



INTELLIGENTE DATENANALYSE IN MATLAB

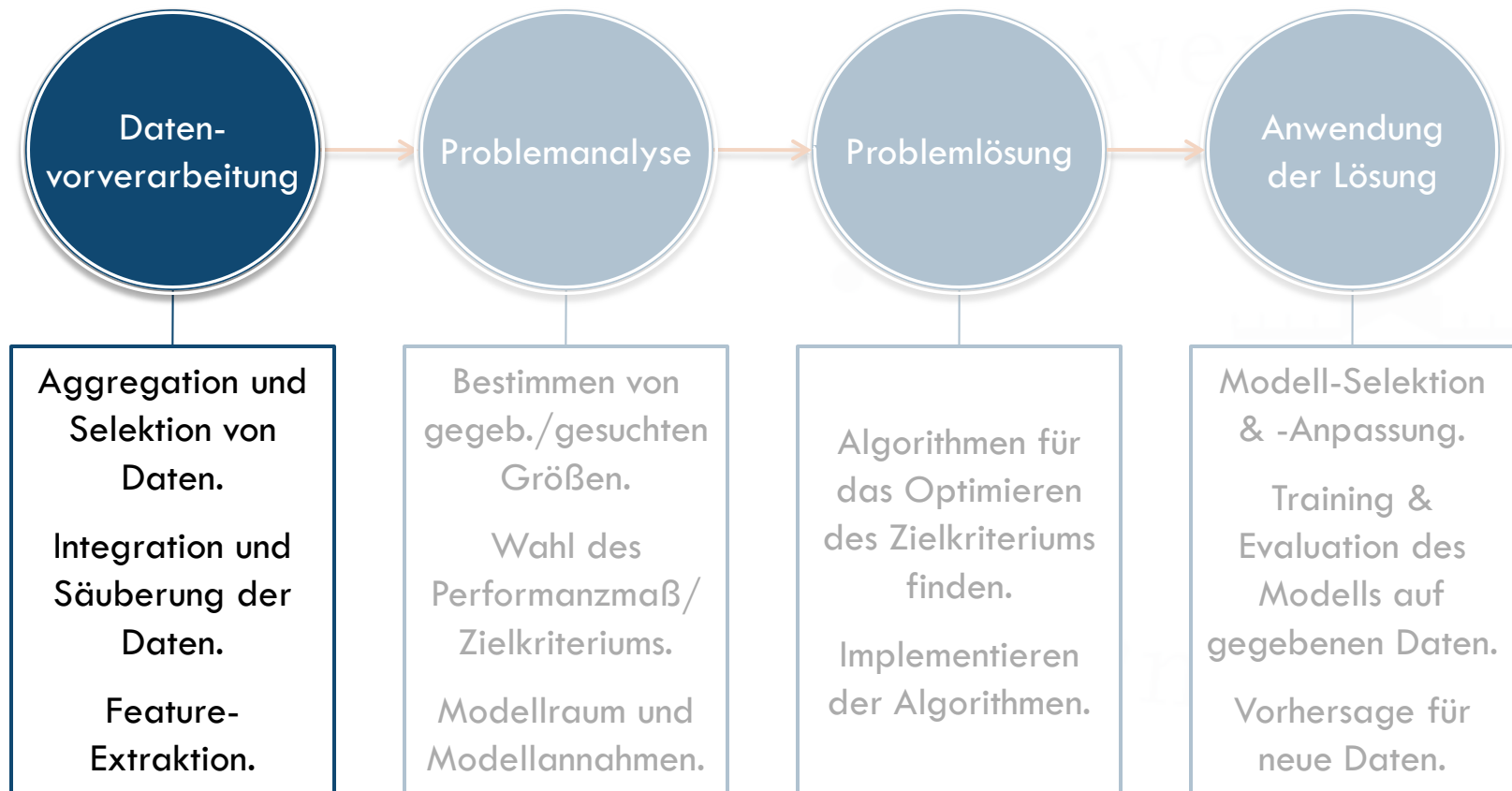
Datenselektion und Datenaufbereitung

Literatur

- I. H. Witten, E. Frank: *Data Mining – Practical Machine Learning Tools and Techniques*.
- J. Han, M. Kamber: *Data Mining – Concepts and Techniques*.
- David A. Forsyth: *Computer Vision – A Modern Approach*.
- Mark S. Nixon und Alberto S. Aguado: *Feature Extraction and Image Processing*.

Überblick

□ Schritte der Datenanalyse:



Überblick

- **Aggregation und Selektion von Daten.**
 - Datenquellen identifizieren und selektieren.
 - Bestimmung von Art und Umfang der Daten.
- **Integration und Säuberung der Daten.**
 - Unvollständige/fehlende Daten.
 - Fehlerhafte und inkonsistente Daten.
- **Feature-Extraktion.**
 - Feature generieren.
 - Daten transformieren.
 - Daten- & Feature-Reduktion.
- **Spezialfall: Bildanalyse.**

Aggregation und Selektion von Daten

Datenquellen identifizieren und selektieren

- Genauigkeit (Anteil Rauschen in den Daten).
- Vollständigkeit (Anteil fehlender/unbrauchbarer Werte).
- Konsistenz (Widerspruch in Daten einer/mehrerer Quellen).
- Aktualität der Datenquelle.
- Glaubwürdigkeit der Datenquelle.
- Mehrwert/Redundanz.
- Verständlichkeit (Bedeutung der Attribute).
- Verfügbarkeit.

Aggregation und Selektion von Daten

Bestimmung von Art und Umfang der Daten

- Stichprobe:
 - Auswahl repräsentativer Daten.
- Umfang der Daten:
 - Menge an unterschiedlichen Instanzen.
 - Menge an unterschiedlichen Attributen.
 - Datendichte/Sparsity.
- Art der Daten:
 - Elementare/zusammengesetzte Daten.

Aggregation und Selektion von Daten

Bestimmung von Art und Umfang der Daten

□ Elementare Datentypen:

- Kontinuierliche Werte (z.B. Temperatur, Alter).
- Ordinale Daten, d.h. diskrete Werte einer geordneten Menge (z.B. Schulnoten, Fahrzeugklassen, Stockwerke).
- Nominale Daten, d.h. diskrete Werte einer ungeordneten Menge (z.B. Wörter, Familienstand, Augenfarbe, Nationalität).

□ Zusammengesetzte Datentypen:

- Tupel/Vektoren, Matrizen, Tensoren (z.B. Bilder).
- Sequenzen (z.B. Text, EKG-Kurve, Audio-/Video-Daten).
- Graphen, verlinkte Daten (z.B. Webseiten, Moleküle).

Integration und Säuberung der Daten

Überblick

- Daten zusammentragen, konvertieren und integrieren in eine kohärente Form, z.B. in einem *Data Warehouse*.
 - Daten-Integration (z.B. unterschiedliche Speicherformate/-orte).
 - Schema-Integration (z.B. Metadaten von verschiedenen Quellen, gleiche/ähnliche Attribute unterschiedlicher Quellen)
 - Daten-Konflikte behandeln (z.B. Umrechnung von Einheiten).
 - Redundante Informationen erkennen (z.B. durch Korrelationsanalyse, Dubletten-Suche).
- Säuberung der Daten.

Integration und Säuberung der Daten

Unvollständige/fehlende Daten

□ Arten von fehlenden Daten:

- Fehlende Werte (Attributbelegungen).
- Fehlende Attribute welche für die Datenanalyse von Interesse wären.
- Aggregierte Werte/Attribute.

□ Ursache für fehlende Daten:

- Zufälliges Fehlen (z.B. Speicherfehler, Fehlfunktion eines Messinstruments).
- Systematisches Fehlen (z.B. Wert zu einem früheren Zeitpunkt unwichtig/unbekannt).
- Daten-Integration (z.B. gelöschte Werte aufgrund von Inkonsistenzen).
- Daten-Aggregation (z.B. aus Datenschutzgründen).
- ...

Integration und Säuberung der Daten

Unvollständige/fehlende Daten

- **Behandlung fehlender Werte:**
 - Betreffende Instanzen/Attribute löschen.
 - Erweiterung des Wertebereichs (z.B. „missing“) und/oder der Attributmenge (z.B. binäres Attribut_XY_bekannt).
 - Fehlende Werte aus Daten schätzen:
 - Mean/Median Imputation (evtl. Klassen-abhängig).
 - Inferenz aus den Daten (z.B. mittels EM-Algorithmus).
 - Fehlende Werte nicht behandeln (späteres Lern-/Analyseverfahren berücksichtigt fehlende Werte).

Integration und Säuberung der Daten

Fehlerhafte und inkonsistente Daten

□ Arten von fehlerhaften Daten:

- Rauschen in numerischen Werten. $\tilde{x} = x + \varepsilon$ $\varepsilon \sim N(0, \sigma^2)$
- Ausreißer/falsch gelabelte Daten.
- Inkonsistenz.

□ Ursache für fehlerhafte Daten:

- Zufällige Störung (z.B. durch Messungenauigkeit).
- Systematische Störung (z.B. technische Grenzen, Formatierungsfehler).
- Unabsichtliche Störung (z.B. falsches Labeling durch menschliche Fehler).
- Absichtliche Störung (z.B. durch Spammer bei Emails).
- Daten-Integration (z.B. fehlerhafte Konvertierung/Datenübertragung, Inkonsistenzen durch sich widersprechende Datenquellen).
- ...

Integration und Säuberung der Daten

Fehlerhafte und inkonsistente Daten

□ Identifizierung fehlerhafter Werte:

- Binning: äquidistante Diskretisierung in *Bins*
⇒ Bins mit einem/wenigen Instanzen enthalten evtl. Ausreißer.
- Clustering: Suche nach Regionen mit hoher Datendichte (*Cluster*)
⇒ Cluster mit einem/wenigen Instanzen enthalten evtl. Ausreißer.
- Active Learning: Widerspruch zwischen Daten und Modell
⇒ auffällige Instanzen werden Menschen zur Kontrolle vorgeschlagen.

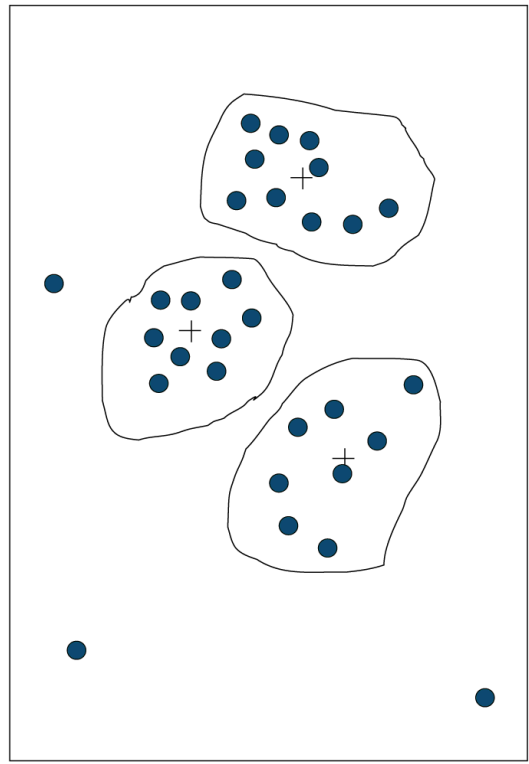
□ Behandlung fehlerhafter Werte:

- Glättung numerischer Werte (z.B. Regression, Moving Average).
- Diskretisierung (z.B. Alter ⇒ {kücken, bivi, uhu}).
- Als fehlend behandeln.

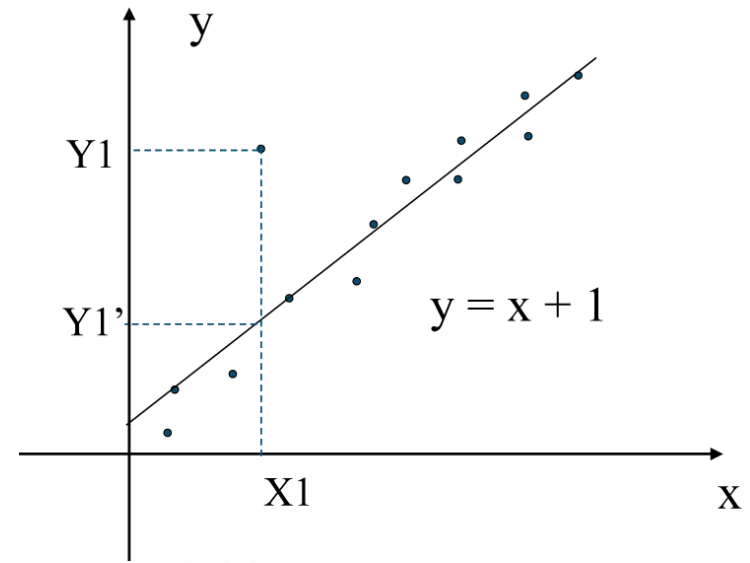
Integration und Säuberung der Daten

Fehlerhafte und inkonsistente Daten

Clustering



Lineare Regression



Feature-Extraktion

Feature generieren aus Texten

□ Text-Vorverarbeitung:

- Konvertierung (z.B. Zeichensatz, Klein-/Großschreibung, HTML-Decoding).
- Tokenisierung.
- Stopwords (z.B. Artikel, Konjunktionen) entfernen.
- Stemming (z.B. Verben in Infinitivform, Substantive in Singularform).

□ Feature aus Texten:

- TF-IDF (Term Frequency – Inverse Document Frequency).
- Bag-of-Words (Wörter sind nominale bzw. binäre Attribute).
- N-Gram (Wort-Tupel der Länge N sind nominale bzw. binäre Attribute).
- OSB (Orthogonal Sparse Bigrams).
- Darstellung als Baum/Graph (z.B. bei XML).

Feature-Extraktion

Feature generieren aus Texten

□ Bag-of-Words:

„Hello World“ $\Rightarrow [0 \dots 0 \ 1 \ 0 \ \dots \ 0 \ 1 \ 0 \ \dots \ 0]^T$

Index von „World“

Index von „Hello“

□ TF-IDF:

$$tf_{i,j} = \frac{freq_{i,j}}{\max_k (freq_{k,j})}$$

$$idf_i = \log \frac{N}{n_i}$$

$$w_{i,j} = tf_{i,j} \cdot idf_i = \frac{freq_{i,j}}{\max_k (freq_{k,j})} \log \frac{N}{n_i}$$

Feature-Extraktion

Feature generieren aus Bildern

□ Bild-Vorverarbeitung:

- Normierung (z.B. Farbwerte, Helligkeit, Farbintensität, Größe, Lage).
- Transformation (z.B. Skalierung, Verschiebung, Drehung).
- Filtern (z.B. Glättung, Kanten hervorhebung).

□ Feature aus Bildern:

- Metainformationen (z.B. Tags, Format, Größe, Farbverteilung).
- Pixelvektor bzw. Pixelbereiche (Umwandlung der Bildmatrix in Vektor).
- FFT (Fast Fourier Transform).
- SIFT (Scale-Invariant Feature Transform).
- HT (Hough Transform).

Feature-Extraktion

Feature generieren aus Strukturen

- **Vorverarbeitung verlinkter Daten:**
 - Umwandlung verlinkter Daten in (un-)gewichteten, (un-)gerichteten Graph.
 - Graph-Konvertierung (z.B. Minimal Spanning Tree/Forest, zusammenhängender Graph, Zyklen-freier Graph).

- **Feature aus verlinkten Daten:**
 - Knoten-Attribute (z.B. Anzahl ein-/ausgehender Kanten).
 - Kanten-Attribute (z.B. verbindende Knoten, Gewicht, Richtung).
 - Häufige Subgraphen, Subtrees, Cliques.

Feature-Extraktion

Daten transformieren

□ Feature-Normalisierung:

□ Min/Max-Normalisierung:
$$x^{new} = \frac{x - x_{\min}}{x_{\max} - x_{\min}} (x_{\max}^{new} - x_{\min}^{new}) + x_{\min}^{new}$$

□ Z-Score-Normalisierung:
$$x^{new} = \frac{x - \mu_x}{\sigma_x}$$

□ Dezimal-Skalierung:
$$x^{new} = |x| \cdot 10^a \quad a = \max_x \{i \in \mathbb{Z} \mid |x| \cdot 10^i < 1\}$$

□ Logarithmische Skalierung:
$$x^{new} = \log_a x$$

Feature-Extraktion

Daten transformieren

□ Instanzen normalisieren:

□ Unabhängige l_a -Normierung aller Datenpunkte: $\mathbf{x}^{new} = \|\mathbf{x}\|_a^{-1} \mathbf{x}$

□ Korrelation zw. Attributen entfernen: $\mathbf{x}^{new} \sim N(\mathbf{0}, \mathbf{I})$

■ LU-Zerlegung der Kovarianz-Matrix $\frac{1}{n} \sum_{i=1}^n (\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^T$.

■ Transformation der Eingaben: $\mathbf{x}^{new} = \mathbf{L}^{-1/2} \mathbf{U}^T (\mathbf{x} - \bar{\mathbf{x}})$

□ Feature-Konstruktion:

□ Kombination elementarer Feature (z.B. $(x_i, x_j) \rightarrow (x_j, \sqrt{x_i x_j}, x_i + x_j)$).

□ Mapping elementarer Feature (z.B. $x_i \rightarrow (x_i, \log x_i, x_i^2)$).

Feature-Extraktion

Daten transformieren

□ Diskretisierung:

- Binning/Clustering: Wertebereich auf Mittelpunkte/Mediane der Bins bzw. Cluster einschränken.
- Entropie-basierte Diskretisierung:
Partitionierung der Menge S in S_1 und S_2 durch $\arg \min_{S=S_1 \cup S_2} \frac{|S_1|}{|S|} H_{S_1} + \frac{|S_2|}{|S|} H_{S_2}$.
- Binärisierung:
 - Numerische Werte (durch Schwellwert).
 - Nominale Werte (jeder Wert des Wertebereichs wird ein eigenes, binäres Attribut).
- Natürliche Partitionierung (z.B. durch Hierarchien/Taxonomien).

Feature-Extraktion

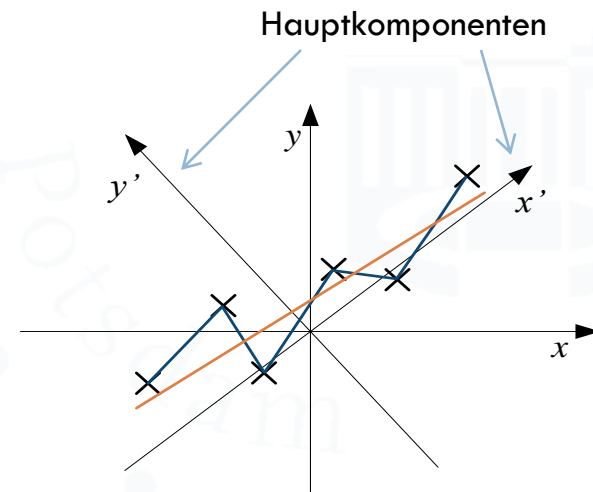
Daten- & Feature-Reduktion

□ Feature-Selektion (Dimensionsreduktion):

- Vorwärts-Selektion bzw. Rückwärts-Elimination (Kombination aus beiden).
- Elimination nach dem Lernen.
- Automatische Selektion beim Lernen (z.B. Entscheidungsbaumlernen).

□ Datenkompression:

- Wavelet-/Fourier-Transformation.
- Hauptkomponenten-/Faktoranalyse.
- Interpolation (z.B. **lineare Interpolation**, Spline-Interpolation).
- Regression (z.B. **lineare Regression**).
- Sampling.



- Bildanalyse notwendig für zahlreiche Anwendungen:
 - ▣ Suche nach Bildern mit/ohne Textbeschreibung (Tags).
 - ▣ Suche nach ähnlichen Bildern bzw. Clustern von Bildern.
 - ▣ Partitionierung bzw. Identifizieren von Objekten in einem Bild (z.B. Gesichter, Buchstaben/Zahlen, Barcode).
 - ▣ Klassifikation dieser Objekte (z.B. OCR).
 - ▣ Rekonstruktion von 3D-Objekten aus mehreren Bildern.
 - ▣ Bildbearbeitung und -kompression (z.B. Entfernen von Störungen, Vermeidung von Artefakten).

Bildanalyse

Problemstellung

- Bildanalyse = Datenvorverarbeitung von Bildern.
- Gegeben: Bilder, d.h. Matrizen mit Grau-/Farbwerten.
- Gesucht: Numerische Attribute welche das Bild möglichst gut beschreiben.
 - Um Ähnlichkeit zwischen Bilder zu bestimmen,
 - Um Bilder zu Partitionieren usw.
- Betrachten nur Feature-Extraktion.
 - Weitere Datenanalyse basiert auf diesen Features.

Bildanalyse

Überblick

- Extraktion von Farbinformationen:
 - Punktbasierte Statistik 1. und 2. Ordnung.
- Informationen über Kanten & Flächen:
 - Kantenerkennung/-verstärkung/-filterung.
 - Glättung/Rauschfilter.
- Muster & Texturen (Anwenden von allgemeinen Filtern):
 - Konvolution.
- Weitere Informationen über Form, Frequenzen usw.:
 - Hough-Transformation.
 - Fourier-Transformation.
 - ...

Bildanalyse

Beispiel



Bildanalyse

Farben

- Punktbasierte Statistik 1. Ordnung:
 - Helligkeit.
 - Kontrast.
 - Verteilung/Histogramm.
 - Entropie/Anisotropiekoeffizient.
- Punktbasierte Statistik 2. Ordnung:
 - Co-Occurence-Matrix.
- Berechnung pro Kanal bei farbigen (mehrkanaligen) Bildern.

Bildanalyse

Farben – Punktbasierte Statistik 1. Ordnung

- Helligkeit:
 - ▣ Mittlerer Grauwert.



$$\mu_g = 92,59$$



$$\mu_g = 153,32$$

Bildanalyse

Farben – Punktbasierte Statistik 1. Ordnung

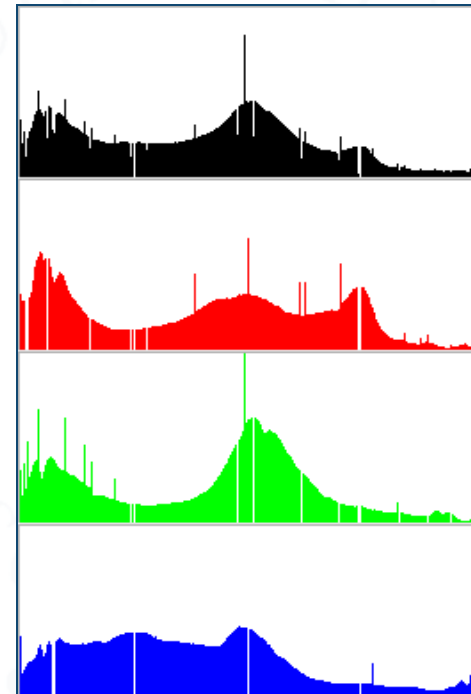
- Kontrast:
 - Globaler Kontrast = Maximaler Unterschied zwischen Grauwerten eines Bildes.
 - Lokaler Kontrast = Maximaler Unterschied zwischen Grauwerten benachbarter 4 (bzw. 8) Pixel.
 - Mittlere quadratische Abweichung (Varianz).



Bildanalyse

Farben – Punktbasierte Statistik 1. Ordnung

- Verteilung/Histogramm:
 - Wahrscheinlichkeitsdichte für Farb-/Grauwerte.

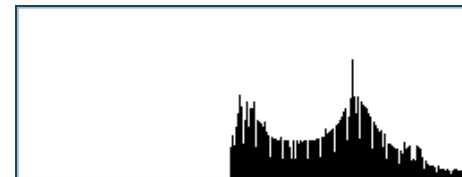


Bildanalyse

Farben – Punktbasierte Statistik 1. Ordnung

□ Verteilung/Histogramm:

- Bei dunklem Bild mit wenig Kontrast viele kleine Grauwerte.
- Bei hellem Bild mit wenig Kontrast viele große Grauwerte.
- Bild mit vorwiegend einem dunklen und einem hellen Bereich erzeugt Histogramm mit zwei lokalen Maxima.



□ Informationsgehalt:

■ Entropie:

$$H_g = -\sum_{g=0}^{2^{bit}} p(g) \log p(g)$$

Relative Häufigkeit
des Grauwerts g

- Durchschnittliche Anzahl benötigter Bits (0 bis bit) für Pixelmenge.
- Maß für maximale Kompression durch Reduktion von Signalredundanz.

■ Anisotropiekoeffizient:

$$\alpha = \frac{-\sum_{g=0}^k p(g) \log p(g)}{H_g}$$

Entropie der 50%
kleinsten Grauwerte

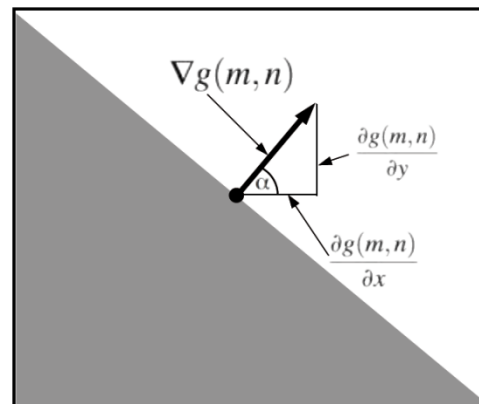
wobei k der kleinstmögliche Grauwert ist mit $\sum_{g=0}^k p(g) \geq 0,5$.

- Maß für die Symmetrie eines Histogramms.

Bildanalyse

Kantenerkennung

- Erste Ableitung einer zweidimensionalen Grauwertfunktion f ist ein Vektor (Gradient)
 - ▣ Dessen Richtung orthogonal zur Kante verläuft,
 - ▣ Der von „dunkel“ nach „hell“ zeigt, und
 - ▣ Dessen Länge ein Maß für die „Kantenstärke“ ist.



$$\nabla g(m,n) = \begin{pmatrix} \frac{\partial g(m,n)}{\partial x} \\ \frac{\partial g(m,n)}{\partial y} \end{pmatrix}$$

$$\alpha = \arctan \left(\frac{\frac{\partial g(m,n)}{\partial y}}{\frac{\partial g(m,n)}{\partial x}} \right)$$

$$(\alpha = \text{atan2}(g_y, g_x))$$

Bildanalyse

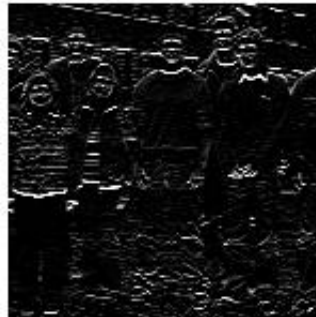
Kantenerkennung

- Approximation der 1./2. Ableitung an einer Position im Bild durch Differenzen zu benachbarten Grauwerten:
 - ▣ Erkennung mittels erster Ableitung (z.B. Prewitt-, Sobel-Filter).
 - ▣ Erkennung mittels zweiter Ableitung (z.B. Laplace-, LoG-Filter).

original



prewitt



sobel



LoG



Bildanalyse

Glättung

- Grauwert an einer Position im Bild ist mittlerer Grauwert benachbarter Pixel:
 - Rechteck-Filter.
 - Gauß-/Binomial-Filter.

original



Rechteck



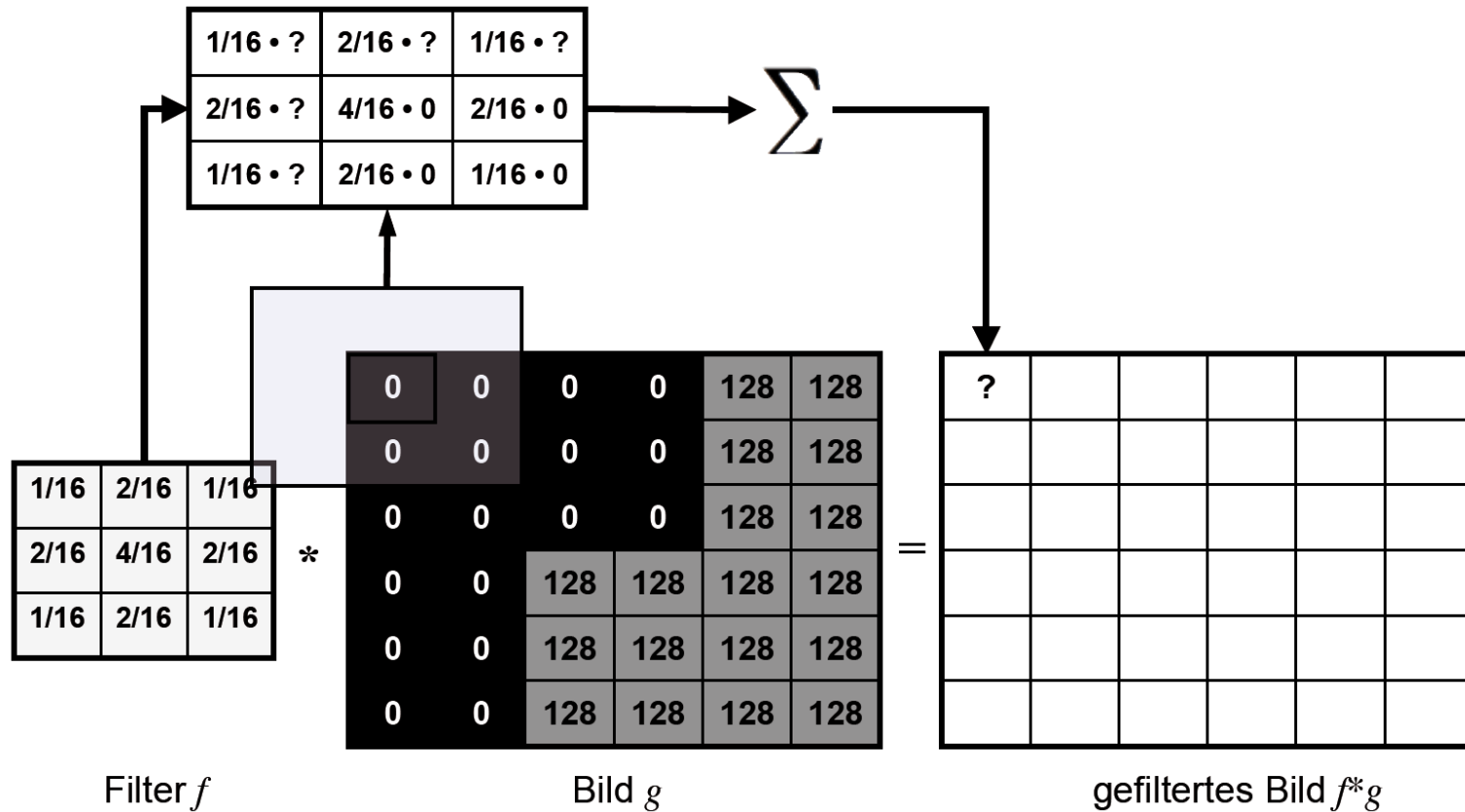
Gauss



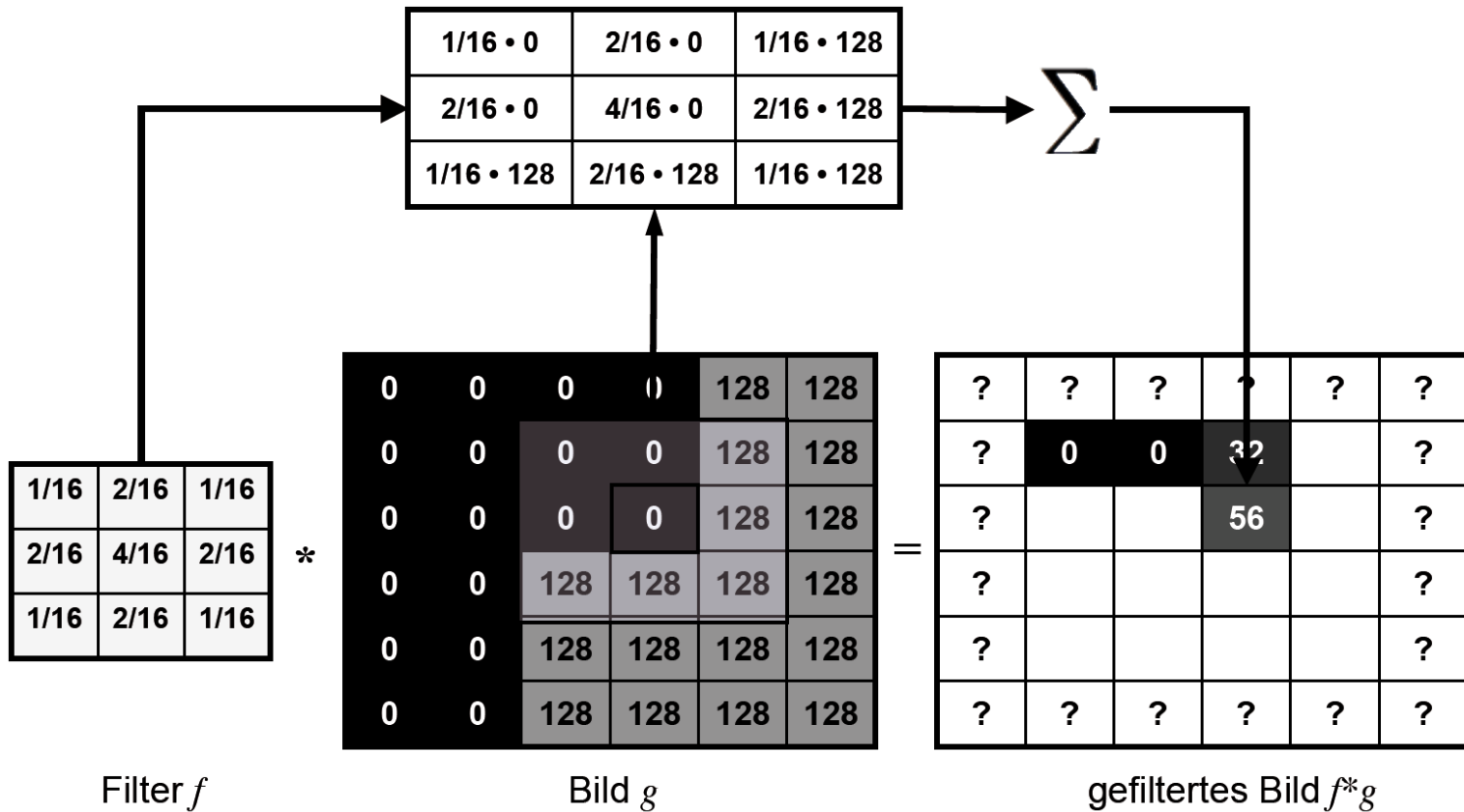
Bildanalyse

Konvolution

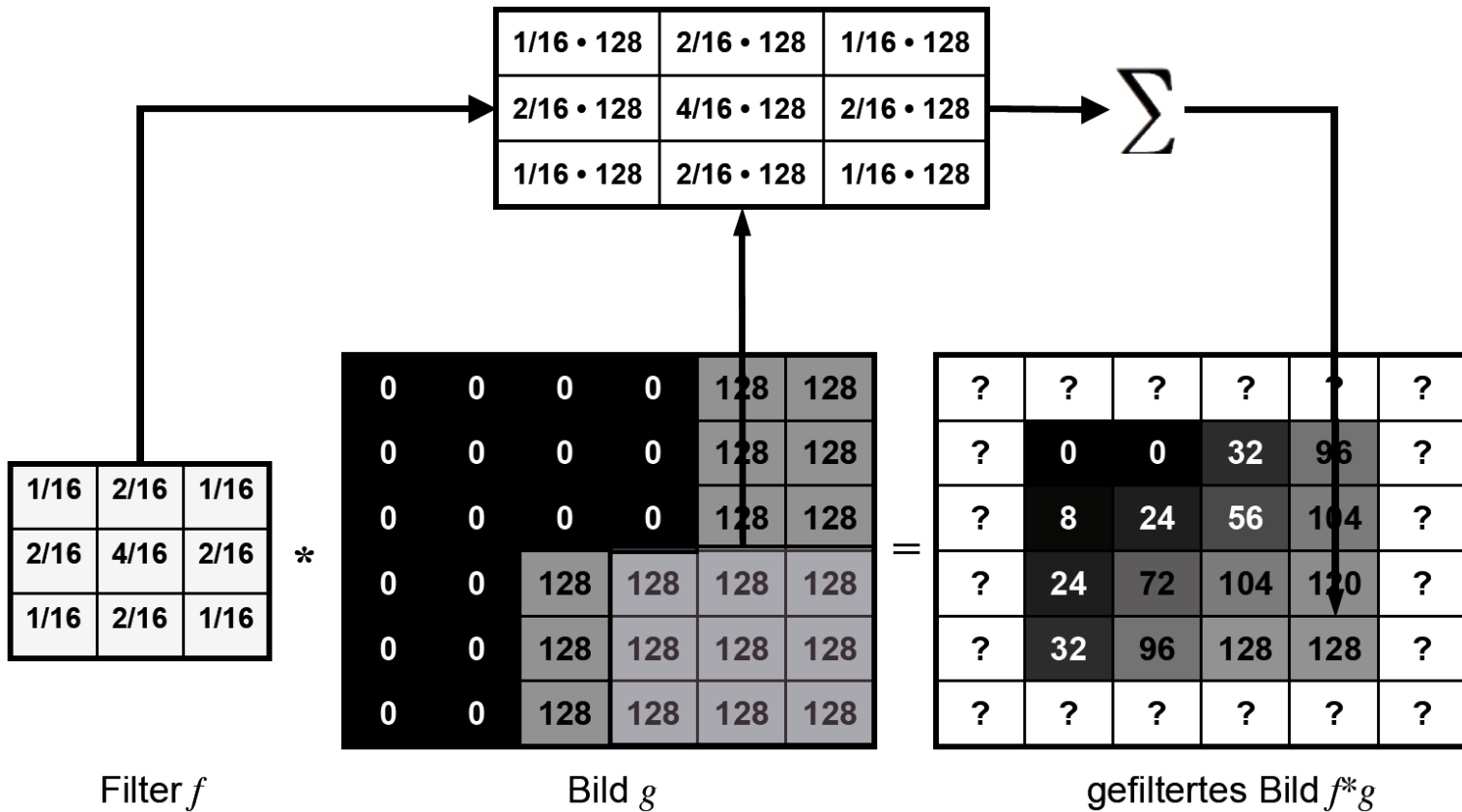
- Konvolution (Faltung) = Anwenden eines Filters:



- Konvolution (Faltung) = Anwenden eines Filters:



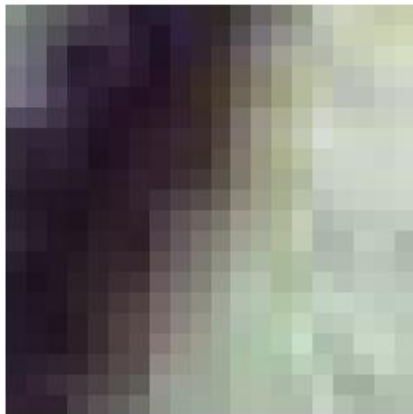
- Konvolution (Faltung) = Anwenden eines Filters:



Bildanalyse

Konvolution

- Beispiel für Anwendung eines Textur-Filters:



Zusammenfassung

- Datenvorverarbeitung entscheidend für Qualität der Datenanalyse.
- Dient der Bereitstellung aller *relevanten* Daten in der *nützlichsten* Form.
- Sie umfasst
 - Aggregation und Selektion von Daten,
 - Integration und Säuberung der Daten und
 - Feature-Extraktion.
- Vorverarbeitung oft zeitaufwendiger als Datenanalyse!