2. Local operators

Now, we address the second class of operators in image processing. To compute a value of the new image, these operators not only use the value at the corresponding position in the input image, but also values from a local neighborhood.

Example 2.1 (The moving average). Let $f : \mathbb{R}^d \to \mathbb{R}$ be a noisy image, i. e. $f = f_0 + n$. Here, $f_0 : \mathbb{R}^d \to [0,1]$ is the (sought) original image without noise and $n : \mathbb{R}^d \to \mathbb{R}$ is noise with mean 0. Since the mean of the noise is 0, a local averaging should reduce the noise. Thus, we consider

$$M_r f(x) = \fint_{B_r(x)} f(y)\,\mathrm{d}y\,, \text{ where } B_r(x) = \left\{ y \in \mathbb{R}^d \,:\, \|y - x\|_2 < r \right\}.$$

This operation is one of the simplest methods for noise reduction and is called *moving average*. For now, we intentionally consider $\mathbb{R}^d$ here instead of Ω to avoid boundary effects.

Instead of a ball in the Euclidean norm, one typically also considers other norms. In the discrete case, for instance, the ∞-norm is particularly easy to compute:

$$M_r^\infty f(x) = \fint_{B_r^\infty(x)} f(y)\,\mathrm{d}y\,, \text{ where } B_r^\infty(x) = \left\{ y \in \mathbb{R}^d \,:\, \|y - x\|_\infty < r \right\}.$$

Using $\chi_{B_r(0)}$, one can express M_r in a different way:

$$M_r f(x) = \int_{\mathbb{R}^d} f(y) \left(\tfrac{1}{|B_r(0)|} \chi_{B_r(0)}(y - x) \right) \mathrm{d}y \underset{\text{Theorem B.7}}{\overset{\phi(y) = y - x}{=}} \int_{\mathbb{R}^d} f(x+y) \left(\tfrac{1}{|B_r(0)|} \chi_{B_r(0)}(y) \right) \mathrm{d}y$$

volume is always same, thus, *1/lB_{r}{0}l

This is the typical way to express a linear filter:

Definition 2.2. For measurable $\psi : \Omega \to \mathbb{R}$ and $f : \mathbb{R}^d \to \mathbb{R}$,

$$(\psi \star f)(x) = \int_\Omega \psi(y) f(x+y)\,\mathrm{d}y$$

is called *cross-correlation* of f and ψ at position $x \in \mathbb{R}^d$. The operation

$$M_\psi : f \mapsto (x \mapsto (\psi \star f)(x))$$

is called *linear filter* with *kernel* ψ.

Proposition 2.3 (Properties of the cross-correlation)**.**

(i) Let $1 \le p, q, r \le \infty$ with $\frac{1}{r} + 1 = \frac{1}{p} + \frac{1}{q}$ (using the convention $\frac{1}{\infty} = 0$), $f \in L^p(\Omega)$ and $g \in L^q(\mathbb{R}^d)$. Then, $(f \star g) \in L^r(\mathbb{R}^d)$ and

$$\|f \star g\|_{L^r(\mathbb{R}^d)} \le \|f\|_{L^p(\Omega)} \|g\|_{L^q(\mathbb{R}^d)}\,.$$

This is the counterpart of Young's convolution inequality for the cross-correlation, cf. Proposition B.10 (i).

If $\Omega = \mathbb{R}^d$, we have $(f \star g)(x) = (g \star f)(-x)$ for all $x \in \mathbb{R}^d$ and

$$\|f \star g\|_{L^r(\mathbb{R}^d)} = \|g \star f\|_{L^r(\mathbb{R}^d)}\,.$$

(ii) *Let* $\psi \in C_c^k(\mathbb{R}^d)$ *and* $f \in L^p(\Omega)$ *with* $1 \le p \le \infty$. *Then,* $(f \star \psi) \in C^k(\mathbb{R}^d)$. *If* α *is a multi-index with* $|\alpha| \le k$, *we have*

$$\frac{\partial^\alpha}{\partial x^\alpha}(f \star \psi) = f \star \frac{\partial^\alpha}{\partial x^\alpha}\psi.$$

If $\Omega = \mathbb{R}^d$, *we also have*

$$\frac{\partial^\alpha}{\partial x^\alpha}(\psi \star f) = (-1)^{|\alpha|}\frac{\partial^\alpha}{\partial x^\alpha}\psi \star f.$$

(iii) *Let* $\psi \in L^1(\mathbb{R}^d)$ *be such that* $\psi \ge 0$ *and* $\int_{\mathbb{R}^d} \psi(x)\,\mathrm{d}x = 1$. *Moreover, for* $\epsilon > 0$, *let*

$$\psi_\epsilon : \mathbb{R}^d \to \mathbb{R}, x \mapsto \tfrac{1}{\epsilon^d}\psi(\tfrac{x}{\epsilon})$$

and let $f \in L^\infty(\mathbb{R}^d)$. *If* f *is continuous in* $x \in \mathbb{R}^d$, *we have*

$$\lim_{\epsilon \to 0}(\psi_\epsilon \star f)(x) = f(x).$$

If f *is uniformly continuous,* $\psi_\epsilon \star f$ *converges to* f *on each compact subset of* $\mathbb{R}^d$ *uniformly.*

Proof.

(i) For $x \in \mathbb{R}^d$ and $r = \infty$, we get using Hölder's inequality (cf. Lemma B.5)

$$|(f \star g)(x)| \le \int_\Omega |f(y)g(x+y)|\,\mathrm{d}y \le \|f\|_{L^p(\Omega)}\,\|g(\cdot + x)\|_{L^q(\Omega)} \le \|f\|_{L^p(\Omega)}\,\|g\|_{L^q(\mathbb{R}^d)}\,.$$

Since the right hand side is independent of x, taking the supremum over $x \in \mathbb{R}^d$ shows

$$\|f \star g\|_{L^\infty(\mathbb{R}^d)} \le \|f\|_{L^p(\Omega)}\,\|g\|_{L^q(\mathbb{R}^d)}\,.$$

Now consider $r = p$, which implies $q = 1$. For $p = \infty$, we have $r = \infty$, which we already covered above. For $p < \infty$, we integrate the p-th power of

$$|(f \star g)(x)| \le \int_\Omega |f(y)g(x+y)|\,\mathrm{d}y\,.$$

and get

$$\int_{\mathbb{R}^d} |(f \star g)(x)|^p\,\mathrm{d}x \le \int_{\mathbb{R}^d}\left(\int_\Omega |f(y)g(x+y)|\,\mathrm{d}y\right)^p \mathrm{d}x\,. \tag{$*$}$$

For $p = 1$, we get using Fubini's theorem (cf. Theorem B.6)

$$\|f \star g\|_{L^1(\mathbb{R}^d)} = \int_{\mathbb{R}^d} |(f \star g)(x)|\,\mathrm{d}x \overset{(*)}{\le} \int_\Omega |f(y)| \int_{\mathbb{R}^d} |g(x+y)|\,\mathrm{d}x\,\mathrm{d}y = \|f\|_{L^1(\Omega)}\,\|g\|_{L^1(\mathbb{R}^d)}\,.$$

For $1 < p < \infty$, we get using p^* with $\frac{1}{p} + \frac{1}{p^*} = 1$ that

$$\begin{aligned}
\int_{\mathbb{R}^d} |(f \star g)(x)|^p\,\mathrm{d}x &\overset{(*)}{\le} \int_{\mathbb{R}^d}\left(\int_\Omega |f(y)|\,|g(x+y)|^{\frac{1}{p}}\,|g(x+y)|^{\frac{1}{p^*}}\,\mathrm{d}y\right)^p \mathrm{d}x \\
&\overset{\text{Hölder}}{\le} \int_{\mathbb{R}^d}\left(\int_\Omega |f(y)|^p\,|g(x+y)|\,\mathrm{d}y\right)^{\frac{p}{p}}\left(\int_\Omega |g(x+y)|\,\mathrm{d}y\right)^{\frac{p}{p^*}} \mathrm{d}x \\
&\overset{\text{Fubini}}{\le} \int_\Omega |f(y)|^p \int_{\mathbb{R}^d} |g(x+y)|\,\mathrm{d}x\,\mathrm{d}y\,\|g\|_{L^1(\mathbb{R}^d)}^{\frac{p}{p^*}} \\
&= \|f\|_{L^p(\Omega)}^p\,\|g\|_{L^1(\mathbb{R}^d)}^{\frac{p}{p^*}+1}\,.
\end{aligned}$$

Taking the p-th root shows

$$\|f \star g\|_{L^p(\Omega)} \leq \|f\|_{L^p(\Omega)} \|g\|_{L^1(\mathbb{R}^d)} .$$

The case $r \neq p$ can be shown similarly, but is more complicated and will not be shown here. The proof for the corresponding statement for the convolution can be found in the book of Lieb and Loss[1].

In case $\Omega = \mathbb{R}^d$, using the substitution rule for multiple variables (Theorem B.7), we get for $x \in \mathbb{R}^d$ that

$$\begin{aligned}(f \star g)(x) &= \int_{\mathbb{R}^d} f(y)g(x+y)\,\mathrm{d}y \overset{\phi(y)=x+y}{=} \int_{\mathbb{R}^d} f(y-x)g(y)\,\mathrm{d}y \\ &= \int_{\mathbb{R}^d} g(y)f(-x+y)\,\mathrm{d}y = (g \star f)(-x).\end{aligned}$$

From this and the substitution rule for multiple variables, we immediately get

$$\|f \star g\|_{L^r(\mathbb{R}^d)} = \|g \star f\|_{L^r(\mathbb{R}^d)} .$$

(ii) Let $i \in \{1, \dots, d\}$ and $x \in \mathbb{R}^d$. We have

$$\frac{(f \star \psi)(x+\epsilon e_i) - (f \star \psi)(x)}{\epsilon} = \int_\Omega f(y) \frac{\psi(x+y+\epsilon e_i) - \psi(x+y)}{\epsilon}\,\mathrm{d}y\,. \tag{$*$}$$

For $g \in C_c^1(\mathbb{R})$, the difference quotient converges uniformly to $\partial_i g$ (will be shown in Lemma 2.4). Thus,

$$\frac{\psi(x+y+\epsilon e_i) - \psi(x+y)}{\epsilon} =: \psi_i^\epsilon(x+y)$$

converges uniformly to $\partial_i\psi(x+y)$ for $y \in \mathbb{R}^d$. In other words, $\|\psi_i^\epsilon - \partial_i\psi\|_{L^\infty(\mathbb{R}^d)} \to 0$ for $\epsilon \to 0$. Since ψ has compact support, there is a compact set $K \subset \mathbb{R}^d$ such that $\operatorname{supp}\psi(x+\cdot) \subset K$ and $\operatorname{supp}\psi_i^\epsilon(x+\cdot) \subset K$ for all $\epsilon \in (0,1)$. With this, we get

$$\left| \int_\Omega f(y)\psi_i^\epsilon(x+y)\,\mathrm{d}y - \int_\Omega f(y)\partial_i\psi(x+y)\,\mathrm{d}y \right| \leq \underbrace{\|\psi_i^\epsilon - \partial_i\psi\|_{L^\infty(\mathbb{R}^d)}}_{\to 0 \text{ for } \epsilon\to 0} \underbrace{\|\chi_K f\|_{L^1(\Omega)}}_{\leq \|\chi_K\|_{L^{p^*}(\Omega)}\|f\|_{L^p(\Omega)}} .$$

This shows that when passing to the limit $\epsilon \to 0$ in $(*)$, integration and limit on the right hand side can be interchanged. Thus, we get

$$\partial_i(f \star \psi) = f \star \partial_i\psi.$$

Iterating this, the first claim follows. In case $\Omega = \mathbb{R}^d$, we have

$$\partial_i(\psi \star f)(x) = \partial_i\left((f \star \psi)(-x)\right) = -(f \star \partial_i\psi)(-x) = -(\partial_i\psi \star f)(x).$$

Iterating this, the second claim follows.

Note that $\partial_i(f \star \psi) = f \star \partial_i\psi$ can also be proven using Corollary B.12, which follows from the dominated convergence theorem (Theorem B.11).

[1] E. H. Lieb and M. Loss. *Analysis*. 2nd ed. Vol. 14. Graduate Studies in Mathematics. American Mathematical Society, 2001.

(iii) Exercise. [1, Theorem 3.13.3] shows the statement for uniformly continuous f. $\square$

Lemma 2.4. *Let $\Omega \subset \mathbb{R}^d$ be open and $f \in C_c^1(\Omega)$. Then, for all $i \in \{1, \dots, d\}$ and $\epsilon \in (0, \infty)$, there is a $\delta \in (0, \infty)$ with*

$$\left| \frac{f(x + he_i) - f(x)}{h} - \partial_i f(x) \right| < \epsilon \text{ for all } x \in \Omega \text{ and } h \in (-\delta, \delta) \setminus \{0\} \text{ with } x + he_i \in \Omega.$$

In other words, the difference quotient of f converges uniformly to $\partial_i f$.

Proof. Let $i \in \{1, \dots, d\}$ and $\epsilon \in (0, \infty)$ be arbitrary, but fixed. Since $f \in C_c^1(\Omega)$, we get $f \in C_c^1(\mathbb{R}^d)$ by extending f with zero. Thus, $\partial_i f \in C_c^0(\mathbb{R}^d)$. Since $\operatorname{supp}(\partial_i f)$ is compact, $\partial_i f$ is even uniformly continuous on its support and, since it is zero outside its support, also uniformly continuous on $\mathbb{R}^d$. In particular, there is a $\delta \in (0, \infty)$ with

$$|\partial_i f(x) - \partial_i f(y)| < \epsilon \text{ for all } x, y \in \mathbb{R}^d \text{ with } |x - y| < \delta.$$

Let $x \in \mathbb{R}^d$ and $h \in (0, \delta)$. The function $g : [0, h] \to \mathbb{R}, t \mapsto f(x + te_i)$ is continuous on $[0, h]$ and differentiable on $(0, h)$ since $f \in C^1(\mathbb{R}^d)$. By the mean value theorem, there is a $\zeta \in (0, h)$ with

$$\partial_i f(x + \zeta e_i) = g'(\zeta) \overset{\text{mean value thm.}}{=} \frac{g(h) - g(0)}{h} = \frac{f(x + he_i) - f(x)}{h}.$$

Then, $|x + \zeta e_i - x| < h < \delta$ and we get

$$\left| \frac{f(x + he_i) - f(x)}{h} - \partial_i f(x) \right| = |\partial_i f(x + \zeta e_i) - \partial_i f(x)| < \epsilon.$$

The case $0 < -h < \delta$ is shown analogously. $\square$

Proposition 2.5. *Let $1 < p \leq \infty$ and q such that $\frac{1}{p} + \frac{1}{q} = 1$. Moreover, let $f \in L^p(\Omega)$ and $g \in L^q(\mathbb{R}^d)$. Then, $f \star g \in C(\mathbb{R}^d)$.*

Proof. Let $x, h \in \mathbb{R}^d$. We have,

$$\begin{aligned}
|(f \star g)(x + h) - (f \star g)(x)| &\leq \int_\Omega |f(y)g(x + h + y) - f(y)g(x + y)| \,\mathrm{d}y \\
&= \int_\Omega |f(y)| \, |g(x + h + y) - g(x + y)| \,\mathrm{d}y \\
&\overset{\text{Hölder}}{\leq} \left(\int_\Omega |f(y)|^p \,\mathrm{d}y \right)^{\frac{1}{p}} \left(\int_\Omega |g(x + h + y) - g(x + y)|^q \,\mathrm{d}y \right)^{\frac{1}{q}} \\
&\leq \|f\|_{L^p(\Omega)} \|T_h g - g\|_{L^q(\mathbb{R}^d)} .
\end{aligned}$$

Here, $T_h g$ is the *translation operator* with h, i. e. $T_h g(x) = g(x + h)$. $\|T_h g - g\|_{L^q}$ converges to 0 for $h \to 0$ (exercise) and with this the claim follows. $\square$

Definition 2.6. Let ψ be as in Proposition 2.3 (iii), i. e. $\psi \in L^1(\mathbb{R}^d)$ with $\psi \geq 0$ and $\int_{\mathbb{R}^d} \psi(x) \,\mathrm{d}x = 1$. If additionally $\psi \in C_c^\infty(\mathbb{R}^d)$, then ψ is called *(positive) mollifier*.

Proposition 2.7. *Let $\Omega \subset \mathbb{R}^d$ be open, $1 \leq p < \infty$ and ψ be a mollifier. For all $f \in L^p(\Omega)$ and all $\delta > 0$, there is $\epsilon > 0$ such that*

$$\|\psi_\epsilon \star f - f\|_{L^p(\Omega)} < \delta.$$

Moreover, $C_c^\infty(\Omega)$ is dense in $L^p(\Omega)$.

Proof. Since $C_c(\Omega)$ is dense in $L^p(\Omega)$ (cf. Proposition B.8), there is a $g \in C_c(\Omega)$ with $\|f-g\|_{L^p(\Omega)} < \frac{\delta}{3}$. Then, extending f, g to $\mathbb{R}^d$ with 0, using the Minkowski inequality (cf. Remark B.4) and Proposition 2.3 (i) with $r = p$ and $q = 1$, we get

$$\begin{aligned}\|\psi_\epsilon \star f - f\|_{L^p(\Omega)} &\leq \|\psi_\epsilon \star f - \psi_\epsilon \star g\|_{L^p(\Omega)} + \|\psi_\epsilon \star g - g\|_{L^p(\Omega)} + \|g - f\|_{L^p(\Omega)} \\ &< \|(f-g) \star \psi_\epsilon\|_{L^p(\mathbb{R}^d)} + \|\psi_\epsilon \star g - g\|_{L^p(\Omega)} + \frac{\delta}{3} \\ &\leq \|f-g\|_{L^p(\mathbb{R}^d)} \|\psi_\epsilon\|_{L^1(\mathbb{R}^d)} + \|\psi_\epsilon \star g - g\|_{L^p(\Omega)} + \frac{\delta}{3} \\ &< \|\psi_\epsilon \star g - g\|_{L^p(\Omega)} + \frac{2\delta}{3}.\end{aligned}$$

Since $g \in C_c(\Omega)$ and we extended g with zero, we get $g \in C_c(\mathbb{R}^d)$. Moreover, there is a $r > 0$ such that $K := \overline{B_r(0) + \operatorname{supp}(g)} \subset \Omega$ where $A + B = \{a + b \,:\, a \in A \wedge b \in B\}$ and supp is the closed support (cf. Definition B.1). Since ψ has compact support and by construction of ψ_ϵ, for all $\epsilon > 0$ small enough, we have $\operatorname{supp}(\psi_\epsilon) \subset B_r(0)$ and thus

$$\operatorname{supp}(\psi_\epsilon \star g) \subset \overline{\operatorname{supp}(\psi_\epsilon) + \operatorname{supp}(g)} \subset K \subset \Omega.$$

Since g is continuous and has compact support, it is also uniformly continuous and bounded. Thus, by Proposition 2.3 (iii), $\psi_\epsilon \star g$ converges uniformly to g on K. Moreover, $\psi_\epsilon \star g$ and g are zero on $\mathbb{R}^d \setminus K$. Thus, there is $\epsilon > 0$ such that $\|\psi_\epsilon \star g - g\|_{L^\infty(K)} < \frac{\delta}{3\|1\|_{L^p(K)}}$. With this, we get

$$\|\psi_\epsilon \star g - g\|_{L^p(\Omega)} = \|\psi_\epsilon \star g - g\|_{L^p(K)} \leq \|\psi_\epsilon \star g - g\|_{L^\infty(K)} \|1\|_{L^p(K)} < \frac{\delta}{3},$$

which shows the first statement. Moreover, we have

$$\|\psi_\epsilon \star g - f\|_{L^p(\Omega)} \leq \|\psi_\epsilon \star g - g\|_{L^p(\Omega)} + \|g - f\|_{L^p(\Omega)} < \frac{2\delta}{3} < \delta.$$

By Proposition 2.3 (ii), we get $\psi_\epsilon \star g \in C^\infty(\mathbb{R}^d)$. Finally, as shown above, we can choose $\epsilon > 0$ so small that additionally $\operatorname{supp}(\psi_\epsilon \star g) \subset \Omega$. Thus, we have $\psi_\epsilon \star g \in C_c^\infty(\Omega)$, which shows the claimed denseness. □

Remark 2.8 (Discrete filter). If $F = (f_i)_{i\in\mathbb{Z}} \in \mathbb{R}^{\mathbb{Z}}$ is a signal and $W = (w_{-1}, w_0, w_1) \in \mathbb{R}^3$, we consider the weighted sum

$$\hat{f}_i = \sum_{j=-1}^{1} w_j f_{i+j}.$$

This corresponds to the discrete correlation of F and W in 1D and also holds accordingly for more general filters $W = (w_{-a}, \dots, w_a) \in \mathbb{R}^{2a+1}$ for $a \in \mathbb{N}_0$. However, one has to take into account that discrete signals $F = (f_i)_{i\in\{1,\dots,m\}}$ only have finitely many components, but the discrete filter needs f_{i+j} for all $i \in \{1, \dots, m\}$ and $j \in \{-1, 0, 1\}$ or $j \in \{-a, \dots, a\}$, respectively. There are different ways to define f_i for $i \in \mathbb{Z} \setminus \{1, \dots, m\}$:

- Zero extension (or zero padding), i. e. $f_i = 0$ for $i \in \mathbb{Z} \setminus \{1, \dots, m\}$:

$$\dots, 0, 0, 0, f_1, f_2, \dots, f_{m-1}, f_m, 0, 0, 0, \dots$$

- Extension by mirroring at the boundary, i.e. $f_i = f_{2m-i}$, $i \in \{m+1, \ldots, 2m-1\}$ (analogously for the left boundary):

$$\ldots, f_4, f_3, f_2, f_1, f_2, \ldots, f_{m-1}, f_m, f_{m-1}, f_{m-2}, f_{m-3}, \ldots$$

- Periodic extension, i.e. $f_i = f_{((i-1) \bmod m)+1}$ for $i > m$ (analogously for $i < 1$):

$$\ldots, f_{m-2}, f_{m-1}, f_m, f_1, f_2, \ldots, f_{m-1}, f_m, f_1, f_2, f_3, \ldots$$

- Constant extension, i.e. $f_i = f_1$ for $i < 1$ and $f_i = f_m$ for $i > m$:

$$\ldots, f_1, f_1, f_1, f_1, f_2, \ldots, f_{m-1}, f_m, f_m, f_m, f_m, \ldots$$

This can be extended to discrete images in 2D in a straightforward manner:

Definition 2.9 (Discrete cross-correlation and convolution)**.** If $F = (f_{i,j})_{(i,j)\in\mathcal{I}}$ is a discrete image and $W = (w_{k,l})_{(k,l)\in\mathcal{J}}$ with $\mathcal{J} := \{(k,l) \in \mathbb{Z}^2 : -a \le k \le a, -b \le l \le b\}$ for $a, b \in \mathbb{N}_0$, the *discrete cross-correlation* is defined as

$$(W \star F)_{i,j} = \sum_{k=-a}^{a} \sum_{l=-b}^{b} w_{k,l} f_{i+k,j+l} \text{ for } (i,j) \in \mathcal{I},$$

where $f_{i+k,j+l}$ for $(i+k, j+l) \notin \mathcal{I}$ is defined according to one of the extensions discussed above. W is called *(filter) kernel* or *(filter) mask* of size $(2a+1) \times (2b+1)$. The mapping

$$(W * F)_{i,j} = \sum_{k=-a}^{a} \sum_{l=-b}^{b} w_{k,l} f_{i-k,j-l} \text{ for } (i,j) \in \mathcal{I},$$

is called *discrete convolution*.

When explicitly specifying all kernel values, we use the notation

$$W = \begin{bmatrix} w_{-a,b} & \cdots & w_{a,b} \\ \vdots & w_{0,0} & \vdots \\ w_{-a,-b} & \cdots & w_{a,-b} \end{bmatrix},$$

which is following the indexing from the figure in Definition 1.1.

Remark 2.10. $(W \star F)$ is linear in F. Thus, filtering with W can be expressed as matrix. To this end, F has to be interpreted as vector, e.g. using *reflected lexicographical* order (lexicographical order: $(a_1, a_2) < (b_1, b_2)$ if $a_1 < b_1$ or $a_1 = b_1 \wedge a_2 < b_2$), i.e.

$$F = \left(\begin{array}{c} f_{1,1} \\ \vdots \\ f_{m_1,1} \\ \hline f_{1,2} \\ \vdots \\ f_{m_1,2} \\ \hline \vdots \\ \hline f_{1,m_2} \\ \vdots \\ f_{m_1,m_2} \end{array}\right) = \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_{m_2} \end{pmatrix}, \text{ where } F_j = \begin{pmatrix} f_{1,j} \\ \vdots \\ f_{m_1,j} \end{pmatrix}.$$

For $W = \begin{bmatrix} w_{-1,1} & w_{0,1} & w_{1,1} \\ w_{-1,0} & w_{0,0} & w_{1,0} \\ w_{-1,-1} & w_{0,-1} & w_{1,-1} \end{bmatrix}$ and zero padding, we get

$$(W \star F) = \begin{pmatrix} W_0 & W_1 & & \\ W_{-1} & W_0 & W_1 & \\ & \ddots & \ddots & \ddots \\ & & W_{-1} & W_0 \end{pmatrix} \begin{pmatrix} F_1 \\ F_2 \\ \vdots \\ F_{m_2} \end{pmatrix}, \text{ where } W_j = \begin{pmatrix} w_{0,j} & w_{1,j} & & \\ w_{-1,j} & w_{0,j} & w_{1,j} & \\ & \ddots & \ddots & \ddots \\ & & w_{-1,j} & w_{0,j} \end{pmatrix}.$$

Remark 2.11 (Denoising filters).

(i) Mean value filters: The moving average over squares M_r^∞ can be expressed straightforwardly as discrete filter. For an edge length of the square of 3 or 5 pixels, we get the filter kernels

$$M_3 = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} \text{ and } M_5 = \frac{1}{25} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

of size 3×3 and 5×5, respectively. In general, for a filter size $a \times b$, the corresponding mean value filter is

$$M_{a,b} = \frac{1}{ab} \underbrace{(1, \dots, 1)}_{\in \mathbb{R}^a} \otimes \underbrace{(1, \dots, 1)}_{\in \mathbb{R}^b} \in \mathbb{R}^{a \times b}.$$

Here, $\otimes : \mathbb{R}^m \times \mathbb{R}^n \to \mathbb{R}^{m \times n}, ((x_i)_i, (y_j)_j) \mapsto (x_i y_j)_{ij}$ denotes the *tensor product.*

Let f be a pixelated image (with zero padding) and F the corresponding discrete image on a cartesian grid with grid size 1. Then, for $a \in \mathbb{N}$, we have

$$M^\infty_{a+\frac{1}{2}} f(x^{\underline{j}}) = (M_{2a+1,2a+1} \star F)_{\underline{j}}.$$

Further mean value filter examples:

$$\frac{1}{5} \begin{bmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \quad \frac{1}{10} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

Mean value filters fulfill $\sum_{k,l} w_{k,l} = 1$.

(ii) Gaussian filter: From Proposition 2.3, we know that $\psi \star f$ is at least as smooth as the kernel ψ. Thus, for smoothing, the *Gaussian filter* with filter width $\sigma > 0$

$$g_\sigma(x) = \frac{1}{\left(\sqrt{2\pi}\sigma\right)^d} \exp\left(\frac{-\|x\|^2}{2\sigma^2}\right)$$

suggests itself, since $g_\sigma \in C^\infty(\mathbb{R}^d)$. The simplest way to discretize this (continuous) filter is to first evaluate g_σ on a discrete grid and then to normalize the result, i. e.

$$(\tilde{G}_\sigma)_{k,l} = \frac{1}{\left(\sqrt{2\pi}\sigma\right)^2} \exp\left(\frac{-(k^2 + l^2)}{2\sigma^2}\right), \qquad G_\sigma = \frac{1}{\sum_{k,l} (\tilde{G}_\sigma)_{k,l}} \tilde{G}_\sigma.$$

This discretization assumes a cartesian grid with grid size 1. Otherwise, $(k^2 + l^2)$ has to be scaled corresponding to the actual grid size.

(iii) Binomial filter: We consider the following construction:

$$
\begin{aligned}
B^0 &= (1), \quad W = \frac{1}{2}(1,1) \\
B^1 &= W \star (0, B^0, 0) = \frac{1}{2}(1,1) \\
B^2 &= W \star (0, B^1, 0) = \frac{1}{4}(1,2,1) \\
B^3 &= W \star (0, B^2, 0) = \frac{1}{8}(1,3,3,1) \\
B^4 &= W \star (0, B^3, 0) = \frac{1}{16}(1,4,6,4,1) \\
&\vdots \\
B^n &= W \star (0, B^{n-1}, 0) \in \mathbb{R}^{n+1}
\end{aligned}
$$

Here, we define the correlation of $f \in \mathbb{R}^n$ with $W = (w_0, w_1)$, a filter of size 2, as

$$
(W \star f)_{i=1}^{n-1} := \sum_{k=0}^{1} w_k f_{i+k}.
$$

The resulting vectors match the normalized rows of Pascal's triangle. Thus, the corresponding filter is called *binomial filter*. The binomial filter of size $a \times b$ is

$$
B_{a,b} = B^{a-1} \otimes B^{b-1}.
$$

In particular, for size 3×3 and 5×5, we get

$$
B_3 = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \text{ and } B_5 = \frac{1}{256}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix}.
$$

For large n, the binomial filters are a good approximation of the Gaussian filter.

(iv) Duto blur: The idea of the *duto blur* is to overlay an image with a smoothed version of itself, i. e. one considers

$$
\lambda F + (1 - \lambda)(G_\sigma \star F)
$$

for $\lambda \in [0, 1]$. The result seems to preserve the sharpness of the input image, while still being blurred.

(v) Median filter: Let $a \in \mathbb{N}$. The mean value filter is the solution of a minimization problem:

$$
(M_{2a+1} \star F)_{i,j} = \frac{1}{(2a+1)^2} \sum_{k=-a}^{a} \sum_{l=-a}^{a} f_{i+k,j+l} = \operatorname*{argmin}_{f \in \mathbb{R}} \sum_{k=-a}^{a} \sum_{l=-a}^{a} |f_{i+k,j+l} - f|^2
$$

This follows from the necessary condition

$$0 \stackrel{!}{=} \frac{\mathrm{d}}{\mathrm{d}f}\left(\sum_{k=-a}^{a}\sum_{l=-a}^{a}|f_{i+k,j+l}-f|^2\right) = 2\sum_{k=-a}^{a}\sum_{l=-a}^{a}(f - f_{i+k,j+l})$$

and the fact that the second derivative of the objective function is constant and positive. Due to the square, the minimization problem is susceptible to outliers in F. This can be fixed by replacing the square with the absolute value. Thus, one has to solve a minimization problem of type

$$\min_{g\in\mathbb{R}}\sum_{i=1}^{n}|g_i - g|\,,$$

where $G = (g_i)_i \in \mathbb{R}^n$ and n odd. The minimizer is known to be the median of the values $g_1, \ldots, g_n$ and can be computed by sorting the vector G: If $\tilde{G}$ contains the entries of G in ascending order, i. e. $\tilde{g}_1 \leq \ldots \leq \tilde{g}_n$, the median is the element in the middle, i. e. $\tilde{g}_{\frac{n+1}{2}}$ (in case n is even, the median is not unique).

Thus, the *median filter* of an image F is defined as the computation of the median over windows of size $(2a+1)\times(2a+1)$. This is a nonlinear filter that is well suited to remove outlier pixels, e. g. salt-and-pepper noise.

If, instead of the median, the minimum or maximum is considered, the corresponding filters are called *minimum filter* and *maximum filter*. These three nonlinear filters are all referred to as *rank filters*.

(vi) Bilateral filter: Using the substitution rule for multiple variables, the application of a linear filter can be expressed as follows:

$$(\psi \star f)(x) = \int_{\mathbb{R}^d}\psi(y-x)f(y)\,\mathrm{d}y$$

Thus, the weight at the position y only depends on the distance of x to y. In particular, the difference of the gray values $f(x)$ and $f(y)$ does not matter. Thus, close to an edge in image intensity, gray values on both sides of the edge are averaged leading to the blurring of edges typical for linear filters. This can be prevented if the gray value distance is taken into account:

$$B_{\psi_1,\psi_2}f(x) = \int_{\mathbb{R}^d}\psi_1(y-x)\psi_2(f(y)-f(x))f(y)\,\mathrm{d}y \Bigg/ \int_{\mathbb{R}^d}\psi_1(y-x)\psi_2(f(y)-f(x))\,\mathrm{d}y$$

Here, $\psi_1 : \mathbb{R}^d \to \mathbb{R}$ and $\psi_2 : \mathbb{R} \to \mathbb{R}$ are weight functions. The denominator is necessary to normalize the adaptive weighting. B_{ψ_1,ψ_2} is called *bilateral filter*. If Gaussian kernels are used as weight functions, the filter is also called *nonlinear* or *selective Gaussian filter*.

This filter can be discretized similarly to the correlation:

$$\sum_{k=-a}^{a}\sum_{l=-b}^{b}\psi_1(k,l)\psi_2(f_{i+k,j+l}-f_{i,j})f_{i+k,j+l} \Bigg/ \sum_{k=-a}^{a}\sum_{l=-b}^{b}\psi_1(k,l)\psi_2(f_{i+k,j+l}-f_{i,j})$$

Compared to linear filters, the computation is more costly, since the normalization depends on the input image and has be computed for every pixel. Note that it is not necessary to extend the image to apply this filter. Instead, in case $(i+k, j+l) \notin \mathcal{I}$, the corresponding weight can be set to 0 due the adaptive normalization.

Remark 2.12 (Derivative filters). Finite difference based derivatives can be expressed as linear filters. Let F be a discrete image on a cartesian grid. Then, the forward difference quotient at the position i, j in direction x_1 is given by

$$\frac{1}{h}\left(F_{i+1,j} - F_{i,j}\right).$$

This corresponds to filtering with the kernel

$$D_{x_1}^{+} = \frac{1}{h}\begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

Correspondingly, the backward difference quotient and the quotients in direction x_2 are

$$D_{x_1}^{-} = \frac{1}{h}\begin{bmatrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad D_{x_2}^{+} = \frac{1}{h}\begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad D_{x_2}^{-} = \frac{1}{h}\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix}.$$

Also the central difference quotient can be expressed as linear filter:

$$D_{x_1}^{c} = \frac{1}{2h}\begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad D_{x_2}^{c} = \frac{1}{2h}\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{bmatrix}.$$

The same is true for higher derivatives:

$$D_{x_1}^{2} = \frac{1}{h^2}\begin{bmatrix} 0 & 0 & 0 \\ 1 & -2 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \quad D_{x_2}^{2} = \frac{1}{h^2}\begin{bmatrix} 0 & 1 & 0 \\ 0 & -2 & 0 \\ 0 & 1 & 0 \end{bmatrix}.$$

For derivative filters, we have $\sum_{k,l} w_{k,l} = 0$, whereas averaging filters fulfill $\sum_{k,l} w_{k,l} = 1$.

Note that the rows / columns containing only zeros are not required to represent the difference quotients as linear filters. They were just added to have square filter kernels. Omitting these zeros, leads to non-square filter kernels, e. g.

$$D_{x_1}^{+} = \frac{1}{h}\begin{bmatrix} 0 & -1 & 1 \end{bmatrix} \text{ and } D_{x_2}^{+} = \frac{1}{h}\begin{bmatrix} 1 \\ -1 \\ 0 \end{bmatrix}.$$

The difference quotients do not benefit from the properties of the correlation (Proposition 2.3), but have the problems typical for finite differences. For instance, consider the oscillating signal

$$\ldots, -1, 1, -1, 1, -1, 1, \ldots$$

The central difference quotient of this signal is zero everywhere, but the signal is not constant. Instead of using difference quotients, it is better to make use of the properties from Proposition 2.3.

Remark 2.13 (Prewitt and Sobel edge detector). Despite their limitations, difference quotients can be used for basic edge detection. To compensate for the problems of difference quotients, they are combined with averaging. More precisely, a difference quotient in one coordinate direction is combined with averaging in the other coordinate direction. If the mean value filter is used for the averaging, this leads to the *Prewitt kernels*

$$D_{x_1}^{\text{Prewitt}} = \frac{1}{2h}(-1,0,1) \otimes \frac{1}{3}(1,1,1) = \frac{1}{6h}\begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix},$$

$$D_{x_2}^{\text{Prewitt}} = \frac{1}{3}(1,1,1) \otimes \frac{1}{2h}(-1,0,1) = \frac{1}{6h}\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}.$$

If the binomial filter $B^2 = \frac{1}{4}(1,2,1)$ is used for the averaging instead, this leads to the *Sobel kernels*

$$D_{x_1}^{\text{Sobel}} = \frac{1}{2h}(-1,0,1) \otimes \frac{1}{4}(1,2,1) = \frac{1}{8h}\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix},$$

$$D_{x_2}^{\text{Sobel}} = \frac{1}{4}(1,2,1) \otimes \frac{1}{2h}(-1,0,1) = \frac{1}{8h}\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}.$$

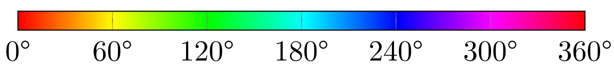
Let F be a discrete image, $G_{x_1} = D_{x_1}^{\text{Prewitt}} \star F$ and $G_{x_2} = D_{x_2}^{\text{Prewitt}} \star F$. Then, $\sqrt{G_{x_1}^2 + G_{x_2}^2}$ is the gradient magnitude, i. e. a simple edge detector. Here, the squares and the square root are computed pixel-wise. Moreover, $\Theta = \operatorname{arctan2}(G_{x_1}, G_{x_2})$ are the gradient directions. Here, arctan2 is the *2-argument arctangent*, i. e.

$$\operatorname{arctan2} : \mathbb{R}^2 \setminus \{0\} \to (-\pi, \pi], (a,b) \mapsto \begin{cases} \arctan\left(\frac{b}{a}\right) & a > 0 \\ \arctan\left(\frac{b}{a}\right) + \pi & a < 0,\, b \geq 0 \\ \arctan\left(\frac{b}{a}\right) - \pi & a < 0,\, b < 0 \\ \frac{\pi}{2} & a = 0,\, b > 0 \\ -\frac{\pi}{2} & a = 0,\, b < 0. \end{cases}$$

The same is true if the Sobel kernels are used instead of the Prewitt kernels.

Remark 2.14. Consider an angle image Θ, i. e. a discrete image with $V = [-\pi, \pi]$. While the value range technically is an interval, it intends to represent the unit circle $S^1 = \left\{x \in \mathbb{R}^2 \,:\, \|x\|_2 = 1\right\}$, i. e. an angle α encodes the corresponding vector $(\cos(\alpha), \sin(\alpha))$. In particular, two end points of the interval, $-\pi$ and π, are equivalent. When visualizing an angle image, this should be taken into account in order to ensure that angles are visualized similarly if they represent unit vectors that are close to each other. This would not be the case if the interval $[-\pi, \pi]$ is simply mapped to gray values similarly to how we handle values in $[0,1]$ for grayscale images. By its construction, the HSV color model offers a solution for this visualization problem.

Like RGB, HSV consist of three channels, hue (H), saturation (S) and value (V). Unlike RGB, the geometry is not cubic, but cylindrical. The hue is the shade of the color and specified as angle. The following color bar illustrates how angles are mapped to color shades:



The saturation encodes the "colorfulness" and is in $[0,1]$, where zero means colorless, i. e. white, one means full color. Finally, the value encodes the brightness and is also in $[0,1]$, where zero means completely dark, i. e. black, one means full brightness.

In order to visualize Θ, we specify the visualization in HSV space, setting $H = \Theta$ (after mapping $[-\pi, \pi]$ to $[0, 2\pi]$ with $\theta \mapsto (\theta + 2\pi) \bmod (2\pi)$). Setting further $S = V = 1$, we get a full color, full brightness visualization of the angle image Θ. Note that to display an image given in HSV space, it still needs to be converted to RGB. This is done by the mapping

$$[0,2\pi] \times [0,1] \times [0,1] \to [0,1]^3, (H,S,V) \mapsto (R,G,B) := \begin{cases} (V,t,p), & i \in \{0,6\} \\ (q,V,p), & i = 1 \\ (p,V,t), & i = 2 \\ (p,q,V), & i = 3 \\ (t,p,V), & i = 4 \\ (V,p,q), & i = 5 \end{cases},$$

where $i = \lfloor H/\frac{\pi}{3} \rfloor$, $f = \left(H/\frac{\pi}{3} - i\right)$, $p = V \cdot (1 - S)$, $q = V \cdot (1 - S \cdot f)$, $t := V \cdot (1 - S \cdot (1 - f))$.

When used to visualize gradient directions, this approach ignores the gradient magnitude. It is very easy to include this though by using the gradient as saturation, i. e. by setting $S = T^{\text{norm}}(\sqrt{G_{x_1}^2 + G_{x_2}^2})$. This way, areas with small gradient are white. If one prefers have those areas to be black instead, one can additionally has to set $V = T^{\text{norm}}(\sqrt{G_{x_1}^2 + G_{x_2}^2})$.

Remark 2.15. A linear filter can be used as feature extractor. As such, linear filters play an integral role in convolutional neural networks (CNNs). In a convolutional layer, multiple filters of small size, e. g. 3×3 or 7×7, are applied to the input data to get a multi-channel output, one channel for each filter. Only the number of filters and their size is prescribed in such a CNN, the weights (filter values) are learned by the network from training data, which creates feature extractors suitable for the desired task, e. g. classification, segmentation, etc.

Remark 2.16 (Canny edge detector). Edges in images or signals are positions at which the value changes significantly. First, we consider the 1D case, i. e. signals. For a smooth signal f, one can define the position of an edge as the position where the slope f' is locally maximal. In particular, at an edge, we then also have $f'' = 0$.

In general, a signal f is not smooth, but Proposition 2.3 (ii) ensures that $\psi \star f \in C^2$ and $(\psi \star f)' = -\psi' \star f$ for a suitable kernel ψ. Assuming some regularity of f (e. g. Proposition 2.3 (iii)), with $\psi_\sigma(x) = \frac{1}{\sigma}\psi(\frac{x}{\sigma})$ we have

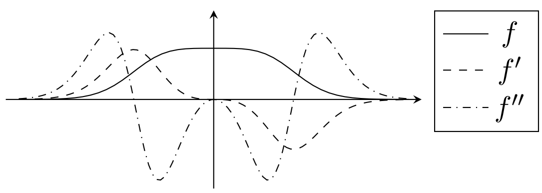
$$\lim_{\sigma \to 0} (\psi_\sigma \star f)(x) = f(x).$$

This gives us a whole family of smoothed images

$$u(x, \sigma) = (\psi_\sigma \star f)(x).$$

Edges exist on different scales, in the sense of rather "coarse" or rather "fine" edges. Thus, a kernel ψ_σ would be desirable for which the amount of edges in $u(\cdot, \sigma)$ scales with σ. The larger σ, the fewer edges. In particular, increasing σ should never add new edges. Moreover, the position of the edges should not move with σ. This leads to the following conditions

$$\begin{aligned} \partial_x^2 u(x,\sigma) > 0 &\Rightarrow \partial_\sigma u(x,\sigma) > 0, \\ \partial_x^2 u(x,\sigma) = 0 &\Rightarrow \partial_\sigma u(x,\sigma) = 0, \\ \partial_x^2 u(x,\sigma) < 0 &\Rightarrow \partial_\sigma u(x,\sigma) < 0. \end{aligned}$$



Solutions of the partial differential equation (PDE)

$$\partial_x^2 u(x,\sigma) = \partial_\sigma u(x,\sigma)$$

fulfill the three properties. In addition to the PDE, we obviously also need

$$u(x,0) = f(x).$$

This initial value problem also occurs in different contexts and models the distribution of heat. Thus, the PDE is called *heat equation.* The unique solution is the convolution with the Gaussian kernel, i. e.

$$u(x,\sigma) = (g_{\sqrt{2\sigma}} * f)(x).$$

The particular "speed" in σ is not important. Thus, also

$$u(x, \tfrac{\sigma^2}{2}) = (g_\sigma * f)(x)$$

is suitable. Thus, $\psi_\sigma = g_\sigma$ is a suitable kernel for edge detection.

In case $d > 1$, $\partial_x^2 u$ is generalized with the *Laplace operator*

$$\Delta_x u(x,\sigma) = \sum_{i=1}^{d} \partial_{x_i}^2 u(x,\sigma).$$

The convolution with $g_{\sqrt{2\sigma}}$ is a solution of the heat equation for all $d \in \mathbb{N}$ (exercise). This convolution can also be expressed as correlation:

$$(g_{\sqrt{2\sigma}} * f)(x) = \int_{\mathbb{R}^d} g_{\sqrt{2\sigma}}(y) f(x-y)\,\mathrm{d}y \overset{\phi(y)=-y}{=} \int_{\mathbb{R}^d} g_{\sqrt{2\sigma}}(-y) f(x+y)\,\mathrm{d}y = (g_{\sqrt{2\sigma}} \star f)(x).$$

The *Canny edge detector* now works as follows: For a given $\sigma > 0$, size and direction of the gradient of $g_\sigma \star f$ are computed, i. e.

$$\begin{aligned}\rho(x) &= \|\nabla(g_\sigma \star f)(x)\|_2 = \sqrt{(\partial_{x_1}(g_\sigma \star f)(x))^2 + (\partial_{x_2}(g_\sigma \star f)(x))^2}\\ \theta(x) &= \operatorname{arctan2}\left(\partial_{x_1}(g_\sigma \star f)(x), \partial_{x_2}(g_\sigma \star f)(x)\right).\end{aligned}$$

Here, the derivatives are computed using

$$\partial_{x_i}(g_\sigma \star f)(x) = (-\partial_{x_i} g_\sigma \star f)(x).$$

It is sufficient to precompute the corresponding derivative filters and then to filter the image with those.

As edges, we consider all points x at which $\rho(x)$ has a strict local maximum in direction $(\sin(\theta(x)), \cos(\theta(x)))$. Discretely (assuming a cartesian grid), θ is rounded to the grid directions $0°, 45°, 90°, \ldots, 315°$ and one checks whether ρ is strictly bigger at the pixel x than at the two neighboring pixels in direction θ and $180° + \theta$. Finally, all edges are disregarded for which $\rho(x)$ is smaller than a specified threshold.

Remark 2.17 (Laplacian sharpening). We want to sharpen an unsharp image. First, we consider the 1D case. Sharpening means emphasizing edges, which are zeros of the second derivative. If one subtracts from a signal f its second derivative f'', the edges in f are amplified (cf. construction in Remark 2.16). Thus, we consider

$$f - \tau f''$$

for $\tau > 0$. The bigger τ, the more the edges are amplified. In the general case, edges occur in different directions. Thus, we need a rotationally invariant classification of the second derivative. The simplest approach is the Laplace operator. Since images are usually not smooth enough, we again use the Gaussian filter and obtain

$$g_\sigma \star f - \tau\Delta(g_\sigma \star f) = (g_\sigma - \tau\Delta g_\sigma) \star f.$$

This operation is called *Laplacian sharpening*.

2.1. Morphological filters

Example 2.18 (Denoising of objects). Let $A \subset \Omega$ be a noisy object that we want to denoise. Noisy here means that there are small, isolated areas in A that don't belong to the original object.

So far, we have considered image denoising. Thus, it seems to suggest itself to consider the characteristic function χ_A instead of the object itself. This way, we could apply our image denoising techniques, but would get an image whose values are not necessarily in $\{0, 1\}$ anymore. Thus, we use a different strategy.

If x belongs to a noise area, we have $\chi_A(x) = 1$, but in the "proximity" of x, there are points y with $\chi_A(y) = 0$. Therefore, we consider a so-called structuring element $B \subset \mathbb{R}^d$ that is slightly larger than the individual noise areas (and usually contains 0). Then, the function

$$f(x) = \begin{cases} 1 & \text{if } \chi_A(x+y) = 1 \text{ for all } y \in B \\ 0 & \text{else} \end{cases}$$

is free of the noisy areas. However, it describes a "thinner" object than the sought one since this operation also removes the boundary areas of the object. This is (approximately) undone by the function

$$g(x) = \begin{cases} 1 & \text{if there is a } y \in B \text{ with } f(x-y) = 1 \\ 0 & \text{else} \end{cases},$$

but noisy areas that are not removed completely in the first step are amplified again. Note that the minus-sign in g is relevant, in case B is not symmetric about 0. Overall, this is a simple but quite viable method for object denoising.

The two operation we have used are called erosion and dilation:

Definition 2.19. Let $f : \mathbb{R}^d \to \{0, 1\}$ and $B \subset \mathbb{R}^d$ nonempty. Then,

$$(f \oplus B)(x) = \begin{cases} 1 & \text{if there is } y \in B \text{ with } f(x+y) = 1 \\ 0 & \text{else} \end{cases}$$

is called *dilation* of f with the *structuring element* B.

$$(f \ominus B)(x) = \begin{cases} 1 & \text{if } f(x+y) = 1 \text{ for all } y \in B \\ 0 & \text{else} \end{cases}$$

is called *erosion* of f with the structuring element B.

Extending this definition to grayscale images requires a different viewpoint:

Lemma 2.20. *Let $f : \mathbb{R}^d \to \{0, 1\}$ and $B \subset \mathbb{R}^d$ nonempty. Then,*

$$(f \oplus B)(x) = \sup_{y \in B} f(x+y) \text{ and } (f \ominus B)(x) = \inf_{y \in B} f(x+y).$$

Proof. Follows immediately from the definition of $\oplus$ and $\ominus$. □

This allows us to define $\oplus$ and $\ominus$ also for grayscale images:

Definition 2.21. Let $f : \mathbb{R}^d \to [0, 1]$ be an image and $B \subset \mathbb{R}^d$ nonempty. Then,

$$(f \oplus B)(x) = \sup_{y \in B} f(x+y)$$

is called *dilation* of the image f with the structuring element B.

$$(f \ominus B)(x) = \inf_{y \in B} f(x+y)$$

is called *Erosion* of the image f with the structuring element B.

Erosion and dilation of an image coincide with the minimum and maximum filter mentioned in Remark 2.11. There, we only introduced these filters in the discrete setting with a rectangle as structuring element.

As we have seen in Example 2.18, it may be helpful to combine $\oplus$ and $\ominus$. The two canonical combinations have special names:

Definition 2.22. Let $f : \mathbb{R}^d \to [0, 1]$ be an image and $B \subset \mathbb{R}^d$ nonempty. Moreover, let

$$-B \coloneqq \{-y \,:\, y \in B\}.$$

Then, the operations

$$f \circ B = (f \ominus B) \oplus (-B) \text{ and } f \bullet B = (f \oplus B) \ominus (-B)$$

are called *opening* and *closing* of f with B.

Example 2.23 (Segmentation with background equalization). The morphological operations can be used as pre-processing step to improve the results of the isodata algorithm from Example 1.13.

If f an image with bright writing on a dark background, $f \circ B$ is an approximation of the background (the erosion removes the thin, bright writing, the dilation corrects the shrinkage caused by the erosion). If one subtracts the estimated background from the image, i. e. considers $f - f \circ B$, brightness variations in the background and small “smudges” are compensated. In case of the blackboard image, the result is improved noticeably if the isodata algorithm is applied to $f - f \circ B$ instead of f. $f - f \circ B$ is called *white-top-hat transform of f*. The analog for closing is $f \bullet B - f$ and called *black-top-hat transform*.

In case of dark writing on a bright background, either the black-top-hat transform is used or the gray values are inverted before the white-top-hat transform is applied. This connection, i. e.

$$f \bullet B - f = -((-f) \circ B) - f = (-f) - ((-f) \circ B),$$

follows from the so-called duality of opening and closing. In addition to this, there are more useful properties.

Remark 2.24 (Properties of the morphological operators). Let $f, g : \mathbb{R}^d \to [0,1]$ and $B \subset \mathbb{R}^d$ nonempty. $\ominus$ and $\oplus$ have the following properties (cf. [1, Theorem 3.29]):

$$\begin{aligned}
\text{duality} \quad & -(f \oplus B) = (-f) \ominus B \\
\text{translational invariance} \quad & (T_h f) \ominus B = T_h(f \ominus B) \\
& (T_h f) \oplus B = T_h(f \oplus B) \\
\text{monotonicity} \quad & f \le g \Rightarrow f \ominus B \le g \ominus B \wedge f \oplus B \le g \oplus B \\
\text{distributivity} \quad & (f \wedge g) \ominus B = (f \ominus B) \wedge (g \ominus B) \\
& (f \vee g) \oplus B = (f \oplus B) \vee (g \oplus B).
\end{aligned}$$

Here, $\vee$ and $\wedge$ denote the pointwise minimum and maximum, respectively, i. e.

$$(f \vee g)(x) \coloneqq \max(f(x), g(x)) \text{ and } (f \wedge g)(x) \coloneqq \min(f(x), g(x))$$

Note that the symbols $\vee$ and $\wedge$ are motivated by the following: For $f = \chi_A$, $g = \chi_B$ with $A, B \subset \mathbb{R}^d$, we have $f \vee g = \chi_{A \cup B}$ and $f \wedge g = \chi_{A \cap B}$.

Except for distributivity, $\circ$ and $\bullet$ also have these properties (cf. [1, Theorem 3.33]):

$$\begin{aligned}
\text{duality} \quad & -(f \bullet B) = (-f) \circ B \\
\text{translational invariance} \quad & (T_h f) \circ B = T_h(f \circ B) \\
& (T_h f) \bullet B = T_h(f \bullet B) \\
\text{monotonicity} \quad & f \le g \Rightarrow f \circ B \le g \circ B \wedge f \bullet B \le g \bullet B
\end{aligned}$$

Moreover, we have

$$\begin{aligned}
\text{non-increasingness} \quad & f \circ B \le f \\
\text{non-decreasingness} \quad & f \le f \bullet B \\
\text{idempotence} \quad & (f \circ B) \circ B = f \circ B \\
& (f \bullet B) \bullet B = f \bullet B.
\end{aligned}$$

Remark 2.25. As Lemma 2.26 will show, the distributivity of $\ominus$ and $\oplus$ still holds when we consider an infinite number of functions. Moreover, dilation and erosion are the only translational invariant operators on binary images that fulfill this generalized distributivity. This will be formalized and shown in Proposition 2.27.

Lemma 2.26 (Infinite distributive law for erosion and dilation). *Let $(f_i)_{i \in I} \subset F(\mathbb{R}^d, [0,1])$ be a set of images indexed by a set I and $B \subset \mathbb{R}^d$ nonempty. Then,*

$$\Big(\bigvee\nolimits_i f_i\Big) \oplus B = \bigvee\nolimits_i (f_i \oplus B) \text{ and } \Big(\bigwedge\nolimits_i f_i\Big) \ominus B = \bigwedge\nolimits_i (f_i \ominus B),$$

where

$$\left(\bigvee_i f_i\right)(x) \coloneqq \sup_i f_i(x) \text{ and } \left(\bigwedge_i f_i\right)(x) \coloneqq \inf_i f_i(x)$$

Here, $F(A,B) \coloneqq \{f : A \to B\}$ denotes the set of all mappings from A to B.

Proof. Let $x \in \mathbb{R}^d$. Then,

$$\begin{aligned}\left(\left(\bigvee_i f_i\right) \oplus B\right)(x) &= \sup_{y\in B}\left(\bigvee_i f_i\right)(x+y) = \sup_{y\in B}\sup_i f_i(x+y) = \sup_i \sup_{y\in B} f_i(x+y)\\ &= \sup_i (f_i \oplus B)(x) = \left(\bigvee_i (f_i \oplus B)\right)(x).\end{aligned}$$

The second statement is shown analogously since also infimums can be interchanged. □

Proposition 2.27. *Let D be a translational invariant operator on binary images, i. e.*

$$D : F(\mathbb{R}^d, \{0,1\}) \to F(\mathbb{R}^d, \{0,1\})$$

with $D(T_h f) = T_h D(f)$ for all $h \in \mathbb{R}^d$ and $f \in F(\mathbb{R}^d, \{0,1\})$. If $D(0) = 0$, $D(1) \neq 0$ and

$$D\left(\bigvee_i f_i\right) = \bigvee_i (D(f_i))$$

for all sets of images $(f_i)_{i\in I} \subset F(\mathbb{R}^d, \{0,1\})$, then there is a nonempty set $B \subset \mathbb{R}^d$ such that

$$D(f) = f \oplus B \text{ for all } f \in F(\mathbb{R}^d, \{0,1\}).$$

If $D(1) = 1$, $D(0) \neq 1$ and

$$D\left(\bigwedge_i f_i\right) = \bigwedge_i (D(f_i))$$

for all sets of images $(f_i)_{i\in I} \subset F(\mathbb{R}^d, \{0,1\})$, then there is a nonempty set $B \subset \mathbb{R}^d$ such that

$$D(f) = f \ominus B \text{ for all } f \in F(\mathbb{R}^d, \{0,1\}).$$

Proof. Let $f : \mathbb{R}^d \to \{0,1\}$ and $F = \{f = 1\}$. Then, $f = \chi_F$. Moreover, $F = \bigcup_{y\in F}\{y\}$ and thus $f = \bigvee_{y\in F} \chi_{\{y\}}$, if we additionally assume $F \neq \emptyset$[2], i. e. $f \not\equiv 0$. Combined with the assumed distributivity of D, we get

$$D(f) = D\left(\bigvee_{y\in F} \chi_{\{y\}}\right) = \bigvee_{y\in F}\left(D(\chi_{\{y\}})\right).$$

Noting that

$$\chi_{\{y\}}(x) = \begin{cases} 1 & x \in \{y\} \\ 0 & \text{else} \end{cases} = \begin{cases} 1 & x - y \in \{0\} \\ 0 & \text{else} \end{cases} = \chi_{\{0\}}(x-y) = (T_{-y}\chi_{\{0\}})(x)$$

for all $x \in \mathbb{R}^d$, the translational invariance of D implies

$$D(\chi_{\{y\}}) = D(T_{-y}\chi_{\{0\}}) = T_{-y}D(\chi_{\{0\}}).$$

[2]This assumptions is necessary since the supremum over the empty set is $-\infty$.

For $x \in \mathbb{R}^d$, the above leads to

$$\begin{aligned}
D(f)(x) &= \bigvee\nolimits_{y \in F} \left(T_{-y} D(\chi_{\{0\}})\right)(x) \\
&= \begin{cases} 1 & \exists y \in \mathbb{R}^d : y \in F \wedge T_{-y} D(\chi_{\{0\}})(x) = 1 \\ 0 & \text{else} \end{cases} \\
&= \begin{cases} 1 & \exists y \in \mathbb{R}^d : f(y) = 1 \wedge D(\chi_{\{0\}})(x-y) = 1 \\ 0 & \text{else} \end{cases} \\
\underset{(z=y-x)}{} &= \begin{cases} 1 & \exists z \in \mathbb{R}^d : f(x+z) = 1 \wedge D(\chi_{\{0\}})(-z) = 1 \\ 0 & \text{else} \end{cases}.
\end{aligned} \tag{$*$}$$

For $B \coloneqq \{D(\chi_{\{0\}})(-\cdot) = 1\}$, we have $D(\chi_{\{0\}})(-z) = 1 \Leftrightarrow z \in B$. The assumption $D(1) \neq 0$ together with $(*)$ ensures that B is nonempty. Recalling the definition of the dilation of a binary image, i. e.

$$(f \oplus B)(x) = \begin{cases} 1 & \exists y \in B : f(x+y) = 1 \\ 0 & \text{else} \end{cases}$$

we have shown $D(f) = f \oplus B$ for $f \not\equiv 0$. For $f \equiv 0$, the statement follows from the assumption $D(0) = 0$, i. e. $D(0) = 0 = 0 \oplus B$. Thus, we have shown the first claim. The second claim can be shown with the first, using the complement mapping $f \mapsto 1 - f$ (exercise). $\square$

Proposition 2.28 (Contrast invariance of erosion and dilation)**.** *Let $B \subset \mathbb{R}^d$ be nonempty. Then, for all continuous, increasing intensity transformations $T : \mathbb{R} \to \mathbb{R}$ and all $f : \mathbb{R}^d \to [0,1]$,*

$$T(f) \ominus B = T(f \ominus B) \text{ and } T(f) \oplus B = T(f \oplus B)$$

Proof. For $x \in \mathbb{R}^d$, we have

$$(T(f) \ominus B)(x) = \inf_{y \in B} T(f(x+y)) = T(\inf_{y \in B} f(x+y)) = T((f \ominus B)(x)),$$

where we have used that continuous, increasing functions can be interchanged with the infimum (exercise). The same is true for the supremum, which shows the claim for the dilation. $\square$

Remark 2.29*.* Dilation and erosion of grayscale images with a structuring element can be further generalized: For this purpose, the "flat" structuring element is replaced by a function. Let $f : \mathbb{R}^d \to [0,1]$, $B \subset \mathbb{R}^d$ nonempty and $b : B \to \mathbb{R}$. Then, dilation and erosion of the image f with b are defined as

$$(f \oplus b)(x) \coloneqq \sup_{y \in B} \left(f(x+y) + b(y)\right) \text{ and } (f \ominus b)(x) \coloneqq \inf_{y \in B} \left(f(x+y) - b(y)\right).$$

With the constant function $b \equiv 0$, these are equivalent to dilatation and erosion with B. The function b can be thought of as a nonflat structuring element that changes the graph of f before the infimum or supremum is evaluated.