Automatic Attribute Threshold Selection for Blood Vessel Enhancement

Kiwanuka, Fred N.; Wilkinson, Michael H.F.

Published in:
EPRINTS-BOOK-TITLE

Automatic Attribute Threshold Selection for Blood Vessel Enhancement

Fred N. Kiwanuka
Johann Bernoulli Institute for
Mathematics and Computer Science,
University of Groningen, The Netherlands
Faculty of Computing & Information
Technology, Makerere University
Kampala, Uganda
F.N.Kiwanuka@rug.nl

Michael H.F. Wilkinson
Johann Bernoulli Institute for
Mathematics and Computer Science,
University of Groningen, The Netherlands
m.h.f.wilkinson@rug.nl

***Abstract*—Attribute filters allow enhancement and extraction of features without distorting their borders, and never introduce new image features. These are highly desirable properties in biomedical imaging, where accurate shape analysis is paramount. However, setting the attribute-threshold parameters has to date only been done manually. This paper explores simple, fast and automated methods of computing attribute threshold parameters based on image segmentation, thresholding and data clustering techniques. Though several techniques perform well on blood-vessel filtering, the choice of technique appears to depend on the imaging mode.**

I. Introduction

Attribute filters [1], [2] allow fast, shape preserving filtering based on properties of the desired image features. By computing some property, or *attribute* of image components, and preserving only those components which have the desired attribute values, it is possible to enhance image features in a scale-invariant way [3]. Attribute filters are a subset of connected operators [1], [2], which means they preserve edges strictly. Image components can either be removed or remain intact but new ones do not emerge. This is a desirable property in biomedical image analysis where accurate shape analysis is of importance. For a recent review see [4].

In their simplest form, attributes are compared against an *attribute-threshold*. Those features with attributes above (or below) the threshold are preserved, the rest are removed. Choosing the *'the best'* attribute threshold λ is done manually, which is subjective.

Usually the threshold is obtained interactively [5] through trial and error. This is particularly tedious if the dynamic range of the attributes is large. Choosing *'the right'* λ is important because it determines what is retained or rejected besides the filtering criteria. In this research we explore automatic computation of this threshold, by adapting conventional automatic grey-level thresholding techniques as well as data clustering techniques to attribute threshold computation.

This work was funded by Nuffic under the NPT project on *Building sustainable ICT Training Capacity in Four Public Universities in Uganda.*

Attribute filter computation is discussed briefly in section II. Discussion of the various methods is in section III. Performance evaluation of the methods in 3D blood vessels enhancement and manual selection is discussed in section IV. We show that several automatic techniques obtain threshold values close to those selected manually in blood vessel enhancement in MR angiography.

II. Attribute Filtering

In the binary case, attribute filters [1], retain those connected components of an image, which meet certain attribute criteria. After computing the connected components, some property or attribute of each component is computed. A threshold is usually applied to these attributes to determine which components are retained, and which removed. Thus, the criterion usually has the form

$$\Lambda(C) = (Attr(C) \geq \lambda) \tag{1}$$

with C the connected component, $Attr(C)$ some real-valued attribute of C and λ the attribute threshold. More formally, attribute filters rely on connectivity openings γ_x, $x \in E$ with E the image domain. In the binary case, $\gamma_x(X)$ returns the foreground component to which x belongs if $x \in X$, and $\emptyset$ otherwise. After extracting the connected components using these connectivity openings, a trivial filter ψ^Λ, based on attribute criterion Λ is applied to each. These are defined as

$$\psi^\Lambda(C) = \begin{cases} C & \text{if } \Lambda(C) \text{ is true} \\ \emptyset & \text{otherwise.} \end{cases} \tag{2}$$

Finally, the attribute filter ψ_Λ based on criterion Λ is defined as

$$\psi_\Lambda(X) = \bigcup_{x \in E} \psi^\Lambda(\gamma_x(X)). \tag{3}$$

Thus, the attribute filter is the union of all connected components which meet the criterion Λ.

For a grey-scale image f, we compute these attributes for the connected components of threshold sets $X_h(f)$ defined as

$$X_h(f) = \{x \in E | f(x) \geq h\}. \tag{4}$$

1051-4651/10 $26.00 © 2010 IEEE
DOI 10.1109/ICPR.2010.566

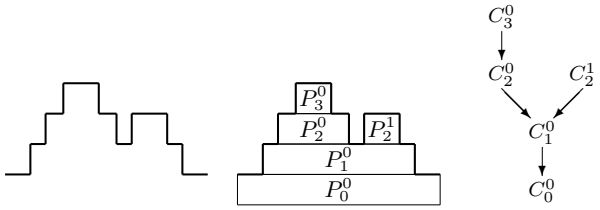


Fig. 1. A 1-D signal f (left), the corresponding peak components (middle) and the Max-Tree (right). Figure after [9].

In principle, we can apply the binary filter to each threshold set and stack the results. A more efficient approach uses the Max-Tree [2] data structure. The nodes C_h^k, with k the node index and h the gray level of the Max-Tree represent connected components for all threshold levels in a data set. These components are referred to as *peak components* and are denoted as P_h^k. The root node represents the set of pixels belonging to the background, and each node has a pointer to its parent. An example of a Max-Tree of a 1-D signal is given in Fig. 1. Each node contains a reference to its parent, its original and filtered grey level and its attribute value, or values, in the case of vector-attribute filtering [6].

The filtering process is separated into three stages: construction, filtering and restitution. During the construction phase, the Max-tree is built from the flat zones of the image, collecting auxiliary data used for computing the node attributes at a later stage. The attributes can be moment-based like non-compactness, elongation, flatness, sparseness [5], non-moment-based like sphericity [7] or size-based such as volume, surface area.

Once the attributes have been stored in the Max-Tree nodes, we can apply the attribute criterion of choice to each node to decide whether or not they should be retained. Various strategies of filtering are discussed in [2], [3], [8].

III. The Methods

A. Robust Automatic Threshold Selection (RATS)

RATS [10] computes a grey-level threshold using an edge-weighted average of the grey levels. We adapt this by defining the an attribute gradient Δ_A as along root path in the Max-Tree as

$$\Delta_A(C) = \frac{Attr(C) - Attr(\mathrm{par}(C)}{f(C) - f(\mathrm{par}(C))} \tag{5}$$

with $\mathrm{par}(C)$ the parent of node C in the Max-Tree, and $f(C)$ denotes the grey level of node C. As this is a one-sided gradient, it corresponds best to the mean $M_A(C)$ attribute value of C and $\mathrm{par}(C)$, i.e.

$$M_A(C) = \frac{Attr(C) + Attr(\mathrm{par}(C)}{2} \tag{6}$$

An optimal attribute threshold can then be computed as

$$T = \frac{\Sigma_{i=0}^{N-1} \Delta_A^2(C_i) M_A(C_i)}{\Sigma_{i=0}^{N-1} \Delta_A^2(C_i)}, \tag{7}$$

with N the number of nodes in the tree.

B. Maximum Entropy Based Classification (MaxEnt)

We adapted a method by [11] to compute maximum entropy. We considered attribute values of nodes as two clusters by clustering the histogram. The maximum entropy sum method is based on the maximization of the attribute measure between the two classes. Let $p_i = h[i]$ be an estimate of the probability node attribute, where i is histogram bin index of the node's attribute and N is the number of histogram bins. If we assume two classes of node attributes, $c1$ and $c2$, the maximum entropy $\psi(T)$ can be computed using the equation:

$$\psi(T) = \ln[p(c1)p(c2)] + \frac{H(t)}{p(c1)} + \frac{H_T - H(t)}{1 - p(c2)} \tag{8}$$

with

$$p(c1) = \Sigma_{i=0}^{t} p_i, \quad p(c2) = \Sigma_{i=t+1}^{N-1} p_i,$$

$$H_T = -\Sigma_{i=0}^{N-1} p_i \ln p_i, \quad \text{and,} \quad H(t) = -\Sigma_{t=0}^{t} p_i \ln p_i.$$

To compute an optimal threshold, we first compute the attributes for all nodes as before and sort them. We then compute a histogram (usually with about 1000 bins). Finally, we compute the entropy using (8) for each t, and select the attribute value corresponding to the bin with the maximum entropy as the optimum threshold.

C. Otsu Based Classification

The Otsu [12] computes an optimal threshold by maximizing the between-class variance $\sigma_{btn}^2(T)$:

$$\sigma_{btn}^2(T) = n_{c1}(T) * n_{c2}(T)[\mu_{c1}(T) - \mu_{c2}(T)]^2 \tag{9}$$

with

$$n_{c1}(T) = \Sigma_{i=0}^{T} p(i), \quad n_{c2}(T) = \Sigma_{i=T+1}^{N-1} p(i),$$

and $\mu_{c1}(T)$ and $\mu_{c2}(T)$ the means of clusters c1 and c2 respectively. We adapt this to attribute thresholding, by sorting the attributes in ascending order, and computing $\sigma_{btn}^2(T)$ for all attribute values present, and selecting the threshold that maximizes the $\sigma_{btn}^2(T)$.

D. K-means Based classification

In this method based on [13], which means we try to approximate the attribute distribution by two univariate distributions, the centroids of which are to be chosen optimally. We considered nodes' attribute values as two clusters. The algorithm is as follows:

- Compute the attribute as in section II.
- Initialize the number of clusters to 2 and assume the centroid of these clusters.
- Take any random attributes as the initial centroids.
- Iterate until convergence by:
  - Determining the centroid attributes
  - Determining the Euclidean distance of each attribute to the centroids
  - Group the node attributes based on minimum distance
- Compute the point of inter-cluster separation.

TABLE I
OPTIMAL ATTRIBUTE THRESHOLDS

Dataset	Manual	Otsu	MaxEnt	RATS	k-means
angiolarge	2.55	0.25	2.13	2.02	2.30
mrt16_angio	2.70	0.20	6.30	0.80	0.80
mrt16_angio2	3.60	0.020	3.20	0.90	3.70
aneurysm	2.00	0.25	3.37	2.34	4.68
angiolarge(kflat)	2.50	2.02	4.60	2.35	3.74

The optimal threshold is the point of inter-cluster separation which is equivalent to the mean of final centroids after convergence. This method can be generalized to multiple clusters most easily.

IV. RESULTS AND DISCUSSION

To test the performance of the methods, We ran tests on two time-of-flight magnetic resonance angiograms (MRA) from http://www.volvis.org (mrt16_angio, mrt16_angio2) and a phase-contrast MRA (angiolarge), all at 12 bit, and a CT angiogram (aneurysm) (also from http://www.volvis.org) at 8 bit grey-level resolution, using the *non-compactness* attribute [5]. We also asked 4 users to interactively find an optimal threshold for extracting blood vessels from the data sets. The average value from the user was used as the basis of comparison.

A. Blood Vessel Filtering

The results are presented in Table I. On angiolarge the methods based on RATS, MaxEnt and k-means are closer to the average response value from the users while Otsu based is too low because so many attribute values are close to zero as shown by the left-most histogram in Fig. 3. The nodes' non-compactness attribute histogram is skewed. Apparently, the other methods are not biased by this, as can be seen in Fig. 2. The three methods also suppress noise well, with the k-means being slightly better that the others.

On the aneurysm data set, RATS, performed well, but all others struggled. Otsu was swamped by noise as before, MaxEnt and k-means removed too much vascular structure.

Time-of-Flight MRAs are very difficult to filter and very noisy. On the mrt16_angio all values from the methods are either too low or too large as compared with the manual value. But on the mrt8_angio2 MaxEnt and k-means are closer to the manually obtained value and are more robust while Otsu and RATS continue to struggle because of so many nodes' attribute values close to zero.

In terms of computation speed the Otsu, maximum entropy and RATS are very fast (0.36 s), whereas k-means is somewhat slower (1.7 s) on a standard Core 2 Duo E8400 at 2.0 GHz for angiolarge, but not prohibitively so.

B. The Effect of k-flat zones

To further improve the performance of the methods we adapted hyperconnectivity k-flat zones [8], which are connected regions of maximal extent, where the total grey level variation is not more than k. This restriction to grey-level range automatically restricts the size to which the regions can grow yielding overlapping pseudo-flat zones which improves enhancement of internal details. The effect of using k-flat zones means that any node in the Max-Tree which within k of an extremum is not considered an independent entity, and their attributes are ignored in any further computation.

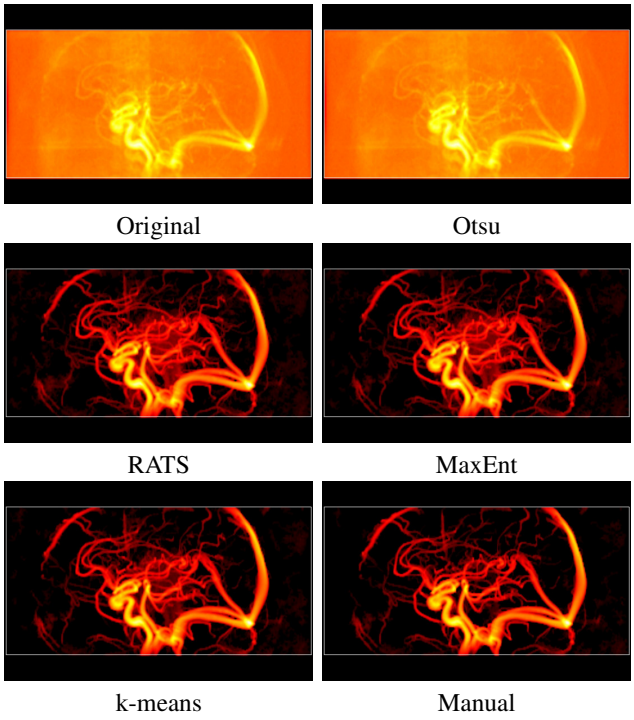
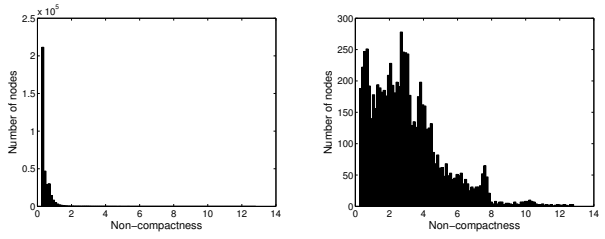


Fig. 2. Blood-vessel filtering on volume `angiolarge`, showing maximum-intensity projections of the original and filtered volumes for the different attribute threshold values from the methods in Table I.

For our purposes, this removes a large number of low-contrast feature, which are predominantly noise, from the computation of the optimal threshold. The effect of this can be seen in Figure 3, right-most graph. The number of nodes has reduced substantially with nodes that do not contribute significantly to the object removed. The proper value of k depends on the data set, and for angiolarge a value of 70 gave good results. From Table I, the Otsu computed value increased by 900% at $k = 70$ and by doing so we are improving the distribution of the histogram by making it less skewed as seen in Fig. 3. The improvement was also reflected in filtering the angiolarge in Fig. 4 where the blood vessels were better filtered as compared to Otsu in Fig. 2. RATS appears to be quite robust to changes in k, but the results of k-means and MaxEnt became worse, becoming too restrictive and showing substantial loss of small vessels.

V. CONCLUSION

In this paper we presented methods for automatic computation of the attribute threshold parameter. The methods are simple, fast and in several cases perform well against



(a) $k = 0$ (b) $k = 70$

Fig. 3. Attribute histograms for $k = 0$ and $k = 70$, showing the effect of the removal of low contrast features from the tree.

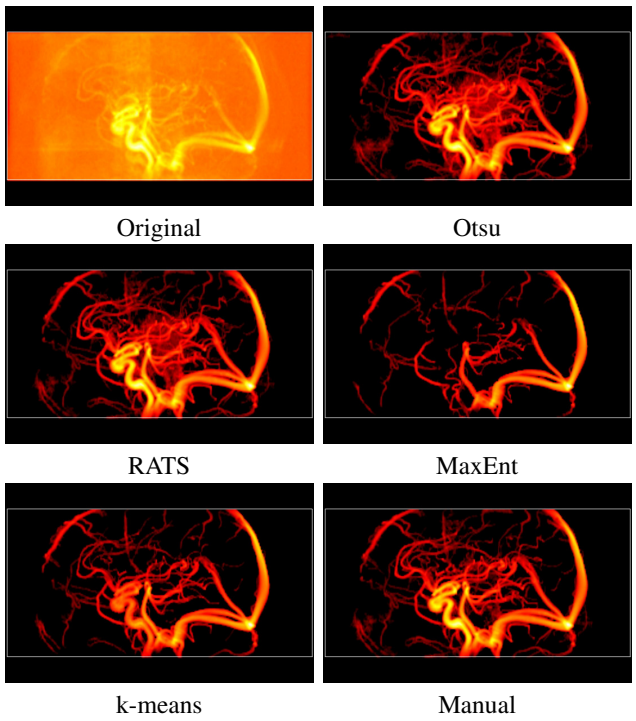


Fig. 4. The effect of k-flat filtering for $k = 70$, using the same methods as in Fig. 2.

the manual interactive method. Which method performed best depended on the data set, probably due to the differing imaging modes. We also show that the Otsu method can be further improved by using hyperconnectivity based on k-flat zones, but this does introduce a new parameter to set. RATS performed well on three data sets, but failed on the time-of-flight MRAs. Of the latter mrt16_angio was problematic to all methods, which might indicate that we need a different attribute in this case.

Though the focus was on Max-Tree based filtering, any other tree structure used for connected filtering could be used instead. All methods extend without change to auto-dual filtering using level-line trees [14] and colour filtering using, e.g., the binary partition tree [15]. All except the RATS-based method, which relies explicitly on a unique parent/child relationship could be extended to even more general connected filters [16], [17].

Apart from automatic filtering, the methods could be used to guide the user rather than to prescribe a value, especially when the attribute's dynamic range is large. In future, we shall consider multilevel thresholding which we believe can improve the accuracy of attribute threshold computation, and the use of multiple attributes, e.g. using vector-attribute filtering [6]. The method based on k-means filtering is particularly relevant in this case, because it extends most easily to high dimensional data. Other unsupervised clustering methods could also be included in such a framework.

REFERENCES

[1] E. J. Breen and R. Jones, "Attribute openings, thinnings and granulometries," *Comp. Vis. Image Understand.*, vol. 64, no. 3, pp. 377–389, 1996.

[2] P. Salembier, A. Oliveras, and L. Garrido, "Anti-extensive connected operators for image and sequence processing," *IEEE Trans. Image Proc.*, vol. 7, pp. 555–570, 1998.

[3] E. R. Urbach, J. B. T. M. Roerdink, and M. H. F. Wilkinson, "Connected shape-size pattern spectra for rotation and scale-invariant classification of gray-scale images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, pp. 272–285, 2007.

[4] P. Salembier and M. H. F. Wilkinson, "Connected operators: A review of region-based morphological image processing techniques," *IEEE Signal Processing Magazine*, vol. 26, no. 6, pp. 136–157, 2009.

[5] M. A. Westenberg, J. B. T. M. Roerdink, and M. H. F. Wilkinson, "Volumetric attribute filtering and interactive visualization using the max-tree representation," *IEEE Trans. Image Proc.*, vol. 16, pp. 2943–2952, 2007.

[6] E. R. Urbach, N. J. Boersma, and M. H. F. Wilkinson, "Vector-attribute filters," in *Mathematical Morphology: 40 Years On, Proc. Int. Symp. Math. Morphology (ISMM) 2005*, Paris, 18-20 April 2005, pp. 95–104.

[7] F. N. Kiwanuka, G. K. Ouzounis, and M. H. F. Wilkinson, "Surface-area-based attribute filtering in 3D," in *Proceedings of the 9th ISMM '09*. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 70–81.

[8] G. K. Ouzounis and M. H. F. Wilkinson, "Hyperconnected attribute filters based on k-flat zones," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2010, in press.

[9] M. H. F. Wilkinson, H. Gao, W. H. Hesselink, J. E. Jonker, and A. Meijster, "Concurrent computation of attribute filters using shared memory parallel machines," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 10, pp. 1800–1813, 2008.

[10] J. Kittler, J. Illingworth, and J. Föglein, "Threshold selection based on a simple image statistic," *Comp. Vision Graph. Image Proc.*, vol. 30, pp. 125–147, 1985.

[11] T. Pun, "A new method for grey-level picture thresholding using the entropy of the histogram," *Signal Processing*, vol. 2, no. 3, pp. 223 – 237, 1980.

[12] N. Otsu, "A threshold selection method from gray-level histograms," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 9, no. 1, pp. 62–66, Jan. 1979.

[13] J. MacQueen, "Some methods for classification and analysis of multivariate observations." in *Proc. 5th Berkeley Symp. Math. Stat. Probab.*, 1967, pp. 281–297.

[14] P. Monasse and F. Guichard, "Scale-space from a level lines tree," *J. Vis. Commun. Image Repres.*, vol. 11, pp. 224–236, 2000.

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

[16] P. Soille, "Beyond self-duality in morphological image analysis," *Image and Vision Computing*, vol. 23, pp. 249–257, 2005.

[17] ——, "Constrained connectivity and connected filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 7, pp. 1132–1145, July 2008.