

VO Videoverarbeitung

Vorlesungseinheit am 24.4.2009:

Bildaufnahme – Räumliche Farbinterpolation

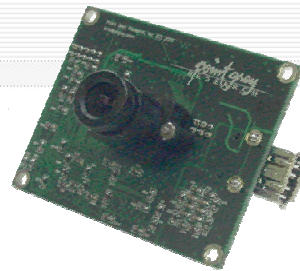
SS 2009

Ao. Univ.-Prof. Margrit Gelautz

(gelautz@ims.tuwien.ac.at)

Bildaufnahme – Thema: Räumliche Farbinterpolation

- Hintergrund
 - Aus Gründen der Kostenersparnis besitzen Digitalkameras häufig nur einen Bildsensor (*single detector*).
- Prinzip
 - Durch Verwendung eines Farbfilters (*Color Filter Array [CFA]*) zwischen Linse und Sensor wird Farbinformation gewonnen.
 - Dabei wird nach einem fixen Schema verschiedenen Bildpunkten der Intensitätswert einer einzelnen Farbkomponente zugeordnet.
 - Zur Rekonstruktion der Farb-Originalbilder sind Methoden zur Berechnung der anderen Farbkomponenten an den jeweiligen Pixelpositionen, an denen diese Farbkomponente nicht gespeichert wurde, erforderlich (*spatial color interpolation* oder *color demosaicking*).



Beispiel einer Kamera

- Dragonfly (Firma Pointgrey)
- Spezifikationen
 - *6-pin IEEE-1394 interface*
 - *Multiple frame rates*
 - *Automatic camera-to-camera synchronization*
 - *640x480 or 1024x768 24-bit true color or 8-bit gray scale images*
 - ...
- Unterstützte Formate (Auszug aus dem Datenblatt)
 - *B&W models: 8-bit Mono*
 - *Color models: 8-bit Bayer tiled image (color space conversion done on the host computer)*

Bayer Pattern

G1	R2	G3	R4	G5
B6	G7	B8	G9	B10
G11	R12	G13	R14	G15
B16	G17	B18	G19	B20
G21	R22	G23	R24	G25

- Es gibt viele verschiedene Color Filter Array Anordnungen
- In der Praxis sehr häufig verwendet wird das *Bayer Pattern*, welches die 3 primären Farben rot, grün und blau für die Filter-Elemente verwendet.
- Die zugehörige Anordnung der Farbfilter-Elemente ist in der Abbildung links gezeigt.

Vergleich: Aufgenommenes Bild und berechnetes Bild



Vom Bildsensor aufgenommenes Intensitäts-/Grauwertbild (*stippled image*).

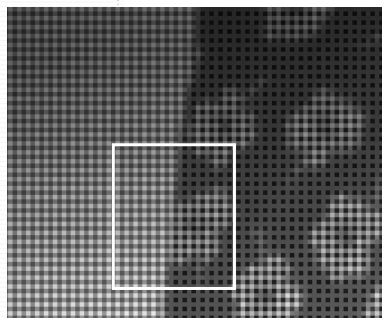


Daraus berechnetes (rekonstruiertes) Farbbild.

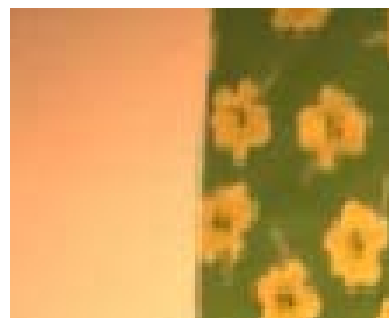
VO Videoverarbeitung (Bildaufnahme & Farbinterpolation)

5

Detailansicht 1



Stippled image (Ausschnitt)

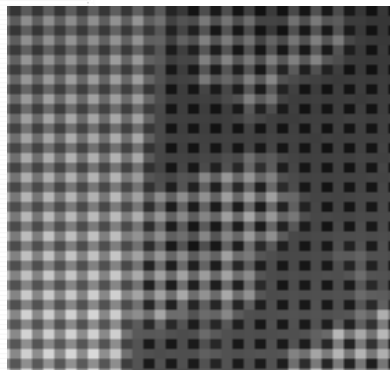


Farbbild (Ausschnitt)

VO Videoverarbeitung (Bildaufnahme & Farbinterpolation)

6

Detailansicht 2



Zoom (siehe Markierung vorige Folie)

G1	R2	G3	R4	G5
B6	G7	B8	G9	B10
G11	R12	G13	R14	G15
B16	G17	B18	G19	B20
G21	R22	G23	R24	G25

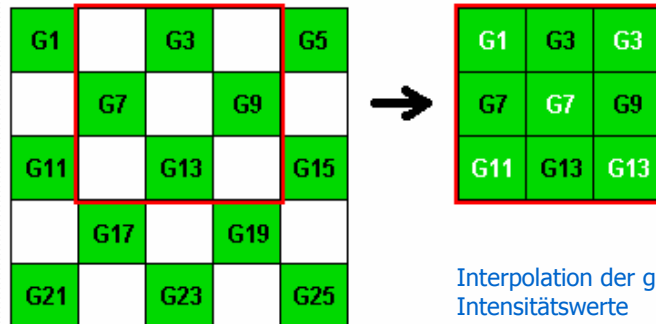
Bayer Pattern

Frage: Kann man im Bild links erkennen, wo die grünen Pixel liegen?

Algorithmen für räumliche Farbinterpolation

- Man unterscheidet zwischen 2 Arten von Algorithmen für räumliche Farbinterpolation:
 - Nicht-adaptive Algorithmen
 - Adaptive Algorithmen
- Nicht-adaptive Algorithmen führen die Interpolation für jedes Pixel nach einem festgelegten Schema durch.
 - Nearest Neighbor Replication
 - Bilinear Interpolation
 - ...
- Adaptive Algorithmen berücksichtigen lokale Eigenschaften der Pixel-Nachbarschaft
 - z.B. Einbeziehung von Kanteninformation

Methode 1: Nearest Neighbor Zuordnung



Interpolation der grünen Intensitätswerte

Zuordnung nach dem *nearest neighbor* Prinzip: In diesem Beispiel wird immer der linke Intensitätswert herangezogen. (Es könnte aber auch der obere, rechte oder untere Wert genommen werden.)

VO Videoverarbeitung (Bildaufnahme & Farbinterpolation)

9

Methode 2: Bilineare Interpolation

G1	R2	G3	R4	G5
B6	G7	B8	G9	B10
G11	R12	G13	R14	G15
B16	G17	B18	G19	B20
G21	R22	G23	R24	G25

- Interpolation grüner Pixel
 - Der Durchschnitt von oberem, unterem, linkem und rechtem Pixel zur Berechnung des zu interpolierenden Pixels: $G8 = (G3+G7+G9+G13) / 4$
- Interpolation roter/blauer Pixel
 - An einer Pixelposition, an der ein Grün-Wert gespeichert ist: Durchschnitt der beiden angrenzenden Pixel in der entsprechenden Farbe: $B9=(B8+B10) / 2$ $R9=(R4+R14) / 2$
 - An einer Pixelposition, an der ein Rot- oder Blau-Wert gespeichert ist: Durchschnitt der diagonal angrenzenden vier Pixel: $R8=(R2+R4+R12+R14)/4$ $B12=(B6+B8+B16+B18)/4$

VO Videoverarbeitung (Bildaufnahme & Farbinterpolation)

10

Ergebnisse der Interpolation



(a) Nearest neighbor

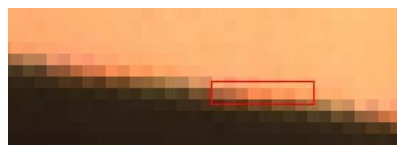


(b) Bilineare Interpolation

Probleme: Blockbildung (a), verschwommenes Bild (b), unschöne Effekte an den Kanten bei (a) und (b).

Vorteile: Geringer Rechenaufwand bei beiden Methoden, insbesondere bei (a).

Artefakte an Kanten



(a) Der sogenannte *zipper effect* an Kanten (z.B. verursacht durch bilineare Interpolation)



(b) Durch aufwändigere Interpolationsmethoden können ruhigere Farbwertübergänge erreicht werden.

Methode 3: Edge Sensing Interpolation - Prinzip

- Die Methode ist ein Beispiel eines adaptiven Algorithmus: räumliche Eigenschaften in Pixelnachbarschaften werden berücksichtigt und beeinflussen die Entscheidung, welche Pixel aus Pixelnachbarschaften zur Berechnung des interpolierten Wertes herangezogen werden.
- Es werden Gradienten gebildet um Kanteninformation im Bild zu berücksichtigen.
- Im folgenden wird die Vorgangsweise bei der Interpolation grüner Pixel basierend auf Gradienteninformation genauer beschrieben.

Methode 3: Edge Sensing Interpolation - Algorithmus

G1	R2	G3	R4	G5
B6	G7	B8	G9	B10
G11	R12	G13	R14	G15
B16	G17	B18	G19	B20
G21	R22	G23	R24	G25

• Interpolation grüner Pixel:

- Definieren von horizontalem und vertikalem Gradienten für jede rote und blaue Position.
z.B. für B8: $\Delta H = |G7 - G9|$ und $\Delta V = |G3 - G13|$
- Definieren eines Threshold Wertes T
- Berechnung der interpolierten Werte:
IF $\Delta H < T$ AND $\Delta V > T$ $G8 = (G7 + G9) / 2$;
ELSE IF $\Delta H > T$ AND $\Delta V < T$ $G8 = (G3 + G13) / 2$;
ELSE $G8 = (G3 + G7 + G9 + G13) / 4$;

Methode 3: Edge Sensing Interpolation - Ergebnis



(a) Zum Vergleich nochmals das Ergebnis der bilinearen Interpolation.



(b) Rekonstruktion mittels *edge sensing interpolation*: Man erkennt eine bessere (schärfere) Rekonstruktion der Kanten als in (a).