Descriptive Image Gradient from Edge-Weighted Image Graph and Random Forests

Raquel Almeida
ImScience and IRISA
PUC Minas and Université de Rennes 1
Brazil and France
raquel.almeida.685026@sga.pucminas.br

Zenilton K. G. Patrocínio Jr
ImScience
PUC Minas
Belo Horizonte, Brazil
0000-0003-0804-1790

Arnaldo de A. Araújo
Computer Science Department
Universidade Federal de Minas Gerais
Belo Horizonte, Brazil
arnaldo@dcc.ufmg.br

Ewa Kijak
Linkmedia, IRISA
Université de Rennes 1
Rennes, France
ewa.kijak@irisa.fr

Simon Malinowski
Linkmedia, IRISA
Université de Rennes 1
Rennes, France
simon.malinowski@irisa.fr

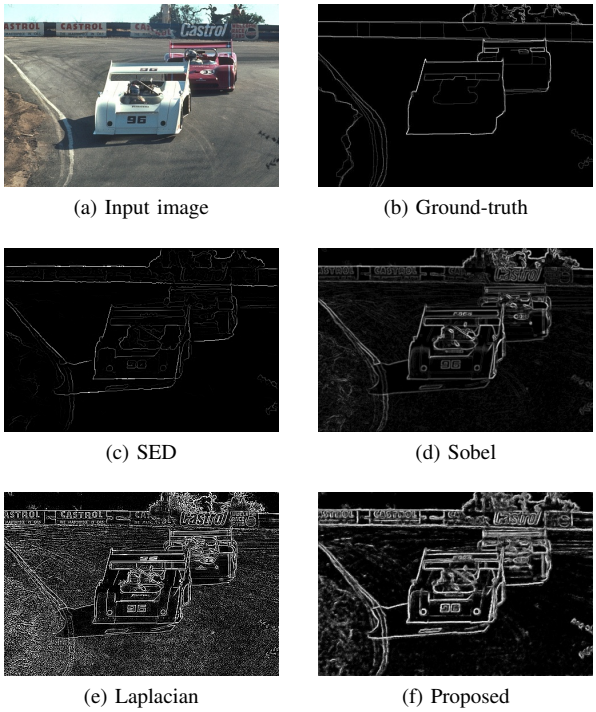
Silvio Jamil F. Guimarães
ImScience
PUC Minas
Belo Horizonte, Brazil
0000-0001-8522-2056

***Abstract*—Creating an image gradient is a transformation process that aims to enhance desirable properties of an image, whilst leaving aside noise and non-descriptive characteristics. Many algorithms in image processing rely on a good image gradient to perform properly on tasks such as edge detection and segmentation. In this work, we propose a novel method to create a very descriptive image gradient using edge-weighted graphs as a structured input for the random forest algorithm. On the one side, the spatial connectivity of the image pixels gives us a structured representation of a grid graph, creating a particular transformed space close to the spatial domain of the images, but strengthened with relational aspects. On the other side, random forest is a fast, simple and scalable machine learning method, suited to work with high-dimensional and small samples of data. The local variation representation of the edge-weighted graph, aggregated with the random forest implicit regularization process, serves as a gradient operator delimited by the graph adjacency relation in which noises are mitigated and desirable characteristics reinforced. In this work, we discuss the graph structure, machine learning on graphs and the random forest operating on graphs for image processing. We tested the created gradients on the hierarchical watershed algorithm, a segmentation method that is dependent on the input gradient. The segmentation results obtained from the proposed method demonstrated to be superior compared to other popular gradients methods.**

I. Introduction

Image segmentation may be considered as a semantic task and it is an active topic of research [1]. This task consists in partitioning perceptually similar pixels into sets of regions representing areas of interest. Usually this task is done in two stages: (i) the extraction of image characteristics that facilitates interpretation and further analysis; and (ii) the mapping of these characteristics into coherent regions. A coherent region is a subjective concept, but according to [2], it must present characteristics such as: (i) uniformity; (ii) continuity; (iii) contrast between adjacent regions; and (iv) well-defined boundaries. Independently on how well-designed a mapping method is, most of them are limited by the characteristics extracted on the first stage. For instance, taking the grey-level contrast on the first stage produces a great variation between regions, but have absolute values very distinct, making it harder to determine which value actually represents a region change.

Image gradients are known to facilitate the analysis by enhancing desirable properties, extracting structural elements of an image usually based on pixel intensities. Fast and



(a) Input image (b) Ground-truth

(c) SED (d) Sobel

(e) Laplacian (f) Proposed

Fig. 1. Examples of gradient computation, for the input image (a). In (c), the SED gradient presents reinforced fuzzy borders of the main objects and small details are in large ignored. Sobel (d) presents very thin edges for both large and small objects, while large uniform regions, such as the asphalt and vegetation are discretely represented. For Laplacian in (e), it is perceived a large amount of noise for objects, edges and patterns. The proposed in (f) computed enhanced borders for both large and small objects and image textures are firmly represented with different simplified patterns.

inexpensive to compute, they are commonly used as a pre-processing step in multiple applications, such as medical analysis [3], [4], text extraction [5], video processing [6], [7] and segmentation [8], [9]. Even with the advent of deep networks, gradient use continues to be relevant due to its performance. Also, the gradients are used as support for some networks, providing enhanced features or reducing computation complexity [10]–[12]. Traditional gradient methods, such as Laplacian and Sobel, are kernel filters for local variation, highlighting the borders of objects and are usually very sensitive to abrupt changes on the original image. In 2014, [13] proposed a method for structured edge detection (SED) that is fast and precise to predict object edges, and became a common approach as image gradient creator for the segmentation task. SED extends the Random Forest (RF) [14] formalism to a general structured output space, using local patches to map similar structured labels to the same discrete label. The gradients produced by SED have enlarged fuzzy borders of the main objects present on the image, while small details and other regions are in large omitted. In Fig. 1, we illustrate the gradient computed by SED, Sobel and Laplacian (Fig. 1 (c), (d) and (e) respectively) from the input image in Fig. 1(a).

The kernel methods and SED are widely used [3]–[12], achieving their goal of enhancing the borders and partially the contrast between regions. In this work, we argue that although the borders constitute an important characteristic of the objects depicted in images, other properties that reflect the uniformity, homogeneity and continuity are also important for the interpretation of coherent regions, particularly on the task of image segmentation.

To reflect these characteristics, we propose a novel method to create gradients that firmly depicts edges of large and small objects as well as uniform regions and patterns on the image, making it a very descriptive image gradient. It is proposed to use edge-weighted graphs aggregated with the Random Forest (RF) [14]. This approach is motivated because the spatial connectivity of the pixels of an image gives us a structured representation of a grid graph, creating a particular transformed space close to the spatial domain of the images, but strengthened with relational aspects. The edge-weighted graph as an image gradient operator acts as a transformation filter on the image, but as in the case of many spatial filters based on local differences, it is subjected to respond strongly to noise. We expect that the attribute selection and implicit regularization process [15] on the RF trees can mitigate this aspect while reinforcing desirable characteristics.

As in many machine learning algorithms, RF requires a systematic input and a strategy must be placed to deal with the dynamic structure of the graphs. The goal is to maintain the main characteristics of the graph—*e.g.*, topology, relationships, essential features—without loosing too much information. A popular approach is graph embedding [16]–[21] that creates good vectorial representations, but is extremely expensive for image graphs in terms of computational resources and time.

Alternatively to graph embedding, authors working with graphs as input for the RF also proposed to create a regular

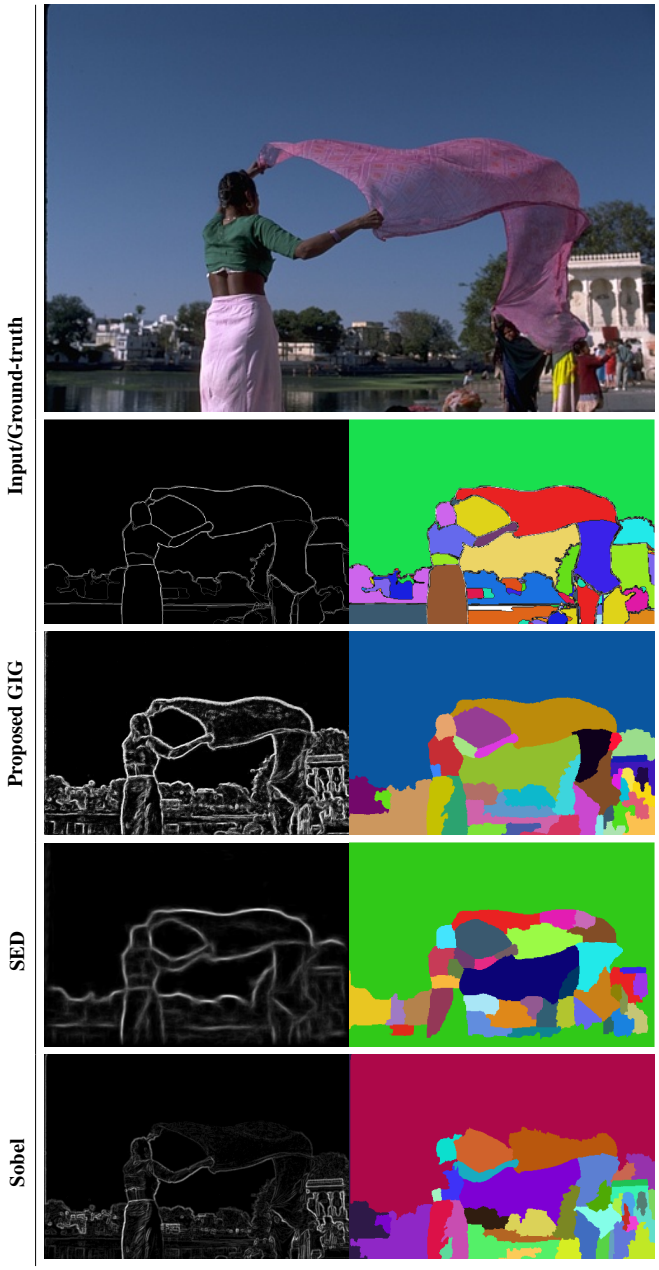


Fig. 2. Illustration of the proposed gradient (GIG) applied as pre-processing step to the task of image segmentation. GIG assisted the segmentation method to retrieve most of the delicate tissue, the main person with integrity and small details like the smaller person, vegetation and building. Presented in comparison with other popular gradient methods, SED and Sobel, that do not retrieve as successfully the same components.

representation by the use of: (i) graph adjacency matrix [22]; (ii) graph topology measures (*e.g.*, centrality, community) [23], [24]; (iii) selection of attributes [25], [26]; (iv) pairwise comparison of graph components [27]; and (v) feature inference methods [28]. We propose to make a selection of attributes that preserves the main features of the original image and incorporates the relational aspects of the graph.

The main contributions of this work are three-fold:

1) The use of graph representations of images as input of Random Forests, in which: (i) the regions of analysis are

delineated in the neighborhoods inside the graph; (ii) the feature space is defined in terms of graph attributes; and (iii) the discrete label attribution is centered on the components, therefore the entirety of a neighborhood is assigned to a single label. It aims to keep the main features of the original image and the information about relational aspects of the graph. Also, the standard discrete label is considerably faster to train than the complex structured output space mapping in SED—to be precise, experiments training each tree of the proposed method and SED took approximately 2 seconds for the former and 4 hours for the latter, using the same CPU.

2) A straightforward strategy to create a systematic input to implement machine learning on grid graphs, a topic of great interest recently due to: (i) its autonomy—the source data becomes virtually irrelevant once the learning system operates on graphs; (ii) the multiple possibilities of applications; and (iii) the graph capacity to represent multivariate information.
3) The description of how RF could be used for image processing by mapping the RF predictions back to the image space in the form of very descriptive image gradients. The quality of the obtained gradients is assessed qualitatively and quantitatively on a segmentation task. Some segmentation samples are presented in Fig. 2, obtained from the proposed method, SED and Sobel.

From now on, we will refer to the proposed method as **graph-based image gradient (GIG)**, illustrated in Fig. 1(e).

This work is organized as follows. In Section II, we give a brief description of graphs, their components and terminologies. An overview on RF and the proposed strategy to apply the graph structured input with attribute selection for learning are given in Section III. In Section IV, we provide some experimental results to demonstrate how these proposed gradients could be successfully used to improve segmentation results of one well-known segmentation algorithm, called hierarchies of watershed [29], in contrast with other popular methods for the gradient, illustrated in Fig. 1. And finally, in Section V, we draw some conclusions.

II. Edge-Weighted Image Graphs

The main concern in graph theory is the interconnection of objects, depicting many data. In this section, we provide a brief description of its components and terminologies, mostly following the notations in [30].

Definition 1. *A (undirected) **graph** $G = (V, E)$ consists of a finite nonempty set of vertices, denoted by V, and a finite set of edges $\{\{u, v\} \mid u, v \in V\}$, denoted by $E \subseteq V \times V$.*

The notion of vertices relates to the representation of the basic components of the data and the edges to the connections and dynamic between them. Furthermore, multiple functions could be associated with each vertex and/or edge, in order to enhance the relational aspects, interpretation of different adjacency relations and insertion of metric properties.

Definition 2. *An **edge-weighted** graph could be denoted by $(G, \mathcal{F})$ in which $\mathcal{F} : V \times V \to \mathbb{R}$ is a function that weights the edges of G. The set of all functions that could be used to weight the edges of a graph is denoted by $\mathcal{F}(E)$.*

The preserved characteristics on the graph depend on the nature of $\mathcal{F}$ and the problem of selecting a function to weight an edge could be considered as a problem of measuring the similarity between two finite sets of points.

Definition 3. *The set E induces a unique **adjacency** relation Γ on V, which associates $u \in V$ with $\Gamma(u) = \{v \in V | (u, v) \in E\}$. Γ is reflexive $((u, u) \in \Gamma(u))$ and symmetric $((u, v) \in \Gamma(u) \iff (v, u) \in \Gamma(v))$.*

The discussions about graph creation and manipulation can be made generic enough to model any data, but, for instance, we are interested on image graphs, in which the spatial connectivity of the pixels gives us a structured representation of a grid graph, close to the spatial domain and strengthened with relational aspects. For the image graph G defined on the image domain, the adjacency relation Γ between the pixels is typically obtained by a structured adjacency relation, such as 4- or 8-adjacency in a grid form, and the set of vertices $V = \{v_1, v_2, \ldots, v_N\}$ represents the N pixels of the image.

The set of functions associated with each vertex is denoted by $f : V \subset \mathrm{Z}^2 \to \mathbb{R}$. Common functions in f include low-level descriptors, variations in the color space or in the gray-scale magnitudes. The latter being notably important as the most common source to calculate the weighting function.

For the set of weighting functions $\mathcal{F}(E)$ of $G(V, \mathcal{F})$, the best candidates are the ones that could characterize similarities, and for such, the Euclidean distance is the most common, defined in E as $\boldsymbol{\mathcal{F}_{\mathbf{euc}}}(u, v) = \sqrt{(f(u) - f(v))^2}$. The edge weights may represent the local variation around a vertex, and serve as an image gradient operator bounded by the adjacency relation. The interaction between the image data and the preserved characteristics on the edge-weighted graph is conditioned by the topology choices, such as the adjacency relation and the properties of the weighting function.

III. Image Gradients from Random Forest Predictions

Weighting edges as an image gradient operator, like many gradient operators, acts as a transformation filter on the image creating a transformed space by changing the contrast of the original image and spreading the intensity levels.

Definition 4. *The graph-based gradient operator for edge-weighted graph $(G, \mathcal{F})$ at vertex u could be defined as:*

$$\nabla_{\mathcal{F}} f(u) = (\partial_{v_1} f(u), \ldots, \partial_{v_k} f(u)), \forall v_i \in \Gamma(u) \qquad (1)$$

where $\partial_v f(u)$ is the edge derivative of f at a vertex $u \in V$ along the edge $e = (u, v) \in E$:

$$\partial_v f(u) = \left.\frac{\partial f}{\partial e}\right|_u = \mathcal{F}(u, v) \qquad (2)$$

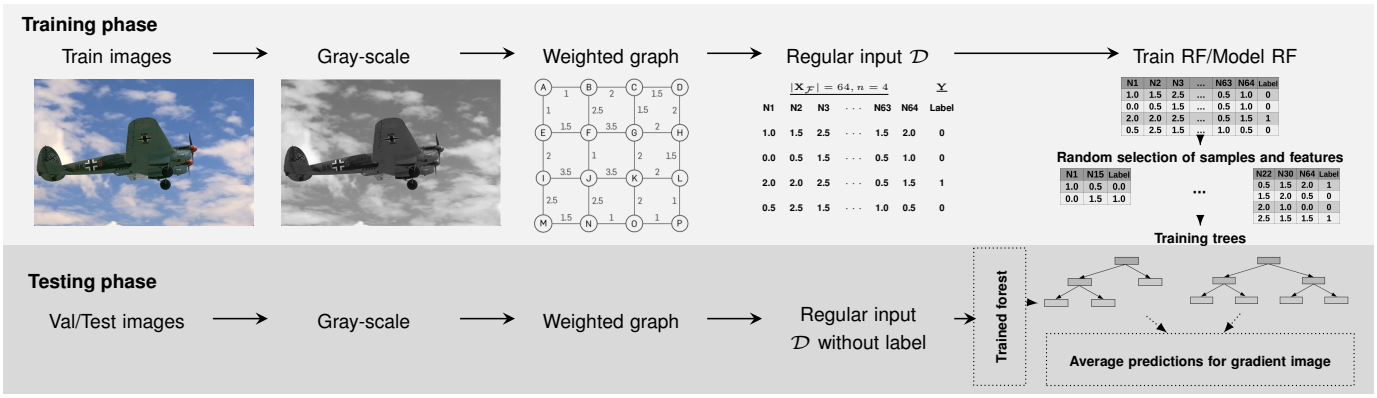


Fig. 3. Proposed framework steps from the input image to the Random forest predictions to compute the gradients. From the set of input images, we transform each image to gray-scale magnitudes, in order to compute the weights for vertices associated with each pixel of the image. The edge-weighted graphs created are grid graphs, here illustrated with the 4-adjacency relation for simplicity. The next step transform the graph structure to a regular representation with the selected attributes, in order to serve as input for the Random Forest model. The regular input for the training set includes the label associated with each vertex created using the unique discrete label on the edge detection ground-truth. During test, each vertex of the test graphs is subjected to the Random Forest prediction, where the estimated values are mapped back to the image coordinates as an intensity value to create the image gradient.

As in the case of many spatial filters based on local differences, the graph-based gradient operator defined for $\mathcal{F}_{\mathbf{euc}}$ is subjected to respond strongly to noise. We expect that the attribute selection on the RF trees can mitigate this aspect and also any eventual poor topology choice while reinforcing desirable characteristics.

A. Random Forest as regularizers

A RF is a non parametric machine learning method that can be used both for classification and regression. The RF predictor consists of M randomized trees. The core of RF algorithm, as proposed by [14], is the randomization of sampled data distributed to supervise the training of independent decision trees, and the aggregation of the results for the final prediction.

In each internal node k of a tree in the forest, there is a split function $h(\mathbf{x}, \theta_k)$ for a query point $\mathbf{x}$ with parameters θ_k. During training, the parameters θ_k are learned, usually by maximizing the information gain I_k to split the data samples covered by k into two subsets with the maximum proportion of instances belonging to the same label. On the test phase, an unseen set of data is applied to h at each split node and the result of the test determines the path the data will perform until it reaches a terminal node with the label prediction.

RF are empirically successful in suppressing noise, although the statistical and mathematical properties of the procedure are still obscure [31]. Some consensus is that the randomness in RF performs as an implicit regularization process, behaving as interpolating classifiers that encourage large consistent regions and reduce the effect of noise [15].

B. Applying Random Forest to edge-weighted graphs

To use the RF implicit regularization process with the local variation representation of the edge-weighted graph, we propose to use the information on the graph edges and vertices to represent the graph on the framework. We represent the regular input of the RF as:

Definition 5. *$\mathcal{D}_n = ((\mathbf{X}_1, \mathbf{Y}_1), \ldots, (\mathbf{X}_n, \mathbf{Y}_n))$ with $n \subseteq |V|$ samples of vertices of the edge-weighted graph $(G, \mathcal{F})$, each represented as a vector $\mathbf{X} \in \mathbb{R}^p$ and label $\mathbf{Y}$.*

In our application, edge-weighted graphs are created from images, each vertice thus corresponds to a pixel. $\mathbf{X}$ is a vector with dimension $p = |\mathbf{G_{att}}|$ for $\mathbf{G_{att}}$ representing a set of selected attributes of the vertices of $(G, \mathcal{F})$. In this work, the selected attributes belong to two categories:

- **vertex attributes ($\mathbf{X}_V$)**, belonging to the set of vertices functions f. Each $v \in V$ is mapped into a set of low-level color descriptors proposed in [32]: from RGB colors of an image pixel, 3 color channels in CIE-LUV color space, 2 normalized gradient magnitude channels and 8 gradient orientation channels are calculated;
- **edge weights ($\mathbf{X}_\mathcal{F}$)**, for a given vertex v and all its adjacent vertices, it is represented by the set of edge weights between them. Therefore $\mathbf{X}_\mathcal{F} = \{\mathcal{F}_{\mathbf{euc}}(u, v) \mid \forall u \in \Gamma(v)\}$. In this work, we go further the immediate neighbours of v and include also the neighbours in the adjacency of the immediate neighbours. Therefore,

$$\mathbf{X}_\mathcal{F} = \{\mathcal{F}_{\mathbf{euc}}(u, v), \mathcal{F}_{\mathbf{euc}}(w, u)\}$$

for all $u \in \Gamma(v)$ and $\forall w \in \Gamma(u)$.

We thus end with $\mathbf{X} = \mathbf{G_{att}} = \{\mathbf{X}_V \mathbf{X}_\mathcal{F}\}$, by concatenating the two sets of selected attributes.

C. Using Random forest to compute gradients

In order to obtain gradients, RF is trained on an edge detection task. Because each vertex of the graph is created from a pixel of the image with a unique label on the ground-truth, all the n entries $\mathbf{X}$ have a unique discrete label $\mathbf{Y} \in \{0, 1\}$ on the task of edge detection.

To obtain the image gradient on this framework, all vertices of a test graph are subjected to the estimations of the RF trained on the input data sample $\mathcal{D}_n$. At inference, instead of

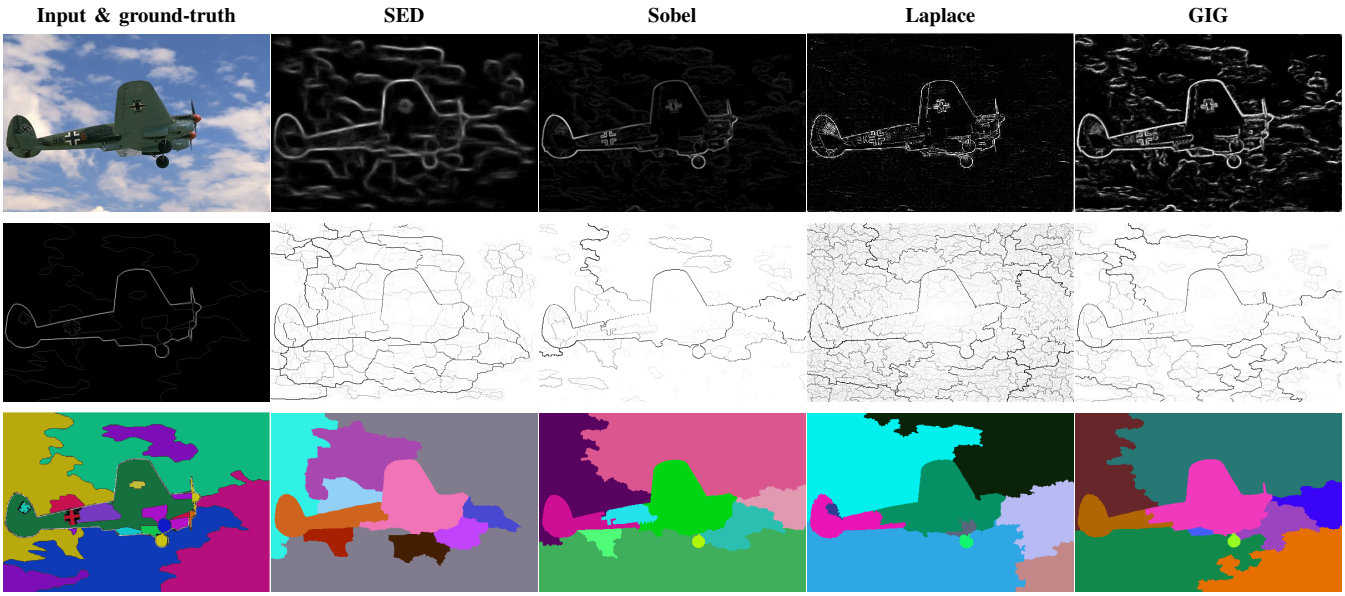


Fig. 4. 1st row: Input image and the gradients created by the compared methods. 2nd row: Boundary ground-truth and watershed hierarchies represented as saliency maps that allow us to visualize the hierarchy of regions. 3rd row: Segmentation ground-truth and the segmented images with 10 regions as criterion.

taking the label prediction, we use the RF as a regression estimator, producing an estimate $m_{M,n}$ from the M trees composing the RF.

Following the notations presented in [33], for a query point $\mathbf{X}$, the j-th tree in M gives the estimate $m : \mathcal{X} \rightarrow \mathbb{R}$:

$$m(\mathbf{X}; \Theta_j, \mathcal{D}) = \sum_{i \in \mathcal{D}^*(\Theta_j)} \frac{\mathbf{1}_{\mathbf{X}_i \in A(\mathbf{X}; \Theta_j, \mathcal{D})} \mathbf{Y}_i}{N(\mathbf{X}; \Theta_j, \mathcal{D})}, \qquad (3)$$

where:

- Θ_j is the random variable for re-sampling and selecting the split directions;
- $\mathcal{D}^*(\Theta_j)$ is the set of sampled data points for the tree construction;
- $A(\mathbf{X}; \Theta_j, \mathcal{D})$ is the tree terminal node containing $\mathbf{X}$;
- $N(\mathbf{X}; \Theta_j, \mathcal{D})$ is the number of data points falling in this terminal node.

The final estimated value, $m_{M,n}(\mathbf{X})$, obtained by averaging the M estimates $m(\mathbf{X}; \Theta_j, \mathcal{D})$ is thus taken as a confidence value that a certain vertex $\mathbf{X}$ indeed represents an edge. These estimated values for vertices are mapped back to the image coordinates as an intensity value to create the image gradient.

An illustration of the proposed framework is provided in Fig. 3 with simplified examples.

IV. Experiments

We evaluate the proposed GIG both qualitatively and quantitatively, and compared it to the widely used gradients from SED, Sobel and Laplacian methods. We perform the evaluation through a hierarchical image segmentation task performed on the Berkeley Segmentation Dataset and Benchmark (BSDS500) [34], consisting of 500 RGB images (200 train, 100 validation and 200 test), each with human labeled segmentations and boundaries. For the qualitative analysis, we provide some resulting gradients and segmentation images and discuss the impact of the different gradients in hierarchical image segmentation. For the quantitative analysis, we use two image partition interpretation metrics on the BSDS dataset.

A. Experimental setup

From the input images, the graphs are created as undirected graphs with structured 8-adjacency relation. For the vectorial representation of the vertex attributes, we explore the low-level descriptors discussed in Sec. III, therefore $|\mathbf{X}_V| = 13$. For the weighted edges attributes, we use $\mathcal{F}(E) = \{\mathcal{F}_{\mathbf{euc}}\}$ to weight the 8 direct adjacent vertices of a certain vertex and each of their subsequent neighbors, therefore $|\mathbf{X}_{\mathcal{F}}| = 64$. Not all values are unique in this representation as the vertices on the path share some neighbours. For the vertices that do not have all neighbors considered on the path, such as the ones created from the pixels on the border of the image, we added a padding value to the missing neighbors.

The training set $\mathcal{D}_n$ is composed of $n = 7,720,050$ vertices, corresponding to a balanced sub-sample of 25% of all vertices from the 200 training images, each labeled with the pixel ground-truth label for the boundaries. We explored training parameters on the validation set. The RF is trained with $M = 150$. All the 77 input features of $\mathbf{X}$ are considered at each split during training (no random feature selection). The quantitative results are presented on the test set, the gradient images being created by the estimated values predicted by the trained RF.

In this work, we do not propose a segmentation approach, we present instead a strategy to extract image characteristics that facilitate interpretation and further analysis. Therefore, in order to evaluate the quality of the gradients, we propose to apply the compared methods on the watershed hierarchies [29].

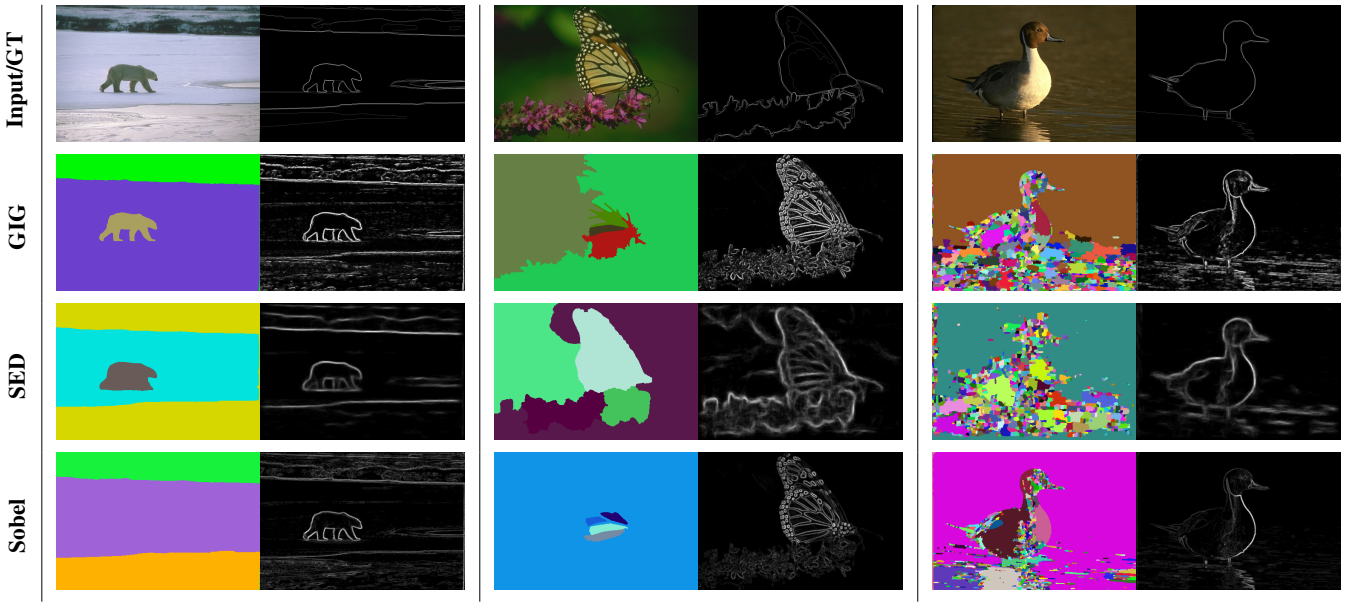


Fig. 5. Segmentation results obtained from GIG, SED and Sobel gradients with varying number of regions (Left: 3, middle: 5, right: 1000).

This intuitive algorithm maps image gradients to segmentation and its performance depends on the gradient input, making it the ideal candidate to evaluate our approach. For the segmentation step, we have used a hierarchical segmentation: the watershed by area. It is worth to mention that, thanks to this hierarchical structure, it is very easy to compute segmentation with an exact number of regions, for instance, from 2 to 5000 regions. This allows us to analyze a small number of regions closer to the ground-truth, as well as a medium number of regions for region consistency, and very large number of regions (1000 and 5000), in which results are similar to a super-pixel segmentation method.

B. Qualitative analysis

We present in Fig. 4 the gradient images obtained from the compared methods for the input images on first row. As SED is a method for edge detection, it generally produces gradient images with soft edges close to the ground-truth boundaries, which guaranties its success on the edge detection task. Nonetheless, other aspects present on the input image, such as textures and small details, are wildly ignored. Sobel present more details, without big distinction (in terms of magnitude of values) for components other than the main object. Laplace in turn is permeated by noise on the object and background (see Fig. 4, 1rst row, for computed gradients). It is important to consider that Sobel and Laplacian depend on parameters definition, such as kernel size. For Sobel we represent the gradient magnitude with the L_2 norm and kernel of size of 3 calculated from the gray-scale image. For the Laplacian we represent the zero-crossing with threshold at 0.04 of maximum value. For the proposed GIG, we have a balance between highlighted strong edges, and different textures and uniform regions presented with homogeneous values distinguishing them.

In the Fig. 4, 2nd row, we present visual representations of the watershed hierarchies created from the gradients. The hierarchies are presented as saliency maps (discussed in [29]) that allow us to visualize and understand the hierarchies. With the saliency maps, we can see the regions of importance mapped by the watershed method, indicating the strength and limitations for the final segmentation. As watershed is a hierarchical model, the regions are stable and causal, meaning that no new region is created or removed, only merged and split, depending on the number of regions criteria. Therefore, the borders visualized on the saliency maps will not change their contours and their strength indicates the regions proximity. Knowing that, we can see on the 3rd row of Fig. 4 the result of the segmentation created using 10 regions as criterion. The strong contours on SED and GIG gives us a good delineation of the main object, while it invades part of the plane with Sobel. All details are lost using SED gradient, while we can recover partially the cross using Sobel, the paddle with GIG and the wheel on both. The background invades both the object and details using Laplace, as result of the noise on the gradient.

In Fig. 5, we present more examples of segmented images from GIG, SED and Sobel gradients, illustrating some variation on the number of regions, including the super-pixel effect with large number of regions (3rd column). On the 1st column, we present a successful instance of the proposed method in which the presence of strong borders and large uniform regions on the input image, captured by the GIG gradient, created a better segmentation. Using SED, the fuzzy edges limit the delineation of the main object, while the noise in Sobel prevents its detection. An observed limitation of the proposed method is presented on the 2nd column: when the input image presents objects with patterns of high-contrast, such as zebras and tigers, the detail of the GIG gradient works against the distinction of the object. This is also partially

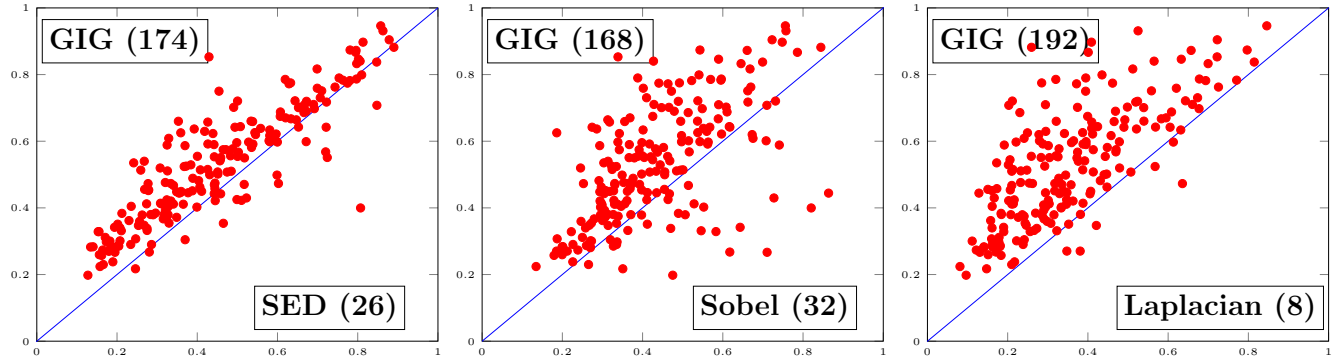


Fig. 6. Pair-wise F-measure results (red dots) on the best scale for each method, counting each image on the test set. The values in the boxes is the number of images that are better for a particular method. Proposed is better than all compared method, with statistical significance (p-value less than 10e-17)

TABLE I
F-MEASURES FOR REGIONS AND PRI PRESENTED IN TERMS OF THE OPTIMAL DATASET SCALE (ODS), OPTIMAL IMAGE SCALE (OIS) AND AVERAGE PRECISION (AP) THROUGH ALL SCALES. PERFECT SCORE=1.

	F-measure for regions			**PRI**
Gradient	**ODS**	**OIS**	**AP**	**ODS**
GIG	**0.620**	**0.688**	**0.507**	**0.786**
SED	0.559	0.617	0.477	0.746
Sobel	0.579	0.655	0.481	0.742
Laplacian	0.511	0.583	0.476	0.741

observed on Sobel, but not with the SED gradient, in which the pattern details are softly represented inside the object. On the super-segmented images in the 3rd column, once again the soft edges on SED works does not produce a good segmentation, while large regions with Sobel are indistinct, producing a lot of very small regions on the main object with little to none on large parts of the duck, the water and the shadow.

In general, the observed results of the proposed framework are very descriptive image gradients in which: (i) object boundaries are highlighted (including the one from very small components); (ii) image textures are firmly represented with different simplified patterns; and (iii) large regions are uniform with distinction of shadow regions. Limitations are perceived for images with objects with patterns of high-contrast.

C. Quantitative analysis

In terms of training time, in the standard discrete label and the graph attribute selection on GIG, we trained all the 150 trees on the RF in less than three minutes, while for SED, in the same CPU, each tree (of eight trees for the presented results) takes approximately four hours. The inference for all the compared methods are similar: a fraction of a second for each image.

For the quantitative metrics, we use two types of image partition interpretation measures, as categorized and defined in [35]: (i) Precision-recall for regions, using a pixel-wise comparison for an overall performance in terms of F-measure; and (ii) Probabilistic Rand Index (PRI), a pixel-wise measure that takes into account the multiple ground-truths presented for each image on the BSDS500 dataset.

Results for both metrics are presented in Table I. The F-measure results for regions are presented in terms of the optimal dataset scale (ODS), optimal image scale (OIS) and average precision (AP) through all scales, and the PRI in terms of ODS. The superior results of GIG on all metrics indicate that the strong borders combined with the uniform region information have a positive impact on the hierarchies of watershed segmentation. Finally, in Fig. 6, we present a pair-wise comparison of individual images on the best scale of each compared method. As one can see, the proposed GIG produces considerably more better segmented images and are all statistically significant (p-value less than 10e-17).

V. CONCLUSIONS

In this work, we explored the outcomes of a novel framework operating on an edge-weighted graph coupled with Random Forest estimates to create very descriptive image gradients. We also outlined the challenges of machine learning on graphs and proposed a strategy to create a systematic input for the random forest framework from the key attributes in an edge-weighted image graph. A qualitative analysis of the produced gradients showed that the proposed method produces gradients where boundaries are highlighted, including very small components, image textures, large uniform regions and distinction of shadow regions. The proposed method and other popular gradient methods were used as input for the watershed hierarchies segmentation method that relies on a good image gradient as input. A quantitative analysis of the produced segmentations confirmed the visual results and demonstrated that the proposed gradient is a better candidate to create image gradients for segmentation. Finally, the proposed approach on the structured input proved to be not only descriptive, but also considerably faster to train than the structured output. For further works, we will study the behaviour of GIG with different hierarchical methods, moreover, we will apply our gradient to region adjacency graphs.

ACKNOWLEDGMENT

The authors thank the Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq – (PQ 310075/2019-0), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES – (Grant STIC-AmSUD TRANSFORM 88881.143258/2017-01 and COFECUB 88887.191730/2018-00), Fundação de Amparo a Pesquisa do Estado de Minas Gerais – FAPEMIG – (Grants PPM-00006-18), PUC Minas and INRIA Associate Team program under the project *Learning on graph-based hierarchical methods for image and multimedia data*. Furthermore, this study was also financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

REFERENCES

[1] H. Mittal *et al.*, "A comprehensive survey of image segmentation: clustering methods, performance parameters, and benchmark datasets," *Multimedia Tools and Applications*, pp. 1–26, 2021.

[2] D. Domínguez and R. R. Morales, *Image segmentation: advances*. Magnum Publishing LLC, 2016, vol. 1.

[3] A. Soni and A. Rai, "Automatic cataract detection using sobel and morphological dilation operation," in *Proceedings of Research and Applications in Artificial Intelligence*. Springer, 2021, pp. 267–276.

[4] M. M. Lakshmi and P. Chitra, "Tooth decay prediction and classification from x-ray images using deep cnn," in *2020 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2020, pp. 1349–1355.

[5] H. Jeong, Y.-C. Choi, and K.-S. Choi, "Parallelization of levelset-based text baseline detection in document images," in *2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC)*. IEEE, 2021, pp. 48–51.

[6] W. T. Honeycutt and E. S. Bridge, "Uncanny: Exploiting reversed edge detection as a basis for object tracking in video," *Journal of Imaging*, vol. 7, no. 5, p. 77, 2021.

[7] S. Eetha, S. Agrawal, and S. Neelam, "Zynq fpga based system design for video surveillance with sobel edge detection," in *2018 IEEE International Symposium on Smart Electronic Systems (iSES)(Formerly iNiS)*. IEEE, 2018, pp. 76–79.

[8] A. Aslam, E. Khan, M. Samar Ansari, and M. Sufyan Beg, "A novel falling-ball algorithm for image segmentation," *arXiv e-prints*, pp. arXiv–2105, 2021.

[9] A. Z. Junejo, S. A. Memon, I. Z. Memon, and S. Talpur, "Brain tumor segmentation using 3d magnetic resonance imaging scans," in *2018 1st International Conference on Advanced Research in Engineering Sciences (ARES)*. IEEE, 2018, pp. 1–6.

[10] P. Naveen and P. Sivakumar, "Adaptive morphological and bilateral filtering with ensemble convolutional neural network for pose-invariant face recognition," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1–11, 2021.

[11] Z. Tu, K. L. Narr, P. Dollar, I. Dinov, P. M. Thompson, and A. W. Toga, "Brain anatomical structure segmentation by hybrid discriminative/generative models," *IEEE Transactions on Medical Imaging*, vol. 27, no. 4, pp. 495–508, 2008.

[12] L. Prabaharan and A. Raghunathan, "An improved convolutional neural network for abnormality detection and segmentation from human sperm images," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 3, pp. 3341–3352, 2021.

[13] P. Dollár and C. L. Zitnick, "Fast edge detection using structured forests," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 8, pp. 1558–1570, 2014.

[14] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[15] A. J. Wyner, M. Olson, J. Bleich, and D. Mease, "Explaining the success of adaboost and random forests as interpolating classifiers," *The Journal of Machine Learning Research*, vol. 18, no. 1, pp. 1558–1590, 2017.

[16] M. F. Demirci and S. Kacka, "Object recognition by distortion-free graph embedding and random forest," in *IEEE 10th International Conference on Semantic Computing*. IEEE, 2016, pp. 17–23.

[17] T. Karunaratne and H. Boström, "Graph propositionalization for random forests," in *International Conference on Machine Learning and Applications*, 2009.

[18] M. Ou, P. Cui, J. Pei, Z. Zhang, and W. Zhu, "Asymmetric transitivity preserving graph embedding," in *22nd ACM International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1105–1114.

[19] A. Temir, K. Artykbayev, and M. F. Demirci, "Image classification by distortion-free graph embedding and KNN-random forest," in *17th Conference on Computer and Robot Vision*. IEEE, 2020.

[20] X. Wang, Y. Gong, J. Yi, and W. Zhang, "Predicting gene-disease associations from the heterogeneous network using graph embedding," in *IEEE International Conference on Bioinformatics and Biomedicine*. IEEE, 2019, pp. 504–511.

[21] S. Zhou *et al.*, "LncRNA-miRNA interaction prediction from the heterogeneous network through graph embedding ensemble learning," in *International Conference on Bioinformatics and Biomedicine*, 2019.

[22] J. Liang and D. Huang, "Laplacian-weighted random forest for high-dimensional data classification," in *IEEE Symposium Series on Computational Intelligence*. IEEE, 2019, pp. 748–753.

[23] B. Ayotte, M. Banavar, D. Hou, and S. Schuckers, "Fast free-text authentication via instance-based keystroke dynamics," *IEEE Transactions on Biometrics, Behavior, and Identity Science*, pp. 377–387, 2020.

[24] K. Sharad and G. Danezis, "An automated social graph de-anonymization technique," in *13th Workshop on Privacy in the Electronic Society*, 2014, pp. 47–58.

[25] J. Kim and J. Lee, "Adaptive directional walks for pose estimation from single body depths," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2020, pp. 1–6.

[26] S. Pachaury, N. Kumar, A. Khanduri, and H. Mittal, "Link prediction method using topological features and ensemble model," in *11th International Conference on Contemporary Computing (IC3)*. IEEE, 2018.

[27] M. Guillame-Bert and A. Dubrawski, "Classification of time sequences using graphs of temporal constraints," *The Journal of Machine Learning Research*, vol. 18, no. 1, 2017.

[28] Y. Gao, X. Li, J. Li, Y. Gao, and N. Guo, "Graph mining-based trust evaluation mechanism with multidimensional features for large-scale heterogeneous threat intelligence," in *IEEE International Conference on Big Data*. IEEE, 2018, pp. 1272–1277.

[29] J. Cousty and L. Najman, "Incremental algorithm for hierarchical minimum spanning forests and saliency of watershed cuts," in *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer, 2011, pp. 272–283.

[30] L. Najman and H. Talbot, *Mathematical morphology: from theory to applications*. John Wiley & Sons, 2013.

[31] E. Scornet, "Random forests and kernel methods," *IEEE Transactions on Information Theory*, vol. 62, no. 3, pp. 1485–1500, 2016.

[32] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *British Machine Vision Conference*. BMVA Press, 2010.

[33] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, no. 2, pp. 197–227, 2016.

[34] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *8th IEEE International Conference on Computer Vision*. IEEE, 2001, pp. 416–423.

[35] J. Pont-Tuset and F. Marques, "Measures and meta-measures for the supervised evaluation of image segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2131–2138.