

Was ist eigentlich Bildverarbeitung ? -3	→ Beschäftigt sich mit Verarbeitung, Speicherung und Darstellung von visuellen 2D oder 3D Informationen aus digitalisierten Daten.
Was nutzt man dafür?	→ Methoden der Signalverarbeitung . Man versteht also Bilder als ein Signal. Alte analoge Fernsehsignale nutzen Zeilenabtastung, wobei analoge, elektromagnetische Wellen genutzt wurden, um die Helligkeit des Bildes über die Zeit einzufangen. EKG-Signale sind aber auch visualisiert und man nutzt Methoden der Signalverarbeitung hier. Man denselben Methoden verarbeitet man Bilder.
Wie kann man digitale Bilder noch auffassen?	→ Digitale Bilddaten sind letztendlich nichts anderes als (bildliche) zwei- oder dreidimensionale Repräsentationen gemessener Signale
Was ist der Unterschied zwischen Bildbearbeitung und Bildverarbeitung?	Bildbearbeitung \neq Bildverarbeitung → Bildbearbeitung legt eine abstrakte Sicht auf Änderungen an Bildern . Das ist im Grund die klassische Bearbeitung von Bildern, die man nutzt, wenn man mit Filtern Bilder auf Instagram verbessert. Man nutzt keine Mathematik oder Algorithmen, sondern einfach seine subjektive Meinung und Sicht. Ich entscheide also selbst, was ich mit Bild mache.
Welche Info kann man aus Bildbearbeitung zum Beispiel isolieren? → 3	→ Bildverarbeitung legt aber die mathematischen/algorithmischen Grundlagen hierfür, die von wiederum (z.B. von Grafikprogrammen) zur Bildbearbeitung genutzt werden. Es geht hier um gleichartige Verarbeitung. Systematische Verarbeitung, bei der man selbst nicht mehr eingreift.
Was sind verwandte Felder der Bildbearbeitung? → 2	→ Programme können aber auch bei beiden benutzt werden. Filter sind nicht exklusiv für eins von beiden. Informationen aus Bildbearbeitung? → Bewegungsbestimmung : Bewegung in Bilderabfolgen erkennen. • Bildsegmentierung : Wie kriege ich bestimmte Strukturen aus dem Bild raus. Wie trenne ich es vom Rest des Bildes. • Bilderkennung und Mustererkennung : Bestimmte Teile des Bildes erkennen und beschreiben können.
	Verwandt: • Maschinelles Sehen : Erzeugung von Bildbeschreibungen aus Bilddaten. Wie kann ich aus einem Bild eine Beschreibung des Bildes erschaffen, die mir die wesentlichen Informationen, die zur Synthese des Bildes oder Rückrekonstruktion nötig sind, notwendig sind. Bsp. Wie groß ist der Tumor im Bild? Ergebnis 3 mm Durchmesser. KI-Nutzung. • Computergrafik : Erzeugung von Bilddaten aus Bildbeschreibungen. Das Gegenteil von Maschinell Sehen. Habe eine Bildbeschreibung und versuche ein Bild daraus zu generieren. AI Image Generierung. Beschreibung kann auch mathematische Formeln sein.
Was ist der Ortsraum und wofür ist der Ortsraum gut?	→ Entspricht der üblichen Darstellungsform eines Bildes → Im Ortsraum wird ein Bild in eine Matrix mit aufgeteilt. → Jedes Bild wird also in eine Matrix bestehend aus Reihen und Spalten aufgeteilt. Bekannt sind die Elemente dieser Matrizen als Pixel des Bildes (nicht des Bildschirms).
Wie erfolgt die Filterung im Ortsraum?	→ Man kann sich Bilder also wie eine Excel-Tabelle vorstellen mit Spalten, Reihen und Werte-Elementen. → Jeder Bildpunkt/ Element der Matrix hat einen ihm zugeordneten Grau- oder Farbwert. → Jeder Pixel ist also Einfarbig, weil er nur 1 Wert enthält. → Filterung im Ortsraum = direkte Manipulation der Werte . Manipulation erfolgt über Rechenvorschriften/Operatoren statt.
Wofür wird Bildverarbeitung in der	Methoden / Konzepte der BV werden angewendet, um eine strukturelle und funktionelle Transparenz des menschlichen Körpers für Prophylaxe, Diagnose und

<p>Medizin und Wissenschaft verwendet?</p>	<p>Therapie zu schaffen.</p> <ul style="list-style-type: none"> → Etwas über den Patienten durch Bildverarbeitung lernen. → Gemessene Daten anschauen nach einer Messung. → Aus gemessenen Daten sinnvolle Bilddaten erschaffen, die beschrieben und interpretiert werden können vom Menschen. → Bildbeschreibung kann wiederum zur Generierung neuer Bilder genutzt werden oder der Bildverarbeitung helfen. → Oft passiert Bildverarbeitung damit Bilder überhaupt generiert werden können. CT-Bilder setzen zum Beispiel Bildverarbeitung voraus, damit man überhaupt aus den gemessenen Daten Bilder generieren kann.
<p>Näherung an die BV: Mensch vs. Maschine</p> <p>Wieso muss man verstehen, wie ein Mensch Bilder sieht und welchen Bezug hat das zur Maschine?</p> <p>Wie unterscheiden Menschen Objektgrenzen? - 5</p> <p>Was besagt die Semiotische Triade? → Wie unterscheidet sich hier der Computer vom Menschen?</p>	<p>Damit Bildverarbeitung digital funktioniert, ist es hilfreich zu verstehen, wie menschliche Betrachter sich Bildern.</p> <ul style="list-style-type: none"> → Bei der computer-bezogenen Betrachtung von Bildern geht man so ähnlich wie beim Menschen vor und seiner Betrachtung von Bildern. → Man fragt sich „wie schaut ein Mensch auf ein Bild“ und fragt sich, wie man diese Ansichten maschinell umsetzen kann und nachbilden kann. <ol style="list-style-type: none"> 1. Menschen stellen Objektgrenzen fest. Wir erkennen also Objekte und unterscheiden Objekte anhand von Grenzen, die wir erkennen. → Wir erkennen Sie Anhand von: <ol style="list-style-type: none"> 1.1. Farb- bzw. Helligkeitsunterschieden 1.2 Strukturänderungen – Glatte vs geriffelte Flächen 1.3 Form/Farbe/Textur haben, die in etwa einer gedanklichen Vorstellung entspricht. 1.4 Erkannte Objekte werden zum Beispiel bzgl. ihrer Lage, Größe, Struktur, Formgebung, ... beurteilt sowie je nach Fragestellung zueinander in Relation gesetzt. Wo liegen Objekte relativ zu mir und anderen Objekten? 1.5 Die Fragestellung → Suche ich nach einem bestimmten Element im Bild oder betrachte ich das Bild als Ganzes. Die Perspektive ändert die Wahrnehmung des Bildes <p><u>Semiotische Triade.</u></p> <div data-bbox="614 1294 1364 1841" data-label="Diagram"> </div> <p>Ein Mensch hat Bilder/Vorstellungen in seinem Kopf, wie ein Objekt aus der realen Welt auszusehen hat.</p> <p>Ein Objekt der realen Welt wäre ein Stuhl, der physisch in der Welt existiert, unabhängig vom Menschenverstand.</p> <p>Aber ein Mensch verbindet etwas mit dem Stuhl – er hat eine Vorstellung, wie ein Stuhl auszusehen hat und wozu es dient (4 Beine, etwas, auf dem man sitzen kann). Diese Vorstellung ist aber erlernt. Der Mensch abstrahiert, was ein Stuhl für ihn ist</p>

basierend auf dem, was er gelernt hat. Basierend auf unserer Abstraktion entscheiden wir, welche Objekte in der echten Welt für uns ein Stuhl ist und was nicht. Die Vorstellung/Begriff ist also etwas, was im Kopf passiert. Wir bilden Klassen von abstrahierten Objekten in unserem Verstand, um Gegenstände wieder zu erkennen. Bezeichner und Symbole sind Worte, die wir benutzen, um die Klassen wiederzuerkennen. Jedermann weiß, was gemeint ist, wenn jemand „Stuhl“ sagt. Jedermann hat aber individuelle Vorstellungen im Kopf. Man sollte Begriffe also genau definieren, damit jedermann versteht, was man unter dem Objekt zu verstehen hat → Dasselbe betrifft auch Bildanalysen: **Man sollte zum Beispiel bei Muster und Bilderkennung ganz genau definieren, wonach man sucht und genaue Vorgaben an den Computer machen.**

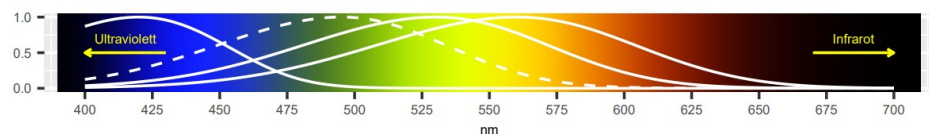
→ Der Computer macht solche Verbindungen nicht!

→ Auch beim Menschen kann es zu Missverständnissen kommen (Kulturelle Unterschiede, Sprachbarrieren), weshalb man Vorstellungen und Bezeichner für Menschen und Computer klar definieren sollte.

→ Der Computer wird auch „Vorstellungen“ und „Begriffe“ von Objekten haben (sonst könnte er diese nicht erkennen), aber seine Vorstellungen und Begriffe stimmen selten mit dem Menschlichen überein. Hier lernt es basierend auf einem Trainingsset.

TLDR → Computer und Mensch sehen und verstehen Objekte der realen Welt auf unterschiedliche Art und Weise, was zu verschiedenen Ergebnissen führen kann.

2. Farbwahrnehmung



Was passiert, wenn man Licht mit einem Prisma aufteilt?

→ Welche Farben stecken im weiß eines Bildschirms und welche Farben sind es?

→ Wieso sind es nur die 3 Farben und welchen Bezug hat das zum Menschen?

Wieso sieht der Mensch trotzdem mehr als 3 Farben?

→ Handelt es sich bei Mensch und Computer um additive oder subtrahierende Farbmischung?

→ Wie unterscheidet sich Mensch und Computer bei der Farbwahrnehmung?

→ Licht bricht in viele Farben.

→ 3 Farben → **Rot, Grün, Blau.** → RGB

→ Menschen haben Zapfen, um Farbe zu erkennen. Dabei hat der Mensch nur Zapfen für Rot, Grün und Blau. Die Absorptionsmaxima der 3 Zapfenarten liegen nämlich im grünen, Roten und blauen Bereich.

→ Die Zapfen im Auge nehmen nicht nur 1 Rot, Blau oder Grünton wahr, sondern Sie erkennen Frequenzen von Wellenlängen. Schaut man sich das Bild an, so sieht man, dass es keinen Bereich gibt, wo die 3 Zapfen die selbe Absorption aufweisen. Dadurch kann man viele Farben wahrnehmen.

→ Der Mensch kombiniert also die Farben **additiv**. Computer auch.

1. Mensch kann Farben, die zu ähnlich sind, schlecht unterscheiden. Bsp. Sehr ähnliche Grüntöne. Der Computer kann Sie sehr leicht unterscheiden.

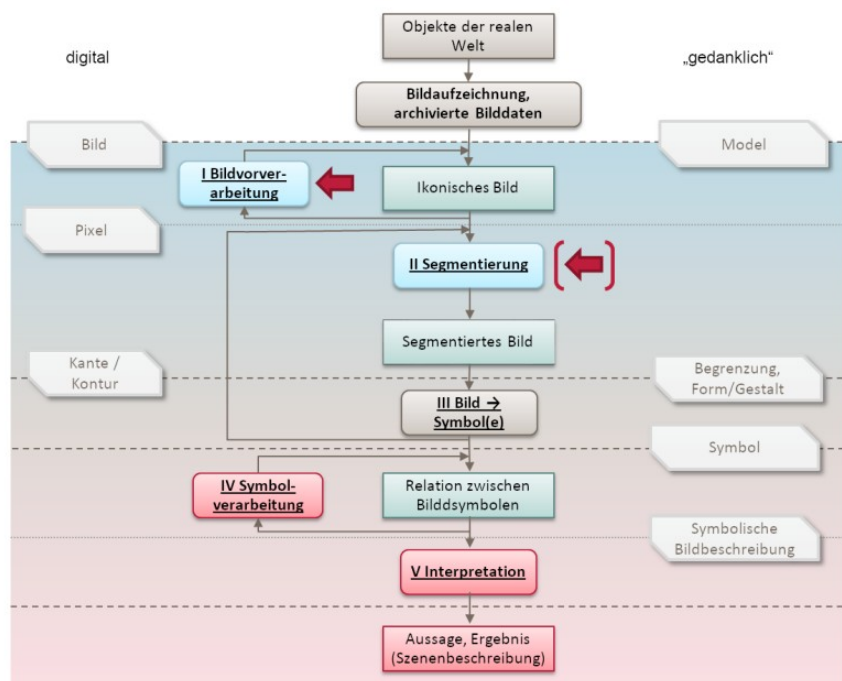
2. Der Mensch sieht nur zwischen 400-720 nm. Der Computer erkennt Sachen auch außerhalb dieses Wellenlängenbereiches (bsp Infrarot).

1. Der Mensch kann so oder so nicht mehr erkennen, da er auf dieselbe Art und Weise Farben erkennt.

2. Durch Additive Farbmischung kann man alle anderen Farben, die der Mensch sehen kann, auf dem Monitor abbilden.

Wieso nutzen viele Bildschirme RGB? -2	Blau ist sehr unempfindlich und man braucht viel Energie, damit man blau erkennt. Bei gleicher Helligkeit ist ein grüner Laser deutlich heller als rot.
Wie unterscheiden sich die 3 RGB Farben - vor allem Blau?	→ Subt. Farbkasten → Farben werden nicht weiß, wenn ich alle Farben kombiniere. Man kann auch nicht alle Farben generieren. Man kriegt brauntöne. Wenn man Farben sieht, so werden alle Wellenlängen absorbiert außer die Wellenlänge der Farbe, die man sieht. Bsp. Pflanzen absorbieren alles, außer grün → Pflanzen wirken grün. Bei der subt. Farbmischung ist es anders. Hier überdecken sich die Absorptionseigenschaften der gemischten Farben. Die Absorptionseigenschaften überlagern sich, sie subtrahieren von einander und nur noch wenige Wellenlängen (die, die nicht überlagert wurden) werden zurückgegeben.
Wie unterscheidet sich additive und subtrah. Farbmischung?	Hier subtrahiere ich Wellenlängen wie ein Filter. → Additive: kann alle 3 Farben beliebig kombinieren und so jede Farbe erschaffen. Hier addiere ich Wellenlängen zu neuen Farben.
Was sind weitere Unterschiede zwischen Mensch und Computer?	→ Menschen fallen auf optische Täuschungen rein, weil unser Gehirn Informationen ergänzt. Der Computer nicht. → Auf den Kontext kommt es (auch) an – menschliche Betrachter beziehen ihn (automatisch) mit ein, BV muss das passende Vorgehen erst beigebracht bekommen. → Das Erkennen und Auswerten von Mustern bzw. Objekten in (digitalisierten) Bildern ist (mit gewissen Ausnahmen) für den Menschen im Idealfall sehr einfach. • → Diesen Vorgang am Rechner “nachzubilden” kann je nach Fragestellung zeitraubend und komplex sein. → Vorsicht: Ergebnisse von Algorithmen können auch schon mal täuschen! → Menschen haben eine genaue Vorstellung von dem, was im Bild “gesucht” wird (hoffentlich...), die Bilder werden anhand dieser Vorstellung und im jeweiligen Kontext (Vorwissen) ausgewertet. → Soll BV sinnvoll eingesetzt werden, muss der Transfer in die digitale Welt gelingen: Menschliches Vorgehen bei der Betrachtung/Auswertung in digitale Prozesse übersetzen bzw. ergänzen → Essenziell: Vorab Fragestellung und Zielsetzung genau überlegen!
Welche Schlussfolgerungen ziehen wir daraus für die Bildbearbeitung und der Arbeit mit Computern?	

Beschreibe den Ablauf der Bildverarbeitung in der Maschine.



Erste Bildvorbereitung → Bevor man das erste richtige Bild hat, werden Signale vorverarbeitet, um ein Bild zu generieren. Das ist zum Beispiel bei CT-Bildern der Fall.

Segmentierung → Bild wird in Teile aufgeteilt und die Teile werden identifiziert, wodurch einzelne Bildelemente zu Symbolen werden, die beschrieben und identifiziert werden können. Um Symbole abzugrenzen werden Oberflächengrenzen (siehe oben mehr dazu) definiert und voneinander unterschieden.

Dann werden Symbole in Relation zueinander gestellt (liegt ein Gegenstand neben einem anderen, ist es vor dem Gegenstand usw.) und interpretiert. Symbolische Bildbeschreibung führt zur Beschreibung der Szene, die auch in maschinellen Sehen benutzt wird.

Was sind Schnittbildverfahren (bsp nennen)? -3

→ CT, MRT, B-Mode-Ultraschall

Mit welchem Verfahren kann ich Stoffwechsel abbilden? - 2

→ PET, Szintigraphie

Welche 3 Faktoren entscheiden über die Verarbeitung und Interpretation von medizinischen Bildgebungsdaten -3?

Was kann abgebildet werden? → Morphologie oder Stoffwechsel
Wie kommen die Bilder zustande? → Schnittbild oder Projektionsverfahren (Röntgen)
Dimension der Daten → 2D oder 3D und kann man räumliche/zeitliche Zusammenhänge erkennen?

Wie bildet der Rechner die Realität als Bild ab?

Die echte Welt ist nicht in Pixel und eine Matrix unterteilt. Deshalb muss der Computer für Rasterbilder eine Matrix generieren. Für jeden Bildpunkt/Pixel wird der Lichtwert notiert – wie hell ist der Punkt.

Wir bilden also immer kleine Ausschnitte der Realität auf einen Zahlenwert ab. Die Größe der Pixel kann Variabel sein. Viel wichtiger ist die Größe der Matrix und die Anzahl der Pixel, weil das darüber entscheidet, wie fein man die Realität

Wann ist es von Vorteil
Pixel zu nutzen, die 6-
eckig sind, nicht 4-eckig?

Was wird bei digitalen Bildern diskretisiert, um digitale Bilder zu erschaffen. Wieso ist das notwendig?

Was ist ein Pixel?

abbilden kann.

2D Bilder können deshalb als Funktion abgebildet werden

Bild = zweidimensionale Lichtintensitätsfunktion $f(x, y) = g$ mit den Ortskoordinaten x, y und dem zugeordneten Grau-/Intensitätswert g

→ Wenn man Nachbarschaften betrachtet.

→ Hat den Vorteil, dass man immer denselben Nachbarn hat.

Die stetigen Elemente der realen Welt können nicht in einem Bild eingefangen werden. Bilder sind diskret, nicht stetig und die Diskretisierung erfolgt auf 2 Ebenen. Ein "digitales Bild" ist ein Bild, bei dem sowohl die **Ortskoordinaten** als auch die **Helligkeitswerte** diskretisiert wurden:

→ Ortskoordinierung : Ich bilde ein reales Objekt auf einem Gitter ab und gebe jedem Punkt im Gitter einen Helligkeitswerte.

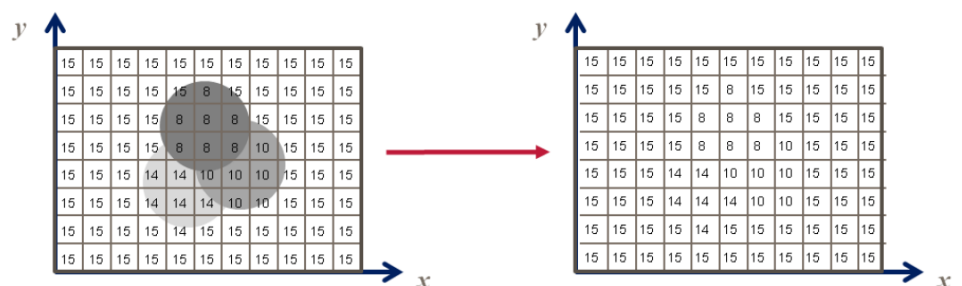
→ Helligkeitswert : Die Helligkeit kann nicht beliebig groß sein. Es gibt eine

bestimmte Anzahl an Helligkeitsabstufungen – z.b. 16 oder 256 Helligkeitsstufen.

Ich kann also Analog einen Wert wie 13,5 messen und muss mich dann entscheiden, ob es 13 oder 14 ist.

→ Man verliert immer Information durch Diskretisierung, weil ich jedem Pixel nur einen Farbwert gebe und jeder Farbwert folgt einer diskreten Anzahl an Helligkeitsstufen.

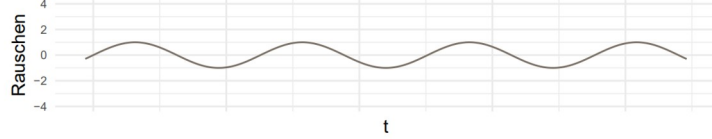
→ Pixel: kleinste Einheit eines digitalen (gerasterten) Bildes (von engl. picture element)



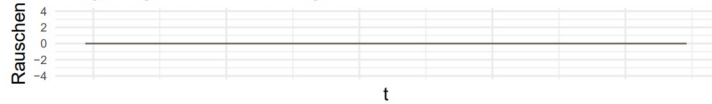
<p>→ Bilder als Matrix können durch eine Formel beschrieben werden. Was sagt die Formel aus (nicht Klausurrelevant)?</p>	<p>Diskrete, deterministische Ortsbereichsfunktion</p> $f : R \times R \rightarrow R$ <ul style="list-style-type: none"> • Abtastung: Diskretisierung des Argumentbereichs (Ortsbereich) • Quantisierung: Diskretisierung des Wertebereichs $\underline{M} \times \underline{N} \rightarrow \underline{G}$ <p>mit</p> $\underline{M} = \{0, 1, \dots, M - 1\},$ $\underline{N} = \{0, 1, \dots, M - 1\},$ $\underline{G} = \{0, 1, \dots, G - 1\}$ <p>Man kann das Bild als Matrix auch als mathematische Funktion beschreiben. In der analogen Welt folgt die Abbildung $R \times R \rightarrow R$, weil reelle Zahlen mit reellen Zahlen Ergebnisse erbringen, die auch reell sind. Ich habe also Werte in dezimalbereich, die die Welt beschreiben – zb. Helligkeitswerte. Ich kann also die Welt beliebig tief beschreiben. Durch die Abtastung erhalte ich natürliche, diskrete Zahlen. Sowohl M,N als auch G sind natürliche Zahlen. Wir bilden also natürliche Zahlen und <u>nicht</u> reelle Zahlen im Raum ab.</p>
<p>Welche Graubilder unterscheidet man -2?</p> <p>Wie sind Farbbilder im Computer aufgebaut</p> <p>→ Wie funktioniert der RGB-Kanal?</p> <p>→ Was ist die Sonderform des Farbkanaals?</p> <p>→ Wo spielt es eine Rolle?</p> <p>→ Kann man Graubilder als Farbbilder abspeichern und wie funktioniert es?</p> <p>→ Was sind 3 gebräuchliche Farbkanaäle?</p>	<p>→ Binärbilder mit 2 Werten</p> <p>→ Grauwertbilder, die 2^n folgen. Bsp $2^4 = 16$ Werte oder $2^8=256$.</p> <p>→ Man hat Farbkanaäle → bsp RGB-Kanal</p> <p>→ Farbbilder verwenden abhängig vom Farbmodell mehrere Farbkanaäle (eine Matrix je Farbkanal).</p> <p>→ Beim RGB-Farbmodell werden z.B. separate Werte für den Rot-, Grün- und Blauanteil des Pixels verwendet. Häufig werden Werte von 0 bis 255 je Farbe (8 Bit) genutzt</p> <p>→ Farbkanal mit Transparenz. Man hat einen 4.Kanal neben den RGB-Kanälen</p> <p>→ Zu den Werten für die Farbkodierung wird noch ein vierter Kanal zugefügt, der sogenannte Alpha-Kanal. Dieser Kanal definiert, ob das Pixel transparent (Wert 0) oder opak (Wert 255) angezeigt wird.</p> <p>→ Der Alpha-Kanal spielt u.a. bei der 3D-Visualisierung von Bildvolumina eine Rolle.</p> <p>→ Ja, dann werden die Farbwerte über alle Farbkanaäle als selber Farbwert abgespeichert. Grauwert 12 würde für rot, grün und blau Kanal also drei mal 12 sein.</p> <p>→ Nimmt mehr Speicher ein.</p> <p>→ RGB (Kanäle für rot/grün/blau)</p> <p>• CMYK (Kanäle für cyan/magenta/yellow/ key = Schwarzanteil, häufig bei Druckern)</p> <p>• HSL (Kanäle für hue/saturation/lightness)</p>
<p>Welche 4 Stichworte sind wichtig, wenn man über</p>	<p>1. Artefakte: Fehlerhafte Darstellung in digitalen Bilddaten <u>aufgrund verschiedener (technischer) Gegebenheiten.</u></p> <p><u>Vorhandene Strukturen werden nicht korrekt abgebildet</u> oder es “erscheinen”</p>

<p>die Qualität digitaler Bilder redet? + definiere sie</p>	<p>eigentlich nicht vorhandene Strukturen. Bsp. Für Erscheinen von Strukturen – Jpeg-Codierung führt Farbflecken ein, die eigentlich nicht da sind. Kleine Bereiche in Bildern, die aufgrund von technischen Begebenheiten oder aufgrund von falschen Berechnungen passiert sind.</p> <p>2.Ortsauflösung: Beim Aufzeichnen / Abtasten verwendete Rastergröße. • Ein feines Raster ermöglicht detailreiche Darstellung, • eine (zu) niedrige Ortsauflösung kann zur Bildung von Artefakten (“Treppenstufen” oder “Bauklötze”) im Bild führen. - <u>Je feiner Raster (je mehr Pixel pro Matrix), desto detaillierter kann ich das Bild abbilden.</u></p> <p>3. Intensitätsauflösung: Zahl “unterscheidbarer Farben/Grauwerte”. • Der darzustellende Farb- bzw. Grauwertebereich wird in äquidistante Intervalle unterteilt (siehe oben). • Hohe Intensitätsauflösung: detailreiche Darstellung, • (zu) niedrige Intensitätsauflösung: es entstehen möglicherweise virtuelle “Konturen” im Bild (=Artefakte), die in der Realität nicht vorhanden sind (auch als Aliasing bezeichnet). Bsp. Wenn ich ein Farbbild als Schwarzweiß abbilde oder nur mit 4 bit, statt 8 bit Channel, dann geht die Information über die Farbe verloren, weil ich nicht mehr in 0 bis 255 Werten abbilde.</p> <p>4. Bei der Wahl der passenden (Orts- und Intensitäts-) Auflösung spielt auch wieder das Abtasttheorem eine Rolle: • Z.B. sollte die feinste im Bild darzustellende Struktur (beispielsweise der Querschnitt durch ein Blutgefäß) durch mindestens 2 (eher mehr) Bildpunkte repräsentiert werden. Nyquist-Shannon.</p>
<p>Signale vs. Rauschen Signal Noise Ratio.</p> <p>Was sagt die Formel für das Signal-Noise-Ratio aus? Wie lautet es?</p> <p>Was sind Ursachen für Rauschen? -2</p>	<p>Signale (allgemein, auch wenn sie nicht bildlich dargestellt werden) liegen üblicherweise zunächst analog vor.</p> <p>Rauschen und Störungen bei der Erfassung von Signalen bzw. Bilddaten sind oft nicht zu vermeiden.</p> <p>Ursache für Rauschen in digitalen Bildern. Unterschiedliche Fehlerquellen, z.B. Eigenschaften von Elektroden/Sensoren, von Verstärkern und Elektronik, elektromagnetische Verschmutzungen • Digitale “Ungenauigkeiten” im Vergleich zum analogen Signal</p> <p>Die Qualität eines aufgezeichneten Signals kann u.a. durch das Signal-Rausch-Verhältnis (engl. signal to noise ratio) wiedergegeben werden:</p> $SNR = \frac{\text{mittlere Leistung des Nutzsignals}}{\text{mittlere Rauschleistung des Störsignals}} = \frac{P_{\text{signal}}}{P_{\text{noise}}}$ <p>Teile mittlere Leistung des Signals durch mittlere Rauschleistung des Störsignals. Je mehr Nutzsignal dominiert, desto größer wird der Wert und er ist dann größer 1. Je mehr Rauschen überwiegt, desto kleiner ist SNR-Wert und es ist kleiner 1. Kann so darstellen, wie verrauscht ein Signal ist.</p> <p>Auf diese Art und Weise kann ich darstellen, wie verrauscht ein Signal ist</p> <p>Das nächste dient der Verdeutlichung, nicht auswendig lernen:</p>

Das $SNR = \frac{\text{mittlere Leistung des Nutzsignals}}{\text{mittlere Rauschleistung des Störsignals}} = \frac{P_{\text{signal}}}{P_{\text{noise}}}$ sollte einen möglichst hohen Wert haben:



jeweils mit Überlagerung durch das Störsignal:

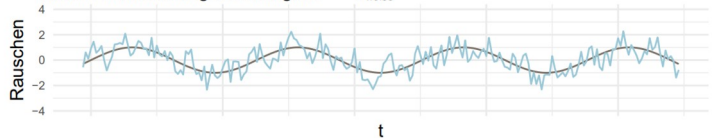


Störung — keine

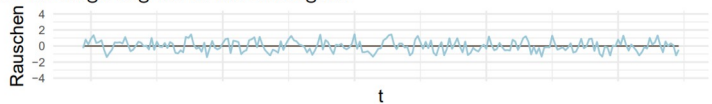
Dargestellt sind zwei verschieden stark ausgeprägte Störsignale (rote und hellblaue Signalverläufe), die dem Nutzsignal (grau) überlagert werden.

Wir haben in der Abbildung zwei Kurven. Eine Sinuskurve und ein konstantes Signal.

Das $SNR = \frac{\text{mittlere Leistung des Nutzsignals}}{\text{mittlere Rauschleistung des Störsignals}} = \frac{P_{\text{signal}}}{P_{\text{noise}}}$ sollte einen möglichst hohen Wert haben:



jeweils mit Überlagerung durch das Störsignal:



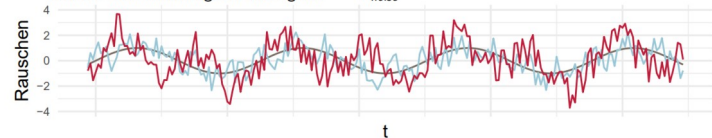
Störung — keine — gering

Dargestellt sind zwei verschieden stark ausgeprägte Störsignale (rote und hellblaue Signalverläufe), die dem Nutzsignal (grau) überlagert werden.

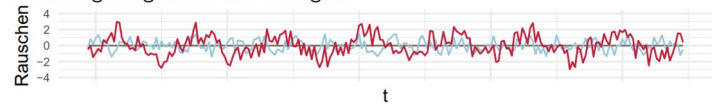
Legt man Rauschen über beide Signale (blaue Linie), dann sieht man, dass man trotz Rauschen noch die Sinuskurve erkennen kann, während man den Charakter der konstanten Linie nicht mehr erkennen kann.

Je weniger das Signal ausgeprägt ist, desto stärker ist die Wirkung des Rauschens auf das Signal und seinen Charakter.

Das $SNR = \frac{\text{mittlere Leistung des Nutzsignals}}{\text{mittlere Rauschleistung des Störsignals}} = \frac{P_{\text{signal}}}{P_{\text{noise}}}$ sollte einen möglichst hohen Wert haben:



jeweils mit Überlagerung durch das Störsignal:



Störung — keine — gering — stark

Dargestellt sind zwei verschieden stark ausgeprägte Störsignale (rote und hellblaue Signalverläufe), die dem Nutzsignal (grau) überlagert werden.

Rauschen kann aber auch schlimmer sein als das Rauschen im Blau.

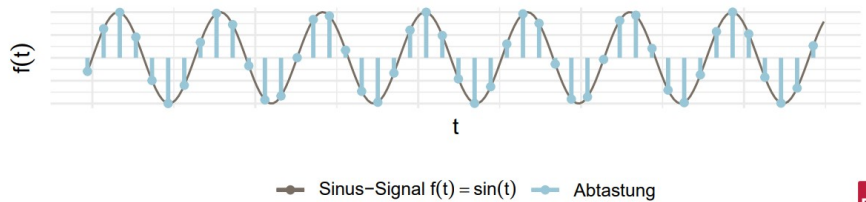
So können Rauschsignale selbst einen ausgeprägten Charakter haben, der die Ausprägung des Signals überdeckt.

Im oberen Bild hat das Rauschsignal selbst eine Sinuskurve. Dadurch kann man die Sinuskurve des Signals schwerer erkennen.

→ Aufpassen, dass man Rauschen möglichst rausbekommt.

Wie korreliert die Ausprägung des Signals mit dem Rauschen?

Abtastung



Wozu tasten wir Bilder ab?

Braucht man mehr oder weniger Abtaststellen für gute Rekonstruktionen?

Mit welcher SI-Einheit werden Frequenzen angegeben?

Was sagt Frequenz aus?
Was entspricht 1 Einheit der Frequenz?

Wie viel sind 44.000 Abtastungen pro Sekunde?

Wie hoch muss die Abtastfrequenz für eine gute Abtastung des Signals sein?

Was sagt Nyquist-Shannon-Theorem?

Man hat ein analoges Signal, welches eine Sinuskurve bildet (graue Linien). Dann hat man Abtaststellen, wo man zum Beispiel sagt, man möchte alle Tausende, 1/1000000 Sekunde oder was auch immer das Signal abtasten.

Die aufgezeichneten analogen Signale werden i.Allg. für die Verarbeitung im Computer in digitale Signale gewandelt: Aus einer kontinuierlichen Funktion wird eine diskrete Funktion (gleiches Prinzip wie bei den Bilddaten).

- Über die Abtastung kann ein analoges Signal im Digitalen abgebildet werden.
- Je mehr Abtaststellen, um so sicherer kann ich das Signal rekonstruieren.

Frequenzen werden mit der SI-Einheit "**Hertz**" (Einheitenzeichen Hz) angegeben

Die Frequenz gibt die **Anzahl sich wiederholender Vorgänge** (Schwingungen) pro Sekunde in einem (periodischen) Signal an.

Eine Schwingung pro Sekunde entspricht 1 Hz.
44.100 Abtastungen pro Sekunde = 44,1 kHz für ein Audiosample

Sie muss mindestens doppelt so hoch sein wie die maximale Frequenz $f_{max} = 1/\text{min. Periodendauer}$ des zugrunde liegenden Signals.

In der Praxis werden meist höhere Abtastraten verwendet, z.B. $f_a > 5 \cdot f_{max}$, um eine möglichst genaue Abtastung des ursprünglichen Signals zu erhalten.

Das Abtasttheorem gilt natürlich auch für digitale Bilder: Die feinste im Bild darzustellende Struktur (bspw. der Querschnitt durch ein Blutgefäß) sollte durch mindestens 2 (eher mehr) Bildpunkte repräsentiert werden.

(Nyquist-Shannon-)Abtasttheorem

Die Abtastfrequenz muss $f_a = \frac{1}{\text{Abtastintervall}} \geq 2 \cdot f_{max}$ erfüllen.

Siehe auch <https://de.wikipedia.org/wiki/Nyquist-Shannon-Abtasttheorem>

Für Bilder:

Nicht Klausurrelevant:


Bei Bildern nutzt man mindestens die doppelte **räumliche Frequenz**:

Man kann sich die räumliche Frequenz als die **Anzahl der Helligkeits- oder Farbwechsel innerhalb eines bestimmten Raumbereichs oder einer bestimmten Fläche vorstellen**. Eine hohe räumliche Frequenz bedeutet, dass im Bild viele solcher Veränderungen auftreten, während eine niedrige räumliche Frequenz bedeutet, dass im Bild nur wenige oder keine Veränderungen auftreten.

Die räumliche Frequenz wird oft in Zyklen pro Einheit (z.B. Zyklen pro Pixel oder Zyklen pro Millimeter) gemessen und gibt an, wie viele Veränderungen innerhalb einer bestimmten Entfernung oder Größe im Bild auftreten. Zum Beispiel würde ein Bild mit vielen feinen Details, wie beispielsweise ein Bild von Haaren oder Blättern, eine höhere räumliche Frequenz aufweisen als ein

Bildanalyse VL

Wieso nimmt man doch oft mehr als $2 * f_{\max}$?	<p>Bild von glatten Flächen mit wenigen Details, wie beispielsweise ein Himmel oder eine Wand.</p> <p>Wenn man eine Sinuskurve mit $2 * f_{\max}$ abbildet, dann kann man Glück haben und man deckt die Sinuskurve perfekt ab oder man verrutscht mit seinem „Abtast“-Signal nach rechts oder links und das Abtastsignal bildet das Original nur versetzt ab – also gar nicht. → Deshalb tastet man mit deutlich mehr als $2 * f_{\max}$, um besser abzutasten.</p>
Welche 2 Arten von Grafiken unterscheidet man?	<p>Blick zurück – Bild als Matrix / Rastergrafik</p> <ul style="list-style-type: none"> • Bild aus rechteckigen Bildelementen zusammengesetzt (Pixel), Diskretisierung erfolgt üblicherweise mit einem äquidistanten „Gitter“ • Eindeutige Lokalisation eines Bildpunktes über die Koordinaten für x, y • Pixelwerte kodieren Farb- (bei mehrkanaligen Bildern) bzw. Intensitätswerte (Grauwertbilder) <p>• Im Gegensatz dazu Vektorgrafik:</p> <ul style="list-style-type: none"> • Bilder werden aus geometrischen „Primitiven“ zusammengesetzt, z.B. Kreise, Rechtecke, Linien, Kurven, ... • Vorteil: beliebig skalierbar • Eine „Bildszene“ kann bei geeigneter Parametrierung der Primitive relativ genau zusammengestellt werden. • Kann z.B. genutzt werden, um aus Rastergrafiken segmentierte Objekte (Tumor, anatomische Struktur o.ä.) in einer Visualisierung einzublenden.
<p>DICOM</p> <p>Wofür steht DICOM?</p> <p>Wo wird der Standard verwendet?</p> <p>Wozu dient DICOM? - 3</p> <p>Welche Dienste bietet DICOM? - 5</p> <p>Wie viele Graustufen unterstützt DICOM-Bilddaten?</p> <p>Wie werden Metadaten gespeichert?</p> <p>Wie werden Attribute definiert?</p>	<p>Abkürzung für “Digital Imaging and Communications in Medicine”, ein sog. offener Standard.</p> <p>Dieser Standard wird in nahezu allen modernen bildgebenden bzw. bildverarbeitenden Systemen in der Medizin verwendet.</p> <p>Definition von Datenstrukturen zur Speicherung von medizinischen Bilddaten und zugehörigen Informationen (sog. Metadaten). Der Standard definiert, wie Bilddaten gespeichert und zwischen unterschiedlichen Systemen ausgetauscht werden können. Ermöglicht eine weitgehende Interoperabilität</p> <p>Netzwerkorientierte Dienste</p> <ul style="list-style-type: none"> • Protokolle zur Übertragung von Bilddaten. • Abfrage von Daten eines Bildarchivs (PACS) durch einen Client. • Integration von Radiologieinformationssystem (RIS) – Picture Archiving and Communication System (PACS) – bildgebender Modalität. - Fensterung (oft mit sog. Presets, d.h. Voreinstellungen für die verbesserte Darstellung bestimmter Gewebearten / Strukturen) und Falschfarbendarstellung - diverse Filter / Algorithmen zur Aufbereitung und weiteren Bearbeitung <p>65.536 Graustufen (CT z.B. $2^{12} = 4.096$).</p> <p>Metainformationen werden über Attribute verwaltet</p> <p>Der DICOM-Standard definiert für jede Modalität, welche Angaben (Attribute)</p>

	<p>zwingend vorgeschrieben, optional oder nur unter bestimmten Voraussetzungen (z.B. Kontrastmittelgabe) vorgeschrieben sind.</p>
<p>Was ist Fensterung?</p> <p>Wieso müssen wir Bilder Fenstern?</p>	<p>Sorgt dafür, dass das, was wir als eingeschränkten Sehbereich haben, entfernt wird, damit die Elemente des Bildes möglichst kontrastreich und gut aufgeteilt dargestellt werden?</p> <p>Die sogenannte Fensterung optimiert den Bildkontrast für die gewünschten Gewebe.</p> <p>Wieso müssen wir Bilder Fenstern?</p> <p>Das menschliche Auge kann nur eine geringe Anzahl an (ähnlichen) Graustufen differenzieren/ unterscheiden (je nach Quelle zwischen 30–100).</p> <p>Auch für die Darstellung medizinischer Datensätze verwendete Monitore können nur eine begrenzte Anzahl an Graustufen abbilden.</p> <p>Problem: Ohne eine Anpassung kommt es evtl. zu einer kontrastarmen Darstellung der aufgezeichneten Strukturen.</p> <p>→ Man muss es also für Mensch besser darstellen, wegen technischen Einschränkungen (was kann Monitor darstellen) und was Mensch sehen kann.</p>
<p>Wie funktioniert Fensterung?</p>	 <p>Man sieht im Abbildung</p> <p>Unten Links ein Grauerthistogramm, der anzeigt, wie sich die Grauwerte im Bild verteilen.</p> <p>Man sieht auch eine Linie.</p> <p>Die Linie sagt, ich bilde die Grauwerte vom niedrigsten Wert (schwarz) bis zum höchsten Wert (weiß) linear ab (normalerweise linear. Abbildung hat es schon modifiziert).</p> <p>→ Man versucht jetzt den interessanten Bereich durch Verstärkung einiger Bereiche und Schwächung anderer Bereiche besser darzustellen.</p> <p>Hier haben wir einen Bereich definiert. Alles, was darunter liegt, wird direkt auf schwarz gestellt, alles, was darüber ist direkt auf weiß.</p> <p>1. window width (Breite: "w") u - Die Fensterbreite bezieht sich auf die Bandbreite von Grauwerten, die in einem Bild dargestellt werden sollen. Sie bestimmt den Kontrast des Bildes, indem sie den Bereich von Grauwerten bestimmt, die als "weiß" und "schwarz" dargestellt werden sollen. Eine größere Fensterbreite bedeutet einen größeren Kontrastbereich, während eine kleinere Fensterbreite bedeutet, dass der Kontrastbereich kleiner ist.</p>
<p>Was stellt man für die Fensterung ein? -2</p>	

Wozu dienen die beiden Einstellungen?

Wie kann man Fensterung bei CT-Bildern ausnutzen?
Wie kann man Gewebe für Fensterungen im CT nutzen?

Welche Einheit nutzt CT dafür?

2. **“level (Zentrum:”c” oder “l”)** - bezieht sich auf den Mittelpunkt des Grauwertbereichs, der als "Mitte" des Bildes dargestellt werden soll. Sie bestimmt die Helligkeit des Bildes und ermöglicht es, den Bereich der Grauwerte zu verschieben, der als "Mittelton" dargestellt wird. Eine höhere Fensterposition bedeutet eine hellere Darstellung des Bildes, während eine niedrigere Fensterposition zu einer dunkleren Darstellung führt.

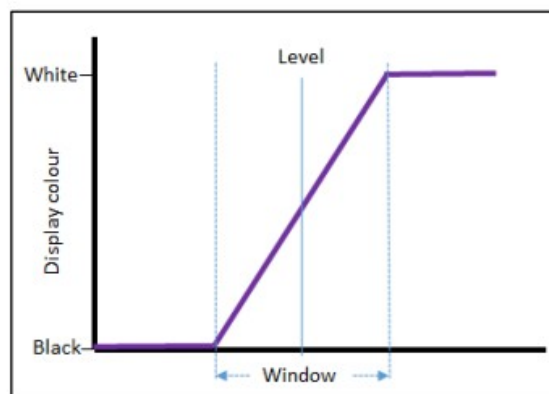


Figure 2.6: Window width and window level in digital imaging.

Einstellungen zur Anpassung der Helligkeit und des Kontrasts eines digitalen Bildes.

CT stellt da, wie gut Röntgenstrahlung absorbiert wird.

Wert 0 ist auf Wasser normiert.

Bestimmte Gewebetypen haben dann entsprechend höhere oder tiefere Werte.

Wenn man weiß, welchen Wert ein bestimmter Gewebetyp hat im CT, so kann man sich schnell das richtige Fenster für das nötige Gewebe erschließen.

Hounsfield-Einheiten (HE)

<p>Was ist Filterung?</p> <p>In welchen 2 Räumen kann man Filterung vornehmen?</p>	<p>→ Es geht darum bestimmte Signalbestandteile zu verbergen und die Signalbestandteile, die man sehen möchte, werden verstärkt.</p> <p>→ Bsp Bildstörung oder Rauschen reduzieren.</p> <p>→ Im Ortsraum oder Frequenzraum.</p> <p>→ Man kann alle Verarbeitungsschritte entweder im einem Raum oder dem anderen machen → zumindest die linearen Ansätze.</p> <p>→ Je nach Anwendungsfall macht es Sinn, bestimmte Filter, Verarbeitungs- oder Rechenschritte eher im Orts- oder Frequenzraum durchzuführen.</p>
<p>Was ist der Frequenzraum? (Ortsraum weiter oben).</p> <p>Kann man Orts in Frequenzraum (und umgekehrt) überführen?</p> <p>Geht Information bei der Umwandlung verloren?</p> <p>Was ist der Nachteil einer Fourier-Transformation?</p> <p>Abbildung als Beispiel: Ortsraum oben vs Frequenzraum unten.</p> <p>Was bedeutet eine Frequenz von 0?</p>	<p>Im Frequenzraum wird das Signal/Bild als Zusammensetzung einzelner Grundfrequenzen (z.B. von Sinuskurven mit verschiedenen Frequenzen) beschrieben.</p> <p>→ Man versteht also darunter ein Fourier-transformiertes Bild.</p> <p>Man nutzt Fourier für die einzelnen Werte im Bild. Man zerlegt das Bild und erhält so Koeffizienten. Kann so gut bestimmte Frequenzbereiche im Bild manipulieren.</p> <ul style="list-style-type: none"> • Die Zerlegung erfolgt bspw. mittels Fourier-Transformation. • Die Koeffizienten der jeweiligen Frequenzen, die diese Zusammensetzung beschreiben entsprechen den Amplituden, mit denen die jeweiligen Frequenzen im Signal/Bild vorkommen. • Die Frequenzen entsprechen dabei den Reziprok-Werten der Wellenlängen. • Bei einer Filterung im Frequenzraum werden bestimmte Frequenzen bzw. Frequenzbereiche herausgefiltert. • Die Rücktransformation in den Ortsraum zur Rekonstruktion des gefilterten Signals erfolgt dann mittels inverser Fourier-Transformation. <p>• Transformation Ortsraum \leftrightarrow Frequenzraum: Jedes Bild oder Signal kann in jedem beliebigen Raum gefiltert werden und wieder rücktransformiert werden (siehe Fouriertransformation und inverse Fouriertransformation).</p> <p>→ Verlustfrei</p> <p>→ Sehr langsam: Transformation dauert sehr lange.</p> <div data-bbox="603 1487 1423 1883"> <p>The top graph shows a signal $f(t)$ over time t from 0 to 30. The signal is a complex wave composed of multiple frequencies. The bottom graph shows the frequency components (Frequenz in Hz) from 0 to 20. The y-axis represents the amplitude (Ausprägung) from 0 to 4000. There are three prominent peaks at 3 Hz, 7 Hz, and 10 Hz, corresponding to the frequencies mentioned in the title 'Original mit 3, 7, 10 Hz'.</p> </div> <p>nur für Signale, auf Bildern schauen wir uns das später noch an...</p> <p>→ Eine Frequenz von 0 wäre eine Gerade → Konstante.</p>

<p>Welche Statistischen Methoden gibt es für die Bildbearbeitung? -4 → In welchem Raum werden Sie verwendet?</p>	<p>Grauerthistogramme, Farbhistogramme,</p> <ul style="list-style-type: none"> • Entropie → Später in KI nochmal wichtig • Mittlerer Grauwert, Kontrast <p>→ Ortsraum.</p> <p>Alles, was man in Excel berechnen kann, kann man auch mit Bilddaten berechnen. Entropie eines Bildes unterscheidet sich nicht von anderen Signalen.</p> <p>Mittlerer Grauwert → Der Mittelwert aller Grauwerte, die im Bild benutzt werden.</p>
<p>Statistische Methode 1: Ortsraum</p> <p>Grauwert- und Farbhistogramme</p> <p>Was bilden diese Histogramme ab?</p> <p>In was können Werte zusammengefasst werden? → Wie nennt man die Klassen?</p> <p>Welche Informationen können durch Histogramme nicht abgebildet werden?</p> <p>Was ist ein Absolutes Histogramm? → Was ist die Summe aller Einträge? → Wovon abhängig?</p> <p>Was ist ein Normiertes Histogramm? → Was ist die Summe aller Einträge? → Was ist der Vorteil?</p>	<div data-bbox="568 651 831 801" data-label="Image"> </div> <div data-bbox="970 651 1442 875" data-label="Figure"> <p>Code: pythoncode/opencv-farbhistogramm.py</p> </div> <p>Farbbilder werden im RGB-Channel farbig abgebildet. Man kann die drei Kanäle im Histogramm abbilden.</p> <p>Y- Achse: Anzahl an Pixel mit diesem Wert. X- Achse: repräsentiert die Helligkeitswerte von 0 bis 255, wobei 0 für Schwarz steht und 255 für alle Channels für Weiß.</p> <ul style="list-style-type: none"> • Ein Grauwert- oder auf mehreren Farbkanälen basierendes Histogramm repräsentiert die Häufigkeitsverteilung der Grauwerte (bzw. Farbwerte je Kanal) der Bildpunkte eines Bildes <p>Die Werte können auch in Klassen (Bins) zusammengefasst werden. Farb-/Intensitätswerte werden dann nicht mehr separat, sondern z.B. in Gruppen von n Werten betrachtet: [0,1,2], [3,4,5], ...</p> <p>Histogramme enthalten keine Orts-, Nachbarschafts- oder sonstige Information über den Bildinhalt. Darstellung häufig in Kurven- oder Balkenform.</p> <p>$H(i)$, $0 \leq i < \max$. Anzahl Bins bzw. Grau-/Intensitätswerte • Auf der Ordinate (y-Achse) werden die absoluten Häufigkeiten der jeweiligen Werte (bzw. Grauwertklassen/Bins) aufgetragen, auf der Abszisse (x-Achse) die Grauwertklassen/Bins.</p> <ul style="list-style-type: none"> • Die Summe aller Werte entspricht der Zahl der Bildpunkte. <p>Das ist im Grunde das Histogramm oben, wo auf der Y-Achse alle Pixel mit der bestimmten Intensität abgebildet werden. Hier werden alle Pixel mit einem bestimmten Wert aufaddiert.</p> <p>→ Abhängig von der Anzahl der Pixel und damit Bildgröße wird Größe der Bins bestimmt.</p> <p>$H_{\text{norm}}(i) = H(i)/(M \cdot N)$ mit $0 \leq i < \max$. Anzahl Bins bzw. Grau-/Intensitätswerte (für ein Bild der Größe $M \times N$).</p> <ul style="list-style-type: none"> • Teilen der Einträge des absoluten Histogramms H durch die Zahl der Pixel ($M \times N$) im Bild. • Gibt für jeden Wert die Wahrscheinlichkeit (auch: $p(g)$) an, dass ein (beliebiges) Pixel diesen Grauwert hat (bzw. der Grauwertklasse i angehört, dann $p(i)$).

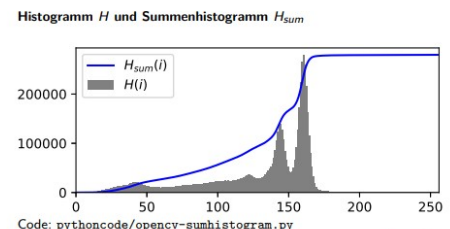
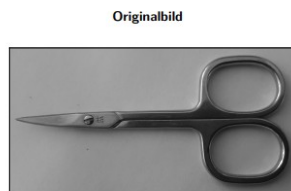
Was ist die Besonderheit von OpenCV bezogen auf den Farbkanal?

Was ist ein **Summenhistogramm**?

Was wird dadurch gut abgebildet (blaue Linie)?

- Summe aller Einträge ist **1**.
Man teilt durch die Größe des Bildes. Setzt so Summe aller Einträge auf 1.
Vorteil: Kann **unterschiedlich große Bilder miteinander vergleichen**.

Aufpassen: Es arbeitet mit BGR, nicht RGB – Andere Reihenfolge.



i, j : Zähler des aktuell zu betrachtenden Bins (Klassen). i, j können den Grau-/Intensitätswerten entsprechen, wenn Anzahl der Grauwerte im Bild der Anzahl der Bins entspricht.

- $H(i)$: Histogrammwert (d.h. Häufigkeit der Grau-/Intensitätswerte) in Bin i
- $Hsum(i) = \sum_{j=0}^i H(j)$

Man bildet nicht den Farbwert an einer Stelle ab, sondern man bildet die Summe.

Man hat also die **Menge, die kleiner oder gleich einem bestimmten Wert im Histogramm sind**.

Man erkennt, wann **neue Werte** hinzukommen.

Man erkennt gut die **Helligkeitsverteilung** im Bild.

Statistische Methode:
→ Ortsraum

Entropie

Was ist Entropie?
Wie kann man die Zahlenwerte interpretieren?

Wo trifft man auf eine Entropie von 0?

Wo wird Entropie



$E = 6.563$



$E = 7.417$



$E = 0$

Die Entropie E eines Bildes ist ein (theoretisches) **Maß für dessen Informationsgehalt**:

• **Hohe Entropie**: Bildinhalt “unberechenbar”, enthält viel “Zufälligkeit” → **geringe Redundanz**.

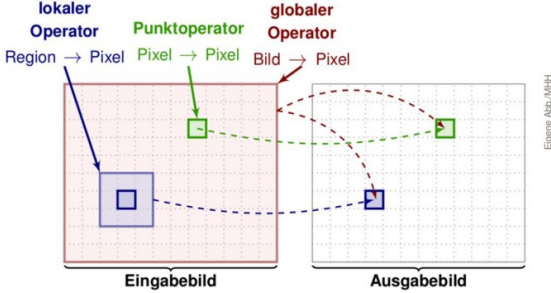
• **Niedrige Entropie**: Bildinhalt “berechenbar(er)”, wenig Zufälligkeit → **hohe Redundanz**.

• Die Berechnung erfolgt unter **Verwendung des Histogramms** (bzw. der Wahrscheinlichkeit des Auftretens eines Grauwertes $p(g)$, \log_2 ist Log. zur Basis 2):

$$E = - \sum_{g \in G} p(g) \cdot \log_2 \left(\frac{1}{p(g)} \right)$$

- Einsatz beispielsweise zur **Festlegung der unteren Schranke für die Bitrate bei Signalkompressionsverfahren**.

<p>verwendet?</p> <p>Wie hängt Entropie und Bildbeschreibung zusammen? Welchen Effekt hat ein Blurring des Bildes</p>	<ul style="list-style-type: none"> • Die Entropie eines homogenen Bildes beträgt 0. <p>Entropie: Man hat also viele Wertveränderungen, wenn die Entropie hoch ist. 0 bedeutet, dass sich die Farbwerte nie ändern entlang des Bildes.</p> <p>→ je mehr Entropie ich habe, desto mehr Daten brauche ich, um Bild zu beschreiben.</p> <p>Blur senkt Entropie</p>
<p>Statistische Methode:</p> <p>Ortsraum</p> <p>Mittlerer Grauwert → Was ist das?</p> <p>Globaler Kontrast → Was ist das? → Was brauche ich für die Berechnung?</p>	<p>Mittlerer Grauwert für ein Bild (f) mit den Abmessungen M, N (m, n = Laufindizes), u.a. als Maß für die Helligkeit eines Bildes oder Basis weiterer Berechnungen. Mittlerer Grauwert → Der Mittelwert aller Grauwerte, die im Bild benutzt werden.</p> $\bar{g} = \frac{1}{M \cdot N} \sum_{m,n} f(m, n)$ <p>Ich teile die Grauwert-Summe durch die Größe des Bildes (alle Pixel).</p> <p>Globaler Kontrast (je nach Quelle gerne etwas unterschiedlich definiert)</p> $q = \frac{1}{M \cdot N} \sum_{m,n} f(m, n) - \bar{g} \text{ oder auch } q = \frac{1}{M \cdot N} \sum_{m,n} (f(m, n) - \bar{g})^2$ <p>Ich brauche den mittleren Grauwert. Ich subtrahiere vom Wert an jeder Stelle den Mittelwert ab. Bilde die Summe und Teile es wieder durch Anzahl der Bildpunkte.</p> <p>→ Maß für, wie unterschiedliche Hell die Punkte im Bild sind.</p> <p>Wie Standardabweichung des Bildes?</p>

<p>Welche Arten von Operatoren gibt es, um Bilder im Ortsraum zu transformieren? - 3</p> <p>Erkläre was alle 3 Arten für die Transformation nutzen?</p>	<p>Punktorientierte Operatoren / Punktoperatoren</p> <ul style="list-style-type: none"> • Werte eines Pixels werden (isoliert) verändert, z.B. Schwellwert, Addition und Subtraktion sowie logische Operationen (bei mehreren Eingabebildern). - Schaue mir den Wert von einem Pixel an. <p>• Lokale Operatoren</p> <ul style="list-style-type: none"> • Werte eines Pixels werden in Abhängigkeit von den Werten der Pixel in der Umgebung (Nachbarschaft) verändert. • Problem: Was geschieht mit den am Rand gelegenen Pixeln? (z.B. Randbereiche duplizieren) → Pixel + seine Umgebung. <p>• Globale Operatoren</p> <ul style="list-style-type: none"> • Benötigen sämtliche Bildpunkte des Eingabebildes, um das Ergebnis für die Ausgabepixel zu berechnen, z.B. Überführung vom Orts- in den Frequenzraum mittels Fouriertransformation. → Spiegeln des Bildes als Beispiel. <p>In der Literatur häufig: Operatoren \Leftrightarrow Filter</p> 
<p>Nenne Beispiele für Punktoperatoren - 5</p>	<p>- Pixelrelation</p> <p>- Histogrammspreizung</p> <p>- Histogrammäqualisation</p> <p>- Invertierung = Verändere den Graustufenwert auf Basis eines globalen Wertes, den ich vorher ermittelt habe.</p> <p>- Schwellwertfilterung = bsp mache aus einem Bildelement ein schwarz weiß Bild, je nachdem ob Wert für den Pixel über oder unter einem bestimmten Wert war.</p> <p style="text-align: right;">1.</p>
<p>Punktoperator – 1 Ortsraum</p> <p>Pixelrelation mit mehreren Eingabebildern, logischen Operatoren.</p>	<p>Pixelrelationen mit mehreren Eingabebildern</p> <ul style="list-style-type: none"> • Arithmetische Operationen wie Subtraktion oder Addition der Eingabebilder, z.B. zum "Einblenden" von Segmentierungsergebnissen in die Originaldaten. → Aus Präsenz: wir haben ein Farbbild in 3 Bilder aufgeteilt, wobei jedes Bild einen Farbchannel darstellt und haben es dann wieder zu einem Bild zusammengebaut. <ul style="list-style-type: none"> • Logische Operationen • UND-Verknüpfung • ODER-Verknüpfung, → Welche Flächen oder Farbwerte treffen zwischen zwei Bildern überein als Bsp.

Punktoperator -2
Ortsraum
Histogrammtransformation -1

Histogrammspreizung

Was ist Spreizung
Wozu ist es gut?

Erfolgt die Spreizung mit
Verlust an Information?

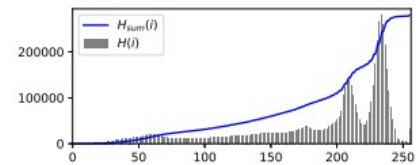
Transformationsfunktion für (Grau-)Werte g :

$$T_{stretch}(g) = \left[G \cdot \frac{g - g_{min}}{g_{max} - g_{min}} \right]$$

Nach Anwendung der Histogrammspreizung



Neue Grauwertverteilung



Zuvor gezeigtes Bild der Schere: die beiden “Gebirge” liegen (zu) dicht aneinander, der verfügbare Grauwertbereich wird nicht vollständig ausgenutzt, das Bild wirkt dunkel.

Nach Histogrammspreizung werden die im Bild enthaltenen Grauwerte $\{g_{min}, \dots, g_{max}\}$ besser ausgenutzt, d.h. **auf den gesamten verfügbaren Bereich $\{0, \dots, G\}$ (z.B. 0 bis 255) abgebildet.**

Die Histogrammspreizung ist eine Methode, um den Kontrast eines Bildes zu verbessern, indem der **gesamte verfügbare Helligkeitsbereich optimal genutzt wird**. Sie erweitert die Verteilung der Helligkeitswerte im Bild, sodass **dunkle Bereiche dunkler und helle Bereiche heller werden**. Die Histogrammspreizung ist eine Technik, bei der ein Bild aufgehellt und kontrastreicher gemacht wird, indem die dunkelsten Pixel im Bild noch dunkler und die hellsten Pixel noch heller werden. Dadurch werden Details im Bild besser sichtbar und das Bild erscheint lebendiger.

Man kann es sich wie das Ausdehnen eines Gummibandes vorstellen, um den gesamten verfügbaren Platz auszunutzen. Die **dunkelsten Pixel werden dabei näher an Schwarz und die hellsten Pixel näher an Weiß geschoben**, wodurch das Bild einen größeren Bereich an Helligkeitswerten abdeckt.

In unserem Beispiel hatte die Schere keine Werte, die über 200 stiegen oder wirklich 0 waren. Wir haben hier also die Werte so verändert, dass es jetzt auch pixel gibt, die den Wert 255 annehmen und wir haben jetzt dunkle Elemente, die fast Schwarz waren (etwas über 0), die jetzt wirklich schwarz sind (auf 0 gesetzt wurden). Man sieht es, wenn man die Bilder vergleicht. Die Ecke oben links ist jetzt komplett weiß, nicht nur hell grau, weil die Wert auf 255 gesetzt wurden.

Da durch die Spreizung kein “Zusammenfassen” von Grauwerten stattfindet, ist **die Transformation nicht verlustbehaftet (und umkehrbar)**. Kein Verlust an

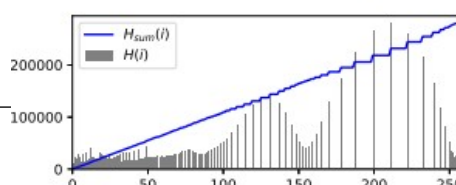
! Nach Anwendung der Histogrammäqualisation

Punktoperator -3
Ortsraum
Histogrammtransformation -2

Histogrammäqualisation


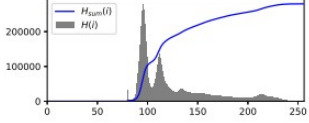



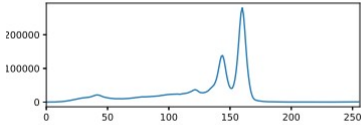


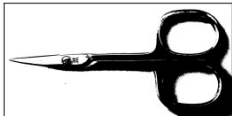
Neue Grauwertverteilung

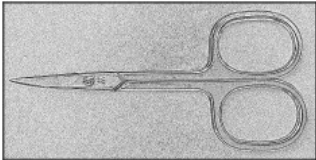
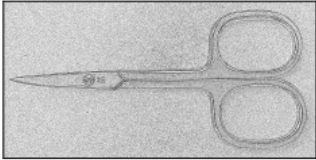


Ursprüngliche Verteilung der Grau-/Intensitätswerte (d.h. das Histogramm des Originalbildes H_{orig}) wird durch die Histogrammäqualisation auf eine **neue, “breitere” und “gleichmäßigere” Verteilung abgebildet**

• Die Transformationsfunktion

<p>Erfolgt die Äquivalisation Verlustbehaftet?</p> <p>Definiere Histogrammäqualisation</p>	<p>verwendet das relative Summenhistogramm $H_{sum,norm}$, also das auf den Wertebereich 0 bis 1 skalierte Summenhistogramm H_{sum}: $T_{eq}(g) = [G \cdot H_{sum,norm}(g)]$</p> <ul style="list-style-type: none"> Die Intensitätswerte sind danach über den gesamten Wertebereich verteilt, neues H_{sum} erscheint "linear". Verlustbehaftet (und nicht umkehrbar), da größere Bereiche der Skala mit Grauwerten geringer Häufigkeit auf wenige Grauwerte komprimiert werden. <p>Ja, es erfolgt mit Verlust an Information. Nicht umkehrbar.</p> <p>Die Histogrammäqualisation ist eine Technik, bei der die Helligkeitswerte in einem Bild so angepasst werden, dass dunkle Bereiche heller und helle Bereiche dunkler werden. Dadurch wird das Bild kontrastreicher und die Helligkeitswerte verteilen sich gleichmäßiger über den gesamten verfügbaren Bereich.</p> <p>Stell dir vor, du hast ein Bild, bei dem die dunklen Bereiche zu dunkel und die hellen Bereiche zu hell sind. Mit der Histogrammäqualisation kannst du die Helligkeitswerte im Bild so anpassen, dass die dunklen Bereiche heller und die hellen Bereiche dunkler werden, wodurch ein ausgewogeneres und kontrastreicheres Bild entsteht. Es ist, als ob man die Farben im Bild neu verteilt, um die bestmögliche Balance zu erreichen.</p> <p>Dadurch sind alle Grauwerte gleichmäßig und gleich oft vertreten.</p>
<p>Punktorperator -4 Ortsraum Histogrammtransformation -3</p> <p>Invertierung</p> <p>Was macht die Invertierung</p> <p>Passiert Invertierung mit Verlust an Inforationen?</p>	<div style="display: flex; align-items: center;"> <div style="flex: 1;"> <p>Nach Anwendung der Invertierung</p>  <p>Neue Grauwertverteilung</p>  <p>Code: pythoncode/opencv-invertierung.py</p> </div> <div style="flex: 1; padding-left: 20px;"> $T_{inv}(g) = [G - g]$ <p>für die Invertierung der (Grau-)Werte g im Bild</p> <p>Die Transformation ist nicht verlustbehaftet (und umkehrbar)</p> </div> </div>
<p>Punktorperator 5</p> <p>Schwellwertfilterung/ Binarisierung.</p> <p>Was ist Binarisierung?</p>	<p>Bei der sog. Binarisierung (auch: Thresholding oder Schwellwertfilterung) wird ein bestimmtes Merkmal ausgesucht, z.B. ob der (Grau-)Wert einen Schwellwert t übersteigt oder nicht.</p> <p>Für jedes Pixel $f_{in}(i, j)$ des zu filternden Bildes wird dann separat entschieden, ob es diesem Merkmal entspricht oder nicht.</p> <p>Man hat also einen Schwellenwert und für jeden Pixel wird untersucht, ob der Wert für diesen Pixel größer oder kleiner als der Schwellenwert ist. Basierend darauf wird dann eine Entscheidung getroffen, ob Entscheidung 1 oder 0 getroffen wird (Binäre</p>

<p>Wie kann ein geeigneter Schwellwert (automatisiert) bestimmt werden?</p>	<p>Entscheidung).</p> <p>Man will ja den Prozess automatisieren und nicht für jeden Pixel einen eigenen Wert bestimmen. → Man kann immer denselben Schwellwert nutzen. → Schwellwertsegmentierung auf Histogrammbasis</p>
<p>Erkläre Schwellwertsegmentierung auf Histogrammbasis</p>	<p>Anwendung von (Grauwert-)Histogrammen u.a. zur Festlegung von Schwellwerten bei schwellwertbasierter Segmentierung.</p> <p>Im Histogramm rechts sehen wir, dass zwei ausgeprägte Peaks im Histogramm zu sehen sind. Ein geeigneter Schwellwert kann zwischen den beiden Peaks liegen = X Achse bei 150.</p> <p>Auch der Blick auf's Summenhistogramm hilft → Stichwort Ableitung, z.B. Berechnung der Steigungen und von Wendepunkten Kurvenverlauf (später mehr dazu).</p> <div data-bbox="1177 533 1437 551" data-label="Text"> <p>Code: pythoncode/opencv-histogram.j</p> </div>    <div data-bbox="1358 958 1433 976" data-label="Text"> <p>PLRI</p> </div>
<p>Wie helfen Summenhistogramme bei der Schwellwertbestimmung?</p>	
<p>Welche Möglichkeiten gibt es noch Schwellenwerte zu berechnen? - 4</p>	<ol style="list-style-type: none"> 1. Nutzung des mittleren/medianen Intensitätswertes wenn erwartet wird, dass Vorder- und Hintergrund(-objekte) in etwa den gleichen Anteil im Bild haben. 2. Berücksichtigung eines bestimmten prozentualen Anteils des gewünschten Objektes im Bild und Festlegung des entsprechenden Schwellwertes anhand des Summenhistogramms.
<p>Erkläre das Prinzip von Otsu.</p>	<ol style="list-style-type: none"> 3. Otsu-Threshold : Bestimmung des Schwellwert wird unter folgenden Bedingungen bestimmt: <ul style="list-style-type: none"> • Minimierung der Streuung/Varianz in den Klassen. • Maximierung der Streuung/Varianz zwischen den Klassen. • Üblicherweise für zwei Klassen (Binarisierung) angewendet. Das Prinzip ist aber auch auf mehr Klassen übertragbar. <p>Otsu schaut sich die Streuung/ Varianz im Bild an. Er schaut sich vor allem die Klassen an – Bereiche, die später weiß und schwarz werden. Er minimiert Varianz innerhalb der Klassen und erhöht es zwischen den Klassen. → Objekte innerhalb einer Klasse sollen möglichst gleich sein. → Verschiedene Klassen sollen möglichst unterschiedlich sein.</p> <p>Grauwertstreuung/-varianz innerhalb der über den jeweiligen Schwellwert definierten Klassen/Regionen soll möglichst klein sein,</p>  <ul style="list-style-type: none"> • die Unterschiede zwischen den Klassen sollen dabei möglichst groß sein. • Es wird der Quotient zwischen den genannten Varianzen berechnet und der Schwellwert t so  <div data-bbox="1209 2074 1426 2092" data-label="Text"> <p>Der berechnete Schwellwert ist t=108.</p> </div>

<p>Was ist das Problem bei Otsu?</p> <p>Wie funktioniert adaptiver Threshold?</p> <p>Worauf muss man achten bei adaptiven Thresholds? Wie kann man es berücksichtigen?</p> <p>Was ist der Vorteil von adaptiven Thresholds?</p>	<p>gewählt, dass dieser Quotient maximal ist.</p> <p>Problem: ungleichmäßiger Ausleuchtung</p> <p>4. Adaptive Thresholds: Wenn feste Threshold nicht helfen.</p> <p>Die vorigen Beispiele zeigten, dass selbst Methoden wie der Otsu-Threshold bspw. bei ungleichmäßiger Ausleuchtung Probleme bekommen. → Nachbarschaft einbeziehen</p> <p>Im Gegensatz zur globalen Schwellwertfilterung, bei der ein fester Schwellenwert für das gesamte Bild verwendet wird, passt sich die adaptive Schwellwertfilterung an lokale Bildbedingungen an, um bessere Ergebnisse zu erzielen.</p> <p>Ein Beispiel einer solchen Filterung ist die adaptive Schwellwertfilterung, z.B. unter Einsatz dieser Strategien:</p> <p>Nutzung des Mittelwerts in einer Umgebung: Der Schwellenwert ist der Mittelwert der Grauwerte (abzüglich einer Konstante C).</p> <p>Hierbei wird für jedes Pixel ein Schwellenwert basierend auf dem Mittelwert der Grauwerte in seiner Umgebung berechnet. Eine Konstante C kann von diesem Mittelwert abgezogen werden, um die Empfindlichkeit des Filters anzupassen. Wenn der Grauwert des Pixels größer als der berechnete Schwellenwert ist, wird das Pixel als Teil des Objekts betrachtet, andernfalls als Hintergrund.</p> <ul style="list-style-type: none"> • Gauss: Der Schwellenwert ergibt sich aus der gaußgewichteten Summe der Nachbarschaftswerte (abzüglich einer Konstante C). <p>Hierbei wird für jedes Pixel ein Schwellenwert berechnet, indem die Nachbarschaftswerte des Pixels mit einer Gauss'schen Gewichtungsfunktion gewichtet und summiert werden. Auch hier kann eine Konstante C vom berechneten Schwellenwert abgezogen werden. Wenn der Grauwert des Pixels größer als der Schwellenwert ist, wird das Pixel als Teil des Objekts betrachtet, andernfalls als Hintergrund.</p> <ul style="list-style-type: none"> • Achtung: Rauschen im Bild vorher beheben! - Bsp Blur-Filter nutzen <p>Robuste Segmentierung:</p> <ul style="list-style-type: none"> - Gut , wenn Kontrast oder Beleuchtung variieren. <p>Reagiert flexibel auf Änderungen im Bild.</p> <div data-bbox="954 472 1414 517"> <p>Code: pythoncode/opencv-adaptive-threshold.py Adapt. Schwellwert mit Mittelwert</p> </div>  <div data-bbox="994 728 1257 750"> <p>Adapt. Schwellwert mit Gauss</p> </div> 
---	--

<p>Lokale Operationen:</p> <p>Nachbarschaften</p>	<div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;"> <p>4er-Nachbarschaft: $N_4[(x, y)]$</p> </div> <div style="text-align: center;"> <p>8er-Nachbarschaft: $N_8[(x, y)]$</p> </div> </div>
<p>Wie unterscheidet sich Nachbarschaften von Punktoperatoren?</p>	<p>Bis hierhin haben wir Punkte aus einem Bild genommen und diese Punkte in ein anderes Bild übertragen. Dabei haben wir Rechenoperatoren genutzt. → Diese Operatoren hingen nur vom Wert des Pixel ab. → Nachbarschaften beziehen auch die Grauwerte der Nachbar-Pixel mit ein.</p>
<p>Welche 2 Arten von Nachbarschaften gibt es in Bildern?</p>	<p>Die 4er Nachbarschaften: Ein Pixel hat mit 4 benachbarten Pixeln gemeinsame Kanten Die 8er-Nachbarschaft: Bei der 8er-Nachbarschaft werden die diagonalen Nachbarn ebenfalls mit betrachtet</p>
<p>Werden meistens gerade oder ungerade Anzahl an Pixeln bei Nachbarschaften beobachtet und warum?</p>	<p>Es können auch größere bzw. anders geformte Nachbarschaften definiert werden. • Bspw. Filterung in 1D: Anwendung nur in Zeilen- oder Spaltenrichtung. → Kann auch weiter erweitern, indem ich die Nachbarpixel von den Nachbarpixeln mit einbeziehe. Habe dann 15, nicht 9 Pixel, die ich beobachte. → Meistens ungerade Anzahl an Pixeln beobachtet. Grund: Damit der mittlere Punkt der Punkt ist, auf den man die Rechnung bezieht (siehe Bild). Gerade auch möglich?</p>
<p>Lokale Operatoren (Nachbar-Pixel mitberücksichtigt) 1</p>	
<p>Was sind Konvolutionsfilter? Wie werden Sie sonst genannt?</p>	<p>→ Falterungsfilter. → Filter, die lokal, nicht auf einzelne Pixel sondern auch Nachbarn, angewendet werden</p>
<p>Welche Filter gehören zu</p>	<p>→ Gruppe: lineare (lokale) Operatoren .</p>

den Konvolutionsfiltern?
- 4

In welche Gruppe an
lokalen Operatoren fallen
diese Filter?

Wodurch unterscheiden
sich diese 4 Filter?

Erkläre das Prinzip hinter
Konvolutionsfiltern

- **Mittelwertfilter,**
- **Gauss-Filter,**
- **Sobel- Operator**
- **Laplace-Operator**

Alle mathematisch identisch, aber **Sie unterscheiden sich durch ihren Kernel?**

Konvolutionsfilter sind lineare (lokale) Operatoren in der Bildverarbeitung, die zur Bearbeitung von Bildern verwendet werden. Sie basieren auf der Berechnung des Werts eines Zielpixels aus **dem Wert des entsprechenden Quellpixels sowie seiner Nachbarn**. Diese Filter nutzen eine Matrix, die als Konvolutionskern, Filterkern, Kernel oder **Faltungsmatrix** bezeichnet wird, um die Gewichtungsfaktoren für die Berechnung festzulegen.

Bei der Anwendung eines Konvolutionsfilters wird der **Filterkern über das Eingangsbild geschoben**, und für jeden Pixel wird die Faltung zwischen dem Filterkern und dem lokalen Bildbereich um den Pixel herum durchgeführt. Die Faltung besteht aus der elementweisen Multiplikation der Werte des Filterkerns mit den entsprechenden Werten im Bildbereich, gefolgt von einer Summation der Ergebnisse. Das Ergebnis dieser Faltung bildet den Wert des Zielpixels im Ausgabebild.

Angenommen, wir haben eine 3x3-Matrix als Eingangsbild:

Eingangsbild:

```
[ 1 2 3 ]
[ 4 5 6 ]
[ 7 8 9 ]
```

Und wir möchten die Faltung mit einem 2x2-Filterkern durchführen:

Filterkern:

```
[ 0.5 0.5 ]
[ 1.0 1.0 ]
```

Die Faltung besteht aus der elementweisen Multiplikation der Werte des Filterkerns mit den entsprechenden Werten im Bildbereich, gefolgt von einer Summation der Ergebnisse. Hier ist der Schritt-für-Schritt-Prozess:

1. Der Filterkern wird über das Eingangsbild geschoben, und für jeden Pixel wird die Multiplikation mit den entsprechenden Werten im Bildbereich durchgeführt.

Erster Pixel (oben links):

Eingangsbild:

```
[ 1 2 3 ]
[ 4 5 6 ]
```


[7 8 9]

Filterkern:

[0.5 0.5]

[1.0 1.0]

Multiplikation:

$(1 * 0.5) + (2 * 0.5) + (4 * 1.0) + (5 * 1.0) =$
12.0

2. Die Multiplikationsergebnisse für jeden Pixel werden summiert, um den Wert des Zielpixels zu berechnen.

Erster Pixel (oben links):

Summation:

12.0

3. Der Prozess wird für jeden Pixel wiederholt, indem der Filterkern über das Eingangsbild verschoben wird.

Zweiter Pixel (oben mittig):

Eingangsbild:

[1 2 3]

[4 5 6]

[7 8 9]

Filterkern:

[0.5 0.5]

[1.0 1.0]

Multiplikation:

$(1 * 0.5) + (2 * 0.5) + (3 * 1.0) + (4 * 1.0) =$
9.0

Summation:

12.0 9.0

4. Der Prozess wird für alle verbleibenden Pixel wiederholt, um die Ausgabematrix zu erhalten.

Ausgabematrix:

cssCopy code

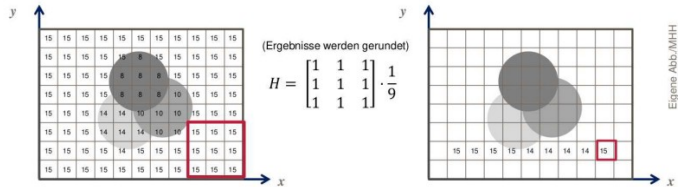
[12.0 9.0]

[24.0 21.0]

In diesem Beispiel haben wir die Faltung mit einem 2x2-Filterkern durchgeführt. Jeder Pixel des Ausgabebildes wurde durch Multiplikation der entsprechenden

<p>Was ist der Unterschied zwischen linearen und nicht linearen Operatoren?</p> <p>Nenne Beispiele für solche Operatoren und in welche Gruppe gehören Sie? - 3</p> <p>→ Wann werden diese Operatoren verwendet?</p>	<p>Werte des Filterkerns mit den Werten im Bildbereich berechnet, gefolgt von einer Summation der Ergebnisse. Die resultierende Ausgabematrix zeigt die gefalteten Werte für jeden Pixel.</p> <p>Die Gewichtungsfaktoren im Konvolutionskern bestimmen, wie stark die umliegenden Pixel den Wert des Zielpixels beeinflussen. Verschiedene Filterkerne erzeugen unterschiedliche Effekte auf das Bild.</p> <p>Nichtlineare Operatoren sind nicht über Konvolution / Faltung berechenbar. → Man kann Sie nicht zurückberechnen auf den Punkt.</p> <p>→ Gruppe: Morphologische Operatoren</p> <p>Morphologische Operatoren sind mathematische Operationen, die in der Bildverarbeitung zur Veränderung und Analyse der Gestalt von Objekten verwendet werden. Sie werden häufig auf Binärbildern angewendet, bei denen jedes Pixel entweder den Wert 0 (Hintergrund) oder den Wert 1 (Objekt) hat. Morphologische Operatoren ermöglichen es, Merkmale wie Form, Größe und Textur von Objekten zu verändern oder zu extrahieren.</p> <p>→ Opening, → Closing</p> <p>→ Rangordnungsfiler: Medianfilter, Erosion, Dilatation</p>
<p>Ist die Zeilen und Spaltenzahl gerade oder ungerade bei lokalen Filterungen? Welche Form hat der Kern?</p> <p>Wie funktioniert der Kern bei der lokalen Filterung?</p>	<ul style="list-style-type: none"> Jeder Bildpunkt des Ausgabebildes $f_{out}(i, j)$ wird durch Multiplikation der Pixel des Eingabebildes f_{in} in in der definierten "Nachbarschaft" mit einem sogenannten Filterkern $H = [h(u, v)]$ berechnet: $f_{out}(i, j) = \sum_{u=0}^M \sum_{v=0}^L h(u, v) \cdot f_{in}(i - k_{off,x} - u, j - k_{off,y} - v)$ <p> M, L : Größe des Filterkerns in x- bzw. y-Richtung i, j : Laufindizes des aktuellen Pixels (x- bzw. y-Richtung) u, v : Laufindizes in der Nachbarschaft des aktuellen Pixels $k_{off,x} = \frac{M-1}{2}, \quad k_{off,y} = \frac{L-1}{2},$ </p> <p>Meist quadratischer Filterkern mit ungerader Zeilen- und Spaltenzahl (z.B. 8er-Nachbarschaft)</p> <p>Filterkern H wird über das Bild gelegt.</p> <ul style="list-style-type: none"> Multiplikation der einzelnen Werte im Filterkern mit den Pixelwerten an der entsprechenden Stelle in der Nachbarschaft. Addition all dieser Einzelergebnisse, ggf. Normierung auf den ursprünglichen Grauwertebereich (falls nicht im Filterkern enthalten). Verschieben des Filterkerns an die nächste Position im Bild

<p>Wie nennt man diese Operatoren?</p>	<ul style="list-style-type: none"> • (Bei Randpixeln entsprechende Strategie anwenden, z.B. Randverdopplung.) • (Vorsicht beim Programmieren, falls das Koordinatensystem der verwendeten Bildverarbeitungsbibliothek z.B. oben links statt unten links anfängt zu zählen – evtl. muss dann der Filterkern entsprechend gespiegelt / gedreht werden.) <p>Diese Operation heißt auch Faltung des Bildes mit dem Kern H.</p>
<p>Lokale Operatoren: Konvolutionsfilter als Hochpass, Tiefpass</p> <p>Wozu nutzt man Hoch und Tiefpassfilter? Was entfernt Hochpassfilter und was entfernt Tiefpassfilter?</p> <p>Was macht ein Tiefpassfilter mit dem Bild? Wie verändert sich das Bild dadurch?</p>	<div data-bbox="603 443 1401 824"> <p>Originalbild mit horizontalem Linienprofil</p> <p>Originalbild mit Linienprofil 45 Grad</p> <p>1D-Grauwertprofil (entlang horizontaler Linie)</p> <p>1D-Grauwertprofil (entlang 45 Grad-Linie)</p> <p>Code: <code>pythoncode/linienprofile.py</code> (Folieninhalt) bzw. <code>pythoncode/linienprofile-interactive.py</code> zur interaktiven Nutzung (Ende mit Taste c; Definition der gewünschten Linie durch Ziehen bei gedrückter linker Maustaste.)</p> </div> <p>Idee: Hochfrequente Signale loswerden, indem man Tiefpassfilter nutzt. Hochpassfilter entfernt niedrige Frequenzen.</p> <p>→ Man kann bestimmte Merkmale so hervorheben oder entfernen.</p> <p>Die starken Übergänge/Kanten werden geglättet. Man hat einen glatteren Übergang zwischen dunkel hin zum hellen. Sprünge werden so entfernt. Es wirkt verwaschener. Man hat dadurch einen Blureffekt.</p> <p>Nutzt man Hochpassfilter und entfernt tiefere Frequenzen, werden die Hintergrundelemente verwaschen, weil gleichförmige Bildelemente noch ähnlicher werden. Die Hochfrequenten Elemente stechen so noch stärker raus.</p> <ul style="list-style-type: none"> • Bestimmung über die erste bzw. zweite Ableitung des Grauwertprofils $f(x)$ (bzw. der Bilddaten in 2D, also $f(x, y)$) möglich. • Erste Ableitung $f'(x)$: Steigung der (Tangente an die) Funktion $f(x)$ (einfache Näherung durch Differenz aufeinander folgender Punkte möglich). <ul style="list-style-type: none"> • $f'(x) > 0$: f streng monoton steigend • $f'(x) = 0$: f lokale Extremstelle von f → Helligkeit ist konstant • $f'(x) < 0$: f streng monoton fallend • Zweite Ableitung $f''(x)$: Beschreibt die Krümmung von f. Gibt an, ob f' steigt oder abfällt. <ul style="list-style-type: none"> • $f''(x) > 0$: linksgekrümmt ($f(x)$ konkav); $f'(x) = 0$ und $f''(x) > 0$: lokales Maximum von $f(x)$ • $f''(x) < 0$: rechtsgekrümmt ($f(x)$ konvex); $f'(x) = 0$ und $f''(x) < 0$: lokales Minimum von $f(x)$ <p>(Wendepunkt, wenn $f''(x) = 0$ und $f'''(x) \neq 0$)</p> <div data-bbox="699 1608 1265 1989"> <p> $f(x) = 5x^3 + 4x^2 - 9x + 10$ (z.B. Grauwertprofil) $f'(x) = 15x^2 + 8x - 9$ (1. Ableitung) $f''(x) = 30x + 2$ (2. Ableitung) </p> </div>


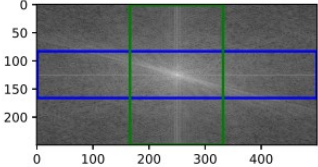
<p>Lokale Operatoren: Konvolutionsfilter – Mittelwertfilter (Tiefpass) I</p> <p>Was für ein Filtertyp ist der Meanfilter?</p> <p>Wie beeinflusst er Rauschen und Kanten?</p> <p>Welchen Einfluss hat die Filterkerngröße auf den Effekt?</p> <p>Erkläre die Idee und das Prinzip hinter dem Filter.</p>	<p>Der Mittelwertfilter ist ein typischer Tiefpassfilter</p> <ul style="list-style-type: none"> • Rauschen (= “hochfrequente” Störungen) wird reduziert, • Kanten (ebenfalls “hochfrequent”) werden teils “verwaschen”. <p>Idee: Hochfrequente Störungen entfernen und führt einen Blureffekt ein.</p> <p><u>Erklärung:</u> 3x3 Filterkern 1 als Wert (vereinfachtes Modell) genutzt. Dieser Filterkern wird über jeden Punkt des Bildes gelegt und man schiebt den Filterkern immer um 1 Punkt weiter. Die 1 bedeutet, dass man den Punkt mit dem Wert 1 multipliziert. Man summiert die Werte in der 3x3 Matrix dann zusammen. Am Ende dividiert man durch die Anzahl der Pixel im Filterkern (hier 9, da 3x3 Kern). Man bildet also den Mittelwert. Jeder Punkt wird durch den Durchschnitt der umgebenden Pixel und des betrachteten Pixels ersetzt. Das verändert <u>nicht</u> die Gesamthelligkeit des Bildes. Man bildet so die Mittelwerte für jeden Punkt verglichen mit seinen Nachbarn im Bild.</p> <p>Das heißt, Pixel, dessen Nachbar-Pixel dunklere/niedrigere Werte hatten, kriegen einen geringeren Mittelwerte als Pixel in hellen Bildbereichen.</p> <p>In unserem Beispiel sieht man, dass Werte, die früher 15 waren, jetzt den Wert 14 haben. Ihre Nachbarn senken also die Werte „näher“ an ihren eigenen, geringeren Wert. Die Farbwerte nähern sich an und Übergänge werden so verwaschen – sind nicht mehr so ausgeprägt.</p> <p>Wenn man den Filterkern erhöht, haben weiter entfernte Nachbarn einen größeren Einfluss auf den Wert - Je größer der Filterkern H, desto stärker der Effekt.</p> 
<p>Lokale Operatoren: Konvolutionsfilter – Gauss-Filter (Tiefpass)</p> <p>Welcher Filtertyp ist der Gaussfilter?</p> <p>Wie funktioniert der Filter?</p> <p>Wie verhält es sich mit der Betonung/ Absteigen der Zahlenwerte?</p> <p>Was ist der Vorteil von Gauss gegenüber</p>	<p>Filtertyp: Tiefpass</p> <p>Idee: All diese Filter unterscheiden sich durch ihre Filterkerne. Der Gaussfilter legt eine „Gausskurve“ über die Matrix. Das heißt, anstelle von nur 1 hat man in der Mitte einen großen Wert, der zu den Seiten hin abnimmt.</p> <ul style="list-style-type: none"> → Starke Betonung auf den mittleren Wert. → Schwächere Betonung auf direkte Nachbarn → Betonung nimmt ab je weiter Nachbar entfernt ist vom Mittleren Pixel. <p>• Einträge des Gauss-Filters $H = [h(x)]$ (1D) bzw. $H = [h(x, y)]$ (2D) werden über die Gauss-Funktion konstruiert:</p> <ul style="list-style-type: none"> • eindimensional für x (y analog): $h(x) = \frac{1}{\sigma\sqrt{2\pi}} \cdot e^{-\frac{x^2}{2\sigma^2}}$ • zweidimensional: $h(x, y) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2}{2\sigma^2}}$, z.B. $H = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$ <p>Randnotiz:</p> <ul style="list-style-type: none"> • Gauss-Filter ist separierbar, d.h. $h(x, y) = h(x) \cdot h(y)$ (Reduktion von Rechenzeit) möglich, d.h. $h(x, y) = \frac{1}{4} \cdot \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} * \frac{1}{4} \cdot \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$ <p>• Isotrop, d.h. richtungsunabhängig. Ich kann Matrix zerlegen. Man zerlegt es in einen Spalten und Zeilenvektor. Siehe Formeln. Man kann Kernel auch kombinieren.</p> <p>→ Reduziert Rechenzeit.</p> <p>• Ebenfalls ein Tiefpassfilter, erhält Kanten besser.</p>

Mittelwertfilter?	<p>→ Bild bleibt schärfer als Mittelwertfilter, aber man hat noch die Glättungseffekte.</p> <p>→ Rauschen wird entfernt, aber Struktur bleibt besser erhalten als Mittelwertfilter.</p>
Der Filter ist Isotrop. Was heißt das? Was ist der Vorteil?	
Lokale Operatoren: Konvolutionsfilter – Differenzenoperator (Hochpass)	<p>• Hochpassfilter heben Kanten bzw. “Spitzen” hervor</p> <p>• Es wird die erste Ableitung f' der Bildfunktion f approximiert (d.h. die Steigung berechnet, z.B. in x- oder y-Richtung).</p> <p>• Beispiel Differenzenoperator</p> $\frac{\partial}{\partial x} f(x, y) \approx f(x, y) - f(x - 1, y)$ $\rightarrow H_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \text{ und}$ $\frac{\partial}{\partial y} f(x, y) \approx f(x, y) - f(x, y - 1)$ $\rightarrow H_y = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -1 & 0 \end{bmatrix}$
Welcher Filtertyp ist der Differenzoperator?	<p>Filtertyp: Hochpass</p>
Was heben diese Filtertypen hervor?	<p>Idee: Es bildet eine Differenz, indem es einen Wert nimmt, den ich bereits benutze und zieht es vom neuen Wert ab. Hier hat man 1 in der Mitte und zieht 1 vom Nachbarn ab.</p> <p>→ Berechne so die Änderungsrate/ Steigung: Wie groß ist der Sprung zum Nachbarpixel?</p>
Was ist die Idee dahinter?	<p>Ausrichtung des Kernels wichtig (siehe Bild der Matrizen):</p> <p>Horizontal: Finde vertikale Kanten, die sich in horizontaler Richtung als Sprung herausbilden.</p>
Welchen Einfluss hat die Ausrichtung der Matrizen?	<p>Vertikal Anordnung des Filters: Finde horizontale Kanten.</p> <p>Grund: Um horizontale Kanten zu erkennen, benötigen wir einen Filter, der empfindlich auf Änderungen in vertikaler Richtung ist. Durch die vertikale Anordnung des Filters wird der Filter über vertikale Bereiche des Bildes verschoben. Wenn eine horizontale Kante vorhanden ist, werden die Pixelwerte oberhalb und unterhalb der Kante unterschiedlich sein. Durch die Subtraktion des benachbarten Pixels von der Mitte des Filters wird die Differenz zwischen den Pixelwerten berechnet. Da horizontale Kanten zu einem deutlichen Sprung in vertikaler Richtung führen, erzeugt die Subtraktion einen größeren Wert, der auf das Vorhandensein einer horizontalen Kante hinweist. Umgekehrt gilt das auch für vertikale Kanten und Horizontale Ausrichtung der Matrix</p>
Lokale Operatoren: Konvolutionsfilter – Roberts-Cross, Sobel (Hochpass)	<p>Roberts-Cross</p> $H_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix} \text{ und}$ $H_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$ <p>Sobel [13]</p> $H_1 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}, H_2 = \begin{bmatrix} 0 & -1 & -2 \\ 1 & 0 & -1 \\ 2 & 1 & 0 \end{bmatrix},$ $H_3 = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \dots$ <p>vs.</p>
Welchem Filtertyp gehören Roberts-Cross und Sobel an?	<p>Filtertyp: Hochpass.</p>
Was ist die Idee hinter Roberstcross? Was kann er besonders gut finden im Bild? Was ist der Nachteil von Differenzialoperator und RoberstCross	<p>Idee hinter Roberts-Cross: Wie Differenzoperator, aber Differenz wird Diagonal gebildet.</p> <p>→ Findet Diagonal angeordnete Übergänge im Bild besser.</p> <p>• Nachteil: Differenz- und Roberts-Cross-Operator sind rauschanfällig.</p> <p>• Gibt es Filter, die gleichzeitig mitteln und ableiten? → Sobel.</p>
Was ist die Idee hinter Sobel?	<p>Idee hinter Sobel: Man nutzt hier zwei Matrizen statt eine. Eine Matrize für die Vertikale (Nullen gehen von rechts nach links) und eine für die Horizontale (Nullen gehen von oben nach unten). Ansonsten sind die Werte wie bei einem versetzten Gaussfilter angeordnet. Mit anderen Worten: Der Wert in der Mitte ist der größte und sinkt mit dem Abstand vom Mittleren Punkt.</p> <p>→ Berücksichtigt die Steigung der Nachbarn besser.</p> <p>Durch Rotation des Kerns weitere 5 Varianten (mehr “Richtungen”).</p> <p>• Leitet über 2 Pixel ab, “mittelt” über zweimal 3 Pixel.</p> <p>• Ergebnis: “stabilere” Kanten.</p>
	<p>Oft mit Gauss zusammen verwendet</p> <p>→ Gauss (Tiefpass) reduziert Rauschen und Sobel (Hochpass) hebt Kanten hervor.</p>

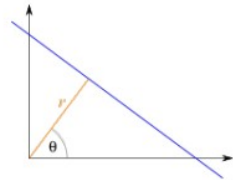
<p>Lokale Operatoren: Konvolutionsfilter – Laplace-Filter (Hochpass)</p> <p>Worauf basiert Laplace? Welche Punkte werden gefunden? Wie hängt Laplace und die 2 Ableitung zusammen?</p> <p>Erkläre das Prinzip</p> <p>Welche Richtungen werden berücksichtigt?</p> <p>Was ist der große Vorteil von Laplace gegenüber Cross, Sobel (anderen Hochpassfiltern)?</p> <p>Was nutzt OpenCV standard gemäß?</p> <p>Wann kann eine Kante nicht also solche erkannt werden?</p> <p>Bei welcher Bildgebungsmethode würde man nicht LaPlace nutzen?</p>	<ul style="list-style-type: none"> • Basiert auf der zweiten Ableitung $f''(x, a)$ der Bildfunktion $f(x, y)$. • Findet Punkte, an denen der "Grauwertverlauf" eine mögl. große Krümmung aufweist. <p>→ Der Kernel ist hier eine Annäherung an die 2. Ableitung. Die Werte werden multipliziert mit dem Kernel und dann addiert. Das Ergebnis ist die Annäherung an die 2. Ableitung.</p> <ul style="list-style-type: none"> • Laplace mit Hsum (Grauwerte gespreizt, invertiert): OpenCV nutzt für den Laplace-Operator intern den Gradienten $G = q \cdot H_2x + H_2y$ aus H_x und H_y des Sobeloperators (Kantenrichtung: $\theta = \arctan(H_x, H_y)$), <ul style="list-style-type: none"> • Formal: $\frac{\partial}{\partial x} f(x, y) + \frac{\partial}{\partial y} f(x, y)$ • Filterkerne $H_{orig} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$, $H_{45} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & -4 & 0 \\ 1 & 0 & 1 \end{bmatrix}$, $H_{sum} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$ mit $H_{sum} = H_{orig} + H_{45}$ (H_{orig}: original, H_{45}: diagonal, Beispiele). <p>Berücksichtigt alle Richtungen. In alle richtungen werden Differenzen gebildet und die Differenzen werden aufsummiert. Die Funktionsweise des Laplace-Kernels beruht auf der Berechnung der Differenz zwischen dem aktuellen Pixelwert und dem Durchschnitt der umliegenden Pixelwerte. Wenn ein Pixel von seinen Nachbarn stark abweicht, deutet dies auf eine Kante oder eine schnelle Helligkeitsänderung hin. Der negative Wert (-4) in der Mitte des Kernels verstärkt diesen Unterschied und trägt zur Erkennung von Kanten bei.</p> <p>Kann Kerne miteinander kombinieren und so zum Gesamtergebnis kommen.</p> <p>+ Berücksichtigt alle Richtungen und damit die Differenzen in alle Richtungen. Cross und Sobel können nur Horizontale oder Vertikale oder nur die Diagonale berücksichtigen. Hier kann man alle Richtungen berücksichtigen. Man kann es bei anderen nur machen, wenn man Sie kombiniert.</p> <p>OpenCV nutzt den Sobel in horizontaler und vertikaler und berechnet dann erst daraus den Gradienten.</p> <p>Wenn der Übergang von weiß zu schwarz größer ist als der Filterkern, wird die Form nicht als Kante erkannt?</p> <p>Bei Ultraschallbildern: Das Rauschen ist dominant. LaPlace würde mehr oder weniger nur das Rauschen segmentieren.</p>
---	--

<p>Lokale Operatoren: Konvolutionsfilter – Keep in Mind</p>	<ul style="list-style-type: none">• Linearität: Jeder Pixelwert durch die linear gewichtete Summe der Quellpixel in der gegebenen Nachbarschaft berechnet• Additivität: Siehe Beispiel Laplace aus HOrig + H45• Assoziativität / Verknüpfung mehrerer Operationen: Anwendung von $(H1 + H2) + (H3 + H4)$ entspricht $H1 + (H2 + H3) + H4$• Kommutativität: Operationen können vertauscht werden, ohne dass sich das Ergebnis verändert, d.h. $H1 + H2 + H3 + H4$ entspricht $H3 + H1 + H4 + H2$ <p>Bei Hochpass/Tiefpass-Filtern wird über Addition und Multiplikation neue Farbwerte für das Bild bestimmt.</p>									
<p>Lokale Operatoren: Morphologische Operatoren I</p> <p>Was nutzt man bei morphologischen Operatoren?</p> <p>Was sind die „Filterkerne“ hier?</p> <p>Auf welche Bilder wird es angewendet? Wofür werden diese Operatoren oft benutzt?</p>	<ul style="list-style-type: none">• Morphologisch heißt hier: die (äußere) Gestalt/Form betreffend. → Man nutzt morphologische Formen für verschiedene Zwecke. → Hier sind die „Filterkerne“ Strukturelemente.• Häufig auf bereits extrahierten “Segmenten” angewendet (z.B. auf Binärbildern). Einsatz auf anderen (Grauwert- oder Farb-) Bilddaten ist aber nicht ausgeschlossen.• Ziele dieser Operatoren:<ul style="list-style-type: none">• z.B. Beseitigung von Störungen (nach einer Segmentierung), bspw. von im Binärbild aufgrund von Rauschen verbliebenen “Punkten”.• Unterstützung bei der Berechnung von Formmerkmalen oder der Suche nach bestimmten Formen.• Die sog. Strukturelemente von morphologischen Operationen entsprechen in etwa dem Faltungskern bei Konvolutionsfiltern (linearen Filtern).• Mit genau angepassten Strukturelementen (nicht unbedingt konv. Form, wie etwa 3×3, ...) können genau definierte Formveränderungen vorgenommen werden.• Beispiel: Scan und Nachbearbeitung von einem Ausdruck mit horizontalen “Streifen” als Basis einer Schrifterkennung mit OCR.• Siehe Beispiele von Rangordnungsfiltren (u.a. Erosion, Dilatation) bzw. weiteren morphologischen Operationen (Opening, Closing, Skelettierung) auf den folgenden Folien. Nicht-linear.									
<p>Lokale Operatoren: Rangordnungsfiltren</p> <p>Welche 3 Rangordnungsfiltren gibt es und wie funktionieren Sie?</p>	<ul style="list-style-type: none">• In einer lokalen Nachbarschaft (analog zu den Konvolutionsfiltern, z.B. 3×3, 5×5 oder anders geformte Struktur) wird die sortierte Reihenfolge der enthaltenen (Grau- bzw. Intensitäts-)Werte ermittelt.• Medianfilter: Jeder Wert im Bild wird durch den Wert ersetzt, der in der sortierten Liste der definierten Nachbarschaftspixel an der mittleren Position steht (d.h. der 5. Wert bei 3×3, der 13. bei 5×5). <p>Schaue mir eine Nachbarschaft an (3×3, 5×5 oder größer). Bringe die Intensitätswerte in eine Reihe und bestimme den Median über den Medianfilter.</p> <ul style="list-style-type: none">• Erosions- und Dilationsfilter: Nach der Sortierung wird der aktuelle Wert durch den minimalen (Erosion) oder maximalen (Dilatation) Wert der sortierten Liste ersetzt. Erosion und Dilatation werden oft auf bereits binarisierten (schwellwertgefilterten) Bildern eingesetzt.• Opening bzw. Closing als Kombination von Erosion und Dilatation → weiter hinten im Foliensatz. <div><p>Das Diagramm illustriert die Wirkung von drei Filtern auf eine 3×3 Nachbarschaft. Die ursprüngliche Nachbarschaft ist:</p><table><tr><td>6</td><td>11</td><td>12</td></tr><tr><td>18</td><td>22</td><td>17</td></tr><tr><td>19</td><td>21</td><td>24</td></tr></table><p>Die sortierte Reihenfolge der Werte ist: 6, 11, 12, 17, 18, 19, 21, 22, 24.</p><ul style="list-style-type: none">Erosion: Der minimale Wert (6) wird ausgewählt. Text: „Helle Bereiche schrumpfen, dunkle wachsen“.Median: Der mittlere Wert (18) wird ausgewählt.Dilatation: Der maximale Wert (24) wird ausgewählt. Text: „Helle Bereiche wachsen, dunkle schrumpfen“.</div>	6	11	12	18	22	17	19	21	24
6	11	12								
18	22	17								
19	21	24								

<p>Welchen Wert nutzt Dilation und Erosion?</p> <p>Ist es umkehrbar?</p> <p>Welchen Einfluss haben Dilation und Erosion auf das Bild?</p> <p>Auf welcher Art von Bild werden die Filter oft angewendet?</p> <p>Wo haben diese Operatoren eine gute Anwendung?</p>	<p>Erosion: kleinster Werte Dilation: größter Wert.</p> <p>→ Es ist nicht umkehrbar.</p> <p>Erosion: Helle Bereiche schrumpfen, dunkle wachsen – Wenn ich immer den kleinsten Wert nehme, dann nehme ich damit Werte, die dunkler sind. → Gut, wenn ich helle Bereiche im Bild loswerden will.</p> <p>Dilation: Helle Bereiche wachsen, dunkle sinken. Gut, wenn ich helle Bereiche im Bild steigern will.</p> <p>Anwendung oft auf binärisierten Bildern.</p> <p>Kann so gut Kanten erkennen. Ich binärisiere ein Bild. Nutze Erosion, um die störenden Elemente loszuwerden und kann dann die gefundenen Kanten über das Originalbild legen.</p>
<p>Lokale Operatoren: Morphologische Operationen – Opening, Closing I</p> <p>Was ist die Idee hinter Opening und Closing?</p> <p>Was ist das Ziel von Closing und Opening? - wann würde ich das eine und wann das andere anwenden?</p> <p>Wie werden Dilation und Erosion bei Opening und Closing verkettet (reihenfolge). Welchen Zweck haben Dilation und Erosion bei beiden?</p> <p>Was ist (zumindest bei med. Bilddaten) häufig als "Vordergrund" (erwünschte Struktur) definiert, was als Hintergrund (zu vernachlässigen)?</p> <p>Was passiert mit den dunklen und hellen Bereichen beim Opening/Closing</p>	<p>Opening (Öffnung):</p> <ul style="list-style-type: none"> • Verkettung von Erosion, gefolgt von Dilation (mit einem am Ankerpunkt gespiegelten Strukturelement S') • Ziel der Erosion: Entferne alle (Teil-) Strukturen, die kleiner als das Strukturelement sind. • Ziel der Dilation: Stelle ursprüngliche Objektgröße/Form wieder her, allerdings ohne die entfernten Teile. <p>• Closing (Schließung):</p> <ul style="list-style-type: none"> • Verkettung von Dilation, gefolgt von Erosion (mit einem am Ankerpunkt gespiegelten Strukturelement S') • Ziel der Dilation: Schließe evtl. in den Strukturen vorhandene Lücken, die kleiner als das Strukturelement sind. • Ziel der Erosion: Stelle ursprüngliche Objektgröße/Form wieder her, allerdings ohne die Lücken. <p>- Weiß Vordergrund</p> <p>• Berücksichtigt wurden in diesem Fall die hellen Pixel, wie in der medizinischen Bildverarbeitung oft üblich, auch wenn für das Auge eigentlich eher die dunklen Bildpunkte für die Schere stehen (Vordergrund) .</p> <p>• Daher wachsen beim Opening scheinbar auch die dunklen Bereiche (Grauwert 0) zusammen, die hellen (max. Grauwert) werden "geöffnet", beim Closing kommt es zu einem "Zusammenwachsen" der dunklen Bereiche.</p> <p>• Je nach Anwendungszweck auf die Definition von Vorder- und Hintergrund achten bei der Anwendung solcher morphologischen Filter!</p> <p>Bei normalen Bildern aus dem Alltag ist Vordergrund dunkler als Hintergrund. Die Filter sind daran angepasst. Man muss bei medizinischen Bildern also aufpassen, weil es dort umgekehrt ist.</p>

<p>Lokale Operatoren: Verkettung morph. Operat. zur Skelettierung</p> <p>Wo hat Verkettung lokaler Operatoren eine gute Anwendung? (Optionale Folie).</p>	<p>Man kann die Operatoren kombinieren, um verschiedene Aufgaben zu erfüllen. → Bsp Skelletierung → Form von Buchstaben auf Mittellinie reduzieren, um Buchstaben zu erkennen.</p> <ul style="list-style-type: none"> • Die bisher vorgestellten Filter/Operatoren können insbesondere für die Vorverarbeitung hilfreich sein. Sie beeinflussen die morphologische Gestalt (Formgebung) meist nur gering. • Was aber, wenn es darum geht, bestimmte Eigenschaften von Strukturen (z.B. Ausrichtung eines Knochens im Röntgenbild/CT) zu bestimmen? • Umfangreiche Berechnung auf (evtl. binarisierter) Originalform? • Oder macht es Sinn, eine Reduktion der abgebildeten Strukturen vorzunehmen?- • Die sogenannte Skelettierung kann hier helfen: • Mehrschrittiges Verfahren zur Datenreduktion. • Verdünnt die dargestellten Strukturen auf eine 1 Pixel breite "Mittellinie". • Wird häufig u.a. im Kontext Schrift-, Fingerabdruck-, Gestenerkennung etc. eingesetzt. • Behält wichtige topologische Merkmale (z.B. Zusammenhang, Löcher) bei. <p>Das Eingabebild sollte binarisiert vorliegen. Ein möglicher Algorithmus:</p> <ol style="list-style-type: none"> 1 Starte mit einem leeren Skeleton (mit identischen Abmessungen wie Originalbild). 2 Wende Opening2 auf das Eingabebild <code>img_in</code> an (→ <code>img_open</code>) an. 3 Ziehe <code>img_open</code> (Ergebnis aus Schritt 2) von <code>img_in</code> ab (→ <code>img_temp</code>). 4 Wende Erosion auf <code>img_in</code> an (→ <code>img_erode</code>). Verfeinere das Skelettbild <code>img_skeleton</code> durch bitweise ODER-Verknüpfung der bestehenden Version von <code>img_skeleton</code> und <code>img_temp</code>. Weise für den nächsten Durchlauf <code>img_in</code> das Ergebnis der Erosion <code>img_erode</code> zu. 5 Starte neuen Durchlauf (das erodierte Bild wird zum Eingabebild) oder breche ab, falls das erodierte Bild leer ist. Das berechnete Skelett ist minimal, d.h. es können keine weiteren Punkte entfernt werden, ohne Zusammenhangeigenschaften zu verletzen
<p>Globale Operatoren am Beispiel der Fouriertransformation</p> <p>Was nutzt man um Bilder von Ortsraum in Frequenzraum zu überführen?</p> <p>Was ist IFFT?</p>	<p>Um Bilder von Ortsraum in Frequenzraum zu überführen nutzt man die FastFourierTransformation.</p> <p>→ Frequenzraum stellt Bildelemente als Frequenz dar. → Verlustfrei und umkehrbar.</p> <p>Rekapitulation: Bildverständnis "konventionell" (Ortsraum)</p> <ul style="list-style-type: none"> • Darstellung der Bilddaten wie aufgenommen. • Pixel entsprechen im digitalen (diskretisierten) Bild den Intensitätswerten an der jeweiligen Ortskoordinate. • Für "menschliche" Auswertung optimal, aber Computer können evtl. mit anderen Darstellungen effizienter umgehen. • Im Gegensatz dazu: Anwendung des Frequenzraums (Fourierraums) • Überführung der Bilddaten in eine Frequenzdarstellung, z.B. mit der Fast Fourier Transformation (FFT). • Ergebnis der FFT sind komplexe Zahlen, ein Pixel repräsentiert hier die Intensität einer Frequenz. • Die FFT ist ohne Verlust durch die inverse FFT (IFFT) umkehrbar und damit das (gefilterte) Bild rekonstruierbar. <div style="display: flex; justify-content: space-around; align-items: flex-end;"> <div style="text-align: center;"> <p>Originalbild</p>  <p><small>rcode/opencv-fft-fft-fft.py</small></p> </div> <div style="text-align: center;"> <p>Spektrum (Intensität)</p>  </div> </div>

<p>Wie sind die Frequenzen im Frequenzraum aufgetragen?</p> <p>Wann ist Filterung im Frequenzraum effizienter als Ortsraum</p>	<ul style="list-style-type: none"> • Grün: niedrige x-Frequenzen; blau: niedrige y-Frequenzen; Zentrum: Mittelwert (Frequenz=0) • Hochpass- und Tiefpassfilterung bspw. durch das Eliminieren der entspr. Bereiche und IFFT. • Faltung im Ortsraum = Multiplikation im Fourierraum. • Multiplikation der Fouriertransformierten des Bildes mit der Fouriertransformierten des Filterkerns, dann IFFT. • Filterung im Frequenzraum kann insbes. bei sehr großen Filterkernen effizienter sein als im Ortsraum! → Platzsparend. • Im Beispiel: rel. ausgeprägter heller Bereich bei den niedrigen Frequenzen, bedingt durch den (großflächig) unscharfen Hintergrund. • FFT als Hochpassfilter • Fouriertransformierte bspw. so maskieren, dass der Bereich niedriger Frequenzen im Zentrum eliminiert wird (z.B. ± 15 in x- und y-Richtung, ausgehend vom Zentrum). • Inverse FFT zeigt dann ein Hochpass-gefiltertes Bild (Kantenfilter).
<p>Defintion “Segmentierung”</p> <p>Wozu ist Segmentierung ist da?</p> <p>Welche 2 Arten von Segmentierungen gibt es?</p> <p>Welche Verfahren gibt es und erkläre sie? -6</p>	<ul style="list-style-type: none"> • Segmentierung = Zerlegung von Bilddaten in Teilbereiche (z.B. Strukturen, denen eine Bedeutung zugewiesen werden kann). Siehe auch semiotische Triade in den Folien zu Teil I. → Man will bestimmte Strukturen erkennen im Bild und aufteilen. • Ziel: Übergang von der numerischen (grauwert-)matrixbasierten Darstellung zu einer symbolischen Repräsentation (durch Punkte, Flächen, Linien oder andere geometrische Objekte). • Es findet eine Datenreduktion statt: (Grauwert-) Bildmatrix → symbolische Beschreibung. • Datenbasierte Segmentierung • Modellbasierte Segmentierung • Es gilt: • Jeder Bildpunkt/Pixel wird einer bestimmten Region zugeordnet (z.B. Vorder- oder Hintergrund, definiertes Objekt, ...). • Ein Bildpunkt/Pixel kann nur einer bestimmten Region zugeordnet sein (überdeckungsfrei) • Für jede Region gilt ein Homogenitätskriterium, das deren Eigenschaften beschreibt (warum gehört ein Bildpunkt zu einer bestimmten Region und nicht zu einer anderen). • Ein “Segment” bildet i.A. ein zusammenhängendes Gebiet. • Pixelorientierte Verfahren, d.h. Histogramm- bzw. bildbasierte Segmentierung: • Zugehörigkeit zu einem Segment wird anhand des Histogramms oder des Bildes festgelegt. • Beispiel: Schwellwertverfahren, z.B. mit Otsu-Threshold • Regionen- oder kantenbasierte Segmentierung: • Segmente werden durch Objektgrenzen (Kanten) bzw. ihre Struktur (“Innenleben”) bestimmt. • Beispiel kantenbasiert: Sobel- oder Laplace-Operator (+ evtl. Verfahren zur Kantenvervollständigung) + Gradientensuche, Wasserscheidentransformation, Snakes (active contours). • Beispiel regionenorientiert: Verfahren betrachten die Punktmengen als Gesamtheit und versuchen zusammenhängende Regionen zu finden, z.B: mittels Region Growing, Region Splitting, • Cluster-Verfahren, bspw. mit k-Means (iterativer Algorithmus) 1 Wahl von k Startpunkten im Bild (Initialisierung der Cluster, hat entscheidenden Einfluss). 2 Zuordnung von Datenpunkten zu den jeweiligen Clustern auf Basis des Abstands, d.h. weise jeden neuen Punkt dem Cluster zu, bei dem sich die Varianz (z.B. auf

	<p>Basis der Intensitätswerte) durch das Zufügen am wenigsten erhöht. 3 Neuberechnung der Cluster-Zentren. 4 Wiederholung ab Schritt 2, bis das Ergebnis stabil ist (d.h. die Lage der Cluster-Zentren sich nicht mehr ändert).</p> <ul style="list-style-type: none"> • Modellbasierte Verfahren • Es wird “Wissen” über die gesuchte Struktur bzw. deren Form verwendet. • Beispiel: Finden eines vorgegebenen Gesichts in einer Fotosammlung anhand eines vorgegebenen Passfotos, Hough-Transformation zur Erkennung geometrischer Objekte (Linien, Kreise, Ellipsen, ...). • Texturorientierte Verfahren • Bildobjekte können neben einheitlicher Wertebereichen (Farb-/Intensitätswerte) auch durch eine einheitliche Textur definiert sein. Beispiele: • Unterschiedlich gefärbte Jeansstoffe mit identischer Webung, die aber über ihre Struktur eindeutig der gleichen Stoffart zugeordnet werden können. • Segmentierung unterschiedlicher Bereiche im Knochen (z.B. Spongiosa vs. andere Bereiche), auch wenn diese ähnliche Intensitätswerte aufweisen (bspw. bei CT-Bilddaten). • Teils bereits an der Grenze zur Klassifikation³ • Auf diese Verfahren werden wir hier allerdings nicht näher eingehen. <p>Zur Erinnerung: Szene → Bildaufnahme → Bildvorverarbeitung → Segmentierung → Merkmalsextraktion → Klassifikation → Aussage</p>
<p>Segmentierung: Beispiel eines modellbasierten Verfahrens</p> <p>Welches Modellbasierte Segmentierungsverfahren kennen wir aus der VL?</p> <p>Was ist das Prinzip dahinter?</p> <p>Was ist der Nachteil der Segmentierung?</p>	<ul style="list-style-type: none"> • Eine sog. Hough-Transformation kann dazu genutzt werden, mathematisch einfach zu parametrierende geometrische Figuren in Bilddaten zu identifizieren [11, 5]. • Idealerweise liegt ein Schwarz-Weiß-Bild nach Kantenfilterung oder Skelettierung vor. • Beispiel: Hough-Transformation für Linien • Im kartesischen Koordinatensystem kann eine Linie durch ihre Geradengleichung $y = m \cdot x + b$ angegeben werden (Parameter m, b) • Die Entsprechung in Polarkoordinatensystem ist $y = \left(-\frac{\cos \theta}{\sin \theta}\right) x + \left(\frac{r}{\sin \theta}\right) \Leftrightarrow r = x \cos \theta + y \sin \theta.$  <p>Fortsetzung Hough-Transformation [11]:</p> <ul style="list-style-type: none"> • Für jeden im Binärbild gesetzten Punkt (x₀, y₀) werden nun alle möglichen Kombinationen mit θ durchgespielt (drei verschachtelte Schleifen) • Jede Kombination aus r und θ wird im sog. Hough-Akkumulator eingetragen (bzw. der Wert an der entsprechenden Stelle im Akkumulator um 1 erhöht). • Sind alle Punkte durchlaufen, passt (vermutlich) diejenige Parameterkombination aus r und θ am besten zur gesuchten Linie im Bild, für die der Hough-Akkumulator den größten Wert hat. • Nachteil der Houghtransformation: rel. hoher Rechenaufwand. • Lässt sich aber durch “schlaue” Aufbereitung evtl. reduzieren. • Je nach Anwendungsfall z.B. Berechnung auf Kantenbild oder einer skelettierten Version starten ;-)

Bildanalyse VL

	<div data-bbox="560 255 770 288">Originalbild der Meise</div> <div data-bbox="649 306 925 445"></div> <div data-bbox="560 456 775 481"><code>Code: pythoncode/opencv-hough.py</code></div> <div data-bbox="1011 255 1321 288">Linientransformation für Zweige</div> <div data-bbox="1102 306 1378 445"></div>
--	---