

Digitale Bildverarbeitung (DBV)

Prof. Dr.-Ing. Heinz-Jürgen Przybilla

Labor für Photogrammetrie

Email: heinz-juergen.przybilla@hs-bochum.de

Tel. 0234-32-10517

Sprechstunde: Montags 13 – 14 Uhr
und nach Vereinbarung

Bildspeicherung - Datenstrukturen -

Datenstrukturen



- Bilddaten sind flächenhaft angeordnete und miteinander korrelierte Zahlenwerte.
- Ihre Speicherung bedarf besonderer Strukturen und Formate.

- Die Strukturierung der Bilddaten unterliegt dabei folgenden Zielen:
 - Datenreduktion / -kompression
 - Schnelle Datenübertragung
 - Schneller Bildaufbau
 - Bildung von Pixelgruppen für Interpretationsaufgaben
 - Raster- / Vektorkonvertierung

■ Bildpunktfolge bei Farbbildern

Bildverschränkt

R G B R G B R G B R G B R G B R G B

...

Zeilenverschränkt

R R R R R R R R R R R R R R R R R R R
G G G G G G G G G G G G G G G G G G G
B B B B B B B B B B B B B B B B B B B

...

Bildverschränkt (1 oder mehrere Dateien)

R R R R R R R R R R R R R R R R R R R
R R R R R R R R R R R R R R R R R R R

...

G G G G G G G G G G G G G G G G G G G
G G G G G G G G G G G G G G G G G G G

...

B B B B B B B B B B B B B B B B B B B
B B B B B B B B B B B B B B B B B B B



■ Differenzbildung

- Bildung von Grauwertdifferenzen zwischen benachbarten Pixeln.
- Die Speicherung der Differenzen erfordert wegen der kleineren Zahlenwerte der Differenzen eine geringere Speichertiefe.

- **Run Length Encoding (RLE – Lauflängen-codierung)**
 - Wiederholen sich benachbarte Grauwerte mehrfach, so kann der Grauwert zusammen mit der Kettenlänge (Anzahl aufeinanderfolgender identischer Grauwerte) gespeichert werden.



■ RLE

Zeichenkette: "AAAAAAAAAAAAAAAAA"

RLE Kodierung: (15) (A)

- Der Wert (15) wird im ersten Byte, dem "count" Byte, gespeichert.
- Der Wert (A) wird im zweiten Byte, dem "symbol" Byte, gespeichert.

■ RLE

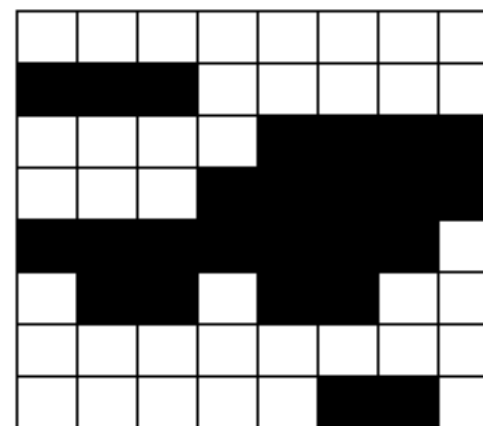
Zeichenkette: "AAAAAAAAAAAAAAAAA"

RLE Kodierung: (15) (A)

- Nach der RLE Komprimierung benötigt die Speicherung der Zeichenkette 2 Bytes statt 15 Bytes.
- Kompressionsfaktor von: $15 / 2 = 7.5$

■ RLE für ein Binärbild

- Hintergrund und Objektpixel werden zeilenweise alternierend gezählt.
- Start mit dem Hintergrundpixel
- Abwechselndes Zählen der Hintergrund (weiss) u. Objektpixel (schwarz)



8
0, 3, 5
4, 4
3, 5
0, 7, 1
1, 2, 1, 2, 2
8
5, 2, 1



- **RLE**
- Anwendung:
 - Strichzeichnungen (Binärbilder)
 - Bei Grauwertbildern häufig Basis für weiterführende Berechnungen, z.B. Erkennung zusammenhängender Flächen, Texturermittlung

- **Quad-Tree (Baumstruktur)**
 - Die Strukturierung der Grauwerte erfolgt in quadratischen Flächen gleicher Grauwerte bzw. Grauwertgruppen.
 - Ausgangsbasis ist das Originalbild, dessen Seitenlängen auf einen 2er Potenzwert erweitert, z.B. 850 Pixel → 1024 (2^{10}).

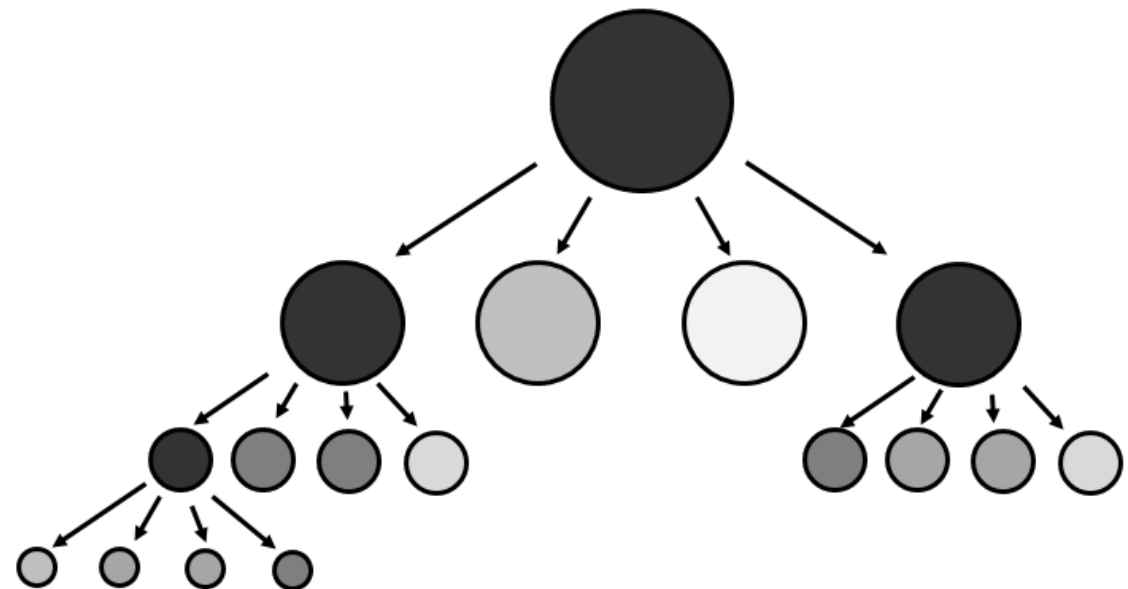


- **Quad-Tree**

- Das erweiterte Bild wird in 4 Quadranten aufgeteilt, welche ggf. weiter unterteilt werden.

■ Quad-Tree

130	131	132	132	128	128	128	128
131	132	132	132	128	128	128	128
132	132	129	129	128	128	128	128
132	132	129	129	128	128	128	128
130	130	130	130	132	132	131	131
130	130	130	130	132	132	131	131
130	130	130	130	131	131	129	129
130	130	130	130	131	131	129	129





- **Quad-Tree**
- Anwendung:
 - Datenkompression
 - Bildsegmentierung
 - Insbesondere bei grauwertreduzierten Bildern



■ Richtungsketten

- Ausgehend von einem Startpixel werden die Nachbarn (N8) mit identischen bzw. ähnlichem Grauwert ermittelt.
- Aus den Richtungsketten werden nachfolgend Richtungsvektoren abgeleitet.



■ **Richtungsketten**

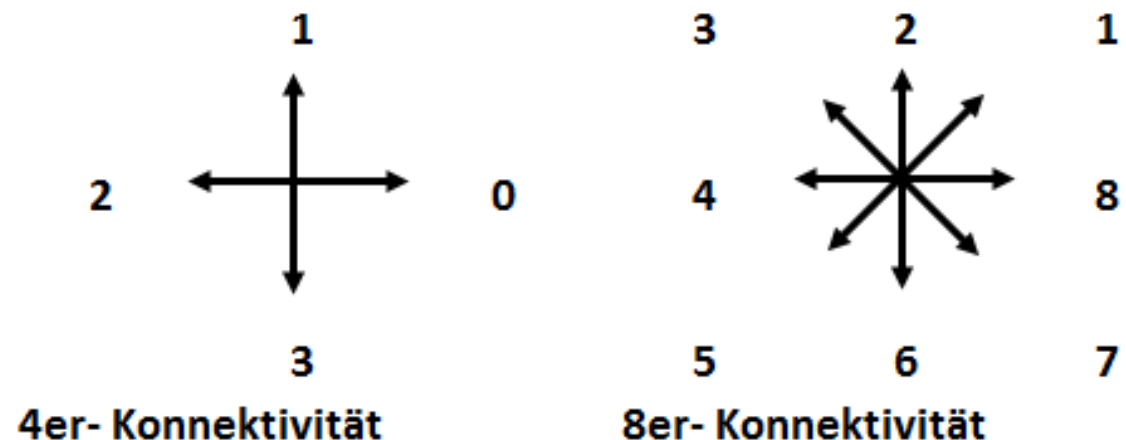
(Chain Code, Freeman Code, Richtungscode)

- Zur Berechnung von Objekten in Binärbildern.
- Beginnend mit einem vorab bekannten Startpunkt folgt ein Kettencode der Objektkontur, typischerweise entgegen dem Uhrzeigersinn.

■ Richtungsketten

(Chain Code, Freeman Code, Richtungscode)

- Abhängig von der Konnektivität ergeben sich unterschiedliche Richtungen für Folgepixel:



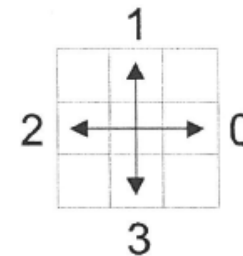
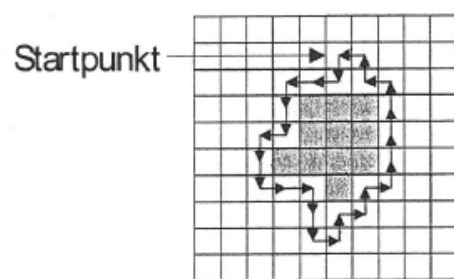
■ Richtungsketten

(Chain Code, Freeman Code, Richtungscode)

- Den Richtungen werden Zahlen zugewiesen, die auch als Freeman-Codes bezeichnet werden.
- Bei 4er-Konnektivität lässt sich die Richtung in 2 Bits, bei 8er-Konnektivität in 3 Bits speichern.

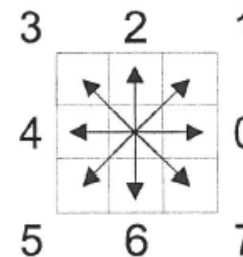
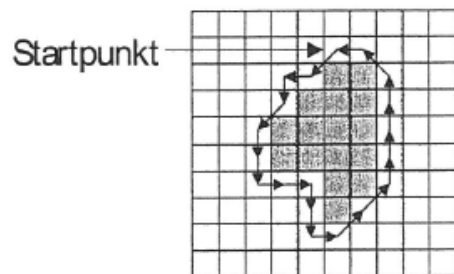
■ Richtungsketten

4er-Konnektivität



3,2,2,3,3,2,3,3,0,0,3,3,0,1,0,1,0,1,1,1,2,1,2

8er-Konnektivität



5,4,6,5,6,6,0,0,6,6,0,1,1,2,2,2,2,3,4



- **Richtungsketten**
- Anwendung:
 - Ermittlung einheitlicher Flächen und Linien mit Vektorisierung (Umringspolygon, Linienpolygon)
 - Raster- / Vektorkonvertierung



■ Bildpyramiden

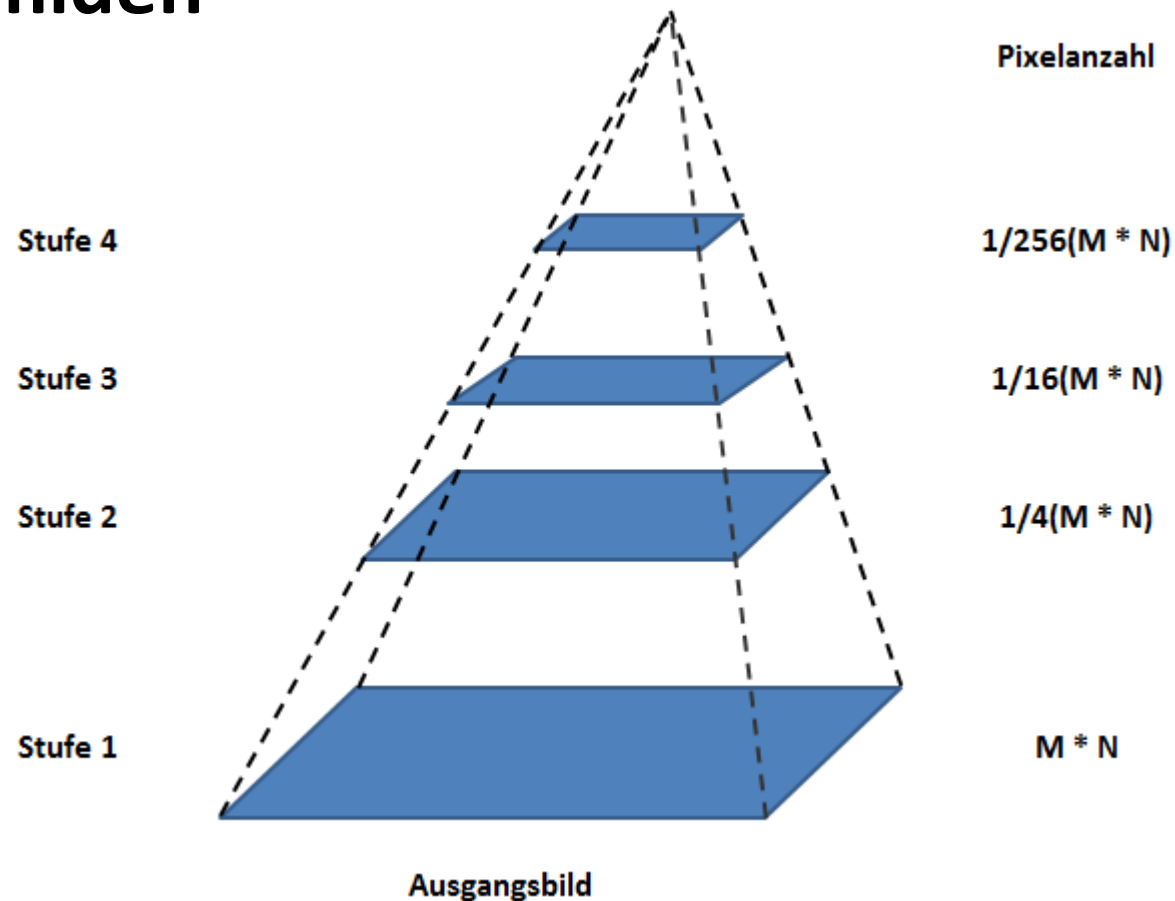
- Zur Bildung von Bildpyramiden wird das Originalbild in der Seitenlänge auf einen 2^n -Wert erweitert.
- Danach werden neue Bilder mit aggregierten (zusammengefassten) Pixelwerten gebildet.



■ Bildpyramiden

- Es entsteht eine Bildfolge (Pyramide) in verschiedenen Auflösungsstufen.
- Die neuen Bilder werden i.d.R. in einer zusätzlichen Datei gespeichert.
- Insgesamt entsteht eine größere Datenmenge (ca. 30% gegenüber dem Originalbild).

■ Bildpyramiden



■ Bildpyramiden



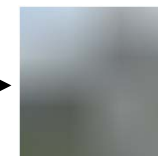
1024x1024



256x256



64x64



4x4



- **Bildpyramiden**
- Anwendung:
 - Schneller Bildaufbau beim Öffnen und Zoomen eines Bildes
 - Der Monitorauflösung entsprechend wird eine geeignete Pyramidenstufe zur Darstellung gewählt