Biclustering and coclustering: Concepts, algorithms and viability for text mining. **Revista de Informatica Teorica e Aplicada**, [S. l.], v. 26, n. 2, p. 81–117, 2019. Disponível em: <https://doi.org/10.22456/2175-2745.89063>

DSG. Norma para Especificação Técnica para Produtos de Conjuntos de Dados Geoespaciais (ET-PCDG). **Portaria Nº - 008 - DCT, de 10 Fev 16**, [S. l.], n. II, p. 195, 2016 a. Disponível em: <https://doi.org/EB80-N-72.001>

DSG. **Especificação Técnica para Controle de Qualidade de Dados Geoespaciais (ET-CQDG)**. [s. l.], 2016b. Disponível em: <http://www.geoportal.eb.mil.br/portal/inde2?id=142>. Acesso em: 6 set. 2020.

DU, Shide *et al.* Differentiable bi-sparse multi-view co-clustering. **IEEE Transactions on Signal Processing**, [S. l.], v. 69, p. 4623–4636, 2021. Disponível em: <https://doi.org/10.1109/TSP.2021.3101979>

FAUVEL, Mathieu *et al.* Spectral and spatial classification of hyperspectral data using SVMs and morphological profiles. **IEEE Transactions on Geoscience and Remote Sensing**, [S. l.], v. 46, n. 11, p. 3804–3814, 2008. Disponível em: <https://doi.org/10.1109/TGRS.2008.922034>

FERREIRA, Daniela Portes Leal. **Divergências de Bregman e total Bregman aplicadas na análise de imagens**. 2017. - Universidade Federal de Uberlândia, Uberlândia, 2017. Disponível em: <https://doi.org/10.14393/UFU.TE.2018.791>. Acesso em: 29 jun. 2022.

GHOLIZADEH, Hamed; ZOEJ, Mohammad Javad Valadan; MOJARADI, Barat. A novel hyperspectral image clustering method based on spectral unmixing. *In*: 2012, **IEEE Aerospace Conference Proceedings**. [S. l.: s. n.] Disponível em: <https://doi.org/10.1109/AERO.2012.6187196>

GOVAERT, Gérard; NADIF, Mohamed. **Co-clustering: Models, Algorithms and applications**. [S. l.]: ISTE Ltd and John Wiley & Sons, Inc., 2014. *E-book*.

HAMEDIANFAR, Alireza; SHAFRI, Helmi Zulhaidi Mohd. Integrated approach using data mining-based decision tree and object-based image analysis for high-resolution urban mapping of WorldView-2 satellite sensor data. **Journal of Applied Remote Sensing**, [S. l.], v. 10, n. 2, p. 025001, 2016. Disponível em: <https://doi.org/10.1117/1.JRS.10.025001>. Acesso em: 25 nov. 2021.

HAN, Jiawei; KAMBER, Micheline; PEI, Jian. **Data Mining: Concepts and Techniques**. 3rd editio ed. [S. l.: s. n.]. *E-book*.

HARALICK, Robert M.; DINSTEN, Its'hak; SHANMUGAM, K. Textural Features for Image Classification. **IEEE Transactions on Systems, Man and Cybernetics**, [S. l.], v. SMC-3, n. 6, p. 610–621, 1973. Disponível em: <https://doi.org/10.1109/TSMC.1973.4309314>

HARTIGAN, J. A. Direct Clustering of a Data Matrix. **Journal of the American Statistical Association**, [S. l.], v. 67, n. 337, p. 123–129, 1972. Disponível em: <https://doi.org/10.1080/01621459.1972.10481214>

HOSSAIN, Mohammad D.; CHEN, Dongmei. Segmentation for Object-Based Image Analysis (OBIA): A review of algorithms and challenges from remote sensing perspective. **ISPRS Journal of Photogrammetry and Remote Sensing**, [S. l.], v. 150, p. 115–134, 2019. Disponível em: <https://doi.org/10.1016/J.ISPRSJPRS.2019.02.009>

HU, Huanjun *et al.* Classification of Very High-Resolution Remote Sensing Imagery Using a Fully Convolutional Network with Global and Local Context Information Enhancements. **IEEE Access**, [S. l.], v. 8, p. 14606–14619, 2020. Disponível em: <https://doi.org/10.1109/ACCESS.2020.2964760>

HU, Shizhe; YAN, Xiaoqiang; YE, Yangdong. Dynamic auto-weighted multi-view co-clustering. **Pattern Recognition**, [S. l.], v. 99, p. 107101, 2020. Disponível em: <https://doi.org/10.1016/J.PATCOG.2019.107101>

HUANG, Shudong *et al.* Spectral co-clustering ensemble. **Knowledge-Based**

Systems, [S. l.], v. 84, p. 46–55, 2015. Disponível em: <https://doi.org/10.1016/j.knosys.2015.03.027>. Acesso em: 25 nov. 2021.

HUBERT, Lawrence; ARABIE, Phipps. Comparing partitions. **Journal of Classification** 1985 2:1, [S. l.], v. 2, n. 1, p. 193–218, 1985. Disponível em: <https://doi.org/10.1007/BF01908075>. Acesso em: 23 ago. 2021.

InterIMAGE. . [s. l.], [s. d.]. Disponível em: <http://www.lvc.ele.puc-rio.br/projects/interimage/pt-br/>. Acesso em: 1 mar. 2022.

ISO. ISO. [s. l.], [s. d.]. Disponível em: <https://www.iso.org/about-us.html>. Acesso em: 10 jan. 2022.

ISO. ISO - ISO 2859-2:1985 - Sampling procedures for inspection by attributes — Part 2: Sampling plans indexed by limiting quality (LQ) for isolated lot inspection. [S. l.], 1985. Disponível em: <https://www.iso.org/standard/7867.html>. Acesso em: 26 out. 2020.

ISO. ISO 2859-1:1999 Sampling procedures for inspection by attributes. **ISO**, [S. l.], 1999.

ISO. ISO - ISO/FDIS 19131 - Geographic information — Data product specifications. 2. ed. [S. l.: s. n.]. *E-book*. Disponível em: <https://www.iso.org/standard/71297.html>. Acesso em: 10 jan. 2022.

ISO. ISO 19157 Geographic information - Data quality. [S. l.: s. n.]. *E-book*. Disponível em: <https://www.iso.org/obp/ui/#iso:std:iso:19157:ed-1:v1:en>. Acesso em: 9 out. 2020.

ISPRS. **2D Semantic Labeling Contest - Potsdam**. [s. l.], 2013. Disponível em: <https://www2.isprs.org/commissions/comm2/wg4/benchmark/2d-sem-label-potsdam/>. Acesso em: 24 jun. 2021.

ISPRS. **The International Society for Photogrammetry and remote Sensing**. [s. l.], 2022. Disponível em: <https://www.isprs.org>. Acesso em: 4 jan. 2022.

IZQUIERDO-VERDIGUIER, Emma; ZURITA-MILLA, Raul. Introducing co-clustering for hyperspectral image analysis. *In*: 2015, **International Geoscience and Remote Sensing Symposium (IGARSS)**. [S. l.: s. n.] p. 465–468. Disponível em: <https://doi.org/10.1109/IGARSS.2015.7325801>

JACCARD, Paul. THE DISTRIBUTION OF THE FLORA IN THE ALPINE ZONE.1. **New Phytologist**, [S. l.], v. 11, n. 2, p. 37–50, 1912. Disponível em:

<https://doi.org/10.1111/J.1469-8137.1912.TB05611.X>. Acesso em: 23 ago. 2021.

KEUPER, Margret *et al.* Motion Segmentation Multiple Object Tracking by Correlation Co-Clustering. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, [S. l.], v. 42, n. 1, p. 140–153, 2020. Disponível em: <https://doi.org/10.1109/TPAMI.2018.2876253>

KUPIDURA, Przemysław. The comparison of different methods of texture analysis for their efficacy for land use classification in satellite imagery. **Remote Sensing**, [S. l.], v. 11, n. 10, 2019. Disponível em: <https://doi.org/10.3390/rs11101233>

LANDIS, J. Richard; KOCH, Gary G. The Measurement of Observer Agreement for Categorical Data. **Biometrics**, [S. l.], v. 33, n. 1, p. 159, 1977. Disponível em: <https://doi.org/10.2307/2529310>

LI, Qianwen; WEI, Zihua; ZHAO, Cairong. Optimized Automatic Seeded Region Growing Algorithm with Application to ROI Extraction. <https://doi.org/10.1142/S0219467817500243>, [S. l.], v. 17, n. 4, 2017. Disponível em: <https://doi.org/10.1142/S0219467817500243>

MAROTTA, Victor. **Aprendizado não supervisionado com o kMeans**. [S. l.], 2019. Disponível em: <https://medium.com/neuronio-br/aprendizado-nao-supervisionado-com-k-means-f4272dee98a0>. Acesso em: 30 jun. 2022.

MENESES, Paulo Roberto *et al.* **Introdução ao Processamento de imagens de Sensoriamento Remoto**. Brasília: [s. n.], 2012. *E-book*.

MOHAJERANI, Sorour; KRAMMER, Thomas A.; SAEEDI, Parvaneh. Cloud Detection Algorithm for Remote Sensing Images Using Fully Convolutional Neural Networks. **2018 IEEE 20th International Workshop on Multimedia Signal Processing, MMSP 2018**, [S. l.], 2018. Disponível em: <https://doi.org/10.1109/MMSP.2018.8547095>. Acesso em: 3 jan. 2022.

NAKAO, Eduardo Kazuo. **Extracao de características e aprendizado nao-supervisionados em imagens hiperespectrais**. 2020. - UNIVERSIDADE FEDERAL DE SÃO CARLOS, [S. l.], 2020. Disponível em: <https://repositorio.ufscar.br/handle/ufscar/12826>

NASCIMENTO, R. F. F. *et al.* O algoritmo SVM: avaliação da separação ótima de classes em imagens CCD-CBERS-2. **XIV Simpósio Brasileiro de Sensoriamento Remoto**, [S. l.], p. 2079–2086, 2009. Disponível em:

<http://glcf.umiacs.umd.edu/index.shtml>. Acesso em: 11 jan. 2022.

NATTINO, F. *et al.* **Clustering Geo-Data Cubes (CGC): A Clustering Tool for Geospatial Applications (Version 0.4.0) [Computer software]**. [S. l.: s. n.] Disponível em: <https://doi.org/10.5281/zenodo.3979172>

NATTINO, Francesco *et al.* CGC: a Scalable Python Package for Co- and Tri-Clustering of Geodata Cubes. **Journal of Open Source Software**, [S. l.], v. 7, n. 72, p. 4032, 2022. Disponível em: <https://doi.org/10.21105/joss.04032>

NIXON, Mark S.; AGUADO, Alberto S. **Feature Extraction & Image Processing for Computer Vision**. 3. ed. [S. l.]: Elsevier, 2013. *E-book*. Disponível em: <https://doi.org/10.1016/C2011-0-06935-1>

PAHL, Martin *et al.* **Arquitetura de um sistema baseado em conhecimento para a interpretação de dados de sensoriamento remoto de múltiplos sensores**. 2003. - Universidade de Hannover, [s. l.], 2003. Disponível em: <http://dnb.ddb.de>. Acesso em: 1 mar. 2022.

PAL, M. Random forest classifier for remote sensing classification. **International Journal of Remote Sensing**, [S. l.], v. 26, n. 1, p. 217–222, 2005. Disponível em: <https://doi.org/10.1080/01431160412331269698>. Acesso em: 11 jan. 2022.

PAPALEXAKIS, Evangelos E.; SIDIROPOULOS, Nicholas D.; BRO, Rasmus. From K-means to higher-way co-clustering: Multilinear decomposition with sparse latent factors. **IEEE Transactions on Signal Processing**, [S. l.], v. 61, n. 2, p. 493–506, 2013. Disponível em: <https://doi.org/10.1109/TSP.2012.2225052>

PASSO, Denilson Pereira *et al.* USO DO SISTEMA INTERIMAGE PARA A IDENTIFICAÇÃO DE ALVOS URBANOS EM IMAGENS DO SATÉLITE WORLDVIEW II. **Revista Brasileira de Cartografia**, [S. l.], v. 65, n. 6, p. 1211–1221, 2013. Disponível em: https://www.researchgate.net/publication/267638575_Uso_do_Sistema_Interimage_para_a_Identificacao_de_Alvos_Urbanos_em_Imagens_do_Satelite_Worldview_II. Acesso em: 1 mar. 2022.

PEDREGOSA, Fabian *et al.* Scikit-learn: Machine Learning in Python. **Journal of Machine Learning Research**, [S. l.], v. 12, p. 2825–2830, 2011. Disponível em: <http://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html> <http://arxiv.org/abs/1201.0490>

PENSA, Ruggero G. *et al.* Co-clustering numerical data under user-defined constraints. **Statistical Analysis and Data Mining**, [S. l.], v. 3, n. 1, p. 38–55, 2010. Disponível em: <https://doi.org/10.1002/sam.10064>

PYTHON SOFTWARE FOUNDATION. **Python 3.9.0**. [s. l.], 2020. Disponível em: <https://www.python.org/downloads/release/python-390/>. Acesso em: 25 ago. 2021.

QGIS. **Bem-vindo ao projeto QGIS**. [S. l.: s. n.] Disponível em: https://www.qgis.org/pt_BR/site/. Acesso em: 7 set. 2020.

QIAN, Xueming *et al.* Social media based event summarization by user–text–image co-clustering. **Knowledge-Based Systems**, [S. l.], v. 164, p. 107–121, 2019. Disponível em: <https://doi.org/10.1016/J.KNOSYS.2018.10.028>

R. KIRAN, Kumar; SAICHANDANA, Bai; SRINIVAS, Kudipudi. Dimensionality reduction and classification of hyperspectral images using genetic algorithm. **Indonesian Journal of Electrical Engineering and Computer Science**, [S. l.], v. 3, p. 503–511, 2016.

RAND, William M. Objective Criteria for the Evaluation of Clustering Methods. **Journal of the American Statistical Association**, [S. l.], v. 66, n. 336, p. 846, 1971. Disponível em: <https://doi.org/10.2307/2284239>

RETORNAZ, Thomas; MARCOTEGUI, Beatriz. Scene text localization based on the ultimate opening. **International Symposium on Mathematical Morphology**, [S. l.], v. 1, p. 177–188, 2007. Disponível em: <http://bibdigital.sid.inpe.br/rep-dpi.inpe.br/ismm@80/2007/03.19.21.38>

ROUSSEEUW, Peter J. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. **Journal of Computational and Applied Mathematics**, [S. l.], v. 20, n. C, p. 53–65, 1987. Disponível em: [https://doi.org/10.1016/0377-0427\(87\)90125-7](https://doi.org/10.1016/0377-0427(87)90125-7)

RUIZ, L. A.; FDEZ-SARRÍA, A.; RECIO, J. A. Texture feature extraction for classification of remote sensing data using wavelet decomposition: a comparative study. **20th ISPRS Congress**, [S. l.], v. 35, n. 1, p. 1109–1114, 2004. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.134.5492&rep=rep1&type=pdf>. Acesso em: 3 jan. 2022.

SAICHANDANA, B. *et al.* Application of BEMD and Hierarchical Image Fusion in Hyperspectral Image Classification. **International Journal of Computer Science and**

Information Security, [S. l.], v. 14, 2016. Disponível em: <https://sites.google.com/site/ijcsis/>. Acesso em: 3 jan. 2022.

SAICHANDANA, B.; SRINIVAS, K. Hyperspectral Image Denoising and Unsupervised Classification. **CSI Communications**, [S. l.], v. 39, n. 9, 2015. Disponível em: <http://csi-india.org.in/Communications/CSIC December 2015.pdf#page=26>

SHANG, Fanhua; JIAO, L. C.; WANG, Fei. Graph dual regularization non-negative matrix factorization for co-clustering. **Pattern Recognition**, [S. l.], v. 45, n. 6, p. 2237–2250, 2012. Disponível em: <https://doi.org/10.1016/J.PATCOG.2011.12.015>

SILLERO, N. *et al.* Biogeographical patterns derived from remote sensing variables: the amphibians and reptiles of the Iberian Peninsula. **Amphibia-Reptilia**, [S. l.], v. 30, n. 2, p. 185–206, 2009. Disponível em: <https://doi.org/10.1163/156853809788201207>. Acesso em: 11 jan. 2022.

SOMODI, Imelda *et al.* Recognition of the invasive species Robinia pseudacacia from combined remote sensing and GIS sources. **Biological Conservation**, [S. l.], v. 150, n. 1, p. 59–67, 2012. Disponível em: <https://doi.org/10.1016/J.BIOCON.2012.02.014>

TASDEMIR, Kadim; MOAZZEN, Yaser; YILDIRIM, Isa. An Approximate Spectral Clustering Ensemble for High Spatial Resolution Remote-Sensing Images. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, [S. l.], v. 8, n. 5, p. 1996–2004, 2015. Disponível em: <https://doi.org/10.1109/JSTARS.2015.2424292>

VALERO, Silvia *et al.* Directional mathematical morphology for the detection of the road network in very high resolution remote sensing images. *In*: 2009, **Proceedings - International Conference on Image Processing, ICIP**. : IEEE, 2009. p. 3725–3728. Disponível em: <https://doi.org/10.1109/ICIP.2009.5414344>. Acesso em: 13 jun. 2016.

WAGNER, Silke; WAGNER, Dorothea. Comparing Clusterings-An Overview *. [S. l.], 2007.

WANG, Jie *et al.* Object-Scale Adaptive Convolutional Neural Networks for High-Spatial Resolution Remote Sensing Image Classification. **IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing**, [S. l.], v. 14, p. 283–299, 2021. Disponível em: <https://doi.org/10.1109/JSTARS.2020.3041859>

WARNER, Timothy. Kernel-Based Texture in Remote Sensing Image Classification. **Geography Compass**, [S. l.], v. 5, n. 10, p. 781–798, 2011. Disponível

em: <https://doi.org/10.1111/J.1749-8198.2011.00451.X>. Acesso em: 3 jan. 2022.

WASEEM, Arbab *et al.* K-Means and ISODATA Clustering Algorithms for Landcover Classification Using Remote Sensing. **Sindh University Research Journal (Science Series)**, [S. l.], v. 48, n. 2, p. 315–318, 2016. Disponível em: https://www.researchgate.net/publication/303971825_K-Means_and_ISODATA_Clustering_Algorithms_for_Landcover_Classification_Using_Remote_Sensing. Acesso em: 30 jun. 2022.

WATRIN, Orlando dos Santos; FILHO, Mario Valério; SANTOS, João Roberto do. Transformação por Componentes Principais e por Decorrelação de Bandas Aplicadas à Caracterização de Feições da Paisagem do Nordeste Paraense 1. **Anais VIII Simpósio Brasileiro de Sensoriamento Remoto**, [S. l.], p. 435–441, 1996.

WU, Xiaojing *et al.* An overview of clustering methods for geo-referenced time series: from one-way clustering to co- and tri-clustering. **International Journal of Geographical Information Science**, [S. l.], v. 34, n. 9, p. 1822–1848, 2020. Disponível em: <https://doi.org/10.1080/13658816.2020.1726922>

WU, Xiaojing; ZURITA-MILLA, Raul; KRAAK, Menno Jan. Co-clustering geo-referenced time series: exploring spatio-temporal patterns in Dutch temperature data. **International Journal of Geographical Information Science**, [S. l.], v. 29, n. 4, p. 624–642, 2015. Disponível em: <https://doi.org/10.1080/13658816.2014.994520>

WU, Zhihuan *et al.* Semantic segmentation of high-resolution remote sensing images using fully convolutional network with adaptive threshold. <https://doi.org/10.1080/09540091.2018.1510902>, [S. l.], v. 31, n. 2, p. 169–184, 2018. Disponível em: <https://doi.org/10.1080/09540091.2018.1510902>. Acesso em: 3 jan. 2022.

YAO, Xiwen *et al.* Revisiting co-saliency detection: A novel approach based on two-stage multi-view spectral rotation co-clustering. **IEEE Transactions on Image Processing**, [S. l.], v. 26, n. 7, p. 3196–3209, 2017. Disponível em: <https://doi.org/10.1109/TIP.2017.2694222>

ZHANG, Xueliang; XIAO, Pengfeng; FENG, Xuezhi. An unsupervised evaluation method for remotely sensed imagery segmentation. **IEEE Geoscience and Remote Sensing Letters**, [S. l.], v. 9, n. 2, p. 156–160, 2012. Disponível em: <https://doi.org/10.1109/LGRS.2011.2163056>

ZHU, Changqing; YANG, Xiaomei. Study of remote sensing image texture analysis and classification using wavelet. <http://dx.doi.org/10.1080/014311698214262>, [S. l.], v. 19, n. 16, p. 3197–3203, 2010. Disponível em: <https://doi.org/10.1080/014311698214262>. Acesso em: 3 jan. 2022.

ZURITA-MILLA, R. *et al.* Clustering geo-data cubes: presentation. In: 2020, **Space and Artificial Intelligence 2020**. [S. l.: s. n.] p. S1–S16. Disponível em: http://spaceandai.ijs.si/Session 3_Raul Zurita-Milla.pdf

APÊNDICE 1

O código Python da função '*classif.py*'.

```
% -----  
%   Fabrício de Almeida Santos, 2021(c)  
%   fabricio_almeid@hotmail.com  
%  
% -----  
  
import sys, os  
  
sys.path.append(os.getcwd())  
from avaliacao import AvaliaClassificacao  
  
from PIL import Image  
import matplotlib.pyplot as plt  
import numpy as np  
import xarray as xr  
import os  
  
from cgc.coclustering import Coclustering  
from skimage.morphology import opening, closing, disk  
from skimage.filters.rank import entropy  
from cgc.Kmeans import Kmeans  
from cgc.utils import mem_estimate_coclustering_numpy,  
calculate_cocluster_averages  
  
class ClassificadorEmLote:  
  
    def __init__(self,  
                 imagePath,  
                 labelPath,  
                 sampleSize=800,  
                 num_clusters_banda=3,  
                 num_clusters_espaciais=10,  
                 max_iterations=50, # maximum number of iterations  
                 conv_threshold=0.001, # error convergence  
                 threshold,  
                 num_runs=10, # number of differently-initialized  
                 runs  
                 disks=[2, 4, 6]):  
        print('Iniciando')  
        self.sampleSize = sampleSize  
        self.num_band_clusters = num_clusters_banda  
        self.num_space_clusters = num_clusters_espaciais  
        self.max_iterations = max_iterations  
        self.conv_threshold = conv_threshold  
        self.nruns = num_runs  
        self.disks = disks  
        print('Abrindo imagem')
```

```

self.images = self.openImagesFromFolder(imagePath)
print('Abrindo label')
self.labels = self.openImagesFromFolder(labelPath)
self.jaccards = []
self.rands = []
self.aris = []
self.k_jaccards = []
self.k_rands = []
self.k_aris = []
self.k_values = []
self.rejected = []
for i, image in enumerate(self.images):
    print('Processando imagem {} de {}'.format(i+1,
len(self.images)))
    cube = self.createCube(image)
    cube = self.insertMorphologyandEntropy(cube)
    data = self.prepareCube(cube)
    for j in range(10):
        res = self.runCoclustering(data)
        banda_cluster, space_cluster, space_cluster_xy, ccimage
= self.formatCCResults(res)
        label =
Image.fromarray(self.labels[i].values.transpose(1,2,0), 'RGB')
        avaliador1 = AvaliaClassificacao(ccimage, label)
        avaliador1.acharClassesCorrelatas()
        avaliador1.agruparClasses()
        jaccard1, rand1, ari1 = avaliador1.calculaIndices()
        space_means, space_cluster = self.calcMeans(data,
banda_cluster, space_cluster)
        try:
            print('Tentando kvalue atÃ© o nÃºmero de classes do
label')
            classes = len(label.getcolors())
            km = self.runKmeans(data, res, classes)
            refined, optimalk =
self.refineKResults(km.compute(), space_cluster)
            kimage = Image.fromarray(np.uint8(255*refined.data))
            print('Iniciando avaliaÃ§Ã£o')
            avaliador2 = AvaliaClassificacao(kimage, label)
            print('Encontrando correlaÃ§Ã£o entre grupos e
classes')
            avaliador2.acharClassesCorrelatas()
            print('Agrupando Grupos de cada classe')
            classesAgrupadas = avaliador2.agruparClasses()
            print('Verificando necessidade de unir grupos')
            uniu = avaliador2.unirClasses(classesAgrupadas)
            if uniu:
                print('Grupos foram unidos, refazendo
correlaÃ§Ã£o')
                avaliador2.acharClassesCorrelatas()
                print('Calculando Ãndices')
                jaccard2, rand2, ari2 = avaliador2.calculaIndices()
                break

```

```

except Exception as e:
    print(e)
    try:
        print('Tentando kvalue atÃ© o nÃºmero de
clusters')
        classes = self.num_space_clusters
        km = self.runKmeans(data, res, classes)
        refined, optimalk =
self.refineKResults(km.compute(), space_cluster)
        kimage =
Image.fromarray(np.uint8(255*refined.data))
        print('Iniciando avaliaÃ§Ã£o')
        avaliador2 = AvaliaClassificacao(kimage, label)
        print('Encontrando correlaÃ§Ã£o entre grupos e
classes')

        avaliador2.acharClassesCorrelatas()
        print('Agrupando Grupos de cada classe')
        classesAgrupadas = avaliador2.agruparClasses()
        print('Verificando necessidade de unir grupos')
        uniu = avaliador2.unirClasses(classesAgrupadas)
        if uniu:
            print('Grupos foram unidos, refazendo
correlaÃ§Ã£o')
            avaliador2.acharClassesCorrelatas()
            print('Calculando Ãndices')
            jaccard2, rand2, ari2 =
avaliador2.calculaIndices()
            break
    except:
        if j == 9:
            print('Rejeitado')
            jaccard2, rand2, ari2 = 0, 0, 0
            optimalk = 0
            self.rejected.append(i)

self.jaccards.append(jaccard1)
self.k_jaccards.append(jaccard2)
self.rands.append(rand1)
self.k_rands.append(rand2)
self.aris.append(ari1)
self.k_aris.append(ari2)
self.k_values.append(optimalk)
print('Pronto')

self.showResults()

def openImagesFromFolder(self, path):
    print('Abrindo imagens da pasta')
    images = []
    if os.path.isdir(path):
        folder = path
        for f in os.listdir(folder):
            if '.tif' in f:

```

```

        try:
            img = xr.open_rasterio(f)
        except:
            try:
                img = xr.open_rasterio(folder+f)
            except:
                continue
        if max(img.shape) > self.sampleSize:
            sliced = self.sliceImage(img)
            images += sliced
        else:
            images.append(img)
    img = ''
else:
    try:
        img = xr.open_rasterio(path)
        if max(img.shape) > self.sampleSize:
            sliced = self.sliceImage(img)
            images += sliced
        else:
            images.append(img)
    img = ''
except:
    pass

print('Foram adicionadas {} imagens'.format(len(images)))
return images

def sliceImage(self, img):
    images = []
    s = self.sampleSize
    samplesInRow = int(img.shape[1]/s)
    samplesInCol = int(img.shape[2]/s)
    for i in range(samplesInCol):
        for j in range(samplesInRow):
            image = img[:,i*s:(i+1)*s,j*s:(j+1)*s]
            images.append(image)
    image = ''

    return images

def createCube(self, img):
    print('Criando cubo')
    bands = range(min(img.shape))
    cube = [img[band,:,:] for band in bands]
    print('Cubo criado')
    return cube

def insertMorphologyandEntropy(self, cube):
    print('Inserindo perfil morfologico e textura')
    addedBands = []
    dims = cube[0].dims
    coords = cube[0].coords

```

```

for band in cube:
    for size in self.disks:
        disco = disk(size)
        opened = opening(band, disco)
        closed = closing(band, disco)
        addedBands.append(xr.DataArray(opened, dims=dims,
coords=coords, name='o{}'.format(size)))
        addedBands.append(xr.DataArray(closed, dims=dims,
coords=coords, name='c{}'.format(size)))

        texture = entropy(band, disco)
        addedBands.append(xr.DataArray(texture, dims=dims,
coords=coords, name='e{}'.format(size)))

    cube += addedBands
    opened, closed, texture = 0, 0, 0
    print('Inseridos')

    return cube

def prepareCube(self, cube):
    print('Preparando cubo')
    bands = range(len(cube))
    d = xr.concat(cube, dim='band')#, coords='minimal',
compat='override')
    d = d.assign_coords(band=('band', bands))
    d = d.stack(space=['x', 'y'])
    location = np.arange(d.space.size) # create a combined (x,y)
index
    d = d.assign_coords(location=('space', location))
    d = d.dropna('space', how='any') # drop pixels that are
null-valued for any of the time indices
    d = d.astype('float')
    self.bandcoords = d.band.coords
    self.spacecoords = d.space.coords
    print('Cubo pronto para a classificacao')
    return d

def runCoclustering(self, array):
    print('Iniciando classificaÃ§Ã£o')
    cc = Coclustering(
        array.data,
        self.num_band_clusters, # number of clusters in the
first dimension of the data set (rows)
        self.num_space_clusters, # number of clusters in the
second dimension of the data set (columns)
        max_iterations=self.max_iterations,
        conv_threshold=self.conv_threshold,
        nruns=self.nruns
    )
    return cc.run_with_threads(nthreads=4, low_memory=True)

def formatCCResults(self, result):

```

```

    print('Formatando resultados')
    banda_clusters = xr.DataArray(result.row_clusters,
    dims='band',
                                coords=self.bandacoords,
                                name='banda cluster')
    space_clusters = xr.DataArray(result.col_clusters,
    dims='space',
                                coords=self.spacecoords,
                                name='space cluster')
    space_clusters_xy = space_clusters.unstack('space')
    ccimage =
Image.fromarray(np.uint8(int(255/(self.num_space_clusters-
1))*space_clusters_xy.values.transpose(1,0)), 'L')

    return banda_clusters, space_clusters, space_clusters_xy,
ccimage

# means 1
def calcMeans(self, array, banda_cluster, space_cluster):
    print('Iniciando cálculo dos clusters means')
    means = calculate_cocluster_averages(
        array.data,
        banda_cluster,
        space_cluster,
        self.num_band_clusters,
        self.num_space_clusters
    )
    means = xr.DataArray(means, dims=('band_clusters',
'space_clusters'))

    # drop co-clusters that are not populated
    means = means.dropna('band_clusters', how='all')
    means = means.dropna('space_clusters', how='all')

    # use first row- and column-clusters to sort
    space_cluster_order = means.isel(band_clusters=0)
    banda_cluster_order = means.isel(space_clusters=0)

    classes = list(set(space_cluster.data))
    for index, pixel in enumerate(space_cluster.data):
        space_cluster.data[index] = classes.index(pixel)

    space_means = means.sel(space_clusters=space_cluster)
    space_means = space_means.unstack('space')

    return space_means, space_cluster

def runKmeans(self, array, results, classes):
    print('Iniciando refinamento Kmeans')
    km = Kmeans(
        array.data,
        clusters=(results.row_clusters, results.col_clusters),

```

```

nclusters=(self.num_band_clusters,self.num_space_clusters),
           k_range=range(1, classes+1),
           kmean_max_iter=100
        )
        return km

def showResults(self, index):
    res = self.ccrests[index]
    banda_clusters = xr.DataArray(res.row_clusters, dims='band',
                                  coords=self.bandacoords,
                                  name='banda cluster')
    space_clusters = xr.DataArray(res.col_clusters,
                                  dims='space',
                                  coords=self.spacecoords,
                                  name='space cluster')
    space_clusters_xy = space_clusters.unstack('space')
    space_clusters_xy.plot.imshow(
        x='x', y='y', levels=range(self.num_space_clusters+1)
    )

# means 2
def refineKResults(self, kresults, space_cluster):
    means_Kmeans = xr.DataArray(
        kresults.cluster_averages,
        coords=(
            ('band_clusters', range(num_band_clusters)),
            ('space_clusters', range(num_space_clusters))
        )
    )

    # drop co-clusters that are not populated
    means_Kmeans = means_Kmeans.dropna('band_clusters',
                                        how='all')
    means_Kmeans = means_Kmeans.dropna('space_clusters',
                                        how='all')
    space_means =
means_Kmeans.sel(space_clusters=space_clusters, drop=True)
    space_means = space_means.unstack('space')

    return (space_means.transpose('y','x','band_clusters')/255),
kresults.k_value

def stratifyIndexes(self, idxList, limits):
    result = []
    limit = 0
    last = 0
    for i, value in enumerate(idxList):
        while value > limits[limit]:
            limit += 1
            result.append((i-last)/len(idxList))
            last = i
    result.append(1-(last/len(idxList)))

```

```

for j in range(limit+1, len(limits)):
    result.append(0.)

return result

def showResults(self):
    limits = [0.2,0.4,0.6,0.8,1]
    jaccards = self.jaccards
    rands = self.rands
    aris = self.aris
    print('Resultados CC')

    print('Média Jaccard:
{}'.format(sum(jaccards)/len(jaccards)))
    print('Má;x Jaccard: imagem {} ->
{}'.format(jaccards.index(max(jaccards)),max(jaccards)))
    print('Min Jaccard: imagem {} ->
{}'.format(jaccards.index(min(jaccards)),min(jaccards)))
    jaccards = sorted(self.jaccards)
    print(self.stratifyIndexes(jaccards, limits))

    print('Média Rand: {}'.format(sum(rands)/len(rands)))
    print('Má;x Rand: imagem {} ->
{}'.format(rands.index(max(rands)),max(rands)))
    print('Min Rand: imagem {} ->
{}'.format(rands.index(min(rands)),min(rands)))
    rands = sorted(self.rands)
    print(self.stratifyIndexes(rands, limits))

    print('Média Ari: {}'.format(sum(aris)/len(aris)))
    print('Má;x Ari: imagem {} ->
{}'.format(aris.index(max(aris)),max(aris)))
    print('Min Ari: imagem {} ->
{}'.format(aris.index(min(aris)),min(aris)))
    aris = sorted(self.aris)
    print(self.stratifyIndexes(aris, limits))

    ks = self.k_values
    jaccards = self.k_jaccards
    rands = self.k_rands
    aris = self.k_aris
    print('Resultados K-means')

    print('Média K-value: {}'.format(sum(ks)/len(ks)))
    print('Má;x K-value: imagem {} ->
{}'.format(ks.index(max(ks)),max(ks)))
    print('Min K-value: imagem {} ->
{}'.format(ks.index(min(ks)),min(ks)))
    klimits = [2,5,8,10,100]
    ks = sorted(self.k_values)
    print(self.stratifyIndexes(ks, klimits))

```

```

    print('Média Jaccard:
{}'.format(sum(jaccards)/len(jaccards)))
    print('Máx Jaccard: imagem {} ->
{}'.format(jaccards.index(max(jaccards)),max(jaccards)))
    print('Min Jaccard: imagem {} ->
{}'.format(jaccards.index(min(jaccards)),min(jaccards)))
    jaccards = sorted(self.k_jaccards)
    print(self.stratifyIndexes(jaccards, limits))

    print('Média Rand: {}'.format(sum(rands)/len(rands)))
    print('Máx Rand: imagem {} ->
{}'.format(rands.index(max(rands)),max(rands)))
    print('Min Rand: imagem {} ->
{}'.format(rands.index(min(rands)),min(rands)))
    rands = sorted(self.k_rands)
    print(self.stratifyIndexes(rands, limits))

    print('Média Ari: {}'.format(sum(aris)/len(aris)))
    print('Máx Ari: imagem {} ->
{}'.format(aris.index(max(aris)),max(aris)))
    print('Min Ari: imagem {} ->
{}'.format(aris.index(min(aris)),min(aris)))
    aris = sorted(self.k_aris)
    print(self.stratifyIndexes(aris, limits))

    print('Rejeitados pelo k-means: {}'.format(self.rejected))

```

APÊNDICE 2

O código Python da função ‘*Kmeans.py*’.

```
% -----  
%   Fabrício de Almeida Santos, 2021(c)  
%   fabricio_almeid@hotmail.com  
%  
% -----  
  
import cv2  
import numpy as np  
import threading  
import time  
SLICE_WIDTH =800  
SLICE_HEIGHT = 800  
criteria = (cv2.TERM_CRITERIA_EPS + cv2.TERM_CRITERIA_MAX_ITER,  
100, 0.2)  
  
image_paths = {}  
for i in range(5):  
    image_paths[f'/home/2_Ortho_RGB/top_potsdam_2_1{i}_RGB.tif'] =  
f'/home/2_Ortho_RGB/top_potsdam_2_1{i}_label.tif'  
  
def slice_image(full_image):  
    slices = []  
    height, width = full_image.shape[:2]  
    samplesInRow = int(width/SLICE_WIDTH)  
    samplesInCol = int(height/SLICE_HEIGHT)  
    for i in range(samplesInCol):  
        for j in range(samplesInRow):  
  
slices.append(full_image[i*SLICE_HEIGHT:(i+1)*SLICE_HEIGHT,  
j*SLICE_WIDTH:(j+1)*SLICE_WIDTH])  
  
    return slices  
  
def run_Kmeans(image, k):  
    # convert to RGB  
    image = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)  
    # reshape the image to a 2D array of pixels and 3 color values  
(RGB)  
    pixel_values = image.reshape((-1, 3))  
    # convert to float  
    pixel_values = np.float32(pixel_values)  
  
    _, labels, (centers) = cv2.Kmeans(pixel_values, k, None,  
criteria, 10, cv2.KMEANS_RANDOM_CENTERS)  
  
    # convert back to 8 bit values  
    centers = np.uint8(centers)  
    # flatten the labels array
```

```

labels = labels.flatten()
# convert all pixels to the color of the centroids
segmented_image = centers[labels.flatten()]
# reshape back to the original image dimension
return segmented_image.reshape(image.shape)

class AvaliaClassificacao:

    def __init__(self, imagem, label):
        self.imagem = imagem
        self.gabarito = label

    def combinacao2a2(self, num):
        return num*(num-1)/2

    def calculaIndices(self):
        ss, sd, ds, dd = 0, 0, 0, 0
        a, b, n = 0, 0, 0
        bi = {}
        for keyE1 in self.mapaImagemOrdenado.keys():
            ai = 0
            for keyE2 in self.mapaImagemOrdenado[keyE1].keys():
                if keyE2 in bi.keys():
                    bi[keyE2] += self.mapaImagemOrdenado[keyE1][keyE2]
                else:
                    bi[keyE2] = self.mapaImagemOrdenado[keyE1][keyE2]
                ai += self.mapaImagemOrdenado[keyE1][keyE2] #soma os ai
            n += self.combinacao2a2(self.mapaImagemOrdenado[keyE1][keyE2]) #soma os nij
            for keyI1 in self.mapaImagemOrdenado.keys():
                for keyI2 in self.mapaImagemOrdenado[keyI1].keys():
                    if keyE1 == keyI1 and keyE2 == keyI2:
                        ss += self.combinacao2a2(self.mapaImagemOrdenado[keyE1][keyE2])
                    elif keyE1 == keyI1:
                        sd += self.mapaImagemOrdenado[keyE1][keyE2]*self.mapaImagemOrdenado[keyI1][keyI2]/2
                    elif keyE2 == keyI2:
                        ds += self.mapaImagemOrdenado[keyE1][keyE2]*self.mapaImagemOrdenado[keyI1][keyI2]/2
                    else:
                        dd += self.mapaImagemOrdenado[keyE1][keyE2]*self.mapaImagemOrdenado[keyI1][keyI2]/2
                a += self.combinacao2a2(ai)

        for k in bi.keys():
            b += self.combinacao2a2(bi[k])

        t = self.combinacao2a2(ss+sd+ds+dd)

```

```

jaccard = ss/(sd+ds+ss)
rand = (ss + dd)/(sd+ds+ss+dd)
ari = (n - (a*b)/t)/((a+b)/2 - a*b/t)
return jaccard, rand, ari

def lerClassesExistentes(self):
    classesImagem = {}
    self.classesGabarito = []

    for key1 in self.mapaImagemOrdenado.keys():
        if key1 not in classesImagem.keys():
            classesImagem[key1] =
sum(self.mapaImagemOrdenado[key1].values())
        for key2 in self.mapaImagemOrdenado[key1].keys():
            if key2 not in self.classesGabarito:
                self.classesGabarito.append(key2)

    self.classesImagem = {k: v for k, v in
sorted(classesImagem.items(), key=lambda item: item[1],
reverse=True)}

def acharClassesCorrelatas(self):
    mapaImagem = {}
    pixelsClasse = {}
    gab = self.gabarito
    img = self.imagem
    for i in range(img.shape[1]):
        for j in range(img.shape[0]):
            if str(img[i,j]) in pixelsClasse.keys():
                pixelsClasse[str(img[i,j])] += 1
            else:
                pixelsClasse[str(img[i,j])] = 1
            if str(img[i,j]) in mapaImagem.keys():
                if str(gab[i,j]) in mapaImagem[str(img[i,j])].keys():
                    mapaImagem[str(img[i,j])][str(gab[i,j])] += 1
                else:
                    mapaImagem[str(img[i,j])][str(gab[i,j])] = 1
            else:
                mapaImagem[str(img[i,j])] = {}
                mapaImagem[str(img[i,j])][str(gab[i,j])] = 1

    self.mapaImagemOrdenado = {}
    self.resultado_abs = {}
    self.resultado_rel = {}
    for key in mapaImagem.keys():
        self.mapaImagemOrdenado[key] = {k: v for k, v in
sorted(mapaImagem[key].items(), key=lambda item: item[1],
reverse=True)}
        self.resultado_abs[key] =
next(iter(self.mapaImagemOrdenado[key].items()))

```

```

        self.resultado_rel[key] =
(self.resultado_abs[key][0],self.resultado_abs[key][1]/pixelsCla
sse[key])

    self.lerClassesExistentes()

    return self.resultado_abs,self.resultado_rel

def agruparClasses(self):
    classesAgrupadas = {}
    for classeG in self.classesGabarito:
        classesAgrupadas[classeG] = []
        for classeI in self.classesImagem.keys():
            if classeG == self.resultado_rel[classeI][0]:
                classesAgrupadas[classeG].append(classeI)

    print(classesAgrupadas)

    return classesAgrupadas

def avaliar(result, label):
    avaliador = AvaliaClassificacao(result, label)
    avaliador.acharClassesCorrelatas()
    avaliador.agruparClasses()
    return avaliador.calculaIndices()

def main(id):
    k_value = 10 #len(np.unique(full_label))
    result = run_Kmeans(samples[id], k_value)
    print(f'Finished classifying image {id}')
    jaccard, rand, ari = avaliar(result, labels[id])
    print(f'Finished evaluating image {id}')
    results.append(result)
    jaccards.append(jaccard)
    rands.append(rand)
    aris.append(ari)

processes = []
results = []
jaccards = []
rands = []
aris = []
count1 = 1

for image, label in image_paths.items():
    print(f'Image {count1}')
    count2 = 1
    full_image = cv2.imread(image)
    full_label = cv2.imread(label)
    samples = slice_image(full_image)
    labels = slice_image(full_label)
    for n in range(len(samples)):

```

```

    print(f'Image {count1}: sample {count2} of {len(samples)}')
    process = threading.Thread(target=main, args=(n,))
    processes.append(process)
    process.start()
    count2 += 1

count1 += 1

for index, p in enumerate(processes):
    print(index)
    while p.is_alive():
        time.sleep(0.1)

print('\n\n\n\n\n')
print('Média Jaccard: {}'.format(sum(jaccards)/len(jaccards)))
print('Má;x Jaccard: imagem {} ->
{}'.format(jaccards.index(max(jaccards)),max(jaccards)))
print('Min Jaccard: imagem {} ->
{}'.format(jaccards.index(min(jaccards)),min(jaccards)))
print(jaccards)
print('Média Rand: {}'.format(sum(rands)/len(rands)))
print('Má;x Rand: imagem {} ->
{}'.format(rands.index(max(rands)),max(rands)))
print('Min Rand: imagem {} ->
{}'.format(rands.index(min(rands)),min(rands)))
print(rands)
print('Média Ari: {}'.format(sum(aris)/len(aris)))
print('Má;x Ari: imagem {} ->
{}'.format(aris.index(max(aris)),max(aris)))
print('Min Ari: imagem {} ->
{}'.format(aris.index(min(aris)),min(aris)))
print(aris)

```

APÊNDICE 3

Resultado, por Ortofoto, do método K-médias, aplicado ao cubo de imagem RGB original, para totalidade das imagens, para os Índice de Jaccard, Índice de Rand, Índice de Rand Ajustado.

Quadro 9 - resultados dos índices para o método K-médias

RESULTADOS – METODO K-MÉDIAS			
Ortofoto	Jaccard (K-médias)	Rand (K-médias)	Rand Ajustado (K-médias)
Ortofoto - 2-10	0,1670	0,5987	0,2851
Ortofoto - 2-11	0,1602	0,5990	0,2751
Ortofoto - 2-12	0,1633	0,5327	0,2799
Ortofoto - 2-13	0,1465	0,6276	0,2553
Ortofoto - 2-14	0,1512	0,5715	0,2619
Ortofoto - 3-10	0,1799	0,6797	0,3022
Ortofoto - 3-11	0,1933	0,6555	0,3204
Ortofoto - 3-12	0,1677	0,6706	0,2857
Ortofoto - 3-13	0,1523	0,6787	0,2634
Ortofoto - 3-14	0,1576	0,6660	0,2714
Ortofoto - 4-10	0,1567	0,6608	0,2700
Ortofoto - 4-11	0,1632	0,6830	0,2797
Ortofoto - 4-12	0,0517	0,8634	0,0969
Ortofoto - 4-13	0,1698	0,6755	0,2888
Ortofoto - 4-14	0,1698	0,6500	0,2891
Ortofoto - 4-15	0,1613	0,6552	0,2768
Ortofoto - 5-10	0,1741	0,6487	0,2944
Ortofoto - 5-11	0,1584	0,6385	0,2727
Ortofoto - 5-12	0,1586	0,5938	0,2731
Ortofoto - 5-13	0,1767	0,6337	0,2987
Ortofoto - 5-14	0,1751	0,6590	0,2967
Ortofoto - 5-15	0,1700	0,6414	0,2893
Ortofoto - 6-7	0,1523	0,5044	0,2639
Ortofoto - 6-8	0,1617	0,6592	0,2772
Ortofoto - 6-9	0,1557	0,6485	0,2688
Ortofoto - 6-10	0,1768	0,6015	0,2990
Ortofoto - 6-11	0,1776	0,5991	0,3001
Ortofoto - 6-12	0,1793	0,6122	0,3019
Ortofoto - 6-13	0,1899	0,6153	0,3158
Ortofoto - 6-14	0,1989	0,6407	0,3299
Ortofoto - 6-15	0,1915	0,6728	0,3195
Ortofoto - 7-7	0,1873	0,6848	0,3143
Ortofoto - 7-8	0,1578	0,6639	0,2719
Ortofoto - 7-9	0,1741	0,6180	0,2952
Ortofoto - 7-10	0,1966	0,5709	0,3243
Ortofoto - 7-11	0,2050	0,6176	0,3383
Ortofoto - 7-12	0,1728	0,5601	0,2933

Ortofoto - 7-13	0,1825	0,5462	0,3071
Média	0,1680	0,6342	0,2855
Desvio padrão	0,0239	0,0580	0,0370
Variância	0,0006	0,0034	0,0014

APÊNDICE 4

Resultado, por Ortofoto, do método *co-clustering*, aplicado ao cubo de imagem RGB integrado, para totalidade das imagens, para os Índice de Jaccard, Índice de Rand, Índice de Rand Ajustado.

Quadro 10 – resultados dos índices para o método co-clustering sem refinamento

RESULTADOS – METODO CO-CLUSTERING SEM REFINAMENTO			
Ortofoto	Jaccard (Co-clustering)	Rand (Co-clustering)	Rand Ajustado (Co-clustering)
Ortofoto - 2-10	0,1669	0,6045	0,2850
Ortofoto - 2-11	0,1712	0,6115	0,2905
Ortofoto - 2-12	0,1671	0,5417	0,2836
Ortofoto - 2-13	0,1554	0,6438	0,2684
Ortofoto - 2-14	0,1608	0,5866	0,2755
Ortofoto - 3-10	0,1874	0,6928	0,3126
Ortofoto - 3-11	0,1909	0,6599	0,3179
Ortofoto - 3-12	0,1686	0,6829	0,2867
Ortofoto - 3-13	0,1610	0,6953	0,2765
Ortofoto - 3-14	0,1617	0,6796	0,2772
Ortofoto - 4-10	0,1594	0,6749	0,2733
Ortofoto - 4-11	0,1696	0,6972	0,2891
Ortofoto - 4-12	0,0438	0,8669	0,0830
Ortofoto - 4-13	0,1704	0,6828	0,2899
Ortofoto - 4-14	0,1714	0,6613	0,2910
Ortofoto - 4-15	0,1766	0,6705	0,2988
Ortofoto - 5-10	0,1774	0,6579	0,2990
Ortofoto - 5-11	0,1629	0,6471	0,2793
Ortofoto - 5-12	0,1628	0,6019	0,2792
Ortofoto - 5-13	0,1676	0,6357	0,2857
Ortofoto - 5-14	0,1893	0,6737	0,3160
Ortofoto - 5-15	0,1793	0,6536	0,3031
Ortofoto - 6-7	0,1540	0,5131	0,2660
Ortofoto - 6-8	0,1850	0,6771	0,3095
Ortofoto - 6-9	0,1720	0,6634	0,2919
Ortofoto - 6-10	0,1752	0,6074	0,2971
Ortofoto - 6-11	0,1649	0,5956	0,2820
Ortofoto - 6-12	0,1776	0,6161	0,3000
Ortofoto - 6-13	0,2049	0,6297	0,3354
Ortofoto - 6-14	0,2050	0,6484	0,3378
Ortofoto - 6-15	0,2001	0,6789	0,3313
Ortofoto - 7-7	0,1852	0,6979	0,3115
Ortofoto - 7-8	0,1637	0,6787	0,2804
Ortofoto - 7-9	0,1908	0,6345	0,3186
Ortofoto - 7-10	0,2082	0,5837	0,3385
Ortofoto - 7-11	0,1961	0,6190	0,3260
Ortofoto - 7-12	0,1728	0,5634	0,2929

Ortofoto - 7-13	0,1900	0,5548	0,3162
Média	0,1728	0,6443	0,2920
Desvio padrão	0,0256	0,0587	0,0396
Variância	0,0007	0,0034	0,0016

APÊNDICE 5

Resultado, por Ortofoto, do método *co-clustering*, com refinamento pelo método K-médias, aplicado ao cubo de imagem integrado, para totalidade das imagens, para os Índice de Jaccard, Índice de Rand, Índice de Rand Ajustado.

Quadro 11 – resultados dos índices para o método co-clustering com refinamento

RESULTADOS – METODO CO-CLUSTERING COM REFINAMENTO			
Ortofoto	Jaccard (Co-clustering refinado)	Rand (Co-clustering refinado)	Rand Ajustado (Co-clustering refinado)
Ortofoto - 2-10	0,4406	0,5368	0,5994
Ortofoto - 2-11	0,4501	0,5524	0,6084
Ortofoto - 2-12	0,5169	0,5591	0,6548
Ortofoto - 2-13	0,3762	0,5789	0,5281
Ortofoto - 2-14	0,4505	0,5593	0,6070
Ortofoto - 3-10	0,3425	0,4820	0,5041
Ortofoto - 3-11	0,3925	0,5088	0,5550
Ortofoto - 3-12	0,3363	0,5192	0,4980
Ortofoto - 3-13	0,3100	0,5152	0,4697
Ortofoto - 3-14	0,3397	0,5344	0,5344
Ortofoto - 4-10	0,3423	0,5472	0,4999
Ortofoto - 4-11	0,3479	0,5828	0,5119
Ortofoto - 4-12	0,0167	0,4479	0,0325
Ortofoto - 4-13	0,3453	0,5095	0,5067
Ortofoto - 4-14	0,3621	0,5016	0,5245
Ortofoto - 4-15	0,3883	0,5439	0,5514
Ortofoto - 5-10	0,3899	0,5388	0,5543
Ortofoto - 5-11	0,3835	0,5276	0,5458
Ortofoto - 5-12	0,4404	0,5748	0,5989
Ortofoto - 5-13	0,4036	0,5190	0,5682
Ortofoto - 5-14	0,4059	0,5789	0,5673
Ortofoto - 5-15	0,3981	0,5536	0,5584
Ortofoto - 6-7	0,5407	0,6077	0,6852
Ortofoto - 6-8	0,3788	0,5363	0,5431
Ortofoto - 6-9	0,3744	0,5479	0,5362
Ortofoto - 6-10	0,4574	0,6071	0,6118
Ortofoto - 6-11	0,4359	0,5504	0,5999
Ortofoto - 6-12	0,4492	0,5957	0,6118
Ortofoto - 6-13	0,4660	0,5921	0,6247
Ortofoto - 6-14	0,4596	0,6079	0,6180
Ortofoto - 6-15	0,3935	0,5703	0,5581
Ortofoto - 7-7	0,4088	0,6353	0,5734
Ortofoto - 7-8	0,3535	0,5769	0,5172
Ortofoto - 7-9	0,4562	0,6120	0,6115
Ortofoto - 7-10	0,5233	0,6295	0,6610
Ortofoto - 7-11	0,4734	0,5689	0,6340

Ortofoto - 7-12	0,5006	0,6145	0,6562
Ortofoto - 7-13	0,5566	0,6547	0,6989
Média	0,4055	0,5600	0,5610
Desvio padrão	0,0881	0,0438	0,1029
Variância	0,0078	0,0019	0,0106

APÊNDICE 6

Resultado, por Ortofoto, do método *co-clustering*, a partir do cubo de imagem RGB original, para totalidade das imagens, para os Índice de Jaccard, Índice de Rand, Índice de Rand Ajustado.

Quadro 12 – resultados dos índices para o método co-clustering sem refinamento

RESULTADOS – METODO CO-CLUSTERING SEM REFINAMENTO			
Ortofoto	Jaccard (Co-clustering)	Rand (Co-clustering)	Rand Ajustado (Co-clustering)
Ortofoto - 2-10	0,1650	0,6010	0,2821
Ortofoto - 2-11	0,1606	0,6051	0,2756
Ortofoto - 2-12	0,1621	0,5390	0,2779
Ortofoto - 2-13	0,1434	0,6364	0,2504
Ortofoto - 2-14	0,1486	0,5756	0,2580
Ortofoto - 3-10	0,1772	0,6860	0,2985
Ortofoto - 3-11	0,1969	0,6632	0,3253
Ortofoto - 3-12	0,1645	0,6783	0,2807
Ortofoto - 3-13	0,1490	0,6868	0,2587
Ortofoto - 3-14	0,1574	0,6736	0,2709
Ortofoto - 4-10	0,1539	0,6679	0,2655
Ortofoto - 4-11	0,1579	0,6900	0,2721
Ortofoto - 4-12	0,0538	0,8720	0,1008
Ortofoto - 4-13	0,1664	0,6831	0,2841
Ortofoto - 4-14	0,1652	0,6546	0,2822
Ortofoto - 4-15	0,1611	0,6613	0,2766
Ortofoto - 5-10	0,1725	0,6540	0,2917
Ortofoto - 5-11	0,1566	0,6435	0,2702
Ortofoto - 5-12	0,1584	0,5985	0,2727
Ortofoto - 5-13	0,1737	0,6384	0,2942
Ortofoto - 5-14	0,1749	0,6637	0,2961
Ortofoto - 5-15	0,1687	0,6444	0,2976
Ortofoto - 6-7	0,1468	0,5050	0,2555
Ortofoto - 6-8	0,1603	0,6629	0,2746
Ortofoto - 6-9	0,1569	0,6544	0,2706
Ortofoto - 6-10	0,1748	0,6050	0,2960
Ortofoto - 6-11	0,1766	0,6039	0,2977
Ortofoto - 6-12	0,1786	0,6166	0,3007
Ortofoto - 6-13	0,1928	0,6210	0,3203
Ortofoto - 6-14	0,1965	0,6460	0,3265
Ortofoto - 6-15	0,1959	0,6783	0,3243
Ortofoto - 7-7	0,1790	0,6911	0,3025
Ortofoto - 7-8	0,1500	0,6740	0,2603
Ortofoto - 7-9	0,1716	0,6218	0,2918
Ortofoto - 7-10	0,1860	0,5672	0,3109
Ortofoto - 7-11	0,2020	0,6214	0,3345
Ortofoto - 7-12	0,1741	0,5632	0,2947

Ortofoto - 7-13	0,1829	0,5504	0,3071
Média	0,1661	0,6394	0,2829
Desvio padrão	0,0237	0,0594	0,0366
Variância	0,0006	0,0035	0,0013

APÊNDICE 7

Resultado, por Ortofoto, do método *co-clustering*, com refinamento pelo método K-médias, a partir do cubo de imagem RGB original, para totalidade das imagens, para os Índice de Jaccard, Índice de Rand, Índice de Rand Ajustado.

Quadro 13 – resultados dos índices para o método co-clustering com refinamento

RESULTADOS – METODO CO-CLUSTERING COM REFINAMENTO			
Ortofoto	Jaccard (Co-clustering refinado)	Rand (Co-clustering refinado)	Rand Ajustado (Co-clustering refinado)
Ortofoto - 2-10	0,4421	0,5761	0,6003
Ortofoto - 2-11	0,4463	0,5401	0,6044
Ortofoto - 2-12	0,5297	0,6041	0,6668
Ortofoto - 2-13	0,3644	0,4735	0,5151
Ortofoto - 2-14	0,4523	0,5465	0,6079
Ortofoto - 3-10	0,3624	0,5114	0,5227
Ortofoto - 3-11	0,4121	0,5574	0,5725
Ortofoto - 3-12	0,3211	0,4273	0,4815
Ortofoto - 3-13	0,3019	0,4344	0,4601
Ortofoto - 3-14	0,3213	0,4373	0,4811
Ortofoto - 4-10	0,3345	0,4708	0,4930
Ortofoto - 4-11	0,3192	0,4444	0,4811
Ortofoto - 4-12	0,0149	0,4058	0,0291
Ortofoto - 4-13	0,3219	0,4304	0,4821
Ortofoto - 4-14	0,3670	0,4813	0,5301
Ortofoto - 4-15	0,3485	0,4778	0,5116
Ortofoto - 5-10	0,3792	0,5019	0,5426
Ortofoto - 5-11	0,3569	0,4566	0,5192
Ortofoto - 5-12	0,4116	0,4741	0,5722
Ortofoto - 5-13	0,3974	0,4894	0,5611
Ortofoto - 5-14	0,3733	0,4897	0,5349
Ortofoto - 5-15	0,3791	0,4783	0,5399
Ortofoto - 6-7	0,5417	0,6064	0,6855
Ortofoto - 6-8	0,3675	0,5048	0,5305
Ortofoto - 6-9	0,3638	0,5064	0,5245
Ortofoto - 6-10	0,4365	0,5262	0,5904
Ortofoto - 6-11	0,4282	0,5223	0,5930
Ortofoto - 6-12	0,4220	0,5245	0,5835
Ortofoto - 6-13	0,4453	0,5505	0,6057
Ortofoto - 6-14	0,3993	0,4871	0,5602
Ortofoto - 6-15	0,3887	0,5323	0,5527
Ortofoto - 7-7	0,3501	0,5015	0,5138
Ortofoto - 7-8	0,3191	0,4205	0,4805
Ortofoto - 7-9	0,4084	0,4978	0,5658
Ortofoto - 7-10	0,5097	0,5640	0,6455
Ortofoto - 7-11	0,4492	0,5513	0,6080

Ortofoto - 7-12	0,4881	0,5574	0,6436
Ortofoto - 7-13	0,5250	0,5757	0,6725
Média	0,3895	0,5036	0,5438
Desvio padrão	0,0875	0,0509	0,1024
Variância	0,0077	0,0026	0,0105