On Morphological Hierarchies for Image Sequences

Caglayan Tuna, Alain Giros, François Merciol, Sébastien Lefèvre

On Morphological Hierarchies for Image Sequences

Caglayan Tuna
Université Bretagne Sud, IRISA
Vannes, France
caglayan.tuna@irisa.fr

Alain Giros
CNES
Toulouse, France
alain.giros@cnes.fr

François Merciol
Université Bretagne Sud, IRISA
Vannes, France
francois.merciol@irisa.fr

Sébastien Lefèvre
Université Bretagne Sud, IRISA
Vannes, France
sebastien.lefevre@irisa.fr

Abstract—**Morphological hierarchies form a popular framework aiming at emphasizing the multiscale structure of digital image by performing an unsupervised spatial partitioning of the data. These hierarchies have been recently extended to cope with image sequences, and different strategies have been proposed to allow their construction from spatio-temporal data. In this paper, we compare these hierarchical representation strategies for image sequences according to their structural properties. We introduce a projection method to make these representations comparable. Furthermore, we extend one of these recent strategies in order to obtain more efficient hierarchical representations for image sequences. Experiments were conducted on both synthetic and real datasets, the latter being made of satellite image time series. We show that building one hierarchy by using spatial and temporal information together is more efficient comparing to other existing strategies.**

I. INTRODUCTION

Hierarchical methods in mathematical morphology, a.k.a. morphological hierarchies represented in tree-based structures, offer an efficient framework for multiscale image analysis [1]. Tree representations have been used for various tasks, including image filtering [2], image segmentation [3] and object detection [4], *etc*. during last decade. Due to the increasing amount and availability of temporal data, efficient processing of image sequences is now a strong requirement in several fields. Lately, hierarchical representations have been used for the analysis of multimodal data [5] and image sequences [6].

In a previous work [6], we have introduced 3 different strategies to build hierarchical representations for image sequences: spatial hierarchy, temporal hierarchy and spatio-temporal hierarchy. Each of these strategies offers a specific insight on the spatio-temporal data that is useful for some tailored applications. Still, comparing the hierarchies built from these strategies, and especially their intrinsic structures, remains necessary to provide this novel framework with proper analytical studies. However, evaluation and comparison of morphological hierarchies remains limited to practical comparisons in the literature.

Indeed, evaluation of morphological hierarchies has been performed in [7] in an application-oriented scenario focused on image segmentation. Similarly, strategies to build binary partition trees were evaluated according to their segmentation capability in [8]. Tree representations were also compared when used in a filtering process [9], and in the context of tree matching [10] when searching for similar objects. This latter approach requires a reference tree structure. In this paper, we aim to compare morphological hierarchies according to their structures rather than based on some practical applications. To the best of our knowledge, structure-based comparison of morphological hierarchies is not common in literature.

We remind that morphological hierarchies can be seen as the result of a hierarchical clustering (HC) conducted on the image. Evaluation of hierarchical clustering (HC) quality has been studied by data analysis community. The Rand Index [11] is one of the aged quality metric and it is still in use to find better HC [12]. Unfortunately, this index is not appropriate for morphological hierarchies due to the unavailability of ground truth clusters. Recently, Dasgupta proposed an unsupervised cost function [13] aiming to determine the quality of hierarchical structures. This cost function is receiving an increasing interest in the HC community [14], [15], [16]. In this paper, we will adapt this metric to morphological hierarchies in order to compare spatial, temporal and spatio-temporal hierarchies according their structural cost. Besides, we will also compare the hierarchies based on their number of nodes and their filtering capability.

Our contributions in this paper are three-fold. First, we show how to project a spatio-temporal tree structure along temporal or spatial dimensions. This is required to map different hierarchies in a common space and to allow their comparison. Second, we review different criteria to assess and compare the structural properties of the hierarchies built from the different strategies introduced in [6]. Finally, we show the superiority of the spatio-temporal hierarchy, for which we propose a new connectivity rule in order to fully exploit the available temporal information. We support our claims and conclusions using synthetic and real datasets, the latter being made of satellite image time series (SITS).

This paper is organized as follows. In Sec. II, we provide mathematical background on hierarchies for still images as well as image sequences. We address in Sec. III the projection of the spatio-temporal hierarchy along the temporal or spatial dimensions. We present the structure assessment criteria used in our study in Sec. IV, and the synthetic and real datasets we are using in Sec. V. We report our experimental results in Sec. VI, before concluding our paper in Sec. VII.

II. BACKGROUND ON MORPHOLOGICAL HIERARCHIES

According to the literature [17], there are several ways to build a morphological hierarchy (a.k.a tree) from an image.

Trees have different building rules and they are named according to their building rule. For instance, maximum values of the image are placed in the leaves of the max-tree [18], while a min-tree has leaves consisting in local minima. We will apply max and min-tree here, but for the sake of simplicity, we will consider only the max-tree notation in this paper. Let us recall that the min-tree is the dual representation of the max-tree, and can be built from the max-tree of the image complement (or negative, invert). However, it should be noted that our proposed comparison methodology can be implemented with other tree-based representations such as tree of shape [19], α-tree [20], or binary partition tree [21].

A. Definition for a Single Image

Let us define I as a gray scale image, Ω as non empty finite set, x as the coordinates of this image and $I(x)$ as the intensity value of this location. First, the upper threshold set is defined as $L^\lambda(I) = \{x \in \Omega \mid I(x) \geq \lambda\}$ according to level λ. This λ varies between maximum (in leaves) and minimum (in root) intensity levels of the image. Ω can be seen as a graph built according to a standard 4- or 8-adjacency relation on the 2D grid. Within each level set, neighboring pixels are grouped together in connected components according to their location and intensities. We consider the set of nodes V gathering all connected components,

$$V(I) = \bigcup_{\forall k, \lambda} C_\lambda^k(I) \tag{1}$$

where C denotes a connected component, k its index in the L^λ of the image I. For the sake of concision, we will omit I from C and V notations in the rest of this paper. Any two nodes are either disjoint or nested. Nodes at the same level have no intersection;

$$\forall k \neq k', C_\lambda^k \cap C_\lambda^{k'} = \emptyset. \tag{2}$$

We will omit k when only one node is enough for the relevant expression. Connected components become nodes of the tree. A tree T consists of set of nodes V and a set of edges E, i.e. $\mathcal{T} = (V, E)$. A tree is an acyclic, directed graph from its unique root to the leaves. An edge E exists between two nodes C_λ and $C_{\lambda'}$ if for $\lambda < \lambda'$ we have $C_{\lambda'} \subset C_\lambda$ and there is no other node in between, i.e. $\nexists C_{\lambda''} : C_{\lambda'} \subset C_{\lambda''} \subset C_\lambda$. The difference between the parent and child nodes is the set of additional pixels which appear and become connected to the parent node at its level λ:

$$C_\lambda \setminus C_{\lambda'} = \{x \in \Omega \mid I(x) = \lambda\}. \tag{3}$$

The additional pixels with value $I(x) = \lambda$ do not belong to any child node of C_λ. They are called ghost pixels and discussed in details in [17]. Each pixel of the image is pretended to have one additional pixel in the same gray level, and these additional ghost pixels are only connected to their original versions. The goal is to avoid information loss and to consider these level set-based trees as possible results of a hierarchical clustering. We will denote ghost nodes as C'. Let C_λ' is a

(a) Spatial hierarchy   (b) Temporal hierarchy
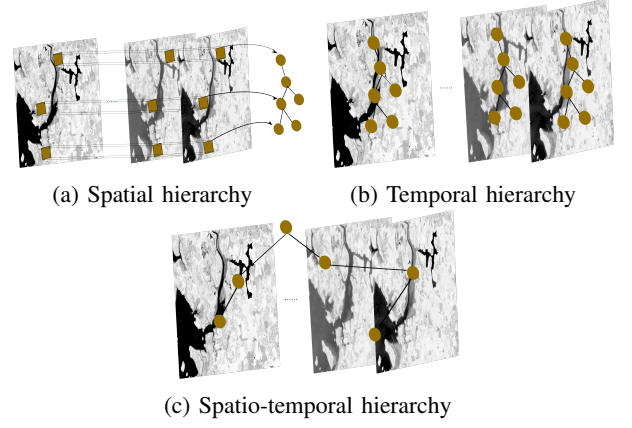
(c) Spatio-temporal hierarchy

Fig. 1: Morphological hierarchies for image sequences (taken from [6]).

ghost node and $C_{\lambda'}$ is the only child of its parent. Then, the parent node of $C_{\lambda'}$ can be defined as

$$C_\lambda = C_{\lambda'} \cup C_\lambda'. \tag{4}$$

Ghost nodes can be defined also as a leaf nodes because they do not have any child by definition.

B. Extension to Image Sequences

We reuse here notations from [6] where morphological hierarchies for image sequences were addressed as illustrated in Fig. 1. We will use the τ subscript to denote the presence of temporal information (i.e. temporal support), i.e. writing I_τ an image sequence instead of the single image I, and $C_{\lambda,\tau}$ the spatio-temporal connected components. Conversely, the ς subscript will be used to emphasize timeless information. We recall here the three different strategies introduced in [6].

1) Spatial hierarchy: According to the literature on multi-variate morphology [22], ordering and projection are the two main ways to build a tree from vectorial data. The spatial hierarchy (SH) strategy requires converting each pixel vector to one scalar value. In the case of ordering, first we order all pixel time series, e.g. according to their distance from a reference vector. Then we can rank them and use the rank information to build the hierarchy. As far as projection is concerned, the idea is to project each pixel time series into a single scalar, timeless spatial value using a given function. Examples of such functions $f : I_\tau(x) \mapsto I_\varsigma(x)$ include mean or median. In our experiments, we will consider the distance-based ranking method to build the rank image and the subsequent spatial tree (noted $\mathcal{T}_\varsigma$ in the sequel). The methodology proposed in this paper can be straightforwardly followed with other SH-related approaches.

2) Temporal hierarchy: Alternatively, the temporal hierarchy (TH) aims to provide a temporal set of trees instead of a single, timeless one as with SH. For a given image sequence I_τ, every single image I_t of the series leads to a given tree $\mathcal{T}_t$. The final set is defined as

$$\mathcal{T}_\tau = \{\mathcal{T}_t\}_{t=[1,n]} \tag{5}$$

where n represent the amount of images in the set. This strategy relates to the marginal approach in the literature [23] where trees are built and subsequently processed separately.

3) Spatio-temporal hierarchy: The last strategy (a.k.a. known as spatio-temporal hierarchy, STH) stacks all frames of the sequence in a spatio-temporal cube, and builds one single tree from the whole set of data. While the trees built with the two previous strategies SH and TH have only a spatial support (shall it be timeless $\mathcal{T}_\varsigma$ or related to a single frame $\mathcal{T}_t$), here the tree intrinsically combines both spatial and temporal dimensions. We call this representation a space-time tree. It is made of nodes with a spatio-temporal support $C_{\lambda,\tau}$, combined in a set V as defined in (1). Furthermore, we can split a spatio-temporal connected component C into its spatial slices extracted at each time stamp t:

$$C_{\lambda,\tau}^k = C_{\lambda,1}^k \cup C_{\lambda,2}^k \cup \ldots \cup C_{\lambda,n}^k. \tag{6}$$

As already indicated, the connected components are built by gathering neighboring pixels following a given spatial adjacency relation, e.g. 4- or 8-connectivity. As we now deal with spatial-temporal connected components and not spatial-only connected components, connectivity rules should be revised accordingly. To illustrate, let us consider the 8-connectivity. The corresponding neighbour set $\mathcal{N}_8$ for a given location $x = (i,j)$ can be defined as

$$\mathcal{N}_8(x) = \{(i',j') \neq (i,j) \mid \max(|i-i'|, |j-j'|) = 1\}. \tag{7}$$

It can be extended to time using 26-connectivity, where a location $(x,t) = (i,j,t)$ has a neighbour set $\mathcal{N}_{26}$:

$$\mathcal{N}_{26}(i,j,t) = \{(i',j',t') \neq (i,j,t) \\ \mid \max(|i-i'|, |j-j'|, |t-t'|) = 1\}. \tag{8}$$

In this paper, we also consider another connectivity rule, called continuous connectivity since the neighbours along the time dimension are not limited to previous and next images. It considers a set $\mathcal{N}_\tau$ of $3 \times 3 \times n - 1$ neighbours:

$$\mathcal{N}_\tau(i,j,t) = \{(i',j',t') \neq (i,j,t) \\ \mid \max(|i-i'|, |j-j'|) = 1\}. \tag{9}$$

Let us observe that this continuous connectivity can even be applied on the temporal dimension only, thus leading to a purely temporal neighbour set of size $n - 1$

$$\mathcal{N}_{\tau^*}(x,t) = \{(x,t') \neq (x,t)\}, \tag{10}$$

and we will also use these different definitions in our projections that will be presented in the next section.

III. TREE PROJECTION

The different strategies reviewed in the previous sections provide trees lying in different domains. Comparing the different trees produced with TH, SH and STH is then not trivial, and requires a common domain in which the comparison could be performed. Once a space-time tree has been built, it can be projected to temporal domain for comparison with

TH, or spatial domain for comparison with SH. But structural differences will be observable among the different trees thanks to the spatial-temporal connectivity rule considered with the space-time tree.

A. Projection in temporal domain

We define the projection of the space-time tree $\mathcal{T}_\tau = (V, E)$ in the temporal domain as the operator Γ that returns a tree defined with a spatial support only, given a specific time stamp $t = [1, n]$:

$$\Gamma^t : \mathcal{T}_\tau \mapsto \mathcal{T}^t. \tag{11}$$

The new tree $\mathcal{T}^t$ will retain nodes and pixels from the given time stamp t. To do so, the operator first removes the nodes which do not contain any pixels from t;

$$V \mapsto V' = V \setminus \{C_\lambda^k \mid C_{\lambda,t}^k = \emptyset\}, \tag{12}$$

thus leading to an intermediate tree $\mathcal{T}' = (V', E')$ where $E' \subseteq E$ only contains edges connecting nodes from V'.

Let us consider a leaf node containing pixels (x,t) present in I_t. Since parent nodes are defined as supersets of their children (see (4)), all the ancestors of the leaf also have support including I_t. However, nodes of $\mathcal{T}'$ may have a spatial support that goes beyond the selected time stamp t. We should then shrink the nodes in V' in the temporal dimension to limit their support to I_t:

$$V' \mapsto V'' = \{C_{\lambda,t}^k \mid C_\lambda^k \in V'\}. \tag{13}$$

Again, we can define a new tree $\mathcal{T}'' = (V'', E')$. Let us observe that since we have not removed any nodes from V' but have instead reduced their spatial support, there is no change in the set of edges E'.

The last operation consists in filtering the tree to remove the nodes that provide duplicate information. These nodes C are those for which there is no associate ghost nodes C' in the selected frame I_t, and whose added-value in the hierarchy was noticeable for some other time stamps. More formally,

$$\begin{aligned} V'' \mapsto V''' &= V'' \setminus \{C_\lambda^k \mid C_{\lambda,t}'^k = \emptyset\} \\ &= V'' \setminus \{C_\lambda^k \mid \nexists\, x : I_t(x) = \lambda\}, \end{aligned} \tag{14}$$

i.e. we remove the nodes with a level that is not corresponding to a visible gray scale in the image I_t. The resulting tree is then defined as $\mathcal{T}^t = (V''', E'')$ where the edge set E'' has been updated according to the retained nodes in V'''. Let us observe that although $V''' \subset V''$, the inclusion relation between edge sets is not kept, i.e. $E'' \not\subset E'$. Indeed, since some internal nodes have been removed, there are now new edges to connect their parent and children nodes. These edges are built as defined in Sec. II-A.

B. Projection in spatial domain

In order to allow comparison with SH, we consider to project the space-time tree in the spatial domain:

$$\Gamma^\varsigma : \mathcal{T} \mapsto \mathcal{T}^\varsigma. \tag{15}$$

Similarly to the previous projection, we rely on successive steps to build Γ^ς. But the main difference is that the resulting tree $\mathcal{T}^\varsigma$ is timeless and thus does not require any time stamp parameter t.

First, we filter out nodes for which timeless information cannot be extracted due to a limited temporal domain. These nodes are identified as those that do not appear in all time stamps of the series, leading to

$$V \mapsto V' = V \setminus \{C_\lambda^k \mid \exists t \colon C_{\lambda,t}^k = \emptyset\}. \tag{16}$$

This leads to a new tree $\mathcal{T}' = (V', E')$ with E' the updated set of edges limited to those connecting the nodes from V'. Similarly to the temporal projection, this first step is not enough since here $\mathcal{T}'$ is not timeless yet. To do so, a second operation is required to remove the spatial locations for which the time series has missing values. In other words, we spatially shrink each node to its locations that cover the full time series.

$$
\begin{aligned}
V' \mapsto V'' = V' \setminus \{C_\lambda^k \mid \\
\exists x, \exists t, \exists t' \neq t \colon x \in C_{\lambda,t}^k, x \notin C_{\lambda,t'}^k\}.
\end{aligned}
\tag{17}
$$

Since this step only reduces the spatial support of the nodes, it does not modify the edge set, thus we have $\mathcal{T}'' = (V'', E')$.

Similarly to what has been proposed for SH, we need to finally apply a projection function on the pixel time series for the timeless nodes. Since we consider here the max-tree, the proposed operation consists in finding the maximum value of each location, or equivalently removing the non-maximum pixels:

$$
\begin{aligned}
V'' \mapsto V''' = V'' \setminus \{(x,t) \mid (x,t) \in C_\lambda^k, \\
\exists t' \neq t, \exists \lambda' > \lambda \colon (x,t) \in C_{\lambda,t}^k, (x,t') \in C_{\lambda',t'}^{k'}\}
\end{aligned}
\tag{18}
$$

Finally, the tree resulting from the projection Γ^ς can be defined as $\mathcal{T}^\varsigma = (V''', E')$.

C. Properties of projected trees

Having defined the two projection methods Γ^t and Γ^ς, we know discuss some properties of the projected trees they return. Let us recall that $\mathcal{T}^\varsigma$ is the only possible projection of the space-time tree in the spatial domain. Conversely, n different $\mathcal{T}^t$ trees can be built from $\Gamma^t(\mathcal{T}^\tau)$. As shown previously, both projections provides only subsets of the initial node sets and do not create new nodes (but this property does not hold for the edge sets with the temporal projection).

Furthermore, we can study the trees in terms of isomorphism. Two trees are isomorphic one to each other if there is a one-to-one mapping from the vertices of one tree to the vertices of the other tree that preserves vertex, edges and labels [24]. We consider here that the label of the nodes is simply the level (λ) and it is preserved when the tree is projected. Let $\mathcal{T} = (E, V)$ be a tree, $\mathcal{T}' = (E', V')$ its sub-tree. There are three types of sub-trees according to the literature [24];

1) *bottom-up sub-trees* if the following three conditions hold: $V' \subseteq V$, $E' \subseteq E$, and if any node $C \in V$ and $C \in V'$, all descendants of C are in V';

2) *induced sub-trees* require $V' \subseteq V$ and $E' \subseteq E$;
3) and *embedded sub-trees* for which we have $V' \subseteq V$, and the ordering of nodes (ancestor and descendant relationship) is preserved.

According to the definition of sub-trees, $\mathcal{T}^\varsigma$ is the induced sub-tree of $\mathcal{T}_\tau$ and $\mathcal{T}^t$ is the embedded sub-tree of $\mathcal{T}_\tau$. The parent-child relationship is changing for $\mathcal{T}^t$ because of (14).

IV. COMPARISON OF TREE REPRESENTATIONS

Thanks to the projections introduced in the previous section, we can now compare a space-time tree with the other hierarchies. We show here three different methods to ensure such a comparison, respectively based on node analysis, cost function and filtering capability.

A. Node analysis

Measuring the number of nodes or connected components contained in a tree provides a simple way to assess its structure. For instance, the number of parent nodes has been suggested in [7]. The number of nodes or vertices refers to the complexity of a tree and it is denoted as $\mid \mathcal{T} \mid$. If this value is high, the tree is considered having a complex structure [25]. Such a complexity can also be observed with attribute analysis from leaves to root such as in [26], where the node attribute path from leaves to root is analysed through a so-called leaf attribute function (LAF). Every leaf has one path to root and the attribute value is changing from leaves to root. When the attribute value changes dramatically from one node to its parent, this node can be used as a significant feature. In order to find such elements, the LAF gradient is computed.

B. Dasgupta's Cost

Dasgupta's cost is an unsupervised measure of the quality of a hierarchical clustering [13]. In a component tree, each internal node can have a leaf node thanks to the modeling with ghost nodes [17]. A morphological hierarchy acts as a hierarchical clustering when ghost nodes are taken into account. Therefore, we adapt here this cost function to component trees in order to compare the cost of different tree representations. The cost of a tree is calculated as a weighted sum of a function involving each leaves' pair:

$$Cost(\mathcal{T}) = \sum_{a,b \in \mathcal{T}} w_{a,b} \cdot \mid leaves(\mathcal{T}[a \vee b]) \mid \tag{19}$$

where $\mathcal{T}[a \vee b]$ is the bottom-up sub-tree rooted at the lowest common ancestor (lca) of a and b leaf nodes and $w_{a,b}$ is the edge weight between nodes a and b. The interpretation of the cost function is to split data to each leaf by cutting edges from the lowest common ancestor. The cost of a tree $\mathcal{T}$ is the sum of the splitting costs from the leaves. A lower cost means a preferable tree. If an edge cut the tree close to the root, it will cause a high cost. It also means that similar objects should be penalized if they are merged in the higher nodes of the tree [27]. Here, each pair of leaf nodes is processed separately. Minimizing this cost provides an optimal tree for the associated graph.

Adapting this cost to morphological hierarchies requires finding edges between connected pixels. In (19), the edge weight information is calculated according to the underlying graph structure from which the tree was built. Gray scale images are vertex or edge weighted graphs [3] and component trees are built from these weighted graphs [28]. The edge weight can then be measured as the absolute difference of intensities in the image [29]. We have defined here the edge weight as the level (λ) difference between leaf pairs. Connected leaf pairs have been determined according to neighbour sets of each location with standard spatial-connectivity (7). Although space-time trees are built with different neighbour sets ((8), (9)), their projected versions lie in the spatial domain and as such they come with a neighbour set definition such as (7). When the trees to be compared come from the same initial image, their graph and edge weights are the same but the amount of leaves of their lca node might differ.

Let C_λ be a leaf node and $C'_{\lambda'}$ is a ghost node. Their lca node could be $C_{\lambda'}$. Cost for this leaves pair is calculated as $Cost = (\lambda' - \lambda) \cdot | C_{\lambda'} |$. This calculation is repeated for each leaves pair. The amount of leaves is equal to the amount of nodes including the ghost nodes.

C. Attribute Filtering

Tree filtering is a common operation used mainly for smoothing and feature extraction (*e.g.*, attribute profiles [30]). The main advantage of tree filtering is to output a non-blurred image where edges between connected components are preserved. When a filtered tree is obtained, the associate image can be reconstructed by assigning new values to the pixels that belong to the pruned nodes. There are three steps in the filtering process: tree building, tree pruning and image reconstruction from the filtered tree;

$$I \mapsto \mathcal{T} \tag{20}$$
$$\mathcal{T} \mapsto \mathcal{T}' = \gamma^h(\mathcal{T}) \tag{21}$$
$$\mathcal{T}' \mapsto I' \tag{22}$$

where I' is the filtered image obtained after filtering the tree with the γ filtering operator and the criteria h.

V. Data

A. Synthetic dataset

In order to compare tree representation strategies, we first consider synthetic images. Fig. 2 illustrates our synthetic image experiments. We describe every image and their corresponding values in the tree with different colors. The structure of projected trees considering the blue and black time stamps are different from their single frame trees. There is only one C_1 for $\mathcal{T}^3$ and it causes a structural difference between $\mathcal{T}_3$ and $\mathcal{T}^3$. Leaves of $\mathcal{T}_3$ and $\mathcal{T}^3$ can be listed as;
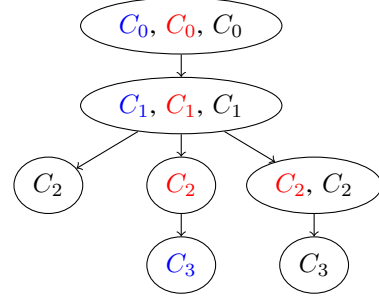
$$leaves(\mathcal{T}_3) = C'_0, C^1_1, C'^2_1, C^1_2, C'^2_2, C_3 \tag{23}$$
$$leaves(\mathcal{T}^3) = C'_0, C'_1, C^1_2, C'^2_2, C_3 \tag{24}$$

We also show $\mathcal{T}_\varsigma$ and $\mathcal{T}^\varsigma$ trees. In order to obtain $\mathcal{T}_\varsigma$, we have considered the mean of each pixel time series. Projected trees
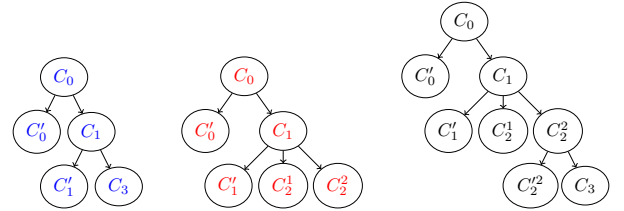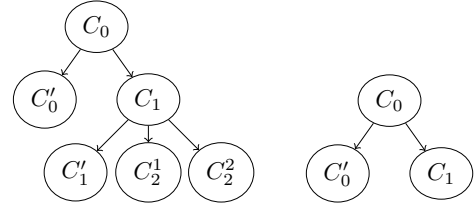

(a) 3 Synthetic images

(b) Space-time tree ($\mathcal{T}$)
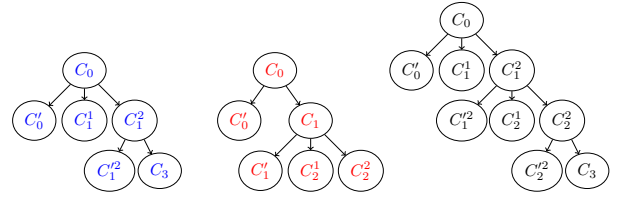
(c) Tree for each image ($\mathcal{T}_1, \mathcal{T}_2, \mathcal{T}_3$)

(d) Temporal projection of space time tree ($\mathcal{T}^1, \mathcal{T}^2, \mathcal{T}^3$)

(e) SH tree and Spatial projection of Space-time tree ($\mathcal{T}_\varsigma, \mathcal{T}^\varsigma$)

Fig. 2: Illustration of trees built using the different strategies and the projected trees.

for both temporal and spatial domains have a lower or equal amount of nodes. Less leaves pair provide less complexity for the corresponding tree. Since $\mathcal{T}_2$ and $\mathcal{T}^2$ are identical, their properties are the same. While the cost of $\mathcal{T}_1$ and $\mathcal{T}^1$ is 108 and 80 respectively, the costs of $\mathcal{T}_3$ and $\mathcal{T}^3$ is 142 and 122 respectively. Namely, the projected trees have a lower or equal cost and a lower or equal amount of nodes for our synthetic image experiment.
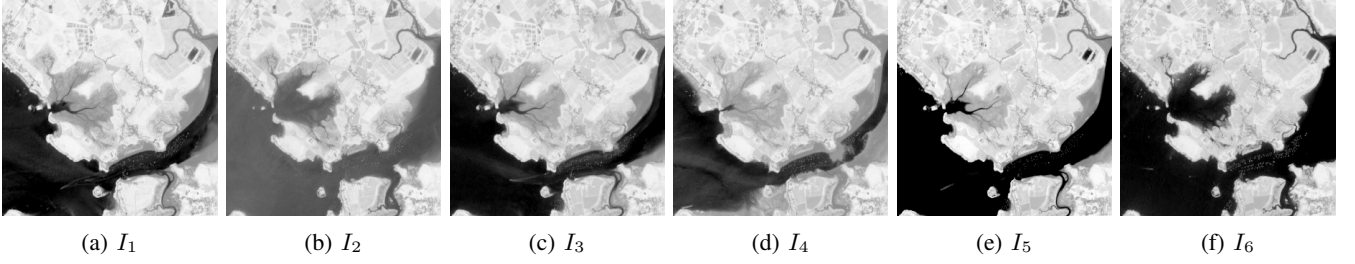
(a) I_1 (b) I_2 (c) I_3 (d) I_4 (e) I_5 (f) I_6

Fig. 3: Sentinel-2 Satellite image time series dataset.

B. Real dataset

In addition to the experiments conducted with a synthetic dataset, we have also considered a real use case with the analysis of Satellite Image Time Series (SITS) on Morbihan, France. The images were acquired with Sentinel-2 Copernicus mission, and the multispectral images were converted as gray scales ones by computing the normalized difference vegetation index (NDVI). We used a sample SITS made of small extracts (300×300px) from 6 observations which were performed in 2018 with a spatial resolution of 10m, and as illustrated in Fig. 3. Sentinel-2 imagery is mainly used for land cover mapping because of its stable viewing angle and short revisit time [31]. Alternatively, we used 50cm panchromatic Pleiades images extracted from Kalideos[1] (300 pixels $\times$ 300 pixels) located at Pontorson, also in Brittany, France. The Pleiades time series is also composed of six images acquired between May 2018 and April 2019. Pleiades offers a very high spatial resolution and it is used to observe small details of Earth. We use the Pleiades dataset for qualitative assessment of the filtering experiments because of their visual superiority, and rely on the Sentinel-2 for LAF and cost experiments.

VI. EXPERIMENTS

One of the challenges faced when analyzing SITS is the intensity change over time. Different structures of hierarchical representations are caused by shape and intensity-based changes. We discuss here these changes according to the three criteria was explained in section IV.

A. Number of node analysis

First, we measured the number of nodes for each $\mathcal{T}_t$ and their corresponding temporal projection of $\mathcal{T}_\tau, i.e. \mathcal{T}^t$. We reported the values in Tab. I. We have considered both 26 and continuous connectivity, and denote the corresponding trees by $\mathcal{T}^t$ and $\mathcal{T}'^t$, respectively. We report results for both max and min-tree separately. The last three rows provide the standard deviation, the average and the total of number of nodes. The numbers of nodes for TH strategy are regular in comparison to the projected trees. Indeed, the number of nodes is variable for both projected trees. When the number of nodes increases for max-tree, it is generally decreasing with min-tree for both projected trees.

[1]https://bretagne.kalideos.fr

TABLE I: Amount of nodes with the SITS dataset.

	$\mathcal{T}_t$		$\mathcal{T}^t$		$\mathcal{T}'^t$	
	Max	Min	Max	Min	Max	Min
$t = 1$	13640	14274	6377	5874	4917	3481
$t = 2$	13577	14231	4471	4523	3688	4007
$t = 3$	14268	14002	2418	5726	2099	2924
$t = 4$	13883	14178	5111	3067	3469	2626
$t = 5$	12495	11592	6726	2966	6178	2862
$t = 6$	15176	13943	4106	6789	1614	5838
std.	804	951	1445	1438	1558	1090
avg.	13839	13703	4818	4824	3631	3623
total	83039	82220	28909	28945	21789	21738

All projected trees have fewer nodes than the corresponding single frame tree built from the same time stamp. Namely, the space-time tree provides a less complex structure. Besides, $\mathcal{T}'^t$ has less nodes than $\mathcal{T}^t$ consistently.
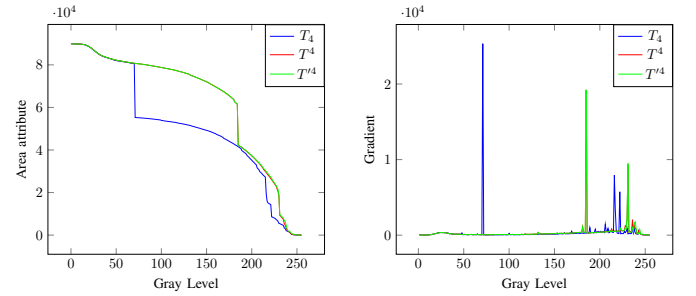
(a) Node Signature   (b) Gradient Curve

Fig. 4: Area attribute signatures of the same location with TH strategy, projected trees and their gradient curve.

Fig. 4 illustrates the average area attribute signature for leaves of $\mathcal{T}_4$, $\mathcal{T}^4$ and $\mathcal{T}'^4$. The maximum area is obtained for the root node and corresponds to image size $300 \times 300 = 90,000$. The gradient curve demonstrates abrupt changes in the node attribute. This curve generally overlaps for $\mathcal{T}^4$ and $\mathcal{T}'^4$. The abrupt change for $\mathcal{T}_4$ appears at a low level which may cause information loss if a filtering or any other process is applied at that level.

B. Cost Analysis

In order to evaluate the usability of the cost function for morphological hierarchies, we first evaluated it with a single frame image and its noise added versions. We added Gaussian noise to an image three times to obtain 3 noisy images. Then,

TABLE II: Cost values for T_4 and trees with noisy images.

	Cost
I_4	**1.00**
I_4 + noise	11.81
I_4 + (2 * noise)	15.19
I_4 + (3 * noise)	18.65

TABLE III: Cost values for Temporal Hierarchy and Temporal Projection.

	$\mathcal{T}_t$		$\mathcal{T}^t$		$\mathcal{T}'^t$	
	Max	Min	Max	Min	Max	Min
$t = 1$	57.2	93.5	23.8	10.4	6.4	5.8
$t = 2$	41.7	82.2	4.4	12.8	3.2	9.9
$t = 3$	54.7	86.5	2.1	10.9	1.00	4.4
$t = 4$	58.6	86.0	12.8	4.5	5.2	2.42
$t = 5$	66.9	70.2	25.6	1.2	23.1	1.00
$t = 6$	70.6	98.4	7.2	20.6	1.3	14.1

TABLE IV: Cost values for Spatial Hierarchy and Spatial Projection.

	Max	Min
$\mathcal{T}_\varsigma$	500.00	1000.00
$\mathcal{T}^\varsigma$	1.5.00	3.00
$\mathcal{T}'^\varsigma$	**1.00**	**1.00**

we observed that the cost was increasing w.r.t the noise level. We have used I_4 image for this experiment and report the cost values in Tab. II.

Table III reports the cost values for the temporal projection and temporal hierarchy. We normalized the cost values according to the lowest values which are $\mathcal{T}'^3$ for max-tree and $\mathcal{T}'^5$ for min-tree.

Since Morbihan is affected by tide, the amount and value of pixels on the sea is changing. While I_1, I_3, I_5 and I_6 have similar colors on the sea, I_2, I_4 are different from them because of the bathymetry. The cost value of T'^3 min-tree is much lower than for T^3, since the continuous connectivity is providing connectivity with I_1 and I_5. Similarly, the projected max-tree T'^4 has a much lower cost than T^4 because it connects I_2 to I_4 and sand pixels are becoming more compact in T'^4.

Table IV shows the cost values for $\mathcal{T}_\varsigma$, $\mathcal{T}^\varsigma$ and $\mathcal{T}'^\varsigma$ as in Table III. We used the distance based ordering with Euclidean distance. According to the Dasgupta's cost, the space-time tree is better than both TH and SH strategies.

C. Filtering

Fig. 5 illustrates our filtering experiments with one of the Pleiades images. We have used all 6 images but we illustrate only one of them which we select for filtering. We have filtered trees with area attribute and the same threshold ($h = 20$) for the sake of comparison. The second row of the figure represents filtered images and its third row their difference with the original image respectively. Filtering removes the small objects such as cars or noises of the images. The amount of changed pixels for T_t, T^t and T'^t is 13,891, 3682 and 2243 respectively. More precisely, filtering with the TH strategy changes many pixels comparing to the projected tree although
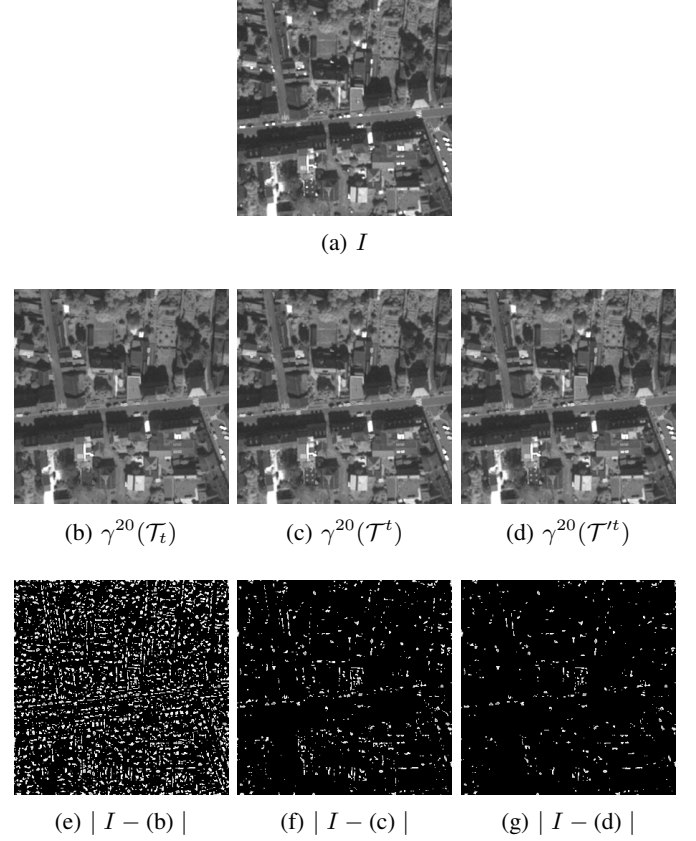

(a) I

(b) $\gamma^{20}(\mathcal{T}_t)$ (c) $\gamma^{20}(\mathcal{T}^t)$ (d) $\gamma^{20}(\mathcal{T}'^t)$

(e) $| I - (b) |$ (f) $| I - (c) |$ (g) $| I - (d) |$

Fig. 5: Filtering results of one image of the Pleiades time series with $\mathcal{T}_t$ and projected trees, along with their residues that highlight changes.

a small threshold is used. Much changes are not desired with such a small threshold. Changes are less for $\mathcal{T}'^t$ comparing to $\mathcal{T}^t$. These results show that the compactness of a tree leads to an efficient filtering process.

VII. CONCLUSION

In a previous work [6], we have proposed different hierarchical representation strategies for image sequences. We have pursued this study in this paper, where we have compared morphological hierarchies to determine the better tree representation for image sequence analysis. Our goal is to select the most relevant and efficient hierarchical representation for subsequent studies.

Comparison of trees requires them to lie in the same domain. To do so, we have proposed projection methods for space-time trees in order to make them comparable with the trees obtained with spatial and temporal hierarchy. We also proposed a continuous connectivity rule to build a space-time tree with lower complexity, lower cost according to [13] and providing interesting filtering capabilities. In our experiments, we have used some gray-scale satellite images and shown that the temporal information provides a less complex hierarchical structure when projected in space or in time.

Among future works, we aim to explore capability of trees with continuous connectivity for real remote sensing based applications such as land-cover mapping, pattern recognition and change detection.

REFERENCES

[1] P. Salembier, A. Oliveras, and L. Garrido, "Antiextensive connected operators for image and sequence processing," *IEEE Transactions on Image Processing*, vol. 7, no. 4, pp. 555–570, 1998.

[2] P. Salembier and M. H. Wilkinson, "Connected operators," *IEEE Signal Processing Magazine*, vol. 26, no. 6, 2009.

[3] B. Perret, J. Cousty, O. Tankyevych, H. Talbot, and N. Passat, "Directed connected operators: Asymmetric hierarchies for image filtering and segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 6, pp. 1162–1176, 2014.

[4] P. Salembier, S. Liesegang, and C. López-Martínez, "Ship detection in sar images based on maxtree representation and graph signal processing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 5, pp. 2709–2724, 2018.

[5] G. Tochon, M. Dalla Mura, M. A. Veganzones, T. Géraud, and J. Chanussot, "Braids of partitions for the hierarchical representation and segmentation of multimodal images," *Pattern Recognition*, vol. 95, pp. 162–172, 2019.

[6] C. Tuna, B. Mirmahboub, F. Merciol, and S. Lefèvre, "Component trees for image sequences and streams," *Pattern Recognition Letters*, 2019.

[7] B. Perret, J. Cousty, S. J. F. Guimarães, and D. S. Maia, "Evaluation of hierarchical watersheds," *IEEE Transactions on Image Processing*, vol. 27, no. 4, pp. 1676–1688, April 2018.

[8] J. F. Randrianasoa, C. Kurtz, P. Gançarski, E. Desjardin, and N. Passat, "Evaluating the quality of binary partition trees based on uncertain semantic ground-truth for image segmentation," in *2017 IEEE International Conference on Image Processing (ICIP)*, 2017, pp. 3874–3878.

[9] A. C. Jalba and M. A. Westenberg, "A comparison of two tree representations for data-driven volumetric image filtering," in *International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing*. Springer, 2011, pp. 405–416.

[10] F. Bracci, U. Hillenbrand, Z.-C. Marton, and M. H. Wilkinson, "On the use of the tree structure of depth levels for comparing 3d object views," in *International Conference on Computer Analysis of Images and Patterns*. Springer, 2017, pp. 251–263.

[11] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846–850, 1971.

[12] M. Bateni, S. Behnezhad, M. Derakhshan, M. Hajiaghayi, R. Kiveris, S. Lattanzi, and V. Mirrokni, "Affinity clustering: Hierarchical clustering at scale," in *Advances in Neural Information Processing Systems*, 2017, pp. 6864–6874.

[13] S. Dasgupta, "A cost function for similarity-based hierarchical clustering," in *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, 2016, pp. 118–127.

[14] M. Charikar, V. Chatziafratis, and R. Niazadeh, "Hierarchical clustering better than average-linkage," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, 2019, pp. 2291–2304.

[15] V. Cohen-Addad, V. Kanade, and F. Mallmann-Trenn, "Hierarchical clustering beyond the worst-case," in *Advances in Neural Information Processing Systems*, 2017, pp. 6201–6209.

[16] V. Cohen-Addad, V. Kanade, F. Mallmann-Trenn, and C. Mathieu, "Hierarchical clustering: Objective functions and algorithms," *Journal of the ACM (JACM)*, vol. 66, no. 4, pp. 1–42, 2019.

[17] P. Bosilj, E. Kijak, and S. Lefèvre, "Partition and inclusion hierarchies of images: A comprehensive survey," *Journal of Imaging*, vol. 4, no. 2, p. 33, 2018.

[18] R. Jones, "Component trees for image filtering and segmentation," in *IEEE Workshop on Nonlinear Signal and Image Processing*, 1997.

[19] P. Monasse and F. Guichard, "Scale-space from a level lines tree," *Journal of Visual Communication and Image Representation*, vol. 11, no. 2, pp. 224–236, 2000.

[20] P. Soille, "Constrained connectivity for hierarchical image partitioning and simplification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 7, pp. 1132–1145, 2008.

[21] P. Salembier and L. Garrido, "Binary partition tree as an efficient representation for image processing, segmentation, and information retrieval," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 561–576, 2000.

[22] E. Aptoula and S. Lefèvre, "A comparative study on multivariate mathematical morphology," *Pattern Recognition*, vol. 40, no. 11, pp. 2914–2929, 2007.

[23] M. Dalla Mura, J. Atli Benediktsson, B. Waske, and L. Bruzzone, "Extended profiles with morphological attribute filters for the analysis of hyperspectral data," *International Journal of Remote Sensing*, vol. 31, no. 22, pp. 5975–5991, 2010.

[24] Y. Chi, R. R. Muntz, S. Nijssen, and J. N. Kok, "Frequent subtree mining–an overview," *Fundamenta Informaticae*, vol. 66, no. 1-2, pp. 161–198, 2005.

[25] C. Ronse, "Ordering partial partitions for image segmentation and filtering: Merging, creating and inflating blocks," *Journal of Mathematical Imaging and Vision*, vol. 49, no. 1, pp. 202–233, 2014.

[26] K. Bhardwaj, S. Patra, and L. Bruzzone, "Threshold-free attribute profile for classification of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 10, pp. 7731–7742, 2019.

[27] D. Ghoshdastidar, M. Perrot, and U. von Luxburg, "Foundations of comparison-based hierarchical clustering," in *Advances in Neural Information Processing Systems*, 2019, pp. 7454–7464.

[28] L. Najman, F. Meyer *et al.*, "A short tour of mathematical morphology on edge and vertex weighted graphs," *Image Processing and Analysis with Graphs: Theory and Practice*, pp. 141–174, 2012.

[29] L. Najman and J. Cousty, "A graph-based mathematical morphology reader," *Pattern Recognition Letters*, vol. 47, pp. 3–17, 2014.

[30] M. Dalla Mura, J. A. Benediktsson, B. Waske, and L. Bruzzone, "Morphological attribute profiles for the analysis of very high resolution images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 48, no. 10, pp. 3747–3762, 2010.

[31] M. Drusch, U. Del Bello, S. Carlier, O. Colin, V. Fernandez, F. Gascon, B. Hoersch, C. Isola, P. Laberinti, P. Martimort *et al.*, "Sentinel-2: Esa's optical high-resolution mission for gmes operational services," *Remote sensing of Environment*, vol. 120, pp. 25–36, 2012.