

Ψηφιακή Επεξεργασία Εικόνας

3^η Εργαστηριακή Αναφορά

Εργαστηριακή Ομάδα 76

Γιουμερτάκης Απόστολος, 2017030142

Κατσούπης Ευάγγελος, 2017030077

Κατασκευή συνάρτησης 2D Convolution

Αρχικά έγινε η κατασκευή του Gaussian kernel με την χρήση της συνάρτησης $f_{special}()$ με ορίσματα τον τύπο του, ("gaussian"), το μέγεθος του αλλά και το μέγεθος της τυπικής απόκλισης.

Για την κατασκευή της δικής μας συνάρτησης, όπου θα πραγματοποιεί την συνέλιξη μεταξύ του φίλτρου και του πίνακα που κρατά τις τιμές της εικόνας, δημιουργήθηκε η `my_conv2D()` που παίρνει σαν ορίσματα το όνομα της εικόνας και τον kernel που δημιουργήσαμε και επιστρέφει τον πίνακα με τις νέες τιμές. Αρχικά, γίνεται η ανάκλαση του φίλτρου, με την χρήση της συνάρτησης `rot90()`, καθώς η περιστροφή πίνακα κατά 90° , οδηγεί στην ανάκλασή του. Στην συνέχεια γίνεται το zero-padding της εικόνας, σύμφωνα με τις διαστάσεις του του φίλτρου, τρόπος που αναλύθηκε εκτενώς στο εργαστήριο 2. Στην συνέχεια διαβάζονται παράθυρα μεγέθους Kernel, από την padded εικόνα, όπου το καθένα από αυτά πολλαπλασιάζεται με τις αντίστοιχες τιμές του reflected φίλτρου και θα δίνει την νέα τιμή της εικόνας από το άθροισμά των στοιχείων του.

Αξίζει να σημειωθεί, πως, λόγω της φύσης του φίλτρου, το οποίο περιέχει τιμές τύπου `double`, μετατρέπουμε τις τιμές της εικόνας αναφοράς, στον ίδιο τύπο, με την χρήση της `im2double()`, για να επεξεργαστεί από το φίλτρο και αντίστοιχα σε τύπο `uint8` με την χρήση της `im2uint8()` για να επιστραφεί στον χρήστη.

Το αποτέλεσμα της συνελιγμένης εικόνας συγκριτικά με την αρχική είναι:



Σχήμα 1: Source Image and Convolved Image

Για να μετρήσουμε την παραμόρφωση που υπέστη η convoluted εικόνα έγινε η χρήση:

ι) Mean square error που ορίζεται ως

$$MSE(I, \tilde{I}) = \frac{1}{MN} \sum_i^N \sum_j^M (I(i, j) - \tilde{I}(i, j))^2$$

ιι) Peak Signal-to-Noise Ratio που ορίζεται ως

$$PSNR(I, \tilde{I}) = 10 \log_{10} \left(\frac{\max_I^2}{MSE(I, \tilde{I})} \right)$$

Τα αποτελέσματα μετά από τις παραπάνω μετρήσεις:

| | MSE | $PSNR$ |
|------------------|---------|---------|
| Convuluted Image | 348.620 | 348.620 |

Άσκηση 2

Μας ζητήθηκε η συνέλιξη της αρχικής εικόνας με το kernel (Gaussian filter), το οποίο κατασκευάσαμε στην Άσκηση 1. Αυτήν τη φορά, η συνέλιξη έγινε με την ήδη υπάρχουσα συνάρτηση `conv2()`, η οποία δέχεται ως όρισμα την εικόνα, το kernel καθώς και το σχήμα της εικόνας που προκύπτει από την συνέλιξη των δυο. Για να πάρουμε εικόνα με το ίδιο μέγεθος με την αρχική (αγνοώντας το padding που γίνεται κατά την συνέλιξη), έπρεπε να θέσουμε το τελευταίο όρισμα (σχήμα της εικόνας) με τιμή "same".

Παρακάτω βλέπουμε την εικόνα μετά την συνέλιξη με την συνάρτηση `conv2()` σε σχέση με την αρχική εικόνα:



Στην συνέχεια υπολογίστηκε το Μέσο Τετραγωνικό Σφάλμα (MSE) καθώς και ο Μέγιστος Σηματο-θορυβικός λόγος (PSNR), για την εικόνα πριν και μετά την συνέλιξη, σύμφωνα με τις παραπάνω σχέσεις. Τα αποτελέσματα φαίνονται στον παρακάτω πίνακα:

| function | <code>my_conv2D()</code> | <code>conv2()</code> |
|----------|--------------------------|----------------------|
| MSE | 348.620 | 348.620 |
| $PSNR$ | 22.707 | 22.707 |

Παρατηρούμε ότι η εικόνα που προκύπτει με την χρήση της `conv2()` είναι ακριβώς ίδια με αυτή που προκύπτει στην Άσκηση 1 με χρήση της δική μας συνάρτησης (`my_conv2D()`).

Άσκηση 3

Σε αυτή την άσκηση, μας ζητήθηκε η συνέλιξη της αρχικής εικόνας με το kernel (Gaussian filter), με την χρήση της συνάρτησης `imfilter()`. Η συνάρτηση αυτή, δέχεται ως όρισμα την εικόνα, το kernel καθώς και κάποιες επιπλέον επιλογές που ελέγχουν το φιλτράρισμα της εικόνας. Η πρώτη (1η) επιλογή είναι η "conv" καθώς θέλουμε συνέλιξη μεταξύ του kernel και η δεύτερη "same" ώστε η εικόνα μετά την συνέλιξη να έχει ίδιο μέγεθος με την αρχική.

Παρακάτω βλέπουμε την εικόνα μετά την συνέλιξη με την συνάρτηση `imfilter()` σε σχέση με την αρχική εικόνα:



Όπως και προηγουμένως, υπολογίστηκε το Μέσο Τετραγωνικό Σφάλμα (MSE) καθώς και ο Μέγιστος Σηματοθορυβικός λόγος (PSNR), για την εικόνα πριν και μετά την συνέλιξη, σύμφωνα με τις παραπάνω σχέσεις. Τα αποτελέσματα φαίνονται στον παρακάτω πίνακα:

| function | <i>my_conv2D()</i> | <i>conv2()</i> | <i>imfilter()</i> |
|-------------|--------------------|----------------|-------------------|
| <i>MSE</i> | 348.620 | 348.620 | 348.620 |
| <i>PSNR</i> | 22.707 | 22.707 | 22.707 |

Παρατηρούμε ότι η εικόνα που προκύπτει με την χρήση της `imfilter()` είναι ακριβώς ίδια με αυτή που προκύπτει στην Άσκηση 1 με χρήση της δική μας συνάρτησης (`my_conv2D`) αλλά και με αυτή από την Άσκηση 2 με χρήση της `conv2()` (τα MSE ταυτίζονται).

Άσκηση 4

Σε αυτή την άσκηση, μας ζητήθηκε η απόδειξη της παρακάτω σχέσης:

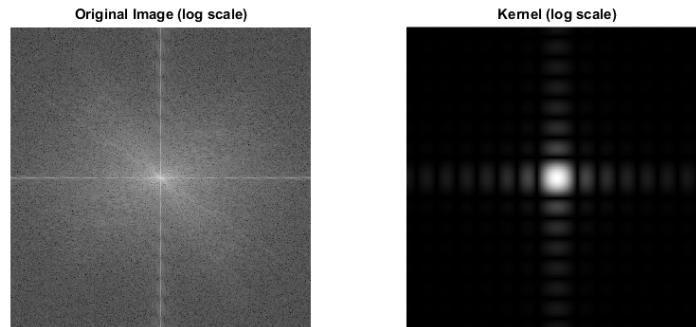
$$f(x, y) * h(x, y) \iff F(u, v)H(u, v)$$

το οποίο σημαίνει ότι η συνέλιξη δύο (2) σημάτων (εικόνας και kernel) στον τομέα του χρόνου είναι ισοδύναμη με το γινόμενό τους στον τομέα της συχνότητας.

Αρχικά, για να υπολογιστεί το γινόμενο των εικόνων στον τομέα της συχνότητας, πρέπει η αρχική εικόνα και το kernel να έχουν τις ίδιες διαστάσεις. Οπότε, κάνουμε zero padding τόσο στην εικόνα όσο και στο φίλτρο. Οι νέες σειρές και στήλες που προστίθενται είναι οι διαστάσεις του kernel για την εικόνα και οι διαστάσεις της εικόνας για το kernel και αυτές προστίθενται στο τέλος του κάθενο.

Στην συνέχεια, με την χρήση του Γρήγορου Μετασχηματισμού Fourier, συναρτήσεις `fft2()` και `fftshift()`, η εικόνα και το kernel μεταφέρονται στον τομέα της συχνότητας.

Παρακάτω βλέπουμε την απεικόνιση της αρχικής εικόνας και του kernel κεντραρισμένα στο φάσμα:



Πολλαπλασιάζοντας αυτά τα δύο (2) στοιχεία, προκύπτει η συνέλιξη τους, που όμως πρέπει να επανέλθει στον τομέα του χρόνου για να εμφανιστεί η επιθυμητή εικόνα.

Με χρήση του Αντίστροφου Μετασχηματισμού Fourier, συνάρτηση `ifft2()`, προκύπτει η εικόνα μετά την συνέλιξη, η οποία έχει μέγεθος μεγαλύτερο της αρχικής. Χρησιμοποιώντας τις σχέσεις:

$$\begin{aligned} cropped_rows &= \lfloor \frac{kernel_rows}{2} + 1 \rfloor \\ cropped_columns &= \lfloor \frac{kernel_columns}{2} + 1 \rfloor \end{aligned}$$

γίνεται περικοπή της εικόνας στο ίδιο μέγεθος με την αρχική.

Παρακάτω βλέπουμε την εικόνα πριν και μετά την συνέλιξη, χρησιμοποιώντας πολλαπλασιασμό στην συγχρότητα.



Όπως και προηγουμένως, υπολογίστηκε το Μέσο Τετραγωνικό Σφάλμα (MSE) καθώς και ο Μέγιστος Σηματοθορυβικός λόγος (PSNR), για την εικόνα πριν και μετά την συνέλιξη. Τα αποτελέσματα φαίνονται στον παρακάτω πίνακα:

| function | <i>my_conv2D()</i> | <i>conv2()</i> | <i>imfilter()</i> | $FFT() \rightarrow IFFT()$ |
|-------------|--------------------|----------------|-------------------|----------------------------|
| <i>MSE</i> | 348.620 | 348.620 | 348.620 | 348.620 |
| <i>PSNR</i> | 22.707 | 22.707 | 22.707 | 22.707 |

Παρατηρούμε ότι και σε αυτή την περίπτωση η διαφορά μεταξύ της αρχικής εικόνας και τις φιλτραρισμένης, είναι ίδια με τις άλλες περιπτώσεις. Επομένως καταλήγουμε στο συμπέρασμα ότι μπορούμε να συνελίξουμε μια εικόνα ή πιο γενικά να φιλτράρουμε μια εικόνα με διαφορετικούς τρόπους, οι οποίοι μας δίνουν τα ίδια αποτελέσματα. Μια διαφορά μεταξύ των διάφορων τρόπων που μελετήθηκαν, είναι ταχύτητα υλοποίησης του φιλτραρίσματος, με την διαδικασία $FFT() \rightarrow IFFT()$ να είναι η πιο γρήγορη.