

Ψηφιακή Επεξεργασία Εικόνας – Άσκηση 2^η

Σοφία Καφρίτσα Γεωργαντά, 2016030136

Σκοπός

Σκοπός της 2^{ης} άσκησης είναι να υλοποιήσουμε την πράξη της συνέλιξης, η οποία είναι από τα βασικά εργαλεία που χρησιμοποιούνται στον τομέα της επεξεργασίας σήματος για τον εντοπισμό ακμών, για την ενίσχυση σήματος. Μας ζητήθηκε να χρησιμοποιήσουμε τρεις τρόπους υλοποίησης: αναπτύσσοντας σε γλώσσα Matlab τη μαθηματική διαδικασία της συνέλιξης, χρησιμοποιώντας τη συνάρτηση `conv2()`, και, τέλος, την `imfilter()`.

Original Image



Επεξεργασία και Ανάλυση

Αρχικά, ορίσαμε το gaussian kernel όπως δόθηκε από την εκφώνηση, κι έπειτα διαβάσαμε την εικόνα. Αρχικό βήμα για να μη χάσουμε πληροφορία κατά τη συνέλιξη, ήταν να κάνουμε padding με μηδενικά ή replication με τα οριακά pixels, ως εξής:

```
padded_image_zero = padarray(I, [floor(size(k,1)/2) floor(size(k,2)/2)]);  
padded_image_replicate = padarray(I, [floor(size(k,1)/2) floor(size(k,2)/2)], 'replicate');
```

1^η περίπτωση:

Μέσα σε δύο for loops διατρέχουμε τα pixels της εικόνας και εφαρμόζουμε το kernel, πάνω σε έναν πίνακα x με προσαρμοσμένο μέγεθος. Προηγουμένως έχουμε δεσμεύσει μνήμη. Επειδή το αποτέλεσμα της συνέλιξης είναι εικόνα με περισσότερα pixels από την αρχική, χρειάζεται να την

Ψηφιακή Επεξεργασία Εικόνας – Άσκηση 2^η

Σοφία Καφρίτσα Γεωργαντά, 2016030136

κάνουμε crop, ώστε να μπορεί να συγκριθεί με την αρχική. Ο κώδικας και τα αποτελέσματα της συνέλιξης φαίνονται παρακάτω:

```
function convImage = convolution( image, kernel )  
    [m1,n1] = size(image);  
    [m2,n2] = size(kernel);  
  
    mn = [m1,n1] + 2*([m2,n2]-1); %518  
  
    a0 = zeros(mn);  
    a0(m2:(end-m2+1),n2:(end-n2+1)) = image;  
  
    convImage = zeros(m2+m1-1,n1+n2-1); %516  
  
    for ii = 1:mn(1)-m2+1 %518-3+1 = 516  
        for jj = 1:mn(2)-n2+1  
            x = a0(ii:ii+m2-1,jj:jj+n2-1);  
            convImage(ii,jj) = x(:)'*kernel(:); %transpose  
        end  
    end  
end
```

Convolution with zero padding



Ψηφιακή Επεξεργασία Εικόνας – Άσκηση 2^η

Σοφία Καφρίτσα Γεωργαντά, 2016030136

Convolution with border replication



Ψηφιακή Επεξεργασία Εικόνας – Άσκηση 2^η

Σοφία Καφρίτσα Γεωργαντά, 2016030136

2^η περίπτωση:

Εφαρμόσαμε τη συνάρτηση `conv2` με την επιλογή “valid”, καθώς θέλουμε η τελική εικόνα να έχει μέγεθος ίδιο με την αρχική, άρα δεν θέλουμε να υπολογίσουμε και τα επιπλέον pixels που προσθέσαμε στο padding/replication. Τα αποτελέσματα φαίνονται παρακάτω:

Convolution with function conv2 and zero padding



Convolution with function conv2 and border replication



Ψηφιακή Επεξεργασία Εικόνας – Άσκηση 2^η

Σοφία Καφρίτσα Γεωργαντά, 2016030136

3^η περίπτωση:

Περάσαμε την εικόνα από το φίλτρο kernel και ορίσαμε παραμέτρους 'same' και 'conv', ώστε να γίνει συνέλιξη μεταξύ τους και, ταυτόχρονα, να πάρουμε το αρχικό μέγεθος. Οι εικόνες φαίνονται παρακάτω:

Filtered Image with zero padding



Filtered Image with Border Replication



Ψηφιακή Επεξεργασία Εικόνας – Άσκηση 2^η

Σοφία Καφρίτσα Γεωργαντά, 2016030136

Αποτελέσματα

Για να συγκρίνουμε τα αποτελέσματα από τον κάθε τρόπο συνέλιξης, δημιουργήσαμε τη συνάρτηση `mse_psnr(image1, image2)`, στην οποία υπολογίζεται με μαθηματικό τρόπο το mse και το psnr μεταξύ δύο εικόνων, ως εξής:

```
function mse_psnr= mse_psnr(image1, image2);  
  
image1 = double(image1);  
image2 = double(image2);  
  
mse = sum((image1(:)-image2(:)).^2) / prod(size(image1))  
psnr = 10*log10(255*255/mse)  
  
end
```

Σύγκριση Αποτελεσμάτων:

	1 ^η περίπτωση		2 ^η περίπτωση		3 ^η περίπτωση	
	MSE	PSNR	MSE	PSNR	MSE	PSNR
Zero padding	25.1956	34.1176	25.1956	34.1176	25.2702	34.1047
Border replication	17.1676	35.7837	17.1676	35.7837	17.2446	35.7643

Σύγκριση μεταξύ zero padding – border replication

Παρατηρούμε ότι το MSE είναι μεγαλύτερο στην μέθοδο zero padding, άρα υπάρχουν μεγαλύτερες αποκλίσεις με την αρχική εικόνα. Παράλληλα, εξασφαλίζουμε μεγαλύτερο PSNR με border replication, άρα περισσότερο χρήσιμο σήμα.

Σύγκριση μεταξύ των τρόπων υλοποίησης

Αρχικά, βλέπουμε ότι η συνέλιξη της 1^{ης} περίπτωσης που υλοποιήθηκε, βγάζει τα ίδια αποτελέσματα με την 2^η περίπτωση που εφαρμόσαμε την `conv2()`. Σε σύγκριση με τη συνάρτηση `imfilter()`, παρατηρούμε ότι τα δεδομένα δεν έχουν πάρα πολύ μεγάλη απόκλιση, καθώς οι διαφορές παρατηρούνται στα δεκαδικά ψηφία. Παρ'όλα αυτά, θα μπορούσαμε να πούμε ότι πιο αποτελεσματικοί είναι οι δύο πρώτοι τρόποι συνέλιξης.