Advances in Spectral-Spatial Classification of Hyperspectral Imagery

Yuliya Tarabalka[1,2],
Jón Atli Benediktsson[2], Jocelyn Chanussot[1]

[1]GIPSA-Lab, Grenoble Institute of Technology, France
[2]University of Iceland, Reykjavik, Iceland
e-mail: yuliya.tarabalka@hyperinet.eu

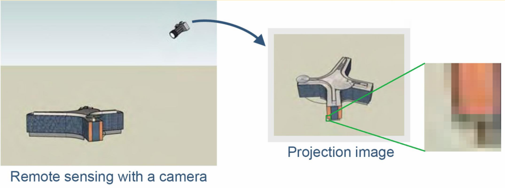
October 26, 2009

Outline

Spectral imagery

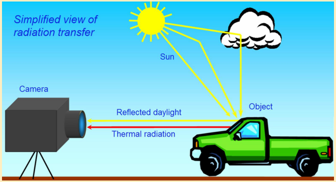
Spatial context

- 2D
- Digital image → composed of pixels

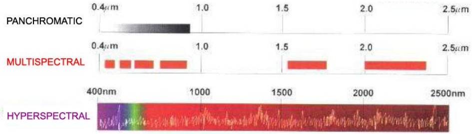


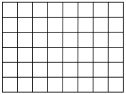
Spectral context

- Measurement of intensity of EM radiations (light)

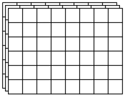


Spectral context

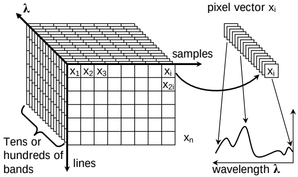




1 band



2-10 bands



Panchromatic

- one grey level value per pixel

Multispectral

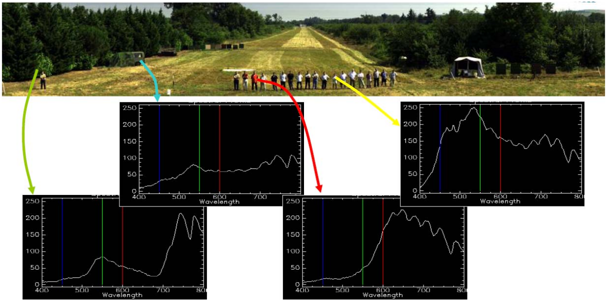
- 2-10 bands
- limited spectral info

Hyperspectral

- tens or hundreds of narrow bands
- detailed spectral info

Hyperspectral image

- Every pixel contains a detailed spectrum (>100 spectral bands)
- More information per pixel → increasing capability to distinguish objects



Hyperspectral image

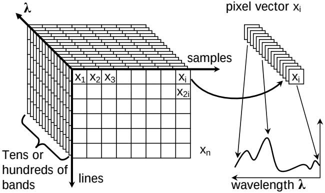
Every pixel contains a detailed spectrum (>100 spectral bands)

+ More information per pixel → increasing capability to distinguish objects

– Dimensionality increases → image analysis becomes more complex

⇓

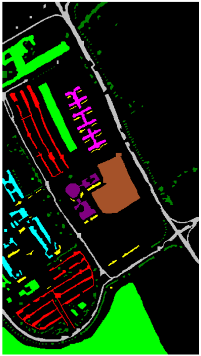
Efficient algorithms for automatic processing are required!



Classification problem

Input ROSIS image
[$610 \times 340 \times 103$]

Ground-truth data



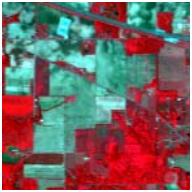
Task

Assign **every** pixel to **one** of the **nine** classes:

alphalt
meadows
gravel
trees
metal sheets
bare soil
bitumen
bricks
shadows

Classification problem (2)

Input AVIRIS image $[145 \times 145 \times 200]$



Ground-truth data

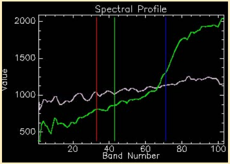
Task

Assign **every** pixel to **one** of the **16** classes: corn-no till, corn-min till, corn, soybeans-no till, soybeans-min till, soybeans-clean till, alfalfa, grass/pasture, grass/trees, grass/pasture-mowed, hay-windrowed, oats, wheat, woods, bldg-grass-tree-drives, stone-steel towers

Classification approaches

Only spectral information

- Spectra of each pixel is analyzed
- Directly accessible
- Variety of methods (e.g. SVM)
  $\rightarrow$ good classification results



$\Rightarrow$

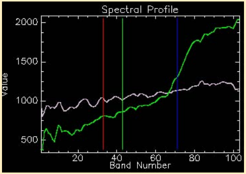
alphalt
meadows
gravel
trees
metal sheets
bare soil
bitumen
bricks
shadows

Overall accuracy = 81.01%

Classification approaches

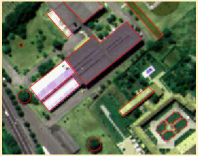
Only spectral information

- Spectrum of each pixel is analyzed
- Directly accessible
- Variety of methods (e.g. SVM)
  → good classification results



Spectral + spatial information

- Info about spatial structures included
- How to define structures?
  - closest neighborhood → not flexible enough
  - adaptive neighborhood (segmentation map)
    → currently investigated



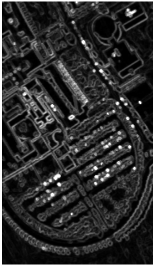
Objective

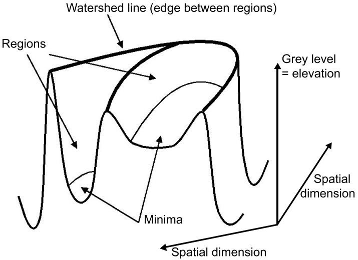
- Segment a hyperspectral image = find an exhaustive partitioning of the image into homogeneous regions
- Spectral info + spatial info $\rightarrow$ classify image

Outline

1. Watershed segmentation

gradient
⇒





Region growing method:

- **Minimum** of a gradient = core of a homogeneous region
- **1 region** = set of pixels connected to 1 local minimum of the gradient
- **Watershed lines** = edges between adjacent regions

1. Watershed segmentation

Tarabalka, Y., Chanussot, J., Benediktsson, J. A., Angulo, J., Fauvel, M., 2008. Segmentation and classification of hyperspectral data using watershed. In Proc. of IGARSS '08, Boston, USA.

Original image

1. Watershed segmentation

Tarabalka, Y., Chanussot, J., Benediktsson, J. A., Angulo, J., Fauvel, M., 2008. Segmentation and classification of hyperspectral data using watershed. In Proc. of IGARSS '08, Boston, USA.

Original image

⇒

Robust Color Morpho Gradient

1. Watershed segmentation

Tarabalka, Y., Chanussot, J., Benediktsson, J. A., Angulo, J., Fauvel, M., 2008. Segmentation and classification of hyperspectral data using watershed. In Proc. of IGARSS '08, Boston, USA.

Original image

⇒

Robust Color Morpho Gradient

⇒

Watershed 11802 regions

1. Watershed segmentation

Tarabalka, Y., Chanussot, J., Benediktsson, J. A., Angulo, J., Fauvel, M., 2008. Segmentation and classification of hyperspectral data using watershed. In Proc. of IGARSS '08, Boston, USA.

Original image

⇒

Robust Color Morpho Gradient

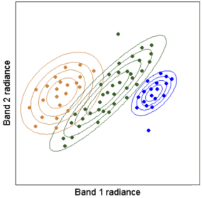
⇒

Watershed 11802 regions

⇒

Edges → to adjacent regions

2. Partitional clustering (EM)

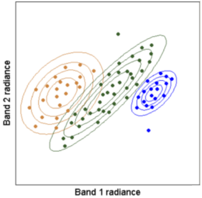


1. Clustering
   - pixels are grouped into C clusters
   - in each cluster $\rightarrow$ pixels drawn from a Gaussian distribution
   - distribution parameters $\rightarrow$ EM algorithm
2. Labeling of connected components

10 clusters

$\Rightarrow$

2. Partitional clustering (EM)



1. Clustering
   - pixels are grouped into C clusters
   - in each cluster → pixels drawn from a Gaussian distribution
   - distribution parameters → EM algorithm
2. Labeling of connected components

10 clusters
⇒
21450 regions

same cluster, but different regions!

3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$

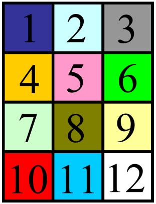
1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2

3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$

1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with
   $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2

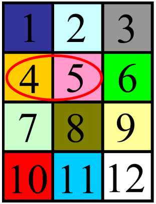


3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$

1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2

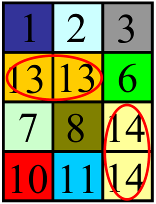


3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$

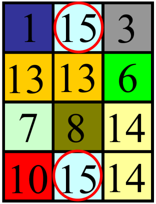
1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2



3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$

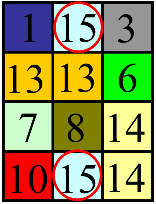


1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with
   $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2

3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$



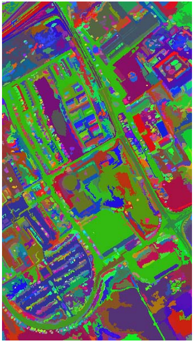
1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with
   $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2

3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos\left(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2}\right)$$

1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2



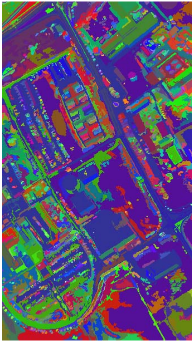
$SCW = 0.0$
7231 regions

3. Hierarchical image segmentation (HSEG,[Tilton98])

- Region growing + Spectral Clustering
- Dissimilarity criterion (DC):
  Spectral Angle Mapper (SAM)
  between the region mean vectors u_i and u_j

$$SAM(u_i, u_j) = \arccos(\frac{u_i \cdot u_j}{\|u_i\|_2 \|u_j\|_2})$$

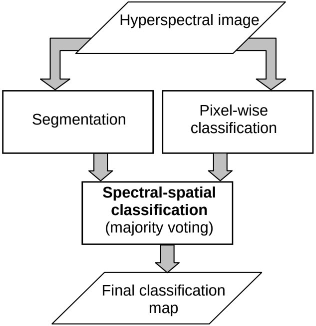
1. Each pixel - one region
2. Find DC_{min} between adjacent regions
3. Merge adjacent regions with $DC = DC_{min}$
4. Merge non-adjacent regions with
   $DC \leq DC_{min} \cdot SpectralClusterWeight$
5. If not converge, go to 2

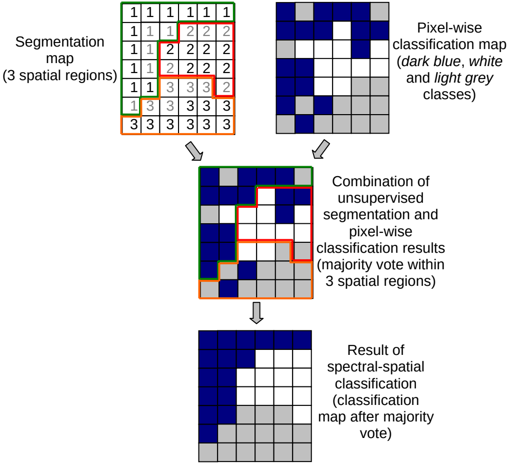


$SCW = 0.1$
7575 regions

Outline

Spectral-spatial classification scheme





Spectral-spatial classification

Original image

Spectral-spatial classification

Original image

⇒

SVM classification

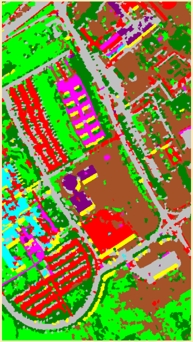
OA = 81.01%
AA = 88.25%

Spectral-spatial classification

SVM classification

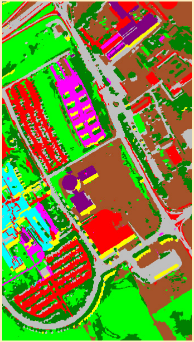
$\Rightarrow$

SVM + Watershed



SVM + Partit.clustering

SVM + HSEG ($SCW = 0.1$)



SVM classification	SVM + Watershed	SVM + Partit.clustering	SVM + HSEG ($SCW = 0.1$)
OA = 81.01%	OA = 85.42%	OA = 93.59%	OA = 93.85%
AA = 88.25%	AA = 91.31%	AA = 94.39%	AA = 97.07%

Classification accuracies (%):

	SVM	+Watersh.	+Part.Cl.	+HSEG		EMP[1]	ECHO
SCW				0.0	0.1		
Overall Acc.	81.01	85.42	93.59	90.00	**93.85**	85.22	87.58
Average Acc.	88.25	91.31	94.39	94.15	**97.07**	90.76	92.16
Kappa Coef.κ	75.86	81.30	91.48	86.86	**91.89**	80.86	83.90
asphalt	84.93	93.64	90.72	73.33	94.77	**95.36**	87.98
meadows	70.79	75.09	**92.73**	88.73	89.32	80.33	81.64
gravel	67.16	66.12	82.09	**97.47**	96.14	87.61	76.91
trees	97.77	98.56	99.21	98.45	98.08	98.37	**99.31**
metal sheets	99.46	99.91	**100**	99.10	99.82	99.48	99.91
bare soil	92.83	97.35	96.78	98.43	**99.76**	63.72	93.96
bitumen	90.42	96.23	92.46	95.92	**100**	98.87	92.97
bricks	92.78	97.92	97.80	98.81	**99.29**	95.41	97.35
shadows	98.11	96.98	97.74	97.11	96.48	97.68	**99.37**

[1] *A. Plaza et al., "Recent advances in techniques for hyperspectral image processing," Remote Sensing of Environment, vol. 113, Suppl. 1, 2009.*

Outline

1. Spectral-spatial classification improves accuracies when compared to pixel-wise classification
2. Several segmentation techniques are investigated
3. The HSEG segmentation map leads to the best classification
4. Obtained classification accuracies > all previous results

However...

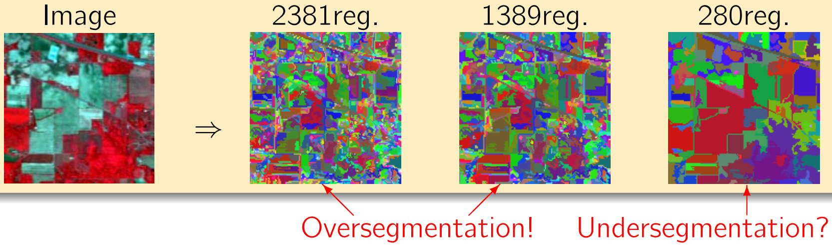
Unsupervised segmentation

- Unsupervised segmentation = exhaustive partitioning into homogeneous regions
- How to define a measure of homogeneity?

Unsupervised segmentation

- Unsupervised segmentation = exhaustive partitioning into homogeneous regions
- How to define a measure of homogeneity?

Hierarchical segmentation (HSEG, [Tilton98])

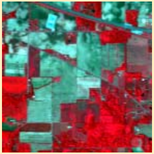


Unsupervised segmentation

- Unsupervised segmentation = exhaustive partitioning into homogeneous regions
- How to define a measure of homogeneity?

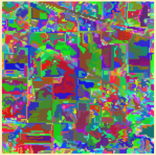
Hierarchical segmentation (HSEG, [Tilton98])

Image

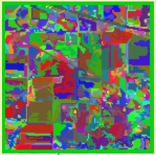


⇒

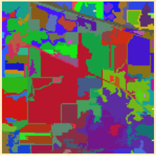
2381reg.



1389reg.



280reg.



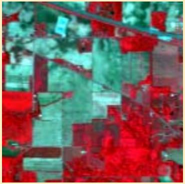
Oversegmentation Undersegmentation?

⇓

Preferred in previous works → not to lose objects

Watershed segmentation (IGARSS'08)

Original image



$\Rightarrow$

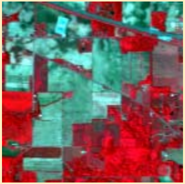
Robust Color Morpho Gradient

$\Rightarrow$

Watershed 1277 regions

Watershed segmentation (IGARSS'08)

Original image

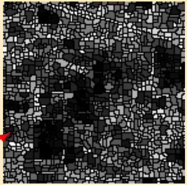


⇒

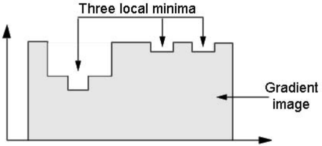
Robust Color Morpho Gradient

⇒

Watershed 1277 regions



Severe oversegmentation!



Every local minimum of the gradient
↓
one region

Marker-controlled segmentation

- Reduce oversegmentation $\Leftarrow$ incorporate *a priori* knowledge into segmentation
- We propose to use **markers**

Determine markers for each region of interest

$\Rightarrow$

Segment an image

One region
in a segmentation map
$\Uparrow$
One marker

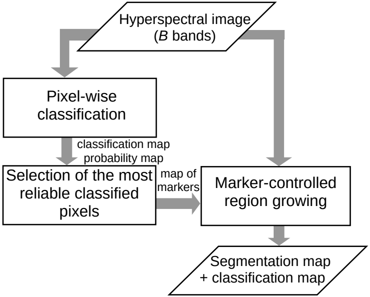
Objective

- Determine markers automatically $\leftarrow$ using results of a pixel-wise classification
- Marker-controlled region growing$\rightarrow$ segment and classify a hyperspectral image

Outline

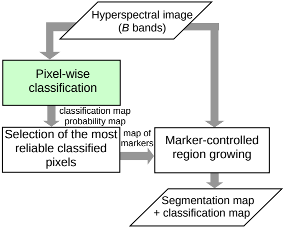
Input

- B-band hyperspectral image $\mathbf{X} = \{\mathbf{x}_j \in \mathbb{R}^B, j = 1, 2, ..., n\}$
- $B \sim 100$

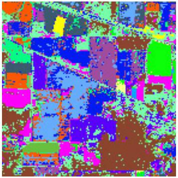


Pixel-wise classification

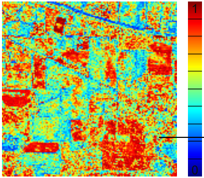
- SVM classifier* → well suited for hyperspectral images
- Output:



classification map



probability map



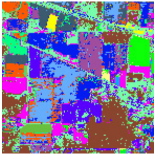
→ probability estimate for each pixel to belong to the assigned class

**C. Chang and C. Lin, "LIBSVM: A library for Support Vector Machines," Software available at http://www.csie.ntu.edu.tw/∼cjlin/libsvm, 2001.*

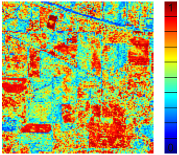
Selection of the most reliable classified pixels

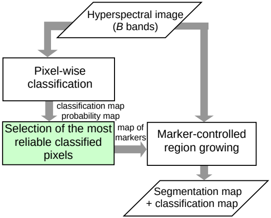
Analysis of classification and probability maps:

classification map



probability map



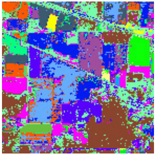


1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%) are used as a marker

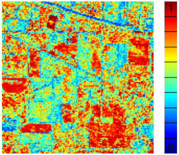
Selection of the most reliable classified pixels

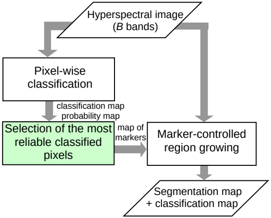
Analysis of classification and probability maps:

classification map



probability map



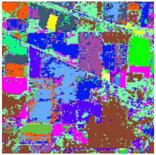


1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%) are used as a marker

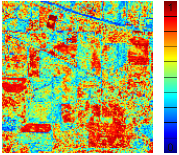
Selection of the most reliable classified pixels

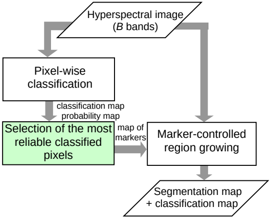
Analysis of classification and probability maps:

classification map



probability map



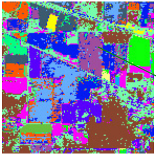


1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%) are used as a marker

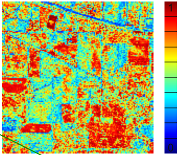
Selection of the most reliable classified pixels

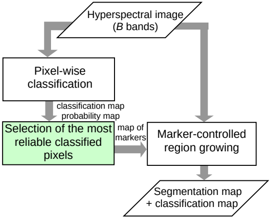
Analysis of classification and probability maps:

classification map



probability map





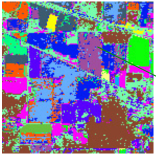
1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) → use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small → its pixels with probabilities $> T\%$ (90%) are used as a marker

Must contain a marker!

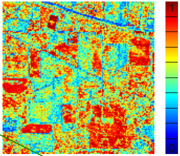
Selection of the most reliable classified pixels

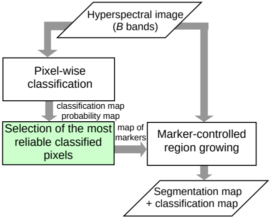
Analysis of classification and probability maps:

classification map



probability map





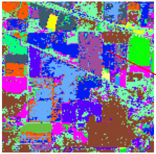
1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) → use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small → its pixels with probabilities > $T\%$ (90%) are used as a marker

Must contain a marker!

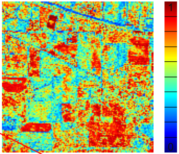
Selection of the most reliable classified pixels

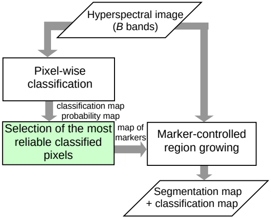
Analysis of classification and probability maps:

classification map



probability map





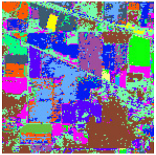
1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%) are used as a marker

Has a marker only if it is very reliable

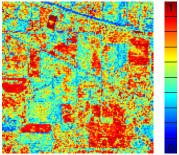
Selection of the most reliable classified pixels

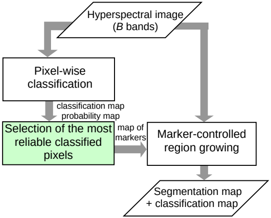
Analysis of classification and probability maps:

classification map



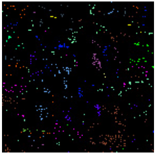
probability map





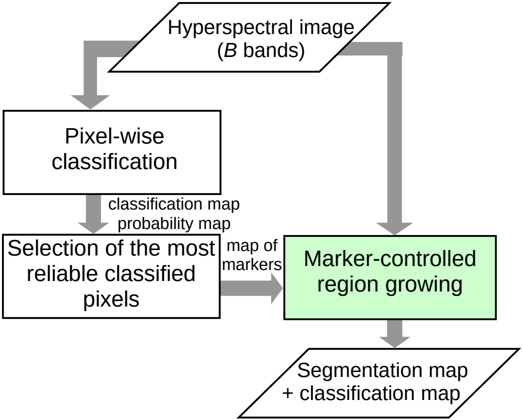
- Each connected component → 1 or 0 marker (2250 regions → 107 markers)
- Marker is not necessarily a connected set of pixels
- Each marker has a class label

map of 107 markers

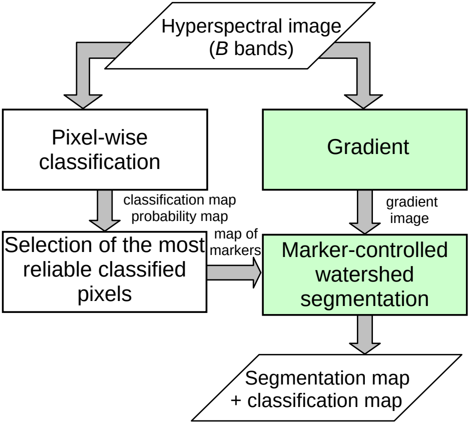


Outline

Marker-controlled region growing

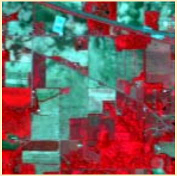


1. Marker-controlled watershed



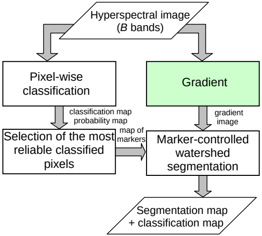
1. Marker-controlled watershed / Gradient

Original hyperspectral image



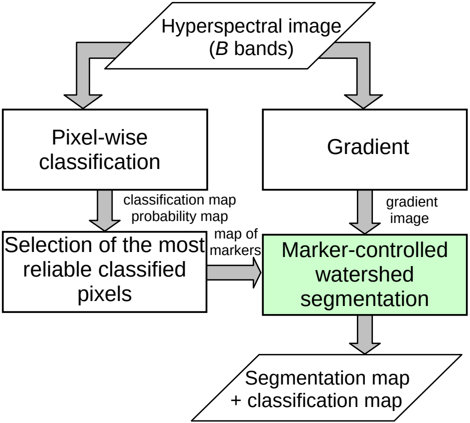
⇒

Robust Color Morphological Gradient*



**Y. Tarabalka et al., "Segmentation and classification of hyperspectral data using watershed," in Proc. of IGARSS'08, Boston, USA, 2008.*

1. Marker-controlled watershed



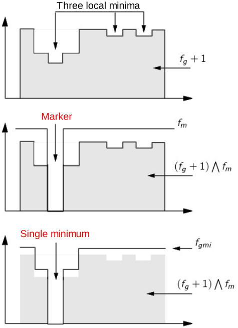
1. Marker-controlled watershed

1. Transform the gradient f_g $\rightarrow$ markers are the only minima
   - Create a marker image:

$$f_m(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \text{ belongs to marker,} \\ t_{max}, & \text{otherwise} \end{cases}$$

   - Compute $(f_g + 1) \bigwedge f_m$
   - Perform minima imposition: morphological reconstruction by erosion of $(f_g + 1) \bigwedge f_m$ from f_m:

$$f_{gmi} = R^{\varepsilon}_{(f_g+1) \bigwedge f_m}(f_m)$$



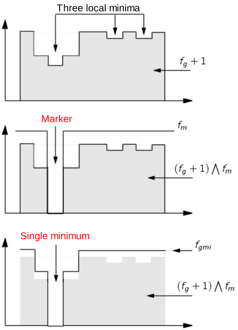
1. Marker-controlled watershed

1. Transform the gradient $f_g \rightarrow$ markers are the only minima
   - Create a marker image:

$$f_m(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \text{ belongs to marker,} \\ t_{max}, & \text{otherwise} \end{cases}$$

   - Compute $(f_g + 1) \bigwedge f_m$
   - Perform minima imposition: morphological reconstruction by erosion of $(f_g + 1) \bigwedge f_m$ from f_m:

$$f_{gmi} = R^{\varepsilon}_{(f_g+1) \bigwedge f_m}(f_m)$$



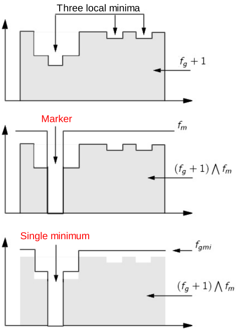
1. Marker-controlled watershed

1. Transform the gradient $f_g \rightarrow$ markers are the only minima
   - Create a marker image:

$$f_m(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \text{ belongs to marker,} \\ t_{max}, & \text{otherwise} \end{cases}$$

   - Compute $(f_g + 1) \bigwedge f_m$
   - Perform minima imposition: morphological reconstruction by erosion of $(f_g + 1) \bigwedge f_m$ from f_m:

$$f_{gmi} = R^{\varepsilon}_{(f_g+1) \bigwedge f_m}(f_m)$$



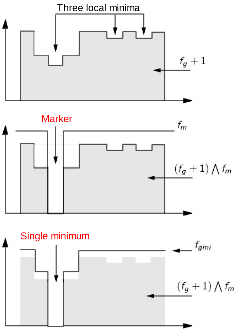
1. Marker-controlled watershed

1. Transform the gradient $f_g \rightarrow$ markers are the only minima
   - Create a marker image:

$$f_m(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \text{ belongs to marker,} \\ t_{max}, & \text{otherwise} \end{cases}$$

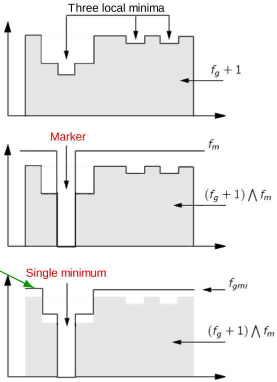
   - Compute $(f_g + 1) \bigwedge f_m$
   - Perform minima imposition: morphological reconstruction by erosion of $(f_g + 1) \bigwedge f_m$ from f_m:

$$f_{gmi} = R^{\varepsilon}_{(f_g+1) \bigwedge f_m}(f_m)$$



1. Marker-controlled watershed

1. Transform the gradient f_g → markers are the only minima
2. Apply watershed on the filtered gradient image f_{gmi} (Vincent and Soille, 1991)

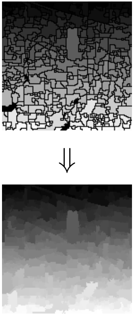


1. Marker-controlled watershed

1. Transform the gradient f_g $\rightarrow$ markers are the only minima
2. Apply watershed on the filtered gradient image f_{gmi} (Vincent and Soille, 1991)

1. Marker-controlled watershed

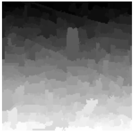
1. Transform the gradient f_g $\rightarrow$ markers are the only minima
2. Apply watershed on the filtered gradient image f_{gmi} (Vincent and Soille, 1991)
3. Assign every watershed pixel to the spectrally most similar neighboring region



1. Marker-controlled watershed

1. Transform the gradient f_g $\rightarrow$ markers are the only minima
2. Apply watershed on the filtered gradient image f_{gmi} (Vincent and Soille, 1991)
3. Assign every watershed pixel to the spectrally most similar neighboring region

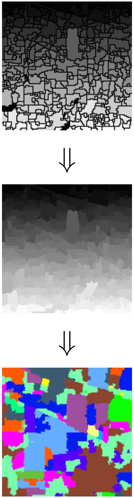
$\Downarrow$



 $\rightarrow$ Several minima in the filtered gradient $\rightarrow$ Several regions in the segmentation map

1. Marker-controlled watershed

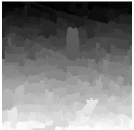
1. Transform the gradient f_g → markers are the only minima
2. Apply watershed on the filtered gradient image f_{gmi} (Vincent and Soille, 1991)
3. Assign every watershed pixel to the spectrally most similar neighboring region
4. Merge regions belonging to the same marker



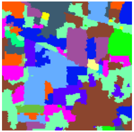
1. Marker-controlled watershed

1. Transform the gradient $f_g \rightarrow$ markers are the only minima
2. Apply watershed on the filtered gradient image f_{gmi} (Vincent and Soille, 1991)
3. Assign every watershed pixel to the spectrally most similar neighboring region
4. Merge regions belonging to the same marker
5. Class of each marker $\rightarrow$ class of the corresponding region

$\Downarrow$

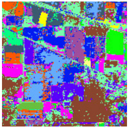


$\Downarrow$

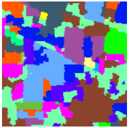


Classification maps & classification accuracies (%)

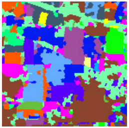
SVM



Markers(107reg.)



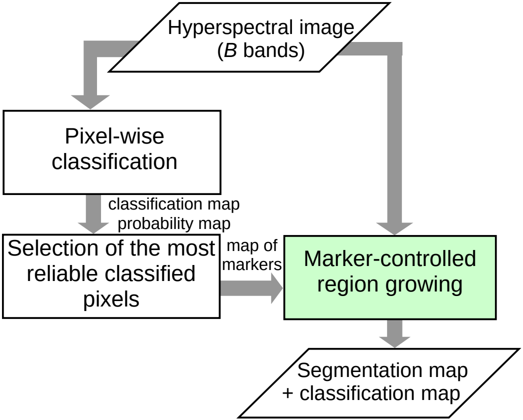
NoMarker*(1277reg.)



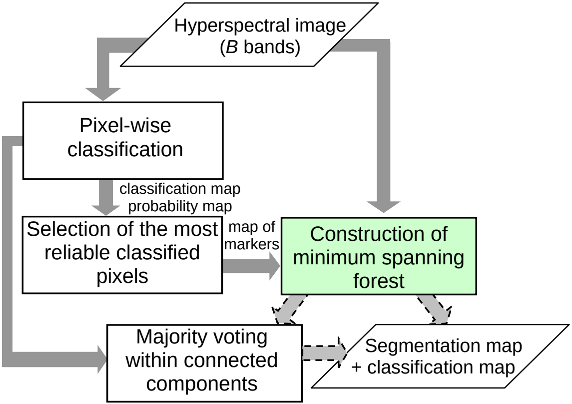
IGARSS'08

	SVM	Markers	NoMarker*
Overall Accuracy	78.17	85.99	**86.63**
Average Accuracy	85.97	86.95	**91.61**
Kappa Coefficient κ	75.33	83.98	**84.83**
Corn-no till	78.18	80.35	**94.22**
Corn-min till	69.64	71.94	**78.06**
Corn	**91.85**	73.37	88.59
Soybeans-no till	82.03	**98.91**	96.30
Soybeans-min till	58.95	**80.48**	68.82
Soybeans-clean till	87.94	84.75	**90.78**
Alfalfa	74.36	**94.87**	**94.87**
Grass/pasture	92.17	**95.30**	95.08
Grass/trees	91.68	92.97	**97.99**
Grass/pasture-mowed	**100**	**100**	**100**
Hay-windrowed	97.72	**99.54**	**99.54**
Oats	**100**	**100**	**100**
Wheat	98.77	**99.38**	**99.38**
Woods	93.01	**99.36**	97.11
Bldg-Grass-Tree-Drives	61.52	55.45	**69.39**
Stone-steel towers	**97.78**	64.44	95.56

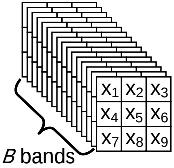
Marker-controlled region growing



2. Construction of a Minimum Spanning Forest (MSF)



2. Construction of a Minimum Spanning Forest (MSF)



map of markers

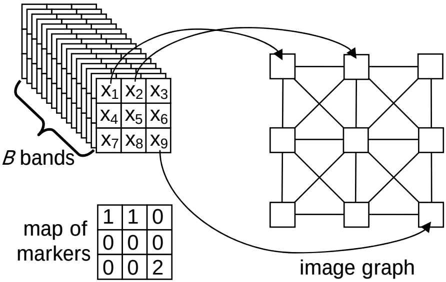
1	1	0
0	0	0
0	0	2

1) Map an image onto a graph

- Weight $w_{i,j}$ indicates the degree of dissimilarity between pixels $\mathbf{x}_i$ and $\mathbf{x}_j$. Spectral Angle Mapper (SAM) distance can be used:

$$w_{i,j} = SAM(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\sum_{b=1}^{B} x_{ib} x_{jb}}{[\sum_{b=1}^{B} x_{ib}^2]^{1/2} [\sum_{b=1}^{B} x_{jb}^2]^{1/2}}\right)$$

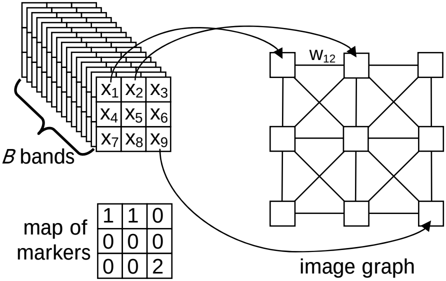
2. Construction of a Minimum Spanning Forest (MSF)



1) Map an image onto a graph
   - Weight $w_{i,j}$ indicates the degree of dissimilarity between pixels $\mathbf{x}_i$ and $\mathbf{x}_j$. Spectral Angle Mapper (SAM) distance can be used:

$$w_{i,j} = SAM(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\sum_{b=1}^{B} x_{ib}x_{jb}}{[\sum_{b=1}^{B} x_{ib}^2]^{1/2}[\sum_{b=1}^{B} x_{jb}^2]^{1/2}}\right)$$

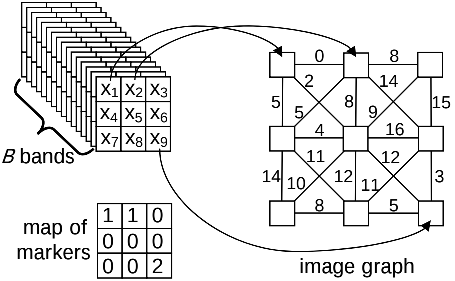
2. Construction of a Minimum Spanning Forest (MSF)



1) Map an image onto a graph
   - Weight $w_{i,j}$ indicates the degree of dissimilarity between pixels $\mathbf{x}_i$ and $\mathbf{x}_j$. Spectral Angle Mapper (SAM) distance can be used:

$$w_{i,j} = SAM(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\sum_{b=1}^{B} x_{ib}x_{jb}}{[\sum_{b=1}^{B} x_{ib}^2]^{1/2}[\sum_{b=1}^{B} x_{jb}^2]^{1/2}}\right)$$

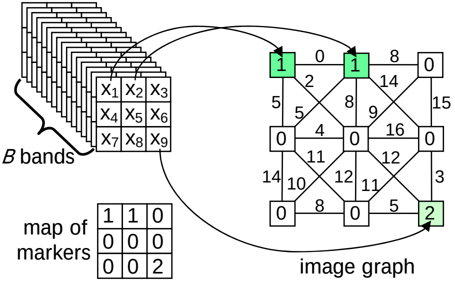
2. Construction of a Minimum Spanning Forest (MSF)



1) Map an image onto a graph
- Weight $w_{i,j}$ indicates the degree of dissimilarity between pixels $\mathbf{x}_i$ and $\mathbf{x}_j$. Spectral Angle Mapper (SAM) distance can be used:

$$w_{i,j} = SAM(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\sum_{b=1}^{B} x_{ib}x_{jb}}{[\sum_{b=1}^{B} x_{ib}^2]^{1/2}[\sum_{b=1}^{B} x_{jb}^2]^{1/2}}\right)$$

2. Construction of a Minimum Spanning Forest (MSF)



1) Map an image onto a graph
   - Weight $w_{i,j}$ indicates the degree of dissimilarity between pixels $\mathbf{x}_i$ and $\mathbf{x}_j$. Spectral Angle Mapper (SAM) distance can be used:

$$w_{i,j} = SAM(\mathbf{x}_i, \mathbf{x}_j) = \arccos\left(\frac{\sum_{b=1}^{B} x_{ib}x_{jb}}{[\sum_{b=1}^{B} x_{ib}^2]^{1/2}[\sum_{b=1}^{B} x_{jb}^2]^{1/2}}\right)$$

2. Construction of a Minimum Spanning Forest (MSF)

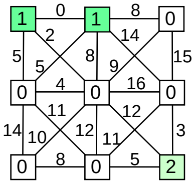
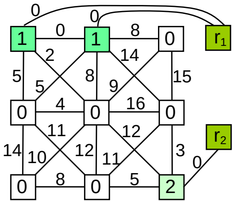


image graph

Given a graph G, a **MSF** F^* rooted on vertices $\{r_1, \ldots, r_m\}$ is:

- a non-connected graph without cycles
- contains all the vertices of G
- consists of connected subgraphs, each subgraph (tree) contains (is rooted on) one root r_i
- sum of the edges weights of F^* is minimal

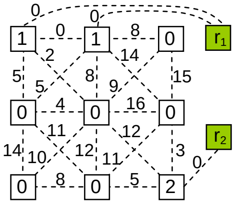
2. Construction of a Minimum Spanning Forest (MSF)



modified graph

2) Add m extra vertices r_i, $i = 1, \ldots, m$ corresponding to m markers

2. Construction of a Minimum Spanning Forest (MSF)

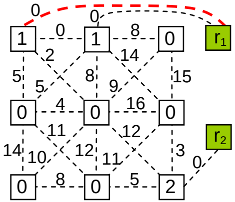


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

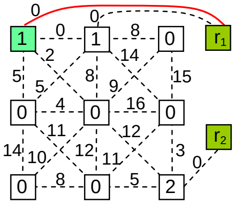


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

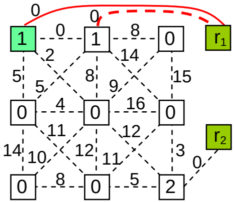


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

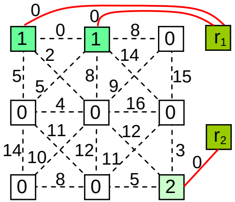


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

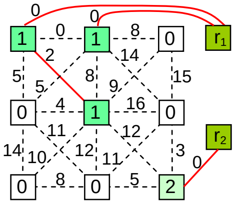


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

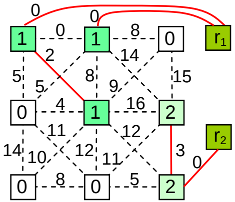


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

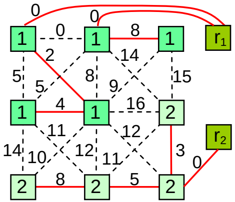


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

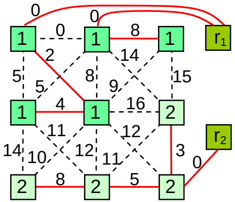


3) Construct a MSF $F^* = (V^*, E^*)$

Initialization: $V^* = \{r_1, r_2, ..., r_m\}$ (roots are in the forest)

1. Choose edge of the modified graph e_{ij} with minimal weight such that $i \in V^*$ and $j \notin V^*$
2. $V^* = V^* \cup \{j\}$, $E^* = E^* \cup \{e_{i,j}\}$
3. If $V^* \neq V$, go to 1

2. Construction of a Minimum Spanning Forest (MSF)

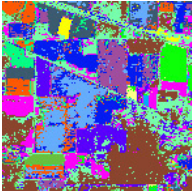


3) Construct a MSF $F^* = (V^*, E^*)$

4) Class of each marker $\rightarrow$ class of the corresponding region
(of all the pixels grown from this marker)

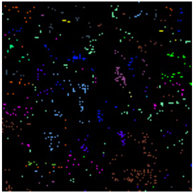
2. Construction of a Minimum Spanning Forest (MSF)

Pixel-wise classification map



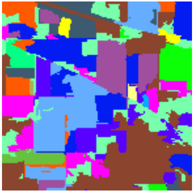
⇒

Map of 107 markers



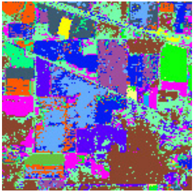
⇒

MSF-based classification map



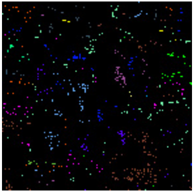
2. Construction of a Minimum Spanning Forest (MSF)

Pixel-wise classification map



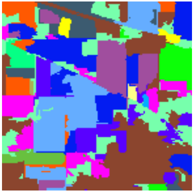
⇒

Map of 107 markers



⇒

MSF-based classification map

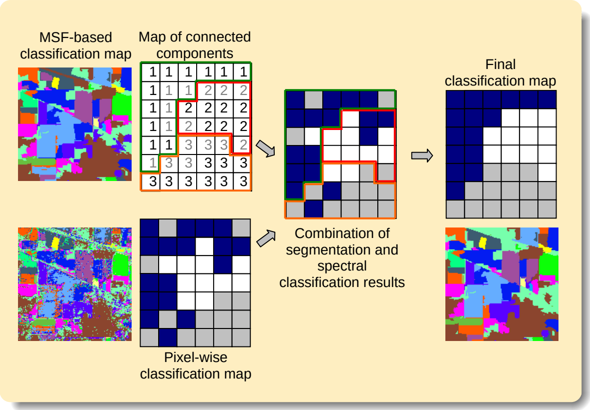


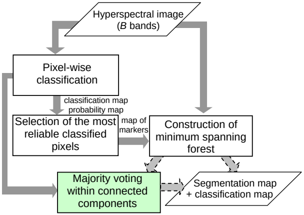
If a marker is classified to the wrong class

⇒

The whole region grown from this marker risks to be wrongly classified!

2. Construction of a Minimum Spanning Forest / Post-processing





Classification accuracies (%)

	SVM	MSF	MSF+MV	M-WHED*	WHED+MV**
Overall Accuracy	78.17	88.41	**91.80**	85.99	86.63
Average Accuracy	85.97	91.57	**94.28**	86.95	91.61
Kappa Coefficient κ	75.33	86.71	**90.64**	83.98	84.83
Corn-no till	78.18	90.97	93.21	80.35	**94.22**
Corn-min till	69.64	69.52	**96.56**	71.94	78.06
Corn	91.85	**95.65**	**95.65**	73.37	88.59
Soybeans-no till	82.03	98.04	93.91	**98.91**	96.30
Soybeans-min till	58.95	**81.97**	**81.97**	80.48	68.82
Soybeans-clean till	87.94	85.99	**97.16**	84.75	90.78
Alfalfa	74.36	**94.87**	**94.87**	**94.87**	**94.87**
Grass/pasture	92.17	94.63	94.63	**95.30**	95.08
Grass/trees	91.68	92.40	97.27	92.97	**97.99**
Grass/pasture-mowed	**100**	**100**	**100**	**100**	**100**
Hay-windrowed	97.72	**99.77**	**99.77**	99.54	99.54
Oats	**100**	**100**	**100**	**100**	**100**
Wheat	98.77	**99.38**	**99.38**	**99.38**	**99.38**
Woods	93.01	97.59	**99.68**	99.36	97.11
Bldg-Grass-Tree-Drives	61.52	68.79	68.79	55.45	**69.39**
Stone-steel towers	**97.78**	95.56	95.56	64.44	95.56

* *Tarabalka et al., IGARSS'09* ** *Tarabalka et al., IGARSS'08*

Classification using markers for the Pavia image

Original image

$\Rightarrow$

Classification using MSF

Classification accuracies for the Pavia image (%):

	SVM	+WHED	+Part.Cl.	+HSEG		MSF	MSF+MV
SCW				0.0	0.1		
Overall Acc.	81.01	85.42	93.59	90.00	**93.85**	84.14	91.08
Average Acc.	88.25	91.31	94.39	94.15	**97.07**	92.35	94.76
Kappa Coef.κ	75.86	81.30	91.48	86.86	**91.89**	79.71	88.30
Asphalt	84.93	93.64	90.72	73.33	**94.77**	93.05	93.16
Meadows	70.79	75.09	**92.73**	88.73	89.32	72.30	85.65
Gravel	67.16	66.12	82.09	**97.47**	96.14	89.15	89.15
Trees	97.77	98.56	**99.21**	98.45	98.08	87.02	91.24
Metal sheets	99.46	99.91	**100**	99.10	99.82	99.91	99.91
Bare soil	92.83	97.35	96.78	98.43	99.76	97.11	**99.91**
Bitumen	90.42	96.23	92.46	95.92	**100**	98.57	98.57
Bricks	92.78	97.92	97.80	98.81	**99.29**	95.66	99.05
Shadows	98.11	96.98	97.74	97.11	96.48	**98.36**	96.23

Outline

1. Classification using Minimum Spanning Forest grown from automatically selected markers:
   - significantly decreases oversegmentation
   - improves classification accuracies
   - provides classification maps with homogeneous regions
2. Robustness of the parameters settings for the marker selection procedure has been experimentally proved

Conclusions

1. Several schemes for spectral-spatial classification of hyperspectral images are proposed and investigated
2. The developed techniques:
    - significantly decrease oversegmentation
    - improve classification accuracies
    - provide classification maps with more homogeneous regions

   when compared to the previously proposed classification methods
3. Classification using MSF gives the best or close to the best classification accuracies for all the tested images

Perspectives

1. Further develop marker-based methods
   - investigate parameter estimation techniques
   - develop new similarity measures
2. Apply and adapt the proposed methods for analysis of multivariate and multisource data in other types of applications
   - medical imaging

Thank you for your attention!