
Bildverarbeitung SoSe 2012

Übung 5

Max Michels, Sebastian Kürten

2 Aufgabe 2

Listing 1: Hauptdatei a2.m

```
1 % load images
2 LENA = imread("lena-bw.png");
3 LADY = imread("old-lady-bw.png");
4
5 %%%% part a %%%%
6
7 % enlarge * 4 with bilinear interpolation
8 % not really part of the homework, but interesting anyway
9 LENA_enlarged4 = bilinear2(LENA);
10 imwrite(uint8(LENA_enlarged4), "lena-enlarged4.png");
11
12 LADY_enlarged4 = bilinear2(LADY);
13 imwrite(uint8(LADY_enlarged4), "old-lady-enlarged4.png");
14
15 % enlarge * 16 with bilinear interpolation
16 LENA_enlarged16 = bilinear4(LENA);
17 imwrite(uint8(LENA_enlarged16), "lena-enlarged16.png");
18
19 LADY_enlarged16 = bilinear4(LADY);
20 imwrite(uint8(LADY_enlarged16), "old-lady-enlarged16.png");
21
22 %%%% part b %%%%
23
24 % shrink to a sixteenth
25 LENA_shrunked = shrink4x4(LENA);
26 imwrite(uint8(LENA_shrunked), "lena-shrunked.png");
27
28 LADY_shrunked = shrink4x4(LADY);
29 imwrite(uint8(LADY_shrunked), "old-lady-shrunked.png");
30
31 % enlarge to original size again
32 LENA_reresized = bilinear4(LENA_shrunked);
33 imwrite(uint8(LENA_reresized), "lena-reresized.png");
34
35 LADY_reresized = bilinear4(LADY_shrunked);
36 imwrite(uint8(LADY_reresized), "old-lady-reresized.png");
```

2.1 Bilineare Interpolation

Es wurde die bilineare Interpolation für die Vergrößerung um den Faktor zwei implementiert. Man kann leicht sehen, dass die Vergrößerung um den Faktor vier durch zweifache Anwendung der zweifachen Vergrößerung erreicht werden kann. Jedoch kann bei zweifacher Vergrößerung für die letzte Zeile und bei vierfacher Vergrößerung für die letzten drei Zeilen nichts berechnet werden.

Listing 2: Bilinear Interpolation: bilinear4.m

```

1 function A = bilinear4(B)
2 % increase the size of an image by factor four by applying
3 % bilinear interpolation to generate unknown pixels
4     A = bilinear2(bilinear2(B));
5 end;

```

Listing 3: Bilinear Interpolation: bilinear2.m

```

1 function A = bilinear2(B)
2 % double the size of an image by applying bilinear interpolation
3 % to generate unknown pixels
4     [h, w] = size(B);
5     A = zeros(h * 2 - 1, w * 2 - 1);
6     % first insert original image's values
7     A([1:2:h*2-1], [1:2:w*2-1]) = B;
8     % then interpolate horizontally
9     for y = [1:2:h*2-1]
10         for x = [2:2:w*2-1]
11             A(y,x) = (A(y,x-1) + A(y,x+1))/2;
12         end;
13     end;
14     % then interpolate vertically
15     for y = [2:2:h*2-1]
16         for x = [1:w*2-1]
17             A(y,x) = (A(y-1,x) + A(y+1,x))/2;
18         end;
19     end;
20 end;

```

2.2 Vergleich mit den Originalbildern

Die vergrößerten Bilder sehen auch auf Originalgröße betrachtet sehr gut aus. Im Vergleich zum Original sind sie im Detail etwas unscharf und man erkennt beim genauen hinsehen leichte horizontale und vertikale Artefakte.

2.3 Verkleinerung und bilineare Rückinterpolation

Die Verkleinerung um Faktoren 4 und 16 wurden wie untenstehend implementiert:

Listing 4: Verkleinerung um Faktor 4: shrink2x2.m

```

1 function A = shrink2x2(B)
2 % shrink an image to a quarter of its original size by calculating the mean
3 % of each 2x2 of the original image to create 1 pixel of the new image.
4     Bd = double(B);
5     [h, w] = size(B);
6     A = zeros(h / 2, w / 2);
7     for y = 1 : (h / 2)
8         for x = 1 : (w / 2)
9             A(y, x) = (ones(1,2) * Bd(y*2-1:y*2, x*2-1:x*2) * ones(2, 1)) ...
                        / 4;
10        end;
11    end;
12 end;

```

Listing 5: Verkleinerung um Faktor 16: shrink4x4.m

```

1 function A = shrink4x4(B)
2 % shrink an image to a sixteenth of its original size by calculating the mean
3 % of each 4x4 of the original image to create 1 pixel of the new image.
4     Bd = double(B);
5     [h, w] = size(B);
6     A = zeros(h / 4, w / 4);
7     for y = 1 : (h / 4)
8         for x = 1 : (w / 4)
9             A(y, x) = (ones(1,4) * Bd(y*4-3:y*4, x*4-3:x*4) * ones(4, 1)) ...
                        / 16;
10        end;
11    end;
12 end;

```

2.4 Vergleich mit den Originalbildern

Die rücktransformierten Bilder sehen deutlich unscharf aus. Beim genauen hinsehen erkennt man deutlich horizontale und vertikale Artefakte.