
Table of Contents

.....	1
Aufgabe 1a)	1
Aufgabe 1b)	2
Aufgabe 1c)	4
Aufgabe 1d)	5
Aufgabe 1e)	6
Aufgabe 1f)	7
Aufgabe 2	8

```
% Aufgabenblatt 1 und 3 waren die Arbeit von Felix Künnecke.
```

```
Aufgabenblatt
```

```
% 4 ist die Version von Dennis Schlage.
```

```
shadingimage = double(imread('shading.jpg'))/255;  
shadingimage = shadingimage(:,:,1);
```

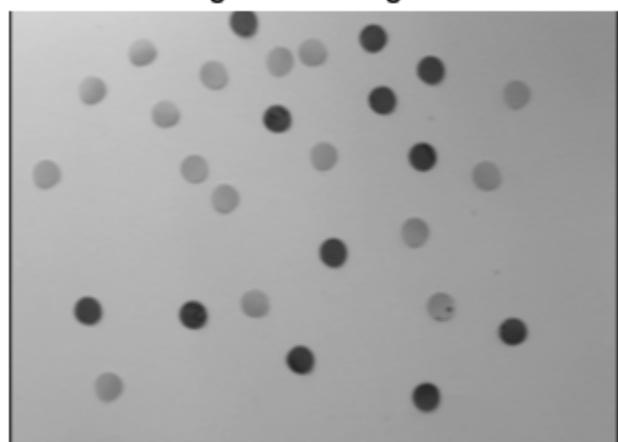
Aufgabe 1a)

```
maske = [1 2 1];  
ergebnisA = conv2(shadingimage, maske);  
figure;  
imshow(ergebnisA);  
title('ergebnisA');  
  
% Warum ist das Ergebnis heller? = Weil die Maske nicht durch die  
% Summe seiner Elemente geteilt wird.  
  
maskeKorrigiert = maske ./ 4  
ergebnisAKorrigiert = conv2(shadingimage, maskeKorrigiert);  
figure;  
imshow(ergebnisAKorrigiert);  
title('ergebnisAKorrigiert');  
  
maskeKorrigiert =  
0.2500    0.5000    0.2500
```

ergebnisA



ergebnisAKorrigiert



Aufgabe 1b)

```
ergebnisB = shadingimage;
ergebnisBS = shadingimage;
ergebnisBV = shadingimage;

for m = 1:50
    ergebnisB = conv2(ergebnisB, maskeKorrigiert);
    ergebnisBS = conv2(ergebnisBS, maskeKorrigiert, 'same');
    ergebnisBV = conv2(ergebnisBV, maskeKorrigiert, 'valid');
end
```

```

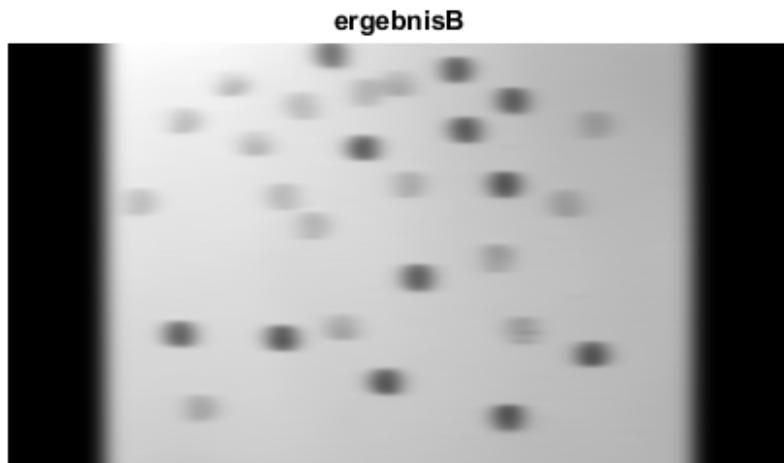
figure;
imshow(ergebnisB, []);
title('ergebnisB');

figure;
imshow(ergebnisBS, []);
title('ergebnisB mit shape=same');

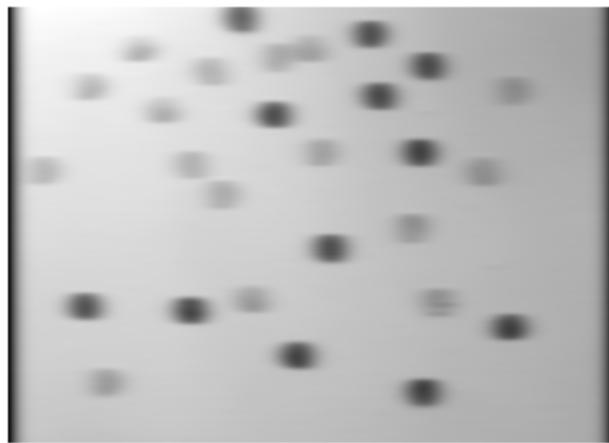
figure;
imshow(ergebnisBV, []);
title('ergebnisB mit shape=valid');

% a) Wie hat sich das Bild optisch verändert?
% Zwei Schwarze Streifen am linken und rechten Rand sind durch das
padding
% hinzugekommen und das Bild wirkt verwaschen.
% b) Wie hat sich die Größe verändert? = Die Spaltenanzahl hat sich in
% jedem Durchlauf um 2 erhöht.
% c) Woher kommen die schwarzen Streifen?
% Werden zum berechnen der Faltung eingefügt in der ersten und letzten
% Spalte des urprünglichen Bildes um die notwendigen Rechenschritte
% auch
% an den Rändern durchführen zu können. Mit dem Parameter 'same'
% werden die
% angehefteten Zeilen bzw. Spalten vor der Ausgabe wieder entfernt und
% der
% Parameter 'valid' sorgt dafür, dass die Pixel an den Rändern, die
% aufgrund ihrer Positionierung nicht die gesamte darüber gelegte
% Maske
% mit benachbarten Pixel des originalen Bildes abdecken können,
% gestrichen
% werden.

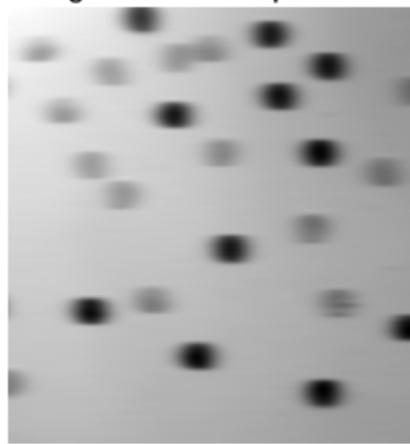
```



ergebnisB mit shpae=same



ergebnisB mit shape=valid



Aufgabe 1c)

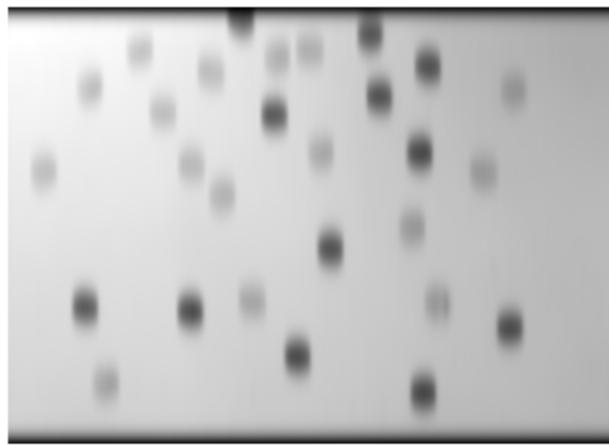
```
ergebnisT = shadingimage;
maskeT = [1;2;1];

for i = 1:50
    ergebnisT = conv2(ergebnisT, maskeT ./ 4, 'same');
end

figure;
imshow(ergebnisT, []);
title('Ergebnis mit transponierter Maske');
```

```
% a) Wie hat sich das Bild optisch verändert?
% Zwei Schwarze Streifen am oberen und unteren Rand sind durch das
padding
% hinzugekommen und das Bild wirkt verwaschen.
% b) Wie hat sich die Größe verändert? = Die Zeilenanzahl hat sich in
jedem
% Durchlauf um 2 erhöht.
% c) Woher kommen die schwarzen Streifen?
% Werden zum berechnen der Faltung eingefügt in der ersten und letzten
% Zeile des urprünglichen Bildes um die notwendigen Rechenschritte
auch an
% den Rändern durchführen zu können.
```

Ergebnis mit transponierter Maske



Aufgabe 1d)

```
ergebnisT2 = ergebnisBS;

for i = 1:50
    ergebnisT2 = conv2(ergebnisT2, maskeT ./ 4, 'same');
end

figure;
imshow(ergebnisT2, []);
title('Erst Zeilenvektor, dann Spaltenvektor')

ergebnisT3 = ergebnisT;

for i = 1:50
    ergebnisT3 = conv2(ergebnisT3, maske ./4, 'same');
end

figure;
```

```
imshow(ergebnisT3, []);
title('Erst Spaltenvektor, dann Zeilenvektor')

% Ändert sich etwas, wenn Sie es in der anderen
% Reihenfolge (zuerst Spalten-, dann Zeilenvektor als Maske)
% durchführen?
% = Keine Unterschiede, die Faltung ist eine assoziative Operation.
```

Erst Zeilenvektor, dann Spaltenvektor



Erst Spaltenvektor, dann Zeilenvektor



Aufgabe 1e)

```
ergebnisMaskenFaltung = conv2(maske, maskeT)
% Ergebnis der Faltung ist die 3x3 Gauß-Maske, was die separierbarkeit
% der Maske beweist.
```

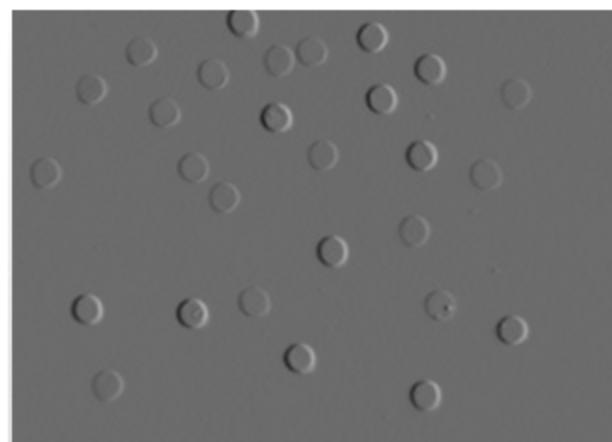
```
ergebnisMaskenFaltung =
```

```
1      2      1  
2      4      2  
1      2      1
```

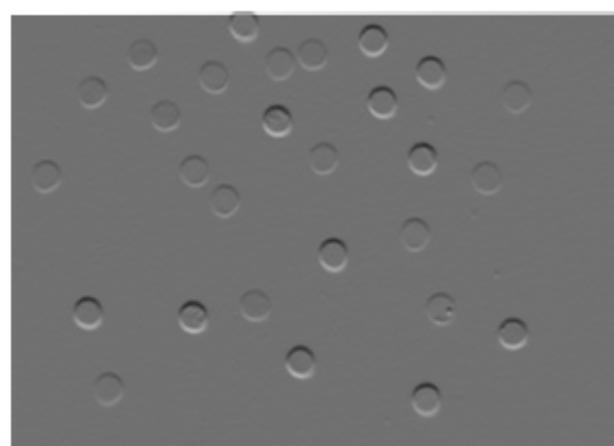
Aufgabe 1f)

```
maskeK = [1 0 -1];  
maskeKT = [1;0;-1];  
  
ergebnisK = conv2(shadingimage, maskeK ./ 2);  
figure;  
imshow(ergebnisK, []);  
title('maskeK');  
  
ergebnisKT = conv2(shadingimage, maskeKT ./ 2);  
figure;  
imshow(ergebnisKT, []);  
title('maskeKT');  
  
% Wie kommt es du den hellen und dunklen Stellen im Ergebnisbild? =  
% Die hellen Stellen im Ergebnisbild mit der Zeilenvektor-Maske  
entstehen  
% dadurch, dass das Ergebnis der Faltung an einer gewissen Position  
immer  
% dann einen größeren Wert einnimmt, wenn der rechts benachbarte Wert  
% größer als der linke ist. Grund dafür ist die Spiegelung der Maske  
% während der Faltungsoperation.  
% Umgekehrt kommen die dunklen Stellen im Ergebnisbild mit der  
% Zeilenvektor-Maske dadurch zustande, dass das Ergebnis der Faltung  
an  
% einer gewissen Position immer dann einen kleineren Wert einnimmt,  
wenn  
% der rechts benachbarte Wert kleiner als der linke ist.  
%  
% Im Ergebnisbild mit der Spaltenvektor-Maske kommen die hellen und  
dunklen  
% Stellen im Bild ebenso zustande wie im Ergebnisbild mit dem  
Zeilenvektor  
% mit dem Unterschied, dass die hellen Stellen sich überall dort zu  
finden  
% sind, wo der darunter liegende Pixel größer ist als der darüber  
liegende  
% und die dunklen Stellen dort, wo der darunter liegende Pixel kleiner  
ist  
% als der darüber liegende.  
%  
% Je größerer die Differenz der betroffenen Benachbarten Pixel, desto  
% heller bzw. dunkler das Ergebnis der Faltung.
```

maskeK



maskeKT



Aufgabe 2

a)

```
bildKorrektur = shadingimage;  
  
maskeG = [1 2 1] ./ 4;  
maskeGT = [1;2;1] ./ 4;  
  
for i=1:800  
    bildKorrektur = padarray(bildKorrektur, [1  
1], 'replicate', 'both');  
    bildKorrektur = conv2(maskeG, maskeGT, bildKorrektur, 'valid');
```

```
end

figure;
imshow(bildKorrektur)
title('Korrekturbild');

% b)
korrigiertesBild = shadingimage ./ bildKorrektur;

figure;
imshow(korrigiertesBild);
title('Korrigiertes Bild');

% c)
figure;
plot(hist(shadingimage(:, 255)));
title('Histogramm Original');

figure;
plot(hist(korrigiertesBild(:, 255)));
title('Histogramm korrigiertes Bild');
```

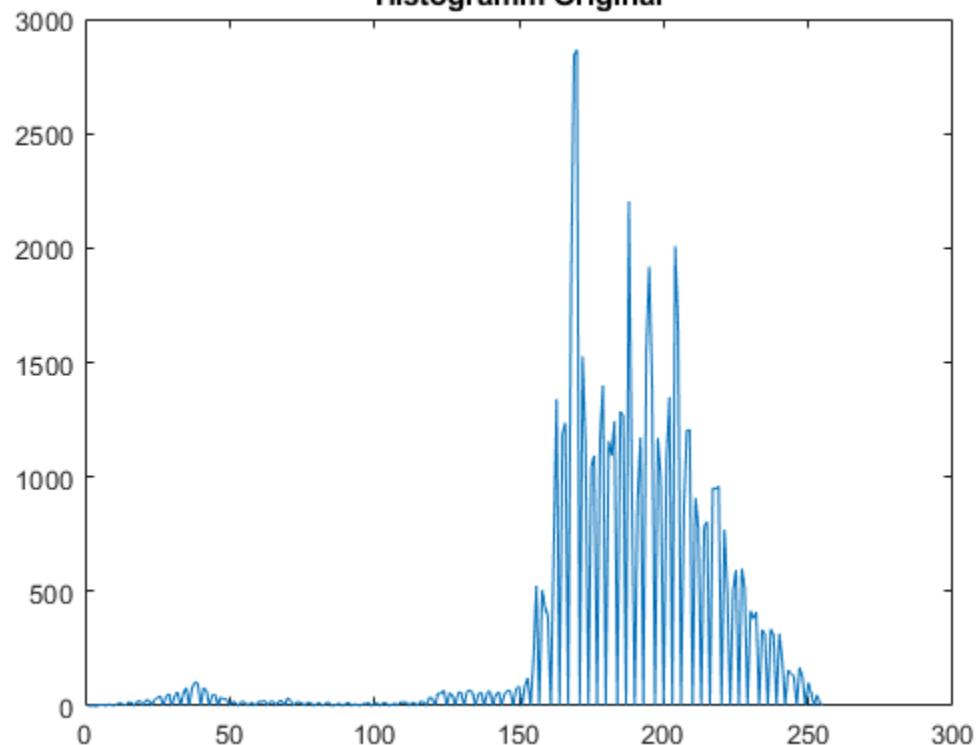
Korrekturbild

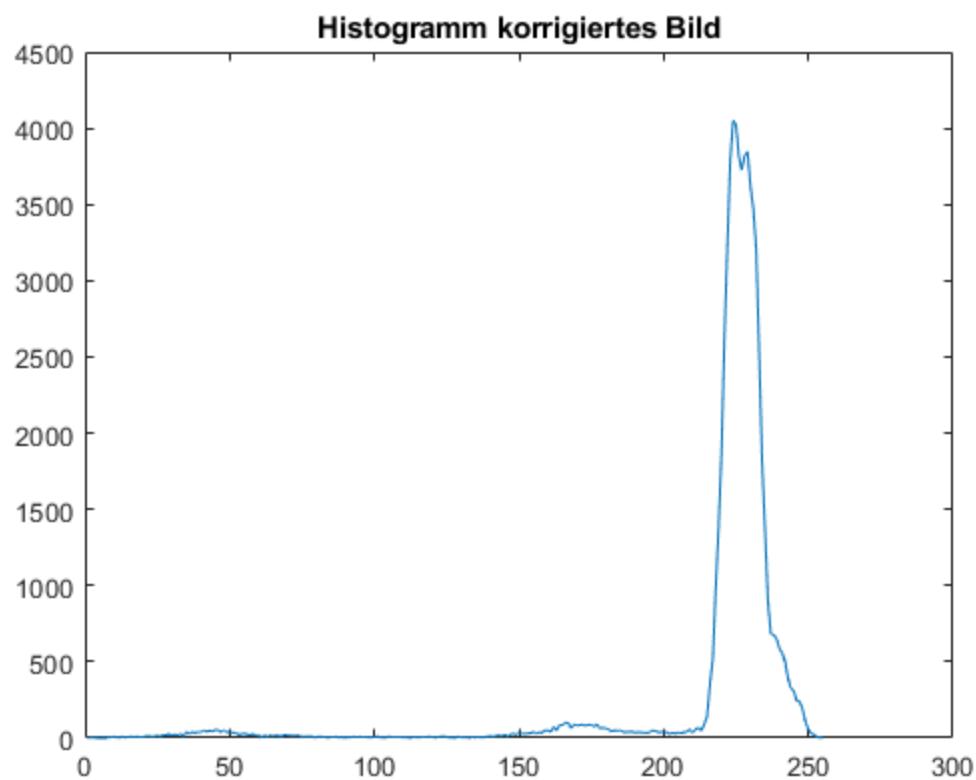


Korrigiertes Bild



Histogramm Original





Published with MATLAB® R2019a