Computer Vision II

Bjoern Andres

Machine Learning for Computer Vision
TU Dresden

April 17, 2020

Notation

Throughout the course, we shall use the following notation:

- We write "iff" as shorthand for "if and only if"
- For any $m \in \mathbb{N}$, we define $[m] = \{0, \ldots, m-1\}$.
- For any set A, we denote by 2^A the power set of A
- For any set A and any $m \in \mathbb{N}$, we denote by $\binom{A}{m} = \{B \in 2^A \mid |B| = m\}$ the set of all m-elementary subsets of A
- For any sets A, B, we denote by B^A the set of all maps from A to B

Bilateral Filter

We recall from the course Computer Vision I operations (filters) on digital images by looking at the example of the bilateral filter. Bilateral filtering is a powerful tool for image de-noising and is implemented, e.g., in GIMP and Adobe Photoshop.

We consider:

- A grid graph $G = (V, E)$ whose nodes we refer to as **pixels**. E.g., in case of a 2-dimensional grid of $n_0 \cdot n_1$ pixels, $V = [n_0] \times [n_1]$
- A non-empty set R whose elements we refer to as **intensities**, **gray values** or **colors**. E.g., $R = [0, 1] \subset \mathbb{R}$ or $R = \{0, \ldots, 255\}$
- A map $f \colon V \to R$ called a **digital image**

Bilateral Filter

Given

- a metric $d_s : V \times V \to \mathbb{R}_0^+$ and a decreasing $w_s \colon \mathbb{R}_0^+ \to [0, 1]$
- a metric $d_r : R \times R \to \mathbb{R}_0^+$ and a decreasing $w_r \colon \mathbb{R}_0^+ \to [0, 1]$
- a $N : V \to 2^V$ that defines, for every pixel $v \in V$, a set $N(v) \subseteq V$ called the **spatial neighborhood** of v
- the $\nu : R^V \to \mathbb{R}^V$, called **normalization**, such that for any digital image $f : V \to R$ and any pixel $v \in V$:

$$\nu(f)(v) = \sum_{v' \in N(v)} w_s(d_s(v, v'))\, w_r(d_r(f(v), f(v'))) \ , \qquad (1)$$

the **bilateral filter** w.r.t. d_s, w_s, d_r, w_r and N is the $\beta : R^V \to \mathbb{R}^V$ such that for any digital image $f : V \to R$ and any pixel $v \in V$:

$$\beta(f)(v) = \frac{1}{\nu(f)(v)} \sum_{v' \in N(v)} w_s(d_s(v, v'))\, w_r(d_r(f(v), f(v')))\, f(v') \qquad (2)$$

Bilateral Filter

Example

- $n_0 = 768$, $n_1 = 1024$, $V = [n_0] \times [n_1]$, $R = [0, 1] \subset \mathbb{R}$
- $d_s(v, v') = \|v - v'\|_2$ and, for a filter parameter $\sigma_s > 0$:

$$w_s(x) = \frac{1}{\sigma_s \sqrt{2\pi}} \exp\left(-\frac{x^2}{2\sigma_s^2}\right) \tag{3}$$

- $d_r(g, g') = |g - g'|$ and, for a filter parameter $\sigma_r > 0$:

$$w_r(x) = \frac{1}{1 + \frac{x^2}{\sigma_r^2}} \tag{4}$$

- for a filter parameter $n \in \mathbb{R}_0^+$:

$$N(v) = \{v' \in V \mid d_s(v, v') < n\} \tag{5}$$

Bilateral Filter

Suggested self-study:

- Implement a bilateral filter for gray-scale images
- Apply your implementation to a digital picture of yours or from the web
- Try different metrics d_s, d_r and weighting functions w_s, w_r
- Try iterating bilateral filtering
- Share and discuss your implementations, outputs and findings via OPAL

Advanced self-study:

- Define, implement and apply bilateral filtering for color images
- Share and discuss your implementations, outputs and findings via OPAL