



ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ

ΕΠΕΞΕΡΓΑΣΙΑ ΦΩΝΗΣ ΚΑΙ ΕΙΚΟΝΑΣ



ΧΑΝΙΑ, ΙΑΝΟΥΑΡΙΟΣ 2012

ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ ΚΡΗΤΗΣ
ΠΑΡΑΡΤΗΜΑ ΧΑΝΙΩΝ
ΤΜΗΜΑ ΗΛΕΚΤΡΟΝΙΚΗΣ

ΜΕΤΑΦΡΑΣΗ :
ΕΠΕΞΕΡΓΑΣΙΑ ΦΩΝΗΣ ΚΑΙ ΕΙΚΟΝΑΣ

SPEECH AND IMAGE PROCESSING

ΚΑΚΑΝΑ ΘΕΟΦΑΝΗ
Α.Μ : 3527

ΕΠΟΠΤΗΣ ΚΑΘΗΓΗΤΗΣ
Δρ.ΚΩΝΣΤΑΝΤΑΡΑΣ ΑΝΤΩΝΙΟΣ

ΧΑΝΙΑ, ΙΑΝΟΥΑΡΙΟΣ 2012

Εισαγωγή

Επεξεργασία εικόνας ονομάζεται οποιαδήποτε μορφή επεξεργασίας , η οποία καλύπτει τις ακόλουθες κατηγορίες:

- Επεξεργασία εικόνας: Είσοδος: Εικόνα → Έξοδος: Εικόνα
- Ανάλυση εικόνας: Είσοδος: Εικόνα → Έξοδος: Αποτελέσματα μετρήσεων
- Κατανόηση εικόνας: Είσοδος: Εικόνα → Έξοδος: Προσδιορισμός στοιχείων υψηλού επιπέδου

Εφαρμογές της επεξεργασίας εικόνας

- Γραφιστική
- δεδομένων
- Επεξεργασία εικόνας για ιατρικούς σκοπούς
- Ανάλυση μικροσκοπικών παρατηρήσεων
- Ταυτοποίηση προσώπων (σύγκριση στα χαρακτηριστικά προσώπου, δακτυλικών αποτυπωμάτων κτλ.)
- Εφαρμογές Τεχνητής Νοημοσύνης: Μηχανική Όραση (Machine vision)

Η **ψηφιακή επεξεργασία εικόνας** αποτελεί έναν ευρύ επιστημονικό κλάδο που αναπτύχθηκε με την ραγδαία εξέλιξη των υπολογιστών. Ο όρος εικόνα χρησιμοποιείται ευρύτερα από την απλή απεικόνιση ενός σκηνικού και περιλαμβάνει την αποτύπωση κάθε είδους πληροφοριών. Τα υπερηχογραφήματα, οι μαγνητικές τομογραφίες, οι δορυφορικές φωτογραφίες κ.α. μπορούν να επεξεργαστούν ως ψηφιακές εικόνες.

Περίληψη

Το θέμα της πτυχιακής εργασίας διαπραγματεύεται αναλυτικά τεχνικές της επεξεργασίας εικόνας μέσω του MATLAB καθώς επίσης παρατίθενται και πολλά παραδείγματα και εφαρμογές. Ανατρέχουμε στο ιστορικό και στην εξέλιξη των ηλεκτρονικών υπολογιστών και κατ' επέκταση και της επεξεργασίας σημάτων και εικόνας. Αναλύουμε από απλές πρακτικές, όπως για παράδειγμα εμφάνιση και εκτύπωση εικόνων, μέχρι πιο περίπλοκες όπως το φιλτράρισμα και η απομόνωση του θορύβου.

Γενικά οι εφαρμογές είναι απλές αλλά εξαρτάται από το είδος της επεξεργασίας που θέλουμε να κάνουμε. Συναντάμε επίσης τις έννοιες του Ιστογράμματος και του crispening και κάνουμε Ανίχνευση Γραμμών χρησιμοποιώντας την Μετατροπή Hough. Εν κατακλείδι παρουσιάζεται ένας εναλλακτικός τρόπος επεξεργασίας εικόνας απλός στην εφαρμογή και πολύ χρηστικός.

Summary

The theme of the thesis deals in detail the techniques of image processing using the MATLAB and also listed several examples and applications .We look back in history and the evolution of computers and , by extension, the image and signal processing.We analyze by simple practices, such as developing and printing images, up to more complex such as filtering and isolation of noise.

Generally applications are simple but depends on the type of process we want to do.We meet well as the concepts of the histogram and crispening effect and do line detection using the Hough Convert. In conclusion we present an alternative way of image processing simple to implement and very utilitarian.

Περιεχόμενα

Ψηφιακή επεξεργασία εικόνας.....	6
Ιστορικό.....	6
Στόχοι.....	7
Ανάγνωση δεδομένων εικόνας.....	7
Γραφή δεδομένων εικόνας.....	8
Καθορισμός Επιπρόσθετης Μορφής-Συγκεκριμένες παράμετροι.....	8
Ανάγνωση και εγγραφή δυαδικών εικόνων σε μορφή 1-BIT.....	8
Μετατροπή μορφής αρχείου γραφικής παράστασης.....	9
Χρησιμοποιώντας την imshow για επίδειξη εικόνων.....	10
Καθορίζοντας την αρχική μεγέθυνση εικόνας.....	11
Έλεγχος της εμφάνισης του γραφήματος.....	12
Χρησιμοποιώντας το Εργαλείο Εικόνας(Image Tool) για εξερεύνηση εικόνων.....	14
Ανοίγοντας το εργαλείο εικόνας.....	14
Καθορίζοντας την αρχική μεγέθυνση εικόνας(Initial Image Magnification).14	
Καθορίζοντας τον Χάρτη χρωμάτων(Colormap).....	14
Εισάγοντας δεδομένα εικόνας από το περιβάλλον εργασίας.....	16
Εξαγωγή δεδομένων εικόνας στο περιβάλλον εργασίας.....	17
Χρήση της συνάρτησης getimage για την εξαγωγή δεδομένων εικόνας....	17
Κλείνοντας το Εργαλείο εικόνας.....	18
Εκτύπωση της εικόνας με το Image Tool.....	18
Χρησιμοποίηση του βοηθήματος Πλοήγησης του Image Tool.....	18
Παίρνοντας πληροφορίες για τα pixel μίας εικόνας.....	19
Μέτρηση χαρακτηριστικών σε μια εικόνα.....	21
Εξαγωγή Τελικού σημείου(Endpoint) και δεδομένων απόστασης(Distance Data).....	22
Προσαρμογή της εμφάνισης του Εργαλείου Απόστασης(Distance Tool)...22	
Συλλέγοντας πληροφορίες για μια εικόνα.....	23
Ρύθμιση της αντίθεσης και της φωτεινότητας μιας εικόνας.....	24
Προβολή πολλαπλών εικόνων.....	25
Εμφάνιση πολλαπλών εικόνων στο ίδιο γράφημα.....	25
Ειδικές Τεχνικές Εμφάνισης.....	26
Προσθέτοντας μπάρα χρωμάτων.....	26
Εμφανίζοντας πολλαπλά πλαίσια εικόνας με μία φορά.....	27
Χωρικοί μετασχηματισμοί.....	28

Αλλαγή του μεγέθους μιας εικόνας.....	28
Καθορισμός του Μεγέθους της Εικόνας Εξόδου.....	28
Περιστρέφοντας μια Εικόνα.....	28
Καθορισμός της Μεθόδου Παρεμβολής.....	29
Περικοπή Εικόνας.....	30
Επιλογή σημείων ελέγχου.....	30
Εξαγωγή Σημείων Ελέγχου στο Περιβάλλον Εργασίας.....	32
Γραμμικό Φιλτράρισμα και Φίλτρο Σχεδίασης.....	32
Φιλτράρισμα χρησιμοποιώντας την imfilter.....	33
Χρησιμοποιώντας Προκαθορισμένους Τύπους Φίλτρων.....	33
Μορφολογικές λειτουργίες.....	34
Διαστολή και Διάβρωση(Dilation and Erosion).....	35
Διαβρώνοντας μια εικόνα.....	35
Συνδιάζοντας Διαστολή και Διάβρωση.....	36
Μορφολογικό Ανοιγμα.....	36
Οι βασικές συναρτήσεις Διαστολής και Διάβρωσης.....	37
Σκελετοποίηση.....	37
Περιμετρικός προσδιορισμός.....	38
Ανάλυση και Ενίσχυση των εικόνων.....	38
Παίρνοντας πληροφορίες για την αξία των pixel και Στατιστική.....	39
Εμφανίζοντας ένα Περιμετρικό Γράφημα Δεδομένων Εικόνας.....	41
Δημιουργώντας το Ιστόγραμμα Εικόνας.....	42
Αναλύοντας μια εικόνα.....	44
Ανίχνευση αιχμών.....	45
Εξιχνίαση Ορίων.....	49
Ανίχνευση Γραμμών χρησιμοποιώντας την Μετατροπή Hough.....	54
Αναλύοντας την Υφή μιας εικόνας.....	61
Συναρτήσεις Υφής.....	62
Προσαρμογή της Εντασης.....	64
Προσαρμογή των τιμών Εντασης σε συγκεκριμένο Εύρος τιμών.....	64
Καθορισμός των Ορίων Προσαρμογής.....	65
Εξισορρόπηση Ιστογράμματος.....	65
Απομάκρυνση Θορύβου.....	66
Φιλτράρισμα Ενδιάμεσου.....	66
Επεξεργασία με βάση την ROI.....	70
Καθορισμός της Περιοχής Ενδιαφέροντος(ROI).....	70
Επιλέγοντας μια Πολυγωνική ROI Διαδραστικά.....	70
Φιλτράροντας μια ROI.....	73
Φιλτράροντας μια Περιοχή της Εικόνας.....	73

Ξεθάμπωμα με Κανονικοποιημένο Φίλτρο.....	74
Ξεθάμπωμα με Τυφλό Αλγόριθμο Αποσυνέλιξης.....	76
Χρησιμοποιώντας την συνάρτηση deconvblind για το ξεθάμπωμα εικόνας.....	77
Λειτουργίες "συρώμενης γειτονιάς".....	82
Εφαρμογή Γραμμικού και Μη Γραμμικού Φιλτραρίσματος.....	82
Επίλογος.....	85
Βιβλιογραφία.....	85

Ψηφιακή επεξεργασία εικόνας

Η ψηφιακή επεξεργασία εικόνας είναι η χρήση αλγορίθμων για να παρουσιάσουν την επεξεργασία εικόνας σε ψηφιακές εικόνες. Ως υποεπίπεδο της ψηφιακής επεξεργασίας σήματος η ψηφιακή επεξεργασία εικόνας έχει πολλά πλεονεκτήματα σε σχέση με την αναλογική επεξεργασία, επιτρέπει ένα πολύ μεγαλύτερο εύρος αλγορίθμων να εφαρμοστούν σαν

δεδομένα εισόδου, και μπορεί να αποτρέψει προβλήματα όπως η συγκέντρωση θορύβου και διαστρέβλωση σημάτων κατά τη διάρκεια της επεξεργασίας.

Ιστορικό

Πολλές από τις τεχνικές της ψηφιακής επεξεργασίας εικόνας, αναπτύχθηκαν κατά την δεκαετία του '60 στο Jet Propulsion Laboratory, MIT, Bell Labs, του Πανεπιστημίου του Maryland και σε μερικά ακόμα μέρη, με εφαρμογή στις δορυφορικές εικόνες, πρότυπα μετατροπής τηλεφραφικής φωτογραφίας, σε ιατρική απεικόνιση, στα τηλέφωνα με χρήση βίντεο, σε αναγνώριση χαρακτήρων και ενίσχυση(απορρόφηση?) φωτογραφιών. Αλλά και το κόστος της επεξεργασίας ήταν αρκετά υψηλό με τον εξοπλισμό των υπολογιστών εκείνης της εποχής.

Κατά την δεκαετία του '70, η ψηφιακή επεξεργασία εικόνας πολλαπλασιάστηκε καθώς φθηνότεροι υπολογιστές και εξειδικευμένο hardware έγιναν διαθέσιμα. Οι εικόνες μπορούσαν πλέον να υποβληθούν σε περαιτέρω επεξεργασία σε πραγματικό χρόνο, για κάποια συνηθισμένα προβλήματα όπως η μετατροπή τηλεοπτικών προτύπων. Δεδομένου ότι οι υπολογιστές γενικής χρήσης έγιναν γρηγορότεροι άρχισαν να αναλαμβάνουν το ρόλο του υλικού σε όλες αλλά και στις περισσότερο εξειδικευμένες και εντατικά υπολογιστικές διαδικασίες.

Κατά την δεκαετία του 2000 με γρήγορους πλέον υπολογιστές και επεξεργαστές σημάτων διαθέσιμους, η ψηφιακή επεξεργασία εικόνας αναδείχθηκε στην πιο κοινή μορφή επεξεργασίας εικόνων, και ευρύτερα χρησιμοποιούμενη όχι μόνο επειδή είναι η πιο ευπροσάρμοστη μέθοδος αλλά επίσης και η πιο οικονομική.

Στόχοι

Η ψηφιακή επεξεργασία εικόνας επιτρέπει την χρήση πολλών περισσότερων σύνθετων αλγορίθμων για επεξεργασία εικόνας και ως εκ τούτου μπορεί να προσφέρει και τα δύο, περιπλοκότερη απόδοση στους απλούς στόχους και την εφαρμογή μεθόδων οι οποίες θα ήταν αδύνατες με τα αναλογικά μέσα .Πιο συγκεκριμένα , η ψηφιακή επεξεργασία εικόνας είναι η μοναδική πρακτική τεχνολογία για:

- Ταξινόμηση
- Εξαγωγή χαρακτηριστικών
- Αναγνώριση σχεδίων
- Προβολή
- Ανάλυση σημάτων πολυ-κλίμακας

Μερικές τεχνικές που χρησιμοποιούνται στην ψηφιακή επεξεργασία εικόνας περιλαμβάνουν:

- Ανάλυση κύριων τμημάτων
- Ανάλυση ανεξάρτητων τμημάτων
- Χάρτες αυτοοργάνωσης
- Κρυμμένα μοντέλα Markov
- Νευρωνικά δίκτυα

Ανάγνωση δεδομένων εικόνας

Η εντολή `imread` διαβάζει μια εικόνα από κάθε υποστηριζόμενο αρχείο γραφικής αναπαράστασης, σε οποιαδήποτε από τα υποστηριγμένα βάθη bit. Τα περισσότερα αρχεία απεικόνισης χρησιμοποιούν 8 bit για να αποθηκεύσουν τις τιμές των pixel. Όταν αυτά διαβάζονται στη μνήμη, το MATLAB τα αποθηκεύει ως κατηγορία `uint8`. Για τις μορφές αρχείου που υποστηρίζουν δεδομένα των 16 bit, PNG και TIFF, το MATLAB αποθηκεύει τις εικόνες ως κατηγορία `uint16`.

Παράδειγμα: `RGB = imread('football.jpg');`

Εγγραφή δεδομένων εικόνας

Η εντολή **imwrite** 'γράφει' μια εικόνα σε ένα αρχείο γραφικής παράστασης σε μια από τις υποστηριζόμενες μορφές .Η βασική σύνταξη της **imwrite** χρησιμοποιεί το όνομα της εικόνας και ένα όνομα αρχείου .Εάν συμπεριληφθεί μια επέκταση στο όνομα αρχείου, το Matlab παράγει την επιθυμητή μορφή αρχείου από αυτό.

Παράδειγμα : `imwrite(X,map,'moon2.bmp');`

Καθορισμός Επιπρόσθετης Μορφής- Συγκεκριμένες παράμετροι

Όταν χρησιμοποιήσετε την **imwrite** με μερικά σχήματα γραφικής παράστασης, μπορείτε να καθορίσετε πρόσθετες παραμέτρους .Για παράδειγμα με PNG αρχεία, μπορείτε να διευκρινίσετε το βάθος των bit σαν πρόσθετη παράμετρο. Αυτό το παράδειγμα γράφει μια εικόνα A σε JPEG αρχείο χρησιμοποιώντας μια πρόσθετη παράμετρο για να καθορίσει την ποιοτική παράμετρο συμπίεσης.

Παράδειγμα : `imwrite(IMG, 'myfile.jpg', 'Quality', 100);`

Ανάγνωση και εγγραφή δυαδικών εικόνων σε μορφή 1-BIT

Σε ορισμένες μορφές αρχείου, μια δυαδική εικόνα μπορεί να αποθηκευτεί σε μορφή ενός bit .Εάν η μορφή αρχείου το υποστηρίζει το MATLAB γράφει τις δυαδικές εικόνες σαν εικόνες ενός bit εξ ορισμού .Όταν διαβάζουμε μια δυαδική εικόνα σε μορφή ενός bit , το MATLAB το αναπαριστά στην επιφάνεια εργασίας σαν λογική σειρά .Αυτό το παράδειγμα διαβάζει μια δυαδική εικόνα και την γράφει σαν αρχείο TIFF .

Επειδή η μορφή TIFF υποστηρίζει εικόνες ενός bit, το αρχείο είναι γραμμένο σε δίσκο σε μορφή του ενός bit.

Παράδειγμα : `BW = imread('moon.jpg');`

`imwrite(BW,'test.tif');`

Μετατροπή μορφής αρχείου γραφικής παράστασης

Για να αλλάξουμε τη μορφή του γραφήματος μιας εικόνας, χρησιμοποιούμε την **imread** για να εισάγουμε την εικόνα στο περιβάλλον εργασίας του Matlab και κατόπιν την εντολή **imwrite** για να την εμφανίσουμε διευκρινίζοντας την κατάλληλη μορφή του αρχείου.

Για να απεικονιστεί, το παράδειγμα χρησιμοποιεί την **imread** για να διαβάσει μια εικόνα bitmap (BMP) μορφής στο περιβάλλον εργασίας του MATLAB. Έπειτα γράφει την bitmap εικόνα σε έναν φάκελο χρησιμοποιώντας σχήμα PNG (Portable Network Graphics-φορητού δικτύου γραφικών).

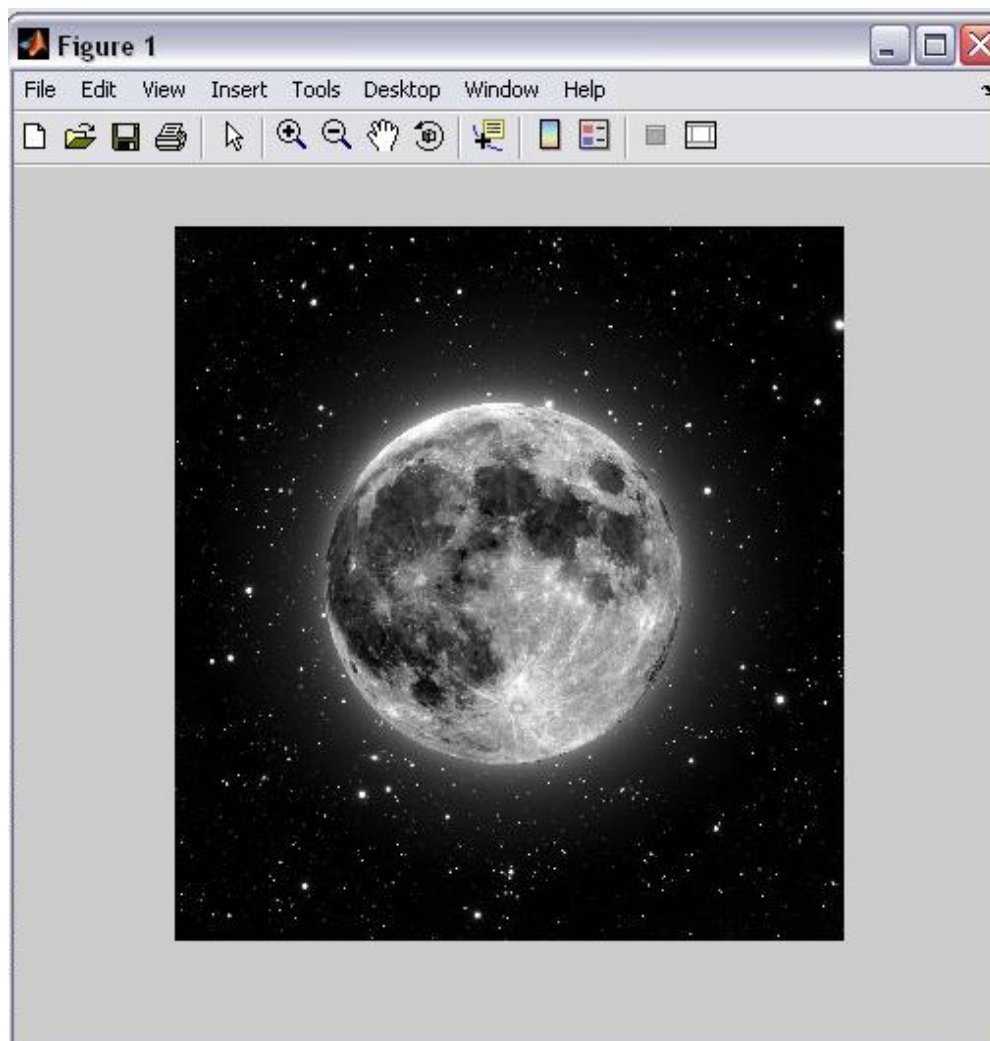
Παράδειγμα : `bitmap = imread('moon.jpg','jpg');`

`imwrite(bitmap,'moon2.png','png');`

Χρησιμοποιώντας την imshow για επίδειξη εικόνων

Μπορείτε να χρησιμοποιήσετε την εντολή **imshow** για να επιδείξετε μια εικόνα που έχει ήδη εισαχθεί στο περιβάλλον του Matlab ή για να επιδείξετε μια εικόνα αποθηκευμένη σε κάποιο αρχείο γραφικής παράστασης. Επί παραδείγματι, ο παρακάτω κώδικας διαβάζει μια εικόνα από το MATLAB και μετά την επιδεικνύει σε παράθυρο γραφήματος του MATLAB.

Παράδειγμα : moon= imread('moon.tif');
imshow (moon);



Καθορίζοντας την αρχική μεγέθυνση εικόνας

Εξ ορισμού, η **imshow** προσπαθεί να επιδείξει την εικόνα στην ολότητά της μεγεθυμένη 100% (ένα pixel της οθόνης για κάθε pixel της εικόνας). Πάραυτα, εάν μια εικόνα είναι πολύ μεγάλη για να χωρέσει σε ένα παράθυρο γραφήματος και όντας μεγεθυμένη 100% , η imshow κλιμακώνει την εικόνα για να χωρέσει στην οθόνη και εμφανίζει ένα προειδοποιητικό μήνυμα.

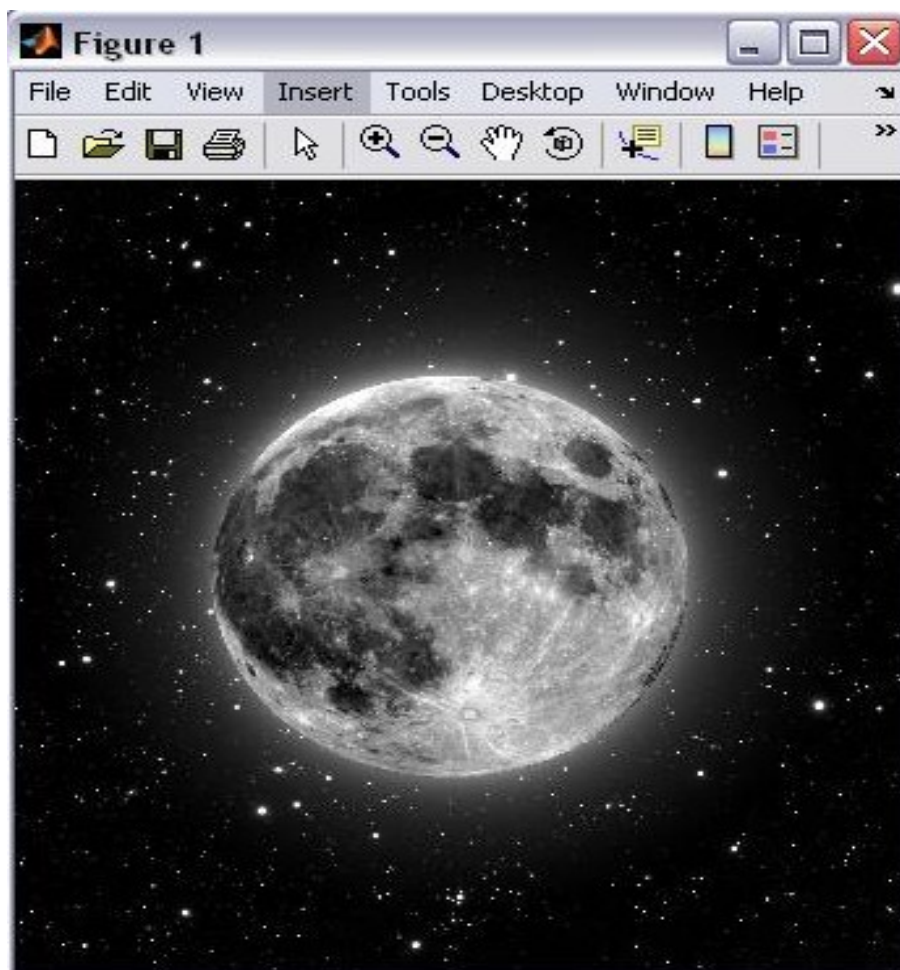
Για να παρακάμψετε την προεπιλεγμένη αρχική συμπεριφορά μεγέθυνσης για μια συγκεκριμένη κλήση της **imshow** , διευκρινίστε την παράμετρο της Αρχικής Μεγέθυνσης (InitialMagnification).Για παράδειγμα, για να δείτε μια εικόνα μεγεθυμένη κατά 50% , χρησιμοποιήστε τον παρακάτω κώδικα.

Παράδειγμα : moon = imread('moon.jpg');
imshow(moon, 'InitialMagnification', 50)

Έλεγχος της εμφάνισης του γραφήματος

Από προεπιλογή, όταν η **imshow** εμφανίζει μια εικόνα σε γράφημα, περιβάλλει την εικόνα με ένα γκρι σύνορο. Μπορείτε να αλλάξετε αυτή την προεπιλογή και να καταστείλετε αυτό το σύνορο χρησιμοποιώντας την παράμετρο 'συνόρου' με την `imshow`, όπως φαίνεται στο ακόλουθο παράδειγμα.

Παράδειγμα : `imshow('moon.jpg','Border','tight')`





Χρησιμοποιώντας το Εργαλείο Εικόνας(Image Tool) για εξερεύνηση εικόνων

Το Image Tool είναι ένα εργαλείο επίδειξης εικόνας που παρέχει επίσης πρόσβαση σε διάφορα άλλα σχετικά εργαλεία, όπως το εργαλείο περιοχής pixel(Pixel Region tool), το εργαλείο πληροφοριών εικόνας(Image Information tool), και το ρυθμιστικό εργαλείο αντίθεσης(Adjust Contrast tool).

Ανοίγοντας το εργαλείο εικόνας

Για να ξεκινήσετε το εργαλείο εικόνας, χρησιμοποιήστε την συνάρτηση **imtool**.

Παράδειγμα : moon = imread('moon.jpg');
imtool(moon)

Καθορίζοντας την αρχική μεγέθυνση εικόνας(Initial Image Magnification)

Όπως η **imshow**, και η συνάρτηση **imtool** επιχειρεί να εμφανίσει μια εικόνα ολόκληρη μεγεθυμένη 100% (ένα pixel της οθόνης για κάθε pixel της εικόνας).

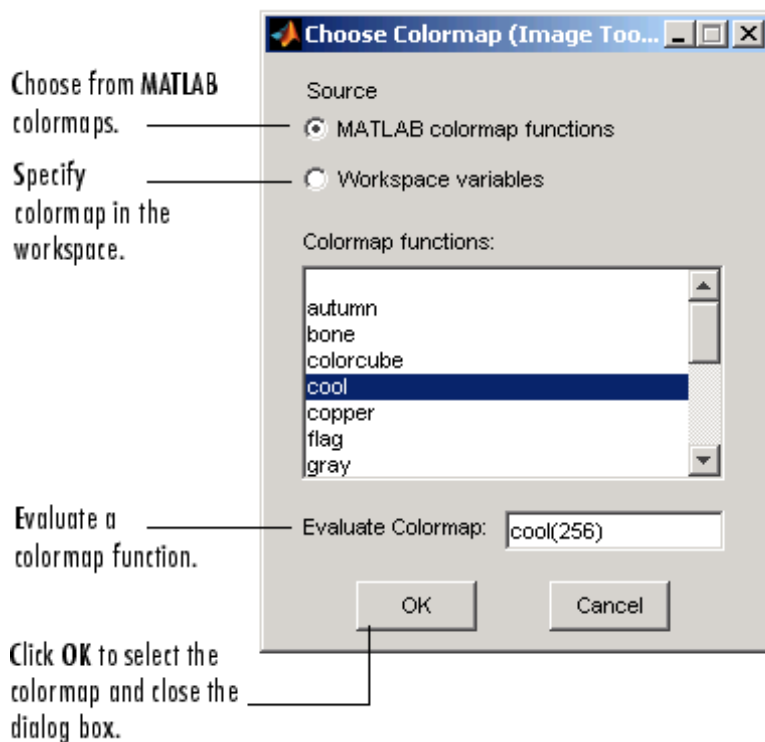
Παράδειγμα : pout = imread('pout.tif');
imtool(pout, 'InitialMagnification', 150)

Καθορίζοντας τον Χάρτη χρωμάτων(Colormap)

Για να καθορίσετε τον χάρτη χρωμάτων, που χρησιμοποιείται για να εμφανίσει μια(**indexed image**) τιμαριθμικά αναπροσαρμοσμένη **εικόνα** ή μια ασπρόμαυρη εικόνα, στο Εργαλείο εικόνας διαλέξτε την επιλογή Διαλέξτε χάρτη χρωμάτων (Choose Colormap) στο menu Εργαλεία.

Αυτή ενεργοποιεί το εργαλείο Διαλέξτε χάρτη χρωμάτων(Choose Colormap tool), που φαίνεται παρακάτω. Χρησιμοποιώντας αυτό το εργαλείο μπορείτε να επιλέξετε έναν από τους χάρτες χρωμάτων του Matlab ή επιλέξετε μια μεταβλητή χάρτη χρώματος από το περιβάλλον του Matlab.

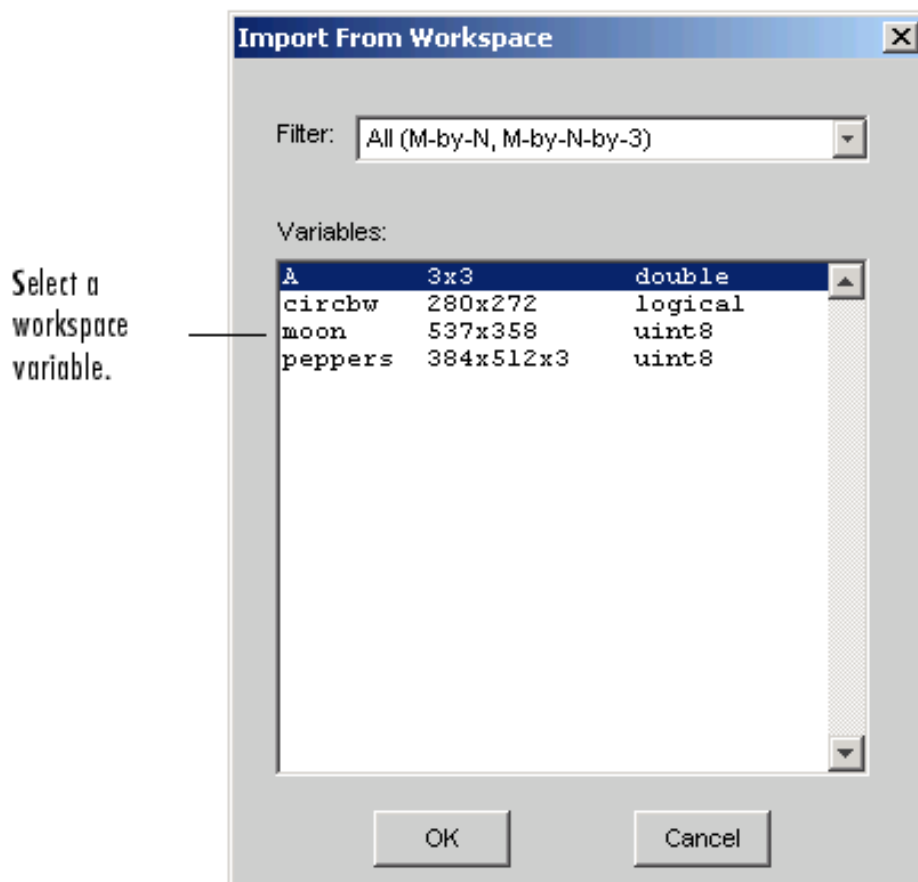
Choose Colormap tool



Εισάγοντας δεδομένα εικόνας από το περιβάλλον εργασίας

Για να εισάγετε δεδομένα εικόνας από το περιβάλλον του MATLAB στο Εργαλείο εικόνας, χρησιμοποιείτε την Εισαγωγή(Import) από την επιλογή Περιβάλλον εργασίας(Workspace) στο φάκελο Εργαλείο εικόνας(Image Tool File) του menu. Στο παράθυρο διαλόγου, που φαίνεται παρακάτω, διαλέγετε την μεταβλητή περιβάλλοντος που θέλετε να εισάγετε μες το περιβάλλον εργασίας.

Εισαγωγή από το χώρο εργασίας του παραθύρου διαλόγου



Εξαγωγή δεδομένων εικόνας στο περιβάλλον εργασίας

Για να εξαγάγετε την εικόνα που εμφανίζεται στο Εργαλείο εικόνας(Image Tool) στην επιφάνεια εργασίας του MATLAB, μπορείτε να χρησιμοποιήσετε την επιλογή Εξαγωγή σε Workspace(Export to Workspace) στο εργαλείο εικόνας του menu Αρχείο εικόνας.

Export Image to Workspace Dialog Box



Χρήση της συνάρτησης `getimage` για την εξαγωγή δεδομένων εικόνας

Μπορείτε επίσης να χρησιμοποιήσετε την συνάρτηση **getimage** για να φέρετε τα δεδομένα εικόνας από το εργαλείο εικόνας(Image Tool) στην επιφάνεια εργασίας του MATLAB. Η **getimage** ανακτά τα δεδομένα εικόνας (CDATA) από το τρέχον αντικείμενο λαβής γραφικών(**Handle Graphics image object**).Επειδή,από προεπιλογή, το Εργαλείο εικόνας δεν κάνει λαβές σε αντικείμενα ορατά, πρέπει να χρησιμοποιήσετε την συνάρτηση Εργαλειοθήκης (Toolbox) **imgca** για να χειριστείτε τους άξονες εικόνας που εμφανίζονται στο Εργαλείο εικόνας.

Παράδειγμα : `moon=getimage(imgca);`

Κλείνοντας το Εργαλείο εικόνας

Για να κλείσετε το παράθυρο του Image Tool, χρησιμοποιήστε το κουμπί Κλείσιμο στην μπάρα τίτλου του παραθύρου, εναλλακτικά επιλέξτε το Κλείσιμο από το εργαλείο του μενού Αρχείο εικόνας. Μπορείτε επίσης να χρησιμοποιήσετε την συνάρτηση **imtool** για να επιστρέψει η λαβή στο Εργαλείο εικόνας και χρησιμοποιήστε την για να κλείσετε το Εργαλείο εικόνας. Όταν κλείσετε το Image Tool, οποιαδήποτε συναφή εργαλεία που είναι ανοικτά επίσης θα κλείσουν.

Παράδειγμα : `imtool close`
`imtool close all`

Εκτύπωση της εικόνας με το Image Tool

Για να εκτυπώσετε την εικόνα που εμφανίζεται στο Image Tool, επιλέξτε την Εκτύπωση σε σχήμα επιλογή(Print to Figure) απο το menu Αρχείο. Το Image Tool ανοίγει ένα ακόμα γράφημα σε παράθυρο και εμφανίζει την εικόνα. Χρησιμοποιήστε την επιλογή Εκτύπωση από το μενού Αρχείο του γραφήματος για να εκτυπώσετε την εικόνα.

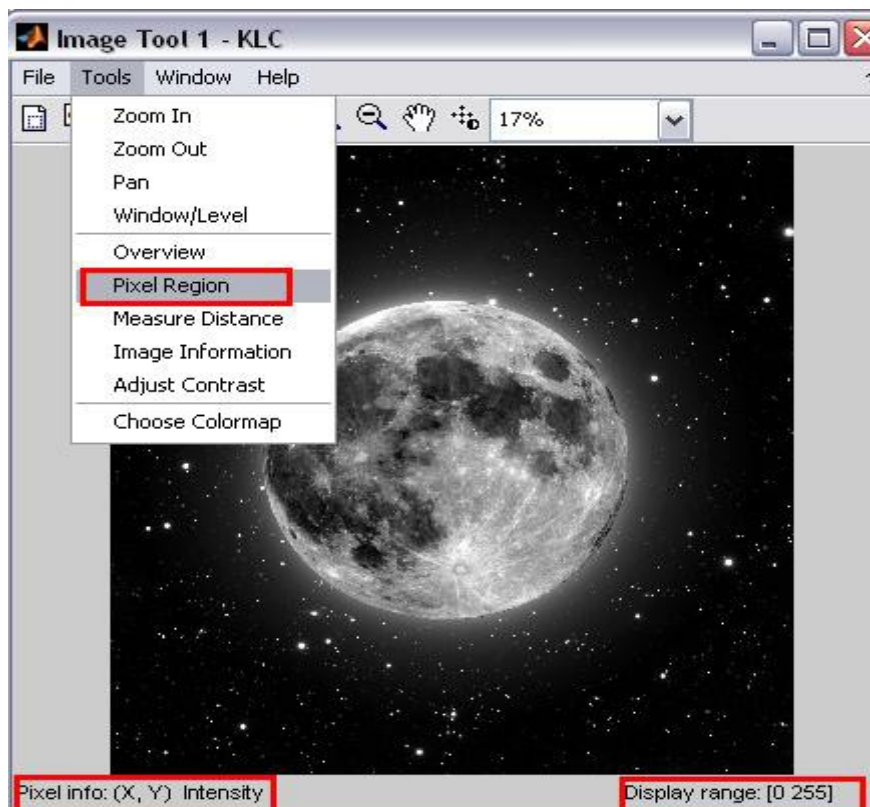
Χρησιμοποίηση του βοηθήματος Πλοήγησης του Image Tool

- Εργαλείο Επισκόπησης(**Overview tool**) : παρέχει μια προβολή του συνόλου της εικόνας για να σας βοηθήσει να καταλάβετε ποιό τμήμα της εικόνας εμφανίζεται στο Image Tool.
- **Pan tool** : σας επιτρέπει να επιλέξετε και να "σύρετε" την εικόνα που εμφανίζεται και να την μεταφέρετε στο Image Tool.
- Εργαλεία zoom(**Zoom tools**) : για να κάνετε zoom στην εικόνα.
- Κουτί μεγέθυνσης(**Magnification Box**) : σας επιτρέπει να ορίσετε την ακριβή μεγέθυνση της εικόνας.



Παίρνοντας πληροφορίες για τα pixel μίας εικόνας

- Εργαλείο πληροφοριών pixel(**Pixel Information tool**) : εμφανίζει την θέση και την αξία του pixel κάτω από τον κέρσορα στο παράθυρο του Image Tool.
- Εμφάνιση του εργαλείου Εύρους(**Range tool**) : εμφανίζει την σειρά εμφάνισης της εικόνας στο παράθυρο του Image Tool.
- Εργαλείο Περιοχής pixel(**Pixel Region tool**) : εμφανίζει μια πολύ κοντινή προβολή των pixels σε συγκεκριμένη περιοχή της εικόνας.



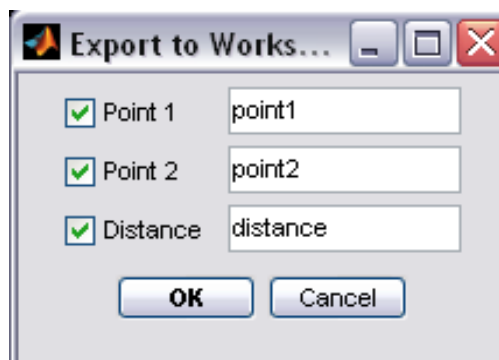
Μέτρηση χαρακτηριστικών σε μια εικόνα

Using the Distance Tool



Εξαγωγή Τελικού σημείου(Endpoint) και δεδομένων απόστασης(Distance Data)

Για να αποθηκεύσετε τις θέσεις του τελικού σημείου και τις πληροφορίες για την απόσταση, πατήστε δεξί-κλικ στο εργαλείο Απόστασης(**Distance tool**) και επιλέξτε το ρίχει Αντιγραφή από την επιλογή Πληροφορίες από το μενού περιβάλλοντος.Το εργαλείο Απόστασης(**Distance tool**) ανοίγει το παράθυρο διαλόγου της Εξαγωγής στο περιβάλλον εργασίας(**Export to Workspace**). Μπορείτε να χρησιμοποιήσετε το παράθυρο διαλόγου για να καθορίσετε τα ονόματα των μεταβλητών που χρησιμοποιούνται για την αποθήκευση αυτών των πληροφοριών.



Προσαρμογή της εμφάνισης του Εργαλείου Απόστασης(Distance Tool)

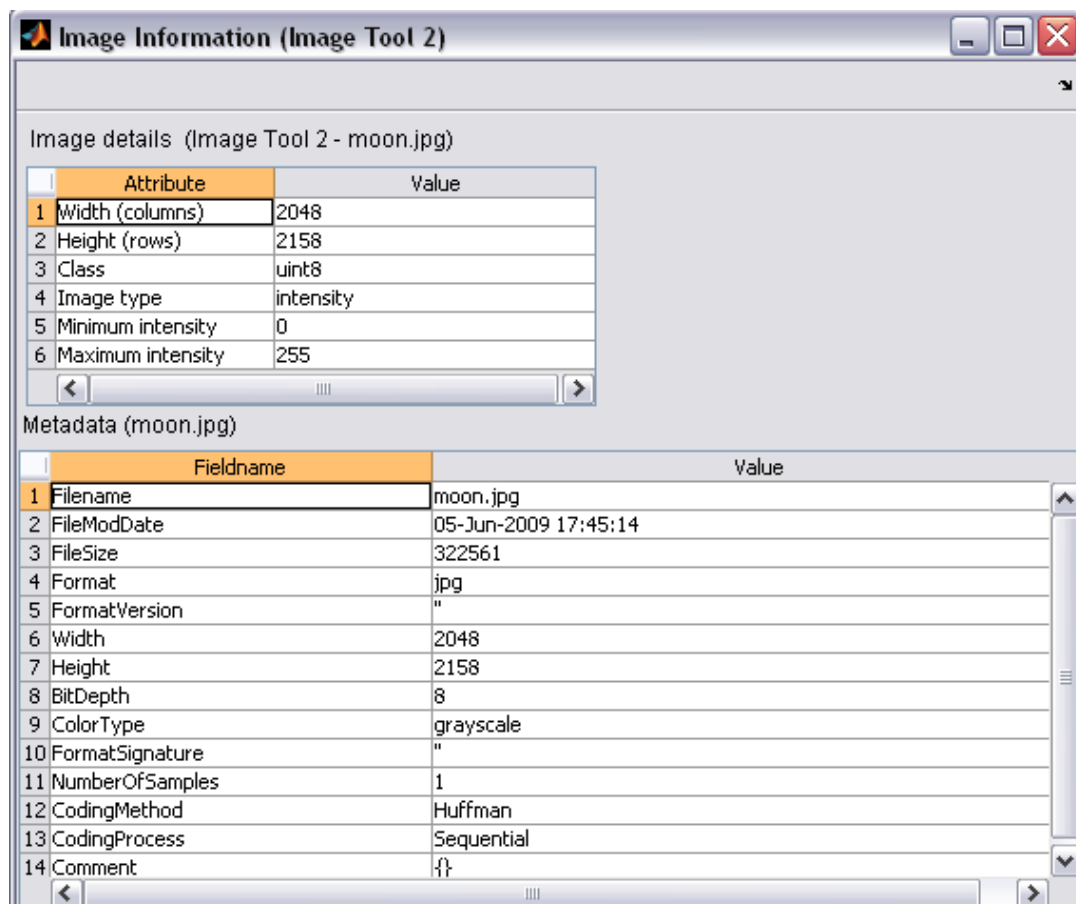
Χρησιμοποιώντας το μενού του Εργαλείου Απόστασης,μπορείτε να προσαρμόσετε πολλές πτυχές του, όσον αφορά την εμφάνιση και την συμπεριφορά του,συμπεριλαμβανομένων:

- Εναλλαγή στην ετικέτα του Distance Tool σε on και off χρησιμοποιώντας την επιλογή ετικέτας Εμφάνισε Απόσταση(**Show Distance**)

- Αλλαγή του χρώματος που χρησιμοποιούταν για να εμφανίζει την γραμμή του Distance tool, χρησιμοποιώντας την επιλογή χρώματος γραμμής(**Set line color**)
- Περιορισμό στην κίνηση του εργαλείου είτε οριζόντια είτε κάθετα χρησιμοποιώντας την επιλογή **Constrain drag**.
- Διαγραφή του αντικειμένου distance tool χρησιμοποιώντας την επιλογή Διαγραφή.
- Δεξί κλικ στο Distance tool για πρόσβαση στο μενού περιβάλλοντος.

Συλλέγοντας πληροφορίες για μια εικόνα

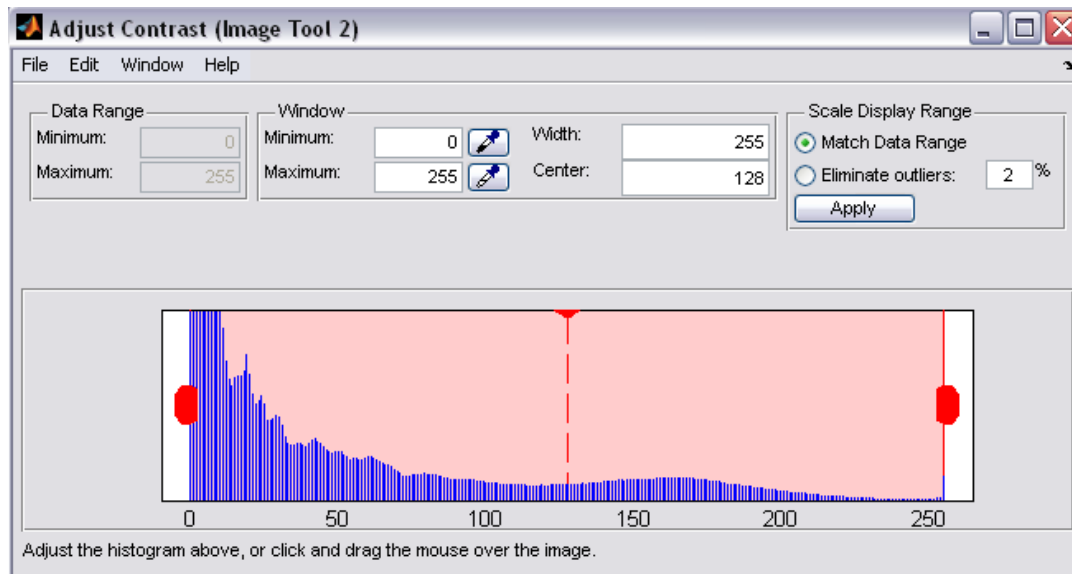
Εκκινήστε το εργαλείο, Πληροφορίες Εικόνας(**Image Information**), κλικάροντας στο κουμπί Image Information που βρίσκεται στην μπάρα εργαλείων του Image Tool.Εναλλακτικά μπορείτε να επιλέξετε την Image Information από το menu Εργαλεία(Tools) στο Image Tool.



Ρύθμιση της αντίθεσης και της φωτεινότητας μιας εικόνας

Για να ρυθμίσετε την αντίθεση και τη φωτεινότητα της εικόνας που προβάλλεται στο Εργαλείο εικόνας(Image Tool), χρησιμοποιήστε το εργαλείο Ρύθμιση αντίθεσης(**Adjust Contrast**).

Εργαλείο Ρύθμισης Αντίθεσης

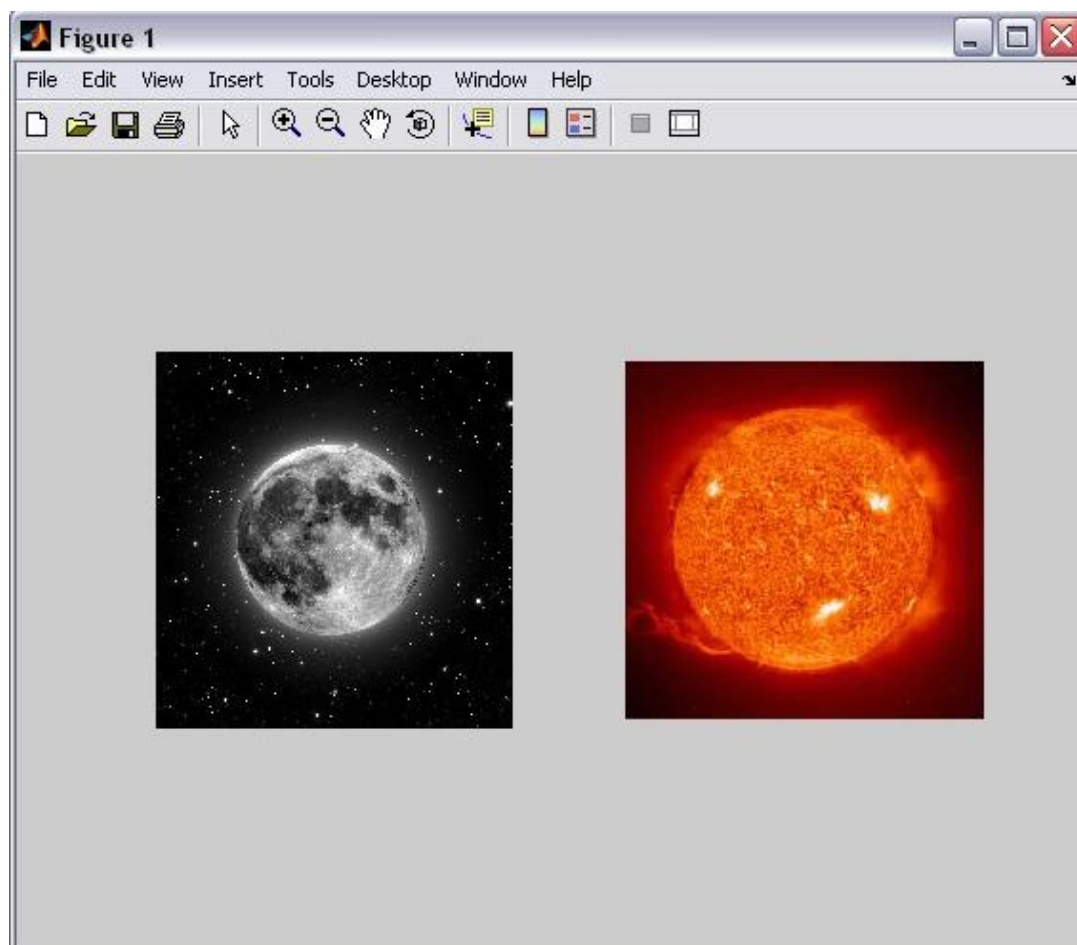


Προβολή πολλαπλών εικόνων

Εμφάνιση πολλαπλών εικόνων στο ίδιο γράφημα

Μπορείτε να χρησιμοποιήσετε την **imshow** συνάρτηση με την **subplot** συνάρτηση του MATLAB ή την **subimage** συνάρτηση για να εμφανίσετε πολλαπλές εικόνες σε ένα μόνο παράθυρο γραφήματος.

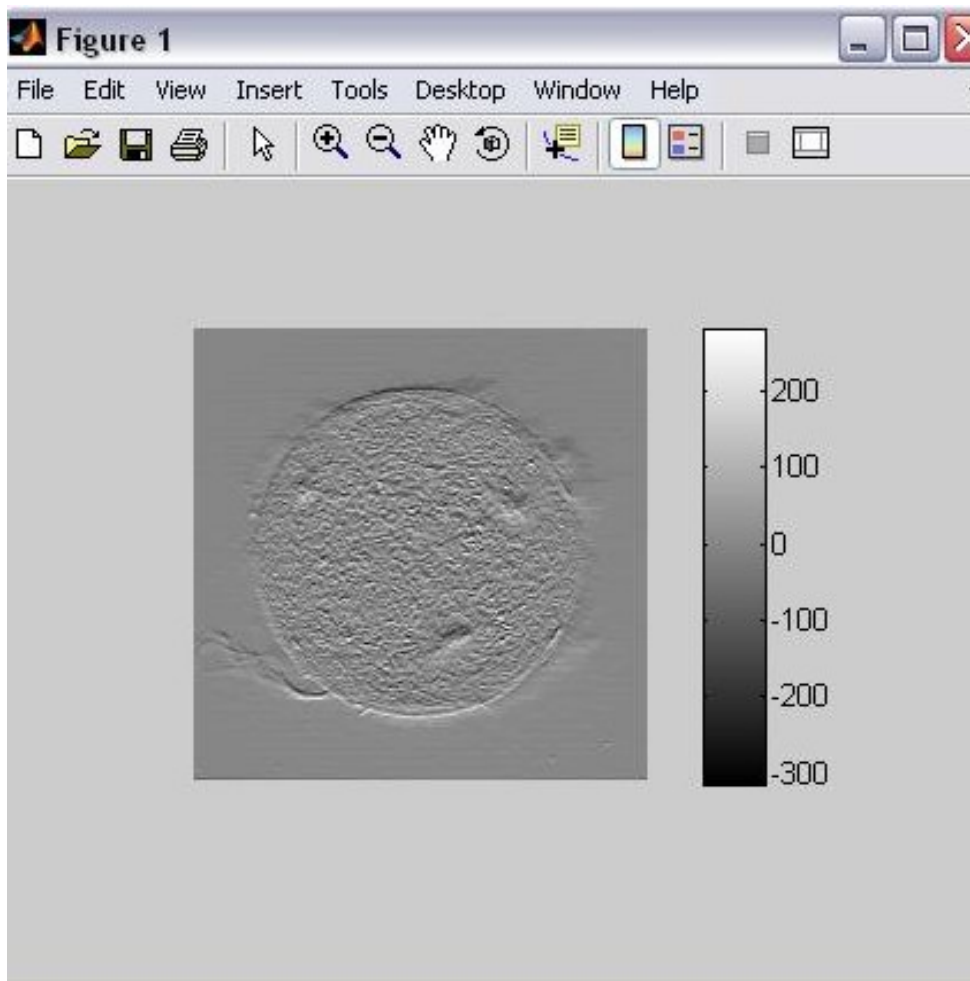
Παράδειγμα : `[X1,map1]=imread('moon.jpg');`
`[X2,map2]=imread('sun.jpg');`
`subplot(1,2,1), imshow(X1,map1)`
`subplot(1,2,2), imshow(X2,map2)`



Ειδικές Τεχνικές Εμφάνισης Προσθέτοντας μπάρα χρωμάτων

Εάν θέλετε να προσθέσετε μπάρα χρωμάτων σε μια εικόνα στο Image Tool, επιλέξτε Εκτύπωση σε Σχήμα(**Print to Figure**) από το μενού Αρχείο του Image Tool. Το Image Tool εμφανίζει την εικόνα σε ξεχωριστά γραφήματα που σε κάθε ένα μπορεί να προστεθεί μπάρα χρωμάτων.

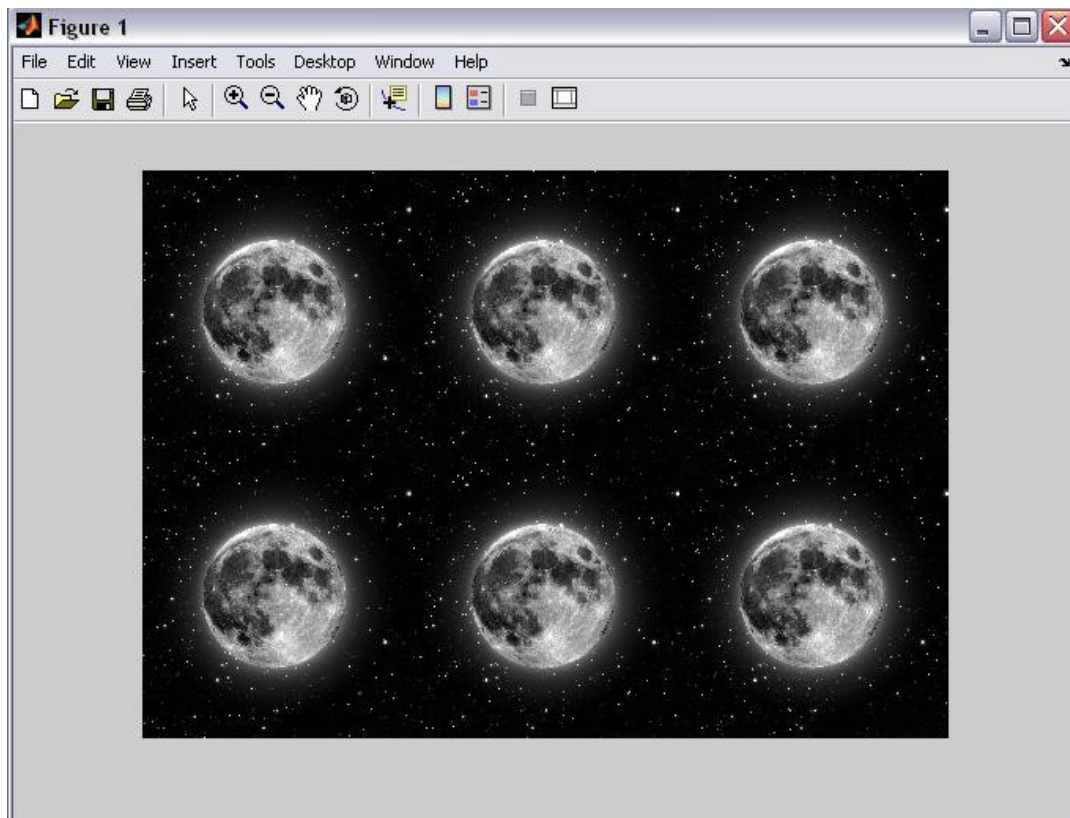
Παράδειγμα : `RGB = imread('sun.jpg');`
`I = rgb2gray(RGB);`
`h = [1 2 1; 0 0 0; -1 -2 -1];`
`I2 = filter2(h,I);`
`imshow(I2,'DisplayRange',[]), colorbar`



Εμφανίζοντας πολλαπλά πλαίσια εικόνας με μία φορά

Για να εμφανίσετε πολλαπλά πλαίσια σε μια σειρά πολλών πλαισίων με μία φορά, χρησιμοποιήστε την συνάρτηση **montage**. Η συνάρτηση αυτή εμφανίζει όλα τα πλαίσια της εικόνας, τακτοποιώντας τα σε ένα ορθογώνιο πλέγμα. Το μοντάζ των εικόνων είναι ένα αντικείμενο της εικόνας. Τα πλαίσια εικόνων μπορεί να είναι σε κλίμακα του γκρι, τιμαριθμικά αναπροσαρμοσμένες, ή εικόνες πραγματικού χρώματος. Αν καθορίσετε το ευρετήριο εικόνων, πρέπει όλα να χρησιμοποιούν τον ίδιο χάρτη χρωμάτων.

Παράδειγμα : `moon = imread('moon.jpg');`
`moonArray = repmat(moon, [1 1 1 6]);`
`montage(moonArray);`



Χωρικοί μετασχηματισμοί

Αλλαγή του μεγέθους μιας εικόνας

Χρησιμοποιώντας την **imresize** εντολή, μπορείτε να καθορίσετε το μέγεθος της εικόνας εξόδου με δύο τρόπους :

1. Με τον καθορισμό του συντελεστή μεγέθυνσης που θα χρησιμοποιηθεί στην εικόνα. Χρησιμοποιήστε
2. Με τον καθορισμό των διαστάσεων της εικόνας εξόδου

```
Παράδειγμα : I = imread('moon.jpg');  
J = imresize(I,1.25);  
imshow(I)  
figure, imshow(J)
```

Καθορισμός του Μεγέθους της Εικόνας Εξόδου

Μπορείτε να καθορίσετε το μέγεθος της εικόνας εξόδου περνώντας ένα διάνυσμα που περιέχει τον αριθμό των γραμμών και των στηλών στην εικόνα εξόδου.

```
Παράδειγμα : I = imread('moon.jpg');  
J = imresize(I,[100 150]);  
imshow(I)  
figure, imshow(J)
```

Περιστρέφοντας μια Εικόνα

Η συνάρτηση **imrotate** δέχεται δύο βασικές παράμετρους :

- την εικόνα που θα περιστραφεί
- την γωνία περιστροφής

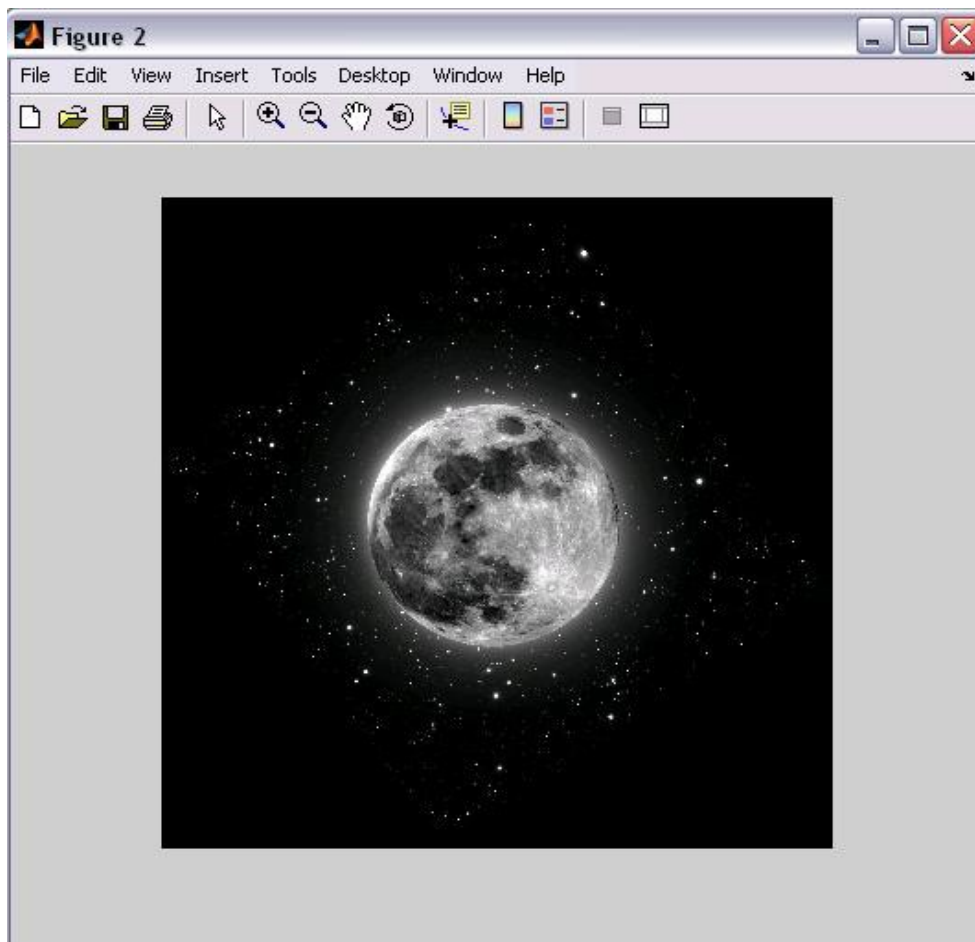
```
Παράδειγμα : J = imrotate(I,35);
```

Καθορισμός της Μεθόδου Παρεμβολής

Εξ ορισμού, η συνάρτηση **imrotate** χρησιμοποιεί την παρεμβολή του κοντινότερου γείτονα για τον προσδιορισμό της αξίας των pixels της εικόνας εξόδου, αλλά μπορείτε να καθορίσετε και άλλες μεθόδους παρεμβολής.

Παράδειγμα :

```
I = imread('moon.jpg');  
J = imrotate(I,35,'bilinear');  
imshow(I)  
figure, imshow(J)
```



Περικοπή Εικόνας

Για να εξάγετε ένα ορθογώνιο τμήμα της εικόνας, χρησιμοποιήστε την εντολή **imcrop**. Η `imcrop` δέχεται και αυτή δύο βασικές παραμέτρους :

- την εικόνα που θα περικοπεί
- τις συντεταγμένες ενός ορθογωνίου που ορίζει την περιοχή κοπής

Παράδειγμα : `imshow moon.jpg`

`I = imcrop;`

`imshow(I);`

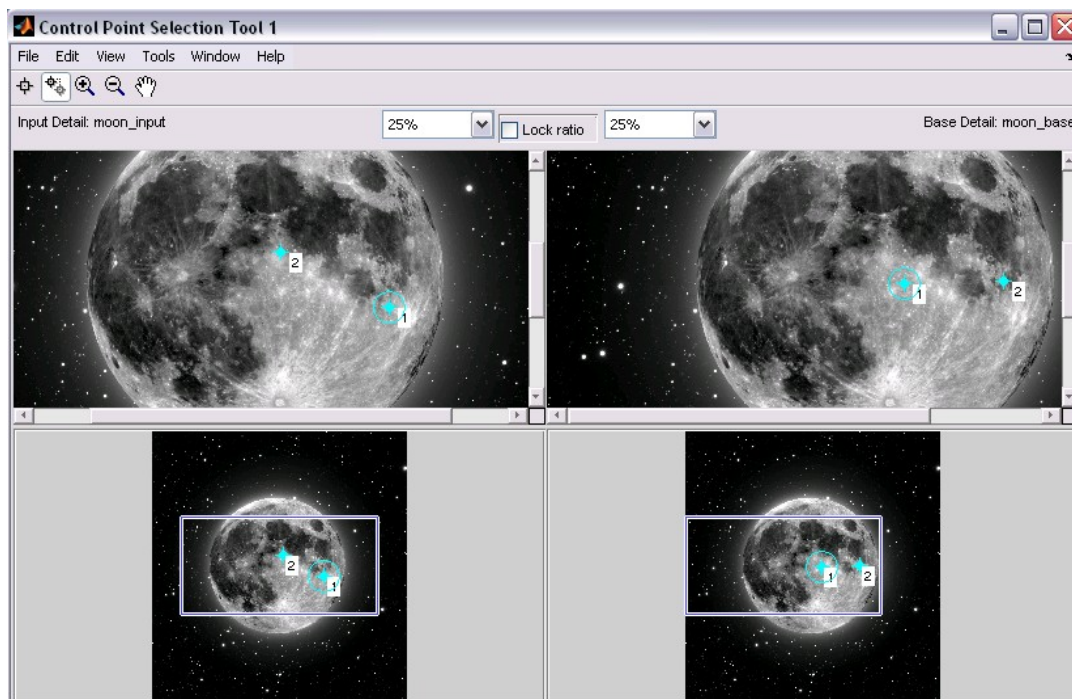
Επιλογή σημείων ελέγχου

Ο καθορισμός σημείων ελέγχου είναι μια διεργασία τεσσάρων σταδίων :

- Εκκινήστε το εργαλείο, καθορίζοντας την εικόνα εισόδου και την βασική εικόνα.
- Χρησιμοποιήστε τα βοηθήματα πλοήγησης για να εξερευνήσετε την εικόνα, ψάχνοντας για οπτικά στοιχεία που μπορείτε να εντοπίσετε και στις δύο εικόνες.
- Η **cpselect** παρέχει πολλούς τρόπους για να πλοηγηθίτε γύρω από την εικόνα, κάνοντας `panning`(τεχνική που δίνει την αίσθηση της κίνησης και της ταχύτητας διατηρώντας το θέμα-αντικείμενο που κινείται ακίνητο στο κάδρο και το `background` κουνιμένο)και ζουμάροντας για να δείτε περιοχές της εικόνας με περισσότερες λεπτομέρειες.
- Καθορίστε τα ζευγάρια που ταιριάζουν στα σημεία ελέγχου στην εικόνα εισόδου και την βασική εικόνα. Σώστε τα σημεία ελέγχου στο περιβάλλον εργασίας του MATLAB.

Παράδειγμα :

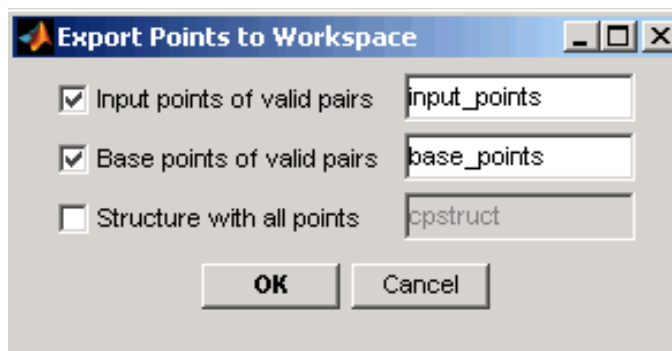
```
moon_base = imread('moon.jpg');  
moon_input = moon_base;  
cpselect(moon_input, moon_base);
```

Εξαγωγή Σημείων Ελέγχου στο Περιβάλλον Εργασίας

Για να σώσετε σημεία ελέγχου στο περιβάλλον εργασίας του MATLAB :

- Επιλέξτε Αρχείο(**File**) από την μπάρα του μενού του εργαλείου Επιλογή Σημείων Ελέγχου
- Επιλέξτε τα Σημεία Εξαγωγής(**Export Points**) από την επιλογή Περιβάλλον Εργασίας. Το εργαλείο Επιλογή Σημείου Ελέγχου εμφανίζει το παρακάτω κουτί διαλόγου :



Γραμμικό Φιλτράρισμα και Φίλτρο Σχεδίασης

Το φιλτράρισμα είναι μια τεχνική για την τροποποίηση ή την ενίσχυση μιας εικόνας. Για παράδειγμα, μπορείτε να φιλτράρετε μια εικόνα για να δώσετε έμφαση σε ορισμένες λειτουργίες ή να αφαιρέσετε άλλες λειτουργίες.

Οι λειτουργίες επεξεργασίας εικόνας υλοποιούνται με φιλτράρισμα συμπεριλαμβάνοντας την εξομάλυνση, το ακόνισμα και βελτίωση των άκρων.

Φιλτράρισμα χρησιμοποιώντας την `imfilter`

Το φιλτράρισμα των εικόνων, είτε με συσχέτιση είτε με συνέλιξη, μπορεί να εκτελεστεί χρησιμοποιώντας την εντολή `imfilter`. Αυτό το παράδειγμα φιλτράρει μια εικόνα με ένα 5-απο-5 φίλτρο που περιέχει ίσα βάρη. Ένα τέτοιο φίλτρο συχνά καλείται και φίλτρο μέσου όρου.

Παράδειγμα :

```
I = imread('moon.jpg');  
h = ones(5,5) / 35;  
I2 = imfilter(I,h);  
imshow(I), title('Original Image');  
figure, imshow(I2), title('Filtered Image')
```



Χρησιμοποιώντας Προκαθορισμένους Τύπους Φίλτρων

Η `fspecial` εντολή παράγει διάφορα είδη από προκαθορισμένα φίλτρα, υπό μορφή πυρήνων συσχέτισης. Το παράδειγμά μας, δείχνει την εφαρμογή ενός φίλτρου όξυνσης και συγκάλυψης (`unsharp masking filter`) σε εικόνα επίπεδου γκρι. Το φίλτρο όξυνσης και συγκάλυψης έχει ως αποτέλεσμα να φτιάχνει τις άκρες και να κάνει την λεπτομέρεια στην εικόνα πιο `crisp`.

(**Crispening** = είναι μια προσαρμογή που μπορούμε να κάνουμε για να εξομαλύνουμε τον θόρυβο σε μια εικόνα)

Παράδειγμα :

```
I = imread('moon.jpg');  
h = fspecial('unsharp');  
I2 = imfilter(I,h);  
imshow(I), title('Original Image')  
figure, imshow(I2), title('Filtered Image')
```



Μορφολογικές λειτουργίες

Η μορφολογία είναι ένα ευρύ σύνολο λειτουργιών επεξεργασίας εικόνας που επιδουκνύει τις εικόνες βασιζόμενη σε σχήματα. Οι μορφολογικές λειτουργίες εφαρμόζουν ένα διαρθρωτικό στοιχείο σε μια εικόνα εισόδου, δημιουργώντας μια εικόνα εξόδου του ίδιου μεγέθους. Οι βασικές μορφολογικές λειτουργίες είναι η διαστολή και η διάβρωση. Σε μια μορφολογική λειτουργία, η αξία του κάθε pixel στην εικόνα εξόδου βασίζεται σε σύγκριση του αντίστοιχου pixel της εικόνας εισόδου με τα γειτονικά του. Διαλέγοντας το μέγεθος και το σχήμα των γειτονικών pixel, μπορείτε να κατασκευάσετε μια μορφολογική λειτουργία που να είναι ευαίσθητη σε συγκεκριμένα σχήματα στην εικόνα εισόδου.

Διαστολή και Διάβρωση(Dilation and Erosion)

Η Διαστολή και η Διάβρωση είναι δύο θεμελιώδης μορφολογικές λειτουργίες. Η Διαστολή προσθέτει pixels στα όρια των αντικειμένων σε μια εικόνα, ενώ η Διάβρωση μετακινεί pixels στα όρια των αντικειμένων. Ο αριθμός των προστιθέμενων ή μετακινούμενων pixels σε μια εικόνα εξαρτάται από το μέγεθος και το σχήμα του δομημένου στοιχείου που χρησιμοποιήθηκε για να επιδειχθεί η εικόνα.

Διαβρώνοντας μια εικόνα

Για να διαβρώσετε μια εικόνα, κάντε χρήση της εντολής **imerode**. Η imrode δέχεται δύο βασικές παραμέτρους :

- Την εικόνα εισόδου προς επεξεργασία (είτε είναι σε αποχρώσεις του γκρι, είτε δυαδική είτε συσκευασμένη δυαδική εικόνα)
- Ένα δομικό στοιχείο αντικειμένου , που επιστρέφεται από την εντολή **strel** ή μια δυαδική μήτρα που καθορίζει την γειτονιά ενός δομικού στοιχείου.

Παράδειγμα :

```
BW1 = imread('moon.jpg');  
SE = strel('arbitrary',eye(13));  
BW2 = imerode(BW1,SE);  
>> imshow(BW1),title('Original Image')  
figure, imshow(BW2),title('Filtered Image')
```



Συνδιάζοντας Διαστολή και Διάβρωση

Η Διαστολή και η Διάβρωση συχνά χρησιμοποιούνται σε συνδιασμό για την εφαρμογή λειτουργιών επεξεργασίας εικόνας. Για παράδειγμα, ο ορισμός ενός μορφολογικού ανοίγματος μιας εικόνας είναι μια διάβρωση ακολουθούμενη από μια διαστολή, χρησιμοποιώντας τα ίδια δομικά στοιχεία και για τις δύο λειτουργίες. Η αντίστοιχη λειτουργία ενός μορφολογικού κλεισίματος μιας εικόνας είναι ακριβώς η αντίστροφη : αποτελείται από διαστολή ακολουθούμενη από διάβρωση με τα ίδια δομικά στοιχεία.

Μορφολογικό Ανοιγμα

Μπορείτε να χρησιμοποιήσετε το μορφολογικό άνοιγμα για να αφαιρέσετε μικρά αντικείμενα από μια εικόνα ενόσω διατηρείται το σχήμα και το μέγεθος μεγαλύτερων αντικειμένων στην εικόνα.

Παράδειγμα :

```
BW1 = imread('moon.jpg');  
SE = strel('rectangle',[40 30]);  
BW2 = imerode(BW1,SE);  
imshow(BW1), title ('Original image')  
figure, imshow (BW2) , title ('Filtered image')
```



Οι βασικές συναρτήσεις Διαστολής και Διάβρωσης

Αυτό το τμήμα περιγράφει δύο κοινές λειτουργίες επεξεργασίας εικόνας που είναι βασισμένες στη Διαστολή και τη Διάβρωση :

1. Σκελετοποίηση
2. Περιμετρικός προσδιορισμός

1.Σκελετοποίηση

Για να μετατρέψετε όλα τα αντικείμενα μιας εικόνας σε γραμμές, χωρίς να μεταβάλλεται η βασική δομή της εικόνας, χρησιμοποιήστε την συνάρτηση **bwmorph** .Αυτή η επεξεργασία είναι γνωστή ως σκελετοποίηση.

Παράδειγμα :

```
BW1 = imread('moon.jpg');  
BW2 = bwmorph(BW1,'skel',Inf);  
imshow(BW1), title ('Original image')  
figure, imshow(BW2), title ('Filtered image')
```



2.Περιμετρικός προσδιορισμός

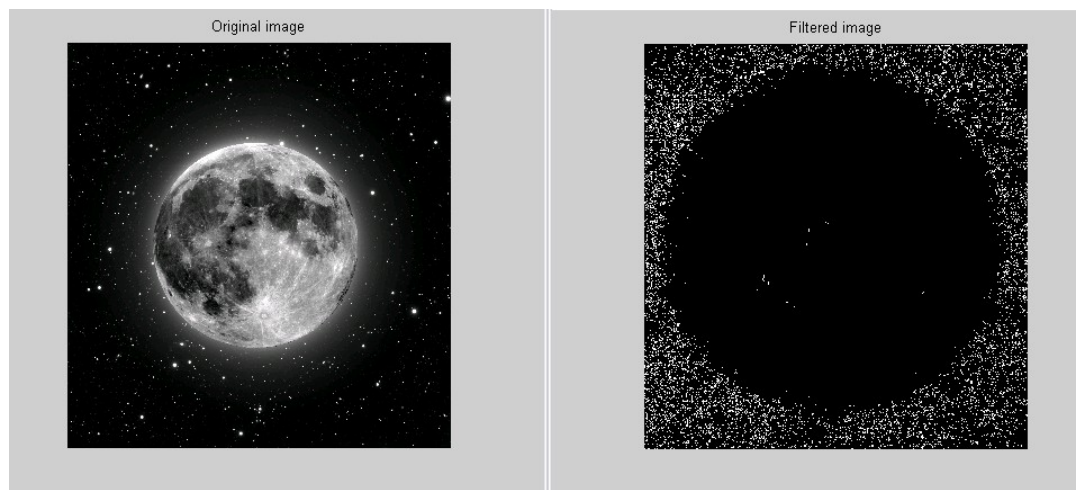
Η συνάρτηση **bwperim** προσδιορίζει τα περιμετρικά pixel των αντικειμένων σε μια δυαδική εικόνα. Κάθε pixel θεωρείται περιμετρικό εάν ικανοποιεί τα παρακάτω δύο κριτήρια :

- Το pixel να είναι ενεργό
- Και τουλάχιστον ένα γειτονικό pixel να είναι απενεργοποιημένο

Για παράδειγμα, ο παρακάτω κώδικας βρίσκει τα περιμετρικά pixel μιας δυαδικής εικόνας σε κυκλωματικό(?) πίνακα.

Παράδειγμα :

```
BW1 = imread('moon.jpg');  
BW2 = bwperim(BW1);  
imshow(BW1), title ('Original image')  
figure, imshow(BW2), title ('Filtered image')
```



Ανάλυση και Ενίσχυση των εικόνων

Αυτό το κεφάλαιο περιγράφει συναρτήσεις που υποστηρίζουν ένα εύρος από σπάντα λειτουργίες επεξεργασίας για ανάλυση και ενίσχυση εικόνων

Παίρνοντας πληροφορίες για την αξία των pixel και Στατιστική

Παρακάτω περιγράφεται πώς μπορείτε να συλλέξετε πληροφορίες για την αξία των δεδομένων που φτιάχνουν μια εικόνα. Τα θέματα που καλύπτονται περιλαμβάνουν :

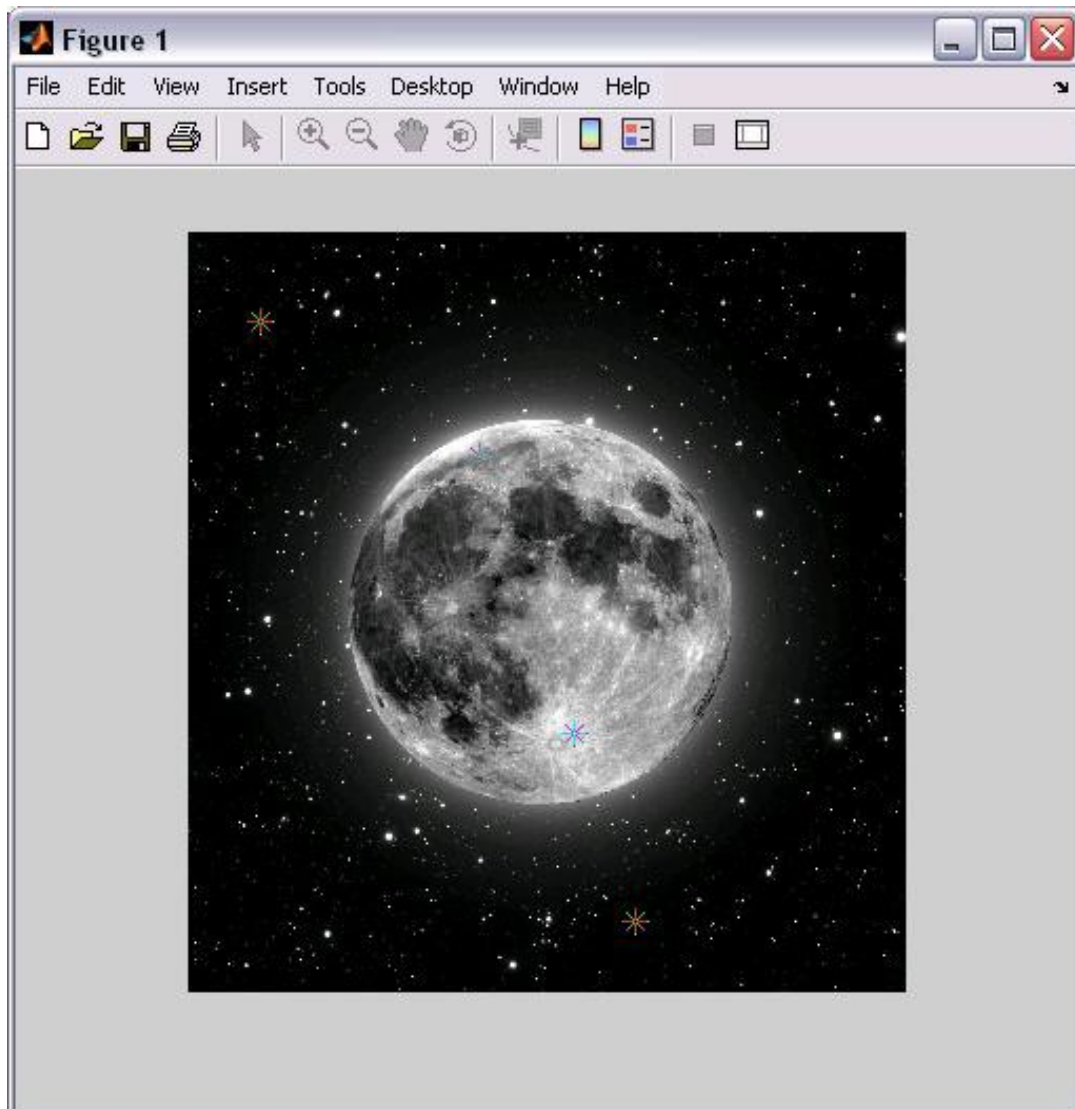
- Παίρνοντας πληροφορίες για τα pixel εικόνας
- Παίρνοντας το Προφίλ Εντασης μιας εικόνας
- Εμφανίζοντας το Περίγραμμα μιας εικόνας
- Δημιουργώντας το Ιστόγραμμα μιας εικόνας
- Συλλέγοντας περιληπτικά Στατιστικά μιας εικόνας
- Υπολογιστικές(?) ιδιότητες για τις περιοχές μιας εικόνας

Εμφανίστε εικόνα `imshow moon.jpg`

Καλείται `imshow`. Όταν καλείται χωρίς ορίσματα εισόδου, η `imshow` συσχετίζεται με την εικόνα στους τρέχον άξονες.

`vals = imshow`

Επιλέξτε τα σημεία που θέλετε να εξετάσετε στην εικόνα κλικάροντας με το ποντίκι. Η `imshow` τοποθετεί αστερίσκο σε κάθε σημείο που επιλέγετε.



Όταν τελειώσετε με την επιλογή σημείων πατήστε ENTER

```
vals =  
173 173 173  
216 216 216  
1 1 1  
1 1 1  
44 44 44
```

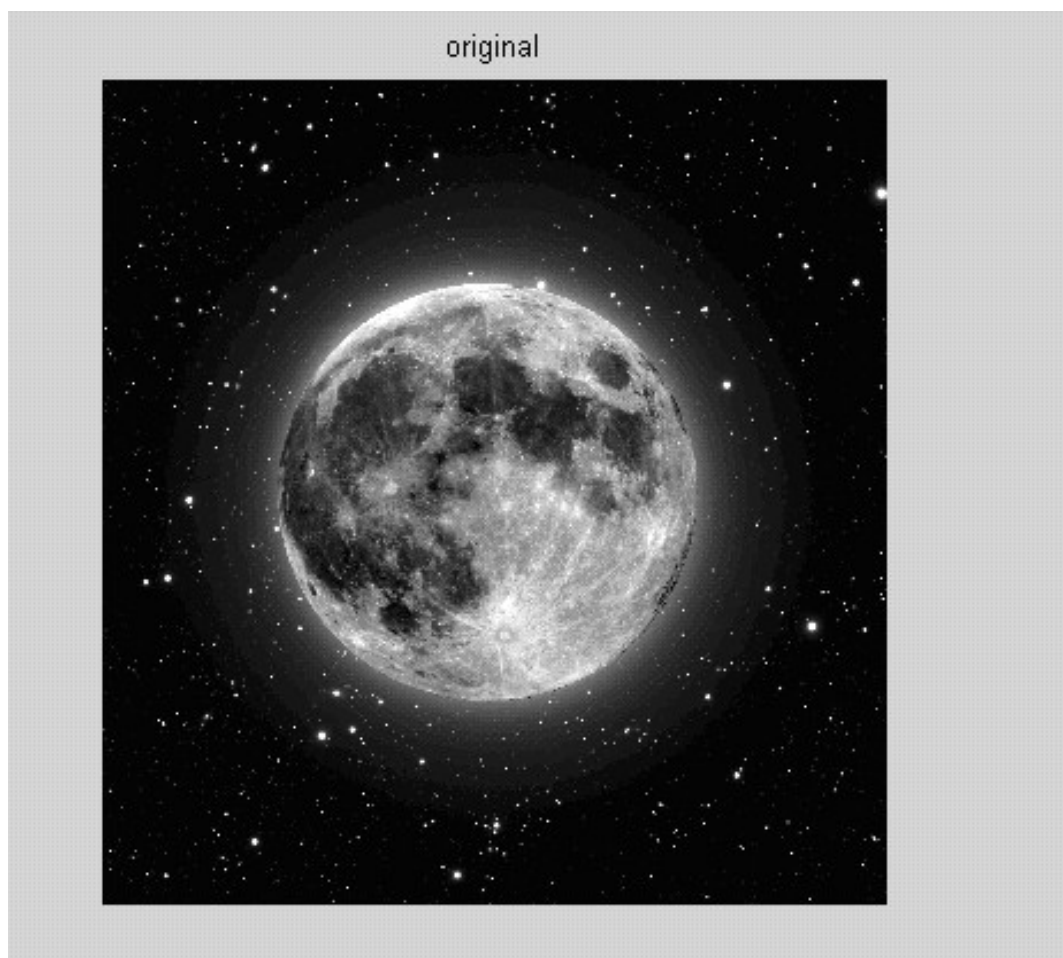
Εμφανίζοντας ένα Περιμετρικό Γράφημα Δεδομένων Εικόνας

Χρησιμοποιήστε την συνάρτηση **imcontour** απο το κουτί εργαλείων για να εμφανίσετε ένα περιμετρικό γράφημα των δεδομένων εικόνας σε εικόνα επιπέδου του γκρι .

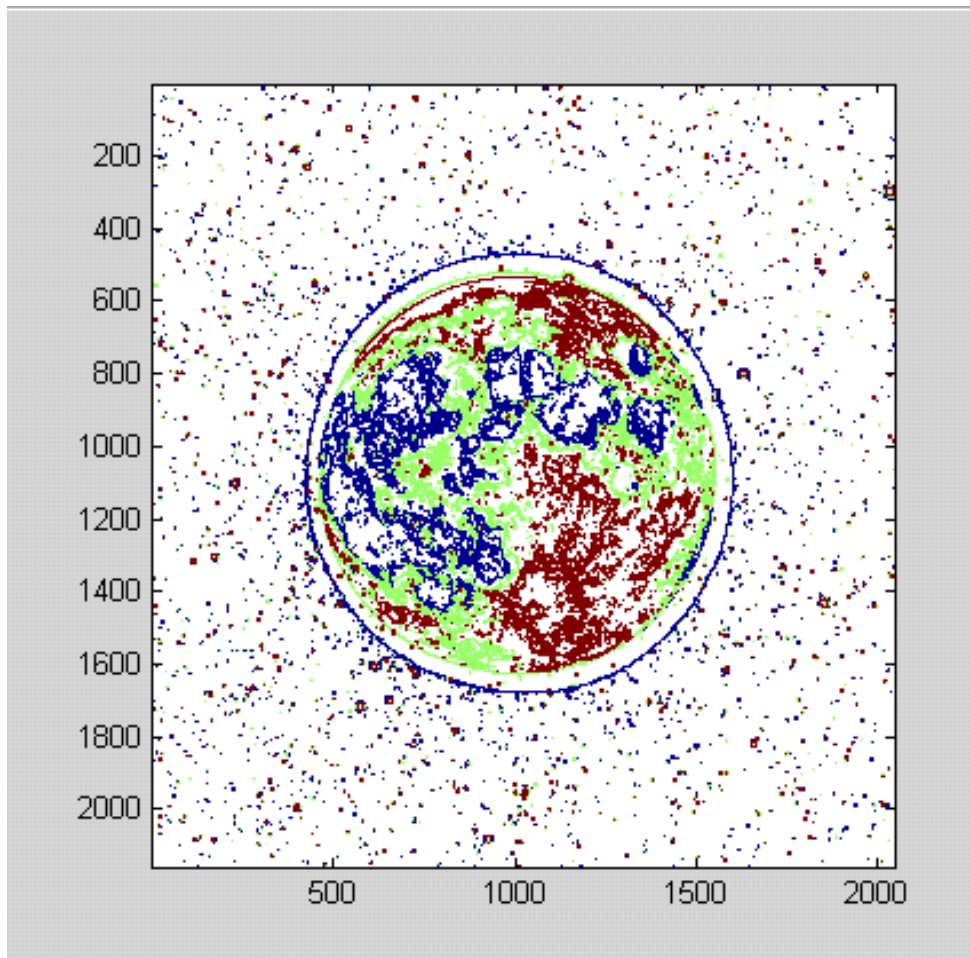
Παράδειγμα :

This command for read a grayscale image and display it

```
I = imread('moon.jpg');  
imshow(I)
```



Εμφανίστε περιμετρικό γράφημα εικόνας επιπέδου γκρί
figure, imcontour(I,3)



Δημιουργώντας το Ιστόγραμμα Εικόνας

Το Ιστόγραμμα εικόνας είναι ένα διάγραμμα που δείχνει την κατανομή των εντάσεων σε μια αριθμικά προσαρμοσμένη εικόνα ή εικόνα επιπέδου γκρί. Για να δημιουργήσετε το Ιστόγραμμα μιας εικόνας χρησιμοποιήστε την συνάρτηση **imhist**. Αυτή η συνάρτηση δημιουργεί το γράφημα του Ιστογράμματος φτιάχνοντας κάδους, ίσης απόστασης μεταξύ τους, όπου ο

καθένας αντιπροσωπεύει εύρος τιμών των δεδομένων.Εν συνεχεία, υπολογίζει τον αριθμό των pixels σε κάθε εύρος.

Παράδειγμα :

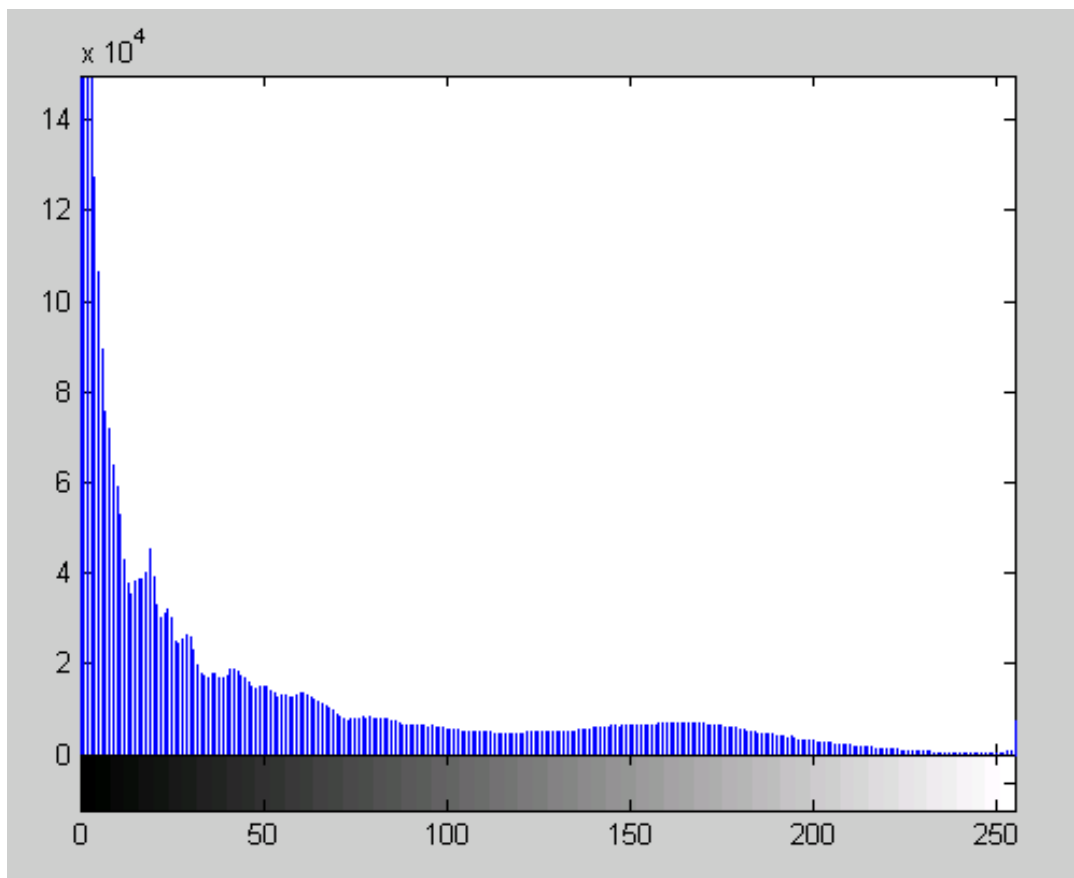
Read image and display it.

```
I = imread('moon.jpg');
```

```
imshow(I)
```



Εμφάνιση Ιστογράμματος εικόνας
figure, imhist(I)



Αναλύοντας μια εικόνα

Αυτό το τμήμα περιγράφει τις τεχνικές ανάλυσης εικόνων που επιστρέφουν πληροφορία για την δομή της εικόνας.

Ανίχνευση αιχμών

Σε μια εικόνα, η αιχμή είναι μια καμπύλη που ακολουθεί ένα μονοπάτι ταχείας αλλαγής στην ένταση της εικόνας. Οι αιχμές είναι συχνά σχετιζόμενες με τα όρια των αντικειμένων σε μια σκηνή. Η ανίχνευση αιχμών χρησιμοποιείται για να αναγνωρίσει τις αιχμές σε μια εικόνα.

Η πιο ισχυρή μέθοδος ανίχνευσης αιχμών είναι η μέθοδος **Canny**. Η μέθοδος **Canny** διαφέρει από άλλες μεθόδους ανίχνευσης αιχμών στο ότι χρησιμοποιεί δύο διαφορετικά κατώφλια (thresholds)- για να ανιχνεύσει ισχυρές και αδύναμες αιχμές (περιλαμβάνει αδύναμες αιχμές στην έξοδο μόνο όταν αυτές συνδέονται με ισχυρές). Αυτή η μέθοδος έχει ως εκ τούτου λιγότερες πιθανότητες από τις άλλες να "ξεγελαστεί" από θόρυβο και ακόμα περισσότερες να ανιχνεύσει πραγματικά αδύναμες αιχμές.

Παράδειγμα :

Read image and display it.

```
I = imread('moon.jpg');
```

```
imshow(I)
```

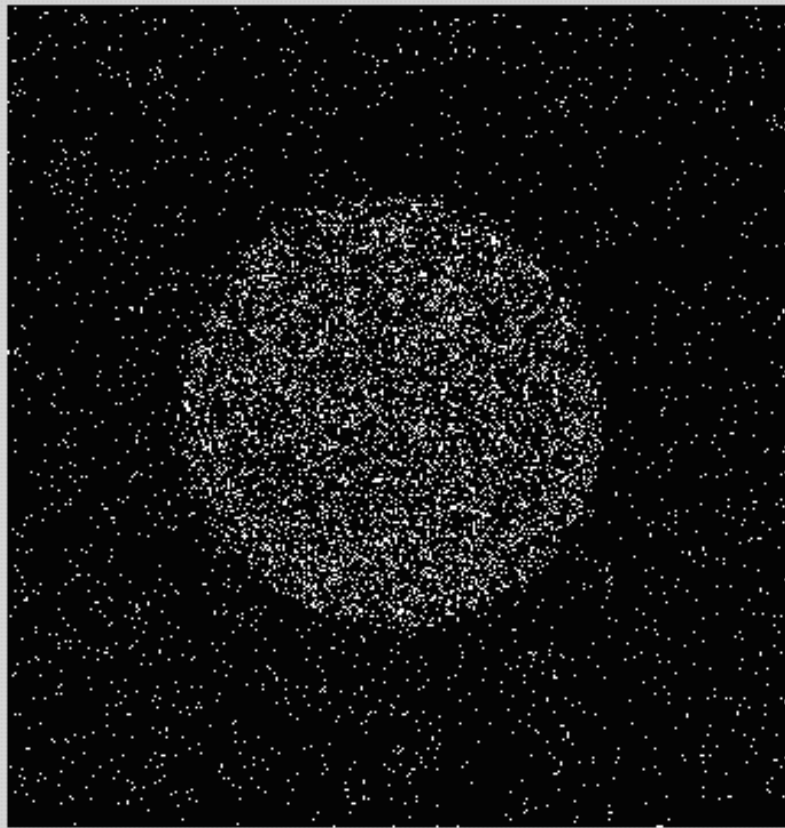


Εφαρμόστε τις **Sobel** και **Canny** μεθόδους ανίχνευσης αιχμών στην εικόνα και εμφανίστε την.

```
BW1 = edge(I,'sobel');  
BW2 = edge(I,'canny');  
imshow(BW1)  
figure, imshow(BW2)
```




SOBEL FILTER



CANNY FILTER

Εξιχνίαση Ορίων

Το κουτι εργαλείων περιλαμβάνει δύο συναρτήσεις που μπορείτε να χρησιμοποιήσετε για να βρείτε τα όρια των αντικειμένων σε μια δυαδική εικόνα.:

- **bwtraceboundary**
- **bwboundaries**

Η συνάρτηση **bwtraceboundary** επιστρέφει τη σειρά και την στήλη, συνταταγμένες, απο όλα τα pixel στα σύνορα ενός αντικειμένου σε μια εικόνα. Θα πρέπει να διευκρινίσετε την τοποθεσία ενός συνοριακού pixel του αντικειμένου ως σημείο εκκίνησης του ίχνους.

Η συνάρτηση **bwboundaries** επιστρέφει τη σειρά και την στήλη, συνταταγμένες, των συνοριακών pixels από όλα τα αντικείμενα σε μια εικόνα.

Παράδειγμα :

```
Read image and display it.  
I = imread('moon.jpg');  
imshow(I)
```



Μετατρέψτε την εικόνα σε μια δυαδική εικόνα. Η **bwtraceboundary** και η **bwboundaries** λειτουργούν μόνο για δυαδικές εικόνες.

```
BW = im2bw(I);  
imshow(BW)
```



Καθορίστε τις συντεταγμένες τις σειράς και της στήλης κάποιου pixel στο σύνορο του αντικειμένου που θέλετε να εξιχνιάσετε. Η **bwboundary** χρησιμοποιεί το συγκεκριμένο σημείο ως σημείο εκκίνησης για να την εξιχνιάση των ορίων.

```
dim = size(BW)
col = round(dim(2)/2)-90;
row = min(find(BW(:,col)))
```

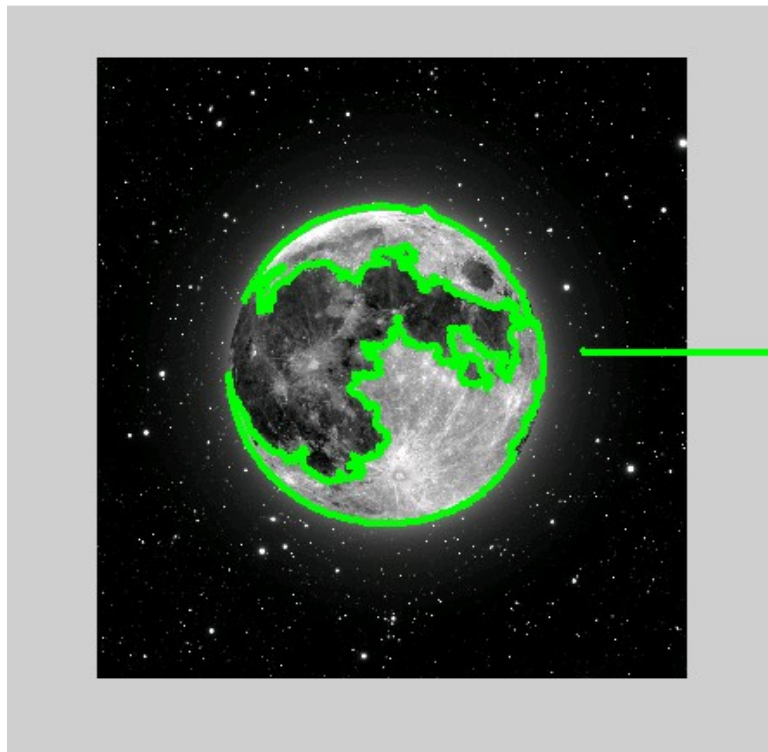
```
dim =
2158 2048
row =
524
```

Καλέστε την **bwtraceboundary** για να εξιχνιάσετε το όριο από κάποιο συγκεκριμένο σημείο. Σαν απαραίτητες προϋποθέσεις, θα πρέπει να διευκρινίσετε μια δυαδική εικόνα, τις συντεταγμένες της σειράς και της στήλης του σημείου εκκίνησης καθώς και την κατεύθυνση του πρώτου βήματος.

```
boundary = bwtraceboundary(BW,[row, col],'N');
```

Εμφανίστε την αρχική εικόνα επιπέδου γκρι και χρησιμοποιήστε τις συντεταγμένες που επιστρέφει η **bwtraceboundary** για να παρουσιάσετε το γράφημα του ορίου της εικόνας.

```
imshow(I)
hold on;
plot(boundary(:,2),boundary(:,1),'g','LineWidth',3);
```



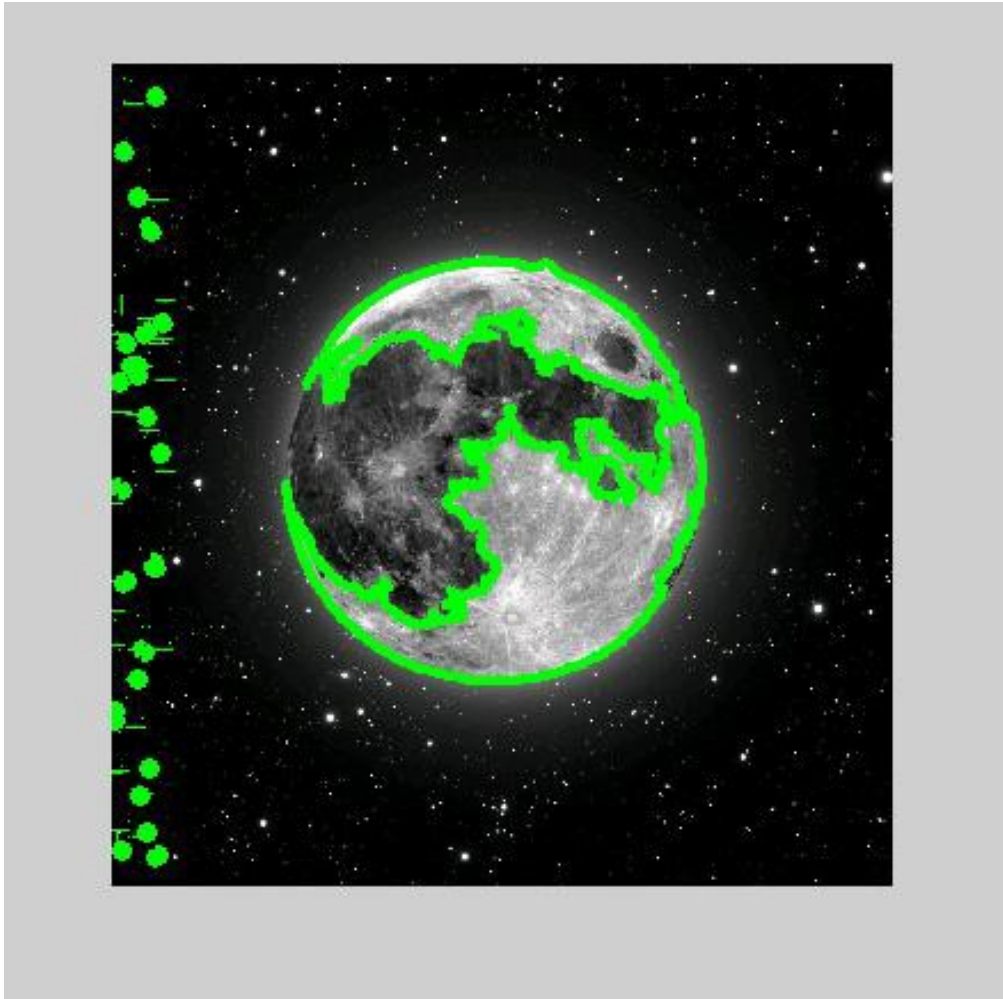
OBJECT
WITH
TRACED
BOUNDARY

Για να εξιχνιάσετε τα όρια όλων των coins σε μια εικόνα χρησιμοποιήστε την συνάρτηση **bwboundaries**.

```
BW_filled = imfill(BW,'holes');  
boundaries = bwboundaries(BW_filled);
```

Για να κάνετε το γράφημα των ορίων από όλα τα coins της αρχικής εικόνας επιπέδου γκρί χρησιμοποιήστε τις συντεταγμένες που επιστρέφει η **bwboundaries**.

```
for k=1:100  
b = boundaries{k};  
plot(b(:,2),b(:,1),'g','LineWidth',3);  
end
```



Ανίχνευση Γραμμών χρησιμοποιώντας την Μετατροπή Hough

Το κουτί εργαλείων της επεξεργασίας εικόνας περιλαμβάνει συναρτήσεις που υποστηρίζουν την Μετατροπή Hough.

- hough
- houghpeaks
- houghlines

Διαβάστε μια εικόνα μέσα στο περιβάλλον εργασίας του MATLAB.

```
I = imread('moon.jpg');
```



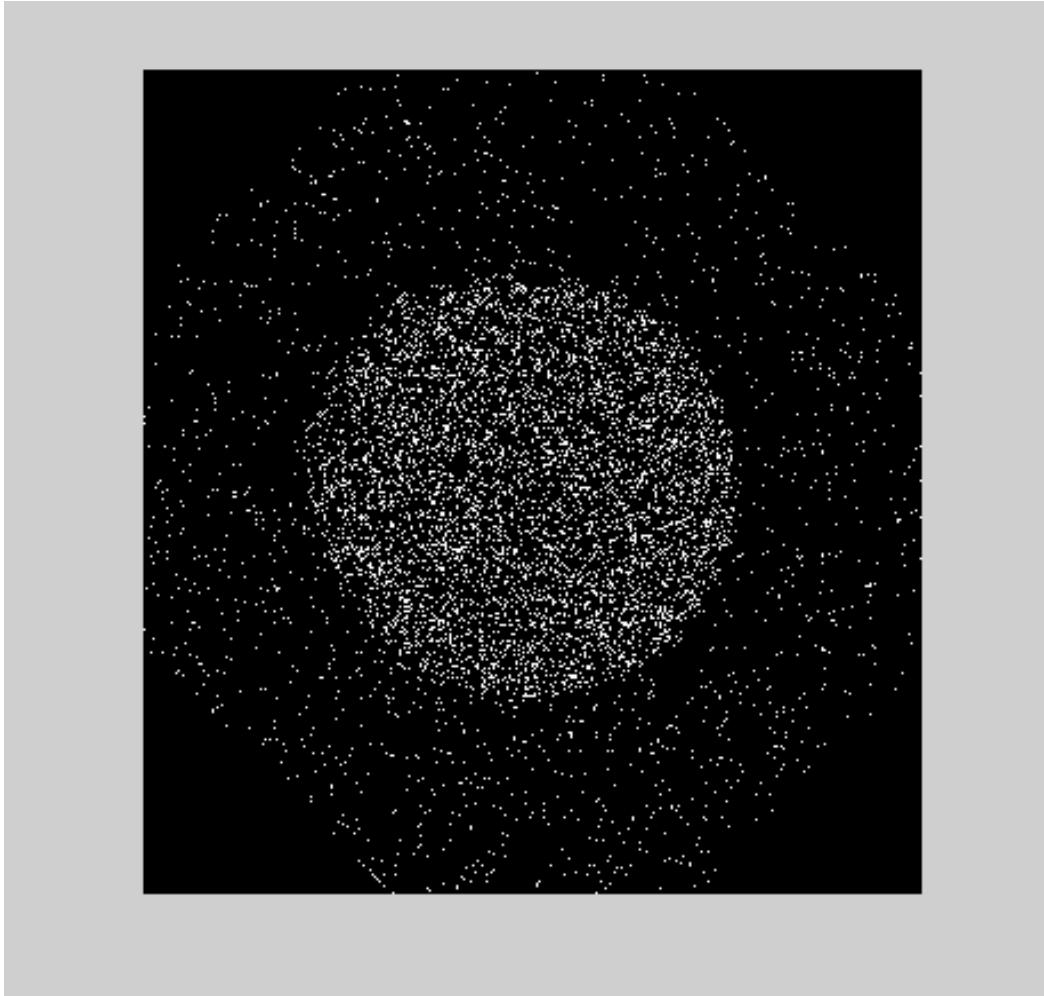
Περιστροφή και κόψιμο της εικόνας.

```
rotI = imrotate(I,45,'crop');
```



Εντοπίζοντας τις αιχμές σε μια εικόνα.

```
BW = edge(rotI,'canny');
```

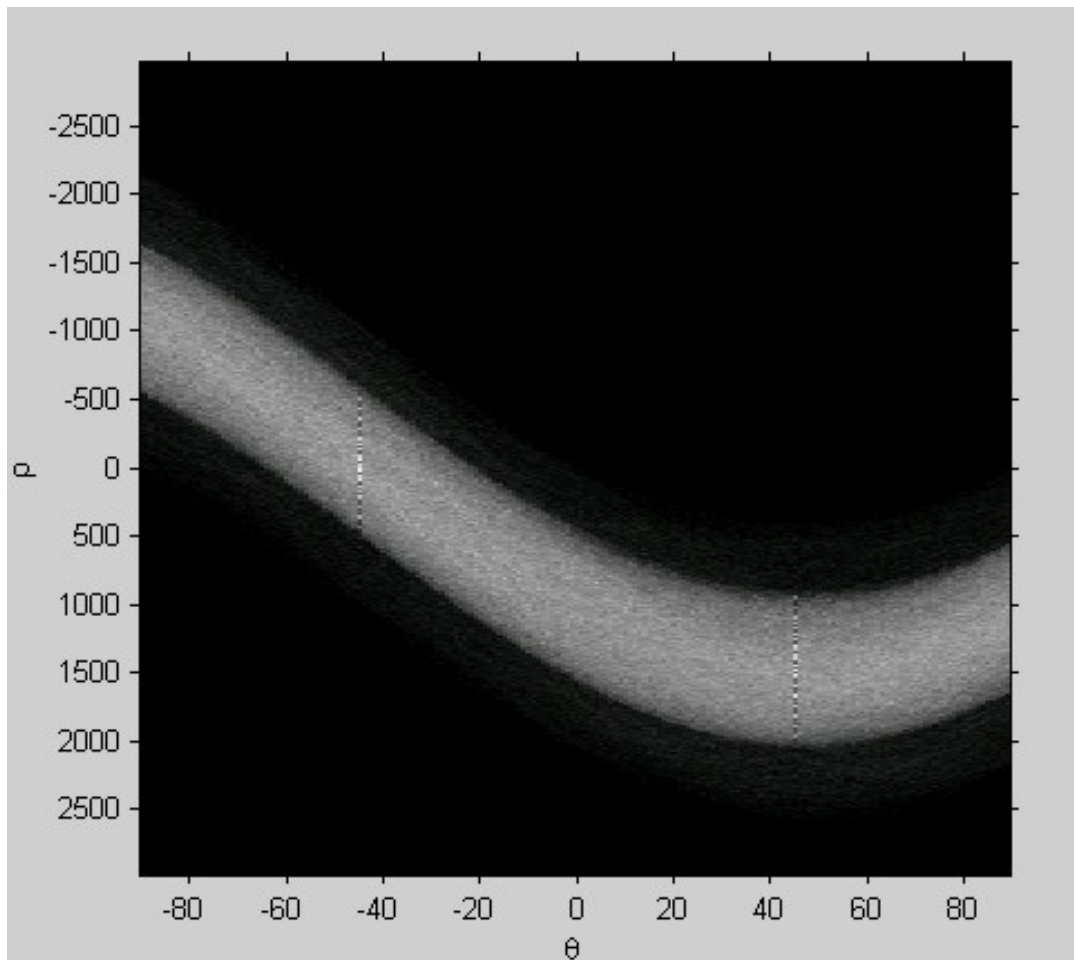


Υπολογίστε την Μετατροπή Hough μιας εικόνας χρησιμοποιώντας την συνάρτηση **hough**

```
[H,theta,rho] = hough(BW);
```

Εμφανίστε την μετατροπή

```
imshow(H,[],'XData',theta,'YData',rho,...  
'InitialMagnification','fit');  
xlabel('\theta'), ylabel('\rho');  
axis on, axis normal, hold on;
```

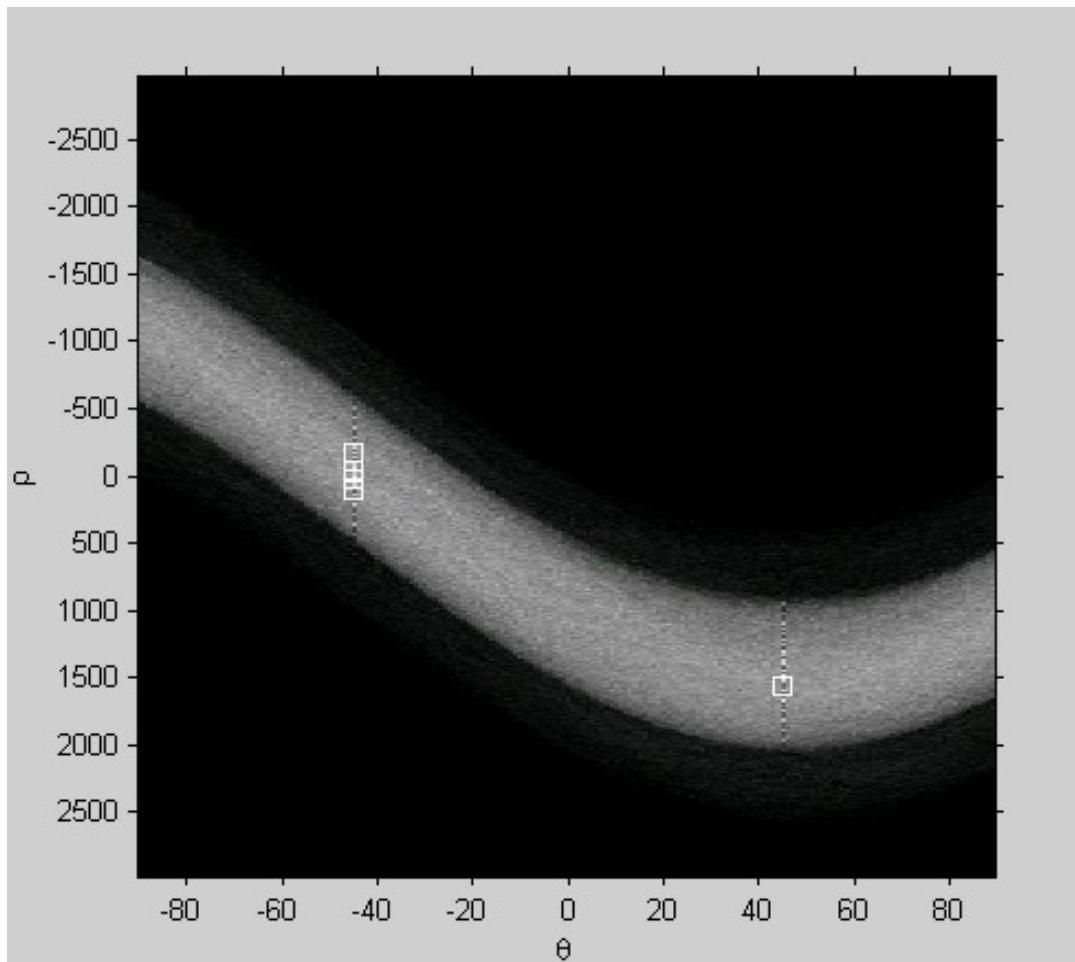


Βρείτε τις κορυφές της μήτρας της μετατροπής Hough, H , χρησιμοποιώντας την συνάρτηση **houghpeaks**.

```
P = houghpeaks(H,5,'threshold',ceil(0.3*max(H(:))));
```

Plot the peaks.

```
x = theta(P(:,2));
y = rho(P(:,1));
plot(x,y,'s','color','white');
```



Βρείτε τις γραμμές σε μια εικόνα.

```
lines = houghlines(BW,theta,rho,P,'FillGap',5,'MinLength',7);
```

Δημιουργήστε το γράφημα που επιθέτει τις γραμμές της αρχικής εικόνας.

```
figure, imshow(rotI), hold on
```

```
max_len = 0;
```

```
for k = 1:length(lines)
```

```
xy = [lines(k).point1; lines(k).point2];
```

```
plot(xy(:,1),xy(:,2),'LineWidth',2,'Color','green');
```

```
%Γράφημα εκκινήσεων και καταλήξεων των γραμμών
```

```
plot(xy(1,1),xy(1,2),'x','LineWidth',2,'Color','yellow');
```

```
plot(xy(2,1),xy(2,2),'x','LineWidth',2,'Color','red');
```

```
%Καθορίστε τα τελικά σημεία του μακρύτερου τμήματος γραμμής
```

```
len = norm(lines(k).point1 – lines(k).point2);
```

```
if ( len > max_len)
```

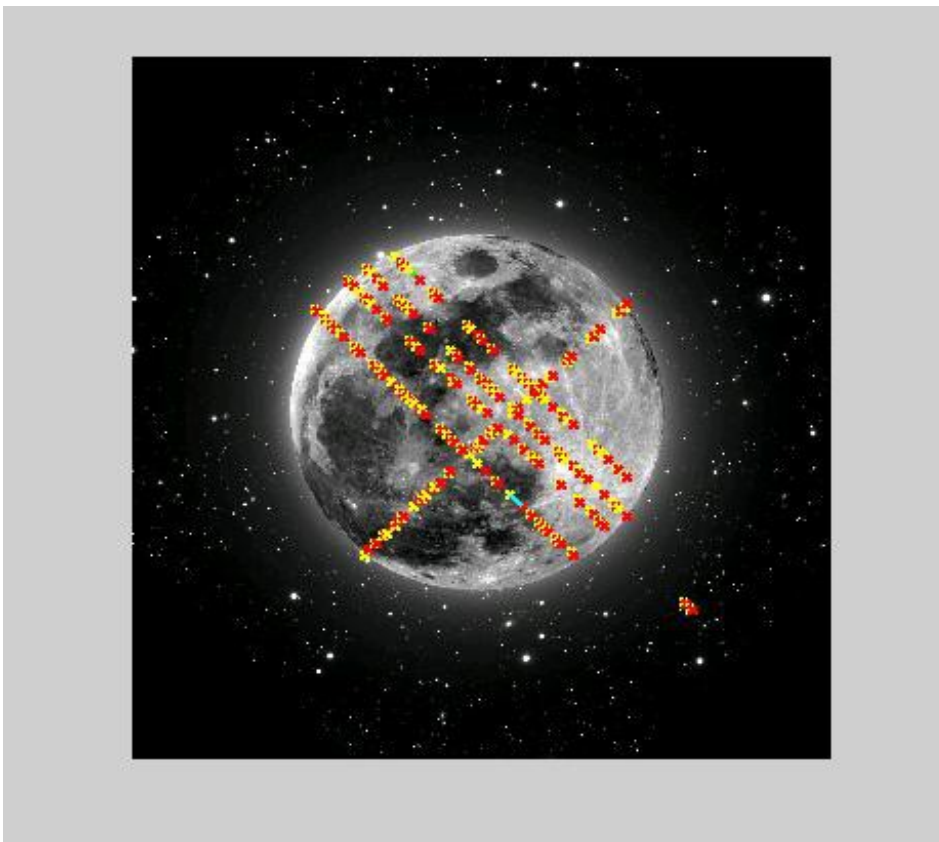
```
max_len = len;
```

```
xy_long = xy;
```

```
end
```

```
end
```

```
%Επισημάνετε το μεγαλύτερο τμήμα της γραμμής  
plot(xy_long(:,1),xy_long(:,2),'LineWidth',2,'Color','cyan');
```



Αναλύοντας την Υφή μιας εικόνας

Το κουτί εργαλείων υποστηρίζει ένα σύνολο λειτουργιών που μπορείτε να χρησιμοποιήσετε για την ανάλυση Υφής. Η ανάλυση Υφής χρησιμοποιείται για σωρεία εφαρμογών, συμπεριλαμβάνοντας την τηλεπισκόπηση, την αυτοματοποιημένη επιθεώρηση και την επεξεργασία εικόνων ιατρικής φύσεως. Η ανάλυση Υφής μπορεί επίσης να χρησιμοποιηθεί για την εύρεση των ορίων υφής καλούμενη και κατάτμηση υφής.

Επιπροσθέτως, η ανάλυση Υφής είναι χρήσιμη όταν τα αντικείμενα σε μια εικόνα χαρακτηρίζονται περισσότερο από την υφή παρά από την ένταση και οι παραδοσιακές τεχνικές κατωφλίου δεν μπορούν να χρησιμοποιηθούν αποτελεσματικά.

Συναρτήσεις Υφής

Read in the image and display it.

```
I = imread('moon.jpg');  
imshow(I)
```



Φιλτράρετε την εικόνα με την συνάρτηση **rangefilt** και εμφανίστε τα αποτελέσματα.

```
K = rangefilt(I);  
figure, imshow(K)
```



Προσαρμογή της Έντασης

Η Προσαρμογή της Έντασης είναι μια τεχνική ενίσχυσης της εικόνας που χαρτογραφεί τις τιμές της έντασης της εικόνας σε καινούριο εύρος τιμών.

Example:

Παράδειγμα :

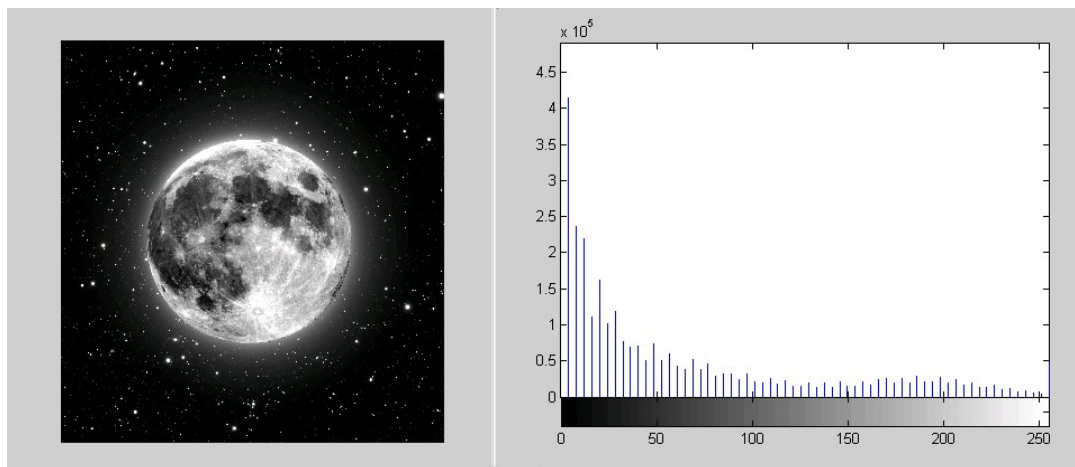
```
I = imread('moon.jpg');  
imshow(I)  
figure, imhist(I,64)
```

Προσαρμογή των τιμών Έντασης σε συγκεκριμένο Εύρος τιμών

Μπορείτε να προσαρμόσετε τις τιμές της έντασης σε μια εικόνα χρησιμοποιώντας την συνάρτηση **imadjust**, όπου επισημαίνετε το εύρος των τιμών της έντασης στην εξερχόμενη εικόνα.

Example:

```
I = imread('moon.jpg');  
J = imadjust(I);  
imshow(J)  
figure, imhist(J,64)
```

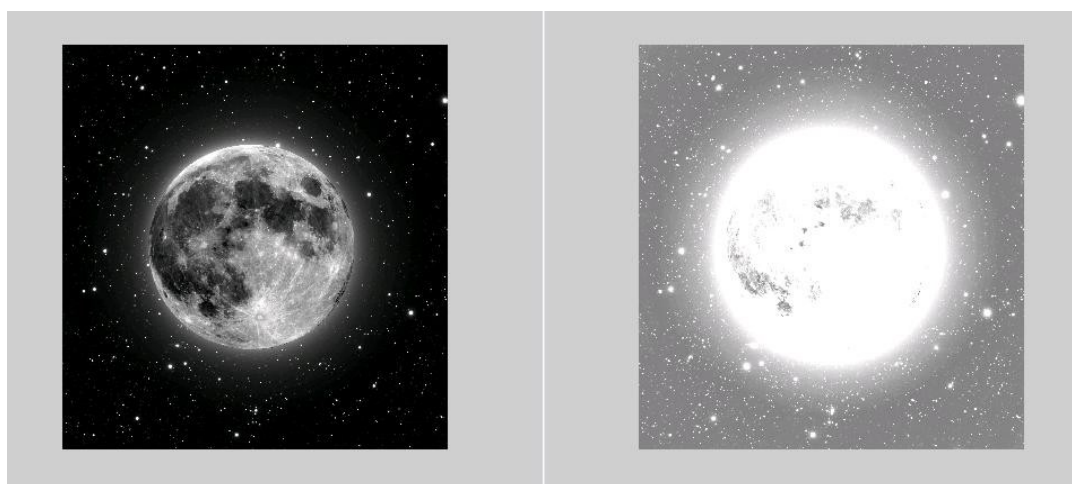


Καθορισμός των Ορίων Προσαρμογής

Μπορείτε προαιρετικά να διευκρινίσετε το εύρος των τιμών εισόδου και εξόδου χρησιμοποιώντας την συνάρτηση **imadjust**. Διευκρινίζετε τα εύρη σε δύο διανύσματα που "περνάτε" στην **imadjust** σαν μεταβλητές. Το πρώτο διάνυσμα ορίζει τις τιμές της χαμηλής και της υψηλής έντασης που θέλετε να χαρτογραφησετε. Το δεύτερο διάνυσμα ορίζει την κλίμακα πάνω στην οποία θέλετε να χαρτογραφησετε τα διανύσματα.

Παράδειγμα :

```
I = imread('moon.jpg');  
J = imadjust(I,[0 0.2],[0.5 1]);  
imshow(I)  
figure, imshow(J)
```

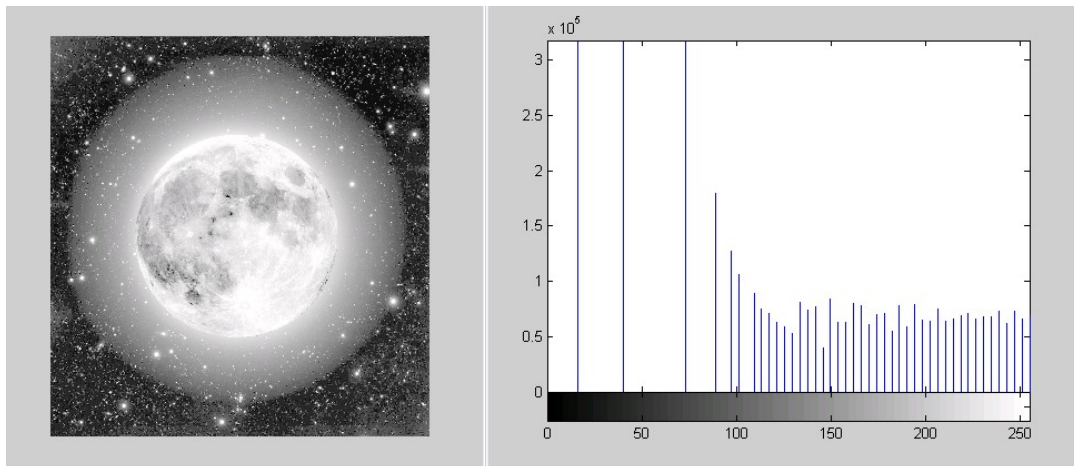


Εξισορρόπηση Ιστογράμματος

Η διαδικασία της προσαρμογής των τιμών έντασης μπορεί να γίνει και αυτόματα μέσω της συνάρτησης **histeq**. Η συνάρτηση **histeq** εκτελεί την Εξισορρόπηση Ιστογράμματος, η οποία περιλαμβάνει την μεταμόρφωση των τιμών της έντασης έτσι ώστε το ιστογράμμα της εξερχόμενης εικόνας περίπου να αντιστοιχεί σε συγκεκριμένο ιστογράμμα.

Παράδειγμα :

```
I = imread('moon.jpg');  
J = histeq(I);  
imshow(J)  
figure, imhist(J,64)
```



Απομάκρυνση Θορύβου

Οι ψηφιακές εικόνες είναι επιρρεπείς σε ποικιλία τύπων του θορύβου. Ο θόρυβος είναι το αποτέλεσμα των σφαλμάτων, κατά την διαδικασία λήψης μιας εικόνας, που οδηγούν σε τιμές pixel που δεν αντικατοπτρίζουν τις πραγματικές εντάσεις της πραγματικής σκηνής. Υπάρχουν διάφοροι τρόποι που ο θόρυβος μπορεί να εισαχθεί σε μια εικόνα, εξαρτάται κυρίως από το πώς έχει δημιουργηθεί η εικόνα.

Φιλτράρισμα Ενδιάμεσου

Το φιλτράρισμα ενδιάμεσου είναι όμοιο με την χρησιμοποίηση ενός φίλτρου μέσου όρου, στο οποίο κάθε pixel εξόδου έχει οριστεί σε έναν μέσο όρο των τιμών των pixel στην γειτονιά του αντίστοιχου pixel εισόδου.

Παράδειγμα :

Διάβασε την εικόνα και εμφάνισέ την.

```
I = imread('moon.jpg');  
imshow(I)
```



Προσθήκη Θορύβου

```
J = imnoise(I,'salt & pepper',0.02);  
figure, imshow(J)
```



Φιλτράρετε την "θορυβώδη" εικόνα με ένα φίλτρο μέσου όρου και εμφανίστε τ' αποτελέσματα.

```
K = filter2(fspecial('average',3),J)/200;  
figure, imshow(K)
```



Τώρα χρησιμοποιήστε ένα φίλτρο ενδιαμέσου για να φιλτράρετε την εικόνα και εμφανίστε πάλι τ' αποτελέσματα. Παρατηρήστε ότι η **medfilt2** συνάρτηση αποδίδει καλύτερα αποτελέσματα στην αφαίρεση του θορύβου και με λιγότερα θολωμένες αιχμές.

```
L = medfilt2(J,[3 3]);  
figure, imshow(K)  
figure, imshow(L)
```



Επεξεργασία με βάση την ROI

Σε αυτό το κεφάλαιο περιγράφεται πως ορίζεται μια περιοχή ενδιαφέροντος (region of interest-ROI) και η εκτέλεση της επεξεργασίας στην περιοχή αυτή.

Καθορισμός της Περιοχής Ενδιαφέροντος(ROI)

Η περιοχή ενδιαφέροντος (ROI) είναι ένα τμήμα μιας εικόνας που θέλετε να φιλτράρετε ή να κάνετε κάποιου άλλου είδους επεξεργασία. Ορίζετε μια ROI δημιουργώντας μια μάσκα, η οποία είναι μια δυαδική εικόνα του ίδιου μεγέθους με την εικόνα που θέλετε να επεξεργαστείτε, με pixels που ορίζουν την ROI με "1" και όλα τα υπόλοιπα pixels με "0".

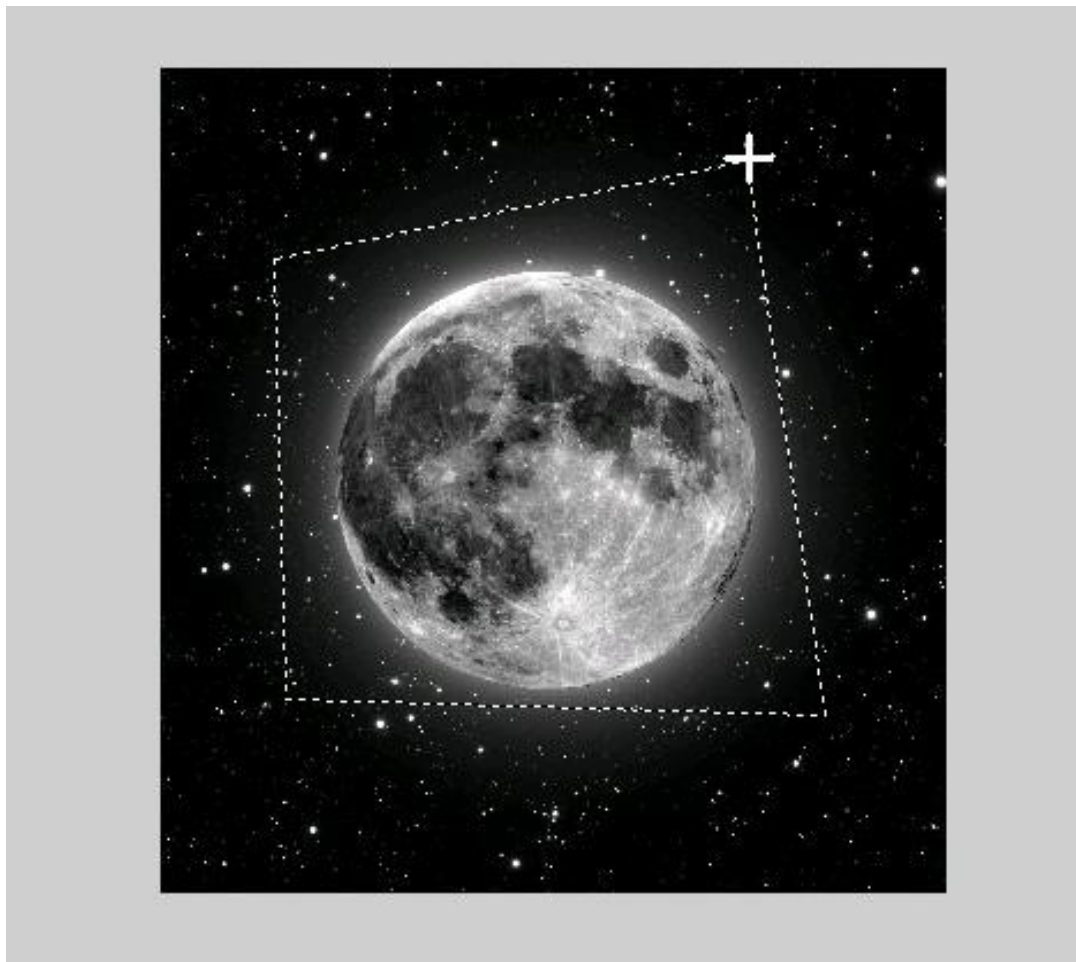
Επιλέγοντας μια Πολυγωνική ROI Διαδραστικά

Χρησιμοποιήστε την συνάρτηση **roi_poly** για να καθορίσετε μια πολυγωνική ROI διαδραστικά σε συγκεκριμένη εικόνα. Για να το κάνετε αυτό, εμφανίστε την εικόνα χρησιμοποιώντας την **imshow** και μετά καλέστε την **roi_poly** χωρίς ορίσματα εισόδου. Καθώς μετακινείτε τον κέρσορα πάνω στην εικόνα που εμφανίσατε πάνω στους άξονες, ο κέρσορας αλλάζει σε σχήμα στόχαστρου. Μπορείτε να καθορίσετε τότε τις κορυφές του πολυγώνου κλικώντας σημεία της εικόνας.

Όταν τελειώσετε με την επιλογή των σημείων πατήστε Return. Η **roipoly** θα σας επιστρέψει μια δυαδική εικόνα του ίδιου μεγέθους με την εικόνα εισόδου, που περιέχει "1" μέσα στο πολύγωνο και "0" έξω από αυτό.

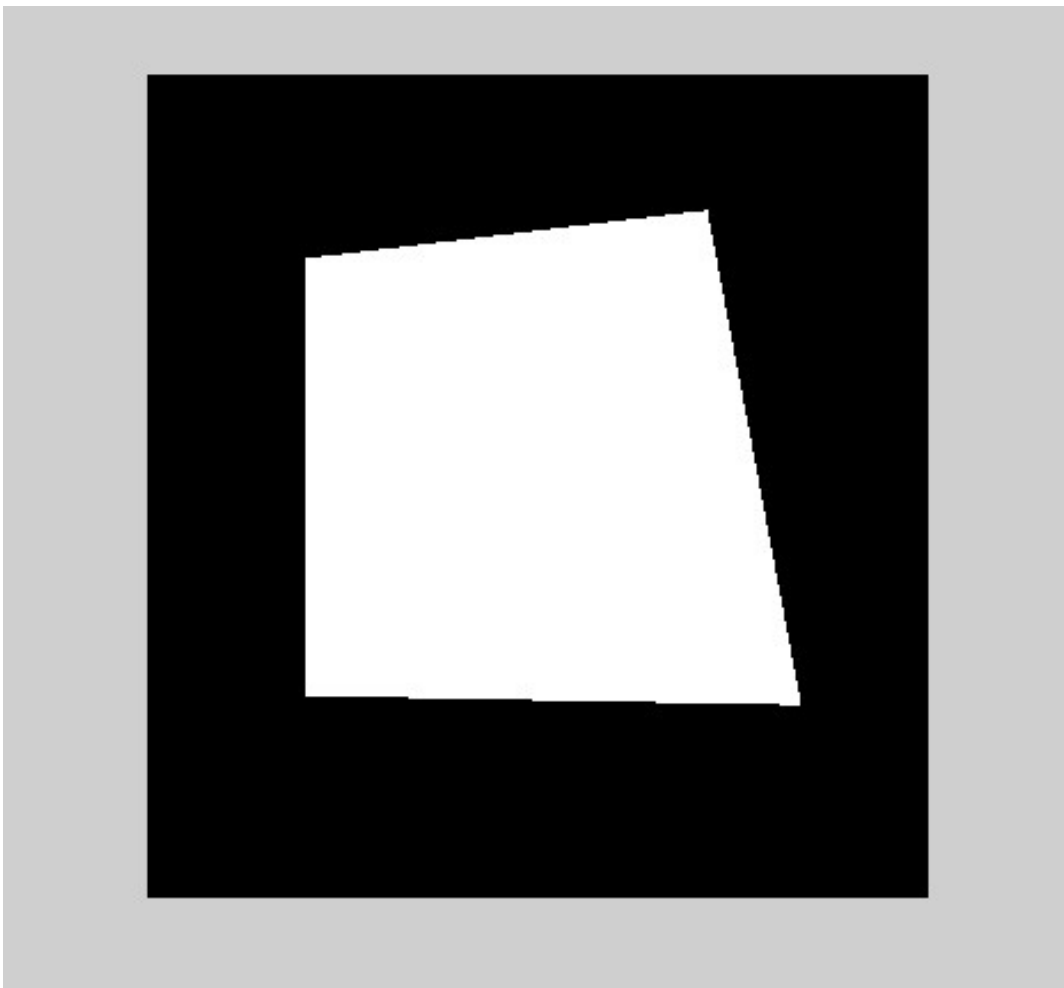
Παράδειγμα :

```
I = imread('moon.jpg');  
imshow(I)  
BW = roipoly;
```



Αφού την επιλέξετε πατήστε ENTER και γράψτε :

```
imshow(BW)  
and you will see something similar
```



Φιλτράροντας μια ROI

Το φιλτράρισμα μιας περιοχής ενδιαφέροντος είναι η διαδικασία κατά την οποία εφαρμόζεται ένα φίλτρο σε κάποια περιοχή της εικόνας, εκεί όπου μια δυαδική μάσκα ορίζει την περιοχή. Για παράδειγμα, μπορείτε να εφαρμόσετε ένα φίλτρο προσαρμογής έντασης για να επιβελιώσετε περιοχές της εικόνας.

Φιλτράροντας μια Περιοχή της Εικόνας

Αυτο το παράδειγμα χρησιμοποιεί φιλτράρισμα με μάσκα για να αυξήσει την αντίθεση σε συγκεκριμένο σημείο της εικόνας.

Διάβασε την εικόνα

```
I = imread('moon.jpg');
```

Δημιούργησε την μάσκα

```
imshow(I)
```

```
BW = roiplot;
```

Δημιούργησε το φίλτρο

```
h = fspecial('unsharp');
```

Κάλυψε την `roifilt2`, διευκρινίζοντας την εικόνα που θα φιλτραρισθεί και το φίλτρο.

```
I2 = roifilt2(h,I,BW);
```

```
imshow(I)
```

```
figure, imshow(I2)
```



Ξεθάμπωμα με Κανονικοποιημένο Φίλτρο

Χρησιμοποιήστε την συνάρτηση **deconvreg** για να ξεθομπώσετε μια εικόνα χρησιμοποιώντας κανονικοποιημένο φίλτρο. Το κανονικοποιημένο φίλτρο μπορεί να χρησιμοποιηθεί αποτελεσματικά όταν οι πληροφορίες που έχουμε για τον επιπρόσθετο θόρυβο είναι περιορισμένες.

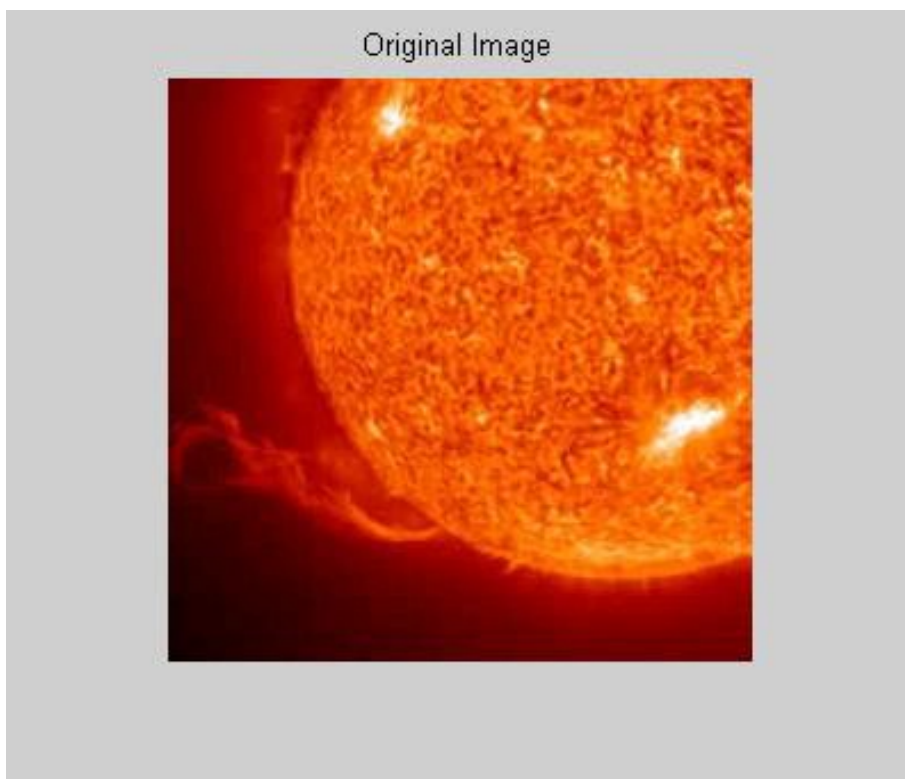
Παράδειγμα:

Read an image into the MATLAB workspace.

```
I = imread('sun.jpg');
```

```
I = I(125+[1:256],1:256,:);
```

```
figure; imshow(I); title('Original Image');
```

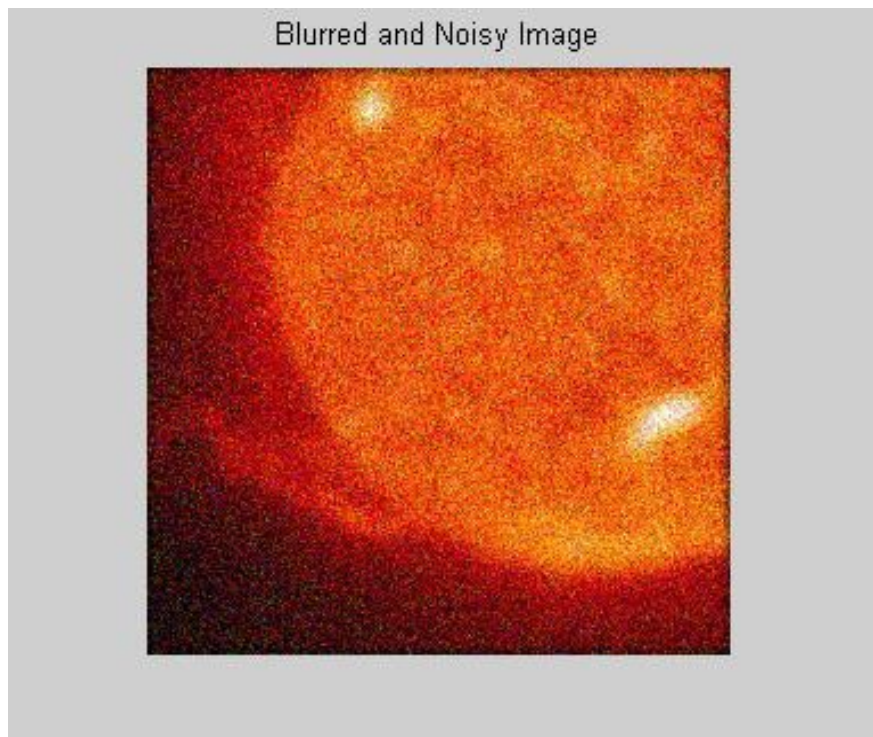


Δημιουργήστε PSF.

```
PSF = fspecial('gaussian',11,5);
```

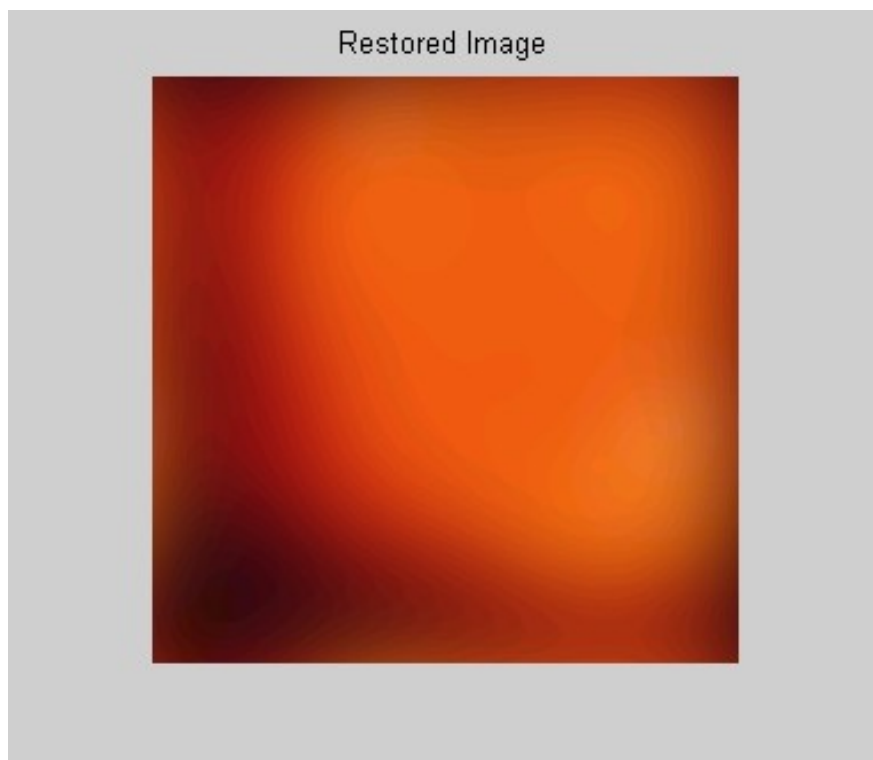
Δημιουργήστε μια προσομοίωση θαμπώματος της εικόνας και προσθέστε θόρυβο

```
Blurred = imfilter(I,PSF,'conv');  
V = .02;  
BlurredNoisy = imnoise(Blurred,'gaussian',0,V);  
figure;imshow(BlurredNoisy);title('Blurred and Noisy Image');
```



Χρησιμοποιήστε την **deconvreg** για να ξεθαμπώσετε την εικόνα, διευκρινίζοντας το PSF που χρησιμοποιήθηκε για να δημιουργηθεί το θάμπωμα καθώς και την ισχύ του θορύβου, NP.

```
NP = V*prod(size(I));  
[reg1 LAGRA] = deconvreg(BlurredNoisy,PSF,NP);  
figure,imshow(reg1),title('Restored Image');
```



Ξεθάμπωμα με Τυφλό Αλγόριθμο Αποσυνέλιξης

Η συνάρτηση **deconvblind**, όπως ακριβώς και η **deconvlucy**, εφαρμόζει διάφορες προσαρμογές στην αρχική Lucy-Richardson, μέγιστης πιθανότητας αλγόριθμος που αντιμετωπίζει πολύπλοκες εργασίες αποκατάστασης της εικόνας.

Χρησιμοποιώντας την συνάρτηση `deconvblind` για το ξεθάμπωμα εικόνας

Για ν' απεικονιστεί η τυφλή αποσυνέλιξη, το παρακάτω παράδειγμα δημιουργεί μια προσομοιωμένη θαμπή εικόνα και μετά χρησιμοποιεί την `deconvblind` συνάρτηση για να την ξεθαμπώσει.

Παράδειγμα :

Read an image into the MATLAB workspace.

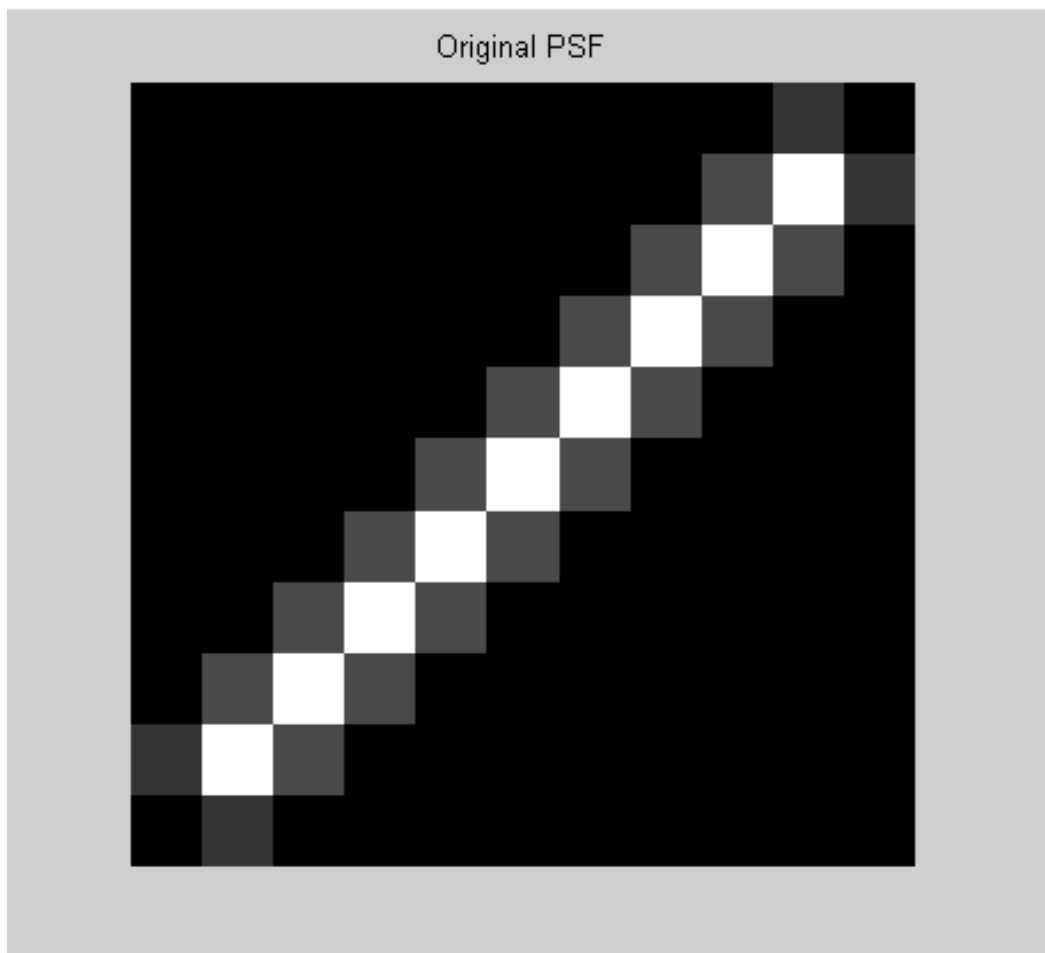
Διάβασε μια εικόνα στο περιβάλλον εργασίας του MATLAB.

```
I = imread('moon1.jpg');  
figure; imshow(I); title('Original Image');
```



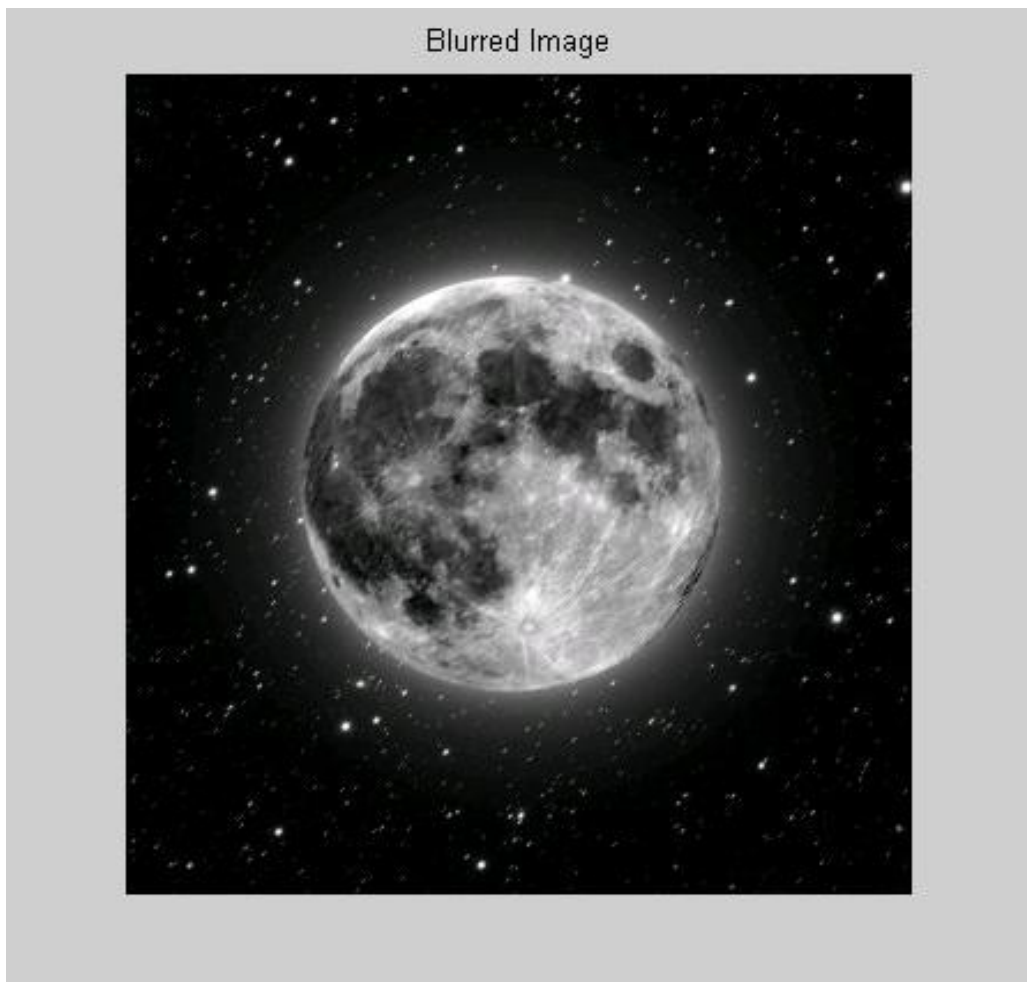
Δημιουργήστε το PSF.

```
PSF = fspecial('motion',13,45);  
figure; imshow(PSF,[],'notruesize'); title('Original PSF');
```



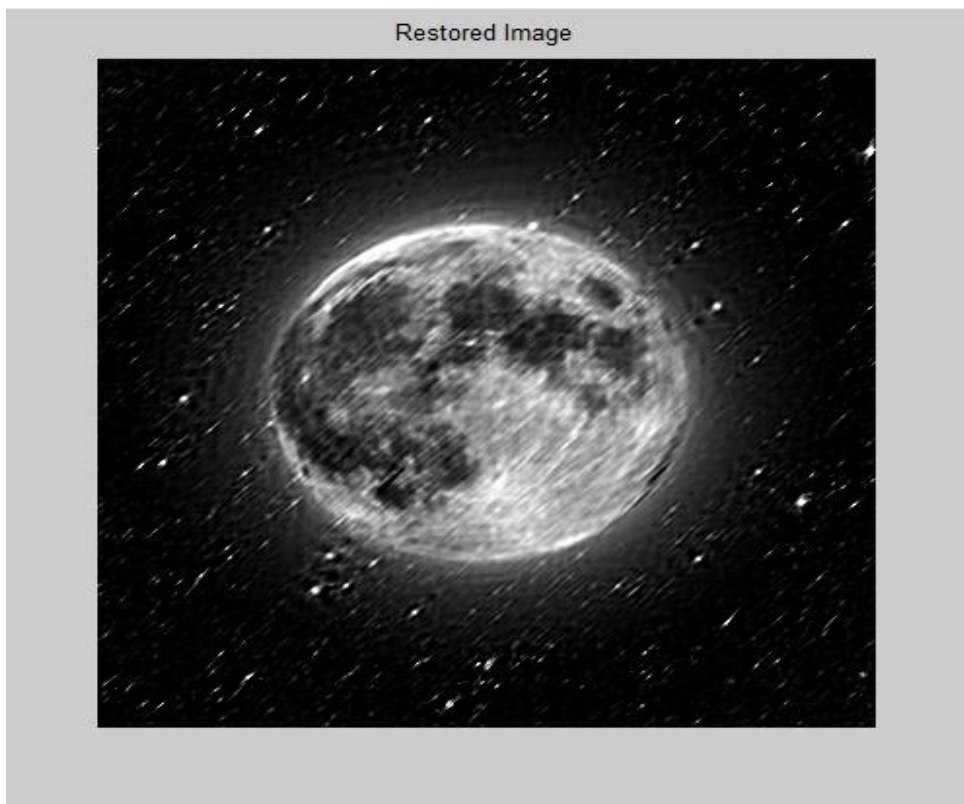
Δημιουργήστε ένα προσομοιωμένο θάμπωμα στη εικόνα

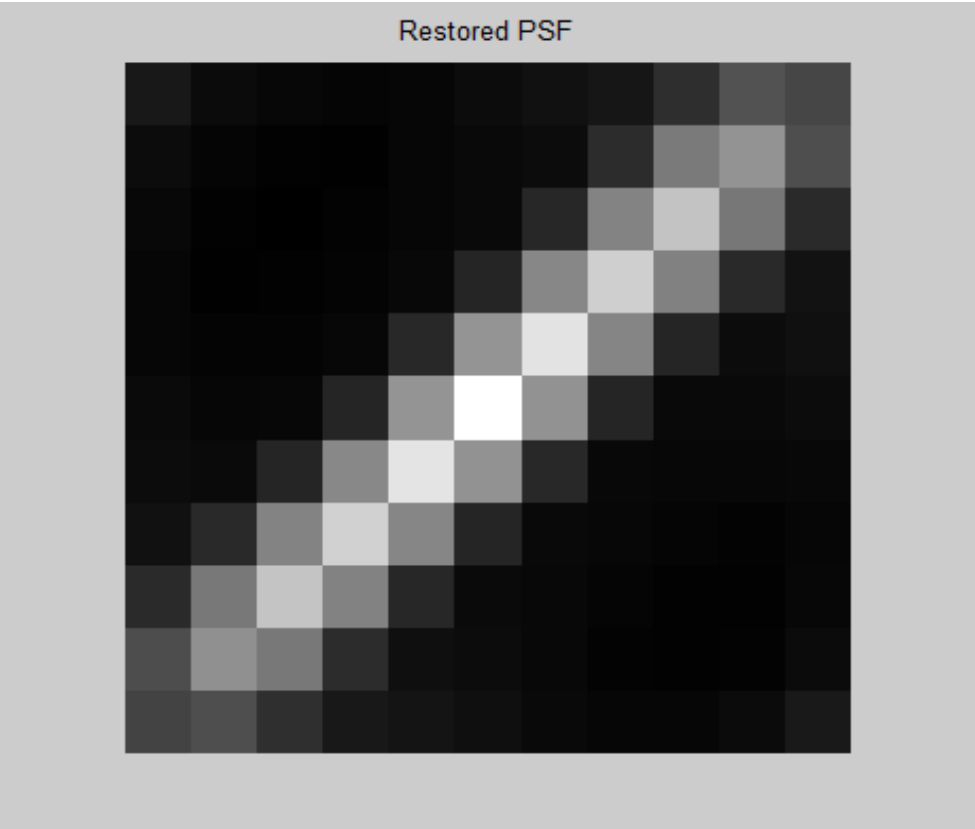
```
Blurred = imfilter(I,PSF,'circ','conv');  
figure; imshow(Blurred); title('Blurred Image');
```



Ξεθαρπώστε την εικόνα, κάνοντας μια αρχική υπόθεση για το μέγεθος του PSF.

```
INITPSF = ones(size(PSF));  
[J P]= deconvblind(Blurred,INITPSF,30);  
figure; imshow(J); title('Restored Image');  
figure; imshow(P,[],'notruesize');  
title('Restored PSF');
```





Λειτουργίες "συρώμενης γειτονιάς"

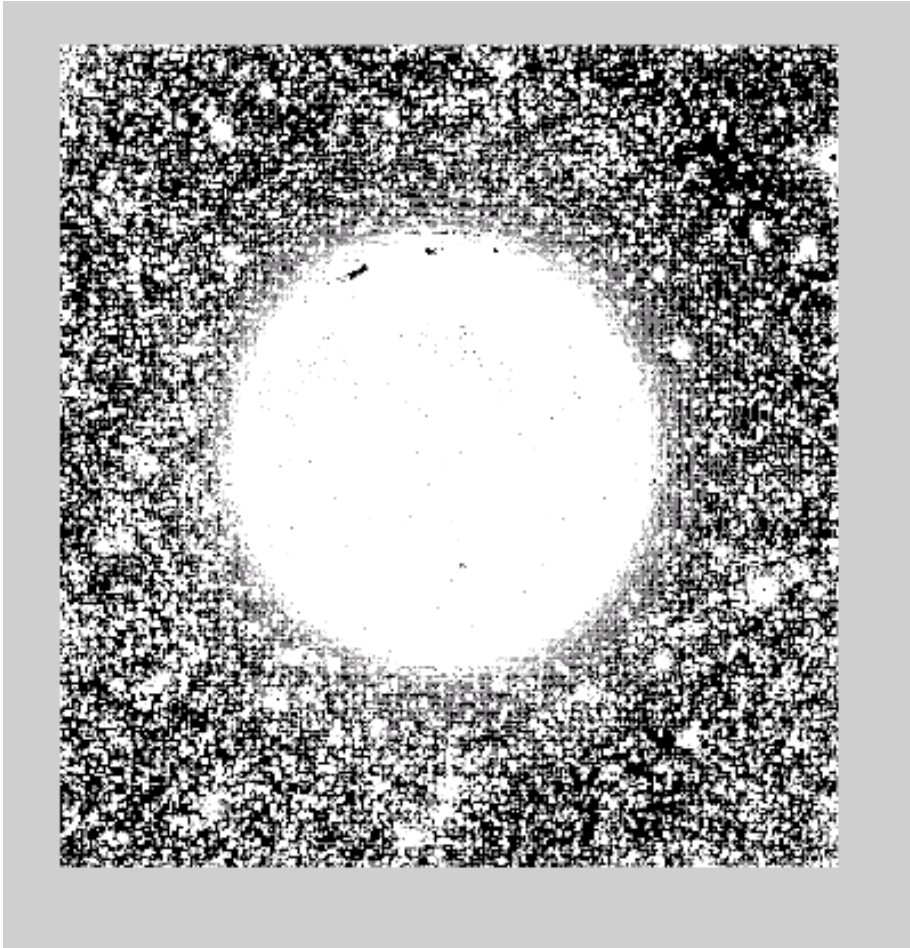
Η λειτουργία της "συρώμενης γειτονιάς" παρουσιάζει ένα pixel σε κάποια χρονική στιγμή, με την τιμή ενός οποιουδήποτε pixel της εικόνας εξόδου καθορισμένο από την εφαρμογή ενός αλγόριθμου στις αντίστοιχες τιμές των γειτονικών pixels της εικόνας εισόδου. Η γειτονιά ενός pixel είναι ένα σύνολο από pixels, καθορισμένα από την σχετική θέση τους με το συγκεκριμένο pixel, που καλείται κεντρικό pixel. Η γειτονιά είναι ένα ορθογώνιο μπλόκ, που καθώς κινείστε από το ένα στοιχείο στο άλλο σε μια μητρική εικόνα, το μπλόκ της γειτονιάς σύρεται προς την ίδια κατεύθυνση.

Εφαρμογή Γραμμικού και Μη Γραμμικού Φιλτραρίσματος

Μπορείτε να χρησιμοποιήσετε τις λειτουργίες "συρόμενης γειτονιάς" για να εφαρμόσετε πολλών ειδών λειτουργίες φιλτραρίσματος.

Για ποικιλία λειτουργιών "συρόμενης γειτονιάς" μπορείτε να εφαρμόσετε την συνάρτηση **nlfilter**. Η **nlfilter** παίρνει σαν όρισματα εισόδου μια εικόνα, το μέγεθος της γειτονιάς και μια συνάρτηση που επιστρέφει μια διαβάθμιση, και επιστρέφει μια εικόνα ίδιου μεγέθους με την αρχική εικόνα εισόδου. Η αξία κάθε pixel της εικόνας εξόδου είναι υπολογισμένη με το πέρασμα από την γειτονιά του αντίστοιχου pixel εισόδου για την συγκεκριμένη λειτουργία.

```
I = imread('moon.jpg');  
I2 = nlfilter(I,[3 3],'std2');  
imshow (I2
```



```
I = imread('moon.jpg');  
f = @(x) max(x(:));  
I2 = nifilter(I,[3 3],f);  
imshow(I);  
figure, imshow(I2);
```



ΕΠΙΛΟΓΟΣ-ΣΥΜΠΕΡΑΣΜΑΤΑ

Τις τελευταίες δεκαετίες η επεξεργασία εικόνας , είδαμε ότι ,έχει εξελιχθεί ραγδαία παράλληλα και με την επανάσταση των ηλεκτρονικών υπολογιστών.Στην πτυχιακή αυτή διαπραγματευτήκαμε έννοιες που είναι συγκερασμός 2 πραγμάτων , της φωτογραφίας και του Matlab.Συμπερασματικά θα σημειώναμε το πόσο εύκολα ένας λάτρης της φωτογραφίας θα μπορούσε να διορθώσει τις εικόνες του μέσω ενός πολύ καλού μαθηματικού και πρακτικού εργαλείου όπως είναι το Matlab.

ΒΙΒΛΙΟΓΡΑΦΙΑ-ΠΗΓΕΣ

- wikipedia
- XDCAM-USER.COM
- Introduction to image processing in Matlab 1 by Kristian Sandberg
Department of Applied Mathematics, University of Colorado