# Formulary
# Image Processing

**Prof. Dr.-Ing. Volker Willert**

thws Technical University of Applied Sciences Würzburg-Schweinfurt

Winter term

### 3D Scene Points & Direction Vectors

- 3D coordinate vector: $\mathbf{X} = [X,\, Y,\, Z]^\top \in \mathbb{R}^3$ • 3D direction vector: $\mathbf{u} = [u_x,\, u_y,\, u_z]^\top \in \mathbb{R}^3$
- homogeneous 3D point: $\overline{\mathbf{X}} = [X,\, Y,\, Z,\, 1]^\top \in \mathbb{R}^4$ • homogeneous 3D vector: $\overline{\mathbf{u}} = [u_x,\, u_y,\, u_z,\, 0]^\top \in \mathbb{R}^4$
- Absolute value: $\|\mathbf{u}\| = \sqrt{\mathbf{u}^\top\mathbf{u}} = \sqrt{u_x^2 + u_y^2 + u_z^2}$ • projection of $\mathbf{v}$ onto $\mathbf{u}$: $\mathbf{v_u} = \frac{\mathbf{u}^\top\mathbf{v}}{\mathbf{u}^\top\mathbf{u}}\mathbf{u}\,, \quad \|\mathbf{v_u}\| = \frac{\mathbf{u}^\top\mathbf{v}}{\|\mathbf{u}\|}$
- scalar product: $\mathbf{u}^\top\mathbf{v} = u_xv_x + u_yv_y + u_zv_z = \cos(\theta)\|\mathbf{u}\|\|\mathbf{v}\| \in \mathbb{R}\,, \quad \mathbf{u}^\top\mathbf{v} = 0\,$, if $\theta = 90°$
- cross product: $\mathbf{u} \times \mathbf{v} = \begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix} \times \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} u_yv_z - u_zv_y \\ u_zv_x - u_xv_z \\ u_xv_y - u_yv_x \end{bmatrix} = \widehat{\mathbf{u}}\mathbf{v} = \begin{bmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \in \mathbb{R}^3$

### 2D image coordinates, straight lines, 2nd order curves & 3D planes

- 2D image point: $\mathbf{x} = [x,\, y]^\top \in \mathbb{R}^2$ • homogeneous 2D image point: $\overline{\mathbf{x}} = [x,\, y,\, 1]^\top \in \mathbb{R}^3$
- homogeneous 2D line equation: $\mathbf{l}^\top\overline{\mathbf{x}} = 0\,, \quad \mathbf{l}^\top = \lambda\,[a,\, b,\, c]\ \forall\,\lambda \neq 0$
- Hessian normal form: $\mathbf{n}^\top\mathbf{x} - d = 0\,,$ distance to origin: $d \geq 0\,,$ normal vector: $\mathbf{n}^\top = [\cos\alpha\,,\, \sin\alpha]$
- intersection of two lines: $\mathbf{l}_1 \times \mathbf{l}_2 = \overline{\mathbf{x}}$ • line from two points: $\overline{\mathbf{x}}_1 \times \overline{\mathbf{x}}_2 = \mathbf{l}$
- parallel lines meet at infinity $\overline{\mathbf{x}}_\infty = [b, -a, 0]^\top$
- 2nd order curves (circles, ellipsoids etc.): $\overline{\mathbf{x}}^\top\mathbf{C}\overline{\mathbf{x}} = 0\,,$ whereas $\mathbf{C} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{12} & c_{22} & c_{23} \\ c_{13} & c_{23} & c_{33} \end{bmatrix} \in \mathbb{R}^{3\times3}$
- A tangent $\mathbf{l}$ to the curve $\mathbf{C}$ at the point $\overline{\mathbf{x}}$ satisfies the equation: $\mathbf{l} = \mathbf{C}\overline{\mathbf{x}}$
- homogeneous 3D plane equation: $\mathbf{e}^\top\overline{\mathbf{X}} = 0\,, \quad \mathbf{e}^\top = \lambda[e_1\,, e_2\,, e_3\,, e_4]\ \forall\,\lambda \neq 0$
- plane defined by 3 points: $\begin{bmatrix} \overline{\mathbf{X}}_1^\top \\ \overline{\mathbf{X}}_2^\top \\ \overline{\mathbf{X}}_3^\top \end{bmatrix} \mathbf{e} = \mathbf{0}$ • intersection of 3 planes: $\begin{bmatrix} \mathbf{e}_1^\top \\ \mathbf{e}_2^\top \\ \mathbf{e}_3^\top \end{bmatrix} \overline{\mathbf{X}} = \mathbf{0}$

### Singular Value Decomposition of a Matrix M

- $\underbrace{\mathbf{M}}_{(m\times n)} = \underbrace{\mathbf{U}}_{(m\times m)}\,\underbrace{\mathbf{S}}_{(m\times n)}\,\underbrace{\mathbf{V}^\top}_{(n\times n)} = \sum_{i=1}^{n} \sigma_i\mathbf{u}_i\mathbf{v}_i^\top = \sigma_1\mathbf{u}_1\mathbf{v}_1^\top + \sigma_2\mathbf{u}_2\mathbf{v}_2^\top + \cdots + \sigma_n\mathbf{u}_n\mathbf{v}_n^\top\,,$ if $m \geq n$

$\mathbf{U} = [\mathbf{u_1},\, \mathbf{u_2}, \ldots,\, \mathbf{u_m}]$ and $\mathbf{V} = [\mathbf{v_1},\, \mathbf{v_2}, \ldots,\, \mathbf{v_n}]$ orthogonal, $\mathbf{S} = \{\sigma_1, \sigma_2, \ldots, \sigma_n\}$ diagonal, $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n$

### Pinhole Camera Model & Radial Distortion

• central projection: $x = -c\frac{X}{Z}\ ,\ y = -c\frac{Y}{Z}$ • projection from world to pixel coordinates: $\lambda\overline{\mathbf{x}}' = \mathbf{\Pi}\overline{\mathbf{X}} = \mathbf{K}\mathbf{\Pi}_0\mathbf{G}\overline{\mathbf{X}}$

• calibration matrix (intrinsics): $\mathbf{K} = \begin{bmatrix} cs_x & cs_\theta & o_x \\ 0 & cs_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$ • standard projection: $\mathbf{\Pi}_0 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$

• extrinsics: $\mathbf{G} = \begin{bmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{bmatrix}$ • rotation matrix: $\mathbf{R} = \begin{bmatrix} \mathbf{r}_1\ ,\ \mathbf{r}_2\ ,\ \mathbf{r}_3 \end{bmatrix}$ • translation vector: $\mathbf{T} = \begin{bmatrix} T_x,\ T_y,\ T_z \end{bmatrix}^\top$

• projection matrix: $\mathbf{\Pi} = \begin{bmatrix} \pi_{11} & \pi_{12} & \pi_{13} & \pi_{14} \\ \pi_{21} & \pi_{22} & \pi_{23} & \pi_{24} \\ \pi_{31} & \pi_{32} & \pi_{33} & \pi_{34} \end{bmatrix} = \mathbf{K}\begin{bmatrix}\mathbf{R}|\mathbf{T}\end{bmatrix}$ • projection center: $\mathbf{o}_{\mathcal{W}} = -\begin{bmatrix}\mathbf{KR}\end{bmatrix}^{-1}\mathbf{KT}$

• projection of a plane: $\lambda\mathbf{x}' = \lambda\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \mathbf{H}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$ • homography: $\mathbf{H} = \mathbf{K}\begin{bmatrix}\mathbf{r}_1\ ,\ \mathbf{r}_2\ ,\ \mathbf{T}\end{bmatrix}$ , if $Z = 0$

• Eliminating radial distortion $\mathbf{x}'_d = (x'_d, y'_d)^\top$ to calculate an undistorted image $\mathbf{x}'_u = (x'_u, y'_u)^\top$:

Forward model: $\mathbf{x}'_u = \mathbf{x}'_d + a_2 r_d^2(\mathbf{x}'_d - \mathbf{c})$ with $r_d = ||\mathbf{x}'_d - \mathbf{c}||$

• normalized coordinates: $\overline{\mathbf{x}} = \mathbf{K}^{-1}\overline{\mathbf{x}}'_u$ (dimensionless!)

### The Lens

• convex lens: $\frac{1}{f} \approx \frac{n-n_0}{n_0}\left(\frac{1}{R_1} + \frac{1}{R_2}\right)$ • focal length: $f \propto \frac{\lambda}{\lambda_0 - \lambda}$ , $\lambda = \frac{\lambda_0}{n} < \lambda_0$ , $\lambda_0 =$ wavelength in vacuum

• working distance: $g = b\frac{G}{B} = \frac{b}{\beta}$

• aspect ratio: $\beta = \frac{B}{G} = \frac{b}{g} = \frac{b-f}{f} \rightarrow \frac{1}{f} = \frac{1}{g} + \frac{1}{b}$ • f-number (aperture): $\kappa = \frac{f}{D}$

• the limits of depth of field $g_h - g_v$ are: $g_v = \frac{gf^2}{f^2 + \kappa\epsilon(g-f)}$ , $g_h = \frac{gf^2}{f^2 - \kappa\epsilon(g-f)} > 0$

• object width $g$, where depth of sharpness $\epsilon$ is optimally small: $g = \frac{2g_h g_v}{g_h + g_v}$

• For a given $\epsilon$ from $g_v(g)$ to $g_h \rightarrow \infty$ the working distance $g$ is: $g \approx \frac{f^2}{\kappa\epsilon}$

• For a given working distance $g$ and the actual object width $g_i$, whereas $g_i \geq g$, the depth of sharpness can be calculated: $\epsilon_i = \frac{(1-\frac{g}{g_i})f^2}{\kappa(g-f)}$

• teleradiography conditions $g \gg f$, $b \approx f$ lead to: $g_h - g_v \approx \frac{2g^2\kappa\epsilon}{f^2}$

### Decomposition and Reconstruction of Gray Value Images G

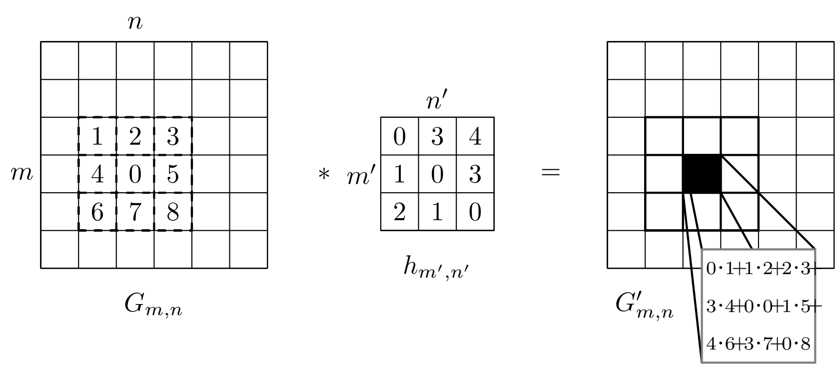
• Decomposition into basic images: $\mathbf{G} = \sum_{m=0}^{M-1}\sum_{n=0}^{N-1} G_{m,n}\mathbf{B}^{m,n}$, with $\mathbf{B}^{m,n} : B^{m,n}_{m',n'} = \begin{cases} 1 & \text{for } m = m' \wedge n = n', \\ 0 & \text{else} \end{cases}$

• 2D-DFT: $G_{m,n} \circ\!\!-\!\!\bullet \hat{G}_{u,v} = \frac{1}{MN}\sum_{m=0}^{M-1}\left(\sum_{n=0}^{N-1} G_{m,n} w_N^{-nv}\right) w_M^{-mu}$, with $w_M = e^{2\pi j/M}$ and $w_N = e^{2\pi j/N}$

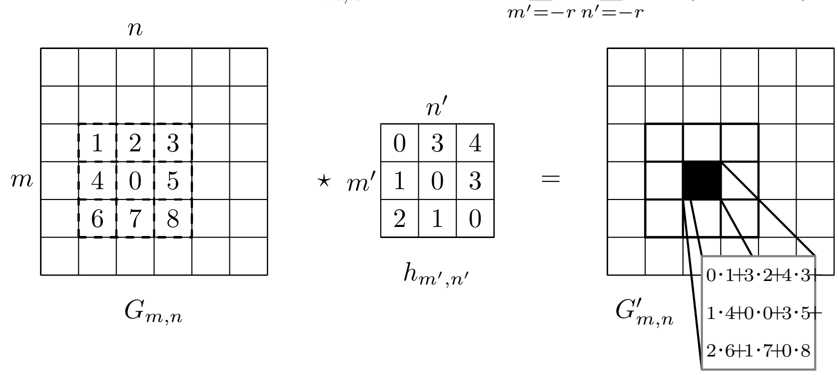
• inverse 2D-DFT: $\hat{G}_{u,v} \bullet\!\!-\!\!\circ G_{m,n} = \sum_{u=0}^{M-1}\left(\sum_{v=0}^{N-1} \hat{G}_{u,v} w_N^{nv}\right) w_M^{mu}$ , with $e^{2\pi j/M} = \cos(2\pi/M) + j\sin(2\pi/M)$

## Convolution and Correlation

• discrete 2D convolution: $G'_{m,n} = \mathbf{H} * \mathbf{G} = \sum_{m'=-r}^{r} \sum_{n'=-r}^{r} h_{-m',-n'} G_{m+m',n+n'}$




• discrete 2D correlation: $G'_{m,n} = \mathbf{H} \star \mathbf{G} = \sum_{m'=-r}^{r} \sum_{n'=-r}^{r} h_{m',n'} G_{m+m',n+n'}$




• convolution theorem: $G'_{m,n} = H_{m,n} * G_{m,n} \circ\!\!-\!\!\!-\!\!\bullet\, MN\hat{H}_{u,v}\hat{G}_{u,v}$

## Linear Filters

• box filter: $\mathbf{R} = \frac{1}{r^2}\mathbf{1}\mathbf{1}^\top \in \mathbb{R}^{r\times r}$

• example 3x3 filter:

$$\begin{bmatrix} 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \\ 1/9 & 1/9 & 1/9 \end{bmatrix}$$

• binomial filter: $\mathbf{B} = \mathbf{b}\mathbf{b}^\top$

$\mathbf{b} = \frac{1}{2^r} \underbrace{[11] * [11] * \cdots * [11]}_{r\text{-times}}$

• example 3x3 filter:

$$\begin{bmatrix} 1/16 & 1/8 & 1/16 \\ 1/8 & 1/4 & 1/8 \\ 1/16 & 1/8 & 1/16 \end{bmatrix}$$

### 1. order differences

• backward row: ${}^{-}\mathbf{D}_x = [1 \ -\dot{1}]$,
• backward col: ${}^{-}\mathbf{D}_y = {}^{-}\mathbf{D}_x^T$

• forward row: ${}^{+}\mathbf{D}_x = [\dot{1} \ -1]$,
• forward col: ${}^{+}\mathbf{D}_y = {}^{+}\mathbf{D}_x^T$

• symmetric row: $\mathbf{D}_{2x} = [\frac{1}{2} \ \dot{0} \ -\frac{1}{2}]$,
• symmetric col: $\mathbf{D}_{2y} = \mathbf{D}_{2x}^T$

• Sobel edge detector:

$\mathbf{S}_x = \mathbf{D}_{2x} * \mathbf{b}^\top =$

$\frac{1}{2}[1 \ \ 0 \ -1] * \frac{1}{4}\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} =$

$\frac{1}{8}\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$ whereas $\mathbf{S}_y = \mathbf{S}_x^\top$

## Convolution Masks 2nd Order Difference Operators

• 2nd order difference: $\mathbf{D}_x^2 = {}^{-}\mathbf{D}_x * {}^{+}\mathbf{D}_x = [1 \ -2 \ \ 1]$ • Laplace operator: $\mathbf{L} = \mathbf{D}_x^2 + \mathbf{D}_y^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$

• LoG-filter: $\mathbf{LoG} \approx \mathbf{L} * \mathbf{B}^p$ • DoG-filter: $\mathbf{DoG} \approx 4 * (\mathbf{B}^{r+2} - \mathbf{B}^r)$, $r = 2, 4, 6, \ldots$

**Nonlinear filters**

• ranking filter: $G(\mathbf{p}) = \mathcal{O}\{G(\mathbf{p}-\mathbf{p}')|\mathbf{p}' \in \mathcal{N}_\mathbf{p}\}\,,$ whereas $\mathcal{O} = \{median,\, min,\, max\}$

• Non maximum suppression: In a 3x3 neighborhood the gray values are compared. If there is a higher value in the neighborhood than the value at the anchor point, then the value of the anchor point is set to zero. Otherwise it remains. In the version of *Canny*, the gradient strength values are compared along the orientation.

• hysteresis filter: By means of two threshold values $\tau_1 > \tau_2$ the image is divided into three classes. If the gradient strength is above the larger threshold, then the pixel is assigned the class *strong edge* $K = 2$. If the gradient strength is below the smaller threshold, then the pixel is assigned the class *no edge* $K = 0$. If the gradient strength is between both thresholds, then the pixel is assigned the class *weak edge* $K = 1$. For each pixel classified as a weak edge, a check is made to see if there is a pixel in a 3x3 neighborhood that was classified as a strong edge. If this is the case, then the class is changed from weak edge to strong edge.

**Morphological Operators**

• erosion: $\mathbf{G} \ominus \mathbf{M}$ (Set of all pixels, for which $\mathbf{M}$ is completely contained in $\mathbf{G}$.)

• dilation: $\mathbf{G} \oplus \mathbf{M}$ (Set of all pixels for which the intersection of $\mathbf{M}$ and $\mathbf{G}$ is not the empty set.)

**Geometrical Transformations**

• forward transform: $\mathbf{x}' = \mathbf{M}(\mathbf{x})$ • backward transform: $\mathbf{x} = \mathbf{M}^{-1}(\mathbf{x}')$

• existence: $|\mathbf{J}(\mathbf{M}(\mathbf{x}))| = \left|\frac{\partial \mathbf{M}(\mathbf{x})}{\partial \mathbf{x}}\right| = \begin{vmatrix} \frac{\partial x'}{\partial x} & \frac{\partial x'}{\partial y} \\ \frac{\partial y'}{\partial x} & \frac{\partial y'}{\partial y} \end{vmatrix}\,,$ $|\mathbf{J}| = 0$: $\mathbf{M}^{-1}$ does not exist, $|\mathbf{J}| \neq 0$: $\mathbf{M}^{-1}$ exists

• nearest neighbor interpolation: $G(\mathbf{x}) = G(\hat{\mathbf{p}})\,,$ wobei $\hat{\mathbf{p}} = argmin_\mathbf{p}\|\mathbf{p}-\mathbf{x}\|$

• bilinear interpolation: $G(\mathbf{x}) = a_{00} + a_{10}x + a_{01}y + a_{11}xy\,,$ whereas

$a_{00} = G(\mathbf{p}_1)\,, a_{10} = G(\mathbf{p}_2) - G(\mathbf{p}_1)\,, a_{01} = G(\mathbf{p}_3) - G(\mathbf{p}_1)\,, a_{11} = G(\mathbf{p}_1) - G(\mathbf{p}_2) - G(\mathbf{p}_3) + G(\mathbf{p}_4)$

**Lokal Image Analysis**

• gradient vector: $\boldsymbol{\nabla}(\mathbf{x}) = \left[\frac{\partial G(\mathbf{x})}{\partial x}, \frac{\partial G(\mathbf{x})}{\partial y}\right]^\top\,,$ $\boldsymbol{\nabla}'(\mathbf{x}') = \mathbf{R}(\theta')\boldsymbol{\nabla}(\mathbf{x})\,,$ $\|\boldsymbol{\nabla}\| = \sqrt{\boldsymbol{\nabla}^\top\boldsymbol{\nabla}}$

• directional derivative: $\boldsymbol{\nabla}^\top\mathbf{n} = \cos(\theta)\frac{\partial G(\mathbf{x})}{\partial x} + \sin(\theta)\frac{\partial G(\mathbf{x})}{\partial y}\,,$ $\theta_0 = \tan^{-1}\left(\frac{\partial G(\mathbf{x})}{\partial y}\left(\frac{\partial G(\mathbf{x})}{\partial x}\right)^{-1}\right)$

• Hessian matrix: $\mathbf{H} = \begin{bmatrix} \frac{\partial^2 G(\mathbf{x})}{\partial x^2} & \frac{\partial^2 G(\mathbf{x})}{\partial x \partial y} \\ \frac{\partial^2 G(\mathbf{x})}{\partial x \partial y} & \frac{\partial^2 G(\mathbf{x})}{\partial y^2} \end{bmatrix}\,,$ $\mathit{\Delta} = trace(\mathbf{H}) = trace\left(\mathbf{R}(\theta')\mathbf{H}\mathbf{R}^\top(\theta')\right)$

• structure tensor: $\mathbf{J}(\mathbf{x}) = \begin{bmatrix} J_{11}(\mathbf{x}) & J_{12}(\mathbf{x}) \\ J_{12}(\mathbf{x}) & J_{22}(\mathbf{x}) \end{bmatrix} = \int\limits_{\mathbf{x}' \in \mathcal{N}(\mathbf{x})} W(\mathbf{x}-\mathbf{x}')\left(\boldsymbol{\nabla}(\mathbf{x}')\boldsymbol{\nabla}(\mathbf{x}')^\top\right)d\mathbf{x}' = \mathbf{W} * \left(\boldsymbol{\nabla}\boldsymbol{\nabla}^\top\right)$

• coherence: $c(\mathbf{x}) = \frac{\sqrt{(J_{22}-J_{11})^2 + 4J_{12}^2}}{J_{11}+J_{22}} = \frac{\lambda_1 - \lambda_2}{\lambda_1 + \lambda_2}$, whereas $\lambda_1, \lambda_2$ are eigenvalues of structure tensor

• Harris corner detector: $det(\mathbf{J}) + k \cdot Spur^2(\mathbf{J}) = \lambda_1\lambda_2 - k(\lambda_1 + \lambda_2)^2 > \tau$, threshold to choose $\tau$

• Shi-Tomasi corner detector: $\min(\lambda_1, \lambda_2) > \tau$