

ΕΠΕΞΕΡΓΑΣΙΑ ΕΙΚΟΝΑΣ

Εργασία 1

ΠΑΝΤΕΛΕΗΜΩΝ ΠΡΩΙΟΣ

ice18390023

9ο Εξάμηνο

ice18390023@uniwa.gr

Πέμπτη 16:00-19:00

Ομάδα χρηστών 18



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
UNIVERSITY OF WEST ATTICA

Υπεύθυνοι καθηγητές

ΕΥΦΡΟΣΥΝΗ ΔΟΥΤΣΗ

Τμήμα Μηχανικών και Πληροφορικής Υπολογιστών
4 Δεκεμβρίου 2022

Περιεχόμενα

1	Εισαγωγή	1
2	Επαναφορά εικόνας από διαστάσεις 256 x 128	1
3	Επαναφορά εικόνας από διαστάσεις 128 x 256	3
4	Επαναφορά εικόνας από διαστάσεις 64 x 64	4
5	Συμπέρασμα	5
6	Appendix A	6

Κώδικες

6.1	Script παραγωγής εικόνων και σύγκρισης με την αρχική	6
-----	--	---

Κατάλογος σχημάτων

1.1	Original image	1
2.1	256 x 128 antialiasing=true	2
2.2	256 x 128 antialiasing=false	2
3.1	128 x 256 antialiasing=true	3
3.2	128 x 256 antialiasing=false	4
4.1	64 x 64 antialiasing=true	5
4.2	64 x 64 antialiasing=false	5
6.1	128x256 bicubic false	9
6.2	128x256 bicubic true	9
6.3	128x256 bilinear false	10
6.4	128x256 bilinear true	10
6.5	128x256 nearest false	11
6.6	128x256 nearest true	11
6.7	256x128 bicubic false	12
6.8	256x128 bicubic true	12
6.9	256x128 bilinear false	13
6.10	256x128 bilinear true	13
6.11	256x128 nearest false	14
6.12	256x128 nearest true	14
6.13	64x64 bicubic false	15
6.14	64x64 bicubic true	15
6.15	64x64 bilinear false	16
6.16	64x64 bilinear true	16
6.17	64x64 nearest false	17
6.18	64x64 nearest true	17

1 Εισαγωγή

Σκοπός της εργασίας, είναι η υπό-δειγματοληψία της εικόνας 1.1 σε 3 διαφορετικά scales και έπειτα η επαναφορά της στο αρχικό scale με τις μεθόδους του nearest neighbour, bilinear και bicubic, όπως και η χρήση του θεωρήματος Nyquist-Shannon.



Σχήμα 1.1: Original image

2 Επαναφορά εικόνας από διαστάσεις 256 x 128

Με την χρήση του nearest neighbour και χωρίς υπερκάλυψη, η εικόνα έχει ποίο απαλές αλλαγές όμως, ενώ με υπερκάλυψη έχει ποίο έντονα σημεία και περισσότερο θόρυβο.

Με την χρήση του bilinear, όταν υπάρχει υπερκάλυψη, το πρόσωπο είναι ποίο ευδιάκριτο, ενώ το παλτό στις άκρες τετραγωνίζει και όταν δεν υπάρχει η επικάλυψη, το πρόσωπο είναι ποίο θολό ενώ το παλτό έχει ομαλές αλλαγές στις άκρες.

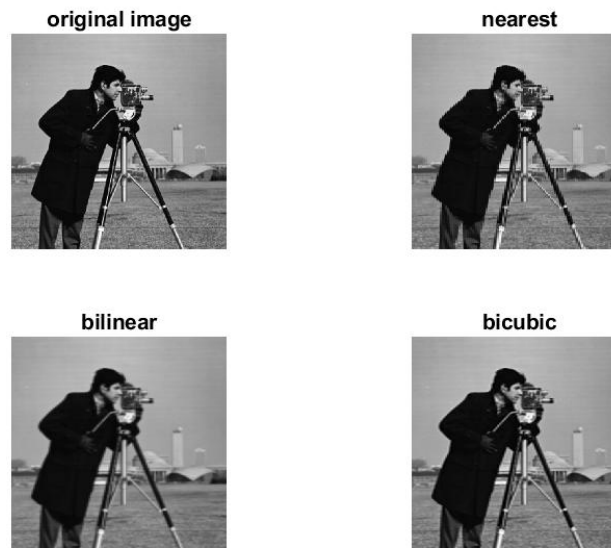
Με την χρήση του bicubic, όταν δεν υπάρχει επικάλυψη, οι αλλαγές στις άκρες είναι ποίο απαλές ενώ σε αυτή που υπάρχει επικάλυψη είναι ελάχιστα ποίο τετραγωνισμένα τα pixel και για αυτό το PSNR είναι περίπου ίδιο (πίνακας 2.1).

Το καλύτερο αποτέλεσμα είναι με την χρήση της μεθόδου bicubic όπου έχει το μεγαλύτερο PSNR ενώ ταυτόχρονα δεν υπάρχει υπερκάλυψη. Ωστόσο, και με υπερκάλυψη το αποτέλεσμα είναι πολύ κάλο και η διαφορά των PSNR είναι περίπου 0.15 dB μόνο.

Πίνακας 2.1: Scale 256 x 128

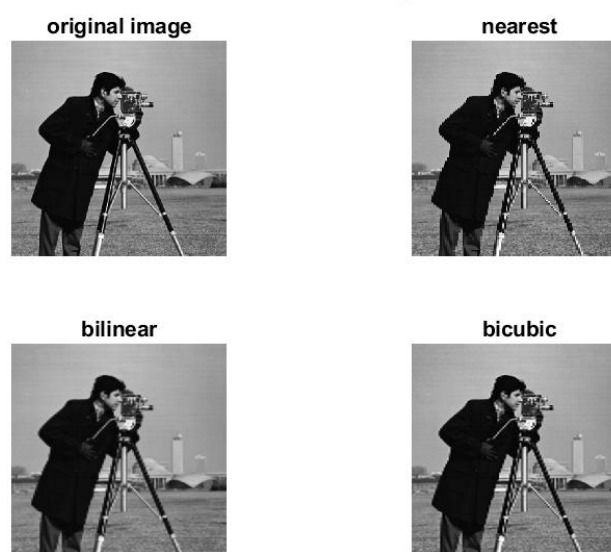
	256 x 128					
	Nearest		Bilinear		Bicubic	
	<i>MSE</i>	<i>PSNR</i>	<i>MSE</i>	<i>PSNR</i>	<i>MSE</i>	<i>PSNR</i>
TRUE	131.7739	26.9325	129.5956	27.0049	82.701	28.9557
FALSE	221.9682	24.6679	95.7959	28.3173	85.6143	28.8053

256 x 128 antialiasing=true



Σχήμα 2.1: 256 x 128 antialiasing=true

256 x 128 antialiasing=false



Σχήμα 2.2: 256 x 128 antialiasing=false

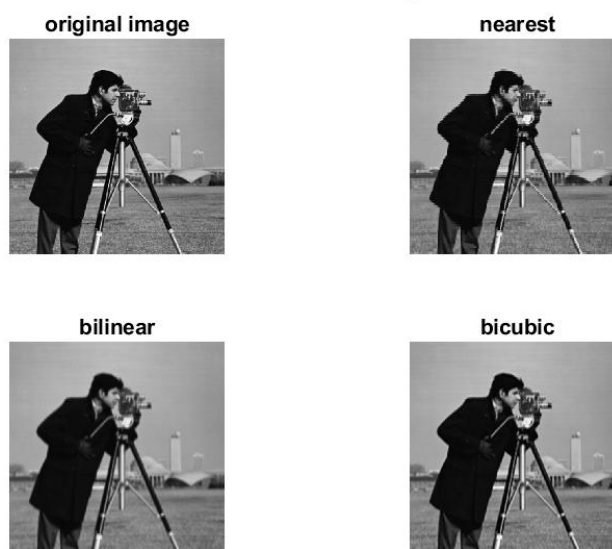
Πίνακας 3.1: Scale 128 x 256

	128 x 256					
	Nearest		Bilinear		Bicubic	
	<i>MSE</i>	<i>PSNR</i>	<i>MSE</i>	<i>PSNR</i>	<i>MSE</i>	<i>PSNR</i>
TRUE	103.3453	27.9879	98.0858	28.2147	55.7339	30.6696
FALSE	150.6307	26.3517	71.5322	29.5858	55.0914	30.72

3 Επαναφορά εικόνας από διαστάσεις 128 x 256

Στην περίπτωση του nearest neighbour και της bilinear, τα αποτελέσματα είναι παρόμοια, εκτός από την bilinear χωρίς υπερκάλυψη που έχει ποίο ομαλές αλλαγές σε σύγκριση με τις άλλες εικόνες αλλά δεν έχει λεπτομέρειες. Ακόμα, με την bicubic, όταν υπάρχει υπερκάλυψη υπάρχουν ποίο έντονα στοιχεία και θόρυβος, ενώ χωρίς υπερκάλυψη το αντίθετο. Ωστόσο, το αποτέλεσμα είναι σχεδόν ίδιο και για αυτό τα PSNR τους έχουν διαφορά μόνο 0.05 dB. Επίσης, συγκριτικά με τις εικόνες 256 x 128 υπάρχουν καλύτερα αποτελέσματα.

128 x 256 antialiasing=true

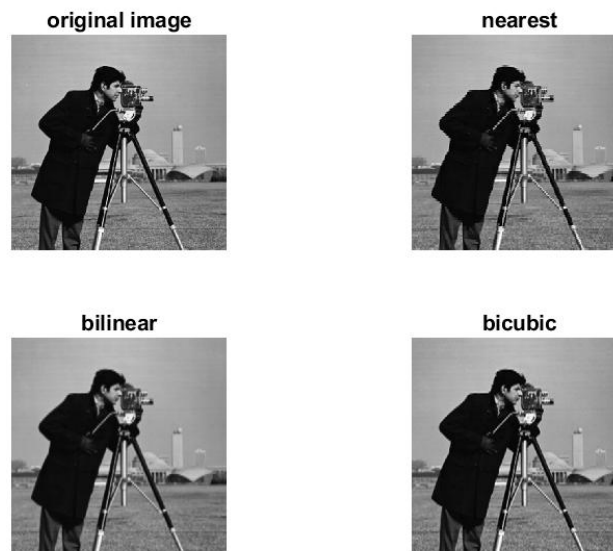


Σχήμα 3.1: 128 x 256 antialiasing=true

Πίνακας 4.1: Scale 64 x 64

	64 x 64					
	Nearest		Bilinear		Bicubic	
	<i>MSE</i>	<i>PSNR</i>	<i>MSE</i>	<i>PSNR</i>	<i>MSE</i>	<i>PSNR</i>
TRUE	388.4638	22.2373	376.3971	22.3743	301.7713	23.334
FALSE	619.5327	20.2102	368.9755	22.4608	400.598	22.1037

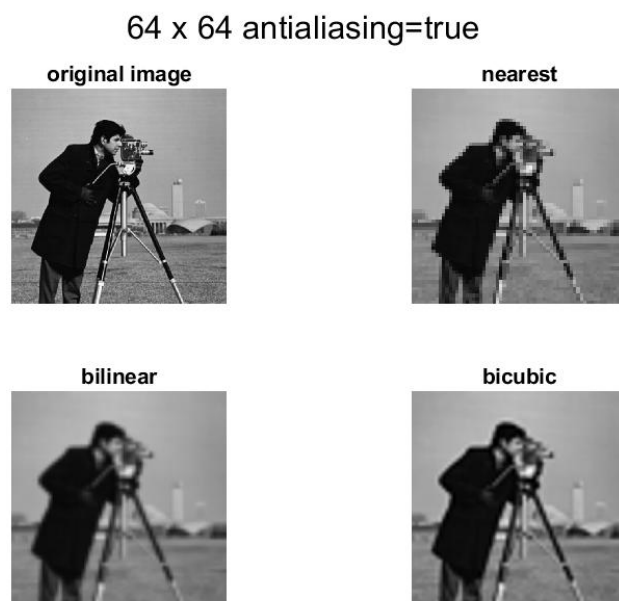
128 x 256 antialiasing=false



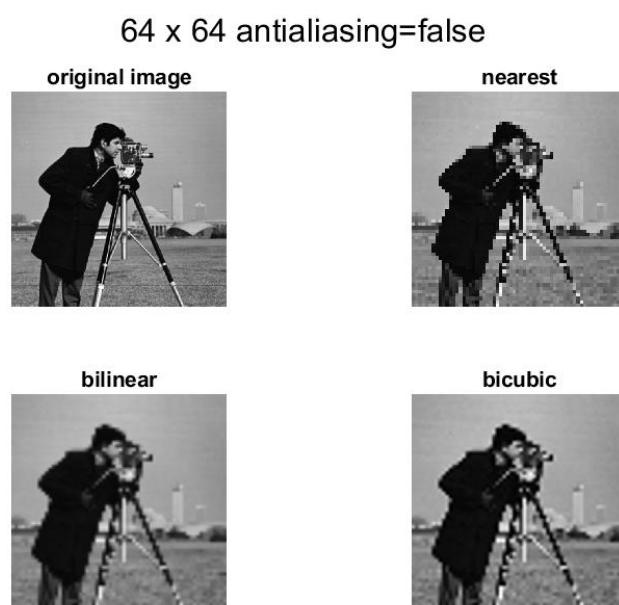
Σχήμα 3.2: 128 x 256 antialiasing=false

4 Επαναφορά εικόνας από διαστάσεις 64 x 64

Στις εικόνες με διαστάσεις 64 x 64, είναι έντονα αντιληπτή η ανάγκη να μην γίνεται υπερκάλυψη. Ωστόσο, η μέθοδος του nearest neighbour είναι η χειρότερη ακόμα και χωρίς υπερκάλυψη.



Σχήμα 4.1: 64 x 64 antialiasing=true



Σχήμα 4.2: 64 x 64 antialiasing=false

5 Συμπέρασμα

Από τα αποτελέσματα των μετρικών MSE και PSNR, είναι φανερό πως η καλύτερη μέθοδος είναι αυτή της bicubic σε συνδυασμό με την τήρηση του θεωρήματος Nyquist-Shannon.

6 Appendix A

```

1 %% init
2
3 % read image
4 original_img = imread('cameraman.tif');
5
6 % true if you want to save images to folder images/
7 output_images_on_folder=false;
8
9 % size of original image
10 N = size(original_img,1); % rows
11 M = size(original_img,2); % columns
12
13 % methods for sampling
14 methods = [ "nearest", "bilinear", "bicubic" ];
15
16 % anti-aliasing true/false options
17 aliasings = [ "false", "true" ];
18
19 % scales for downsampling
20 % row , column
21 sample_scales = [ 1/2 , 1/4;
22                  1/4 , 1/2;
23                  1/8 , 1/8];
24
25 %% downsampling and upsampling
26
27 % array with image based on scale method and anti-aliasing
28 % out(img=[X,Y], scale=3, methods=3, aliasing=2)
29
30 % This is how for loop works
31 % for each scale
32 % for each Antialiasing=false/true
33 % for each method
34 % store out
35
36
37 for k = 1:size(sample_scales,1)
38
39     scale = [N M].*sample_scales(k,:);
40
41     for aliasing=0:1 %size(aliasings,2)
42
43         for l=1:size(methods,2)
44
45             method = methods(l);
46
47             tmp = imresize(original_img,scale, method, "Antialiasing", aliasing);
48             out(:,k,l,aliasing+1) = imresize(tmp,[N M], method, "Antialiasing", aliasing);
49
50             if output_images_on_folder == true
51                 imwrite(out(:,k,l,aliasing+1), "images/" + ...
52                     scale(1) + "x" + scale(2) + "_" + ...
53                     method + "_" + aliasings(aliasing+1) + ".jpg", "jpg");
54             end
55         end
56     end

```



```

57 end
58
59
60 %% show images
61
62 % out(img=[X,Y], scale=3, methods=3, aliasing=2)
63
64 % for each aliasing
65 % for each scale
66 % for each method
67 % subplot 2x2
68
69 for i = 1:size(aliasings,2)
70
71     for j = 1:size(sample_scales,1)
72
73         figure
74         subplot(2,2,1)
75         imshow(original_img)
76         title('original image')
77
78         for k = 1:size(methods,2)
79
80             subplot(2,2,k+1)
81             imshow(out(:,j,k,i))
82             title(methods(k))
83         end
84         sgtitle(strcat(num2str(N*sample_scales(j,1)), " x ", num2str(M*sample_scales(j,2)),...
85 " antialiasing=", aliasings(i)))
86
87         if output_images_on_folder == true
88             print(strcat(num2str(N*sample_scales(j,1)), " x ",...
89 num2str(M*sample_scales(j,2)),...
90 " antialiasing=", aliasings(i)),'-djpeg')
91         end
92     end
93 end
94
95
96 %% MSE PSNR
97
98 % out(img=[X,Y], scale=3, methods=3, aliasing=2)
99
100 % for each aliasing
101 % for each scale
102 % for each method
103 % subplot 2x2
104
105 for i = 1:size(aliasings,2)
106     for j = 1:size(sample_scales,1)
107
108         scale = [N M].*sample_scales(j,:);
109
110         fprintf('\n %d x %d Aliasing=\'%s\' \n',....
111             scale(1), scale(2), aliasings(i));
112
113         for k = 1:size(methods,2)

```

```
115     err_mse = immse(original_img, out(:,j,k,i));
116     dB_psnr = 10.*log10(255^2/err_mse);
117     fprintf(' MSE=%0.4f PSNR=%0.4f (%s)\n',...
118           err_mse, dB_psnr, methods(k));
119
120     end
121 end
122 end
```

Κώδικας 6.1: Script παραγωγής εικόνων και σύγκρισης με την αρχική



Σχήμα 6.1: 128x256 bicubic false



Σχήμα 6.2: 128x256 bicubic true



Σχήμα 6.3: 128x256 bilinear false



Σχήμα 6.4: 128x256 bilinear true



Σχήμα 6.5: 128x256 nearest false



Σχήμα 6.6: 128x256 nearest true



Σχήμα 6.7: 256x128 bicubic false



Σχήμα 6.8: 256x128 bicubic true



Σχήμα 6.9: 256x128 bilinear false



Σχήμα 6.10: 256x128 bilinear true



Σχήμα 6.11: 256x128 nearest false



Σχήμα 6.12: 256x128 nearest true



Σχήμα 6.13: 64x64 bicubic false



Σχήμα 6.14: 64x64 bicubic true



Σχήμα 6.15: 64x64 bilinear false



Σχήμα 6.16: 64x64 bilinear true



Σχήμα 6.17: 64x64 nearest false



Σχήμα 6.18: 64x64 nearest true