3. Spatial transforms

3.1. Geometric operations
(Reading: Castleman, 1996, pp. 115-138)

- a geometric operation is defined as
  \[ g_o(x, y) = g_i(x', y') = g[ a(x, y), b(x, y) ] \]  
  (3.1)

where \( a() \) and \( b() \) are the transformation equations that map an input image \( g_i \) onto an output image \( g_o \)

3.1.1. Interpolation

- as the pixel coordinates resulting out of most transformations do not coincide with the grid coordinates, the grey value of transformed pixels has to be derived through pixel filling or interpolation; in forward mapping (see Fig. 3.1), the transformed pixel value in the output image is derived from the four input image pixels mapping onto it, such that each pixel’s grey value \( G_o \) is given by the sum of the areal fractions
  \[ G_o(x, y) = \sum_k f_k G_k \]  
  (3.2)

Fig. 3.1: Interpolation schemes for forward mapping (left) and inverse mapping (right, Jähne, 1997).

- a more sophisticated approach consists of deriving the grey values of the pixels in the transformed image through appropriate interpolation in the original image (Fig. 3.1, inverse mapping)
- the simplest interpolation technique, nearest-neighbour or zero-order interpolation, consists in assigning the pixel a grey value that corresponds to that of its nearest neighbour
- bilinear (first-order) interpolation involves fitting a hyperbolic paraboloid such that
  \[ G(x, y) = ax + by + cxy + d \]  
  (3.3)

- from the four corner coordinate pixels (see Fig. 3.2) any inlying point can be derived through sequential interpolation:
  \[ G(x, 0) = G(0, 0) + x[G(1, 0) - G(0, 0)] \]
  \[ G(x, 1) = G(0, 1) + x[G(1, 1) - G(0, 1)] \]
  \[ G(x, y) = G(x, 0) + y[G(x, 1) - G(x, 0)] \]  
  (3.4)

- applications which demand continuous interpolation (also of the derivatives) require spline-based interpolation or more sophisticated algorithms implemented in the frequency domain (for details see, e.g., Jähne, 1997); another option is to represent object outlines in a classified image by splines or other continuous functions
3.1.2. Geometric transforms
(Reading: Schowengerdt, 1997, pp. 331-344)

- affine transforms are linear coordinate transforms that can be broken down into a sequence of basic operations: translation, rotation, scaling, stretching, and shearing; these can be expressed in matrix notation (using homogeneous coordinates in which the x-y plane is represented by z=1 in x-y-z space)

- translation by $x_0$, $y_0$ such that $a(x,y)=x+x_0$, $b(x,y)=y+y_0$:

$$
\begin{bmatrix}
a(x,y) \\
b(x,y) \\
1
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & x_0 \\
0 & 1 & y_0 \\
1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix}
$$

(3.5)

- stretching by $c$ and $d$ such that $a(x,y)=x/c$, $b(x,y)=y/d$:

$$
\begin{bmatrix}
a(x,y) \\
b(x,y) \\
1
\end{bmatrix} =
\begin{bmatrix}
1/c & 0 & 0 \\
0 & 1/d & 0 \\
1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix}
$$

(3.6)

- rotation by the angle $\theta$, such that $a(x,y)=x\cos\theta - y\sin\theta$, $b(x,y)=x\sin\theta + y\cos\theta$:

$$
\begin{bmatrix}
a(x,y) \\
b(x,y) \\
1
\end{bmatrix} =
\begin{bmatrix}
\cos\theta & -\sin\theta & 0 \\
\sin\theta & \cos\theta & 0 \\
1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix}
$$

(3.7)

- a full affine transformation can then be described as the matrix product of its component operations, e.g., for a rotation about a point $x_0,y_0$ other than the origin is given by

$$
\begin{bmatrix}
a(x,y) \\
b(x,y) \\
1
\end{bmatrix} =
\begin{bmatrix}
1 & 0 & x_0 \\
0 & 1 & y_0 \\
1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
\cos\theta & -\sin\theta & 0 \\
\sin\theta & \cos\theta & 0 \\
1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
1
\end{bmatrix}
$$

(3.8)

- for some applications, such as the mapping of ground control points (GCPs) from a distorted satellite image to a reference image (or map projection), non-linear distortion precludes the application of affine transforms; in the case of map projections, the transformation equations are known (see, e.g., Snyder J. P., 1982: Map projections used by the U.S. Geological Survey. *Geol. Surv. Bull., 2nd ed., 1532, 313pp.*) and can be applied accordingly
- rectification of distorted images for which the transformation equations are not known (e.g., a problem that might be encountered in geolocation of satellite imagery or aerial photography based on GCPs), requires a different approach; one such approach consists of deriving a generic polynomial model to relate the image coordinates \( x', y' \) to that of a reference \( x_r, y_r \).

\[
x' = \sum_{i=0}^{N} \sum_{j=0}^{N-1} a_{ij} x_r^i y_r^j
\]
\[
y' = \sum_{i=0}^{N} \sum_{j=0}^{N-1} b_{ij} x_r^i y_r^j
\]

(3.9)

- for most applications 2-order polynomials are sufficient, such that

\[
x' = a_{00} + a_{10} x_r + a_{01} y_r + a_{11} x_r y_r + a_{20} x_r^2 + a_{02} y_r^2
\]
\[
y' = b_{00} + b_{10} x_r + b_{01} y_r + b_{11} x_r y_r + b_{20} x_r^2 + b_{02} y_r^2
\]

(3.10)

- now, for \( M \) pairs of GCPs located in the image and the reference, equation 3.10 can be written for the \( M \) sets of equations in matrix notation

\[
\begin{bmatrix}
  x'_1 \\
  x'_2 \\
  \vdots \\
  x'_M
\end{bmatrix}
= 

\begin{bmatrix}
  1 & x_{r,1} & y_{r,1} & x_{r,1}y_{r,1} & x_{r,1}^2 & y_{r,1}^2 \\
  1 & x_{r,2} & y_{r,2} & x_{r,2}y_{r,2} & x_{r,2}^2 & y_{r,2}^2 \\
  \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
  1 & x_{r,M} & y_{r,M} & x_{r,M}y_{r,M} & x_{r,M}^2 & y_{r,M}^2
\end{bmatrix}
\begin{bmatrix}
  a_{00} \\
  a_{01} \\
  a_{10} \\
  a_{11} \\
  a_{20} \\
  a_{02}
\end{bmatrix}
\]

(3.11)

\[
X = RA
\]

and similarly for the \( y' \) coordinates

- if \( M \) is equal to the number of polynomial coefficients \( K \), \( A \) and \( B \) can be obtained by inverting \( R \):

\[
A = R^{-1}X
\]
\[
B = R^{-1}Y
\]

(3.12)

- for cases where \( M > K \), least-square techniques can be applied to solve for \( A \) and \( B \) (for details see Schowengerdt, 1997, p. 337ff.)

3.1.3. Analysis of stereoscopic images
(Reading: Castleman, 1996, pp. 585-599)

- because the reconstruction of the 3-dimensional topography of image pairs requires geometric transformation of images, it will be shortly discussed at this point
- for two bore-sighted cameras (i.e. with parallel optical axes) as shown in Fig. 3.3, the image of a given point in 3-D space \( X_0, Y_0, Z_0 \) on the image plane of the left camera \( l \) with focal length \( f \) (such that \( Z = -f \) in the image plane), is given by

\[
X_l = -X_0 \frac{f}{Z_0} \quad Y_l = -Y_0 \frac{f}{Z_0}
\]

(3.13)

the corresponding right camera image is then given by a shift \( d \) in the image position relative to the origin which coincides with the lens \( l \):
\( X_r = -(X_0 + d) \frac{f}{Z_0} - d \quad Y_r = -Y_0 \frac{f}{Z_0} \)  

- defining a coordinate system in the image plane that is rotated by 180° relative to the world coordinate system such that \( x_l = -X_r, y_l = -Y_r, x_r = -X_r - d, y_r = -Y_r, \) the image coordinates for the point \( X_0, Y_0, Z_0 \) are:

\[
\begin{align*}
\quad x_l & = X_0 \frac{f}{Z_0} & y_l & = Y_0 \frac{f}{Z_0} \\
\quad x_r & = (X_0 + d) \frac{f}{Z_0} & y_r & = Y_0 \frac{f}{Z_0}
\end{align*}
\]

rearranging and solving for \( Z_0 \) yields:

\[
\begin{align*}
\quad X_0 & = x_l \frac{f}{Z_0} = x_r \frac{Z_0}{f} - d \\
\quad Z_0 & = \frac{f d}{x_r - x_l}
\end{align*}
\]

- this is referred to as the normal-range equation and relates the Z-coordinate (i.e. topography) of any given point to the corresponding dislocation in x-direction in the image pair; as both \( f \) and \( d \) are typically small whereas \( Z_0 \) may be quite large, determination of \( x_r-x_l \) to a high degree of accuracy is crucial

- the true-range equation is derived trigonometrically as

\[
\begin{align*}
\quad \frac{R}{Z_0} & = \frac{\sqrt{f^2 + x_l^2 + y_l^2}}{f} \\
\quad R & = \frac{d \sqrt{f^2 + x_l^2 + y_l^2}}{x_r - x_l}
\end{align*}
\]

which for many systems (with \( X_0, Y_0 << Z_0 \) and \( x_l, y_l << f \)) can be approximated by equation 3.16

- while the derivation of the Z-topography is fairly straightforward (at least for bore-sighted systems), the determination of \( x_r-x_l \) to a sufficient degree of accuracy can be quite a challenge; in most applications the magnitude of the dislocation cannot be derived explicitly but has to determined for an entire image pair based on correlation techniques; some of these aspects will be covered at a later stage; however, a technique explicitly developed for stereo pair matching has been presented by Sun (Digital Image Computing: Techniques and Applications, pp.95-100, Massey University, Auckland, New Zealand, 10-12 December 1997) and the algorithm has been implemented at http://extra.cmis.csiro.au/IA/changs/stereo/;

Fig. 3.3: Camera arrangement for stereoscopic imaging (Castleman, 1996).
3.2. Spatial filtering
(Reading: Gonzalez and Wintz, 1987, pp. 161-179; Castleman, 1996, pp. 154-166; Jähne, 1995, 100ff.)

- while point operations discussed in Section 2 can be thought of as the simplest form of a spatial operation (operating on a neighbourhood of 1,1 for each pixel), in a more general context transformations or filtering in the spatial domain correspond to the transformation of an input image \( g_i(x,y) \) to an output image \( g_o(x,y) \) through convolution with a linear, position-invariant\(^1\) operator (also referred to as mask or convolution kernel) \( h \), such that

\[
g_o(x, y) = h(x, y) * g_i(x, y)
\]

(as outlined in Section 4, this spatial operation corresponds to a multiplication of the Fourier transforms \( H(xu,v) \) and \( G(u,v) \) of the image and the kernel in the frequency domain)

- in its discrete form, the convolution can be represented by the summation over a kernel \( k \) of size \( m,n \) for the product \( k() g() \) of every \( x,y \) of the image pixel coordinates (Fig. 3.4) such that

\[
g_o(x, y) = \frac{1}{C} \sum_{i} \sum_{j} k(m,n) g_i(x + m, y + n)
\]

- the scaling factor \( 1/C \) is typically given as \( \sum \sum k(m,n) \), with kernels mostly odd-sized (e.g., 3x3) with the origin at the center coordinate

\[
\text{Fig. 3.4: Discrete convolution (Castleman, 1996).}
\]

- as will be outlined in more detail in Section 4, the three major types of linear filters can be represented both in the spatial and in the frequency-domain (Fig. 3.5); with lowpass filters mostly applied to smooth images and reduce the impact of impulse noise on image processing and high-pass filters playing an important role in edge and feature detection; in general terms, a convolution can be applied to remove the

\(^1\) An operator \( h \) is linear if its application to a combined set of images \( g_k \) (multiplied by a scalar factor \( a_k \)) yields the same result as the operation on the individual component images:

\[
h\left( \sum_k a_k g_k \right) = \sum_k a_k h(g_k)
\]

which implies that linear operations can be performed on images decomposed into subunits rather than the whole image.

Position- or shift-invariance requires that the result of an operation is the same regardless of the position of the image or the operator, which also aids in breaking down more complex operations into simpler component steps.
effects of its inverse (determined or provided a priori) on the image, such as artifacts resulting out of errors in the imaging system

![Fig. 3.5: Cross-sections of frequency-domain (top) and spatial-domain filters (Gonzalez and Woods, 1992).](image)

### 3.2.1. Smoothing filters

- as shown in Fig. 3.5, lowpass or smoothing filters reduce the high-frequency component of the image; the range of frequencies impacted by the spatial convolution is given by the filter size, with the magnitude of the scaling factor and the shape of kernel \( k(m,n) \) determining the amplitude of the transformation (Fig. 3.6 shows simple low-pass filter kernels of different sizes); the impact of the filter operation can be modulated by, e.g., sampling a Gaussian distribution function (Fig. 3.5a) and modifying its statistical moments

![Fig. 3.6: Kernels for smoothing filters (Gonzalez and Woods, 1992).](image)

- linear convolutions are associated with a (for large kernels substantial) loss of image information; for image enhancement purposes, so-called rank-order filters (non-linear) achieve better results; for rank order filtering, the set of pixels sampled by a mask of size \( m,n \) is ordered and the center pixel (or origin) is transformed to the \( n \)-th element of the sorted list (e.g., median, minimum, maximum); as discussed in more detail in section 3.3, these non-linear filters also form the basis for morphological filtering techniques

![Fig. 3.7: Schematic depiction of 3x3 median (rank-order) filter (Jähne, 1997).](image)
3.2.2. Sharpening filters

- as shown in Fig. 3.5b, the **high-pass filter** selectively operates on the high-frequency component of images; because the integral of the filter function over the entire interval is zero (corresponding to a zero-sum of the discrete filter kernel, see Fig. 3.9); no change occurs in image regions that are of uniform grey value, whereas fine details are "sharpened", e.g., in applications where a remedy for the effects of blurry optics is required; since this operation also enhances the noise component, the weight of the central pixel in the kernel may be increased to values >8 in a 3x3 mask (Fig. 3.9) which is often referred to as high-boost filtering and represents a compromise between sharpening and maintaining image integrity

![Fig. 3.9: Sharpening kernel (Gonzalez and Woods, 1992).](image)

- **derivative filters** are a variant of sharpening filters that respond to spatial grey-value gradients in an image (Fig. 3.10); the gradient vector of a function \( f(x,y) \) is given by

\[
\nabla f = \begin{bmatrix}
\frac{\partial f}{\partial x} \\
\frac{\partial f}{\partial y}
\end{bmatrix}
\]  

(3.20)

with derivative filters based on the determination of the magnitude of this vector

\[
|\nabla f| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}
\]  

(3.21)

- implementation of gradient operators on discrete images can be achieved in several ways (see exemplary pixel neighbourhood and operators in Fig. 3.11); differencing of the absolute values is the most common approach, with, e.g., so-called Roberts cross-gradient operators (corresponding to a 2x2 kernel) based on the approximation

\[
\nabla f \approx |z_5 - z_9| - |z_6 - z_8|
\]  

(3.22)

- for 3x3 kernels, this approximation can be implemented by so-called Prewitt operators (Fig. 3.11) which compute the sum of differences
\[ \nabla f = \left| (z_7 + z_8 + z_9) - (z_1 + z_2 + z_3) \right| + \left| (z_3 + z_6 + z_9) - (z_1 + z_4 + z_7) \right| \]  
(3.23)

- for some applications, the second-order derivative (Laplacian) of an image needs to be computed and can be approximated as shown

\[ \nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \]

\[ \nabla f \approx 4z_5 \left| \left( z_2 + z_4 + z_6 + z_8 \right) \right| \]  
(3.24)

Fig. 3.10: First and second derivative across a boundary in an image (Gonzalez and Woods, 1992).

Fig. 3.11: Image segment (a) and kernels for gradient operators (b to d, for details see text; Gonzalez and Woods, 1992).
3.3. Morphological filtering

- as outlined in the previous section, the convolution of an image with a linear filter mask involves the summation over the individual products of mask and image matrix elements (Fig. 3.4); for non-linear filters this product summation is replaced by a ranking (in rank-order filters such as the median filter shown in Figs. 3.7 and 3.8) or a logical operation (for morphological filters) as shown in Fig. 3.12

![Fig. 3.12: Transformation of an input image with a non-linear, morphological operator (Castleman, 1996).](image)

- rank-order filters can be very powerful operators for image enhancement and extraction of image subsets (two examples will be discussed in class, with the median filter as one of them also shown in Fig. 3.8)

- morphological filters operate on binary images, with each pixel represented by either 0 or 1 as a result of a segmentation, e.g. through an algorithm that distinguishes between a class of relevant features and a background signal; while the examples discussed here and in the class are based on square masks (mostly 3x3, as in Fig. 3.12), morphological filter masks (also called structuring elements) can be of any shape, including composite masks that are broken down into components sequentially operating on the image

- the two basic operations in morphological image processing are the erosion and the dilation; the elimination of all pixels adjacent to the outer margin of a feature in an image is referred to as erosion (assuming here and throughout the text, that features are characterized by 0 and the background is 1 or 255 in binary images); in the terms of set theory, the result $E$ of an erosion is the set of all points $x,y$ such that $S$ translated by $x,y$ is fully contained within the feature set $B$

$$E = B \otimes S = \{x, y | S_{xy} \subseteq B\}$$  \hspace{1cm} (3.25)

- an example of an erosion and other morphological processes is shown in Fig. 3.13; note how connected features end up disjointed if the bridges between the features are smaller than one structuring element diameter

- a dilation integrates all pixels into an object that border on its perimeter; this is achieved by reflecting the structuring element $B$ about its origin and then translating this by $x,y$ across the image, the result of the dilation $D$ is the set of all $x,y$ for which $S'$ and $B$ overlap by at least one element:

$$D = B \oplus S = \{x, y | (S'_{xy} \cap B) \subseteq B\}$$  \hspace{1cm} (3.26)

- sequential erosion and dilation or vice versa are known as opening $O(B)$ and closing $C(B)$, respectively:
\[ O(B) = B \cdot S = (B \odot S) \oplus S \]
\[ C(B) = B \div S = (B \oplus S) \ominus S \]  

(3.26)

Fig. 3.13: Opening through sequential erosion (b,c) and dilation (d,e) and closing through sequential dilation (f,g) and erosion (h,i) of a feature (a) (Gonzalez and Woods, 1992).

- thus erosion results in the elimination of features smaller than the size of the structuring element, separation of features joined by thin bridges and a reduction in the roughness of feature contours; sequential erosion with increasing mask sizes and determination of the corresponding change in total feature/background area is an efficient and robust method of size analysis

- morphological operators, in particular when combined with conditional criteria that, e.g., ensure connectivity between objects, are powerful instruments in the pre-processing and analysis of image data, such as in the case of skeletonization or Euclidean distance transforms (a variant of the binary morphological operation, see Fig. 3.14)

Fig. 3.14: Binarized image (left), euclidean distance map (center, grey value denotes distance from margin of features), skeletonized image (right).