Institute of Informatics
Eötvös Loránd University
Budapest, Hungary

Basic Algorithms for Digital Image Analysis: a course

Dmitrij Csetverikov

with help of Attila Lerch, Judit Verestóy, Zoltán Megyesi, Zsolt Jankó and Levente Hajder

http://visual.ipan.sztaki.hu

Lecture 10: Image segmentation

- Region-oriented segmentation
  - Definition
  - Region growing
  - Region merging
  - Region growing by merging
  - Quadtrees
  - Split-and-merge algorithms
  - Comparison of growing, merging and split-and-merge
- Other segmentation methods

Region-oriented segmentation

Goal: Obtain homogeneous connected regions.

Basic formulation of segmentation: Partition the image R into n subregions $R_1, R_2, \ldots, R_n$ so that

1. $\bigcup_{i=1}^{n} R_i = R$
2. R_i is a connected region, $i = 1, 2, \ldots, n$
3. $R_i \cap R_j = \emptyset$ for all i and $j, i \neq j$
4. $P(R_i) = TRUE$ for $i = 1, 2, \ldots, n$
5. $P(R_i \cup R_j) = FALSE$ for all adjacent i, j $(i \neq j)$

where $P(R_i)$ is a logical **homogeneity predicate** over the points in region R_i: $P(R_i) = TRUE$ means that all pixels in R_i have similar properties, that is, R_i is homogeneous.

Examples of homogeneity predicates for a region:

- Difference between max and min greyvalues is small.
- Difference between any pixel and mean greyvalue is small.

Meaning of conditions (1–5):

1. Completeness: Each pixel is assigned to a region.
2. Connectedness: Points in each region are connected.
3. Regions are disjoint.
4. Each region is homogeneous.
5. Any union of adjacent regions is inhomogeneous: Minimise number of regions.

Region growing and region merging

Region growing: Group pixels or subregions into larger regions while the homogeneity criterion is satisfied.

A simple form of region growing is **pixel aggregation**:

- Select **seed points** and a homogeneity criterion.
- Grow regions around each seed point by appending those neighboring pixels that have similar properties.

Pixel aggregation is usually followed by **region merging**.

- Region merging is a form of region growing.
- It can be applied as a separate operation, or be incorporated into an iterative segmentation algorithm.

Algorithm 1: Region growing by pixel aggregation

1. **Initialisation**: Select N seed pixels S_i, $i = 1, 2, \ldots, N$, and a threshold T. Initialise N regions $R_i^{(0)}$ as the seed pixels S_i. Initialise N region mean values $M_i^{(0)}$ as the greyvalues of S_i.
2. At each iteration k, find **border pixels** of each region $R_i^{(k)}$.
3. Search the 8-neighbourhood of each border pixel of each region. Assign a pixel $p = f(x, y)$ to $R_i^{(k)}$ if
   - p is a 8-neighbour of a border pixel of $R_i^{(k)}$ **and**
   - p has not been assigned yet **and**
   - greyvalue of p is sufficiently close to region mean: $|f(x, y) - M_i^{(k)}| \leq T$
4. Stop if no region growing was possible. Otherwise, update $M_i^{(k)}$ and iterate by going to step 2.

Different homogeneity criteria can be used. In the example below:

- the criterion is the maximum allowed absolute difference T within region;
- region merging is applied after growing.

0	0	5	6	7
1	1	5	8	7
0	***1***	6	***7***	7
2	0	7	6	6
0	1	5	6	5

(a)

a	a	b	b	b
a	a	b	b	b
a	a	b	b	b
a	a	b	b	b
a	a	b	b	b

(b)

a	a	a	a	a
a	a	a	a	a
a	a	a	a	a
a	a	a	a	a
a	a	a	a	a

(c)

Example of pixel aggregation followed by region merging: (a) Original image matrix with two seeds indicated. (b) Result for $T = 4$. (c) Result for $T = 8$.

Major problem with pixel aggregation: **How to select seeds?**

- Result depends heavily on choice and order of seeds.

Region merging

Region merging is a form of region growing. This operation is applied when adjacent regions have similar properties.

- For example, this happens in pixel aggregation when **several seeds** are given for one region.

Different **merging criteria**: Merge two adjacent regions R_i and R_j if

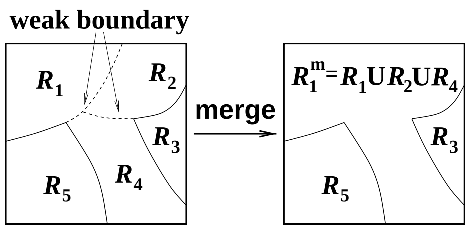
- $P_{agg}(R_i \cup R_j) = TRUE$, where P_{agg} is the criterion used for aggregation, **or**
- $P(R_i \cup R_j) = TRUE$, where P is another homogeneity predicate, **or**
- border between R_i and R_j has no *strong edges* (gradient $> T_{high}$), **or**
- border between R_i and R_j has many *weak locations* (gradient $< T_{low}$),

Region growing by merging

Merging and growing can be combined with **image division** (splitting). This can be done in different ways. The goal is to avoid interactive **specification of seeds**.

Basic idea of region growing by merging:

- Divide image into **atomic regions** of constant greylevel, or other local property.
- Merge similar adjacent regions sequentially until the adjacent regions become sufficiently different.



Region growing by merging.

Summary of region growing

Advantages of region growing:

- Connected regions guaranteed.
- Possibility to incorporate *prior knowledge* about regions: shape, size, texture, spatial relations, etc.

  ⇒ Regions obtained have desired properties.

Drawbacks:

- Prior information may be needed: seeds, starting points.
- **Heuristics** needed: for example, merging rules ('weak borders', etc.)
- Intrinsically **sequential**: no parallel implementation possible.
- **Order-dependence**: Result depends on the order in which pixels are examined.

Possible solutions: Look-ahead, back-tracking.

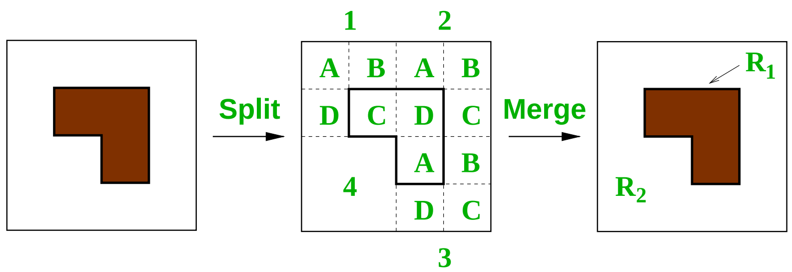
Split-and-merge method and quadtrees

The **split-and-merge** algorithm uses a homogeneity predicate P and has two stages:

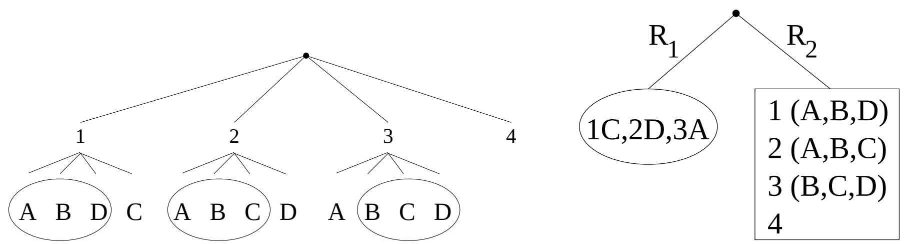
- Top-down: Split image into homogeneous quadrant regions
- Bottom-up: Merge similar adjacent regions

Algorithm 2: Split-and-merge algorithm

1. Top-down: Successively subdivide image and regions into smaller quadrant regions until $P(R_i) = TRUE$ for each R_i. Obtain a **quadtree** structure.
2. Bottom-up: At each level, merge any adjacent regions R_i and R_j for which $P(R_i \cup R_j) = TRUE$.
3. Repeat steps 1 and 2 until no further splitting/merging is possible.



stages of segmentation



quadtree split and merge

segmented regions

Segmentation by split-and-merge approach.

Summary of split-and-merge

Advantages of split-and-merge:

- Connected regions guaranteed
- Some (limited) possibility to incorporate geometric knowledge

Drawbacks:

- Result may depend on position and orientation of image with respect to raster
- Imposes rectangular structure on segmented image
- Relatively large computational load and storage requirements
- Simple quadtree only supports vertical dataflow
  - Adjacent regions to be merged can be 'far away' in quadtree

Quadtrees are widely used in GIS (Geographic Information Systems).

Comparison of a growing and a merging algorithm


image with seeds


grow $T = 20$


grow $T = 30$


grow $T = 40$

- When $T = 20$, face regions with large variation cannot grow.

  $\Rightarrow$ Dark hair region with small variation spreads and occupies large area
- When $T = 30$, face regions can grow and occupy larger areas.

  $\Rightarrow$ Dark hair region shrinks
- Result depends on order and positions of seeds.


merge $T = 20$


merge $T = 30$


merge $T = 40$


merge $T = 20$

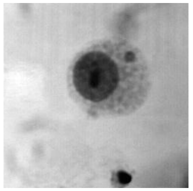

merge $T = 40$

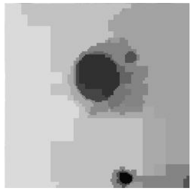

grow $T = 20$

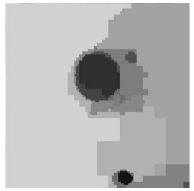

grow $T = 40$

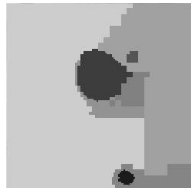
- As T increases, number of regions decreases, merging segmentation becomes coarser. **Dark hair region extends.**
  - In **region growing**, the effect is just the **opposite**: dark region shrinks.

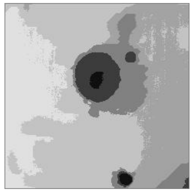
Comparison of split-and-merge and region merging

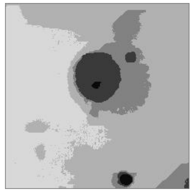

cell image


SaM $T = 40$


SaM $T = 50$


SaM $T = 60$


merge $T = 40$


merge $T = 50$


merge $T = 60$

- **Rectangular structure** of splitting is visible in split-and-merge images.
- Merging segmentation looks finer.

Other segmentation methods

- **Model-based** segmentation: Partition image into segments of known shape and other properties.
- **Edge-based** segmentation: Find object contours by edge tracing.
- **Texture-based** segmentation: Apply region-based techniques to textural properties of regions.
- **Motion-based** segmentation: In an image sequence, locate objects based on their motion.
- **Rule-based** segmentation: Expert system with general-purpose knowledge about images and grouping criteria.