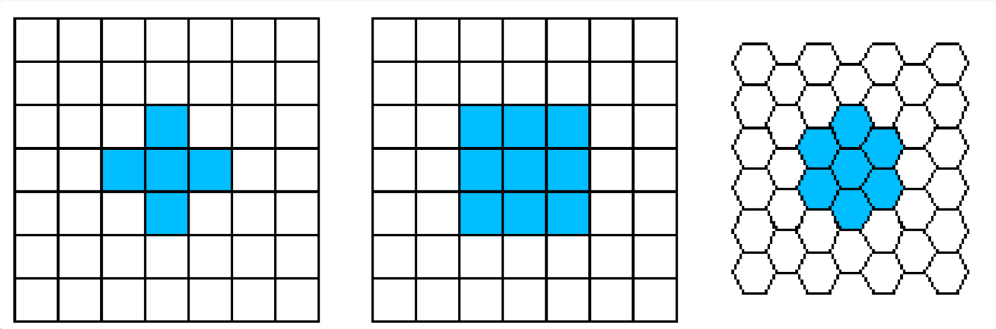
5. Lokale Operationen

Quellen:

- EVC_Skriptum_CV, p.24 bis EVC_Skriptum_CV, p.28

Nachbarschaften:



- Eine Nachbarschaft bezeichnet eine kleine, definierte Bildregion um ein Pixel, um Bildverarbeitungsoperationen durchzuführen.

Vierer-Nachbarschaft

- Jedes Pixel P hat 2 horizontale und 2 vertikale Nachbarn.
- Koordinaten des Pixels P: (u, v).
- Koordinaten der vier D-Nachbarn:
  $(u-1, v), (u+1, v), (u, v-1), (u, v+1)$
- Eine Vierer-Nachbarschaft besteht aus 5 Punkten (Pixel P + 4 D-Nachbarn).

Achter-Nachbarschaft

- Neben den D-Nachbarn hat jedes Pixel P auch 4 diagonale Nachbarn.
- Koordinaten der diagonalen Nachbarn:
  $(u-1, v-1), (u+1, v+1), (u-1, v+1), (u+1, v-1)$
- Eine Achter-Nachbarschaft besteht aus 9 Punkten (Pixel P + 4 D-Nachbarn + 4 diagonale Nachbarn).

Abstand der Nachbarn

- Der Abstand der Nachbarn wird durch die Metrik festgelegt:
  - Euklidische Metrik: Abstand beträgt $\sqrt{2}$.
  - Manhattan-Metrik: Abstand beträgt 2.

Was sind lokale Operationen

Punktoperationen

- Der neue Wert eines Bildelements hängt ausschließlich vom ursprünglichen Bildwert an derselben Position ab.
- siehe 4. Punktoperationen

Lokale Operationen (Filter)

- Ähnlichkeit zu Punktoperationen: Auch hier besteht eine 1:1-Abbildung der Bildkoordinaten, d. h., die Geometrie des Bildes bleibt unverändert.
- Unterschied zu Punktoperationen: Das Ergebnis wird nicht nur aus einem einzigen Ursprungspixel berechnet, sondern aus mehreren Pixeln des Originalbildes.
- Die Koordinaten der Quellpixel sind bezüglich der aktuellen Position (u, v) definiert und bilden eine zusammenhängende Region.

Filterregion

- Die Größe der Filterregion bestimmt, wie viele ursprüngliche Pixel zur Berechnung des neuen Pixelwerts beitragen und damit das räumliche Ausmaß des Filters.
- Eine gängige Filtergröße ist 3 × 3, zentriert in der Achter-Nachbarschaft um die Koordinate (u, v).
- Die Form der Filterregion muss nicht quadratisch sein, sondern kann beliebige Formen annehmen.

Lineare Filter

- Bezeichnung: Lineare Filter verbinden die Pixelwerte innerhalb der Filterregion in linearer Form, d. h., durch eine gewichtete Summation.
- Beispiel: Die lokale Mittelwertbildung ist ein einfaches Beispiel, bei dem alle neun Pixel der 3 × 3 Filterregion mit der Gewichtung $1/9$ summiert werden.

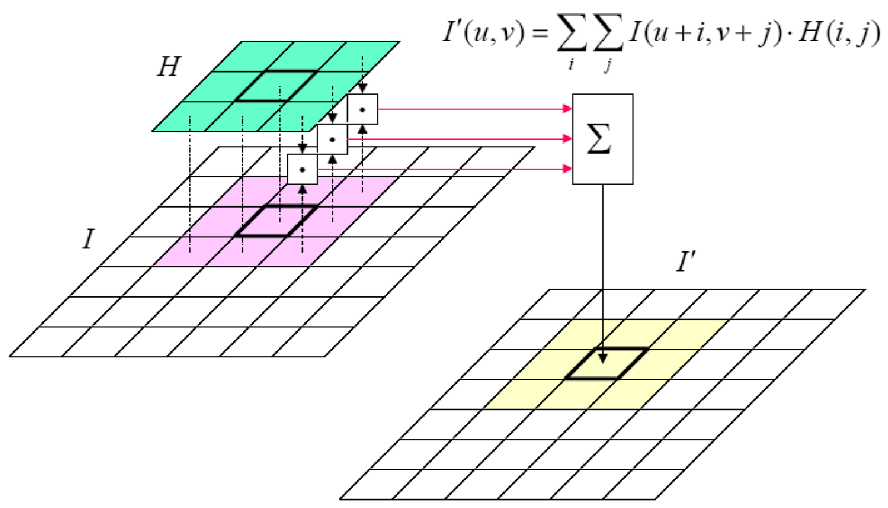


Abbildung 30: Berechnung einer Faltung (Convolution)

Filtermatrix

- Definition: Eine Filtermatrix oder Filtermaske $F(i,j)$ spezifiziert die Größe, Form und die zugehörigen Gewichte der Filterregion.
  - Die Größe der Matrix entspricht der Größe der Filterregion.
  - Jedes Element $F(i,j)$ der Matrix definiert das Gewicht des entsprechenden Pixels.
- Einzigartigkeit: Das Ergebnis eines linearen Filters ist eindeutig und vollständig durch die Koeffizienten in der Filtermatrix bestimmt.

Anwendung des Filters

- Die Anwendung eines linearen Filters auf ein Bild erfolgt durch folgende Schritte:
  1. Positionierung der Filterfunktion F: Die Filtermatrix F wird so über das Bild I positioniert, dass ihr Koordinatenursprung $F(0,0)$ auf das aktuelle Bildelement $I(u,v)$ fällt.
  2. Multiplikation und Summation: Alle Bildelemente in der Filterregion werden mit den jeweils darüber liegenden Filterkoeffizienten multipliziert und die Ergebnisse werden summiert.
  3. Speichern des Ergebnisses: Die resultierende Summe wird an der entsprechenden Position im Ergebnisbild $I'(u,v)$ gespeichert.

Berechnung für den 3 × 3 Filter

- Für einen 3 × 3 Filter des neuen Bildes $I'(u,v)$ wird der Wert für jedes Pixel wie folgt berechnet:
  - Die Schritte 1–3 werden an jeder Position (u,v) im Bild wiederholt, um das gefilterte Bild zu erhalten.

Tiefpassfilter

Unterscheidung zwischen Tiefpass- und Hochpassfiltern

- Tiefpassfilter:
  - Filtern hohe Frequenzen heraus und lassen niedrige Frequenzen passieren.
  - Eignen sich für Rauschunterdrückung bzw. als Glättungsoperatoren.
  - Bekannte Tiefpassfilter: Mittelwertfilter und Gauß-Filter.
- Hochpassfilter:
  - Filtern tiefe Frequenzen heraus und lassen hohe Frequenzen passieren.
  - Eignen sich z. B. für die Kantendetektion. (siehe 8. Bildmerkmale - Interest Points)

Mittelwertfilter (Box-Filter)

- Filtermaske: Besteht aus lauter gleichen Gewichten (1), einfachste Form aller Tiefpassfilter.
- Nachteile:
  - Scharf abfallende Ränder und unoptimales Frequenzverhalten.
  - Alle Bildelemente haben das gleiche Gewicht, wodurch das Zentrum nicht stärker gewichtet wird als die Ränder.

Gauß-Filter

- Filtermaske: Entspricht einer diskreten, zweidimensionalen Gauß-Funktion.
  - Beispiel für eine Filtermaske (für $\sigma = 0.5$):

$$F_{Gauss} = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

- Eigenschaften:
  - Das mittlere Bildelement erhält das maximale Gewicht.
  - Die Werte der übrigen Koeffizienten nehmen mit zunehmender Entfernung zur Mitte kontinuierlich und gleichmäßig ab (isotrop).
  - Standardabweichung σ bestimmt den „Radius“ der glockenförmigen Funktion und beeinflusst die Stärke der Glättung.

Filtermaske für einen 3 × 3 Gauß-Filter mit $\sigma = 0.5$

- Die resultierende Filtermaske lautet:

$$F_{Gauss} = \begin{bmatrix} 0.011 & 0.084 & 0.011 \\ 0.084 & 0.619 & 0.084 \\ 0.011 & 0.084 & 0.011 \end{bmatrix}$$

- Summe der Koeffizienten muss 1 betragen, was durch Division aller Koeffizienten durch deren Summe erreicht wird.

Wichtige Hinweise

- Größere Filtermasken führen zu einer besseren Approximation der Gauß-Funktion, aber ändern nicht das Glättungsverhalten.
- Die Stärke der Glättung kann durch die Standardabweichung σ variiert werden.
- Ein Mittelwertfilter mit einer 3 × 3 Filtermaske führt zu einem befriedigenden Ergebnis, aber der Gauß-Filter wird im Allgemeinen bevorzugt.

Differenzfilter

Interpretation negativer Filterkoeffizienten

- Wenn einzelne Filterkoeffizienten negativ sind, kann die Filteroperation als Differenz zweier gewichteter Summen verstanden werden:
  $\text{Summe positiver Gewichtungen} - \text{Summe negativer Gewichtungen}$
- Innerhalb der Filterregion R werden:
  - Positive Koeffizienten → positiv gewichtete Pixel.
  - Negative Koeffizienten → negativ gewichtete Pixel.

Beispiel: Laplace-Filter

- Filtermatrix:
  $$F_{Laplace} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$
- Berechnet die Differenz zwischen dem zentralen Pixel (-4) und den 4 umliegenden Pixeln (Vierer-Nachbarschaft).
- Übrige 4 Pixel (diagonale Nachbarn) haben Koeffizienten 0 → werden nicht berücksichtigt.

Wirkung der Differenzbildung

- Gegenteil zur Durchschnittsbildung:
  - Durchschnitt → Glättung von Intensitätsunterschieden.
  - Differenz → Verstärkung von Intensitätsunterschieden.
- Anwendungen:
  - Kanten- und Konturverstärkung
  - Bildschärfung
- → Differenzfilter sind Hochpassfilter.

Mathematische Grundlage

- Hochpassfilter basieren auf Ableitungen der Bildfunktion $g(x, y)$:
  - Erste Ableitung → Gradientenfilter

- Zweite Ableitung → Laplace-Filter

Nachbearbeitung des Ergebnisbildes

- Ergebnis enthält oft positive und negative Grauwerte.
- Mögliche Nachbearbeitungen:
    - Normierung auf z. B. $[0, 255]$
    - Betragsbildung: $|g(x, y)|$

Anwendung in Software (z. B. Adobe Photoshop)

- Umsetzung durch sogenannte „Custom Filter"
- Filter mit:
    - Ganzzahligen Koeffizienten
    - Skalierungsfaktor (Scale)
    - Offset-Wert, um negative Ergebnisse in den sichtbaren Intensitätsbereich zu verschieben.

Bildrandproblem

- Beim Anwenden von Filtern kann es zu Problemen an den Bildrändern kommen.
- Ursache: Die Filterregion überschreitet den Bildbereich, es fehlen passende Pixelwerte → das Filterergebnis kann nicht berechnet werden.
- Es gibt keine mathematisch exakte Lösung für das Problem
- andere Probleme mit Bildrand siehe: 7. Clipping und Antialiasing

Methoden zum Umgang mit dem Randproblem

1. Einsetzen eines konstanten Werts
    - Beispiel: 0 (schwarz)
    - Nachteil: Verkleinert den sichtbaren Bildbereich.
    - Nicht akzeptabel in den meisten Anwendungen.
2. Beibehalten der ursprünglichen (ungefilterten) Bildwerte
    - Filter wird nicht auf die Randpixel angewendet.
    - Nachteil: Inkonsistente Bildverarbeitung; ebenfalls nicht ideal.
3. Annahme künstlicher Pixelwerte außerhalb des Bildbereichs:
    - (a) Konstanter Wert außerhalb des Bildes (z. B. schwarz oder grau)
        - Kann bei großen Filtern zu starken Verfälschungen an den Rändern führen.
    - (b) Fortsetzung der Randpixel
        - Randwerte des Bildes werden nach außen hin fortgeführt.
        - Geringe Verfälschung → bevorzugte Methode

- (c) Zyklische Wiederholung des Bildes
  - Das Bild wird horizontal und vertikal periodisch fortgesetzt.

Formale Eigenschaften lineare Filter

Ursprung und Definition

- Lineare Filter basieren auf dem mathematischen Konzept der linearen Faltung (engl. *linear convolution*).
- Sie verknüpft zwei Funktionen gleicher Dimensionalität, kontinuierlich oder diskret.
- Für diskrete, 2D-Funktionen I (Bild) und F (Filtermatrix) ist die Faltung definiert als: $I' = I * FI'$
- Dabei gilt (mit Berücksichtigung der Koordinatenumkehr):

  $I'(u,v) = \sum_i \sum_j I(u-i, v-j) \cdot F(-i,-j)$
- Die ursprüngliche lineare Filterdefinition entspricht einer linearen Korrelation, da hier keine Spiegelung der Filtermatrix erfolgt.

Eigenschaften der linearen Faltung

1. Kommutativität:
   $I * F = F * I$
   → Reihenfolge von Bild und Filter spielt keine Rolle.
2. Linearität:
   - Skalierung eines Bildes:
     $(a \cdot I) * F = a \cdot (I * F)$
   - Addition zweier Bilder:
     $(I_1 + I_2) * F = (I_1 * F) + (I_2 * F)(I1 + I2) * F = (I1 * F) + (I2 * F)$
3. Assoziativität:
   $A * (B * C) = (A * B) * C$
   → Filter können beliebig kombiniert und umgruppiert werden.

Konsequenz: Separierbarkeit

- Ein Filterkern F kann als Faltungsprodukt kleinerer Filterkerne beschrieben werden: $F = F_1 * F_2 * \cdots * F_n$
- Besonders nützlich: Trennung in zwei eindimensionale Filter:

Beispiel:

- $F_x = \begin{bmatrix} 1 & 2 & 1 \end{bmatrix}$

- $F_y = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix}$

Kombiniert:
$F_{xy} = F_x * F_y$

- Vorteil: Reduktion der Rechenkomplexität
  - Normal: $3 \times 5 = 15$ Operationen pro Pixel
  - Separiert: $5 + 3 = 8$ Operationen pro Pixel

Nicht lineare Filter

Nachteil linearer Filter

- Lineare Filter glätten nicht nur Störungen, sondern auch gewollte Bildstrukturen wie:
  - Punkte
  - Kanten
  - Linien
- → Bildqualität leidet: Strukturen werden verwischt.
- → Einschränkung ihrer Anwendung bei Struktur- oder Kantenerhaltung.

Rangordnungsfilter (engl. *rank value filters*)

- Nichtlineare Filteroperationen
- Kombination von benachbarten Pixeln durch Vergleichen und Selektieren, statt Gewichten und Addieren.
- Funktionsweise:
  - Alle Grauwerte innerhalb der Filtermaske werden sortiert (aufsteigend).
  - Es wird ein bestimmter Rang aus dieser Liste ausgewählt.
  - Dieser Wert ersetzt das zentrale Pixel.

Typen von Rangordnungsfiltern

- Medianfilter:
  - Wählt den mittleren Wert (Median) der sortierten Grauwerte.
  - Besonders wirksam bei der Entfernung von Ausreißern (z. B. Salz-und-Pfeffer-Rauschen).
- Minimumfilter:
  - Wählt den kleinsten Grauwert in der Region.
- Maximumfilter:
  - Wählt den größten Grauwert in der Region.

Vorteile

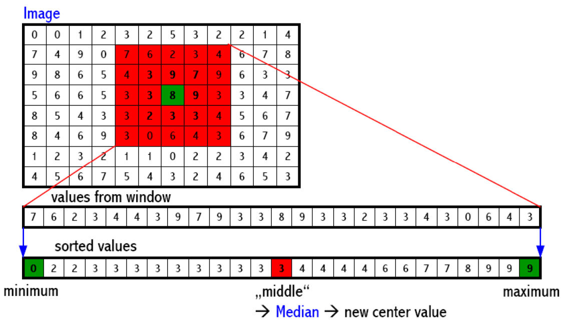
- Besser geeignet zur Erhaltung von Kanten und feinen Strukturen.
- Besonders effektiv bei nicht-gausschem Rauschen.

Definition dieser Filter:

$I(u,v) = \min\{I(u+i, v+j) \text{ für } (i,j) \in R\} = \min(R_{u,v})$ bzw. $I(u,v) = \max\{I(u+i, v+j) \text{ für } (i,j) \in R\} = \max(R_{u,v})$, wobei $R_{u,v}$ die Region der Bildwerte bezeichnet, die an der aktuellen Position (u,v) von der Filterregion überdeckt werden. Die Abbildung 31 zeigt die Anwendung von 3x3-Min- und -Max-Filtern auf ein Grauwertbild, das künstlich mit SSalt-and-PepperSStörungen versehen wurde (zufällig platzierte weiße und schwarze Punkte). Der

Abbildung 31: Min-/Max-Filter Anwendung

Min-Filter entfernt die weißen (Salt) Punkte, denn ein einzelnes, weißes Pixel wird innerhalb der 3x3-Filterregion R immer von kleineren Werten umgeben, von denen einer den Minimalwert liefert. Gleichzeitig werden durch das Min-Filter aber dunkle Strukturen räumlich erweitert. Der Max-Filter hat genau den gegenteiligen Effekt.



Grundproblem bei der Filterung

- Kein Filter kann *automatisch* unterscheiden zwischen:
  - wichtigen Strukturen (z. B. Kanten, Details)
  - unerwünschten Störungen (z. B. Rauschen)
- → Perfekter Filter existiert nicht.
- Jeder Filter trifft eine "blinde Entscheidung", ob ein Pixel zur Struktur oder zur Störung gehört.

Medianfilter – ein sinnvoller Kompromiss

- Ziel: Störungen entfernen, aber Strukturen besser erhalten als bei linearen Filtern.
- Definition:
  Jedes Bildelement I(u, v) wird durch den Median der Pixelwerte innerhalb einer Filterregion R ersetzt: $I(u,v) = median(Ru, v)$

- Der Median aus einer sortierten Liste von $2K + 1$ Pixelwerten p_i ist der mittlere Wert: $median(p0, \ldots, p2K) = pK$ (sofern $pi \leq pi + 1$)

Beispielhafte Wirkung

- Abbildung 32 (gedanklich):
  - Linkes Bild: Originalbild mit Salt-and-Pepper-Rauschen.
  - Mittleres Bild: Nach Anwendung eines Mittelwertfilters – Störungen teilweise noch sichtbar.
  - Rechtes Bild: Nach Anwendung eines Medianfilters – Störungen besser entfernt, Strukturen erkennbar erhalten.

Abbildung 32: Anwendung von Mittelwert- und Medianfilter

Vorteile des Medianfilters

- Robust gegenüber Ausreißern
  - Besonders effektiv bei impulsartigem Rauschen wie Salt-and-Pepper-Noise
- Erhält Kanten besser als der Mittelwertfilter
- Nichtlinear, daher nicht anfällig für lineare Glättungsverluste