Design of Pattern Classifiers using Optimum-Path Forest with Applications in Image Analysis

Alexandre Xavier Falcão

Visual Informatics Laboratory - Institute of Computing - University of Campinas

afalcao@ic.unicamp.br

www.ic.unicamp.br/~afalcao/talks.html

Motivation

- New technologies for data acquisition have provided large datasets with millions (or more) of samples for statistical analysis.

Motivation

- New technologies for data acquisition have provided large datasets with millions (or more) of samples for statistical analysis.
- We need more efficient and effective pattern recognition methods in this scenario.

Motivation

- New technologies for data acquisition have provided large datasets with millions (or more) of samples for statistical analysis.
- We need more efficient and effective pattern recognition methods in this scenario.
- The applications are in many fields of the sciences and engineering.

Motivation

- New technologies for data acquisition have provided large datasets with millions (or more) of samples for statistical analysis.
- We need more efficient and effective pattern recognition methods in this scenario.
- The applications are in many fields of the sciences and engineering.
- Our main focus has been on image analysis, where samples may be pixels, images, or image objects (e.g., contours, regions).

Research aim

We aim at effective and efficient image pattern classifiers for situations where

Research aim

We aim at effective and efficient image pattern classifiers for situations where

- clusters/classes may present arbitrary shapes in the feature space,

Research aim

We aim at effective and efficient image pattern classifiers for situations where

- clusters/classes may present arbitrary shapes in the feature space,
- clusters/classes may overlap each other at some extent, and

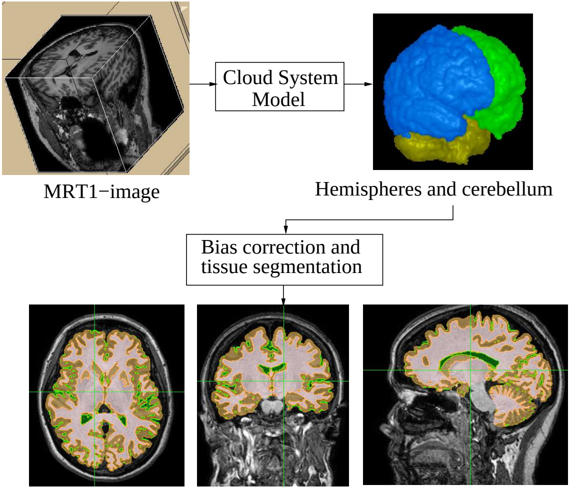
Research aim

We aim at effective and efficient image pattern classifiers for situations where

- clusters/classes may present arbitrary shapes in the feature space,
- clusters/classes may overlap each other at some extent, and
- a class may consist of multiple clusters.

Objectives

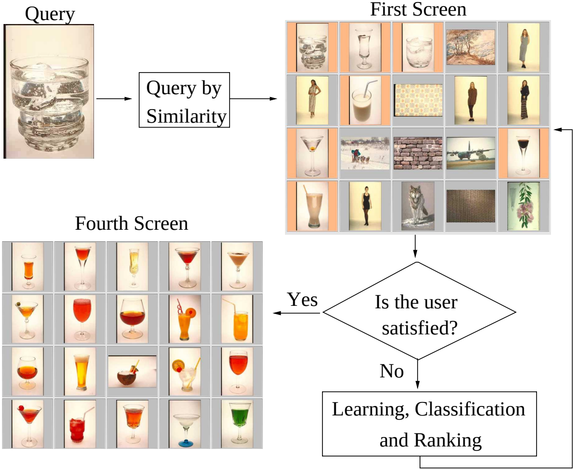
This talk presents a recent graph-based methodology for the design of pattern classifiers [1, 2], and two of its successful applications:



- Brain tissue segmentation in 3D MR-images [3, 4].

Objectives

This talk presents a recent graph-based methodology for the design of pattern classifiers [1, 2], and two of its successful applications:



- Brain tissue segmentation in 3D MR-images [3, 4].
- Content-based image retrieval using relevance feedback [5].

Advantages

- This methodology can be used for supervised, semi-supervised, and unsupervised learning.

Advantages

- This methodology can be used for supervised, semi-supervised, and unsupervised learning.
- It has been successfully applied to many other applications [6, 7, 8, 9, 10, 11, 12, 13, 14].

Advantages

- This methodology can be used for supervised, semi-supervised, and unsupervised learning.
- It has been successfully applied to many other applications [6, 7, 8, 9, 10, 11, 12, 13, 14].
- The method for brain tissue segmentation is fully automatic, accurate, and one of the fastest approaches to date.

Advantages

- This methodology can be used for supervised, semi-supervised, and unsupervised learning.
- It has been successfully applied to many other applications [6, 7, 8, 9, 10, 11, 12, 13, 14].
- The method for brain tissue segmentation is fully automatic, accurate, and one of the fastest approaches to date.
- The method for CBIR can satisfy the user in a few iterations of relevance feedback with significant gain in effectiveness over other state-of-the-art approaches.

Organization

- Overview of the pattern recognition problem for images.

Organization

- Overview of the pattern recognition problem for images.
- Our graph-based methodology.

Organization

- Overview of the pattern recognition problem for images.
- Our graph-based methodology.
- Unsupervised learning for brain tissue segmentation.

Organization

- Overview of the pattern recognition problem for images.
- Our graph-based methodology.
- Unsupervised learning for brain tissue segmentation.
- Supervised learning for content-based image retrieval.

Organization

- Overview of the pattern recognition problem for images.
- Our graph-based methodology.
- Unsupervised learning for brain tissue segmentation.
- Supervised learning for content-based image retrieval.
- How do we deal with large training sets?

Organization

- Overview of the pattern recognition problem for images.
- Our graph-based methodology.
- Unsupervised learning for brain tissue segmentation.
- Supervised learning for content-based image retrieval.
- How do we deal with large training sets?
- Conclusions and open problems.

Overview

- A sample s of a dataset $\mathcal{Z}$ may be a pixel, image, or object.

Overview

- A sample s of a dataset $\mathcal{Z}$ may be a pixel, image, or object.
- Dataset $\mathcal{Z}$ may be described by multiple pairs $D_i = (v_i, d_i)$, $i = 1, 2, \ldots, n$, called simple descriptors, where

- A sample s of a dataset $\mathcal{Z}$ may be a pixel, image, or object.
- Dataset $\mathcal{Z}$ may be described by multiple pairs $D_i = (v_i, d_i)$, $i = 1, 2, \ldots, n$, called simple descriptors, where
  - v_i assigns a feature vector $\vec{v}_i(s)$ to any sample $s \in \mathcal{Z}$, and

- A sample s of a dataset $\mathcal{Z}$ may be a pixel, image, or object.
- Dataset $\mathcal{Z}$ may be described by multiple pairs $D_i = (v_i, d_i)$, $i = 1, 2, \ldots, n$, called simple descriptors, where
  - v_i assigns a feature vector $\vec{v}_i(s)$ to any sample $s \in \mathcal{Z}$, and
  - d_i computes a distance $d_i(s,t)$ between samples s and t in the feature space (e.g., $d(s,t) = \|\vec{v}_i(t) - \vec{v}_i(s)\|$).

Overview

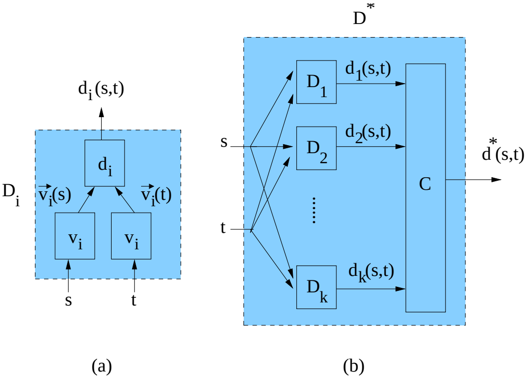
- A sample s of a dataset $\mathcal{Z}$ may be a pixel, image, or object.
- Dataset $\mathcal{Z}$ may be described by multiple pairs $D_i = (v_i, d_i)$, $i = 1, 2, \ldots, n$, called simple descriptors, where
  - v_i assigns a feature vector $\vec{v}_i(s)$ to any sample $s \in \mathcal{Z}$, and
  - d_i computes a distance $d_i(s,t)$ between samples s and t in the feature space (e.g., $d(s,t) = \|\vec{v}_i(t) - \vec{v}_i(s)\|$).
- Simple descriptors may require more complex distance functions to compare color properties [15], shape characteristics [16], and texture information [17] between samples.

Overview

A set $\mathbf{\Delta} = \{D_1, D_2, \ldots, D_k\}$ of simple descriptors may also be combined into a composite descriptor $D^* = (\mathbf{\Delta}, C)$, where C optimally combines their distance functions.

Overview

A set $\mathbf{\Delta} = \{D_1, D_2, \ldots, D_k\}$ of simple descriptors may also be combined into a composite descriptor $D^* = (\mathbf{\Delta}, C)$, where C optimally combines their distance functions.



We have found C by genetic programming [18].

Methodology

- A training set $\mathcal{T} \subset \mathcal{Z}$ is a graph $(\mathcal{T}, \mathcal{A})$, whose nodes are its samples and arcs are defined by an adjacency relation $\mathcal{A}$.

Methodology

- A training set $\mathcal{T} \subset \mathcal{Z}$ is a graph $(\mathcal{T}, \mathcal{A})$, whose nodes are its samples and arcs are defined by an adjacency relation $\mathcal{A}$.
- Representative samples (called prototypes) are estimated for each class/cluster, forming a subset $\mathcal{S} \subset \mathcal{T}$.

Methodology

- A training set $\mathcal{T} \subset \mathcal{Z}$ is a graph $(\mathcal{T}, \mathcal{A})$, whose nodes are its samples and arcs are defined by an adjacency relation $\mathcal{A}$.
- Representative samples (called prototypes) are estimated for each class/cluster, forming a subset $\mathcal{S} \subset \mathcal{T}$.
- The prototypes compete among them by offering optimum paths to the remaining samples in $\mathcal{T} \backslash \mathcal{S}$, according to some path-value function. We expect that a sample $s \in \mathcal{T}$ is more strongly connected to a prototype from its class/cluster than to any other.

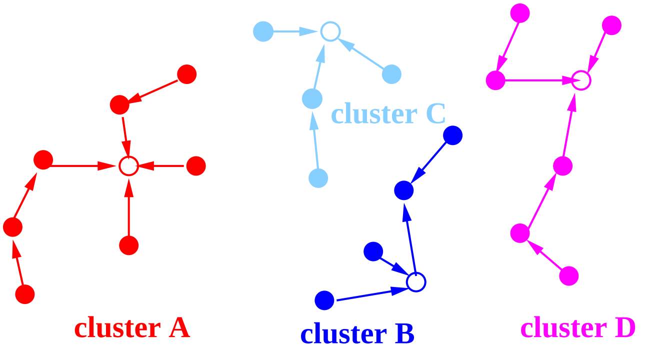
Methodology

- A training set $\mathcal{T} \subset \mathcal{Z}$ is a graph $(\mathcal{T}, \mathcal{A})$, whose nodes are its samples and arcs are defined by an adjacency relation $\mathcal{A}$.
- Representative samples (called prototypes) are estimated for each class/cluster, forming a subset $\mathcal{S} \subset \mathcal{T}$.
- The prototypes compete among them by offering optimum paths to the remaining samples in $\mathcal{T} \backslash \mathcal{S}$, according to some path-value function. We expect that a sample $s \in \mathcal{T}$ is more strongly connected to a prototype from its class/cluster than to any other.
- The result is an optimum-path forest (a pattern classifier) with roots in $\mathcal{S}$.

Methodology

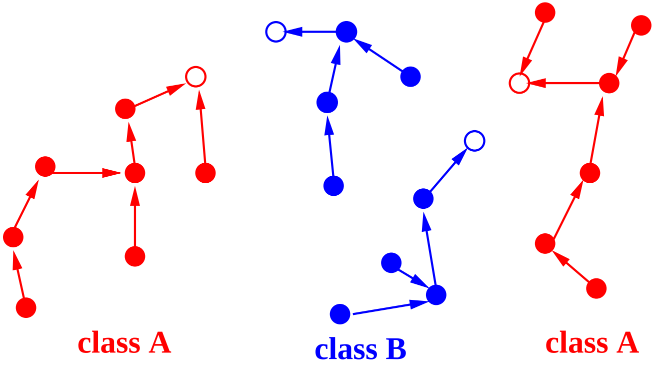
- A training set $\mathcal{T} \subset \mathcal{Z}$ is a graph $(\mathcal{T}, \mathcal{A})$, whose nodes are its samples and arcs are defined by an adjacency relation $\mathcal{A}$.
- Representative samples (called prototypes) are estimated for each class/cluster, forming a subset $\mathcal{S} \subset \mathcal{T}$.
- The prototypes compete among them by offering optimum paths to the remaining samples in $\mathcal{T} \backslash \mathcal{S}$, according to some path-value function. We expect that a sample $s \in \mathcal{T}$ is more strongly connected to a prototype from its class/cluster than to any other.
- The result is an optimum-path forest (a pattern classifier) with roots in $\mathcal{S}$.
- The class/cluster label of a new sample $t \in \mathcal{Z} \backslash \mathcal{T}$ is obtained from $d^*(s, t)$ using some samples $s \in \mathcal{T}$ and local attributes of the forest [4, 13].

In unsupervised learning, each prototype propagates a distinct cluster label to the remaining training nodes of its tree.



Each cluster is one optimum-path tree.

In supervised learning, each prototype propagates its class label to the remaining training nodes of its tree.



Each class is one optimum-path forest.

Methodology

- Training essentially maximizes (minimizes) a connectivity map

$$V(t) = \max_{\forall \pi_t \in \Pi(\mathcal{T},\mathcal{A},t)} \{f(\pi_t)\}$$

by considering the set $\Pi(\mathcal{T}, \mathcal{A}, t)$ of all paths with terminus $t \in \mathcal{T}$ and path-value function f.

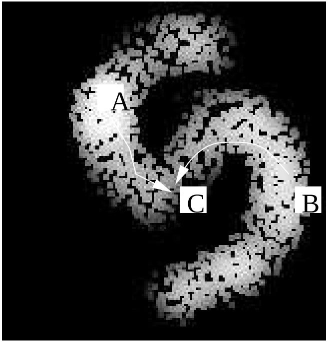
Methodology

- Training essentially maximizes (minimizes) a connectivity map

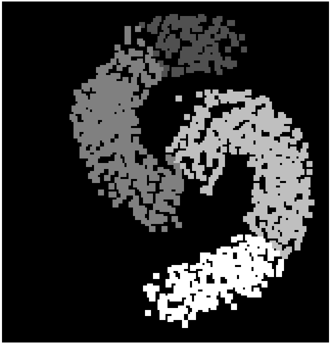
$$V(t) = \max_{\forall \pi_t \in \Pi(\mathcal{T},\mathcal{A},t)} \{f(\pi_t)\}$$

  by considering the set $\Pi(\mathcal{T}, \mathcal{A}, t)$ of all paths with terminus $t \in \mathcal{T}$ and path-value function f.
- The solution is obtained by dynamic programming (i.e., Dijkstra's algorithm extended to multiple sources and more general path-value functions) [19].

For clustering, we can estimate a probability density function (pdf) and the maxima of the pdf compete with each other, such that each cluster will be an optimum-path tree rooted at one maximum of the pdf.

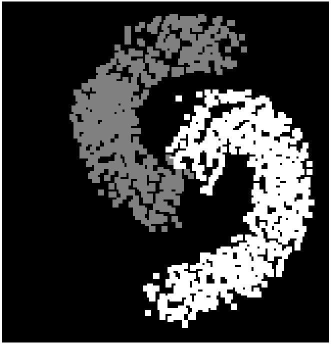


For clustering, we can estimate a probability density function (pdf) and the maxima of the pdf compete with each other, such that each cluster will be an optimum-path tree rooted at one maximum of the pdf.



The method first finds the maxima, and then propagates their labels. Irrelevant maxima may also be eliminated by choice of function f.

For clustering, we can estimate a probability density function (pdf) and the maxima of the pdf compete with each other, such that each cluster will be an optimum-path tree rooted at one maximum of the pdf.

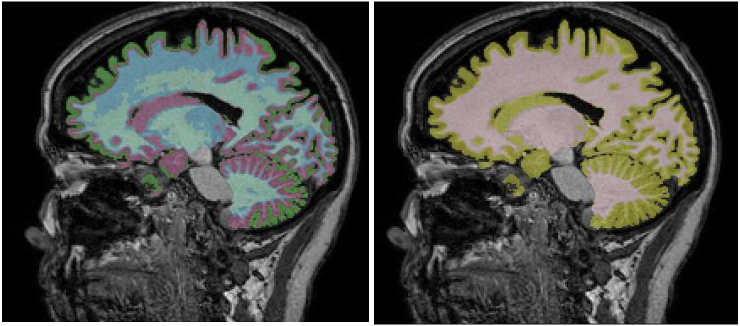


The method first finds the maxima, and then propagates their labels. Irrelevant maxima may also be eliminated by choice of function f.

Unsupervised learning

This may solve segmentation by pixel clustering or at least create a reduced number of regions to be merged into object/background.

This may solve segmentation by pixel clustering or at least create a reduced number of regions to be merged into object/background.



Prior information is usually required to assign class labels to the clusters.

Unsupervised learning

The unlabeled training samples form a knn-graph $(\mathcal{T}, \mathcal{A}_k)$ with adjacency relation

$$\mathcal{A}_k \quad : \quad (s,t) \in \mathcal{A}_k \text{ (or } t \in \mathcal{A}_k(s)) \text{ if } t \text{ is } k \text{ nearest neighbor of } s \text{ using the underlying distance space.}$$

The best value of k is the one whose clustering produces a minimum normalized graph cut in $(\mathcal{T}, \mathcal{A}_k)$.

The graph is weighted on the arcs $(s,t) \in \mathcal{A}_k$ by $d^*(s,t)$ and on the nodes by the pdf $\rho(s)$.

$$\rho(s) = \frac{1}{\sqrt{2\pi\sigma^2}|\mathcal{A}_k(s)|} \sum_{\forall t \in \mathcal{A}_k(s)} \exp\left(\frac{-{d^*}^2(s,t)}{2\sigma^2}\right)$$

where $\sigma = \frac{d_f}{3}$ and $d_f = \max_{\forall (s,t) \in \mathcal{A}_k}\{d^*(s,t)\}$. The pdf is usually normalized within an interval $[1, K]$ to facilitate the choice of relevant maxima and extra arcs are added in $\mathcal{A}_k$ to guarantee arc symmetry on the plateaus of the pdf.

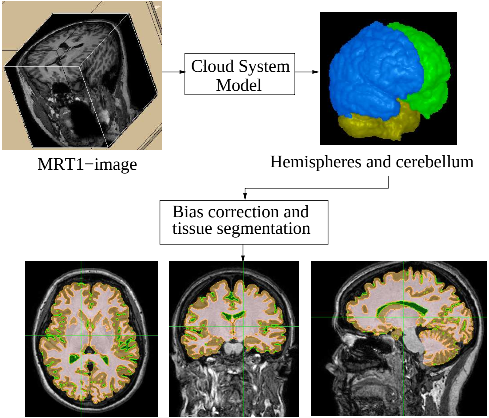
Unsupervised learning

The connectivity map $V(t)$ is maximized for

$$
\begin{aligned}
f_{\min}(\langle t \rangle) &= \begin{cases} \rho(t) & \text{if } t \in \mathcal{S} \\ \rho(t) - 1 & \text{otherwise} \end{cases} \\
f_{\min}(\pi_s \cdot \langle s, t \rangle) &= \min\{f_{\min}(\pi_s), \rho(t)\}
\end{aligned}
$$

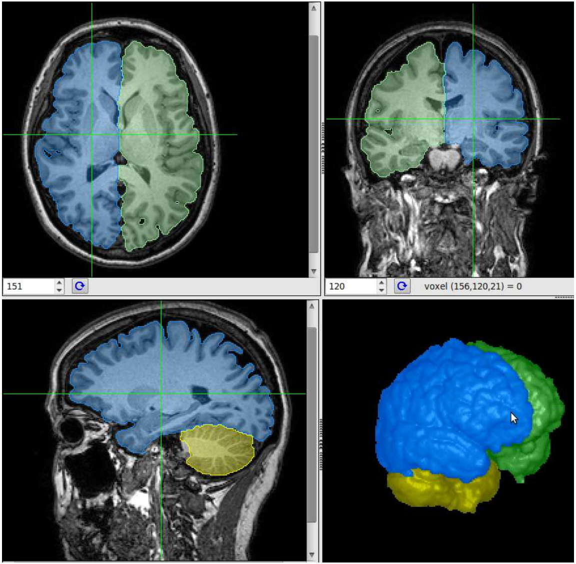
where $\mathcal{S}$ is the root set (maxima) found on-the-fly.

Brain tissue segmentation

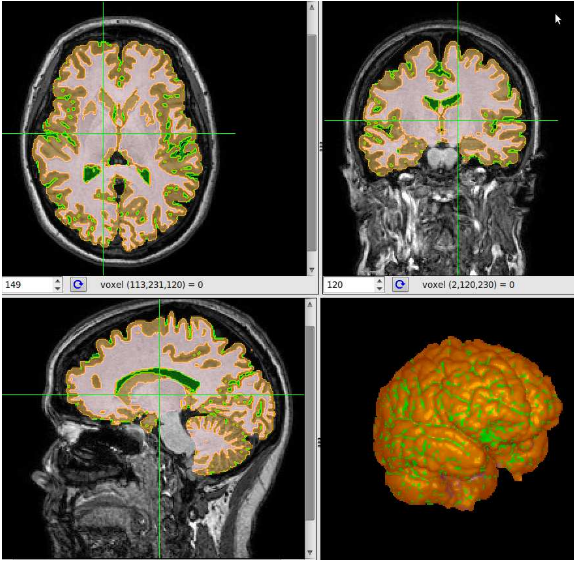


After brain segmentation and bias correction, brain voxels are first classified into CSF or GM+WM and then classified into GM or WM, because the method requires different parameters in each case.

Brain tissue segmentation



Brain tissue segmentation



Brain tissue segmentation

A cloud system model (CSM) consists of three elements [20, 21, 22]:

A cloud system model (CSM) consists of three elements [20, 21, 22]:

- A set of fuzzy objects (clouds), which indicates uncertainty regions with values strictly lower than 1 and higher than 0 for each object.

A cloud system model (CSM) consists of three elements [20, 21, 22]:

- A set of fuzzy objects (clouds), which indicates uncertainty regions with values strictly lower than 1 and higher than 0 for each object.
- A delineation algorithm, whose execution is constrained in the uncertainty region.

Brain tissue segmentation

A cloud system model (CSM) consists of three elements [20, 21, 22]:

- A set of fuzzy objects (clouds), which indicates uncertainty regions with values strictly lower than 1 and higher than 0 for each object.
- A delineation algorithm, whose execution is constrained in the uncertainty region.
- A criterion function, which assigns a score to any set of delineated objects.

A cloud system model (CSM) consists of three elements [20, 21, 22]:

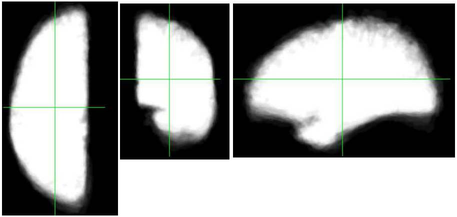
- A set of fuzzy objects (clouds), which indicates uncertainty regions with values strictly lower than 1 and higher than 0 for each object.
- A delineation algorithm, whose execution is constrained in the uncertainty region.
- A criterion function, which assigns a score to any set of delineated objects.

The method captures possible shape variations and groups them into a few CSMs.

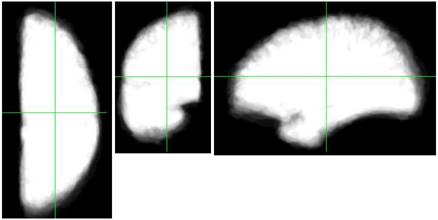
Brain tissue segmentation

Orthogonal cuts of the 3D clouds

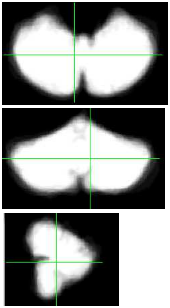
Right Hemisphere



Left Hemisphere

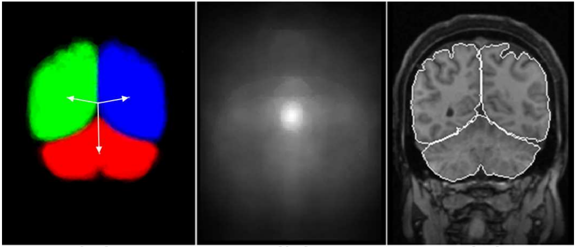


Cerebellum



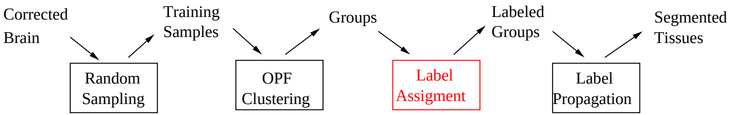
Brain tissue segmentation

Brain segmentation consists of a search for the translation to the image location which produces the highest score, when the reference point of the most suitable CSM is at that position.


(a) (b) (c)

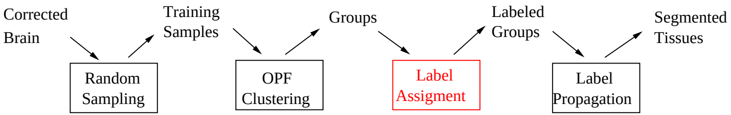
We use four 3D CSMs for brain segmentation.

Brain tissue segmentation



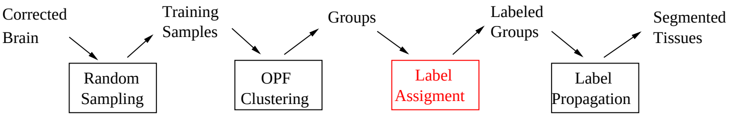
In each step of voxel clustering.

Brain tissue segmentation



In each step of voxel clustering.

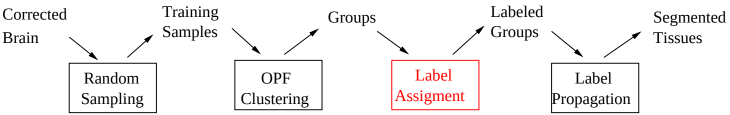
- Let $\mathcal{Z}$ be a set of brain voxels from two classes (either CSF/GM+WM or GM/WM).



In each step of voxel clustering.

- Let $\mathcal{Z}$ be a set of brain voxels from two classes (either CSF/GM+WM or GM/WM).
- A feature vector $\vec{v}(t)$ is assigned to every voxel $t \in \mathcal{Z}$ and $d(s,t) = \|\vec{v}(t) - \vec{v}(s)\|$.

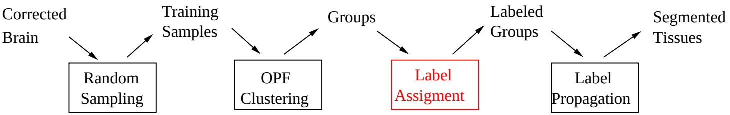
Brain tissue segmentation



In each step of voxel clustering.

- Let $\mathcal{Z}$ be a set of brain voxels from two classes (either CSF/GM+WM or GM/WM).
- A feature vector $\vec{v}(t)$ is assigned to every voxel $t \in \mathcal{Z}$ and $d(s,t) = \|\vec{v}(t) - \vec{v}(s)\|$.
- A small training set $\mathcal{T} \subset \mathcal{Z}$ is obtained by random sampling.

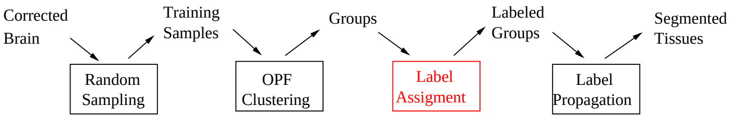
Brain tissue segmentation



In each step of voxel clustering.

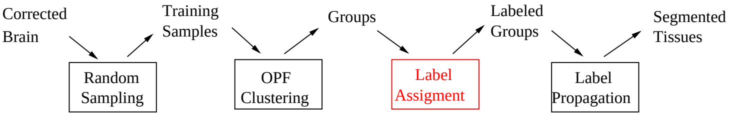
- Let $\mathcal{Z}$ be a set of brain voxels from two classes (either CSF/GM+WM or GM/WM).
- A feature vector $\vec{v}(t)$ is assigned to every voxel $t \in \mathcal{Z}$ and $d(s,t) = \|\vec{v}(t) - \vec{v}(s)\|$.
- A small training set $\mathcal{T} \subset \mathcal{Z}$ is obtained by random sampling.
- The OPF clustering can find groups of voxels in $\mathcal{T}$, mostly from a same class.

Brain tissue segmentation

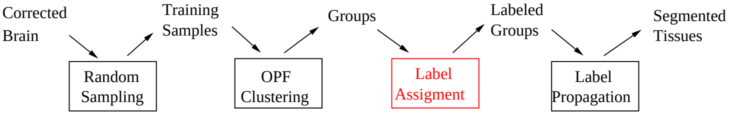


- Class labels are assigned to each group and propagated to the remaining voxels in $\mathcal{Z}$.

Brain tissue segmentation

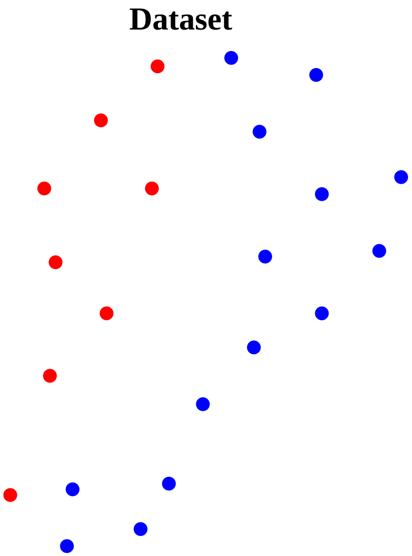


- Class labels are assigned to each group and propagated to the remaining voxels in $\mathcal{Z}$.
- For MRT1-images, label assignment is done from the darkest to the brightest cluster until the size proportion p between the classes is the closest to a previously estimated value p_T, which is obtained by automatic thresholding.

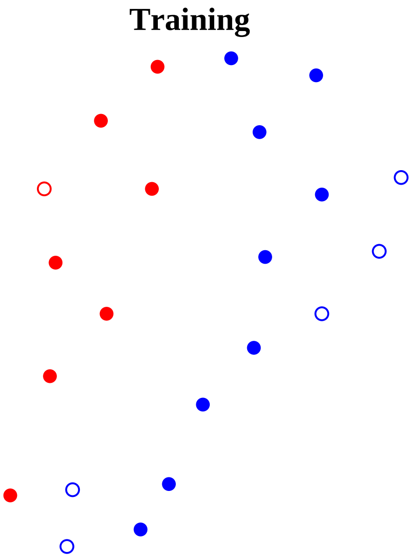


- Class labels are assigned to each group and propagated to the remaining voxels in $\mathcal{Z}$.
- For MRT1-images, label assignment is done from the darkest to the brightest cluster until the size proportion p between the classes is the closest to a previously estimated value p_T, which is obtained by automatic thresholding.

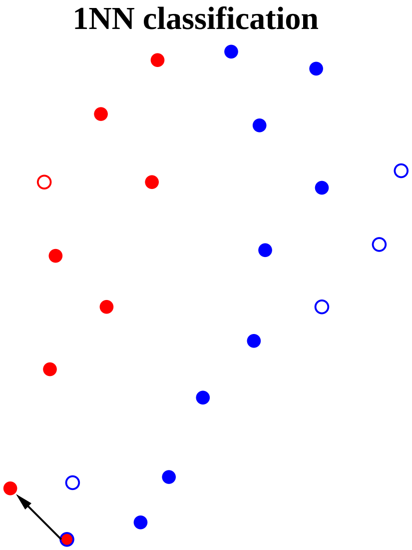
The entire process takes less than 2 minutes in a Core i7 PC, being about 1 minute for brain segmentation using 4 CSMs, and less than 1 minute for bias correction and tissue segmentation.



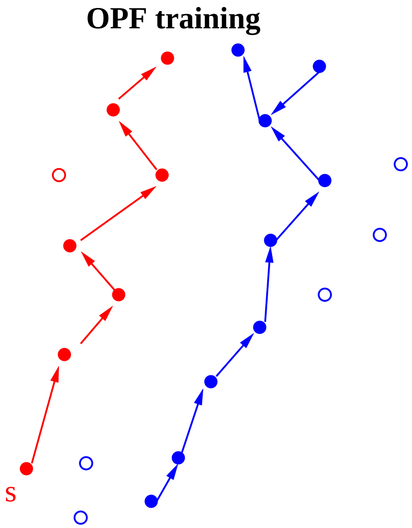
- Consider samples from two classes of a dataset.



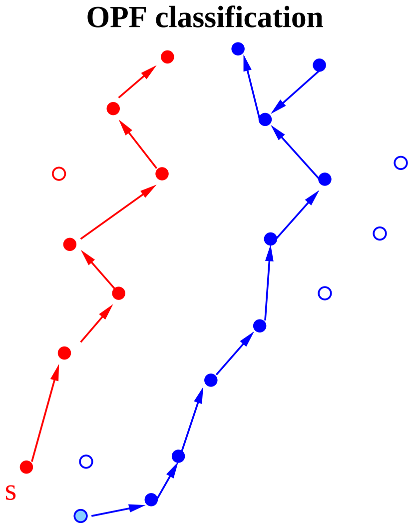
- Consider samples from two classes of a dataset.
- A training set (filled bullets) may not represent data distribution.



- Consider samples from two classes of a dataset.
- A training set (filled bullets) may not represent data distribution.
- Classification by nearest neighbor fails, when training samples are close to test samples (empty bullets) from other classes.



- We can create an optimum-path forest, where $V(s)$ is penalized when s is not closely connected to its class.



- We can create an optimum-path forest, where $V(s)$ is penalized when s is not closely connected to its class.
- $V(s)$ can then be used to reduce the power of s to classify new samples.

- We interpret $(\mathcal{T}, \mathcal{A})$ as a complete graph with undirected arcs between training samples.

Supervised learning

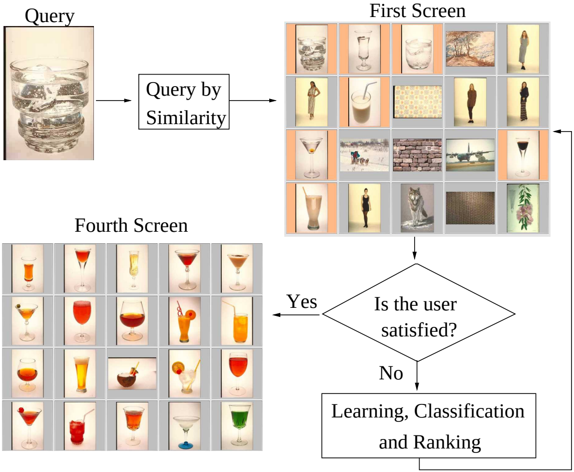
- We interpret $(\mathcal{T}, \mathcal{A})$ as a complete graph with undirected arcs between training samples.
- The prototypes are estimated as all samples that share an arc between distinct classes in a minimum-spanning tree of $(\mathcal{T}, \mathcal{A})$.

Supervised learning

- We interpret $(\mathcal{T}, \mathcal{A})$ as a complete graph with undirected arcs between training samples.
- The prototypes are estimated as all samples that share an arc between distinct classes in a minimum-spanning tree of $(\mathcal{T}, \mathcal{A})$.
- For a given set $\mathcal{S} \subset \mathcal{T}$ of prototypes from all classes, the connectivity map $V(t)$ is minimized for

$$
\begin{aligned}
f_{\max}(\langle t \rangle) &= \begin{cases} 0 & \text{if } t \in \mathcal{S} \\ +\infty & \text{otherwise} \end{cases} \\
f_{\max}(\pi_s \cdot \langle s, t \rangle) &= \max\{f_{\max}(\pi_s), d^*(s,t)\}
\end{aligned}
$$

where $d^*(s,t)$ is the distance between s and t.



The relevant and irrelevant images are the nodes of $(\mathcal{T}, \mathcal{A})$, whose arcs are weighted by a color descriptor [15].

In each iteration of relevance feedback.

- The training set $\mathcal{T}$ increases with new relevant (irrelevant) images as indicated by the user.

In each iteration of relevance feedback.

- The training set $\mathcal{T}$ increases with new relevant (irrelevant) images as indicated by the user.
- An OPF classifier is reprojected and used to select relevant candidates from the image database $\mathcal{Z}$.

Content-based image retrieval

In each iteration of relevance feedback.

- The training set $\mathcal{T}$ increases with new relevant (irrelevant) images as indicated by the user.
- An OPF classifier is reprojected and used to select relevant candidates from the image database $\mathcal{Z}$.
- The relevant candidates are ordered based on their average distances to the relevant prototypes.

Show software.

Dealing with large training sets

- When $\mathcal{T}$ is a large training set, our problem becomes the identification of the most representative samples to constitute a reduced training set $\mathcal{T}^* \subset \mathcal{T}$.

Dealing with large training sets

- When $\mathcal{T}$ is a large training set, our problem becomes the identification of the most representative samples to constitute a reduced training set $\mathcal{T}^* \subset \mathcal{T}$.
- We have fixed the size of $\mathcal{T}^*$ and replaced classification errors in $\mathcal{T} \backslash \mathcal{T}^*$ by non-representative samples in $\mathcal{T}^*$ [1].

Dealing with large training sets

- When $\mathcal{T}$ is a large training set, our problem becomes the identification of the most representative samples to constitute a reduced training set $\mathcal{T}^* \subset \mathcal{T}$.
- We have fixed the size of $\mathcal{T}^*$ and replaced classification errors in $\mathcal{T} \backslash \mathcal{T}^*$ by non-representative samples in $\mathcal{T}^*$ [1].
- We have recently devised a method that learns a minimum size of $\mathcal{T}^*$ in order to obtain a desired classification accuracy on $\mathcal{T} \backslash \mathcal{T}^*$.

Conclusions and open problems

- OPF classifiers can be designed by choice of a method for prototype estimation, adjacency relation and connectivity function.

Conclusions and open problems

- OPF classifiers can be designed by choice of a method for prototype estimation, adjacency relation and connectivity function.
- Successful examples are brain tissue segmentation using voxel clustering and CBIR using active learning and image classification.

Conclusions and open problems

- OPF classifiers can be designed by choice of a method for prototype estimation, adjacency relation and connectivity function.
- Successful examples are brain tissue segmentation using voxel clustering and CBIR using active learning and image classification.
- Open problems include new learning algorithms (supervised, unsupervised and semi-supervised), especially to reduce large training sets, and new methods and applications based on the OPF methodology.

Conclusions and open problems

- OPF classifiers can be designed by choice of a method for prototype estimation, adjacency relation and connectivity function.
- Successful examples are brain tissue segmentation using voxel clustering and CBIR using active learning and image classification.
- Open problems include new learning algorithms (supervised, unsupervised and semi-supervised), especially to reduce large training sets, and new methods and applications based on the OPF methodology.
- The C source code is available in `www.ic.unicamp.br/~afalcao/libopf`.

Acknowledgments

- Special acknowledgments to

Acknowledgments

- Special acknowledgments to
  - João Paulo Papa (UNESP) and Leonardo Rocha (Schlumberger).

Acknowledgments

- Special acknowledgments to
  - João Paulo Papa (UNESP) and Leonardo Rocha (Schlumberger).
  - Paulo Miranda (UNICAMP), Fábio Cappabianco (UNIFESP), and Jay Udupa (UPENN).

Acknowledgments

- Special acknowledgments to
  - João Paulo Papa (UNESP) and Leonardo Rocha (Schlumberger).
  - Paulo Miranda (UNICAMP), Fábio Cappabianco (UNIFESP), and Jay Udupa (UPENN).
  - André Silva (UNICAMP) and Léo Pini Magalhães (UNICAMP).

Acknowledgments

- Special acknowledgments to
  - João Paulo Papa (UNESP) and Leonardo Rocha (Schlumberger).
  - Paulo Miranda (UNICAMP), Fábio Cappabianco (UNIFESP), and Jay Udupa (UPENN).
  - André Silva (UNICAMP) and Léo Pini Magalhães (UNICAMP).
- The organizing and program committees of CIARP 2010.

Acknowledgments

- Special acknowledgments to
  - João Paulo Papa (UNESP) and Leonardo Rocha (Schlumberger).
  - Paulo Miranda (UNICAMP), Fábio Cappabianco (UNIFESP), and Jay Udupa (UPENN).
  - André Silva (UNICAMP) and Léo Pini Magalhães (UNICAMP).
- The organizing and program committees of CIARP 2010.
- FAPESP, CNPq, and UNICAMP.

Acknowledgments

Thank you

[1] J.P. Papa, A.X. Falcão, and C.T.N. Suzuki.
Supervised pattern classification based on optimum-path forest.
Intl. Journal of Imaging Systems and Technology, 19(2):120–131, Jun 2009.

[2] L.M. Rocha, F.A.M. Cappabianco, and A.X. Falcão.
Data clustering as an optimum-path forest problem with applications in image analysis.
Intl. Journal of Imaging Systems and Technology, 19(2):50–68, Jun 2009.

[3] F. Cappabianco, A.X. Falcão, and L.M. Rocha.
Clustering by optimum path forest and its application to automatic GM/WM classification in MR-T1 images of the brain.
In *The Fifth IEEE Intl. Symp. on Biomedical Imaging (ISBI)*, pages 428–431, Paris, France, 2008.

[4] Fábio A.M. Cappabianco, A.X. Falcão, Clarissa L. Yasuda, and J. K. Udupa.
MR-Image Segmentation of Brain Tissues based on Bias Correction and Optimum-Path Forest Clustering.
Technical Report IC-10-07, Institute of Computing, University of Campinas, March 2010.

[5] A.T. Silva, A.X. Falcão, and L.P. Magalhães.
A new CBIR approach based on relevance feedback and optimum-path forest classification.
Journal of WSCG, 18(1-3):73–80, 2010.

[6] T.V. Spina, J.A. Montoya-Zegarra, A.X. Falcão, and P.A.V. Miranda.
Fast interactive segmentation of natural images using the image foresting transform.
In *Proc. of the 16th Intl. Conf. on Digital Signal Processing*, Santorini, Greece, 2009. IEEE.

[7] A.I. Iliev, M.S. Scordilis, J.P. Papa, and A.X. Falcão.
Spoken emotion recognition through optimum-path forest classification using glottal features.
Computer Speech and Language, 24(3):445–460, Jul 2010.

[8] G. Chiachia, A.N. Marana, J.P. Papa, and A.X. Falcão.
Infrared face recogniton by optimum-path forest.
In *Proc. of the 16th Intl. Conf. on Systems, Signals, and Image Processing*, Chalkida, Greece, 2009. IEEE.

[9] R.J. Pisani, J.P. Papa, C.R.L. Zimback, A.X. Falcão, and A.P. Barbosa.
Land use classification using optimum-path forest.
In *XIV Brazilian Remote Sensing Symposium (SBSR)*, pages 7063–7070, Natal-RN, 2009. MCT-INPE.

[10] J.P. Papa, A.X. Falcão, A.M. Levada, D. Corrêa, D. Salvadeo, and N.D.A. Mascarenhas.

Fast and accurate holistic face recognition using optimum-path forest.
In *Proc. of the 16th Intl. Conf. on Digital Signal Processing*, Santorini, Greece, 2009. IEEE.

[11] I. V. Guilherme, A. N. Marana, J. P. Papa, G. Chiachia, L. C. S. Afonso, K. Miura, M. V. D. Ferreira, and F. Torres.
Petroleum well drilling monitoring through cutting image analysis and artificial intelligence techniques.
Engineering Applications of Artificial Intelligence, 2010.

[12] C.O. Ramos, A.N. Souza, J.P. Papa, and A.X. Falcão.
A new approach for non-technical losses detection based on optimum-path forest.
IEEE Trans. on Power Systems, PP(99):1–9, July 2010.

[13] J. P. Papa, F. A. M. Cappabianco, and A. X. Falcão.
Optimizing optimum-path forest classification for huge datasets.
In *Proc. of the 20th International Conference on Pattern Recognition*, pages 4162–4165, Istanbul, Turkey, Aug 2010.

[14] J.P. Papa, A.A. Spadotto, A.X. Falcão, R. C. Guido, and J. C. Pereira.
Robust biomedical signal recognition though optimum-path forest.
Artificial Intelligence in Medicine, 2010.
to appear.

[15] R. O. Stehling, M. A. Nascimento, and A. X. Falcao.
A compact and efficient image retrieval approach based on border/interior pixel classification.
In *CIKM '02: Proceedings of the eleventh international conference on Information and knowledge management*, pages 102–109, New York, NY, USA, 2002. ACM.

[16] F.A. Andaló, P.A.V. Miranda, R. da S. Torres, and A.X.Falcão.
Shape feature extraction and description based on tensor scale.
Pattern Recognition, 43(1):26–36, Jan 2010.

[17] J.A. Montoya-Zegarra, J.P. Papa, N.J. Leite, R.S. Torres, and A.X. Falcão.
Learning how to extract rotation-invariant and scale-invariant features from texture images.
EURASIP Journal on Advances in Signal, 2008.

[18] R.S. Torres, A.X. Falcão, M.A. Gonçalves, J.P. Papa, B. Zhang, W. Fan, and E.A. Fox.
A genetic programming framework for content-based image retrieval.
Pattern Recognition, 42:217–312, Feb 2009.

[19] A.X. Falcão, J. Stolfi, and R.A. Lotufo.
The image foresting transform: Theory, algorithms, and applications.
IEEE Trans. on Pattern Analysis and Machine Intelligence, 26(1):19–29, 2004.

[20] P.A.V. Miranda, A.X. Falcão, and J.K. Udupa.
CLOUDS: A model for synergistic image segmentation.
In *Proc. of the ISBI*, pages 209–212, Paris, France, 2008.

[21] P.A.V. Miranda, A.X. Falcão, and J.K. Udupa.
Cloud bank: A multiple clouds model and its use in MR brain image segmentation.
In *Proc. of the ISBI*, pages 506–509, Boston, MA, 2009.

[22] P.A.V. Miranda, A.X. Falcão, and J.K. Udupa.
Cloud models: Their construction and employment in automatic MRI segmentation of the brain.
Technical Report IC-10-08, Institute of Computing, University of Campinas, Mar 2010.