Remote Sensing & Photogrammetry W2

Beata Hejmanowska
Building C4, room 212,
phone: +4812 617 22 72
605 061 510
galia@agh.edu.pl

LANDSAT

- **LANDSAT Orbit**
- **LANDSAT 4,5 MSS Sensor Characteristics**
- **LANDSAT TM, ETM+ Sensor Characteristics**

Type	Sun-Synchronous
Altitude	705 km
Inclination	98.2 deg
Period	99 min
Repeat Cycle	16 days

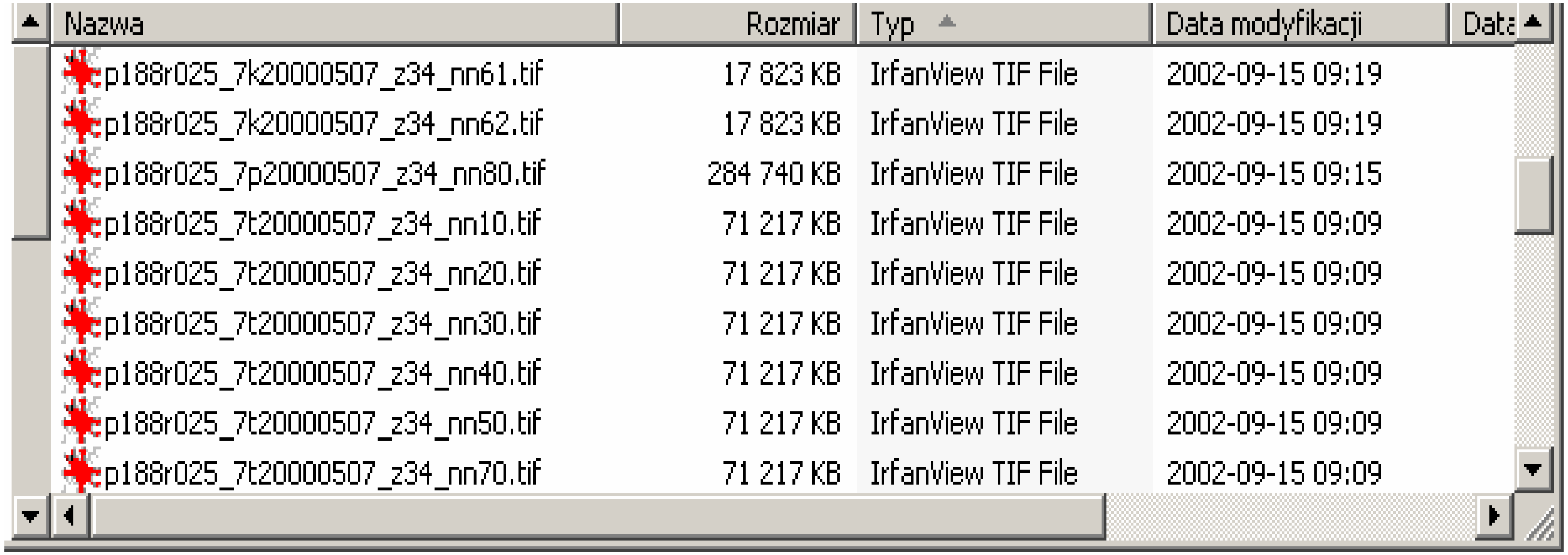
	Band	**Wavelength (µm)**	**Resolution (m)**
Blue	**1**	0.45 - 0.52	30
Green	**2**	0.52 - 0.60	30
Red	**3**	0.63 - 0.69	30
Near IR	**4**	0.76 - 0.90	30
SWIR	**5**	1.55 - 1.75	30
Thermal IR	**6**	10.40 - 12.50	120 (TM) 60 (ETM+)
SWIR	**7**	2.08 - 2.35	30
Panchromatic		0.5 - 0.9	15

	Band	**Wavelength (µm)**	**Resolution (m)**
Green	**1**	0.5 - 0.6	82
Red	**2**	0.6 - 0.7	82
Near IR	**3**	0.7 - 0.8	82
Near IR	**4**	0.8 - 1.1	82

Landsat Data is available for FREE

- Path: 188, Row: 25
- elp188r025_7t20000507.tar.gz
- GEOTIF

	Band	Wavelength (µm)	Resolution (m)
Blue	**1**	0.45 - 0.52	30
Green	**2**	0.52 - 0.60	30
Red	**3**	0.63 - 0.69	30
Near IR	**4**	0.76 - 0.90	30
SWIR	**5**	1.55 - 1.75	30
Thermal IR	**6**	10.40 - 12.50	120 (TM) 60 (ETM+)
SWIR	**7**	2.08 - 2.35	30
Panchromatic		0.5 - 0.9	15



Nazwa	Rozmiar	Typ	Data modyfikacji	Data
p188r025_7k20000507_z34_nn61.tif	17 823 KB	IrfanView TIF File	2002-09-15 09:19	
p188r025_7k20000507_z34_nn62.tif	17 823 KB	IrfanView TIF File	2002-09-15 09:19	
p188r025_7p20000507_z34_nn80.tif	284 740 KB	IrfanView TIF File	2002-09-15 09:15	
p188r025_7t20000507_z34_nn10.tif	71 217 KB	IrfanView TIF File	2002-09-15 09:09	
p188r025_7t20000507_z34_nn20.tif	71 217 KB	IrfanView TIF File	2002-09-15 09:09	
p188r025_7t20000507_z34_nn30.tif	71 217 KB	IrfanView TIF File	2002-09-15 09:09	
p188r025_7t20000507_z34_nn40.tif	71 217 KB	IrfanView TIF File	2002-09-15 09:09	
p188r025_7t20000507_z34_nn50.tif	71 217 KB	IrfanView TIF File	2002-09-15 09:09	
p188r025_7t20000507_z34_nn70.tif	71 217 KB	IrfanView TIF File	2002-09-15 09:09	

Image processing

- Pre-processing – later will be explained
- **Image enhancement**
- Data extraction – later will be explained

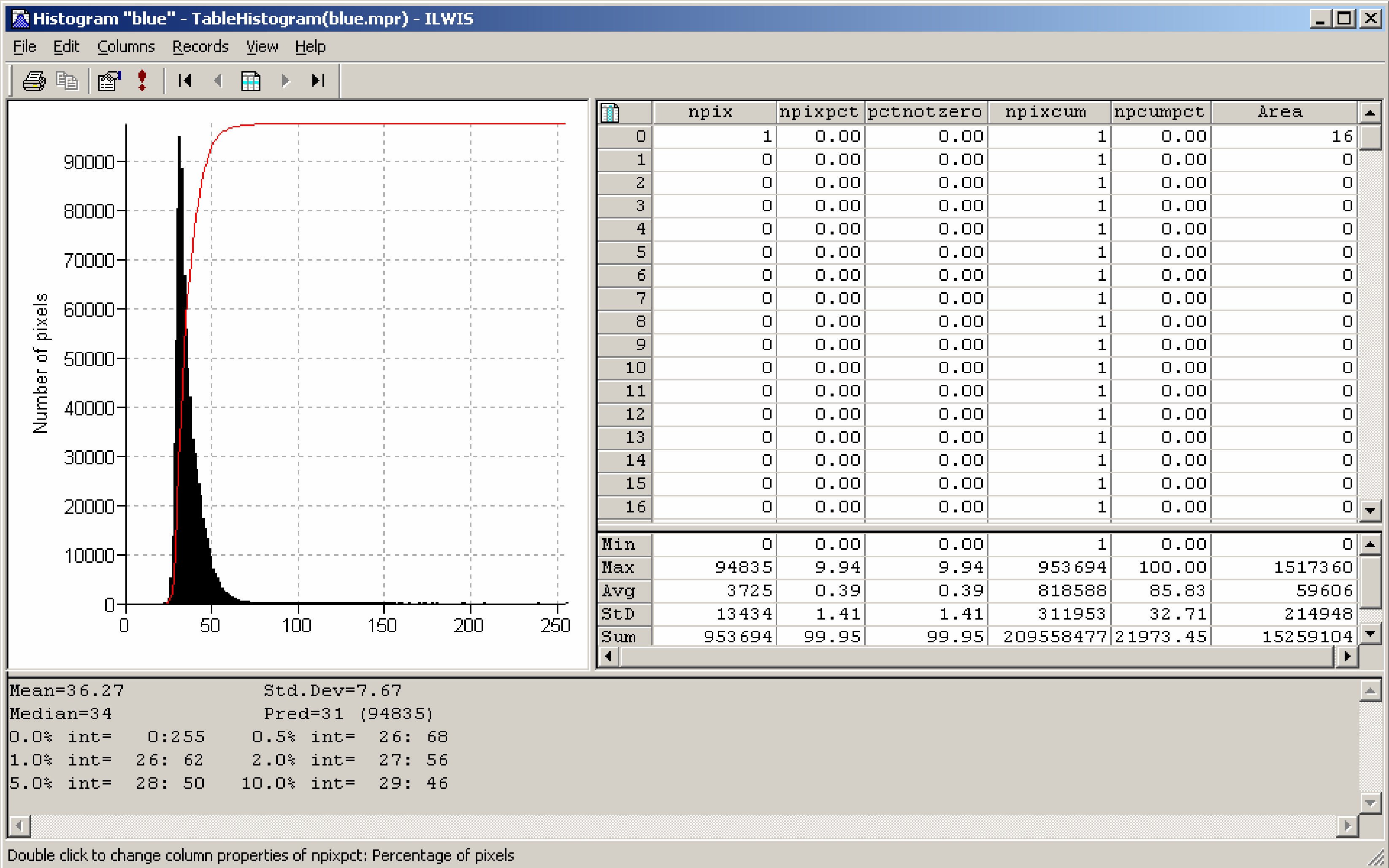
Digital image

- Histogram
- Contrast stretching
- Color composite

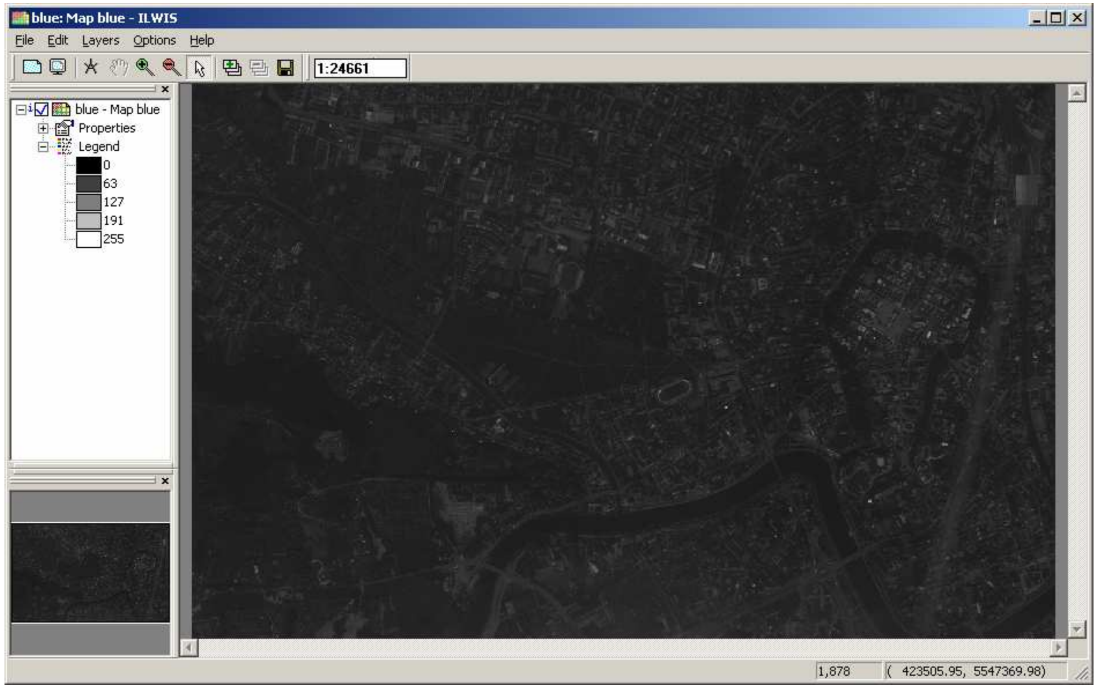
- Data:
  - IKONOS (B, G, R, IR, PAN)
  - LANDSAT (TM1, TM2, TM3, TM4, TM5, TM6, TM7, TM8)

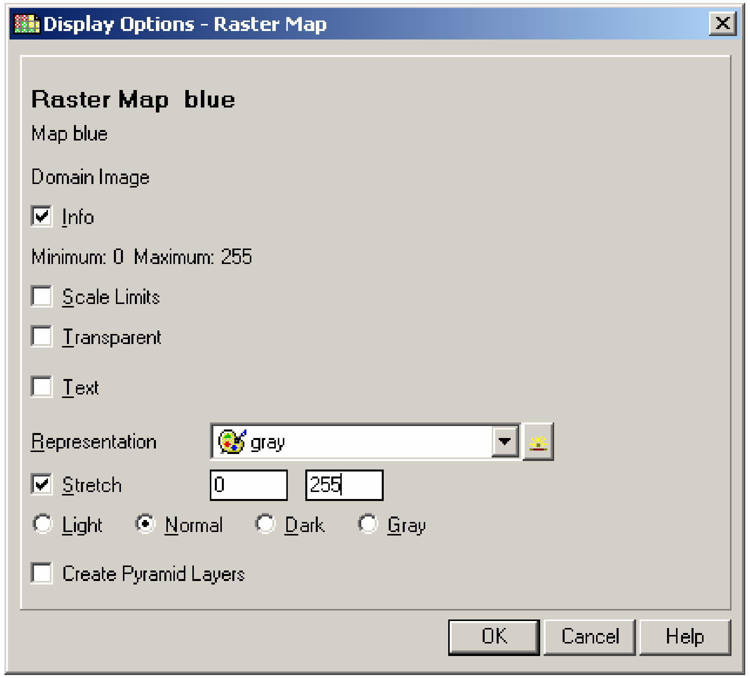
Digital image

Histogram

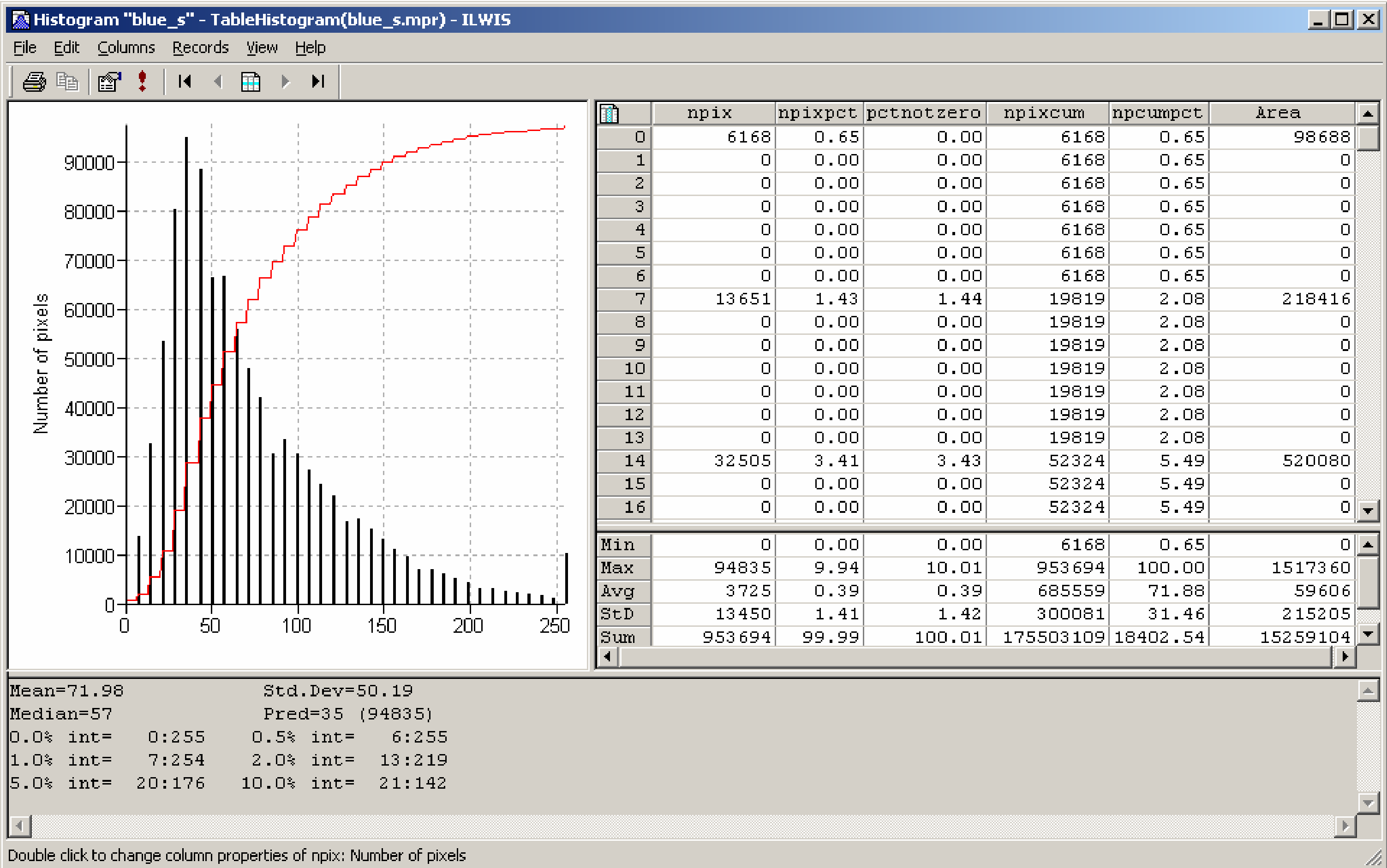

Histogram "blue" - TableHistogram(blue.mpr) - ILWIS
File Edit Columns Records View Help
Number of pixels
90000
80000
70000
60000
50000
40000
30000
20000
10000
0
0 50 100 150 200 250
npix npixpct pctnotzero npixcum npcumpct Area
0 1 0.00 0.00 1 0.00 16
1 0 0.00 0.00 1 0.00 0
2 0 0.00 0.00 1 0.00 0
3 0 0.00 0.00 1 0.00 0
4 0 0.00 0.00 1 0.00 0
5 0 0.00 0.00 1 0.00 0
6 0 0.00 0.00 1 0.00 0
7 0 0.00 0.00 1 0.00 0
8 0 0.00 0.00 1 0.00 0
9 0 0.00 0.00 1 0.00 0
10 0 0.00 0.00 1 0.00 0
11 0 0.00 0.00 1 0.00 0
12 0 0.00 0.00 1 0.00 0
13 0 0.00 0.00 1 0.00 0
14 0 0.00 0.00 1 0.00 0
15 0 0.00 0.00 1 0.00 0
16 0 0.00 0.00 1 0.00 0
Min 0 0.00 0.00 1 0.00 0
Max 94835 9.94 9.94 953694 100.00 1517360
Avg 3725 0.39 0.39 818588 85.83 59606
StD 13434 1.41 1.41 311953 32.71 214948
Sum 953694 99.95 99.95 209558477 21973.45 15259104
Mean=36.27 Std.Dev=7.67
Median=34 Pred=31 (94835)
0.0% int= 0:255 0.5% int= 26: 68
1.0% int= 26: 62 2.0% int= 27: 56
5.0% int= 28: 50 10.0% int= 29: 46
Double click to change column properties of npixpct: Percentage of pixels

Visualisation without contrast stretching

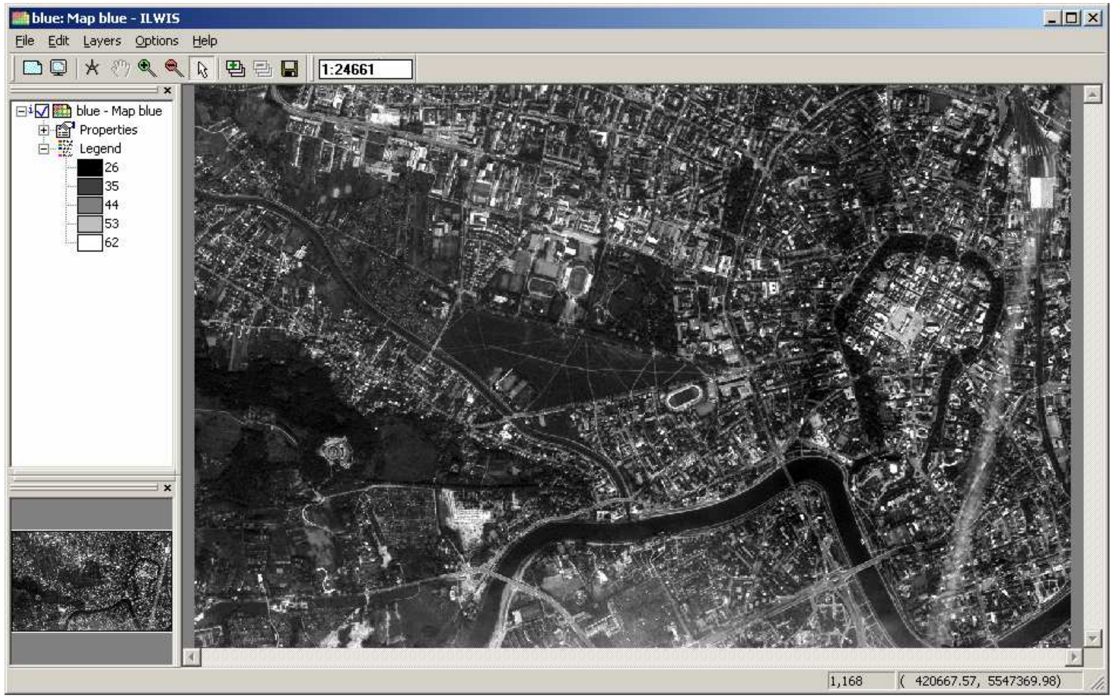


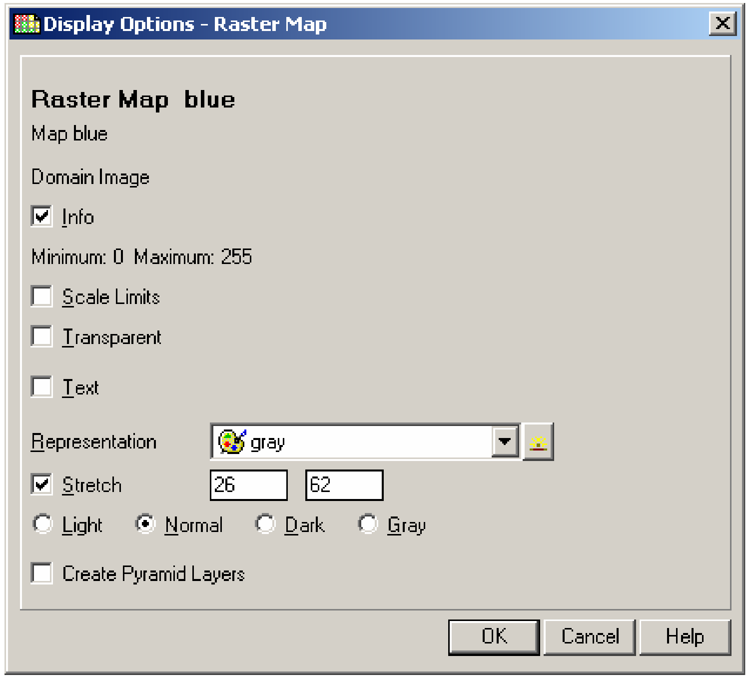


Histogram stretchnig

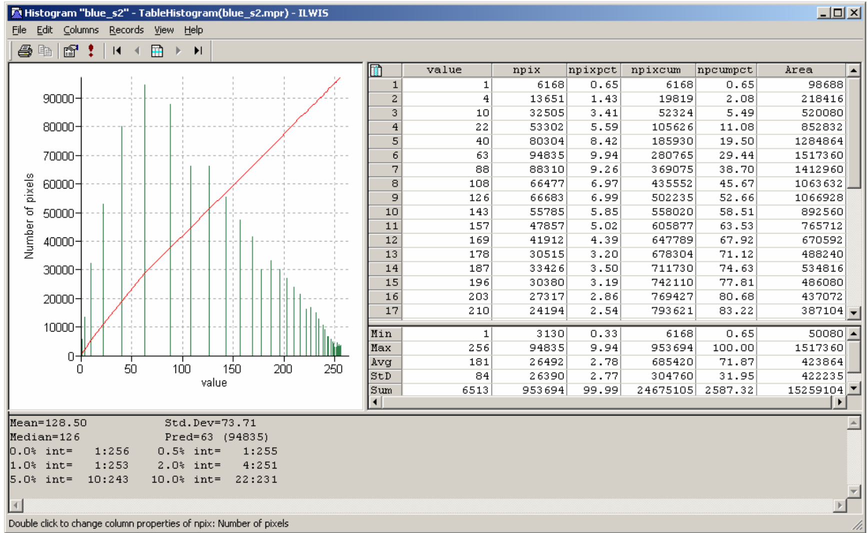


Visualisation after stretching

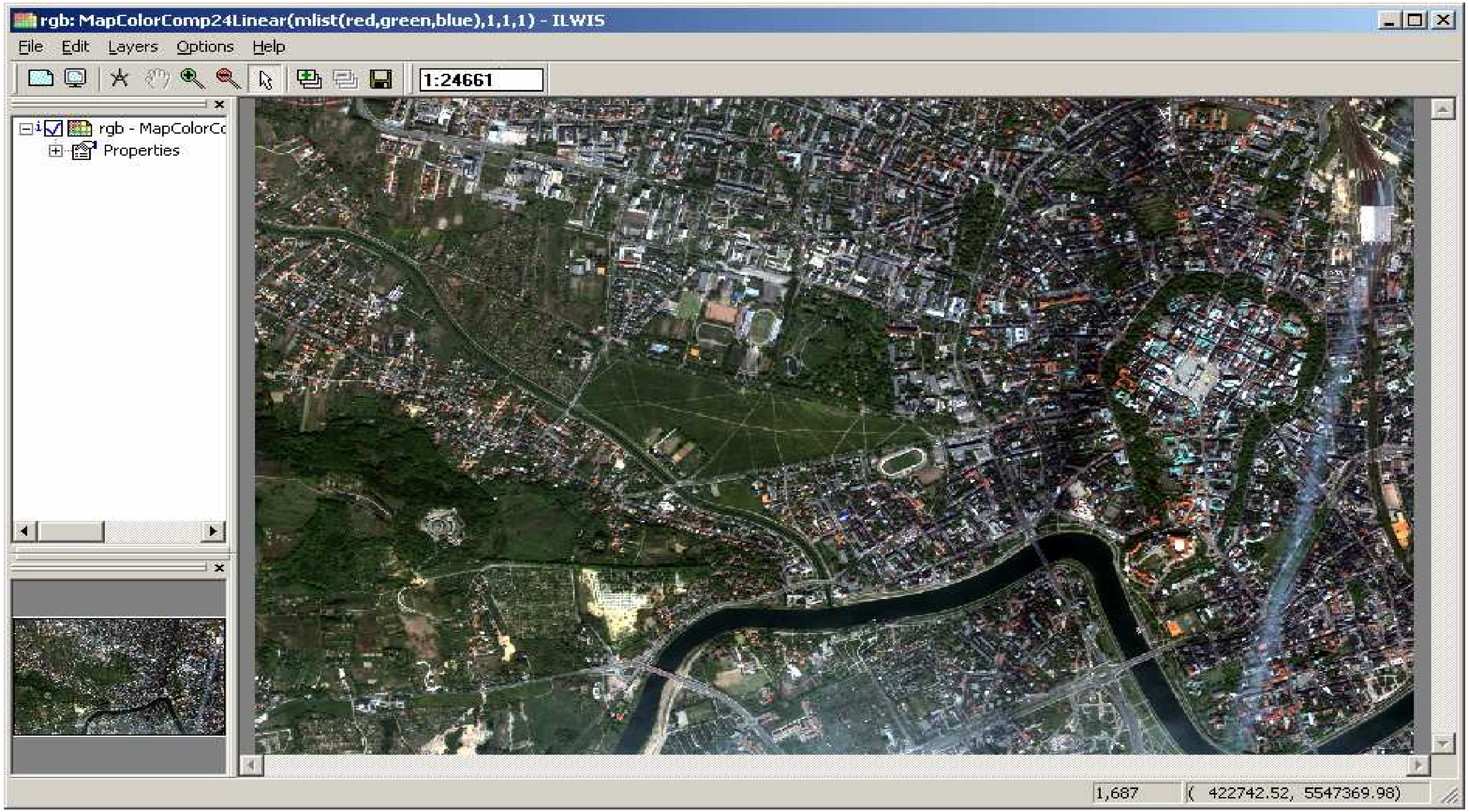




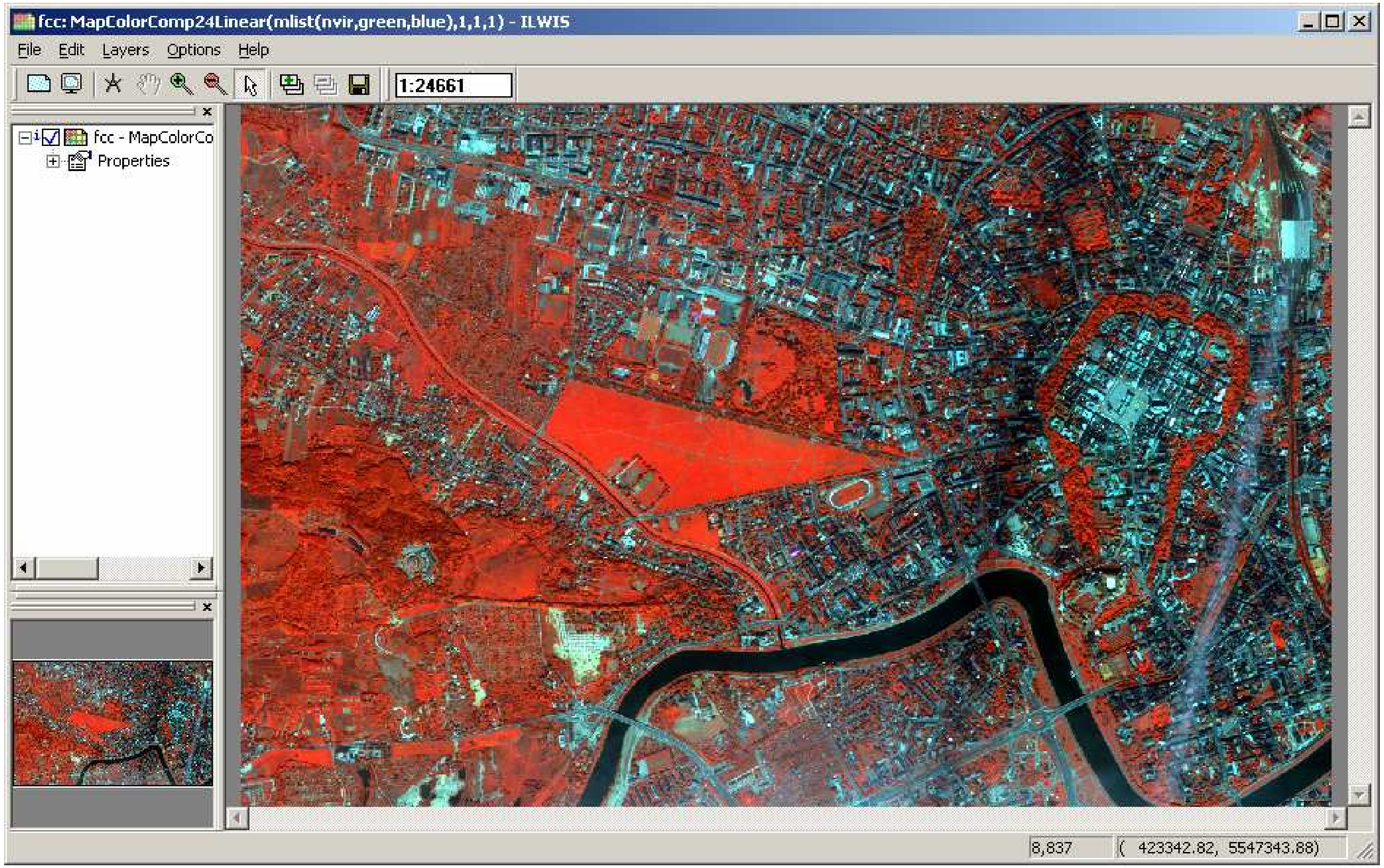
Visualisation after equalization



Color composite



Color composite

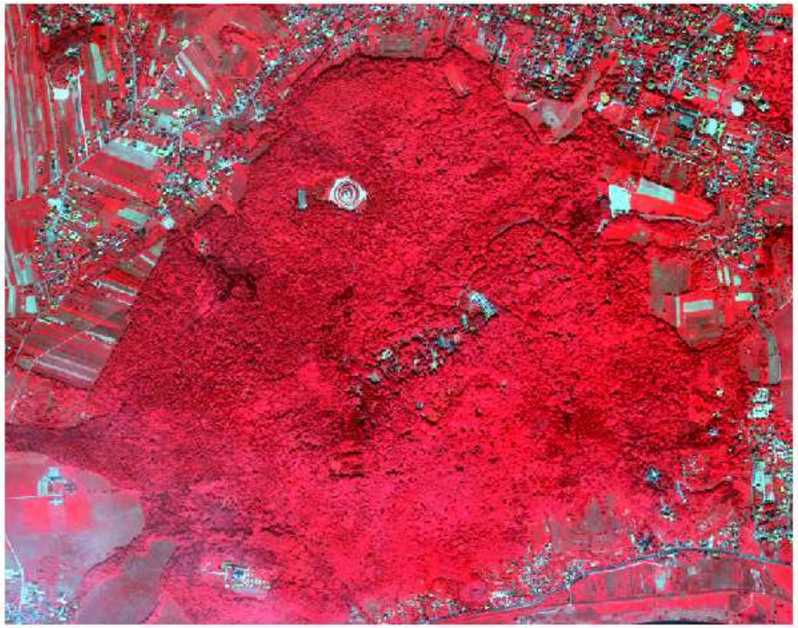


Interpretation key

Part of the image with the object and its descriptions

Interpretation key

Interpretation key



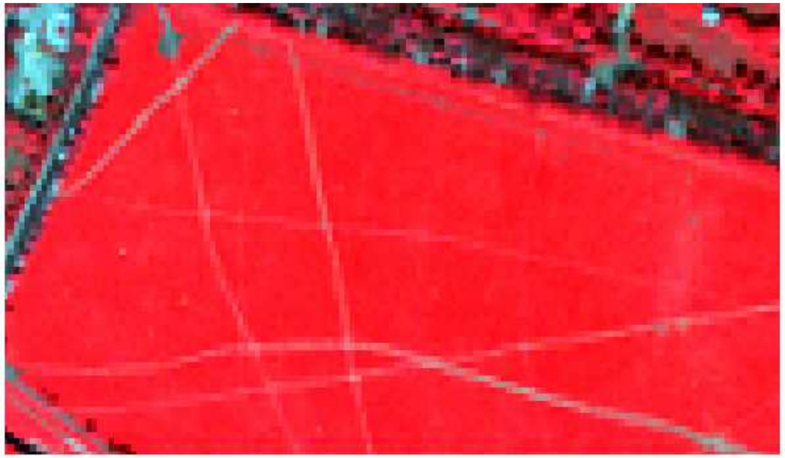
Forest - 1

•Color- red
•Structure – coarse, barankowa

Structure:

•smooth (amorphous)
•fine
•coarse
•…….

Interpretation key



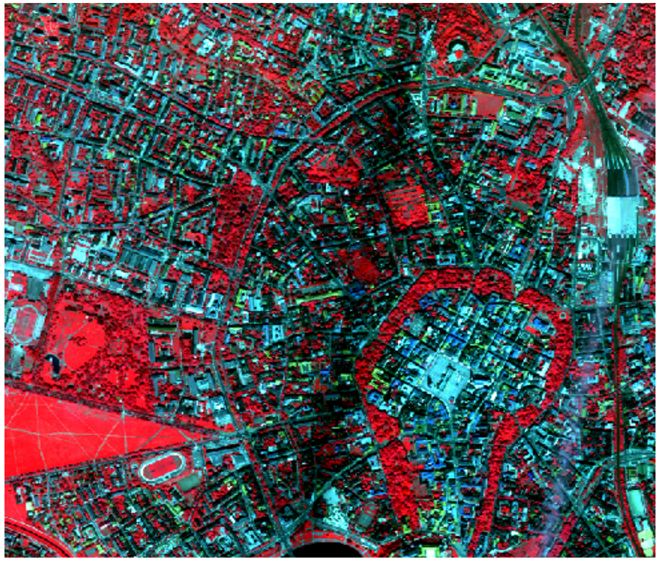
Grass - 2

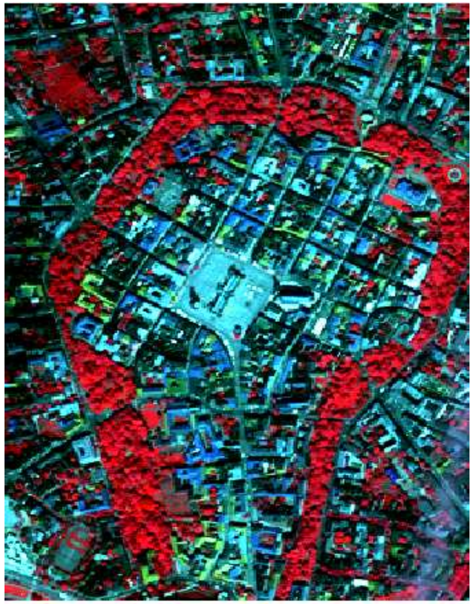
•Color– ligh red
•Strukture – amophous

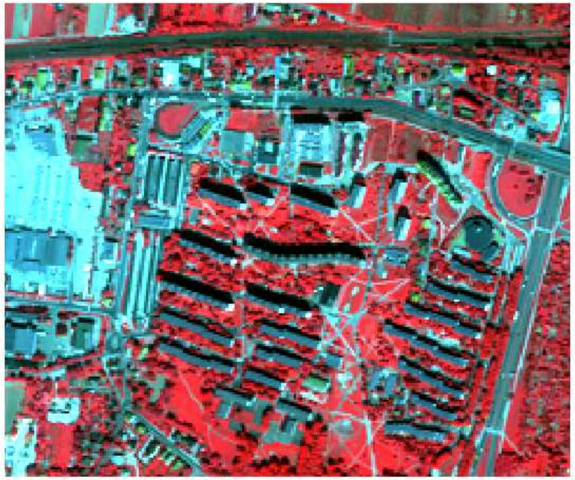
Structure:

•smooth (amorphous)
•fine
•coarse
•…….

Interpretation key







Urban- 3

- Color
  - steel
  - gree
- Strukture
  - coarse
  - fine

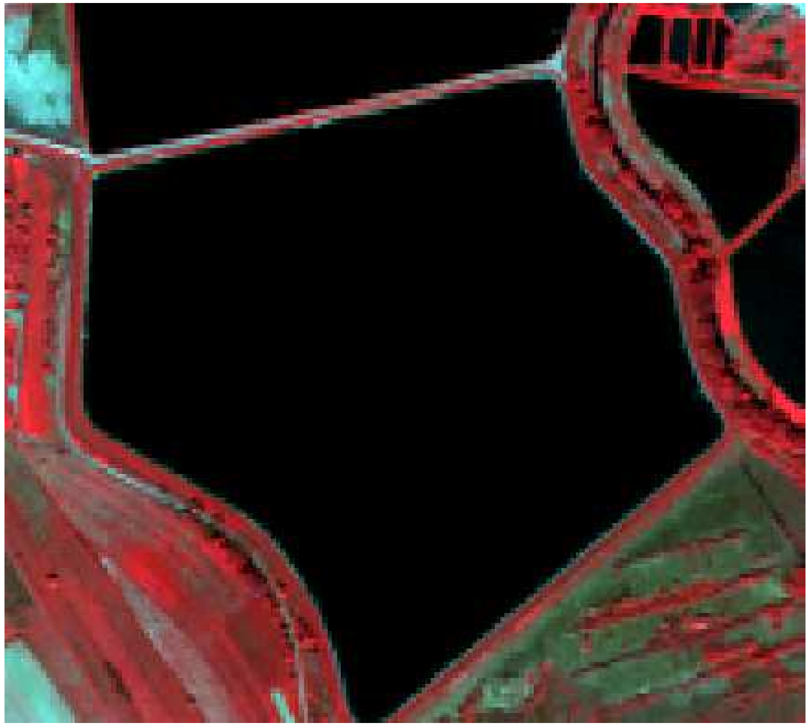
Structure:

- smooth (amorphous)
- fine
- coarse
- spotted
- …….

Interpretation key

Water - 4

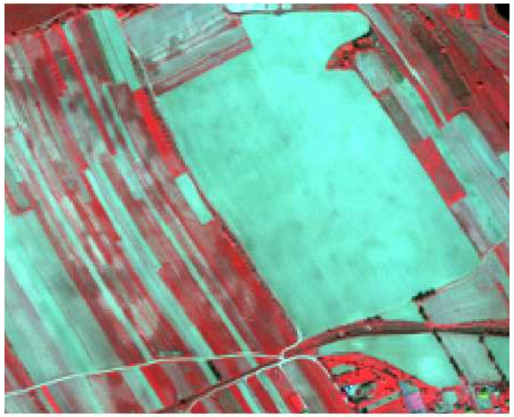
•Kolor – black
•Struktura – amorphous

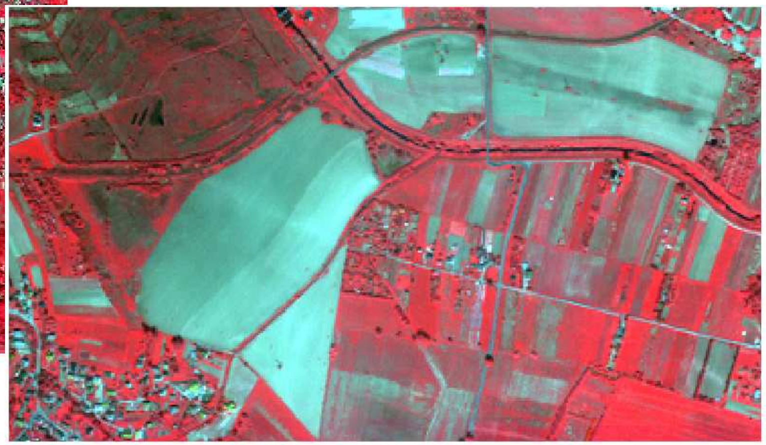


Structure:

•smooth (amorphous)
•fine
•coarse
•spotted
•…….

Interpretation key





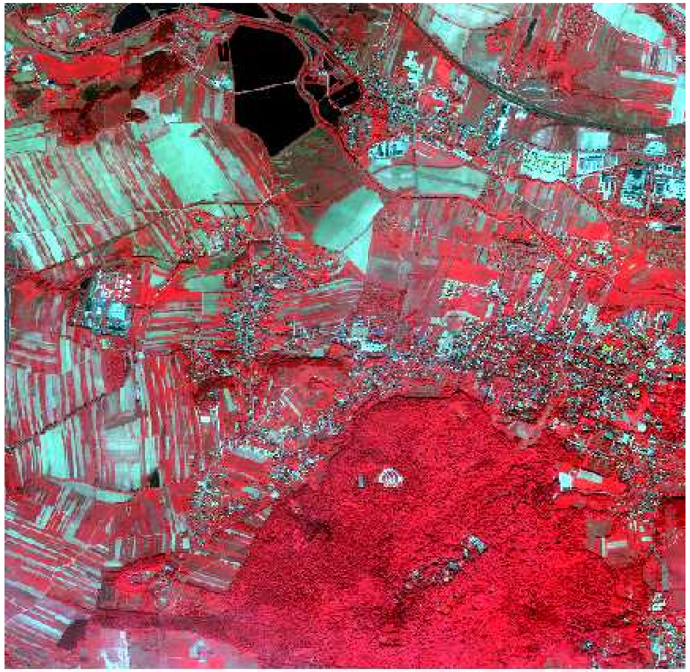
Bare soils - 5

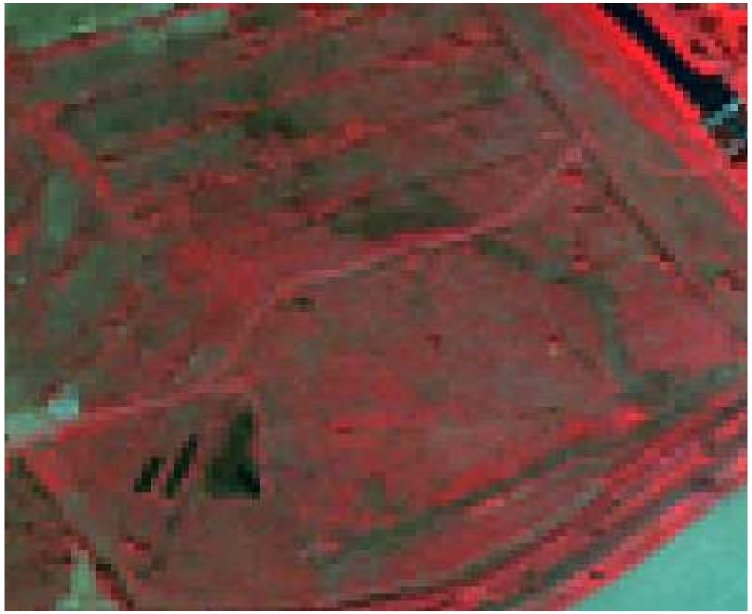
•Color – cyan - bright
•Structure – spotted

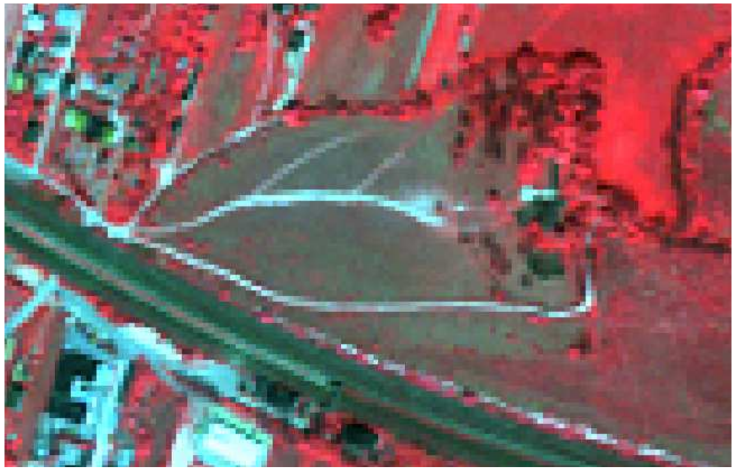
Structure:

•smooth (amorphous)
•fine
•coarse
•spotted
•…….

Interpretation key







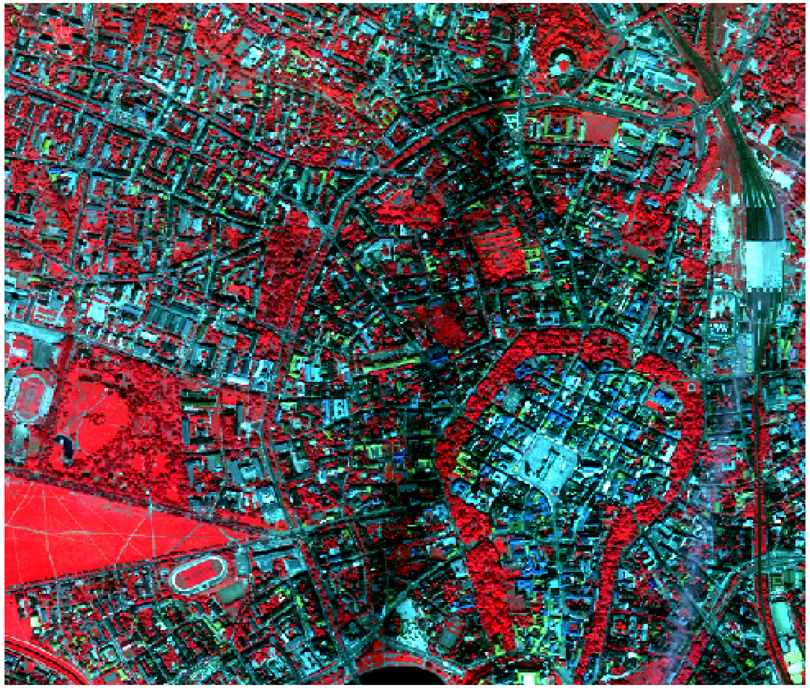
Wastelands- 6

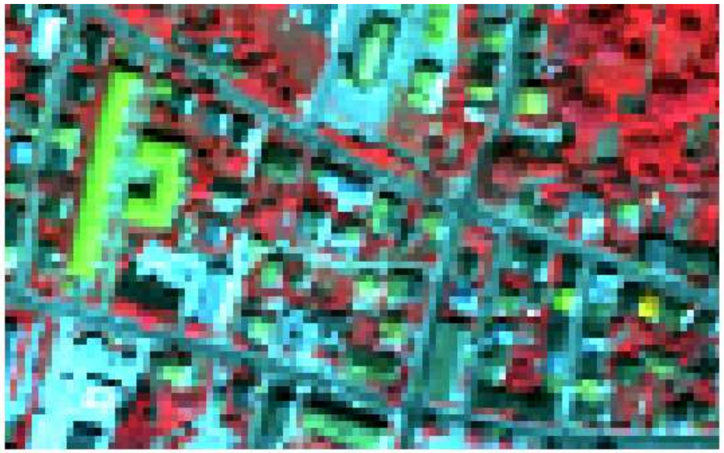
•Color - dark grees, with red strips
•Strukture – spotted

Structure:

•smooth (amorphou
•fine
•coarse
•spotted
•…….

Interpretation key





Roofs res 7

•Color - green
•Strukture - amorphous

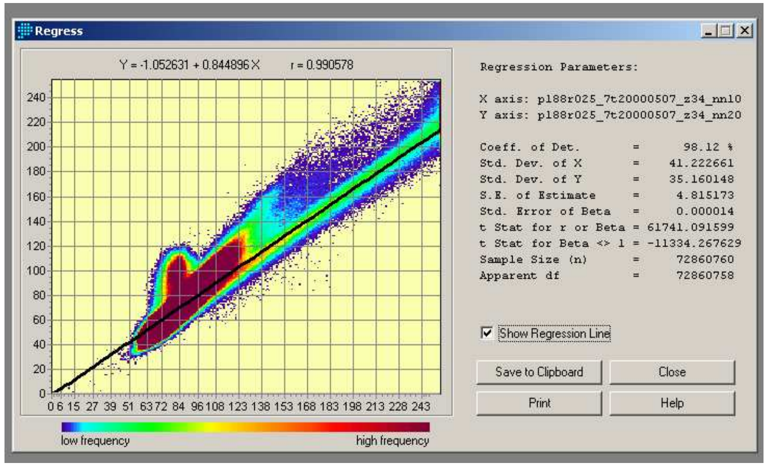
Structure:

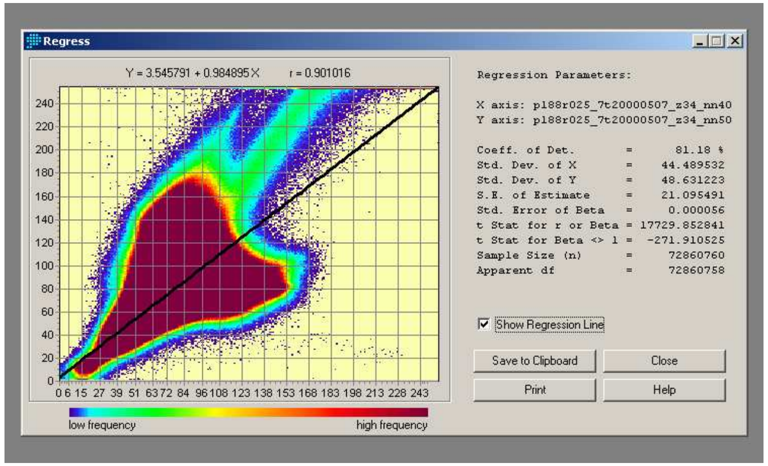
•smooth (amorphous)
•fine
•coarse
•spotted
•…….

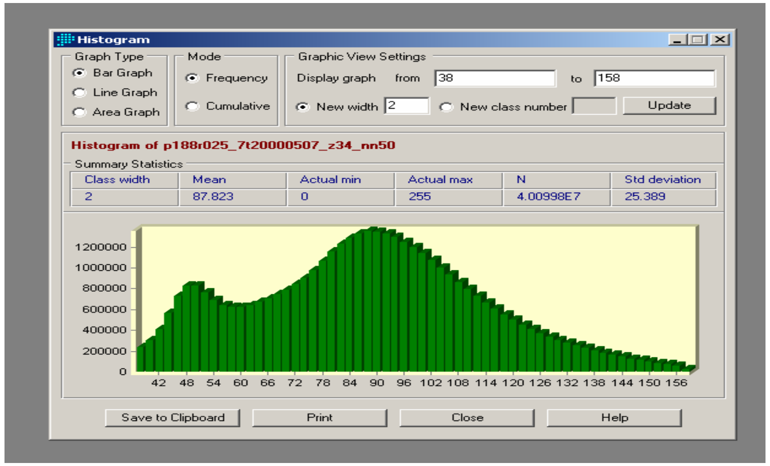
Image classification

- Unsupervised
- Supervised

Base of automatic classification







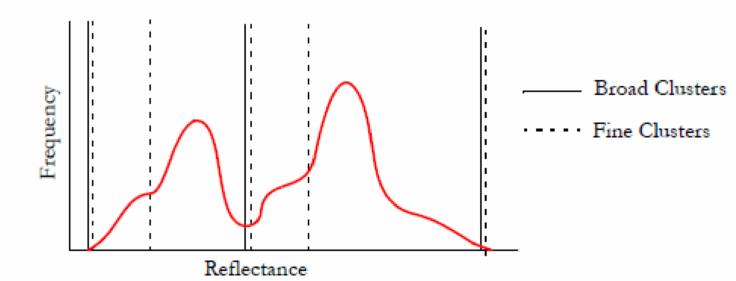
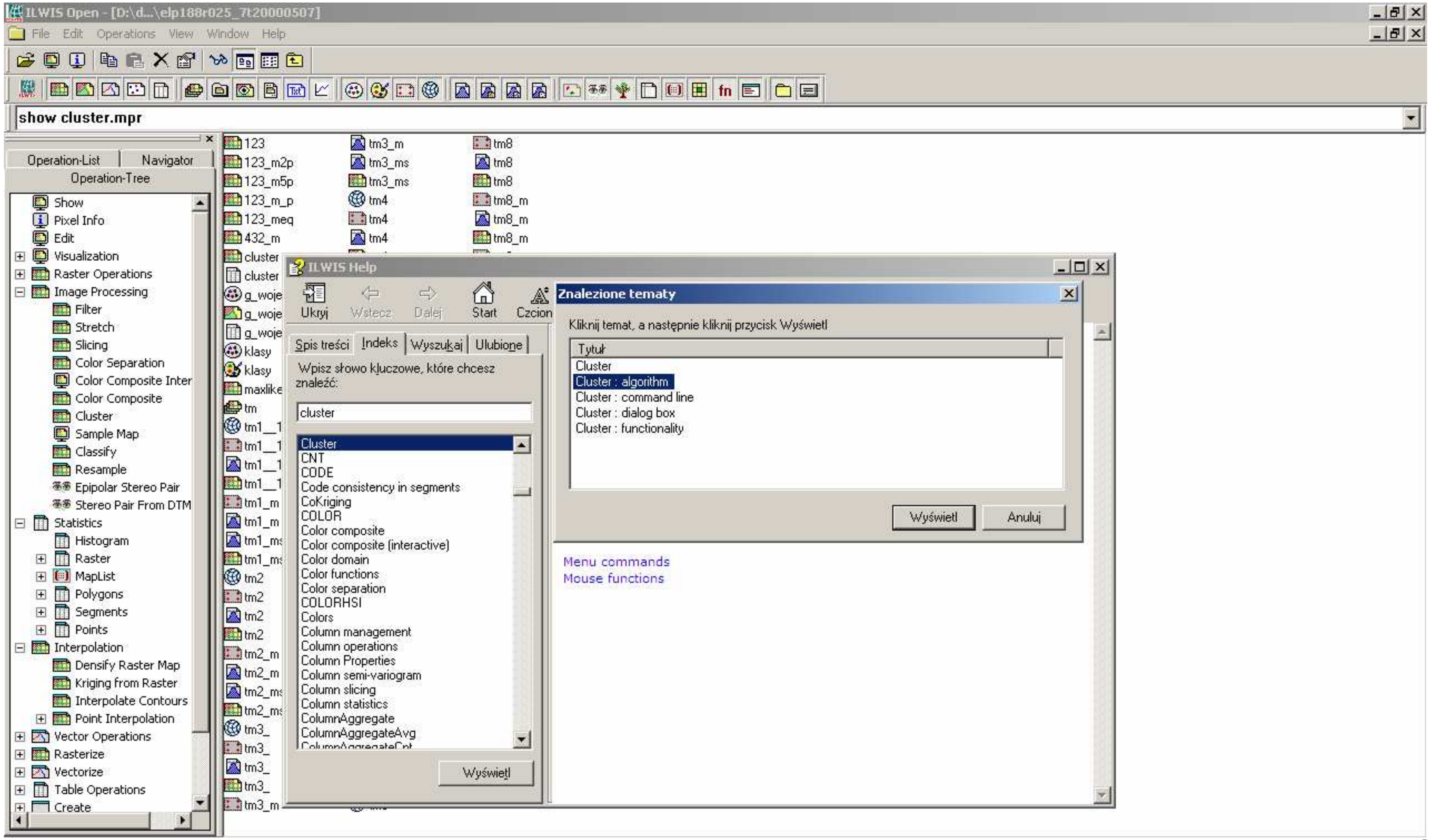
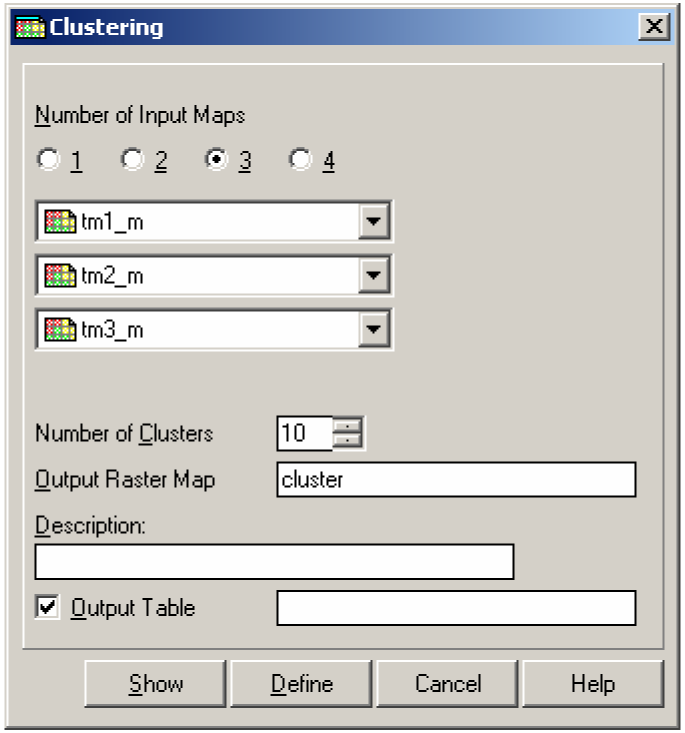


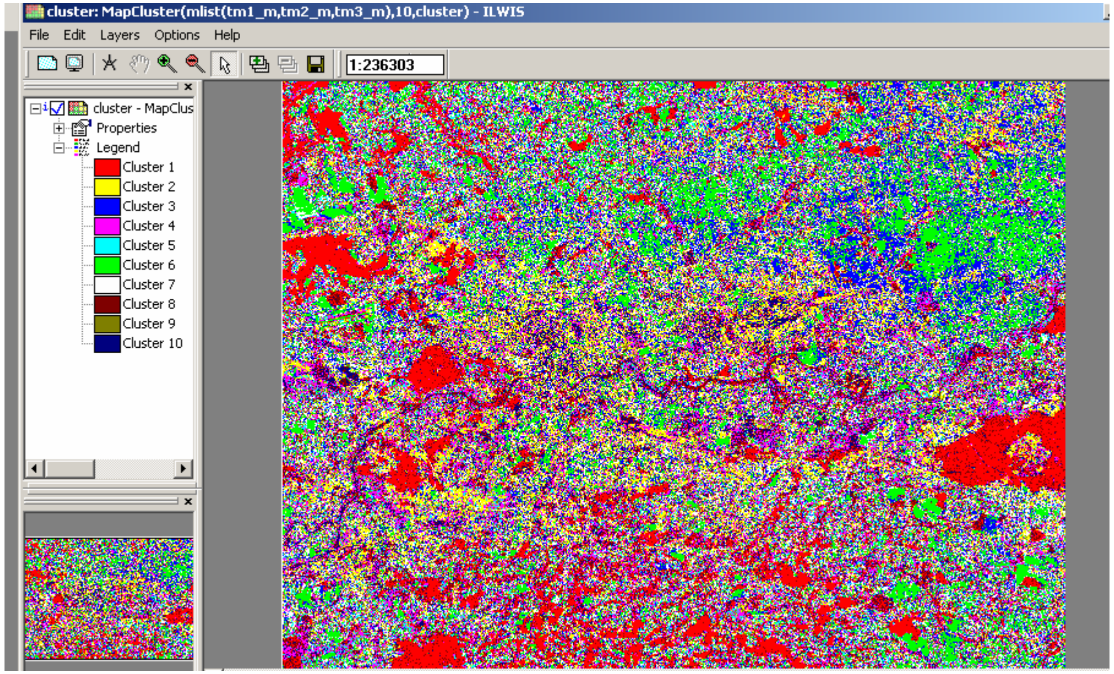
Figure 1

Cluster unsupervised classifcation



Cluster


Clustering
Number of Input Maps
1
2
3
4
tm1_m
tm2_m
tm3_m
Number of Clusters
10
Output Raster Map
cluster
Description:
Output Table
Show
Define
Cancel
Help


cluster: MapCluster(mlist(tm1_m,tm2_m,tm3_m),10,cluster) - ILWIS
File
Edit
Layers
Options
Help
1:236303
cluster - MapClus
Properties
Legend
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Cluster 5
Cluster 6
Cluster 7
Cluster 8
Cluster 9
Cluster 10

Statistics

$$War(X) = \frac{n\sum x^2 - (\sum x)^2}{n(n-1)}$$

$$\delta_x = \sqrt{War(X)}$$

$$Kow(X,Y) = \frac{1}{n}\sum_{i=1}^{n}(x_i - \mu_x)(y_i - \mu_y)$$

$$\rho_{x,y} = \frac{Kow(X,Y)}{\delta_x \delta_y}$$

Variable - example

10	12	15
10	12	12
10	10	15

TM1

mean = 12

variance = 4

Standard deviation = 2

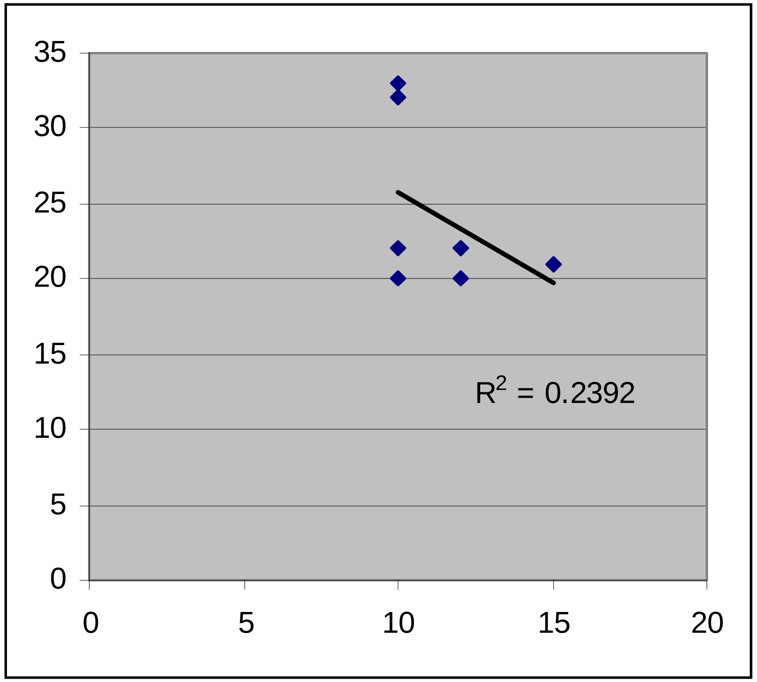
22	20	21
20	22	22
33	32	21

TM2

mean =24

variance =25

Standard deviation = 5



Covariance = -4.5

Correlation cooeficient = -0.489

Correlation between channels

MatrixCorr(grupa_w) - ILWIS

File Edit View Help

Mean per band:

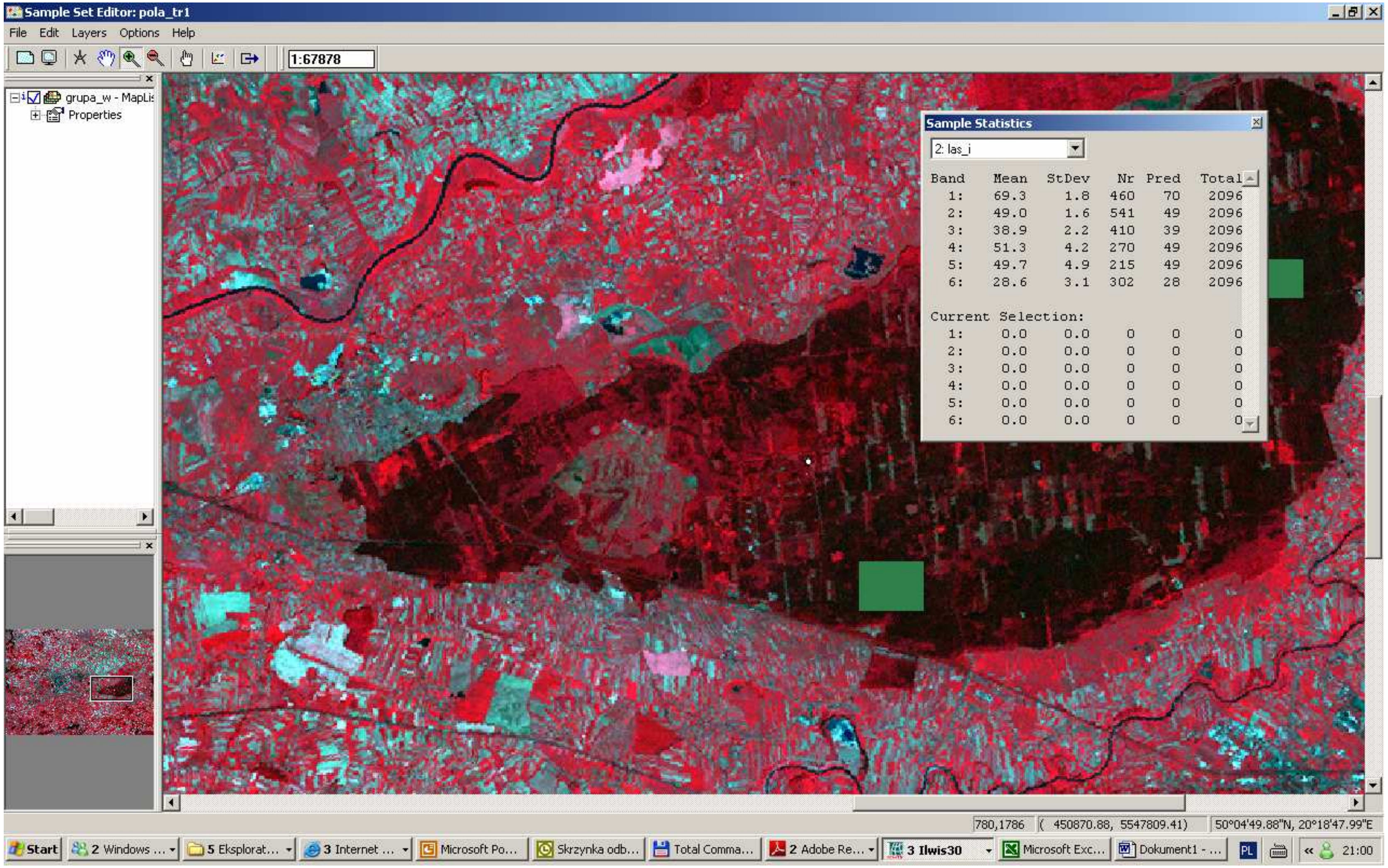
80.00	66.94	61.19	90.64	92.90	61.09

Std. per band:

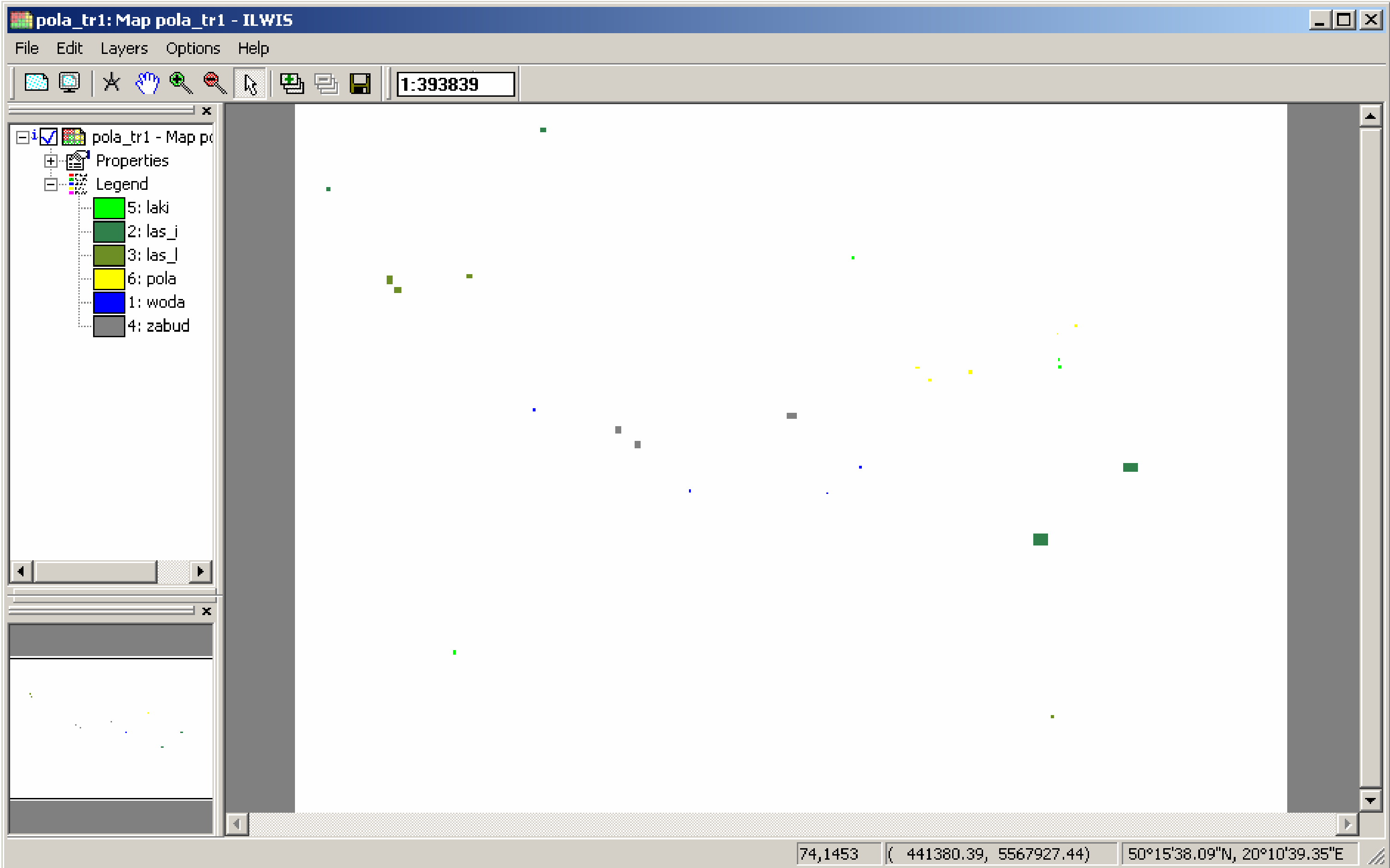
9.89	12.42	21.73	19.89	25.27	28.89

	grupa_w_1	grupa_w_2	grupa_w_3	grupa_w_4	grupa_w_5	grupa_w_6
grupa_w_1	1.00	0.95	0.95	-0.26	0.79	0.87
grupa_w_2	0.95	1.00	0.97	-0.11	0.84	0.87
grupa_w_3	0.95	0.97	1.00	-0.26	0.84	0.92
grupa_w_4	-0.26	-0.11	-0.26	1.00	0.04	-0.24
grupa_w_5	0.79	0.84	0.84	0.04	1.00	0.94
grupa_w_6	0.87	0.87	0.92	-0.24	0.94	1.00

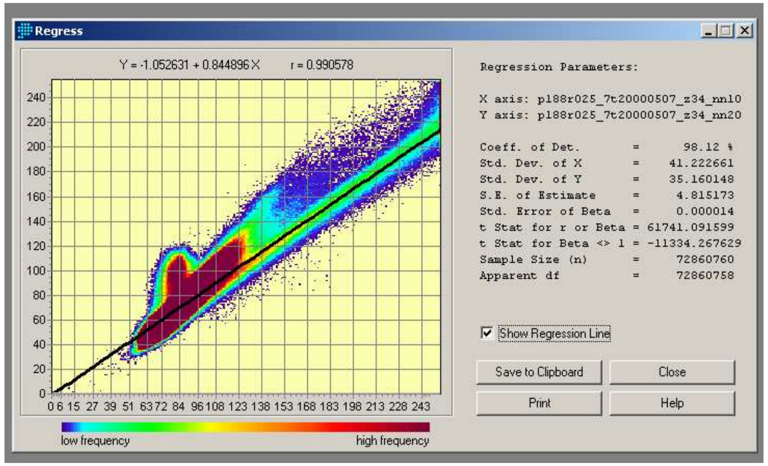
Training fields

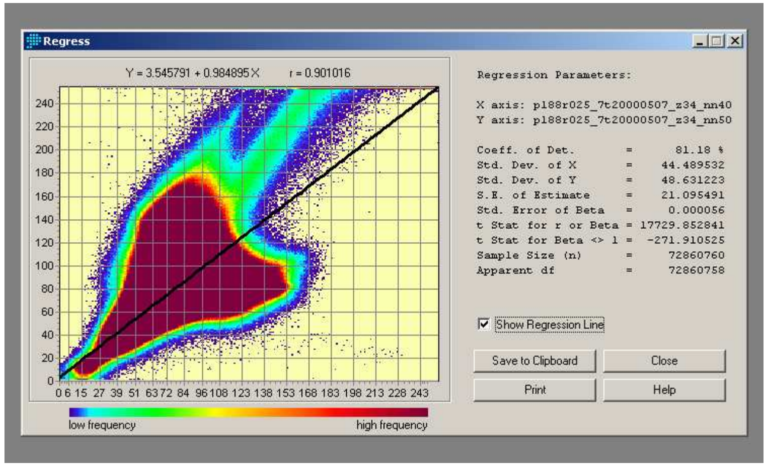


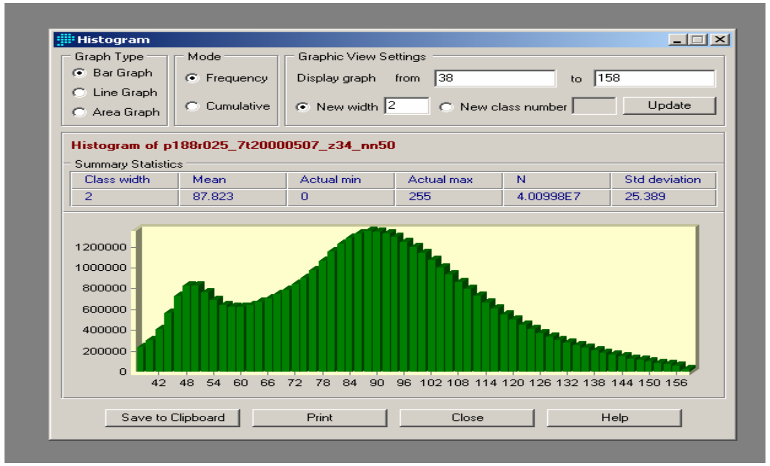
Training fields



Base of automatic classification







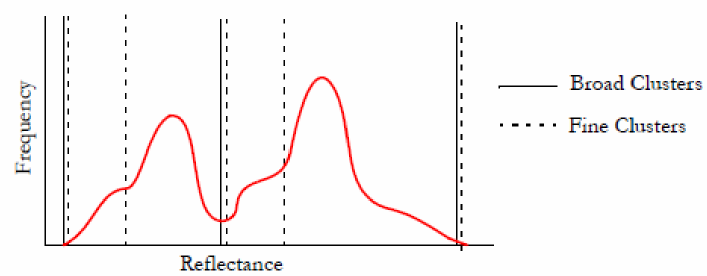
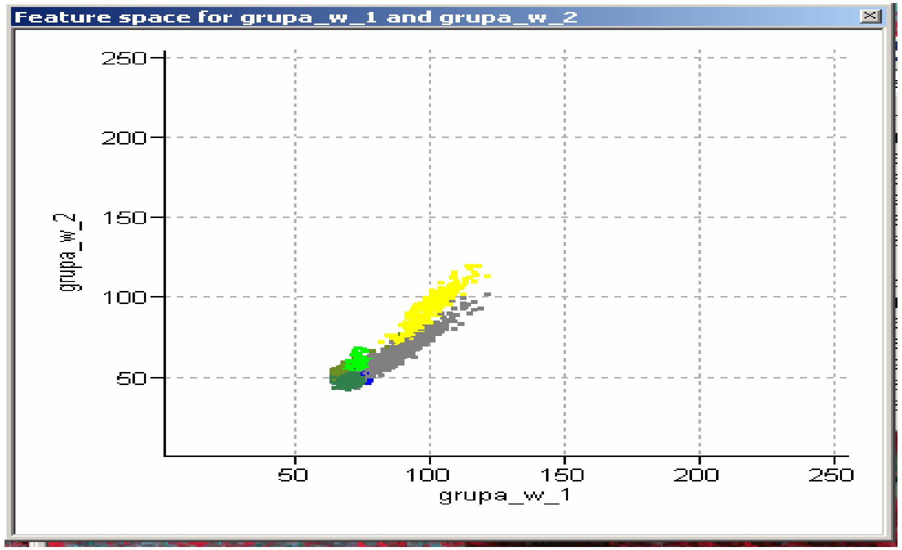
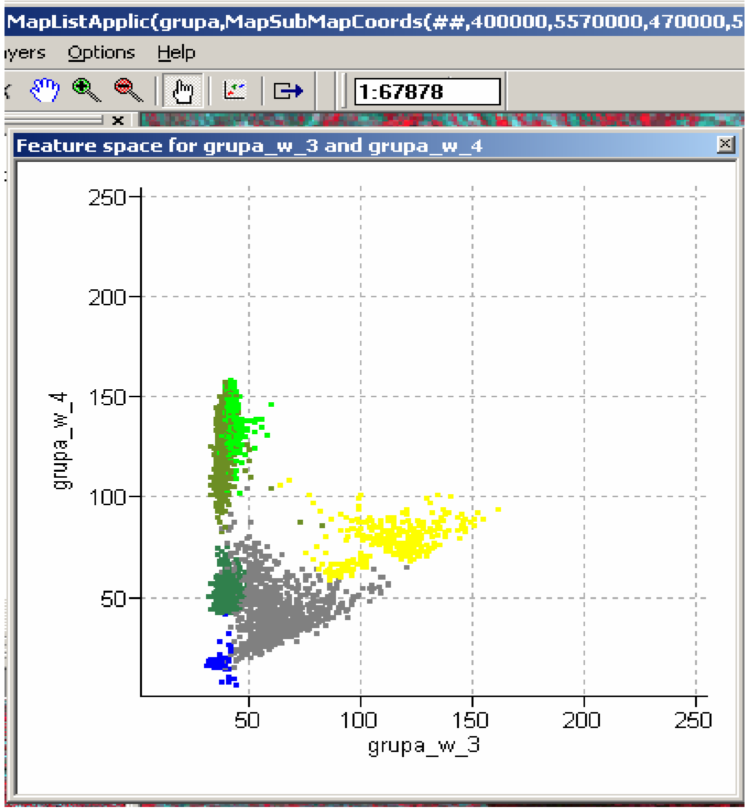


Figure 1

Statistics of training fields





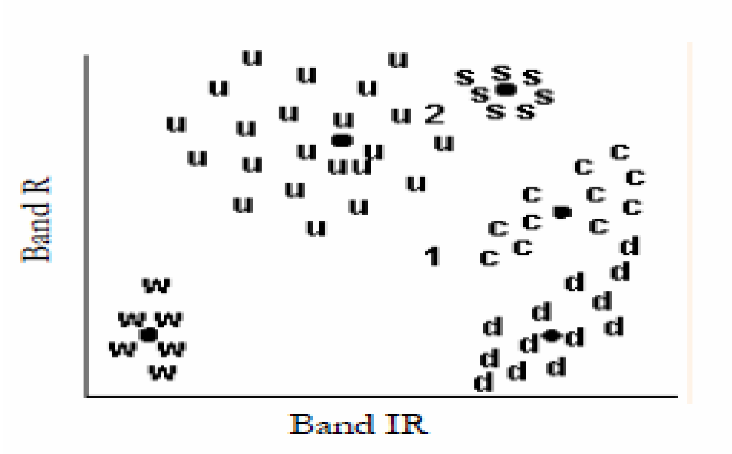
Classification methods

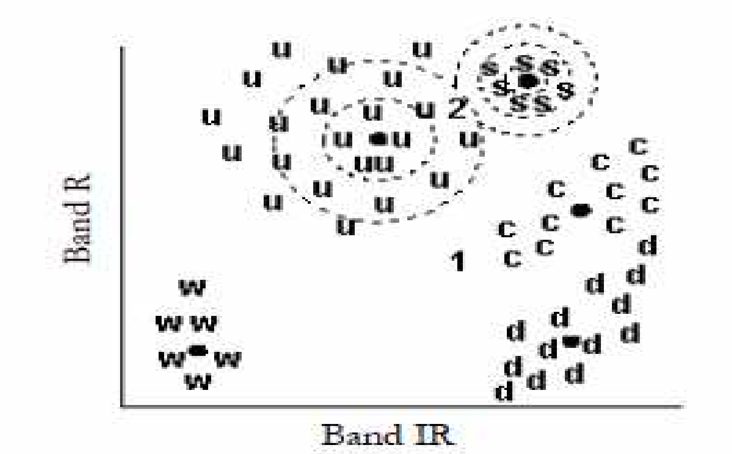
- The following classification methods are available:
- Box classifier, using a multiplication factor,
- Minimum Distance, optionally using a threshold value,
- Minimum Mahalanobis distance, optionally using a threshold value,
- Maximum Likelihood, optionally using a threshold value,
- Maximum Likelihood including Prior Probabilities, optionally using a threshold value.

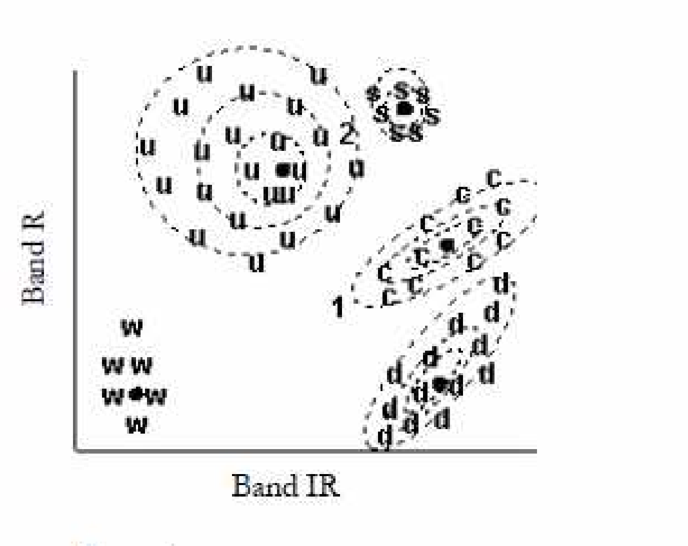
Classification methods

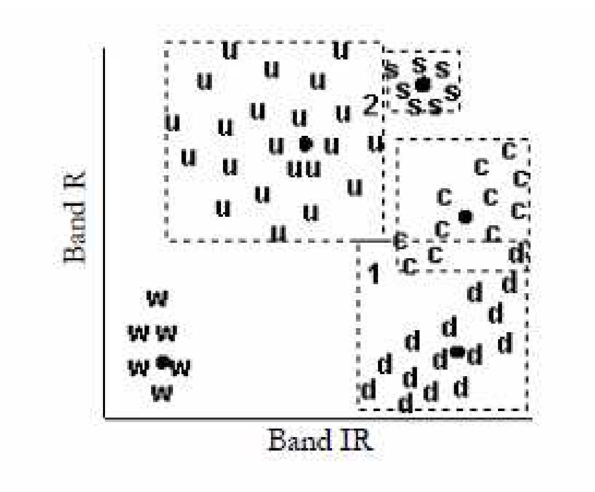
- Prior to any classification, empirical statistics are drawn from the training pixels in the input sample set. These sample statistics are calculated per class of training pixels and per band. For instance, for a single class (*i*), *n* mean values are calculated when there are *n* input bands; these *n* mean values together are called the class mean (vector) for that class (**m**i).
- Depending on the selected classification method, the following statistics are calculated:
- for each *class i* of training pixels:
  - the means of training pixels per band (**m**i),
  - in case of box classifier: the variance of the training pixels per band,
  - the standard deviation of the training pixels per band (should be > 0),
  - the predominant value (mode) per band,
  - in case of Minimum Mahalanobis distance, Maximum Likelihood and Prior Probability classifier: an *n* x *n* variance-covariance matrix (Vi) which stores class variance per band, and class covariance between bands.
- For each feature vector to be classified, these statistics are used to calculate the shortest 'distance' towards the training classes. All classification decisions are thus based on these statistical empirical parameters.

Supervised classification









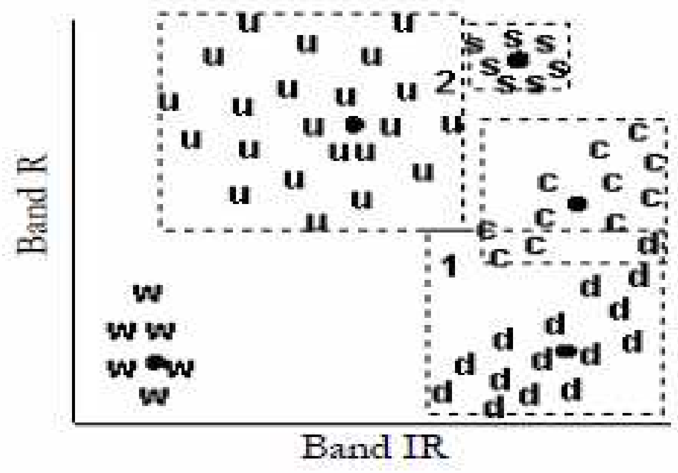
Box classifier

For each class, a multi-dimensional box is drawn around the class mean.

For each class, the size of the box is calculated as:

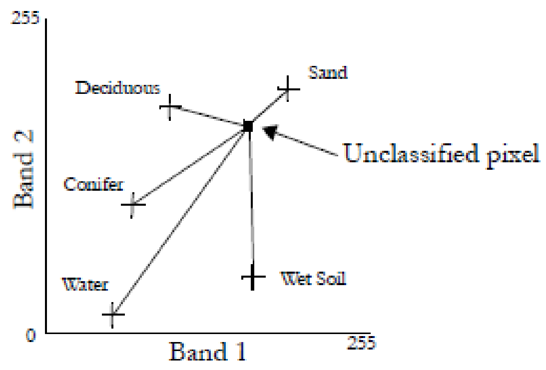
(class mean ± standard deviation per band) * multiplication factor

- If a feature vector falls inside a box, then the corresponding class name is assigned.
- if a feature vector falls within two boxes, the class name of the box with the smallest product of standard deviations is assigned, i.e. the class name of the smallest box.
- if a feature vector does not fall within a box, the undefined value is assigned.

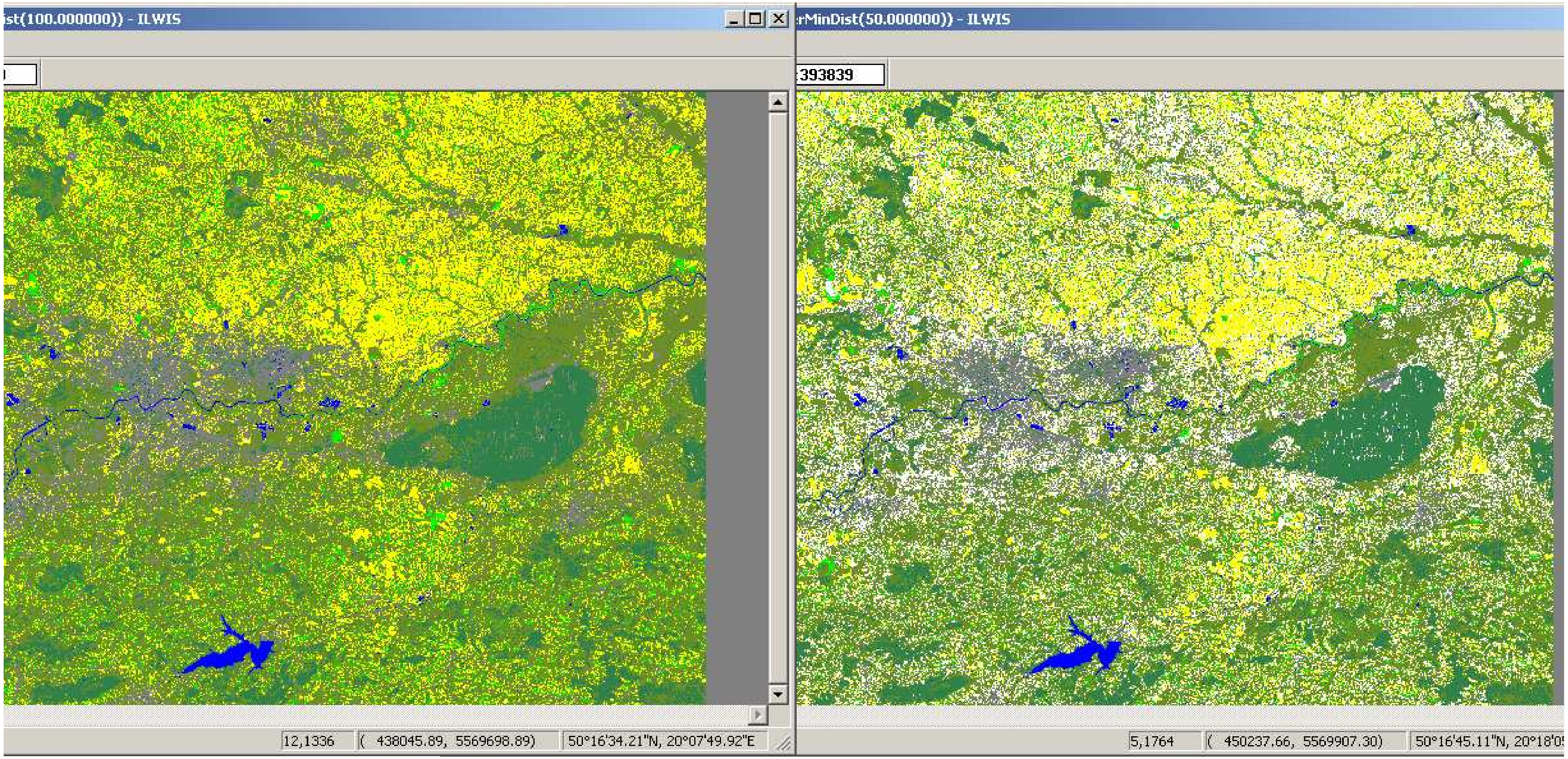


Minimum Distance to Mean

- For each feature vector, the distances towards class means are calculated.
- The shortest Euclidian distance to a class mean is found;
- if this shortest distance to a class mean is smaller than the user-defined threshold, then this class name is assigned to the output pixel.
- else the undefined value is assigned.



Mindist (100 i 50)



Minimum Mahalanobis distance:

For each feature vector, the Mahalanobis distances towards class means are calculated. This includes the calculation of the variance-covariance matrix V for each class *i*.

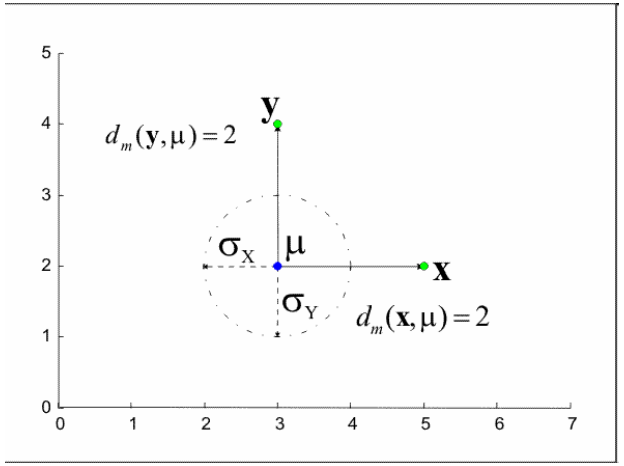
The Mahalanobis distance is calculated as:

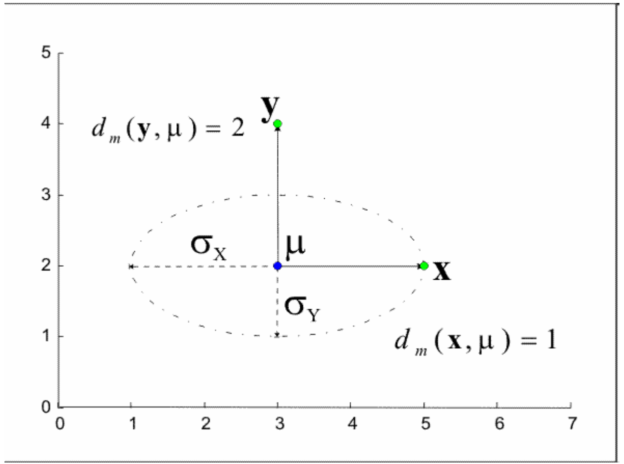
$$di(\mathbf{x}) = \mathbf{y}^T V_i^{-1} \mathbf{y}$$

For an explanation of the parameters, see Maximum Likelihood classifier.

- For each feature vector **x**, the shortest Mahalanobis distance to a class mean is found;
- if this shortest distance to a class mean is smaller than the user-defined threshold, then this class name is assigned to the output pixel.
- else the undefined value is assigned.

Machalanobis distance





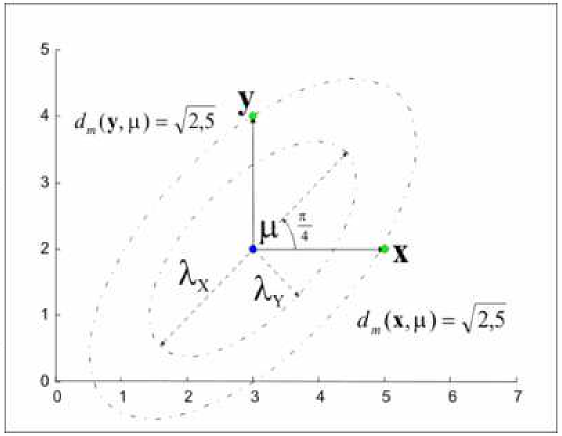
$$d_m(\mathbf{x},\boldsymbol{\mu}) = \sqrt{(x_1-\mu_1)^2+\ldots+(x_n-\mu_n)^2} = \sqrt{(\mathbf{x}-\boldsymbol{\mu})\mathbb{I}^{-1}(\mathbf{x}-\boldsymbol{\mu})^T}$$

$$d_m(\mathbf{x},\boldsymbol{\mu}) = \sqrt{\frac{(x_1-\mu_1)^2}{\sigma_1^2}+\ldots+\frac{(x_n-\mu_n)^2}{\sigma_n^2}} = \sqrt{(\mathbf{x}-\boldsymbol{\mu})D^{-1}(\mathbf{x}-\boldsymbol{\mu})^T}$$

gdzie D jest macierzą diagonalną $\mathrm{diag}(\sigma_1^2,\sigma_2^2,\ldots,\sigma_n^2)$.

Machalanobis distance

$di(\mathbf{x}) = \mathbf{y}^T V_i^{-1} \mathbf{y}$



Maximum Likelihood

For each feature vector, the distances towards class means are calculated. This includes the calculation of the variance-covariance matrix V for each class *i*.

The formula used in Maximum Likelihood reads:

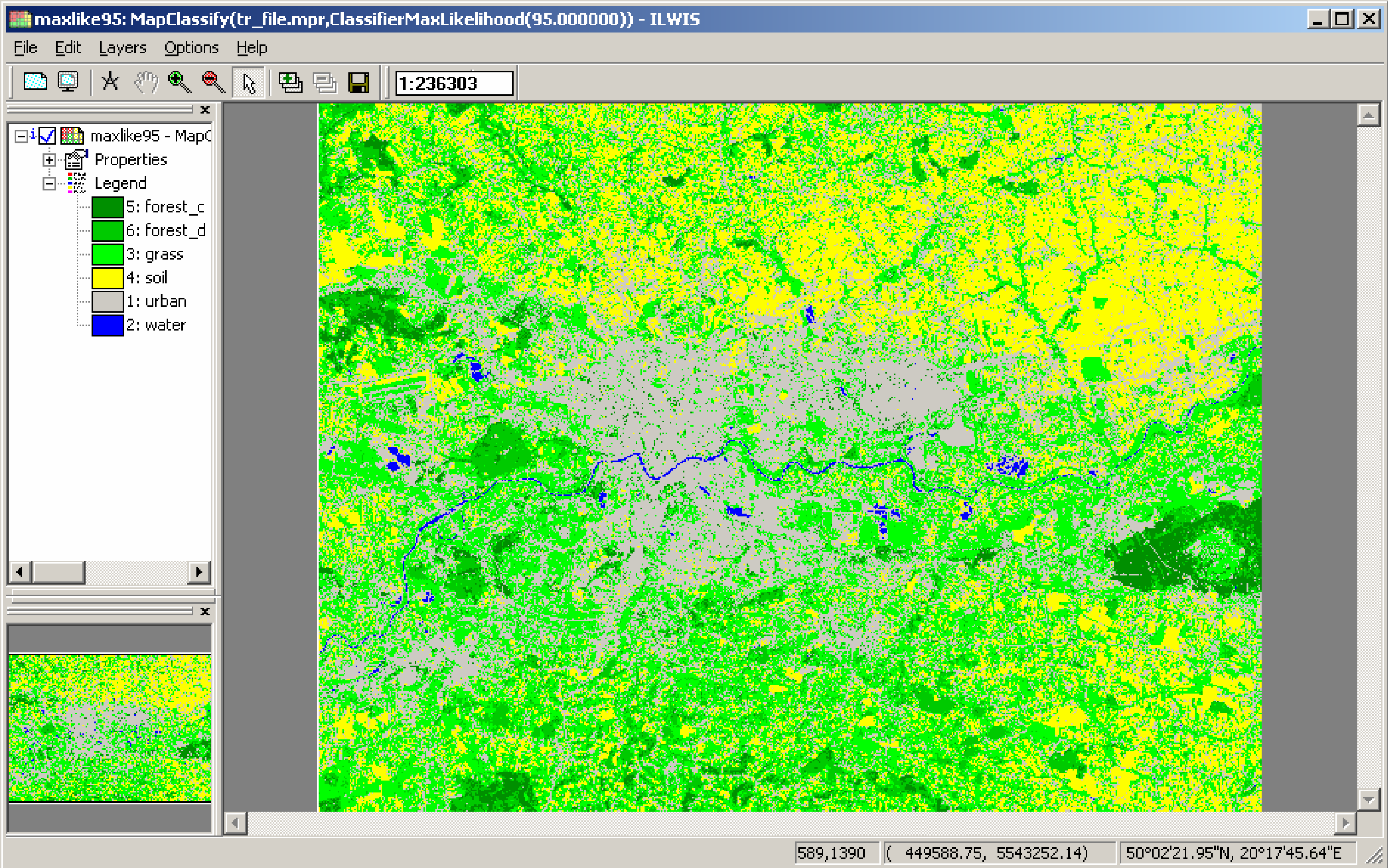
$d_i(\mathbf{x}) = \ln|V_i| + \mathbf{y}^T V_i^{-1} \mathbf{y}$

where:

- d_i - distance between feature vector ($\mathbf{x}$) and a class mean ($\mathbf{m}_i$) based on probabilities
- V_i - the *n* x *n* variance-covariance matrix of class *i*, where *n* is the number of input bands
- $|V_i|$ - determinant of V_i
- V_i^{-1} - the inverse of V_i
- **Y - x** - **m**i ; is the difference vector between feature vector **x** and class mean vector $\mathbf{m}_i$
- $\mathbf{y}^T$ - the transposed of **y**

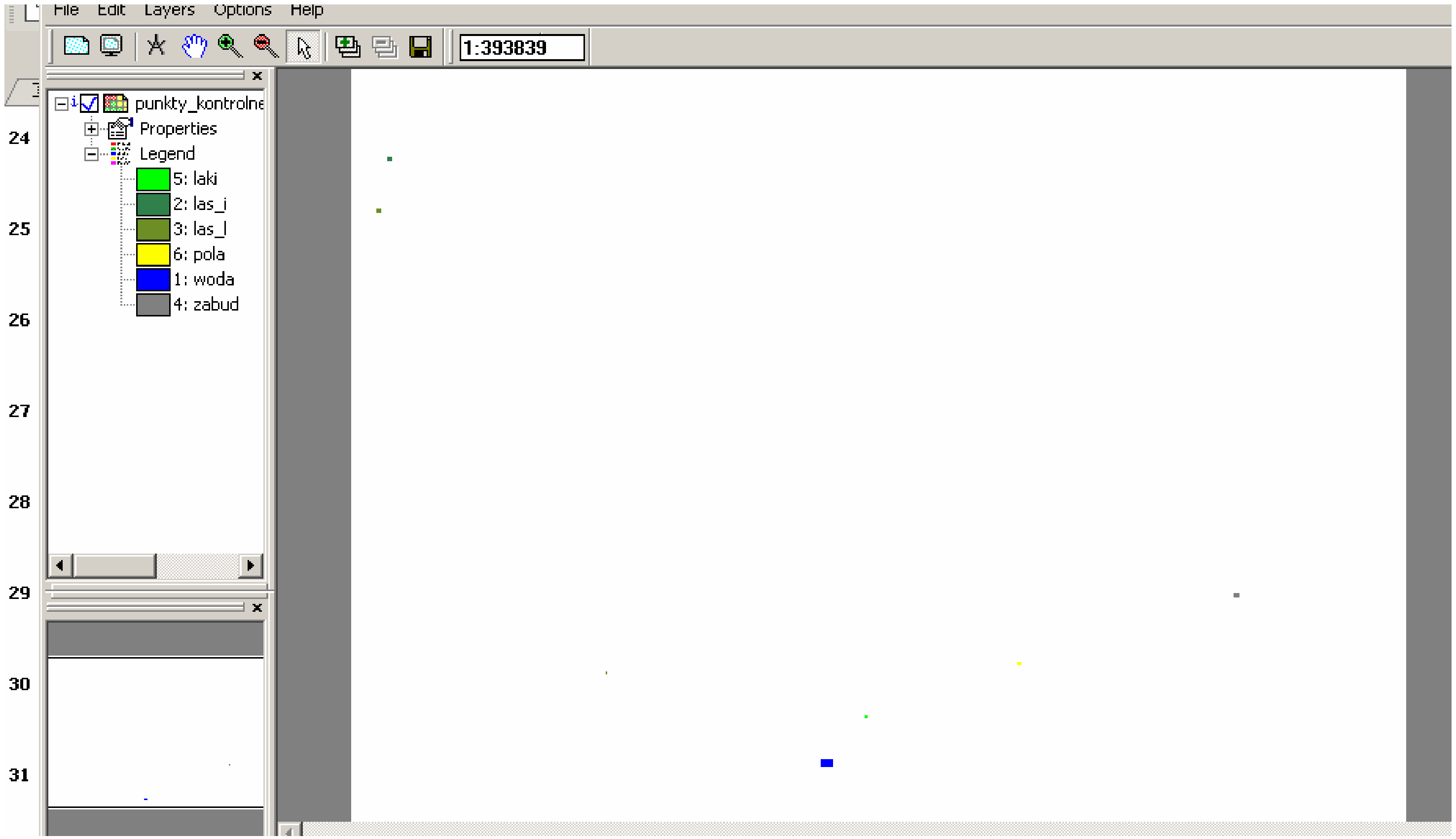
- For each feature vector **x**, the shortest distance di to a class mean $\mathbf{m}_i$ is found;
- if this shortest distance to a class mean is smaller than the user-defined threshold, then this class name is assigned to the output pixel.
- else the undefined value is assigned.

Maximum Likelihood



Accuracy analysis

- Control fields



Accuracy analysis – cross matrix

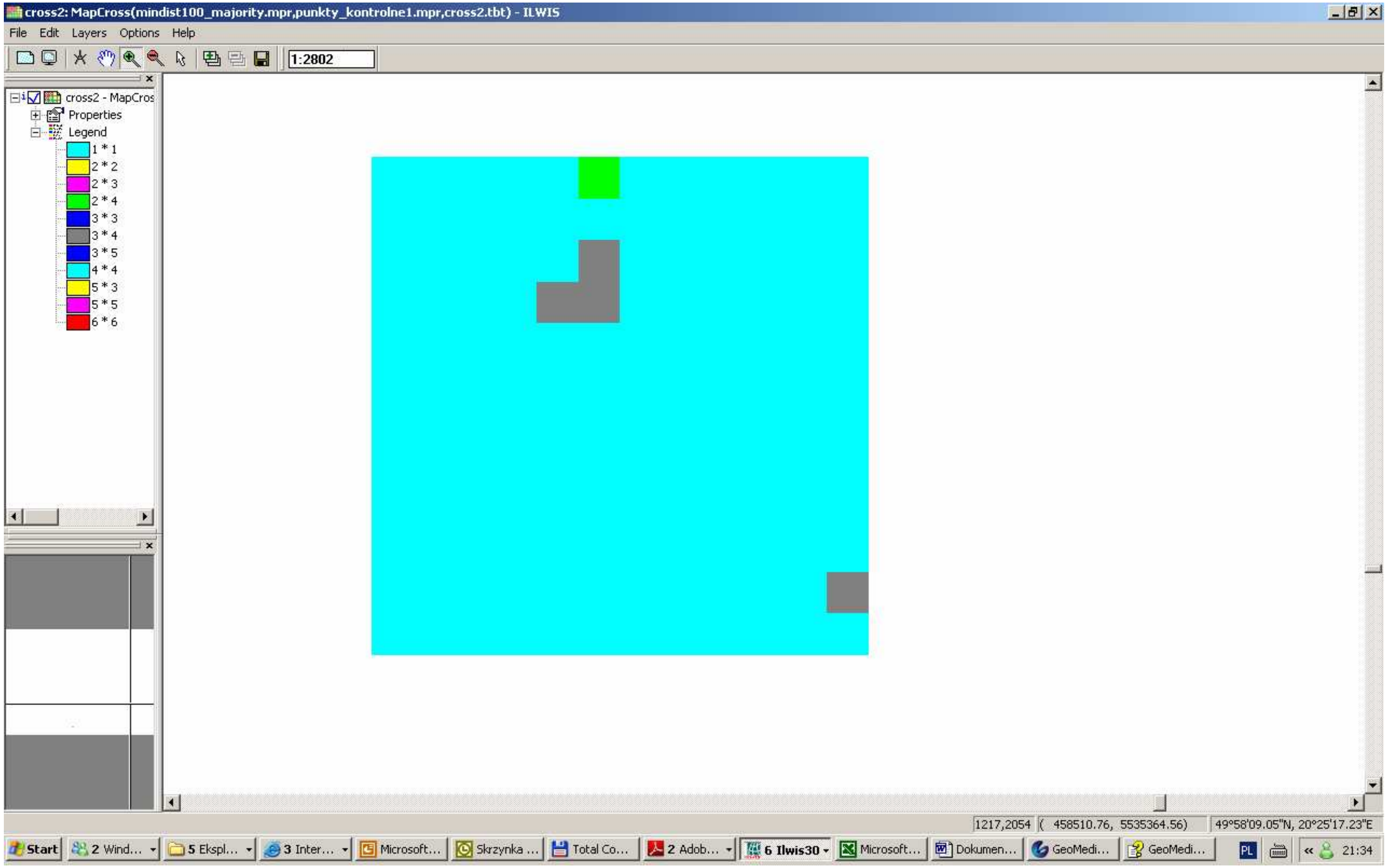
MatrixConfusion(D:\elp188r025_7t20000507\cross_tab1.tbt,punkty_kontrolne1,mindist100,NPix) - ILWIS

File Edit View Help

```
Average Accuracy    =  80.20 %
Average Reliablity =  81.17 %
Overall Accuracy    =  89.96 %
```

	laki	las_i	las_l	pola	woda	zabud	UNCLASSII	ACCURACY
laki	9	0	54	0	0	0	0	0.14
las_i	0	117	0	0	0	0	0	1.00
las_l	14	3	150	0	0	0	0	0.90
pola	0	0	0	81	0	0	0	1.00
woda	0	0	0	0	464	0	0	1.00
zabud	1	9	14	9	0	111	0	0.77
RELIABILITY	0.38	0.91	0.69	0.90	1.00	1.00		

Accuracy analysis



Accuracy analysis

$$I_{CP} = \frac{\sum_{p=1}^{k} n_{pp}}{n},$$

Overall accuracy

$$A_P = \frac{n_{jj}}{\sum_{i=1}^{k} n_{ij}},$$

Producer's accuracy

$$A_U = \frac{n_{ii}}{\sum_{j=1}^{k} n_{ij}}.$$

User's accuracy

nka Drukuj Opcje

Interpretation of a confusion matrix:

Consider the following example of a confusion matrix:

		CLASSIFICATION RESULTS							
		forest	bush	crop	urban	bare	water	unclass	ACC
GROUND TRUTH	forest	440	40	0	0	30	10	10	0.83
	bush	20	220	0	0	40	10	20	0.71
	crop	10	10	210	10	50	10	60	0.58
	urban	20	0	20	240	100	10	40	0.56
	bare	0	0	10	10	230	0	10	0.88
	water	0	20	0	0	0	240	10	0.89
	REL	0.90	0.76	0.88	0.92	0.51	0.86		

Average accuracy = 74.25%
Average reliability = 80.38%
Overall accuracy = 73.15%

In the example above:

- unclass represents the Unclassified column,
- ACC represents the Accuracy column,
- REL represents the Reliability column.

Explanation:

- Rows correspond to classes in the ground truth map (or test set).
- Columns correspond to classes in the classification result.
- The *diagonal elements* in the matrix represent the number of correctly classified pixels of each class, i.e. the number of ground truth pixels with a certain class name that actually obtained the same class name during classification. In the example above, 440 pixels of 'forest' in the test set were correctly classified as 'forest' in the classified image.
- The *off-diagonal elements* represent misclassified pixels or the classification errors, i.e. the number of ground truth pixels that ended up in another class during classification. In the example above, 40 pixels of 'forest' in the test set were classified as 'bush' in the classified image.
  - Off-diagonal *row* elements represent ground truth pixels of a certain class which were excluded from that class during classification. Such errors are also known as *errors of omission* or exclusion. For example, 50 ground truth pixels of 'crop' were excluded from the 'crop' class in the classification and ended up in the 'bare' class.
  - Off-diagonal *column* elements represent ground truth pixels of other classes that were included in a certain classification class. Such errors are also known as *errors of commission* or inclusion. For example, 100 ground truth pixels of 'urban' were included in the 'bare' class by the classification.
- The figures in column *Unclassified* represent the ground truth pixels that were found not classified in the classified image.

Accuracy (also known as producer's accuracy): The figures in column *Accuracy* (ACC) present the accuracy of your classification: it is the fraction of correctly classified pixels with regard to all pixels of that ground truth class. For each class of ground truth pixels (row), the number of correctly classified pixels is divided by the total number of ground truth or test pixels of that class. For example, for the 'forest' class, the accuracy is 440/530 = 0.83 meaning that approximately 83% of the 'forest' ground truth pixels also appear as 'forest' pixels in the classified image.

Reliability (also known as user's accuracy): The figures in row *Reliability* (REL) present the reliability of classes in the classified image: it is the fraction of correctly classified pixels with regard to all pixels classified as this class in the classified image. For each class in the classified image (column), the number of correctly classified pixels is divided by the total number of pixels which were classified as this class. For example, for the 'forest' class, the reliability is 440/490 = 0.90 meaning that approximately 90% of the 'forest' pixels in the classified image actually represent 'forest' on the ground.

The *average accuracy* is calculated as the sum of the accuracy figures in column Accuracy divided by the number of classes in the test set.
The *average reliability* is calculated as the sum of the reliability figures in column Reliability divided by the number of classes in the test set.
The *overall accuracy* is calculated as the total number of correctly classified pixels (diagonal elements) divided by the total number of test pixels.

From the example above, you can conclude that the test set classes 'crop' and 'urban' were difficult to classify as many of such test set pixels were excluded from the 'crop' and the 'urban' classes, thus the areas of these classes in the classified image are probably underestimated. On the other hand, class 'bare' in the image is not very reliable as many test set pixels of other classes were included in the 'bare' class in the classified image, thus the area of the 'bare' class in the classified image is probably overestimated.

Note:
The results of your confusion matrix highly depend on the selection of ground truth / test set pixels. You may find yourself in a situation of the chicken-egg problem with your