Int J Appl Earth Obs Geoinformation 76 (2019) 218–230

Contents lists available at ScienceDirect

Int J Appl Earth Obs Geoinformation

journal homepage: www.elsevier.com/locate/jag

SegOptim—A new R package for optimizing object-based image analyses of high-spatial resolution remotely-sensed data

João Gonçalves[a,b,*], Isabel Pôças[c,d], Bruno Marcos[a,b], C.A. Mücher[e], João P. Honrado[a,b]

[a] *CIBIO – Research Center in Biodiversity and Genetic Resources, InBIO – Research Network in Biodiversity and Evolutionary Biology, Associate Laboratory, Campus Agrário de Vairão, Universidade do Porto, 4485-601 Vairão, Portugal*

[b] *Faculty of Sciences, University of Porto (FCUP), Edifício FC4 (Biologia), Rua do Campo Alegre, s/n, 4169-007 Porto, Portugal*

[c] *Linking Landscape, Environment, Agriculture and Food, Instituto Superior de Agronomia, Universidade de Lisboa, Lisboa 1349-017, Portugal*

[d] *Geo-Space Sciences Research Centre, Universidade do Porto, Porto 4169-007, Portugal*

[e] *Wageningen Environmental Research (Alterra), Earth Informatics subdivision, Wageningen University, Droevendaalsesteeg 3 6708PB Wageningen, the Netherlands*

A R T I C L E I N F O

Keywords:
Geographic object-based image analysis
GEOBIA
Image segmentation
Supervised classification
Genetic algorithms
Optimization
High-spatial resolution
Open-source software
R package

A B S T R A C T

Geographic Object-based Image Analysis (GEOBIA) is increasingly used to process high-spatial resolution imagery, with applications ranging from single species detection to habitat and land cover mapping. Image segmentation plays a key role in GEOBIA workflows, allowing to partition images into homogenous and mutually exclusive regions. Nonetheless, segmentation techniques require a robust parameterization to achieve the best results. Frequently, inappropriate parameterization leads to sub-optimal results and difficulties in comparing distinct methods.

Here, we present an approach based on Genetic Algorithms (GA) to optimize image segmentation parameters by using the performance scores from object-based classification, thus allowing to assess the adequacy of a segmented image in relation to the classification problem. This approach was implemented in a new R package called *SegOptim*, in which several segmentation algorithms are interfaced, mostly from open-source software (*GRASS GIS, Orfeo Toolbox, RSGISLib, SAGA GIS, TerraLib*), but also from proprietary software (*ESRI ArcGIS*). *SegOptim* also provides access to several machine-learning classification algorithms currently available in R, including Gradient Boosted Modelling, Support Vector Machines, and Random Forest.

We tested our approach using very-high to high spatial resolution images collected from an Unmanned Aerial Vehicle (0.03 – 0.10 m), WorldView-2 (2 m), RapidEye (5 m) and Sentinel-2 (10 – 20 m) in six different test sites located in northern Portugal with varying environmental conditions and for different purposes, including invasive species detection and land cover mapping. The results highlight the added value of our novel comparison of image segmentation and classification algorithms. Overall classification performances (assessed through cross-validation with the Kappa index) ranged from 0.85 to 1.00. Pilot-tests show that our GA-based approach is capable of providing sound results for optimizing the parameters of different segmentation algorithms, with benefits for classification accuracy and for comparison across techniques. We also verified that no particular combination of an image segmentation and a classification algorithm is suited for all the tasks/objectives. Consequently, it is crucial to compare and optimize available methods to understand which one is more suited for a certain objective.

Our approach allows a closer integration between the segmentation and classification stages, which is of high importance for GEOBIA workflows. The results from our tests confirm that this integration has benefits for comparing and optimizing both processes. We discuss some limitations of the *SegOptim* approach (and potential solutions) as well as a future roadmap to expand its current functionalities.

1. Introduction

Geographic Object-based Image Analysis (GEOBIA) is a recent sub-discipline of Geographic Information Science that, according to Hay and Castilla (2008), is "(…) *devoted to developing automated methods to partition remote sensing imagery into meaningful image-objects, and*

* Corresponding author at: CIBIO – Research Center in Biodiversity and Genetic Resources, InBIO – Research Network in Biodiversity and Evolutionary Biology, Associate Laboratory, Campus Agrário de Vairão, Universidade do Porto, 4485-601 Vairão, Portugal.
E-mail address: joaofgo@gmail.com (J. Gonçalves).

https://doi.org/10.1016/j.jag.2018.11.011
Received 13 October 2017; Received in revised form 24 November 2018; Accepted 27 November 2018
Available online 13 December 2018
0303-2434/

assessing their characteristics through spatial, spectral and temporal scales, so as to generate new geographic information (…)". More specifically, object-based analysis of Earth Observation (EO) data allows to describe the imaged reality using spectral, textural, spatial, contextual and topological features in a multi-scalar and integrated fashion (Lang, 2008). GEOBIA benefitted from the availability of (very-)high spatial resolution imagery, from vast progresses in image segmentation as well as software bridging image processing and GIS functionalities in an object-based environment ready for exploration and analysis (Blaschke, 2010).

GEOBIA concepts and methods have been used in several contexts and reported in a large and growing body of literature. Image segmentation plays an important role in GEOBIA (Blaschke, 2010; Lang, 2008). Independently of the method used, segmentation provides the elementary blocks used for object-based image analysis (Blaschke, 2010), interpretation, classification and modelling. Segmentation involves the partitioning of an image into a set of jointly exhaustive and mutually disjoint regions (a.k.a. segments, composed by multiple image pixels), that are internally more homogeneous and similar, compared to adjacent ones. Image segments are then related to geographic objects of interest (e.g., forests, agricultural or urban areas) through some form of object-based classification (Castilla and Hay, 2008). Image segments are created on the basis of one or more homogeneity (and merging) criteria in one or more dimensions of a feature space (Blaschke, 2010). As such, these segments have additional information related to the properties and moments of the distribution of spectral data from each individual contained pixel as well as contextual, morphological and spatial information (Blaschke, 2010; Hay and Castilla, 2008; van der Werff and van der Meer, 2008). Image segments should be meaningful in respect to a particular task and their properties should allow to convert them into useful geographic objects. This is especially challenging because widely different segmentation results can be obtained by varying parameter values of existing algorithms (Dragut et al., 2014; Liu et al., 2012). Therefore, defining objective criteria is needed to address this problem, allowing to identify which image segmentation algorithms and parameterization may provide optimal solutions in each case.

In order to better understand how image segmentation parameters are typically defined by users, we performed a semi-systematic, sample-based review of the current literature (see details in Supporting Information – Appendix S1). We found that 44% of 72 randomly selected papers (out of 1067 retrieved in our search) did not explicitly mentioned how image segmentation parameters were defined or tuned (Supp. Info. Appendix S1/Figure S1–1a). From publications that did mentioned this, 28% corresponded to specific procedures, 19% used a visual interpretation/trial-and-error approach and 9% used the ESP tool (Dragut et al., 2014) available only for *eCognition* multi-resolution segmentation (Baatz and Schape, 2000). Only very few examples (*ca.* 7%) explicitly integrated both image segmentation and classification steps for tuning parameters.

Although several works did not state or used trial-and-error approaches for defining image segmentation parameters, there are rigorous alternatives to perform this task. For example, ESP is an automatic tool for *eCogniton*® proprietary software capable of determining a set of suitable 'scale' parameters for multi-resolution image segmentation (Dragut et al., 2014). This unsupervised approach uses changes in local variance to detect scale transitions in spatial data. Other works have employed a supervised approach to this problem, comparing a set of reference objects with those obtained by image segmentation. Clinton et al. (2010) proposed several discrepancy measures to assess segmentation quality between sets based on area, location, or the combination of both aspects. Based on this work, Liu et al. (2012) devised a supervised discrepancy measure named *Euclidean Distance 2* (ED2) to evaluate segmentation quality based on geometric and arithmetic similarity and used it to define the best parameters for segmentation (Liu et al., 2012; Novelli et al., 2017). Räsänen et al. (2013) compared several discrepancy measures for mapping boreal forests and concluded that it is crucial to state the objectives of image segmentation and that discrepancy evaluation measures should be used with care. Moreover, some of the above methods are software or algorithm-specific (such as ESP tool) or focus on geometric, positional or areal similarity between segments and reference data and thus, are not integrated with classification procedures.

In order to overcome some of these limitations and to generalize the process of comparing different segmentation algorithms and optimizing their parameters, we developed a solution that integrates both processing steps (segmentation and supervised classification) in a single and unified workflow. Genetic Algorithms (GA) are then used to optimize image segmentation parameter values in relation to the classification problem. GA's are broadly included in the class of evolutionary algorithms and constitute a type of computational search method used to find exact and approximate solutions to a given optimization problem (Scrucca, 2013). GA's are employed here due to their flexible and multi-purpose nature as a stochastic search technique capable of solving optimization problems both for continuous and discrete functions, by mimicking the biological principles of evolution and natural selection (Haupt and Haupt, 2004; Scrucca, 2013).

Our approach, based on GA optimization, was implemented in a new open-source R package named *SegOptim* which can be used to tune image segmentation parameters in the context of supervised classification of EO images. For assessing the proposed approach and toolkit we devised a set of tests using high (10–20 m), very-high (2–5 m) and 'ultra'-high (0.03-0.10 m) spatial resolution images, collected from distinct EO platforms in six test sites in northern Portugal with different environmental and landscape conditions, and in which image segmentation was used for different purposes. *SegOptim* package is directed towards users interested in applying, comparing and optimizing object-based algorithms to image segmentation and classification of EO data. We provide access to the package source code through a *Bitbucket* repository: https://bitbucket.org/joao_goncalves/segoptim, and a tutorial describing the package functionalities: https://segoptim.bitbucket.io/docs/ (*in prep*).

2. Methods

2.1. The proposed optimization approach

2.1.1. Implementing the optimization approach in R

The R environment and language for statistical computing (R Development Core Team, 2017) provided the computational environment to implement *SegOptim* – the R package implementing the optimized GA-based approach to multi-technique image segmentation and GEOBIA supervised classification.

R has been gaining more popularity among the Remote Sensing (RS)/EO and GIS communities due to its ability to handle raster datasets (*raster* package (Hijmans, 2016)) and (pre-)process EO data (packages such as *RStoolbox* (Leutner and Horning, 2017), *landsat* (Goslee, 2011) or *satellite* (Nauss et al., 2015)).

However, currently there is no R package providing image segmentation functionalities specifically for EO data. That is partly due to the inherent complexity of these algorithms and their implementations as well as constraints imposed by the type, formats and size of the data used. One way to circumvent this limitation, and the strategy adopted here, is to use 3rd party software to perform the image segmentation stage (Fig. 1). The segmented image, outputted by an algorithm, is then loaded into R for further analysis. R is especially well tailored for running image classification since it has a large number of available algorithms for this purpose.

Using this architecture, R acts as an interface between 3rd party programs (used for image segmentation), centralizing all data required for performing classification and optimization.

SegOptim is capable of handling both *single-class* problems (defined as a type of supervised classification problem with only one class with

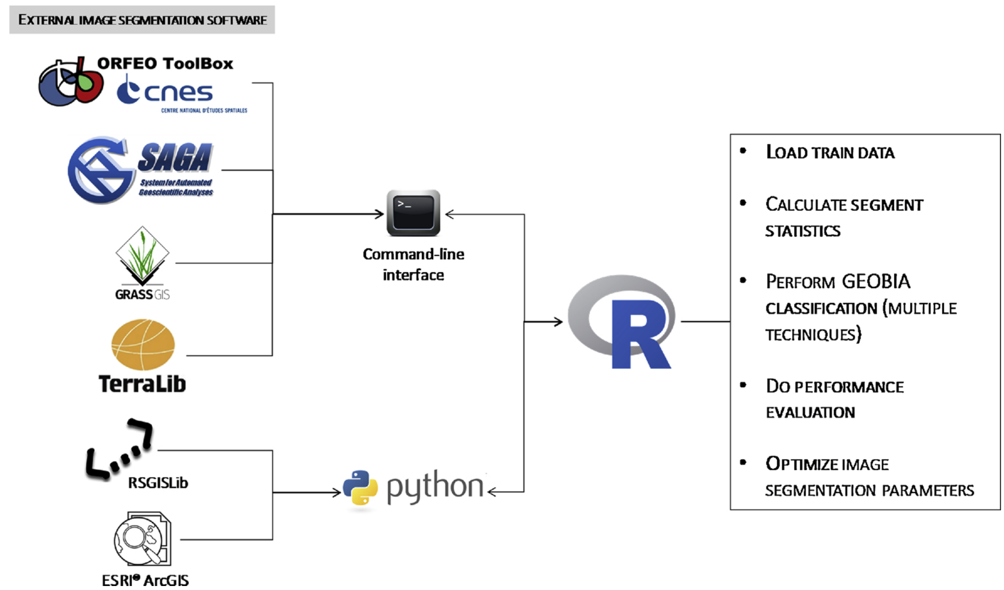

Fig. 1. The architecture of the software package *SegOptim*.

two mutually exclusive levels, e.g., species presence / absence, burned / unburned, cut / uncut forest) as well as *multi-class* (defined as a type of supervised classification problem with multiple, n ≥ 2, classes).

2.1.2. Genetic Algorithms: fitness function definition

In GA computation, at a certain stage of evolution, a population is composed by a number of individuals each representing a possible solution to the problem. Individuals are composed by genes which control the inheritance and passing of information between individuals. The capacity of individuals to solve a certain problem is determined by the *fitness function* and only the fittest individuals reproduce and pass their genetic information to their offspring. At each generation, the GA evolutionary strategy discards the less fit individuals (or solutions) and new ones are generated by the reproduction of the fittest (Scrucca, 2013). This is accomplished by several genetic operators in two stages: *exploration* and *exploitation*. The first aims at exploring the search space and is performed by operators such as *crossover* (which forms new offspring by combining parts of the genetic information from two parent individuals) and *mutation* (which randomly alters the values of genes in parents). Complementarily, exploitation, aims to reduce the diversity in the populations and focus on higher quality solutions. For this matter a *selection* strategy is employed thus allowing the fittest individuals to persist (Rothlauf, 2011). The GA evolution process is stopped if some convergence criteria is attained.

GA's require the definition of a *fitness function* (also defined as objective or cost function) which provides a single score ranking the "merit" of each solution and summarising how close it is to the established set of goals. In *SegOptim*, the goal of the fitness function is to improve/optimize image segmentation in order to maximize the thematic accuracy of the supervised classification based on a predefined set of image inputs such as labelled train input data, a set of classification features and finally, control parameters. To this purpose, GA is responsible for iteratively optimizing the segmentation algorithm arguments in order to improve classification accuracy (Box 1 and Fig. 2).

For evaluating the merit of each solution, *SegOptim* uses either 5-fold, 10-fold or holdout cross-validation. Depending if the problem is single- or multi-class, it is possible to calculate several classification performance metrics that summarise the confusion matrix. For single-class problems, the Area Under the Receiver Operating Curve (AUC), Cohen Kappa, Peirce Skill score (or true-skill score; PSS) and the Gerrity skill score (GSS) are available, while for multi-class, accuracy, Kappa, PSS and GSS can be used. The user can also define his own fitness/performance function, calculating any meaningful score from the confusion matrix. For testing purposes, we selected Kappa as the fitness value as it is a widely used metric that can be applied for both single- and multi-class classification and accounts for the accuracy simply obtained by chance (Kuhn and Johnson, 2013).

2.1.3. Genetic Algorithm parameterization and setup

Running GA requires the setup of several parameters responsible for controlling the initial population and the exploration/exploitation stages (Table 1).

Given the time required to generate a single solution, the parameters controlling GA were setup in a 'conservative fashion' meaning that population size was relatively low (but enough to generate diverse solutions in the exploration stage). This parameterization was suited to find an approximate solution with a relatively small population size and a relatively low number of iterations thus allowing to decrease computation time and keep processes tractable. Parameters were set equal across all tests to make comparisons meaningful and avoid differences generated by different GA settings.

GA also requires the search space to be constrained and therefore the user must define the minimum/maximum values of each image segmentation parameter for each algorithm. This is critical to achieve good results as well as to avoid parameter combinations that provide less fit solutions and thus increase computing time (see details in Supp. Info. – Appendix S2).

2.1.4. Image segmentation algorithms available in the package

SegOptim currently interfaces with six external state-of-the-art programs responsible for performing the image segmentation stage (Table 2). From these, a total of seven image segmentation algorithms are available. Four types of algorithms can be identified (based on similar characteristics) albeit differences in their computational implementations: (i) mean-shift (*ESRI ArcGIS* and *Orfeo Toolbox*) (Comaniciu and Meer, 2002; Michel et al., 2015); (ii) region growing and merging (*GRASS GIS*, *SAGA GIS* and *TerraLib*) (Adams and Bischof, 1994; Bechtel et al., 2008; Hojjatoleslami and Kittler, 1998; Tremeau and Borel, 1997); (iii) Shepherd k-means iterative elimination (*RSGISLib*) (Clewley et al., 2014); and, (iv) Multi-resolution image segmentation (also in *TerraLib*) (Baatz and Schape, 2000).

2.1.5. Supervised classification algorithms

Currently, *SegOptim* package offers several machine-learning

Box 1
Pseudo-code describing the workflow for the proposed GA fitness function.

1. Run the image segmentation;
2. Load training data into the segmented image;
3. Calculate segment statistics (e.g., mean, standard-deviation) for classification features;
4. Merge training and segment statistics data (steps 2-3);
5. Do train/test data partitions (e.g., 5-fold cross-validation);
6. For each train/test set:
    6.1. Train the supervised classifier;
    6.2. Evaluate results for the test set;
7. Return the average evaluation score across all train/test rounds (fitness value);

classification algorithms which can be used to perform GEOBIA classification stage, these are: (i) Gradient Boosted Regression and Classification (GBM) available through *gbm* package (Ridgeway, 2017); (ii) Random Forests (RF) accessible in R through *randomForest* package (Liaw and Wiener, 2002); and (iii) Support Vector Machines (SVM) available via *e1071* package (Meyer et al., 2017). These techniques (along with available segmentation methods) were used and compared through a series of tests for several distinct sites and image classification problems. The default parameterization for each method was used.

2.1.6. Input data

Briefly described, the *SegOptim* package relies on three basic data inputs used to perform image segmentation and object-based classification (Fig. 2). These are:

- Training data: a single-layer raster dataset containing user-labelled samples for training a classifier;
- Segmentation features: a multi-layer raster dataset with features used only for the segmentation stage (e.g., spectral bands);
- Classification features: a multi-layer raster dataset with features used for classification (e.g., spectral data, band ratios, spectral vegetation indices, texture features, topographic data).

2.2. Testing the optimized approach

2.2.1. Input training data

Six test sites located in northern Portugal were used to test *SegOptim* optimization approach, comprising three single-class and three multi-class problems. These tests represent a diverse array of ecological applications, scales and domains (e.g., land use/cover mapping, habitat conservation, invasive species) as well as of environmental conditions, ranging from coastal, lowlands up to mountainous areas and different levels of landscape heterogeneity (Table 3).

The first set (single-class tests; Table 3) targeted the detection of invasive species associated to sand dune environments (*Carpobrotus* spp.; E1) or forest areas (*Acacia* spp.; E3 and E5). In the second set (multi-class tests; Table 3), we used *SegOptim* to improve segmentation parametrization for producing maps of natural habitat types (E2; Gonçalves et al. (2016)) or land use/cover maps (E4 and E6; the target classes are listed in Supp. Info. – Appendix S3).

Regarding the collection and/or digitization of training data, for test-sites E1 (sample size, n = 80) and E2 (n = 25) we followed a systematic sampling approach using a regular sampling grid to select sample locations later used in the field to collect ground-truth data. In test site E3 we used purposive sampling (n = 40) to locate the target species in the field, complemented with locations digitized from *GoogleEarth* imagery (Monteiro et al., 2017). In test site E4, we used a systematic sampling approach (n = 271) for digitizing training locations based on *GoogleEarth* imagery. For test site E5 we followed a purposive sampling scheme for locating patches of *Acacia dealbata (n = 108)* complemented with a set of random samples for the species absence digitized from *GoogleEarth* imagery. In test site E6 we used a stratified random sampling approach with strata defined through clustering of several layers related to topography, soils and other datasets (Bastos et al., 2016) for collecting training samples (n = 1545) based on ground-truth data, ancillary land cover data and *GoogleEarth*

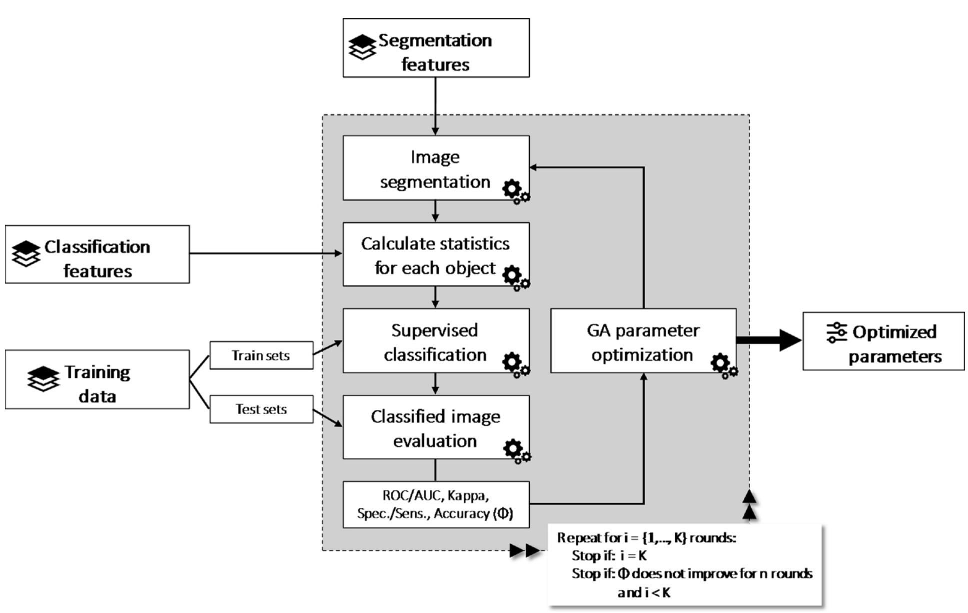

Fig. 2. Schematic representation of the fitness function.

Table 1
GA parameters used for running the image segmentation tests.

GA acronym	Definition	Value
popSize	The population size (number of individuals or solutions) generated at each round.	20
pcrossover	The probability of crossover between pairs of chromosomes. Typically defined as a large value close to 1.	0.8
pmutation	The probability of mutation in a parent chromosome. Usually mutation occurs with a small probability.	0.2
elitism	The number of best fitness individuals to survive at each round.	1
maxiter	The maximum number of iterations to run before the GA search is halted.	100
run	The number of consecutive generations without any improvement in the best fitness value before the GA is stopped.	20

Table 2
List of software packages and algorithms for image segmentation currently available in *SegOptim*. Also includes the list of currently optimizable parameters by *SegOptim*.

Software / package	Version	Algorithm(s)	Optimizable image segmentation parameters
ESRI® ArcGIS	10.3.1	Mean-shift	Spatial detail Spectral detail Minimum segment size
GRASS GIS	7.2.0	Region growing	Merge threshold Minimum segment size
Orfeo Toolbox	5.10.1	Large-scale mean-shift	Spatial detail Spectral detail Minimum segment size
RSGISLib	3.2	Shepherd k-means iterative elimination	Number of clusters Spectral threshold Minimum segment size
SAGA GIS	4.0.1	Simple region growing	Bandwidth Gaussian weighting bandwidth Variance in feature space Variance in position space
TerraLib	5.2.1	Mean region growing	Merge threshold Minimum segment size
		Multi-resolution image segmentation	Compactness weight Colour weight Minimum segment size Merging threshold

imagery.

SegOptim uses a simple threshold rule for performing the conversion of user-defined training samples or areas (in a raster representation) into training segments. This uses a threshold value ($t =]0,\ 1]$) defined by the user and the areal proportion ($a_c = [0,\ 1]$) of the segment covered by each class $c = \{0,\ 1\}$ (plus the background or *NoData* class). Then, the following rule is applied for single-class problems:

$$\begin{cases} 1,\ if\ a_{c=1} \geq t \\ 0,\ if\ a_{c=0} = 0 \\ Null,\ otherwise \end{cases}$$

For multi-class problems, this threshold rule operates in a slightly different manner. First, a cross-tabulation matrix is generated by calculating the proportion of each segment covered by each land cover class (Fig. 3).

After determining the majority class (i.e., the class with higher areal coverage) the train function verifies if that class coverage is above or equal to t, if not, the segment is removed from the training set. In the example (Fig. 3), if $t = 0.5$, then segments with identifiers $SID = \{1{,}2, N\}$ would be kept while $SID = \{3\}$ would be removed. Overall, the higher the value of the threshold, t, the 'purer' the training segments will be.

2.2.2. Image data pre-processing and segmentation features

Input imagery used for *SegOptim* testing was selected to cover a diverse set of spatial resolutions (from high at 20 m to 'ultra'-high at 0.03 m), different EO platforms (from airborne: SenseFly® UAV, to spaceborne: WorldView-2, RapidEye and Sentinel-2a) and different spectral resolutions (from 3 up to 10 bands; Table 3 and Fig. 4).

Image data from a SenseFly® UAV were acquired and pre-processed according to Gonçalves et al. (2016) for test site E1 on 2013-03-17, and for test site E2 on 2014-11-20. Raw digital number (DN) values were used for image segmentation and classification. For test site E3, one WorldView-2 image for 2013-06-23 was pre-processed and orthorectified according to Monteiro et al. (2017). For E4, one RapidEye image for 2014-10-29 (not previously used) was freely available from ESA's EOLi-SA Catalogue and Ordering Service and the Copernicus Urban Atlas initiative. We pre-processed image data and radiometrically calibrated to top-of-the-atmosphere (TOA) reflectance using at-sensor metadata. As for test sites E5 and E6, Sentinel-2a data (also not previously used) was collected from ESA's Sentinel DataHub (URL: https://scihub.copernicus.eu) for two different dates: 2016-03-14 and 2016-07-19 and processed to bottom-of-the-atmosphere (BOA) reflectance (level L2A) using SNAP with *sen2cor* atmospheric correction plugin (Mueller-Wilm et al., 2016). Segmentation features mainly coincide with spectral bands used as ancillary information to acquire ground-truth data in the field or for manually digitizing training samples. The segmentation feature space encompassed less dimensions (in comparison to the classification feature space) to allow fast processing needed to run segmentation. For E1, E2, E3, and E4 the Red (R), Green (G) and Blue (B) bands were employed while for E5 and E6 we used the R, G, B bands plus the Near-infrared (NIR). In particular, for test site E6 we used these four bands for the two available dates (March and July 2016) to improve segmentation results and capture seasonal differences.

2.2.3. Classification features

Classification features were used to extract signatures related to each class of interest. These data enable the classifier to learn from class attributes and later use those signatures to label each object in the image. For each test site, several types of classification features were used (Table 4), namely: (a) spectral data – reflectance values or raw DN values in case of UAV images; (b) alternative colour spaces (other than RGB but with better separability between components; e.g., XYZ, YUV); (c) multiple combinations of band ratios in the form: $BR = b_i\ /\ b_j$ where b_i and b_j are any two different spectral bands; (d) vegetation and/or water indices (e.g., NDVI, EVI, NDWI); (e) multiple combinations of Normalized Difference Indices (NDI) in the form $NDI = (b_i - b_j)/(b_i + b_j)$; (f) texture operators such as Haralick (Haralick, 1979) (HL; e.g., energy, entropy, correlation, inertia), Local Statistics (LS; e.g., mean, variance) or Structural Feature Set (Huang et al., 2007) (SFS; e.g., Length, Width, PSI, W-Mean) based on different kernel sizes and parameterizations; and (g) topographic features (e.g., elevation, slope, curvature) or surface elevation.

2.2.4. Feature importance evaluation

Based on the best solution for each test site, we calculated feature importance scores from the random forest algorithm. This score measures the total decrease in node impurities from splitting on each feature, averaged over all trees. For classification, the node impurity (defined broadly as how well the trees split the data) was measured by the Gini index. Calculations used the 'importance' function from *randomForest* R package (Liaw and Wiener, 2002). Due to differences in

Table 3
List of test sites and description of testing objectives (UL – upper-left corner and LR – lower-right corner coordinates in WGS 1984 geographic). The six test sites (E1-E6) are listed based on the spatial resolution of input EO imagery.

Site code	Site name and description	Problem type	Classification task / context	EO platform	Spatial resolution	Spectral / radiometric resolution	Test area size (pixels)
E1	*Mindelo/Árvore, V. Conde* UL: (41°19'42.39"N, 8°44'14.4"W) LR: (41°19'41.42"N, 8°44'13.11"W) Elevation: 8-10 m	Single-class	*Carpobrotus spp.* invasive species mapping (2 levels: invaded/non-invaded)	SenseFly® UAV	0.06 - 0.10 m	4 bands (VIS, NIR); 8-bit	1000 × 1000
E2	*Serra d'Arga* mountain range, Caminha UL: (41°49'16.93"N, 8°42'51.53"W) LR: (41°49'14.98"N, 8°42'48.95"W) Elevation: 741-787 m	Multi-class	Natural habitats mapping (9 target classes)	SenseFly® UAV	0.03 m	3 bands (VIS); 8-bit	1000 × 1000
E3	*Vilar da Veiga, Peneda-Gêres* National Park UL: (41°44'14.11"N, 8°10'53.58"W) LR: (41°43'8.65"N, 8°9'27.89"W) Elevation: 290-957 m	Single-class	*Acacia spp.* invasive species mapping (2 levels: invaded/non-invaded)	WorldView-2	2 m	8 bands (VIS, NIR); 11-bit	1000 × 1000
E4	*Viana do Castelo* city area UL: (41°42'51.48"N, 8°51'53.13"W) LR: (41°40'9.05"N, 8°48'17.26"W) Elevation: 0-70 m	Multi-class	Land cover/use mapping (9 target classes)	RapidEye	5 m	5 bands (VIS, NIR); 12-bit	1000 × 1000
E5	*Caminha/V. Castelo* municipalities UL: (41°49'53.78"N, 8°51'25.79"W) LR: (41°41'46.51"N, 8°43'9.37"W) Elevation: 5-742 m	Single-class	*Acacia dealbata* invasive species mapping (2 levels: invaded/non-invaded)	Sentinel-2a	10 m	4 bands (VIS, NIR); 12-bit	1150 × 1500
E6	*Vez* river watershed, *Arcos de Valdevez* municipality UL: (42°1'3.7"N, 8°31'31.62"W) LR: (41°48'16.6"N, 8°15'22.24"W) Elevation: 13-1419 m	Multi-class	Land cover/use mapping (10 target classes)	Sentinel-2a	10-20 m	10 bands (VIS, NIR, SWIR); 12-bit	2250 × 2350

SID	Class_1	Class_2	Class_3	…	Class_n	Majority_class
1	0.2	0.8	0.0	…	0.0	2
2	0.5	0.2	0.1	…	0.2	1
3	0.2	0.2	0.2	…	0.4	n
…	…	…	…	…	…	…
N	0.0	0.0	1.0	…	0.0	3

Fig. 3. An example cross-tabulation by segment and land cover class (SID: segment unique identifier).

feature importance scores (imp_f) between test sites, we normalized them as $nimp_f = imp_f / max(imp_f)$. Then, we obtained the maximum $nimp_f$ value for each type of feature (e.g., NDI combinations, topography/surface, texture) in each test site.

2.2.5. Computing time measurements

For comparing the amount of time taken by each segmentation algorithm to achieve an optimal solution, we measured the computing times required to achieve this objective for all test sites with the random forest algorithm in three replicate rounds. These measures were obtained in a desktop PC running MS Windows 7 Enterprise 64-bit SP1, with an Intel® Core™ i7-4770S CPU @ 3.10 GHz (4 cores, 8 threads), 16Gb RAM DDR3 and an SSD drive (SATA-II 3.0Gb/s). To speed-up computations, we used five parallel cores in GA runs.

3. Results

3.1. Classification performance across test sites

Supervised classification performance for 5-fold cross-validation, based on optimized image segmentation, attained very good to excellent results across the six test sites (see Table 5 and Fig. 5). Kappa values ranged from 0.96 (E1) to 1.00 (E3) for single-class tests and, between 0.85 (E2) to 0.94 (E6) for multi-class tests (see also Supp. Info. – Appendix S4 with confusion matrices and producer/user accuracies).

Overall, no particular image segmentation algorithm outperformed its competitors in our test. Still, *TerraLib* and SAGA region growing-based algorithms were selected twice. In contrast, for classification methods, the RF algorithm has shown very good overall performance being selected five out of six times as the best performing classification algorithm (in some instances tied with other techniques, E3 and E6).

3.2. Performance differences and optimization results per test site

When analysing the distribution of paired absolute differences of performance (Kappa index) for each test site across segmentation algorithms (median = 0.03) and across classification algorithms (for the same segmentation method; median = 0.06) we found a significant difference ($p < 0.01$) with larger performance differences arising from the classification algorithm used, rather than the type of segmentation method. However, when looking at disaggregated differences per test site, in fact, for half of the situations we found greater differences originated by the segmentation algorithm [although not all were significant, E1: not-significant, E3: ($p < 0.01$) and E4: ($p < 0.1$)] and half otherwise [E2: ($p < 0.001$), E5: ($p < 0.01$) and E6: ($p < 0.01$)].

Results also showed a large difference, equal to 0.25 (in average), between the best and worst performer combination of segmentation/classification algorithms for each test site although these comparisons are always based on optimized setups (Table 5).

The best segmentation algorithm *per* test site were E1: SAGA GIS Simple Region Growing, E2: RSGISLib Shepherd iterative elimination, E3: multiple segmentation algorithms were selected, E4: GRASS GIS region growing, E5: TerraLib Mean Region Growing and, for test site E6: Orfeo Toolbox Mean-shift. The complete set of all optimized parameters can be found in Supp. Info. – Appendix S5.

In terms of improvement degree due to GA parameter optimization, we also recorded widely different results across tests, varying from relatively low improvement (observed in E6 or E2; Fig. 6) to high improvement (E1, E4 or E5; Fig. 6). Despite some differences in the degree of improvement, GA optimization was always capable of bettering segmentation parameters in comparison to the initial settings or to the average.

3.3. Feature importance

Due to differences between test sites in terms of input classification features (related to dissimilarities in spatial and spectral resolutions of the tested images), extracting general patterns of feature importance is difficulted. Nonetheless, results show that, in decreasing order of importance, NDI combinations, spectral band data (in E6 for different seasons, spring and summer) combined with features extracted from topography/land surface provided good results (Fig. 7). Along these, commonly used vegetation/water indices and band ratios also provide relevant features for classification (for more details see Supp. Info. – Appendix S6).

3.4. Computing times

Computing time measurements (Fig. 8) show that ArcGIS/Mean-shift (avg. = 1.12h, std.-dev. = 1.17h), SAGA Region-growing (1.51h, 1.63h), and RSGISLib Shepherd (2.15h, 2.32h) were the fastest three algorithms in the tests, while TerraLib Baatz (2.23h, 1.85h), GRASS Region-growing (2.27h, 1,90h), TerraLib mean region-growing (2.34h, 2.05h), and Orfeo Toolbox Large-scale mean-shift (2.92h, 3.06h), performed relatively slower. A linear model, used to predict computing times, showed that the number of pixels of the input image along with the number of GA iterations, are capable of explaining *ca.* 82% of the variance ($R^2 = 0.82$, $F(2,124) = 273.2$, $p < 0.001$; see also Supp. Info. – Appendix S7). Despite both variables were considered significant for explaining computing times, image size was clearly more important ($t = 16.10$, $p < 0.001$).

4. Discussion

4.1. Main contributions to improve GEOBIA workflow

In this study we developed a generic approach to object-based analysis comparison and optimization which integrates two crucial steps in GEOBIA workflows typically exhibiting mutual dependencies: image segmentation and classification (Baatz et al., 2008). Due to their flexibility and power, we selected GA as a suitable method to optimize and fine-tune segmentation parameter values. Overall, the use of GA for optimizing image segmentation parameters with the aim of maximizing the thematic accuracy of supervised classification, together with the comparison of multiple techniques, represents a significant advancement in GEOBIA workflows.

A new open-source R package (*SegOptim*) implements the proposed approach with a rich set of features for testing, comparing and optimizing different image segmentation and classification algorithms within the R environment and computing language (R Development Core Team, 2017). The several tests performed for this study strongly suggest that *SegOptim* provides a novel and robust comparison of several state-of-the-art algorithms and software focused on the object-based analysis of EO data for different application contexts [see also, e.g., (Li et al., 2016; Marpu et al., 2010; Meinel and Neubert, 2004; Teodoro and Araújo, 2016)].

4.2. Performance comparison and best option selection

Given their wide accessibility for users, comparisons focused on open-source software for GIS and Remote Sensing analyses. GA-

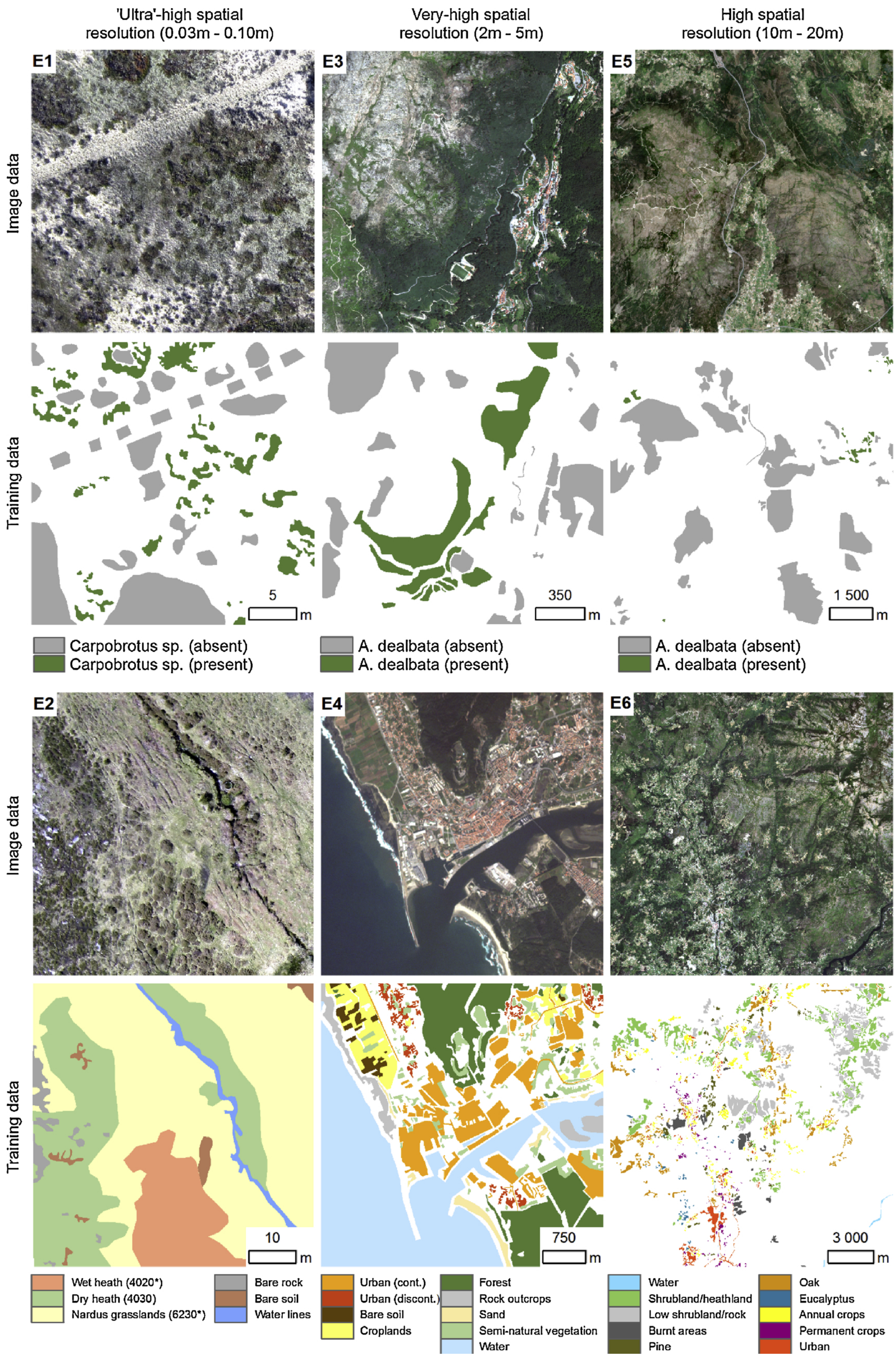

Fig. 4. Segmentation features and training data used for testing the GA-based approach in *SegOptim* package for single-class problems (detection of invasive species; E1, E3 and E5 test sites in top-two rows) and multi-class (habitat and land cover/use mapping; E2, E4 and E6 test sites in bottom-two rows).

Table 4
Types of classification features used for each test site. NDI – Normalized Difference Index combinations based on any two different bands; HL – Haralick texture features; LS – Local Statistics features; SFS – Structural Feature Set texture features. (*) for the features in E6, two images for March and July 2016 were used to calculate classification features.

Test site	Classification features								
	Spectral bands	Alternative colour spaces	Band ratios	Vegetation / water indices	NDI combinations	Texture operators			Topography / surface elevation
						HL	LS	SFS	
E1	✓	✓	✓	✓	✓				
E2	✓		✓			✓	✓	✓	✓
E3	✓			✓	✓				
E4	✓			✓	✓				
E5	✓			✓	✓				
E6	✓*			✓*	✓*				✓

Table 5
Results for 5-fold cross-validation using the Kappa index to evaluate object-based classification performance based on *SegOptim*'s GA-optimized image segmentation for each algorithm. Values highlighted in bold represent the best solutions for each test site. E1, E3 and E5 are single-class problems while the remaining E2, E4 and E6 are multi-class problems.

Segmentation software / algorithm	Classification method	Test site					
		E1	E2	E3	E4	E5	E6
ESRI® ArcGIS – Mean-shift	RF	0.96	0.78	0.77	0.86	0.91	0.92
	SVM	0.90	0.74	0.78	0.83	0.94	0.93
	GBM	0.95	0.59	0.84	0.83	0.82	0.87
GRASS GIS – Region Growing	RF	0.87	0.79	0.97	**0.92**	0.95	0.91
	SVM	0.88	0.73	0.94	0.70	0.93	0.93
	GBM	0.90	0.62	0.95	0.85	0.87	0.81
SAGA GIS – Simple Region Growing	RF	0.93	0.79	**1.00**	0.88	0.91	0.93
	SVM	0.84	0.76	**1.00**	0.87	0.87	0.92
	GBM	**0.96**	0.66	**1.00**	0.84	0.82	0.86
Orfeo Toolbox – Large Scale Mean-shift	RF	0.89	0.80	0.92	0.88	0.94	**0.94**
	SVM	0.86	0.73	0.90	0.86	0.90	**0.94**
	GBM	0.89	0.61	0.83	0.84	0.78	0.87
RSGISLib – Shepherd Iterative Elimination	RF	0.79	**0.85**	0.99	0.81	0.86	0.86
	SVM	0.75	0.74	0.99	0.79	0.87	0.86
	GBM	0.75	0.61	0.99	0.76	0.71	0.76
TerraLib – Mean Region Growing	RF	0.92	0.80	**1.00**	0.69	**0.97**	0.90
	SVM	0.94	0.72	**1.00**	0.66	0.97	0.73
	GBM	0.88	0.61	0.92	0.60	0.84	0.81
TerraLib – Multiresolution Baatz-Schäpe	RF	0.79	0.81	**1.00**	0.88	0.92	0.93
	SVM	0.93	0.74	**1.00**	0.87	0.93	0.93
	GBM	0.88	0.63	**1.00**	0.85	0.86	0.85

optimized results for image segmentation and object-based classification allowed to identify which algorithms performed best for each situation. Tests encompassed a wide range of sites across northern Portugal from coastal beach environments or urban settlements up to complex mosaic landscapes in mountainous areas covering distinct environmental conditions, landscape patterns and dynamics. Different input EO imagery were also tested, ranging from 'ultra'-high resolution UAV data (0.03 - 0.10 m), very-high resolution (WorldView2, 2 m - RapidEye, 5 m) to high resolution Sentinel-2a (10–20 m) multi-temporal images. Calibration conditions for image segmentation, classification and optimization were also diverse encompassing a wide range of problems and objectives from single-class (aiming to identify and map the distribution of invasive species) to multi-class problems (with the objective of discriminating and mapping natural habitats or land cover/use classes).

Our results highlight that no particular combination of an image segmentation and a classification algorithm is suited for all the tasks/objectives, an empirical verification of the "*no free-lunch theorem*" [NFLT; (Wolpert, 1996; Wolpert and Macready, 1997)] for object-based image classification problems. Broadly described, the NFLT (which, among other, applies to machine-learning and optimization problems) implies that learning algorithms "cannot be universally good" across all problems and domains (Magdon-Ismail, 2000). This seems to apply also in our particular case and especially for image segmentation algorithms. Besides theoretical implications, this means that comparing and optimizing procedures is crucial for determining which ones are more suited for a particular objective. *SegOptim* provides the required functionalities for this purpose. Despite these results, the RF classifier has shown good overall performance across tests, which is coherent with previous findings (Fernández-Delgado et al., 2014). However, this may be a consequence of this algorithm requiring much less fine-tuning in comparison to other techniques such as SVM (Breiman, 2001).

The results from our tests also showed that, depending on the specific problem under consideration, either segmentation algorithms produce greater differences in object-based analyses or (more likely) different classifiers will generate stronger differences. Still, most studies focus only on comparing performance differences between classifiers [e.g., Li et al. (2016)]. In addition, algorithm comparisons may be of particular importance given that we found strong differences between "best" and "worst" performers for the same task/objective, which may imply that the probability of obtaining sub-optimal results without comparing available methods and/or optimizing its parameters is fairly high.

Deciding upon which features to use (e.g., spectral, texture, shape) and pre-selecting those features is key to any GEOBIA workflow in order to obtain good classification results (Ma et al., 2017). Although feature selection optimization is at the moment not included in the package, users should be aware of this issue and exploit different types of features, thus taking advantage of *R*'s ability to analyse data and iteratively feed classification pipelines. Inspecting feature importance scores also provides a way to support decisions, as was done here (cf. Fig. 7 and Supp. Info. – Appendix S6). Our results show that using multiple normalized difference combinations of image bands provided the best results in our tests. In future *SegOptim* releases we aim to improve these functionalities and provide better guidance to users.

Besides the "*best solution*" (i.e., the one with best performance), other solutions may also be visually inspected because these may also provide good results worth to consider. Lang (2008) considers that human perception or cognition skills required to evaluate the delineated and classified objects is still the "ultimate benchmark" for assessment. In the same sense, user control and expert decision are still required to assess situations of potential under- or over-segmentation, and to make informed choices, similarly to what Grybas et al. (2017) found for unsupervised segmentation parameter optimization.

An alternative to selecting the "best solution" could be to ensemble multiple solutions (a.k.a. "classifier stacking" or "decision fusion"). Although we could not find any examples combining different segmentation methods, 'fusing' multiple classifiers is often performed with gains in terms of accuracy (Clinton et al., 2015; Löw et al., 2015; Oza and Tumer, 2008).

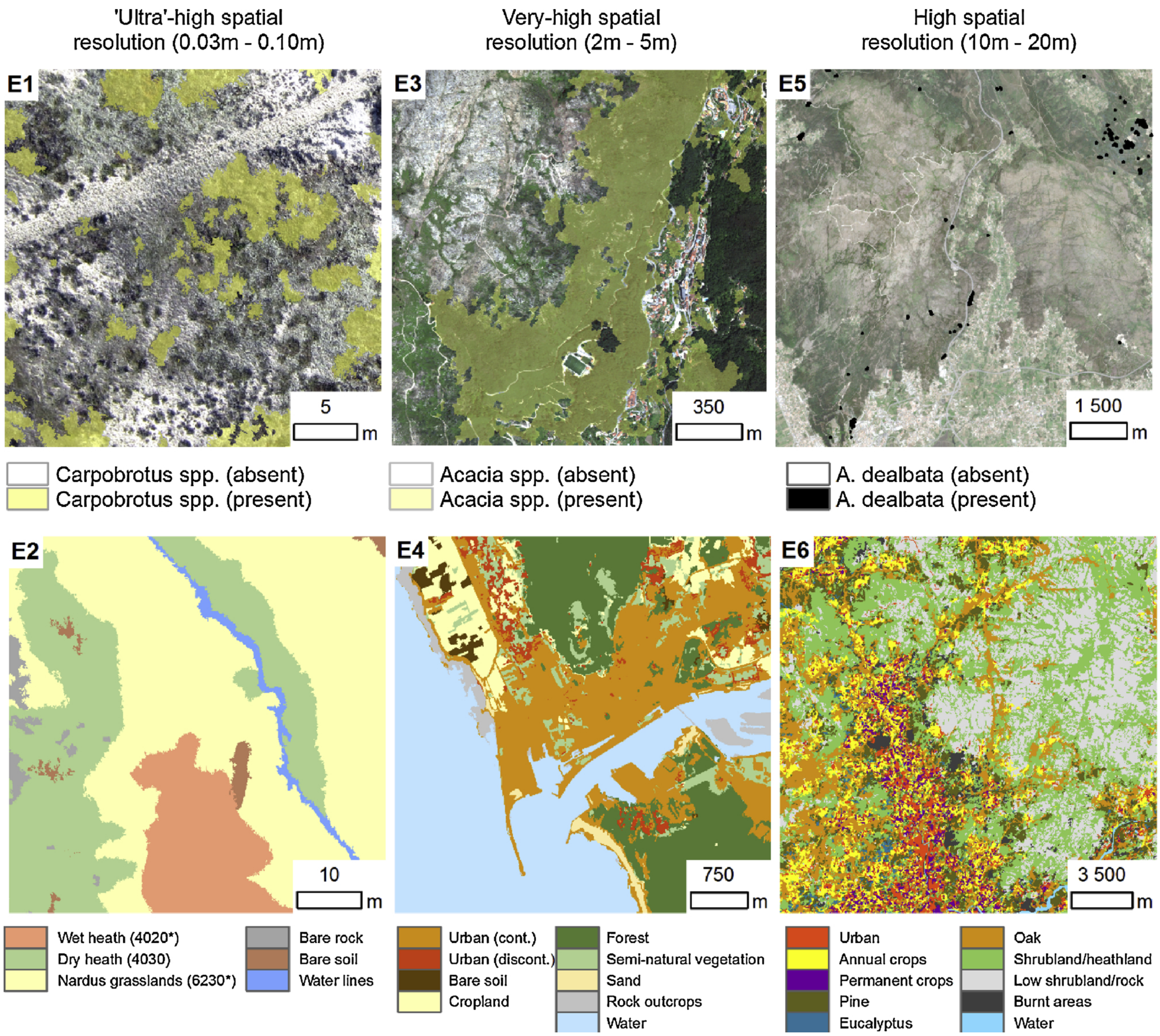

Fig. 5. Classification outputs generated from the best GA-based solutions for each test-site. E1, E3 and E5 are single-class problems while the remaining E2, E4 and E6 are multi-class problems. E1 depicts the distribution of *Carpobrotus* sp. invasive species using SAGA GIS Simple Region Growing segmenter and GBM classifier; E2 maps different natural habitats using RSGISLib Shepherd Iterative Elimination segmenter and Random Forest (RF) classifier; E3 shows the distribution of *Acacia dealbata* invasive species (yellow patches) using TerraLib Baatz-Schäpe multi-resolution segmenter and RF classifier; E4 shows land cover/use classification using GRASS GIS Region Growing segmenter and RF classifier; E5 shows the distribution of small patches of the invasive species *Acacia dealbata* across the landscape using TerraLib Mean Region Growing segmenter and RF classifier; and, E6 shows land cover/use mapped with Orfeo Toolbox Mean-shift segmenter and RF classifier. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

4.3. A general framework for GEOBIA optimization

GEOBIA has become a popular toolset for performing land cover classification among many other applications. However, the critical step of defining the best values for image segmentation parameters remains problematic. Several works proposed different unsupervised (Dragut et al., 2014; Grybas et al., 2017) or supervised methods (Clinton et al., 2010; Liu et al., 2012; Montaghi et al., 2013) for this purpose, with both groups of methods holding advantages and caveats.

Our supervised approach however, differs from what we found in the literature, since previous approaches typically measure the geometric, arithmetic, topological or spectral discrepancy between a reference dataset and image segments. Instead, the cornerstone of our approach is the close integration between both the image segmentation and the classification steps common in GEOBIA workflows (Baatz et al., 2008; Blaschke, 2010; Lang, 2008). Hence, the objective was to use GA for optimizing image segmentation parameters, with the aim of maximizing the thematic accuracy of supervised classification, thus benefiting both steps in an iterative fashion. This way we provide a relatively simple and generalizable approach for optimizing object-based image analysis and compare multiple techniques. From our tests, we observed that this approach allows to define segmentation parameters which balance between the over- and under-segmentation spectrum across all target classes, without a relevant loss in overall classification performance.

Although GA-based optimization may help to improve object-based analyses, it is still crucial that the human operator carefully defines every input and monitors the optimization process. More specifically, in some preliminary runs we found that GA can provide useless or low performance results if (for example) segmentation parameter bounds are defined too broadly. It is up to the user to rectify those kinds of situations. To profit from GA-based optimization, and to avoid high

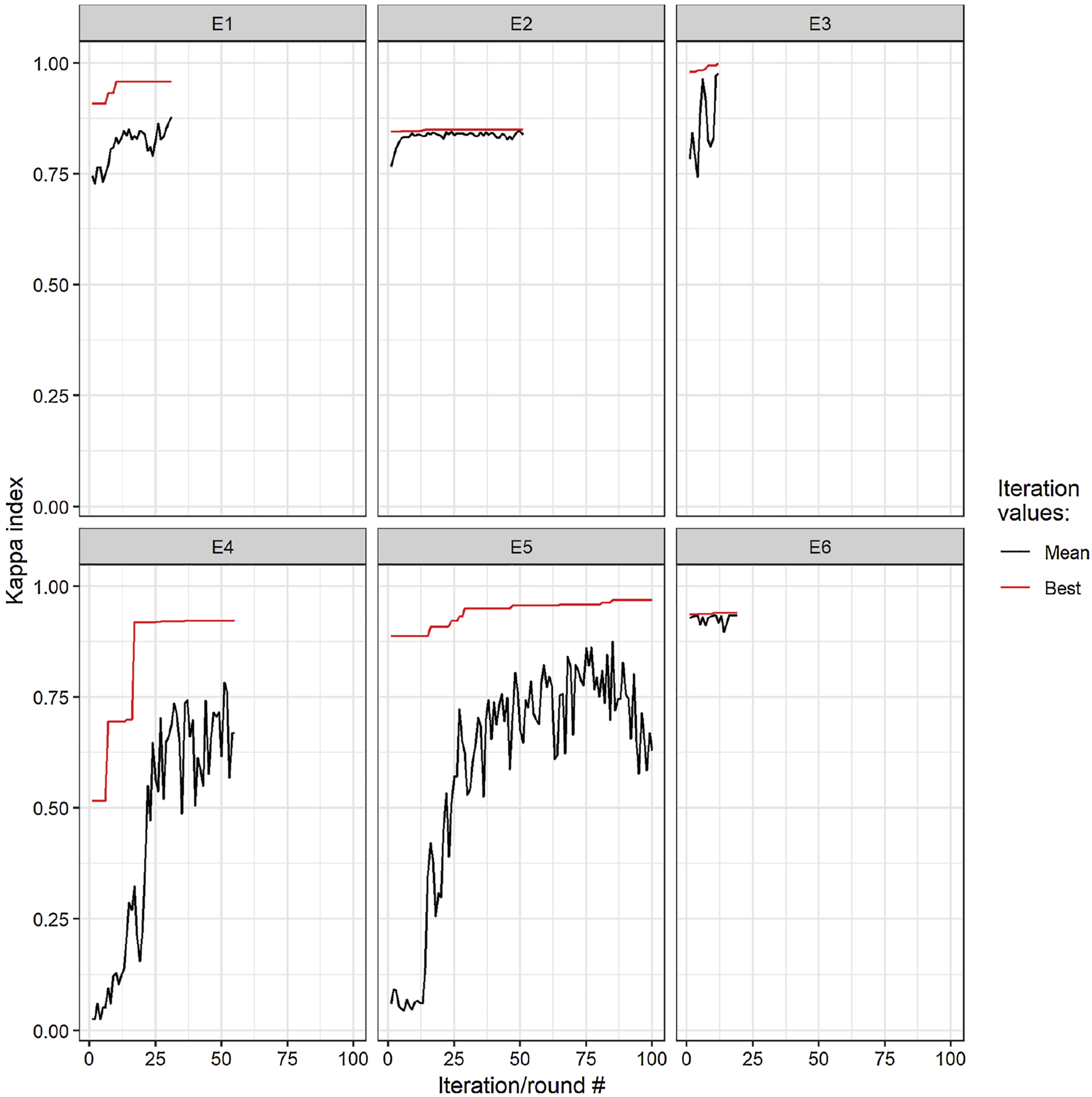

Fig. 6. GA optimization paths for each test site showing different convergence profiles in terms of speed: fast for E1, E3 and E6, intermediate for E2 and E4, and slow for E5. These paths represent only the best performing solutions (*i.e.*, combination of an image segmentation and classification algorithms) for each test site. The black line represents the average Kappa for all individuals (n = 20) at each round of the GA optimizer while the red line shows the Kappa value of the best solution by round. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

computational costs, the user must carefully set up its parameters as well as the search space.

4.4. Future developments and outlook

The volume of data has grown immensely in recent years due to high- and very-high spatial resolution EO platforms. “Big data” poses several problems and challenges for processing, and not all available GEOBIA algorithms and software are up to the task [but see, e.g., Michel et al. (2015) or Happ et al. (2016)]. The *SegOptim* package by performing complex analyses in a desktop-based environment is therefore limited to the available resources in terms of RAM, CPU power and disk space/speed. One way to circumvent this problem, and still optimize object-based analysis, is to extract smaller (and representative) regions-of-interest (ROI) preferentially through an adequate spatial sampling design (e.g., Köhl et al., 2006) and process each ROI independently. By inspecting the distribution properties of the target segmentation parameters for all ROI’s it is still possible to obtain useful information without having to use a complete scene.

Another limitation of our GEOBIA optimization approach is that GA require some time to process, varying from minutes to several hours depending on the segmentation algorithm, on its parametrization and, more strongly, on the input image size and on the number of GA iterations used for convergence (Fig. 7). A possible solution to speed-up computations is to use parallel processing, already implemented in R’s ‘GA’ package (Scrucca, 2013). As mentioned earlier, another possible solution is to provide a thoughtful and problem-specific configuration of GA optimization parameters (among other) leading to faster and

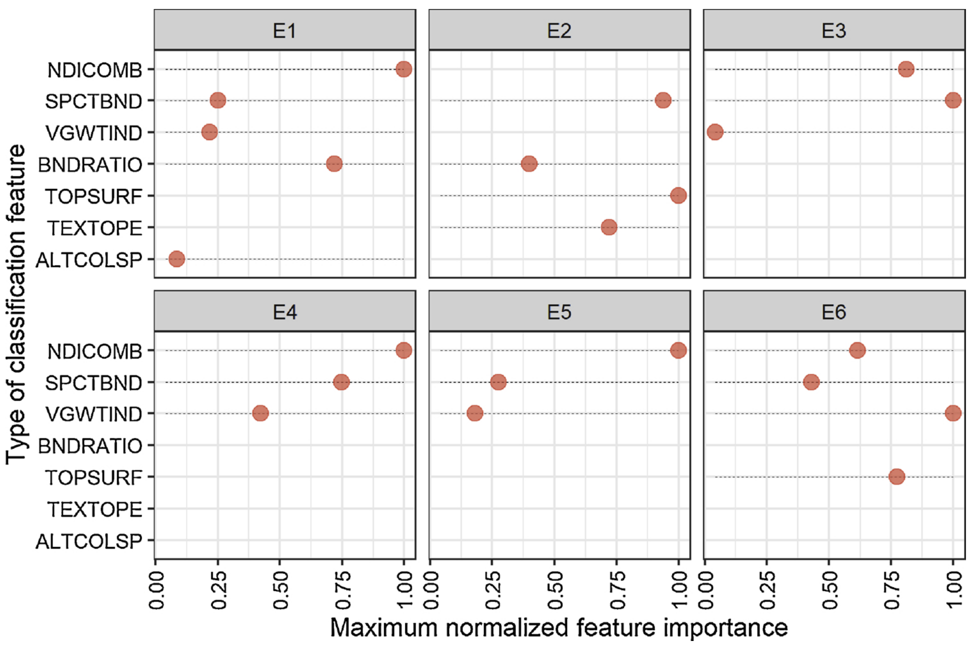

Fig. 7. Maximum normalized feature importance by type calculated from random forest algorithm for the best solution *per* test site (E1 to E6). Feature types are ordered by their usage frequency in the tests (left side top: more frequent, bottom: less frequent). Feature types are: NDICOMB – normalized difference index combinations, SPCBND – spectral bands, VGWTIND – vegetation or water spectral indices, BNDRATIO – band ratios, TOPSURF – topography and surface elevation, TEXTOPE – texture operators, ALTCOLSP – alternative color spaces.

more meaningful results.

Besides these computational limitations, our proposed approach focuses on devising a non-hierarchical, single-level classification for the area of interest, thus inherently simplifying some aspects, and making it less applicable to some settings or goals. Albeit 'simple', our optimized object-based classification approach may still be useful and adaptable to several situations. In addition, future developments of the package aiming to explore hierarchical, contextual and topological properties and relations between objects as features for classification will pave the way to obtain enhanced performances. Moreover, further developments (already on the way) will also include discrepancy measures between reference polygons and corresponding segments (Liu et al., 2012) for optimizing image segmentation.

In fact, *SegOptim* open-source R package offers users the possibility to test, compare and optimize different GEOBIA methods related to image segmentation and classification with benefits for producing thematic maps for land cover characterization or mapping specific classes based on EO image data. Besides these features, this package

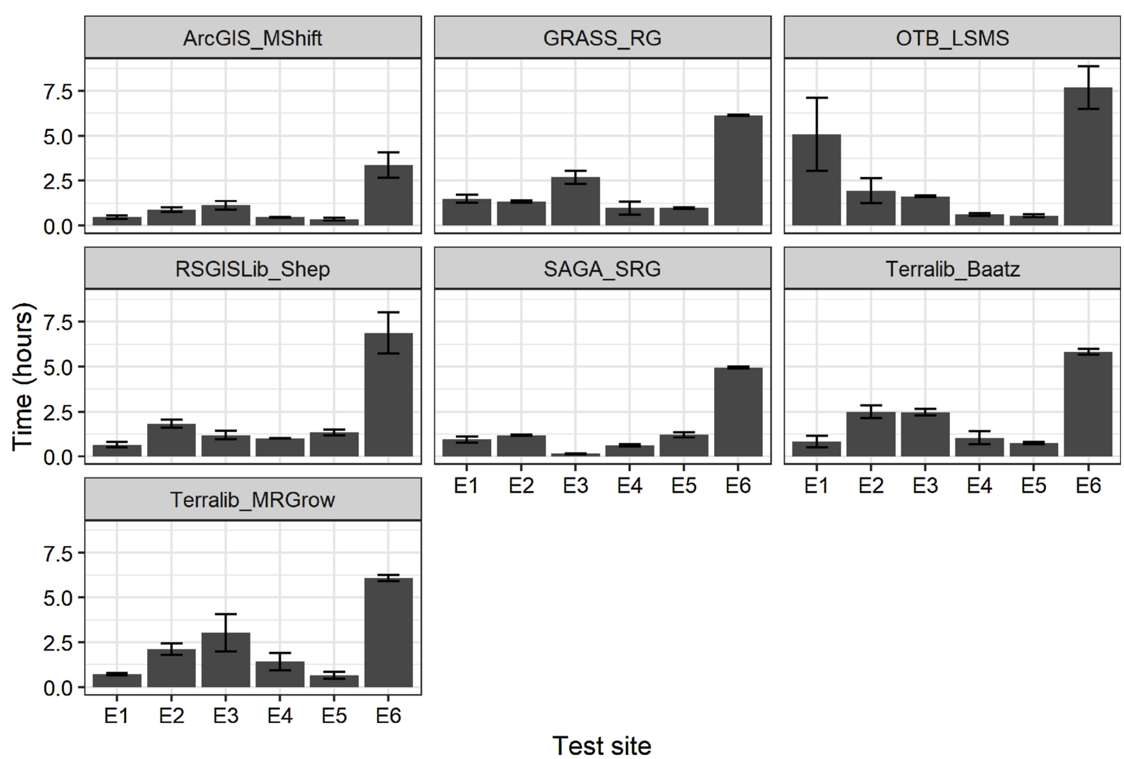

Fig. 8. Average computing times taken to achieve the optimal solution in GA (in hours ± std.-error, n = 3). Each bar represents one test site (E1 to E6) and each plot an image segmentation technique (ArcGIS_MShift – ESRI ArcGIS mean-shift, GRASS_RG – GRASS region-growing, OTB_LSMS – Orfeo Toolbox large scale mean-shift, RSGISLib_Shep – RSGISLib Shepherd k-means iterative elimination algorithm, SAGA_SRG – SAGA simple region-growing, Terralib_Baatz – TerraLib Baatz-Schäpe multi-resolution segmentation, Terralib_MRG – TerraLib mean region-growing). Time measurements used the Random Forest algorithm for classification.

also offers the possibility to use several unsupervised classification methods and compare different solutions through cluster validity indices. Exploring the latter in detail is, however, outside the scope of this study. By providing free access to *SegOptim* we aim to overcome limitations for accessing methods that may be useful for all users interested in applying these techniques.

Still, *SegOptim* is only giving its first steps as a computational solution for such a complex problem. The roadmap of future developments includes (among others) user-defined fitness functions based on cross-validation results, integrating and testing more algorithms already available (for segmentation but mostly for classification), multi-classifier stacking/fusion, calculation of shape and context metrics for segments, the possibility of evaluating and performing optimization through supervised discrepancy measures or even unsupervised techniques. We are hopeful that researchers and practitioners may find the package useful for their activities and research, allowing to test, adjust and improve its functionalities over time.

Acknowledgments

J. Gonçalves, I. Pôças and B. Marcos were financially supported by FCT (Portuguese Foundation for Science and Technology), through grants SFRH/BD/90112/2012, SFRH/BPD/79767/2011 and SFRH/BD/99469/2014, respectively, funded by national and European funds (ESF), through POPH-QREN (2007-2013) and/or CSF (2014-2020).

References

Adams, R., Bischof, L., 1994. Seeded region growing. IEEE Trans. Pattern Anal. Mach. Intell. 16, 641–647.
Baatz, M., Hoffmann, C., Willhauck, G., 2008. Progressing from object-based to object-oriented image analysis. In: Blaschke, T., Lang, S., Hay, G.J. (Eds.), Object-Based Image Analysis - Spatial Concepts for Knowledge-Driven Remote Sensing Applications, 1st ed. Springer-Verlag, Berlin.
Baatz, M., Schape, A., 2000. multiresolution segmentation: an optimization approach for High quality multi-scale image segmentation. In: Strobl, J., Blaschke, T., Griesbner, G. (Eds.), Angewandte Geographische Informations-Verarbeitung, XII, Wichmann Verlag, Karlsruhe, Germany, pp. 12–23.
Bastos, R., D'Amen, M., Vicente, J., Santos, M., Yu, H., Eitelberg, D., Gonçalves, J., Civantos, E., Honrado, J., Cabral, J.A., 2016. A multi-scale looping approach to predict spatially dynamic patterns of functional species richness in changing landscapes. Ecol. Indic. 64, 92–104.
Bechtel, B., Ringeler, A., Boehner, J., 2008. Segmentation for object extraction of trees using MATLAB and SAGA. In: Boehner, J., Blaschke, T., Montanarella, L. (Eds.), SAGA - Seconds Out. Hamburger Beitraege zur Physischen Geographie und Landschaftsoekologie, pp. 59–70.
Blaschke, T., 2010. Object based image analysis for remote sensing. Isprs J. Photogramm. Remote. Sens. 65, 2–16.
Breiman, L., 2001. Random forests. Mach. Learn. 45, 5–32.
Castilla, G., Hay, G.J., 2008. Image objects and geographic objects. In: Blaschke, T., Lang, S., Hay, G.J. (Eds.), Object-Based Image Analysis - Spatial Concepts for Knowledge-Driven Remote Sensing Applications, 1st ed. Springer-Verlag, Berlin.
Clewley, D., Bunting, P., Shepherd, J., Gillingham, S., Flood, N., Dymond, J., Lucas, R., Armston, J., Moghaddam, M., 2014. A python-based open source system for geographic object-based image analysis (GEOBIA) utilizing raster attribute tables. Remote Sens. (Basel) 6, 6111.
Clinton, N., Holt, A., Scarborough, J., Yan, L., Gong, P., 2010. Accuracy assessment measures for object-based image segmentation goodness. Anglais 76, 289–299.
Clinton, N., Yu, L., Gong, P., 2015. Geographic stacking: decision fusion to increase global land cover map accuracy. Isprs J. Photogramm. Remote. Sens. 103, 57–65.
Comaniciu, D., Meer, P., 2002. Mean shift: a robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. 24, 603–619.
Dragut, L., Csillik, O., Eisank, C., Tiede, D., 2014. Automated parameterisation for multi-scale image segmentation on multiple layers. ISPRS J. Photogramm. Remote Sens. 88, 119–127.
Fernández-Delgado, M., Cernadas, E., Barro, S., 2014. Do we Need Hundreds of Classifiers to Solve Real World Classification Problems? J. Mach. Learn. Res. 15, 3133–3181.
Gonçalves, J., Henriques, R., Alves, P., Sousa-Silva, R., Monteiro, A.T., Lomba, Â., Marcos, B., Honrado, J., 2016. Evaluating an unmanned aerial vehicle-based approach for assessing habitat extent and condition in fine-scale early successional mountain mosaics. Appl. Veg. Sci. 19, 132–146.
Goslee, S.C., 2011. Analyzing Remote Sensing Data in R: The landsat Package. 43, 25.
Grybas, H., Melendy, L., Congalton, R.G., 2017. A comparison of unsupervised segmentation parameter optimization approaches using moderate- and high-resolution imagery. GIsci. Remote Sens. 54, 515–533.
Happ, P.N., Ferreira, R.S., Costa, G.A.O.P., Feitosa, R.Q., Bentes, C., Farias, R., Achanccaray, P.M., 2016. Interseg: A Distributed Image Segmentation Tool.
Haralick, R.M., 1979. Statistical and structural approaches to texture. Proceedings of the IEEE 67, 786–804.
Haupt, R.L., Haupt, S.E., 2004. Practical Genetic Algorithms, 2nd ed. ed. John Wiley & Sons, Hoboken, New Jersey.
Hay, G.J., Castilla, G., 2008. Geographic object-based image analysis (GEOBIA): a new name for a new discipline. In: Blaschke, T., Lang, S., Hay, G.J. (Eds.), Object-Based Image Analysis - Spatial Concepts for Knowledge-Driven Remote Sensing Applications, 1st ed. Springer-Verlag, Berlin.
Hijmans, R.J., 2016. raster: Geographic Data Analysis and Modeling. R package version 2. 5-8. URL: https://CRAN.R-project.org/package = raster.
Hojjatoleslami, S.A., Kittler, J., 1998. Region growing: a new approach. Ieee Trans. Image Process. 7, 1079–1084.
Huang, X., Zhang, L., Li, P., 2007. Classification and extraction of spatial features in urban areas using high-resolution multispectral imagery. Ieee Geosci. Remote. Sens. Lett. 4, 260–264.
Köhl, M., Magnussen, S., M., M, 2006. Sampling Methods, Remote Sensing and GIS Multiresource Forest Inventory. Springer-Verlag, Berlin Heidelberg.
Kuhn, M., Johnson, K., 2013. Applied Predictive Modeling, 1st ed. ed. Springer, New York.
Lang, S., 2008. Object-based image analysis for remote sensing applications: modeling reality – dealing with complexity. In: Blaschke, T., Lang, S., Hay, G.J. (Eds.), Object-Based Image Analysis - Spatial Concepts for Knowledge-Driven Remote Sensing Applications, 1st ed. Springer-Verlag, Berlin.
Leutner, B., Horning, N., 2017. RStoolbox: Tools for Remote Sensing Data Analysis. http://bleutner.github.io/RStoolbox, https://github.com/bleutner/RStoolbox. .
Li, M., Ma, L., Blaschke, T., Cheng, L., Tiede, D., 2016. A systematic comparison of different object-based classification techniques using high spatial resolution imagery in agricultural environments. Int. J. Appl. Earth Obs. Geoinf. 49, 87–98.
Liaw, A., Wiener, M., 2002. Classification and regression by randomForest. R News 2, 18–22.
Liu, Y., Bian, L., Meng, Y., Wang, H., Zhang, S., Yang, Y., Shao, X., Wang, B., 2012. Discrepancy measures for selecting optimal combination of parameter values in object-based image analysis. Isprs J. Photogramm. Remote. Sens. 68, 144–156.
Löw, F., Conrad, C., Michel, U., 2015. Decision fusion and non-parametric classifiers for land use mapping using multi-temporal RapidEye data. Isprs J. Photogramm. Remote. Sens. 108, 191–204.
Ma, L., Li, M., Ma, X., Cheng, L., Du, P., Liu, Y., 2017. A review of supervised object-based land-cover image classification. Isprs J. Photogramm. Remote. Sens. 130, 277–293.
Magdon-Ismail, M., 2000. No free lunch for noise prediction. Neural Comput. 12, 547–564.
Marpu, P.R., Neubert, M., Herold, H., Niemeyer, I., 2010. Enhanced evaluation of image segmentation results. J. Spat. Sci. 55, 55–68.
Meinel, G., Neubert, M., 2004. A comparison of segmentation programs for high resolution remote sensing data. Int. Arch. Photogr. Remote Sens. Spatial Inf. Sci. 35, 1097–1105.
Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., 2017. e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien. R package version 1.6-8. https://CRAN.R-project.org/package = e1071. .
Michel, J., Youssefi, D., Grizonnet, M., 2015. Stable mean-shift algorithm and its application to the segmentation of arbitrarily large remote sensing images. Ieee Trans. Geosci. Remote. Sens. 53, 952–964.
Montaghi, A., Larsen, R., Greve, M.H., 2013. Accuracy assessment measures for image segmentation goodness of the Land Parcel Identification System (LPIS) in Denmark. Remote. Sens. Lett. 4, 946–955.
Monteiro, A., Gonçalves, J., Fernandes, R., Alves, S., Marcos, B., Lucas, R., Teodoro, A., Honrado, J., 2017. Estimating invasion success by non-native trees in a national park combining WorldView-2 very high resolution satellite data and species distribution models. Diversity (Basel) 9, 6.
Mueller-Wilm, U., Devignot, O., Pessiot, L., 2016. S2 MPC: Sen2Cor Configuration and User Manual (S2-PDGS-MPC-L2A-SUM-v2.3). ESA.
Nauss, T., Meyer, H., Detsch, F., Appelhans, T., 2015. Manipulating Satellite Data With Satellite (R-package version 0.2.0; URL:. https://cran.r-project.org/web/packages/satellite).
Novelli, A., Aguilar, M., Aguilar, F., Nemmaoui, A., Tarantino, E., 2017. AssesSeg—a command line tool to quantify image segmentation quality: a test carried out in southern Spain from satellite imagery. Remote Sens. (Basel) 9, 40.
Oza, N.C., Tumer, K., 2008. Classifier ensembles: select real-world applications. Inf. Fusion 9, 4–20.
R Development Core Team, 2017. R: A Language and Environment for Statistical Computing; URL: https://www.R-project.org/. R Foundation for Statistical Computing, Vienna, Austria.
Räsänen, A., Rusanen, A., Kuitunen, M., Lensu, A., 2013. What makes segmentation good? A case study in boreal forest habitat mapping. Int. J. Remote Sens. 34, 8603–8627.
Ridgeway, G., 2017. Gbm: Generalized Boosted Regression Models. R Package Version 2. pp. 1–3. https://CRAN.R-project.org/package = gbm.
Rothlauf, F., 2011. Optimization methods. Design of Modern Heuristics. Springer-Verlag, Berlin, Heidelberg.
Scrucca, L., 2013. GA: a package for genetic algorithms in R. J. Stat. Softw. 53, 1–37.
Teodoro, A.C., Araújo, R., 2016. Comparison of performance of object-based image analysis techniques available in open source software (Spring and Orfeo Toolbox/Monteverdi) considering very high spatial resolution data. J. Appl. Remote Sens. 10, 1–22.
Tremeau, A., Borel, N., 1997. A region growing and merging algorithm to color segmentation. Pattern Recognit. 30, 1191–1203.
van der Werff, H.M.A., van der Meer, F.D., 2008. Shape-based classification of spectrally identical objects. Isprs J. Photogramm. Remote. Sens. 63, 251–258.
Wolpert, D.H., 1996. The lack of a priori distinctions between learning algorithms. Neural Comput. 8, 1341–1390.
Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. Ieee Trans. Evol. Comput. 1, 67–82.