Ψηφιακή Επεξεργασία εικόνας

Αναφορά 1

1-2)

Για την υποδειγματοληψία (downsampling) χρησιμοποίησα τη συνάρτηση imresize της Matlab η οποία δέχεται σαν ορίσματα την αρχική εικόνα, τον αριθμό σειρών και στηλών σε ένα vector 2 στοιχείων και σαν τρίτο όρισμα δέχεται μια μέθοδο παρεμβολής. Έτσι όταν σαν τρίτο όρισμα βάλουμε “nearest” το antialiasing είναι false και επομένως δεν εκτελείται antialiasing κατά τη συρρίκνωση της εικόνας. Αντιθέτως, εάν σαν τρίτο όρισμα βάλουμε είτε “bicubic” είτε “bilinear” τότε το antialiasing είναι true και εκτελείται antialiasing κατά την συρρίκνωση.

<u>Κώδικας συρρίκνωσης:</u>

```
shrinkNoAntialiasing = imresize(image, [rows/2, columns/4], 'nearest'); %shrinking image without antialiasing (antialiasing false)
shrinkAntialiasing = imresize(image, [rows/2, columns/4], 'bicubic'); %shrinking image with antialiasing (antialiasing true)

shrinkNoAntialiasing = imresize(image, [rows/4, columns/2], 'nearest'); %shrinking image without antialiasing (antialiasing false)
shrinkAntialiasing = imresize(image, [rows/4, columns/2], 'bicubic'); %shrinking image with antialiasing (antialiasing true)

shrinkNoAntialiasing = imresize(image, [rows/8, columns/8], 'nearest'); %shrinking image without antialiasing (antialiasing false)
shrinkAntialiasing = imresize(image, [rows/8, columns/8], 'bicubic'); %shrinking image with antialiasing (antialiasing true)
```

Η συρρίκνωση έγινε με και χωρίς antialiasing και για τις τρεις περιπτώσεις που ζητήθηκαν στην άσκηση, δηλαδή για $\frac{1}{2}$ στις γραμμές και $\frac{1}{4}$ στις στήλες, $\frac{1}{4}$ στις γραμμές και $\frac{1}{2}$ στις στήλες και τέλος για $\frac{1}{8}$ στις γραμμές και $\frac{1}{8}$ στις στήλες.

3)

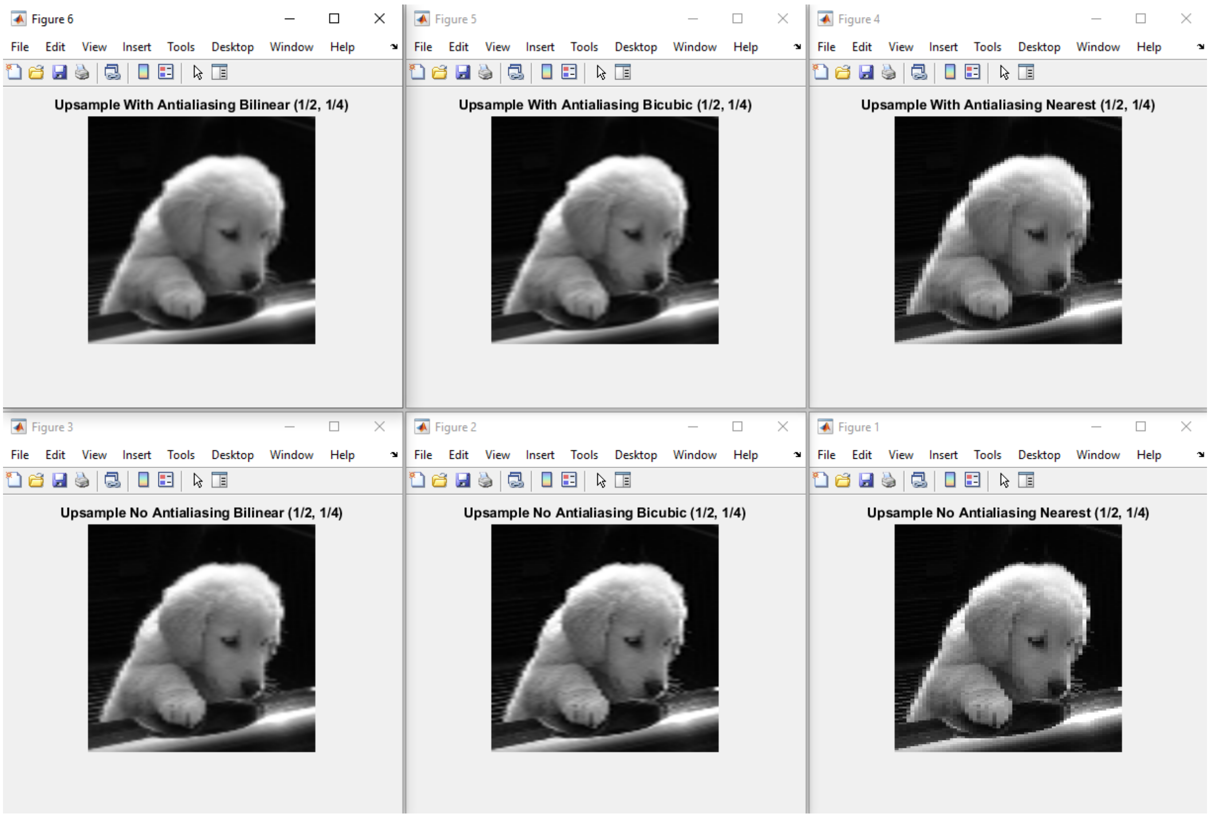
Για την ανακατασκευή της εικόνας χρησιμοποιήθηκε και πάλι η συνάρτηση imresize με τρεις διαφορετικές μεθόδους παρεμβολής (nearest-neighbor, bicubic, bilinear) και για τις δύο περιπτώσεις υποδειγματοληψίας(με antialiasing - χωρίς).

<u>Κώδικας ανακατασκευής:</u>

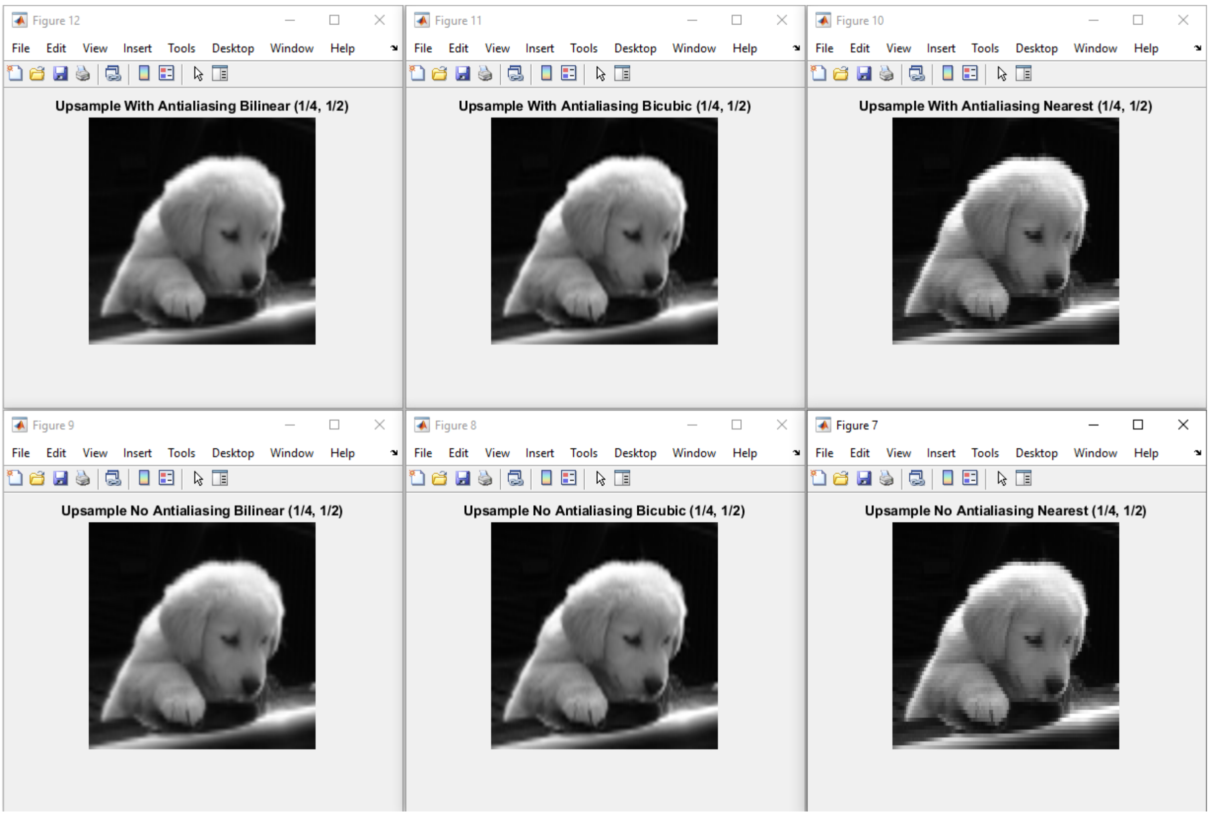
```
upscaleNearestNoAntialiasing1 = imresize(shrinkNoAntialiasing, [rows, columns], 'nearest');
upscaleBicubicNoAntialiasing1 = imresize(shrinkNoAntialiasing, [rows, columns], 'bicubic');
upscaleBilinearNoAntialiasing1 = imresize(shrinkNoAntialiasing, [rows, columns], 'bilinear');

upscaleNearestAntialiasing1 = imresize(shrinkAntialiasing, [rows, columns], 'nearest');
upscaleBicubicAntialiasing1 = imresize(shrinkAntialiasing, [rows, columns], 'bicubic');
upscaleBilinearAntialiasing1 = imresize(shrinkAntialiasing, [rows, columns], 'bilinear');
```

Αποτελέσματα ανακατασκευής:

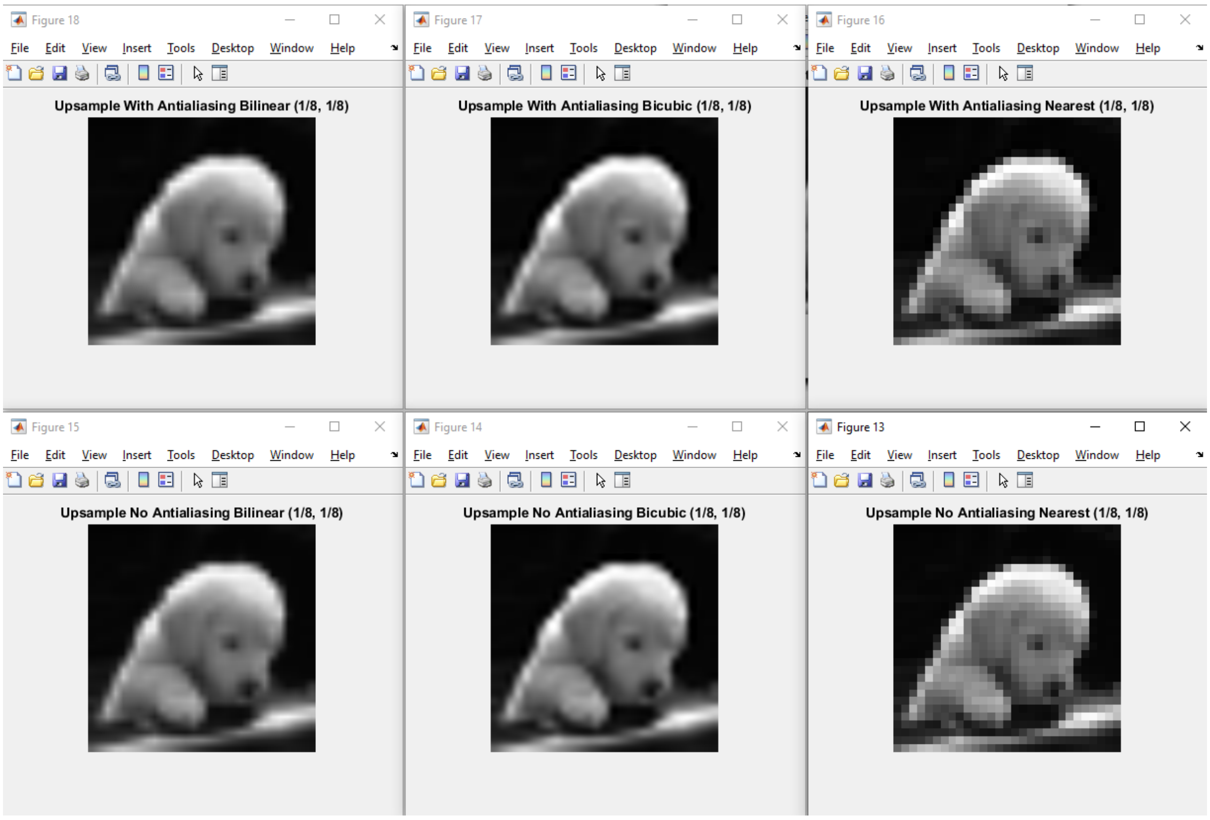
- ½ x ¼



- ¼ x ½



- ⅛ x ⅛



Η μέθοδος ελάχιστων τετραγώνων (Mean Square Error - MSE) μετριέται με βάση το 0, επομένως όσο πιο μικρή η τιμή τόσο καλύτερα. Ο λόγος σήματος προς θόρυβο (Peak Signal to Noise Ratio - PSNR) είναι ένα μέγεθος που σχετίζεται άμεσα με το Mean Square Error (αντίστροφη λογαριθμική κλίμακα και επομένως όσο υψηλότερη τιμή τόσο καλύτερα). Χρησιμοποιώντας τη Μέθοδο ελάχιστων τετραγώνων (Mean Square Error - MSE) παρατήρησα ότι κατά την υποδειγματοληψία με antialiasing, με τη μέθοδο παρεμβολής “bicubic” το MSE είναι μικρότερο απ ότι με τη μέθοδο “bilinear” άρα παράγονται καλύτερης ποιότητας εικόνες. Για αυτό το λόγο χρησιμοποίησα τη μέθοδο “bicubic” για το antialiasing της υποδειγματοληψίας.

<u>MSE - PSNR αποτελέσματα:</u>

- ½ x ¼

MSEnoAntialiasingNearest = 22.3734	PSNRnoAntialiasingNearest = 24.8519
MSEnoAntialiasingBicubic = 17.5407	PSNRnoAntialiasingBicubic = 27.8055
MSEnoAntialiasingBilinear = 17.6187	PSNRnoAntialiasingBilinear = 27.7221
MSEAntialiasingNearest = 19.2095	PSNRAntialiasingNearest = 26.5244
MSEAntialiasingBicubic = 14.7325	PSNRAntialiasingBicubic = 29.7472
MSEAntialiasingBilinear = 16.2949	PSNRAntialiasingBilinear = 28.7238

- ¼ x ½

MSEnoAntialiasingNearest2 = 25.2209	PSNRnoAntialiasingNearest2 = 24.0664
MSEnoAntialiasingBicubic2 = 19.7536	PSNRnoAntialiasingBicubic2 = 26.6765

MSEnoAntialiasingBilinear2 = 19.6964	PSNRnoAntialiasingBilinear2 = 26.7441
MSEAntialiasingNearest2 = 23.8224	PSNRAntialiasingNearest2 = 25.7376
MSEAntialiasingBicubic2 = 16.9581	PSNRAntialiasingBicubic2 = 28.7664
MSEAntialiasingBilinear2 = 18.8924	PSNRAntialiasingBilinear2 = 27.9330

- ⅛ x ⅛

MSEnoAntialiasingNearest3 = 39.5563	PSNRnoAntialiasingNearest3 = 19.7079
MSEnoAntialiasingBicubic3 = 33.2541	PSNRnoAntialiasingBicubic3 = 22.4386
MSEnoAntialiasingBilinear3 = 33.9066	PSNRnoAntialiasingBilinear3 = 22.3070
MSEAntialiasingNearest3 = 41.3623	PSNRAntialiasingNearest3 = 20.9034
MSEAntialiasingBicubic3 = 33.4438	PSNRAntialiasingBicubic3 = 23.3547
MSEAntialiasingBilinear3 = 35.8627	PSNRAntialiasingBilinear3 = 22.5144

Για την πρώτη και δεύτερη περίπτωση (½ x ¼ και ¼ x ½) παρατηρώ ότι το MSE είναι μικρότερο και το PSNR μεγαλύτερο στις εικόνες με antialiasing, που σημαίνει ότι βρίσκονται πιο κοντά στην αρχική εικόνα (καλύτερη ποιότητα). Στην τελευταία περίπτωση (⅛ x ⅛) το MSE είναι μικρότερο στις εικόνες χωρίς antialiasing το οποίο συμβαίνει διότι η εικόνα ήταν πολύ μικρή (μικρός αριθμός pixel) και επομένως δεν είχε καλή επίδραση το antialiasing. Τέλος, παρατηρώ ότι το MSE είναι μικρότερο με τη μέθοδο Bicubic, δεύτερη καλύτερη μέθοδος φαίνεται να είναι αυτή της διγραμμικής (Bilinear) και τρίτη η μέθοδος η nearest-neighbor.