Image Classification and Object Recognition for Remote Sensing and Medical Image Analysis

Selim Aksoy

Department of Computer Engineering
Bilkent University
Bilkent, 06800, Ankara, Turkey
saksoy@cs.bilkent.edu.tr

October 23, 2013

Outline

- Part I: Automatic Detection of Geospatial Objects Using Multiple Hierarchical Segmentations
- Part II: Automatic Mapping of Agricultural Objects Using Very High Spatial Resolution Satellite Imagery
- Part III: Detection of Compound Structures in Very High Spatial Resolution Images
- Part IV: Segmentation and Classification of Cervical Cell Images

Bilkent University

- Founded in 1984 as the first private university in Turkey
- 1,200-acre campus located in Ankara
- 13,000 students
- 1,000 faculty members from 40 different countries

Bilkent University

RETINA Vision and Learning Group

- 2 faculty members
- 10+ graduate students
- Research on
  - Computer vision
  - Pattern recognition
  - Machine learning
  - Data mining
- Funding sources:
  - Scientific and Technological Research Council of Turkey
  - State Planning Agency
  - European Commission

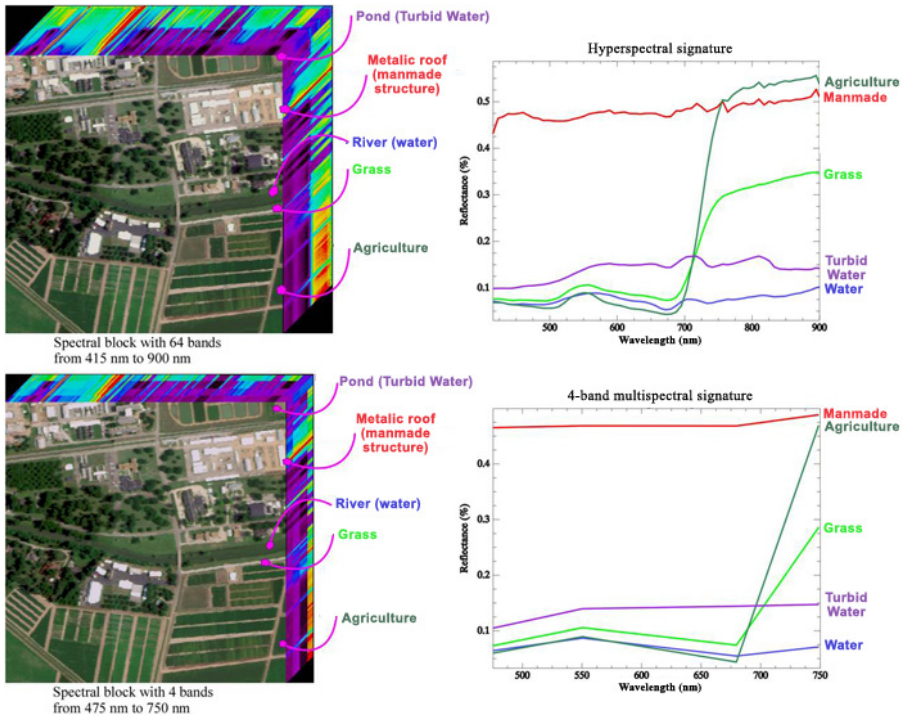
Part I

Automatic Detection of Geospatial Objects Using Multiple Hierarchical Segmentations

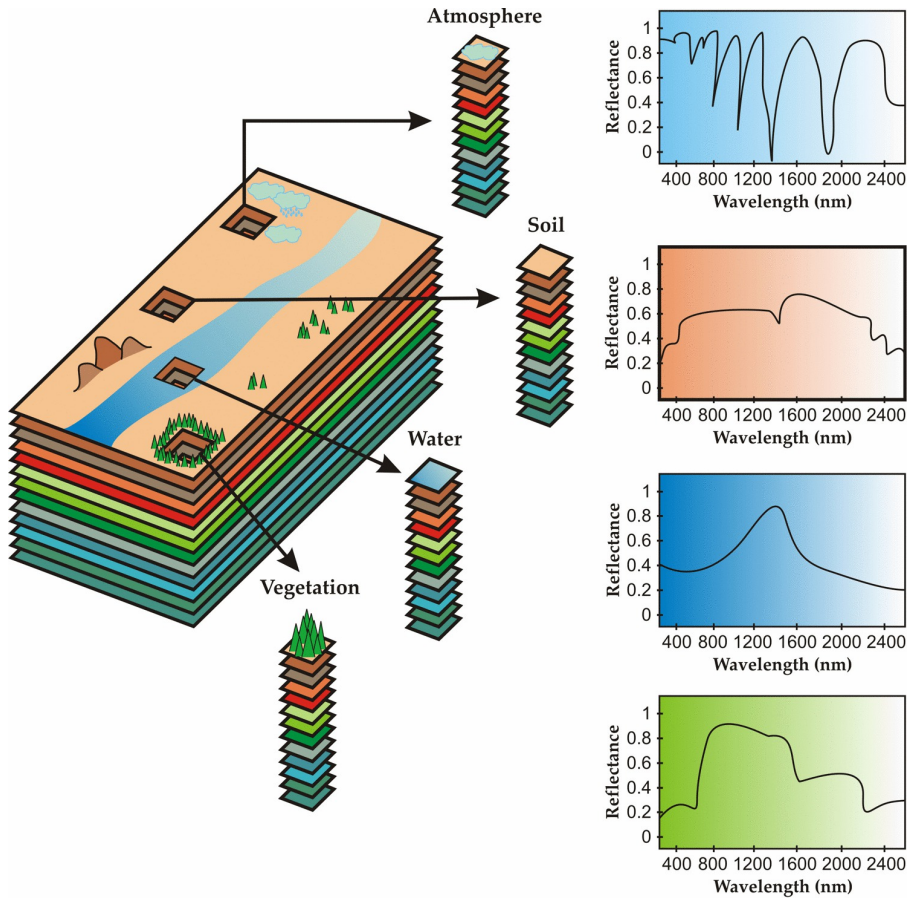
Pattern recognition in remote sensing

- Development of new pattern recognition techniques for the analysis of data collected from satellites and airborne sensors has been a popular research topic for several decades.
- Large volumes of data acquired from the last generation sensors require new advanced algorithms and techniques for automatic analysis.
- This data volume, together with new applications require new interdisciplinary work involving the application of novel pattern recognition techniques to unsolved problems in remote sensing image analysis.

Motivation



https://www.e-education.psu.edu/geog883kls/node/463



http://dx.doi.org/10.1109/JSTARS.2012.2194696

Figure 1: Multispectral and hyperspectral imaging.

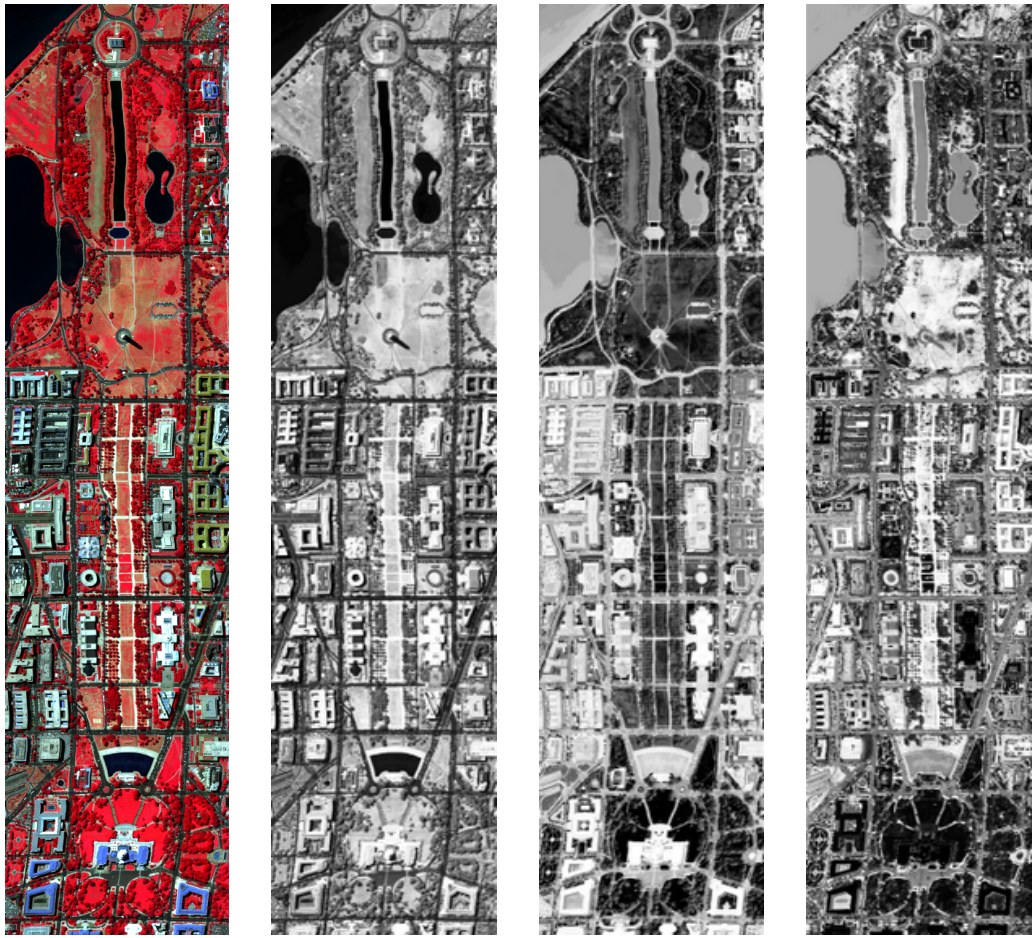
Image segmentation

- Most of the segmentation work in remote sensing literature are based on merging neighboring pixels according to user-defined thresholds on their spectral similarity.
- Our segmentation algorithm consists of the following steps:
  1. Principal components analysis
  2. Morphological profile extraction
  3. Hierarchical segment extraction
  4. Segment selection

Joint work with Gökhan Akçay, Bilkent University

Image segmentation

Principal components analysis



(a) Orig. (b) 1st PC (c) 2nd PC (d) 3rd PC

Figure 2: *DC Mall* image and its PCA bands.

Image segmentation

Principal components analysis

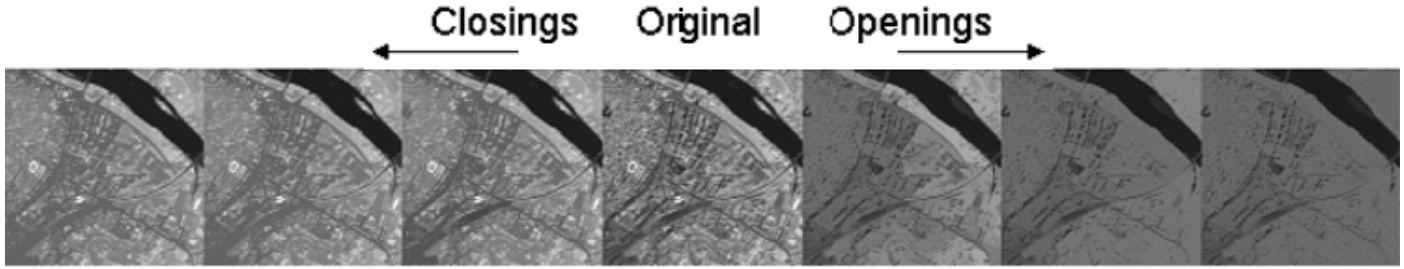
(a) Original (b) 1st PC (c) 2nd PC (d) 3rd PC

Figure 3: *Pavia Centre* image and its PCA bands.

Image segmentation

Morphological profiles

- Opening by reconstruction isolates structures that are brighter than their surroundings.
- Closing by reconstruction isolates structures that are darker than their surroundings.
- These operations are applied using increasing structuring element (SE) sizes to generate morphological profiles.



- Structuring element: disk
- Variables:
  - Number of openings/closings
  - Radius increment (step size)

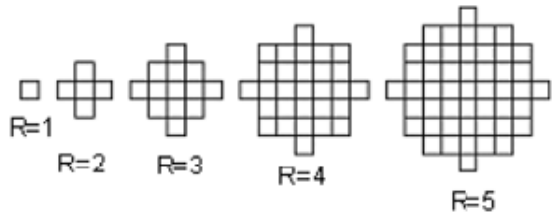
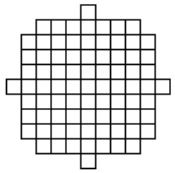


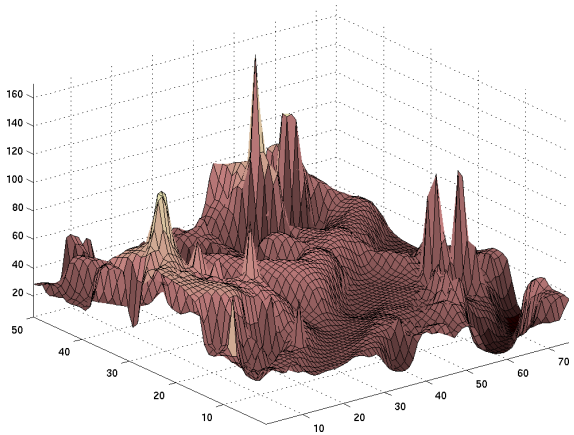
Image segmentation

Morphological profiles

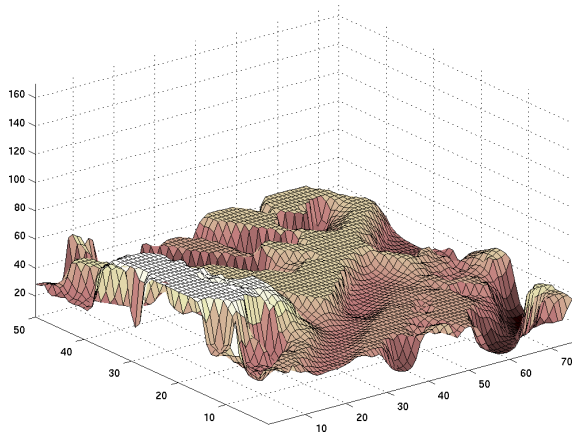
(a) Grayscale image



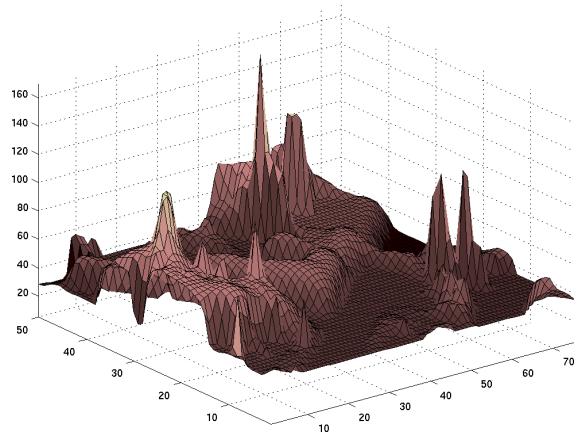
(b) SE



(c) 3D representation



(d) Opening by rec.

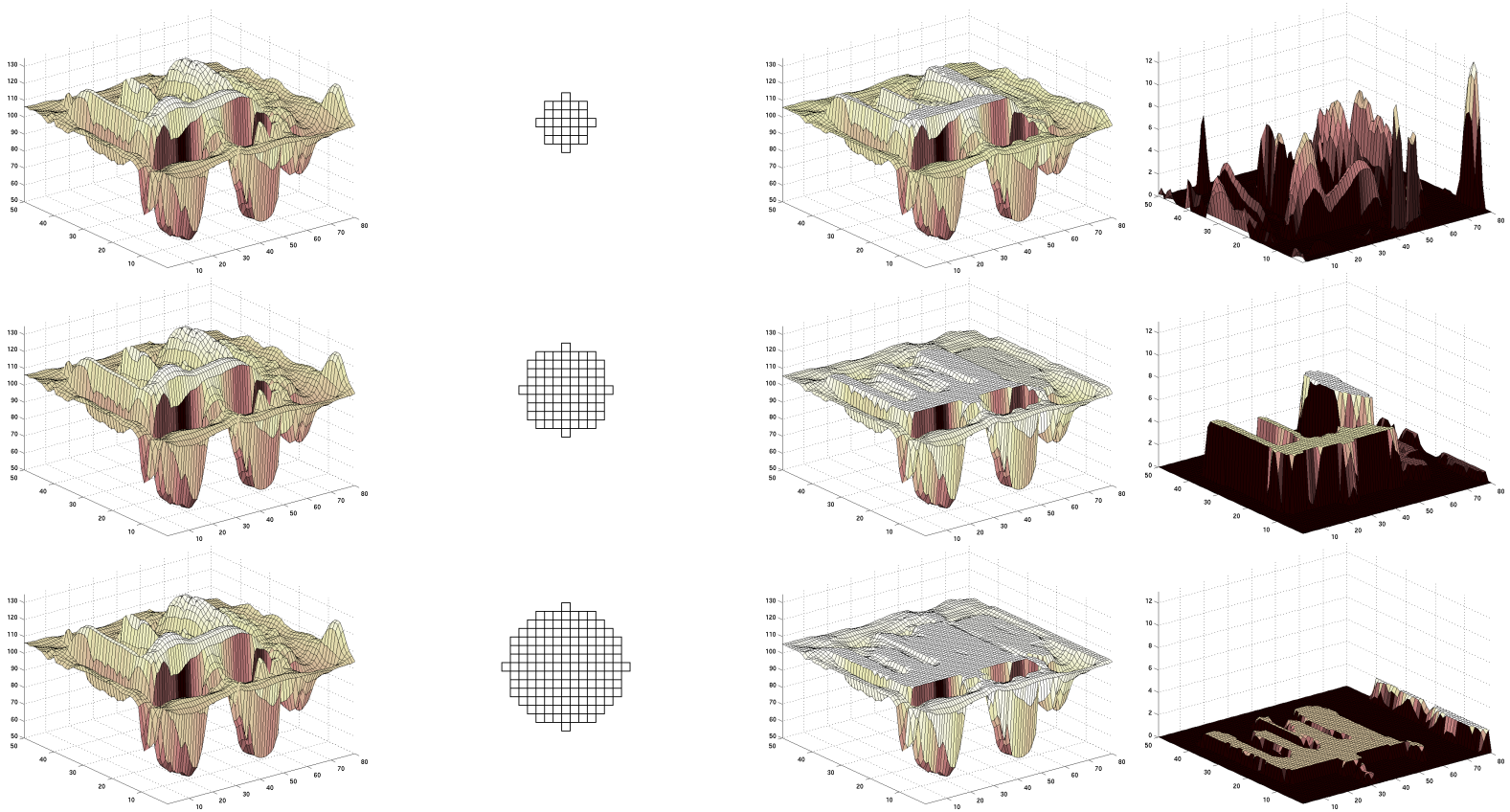


(e) Closing by rec.

Figure 4: Opening and closing by reconstruction example.

Image segmentation

Morphological profiles



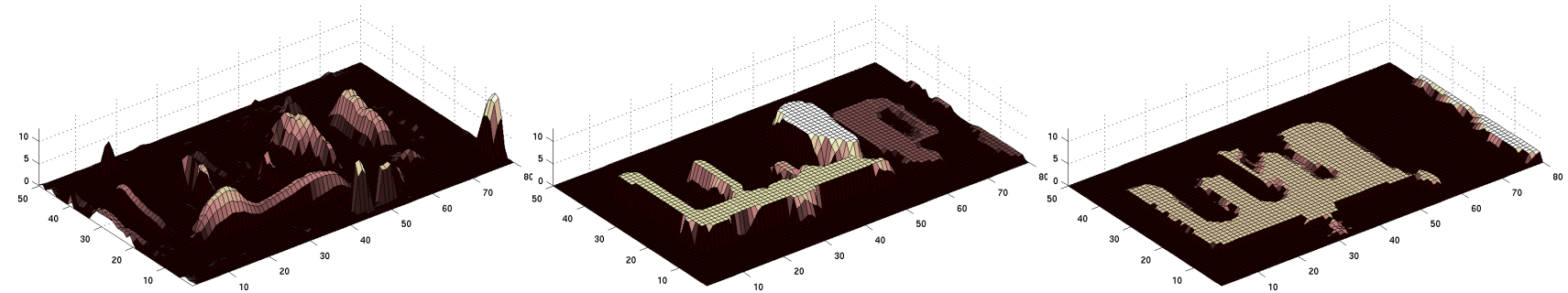
(a) Grayscale (b) SE (c) Opening MP (d) Derivative MP

Figure 5: Example morphological profile (MP) and its derivative (DMP).

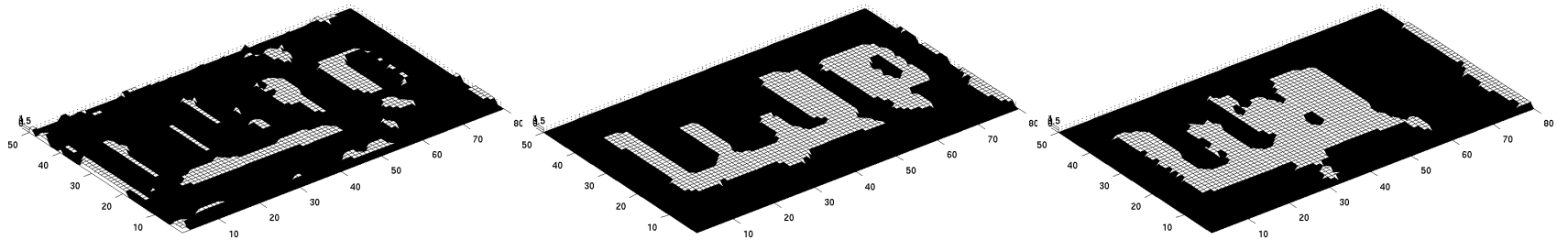
Image segmentation

Morphological profiles

- Our idea: Each neighboring group of pixels with a positive DMP is a candidate segment for the final segmentation.



(a) Opening DMP



(b) Thresholding at DMP > 0

Figure 6: Example connected components with DMP > 0.

Image segmentation overview

1. Apply o/c by reconstruction using SEs in increasing sizes.
2. Apply connected component analysis to the thresholded DMP at each scale.
   - Each connected component is a candidate meaningful segment.
   - Connected components are contained within each other in a hierarchy.
   - Using these candidate segments, construct a hierarchical tree.
3. Search for the most meaningful connected components in the tree.

Image segmentation

Hierarchical segment extraction

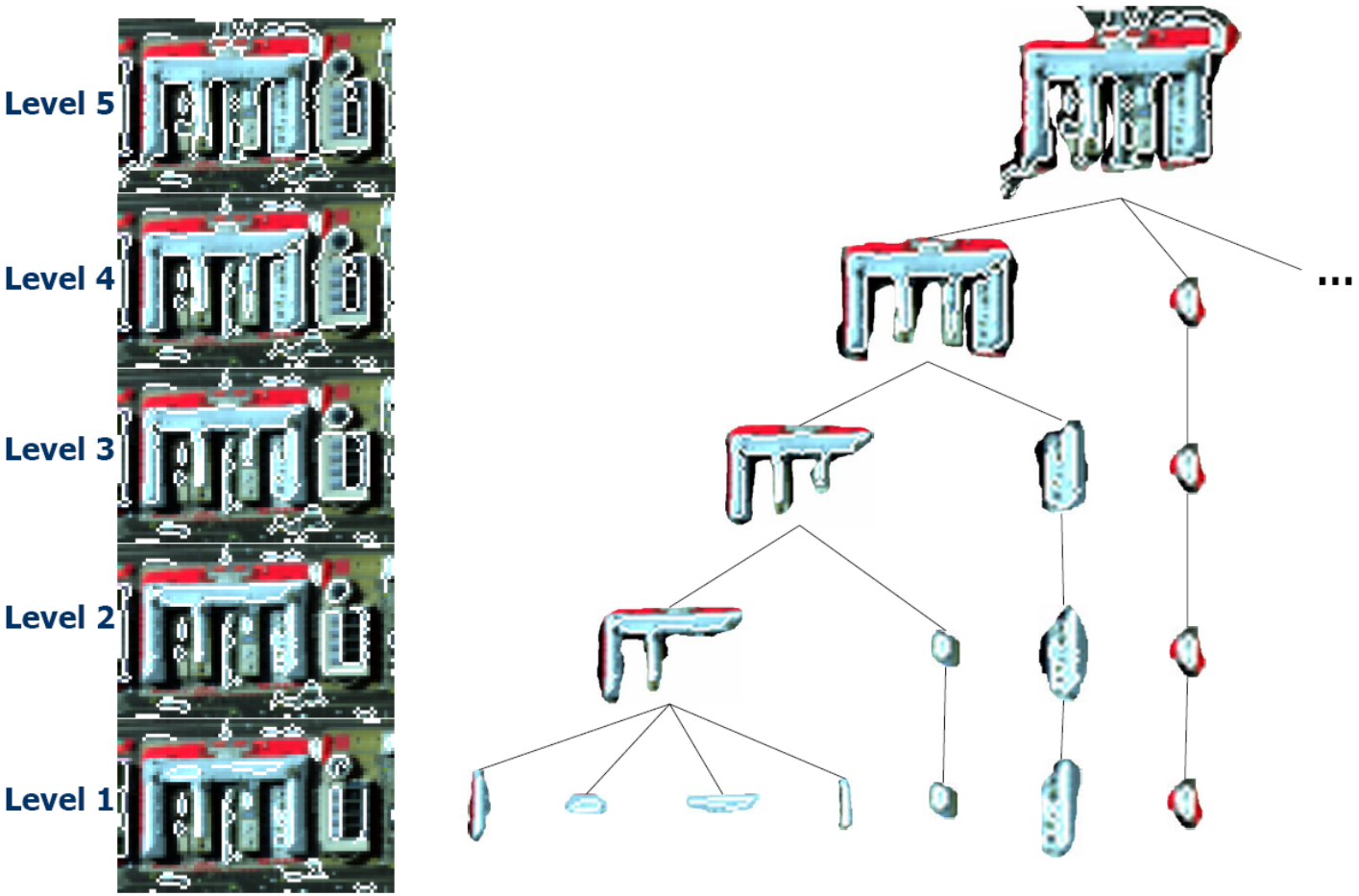


Figure 7: An example tree where each node is a candidate segment.

Image segmentation

Segment selection

- A segment must be as homogeneous as possible.
- Homogeneity (D): the difference between the standard deviation of the spectral information of a node and its parent.
- We expect a significant increase in the standard deviation when a structure merges with another.
- However, only the homogeneity factor will favor small structures.

Image segmentation

Segment selection

- As a competing factor, we want to select segments that are as large as possible: number of pixels in the segment (C).
- The goodness measure M for a node n is defined as

$$M(n) = D(n, parent(n)) \times C(n).$$

- The components that optimize this measure are selected using a two-pass algorithm on the tree.

Image segmentation

Segment selection

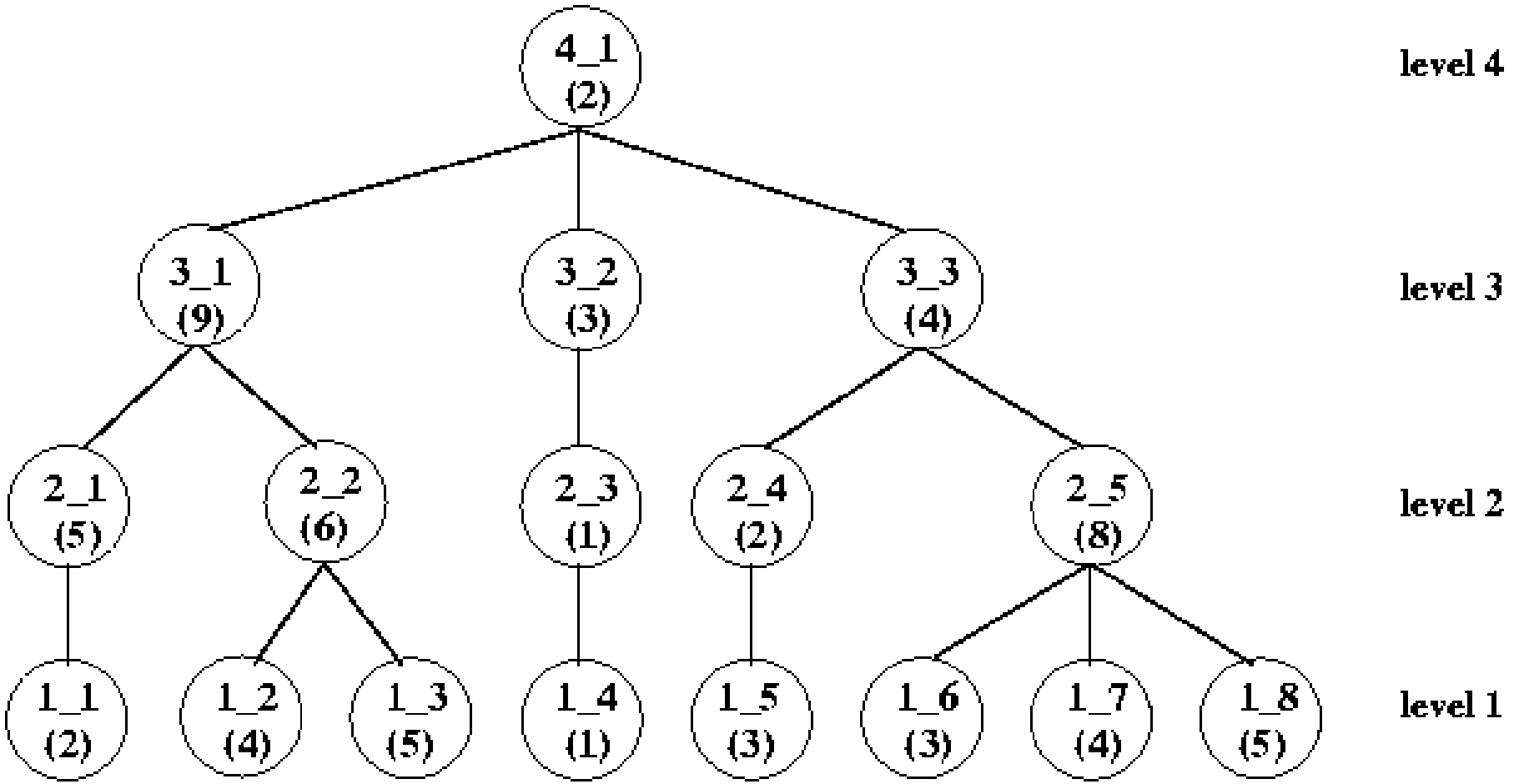


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Segment selection

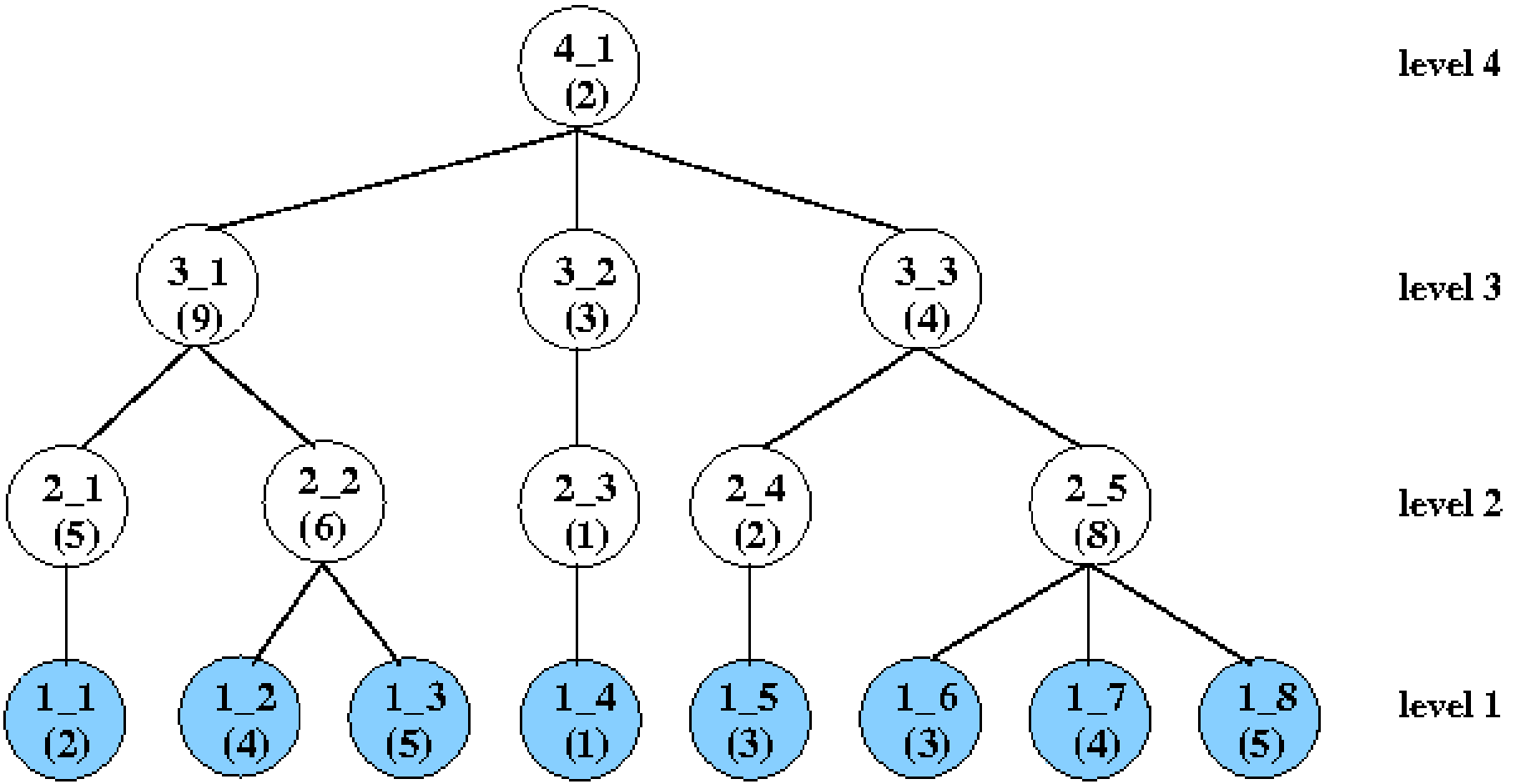


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Segment selection

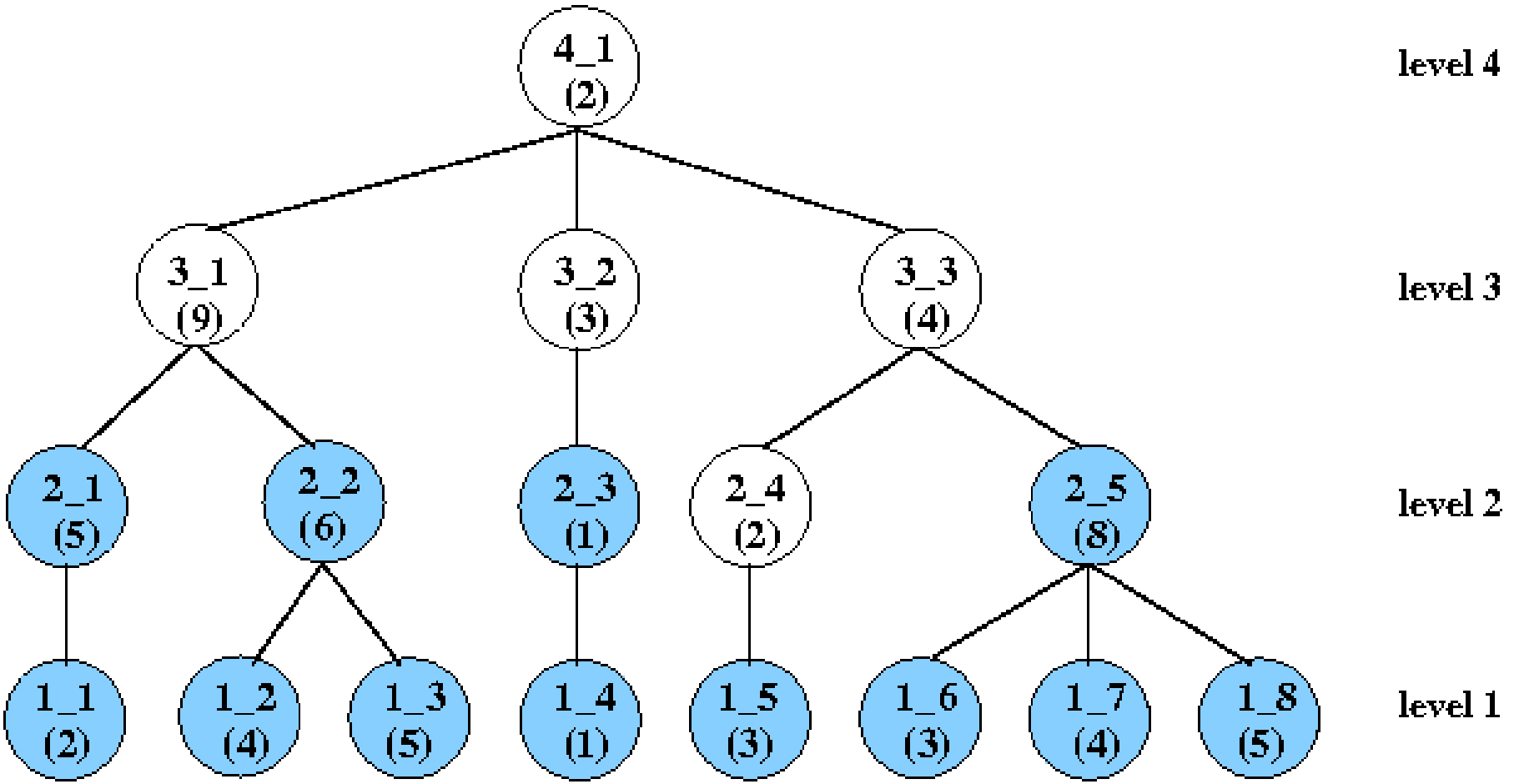


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Segment selection

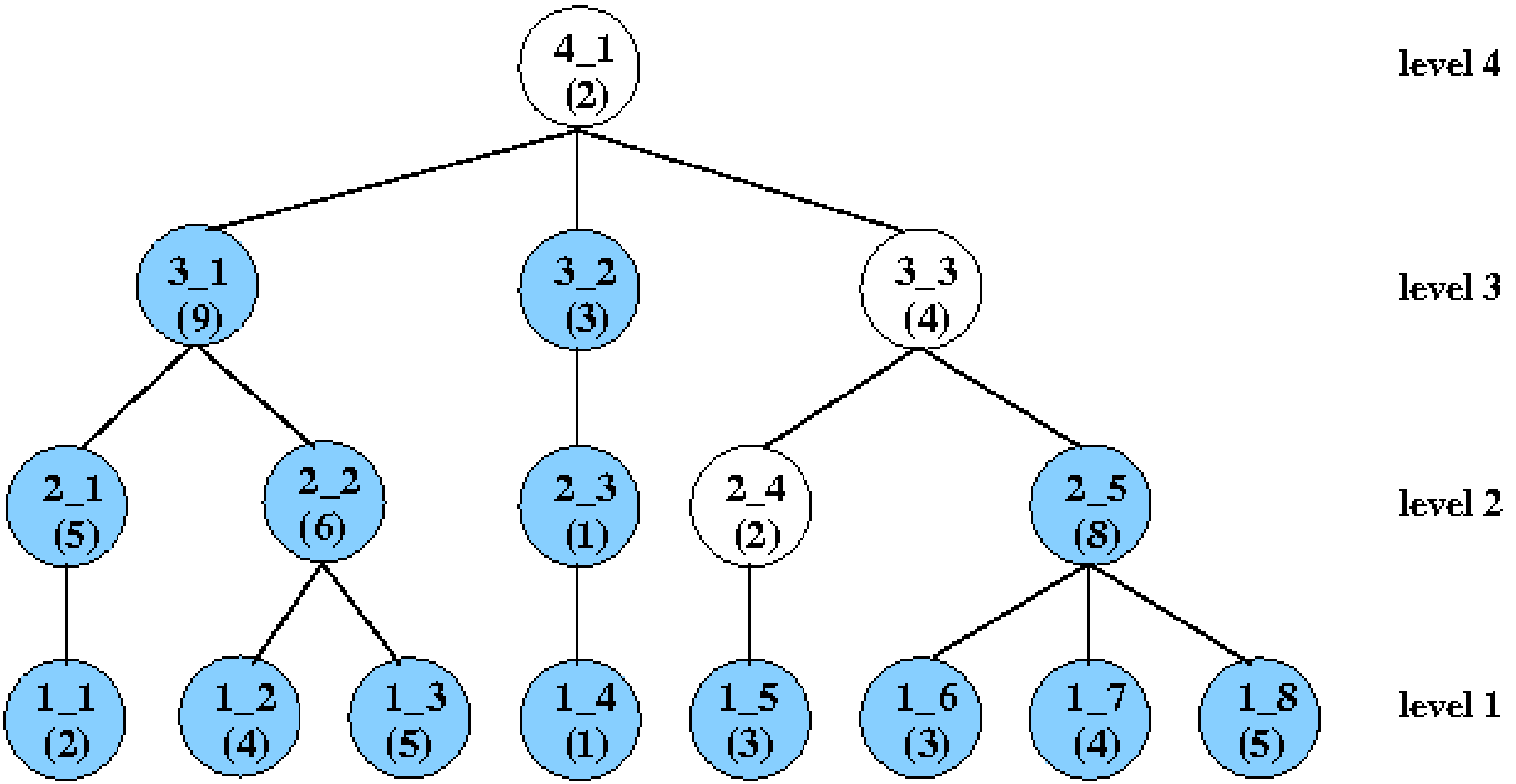


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Segment selection

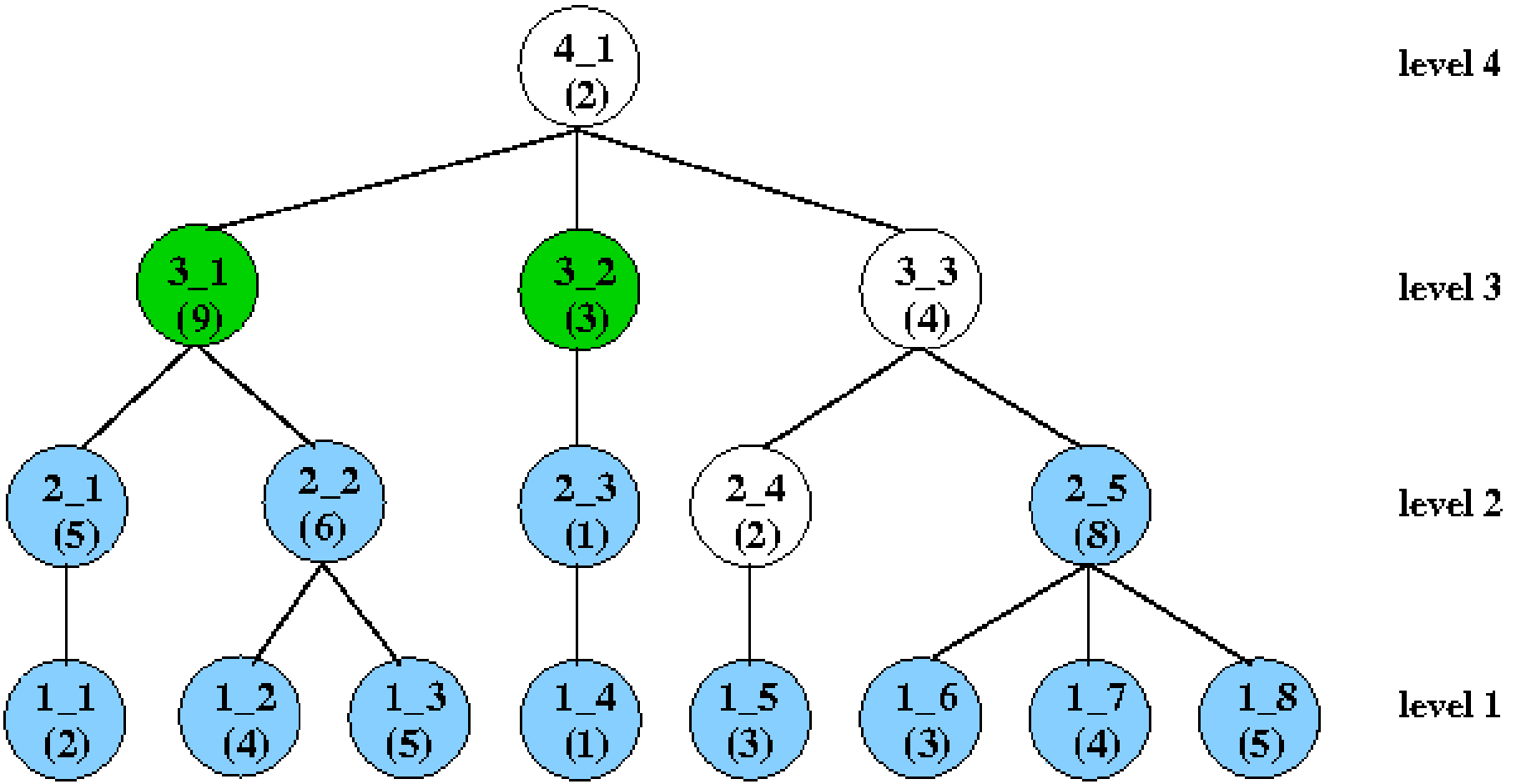


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Segment selection

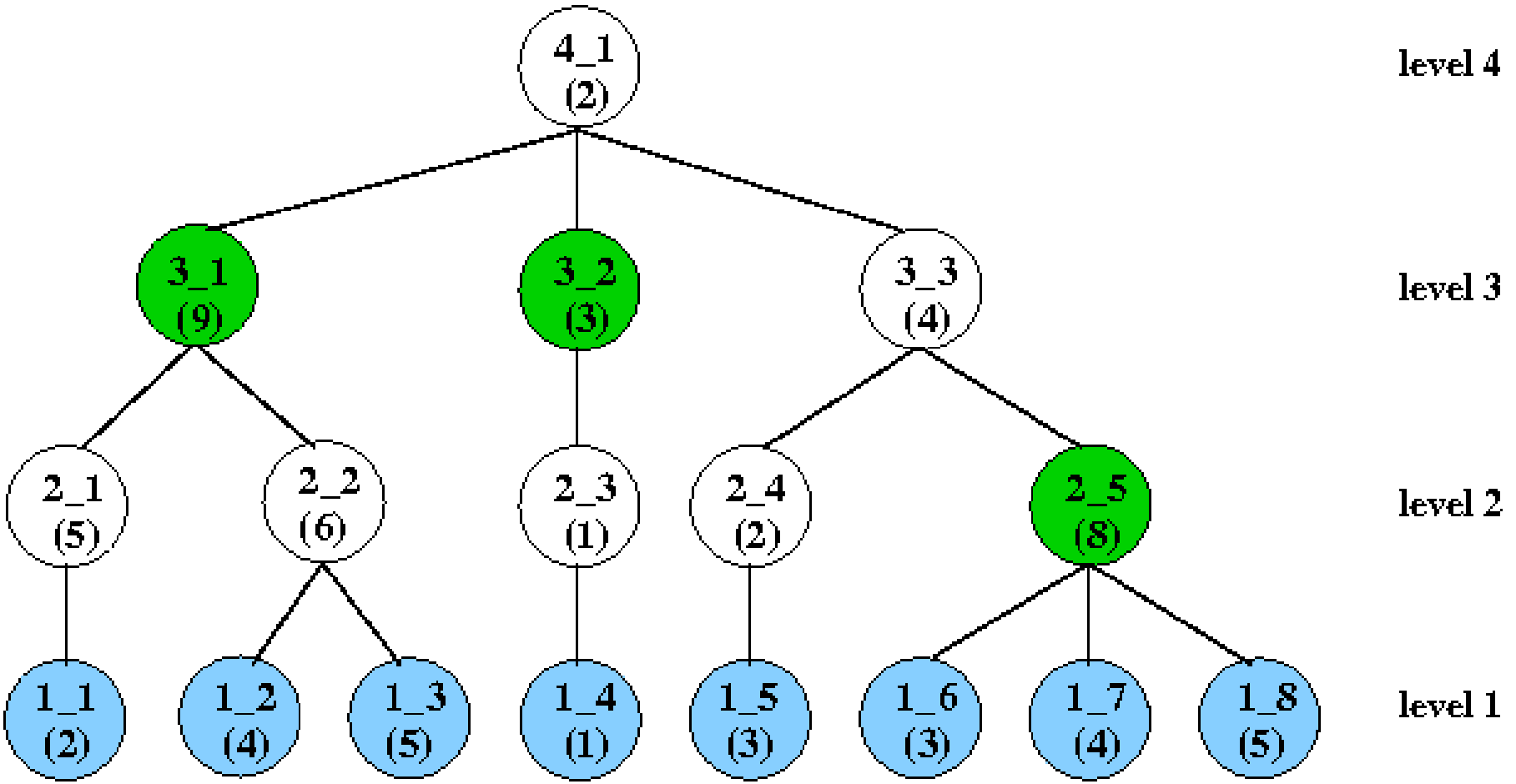


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Segment selection

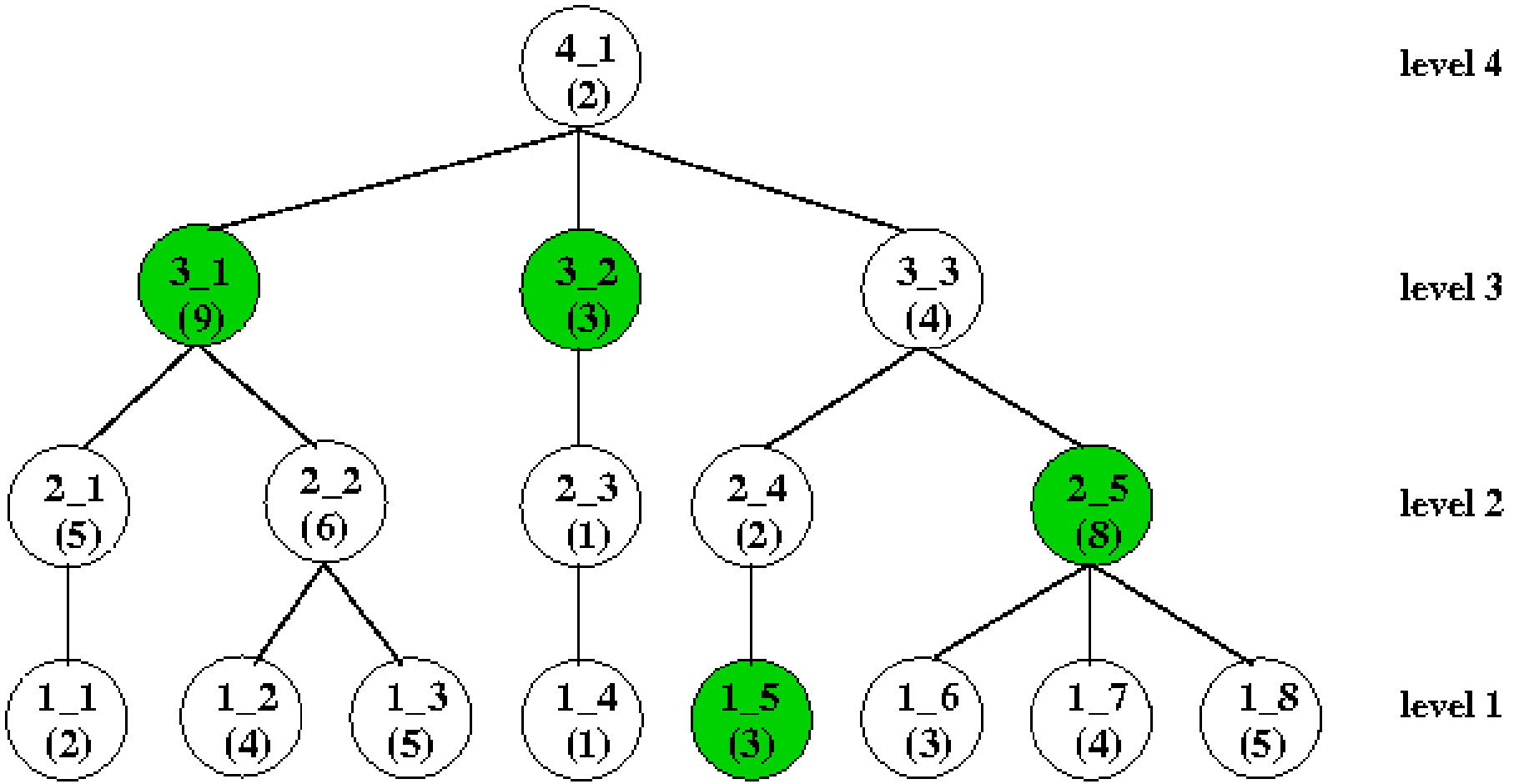


Figure 8: An example run of the two-pass algorithm.

Image segmentation

Examples

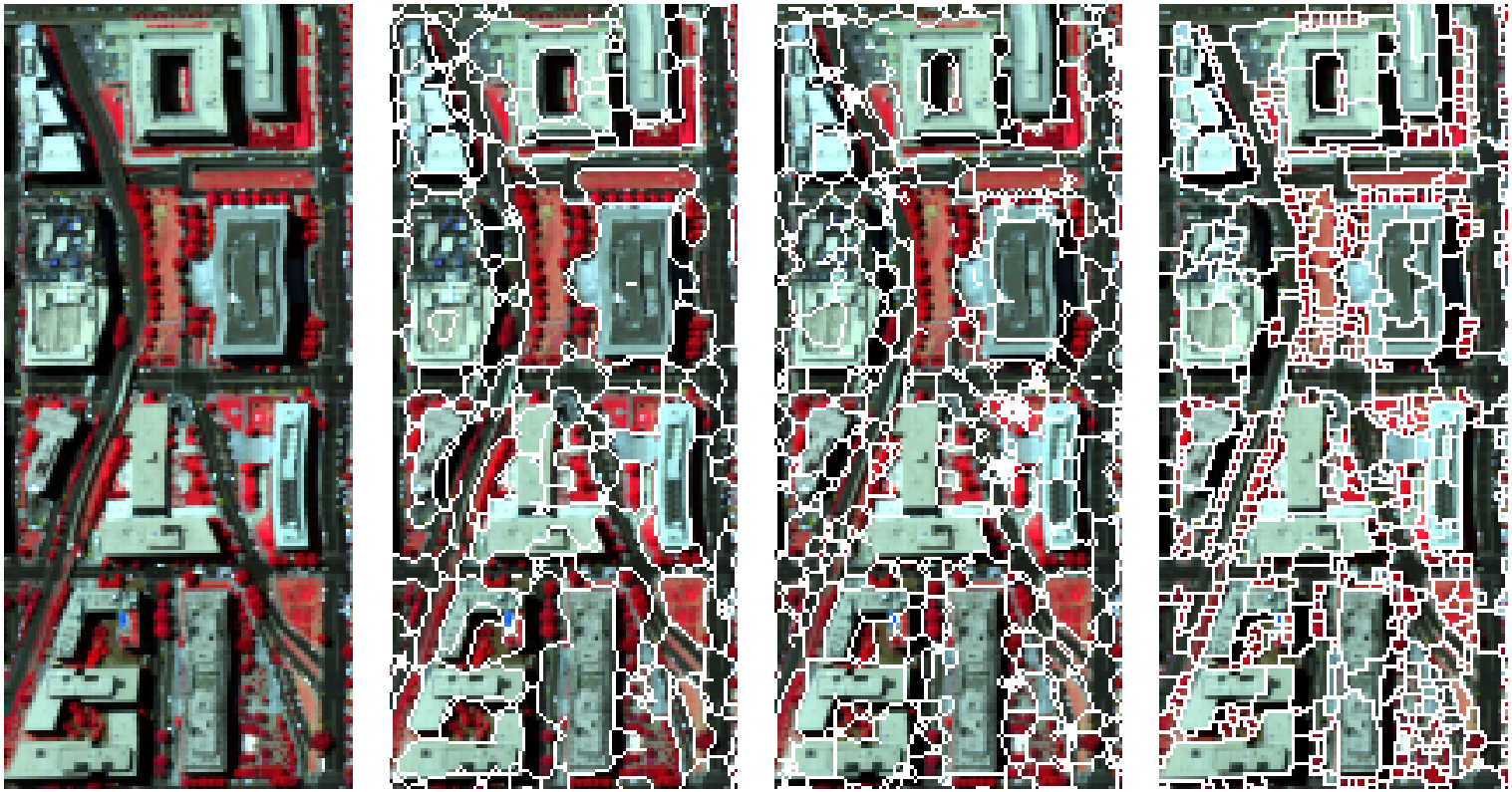


Figure 9: Example segmentation results for the *DC Mall* data set: false color, result of the proposed approach, result of Pesaresi-Benediktsson, result of watershed segmentation.

Image segmentation

Examples

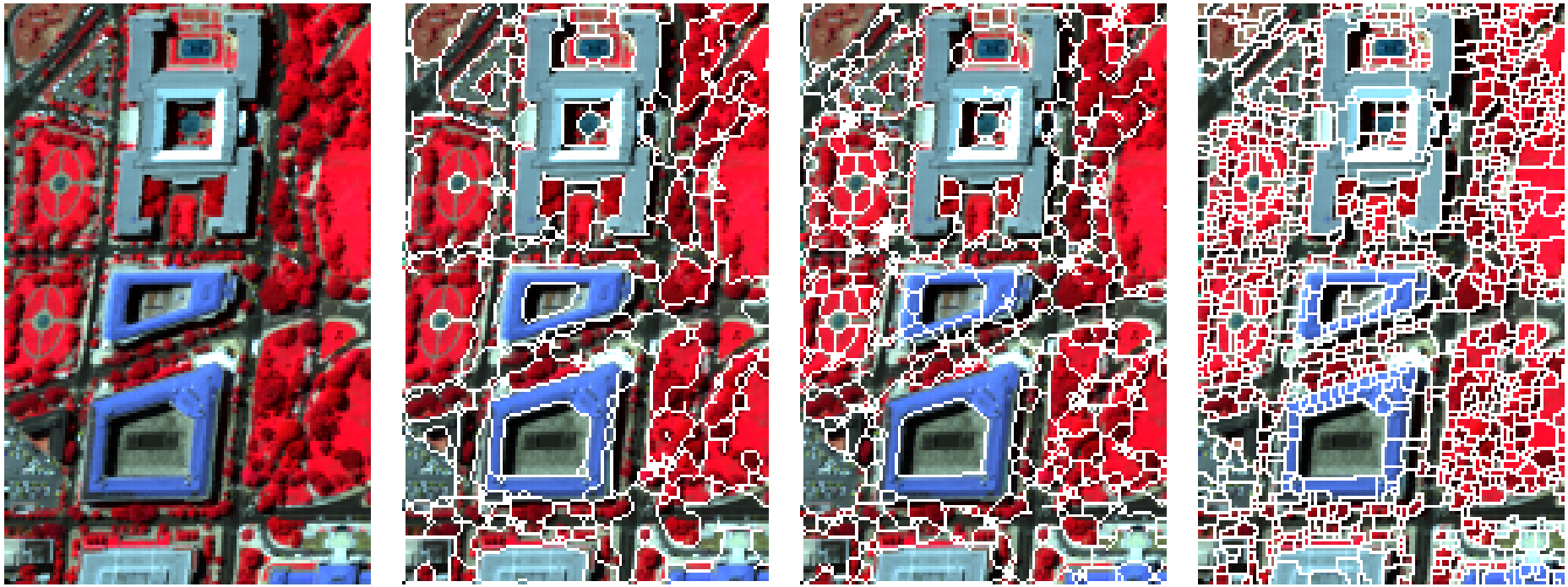


Figure 10: Example segmentation results for the *DC Mall* data set: false color, result of the proposed approach, result of Pesaresi-Benediktsson, result of watershed segmentation.

Image segmentation

Examples

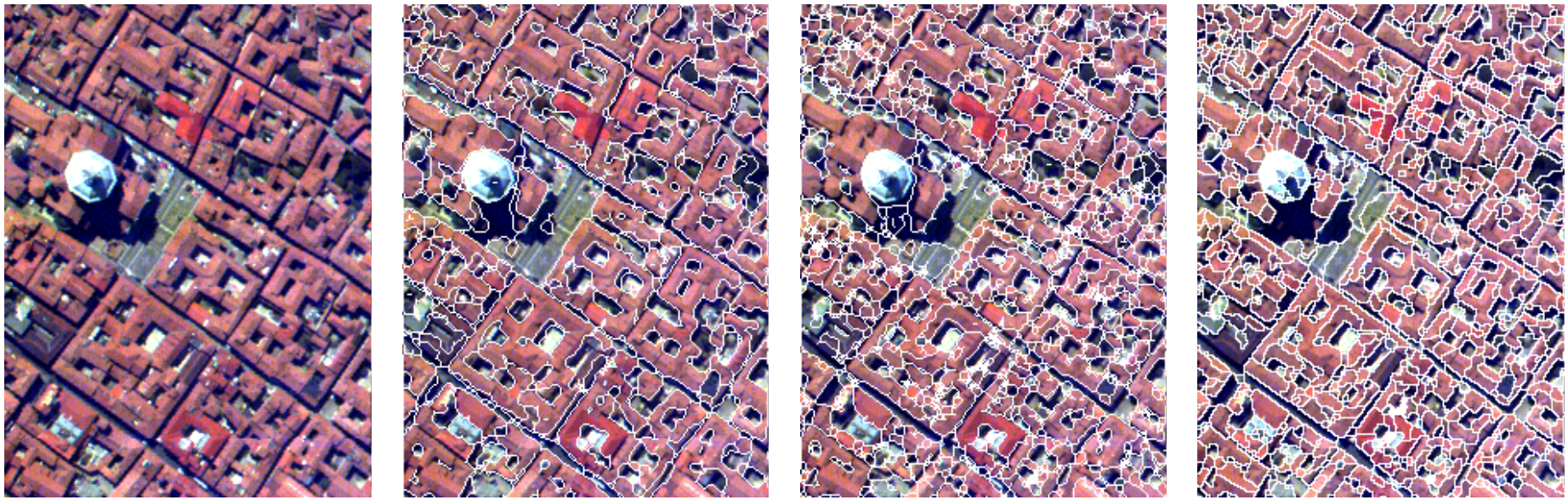


Figure 11: Example segmentation results for the *Pavia Centre* data set: false color, result of the proposed approach, result of Pesaresi-Benediktsson, result of watershed segmentation.

Image segmentation

Examples

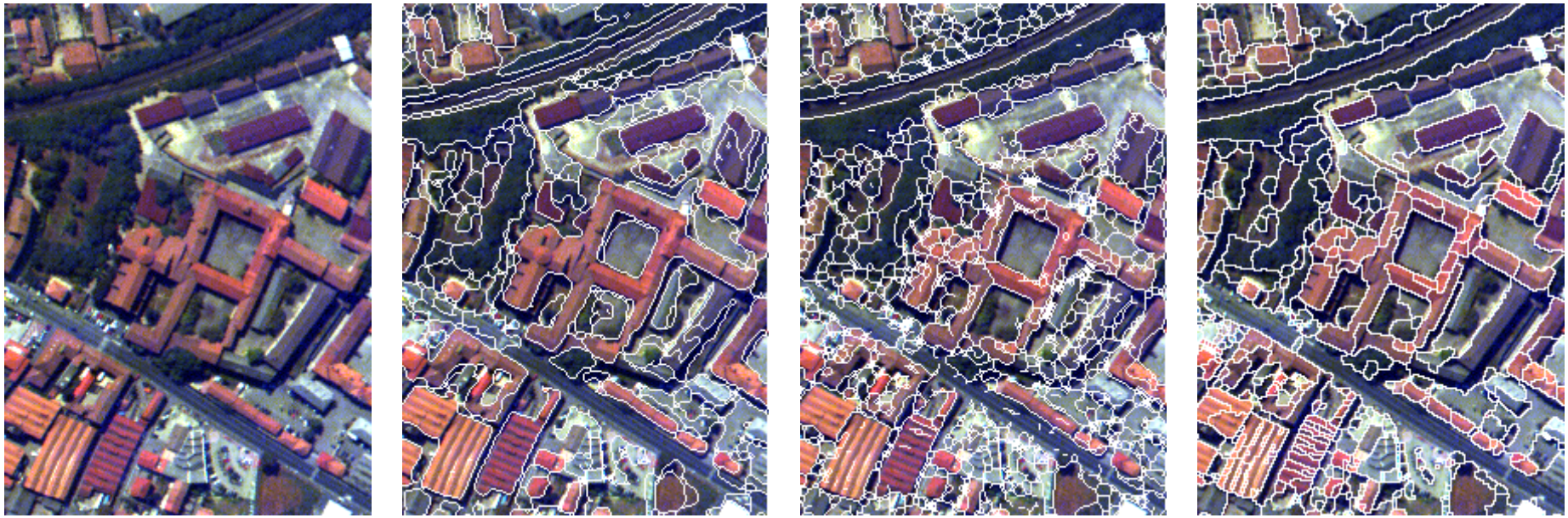


Figure 12: Example segmentation results for the *Pavia Centre* data set: false color, result of the proposed approach, result of Pesaresi-Benediktsson, result of watershed segmentation.

Image segmentation

Examples

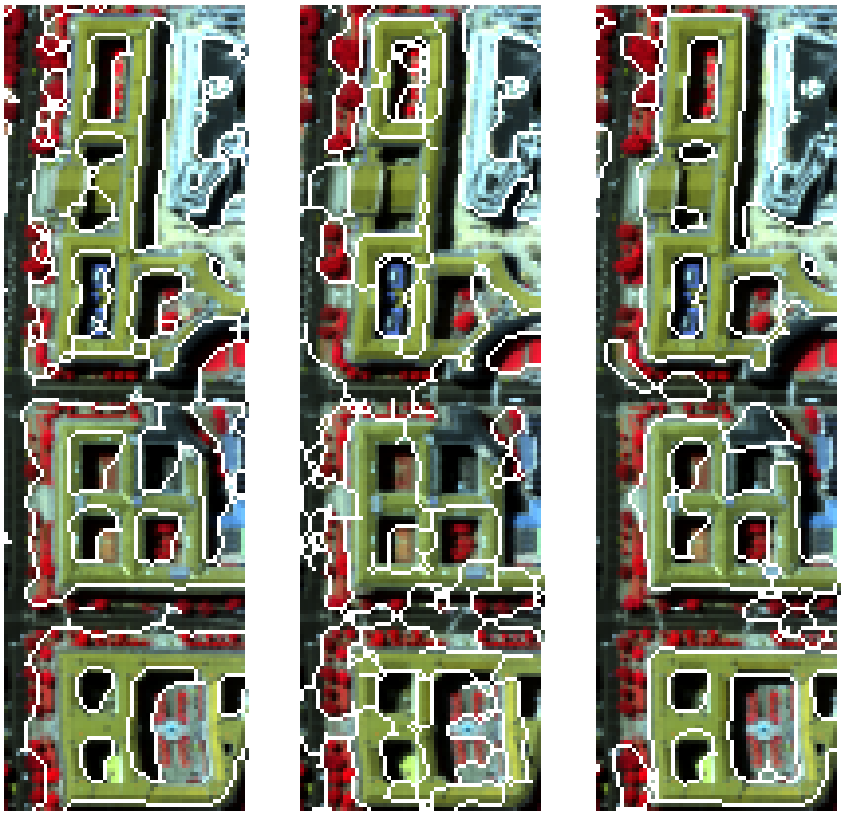


Figure 13: Example segmentation results for the *DC Mall* data set: first, second and third PCA bands.

Grouping segments for object detection

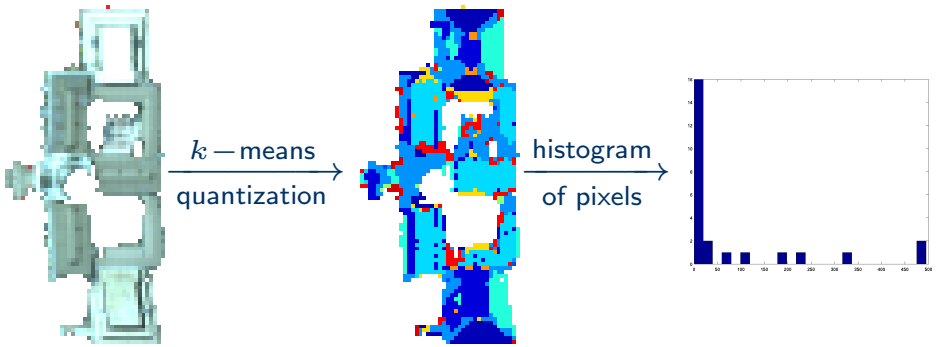
- Different structures appear more clearly in different principal components.
- Information from multiple PCA components must be combined for better overall detection.
- Assumption: for a particular structure (e.g., building),
  - the "good" segments (i.e., the ones containing a building) will all have similar features,
  - the "bad" segments (i.e., the ones containing multiple objects) will be described by a random mixture of features.

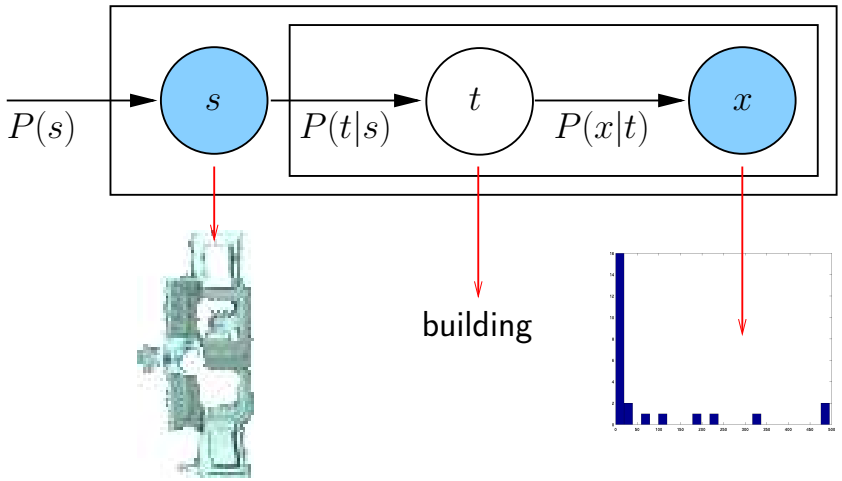
Grouping segments for object detection

- The selection process is formulated as a grouping problem within the space of a large number of candidate segments obtained from multiple segmentations.
- The grouping problem is solved using the probabilistic Latent Semantic Analysis (PLSA) algorithm.
- The resulting groups correspond to different types of objects in the image.
- The overall object detection algorithm is automatic.

Grouping segments for object detection

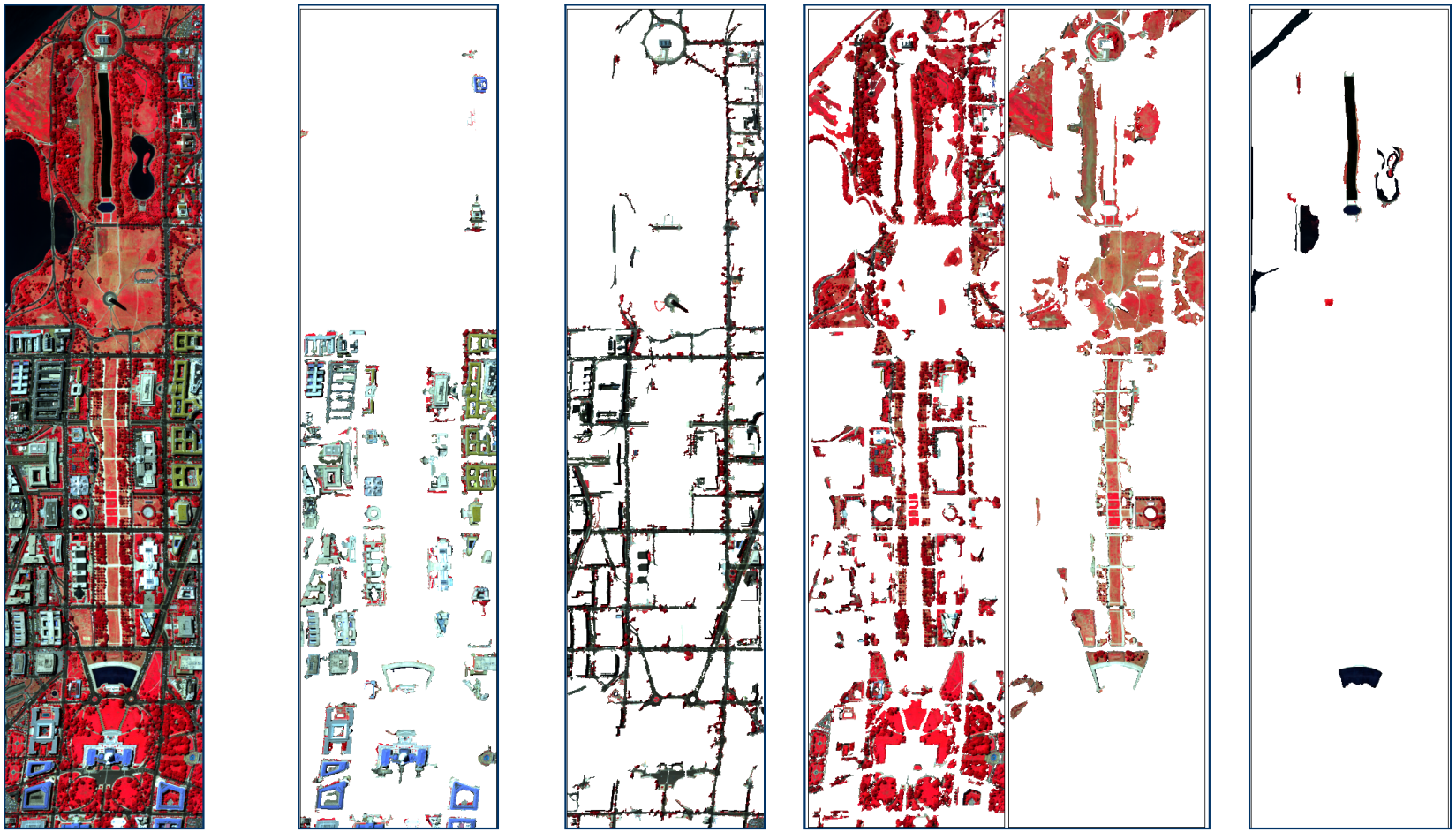
- Each segment is modeled using the statistical summary of its pixel content (e.g., quantized spectral values).





- The generative model for the joint probability of segments and their features is learned using the Expectation-Maximization (EM) algorithm.

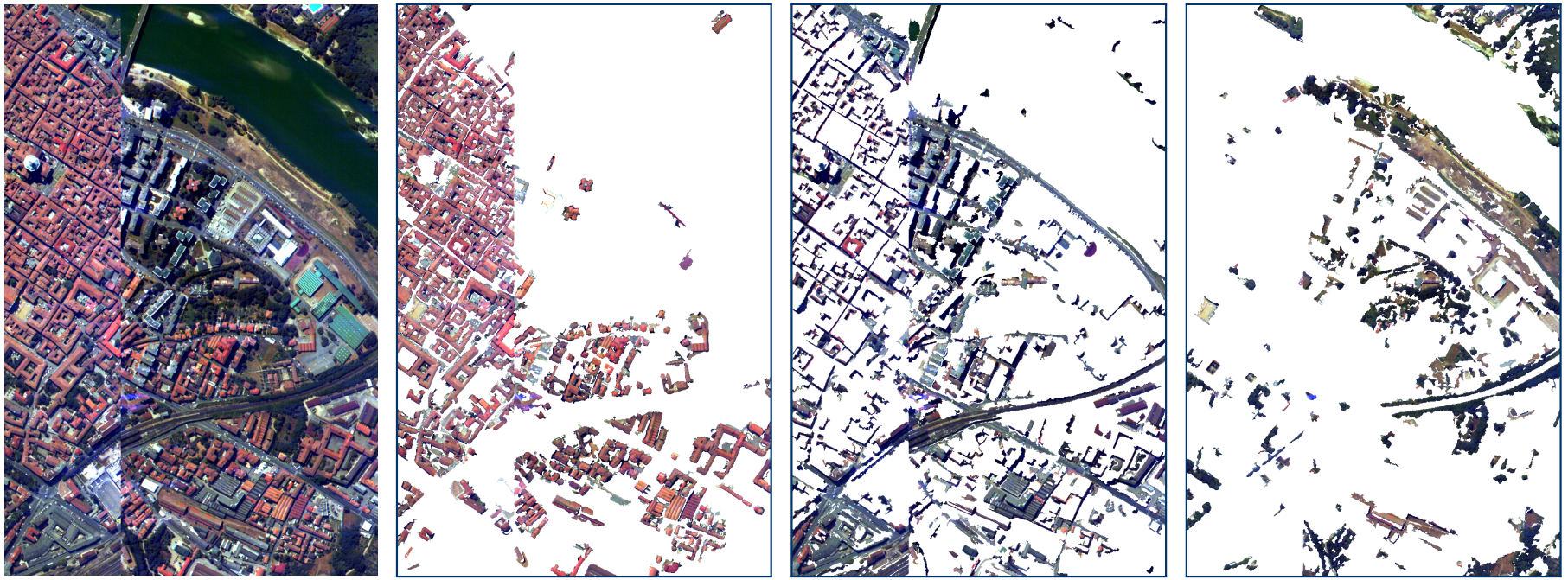
Experiments



(a) False color (b) Buildings (c) Roads (d) Vegetation (e) Water

Figure 14: Examples of object detection for the *DC Mall* data set.

Experiments

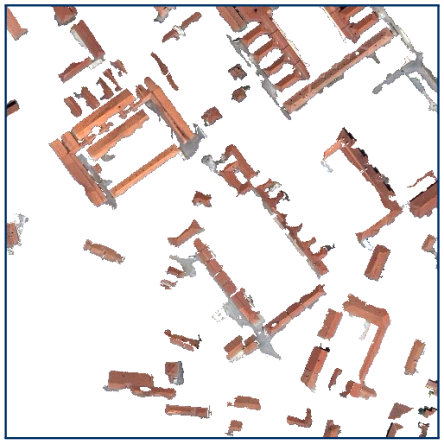


(a) False color (b) Buildings (c) Roads (d) Vegetation

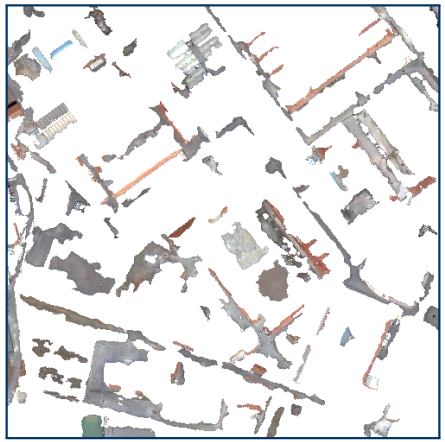
Figure 15: Examples of object detection for the *Pavia Centre* data set.

Experiments

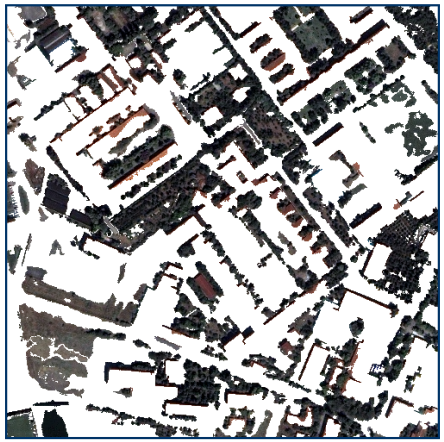
(a) RGB



(b) Buildings



(c) Roads



(d) Vegetation

Figure 16: Examples of object detection for the *Ankara* data set.

Part II

Automatic Mapping of Agricultural Objects Using Very High Spatial Resolution Satellite Imagery

Introduction

- Remote sensing has been a valuable tool for the planning, control, maintenance, and monitoring of agricultural sites.
- Generalizability and robustness of automatic techniques are particularly important when the analysis goes beyond local sites to cover a wide range of landscapes.

Introduction

- The goal of this study is to develop automatic methods for mapping of target landscape features in very high spatial resolution images.
- The target objects of interest in this talk are
  - *hedges* that are linear strips of woody vegetation, and
  - *orchards* that are composed of regular plantation of individual trees.

Joint work with Gökhan Akçay and Zeki Yalnız, Bilkent University

Hedge detection

Overview

(a) Baden-Württemberg, Germany (b) Decin, Czech Republic

Figure 17: Example Quickbird images containing hedges.

Hedge detection

Overview

(a) Paphos, South Cyprus

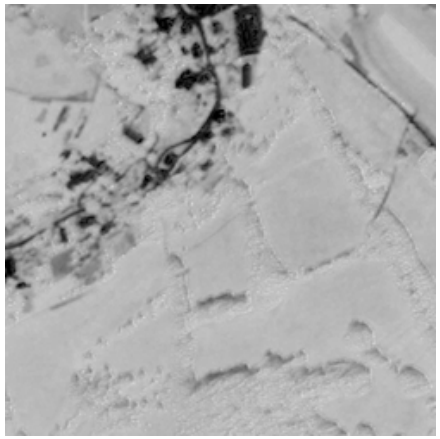
(b) Paphos, South Cyprus

Figure 18: Example Quickbird images containing hedges.

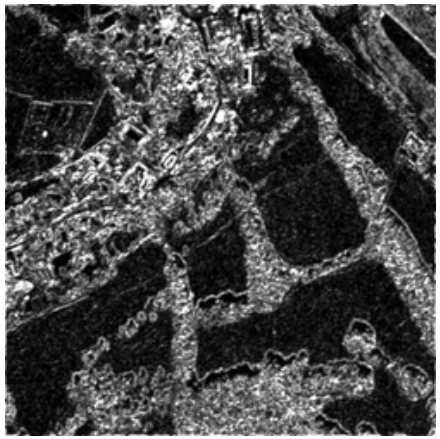
Hedge detection

Feature extraction

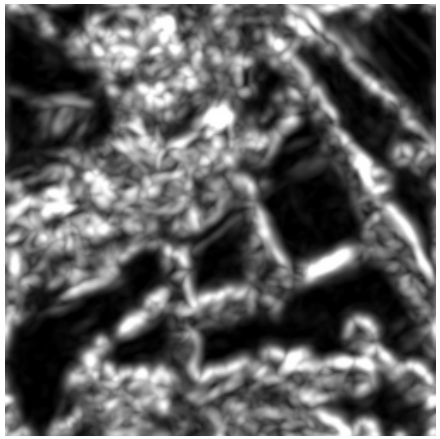
(a) Multispectral



(b) NDVI

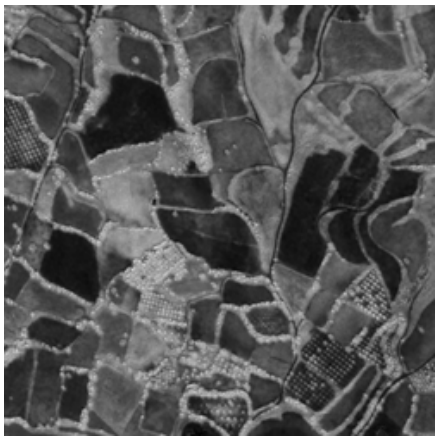


(c) Gabor - scale 1

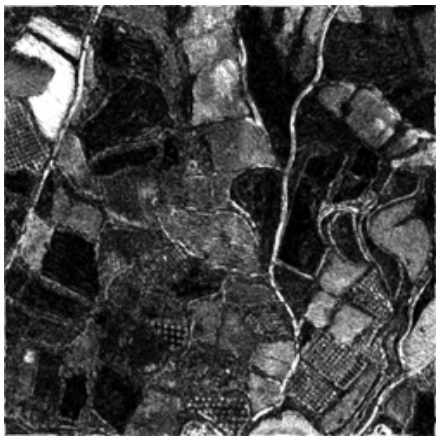


(d) Gabor - scale 6

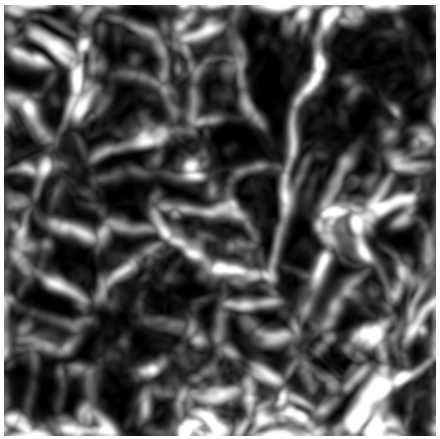
(e) Multispectral



(f) NDVI



(g) Gabor - scale 1



(h) Gabor - scale 6

Figure 19: Features for example images.

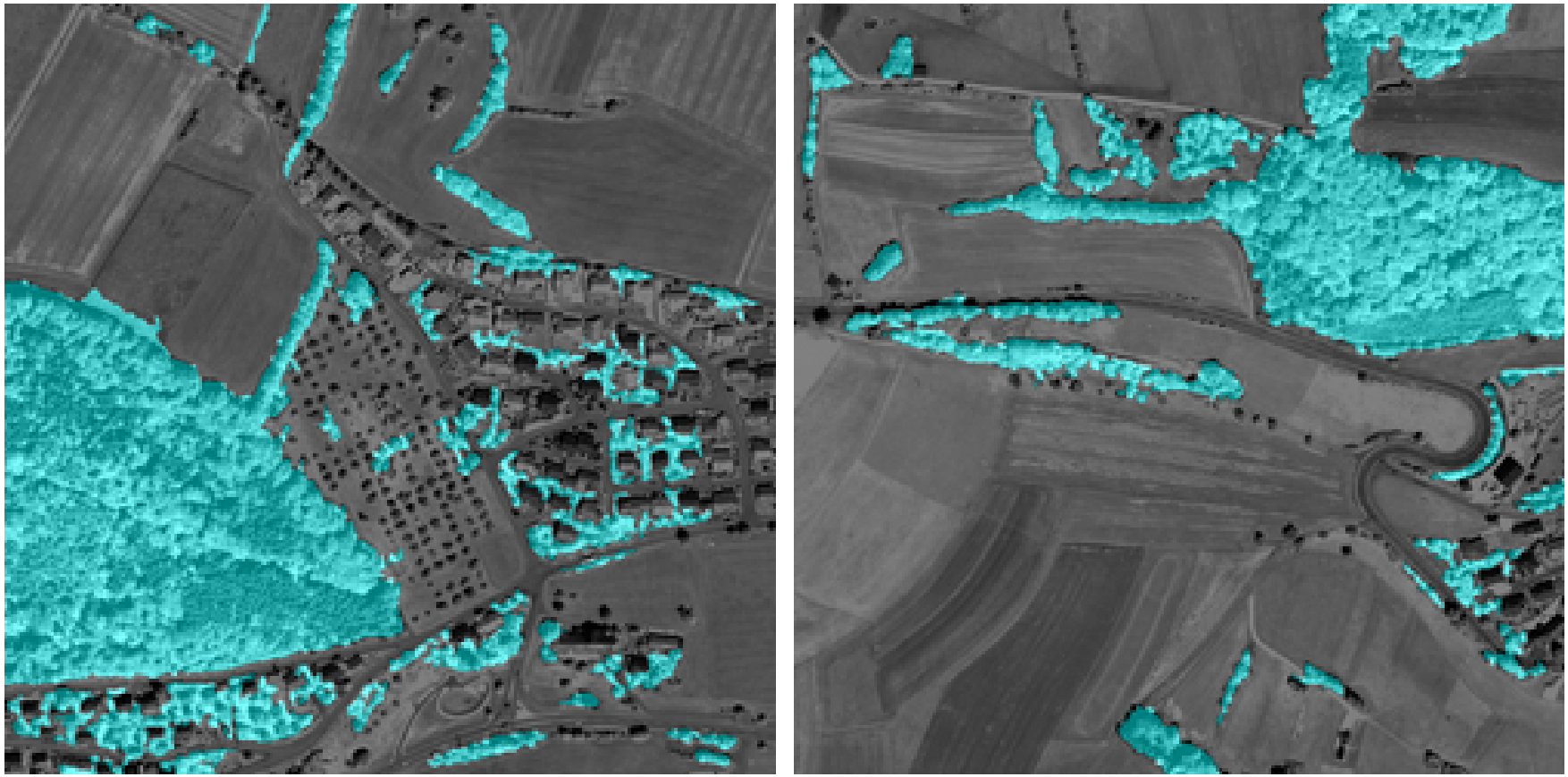
Hedge detection

Identification of candidate objects

- The next step was to find the image areas that gave high responses to the extracted features.
- A two-step decision process was employed:
  1. A threshold on NDVI was used to separate green vegetation from the rest of the land cover.
  2. Classifiers trained on multispectral values and texture features were used to label pixels as belonging to woody vs. non-woody vegetation.

Hedge detection

Identification of candidate objects

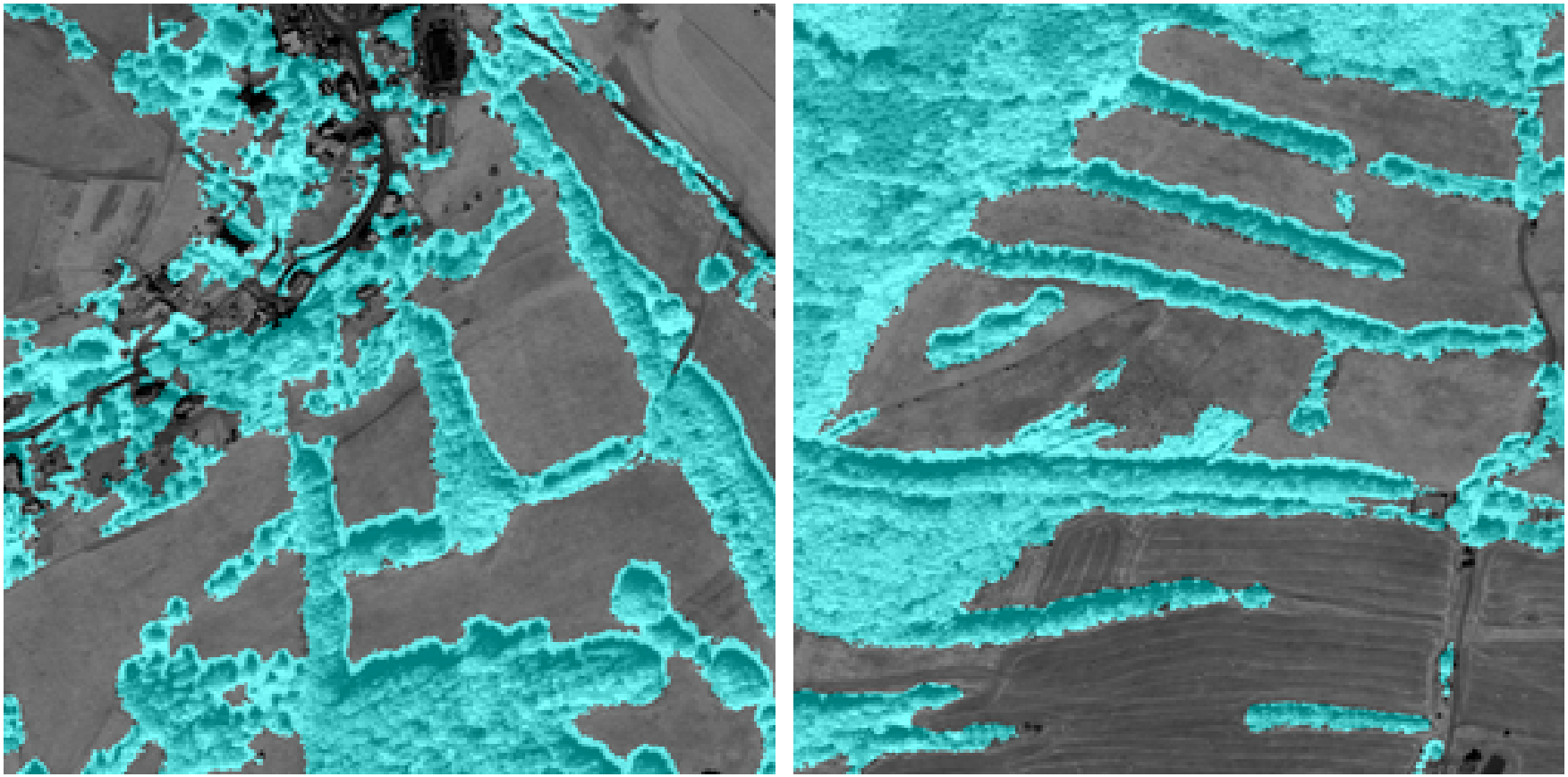


(a) (b)

Figure 20: Woody vs. non-woody vegetation classification.

Hedge detection

Identification of candidate objects

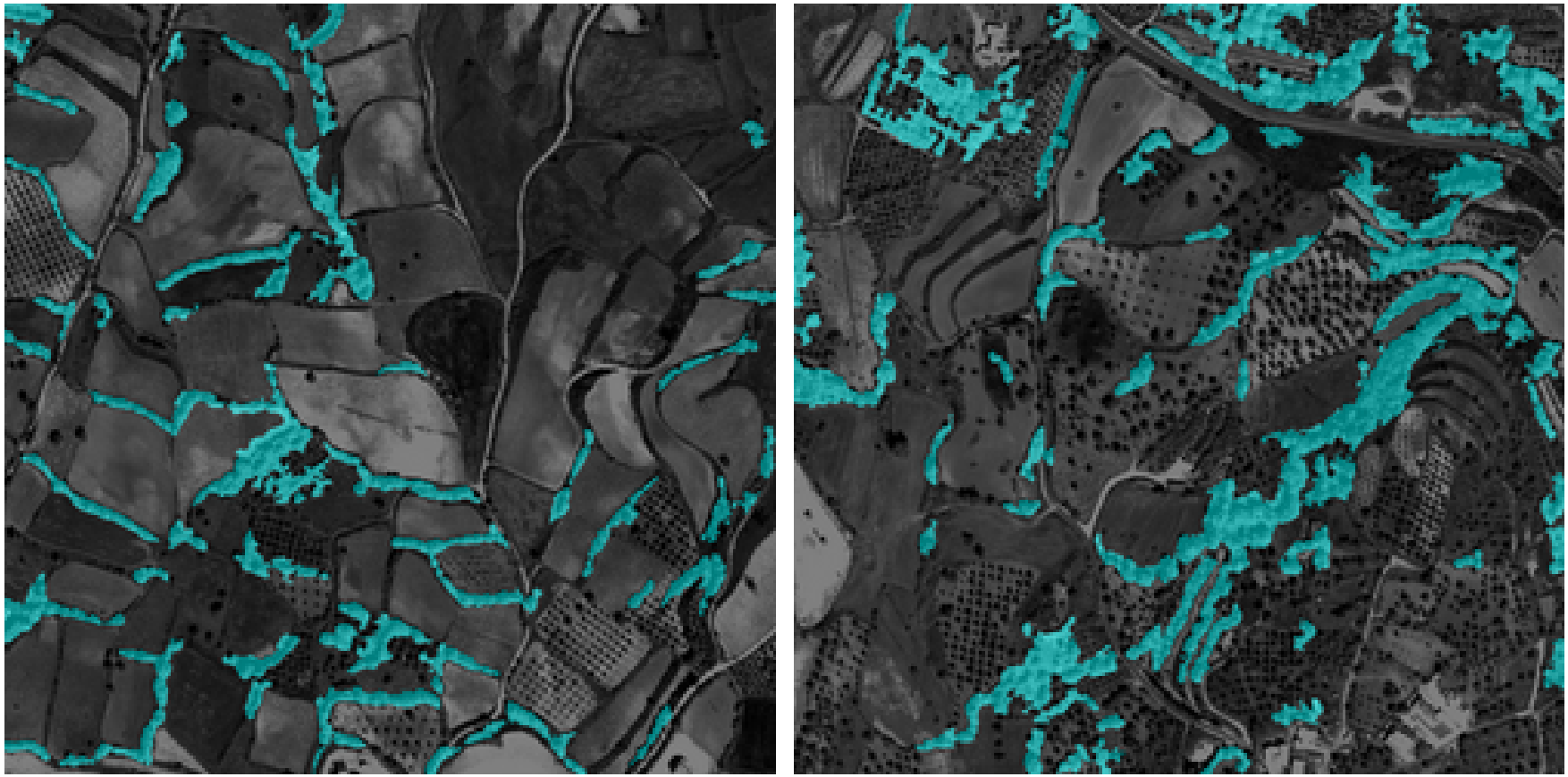


(a) (b)

Figure 21: Woody vs. non-woody vegetation classification.

Hedge detection

Identification of candidate objects



(a) (b)

Figure 22: Woody vs. non-woody vegetation classification.

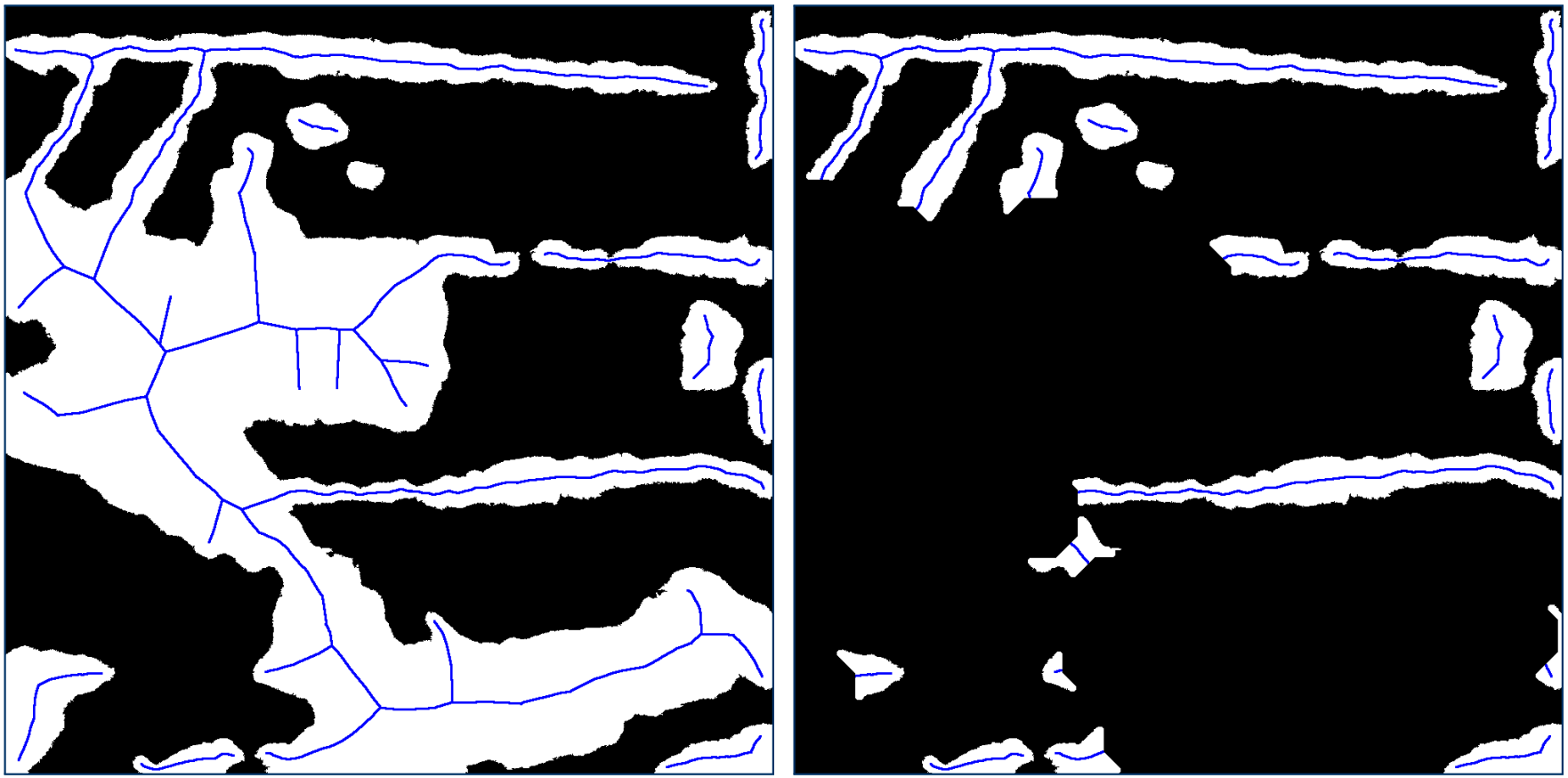
Hedge detection

Detection of target objects

- Given the candidate objects, object level shape information was used so that the objects could be labeled as target or rejected.
- The detection method used
  - morphological filtering to locate the woody vegetation areas that fell within the width limits of an acceptable hedge, and
  - skeletonization and an iterative least-squares fitting procedure that quantified the linearity of the objects.
- Aspect ratio (length/width) was computed as the shape feature.

Hedge detection

Detection of target objects

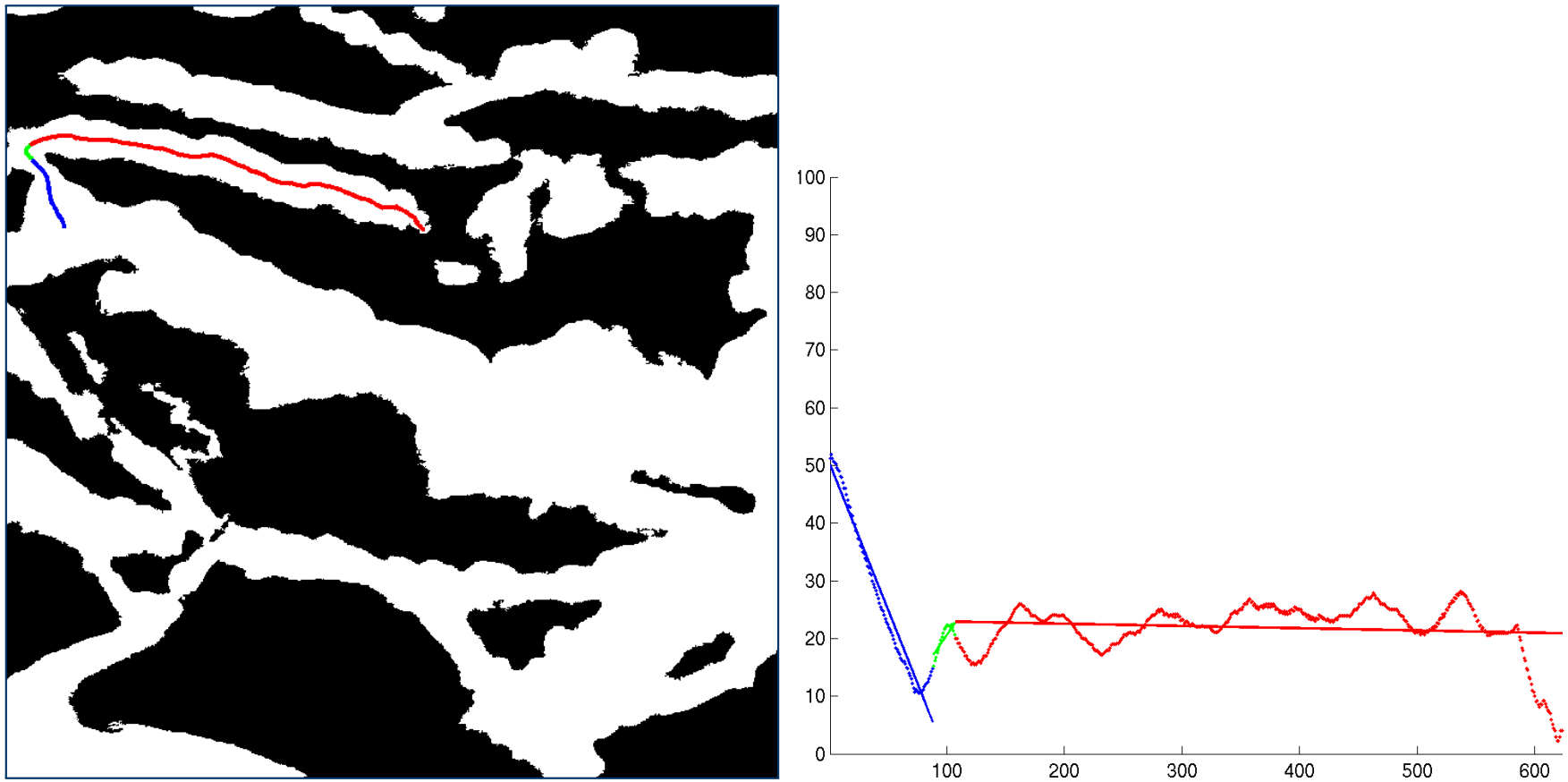


(a) (b)

Figure 23: Elimination of large woody areas using morphological filtering.

Hedge detection

Detection of target objects

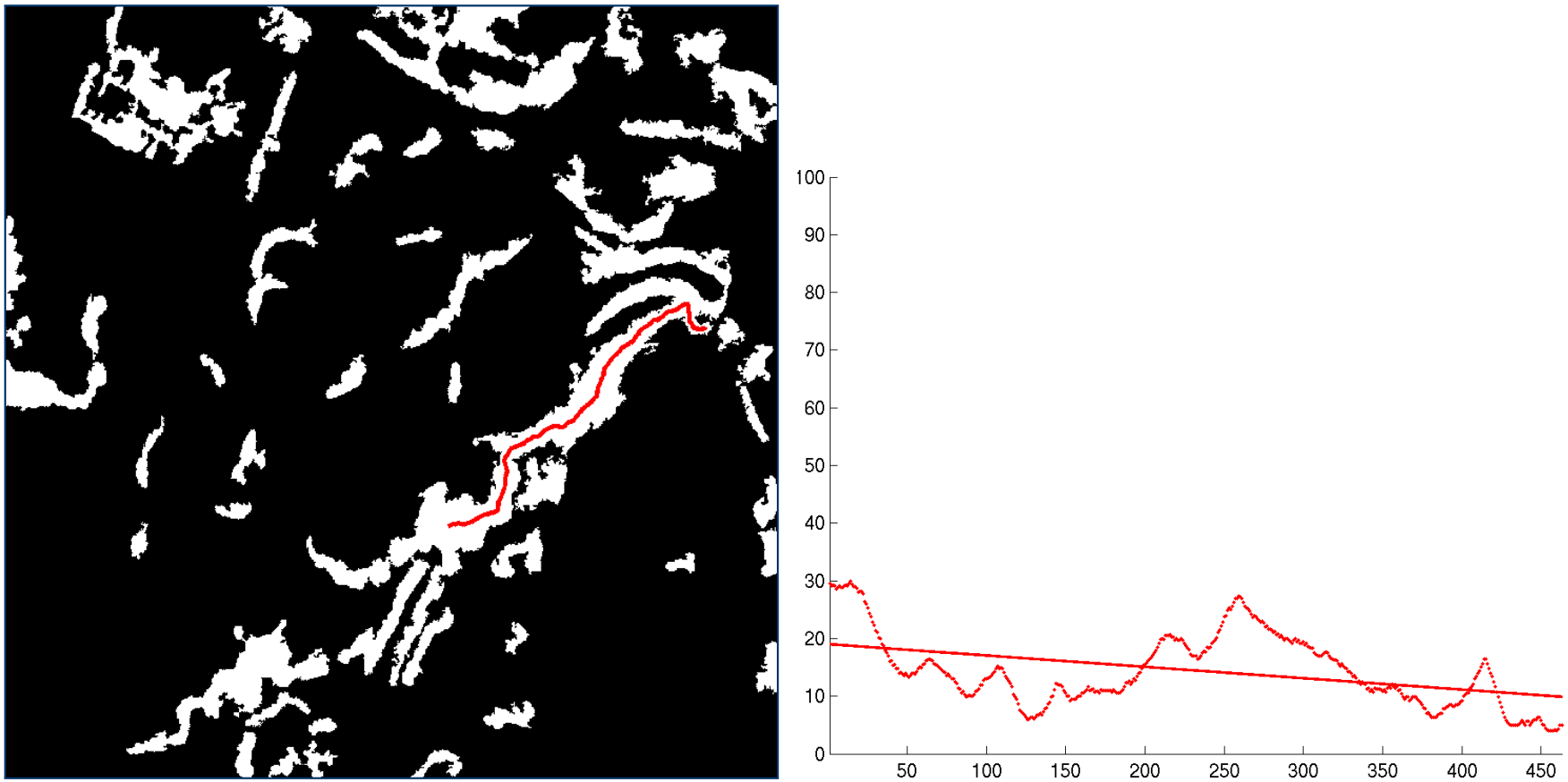


(a) (b)

Figure 24: Iterative least-squares line fitting based segment selection.

Hedge detection

Detection of target objects

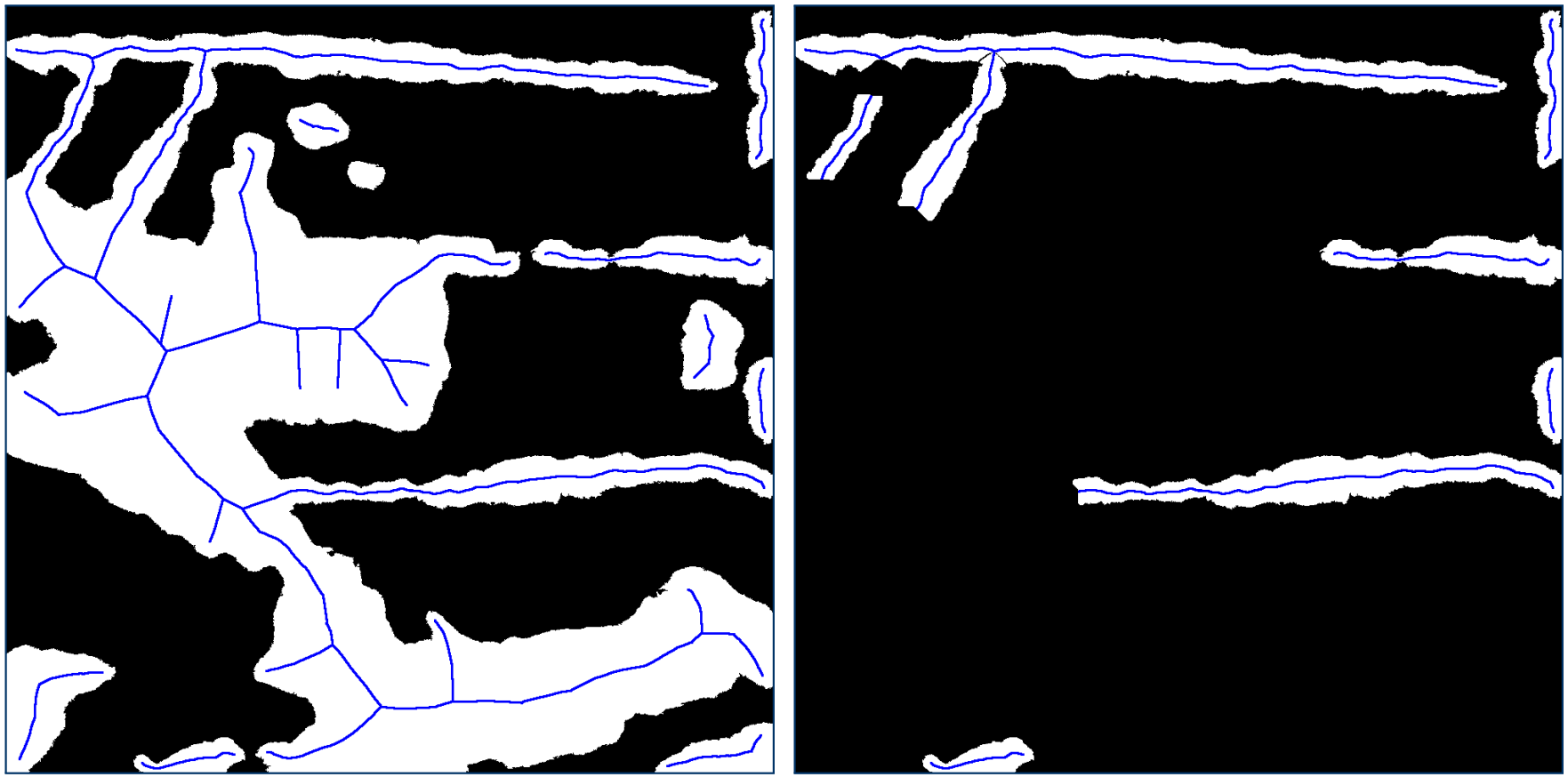


(a) (b)

Figure 25: Iterative least-squares line fitting based segment selection.

Hedge detection

Detection of target objects

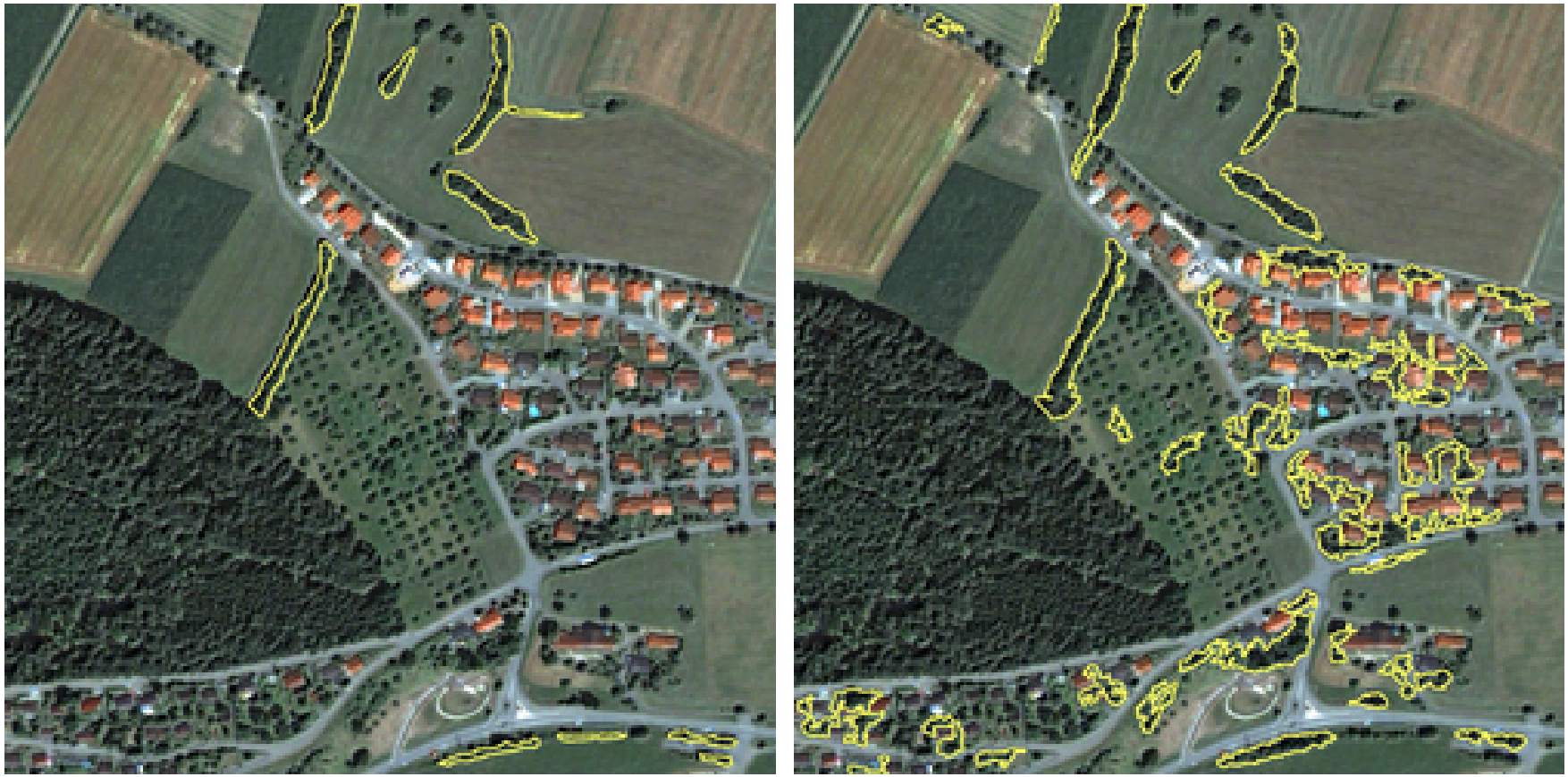


(a) (b)

Figure 26: Final set of segments selected as linear.

Hedge detection

Performance evaluation

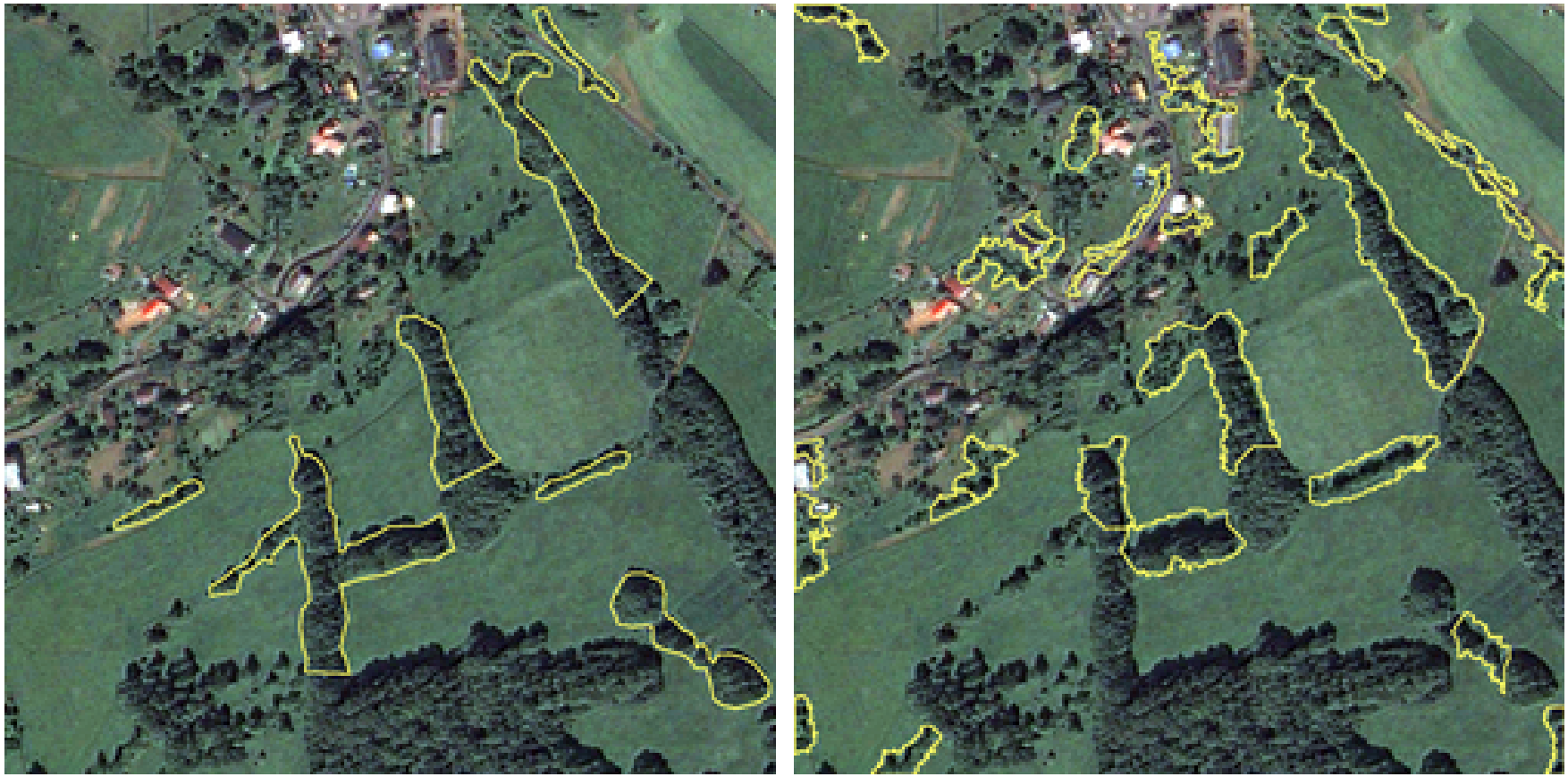


(a) Reference objects (b) Output objects

Figure 27: Example results for hedge detection.

Hedge detection

Performance evaluation

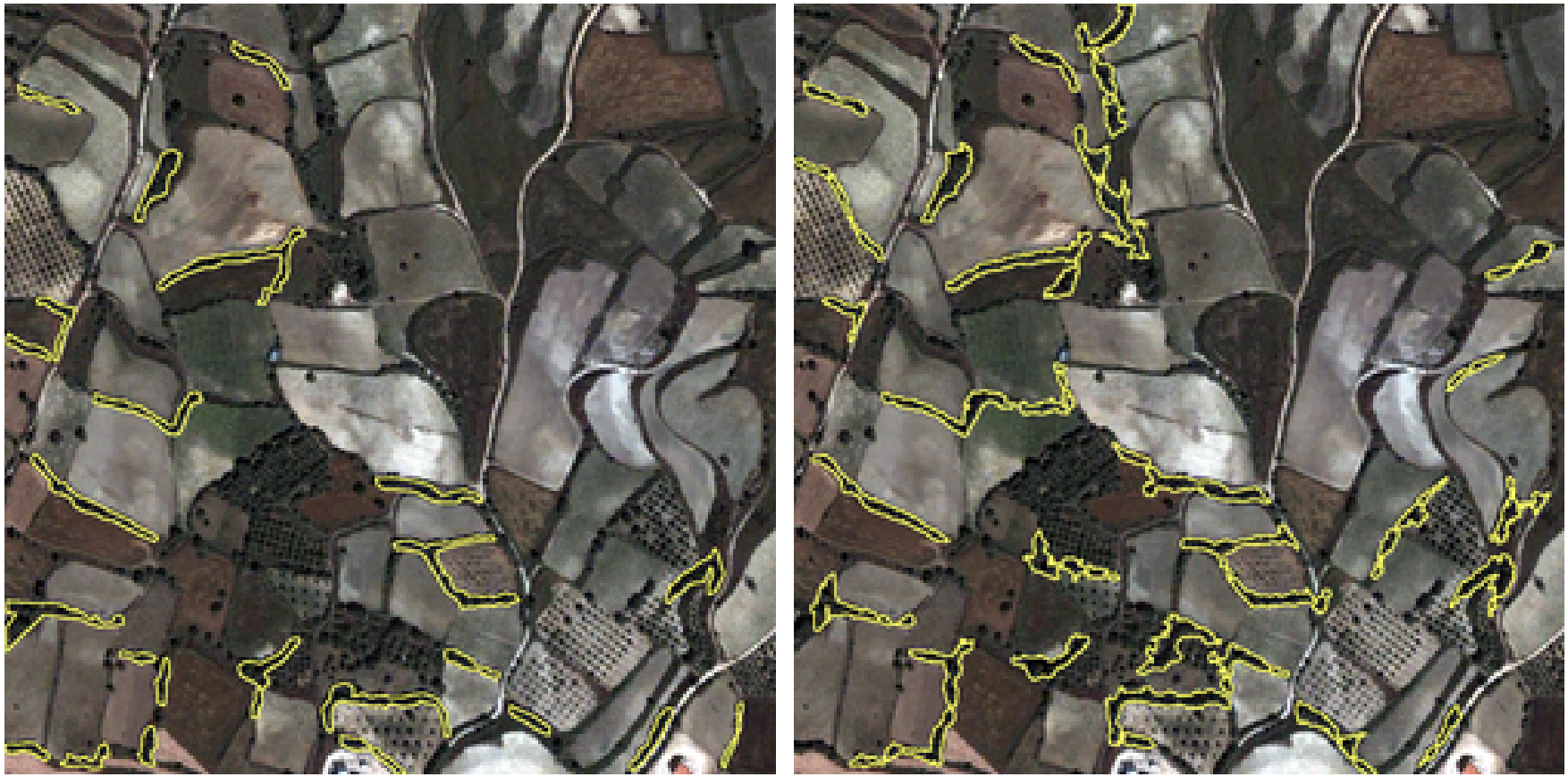


(a) Reference objects (b) Output objects

Figure 28: Example results for hedge detection.

Hedge detection

Performance evaluation



(a) Reference objects (b) Output objects

Figure 29: Example results for hedge detection.

Hedge detection

Performance evaluation

(a) Reference objects (b) Output objects

Figure 30: Example results for hedge detection.

Orchard detection

Overview

- It may not be possible to discriminate between certain terrain classes such as orchards, vineyards, forests, and fields using only spectral information.
- Texture analysis is a promising technique because of its potential for modeling the image data in terms of texture primitives appearing in a repetitive arrangement.
- Our texture model for the orchards involved individual trees and the regularity of their planting patterns.

Orchard detection

Overview

(a) Giresun, Turkey

(b) Giresun, Turkey

Figure 31: Example Quickbird images containing orchards.

Orchard detection

Overview

(a) Izmir, Turkey

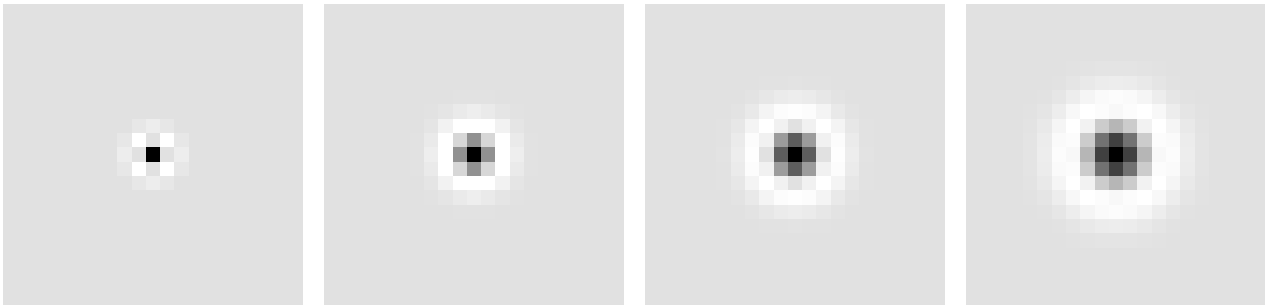
(b) Izmir, Turkey

Figure 32: Example Google Earth images containing orchards.

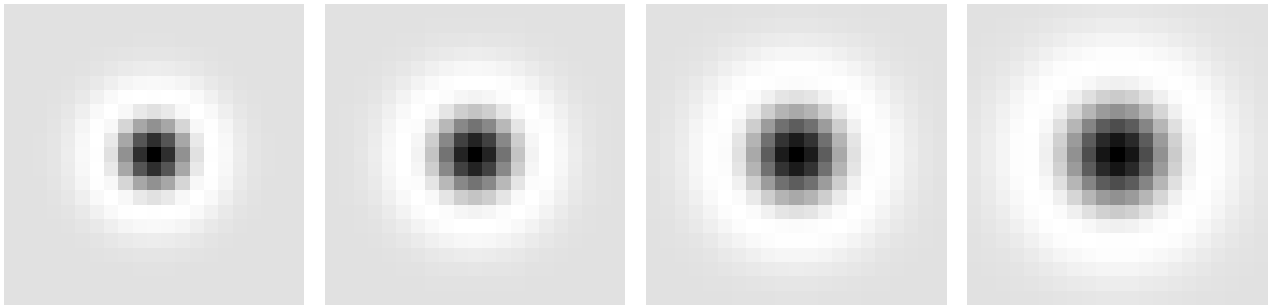
Orchard detection

Pre-processing

- The first step was the enhancement of tree-like objects using multi-scale isotropic filters.



(a) Gran. 2 (b) Gran. 3 (c) Gran. 4 (d) Gran. 5

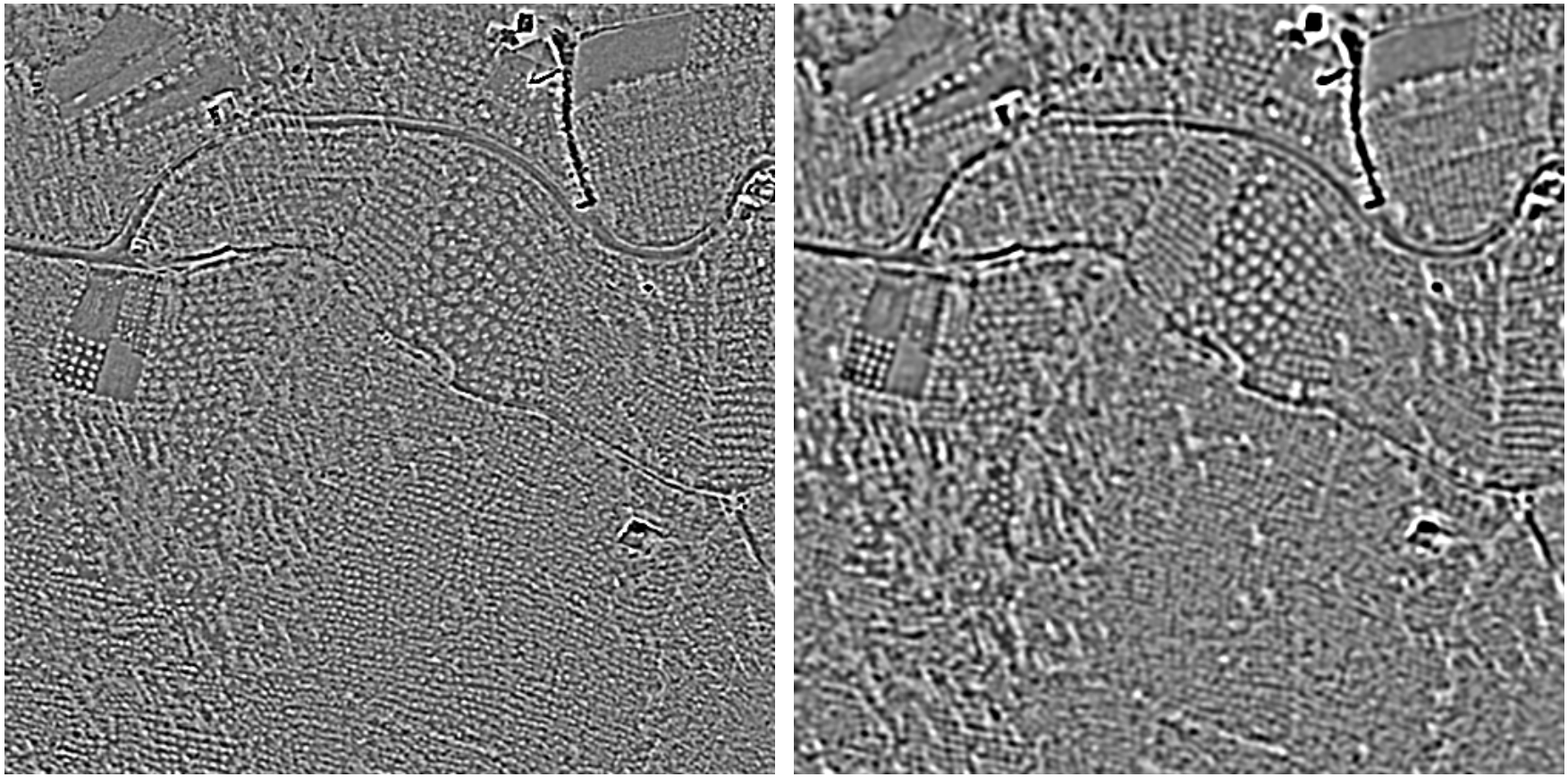


(e) Gran. 6 (f) Gran. 7 (g) Gran. 8 (h) Gran. 9

Figure 33: Spot filters for different granularities.

Orchard detection

Pre-processing



(a) Granularity (scale) 3 (b) Granularity (scale) 6

Figure 34: Example spot filter outputs.

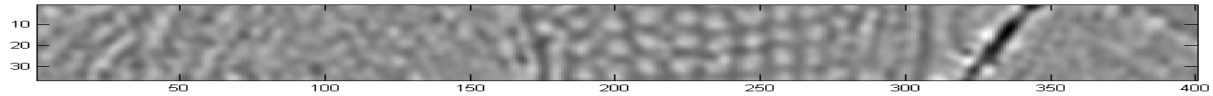
Orchard detection

Projection profiles and regularity detection

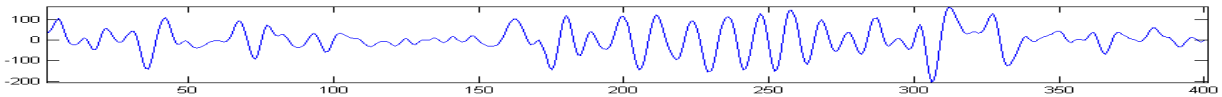
- The pixels corresponding to local maxima in the filter responses indicated possible locations of such objects.
- In a neighborhood with a regular structure, the locations of local maxima along a scan line with an orientation that matched the dominant direction of this structure also had a regular pattern.
- We used projection profiles to quantify the regularity of the trees along different orientations.
- The scores for a particular pixel for all orientations and all granularities were denoted as the regularity spectrum of that pixel.

Orchard detection

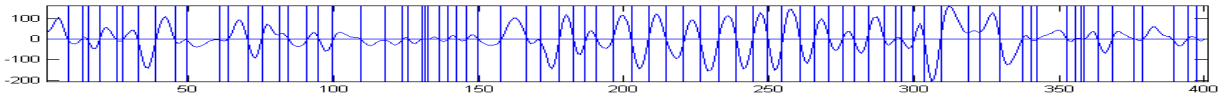
Projection profiles and regularity detection



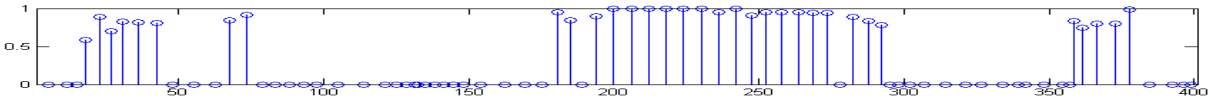
(a) A window cropped from the spot filter response of a QuickBird image



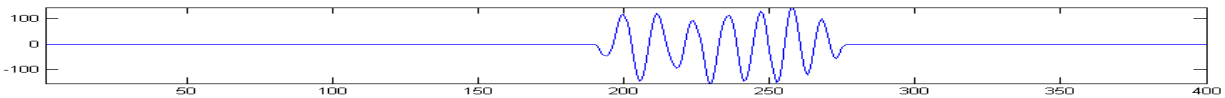
(b) Vertical projection profile of the window



(c) Segmentation of the profile into its peaks and valleys



(d) Regularity scores

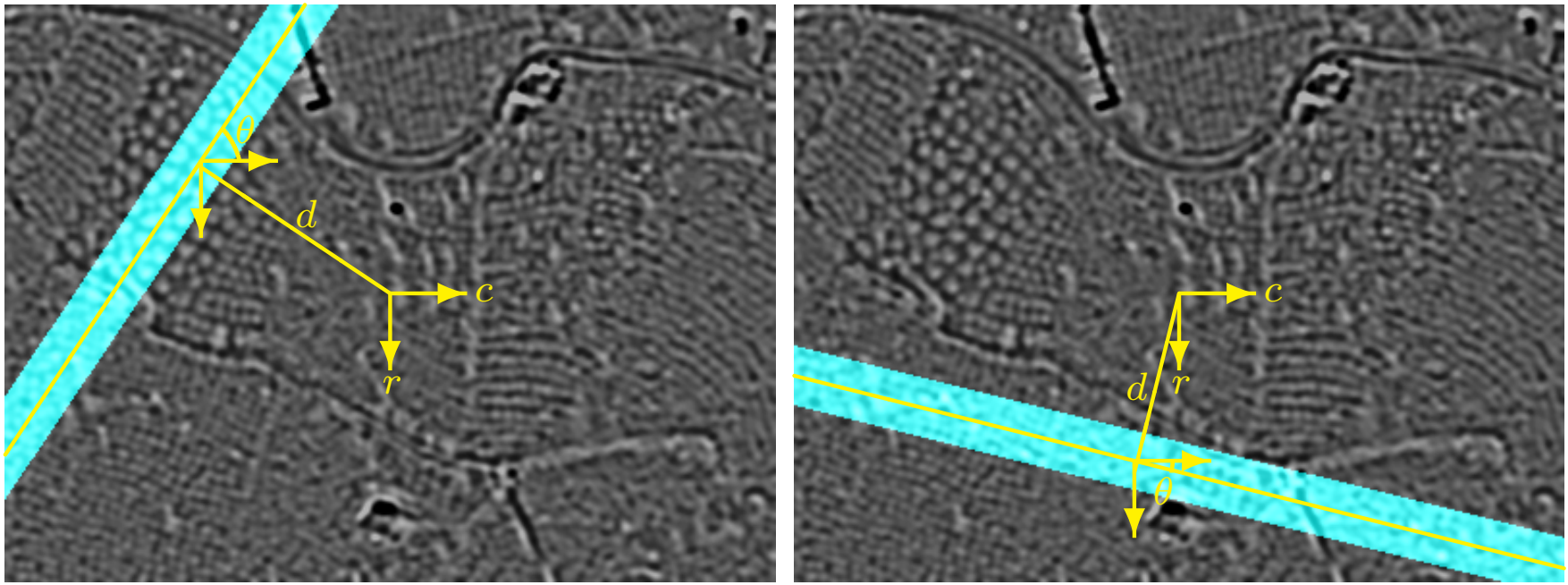


(e) Periodic intervals located according to the largest values in (d)

Figure 35: Periodicity analysis of the projection profile of an image window.

Orchard detection

Multi-orientation and multi-granularity regularity analysis



(a) $d = -120, \theta = -55°, \delta = 30$ (b) $d = 90, \theta = 15°, \delta = 30$

Figure 36: Example windows for computing the projection profiles.

Orchard detection

Multi-orientation and multi-granularity regularity analysis

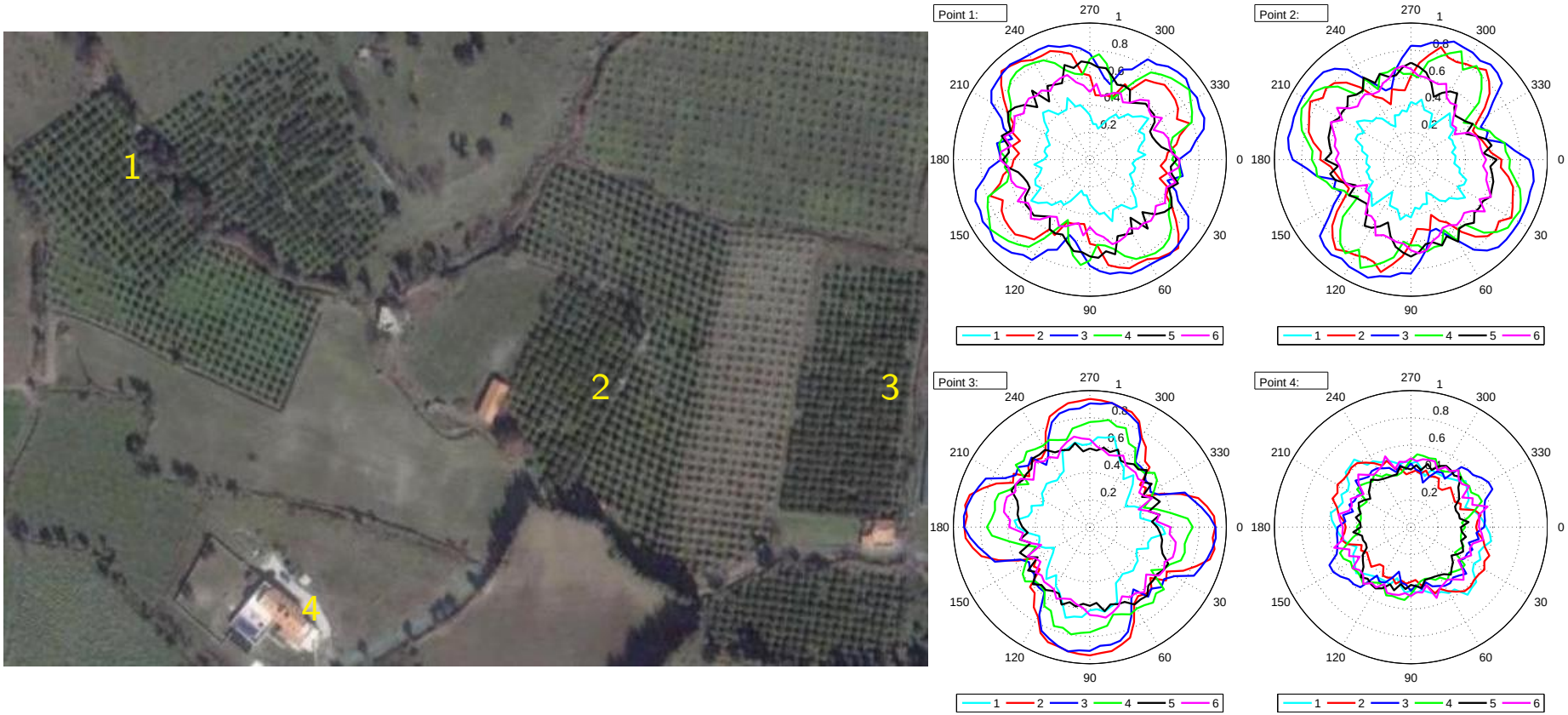


Figure 37: Example regularity spectra for pixels belonging to different structures.

Orchard detection

Regularity spectrum and texture segmentation

- Pixels having very high regularity scores were used as seeds.
- A region growing process was run to extend these seeds toward neighboring pixels having similar spectra.
- A second growing process that involved merging regions was used to obtain the final segmentation.

Orchard detection

Regularity spectrum and texture segmentation

(a) Reference data

(b) Seeds for $\tau_h = 0.85$

(c) Candidates for $\tau_l = 0.80$

(d) Growing results for $\tau_d = 0.05$

(e) Merging results for $\tau_d = 0.05$

(f) Merging results for $\tau_d = 0.06$

Figure 38: Illustration of the segmentation process.

Orchard detection

Performance evaluation

(a) Multispectral image

Figure 39: Example results for orchard detection in the Giresun data set.

Orchard detection

Performance evaluation

(a) Multispectral image (b) Reference data

Figure 39: Example results for orchard detection in the Giresun data set.

Orchard detection

Performance evaluation

(a) Multispectral image (b) Reference data (c) Regularity scores

Figure 39: Example results for orchard detection in the Giresun data set.

Orchard detection

Performance evaluation

(a) Multispectral image (b) Reference data (c) Detection results

Figure 39: Example results for orchard detection in the Giresun data set.

Orchard detection

Performance evaluation

(a) Multispectral image

Figure 40: Example results for orchard detection in the Izmir data set.

Orchard detection

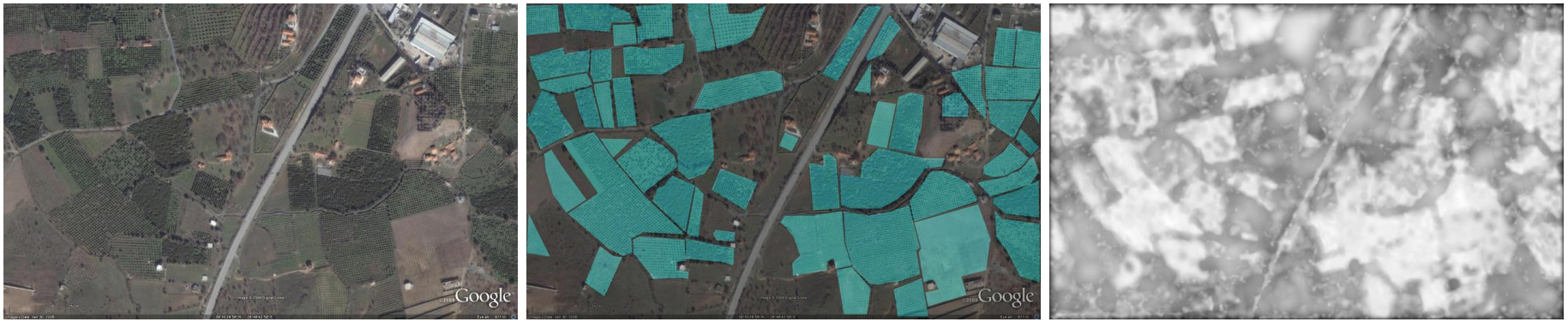
Performance evaluation

(a) Multispectral image (b) Reference data

Figure 40: Example results for orchard detection in the Izmir data set.

Orchard detection

Performance evaluation

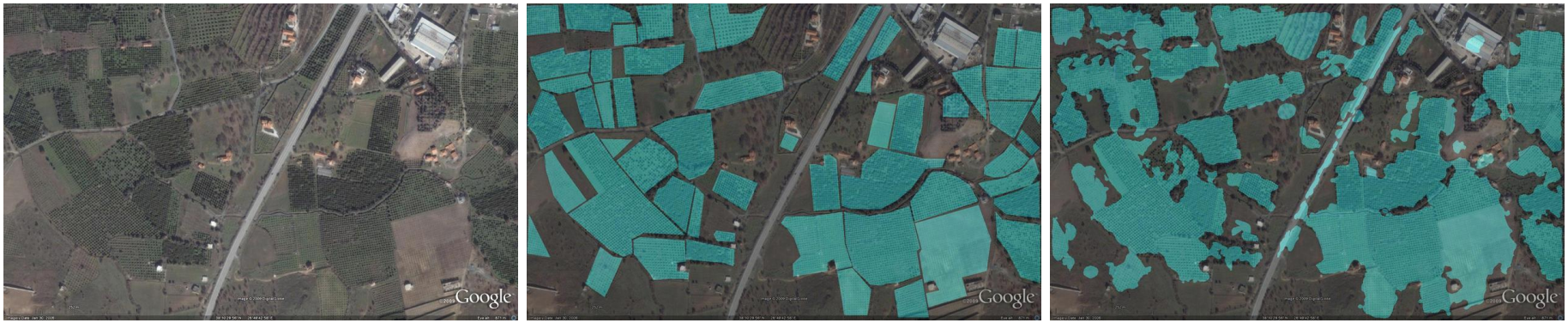


(a) Multispectral image (b) Reference data (c) Regularity scores

Figure 40: Example results for orchard detection in the Izmir data set.

Orchard detection

Performance evaluation

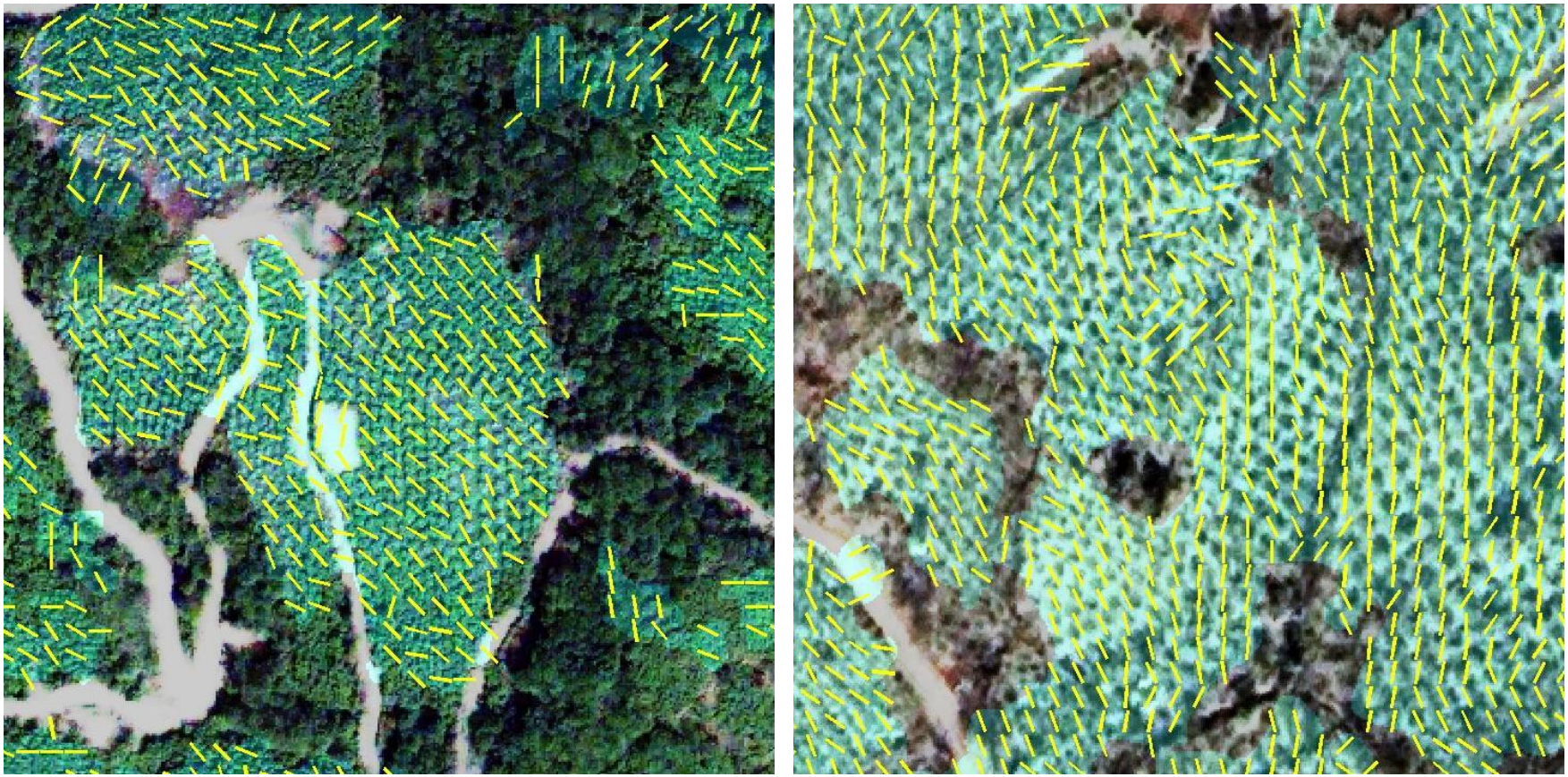


(a) Multispectral image (b) Reference data (c) Detection results

Figure 40: Example results for orchard detection in the Izmir data set.

Orchard detection

Performance evaluation



(a) (b)

Figure 41: Local details of orchard detection in the Giresun data set.

Orchard detection

Performance evaluation

(a) (b)

Figure 42: Local details of orchard detection in the Izmir data set.

Orchard detection

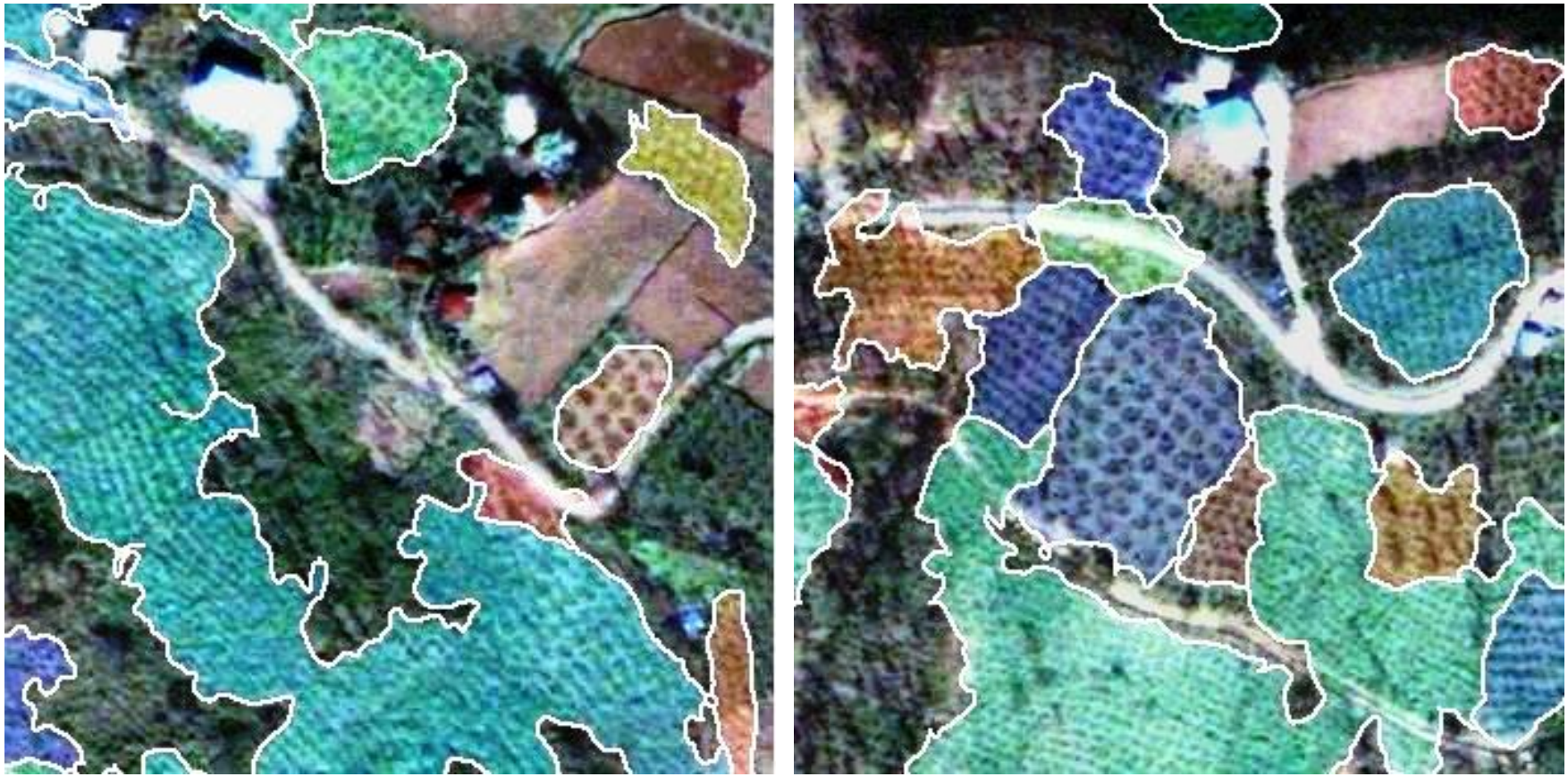
Performance evaluation

(a) (b)

Figure 43: Local details of orchard segmentation in the Giresun data set.

Orchard detection

Performance evaluation

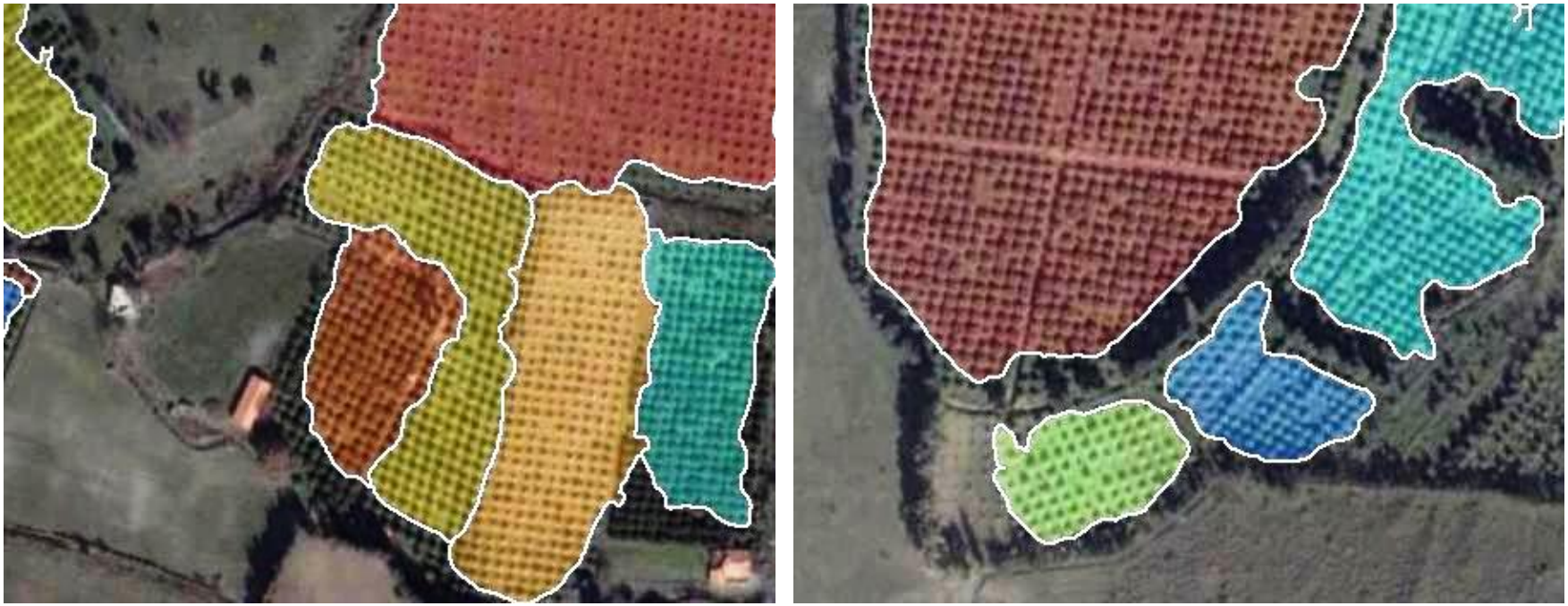


(a) (b)

Figure 44: Local details of orchard segmentation in the Giresun data set.

Orchard detection

Performance evaluation



(a) (b)

Figure 45: Local details of orchard segmentation in the Izmir data set.

Orchard detection

Performance evaluation

(a) (b)

Figure 46: Local details of orchard segmentation in the Izmir data set.

Part III

Detection of Compound Structures in Very High Spatial Resolution Images

Introduction

- A common approach for object recognition is to segment the images into homogeneous regions.
- However, such homogeneous regions often correspond to very small details in very high spatial resolution (VHR) images.
- An alternative is to model the spatial arrangements of simple image regions to identify complex region groups.
- Examples of such region groups, also called *compound structures*, include different types of residential, commercial, industrial, and agricultural areas.

Introduction

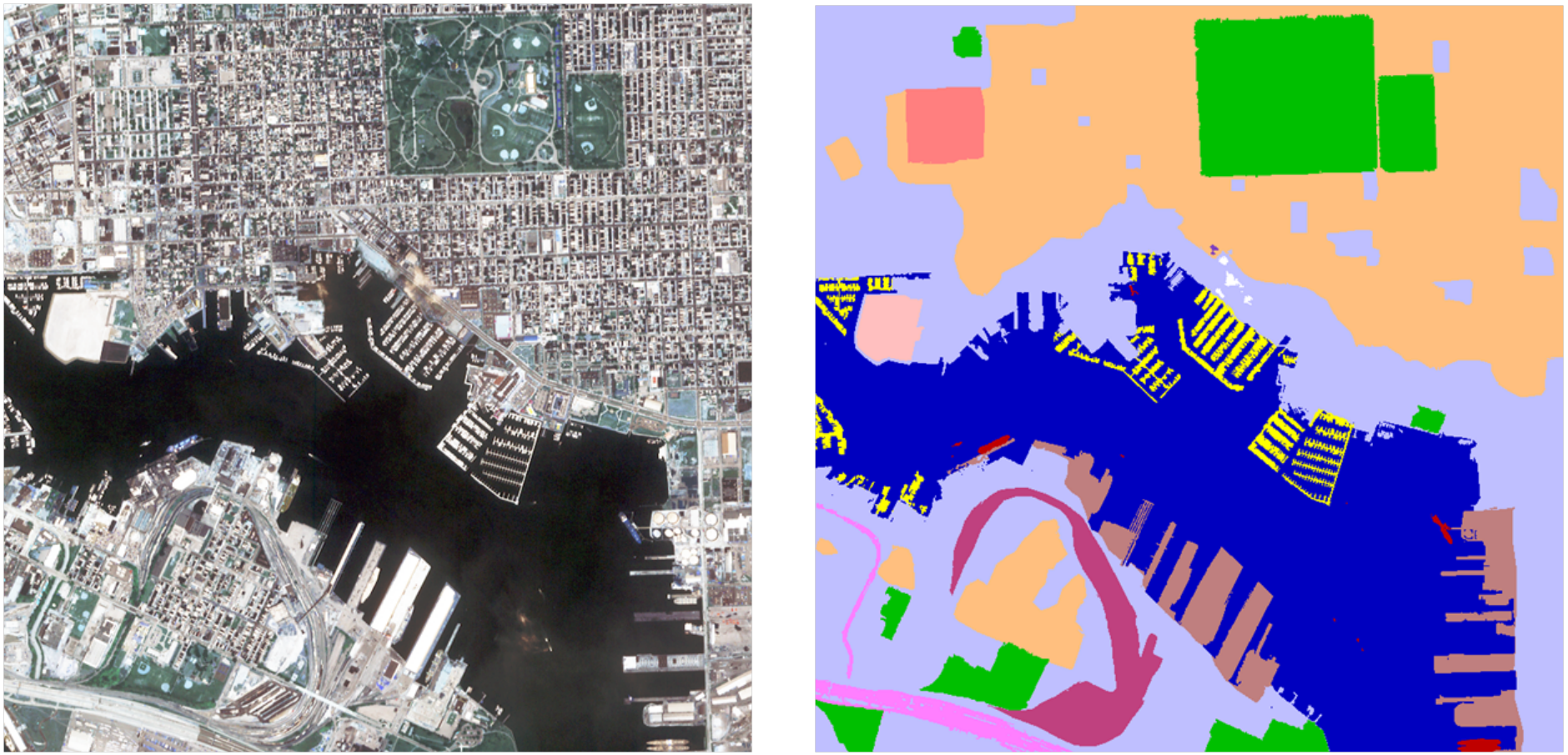


Figure 47: An Ikonos image of Baltimore, and some compound structures of interest: residential, commercial, park, marina, housing project, etc.

Introduction

- Compound structures are comprised of different spatial arrangements of *primitive objects*.
- Our framework involves *statistical* modeling of the features of primitive objects and *structural* modeling of their arrangements using graphs.
- We have developed methods that use
  - graph-based knowledge discovery,
  - graph-based texture analysis,
  - hierarchical clustering.

Detection using graph-based knowledge discovery

- Compound structures can be defined in terms of frequent occurrences of primitive region types in particular spatial arrangements.
- Given a segmentation, features of neighboring region pairs are incorporated in a spatial co-occurrence space.
- Density estimation in the co-occurrence space identifies groups of related region pairs.

Joint work with Daniya Zamalieva, Bilkent University, and James C. Tilton, NASA Goddard Space Flight Center

Detection using graph-based knowledge discovery

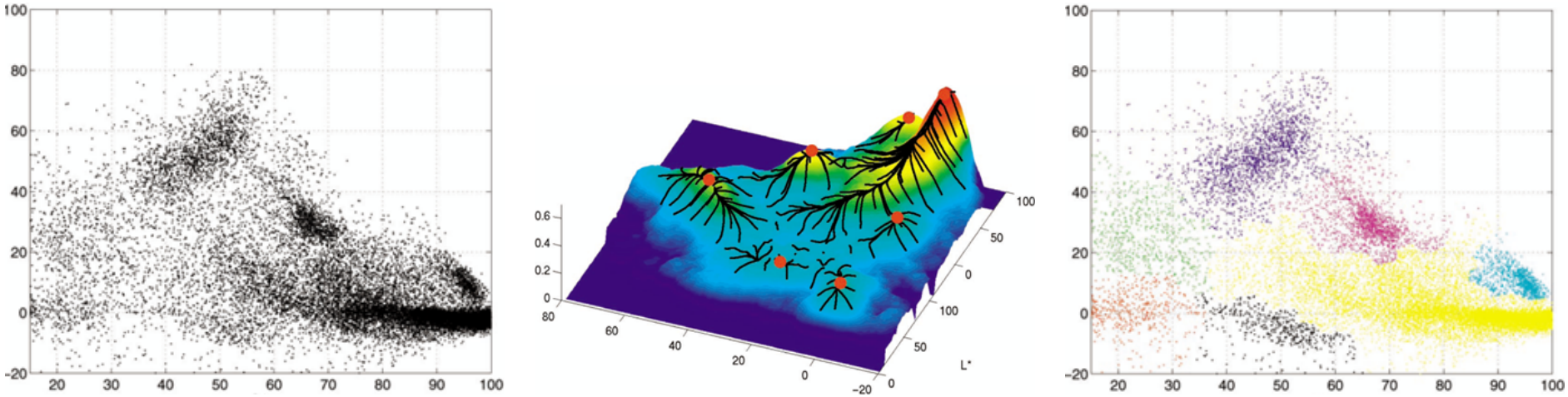


Figure 48: A 2D illustration of the spatial co-occurrence space.

Detection using graph-based knowledge discovery

- A graph is used to encode the spatial structure where there is a vertex for each primitive region and the edges connect the vertex pairs that correspond to the modes of the density estimate.
- Finally, a frequent subgraph discovery algorithm produces parts of high-level compound structures.

Detection using graph-based knowledge discovery

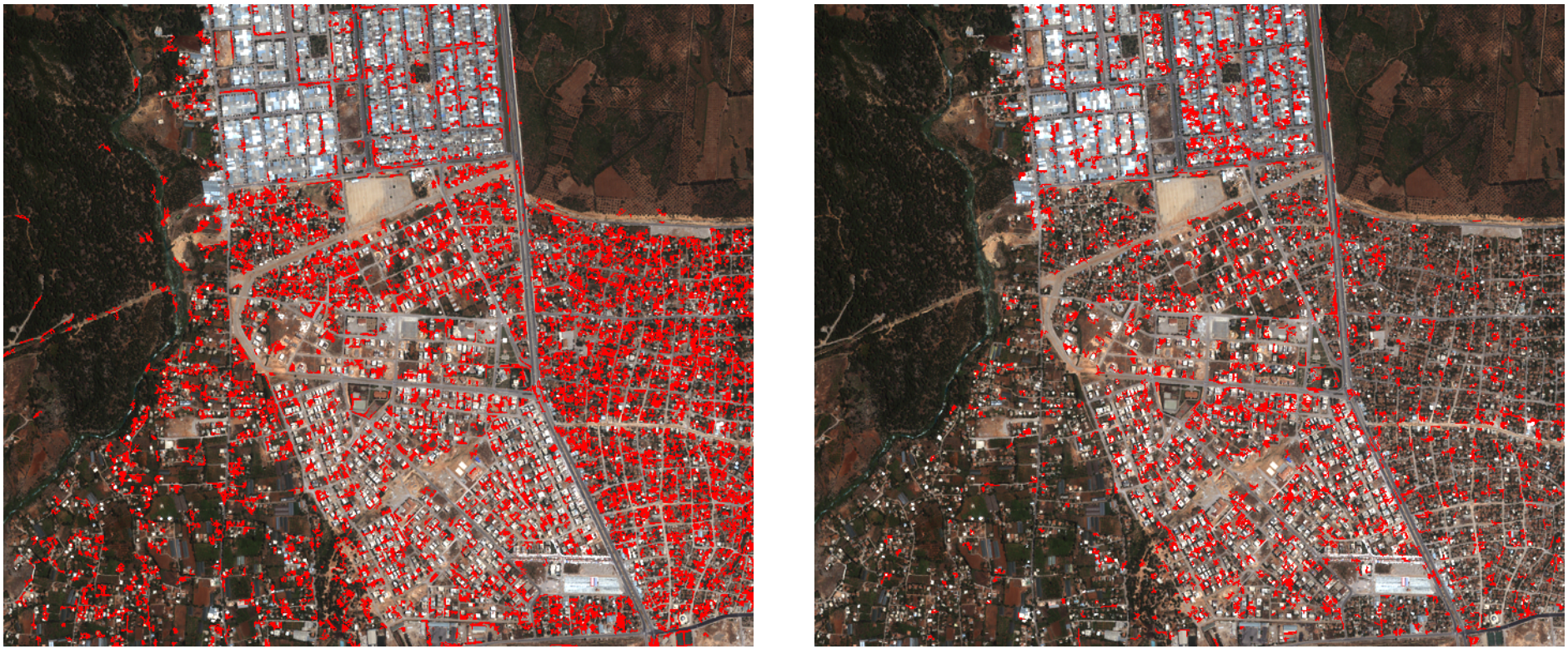


Figure 49: Example substructures obtained by graph analysis, and the corresponding region groups in a multispectral Ikonos image of Antalya.

Detection using graph-based knowledge discovery

Figure 50: Example substructures obtained by graph analysis, and the corresponding region groups in a multispectral Ikonos image of Antalya.

Detection using graph-based knowledge discovery

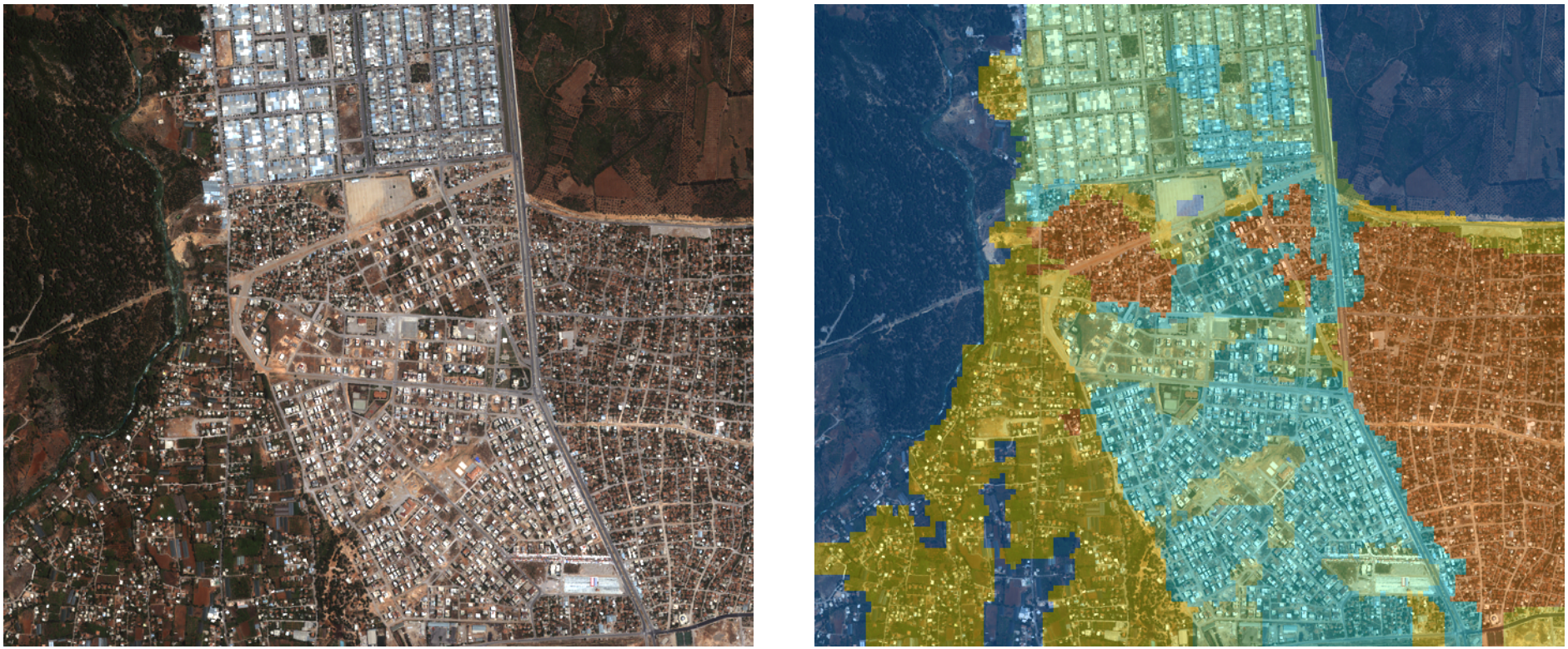


Figure 51: Example segmentation obtained by clustering the subgraph histograms within sliding image windows.

Detection using graph-based texture analysis

- Another model that encodes the spatial arrangements of primitive regions uses relative angles.
- After finding the neighbors of each primitive, the goal is to group these primitives into clusters so that they can be automatically classified as regular or irregular.
- To determine the most important neighbors of each primitive, the minimum spanning tree of the graph is constructed using the distances of Voronoi neighbors.

Joint work with Emel Doğrusöz, Bilkent University

Detection using graph-based texture analysis

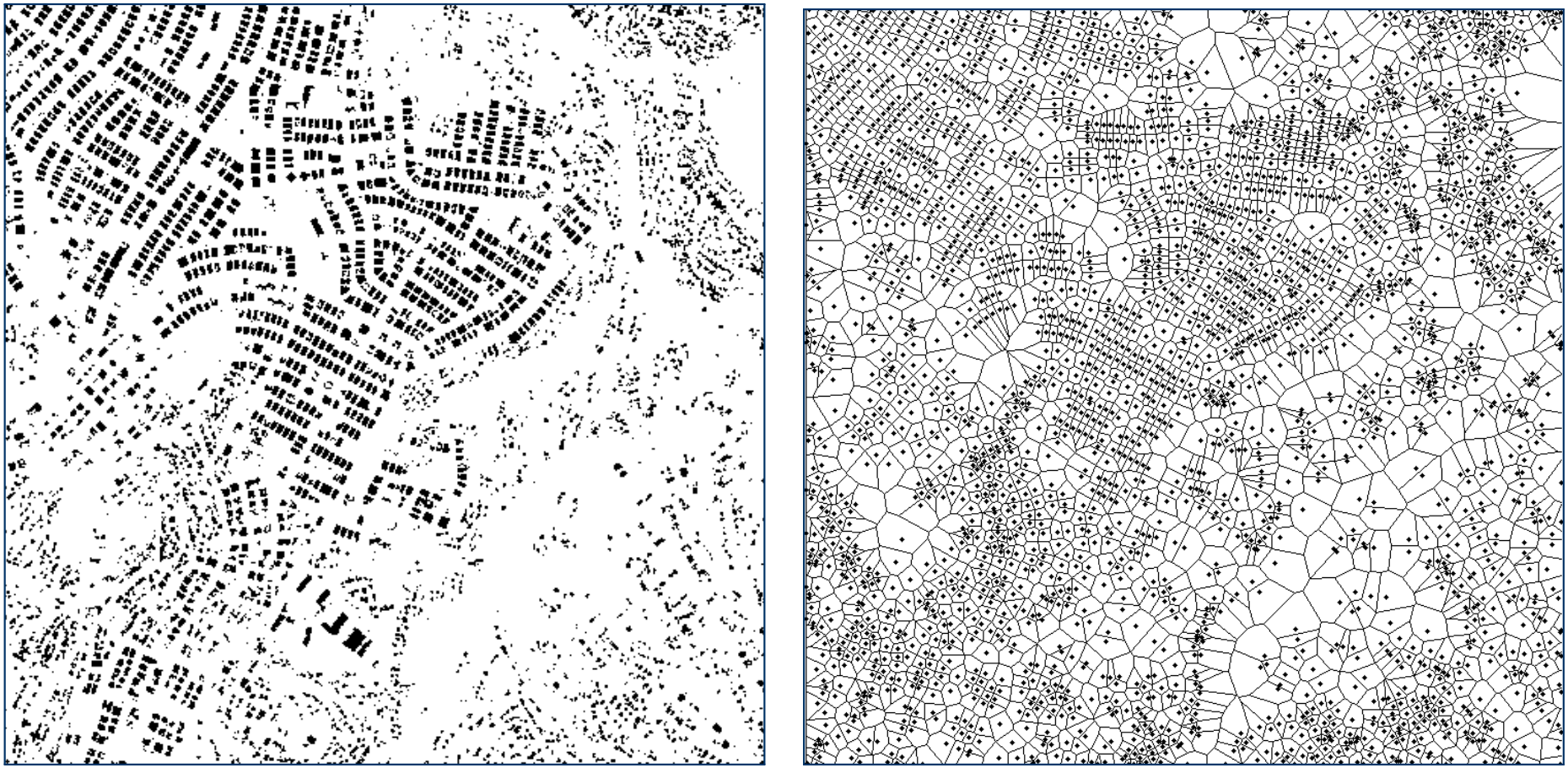
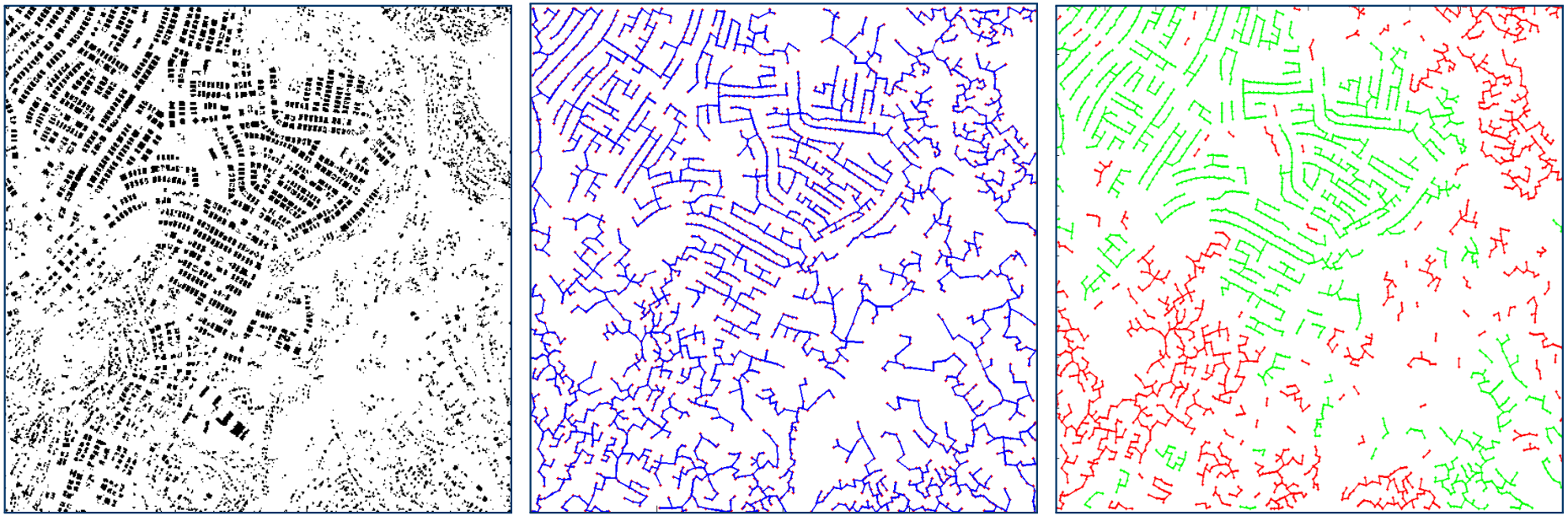


Figure 52: Example for modeling neighboring objects using Voronoi tessellation of a map of buildings detected in an Ikonos image of Ankara.

Detection using graph-based texture analysis

- When the angles between primitives are examined, it can be seen that in organized neighborhoods the angle distribution has peaks around 90 and 180 degrees.
- On the other hand, when there is no specific arrangement of primitives, random angle distributions are observed with no considerable peaks.

Detection using graph-based texture analysis



(a) Example image with different settlement patterns (b) Minimum spanning tree (c) Labeled clusters

Figure 53: Example graphs and labeling of clusters (green: organized, red: unorganized).

Detection using graph-based texture analysis

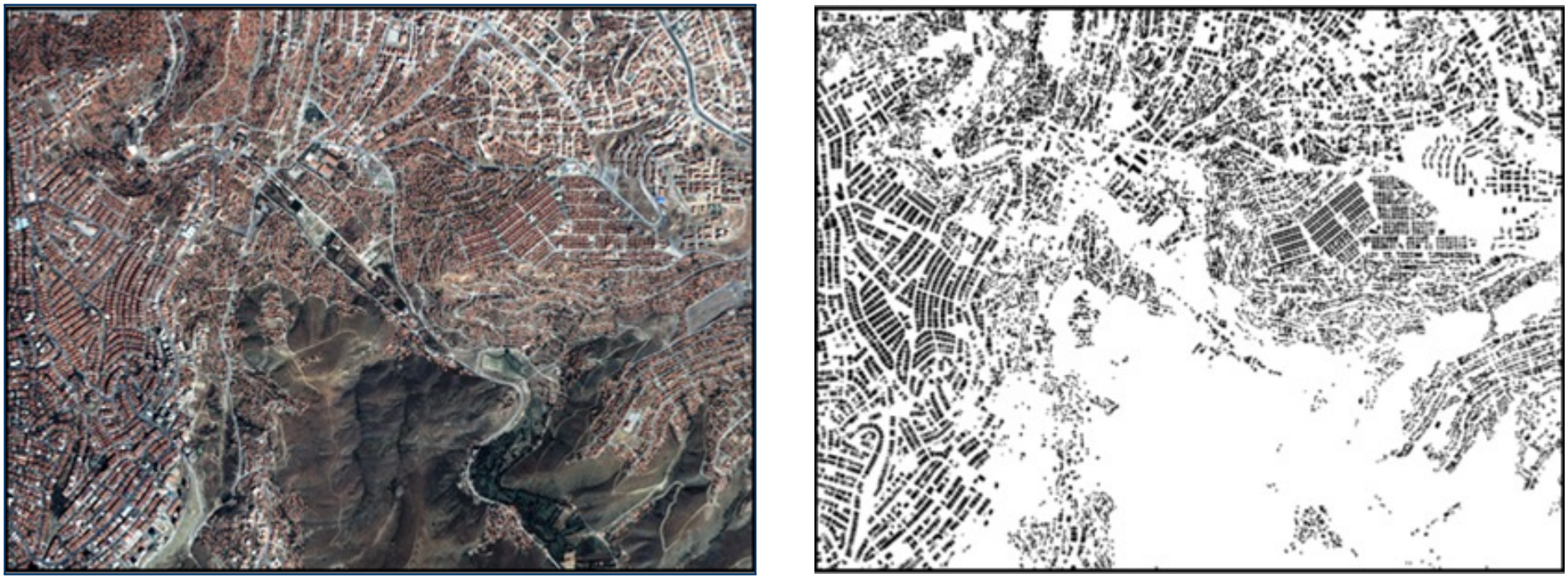


Figure 54: Example results for the detection of organized versus unorganized settlements in a pan-sharpened Ikonos image of Ankara.

Detection using graph-based texture analysis

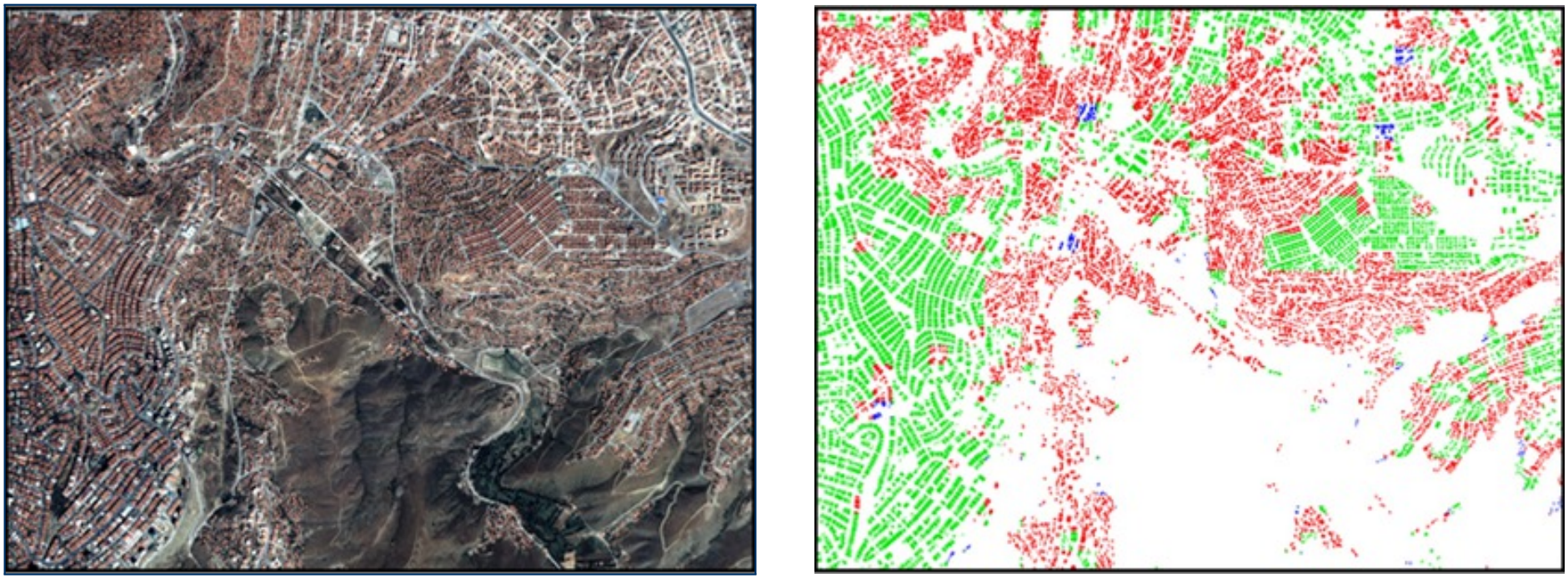


Figure 54: Example results for the detection of organized versus unorganized settlements in a pan-sharpened Ikonos image of Ankara.

Detection using hierarchical clustering

- Our final model uses attributed relational graphs where the primitive objects form the vertices.
- We connect every neighboring vertex pair with an edge.
- We use a threshold on the distance between the centroids of object pairs to determine the neighbors.

Joint work with Gökhan Akçay, Bilkent University

Detection using hierarchical clustering

Figure 55: Examples of graph construction. The vertices considered as neighbors based on proximity analysis are connected with red edges.

Detection using hierarchical clustering

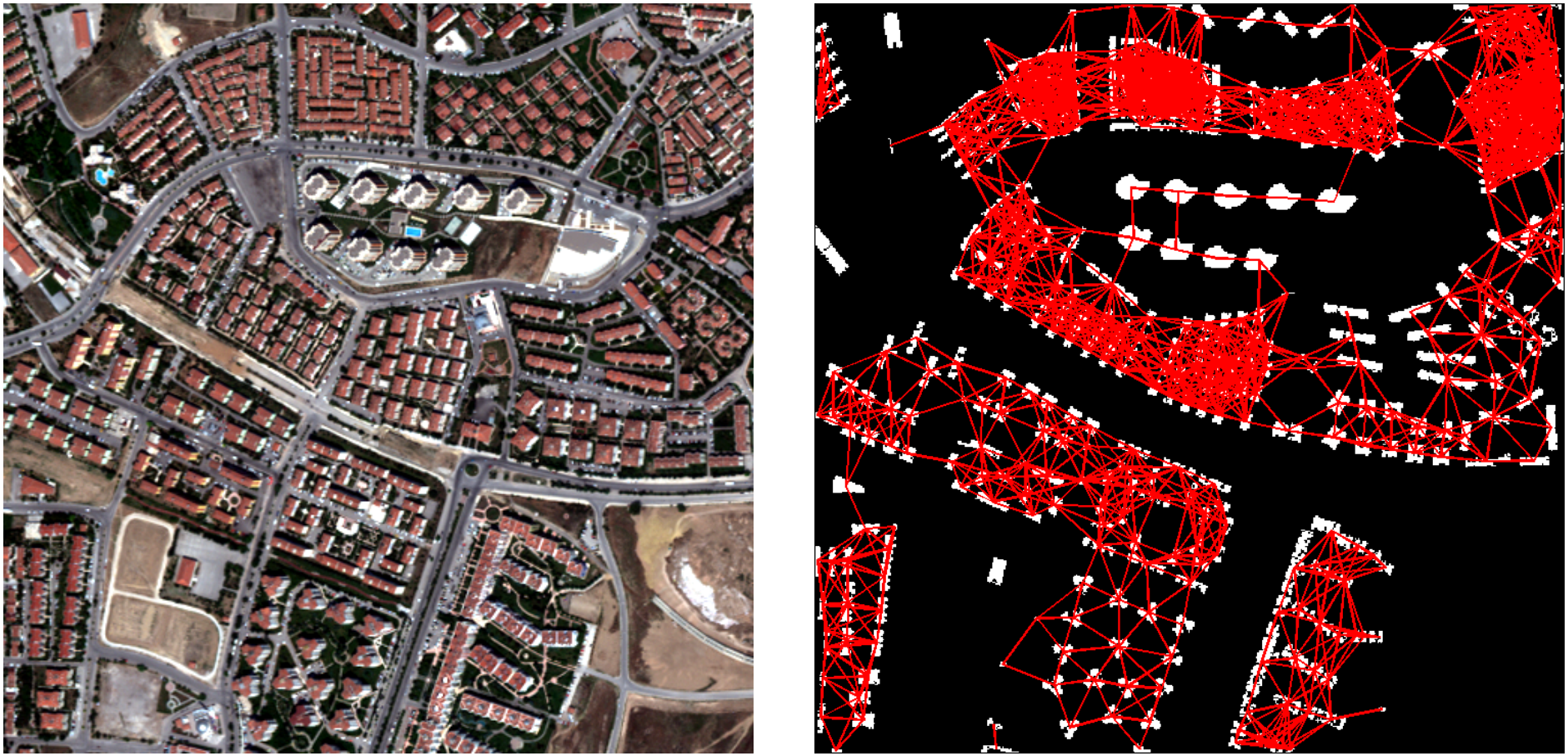


Figure 55: Examples of graph construction. The vertices considered as neighbors based on proximity analysis are connected with red edges.

Detection using hierarchical clustering

- The statistical features that summarize the spectral content and the shape of each individual object consist of
  - mean values of the pixels within the object for each spectral band,
  - area,
  - eccentricity, and
  - centroid location.

Detection using hierarchical clustering

- The structural features represent the spatial layout of each object with respect to its neighbors.
- Aligned groups of objects are found by checking all possible subsets having at least three objects.
- The set of structural features computed for each object group consists of
  - orientation of the fitted line, and
  - mean of the centroid distances.

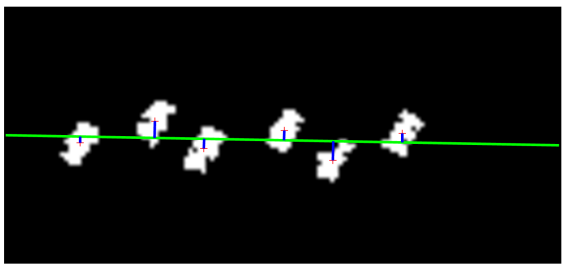
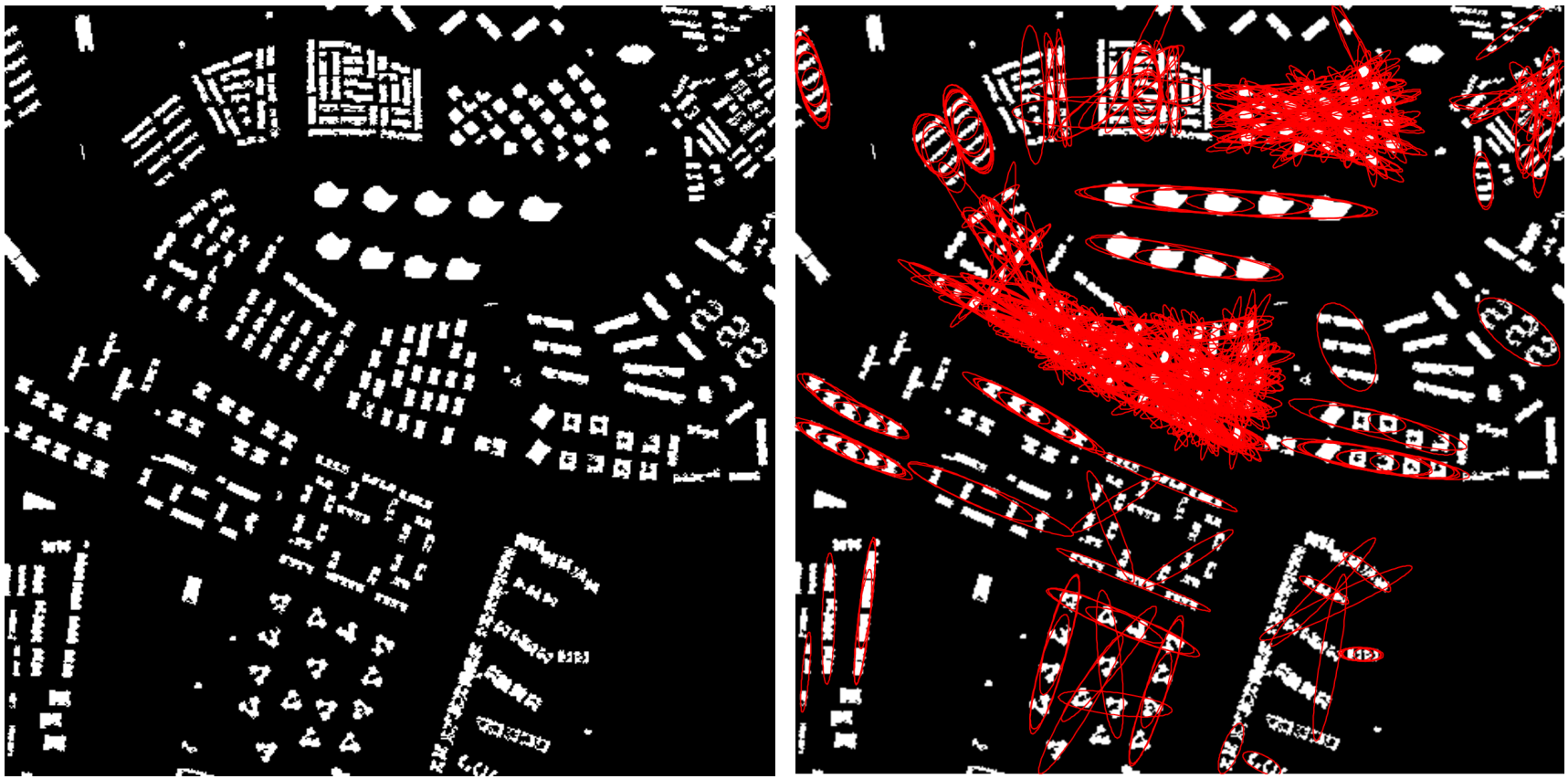


Figure 56: Illustration of object alignment.

Detection using hierarchical clustering



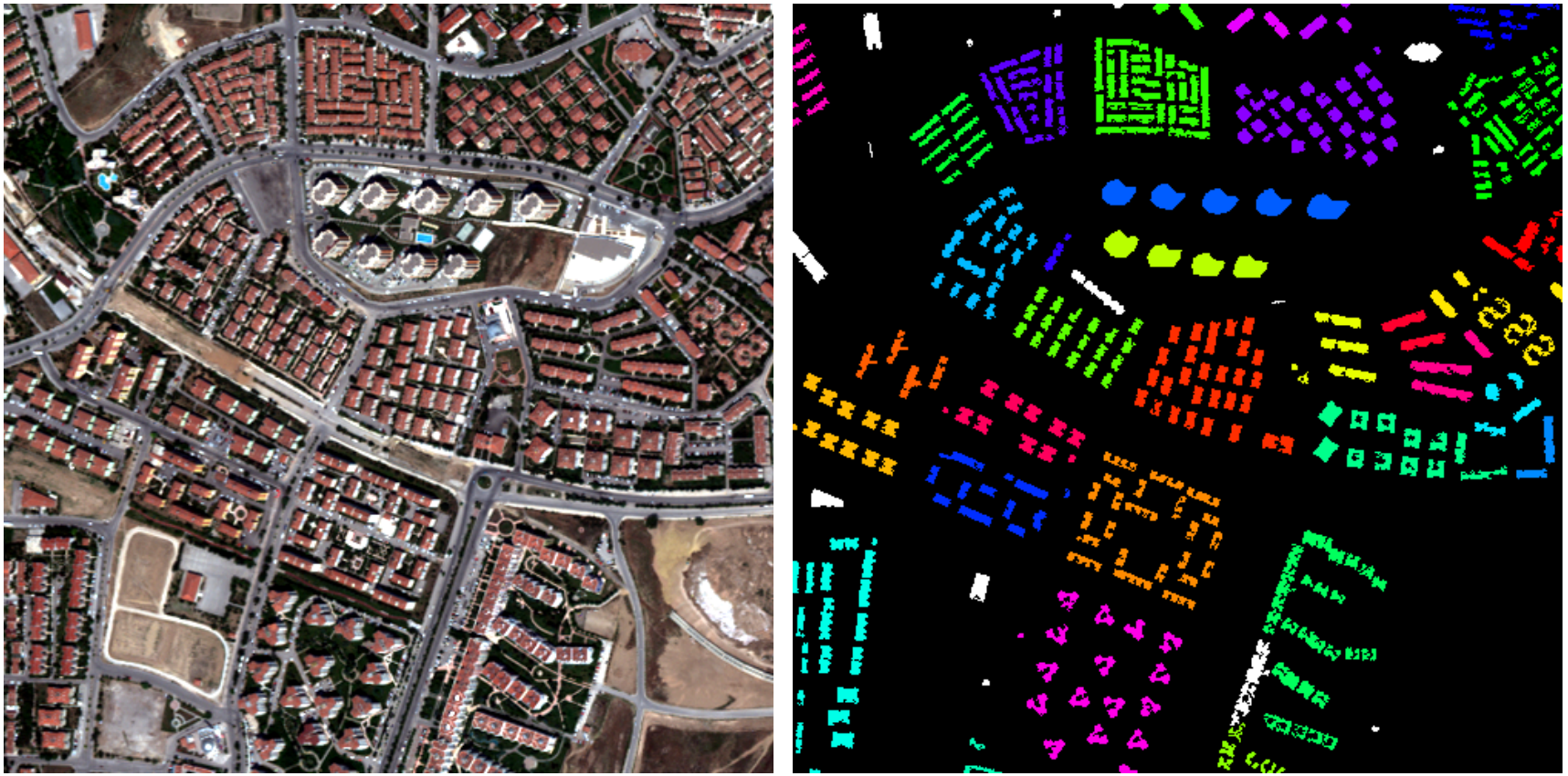
(a) Building mask (b) Alignment detection

Figure 57: Examples of alignment detection in an image with 418 buildings. All groups of buildings satisfying the alignment criteria are shown in (b).

Detection using hierarchical clustering

- After each vertex is assigned statistical and structural features, the next step is to group these objects via clustering.
- Once the statistical and structural distances are computed for each neighboring object pair, agglomerative hierarchical clustering iteratively groups these objects.

Detection using hierarchical clustering



(a) Ankara image (b) Combined clustering

Figure 58: Example clustering results on a multispectral WorldView-2 image of Ankara. Different groups are shown in different colors.

Detection using hierarchical clustering

- Each group is modeled using a Markov random field that uses area, eccentricity, and orientation of individual regions, and proximity and relative orientation of region pairs.
- Given a query region group, the distribution of its features is learned, and retrieval is performed by ranking other groups according to their probabilities under this distribution.

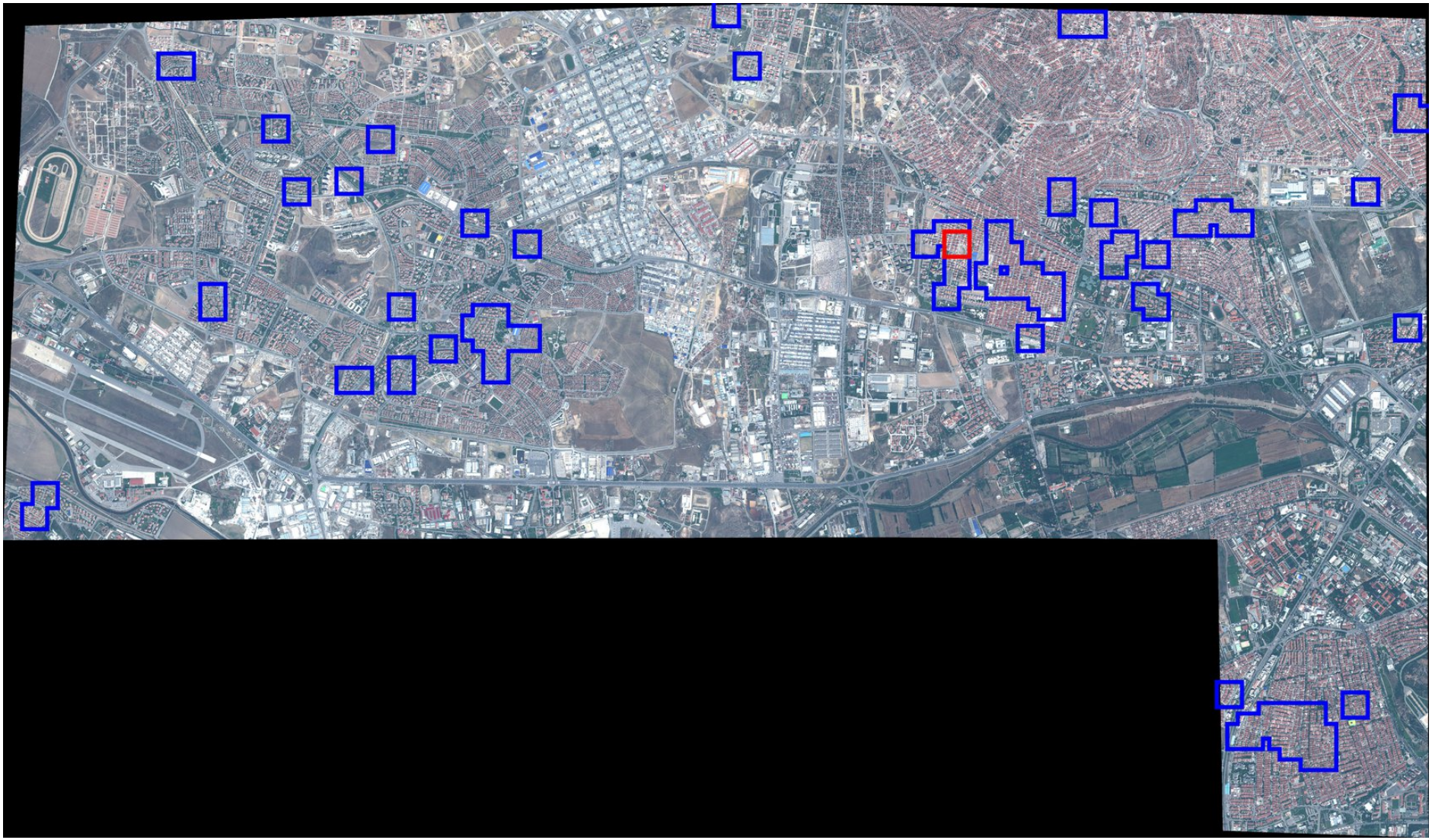


Figure 59: Example retrieval results on a WorldView-2 image of Ankara. The query window is shown as red and the top 100 results are shown as blue.

Figure 60: The query window and the top 15 results in the WorldView-2 Ankara image.

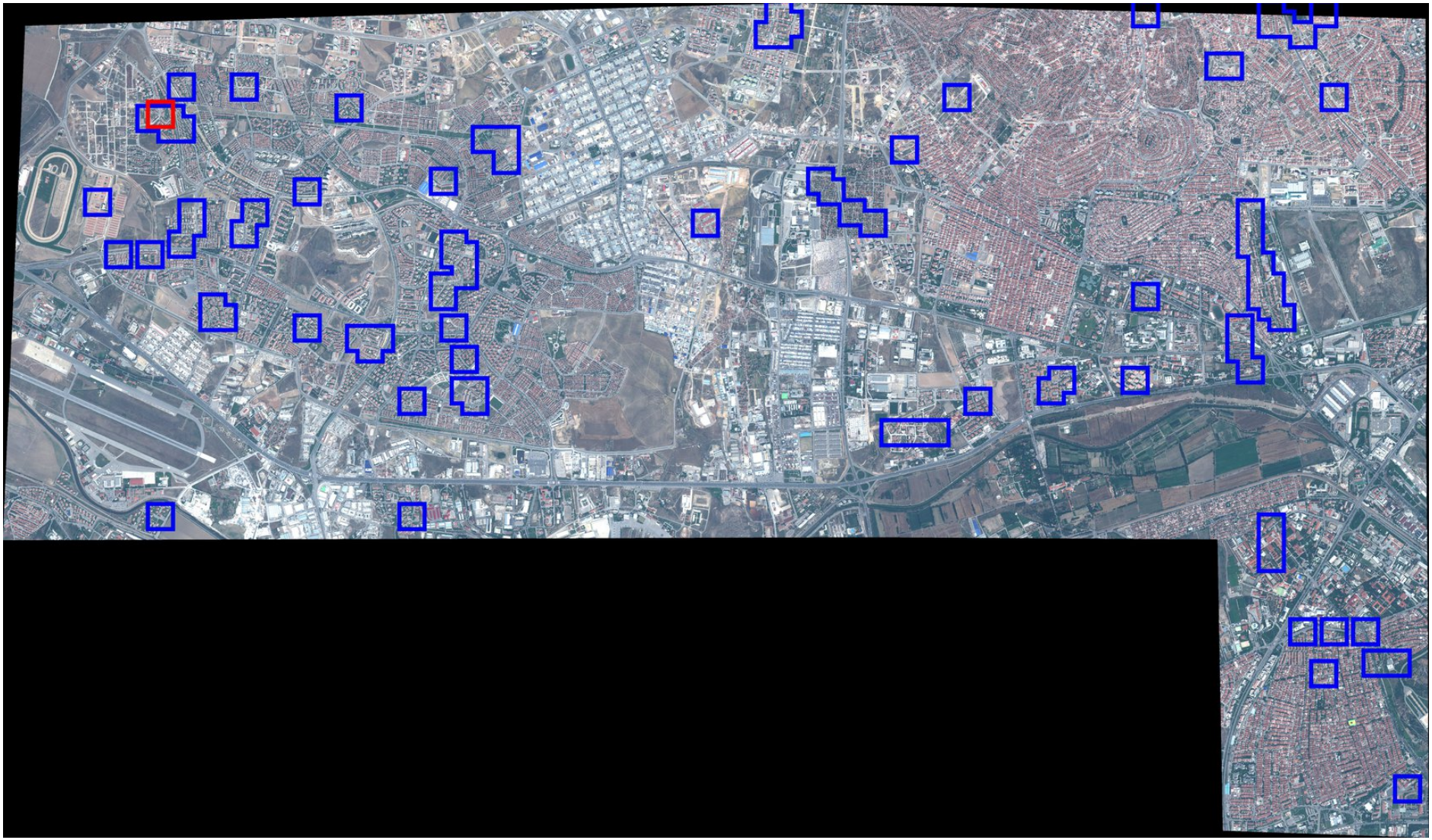


Figure 61: Example retrieval results on a WorldView-2 image of Ankara. The query window is shown as red and the top 100 results are shown as blue.

Figure 62: The query window and the top 15 results in the WorldView-2 Ankara image.

Part IV

Segmentation and Classification of Cervical Cell Images

Motivation

- Cervical cancer has a worldwide significant impact with nearly $500,000$ new cases and nearly $250,000$ deaths reported annually.
- However, cervical cancer is a preventable disease.
  - It develops during a long duration.
  - It can be treated if detected early.

Joint work with Aslı Gençtav, Bilkent University, and Sevgen Önder, Hacettepe University

Motivation

- Pap smear test is a manual screening procedure to detect cervical cancer by grading cervical cells based on the properties of their nuclei and cytoplasms.
- There is a possibility of inaccurate diagnosis because of human errors resulting from intra- and inter-observer variability.
- A computer-assisted screening system can be used for aiding the early diagnosis of cervical cancer.
- The key step of such a system is the accurate segmentation of cells along with their nuclei and cytoplasms.

Motivation

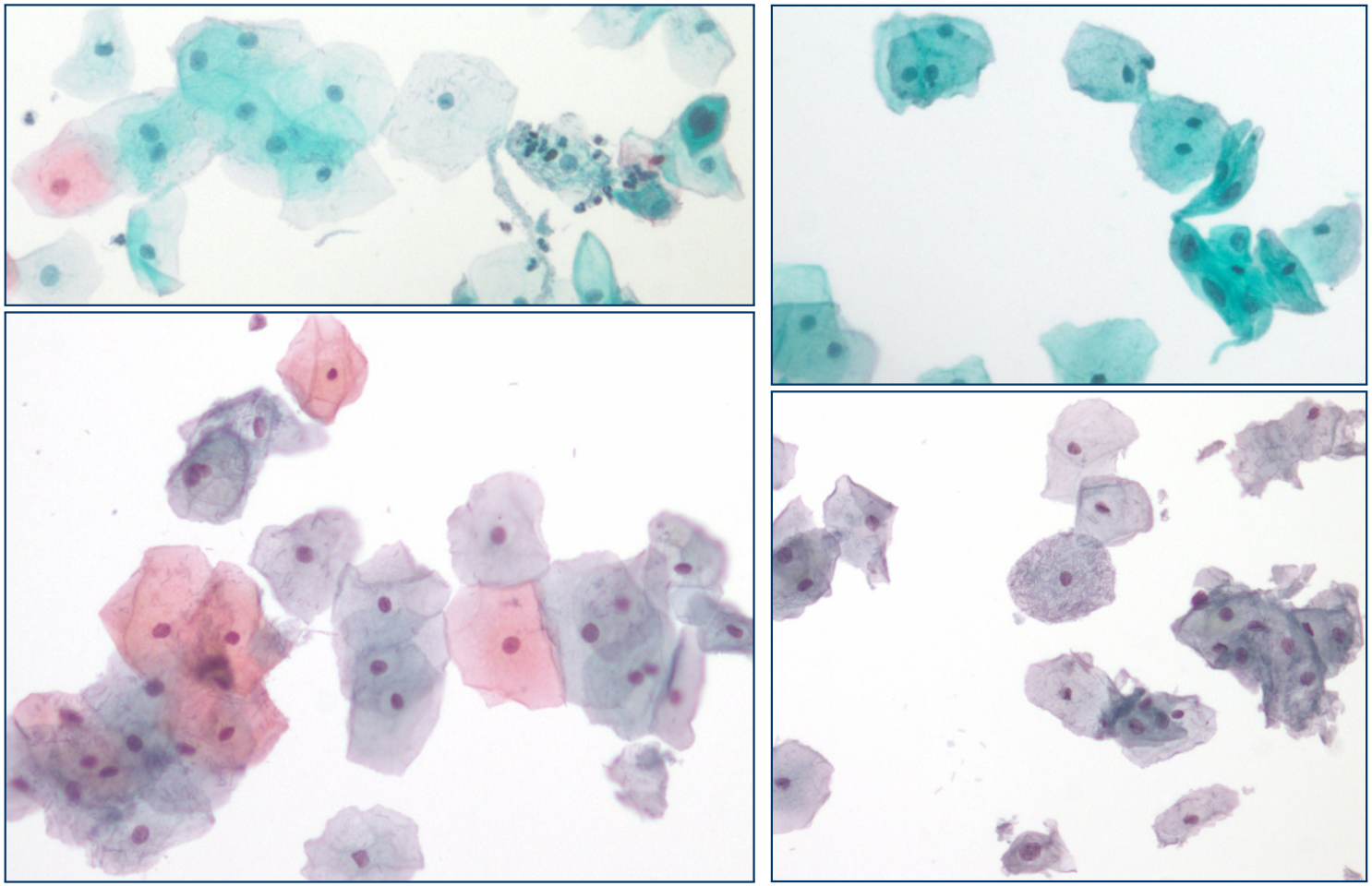


Figure 63: Example pap smear images involving multiple overlapping cells with inconsistent staining and poor contrast.

Overview

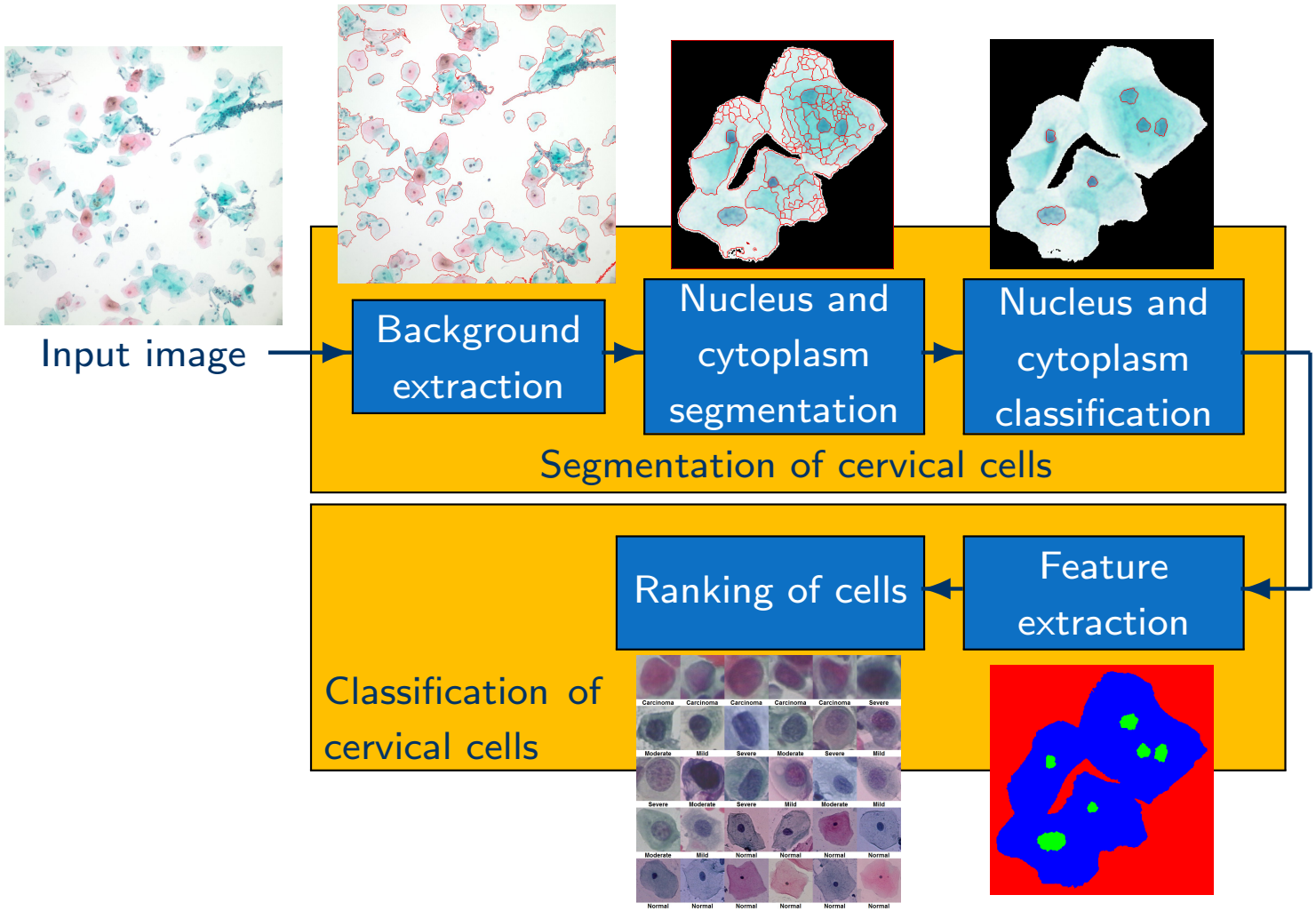


Figure 64: Overview of the approach.

Data sets

Herlev data

- This data set was developed by the Department of Pathology at Herlev University Hospital and the Department of Automation at Technical University of Denmark to provide benchmark data for comparing classification methods.
- The data set consists of 917 images of single cells.
- Cyto-technicians and doctors manually classified each cell into one of 7 classes.

Data sets

Herlev data

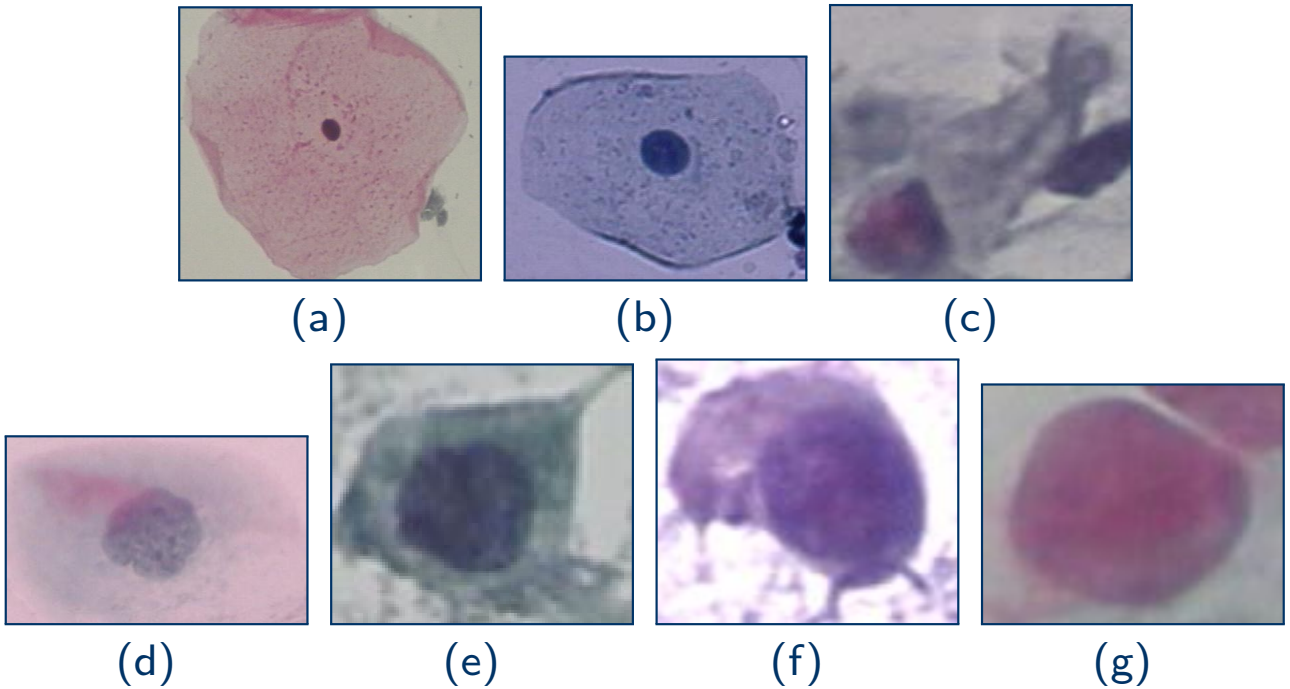


Figure 65: Examples from the Herlev data set. The cells belong to (a) superficial squamous, (b) intermediate squamous, (c) columnar, (d) mild dysplasia, (e) moderate dysplasia, (f) severe dysplasia, and (g) carcinoma in situ classes. The classes in the first row are considered to be normal and the ones in the second row are considered to be abnormal. Average image size is 156×140 pixels.

Data sets

Hacettepe data

- This data set was prepared by Dr. Sevgen Önder at the Department of Pathology, Hacettepe University Hospital in Ankara.
- The data was collected from the Pap test slides of 18 different patients.
- There are 82 images taken at $20\times$ magnification.
- The size of each image is 2048×2048 pixels.

Data sets

Hacettepe data

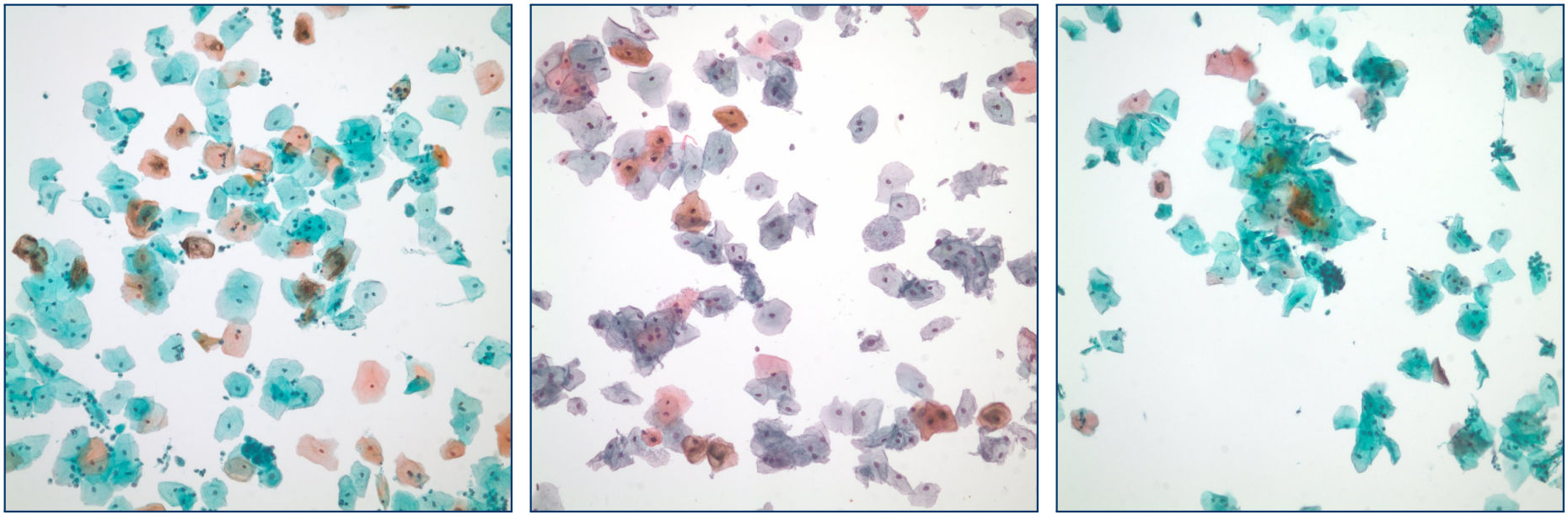


Figure 66: Example Pap smear test images from the Hacettepe data set.

Overview

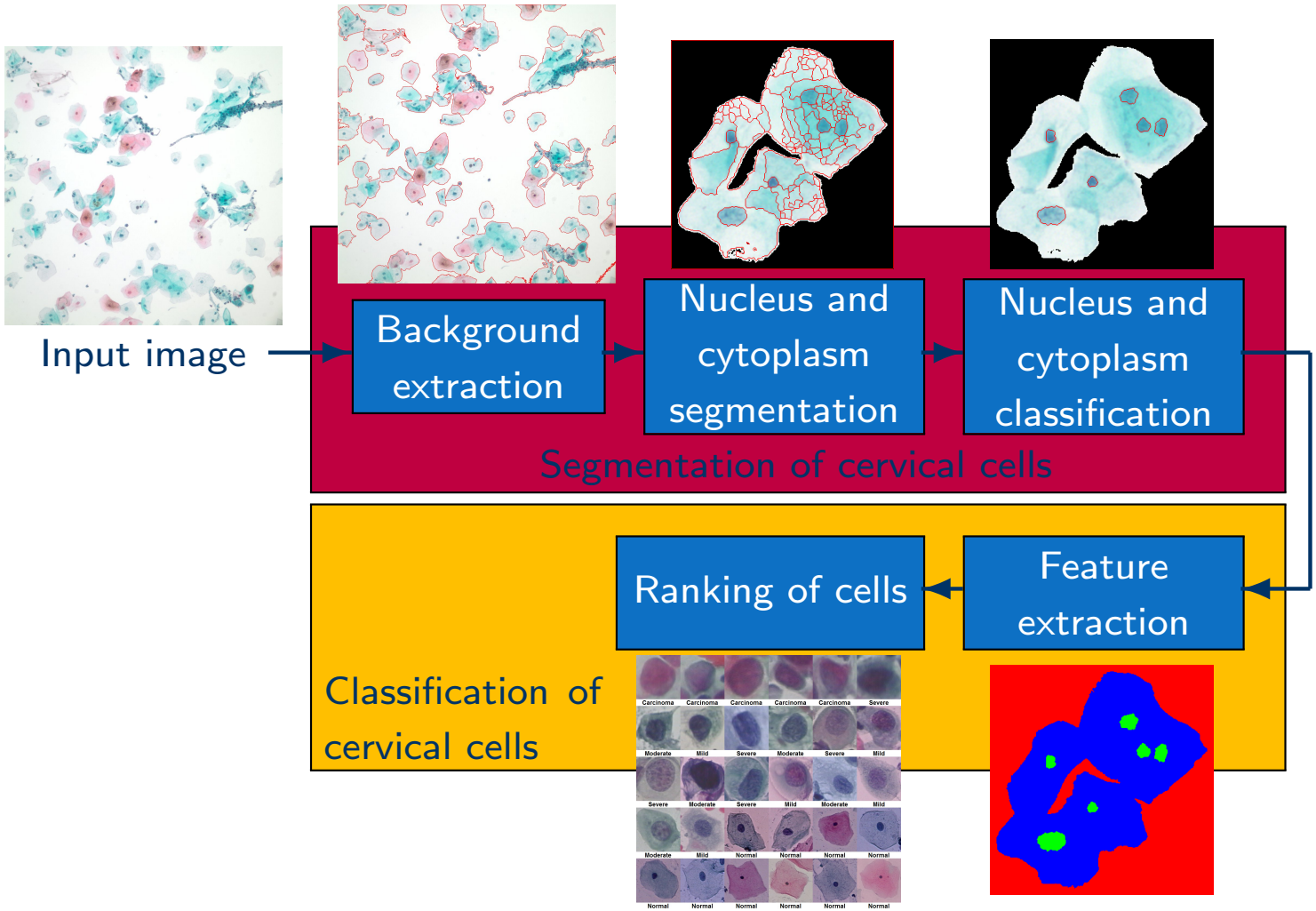


Figure 64: Overview of the approach.

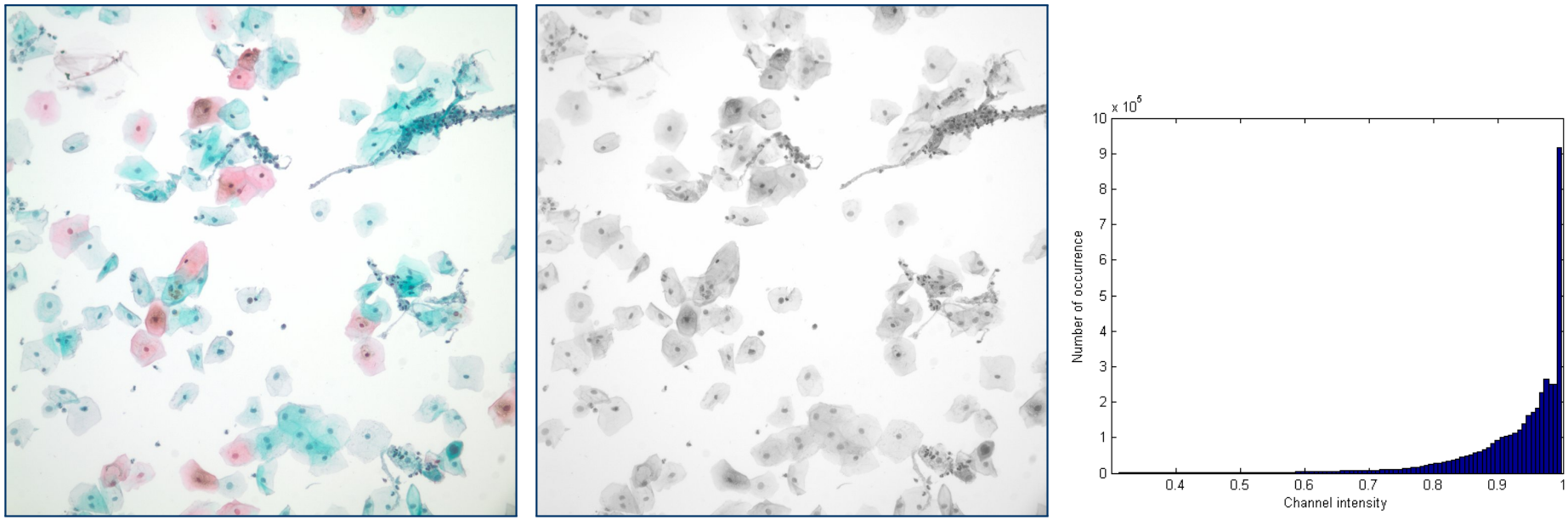
Segmentation of cervical cells

Background extraction

- The first step is to identify the cell regions in the image.
- This can be performed by removing the white areas by thresholding.
- However, choosing a threshold is difficult due to inhomogeneous illumination.
- We developed an automatic procedure using the minimum-error thresholding criterion.

Segmentation of cervical cells

Background extraction

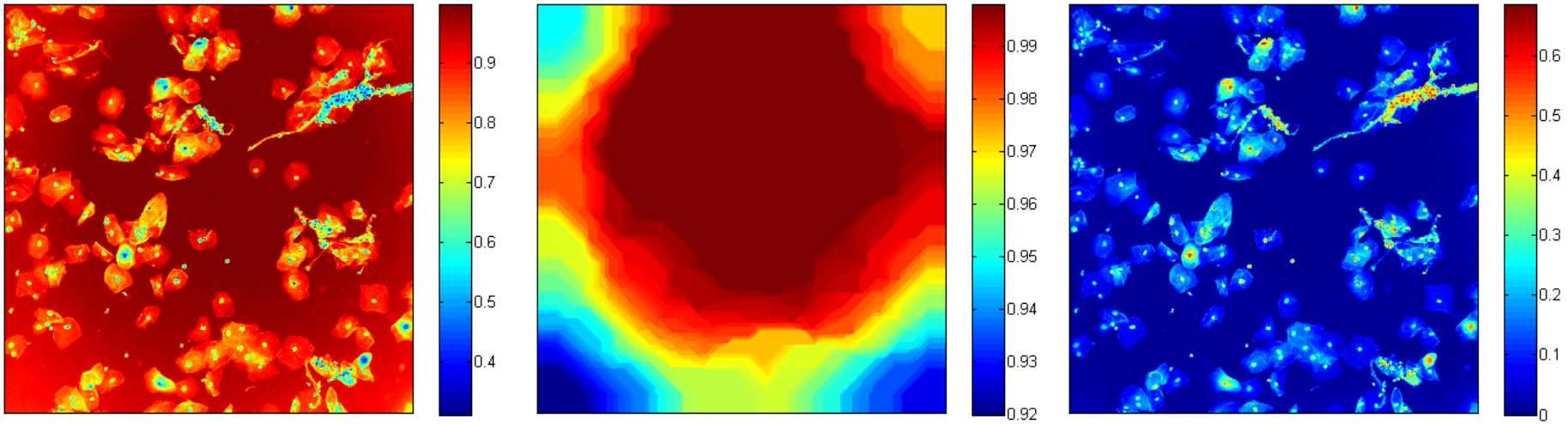


(a) (b) (c)

Figure 67: (a) Pap smear image in RGB color space. (b) L channel of the image in CIE Lab color space. (c) Histogram of the L channel.

Segmentation of cervical cells

Background extraction

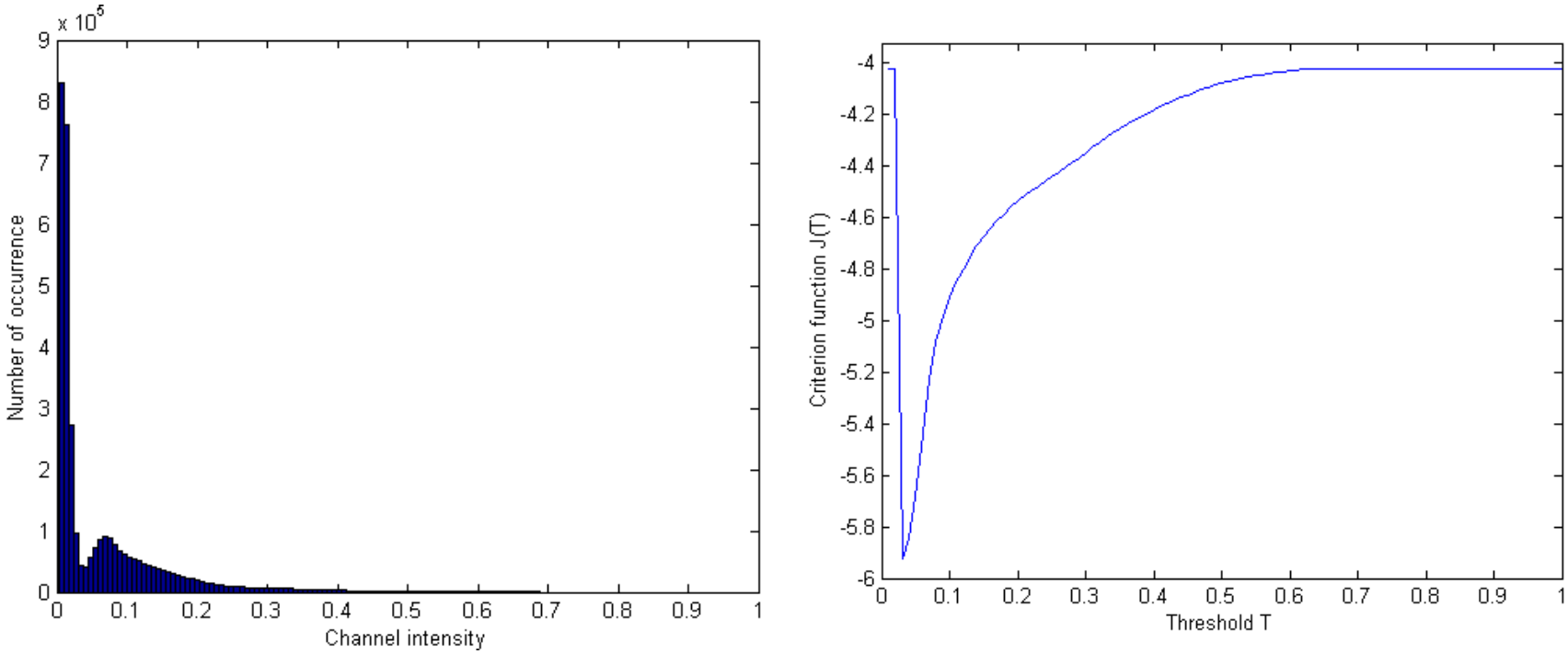


(a) (b) (c)

Figure 68: Black top-hat transform for eliminating inhomogeneous illumination. (a) L channel of the image. (b) Closing with a large structuring element. (c) Illumination-corrected L channel.

Segmentation of cervical cells

Background extraction

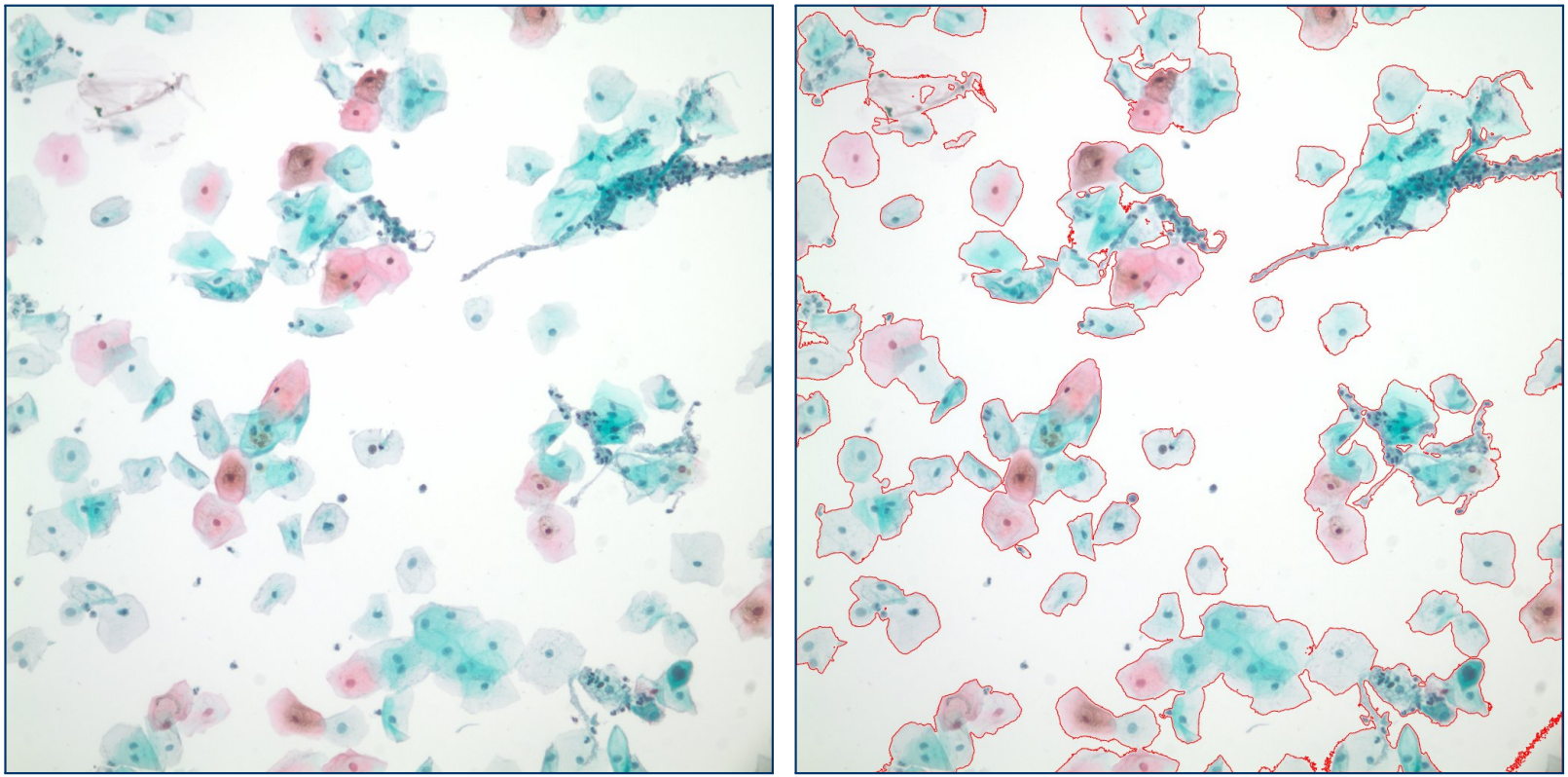


(a) (b)

Figure 69: (a) Histogram of the illumination-corrected L channel. (b) Criterion for automatic thresholding.

Segmentation of cervical cells

Background extraction



(a) (b)

Figure 70: (a) Pap smear image. (b) Regions found by thresholding at 0.03.

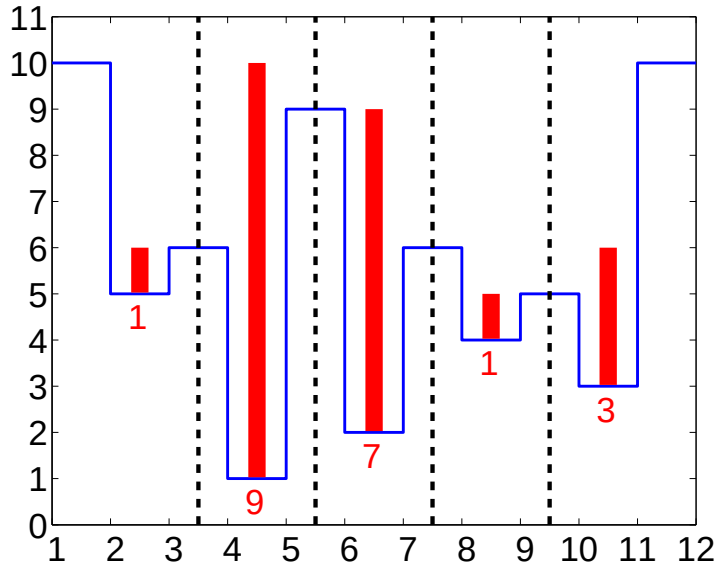
Segmentation of cervical cells

Nucleus and cytoplasm segmentation

- The next step is to obtain an accurate separation of nuclei from cytoplasm within the cell regions.
- There are important constraints such as inconsistent staining, unknown number of nuclei, and overlapping cells.
- Therefore, we focused on the segmentation of nuclei from the rest of the image by applying multi-scale watershed segmentation based on dynamics.

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

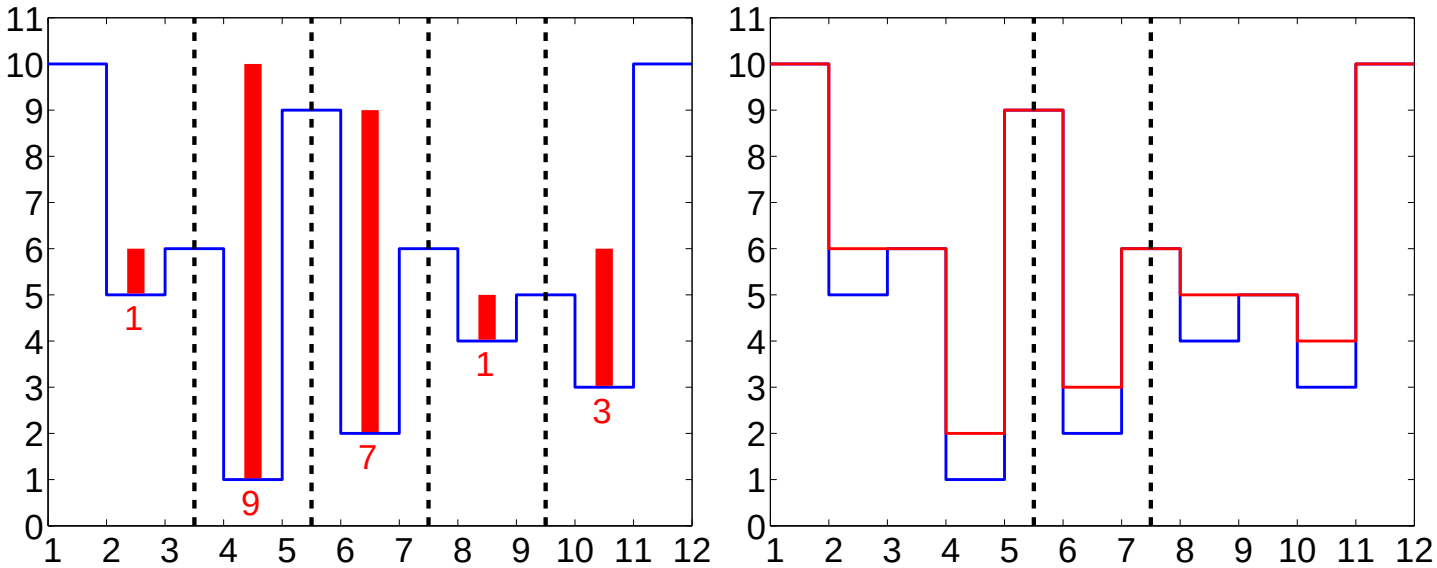


(a) Scale 0

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

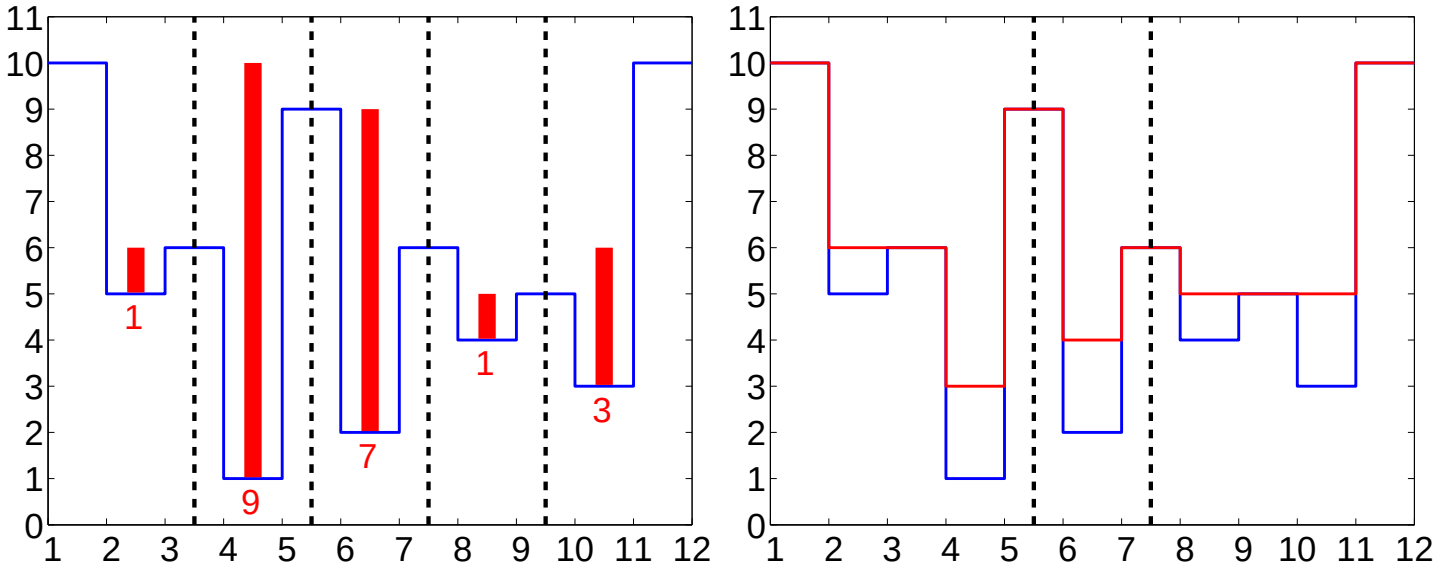


(a) Scale 0 (b) Scale 1

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

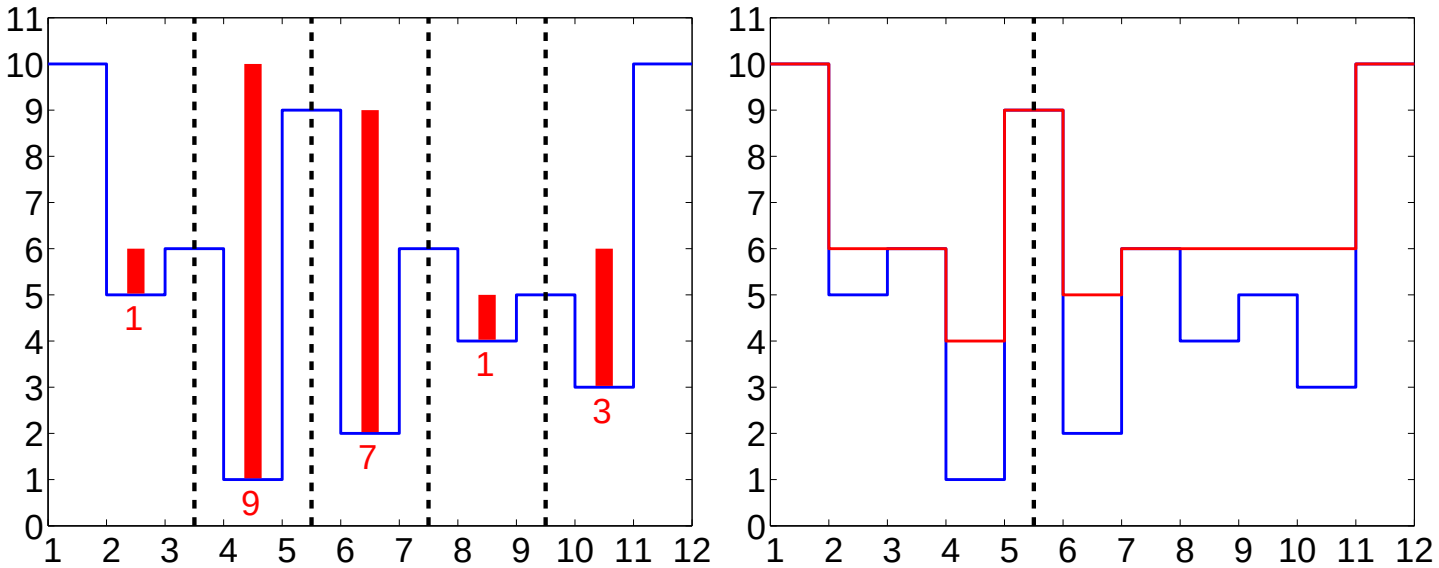


(a) Scale 0 (b) Scale 2

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

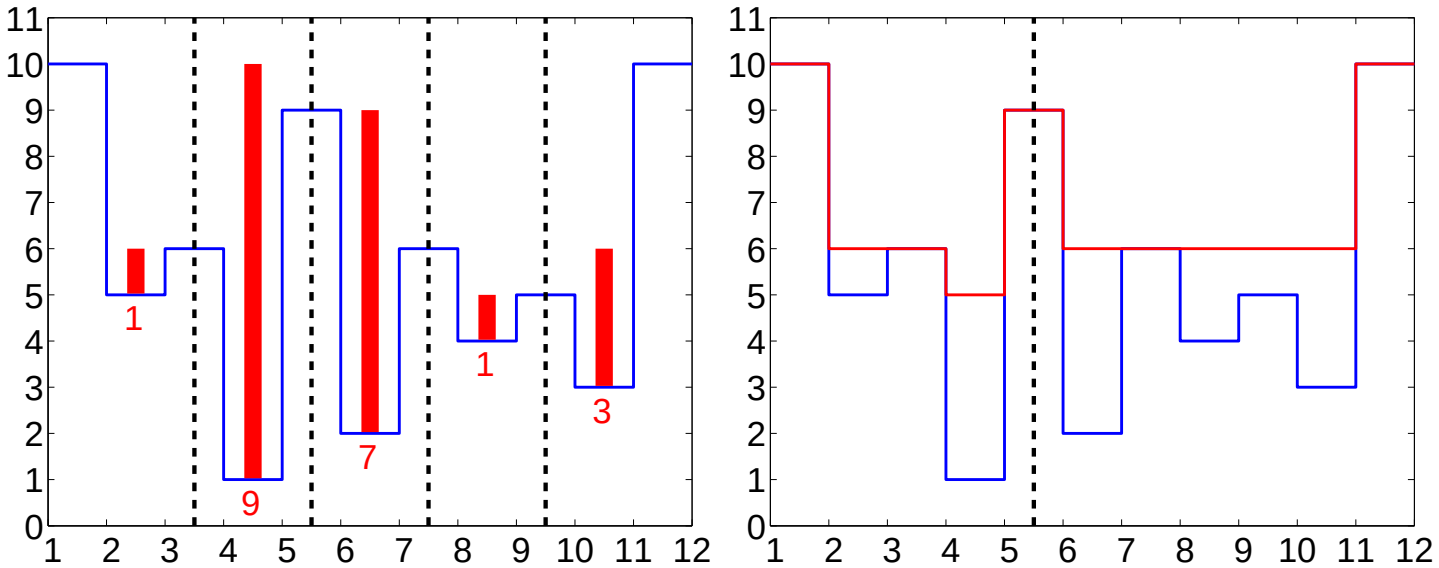


(a) Scale 0 (b) Scale 3

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

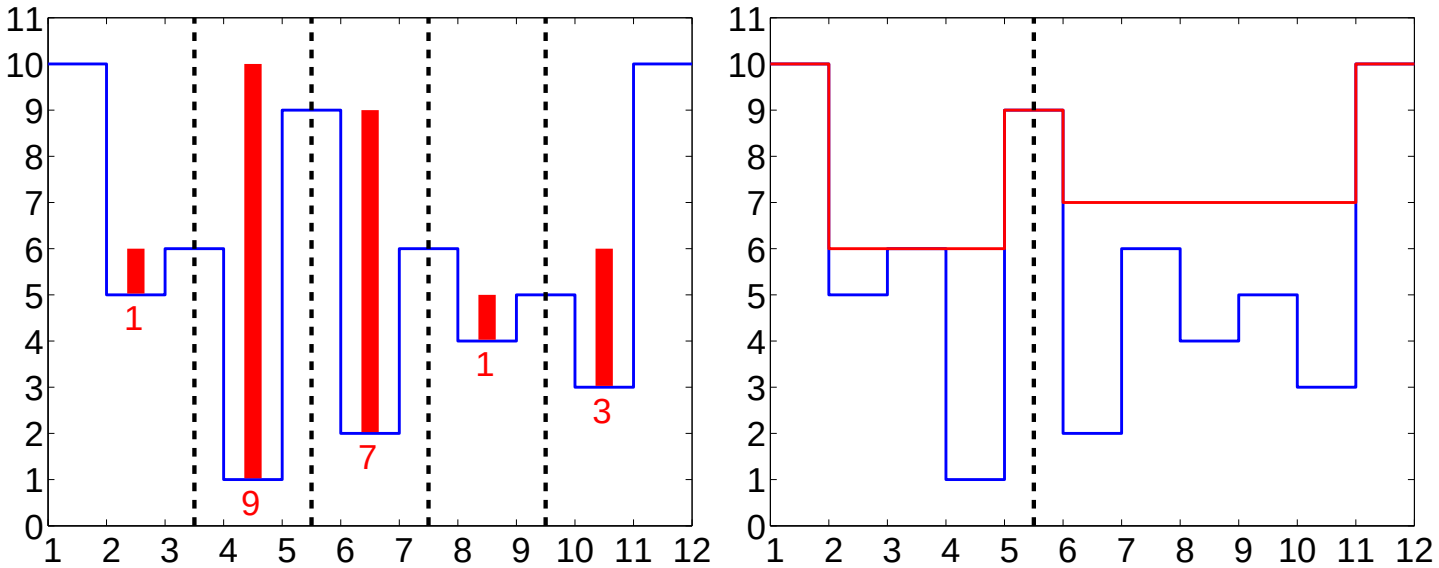


(a) Scale 0 (b) Scale 4

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

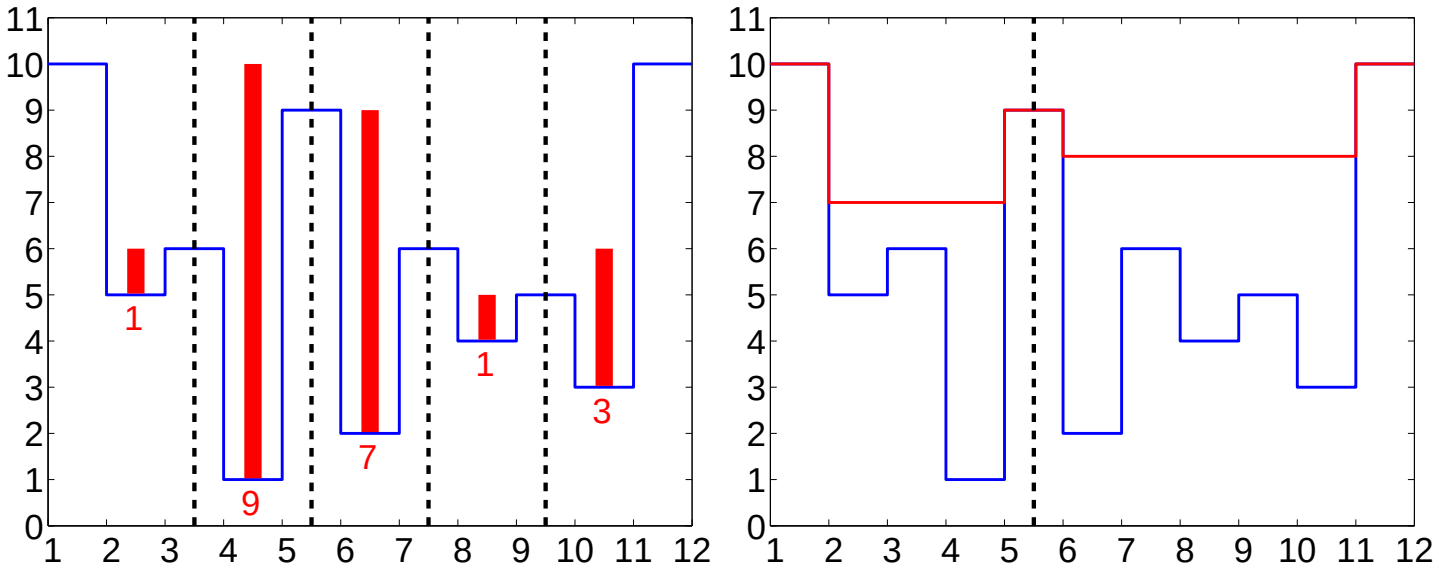


(a) Scale 0 (b) Scale 5

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

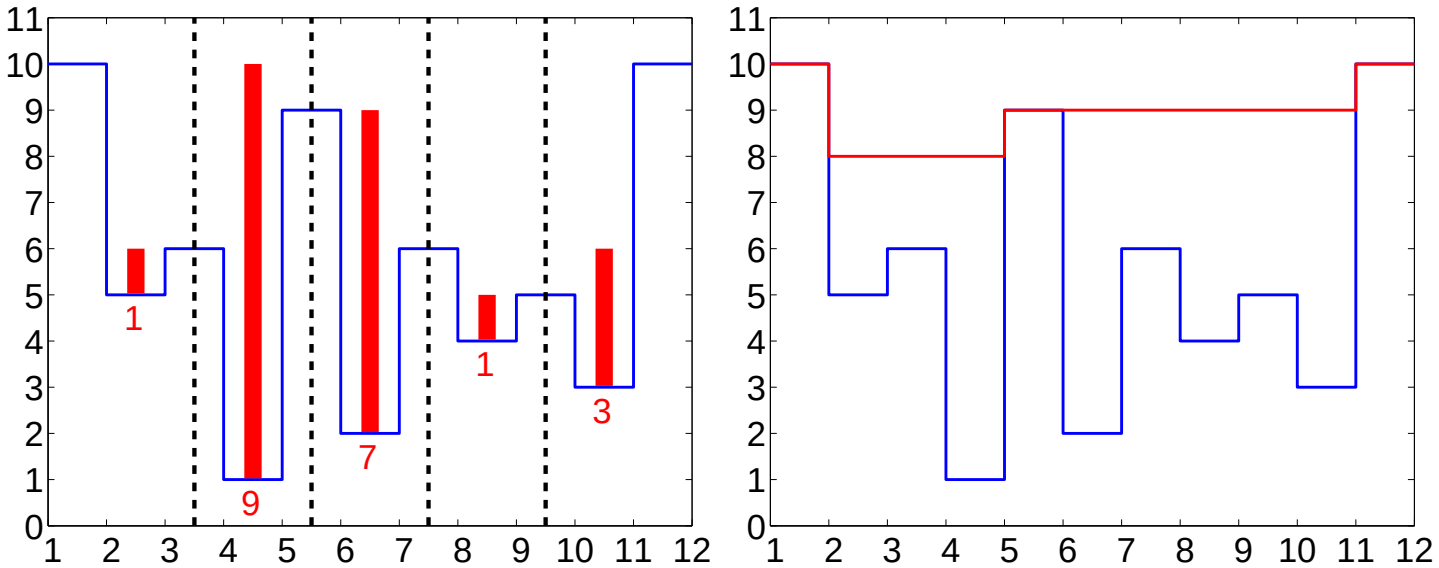


(a) Scale 0 (b) Scale 6

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

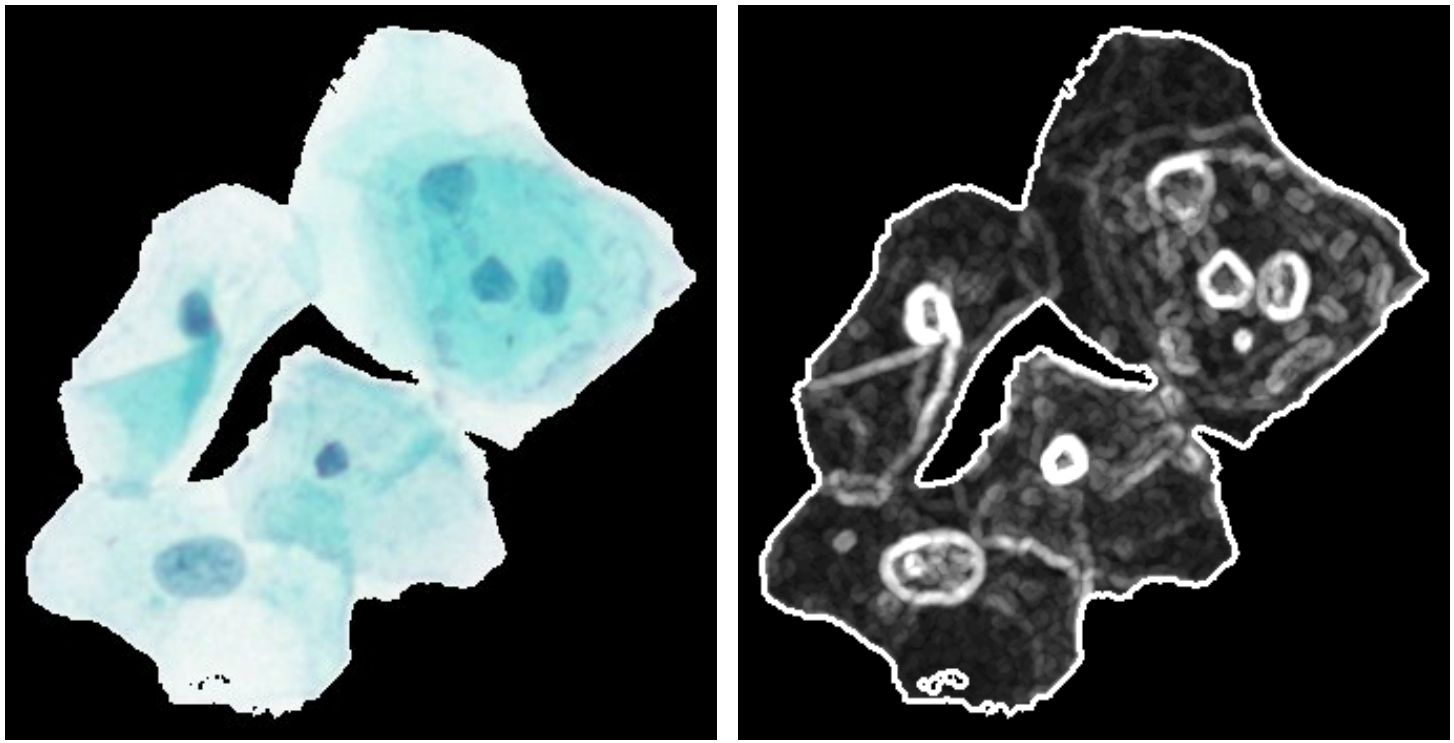


(a) Scale 0 (b) Scale 7

Figure 71: One-dimensional synthetic signal (blue) and watersheds (black) after iterative elimination of local minima (red). The dynamic of each initial regional minimum is also shown as red bars in (a). In all figures, the y-axis simulates the signal values (e.g., image gradient) and the x-axis simulates the domain of the signal (e.g., pixel locations).

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

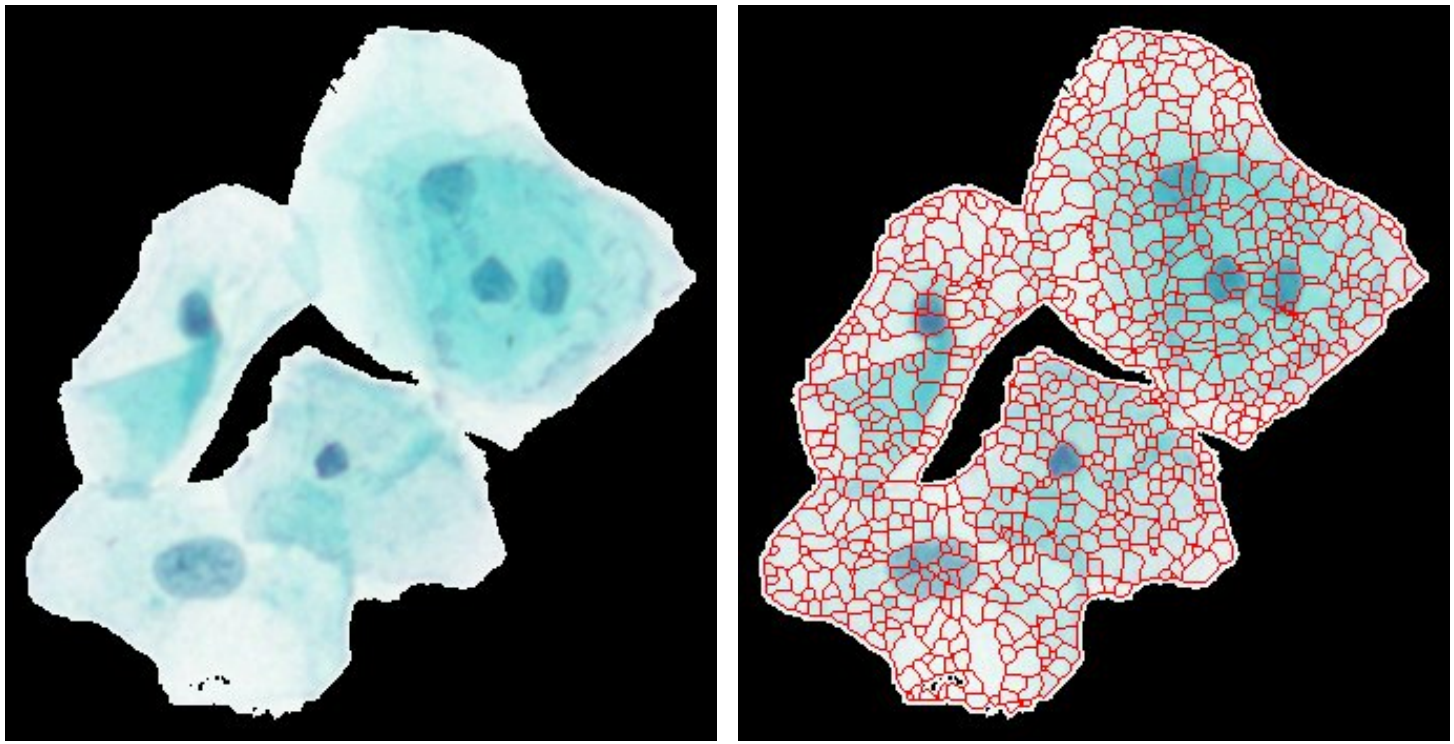


(a) (b)

Figure 72: (a) Cell image. (b) Gradient of the L channel that is used for computing the dynamics.

Segmentation of cervical cells

Nucleus and cytoplasm segmentation



(a) (b)

Figure 73: Candidate segments obtained by multi-scale watershed segmentation.

Segmentation of cervical cells

Nucleus and cytoplasm segmentation

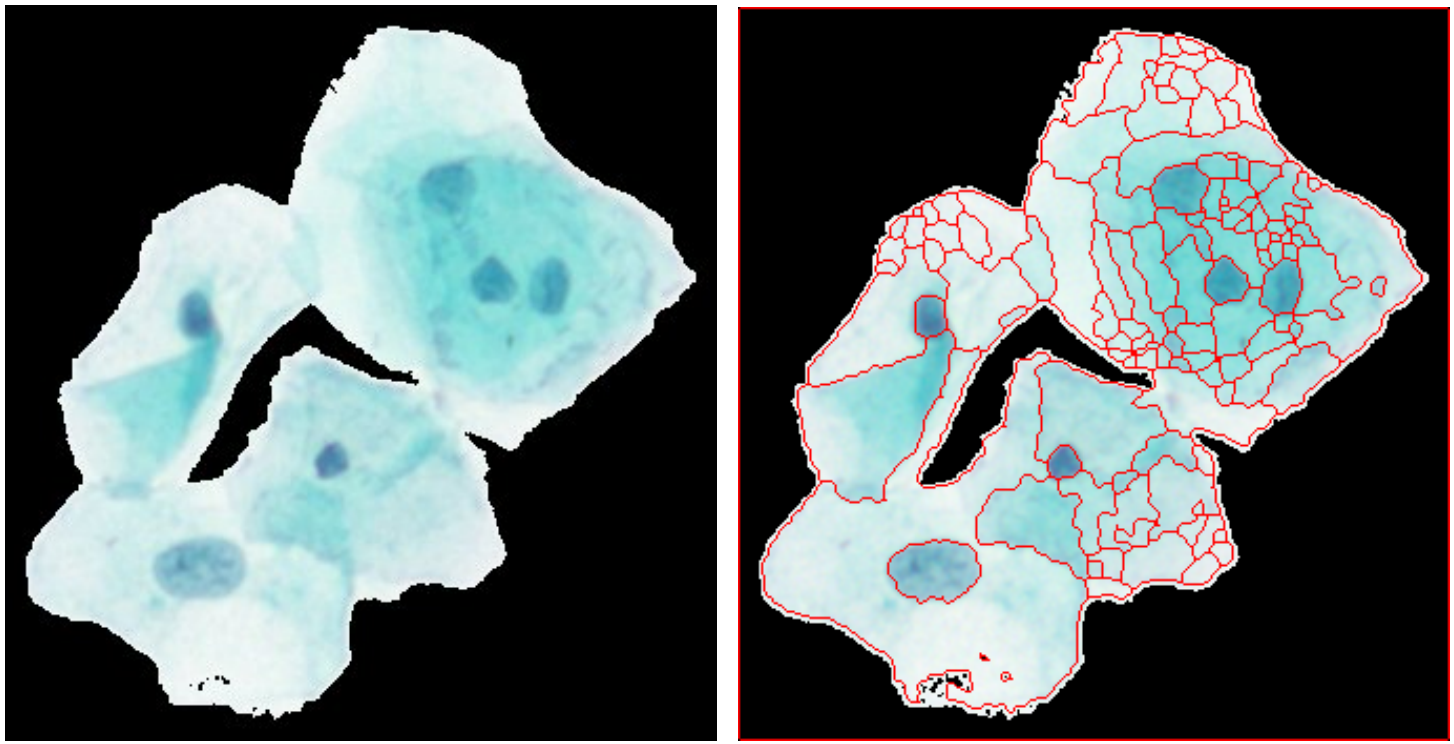
- The goal is to select the nuclei among all regions appearing at different levels of the hierarchy.
- We define a goodness measure for each node as

$$homogeneity \times circularity.$$

- *Homogeneity*: spectral similarity between a node and its parent node measured using an F-statistic based on mean and variance.
- *Circularity*: inverse of the eccentricity of the node.
- The nodes that optimize this measure are selected using a two-pass algorithm on the hierarchical tree.

Segmentation of cervical cells

Nucleus and cytoplasm segmentation



(a) (b)

Figure 74: Segmentation result of the example cell image.

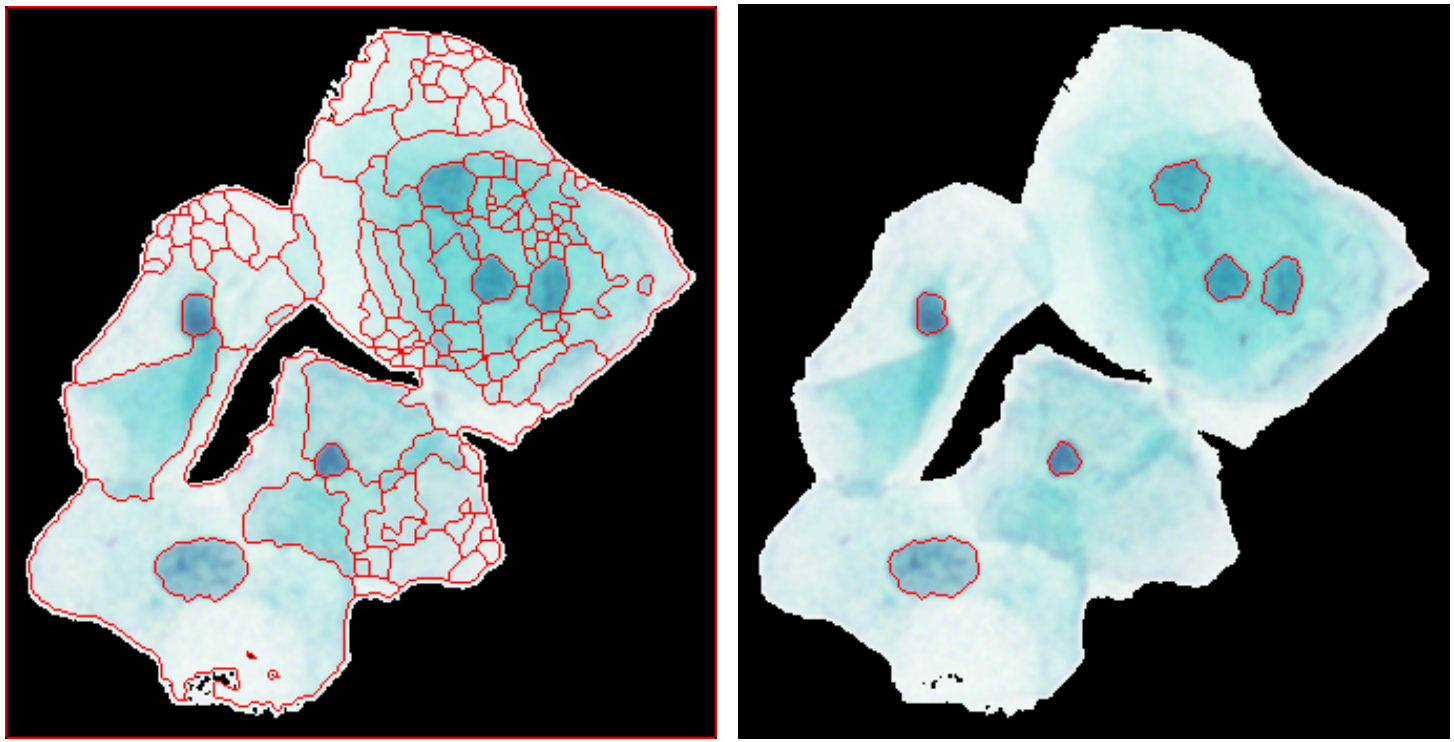
Segmentation of cervical cells

Nucleus and cytoplasm classification

- Finally, we classify the segments as nucleus or cytoplasm based on their size, mean intensity, circularity, and homogeneity attributes.
- The classification is performed using a combination of a Bayesian classifier, a decision tree, and an SVM classifier.
- The final cytoplasm region is calculated as the union of all segments classified as cytoplasm.

Segmentation of cervical cells

Nucleus and cytoplasm classification



(a) (b)

Figure 75: (a) Segmentation result. (b) Classification of the example cell image into nuclei and cytoplasm regions.

Segmentation of cervical cells

Experiments

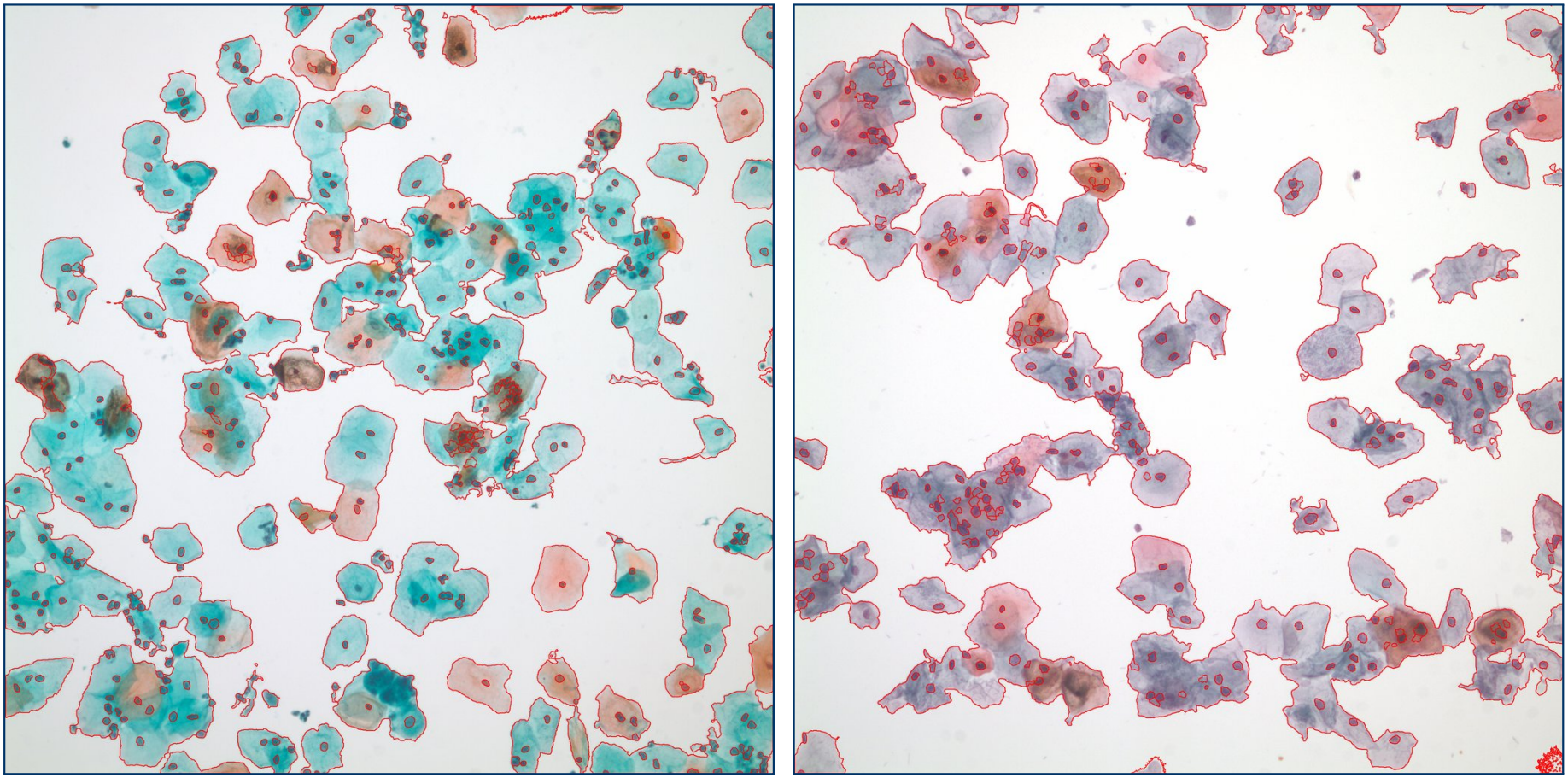


Figure 76: Example segmentation results for the Hacettepe data.

Segmentation of cervical cells

Experiments

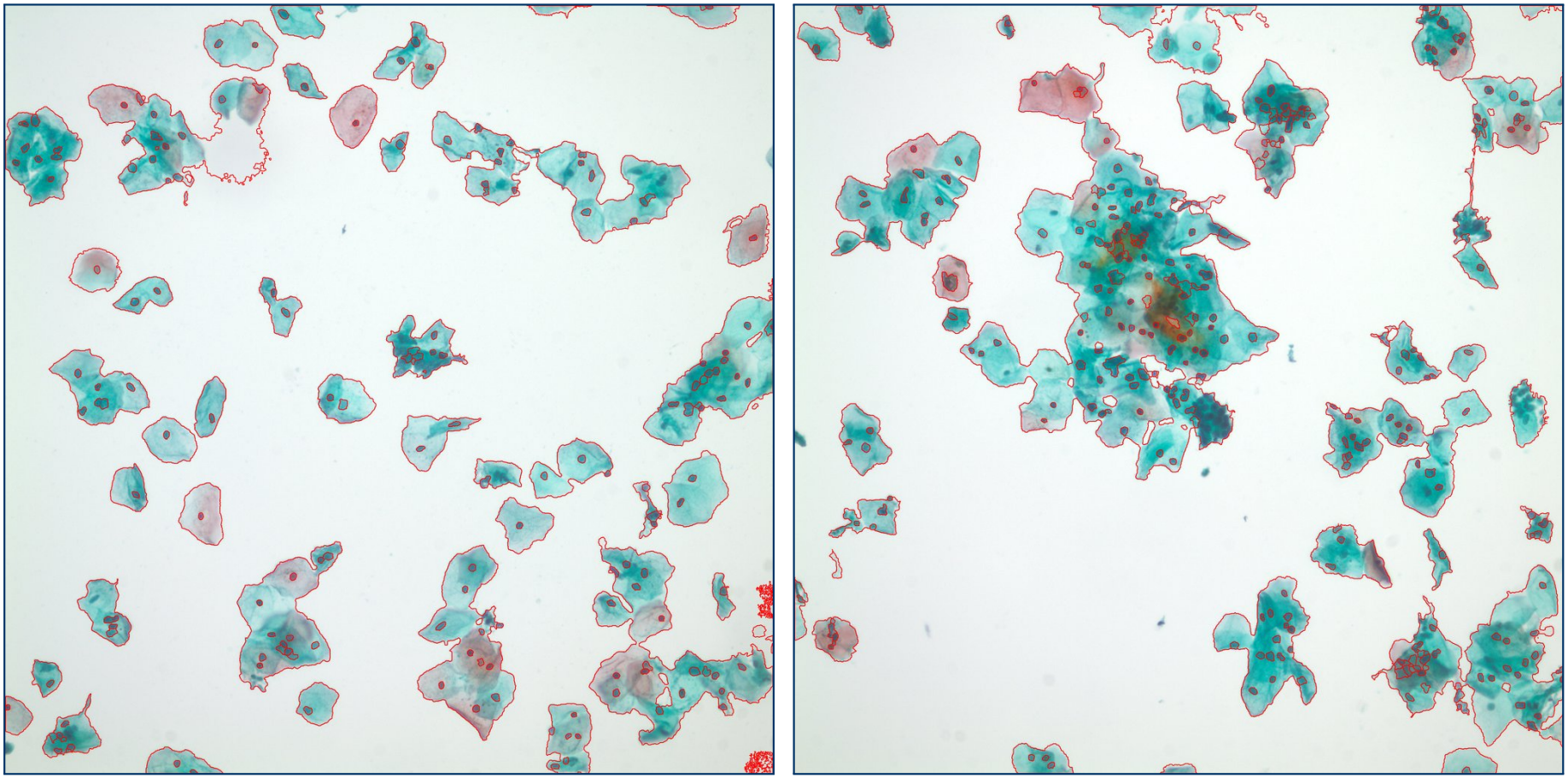


Figure 77: Example segmentation results for the Hacettepe data.

Segmentation of cervical cells

Experiments

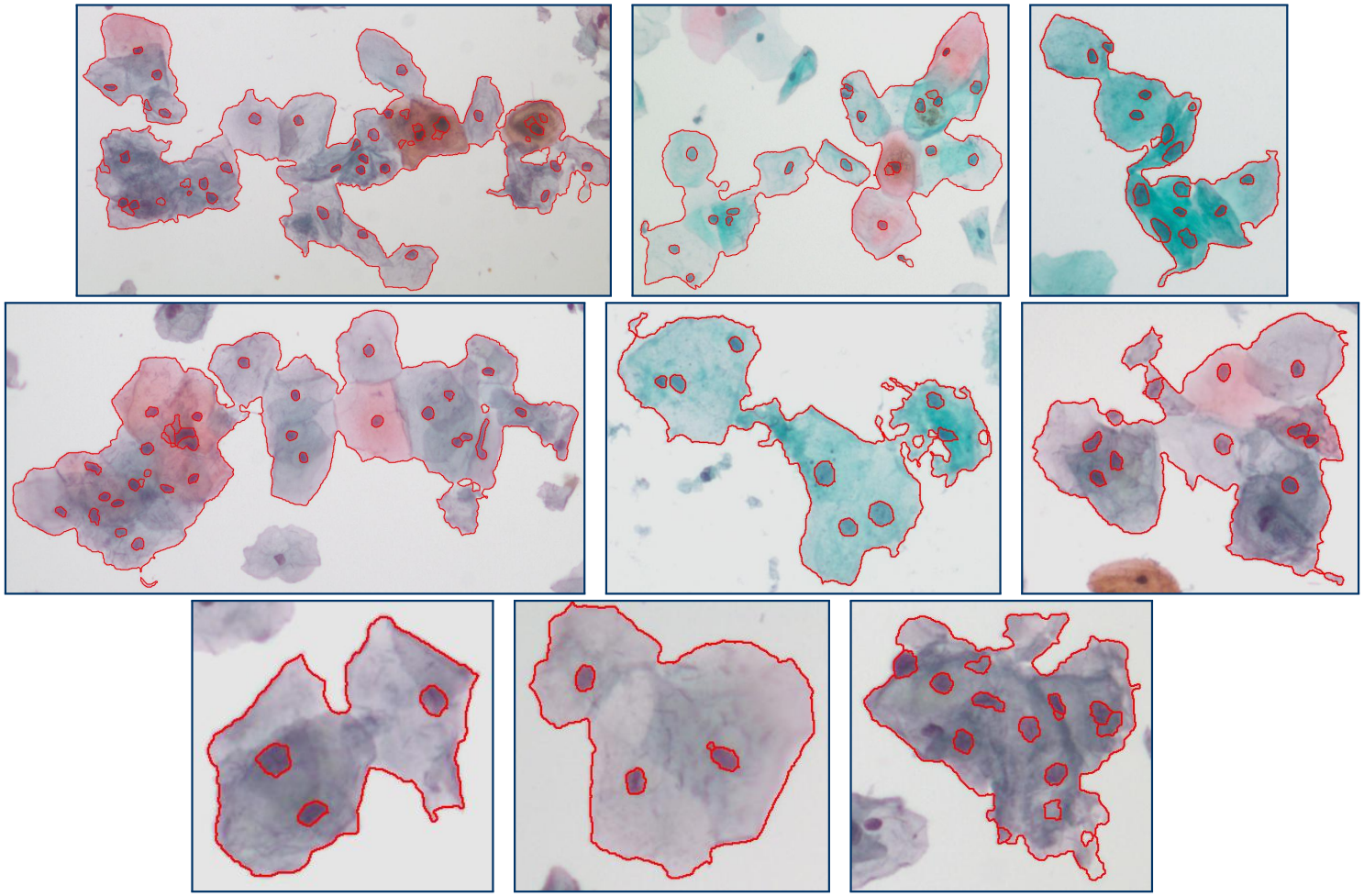


Figure 78: Example segmentation results for the Hacettepe data.

Overview

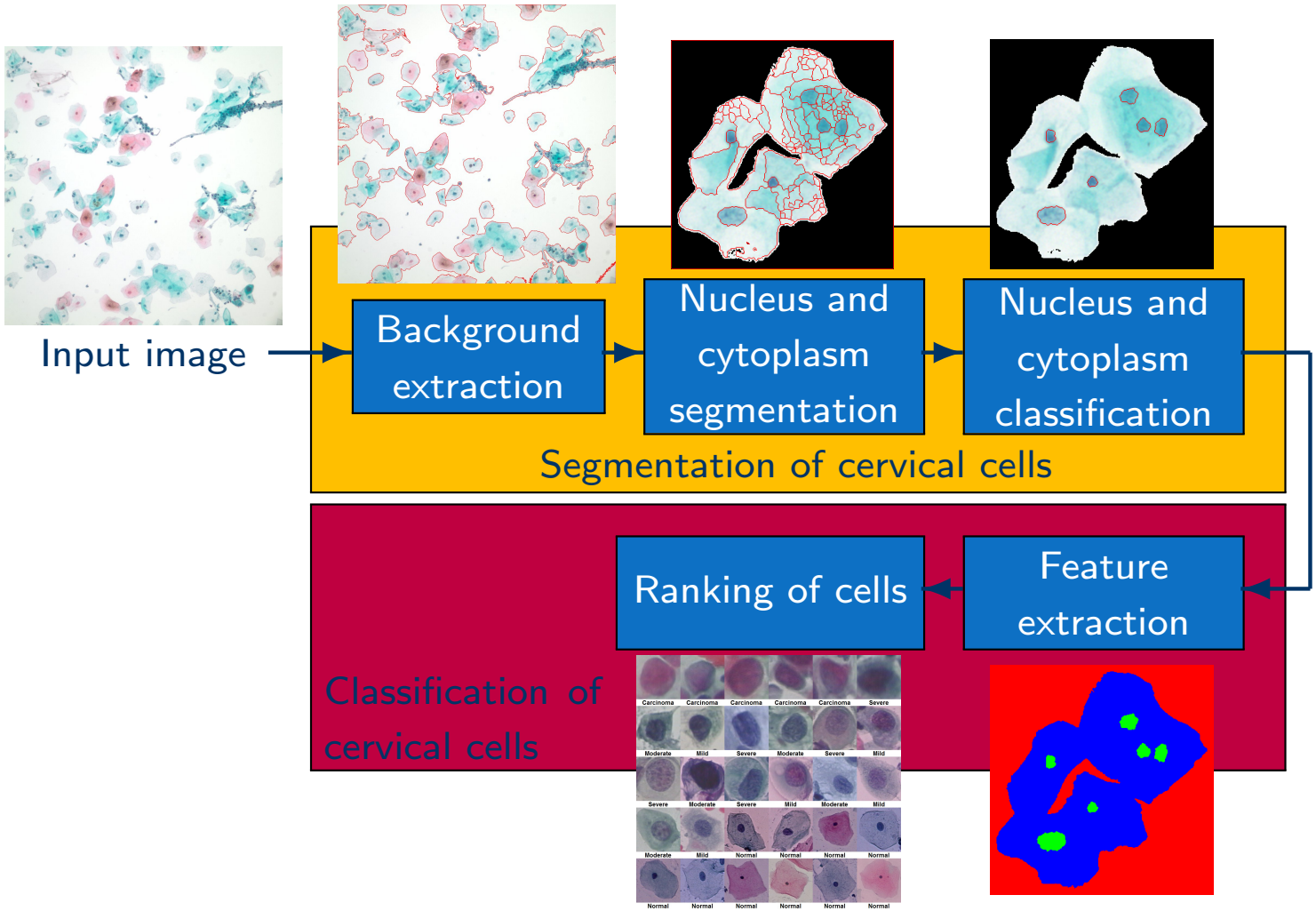


Figure 64: Overview of the approach.

Classification of cervical cells

- The classification step aims to obtain an unsupervised ordering of the cells in a Pap smear image according to their abnormality degree.
  - The ordered cells can be used for fast browsing of the image content.
  - Sufficiently large and representative training data may not be available for supervised classification.
- We rank the nuclei in the image by a linear ordering of the leaves of the dendrogram resulting from hierarchical clustering of cell features.
- Then, this ranking is improved using an optimal leaf ordering algorithm.

Classification of cervical cells

Feature extraction

1. Nucleus area
2. Nucleus brightness
3. Nucleus longest diameter
4. Nucleus shortest diameter
5. Nucleus elongation
6. Nucleus roundness
7. Nucleus perimeter
8. Nucleus maxima
9. Nucleus minima
10. Cytoplasm area
11. Nucleus/cytoplasm ratio
12. Cytoplasm brightness
13. Cytoplasm maxima
14. Cytoplasm minima

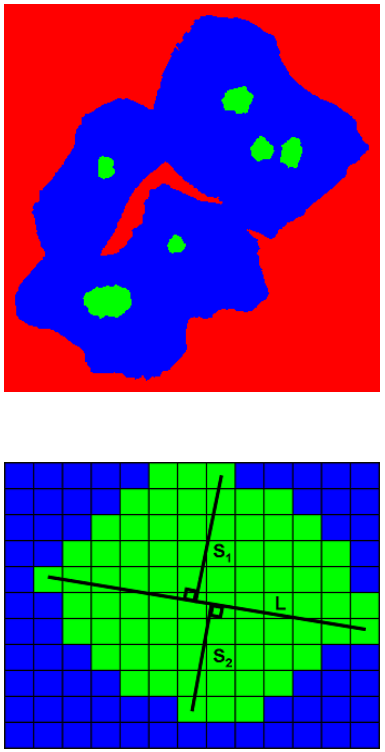


Figure 79: Example nuclei (green) surrounded by cytoplasm (blue).

Classification of cervical cells

Ranking of cervical cells

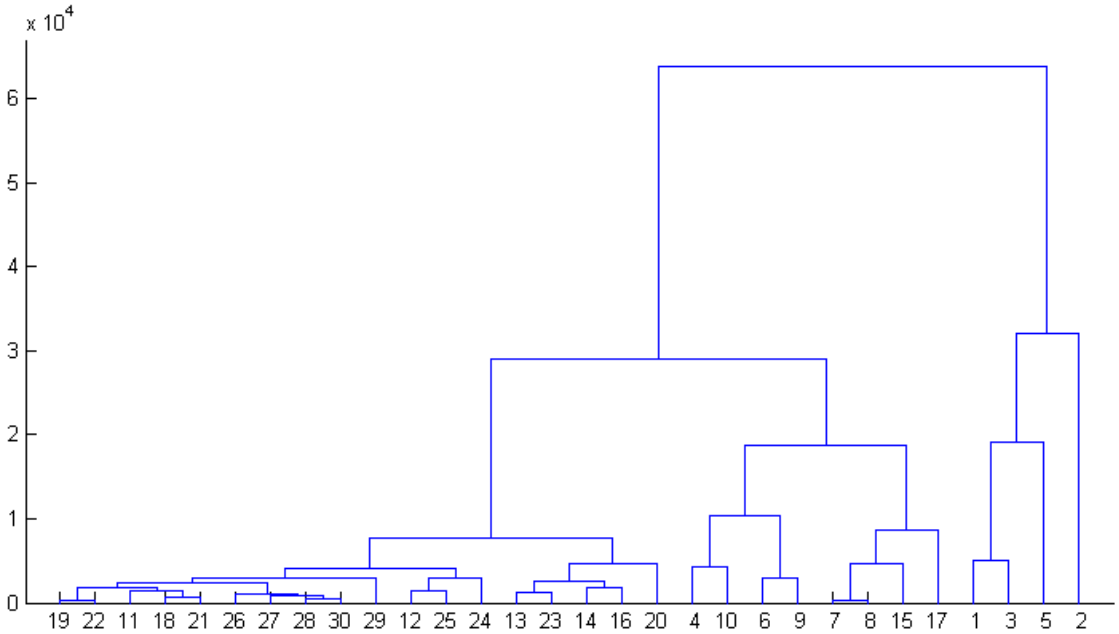


Figure 80: The binary tree resulting from hierarchical clustering of 30 cells randomly selected from the Herlev data (normal superficial $(1-5)$, normal intermediate $(6-10)$, mild dysplasia $(11-15)$, moderate dysplasia $(16-20)$, severe dysplasia $(21-25)$, carcinoma in situ $(26-30)$).

Classification of cervical cells

Ranking of cervical cells

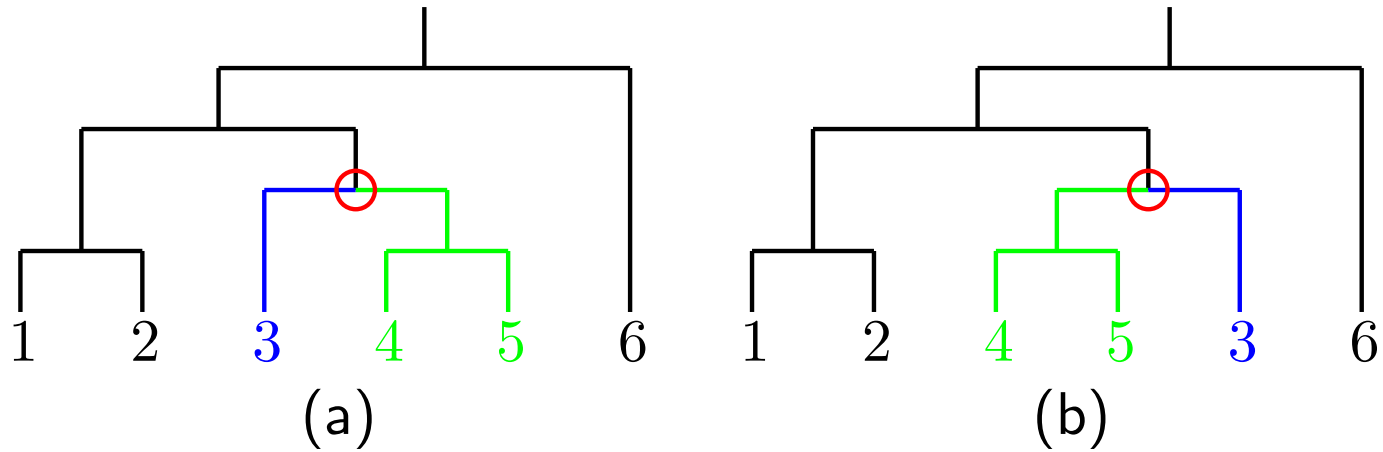
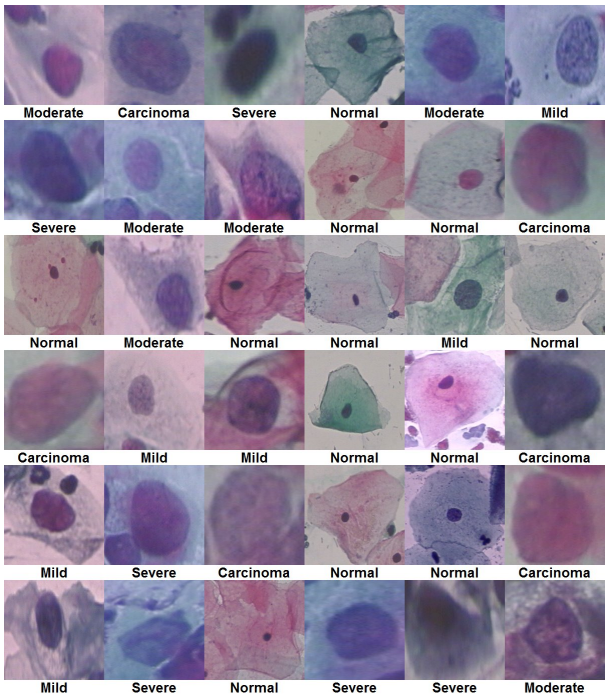


Figure 81: (a) An example binary tree T. (b) A linear leaf ordering consistent with T obtained by flipping the node marked by the red circle. An optimal leaf ordering algorithm can be used for finding the flips required for maximizing the sum of similarities between adjacent leaves.

Classification of cervical cells

Ranking of cervical cells

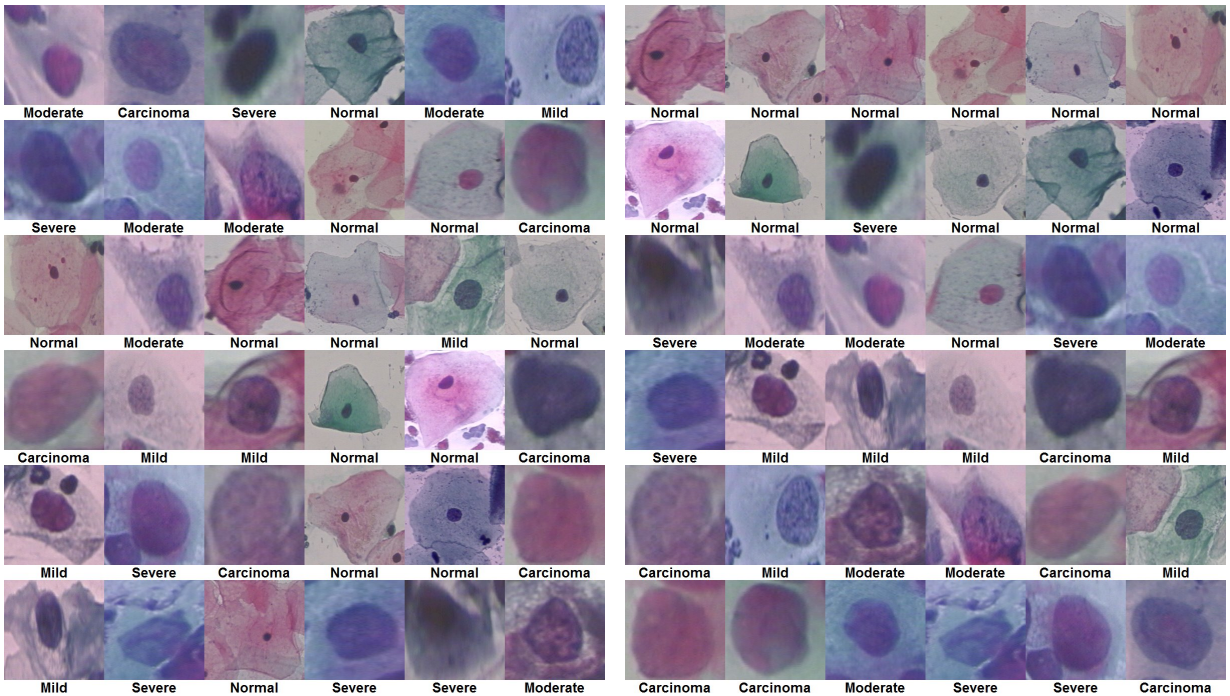


(a)

Figure 82: (a) Random ordering of 30 cells.

Classification of cervical cells

Ranking of cervical cells

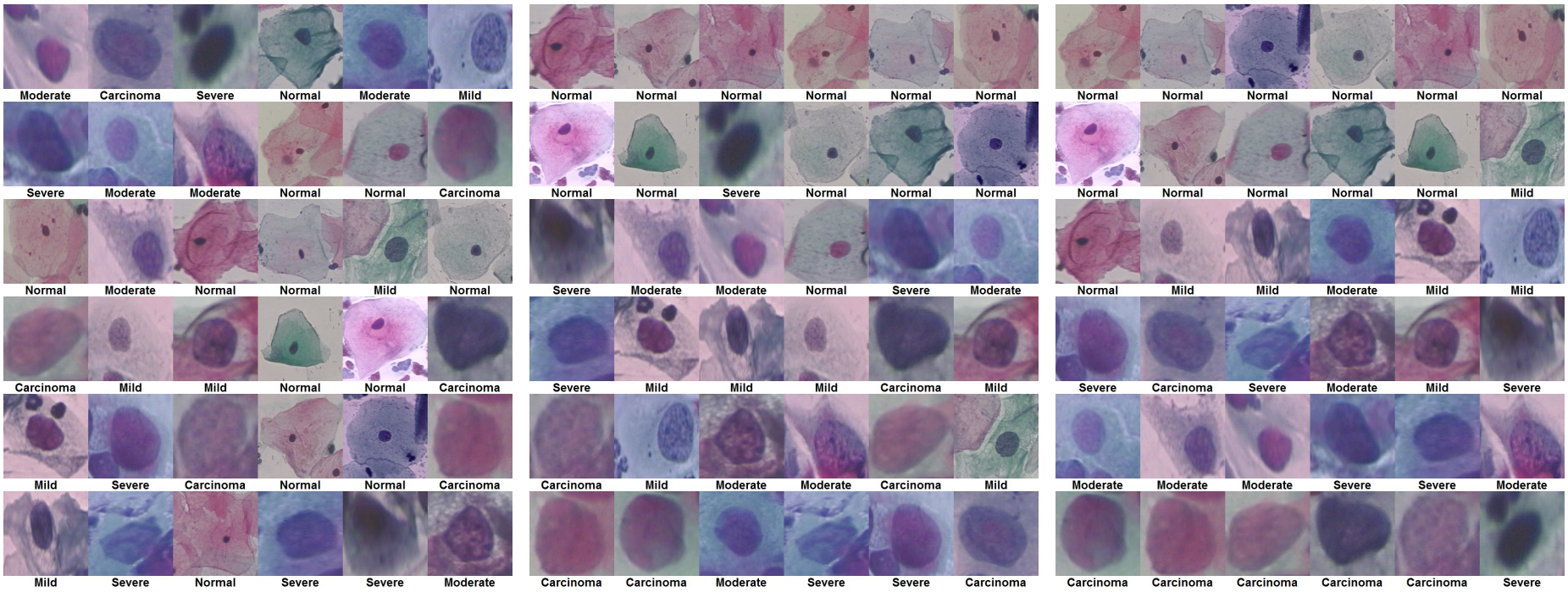


(a) (b)

Figure 82: (a) Random ordering of 30 cells. (b) Initial linear ordering of the leaves of the original tree.

Classification of cervical cells

Ranking of cervical cells



(a) (b) (c)

Figure 82: (a) Random ordering of 30 cells. (b) Initial linear ordering of the leaves of the original tree. (c) Result of the optimal leaf ordering algorithm.