FEATURE EXTRACTION IN HIGH-RESOLUTION RASTER IMAGES USING NEURAL NETWORKS

Adam IWANIAK[†], Tomasz KUBIK[‡], Witold PALUSZYŃSKI[‡], Przemysław TYMKÓW[†]

[†]Faculty of Environmental Engineering and Geodesy
Agricultural University of Wrocław, Poland

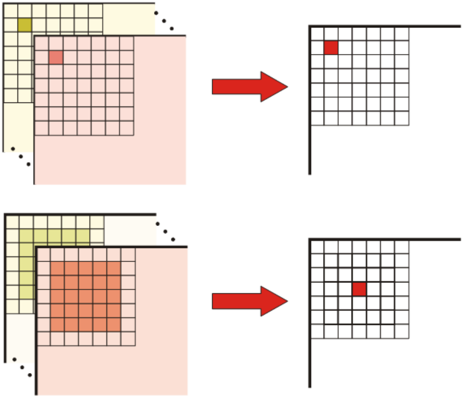
[‡]Institute of Computer Engineering, Control and Robotics
Wrocław University of Technology

Wrocław, 24 June 2006

Motivation

- Recognize features and objects on high-resolution raster images (such as: buildings, roads, or trees, etc.) in unmanned way.
- Provide support for solving problems related to the raster images processing.
- Be in line with recent developments in GIS (providing texture processing algorithms).
- ...

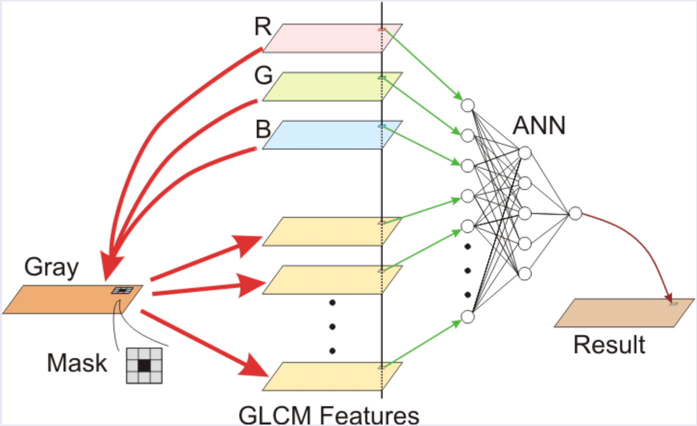
Methods of data processing



What kind of features should be used? For example: RGB intensities, spectral intensities, various statistical features, textural features, etc.

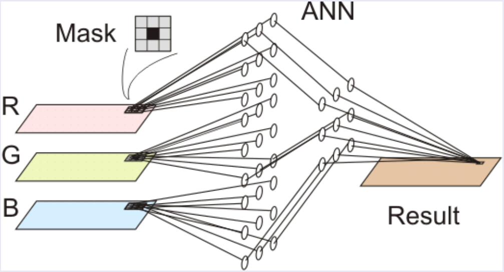
Methods of data processing

Processing by the use of RGB and GLCM based features



Methods of data processing

Processing by the use of RGB with a moving mask



GLCM and texture features

Gray level co-occurence matrix (GLCM):

$$V_{l,\alpha}(i,j) = |\{((r,s),(t,v)) : I(r,s) = i, I(t,v) = j\}|$$

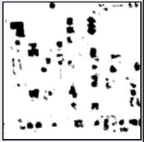
$i,j = 0,\ldots,N-1$, l,α - distance and direction angle,
N - number of gray levels, $I(x,y)$- image pixel at position (x,y),
$(t,v) = (r + l\cos\alpha, s + l\sin\alpha)$.

Symmetrical and normalized matrix:

$$P_{i,j} = \frac{\bar{V}_{l,\alpha}(i,j)}{\sum_{i,j=0}^{N-1} \bar{V}_{l,\alpha}(i,j)} \quad \text{where} \quad \bar{V}_{l,\alpha} = \frac{V_{l,\alpha} + V_{l,\alpha}^T}{2}$$

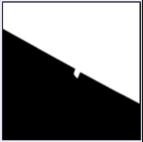
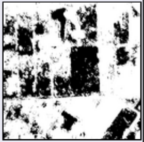
contrast	dissimilarity	homogenity	max	ASM	energy	entropy
$\sum_{i,j=0}^{N-1} P_{i,j}(i-j)^2$	$\sum_{i,j=0}^{N-1} P_{i,j}\,\lvert i-j\rvert$	$\sum_{i,j=0}^{N-1} \frac{P_{i,j}}{1+(i-j)^2}$	$Max(P_{i,j})$	$\sum_{i,j=0}^{N-1} P_{i,j}^2$	$\sqrt{\text{ASM}}$	$\sum_{i,j=0}^{N-1} P_{i,j}(-\ln P_{i,j})$

Training data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification		
			25×25	15×15	7×7
Trees					
MBuildings					

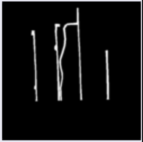
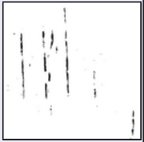
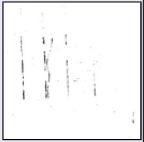
Experimental results

Training data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification		
			25×25	15×15	7×7
Forests					
LFields					

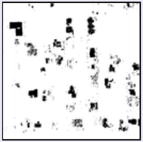
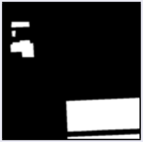
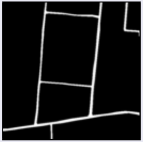
Experimental results

Training data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification		
			25×25	15×15	7×7
GRoads					
Water					

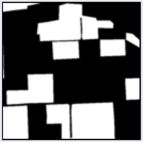
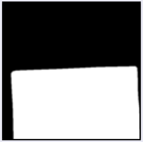
Experimental results

Training data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification	
			GLCM+RGB	RGB 7×7
MBuildings				
PBuildings				
URoads				

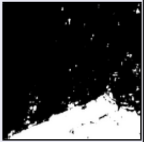
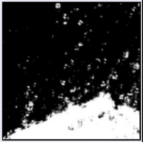
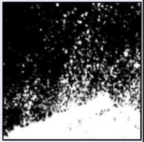
Experimental results

Training data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification	
			GLCM+RGB	RGB 7×7
ZFields				
MUrban				
PUrban				

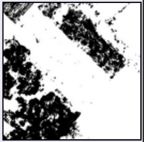
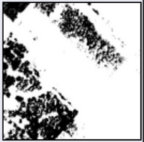
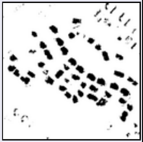
Experimental results

Testing data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification		
			25×25	15×15	7×7
Forests					
LFields					

Experimental results

Testing data (raster images with resolution of 1m)

Type	Original image	Reference image	Results of classification		
			25×25	15×15	7×7
Trees					
MBuildings					

Measures of classification quality

For the confusion matrix $A = [a_{ij}]$ where a_{ij} is the number of sample pixels from the jth class that have been classified as belonging to the ith class

- $u_i = \frac{a_{ii}}{a_{ri}}$, where $a_{ri} = \sum_i a_{i.}$
- $p_i = \frac{a_{ii}}{a_{ci}}$, where $a_{ci} = \sum_i a_{.i}$
- $d = \sum_i a_{ii}/a_t$, where a_t is the total number of pixels
- $\hat{\kappa} = (P_o - P_e)/(1 - P_e)$ where $P_o = \sum_i a_{ii}/a_t$ and $P_e = \sum_i a_{ri}a_{ci}/a_t^2$
- $uDW = \frac{1}{K}\sum_i err_i$ where

$$err_{ij} = \begin{cases} \frac{a_{ij}}{a_{ri}} & \text{if } a_{ri} > 0 \text{ and } i \neq j \\ 0 & \text{otherwise} \end{cases}$$

$$err_i = \sum_j err_{ij} = \begin{cases} 1 - \frac{a_{ii}}{a_{ri}} = 1 - u_i & \text{if } a_{ri} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Experimental results

Quality measure for the examples of *Testing Set*								
Type	Mask	κ	u_1	u_2	p_1	p_2	d	uDW
Forest	7×7	0.857	0.997	0.810	0.635	0.999	0.857	0.192
	15×15	0.926	0.995	0.894	0.815	0.997	0.926	0.111
	25×23	0.939	0.995	0.912	0.849	0.997	0.939	0.093
LFields	7×7	0.898	0.788	0.924	0.701	0.949	0.898	0.288
	15×15	0.888	0.759	0.919	0.690	0.941	0.888	0.322
	25×23	0.877	0.717	0.918	0.691	0.926	0.877	0.366
Trees	7×7	0.914	0.348	0.941	0.218	0.968	0.915	0.710
	15×15	0.926	0.441	0.937	0.139	0.986	0.926	0.622
	25×23	0.920	0.418	0.946	0.288	0.969	0.920	0.636
MBuild.	7×7	0.975	0.806	0.977	0.325	0.997	0.975	0.217
	15×15	0.976	0.783	0.979	0.392	0.996	0.976	0.237
	25×23	0.977	0.753	0.982	0.461	0.995	0.977	0.265

Summary

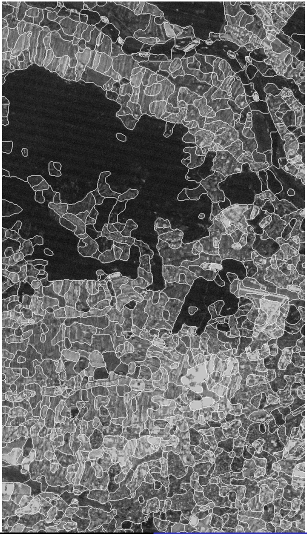
- The experiments have been conducted with color aerial photographs, as well as some satellite images converted to RGB color space, at 1m and 0.5m resolutions.
- Various architectures of feed-forward neural networks have been experimented with.
- Along with RGB intensities, including a pixel's surrounding neighborhood (mask), textural features have been used as inputs for classification.
- Visual evaluation was used to judge the results, as well as more objective statistical classification error measures.
- The results from these experiments were good or excellent for some types of spatial features (water and some types of forests), satisfactory for some other features (fields), and barely acceptable for still other features (urban areas or roads).

Conclusions

- Different neural network configurations and processing parameters (mask size), are required to achieve optimal results for various kinds of spatial features.
- Texture parameters enhance recognition for some features, but worsen it for others
- It appeared that for greater masks computation was not reasonable due to the processing time
- The results presented here are highly promising, but more work is needed. Areas to be covered:
  - comparing our results to those obtained by traditional approaches, eg. manual, simple spectral analysis, filtering, etc.
  - (guidelines for) semi-automatic neural network configuration and training for a specific processing task
  - image preprocessing to account for: differences in image resolution, spectral/color balance characteristics od the imaging system, light quality, etc.

Conclusions

- It is a good idea to combine method proposed with other methods, for example with EdgeFlow Segmentation



Conclusions

Thank you