Classification of Hyperspectral Images Using Automatic Marker Selection and Minimum Spanning Forest

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

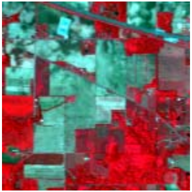
[1]GIPSA-Lab, Grenoble Institute of Technology, France
[2]University of Iceland, Reykjavik, Iceland
e-mail: yuliya.tarabalka@hyperinet.eu

August 26, 2009

Outline

Classification problem

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



Ground-truth data

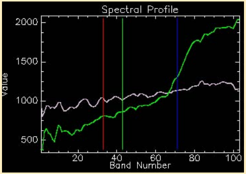
Task

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

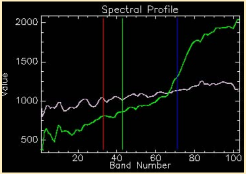
- Spectrum of each pixel is analyzed
- Directly accessible
- Kernel-based methods (e.g. SVM)
  $\rightarrow$ good classification results



Classification approaches

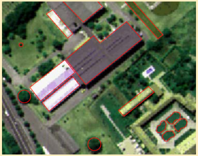
Only spectral information

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



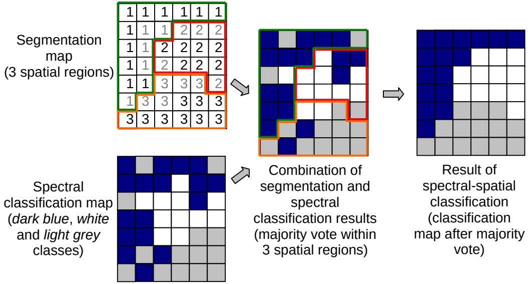
Spectral + spatial information

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



Our previous research

- Segment a hyperspectral image = find an exhaustive partitioning of the image into homogeneous regions
- Spectral info + spatial info → classify image (majority vote within each region)



Unsupervised segmentation

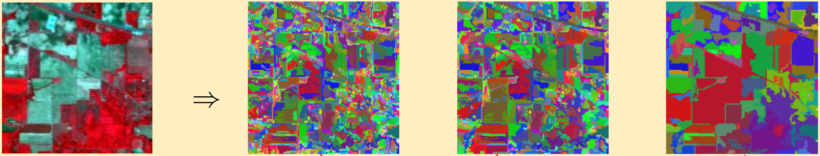
- Unsupervised segmentation = exhaustive partitioning into homogeneous regions
- How to define a measure of homogeneity?

Unsupervised segmentation

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

Hierarchical segmentation (*Tilton, IGARSS'98*)

Image ⇒ 2381reg. 1389reg. 280reg.



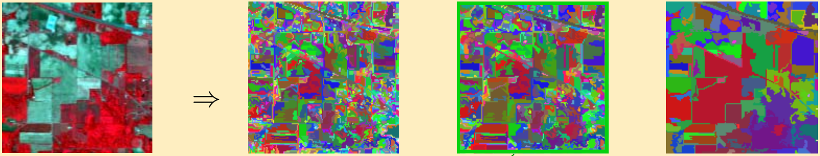
Oversegmentation! Undersegmentation?

Unsupervised segmentation

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

Hierarchical segmentation (*Tilton, IGARSS'98*)

Image ⇒ 2381reg. 1389reg. 280reg.



Oversegmentation Undersegmentation?

⇓

Preferred in previous works → not to lose objects

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

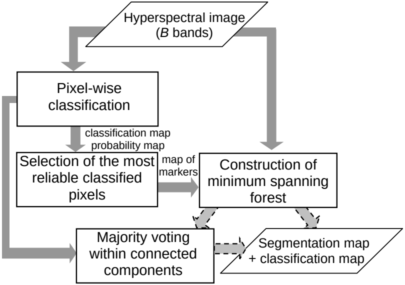
One region
in a segmentation map
$\Uparrow$
One marker

Objective

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

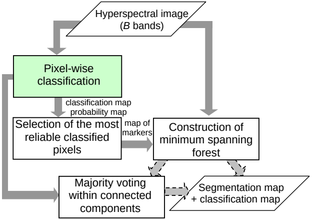
Input

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

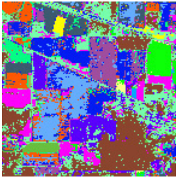


Pixel-wise classification

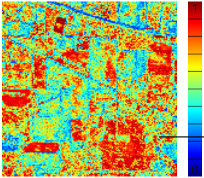
- SVM classifier* → well suited for hyperspectral images
- Output:



classification map



probability map



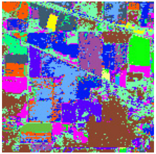
probability estimate for each pixel to belong to the assigned class

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

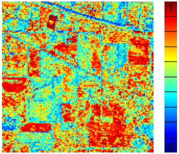
Selection of the most reliable classified pixels

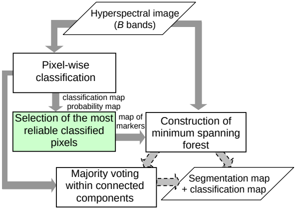
Analysis of classification and probability maps:

classification map



probability map



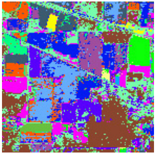


1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%)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

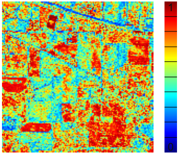
Selection of the most reliable classified pixels

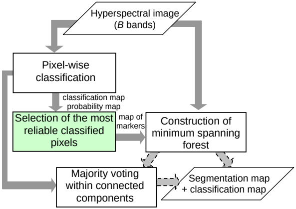
Analysis of classification and probability maps:

classification map



probability map



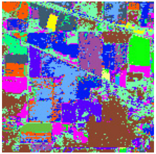


1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) → use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small → its pixels with probabilities > $T\%$ (90%) are used as a marker

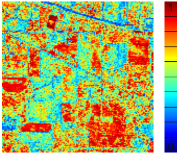
Selection of the most reliable classified pixels

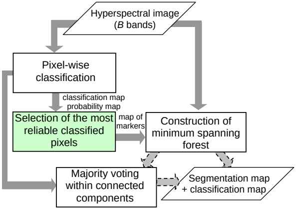
Analysis of classification and probability maps:

classification map



probability map



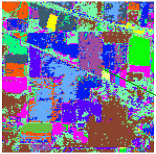


1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) → use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small → its pixels with probabilities > $T\%$ (90%) are used as a marker

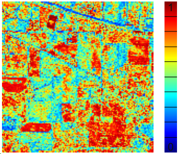
Selection of the most reliable classified pixels

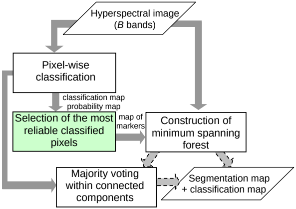
Analysis of classification and probability maps:

classification map



probability map





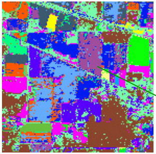
1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%)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

Must contain a marker!

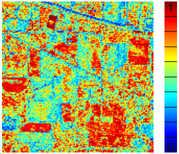
Selection of the most reliable classified pixels

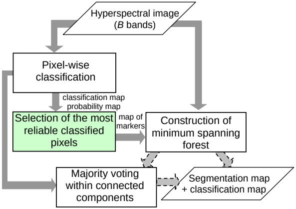
Analysis of classification and probability maps:

classification map



probability map





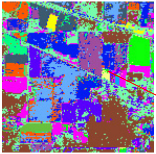
1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%)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

Must contain a marker!

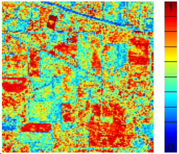
Selection of the most reliable classified pixels

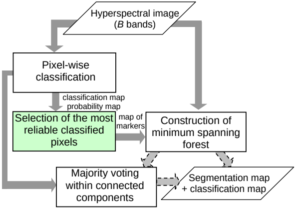
Analysis of classification and probability maps:

classification map



probability map





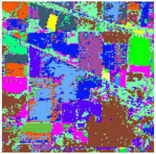
1. Perform connected components labeling of the classification map
2. Analyse each connected component:
   - If it is large (> 20 pixels) $\rightarrow$ use $P\%$ (5%) of its pixels with the highest probabilities as a marker
   - If it is small $\rightarrow$ its pixels with probabilities $> T\%$ (90%)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

Has a marker only if it is very reliable

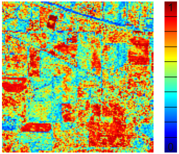
Selection of the most reliable classified pixels

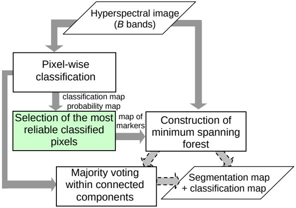
Analysis of classification and probability maps:

classification map



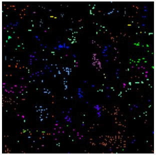
probability map



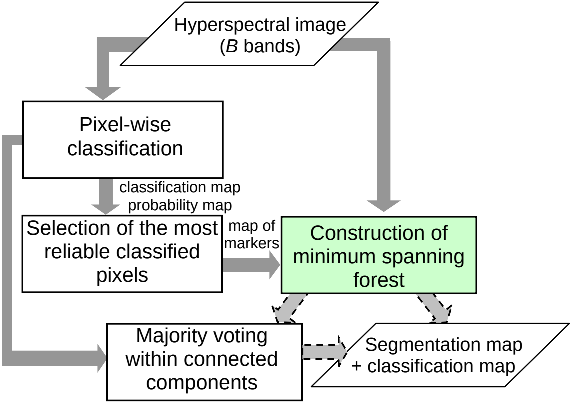


- Each connected component → 1 or 0 marker (2250 regions → 107 markers)
- Marker is not necessarily a connected set of pixels
- Each marker has a class label

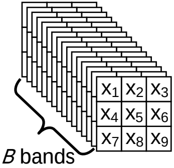
map of 107 markers



Construction of a Minimum Spanning Forest (MSF)



Construction of a Minimum Spanning Forest (MSF)



map of markers

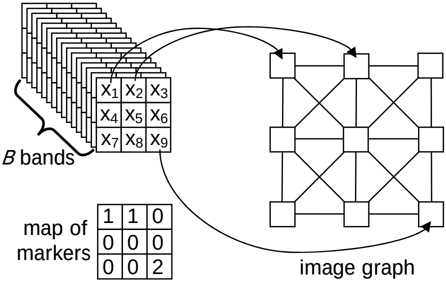
1	1	0
0	0	0
0	0	2

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:

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

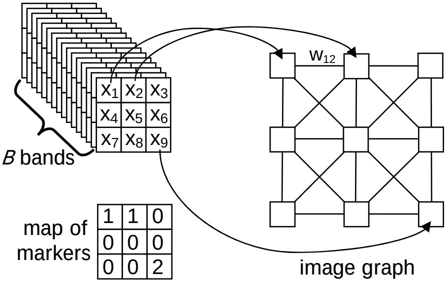
Construction of a Minimum Spanning Forest (MSF)



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:

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

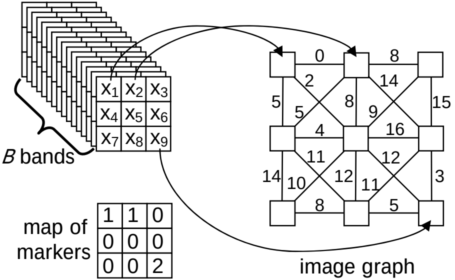
Construction of a Minimum Spanning Forest (MSF)



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:

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

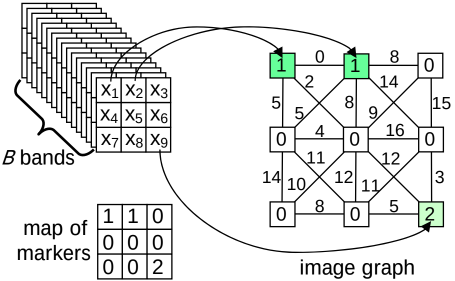
Construction of a Minimum Spanning Forest (MSF)



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:

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

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:

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

Construction of a Minimum Spanning Forest (MSF)

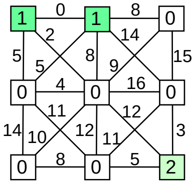
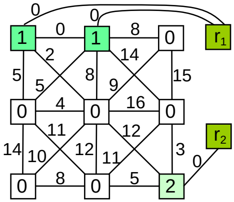


image graph

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

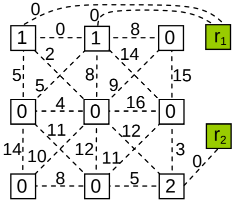
Construction of a Minimum Spanning Forest (MSF)



modified graph

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

Construction of a Minimum Spanning Forest (MSF)

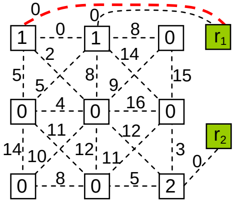


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

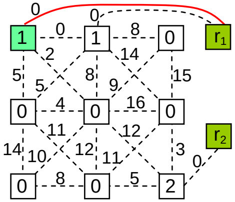


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

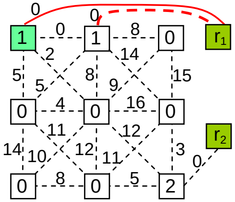


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

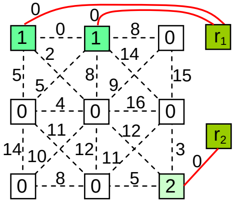


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

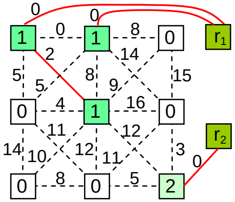


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

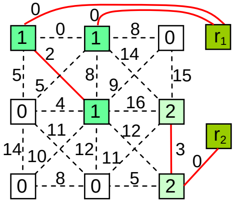


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

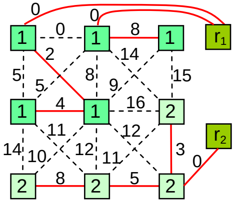


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

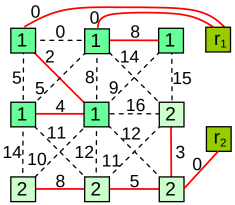


3) Construct a MSF $F^* = (V^*, E^*)$

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

Construction of a Minimum Spanning Forest (MSF)

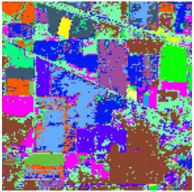


3) Construct a MSF $F^* = (V^*, E^*)$

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

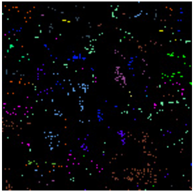
Construction of a Minimum Spanning Forest (MSF)

Pixel-wise classification map



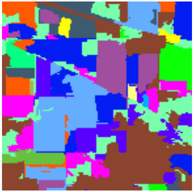
⇒

Map of 107 markers



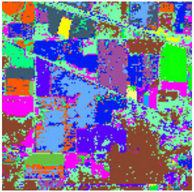
⇒

MSF-based classification map



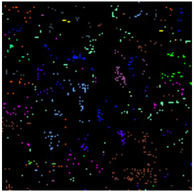
Construction of a Minimum Spanning Forest (MSF)

Pixel-wise classification map



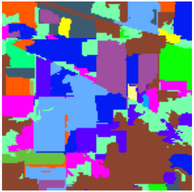
⇒

Map of 107 markers



⇒

MSF-based classification map

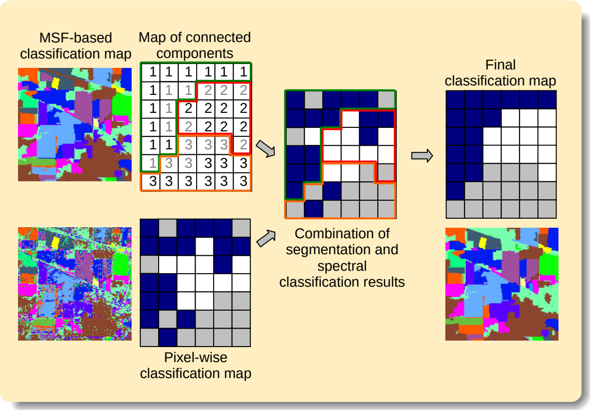


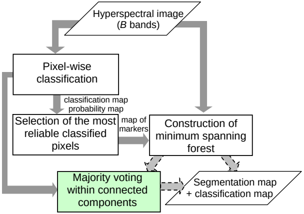
If a marker is classified to the wrong class

⇒

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

Majority voting within connected components





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*

Conclusions and perspectives

Conclusions

1. Method for automatic selection of markers is proposed
2. Scheme for segmentation and classification of hyperspectral images is developed
3. The proposed method:
   - significantly decreases oversegmentation
   - improves classification accuracies
   - provides classification maps with homogeneous regions

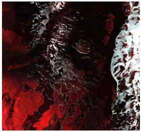
Perspectives

- Use marker selection + other image segmentation methods

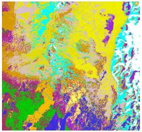
Thank you for your attention!

Classification of the Hekla image

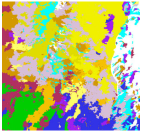
AVIRIS image
[560 × 600 × 157]



SVM



MSF+MV



Classification accuracies (%)

	SVM	MSF	MSF+MV
Overall Accuracy	88.56	90.34	**98.96**
Average Accuracy	89.44	94.89	**98.45**
Kappa Coefficient κ	86.91	89.04	**98.80**
Andesite lava 1970	88.36	**100**	**100**
Andesite lava 1980 I	87.25	92.11	**100**
Andesite lava 1980 II	88.24	96.96	**99.86**
Andesite lava 1991 I	84.94	73.19	**99.55**
Andesite lava 1991 II	**93.33**	88.89	88.89
Andesite lava with moss cover	94.24	**98.46**	**98.46**
Hyaloclastite formation	87.54	99.53	**99.68**
Lava covered with tephra/scoria	91.69	95.08	**97.38**
Rhyolite	85.88	96.89	**100**
Scoria	74.20	**97.60**	**97.60**
Firn and glacier ice	**100**	**100**	**100**
Snow	97.59	**100**	**100**