Graph-based methods and segmentation

Filip Malmberg

This lecture

- Image processing using *graphs*.
- Image segmentation using *minimal graph cuts*

Part 1: Image Processing Using Graphs

What is an image?

"We will sometimes regard a *picture* as being a real-valued, non-negative function of two real variables; the value of this function at a point will be called the *gray-level* of the picture at the point."

Rosenfeld, *Picture Processing by Computer*, ACM Computing Surveys, 1969.

What is a digital image?

Storing the (continuous) image in a computer requires digitization, e.g.

- Sampling (recording image values at a finite set of *sampling points*).
- Quantization (discretizing the continuous function values).

Typically, sampling points are located on a Cartesian grid.

Generalized images

This basic model can be generalized in several ways: (cf. Lecture 1)

- Generalized image modalities (e.g., multispectral images)
- Generalized image domains (e.g. video, volume images)
- Generalized sampling point distributions (e.g. non-Cartesian grids)

The methods we develop in image analysis should (ideally) be able to handle this.

Why graph-based?

- Discrete and mathematically simple representation that lends itself well to the development of efficient and provably correct methods.
- A minimalistic image representation – flexibility in representing different types of images.
- A *lot* of work has been done on graph theory in other applications, We can re-use existing algorithms and theorems developed for other fields in image analysis!

Graphs, basic definition

- A graph is a pair $G = (V, E)$, where
  - V is a set.
  - E consists of pairs of elements in V.
- The elements of V are called the *vertices* of G.
- The elements of E are called the *edges* of G.

Graphs basic definition

- An edge spanning two vertices v and w is denoted $e_{v,w}$.
- If $e_{v,w} \in E$, we say that v and w are *adjacent*.
- The set of vertices adjacent to v is denoted $\mathcal{N}(v)$.

Example

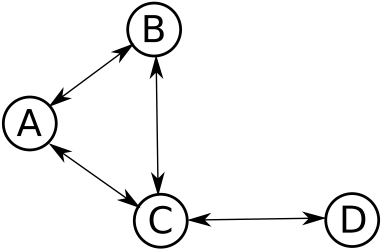

Figure 1: A drawing of an undirected graph with four vertices $\{A, B, C, D\}$ and four edges $\{e_{A,B},\ e_{A,C},\ e_{B,C},\ e_{C,D}\}$.

Example

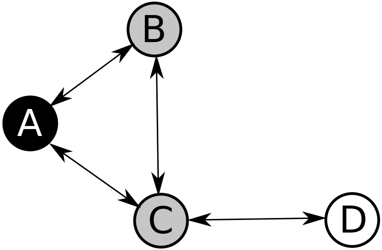

Figure 2: The set $\mathcal{N}(A) = \{B, C\}$ of vertices adjacent to A.

Images as graphs

- Graph based image processing methods typically operate on *pixel adjacency graphs*, i.e., graphs whose vertex set is the set of image elements, and whose edge set is given by an adjacency relation on the image elements.
- Commonly, the edge set is defined as all vertices v, w such that

$$d(v, w) \leq \rho \; . \tag{1}$$

- This is called the *Euclidean adjacency relation*.

Pixel adjacency graphs, 2D

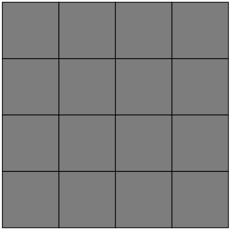

Figure 3: A 2D image with 4×4 pixels.

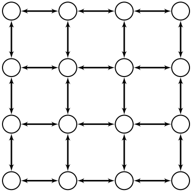

Figure 4: A 4-connected pixel adjacency graph.

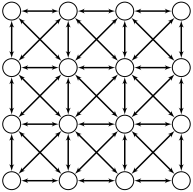

Figure 5: A 8-connected pixel adjacency graph.

Pixel adjacency graphs, 3D

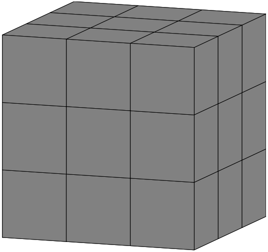

Figure 6: A volume image with $3 \times 3 \times 3$ voxels.

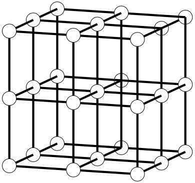

Figure 7: A 6-connected voxel adjacency graph.

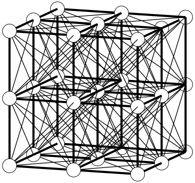

Figure 8: A 26-connected voxel adjacency graph.

Foveal sampling

"Space-variant sampling of visual input is ubiquitous in the higher vertebrate brain, because a large input space may be processed with high peak precision without requiring an unacceptably large brain mass." [6]

Figure 9: Ducks. (Image from Grady 2004)

Foveal sampling

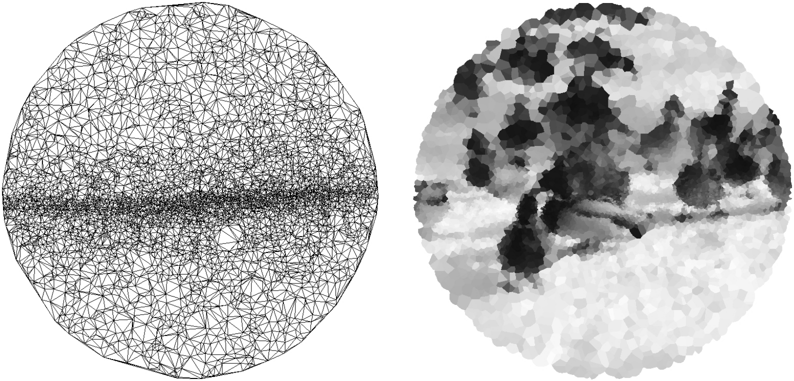

Figure 10: Left: Retinal topography of a Kangaroo. Right: Re-sampled duck image. (Images from Grady 2004)

Region adjacency graphs

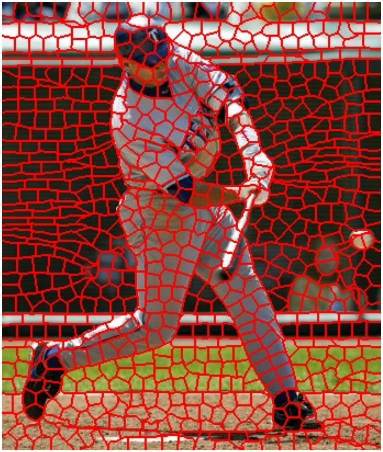

Figure 11: An image divided into superpixels

Directed and undirected graphs

- The pairs of vertices in E may be ordered or unordered.
  - In the former case, we say that G is directed.
  - In the latter case, we say that G is undirected.
- In this lecture, we will mainly consider undirected graphs.

Paths

A *path* is an ordered sequence of vertices where each vertex is adjacent to the previous one.

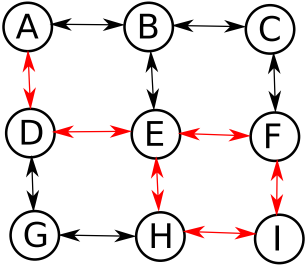

Figure 12: A path $\pi = \langle A, D, E, H, I, F, E \rangle$.

Example, Simple path

A path is *simple* if it has no repeated vertices. Often, simplicity of paths is implied, i.e., the word “simple” is ommited.

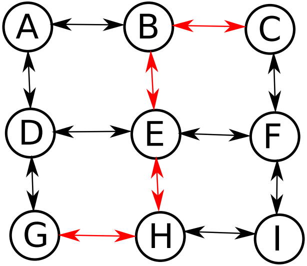

Figure 13: A simple path $\pi = \langle G, H, E, B, C \rangle$.

Paths and connectedness

- Two vertices v and w are *linked* if there exists a path that starts at v and ends at w. We use the notation $v \underset{G}{\sim} w$. We can also say that w is *reachable* from v.
- If all vertices in a graph are linked, then the graph is *connected*.

Subgraphs and connected components

- If G and H are graphs such that $V(H) \subseteq V(G)$ and $E(H) \subseteq E(G)$, then H is a *subgraph* of G.
- If H is a connected subgraph of G and
  - $v \underset{G}{\not\sim} w$ for all vertices $v \in H$ and $w \notin H$,
  - (for any pair of vertices $v, w \in H$ it holds that $e_{v,w} \in E(H)$ iff $e_{v,W} \in E(G)$),

  then H is a *connected component* of G.

Example, connected components

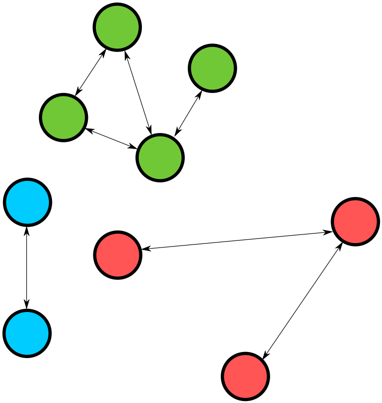

Figure 14: A graph with three connected components.

Graph segmentation

- To segment an image represented as a graph, we want to partition the graph into a number of separate connected components.
- The partitioning can be described either as a *vertex labeling* or as a *graph cut*.

Vertex labeling

We associate each vertex with an element in some set L of *labels*, e.g., $L = \{object, background\}$.

Definition, vertex labeling

A (vertex) labeling $\mathcal{L}$ of G is a map $\mathcal{L} : V \to L$.

Graph cuts

- Informally, a (graph) cut is a set of edges that, if they are removed from the graph, separate the graph into two or more connected components.

Definition, Graph cuts

Let $S \subseteq E$, and $G' = (V, E \setminus S)$. If, for all $e_{v,w} \in S$, it holds that $v \underset{G'}{\not\sim} w$, then S is a *(graph) cut* on G.

Example, cuts

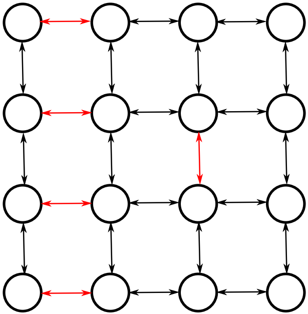

Figure 15: A set of edge (red) that do *not* form a cut.

Example, cuts

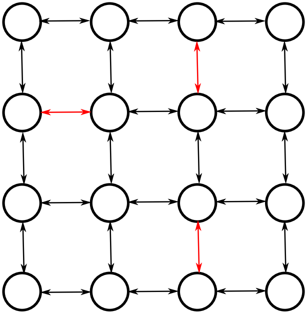

Figure 16: A set of edge (red) that do *not* form a cut.

Example, cuts

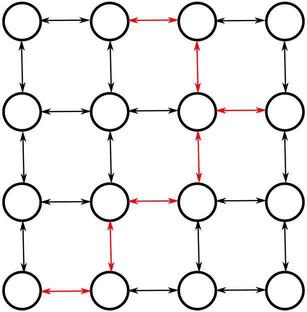

Figure 17: A set of edge (red) that form a cut.

Relation between labelings and cuts

Definition, labeling boundary

The boundary $\partial\mathcal{L}$, of a vertex labeling is the edge set $\partial\mathcal{L} = \{e_{v,w} \in E \mid \mathcal{L}(v) \neq \mathcal{L}(w)\}$.

Theorem

For any graph $G = (V, E)$ and set of edges $S \subseteq E$, the following statements are equivalent: [7]*

1. *There exists a vertex labeling $\mathcal{L}$ of G such that $S = \partial\mathcal{L}$.*
2. *S is a cut on G.*

*) Provided that $|L|$ is "large enough".

Relation between labelings and cuts

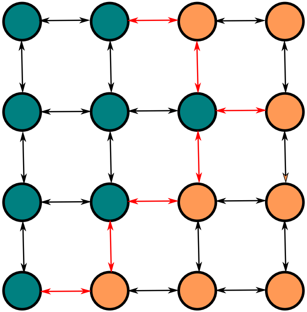

Figure 18: Duality betwen cuts and labelings.

Summary, Part 1

- Basic graph theory
  - Directed and undirected graphs
  - Paths and connectedness
  - Subgraphs and connected components
- Images as graphs
  - Pixel adjacency graphs in 2D and 3D
  - Alternative graph constructions
- Graph partitioning
  - Vertex labeling and graph cuts