Graph-based Supervoxel Computation from Iterative Spanning Forest

Carolina Stephanie Jerônimo de Almeida, Felipe Belèm, Sarah A. Carneiro, Zenilton Patrocínio Jr, Laurent Najman, Alexandre Xavier Falcao, Silvio Jamil Ferzoli Guimarães

Graph-based Supervoxel Computation from Iterative Spanning Forest [⋆]

Carolina Jerônimo[1], Felipe Belém[3][0000−0002−6037−5977], Sarah A. Carneiro[2][0000−0001−7653−8614], Zenilton K. G. Patrocínio Jr[1][0000−0003−0804−1790], Laurent Najman[2][0000−0002−6190−0235], Alexandre Falcão[3], and Silvio Jamil F. Guimarães[1,2][0000−0001−8522−2056]

[1] Laboratory of Image and Multimedia Data Science (ImScience), Pontifical Catholic University of Minas Gerais 31980–110, Brazil,
{zenilton,sjamil}@pucminas.br

[2] LIGM, Univ Gustave Eiffel, CNRS, ESIEE Paris, F-77454 Marne-la-Vallée
{sarah.alcar,laurent.najman}@esiee.fr

[3] Laboratory of Image Data Science (LIDS), University of Campinas, São Paulo 13083–852, Brazil, {afalcao,felipe.belem}@ic.unicamp.br

Abstract. Supervoxel segmentation leads to major improvements in video analysis since it generates simpler but meaningful primitives (*i.e.*, supervoxels). Thanks to the flexibility of the Iterative Spanning Forest (ISF) framework and recent strategies introduced by the Dynamic Iterative Spanning Forest (DISF) for superpixel computation, we propose a new graph-based method for supervoxel generation by using iterative spanning forest framework, so-called ISF2SVX, based on a pipeline composed by four stages: (a) graph creation; (b) seed oversampling; (c) IFT-based superpixel delineation; and (d) seed set reduction. Moreover, experimental results show that ISF2SVX is capable of effectively describing the video's color variation through its supervoxels, while being competitive for the remaining metrics considered.

Keywords: Graph-based method · Supervoxel computation · Iterative Spanning Forest.

1 Introduction

In image and video applications, it is often necessary to separate the objects from its background for subsequent analysis. One approach generates groups of connected elements (*i.e.*, superpixels or supervoxels) which shares a common property (*e.g.*, color and texture). By generating numerous groups, the object can be effectively defined by its comprising regions, being the major premise of

[⋆] The authors thank Conselho Nacional de Desenvolvimento Científico e Tecnológico – CNPq – (PQ 310075/2019-0), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES – (Grant COFECUB 88887.191730/2018-00) and Fundação de Amparo à Pesquisa do Estado de Minas Gerais – FAPEMIG – (Grants PPM-00006-18).

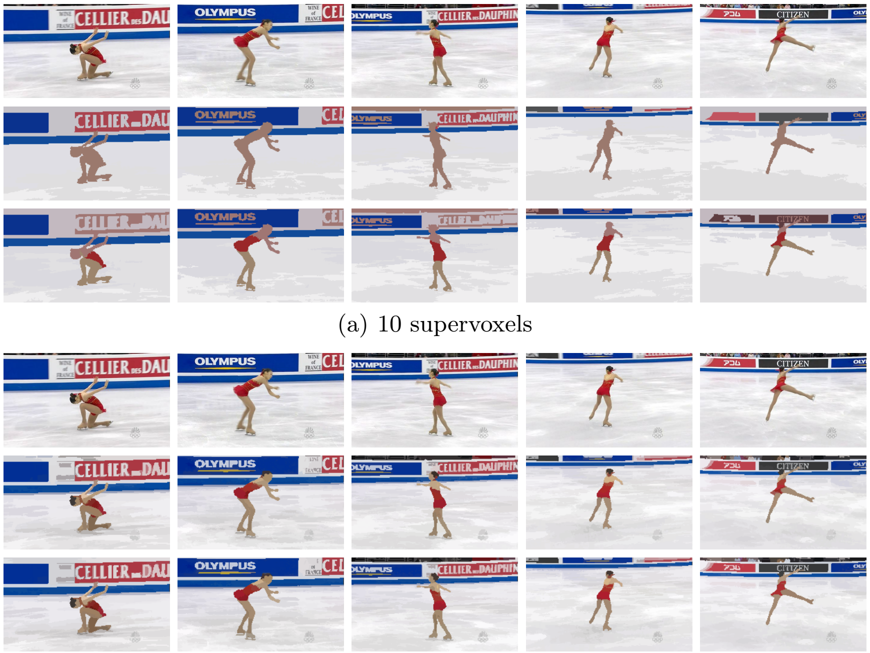
superpixel and supervoxel segmentation algorithms. Such methods are applied in many contexts such as: (i) object detection [13, 11]; (ii) cloud connectivity [20, 14, 22]; and (iii) long-range tracking [16].

For early video processing, one can interpret it as a three-dimensional spatiotemporal volume and segment its objects. The *graph-based supervoxel* (GB) [9] is a image segmentation method based on graphs, presenting good boundary adherence but it is so computationally expensive. The *hierarchical GB* (GBH) [12] considers the GB strategy for computing a hierarchical iterative method; while the *stream GBH* (sGBH) [24] extends the latter for online video segmentation. The authors in [17] proposed the method *cp-HOScale* that improves GBH by computing the whole hierarchy without increasing the computational cost. Although GBH overcomes GB speed performance drawback, it does not guarantee the generation of the desired number of supervoxels. Analogous for GB and GBH, MeanShift [15] and *Segmentation by Weighted Aggregation* (SWA) [7] optimize the normalized cuts criterion, in which SWA performs it hierarchically. However, while MeanShift presents a fair delineation performance, SWA does not guarantee to produce the exact number of supervoxels. Three properties are desirable in video supervoxel segmentation: (i) spatiotemporal boundary adherence; (ii) computational efficiency; and (iii) ability to control the number of supervoxels generated. However, no supervoxel segmentation algorithm has all these characteristics [21].

Recent advances in superpixel segmentation (*e.g.*, deep learning strategies) are strongly related to the image dimensionality and, thus, an extension for video might not guarantee the same performance as the one reported. Thus, the improvements in both categories are often self-contained, significantly limiting their possible improvements. As an example, while GB [9] and GBH [12] equivalents are considered state-of-the-art methods in video segmentation, in superpixel segmentation, they were surpassed by a large set of newer and more effective approaches [19, 2]. Finally, although the authors in [2] discuss how hierarchical superpixel methods might propagate errors to coarser levels, one can see that such a statement holds for hierarchical supervoxel segmentation, which is often considered to be a desirable property [21].

Inspired by the *Iterative Spanning Forest* (ISF) [19], a recent superpixel segmentation framework, in this work, we propose a supervoxel segmentation framework for video segmentation, named *ISF for Supervoxels* (ISF2SVX). Similar to ISF, our approach is composed of independent steps: (i) graph construction; (ii) seed sampling; (iii) supervoxel generation; and (iv) seed recomputation. In step (i), the video volume is converted to a directed graph representation which will be used as input for determining the seeds in step (ii). Then, for several iterations, ISF2SVX generates supervoxels through the *Image Foresting Transform* (IFT) [8] using improved seed sets — in steps (iii) and (iv), respectively. Figure 1 illustrates examples of results obtained by ISF2SVX by changing the strategy for seed sampling (grid and random) for 10 and 500 supervoxels.

This paper is organized as follows. In Section 2, important concepts used in this work such as graphs and IFT are clarified. In Section 3, the methodology



(a) 10 supervoxels

(b) 500 supervoxels

Fig. 1. Examples of video segmentations for a video extracted of the GATech. The original frames are illustrated in the first row. We illustrate examples of the proposed method changing the seed sampling. We illustrate results for (a) 10 and (b) 500 supervoxels, considering grid and random seed sampling (second and third rows). Each resulting region is colored by its mean color.

for the proposed segmentation approach is explained. In Section 4, we describe the experiments performed and compare the achieved results to other methods. Finally, some concluding notes and suggestions for future work are presented in Section 5.

2 Theoretical Background

In this section, we explain the necessary concepts and techniques related to our proposal. We first introduce some graph notions to present the core delineation algorithm of our proposal: *Image Foresting Transform* (IFT) [8] framework.

2.1 Graph

A *video* V can be represented as a pair $\mathsf{V} = (\mathcal{V}, \mathbf{I})$ in which $\mathcal{V} \subseteq \mathbb{N}^3$ denotes the set of *volume elements* (*i.e.*, voxels), and $\mathbf{I}$ maps every $v \in \mathcal{V}$ to a feature vector $\mathbf{I}(v) \in \mathbb{R}^m$. One can see that, for $m = 3$, V is a colored video (*e.g.*, RGB or CIELAB colorspaces). It is possible to create a *simple graph* (*i.e.*, no loops and no parallel edges) $\mathsf{G} = (\mathcal{N}, \mathcal{E}, \mathbf{I})$, derived from V, in which $\mathcal{N} \subseteq \mathcal{V}$ denotes the *vertex* set and $\mathcal{E} \subset \mathcal{N}^2$, the *edge* set. Two nodes $v_i, v_j \in \mathcal{N}$ are said to be *adjacent* if $(v_i, v_j) \in \mathcal{E}$. In this work, the elements in $\mathcal{E}$ are *arcs* (*i.e.*, G is a *digraph*). Consider $\pi_{s \rightsquigarrow t} = \langle s = v_1, v_2, \ldots, v_n = t \rangle$ to be a finite sequence of adjacent nodes (*i.e.*, a *path*) in which $(v_i, v_{i+1}) \in \mathcal{E}$ for $1 \leq i < n$. For simplicity, we may omit the path *origin* voxel by writing π_t. For $n = 1$, $\pi_t = \langle t \rangle$ is said to be *trivial.* We denote the *extension* of a path π_s by an arc $(s,t) \in \mathcal{E}$ as $\pi_s \cdot \langle s, t \rangle$ with the two instances of s being merged into one.

2.2 IFT

The *Image Foresting Transform* (IFT) [8] is a framework for the development of image processing operators based on connectivity and has been used to reduce image processing tasks as optimum-path forest computations over the image graph. As indicated by the authors [8], the IFT is independent of the input's dimensions and, therefore, the relation between pixels (or voxels) in such dimensionality can effectively be represented by an *adjacency relation* between them. In this work, we consider the IFT version restricted to a *seed set* $\mathcal{S} \subset \mathcal{N}$.

For a given arc $(s,t) \in \mathcal{E}$, it is possible to assign a non-negative *arc-cost* value $\mathbf{w}_*(s,t) \in \mathbb{R}^+$ through an *arc-cost function* $\mathbf{w}_*$. A common approach is to compute the ℓ_2-norm between the nodes' features — *i.e.,* $\|\mathbf{I}(s) - \mathbf{I}(t)\|_2$ for $s, t \in \mathcal{N}$. Consider $\mathit{\Pi}_{\mathsf{G}}$ the set of all possible paths in G. Then, a *connectivity function* $\mathbf{f}_*$ maps every path in $\mathit{\Pi}_{\mathsf{G}}$ to a *path-cost value* $\mathbf{f}_*(\pi_t) \in \mathbb{R}^+$. One of the most effective connectivity functions for object delineation is the $\mathbf{f}_{\max}$ function:

$$\begin{aligned} \mathbf{f}_{\max}(\langle t \rangle) &= \begin{cases} 0 & \text{if } t \in \mathcal{S}, \\ +\infty & \text{otherwise} \end{cases} \\ \mathbf{f}_{\max}(\pi_s \cdot \langle s, t \rangle) &= \max\{\mathbf{f}_{\max}(\pi_s), \mathbf{w}_*(s,t)\} \end{aligned} \tag{1}$$

A path π_t^* is said to be *optimum* if, for any other path $\tau_t \in \mathit{\Pi}_{\mathsf{G}}$, $\mathbf{f}_*(\pi_t^*) \leq \mathbf{f}_*(\tau_t)$.

Let $\mathbf{C}$ be a *cost map* in which assigns, to every path $\pi_t \in \mathit{\Pi}_{\mathsf{G}}$, its respective path-cost value $\mathbf{f}_*(\pi_t)$. The IFT algorithm minimizes $\mathbf{C}(t) = \min_{\forall \pi_t \in \mathit{\Pi}_{\mathsf{G}}}\{\mathbf{f}_*(\pi_t)\}$ whenever $\mathbf{f}_*$ satisfies certain conditions [5]. First, the IFT assigns path-costs to all trivial paths accordingly and, then, it computes optimum paths in a non-decreasing order, from the seeds to the remaining nodes in the graph. Therefore, independently if $\mathbf{f}_*$ suffices the desired properties in [5], the IFT always generates a spanning forest and, consequently, each supervoxel is an unique tree. During the segmentation process, a *predecessor map* $\mathbf{P}$ is generated and defined. Such map assigns any node $t \in \mathcal{N}$ to its *predecessor* s in the optimum path $\pi_s^* \cdot \langle s, t \rangle$, or to a distinctive marker $nil \notin \mathcal{N}$ — in such case, t is said to be a *root* of $\mathbf{P}$. In this work, every seed is a root of $\mathbf{P}$. One may see that $\mathbf{P}$ is a representation of an *optimum-path forest*, and it allows to recursively obtain the optimum-path root $\mathbf{R}(t)$ of t and its root's label $\mathbf{L}(\mathbf{R}(t))$.

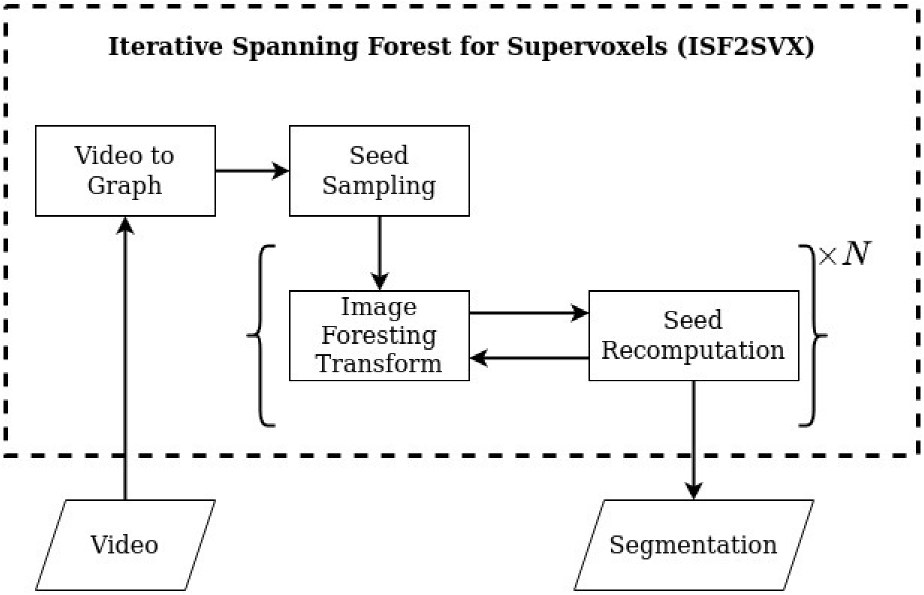

Iterative Spanning Forest for Supervoxels (ISF2SVX)
Video to Graph
Seed Sampling
Image Foresting Transform
Seed Recomputation
×N
Video
Segmentation

Fig. 2. Diagram of the proposed methodology for video supervoxel segmentation.

3 A strategy for supervoxel computation based on Iterative Spanning Forest

In this section, we present our approach for supervoxel computation based on *Iterative Spanning Forest* (ISF) [19] superpixel framework. Our proposal, so-called ISF2SVX, adopts a four step methodology: (a) graph creation; (b) seed sampling; (c) IFT-based supervoxel delineation; and (d) seed set recomputation. This pipeline is illustrated in Figure 2. Although our framework permits conceiving uncountable distinct variants, in this work, we assess some of the latest findings proposed by the ISF-based *Dynamic and Iterative Spanning Forest* (DISF) [2] method, which has proven to be more effective than state-of-the-art superpixel segmentation methods.

3.1 Video to Graph

Differently from 2D and 3D images, the presence of the same object in between frames imposes a major challenge for generating temporally coherent supervoxels. Therefore, it is recommended that the arcs and their respective arc-costs should reflect such condition. In this work, we operate on a video $\mathsf{V} = (\mathcal{V}, \mathbf{I})$ whose graph $\mathsf{G} = (\mathcal{N}, \mathcal{E}, \mathbf{I})$ is modeled as a single volume of nodes, and the outgoing arcs $(s, t) \in \mathcal{E}$ of a node s, for any $t \in \mathcal{N}$ which $s \neq t$, are defined, for instance, by an *adjacency relation* (*e.g.*, 26-adjacency). Another possibility to transform the video into a graph may consider motion information, like optical flow, in order to guide the edge creation, however this strategy is out-of-the-scope of this work since our main aim here is to study the behaviour of a simple adjacency relation.

3.2 Seed Sampling

For a given graph G, the second step generates the seed set $\mathcal{S} \subset \mathcal{N}$ for the first iteration of the IFT algorithm. In [2], the authors pointed out the drawbacks of initially sampling a number $N_0 \in \mathbb{N}$ of seeds approximate to the desired number $N_f \in \mathbb{N}$ of superpixels (or supervoxels). The relevance of a seed — which promotes effective delineation — is related to its location in a graph and, therefore, a strategy of oversampling may overcome the latter by increasing the probability of such seed being inserted in $\mathcal{S}$.

Most methods adopt a grid sampling scheme [1] (hereinafter named GRID) by selecting equally distanced seeds within the graph. Given a desired number $N_0 \in \mathbb{N}$ of seeds, and by computing an approximate supervoxel size $s = \frac{\mathcal{N}}{N_0}$, one can determine the stride d between seeds as $\sqrt[n]{s}$, where n denotes the data dimensionality — *i.e.*, $n = 2$ and $n = 3$, for 2D images, and for 3D images and videos, respectively. Finally, for avoiding seeds in high contrast regions (*i.e.*, probable object boundaries), the seeds are shifted to the lowest gradient position defined in an 8- or 26- neighborhood, for $n = 2$ or $n = 3$, respectively.

Considering an oversampling GRID strategy, d decreases sharply and, therefore, the proximity between seeds favors extreme competition, which often leads to better object delineation [2]. However, due to the excessive number of seeds, one can presume that a random selection of N_0 initial seeds can result in an even distribution in the graph, without compromising the selection of relevant ones. In this work, we propose such random oversampling strategy, named RND.

3.3 Supervoxel Generation

Once seeds are sampled, the supervoxels are generated using the IFT algorithm considering a connectivity function $\mathbf{f}_*$ and an arc-cost function $\mathbf{w}_*$. In this work, we consider the $\mathbf{f}_{\max}$ connectivity function for computing the path-costs.

In [19], the authors recall an arc-cost function $\mathbf{w}_1(p,q) = (\alpha\|\mathbf{I}(\mathbf{R}(p)) - \mathbf{I}(q)\|_2)^\beta + \|p - q\|_2$ in which $\alpha \in \mathbb{R}^+_*$ permits the user to control the regularity of the superpixels, and to control their adherence to boundaries through a factor $\beta \in \mathbb{R}^+_*$. However, superpixel and supervoxel regularity tends to prejudice the object delineation performance [2].

In DISF, the arc-costs are computed dynamically considering mid-level superpixel features, using a function first proposed in [3]. Let $\mathcal{T}_x \subset \mathcal{N}$ be an optimum-path *growing* tree rooted in a node $x \in \mathcal{N}$, and let $\mu(\mathcal{T}_x)$ be its mean feature vector. Then, the arc-cost function $\mathbf{w}_2$ can be formally defined as $\mathbf{w}_2(p,q) = \|\mu(\mathcal{T}_{\mathbf{R}(p)}) - \mathbf{I}(q)\|_2$. The function $\mathbf{w}_2$ has proven to be more effective than classic arc-cost functions for both superpixel segmentation [2] and for interactive image segmentation [3].

However, since the arc-costs are computed dynamically, $\mathbf{w}_2$ may generate discrepant segmentations, especially in regions with distinct colors, but equal gradient variation. Moreover, the order of arc evaluation during the IFT may also affect the aforementioned results. Therefore, in this work, we address such instability by computing a root-based arc-cost function $\mathbf{w}_3 = \|\mathbf{I}(\mathbf{R}(p)) - \mathbf{I}(q)\|_2$. Although it is not a dynamic estimation, root-based functions often present top delineation performance [19].

3.4 Seed Recomputation

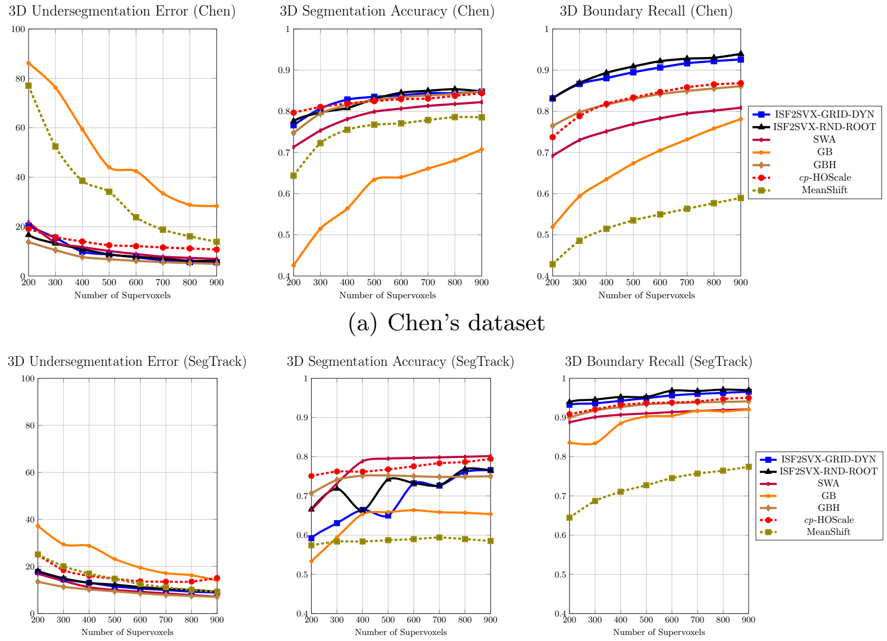
In ISF2SVX, the fourth step aims to update the seed set $\mathcal{S}$ in order to improve the supervoxel delineation for subsequent iterations. Such update can be performed by including, shifting or removing the seeds in $\mathcal{S}$, but respecting as much as possible the desired final number of supervoxels $N_f \in \mathbb{N}$ in the last iteration. Since, in this work, an oversampling strategy is presented, it is important to note that $N_0 \gg N_f$.

Since the presence of relevant seeds is expected — due to oversampling —, in [2], the authors proposed a new methodology for seed recomputation: removing irrelevant seeds based on a certain criterion. The motivation for that consists in promoting the growth of relevant superpixels (or supervoxels), by removing the irrelevant ones and maintaining the competition among the primers. At each iteration $i \in \mathbb{N}$, $\mathbf{M}(i) = \max\{N_0 \exp^{-i}, N_f\}$ relevant seeds are maintained for the subsequent iteration $i+1$, while the remaining ones are discarded. In DISF, the stopping criterion is reaching the desired number of superpixels, which is often less than 10 — a common value for many iterative methods.

The $\mathbf{M}(i)$ relevant seeds may be selected by a combination of their sizes and contrast [2] in which the former indicates the supervoxel's growth ability, and the latter, whether the supervoxel is located in a homogeneous region (thus, probably irrelevant). Let $\mathcal{B}$ be a *tree adjacency relation*, which defines the immediate neighbors of any supervoxel. Then, with the use of a priority queue, a relevance of a seed s can be measured by a function $\mathbf{V}(s) = \frac{|\mathcal{T}_s|}{|\mathcal{N}|} \min_{\forall(\mathcal{T}_s,\mathcal{T}_r)\in\mathcal{B}}\{\|\mu(\mathcal{T}_s) - \mu(\mathcal{T}_r)\|_2\}$, which $\mathcal{T}_r$ is an *adjacent supervoxel* of $\mathcal{T}_s$.

4 Experimental analysis

We evaluated all methods considering the Chen [4] and Segtrack [18] datasets, both containing groundtruth annotations. Using the LIBSVX [23] library, we selected five classic evaluation metrics: (a) 3D boundary recall (BR); (b) 3D segmentation accuracy (SA); (c) 3D undersegmentation error (UE); (d) Explained variation (EV); and (e) Mean duration. BR measures the quality of the spatiotemporal boundary delineation, while SA measures the fraction of groundtruth segments which are correctly classified (*i.e.*, higher is better for both). UE calculates the fraction of object supervoxels overlapping background voxels — and vice-versa — (*i.e.*, lower is better). EV measures the method's ability to describe the video's color variations through its supervoxels (*i.e.*, higher



(a) Chen's dataset

(b) Segtrack dataset

Fig. 3. A comparison between our method ISF2SVX, and the methods *cp*-HOScale GB, GBH, SWA, and MeanShift when applied to Chen and SegTrack datasets. The comparison is based on the following metrics: (i) 3D undersegmentation error; (ii) 3D segmentation accuracy; (iii) 3D boundary recall .

is better). Finally, the mean supervoxel duration measures if a supervoxel perpetuates throughout the frames, indicating a temporal coherence to the object which it compounds (*i.e.*, higher is better).

In this work, we propose two ISF2SVX variants. One, named ISF2SVX-GRID-DYN, oversamples using GRID and computes supervoxels considering the arc-cost function $\mathbf{w}_2$. The other, ISF2SVX-RND-ROOT, oversamples using RND, and considers the $\mathbf{w}_3$ function. We compared our approaches with different state-of-the-art methods: (i) GB [9]; (ii) GBH [12]; (iii) SWA [7]; (iv) MeanShift [15]; and (v) *cp*-HOScale [17]. The number of supervoxels varied from 200 to 900 and, for the baselines, the recommended parameter settings were used.

4.1 Quantitative analysis

Considering the undersegmentation error, it is possible to observe in the plots in Figure 3 that both of the ISF2SVX variations managed to be compatible or even better than the compared works. Although for a smaller number of supervoxels, the GBH method performs slightly better in both Chen and Segtrack dataset, one can notice that after 700 supervoxels our methods are consistent with GBH.

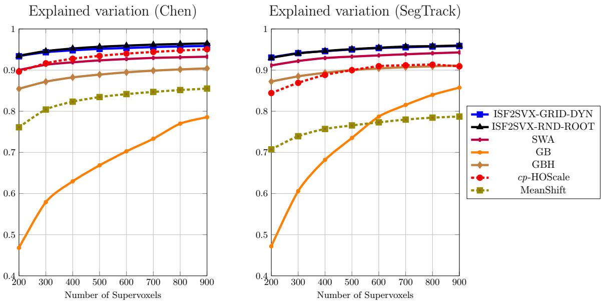


Fig. 4. A comparison between our method ISF2SVX, and the methods *cp*-HOScale GB, GBH, SWA, and MeanShift when applied to Chen and SegTrack datasets. The comparison is based on the explained variation.

This may indicate that the construction of the hierarchy is not so beneficial to prevent supervoxel leaks, since region merging errors can occur and make the leak even more evident.

When taking into account the segmentation accuracy (Figure 3), we notice that ISF2SVX has produced well-defined segmentations as well as some of the compared baseline methods. Although it is possible to detect an instability related to the segmentation accuracy in the Segtrack dataset, this instability is given by a single video whose object of interest is significantly small and more than one supervoxel composes it. Thus, the calculation of this metric and, consequently, the average performance were affected due to the small dataset size.

Boundary recall results also indicate that ISF2SVX is superior to all methods. We are able to observe that for Chen dataset our approaches can yield even better metrics that its competitors compared to the Segtrack dataset (Figure 3). This can be associated with the fact that path-based, and more specifically IFT-based, methods are known to be effective solutions in object delineation [2, 19].

In Figure 4 is possible to observe that, since IFT minimizes the accumulated cost of the path, the internal variation of supervoxels tends to be greatly minimized. Thus, we can see better explained variation metric results for our variations when compared to previous studies. In addition, as there are no regularity constraints, the competition between seeds becomes more intense and, therefore, leads to a lower probability of incorporating dissimilar voxels. Furthermore, as one can see in Figure 5, ISF2SVX, in both variants, outperforms the other methods in terms of mean duration. It is worth to mention that this metric tries to capture the temporal coherence of the supervoxels.

4.2 Qualitative analysis

In Figure 6, we compare the variant ISF2SVX-GRID-DYN with the baselines in a single video. As one may see, our approach manages to generate large supervoxels in non-significant regions (*e.g.*, the grass), while effectively delineates even small important regions (*e.g.*, the player's head). In contrast, for 100 supervoxels, all baselines generates too many small and irrelevant supervoxels. For 20 supervoxels, such quantity is severely reduced at the expense of degrading the object delineation performance.

When comparing the mean execution time in such video — given an interval of $[200, 900]$ supervoxels —, ISF2SVX obtains a speed-up of 5.9 and 7.1 against the second and third fastest baselines (*i.e.*, GBH and *cp*-HOScale, respectively). Although *cp*-HOScale manages to compute the whole hierarchy — thus is capable of obtaining many segmentations without requiring any recomputation —, it is unlikely that, in an application, the user would need to manipulate all the levels, and not a small subset of those (*i.e.*, use a dense over a sparse hierarchy). Finally, the speed-up of ISF2SVX over GB is 0.95, being slightly slower than GB. However, due to recent findings [10, 6] and for a suitable definition of components (*e.g*, GRID sampling and $\mathbf{w}_3$ arc-cost function), it is possible to further improve the speed of ISF2SVX without prejudicing the object delineation performance.

5 Final Remarks and Future Studies

In this paper, we propose a new supervoxel segmentation framework, named *Iterative Spanning Forest for Supervoxels* (ISF2SVX), which was inspired by the *Iterative Spanning Forest* (ISF) superpixel segmentation framework. Our approach not only benefits from recent improvements in superpixel segmentation, but also permits the development of effective video segmentation algorithms through the definition of its components. Results show that ISF2SVX variants outperforms state-of-the-art methods with a great margin in two datasets, especially in terms of delineation and color description (by its supervoxels).

For further works, we would like to study the behaviour of ISF2SVX considering more descriptive and discriminative arc-cost functions for video segmentation since the ones presented here relies on the color gradient between an element and its conquering tree (or root). Moreover, instead of early video segmentation (or supervoxel generation), we will study strategies for streaming the video segmentation method. Furthermore, strategies for seed oversampling location will be an interesting direction since we may learn good positions to the set of seeds.

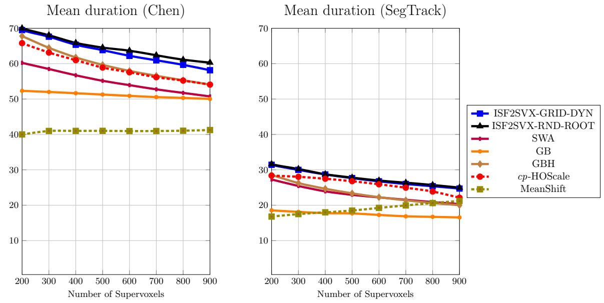
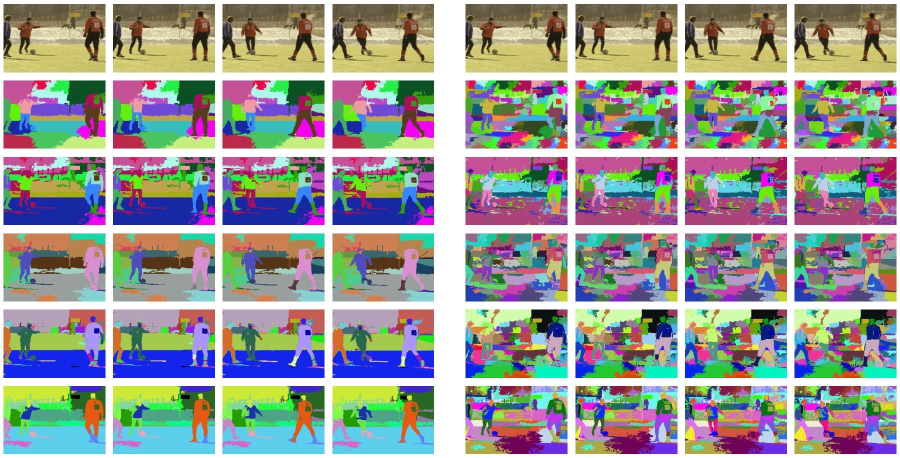


Fig. 5. A comparison between our method ISF2SVX, and the methods *cp*-HOScale GB, GBH, SWA, and MeanShift when applied to Chen and SegTrack datasets. The comparison is based on the mean duration.



(a) 20 supervoxels

(b) 100 supervoxels

Fig. 6. Example extracted from Chen dataset. The first row are the original frames, the following rows, from top to bottom are results with 20 and 100 supervoxels obtained from GB, GBH, SWA, *cp*-HOScale, and ISF2SVX-GRID-DYN.

References

1. Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S.: Slic superpixels compared to state-of-the-art superpixel methods. Transactions on Pattern Analysis and Machine Intelligence **34**(11), 2274–2282 (2012)
2. Belém, F., Guimarães, S., Falcão, A.: Superpixel segmentation using dynamic and iterative spanning forest. Signal Processing Letters **27**, 1440–1444 (2020)
3. Bragantini, J., Martins, S.B., Castelo-Fernandez, C., Falcão, A.X.: Graph-based image segmentation using dynamic trees. In: Iberoamerican Congress on Pattern Recognition. pp. 470–478. Springer (2018)
4. Chen, A., Corso, J.: Propagating multi-class pixel labels throughout video frames. In: Western New York Image Processing Workshop. pp. 14–17 (2010)
5. Ciesielski, C., Falcão, A., Miranda, P.: Path-value functions for which dijkstra's algorithm returns optimal mapping. Journal of Mathematical Imaging and Vision **60**(7), 1025–1036 (2018)
6. Condori, M.A., Cappabianco, F.A., Falcão, A.X., Miranda, P.A.: An extension of the differential image foresting transform and its application to superpixel generation. Journal of Visual Communication and Image Representation **71**, 102748 (2020)
7. Corso, J., Sharon, E., Dube, S., El-Saden, S., Sinha, U., Yuille, A.: Efficient multilevel brain tumor segmentation with integrated bayesian model classification. Transactions on Medical Imaging **27**(5), 629–640 (2008)
8. Falcão, A., Stolfi, J., Lotufo, R.: The image foresting transform: theory, algorithms, and applications. Transactions on Pattern Analysis and Machine Intelligence **26**(1), 19–29 (2004)
9. Felzenszwalb, P., Huttenlocher, D.: Efficient graph-based image segmentation. International Journal of Computer Vision **59**(2), 167–181 (2004)

10. Gonçalves, H.M., de Vasconcelos, G.J., Rangel, P.R., Carvalho, M., Archilha, N.L., Spina, T.V.: cudaift: 180x faster image foresting transform for waterpixel estimation using cuda. In: VISIGRAPP (4: VISAPP). pp. 395–404 (2019)
11. Griffin, B.A., Corso, J.J.: Video object segmentation using supervoxel-based gerrymandering. arXiv preprint arXiv:1704.05165 (2017)
12. Grundmann, M., Kwatra, V., Han, M., Essa, I.: Efficient hierarchical graph-based video segmentation. In: Computer Vision and Pattern Recognition (CVPR). pp. 2141–2148. IEEE (2010)
13. Oneata, D., Revaud, J., Verbeek, J., Schmid, C.: Spatio-temporal object detection proposals. In: eccv. pp. 737–752. Springer (2014)
14. Papon, J., Abramov, A., Schoeler, M., Worgotter, F.: Voxel cloud connectivity segmentation-supervoxels for point clouds. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2027–2034 (2013)
15. Paris, S., Durand, F.: A topological approach to hierarchical segmentation using mean shift. In: 2007 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2007)
16. Sheng, H., Zhang, X., Zhang, Y., Wu, Y., Chen, J., Xiong, Z.: Enhanced association with supervoxels in multiple hypothesis tracking. IEEE Access **7**, 2107–2117 (2018)
17. Souza, K., Araújo, A., Patrocínio Jr., Z., Guimarães, S.: Graph-based hierarchical video segmentation based on a simple dissimilarity measure. Pattern Recognition Letters **47**, 85–92 (2014)
18. Tsai, D., Flagg, M., Nakazawa, A., Rehg, J.: Motion coherent tracking using multi-label mrf optimization. International Journal of Computer Vision **100**(2), 190–202 (2012)
19. Vargas-Muñoz, J., Chowdhury, A., Alexandre, E., Galvão, F., Miranda, P., Falcão, A.: An iterative spanning forest framework for superpixel segmentation. Transactions on Image Processing **28**(7), 3477–3489 (2019)
20. Verdoja, F., Thomas, D., Sugimoto, A.: Fast 3d point cloud segmentation using supervoxels with geometry and color for 3d scene understanding. In: 2017 IEEE International Conference on Multimedia and Expo (ICME). pp. 1285–1290. IEEE (2017)
21. Wang, B., Chen, Y., Liu, W., Qin, J., Du, Y., Han, G., He, S.: Real-time hierarchical supervoxel segmentation via a minimum spanning tree. Transactions on Image Processing **29**, 9665–9677 (2020)
22. Wu, F., Wen, C., Guo, Y., Wang, J., Yu, Y., Wang, C., Li, J.: Rapid localization and extraction of street light poles in mobile lidar point clouds: A supervoxel-based approach. IEEE Transactions on Intelligent Transportation Systems **18**(2), 292–305 (2016)
23. Xu, C., Corso, J.: LibSVX: A supervoxel library and benchmark for early video processing. International Journal of Computer Vision **119**(3), 272–290 (2016)
24. Xu, C., Xiong, C., Corso, J.: Streaming hierarchical video segmentation. In: European Conference on Computer Vision. pp. 626–639. Springer (2012)