1. Στην εκφώνηση της άσκησης ο αλγόριθμος SIFT χωρίζεται σε 4 στάδια και έπειτα αναλύονται 5 στάδια. Θα αναφερθούμε στην Εύρεση Τοπικών Σημείων Κλειδιών και στην Ακριβής Εύρεση Τοπικών Σημείων Κλειδιών ως δύο διαφορετικά στάδιο, όπως και αναφέρονται στην ανάλυση των σταδίων.

Τα 5 στάδια του αλγορίθμου SIFT εντοπίζονται στις παρακάτω γραμμές του αρχείου SIFT\_feature.m :

1) Ανίχνευση Ακροτάτων στο Χώρο Κλίμακας: 20 - 68

2) Εύρεση Τοπικών Σημείων Κλειδιών: 69 - 116

3) Ακριβής Εύρεση Τοπικών Σημείων Κλειδιών: 117 - 189

4) Διόρθωση Κατεύθυνσης: 190 - 260

5) Τοπικός Περιγραφέας Σημείου Κλειδιού: 261 - 344

2.

Αρχικά διαβάζουμε την εικόνα cameraman.tif, αλλάζουμε το μέγεθός της σε 256x256 και τη μετατρέπουμε σε double.

Ανίχνευση Ακροτάτων στο Χώρο Κλίμακας

Αρχικοποιούμε τη διασπορά του Gaussian πυρήνα (sigma0=sqrt(2)) και ορίζουμε ότι ο χώρος κλίμακας θα αποτελείται από 3 οκτάβες, με 3 εικόνες ανά οκτάβα.

D=cell(1,octave): δημιουργούμε ένα array άδειων μητρώων ([[],[],[]]). Είναι το array στο οποίο θα αποθηκεύσουμε την πυραμίδα.

D(i)=mat2cell(zeros(row\*2^(2-i)+2,colum\*2^(2-i)+2,level),row\*2^(2-i)+2,colum\*2^(2-i)+2,level): δημιουργούμε ένα array μηδενικών σε κάθε κελί του array D, όπου οι διαστάσεις του καθορίζονται από το i. Σε κάθε κελί δημιουργούμε έναν nxnx3 array, το οποίο αποτελεί την οκτάβα και τα επίπεδά της.

temp\_img=kron(img,ones(2)): διπλασιάζουμε την εικόνα σε μέγεθος, δημιουργώντας ουσιαστικά με κάθε pixel μια γειτονία 4pixel από τον εαυτό του.

temp\_img=padarray(temp\_img,[1,1],'replicate'): προσθέτει ένα στοιχείο padding σε κάθε διάσταση χρησιμοποιώντας τα ακριανά σημεία.

Κάνουμε plot την αρχική εικόνα και την padded εικόνα.

Δημιουργία πυραμίδας:

Δημιουργούμε την πυραμίδα μέσω δυο for, η εξωτερική δημιουργεί τις οκτάβες και η εσωτερική τα επίπεδα κάθε οκτάβας.

Για κάθε επίπεδο κάθε οκτάβας, αρχικοποιούμε την κλίμακα (scale) και δημιουργούμε ένα 1D Gaussian φίλτρο με τυπική απόκλιση ίση με scale.

Για να υπολογίσουμε κάθε επίπεδο κάθε οκτάβας, υπολογίζουμε τη συνέλιξη της temp\_img με το Gaussian φίλτρο, έπειτα τη διαφορά των Gaussian της συνελιγμένης εικόνας με την εικόνα αναφοράς και αποθηκεύουμε την εικόνα στο αντίστοιχο array του προσωρινού πίνακα D.

Αν βρισκόμαστε στο 3ο επίπεδο οποιασδήποτε οκτάβας η τελευταία εικόνα του παρόντος επιπέδου, γίνεται η εικόνα αναφοράς (temp\_img) του επόμενου επιπέδου, αφού πρώτα αφαιρέσουμε την πρώτη γραμμή των περιφερειακών pixel.

Αφού ολοκληρωθεί μια οκτάβα, αποθηκεύουμε τα επίπεδά της στο i-οστό κελί του πίνακα D, υποδειγματοληπτούμε την εικόνα αναφοράς κατά 2 και έπειτα την κάνουμε padding κατά ένα σημείο σε κάθε μεριά της κάθε διάστασης χρησιμοποιώντας τα ακριανά σημεία.

Εύρεση Τοπικών Σημείων Κλειδιών

Αρχικοποιούμε τις μεταβλητές που θα χρειαστούμε κατά την εύρεση τοπικών σημείων κλειδιών, την interval που ορίζει σε πιο επίπεδο κάθε οκτάβας θα τελειώσει η αναζήτηση, ενός πίνακα (extrema) στον οποίο θα αποθηκεύουμε τις πληροφορίες των σημείων κλειδιών και τη flag που θα αποτελεί το δείκτη με βάση τον οποίο θα διαπερνάμε τον πίνακα extrema για την αποθήκευση πληροφοριών.

Η αναζήτηση τον τοπικών σημείων κλειδιών γίνεται μέσω 3ων for. Η πρώτη for διαπερνά τις οκτάβες, η δεύτερη τα επίπεδα κάθε οκτάβας (ξεκινώντας από το δεύτερο έως το interval) και η τρίτη τα pixel κάθε επιπέδου. Σε κάθε οκτάβα, οι δύο τελευταίες σειρές και στήλες αγνοούνται λόγω του padding που έχουμε κάνει.

Για κάθε pixel του επιπέδου από το οποίο ξεκινάμε (της εκάστοτε οκτάβας), παίρνουμε μια περιοχή 3x3 από το ίδιο, το προηγούμενο και το επόμενο επίπεδο και εντοπίζουμε την μέγιστη και ελάχιστη τιμή του pixel μεταξύ αυτών των 3ων 3x3 περιοχών. Αν το μεσαίο pixel της 3x3 περιοχής του επιπέδου που ερευνούμε είναι αυτό με την μέγιστη ή την ελάχιστη τιμή, αποθηκεύουμε τις πληροφορίες του στον πίνακα extrema (οκτάβα, επίπεδο, pixel και αν είναι μέγιστο ή ελάχιστο (1, -1 αντίστοιχα)).

Αφού εντοπίσουμε τα τοπικά σημεία κλειδιά, απαλείφουμε από τον πίνακα extrema τις περιττές θέσεις. Υπολογίζουμε τις x, y συντεταγμένες των σημείων κλειδιών κάνοντάς τες scaling ταυτόχρονα και έπειτα τις κανονικοποιούμε ώστε να είναι συμβατές μεταξύ διαφορετικών οκτάβων. Τέλος, κάνουμε plot την εικόνα μας, με τα μαρκαρισμένα τοπικά σημεία κλειδιά.

Ακριβής Εύρεση Τοπικών Σημείων Κλειδιών

Αρχικοποιούμε το threshold για το φιλτράρισμα των σημείων σε περιοχές ομοιογενούς φωτεινότητας και για τα σημεία που αντιστοιχούν σε ακμές, υπολογίζουμε τον αριθμό των σημείων κλειδιών που βρήκαμε στο προηγούμενο βήμα και δημιουργούμε δύο 2D συνελίξεις για να υπολογίσουμε αργότερα την δεύτερης τάξης παράγωγο στους x, y άξονες.

Με την πρώτη for υπολογίζουμε τη συνάρτηση D για κάθε σημείο κλειδί, για κάθε επίπεδο κάθε οκτάβας. Συγκεκριμένα, υπολογίζουμε τη δεύτερης τάξης παράγωγο για κάθε επίπεδο κάθε οκτάβας και έπειτα ανανεώνουμε το εκάστοτε επίπεδο της πυραμίδας, ενσωματώνοντας πληροφορία από την δεύτερης τάξης παραγώγου ως προς τους άξονες x,y καθώς και της αρχικής πληροφορίας της εικόνας. Αυτό θα μας βοηθήσει στο να φιλτράρουμε μετέπειτα τα σημαντικότερα σημεία κλειδιά.

Με την δεύτερη for, απορρίπτουμε τα σημεία τα οποία ανήκουν σε περιοχές ομοιογενούς ομοιότητας. Για κάθε σημείο κλειδί που εντοπίσαμε (extr\_volume), υπολογίζουμε τις x, y συντεταγμένες του με βάση τις πληροφορίες που έχουμε αποθηκεύσει στον extrema array, κάνοντάς τες scale με την αντίστοιχη οκτάβα τους. Κάνουμε μία μετατροπή στις συντεταγμένες χρησιμοποιούμε δεικτοδότηση με βάση το 1 και υπολογίζουμε και την z συντεταγμένη, η οποία είναι ουσιαστικά το επίπεδο της οκτάβας που μας ενδιαφέρει. Ανακτούμε την τιμή του pixel που μας ενδιαφέρει από την DoG πυραμίδα και αν είναι μικρότερο κατά απόλυτη τιμή του threshold τα ορίζουμε ως μη ενδιαφέροντα σημεία.

Βρίσκουμε τις θέσεις όπου εντοπίσαμε μη ενδιαφέροντα σημεία και δημιουργούμε ένα μητρώο με ένα μητρώο το οποίο περιέχει διευρυμένα indices για κάθε σημείο. Αυτό το κάνουμε για να εξασφαλίσουμε την εξάλειψη γειτονικών σημείων με το ανεπιθύμητο, τα οποία πολύ πιθανόν να περιέχουν παρόμοια πληροφορία με αυτό. Τέλος, απομακρύνουμε τα ανεπιθύμητα σημεία από τον πίνακα των ακροτάτων και υπολογίζουμε εκ νέου τον αριθμό των ακροτάτων που έχουμε. Υπολογίζουμε τις συντεταγμένες x, y κάνοντάς τες και μετατροπή σε δεικτοδότηση με βάση το 1, προσθέτοντας 1 στο τελικό αποτέλεσμα των υπολογισμών. Έπειτα, υπολογίζουμε τα x,y που αντιστοιχούν στις διαστάσεις της αρχικής εικόνας και προβάλουμε τα σημεία αυτά πάνω στην αρχική εικόνα.

Για κάθε σημείο κλειδί, υπολογίζουμε τις συντεταγμένες x, y που αντιστοιχούν στην πραγματική εικόνα και έπειτα υπολογίζουμε την δεύτερης τάξης παράγωγο στο αντίστοιχο σημείο. Υπολογίζουμε την ορίζουσα, το ίχνος του Hessian μητρώου και ορίζουμε το threshold. Αν η ορίζουσα είναι αρνητική, δηλαδή η περιοχή που βρισκόμαστε δεν παρουσιάζει κάποια καμπυλότητα, ή το ίχνος του Hessian μητρώου είναι μεγαλύτερο του threshold μαρκάρουμε το pixel ως μη ενδιαφέρον.

Κάνουμε τα ίδια βήματα με πριν για την απομάκρυνση των μη ενδιαφερόντων σημείων, υπολογίζουμε τις συντεταγμένες των τελικών σημείων κλειδιών, με βάση τις διαστάσεις της αρχικής εικόνας μας και τα προβάλουμε πάνω στην αρχική εικόνα.