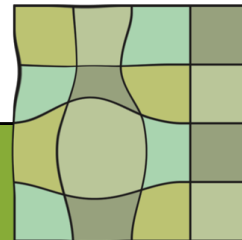

Grundlagen der Bildverarbeitung

Digitale Bilder

Prof. Dr. Klaus Tönnies

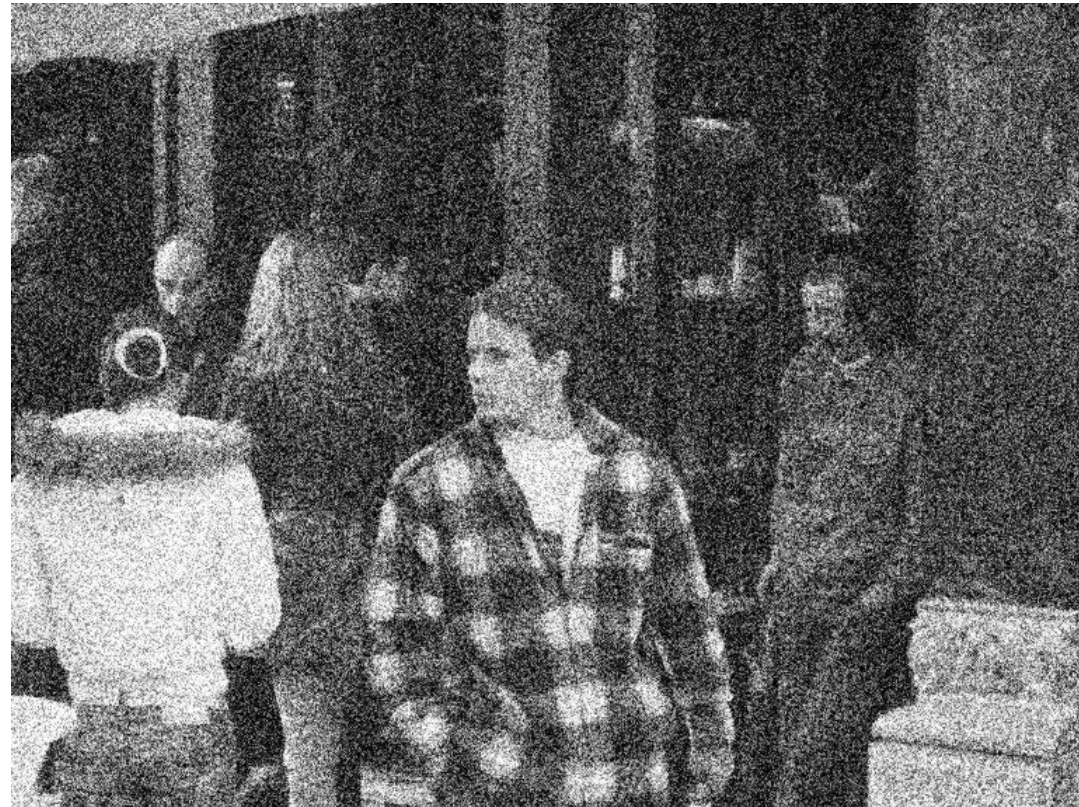


Bildverarbeitung
&
Bildverstehen



Famous Last Question

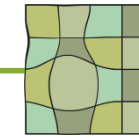
***Was könnte/sollte getan werden, um
die Anzahl
der Personen
zu zählen?***





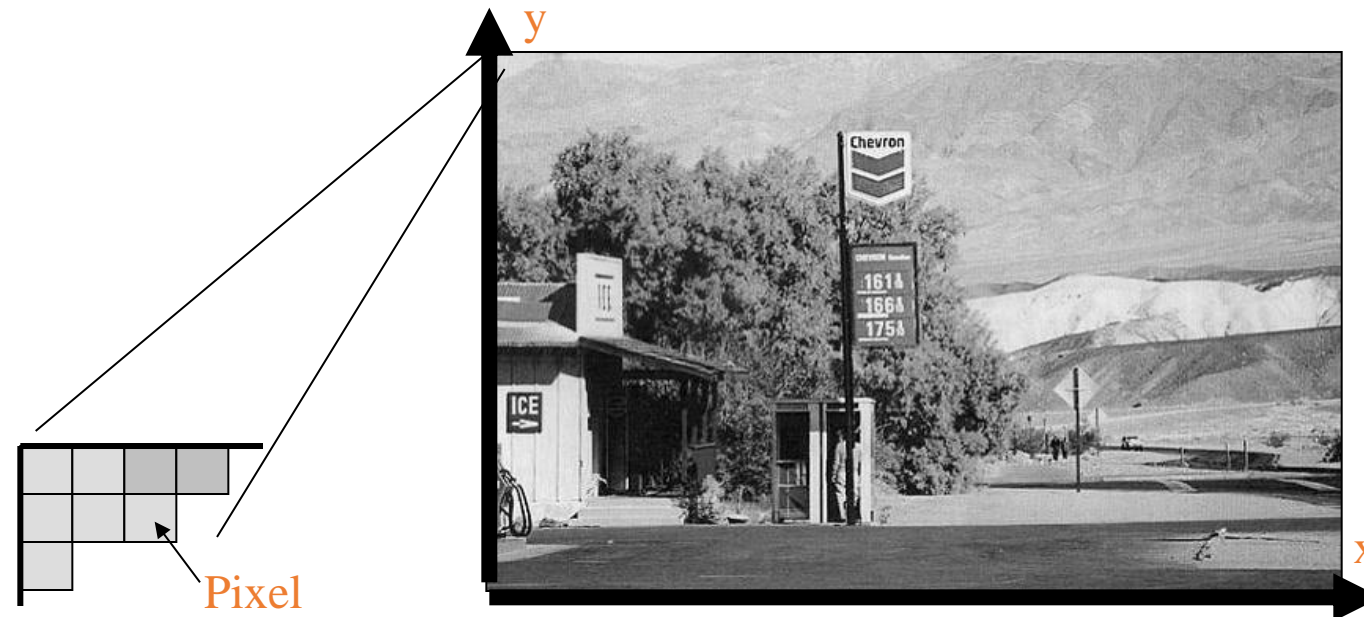
Digitale Bilder

- Was ist ein digitales Bild und welche Information wird gespeichert
 - Digitales Foto
 - Grundlegende Eigenschaften
 - Bildformate

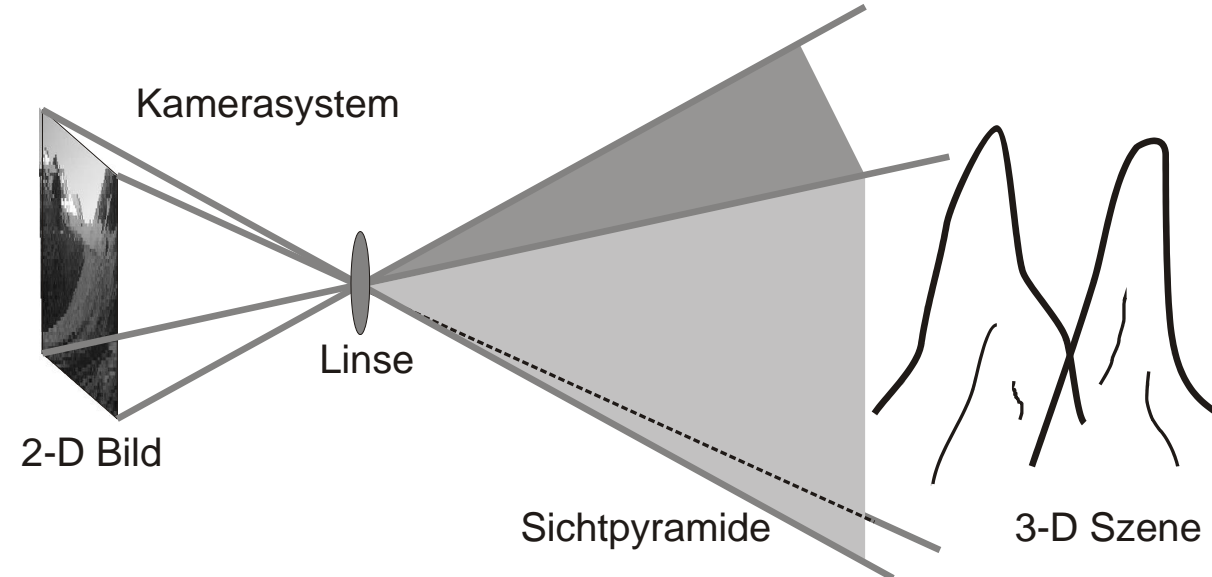


Digitales Foto

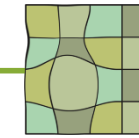
- Das Bild besteht aus einer Menge von Bildelementen (**P**ixel von „p**i**cture **e**lement“).
- Definitionsbereich: Ausdehnung in x- und y-Richtung, sowie Wellenlänge:
 $x_{\min} \leq x < x_{\max}$, $y_{\min} \leq y < y_{\max}$, $\lambda_{\min} \leq \lambda < \lambda_{\max}$.
- Wertebereich: Intensität
 $R_{\min} \leq R(x, y, \lambda) < R_{\max}$
- Definitions- und Wertebereich sind beschränkt.



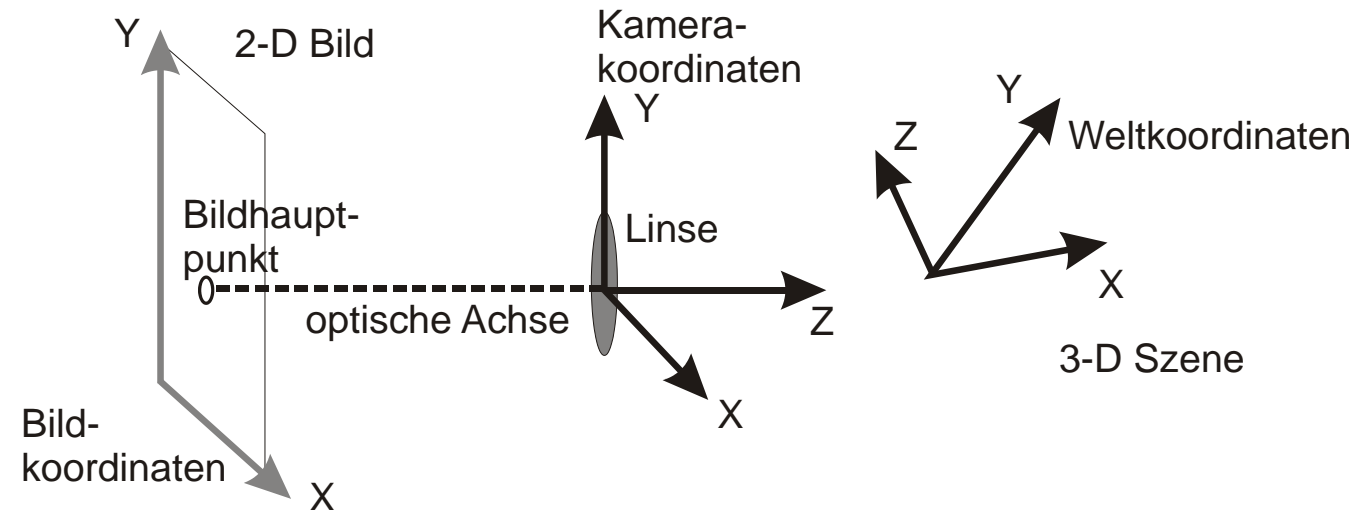
Projektion bei der Aufnahme



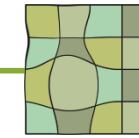
- Alle Punkte außerhalb der Sichtpyramide werden nicht abgebildet.
- Alle verdeckten Punkte in der Sichtpyramide werden nicht abgebildet.
- Von den abgebildeten Punkten geht die Tiefeninformation verloren.



Koordinatensysteme



- *Weltkoordinatensystem:* 3-D, reell, relativ zu einem Punkt in der Welt.
- *Kamerakoordinatensystem:* 3-D, reell, relativ zur Kamera.
- *Bildkoordinatensystem:* 2-D, reell oder ganzzahlig, begrenzt, relativ zum CCD-Chip.



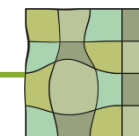
Transformation und Projektion

- Transformation: Rotation und Translation des Kamerakoordinatensystems relativ zum Weltkoordinatensystem:

$$\begin{pmatrix} x_w & y_w & z_w & 1 \end{pmatrix} = \begin{pmatrix} r_{11} & r_{12} & r_{13} & t_x \\ r_{21} & r_{22} & r_{23} & t_y \\ r_{31} & r_{32} & r_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}.$$

- Projektion der Kamerakoordinaten in die Bildkoordinaten:

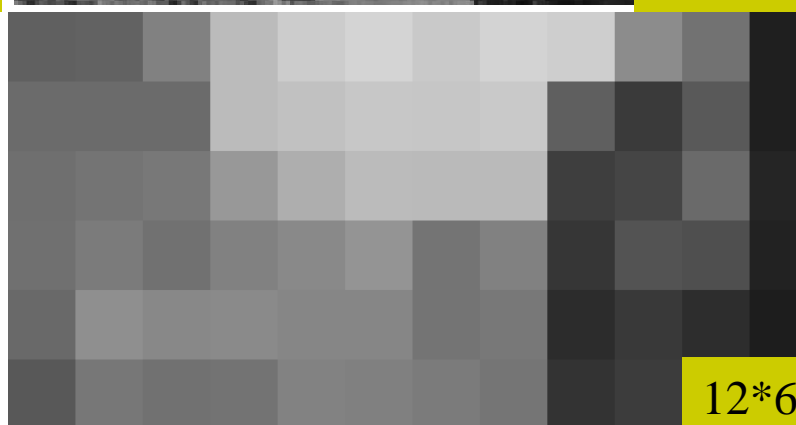
$$\begin{pmatrix} x_i & y_i & -f & 1 \end{pmatrix} := \begin{pmatrix} -f \frac{x_c + d_x}{z_c} & -f \frac{y_c + d_y}{z_c} & -f & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & d_x \\ 0 & 1 & 0 & d_y \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/f & 0 \end{pmatrix} \begin{pmatrix} x_c \\ y_c \\ z_c \\ 1 \end{pmatrix}.$$



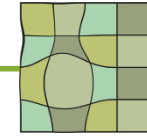
Ortsauflösung

← M →

↑
 N
↓

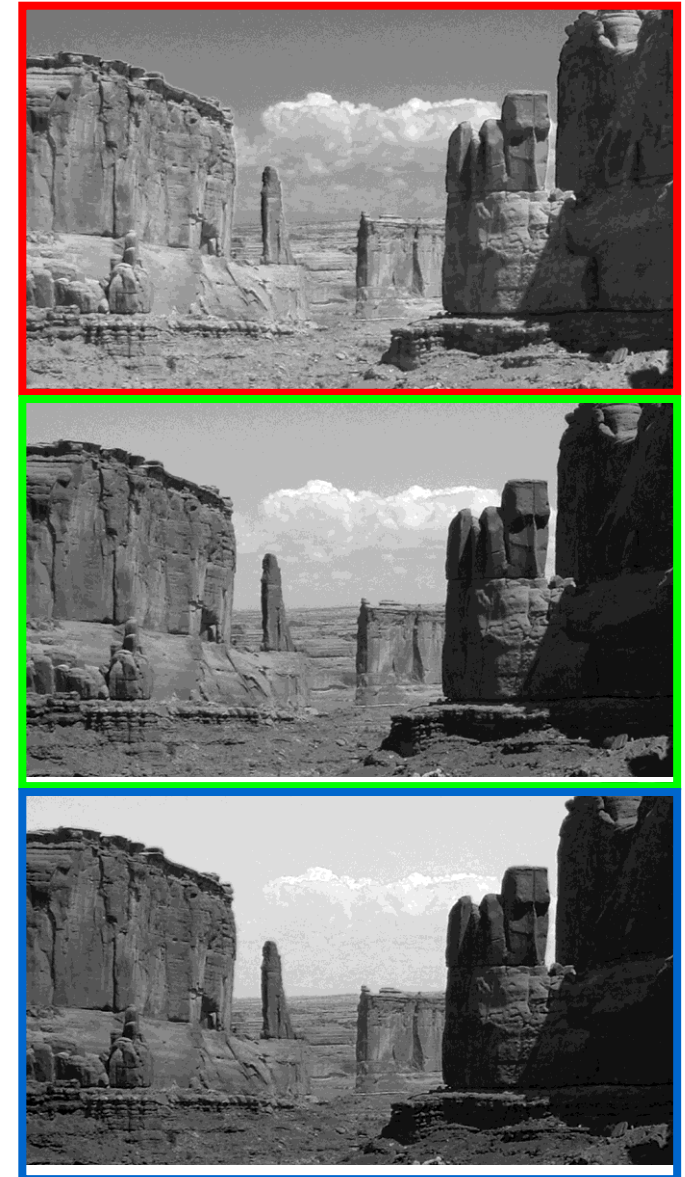


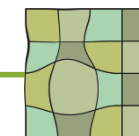
$$m = \left\lfloor M \cdot \frac{x - x_{\min}}{x_{\max} - x_{\min}} \right\rfloor \text{ und } n = \left\lfloor N \cdot \frac{y - y_{\min}}{y_{\max} - y_{\min}} \right\rfloor \text{ für } x_{\min} \leq x < x_{\max}, y_{\min} \leq y < y_{\max}$$



RGB Farbmodell

- Jede Farbe wird durch ein Tripel (rot, grün, blau) repräsentiert.



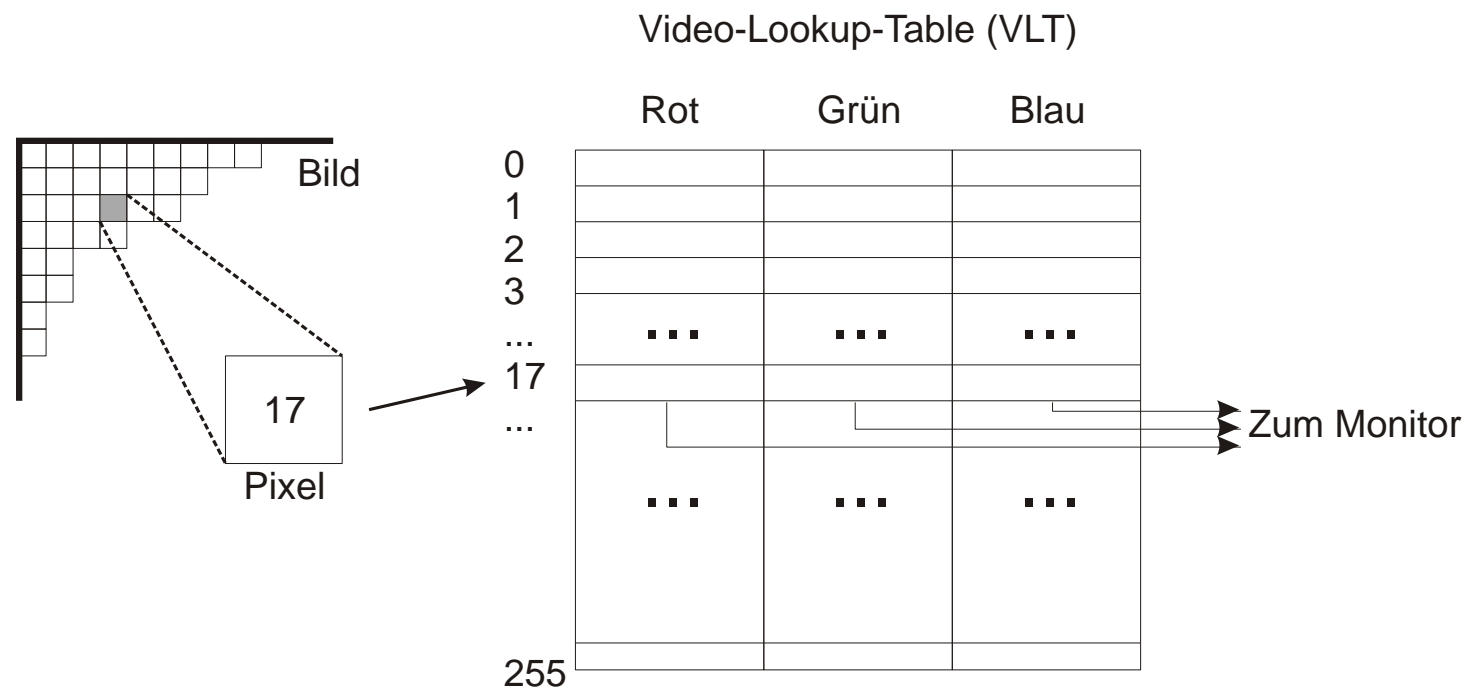


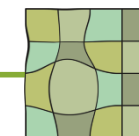
Repräsentation eines Bildes

- Es werden (meist) 256 Intensitätsstufen unterschieden (= 1 Byte)

- Farbe wird
 - als Tripel (r,g,b) repräsentiert.
 - oder über einen Index in eine Farbtabelle repräsentiert.

- Video-Lookup-Table (VLT)





Kontrastauflösung

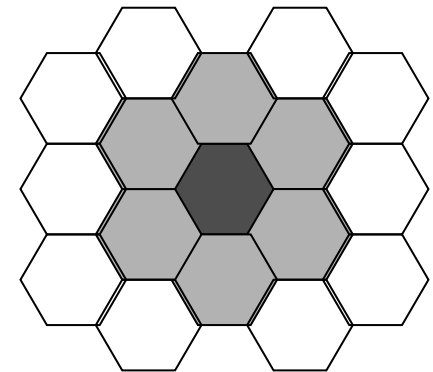
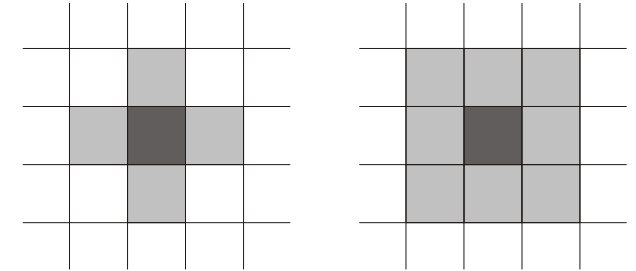


Reduzierung der Kontrastauflösung mindert die Erkennbarkeit weniger als die der Ortsauflösung



Nachbarschaft

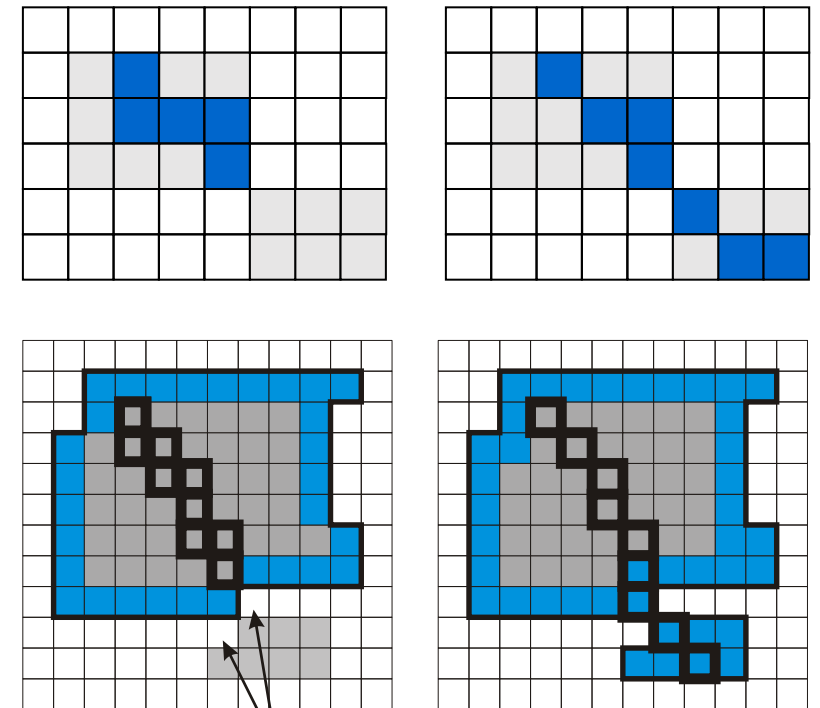
- Quadratische (rechteckige) Gitter:
 - *4-Nachbarschaft*: Pixel haben eine gemeinsame Seite.
 - *8-Nachbarschaft*: Pixel haben eine gemeinsame Seite oder einen gemeinsamen Eckpunkt.
- Hexagonale Gitter:
 - *6-Nachbarschaft*: Pixel haben eine gemeinsame Seite und zwei gemeinsame Eckpunkte.



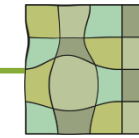


Pfad und zusammenhängendes Gebiet

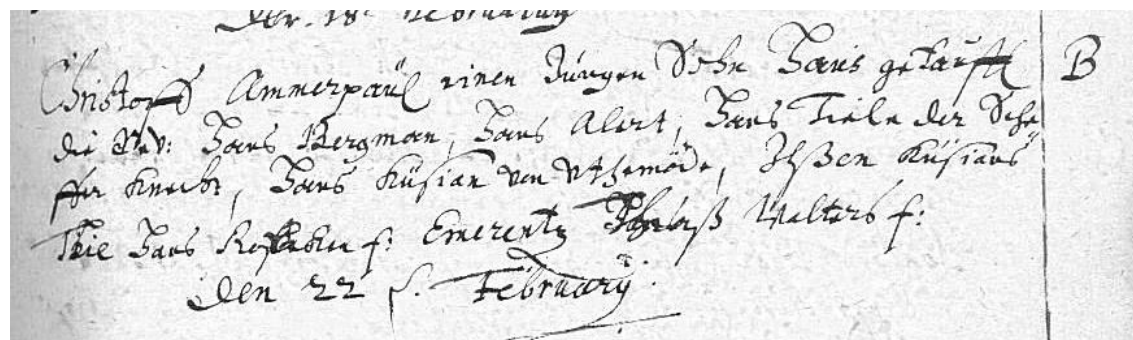
- *Pfad*: Folge von benachbarten Pixeln, die eine Homogenitätsbedingung erfüllen.
 - 4-benachbarte Pfade unterscheiden sich von 8-benachbarten Pfaden.
- *Zusammenhängendes Gebiet*: Menge aller Pixel zwischen denen Pfade existieren.
- Die Nachbarschaftsdefinitionen in Vorder- und Hintergrund müssen unterschiedlich sein.



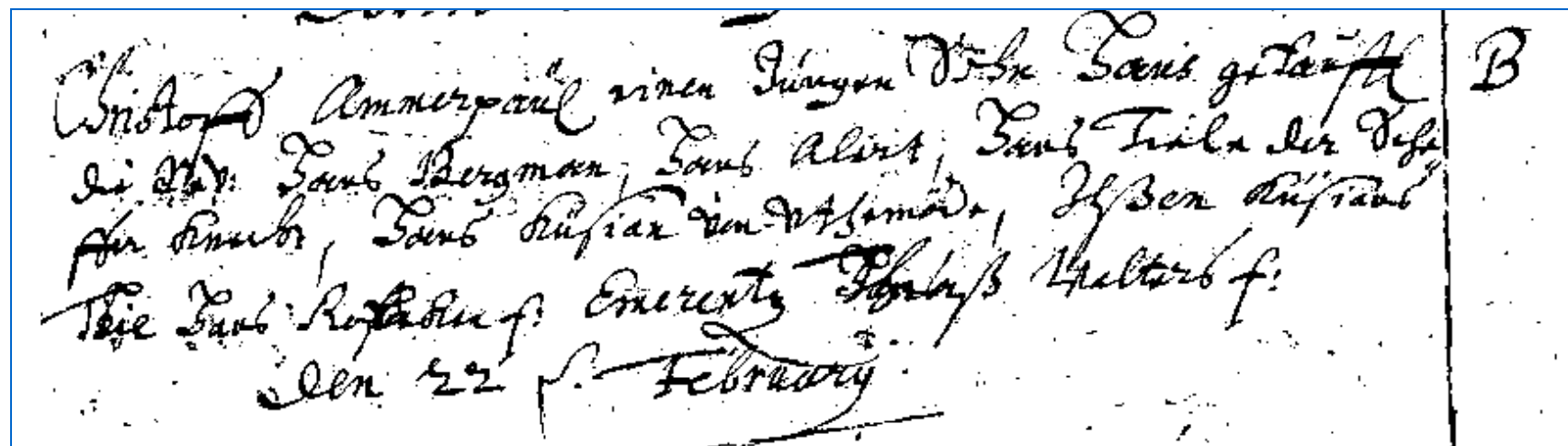
Hier zwischen sollte ein Pfad sein

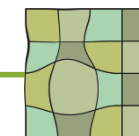


Bildanalyse und Zusammenhängende Gebiete



Geeignete Homogenitätskriterien
erzeugen zusammenhängende Gebiete,
deren Merkmale mit der gesuchten
Semantik korrelieren



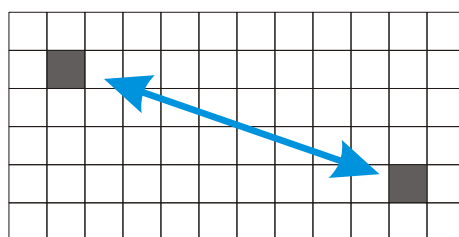


Abstandsmaße

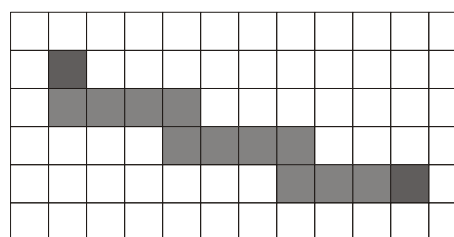
$$D_e(x_1, y_1; x_2, y_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$
 Euklidische Distanz

$$D_4(x_1, y_1; x_2, y_2) = |x_1 - x_2| + |y_1 - y_2|$$
 Cityblock-Distanz

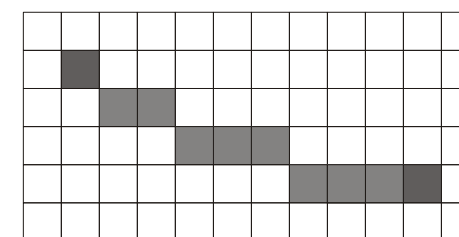
$$D_8(x_1, y_1; x_2, y_2) = \max(|x_1 - x_2|, |y_1 - y_2|)$$
 Schachbrett-Distanz



Euklidische Distanz



Cityblock-Distanz



Schachbrettdistanz

D_4 bzw. D_8 ist die kürzeste Distanz zwischen zwei Pixeln unter 4- bzw. 8-Nachbarschaft



Exkurs: Normen

- Die drei Distanzmaße sind Vektornormen.
- Die Norm $\|a\|$ eines Vektors a ist eine Zahl.
- Bedingungen für Normen:

$$\|\vec{a}\| \geq 0, \|\vec{a}\| = 0 \Rightarrow \vec{a} = \vec{0}$$

$$\|c \cdot \vec{a}\| = |c| \cdot \|\vec{a}\|$$

$$\|\vec{a} + \vec{b}\| \leq \|\vec{a}\| + \|\vec{b}\|$$

- Lp-Normen:

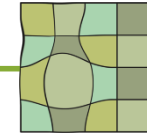
$$\|\vec{a}\|_p = \left(\sum_{i=1}^N |a_i|^p \right)^{\frac{1}{p}}, \|\vec{a}\|_p \geq \|\vec{a}\|_q, \text{ falls } p < q$$

- L1-Norm: Cityblock-Distanz, L2-Norm: euklidische Distanz, L^∞ -Norm: Schachbrett-Distanz



Gängige Bildformate

- Bilder können unterschiedlich komfortabel gespeichert werden.
- Die meisten Graphikprogramme und Graphikentwicklungsumgebungen bieten Filter zum Lesen und Schreiben gängiger Formate an.
- Gängige Formate sind zum Beispiel
 - PPM, PGM, PBM: portable pixel/grey/bit map (SUN)
 - BMP: Windows Bitmap
 - PNG: Portable Network Graphics
 - TIFF, TIF: Tagged Image File Format
 - JPG, JPEG: Joint Photographers Expert Group



PPM, PGM (, PBM)

- Reihe von proprietären, sehr einfachen Formaten von SUN
- Keine Datenkompression
- ASCII- oder als Binärdatei
- Struktur:
 - id: P1,P4 (PBM); P2,P5 (PGM); P3,P6 (PPM)
 - Bildbreite, Bildhöhe
 - Maximaler Grauwert
 - Pixelwerte
 - (Kommentarzeilen, beginnend mit #)
- Beispiel (Grauwertdatei)

```
P2
10 5
10
0 0 0 0 0 0 0 0 0 0
1 2 2 2 2 2 2 1 1 1
2 4 4 4 7 7 7 7 7 7
1 2 2 2 2 10 10 10 10 2
0 0 0 0 0 0 0 0 0 0
```



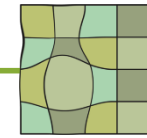
BMP

- Von Microsoft mit Windows eingeführt
 - Unterstützt unterschiedliche Bittiefen von 1,4,8,16,24, oder 32 Bit per Pixel (bpp)
 - Gebräuchlich sind 8 Bit (für Grauwertbilder) und 24 (für 3-Kanal-Farbbilder)
 - Bilddaten werden durch Run-Length-Encoding komprimiert (VL 7 Bildkompression)
 - Es können Lookup-Tabellen (bei Bittiefe 1,4,8) für die Repräsentation von Farbbildern gespeichert werden.
- Es existieren mehrere Versionen des BMP-Formats (Version 3,4,5)
 - Die meisten BMP-Reader unterstützen Version 3 und 8 bzw. 24 bpp-Repräsentationen.
 - Transparenzkanal (Alpha-Kanal) ist in Version 3 nicht vorgesehen (wohl aber in Version 4 und 5), auch wenn einige Programme das letzte Byte einer 32-bpp-Repräsentation so interpretieren.



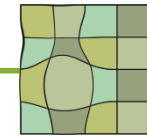
PNG

- Ersatz für GIF, aber ohne Möglichkeit zur Speicherung von Animationen (eigenes Format MNG; wurde nie standardisiert)
- wird von Web-Browsern unterstützt
- Pixelformate
 - Farbpalettenbilder mit 8-Bit-Pixeln
 - 1-, 4-, 8-, 16-Bit Grauwertbilder
 - 3×8 oder 3×16-Bit-Farbbilder (RGB)
 - Alpha-(Transparenz-)kanal mit 8 bzw. 16-Bit
- Kompression
 - Zweistufig: Vorfilterung durch Differenzbildung, anschließend ZIP-Kompression (Kombination aus Lempel-Ziv und Huffman-Kodierung)
 - Kompressionsrate besser als BMP, auch – durch Vorfilterung – besser als bei GIF



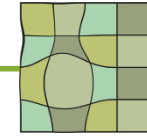
TIFF

- Sehr variables (und komplexes) Format
 - Ein oder mehrere Bilder (z.B. auch Thumbnail-Bild und original)
 - Bildpunkte können Einzelwerte (Grauwerte) oder Gruppen von Einzelwerten (z.B. RGB) sein
 - Bildwerte können Integer oder Floats sein
 - unterstützt verschiedene Farbmodelle
 - unterstützt eine Reihe von (verlustfreien) Kompressionsmethoden
 - Alpha- (Transparenz-) Werte können gespeichert werden
- Realisation
 - Nach Identifikation (u.a. durch Magic Number 42) folgt Offset-Angabe zum ersten Bild
 - Bilder werden durch eine Folge von Tags repräsentiert
 - Tags bestehen aus ID (z.B. Anzahl der Pixel je Zeile) gefolgt von den tag-spezifischen Daten
 - Nach dem Bild folgt Offset zum nächsten Bild
- Verwendung
 - überall da, wo hochwertige Bilder benötigt werden (z.B. Druckindustrie)



JPEG

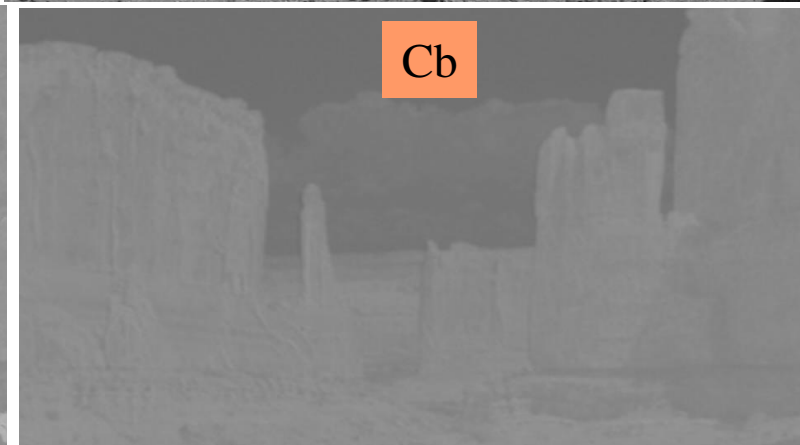
- Standardisiertes Format zur effizienten, aber verlustbehafteten Speicherung von Bildern
- Speicherung von Grauwert- und Farbbildern
- Farbbildrepräsentation als YCbCr-Repräsentation (PAL)
 - Y: gewichtete Kombination von R,G,B (entspricht etwa dem Grauwertbild)
 - Cb, Cr: um Y-Wert korrigierter Blau- bzw. Rot-Anteil im Bild (Farbabweichungen)
- Algorithmus zur Erzeugung der JPEG-Repräsentation (Details in VL 7: Kompression)
 - Umrechnung in YCbCr-Raum
 - Verlustbehaftete Filterung der Cb-, Cr- Anteile
 - Einteilung des Y-Bilds in 8×8-Blöcke und verlustbehaftete Kompression dieser Blöcke
 - Verlustfreie Kompression des Restsignals

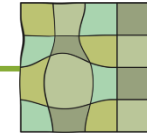


Exkurs YCrCb-Farbraum

- Abbildungsvorschrift

$$\begin{pmatrix} Y \\ Cr \\ Cb \end{pmatrix} = \begin{pmatrix} 0 \\ 128 \\ 128 \end{pmatrix} + \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$





JPEG Blockkompression

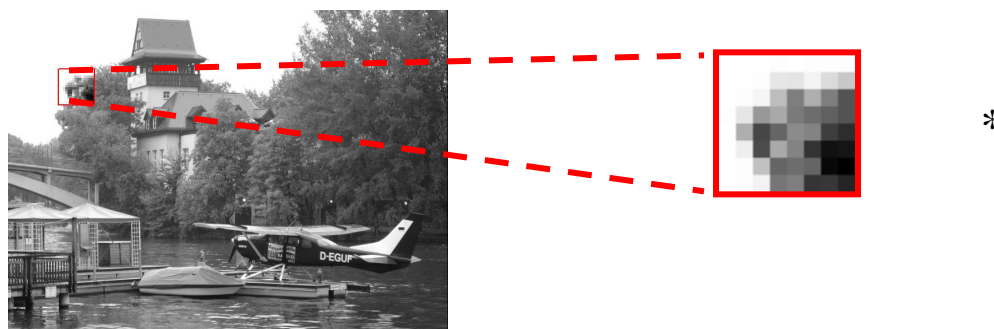
- Bild wird Block für Block komprimiert gespeichert
- Blockgröße ist 8×8 Pixel
- Kompression des Blocks durch Transformation in effizientere Repräsentation

Blockzerlegung

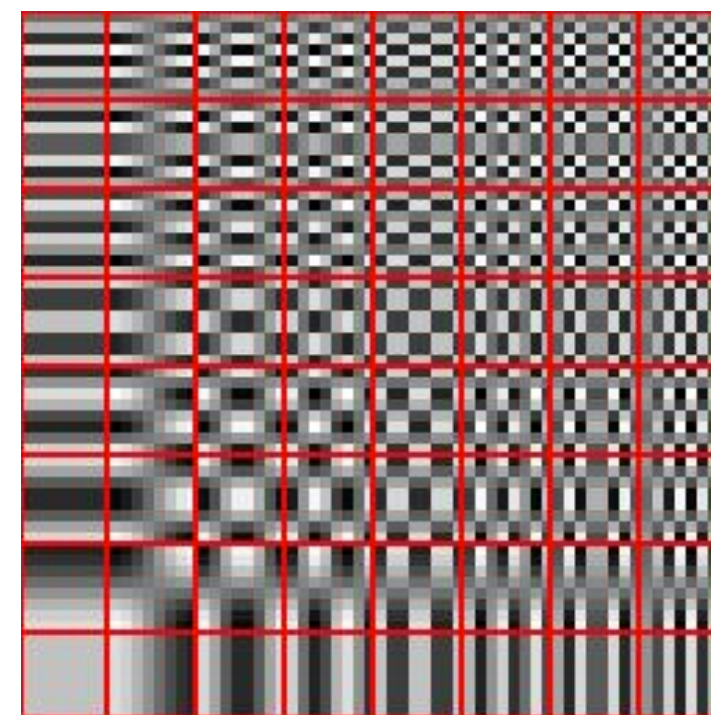


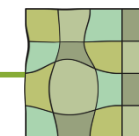
Transformation des Blocks

- Bildung gewichteter Summen mit sogenannten Basisfunktionen
- Es gibt soviel Basisfunktionen, wie es Pixel in einem Block gibt
- Jede Gewichtung mit einer Basisfunktion ergibt genau einen Wert

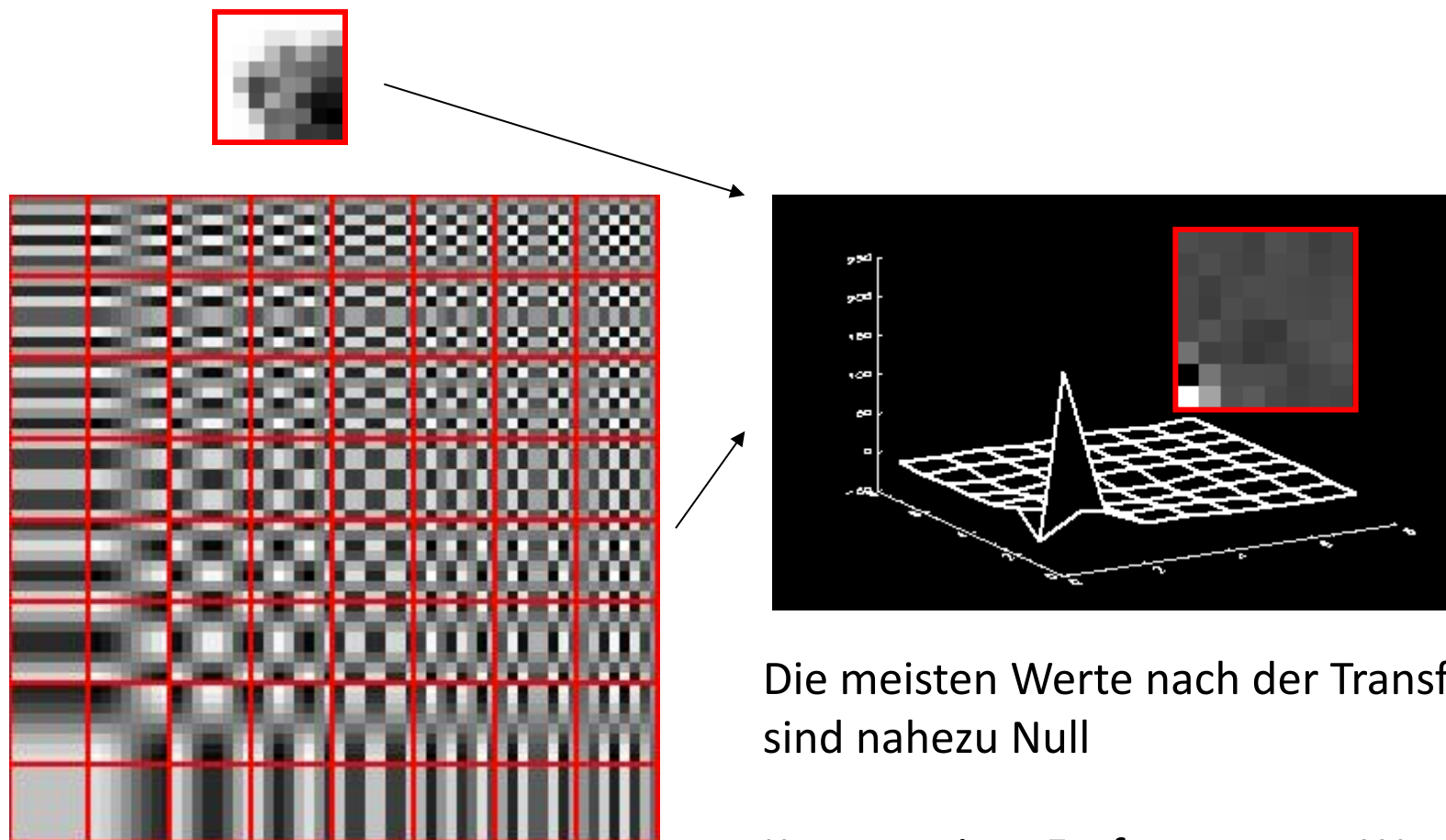


Die 64 Basisfunktionen (je 8×8 Pixel groß)
für die JPEG-Kompression



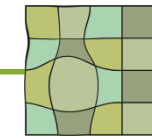


Resultat



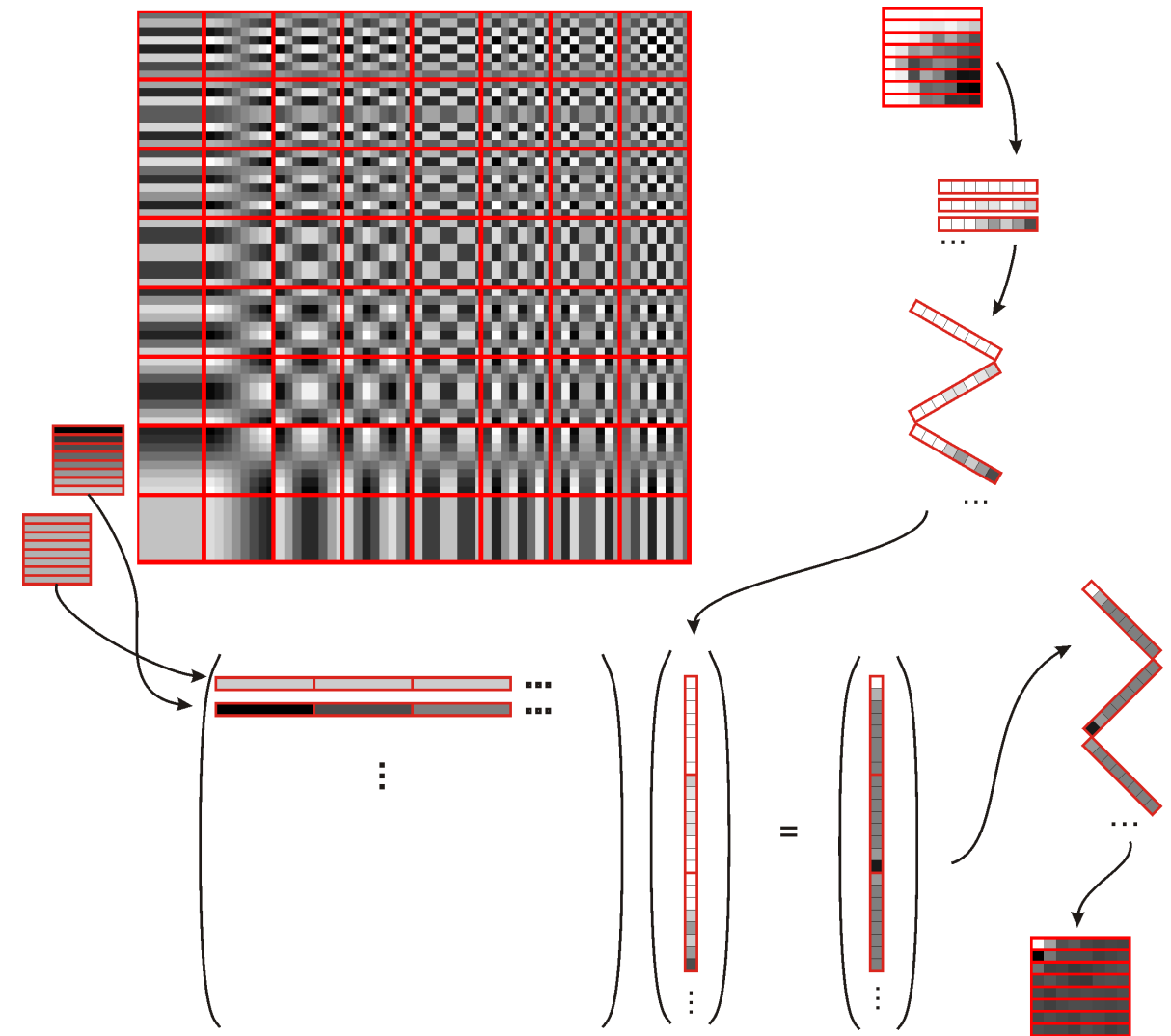
Die meisten Werte nach der Transformation
sind nahezu Null

Kompression: Entfernung von Werten $< \epsilon$



Berechnung

- Bild als Vektor
- Basisfunktionen als Sequenz von Vektoren in Matrix
- Operation als Matrixmultiplikation
 $Ax=b$





Fragen

- Wie verändert sich das Bild, wenn Werte nahe Null **gelöscht** werden?
- Wie kann man aus dem transformierten Bild das Original zurückgewinnen (**Invertierung** von **A**)
- Trennt die Basis **A zuverlässig** wichtige und unwichtige Elemente (Operator für relevante Bildanteile)?
- Ist die Transformation für **beliebige Bilder** (und auch für beliebige Größen der Basismatrizen) ausführbar?



Was Sie heute gelernt haben sollten

- Was sind digitale Bilder, wie entstehen digitale Bilder
- Grundlegende Eigenschaften von digitalen Bildern
- Gängige Bildformate
- Einstieg in lineare Bildtransformationen



Famous Last Question

Wie könnte diese Information “unsichtbar” in das RGB-Bild eingebettet werden?

