



ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

**ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε
έγγραφα από κάμερα**

Δήμητρα Ν. Πάνου

Επιβλέποντες **Βασίλειος Γάτος**, Ερευνητής ΕΚΕΦΕ **Δημόκριτος**
Σέργιος Θεοδωρίδης Καθηγητής ΕΚΠΑ

ΑΘΗΝΑ

ΝΟΕΜΒΡΙΟΣ 2016

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφα από κάμερα

Δήμητρα Ν. Πάνου

A.M.: 1115201000109

ΕΠΙΒΛΕΠΟΝΤΕΣ: **Βασίλειος Γάτος**, Ερευνητής ΕΚΕΦΕ Δημόκριτος
Σέργιος Θεοδωρίδης Καθηγητής ΕΚΠΑ

ΠΕΡΙΛΗΨΗ

Στόχος της παρούσας πτυχιακής είναι η κατασκευή μιας εφαρμογής η οποία αφαιρεί το περιθώριο και διορθώνει την παραμόρφωση των εγγράφων που έχουν ληφθεί από κάμερα. Συγκεκριμένα, εφαρμόστηκαν και ελέγχθηκαν δύο μέθοδοι, η πρώτη βασίζεται στον Harris Corner Detection και η δεύτερη στον μετασχηματισμό Hough. Η πρώτη μέθοδος βρίσκει τα όρια του εγγράφου ελέγχοντας τις γωνίες της εικόνας ενώ η δεύτερη βρίσκει πρώτα τις ευθείες που σχηματίζονται από το περίγραμμα της εικόνας. Με βάση τα πειραματικά αποτελέσματα χρησιμοποιήθηκε η καλύτερη μέθοδος καθώς και ο καλύτερος συνδυασμός παραμέτρων σε ένα site όπου είναι διαθέσιμη on-line η συγκεκριμένη εφαρμογή.

ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ: Επεξεργασία Εικόνας

ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ: διόρθωση προοπτικής, αφαίρεση περιθωρίου, ανίχνευση ακμών, ανίχνευση γωνιών

ABSTRACT

The goal of this bachelor thesis is the creation of an application that rectifies a document's perspective distortion and removes its borders. The documents used are pictures taken by a camera. Specifically, the implementation of this application uses two methods. One method is based on Harris corner detection algorithm, while the other one uses Hough transformation. The first method finds document boundaries by specifying document corners, whereas the second one detects the maximum edges that make up the image's contour. We compare the two methods and show that the method that uses Hough transformation appears to be more effective. Based on the experiments conducted using each method, we extracted the best method and the most effective sets of parameters and used them in the final version of the application.

SUBJECT AREA: Image Processing

KEYWORDS: perspective correction, border removal, edge detection, corner detection

Η παρούσα πτυχιακή είναι αφιερωμένη στην οικογένειά μου

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να ευχαριστήσω τον επιβλέποντα της πτυχιακής μου, κύριο Βασίλειο Γάτο για το χρόνο που αφιέρωσε και την αφοσίωση που έδειξε στη σωστή καθοδήγησή μου και για την ευκαιρία που μου έδωσε να ασχοληθώ με έναν ερευνητικό τομέα παρόλο που ξεκινώντας δεν είχα όλο το απαραίτητο γνωστικό υπόβαθρο. Θα ήθελα επίσης να τον ευχαριστήσω για την πίστη του στις δυνατότητές μου και για την πολύτιμη γνώση και εμπειρία που απέκτησα κατά τη διάρκεια της συνεργασίας μας, η οποία θα επηρεάσει σημαντικά την υπόλοιπη ακαδημαϊκή μου πορεία. Επίσης θα ήθελα να ευχαριστήσω όλη την ομάδα του εργαστηρίου, κυρίως τον Γιώργο Ρετσινά, για την καθοδήγηση και τις πολύτιμες συμβουλές που βοήθησαν στην ολοκλήρωση της παρούσας πτυχιακής.

ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ	15
1. ΕΙΣΑΓΩΓΗ	16
2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ	18
2.1 Βελτίωση ποιότητας εικόνας	18
2.1.1 Μορφολογικά φίλτρα	18
2.1.2 Dilation	18
2.1.3 Erosion	19
2.1.4 Opening.....	19
2.1.5 Closing	19
2.2 Εντοπισμός Ακμών.....	20
2.2.1 Sobel Operator	21
2.2.1 Canny edge detection	23
2.3 Εντοπισμός γωνιών με τον Αλγόριθμο Harris	29
2.3 Εντοπισμός ευθειών	32
2.3.1 Hough Transform	33
2.4 Homography	38
3. ΤΕΧΝΙΚΕΣ ΔΙΟΡΘΩΣΗΣ ΠΡΟΟΠΤΙΚΗΣ ΚΑΙ ΑΦΑΙΡΕΣΗΣ ΠΕΡΙΘΩΡΙΟΥ.....	40
3.1 Υπάρχουσες Τεχνικές.....	40
3.2 Μεθοδολογίες που χρησιμοποιούν την εύρεση των ορίων του εγγράφου	41
3.2.1 Διόρθωση της προοπτικής παραμόρφωσης με τη χρήση του Harris Corner detector	42
3.2.2 Διόρθωση της προοπτικής παραμόρφωσης με τη χρήση του Hough Transform.....	44
3.3 Μεθοδολογίες που χρησιμοποιούν τη διάταξη της σελίδας.	49
3.4 Μεθοδολογίες που χρησιμοποιούν ειδικές διορθώσεις περιεχομένου	50
3.5 Εργαλεία και υπάρχουσες εφαρμογές	52
4. ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΜΕΘΟΔΟΙ	55
4.1 Μέθοδος Harris.....	55

4.2 Μέθοδος Hough.....	63
4.3 Homography	73
5. ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ	76
5.1 Evaluation	76
6. ΥΛΟΠΟΙΗΣΗ	89
6.1 Προγράμματα - Συναρτήσεις	90
6.2 Web Εφαρμογή	98
6.2.1 Απαιτήσεις του συστήματος	99
6.2.2 Ανάπτυξη	99
7. ΣΥΜΠΕΡΑΣΜΑΤΑ.....	105
ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ	107
ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ	109
ΑΝΑΦΟΡΕΣ	110

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

Σχήμα 1: Διάγραμμα ροής αλγορίθμου Harris	57
Σχήμα 2: Διάγραμμα ροής μετασχηματισμού Hough	67
Σχήμα 3 : Διάγραμμα Classic Hough : Ανίχνευση τεσσάρων ολικών μεγίστων σε ποσοστά ανοχής 5-20%.....	83
Σχήμα 4: Διάγραμμα Hough_VP: Ανίχνευση τεσσάρων τοπικών μεγίστων σε ποσοστά ανοχής 5-20%.....	83
Σχήμα 5: Διάγραμμα μεθόδων Harris σε ποσοστά ανοχής 5-20%	84
Σχήμα 6: Διάγραμμα Hough_Th : Ανίχνευση με threshold.....	84

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

Εικόνα 1: sobel φίλτρο ως προς x και y κατεύθυνση	22
Εικόνα 2: Αρχική εικόνα	25
Εικόνα 3: Gaussian blur.....	26
Εικόνα 4: Finding Gradients	26
Εικόνα 5: Finding Gradients	27
Εικόνα 6: Non max suppression	27
Εικόνα 7: Non max suppression	27
Εικόνα 8: Double Thresholding.....	27
Εικόνα 9: Edge tracking by hysteresis	28
Εικόνα 10: Final Image	28
Εικόνα 11: Αντιστοίχιση μιας μοναδικής γραμμής σε ένα σημείο στον Hough χώρο ...	33
Εικόνα 12: Αντιστοίχιση κάθε σημείου σε μια ευθεία και την ευθεία που διέρχεται ως το σημείο τομής στον Hough χώρο.....	34
Εικόνα 13: Η ευθεία που διέρχεται από δύο σημεία στον Ευκλείδειο χώρο είναι το σημείο που τέμνονται οι ευθείες των σημείων στο χώρο Hough	35
Εικόνα 14: Δεξιά η αρχική εικόνα σε grayscale, αριστερά η εικόνα μετά από εφαρμογή του αλγόριθμου Canny	36
Εικόνα 15: Δεξιά η ανίχνευση ευθειών και αριστερά η μετατροπή τους σε πεπερασμένες	37
Εικόνα 16: Ο μετασχηματισμός της επιφάνειας με τη χρήση Homography.....	38
Εικόνα 17: Αλλοίωση των διαστάσεων του αντικειμένου μέσω της κάμερας	40
Εικόνα 18: Βήματα της μεθόδου Geetha Kiran A. και Murali S.	44
Εικόνα 19: Βήματα της μεθόδου R.Baumann, C.Blackwell και W. B. Seales	46
Εικόνα 20: Border noise removal through page frame detection	49
Εικόνα 21: Correction of Perspective Distortion using Layout Information. Κατάτμηση σε εικόνα , κείμενο , γραφικά.	50

Εικόνα 22: Perspective correction of Hindi Text. Προσδιορισμός των οριζόντιων και καθετων vanishing points	51
Εικόνα 23: Finescanner	53
Εικόνα 24: Genius Scan	53
Εικόνα 25: Office Lens.....	54
Εικόνα 26:Scannable.....	54
Εικόνα 27: Capture Board	55
Εικόνα 28: Εφαρμογή του τελεστή Sobel.....	59
Εικόνα 29: Εφαρμογή του JFC	59
Εικόνα 30: Εφαρμογή του τελεστή Sobel.....	59
Εικόνα 31: Εφαρμογή του JFC	59
Εικόνα 32: Dilation με element size (3,3).....	60
Εικόνα 33: Dilation με element size (5,5).....	60
Εικόνα 34: Dilation με element size (3,3).....	60
Εικόνα 35: Dilation με element size (5,5).....	60
Εικόνα 36: Find Max Contour	61
Εικόνα 37: Αρχική εικόνα.....	61
Εικόνα 38: Harris Corner Detection με Blocksize = 3	61
Εικόνα 39: Harris Corner Detection με Blocksize = 4	61
Εικόνα 40: Harorner Detection με Blocksize = 5.....	61
Εικόνα 41: Επιλογή σημείων	62
Εικόνα 42: Harris με εφαρμογή του Canny edge detector	63
Εικόνα 43: Harris με εφαρμογή του Sobel φίλτρου	63
Εικόνα 44: Harris με εφαρμογή του Canny edge detector	63
Εικόνα 45: Harris με εφαρμογή του Sobel φίλτρου	63
Εικόνα 46: Αρχική εικόνα με παραμόρφωση προοπτικής.....	69
Εικόνα 47: Ανίχνευση ακμών με τον αλγόριθμο JF Canny	69

Εικόνα 48: Accumulator των εικόνων 31, 32, 33, 34.....	70
Εικόνα 49: Η δεξιά στήλη εικόνων παρουσιάζει την εύρεση ευθειών με τη χρήση τεσσάρων τοπικών μεγίστων (μέθοδος 2) και η αριστερή στήλη απεικονίζει την επιλογή ευθειών με τη χρήση της κατωφλίωσης.	71
Εικόνα 50: Σύγκριση μεθόδων ανίχνευσης ευθειών. Τελικό Στάδιο -ανίχνευσης γωνιών. Η πρώτη στήλη παρουσιάζει τη Classic Hough η δεύτερη τη Hough_Vertical_Parallel και η τρίτη Hough_Threshold 3 στο τελικό στάδιο πριν την εφαρμογή ομογραφίας.	73
Εικόνα 51: Στάδιο διόρθωσης της προοπτικής παραμόρφωσης με την εφαρμογή προβολικού μετασχηματισμού (ομογραφίας). Στη δεξιά στήλη βρίσκονται οι αρχικές παραμορφωμένες εικόνες ενώ αριστερά οι τελικές.....	76
Εικόνα 52: Ανίχνευση ακμών	77
Εικόνα 53: Ανίχνευση ευθειών	77
Εικόνα 54: Εύρεση τεσσάρων γωνιών.....	77
Εικόνα 55: Ανίχνευση ακμών	78
Εικόνα 56: Ανίχνευση ευθειών	78
Εικόνα 57: Εύρεση γωνιών.....	78
Εικόνα 58: Ανίχνευση ακμών	78
Εικόνα 59: Ανίχνευση ευθειών.....	78
Εικόνα 60: Εύρεση γωνιών.....	78
Εικόνα 61: Hough_VP λανθασμένη ανίχνευση ευθειών.....	86
Εικόνα 62: Hough_Th σωστή ανίχνευση ευθειών	86
Εικόνα 63: Γωνίες που προκύπτουν από την Εικόνα 61	86
Εικόνα 64: Γωνίες που προκύπτουν από την Εικόνα 62	86
Εικόνα 65: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP, δεξιά το αποτέλεσμα της Hough_Th.....	87
Εικόνα 66: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP δεξιά το αποτέλεσμα της Hough_Th.....	87
Εικόνα 67: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP, δεξιά το αποτέλεσμα της Hough_Th.....	87

Εικόνα 68: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP, δεξιά το αποτέλεσμα της Hough_Th.....	88
Εικόνα 69: Αρχική σελίδα	101
Εικόνα 70: Upload Image	102
Εικόνα 71: Homography with Corner Detection	103
Εικόνα 72: Homography with Hough Transform	104

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

Πίνακας 1: Αλγόριθμος Harris.....	30
Πίνακας 2: Παράμετροι προγράμματος	80
Πίνακας 3: Αποτελέσματα εκτέλεσης των διαφορετικών συνδυασμών τιμών παραμέτρων των μεθόδων Hough Transform	80
Πίνακας 4: Αποτελέσματα εκτέλεσης των διαφορετικών συνδυασμών τιμών παραμέτρων των μεθόδων Harris.....	82
Πίνακας 5: Συναρτήσεις της υλοποίησης Hough	94
Πίνακας 6: Συναρτήσεις της υλοποίησης Harris	96
Πίνακας 7 : Συναρτήσεις της υλοποίησης Homography	98

ΠΡΟΛΟΓΟΣ

Η παρουσίαση της μελέτης που ακολουθεί εντάσσεται στα πλαίσια της πτυχιακής εργασίας του Προγράμματος Σπουδών του τμήματος Πληροφορικής και Τηλεπικοινωνιών του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών (ΕΚΠΑ). Η επιλογή του θέματος προέκυψε μετά από πρόταση του επιβλέποντα της πτυχιακής μου, κυρίου Βασίλειου Γάτου.

Η διόρθωση της προοπτικής παραμόρφωσης σε εικόνες από κάμερα είναι ένα καινούριο ερευνητικό κομμάτι στον τομέα της Επεξεργασίας εικόνας με πολύ ενδιαφέρον. Πολλές εφαρμογές και εργαλεία έχουν αναπτυχθεί και αναπτύσσονται βασισμένα σε αυτό το ερευνητικό κομμάτι, γνωστά με τον όρο document scanners. Στόχος της πτυχιακής είναι η μελέτη των μεθόδων επεξεργασίας εικόνας που χρησιμοποιούν υπάρχουσες γνωστές εφαρμογές και εργαλεία της αγοράς και η υλοποίηση μιας παρόμοιας εφαρμογής στηριγμένη σε αντίστοιχες μεθοδολογίες.

Κατά την ασχολία μου με την παρούσα εργασία, εμβάθυνα σε διάφορους τομείς της επεξεργασίας εικόνας και της ανάλυσης εγγράφων που με οδήγησαν στην απόκτηση καινούριων ενδιαφερόντων.

Πριν την παρουσίαση της εργασίας αυτής, θα ήθελα ακόμη μια φορά να ευχαριστήσω θερμά τον υπεύθυνο καθηγητή μου για την υπομονή και το ενδιαφέρον που έδειξε, καθώς και για τις στοχευμένες παρατηρήσεις που έπαιξαν σημαντικό ρόλο στη σωστή καθοδήγησή μου. Τέλος, θέλω να ευχαριστήσω θερμά όλους όσους μου συμπαραστάθηκαν ή και με ανέχθηκαν κατά τη διάρκεια της εκπόνησής της

1. ΕΙΣΑΓΩΓΗ

Τόσο στην κοινωνική ζωή όσο και στον εργασιακό τομέα γίνεται ολοένα και πιο επιτακτική η ανάγκη για κοινή χρήση και διαμοίραση ψηφιακών εγγράφων. Άμεσος και εύκολος τρόπος για την αποτύπωση ψηφιακών εγγράφων είναι οι φωτογραφίες. Σημαντικός παράγοντας γι' αυτό αποτελούν τα smartphones τα οποία δίνουν στον χρήστη την ευκαιρία της ψηφιοποίησης των εγγράφων βάζοντας απλώς το έγγραφο κάτω από την κάμερα του smartphone. Η ευκολία και απλότητα που προσφέρουν σα λύση καθώς και η δυνατότητά τους να υποστηρίζουν τη λήψη και ψηφιοποίηση εγγράφων χάρη στα χαρακτηριστικά της τεχνολογίας τους τα καθιστούν πρώτη επιλογή. Σήμερα παρατηρείται όλο και πιο έντονο το φαινόμενο της χρήσης των κινητών συσκευών για την λήψη και αποθήκευση εγγράφων σε σχέση με τη χρήση του scanner (σαρωτή). Σε αντίθεση όμως με τα scanner, που χρησιμοποιούν την άμεση επαφή με το έγγραφο ώστε αυτό να αντιγραφεί και να ψηφιοποιηθεί, οι σύγχρονες κάμερες εισάγουν νέους κινδύνους παραμορφώσεων στο έγγραφο. Ο θόρυβος που προέρχεται από το φωτισμό, την ένταση καθώς και το υπόβαθρο, η στρέβλωση της εικόνας, η καμπυλότητα είναι μερικά από τα νέα προβλήματα που εισάγει η χρήση της κάμερας. Το πιο σημαντικό είναι η παραμόρφωση της προοπτικής, το φαινόμενο κατά το οποίο το εικονιζόμενο αντικείμενο παρουσιάζεται με αλλοιωμένες διαστάσεις λόγω της θέσης του παρατηρητή φωτογράφου. Το πρόβλημα αυτό δεν μπορεί να λυθεί με απλές μορφές επεξεργασίας, όπως περικοπή (cut) και διόρθωση στροφής (rotate), αλλά είναι απαραίτητη η χρήση γεωμετρικών μετασχηματισμών για την επαναφορά της εικόνας στην αρχική της μορφή. Στόχος αυτής της πτυχιακής είναι η μελέτη και εύρεση μεθόδων διόρθωσης της λανθασμένης προοπτικής της εικόνας που προκύπτει λόγω της παραμόρφωσης των φακών. Παρουσιάζονται διάφορες υπάρχουσες τεχνικές, δυο εκ των οποίων γίνεται εκτενέστερη μελέτη και υλοποίηση.

Για να είναι ποιο εύκολη η κατανόηση και η αξιολόγηση της εργασίας από τον αναγνώστη, έχει χωριστεί σε κεφάλαια. Το πρώτο κεφάλαιο, θα ξεκινήσει με μια σύντομη εισαγωγή και θα κλείσει με μια γενική επισκόπηση στα ερευνητικά πεδία και στις τεχνολογίες που συμμετέχουν στην εργασία.

Στο Δεύτερο κεφάλαιο, θα επεξηγηθούν έννοιες κλειδιά στην εργασία, όπως είναι το στάδιο της προ επεξεργασίας της εικόνας και οι μέθοδοι βελτίωσης της ποιότητας. Στη συνέχεια, θα αναλυθούν μέθοδοι εξαγωγής των χαρακτηριστικών της εικόνας (feature extraction) όπως είναι η ανίχνευση γωνιών (corner detection), ευθειών (line detection) και ακμών (edge detection). Στο κεφάλαιο αυτό παρουσιάζονται γνωστοί αλγόριθμοι της επεξεργασίας εικόνας και αποτελεί ως θεωρητική βάση για την ανάπτυξη των επόμενων κεφαλαίων. Στο τέλος του κεφαλαίου αυτού ο αναγνώστης πρέπει να είναι σε θέση να κατανοεί τις βασικές έννοιες που χρησιμοποιούνται στην εργασία.

Στο τρίτο κεφάλαιο, θα αναφερθούμε σε υπάρχουσες μεθοδολογίες που λύνουν το πρόβλημα της προοπτικής παραμόρφωσης και θα εστιάσουμε στην ανάπτυξη των δυο μεθοδολογιών που χρησιμοποιήθηκαν. Θα δείξουμε πως χρησιμοποιούνται αλγόριθμοι

που ήδη αναφέρθηκαν στο δεύτερο κεφάλαιο και πώς μπορούν να συνδυαστούν για να επιτευχθεί το επιθυμητό αποτέλεσμα.

Στο τέταρτο κεφάλαιο, θα επικεντρωθούμε σε κάποιες βελτιώσεις και παραλλαγές των αναφερόμενων μεθοδολογιών για την καλύτερη αντιμετώπιση του προβλήματος και την εξαγωγή εγκυρότερων αποτελεσμάτων. Για κάθε μέθοδο που χρησιμοποιήθηκε θα περιγραφούν αναλυτικά όλα της τα στάδια και θα γίνει αναφορά όλων των επεκτάσεων που χρειάστηκαν για την ανάπτυξη της τελικής υλοποίησης .

Στο πέμπτο κεφάλαιο, θα παρουσιαστούν τα πειραματικά αποτελέσματα της ανίχνευσης και της κατηγοριοποίησης και θα αποτιμηθούν οι μέθοδοι που χρησιμοποιήθηκαν με την παράθεση διαγραμμάτων.

Στο έκτο κεφάλαιο, θα γίνει ανάλυση της εφαρμογής. Το κεφάλαιο αυτό χωρίζεται σε δύο βασικά μέρη. Το πρώτο, θα ασχοληθεί με τα προγράμματα που υλοποιήθηκαν για την επεξεργασία της εικόνας, δηλαδή την ανίχνευση των γωνιών, την αφαίρεση του περιθωρίου και τη διόρθωση της προοπτικής. Το δεύτερο μέρος, θα εστιάσει στην εφαρμογή που υλοποιήθηκε με βάση τις παραπάνω τεχνικές και ο χρήστης μπορεί να χρησιμοποιήσει ώστε να επιβεβαιώσει οπτικά την εγκυρότητα των μεθόδων.

Τέλος, το έβδομο κεφάλαιο θα αποτελέσει μια σύνοψη όσων μελετήθηκαν και θα εξαχθούν συμπεράσματα με βάση τα αποτελέσματα που παρουσιάζονται στο πέμπτο κεφάλαιο καθώς και για την όλη πορεία της συγκεκριμένης εργασίας.

2. ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

2.1 Βελτίωση ποιότητας εικόνας

2.1.1 Μορφολογικά φίλτρα

Η μαθηματική μορφολογία είναι ένα εργαλείο που συμβάλλει στην επεξεργασία των εικόνων για την εξαγωγή πληροφοριών και είναι χρήσιμη για τον προσδιορισμό του περιγράμματος, την εύρεση ακμών του αντικειμένου, την βελτίωση της ανάλυσης, την αφαίρεση θορύβων και την κατάτμηση της εικόνας [1]. Οι μορφολογικοί τελεστές είναι μέθοδοι για την επεξεργασία δυαδικών ή grayscale εικόνων βασισμένοι στη γεωμετρία που παρουσιάζουν.

Οι δύο βασικότεροι τελεστές στη μαθηματική μορφολογία είναι η συστολή (erosion) και η διαστολή (dilation). Αυτές οι λειτουργίες συνήθως χρησιμοποιούνται ταυτόχρονα. Ο τρόπος με τον οποίο λειτουργούν οι τελεστές είναι ο εξής: λαμβάνουν ως είσοδο δεδομένων μια εικόνα και ένα δομικό στοιχείο και στην συνέχεια εφαρμόζουν στην εικόνα ένα τελεστή και επιστρέφουν σαν έξοδο την επεξεργασμένη εικόνα η οποία έχει τις ίδιες διαστάσεις με την αρχική.

Το δομικό στοιχείο το οποίο είναι βασικό μέρος των μορφολογικών τελεστών μπορεί να είναι ακαθόριστου σχήματος. Συνήθως το μέγεθός του είναι 3×3 και μπορεί να έχει τύπο τετράγωνο, σταυρωτό ή ελλειψοειδές.

2.1.2 Dilation

Το βασικό αποτέλεσμα αυτού του τελεστή σε μια δυαδική εικόνα είναι η σταδιακή διεύρυνση των ορίων. Αυτό έχει ως αποτέλεσμα προσθήκη pixel στα όρια του αντικειμένου με αποτέλεσμα να μεγαλώνει το μέγεθος του ενώ μικραίνουν οι τρύπες που υπάρχουν στις περιοχές αυτές. Το δομικό στοιχείο καθορίζει επακριβώς την επίδραση που θα έχει ο τελεστής διαστολής στην αρχική εικόνα που δίνουμε ως είσοδο ενώ το μέγεθός του καθορίζει την περιοχή που θα εφαρμοστεί, επομένως και το μέγεθος της διαστολής που θα υποστεί η εικόνα.

Ας υποθέσουμε ότι E είναι ο Ευκλείδειος χώρος, A είναι το σύνολο των ευκλείδειων συντεταγμένων που αντιστοιχούν στη δυαδική είσοδο της εικόνας, B το σύνολο των συντεταγμένων για το δομικό στοιχείο και Bz είναι μια μεταφορά του B έτσι ώστε η αρχή του να είναι το z . Η διαστολή του A από το B ορίζεται το σύνολο όλων των σημείων z της τομής του Bz με το A όπου το A είναι διάφορο του κενού. Η μαθηματική έκφραση της διαστολής είναι η εξής:

$$A \oplus B = \{z \in E | (B_z) \cap A \neq \emptyset\} \quad (1)$$

2.1.3 Erosion

Ο τελεστής της συστολής ή διάβρωσης είναι ο δεύτερος βασικός τελεστής και εμφανίζει το αντίθετο αποτέλεσμα με τον προηγούμενο τελεστή. Διαβρώνει τα όρια που βρίσκονται στο προσκήνιο και οδηγεί σε συρρίκνωση των ρixel σε αυτές τις περιοχές ενώ μεγαλώνει τις τρύπες. Το αποτέλεσμα του erosion είναι η λέπτυνση των ορίων του αντικειμένου έτσι ώστε το αποτέλεσμα της εικόνας να είναι ποιοτικότερο. Και σε αυτήν την περίπτωση εφαρμόζεται όπως στη διαστολή ένα δομικό στοιχείο που καθορίζει πόσο πρόκειται να συσταλεί η εικόνα και με ποιον τρόπο. Μπορεί να χρησιμοποιηθεί για να διαχωρίσει αντικείμενα και να αφαιρέσει μικρές προεξοχές. Με βάση τον παραπάνω μαθηματικό τύπο, η συστολή του A από το B ορίζεται ως εξής:

$$A \ominus B = \{z \in E | (B_z) \cap A \neq \emptyset\} \quad (2)$$

Με τη χρήση των δυο τελεστών που ορίστηκαν, μπορούν να προκύψουν οι συνδυασμοί τους opening και closing που ικανοποιούν διαφορετικά αιτήματα της προ-επεξεργασίας.

2.1.4 Opening

Προκύπτει από χρήση πρώτα του τελεστή erosion και ακολουθείται από dilation. Το βασικό αποτέλεσμα του opening είναι περίπου το ίδιο με του erosion χωρίς να έχει τον ίδιο βαθμό διάβρωσης. Η επίδραση του τελεστή είναι να διατηρήσει στο προσκήνιο περιοχές που έχουν παρόμοιο σχήμα με το δομικό στοιχείο ή περιοχές που περιέχουν εξ ολοκλήρου το δομικό στοιχείο εξαλείφοντας όλες τις υπόλοιπες περιοχές του foreground ρixel. Επίσης, εξομαλύνει το περίγραμμα ενός αντικειμένου εξαλείφοντας τις προεξοχές και σπάζοντας μικρά συνδεδεμένα τμήματα. Ο μαθηματικός τύπος του opening ορίζεται ως εξής:

$$A \circ B = (A \ominus B) \oplus B \quad (3)$$

2.1.5 Closing

Πρόκειται για το αντίστροφο του opening, δηλαδή προκύπτει από χρήση πρώτα του τελεστή dilation και ακολουθείται από erosion. Ο τελεστής αυτός εξομαλύνει το περίγραμμα ενός αντικειμένου συγχωνεύοντας τα στενά θραύσματα και συμπληρώνοντας τις οπές. Ο μαθηματικός τύπος του closing ορίζεται ως εξής:

$$A \cdot B = (A \oplus B) \ominus B \quad (4)$$

2.2 Εντοπισμός Ακμών

Ως ακμές, ορίζουμε τις θέσεις σε μια εικόνα οι οποίες αντιστοιχούν στα όρια του αντικειμένου που απεικονίζεται, δηλαδή το περίγραμμά του. Σε μια ασπρόμαυρη εικόνα ο όρος ακμές, αναφέρεται σε αλλαγές που παρουσιάζει η εικόνα ως προς την φωτεινότητα και την ένταση μεταξύ των γειτονικών περιοχών της. Οι αλλαγές αυτές, συνήθως αντιστοιχούν σε διαφοροποίηση ιδιοτήτων της απεικόνισης τρισδιάστατων αντικειμένων, όπως αλλαγές της υφής, του βάθους, των ορίων των αντικειμένων, το διαφορετικό φωτισμό και την αντανάκλαση. Επομένως, η ανίχνευση ακμών αναφέρεται στη διαδικασία αναγνώρισης και εντοπισμού απότομων αλλαγών στην ένταση των pixel (ασυνέχειες) που χαρακτηρίζουν τα όρια των αντικειμένων σε μια σκηνή.

Η ανίχνευση των ορίων ενός εγγράφου και των χαρακτηριστικών του είναι πολύ σημαντική για την επεξεργασία του εγγράφου και την ορθή αναγνώριση του κειμένου. Ο εντοπισμός ακμών αποτελεί την βάση για μετέπειτα επεξεργασία μιας εικόνας γι' αυτό πρέπει να δίνει αξιόπιστα αποτελέσματα και να υλοποιείται αποδοτικά. Ωστόσο, ο εντοπισμός των ακμών μιας εικόνας παρουσιάζει αρκετές δυσκολίες λόγω των δυσδιάκριτων αλλαγών στην φωτεινότητα της καθώς και της παρουσίας θορύβου που μπορεί να προέρχεται από διαφορετικό φωτισμό και σκίαση. Όλα αυτά μπορούν να οδηγήσουν στην ανίχνευση εσφαλμένων ακμών αλλοιώνοντας τα όρια των αντικειμένων

Κλασικές μέθοδοι ανίχνευσης ακμών [15] – [17] περιλαμβάνουν συνέλιξη της εικόνας με έναν τελεστή (ένα φίλτρο 2-D), ο οποίος είναι κατασκευασμένος να είναι ευαίσθητος σε εικόνες με μεγάλες κλίσεις, ενώ επιστρέφει τιμές του μηδενός σε ομοιόμορφες περιοχές. Υπάρχει ένας εξαιρετικά μεγάλος αριθμός τελεστών εύρεσης ακμών, καθένας σχεδιασμένος με διαφορετική ευαισθησία ανάλογα με τον τύπο ακμών. Μεταβλητές που εμπλέκονται στην επιλογή κάθε τελεστή περιλαμβάνουν τον προσανατολισμό των ακμών, τη δομή τους καθώς και το περιβάλλον θορύβου. Κάθε τελεστής έχει τη δική του γεωμετρία που προδίδει μια χαρακτηριστική κατεύθυνση στην οποία είναι πιο ευαίσθητος στην ανίχνευση. Οι τελεστές μπορούν να διαμορφωθούν ώστε να εντοπίζουν μόνο οριζόντιες, κάθετες, ή διαγώνιες ακμές.

Υπάρχουν πολλοί τρόποι για να διεκπεραιωθεί η ανίχνευση ακμών. Εν τούτοις, η πλειοψηφία των διαφόρων μεθόδων μπορεί να χωριστεί σε δυο κατηγορίες, αυτή της βαθμωτής μεταβολής (gradient) και του Laplace. Η μέθοδος της βαθμωτής μεταβολής, ανιχνεύει τις ακμές εντοπίζοντας το μέγιστο και το ελάχιστο της πρώτης παραγώγου της εικόνας.

2.2.1 Sobel Operator

Ο τελεστής Sobel [3] χρησιμοποιείται στην επεξεργασία εικόνας, κυρίως στους αλγόριθμους ανίχνευσης ακμών. Είναι ένας διακριτός τελεστής διαφοροποίησης που υπολογίζει μια προσέγγιση της κλίσης της συνάρτησης έντασης της εικόνας. Σε κάθε σημείο της εικόνας, το αποτέλεσμα του τελεστή Sobel είναι είτε το αντίστοιχο διάνυσμα κλίσης, είτε το μέτρο του διανύσματος. Η μέθοδος του τελεστή Sobel βασίζεται στη συνέλιξη της εικόνας με ένα μικρό, διαχωριστικό φίλτρο αέριων τιμών στην οριζόντια και κάθετη κατεύθυνση.

Σχεδίαση φίλτρου Sobel

Οι περισσότερες μέθοδοι ανίχνευσης ακμών λειτουργούν υπό την υπόθεση ότι μια ακμή θα εντοπίζεται σε σημεία ασυνέχειας στη συνάρτηση συχνότητας ή μεγάλης μεταβολής στην ένταση μεταξύ γειτονικών pixel που υπολογίζεται μέσω της κλίσης (gradient) [3] [4]. Με βάση αυτή την υπόθεση, αν υπολογίσουμε το gradient της έντασης σε όλη την εικόνα και βρούμε τα σημεία στα οποία μεγιστοποιείται, τότε θεωρητικά βρίσκουμε την ακμή. Η μεταβολή της εξομάλυνσης της εικόνας παρουσιάζεται ως είναι ένα διάνυσμα του οποίου τα στοιχεία δείχνουν πόσο γρήγορα αλλάζουν οι τιμές των pixel με την απόσταση στις δύο κατευθύνσεις. Έτσι, τα στοιχεία του διανύσματος μπορούν να βρεθούν χρησιμοποιώντας την ακόλουθη προσέγγιση:

$$\frac{\partial f(x, y)}{\partial x} = \Delta x = \frac{f(x + dx, y) - f(x, y)}{dx} \quad (5)$$

$$\frac{\partial f(x, y)}{\partial y} = \Delta y = \frac{f(x, y + dy) - f(x, y)}{dy} \quad (6)$$

όπου dx και dy δείχνουν την απόσταση στη κατεύθυνση x και y αντίστοιχα. Σε διακριτές εικόνες, μπορούμε να σκεφτόμαστε τα dx και dy ως τον αριθμό των pixel ανάμεσα σε δύο σημεία. $dx = dy = 1$ είναι το σημείο με συντεταγμένες pixel (i, j) και έτσι :

$$\Delta x = f(i + 1, j) - f(i, j) \quad (7)$$

$$\Delta y = f(i, j + 1) - f(i, j) \quad (8)$$

Για να μπορέσουμε να ανιχνεύσουμε την ύπαρξη ασυνέχειας της συχνότητας, μπορούμε να υπολογίσουμε την αλλαγή της κλίσης στο (i, j) . Αυτό μπορεί να γίνει βρίσκοντας το ακόλουθο μέτρο μεγέθους:

$$M = \sqrt{(\Delta x)^2 + (\Delta y)^2} \quad (9)$$

και η κατεύθυνση της θ δίνεται από τον τύπο:

$$\theta = \arctan\left(\frac{\Delta y}{\Delta x}\right) \quad (10)$$

Ο τελεστής Sobel είναι ένα παράδειγμα σχεδίασης φίλτρου με τη μέθοδο εύρεσης gradient. Είναι ένας διακριτός τελεστής διαφοροποίησης που υπολογίζει μια προσέγγιση της κλίσης της συνάρτησης έντασης της εικόνας. Οι διαφορετικοί τελεστές στις εξισώσεις (19) και (20) αντιστοιχούν στην συνέλιξη της εικόνας με τις ακόλουθες μάσκες:

$$\Delta x = \begin{bmatrix} -1 & 1 \\ 0 & 0 \end{bmatrix}, \Delta y = \begin{bmatrix} -1 & 0 \\ 1 & 0 \end{bmatrix}$$

- i. Στη συνέχεια τα βήματα είναι : Η πάνω αριστερή γωνία της αντίστοιχης μάσκας εφαρμόζεται πάνω σε κάθε pixel της εικόνας
- ii. Υπολογίζεται μια τιμή για το Δx ή Δy χρησιμοποιώντας τους συντελεστές της μάσκας σε ένα άθροισμα με βάρη της τιμής των pixel (i, j) και των γειτονικών.
- iii. Αντί να βρίσκουμε προσεγγιστικά τα gradient στις κατευθύνσεις x και y , η προσέγγιση των στοιχείων της κλίσης μπορεί να γίνει στις κατευθύνσεις 45° και 135° από τους αντίστοιχους άξονες.

Το μέγεθος της μάσκας καθορίζει τις αποστάσεις μεταξύ δύο γειτονικών ακμών και τη λεπτομέρεια ανίχνευσής τους. Οι μάσκες που αναφέρθηκαν παραπάνω είναι κάπως μικρές και μπορεί να θεωρηθούν ακατάλληλες. Ωστόσο, η χρήση μιας μεγάλης μάσκας παρουσιάζει το πλεονέκτημα της μείωσης των λαθών λόγω θορύβου, παίρνοντας ένα τοπικό μέσο στην γειτονιά που ορίζει το μέγεθος της. Από την άλλη, με τη χρήση μιας μάσκας περιττού μεγέθους δίνεται περισσότερο βάρος στο κεντρικό pixel και επιτυγχάνεται καλύτερη εκτίμηση που βασίζεται σε αυτό.

Παρακάτω παρουσιάζονται οι μάσκες του τελεστή ακμών Sobel:

$$\Delta x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}, \Delta y = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Εικόνα 1: sobel φίλτρο ως προς x και y κατεύθυνση

Ο τελεστής υπολογίζει την κλίση της έντασης της εικόνας σε κάθε σημείο, δίνοντας την κατεύθυνση της μεγαλύτερης δυνατής αύξησης από ανοιχτό σε σκούρο και τον ρυθμό αλλαγής στην κατεύθυνση αυτή. Έτσι, το αποτέλεσμα δείχνει πόσο 'απότομα' ή 'ήπια'

αλλάζει η ένταση της εικόνας στο σημείο αυτό και πόσο πιθανό είναι το κομμάτι αυτό της εικόνας να αντιστοιχεί σε μια ακμή, καθώς και την πιθανότητά αυτή η ακμή να είναι προσανατολισμένη. Πρακτικά, ο υπολογισμός του μέτρου gradient είναι πιο αξιόπιστος και εύκολος από τον υπολογισμό κατεύθυνσης. Μαθηματικά, η κλίση μιας συνάρτησης δύο μεταβλητών (συνάρτησης έντασης εικόνας) σε κάθε εικόνα είναι ένα δισδιάστατο διάνυσμα με στοιχεία που δίνονται από τις παραγώγους στην οριζόντια και στην κάθετη διεύθυνση. Σε κάθε σημείο, το διάνυσμα κλίσης δείχνει προς την κατεύθυνση της μεγαλύτερης δυνατής αύξησης, και το μέγεθος του διανύσματος κλίσης αντιστοιχεί στον ρυθμό μεταβολής εκείνης της κατεύθυνσης. Αυτό σημαίνει ότι το αποτέλεσμα του τελεστή Sobel σε κάποιο σημείο που βρίσκεται σε μια περιοχή συνεχής έντασης της εικόνας είναι το μηδενικό διάνυσμα και σε κάποιο σημείο σε μια ακμή είναι ένα διάνυσμα που δείχνει πάνω από την ακμή, από σκούρες σε πιο φωτεινές τιμές.

Ο τελεστής Sobel, εστιάζει σε περιοχές υψηλής χωρικής συχνότητας οι οποίες αντιστοιχούν σε ακμές. Η μέθοδος αυτή υπολογίζει την απόκλιση της έντασης σε κάθε pixel και ως αποτέλεσμα βγάζει την πιο πιθανή μέγιστη αύξηση από τις πιο φωτεινές περιοχές στις σκουρόχρωμες καθώς και τη μεταβολή αυτής. Παρουσιάζει δηλαδή τη μεταβολή της εξομάλυνσης της εικόνας, μεταβολή για το κάθε pixel (gray level value), γι' αυτό και τα σημεία με τη μεγαλύτερη μεταβολή έχουν τη μεγαλύτερη πιθανότητα να είναι ακμές.

Τα βασικά πλεονεκτήματα του τελεστή Sobel βρίσκονται στην απλότητά του και στο γεγονός ότι μπορεί να ανιχνεύσει ακμές και προσανατολισμούς ακμών εύκολα λόγω της προσέγγισης του μεγέθους της κλίσης. Ωστόσο, ένα μειονέκτημα είναι ευαισθησία που παρουσιάζει στο θόρυβο. Το μέγεθος των ακμών θα υποβιβαστεί όσο αυξάνεται το επίπεδο θορύβου στην εικόνα και αυτό έχει ως αποτέλεσμα να χάνεται η ακρίβεια του τελεστή Sobel όσο μειώνεται το μέγεθος των ακμών. Γενικά, η μέθοδος Sobel δεν μπορεί να παράγει ακριβή ανίχνευση ακμών με λεπτές και λείες ακμές.

2.2.1 Canny edge detection

Η ανίχνευση ακμών αποσκοπεί στο να μειώσει σημαντικά την ποσότητα των δεδομένων μίας εικόνας διατηρώντας παράλληλα τις δομικές της ιδιότητες που είναι αναγκαίες για την περαιτέρω επεξεργασία της. Πολλοί αλγόριθμοι έχουν αναπτυχθεί και ένας από τους σημαντικότερους είναι ο αλγόριθμος Canny που αναπτύχθηκε από τον JohnF. Canny το 1986 [4]. Παρόλο που είναι αρκετά παλιός αλγόριθμος, έχει γίνει ένας από τους πρότυπους αλγορίθμους για την ανίχνευση ακμών και χρησιμοποιείται ακόμα στον τομέα της έρευνας. Ο στόχος του JFC ήταν να αναπτύξει έναν βέλτιστο αλγόριθμο ως προς τα παρακάτω κριτήρια:

Ανίχνευση: Πρώτο κριτήριο αποτελεί η ελαχιστοποίηση του σφάλματος. Η πιθανότητα ανίχνευσης πραγματικών ακμών θα πρέπει να μεγιστοποιηθεί ενώ ταυτόχρονα να ελαχιστοποιηθεί η λανθασμένη ανίχνευση σημείων που δεν αποτελούν ακμές. Αυτό ουσιαστικά αντιστοιχεί σε μεγιστοποίηση του λόγου σήματος προς θόρυβο.

Εντοπισμός: Δεύτερο κριτήριο είναι οι σωστή τοποθέτηση των ακμών. Οι ακμές που προκύπτουν από την αναζήτηση θα πρέπει να είναι όσο το δυνατόν πιο κοντά στις πραγματικές ακμές.

Αριθμός ανταποκρίσεων: Μια πραγματική ακμή δεν θα πρέπει να βρεθεί σε περισσότερες από μία ανιχνευμένες ακμές (Περιλαμβάνεται και στην πρώτη απαίτηση)

Αλγόριθμος Canny Edge detection

- Smoothing : Εξομάλυνση της εικόνας ώστε να μειωθεί ο θόρυβος
- Finding gradients: Οι ακμές γίνονται εντονότερες εκεί όπου η κλίση γίνεται μεγαλύτερη
- Non-maxima suppression: Μόνο τα μέγιστα μπορούν να σημειωθούν ως ακμές
- Double thresholding : Οι σημαντικές ακμές επιλέγονται με βάση ένα πάνω και ένα κάτω όριο
- Edge tracking by hysteresis: Οι τελικές ακμές καθορίζονται από την αφαίρεση όλων των ακμών που δε συνδέονται σε μια συγκεκριμένη ακμή (strong edge)

Περιγραφή του κάθε βήματος :

Smoothing

Ο θόρυβος που υπάρχει στις εικόνες στις οποίες η λήψη έγινε με τη χρήση κάμερας είναι αναπόφευκτος. Για να μην υπάρχουν λανθασμένες ακμές αυτός ο θόρυβος πρέπει να αφαιρεθεί. Για τον λόγο αυτό η εικόνα πρώτα εξομαλύνεται χρησιμοποιώντας ένα Gaussian φίλτρο. Ο πυρήνας ενός Gaussian φίλτρου με τυπική απόκλισης σ , είναι ένας πίνακας με σταυρωτά στοιχεία ώστε τα μεγαλύτερα βάρη να γειτονεύουν με το κεντρικό ρίχει όπως φαίνεται στο παράδειγμα:

$$B = \frac{1}{159} \cdot \begin{pmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{pmatrix}$$



Εικόνα 2: Αρχική εικόνα



Εικόνα 3: Gaussian blur

Finding Gradients

Ο αλγόριθμος Canny βρίσκει ακμές όπου η grayscale ένταση της εικόνας εμφανίζει τη μεγαλύτερη αλλαγή. Αυτές οι περιοχές βρίσκονται υπολογίζοντας τις παραγώγους της εικόνας. Η παράγωγος του κάθε ρικελ στην εικόνα που έχει εξομαλυνθεί θα προσδιορίζεται εφαρμόζοντας τον τελεστή Sobel [3]. Το πρώτο βήμα είναι η σύγκλιση της παραγώγου στην x και y διεύθυνση αντίστοιχα από την εφαρμογή των πυρήνων που φαίνεται στην παρακάτω εξίσωση:

$$K_{Gx} = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$$

$$K_{Gy} = \begin{pmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$$

Το μέτρο του ολικού gradient υπολογίζεται από τα G_x , G_y σαν το μέτρο της απόστασης τους όπως φαίνεται παρακάτω. Πολλές φορές απλοποιείται εφαρμόζοντας για μέτρο την απόσταση Manhattan ώστε να περιοριστεί η πολυπλοκότητα.

$$|G| = \sqrt{G_x^2 + G_y^2}$$

$$|G| = |G_x| + |G_y|$$

Όπου G_x , G_y τα gradients στην x και y διεύθυνση αντίστοιχα. Μία εικόνα των μεγεθών gradient συνήθως περιέχει τις ακμές που αναζητάμε. Ωστόσο, δεν μας δείχνουν ακριβώς που βρίσκονται γι' αυτό εκτός από το μέτρο πρέπει να προσδιοριστεί και η κατεύθυνση όπως φαίνεται στην εξίσωση:

$$\theta = \arctan \frac{|G_y|}{|G_x|}$$



Εικόνα 3: Gaussian blur



Εικόνα 4: Finding Gradients

Non maxima suppression

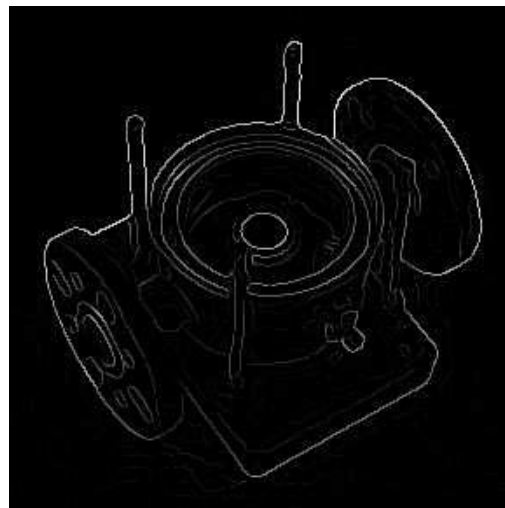
Ο σκοπός αυτού του βήματος, είναι η μετατροπή μιας «θολής» ακμής σε εικόνα gradient μεγεθών ώστε να οξυνθούν οι ακμές. Η λογική αυτού του βήματος είναι ότι διατηρούνται τα τοπικά μέγιστα στην gradient εικόνα και αφαιρούνται όλα τα υπόλοιπα με αποτέλεσμα να δίνεται περισσότερο βάρος στις πιο έντονες ακμές.

Ο αλγόριθμος για το κάθε pixel στην gradient εικόνα είναι :

1. Στρογγυλοποίηση της κατεύθυνσης θ του gradient κοντά στις 45° , ανάλογα με τη χρήση μιας γειτονιάς των 8 συνδεδεμένων pixel.
2. Σύγκριση της ισχύς ακμής ενός pixel με την ισχύ ακμής ενός άλλου pixel στην θετική και αρνητική κατεύθυνση gradient. Αν η κατεύθυνση του είναι κάθετη ($\theta = 90^\circ$), συγκρίνεται με τα pixels στην ίδια διεύθυνση.
3. Αν η ισχύς ακμής του τρέχοντος pixel είναι μεγαλύτερη σε σχέση με των άλλων pixels στην ίδια γειτονιά, τότε αυτή διατηρείται και μηδενίζονται οι γειτονικές.



Εικόνα 5: Finding Gradients



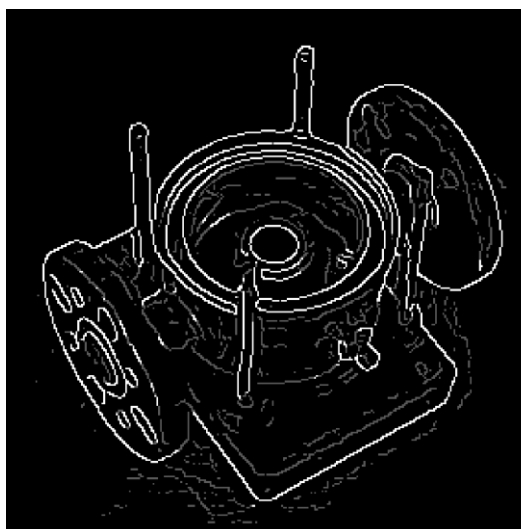
Εικόνα 6: Non max suppression

Double thresholding

Τα ακριανά pixels που απομένουν από το τρίτο βήμα του αλγορίθμου non-maxima suppression (Εικόνα 7) μπορεί να είναι αληθινές ακμές αλλά υπάρχει περίπτωση να πρόκειται και για θορύβους ή χρωματικές αλλαγές που προκαλούνται στην εικόνα π.χ. λόγω τραχιάς επιφάνειας. Ο πιο απλός τρόπος να διακρίνουμε αυτές τις παραμορφώσεις, είναι να χρησιμοποιήσουμε ένα κάτω όριο έτσι ώστε μόνο οι ισχυρότερες από αυτό το όριο ακμές να διατηρούνται. Ο αλγόριθμος Canny edge detection χρησιμοποιεί διπλά όρια, δηλαδή και κάτω και άνω όριο. Τα pixels που είναι πιο ισχυρά από το πάνω όριο μαρκάρονται ως ισχυρά pixels ενώ τα pixels που δεν ξεπερνούν το κάτω όριο παραλείπονται (Εικόνα 8). Τα pixels που βρίσκονται μεταξύ των ορίων μαρκάρονται ως αδύναμα pixels.



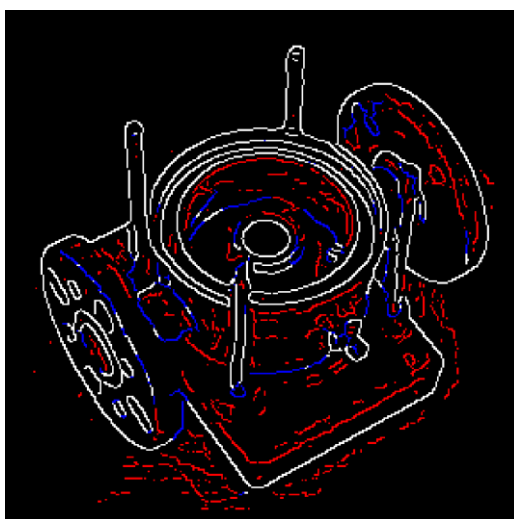
Εικόνα 7: Non max suppression



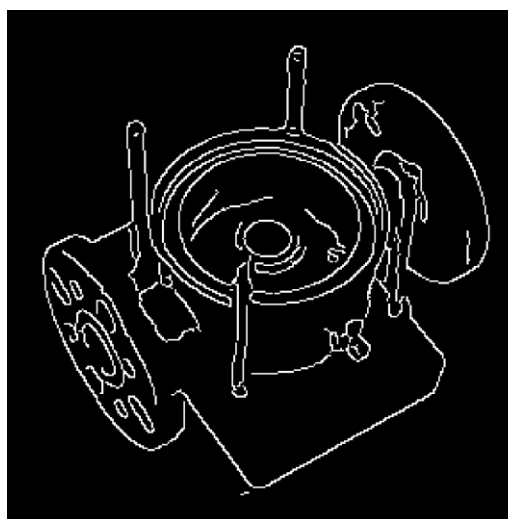
Εικόνα 8: Double Thresholding

Edge tracking by hysteresis

Στο βήμα αυτό οι ισχυρές ακμές συμπεριλαμβάνονται κατευθείαν στην τελική εικόνα ως ακμές, ενώ οι αδύναμες ακμές συμπεριλαμβάνονται μόνο εάν συνδέονται με τις ισχυρές ακμές. Η λογική είναι ότι ο θόρυβος και οι άλλες μικρότερες διακυμάνσεις είναι απίθανο να έχουν μια ισχυρή ακμή (με την κατάλληλη προσαρμογή των ορίων). Έτσι, οι ισχυρές ακμές (Εικόνα 9) θα οφείλονται σχεδόν μόνο σε πραγματικές ακμές στην αρχική εικόνα. Οι αδύναμες ακμές μπορεί να οφείλονται είτε στην ύπαρξη πραγματικών ακμών είτε λόγω θορύβου ή χρωματικών διακυμάνσεων. Με την ανάλυσή τους, απορρίπτονται οι λανθασμένες ακμές και κρατούνται οι ισχυρές και οι αδύναμες που συνδέονται με τις ισχυρές (Εικόνα 10).



Εικόνα 9: Edge tracking by hysteresis



Εικόνα 10: Final Image

2.3 Εντοπισμός γωνιών με τον Αλγόριθμο Harris

Ο αλγόριθμος εύρεσης γωνιών των Harris και Stephens αποτελεί βελτιωμένη εκδοχή ενός παλαιότερου αλγόριθμου του Morave [5] και προτάθηκε από τους Harris και Stephens το 1988 [6]. Είναι ένας από τους πιο διαδεδομένους αλγορίθμους εύρεσης σημείων, κυρίως επειδή δεν επηρεάζεται από την περιστροφή, κλιμάκωση, φωτισμό και θόρυβο της εικόνας. Βασίζεται στην τοπική συνάρτηση αυτοσυσχέτισης ενός σήματος, η οποία μετρά τις τοπικές αλλαγές του σήματος με παράθυρα που μετακινούνται προς κάθε κατεύθυνση κατά ένα πολύ μικρό ποσό.

Η βασική λογική είναι ότι για κάθε σημείο ελέγχεται η συνάρτηση φωτεινότητας I για μικρές μετακινήσεις γύρω από αυτό και ως γωνίες χαρακτηρίζονται τα σημεία εκείνα στα οποία οι μεταβολές της τιμής της συνάρτησης φωτεινότητας είναι μεγάλες δίνεται από την παρακάτω σχέση :

$$E(u, v) = \sum_{x,y} w(x, y) [I(x + u, y + v) - I(x, y)]^2 \quad (11)$$

όπου $I(x, y)$ αντιπροσωπεύει τη συνάρτηση εικόνας, (x, y) τα σημεία στο παράθυρο Ω με κέντρο το (x, y) , και $w(x, y)$ μια Gaussian kernel συνάρτηση τοποθετημένη στο (x, y) .

Το παράθυρο μπορεί να είναι βηματικό (1 μόνο εντός του παραθύρου και 0 κενός) είτε Gaussian (δίνοντας μεγαλύτερο βάρος στο κέντρο). Για μικρές αποστάσεις (u, v) η μετατοπισμένη εικόνα μπορεί να προσεγγιστεί με ανάπτυγμα Taylor ως εξής:

$$I(x + u, y + v) \approx I(x, y) + uI_x(x, y) + vI_y(x, y) \quad (12)$$

Θεωρώντας τον 2×2 πίνακα αυτοσυσχέτισης A (Harris και Stephens Matrix)

$$A = \sum_{x,y} w(x, y) \begin{pmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{pmatrix} \quad (13)$$

Η διαφορά στην φωτεινότητα γράφεται:

$$E(u, v) \approx (u \ v) A \begin{pmatrix} u \\ v \end{pmatrix} \quad (14)$$

Στην πράξη και η αρχική εικόνα εξομαλύνεται με ένα Gaussian πυρήνα G_σ με τυπική απόκλιση σ (κλίμακα διαφόρισης), πέρα από την χρήση του πυρήνα G_ρ με τυπική απόκλιση ρ (κλίμακα ολοκλήρωσης) ως το τυπικό παράθυρο $w(x, y)$. Συνεπώς, ο πίνακας των Harris και Stephens γράφεται:

$$A(x, y, \sigma) = G_\rho * \begin{bmatrix} \left(\frac{\partial I_\sigma}{\partial x} \cdot \frac{\partial I_\sigma}{\partial x}\right)(x, y) & \left(\frac{\partial I_\sigma}{\partial x} \cdot \frac{\partial I_\sigma}{\partial y}\right)(x, y) \\ \left(\frac{\partial I_\sigma}{\partial x} \cdot \frac{\partial I_\sigma}{\partial y}\right)(x, y) & \left(\frac{\partial I_\sigma}{\partial y} \cdot \frac{\partial I_\sigma}{\partial y}\right)(x, y) \end{bmatrix}, \quad I_\sigma = G_\sigma * I \quad (15)$$

Οι Harris και Stephens πρότειναν, για λόγους απλότητας, το εξής κριτήριο γωνιότητας:

$$R = \det(A) - k \cdot \text{trace}^2(A)$$

Όπου k μία προκαθορισμένη παράμετρος ευαισθησίας. Επειδή $\det(A) = \lambda_1 \lambda_2$ και $\text{trace}(A) = \lambda_1 + \lambda_2$, το κριτήριο R παίρνει μεγάλες τιμές σε σημεία που αντιστοιχούν σε γωνίες. Τα τελικά σημεία ενδιαφέροντος (γωνίες) προκύπτουν από την επιλογή μόνο των τοπικών μεγίστων του κριτηρίου γωνιότητας και απόρριψη σημείων κάτω από κάποιο προκαθορισμένο όριο (non-maxima suppression).

Ο αλγόριθμος συνοψίζεται στα παρακάτω βήματα:

Πίνακας 1: Αλγόριθμος Harris

Step 1: Compute x and y derivatives of image

$$I_x = G_\sigma^x * I \quad I_y = G_\sigma^y * I$$

Step 2: Compute products of derivatives at every pixel

$$I_{xx} = I_x \cdot I_x \quad I_{yy} = I_y \cdot I_y \quad I_{xy} = I_x \cdot I_y$$

Step 3: Compute the sums of the products of derivatives at each pixel

$$S_{x2} = G_{\sigma'} * I_{xx} \quad S_{y2} = G_{\sigma'} * I_{yy} \quad S_{xy} = G_{\sigma'} * I_{xy}$$

Step 4: Define at each pixel (x, y) the matrix

$$H(x, y) = \begin{bmatrix} S_{x2}(x, y) & S_{xy}(x, y) \\ S_{xy}(x, y) & S_{y2}(x, y) \end{bmatrix}$$

Step 5: Compute the response of the detector at each pixel

$$R = \det(H) - k(\text{Trace}(H))^2$$

Step 6: Threshold on value of R. Compute non max suppression.

Αποδεικνύεται ότι οι ιδιοτιμές του παραπάνω πίνακα M είναι ανάλογες με τις κύριες καμπυλότητες της επιφάνειας της εικόνας I . Η τελική εκτίμηση γίνεται λοιπόν μέσω των ιδιοτιμών του πίνακα M . Έστω λ_1, λ_2 είναι οι ιδιοτιμές του πίνακα M , διακρίνονται 3 περιπτώσεις:

1. Αν λ_1 και λ_2 είναι μικρές, τότε η συνάρτηση φωτεινότητας της εικόνας έχει σχεδόν σταθερή ένταση (οι μετακινήσεις γύρω από το σημείο που ελέγχεται έχουν σχεδόν μηδαμινή επίδραση στην συνάρτηση I).
2. Αν μία από τις ιδιοτιμές λ_1 ή λ_2 είναι μεγάλη και η άλλη μικρή, τότε η τοπική αυτοσυσχέτιση παρουσιάζει κορυφή με αποτέλεσμα μια μικρή μετακίνηση κατά μήκος της κορυφής να προκαλεί μικρή αλλαγή στην συνάρτηση I στη μία διεύθυνση και σημαντική στην άλλη. Αυτό υποδεικνύει την ύπαρξη ακμής.
3. Αν και οι δύο ιδιοτιμές είναι μεγάλες, τότε ακόμα και μικρή μετακίνηση σε οποιαδήποτε διεύθυνση έχει σαν αποτέλεσμα μεγάλη αλλαγή στην συνάρτηση E . Αυτό υποδεικνύει την ύπαρξη γωνίας.

Για να αποφευχθεί ο αναλυτικός υπολογισμός των ιδιοτιμών λ_1, λ_2 , επινοήθηκε ένα μέτρο που εκφράζει την απόκριση των παραπάνω περιπτώσεων. Αν $\det(A)$ και $\text{trace}^2(A)$

είναι η ορίζουσα (determinant) και ίχνος (trace) του πίνακα $A(x, y)$, τότε η απόκριση της γωνιότητας (cornerness response) ορίζεται ως εξής:

$$R(x, y) = \det[A(x, y)] - k \cdot \text{tr}^2[A(x, y)],$$

Όπου $\det(A) = \lambda_1 \lambda_2$ και $\text{trace}(A) = \lambda_1 + \lambda_2$, ενώ η παράμετρος k παίρνει παραδοσιακά τις τιμές στο διάστημα $[0.04, 0.06]$

Η απόκριση R παίρνει μεγάλες θετικές τιμές στα σημεία που υπάρχουν γωνίες, αρνητικές τιμές στα σημεία που υπάρχουν ακμές και μικρές τιμές σε ομαλές περιοχές.

Η παραπάνω προσέγγιση παρουσιάζει τα εξής προβλήματα:

- Επειδή ο πίνακας συμμεταβολής περιέχει σαν στοιχεία τα τετράγωνα του βαθμίδας της έντασης, η απόκριση R εξαρτάται από τις βαθμίδες της έντασης στην τέταρτη δύναμη, πράγμα που συνεπάγεται υψηλή εξάρτηση της R από τις αποκλίσεις της αντίθεσης (contrast) της εικόνας.
- Η εξάρτηση της R από την αντίθεση της εικόνας κάνει δύσκολο τον προσδιορισμό του κατωφλιού, πάνω από το οποίο η απόκριση R θεωρείται ότι παίρνει μεγάλες τιμές καθώς επίσης και της παραμέτρου k .

Ένα άλλο μέτρο της απόκρισης R που εκφράζει την γωνιότητα και αποφεύγει την παράμετρο k είναι:

$$R(x, y) = \frac{\det[A(x, y)]}{\text{trace}[A(x, y)]}$$

Αν μια από τις ιδιοτιμές τείνει στο μηδέν τότε η $\det(A)$ και επομένως η R τείνει στο μηδέν. Αν οι ιδιοτιμές είναι παρόμοιες αλλά μικρές, τότε η R και πάλι τείνει στο μηδέν. Όμως, αν οι ιδιοτιμές είναι παρόμοιες αλλά μεγάλες τότε η $\det(A)$ είναι πολύ μεγαλύτερη του $\text{trace}(A)$ και επομένως η R είναι μεγάλη.

Τέλος, ένα υποψήφιο σημείο που επιλέγεται σαν σημείο που αντιπροσωπεύει το σημείο ενδιαφέροντος αν η απόκριση του R , είναι ένα τοπικό μέγιστο 8-γειτόνων με τη μέθοδο απαλοιφής των μη μεγίστων (non maxima suppression)

2.3 Εντοπισμός ευθειών

Όταν μια εικόνα θα υποστεί επεξεργασία και θα περάσει από διαφορετικές περιοχές της ανάλυσης εικόνων όπως αναγνώριση αντικειμένων, είναι σημαντική η μείωση των δεδομένων διατηρώντας τα σημαντικά χαρακτηριστικά της και τη δομική της πληροφορία. Τεχνικές ανίχνευσης ακμών καθιστούν δυνατή τη μείωση ενός ποσοστού δεδομένων. Εάν γραμμές, ελλείψεις και άλλες γεωμετρικές πληροφορίες μπορούσαν να προσδιοριστούν μέσω συγκεκριμένων μαθηματικών σχέσεων η ποσότητα δεδομένων θα περιοριζόταν ακόμα περισσότερο. Αυτός ήταν και ο λόγος του σχεδιασμού του μετασχηματισμού Hough [8] [9] από τους Richard Duda και Peter Hart το 1972 [7], οποίος αρχικά σχεδιάστηκε για την αναγνώριση των περισσότερων πιθανών γραμμών σε μια εικόνα. Ωστόσο, ο συνδυασμός γραμμών μπορεί να προσεγγίσει πολλά άλλα σχήματα και έτσι η χρήση του μετασχηματισμού Hough έχει γενικευτεί στην ανίχνευση αντικειμένων καθιστώντας τον μια από τις πιο διαδεδομένες τεχνικές εντοπισμού ορίων σε περιοχές.

2.3.1 Hough Transform

Ευθείες στον Hough Χώρο

Κάθε ευθεία σε μια 2D εικόνα μπορεί να περιγραφεί μοναδικά από δυο παραμέτρους α και β όπως παρουσιάζεται στην εξίσωση 16.

$$y = ax + b \quad (16)$$

όπου (x, y) είναι ένα σημείο του χώρου της εικόνας.

Κάθε ευθεία στον δυσδιάστατο χώρο της εικόνας μπορεί να περιγραφεί μέσω ενός σημείου (α, b) στον χώρο του Hough όπου

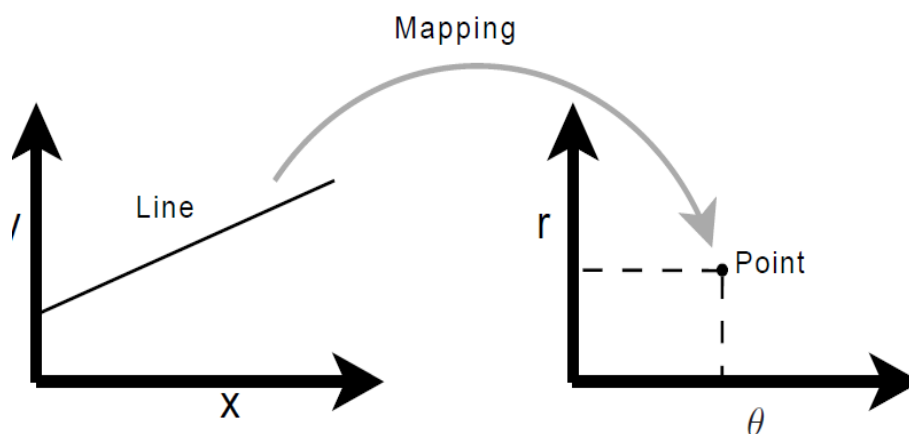
$$b = -\alpha x + y \quad (17)$$

Αυτή η μορφή δεν εξυπηρετεί την αναπαράσταση κάθετων ευθειών. Γι' αυτό το λόγο αντί να γίνεται η χρήση των παραπάνω ευθειών χρησιμοποιούνται οι πολικές συντεταγμένες. Χρησιμοποιούνται δυο παράμετροι, θ και r , για να περιγραφούν οι ευθείες στο χώρο, όπου θ είναι κλίση της ευθείας και r η απόσταση από τον άξονα αντίστοιχα. Κάθε σημείο στον χώρο (x, y) αντιστοιχίζεται σε σημεία της καμπύλης $x_i \cos\theta + y_i \sin\theta = r$ στον Hough χώρο (Εικόνα 13).

$$r = x \cos\theta + y \sin\theta \quad (18)$$

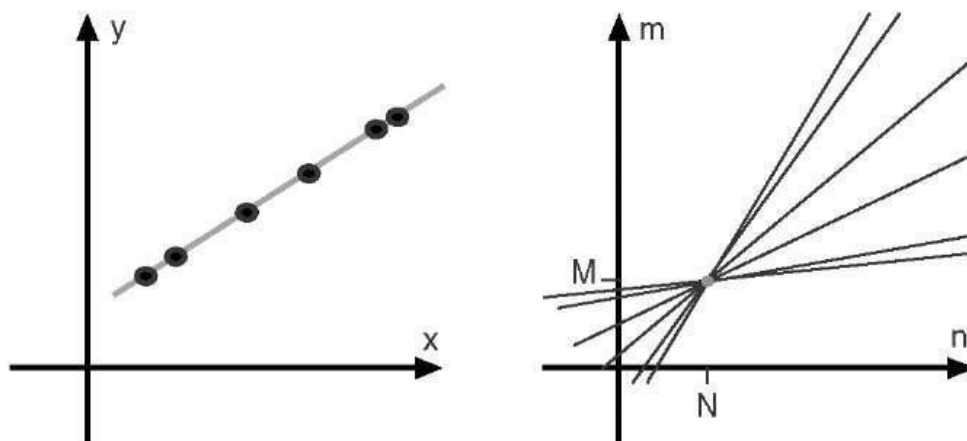
$$y = -\frac{\cos\theta}{\sin\theta}x + \frac{r}{\sin\theta} \quad (19)$$

Όλες οι γραμμές μπορούν να περιγραφούν με αυτή τη μορφή για $\theta \in [0, 180]$ και $r \in \mathbb{R}$. Επομένως, στον Hough χώρο κάθε γραμμή περιγράφεται από μόνο ένα σημείο συναρτήσει ενός μοναδικού σετ παραμέτρων (θ, r) . Η αντιστοιχία του κάθε σημείου φαίνεται στην Εικόνα 11.



Εικόνα 11: Αντιστοίχιση μιας μοναδικής γραμμής σε ένα σημείο στον Hough χώρο

Η κύρια ιδέα του μετασχηματισμού Hough [7], είναι η αντιστοίχιση ενός σημείου σε πολλές ευθείες. Από ένα σημείο περνάνε άπειρες ευθείες και κάθε ευθεία αναπαρίσταται στον Hough χώρο ως ένα σημείο (θ_0, r_0) . Επομένως, η μετατροπή της κάθε ευθείας σε σημείο του χώρου Hough δημιουργεί μια νέα ευθεία που θα περιγράψει το αρχικό σημείο. Δηλαδή, ένα σημείο στον Ευκλείδειο χώρο αποτελεί μια ευθεία στον Hough χώρο όπως εμφανίζεται στην παρακάτω εικόνα (Εικόνα 12)



Εικόνα 12: Αντιστοίχιση κάθε σημείου σε μια ευθεία και την ευθεία που διέρχεται ως το σημείο τομής στον Hough χώρο

Οι τεχνικές ανίχνευσης ακμών διακρίνονται στις τοπικές οι οποίες χρησιμοποιούν πληροφορία σε μια γειτονιά της εικόνας και τις καθολικές που χρησιμοποιούν όλη την πληροφορία της εικόνας.

Αλγόριθμος

Ο αλγόριθμος που ανιχνεύει ευθείες γραμμές μπορεί να χωριστεί στα παρακάτω βήματα:

Βήμα 1 : Εύρεση των ακμών χρησιμοποιώντας ανιχνευτή ακμών π.χ. Canny [2]

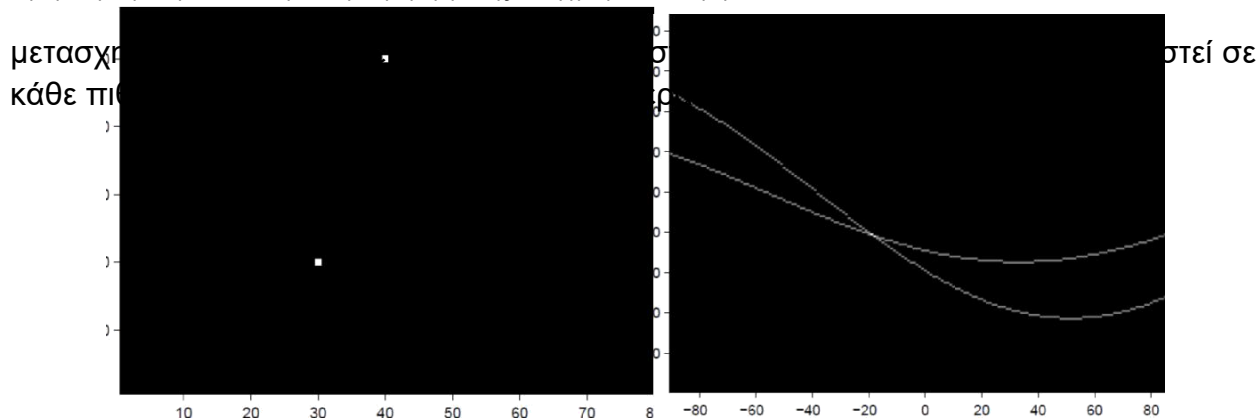
Βήμα 2 : Χαρτογράφηση των ακμών στο χώρο Hough και αποθήκευση σε πίνακες συσσώρευσης.

Βήμα 3 : Ερμηνεία του συσσωρευτή να αποδώσει γραμμές απείρου μήκους. Η ερμηνεία γίνεται από κατωφλίωση και ενδεχομένως και άλλους περιορισμούς

Βήμα 4 : Μετατροπή των γραμμών που εκτείνονται μέχρι το άπειρο σε πεπερασμένες γραμμές.

Μετασχηματισμός στο χώρο του Hough

Ο μετασχηματισμός Hough παίρνει ως είσοδο ένα δυαδικό χάρτη ακμών και επιχειρεί να εντοπίσει ακμές που έχουν τοποθετηθεί σαν ευθείες γραμμές. Η βασική ιδέα του



Εικόνα 13: Η ευθεία που διέρχεται από δύο σημεία στον Ευκλείδειο χώρο είναι το σημείο που τέμνονται οι ευθείες των σημείων στο χώρο Hough

Ένας τυπικός χάρτης ακμών περιλαμβάνει πολλά σημεία όμως η αρχή για την ανίχνευση γραμμών είναι η εξής: Κάθε ακμή μετατρέπεται σε γραμμή στο χώρο του Hough και περιοχές όπου οι περισσότερες γραμμές στο χώρο Hough τέμνονται ονομάζονται αληθινές γραμμές στον χάρτη ακμών.

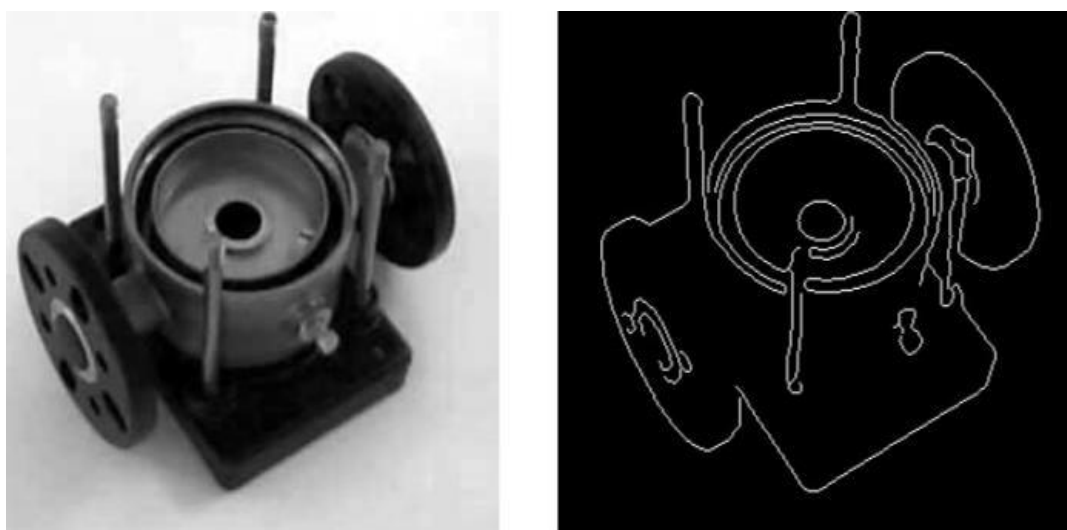
Πίνακας συσσώρευσης Hough

Για να ορίσουμε τις περιοχές όπου οι περισσότερες γραμμές στο χώρο Hough τέμνονται χρησιμοποιούμε έναν πίνακα συσσώρευσης (accumulator), ο οποίος καλύπτει όλο τον Hough χώρο. Όταν μια ακμή μετασχηματίζεται, οι θέσεις του πίνακα αυξάνονται για όλες τις γραμμές που μπορούν να περάσουν από αυτό το σημείο. Η ανάλυση του συσσωρευτή ορίζει την ακρίβεια με την οποία μπορούν να ανιχνευθούν οι γραμμές. Οι μεταβλητές r και θ χρησιμοποιούνται για να ορίσουν ένα ρικελ και μία μοίρα αντίστοιχα. Σε γενικές γραμμές, ο αριθμός διαστάσεων του πίνακα συσσώρευσης αντιστοιχεί στον αριθμό των αγνώστων παραμέτρων στο πρόβλημα του μετασχηματισμού Hough. Έτσι για μια έλλειψη, απαιτείται ένας πενταδιάστατος χώρος (οι συντεταγμένες του κέντρου, το μήκος των κύριων και δευτερευόντων αξόνων και η γωνία τους). Για γραμμές δύο διαστάσεων αρκούν οι μεταβλητές r και θ . Γι' αυτό το λόγο, είναι δυνατή η απεικόνιση του περιεχομένου του συσσωρευτή. Κάθε κελί του πίνακα συσσώρευσης είναι ένα ζεύγος (r, θ) που περιγράφει μια ευθεία η οποία μπορεί να τέμνει πολλά σημεία της εικόνας. Για να αποφύγει η ο αλγόριθμος Hough τα ανεπιθύμητα αποτελέσματα, χρησιμοποιεί ένα σύστημα ψήφων. Έτσι, κάθε φορά που μια γραμμή περνάει από ένα κελί, η τιμή του κελιού αυτού αυξάνεται κατά μια μονάδα. Έπειτα από όλο τον υπολογισμό για κάθε κελί του πίνακα συσσώρευσης, καταλήγουμε με μια συσσωρευμένη εικόνα όπου σημεία δίνουν ψήφους σε σημεία.

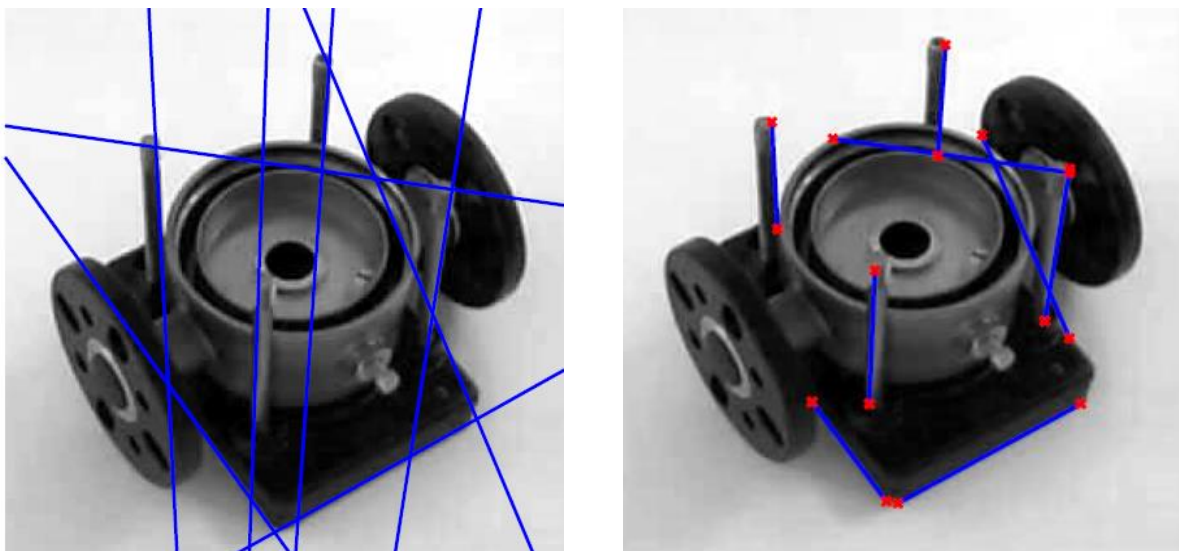
Ανίχνευση γραμμών στο άπειρο:

Ένα παράδειγμα ανίχνευσης ευθειών και μετατροπή τους σε πεπερασμένες ακμές φαίνεται στην Εικόνα 15.

Ο πιο βασικός τρόπος για να ανιχνευθούν είναι να τεθεί ένα κάτω όριο και να θεωρηθούν όλες οι γραμμές του πίνακα συσσώρευσης ως ακμές αν βρίσκονται πάνω από το όριο που έχει οριστεί. Το όριο θα μπορούσε να είναι το 30% των μεγαλύτερων τιμών του συλλέκτη (accumulator). Αυτή η προσέγγιση κάποιες στιγμές αρκεί, όμως σε πολλές περιπτώσεις πρέπει να εφαρμόζονται επιπλέον περιορισμοί. Για αρκετές εισόδους του πίνακα συσσώρευσης τα κελιά γύρω από μια πραγματική γραμμή θα παίρνουν μεγάλες τιμές. Κατά συνέπεια, ένα απλό άνω όριο έχει την τάση να ανιχνεύει διάφορες γραμμές (σχεδόν ίδιες) για κάθε πραγματική γραμμή. Για να αποφευχθεί αυτό παίρνονται ως τιμές τα τοπικά μέγιστα μέσα σε μια γειτονιά (suppression neighborhood) έτσι ώστε να αποφευχθεί να εντοπιστούν γειτονικές γραμμές και οι δύο αυτές γραμμές να έχουν σημαντικές διαφορές πριν εντοπιστούν.



Εικόνα 14: Δεξιά η αρχική εικόνα σε grayscale, αριστερά η εικόνα μετά από εφαρμογή του αλγρίθμου Canny



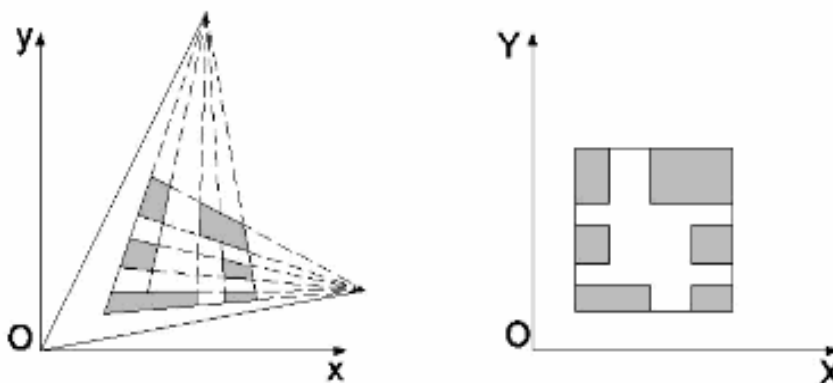
Εικόνα 15: Δεξιά η ανίχνευση ευθειών και αριστερά η μετατροπή τους σε πεπερασμένες

Ο τρόπος που γίνεται η κβάντιση του πίνακα συσσώρευσης για τα r και τα θ καθώς και το σύστημα ψήφου που θεωρούμε, μπορεί να επηρεάσει σημαντικά τα αποτελέσματα της τελικής εικόνας. Αν για κάθε $pixel$ που περνάει η ευθεία αυξάνουμε κάθε κελί του accumulator κατά μια μονάδα, δίνουμε αυτομάτως την ίδια βαρύτητα σε όλες τις ευθείες αρκεί να περνάνε ακόμα και από ένα πολύ μικρό κομμάτι του $pixel$ αυτού. Όσο η κβάντιση μεγαλώνει και έχουμε περισσότερα $pixel$ ανά κελί τα αποτελέσματα χάνουν την εγκυρότητά τους. Λύση σε αυτό το πρόβλημα δίνει το σύστημα ψηφοφορίας με βάρη σύμφωνα με το οποίο διαφοροποιούνται οι ψήφοι που θα δοθούν σε κάθε ευθεία ανάλογα με το που τέμνει τα $pixel$ της εικόνας.

2.4 Homography

Ο γενικότερος μετασχηματισμός που αφορά τον χώρο των δύο διαστάσεων και είναι ιδιαίτερα σημαντικός για την φωτογραμμετρία είναι ο προβολικός μετασχηματισμός ή αλλιώς ομογραφία (homography). Η φωτογραφική μηχανή που χρησιμοποιούμε για την απεικόνιση της δομής του χώρου εκτελεί ουσιαστικά έναν προβολικό μετασχηματισμό. Δεν αφορά μετασχηματισμούς μεταξύ απεικονίσεων (αναλογική και ψηφιακή εικόνα, χάρτης) που μετρούνται σε διαφορετικά συστήματα αναφοράς. Η βασική χρήση του είναι στην φωτογραμμετρική απόδοση (αποτύπωση) του χώρου από μετρήσεις σε εικόνες. Εξαιτίας του δισδιάστατου χαρακτήρα του είναι δυνατή μόνον η απόδοση αντικειμένων (απολύτως ή κατά ικανή προσέγγιση) επιπέδων. Οι ιδιότητες προβολικού μετασχηματισμού:

- Δεν διατηρεί το σχήμα, το μήκος των ευθύγραμμων τμημάτων, γωνίες, αποστάσεις και λόγους μηκών. Για παράδειγμα, ένας κύκλος ενδέχεται να μετασχηματιστεί σε έλλειψη και ένα ορθογώνιο σε ρόμβο με χρήση προβολικού μετασχηματισμού
- Διατηρείται η ευθύτητα. Αυτό σημαίνει ότι συνευθειακά σημεία θα μετασχηματιστούν σε συνευθειακά σημεία. Αυτή η ιδιότητα, η διατήρηση δηλαδή των ευθειών είναι ουσιαστικά αυτή που ορίζει τους προβολικούς μετασχηματισμούς.



Εικόνα 16: Ο μετασχηματισμός της επιφάνειας με τη χρήση Homography

Προβολικότητα ή προβολικός μετασχηματισμός είναι ένας αντιστρέψιμος μετασχηματισμός h από το P^2 στον εαυτό του έτσι ώστε τρία σημεία x_1, x_2, x_3 ανήκουν στην ίδια ευθεία αν και μόνο αν τα $h(x_1), h(x_2)$ και $h(x_3)$ είναι συνευθειακά. Η αντιστροφή ενός προβολικού μετασχηματισμού καθώς και ο συνδυασμός δύο προβολικών μετασχηματισμών αποτελούν νέους προβολικούς μετασχηματισμούς. Μια απεικόνιση h από τον P^2 στον εαυτό του ($P^2 \rightarrow P^2$) συνιστά προβολικό μετασχηματισμό εάν υφίσταται ένας πίνακας H 3×3 τέτοιος ώστε για κάθε σημείο x του P^2 να ισχύει η ισότητα

$$h(x) = Hx$$

Πράγματι έστω ότι έχουμε τρία συνευθειακά σημεία x_1, x_2, x_3 τα οποία ανήκουν στην ευθεία l .

Τότε $ITx_i = 0$ για $i = 1, \dots, 3$. Έστω ότι H είναι ένας μη αντιστρέψιμος 3×3 πίνακας. Ισχύει ότι

$$ITH - 1Hx_i = 0 \quad (21)$$

Επομένως τα σημεία Hx_i ανήκουν στην ευθεία $H^{-T}l$ άρα είναι συνευθειακά. Τελικά διατηρείται η ιδιότητα της συγγραμικότητας και άρα πρόκειται για προβολικό μετασχηματισμό.

Ο επίπεδος προβολικός μετασχηματισμός είναι γραμμικός μετασχηματισμός που εφαρμόζεται σε ομογενή διανύσματα τριών στοιχείων και εκφράζεται από έναν πίνακα 3×3 σύμφωνα με τη σχέση.

$$\begin{pmatrix} x_1' \\ x_2' \\ x_3' \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} \leftrightarrow x' = Hx \quad (22)$$

Εάν στην παραπάνω εξίσωση ο πίνακας H πολλαπλασιαστεί με μη μηδενική σταθερά, ο προβολικός μετασχηματισμός παραμένει ανεπηρέαστος. Ο H αποτελεί έναν ομογενή πίνακα εφόσον σημασία έχει μόνο ο λόγος των στοιχείων του πίνακα. Στον προβολικό μετασχηματισμό υπάρχουν οκτώ βαθμοί ελευθερίας μεταξύ των εννέα στοιχείων και μπορούν να σχηματιστούν οκτώ διαφορετικοί λόγοι. Η συνηθέστερη εφαρμογή του προβολικού μετασχηματισμού είναι η διόρθωση της παραμόρφωσης της οπτικής που εμφανίζουν οι εικόνες. Είναι γνωστό πως η κεντρική προβολή ενός επίπεδου αντικειμένου στην εικόνα συνδέεται με το επίπεδο του χώρου μέσω ενός προβολικού μετασχηματισμού. Αποτέλεσμα της εφαρμογής του μετασχηματισμού αυτού, είναι η αντιστοίχιση μιας νέας εικόνας στην οποία τα αντικείμενα του επιπέδου εμφανίζονται με το σωστό τους γεωμετρικό σχήμα και υπό σωστή κλίμακα. Ο υπολογισμός του προβολικού μετασχηματισμού στηρίζεται στην αντιστοίχιση τεσσάρων τουλάχιστον σημείων της εικόνας με τέσσερα σημεία του επιπέδου. Κάθε σημείο p' του νέου επιπέδου μπορεί να περιγραφεί συναρτήσει του σημείου $p(x, y)$ με εξισώσεις της μορφής:

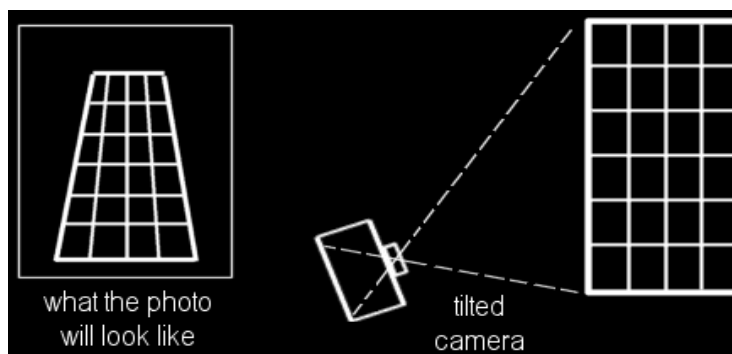
$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \quad (23)$$

$$y' = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}} \quad (24)$$

3. ΤΕΧΝΙΚΕΣ ΔΙΟΡΘΩΣΗΣ ΠΡΟΟΠΤΙΚΗΣ ΚΑΙ ΑΦΑΙΡΕΣΗΣ ΠΕΡΙΘΩΡΙΟΥ

3.1 Υπάρχουσες Τεχνικές

Η διόρθωση της προοπτικής είναι από μόνης της ένα μεγάλο και σημαντικό κομμάτι της επεξεργασίας εικόνας. Εξετάζει πληθώρα παραμέτρων και προβλημάτων και συνεχώς εξελίσσονται οι προτάσεις και οι μεθοδολογίες που στοχεύουν στην απλούστευση και την επίλυση αυτών των προβλημάτων. Παρόλο που υπάρχουν πολλές προσεγγίσεις, στην πραγματικότητα είναι πολύ περιορισμένες καθώς όλες έχουν περιορισμούς στα μοντέλα που εξετάζουν. Ελάχιστες προτάσεις είναι γενικές, καλύπτοντας όλες τις περιπτώσεις, και συνήθως αποτελούν βάση για την διαμόρφωση νέων εξειδικευμένων μοντέλων. Οι διαφορετικές παραδοχές καθώς και το πρόβλημα που προσεγγίζεται και μοντελοποιείται στα πλαίσια μιας μεθοδολογίας οδηγεί και σε μια πρώτη κατηγοριοποίησή της. Το πρόβλημα εδώ αποτελεί η προοπτική παραμόρφωση που προκύπτει από τη φωτογράφιση με ψηφιακή κάμερα. Ο παρατηρητής που φωτογραφίζει το αντικείμενο δεν βρίσκεται στο ίδιο επίπεδο με αυτό με αποτέλεσμα να μην απεικονίζεται η μπροστινή όψη με τα χαρακτηριστικά που την αντιλαμβάνεται ο παρατηρητής, αλλοιώνοντας έτσι τις διαστάσεις του αντικειμένου που εικονίζεται είτε πρόκειται για πρόσωπο είτε για έγγραφο είτε για μια σκηνή. Οι διαστάσεις που βρίσκονται πιο κοντά στη φωτογραφική κάμερα απεικονίζονται μεγαλύτερες από τις πιο απομακρυσμένες. Έτσι, κάτι που στην αντίληψη του παρατηρητή θα σχεδιαζόταν σαν ένα παραλληλόγραμμο μπορεί στην εικόνα να έχει διαστάσεις τραapeζίου ή τριγώνου. Ένα χαρακτηριστικό παράδειγμα φαίνεται στην παρακάτω εικόνα.



Εικόνα 17: Αλλοίωση των διαστάσεων του αντικειμένου μέσω της κάμερας

Στόχος λοιπόν είναι η διόρθωση αυτής της οπτικής παραμόρφωσης και η αποκατάσταση των αρχικών διαστάσεων του αντικειμένου. Εάν η εικόνα που εξετάζουμε περιέχει μια σκηνή ή κάποιο αντικείμενο, ένας τρόπος είναι η προσέγγιση των γεωμετρικών χαρακτηριστικών του είτε αναζητώντας γεωμετρικά σχήματα είτε εφαρμόζοντας ένα πλαίσιο για την εύρεση του περιγράμματος, είτε διαχωρίζοντας το υπόβαθρο από την

εικόνα. Πολλοί μέθοδοι και τεχνικές ασχολούνται με τη γεωμετρία της εικόνας εξετάζοντας τα όρια του εικονιζόμενου αντικειμένου. Εάν η εικόνα περιέχει κείμενο, θα μπορούσαν να εφαρμοστούν τεχνικές ανάλυσης κειμένου οι οποίες θα έφερναν ποιοτικότερα αποτελέσματα και λιγότερες απώλειες πληροφορίας. Μια πρώτη κατηγοριοποίηση εδώ θα μπορούσε να έχει σχέση με το είδος των φωτογραφιών αν περιέχουν ή όχι κείμενο.

Παρακάτω παρουσιάζονται κάποιες γενικές μέθοδοι για την διόρθωση της προοπτικής σε έγγραφα από κάμερα.

3.2 Μεθοδολογίες που χρησιμοποιούν την εύρεση των ορίων του εγγράφου

Ένα σημαντικό κομμάτι στις εικόνες που αναπαριστούν έγγραφα είναι το όριο διακριτότητας, δηλαδή ένα ελάχιστο όριο κατά το οποίο μπορούμε να διακρίνουμε και να διαβάσουμε τα κείμενα του εγγράφου. Για να μπορεί να γίνει αυτό είναι πολύ χρήσιμο να μπορούν να προσδιοριστούν τα όρια του εγγράφου μέσα σε ένα σαφές ορθογώνιο σύνορο. Προχωρώντας σε αυτό το βήμα μπορούμε στη συνέχεια να διορθώσουμε την εικόνα χρησιμοποιώντας διάφορες τεχνικές που βασίζονται στον προσδιορισμό των ορίων του εγγράφου [18]-[20]. Η μεθοδολογία που ακολουθήσαμε στα πλαίσια της πτυχιακής ανήκει σε αυτή την κατηγορία και περιγράφεται αναλυτικότερα παρακάτω. Μερικές υποκατηγορίες που μπορούμε να διακρίνουμε σύμφωνα με [21] είναι :

a) Εύρεση του aspect ratio του εγγράφου

Στις περισσότερες φωτογραφίες εγγράφων η εύρεση ενός τετράπλευρου ορίου που αντιστοιχεί στα όρια του εγγράφου είναι πολύ εύκολη. Με χρήση του προβολικού μετασχηματισμού το τετράπλευρο μετατρέπεται σε παραλληλόγραμμο. Οι κορυφές του αρχικού τετράπλευρου χρησιμοποιούνται σαν είσοδο [10] (βλέπε 3.2.1 Εικόνα 18)για να υπολογιστεί και να εφαρμοστεί η homography ανάμεσα στην όψη από την οποία τραβήχτηκε η φωτογραφία και την πρόσοψη η οποία είναι ιδανική για επεξεργασία εγγράφων. Στην περίπτωση που οι αρχικές αναλογίες του ορθογωνίου είναι γνωστές τότε οι κορυφές του τετράπλευρου της εικόνας μπορούν να αντιστοιχηθούν με τις γωνίες του ορθογωνίου. Έτσι θα μπορούσε να επιτευχθεί η διόρθωση.

b) Εύρεση των παραλλήλων και κάθετων γραμμών.

Ένας άλλος τρόπος είναι η εύρεση 2 ζευγών παραλλήλων και δύο ζευγών κάθετων γραμμών στο έγγραφο [18] [11] [14]. Αυτή η τακτική (βλέπε 3.2.2) είναι χρήσιμη αν η ακριβής αναλογία δεν είναι διαθέσιμη. Επιπλέον η ανίχνευση γραμμών είναι πιο αξιόπιστη από την ανίχνευση σημείων . Η γραμμή που διέρχεται από τα δύο σημεία της τομής των ζευγών γραμμών είναι ένας προβολικός μετασχηματισμός της γραμμής αυτής στο άπειρο. Ο μετασχηματισμός που αντιστοιχεί στην παρατηρούμενη γραμμή στο άπειρο εφαρμόζεται για την αφαίρεση του προβολικού στοιχείου της homography. Αν

$$I = [t_1 \quad t_2 \quad t_3]$$

αντιπροσωπεύει την παρατηρούμενη γραμμή στο άπειρο τότε ο αμιγής μετασχηματισμός για τη διόρθωση της προοπτικής δίνεται από τον πίνακα:

$$H_p = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ I_1 & I_2 & I_3 \end{pmatrix}$$

Με την εφαρμογή του προβολικού μετασχηματισμού δεν διατηρείται ούτε η παραλληλία ούτε η καθετότητα. Αφαιρώντας τον αμιγή προβολικό μετασχηματισμό η καθετότητα δεν διατηρείται ενώ η παραλληλία παραμένει. Το αφινικό στοιχείο μπορεί να προσδιοριστεί χρησιμοποιώντας ζεύγη κάθετων γραμμών. Έχει προσδιοριστεί μετασχηματισμός ο οποίος μπορεί να διορθώσει την γωνία μεταξύ ενός ζεύγους γραμμών. Με την εύρεση μετασχηματισμού που αντιστοιχεί σε ένα ζευγάρι κάθετων γραμμών μπορούμε να αφαιρέσουμε το αφινικό στοιχείο. Όταν σε μία εικόνα έχει αφαιρεθεί και ο προβολικός μετασχηματισμός και το αφινικό στοιχείο τότε μένει μόνο ένα στοιχείο ομοιότητας. Αυτή η εικόνα είναι ιδανική για συμβατικά έγγραφα.

Η παρούσα πτυχιακή προσεγγίζει τη λύση του προβλήματος της προοπτικής παραμόρφωσης γεωμετρικά και η μεθοδολογίες που χρησιμοποιεί συγκαταλέγονται στη συγκεκριμένη κατηγορία.

3.2.1 Διόρθωση της προοπτικής παραμόρφωσης με τη χρήση του Harris Corner detector

Η μέθοδος αυτή χρησιμοποιεί τον Harris Corner detector για την ανίχνευση των τεσσάρων κορυφών του εγγράφου και βασίζεται στην εργασία των Geetha Kiran A και Murali S [10] το 2013. Η μέθοδος αυτή μπορεί να περιγραφεί συνοπτικά από τα παρακάτω στάδια (Εικόνα 18) :

1. Προεπεξεργασία
2. Ανίχνευση ακμών
3. Εφαρμογή του αλγορίθμου Harris
4. Εύρεση τεσσάρων επιθυμητών γωνιών
5. Υπολογισμός homography
6. Χρήση του πίνακα homography για τον μετασχηματισμό της εικόνας

Πιο αναλυτικά η παρουσίαση των σταδίων:

Προεπεξεργασία:

Αρχικά η εικόνα μετατρέπεται σε grayscale και στη συνέχεια εφαρμόζεται εύρεση ακμών

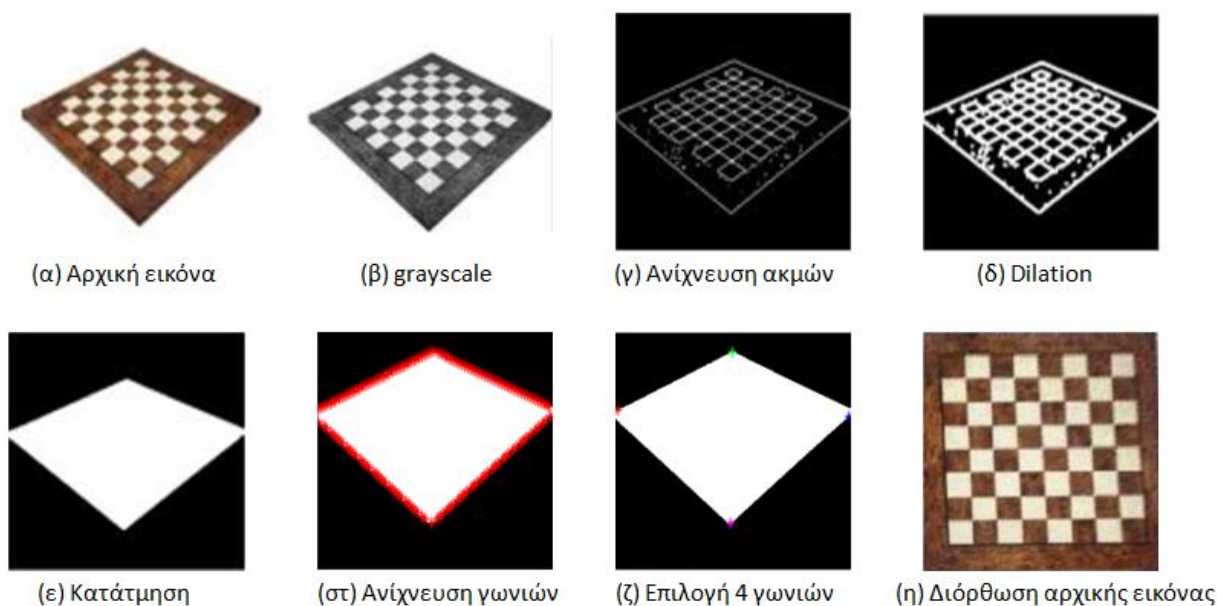
Κατάτμηση Εικόνας

Αν και θεωρείται κομμάτι της προ επεξεργασίας δεν παύει να είναι το σημαντικότερο στάδιο καθώς τα αποτελέσματά του επηρεάζουν σημαντικά την εγκυρότητα της διόρθωσης της προοπτικής καθώς και την ποιότητα του τελικού αποτελέσματος. Στο στάδιο αυτό τροφοδοτείται η εξομαλυμένη εικόνα σε έναν τελεστή ανίχνευσης ακμών. Η μέθοδος αυτή χρησιμοποιεί τον τελεστή εύρεσης ακμών Sobel [3]. Ο τελεστής Sobel, όπως αναφέρθηκε στο κεφάλαιο 2, υπολογίζει τα gradient ως προς τις κατευθύνσεις x και y και στη συνέχεια της συνδυάζει για να βρει την ολική gradient εικόνα.

Μετά την εφαρμογή του Sobel φίλτρου χρησιμοποιούνται μορφολογικές πράξεις dilation και erosion για την κατάτμηση της εικόνας (Image Segment) [1] [2]. Μεγαλύτερες τιμές παίρνει το δομικό στοιχείο του dilation ώστε να κλείσει το περίγραμμα του εγγράφου της εικόνας, το οποίο σε περίπτωση αρκετού θορύβου έχει χαθεί με αποτέλεσμα να χάνονται οι πραγματικές ακμές. Με την εφαρμογή του dilation διογκώνονται οι ακμές και βελτιώνονται τα αποτελέσματα ως προς τις εξωτερικές ακμές (ακμές περιγράμματος). Η εφαρμογή του erosion επιτυγχάνει την επαναφορά του αντικειμένου στις αρχικές του διαστάσεις. Η κατατμημένη εικόνα χρησιμοποιείται σαν είσοδος στο επόμενο βήμα που είναι η ανίχνευση γωνιών.

Harris Corner Estimation

Στο στάδιο αυτό οι γωνίες υπολογίζονται εφαρμόζοντας τον αλγόριθμο των Harris και Stephens [6]. Το σκορ που υπολογίζεται για κάθε pixel στον ανιχνευτή γωνιών βασίζεται σε δύο ιδιοτιμές ενός πίνακα. Μεταξύ των διαθέσιμων σημείων που έχουν ανιχνευτεί ως γωνίες γίνεται η επιλογή των τεσσάρων γωνιών που θα χρησιμοποιηθούν για το μετασχηματισμό της επιφάνειας. Εφόσον έχουν λειτουργήσει σωστά τα προηγούμενα βήματα οι γωνίες που ανιχνεύονται εντοπίζονται σε ομάδες κοντά στα πραγματικά σημεία και η επιλογή των τεσσάρων γίνεται με βάση το μεγαλύτερο τετράπλευρο που μπορεί να οριστεί με κάθε συνδυασμό. Τέλος, εφόσον έχουν προσδιοριστεί τα τέσσερα σημεία εφαρμόζεται ο υπολογίζεται η homography όπως περιγράφεται στο [11]. Οι δυο τεχνικές που έχουν επιλεχθεί για εκτενέστερη μελέτη, δε διαφέρουν στο τελικό βήμα αλλά ένα στάδιο πριν εφαρμοστεί η homography γι' αυτό και στο πέμπτο κεφάλαιο που γίνεται αποτίμηση των μεθόδων, η σύγκρισή τους γίνεται μέχρι το στάδιο εύρεσης γωνιών.



Εικόνα 18: Βήματα της μεθόδου Geetha Kiran A. και Murali S.

3.2.2 Διόρθωση της προοπτικής παραμόρφωσης με τη χρήση του Hough Transform

Σε αντίθεση με την προηγούμενη μέθοδο που είχε ως σύμβαση την ύπαρξη και των τεσσάρων κορυφών, δηλαδή να βρίσκεται όλο το σώμα του περιγράμματος εντός της εικόνας, η παρούσα μέθοδος έχει ως σύμβαση την εμφάνιση των άνω και κάτω ορίων του αντικειμένου στην εικόνα και εστιάζει στην εύρεση μόνο αυτών των ορίων. Η μέθοδος αυτή χρησιμοποιεί τον Hough Transform [7] για την ανίχνευση δύο κορυφών του εγγράφου και βασίζεται στην δουλειά των Ryan Baumann, Christopher Blackwell και W. Brent Seales [11] το 2012

Μπορεί να περιγραφεί συνοπτικά από τα παρακάτω στάδια (Εικόνα 19):

1. Κατάτμηση εικόνας - διαχωρισμός εικόνας από υπόβαθρο
2. Χρήση του Αλγορίθμου του Canny για εύρεση ακμών και χρήση dilation
3. Hough line detection για την εύρεση των ευθειών
4. Classification (ομαδοποίηση) των ευθειών σε άνω και κάτω γραμμές περιγράμματος
5. Εύρεση των μέσων τιμών των ευθειών για άνω και κάτω και εύρεση των 2 αντιπροσωπευτικών μέσων τους.
6. Υπολογισμός homography
7. Χρήση του πίνακα homography για τον μετασχηματισμό της εικόνας

Προεπεξεργασία εικόνας

Περιλαμβάνει τα βήματα 1 και 2. Αρχικά η εικόνα μετατρέπεται σε grayscale και στη συνέχεια εφαρμόζεται ένα Gaussian φίλτρο για διάφορες τιμές της τυπικής απόκλισης σ ώστε να εξομαλυνθεί και να είναι ευκολότερο να εντοπιστούν τα σημαντικά χαρακτηριστικά της. Στη συνέχεια γίνεται εφαρμογή μιας μάσκας ώστε να γίνει σωστά ο

διαχωρισμός εικόνας και υποβάθρου. Έπειτα, χρησιμοποιείται η τεχνική ανίχνευσης ακμών του Canny για να εντοπιστούν τα χαρακτηριστικά της εικόνας και να γίνει πιο εύκολα η εύρεση ακμών που γίνεται σε επόμενο βήμα.

Εύρεση ευθειών Hough

Χρησιμοποιείται ο probabilistic Hough transform [12] μια μέθοδος που αντί για ευθείες, επιστρέφει ευθύγραμμα τμήματα για κάθε ευθεία που ανιχνεύει. Στη συνέχεια, γίνεται κατηγοριοποίηση των ευθύγραμμων τμημάτων σε άνω και κάτω ευθύγραμμα τμήματα της εικόνας περιορίζοντας τα σε αυτά που έχουν κλίση $\pm 45^\circ$, απορρίπτοντας τις γραμμές που βρίσκονται στα δεξιά και αριστερά της εικόνας.

Σε δεύτερο βήμα, έχοντας σε δυο ομάδες τις άνω και κάτω ευθείες, γίνεται εύρεση των μέσων τιμών των ευθειών. Με αυτόν τον τρόπο, περιορίζονται οι ευθείες σε δυο και στη συνέχεια, επεκτείνονται ώστε να φτάσουν τα όρια της εικόνας.

Τέλος, δημιουργείται ένας νέος πίνακας εικόνας του τετράπλευρου που σχηματίζεται από τις γραμμές με την παραδοχή ότι η άνω ευθεία θα βρίσκεται στο υψηλότερο σημείο ώστε να σχηματίζεται ίσια γραμμή και παρόμοια η κάτω ευθεία στο κατώτερο όριο ώστε να σχηματίζεται ένα παραλληλόγραμμο. Χρησιμοποιούμε τα σημεία της νέας εικόνας για τον υπολογισμό και την εφαρμογή του προβολικού μετασχηματισμού και το αποτέλεσμα είναι η διόρθωση της παραμόρφωσης της εικόνας (Εικόνα 19).



Εικόνα 19: Βήματα της μεθόδου R.Baumann, C.Blackwell και W. B. Seales

Homography

Το τελευταίο στάδιο είναι ίδιο και για τις δυο μεθόδους. Η homography για τον 2D ορίζεται ως ένας 3x3 ομογενείς πίνακας που αντιστοιχίζει κάθε σημείο $p(x,y)$ σε μια επιφάνεια με ένα σημείο $p'(x',y')$ με την παρακάτω σχέση

$$P' = H \cdot P \quad (15)$$

Η homography μπορεί να περιγραφεί ως ένας γεωμετρικός προβολικός μετασχηματισμός. όπου τα p και p' μετατρέπονται σε ομογενείς συντεταγμένες

$$\begin{pmatrix} wx' \\ wy' \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{21} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix}$$

Λύνοντας τις

$$x' = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}} \quad \text{και} \quad y' = \frac{h_{12}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}$$

Έχουμε:

$$h_{12}x + h_{22}y + h_{23} - h_{31}xy' - h_{32}yy' - h_{33}y' = 0$$

$$h_{11}x + h_{12}y + h_{13} - h_{31}xx' - h_{32}yx - h_{33}x' = 0$$

Ο πίνακας A είναι ένας 2 x 9 πίνακας βασισμένος στις γνωστές συντεταγμένες x_i, y_i, x'_i, y'_i και γράφεται ως εξής:

$$\begin{bmatrix} x_i & y_i & 1 & 0 & 0 & 0 & -x'_i x_i & -x'_i y_i & -x'_i \\ 0 & 0 & 0 & x_i & y_i & 1 & -y'_i x_i & -y'_i y_i & -y'_i \end{bmatrix} \quad (18)$$

Τελικά η εξίσωση (15) μπορεί να γραφτεί ως:

$$P' = H \cdot P \rightarrow A_i \cdot h = 0 \text{ για } i = 1,2,3,4. \quad (19)$$

Όπου A_i είναι ένας 2 x 9 πίνακας και ο h είναι ένας 9 x 1 πίνακας ενώ το αποτέλεσμα είναι ένας

2 x 1 πίνακας. Αν προσπαθήσουμε να γράψουμε όλες τις τέσσερις σχέσεις πινάκων σε μία εξίσωση τότε το αποτέλεσμα θα είναι το εξής:

$$\begin{pmatrix} A_1 \\ A_2 \\ A_3 \\ A_4 \end{pmatrix} \cdot h = 0 \quad (20)$$

Όπου οι διαστάσεις του πίνακα A_i είναι $(4 \times 2) \times 9 = 8 \times 9$ και του πίνακα h 9×1 και το αποτέλεσμα ένας 8×1 πίνακας. Η συνισταμένη έκφραση μπορεί να γραφτεί ως εξής:

$$A_i \cdot h = 0 \quad (21)$$

Συνεπάγεται ότι υπάρχουν 8 γραμμικές εξισώσεις και 9 άγνωστοι. Προσθέτοντας τον περιορισμό $\|h\| = 1$, ο οποίος απλοποιείται στον $A \cdot h = 0$. Η επίλυση γίνεται με τη χρήση της Singular Value Decomposition για τέσσερα σημεία που αντιστοιχούν σε δύο επίπεδα.

Ο μετασχηματισμός δίνεται από την επίλυση όπως εξηγήθηκε παραπάνω. Το στοιχείο A_{1-s} δίνεται από:

$$\begin{bmatrix} X1 & Y1 & 1 & 0 & 0 & 0 & -X1 \cdot x1 & -Y1 \cdot x1 \\ 0 & 0 & 0 & X1 & Y1 & 1 & -X1 \cdot y1 & -Y1 \cdot y1 \\ X2 & Y2 & 1 & 0 & 0 & 0 & -X2 \cdot x2 & -Y2 \cdot y2 \\ 0 & 0 & 0 & X2 & Y2 & 1 & -X2 \cdot y2 & -Y2 \cdot y2 \\ X3 & Y3 & 1 & 0 & 0 & 0 & -X3 \cdot x3 & -Y3 \cdot x3 \\ 0 & 0 & 0 & X3 & Y3 & 1 & -X3 \cdot y3 & -Y3 \cdot y3 \\ X4 & Y4 & 1 & 0 & 0 & 0 & -X4 \cdot x4 & -Y4 \cdot x4 \\ 0 & 0 & 0 & X4 & Y4 & 1 & -X4 \cdot y4 & -Y4 \cdot y4 \end{bmatrix}$$

Και τα διανύσματα δίνονται ως

$$h_{1-s} = (\alpha_{11}, \alpha_{12}, \alpha_{13}, \alpha_{21}, \alpha_{22}, \alpha_{23}, \alpha_{31}, \alpha_{32})^T,$$

$$A_9 = (x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4)^T$$

Όταν ο μετασχηματισμός εφαρμόζεται σε παραμορφωμένες εικόνες η παραμόρφωση θα διορθωθεί.

Συνοψίζοντας ο μετασχηματισμός ακολουθεί τα εξής:

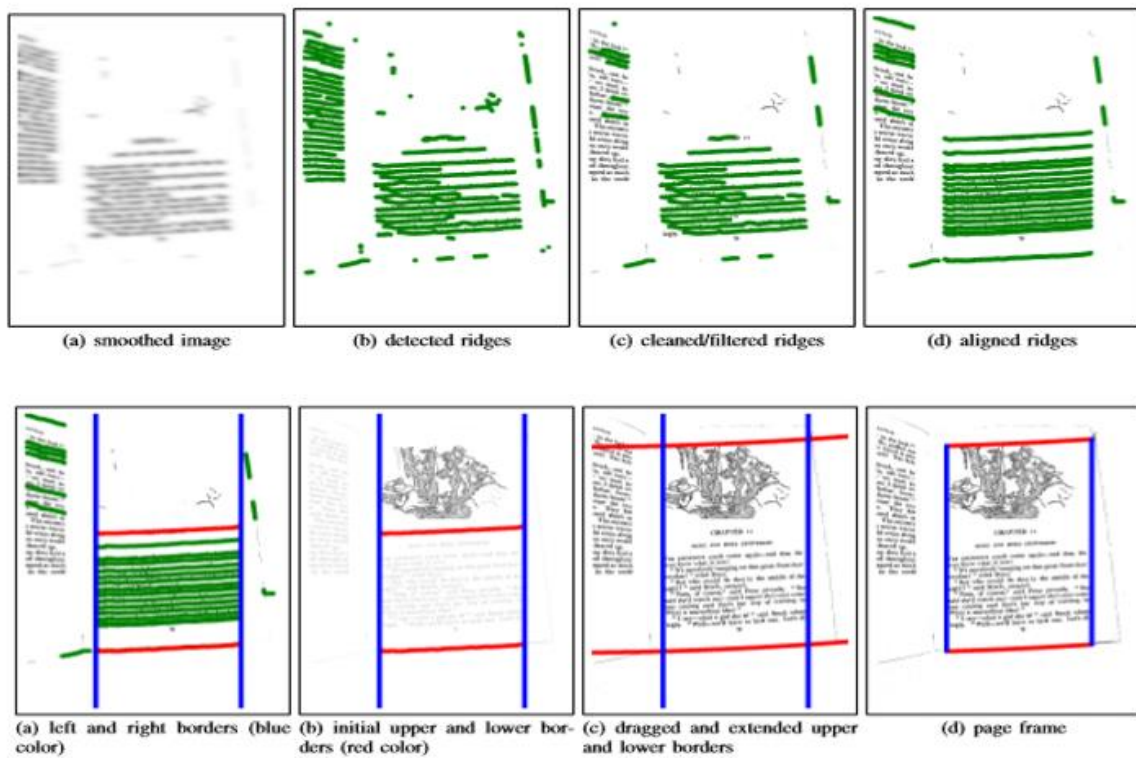
A) Εισάγει την κατακερματισμένη εικόνα και τα τέσσερα γωνιακά σημεία

B) Ο προβολικός μετασχηματισμός εφαρμόζεται στις συντεταγμένες της εικόνας εισόδου

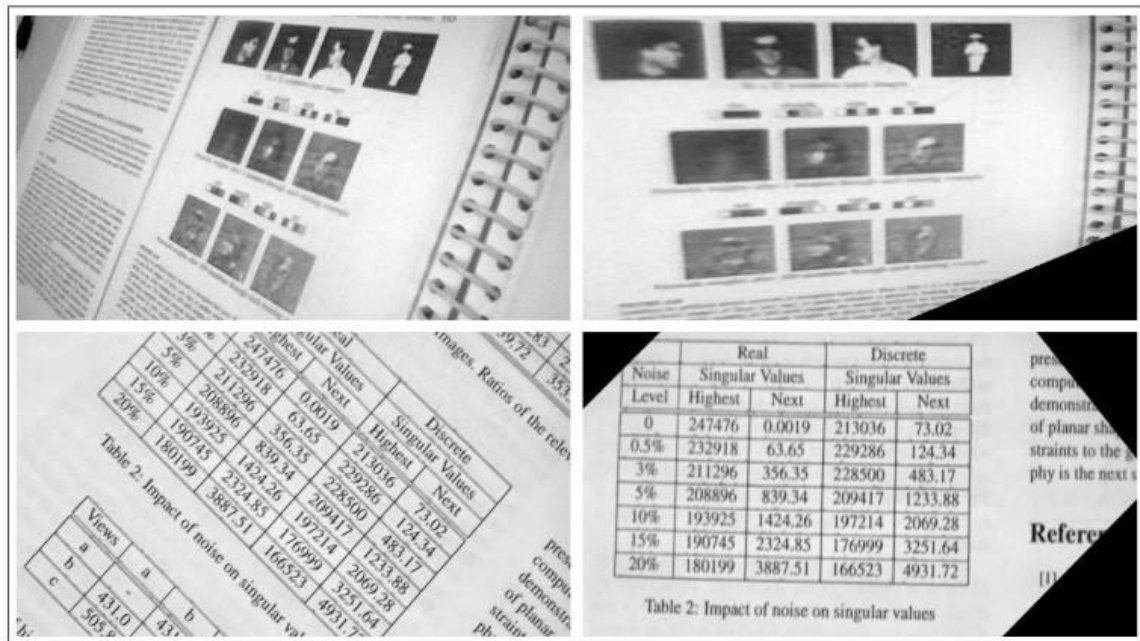
Γ) Η εικόνα που προκύπτει είναι η μετασχηματισμένη εικόνα.

3.3 Μεθοδολογίες που χρησιμοποιούν τη διάταξη της σελίδας.

Η διάταξη του αρχείου μπορεί να χρησιμοποιηθεί για διόρθωση της προοπτικής σε μια εικόνα εγγράφου. Ένα μεγάλο πλήθος τεχνικών προσδιορίζει τα χαρακτηριστικά της εικόνας και τη χωρίζει σε τμήματα (εικόνα, κείμενο, γραφικά βλέπε Εικόνα 21). Με βάση τη δουλειά των Syed Saqib Bukhari, Faisal Shafait, Thomas M. Breuel [22], αφαιρώντας τα γραφικά στοιχεία και κάνοντας κατάτμηση κειμένου σε επίπεδο γραμμών είναι εύκολο να προσδιοριστούν τα όρια του κειμένου και να γίνει η διόρθωση της εικόνας (Εικόνα 20).



Εικόνα 20: Border noise removal through page frame detection



Εικόνα 21: Correction of Perspective Distortion using Layout Information. Κατάτμηση σε εικόνα , κείμενο , γραφικά.

Η διόρθωση των πινάκων και μορφών οδηγεί σε μεγάλο αριθμό εξισώσεων για την εκτίμηση της homography και τη διόρθωση της προοπτικής. Όταν υπάρχουν πάνω από τέσσερα σημεία ανταπόκρισης (ή 8 εξισώσεις) για την εκτίμηση της homography η διόρθωση μπορεί να γίνει καλύτερα. Για τον υπολογισμό της νέας εικόνας χρησιμοποιείται το homography και τα τέσσερα σημεία ανταπόκρισης λύνοντας την

$$x_i = Hx'_i$$

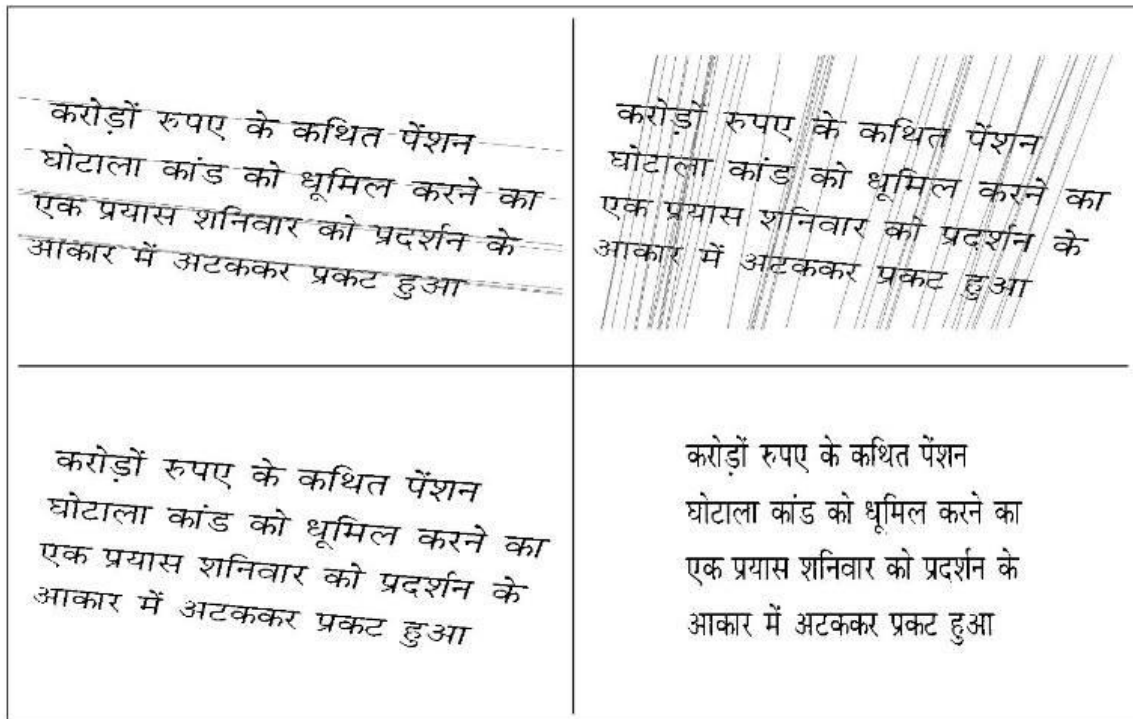
Για παραπάνω από τέσσερα σημεία το σύστημα των ομογενών εξισώσεων είναι της μορφής $Ah = 0$ όπου η Homography H διατάσσεται σαν ένα 9×1 διάνυσμα h . Η λύση αυτού του συστήματος εξισώσεων είναι το ιδιοδιάνυσμα που αντιστοιχεί στη μικρότερη ιδιοτιμή του $A^T A$. Για την αριθμητική σταθερότητα, οι συντεταγμένες της εικόνας στον A είναι κανονικοποιημένες ώστε να είναι επικεντρωμένες γύρω από το 0 και έχουν μονάδες διακύμανσης. Τα όρια του κειμένου μπορούν να χρησιμοποιηθούν για τον υπολογισμό της Homography [11]. Όταν η εικόνα δεν περιέχει γραφικά στοιχεία, η homography προκύπτει από την δομή του ίδιου του κειμένου.

3.4 Μεθοδολογίες που χρησιμοποιούν ειδικές διορθώσεις περιεχομένου

Τα όρια του εγγράφου και η διάταξη της σελίδας δίνουν χρήσιμες πληροφορίες που μπορούν να χρησιμοποιηθούν στη διόρθωση της προοπτικής της εικόνας. Όμως, όταν το κείμενο δεν περιέχει αρκετές προτάσεις ή λέξεις, λίγα πράγματα είναι γνωστά για την διάταξη της σελίδας ή για τα όριά της. Η ακριβής γνώση για τα ορθογώνια ή τα τετράπλευρα δεν υπάρχει και ως εκ τούτου χρειαζόμαστε συγκεκριμένες πληροφορίες που είναι χρήσιμοι για την διόρθωση της προοπτικής.

Αυτή η κατηγορία μεθόδων χρησιμοποιεί κάθε γραμμή του κειμένου για να διορθώσει την παραμόρφωση της προοπτικής. Η μέθοδος που προτείνεται στο «Goal-oriented Rectification of Camera-Based Document Images» [23] διορθώνει την προοπτική και την καμπυλότητα που παρουσιάζεται στην εικόνα εφαρμόζοντας το μετασχηματισμό μια φορά στα όρια του κειμένου όπως γίνεται με τις τεχνικές της προηγούμενης κατηγορίας και στη συνέχεια εφαρμόζει για δεύτερη φορά έναν μετασχηματισμό στις γραμμές που έχει κάνει κατάτμηση λειτουργώντας με παρόμοιο τρόπο αλλά σε επίπεδο γραμμής αυτή τη φορά.

Μια άλλη βασική ιδέα που παρουσιάζεται στο [21] στηρίζεται στην εύρεση των οριζόντιων και κάθετων vanishing points. Πρώτα, προσδιορίζονται τα οριζόντια vanishing points με τον αλγόριθμο του [21] και στη συνέχεια με τη χρήση της τεχνικής του μετασχηματισμού Hough φέρνονται ευθείες, από τις οποίες απορρίπτονται όσες βρίσκονται κοντά στο οριζόντιο vanishing point και τέμνονται στο κάθετο vanishing point. Με αυτή τη διαλογή υπολογίζεται ένα σετ ευθειών από τις οποίες υπολογίζεται το κάθετο vanishing point.



Εικόνα 22: Perspective correction of Hindi Text. Προσδιορισμός των οριζόντιων και κάθετων vanishing points

3.5 Εργαλεία και υπάρχουσες εφαρμογές

Κάποιες εφαρμογές που συναντούμε στην αγορά :

- Evernote Scannable

<https://evernote.com/products/scannable/>

- Genius-scan

<http://appcrawlr.com/android/genius-scan-pdf-scanner>

- Cam scanner

<https://play.google.com/store/apps/details?id=com.intsig.camscanner>

- Office lens

<https://www.microsoft.com/el-gr/store/apps/office-lens/9wzdncrfj3t8>

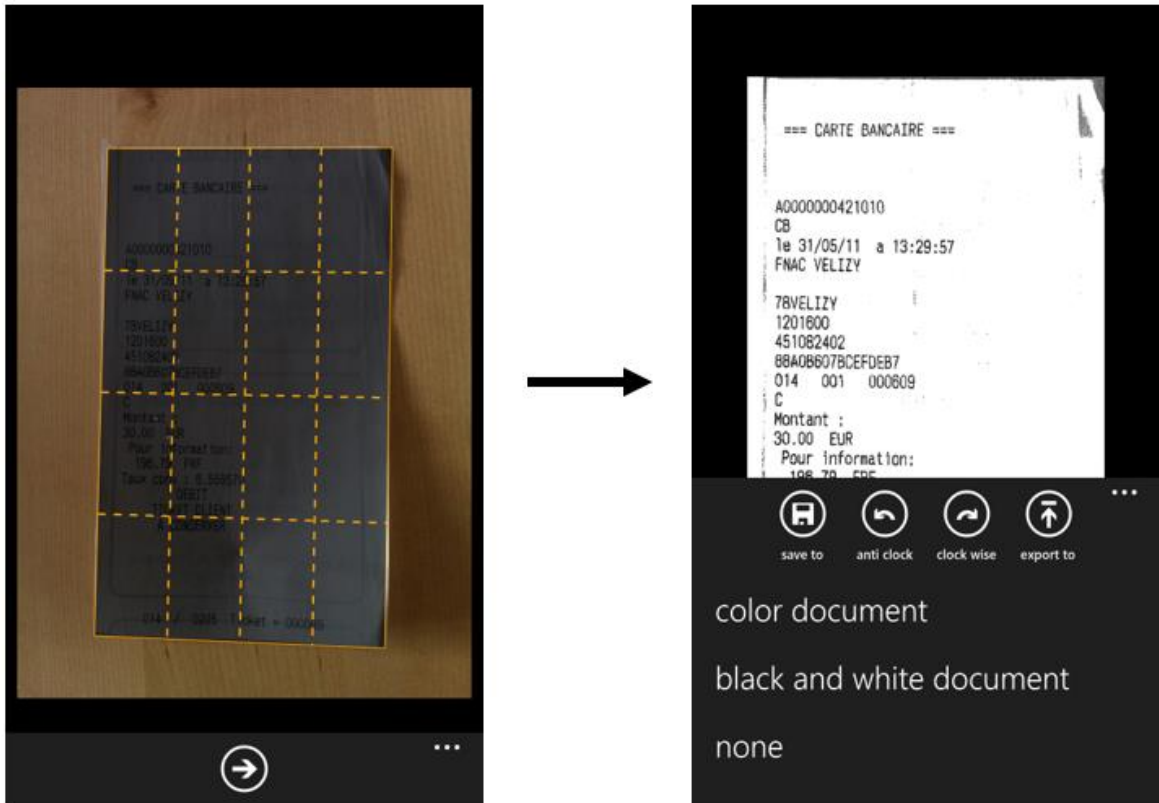
- Captureboard

<https://itunes.apple.com/gr/app/captureboardpro-scan-whiteboard/id619547169?mt=8>

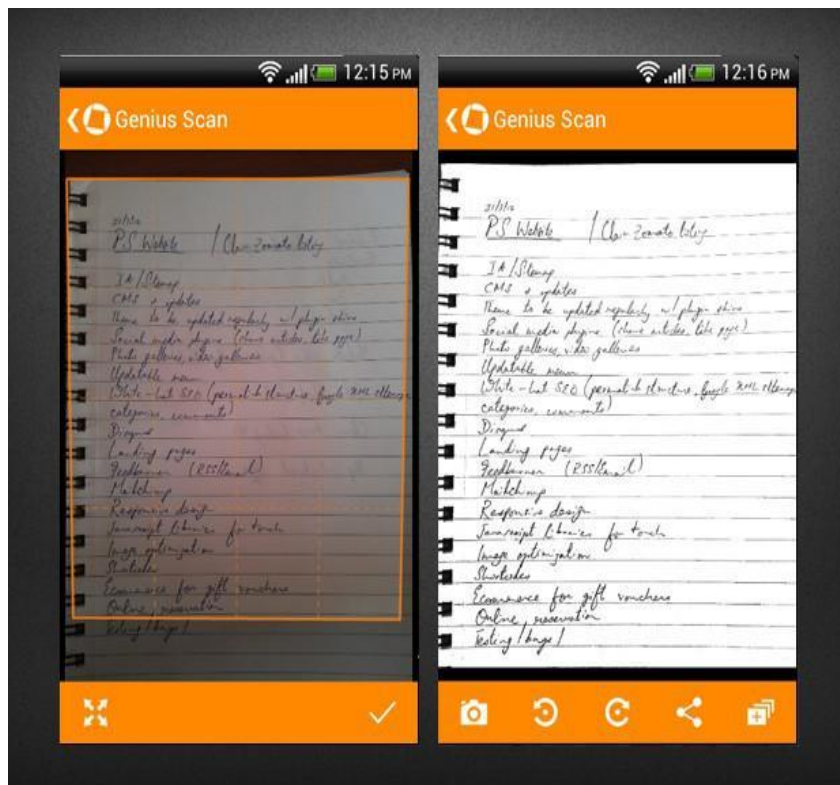
- Finescanner

<http://www.finescanner.com/>

Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφα από κάμερα

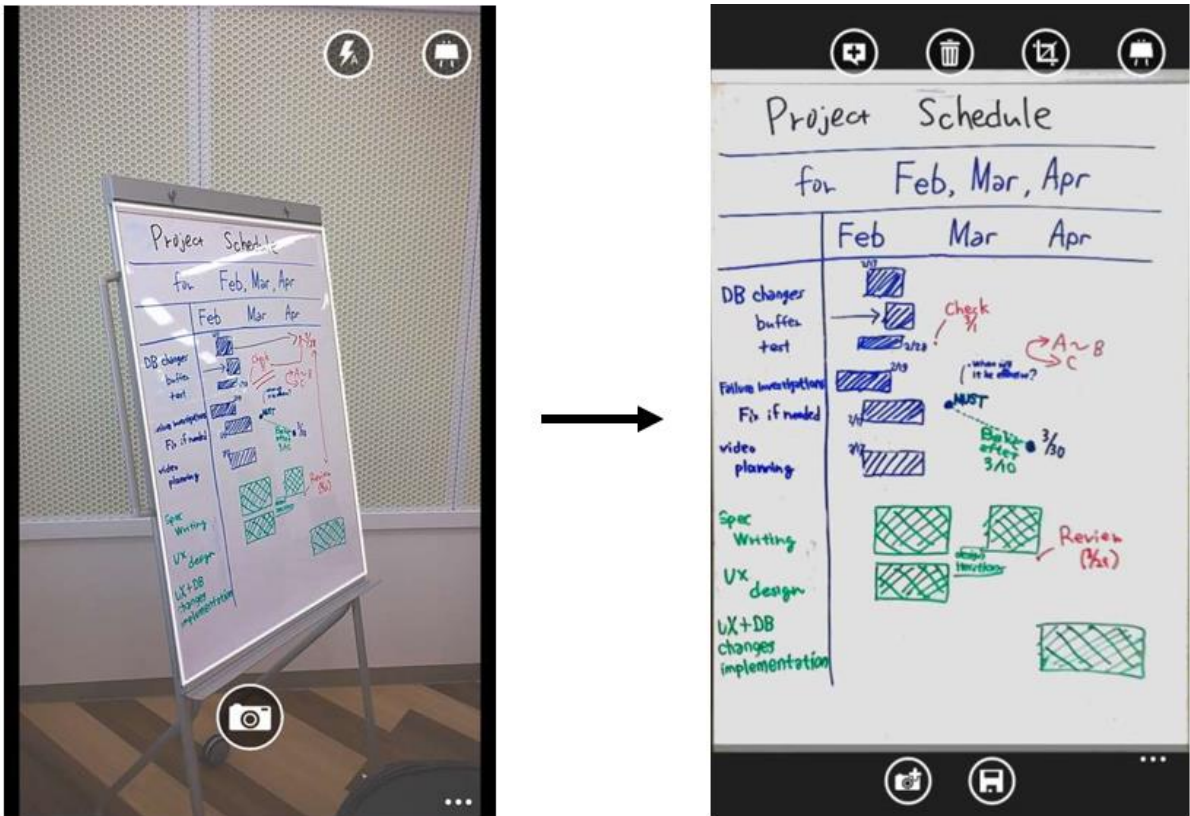


Εικόνα 23: Finescanner

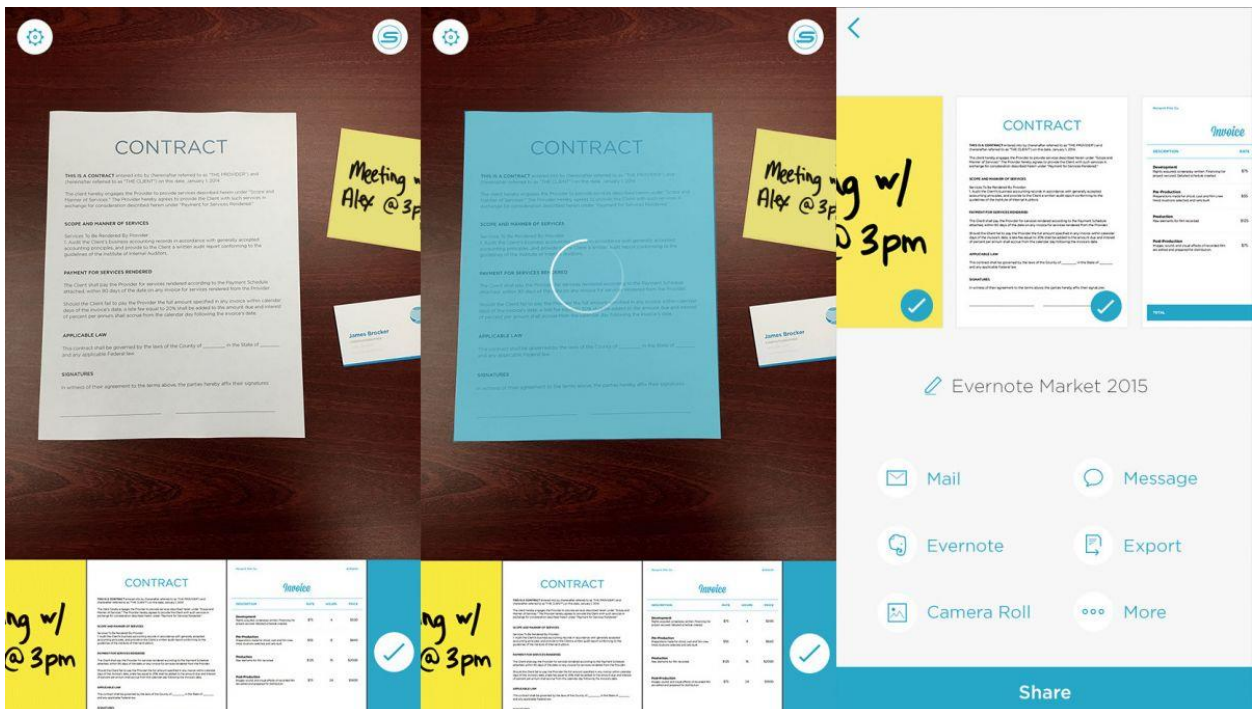


Εικόνα 24: Genius Scan

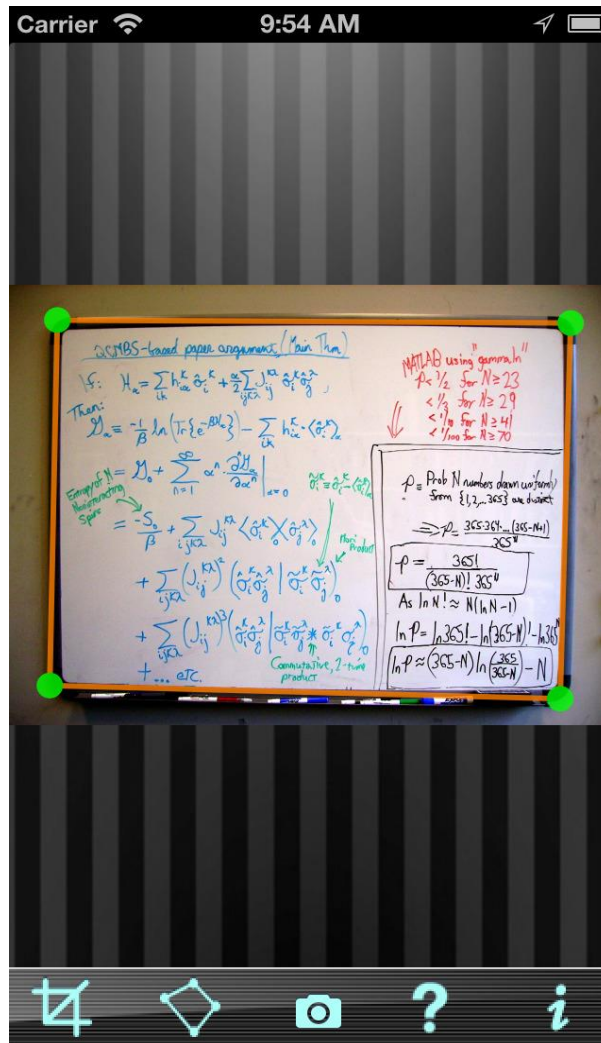
Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφα από κάμερα



Εικόνα 25: Office Lens



Εικόνα 26: Scannable



Εικόνα 27: Capture Board

4. ΠΡΟΤΕΙΝΟΜΕΝΕΣ ΜΕΘΟΔΟΙ

Στο κεφάλαιο αυτό θα παρουσιαστούν κάποιες επεκτάσεις που προσαρτήθηκαν στις παραπάνω μεθοδολογίες [10] [11] ανά στάδιο για να βελτιωθούν τα αποτελέσματα που παρουσίασαν.

4.1 Μέθοδος Harris

Με βάση τη μεθοδολογία που προτάθηκε από τους Geetha Kiran A και Murali S [10] πραγματοποιήθηκε η υλοποίηση και η εξέταση τρόπων βελτίωσης των αποτελεσμάτων. Στα βήματα που αναφέρθηκαν στο κεφάλαιο 3 παράγραφος 3.2.1 έγιναν κάποιες αλλαγές, τα αποτελέσματα των οποίων συγκρίνουμε με τα αποτελέσματα της κοντινότερης στο [10] μεθοδολογίας. Συνοπτικά οι αλλαγές είναι:

- Προσθήκη Smoothing με (Gaussian blur ή median blur)
- Αλλαγή μεθόδου εύρεσης ακμών (Canny algorithm)
- Προσθήκη δυνατότητας αλλαγής παραθύρου στο dilation
- Αντικατάσταση του erosion με εύρεση των contours
- Χρήση του μεγίστου contour και απομάκρυνση του θορύβου

- Εύρεση του μέγιστου τετραπλεύρου

Αλλαγή στον υπολογισμό της ομογραφίας

- Διαστάσεις νέας εικόνας (πλάτος : μικρότερη οριζόντια απόσταση, μήκος : μικρότερη κάθετη απόσταση μεταξύ δυο σημείων)
- Υπολογισμός ομογραφίας με δυο τρόπους είτε για $||H|| = 1$, είτε $h33 = 1$
- Υπολογισμός εικόνας μέσω της αντιστροφής της ομογραφίας

Σαν υλοποίηση του [10] θεωρούμε τα αποτελέσματα που προκύπτουν από τα παραπάνω βήματα για τη χρήση median blur στο στάδιο εξομάλυνσης και Sobel operator στην ανίχνευση ακμών .Πιο αναλυτικά η περιγραφή των σταδίων περιγράφεται παρακάτω.

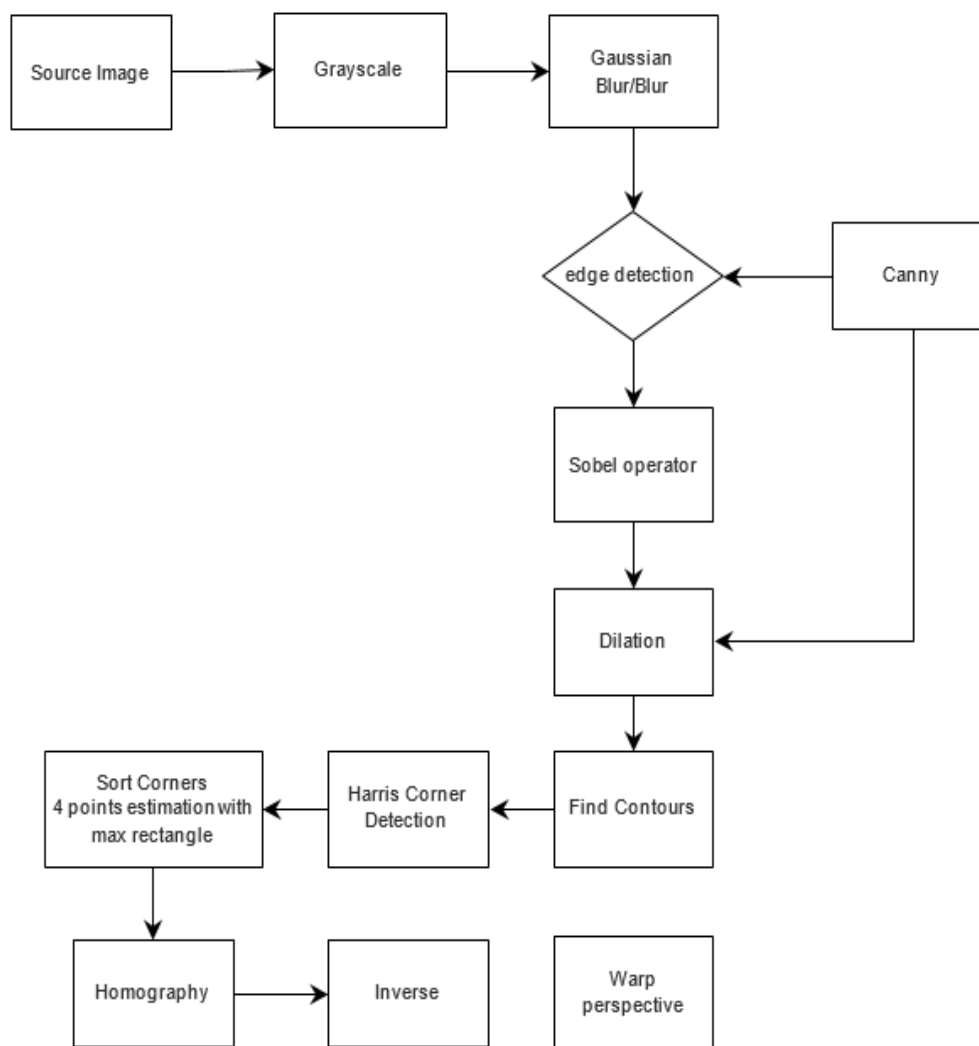
Αρχικά, μετά την μετατροπή της εικόνας σε grayscale, εφαρμόζεται ένα Gaussian φίλτρο για διάφορες τιμές της τυπικής απόκλισης σ ώστε να εξομαλυνθεί και να είναι ευκολότερο να εντοπιστούν τα σημαντικά χαρακτηριστικά της. Στο στάδιο της κατάτμησης έγιναν κάποιες αλλαγές για καλύτερα αποτελέσματα. Αρχικά, στη μέθοδο για εύρεση ακμών χρησιμοποιήθηκε εναλλακτικά ο αλγόριθμος Canny που κάνει καλύτερη εκτίμηση των ακμών. Επίσης η μέθοδος Sobel που χρησιμοποιείται ως αρχική ανίχνευση ακμών είναι ενσωματωμένη στη συνάρτηση του Harris. Μετά την εύρεση ακμών ακολουθεί το κομμάτι που αναφέρεται ως κατάτμηση εικόνας στο οποίο σύμφωνα με την εργασία των Geetha Kiran.A και Murali.S [10] εφαρμόζεται dilation και ακολουθείται από erosion. Στην πράξη παρατηρήθηκε ότι τα αποτελέσματα μετά την εφαρμογή του erosion δεν ήταν ικανοποιητικά και όπως παρουσιάζοταν στο [10] (Εικόνα 17) με αποτέλεσμα το επόμενο βήμα που ήταν η ανίχνευση γωνιών να μη μπορεί να δώσει τα αναμενόμενα αποτελέσματα. Δοκιμάστηκαν διάφορες τιμές για το στάδιο της κατάτμησης αλλά κανένας συνδυασμός δεν βελτίωνε πολύ τα αποτελέσματα. Οι γωνίες που ανιχνεύονταν από τον αλγόριθμο Harris βρίσκονταν τυχαία σε όλη την έκταση της εικόνας και δεν ήταν δυνατό να περιοριστούν στις τέσσερις πραγματικές καθώς δεν υπήρχαν κάποια κριτήρια. Κάθε περίπτωση χρειαζόταν ξεχωριστό threshold για να έχει έγκυρα αποτελέσματα. Ο αλγόριθμος Harris φαίνεται να μη δουλεύει τόσο καλά όσο φαίνονται λεπτομέρειες της εικόνας. Ο μόνος τρόπος για τη σωστή ανίχνευση γωνιών με τον αλγόριθμο αυτό είναι η βελτίωση του τμήματος της προεπεξεργασίας.

Αρχικά ,δοκιμάστηκε η εφαρμογή του floodfill, του OpenCV μια άλλη μορφολογική πράξη που γεμίζει το εσωτερικό του περιγράμματος ενός αντικειμένου μετατρέποντάς το σε πολύ καλή είσοδο για την ανίχνευση γωνιών. Αυτό είχε καλύτερα αποτελέσματα αλλά δούλευε μόνο για περιορισμένες εικόνες. Η λύση δόθηκε προσδιορίζοντας το περίγραμμα του αντικειμένου με τη χρήση findContours OpenCV. Έτσι μετά το στάδιο του dilation βρίσκονται όλα τα περιγράμματα και γεμίζουμε το εσωτερικό του μέγιστου. Γίνεται έλεγχος εάν έχει κλείσει το περίγραμμα πριν αφαιρεθεί οποιαδήποτε χρήσιμη πληροφορία. Επιλέγεται το μέγιστο περίγραμμα εφόσον το μέγεθός του ξεπερνάει ένα κατώτερο όριο που έχει τεθεί και αποτελεί ένα ποσοστό του μεγέθους της εικόνας (15%).

Με τη χρήση των contours διαχωρίζεται το υπόβαθρο από την πληροφορία και οι γωνίες που προκύπτουν από την εφαρμογή του αλγορίθμου Harris και Stephens είναι σε ομάδες

συγκεντρωμένες στις τέσσερις γωνίες του εγγράφου. Επομένως, επιλέγεται η τετράδα που δημιουργεί το μεγαλύτερο τετράπλευρο (Εικόνα 32)και εφαρμόζεται η ομογραφία.

Παρακάτω βλέπουμε ένα διάγραμμα ροής του αλγορίθμου που υλοποιήθηκε.



Σχήμα 1: Διάγραμμα ροής αλγορίθμου Harris

Harris Corner Detection

Step 1: Εξομάλυνση εικόνας με τη χρήση Gaussian Blur ή Median Blur

Step 2: Χρήση ενός αλγορίθμου ανίχνευσης ακμών στην εξομαλυμένη grayscale εικόνα. Για το edge detection γίνεται χρήση του Sobel operator ή του αλγορίθμου Canny ανάλογα με την τιμή μιας μεταβλητής που έχουμε ορίσει, τη method_flag, η οποία παίρνει τιμές 2 και 1 αντίστοιχα.

Step 3: Χρήση του μορφολογικού τελεστή Dilation με δομικό στοιχείο το RECT και μέγεθος παραθύρου να ισούται με τρία ή πέντε αναλόγως την τιμή της blur_flag μεταβλητής (1 , 0 αντίστοιχα)

Step 4: Εύρεση του περιγράμματος (Find contours) της edged εικόνας και διατήρηση μόνο του μέγιστου περιγράμματος αν αυτό ξεπερνά ένα ποσοστό του μεγέθους της εικόνας (10-15%)

Step 5 : Χρήση του Harris Corner Detector για την ανίχνευση των γωνιών

Step 6 : Αφαίρεση των εσωτερικών συνευθειακών σημείων. Επιλογή όλων των τριάδων από το σύνολο σημείων και έλεγχος αν είναι και τα τρία συνευθειακά. Σε περίπτωση που είναι, θα απορριφθεί το εσωτερικό σημείο. Μετά από αυτό το βήμα και τον περιορισμό των σημείων σε σημεία που αντιστοιχούν στο περίγραμμα γίνεται εύρεση τεσσάρων σημείων που σχηματίζουν το μεγαλύτερο τετράπλευρο που σχηματίζεται από όλες τις τετράδες σημείων.

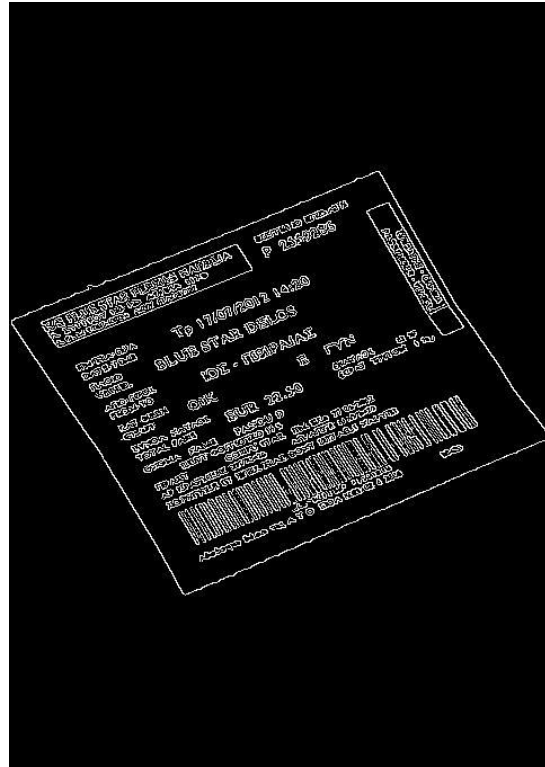
Step 7 : Εφαρμογή της homography

Παρακάτω παρουσιάζονται τα αποτελέσματα από την υλοποίηση της μεθόδου:

Step 2: Edge detection



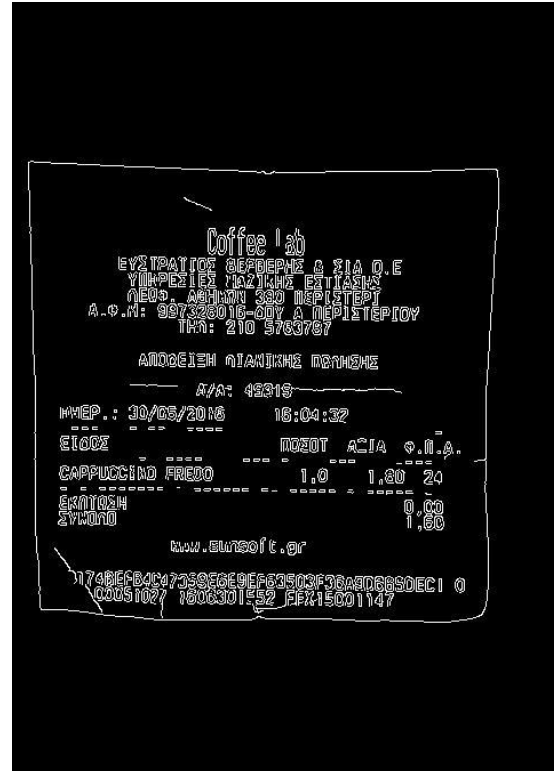
Εικόνα 28: Εφαρμογή του τελεστή Sobel



Εικόνα 29: Εφαρμογή του JFC

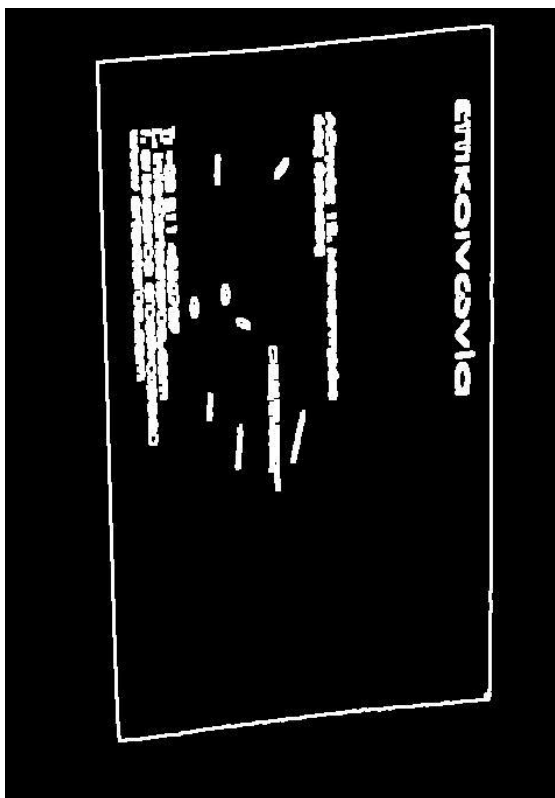


Εικόνα 30: Εφαρμογή του τελεστή



Εικόνα 31: Εφαρμογή του JFC

Step 3: Dilation



Εικόνα 32: Dilation με element size (3,3)



Εικόνα 33: Dilation με element size (5,5)

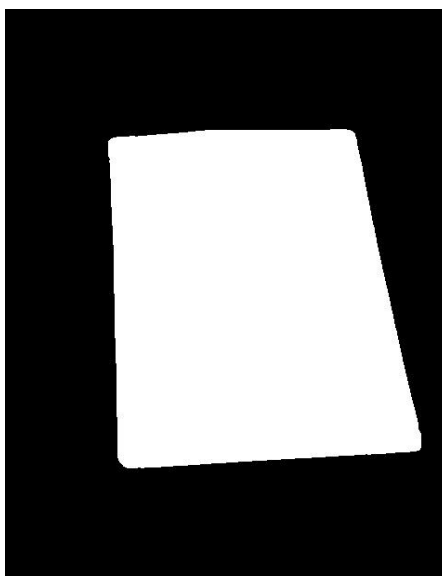


Εικόνα 34: Dilation με element size (3,3)

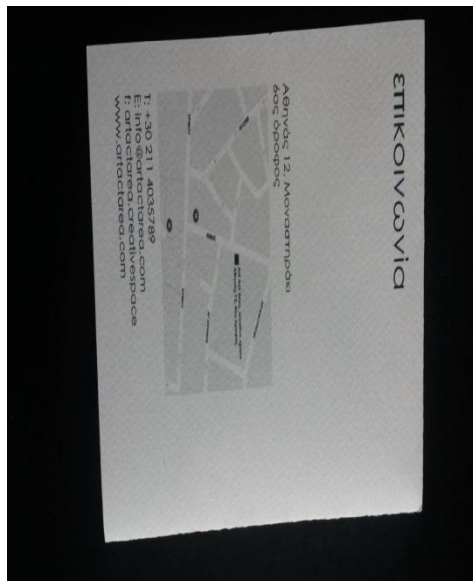


Εικόνα 35: Dilation με element size (5,5)

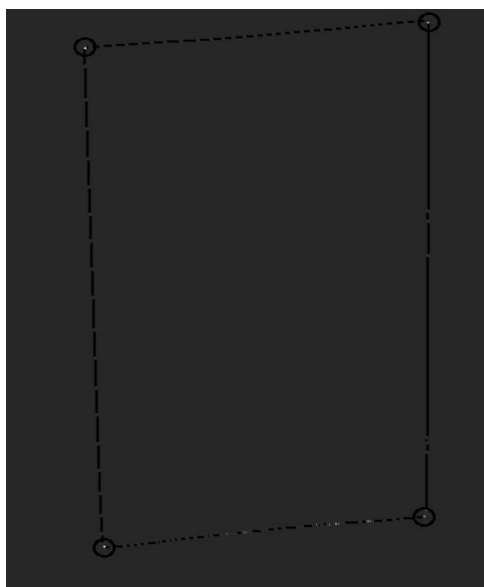
Step 4, 5: Find contours & Harris Corner Detector



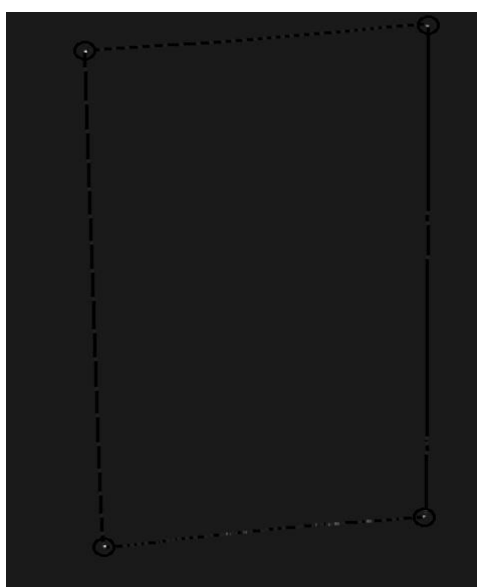
Εικόνα 36: Find Max Contour



Εικόνα 37: Αρχική εικόνα



Εικόνα 38: Harris Corner Detection με Blocksize = 3



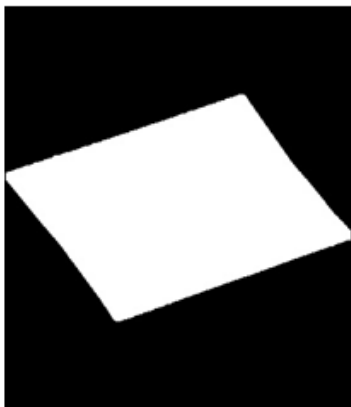
Εικόνα 39: Harris Corner Detection με Blocksize = 4



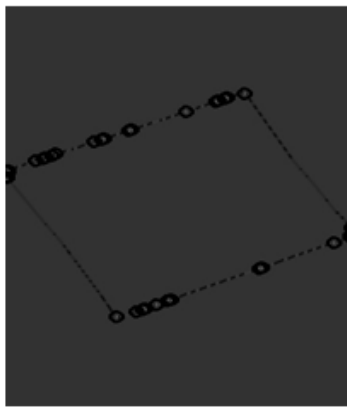
Εικόνα 40: Harorner Detection με Blocksize = 5

Με τη δοκιμή διάφορων τιμών για το μέγεθος του παραθύρου παρατηρείται ότι όταν γίνεται χρήση του τελεστή Sobel η καλύτερη τιμή παραθύρου για ορθότερα αποτελέσματα γωνιών ισούται με τρία. Αντίστοιχα, σε περίπτωση που χρησιμοποιηθεί ο αλγόριθμος Canny η τιμή του παραθύρου δίνει καλύτερα αποτελέσματα όταν ισούται με πέντε. Παρακάτω, παρατηρείται πως αλλάζει η ανίχνευση των γωνιών με τη χρήση διαφορετικού αλγορίθμου ανίχνευσης ακμών.

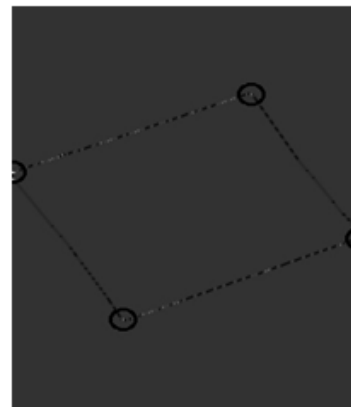
Step 6 : Εύρεση του μέγιστου τετράπλευρου



Εύρεση περιγράμματος



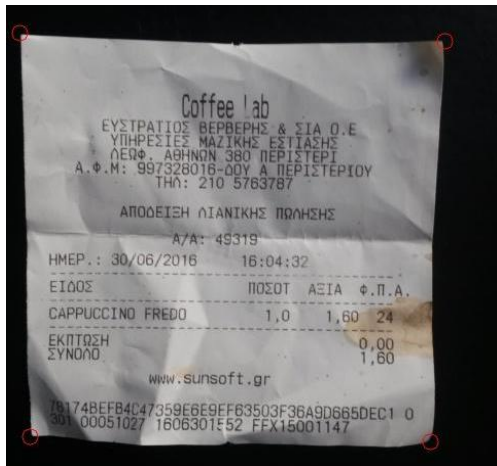
Ανίχνευση γωνιών



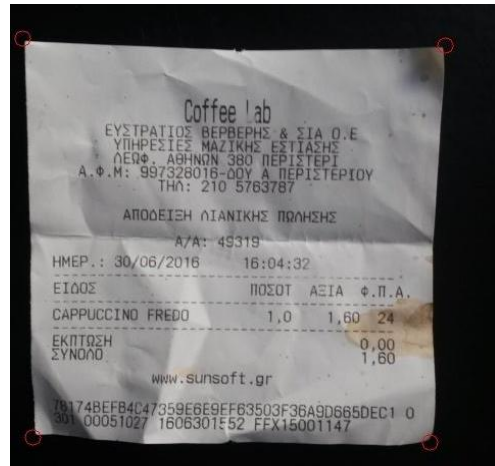
Επιλογή 4 γωνιών

Εικόνα 41: Επιλογή σημείων

Αποτελέσματα του έκτου βήματος



Εικόνα 42: Harris με εφαρμογή του Canny edae detector



Εικόνα 43: Harris με εφαρμογή του Sobel φίλτρου



Εικόνα 44: Harris με εφαρμογή του Canny edge detector



Εικόνα 45: Harris με εφαρμογή του Sobel φίλτρου

Στη δεύτερη στήλη είναι τα αποτελέσματα του αλγορίθμου Harris και Stephens με τη χρήση Sobel φίλτρου. Παρατηρούμε ότι με τη μέθοδο αυτή βρίσκονται λανθασμένες θέσεις για σημεία γωνιών οι οποίες θεωρούνται σωστές όταν γίνεται αποτίμηση της μεθόδου με κάποιο μεγαλύτερο περιθώριο ανοχής. Σε σύγκριση με την πρώτη στήλη, ο Sobel δε βγάζει τόσο καλά αποτελέσματα όσο ο JFC.

4.2 Μέθοδος Hough

Με βάση τη μεθοδολογία που προτάθηκε από τους Ryan Baumann, Christopher Blackwell και W. Brent Seales [11] πραγματοποιήθηκε η υλοποίηση και η εξέταση τρόπων βελτίωσης των αποτελεσμάτων. Στα βήματα που αναφέρθηκαν στο κεφάλαιο 3 παράγραφος 3.2.2, έγιναν κάποιες αλλαγές τα αποτελέσματα των οποίων συγκρίνουμε με τα αποτελέσματα της κοντινότερης στο [11] μεθοδολογίας. Συνοπτικά οι αλλαγές είναι:

- Προσθήκη Smoothing με (Gaussian blur)

- Κβάντιση του πίνακα για τις τιμές ρ και θ
- Αλλαγή της συνάρτησης που επιστρέφει τις ευθείες (αντί για τη συνάρτηση Probabilistic hough που επιστρέφει ευθύγραμμα τμήματα υλοποιήθηκε η κλασική μέθοδος hough και χρησιμοποιήθηκαν ως σημεία για την ομογραφία τα σημεία τομής.
- Αλλαγή της κλασικής συνάρτησης που βρίσκει τις ευθείες με 2 παραλλαγές
- Πρώτη παραλλαγή:
 - Εύρεση ολικού μέγιστου του πίνακα συσσώρευσης
 - Εύρεση των τιμών του ρ και θ
 - Υπολογισμός της ευθείας που του αντιστοιχεί
 - Διαγραφή των γειτονικών στοιχείων σε μικρού μεγέθους γειτονιά που εξαρτάται σε ένα βαθμό από την κβάντιση των ρ και θ .
 - Επανάληψη της διαδικασίας αυτής για την εύρεση μιας ευθείας με κοντινές τιμές του θ στον πίνακα συσσώρευσης με τη χρήση ενός περιθωρίου ανοχής
 - Επανάληψη της διαδικασίας αυτής για την εύρεση μιας ευθείας με κοντινές τιμές του $\theta+90$ στον πίνακα συσσώρευσης με τη χρήση του ίδιου περιθωρίου ανοχής
 - Εύρεση τεσσάρων ευθειών
 - Εύρεση των σημείων τομής
- Δεύτερη παραλλαγή:
 - Εύρεση τοπικού μέγιστου του πίνακα συσσώρευσης με βάση ένα κατώφλι
 - Τιμή κατωφλίου είτε ως παράμετρος είτε συναρτήσει των μεγεθών της εικόνας
 - Εύρεση των τιμών του ρ και θ
 - Υπολογισμός της ευθείας που αντιστοιχεί στο παραπάνω μέγιστο
 - Διαγραφή των γειτονικών στοιχείων σε μικρού μεγέθους γειτονιά που εξαρτάται σε ένα βαθμό από την κβάντιση των ρ και θ .
 - Επανάληψη της διαδικασίας αυτής όλα τα pixel
 - Επιστροφή πολλών ευθειών
 - Έλεγχος ευθειών μέσω συνθηκών καθετότητας και παραλληλίας ώστε να περιοριστούν
 - Εύρεση των σημείων τομής
 - Έλεγχος των σημείων ανά τριάδες ώστε να απορριφθούν τα εξωτερικά και τα εσωτερικά
 - Υπολογισμός μεγαλύτερου τετράπλευρου με βάση τις γωνίες που έχουν μείνει.

Η μεθοδολογία που αποτέλεσε τη βάση, συγκρίθηκε με τις υπόλοιπες που προτάθηκαν και είναι πιο κοντά στην [11] περιγράφεται από τα βήματα του 3.2.2 και τις αλλαγές που προσαρτήθηκαν και αναφέρονται παραπάνω χρησιμοποιώντας τον κλασικό Hough transform για την εξαγωγή των ευθειών. Παρακάτω παρουσιάζονται πιο αναλυτικά πως κινηθήκαμε:

Η μέθοδος που αναφέρεται στο [11] έχει ως σύμβαση την ύπαρξη άνω και κάτω ορίων του εγγράφου, χωρίς να ενδιαφέρεται για τα δεξιά και αριστερά όρια. Κατατάσσει τις ευθείες που ανιχνεύει σε άνω και κάτω ευθείες σύμφωνα με το κέντρο και στη συνέχεια

επιλέγει για δυο ευθείες ένα μέσο όρο της κάθε κατηγορίας. Τα αποτελέσματα που παρουσιάζονται στη μέθοδο αυτή φαίνεται να περιέχουν και background (υπόβαθρο) καθώς για δεξιά και αριστερά όρια επιλέγονται τα όρια της ίδιας της εικόνας. Ένα άλλο αρνητικό που παρουσιάζει είναι στο σημείο που επιλέγονται για δυο ευθείες οι μέσες τιμές των άνω ευθειών και κάτω για κάθε περίπτωση. Αυτό το βήμα δεν παρουσιάζει τα αναμενόμενα αποτελέσματα. Για να δουλέψει απαραίτητη προϋπόθεση είναι όλες οι ευθείες να είναι μεταξύ τους κοντινές και να βρίσκονται στην περιοχή που είναι τα πραγματικά όρια του εγγράφου απορρίπτοντας όλες τις εσωτερικές και εξωτερικές ευθείες που μπορεί να ανιχνεύσει ο αλγόριθμος. Για να επιτευχθεί κάτι τέτοιο πρέπει να γίνει ιδανικό segmentation της εικόνας κάτι που αναφέρθηκε στον προηγούμενο αλγόριθμο και αντιμετωπίστηκε σαν πρόβλημα. Δηλαδή, για να δουλέψει σωστά αυτή η μέθοδος πρέπει να τεθεί και δεύτερος περιορισμός που αφορά το περίγραμμα. (να μπορεί να ανιχνευτεί και να κλείνει).

Η μέθοδος που ακολουθήσαμε στην παρούσα πτυχιακή για την βελτίωση των αποτελεσμάτων στηρίζεται στην κλασική ανίχνευση ευθειών του μετασχηματισμού hough και στοχεύει στην εύρεση όλων των ορίων του εγγράφου για την αποτελεσματική αφαίρεση του περιθωρίου της εικόνας. Με τη χρήση με 2 ζευγών ευθειών κάθετων και παράλληλων προσδιορίζεται το περίγραμμα του εγγράφου και τα τέσσερα σημεία τομείς χρησιμοποιούνται για τον υπολογισμό της homography όπως περιγράφηκε και πιο πάνω. Αυτός ο τρόπος θα αναφέρεται για συντομία και ευκολία ως Classic Hough

Αρχικά, χρησιμοποιείται η Classic Hough η οποία θεωρείται ότι είναι πιο κοντά στην πρόταση [11]. Σε μια εικόνα που έχει γίνει καλή ανίχνευση ακμών οι ευθείες που θα περνάνε από τις ακμές θα σημειώνουν μεγαλύτερη τιμή στον πίνακα συσσώρευσης. Έτσι, μια πρώτη προσέγγιση των πραγματικών ακμών είναι η εύρεση των ευθειών των τεσσάρων μεγίστων τιμών στον accumulator. Κάτι τέτοιο φαίνεται να λειτουργεί σε ιδανικές εικόνες. Οι εικόνες που έχουν τραβηχτεί από κάμερα κινητού έχουν μεγάλα ποσοστά θορύβου με αποτέλεσμα να ανιχνεύονται πολλά σημεία για σημεία ακμών, Τα τέσσερα ολικά μέγιστα δεν φαίνονται να δουλεύουν σε αυτή την περίπτωση καθώς οι τέσσερις ευθείες ανιχνεύονται πολύ κοντά μεταξύ τους, Ένας τρόπος για να βελτιωθεί η παραπάνω περίπτωση είναι η χρήση του local maxima suppression. Η ανίχνευση γίνεται σε μια γειτονιά από την οποία προκύπτει ένα τοπικό μέγιστο και έτσι εντοπίζονται τέσσερα τοπικά μέγιστα.

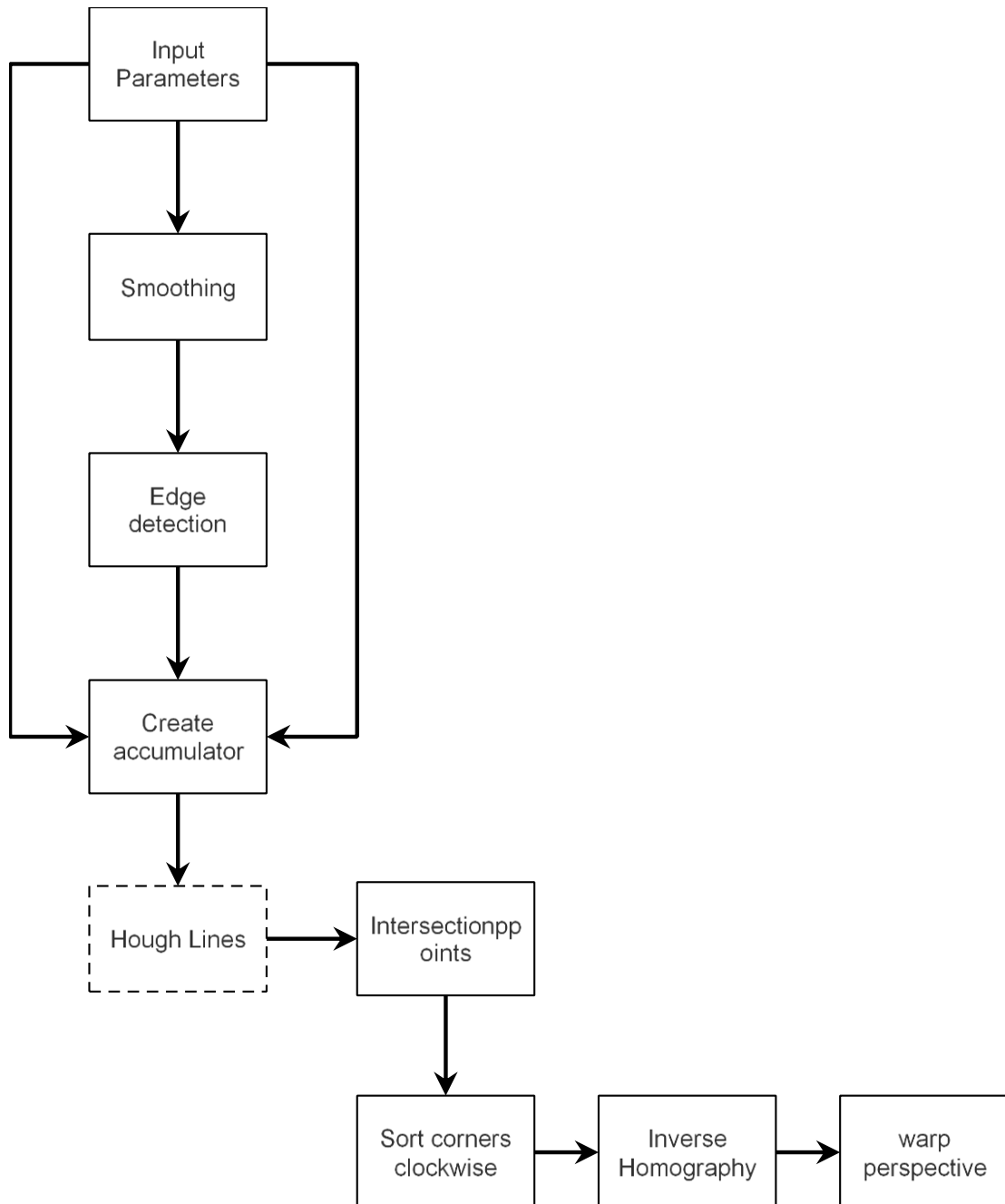
Ο accumulator έχει στον οριζόντιο άξονα τις τιμές των θ και στον κάθετο τις τιμές των ρ . Αρχικά η συμπλήρωση του πίνακα γινόταν ανά ένα pixel και ανά μια μοίρα. Στη συνέχεια, για καλύτερη απόδοση έγινε κβάντιση του πίνακα ως προς τα θ και ρ και χρησιμοποιήθηκαν κάποιες μεταβλητές που καθορίζουν πόσες τιμές θ και ρ αντιστοιχούν σε κάθε κελί του πίνακα.

Η χρήση των τοπικών μεγίστων φαίνεται να βελτιώνει τα αποτελέσματα όμως δεν παύει να επιλέγει λανθασμένες ευθείες για ακμές. Μια καλύτερη προσέγγιση είναι η προσθήκη περιορισμών για την καλύτερη ανίχνευση ευθειών. Αρχικά, υπολογίζεται το ολικό μέγιστο στον πίνακα συσσώρευσης και με βάση τις τιμές που έχει για τις παραμέτρους θ και ρ

γίνεται μια αναζήτηση στον πίνακα συσσώρευσης ευθειών που είναι παράλληλες (με κάποιο περιθώριο ανοχής) στην ευθεία που αντιπροσωπεύει το μέγιστο κελί. Οι ευθείες αναζητούνται ανά γειτονιές όπως περιεγράφηκε και πιο πάνω και επιλέγεται η ευθεία με τη μέγιστη τιμή. Η ίδια διαδικασία εφαρμόζεται για την επιλογή δυο κάθετων ευθειών με το ίδιο περιθώριο ανοχής.

Η παραπάνω μέθοδος παρουσίασε το μεγαλύτερο ποσοστό εγκυρότητας σε αποτελέσματα. Για λόγους συντομίας θα αναφέρεται ως Hough Vertical Parallel (Hough_VP). Ωστόσο, παρουσιάζει κάποιες λανθασμένες εκτιμήσεις και απορρίπτει κάποιες εξωτερικές ευθείες. Γι' αυτό χρησιμοποιήθηκε και μια τρίτη παραλλαγή σύμφωνα με την οποία επιλέγονται περισσότερες ευθείες και θα αναφέρεται ως Hough_Threshold (Hough_Th). Για να μην απορριφθούν ευθείες θέτεται ένα όριο (κατώφλι) για την επιλογή τους ανά γειτονιές και γίνεται διαλογή των σημείων μετά την εύρεση τους από τις τομές των ευθειών. Με περιορισμούς που θέτονται επιλέγονται τα σημεία που σχηματίζουν το μέγιστο τετράπλευρο απορρίπτοντας τα εσωτερικά σημεία. Η παραλλαγή αυτή φαίνεται να συμπληρώνει την προηγούμενη παραλλαγή καθώς καλύπτει κάποιες περιπτώσεις λαθών που εμφανίζει. Ωστόσο, δεν παρουσιάζει καλύτερα ποσοστά εγκυρότητας από την Hough_VP καθώς απορρίπτει πολλά σημεία και εμφανίζει περιπτώσεις που δεν βρίσκει σημεία.

Παρακάτω φαίνεται το διάγραμμα ροής της μεθόδου (Σχήμα 2). Με διακεκομμένες αναπαρίσταται το μεταβλητό κομμάτι των παραλλαγών. Στη θέση των Hough lines μπαίνει μια από τις Classic Hough, Hough_VP, Hough_Th.



Σχήμα 2: Διάγραμμα ροής μεθόδου Baumann- Blackwell - Brent Seales [11].

Hough Transform

Step 1: Εξομάλυνση εικόνας με τη χρήση Gaussian Blur.

Step 2 : Χρήση ενός αλγορίθμου ανίχνευσης ακμών στην εξομαλυμένη grayscale εικόνα. Για το edge detection γίνεται χρήση του αλγορίθμου Canny .

Step 3 : Χρήση του μορφολογικού τελεστή Dilation με στοιχείο το RECT και Kernel size ισούται με τρία.

Step 4 : Δημιουργία του accumulator matrix για κβαντισμένες τιμές του ρ και του θ . Αύξηση κάθε κελιού του πίνακα κατά 1 αν η ευθεία που αντιπροσωπεύει περνάει από ένα pixel της εικόνας. Η διαδικασία αυτή γίνεται για κάθε pixel και συμπληρώνεται ο πίνακας συσσώρευσης.

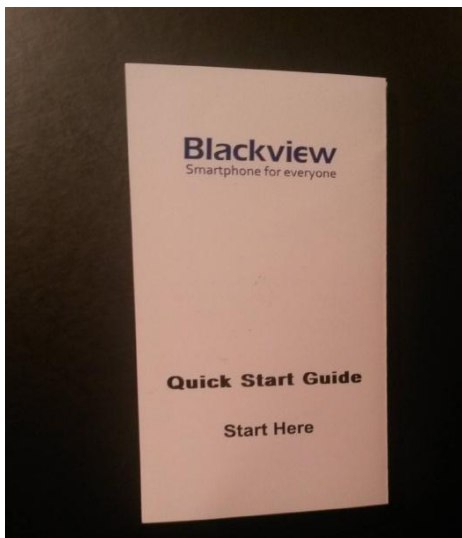
Step 5 : Επιλογή τεσσάρων ή περισσότερων ευθειών που παρουσιάζουν μέγιστες τιμές στον πίνακα συσσώρευσης, υποθέτοντας πως με καλή ανίχνευση των ακμών , υψηλότερες τιμές στον accumulator matrix θα εμφανίζουν ευθείες που περνάνε από τις πραγματικές ακμές.

Step 6 : Εύρεση των σημείων τομής των τεσσάρων ευθειών. Σε περίπτωση που έχει γίνει επιλογή μεγαλύτερου πλήθους ευθειών γίνεται έλεγχος και απορρίπτονται κάποιες ευθείες και σε επόμενο βήμα γίνεται η επιλογή των σημείων.

Step 7 : Εφαρμογή της homography

Παρακάτω παρουσιάζονται τα αποτελέσματα από την υλοποίηση της μεθόδου:

Step 2: Edge Detection (Canny)

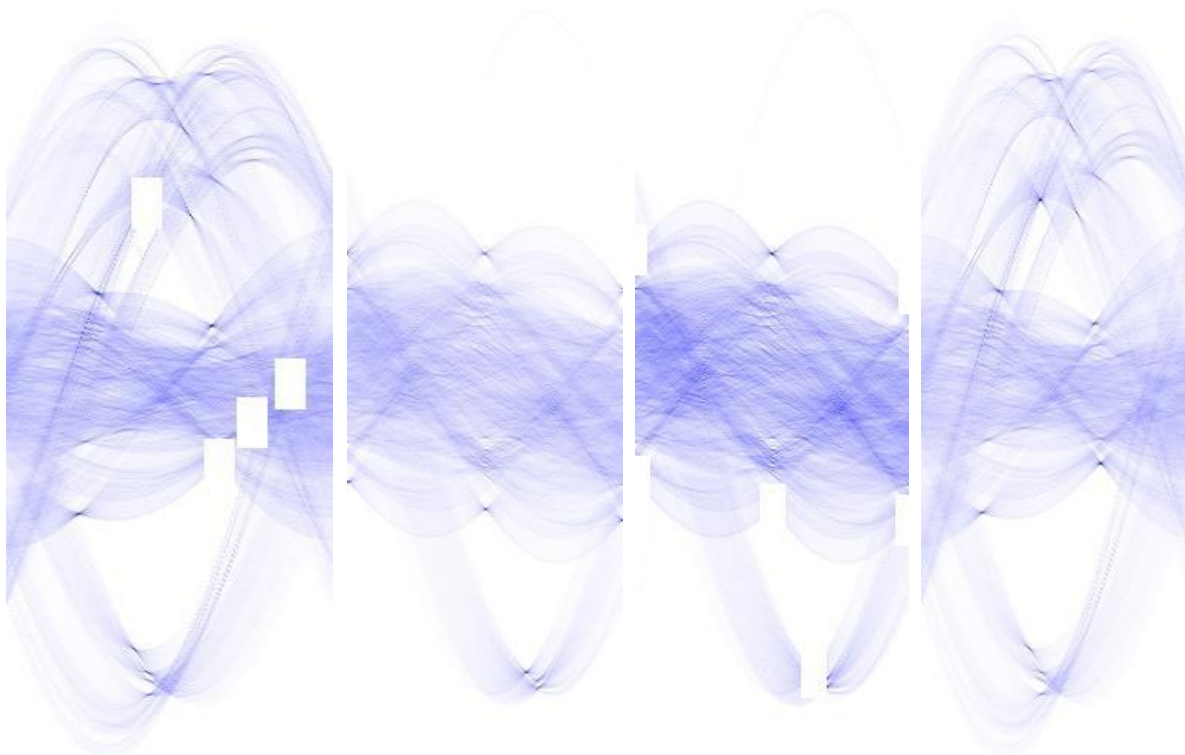


Εικόνα 46: Αρχική εικόνα με παραμόρφωση προοπτικής



Εικόνα 47: Ανίχνευση ακμών με τον αλγόριθμο JF Canny

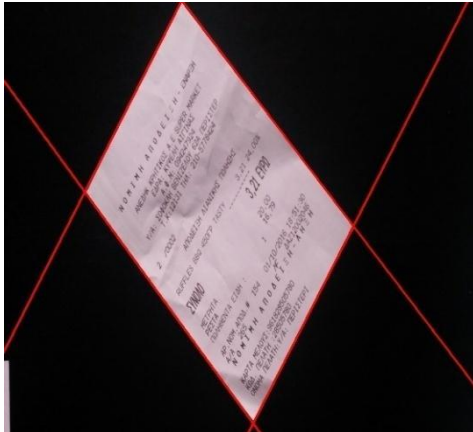
Step 4: Create accumulator matrix



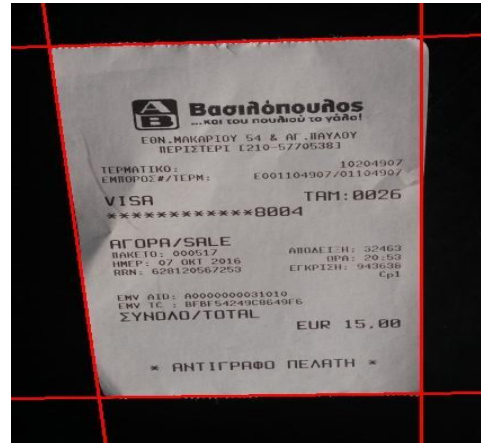
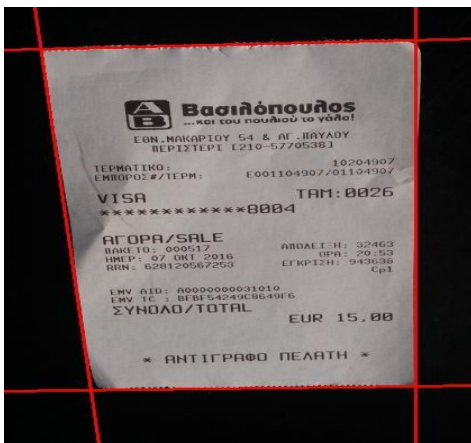
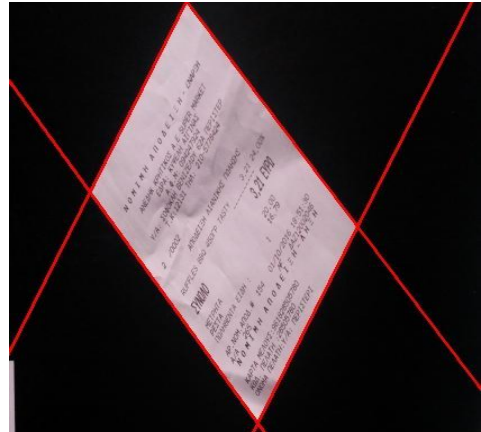
Εικόνα 48: Accumulator των εικόνων 31, 32, 33, 34

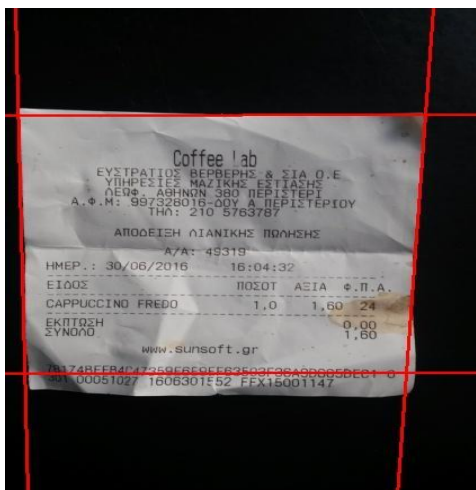
Step 5: Get Lines

Hough_VP



Hough_Th





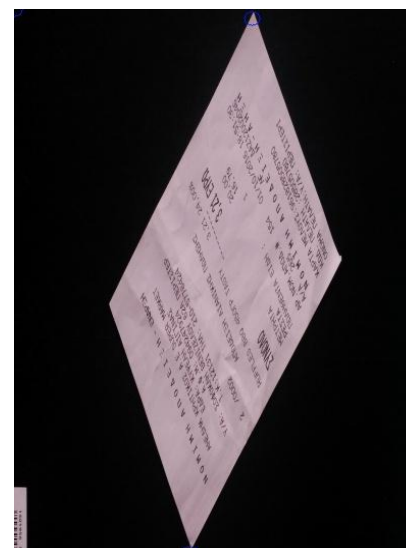
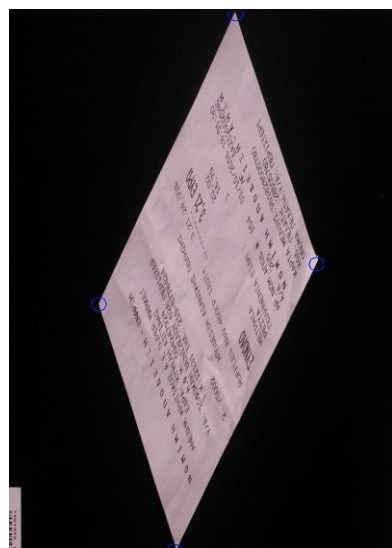
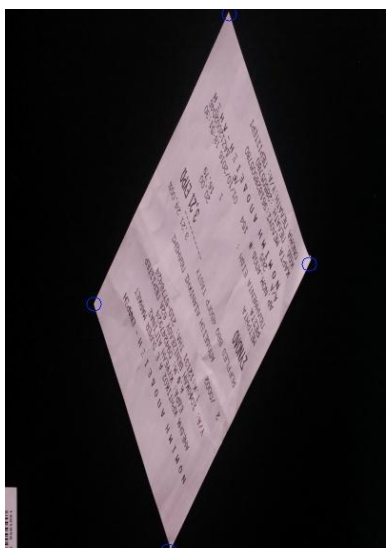
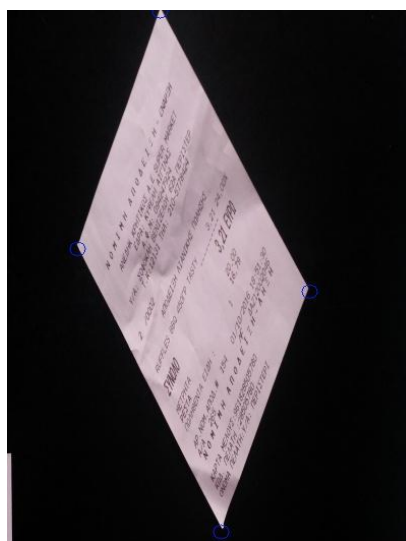
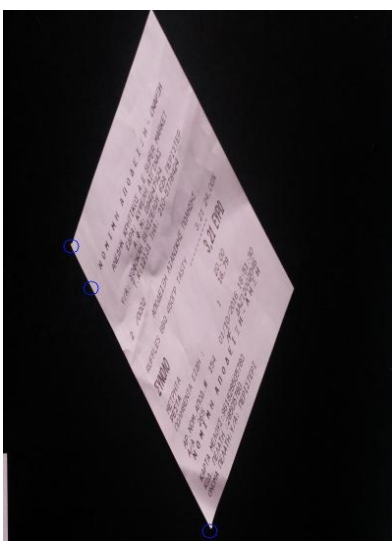
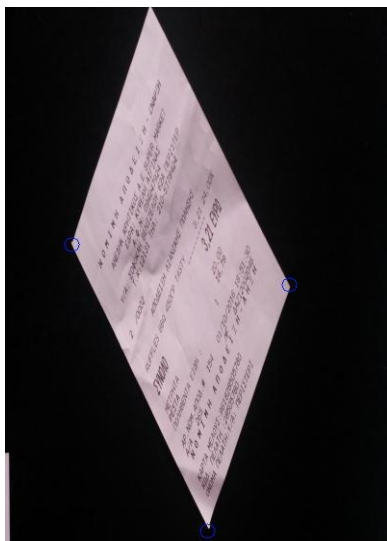
Εικόνα 49: Η δεξιά στήλη εικόνων παρουσιάζει την εύρεση ευθειών με τη χρήση τεσσάρων τοπικών μεγίστων (μέθοδος 2) και η αριστερή στήλη απεικονίζει την επιλογή ευθειών με τη χρήση της κατωφλίωσης.

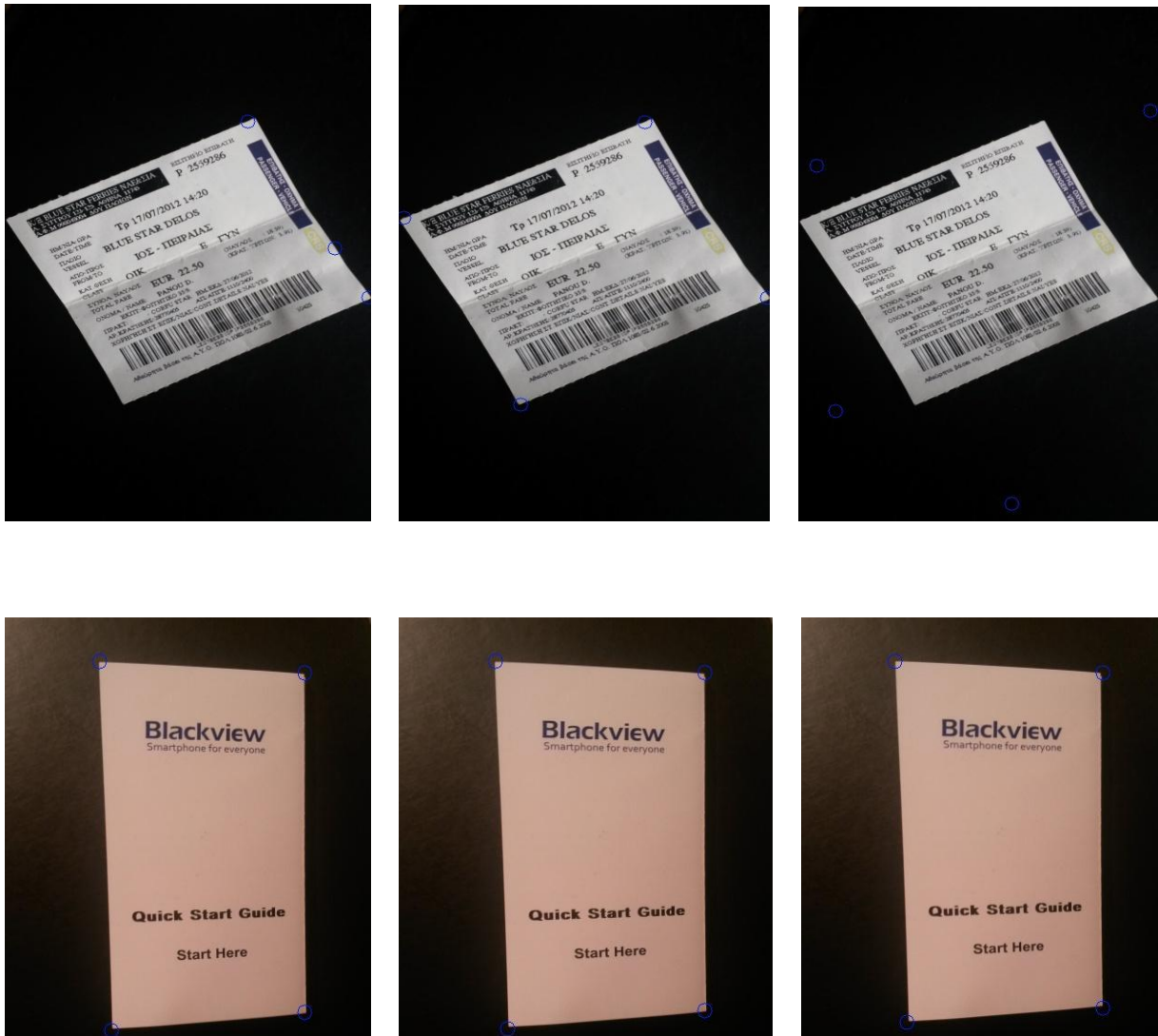
Σύγκριση αποτελεσμάτων των μεθόδων

Classic Hough

Hough_VP

Hough_Th



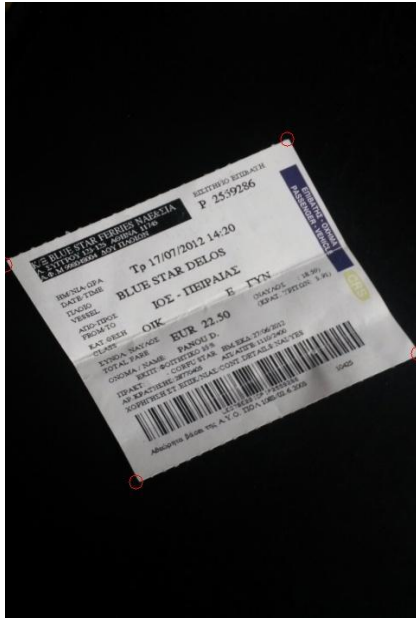


Εικόνα 50: Σύγκριση μεθόδων ανίχνευσης ευθειών. Τελικό Στάδιο -ανίχνευσης γωνιών. Η πρώτη στήλη παρουσιάζει τη Classic Hough η δεύτερη τη Hough_Vertical_Parallel και η τρίτη Hough_Threshold 3 στο τελικό στάδιο πριν την εφαρμογή ομογραφίας.

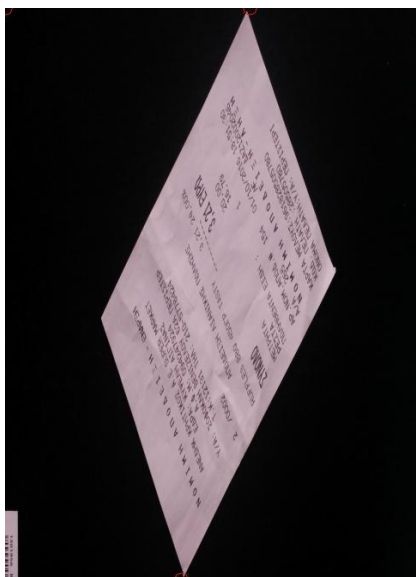
4.3 Homography

Τελευταίο βήμα αποτελεί ο υπολογισμός και η εφαρμογή του πίνακα της ομογραφίας. Για την επίλυση του πίνακα και την εύρεση των εννέα στοιχείων που τον απαρτίζουν χρησιμοποιείται η μέθοδος Gaussian elimination ή αλλιώς Gauss–Jordan elimination. Αποτέλεσμα αυτής είναι η εύρεση του διαγώνιου πίνακα, επομένως και λύση του συστήματος. Σε επόμενο βήμα βρίσκεται ο αντίστροφος πίνακας της ομογραφίας και με βάση αυτού συμπληρώνονται οι τιμές των πεδίων του τελικού πίνακα-εικόνας. Γνωρίζοντας τις θέσεις των ρixel που θα έχει ο τελικός πίνακας χρησιμοποιούμε τους ίδιους τύπου από Homography και κινούμαστε αντίστροφα. Για την συμπλήρωση του κάθε ρixel της τελικής εικόνας χρησιμοποιούνται οι τύποι (22) - (24) για την εύρεση των θέσεων της αρχικής εικόνας και στη συνέχεια αντιγράφεται η του ρixel της θέσης που βρήκαμε στην θέση που αναζητάμε στην τελική εικόνα. Παρακάτω βλέπουμε μερικά αποτελέσματα από αυτό το βήμα.

Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφα από κάμερα



Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφα από κάμερα





Εικόνα 51: Στάδιο διόρθωσης της προοπτικής παραμόρφωσης με την εφαρμογή προβολικού μετασχηματισμού (ομογραφίας). Στη δεξιά στήλη βρίσκονται οι αρχικές παραμορφωμένες εικόνες ενώ αριστερά οι τελικές

5. ΠΕΙΡΑΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ

5.1 Evaluation

Το κομμάτι αυτό είναι το πιο σημαντικό για την κατανόηση της εγκυρότητας των μεθόδων και των αποτελεσμάτων τους. Για την αποτίμηση των παραπάνω μεθόδων χρησιμοποιήθηκαν τα πραγματικά αποτελέσματα σύμφωνα με την εκτίμηση του χρήστη. Η σύγκριση των αποτελεσμάτων γίνεται ένα βήμα πριν την εκτέλεση του αλγορίθμου που διορθώνει την προοπτική (homography). Δημιουργήθηκε λοιπόν ένα αρχείο με τα πραγματικά σημεία των γωνιών για το dataset.

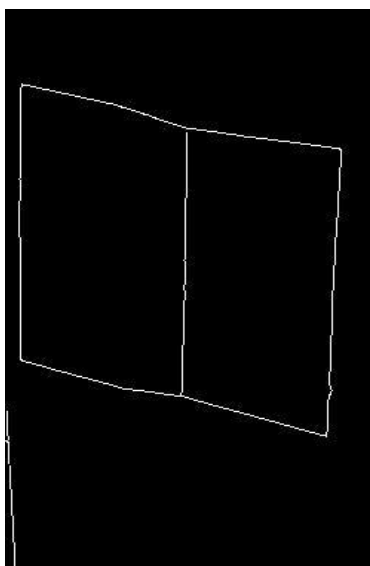
Για κάθε εικόνα, διαβάζεται η τετράδα των σημείων που χαρακτηρίζει το περίγραμμα του εγγράφου που εικονίζει και συγκρίνεται με τα αποτελέσματα των μεθόδων. Κάθε πραγματικό σημείο συγκρίνεται, χρησιμοποιώντας την Ευκλείδεια απόσταση, με ένα προς ένα τα σημεία που έχουν εξάγει οι μέθοδοι που μελετήθηκαν. Τα αποτελέσματα είναι hit or miss. Για να θεωρηθεί έγκυρο ένα αποτέλεσμα, απαραίτητη προϋπόθεση είναι να έχουν βρεθεί και τα τέσσερα σημεία του περιγράμματος. Επίσης, θα πρέπει να βρίσκονται σε απόσταση εντός μιας ακτίνας. Η ακτίνα αυτή ισοδυναμεί με το περιθώριο λάθους το οποίο είναι ένα ποσοστό της διαγωνίου που ορίζουν τα πραγματικά σημεία.

Το ποσοστό αυτό παίρνει δοκιμαστικές τιμές 5% έως 20%.

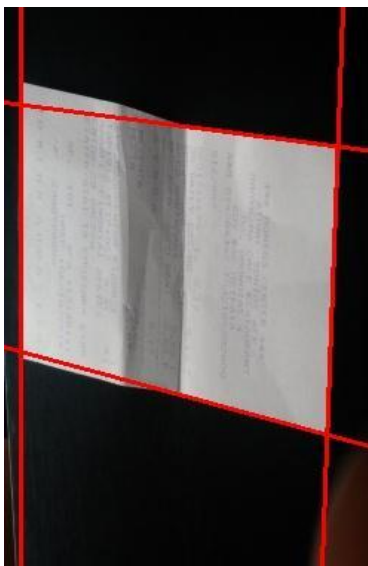
Παρακάτω παρουσιάζονται κάποια από τα αποτελέσματα για κάθε ποσοστό.

Ποσοστό 5%

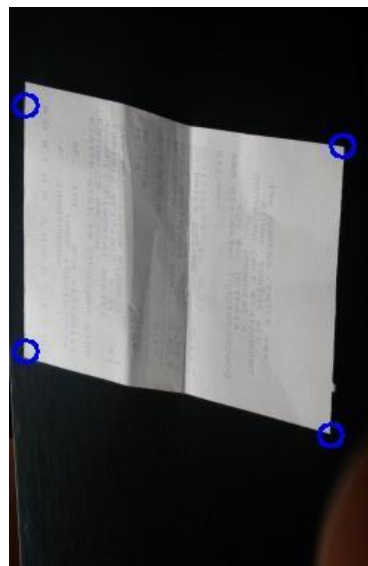
Το ποσοστό αυτό αρχικά εκτιμήθηκε ότι θα δίνει έγκυρα αποτελέσματα. Ωστόσο, διαπιστώθηκε πως απέριπτε σύνολα σημείων που βρίσκονταν πολύ κοντά στα πραγματικά σημεία. Παρακάτω βλέπουμε κάποιες κατηγορίες περιπτώσεων που δε λειτούργησε καλά αυτό το ποσοστό.



Εικόνα 52: Ανίχνευση ακμών



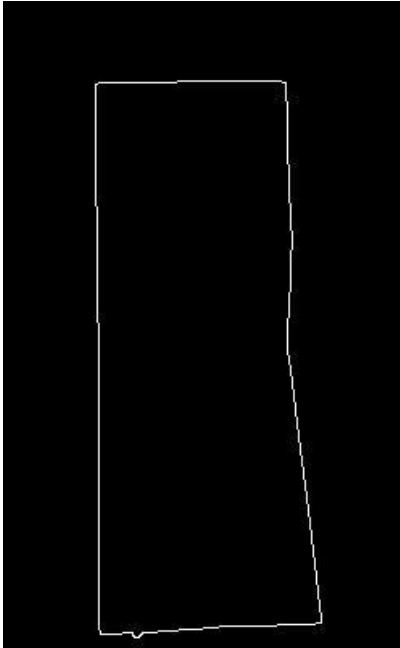
Εικόνα 53: Ανίχνευση ευθειών



Εικόνα 54: Εύρεση τεσσάρων γωνιών

Ποσοστό 10%

Το ποσοστό αυτό είναι το ιδανικό για να καλύψει τα λάθη της παραπάνω περίπτωσης και φαίνεται να είναι πιο κοντά στα πραγματικά αποτελέσματα με το περιθώριο ανοχής του. Παρόλα αυτά, δε λειτουργεί τόσο καλά όταν το έγγραφο χωρίζεται και τα τμήματά του έχουν διαφορετική κλίση όπως φαίνονται στην παρακάτω εικόνα. Σε τέτοιες περιπτώσεις αδυνατεί η ίδια ευθεία να προσεγγίσει τις πραγματικές ακμές. Γι' αυτό η μόνη λύση είναι να μεγαλώσει το περιθώριο λάθους ώστε να υπάρχει μεγαλύτερη ανοχή στα αποτελέσματα.



Εικόνα 55: Ανίχνευση ακμών



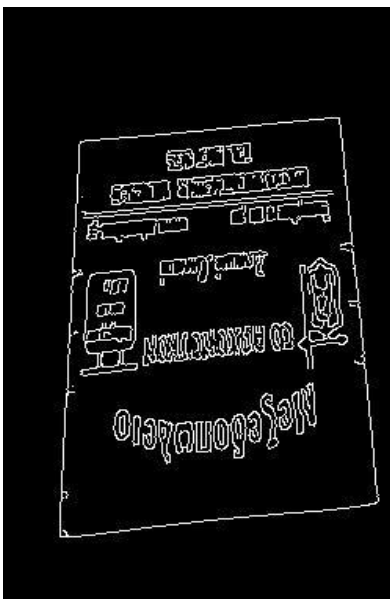
Εικόνα 56: Ανίχνευση ευθειών



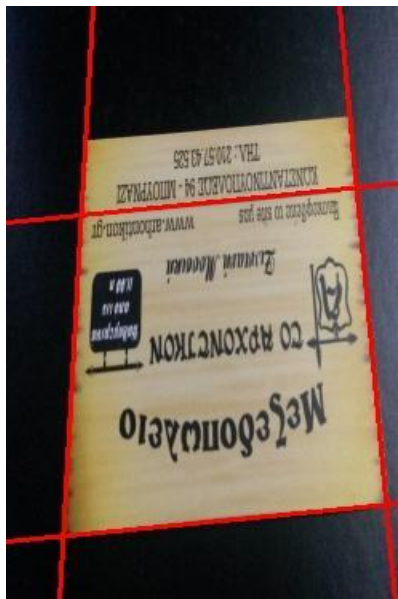
Εικόνα 57: Εύρεση γωνιών

Ποσοστό 20%

Το ποσοστό αυτό χρησιμοποιήθηκε ώστε να επιλύσει το παραπάνω πρόβλημα που προκύπτει από την απόρριψη έγκυρων αποτελεσμάτων όπως παρατηρούμε στην εικόνα 1. Ωστόσο, είναι αρκετά μεγάλο ποσοστό ανοχής με αποτέλεσμα να θεωρούνται έγκυρα κάποια αποτελέσματα που έχουν χάσει κομμάτια της πληροφορίας τους και δεν έχουν τις ίδιες αναλογίες που θεωρούμε με τα αρχικά εκτιμώμενα σημεία. Ένα λανθασμένο αποτέλεσμα που θα θεωρηθεί σωστό σύμφωνα με την περίπτωση που αναφέρθηκε παρατηρείται στις παρακάτω εικόνες (Εικόνες 58,59,60)



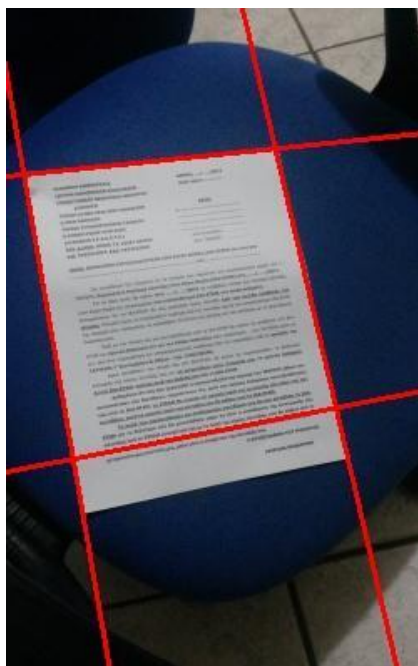
Εικόνα 58: Ανίχνευση ακμών



Εικόνα 59: Ανίχνευση ευθειών



Εικόνα 60: Εύρεση γωνιών



Παρακάτω βλέπουμε τα αποτελέσματα της αποτίμησης για όλες τις μεθόδους:

Για την μέθοδο Hough παράμετροι που τις ελέγχει ο χρήστης είναι :

- rho scaling : Παράμετρος που καθορίζει την κβάντιση του πίνακα συσσώρευσης ως προς το ρ . Πειραματικές τιμές το διάστημα [1,2]
- theta scaling : Παράμετρος που καθορίζει την κβάντιση του πίνακα συσσώρευσης ως προς τη γωνία θ . Ενδεικτικές τιμές στο διάστημα [1/2,3/2]
- Method flag : Παράμετρος που καθορίζει τη συνάρτηση που θα επιλεγεί για την εύρεση των hough lines.

Classic Hough -> 1 Hough_VP -> 2 Hough_Th -> 3

Στη μέθοδο Harris με κάθε κλήση του προγράμματος ο χρήστης μπορεί να καθορίσει τις τιμές των παρακάτω παραμέτρων:

- blur : Παράμετρος που καθορίζει τη συνάρτηση που θα επιλεγεί για την εξομάλυνση της εικόνας. Λαμβάνει την τιμή 1 για Gaussian blur και 2 για Median blur.
- dilation: Παράμετρος που καθορίζει το μέγεθος του δομικού στοιχείου : Ενδεικτικές τιμές (3,3) και (5,5)
- method flag : Παράμετρος που καθορίζει τη συνάρτηση που θα επιλεγεί για την εύρεση των ακμών. Χρησιμοποιείται η μέθοδος Canny και ο Sobel operator για τις οποίες παίρνει τιμές 1 ή 2 αντίστοιχα
- blocksize : Παράμετρος της συνάρτησης harris_corners η οποία καθορίζει το μέγεθος του παραθύρου που ελέγχεται η ένταση των pixel.

Στο εξεταζόμενο dataset δοκιμάστηκαν αρκετοί συνδυασμοί από τιμές παραμέτρων και μεθόδους και σύμφωνα με τα αποτελέσματα καταλήξαμε στις παραπάνω ενδεικτικές τιμές

που αναγράφονται στην κάθε παράμετρο. Παρακάτω παρουσιάζονται τα αποτελέσματα από το evaluation για κάθε μέθοδο

Πίνακας 2: Παράμετροι προγράμματος

Parameters	Actions/Values		
flag_method	Hough Classical flag value = 1 default option	Hough_VP flag value = 2	Hough_Th flag value = 3
flag_homography	Flag value = true Compute Homography		
Homography (flag)	Case $ H = 1$ flag value = 0	Case H33 = 1 flag value = 1	
rho_scaling	Translation of neighbor lines Height of accumulator matrix		
theta_scaling	Step of angle θ Width of accumulator matrix		
flag_edge_detection	Edge detection Method Sobel operator + Dilation flag value = 1 : default	Edge detection Method Canny flag value = 0	
flag_dilation	Kernel size (3,3) flag value = 1 : default	Kernel size (5,5) flag value = 1	
flag_blur	GaussianBlur flag value = 1 : default	Mean blur flag value = 2	
flag_print	Show method in steps		
Thresh	threshold value \rightarrow [0 -200]		

Hough Transform

Πίνακας 3: Αποτελέσματα εκτέλεσης των διαφορετικών συνδυασμών τιμών παραμέτρων των μεθόδων Hough Transform

Method	rho	theta	Name	evaluation error 5%	evaluation error 10%	evaluation error 20%
	1	0.5	method_1_rho_1.0_theta_0.5	52.00%	57.00%	60.00%
	1	1	method_1_rho_1.0_theta_1.0	51.00%	53.00%	57.00%
	1	1.5	method_1_rho_1.0_theta_1.5	52.00%	57.00%	60.00%

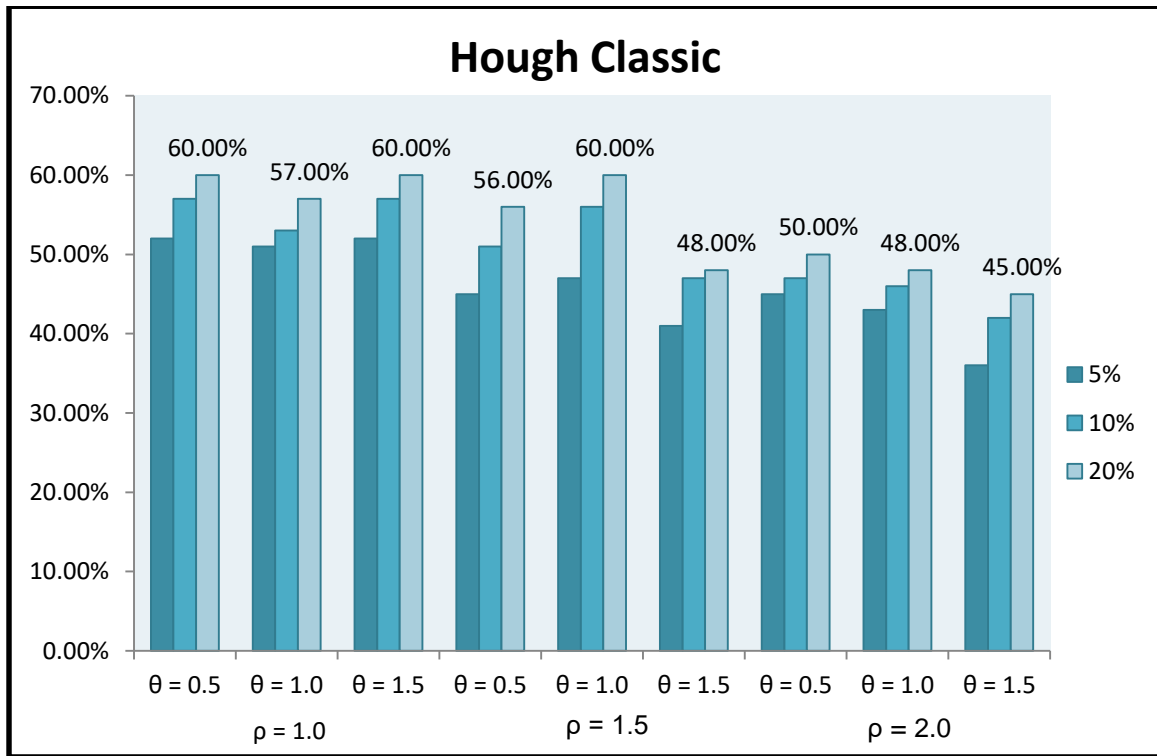
	1.5	0.5	method_1_rho_1.5_theta_0.5	45.00%	51.00%	56.00%
	1.5	1	method_1_rho_1.5_theta_1.0	47.00%	56.00%	60.00%
Classic Hough	1.5	1.5	method_1_rho_1.5_theta_1.5	41.00%	47.00%	48.00%
	2	0.5	method_1_rho_2.0_theta_0.5	45.00%	47.00%	50.00%
	2	1	method_1_rho_2.0_theta_1.0	43.00%	46.00%	48.00%
	2	1.5	method_1_rho_2.0_theta_1.5	36.00%	42.00%	45.00%
	1	0.5	method_2_rho_1.0_theta_0.5	63.00%	67.00%	70.00%
	1	1	method_2_rho_1.0_theta_1.0	65.00%	68.00%	72.00%
	1	1.5	method_2_rho_1.0_theta_1.5	61.00%	67.00%	72.00%
	1.5	0.5	method_2_rho_1.5_theta_0.5	52.00%	60.00%	67.00%
Hough_VP	1.5	1	method_2_rho_1.5_theta_1.0	58.00%	70.00%	76.00%
	1.5	1.5	method_2_rho_1.5_theta_1.5	56.00%	65.00%	67.00%
	2	0.5	method_2_rho_2.0_theta_0.5	58.00%	63.00%	68.00%
	2	1	method_2_rho_2.0_theta_1.0	62.00%	67.00%	70.00%
	2	1.5	method_2_rho_2.0_theta_1.5	52.00%	58.00%	62.00%
	1	0.5	method_3_rho_1.0_theta_0.5	57.00%	65.00%	66.00%
	1	1	method_3_rho_1.0_theta_1.0	62.00%	68.00%	70.00%
	1	1.5	method_3_rho_1.0_theta_1.5	56.00%	57.00%	58.00%
	1.5	0.5	method_3_rho_1.5_theta_0.5	58.00%	70.00%	70.00%
Hough_Th	1.5	1	method_3_rho_1.5_theta_1.0	57.00%	68.00%	72.00%
	1.5	1.5	method_3_rho_1.5_theta_1.5	58.00%	66.00%	72.00%
	2	0.5	method_3_rho_2.0_theta_0.5	50.00%	61.00%	68.00%

	2	1	method_3_rho_2.0_theta_1.0	51.00%	61.00%	68.00%
	2	1.5	method_3_rho_2.0_theta_1.5	48.00%	62.00%	68.00%

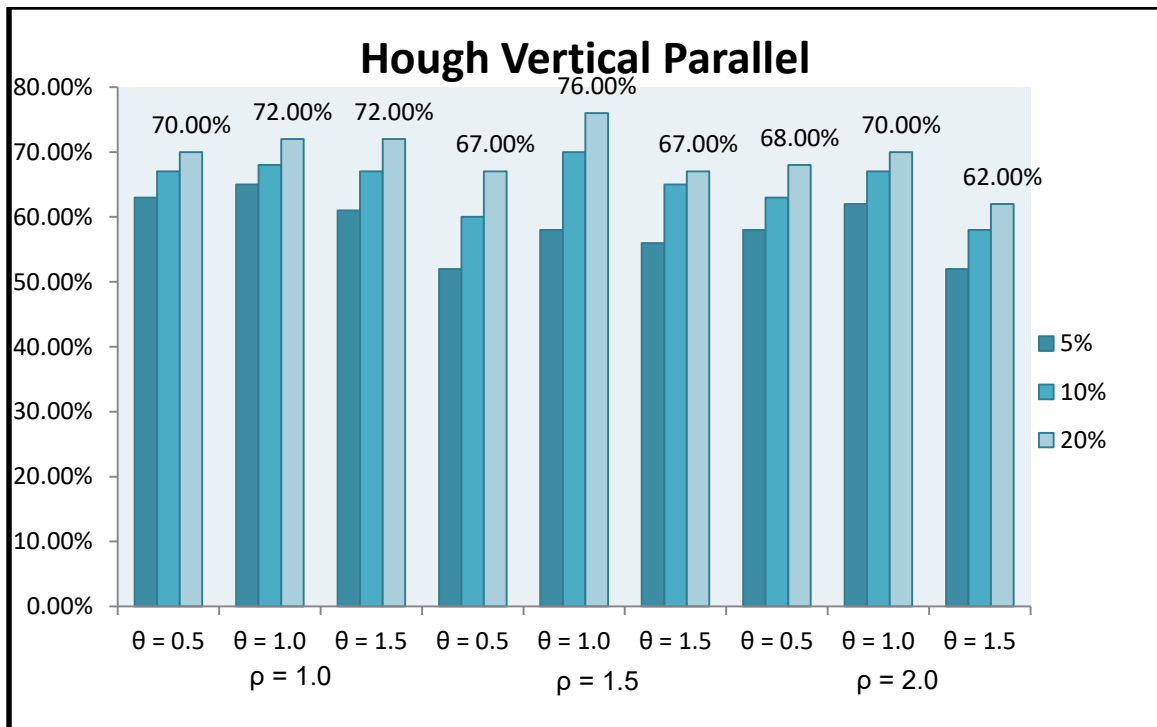
Harris Algorithm

Πίνακας 4: Αποτελέσματα εκτέλεσης των διαφορετικών συνδυασμών τιμών παραμέτρων των μεθόδων Harris

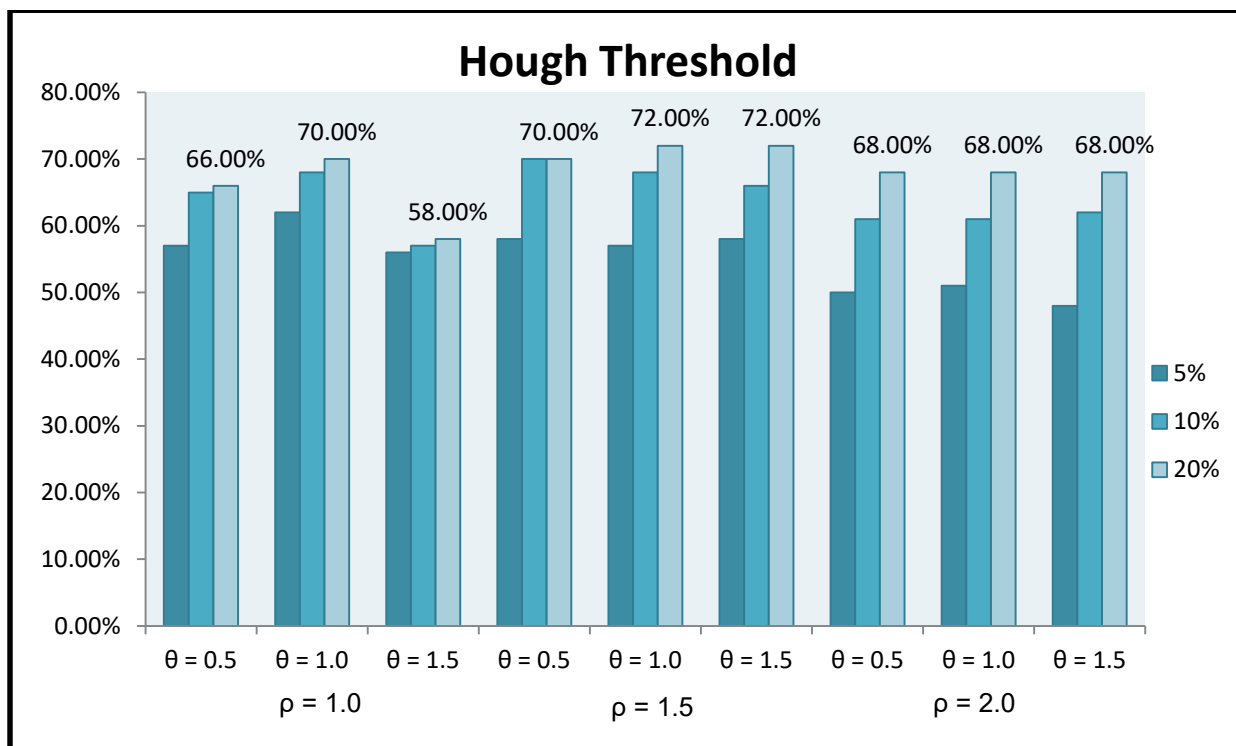
Method	Dilation Kernel Size	Blur	Name	evaluation error 5%	evaluation error 10%	evaluation error 20%
	(3,3)	Mean	corner_1_dilation_1.0_blur	37.00%	41.00%	56.00%
	(3,3)	Gaussian	corner_1_dilation_1.0_gaussian_blur	38.00%	42.00%	57.00%
Canny	(5,5)	Mean	corner_1_dilation_2.0_blur.cout	32.00%	39.00%	48.00%
	(5,5)	Gaussian	corner_1_dilation_2.0_gaussian_blur	33.00%	40.00%	50.00%
	(3,3)	Mean	corner_2_dilation_1.0_blur	31.00%	38.00%	50.00%
	(3,3)	Gaussian	corner_2_dilation_1.0_gaussian_blur	31.00%	38.00%	50.00%
Sobel	(5,5)	Mean	corner_2_dilation_2.0_blur	31.00%	30.00%	46.00%
	(5,5)	Gaussian	corner_2_dilation_2.0_gaussian_blur	31.00%	30.00%	46.00%



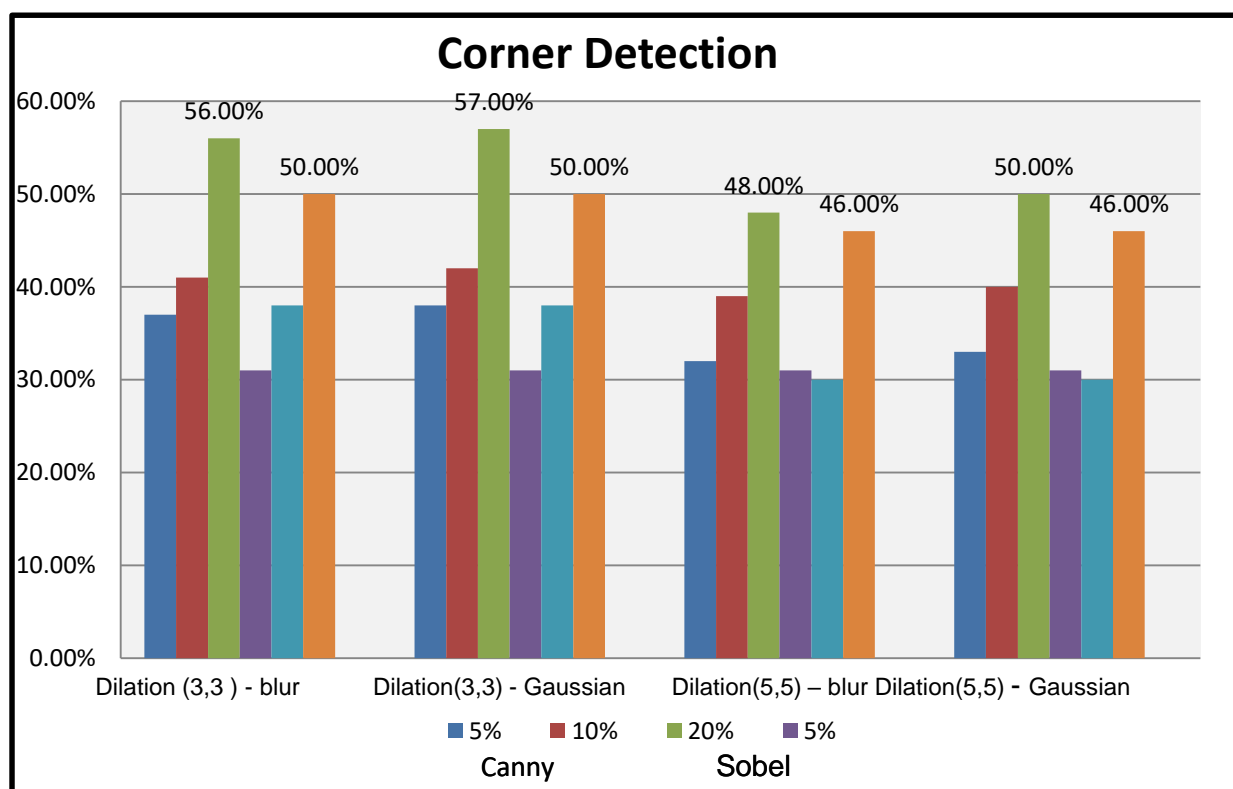
Σχήμα 3 : Διάγραμμα Classic Hough : Ανίχνευση τεσσάρων ολικών μεγίστων σε ποσοστά ανοχής 5-20%



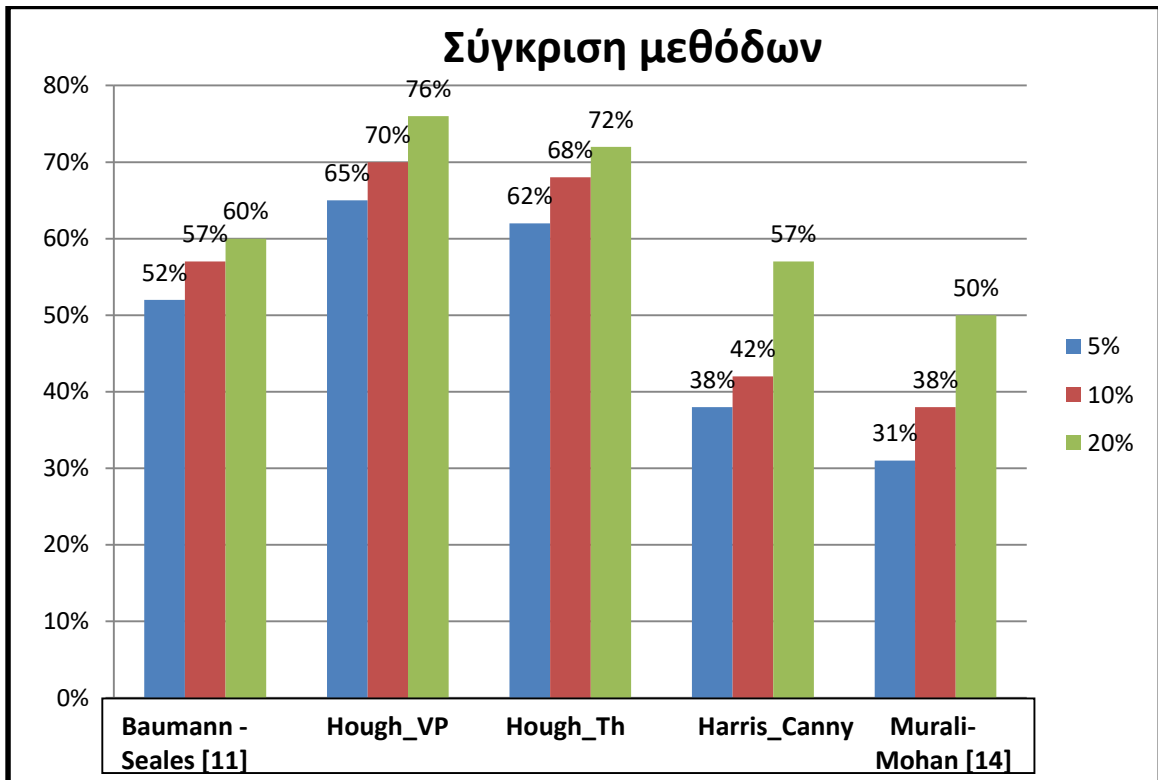
Σχήμα 4: Διάγραμμα Hough_VP: Ανίχνευση τεσσάρων τοπικών μεγίστων σε ποσοστά ανοχής 5-20%



Σχήμα 6: Διάγραμμα Hough_Th : Ανίχνευση με threshold



Σχήμα 5: Διάγραμμα μεθόδων Harris σε ποσοστά ανοχής 5-20%



Σχήμα 7: Παρουσίαση καλύτερων ποσοστών των μεθόδων

Η Hough_VP βγάζει καλύτερα αποτελέσματα σύμφωνα με τα ποσοστά εγκυρότητας σε σχέση με τις άλλες δυο περιπτώσεις. Στη μέθοδο Baumann - Seales [11] οι ευθείες εξαγονται από τον πίνακα συσσώρευσης χρησιμοποιώντας μια τετράδα τοπικών μεγίστων ενώ η Hough_VP είναι παραλλαγή της πρώτης με μόνη διαφορά κάποιους παραπάνω ελέγχους που απαιτούν την εύρεση ακριβώς δυο κάθετων και δυο παράλληλων μεταξύ τους ευθειών. Είναι λογικό να εμφανίζουν τέτοιες διαφορές στα ποσοστά. Η Hough_Th λειτουργεί λίγο διαφορετικά καθώς δέχεται για ευθείες όλες εκείνες που η ψήφος τους στο αντίστοιχο κελί βρίσκεται πάνω από ένα ποσοστό κοντά στο 30% των μεγίστων τιμών, επιλέγοντας έτσι τα τέσσερα σημεία από πολλές ευθείες απορρίπτοντας σημεία και όχι ευθείες. Η Hough_Th δεν βγάζει τις ίδιες λανθασμένες περιπτώσεις με την Hough_VP. Μάλιστα υπερέχει για κάποια δεδομένα έναντι της Hough_VP καθώς δεν απορρίπτει ευθείες αλλά τις επιλέγει όλες πάνω από ένα threshold και κάνοντας τους ελέγχους μετά καταλήγει στα σωστά ενώ η Hough_VP μπορεί να χάνει σε κάποιες περιπτώσεις που επιλέγονται γειτονικές ευθείες παίρνοντας σαν αποτέλεσμα ένα τμήμα του εικονιζόμενου αντικειμένου. Αυτό ίσως λυθεί με αύξηση του παραθύρου της γειτονιάς αλλά αν η γειτονιά είναι πολύ μεγάλη υπάρχει περίπτωση να χαθούν κάποιες ευθείες. Από τη άλλη, η Hough_Th σε κάποιες περιπτώσεις απορρίπτει ευθείες που δεν ικανοποιούν τα κριτήριά της με αποτέλεσμα στην έξοδο να υπάρχουν λιγότερες

Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφο από κάμερα

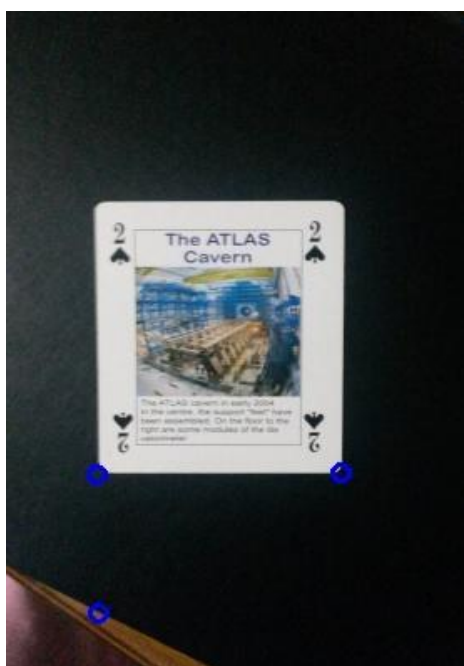
από τέσσερις γωνίες . Σε αντίθεση με τη Classic Hough και Hough_VP , οι οποίες βγάζουν πάντα τέσσερα σημεία σαν έξοδο. Παρακάτω βλέπουμε περιπτώσεις που υπερσχύει η κάθε μέθοδος.



Εικόνα 61: Hough_VP
λανθασμένη ανίχνευση ευθειών



Εικόνα 62: Hough_Th σωστή
ανίχνευση ευθειών



Εικόνα 63: Γωνίες που
προκύπτουν από την Εικόνα 61



Εικόνα 64: Γωνίες που
προκύπτουν από την Εικόνα 62



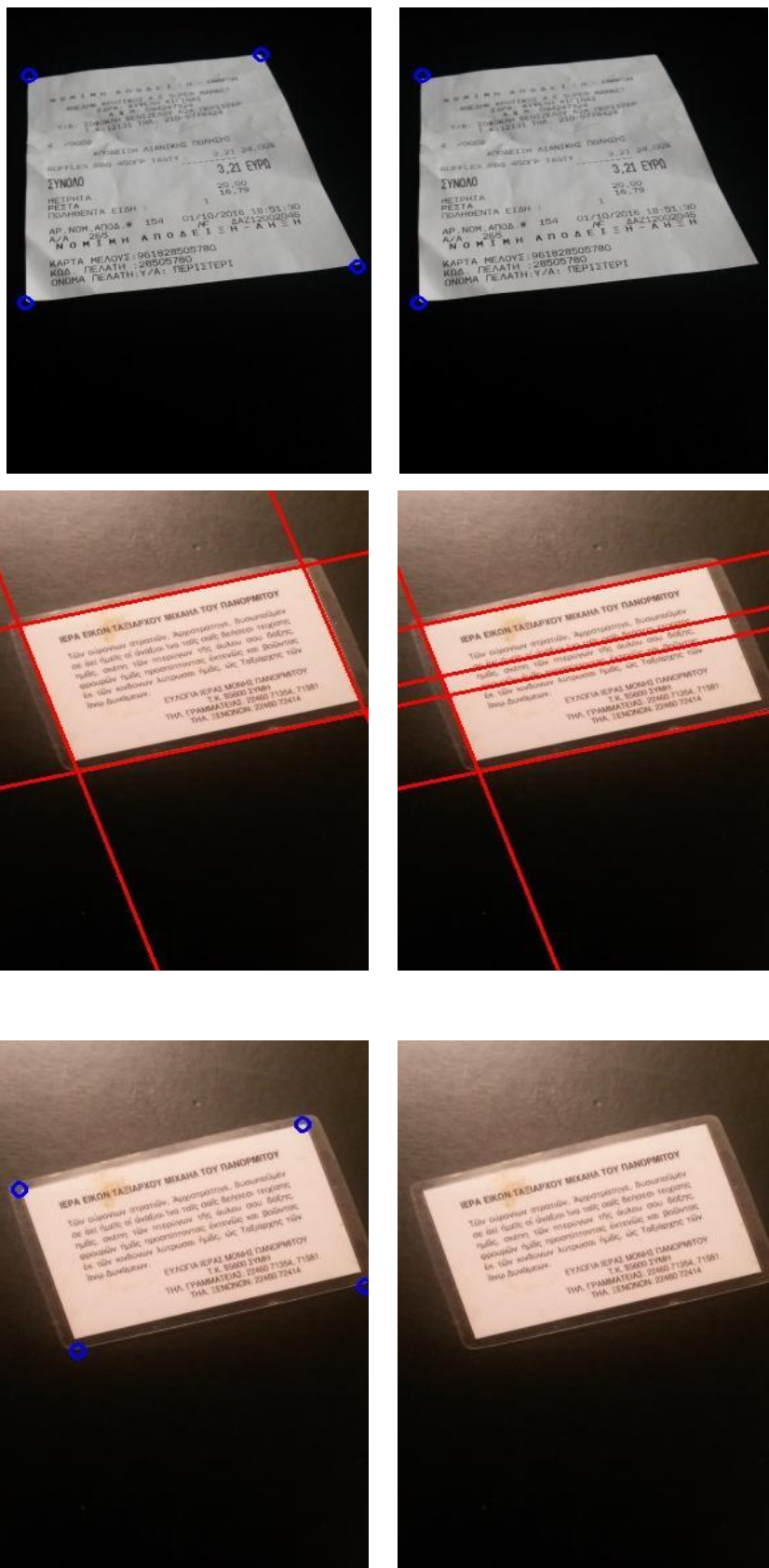
Εικόνα 65: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP, δεξιά το αποτέλεσμα της Hough_Th



Εικόνα 66: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP δεξιά το αποτέλεσμα της Hough_Th



Εικόνα 67: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP, δεξιά το αποτέλεσμα της Hough_Th



Εικόνα 68: Αριστερά το αποτέλεσμα της εφαρμογής της Hough_VP, δεξιά το αποτέλεσμα της Hough_Th

6. ΥΛΟΠΟΙΗΣΗ

Η υλοποίηση μπορεί να χωριστεί σε δυο τμήματα. Το πρώτο κομμάτι, έχει να κάνει με την επεξεργασία εικόνας και τους αλγόριθμους που χρησιμοποιήθηκαν για τη διόρθωση της προοπτικής παραμόρφωσης. Το δεύτερο κομμάτι, περιλαμβάνει τη Web εφαρμογή που ουσιαστικά αποτελείται από το User Interface με το οποίο ο χρήστης θα χρησιμοποιεί τα προγράμματα που αναπτύσσονται στο πρώτο κομμάτι και που υλοποιήθηκαν σύμφωνα με τις μεθόδους [10],[11] . Τα προγράμματα υλοποιήθηκαν σε γλώσσα προγραμματισμού C++ σε συνδυασμό με τη βιβλιοθήκη της C++ OpenCV.

Προϋποθέσεις της OpenCV

- GCC 4.4.x or later
- GNU C++ compiler g++-4.4.x
- CMake 2.8.7 or higher
- Git
- GTK+2.x or higher, including headers (libgtk2.0-dev)
- pkg-config
- [optional] Python 2.6 or later and Numpy 1.5 or later with developer packages (python-dev, python-numpy)
- ffmpeg or libav development packages: libavcodec-dev, libavformat-dev, libswscale-dev
- [optional] libtbb2 libtbb-dev
- [optional] libdc1394 2.x
- [optional] libjpeg-dev, libpng-dev, libtiff-dev, libjasper-dev, libdc1394-22-dev
- [optional] CUDA Toolkit 6.5 or higher

Installation

```
[compiler] sudo apt-get install build-essential
```

```
[required] sudo apt-get install cmake git libgtk2.0-dev pkg-config libavcodec-dev  
libavformat-dev libswscale-dev
```

```
[optional] sudo apt-get install python-dev python-numpy libtbb2 libtbb-dev libjpeg-dev  
libpng-dev libtiff-dev libjasper-dev libdc1394-22-dev
```

Επίσης για να εκτελεστούν τα προγράμματα που αναπτύχθηκαν απαραίτητη είναι η μεταφορά της παρακάτω βιβλιοθήκης στις με την παρακάτω εντολή

```
locate /usr/local/share/OpenCV/3rdparty/lib/libippicv.a  
sudo cp /usr/local/share/OpenCV/3rdparty/lib/libippicv.a /usr/local/lib/
```

6.1 Προγράμματα - Συναρτήσεις

Hough

```
run: ./hough -s <image_Path_Name> -T <threshold> -r <rho_scaling value> -t  
<theta_scaling value> -m <method number> -h <homography flag> -p <show results  
flag> -f <Filename>
```

```
int main(int argc, char** argv)
```

Input :

img_path: ImageName

filename: name of corners output

threshold: threshold Hough lines

rho_scaling, theta_scaling:

print_flag : για την εμφάνιση των ενδιάμεσων αποτελεσμάτων

homography_flag : για την κλήση της συνάρτησης homography

method_flag : για τον τρόπο εξαγωγής των ευθειών

Use:

1. Καλεί τη συνάρτηση doTransform και

2. Κατατάσσει τα σημεία με τη φορά του ρολογιού
3. Ανοίγει ένα αρχείο με όνομα το περιεχόμενο της μεταβλητής filename
4. Γράφει τα σημεία που επιστρέφει η συνάρτηση doTransform αν δεν βρει σημεία ή είναι λιγότερα γράφει στη θέση τους -1

Output: αρχείο filename

vector<cv::Point2d> doTransform(std::string file_path, int threshold)

Input:

file_path: ImageName

threshold: threshold Hough lines

Use:

Βρίσκει τις ευθείες hough για την εικόνα που παίρνει σαν είσοδο και υπολογίζει τα σημεία τομής τους επιστρέφοντας τις γωνίες του περιγράμματος.

1. Καλεί τις συναρτήσεις GaussianBlur και Canny
2. Δημιουργεί ένα αντικείμενο της κλάσης Hough
3. Καλεί τη συνάρτηση Transform της κλάσης Hough με ορίσματα :
 1. τον πίνακα της εικόνας , για την οποία έχουν βρεθεί οι ακμές της στο βήμα 1
 2. τις διαστάσεις της, width και height
 3. τις παραμέτρους από τη γραμμή εντολών, rho_scaling και theta_scaling, για την κβάντιση του πίνακα συσσώρευσης (accumulator) ως προς ρ και θ.
4. Ανάλογα με την τιμή της παραμέτρου method θα κληθεί η αντίστοιχη συνάρτηση που επιστρέφει τις ευθείες (Getlines)
5. Καλεί τη συνάρτηση computeIntersect για την εικόνα με παραμέτρους τις διαστάσεις της και τις ευθείες που υπολογίστηκαν στο βήμα 4.
6. Επιστρέφει τις γωνίες

Output : vector of corners

vector<cv::Point2d> computeIntersect(cv::Mat& img, vector< pair<pair<int, int>, pair<int, int> > >& lines, int h, int w)

Input : hough lines από τη συνάρτηση hough::Transform , διαστάσεις εικόνας, εικόνα

Use: Υπολογίζει τις γωνίες από τις τομές των ευθειών τις ,οποίες παίρνει ως παράμετρο, και τις επιστρέφει στη συνάρτηση doTransform

1. Υπολογίζει τα σημεία τομής των ευθειών
2. Ελέγχει αν οι ευθείες είναι κάθετες
3. Ελέγχει όλες τις περιπτώσεις για τις κλίσεις τους
4. Καλεί τη remove για να περιορίσει τις γωνίες σε τέσσερις
5. Επιστρέφει τα σημεία που είναι έγκυρα για γωνίες

Output: vector of corners

void remove(vector< cv::Point2d >& points, double error, cv::Mat& img)

Input : ένα vector με τις γωνίες που υπολογίστηκαν στην computeIntersect συνάρτηση

Use: Ελέγχει τις αποστάσεις για κάθε τριάδα σημείων .Αφαιρεί τα εσωτερικά σημεία πάνω στις ακμές

Output: -

bool areVertical(int x1, int y1, int x2, int y2, int x3, int y3, int x4, int y4, int precision)

Input : τετράδα σημείων, περιθώριο precision

Use: Ελέγχει τις ευθείες που δημιουργούν τα τέσσερα σημεία αν είναι κάθετες μεταξύ τους βρίσκοντας τα λ τους.

Output: Επιστρέφει true/false

Κλάση Hough

Περιέχει έναν πίνακα συσσώρευσης accumulator, τις διαστάσεις τις εικόνας img_w img_h, τις διαστάσεις του πίνακα accumulator_w , accumulator_h και δυο παραμέτρους κβάντισης του πίνακα , rho_scaling και theta_scaling ,ως προς ρ και θ αντίστοιχα.

Συναρτήσεις της κλάσης Hough

int Transform(unsigned char* img_data, int w, int h , double rs, double ts)

Input : Πίνακας με τα δεδομένα της εικόνας , διαστάσεις εικόνας (w, h), παράμετροι rho_scaling , theta_scaling που καθορίζουν το μέγεθος του accumulator και τη κβάντιση των τιμών των μεταβλητών ρ και θ.

Use: Βρίσκει τις διαστάσεις του πίνακα συσσώρευσης συναρτήσει των παραμέτρων rho_scaling , theta_scaling. Δημιουργεί τον πίνακα συσσώρευσης και υπολογίζει τις τιμές του.

Output : ο accumulator βρίσκεται στην κλάση hough

std::pair< std::pair<int, int>, std::pair<int, int> > magic(int &theta, int & error)

Input : παράμετρο θ και το περιθώριο λάθους

Use:

1. Υπολογίζει την ευθεία που έχει μέγιστη τιμή στον accumulator για το διάστημα $[\theta - \text{error}, \theta + \text{error}]$.
2. Βρίσκει τον τύπο της ευθείας και την υπολογίζει για τις τιμές ρ και θ που εμφανίζεται η μέγιστη τιμή
3. Ορίζει μια γειτονιά γύρω από την τιμή που βρήκε και συμπληρώνει με μηδενικά τις τιμές του πίνακα σε όλη τη γειτονιά

Output : Επιστρέφει την ευθεία

std::vector< std::pair< std::pair<int, int>, std::pair<int, int> > > GetLines(int threshold)

Use: Classic Hough [11]

1. Εύρεση τεσσάρων μεγίστων τιμών στον πίνακα accumulator
2. Εκκαθάριση της γειτονιάς τους
3. Υπολογισμός των ευθειών τους
4. Αποθήκευση κάθε ευθεία σε έναν vector

Output : vector των τεσσάρων ευθειών

std::vector< std::pair< std::pair<int, int>, std::pair<int, int> > > GetLines2(int threshold)

Use: Hough_Vertical_Parallel

1. Εύρεση μιας μέγιστης τιμής στον πίνακα accumulator
2. Εκκαθάριση της γειτονιάς της
3. Για $\theta = \theta_{ολ_Μαξ}$ κλήση της συνάρτησης magic (υπολογισμός μιας παράλληλης)
4. Για $\theta = \theta_{ολ_Μαξ} + 90$ κλήση της συνάρτησης magic (υπολογισμός δυο κάθετων)
5. Αποθήκευση κάθε ευθείας σε έναν vector

Output : vector των τεσσάρων ευθειών

std::vector< std::pair< std::pair<int, int>, std::pair<int, int> > > GetLines3(int threshold)

Use: Hough_Threshold

1. Εύρεση όλων των κελιών που έχουν τιμή μεγαλύτερη από το μέγιστο των τιμών του πίνακα accumulator
2. Για όλα τα κελιά που βρέθηκαν γίνεται εκκαθάριση της γειτονιάς τους
3. Υπολογισμός των ευθειών τους
4. Αποθήκευση κάθε ευθείας σε έναν vector

Output : Επιστροφή vector των ευθειών

Πίνακας 5: Συναρτήσεις της υλοποίησης Hough

doTransform
computeIntersect
Dilation
Remove
areVertical
Transform
GetLines

GetLines2
GetLines3

Harris Corner Detection

```
run: ./Corner_detection -s <ImageName> -m <Method> -d <dilation element size> -b <Blur type> -h <homography flag> -p <show result flag> -f <Filename>
```

```
int main( int argc, char** argv )
```

Input :

img_path : ImageName

filename : name of corners output

print_flag : για την εμφάνιση των ενδιάμεσων αποτελεσμάτων

homography_flag : για την κλήση της συνάρτησης homography

method_flag : για την μέθοδο εύρεσης ακμών

dilation_flag : μέγεθος του δομικού στοιχείου

flag_blur : τύπος συνάρτησης blur

flag_best_perspective : κλήση της έτοιμης συνάρτησης homography του OpenCV για σύγκριση αποτελεσμάτων

Use:

1. Καλεί τις συναρτήσεις GaussianBlur/ blur ανάλογα με την τιμή της παραμέτρου flag_blur
2. Καλεί τις συναρτήσεις Sobel_operator/Canny ανάλογα με την τιμή της παραμέτρου method_flag

3. Καλεί τη συνάρτηση Dilation
4. Καλεί τη συνάρτηση thresh_callback για την εύρεση ενός περιγράμματος που αντιστοιχεί στο αντικείμενο της εικόνας και το γεμίζει
5. Καλεί τη συνάρτηση Harris_corner_detection η οποία επιστρέφει έναν πίνακα γωνιών
6. Αποθηκεύει τα σημεία και τα ταξινομεί clockwise
7. Ανοίγει ένα αρχείο με όνομα το περιεχόμενο της μεταβλητής filename
8. Γράφει τα σημεία που επιστρέφει η συνάρτηση Harris_corner_detection

Output : αρχείο filename

vector<Point2f> Harris_corner_detection(Mat & image)

Input : κατώφλι για το κριτήριο γωνιότητας , μέγεθος παραθύρου

Use: Επιλέγει τις γωνίες σένα μέγεθος παραθύρου που έχουν μεγαλύτερη τιμή από το κατώφλι

Output : ένα vector με τις γωνίες που υπολογίστηκαν

Mat Sobel_operator(Mat & image)

Input : Εικόνα

Use: Υπολογίζει τις ακμές με τη μέθοδο Sobel

1. Υπολογίζει ξεχωριστά Gradient X και Gradient Y
2. Καλεί την addWeighted για να προσεγγίσει το ολικό Gradient
3. Ελέγχει τη μεταβλητή print_flag για να εμφανίσει τα αποτελέσματα
4. Επιστρέφει την εικόνα

Output : Εικόνα με ακμές

Πίνακας 6: Συναρτήσεις της υλοποίησης Harris

Harris Corner Detection
Sobel_operator
Erosion

Dilation
thresh_callback
Harris_corner_detection
four_point_estimation
sortCorners

Homography

void jordan(float x1, float y1, float x2, float y2, float x3, float y3, float x4, float y4, float X1, float Y1, float X2, float Y2, float X3, float Y3, float X4, float Y4, float H[9], int flag = 0)

Input : δυο τετράδες σημείων, η μια είναι οι γωνίες του αντικειμένου της εικόνας και η άλλη τα σημεία της νέας επιφάνειας στην οποία θα προβληθεί (source points , destination points) , έναν πίνακα για τη Homography και μια μεταβλητή flag η οποία καθορίζει την συνθήκη υπολογισμού της Homography , $||H|| = 1$ ή $H_{33} = 1$.

Use: Βρίσκει τον διαγώνιο πίνακα A και επιλύει το σύστημα για να υπολογιστεί ο πίνακας H στον οποίο αποθηκεύει τα αποτελέσματα.

Output : -

void ComputeHomography(vector<Point2f>& src_corners, vector<Point2f>& dst_corners, Mat src)

Input : ένας vector με τις γωνίες της αρχικής εικόνας , ένας vector με τις γωνίες της τελικής εικόνας και η αρχική εικόνα

Use: Υπολογίζει τη Homography και εφαρμόζει το μετασχηματισμό για να δημιουργηθεί η νέα εικόνα.

1. Καλεί τη συνάρτηση jordan για να υπολογίζει τον πίνακα H
2. Υπολογίζει τον αντίστροφο του πίνακα H
3. Γεμίζει τα στοιχεία της εικόνας βρίσκοντας τη θέση που αντιστοιχεί στην αρχική εικόνα και αντιγράφει το περιεχόμενό της

Output : -

vector<Point2f> findRectangle(vector<Point2f>& corners)

Input : ένας vector με τις γωνίες της αρχικής εικόνας

Use:

1. Ταξινομεί τα σημεία καλώντας τη `sortCorners`
2. Υπολογίζει το πλάτος της νέας εικόνας παίρνοντας την μέγιστη απόσταση των δυο άνω σημείων και των δυο κάτω.
3. Υπολογίζει το ύψος παίρνοντας την μέγιστη απόσταση των δυο αριστερών σημείων και των δυο δεξιών
4. Έχοντας βρει ύψος και πλάτος της νέας εικόνας κατασκευάζει τα σημεία με βάση αυτά και ότι το πρώτο είναι το (0,0)
5. Επιστρέφει τον πίνακα με τα νέα σημεία

Output : ένας vector με τις γωνίες της νέας εικόνας

Πίνακας 7 : Συναρτήσεις της υλοποίησης Homography

Jordan
ComputeHomography
findRectangle

6.2 Web Εφαρμογή

Στόχος είναι η υλοποίηση ενός user interface με το οποίο ο χρήστης θα έχει τη δυνατότητα να κάνει upload μια εικόνα και η εφαρμογή θα επιστρέφει τη διορθωμένη εικόνα.

Στην εφαρμογή που υλοποιήθηκε γίνεται χρήση των παραπάνω προγραμμάτων που αναπτύχθηκαν για τη μελέτη των μεθόδων και την εξαγωγή συμπερασμάτων. Συγκεκριμένα, επιλέγονται οι παράμετροι που έδωσαν καλύτερα ποσοστά και για τις δυο μεθόδους, Hough και Harris.

Για τη μέθοδο Hough χρησιμοποιήθηκαν οι τιμές $\rho = 1.5$, $\theta = 1.0$ και για την εύρεση των ακμών χρησιμοποιήθηκε η παραλλαγή 2 της μεθόδου. Για τη μέθοδο Harris χρησιμοποιήθηκε Gaussian Blur με μέγεθος πυρήνα 3 και $\sigma = 1.5$, για την ανίχνευση ακμών χρησιμοποιήθηκε ο Canny Edge detector με `threshold = 100`, για Dilation χρησιμοποιήθηκε το δομικό στοιχείο RECT (τετράγωνο) με μέγεθος φίλτρου 3x3 και για

τον αλγόριθμο των Harris και Stephens χρησιμοποιήθηκε μέγεθος block = 3 και threshold = 100. Ο χρήστης μπορεί να επιλέξει με ποια μέθοδο θα γίνει η διόρθωση της εικόνας και αν η επιλογή του δεν έχει ικανοποιητικά αποτελέσματα μπορεί να συγκρίνει επιλέγοντας την άλλη.

Η συγκεκριμένη εφαρμογή χρησιμοποίησε τον Nginx ως web server. Ένας server (διακομιστής), μπορεί να είναι οποιοσδήποτε υπολογιστής που σαν στόχο έχει να εξυπηρετεί άλλους υπολογιστές. Παρακάτω, βλέπουμε κάποια απαραίτητα βήματα που πρέπει να γίνουν και κάποιες προϋποθέσεις που πρέπει να έχει ένας υπολογιστής για να φιλοξενήσει την παρούσα εφαρμογή ως server.

6.2.1 Απαιτήσεις του συστήματος

Μια λίστα με τις απαιτήσεις για να μπορέσει να λειτουργήσει ο web server:

- RAM (minimum) 256 MB
- disk space (minimum) 10 GB disk space

OS: Ένα από τα παρακάτω λειτουργικά συστήματα x86_64 Linux flavors: Redhat (RHEL 5/6), Debian GNU/Linux stable, Ubuntu LTS. Any 64-bit Linux είναι εντάξει και συνίσταται για την υποστήριξη που διαθέτη και τη σταθερότητά που προσφέρει.

Web server: nginx (versions > 1.2.0) (η version που χρησιμοποιήθηκε: nginx/1.4.6 Ubuntu 4.17)

PHP: php5-fpm (versions > 5.3.8) (mine PHP 5.5.9-1)

Τα βήματα για τη λειτουργία ενός Nginx web server

Step 1 : Install source repositories
Step 2 : Install the NGINX web server
Step 3 : Install PHP-FPM for processing
Step 4 : Set up NGINX
Step 5 : Set up PHP-FPM
Step 6 : Configure NGINX to use PHP Processor (enable php)
Step 7 : Test configuration

6.2.2 Ανάπτυξη

Η εφαρμογή υλοποιήθηκε σε γλώσσα προγραμματισμού PHP (για το κομμάτι του server) και Javascript (για το κομμάτι του client). Μετά την εγκατάσταση της PHP 5.5.9-1 θα

πρέπει να γίνουν ρυθμίσεις στο αρχείο `php.ini`. Οι αλλαγές που έγιναν αφορούσαν το μέγιστο μέγεθος αρχείου που μπορεί να ανεβάσει ο χρήστης, το χρόνο αναμονής μετά από κάθε αίτημα καθώς και το πλήθος των ταυτόχρονων uploads που μπορεί να επεξεργαστεί.

Οι γλώσσες προγραμματισμού που υποστηρίζει ο Nginx web server είναι η Perl, python και php. Μπορούμε να τρέχουμε παραπάνω από ένα web server όταν αντιστοιχίσουμε κάθε server στη δικιά του ξεχωριστή θύρα και διεύθυνση. Συνήθως με την πρώτη εγκατάσταση ενός server η πρώτη θύρα που επιλέγεται είναι η θύρα 80. Εάν δεν έχουμε καταχωρίσει κάποια υπαρκτή διεύθυνση μπορούμε, για τους σκοπούς ανάπτυξης της εφαρμογών, να χρησιμοποιήσουμε την 127.0.0.1. Όταν θα τρέχουμε τον Nginx θα δέχεται αιτήσεις που θα έχουν προορισμό την καθορισμένη θύρα και διεύθυνση που έχουμε καταχωρήσει. Υπάρχουν δυο αρχεία καταγραφής που είναι σημαντικά και θα πρέπει να αναφερθούμε σε αυτά, είναι το αρχείο καταγραφής `access_log` και το `error_log`. Το `access_log` είναι το αρχείο που περιέχει όλες τις αιτήσεις που στέλνονται από τους πελάτες (client) στους εξυπηρετητές (server). Σημαντικές πληροφορίες που μπορούμε να αντλήσουμε από το αρχείο `access_log` είναι η διεύθυνση ip του client, το όνομα του έγγραφου που στάλθηκε, την κατάσταση που βρίσκεται ο http κώδικας, την ώρα που συνέβη η αίτηση του έγγραφου. Το `error_log` είναι το αρχείο καταγραφής που περιέχει σημαντικές πληροφορίες για την λειτουργία του server. Συνήθως όταν παρουσιάζεται ένα πρόβλημα στο server θα πρέπει να ανατρέξουμε στο αρχείο `error_log` για την επίλυση του.

Μετά την εγκατάσταση του NGINX μπορούν να ρυθμιστούν οι διευθύνσεις που θα του αντιστοιχούν (virtual hosts). Στον φάκελο `/etc/nginx/` μπορούν να γίνουν οι κατάλληλες ρυθμίσεις όλων των διευθύνσεων που θα φιλοξενεί ο εξυπηρετητής. Συγκεκριμένα στον φάκελο `sites-available` βρίσκονται όλα τα virtual hosts που φιλοξενούνται από τον server ενώ στο φάκελο `sites-enabled`, βρίσκονται όλα τα virtual hosts εφόσον είναι ήδη ενεργά. Ο φάκελος `sites-available` περιέχει ένα configuration file των virtual hosts. Ο φάκελος `sites-enabled` ουσιαστικά έχει ένα link (σύνδεσμο) που δείχνει στο αρχείο του φακέλου `sites-available`. Αυτό επιτυγχάνεται με την παρακάτω εντολή

```
ln -s /etc/nginx/sites-available/THE.VIRTUAL.HOST.FILENAME /etc/nginx/sites-enabled/THE.VIRTUAL.HOST.FILENAME
```

Αλλαγές στο configuration file

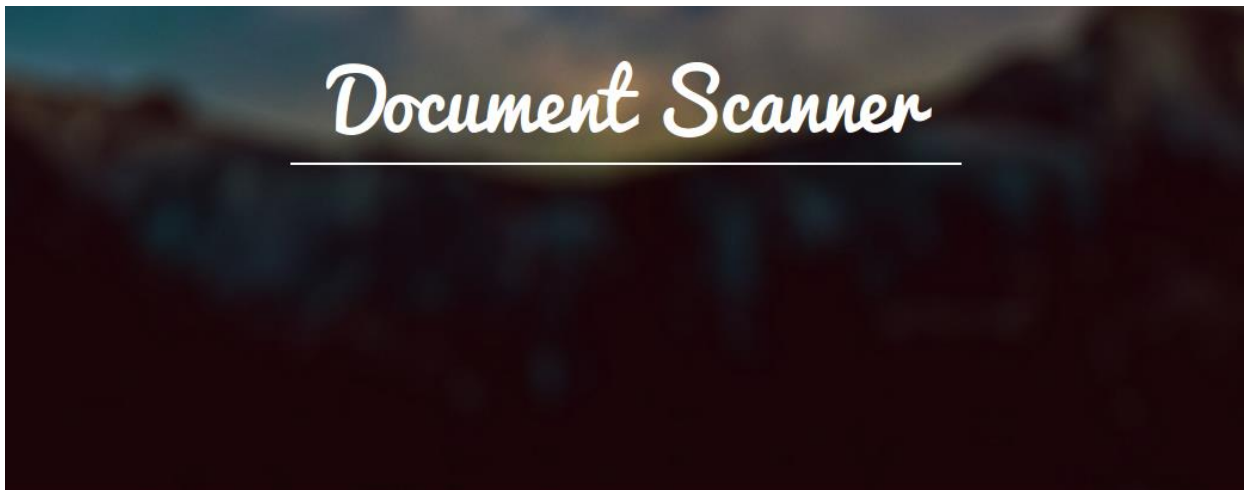
- Αλλαγή ονόματος εξυπηρετητή (Server name)
- Αλλαγή αριθμού θύρας που θα χρησιμοποιηθεί
- Αλλαγή της τοποθεσίας root `/var/www/default/html` στον αντίστοιχο φάκελο που βρίσκεται η εφαρμογή
- Αποσχολιασμός του κώδικα για PHP και προσθήκη του φακέλου που βρίσκεται η PHP
- Αλλαγή του `hash_bucket_size` στο αρχείο `etc/nginx/nginx.conf`

Αφαίρεση περιθωρίου και διόρθωση παραμόρφωσης σε έγγραφα από κάμερα

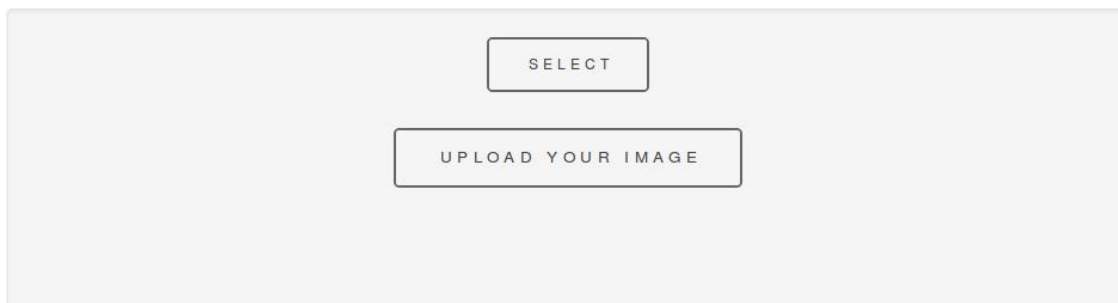
Στη συνέχεια για να ξεκινήσει ο Server χρησιμοποιούμε την εντολή

Nginx restart and php5-fpm restart

Παρακάτω παρουσιάζονται κάποιες εικόνες από την υλοποίηση.



UPLOAD A LOCAL IMAGE



Εικόνα 69: Αρχική σελίδα

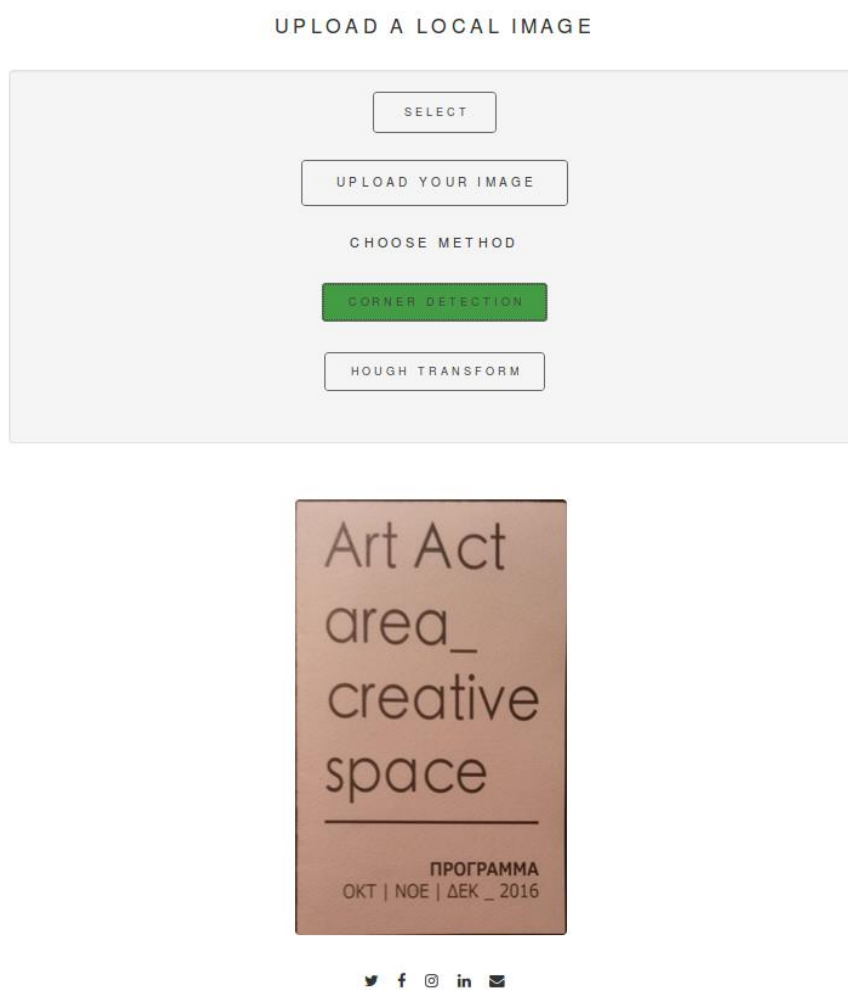
CHOOSE METHOD

CORNER DETECTION

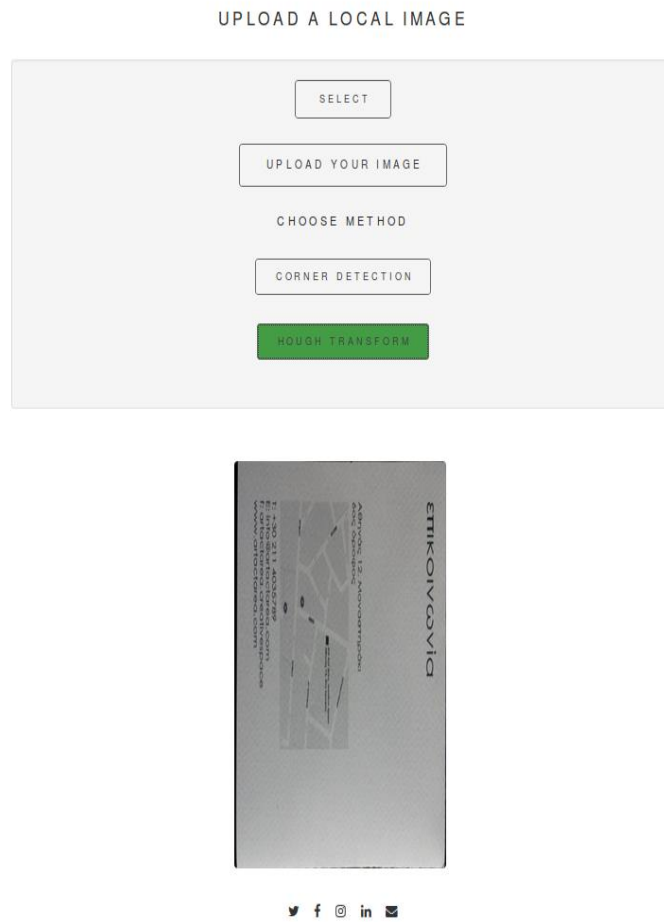
HOUGH TRANSFORM



Εικόνα 70: Upload Image



Εικόνα 71: Homography with Corner Detection



Εικόνα 72: Homography with Hough Transform

7. ΣΥΜΠΕΡΑΣΜΑΤΑ

Στην παρούσα πτυχιακή ερευνήθηκαν και υλοποιήθηκαν δυο μέθοδοι διόρθωσης της προοπτικής παραμόρφωσης σε εικόνες από κάμερα. Η μια μέθοδος βασίζεται στην εργασία των Geetha Kiran A. και Murali S [10] και χρησιμοποιεί ανίχνευση γωνιών με τη χρήση του Harris corner detector και η άλλη μέθοδος βασίζεται στην εργασία των Ryan Baumann, Christopher Blackwell και W. Brent Seale [11] και χρησιμοποιεί ανίχνευση ευθειών με Hough Transform. Η υλοποίηση που έγινε στα πλαίσια της παρούσας πτυχιακής μελετά τις παραπάνω μεθοδολογίες και αποσκοπεί στην βελτίωση των αποτελεσμάτων τους. Χρησιμοποιήθηκε το ίδιο σετ δεδομένων και για τις δυο υλοποιήσεις ώστε να μπορεί να γίνει η αποτίμησή τους.

Αρχικά στην πρώτη μέθοδο, εξετάστηκαν τρόποι βελτίωσης των ενδιάμεσων σταδίων ώστε να επιτευχθεί καλύτερη ανίχνευση γωνιών στο προτελευταίο στάδιο που είναι η εφαρμογή του Harris. Επίσης, έγινε εισαγωγή κάποιων παραμέτρων για την εύρεση κάποιας αποδοτικότερης λύσης. Με βάση τα αποτελέσματα των πινάκων του πέμπτου κεφαλαίου διαπιστώνουμε πως με τη χρήση ανιχνευτή ακμών Canny ανεβαίνουν πολύ τα ποσοστά σε εγκυρότητα (10% διαφορά σε σύγκριση με Sobel που προτάθηκε στο [10]), ενώ αλλαγές στον τρόπο εξομάλυνσης και στο μέγεθος του δομικού στοιχείου κατά την εφαρμογή μορφολογικών πράξεων παρουσιάζουν μικρές διακυμάνσεις στα ποσοστά (1%-3%). Επομένως, διαπιστώνουμε πως μεγαλύτερη έμφαση πρέπει να δοθεί στον τρόπο που θα γίνει η ανίχνευση ακμών. Με μια καλύτερη ανίχνευση ακμών που βασίζεται σε machine learning (μηχανική μάθηση) θα βελτιώνονταν σημαντικά τα αποτελέσματα και η εγκυρότητα της μεθόδου.

Στη δεύτερη μέθοδο χρησιμοποιήθηκαν τρεις διαφορετικοί τρόποι για την εξαγωγή των ευθειών από τον πίνακα συσσώρευσης. Ο πρώτος ακολουθεί τη μέθοδο που προτείνεται στο [11] και είναι η κλασική ανίχνευση ευθειών hough. Χρησιμοποιήθηκε ως βάση για τη σύγκριση των άλλων δυο παραλλαγών. Με βάση των πίνακα σύγκρισης του πέμπτου κεφαλαίου παρατηρούμε πως και οι δύο τρόποι Hough_VP και Hough_Th έχουν υψηλότερα ποσοστά από τον κλασικό τρόπο που χρησιμοποιήθηκε ως βάση (10-16% διαφορά). Η Hough_VP ήταν πολύ κοντά σε ποσοστά με την Hough_Th, που δεν φάνηκε να την επηρεάζει σημαντικά ο τρόπος που κβαντίζεται ο πίνακας συσσώρευσης και αυτό γιατί τα αποτελέσματά της βασίζονται στην επιλογή ενός κατωφλίου. Με βάση τα αποτελέσματα και των τριών η Hough_VP σε αυτήν την περίπτωση είχε τα καλύτερα αποτελέσματα. Στηρίζεται στην εύρεση τεσσάρων τοπικών μεγίστων του πίνακα συσσώρευσης με σχέσεις παραλληλίας και καθετότητας στα ζεύγη ευθειών. Η αδυναμία που παρουσιάζει μπορεί να επιλυθεί με καλύτερη ανίχνευση ακμών ή αύξηση του παραθύρου της γειτονιάς κατά τον έλεγχο των τοπικών μεγίστων. Η απόδοση της Hough_Th εξαρτάται από την τιμή του threshold που θα εφαρμοστεί για την επιλογή των ευθειών.

Με τη σύγκριση των δυο διαφορετικών μεθοδολογιών διαπιστώνουμε πως η μέθοδος που χρησιμοποιεί το μετασχηματισμό Hough, παρουσιάζει μεγαλύτερα ποσοστά εγκυρότητας ενώ η μέθοδος που στηρίζεται σε ανίχνευση γωνιών, παρουσιάζει μεγαλύτερη ευαισθησία στο θόρυβο και είναι λιγότερο σταθερή συγκριτικά με τις μεθόδους του Hough.

Μια πρόταση βελτίωσης της πρώτης μεθόδου είναι χρήση του FAST που βασίζεται σε εκπαίδευση του αλγορίθμου εύρεσης γωνιών και θα έχει πολύ καλύτερα αποτελέσματα

από τη μέθοδο Harris. Μια άλλη πρόταση βελτίωσης αφορά και τις δυο μεθόδους και σχετίζεται με τη χρήση καλύτερων τεχνικών ανίχνευσης ακμών ώστε τα επόμενα στάδια να εμφανίσουν καλύτερα αποτελέσματα και να περιοριστούν οι έλεγχοι για την εύρεση τεσσάρων γωνιών. Τέλος, μια τελευταία πρόταση βελτίωσης αφορά το κοινό κομμάτι των δυο μεθοδολογιών, δηλαδή την εφαρμογή της ομογραφίας. Βασική προϋπόθεση και για τις δυο περιπτώσεις τέθηκε η εύρεση τεσσάρων σημείων για την εφαρμογή του προβολικού μετασχηματισμού. Έτσι και στις δυο περιπτώσεις εφαρμόστηκαν τρόποι περιορισμού των σημείων που εξάγουν ώστε να προκύψουν τέσσερα. Οι έλεγχοι όμως που προστίθενται για τον περιορισμό των σημείων παρουσιάζουν κάποια σφάλματα καθώς εξαρτώνται από την εικόνα. Ένας τρόπος βελτίωσης θα ήταν η χρήση παραπάνω σημείων για τον υπολογισμό της ομογραφίας. Αυτό μπορεί να επιτευχθεί με τη μέθοδο RANSAC που χρησιμοποιεί σημειοσύνολα για τον υπολογισμό της ομογραφίας και θα εμφανίσει την καλύτερη τετράδα σημείων.

ΠΙΝΑΚΑΣ ΟΡΟΛΟΓΙΑΣ

Ξενόγλωσσος όρος	Ελληνικός Όρος
Accumulator	Πίνακας συσσώρευσης
Accuracy	Ακρίβεια
Average	Μέσος
Binarization	Δυαδική μετατροπή
Blur	Θόλωση
Blocksize	Μέγεθος παραθύρου
Classification	Κατηγοριοποίηση, ομαδοποίηση
Closing	Κλείσιμο
Corner detection	Ανίχνευση γωνιών
Contour	Περίγραμμα
Client	Πελάτης
Cornerness response	Απόκριση γωνιότητας
Clustering	Ταξινόμηση
Dewarping	Διόρθωση καμπυλότητας
Dilation	Διαστολή
Discriptor	Περιγραφικό στοιχείο
Edge detection	Ανίχνευση ακμών
Element	Στοιχείο
Element size	Μέγεθος στοιχείου
Element type	Τύπος στοιχείου
Erosion	Διάβρωση
Evaluation	Αποτίμηση
Feature extraction	Εξαγωγή χαρακτηριστικών
Gradient	Μέγεθος κλίσης
Grayscale	Σε αποχρώσεις του γκρι
Homography	Ομογραφία
Image Processing	Επεξεργασία εικόνας
Intensity image	Εικόνα έντασης
Mapping	Αντιστοίχιση
Method	Μέθοδος
Moments	Ροπές
Morphological operations	Μορφολογικές πράξεις
Kernel	Πυρήνας, Φίλτρο
Line detection	Εύρεση ευθειών
Link	Σύνδεσμος
Opening	Άνοιγμα
Operator	Τελεστής
Pattern Recognition	Αναγνώριση προτύπων
Perspective	Προοπτική
Perspective correction	Διόρθωση της προοπτικής
Pixel	Εικονοστοιχείο
Projective	Προβολικός
Pre – processing	Προεπεξεργασία
Post – processing	Μετεπεξεργασία
Process	Επεξεργασία
Run	Εκτέλεση

Segmentation	Κατάτμηση
Server	Διακομιστής
Scanner	Σαρωτής
Skew Correction	Διόρθωση στροφής
Smoothing	Εξομάλυνση
Transformation	Μετασχηματισμός
Threshold	Κατώφλι , όριο
Thresholding	Κατωφλίωση
Vanishing point	Σημείο στο άπειρο

ΣΥΝΤΜΗΣΕΙΣ – ΑΡΚΤΙΚΟΛΕΞΑ – ΑΚΡΩΝΥΜΙΑ

OCR	Optical Character Recognition
CIL	Computational Intelligence Laboratory
DB	Database
RGB	Red Green Blue Color Model
JFC	John F. Canny
OpenCV	Open Source Computer Vision Library
APP	Application
ΙΠΤ	Ινστιτούτο Πληροφορικής και Τηλεπικοινωνιών
ΕΚΕΦΕ	Εθνικό Κέντρο Ερευνών και Φυσικών Επιστημών
ΕΚΠΑ	Εθνικό και Καποδιστριακό Πανεπιστήμιο Αθηνών

ΑΝΑΦΟΡΕΣ

- [1] J. Serra, *Image Analysis and Mathematical Morphology*. New Year Academic Press, 1982.
- [2] Diya Chudasama, Tanvi Patel, Shubham Joshi "Image Segmentation using Morphological Operations" *International Journal of Computer Applications* (0975 –8887) Volume 117 –No. 18, May 2015.
- [3] E. Sobel. *Camera Models and Machine Perception*. PhD thesis, Electrical Engineering Department, Stanford University, 1970.
- [4] J. F. Canny. "A computational approach to edge detection". *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679-697, 1986.
- [5] H. Moravec (1980) "Obstacle avoidance and navigation in the real world by a seeing robot rover", Technical Report CMU-RI-TR-3, Carnegie-Mellon University, Robotics Institute.
- [6] C. Harris & M.J. Stephens (1988) "A combined corner and edge detector", In *Alvey Vision Conference*, Vol 15, pp 147–152
- [7] Duda, R. O. and P. E. Hart, "Use of the Hough Transformation to Detect Lines and Curves in Pictures," *Comm. ACM*, Vol. 15, pp. 11–15 (January, 1972)
- [8] Hough, P.V.C. *Method and means for recognizing complex patterns*, U.S. Patent 3,069,654, Dec. 18, 1962.
- [9] P.V.C. Hough, *Machine Analysis of Bubble Chamber Pictures*, Proc. Int. Conf. High Energy Accelerators and Instrumentation, 1959
- [10] Geetha Kiran A. and Murali S. "Automatic rectification of perspective distortion from a single image using plane homography" ,*International Journal on Computational Sciences & Applications* (IJCSA) Vol.3, No.5, October 2013.
- [11] Ryan Baumann, Christopher Blackwell, and W. Brent Seales (2011) "Automatic Perspective Correction of Manuscript Images".
- [12] RS Stephens, Probabilistic approach to the Hough transform, *Image and Vision Computing*, Volume 9, Issue 1, the first BMVC 1990, February 1991, Pages 66-71.
- [13] Elan Dubrofsky "Homography Estimation", Master's Essay, Dept. Computer Sciences, Univ. British Columbia, 2009.
- [14] S.Murali, N.Avinash & Mohan S (2006) "Rectification of perspective distortion using plane homography and clues from Hough transform", *Indian conference on Computer Vision, Graphics and Image Processing*.
- [15] E. Argyle. "Techniques for edge detection," *Proc. IEEE*, vol. 59, pp. 285-286, 1971.
- [16] R. C. Gonzalez and R. E. Woods. "Digital Image Processing". 2nd ed. Prentice Hall, 2002.
- [17] M. Heath, S. Sarkar, T. Sanocki, and K.W. Bowyer. "Comparison of Edge Detectors: A Methodology and Initial Study". *Computer Vision and Image Understanding*, vol. 69, no. 1, pp. 38-54 Jan. 1998
- [18] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.
- [19] Y. Yakimovsky, "Boundary and object detection in real world images". *JACM*, vol. 23, no. 4, pp. 598-619, Oct. 1976.
- [20] N. Stamatopoulos, B. Gatos and A. Kesidis, "Automatic Borders Detection of Camera Document Images", 2nd International Workshop on Camera-Based Document Analysis and Recognition (CBDAR'07), pp.71-78, Curitiba, Brazil, September 2007
- [21] L. Jagannathan and C. V. Jawahar "Perspective Correction Methods for Camera-Based Document Analysis" *International Institute of Information Technology Gachibowli, Hyderabad, India* 2005
- [22] Syed Saqib Bukhari, Faisal Shafait, Thomas M. Breuel. "Border Noise Removal of Camera-Captured Document Images using Page Frame Detection", 4th Int. Workshop on Camera-Based Document Analysis and Recognition, (CBDAR-11), Beijing, China, Sep. 2011.
- [23] N. Stamatopoulos, B. Gatos, I. Pratikakis and S.J. Perantonis, "Goal-oriented Rectification of Camera-Based Document Images", *IEEE Transactions on Image Processing*, vol. 20, no. 4, pp. 910-920, 2011
- [24] C. Monnier, V. Ablavsky, S. Holden, M. Snorrason "Sequential correction of perspective warp in camera-based documents" *Document Analysis and Recognition*, 2005. *Proceedings 8th International Conference*