Classwise image analysis in remote sensing

Sébastien Lefèvre and Laetitia Chapel

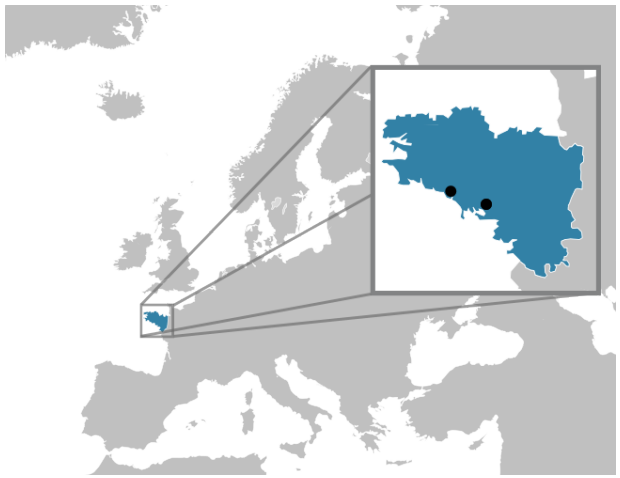
IRISA – http://www.irisa.fr
University of South Brittany – http://www.univ-ubs.fr

EPFL — February 12th, 2013

University of South Brittany

A recent and small university

- Officially created in 1995 in Vannes and Lorient (faculties established in 70s)
- 8500+ students, 250+ faculty members (+150 lecturers)
- 4th French university for professional insertion



IRISA

The main research institute in computer science in Brittany

- Officially created in 1975 in Rennes (established in 1970)
- Joint research institute between CNRS, INRIA, Universities (BS, Rennes), Graduate Schools (INSA, ENS, Telecom)
- 250+ permanent researchers, 200 PhD students (within UBS in Vannes: 20 faculty members, 20 PhD students)
- 7 research departments: Large-scale Systems; Networks, Telecommunications and Services; Architecture; Language and Software Engineering; Digital Signals and Images, Robotics; Media and Interactions; Data and Knowledge Management

OBELIX

Environment Observation through Complex Images

- machine learning and data mining
- image analysis and processing
- links between physical models and observation data
- visual analytics
- in the context of remote sensing of environment

The team

- 5 faculty members at UBS (Vannes)
- 1 senior researcher at OSUR/CNRS (Rennes)
- (still a few) young researchers...
- **AND open positions (Ass.Prof, Postdoc, PhD)**

Outline

Classwise image analysis in remote sensing

1. Introduction
2. Classification
3. Feature extraction
4. Segmentation
5. Conclusion

Outline

Classwise image analysis in remote sensing

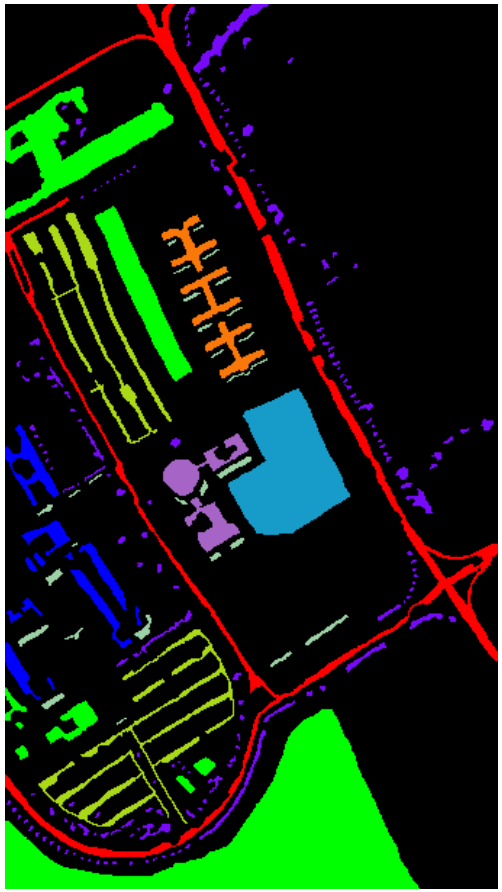
1. Introduction
2. Classification
3. Feature extraction
4. Segmentation
5. Conclusion

Outline

Classwise image analysis in remote sensing

1. Introduction
2. Classification
3. Feature extraction
4. Segmentation
5. Conclusion

color composite

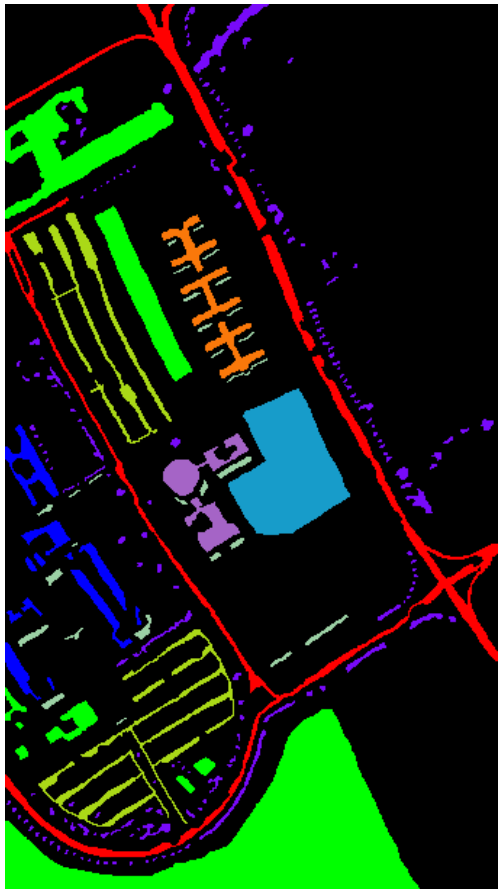


reference data

Hyperspectral images

- hundreds of spectral bands, each being a grayscale image
- several classes of interest
- illustration with Pavia University dataset
  - 102 spectral bands
  - 1096×715 pixels
  - 9 classes of interest

color composite



reference data

Classification

- Every pixel is described by a vector of attributes
  ⇒ huge amount of high dimensional data
- Several classes of interest
  ⇒ training of several classifiers

Issues

1. Processing a huge amount of high dimensional data leads to significant time and memory requirements
2. Most classifiers suffer the dimensionality curse: the classification accuracy decreases with the dimension of the data when the number of available pixels is fixed.

SVM: state of the art classification technique in this context

- alleviate the dimensionality curse
- generally outperform traditional classification techniques
- but
  - they learn a boundary between classes $\Rightarrow$ they have to be retrained when a new class is added
  - their performances are not optimal when the number of training examples is low

Issues

1. Processing a huge amount of high dimensional data leads to significant time and memory requirements
2. Most classifiers suffer the dimensionality curse: the classification accuracy decreases with the dimension of the data when the number of available pixels is fixed.

SVM: state of the art classification technique in this context

- alleviate the dimensionality curse
- generally outperform traditional classification techniques
- but
  - they learn a boundary between classes $\Rightarrow$ they have to be retrained when a new class is added
  - their performances are not optimal when the number of training examples is low

PerTurbo algorithm

- Idea from computer graphics: sampling point clouds, keeping the most informative points
- Need for metric to quantify the modification of the definition of the object surface induced by a new sample
- The spectrum of $\Delta(\mathcal{M})$ can be approximated by

$$K(x_i, x_j) = K_{ij} = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$$

- The perturbation of the spectrum of $\mathbf{K}$ is used to evaluate the interest of a sample

$$1 - K_{\widetilde{\mathbf{x}}}^T \mathbf{K}(\mathcal{S})^{-1} K_{\widetilde{\mathbf{x}}}$$

with $K_{\widetilde{\mathbf{x}}} = K(\mathcal{S}, \widetilde{\mathbf{x}})$ whose ith term is $K(\mathbf{x}^i, \widetilde{\mathbf{x}})$

Classical classification problem

- Training set $\mathcal{S} = \left\{(\mathbf{x}^1, y_1), ..., (\mathbf{x}^N, y_N)\right\} \in \mathbb{R}^d \times \{u_1, ..., u_L\}$
- $\mathcal{S}_\ell$ is the set of all the training examples with label u_ℓ belonging to $\mathcal{S}$
- **Aim:** build a predictive function that associates to each pixel $\mathbf{x^n}$ its corresponding class y_n

Classwise manifold learning

- For each class ℓ, a dedicated manifold $\mathcal{M}_\ell$ is considered
- Each manifold are learned independently and represented by a particular matrix $\mathbf{K}_\ell$
- $\mathbf{K}_\ell$ is also the classical Gram matrix in the Gaussian RKHS

Perturbo algorithm

Training step

$\forall$ class ℓ, $\mathbf{K}(\mathcal{S}_\ell)^{-1}$ is computed.

Testing step

For each test sample $\widetilde{\mathbf{x}}$

1. Compute the perturbation of $\mathcal{M}_\ell$ induced by $\widetilde{\mathbf{x}}$:

$$\tau(\widetilde{\mathbf{x}}, \mathcal{M}_\ell) = 1 - K_{\widetilde{\mathbf{x}}}^T \mathbf{K}(\mathcal{S}_\ell)^{-1} K_{\widetilde{\mathbf{x}}}$$

with $K_{\widetilde{\mathbf{x}}} = K(\mathcal{S}_\ell, \widetilde{\mathbf{x}})$ whose ith term is $K(\mathbf{x}^i, \widetilde{\mathbf{x}})$

2. Predict the class with the least induced perturbation, which reads as:

$$\arg\min_\ell \tau(\widetilde{\mathbf{x}}, \mathcal{M}_\ell)$$

Perturbo algorithm

Training step

$\forall$ class ℓ, $\mathbf{K}(\mathcal{S}_\ell)^{-1}$ is computed.

Testing step

For each test sample $\widetilde{\mathbf{x}}$

1. Compute the perturbation of $\mathcal{M}_\ell$ induced by $\widetilde{\mathbf{x}}$:

$$\tau(\widetilde{\mathbf{x}}, \mathcal{M}_\ell) = 1 - K_{\widetilde{\mathbf{x}}}^T \mathbf{K}(\mathcal{S}_\ell)^{-1} K_{\widetilde{\mathbf{x}}}$$

   with $K_{\widetilde{\mathbf{x}}} = K(\mathcal{S}_\ell, \widetilde{\mathbf{x}})$ whose ith term is $K(\mathbf{x}^i, \widetilde{\mathbf{x}})$

2. Predict the class with the least induced perturbation, which reads as:

$$\arg\min_\ell \tau(\widetilde{\mathbf{x}}, \mathcal{M}_\ell)$$

Perturbo algorithm

Training step

$\forall$ class ℓ, $\mathbf{K}(\mathcal{S}_\ell)^{-1}$ is computed.

Testing step

For each test sample $\widetilde{\mathbf{x}}$

1. Compute the perturbation of $\mathcal{M}_\ell$ induced by $\widetilde{\mathbf{x}}$:

$$\tau(\widetilde{\mathbf{x}}, \mathcal{M}_\ell) = 1 - K_{\widetilde{\mathbf{x}}}^T \mathbf{K}(\mathcal{S}_\ell)^{-1} K_{\widetilde{\mathbf{x}}}$$

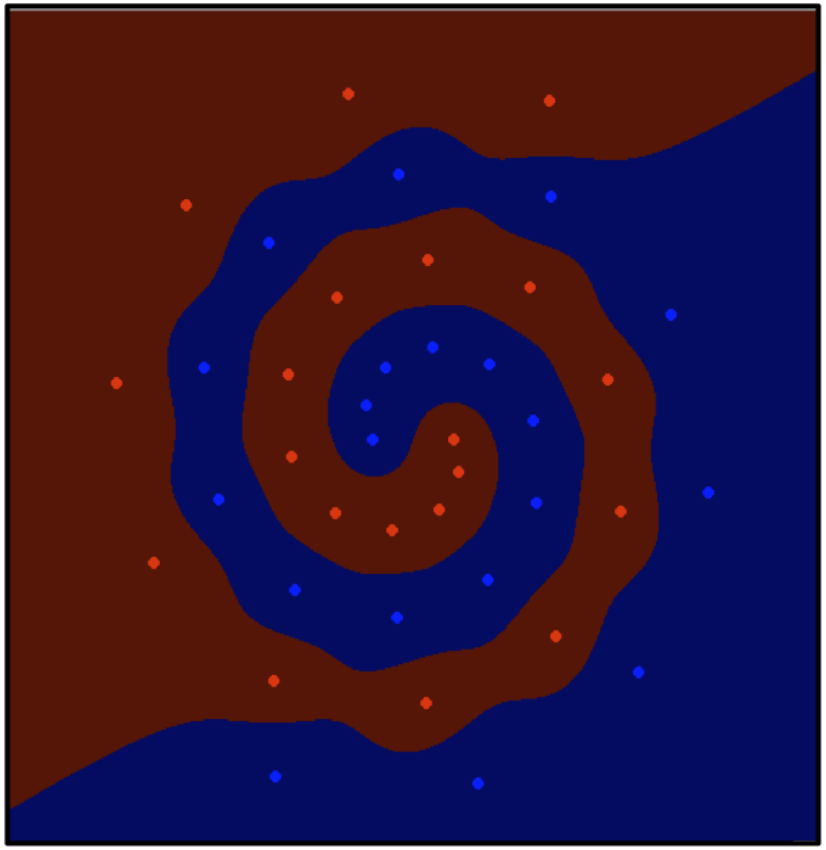
   with $K_{\widetilde{\mathbf{x}}} = K(\mathcal{S}_\ell, \widetilde{\mathbf{x}})$ whose ith term is $K(\mathbf{x}^i, \widetilde{\mathbf{x}})$

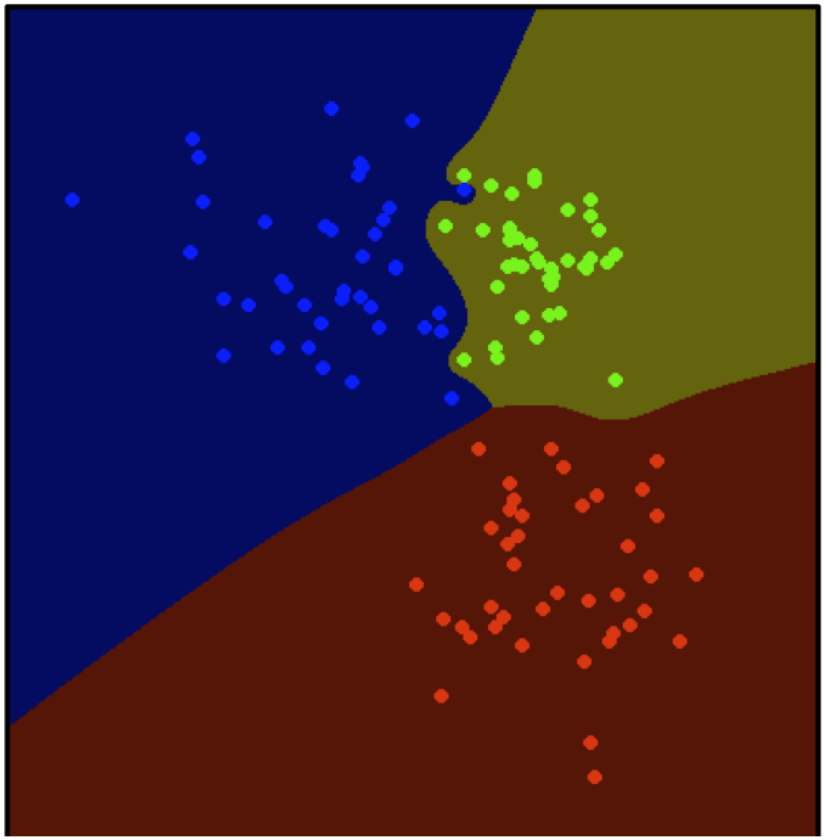
2. Predict the class with the least induced perturbation, which reads as:

$$\arg\min_\ell \tau(\widetilde{\mathbf{x}}, \mathcal{M}_\ell)$$

Perturbo algorithm

In the seminal paper, it shows similar performances in comparison with SVM on several toy examples





Testing step

PerTurbo's main characteristics

- provides a class-wise classification
- non-parametric method: no need for explicit formulation of the decision function like in SVM.
- very simple, easy to implement and involves few parameters to tune

Regularization

- works as long as $\mathbf{K}(\mathcal{S}_\ell)$ is invertible
- Tikhonov regularization:

$$\widetilde{\mathbf{K}}(\mathcal{S}_\ell) = \mathbf{K}(\mathcal{S}_\ell) + \alpha_\ell \cdot I$$

PerTurbo's main characteristics

- provides a class-wise classification
- non-parametric method: no need for explicit formulation of the decision function like in SVM.
- very simple, easy to implement and involves few parameters to tune

Regularization

- works as long as $\mathbf{K}(\mathcal{S}_\ell)$ is invertible
- Tikhonov regularization:

$$\widetilde{\mathbf{K}}(\mathcal{S}_\ell) = \mathbf{K}(\mathcal{S}_\ell) + \alpha_\ell \cdot I$$

Datasets

3 hyperspectral scenes

- **Pavia University**: 102 spectral bands, 7456 pixels for training, 148197 for testing, 9 classes of interest
- **Pavia Centre**: 103 spectral bands, 3921 pixels for training, 42776 for testing, 9 classes of interest
- **Indian Pines**: 224 spectral bands, 695 pixels for training, 9554 for testing, 16 classes of interest

Parameter settings

- **SVM**: grid search for (σ, C), one-vs-all strategy
- **PerTurbo**: grid search for σ and (σ, α)
- each experiment is repeated 20 times for pines dataset

Classification accuracies

Pavia Centre

class	1	2	3	4	5	6	7	8	9	OA	AA
PerTurbo	100	100	100	100	100	100	100	100	100	100	100
SVM	99.9	99.9	99.9	100	100	99.9	98.8	99.8	99.9	99.4	99.8

$\Rightarrow$ similar performances

Pavia University

class	1	2	3	4	5	6	7	8	9	OA	AA
PerTurbo (i)	84.2	78.0	37.4	99.0	99.2	68.7	88.5	97.2	99.9	80.5	83.6
PerTurbo (t)	83.2	74.8	41.2	99.2	99.2	76.4	99.0	96.4	99.9	80.0	85.5
SVM	83.5	70.5	69.7	99.0	99.3	92.0	90.8	92.6	98.8	81.0	88.5

$\Rightarrow$ Accuracies are close, SVM outperform PerTurbo especially on 2 classes.

Classification accuracies

Pavia Centre

class	1	2	3	4	5	6	7	8	9	OA	AA
PerTurbo	100	100	100	100	100	100	100	100	100	100	100
SVM	99.9	99.9	99.9	100	100	99.9	98.8	99.8	99.9	99.4	99.8

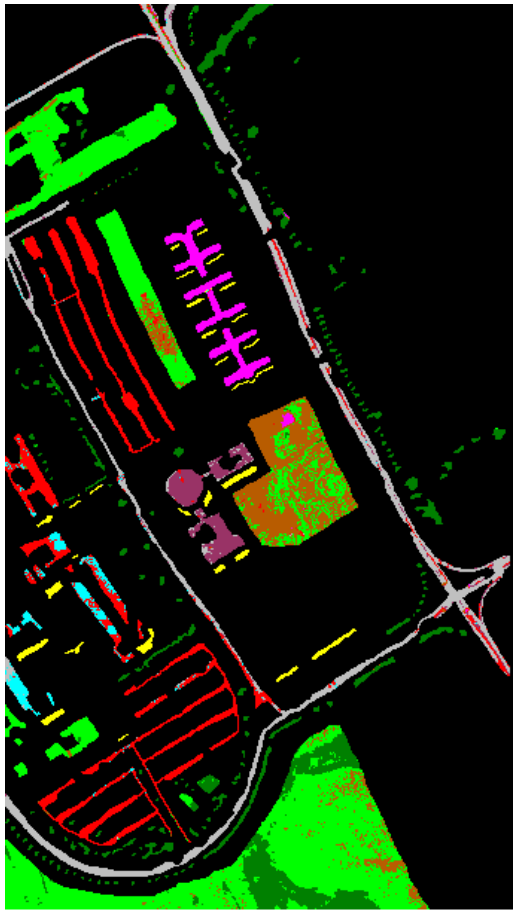
$\Rightarrow$ similar performances

Pavia University

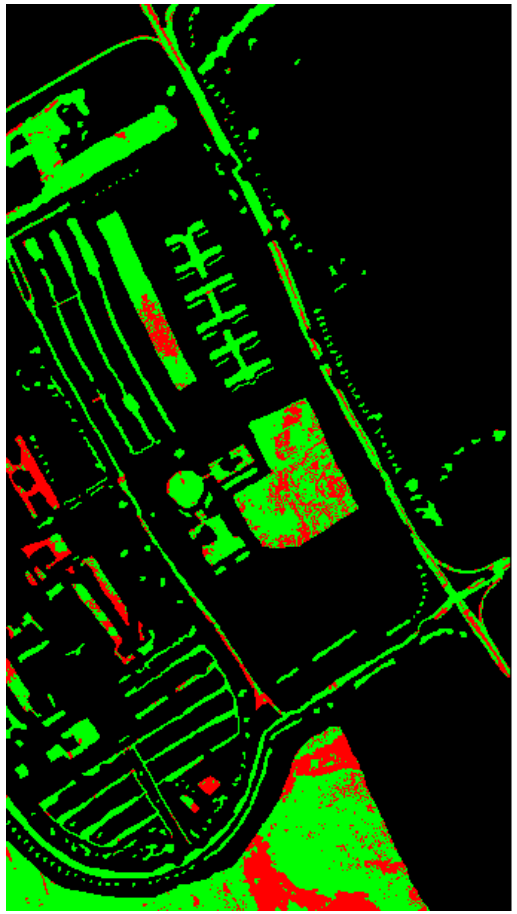
class	1	2	3	4	5	6	7	8	9	OA	AA
PerTurbo (i)	84.2	78.0	37.4	99.0	99.2	68.7	88.5	97.2	99.9	80.5	83.6
PerTurbo (t)	83.2	74.8	41.2	99.2	99.2	76.4	99.0	96.4	99.9	80.0	85.5
SVM	83.5	70.5	69.7	99.0	99.3	92.0	90.8	92.6	98.8	81.0	88.5

$\Rightarrow$ Accuracies are close, SVM outperform PerTurbo especially on 2 classes.

reference data



PerTurbo result



true/wrong classification

Classification accuracies

Indian pines

No	Name	# train	# test	PerTurbo (i)	PerTurbo (t)	SVM
1	Alfalfa	15	31	68.5 (23.4)	66.7 (24.4)	76.1 (8.8)
2	Corn-notill	50	1378	72.4 (24.9)	77.6 (19.8)	60.3 (4.5)
3	Corn-mintill	50	780	68.5 (21.1)	71.0 (20.1)	64.1 (5.2)
4	Corn	50	187	74.3 (24.4)	70.6 (23.8)	80.1 (5.8)
5	Grass-pasture	50	433	66.8 (22.4)	67.0 (23.6)	88.2 (2.9)
6	Grass-trees	50	680	72.0 (23.1)	77.2 (21.2)	93.5 (2.3)
7	Grass-pasture-mowed	15	13	68.5 (22.4)	71.5 (20.0)	91.9 (6.4)
8	Hay-windrowed	50	428	74.0 (22.1)	73.7 (20.5)	95.2 (1.8)
9	Oats	15	5	69.1 (21.7)	71.2 (21.0)	97.0 (7.3)
10	Soybean-notill	50	922	76.1 (19.3)	75.1 (21.5)	73.8 (4.8)
11	Soybean-mintill	50	2405	69.9 (19.7)	73.2 (20.0)	58.6 (3.9)
12	Soybean-clean	50	543	73.2 (22.8)	71.5 (22.1)	66.8 (4.5)
13	Wheat	50	155	70.4 (22.7)	67.5 (24.4)	97.8 (1.4)
14	Woods	50	1215	74.7 (19.7)	74.0 (21.6)	86.1 (3.3)
15	Buildings-Grass-Trees-Drives	50	336	70.1 (18.7)	72.2 (20.5)	67.9 (4.4)
16	Stone-Steel-Towers	50	43	71.4 (21.3)	74.6 (19.9)	95.8 (2.3)
			OA	68.85 (5.6)	68.60 (5.4)	71.9 (1.2)
			AA	71.24	72.15	80.8

$\Rightarrow$ SVM outperform PerTurbo, High variability for PerTurbo (to be investigated)

Limitations of pixelwise classification

Spectral signatures of pixels are of first importance but other features may be extracted:

- texture
- shape

Morphological profile (DMP) have achieved great success in remote sensing. They are based on series of morphological filterings.

Outline

1 Introduction

2 Classification

3 Feature extraction

4 Segmentation

5 Conclusion

Morphological series

Definition

Opening and closing series are defined by:

$$\Pi_\psi(f) = \{\Pi_{\psi_\lambda}(f) \mid \Pi_{\psi_\lambda}(f) = \psi_\lambda(f), \forall \lambda \in \{0, \ldots, n\}\}$$

with $\psi = \gamma$ ou $\psi = \phi$, $\psi_0(f) = f$ and increasing SE size λ.

Dual series

Series Π_γ and Π_ϕ can be combined:

$$\Pi(f) = \left\{ \Pi_c(f) \mid \Pi_c(f) = \begin{cases} \Pi_{\phi_{-c}}(f), \forall c \in \{-n, \ldots, -1\} \\ \Pi_{\gamma_c}(f), \forall c \in \{1, \ldots, n\} \end{cases} \right\}$$

Morphological series

Definition

Opening and closing series are defined by:

$$\Pi_\psi(f) = \{\Pi_{\psi_\lambda}(f) \,|\, \Pi_{\psi_\lambda}(f) = \psi_\lambda(f), \forall \lambda \in \{0, \ldots, n\}\}$$

with $\psi = \gamma$ ou $\psi = \phi$, $\psi_0(f) = f$ and increasing SE size λ.

Dual series

Series Π_γ and Π_ϕ can be combined:

$$\Pi(f) = \left\{ \Pi_c(f) \,|\, \Pi_c(f) = \begin{cases} \Pi_{\phi_{-c}}(f), \forall c \in \{-n, \ldots, -1\} \\ \Pi_{\gamma_c}(f), \forall c \in \{1, \ldots, n\} \end{cases} \right\}$$

Morphological series

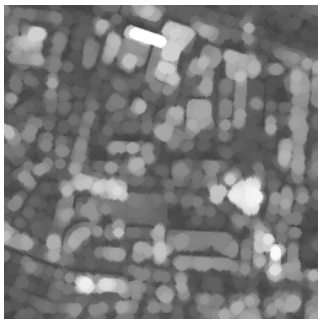
Image series Π_γ built from openings γ with a disc SE of increasing radius λ:

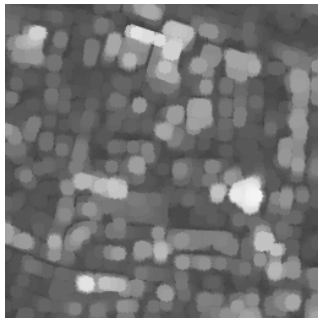
$\lambda = 0$

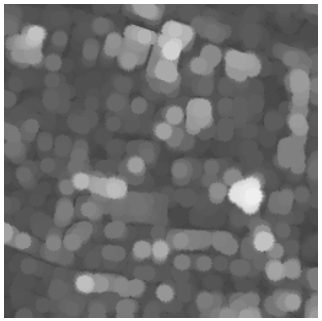
$\lambda = 1$

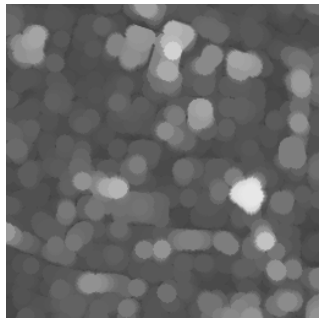
$\lambda = 2$

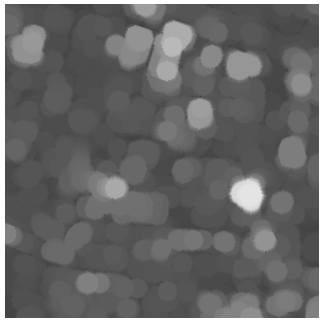
$\lambda = 3$

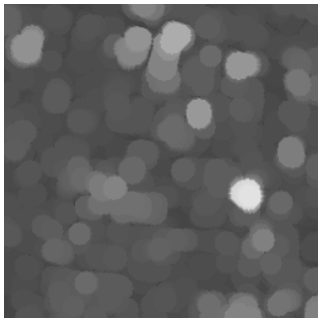

$\lambda = 4$


$\lambda = 5$


$\lambda = 6$

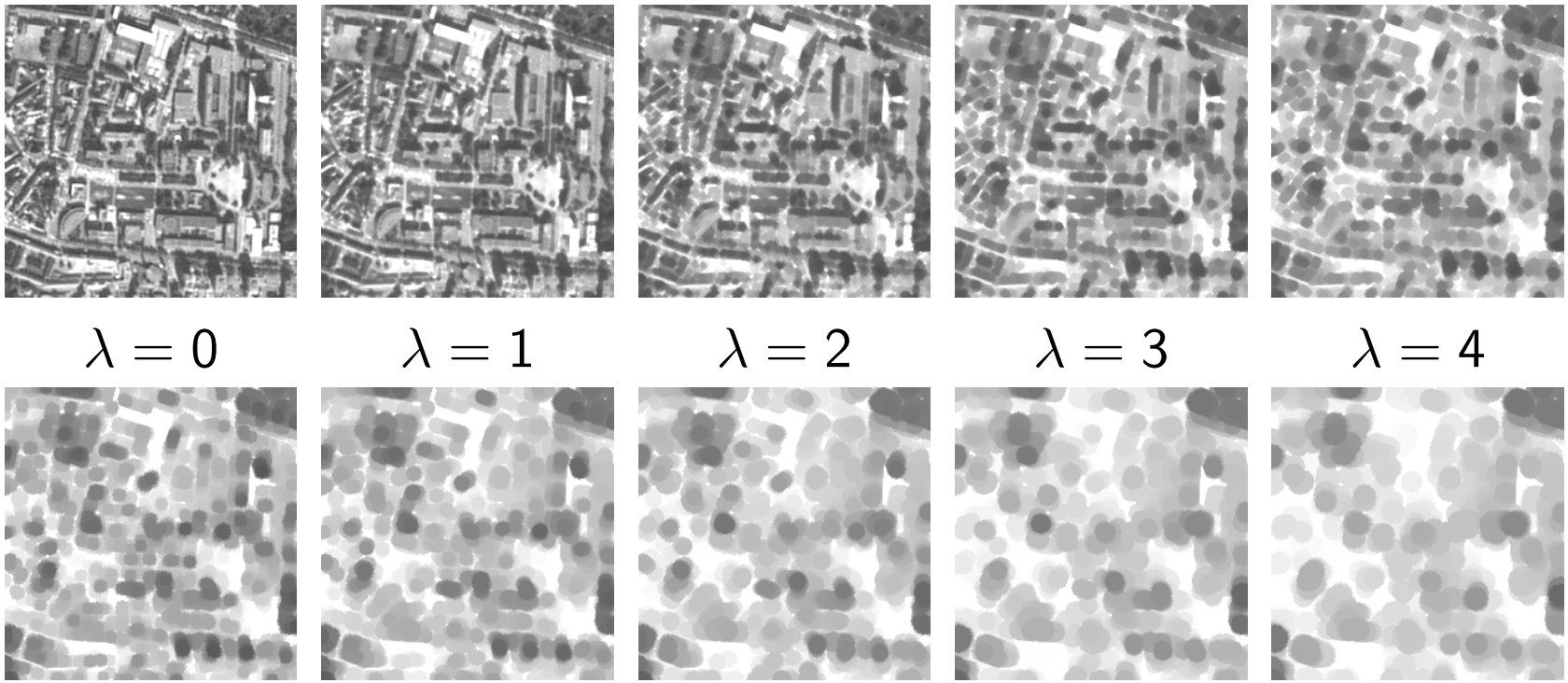

$\lambda = 7$


$\lambda = 8$


$\lambda = 9$

Morphological series

Image series Π_ϕ built from closings ϕ with a disc SE of increasing radius λ:



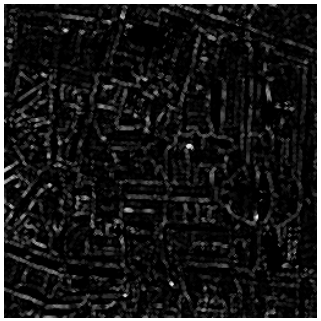
$\lambda = 0$ $\lambda = 1$ $\lambda = 2$ $\lambda = 3$ $\lambda = 4$

$\lambda = 5$ $\lambda = 6$ $\lambda = 7$ $\lambda = 8$ $\lambda = 9$

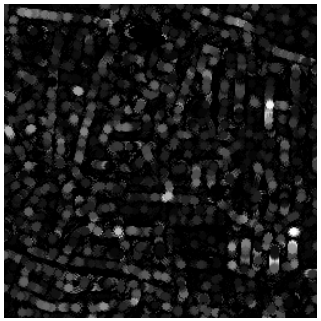
Differential series

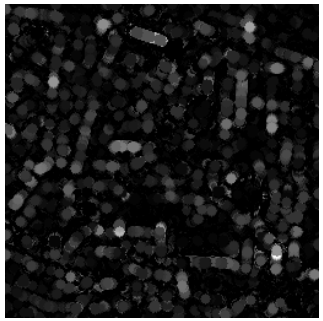
Most often we consider differential series Δ:

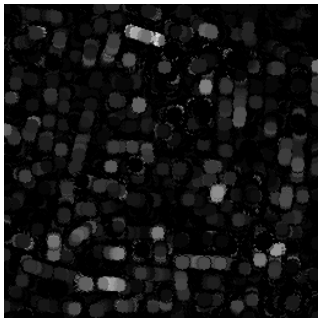
$$\Delta_\psi(f) = \left\{ \Delta_{\psi_\lambda}(f) \mid \Delta_{\psi_\lambda}(f) = \Pi_{\psi_\lambda}(f) - \Pi_{\psi_{\lambda-1}}(f) \right\}$$

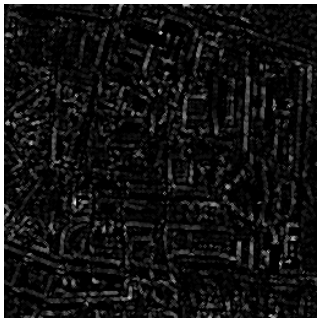

γ_2

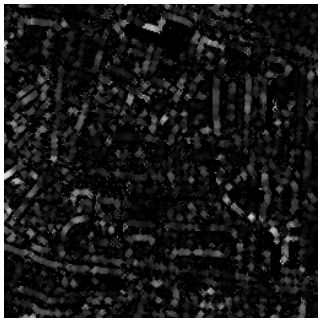
γ_3

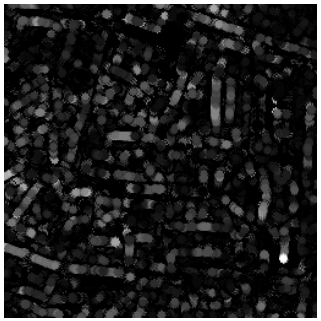

γ_4

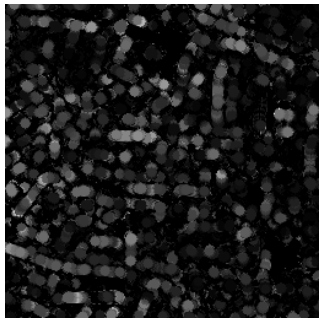

γ_5

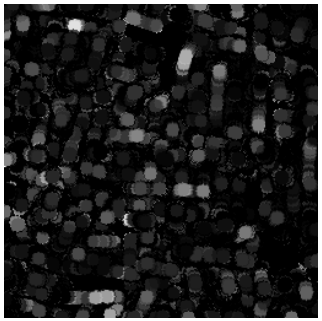

γ_6


ϕ_2


ϕ_3


ϕ_4


ϕ_5


ϕ_6

Reconstruction series

Reconstruction filters ψ allow to better preserve shapes

γ_2

γ_3

γ_4

γ_5

γ_6

ϕ_2

ϕ_3

ϕ_4

ϕ_5

ϕ_6

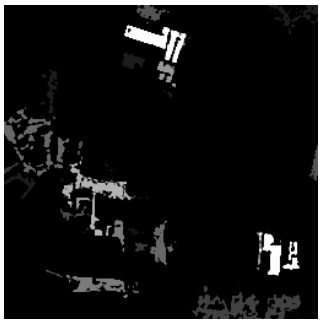
DMP: Differential Morphological Profiles

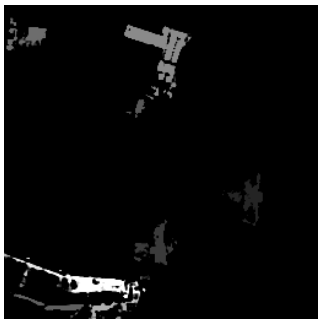
Introduced by Benediktsson and Pesaresi (TGRS 2001 & 2003) and defined in every pixel (x, y) from series Δ

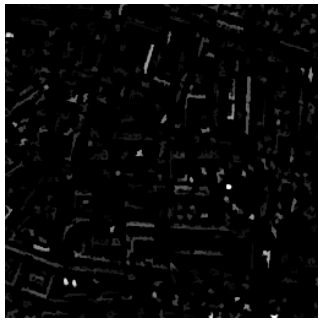
γ_2

γ_3

γ_4

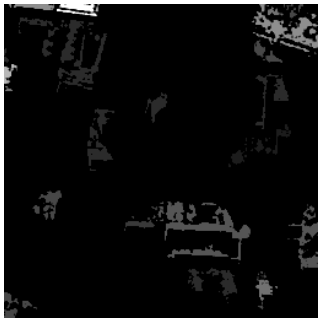

γ_5

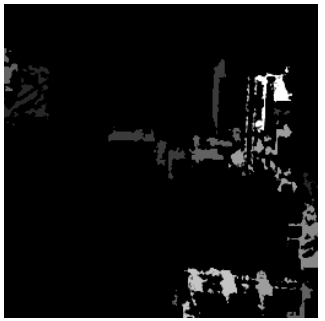

γ_6


ϕ_2

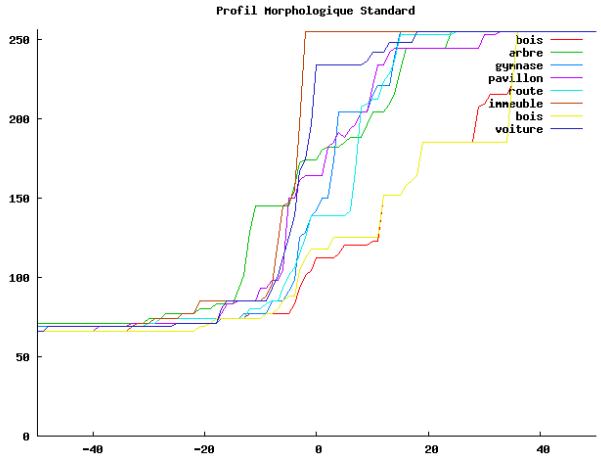
ϕ_3

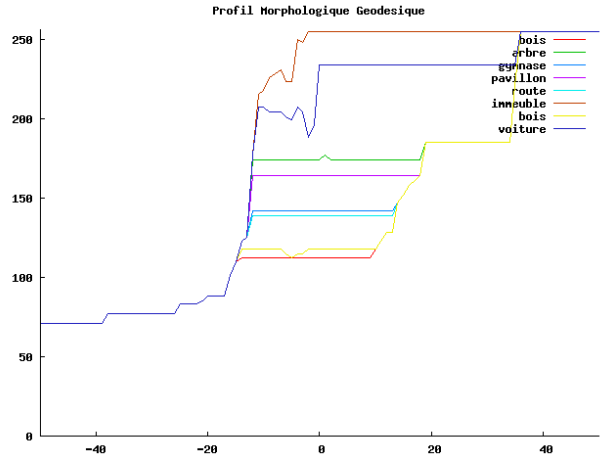
ϕ_4

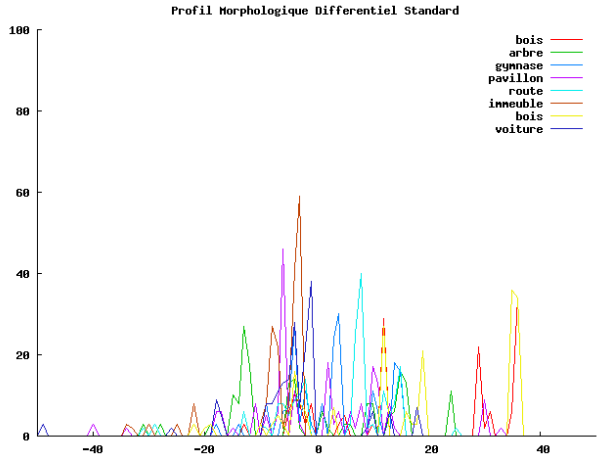

ϕ_5

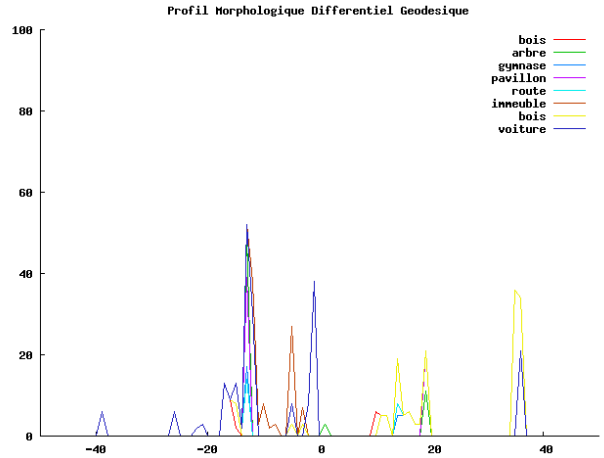

ϕ_6

Comparison of morphological profiles





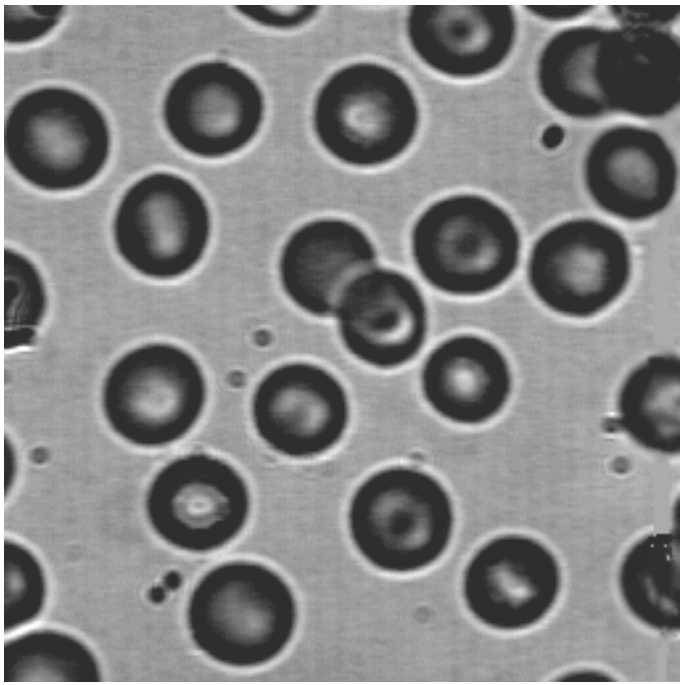




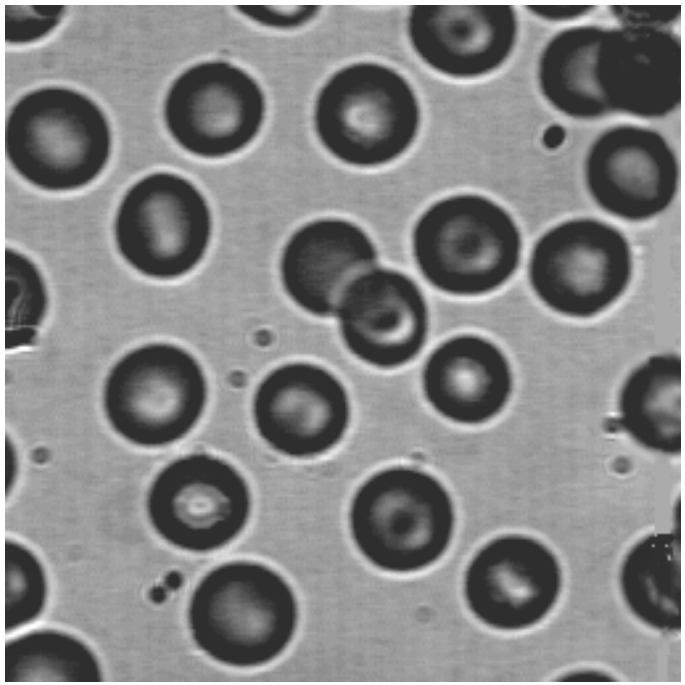
Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels



Gaussian smoothings

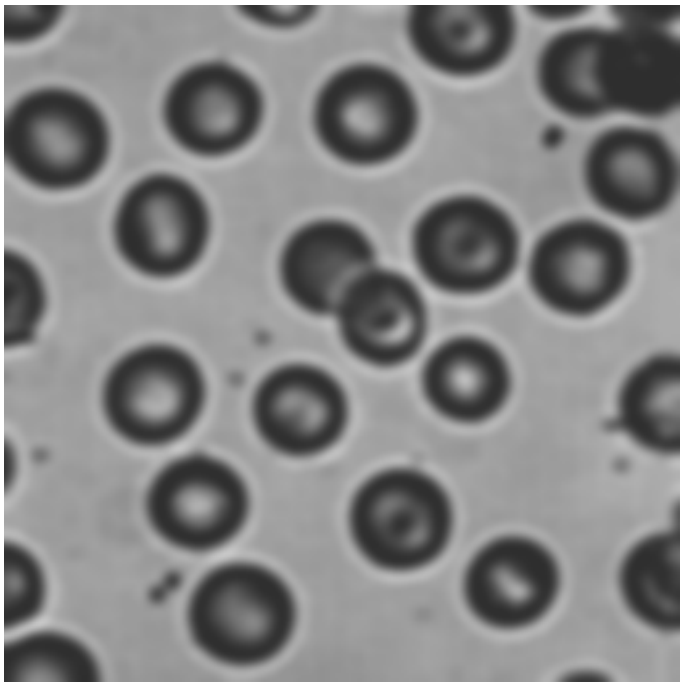


openings by reconstruction

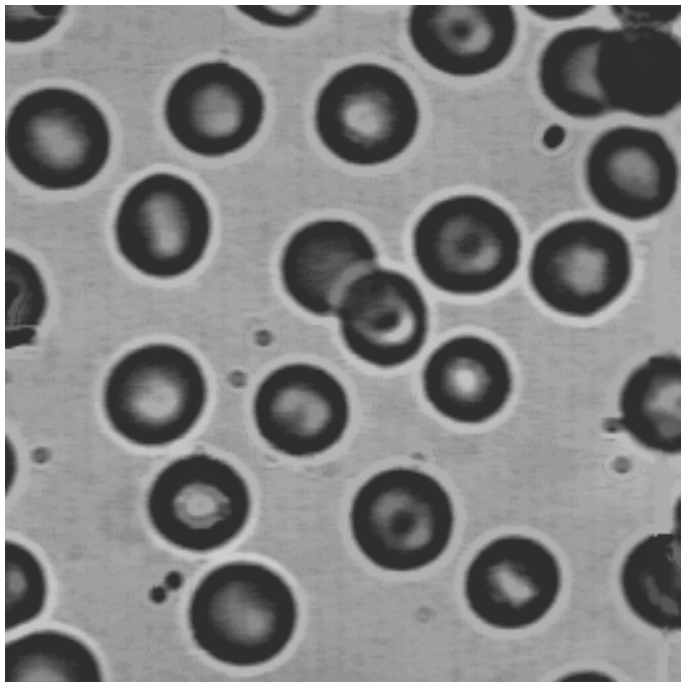
Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels



Gaussian smoothings

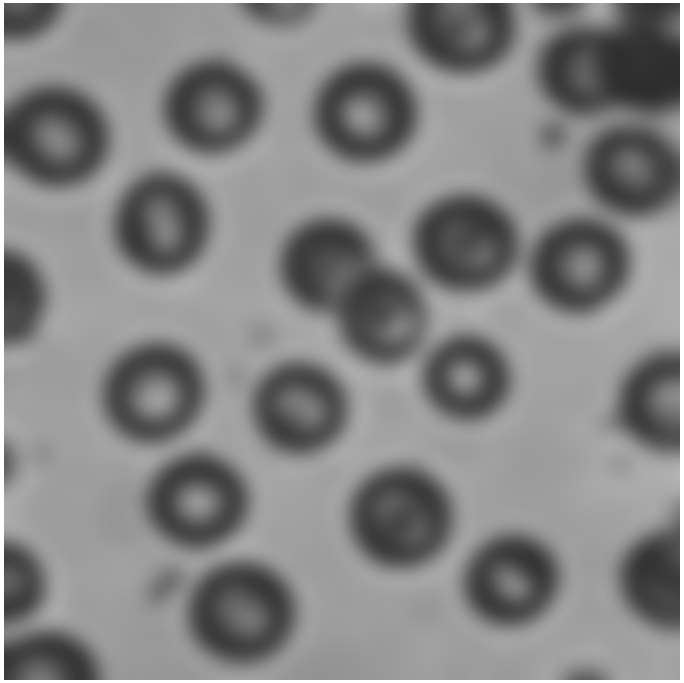


openings by reconstruction

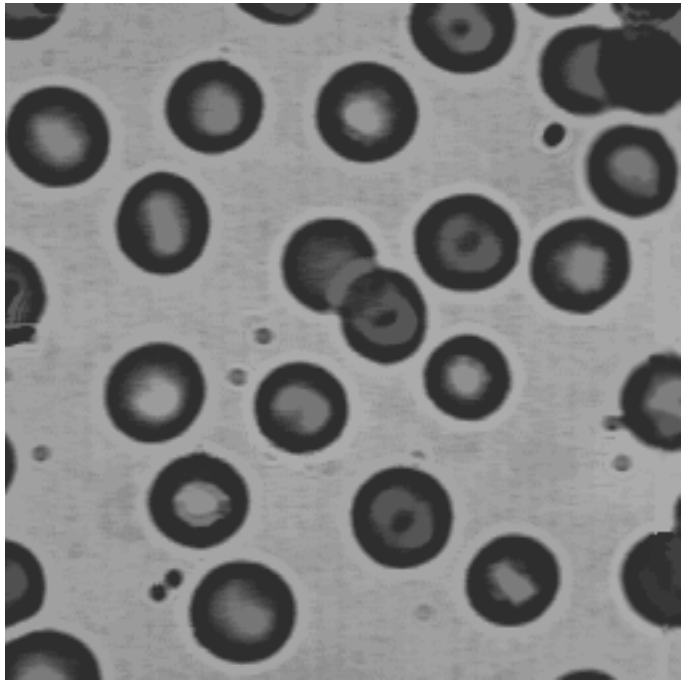
Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels



Gaussian smoothings

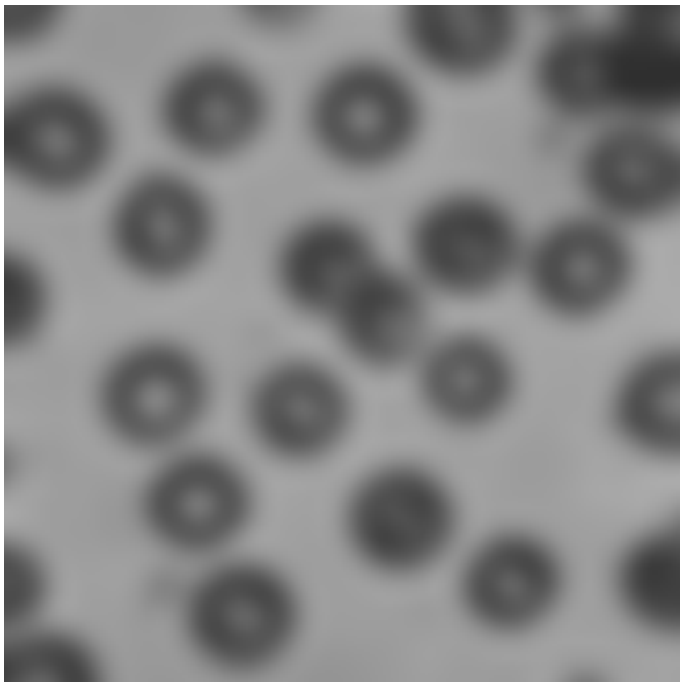


openings by reconstruction

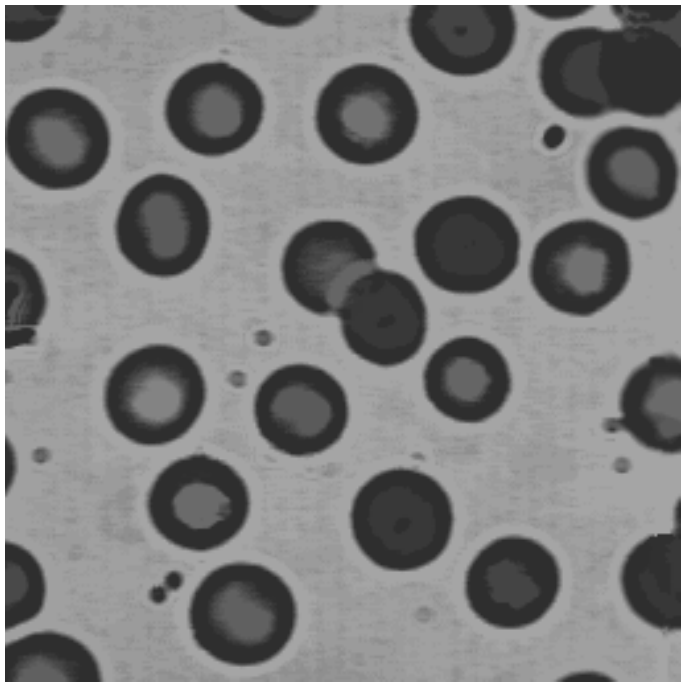
Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels



Gaussian smoothings

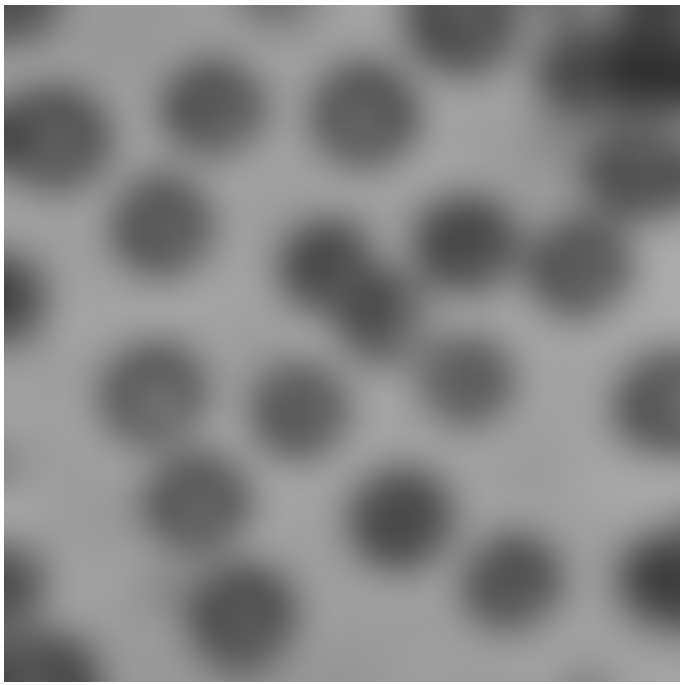


openings by reconstruction

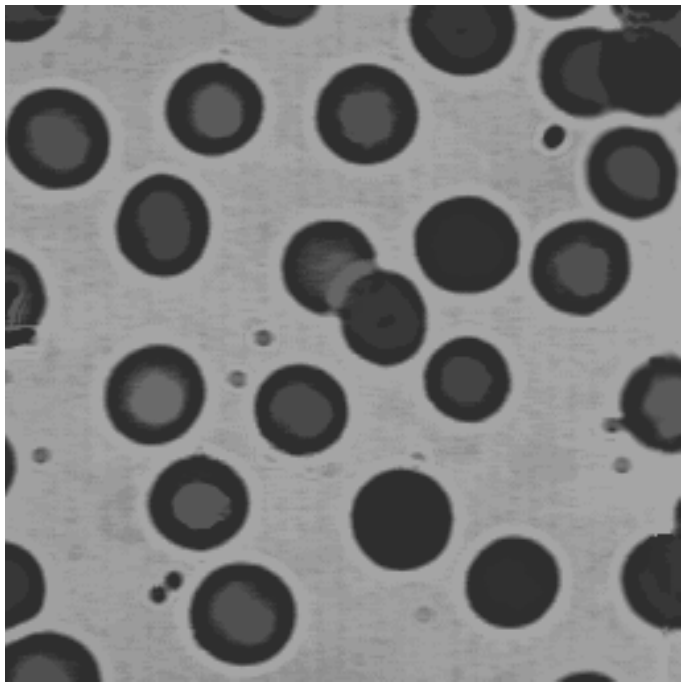
Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels



Gaussian smoothings



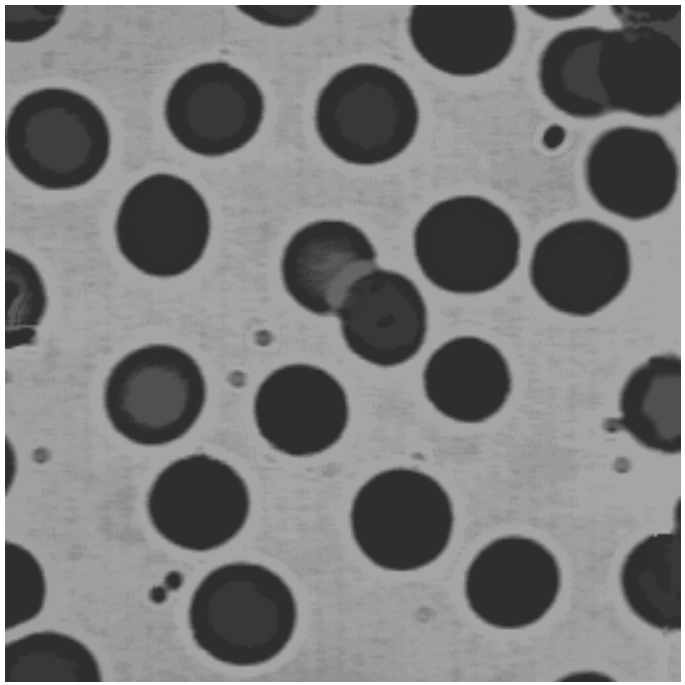
openings by reconstruction

Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels

Gaussian smoothings



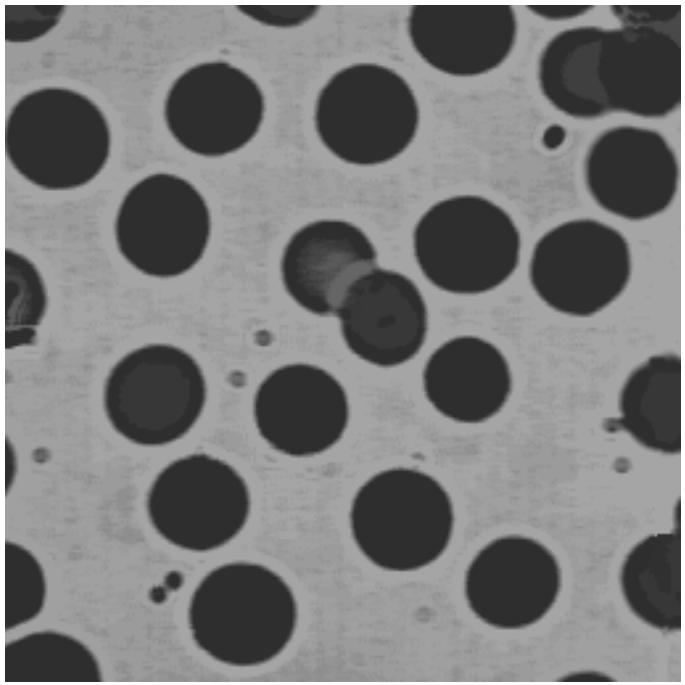
openings by reconstruction

Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels

Gaussian smoothings



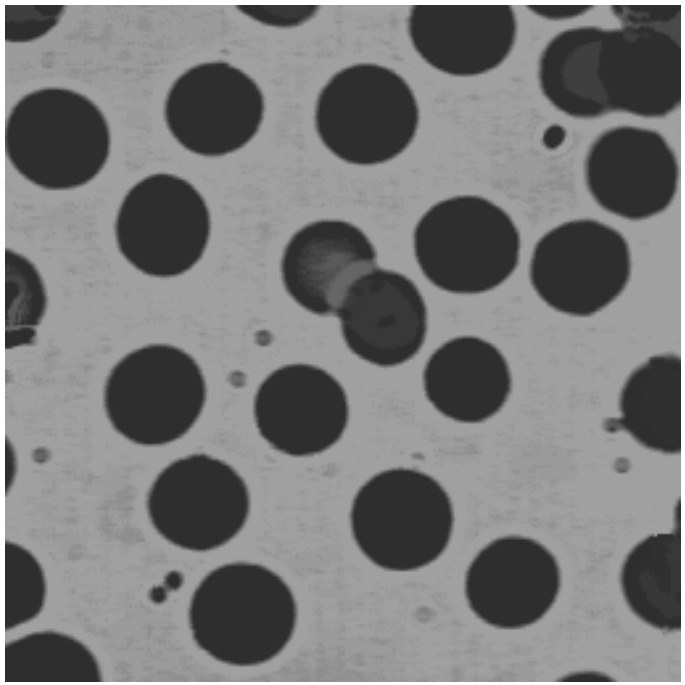
openings by reconstruction

Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels

Gaussian smoothings



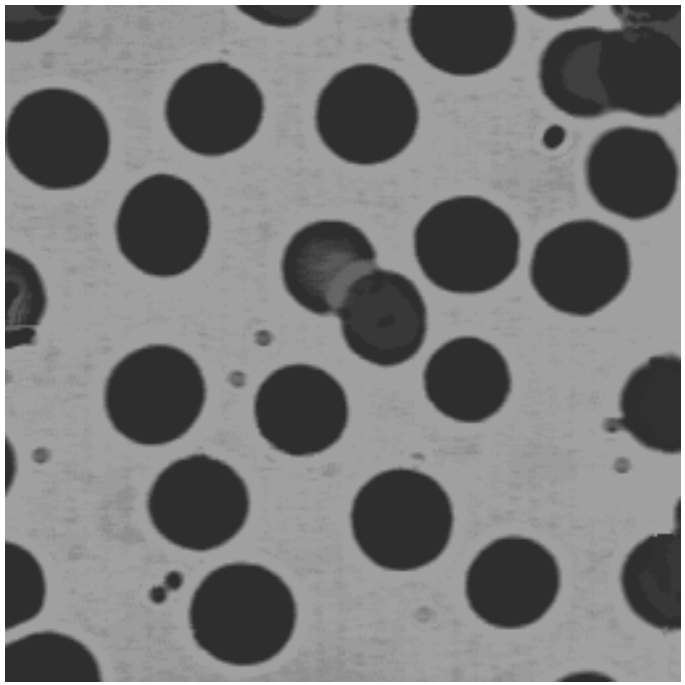
openings by reconstruction

Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels

Gaussian smoothings



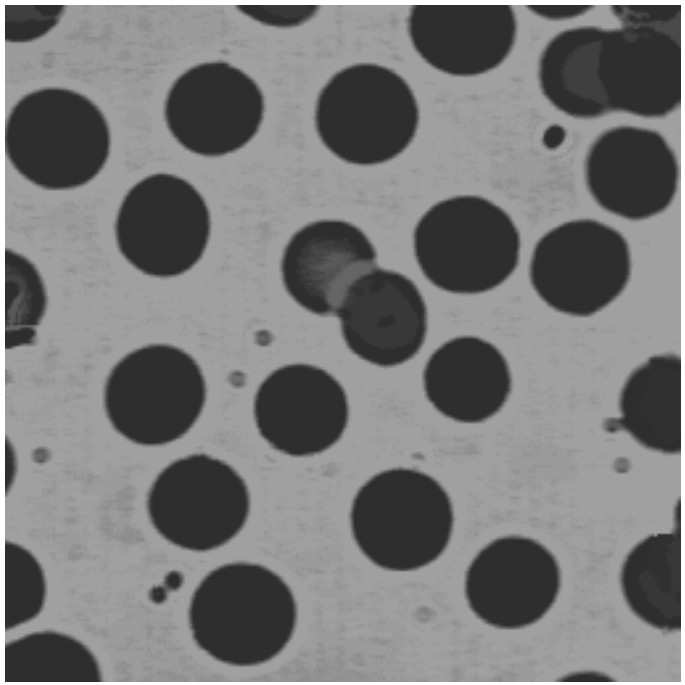
openings by reconstruction

Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels

Gaussian smoothings



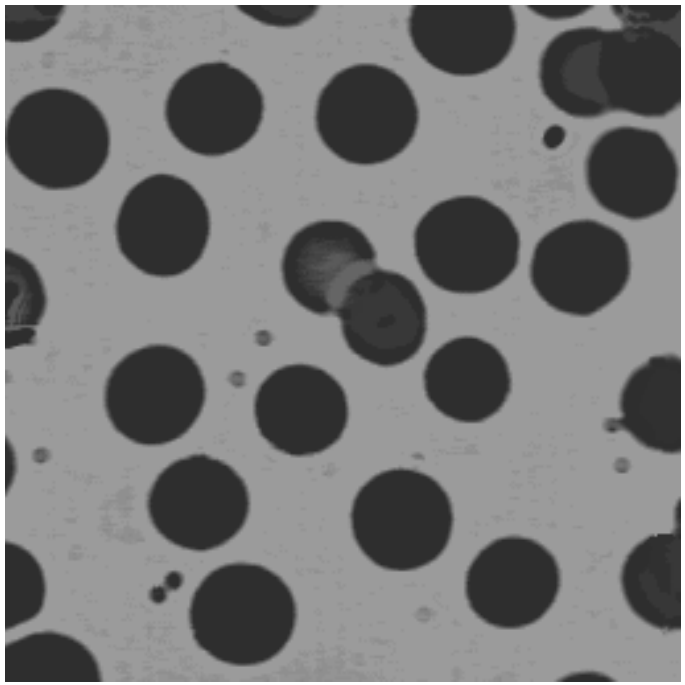
openings by reconstruction

Morphological scale-spaces

Multiscale representations

- built from successive applications of morphological filters
- used to produce morphological descriptions of images or pixels

Gaussian smoothings



openings by reconstruction

DMP for multispectral data

Multivariate morphology

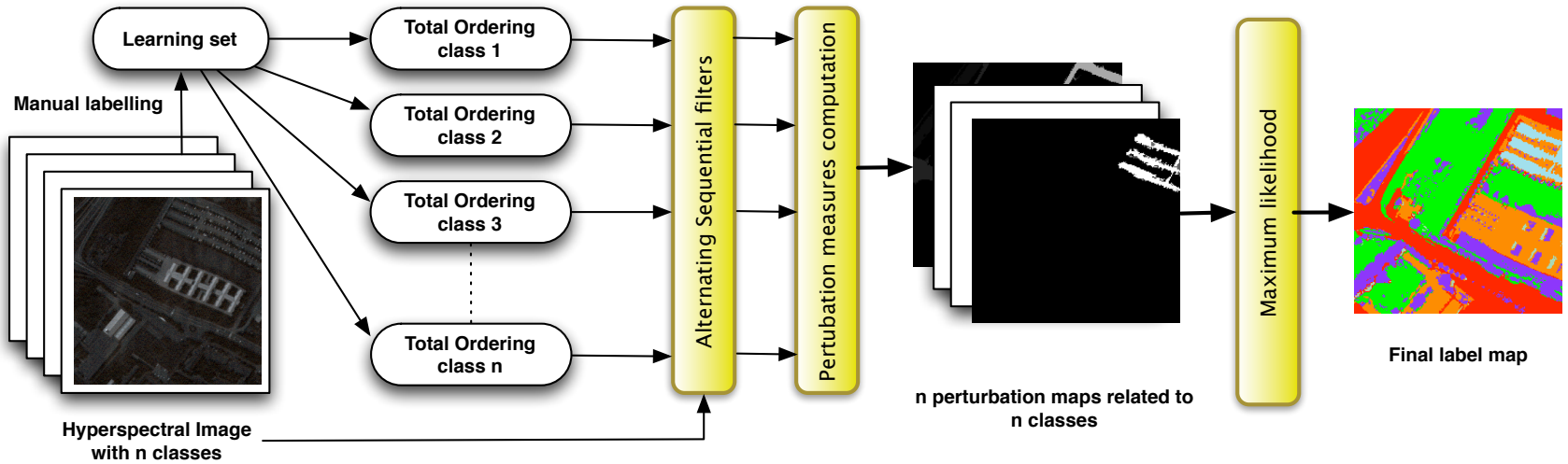
- marginal processing: each spectral band is a grayscale image
- vectorial processing: pixel values are vectors

Vectorial strategies

- partial ordering e.g. marginal strategy (but lack of correlation)
- pre-ordering e.g. reduced ordering (but no unique extrema)
- total ordering e.g. lexicographical ordering (but unbalanced)

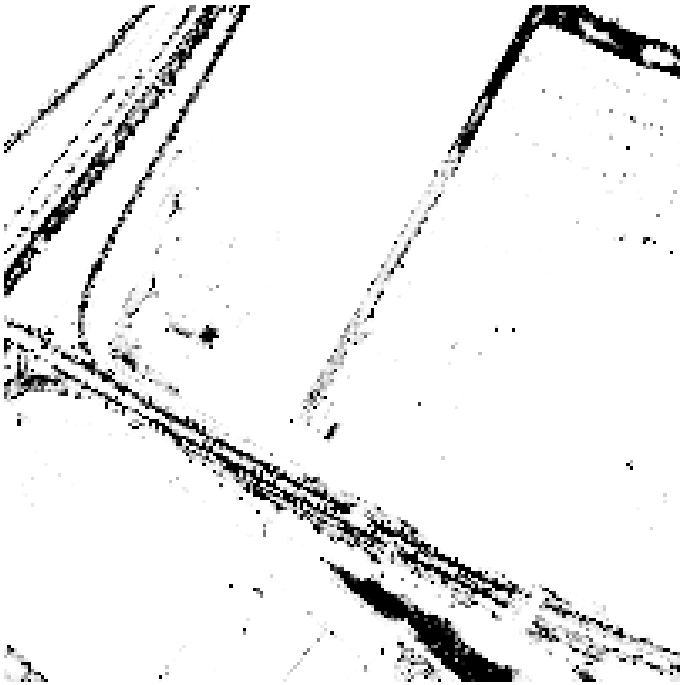
Supervised feature extraction

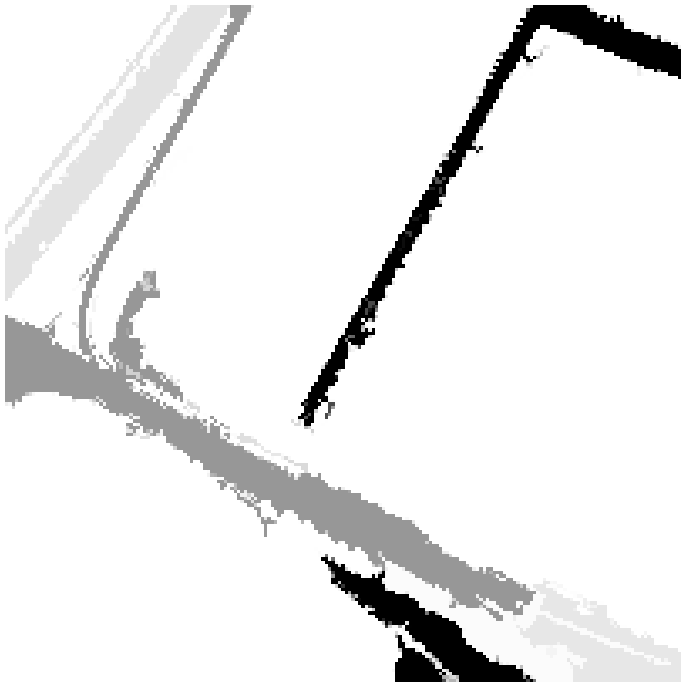
A perturbo-based ordering: $\mathbf{x_a} >_\ell \mathbf{x_b} \Rightarrow \tau_\ell(\mathbf{x_a}) > \tau_\ell(\mathbf{x_b})$
where $\tau_\ell(\cdot) = \tau(\cdot, \mathcal{M}_\ell)$
($+$ lexicographical comparison of $\mathbf{x_a}$ and $\mathbf{x_b}$ to ensure totality).



Supervised feature extraction

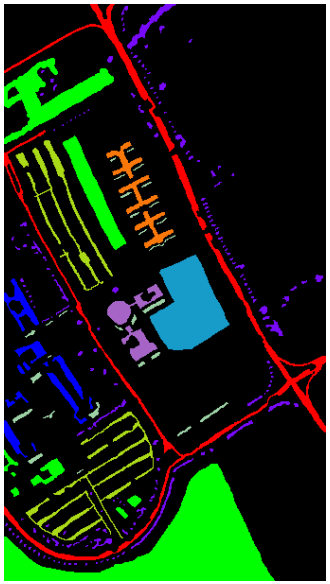
Here the DMP is replaced by an alternate sequential filter.

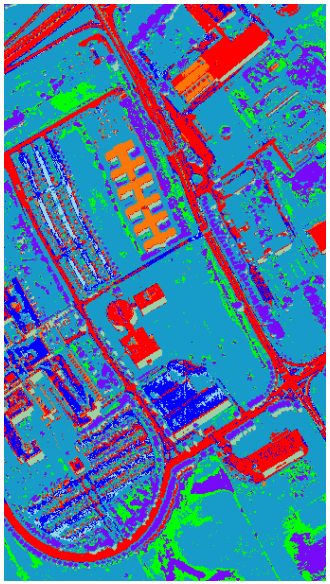


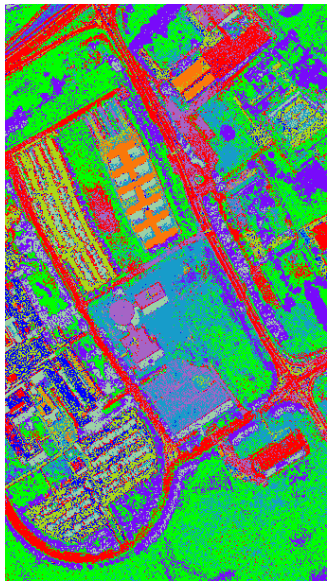


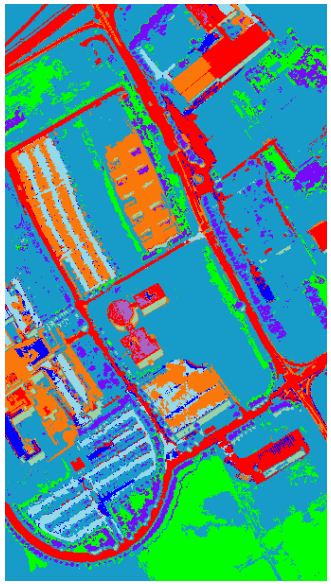
Supervised feature extraction

input


ground truth


Perturbo


EMP


Proposed

Outline

1 Introduction

2 Classification

3 Feature extraction

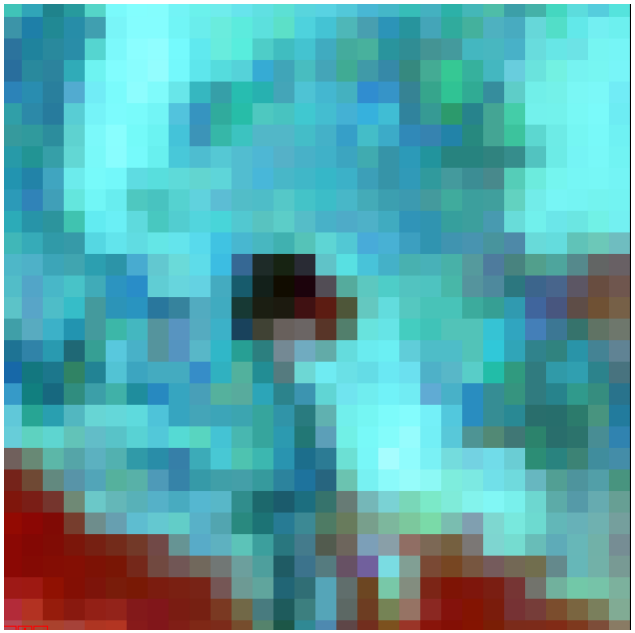
4 Segmentation

5 Conclusion

High and Very High Resolution Remote Sensing

Context

- new sensors provide images with very high spatial resolution
- analysis scale moves from the pixel level to the object level
- new techniques are required (e.g., object segmentation)

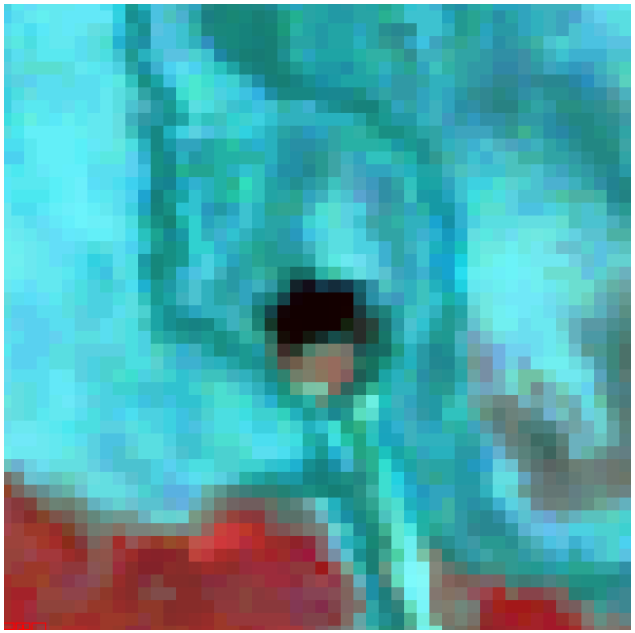


Landsat 60m / pixel

High and Very High Resolution Remote Sensing

Context

- new sensors provide images with very high spatial resolution
- analysis scale moves from the pixel level to the object level
- new techniques are required (e.g., object segmentation)

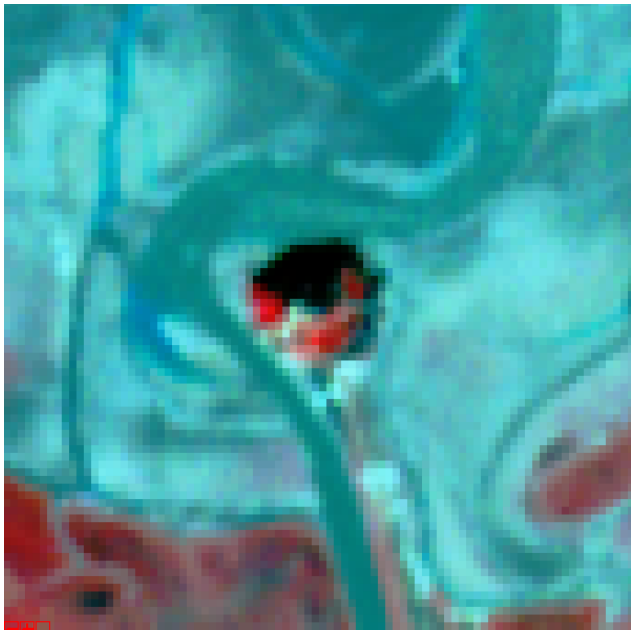


Landsat 30m / pixel

High and Very High Resolution Remote Sensing

Context

- new sensors provide images with very high spatial resolution
- analysis scale moves from the pixel level to the object level
- new techniques are required (e.g., object segmentation)

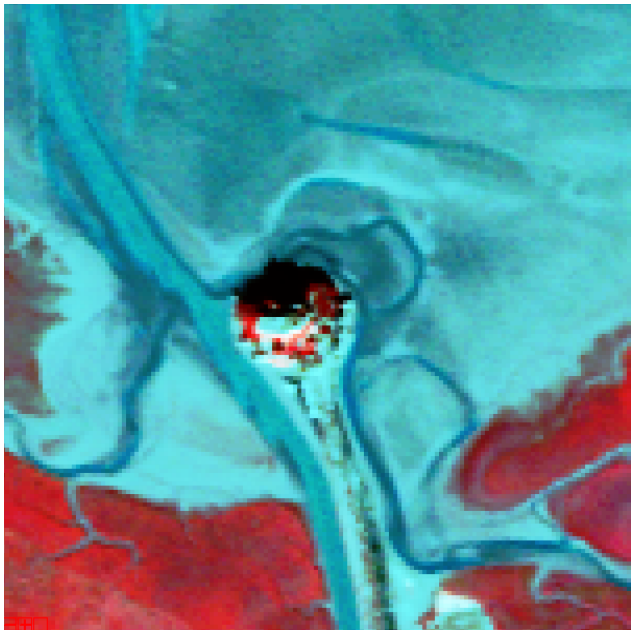


Aster 15m / pixel

High and Very High Resolution Remote Sensing

Context

- new sensors provide images with very high spatial resolution
- analysis scale moves from the pixel level to the object level
- new techniques are required (e.g., object segmentation)

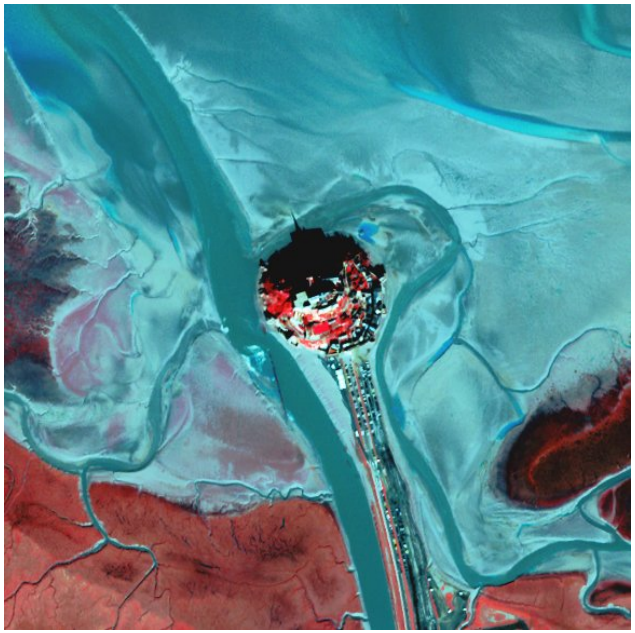


Spot5 10m / pixel

High and Very High Resolution Remote Sensing

Context

- new sensors provide images with very high spatial resolution
- analysis scale moves from the pixel level to the object level
- new techniques are required (e.g., object segmentation)

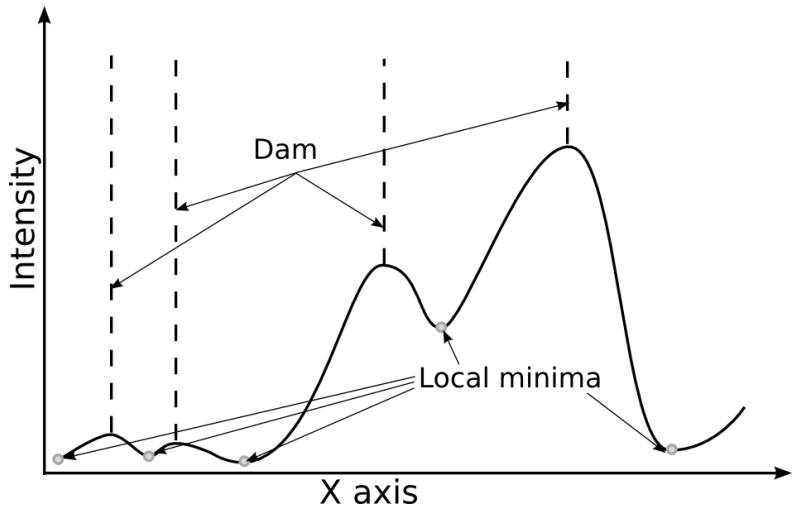


Quickbird 2.4m / pixel

Segmentation with the Watershed Transform

Immersion paradigm

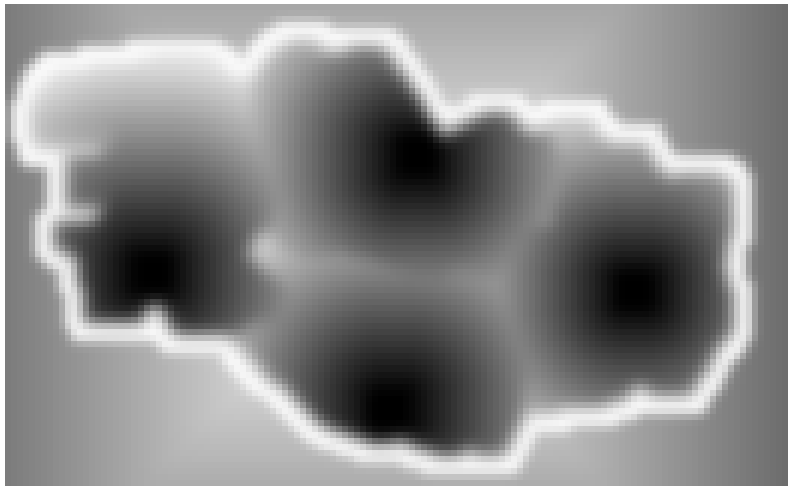
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

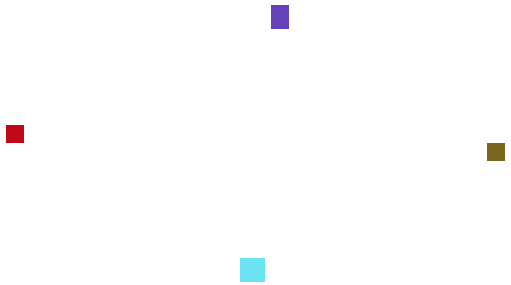
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

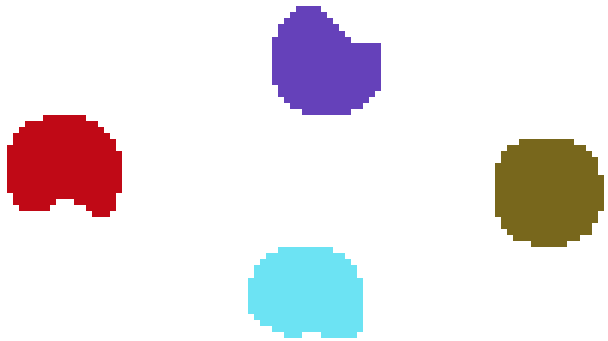
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

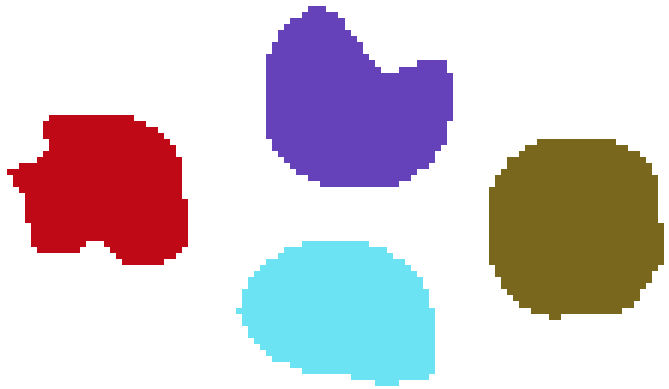
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

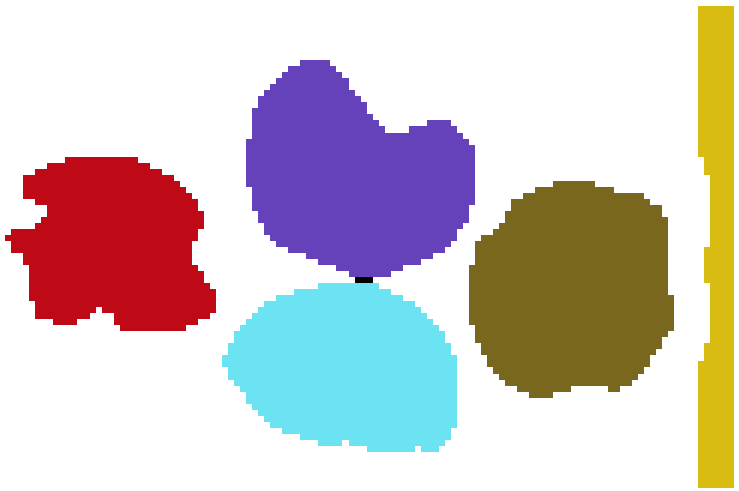
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

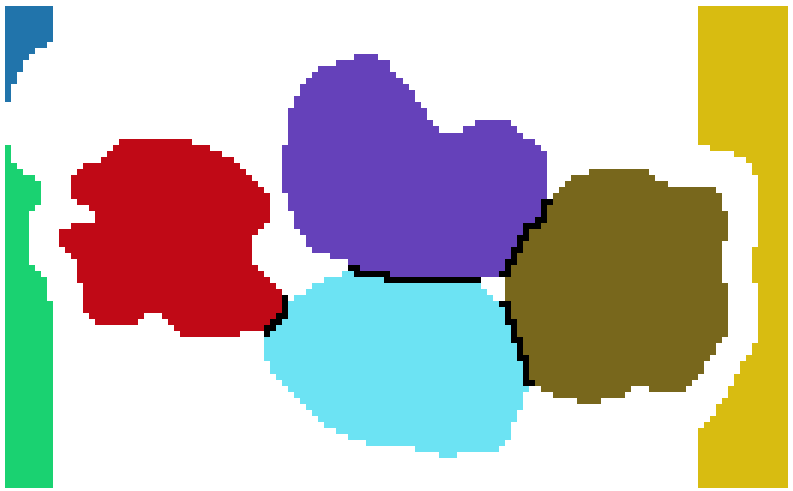
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

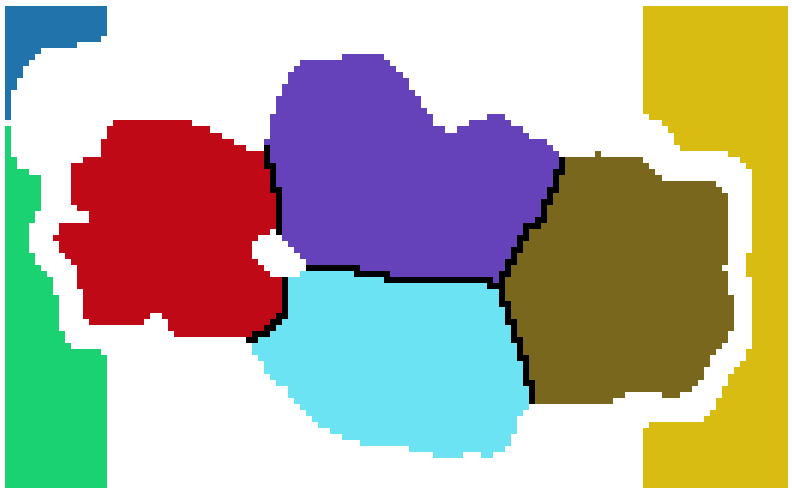
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

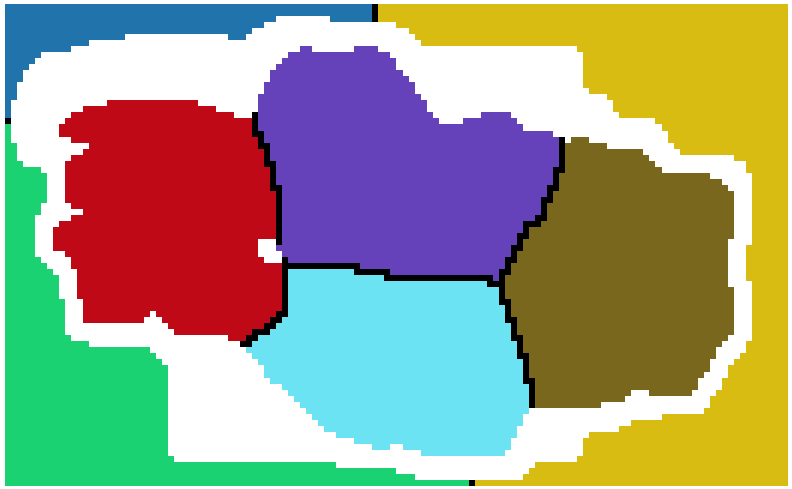
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

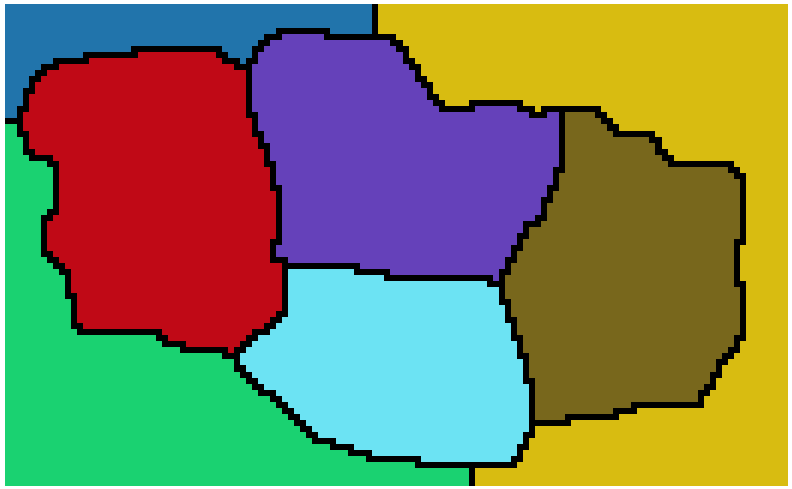
Immersion paradigm

1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines

Segmentation with the Watershed Transform

Immersion paradigm

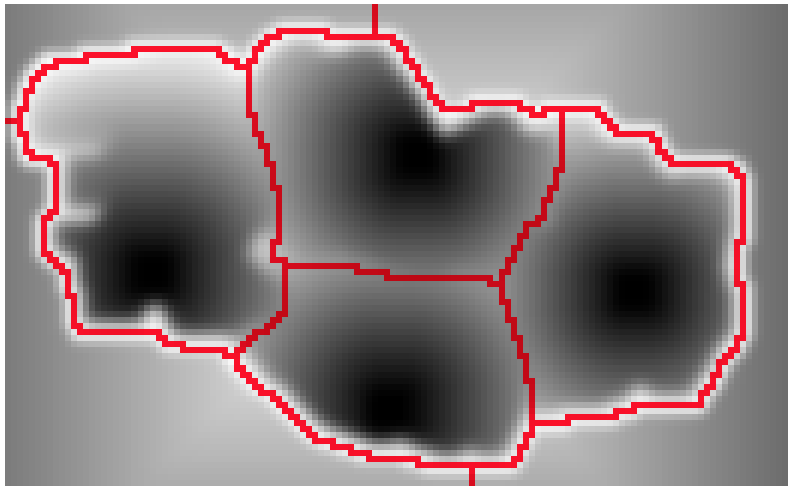
1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Segmentation with the Watershed Transform

Immersion paradigm

1. the input image is considered as a topographic surface
2. surface is flooded from its local minima (catchment basins)
3. dams are built to avoid water merging
4. after flooding the whole surface, dams are watershed lines



Shortcomings of the Watershed Transform

Shortcomings

The method is very prone to oversegmentation, solutions are:

- using some markers to identify expected regions
- rely on some oversegmentation reduction techniques
- multispectral objects

Proposed solutions

Involve Machine Learning!

- before the segmentation process
- after the segmentation process
- or even during the segmentation process

Shortcomings of the Watershed Transform

Shortcomings

The method is very prone to oversegmentation, solutions are:

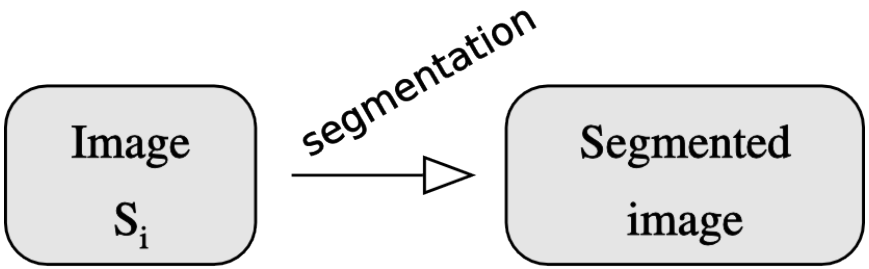
- using some markers to identify expected regions
- rely on some oversegmentation reduction techniques
- multispectral objects

Proposed solutions

Involve Machine Learning!

- before the segmentation process
- after the segmentation process
- or even during the segmentation process

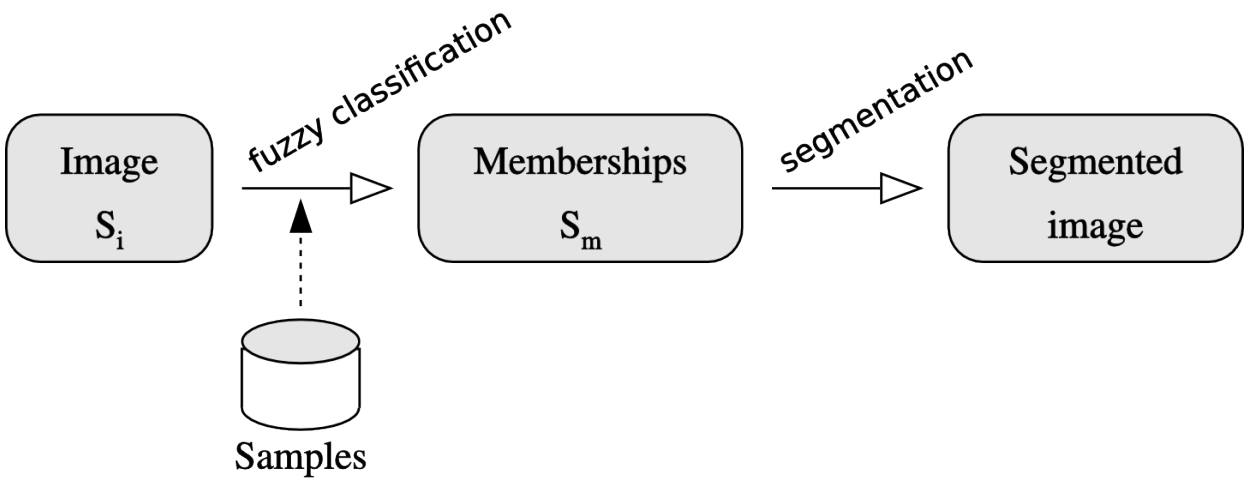
Supervised classification with non-spatial samples



How the samples may be used

- preprocessing: learning set to modify the input space for watershed segmentation
- postprocessing: test set to optimize oversegmentation reduction parameters

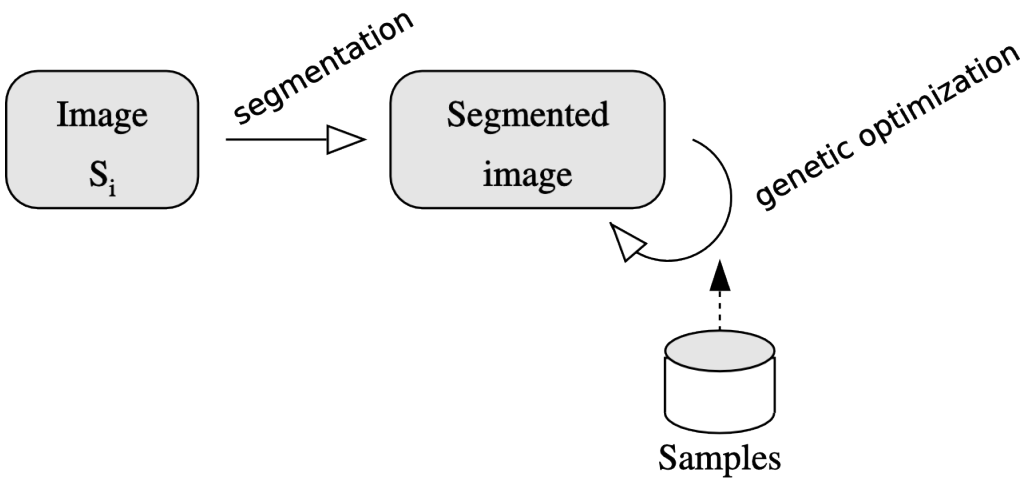
Supervised classification with non-spatial samples



How the samples may be used

- preprocessing: learning set to modify the input space for watershed segmentation
- postprocessing: test set to optimize oversegmentation reduction parameters

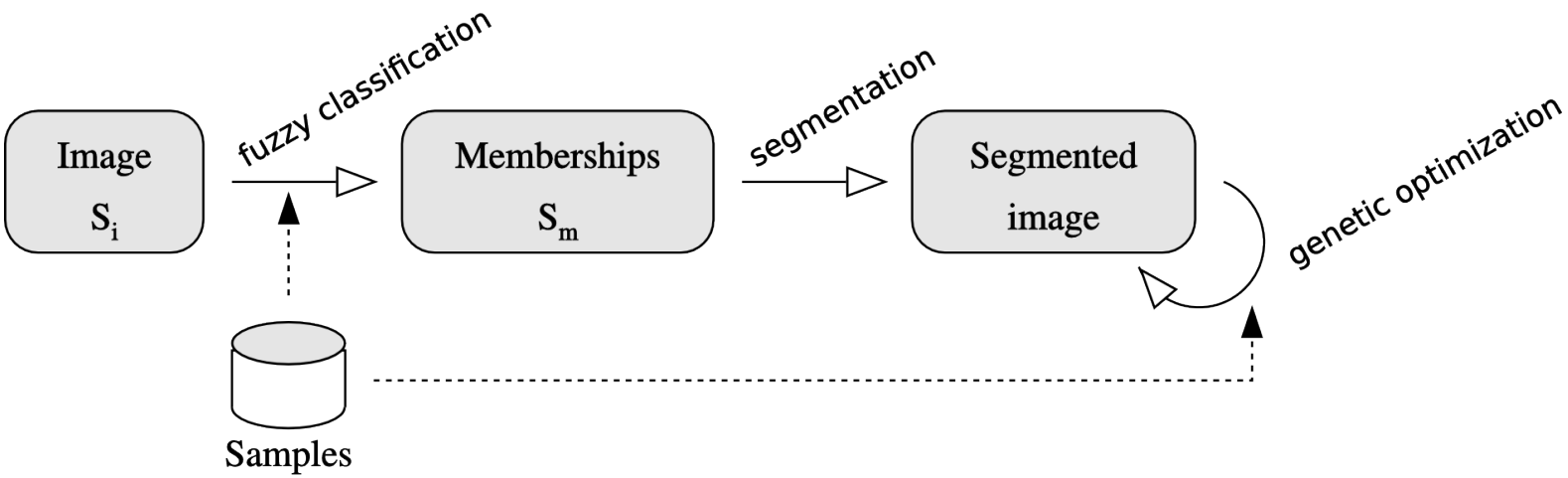
Supervised classification with non-spatial samples



How the samples may be used

- preprocessing: learning set to modify the input space for watershed segmentation
- postprocessing: test set to optimize oversegmentation reduction parameters

Supervised classification with non-spatial samples



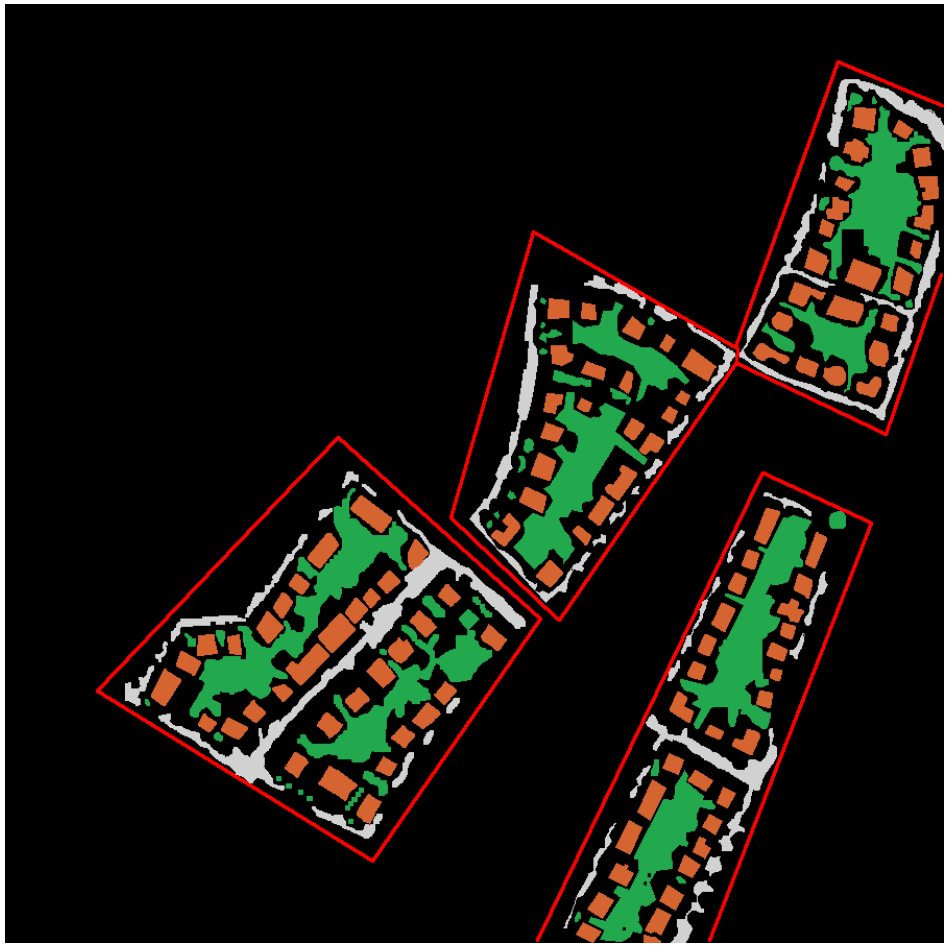
How the samples may be used

- preprocessing: learning set to modify the input space for watershed segmentation
- postprocessing: test set to optimize oversegmentation reduction parameters

Application to segmentation of urban scenes

Input image

Application to segmentation of urban scenes



Learning sets

Application to segmentation of urban scenes

Unsupervised watershed

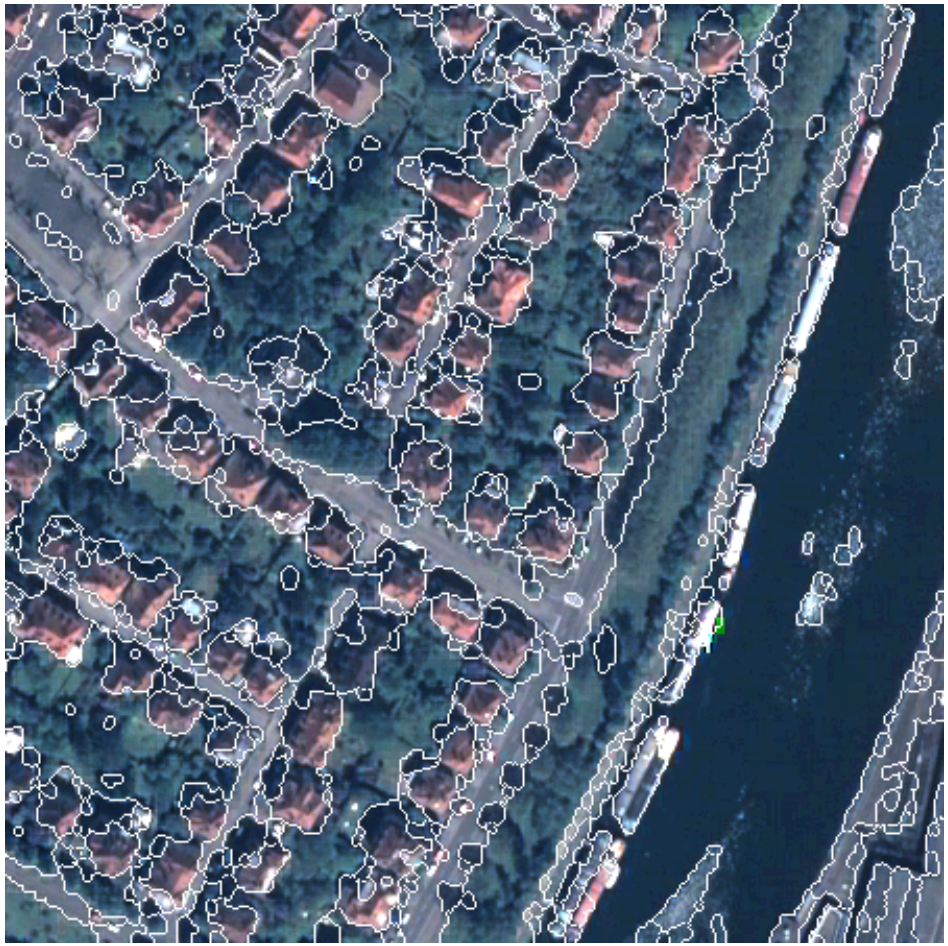
Application to segmentation of urban scenes

(Pre-)Supervised watershed

Application to segmentation of urban scenes

(Post-)Supervised watershed

Application to segmentation of urban scenes



(Pre+Post-)Supervised watershed

Application to segmentation of urban scenes

ENVI segmentation

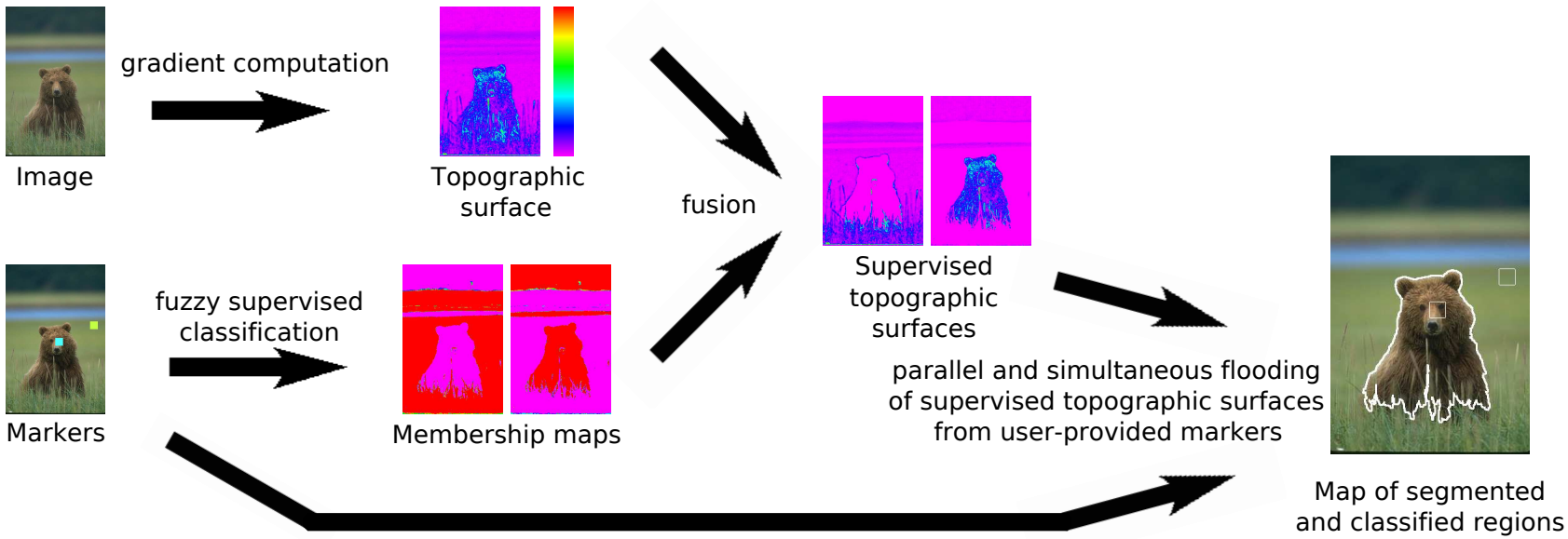
Application to segmentation of urban scenes

E-cognition segmentation

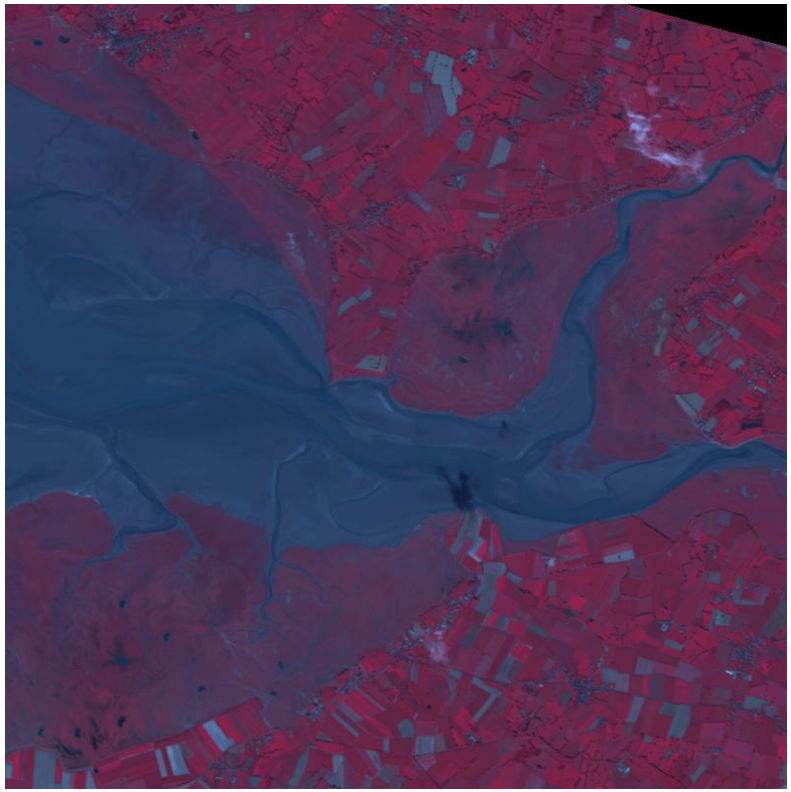
Supervised classification with spatial samples

Principle

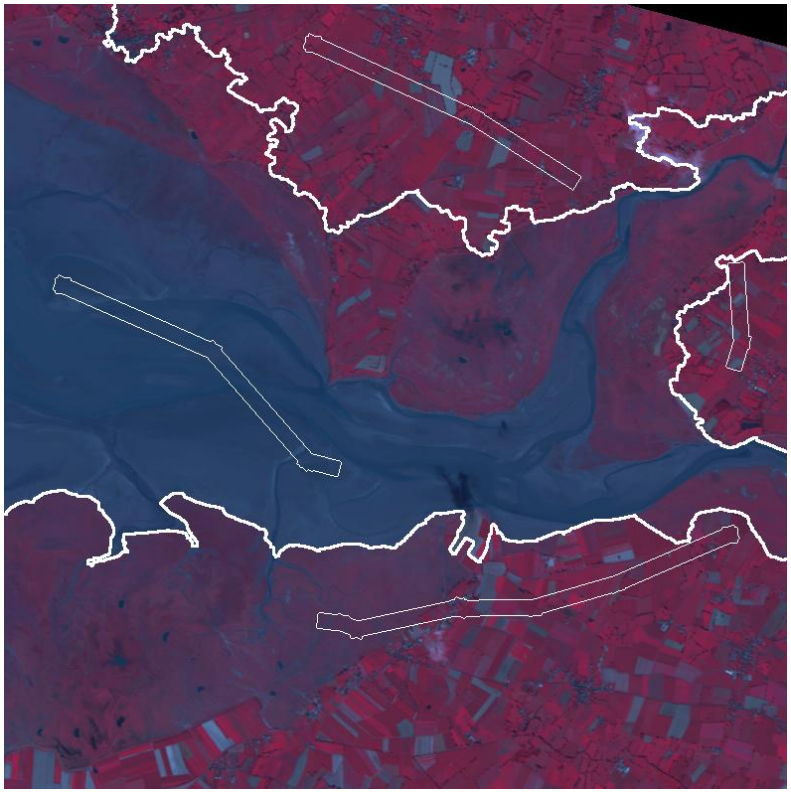
- learning sets are markers in watershed segmentation
- multiple surface are flooded simultaneously



Application to interactive coastline extraction

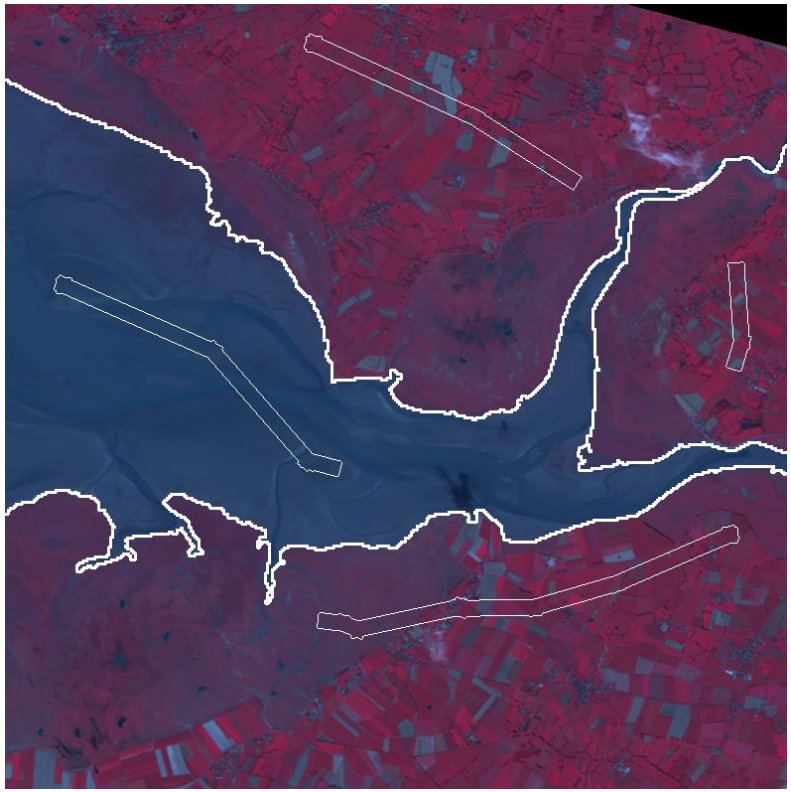


input image

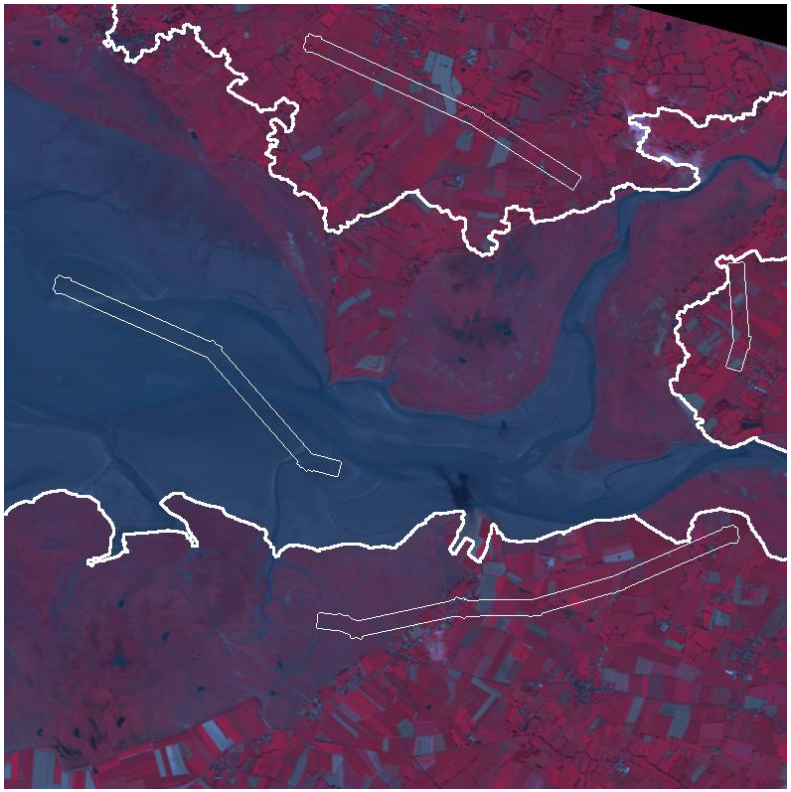


marker-based watershed

Application to interactive coastline extraction

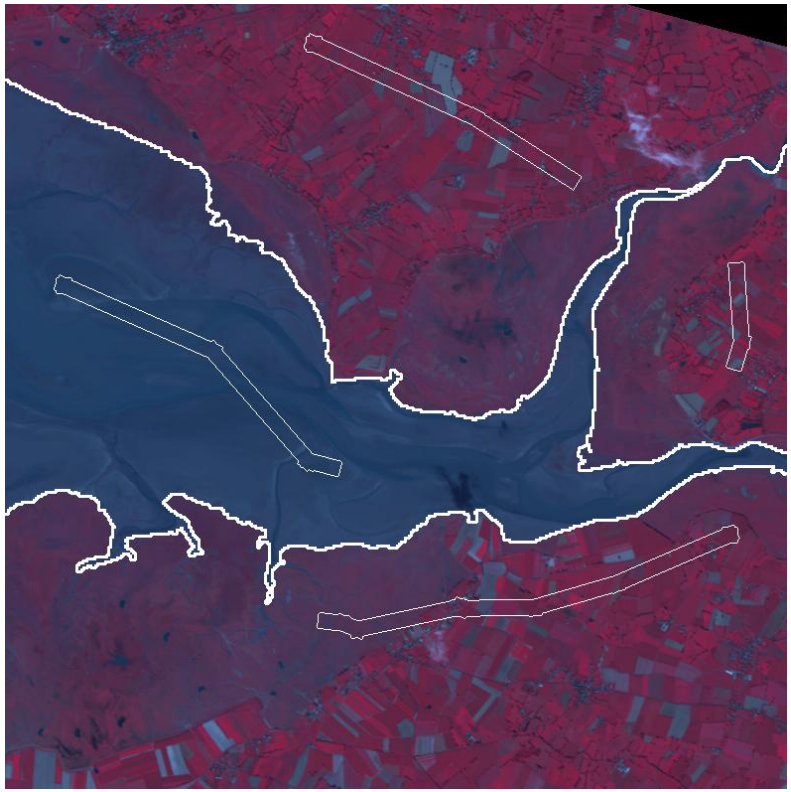


using spectral markers

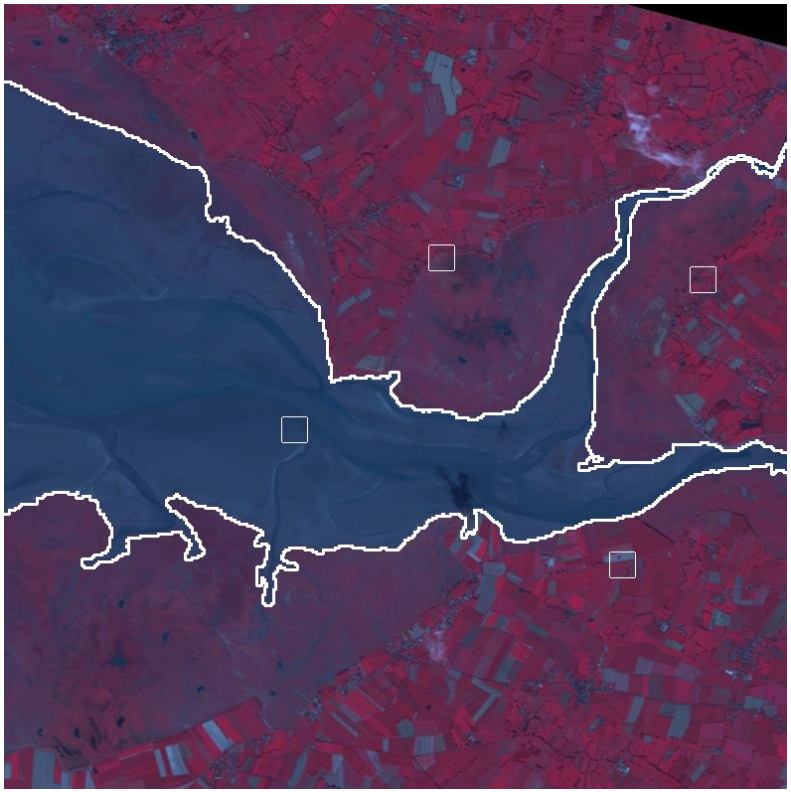


marker-based watershed

Application to interactive coastline extraction

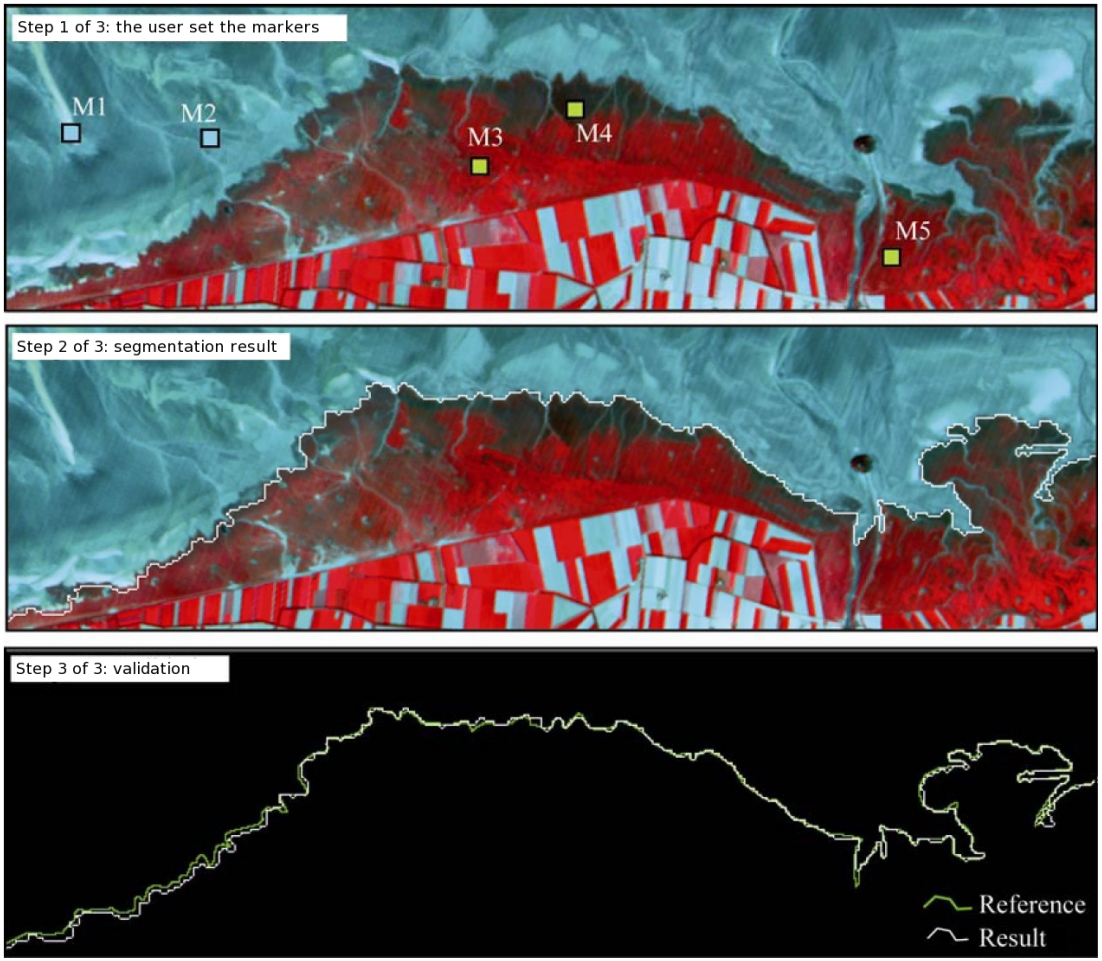


using spectral markers

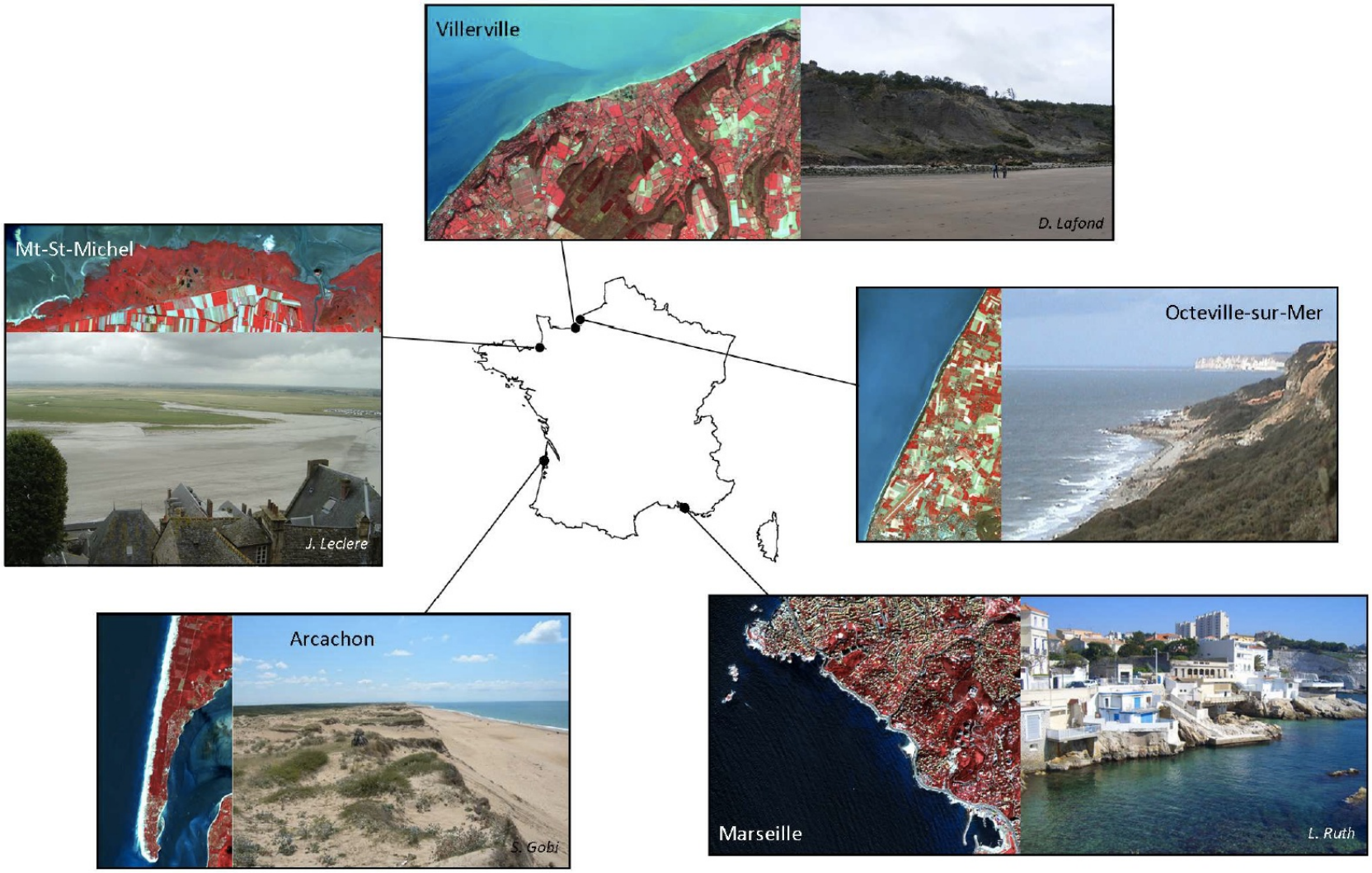


even with smaller markers

Application to interactive coastline extraction



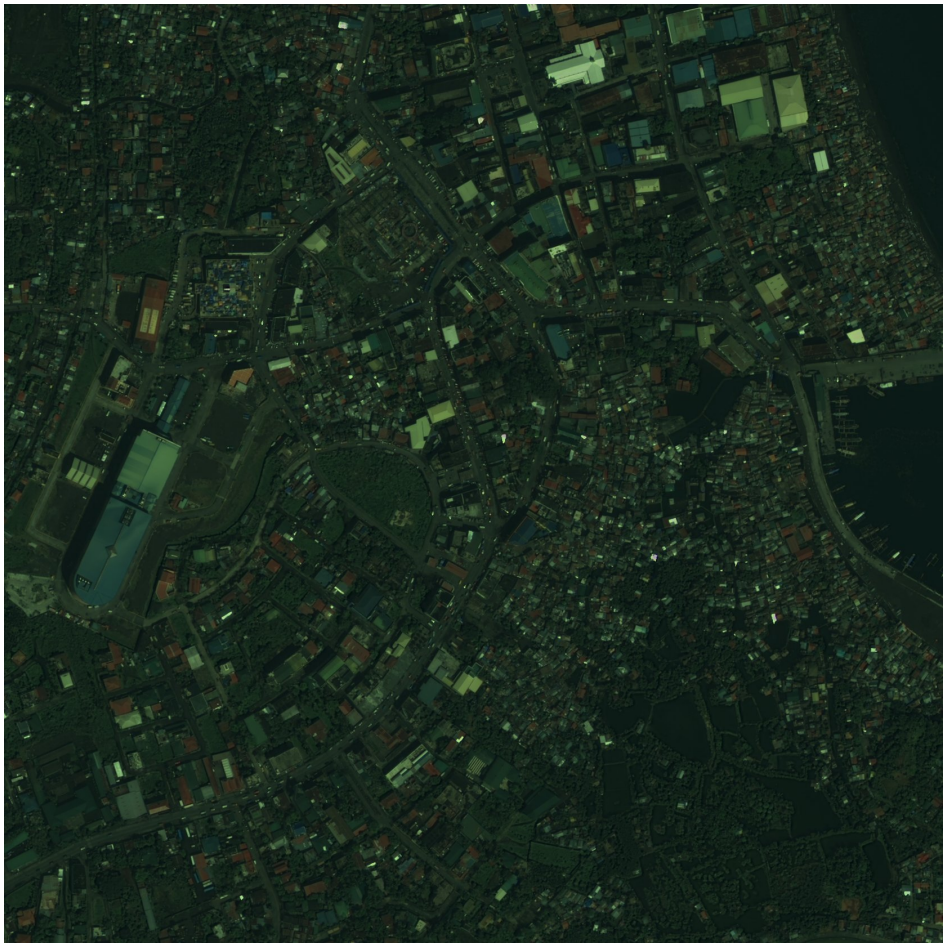
Application to interactive coastline extraction



Building detection challenge

Context

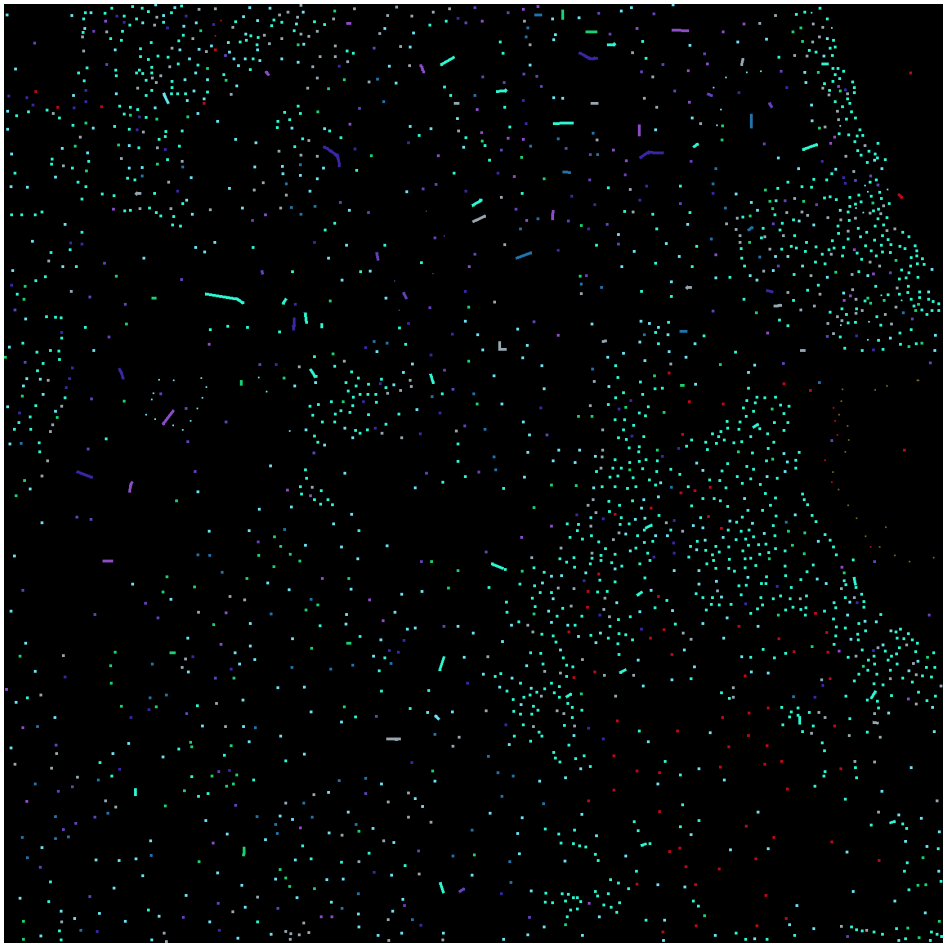
- objective: relevance of general user-driven morphological tools ?
- attempt #1: fully supervised strategy (3000 buildings / 90 minutes of mouse clicking)
- attempt #2: weakly-supervised strategy (14 markers)



Building detection challenge

Context

- objective: relevance of general user-driven morphological tools ?
- attempt #1: fully supervised strategy (3000 buildings / 90 minutes of mouse clicking)
- attempt #2: weakly-supervised strategy (14 markers)



Building detection challenge

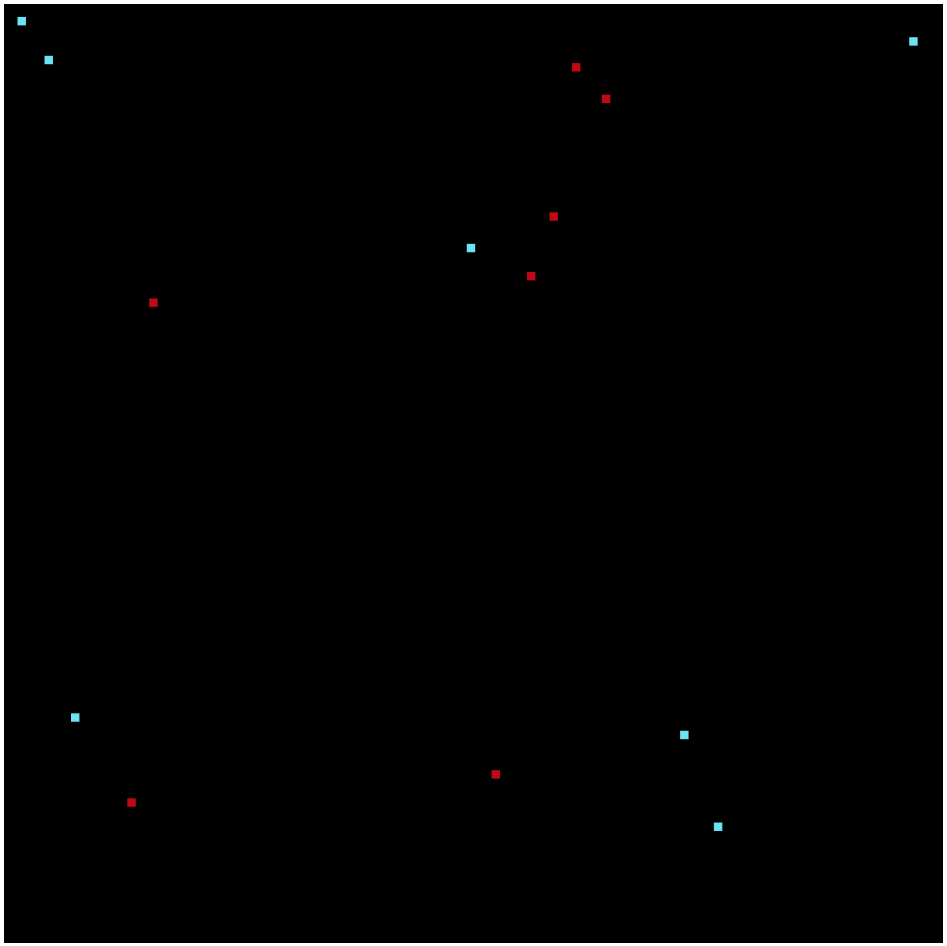
Context

- objective: relevance of general user-driven morphological tools ?
- attempt #1: fully supervised strategy (3000 buildings / 90 minutes of mouse clicking)
- attempt #2: weakly-supervised strategy (14 markers)

Building detection challenge

Context

- objective: relevance of general user-driven morphological tools ?
- attempt #1: fully supervised strategy (3000 buildings / 90 minutes of mouse clicking)
- attempt #2: weakly-supervised strategy (14 markers)



Building detection challenge

Context

- objective: relevance of general user-driven morphological tools ?
- attempt #1: fully supervised strategy (3000 buildings / 90 minutes of mouse clicking)
- attempt #2: weakly-supervised strategy (14 markers)

Building detection challenge

Results

- published in PRRS'08 and more recently in PRL'09
- precision/recall using multiple evaluation criteria
- very good performance with no tuning

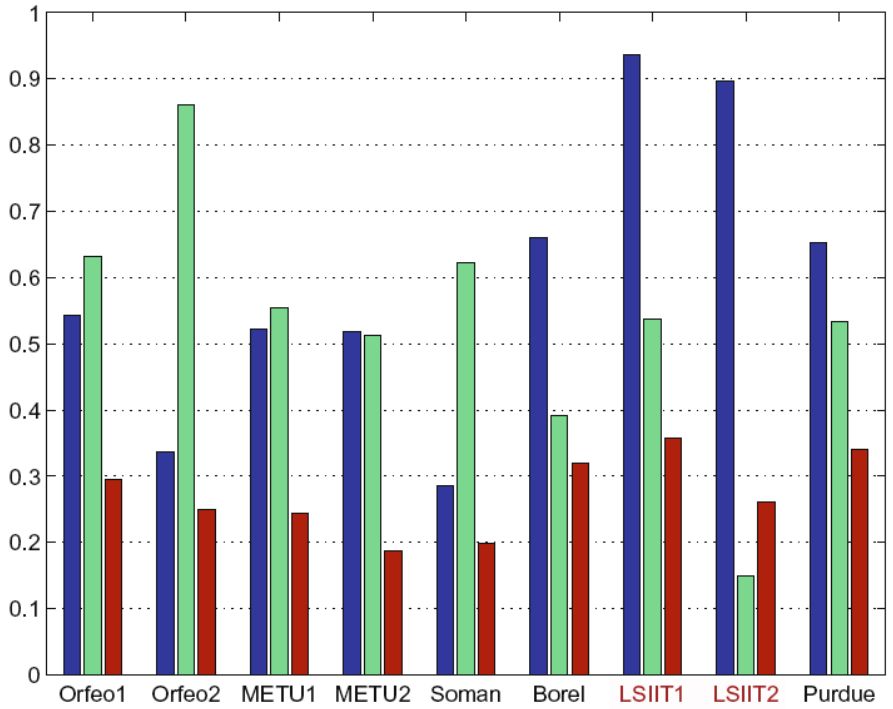


Fig. 5. Precision (blue), recall (green), and detection accuracy (red) scores obtained using the bipartite graph matching algorithm for the results in Fig. 4. (For interpretation of the references in color in this figure legend, the reader is referred to the web version of this article.)

Building detection challenge

Results

- published in PRRS'08 and more recently in PRL'09
- precision/recall using multiple evaluation criteria
- very good performance with no tuning

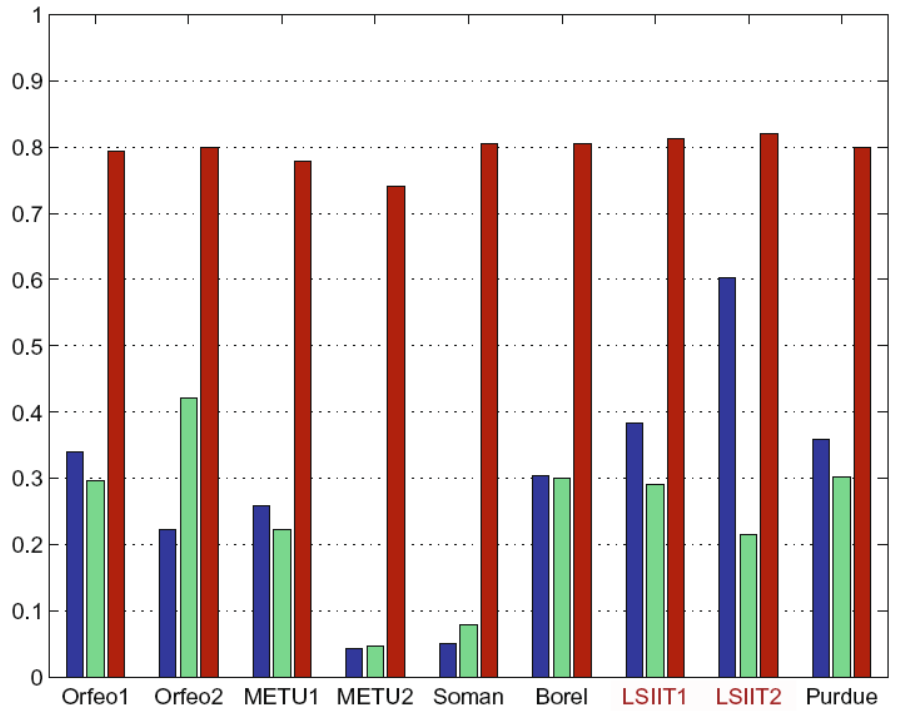


Fig. 6. Precision (blue), recall (green), and detection accuracy (red) scores obtained using the Hoover algorithm for the results in Fig. 4. (For interpretation of the references in color in this figure legend, the reader is referred to the web version of this article.)

Building detection challenge

Results

- published in PRRS'08 and more recently in PRL'09
- precision/recall using multiple evaluation criteria
- very good performance with no tuning

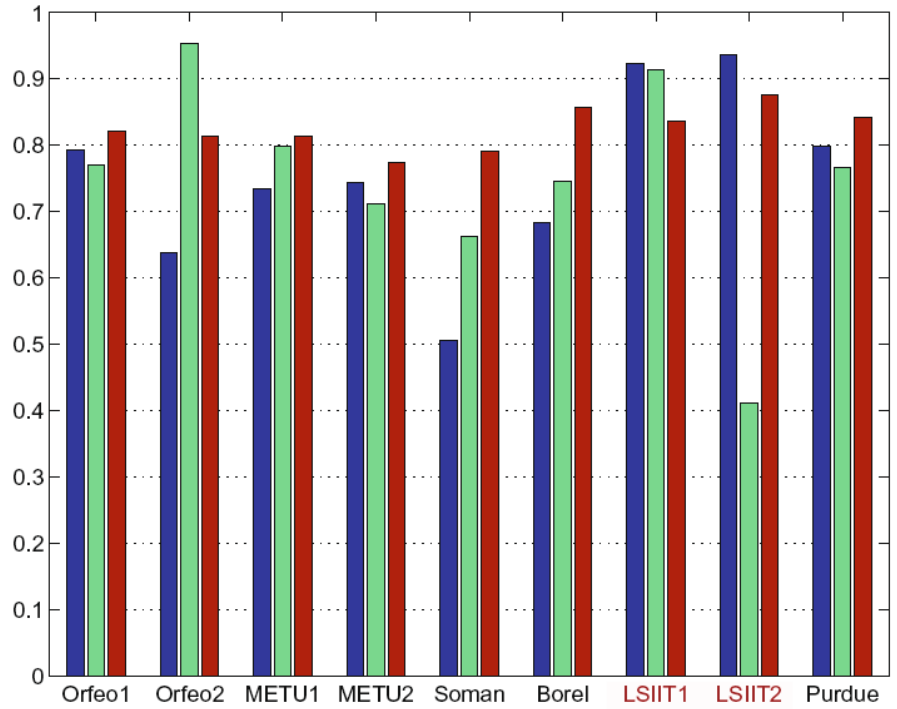


Fig. 7. Precision (blue), recall (green), and detection accuracy (red) scores obtained using the proposed multi-object maximum overlap matching algorithm and the Mallows measure for the results in Fig. 4. (For interpretation of the references in color in this figure legend, the reader is referred to the web version of this article.)

Building detection challenge

Results

- published in PRRS'08 and more recently in PRL'09
- precision/recall using multiple evaluation criteria
- very good performance with no tuning

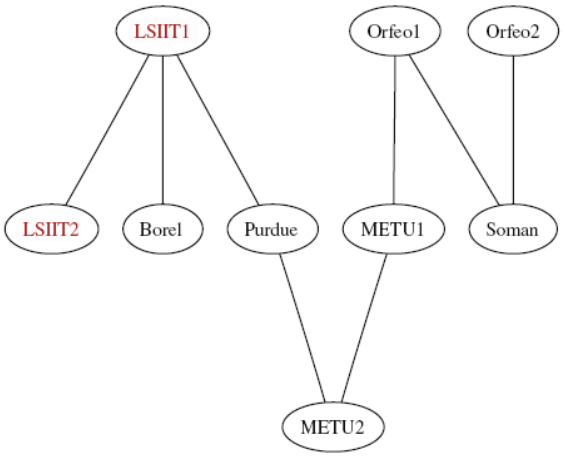
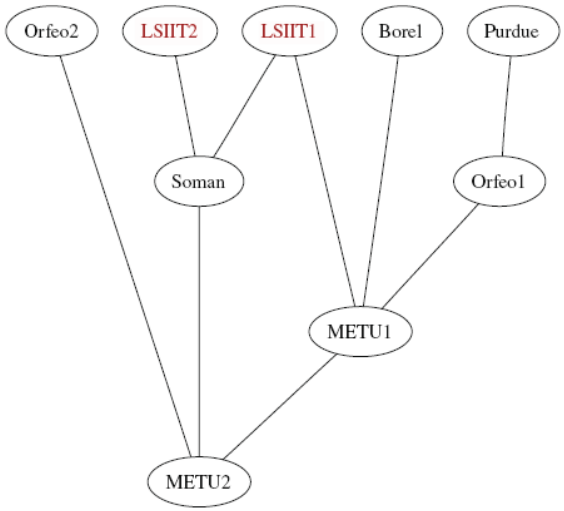


Fig. 8. The Hasse diagram and the corresponding ranking for the scores in Fig. 5 obtained using the bipartite graph matching algorithm.

Building detection challenge

Results

- published in PRRS'08 and more recently in PRL'09
- precision/recall using multiple evaluation criteria
- very good performance with no tuning



Rank 1: Purdue
Rank 2: LSIIT1
Rank 3: LSIIT2
Rank 4: Borel
Rank 5: Orfeo2
Orfeo1
Rank 7: Soman
Rank 8: METU1
Rank 9: METU2

Fig. 9. The Hasse diagram and the corresponding ranking for the scores in Fig. 6 obtained using the Hoover algorithm.

Building detection challenge

Results

- published in PRRS'08 and more recently in PRL'09
- precision/recall using multiple evaluation criteria
- very good performance with no tuning

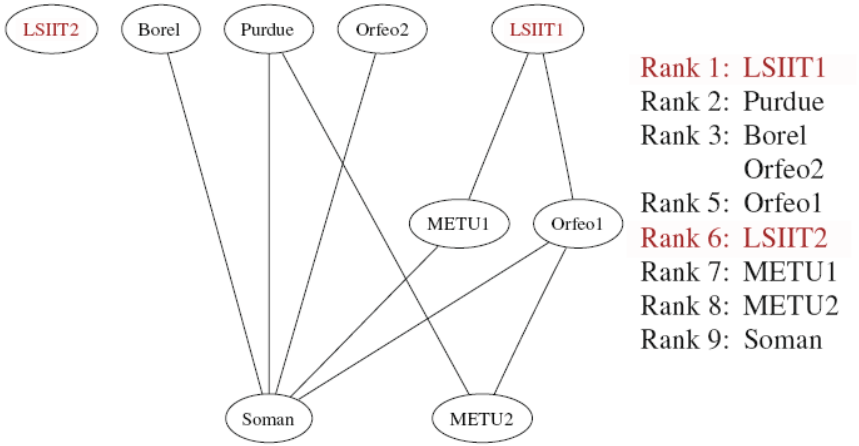


Fig. 10. The Hasse diagram and the corresponding ranking for the scores in Fig. 7 obtained using the proposed multi-object maximum overlap matching algorithm and the Mallows measure.

Outline

1 Introduction

2 Classification

3 Feature extraction

4 Segmentation

5 Conclusion

Classwise strategy is a relevant alternative to be further explored.
Open issues: outliers/misclassified points, active strategies

ML can even improve early stages in image analysis.
New trends in machine learning (combined with advances in image processing) are expected to have high impact in remote sensing.

Some further readings . . .

For any details / publications: `firstname.lastname@univ-ubs.fr` or `firstname.lastname@irisa.fr`
`http://www-irisa.univ-ubs.fr` or `http://www.irisa.fr`

Machine Learning

- Courty, Burger, Laurent (2011) *PerTurbo: a new classification algorithm based on the spectrum perturbations of the Laplace-Beltrami operator*, European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)
- Chapel, Burger, Courty, Lefèvre (2012) *Classwise hyperspectral image classification with perturbo method*, IEEE International Geosciences and Remote Sensing Symposium (IGARSS)

Feature extraction

- Aptoula, Lefèvre (2012) *A Comparative Study on Multivariate Mathematical Morphology*, Pattern Recognition 40(11):2914–2929
- Courty, Aptoula, Lefèvre (2012) *A Classwise Supervised Ordering Approach for Morphology Based Hyperspectral Image Classification*, IAPR International Conference on Pattern Recognition (ICPR)

Image segmentation

- Derivaux, Forestier, Wemmert, Lefèvre (2010) *Supervised image segmentation using watershed transform, fuzzy classification and evolutionary computation*, Pattern Recognition Letters, 31(15):2364–2374
- Lefèvre (2007) *Knowledge from markers in watershed segmentation*, IAPR International Conference on Computer Analysis of Images and Patterns (CAIP), LNCS 4673, pages 579–586
- Puissant, Lefèvre, Rougier, Malet (2012) *Automated mapping of coastline from high resolution satellite images using supervised segmentation*, International Conference on Geographic Object-Based Image Analysis (GEOBIA), pages 515–517