Combinatorial space of watershed hierarchies for image characterization

Amin Fehri, Santiago Velasco-Forero, Fernand Meyer

Version of Record: https://www.sciencedirect.com/science/article/pii/S0167865519303162
Manuscript_c5f5ca6575e4f8c8bb967a284bf506d3

Pattern Recognition Letters
journal homepage: www.elsevier.com

Combinatorial space of watershed hierarchies for image characterization

Amin Fehri[a,**], Santiago Velasco-Forero[a], Fernand Meyer[a]

[a] *Center of Mathematical Morphology, Mines ParisTech, PSL Research University*

Article history:
Received 15 June 2019
Received in final form 14 August 2019

Communicated by A. Fehri

Keywords: Mathematical morphology, Hierarchies, Gromov-Hausdorff distance, Combination of hierarchies

ABSTRACT

We propose a framework for image characterization using hierarchies of segmentations. For this purpose, we structure the space of hierarchies using the Gromov-Hausdorff distance. We propose different ways of combining hierarchies and study their properties thanks to the GH distance. We then expose how to leverage the combinatorial space of hierarchies to derive efficient image representations. This framework opens a path for a controlled exploration and use of the combinatorial space of hierarchies.

1. Introduction

Hierarchical clusterings are mathematical structures that have proven to be useful multiscale representations for a variety of tasks. In this paper, we work with hierarchies generated upon images, namely hierarchies of segmentations. Hierarchies of segmentations have been used extensively throughout the literature. One can cite (Arbeláez et al., 2014; Cousty et al., 2018; Xu et al., 2016; Wei et al., 2018). Although we focus on images representations in this work, the approaches presented can be extended to any object that can be modeled as a graph. Hierarchical segmentations are interesting to characterize images as they are able to extract the relevant information that is distributed across scales. However, there is no single hierarchy that can capture all the desired features from an image, as these features are dependent upon the application and the type of image. This is why, in this paper, we propose a multi-model approach by considering several hierarchies for each image. An interesting analogy is to see each hierarchy as a filter of *pure* color. To obtain the best contrast between objects of an image, one has to choose the best color filter. For example, in an ophthalmics image of the back of the eye, the best way to separate the retina from the red vessels is to apply a green filter: this isolates vessels and makes their subtraction from the image easier. In a similar way, morphological hierarchies can be seen as *pure* geometrical filters that can be combined to obtain *derived* hierarchies that can discriminate complex structures. We will expose two ways to do so: one *sequential*, corresponding to chainings of hierarchies, and one *parallel*, corresponding to functional combinations of hierarchies. In a way, we multiply the viewpoints on an image in a controlled and understandable manner. Furthermore, we propose to structure the resulting combinatorial space of hierarchies by using the Gromov-Hausdorff (GH) distance to structure it as a metric space, as well as a visualization tools for a better interpretability of results. Finally, we show how one can build and use interhierarchy distance matrices as efficient condensed features of images for various applications. This part of the paper uses elements from previous work (Fehri et al., 2018).

The remaining of the paper is organized as follows. Section 2 recalls the theoretical framework to obtain morphological hierarchies within a graph-based framework, and gives some references. In Section 3, we show how one can structure the combinatorial space of hierarchies as a metric space in order to study its properties. In Sections 4 and 5, we introduce the sequential and parallel combinations of hierarchies, and make use of the tools introduced before to study their properties. Notably, we show how hierarchies having as support the same tree, and differing only by the weights on the edges of this tree, can generate a new hierarchy holding the same tree as support and new edge weights computed from the initial edge weights. Finally, Section 6 shows the interest of interhierarchy distance matrices as features for image characterization.

** Corresponding author
e-mail: amin.fehri@mines-paristech.fr (Amin Fehri)

2. Morphological hierarchies

2.1. Theoretical framework

In this section, we explain to the reader how to construct and use graph-based hierarchical segmentation, as well as remind some properties of the objects we work with. For each image, let us suppose that a fine partition is produced by an initial segmentation (for instance a set of superpixels (Achanta et al., 2012; Machairas et al., 2015) or the basins produced by a classical watershed algorithm (Meyer and Beucher, 1990)) and contains all contours making sense in the image. We define a dissimilarity measure between adjacent tiles of this fine partition. One can then see the image as a graph, the *region adjacency graph* (RAG), in which each node represents a tile of the partition; an edge links two nodes if the corresponding regions are neighbors in the image; the weight of the edge is equal to the dissimilarity between these regions. Formally, we denote this graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, where $\mathcal{V}$ corresponds to the image domain or set of pixels/fine regions, $\mathcal{E} \subset \mathcal{V} \times \mathcal{V}$ is the set of edges linking neighbour regions, $\mathbf{W} : \mathcal{E} \to \mathbb{R}^+$ is the dissimilarity measure usually based on local gradient information (or color or texture), for instance $\mathbf{W}(i, j) \propto |\mathbf{I}(v_i) - \mathbf{I}(v_j)|$ with $\mathbf{I} : \mathcal{V} \to \mathbb{R}$ representing the image intensity. The edge linking the nodes p and q is designated by e_{pq} . A path is a sequence of nodes and edges: an example of a path linking the nodes p and s is the set $\{p, e_{pt}, t, e_{ts}, s\}$. A *connected graph* is a graph where each pair of nodes is connected by a path. A *cycle* is a path whose extremities coincide. A *tree* is a connected graph without cycle. A *spanning tree* is a tree containing all nodes. A *minimum spanning tree* (MST) $\mathcal{MST}(\mathcal{G})$ of a graph $\mathcal{G}$ is a spanning tree with minimal possible weight (the weight of a tree being equal to the sum of the weights of its edges), obtained for example using the Prim's algorithm (Prim, 1957). A *forest* is a collection of trees. A *partition* P of a set $\mathcal{V}$ is a collection of subsets of $\mathcal{V}$, such that the whole set $\mathcal{V}$ is the disjoint union of the subsets in the partition, i.e., $\mathsf{P} = \{\mathsf{R}_1, \mathsf{R}_2, \ldots, \mathsf{R}_k\}$, such that $\forall i, \mathsf{R}_i \subseteq \mathcal{V}$; $\forall i \neq j, \mathsf{R}_i \cap \mathsf{R}_j = \emptyset$; $\bigcup_i^k \mathsf{R}_i = \mathcal{V}$. Cutting all edges of the $\mathcal{MST}(\mathcal{G})$ having a valuation superior to a threshold λ leads to a minimum spanning forest (MSF) $\mathcal{F}(\mathcal{G}, \lambda)$, i.e. to a partition of the graph. Note that the obtained partition is the same that one would have obtained by cutting edges superior to λ directly on $\mathcal{G}$ (Najman et al., 2013). Since working on the $\mathcal{MST}(\mathcal{G})$ is less costly and provides similar results regarding graph-based segmentation, we work only with the $\mathcal{MST}(\mathcal{G})$ in the sequel. So cutting edges by decreasing valuations gives an *indexed hierarchy of partitions* $(\mathcal{H}, \boldsymbol{\lambda})$, with $\mathcal{H}$ a *hierarchy of partitions* i.e. a chain of nested partitions $\mathcal{H} = \{\mathsf{P}_0, \mathsf{P}_1, \ldots, \mathsf{P}_n | \forall j, k, \quad 0 \quad \leq j \leq k \leq n \Rightarrow \mathsf{P}_j \sqsubseteq \mathsf{P}_k\}$, with P_n the single-region partition and P_0 the finest partition on the image, and $\boldsymbol{\lambda} : \mathcal{H} \to \mathbb{R}^+$ being a stratification index that corresponds to the *ultrametric distance* defining the hierarchy and verifying $\boldsymbol{\lambda}(\mathsf{P}) < \boldsymbol{\lambda}(\mathsf{P}')$ for two nested partitions $\mathsf{P} \subset \mathsf{P}'$. This process is otherwise called *single-linkage hierarchical clustering* in the literature (Kaufman and Rousseeuw, 2009).

This increasing map allows us to value each contour according to the level of the hierarchy for which it disappears: this is the *saliency* of the contour (corresponding to the ultrametric distance between the two regions it separates), and we consider that the higher the saliency, the stronger the contour. For a given hierarchy, the image in which each contour takes as value its saliency is called *Ultrametric Contour Map* (UCM)(Arbelaez et al., 2011) or *saliency map* (Cousty et al., 2018).

We refer to a hierarchy built on a graph with edge weights expressing local contrast as to a *trivial* hierarchy. Whatever the intended use of hierarchical representations, for example the extraction of a segmentation out of a hierarchy (Guigues et al., 2006; Kiran and Serra, 2014), the trivial hierarchy is usually not the more adapted one to work with in order to obtain the best results. It is thus interesting to look for more informative dissimilarities adapted to the content of images, so that the simplest methods are sufficient to obtain the desired results, for example computing interesting partitions. As these hierarchies are defined as ultrametric distances on a set of nodes, we can either aim at learning these ultrametrics (Wolf et al., 2017) or at designing them in order for them to capture certain types of information. Several morphological hierarchical techniques exist to do the latter.

2.2. A variety of morphological hierarchies

Morphological hierarchies are representations capturing information across scales with an emphasis put on shape and size features. We hereby remind the reader of some of them known as watershed hierarchies, while insisting on the fact that approaches and methods proposed throughout the rest of this paper can be used with any type of hierarchy.

Seminal works on morphological hierarchies include the dynamics hierarchy exhibiting contrasted regions (Grimaud, 1992) (and corresponding to the trivial hierarchy), or the area-based and volume-based watershed hierarchies (Vachier and Meyer, 1995) extending the dynamics hierarchy by taking into account sizes of regions as well. The waterfall hierarchy, first described in the context of a topographic surface flooding (Beucher, 1990), has then been extended on graphs (Meyer, 2015b). The waterfall hierarchy highlights the nested structure of the catchment basins of a topographic surface. By flooding each catchment basin of a topographic surface up to the level of its lowest pass point, a new simpler topographic surface is produced, whose catchment basins result from the merging of catchment basins of the initial surface. The stochastic watershed (SWS), introduced in (Angulo and Jeulin, 2007) on a simulation basis and extended with a graph-based approach in (Meyer, 2015a), is a versatile tool to construct hierarchies. The seminal idea is to operate multiple times marker-based segmentation with random markers and valuate each edge of the image by its frequency of appearance in the resulting segmentations. The same results can be obtained efficiently on graphs without the need for simulation within the framework described in section 2.1. The output of the SWS algorithm is a hierarchy highlighting specific types of regions at different scales. It is very versatile as the type of markers spread, as long as the probabilistic law governing markers distribution, can be adapted for various tasks (Fehri et al., 2017).

3. Structuring the combinatorial space of hierarchies

In the previous section, we reminded the reader of a versatile framework to characterize images through numerous different hierarchies. Furthermore, as we shall see in the next sections, hierarchies can be combined in various ways, and we then face an explosion of their number. Hence, we need a way to structure the combinatorial space of hierarchies.

3.1. The Gromov-Hausdorff distance between hierarchies

3.1.1. Definition

In order to operate this structuring, one needs to define a distance between these ultrametric spaces. The Gromov-Hausdorff distance (GH distance) (Gromov, 2007) defined between metric spaces constitutes such a distance. By reducing this distance to the subclass of ultrametric spaces, we can in particular quantify the relative contributions of different hierarchical clusterings. This distance, used intensively in several fields such as phylogenetics and data mining (Felsenstein, 2014), has also notably been used in image processing as a way to estimate the similarity between two points clouds (Mémoli, 2004). Let us consider two ultrametric spaces $(X_1, \boldsymbol{\lambda}_\alpha)$ and $(X_2, \boldsymbol{\lambda}_\beta)$. One supposes that we have defined two functions $f : X_1 \to X_2$ and $g : X_2 \to X_1$ that are maps from one space to the other. The GH-distance is expressed as:

$$\mathrm{d_{GH}}(X_1, X_2) := \frac{1}{2} \min_{f,g} \max(dis(f), dis(g), dis(f, g)) \qquad (1)$$

With the distorsion $dis(f)$ and the joint distorsion $dis(f, g)$ defined as:

$$\begin{cases} dis(f) := \max_{(x,x')\in X_1^2} |\boldsymbol{\lambda}_\alpha(x, x') - \boldsymbol{\lambda}_\beta(f(x), f(x'))| \\ dis(f, g) := \max_{x\in X_1, x'\in X_2} |\boldsymbol{\lambda}_\alpha(x, g(x')) - \boldsymbol{\lambda}_\beta(x', f(x))| \end{cases} \qquad (2)$$

Intuitively, it measures how close can we get to an isometric (distance-preserving) embedding between two metric spaces.

To determine (1) for two hierarchies defined over different sets, one must match data points before any distance computation, which is a computationally heavy operation that leads some authors to provide heuristics to approximate it in specific configurations (Agarwal et al., 2015). In our case we are dealing with structures that are much simpler: instead of considering objects-to-objects distances, we compute them for hierarchies based upon the same fine partition of an image. So the considered metric spaces differ only by their metrics and not by the space they cover, which means that the two distortions are symmetrical and equal to the joint-distortion as well. Thus, the GH distance (1) simply becomes:

$$\mathrm{d_{GH}}((X, \boldsymbol{\lambda}_\alpha), (X, \boldsymbol{\lambda}_\beta)) = \max_{x,x'\in X} |\boldsymbol{\lambda}_\alpha(x, x') - \boldsymbol{\lambda}_\beta(x, x')|. \qquad (3)$$

Provided with such a distance, we can quantify the relative contributions of different hierarchies built upon the same image. It is interesting to note that a new level of image representation naturally emerges from this approach, as illustrated in Fig. 1. Furthermore, this provides us with a condensed representation leveraging the information provided by all the different levels of these different hierarchies. Let us indeed consider an image $\mathbf{I}$ and a set of complementary hierarchies $((\mathcal{H}_1, \boldsymbol{\lambda}_1), \ldots, (\mathcal{H}_N, \boldsymbol{\lambda}_N))$ built upon this image. It is then straightforward to compute the GH distance between these hierarchies, as they constitute ultrametric spaces upon the same set. We take advantage of it by building the following symmetrical distance matrix:

$$M(\mathbf{I}, (\mathcal{H}_1, \ldots, \mathcal{H}_N)) = \left[\mathrm{d_{GH}}(\boldsymbol{\lambda}_i, \boldsymbol{\lambda}_j)\right]_{(i,j)\in\{1,\ldots,N\}^2} \qquad (4)$$

Such a representation opens new ways to suppress redundancies and create a restrained descriptive family of hierarchies. It also allows to study the properties of hierarchical combinations or visualize the effects of several different hierarchies.

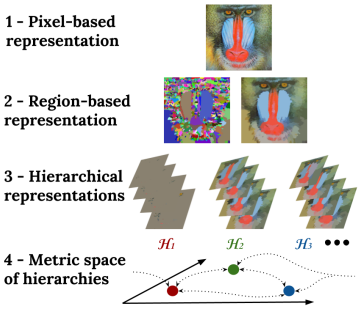


Fig. 1: The different image representations considered in this work: 1) The finer level is the level of the image itself. 2) Computing a fine partition of the image already constitutes a rougher model of it, that we can see as a region-based representation. 3) A structuring of the regions of the fine partition as a hierarchical segmentation. Each hierarchy is fully characterized by the set of points that it regroups as long as by the ultrametric it defines between these points. 4) One can structure the space of hierarchies itself into a metric space. Doing so opens a path to study the properties of different hierarchical clustering methods, and to evaluate to which extent they differ and are complementary.

3.1.2. Ultrametric Normalization

To make sense of GH distances, one must be sure that ultrametric values are commensurable and of the same order of magnitude, as hierarchical clusterings may be of very different natures. Indeed, their ultrametric values can represent different things, as for example the surface-based (resp. volume-based) extinction hierarchy ultrametric values are surfaces (resp. volumes), and the waterfall hierarchy ultrametric values represent stacking orders. Furthermore, hierarchies can have different scales, as typically SWS hierarchies have their probabilistic values in the range $[0, 1]$, whereas other types of hierarchies do not necessarily have ultrametric values in a fixed range. This is why, in order to properly compare and combine hierarchies, we normalize their ultrametric values. This has the effect of aligning them. Accordingly, we propose a way to normalize these values with respect to the number of regions in each level of the hierarchy. Let $(\mathcal{H}, \boldsymbol{\lambda})$ be an ultrametric hierarchy, with $\boldsymbol{\lambda} : \mathcal{H} \mapsto \mathbb{R}_+$ which is strictly increasing with the inclusion order over $\mathcal{H}$. Let us denote $N = card(H)$, $(\mathsf{P}_0, ..., \mathsf{P}_N)$ the nested series of partitions associated with the hierarchy, and $(n_0, ..., n_N)$ the numbers of regions in these partitions, with $0 < n_0 \leq n_1 \leq ... \leq n_N = N$. Then we take as a *normalized ultrametric*, $\tilde{\boldsymbol{\lambda}} : \begin{cases} \mathcal{H} \to [0, 1] \\ \mathsf{P}_i \mapsto \frac{N-n_i}{N} \end{cases}$.

3.2. Using dimensionality reductions techniques to visualize the relative descriptive power of each hierarchy

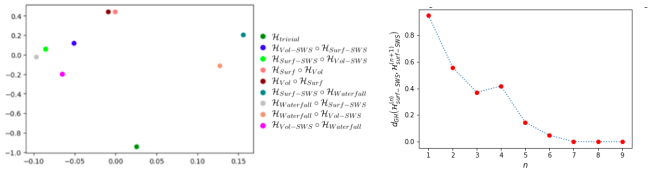
We would like to visualize the relative contributions of different hierarchical clustering methods constructed upon the same points, i.e. the same regions of a fine partition. We have seen in section 3.1.1 that in such a case, computing GH distances between hierarchies is straightforward. This appears interesting as computing the distance between two hierarchies can give us an idea of their relative contributions, provided that we normalize their values first, as described in section 3.1.2.

Let us consider a set of indexed hierarchies $\mathcal{H} = \{(\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2), \ldots, (\mathcal{H}_{|\mathcal{H}|}, \lambda_{|\mathcal{H}|})\}$. In this paper, we use two dimensionality reduction techniques: the Multidimensional Scaling (MDS) (Borg and Groenen, 2003) and the t-distributed stochastic neighbor embedding (t-SNE) (Maaten and Hinton, 2008). Whether it is to apply MDS or t-SNE, we start from the symmetrical matrix D introduced in (4), filled with GH distances between pairs of hierarchies, and with size $|\mathcal{H}|\times|\mathcal{H}|$. MDS and t-SNE can then be used for different purposes. If we have computed the interhierarchy matrices for a given image, using MDS allows us to visualize the distances between hierarchies for this image. This way, we can study their interrelations for this specific image, determine their complementarity or redundancy, and more generally define patterns between them. Since we are not using a Euclidean distance but a GH one, we make use of the usual metric MDS to project distances between hierarchies from a space of dimension $|\mathcal{H}|\times|\mathcal{H}|$ into a low-dimensional space. Given a reference hierarchy (for example the trivial hierarchy), this gives us an idea of how the hierarchies distance themselves from one another by looking at how they place themselves relatively to the reference hierarchy on the resulting figure. This way, we can estimate the respective contributions of each hierarchical construction scheme. In a complementary fashion, one may compute for several images the same hierarchies and corresponding distance matrices, and characterize images this way, which will be the topic of section 6. We then make use of t-SNE to project such images representations from a space of dimension $|\mathcal{H}|\times|\mathcal{H}|$ to a space of dimension two or three. This way, we can estimate whether the departure hierarchies set was discriminative enough to separate different types of images. In the next section, we present different ways to combine hierarchies and make use of the techniques we introduced to study some of their properties.

4. Sequential combinations of hierarchies through chaining

4.1. Definition

In section 2 we reminded how to obtain morphological hierarchies within a graph-based framework, using the following procedure: (i) Get a fine partition $\mathcal{FS}(\mathbf{I})$ of the image $\mathbf{I}$, (ii) Construct a region adjacency graph $\mathcal{G}$, (iii) Compute a minimum spanning tree $\mathcal{MST}$ of $\mathcal{G}$, (iv) Using this hierarchical structure, compute new valuations for the MST's edges, which leads to a new hierarchy. We now draw the reader's attention to the fact that departing from a tree which valuations are those of the initial graph, one obtains a tree with identical structure but different edge valuations. To go further, this new tree can



(a) (b)

Fig. 2: (a) Visualization using MDS of the distances between different sequential combinations of hierarchies. We note that these hierarchies do not commute. (b) Convergence of chainings of surface-based SWS hierarchies.

then be used as a departure point for a similar construction but based upon different criteria and parameters. More specifically, at the end of step 3, we have obtained a MST with valuations $\mathbf{W}_0$. After step 4, we obtain a new MST with the same structure but different valuations $\mathbf{W}_1$. This process can be iterated as much as desired, so that it takes in entry a MST with valuations $\mathbf{W}_k$ and outputs a MST with the same structure and with valuations $\mathbf{W}_{k+1}$. We call this process *chaining* or *composition* of hierarchies, and denote $\mathcal{H}_2 \circ \mathcal{H}_1$ a chaining of a hierarchy $\mathcal{H}_1$ followed by a hierarchy $\mathcal{H}_2$, and $\mathcal{H}^{(n)} = \underbrace{\mathcal{H} \circ \ldots \circ \mathcal{H}}_{n \text{ times}}$. It can be seen as a form of *sequential* combination, as in such a process, each hierarchy is built starting from the preceding one. Note that one can iterate the same hierarchy several times, but also chain different hierarchies. When chaining hierarchies, we normalize the ultrametric values at each step in the way that was described in section 3.1.2, as it permits to align them.

4.2. Analysis

In this section, we want to study the properties of such chainings, and notably determine whether they: (i) commute, i.e. do we have $\mathcal{H}_1 \circ \mathcal{H}_2 = \mathcal{H}_2 \circ \mathcal{H}_1$?, (ii) converge, i.e. are there hierarchies $\mathcal{H}$, for which $\lim_{n\to+\infty} \mathrm{d_{GH}}(\mathcal{H}^{(n)}, \mathcal{H}^{(n+1)}) = 0$? Note that for (ii) this is actually the definition of a Cauchy series, but since the ultrametric spaces we work with are bounded, it is equivalent to show that a series is Cauchy or convergent in the Gromov-Hausdorff sense. Since we have now at our disposal a distance between hierarchies, we can address these questions. For better visualization, we make use of multidimensional scaling (MDS) to project the metric space of hierarchies equipped with the GH distance into a subspace of dimension two, as explained in section 3.2. We can then visualize the effect of the chaining of several types of hierarchies on the same plot, in order to have an idea of the effect of each one. For illustration purposes, we rely on the example image in Fig. 1, but results presented were consistent on many examples.

4.2.1. Commutation

In order to test if hierarchies commute, we generate some hierarchies for a given image, compute the distances between them and use MDS to visualize the output. It appears that none of the hierarchical schemes that we have tested commute. We illustrate this in Fig. 2(a) for different chainings.

4.2.2. Convergence

We also want to know whether convergence properties are observable when chaining hierarchical clusterings. For example, starting from the image and fine segmentation presented in Fig. 1, we compute the GH-distance between successive chainings of surface-based SWS hierarchies: we observe that this distance is decreasing until reaching zero, as illustrated in Fig. 2(b). This property has been observed for all hierarchies (introduced in section 2.2) when they are being chained: hierarchical chainings seem to converge toward fixed points in the hierarchical space, i.e. to given hierarchical organizations of all the regions of the fine partition. Furthermore, these fixed points differ depending on the type of hierarchy that is being chained. Formal proofs are required to confirm the validity of these results, which is beyond the scope of this paper.

5. Parallel combinations of hierarchies

5.1. Definition

In section 4, we have presented sequential combinations of hierarchies, corresponding to successive hierarchical constructions, for which each hierarchy was built starting from the preceding one. In this section, we study *parallel* combinations of hierarchies that are obtained by functional combinations of the ultrametrics describing different hierarchies.

5.2. General case

When considering two hierarchies $(\mathcal{H}_1, \lambda_1)$ and $(\mathcal{H}_2, \lambda_2)$, their ultrametrics induce a distance between any two points p, q of the graph, respectively $\lambda_1(p,q)$ and $\lambda_2(p,q)$. We can then choose any function $\oplus : \mathbb{R}^2 \to \mathbb{R}$ to obtain a new dissimilarity $\oplus(\lambda_1, \lambda_2)$. We alternatively will also write $\lambda_1 \oplus \lambda_2$ in the following. However, in the general case, this new dissimilarity will no longer be an ultrametric. To obtain an ultrametric from $\underline{\oplus(\lambda_1, \lambda_2)}$, one can thus compute the subdominant ultrametric $\overline{\oplus(\lambda_1, \lambda_2)}$ associated with this dissimilarity, namely the largest ultrametric distance below it. The exploration of the hierarchical space with such processes has notably been studied in (Maia et al., 2017), with infimum, supremum and linear combinations of hierarchies. For example, we can in particular consider the supremum or infimum of hierarchies. We remind the reader that an order relation over the set of hierarchies is: $\mathcal{H}_1 < \mathcal{H}_2$ can be read "$\mathcal{H}_1$ is finer than $\mathcal{H}_2$", and means that $\mathcal{H}_1$ has more regions than $\mathcal{H}_2$ at each level. The infimum of two hierarchies $\mathcal{H}_1$ and $\mathcal{H}_2$ is written $\mathcal{H}_1 \wedge \mathcal{H}_2$ or INF$(\mathcal{H}_1, \mathcal{H}_2)$ and is defined by its ultrametric being the supremum of the ultrametrics of both hierarchies $\lambda = \lambda_1 \vee \lambda_2$. Indeed, if $\oplus = \vee$, we have:

$$\forall (p,q,r), \begin{cases} \lambda_1(p,q) \leq \lambda_1(p,r) \vee \lambda_1(r,q) \\ \lambda_2(p,q) \leq \lambda_2(p,r) \vee \lambda_2(r,q) \end{cases} \tag{5}$$

Then, $\lambda_1 \vee \lambda_2(p,q) = \lambda_1(p,q) \vee \lambda_2(p,q)$

$$\leq [\lambda_1(p,r) \vee \lambda_1(r,q)] \vee [\lambda_2(p,r) \vee \lambda_2(r,q)]$$
$$\leq [\lambda_1(p,r) \vee \lambda_2(p,r)] \vee [\lambda_1(r,q) \vee \lambda_2(r,q)]$$

Thus the commutativity and associativity of $\vee$ operator make the computation of the associated ultrametric easy: we just have to assign to each edge the valuation $\lambda_1 \vee \lambda_2$. However, in most cases for a function $\oplus : \begin{cases} \mathbb{R}^2 \to \mathbb{R} \\ (\lambda_1, \lambda_2) \mapsto \oplus(\lambda_1, \lambda_2) \end{cases}$, we have:

$$\begin{aligned} \lambda_1 \oplus \lambda_2(p,q) &= \lambda_1(p,q) \oplus \lambda_2(p,q) \\ &\leq [\lambda_1(p,r) \vee \lambda_1(r,q)] \oplus [\lambda_2(p,r) \vee \lambda_2(r,q)], \end{aligned}$$

and as the function $\oplus$ is not necessarily distributive with respect to the function $\vee$, we cannot obtain an ultrametric by simply computing $\lambda_1 \oplus \lambda_2$, and must instead compute the subdominant ultrametric $\overline{\lambda_1 \oplus \lambda_2}$. In particular and for example, the supremum of two hierarchies $\mathcal{H}_1$ and $\mathcal{H}_2$ is written $\mathcal{H}_1 \vee \mathcal{H}_2$ or SUP$(\mathcal{H}_1, \mathcal{H}_2)$, and is the smallest hierarchy larger than $\mathcal{H}_1$ and $\mathcal{H}_2$. Its ultrametric is $\lambda = \overline{\lambda_1 \wedge \lambda_2} \neq \lambda_1 \wedge \lambda_2$.

5.3. Simpler parallel combinations between hierarchies built upon the same MST

In our work, we are in a particular situation, as we often build different hierarchies upon the same initial tree. Indeed, starting from a $\mathcal{MST}$ of the RAG $\mathcal{G}$ associated with a fine partition of the image, we generate a new set of valuations on this tree. The resulting ultrametric is the one induced by this MST, and for any pair p, q of nodes of the graphs, its value is equal to the maximal weight of edges on the unique path linking p to q in the MST. In such circumstances, one can easily combine two hierarchies $(\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2)$ for a combination function $\oplus$ verifying a given property, as we shall see hereafter. First, we remind the reader of the path optimality property characterizing any MST.

Theorem 1 (Path optimality (Hu, 1961))**.** *A spanning tree $\mathcal{MST} = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}})$ of a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$ is a minimum spanning tree if and only if it satisfies the following inequality:*

$$\forall \mathsf{e}_{k,l} \notin \mathcal{E}_{\mathcal{MST}}, \forall \mathsf{e}_{i,j} \in \mathfrak{B}, W_{k,l} \geq W_{i,j}, \tag{6}$$

where $\mathfrak{B}$ is the unique path from node k to node l in the $\mathcal{MST}$. Stated otherwise, if an edge e_{pq} does not belong to the $\mathcal{MST}$, then the weight W_{pq} is superior or equal to the weight of all edges belonging on the path connecting p and q in the $\mathcal{MST}$.

We can now expose a theorem ensuring us a fast parallel combination of hierarchies built upon the same MST in most cases.

Theorem 2. *Let us consider two edge-weighted graphs $\mathcal{G}_1 = (\mathcal{V}, \mathcal{E}, \mathbf{W}_1)$ and $\mathcal{G}_2 = (\mathcal{V}, \mathcal{E}, \mathbf{W}_2)$ defined over the same set of nodes. Let us also suppose that their respective MST: $\mathcal{MST}_1 = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \mathbf{W}_1)$ and $\mathcal{MST}_2 = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \mathbf{W}_2)$ have the same structure, i.e. the same nodes and edges but different edge weights. Let $\oplus$ be a function such that:*

$$\begin{aligned} \forall (x_1, x_2, y_1, y_2) \in \mathbb{R}_+^4, (x_1 \leq x_2) \text{ and } (y_1 \leq y_2) \\ \Rightarrow \oplus(x_1, y_1) \leq \oplus(x_2, y_2) \end{aligned} \tag{7}$$

Then the tree $T_{1,2} = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \oplus(\mathbf{W}_1, \mathbf{W}_2))$ is a MST of the graph $\mathcal{G}_{1,2} = (\mathcal{V}, \mathcal{E}, \oplus(\mathbf{W}_1, \mathbf{W}_2))$.

Proof. Let e_{pq} be any edge of $\mathcal{G}_1/\mathcal{G}_2$ (they share the same set of edges, but not the same edge weights). By path optimality (cf. theorem 1), $\forall \mathsf{e}_{st} \in \mathfrak{B}_{\mathcal{MST}}(p,q) : W^1_{st} \leq W^1_{pq}$ and $W^2_{st} \leq$

W^2_{pq}. Thus, if $\oplus$ verifies (7), we have: $\forall \mathsf{e}_{st} \in \mathfrak{B}_{\mathcal{MST}}(p,q) : \oplus(W^1_{st}, W^2_{st}) \leq \oplus(W^1_{pq}, W^2_{pq})$ Thus by path optimality, the tree $\mathrm{T}_{1,2} = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \oplus(\mathbf{W}_1, \mathbf{W}_2))$ is a MST of the graph $\mathcal{G}_{1,2} = (\mathcal{V}, \mathcal{E}, \oplus(\mathbf{W}_1, \mathbf{W}_2))$. □

Corollary 2.1. *Let us consider two hierarchies $(\mathcal{H}_1, \lambda_1)$, $(\mathcal{H}_2, \lambda_2)$ defined over the same graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{W})$, and constructed upon two MST sharing the same structure $\mathcal{MST}_1 = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \mathbf{W}_1)$ and $\mathcal{MST}_2 = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \mathbf{W}_2)$. Then theorem 2 ensures us that for any function $\oplus$ verifying (7), the MST of $\mathcal{G}_{1,2} = (\mathcal{V}, \mathcal{E}, \overline{\oplus(\lambda_1, \lambda_2)})$ is $T_{1,2} = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \oplus(\mathbf{W}_1, \mathbf{W}_2))$.*

Thus, when combining hierarchies with a function $\oplus$ verifying (7), one can simply apply this function to edge weights of both MST and directly infer the subdominant ultrametric associated with this combination:

$$\begin{cases} \forall \mathsf{e}_{pq} \in \mathcal{E}_{\mathcal{MST}}, \overline{\oplus(\lambda_1, \lambda_2)}(p,q) = \oplus(\lambda_1, \lambda_2)(p,q) \\ \forall \mathsf{e}_{pq} \notin \mathcal{E}_{\mathcal{MST}}, \overline{\oplus(\lambda_1, \lambda_2)}(p,q) = \\ \qquad \bigvee\{W_{st}, \mathsf{e}_{st} \in \mathfrak{B}_{pq} \subset \mathrm{T}_{1,2} = (\mathcal{V}, \mathcal{E}_{\mathcal{MST}}, \oplus(\mathbf{W}_1, \mathbf{W}_2))\} \end{cases} \tag{8}$$

This procedure is less computationally costly than the one consisting in computing the function $\oplus$ over all edges of the two complete graphs and extracting the subdominant ultrametric consequently, as in (Maia et al., 2017). Note that the condition given by (7) is verified by many two-variables functions of particular interest: the supremum, infimum, any linear combination with positive coefficients, as well as the logical operators AND and OR between probabilistic variables. To sum up, our approach consists in choosing a MST from the initial graph and then working with its structure to generate new hierarchies. Combining hierarchies is then often straightforward, and we can easily obtain structurings of the image translating complex yet understandable properties of it. We now present some of these possible combinations.

5.4. Different possible parallel combinations

Different possible parallel combinations of hierarchies are listed in table 1.

5.4.1. Supremum, infimum, and mean

Since we are generating different hierarchies starting from the same MST, and since the SUP, INF and MEAN functions verify (7) of theorem 2, we can simply compute these functions on the MST and infer the subdominant ultrametrics from them. Note that the same property applies to any linear combination (with positive coefficients) of hierarchies.

5.4.2. Logical operators of probabilistic ultrametrics

As we have noted it in previous sections, hierarchies have a discriminative power that allows us to discriminate contours in a controlled way. Among the possible hierarchies, the SWS model presents a versatility that makes it extremely interesting for the characterization of scenes or images. In addition, it provides to each contour of the fine partition a probability value. This facilitates their combinations as well as their interpretation. In this specific case when ultrametric values correspond to probabilities, new possible combinations can be considered through the effect of the boolean operators AND and OR between two ultrametrics associated with two probabilistic events. Their expressions for two input ultrametrics are given in table 1, supposing that the two events are independent. It appears that the two-variable functions that allow for their computation verify (7) of theorem 2, and we are thus provided with an efficient way to compute them when combining two hierarchies that share the same MST. Note that the NOT operator does not verify this property, and we are thus replaced in the general case of section 5.2 to compute the subdominant ultrametric for it. Using and combining probabilistic hierarchies is interesting for several reasons. First, it is a way to regroup different types of experiments in an homogeneous set of representations and homogeneous measures. Furthermore, starting from a large and homogeneous set of SWS hierarchies, one can combine them in an understandable manner. For instance, the AND and OR of two hierarchies have straightforward effects. Furthermore, this opens an exploration path for potentially complex combinations using binary logical expressions. We can indeed use all boolean operators and this way build hierarchies combining diverse characteristics, in a way that is more understandable and refined than with the SUP/INF combinations, and more interpretable than linear combinations such as the mean.

Table 1: Top: Supremum, infimum and mean of ultrametrics. Bottom: Probabilistic combinations of ultrametrics.

Type of combination	**Associated ultrametric**
INF $((\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2))$	SUP(λ_1, λ_2)
SUP $((\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2))$	INF(λ_1, λ_2)
MEAN $((\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2))$	$\frac{1}{2}(\lambda_1 + \lambda_2)$
AND $((\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2))$	$\lambda_1 \times \lambda_2$
OR $((\mathcal{H}_1, \lambda_1), (\mathcal{H}_2, \lambda_2))$	$\lambda_1 + \lambda_2 - (\lambda_1 \times \lambda_2)$
NOT $((\mathcal{H}, \lambda))$	$1 - \lambda$

6. Image characterization using distances between hierarchies

6.1. A structured richness of representations

As we have seen, we can generate at will various multi-level representations of the images highlighting various types of regions having given properties (e.g. elongation, surfaces equilibrium, contrast). An additional layer of complexity can be added via the possibility to combine hierarchies to obtain new ones. Each hierarchy then expresses certain particular images characteristics. To use the analogy of colors, each hierarchy is a black and white image resulting from the passage of a color image through a colored filter. Multiplying these filters makes it possible to obtain many black and white images that characterize the distribution of colors in the image. Finally, each hierarchy provided with its ultrametric is a metric space. The GH distance measures the distance between such metric spaces. This is why, as we have seen in section 3, the space of hierarchies provided with the GH distance can itself be structured as a metric space. We argue that the wealth of controlled understandable options to generate hierarchies, as long as the possibility to measure

their relative specific descriptive power, can lead to powerful image features.

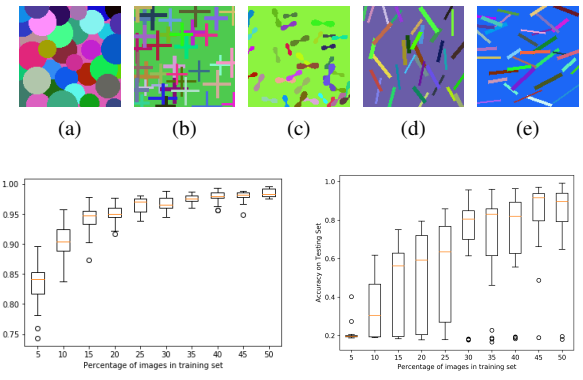
6.2. A condensed and descriptive image feature

Once provided with a family of hierarchies, one can wonder if there are ways to use the different information they provide to characterize images. The usual approach to do so it to extract information at various levels of the hierarchies (Farabet et al., 2012), which often requires a hard parameter-tuning. Furthermore, it obliterates the interesting property that hierarchical segmentations are more informative than flat segmentations as they capture simultaneously cluster structures at all levels of granularity. Instead, we propose to make use of the GH interhierarchy distance matrix, introduced in (4) of section 3.1.1, as a feature capturing the relative specific descriptive power of several hierarchies applied on the same image. This is a condensed representation leveraging the information provided by all the different levels of these different hierarchies. Since this matrix is symmetrical, we retain for each image its upper triangular part only. This constitutes a descriptor of the image for which we only had to specify the high-level parameters governing the hierarchies generation. We note that a similar approach has been already explored for comparison of Brain Networks in (Lee et al., 2011), and that it can be linked with persistent-based descriptors (Li et al., 2014).

6.3. Experimental results: Dead leaves process classification

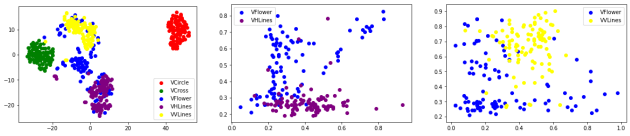
In a first experiment (Fehri et al., 2018), we highlight the discriminative power of the unsupervised hierarchical features we introduced, as long as their understandability. In the spirit of (Yan and Zhou, 2017), we want to test if these features capture pertinent information leading to a quicker understanding of the images. To do so, we consider a classification problem on a set of simulated images from different dead leaves process (Jeulin, 1997; Matheron, 1975), namely five classes with 100 images each with different primary grains: circles, crosses, flowers, horizontal and vertical lines. In a dead leaves model, two dimensional textured surfaces (which are called "leaves" or "primary grains") are sampled from a shape and size distribution and then placed on the image plane at random positions, occluding one another to produce an image. It is well-known that such a model creates images which share many properties with natural images such as scale invariance and other statistical properties (Pitkow, 2010). For each of these images, we compute the following hierarchies: trivial, surface-based SWS hierarchies with structuring elements of various sizes and forms (cross, circle, diagonals, horizontal and vertical lines), as long as logical combinations AND and OR of these SWS hierarchies. Then we generate for each of these images the interhierarchy distance matrices of (4).

We can then use these matrices as features in a classical classification pipeline using a linear support vector machines (SVM) to classify images of each class. We notice that the system can learn with very few examples how to discriminate properly these five classes. For comparison, we conduct the same experiment using a Convolutional Neural Network (CNN) with



(f) Linear SVM on proposed features (g) CNN[1] on proposed features

Fig. 3: (a)-(e) False-color representation of simulated images by dead leaves model with different primary grains. (f)-(g) Accuracy vs the number of images in the training set for 25 repetitions.



(a) t-SNE (b) L_1-SVM (c) L_1-SVM

Fig. 4: (a) We notice that the classes "Flowers" and "Horizontal Lines" are not well separated (b)These two distances between hierarchies provide a geometrical understanding of the images content. Projecting along these features does indeed separate these classes efficiently. (c) The same can be done for example for the classes "Flowers" and "Vertical Lines"

a two-layers architecture [1] without image augmentation for a fair comparison. In Fig. 3(f)(g) are represented for both experiments the evolutions of the average F-score with respect to the percentage of images used in the training set. In the first experiment (using the distance matrices as features), we notice that using only 5% of them (so 25 images out of 500) already leads to a 85% F-score over the remaining images, and that this figure quickly goes up. In the CNN experiment, the number of required training images to get to the same results is significantly larger (≈ 225). It is thus as if, on the contrary to CNN that have a black-box behavior, our approach shows what is often referred to as an "aha moment", i.e. a moment of sudden realization and comprehension (Yan and Zhou, 2017). This translates a form of understanding of the content of the image, which is corroborated by the study of the importance of which specific interhierarchy distances were the more useful to discriminate between two types of classes. For example, discriminating between horizontal and vertical lines will mainly be due to $\mathrm{d_{GH}}(\mathcal{H}_{surf-VertSE}, \mathcal{H}_{surf-HorizSE})$, while discriminating between crosses and circles will mainly be due to $\mathrm{d_{GH}}(\mathcal{H}_{surf-CrossSE}, \mathcal{H}_{surf-HexSE})$. A visualization of the quality of the features space thus generated

[1] (12 Conv + 12 Conv + MaxPolling(3×3) + Dropout(0.3)) + (24 filters + 24 filters + MaxPolling(3×3)+ Dropout(.5)) + FullyCon64 + Dropout (.5) + SoftMax. Categorical cross-entropy as loss function and adaptive gradient (Adagrad) as optimizer.

can be found in Fig. 4(a), where we project the features in a space of two dimensions using the t-SNE algorithm (Maaten and Hinton, 2008). Furthermore, using the variable selection method L_1-SVM (Zhu et al., 2004), we can isolate the more discriminative distances for two specific classes to separate. For example, the t-SNE visualization in Fig. 4(a) shows us that discriminating between the classes "Flowers" and "Horizontal Lines" is not straightforward. The more discriminative variable between these two classes is the distance between $\mathcal{H}_{surf-VertSE}$ and $\mathcal{H}_{\mathrm{AND}(surf-VertSE,surf-HexSE)}$: this is a geometrical interpretation of the image content, as they respectively capture straight lines and lines with a protuberance (i.e. flowers). Projecting the distances features onto the subspace of the two more discriminative variables properly separates these two classes, as can be seen in Fig. 4(b).

7. Conclusion

In this paper, we have introduced a coherent framework to analyze and characterize images using a multi-model hierarchical approach. In particular, combinations of hierarchies allow us for a rich and structured exploration of image properties. This combinatorial space has been structured and some of its properties studied thanks to the use of the Gromov-Hausdorff distance, and we have seen how interhierarchy distance matrices can be used as efficient and condensed image representations. To go further, a more detailed and formal study of combinations of hierarchies could be conducted, and the use of other possible distances between metric spaces considered. Among all possible combinations, the logical combinations of SWS hierarchies seem the most interesting to study further, as they pave the way to complex yet understandable combinations using binary logical expressions. Finally, one may use the interhierarchy distance matrices as features to do unsupervised characterization of an image space, to permit a qualitative understanding of their content, and to detect anomalies.

References

Achanta, R., Shanji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2012. SLIC super pixels compared to state-of-the-art super pixel methods. IEEE transactions on pattern analysis and machine intelligence .

Agarwal, P.K., Fox, K., Nath, A., Sidiropoulos, A., Wang, Y., 2015. Computing the gromov-hausdorff distance for metric trees, in: International Symposium on Algorithms and Computation, Springer. pp. 529–540.

Angulo, J., Jeulin, D., 2007. Stochastic watershed segmentation. International Symposium on Mathematical Morphology 1, 265–276.

Arbelaez, P., Maire, M., Fowlkes, C., Malik, J., 2011. Contour detection and hierarchical image segmentation. IEEE Transactions on pattern analysis and machine intelligence 33, 898–916.

Arbeláez, P., Pont-Tuset, J., Barron, J.T., Marques, F., Malik, J., 2014. Multiscale combinatorial grouping, in: Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 328–335.

Beucher, S., 1990. Segmentation d'images et morphologie mathématique. Ph.D. thesis. Ecole Nationale Supérieure des Mines de Paris.

Borg, I., Groenen, P., 2003. Modern multidimensional scaling: Theory and applications. Journal of Educational Measurement 40, 277–280.

Cousty, J., Najman, L., Kenmochi, Y., Guimarães, S., 2018. Hierarchical segmentations with graphs: quasi-flat zones, minimum spanning trees, and saliency maps. Journal of Mathematical Imaging and Vision 60, 479–502.

Farabet, C., Couprie, C., Najman, L., LeCun, Y., 2012. Learning hierarchical features for scene labeling. IEEE transactions on pattern analysis and machine intelligence 35, 1915–1929.

Fehri, A., Velasco-Forero, S., Meyer, F., 2017. Prior-based hierarchical segmentation highlighting structures of interest, in: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing, Springer. pp. 146–158.

Fehri, A., Velasco-Forero, S., Meyer, F., 2018. Characterizing images by the gromov-hausdorff distances between derived hierarchies. IEEE ICIP .

Felsenstein, J., 2014. Inferring phylogenies. volume 2. Sinauer associates Sunderland, MA.

Grimaud, M., 1992. New measure of contrast: the dynamics, in: Proceeding of SPIE, International Society for Optics and Photonics. pp. 292–305.

Gromov, M., 2007. Metric structures for Riemannian and non-Riemannian spaces. Springer Science & Business Media.

Guigues, L., Cocquerez, J.P., Le Men, H., 2006. Scale-sets image analysis. International Journal of Computer Vision 68, 289–317.

Hu, T., 1961. Letter to the editorthe maximum capacity route problem. Operations Research 9, 898–900.

Jeulin, D., 1997. Dead leaves models: From space tessellation to random functions, in: Advances in Theory and Applications of Random Sets. World Scientific Publishing, pp. 137–156.

Kaufman, L., Rousseeuw, P.J., 2009. Finding groups in data: an introduction to cluster analysis. volume 344. John Wiley & Sons.

Kiran, B.R., Serra, J., 2014. Global–local optimizations by hierarchical cuts and climbing energies. Pattern Recognition 47, 12–24.

Lee, H., Chung, M.K., Kang, H., Kim, B.N., Lee, D.S., 2011. Computing the shape of brain networks using graph filtration and gromov-hausdorff metric, in: International Conference on Medical Image Computing, Springer. pp. 302–309.

Li, C., Ovsjanikov, M., Chazal, F., 2014. Persistence-based structural recognition, in: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1995–2002.

Maaten, L.v.d., Hinton, G., 2008. Visualizing data using t-SNE. Journal of machine learning research 9, 2579–2605.

Machairas, V., Faessel, M., Cárdenas-Peña, D., Chabardes, T., Walter, T., Decencière, E., 2015. Waterpixels. IEEE transactions on image processing 24, 3707–3716.

Maia, D.S., de Albuquerque Araujo, A., Cousty, J., Najman, L., Perret, B., Talbot, H., 2017. Evaluation of combinations of watershed hierarchies, in: International Symposium on Mathematical Morphology, Springer. pp. 133–145.

Matheron, G., 1975. Random sets and integral geometry. Wiley New York.

Mémoli, F.and Sapiro, G., 2004. Comparing point clouds, in: ACM SIGGRAPH symposium on Geometry processing, ACM. pp. 32–40.

Meyer, F., 2015a. Stochastic watershed hierarchies, in: ICAPR, pp. 1–8.

Meyer, F., 2015b. The waterfall hierarchy on weighted graphs, in: International Symposium on Mathematical Morphology and Its Applications to Signal and Image Processing, Springer. pp. 325–336.

Meyer, F., Beucher, S., 1990. Morphological segmentation. Journal of visual communication and image representation 1, 21–46.

Najman, L., Cousty, J., Perret, B., 2013. Playing with kruskal: algorithms for morphological trees in edge-weighted graphs, in: Mathematical Morphology and Its Applications to Signal and Image Processing. Springer, pp. 135–146.

Pitkow, X., 2010. Exact feature probabilities in images with occlusion. Journal of vision 10, 42–42.

Prim, R.C., 1957. Shortest connection networks and some generalizations. Bell Labs Technical Journal 36, 1389–1401.

Vachier, C., Meyer, F., 1995. Extinction value: a new measurement of persistence, in: IEEE Workshop on Non Linear Signal/Image Processing, pp. 254–257.

Wei, X., Yang, Q., Gong, Y., Ahuja, N., Yang, M.H., 2018. Superpixel hierarchy. IEEE Transactions on Image Processing 27, 4838–4849.

Wolf, S., Schott, L., Kothe, U., Hamprecht, F., 2017. Learned watershed: end-to-end learning of seeded segmentation, in: IEEE International Conference on Computer Vision, pp. 2011–2019.

Xu, Y., Carlinet, E., Géraud, T., Najman, L., 2016. Hierarchical segmentation using tree-based shape spaces. Transactions on pattern analysis and machine intelligence 39, 457–469.

Yan, Z., Zhou, X.S., 2017. How intelligent are convolutional neural networks? arXiv preprint arXiv:1709.06126 .

Zhu, J., Rosset, S., Tibshirani, R., Hastie, T.J., 2004. 1-norm support vector machines, in: Advances in NIPS, pp. 49–56.