Computer Vision I

Bjoern Andres, Holger Heidrich

TU Dresden

Winter Term 2021/2022

We consider:

- $n_0, n_1 \in \mathbb{N}$ called the height and width of a digital image, $V = [n_0] \times [n_1]$ called the set of pixels, and the grid graph $G = (V, E)$
- A non-empty set R whose elements are called colors
- A function $x\colon V \to R$ called a digital image

The task of pixel classification is concerned with making decisions at the pixels, e.g., decisions $y : V \to \{0, 1\}$ indicating whether a pixel $v \in V$ is of interest ($y_v = 1$) or not of interest ($y_v = 0$).

Pixel classification

Source: `https://www.pexels.com/photo/nature-flowers-garden-plant-67857/`

For instance, we may wish to map to 1 precisely those pixels of the above image that depict the yellow part of any of the flowers.

We begin with a trivial mathematical abstraction of the task of pixel classification:

Definition. For any $c\colon V \to \mathbb{R}$, the instance of the **trivial pixel classification problem** w.r.t. c has the form

$$\min_{y \in \{0,1\}^V} \sum_{v \in V} c_v \, y_v \tag{1}$$

In practice, we would seek to construct the function c w.r.t. the image in such a way that

- $c_v < 0$ if we consider $y_v = 1$ the right decision
- $c_v > 0$ if we consider $y_v = 0$ the right decision

Pixel classification

Assuming the decision for a pixel $v \in V$ depends on the color $x_v \in R$ of that pixel only, we can

- construct a function $\xi \colon R \to \mathbb{R}$
- define $c_v = \xi(x_v)$ for any $v \in V$.

In some practical applications, e.g. photo editing, a suitable function ξ can be constructed manually, typically with the help of carefully designed GUIs.

Assuming the decision for a pixel $v \in V$ depends on the location v and on the colors of all pixels in a neighborhood $V_d(v) \subseteq V$ around v, e.g.

$$V_d(v) = \{w \in V \mid \|v - w\|_{\max} \leq d\} \ ,$$

we can

- construct, for any pixel v, a function $\xi_v \colon R^{V_d(v)} \to \mathbb{R}$ that assigns a real number $\xi_v(x')$ to any coloring $x' \colon V_d(v) \to R$ of the d-neighborhood of v
- define $c_v = \xi(x_{V_d(v)})$ for any $v \in V$.

The task of constructing such functions ξ_v is typically addressed by means of **machine learning**, e.g., logistic regression or a CNN.

In practice, solutions to the trivial pixel classification problem can be improved by exploiting **prior knowledge** about feasible combinations of decisions.

Firstly, we consider prior knowledge saying that decisions at neighboring pixels $v, w \in V$ are more likely to be equal ($y_v = v_w$) than unequal ($y_v \neq y_w$).

Definition. For any $c \colon V \to \mathbb{R}$ and any $c' \colon E \to \mathbb{R}_0^+$, the instance of the **smooth pixel classification problem** w.r.t. c and c' has the form

$$\min_{y \in \{0,1\}^V} \quad \underbrace{\sum_{v \in V} c_v \, y_v + \sum_{\{v,w\} \in E} c'_{\{v,w\}} \, |y_v - y_w|}_{\varphi(y)} \tag{2}$$

A naïve algorithm for this problem is local search with a transformation $T_v\colon \{0,1\}^V \to \{0,1\}^V$ that changes the decision for a single pixel, i.e., for any $y\colon V \to \{0,1\}$ and any $v, w \in V$:

$$T_v(y)(w) = \begin{cases} 1 - y_w & \text{if } w = v \\ y_w & \text{otherwise} \end{cases} .$$

Initially, $y\colon V \to \{0,1\}$ and $W = V$
while $W \neq \emptyset$
 $W' := \emptyset$
 for each $v \in W$
  if $\varphi(T_v(y)) - \varphi(y) < 0$
   $y := T_v(y)$
   $W' := W' \cup \{w \in V \mid \{v,w\} \in E\}$
 $W := W'$

- So far, we have studied a local search algorithm for the smooth pixel classification problem.
- On the one hand, this algorithm is easy to implement and has straight-forward generalizations, e.g., to the case of more than two classes.
- On the other hand, it does not necessarily solve smooth pixel classification with two classes to optimality.
- Next, we will reduce the smooth pixel classification problem with two classes to the well-known **minimum st-cut problem** that can be solved exactly and efficiently.
- The notes are organized as follows
  - Definition of the minimum st-cut problem
  - Submodularity
  - Reduction of the smooth pixel classification problem

Definition 1

A 5-tuple $N = (V, E, s, t, \gamma)$ is called a **network** iff (V, E) is a directed graph and $s \in V$ and $t \in V$ and $s \neq t$ and $\gamma : E \to \mathbb{R}_0^+$.

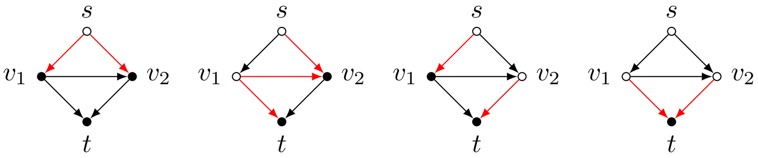
The nodes s and t are called the **source** and the **sink** of N, respectively.

For any edge $e \in E$, γ_e is called the **capacity** of e in N.

Definition 2

Let (V, E) be a directed graph. Let $s \in V$ and $t \in V$ and $s \neq t$.

- $X \subseteq V$ is called an st-**cutset** of (V, E) iff $s \in X$ and $t \notin X$.
- $Y \subseteq E$ is called an st-**cut** of (V, E) iff there exists an st-cutset X such that $Y = \{vw \in E \mid v \in X \wedge w \notin X\}$.



Definition 3

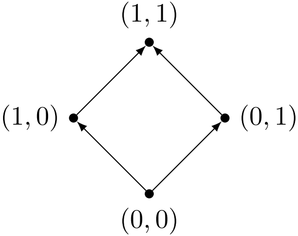
The instance of the **Minimum** st**-Cut Problem** w.r.t. a network $N = (V, E, s, t, \gamma)$ is to

$$\begin{aligned} \min_{x \in \{0,1\}^V} \quad & \sum_{vw \in E} x_v \, (1 - x_w) \, \gamma_{vw} && (3) \\ \text{subject to} \quad & x_s = 1 && (4) \\ & x_t = 0 && (5) \end{aligned}$$

Definition 4

A **lattice** $(S, \preceq)$ is a set S, equipped with a partial order $\preceq$, such that any two elements of S have an infimum and a supremum w.r.t. $\preceq$.

Example. $(\{0,1\}^2, \preceq)$ with $\preceq := \{(s,t) \in S \times S \mid s_1 \leq t_1 \wedge s_2 \leq t_2\}$.



For any $s, t \in \{0,1\}^2$,

$$\sup(s,t) = (\max\{s_1, t_1\}, \max\{s_2, t_2\})$$
$$\inf(s,t) = (\min\{s_1, t_1\}, \min\{s_2, t_2\})$$

Definition 5

A function $f : S \to \mathbb{R}$ is called **submodular** w.r.t. a lattice $(S, \preceq)$ iff

$$\forall s, t \in S \qquad f(\inf(s,t)) + f(\sup(s,t)) \leq f(s) + f(t) \ . \tag{6}$$

Lemma 6

For any $f : \{0,1\}^2 \to \mathbb{R}$*, the following statements are equivalent.*

1. f *is is submodular w.r.t. the the lattice* $(\{0,1\}^2, \preceq)$
2. $f(0,0) + f(1,1) \leq f(1,0) + f(0,1)$
3. *The unique form*

$$c_\emptyset + c_{\{1\}}x_1 + c_{\{2\}}x_2 + c_{\{1,2\}}x_1x_2$$

of f *is such that* $c_{\{1,2\}} \leq 0$.

Proof.

- $f(0,0) + f(1,1) \leq f(1,0) + f(0,1)$ is the only condition in

$$\forall s,t \in S \qquad f(\inf(s,t)) + f(\sup(s,t)) \leq f(s) + f(t)$$

  which is not generally true. Thus, (1.) is equivalent to (2.).

- We have

$$\begin{aligned} f(0,0) &= c_\emptyset \\ f(1,0) &= c_\emptyset + c_{\{1\}} \\ f(0,1) &= c_\emptyset \qquad\quad + c_{\{2\}} \\ f(1,1) &= c_\emptyset + c_{\{1\}} + c_{\{2\}} + c_{\{1,2\}} \quad . \end{aligned}$$

  Therefore,

$$c_{\{1,2\}} = f(1,1) - f(1,0) - f(0,1) + f(0,0)$$

  and thus, (2.) is equivalent to (3.).

Lemma 7

The sum of finitely many submodular functions is submodular.

Lemma 8

For every $f : \{0,1\}^2 \to \mathbb{R}$*, there exist unique* $a_0 \in \mathbb{R}$ *and* $a_1, a_{\bar{1}}, a_2, a_{\bar{2}}, a_{12}, a_{\bar{1}2} \in \mathbb{R}_0^+$ *such that*

$$a_1 a_{\bar{1}} = a_2 a_{\bar{2}} = a_{12} a_{\bar{1}2} = 0 \tag{7}$$

and

$$\begin{aligned} \forall x \in \{0,1\}^2 \quad f(x) = \quad & a_0 \\ & + a_1 x_1 + a_{\bar{1}}(1 - x_1) \\ & + a_2 x_2 + a_{\bar{2}}(1 - x_2) \\ & + a_{12} x_1 x_2 + a_{\bar{1}2}(1 - x_1) x_2 \ . \end{aligned} \tag{8}$$

Proof.

- Comparison of (8) with the unique form

$$c_{\emptyset} + c_{\{1\}} x_1 + c_{\{2\}} x_2 + c_{\{1,2\}} x_1 x_2$$

  yields

$$\begin{aligned} a_0 + a_{\bar{1}} + a_{\bar{2}} &= c_{\emptyset} \\ a_1 - a_{\bar{1}} &= c_{\{1\}} \\ a_2 - a_{\bar{2}} + a_{\bar{1}2} &= c_{\{2\}} \\ a_{12} - a_{\bar{1}2} &= c_{\{1,2\}} \end{aligned} \tag{9}$$

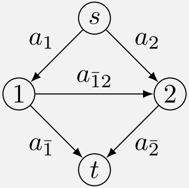
- By these equations (from bottom to top), (7) and c define a uniquely.

Lemma 9 (Kolmogorov and Zabih)

For every **submodular** $f : \{0,1\}^2 \to \mathbb{R}$ *and its unique coefficient* $a_0 \in \mathbb{R}$ *from Lemma 8,*

$$\min_{x \in \{0,1\}^2} f_x - a_0 \tag{10}$$

is equal to the weight of a **minimum** st**-cut** *in the graph below whose edge weights are the (unique, non-negative) coefficients from Lemma 8.*



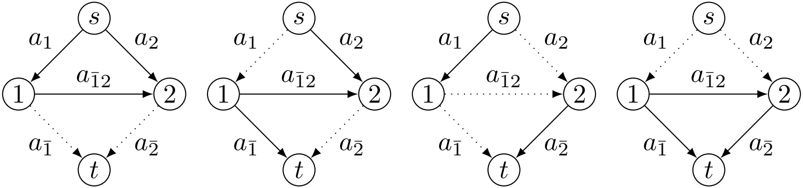
Moreover, f *is minimal at* $\hat{x} \in \{0,1\}^2$ *iff* $\{j \in \{1,2\} \mid \hat{x}_j = 0\}$ *is a* **minimum** st**-cutset** *of the above graph.*

Proof.

- Submodularity of f implies $a_{12} = 0$ in (9), by Lemma 6 and (7).
- Comparison of the four possible minima of f,

$$
\begin{aligned}
f(0,0) &= a_0 + a_{\bar{1}} + a_{\bar{2}} \\
f(1,0) &= a_0 + a_1 + a_{\bar{2}} \\
f(0,1) &= a_0 + a_{\bar{1}} + a_2 + a_{\bar{1}2} \\
f(1,1) &= a_0 + a_1 + a_2 + a_{12} \quad ,
\end{aligned}
$$

with the four possible minimum cuts below proves the Lemma.



Definition 10

For any smooth pixel classification problem

$$\min_{y \in \{0,1\}^V} \quad \underbrace{\sum_{v \in V} c_v \, y_v + \sum_{\{v,w\} \in E} c'_{\{v,w\}} \, |y_v - y_w|}_{\varphi(y)} \tag{11}$$

the **induced minimum st-cut problem** is defined by the network (V', E', s, t, γ) such that $V' = V \cup \{s, t\}$,

$$\begin{aligned} E' = & \{(s,v) \in V'^2 \mid c_v > 0\} \cup \{(v,t) \in V'^2 \mid c_v < 0\} \\ & \cup \{(v,w) \in V'^2 \mid \{v,w\} \in E\} \end{aligned} \tag{12}$$

and $\gamma \colon E' \to \mathbb{R}_0^+$ such that

$$\forall (s,v) \in E' \colon \quad \gamma_{(s,v)} = c_v \tag{13}$$

$$\forall (v,t) \in E' \colon \quad \gamma_{(v,t)} = -c_v \tag{14}$$

$$\forall \{v,w\} \in E \colon \quad \gamma_{(v,w)} = \gamma_{(w,v)} = c'_{\{v,w\}} \quad . \tag{15}$$

Lemma 11

For any smooth pixel classification problem w.r.t. a pixel grid graph $G = (V, E)$ *and the induced minimum* st*-cut problem with the network* (V', E', s, t, γ), $\hat{y} : V \to \{0, 1\}$ *is an optimal pixel classification iff* $\{v \in V \mid \hat{y}_v = 0\}$ *is an optimal* st*-cutset.*

Proof (sketch). The function φ is submodular, by Lemma 7 and $c' > 0$. The statement holds by Lemma 8 and the fact that for all $y \in \{0, 1\}^V$:

$$\varphi(y) = \sum_{v \in V} c_v \, y_v + \sum_{\{v,w\} \in E} c'_{\{v,w\}} \, (y_v (1 - y_w) + (1 - y_v) y_w) \quad .$$