

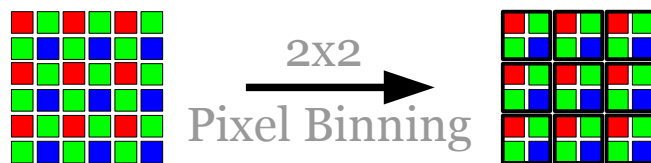
Debayeringverfahren

Thomas Noack, Nikolai Kosjar

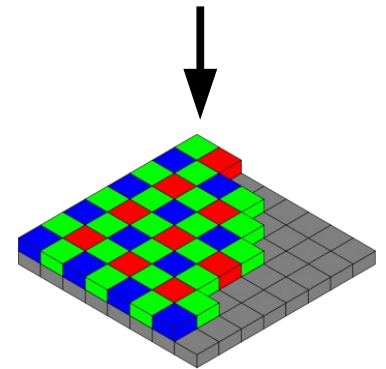
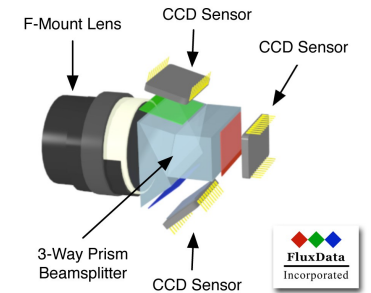
19. Mai 2010

„Was bisher geschah...“

- Reduktion der Herstellungskosten durch Einsatz von nur noch einem CCD-Sensor mit **Bayer-Filter**
- Problem: Bayer Image → „Full Color Image“
 - → Interpolation fehlender Farbwerte nötig
- Pixelbinning: Zusammenfassung von Pixeln



Schematic View of 3-CCD Camera



- **Aber:** Auflösung verringert, unschöne Artefakte, Farbfehler - andere Verfahren können das besser, richtig?



Agenda

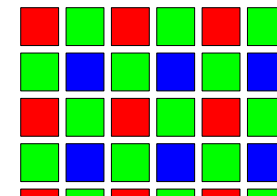
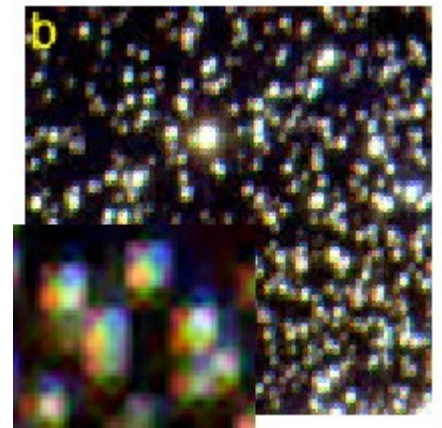
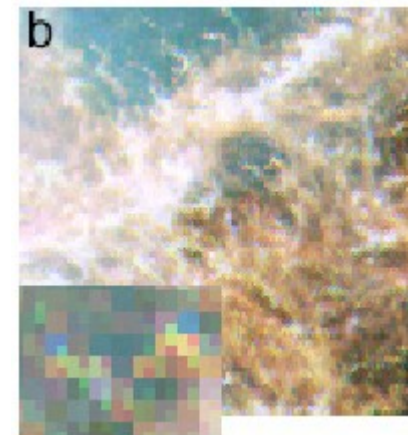
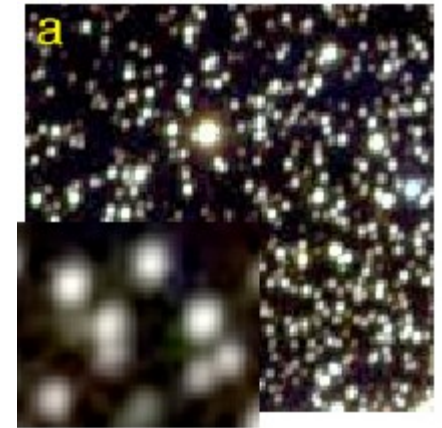
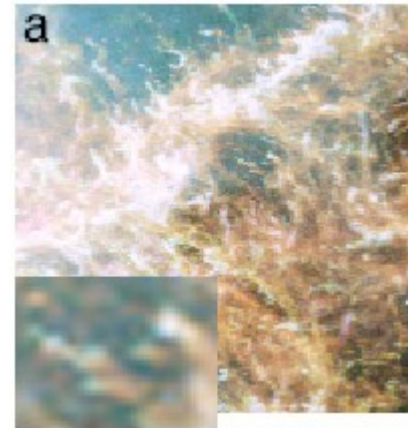
- Verbreitete Verfahren
 - Nearest Neighbour
 - Bilineare & bikubische Interpolation
 - Variable Number of Gradients (VNG)
 - Patterned Pixel Grouping (PPG)
- Microsoft Algorithmus als weiteres Verfahren
- Zusammenfassung

Agenda

- Verbreitete Verfahren
 - **Nearest Neighbour**
 - Bilineare & bikubische Interpolation
 - Variable Number of Gradients (VNG)
 - Patterned Pixel Grouping (PPG)
- Microsoft Algorithmus als weiteres Verfahren
- Zusammenfassung

Nearest Neighbour Interpolation

- Übernahme Farbwert vom benachbarten Pixel (mit gleichem Farbanteil)
- Einfachste Methode
- Schlechte Qualität
 - Auflösung wird verringert
 - Farbfehler an Kanten („color fringing“)

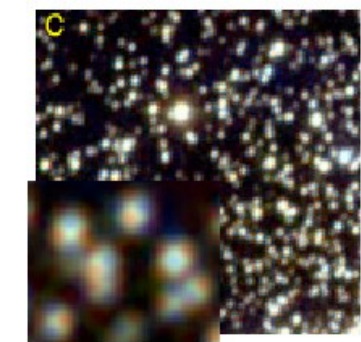
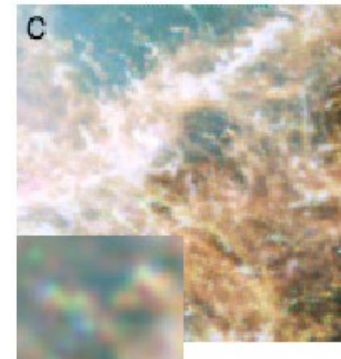
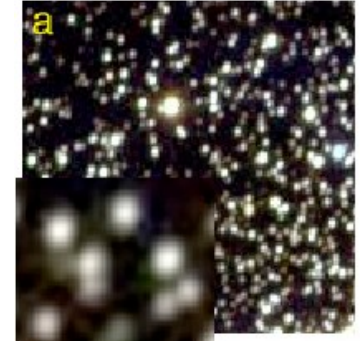
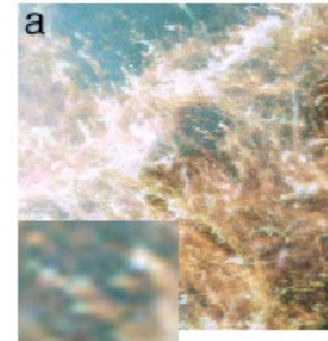


Agenda

- Verbreitete Verfahren
 - Nearest Neighbour
 - **Bilineare & bikubische Interpolation**
 - Variable Number of Gradients (VNG)
 - Patterned Pixel Grouping (PPG)
- Microsoft Algorithmus als weiteres Verfahren
- Zusammenfassung

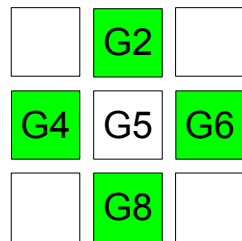
Bilineare Interpolation

- Bilde Mittelwert des benötigten Farbwertes aus der 2x2-Nachbarschaft
- Relativ schnell
- Bessere Qualität als Nearest Neighbour:
 - Weicher (Anti-Aliasing-Effekt)
 - Weniger Artefakte
 - Auflösung beibehalten
- Aber: Immer noch „Zipper-Artefakte“ an Kanten

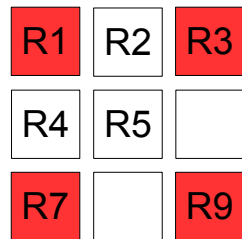


Bilineare Interpolation

- Beispiel



$$G5 = (G2+G4+G6+G8)/4$$



$$R2 = (R1+R3)/2$$

$$R4 = (R1+R7)/2$$

$$R5 = (R1+R3+R7+R9)/4$$

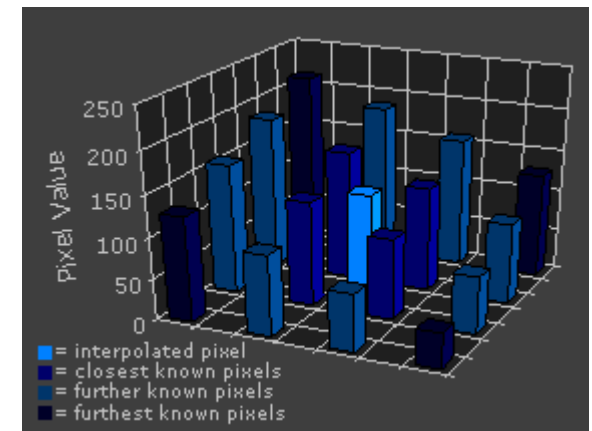
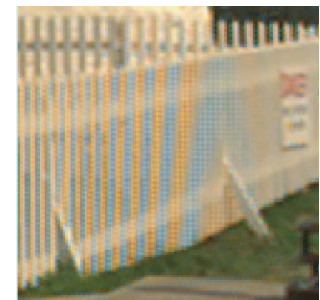
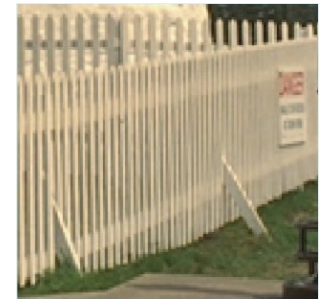
(für blau analog)

Agenda

- Verbreitete Verfahren
 - Nearest Neighbour
 - Bilineare & **bikubische Interpolation**
 - Variable Number of Gradients (VNG)
 - Patterned Pixel Grouping (PPG)
- Microsoft Algorithmus als weiteres Verfahren
- Zusammenfassung

Bikubische Interpolation

- Betrachte 4x4 Nachbarschaft: Bilde gewichteten Mittelwert - Pixel die näher dran, sind stärker gewichtet
- Mittlere Verarbeitungszeit
- Mittlere Qualität
 - Schärfere Bilder
 - Ideale Kombination aus Verarbeitungszeit und Qualität
- Aber: Immer noch „Zipper-Artefakte“ an Kanten

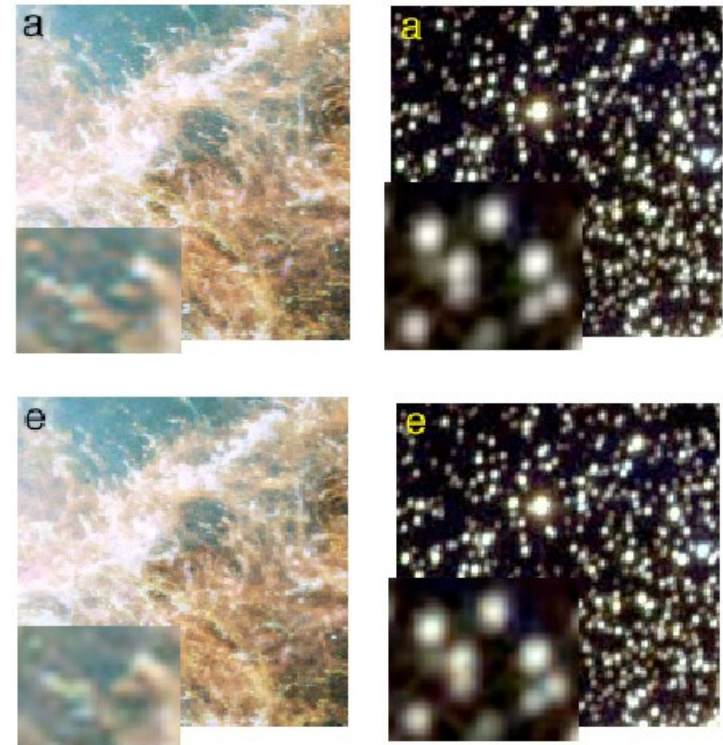


Agenda

- Verbreitete Verfahren
 - Nearest Neighbour
 - Bilineare & bikubische Interpolation
 - ➔ **Variable Number of Gradients (VNG)**
 - Patterned Pixel Grouping (PPG)
- Microsoft Algorithmus als weiteres Verfahren
- Zusammenfassung

Variable Number of Gradients

- Bestimme für jeden Pixel 8 gemittelte Gradients, wähle dann die „guten“ aus und bestimme aus diesen den fehlenden Farbanteil
- Sehr aufwendiges Verfahren
- Super Qualität, sehr nahe am Original



Variable Number of Gradients

- Einleitendes Beispiel:

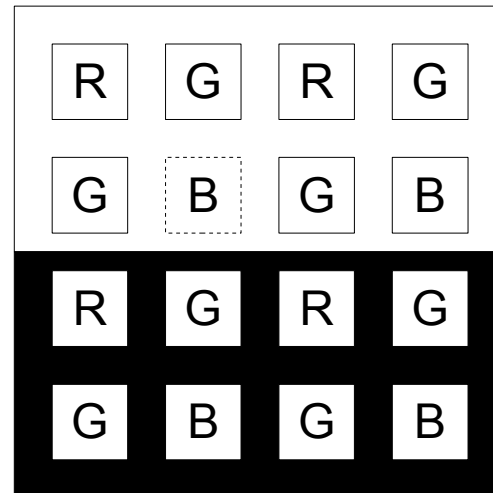
- Gesucht:** Grünwert bei

B

Ziel: Artefakte an Kanten-
übergängen reduzieren
(Zipper-Effekt)

Reales Bild
ist hier weiss

Reales Bild
ist hier schwarz



Variable Number of Gradients

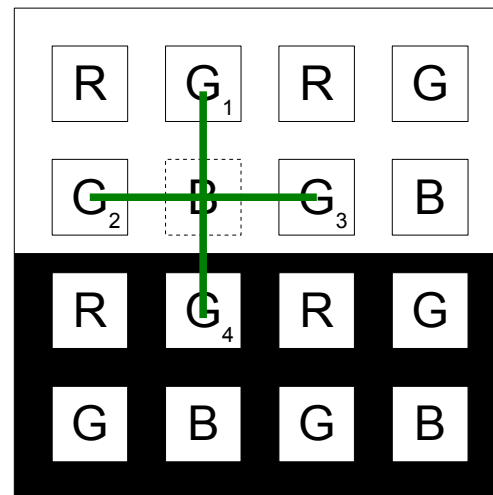
- Einleitendes Beispiel:

- Gesucht:** Grünwert bei B

Ziel: Artefakte an Kanten-
übergängen reduzieren
(Zipper-Effekt)

Reales Bild
ist hier weiss

Reales Bild
ist hier schwarz



$$|G1 - G4| > |G2 - G3|$$

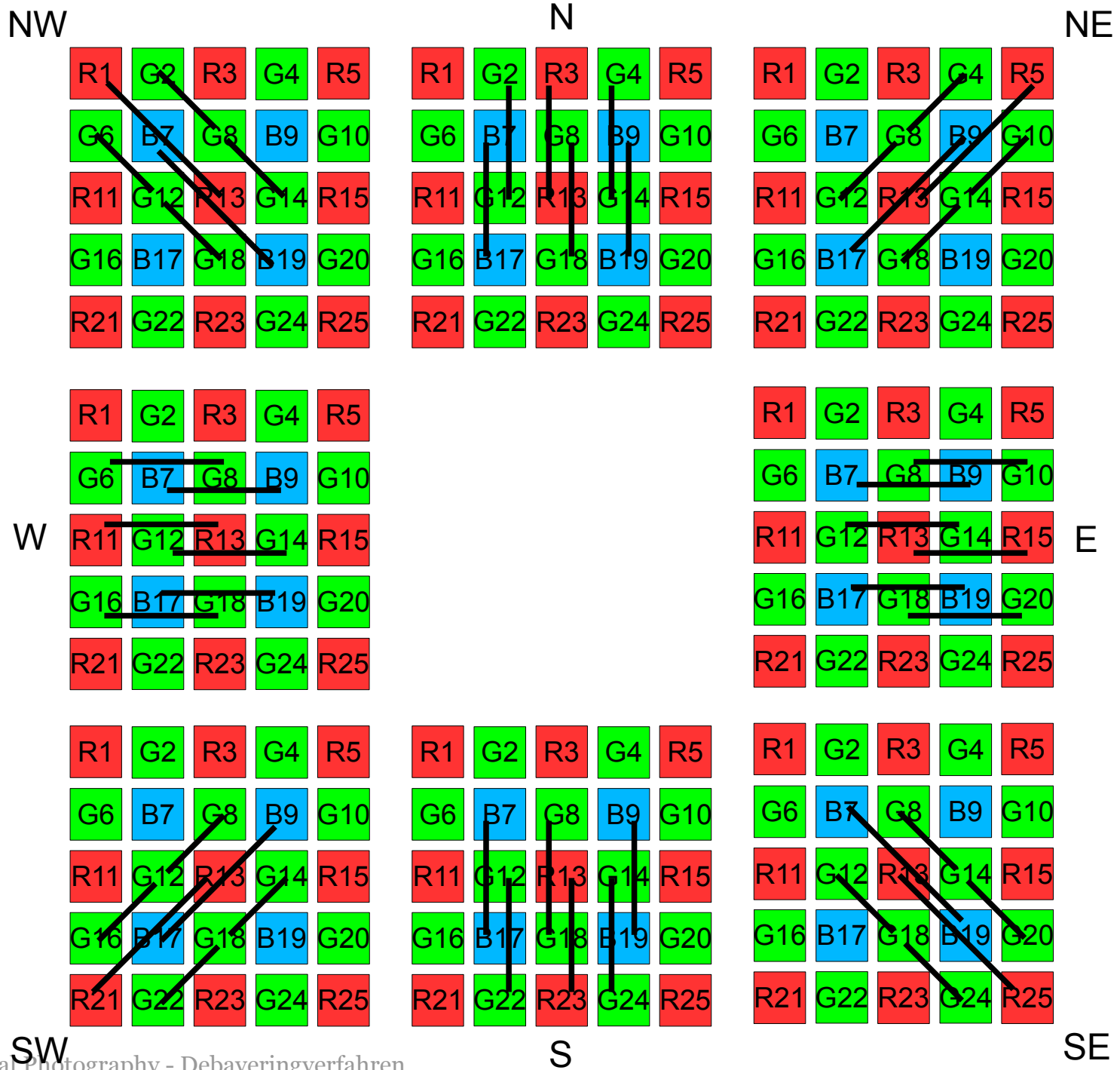
$|G1 - G4|$ ist ein stärkerer
Übergang, also bei Berechnung
lieber herausnehmen!

Variable Number of Gradients

- Beispiel
 - **Gesucht:** G13 und B13 bei R13

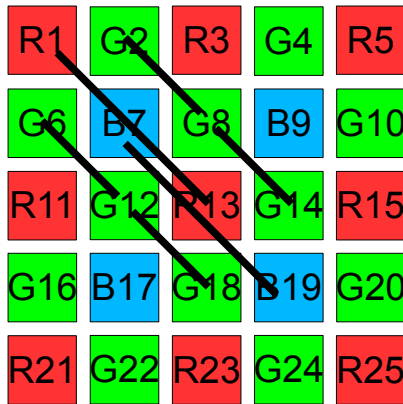
R1	G2	R3	G4	R5
G6	B7	G8	B9	G10
R11	G12	R13	G14	R15
G16	B17	G18	B19	G20
R21	G22	R23	G24	R25

■ **Gesucht: G13 und B13 an R13**

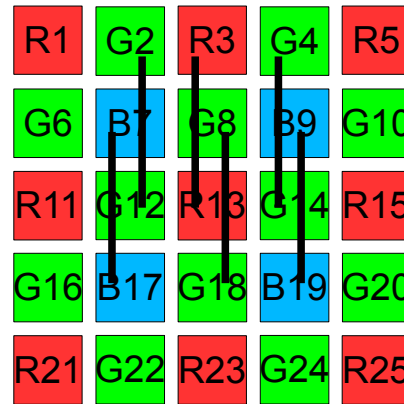


■ **Gesucht: G13 und B13 an R13**

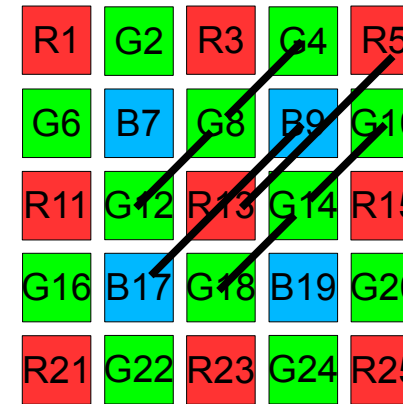
12 = NW



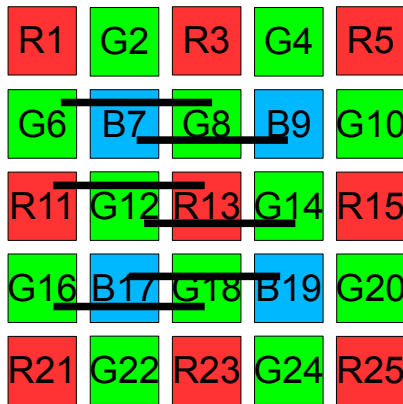
N = 12



NE = 4



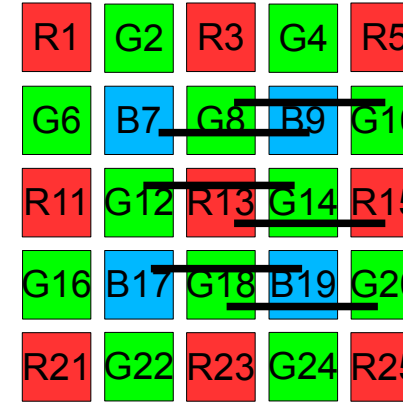
8 = W



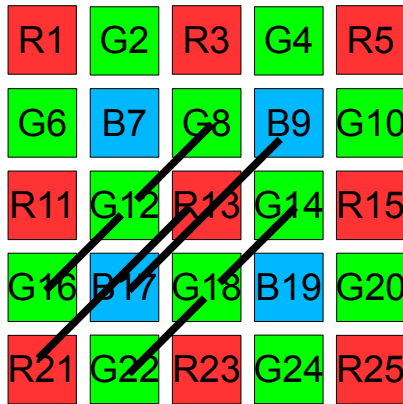
Schwellwert ermitteln:

$$T = k_1 * \min + k_2 * (\max - \min) = 11$$

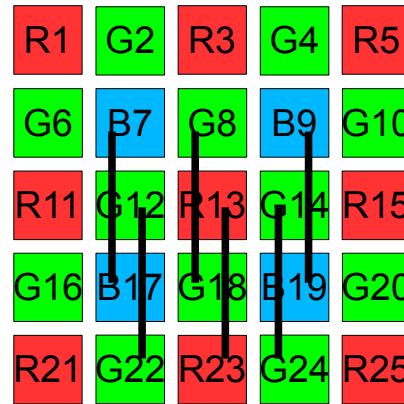
E = 13



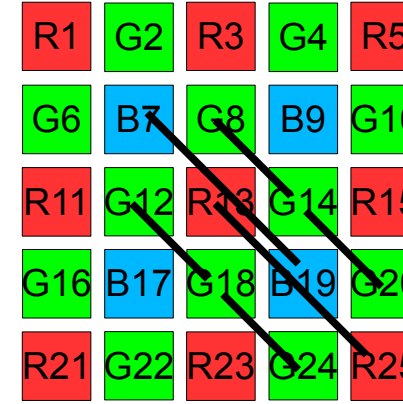
14 = SW



S = 7

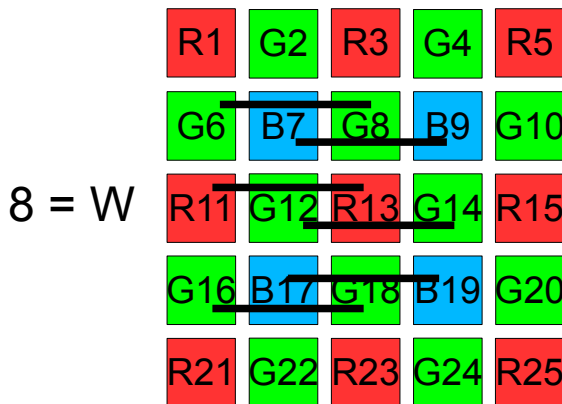
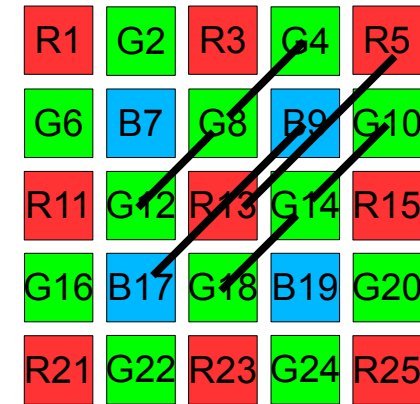


SE = 7



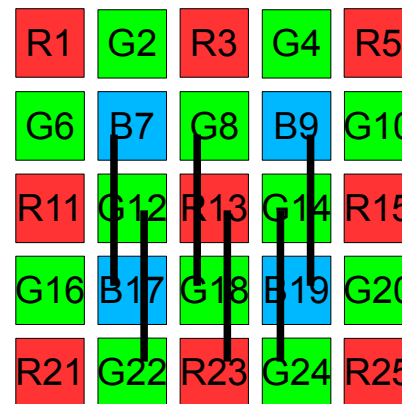
- **Gesucht: G13 und B13 an R13**

NE = 4

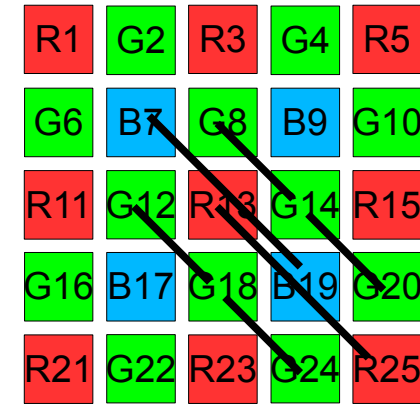


Schwellwert ermitteln:

$$T = k1 * \min + k2 * (\max - \min) = 11$$



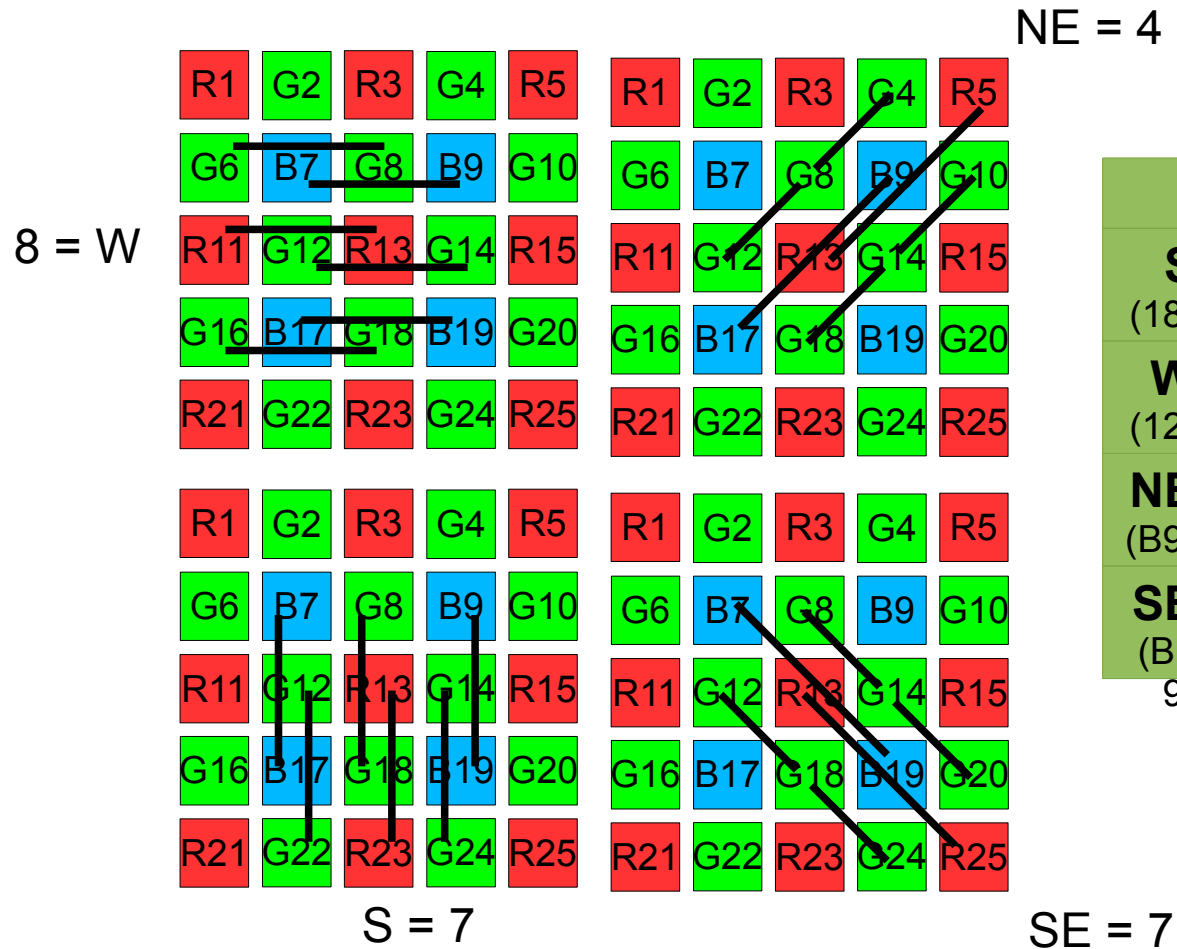
S = 7



SE = 7

Variable Number of Gradients

- **Gesucht:** G13 und B13 bei R13



	G	B	R
S (18)	G18	(B17+B19)/2	(R13+R23)/2
W (12)	G12	(B7+B17)/2	(R11+R13)/2
NE (B9)	(G4+G8+G10 +G14)/4	B9	(R13+R5)/2
SE (B19)	(G14+G18+G20+G24)/4	B19	(R13+R25)/2

$$\begin{array}{ccc} \parallel & \parallel & \parallel \\ G_s & B_s & R_s \end{array}$$

$$G13 = R13 + (G_s - R_s)/4$$

$$B13 = R13 + (B_s - R_s)/4$$

Agenda

- Verbreitete Verfahren
 - Nearest Neighbour
 - Bilineare & bikubische Interpolation
 - Variable Number of Gradients (VNG)
 - **Patterned Pixel Grouping (PPG)**
- Microsoft Algorithmus als weiteres Verfahren
- Zusammenfassung

Patterned Pixel Grouping

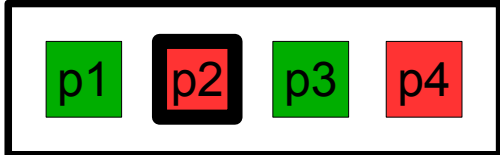
- Annahme:
- die Farbwerte korrelieren mit der Luminanz der Pixel
- da vor allem die grünen Pixel die Luminanz transportieren heißt dies, dass die Differenzen $|R-G|$ und $|B-G|$ sich nur allmählich ändern

Patterned Pixel Grouping

- Phase 1: Grün-Rekonstruktion
 - a) Gradientenberechnung
 - b) Interpolation
- Phase 2: B/R auf G Rekonstruktion
- Phase 3: B auf R / R auf B Rekonstruktion
 - a) Gradientenberechnung
 - b) Interpolation

Patterned Pixel Grouping (1a)

- Es werden 4 Gradienten (in diesem Fall für p2) berechnet, die durch das folgende Muster gebildet werden (Grün auf Blau analog):

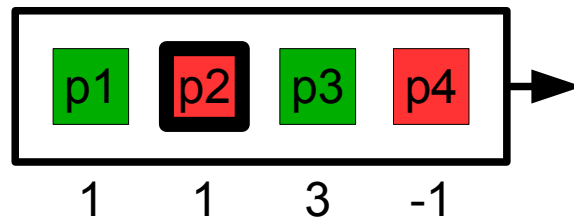


$\Delta X = 2 \cdot |R(p2) - R(p4)| + |G(p1) - G(p3)|$

- Dieses wird einmal in jede Richtung X aus {N, O, S, W} durch Drehung um 90° jeweils um den Punkt p2 durchgeführt
- Anschließend wird die Richtung Y mit dem niedrigsten Gradienten ausgewählt

Patterned Pixel Grouping (1b)

- Nun wird der Grünwert an p2 mit den Pixeln von Y errechnet, wobei die Pixel mit folgenden Gewichten in die Berechnung eingehen :



$$G(p2) = (G(p1) + 3 \cdot G(p3) + R(p2) - R(p4)) / 4$$

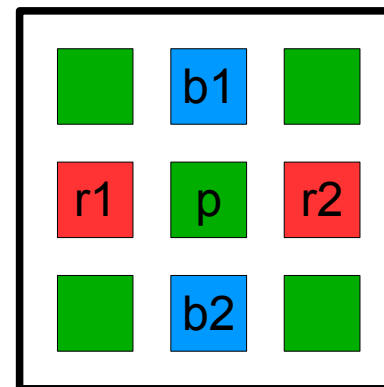
Patterned Pixel Grouping (2)

- Nun wird an den ursprünglich ausschließlich grünen Pixeln Blau und Rot mit der Funktion `hue_transit` interpoliert:

$$R(p) = \text{hue_transit}(G(r1), G(p), G(r2), R(r1), R(r2))$$

$$B(p) = \text{hue_transit}(G(r1), G(p), G(b2), B(b1), B(b2))$$

(analog in blau-grün Zeilen
um 90° gedreht)



Patterned Pixel Grouping

- function hue_transit(l1, l2, l3, v1, v3)

- Falls ($l1 < l2 < l3$ oder $l1 > l2 > l3$)



- Dann gib zurück:

- $$v1 + ((v3 - v1) \cdot (l2 - l1)) / (l3 - l1)$$

- Sonst gib zurück:

- $$(v1 + v3) / 2 + (2 \cdot l2 - l1 - l3) / 4$$

- *Anm: hue_transit kann horizontal, vertikal und diagonal angewendet werden.*

Patterned Pixel Grouping (3)

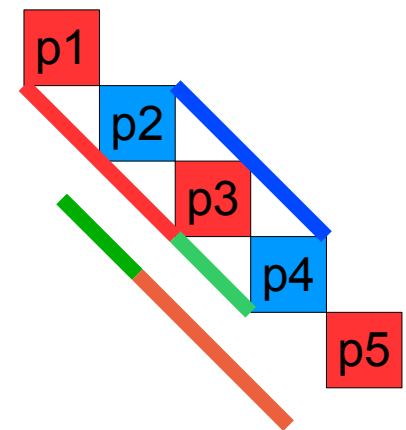
- Es werden Gradienten an ursprünglich roten Pixeln in diagonaler Richtung gebildet (für blaue Pixel analog):

$$\Delta_{nw} = \Delta_b + \Delta_{r1} + \Delta_{r2} + \Delta_{g1} + \Delta_{g2}$$

Δ_{ne} analog (90° gedreht)

- Suche nun den kleineren Gradienten und führe hue_transit mit den entsprechenden Pixeln durch:

$$B(P3) = \text{hue_transit}(G(p2), G(p3), G(p4), \\ B(p2), G(p4))$$



Agenda

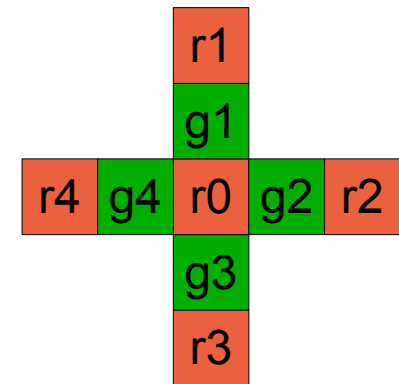
- Verbreitete Verfahren
 - Nearest Neighbour
 - Bilineare & bikubische Interpolation
 - Variable Number of Gradients (VNG)
 - Patterned Pixel Grouping (PPG)
- ➔ **Microsoft Algorithmus als weiteres Verfahren**
- Zusammenfassung

Microsoft Algorithmus

- Der Microsoft Algorithmus ist eine Erweiterung der bilinearen Interpolation um einen Korrekturwert
- Annahme:
 - die Farbwerte korrelieren miteinander
 - Das heißt, dass sich die Differenzen $|B-R|$, $|R-G|$ und $|B-G|$ in räumlicher Nähe nur allmählich ändern

Microsoft Algorithmus

- Um Grün an einem roten Punkt zu approximieren, wird der bilinear approximierten Wert von Grün mit einem Korrekturwert (Gradient) der umgebenen roten Pixel korrigiert:
- $g_0 = g_{Bi} + \alpha \cdot \Delta r$
 - $g_{Bi} = (g_1 + g_2 + g_3 + g_4) / 4$
 - $\Delta r = r_0 - (r_1 + r_2 + r_3 + r_4) / 4$
 - α gibt die Intensität der Korrektur an
- Anm.: Grün auf blauen Pixel analog

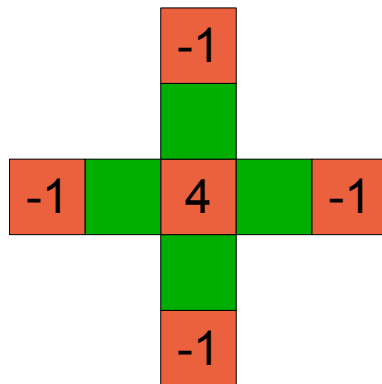


Microsoft Algorithmus

- Rot auf Grün (analog Blau auf Grün):
 - $r = r_{Bi} + \beta \cdot \Delta g$
- Rot auf Blau (analog Blau auf Rot):
 - $r = r_{Bi} + \gamma \cdot \Delta b$
- r_{Bi} bzw. b_{Bi} ist die entsprechende bilineare Interpolation
- Als Korrekturintensitäten α , β und γ werden die Werte $\alpha=1/2$, $\beta=5/8$ und $\gamma=3/4$ gewählt
- Für Δg und Δb gelten folgende Koeffizienten:

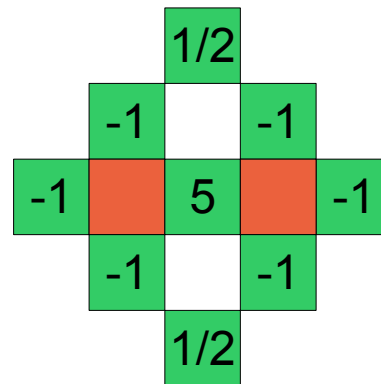
Microsoft Algorithmus

- Wichtung der umgebenden Pixel im Korrekturwert:



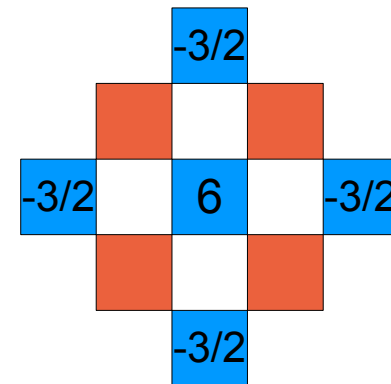
Grün auf Rot

Analog:
Grün auf Blau



Rot auf Grün
(evtl. um 90° gedreht)

Analog:
Blau auf Grün



Rot auf Blau

Analog:
Blau auf Rot

Agenda

- Verbreitete Verfahren
 - Nearest Neighbour
 - Bilineare & bikubische Interpolation
 - Variable Number of Gradients (VNG)
 - Patterned Pixel Grouping (PPG)
- Microsoft Algorithmus als weiteres Verfahren
- **Zusammenfassung**

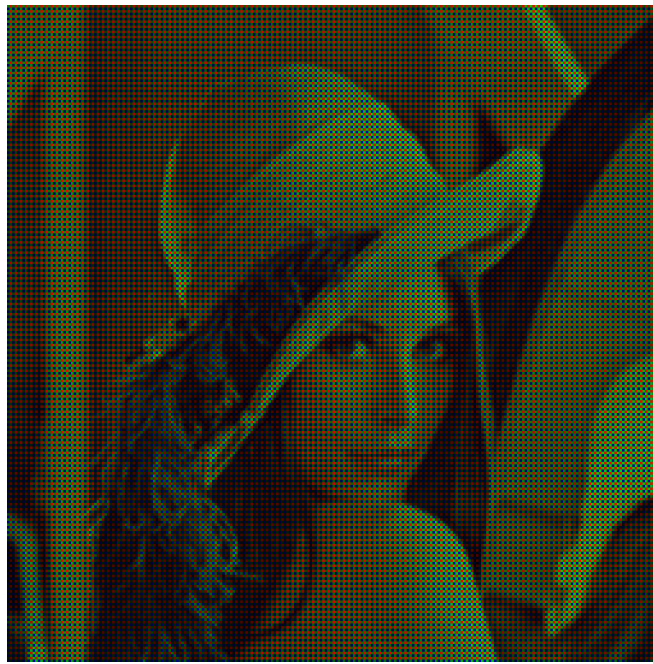
Zusammenfassung

	Pro	Contra	Anwendung
Pixel Binning	<ul style="list-style-type: none"> • Sehr schnell 	<ul style="list-style-type: none"> • Auflösung verringert • Helligkeit erhöht, Rauschen vermindert 	
Nearest Neighbour	<ul style="list-style-type: none"> • Sehr schnell 	<ul style="list-style-type: none"> • Auflösung verringert • Farbfehler 	Anfänge der Digicams
Bilinear (vgl. mit Nearest Neighbour)	<ul style="list-style-type: none"> • Weicheres Ergebnis • Weniger Artefakte 	<ul style="list-style-type: none"> • Immer noch Zipper-Artefakte 	Webcams
Bikubisch	<ul style="list-style-type: none"> • Guter Kompromiss aus Berechnungsaufwand und Qualität 		Adobe Photoshop, In-Camera Processing, Druckertreiber
Variable Number of Gradients (VNG)	<ul style="list-style-type: none"> • Super Qualität 	<ul style="list-style-type: none"> • Berechnungsintensiv 	
Patterned Pixel Grouping (PPG)	<ul style="list-style-type: none"> • Schneller als VNG • Weniger Farb-Artefakte als VNG 	<ul style="list-style-type: none"> • Im Durchschnitt schlechtere Quali als VNG 	

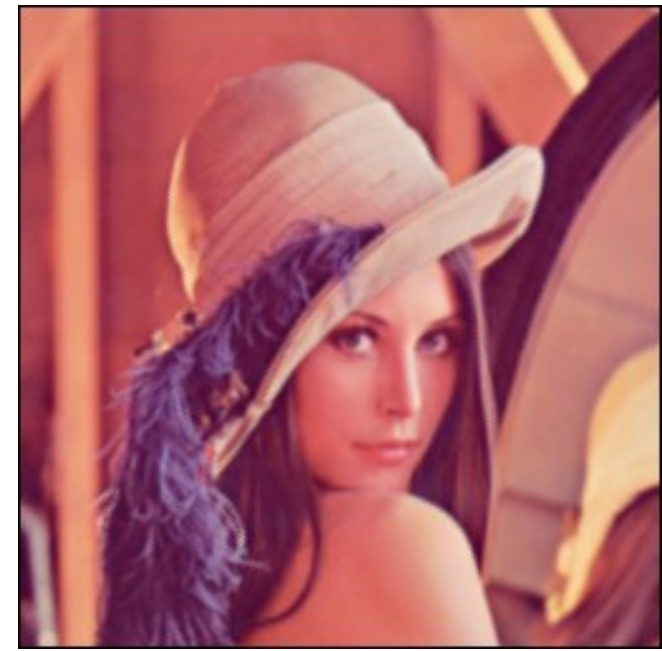
- Tradeoff: Geschwindigkeit ↔ Genauigkeit/Artefakte/Farbfehler



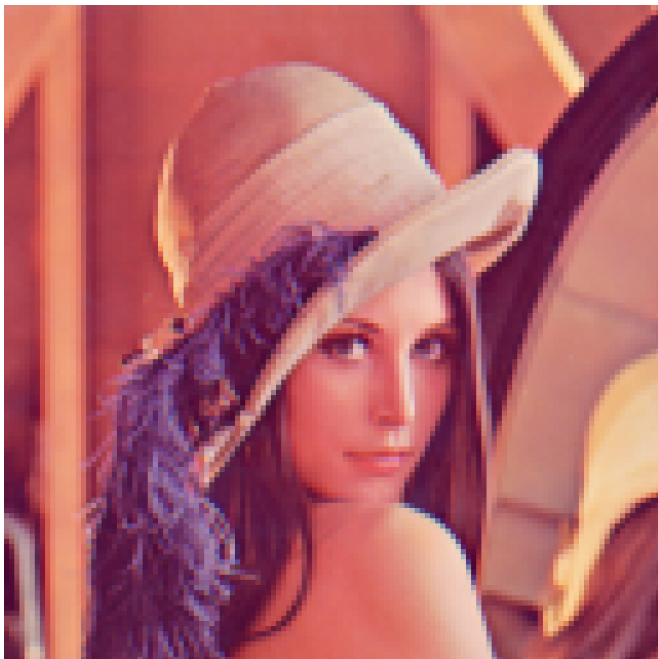
Original



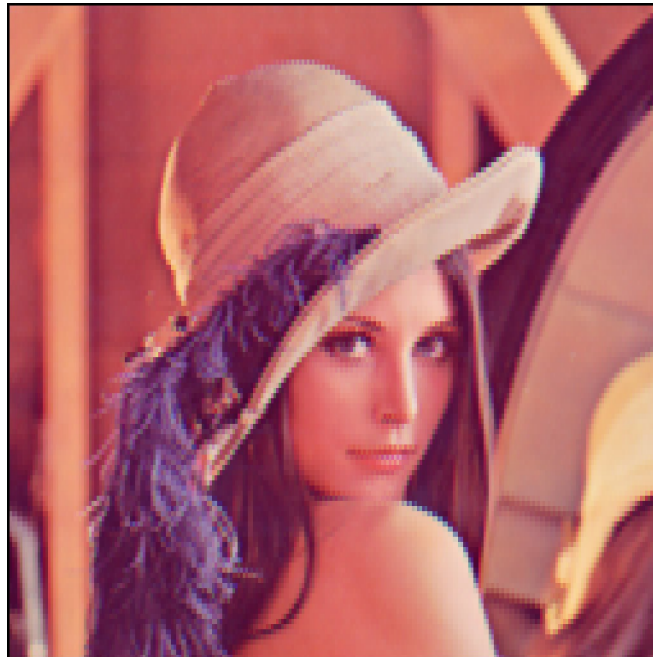
Bayer pattern



Bilinear



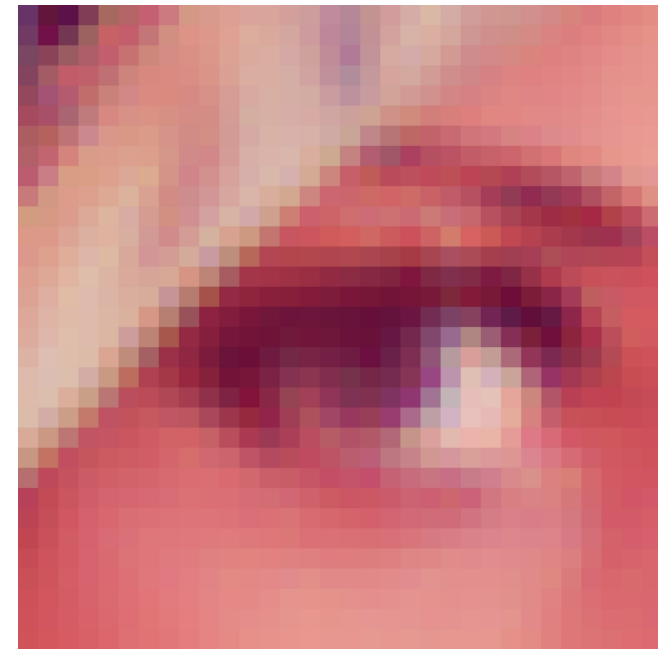
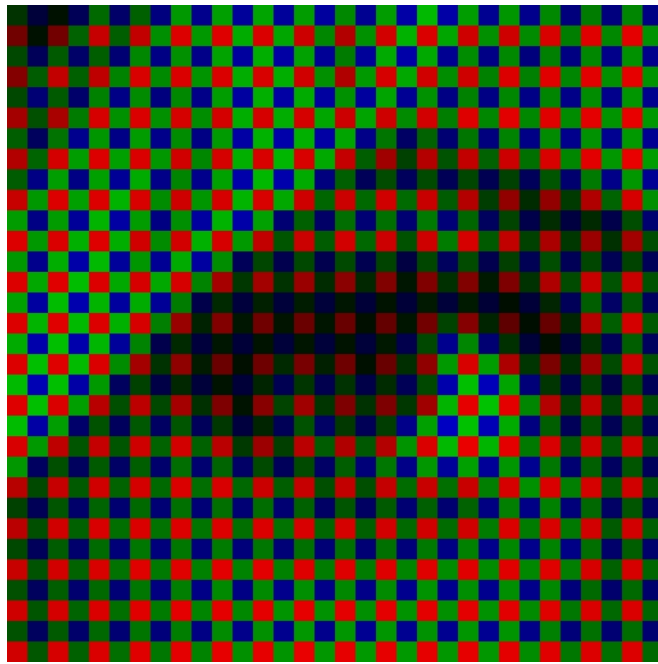
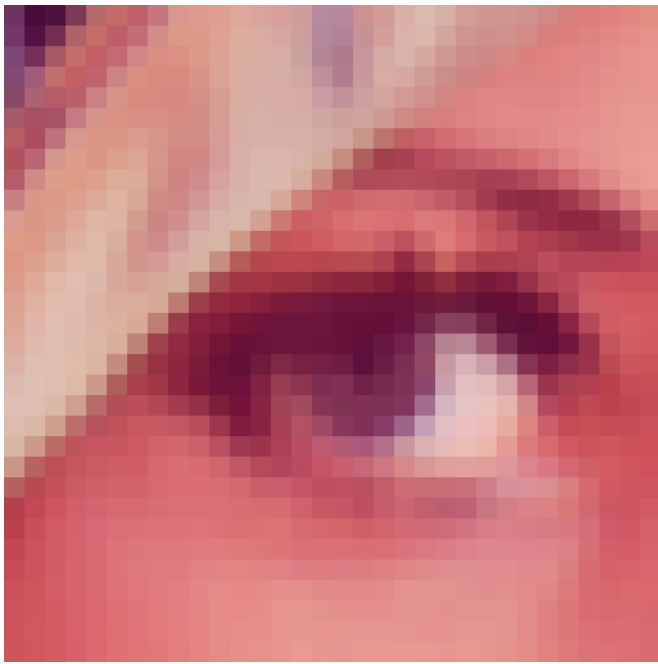
Pixel binning (vergrößert)



Nearest neighbor



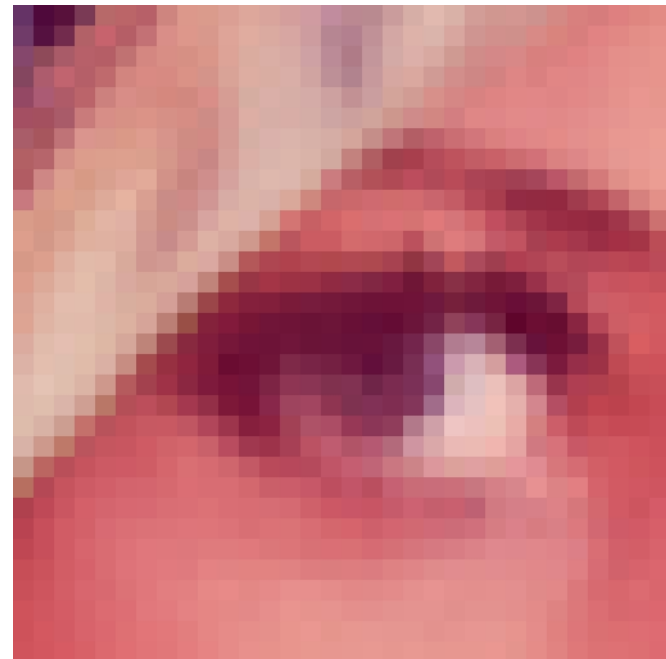
VNG



Original

Bayer pattern

Bilinear



Pixel binning (vergrößert)

Nearest neighbor

VNG

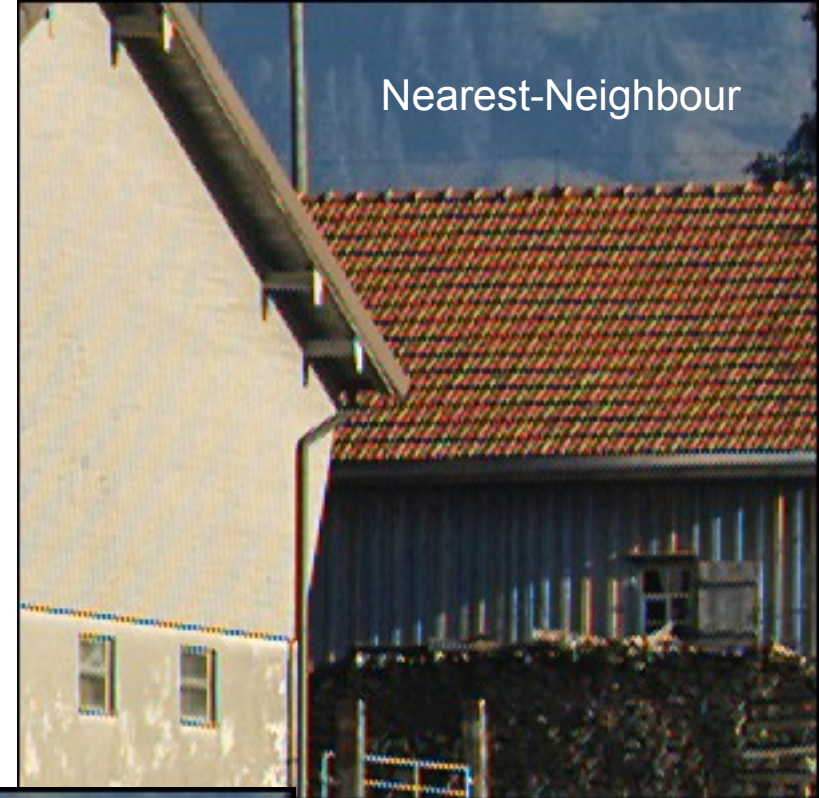
ORIGINAL



Pixelbinning



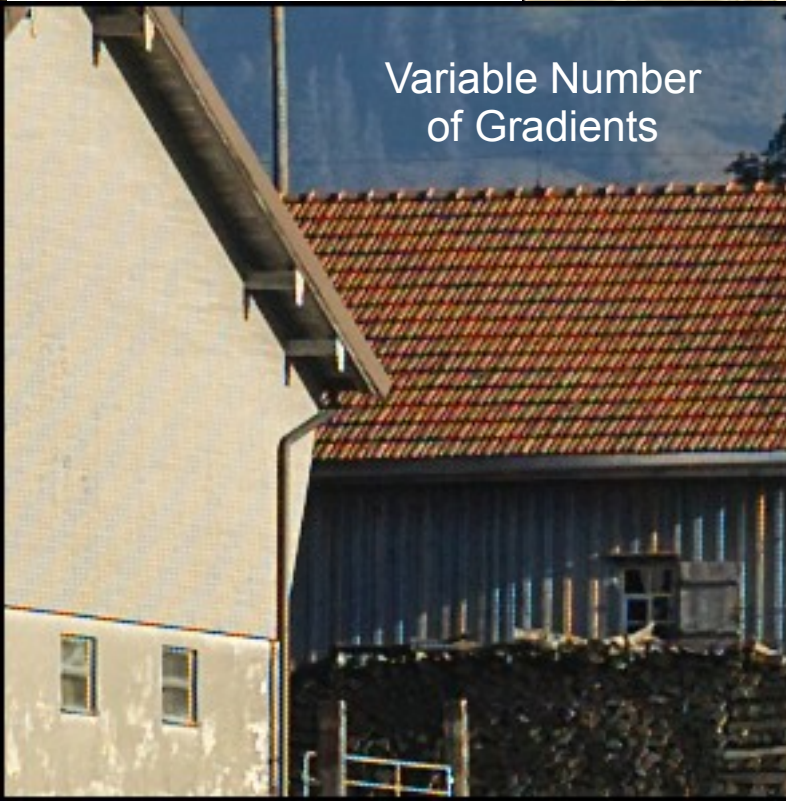
Nearest-Neighbour



Bilinear



Variable Number of Gradients



Ver-
gleich

Debayeringverfahren

Q&A / Diskussion / Bewertung