




Efficient Multiscale Object-based Superpixel Framework


Felipe C. Belém   [University of Campinas, Pontifical Catholic University of Minas Gerais, Université Gustave Eiffel | felipebelem@pucminas.br]

Benjamin Perret  [Université Gustave Eiffel | benjamin.perret@esiee.fr]

Jean Cousty  [Université Gustave Eiffel | jean.cousty@esiee.fr]

Silvio J. F. Guimarães  [Pontifical Catholic University of Minas Gerais | sjamil@pucminas.br]

Alexandre X. Falcão  [University of Campinas | afalcao@ic.unicamp.br]

 Department of Computer Science, Pontifical Catholic University of Minas Gerais, Av. Dom José Gaspar, 500, Coração Eucarístico, Belo Horizonte, MG, 30535-901, Brazil

Received: 19 March 2024 • **Accepted:** 19 December 2024 • **Published:** 23 May 2025

Abstract Superpixel segmentation can be used as an intermediary step in many applications, often to improve object delineation and reduce computer workload. However, classical methods do not incorporate information about the desired object. Deep-learning-based approaches consider object information, but their delineation performance depends on data annotation. Additionally, the computational time of object-based methods is usually much higher than desired. In this work, we propose a novel superpixel framework which exploits object information being able to generate a multiscale segmentation on-the-fly. Our method starts off from seed oversampling and repeats optimal connectivity-based superpixel delineation and object-based seed removal until a desired number of superpixels is reached. It generalizes recent superpixel methods, surpassing them and other state-of-the-art approaches in efficiency and effectiveness according to multiple delineation metrics.

Keywords: Superpixel Delineation, Image Foresting Transform, Object Saliency Map, Image Segmentation

1 Introduction

Some algorithms partition an image into several disjoint groups of connected pixels, named *superpixels*, that share a common property (e.g., color, or texture). Thus, a superpixel-based image representation provides more contextual information than a pixel-based one while reducing the computer workload. Due to these properties, superpixel segmentation is often used as an intermediary step in medical applications (e.g., Dhore and Abin [2021]; Zhou *et al.* [2019]; Liu *et al.* [2019]), semantic segmentation (e.g., Yi *et al.* [2022]; Ding *et al.* [2023]), plant diseased-leaf detection (e.g., Zhang *et al.* [2018]), pedestrian segmentation (e.g., Yu *et al.* [2019]), among other examples.

Most methods model the superpixel segmentation problem as a pixel-clustering task. The most popular one is *Simple Linear Iterative Clustering* (SLIC), proposed by Achanta *et al.* [2012], whose efficiency and simplicity inspired several and more accurate algorithms, like those from Liu *et al.* [2018]; Li and Chen [2015]. Other methods, like *Superpixel Hierarchy* (SH) and the *Entropy Rate Superpixels* (ERS), proposed by Wei *et al.* [2018]; Liu *et al.* [2011], respectively, consider the superpixel generation problem as a graph-based approach. SH recurs to Borůvka's minimum spanning forest algorithm, while ERS maximizes the energy of a random walk in the graph's topology. Similarly, works based on *Iterative Spanning Forest* (ISF), conceived by Vargas-Muñoz *et al.* [2019], and *Dynamic ISF* (DISF), from Belém *et al.* [2020b], builds a graph from the input image and define superpixels by minimizing path-costs starting from an initial set of points, called *seeds*. As a result, path-based

approaches offer a more effective solution at the expense of presenting a slightly higher computational cost. However, all the aforementioned methods are unaware of the object of interest in the image. Thus, instead of ensuring the boundaries of interest (*i.e.*, relevant borders), they pursue the preservation of all detectable borders, including many irrelevant ones.

Recently, several deep-learning-based approaches have been proposed for generating superpixels in which the breakthrough of end-to-end trainable methods was *Superpixel Sampling Networks* (SSN), from Jampani *et al.* [2018] (further improved in the work of Yang *et al.* [2020]). These methods incorporate object information during training, but require data annotation, which may be scarce in some applications (e.g., medical and biological). Although unsupervised methods have also been proposed, exemplified from the works of Yu *et al.* [2021]; Suzuki [2020], they all present moderate boundary delineation considering different datasets, even those of the same domain (e.g., natural image datasets).

Some algorithms consider the object information represented as a monochromatic image (*i.e.*, an object saliency map) in which pixel intensity denotes an object membership value. These object-based methods use such information to favor the delineation of the object of interest at the expense of others, resulting in a more accurate superpixel segmentation. However, exhaustively exploiting saliency information can lead to higher computational cost and saliency quality dependency, as seen in *Object-based Iterative Spanning Forest* (OISF) variants proposed by Belém *et al.* [2019a, 2020a]. Conversely, *Object-based*

DISF (ODISF), from Belém *et al.* [2021], overcomes the latter drawback while still being slightly slower than object-unaware methods.

Due to SLIC's simplicity and efficiency, the vast majority of superpixel methods adopt the same three-step pipeline: (i) seed sampling; (ii) superpixel generation; and (iii) seed recomputation; in which step (i) is applied once and, subsequently, steps (ii) and (iii) are repeated for a given number of iterations. The major advantage of seed-based superpixel segmentation is that it is applicable to any object, exploiting its inherent existence in the image. Moreover, by recomputing the seeds (for the next iteration), the intra-superpixel dissimilarity and, thus, errors are minimized. Despite SLIC (Achanta *et al.* [2012]) presents fair effectiveness, recent and more accurate approaches benefit from variations of the same pipeline (*e.g.*, Li and Chen [2015]; Liu *et al.* [2018]; Vargas-Muñoz *et al.* [2019]; Xiao *et al.* [2018]).

This three-step pipeline presents several drawbacks. First, the number of seeds sampled in step (i) approximates the desired number of superpixels. While it may not choose seeds that promote accurate object delineation (*i.e.*, relevant), especially when the object is small, using object-based strategies usually leads to higher computational time, as noted by Belém *et al.* [2019b]. Second, as Wei *et al.* [2018] argues, several executions of the above pipeline are required to produce a multiscale superpixel segmentation. Third, reallocating a seed may not guarantee the accurate delineation in a previous iteration. Finally, imposing a strict number of iterations for segmentation may force the execution of unnecessary iterations, notably when no convergence is met.

A novel strategy based on the previous pipeline was recently proposed and exploited by DISF and ODISF (Belém *et al.* [2020b, 2021]). In step (i), a high initial number of seeds is selected to increase the chances of sampling relevant seeds through faster approaches. A portion of the most irrelevant seeds is removed at each iteration in step (iii), based on an exponential curve, until reaching the desired number of superpixels. Thus, the most relevant ones are rest unaltered for accurate delineation in the next iteration. As one may notice, this new approach of oversampling and iterative seed removal, by design, permits obtaining a multiscale segmentation in a single execution. Moreover, as the desired number of superpixels increases, fewer iterations are performed for segmentation. However, as a direct consequence of the experiments performed in Belém *et al.* [2020b, 2021], several assumptions raised in both works were not properly tested nor evaluated. For instance, the number of seeds sampled in step (i) is not determined by optimization but by empirical testing, which may lead to an undesired use of memory for storing such a quantity of seeds. Also, both methods consider a size and color-based relevance criterion in step (iii), which is uncertain whether size or color would be sufficient. Finally, although a strict quantity of iterations is not imposed, such a number is indirectly defined by the initial seed quantity, making it challenging to be intuitively set and possibly leading to unnecessary iterations for segmentation.

In this work, we propose a **generalization** of DISF

and ODISF, overcoming their inability to directly control the number of iterations for segmentation, assisting in the construction of effective and efficient multiscale superpixel segmentation variants. By reformulating the seed removal step, our proposal, named Previous Superpixels through Iterative CLearcutting (pSICLE), is capable of limiting the algorithm to perform a maximum number of iterations, such that it can obtain DISF's and ODISF's equivalent results if the seed removal decay is appropriately set. Moreover, by not incorporating saliency information during delineation, pSICLE is robust to incorrect saliency estimations while producing highly accurate superpixels. Since ideal saliency estimation is not yet feasible, errors are often present, and considering a limited influence of such incorrect values assists in preventing delineation errors. Finally, pSICLE can generate an object-based multiscale segmentation on the fly if the quantity of seeds at each iteration is previously defined (see Figure 1). We also extensively evaluate different configurations (including new proposals) for each pipeline's step. Results show that pSICLE can accurately delineate boundaries of various objects from distinct datasets, being faster than SLIC (Achanta *et al.* [2012]) in most practical cases.

Our main contributions may be summarized as follows:

- Seed sampling** An equidistant seed sampling strategy that distributes seeds proportionally along each image axis;
- Seed removal** Regarding seed removal, we propose: (i) new criteria that accurately maintain those promoting effective object delineation; and (ii) a procedure that establishes a maximum non-strict number of iterations for improving speed by avoiding unnecessary iterations for segmentation;
- Multiscale approach** A novel framework for the design of object-based multiscale superpixel segmentation methods;
- On-the-fly** pSICLE-based methods that are robust to object saliency errors and dismiss unnecessary segmentation iterations while generating multiscale boundaries on the fly;
- Easy-to-use** Optimization and extensive analysis of each OURS's step, leading to an easy-to-use framework with accurate and fast delineation in different image domains;

One may note the similarities between this work and several others, for instance the SICLE framework proposed by Belém *et al.* [2022, 2023]. However, we emphasize the main distinctions between them. Since we first provide the mathematical foundations of SICLE (thus the name previous SICLE) and we perform a deep and thorough analysis of all of its steps, the contributions here presented assisted in the development of a variety of effective and efficient approaches. Furthermore, since our results show equivalent delineation results for different oversampling strategies, subsequent works achieved faster and simpler seed selection strategies. Likewise, the findings regarding the equivalent performance of connectivity functions assisted in works opting for more suitable functions, which allows differential computations. Also, our results on evaluating the seed removal criterion favored the development of effective

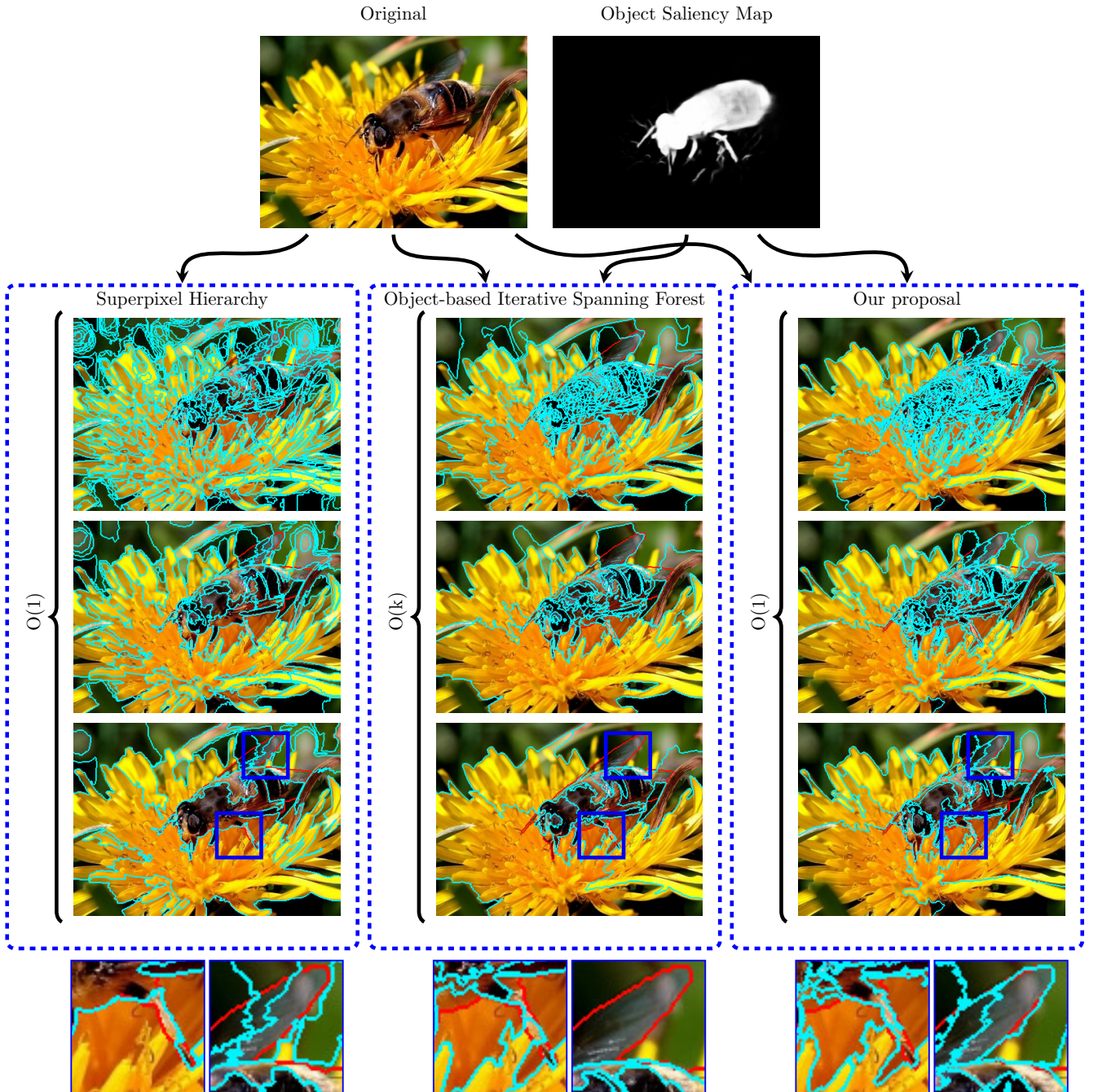


Figure 1. Multiscale superpixel segmentation comparison between two state-of-the-art methods and our proposal. It required three scales (*i.e.*, $k = 3$): 500, 100, and 25 superpixels, respectively. Our proposal achieves the best of both approaches whilst achieving superior performance. The red lines indicate the object boundary, whereas the cyan ones, superpixel borders. The object saliency map was generated from Qin *et al.* [2020].

variants that prioritize object delineation and superpixel compacity. Finally, from our ablation study in the quality of the saliency map, such works studied different approaches for improving segmentation through parametrization of the saliency influence in generating superpixels.

This paper is organized as follows. In Section 2, we present an overview of the state-of-the-art in superpixel segmentation. We describe our theoretical background in Section 3 and, subsequently, detail our proposed framework in Section 4. Finally, in Sections 5 and 6, we discuss the experimental results, and we draw conclusions and possible future work, respectively.

2 Related Works

In this section, we present some of the most recent and notable works in superpixel segmentation. One may refer to the work of Stutz *et al.* [2018] for a deeper discussion on this topic. We emphasize that our work does not include a systematic review of superpixel segmentation, and consequently, our selection of those deemed as notable is based on effectivity, efficiency, and popularity. Therefore, our goal is to compare the most effective and the most efficient amongst the methods described here. Although several approaches could be considered in the evaluation, such as the work of Ding *et al.* [2023], more

Properties	Achanta et al. [2012]	Li and Chen [2015]	Wei et al. [2018]	Tu et al. [2018] ¹	Belém et al. [2020a]	pSICLE
Boundary Recall	0.864	0.893	0.949	0.826	0.952	0.978
Under-segmentation Error	0.016	0.013	0.011	0.015	0.008	0.010
Speed (ms)	507	850	941	- ²	2832	468
Multiscale Segmentation	No	No	Yes	No	No	Yes
Object Information	No	No	No	Yes	Yes	Yes
Error Propagation	No	No	Yes	Yes	Yes	No

¹ Deep-learning-based algorithm.

² For a single segmentation, the time for loading the network exceeds OISF's.

Table 1. Average pSICLE performance compared to state-of-the-art superpixel algorithms for 750 superpixels. The algorithms were chosen amongst those presented in this section, each one representing its own category, based on their delineation performance and speed. It was considered a grayscale dataset whose images are 512×512 . The time for estimating the saliency using the proposal from Qin et al. [2020] was not computed for this table.

experiments are required to evaluate whether such an intermediary segmentation step is appropriate for general superpixel segmentation. In Section 2.1, we review those methods that do not incorporate prior object information. Subsequently, we present those that use deep-learning strategies for generating superpixels in Section 2.2. Lastly, in Section 2.3, we discuss the algorithms that consider object information represented by an object saliency map. Table 1 compares different state-of-the-art methods, chosen by their delineation performance and speed on their own category, against our proposal considering a grayscale medical image dataset.

2.1 Classical Methods

We may broadly classify the majority of the classical methods by their strategy: (i) clustering-based; (ii) graph-based; and (iii) path-based. In the first group, the superpixels are generated through pixel clustering. *Simple Linear Iterative Clustering* (SLIC), proposed by Achanta et al. [2012], is the most popular method of such group, and it uses an adapted K-means on a 5-dimensional feature space for a fast but moderate delineation. Similarly, *Linear Spectral Clustering* (LSC), conceived by Li and Chen [2015], applies a weighted K-means in a deliberately tailored 10-dimensional space for producing superpixels with high boundary adherence while slightly increasing the computational time. Some works adopt different clustering techniques for superpixel generation, such as Gaussian Mixture Models or Density-based Spatial Clustering of Applications with Noise (*i.e.*, the works from Ban et al. [2018]; Shen et al. [2016], respectively). Clustering-based methods often present low computational complexity by recurring to a restricted search scope for clustering, yielding more compact superpixels. However, such a strategy may also lead to disconnected superpixels, forcing a post-processing step for guaranteeing connectivity and not ensuring the desired number of superpixels. Furthermore, the delineation performance is fair compared to other groups of superpixel methods. Finally, none of the aforementioned approaches are able to produce a multiscale segmentation on the fly, thus requiring several executions of the same algorithm.

We may list the works of Liu et al. [2011]; Wei et al. [2018], namely *Entropy Rate Superpixels* (ERS) and the *Superpixel Hierarchy* (SH), as the most popular and the

most effective graph-based methods, respectively. Unlike clustering methods, graph-based tend to produce highly accurate superpixel delineation. In ERS, the superpixel segmentation is modeled as an optimization problem of the entropy of a random walk in the graph's topology. While ERS requires a high computational time, even when using greedy strategies, SH uses the Borůvka algorithm for generating superpixels with high boundary adherence in linear time. Since it is a hierarchical method, errors in the lower levels are propagated to the upper ones, compromising its delineation performance. Thus, although ERS could be modified to generate a hierarchy and generate a multiscale segmentation in a single execution, such error propagation may severely impact the final object delineation.

Path-based approaches define superpixels by path searching from each seed to its most similar pixels, ensuring superpixel connectivity. The methods based on the *Image Foresting Transform* (IFT) proposed by Falcão et al. [2004] algorithm often present top object delineation at the expense of not easily controlling the superpixel compacity, as Stutz et al. [2018] argue. *Iterative Spanning Forest* (ISF), from Vargas-Muñoz et al. [2019], is a three-step seed-based framework that generates superpixels through iterations of IFT executions on improved seed sets. Based on the latter, Galvão et al. [2018] proposed the *Recursive ISF* (RISF) algorithm to obtain a sparse-hierarchical superpixel segmentation with no efficiency deterioration. Recently, Belém et al. [2020b] proposed *Dynamic ISF* (DISF), which starts from seed oversampling and iteratively removes seeds based on a relevance criterion applied to the IFT-based superpixel delineations, achieving highly accurate segmentation. Although this group also contains non-IFT-based methods like the work from Achanta and Süsstrunk [2017], they usually present fair delineation performance. Similarly to graph-based approaches, path-based ones are more effective than clustering strategies but are significantly slower for generating superpixels. Aside from DISF and RISF, these methods are unable to generate a multiscale segmentation on the fly.

Some methods, inspired by clustering-based approaches, aim to increase the number of superpixels in regions characterized by higher content density (*i.e.*, *content-sensitive*). For instance, Liu et al. [2018] proposed *Intrinsic Manifold SLIC* (IMSLIC), which maps every pixel to a 2-dimensional manifold and measures the superpixel density through its area. Other examples, like the works of Xiao

et al. [2018]; Wu et al. [2021], generate content-sensitive superpixels by improving the separability in the feature space. However, unlike IMSLIC, they require a post-processing step to ensure superpixel connectivity. Like clustering-based algorithms, content-sensitive methods need to perform several executions to produce a multiscale segmentation.

2.2 Deep-learning Methods

Although classical methods often present high delineation accuracy, they are unaware of the user’s desire to segment a particular object. On the other hand, deep-learning methods are driven by the information inferred during training (e.g., important borders). For a period of time, some works, such as the one by Awaisu et al. [2019], aimed at generating superpixels using deep features, even though the benefits of this strategy are not yet solid, as argued by Tu et al. [2018]. Thus, Tu et al. [2018] proposed a novel loss function named *SEgmentation Aware Loss* (SEAL) for learning the pixel affinities for the underlying ERS algorithm. Such a method, named SEAL-ERS, was further improved in the work of Peng et al. [2022] in terms of speed. Still, as one may argue, SEAL-ERS is dependable on the quality of estimating such affinities since incorrect inferences can prejudice the entropy computation performed by its backbone method. Furthermore, by using ERS as the underlying superpixel method, SEAL-ERS inherits its drawbacks, including high computational complexity and error propagation in the hierarchy.

Another common approach is to design end-to-end trainable networks for superpixel segmentation, either supervised or unsupervised. Not only do both approaches require a significant amount of data for training, especially the former (which also needs annotations), but they often present high computational time for a single image segmentation. Also, to generate a multiscale segmentation, the user would have to perform several executions of the same algorithm, one for each scale. Moreover, if the user desires a different object of interest, one may obtain annotated data and retrain the network for parameter optimization. Note that the latter is also a drawback of SEAL-ERS and alike. We may cite the popular *Superpixel Sampling Networks* (SSN), from Jampani et al. [2018], and *Fully Convolutional Network* (FCN), from Yang et al. [2020], as examples of supervised methods. Regarding unsupervised ones, we list *Regular Information Maximization* (RIM) and *Edge-Aware Superpixel Segmentation* (EASS), proposed by Suzuki [2020]; Yu et al. [2021], respectively, as two of the most recent representatives of this group.

2.3 Object-based Methods

We term an algorithm object-based when it incorporates prior object information, often represented by object saliency maps. *Saliency-based Superpixels* (SS) from Xu et al. [2014] can be classified as such, even though such a term was first coined by Belém et al. [2019a]. Recent object-based solutions can be seen as generalizations of previous methods for handling saliency maps, such as *Object-based*

ISF (OISF), proposed by Belém et al. [2019a, 2020a]. In OISF, the user controls the superpixel displacement and morphology concerning the map’s estimation. While it is slow and it is highly dependable on the map quality (i.e., propagates the maps’ errors), Belém et al. [2021] proposed *Object-based DISF* (ODISF), which offers a highly accurate and faster solution with minimum influence of saliency errors. For SS, a major drawback involves a strict perception of saliency (incorporated within the superpixel generation procedure), which is surrogated by OISF and ODISF since the saliency is computed externally to their pipeline. Yet, aside from being highly influenced by incorrect saliency estimations, OISF requires several time-consuming executions to generate a multiscale segmentation. While ODISF can generate a multiscale segmentation, both ODISF and, especially OISF, perform unnecessary iterations for segmentation. As an example, one may need a single iteration if a gold-standard saliency map (i.e., the ground truth) is provided, independently of the number of superpixels generated.

3 Theoretical Background

In this section, we present important definitions regarding our proposal. First, we recall basic graph notions in Section 3.1 and, subsequently, in Section 3.2, we detail the *Image Foresting Transform* (IFT) Falcão et al. [2004], the core algorithm for delineation.

3.1 Image Graph

An *image* I can be defined as a pair $\langle \mathcal{P}, \mathbf{F} \rangle$ in which $\mathcal{P} \subset \mathbb{Z}^2$ is the set of *picture elements* (i.e., pixels) and \mathbf{F} maps every $p \in \mathcal{P}$ to its *feature vector* $\mathbf{F}(p) \in \mathbb{R}^m$, given $m \in \mathbb{N}^*$. If $m > 1$, I is a *colored image* (e.g., CIELAB), otherwise, I is *grayscale*. Lastly, an *object saliency map* $O = \langle \mathcal{P}, \mathbf{O} \rangle$ is a grayscale image whose feature indicates a proportional likelihood $\mathbf{O}(p) \in [0, 1]$ of a pixel p belonging to an object of interest.

An *image digraph* G derived from I is a pair $\langle \mathcal{V}, \mathcal{A} \rangle$ such that $\mathcal{V} \subseteq \mathcal{P}$ is the *vertex set* and $\mathcal{A} \subset \mathcal{V}^2$ is the set of *arcs*. Two vertices x, y are said to be *adjacents* if $\exists \langle x, y \rangle$ or $\langle y, x \rangle \in \mathcal{A}$. A common approach for establishing the arcs in \mathcal{A} is based on the Euclidean spatial distance between the vertices. Therefore, for any $x, y \in \mathcal{V}$ and a radius $r \in \mathbb{R}_+$, we may obtain the arc set $\mathcal{A}^r = \{ \langle x, y \rangle : \|x - y\|_2 \leq r \}$. As one may see, $\mathcal{A}^{\sqrt{2}}$ contains the arcs defined in an 8-neighborhood. Throughout this paper, G has neither loops nor parallel edges (i.e., G is a *simple digraph*).

We may define a (*directed*) *path* ρ as a sequence of distinct vertices $\langle v_i \rangle_{i=1}^k$ in which $\langle v_i, v_{i+1} \rangle \in \mathcal{A}$ for $i < k$. ρ is *trivial* when $k = 1$, and *non-trivial*, otherwise. For $1 < i < k$, v_i is said to be the *predecessor* of v_{i+1} and the *successor* of v_{i-1} . We may refer to v_1 and v_k as *origin* (or *root*) and *terminal* vertices of ρ , respectively. For simplicity, we can exhibit both either by $\rho_{v_1 \rightsquigarrow v_k}$ or by ρ_{v_k} . Finally, $\rho_y = \rho_x \odot \langle x, y \rangle$ denotes the path ρ_y resultant from concatenating ρ_x with an arc $\langle x, y \rangle$.

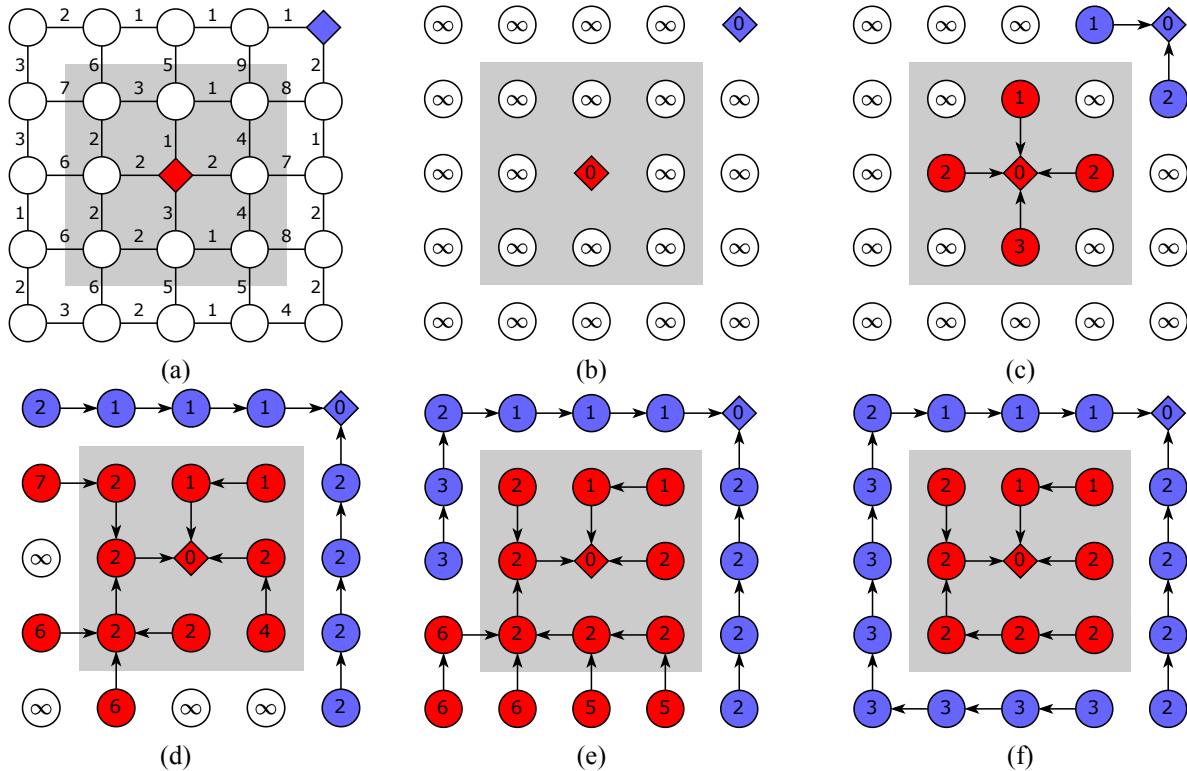


Figure 2. Example of IFT execution in a toy example considering two seeds of distinct labels (depicted by colored diamonds) and the \mathbf{f}_{\max} connectivity function. (a) Original graph with precomputed edge-costs. (b) Initialization of path-costs and mappings. (c-e) Non-subsequential IFT executions. (f) Resulting optimum-path-forest.

3.2 Image Foresting Transform

The *Image Foresting Transform* (IFT), proposed by Falcão *et al.* [2004], is an elegant framework for developing connectivity-based image operators and can be computed using a generalization of the Dijkstra’s algorithm. Due to its object delineation performance, it has been used in several applications, especially for superpixel segmentation (e.g., Martins *et al.* [2019]; Sousa *et al.* [2019]; Castelo-Fernandez and Falcão [2019]; Vargas-Muñoz *et al.* [2019]; Belém *et al.* [2020b]; Galvão *et al.* [2018]). In this work, we consider its seed-restricted variant. That is, for a given *seed set* $\mathcal{S} \subset \mathcal{V}$, the IFT finds paths with optimum cost from any seed to every $p \in \mathcal{V} \setminus \mathcal{S}$.

A *path-cost function* \mathbf{f}_* assigns a cost $\mathbf{f}_*(\rho) \in \mathbb{R}_+$ to any $\rho \in \mathcal{P}$, considering \mathcal{P} to be the set of all possible paths in G . Similarly to \mathbf{f}_* , an *arc-cost function* $\mathbf{w}_*(x, y) \in \mathbb{R}$ defines a cost to any arc $\langle x, y \rangle \in \mathcal{A}$. A popular option of path-cost function is the \mathbf{f}_{\max} function (Equation 1) due to its accurate object delineation performance, as reported in Belém *et al.* [2020b]; Bragantini *et al.* [2018]:

$$\mathbf{f}_{\max}(\langle x \rangle) = \begin{cases} 0, & \text{if } x \in \mathcal{S} \\ +\infty, & \text{otherwise} \end{cases} \quad (1)$$

$$\mathbf{f}_{\max}(\rho_x \odot \langle x, y \rangle) = \max \{ \mathbf{f}_{\max}(\rho_x), \mathbf{w}_*(x, y) \}$$

A path ρ_x is said to be *optimum* if $\mathbf{f}_*(\rho_x) \leq \mathbf{f}_*(\tau_x)$ for any other $\tau_x \in \mathcal{P}$ irrespective of its origin. As one may note, if $\mathbf{w}_*(x, y) = \mathbf{w}_*(y, x)$, one may define an *image graph* over an *image digraph*. However, several IFT-based solutions, such as the *Oriented IFT* proposed by Miranda [2014], recur to the concept of *orientation*, which weights

differently based on the direction of the arc. Thus, we opt for a more general explanation of our backbone approach for possible future endeavors.

By electing a unique optimum-path ρ_x for all $x \in \mathcal{V}$, we define the *predecessor map* \mathbf{P} as an acyclic map which assigns x either to its predecessor in ρ_x or to a distinctive marker $\blacktriangle \notin \mathcal{V}$ when x is the root of ρ_x (and, consequently, of \mathbf{P}). Note that we can recursively map x to its root $\mathbf{R}(x) \in \mathcal{S}$ through \mathbf{P} . Starting at the seeds, the IFT framework builds \mathbf{P} through path concatenation while it minimizes a *cost map* $\mathbf{C}(x) = \min_{\rho_x \in \mathcal{P}} \{ \mathbf{f}_*(\rho_x) \}$. As noted by Mansilla and Miranda [2016], even if \mathbf{f}_* does not satisfy certain properties, the resulting paths still present desirable properties for superpixel segmentation. For any $s \in \mathcal{S}$, \mathbf{P} defines an *optimum-path tree* (or, equivalently, a superpixel) $\mathbf{T}(s) = \{ x : \mathbf{R}(x) = s \}$ rooted in s whose paths are more closely connected to its seed than to any other. Finally, we can calculate its mean feature vector $\bar{\mathbf{F}}(\mathbf{T}(s))$ and mean saliency value $\mathbf{O}(\mathbf{T}(s))$ by $\sum_{x \in \mathbf{T}(s)} \mathbf{F}(x) / |\mathbf{T}(s)|$ and $\sum_{x \in \mathbf{T}(s)} \mathbf{O}(x) / |\mathbf{T}(s)|$, respectively. Figure 2 presents an example of an IFT execution in a toy example considering \mathbf{f}_{\max} , but with precomputed arc-costs.

4 pSICLE Framework

In this section, we describe our proposed framework, named *Previous Superpixels through Iterative CLEArcutting* (pSICLE). Sections 4.1, 4.2, and 4.3 detail, in this order, the steps of the pipeline shown in Figure 3, whose algorithm is described in Section 4.4.

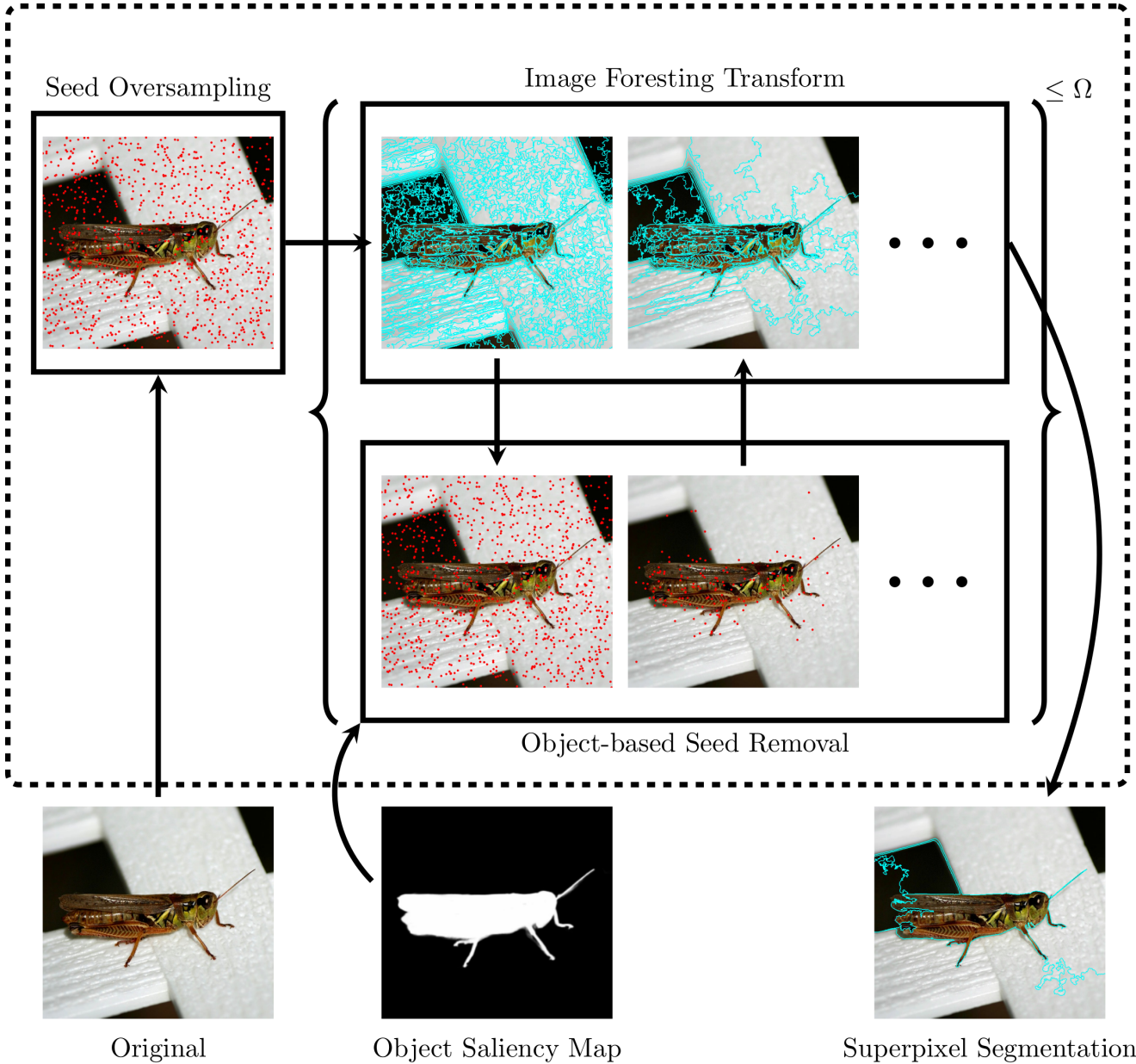


Figure 3. pSICLE framework starting from $N_0 = 1000$ random seeds and requiring $N_f = 5$ superpixels. For the IFT, the f_{\max} connectivity function and the w_2 arc-cost function were considered. And for the seed removal step, we considered the \mathbf{V}_4 restricted to \mathbf{V}_{obj} . The object saliency map was generated from Qin et al. [2020].

4.1 Seed Oversampling

The first step of pSICLE is to select N_0 initial seeds for generating N_f superpixels, given $N_0, N_f \in \mathbb{N}^*$. Most seed-based algorithms aim to build a set $\mathcal{S} \subset \mathcal{V}$ containing the seeds that promote accurate object delineation (i.e., *relevant*) using the orthodox approach of sampling $N_0 \approx N_f$ seeds, making precision crucial. However, while object-based methods use a precise but slow sampling algorithm (cf., Belém et al. [2019a,b]), non-object-based ones paradoxically opt for a fast but imprecise procedure (e.g., Achanta et al. [2012]; Liu et al. [2018]; Li and Chen [2015]; Vargas-Muñoz et al. [2019]). Thus, although precision can lead to more effective delineation, it is inversely correlated to the algorithm’s efficiency.

On the other hand, we argue that *oversampling* (i.e., $N_0 \gg N_f$) offers a better solution since it pursues recall

over precision. Given such a high quantity, it is expected that a portion of the seeds in \mathcal{S} are relevant. Also, since seeds with similar features and within the same region tend to be similarly relevant, it is unnecessary to select all relevant ones. Given that, we infer that the drawbacks of using fast but imprecise sampling strategies are overcome by oversampling. Thus, the objective is to guarantee the presence of the relevant seeds while removing the irrelevant ones throughout the iterations to ensure N_f superpixels in the last iteration (see Section 4.3).

A classic sampling strategy, hereafter named GRID, in reference to the work of Achanta et al. [2012], is to select equally-distanced seeds in a grid-like pattern. However, in most works, the image’s axis are assumed to be equal, leading to an uneven distribution of the seeds. In this work, we reformulate such an approach to consider any image dimensionality. Let $\varrho_* \in]0, 1[$ be the proportion of a given

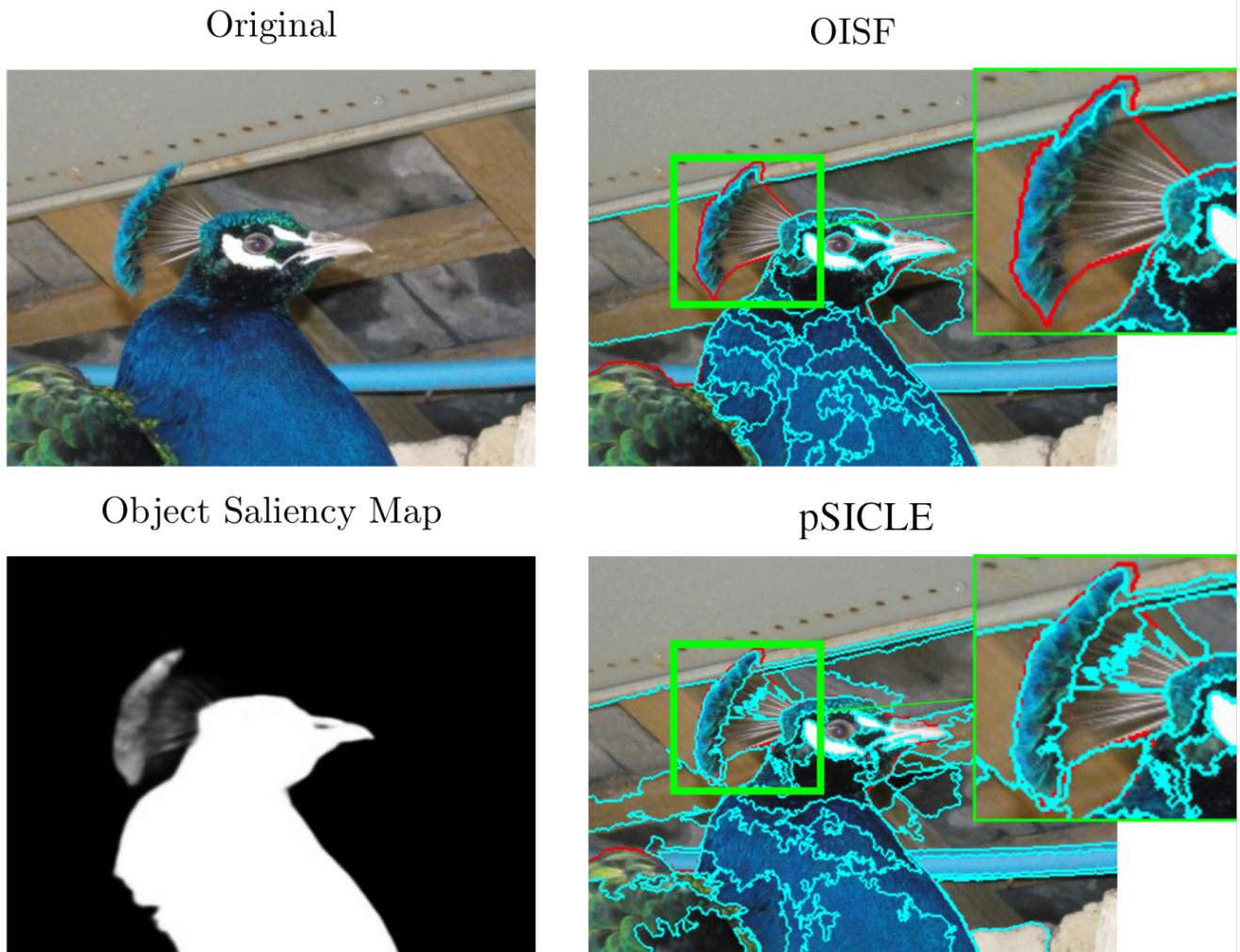


Figure 4. Comparison between pSICLE and OISF (Belém *et al.* [2019a, 2020a]) regarding the dependability of the object saliency map quality. It was sampled a number of $N_0 = 1000$ random seeds for a desired number $N_f = 50$ superpixels. For the IFT, the f_{\max} connectivity function and the w_2 arc-cost function were considered. And for the seed removal step, we considered the V_4 restricted to V_{obj} . The red lines indicate the object boundary, whereas the cyan ones, superpixel borders. The object saliency map was generated from Qin *et al.* [2020].

axis over all image's axis. Then, $N = (\varrho_X \cdot c) \cdot (\varrho_Y \cdot c)$ seeds are sampled considering a constant $c = \sqrt{N_0 / (\varrho_X \cdot \varrho_Y)}$. It is possible to determine the stride between seeds in each axis by dividing its length by $c \cdot \varrho_*$. As one may notice, $N \approx N_0$. Unlike most methods, in this work, we do not shift seeds to the lowest gradient position (in the neighborhood), given their questionable importance in oversampling.

However, one may question the benefits of GRID in oversampling. That is, as N_0 increases, the stride between seeds decreases sharply (*i.e.*, they tend to be closer), and, as a consequence, dispersing seeds equidistantly may not lead to improvements in delineation. Also, such a strategy can become challenging if the user provides a non-rectangular mask representing a region of interest to be segmented: it is necessary to ensure the equidistance rule and best approximate N to the initial number of seeds N_0 desired by the user. Conversely, we argue in favor of random seed oversampling (RND), given its simplicity in implementation, especially for the latter case. Furthermore, backed up by the premise of seed relevance redundancy, one may expect that such an approach can generate seed sets with accurate object delineation results.

4.2 Superpixel Generation

In pSICLE framework, superpixels are generated by the seed-restricted version of the IFT (Falcão *et al.* [2004]) framework. Recently, an object-based path-cost function, formulated by Belém *et al.* [2019a], controls the superpixel morphology with respect to F or O . However, aside from requiring parameter optimization, we claim no need to use either such function or any prior object information in delineation. Clearly, the ground truth (*i.e.*, the ideal saliency map) is defined by the borders in F during annotation by the user, and considering such a map can only reinforce the preexistent borders. Also, given it is yet unfeasible to generate the ideal map, subjecting delineation to a piece of information produced by inference can deteriorate segmentation, especially by reinforcing the borders derived from incorrect estimations. Figure 4 shows the comparison between pSICLE and OISF, which uses such object-based path-cost function. Therefore, in this work, we consider the f_{\max} function due to its reportedly accurate performance in several works (*e.g.*, Belém *et al.* [2020b, 2021]; Bragantini *et al.* [2018]).

Bragantini *et al.* [2018] propose a novel arc-cost function whose segmentation results surpassed many state-of-the-art methods in terms of interactive segmentation. The *Dynamic IFT* (DYN) estimates the arc-costs with respect to the tree’s mean features when it reaches the respective vertex to be conquered (*i.e.*, while it is *growing*). More formally, \mathbf{w}_1 can be defined as $\mathbf{w}_1(x, y) = \|\bar{\mathbf{F}}(\mathbf{T}(s)) - \mathbf{F}(y)\|_2$ in which $s = \mathbf{R}(x)$.

However, \mathbf{w}_1 is highly unstable. As one may see, the tree’s mean feature vector may differ for different tie-breaking policies in the IFT, which can impact the tree’s composition and characteristics and produce, thus, distinct delineations. Conversely, we argue in favor of considering the seed’s features over its tree’s mean vector. First, they promote stability since their features are immutable throughout the execution of the IFT. Furthermore, given that superpixels, by definition, minimize dissimilarity, one may argue that the tree’s features may resemble its seed’s (see Figure 6). Therefore, both arc-cost functions may present similar delineations, especially for regions with high color dissimilarity (*e.g.*, object borders). In this work, we evaluate the use of a root-based function $\mathbf{w}_2(x, y) = \|\mathbf{F}(\mathbf{R}(x)) - \mathbf{F}(y)\|_2$, hereafter named ROOT.

4.3 Seed Removal Criterion

As presented in Section 4.1, \mathcal{S} is produced through oversampling, and it is necessary to remove $N_0 - N_f$ irrelevant seeds for generating N_f superpixels in the last IFT execution. Since determining whether a seed is relevant or not based on its particular features, in pSICLE, we take advantage of the information generated from the previous IFT execution. At each iteration, we let the seeds within \mathcal{S} manifest their possible relevance (through competition) during the IFT and include their superpixel features in their relevance computation.

One may note that the relevance of a seed is relatively defined (*e.g.*, proximity to the object). Thus, since the ideal final set of seeds is not known beforehand, we determine the relevance of a seed by comparing it with its pairs. Then, “perturb” the most relevant by removing a portion of the least relevant, and test their relevance in the subsequent IFT execution. In this approach, the most relevant seeds are consistently labeled as such throughout the iterations.

At the first iteration, it is expected that, aside from numerous irrelevant seeds, many regions contain several equally relevant seeds. By ordering the seeds by their relevance, the portion of the least ones should present instances of the previous cases and, thus, removing a large quantity should impact the object minimally. Throughout the iterations, the number of seeds decreases, and the least relevant ones are more defined by relation (*i.e.*, less relevant than its pairs) than by estimation (*i.e.*, almost zero relevance). Moreover, fewer seeds are directly associated with reducing the seed competition, leading to more volatile superpixels and major delineation errors (especially “leakings”). In such an environment, removing smaller quantities should prevent superpixel growth instability and minimize errors. Therefore, we chose an exponential function $\mathbf{M}(i) \in \mathbb{N}^*$ for computing the number of relevant seeds to be maintained for

the subsequent iteration $i + 1 \in \mathbb{N}^*$.

The major drawback of this strategy is not bounding the number of iterations for reaching the desired number N_f of superpixels, possibly compromising its efficiency. One may argue in favor of imposing a strict number of Ω iterations for segmentation for any quantity of superpixels desired. However, we argue that the necessary number of iterations depends on N_f . For instance, for high competition environments, intra-superpixel dissimilarity is significantly minimized and, thus, applying more iterations for subtle variations in such configuration should minimally impact the object delineation. Conversely, as N_f decreases, removing fewer portions of seeds (*i.e.*, more iterations) assists in stabilizing the superpixel growth. Therefore, we limited the number of iterations of pSICLE to be at most Ω by setting $\mathbf{M}(i) = \max\{(N_0)^{1-\omega^i}, N_f\}$ in which $\omega = 1/(\Omega - 1)$. For obtaining DISF and ODISF (Belém *et al.* [2020b, 2021]) segmentations, one may set $\omega = \log_{N_0} e$, since:

$$\begin{aligned} \mathbf{M}(i) &= \max\{(N_0)^{1-\omega^i}, N_f\} \\ &= \max\{(N_0)(N_0)^{-i \log_{N_0}(e)}, N_f\} \\ &= \max\{(N_0)(e)^{-i}, N_f\} \end{aligned}$$

where e is the Euler number.

As mentioned, we exploit the optimum-path forest generated from the previous iteration for computing the relevance of each seed $s \in \mathcal{S}$. That is, its relevance $\mathbf{V}_*(s) \in \mathbb{R}_+$ is estimated based on the characteristics of its superpixel $\mathbf{T}(s)$. Given that, in IFT (Falcão *et al.* [2004]), a seed’s capability of conquering similar vertices is reflected in its superpixel’s size, we propose a size-based criterion $\mathbf{V}_1(s) = |\mathbf{T}(s)|/|\mathcal{V}|$ for computing the relevance of s .

However, since the background superpixels are often larger than the object’s, \mathbf{V}_1 may undesirably favor the former. On the other hand, the typical homogeneity of the background could be differed by the heterogeneity of the object through superpixel contrast. For that, we first describe the color dissimilarity $\mathbf{G}_{\mathbf{F}}(s, t) \in \mathbb{R}_+$ between the mean features of two trees by $\|\bar{\mathbf{F}}(\mathbf{T}(s)) - \bar{\mathbf{F}}(\mathbf{T}(t))\|_2$, given $s, t \in \mathcal{S}$. Then, we define the *tree’s neighbors* (or adjacents) by $\mathbf{A}(s) = \{t : \exists \langle x, y \rangle \in \mathcal{A}\}$ for $x \in \mathbf{T}(s), y \in \mathbf{T}(t)$, and $s \neq t$. Finally, we can formulate two novel color-based criteria: $\mathbf{V}_2(s) = \min_{t \in \mathbf{A}(s)} \{\mathbf{G}_{\mathbf{F}}(s, t)\}$ and $\mathbf{V}_3(s) = \max_{t \in \mathbf{A}(s)} \{\mathbf{G}_{\mathbf{F}}(s, t)\}$; which aims for the minimum and maximum contrast, respectively, amongst the seed’s neighbors.

Similarly to \mathbf{V}_1 , considering only \mathbf{V}_2 or \mathbf{V}_3 may not be sufficient for selecting relevant seeds. High contrast regions can indicate noises or a well-defined object border. Conversely, low-contrast ones are often located within objects or nearby poorly defined borders. Thus, given that both size and contrast are important for describing the relevance of the seed, we propose two new criteria by combining the size-based and contrast-based estimations. The $\mathbf{V}_4(s) = \mathbf{V}_1(s) \cdot \mathbf{V}_2(s)$ favors large superpixels and prioritizes the minimum contrast amongst the superpixel neighbors, while $\mathbf{V}_5(s) = \mathbf{V}_1(s) \cdot \mathbf{V}_3(s)$ also favors large superpixels, but aims for the maximum contrast.

Still, the relevance of a seed s is strongly tied to which object is desired, making it irrelevant when it does not

delineate the expected one. That is, one may disregard the relevance estimation if one knows that such seed (and its superpixel) does not influence the object delineation (e.g., far from the object of interest). If an object is desired, then more superpixels should be within or near it in order to promote its accurate delineation. In such case, the probable object location can assist in removing seeds that are not within (or nearby) the object of interest. First, analogously to $\mathbf{G}_F(s, t)$ and considering the trees' mean saliency, we define $\mathbf{G}_O(s, t) = \|\overline{\mathbf{O}}(\mathbf{T}(s)) - \overline{\mathbf{O}}(\mathbf{T}(t))\|_2$, given $s, t \in \mathcal{S}$. Then, we propose an object-based criterion $\mathbf{V}_{\text{obj}}(s) = \mathbf{V}_*(s) \cdot \max\{\overline{\mathbf{O}}(\mathbf{T}(s)), \max_{t \in \mathbf{A}(s)} \{\mathbf{G}_O(s, t)\}\}$. By setting $\mathbf{O}(p) = 1 \forall p \in \mathcal{V}$ whenever prior object information is absent, one may see that $\mathbf{V}_{\text{obj}}(s) = \mathbf{V}_*(s)$.

4.4 Algorithm

Since our proposal is a superpixel generation framework, several variants can be straightforwardly built by changing the seed sampling strategy, the path-cost function, the arc-cost function, the seed decreasing factor, the seed removal criterion, or all of them simultaneously. Given that our algorithm (see Algorithm 1) is bounded by the IFT, which is $O(|\mathcal{V}| \log |\mathcal{V}|)$, all pSICLE variants share the same $O(|\mathcal{V}| \log |\mathcal{V}|)$ complexity. As the last step, one may obtain a label $\mathbf{L}(p) \in \{1, \dots, N_f\}$ from \mathbf{R} for each vertex $p \in \mathcal{V}$ by copying the label of its respective seed (i.e., $\mathbf{L}(p) = \mathbf{L}(\mathbf{R}(p))$).

Algorithm 1 pSICLE superpixel framework

Input: Image graph $G = \langle \mathcal{V}, \mathcal{A} \rangle$, object saliency map \mathbf{O} , initial seed quantity N_0 , N_f superpixels, \mathbf{f}_* path-cost function, \mathbf{w}_* arc-cost function, \mathbf{V}_{obj} object-based relevance criterion, ω decrease factor

Output: Label map \mathbf{L}

```

1:  $\mathcal{S} \leftarrow \text{SeedSampling}(N_0)$ 
2:  $i \leftarrow 0$ 
3: while  $|\mathcal{S}| < N_f$  do
4:    $(\mathbf{C}, \mathbf{P}, \mathbf{R}) = \text{IFT}(G, \mathcal{S}, \mathbf{f}_*, \mathbf{w}_*)$ 
5:    $m = \max\{(N_0)^{1-\omega_i}, N_f\}$ 
6:    $\mathcal{S}' \leftarrow \text{IrrelevantSeeds}(G, \mathcal{S}, \mathbf{R}, m, \mathbf{O}, \mathbf{V}_{\text{obj}})$ 
7:    $\mathcal{S} \leftarrow \mathcal{S} \setminus \mathcal{S}'$ 
8:    $i \leftarrow i + 1$ 
9: end while
10:  $\mathbf{L} \leftarrow \text{RootLabelling}(G, \mathcal{S}, \mathbf{R})$ 

```

5 Experimental Results

In this section, we detail the experimental framework for evaluating our proposal and discuss the obtained results. In Section 5.1, the experimental setup is presented, and, in Section 5.2, we perform an ablation study over all pSICLE' steps. Finally, we compare our proposal against state-of-the-art methods both quantitatively and qualitatively (i.e., Sections 5.3 and 5.4).

5.1 Experimental Setup

In order to evaluate the performance of each method for different domains, we selected four datasets whose object of interest is clearly defined (i.e., not ambiguous). The *Extended Complex Saliency Scene Dataset* (ECSSD), created by Shi et al. [2015], is a well-known salient object detection dataset composed of 1000 natural images with distinct and complex objects. Another natural image dataset, *Insects*, was created by Mansilla and Miranda [2016] and has 130 images of insects whose thin legs impose a significant challenge in delineation. To evaluate objects with smooth borders, we opted for two different medical image datasets. The *Liver* dataset, seen in the work of Vargas-Muñoz et al. [2019], contains 40 CT slices of the human liver, whose grayscale borders are difficult to define. Finally, the *Parasites* dataset, used in the work of Belém et al. [2019a], has 72 colored images of helminth eggs in which an impurity may be attached to them. We randomly selected 70% of the data for testing and 30% for training. For generating the object saliency maps, we opted for the work of Qin et al. [2020], which is an effective deep learning approach named U²-Net, suitable for small datasets like Liver and Parasites. We trained such estimator using the training set and considering its default parameters.

We selected several state-of-the-art superpixel methods based on their effectiveness and efficiency: (i) SLIC, by Achanta et al. [2012]¹; (ii) LSC, by Li and Chen [2015]²; (iii) SH, by Wei et al. [2018]³; (iv) ERS, by Liu et al. [2011]⁴; (v) OISF-OSMOX, by Belém et al. [2020a]; and (vi) SEAL-ERS, by Tu et al. [2018]⁵. For all methods, the recommended parameter configuration was set. Our code is publicly available⁶. Given that it is burdensome to evaluate all possible pSICLE configurations, we elected the following one as starting point for our analysis based on previous works. We sampled using GRID with $N_0 = 8000$ seeds and considered \mathbf{w}_1 arc-cost function as seen in Bragantini et al. [2018]; Belém et al. [2020b], and used the \mathbf{V}_{obj} , considering \mathbf{V}_4 relevance criterion, as presented in the work of Belém et al. [2021]. Finally, similar to most seed-based methods, like those from Achanta et al. [2012]; Li and Chen [2015]; Vargas-Muñoz et al. [2019], we set the maximum number of iterations $\Omega = 10$.

Due to the high correlation between several superpixel segmentation metrics, as noticed by Stutz et al. [2018], we chose the two most popular ones. The *Boundary Recall* (BR), as proposed in the work of Stutz et al. [2018], calculates the percentage of the object boundaries correctly overlapped by a superpixel border given a certain tolerance radius. Neubert and Protzel [2012] proposed the *Under-Segmentation Error* (UE) that measures the error resultant from superpixels overlapping multiple ground-truths. We aim for higher values of BR and lower values of UE. For all experiments, we set $N_f \in [25, 750]$.

¹<https://www.epfl.ch/labs/ivrl/research/slic-superpixels/>

²<https://jschenth.wweebly.com/projects.html>

³<https://github.com/semiquark1/boruvka-superpixel>

⁴<https://github.com/mingyuliutw/EntropyRateSuperpixel>

⁵<https://github.com/wctu/SEAL>

⁶<https://github.com/LIDS-UNICAMP/SICLE>

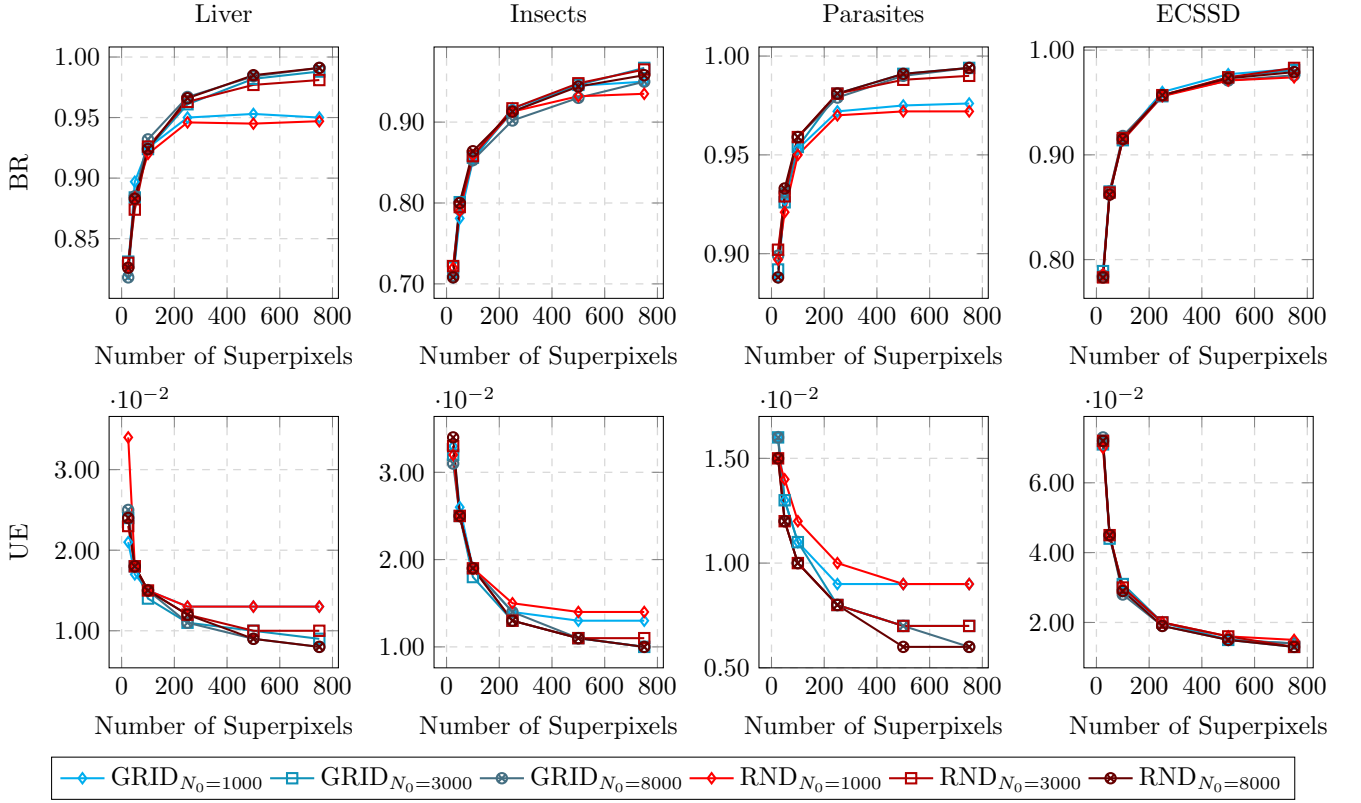


Figure 5. Results of pSICLE variants with varying oversampling strategy (GRID and RND) and $N_0 \in \{8000, 3000, 1000\}$. We have considered the \mathbf{f}_{\max} with \mathbf{w}_1 for superpixel estimation, \mathbf{V}_{obj} with \mathbf{V}_4 criterion, and using the saliency estimated from Qin *et al.* [2020].

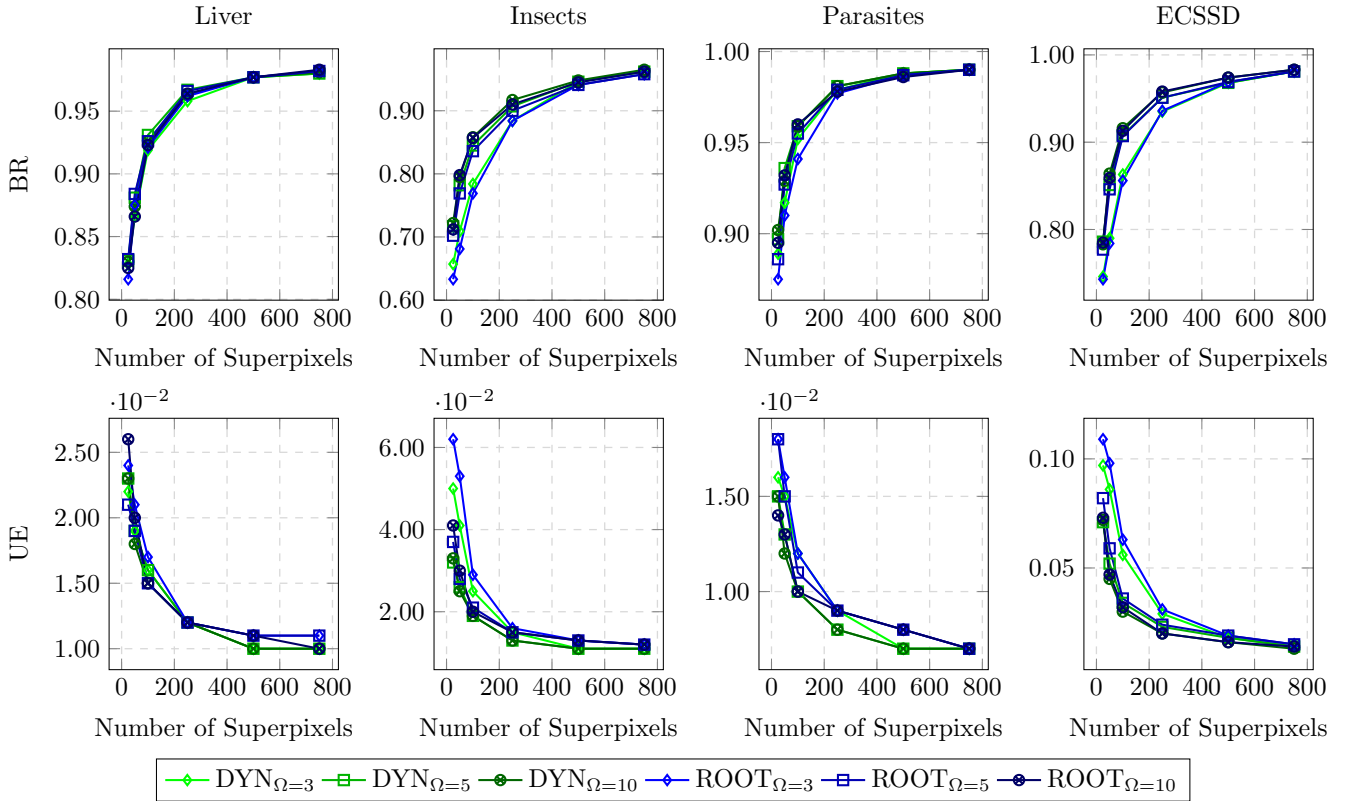


Figure 6. Results of pSICLE variants with varying $\mathbf{w}_* \in \{\mathbf{w}_1, \mathbf{w}_2\}$ function and $\Omega \in \{3, 5, 10\}$. We have considered $N_0 = 3000$ using the RND strategy, the \mathbf{f}_{\max} for superpixel estimation, \mathbf{V}_{obj} with \mathbf{V}_4 criterion, and using the saliency estimated from Qin *et al.* [2020].

5.2 Ablation Study

We analyze the impacts on the oversampling task by altering the strategy (*i.e.*, GRID, or RND) and N_0 , as Figure 5 shows.

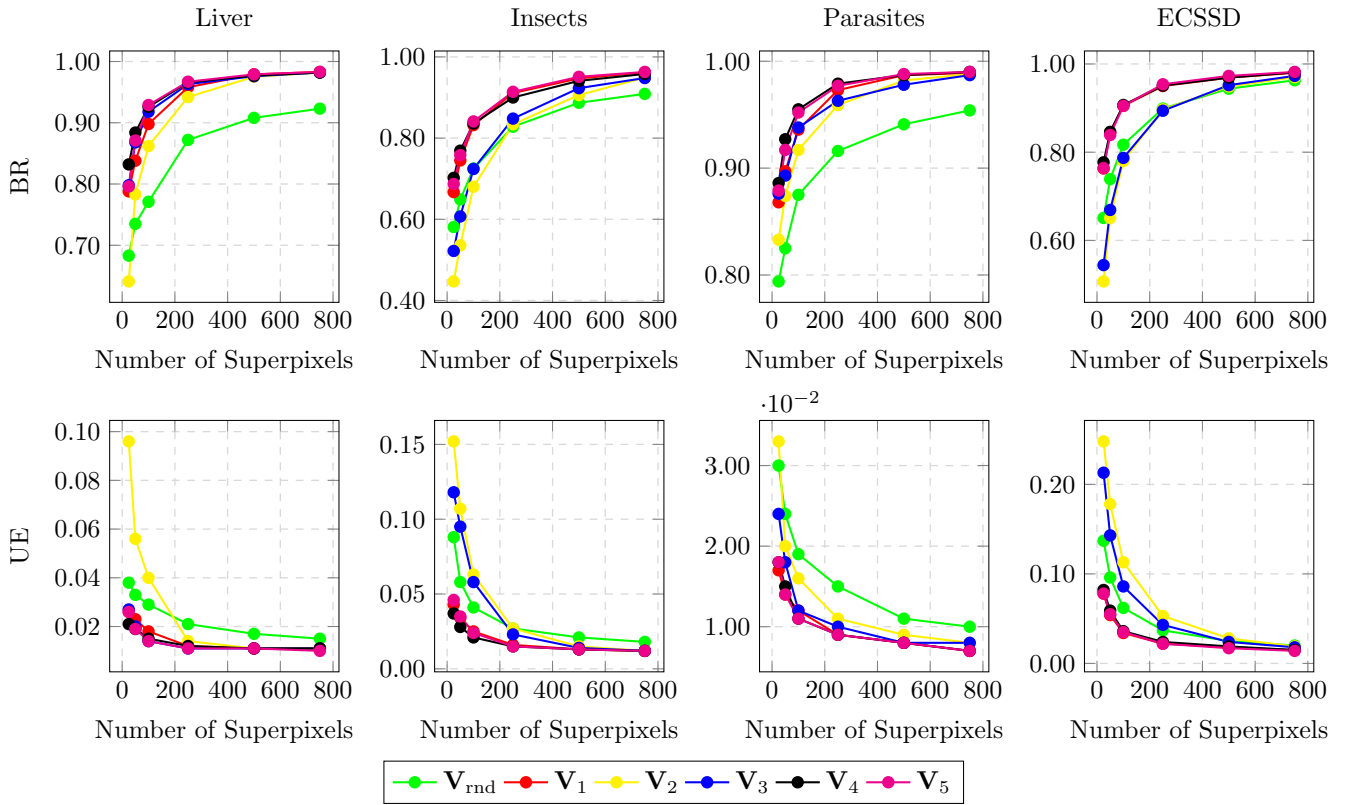


Figure 7. Results of pSICLE variants with varying $V_* \in \{V_{\text{rnd}}, V_1, V_2, V_3, V_4, V_5\}$ function, all subjected to V_{obj} . We have considered $N_0 = 3000$ using the RND strategy, the f_{max} and the w_2 for superpixel estimation, and using the saliency estimated from Qin *et al.* [2020].

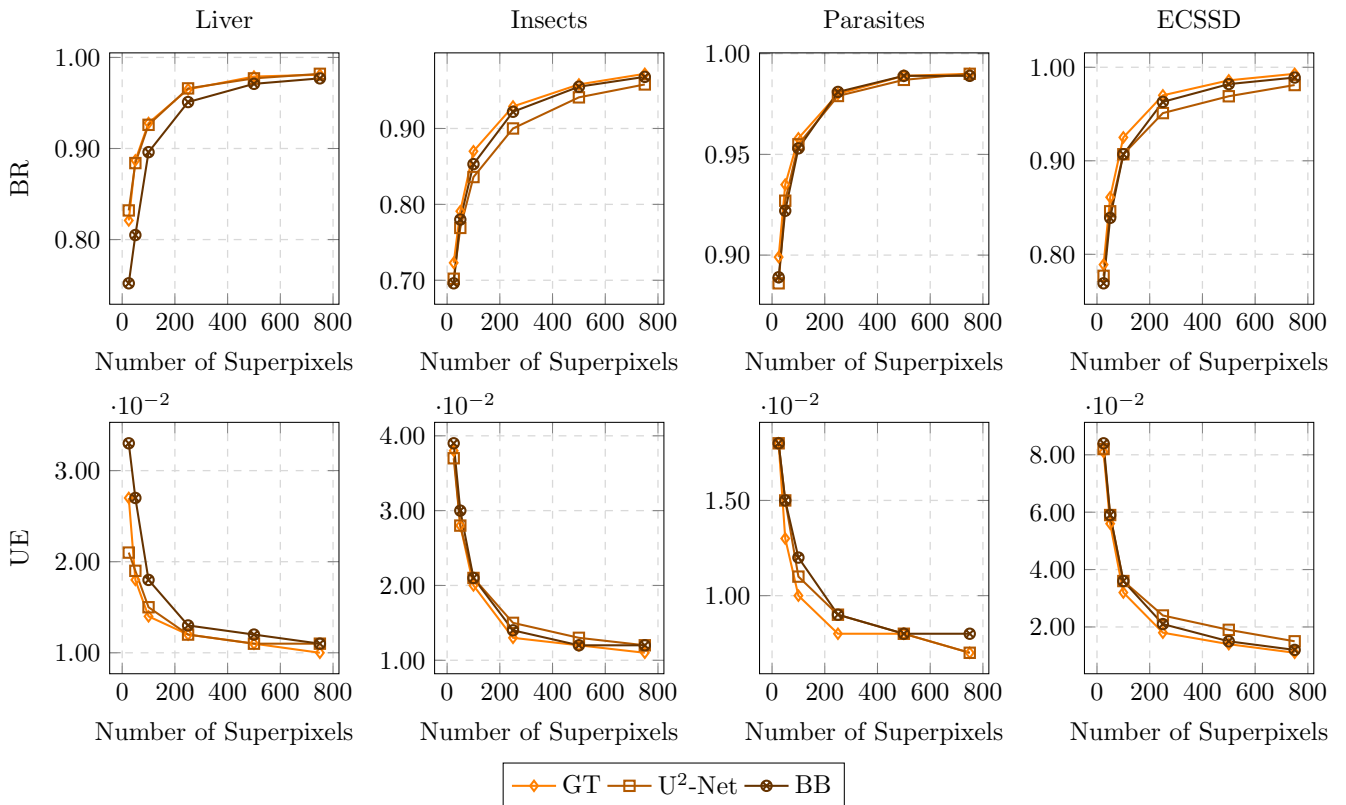


Figure 8. Results of pSICLE variants with varying saliency maps: Ground truth (GT); U^2 -Net from Qin *et al.* [2020]; and Minimum Bounding Box (BB). We have considered $N_0 = 3000$ using the RND strategy, the f_{max} and the w_2 for superpixel estimation, and V_{obj} with V_4 criterion.

It is possible to see that both strategies achieved a similar performance, irrespective of N_0 , leading to the conclusion

that pattern has no substantial influence in pSICLE. On the other hand, we may also affirm that there are multiple ways

to obtain a relevant seed set, given that different strategies achieved equivalent performance. That is, such results support the idea of relevance redundancy amongst seeds.

In terms of N_0 , it is possible to see that 1000 seeds are sufficient for accurate delineation in the ECSSD dataset but insufficient for the remaining ones. We argue that this event is influenced by the images' size: the bigger the images, the more seeds are required in oversampling. However, seeing the overlap between 8000 seeds and 3000 seeds, one can infer that the mere increase of N_0 is insufficient for improving the delineation performance. As we argue, the precise number of seeds for maximizing delineation depends on the image's dimensionality and the objects' characteristics. For instance, a big but homogeneous object might require few seeds for accurate delineation. On the other hand, a medium-sized object with significant feature variation (e.g., complex textures, different colors) may demand more seeds for its proper delineation. Since the object of interest is subjectively determined by the user, establishing such an optimal quantity is a challenging problem. Finally, as one may note, independently of N_0 , the segmentation result is more influenced by the final number of superpixels N_f . Therefore, based on our results and for lesser memory consumption, we opt for randomly oversampling 3000 seeds.

Figure 6 shows the results for varying arc-cost functions (i.e., DYN and ROOT) and limiting the maximum number of iterations $\Omega \in \{3, 5, 10\}$. The equal performance between estimations can be explained by the possible similarity between the seed's and tree's features. Interestingly, $\Omega = 3$ is sufficient for homogeneous objects (i.e., liver, and parasite egg) but insufficient for more complex ones. We argue that the number of iterations for seed relevance manifestation is proportional to the object's complexity. However, when comparing $\Omega = 5$ and $\Omega = 10$ in all datasets, there are no significant improvements, indicating a possible upper bound (i.e., performing unnecessary iterations). Therefore, we limited pSICLE to perform, at most, five iterations using the ROOT function, which allows future improvements, as performed by the works of Gonçalves *et al.* [2019]; Falcão and Bergo [2004]; Belém *et al.* [2023].

We have also evaluated our proposed seed removal criteria $\{\mathbf{V}_i\}_{i=1}^5$ subjected to \mathbf{V}_{obj} alongside a random selection criterion \mathbf{V}_{rnd} as baseline (Figure 7). Unsurprisingly, the random criterion is the worst approach for all datasets. However, it is interesting that, while contrast-only criteria perform fairly for datasets where contrast is crucial (i.e., Liver and Parasites), the size-only criterion achieves top performance in all datasets. We argue that \mathbf{V}_{obj} manages to overcome their possible biases by constraining the seed selection near (or within) the desired object. When combining both contrast and size estimations, we can see a slight improvement for Liver and Parasites, and no performance degradation for the remaining ones. Furthermore, if no map is provided, such a combination can overcome their particular biases (cf. Belém *et al.* [2020b]). Thus, we opt for maintaining the \mathbf{V}_4 subjected to \mathbf{V}_{obj} .

Finally, as Figure 8 shows, we assessed pSICLE considering maps of different qualities for analyzing the influence of the map's borders in delineation. Aside from those generated by the U²-Net, used throughout this paper,

we considered the ground truth (GT) and the object's minimum bounding box (BB) for emulating the generation of maps with ideal and rough object borders, respectively. We emphasize that the GT maps are only used in this particular analysis. It is important to note that, differently from other object-based strategies (e.g., Belém *et al.* [2019a, 2020a]), the delineation performance of pSICLE is not impacted by the quality of the map's border. As a consequence, and setting aside that generating ideal maps is yet an unfeasible task, we claim that this property is positive since it disregards the need to improve the object saliency map whenever it is not the problem's main goal.

5.3 Quantitative Analysis

We compared our best pSICLE variant against several state-of-the-art methods, as illustrated in Figure 9. That is, $N_0 = 3000$ seeds sampled by RND, ROOT arc-cost function, $\Omega = 5$ iterations, and \mathbf{V}_{obj} with \mathbf{V}_4 criterion. It is possible to see that pSICLE is consistently accurate for both metrics in all datasets. When considering different objects with distinct features, our approach's strategy of oversampling and object-based seed removal is proven more effective than the typical approach of seed recomputation seen in SLIC, LSC and OISF (Achanta *et al.* [2012]; Li and Chen [2015]; Belém *et al.* [2020a], respectively). When analyzing the UE, the results support the findings of Belém *et al.* [2021, 2019b], claiming that object-based strategies offer better superpixel leaking prevention.

For comparison, table 2 shows the computational time required for each of the fastest methods for the ECSSD (Shi *et al.* [2015]) dataset on a 64-bit Intel (R) Core(TM) i5-5200U PC with CPU speed of 2.20Ghz. Given that both pSICLE and OISF do not compute saliency but only use it when provided, the computational time for generating saliency maps was not taken into account. In terms of time complexity, pSICLE is $O(|\mathcal{V}| \log |\mathcal{V}|)$ bounded by the IFT (Falcão *et al.* [2004]) executions. Although SLIC is $O(|\mathcal{V}|)$, we can see that, in practice, pSICLE tends to be faster since it is limited by, at most, 5 iterations. One could argue in favor of SH (Wei *et al.* [2018]), which is $O(|\mathcal{V}|)$, given its property of computing a multiscale segmentation in a single execution. However, pSICLE has such property if one sets the desired quantity of superpixels (i.e., seeds) at each iteration. Lastly, pSICLE is a more efficient object-based solution than OISF.

5.4 Qualitative Analysis

We compare some segmentation results between pSICLE and other state-of-the-art methods in Figure 10. The ERS (Liu *et al.* [2011]) method achieves effective object delineation only for the insect image, whereas it performs fairly for the other images. Conversely, SH presents better and, more importantly, more consistent results for all images. However, the lack of object information compromises the object delineation of the desired object in several parts. For example, both object-based methods increased the superpixel resolution within (or near) the object and improved its delineation. Still, for OISF, the incorrect

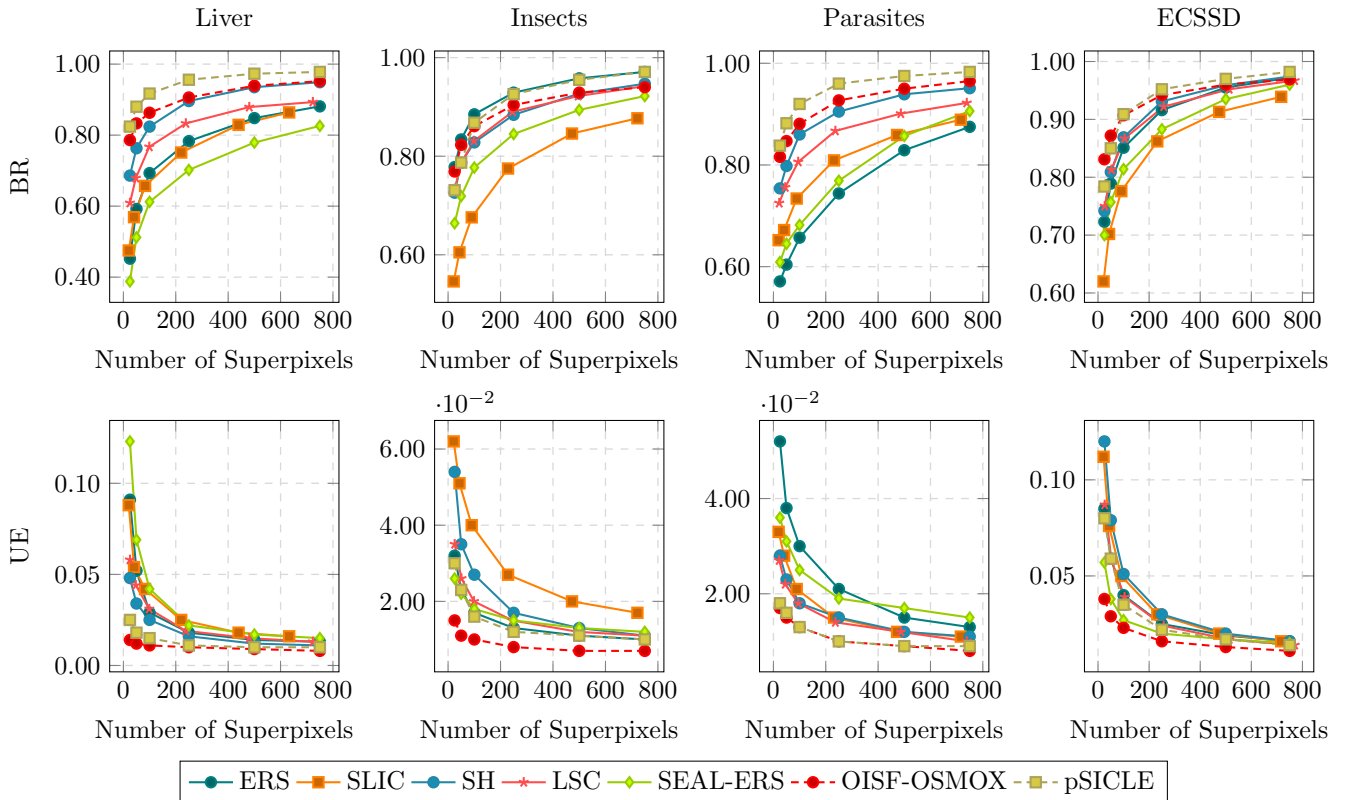


Figure 9. Results of pSICLE against several state-of-the-art methods. We have considered $N_0 = 3000$ using the RND strategy, the f_{\max} and the w_2 for superpixel estimation, V_{obj} with V_4 criterion, and using the saliency estimated from Qin *et al.* [2020].

N_f	SLIC	SH	OISF-OSMOX	pSICLE
25	0.472±0.026	0.916±0.035	3.091±0.628	0.541±0.067
100	0.473±0.025	0.915±0.032	1.757±0.343	0.445±0.060
750	0.477±0.025	0.916±0.032	1.182±0.229	0.361±0.053

Table 2. Average execution time for each of the fastest methods in the ECSSD dataset. The best performance for each quantity is depicted in bold. We have considered $N_0 = 3000$ using the RND strategy, the f_{\max} and the w_2 for superpixel estimation, V_{obj} with V_4 criterion, and using the saliency estimated from Qin *et al.* [2020]. The time for estimating the saliency was not computed for this table.

estimations in the map contributed to the deterioration of the object delineation in several parts. Finally, pSICLE achieves effective delineation for all objects irrespective of saliency errors.

6 Conclusion and Future Work

In this work, we propose a novel object-based superpixel segmentation framework named *Superpixels through Iterative CLEARcutting* (pSICLE), which is a generalization of two state-of-the-art methods (*i.e.*, Belém *et al.* [2020b, 2021]). It starts off from oversampling and, through several iterations, generates superpixels from the seed set and removes a portion of irrelevant seeds to preserve the accurate object delineation from the previous iteration. We show that pSICLE is consistently effective by evaluating its performance against several state-of-the-art methods in different datasets. Moreover, results show that, in most practical cases, pSICLE can be faster than SLIC, proposed by Achanta *et al.* [2012]. For future endeavors, we intend to extend pSICLE for 3D and video segmentation while exploring its capability of being an interactive segmentation

tool.

Declarations

Funding

This work was financially supported by the Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq – (Universal 407242/2021-0, International 442950/2023-3, PQ 306573/2022-9, PQ 304711/2023-3, 310075/2019-0), the Fundação de Amparo a Pesquisa do Estado de Minas Gerais – FAPEMIG – (PPM-00006-18, APQ-01079-23 and APQ-05058-23), the Fundação de Amparo a Pesquisa do Estado de São Paulo – FAPESP – (CPA-FAPESP 2023/14427-8) and the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES – Finance code 001 (COFECUB 88887.191730/2018-00 and STIC-AMSUD 88887.878869/2023-00).

Authors’ Contributions

All authors have equally contributed in this work.

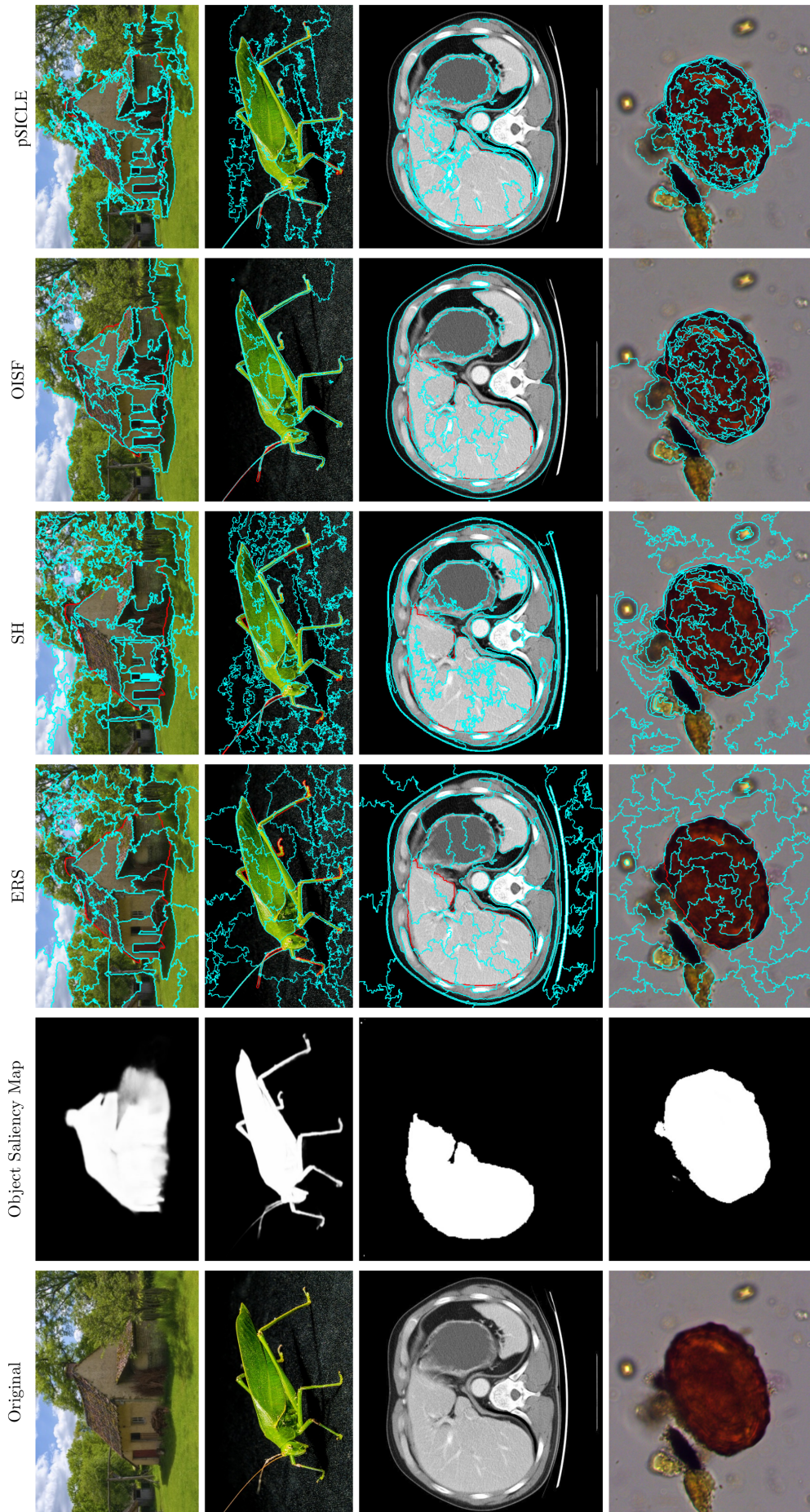


Figure 10. Qualitative comparison between pSICLE and several state-of-the-art methods. The desired number of superpixels was $N_f = 50$. The red lines indicate the object boundary, whereas the cyan ones, superpixel borders. We have considered $N_0 = 3000$ using the RND strategy, the f_{max} and the w_2 for superpixel estimation, V_{obj} with V_4 criterion, and using the saliency estimated from Qin et al. [2020].

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials

Apart from Liver and Parasites, which are private datasets, Insects and ECSSD can be found online, alongside the baselines used in this work. Our method pSICLE can be found online at <https://github.com/LIDS-UNICAMP/SICLE>.

References

- Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Süsstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2282. DOI: 10.1109/TPAMI.2012.120.
- Achanta, R. and Süsstrunk, S. (2017). Superpixels and polygons using simple non-iterative clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4895–4904. DOI: 10.1109/CVPR.2017.520.
- Awaisu, M., Li, L., Peng, J., and Zhang, J. (2019). Fast superpixel segmentation with deep features. In *Advances in Computer Graphics*, volume 11542, pages 410–416. DOI: 10.1007/978-3-030-22514-8_38.
- Ban, Z., Liu, J., and Cao, L. (2018). Superpixel segmentation using gaussian mixture model. *IEEE Transactions on Image Processing*, 27(8):4105–4117. DOI: 10.1109/TIP.2018.2836306.
- Belém, F., Borlido, I., João, L., Perret, B., Cousty, J., Guimarães, S., and Falcão, A. (2022). Fast and effective superpixel segmentation using accurate saliency estimation. In *Discrete Geometry and Mathematical Morphology*, pages 261–273. DOI: 10.1007/978-3-031-19897-7_21.
- Belém, F., Borlido, I., João, L., Perret, B., Cousty, J., Guimarães, S., and Falcão, A. (2023). Novel arc-cost functions and seed relevance estimations for compact and accurate superpixels. *Journal of Mathematical Imaging and Vision*, 65(5):770–786. DOI: 10.1007/s10851-023-01156-9.
- Belém, F., Cousty, J., Perret, B., Guimarães, S., and Falcão, A. (2021). Towards a simple and efficient object-based superpixel delineation framework. In *34th Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 346–353. DOI: 10.1109/SIBGRAPI54419.2021.00054.
- Belém, F., Guimarães, S., and Falcão, A. (2019a). Superpixel segmentation by object-based iterative spanning forest. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, volume 11401, pages 334–341. DOI: 10.1007/978-3-030-13469-3_39.
- Belém, F., Guimarães, S., and Falcão, A. (2020a). Superpixel generation by the iterative spanning forest using object information. In *33rd Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 22–28. Workshop of Thesis and Dissertations. DOI: 10.5753/sibgrapi.est.2020.12979.
- Belém, F., Guimarães, S., and Falcão, A. (2020b). Superpixel segmentation using dynamic and iterative spanning forest. *Signal Processing Letters*, 27:1440–1444. DOI: 10.1109/LSP.2020.3015433.
- Belém, F., Melo, L., Guimarães, S., and Falcão, A. (2019b). The importance of object-based seed sampling for superpixel segmentation. In *32nd Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 108–115. DOI: 10.1109/SIBGRAPI.2019.00023.
- Bragantini, J., Martins, S., Castelo-Fernandez, C., and Falcão, A. (2018). Graph-based image segmentation using dynamic trees. In *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*, pages 470–478. DOI: 10.1007/978-3-030-13469-3_55.
- Castelo-Fernandez, C. and Falcão, A. (2019). Learning visual dictionaries from class-specific superpixel segmentation. In *Computer Analysis of Images and Patterns*, pages 171–182. DOI: 10.1007/978-3-030-29888-3_14.
- Dhore, S. and Abin, D. (2021). Chest x-ray segmentation using watershed and super pixel segmentation technique. In *International Conference on Communication Information and Computing Technology (ICCICT)*, pages 1–4. DOI: 10.1109/ICCICT50803.2021.9510078.
- Ding, J., Xue, N., Xia, G., Schiele, B., and Dai, D. (2023). Hgformer: Hierarchical grouping transformer for domain generalized semantic segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15413–15423. DOI: 10.1109/CVPR52729.2023.01479.
- Falcão, A. and Bergo, F. (2004). Interactive volume segmentation with differential image foresting transforms. *IEEE Transactions on Medical Imaging*, 23(9):1100–1108. DOI: 10.1109/TMI.2004.829335.
- Falcão, A., Stolfi, J., and Lotufo, R. (2004). The image foresting transform: Theory, algorithms, and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(1):19–29. DOI: 10.1109/TPAMI.2004.1261076.
- Galvão, F., Falcão, A., and Chowdhury, A. (2018). RISE: recursive iterative spanning forest for superpixel segmentation. In *31st Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 408–415. DOI: 10.1109/SIBGRAPI.2018.00059.
- Gonçalves, H., Vasconcelos, G., Rangel, P., Carvalho, M., Archilha, N., and Spina, T. (2019). cudaift: 180x faster image foresting transform for waterpixel estimation using cuda. In *14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, volume 4, pages 395–404. DOI: 10.5220/0007402703950404.
- Jampani, V., Sun, D., Liu, M., Yang, M., and Kautz, J. (2018). Superpixel sampling networks. In *European Conference on Computer Vision (ECCV)*, volume 11211, pages 363–380. DOI: 10.1007/978-3-030-01234-2_22.
- Li, Z. and Chen, J. (2015). Superpixel segmentation using linear spectral clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1356–1363. DOI: 10.1109/CVPR.2015.7298741.

- Liu, C., Zhao, R., and Pang, M. (2019). A fully automatic segmentation algorithm for ct lung images based on random forest. *Medical Physics*, 47(2):518–529. DOI: 10.1002/mp.13939.
- Liu, M., Tuzel, O., Ramalingam, S., and Chellappa, R. (2011). Entropy rate superpixel segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2097–2104. DOI: 10.1109/CVPR.2011.5995323.
- Liu, Y., Yu, M., Li, B., and He, Y. (2018). Intrinsic manifold SLIC: A simple and efficient method for computing content-sensitive superpixels. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(3):653–666. DOI: 10.1109/TPAMI.2017.2686857.
- Mansilla, L. and Miranda, P. (2016). Oriented image foresting transform segmentation: Connectivity constraints with adjustable width. In *29th Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 289–296. DOI: 10.1109/SIBGRAPI.2016.047.
- Martins, S., Ruppert, G., Reis, F., Yasuda, C., and Falcão, A. (2019). A supervoxel-based approach for unsupervised abnormal asymmetry detection in MR images of the brain. In *IEEE 16th International Symposium on Biomedical Imaging (ISBI)*, pages 882–885. DOI: 10.1109/ISBI.2019.8759166.
- Miranda, P. and Mansilla, L. (2014). Oriented image foresting transform segmentation by seed competition. *IEEE Transactions on Image Processing*, 23(1):389–398. DOI: 10.1109/TIP.2013.2288867.
- Neubert, P. and Protzel, P. (2012). *Superpixel Benchmark and Comparison*, volume 6. KIT Scientific Publishing. Book.
- Peng, H., Aviles-Rivero, A., and Schonlieb, C. (2022). Hers superpixels: Deep affinity learning for hierarchical entropy rate segmentation. pages 72–81. DOI: 10.1109/WACV51458.2022.00015.
- Qin, X., Zhang, Z., Huang, C., Dehghan, M., Zaiane, O., and Jagersand, M. (2020). U2-net: Going deeper with nested u-structure for salient object detection. *Pattern Recognition*, 106:107404. DOI: 10.1016/j.patcog.2020.107404.
- Shen, J., Hao, X., Liang, Z., Liu, Y., Wang, W., and Shao, L. (2016). Real-time superpixel segmentation by DBSCAN clustering algorithm. *IEEE Transactions on Image Processing*, 25(12):5933–5942. DOI: 10.1109/TIP.2016.2616302.
- Shi, J., Yan, Q., Xu, L., and Jia, J. (2015). Hierarchical image saliency detection on extended cssd. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(4):717–729. DOI: 10.1109/TPAMI.2015.2465960.
- Sousa, A., Martins, S., Falcão, A., Reis, F., Bagatin, E., and Irion, K. (2019). Altis: A fast and automatic lung and trachea ct-image segmentation method. *Medical Physics*, 46(11):4970–4982. DOI: 10.1002/mp.13773.
- Stutz, D., Hermans, A., and Leibe, B. (2018). Superpixels: An evaluation of the state-of-the-art. *Computer Vision and Image Understanding*, 166:1–27. DOI: 10.1016/j.cviu.2017.03.007.
- Suzuki, T. (2020). Superpixel segmentation via convolutional neural networks with regularized information maximization. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2573–2577. DOI: 10.1109/ICASSP40776.2020.9054140.
- Tu, W., Liu, M., Jampani, V., Sun, D., Chien, S., Yang, M., and Kautz, J. (2018). Learning superpixels with segmentation-aware affinity loss. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 568–576. DOI: 10.1109/CVPR.2018.00066.
- Vargas-Muñoz, J., Chowdhury, A., Alexandre, E., Galvão, F., Miranda, P., and Falcão, A. (2019). An iterative spanning forest framework for superpixel segmentation. *IEEE Transactions on Image Processing*, 28(7):3477–3489. DOI: 10.1109/TIP.2019.2897941.
- Wei, X., Yang, Q., Gong, Y., Ahuja, N., and Yang, M. (2018). Superpixel hierarchy. *IEEE Transactions on Image Processing*, 27(10):4838–4849. DOI: 10.1109/TIP.2018.2836300.
- Wu, J., Liu, C., and Li, B. (2021). Texture-aware and structure-preserving superpixel segmentation. *Computers and Graphics*, 94:152–163. DOI: 10.1016/j.cag.2020.12.002.
- Xiao, X., Zhou, Y., and Gong, Y. (2018). Content-adaptive superpixel segmentation. *IEEE Transactions on Image Processing*, 27(6):2883–2896. DOI: 10.1109/TIP.2018.2810541.
- Xu, L., Zeng, L., and Wang, Z. (2014). Saliency-based superpixels. *Signal, Image and Video Processing*, 8(1):181–190. DOI: 10.1007/s11760-013-0520-8.
- Yang, F., Sun, Q., Jin, H., and Zhou, Z. (2020). Superpixel segmentation with fully convolutional networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13961–13970. DOI: 10.1109/CVPR42600.2020.01398.
- Yi, S., Ma, H., Wang, X., Hu, T., Li, X., and Wang, Y. (2022). Weakly-supervised semantic segmentation with superpixel guided local and global consistency. *Pattern Recognition*, 124:108504. DOI: 10.1016/j.patcog.2021.108504.
- Yu, Y., Makihara, Y., and Yagi, Y. (2019). Pedestrian segmentation based on a spatio-temporally consistent graph-cut with optimal transport. *IPSJ Transactions on Computer Vision and Applications*, 11:10. DOI: 10.1186/s41074-019-0062-2.
- Yu, Y., Yang, Y., and Liu, K. (2021). Edge-aware superpixel segmentation with unsupervised convolutional neural networks. In *IEEE International Conference on Image Processing (ICIP)*, pages 1504–1508. DOI: 10.1109/ICIP42928.2021.9506289.
- Zhang, S., Wang, H., Huang, W., and You, Z. (2018). Plant diseased leaf segmentation and recognition by fusion of superpixel, K-means and PHOG. *Optik*, 157:866–872. DOI: 10.1016/j.ijleo.2017.11.190.
- Zhou, J., Ruan, J., Wu, C., Ye, G., Zhu, Z., Yue, J., and Zhang, Y. (2019). Superpixel segmentation of breast cancer pathology images based on features extracted from the autoencoder. In *IEEE 11th International Conference on Communication Software and Networks (ICCSN)*,

pages 366–370. DOI: 10.1109/ICCSN.2019.8905358.