From Hyperconnections to Hypercomponent Tree: Application to Document Image Binarization

Benjamin Perret[1], Sébastien Lefèvre[1], Christophe Collet[1]
and Eric Slezak[2]

[1] LSIIT UMR 7005, University of Strasbourg CNRS,
LSIIT, Pôle API, Bd Sébastien Brant, BP 10413, 67412 Illkirch Cedex France.
{bperret,lefevre,c.collet}@unistra.fr
[2] University of Nice-Sophia Antipolis in the Cassiopée Laboratory (OCA, UMR 6202),
CNRS-Observatoire de la Côte d'Azur, Boulevard de l'Observatoire, BP 4229, 06304 Nice, France.
eric.slezak@oca.eu

Abstract In this paper, we propose an extension of the component tree based on flat zones to hyperconnections (h-connections). The tree is defined by a special order on the h-connection and allows non flat nodes. We apply this method to a particular fuzzy h-connection and we give an efficient algorithm to transform the component tree into the new fuzzy h-component tree. Finally, we propose a method to binarize document images based on the h-component tree and we evaluate it on the DIBCO 2009 benchmarking dataset: our novel method places first or second according to the different evaluation measures.
Keyworks Hyperconnection; Hierarchical Representation; Document Image Binarization.

1 Introduction

Hierarchical and tree based representations have become very topical in image processing. In particular, the component tree (or Max-Tree) has been the subject of many studies and practical works. Nevertheless, the component tree inherits the weaknesses of the flat zone approach, namely its high sensitivity to noise, gradients and difficulty to manage disconnected objects. Even if some solutions have been proposed to preserve the component tree [5, 4], it seems that a more general framework for *grayscale component tree* [1] based on non flat zones becomes necessary.

In this article, we propose a method to design grayscale component tree based on h-connections. The h-connection theory has been proposed in [7] and developed in [1, 3, 4, 8, 9]. It provides a general definition of the notion of connected component in arbitrary lattices. In Sec. 2, we present the h-connection theory and a method to generate a related hierarchical representation. This method is applied to a fuzzy h-connection in Sec. 3 where an algorithm is given to transform a Max-Tree into the new grayscale component tree. In Sec. 4, we illustrate the interest of this tree with an application on document image binarization.

2 H-component Tree

This section presents the basis of the h-connection theory [7, 1] and gives a definition of the h-component tree. The construction of the tree is based on the z-zones [1] of the h-connection, together with a special partial ordering. Z-zones are particular regions where all points generate the same set of hyperconnected (h-connected) components and the entire image can be divided into such zones. Under a given condition, the Hasse diagram obtained in this way is acyclic and provides a tree representation.

Let $\mathcal{L}$ be a complete lattice furnished with the partial ordering $\leq$, the infimum $\bigwedge$, the supremum $\bigvee$. The least element of $\mathcal{L}$ is denoted by $\bot = \bigwedge \mathcal{L}$. We assume the existence of a sup-generating family $\mathcal{S}$ in $\mathcal{L}$. A h-connection on $\mathcal{L}$ is given by a subset $\mathcal{C}^+$ of $\mathcal{L}$ which contains the h-connected elements of $\mathcal{L}$, and an overlap criterion $\Cap$ (i.e. a predicate on subsets of $\mathcal{L}$) such that:

1. $\bot \in \mathcal{C}^+$ and $\forall s \in \mathcal{S},\ s \in \mathcal{C}^+$: the least element and the sup-generators are h-connected.
2. $\forall A \subseteq \mathcal{C}^+,\ \Cap A \Rightarrow \bigvee A \in \mathcal{C}^+$ the supremum of overlapping elements is h-connected.

Moreover, $\Cap$ must be decreasing $\forall A \subseteq \mathcal{P}(\mathcal{L}), \not\Cap\ A \Rightarrow \forall b \in \mathcal{L}, \not\Cap\ \{b\} \bigcup A$. Being given an element $a \in \mathcal{L}$, the maximal h-connected elements lower than a are called the h-components of a; they are noted: $\gamma^*(a) = \{h \in \mathcal{C}^+ \mid h \leq a, \forall h' \in \mathcal{C}^+, h \leq h' \leq a \Rightarrow h = h'\}$. Then, the h-connected opening of an element $a \in \mathcal{L}$ by the marker $s \in \mathcal{S}$ is given by $\gamma_s(a) = \bigvee \{h \in \gamma^*(a) \mid s \leq h\}$. Contrary to connected openings of traditional set or lattice connections, the result of a h-connected opening is generally not h-connected. Finally, the z-operators [1] group sup-generators into equivalence classes. Being given an element $a \in \mathcal{L}$, the equivalence relation $\overset{a}{\sim}$ is defined by: $\forall s_1, s_2 \in \mathcal{S}$, $s_1 \overset{a}{\sim} s_2 \Leftrightarrow \gamma_{s_1}(a) = \gamma_{s_2}(a)$. Then, the z-operator $\zeta_a(s)$ of the sup-generator s is given by: $\zeta_a(s) = \bigvee \left\{ s' \in \mathcal{S} \mid s' \overset{a}{\sim} s \right\}$. The result of a z-operator is called a z-zone.

Fig. 1 shows how to decompose a function into its h-components and its associated z-zones. The h-connection used in this example is composed of all functions having a unique maximum [1].

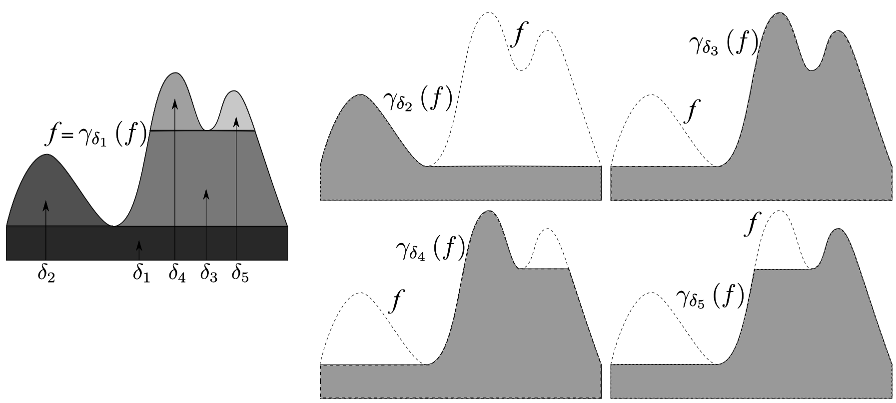

Figure 1: Decomposition of the function f with the h-connection of functions with a unique maximum: left) function f, five pulses $\delta_1, \ldots, \delta_5$ representatives of the five equivalence classes of the relation $\overset{f}{\sim}$. The different gray levels represent the results of the z-operators: $\zeta_{\delta_i}(f)$. The sup-generator δ_1 (and its equivalence class) are associated to the whole set of h-components of f and thus $\gamma_{\delta_1}(f) = f$, right) the h-openings of $\delta_2, \ldots, \delta_5$.

We now propose a way to build a h-component tree from the z-zones. In the following, we consider the lattice of functions $\mathcal{I} = \mathcal{L}^E$, where E is a discrete finite domain and $\mathcal{L}$ the space of values. The support of a function $f \in \mathcal{I}$ is the set of points where the function is different from $\bot$: $\mathrm{supp}(f) = \{p \in E \mid f(p) \neq \bot\}$. We define the partial ordering $\preceq$ on $\mathcal{I}$ by:

$$\forall x,\ y \in \mathcal{I}, \quad x \preceq y \Leftrightarrow \big(\mathrm{supp}(x) \subseteq \mathrm{supp}(y)\big) \wedge \big(y_{/\mathrm{supp}(x)} \leq x\big) \tag{1}$$

where $y_{/\mathrm{supp}(x)}$ is the function y restricted to $\mathrm{supp}(x)$. Fig. 2 demonstrates shows pairs of functions that are respectively comparable and not comparable according to $\preceq$.

Then, a graph-based representation of the image $a \in \mathcal{I}$ is obtained using the Hasse diagram (G_a, V_a) of the z-zones defined by the order $\preceq$. The graph (G_a, V_a) is defined by: $G_a = \zeta(a) \cup \{\bot\}$ and $V_a = \left\{(x, y) \in G_a^2 \mid x \neq y,\ x \preceq y, \forall z \in G_a,\ \ x \preceq z \prec y \Rightarrow x = z\right\}$. The least element $\bot$ is added to G to ensure that it is connected but can be omitted if it is not necessary. We can show that if each z-zone is either disjoint or comparable according to $\preceq$ then the graph is acyclic and we call it the h-component tree. Then, the h-component tree can be used in a similar way as the traditional Max-Tree to perform filtering, detection, segmentation with the advantage to support non flat nodes.

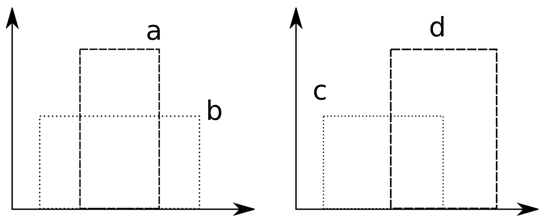

Figure 2: Left: two functions a and b such that $a \preceq b$. Right: two functions c and d that are not comparable according to $\preceq$.

3 Fuzzy H-Connection

We experiment the proposed h-component tree with a fuzzy h-connection [3] and we give an algorithm to transform a Max-Tree into the h-component tree. We now consider the lattice $\mathcal{I}$ of functions from a finite discrete domain E to the interval $[0,1]$. We assume that E is equipped with a primary set connection $\mathcal{C}$ [6]. We define the connectivity measure c_f of the image f, for all $x, y \in E$ (adapted from [3]) by $c_f(x,y) = \max_{M \in P_{x,y}} \min_{p \in M} f(p)$, with $P_{x,y} = \{M \in \mathcal{C} \,|\, x \in M, y \in M\}$ the set of all connected sets containing x and y. If $P_{x,y}$ is empty, we set $c_f(x,y) = 0$ meaning that x and y are not connected. The fuzzy h-connection $\mathcal{C}^+_{f,\tau}$ of level $\tau \in [0,1]$ is then defined by (adapted from [3]): $\mathcal{C}^+_{f,\tau} = \{f \in \mathcal{I} \,|\, \forall x,\ y \in E, 1 - \min(f(x), f(y)) + c_f(x,y) \geq \tau\}$. The discriminant points of fuzzy h-connected functions are indeed regional maxima: a function is fuzzy h-connected if, for each pair of regional maxima, the difference between the height of the lowest maximum and the height of the saddle point between the two maxima is lower than $1-\tau$. Figure 3 shows an example of such a τ h-connected function.

The fuzzy h-component tree can be computed from the Max-Tree using algorithm 1. The pseudocode assumes that each node is equipped with two attributes, the level and the peak level (highest level in its branch), and a function *child(n)* that returns the n-th child. The procedure *delete node*, deletes the given node and gives all its children to its father, while the procedure *delete branch* deletes the given node and all its children. Finally, the procedure *correctPointList* corrects the list of points attached to the given node according to its new level.

The algorithm proceeds in two steps. The first loop performs three actions: it removes irrelevant nodes that are neither local maxima nor saddle zones, it removes regional maxima that are absorbed by the parameter τ and it updates the level of all nodes according to τ. The second loop has two aims: it removes irrelevant nodes (same criteria as during the first pass) that may have appeared during the node removal process of the first pass and it corrects the set of points

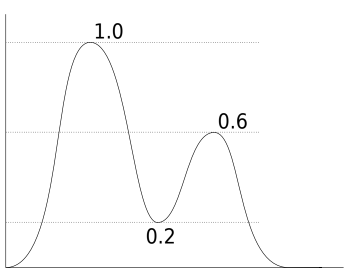

Figure 3: This function with two regional maxima at levels 1.0 and 0.6 belongs to $\mathcal{C}^+_{f,\tau}$ for all $\tau \leq 0.6$.

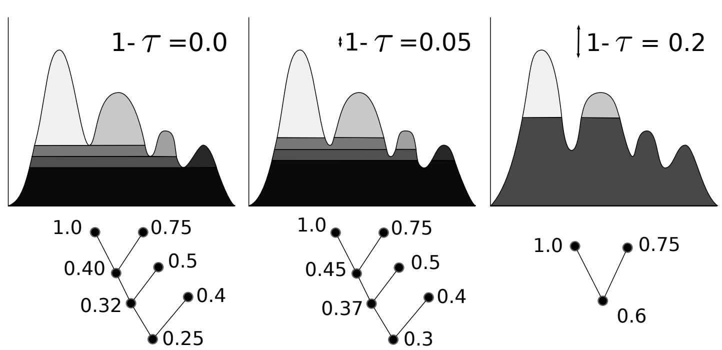

Figure 4: The h-component tree for a function with different values of τ. For each value of τ: the signal with the associated z-zones (top) and its h-component tree with the level of each node (bottom).

Algorithm 1: Procedure to transform a max-tree into the h-component tree of $\mathcal{C}^+_{f,\tau}$.

input : The image im, the max-tree of im, and the parameter τ of the fuzzy h-connection

```
for all nodes n from root to leaves do
    while n has exactly one child do
        node c = n.child(1);
        n.level = c.level;
        delete node c;
    if (n is not the root AND n.parent.level ≥ n.peakLevel) then delete branch of node n;
    else n.level = min(n.level + 1 − τ, n.peakLevel);
for all nodes n from root to leaves do
    while n has exactly one child do
        node c = n.child(1);
        n.level = c.level;
        delete node c;
    correctPointList(im,n);
```

associated to each node according to its new level. At the end of the algorithm, all attributes can be correctly computed according to the $\mathcal{C}^+_{f,\tau}$ h-connection. Figure 4 presents examples of h-component trees for different values of τ.

4 Application

We test the use of the h-component tree in the context of document image binarization and we propose a novel method based on background removal. The method processes the document image in three steps: 1) removal of the background using the h-component tree, 2) adaptive thresholding, and 3) post-processing. The background identification is based on the evolution of the area of the h-component tree nodes compared to their gray level. Formally, being given a leaf L of the fuzzy h-component tree, we look for the largest set of nodes $\{N_0, \ldots, N_m\}$ of respective level T_i and area A_i, such that $L = N_0$, $\forall i = 1, \ldots, m$, N_i is the father of N_{i-1}, $T_0 - T_m > c_4$ and:

$$\forall i = 1, \ldots, m: \; A_i < c_3 \vee \left(\frac{A_i - A_0}{T_i - T_0} < c_1 \wedge \frac{A_i - A_{i-1}}{T_i - T_{i-1}} < c_2 \right) \tag{2}$$

with c_1, c_2, c_3 and c_4 four thresholds. c_1 constrains the global slope and c_2 constrains its derivative. c_3 allows to neglect small nodes (strong noise on the curve) and c_4 defines a minimum contrast level. c_1 and c_2 are scale invariant while c_3 and c_4 defines minimal size and contrast below which we consider that a node is not significant. Finally, the local background is the reconstruction of the tree where the branch of N_m has been removed. The global background is defined as the infimum of all local backgrounds. Values of the thresholds were determined empirically by observing area-level curves of leaves belonging to the foreground and to the background. They are set to: $c_1 = -8 \times 10^4$, $c_2 = -2 \times 10^6$, $c_3 = 2 \times 10^3$, $c_4 = 0.1$ and the parameter τ of the fuzzy connection is set to $10/255$.

Then, the adaptive thresholding is based on the values of the image edges. The edges are detected using a Sobel operator with an Otsu thresholding. Finally, the post-processing is composed of a closing and an opening by reconstruction. We applied this method to the DIBCO 2009 benchmarking dataset [2]. This dataset was used for a contest (43 algorithms tested) during the ICDAR 2009 conference to evaluate binarization algorithms on a set of handwritten and printed document images representative of the various difficulties of this issue. We obtained a F-Measure score of 91.24 (that would have placed first in the contest) and a Peak Signal to Noise Ratio of 18.30 (second place). Fig. 5 presents two examples of handwritten document image binarization. It shows the quality of the background removal method and the subsequent thresholding.

Figure 5: Document image binarization, each line presents one document image from the DIBCO 2009 dataset [2]. From left to right: image, image after background removal, and result.

5 Conclusion

In this article, we proposed a general definition of grayscale component tree based on the h-connection theory which has the advantage to produce non flat zones. We show how it can be applied to a fuzzy h-connection recently proposed [3] and we give an algorithm to transform the Max-Tree into the grayscale component tree based on the fuzzy h-connection. Finally, we experiment our approach on the issue of document image binarization. The proposed algorithm is evaluated on the DIBCO 2009 dataset [2]: the very good results assert the interest of this tool for various image processing issues. In future works, we plan to explore the theoretical properties of our approach to derive general properties about the operators based on the h-component tree.

References

[1] U. Braga-Neto and J. Goutsias. A theoretical tour of connectivity in image processing and analysis. *JMIV*, 19(1):5–31, 2003.

[2] B. Gatos, K. Ntirogianni, and I. Pratikakis. ICDAR 2009 document image binarization contest (DIBCO). In *ICDAR*, pages 1375–1382, 2009.

[3] O. Nempont, J. Atif, E. Angelini, and I. Bloch. A new fuzzy connectivity measure for fuzzy sets. *JMIV*, 34(2):107–136, 2009.

[4] G.K. Ouzounis. *Generalized Connected Morphological Operators for Robust Shape Extraction*. PhD thesis, University of Groningen, 2009.

[5] G.K. Ouzounis and M.H.F. Wilkinson. Mask-based second-generation connectivity and attribute filters. *PAMI*, 29(6):990–1004, 2007.

[6] J. Serra. *Image Analysis and Mathematical Morphology. II: Theoretical Advances.* Academic Press, London, UK, 1988.

[7] J. Serra. Connectivity on complete lattices. *JMIV*, 9(3):231–251, 1998.

[8] M.H.F. Wilkinson. An axiomatic approach to hyperconnectivity. In *ISMM*, volume 5720 of *LNCS*, pages 35–46, 2009.

[9] M.H.F. Wilkinson. Hyperconnectivity, attribute-space connectivity and path openings: Theoretical relationships. In *ISMM*, volume 5720 of *LNCS*, pages 47–58, 2009.