3. Για να διαπιστώσουμε ότι η κρουστική απόκριση της σχέσης [5] αποτελεί διακριτό ισοδύναμο του Gaussian πυρήνα, μπορούμε να συγκρίνουμε τις τιμές που παίρνει η διακριτή προσέγγιση του Gaussian πυρήνα για συγκεκριμένο σ. Συγκεκριμένα, παίρνουμε τις τιμές του Gaussian πυρήνα για τις διακριτές τιμές, από -2 έως 2, τις κανονικοποιούμε και αφαιρούμε από αυτές τις κανονικοποιημένες τιμές της h. Όσο πιο κοντά στο μηδέν είναι η διαφορά τους, τόσο πιο καλή η προσέγγιση του πυρήνα από την h. Για σ=1.1, παίρνουμε μία πολύ καλή προσέγγιση του πυρήνα από τη σχέση [5]. Παρακάτω δίνεται ο κώδικάς για την επαλήθευσή του παραπάνω, μαζί με την έξοδό του:

sigma = 1.1;

n = -2:2; % range of values for n for the Gaussian kernel

% Continuous Gaussian kernel

g\_continuous = exp(-n.^2 / (2 \* sigma^2)) / (sqrt(2 \* pi) \* sigma);

% Normalize the kernel

g\_continuous = g\_continuous / sum(g\_continuous);

% Impulse response

h = [1 4 6 4 1] / 16;

% Normalize the impulse response

h = h / sum(h);

% Compare the normalized kernels

difference\_continuous = g\_continuous - h;

disp('Continuous Gaussian Kernel:');

disp(g\_continuous);

disp('Normalized Impulse Response (h):');

disp(h);

disp('Difference between Continuous Gaussian Kernel and h:');

disp(difference\_continuous);

Έξοδος:

Continuous Gaussian Kernel:

Columns 1 through 5

0.0708 0.2445 0.3695 0.2445 0.0708

Normalized Impulse Response (h):

Columns 1 through 5

0.0625 0.2500 0.3750 0.2500 0.0625

Difference between Continuous Gaussian Kernel and h:

Columns 1 through 5

0.0083 -0.0055 -0.0055 -0.0055 0.0083

6. pyr\_Expand()

Μέσω αυτής της συνάρτησης κάνουμε up-sample το επίπεδο n+1 της Gaussian πυραμίδας, προκειμένου να το χρησιμοποιήσουμε για τη δημιουργία του επιπέδου n της Laplacian πυραμίδας.

Δημιουργούμε αρχικά έναν 1D πυρήνα που καθορίζει το βάρος με το οποίο θα συνεισφέρει κάθε pixel της εικόνας εισόδου, στις interpolated τιμές της διευρυμένης εικόνας. Έπειτα, από τον 1D πυρήνα, δημιουργούμε τον 2D πυρήνα που θα χρησιμοποιήσουμε για να κάνουμε up-sample την εικόνα εισόδου. Κανονικοποιούμε τον πυρήνα πολλαπλασιάζοντας x4 ώστε να διατηρήσουμε την ενέργεια κατά το interpolation.

Για να συνθέσουμε τη διευρυμένη εικόνα, κάνουμε διαπέραση σε κάθε μία από τις διαστάσεις της εικόνας εισόδου. Για κάθε επίπεδο, δημιουργούμε μία padded εικόνα ως προς τις γραμμές και μία ως προς τις στήλες και έπειτα εφαρμόζουμε τα διαφορετικά μέρη του πυρήνα για να δημιουργήσουμε τα κομμάτια που θα απαρτίζουν την τελική διευρυμένη εικόνα. Τέλος, ενώνουμε τα κομμάτια για να δημιουργήσουμε το εκάστοτε διευρυμένο επίπεδο της εικόνας.

pyr\_Reduce()

Μέσω αυτής της συνάρτησης κάνουμε down-sample την εικόνα εισόδου. Τη χρησιμοποιούμε για να δημιουργήσουμε την Gaussian πυραμίδα.

Δημιουργούμε και πάλι έναν 2D πυρήνα, όπως και στην pyr\_Expand(), χωρίς όμως να απαιτείται εδώ η κανονικοποίησή του.

Για να συνθέσουμε την down-sampled εικόνα, ενεργούμε πάνω σε κάθε διάσταση της εικόνα εισόδου. Συγκεκριμένα, εφαρμόζουμε τον Gaussian πυρήνα πάνω σε κάθε διάσταση και συνθέτουμε κάθε “μειωμένο” επίπεδο, επιλέγοντας ανά δύο κάθε pixel της φιλτραρισμένης εικόνας.

pyr\_Reconstruct()

Με αυτή τη συνάρτηση συνθέτουμε από μια Laplacian πυραμίδα την εικόνα. Συγκεκριμένα, ενισχύουμε κάθε επίπεδο με τις λεπτομέρειες του επιπέδου πάνω από αυτό και καταλήγουμε στο πρώτο επίπεδο, το οποίο περιέχει την τελική ενισχυμένη εικόνα.

gen\_Pyr()

Με αυτή τη συνάρτηση δημιουργούμε Gaussian και Laplacian πυραμίδες. Οι Gaussian πυραμίδες δημιουργούνται μέσω της pyr\_Reduce(), που αναφέραμε παραπάνω, και οι Laplacian μέσω της pyr\_Expand() και με χρήση της Gaussian πυραμίδας που δημιουργήθηκε για την εικόνα.

pyr\_Blend.m

Αποτελεί τον κώδικα με τον οποίο συνθέτουμε μία ομοιόμορφα ενωμένη εικόνα ενός πορτοκαλιού και ενός μήλου.

Αρχικά για τις δύο εικόνες, αφού της φέρουμε στο ίδιο μέγεθος, δημιουργούμε μια Laplacian πυραμίδα, 5 επιπέδων, για κάθε μία εξ αυτών. Έπειτα, δημιουργούμε μία μάσκα για κάθε μία, οι οποίες ορίζουν το εύρος της κάθε εικόνας που θα χρησιμοποιηθεί για την συνένωσή τους. Για να είναι ομοιόμορφα ενωμένες οι δύο εικόνες, εφαρμόζουμε στις μάσκες ένα Gaussian φίλτρο, ώστε να υπάρχει ομοιόμορφη μετάβαση από την περιοχή των που θα χρησιμοποιήσουμε προς αυτή που δεν θα χρησιμοποιήσουμε σε κάθε εικόνα.

Για να δημιουργήσουμε τη Laplacian πυραμίδα της ενωμένης εικόνας, εφαρμόζουμε την αντίστοιχη μάσκα σε κάθε επίπεδο της Laplacian πυραμίδας της κάθε εικόνας και ενώνουμε αυτές τις εικόνες για να δημιουργήσουμε το αντίστοιχο επίπεδο της πυραμίδας της ενωμένης εικόνας.

Τέλος, για να ανακτήσουμε την ενωμένη εικόνα, χρησιμοποιούμε την pyr\_Reconstruct() στην Laplacian πυραμίδα που δημιουργήσαμε. Εναλλακτικά, επίσης μπορούμε να εφαρμόσουμε κατευθείαν τις μάσκες πάνω στις αρχικές εικόνες και να τις ενώσουμε. Στη συγκεκριμένη περίπτωση το οπτικό αποτέλεσμα δεν έχει κάποια αισθητή διαφορά.