

# IMPACTS OF A RESOLUTION PYRAMID ON GIBBS RANDOM FIELD CLASSIFICATION

Christian Becker\*, Torsten Büschenfeld\*, and Joern Ostermann

Institut für Informationsverarbeitung  
Leibniz Universität Hannover  
Appelstraße 9a, 30167 Hannover, Germany  
{becker, bfeld, ostermann}@tnt.uni-hannover.de  
<http://www.tnt.uni-hannover.de/project/wipka/>

**KEY WORDS:** Segmentation, Texture, Classification, Multiresolution, Hierarchical, Spatial, IKONOS

## ABSTRACT:

In this paper we describe how to extend the Markov/Gibbs random field model used for texture classification to a multi-resolution approach in the context of land cover analysis. Our method allows a scale-invariant analysis of image data by applying single scale classification on different resolution levels of the input image. Finally, the independent but highly correlated outcomes are joined. Evaluating a set of four different landcover classes, sampled in five different resolutions leads to 15000 combination results. Based on ground truth data of land coverage, all results were analysed for their correctness. We present an appropriate strategy, using reclassification results to dynamically choose one combination. Our experiments expose a clear tendency towards certain combinations which support the proposed approach. By prioritising those resolutions with a better reclassification result we achieve a median correctness ratio of 99% when incorporating all regions of our test scene.

## KURZFASSUNG:

Dieser Aufsatz zeigt, wie das 'Markov/Gibbs random field'-Modell zur Texturklassifikation im Kontext der Landnutzungsanalyse um einen Multiresolutionsansatz erweitert werden kann. Die dargestellte Methode ermöglicht es, eine skalierungsunabhängige Analyse der Bilddaten durchzuführen. Zunächst wird die Klassifikation auf jeder Ebene einer aus dem Eingangsbild erstellten Auflösungspyramide durchgeführt. Im Anschluss werden die einzelnen, stark korrelierten Klassifikationsergebnisse zusammengeführt. Durch das Auswerten von vier Landnutzungsklassen, die in fünf verschiedenen Auflösungsstufen vorliegen, ergeben sich 15000 mögliche Kombinationen, die Ergebnisse zu vereinen. Basierend auf Referenzdaten für die Landnutzung sind alle Resultate auf ihre Korrektheit hin untersucht worden. Es wird eine Strategie vorgestellt, mit der eine Kombination dynamisch anhand der Reklassifikation gewählt. Experimente zeigen eine deutliche Tendenz zu bestimmten Auflösungen, welche die vorgestellte Strategie bestätigen. Indem Auflösungen mit einem höheren Reklassifikationsergebnis bevorzugt werden, erreichen wir eine Korrektheitsrate von 99% bezogen auf den Median sämtlicher im Testdatensatz enthaltenen Regionen.

## 1 INTRODUCTION

Today, automation and semi-automation for quality assurance processes steadily gain more interest due to increasing amounts of mostly digital data. When looking at land cover analysis, not only is digital imagery used for interpretation, but as well for verification and update of tagged regions stored in GIS databases. Thus, the efficiency of today's workflow – which is generally done manually – can be greatly improved by using classification algorithms. Apart from fully automated systems, semi-automated systems which recommend particular regions to be verified manually, provide a very robust, yet still efficient way to disburden human operators.

In our knowledge-based interpretation and verification system GeoAIDA (Becker et. al., 2008) several operators are used to hierarchically analyse image data. Area processing is done using a classification system which incorporates a Markov/Gibbs random field model as proposed by Gimel'farb (Gimel'farb, 1996). Results of the interpretation show an overall very high quality, anyhow, they vary both in quality and accuracy, depending on parameters of the input data, e. g. spatial resolution of the images. Furthermore, land cover classes feature diverse structure information in different scales. Hence, we extend the classification approach, using a multi-resolution method.

We use the single scale Markov/Gibbs random field model to

process each level of a resolution pyramid, consisting of subsequently subscaled images. This publication aims at presenting an appropriate strategy to merge segmentation results of the resolution pyramid. An overview of this process can be seen in Figure 1. The Markov/Gibbs random field model, providing the

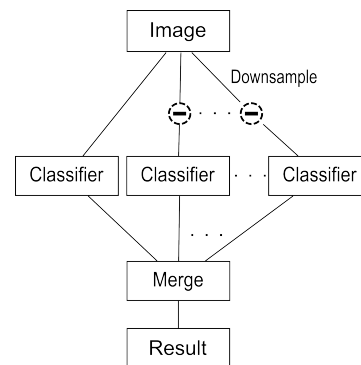


Figure 1: System structure

foundation for our texture classifier, proved itself reliable, robust and accurate for practical application in a semi-automatic verification and update system for GIS data (Helmholz et. al., 2007). For an introduction to Markov random fields and Gibbs distribu-

tions see (Pérez, 1996).

## 2 FEATURE EXTRACTION USING A MARKOV/GIBBS RANDOM FIELD MODEL

As already stated in section 1, the Markov/Gibbs random field model is used as a classifier for each resolution level. In the following, this classifier will be referenced as *Gibbs-classifier*. The model is characterised by describing the image as a Markov field while incorporating Gibbs probability distributions. Using a local region at a time, a pixel-wise classification result is obtained.

With the Markov/Gibbs random field, textural similarities are modeled by several pairwise pixel interactions. Within the image, i. e. the 2D lattice Markov random field, structures are described by so called clique families, that basically contain information about characteristic pixel pairs and their distance to each other. The probability of a clique is modelled by the Gibbs potentials. The higher this potential, the more probable is this specific pixel configuration.

If limited to a single channel, the major information is given by intensity differences for different distances. Thus, Grey Level Difference Histograms (GLDHs) are used for feature extraction. Referring to multiple channels, we also use inter channel difference histograms. Three channels are used for texture analysis. Since we aim at an area-wide classification including both nature and settled area, IRG (Infrared, Red, Green) images are generally better suited (due to chlorophyll-caused Ir emissions) than RGB (Red, Green, Blue).

## 3 MULTI-RESOLUTION IMAGING

Not surprisingly, the quality and composition of input images is very important for texture classification. One crucial aspect is the spatial resolution. The higher the resolution, the more detailed the observed scene. While such detailed information can be indispensable for some applications, it also might provide irrelevant data for others. Detailed structures may suffer from irregularity within local processing areas, which complicates classification. Apart from this, high spatial resolution requires massive amounts of data to process, which is neither desired nor always needed. Hence, when aiming at optimal input data for a classification problem, resolution is one of the key aspects.

In our project, classification is used for segmenting the scene by given usage classes, e. g. *settlement*, *cropland*, *industry*, and *forest*. As already described in section 2, this classification is done pixel-wise by incorporating a local neighbourhood around each pixel.

The type of data that is contained in a region depends on the scene, spatial resolution and region size. The importance of resolution and region size for recognition can easily be seen in Figure 2. Both images show a  $32 \times 32$  pixel section with texture from a settlement. In the image on the left part of the figure each pixel corresponds to an area with  $1\text{m}^2$ . While some structures like houses (with shadows) and gardens can be recognised, it would be hard to distinguish between different classes of population density. With smaller region size, this would be even more problematic. On the right an 8m resolution image is depicted, which includes the 1m cutout as well. Here, every pixel covers  $64\text{m}^2$ . A region of this size covers several houses, streets and gardens. While it would be not possible to distinguish individual houses, the nature of the settlement stands out nicely.

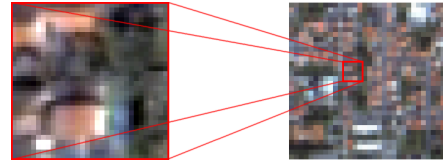


Figure 2: 1m resolution cutout (l) of 8m resolution texture (r). Both windows are  $32 \times 32$  pel.

In our application, the texture classifier has to differentiate between several classes, that is textures. These textures might vary greatly in terms of area coverage i.e. how detailed the texture is structured. Thus, there is nothing like a fixed region size or resolution so that such a classifier might work optimally. Another aspect limiting the appropriate region size is that with growing sizes processing time is increasing non-linearly. Since our project is aiming to be used for production and quality assessment of geo-spatial data on a regular basis, processing time is a crucial point. Hence, an approach to use small region sizes without crippling the texture classification is needed: multi-hierarchical classification.

For maintenance reasons the multi-hierarchical approach is not included into the Gibbs-classifier itself. Therefore, further single-scale segmentation methods can be easily adopted to the framework. We opted for a structure as depicted in Figure 1. The same classifier is used in several runs. Each run classifies the same scene, but with decreasing image resolution. The results are finally merged to an overall result.

Merging several classification results is a non-trivial task. When classifying an unknown scene, no result is per se superior to a competing result. The only way to obtain a merging strategy is by deriving it from a training sample.

### 3.1 Merging Strategy

From the supervised training of the Gibbs-classifier a training set is available, that can be used to determine a merging strategy. Given a perfect training sample and loose time constraints, the optimal strategy could be found through a brute force approach: By evaluating every possible combination of results the best strategy can be found. Unfortunately both requirements cannot be fulfilled in practice. Our system has to cope with fast changing scenarios. Thus, training has to be done quite regularly and fast. To always have access to a perfect training sample is even more limiting. Without such a sample a brute force approach is very prone to overfitting and other problems. For a robust algorithm we needed to introduce some constraints.

The initial classification steps are as follows: each texture class has a resolution level on which it can be segmented optimally, depending on structure size of the observed objects. The level might not be identically for different classes, though. The best level for each class is estimated by reclassifying the training sample:

1. Train classifier with training data.
2. Reclassify training data.
3. Compute (re-)classification result using training data.
4. Downsample training data and proceed with step 1.

From this we get one reclassification quality result for each resolution level. Each result shows how a trained texture class performs on a given resolution. Just determining the best resolution

level for each class and superimposing them into a final result is not optimal, though. In the general case the results are conflicting. We came up with a solution that uses these conflicts to gain an advantage:

Class-wise true and false positives are calculated, then normalised by the resolution and region size. From this data we especially obtain which resolution is best suited for a given class, of course limited to the offered resolution levels. For each class  $c$ , the resolution level which maximises the term  $q(c) := t_p - f_p$  with  $t_p$  true positive,  $f_p$  false positive is selected. The result for a class  $c$  dominates the result for a class  $c'$  if  $q(c) \geq q(c')$ .

$q(c)$  expresses how accurate a class  $c$  can be classified. It also penalises all-too tolerant segmentation. Classification results for the classes are written to a label image in a subsequent manner. Thereby, results might be overwritten, hence, the insertion order affects the result. In detail, it allows better results to correct weaker results' false negatives. Thus, good results improve weaker results. All pixels, that have not been classified yet, adopt the classification from the best single-resolution result.

## 4 RESULTS AND DISCUSSION

### 4.1 Input and Reference Data Set

As ground-truth data for evaluation of our results, area coverage information from the German ATKIS (see (AdF, 1997)) was used. Coverage classes *forest*, *settlement*, *cropland*, and *industry* were selected for verification, nearly covering the test scene completely. The scene is divided into 2606 regions (ATKIS objects) belonging to either one of the four classes. This high quality data was used to generate reference label images with 1m resolution. ATKIS object information was kept to enable independent evaluation of each region. This includes determination of the ratio of success, i. e. how many pixels of a region have been classified correctly. Therefore, the contribution of the regions to the final result is independent of their size. This setup and evaluation meets the requirements for texture classification used for production or quality assessment of geo-spatial data.

In our setup we are classifying IKONOS satellite imagery pan-sharpened to a resolution of 1m. We are using the red, green and near-infrared channels. The Gibbs-classifier is set to a fixed region size of  $32 \times 32$  pixels. We allowed for five resolution levels: 1, 2, 4, 8 and 16 meters. Our strategy was tested on a scene showing the surroundings of Weiterstadt/Germany, covering an area of about  $5.7 \text{ km}^2$  ( $2340\text{m} \times 2432\text{m}$ ). The classifier was trained with manually selected samples. They were taken from a nearby area, however, they were not part of the scene used for our classification tests.

### 4.2 Reclassification Results

From the sample described in section 4.1, the classifier generated the parameters for each resolution level as described in section 2. The multi-resolution strategy was determined according to the approach in section 3. Reclassification results for all resolutions are shown in Table 1. The table has to be read for each texture class and resolution level. On the diagonals, written in bold, the true positives for each texture class can be found. Other values in the same row are false negative, while other values in the same column (and for the same resolution) are false positive values. The remaining values show the true negatives. It can be noticed that results vary greatly between resolution levels, and also within each level. The 8m level provides the best

Resolution	Classes	Ind.	Set.	For.	Cro
1m	Industry	<b>42.5</b>	21.4	4.2	31.8
	Settlement	18.8	<b>51.4</b>	14.3	15.4
	Forest	0.1	1.4	<b>96.5</b>	2.0
	Cropland	2.2	1.4	0.5	<b>95.8</b>
2m	Industry	<b>38.6</b>	16.1	3.5	41.8
	Settlement	4.5	<b>75.4</b>	8.9	11.1
	Forest	0.1	0.1	<b>98.1</b>	1.7
	Cropland	0.4	0.5	0.3	<b>98.8</b>
4m	Industry	<b>49.4</b>	14.1	1.7	34.8
	Settlement	1.3	<b>90.8</b>	2.5	5.4
	Forest	0.1	0.0	<b>98.7</b>	1.2
	Cropland	0.3	0.7	0.0	<b>99.0</b>
8m	Industry	<b>76.0</b>	10.1	2.3	11.7
	Settlement	0.1	<b>97.2</b>	1.2	1.6
	Forest	0.0	0.0	<b>99.5</b>	0.4
	Cropland	0.0	0.2	0.5	<b>99.3</b>
16m	Industry	<b>16.8</b>	56.2	8.2	18.8
	Settlement	0.0	<b>94.1</b>	4.6	1.3
	Forest	0.0	1.2	<b>98.7</b>	0.1
	Cropland	0.0	6.4	2.3	<b>91.3</b>

Table 1: Reclassification results

classification result, as can be seen when summing up the diagonals for each level. When searching for the best level for each texture class independently, again the 8m level is determined to be superior for each class. Thus, the 8m resolution is chosen by our algorithm. Since results within one resolution level do not compete, the result is independent from the insertion order in this case.

### 4.3 Scene Evaluation Results

To test our merging strategy, we compare the results to all possible combinations of levels and class order: Each class can appear in any of five resolution levels. Moreover the results can be merged in any order. This leads to  $5^4! = 15000$  combinations.

In Figure 3 the quality of all possible combinations can be found. To get a one-dimensional order for those combinations, we introduce the term abstraction value: The resolutions (in meter) of a combination are summed up, linearly weighted by the insertion order. By doing this, it is taken into account that previous results can be altered by following results. In short, a high abstraction value denotes high influence from low resolution results. The quality for a result is measured by averaging all true positive ratios. This is feasible, because false positives, true negatives and false negative ratios for a class are effecting the true positive values of other classes. For averaging, we use the median so that outliers are not influencing the result too strongly. The diamond shaped entry represents the combination chosen from our algorithm. On first sight it becomes clear, that a multi-resolution approach has great impact on the performance of the system. The quality between the possible combinations are varying heavily, from 0.68 for the worst to 0.99 for the best result. The combination chosen by our algorithm performs nearly optimal.

Apart from this, some more results are emphasized in the graph. If looking at Table 1 again, one can see, that the results for cropland hardly differ within resolutions of 2m, 4m, and 8m. The same applies for forrest within the 4m, 8m, and 16m resolution. Therefore, assigning these differences to noise, we evaluated exemplary results to show the robustness of our approach. Using the proposed merging strategy yields to high quality results as well, which are depicted as cyan circles.

In Figure 4 the variance of the quality term within each combination is shown. As can be seen, the variance is increasing

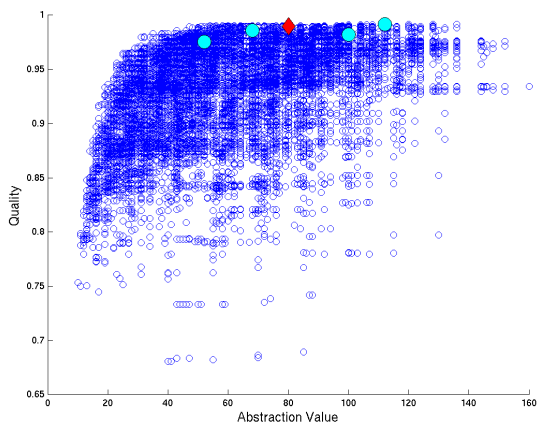


Figure 3: Mean quality for each combination

monotonously with increasing abstraction value. This effect is due to objects having complex shape or being very small. Such objects suffer disadvantages in low resolutions. Furthermore, the samples are roughly divided into two clusters. The upper cluster is formed by combinations with high impact of the lowest resolution (16 meters). This resolution leads to unreliable results.

Now we want to take a closer look at the impact of resolution

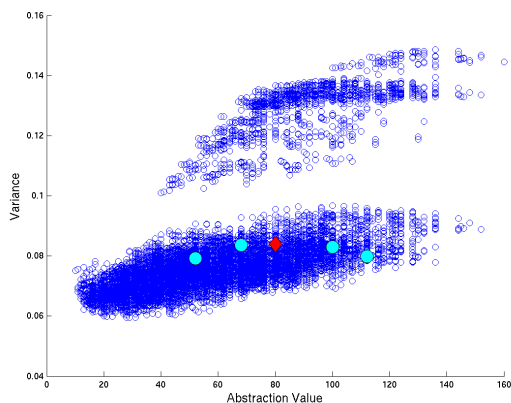


Figure 4: Variance of quality for each combination

and insertion order separately. In the left diagram in Figure 5 we show how much the quality varies when using a fixed combination of resolutions but changing the classes for them. This shows the importance of the order for inserting results of the different texture classes. In the right diagram, the order in which the classes are used for the result is fix, but we vary the resolution for each of the classes. For lack of a meaningful order for the groups we limit this diagram to one dimension only. It appears, resolution and insertion order are both affecting the performance of the system in equal parts.

## 5 CONCLUSIONS

As shown in section 4, the use of several resolution levels is necessary to obtain optimal results for scale invariant texture classification. The relevance of different resolutions was depicted in the previous section. Here, correctness of classification referring to ground truth data of the whole scene ranges from 0.68 to 0.99. The proven Markov/Gibbs model is still an adequate foundation

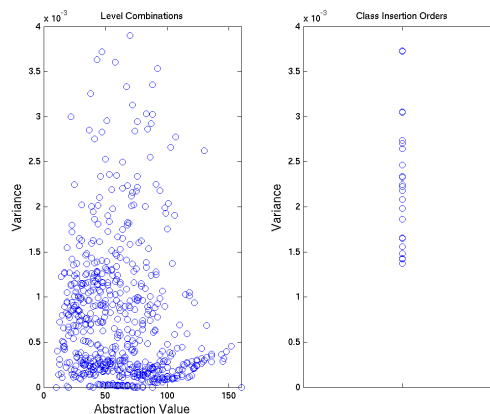


Figure 5: Impact of resolution level

for today's classification tasks in remote sensing. A brute force comparison of all configurations depending on merging order and resolution level was evaluated. To stay flexible, a merging strategy is needed, which only incorporates information known at the time of learning. We presented a simple yet fast and reliable approach to combine the resolution levels. Since the training process offers reclassification results as a rough quality measure, resolutions with a higher reclassification are prioritised in the merging process. The order is then given by a classification quality (certainty), which penalises false negative results.

The experiments clearly show that classification is optimal within our test set, while using four landcover classes. As shown, with our strategy a correctness rate of 99% is achieved.

## ACKNOWLEDGMENTS

The work is funded by BKG, Frankfurt (Main) and AGeoBw, Euskirchen, Germany. We gratefully acknowledge this support.

## REFERENCES

- Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der Bundesrepublik Deutschland (AdF), 1997. "ATKIS – Amtlich Topographisch-Kartographisches Informationssystem", Germany. <http://www.atkis.de> (accessed 14.04.2009)
- Becker, C., Ziems, M., Büschenfeld, T., Heipke, C., Müller, S., Ostermann, J., Pahl, M., 2008. Multi-Hierarchical Quality Assessment of Geospatial Data. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Beijing, China, Vol. XXXVII, Part B2, pp. 779 ff.
- Gimel'farb, G. L.; 1996. Texture Modeling by Multiple Pairwise Pixel Interactions. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(11), pp. 1110–1114.
- Helmholz, P., Becker, C., Heipke, C., Müller, S., Ostermann, J., Pahl, M., Ziems, M., 2007. Semi-Automatic Verification of Geo-data for Quality Management and Updating of GIS. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, Urumchi, Xinjiang, China, Vol. XXXVI, Part 4, pp. 9–13.
- Pérez, P.; 1998. Markov Random Fields and Images. *CWI Quarterly*, 11(4), pp. 413–437.