Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

MIN-Fakultät
Fachbereich Informatik
Arbeitsbereich SAV/BV (KOGS)

IP2: Image Processing in Remote Sensing
10. Image Processing I: Classification and Segmentation

Summer Semester 2014

Benjamin Seppke

Agenda

- The Classification task
  - Photointerpretation
  - Quantitative Analysis
- Supervised Classification
  - Maximum Likelihood
  - Minimum Distance
  - Context Classification
- Unsupervised Classification
  - (Basic) Clustering
  - Hierarchical Clustering

The Classification Task

- Origin: Photointerpretation
  - Remote Sensing Experts “visually inspect” air/space borne images
  - Search for “meaningful objects”
- Computer based: Quantitative analysis
  - Vision system classifies image areas and/or meaningful objects automatically
  - No classification expert needed → Decision based on the software results
- In practice: Computer-aided/assisted classification
  - Image processing / Computer Vision Software predicts potential areas of interest
  - Expert focuses on proposed regions

Photointerpretation vs. Quantitative Analysis

Photointerpretation (by human experts)	Quantitative analysis (by computer)
On a large scale relative to the pixel size	At individual pixel level
Inaccurate area estimates	Accurate area estimates possible
Only limited (visible) spectral analysis	Can perform true multispectral analysis
Can assimilate only a limited number of distinct brightness levels (say 16 levels in each feature)	Can make use of all available brightness levels in all features (e.g. 256, 1024, 2048)
Shape determination is easy	Shape determination involves complex software decisions
Spatial information is easy to use in a qualitative sense	Limited techniques available for making use of spatial data

from Richards, 2006

Example: Photointerpretation

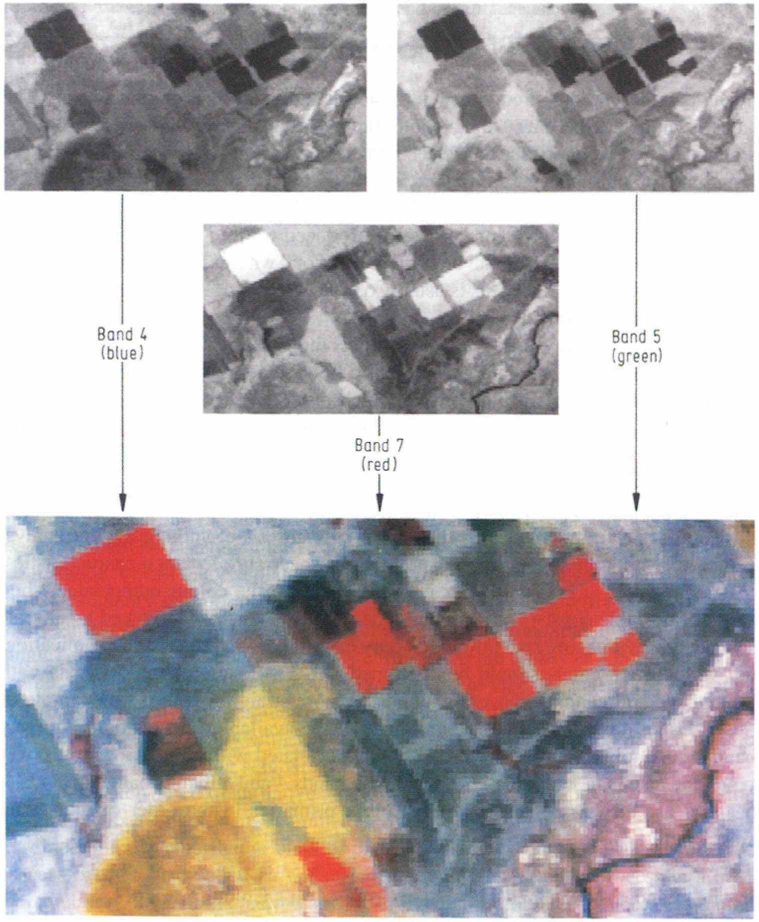


Fig. 3.1. Formation of a Landsat multispectral scanner false colour composite by displaying the infrared band as red, the visible red band as green and the visible green band as blue

from Richards, 2006

Example: Quantitative Analysis

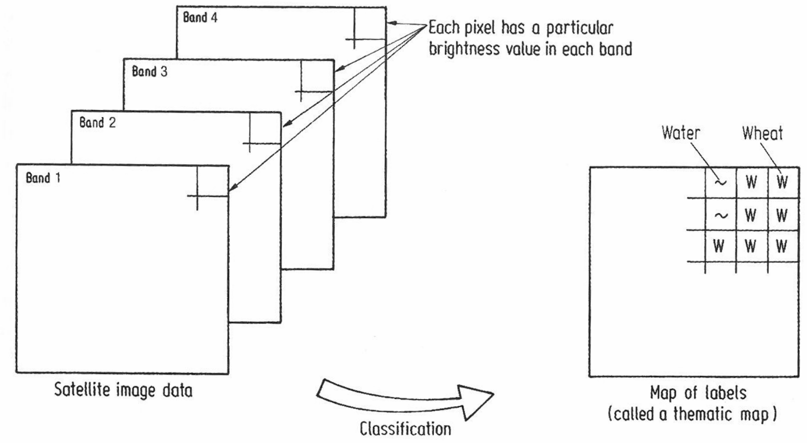

Band 4
Band 3
Band 2
Band 1
Each pixel has a particular brightness value in each band
Water
Wheat
~
W
Satellite image data
Classification
Map of labels (called a thematic map)

Fig. 3.7. The role of classification in labelling pixels in remote sensing image data

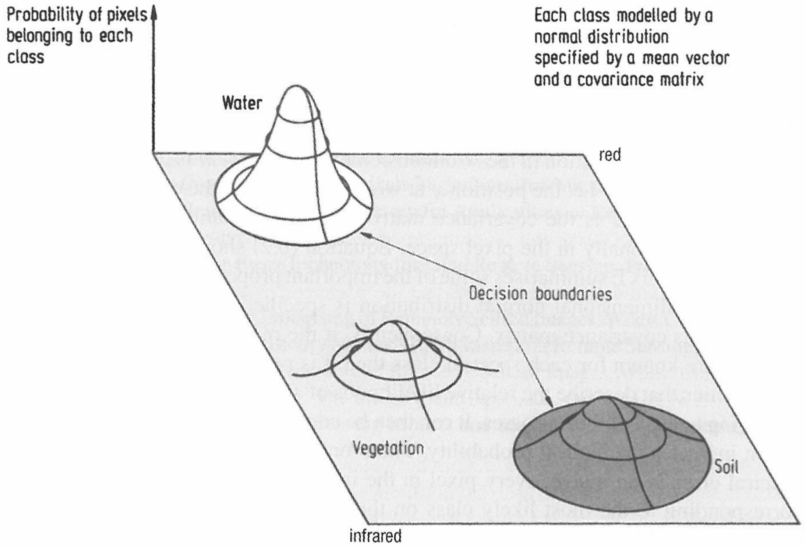

Probability of pixels belonging to each class
Each class modelled by a normal distribution specified by a mean vector and a covariance matrix
Water
red
Decision boundaries
Vegetation
Soil
infrared

Fig. 3.8. Two dimensional multispectral space with the spectral classes represented by Gaussian probability distributions

from Richards, 2006

Classification Techniques

- Supervised Classification
  - Maximum likelihood classifiers
  - Minimum distance classification
  - Context based approaches
  - Machine learning approaches
- Unsupervised Classification (Clustering)
  - "Greedy" Clustering (migrating means)
  - k-Means clustering
  - Hierarchical clustering

Steps in Supervised Classification

1. Decide the set of ground cover types into which the image is to be segmented. Exemplary classes: Water, urban regions, farmland etc.
2. Choose representative/prototype pixels from each of the desired set of classes. These pixels are said to be the ***training data***. If compact, the range of the training data is referred to as ***training field***. (< 1% of data)
3. Use the training data to estimate the parameters of the particular classifier to be used. The set of parameters is said to be the signature of that class.
4. Use the trained classifier to label/classify each pixel in the image with one of the desired ground cover types. (> 99% of the data)
5. Produce tabular summaries or thematic maps of the classification
6. Assess the accuracy of the final result using a labeled ***test data*** set.

Maximum Likelihood Classification

- Most common supervised classification method w.r.t Remote Sensing data.
- Based on Bayes‘ classification:
  - Let the spectral classes for an image be represented by:
    $\omega_i, \quad i = 1, \ldots M$
  - The determination of a class for a pixel vector $\vec{x}$ is based on the conditional probabilities:
    $p(\omega_i \mid \vec{x}), \quad i = 1, \ldots M$
  - The classification is performed based on:
    $\vec{x} \in \omega_i, \quad if \quad \underset{i \neq j}{\forall} p(\omega_i \mid \vec{x}) > p(\omega_j \mid \vec{x})$
- Main problem: The conditional probabilities $p(\omega_i \mid \vec{x})$ are unknown!

Maximum Likelihood Decision Rule

- Assumption: Sufficient and representative training data
- Estimate probability from training data: The chance of finding a pixel of class ω_i at $\vec{x}$:

$$p\left(\vec{x} \mid \omega_i\right)$$

- The desired conditional probabilities can be derived using Bayes' theorem:

$$p\left(\omega_i \mid \vec{x}\right) = \frac{p\left(\vec{x} \mid \omega_i\right) p\left(\omega_i\right)}{p\left(\vec{x}\right)}, \quad i = 1, \ldots M$$

- Using this, the classification rule can be rewritten as:

$$\vec{x} \in \omega_i, \quad if \quad \underset{i \neq j}{\forall} \underbrace{p\left(\vec{x} \mid \omega_i\right) p\left(\omega_i\right)}_{g_i(\vec{x})=\ln(\ldots)} > \underbrace{p\left(\vec{x} \mid \omega_j\right) p\left(\omega_j\right)}_{g_j(\vec{x})=\ln(\ldots)}$$

Discriminant functions g_i and g_j.
Usually in log-space!

- Note: $p\left(\omega_i\right)$ still has to be determined by expert knowledge!

Multivariate Normal Class Models

- Assumption: All probability distributions are Gaussian

→ Mathematical simplifications and well known properties for multivariate models

- The chance of finding a pixel of class ω_i at $\vec{x}$ is now:

$$p(\vec{x} \mid \omega_i) = \frac{1}{\boxed{\sqrt[N]{2\pi}}\sqrt{|\Sigma_i|}} e^{-\frac{1}{2}(\vec{x}-\vec{\mu}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{\mu}_i)}$$

No influence on discriminant functions

- The resulting discriminant function is:

$$g_i(\vec{x}) = \ln\left(p(\omega_i)\right) - \frac{1}{2}\ln\left(\sqrt{|\Sigma_i|}\right) - \frac{1}{2}(\vec{x}-\vec{\mu}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{\mu}_i)$$

$\vec{\mu}_i$ Mean vector

Σ_i Covariance Matrix

- Since $p(\omega_i)$ is (often) the same for all i and can thus be removed:

$$g_i(\vec{x}) = -\frac{1}{2}\ln\left(\sqrt{|\Sigma_i|}\right) - \frac{1}{2}(\vec{x}-\vec{\mu}_i)^T \Sigma_i^{-1} (\vec{x}-\vec{\mu}_i)$$

Decision Surfaces and Thresholds

- Two discriminant functions are (pairwise) equal at **decision surfaces**:

  $$g_i(\vec{x}) - g_j(\vec{x}) = 0$$

- Due to the quadratic construction of multivariate normal distributions → decision surfaces are parabolas!
- To dismiss pixels with comparably low decision values, **thresholds** are introduced:

  $$\vec{x} \in \omega_i, \quad if \quad \underset{i \neq j}{\forall} g_i\left(\vec{x}\right) > g_j\left(\vec{x}\right) \quad \wedge \quad g_i\left(\vec{x}\right) > T_i$$

- Thresholds may be derived according to the used probability distribution. *An example for multivariate normal distribution can be found in Richards 2006*

Example: Thresholds

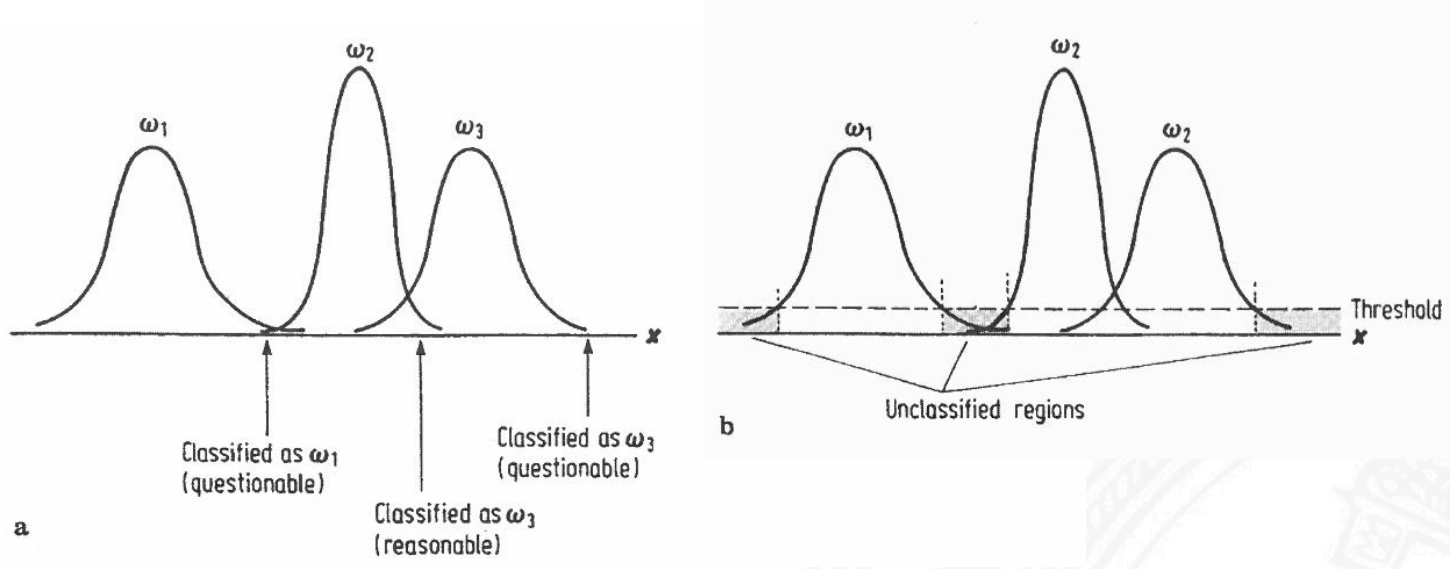


Fig. 8.1. **a** Illustration of poor classification for patterns lying near the tails of the distribution functions of all spectral classes; **b** Use of a threshold to remove poor classification

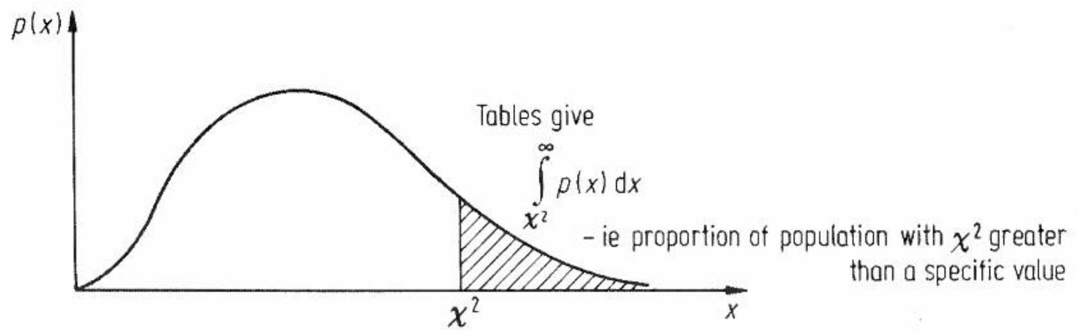


Fig. 8.2. Use of the χ^2 distribution for obtaining classifier thresholds

from Richards, 2006

Minimum Distance Classification

- Effectiveness of maximum likelihood depends on reliable estimates of the mean vector and covariance matrix (for multivariate normal distributions):
  - Problems may arise if too few training samples are available
  - Reason: Covariance matrix can not be estimated reliable
- Alternative: Rely on the estimates of the means!
  - → Minimum distance to class means classifier
  - → Only class means are trained
  - → Decision on class dependency is due to the closest mean
- Note:
  - Much faster technique
  - Less powerful w.r.t. maximum likelihood classification
  - Does not model any asymmetry, all classes are assumed to be hyper spheres in feature space

Minimum Distance Discriminant Function

- Starting with the squared Euclidian distances of a point to the class mean:

$$d\left(\vec{x},\vec{\mu}_i\right)^2 = \left(\vec{x}-\vec{\mu}_i\right)^T\left(\vec{x}-\vec{\mu}_i\right) = \left(\vec{x}-\vec{\mu}_i\right)\cdot\left(\vec{x}-\vec{\mu}_i\right)$$

- Expanding the product gives:

$$d\left(\vec{x},\vec{\mu}_i\right)^2 = \vec{x}\cdot\vec{x} - 2\vec{\mu}_i\cdot\vec{x} + \vec{\mu}_i\cdot\vec{\mu}_i$$

- Classification is based on:

$$\vec{x}\in\omega_i,\quad \mathit{if}\quad \underset{i\neq j}{\forall} d_i\left(\vec{x},\mu_i\right)^2 < d_j\left(\vec{x},\mu_j\right)^2$$

- Since the square of the point itself is common to all distances, it may be removed. The discriminant function is:

$$\vec{x}\in\omega_i,\quad \mathit{if}\quad \underset{i\neq j}{\forall} g_i\left(\vec{x}\right) > g_j\left(\vec{x}\right)$$

$$\mathit{with:}\quad g_i\left(\vec{x}\right) = 2\vec{\mu}_i\cdot\vec{x} - \vec{\mu}_i\cdot\vec{\mu}_i$$

Remarks on Minimum Distance Classification

- It can be shown, that the minimum distance classification is a degraded maximum likelihood classification with degraded (constant) standard deviation:

$$\Sigma_i = \sigma^2 \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix}$$

- **Decision surfaces** are present where two class centers have the same distance to the point:

  $$2\left(\vec{\mu}_i - \vec{\mu}_j\right) \cdot \vec{x} - \left(\vec{\mu}_i \cdot \vec{\mu}_i - \vec{\mu}_j \cdot \vec{\mu}_j\right) = 0$$

  defines a linear surface, thus the decision surfaces are given as hyper planes
- **Thresholds** can be defined by means of minimum (squared) distances to a class mean.

Context Classification

- Concept of a spatial context
  - So far only classifications "per pixel"
  - But: Sensor acquired portions of energy from adjacent pixels, too!
  - Spatial neighborhoods may not be neglected
- Context sensitive methods make use of this spatial pixel neighborhoods for classification!
- Methods may become context sensitive by adding:
  - Pre-processing (e.g. median filter)
  - Post classification filtering (e.g. 3x3 window decision function)
  - Probabilistic label relaxation
    - More complex
    - Logical consistent integration of region properties w.r.t. classification process possible!

Basic Context Classification Algorithm I

- Starting point: Classification already finished:
  - Each pixel is assigned to one class (of e.g. max likelihood)
  - Assignment probabilities for other classes are existing. Let $p_m(\omega_i)$ denote the set of prob. For a pixel m:

    $$p_m(\omega_i) \quad i = 1, \ldots M$$

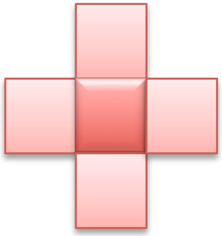
  - Additionally, all probabilities for a given m should sum up to 100%:

    $$\underset{m \in I}{\forall} \quad \sum_{i=1}^{M} p_m(\omega_i) = 1$$

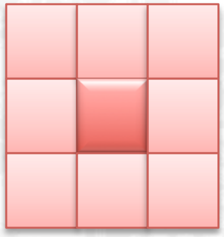
- Define a (simple) neighborhood around a pixel m:

Examples:

4-connected:



8-connected:



Basic Context Classification Algorithm II

- Neighborhood function
  - Describes the influence of neighbored pixels by means of a neighborhood function. Influence of neighbors is given by:

$$p'_m(\omega_i) = \frac{p_m(\omega_i) Q_m(\omega_i)}{\sum_{i=1}^{M} p_m(\omega_i) Q_m(\omega_i)} \qquad i = 1, \dots M$$

  - Is often applied iteratively.
- Choice of the neighborhood function
  - Compatibility of two assignments: p_{nm}
  - Here: Probability that ω_i is the correct label for pixel m if ω_j is the correct label for pixel n.
  - Sum up all neighbor contributions to compute the arithmetic mean:

$$Q_m(\omega_i) = \sum_n d_n \sum_j p_{mn}(\omega_i \mid \omega_j)\, p_n(\omega_j)$$

Basic Context Classification Algorithm III

- How to determine the coefficients for p_{nm}?
  - Spatial region model may be available (e.g. field/acre sizes)
  - Compute values from Ground Truth
- When to stop the iterative process?
  - Many (> 500) iterations → very time consuming. Really needed?
  - No! Observations:
    - Most change in the first few iterations
    - Changes are vanishing later
  - Embed controlling mechanisms:
    - Stop on Ground Truth correspondence
    - Stop on propagation limits, based on non-local decorrelation
    - Stop on change measure
  - Alternative: Reduce the neighbors' influence with iteration.

Context Classification: Basic Example

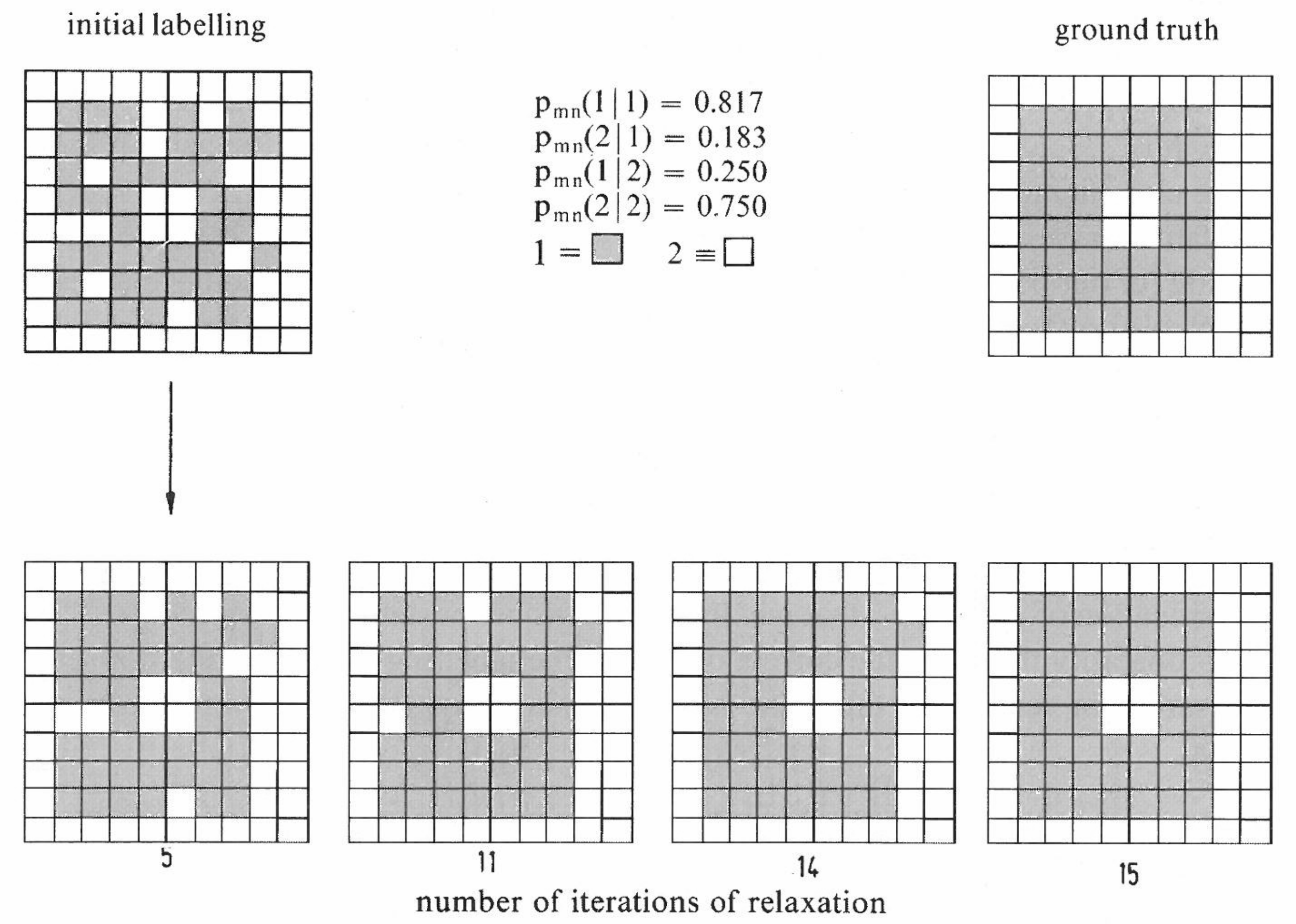


Fig. 8.9. Simple demonstration of pixel relaxation labelling

from Richards, 2006

Context Classification: Landsat MSS

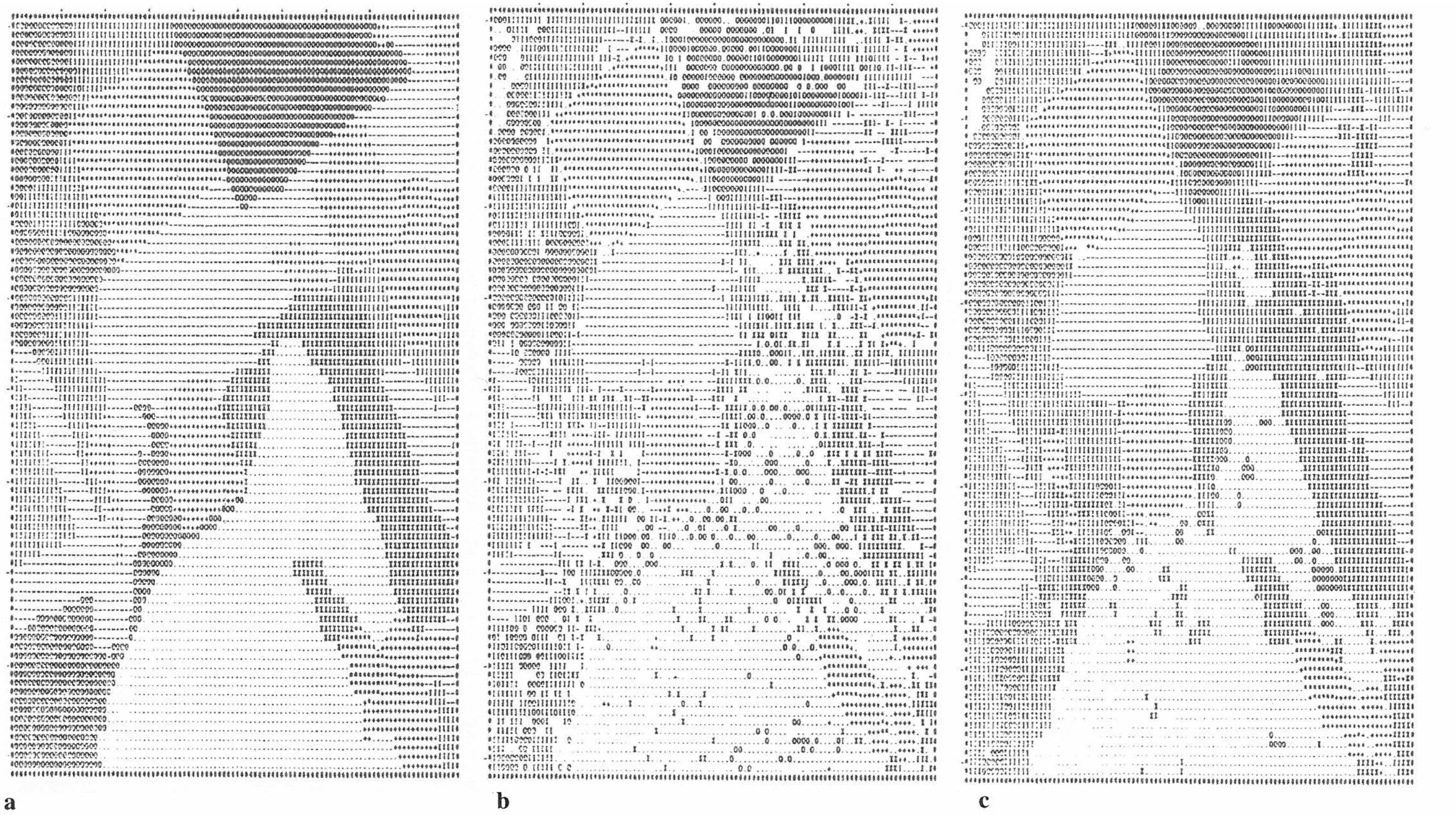


Fig. 8.10. a Ground truth for the left-hand side of the image in Fig. 3.1. The symbols are: · = red soil, ∗ = cotton crop, 0 = bare soil (low moisture), I = dry bare soil, + = early vegetation growth, X = mixed bare soil, − = bare soil (moist or ploughed). **b** Result of a maximum likelihood classification of Landsat MSS data. **c** Result of applying relaxation labelling to the result in **b**, incorporating a reduction in the neighbour weights with iteration

from Richards, 2006

Other Methods for Supervised Classification
(which will not be covered here)

- Linear discriminant functions
  - Perceptron Learning as training approach
  - Threshold decider
- Support Vector approaches
  - Basic case: Classes are linear separable
  - Use of Kernel-tricks for non-linear decisions
- Classifier Networks
  - Neural Networks
  - Multilayer perceptrons + Back propagation learning rule
- Classifier Cascades
  - Homogenous vs. heterogeneous classifiers
  - Sort-Out-Early to save computation time

Classification Techniques

- Supervised Classification
  - Maximum Likelihood Classifiers
  - Minimum Distance Classification
  - Context based approached
  - Machine Learning Approaches
- Unsupervised Classification (Clustering)
  - "Greedy" Clustering (Migrating Means)
  - k-Means Clustering
  - Hierarchical Clustering

Fundamentals of Unsupervised Classification

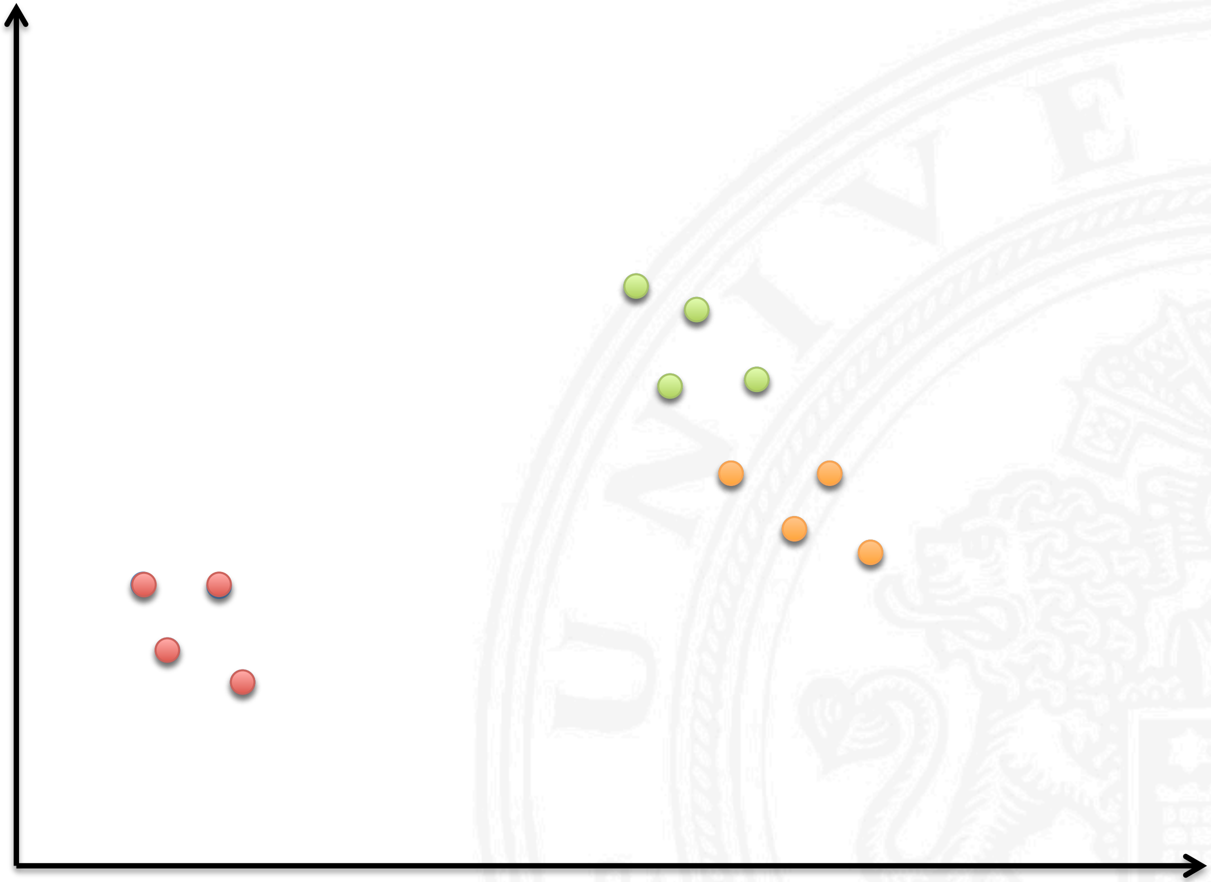
- Successful application of the maximum likelihood approach depends on:
  - Correct delineation of spectral classes
  - Unimodal normal distributions
- What if these requirements are not met?
  - Multimodal distributions are complicated to model
  - Clustering approaches are a good alternative
- Similarity metrics and clustering criteria
  - Clustering takes place in (high-dimensional) spectral/feature space
  - Criteria for clustering needed:
    - Distance measure (e.g. Euclidean) is commonly used
    - Compare with “Minimum Distance Classification”
  - Accuracy control: Sum of the squared error over all clusters

Greedy Clustering Algorithm

If the maximum distance to the cluster centers is known:

1. Assign an arbitrary point in feature space to the first cluster
2. For each (unassigned) point:
    1. Compute the distance to all cluster means
    2. If distance is below threshold t:
        1. Assign it to the cluster of min. distance and
        2. Update the clusters mean.
3. If no point was assigned to a cluster:
    1. If all points are assigned: done!
       Else:
        1. Select the point with the max. distance to all cluster means and
        2. Assign it to a newly introduced label
        3. Proceed with (2.)

Example: Greedy Clustering Algorithm



Summary: Greedy Clustering Algorithm

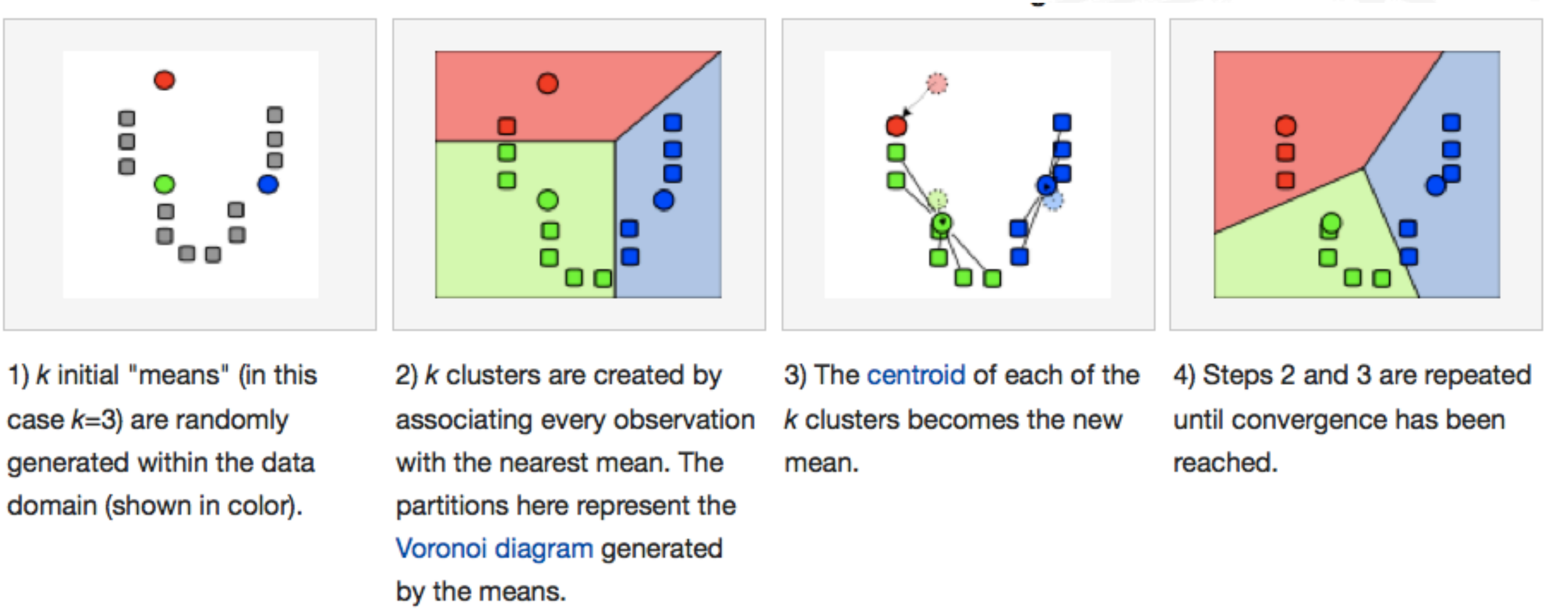
- Advantages:
  - Only maximum cluster distance needed
  - Fast approach, even for high-dimensional feature spaces with a lot of features
  - Easy to implement
- Drawbacks:
  - Strong dependency on the order of feature points
  - No correction step
  - Maximum cluster distance my be hard to determine
  - Does not take advantage of the sum of squared errors

k-Means Clustering Algorithm

If the count of clusters k is known:

1. Assign k arbitrary points in feature space to represent cluster
2. For each point:
    1. Compute the distance to all cluster means
    2. Assign it to the cluster of min. distance
3. If assignments have not changed: done!
4. Else: For each cluster:
    1. Compute the new cluster (arithmetic) mean
    2. Proceed with (2.)

Example: k-Means Clustering Algorithm



1) *k* initial "means" (in this case *k*=3) are randomly generated within the data domain (shown in color).

2) *k* clusters are created by associating every observation with the nearest mean. The partitions here represent the Voronoi diagram generated by the means.

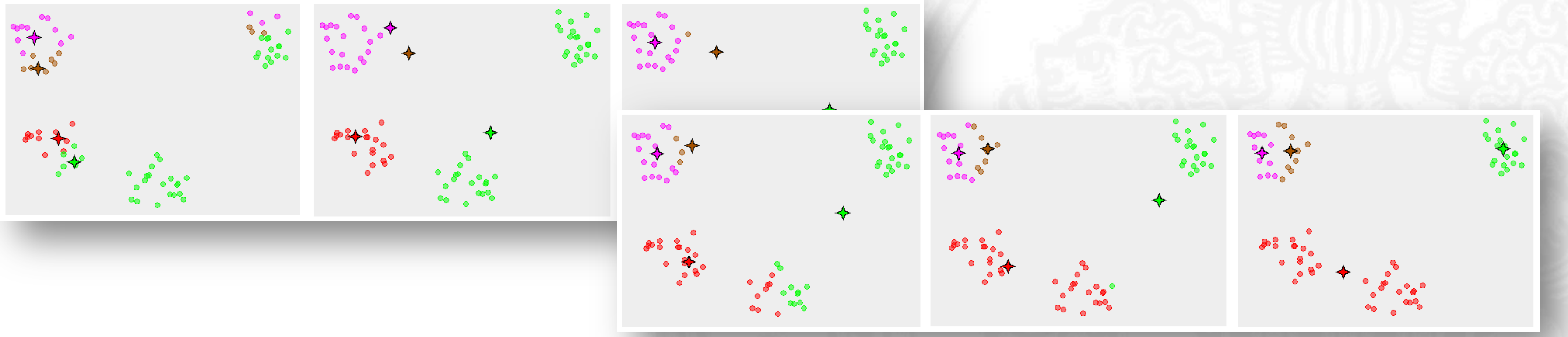
3) The centroid of each of the *k* clusters becomes the new mean.

4) Steps 2 and 3 are repeated until convergence has been reached.

from Wikipedia, 2014

Summary: *k*-Means Clustering Algorithm

- Outperforms greedy algorithm in many cases
- But:
  - *k* may be unknown
  - Higher time complexity than greedy algorithm
  - Still some dependency to starting point selection
  - Euclidean Distance → spherical cluster model
  - Converges, but not necessary to the global minimum:

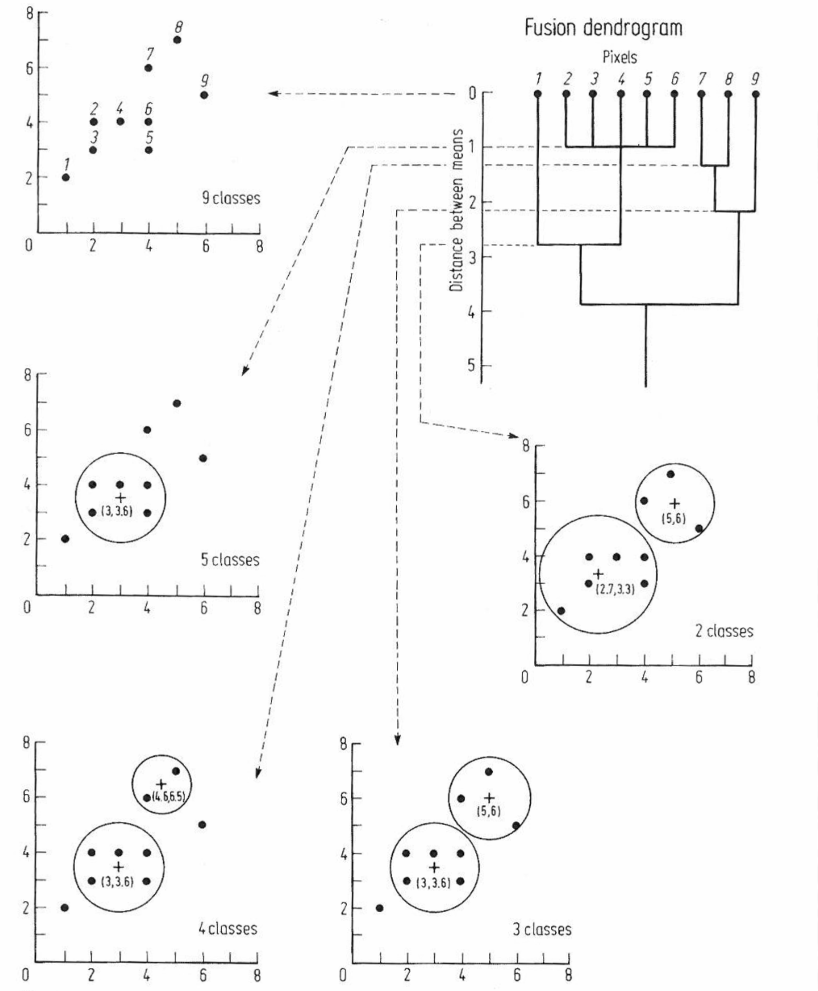


from Wikipedia, 2014

Agglomerative Hierarchical Clustering

- First: Start with an over segmentation
  - At Pixel level
  - Pre-Clustering
- Then: Systematically merge hierarchically until only one cluster remains
- Output: "History" of merging, typically displayed on a fusion dendrogram:
  - Long vertical sections: stable (equal) groups
  - Small vertical sections: unequal (unnatural) groups
- Two variations are used:
  - Agglomerative: Bottom-Up
  - Divisive: Top-Down

Example: Agglomerative Hierarchical Clustering



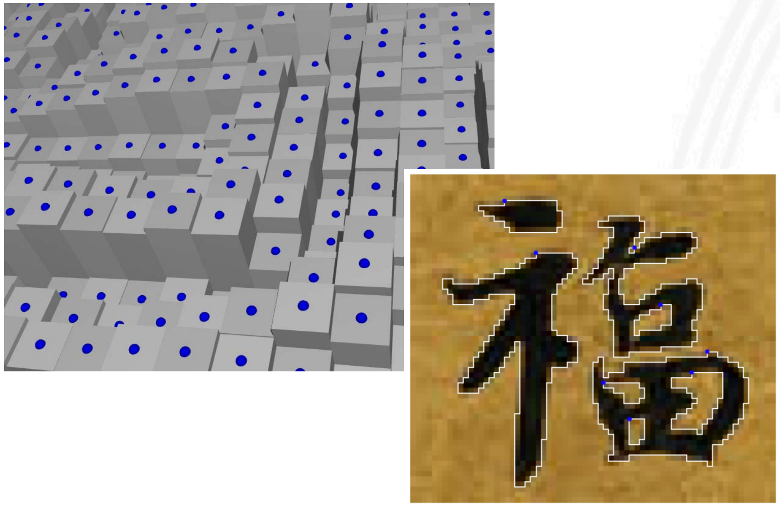
from Richards, 2006

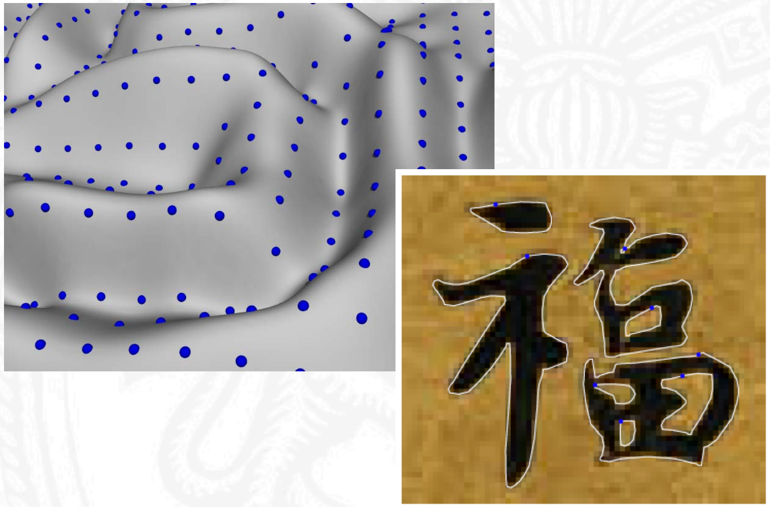
Comparison with Image Segmentation

- Presented clustering and classification techniques:
  - Defined in feature space
  - Often sparse results
- Image segmentation techniques:
  - Defined in(usually 2D) image space
  - Often complete subdivision as results
- Idea – related to Agglomerative Hierarchical Clustering:
  - Determine "superpixel" by image segmentation techniques
  - Iterative Merging of superpixel to form regions
    - Merging may be based of different region properties
  - Cancellation/Stopping criterion

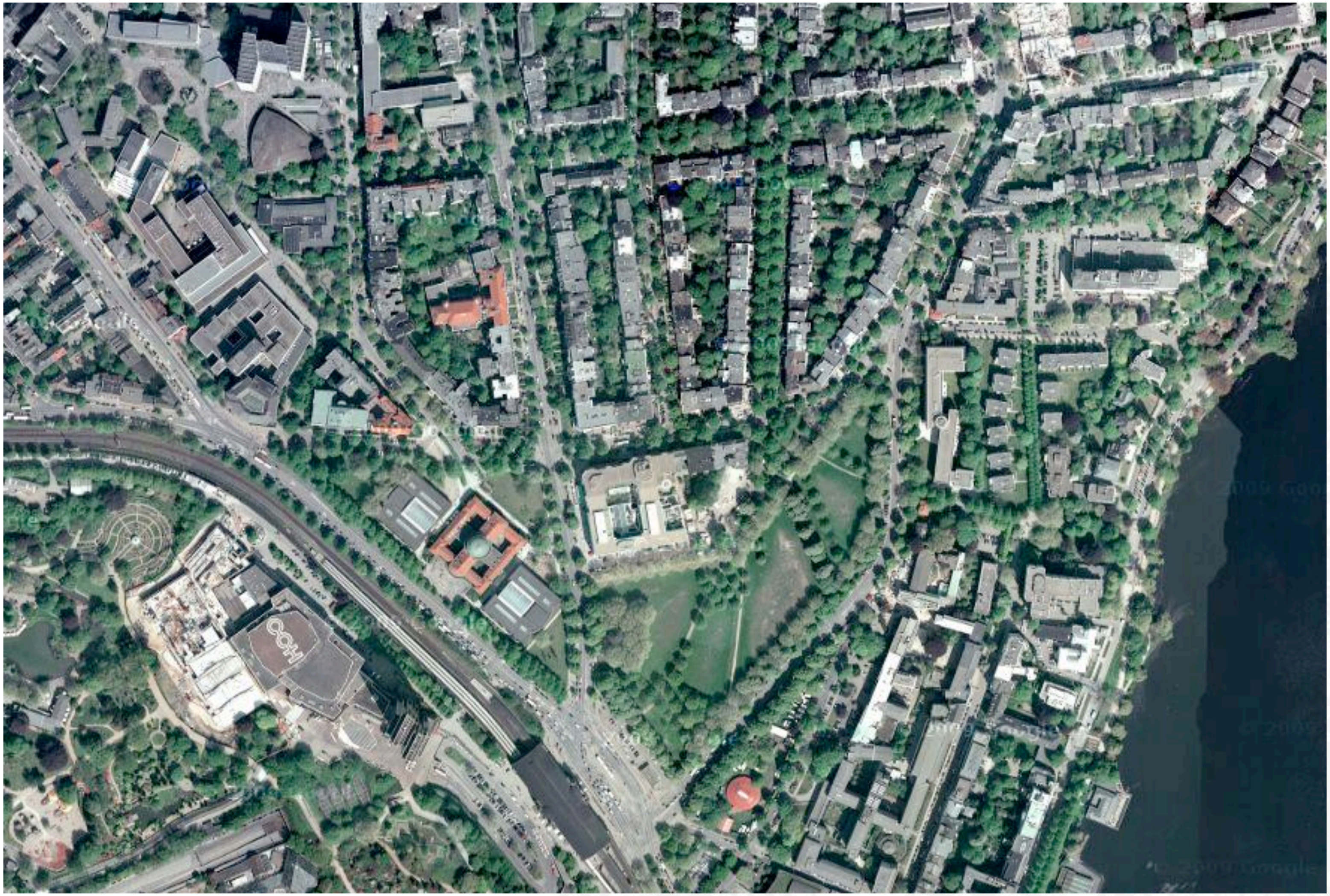
Example: The Watershed Transform

- Idea: Find “watersheds“ of the image gradient
- Implementation:
  - Dipping of the image function into water (Vincent & Soille)
  - (Fast-) Union-Find approach (Roerdink & Meijster)
  - Subpixel-based approach (Meine und Koethe)

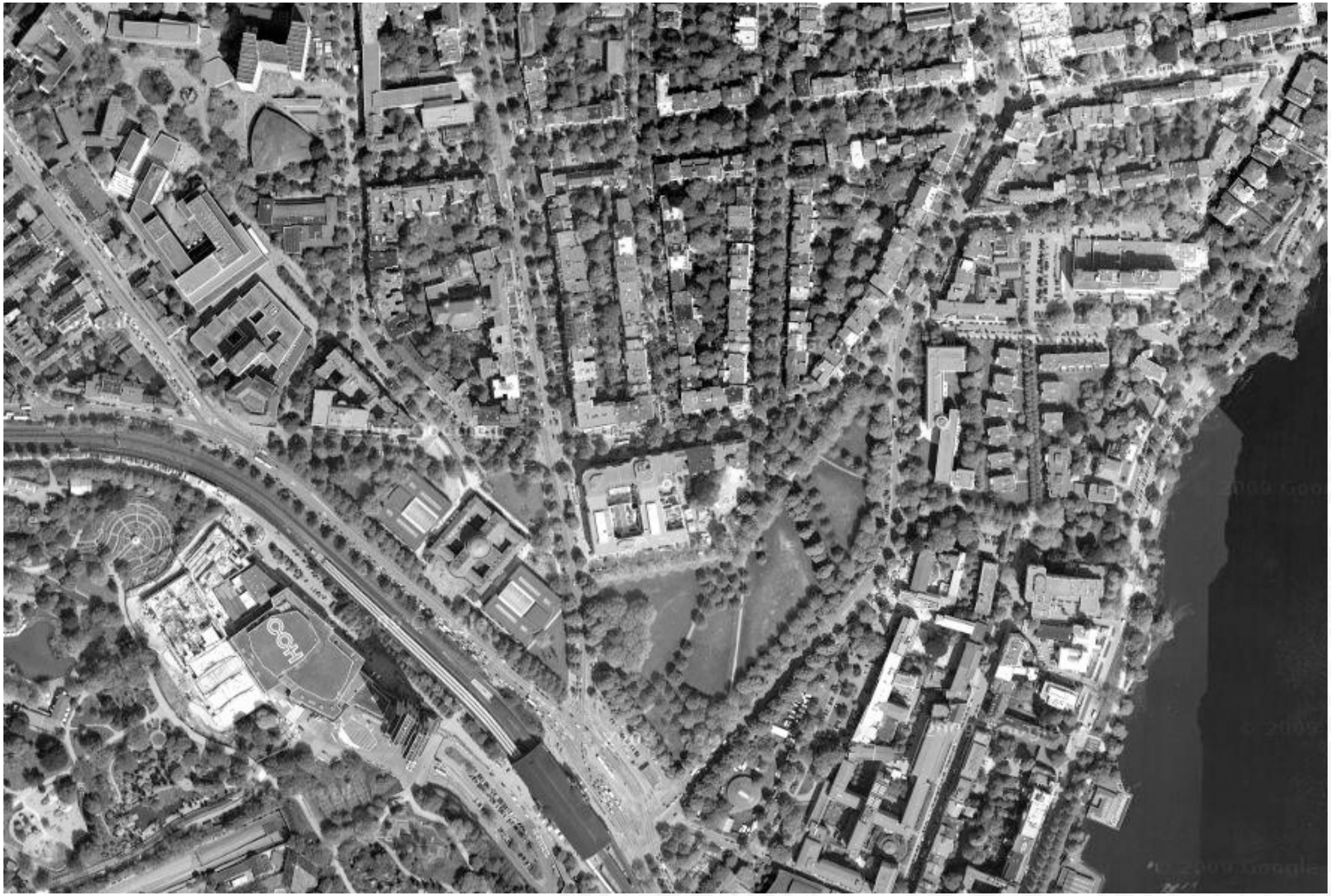




Example: Merging using Watersheds



Example: Merging using Watersheds



Example: Merging using Watersheds

Merging function: Difference of mean intensities