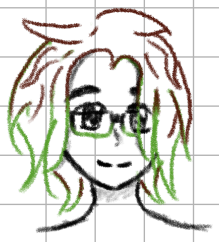
## Disclaimer

Das ist nur eine Zusammenfassung und kein Ersatz für das Skriptum / die VOs. Keine Garantie darauf, dass alles so stimmt, wie es hier steht. Das ist nur meine Interpretation der Inhalte. Falls etwas unklar sein sollte, bitte im Skriptum nachschauen. Einige Unterkapitel (wie z.B. "Andere Objektrepräsentationen") könnten fehlen, da ich sie als unwichtig/nicht prüfungsrelevant erachtet hab. Jegliche Bilder gehören den Urhebern des Skriptums.



Viel Spaß beim Lernen!

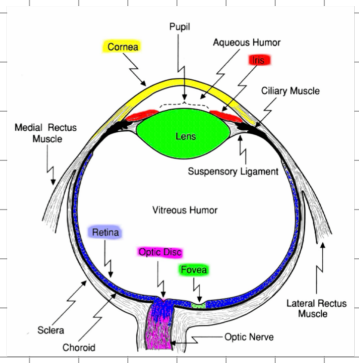
## Inhaltsverzeichnis



(nur 5/6 Kapitel, da ich die Einführung bzw. das 1. Kapitel ausgelassen hab)

# CV-Zusammenfassung

## Das menschliche Auge



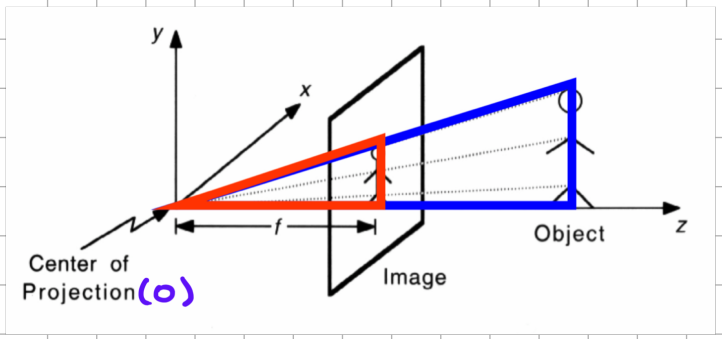

Vorgang des Sehens:

1. Linse fokussiert das Abbild
2. Bild trifft seitenverkehrt und auf dem Kopf an der lichtempfindlichen Membran (Retina) auf
3. Fotorezeptoren in der Retina werden durch das Licht angeregt
4. Als Reaktion darauf werden neuronale Impulse generiert, welche zum Gehirn weitergeleitet werden

2 Arten von Fotorezeptoren vorhanden: Stäbchen & Zapfen

Fovea centralis: Punkt des schärfsten Sehens

## Perspektivische Projektion




Lochkamera = einfachstes Aufnahmegerät, keine geometrischen Verzerrungen aber enthallten

Brennweite: Entfernung zwischen Bildebene & Zentrum der Projektion

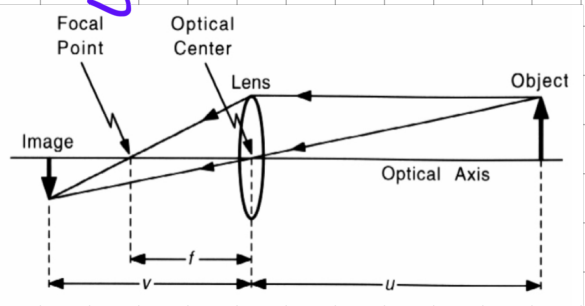
optische Achse: Gerade, die normal auf die Bildebene steht

Dreiecksbeziehungen: $\frac{y}{f} = \frac{f}{Z}$, $x = \frac{f}{Z}X$, $y = \frac{f}{Z}Y$, wobei $y$ = Projektion in der Bildebene, $Y$ = Objektgröße, $Z$ = Entfernung zu O

Projektion ≠ linear ≠ affin; Projektion = 1:n Abbildung

# Linsen

für scharfe Projektion mögl. kleines Loch notwendig, das wenig Licht durchlässt, lange Belichtungszeit notwendig




Modell einer "dünnen Linse":
Lochblende wird durch unendlich dünne, symmetrische Linse ersetzt

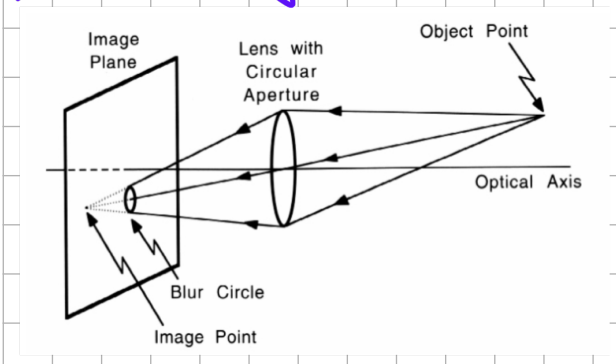
⇒ gleiche Abbildungsgeometrie wie bei Lochkamera (siehe Seite 2)

einfache Linsengleichung: $\frac{1}{u} + \frac{1}{v} = \frac{1}{f}$ (wenn $u = \infty$, dann $v = f$, sonst werden sie hinter dem Brennpunkt abgebildet, wie in der linken Graphik)

## Tiefenschärfebereich (DOF)

theoretisch: Schärfe hängt von der Entfernung des Objekts ab

praktisch: Objekte werden in bestimmtem DOF scharf abgebildet




DOF = Entfernung zwischen nähesten und weitesten Objekt der Szene, welches scharf dargestellt werden kann

⇒ wird von Sensorauflösung & Linsengröße beeinflusst:
größere Linse = mehr Strahlen gelangen auf die Bildebene → kleineres DOF
höhere Sensorauflösung = Blur Circles werden besser abgebildet

Blende = Öffnung vor der Linse, bestimmt Kegelwinkel der Strahlen

Blur Circle, falls Linse zu weit vorne oder hinten ⇒ unscharfes Bild

## Radiometrie

= Methoden zur Messung elektromagnetischer Strahlung

Strahldichte (= radience): Menge des reflektierten Lichts eines Punktes

Bestrahlungsstärke / Helligkeit (= irradience): projizierte Lichtmenge des Pkt.

spektrale Bestrahlungsstärke (= spectral irradience): Strahlungsaufkommen

## Bildauflösung

= Sammelbegriff für Detailreichtum eines Bildes

- radiometrische Auflösung:
  Anzahl der Graustufenwerte bzw. Bits (typisch: 8 Bit = $2^8$ Graustufen)
  Mensch kann ca. 120 unterscheiden, daher reichen $2^8 = 256$ Stufen
  weniger Gs. = weniger Speicherbedarf = geringere Qualität
- Sensorauflösung:
  Anzahl der Pixel pro Zeile & Spalte (z.B. 640x840)
  mögliche Einheiten: Megapixel, dots per inch (dpi), pixel per inch (ppi)
- räumliche Auflösung:
  Fähigkeit des Sensors, benachbarte Signale getrennt zu erfassen
  definiert durch minimale Entfernung zwischen Merkmalen bzw. der
  größe eines Merkmals; Einheit: L/mm (Linien pro Millimeter)
- zeitliche Auflösung:
  Anzahl der aufgenommenen Bilder pro Sekunde (fps)

Auflösungsgrenze = festgelegt durch Nyquist-Shannon-Theorem

## Sampling

= Umwandlungsprozess - Signal (kontin.) → numerische Folge (diskret)

Nyquist-Shannon-Theorem: begrenztes, analoges Signal kann perfekt von unendlichen Samplefolge rekonstruiert werden, wenn Samplingrate > 2B Samples pro Sekunde, mit B = höchstvorkommende Frequenz

## Bildsensoren

= Bauelement, was optisches Bild in ein elektronisches Signal umwandelt

zwei Arten von Pixelsensoren:

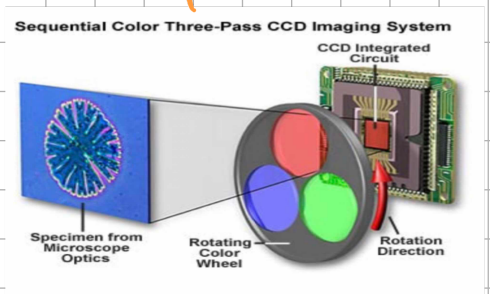
- digitale Charge-Coupled Device (CCD):
  bestehen aus Matrix lichtempfindlicher Photodioden, MOS-Kondensatoren, Feldeffekttransistoren, Steuerleitungen, Leitungspfaden
  bei gegebener Belichtungszeit (=exposure time) passiert folgendes:
  Licht trifft auf CCD-Sensor → elektrische Ladung baut sich auf → Kondensator speichert sie → Spannung wird von dort Zeile für Zeile ausgelesen → Information wird in digitale Daten konvertiert
  ⇒ arbeitet wie analge Kameras mit Flächenintegral ⇒ ist linear
  rauscharm, lichtempfindlich, hohe Uniformität
- Complementary Metal-Oxide-Semiconductor (CMOS)
  besitzt aktive Pixelsensoren, Pixel individuell auslesbar
  nicht-linearer Output, da getrennte Betrachtung der Nachbarn
  Funktionsweise entspricht menschlicher Bildaufnahme im Auge

geringerer Energieverbrauch, schnellere Lesegeschwindigkeit, Herstellung billiger

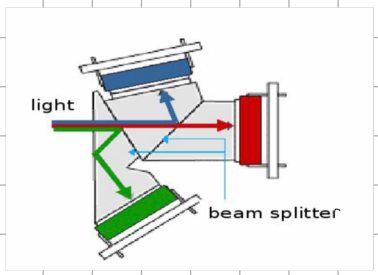
## Farbe auf CCD Sensoren

Bildsensoren können nur Helligkeit messen, daher Farbfilter nötig drei Arten Farbe zu erfassen (& weitere 2 darunter):

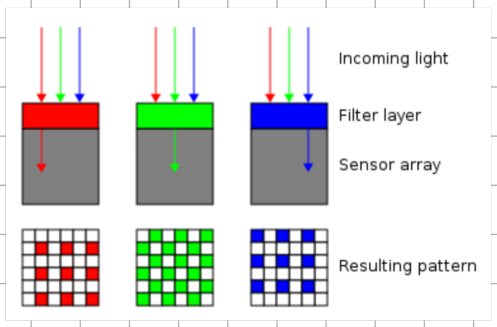
- Field Sequential Technik = drei farblose Bilder werden mit R, G & B-Filtern nach einander aufgenommen & kombiniert
  => resultiert oft in Geisterbildern

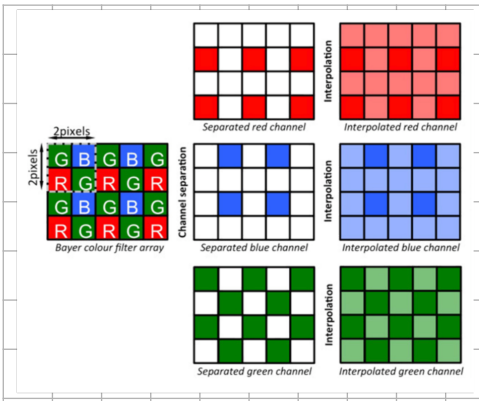



- 3 Chip-CCD Kameras = jeder der drei CCDs nimmt einen Farbkanal auf, Positionierung schwierig, da der Strahl dieselbe Stelle in allen CCDs treffen muss



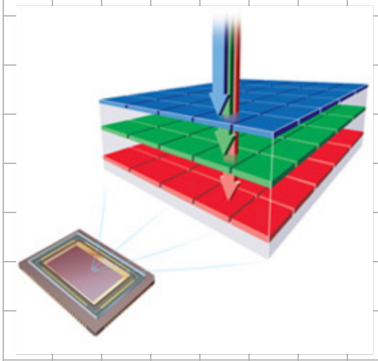

- Color Filter Array (CFA) / Color Filter Mosaic (CFM):
  Mosaik aus Farbfiltern wird über jedes Pixel des Bildsensors plaziert
  Bayer Filter: 25% R, 50% G, 25% B
  => ans menschliche Auge angepasst,
  roher Output = Bayer-Muster Bild (= bayer-pattern image)
  demosaicing = tatsächliche Pixelfarbe durch Interpolationsalgorithmen anhand benachbarter Pixel berechnen






einfache Interpolationsverfahren:
berechnen Farbwert aus gleichfarbigen Nachbarspixel (siehe Abb. links)
⇒ Vorgehen normal auf Kanten problematisch

andere Verfahren: a) versuchen Interpolation entlang der Kanten



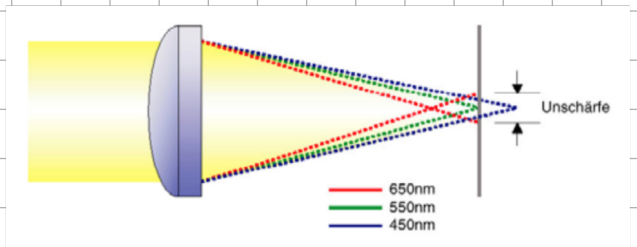
b) nehmen an, dass Flächenfarben bei unterschiedlichen Lichtverhältnissen gleich sind und Kanäle korrelieren
⇒ Versuch, Farbverhältnisse (z.B. rot-grün) beizubehalten

- Foveon X3 Sensor (= Kombination aus 3CCD & CFA)
drei vertikal gestapelte Photodioden (siehe Abb. links oben), die jeweils auf verschiedene Wellenlängen reagieren
⇒ resultiert in RGB-Daten & einwandfreie Bilddarstellung
- Multi-Shot Technologie: verwendet bewegbare Sensoren, die mit Präzisions-Piezomotor in Pixelschritten bewegt werden
⇒ mehr Farbinformation als bei Einzelaufnahme

## Ideale vs. Reale Kamera

⇒ es kommt zu Störungen, wegen nicht-idealen Linsen & Sensoren

- Optische Aberration:
Licht fällt nach der Transmission nicht in einen Punkt zusammen



- Chromatische Aberration:

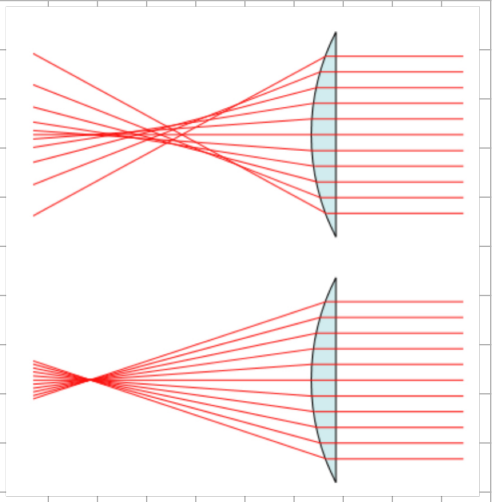
  Linse ist nicht imstande alle Farben auf einen Punkt zu fokussieren

  Ursache: die Linse bricht unterschiedliche Wellenlängen anders

- Sphärische Aberration:

  Strahlen weiter am Rand werden in einem Punkt fokussiert, als mittlere Strahlen

  Ursache: Kugeloberfläche der Linse



- Linsen Verzerrung:

  gerade Linien bleiben nicht gerade, sie werden nach innen oder außen verzerrt

  Ursache: Kugeloberfläche der Linse

## Weißabgleich

= gibt vor, welche Farbe weiß ist; passt Kamera an Ort an

Einheit der Farbtemperatur: Kelvin (z.B. Glühbirne = 3200 Kelvin)

⇒ kann vor oder nach der Aufnahme durchgeführt werden & erfolgt manuell oder automatisch

## Digitales Bild - Dateiformat

kontinuierliche/analoge Daten = Zahl kann unendlich viele mögliche Werte annehmen, ⇒ umwandlung von diesen = Digitalisierung

Rasterbild = Raster von diskreten Werten

Vektorbild = Bild in Form von geometrischen Objekten gespeichert

benötigter Speicher in Vektorformat: $L \times N \times B \times c$ Bit
($L$ = Zeilen, $N$ = Spalten, $2^B$ Graustufen, $B$ = Bitanzahl, $c$ = Farbkanäle)

## Raster-Bildformate

RAW-Format: speichert Farbwert ohne Nachbearbeitung
muss erst interpoliert werden

andere Formate: Bitmap (BMP), Portable Network Graphics (PNG)

Aufbau: Strukturen fixer & variabler Größe, in vordefinierter Sequenz
↳ z.B. Header mit Metadaten (z.B. Aufnahmedatum)

## Vektor-Bildformate

enthalten Geometrische Beschreibung, müssen ev. rasterisiert werden

Computer Graphics Metafile (CGM): graphische Elemente werden in Quelltextdatei gespeichert, später zu Binärdatei kompiliert

Windows Metafile (WMF): beinhalten Bitmap & Vektorg.-Teile

## Bildkompression

= Reduzierung irrelevanter/redundanter Bildinformationen

Verlustfrei: bevorzugt, wenn Kompressionsartefakte unerwünscht

Verlustbehaftet: geeignet, wenn Genauigkeitsverlust akzeptabel ist

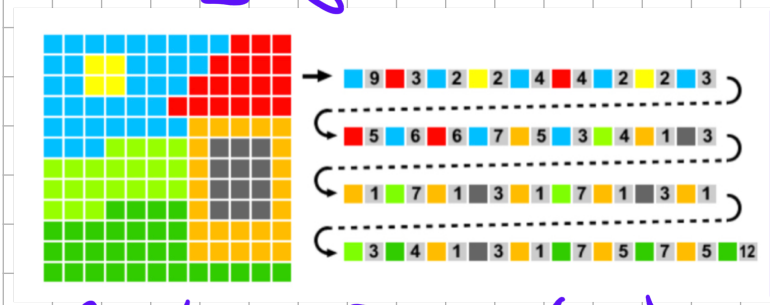
"visually lossless": verlustbehaftet + unbemerkbare Änderungen

## Verlustfreie Kompression

⇒ Komprimierung, sodass exakte Rekonstruierung möglich:

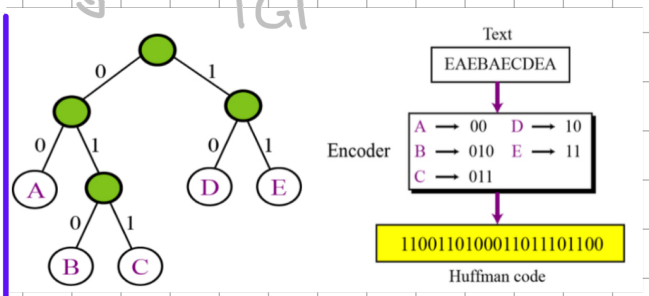
1. Statistisches Modell der Daten erstellen
2. Abbildung auf Bitfolge, sodass "wahrscheinlich" = kürzerer Output

Codierungsalgorithmen:

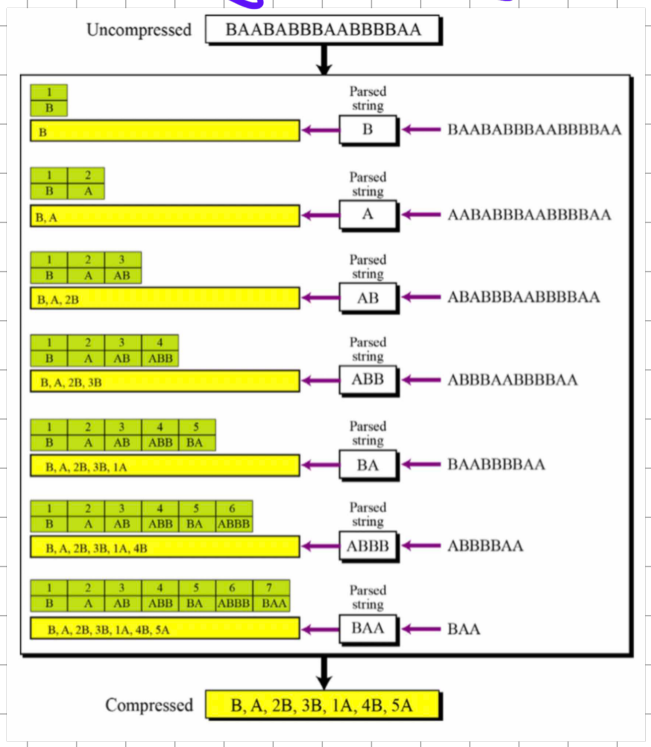



Run Length Encoding (RLE)

Überschneidung mit TGI




Huffman-Codierung




Lempel Zif (LZ):

Eingabebytes werden gesammelt, bis Zeichen ohne Eintrag kommt

Sequenz wird ohne Zeichen ausgegeben & Code (mit Zeichen) zur Tabelle hinzugefügt

## Verlustfreie Bildformate

Graphics Interchange Format (GIF): große Unterstützung

Portable Network Graphics (PNG): nur RGB, PKZIP-Kompression

Tagged Image File Format (TIFF): kann mehrere Bilder mit versch. Eigenschaften enthalten

# Verlustbehaftete Kompression

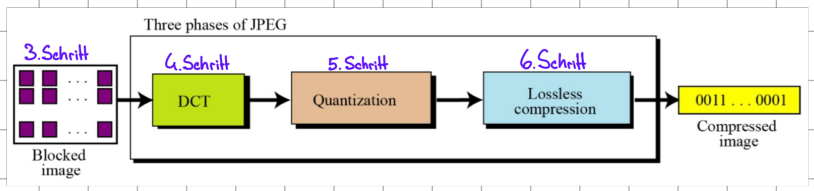
⇒ Entfernung bestimmter Datenteile, meistens jene, die schwach/nicht wahrgenommen werden können

# JPEG

⇒ Kompression von kontinuierlichen Bildern

Ziel: durchschnittliche Datenreduktion um Faktor 1:16

Codierungsprozess:




1. Umwandlung RGB-> YCbCR (Y = Helligkeit, Cb = Blue-Yellow, Cr = Red-Green)
2. Halbierung der räumlichen Farbauflösung (Cb & Cr)
3. Aufteilung des Bildes in 8x8 Blöcke
4. Berechnung des Frequenzspektrum jedes Blockes mittels diskreten Kosinustransformationen
5. Quantisierung der 64 Spektralkoeffizienten, anhand Quantisierungstabelle, welche Qualität & Schärfe bestimmt
   a. Division der DCT-Koeffizienten durch Quantisierungswerte
   b. Rundung auf nähesten Integer-Wert
6. Verlustfreie Kompression der Spektralkomponenten

⇒ visuell verlustfreie Kompression bis $\frac{1}{5}$, ab da verlustbehaftet

Schwachpunkt: Verhalten bei abrupten Übergängen

## Diskrete Cosinus Transformation (DCT)

Variation der Fouriertransformation, stellt Signale als Überlagerungen von Cosinuswellen versch. Frequenz & Amplitude dar

$$F(u,v) = \sum_{x=0}^{7}\sum_{y=0}^{7} \alpha(u)\alpha(v) \cdot f(x,y) \cdot \cos\left[\frac{\pi}{8}\left(x+\frac{1}{2}\right)u\right] \cdot \cos\left[\frac{\pi}{8}\left(y+\frac{1}{2}\right)v\right]$$

$u$ = horizontale Ortsfrequenz $(0 \leq u < 8)$, $v$ = vertikale Ortsfrequenz $(0 \leq u < 8)$, $\alpha(u)$ = normalisierter Skalierungsfaktor für Orthonormalität (= ortogonalität mit Norm/Vektorlänge = 1),
$f(x,y)$ = Pixelwert von $(x,y)$, $F(u,v)$ = DCT-Wert auf $(u,v)$

## Video Kompression

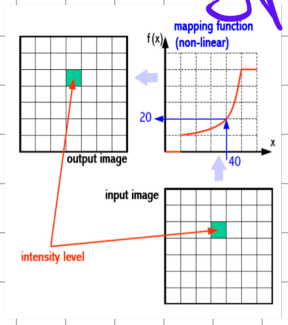
Kompressionsalgorithmen + Codecs für Bild- & Bewegungskompression
⇒ meistens verlustbehaftete Kompression, da Unterschied kaum merkbar
Makroblock = quadratischer Bildausschnitt
Größerer Unterschied zwischen Frames = mehr Codierung nötig
Interframe-Kompression: betrachtet 1-2 Frames vor & nach aktuellen Frame
Intraframe-Kompression: betrachtet nur aktuellen Frame (Bildkompression)
MPEG (Moving Picture Experts Group): arbeitet symmetrisch, da Codierung komplexer als Decodierung

## Punktoperationen (hier kurz P-op.)

= Operationen, die nur einzelne Pixel betreffen (neuer Pixelwert hängt nur vom Pixel $I(u,v)$ an derselben Stelle ab)

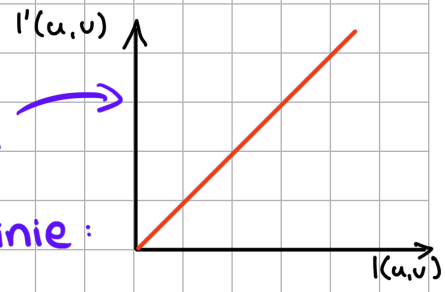
d.h. es gilt $I'(u,v) \leftarrow f(I(u,v))$ bzw. $a' \leftarrow f(a)$, wobei $f$ eine beliebige Funktion ist & $I(u,v)$ der jeweilige Pixel

- homogene P-op.: $f()$ ist unabhängig von den Bildkoordinaten und für alle Pixel gleich
- inhomogene P-op.: berücksichtigt Bildkoordinaten: $I'(u,v) \leftarrow g(I(u,v),u,v)$
- affine P-op.: Lineare Unterklasse der homogenen P-op.: $I'(u,v) \leftarrow a \cdot I(u,v) + b$, mit $a,b \in \mathbb{R}$; führt ggf. zu: Helligkeits- ($a=1, b \neq 0$), Konstrasteränderungen ($a \neq 1$)
- Abbildungsfunktion (mapping function)/Kennlinie:
  - kann (stufenweise) linear oder nichtlinear sein
  - neuer Helligkeitswert wird entsprechend dieser Funktion zugewiesen




## Identitätsfunktion & Invertierung

Identitätsfunktion: einfachste Funktion, Kennlinie:




Invertierung: $I'(u,v) \leftarrow -I(u,v) + q = q - I(u,v)$

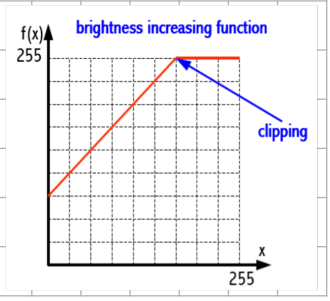
## Schwellwertoperationen (thresholding)

$p_{th}$ = Schwellwert: $I'(u,v) \leftarrow f_{th}(I(u,v)) = \begin{cases} p_0 \text{ für } I(u,v) < p_{th} \\ p_1 \text{ für } I(u,v) \geq p_{th} \end{cases}$

## Kontrast & Helligkeit

Kontrasterhöhung um 50%: $I'(u,v) \leftarrow I(u,v) \cdot 1{,}5$

Clipping: Ergebnisse außerhalb des Wertebereichs



auf Mini-/Maximalwert setzen (bei allen affinen P-op. nötig)

## Histogramm (hier kurz Hg. oder Histg.)

= Bildstatistiken, geben Infos über Eigenschaften von Bildern, wie z.B. Belichtungsfehler & Verarbeitungsschritte, indem sie zeigen, wie oft jeweilige Grau-/Farbwerte vorkommen

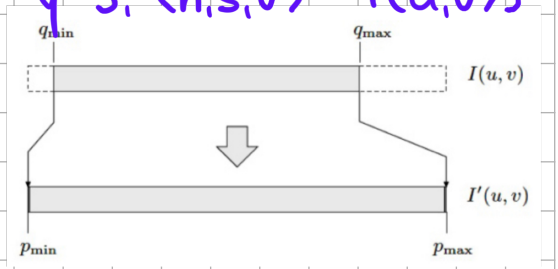
bei Graustufenbildern I gilt: $H(x) = \text{card}\{(u,v) \mid I(u,v) = x\}$, für $\{0 \leq x \leq q\}$

(card = Anzahl der Elemente einer Menge, x = Grauwert)

Auswirkungen von P-op:
- Helligkeiterhöhung ⇒ Verschiebung nach rechts
- Kontrasterhöhung ⇒ breiteres Histogramm
- Invertierung ⇒ Spiegelung

Zusammenfallen von Histogrammlinien ⇒ Verlust von Bildinformation

Farbhistogramm bei RGB oder HSV zu umfangreich, daher nur 2D, wie z.B. HS: $H(x,y) = \text{card}\{(u,v) \mid x = h \wedge y = s, (h,s,v) \leftarrow I(u,v)\}$




## Histogrammnormalisierung

= Abbildung aller Grauwerte auf den Wertebereich $[p_{min}, p_{max}]$, effektive Methode zur Kontraständerung

Formel: $I'(u,v) = (p_{max} - p_{min}) \cdot \frac{I(u,v) - q_{min}}{q_{max} - q_{min}}$ (q = Intensität, mit $q_{min} \neq q_{max}$)

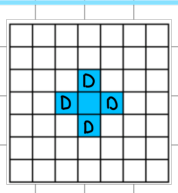
Nachteil: geringe Robustheit gegen Ausreißer, daher bildet man diese auf die Maximalwerte ab (Histg. dehnung)

akkumuliertes Histogramm: $H_a(x) = \sum_{k=0}^{x} H(k)$

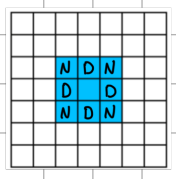
## Histogrammequalisierung / Histogrammausgleich

= Veränderung des Bildes durch P-op. bis Hg. gleichförmig verteilt
echte Gleichverteilung unmöglich, da Spitzen nicht entfernbar
Ziel: Kontrast für häufige Grauwertbereiche erhöhen (statt für gesamtes Bild wie bei Histogrammnormalisierung)
Formel: $H_n(x) = \frac{q}{H_a(q)} \cdot H_a(x)$ ($q$: kleine Zahl $q$) ≠ affine P-op.

## Nachbarschaften




Vierer-Nachbarschaft:
D-Nachbarn




Achter-Nachbarschaft:
D-& N-Nachbarn

Abstand der N-Nachbarn: $\sqrt{2}$ (Euklidische Metrik) / 2 (Manhattan-Metrik)

## Lokale Operationen / Filter

⇒ Ergebnis berechnet sich aus Ursprungspixel & Nachbarschaft
Filterform muss nicht quadratisch sein, Form beliebig

## Lineare Filter

⇒ verknüpfen Pixelwerte im Filter durch gewichtete Summation
lokale Mittelwertbildung: $I'(u,v) = \frac{1}{9} \sum_{j=-1}^{1} \sum_{i=-1}^{1} I(u+i, v+j)$
Filtermatrix / Filtermaske: Matrix von Filterkoeffizienten $F(i,j)$
allgemeine Formel: $I'(u,v) = \sum_{j=-1}^{1} \sum_{i=-1}^{1} I(u+i, v+j) \cdot F(i,j)$
Nachteil: Reduktion der Bildqualität, da auch beabsichtigte Strukturen verwischt werden

# Tiefpassfilter & Hochpassfilter

Tiefpass: filtert hohe Frequenzen heraus, z.B. für Rauschunterdrückung oder Glättungsoperatoren

- Mittelwertfilter/Boxfilter: einfachster Tiefpassfilter, da alle Gewichte gleich sind
- Gauß-Filter: $G_\sigma(x,y) = e^{\frac{x^2+y^2}{2\sigma^2}}$

  $\sigma$ = "Radius"/Glättungsstärke

  Mitte stärker gewichtet als die Ränder

  größerer Filter ≠ anderes Glättungsverhalten

Hochpass: filtert tiefe Frequenzen heraus, z.B. für Kantenerkennung

## Differenzfilter ↙ Hochpassfilter

⇒ interpretieren Filter mit negativen Koeffizienten als Differenz von zwei Summen (-a+b=a-b) innerhalb einer Filterregion: Summe der positiven Koeffizienten - Summe der negativen

Laplace-Filter: Differenz zwischen zentralen Wert (-4) und Summe aller D-Nachbarn (1+1+1+1=4)

$$F_{Laplace} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

⇒ beruhen auf zweite Ableitung der Bildfunktion

Gradientenfilter: beruhen auf erste Ableitung der Bildfunktion

Normierung auf [0...255]/Betrag nötig, damit alle Werte ≥ 0 sind

## Bildrandproblem

⇒ Filterkoeffizienten der Nachbarschaft liegen außerhalb des Bildes, daher folgende Randbehandlung möglich:

- statt Filterergebnisse Konstanten Wert einsetzen } verkleinern das Bild
- Bildwerte ungefiltert beibehalten
- Randbereiche berechnen und annehmen, dass: (verändern das Bild bei großen Filtern stark)
  - a) Pixel außerhalb einen konstanten Wert haben
  - b) sich die Randpixel fortsetzen } bevorzugte Varianten
  - c) sich das Bild in alle Richtungen wiederholt

## Formale Eigenschaften linearer Filter

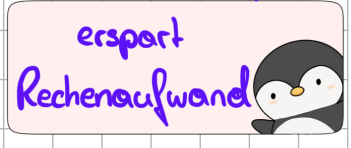
Lineare Faltung: $I'(u,v) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} I(u-i, v-j) \cdot F(i,j)$ bzw. $I' = I * F$

Lineare Korrelation: $I'(u,v) = \sum_{j=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} I(u+i, v+j) \cdot F(-i,-j)$ "convolution"

Eigenschaften: Kommutativität, d.h. $I * F = F * I$

Linearität, d.h. $(a \cdot I) * F = I * (a \cdot F) = a \cdot (I * F)$

daraus folgt:
- Assoziativität, d.h. $A * (B * C) = (A * B) * C$
- Seperierbarkeit, d.h. $F = F_1 * F_2 * F_3 * \ldots * F_n$
  - ↳ z.B. $F_{x,y} = F_x * F_y = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$, mit $F_x = [1\ 1\ 1]$, $F_y = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$



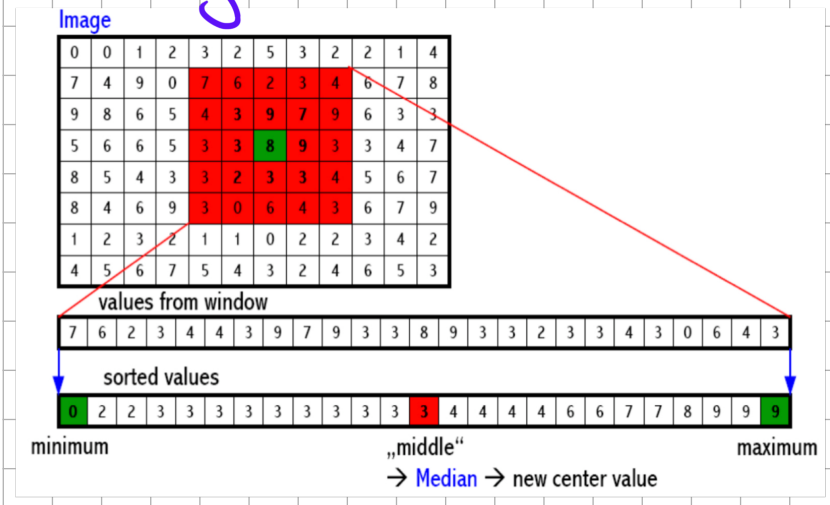

## Nicht-Lineare Filter

Rangordnungsfilter: Kombinieren benachbarte Pixel durch aufsteigende Sortierung & Selektierung des Minimums, Maximums oder Medians

Minimumfilter: $I'(u,v) = \min\{I(u+i, v+i) \mid (i,j) \in R\} = \min(R_{u,v})$

Maximumfilter: $I'(u,v) = \min\{I(u+i, v+i) \mid (i,j) \in R\} = \max(R_{u,v})$

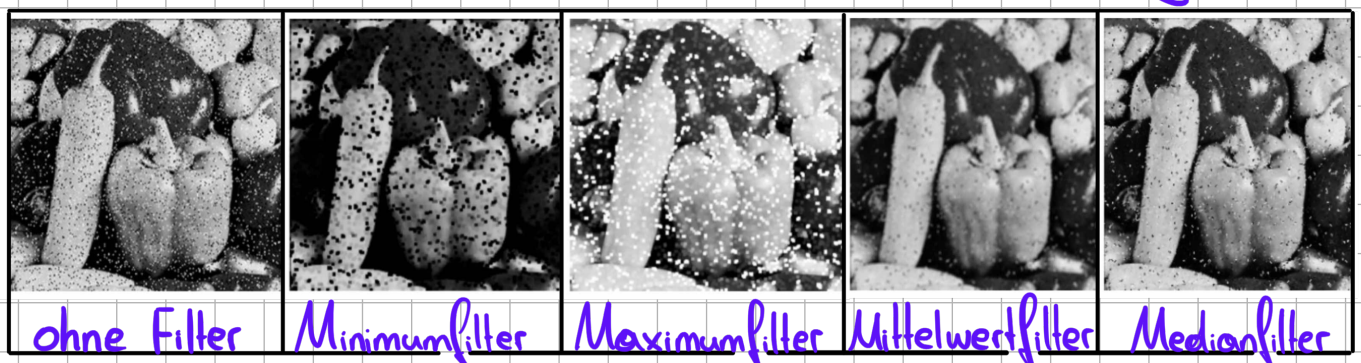
$R_{u,v}$ = Region der Bildwerte, über denen aktuell der Filter liegt




Medianfilter: ersetzt jeden Pixel mit dem Median seiner Filterregion R, wobei gilt:

$\text{median}(p_0, p_1, \dots p_k \dots p_{2u}) = p_k$

(siehe Abbildung links)




## Bildschärfung

⇒ erfolgt durch Anhebung der hochfrequenten Bildanteile

bei Schärfung von 2D-Bildern: Laplace-Operator $\Delta^2$ ($f''(x)$)

Laplace-Filter: $H^L = \Delta^2 f(x,y) = \frac{\partial^2 f}{\partial^2 x}(x,y) + \frac{\partial^2 f}{\partial^2 y}(x,y)$, wobei

$$\frac{\partial^2 f}{\partial^2 x}(x,y) = f(x+1,y) + f(x-1,y) - 2f(x,y)$$

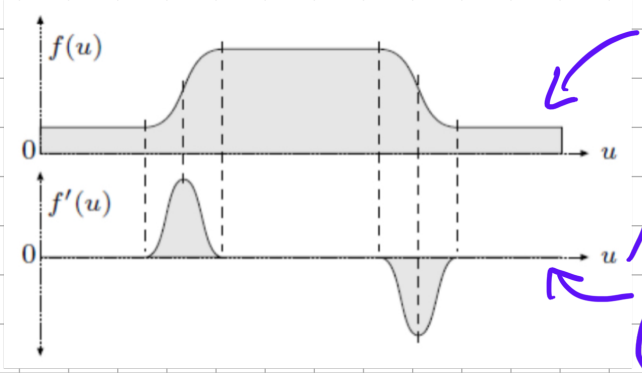
Anwendung des Filters: $I' = I - w \cdot (H^L * I)$, Schärfe von $H^L$ abhängig

## Kanten

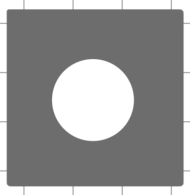
= Orte, wo sich die Intensität auf kleinem Raum stark ändert

⇒ tragen nicht-redundante Informationen, d.h. sind wichtig

## Gradienten-basierte Kantendetektion

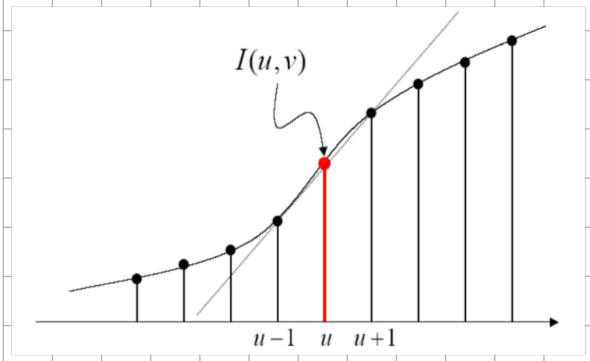



Graph von einem Bild, was ca so aussieht:



1. Ableitung davon (Ausschläge dort, wo Kanten sind)

diskrete Funktionen nicht differenzierbar ⇒ Annäherung notwendig




Berechnung durch Anstieg der Tangente an der Stelle u, wie Links gezeigt; gleicher Vorgang vertikal

partielle Ableitung = Ableitung einer mehrdimensionalen Funktion in einer Variable

Gradient (=Steigung): $\Delta I = \begin{pmatrix} \frac{\partial I}{\partial u} \\ \frac{\partial I}{\partial v} \end{pmatrix}, \frac{\partial I}{\partial u}(u,v), \frac{\partial I}{\partial v}(u,v)$

$|\Delta I| = \sqrt{\left(\frac{\partial I}{\partial u}\right)^2 + \left(\frac{\partial I}{\partial v}\right)^2}$ = rotationsinvariant = Orientierung des Bildes ist egal

Filterantwort ist richtungsabhängig

## Kantendetektionsfilter

isotroper Kantendetektor = richtungsunabhängig (z.B. Laplace-Operator)

Betrag/Kantenstärke & -richtung im Gradienten enthalten

Prewitt-Operator:

$$H_x^P = \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \quad H_y^P = \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix}$$

Sobel fast gleich - doppelt so hohe Glättung & stärker gewichtete mittlere Zeile

Gradientenwerte: $D_x(u,v) = H_x * I$ und $D_y(u,v) = H_y * I$

Kantenstärke/Betrag: $E(u,v)=\sqrt{D_x(u,v)^2+D_y(u,v)^2}$

Richtung (Prewitt): $\frac{\Delta f}{\Delta u}(u) \approx \frac{f(u+1)-f(u-1)}{2}$, (Sobel): $\phi(u,v)=\tan^{-1}\left(\frac{D_y(u,v)}{D_x(u,v)}\right)$

ältester Kantenoperator = Roberts-Operator; verwendet kleinsten Filter:

$$H_1^R=\begin{pmatrix}0&1\\-1&0\end{pmatrix} \quad H_2^R=\begin{pmatrix}1&0\\0&-1\end{pmatrix}$$

⇒ reagiert stark auf diagonale Kanten

generelle Regel: bessere Kantendetektion = größere Richtungsabhängigkeit

↳ Lösung: mehrere richtungsabhängige Filter, wie z.B. Kirsch-Op.:

$$H_1^K=\begin{pmatrix}-1&0&1\\-2&0&2\\-1&0&1\end{pmatrix} \quad H_2^K=\begin{pmatrix}-2&-1&0\\-1&0&1\\0&1&2\end{pmatrix} \quad H_3^K=\begin{pmatrix}-1&-2&-1\\0&0&0\\1&2&1\end{pmatrix} \quad H_4^K=\begin{pmatrix}0&-1&-2\\1&0&-1\\2&1&0\end{pmatrix} \quad H_5^K=\begin{pmatrix}1&0&-1\\2&0&-2\\1&0&-1\end{pmatrix} \quad H_6^K=\begin{pmatrix}2&1&0\\1&0&-1\\0&-1&-2\end{pmatrix} \quad H_7^K=\begin{pmatrix}1&2&1\\0&0&0\\-1&-2&-1\end{pmatrix} \quad H_8^K=\begin{pmatrix}0&1&2\\-1&0&1\\-2&-1&0\end{pmatrix}$$

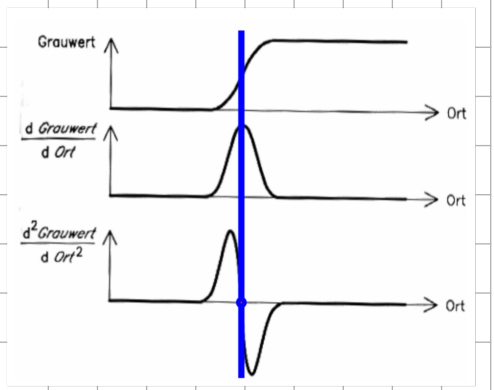
$E^K(u,v)=\max(D_1(u,v), D_2(u,v), \dots D_8(u,v))$, Vorteil: keine Wurzel

## Kantendetektion mit zweiten Ableitungen

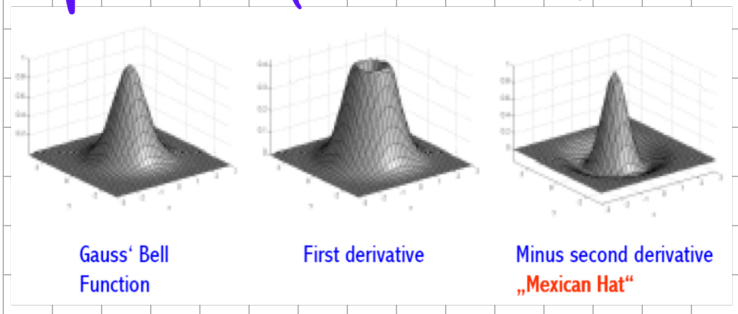
verwendet Nulldurchgänge der 2. Ableitung, um Kanten zu erkennen

bekantester Vertreter: Laplace-Operator

↳ richtungunabhängig, aber rauschanfällig, daher muss geglättet werden




Laplacian-of-Gaussian Operator (LoG): $f(x,y)=\frac{1}{2\pi\sigma^2}e^{\frac{x^2+y^2}{2\sigma^2}}$




kontinuierliche Repräsentation:

$$g(x,y)=\frac{\partial^2 f}{\partial x^2}+\frac{\partial^2 f}{\partial y^2}=$$

$$=-\frac{1}{\pi\sigma^4}e^{\frac{x^2+y^2}{2\sigma^2}}\left(1-\frac{x^2+y^2}{2\sigma^2}\right)$$

## Canny-Operator

"Multi-Scale": skaliert Bild, damit Filter globale Kanten erkennt

Canny: - verwendet Satz von großen, gerichteten Filtern
- schaut auf verschiedenen Auflösungsebenen
- fügt sie zu einem Kantenbild (edge map) zusammen

drei Ziele: a) Verringerung falscher Markierungen
b) möglichst gute Kantendetektion
c) eine Markierung pro Kante liefern

weitere Infos: - im Kern ein Gradientenverfahren
- stellt durch Maximums-Suche c) sicher
- verwendet meistens "Single Scale"
- stellt durch Filterradius b) sicher
- gegenüber anderen Kantendetektoren bevorzugt