Watersheds for Supervised Classification

Aditya Challa, Sravan Danda, B Daya Sagar, Laurent Najman

Watersheds for Supervised Classification

Aditya Challa, *Student Member, IEEE,* Sravan Danda, *Student Member, IEEE,* B. S. Daya Sagar, *Senior Member, IEEE* and Laurent Najman, *Senior Member, IEEE*

***Abstract*—Watershed technique from mathematical morphology (MM) is one of the most widely used operators for image segmentation. Recently watersheds are adapted to edge weighted graphs, allowing for wider applicability. However, a few questions remain to be answered - (a) How do the boundaries of the watershed operator behave? (b) Which loss function does the watershed operator optimize? (c) How does watershed operator relate with existing ideas from machine learning. In this article, a framework is developed, which allows one to answer these questions. This is achieved by generalizing the maximum margin principle to maximum margin partition and proposing a generic solution, MORPHMEDIAN, resulting in the maximum margin principle. It is then shown that watersheds form a particular class of MORPHMEDIAN classifiers. Using the ensemble technique, watersheds are also extended to ensemble watersheds. These techniques are compared with relevant methods from literature and it is shown that watersheds perform better than SVM on some datasets, and ensemble watersheds usually outperform random forest classifiers.**

***Index Terms*—Classification, Machine Learning, Mathematical Morphology, Maximum Margin Principle, Watersheds**

I. INTRODUCTION

THE problem of supervised classification is stated as - Given a labelled dataset $\{(x_i, y_i)\}$, find the labels for unlabelled data points $\{\hat{x_i}\}$. There exists several possible approaches to solve the problem [1]. One of the classic approaches is that of using Support Vector Machines (SVM) which relies on maximum margin classifier [2]. Several of these techniques depend on vector space structure of the underlying space. An alternate view can be proposed, based on lattices, by using ideas from Mathematical Morphology (MM). In this article, we analyze the use of watersheds from MM for supervised classification.

Mathematical Morphology is a theory of non-linear operators using lattices, developed by G. Matheron and J. Serra in 1960s [3], [4]. One of the main operators in MM is that of *watersheds*. Originally developed for image segmentation, watersheds rely on either the *drop of water* principle or the *principle of flooding* to develop the algorithms for image segmentation (See watershed related chapters in [4] for detailed history). In [5], the authors extended the watershed principle to edge weighted graphs, and have established its links to the minimum spanning tree. The following algorithm describes a variant of the watershed algorithm proposed in [5] with arbitrary seeds, referred to as *Minimum Spanning Forest (MSF)-Watershed* in the rest of the article.

Aditya Challa, Sravan Danda and B.S.Daya Sagar are with the Systems Science and Informatics Unit, Indian Statistical Institute, Bangalore, Karnataka, India 560059. E-mail: aditya.challa.20@gmail.com, sravan8809@gmail.com, bsdsagar@isibang.ac.in

Laurent Najman is with Université Paris-Est, LIGM, Equipe A3SI, ESIEE, France. E-mail: laurent.najman@esiee.fr

Input: A finite edge weighted graph $G = (V, E, W)$, labelled seeds $S \subset V$.
Output: Partition of V

```
Set E' = ∅
for e = (e_x, e_y) in sorted edge set E do
    if both e_x and e_y are labelled then
        pass
    else
        E' ← E' ∪ (e_x, e_y)
        Assign the same label to both e_x and e_y.
    end if
end for
return Partition generated by E'.
```

Although the above algorithm is developed for the purposes of image segmentation, it can clearly be used for supervised learning as well. This is achieved by taking the vertex set $V = \{x_i\} \cup \{\hat{x_i}\}$. Then, taking the seeds to be the set $S = \{x_i\}$, and defining edge weights appropriately, one can use the watershed algorithm above for classification.

Watersheds have been used as a part of classification in images [6] and related algorithms have been used for data analysis as well. In [7], [8] the authors use the related image foresting transform [9] for supervised classification. In [10], the authors used watershed along with convolution neural networks (CNN) to obtain state-of-art results in CREMI challenge. Watersheds are also a part of the state of art image segmentation technique COB [11].

However, a few fundamental questions remain - (a) How do the boundaries of the watersheds behave when used as a classifier? (b) Which loss function does watershed optimize? (c) How does watershed relate with existing ideas from machine learning? To truly understand the applicability of watersheds for learning, it is important to understand the answers to above questions.

The aim of this article is to understand the behavior of watersheds when used as classifiers, and not to obtain state-of-art results. The main contributions are - (i) We develop the framework generalizing maximum margin principle to sets equipped with dissimilarity measure. This leads to *maximum margin partitions* (ii) We propose a simple classifier, MORPHMEDIAN, and prove that it always returns a maximum margin partition. (iii) It is then shown that watershed is a specific case of MORPHMEDIAN, and hence identifies the optimization problem solved by watersheds as classifier. (iv) Using the technique of ensembles, we extend the watersheds to *Ensemble Watersheds*. (v) MSF-Watershed, Ensemble Watersheds and related methods are compared on datasets from [12].

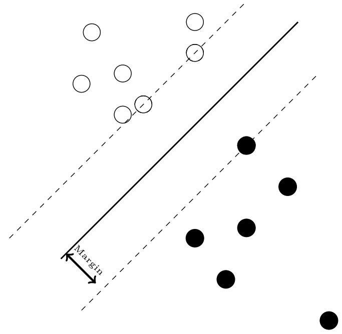


Fig. 1. Example explaining the the maximum margin principle. (see text)

II. Maximum Margin Partition and morphMedian

Recall that the technique of SVM is developed based on the principle of maximum margin. See [2] for details. From Fig. 1, SVM identifies the boundary such that the minimum *margin* for all points is maximized. We now generalize this principle. Assume that the set of points is given by V and there exists a measure ρ on V such that $\rho(x, y)$ indicates the dissimilarity between x and y. Note that ρ need not be a metric, in particular the constraint of triangle inequality need not hold. One can extend this to subsets $X, Y \subseteq V$ as well by

$$\rho(X, Y) = \min_{x \in X, y \in Y} \rho(x, y) \quad (1)$$

With this framework, the classification problem can be restated as - Given (V, ρ) and labelled sets $X_0, X_1 \subset V$ labelled 0 and 1 respectively, identify the partition $V = M_0 \cup M_1$ such that $X_0 \subset M_0$ and $X_1 \subset M_1$. Clearly $X_0 \cap X_1 = \emptyset$. For ease of exposition only two classes are considered, although all the definitions can be extended to a generic k class problem as well.

In this case, drawing parallels from Fig. 1, one can define the margin between a point $x \in X_0$ and the boundary using $\rho(x, M_1)$. For the entire set X_0, the margin is then defined as $\rho(X_0, M_1)$. Similarly, the margin for X_1 is defined as $\rho(X_1, M_0)$. Hence, one can use the principle of maximum margin in generic spaces (V, ρ) as well. This is defined below.

Definition 1 (Maximum Margin Partition)**.** *Let (V, ρ) be a set of points equipped with a dissimilarity measure. Let $X_0, X_1 \subset V$ denote the labelled subset of points with labels 0 and 1 respectively. A partition $V = M_0 \cup M_1$ with $X_i \subset M_i$ for $i = 0, 1$, is called the maximum margin partition if it maximizes*

$$\min \{\rho(X_0, M_1), \rho(X_1, M_0)\} \quad (2)$$

A natural question which arises from the above definition is - Can we characterize the maximum margin partitions? Consider the following definition of morphMedian.

Definition 2 (morphMedian)**.** *Given the notation as above, a partition $V = M_0 \cup M_1$ is called a* morphMedian *partition if*

1) $x \in M_0$ *if* $\rho(X_0, x) < \rho(X_1, x)$.
2) $x \in M_1$ *if* $\rho(X_1, x) < \rho(X_0, x)$.

In simple words, a morphMedian partition ensures that all points labelled 0 are closer to X_0 than X_1 and vice versa. Note that on the boundary, where $\rho(X_0, x) = \rho(X_1, x)$, the points can be labelled arbitrarily. The term morphMedian is used since this definition is inspired from the definition of morphological median defined in [13]. See [14] for more details about morphological median. These definitions result in the following theorem, central to this section.

Theorem 1. *Every* morphMedian *partition is a maximum margin partition.*

Proof. Firstly note that all morphMedian partitions have the same value for margin, since the labelling of two morphMedian partitions only differ in terms which are equal. Hence, if one can show that for any maximum margin partition, it is possible to construct a morphMedian partition with greater or equal margin, then the proof is done.

Let $M = M_0 \cup M_1$ be any maximum margin partition. If for all $x \in V$, the conditions in definition 2 hold, then M is a morphMedian partition and there is nothing to prove.

Otherwise, there exists a $z \in M_0$ such that $\rho(X_0, z) > \rho(X_1, z)$ or there exists a $z \in M_1$ such that $\rho(X_1, z) > \rho(X_0, z)$. If there exists a $z \in M_0$ such that $\rho(X_0, z) > \rho(X_1, z)$, then consider the following partition $\overline{M} = \overline{M}_0 \cup \overline{M}_1$ where

$$\begin{aligned} \overline{M}_0 &= M_0 \setminus z \\ \overline{M}_1 &= M_1 \cup z \end{aligned}$$

Then, we have that $\overline{M}$ has a margin greater than or equal to M, as shown below.

$$\begin{aligned} & \min \{\rho(X_0, \overline{M}_1), \rho(X_1, \overline{M}_0)\} \\ = & \min \{\rho(X_0, M_1), \rho(X_0, z), \rho(X_1, \overline{M}_0)\} \\ \geq & \min \{\rho(X_0, M_1), \rho(X_1, z), \rho(X_1, \overline{M}_0)\} \\ = & \min \{\rho(X_0, M_1), \rho(X_1, M_0)\} \end{aligned}$$

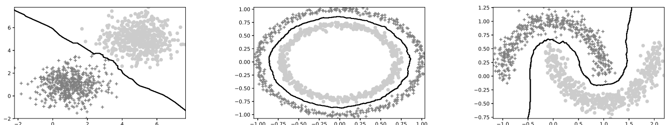
Similarly, it follows that if there exists a $z \in M_1$ such that $\rho(X_1, z) > \rho(X_0, z)$, then, once again it is possible to construct a partition with greater or equal margin.

Repeat the above procedure until there does not exist z which violates conditions in definition 2. The end of this procedure results in a morphMedian partition. Hence proved. □

Note the similarity between morphMedian and 1-Nearest Neighbor (1-NN) method for classification. The main difference between these two methods is that while 1-NN method classically considers a distance, morphMedian generalizes this to any dissimilarity measure.

III. Watersheds as Classifiers

Definitions 1, 2 and theorem 1 allow us to characterize the behavior of watersheds as classifiers. Let $G = (V, E, W)$ be an edge weighted graph. V is a set of vertices consists of both labelled and unlabelled points. $E \subset V \times V$ denotes the set of edges and $W : E \to \mathbb{R}^+$ denotes the weight (dissimilarity measure) assigned to each edge. Given the edge weighted graph, recall that watershed returns the partition given by algorithm described in section I.



(a) (b) (c)

Fig. 2. Figure illustrating boundaries obtained by MSF-watershed. (a) Two blobs dataset (b) Two circles dataset (c) Two moons dataset. All datasets are generated using sklearn [17]. The boundaries are sketched using the 1-Nearest Neighbor classifier as discussed in the text.

Define, for $x, y \in V$,

$$\rho_{max}(x,y) = \min_{\pi \in \Pi(x,y)} \max_{e \in \pi} W(e) \tag{3}$$

where $\Pi(x,y)$ indicates the set of all paths between x and y in G. $\rho_{max}(.,.)$ is also referred to as *pass value* [15]. Intuitively, it reflects the minimum height one has to reach to move from x to y. We then have the following theorem.

Theorem 2. *Given an edge weighted graph $G = (V, E, W)$, MSF-watershed returns a* MORPHMEDIAN *partition with respect to (V, ρ_{max}). And hence, it gives a maximum margin partition in the space (V, ρ_{max}).*

The proof of the above theorem follows from noting links between MSF-watershed and minimum spanning tree. For more details do refer to [16]. Theorem 2 characterizes the behavior of watersheds as a classifier.

Fig. 2 denotes the boundaries obtained when considering a few toy datasets. Intuitively, the watershed partitions the graph by removing edges between points which are farthest apart. This implies that the boundary will be in between two classes with the least density of points. This is reflected in Fig. 2 as well.

One seeming shortcoming of the watershed classifiers is that both the train and test datasets are assumed to be known. However, one can easily mitigate this thanks to the properties of MSF-watershed. To classify a new data point, not in the vertex set of the graph, one can simply use 1-Nearest Neighbor method without compromise. This is because, for any new data point, its 1-Nearest Neighbor is known to be on the Minimum Spanning Tree. This implies that, labelling a new data point after the initial labelling of the vertex set and labelling the data point along with the existing dataset would result in the same labels. This is summarized in the following proposition.

Proposition 1. *Using 1-Nearest Neighbor to classify 'new' data points is consistent with MSF-watershed classifier.*

A. Other Classifiers in the Framework of Maximum Margin Partition

The maximum margin partition framework discussed above also encompasses several known classifiers as well.

1) **IFT-SUM:** Instead of using $\rho_{max}(.,.)$ as in (3), one can use the following measure

$$\rho_{sum}(x,y) = \min_{\pi \in \Pi(x,y)} \sum_{e \in \pi} W(e) \tag{4}$$

This is the classic shortest path distance, which can be efficiently calculated using the Image Foresting Transform (IFT) as described in [9]. This too can be used as a classifier as described in [7].

2) **Random Walk (RW):** An alternate approach to extending the local edge weights to $V \times V$ is by using the distances given by the Laplacian, one of which is the random walk distance as described in [18]. This induces a measure on $V \times V$. This is also referred to as label propagation [19].

3) **Power-Watershed (PW):** In [20], [21] the authors extend the MSF-watershed to use watersheds along with random walk giving good results for seeded image segmentation. It is shown in [20] that it is indeed a special case of MSF-watershed and hence also fits into the maximum margin framework.

B. Ensemble Watersheds

Observe that watersheds rely on the edge weights of the graph. In fact, once the graph G has been chosen, apart from edge weights there are *no* parameters for the watershed classifier. Thus, in situations where there exist a lot of redundant features, it is possible that a simple L_2 norm between the features would not reflect the dissimilarities well. This can be improved by considering watershed using subset of features and *ensemble* the results. Ensemble is a technique used widely in machine learning, in particular for random forests. See [1] for more details and references about ensemble techniques. We now adapt this to MSF-watersheds as well.

The algorithm for using ensemble watersheds is described below.

Input: Edge weighted graph $G = (V, E, W)$, labelled seeds $S \subset V$, $\tau_S :=$ Sampling percentage of seeds, $\tau_F :=$ Sampling percentage of features.

Output: Labelling of V

```
for i ∈ {1, 2, ··· number_iterations} do
    Considering random subset of feature (τ_F percent), construct the new weight function W'.
    Using τ_S percent of labelled data points, compute the watershed using the graph G' = (V, E, W').
    Compute estimate of accuracy using out-of-box samples, that is samples which are not used for labelling.
end for
Using estimates of accuracy as weights, compute the weighted average of the labels obtained by watersheds.
return Labels computed by taking the maximum of the average accuracies.
```

Note that the adjacency relation E does not change across different estimators. This is because - either (a) adjacency relation is dictated by the domain, as is the case for images, in which case one need not change E or (b) the adjacency relation is computed using $k-$nearest neighbor graphs, in which case, intuitively, the data spans a lower dimensional manifold in a higher dimensional space. The $k-$nearest neighbor graph constructed is expected to reflect the structure of this manifold. Hence, it makes sense to use the same graph with weights dictated by a subset of features. Also, in general, constructing

TABLE I
RESULTS OBTAINED USING DIFFERENT METHODS ON DATASETS FROM [12]

Method	Digit1	USPS	COIL(binary)	BCI	g241c	COIL	g241n
MSF-Watershed	96.26±1.06	**95.70±0.80**	**99.76±0.24**	51.14±2.98	55.03±1.88	**95.59±0.83**	56.16±1.82
IFT-SUM	96.55±0.61	**95.33±0.68**	96.18±1.18	53.73±2.09	61.40±1.41	89.77±1.41	64.75±1.01
RW	**98.12±0.57**	91.70±1.17	95.86±1.17	53.45±2.58	70.27±5.18	91.42±0.95	76.32±3.47
PW	**97.94±0.54**	89.66±1.10	95.86±1.17	51.98±2.46	70.27±5.18	91.42±0.95	76.32±3.47
SVM	93.10±0.97	90.60±0.86	56.25±0.65	**59.44±3.67**	**84.38±0.97**	22.23±1.18	**84.79±1.29**
1-NN	96.55±0.61	**95.33±0.68**	96.18±1.18	53.73±2.09	61.40±1.41	89.77±1.41	64.78±1.01
RFC	95.99±0.56	88.53±0.70	92.63±1.17	**58.89±2.41**	75.92±0.60	91.09±1.32	73.45±1.00
Ensemblewatershed	**98.00±0.52**	92.69±1.25	**99.92±0.16**	52.23±2.20	65.20±3.27	**94.88±0.76**	68.39±2.62

the graph is a computationally expensive operation and using the same adjacency relation helps in implementing ensemble watersheds efficiently.

Remark: Although the technique of ensemble can be used with other techniques as well, the aim in this article is to understand MSF-watersheds and hence only ensemble of MSF-watersheds is considered.

IV. COMPARISON WITH OTHER CLASSIFIERS

In this section, we compare the results of the classifiers with the relevant classifiers - (i) Support Vector Machines (SVM) since the concept of maximum margin partitions developed here is an extension of the maximum margin principle on which SVM is based. In these experiments SVM is used with *rbf* kernel. (ii) 1-Nearest Neighbor (1-NN) due to the similarity between MORPHMEDIAN and 1-NN methods. (iii) Random Forest Classifier (RFC) since we consider the ensemble technique. All implementations of these classifiers are taken from [17].

The collection datasets are taken from chapter 21 of [12], since these datasets were designed to reflect several properties of real datasets. The $k-$nearest neighbor graphs are then constructed on these datasets which then are used as input to the classifiers. The k is chosen to be the least multiple of 10 so that the graph is connected. Also, each method is run 20 times for each datasets, taking random 20% of the data as train data. The results are shown in table I. The code to generate the results are available at [22].

In general, MSF-watershed performs better than SVM on a some datasets and worse on a few. Intuitively, MSF-watershed relies heavily on the manifold structure of the data - The data is assumed to be lower dimensional manifold in a higher dimensional space. Datasets *g241c* and *g241n* belong to this category. SVM on the other hand cannot predict highly non-linear boundaries which is the case with *COIL* dataset.

Ensemble watersheds in general work better than watersheds thanks to feature sampling. Ensemble watersheds also outperform random forest classifier in most cases. However, in cases where the entire feature set is a requirement for identification of the class, other methods work better. Thus, both random forest and ensemble watershed do not perform well on the *USPS* dataset, while the simple MSF-watershed and 1-NN achieves the best result.

Intuitively, the classifiers based on edge weighted graphs do not perform well on datasets, where local distances do not reflect the dissimilarities, for instance, in high dimensional scenarios. This can however be mitigated by using other machine learning architectures for estimating the weights.

V. CONCLUSION AND FUTURE PERSPECTIVES

To summarize, a framework for using watersheds as classifiers is developed. This is achieved by extending the maximum margin principle to maximum margin partitions. MORPHMEDIAN is proposed, and it is proved that MORPHMEDIAN always returns a maximum margin partition. It is also proved that watersheds are a specific case of MORPHMEDIAN, and hence returns a maximum margin partition. The technique is illustrated using toy datasets to understand the behavior of the boundaries. We also illustrate how watersheds can be combined with other ideas from machine learning, by considering the ensemble technique. Adapting the ensemble technique to watersheds is discussed in detail. Further, these techniques are compared with other relevant methods from literature on datasets from [12], showing that ensemble watersheds generally outperform random forests.

The aim of this article is not to present state-of-art results, but to understand the behavior of watersheds as classifiers better. It is expected that this understanding would result in better classifiers. For instance, one can infer from the framework that obtaining good measures of local edge weights E would result in better classifiers. Hence, one can use techniques such as neural networks to estimate the edge weights, which further can improve the accuracy of the classifiers. This is a topic of further research.

ACKNOWLEDGMENTS

Aditya Challa and Sravan Danda would like to thank Indian Statistical Institute for the funding provided to get this research done. Aditya Challa would also like to thank CEFIPRA for the support. The work of B. S. D. Sagar was supported by the Department of Science and Technology-Science and Engineering research Board (DST-SERB) and the Indian Space Research Organization (ISRO) under Grant EMR/2015/000853 SERB and Grant ISRO/SSPO/Ch-1/2016-17. Laurent Najman received funding from the Agence Nationale de la Recherche, contract ANR-14-CE27-0001 GRAPHSIP) and through “Programme d’Investissements d’Avenir" (LabEx BEZOUT ANR-10-LABX-58).

REFERENCES

[1] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, 2nd ed. Springer, 2009. [Online]. Available: http://www-stat.stanford.edu/~tibs/ElemStatLearn/

[2] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, Sep 1995. [Online]. Available: https://doi.org/10.1023/A:1022627411411

[3] J. Serra, *Image Analysis and Mathematical Morphology*. Orlando, FL, USA: Academic Press, Inc., 1983.

[4] L. Najman and H. Talbot, *Mathematical Morphology: From Theory to Applications*. John Wiley & Sons, 2013.

[5] J. Cousty, G. Bertrand, L. Najman, and M. Couprie, "Watershed cuts: Minimum spanning forests and the drop of water principle," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 8, pp. 1362–1374, 2009. [Online]. Available: https://doi.org/10.1109/TPAMI.2008.173

[6] Y. Tarabalka, J. Chanussot, and J. A. Benediktsson, "Segmentation and classification of hyperspectral images using watershed transformation," *Pattern Recognition*, vol. 43, no. 7, pp. 2367–2379, 2010. [Online]. Available: https://doi.org/10.1016/j.patcog.2010.01.016

[7] W. P. Amorim, A. X. Falcão, and M. H. de Carvalho, "Semi-supervised pattern classification using optimum-path forest," in *27th SIBGRAPI Conference on Graphics, Patterns and Images, SIBGRAPI 2014, Rio de Janeiro, Brazil, August 27-30, 2014*. IEEE Computer Society, 2014, pp. 111–118. [Online]. Available: https://doi.org/10.1109/SIBGRAPI.2014.45

[8] W. P. Amorim, A. X. Falcão, J. P. Papa, and M. H. de Carvalho, "Improving semi-supervised learning through optimum connectivity," *Pattern Recognition*, vol. 60, pp. 72–85, 2016. [Online]. Available: https://doi.org/10.1016/j.patcog.2016.04.020

[9] A. X. Falcão, J. Stolfi, and R. de Alencar Lotufo, "The image foresting transform: Theory, algorithms, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 19–29, 2004. [Online]. Available: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2004.10012

[10] S. Wolf, L. Schott, U. Köthe, and F. A. Hamprecht, "Learned watershed: End-to-end learning of seeded segmentation," in *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*. IEEE Computer Society, 2017, pp. 2030–2038. [Online]. Available: https://doi.org/10.1109/ICCV.2017.222

[11] K. Maninis, J. Pont-Tuset, P. Arbelaez, and L. V. Gool, "Convolutional oriented boundaries: From image segmentation to high-level tasks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 819–833, 2018. [Online]. Available: https://doi.org/10.1109/TPAMI.2017.2700300

[12] O. Chapelle, B. Schlkopf, and A. Zien, *Semi-Supervised Learning*, 1st ed. The MIT Press, 2010.

[13] J. Serra, "Hausdorff distances and interpolations," in *Proceedings of the Fourth International Symposium on Mathematical Morphology and Its Applications to Image and Signal Processing*, ser. ISMM '98. Norwell, MA, USA: Kluwer Academic Publishers, 1998, pp. 107–114. [Online]. Available: http://dl.acm.org/citation.cfm?id=295095.295119

[14] A. Challa, S. Danda, B. S. D. Sagar, and L. Najman, "Some properties of interpolations using mathematical morphology," *IEEE Trans. Image Processing*, vol. 27, no. 4, pp. 2038–2048, 2018. [Online]. Available: https://doi.org/10.1109/TIP.2018.2791566

[15] L. Najman, "On the equivalence between hierarchical segmentations and ultrametric watersheds," *Journal of Mathematical Imaging and Vision*, vol. 40, no. 3, pp. 231–247, 2011. [Online]. Available: https://doi.org/10.1007/s10851-011-0259-1

[16] J. Cousty, G. Bertrand, L. Najman, and M. Couprie, "Watershed cuts: Thinnings, shortest path forests, and topological watersheds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 925–939, 2010. [Online]. Available: https://doi.org/10.1109/TPAMI.2009.71

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[18] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, 2006. [Online]. Available: https://doi.org/10.1109/TPAMI.2006.233

[19] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," Carnegie Mellon University, Tech. Rep., 2002.

[20] C. Couprie, L. J. Grady, L. Najman, and H. Talbot, "Power watershed: A unifying graph-based optimization framework," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 7, pp. 1384–1399, 2011. [Online]. Available: https://doi.org/10.1109/TPAMI.2010.200

[21] L. Najman, "Extending the power watershed framework thanks to Γ-convergence," *SIAM J. Imaging Sciences*, vol. 10, no. 4, pp. 2275–2292, 2017. [Online]. Available: https://doi.org/10.1137/17M1118580

[22] A. Challa. Morphological median. [Online]. Available: https://github.com/ac20/morphMedian