



**ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ**  
**ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ**  
**ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ**

## **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Επεξεργασία βίντεο πραγματικού χρόνου για αναγνώριση και εντοπισμό  
προβλημάτων σε δίκτυα ιχθυοκαλλιέργειας**

**Real-time video processing for detection and localization of irregular struc-  
tures on fishing nets**

**Πασπαλάκης Σταύρος**

**Επιβλέπων Καθηγητής:** Ζερβάκης Μιχάλης, Καθηγητής

**Εξεταστική επιτροπή :** Ζερβάκης Μιχάλης, Καθηγητής

Δόλλας Απόστολος, Καθηγητής

Παπανδρουλάκης Νίκος, Διευθυντής Ερευνών, Ι.ΘΑ.Β.Β.ΥΚ.

# Abstract

Fish-cage dysfunction in aquaculture installations can trigger significant negative consequences affecting the operational costs. Low oxygen levels, due to excessive fooling, lead to decreased growth performance and feed efficiency. Therefore, frequent periodic inspection of fish-cage nets is required, but this task can become quite expensive with the traditional means of employing professional divers that perform visual inspections at regular time intervals. The modern trend in aquaculture is to take advantage of IT technologies with the use of a small-sized, low-cost Autonomous Underwater Vehicle (AUV), permanently residing within a fish cage and performing regular video inspection of the infrastructure for the entire net surface. In the present study, we explore specialized image processing schemes to detect net holes of multiple area size and shape. These techniques are designed with the vision to provide robust solutions that take advantage of either global or local image structures as to provide efficient inspection of multiple net holes.

# Ευχαριστίες

Θα ήθελα να ευχαριστήσω:

Πρωτίστως τον επιβλέποντα καθηγητή μου κ. Ζερβάκη Μιχάλη για την πολύτιμη βοήθεια και καθοδήγησή του καθ' όλη τη διάρκεια εκπόνησης της διπλωματικής μου εργασίας καθώς επίσης και για την εμπιστοσύνη που μου έδειξε δίνοντας μου την δυνατότητα να εργαστώ στον τομέα που επιθυμούσα.

Επίσης τον καθηγητή κ. Δόλλα Απόστολο και τον Δρ. Παπανδρουλάκη Νίκο για τη συνεισφορά τους ως μέλη της εξεταστικής επιτροπής.

Την υπεύθυνη του εργαστηρίου κυρία Κωνσταντία Μοιρογιώργου για τον πολύτιμο χρόνο που αφιέρωσε και την καθοδήγησή της από την αρχή μέχρι το τελικό στάδιο της διπλωματικής μου εργασίας καθώς και για την στήριξη και ενθάρρυνσή της.

Τους φίλους μου για όλες τις όμορφες εμπειρίες στα Χανιά και την ουσιαστική υποστήριξη. Τέλος θα ήθελα να ευχαριστήσω την οικογένεια μου για όλη την στήριξη που μου παρείχε σε όλη αυτή την προσπάθεια.

# Πίνακας Περιεχομένων

Abstract.....	1
Ευχαριστίες.....	2
Πίνακας Περιεχομένων Εικόνων .....	4
Κεφάλαιο 1 : Εισαγωγή .....	6
1.1 Κίνητρο.....	6
1.2 Σκοπός εργασίας.....	7
1.3 Καινοτομίες εργασίας.....	9
Κεφάλαιο 2 : Συναφείς εργασίες.....	10
2.1 Εισαγωγή .....	10
2.2 Συναφείς Εργασίες.....	13
Κεφάλαιο 3 : Θεωρητικό υπόβαθρο .....	16
3.1 Εισαγωγή .....	16
3.2 Μέθοδος κατωφλίσωσης .....	16
3.2 Global processing προσέγγιση.....	17
3.3 Local processing προσέγγιση.....	18
Κεφάλαιο 4 : Προτεινόμενες μεθοδολογίες.....	25
4.1 Εισαγωγή - Global process.....	26
4.1.1 Ανάγνωση και επεξεργασία εικόνας .....	29
4.1.2 Καταμερισμός εικόνας (Pool Block).....	32
4.1.3 Καταμερισμός σε Sub Blocks και υπολογισμός έντασης.....	34
4.1.4 Υπολογισμός αθροιστικής συνάρτησης (cumulative sum) .....	36
4.1.5 Εντοπισμός πιθανών τρυπών .....	38
4.1.6 Αποτελέσματα .....	39
4.2 Εισαγωγή - Local process .....	44
4.2.1 Διαδικασία ανίχνευσης ακμών.....	47
4.2.2 Εφαρμογή Hough transform και εντοπισμός ευθειών.....	48
4.2.3 Max Peaks (Εντοπισμός Κύριων Κατευθύνσεων).....	52
4.2.4 Εντοπισμός μέσης απόστασης διχτυών .....	56
4.2.5 Εντοπισμός πιθανών τρυπών .....	61

4.2.6 Αποτελέσματα .....	64
Κεφάλαιο 5 : Συμπεράσματα και μελλ/κή έρευνα.....	78
5.1 Σύνοψη - Συμπεράσματα.....	78
5.2 Προτάσεις για μελλοντική έρευνα .....	80
Βιβλιογραφία.....	81

## Πίνακας Περιεχομένων Εικόνων

<b>Εικόνα 2.1</b> Καθολικά και Τοπικά χαρακτηριστικά .....	11
<b>Εικόνα 3.1</b> Σχέση pdf και cdf .....	18
<b>Εικόνα 3.2</b> Χώρος εικόνας και παραμετρικός χώρος για μια ευθεία .....	20
<b>Εικόνα 3.3</b> Απεικόνιση σχέσης μεταξύ χώρου εικόνας και παραμετρικού χώρου για ένα σημείο .....	20
<b>Εικόνα 3.4</b> Απεικόνιση N σημείων στους δύο χώρους Image & Hough space .....	21
<b>Εικόνα 3.5</b> Η σχέση του $(\theta, \rho)$ στο καρτεσιανό σύστημα .....	22
<b>Εικόνα 3.6</b> Απεικόνιση σχέσης ενός σημείου με τον $(\theta, \rho)$ χώρο .....	23
<b>Εικόνα 3.7</b> Σχέση σημείων με ημιτονοειδείς γραμμές στον Hough space.....	24
<b>Εικόνα 4.1</b> Block Diagram πρώτης μεθόδου .....	27
<b>Εικόνα 4.2</b> Block Diagram συναρτήσεων πρώτης μεθόδου.....	28
<b>Εικόνα 4.3</b> Επιλογές γειτνιαζόντων αντικειμένων .....	30
<b>Εικόνα 4.4</b> Παράδειγμα binary εικόνας .....	30
<b>Εικόνα 4.5</b> Gray / Binary / Invert image .....	31
<b>Εικόνα 4.6</b> Pool block .....	33
<b>Εικόνα 4.7</b> Sub block .....	35
<b>Εικόνα 4.8</b> Ιστόγραμμα με/χωρίς τρύπα .....	35
<b>Εικόνα 4.9</b> Λανθασμένες αναζητήσεις.....	36
<b>Εικόνα 4.10</b> Καταμερισμός της έντασης της εικόνας.....	37
<b>Εικόνα 4.11</b> Αθροιστική/ Threshold .....	37
<b>Εικόνα 4.12</b> Τελικό αποτέλεσμα εντοπισμού .....	38
<b>Εικόνα 4.13</b> Παράδειγμα 2: Καταμερισμός block δεύτερης εικόνας/ Ιστόγραμμα έντασης .....	40
<b>Εικόνα 4.14</b> Παράδειγμα 2: Αθροιστική συνάρτηση / Threshold/ Τελικό αποτέλεσμα.....	40
<b>Εικόνα 4.15</b> Παράδειγμα 3: Καταμερισμός εικόνας/ Ιστόγραμμα έντασης .....	41
<b>Εικόνα 4.16</b> Παράδειγμα 3: Αθροιστική συνάρτηση/ Threshold/ Τελικό αποτέλεσμα.....	42
<b>Εικόνα 4.17</b> Block Diagram δεύτερης μεθόδου .....	45
<b>Εικόνα 4.18</b> Block Diagram συναρτήσεων δεύτερης μεθόδου.....	46
<b>Εικόνα 4.19</b> Peaks του Hough/ Οι αντίστοιχες ευθείες .....	49
<b>Εικόνα 4.20</b> Κατανομή όλων των peaks με τιμή μεγαλύτερη του μηδενός .....	50
<b>Εικόνα 4.21</b> Threshold κατανομής .....	51

<b>Εικόνα 4.22</b>	Peaks των strong lines/ Αντίστοιχες γραμμές .....	51
<b>Εικόνα 4.23</b>	Τα max peaks του accumulator bin.....	52
<b>Εικόνα 4.24</b>	Γράφημα των τιμών που παίρνουν τα max peaks .....	53
<b>Εικόνα 4.25</b>	Blue peaks/ Αντίστοιχες ευθείες .....	54
<b>Εικόνα 4.26</b>	Εντοπισμός των δύο maxima peak .....	55
<b>Εικόνα 4.27</b>	Κύριες ευθείες.....	55
<b>Εικόνα 4.28</b>	Απεικόνιση απόστασης μεταξύ γραμμών (1ης ομάδας) .....	57
<b>Εικόνα 4.29</b>	Απεικόνιση απόστασης μεταξύ γραμμών (2ης ομάδας) .....	58
<b>Εικόνα 4.30</b>	Ιστογράμματα 1ης ομάδας ευθειών .....	59
<b>Εικόνα 4.31</b>	Ιστογράμματα 2ης ομάδας ευθειών .....	60
<b>Εικόνα 4.32</b>	Το πλέγμα που δημιουργήσαμε .....	62
<b>Εικόνα 4.33</b>	Εντοπισμός τρυπών .....	63
<b>Εικόνα 4.34</b>	Παράδειγμα 2 / Gray / Binary / Inverse binary` .....	64
<b>Εικόνα 4.35</b>	Παράδειγμα 2 Threshold / Peaks στον Hough / Strong lines.....	65
<b>Εικόνα 4.36</b>	Παράδειγμα 2 Blue Peaks / Ιστόγραμμα max peaks.....	67
<b>Εικόνα 4.37</b>	Παράδειγμα 2 Maxima / Κύριες κατευθύνσεις .....	68
<b>Εικόνα 4.38</b>	Παράδειγμα 2 πλέγμα / Εντοπισμός τρύπας.....	69
<b>Εικόνα 4.39</b>	Παράδειγμα 3 / Gray / Binary / Inverse binary .....	70
<b>Εικόνα 4.40</b>	Παράδειγμα 3 Threshold / Max peaks / Αντίστοιχες ευθείες.....	72
<b>Εικόνα 4.41</b>	Παράδειγμα 3 Blue peaks / Ιστόγραμμα max peaks.....	73
<b>Εικόνα 4.42</b>	Παράδειγμα 3 Maxima / Κύριες κατευθύνσεις .....	75
<b>Εικόνα 4.43</b>	Παράδειγμα 3 Πλέγμα / Εντοπισμός τρύπας.....	77

# Κεφάλαιο 1 : Εισαγωγή

## 1.1 Κίνητρο

Η μηχανική όραση αποτελεί ένα από τα πλέον σημαντικά πεδία της τεχνικής νόησης.

Είναι ένας κλάδος ιδιαίτερα αναπτυσσόμενος στον οποίον οι επιστήμονες προσπαθούν να αντιμετωπίσουν προβλήματα του σύγχρονου κόσμου. Σκοπός της είναι η αναπαραγωγή, μέσω αλγορίθμων, της αίσθησης της ανθρώπινης όρασης.

Ένας απ' τους βασικούς τομείς στους οποίους έχει διεισδύσει η μηχανική όραση τα τελευταία χρόνια είναι και οι ιχθυοκαλλιέργειες. Αυτές αποτελούν σήμερα έναν από τους ταχύτερα αναπτυσσόμενους κλάδους της ζωικής παραγωγής. Σύμφωνα με τον Παγκόσμιο Οργανισμό Τροφίμων (FAO) μέχρι το 2030 πάνω από το 65% των αλιευτικών προϊόντων θα προέρχεται από την υδατοκαλλιέργεια.

Η Ελλάδα τα τελευταία 25 χρόνια έχει αξιοποιήσει το ανταγωνιστικό πλεονέκτημα του Ελληνικού αρχιπελάγους με τον πλέον δυναμικό τρόπο. Εκτιμάται ότι το 63% της εγχώριας παραγωγής αλιευτικών προϊόντων προέρχεται από την υδατοκαλλιέργεια [1], μετατρέποντας έτσι τον κλάδο σε έναν από τους ταχύτερα αναπτυσσόμενους της πρωτογενούς παραγωγής της χώρας και καθιστώντας τον έναν από τους πιο σημαντικούς για την εγχώρια οικονομία. Παράλληλα, παρατηρήθηκε μεγάλη στροφή προς τις ιχθυοκαλλιέργειες [2] η οποία ανάγκασε το σχετιζόμενο ανθρώπινο δυναμικό σε μια αναζήτηση αξιόπιστων λύσεων στα προβλήματα που ταλανίζουν τον τομέα.

Οι σύγχρονοι μέθοδοι ιχθυοκαλλιέργειας παρουσιάζουν την τάση της μεταφοράς των μονάδων παραγωγής μακριά από τις ακτές, σε περιβάλλοντα κατάλληλα δομημένα για τη καλύτερη απόδοση των μονάδων παραγωγής και τη μικρότερη επίπτωση στο συνολικό οικοσύστημα. Παρόλ' αυτά παρουσιάζονται προβλήματα τα οποία πρέπει να επιλύονται με την εμφάνισή τους για την αποδοτικότερη λειτουργία των δομών. Ένα από αυτά είναι η απόδραση ψαριών από τους ιχθυοκλωβούς, λόγω της αναπόφευκτης φθοράς ή λόγω ατυχήματος. Αυτό έχει σαν αποτέλεσμα τόσο την οικονομική απώλεια μονάδων όσο και τη λεγόμενη «γενετική ρύπανση» από την είσοδο εκτρεφόμενων ατόμων στο φυσικό περιβάλλον. Επίσης αγκάθι παραμένει και η ενεργή συμμετοχή του ανθρώπινου παράγοντα, όπως οι δύτες, για το χειρισμό του εξοπλισμού. Είναι κάτι που αυξάνει το κόστος και το χρόνο διεκπεραίωσης μιας λειτουργίας.

Η σύνδεση IT τεχνολογιών με τις ιχθυοκαλλιέργειες θεωρείται αναπόφευκτη για να λυθούν τα παραπάνω προβλήματα. Υποθαλάσσιες κάμερες, ρομποτικά οχήματα και αισθητήρες συνδράμουν στην καλύτερη παρακολούθηση των δομών, την καταγραφή και μέτρηση δεδομένων καθώς και στην ενημέρωση σε πραγματικό χρόνο (real time) των χρηστών μέσα από εξυπηρετητές (servers), εφαρμογές (applications) ή emails [3], [4], [5], [6], [7].

Μια λύση που μπορεί να οδηγήσει στη μείωση ανθρώπινου δυναμικού είναι η χρήση αυτοματοποιημένων συστημάτων επιτήρησης, όπως αυτή των αυτόνομων οχημάτων (AUV) τα οποία παραμένουν μέσα στις δομές, καταγράφοντας το υποθαλάσσιο περιβάλλον. Με τη χρήση ειδικών προγραμμάτων (software) δίνεται η δυνατότητα της αυτόματης ανάλυσης και του εντοπισμού προβλημάτων εντός των μονάδων παραγωγής, γι' αυτό ο ορθός κι αποτελεσματικός τρόπος ανάπτυξης αλγορίθμων επεξεργασίας των εικόνων παίζει σημαντικό ρόλο.

Εν κατακλείδι, η ανάπτυξη μεθοδολογιών οι οποίες επιλύουν τα παραπάνω, μπορούν να συνεισφέρουν στον έγκαιρο εντοπισμό προβλημάτων που παρουσιάζονται στον τομέα των ιχθυοκαλλιεργειών.

## 1.2 Σκοπός εργασίας

Στόχος της παρούσας διπλωματικής είναι η επεξεργασία υποθαλάσσιων εικόνων από δίκτυα ιχθυοκλωβών και ο εντοπισμός ενδεχόμενων ασυνεχειών πάνω σε αυτά (πχ. τρύπες κτλπ).

Οι εικόνες που χρησιμοποιήθηκαν στην παρούσα εργασία περιέχουν δίκτυα διαφορετικού σχήματος και μεγέθους. Με δεδομένο ότι, οι υποθαλάσσιες συνθήκες δημιουργούν αλλοιώσεις των δικτύων, προσπαθήσαμε με αυτό τον τρόπο να εξετάσουμε τις μεθόδους μας σε διαφορετικές καταστάσεις του πλέγματος κάθε φορά.

Ακόμη πειραματιστήκαμε σε επίπεδο παραμόρφωσης του δικτυού. Εξετάσαμε την κλίση του καθώς και το μέγεθος της τρύπας κάθε φορά, μέσω συνδυασμένων τεχνικών παράλληλης μετατόπισης (translation), περιστροφής (rotation) και αλλαγή μεγέθους (resize) πάνω σε κάθε εικόνα.

Η διαδικασία ανίχνευσης των ασυνεχειών προσομοιάζει με τεχνικές αναγνώρισης προτύπων σε ανόμοιες δομές (pattern irregularities). Κλασικές μέθοδοι εντοπισμού ακμών (edge detection) όπως ο εντοπισμός ακμών Canny (Canny edge detector) ή μια απλή εφαρμογή του μετασχηματισμού Hough, δεν μας δίνουν αξιόπιστα αποτελέσματα.

Ο κύριος λόγος είναι ο θόρυβος που δημιουργείται στις υποθαλάσσιες εικόνες από τον χαμηλό φωτισμό, την υποθαλάσσια χλωρίδα που ενδεχομένως να υπάρχει πάνω στο δίκτυο ή τέλος από τις παραμορφώσεις που μπορεί να έχει το πλέγμα λόγω της κυμάτωσης του νερού. Αν επιχειρήσουμε να εφαρμόσουμε τις παραπάνω μεθόδους σε μια τέτοια εικόνα τα αποτελέσματα που θα λάβουμε δεν θα αντιστοιχούν στις προσδοκίες μας.

Για την εξαγωγή ικανοποιητικών αποτελεσμάτων χρειαζόμαστε αρχικά μια καλή επεξεργασία της εικόνας, ώστε τα αντικείμενα πάνω σε αυτήν να είναι ευδιάκριτα, ενώ στη συνέχεια χρειάζονται μέθοδοι ικανοί να επεξεργαστούν το παραπάνω αποτέλεσμα στο βαθμό που ξεπερνούν προβλήματα όπως ο θόρυβος, οι ασυνέχειες, οι ανομοιομορφίες κα.

Απ' τη σκοπιά της μηχανικής όρασης και της ψηφιακής επεξεργασίας υπάρχουν ποικίλες προσεγγίσεις στον τομέα της αναγνώρισης προτύπων (pattern recognition) για την ανίχνευση ανόμοιων δομών πάνω σε εικόνες. Μια από τις πρώτες προσεγγίσεις που έγιναν στον χώρο ήταν αυτή των Leung & Malik στην εργασία «Detecting, localizing and grouping repeated scene elements from an image» [21] η οποία επικεντρώνεται στην ανάλυση, στην επεξεργασία και στην ομαδοποίηση όμοιων γειτονικών δομών οι οποίες έχουν υποστεί κάποια παραμόρφωση (affine transformation).



Μια άλλη προσέγγιση έγινε στην εργασία «Periodic pattern of texture analysis and synthesis based on texels distribution» [22] η οποία θέτει ως στόχο την αναζήτηση αντικειμένων πάνω στην εικόνα με σκοπό την σύνθεση και εξαγωγή ενός πιο σύνθετου μοτίβου με παρόμοια χαρακτηριστικά.

Άλλες προσεγγίσεις, επικεντρώνονται στην αναζήτηση περιοδικών δομών μέσω της τριγωνικής μεθόδου Delaunay (Delaunay triangulation) ή στην κατά τόπους στατιστική περιοδικότητα όπως το «Near-regular texture analysis and manipulation» [24].

Οι παραπάνω προσεγγίσεις όπως και συναφείς άλλες εργασίες οι οποίες παρουσιάζονται αναλυτικά στο κεφάλαιο 2 μας εισήγαγαν σε πρώτη φάση στον κόσμο της αναγνώρισης προτύπων (pattern recognition).

Αρχικά οι τεχνικές που υλοποιούνται στις παραπάνω εργασίες, μας βοήθησαν να κατανοήσουμε τον τρόπο με τον οποίο μπορούμε να κάνουμε την ανίχνευση ενός μοτίβου, την ανάλυση επαναλαμβανόμενων δομών καθώς και στην ομαδοποίηση όμοιων αντικειμένων. Τέλος η ιδέα της επεξεργασίας και σύνθεσης νέων όμοιων δομών ήταν καταλυτική στην υλοποίηση της δεύτερης μεθόδου μας, στην οποία παράγουμε ένα πλέγμα από ευθείες γραμμές παράλληλα στο υπάρχον δίκτυ της εικόνας.

Όσον αφορά τώρα στο σκέλος της αρχικής επεξεργασίας μιας υποθαλάσσιας εικόνας έχουν υπάρξει αξιόλογες προτάσεις στην βιβλιογραφία. Ενδεικτικά αναφέρουμε την εργασία «Underwater Image Enhancement Based on Removing Light Source Color and Dehazing» [40] η οποία παίρνει υπόψιν της το βάθος και τη διάχυση του φωτός, ανακατασκευάζοντας συνολικά την υποθαλάσσια εικόνα. Παρ' όλα αυτά η αρχική επεξεργασία και βελτίωση των υποθαλάσσιων εικόνων δεν αποτελεί τον κάθε αυτό σκοπό της εργασίας.

Εν τέλη στην παρούσα διπλωματική εργασία επικεντρωθήκαμε κυρίως στις δομές των διχτύων και στην αναζήτηση ασυνεχειών πάνω σε αυτά. Προτείνουμε δύο διαφορετικές μεθόδους επεξεργασίας και ανάλυσης, οι οποίες είναι χρήσιμες στην εξαγωγή συμπερασμάτων πάνω σε ανόμοια δομικά στοιχεία στο ίδιο μοτίβο.

Η πρώτη μέθοδος που αναπτύσσουμε βασίζεται στη στατιστική ανάλυση κατακερματισμένων (segmented) τμημάτων της εικόνας μέσω της τιμής της έντασης που περιέχεται στα τμήματα αυτά. Ενώ η δεύτερη βασίζεται κυρίως στην εξαγωγή μοτίβων βασισμένη στην τεχνική του Hough transform.

Η υπόλοιπη διπλωματική εργασία διαρθρώνεται με τον εξής τρόπο:

Στο δεύτερο κεφάλαιο αναφέρουμε την υπάρχουσα κατάσταση στην βιβλιογραφία γύρω από την επισκόπηση διχτύων ιχθυοκαλλιέργειας. Στο τρίτο κεφάλαιο αναφέρουμε το θεωρητικό υπόβαθρο γύρω από τις δύο μεθόδους που υλοποιήσαμε. Στο τέταρτο κεφάλαιο κάνουμε λόγο για τις δύο διαφορετικές προσεγγίσεις που υλοποιήσαμε, ενώ, τέλος, στο πέμπτο καταλήγουμε στα συμπεράσματα της εργασίας μας.

## 1.3 Καινοτομίες εργασίας

Οι δύο προτεινόμενες μέθοδοι συνδυάζουν δύο διαφορετικές τεχνικές προσεγγίσεις οπτικής ανάλυσης εικόνων. Η πρώτη εστιάζει σε μια τοπική επεξεργασία μέσω στατιστικών αναλύσεων και ιστογραμμάτων (local processing), όπου το πλέγμα χωρίζεται σε μέρη απ' τα οποία αντλούμε πληροφορία μέσω των τιμών της έντασης που περιέχουν, για την ύπαρξη ή μη κάποιας ασυνέχειας (πχ τρύπα). Ενώ, η δεύτερη, υλοποιεί μια ολική (global) ανάλυση της εικόνας με τη βοήθεια του μετασχηματισμού Hough έχοντας ως αντικείμενο μελέτης το πλέγμα ως συνολική δομή.

Η ιδέα δημιουργίας δύο μεθόδων προσέγγισης έγκειται στο γεγονός ότι οι συνθήκες κάτω από τις οποίες γίνεται η λήψη του βίντεο ή της εικόνας από κάποιο AUV είναι απρόβλεπτες, ακόμα και στις πιο καλά προστατευόμενες δομές. Γνωρίζουμε πως στις περισσότερες περιπτώσεις οι υποθαλάσσιες εικόνες περιέχουν θόρυβο, σκουπίδια ή παρουσιάζουν έλλειψη φωτεινότητας λόγω σκέδασης. Ακόμη τα ίδια τα πλέγματα των δίχτων συχνά παρουσιάζονται παραμορφωμένα ή αντεστραμμένα λόγω της κυμάτωσης.

Δεδομένων όλων των παραπάνω θεωρούμε πως ο συνδυασμός των δύο μεθόδων οι οποίες εφαρμόζουν διαφορετικές τεχνικές προσεγγίσεις θα μας δώσει ικανά και αξιόπιστα συμπεράσματα στο τέλος μιας ολοκληρωμένης διαδικασίας, πάνω στα αντικείμενα μελέτης, γνωρίζοντας πάντα τις δυνατότητες και τις αδυναμίες της κάθε μιας.

Για παράδειγμα η τοπική (local) μέθοδος μπορεί να χρησιμοποιηθεί σε μια λεπτομερή ανάλυση της δομής του πλέγματος μέσω της μοντελοποίησης επαναλαμβανόμενων μοτίβων (πχ. παράλληλων γραμμών) και στη σύγκρισή τους. Αυτό είναι κάτι που μπορεί να μας δώσει με μεγάλη ακρίβεια το σημείο ασυνέχειας πάνω στην εικόνα, όταν αυτή έχει δυσδιάκριτα σημεία. Βασικό μειονέκτημα της τοπικής μεθόδου είναι η μεγάλη πολυπλοκότητα της. Από την άλλη μεριά, η ολική (global) μέθοδος έχει μια σχετικά μικρή πολυπλοκότητα και βασίζεται στη γρήγορη ανάλυση και εξαγωγή αποτελεσμάτων χρησιμοποιώντας τον παραλληλισμό. Ενδείκνυται να εφαρμόζεται σε περιπτώσεις όπου έχουμε καθαρή απεικόνιση του πλέγματος.

Εντέλει μια εναλλαγή της ολικής (global) μεθόδου σε μια εικόνα, απ' όπου μπορούμε να εστιάσουμε σε συγκεκριμένα παράθυρα έντασης, σε συνδυασμό με μια λεπτομερή τοπική διερεύνηση ασυνεχειών πάνω στα παράθυρα αυτά, μπορεί να αναδείξει με μεγάλη ακρίβεια ενδεχόμενα προβλήματα που μπορεί να υπάρχουν σε συγκεκριμένες περιοχές πάνω στο δίκτυο.

Όλα αυτά φαίνονται καθαρά στο κεφάλαιο 4, όπου γίνεται αναλυτικότερη επεξήγηση των αποτελεσμάτων.

Για την υλοποίηση των αλγορίθμων χρησιμοποιήσαμε το λογισμικό της MATLAB, ενώ οι εικόνες που χρησιμοποιήθηκαν είναι αλιευμένες από το διαδίκτυο και τραβηγμένες κάτω από άγνωστες συνθήκες. Προσπαθήσαμε να χρησιμοποιήσουμε εικόνες οι οποίες να περιέχουν μια ποικιλία του περιεχόμενου τους, πειραματιζόμενοι τόσο στη φωτεινότητα, όσο και στις διαφορετικές κατευθύνσεις που μπορεί να έχουν τα δίκτυα, καθώς και στο μέγεθος των κελιών του πλέγματος αλλά και των τρυπών που αυτά σχηματίζουν.

## Κεφάλαιο 2 : Συναφείς εργασίες

### 2.1 Εισαγωγή

Με την εισαγωγή της IT τεχνολογίας στον τομέα των ιχθυοκαλλιεργειών, προστέθηκε μια πλειάδα νέων τρόπων παρακολούθησης των δομών τους. Για παράδειγμα, τα ρομποτικά οχήματα τύπου ROV's (Remotely Operated underwater Vehicle), τα αυτόνομα οχήματα AUV's (Autonomous Underwater Vehicle) [9], [10], τα περισκοπικά οχήματα OSV (omnidirectional surface vehicle) [11], οι στατικές κάμερες, οι αισθητήρες κα. είναι ικανά να μεταδίδουν πληροφορίες σε πραγματικό χρόνο για τον υποθαλάσσιο χώρο των εγκαταστάσεων.

Τα δεδομένα που εξάγουμε από τα βίντεο είτε από online είτε από offline πλατφόρμες είναι αρκετά σύνθετα λόγω των απρόβλεπτων συνθηκών που επικρατούν στον υποθαλάσσιο χώρο.

Αρκετές εργασίες (papers) έχουν γραφτεί πάνω στο εν λόγω ζήτημα είτε αυτό έχει να κάνει με τη βελτίωση και αποκατάσταση των εικόνων όπως η «Denoising and contrast enhancement fusion based on white balance for underwater images» [12] και η « An Algorithm for Image Restoration Based on Underwater Video Series» [13], είτε αν το ενδιαφέρον μας εστιάζεται πάνω στην αυτόματη αναζήτηση ψαριών όπως η «Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data» [14], και η «Towards automating underwater measurement of fish length: a comparison of semiautomatic and manual stereo-video measurements» [15]. Τέλος αρκετές αναφορές έχουν γίνει και για το ζήτημα των δικτύων και για την περιπλοκότητα ανίχνευσης προβλημάτων πάνω σε αυτά. Ορισμένες από αυτές αναφέρουμε παρακάτω (στο κεφάλαιο 2.2).

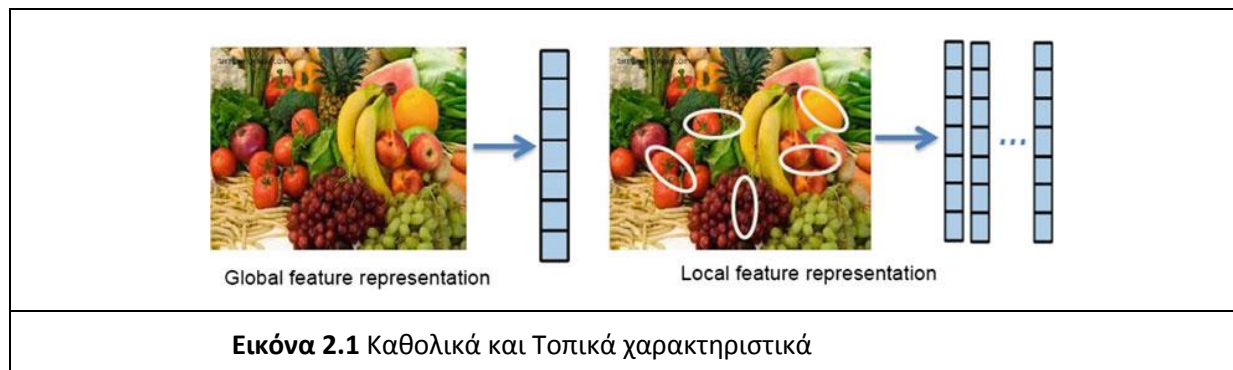
Μπορεί κανείς εύκολα να λογιστεί πως η αναζήτηση ασυνεχειών/ανομοιομορφιών πάνω σε μια εικόνα από πλέγμα δικτύων προσομοιάζει με την αναζήτηση ενός μοτίβου (pattern), του οποίου τα στοιχεία έχουν υποστεί παραμόρφωση. Η παραμόρφωση αυτή μπορεί να έχει διάφορες μορφές, είτε την περιστροφή και αντιστροφή του πλέγματος (rotation), είτε την μετατόπιση του στο χώρο (translation) είτε τέλος την στρέβλωσή του (stretching). Βάσει αυτής της πολυπλοκότητας η δομική ανάλυση (texture analysis) τέτοιων μοτίβων παίζει σημαντικό ρόλο σε εφαρμογές όπως είναι η αναζήτηση αντικειμένων (object detection), η αναγνώριση προτύπων (pattern recognition), ο εντοπισμός επιφανειών (surface detection) κτλπ.

Η προσέγγιση πάνω σε μια εικόνα με σκοπό την εξαγωγή των χαρακτηριστικών της όπως το χρώμα, η υφή ή το σχήμα, σύμφωνα με την εργασία «Image Features Detection, Description and Matching» [16], μπορεί να γίνει με δύο τρόπους: Είτε μέσω ολικής επεξεργασίας (global processing) όπου εστιάζουμε στα καθολικά χαρακτηριστικά της εικόνας, είτε μέσω τοπικής επεξεργασίας (local processing) όπου εστιάζουμε σε συγκεκριμένες περιοχές απ' τις οποίες αναζητούμε συγκεκριμένα τοπικά χαρακτηριστικά (key points).

Για παράδειγμα όταν θέλουμε να διακρίνουμε δύο εικόνες, με την πρώτη να περιλαμβάνει μια θάλασσα (μπλε) και την δεύτερη να περιλαμβάνει ένα δάσος (πράσινο), οι καθολικοί παράμετροι (global descriptors) όπως το χρώμα της εικόνας, θα δώσουν διαφορετικούς πίνακες (από pixel) σε κάθε περίπτω-

ση.

Απ' την άλλη πλευρά, ο κύριος στόχος των τοπικών χαρακτηριστικών είναι να αναπαριστά (represent) και να εξάγει συμπεράσματα για την εικόνα με βάση ορισμένες περιοχές, παραμένοντας αμετάβλητο σε αλλαγές όπως πχ. ο φωτισμός.



Η χρήση καθολικών (global) χαρακτηριστικών όπως το χρώμα, υφή (texture) κτλπ είναι εξαιρετικά χρήσιμα στην σύγκριση εικόνων, παρόλα αυτά παρουσιάζει αδυναμία λόγω ευαισθησίας στις απότομες μεταβολές. [17]

Ενώ απ' την άλλη μεριά τα τοπικά (local) χαρακτηριστικά είναι ιδανικά για την ανίχνευση και κατηγοριοποίηση αντικειμένων ή άλλων συμβάντων πάνω σε μια δοσμένη σκηνή της εικόνας, έχοντας παράλληλα χαμηλό υπολογιστικό κόστος [18] που είναι ιδανικό για πλατφόρμες όπως τα κινητά, τα οποία συνήθως περιέχουν δυνατότητες χαμηλής επεξεργασίας.

Παρόλα αυτά οι σύγχρονες μέθοδοι προτείνουν μια αλληλουχία ή και συνδυασμό global και local μεθόδων για την επίτευξη καλύτερων αποτελεσμάτων.

Σύμφωνα με τις εργασίες «Texture Analysis Methods – A Review [19] και «Texture image analysis and texture classification methods - A review» [20] τα κύρια ζητήματα που μας ενδιαφέρουν όταν έχουμε εικόνες από textures είναι τα εξής:

- Feature extraction
- Texture segmentation
- Texture classification
- Texture synthesis
- Texture shape extraction

Ενώ οι τέσσερις κύριοι μέθοδοι προσέγγισης μπορούν να κατηγοριοποιηθούν ως εξής:

- Structural methods (*Edge features, Morphological filter etc.*)
- Statistical methods (*Histogram, Co-occurrence matrix, 1st/2nd/Higher order statistics etc.*)
- Model – based methods (*AR model, Random fields models etc.*)
- Transform – based methods (*Gabor, Wavelet, Curvelet transform etc.*)

Αναλυτικότερα, οι δομικές μέθοδοι (structural methods) προσεγγίζουν μια εικόνα με βάση τη χωρική/γεωμετρική της διάταξη. Βασικά χαρακτηριστικά για τον εντοπισμό ενός μπορούν να θεωρηθούν τα pixels ή μια περιοχή από αυτά, ένα γεωμετρικό σχήμα όπως η ευθεία γραμμή, ο κύκλος κ.α.

Ως texture ορίζουμε το γεωγραφικό χώρο πάνω στην εικόνα που αντιστοιχεί σε ένα αντικείμενο με κοινά χαρακτηριστικά. Σε μια εικόνα με δίκτυα, ως texture μπορούμε να ορίσουμε την περιοχή του πλέγματος όπου οι οριζόντιες και οι κάθετες ευθείες σχηματίζουν ένα παραλληλόγραμμο ή τετράγωνο σχήμα πάνω στην εικόνα.

Σαν παράδειγμα μπορούμε να αναφέρουμε το φιλτράρισμα μιας εικόνας από κηλίδες (blobs) ή κέρματα, όπου ο εντοπισμός και διαχωρισμός των αντικειμένων γίνεται με βάση το γεωγραφικό/μετρικό σχήμα και τον αριθμό των pixels που περιλαμβάνει κάθε περιοχή.

Όσον αφορά τις μεθόδους στατιστικής προσέγγισης (statistical methods), αυτές εκτελούν μια σειρά στατιστικών υπολογισμών βασιζόμενες στην κατανομή της έντασης της φωτεινότητας των pixels.

Οι μέθοδοι model - based βασίζονται στην κατασκευή μοντέλων τα οποία δεν θα χρησιμοποιούνται μόνο στην περιγραφή ενός texture αλλά κυρίως στη σύνθεση νέων εικόνων από αυτό. Οι παράμετροι του μοντέλου καταγράφουν τις ιδιότητες του texture. Οι μέθοδοι αυτοί χρησιμοποιούνται κυρίως για την μελέτη φυσικών φαινομένων.

Τέλος οι μέθοδοι μετασχηματισμού (transform methods) κυρίως μετασχηματίζουν την εικόνα (μέσω συναρτήσεων) έτσι ώστε τα textures να είναι πιο ευδιάκριτα στον νέο χώρο. Οι πιο συνηθισμένες μέθοδοι υλοποίησης είναι ο Wavelet transform, ο Gabor κ.α.

Στην παρούσα διπλωματική η πρώτη μέθοδος υλοποίησης εφαρμόζει μια ολική (global) προσέγγιση πάνω στην εικόνα βασιζόμενη στην στατιστική ανάλυση της έντασης (statistical method) μέσω της χρήσης ιστογραμμάτων, ενώ η δεύτερη μέθοδος υλοποιεί μια πιο δομική προσέγγιση (structural method) εστιάζοντας στις τοπικές ακμές (local edges) του πλέγματος τα οποία σε συνδυασμό με την τεχνική του μετασχηματισμού Hough μας δίνουν το μοτίβο των ευθειών του δικτυού. Η μέθοδος του μετασχηματισμού Hough είναι μια τεχνική αναζήτησης αντικειμένων πάνω στην εικόνα, τα οποία περιγράφονται μέσα από κλειστές μαθηματικές σχέσεις.

Λεπτομερέστερη αναφορά τόσο σε θεωρητικό όσο και μεθοδολογικό πλαίσιο πάνω στις προαναφερθείσες έννοιες γίνεται στα κεφάλαιο 3 και 4.

## 2.2 Συναφείς Εργασίες

Πολλές ενδιαφέρουσες εργασίες έχουν αναπτυχθεί πάνω στην αναζήτηση όμοιων και ανόμοιων μοτίβων (regular/irregular pattern).

Μια βασική εργασία είναι η «*Detecting, localizing and grouping repeated scene elements from an image*» [21] η οποία έχει σαν σκοπό την ομαδοποίηση όμοιων αντικειμένων πάνω στην εικόνα. Αρχικά αναπτύσσεται ένας αλγόριθμος ανίχνευσης και εντοπισμού ενός δειγματικού στοιχείου (sample element). Στη συνέχεια αναζητούνται γειτνιάζοντα στοιχεία τα οποία παρά τους μετασχηματισμούς που μπορεί να έχουν υποστεί προσομοιάζουν με το αρχικό αντικείμενο (από την λήψη της κάμερας, από το θόρυβο κτλπ.). Ενώ τέλος ομαδοποιούνται τα στοιχεία του επαναλαμβανόμενου μοτίβου και εξάγονται από την εικόνα.

Συναφής εργασία με την προηγούμενη είναι και η «*Periodic pattern of texture analysis and synthesis based on texels distribution*» [22]. Εδώ όχι μόνο ανιχνεύονται και κατηγοριοποιούνται τα textures της εικόνας σε τοπικό επίπεδο αλλά χρησιμοποιούνται περαιτέρω τεχνικές μοντελοποίησης (πχ texture modeling όπως η synthesis) με σκοπό τη δημιουργία νέων παρόμοιων περιοδικών δομών.

Άλλες μέθοδοι εξάγουν το μοτίβο των textures μέσα από την ολική περιοδικότητα της εικόνας (global periodicity) σε συνδυασμό με την μέθοδο τριγωνοποίησης Delaunay (Delaunay triangulation) όπως στην εργασία «*Texel Based Regular and Near-Regular Texture Characterization*» [23]. Αρχικά εντοπίζεται η θέση και το σχήμα ενός δειγματικού texture με την ίδια μεθοδολογία που χρησιμοποιήθηκε στην εργασία [21] ενώ στη συνέχεια κατηγοριοποιείται κιόλας.

Στην συνέχεια εντοπίζονται γειτονικά textures τα οποία κατηγοριοποιούνται σε σχέση με το αρχικό (πχ αν ένα texture είναι δεξιόστροφο ή αριστερόστροφο). Τέλος εξάγονται τα γειτνιάζοντα αντικείμενα καθώς και η περιοδικότητά τους. Η παρούσα εργασία προσφέρει παραπάνω στοιχεία σε σχέση με την προαναφερθείσα, μιας και δίνει την δυνατότητα να γνωρίζουμε επιπλέον την απόσταση που έχουν μεταξύ τους τα textures καθώς και την κατεύθυνσή τους (μέσω της κατηγοριοποίησης).

Η εργασία «*Near-Regular Texture Analysis and Manipulation*» [24] χρησιμοποιεί μια πιο τοπική προσέγγιση, επικεντρώνοντας περισσότερο στα γεωμετρικά χαρακτηριστικά, στο χρώμα και την ένταση συγκεκριμένων περιοχών σε συνδυασμό με την περιοδικότητα με σκοπό να εξάγει παραμετρικά μοντέλα. Με αυτό τον τρόπο μπορεί στη συνέχεια παίρνοντας ένα texture να συνθέσει ολόκληρες περιοχές ή ακόμα και να τις χειραγωγήσει (manipulate) κατά το δοκούν. Για παράδειγμα παίρνοντας ένα texture μπορεί να δημιουργήσει μια νέα εικόνα ή ένα μοτίβο με όμοια χαρακτηριστικά.

Μια από τις πιο ολοκληρωμένες δουλειές που βρήκαμε πάνω στην αναζήτηση ασυνεχειών σε δίχτυα είναι η εργασία «*An adaptive method of damage detection for fishing nets based on image processing technology*» [25]. Οι συγγραφείς κατηγοριοποιούν τους τρόπους με τους οποίους μπορεί να εντοπιστεί μια ασυνέχεια πάνω στο πλέγμα σε τρεις ομάδες: Πρωτίστως μέσω ενσωματωμένων συστημάτων (embedded detection), διαφορετικά μέσω ηχητικής ανίχνευσης (sonar detection) είτε τέλος μέσω επεξεργασίας της εικόνας (image analysis).

Θεωρούν πως η τοποθέτηση σύρματος ή αισθητήρα (embedded method) πάνω στο δίχτυ δεν θεωρείται καλή προσέγγιση, μιας και το δίχτυ της ιχθυοκαλλιέργειας επιφορτίζεται με περισσότερο βάρος το οποίο σε βάθος χρόνου προκαλεί φθορά. Ο δεύτερος τρόπος προσέγγισης (sonar detection) απορρίπτεται για λόγους κόστους σύμφωνα με τους συγγραφείς, καταλήγοντας έτσι δια της αφαιρετικής στην τρίτη μέθοδο (image analysis) ως καταλληλότερη. Στη συνέχεια οι συγγραφείς προτείνουν στην εργασία τους δύο τρόπους προσέγγισης.



Η πρώτη βασίζεται σε ένα νευρωνικό δίκτυο που έχει σαν σκοπό την αναζήτηση των κόμβων του πλέγματος (netting node analysis). Πιο συγκεκριμένα, δημιουργούν ένα παράθυρο πάνω στην εικόνα στις αποχρώσεις του γκρι, το οποίο την διατρέχει και εντοπίζει την ένταση που υπάρχει σε αυτή. Στη συνέχεια, μέσω της διακύμανσης της έντασης και της μεθόδου Harris corner, ανάλογα με το βάρος του σημείου, ορίζουν αν κάτι αποτελεί γωνία ή όχι.

Η λογική της μεθόδου είναι ότι για κάθε σημείο ελέγχεται η ένταση για μικρές μετακινήσεις γύρω από αυτό. Ως γωνίες χαρακτηρίζονται τα σημεία εκείνα στα οποία οι μεταβολές της τιμής της συνάρτησης φωτεινότητας είναι μεγάλες. Η μεταβολή αυτή εκφράζεται μέσω της συνάρτησης  $E$  που δίνεται από τη σχέση :

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

Ο όρος  $w(x, y)$  εκφράζει το παράθυρο εντός του οποίου ελέγχεται η μεταβολή αυτή, ενώ  $u$  και  $v$  είναι η οριζόντια και κάθετη μετακίνηση αντίστοιχα γύρω από το σημείο  $x, y$ .

Αντικείμενο αναζήτησης είναι οι ευθείες του πλέγματος, ενώ ο σκοπός τους είναι ο εντοπισμός των κόμβων που δημιουργούν τα δίκτυα. Θεωρούν ότι η κατανομή των εντοπισμένων γωνιών στα συγκεκριμένα σημεία θα έχει μια συγκεκριμένη μορφή (τετράγωνη), ενώ στα σημεία όπου είναι κομμένο το δίκτυο η κατανομή των ακμών θα δημιουργεί έναν κύκλο, λόγω ασυνέχειας.

Το βασικό μειονέκτημα της μεθόδου είναι η αδυναμία εντοπισμού γωνιών σε εικόνες με χαμηλή φωτεινότητα.

Στη δεύτερη μέθοδο οι συγγραφείς αλλάζουν τρόπο προσέγγισης και βάζουν στο επίκεντρο της έρευνας τους όχι τις ευθείες του πλέγματος αλλά τις τρύπες που υπάρχουν σε αυτό. Μετατρέπουν την εικόνα σε ασπρόμαυρη (binary) και την κατακερματίζουν σε παράθυρα. Ανάλογα με τις τρύπες που υπάρχουν στο πλέγμα σε συνδυασμό με τα τοπικά σημεία (local peaks) που εντόπισαν στην προηγούμενη μέθοδο (Harris corner), συγκρίνουν την ένταση της εκάστοτε περιοχής μέσω ιστογραμμάτων.

Ο προαναφερθείς αλγόριθμος (εργασία [25]) εντοπισμού ασυνεχειών στα δίκτυα χρησιμοποιείται και στην εργασία «*Overview of the application of computer vision technology in fish farming*» [26] αν και δεν είναι αυτός κάθε αυτός ο βασικός σκοπός της.

Μια επιπλέον ολοκληρωμένη εργασία είναι η «*Automatic Inspection of Cage Integrity with Underwater Vehicle*». [27] Ο συγγραφέας έχει ως αντικείμενο αναζήτησης τις ευθείες του πλέγματος, τις οποίες εντοπίζει μέσω του μετασχηματισμού Hough, ενώ στη συνέχεια εφαρμόζει μια αναζήτηση κατά βάθος (DFS – Depth First Search) πάνω στο εντοπισμένο πλέγμα. Πιο συγκεκριμένα, αρχικά στο κομμάτι της προ επεξεργασίας του πλέγματος, ο συγγραφέας αφού φιλτράρει ολόκληρη την εικόνα μέσα από τα κανάλια RGB και HSV καταλήγει στο κόκκινο κανάλι (Red Channel) ως καταλληλότερο για το διαχωρισμό προσκηνίου (foreground) και φόντου (background). Στη συνέχεια καταμερίζει την εικόνα σε υπο-τμήματα (sub blocks) και αναζητεί εκεί το κατάλληλο κατώφλι κάθε φορά. Για την εύρεση κατάλληλου κατωφλίου χρησιμοποιεί πρώτα ένα χαμηλοπερατό φίλτρο (low-pass Gaussian filter) ώστε να εξομαλύνει τις ακμές της εικόνας και στη συνέχεια ένα Laplacian όπου αναζητεί τα σημεία της πρώτης παραγωγής που είναι πάνω απ' το μηδέν. Τέλος αναζητεί τις ακμές με τη μέθοδο Canny (Canny edge detector) σε συνδυασμό με τον μετασχηματισμό Hough ώστε να εντοπίσει τις γραμμές του πλέγματος του δικτύου. Η αναζήτηση των ασυνεχειών στην ασπρόμαυρη (binary) εικόνα γίνεται μέσω ενός αλγορίθμου DFS (depth first search). Εκεί αναζητεί τα άσπρα pixel στον κάθετο και στον οριζόντιο άξονα τα οποία κείτονται πάνω στις εντοπισμένες γραμμές, έως ότου φτάσει σε κάποια ασυνέχεια. Κάθε γραμμή αποθηκεύεται σε πίνακα και στο τέλος της διαδικασίας συγκρίνονται οι αποστάσεις μεταξύ των γραμμών.

Αν υπάρχει ασυνέχεια σε κάποιο σημείο η απόσταση των γραμμών στο συγκεκριμένο σημείο θα είναι μεγαλύτερο από το υπόλοιπο μέσο όρο.

Μια ακόμη ολοκληρωμένη εργασία είναι η «*Damage detection in fish farming nets using video analysis*» [28]. Οι συγγραφείς στην διαδικασία της προ-επεξεργασίας (pre-process) βρίσκουν το κατάλληλο κατώφλι χρησιμοποιώντας το Gaussian filter ενώ στη συνέχεια «σκελετοποιούν» τις γραμμές του πλέγματος (skeletonization method) συρρικνώνοντάς τις στο ελάχιστο. Σκοπός είναι να μετατρέψουν κάθε ευθεία σε γραμμή πλάτους 1-pixel ώστε να μειωθεί ο θόρυβος (fouling) που μπορεί να υπάρχει πάνω στο δίχτυ λόγω της υποθαλάσσιας χλωρίδας. Στη συνέχεια χρησιμοποιούν δύο κατ' ουσία μεθόδους ανίχνευσης τρυπών πάνω στα δίχτυα. Η πρώτη ονομάζεται Knot Density Analysis (KDA) όπου αρχικά αναζητεί τους κόμβους του πλέγματος (πάνω στην skeletonized εικόνα), ενώ στη συνέχεια αναζητούν την πυκνότητα ανά περιοχή καθώς και την μέση απόσταση μεταξύ των σημείων. Σε περίπτωση εντοπισμού περιοχής με παρεκκλίνοντα δεδομένα, αυτομάτως χαρακτηρίζεται ως ασυνέχεια. Η δεύτερη μέθοδος ονομάζεται Large Region Analysis (LRA) όπου αναζητούμε στην skeletonized εικόνα τα περιγράμματα (contours) του πλέγματος. Χωρίζεται έτσι όλο το δίχτυ σε περιοχές, όπου στη συνέχεια συγκρίνουμε το μέγεθος κάθε μιας με τον συνολικό μέσο όρο της εικόνας. Σε γενικές περιπτώσεις και οι δύο μέθοδοι λειτουργούν ικανοποιητικά. Βασικό μειονέκτημα της πρώτης μεθόδου (KDA) είναι ότι παραβλέπει πολλές τρύπες με μικρό μέγεθος, ενώ της δεύτερης (LRA) είναι ότι ο εντοπισμός των περιοχών εν πολλύς βασίζεται στην διαδικασία της προ-επεξεργασίας. Αν η εικόνα έχει χαμηλό φωτισμό ή περιέχει με θόρυβο, η «σκελετοποίηση» μπορεί να δώσει λανθασμένα περιγράμματα.

Τέλος ένας συχνός τρόπος ανίχνευσης των υποθαλάσσιων ιχθυοκαλλιιεργειών είναι μέσω νευρωνικών δικτύων. Ένα χαρακτηριστικό παράδειγμα τέτοια μεθόδου είναι το YOLO (You Only Look Once) από την εργασία «*You Only Look Once: Unified, Real-Time Object Detection*». [29] Στην παραπάνω εργασία η αναγνώριση αντικειμένων υλοποιείται μέσα από μια επαναληπτική διαδικασία καταμερισμού της εικόνας σε bounding boxes και κατηγοριοποίησης τους. Στη συνέχεια το νευρωνικό δίκτυο προβλέπει, κατηγοριοποιεί και ομαδοποιεί τα bounding boxes ανάλογα με τις ομοιότητά τους. Το τελικό αποτέλεσμα είναι ο εντοπισμός των αντικειμένων.

Δύο εργασίες: η «*Omnidirectional Surface Vehicle for Fish Cage Inspection*» [30] και η «*Underwater Fishnet and Hole Detection using Miniature Underwater Robot*» [31] που ασχολούνται με την υποθαλάσσια αναζήτηση δικτύων χρησιμοποιούν αυτούσιο τον παραπάνω νευρωνικό δίκτυο με ικανοποιητικά αποτελέσματα. Ενδεχόμενο μειονέκτημα των παραπάνω εργασιών είναι πως οι εικόνες-παραδείγματα που χρησιμοποιούν είναι παρμένες μέσα σε πισίνα. Και άρα ο αλγόριθμος δεν έχει χρησιμοποιηθεί σε πραγματικές υποθαλάσσιες συνθήκες.

Με δεδομένο ότι η εισχώρηση και η εφαρμογή του computer vision στον υποθαλάσσιο χώρο είναι μια σχετικά καινούργια προσπάθεια, οι παραπάνω εργασίες μας βοήθησαν αρκετά ώστε να αντιληφθούμε σφαιρικά το πρόβλημα της υποθαλάσσιας επεξεργασίας και τα προβλήματα που ενδεχομένως να συναντούσαμε.

Οι πρώτες εργασίες που αναφέραμε έπαιξαν καθοριστικό ρόλο στην κατεύθυνση που θα έπρεπε να κινηθούμε πάνω στο χώρο της αναγνώρισης ανόμοιων προτύπων (irregular recognition), ενώ οι ολοκληρωμένες εργασίες μας βοήθησαν να προσεγγίσουμε το πρόβλημα πιο επιστάμενα.

Οι δύο μέθοδοι που υλοποιήσαμε προήλθαν ως ιδέες από τις δύο παραπάνω ολοκληρωμένες εργασίες που παρουσιάσαμε. Η πρώτη με τη χρησιμοποίηση του μετασχηματισμού Hough και την αναζήτηση των κατευθύνσεων του πλέγματος και η δεύτερη μια πιο ολική (global) προσέγγιση, μέσω της έντασης μέρους ή όλου του φάσματος της εικόνας.



## Κεφάλαιο 3 : Θεωρητικό υπόβαθρο

### 3.1 Εισαγωγή

Για να μπορέσουν να γίνουν πιο κατανοητές οι μέθοδοι που υλοποιήσαμε, θα κάνουμε μια εισαγωγή στο θεωρητικό υπόβαθρο της εκάστοτε διαδικασίας. Θα προσπαθήσουμε ευσύνοπτα να εξηγήσουμε ποια ήταν η διαδικασία της προ-επεξεργασίας των εικόνων (pre-process), πως εντοπίζονται οι ακμές στις εικόνες, πως αυτές συνδέονται με τα όρια των περιοχών που καθορίζουν τα αντικείμενα και τέλος θα μιλήσουμε για τον μετασχηματισμό Hough και πως αυτός λειτουργεί.

Αρχικά κατάτμηση ονομάζουμε την διαδικασία κατά την οποία διαιρούμε την εικόνα σε ομοιόμορφες περιοχές, οι οποίες ιδανικά αντιστοιχούν σε κάποιο αντικείμενο. Το επίπεδο της λεπτομέρειας στο οποίο πραγματοποιείται αυτή η υποδιαίρεση, εξαρτάται από το πρόβλημα που θέλουμε να επιλύσουμε.

Η κατάτμηση αποτελεί σχεδόν πάντα το πρώτο βήμα για την ανάλυση μιας εικόνας και μπορεί να επιτευχθεί με τον εντοπισμό είτε των σημείων ασυνέχειας της εικόνας (όπως η απότομη μεταβολή της φωτεινότητας ή του χρώματος) είτε μέσω της ομαδοποίησης των ομοιόμορφων τιμών κάποιων pixel.

Τα σύνολα σημείων ασυνέχειας τα οποία καθορίζουν το όριο διαχωρισμού δύο περιοχών τα ονομάζουμε ακμές. Ο συνηθέστερος τρόπος ανίχνευσης αυτών των τοπικών μεταβολών της έντασης μπορεί να πραγματοποιηθεί υπολογίζοντας την κλίση της πρώτης ή δεύτερης τάξεως παραγώγου και κατόπιν την εφαρμογή κατωφλίσωσης για την επιλογή των μεγίστων.

Η επιλογή κατωφλίου είναι χρήσιμη για να αποφασίσουμε αν ένα pixel είναι μέρος μιας ακμής ή του φόντου της εικόνας. Μια απ' τις πιο γνωστές μεθόδους κατωφλίσωσης είναι το Otsu threshold. Άλλες μέθοδοι εύρεσης ακμών είναι η Laplacian, η Laplacian of Gaussian (LoG), η pdf, η εντροπία κα.

### 3.2 Μέθοδος κατωφλίσωσης

Η μέθοδος κατωφλίσωσης που χρησιμοποιήσαμε στις εικόνες μας είναι η τεχνική Otsu [41].

Η μέθοδος αυτή βασίζεται στην καθολική (global) επεξεργασία του ιστογράμματος της εικόνας και στον προσδιορισμό κατωφλίου βάσει του κριτηρίου της μεγιστοποίησης του διαχωρισμού μεταξύ των περιοχών του φόντου (background pixels) από αυτά του προσκηνίου (foreground pixels). Στόχος είναι η ελαχιστοποίηση της διασποράς του πεδίου μεταξύ αυτών των δύο περιοχών.

Η μέθοδος αυτή επιλέχθηκε, καταρχήν για τον λόγο ότι οι εικόνες που εξετάσαμε ήταν σχετικά απλές και τα αντικείμενα μελέτης ευδιάκριτα, χωρίς τεράστιες εναλλαγές στην ένταση αλλά κυρίως χωρίς εμβόλιμα αντικείμενα πάνω σε αυτές (όπως ψάρια). Επομένως μια ολική (global) μέθοδος μας ικανοποιεί σε πρώτο στάδιο, στο να διαχωρίσουμε το πλέγμα της εικόνας. Σε εικόνες χωρίς μεγάλες διακυμάνσεις στο ιστόγραμμα, το κατώφλι εντοπίζεται στην «κοιλάδα» (valley) ανάμεσα στις δύο κορυφές. Τέ-

λος η μέθοδο Otsu μας δίνει την ευχέρεια να εντοπίζει αυτόματα το όριο κατωφλίσωσης της εικόνας, κάτι πολύ σημαντικό, αν το όριο αυτό ικανοποιεί τις απαιτήσεις που έχουμε στον απλό διαχωρισμό εικόνας και αντικειμένου.

Για εικόνες όπως αυτές που χρησιμοποιήσαμε στα παραδείγματα μας η εν λόγω μέθοδος, μας έδωσε ικανοποιητικά αποτελέσματα με χαμηλό υπολογιστικό κόστος. Βασικό μειονέκτημα της μεθόδου είναι ευαισθησία στον θόρυβο και η αδυναμία να διαχωρίσει περιοχές και αντικείμενα χωρίς να αλλοιώσει το αποτέλεσμα σε μια τέτοια περίπτωση. Δεδομένου του γεγονότος ότι η παρούσα διπλωματική επεξεργάζεται υποθαλάσσιες εικόνες με χαμηλή φωτεινότητα, σκέδαση, κακή εστίαση, θόρυβο απ' την υποθαλάσσια χλωρίδα κλπ. μια απλή εφαρμογή της τεχνικής Otsu ως καθολικής μεθόδου κατωφλίσωσης της εικόνας δεν μπορεί να μας εξάγει εικόνες με ικανοποιητικό αποτέλεσμα προς επεξεργασία.

Ανεξαρτήτως του παραπάνω προβλήματος η μεθοδολογία μας δεν επικεντρώθηκε τόσο στην επεξεργασία και αποκατάσταση της αρχικής εικόνας αλλά στην αναγνώριση των δομών της, την εξαγωγή του μοτίβου και την ανίχνευση ασυνεχειών πάνω σε αυτό. Για τον λόγω αυτό επιλέξαμε εικόνες με σχετικά ομοιόμορφη κατανομή φωτεινότητας ικανές προς επεξεργασία με απλά εργαλεία.

Πάραυτα μια ενδεχομένως προτεινόμενη λύση στο παραπάνω πρόβλημα, θα ήταν η εφαρμογή ομομορφικών φίλτρων (homomorphic filtering) σαν πρώτο βήμα με σκοπό την βελτίωση του contrast και του sharpness της εικόνας πριν περάσουμε στην διαδικασία της κατωφλίσωσης [32].

## 3.2 Global processing προσέγγιση

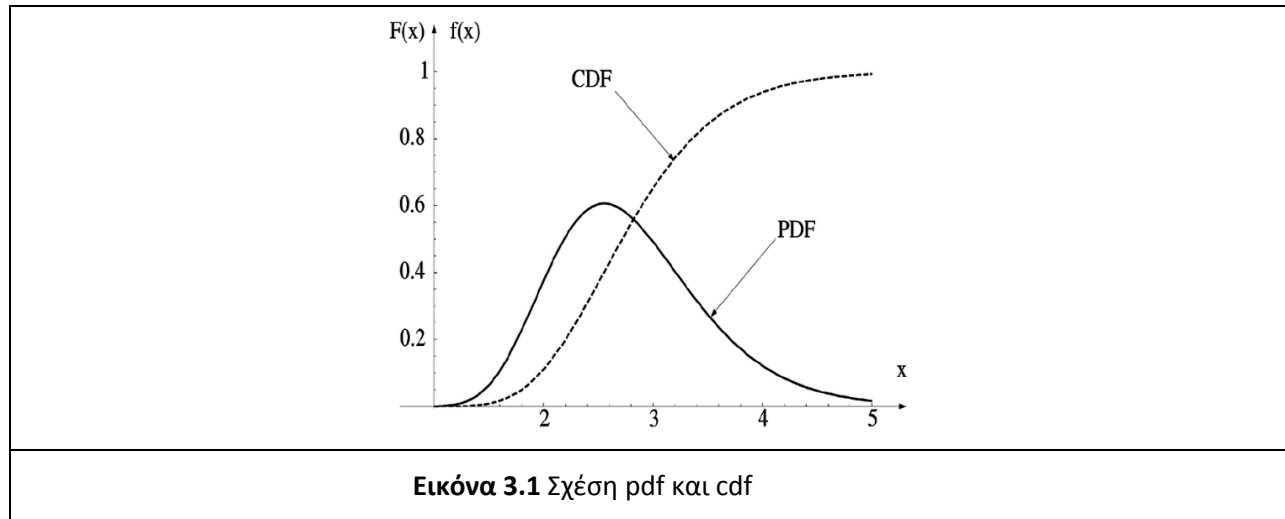
Στην πρώτη μέθοδο (global processing) για να μπορέσουμε να ξεπεράσουμε κάποιες εσφαλμένες ανιχνεύσεις (false detection) του αλγορίθμου μας, χρησιμοποιήσαμε ως μέθοδο εντοπισμού του κατάλληλου κατωφλίου το γόνατο (knee-point) της αθροιστικής συνάρτησης κατανομής των τιμών (cdf).

Για να γίνει πιο κατανοητή η υλοποίηση στο επόμενο κεφάλαιο θα αναπτύξουμε εισαγωγικά τον τρόπο λειτουργίας μιας τέτοιας μεθόδου.

Αφού έχουμε κατακερματίσει την εικόνα σε blocks και στη συνέχεια σε υπο-μπλόκς (sub blocks), βρίσκουμε την κατανομή της έντασης σε κάθε sub block. Υπολογίζουμε και απεικονίζουμε, δηλαδή, την ποσότητα των άσπρων pixel (τιμή pixel ίση με 1, δεδομένου ότι έχουμε αντιστρέψει την binary εικόνα) που περιέχει κάθε παράθυρο του sub block. Η κατανομή έχει διακριτές τιμές οι οποίες απεικονίζονται μέσω ιστογραμμάτων.

Έστω η συνάρτηση κατανομής  $F_X(x) = P(X \leq x)$ , όπου  $X$  μια πραγματική τυχαία μεταβλητή. Αυτή η συνάρτηση δίνει την πιθανότητα το  $X$  να πάρει τιμές μικρότερες ή ίσες του  $x$ .

Η αθροιστική κατανομή είναι μια αύξουσα συνάρτηση, στην παρακάτω εικόνα απεικονίζεται σε σχέση με μια συνάρτηση κανονικής κατανομής.



Αφού βρούμε την αθροιστική κατανομή για κάθε υπο-παράθυρο (sub block) της εικόνας, αναζητούμε το σημείο απότομης ακμής πάνω στην αθροιστική κατανομή (cdf). Το σημείο αυτό λέγεται γόνατο ή αγκώνας της κατανομής (knee/elbow point) [33] και αποτελεί το κατώφλι, κάτω απ' το οποίο δηλώνουμε εάν ένα υπο-παράθυρο (sub block) περιέχει ασυνέχεια. Αν η τιμή της έντασης μέσα σε αυτό βρίσκεται κάτω από την τιμή του κατωφλίου τότε το σημείο αυτό χαρακτηρίζεται ως τρύπα πάνω στην εικόνα.

### 3.3 Local processing προσέγγιση

Η διαδικασία ανίχνευσης ακμών ταυτοποιεί σύνολα σημείων τα οποία ανήκουν πάνω στις ακμές της εικόνας. Στην πράξη όμως, τα εικονοστοιχεία της εικόνας, σπάνια χαρακτηρίζουν πλήρως τις ακμές, εξαιτίας του θορύβου, της ανομοιόμορφης φωτεινότητας κτλπ. Γι' αυτούς τους λόγους, η ανίχνευση ακμών συνήθως ακολουθείται από την εφαρμογή αλγορίθμων σύνδεσης, οι οποίοι έχουν σχεδιαστεί για να συνδέουν τα εικονοστοιχεία των ακμών σε πιο συνεκτικά (robust) περιγράμματα ακμών ή περιοχών.

Υπάρχουν τρεις τεχνικές για τη συνένωση ακμών:

- Η τοπική επεξεργασία
- Η επεξεργασία βασισμένη σε περιοχές
- Η ολική επεξεργασία χρησιμοποιώντας Hough transform

Οι δύο πρώτες τεχνικές βρίσκουν εφαρμογή σε περιπτώσεις που διαθέτουμε μερική γνώση για τα εικονοστοιχεία που ανήκουν στα επιμέρους αντικείμενα της εικόνας. Σε μη δομημένα περιβάλλοντα όπου το μόνο που έχουμε είναι μια εικόνα ακμών και απουσιάζει πλήρως η γνώση μας για το που βρίσκονται τα αντικείμενα ενδιαφέροντος, χρησιμοποιούμε τεχνικές ολικής επεξεργασίας.

Ο μετασχηματισμός Hough είναι μια τέτοια τεχνική όπου με βάση τα pixel των ακμών αναζητούμε αντικείμενα τα οποία περιγράφονται από κλειστές μαθηματικές σχέσεις (πχ. γραμμές, κύκλοι, ελλείψεις).

Ο μετασχηματισμός Hough αποτελεί μια τεχνική ομαδοποίησης ακμών της εικόνας μέσα από μια διαδικασία αθροιστικής ψηφοφορίας (accumulation procedure). Αποτέλεσμα της οποίας είναι η εξαγωγή μοτίβων μέσα από τα σύνολα αυτών των σημείων (καμπύλες, ευθείες, κύκλοι κλπ) [34].

Η τεχνική αυτή πρωτοανακαλύφθηκε από τον Paul Hough το 1962 και σκοπό είχε την ανίχνευση της κίνησης σωματιδίων υψηλής ενέργειας. Η επέκταση του βασικού μετασχηματισμού και η εφαρμογή του τα επόμενα χρόνια στην επιστήμη της υπολογιστικής όρασης έδωσε την δυνατότητα στην ανίχνευση αφηρημένων αντικειμένων όπως είναι ο κύκλος και οι ελλείψεις και οι ευθείες μέσα από σύνολα σημείων που μπορεί να υπάρχουν στην εικόνα [35].

Η κατανόηση της λειτουργίας του μετασχηματισμού Hough αποτελεί σημαντικό στάδιο στην κατανόηση και μοντελοποίηση όλης της μεθοδολογίας μας.

Ξεκινώντας λοιπόν την περιγραφή του αλγορίθμου, ορίζουμε ως χώρο εικόνας (image space) τον χώρο εκείνον που περιγράφεται από ένα σύνολο σημείων  $(x,y)$ . Το κάθε σημείο του χώρου αυτού αποτελεί ένα ζεύγος δύο πραγματικών αριθμών και αναπαρίσταιται από μια κουκίδα (edge) πάνω στην εικόνα.

Εάν θέλουμε να μοντελοποιήσουμε μια γραμμή στο χώρο αυτό, τότε μπορούμε να την αναπαραστήσουμε ως ένα σύνολο σημείων που επαληθεύουν την παρακάτω μαθηματική εξίσωση:

$$y-m \cdot x -b =0 \Leftrightarrow y=m \cdot x+b \quad (1)$$

Όπου το  $m$  αποτελεί την κλίση της ευθείας:

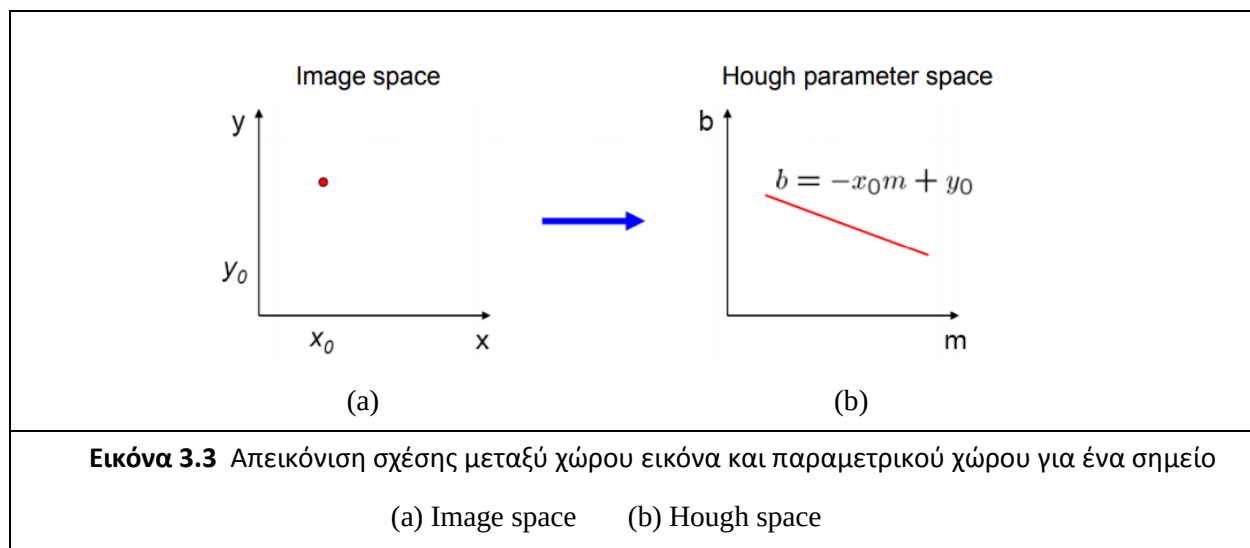
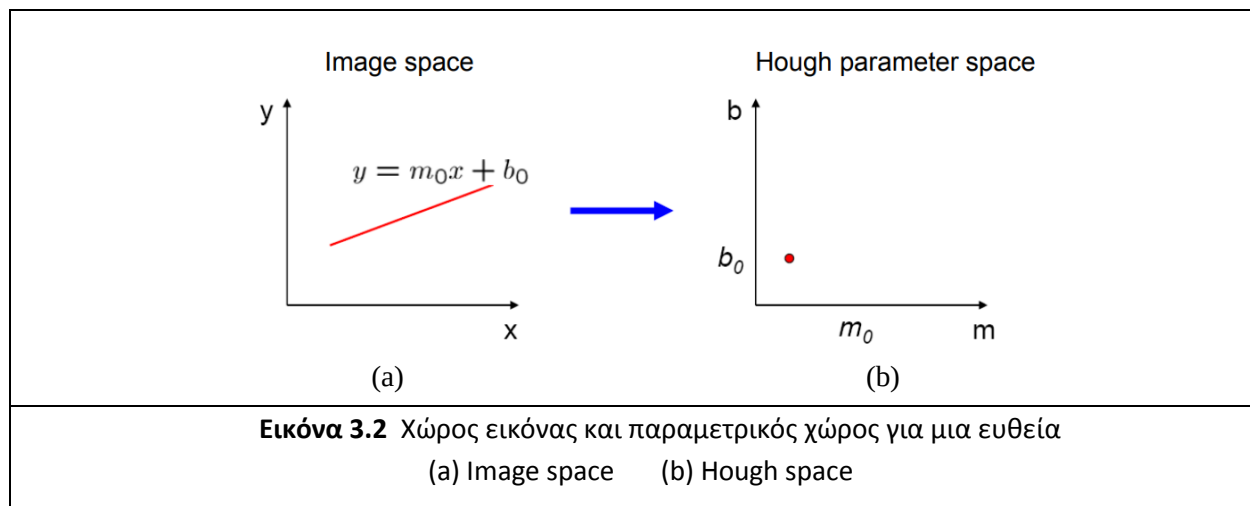
$$m=\frac{\Delta y}{\Delta x}$$

ενώ το  $b$  μας δίνει το σημείο το οποίο τέμνεται με τον γγ' άξονα.

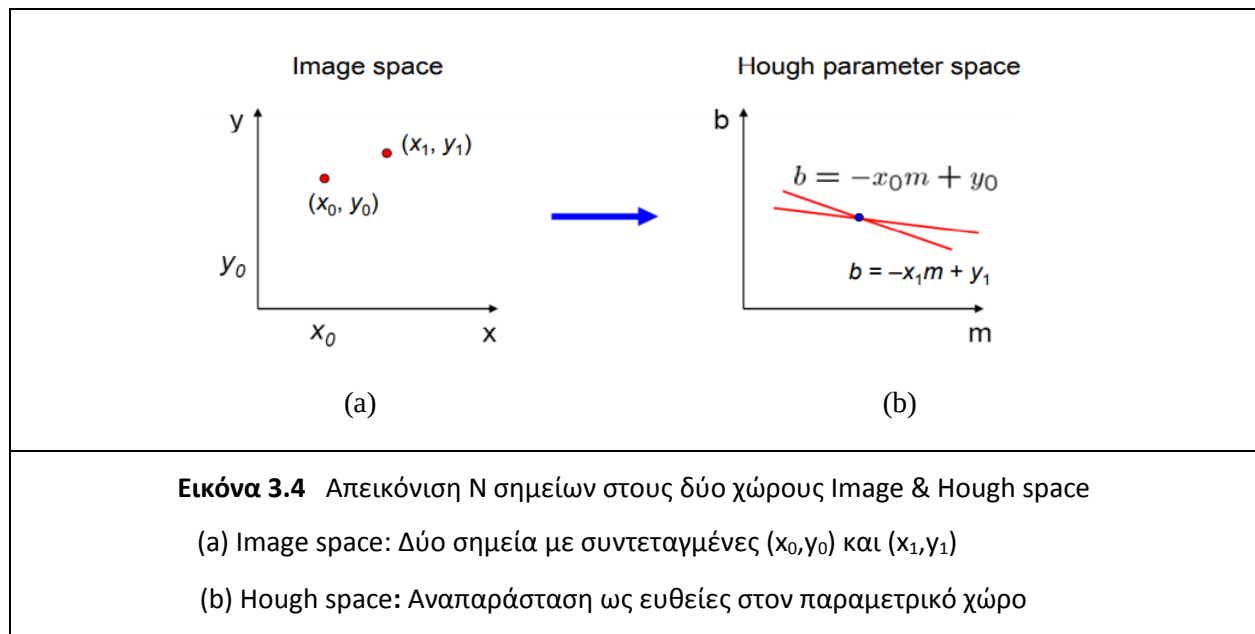
Με βάση την εξίσωση (1) μπορούμε να δημιουργήσουμε ένα νέο χώρο στο σύνολο των πραγματικών αριθμών ο οποίος ονομάζεται χώρος παραμέτρων (ή αλλιώς Hough space).

Επι της ουσίας εφαρμόζουμε μια διαφορετικού τύπου αναπαράσταση της εικόνας μας. Όπου για σύστημα συντεταγμένων θα έχουμε τις παραμέτρους  $(m,b)$ , δηλαδή τον συντελεστή διεύθυνσης και την σταθερά μιας ευθείας, αντί για τους συντελεστές  $(x,y)$ .

Με βάση τα παραπάνω κάθε μια ευθεία στον χώρο της εικόνας αντιστοιχίζεται με ένα σημείο στον χώρο των παραμέτρων και το ανάποδο, ενώ ένα σημείο στον παραμετρικό χώρο  $(m_0, b_0)$  αντιστοιχεί σε μια ευθεία στον image space.



Επομένως μια ευθεία που τέμνει νοητά  $N$  σημεία στο χώρο της εικόνας, θα απεικονίζεται στον παραμετρικό χώρο ως διασταύρωση  $N$  γραμμών [36].



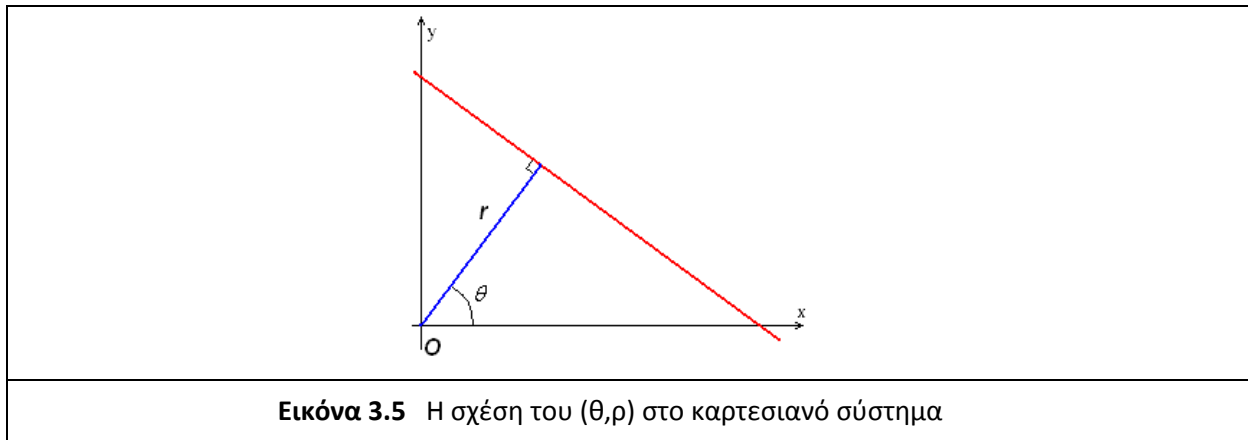
Το πρόβλημα με την ως άνω τεχνική είναι ότι παρουσιάζει αδυναμία στις οριακές τιμές. Εάν στην εικόνα υφίσταται ένα σημείο το οποίο τέμνεται από μια ευθεία με κλίση  $90^\circ$  (παράλληλη στον  $yy'$ ) αυτό θα σημαίνει ότι ο συντελεστής διεύθυνσης της  $m$  της ευθείας θα τείνει προς το άπειρο, κάτι που μας καθιστά αδύνατη την εύρεση και απεικόνιση του στον παραμετρικό χώρο.

Το πρόβλημα είναι υπαρκτό και πολύ σημαντικό, μιας και δεν ξέρουμε πριν την εκτέλεση του μετασχηματισμού αν υπάρχουν ή όχι κάθετες ευθείες στην εικόνα μας.

Η λύση σε αυτό το πρόβλημα ήρθε το 1972 από τους R. Duda και P. Hart οι οποίοι αντικατέστησαν τις παραμέτρους που χρησιμοποιούνταν για τον προσδιορισμό μιας ευθείας, από  $(m, b)$  σε  $(\rho, \theta)$ .

Όπου  $\rho$  είναι μια κάθετη ευθεία, και δηλώνει την ελάχιστη απόσταση μιας ευθείας από την αρχή των αξόνων, ενώ  $\theta$  είναι η γωνία που σχηματίζεται ανάμεσα στο  $\rho$  και τον άξονα των  $x$ . Ο μετασχηματισμός των Duda & Hart ονομάζεται αλλιώς και γενικευμένος μετασχηματισμός Hough (generalized Hough transform) [37].

Η σχέση  $(\theta, \rho)$  απεικονίζεται στην παρακάτω εικόνα [38].



Επομένως μια ευθεία σύμφωνα με τον γενικευμένο μετασχηματισμό Hough μπορεί να γραφτεί και ως εξής:

$$m = -\frac{\cos\theta}{\sin\theta}$$

$$b = \frac{r}{\sin\theta}$$

$$y = m \cdot x + b \Leftrightarrow y = -\frac{\cos\theta}{\sin\theta} \cdot x + \frac{\rho}{\sin\theta} \Leftrightarrow \rho = x \cdot \cos\theta + y \cdot \sin\theta$$

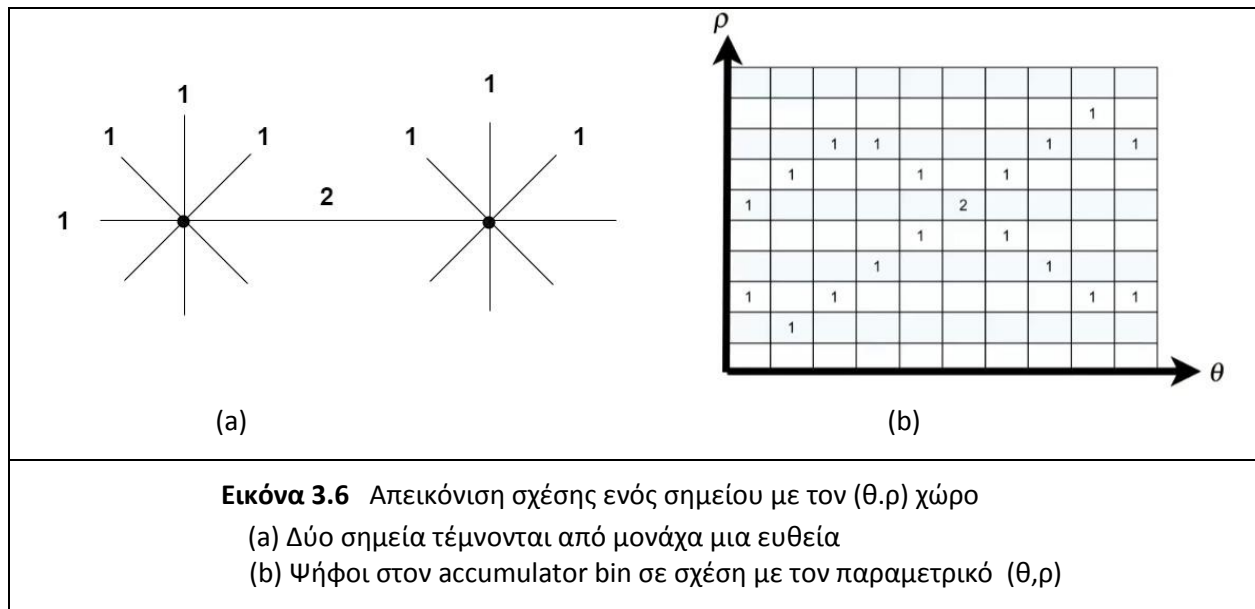
$$\rho = x \cdot \cos\theta + y \cdot \sin\theta \quad (2) \quad \theta \in [0, 2\pi) \quad \text{και} \quad \rho \in \mathbf{R}$$

Μεταβαίνουμε δηλαδή απ' το  $(x, y)$  σύστημα συντεταγμένων της εικόνας μας σε ένα άλλο σύστημα απεικόνισης με άξονες  $(\theta, \rho)$ . Με το  $\theta$  να παίρνει τιμές στο εύρος του  $[0 : 360)$  ενώ το  $\rho$  να έχει καθορισμένη τιμή, και ορίζεται από το μέγεθος της εικόνας. Ο παραμετρικός χώρος (Hough space) συνεχίζει να υπάρχει μόνο που είναι κβαντισμένος, δηλαδή είναι χωρισμένος σε κελιά, όπου το κάθε κελί αντιπροσωπεύει το εύρος τιμών  $(\theta, \rho)$ .

Ο πίνακας αυτός ονομάζεται αλλιώς και κουτί συσσώρευσης (accumulator bin).

Τουτέστιν, σκοπός της μεθόδου είναι ο εντοπισμός γραμμών, μέσω των ακμών που υπάρχουν στην εικόνα. Γνωρίζουμε πως από ένα σημείο μπορούν να διέλθουν άπειρο πλήθος ευθειών. Για κάθε μια ευθεία που τέμνει αυτό το σημείο δίνεται και μια ψήφος στο αντίστοιχο  $(\theta, \rho)$  κελί του αθροιστικού πίνακα (accumulator bin).

Επίσης γνωρίζουμε πως σε έναν Ευκλείδειο χώρο, από δύο σημεία μπορεί να διέλθει μονάχα μια ευθεία.



Η ευθεία που θα περάσει από δύο σημεία θα πάρει δύο ψήφους μιας και είναι μοναδική.

Αναλογικά σκεπτόμενοι, οι ευθείες της εικόνας που τέμνουν αρκετά σημεία θα έχουν και μεγάλο αριθμό ψήφων στο αθροιστικό πίνακα.

Συνοψίζοντας λοιπόν, για κάθε σημείο (edge) της εικόνας αντικαθιστούμε τις συντεταγμένες ( $x_0, y_0$ ) στην εξίσωση (2) και στη συνέχεια αναζητούμε την τιμή του  $\rho$  σε όλο το φάσμα των τιμών του  $\theta$ .

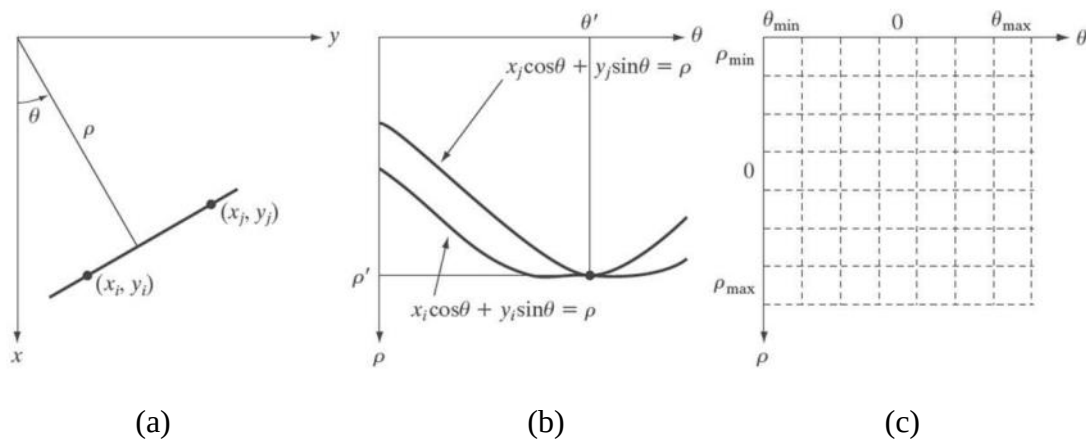
Σε κάθε ένα κελί του ζεύγους ( $\theta, \rho$ ) βάζουμε και από μια ψήφο. Παρατηρούμε λοιπόν ότι, το ζεύγος ( $\theta, \rho$ ) στον παραμετρικό χώρο δημιουργεί μια αρμονική ημιτονοειδής καμπύλη για κάθε ένα σημείο της εικόνας.

Τα σημεία στα οποία τέμνονται οι ημιτονοειδής καμπύλες στον παραμετρικό χώρο είναι και εκείνα με τους περισσότερους ψήφους στον αθροιστικό πίνακα.

Σκοπός μας είναι να βρούμε τα κελία του πίνακα (ζεύγη ( $\theta, \rho$ )) τα οποία έχουν τις μέγιστες τιμές.

Αυτά τα κελιά θα περιγράφουν μοναδικά τις ευθείες που υπάρχουν στην εικόνα.





**Εικόνα 3.7** Σχέση σημείων με ημιτονοειδές γραμμές στον Hough space

- (a) Χώρος της εικόνας με δύο σημεία  $(x_i, y_i)$  και  $(x_j, y_j)$  καθώς και η ευθεία που τα τέμνει
- (b) Ημιτονοειδείς αναπαράσταση των αντίστοιχων σημείων
- (c) Απεικόνιση accumulator bin

## Κεφάλαιο 4 : Υλοποίηση προτεινόμενων μεθοδολογιών

Σε αυτό το κεφάλαιο θα μιλήσουμε για τον τρόπο που υλοποιήσαμε τις δύο προτεινόμενες μεθόδους, θα γίνει αναλυτική περιγραφή των προσεγγίσεων που ακολουθήσαμε ενώ τέλος παρουσιάζουμε τα αποτελέσματα και τα κρίσιμα σημεία τα οποία λάβαμε υπόψιν κατά τη διάρκεια της υλοποίησης.

Η πρώτη μέθοδος βασίζεται στη στατιστική ανάλυση της έντασης καταμερισμένων περιοχών εικόνας. Η μέθοδος αυτή βασίζεται σε μια ολική (global) κατάτμηση της εικόνας σε μέρη, αναζητώντας σε αυτά μέσω μιας αθροιστικής συνάρτησης της έντασης τα σημεία του ιστογράμματος τα οποία παρουσιάζουν χαμηλές τιμές. Η περιοχές αυτές ενδεχομένως να παρουσιάζουν ασυνέχειες στο δίκτυο.

Η δεύτερη μέθοδος βασίζεται σε μια τοπική προσέγγιση (local processing) . Κύριος σκοπός αυτής της μεθόδου είναι ο εντοπισμός των γραμμών του πλέγματος με βάση τις ακμές (edges) που περιέχονται πάνω σε μια εικόνα μέσω του μετασχηματισμού Hough. Στη συνέχεια υλοποιείται μια σύγκριση της απόστασης μεταξύ πραγματικού δικτυού και εντοπισμένου πλέγματος ευθειών, μέσα απ' το οποίο ανιχνεύουμε την ύπαρξη ή μη ασυνέχειας στην εικόνα

Και οι δύο μεθοδολογίες υλοποιούνται πάνω σε τρεις εικόνες. Προσπαθήσαμε να πάρουμε σαν δείγματα εικόνες οι οποίες ποικίλουν τόσο στο μέγεθος του δικτυού όσο και σε ενδεχόμενες τρύπες που υπάρχουν πάνω σε αυτό. Οι εικόνες που χρησιμοποιήσαμε είναι αλιευμένες απ' το διαδίκτυο κάτω από άγνωστες σε εμάς συνθήκες. Στις υποθαλάσσιες εικόνες λόγω αντικειμενικών συνθηκών που μπορούν να υπάρχουν, όπως είναι η έλλειψη φωτεινότητας, ζωικός ή φυτικός θόρυβος πάνω στα δίκτυα, παραμορφώσεις του δικτυού λόγω κυμάτωσης του νερού κτλπ. μας καθιστούν δύσκολο το έργο της επεξεργασίας τους. Για να μπορέσουμε να φέρουμε τις εικόνες μας σε ένα ικανό επίπεδο ώστε να αντλήσουμε πληροφορίες χρειαζόμαστε κάποιες μεθόδους κατάτμησης και φιλτραρίσματος τους.

Γι' αυτό το λόγο στα πρώτα στάδια των αλγορίθμων υλοποιείται μια επεξεργασία ικανή να ξεπεράσει αυτές τις αδυναμίες και να φέρει την εικόνα μας σε ένα καλό επίπεδο ανάλυσης. Η επεξεργασία που υλοποιούμε και στις δύο μεθόδους είναι σε πρωτόλειο επίπεδο, το οποίο όμως παρόλα αυτά μας επιτρέπει να τις επεξεργαστούμε σε ικανοποιητικό βαθμό. Σκοπός μας είναι η ανάλυση το πλέγματος και η ανίχνευση ασυνεχειών πάνω στο μοτίβο του.

## 4.1 Εισαγωγή - Global process

Στο πρώτο κεφάλαιο θα αναλύσουμε τη μέθοδο της ολικής επεξεργασίας (global process) που υλοποιήσαμε. Η υλοποίηση έγινε αρχικά μέσω του καταμερισμού της εικόνας σε τμήματα, μέσα στα οποία αναζητήσαμε το ποσοστό της έντασης του κάθε παραθύρου. Ο καταμερισμός της έντασης μέσα από στατιστικά μοντέλα, όπως η αθροιστική κατανομή, έδωσε ικανοποιητικά αλλά όχι πλήρως αποδοτικά αποτελέσματα, κάτι που θα αναλυθεί παρακάτω.

*Τα στάδια που ακολουθούμε σε αυτήν τη μέθοδο είναι πέντε:*

- *Ανάγνωση και επεξεργασία εικόνας*
- *Καταμερισμός εικόνας (Pool Block)*
- *Καταμερισμός κάθε Block σε Sub Blocks και υπολογισμός έντασης*
- *Υπολογισμός αθροιστικής συνάρτησης (cumulative sum) και εντοπισμός threshold*
- *Απεικόνιση πιθανών τρυπών*

Για την υλοποίηση της παραπάνω μεθόδου δημιουργήσαμε επτά συναρτήσεις στο πρόγραμμα της MATLAB. Οι ενέργειες παρουσιάζονται ευσύννοπτα παρακάτω, ενώ ακολουθούν τα διαγράμματα ροής της μεθόδου στην Εικόνα 4.1 και του αλγορίθμου που υλοποιήθηκε στην MATLAB στην Εικόνα 4.2:

Αρχικά υλοποιήσαμε την κύρια συνάρτηση (main) η οποία διαβάζει την εικόνα και υλοποιεί την πρώτη επεξεργασία της (pre-process διαδικασία). Στη συνέχεια καλούμε μια συνάρτηση η οποία κατακερματίζει την εικόνα μας σε ισόποσα μέρη (μπλόκς). Τα χωριζόμενα αυτά μέρη (pool blocks) αποτελούν τους χώρους αναζήτησης ενδεχόμενων ασυνεχειών που μπορούν να υπάρχουν πάνω στην εικόνα. Έπειτα καλούμε μια δεύτερη συνάρτηση (sub block split) η οποία αρχικά δέχεται σειριακά κάθε ένα από τα προαναφερθέντα μπλόκς, όπου στη συνέχεια τα κατακερματίζει σε ακόμη μικρότερα (sub blocks). Μέσα σε αυτά τα υπο-μπλόκς (ή υπο-παράθυρα) αναζητούμε την ποσότητα της έντασης η οποία περιέχεται σε κάθε ένα παράθυρο. Τέλος αποθηκεύουμε την τιμή του κάθε παραθύρου σε έναν πίνακα. Ως ποσότητα της έντασης ενός υπο-μπλοκ ορίζουμε την ποσότητα των λευκών pixels που περιέχονται σε αυτό.

Αφού έχουμε εντοπίσει και αποθηκεύσει την τιμή της έντασης κάθε υπο-παραθύρου, δημιουργούμε ένα ιστόγραμμα κατανομής της ποσότητας των εντάσεων. Έπειτα υπολογίζουμε την αθροιστική κατανομή του ιστογράμματος αυτού.

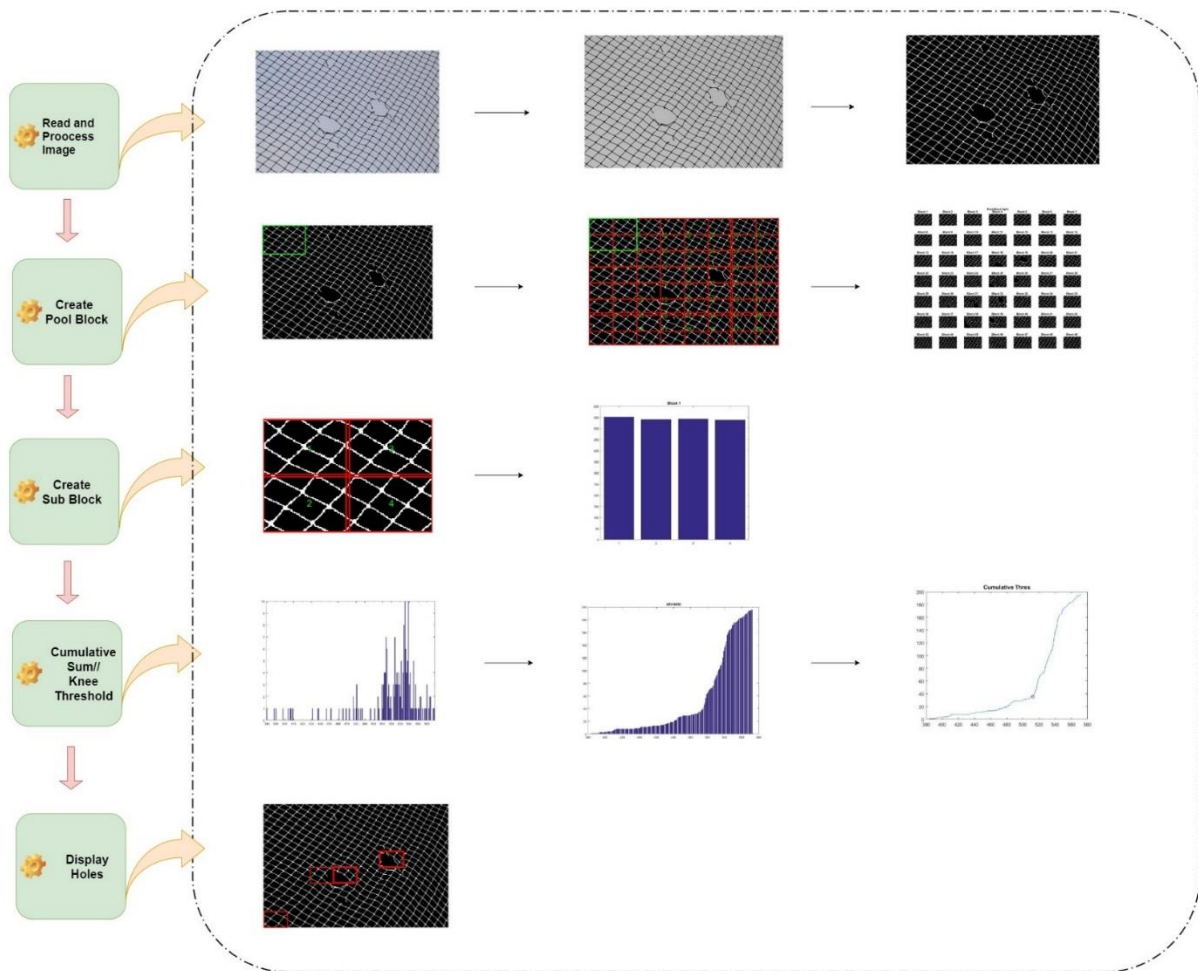
Τέλος αναζητούμε πάνω στην καμπύλη το «γόνατο» αθροιστικής συνάρτησης. Το σημείο αυτό μας δείχνει το όριο κάτω από το οποίο θεωρούμε κάτι ως ασυνέχεια πάνω στην εικόνα μας.

Με το πέρας της υλοποίησης των παραπάνω ενεργειών επιστρέφουμε στην αρχική μας συνάρτηση (main), απ' όπου υλοποιούμε την απεικόνιση των εντοπισμένων ασυνεχειών πάνω στην εικόνα.

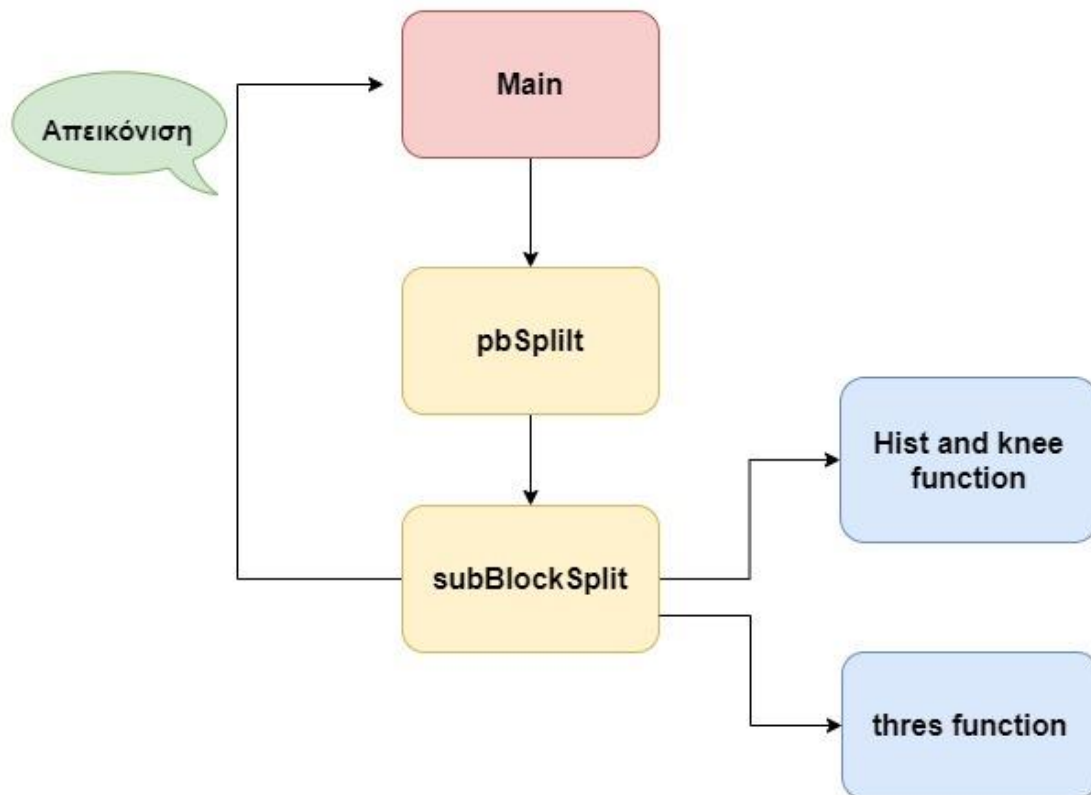
---

Block diagram της διαδικασίας

---



Εικόνα 4.1 Block Diagram πρώτης μεθόδου



Εικόνα 4.2 Block Diagram συναρτήσεων πρώτης μεθόδου

#### 4.1.1 Ανάγνωση και επεξεργασία εικόνας

Στο πρώτο στάδιο του αλγορίθμου μας πραγματοποιείται η επεξεργασία της εικόνας. Εδώ ο αντικειμενικός σκοπός είναι η αφαίρεση του θορύβου που ενδεχομένως να υπάρχει και η ανάδειξη των πραγματικών ακμών της εικόνας. Μια καλή επεξεργασία θα μας δώσει τις ευκρινείς ακμές του διχτυού αφαιρώντας τις αχνές ή «σπασμένες» ακμές.

Για την υλοποίηση των παραπάνω, αναζητήσαμε την το καθολικό όριο  $T$  (global threshold) της gray εικόνας μας, χρησιμοποιώντας τη μέθοδο Otsu. Η συνάρτηση αυτή εντοπίζει το ελάχιστο κατώφλι (threshold) έντασης της εικόνας, με σκοπό να διαχωρίσουμε το foreground από το background.

Στη συνέχεια χρησιμοποιώντας το παραπάνω όριο (threshold) μετατρέπουμε την εικόνα σε binary. Όπου ορίζουμε ως 1 (άσπρο) το φόντο και ως 0 (μαύρο) τα δίκτυα.

Παρά το γεγονός ότι τα αντικείμενα μελέτης μας (δίκτυα) έχουν διακριθεί από το φόντο, ενδέχεται να υπάρχουν οριακά σημεία του πλέγματος τα οποία έχουν παρόμοιες εντάσεις με αυτό.

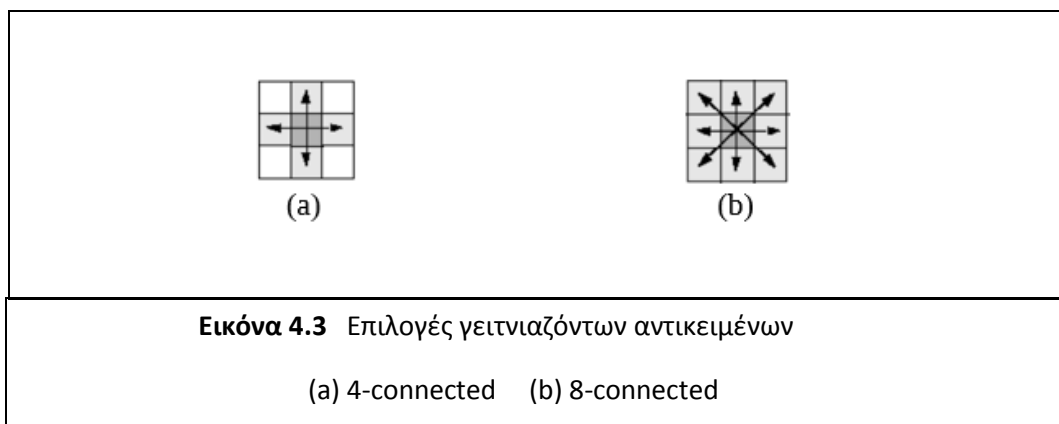
Για να πετύχουμε την συνεκτικότητα μεταξύ των pixel εφαρμόζουμε μια επιπλέον συνάρτηση, η οποία ανήκει στις συναρτήσεις μορφολογικής ανακατασκευής (morphological reconstruction) και μας βοηθάει στη δημιουργία συνεκτικών συνόρων (boundaries).

Η λογική της εν λόγω συνάρτησης είναι η εξής: Γνωρίζοντας ότι τα pixel (0,0) ανήκουν το φόντο (background) της εικόνας, εφαρμόζουμε μια flood-fill operation. Τα pixel τα οποία δεν θα επηρεαστούν από την ενέργεια αυτή θα ανήκουν αναγκαστικά εντός των συνόρων του αντικειμένου.

Η λογική πίσω από τον αλγόριθμο της «πλημμύρας» (flood-fill operation) έγκειται στην συνένωση όμοιου χρώματος/έντασης pixel. Δεδομένου ενός αρχικού πίνακα (στην περίπτωσή μας μιας εικόνας) αναζητούμε βάση ενός αρχικού κόμβου γειτνιάζοντα σημεία με τον αρχικό κόμβο. Όσα σημεία γειτνέουν με τον αρχικό κόμβο αλλάζουν το χρώμα τους ως προς το αρχικό.

Σαν όρισμα η συνάρτηση παίρνει την binary εικόνα καθώς και το όριο συνεκτικότητας που θέλουμε να εφαρμόζουμε. Εμείς επιλέγουμε σαν όριο συνεκτικότητας το 8-connected [\[39\]](#).

Στην εικόνα 4.3 βλέπουμε τις δύο επιλογές συνεκτικότητας που μας δίνει η συνάρτηση:



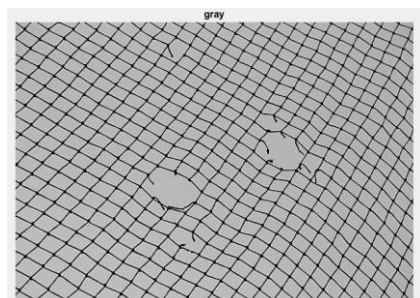
Αυτό το κάνουμε για να οριοθετήσουμε με τον καλύτερο δυνατό τρόπο το τι είναι αντικείμενο και τι όχι πάνω στην εικόνα. Καθώς και να αναδείξουμε τα αντικείμενα μελέτης μας (δίκτυα) αποκομμένα από το φόντο.

Για παράδειγμα, στην παρακάτω binary εικόνα, υπάρχει ένα αντικείμενο στο προσκήνιο (foreground object) που αντιστοιχεί στα pixels με τιμή ίση με 1 και δύο αντικείμενα στο φόντο (background objects) με τιμή 0, αυτό πέριξ των άσων και ο κύκλος εντός αυτών.

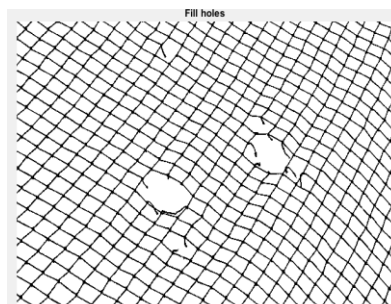
0	0	0	0	0	0	0	0
0	1	1	1	1	1	0	0
0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	0
0	1	0	0	0	1	0	0
0	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

**Εικόνα 4.4** Παράδειγμα binary εικόνας

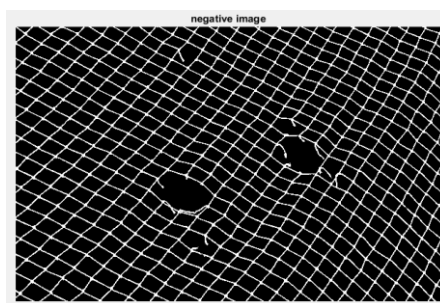
Τέλος αντιστρέφουμε την εικόνα έτσι ώστε να έχουμε ως 1 (άσπρο) τα δίχτυα και ως 0 (μαύρο) το φόντο.



(a)



(b)



(c)

**Εικόνα 4.5** Gray / Binary / Invert image

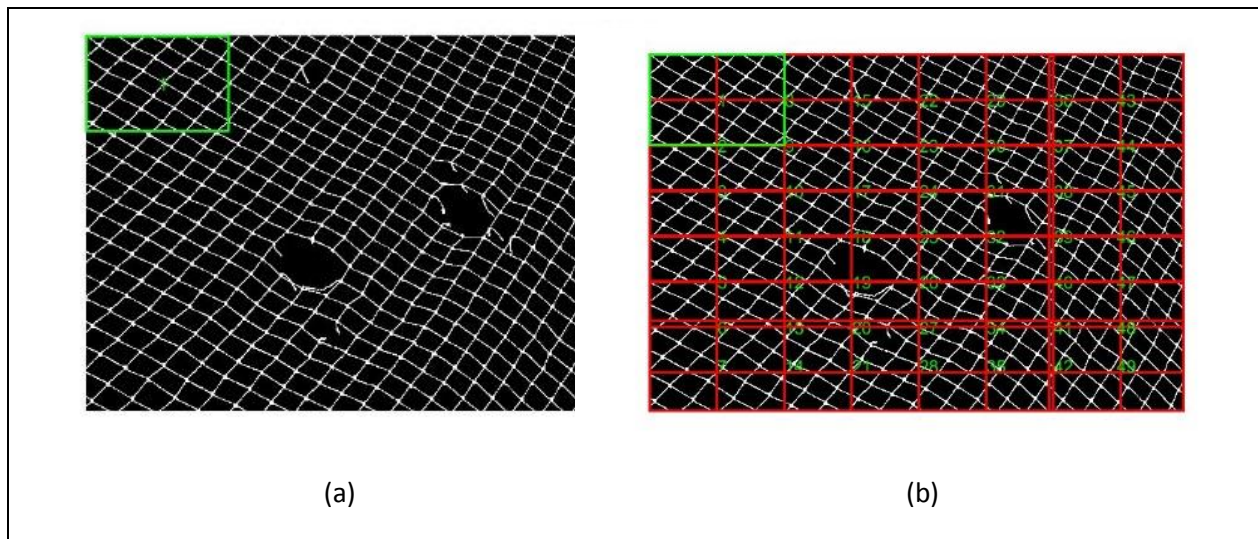
(a) Gray Image    (b) Binary Image    (c) Inverse Binary Image

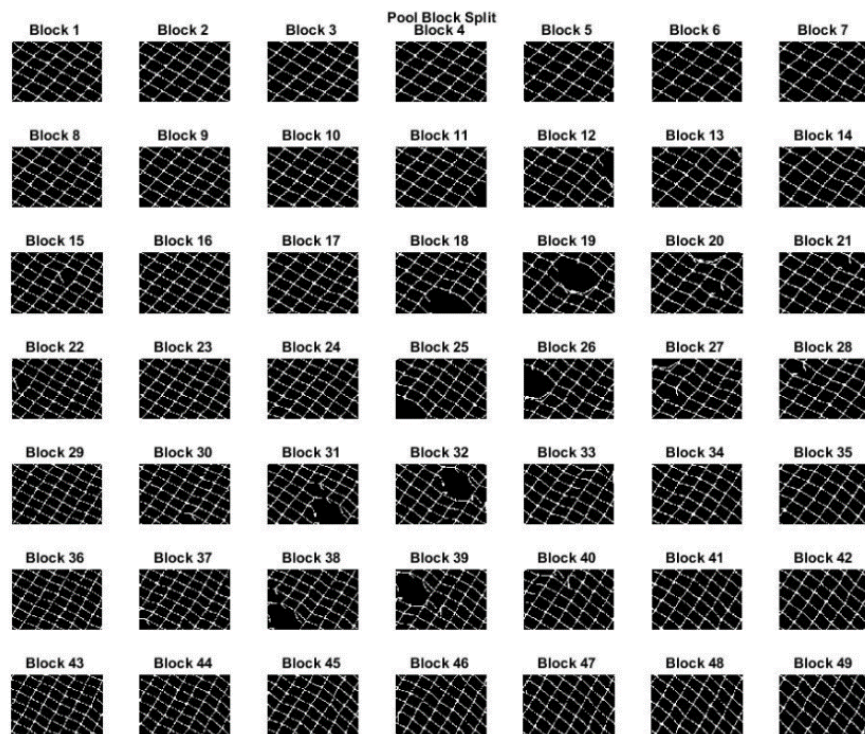


#### 4.1.2 Καταμερισμός εικόνας (Pool Block)

Αφού έχουμε αντιστρέψει την εικόνα μας, δημιουργούμε ένα τετράγωνο παράθυρο που έχει μέγεθος 25% της συνολικής εικόνας. Στην συνέχεια κατακερματίζουμε την εικόνα μας σε παράθυρα μεγέθους ίσου με το αρχικό, βάζοντας επικάλυψη μεταξύ τους 50%. Ο σκοπός της όλης διαδικασίας είναι να μπορέσουμε να «σπάσουμε» την εικόνα μας σε blocks τα οποία θα περιέχουν μέσω των αλληλοκαλύψεων μια ικανοποιητική πρόσβαση σε πολλαπλές όψεις της εικόνας με παρόμοιο περιεχόμενο δικτυού. Προσπαθούμε δηλαδή να εγκολπώσουμε ενδεχόμενες τρύπες του δικτυού σε διαφορετικά blocks ώστε να εντοπίσουμε μέσω της στατιστικής ανάλυσης των εντάσεων το περιεχόμενο των ασυνεχειών στην εικόνα.

Στην πρώτη εικόνα με πράσινο χρώμα αναπαριστούμε το αρχικό block, στην δεύτερη εικόνα απεικονίζουμε την εικόνα καταμερισμένα στα blocks (με επικάλυψη 50%), ενώ στην τρίτη εικόνα απεικονίζουμε το μέρος που καταλαμβάνει κάθε block της εικόνας μας (pool block)





(c)

#### Εικόνα 4.6 Pool block

- (a) Με πράσινο απεικονίζεται το block .  
Το μέγεθος κάθε παραθύρου είναι 25% του μεγέθους της εικόνας.
- (b) Καταμερισμός της εικόνας σε block με 50% overlap.
- (c) Απεικόνιση όλων των blocks ( pool block). Στην συγκεκριμένη εικόνα δημιουργήθηκαν 49 blocks. Στην συγκεκριμένη εικόνα δημιουργήθηκαν 49 blocks.

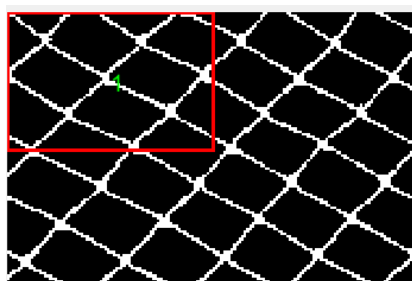
### 4.1.3 Καταμερισμός σε Sub Blocks και υπολογισμός έντασης

Αφού έχουμε χωρίσει την εικόνα μας σε τμήματα, στο επόμενο βήμα διαιρώ κάθε παράθυρο της εικόνας σε ακόμη μικρότερα παράθυρα. Η διαίρεση γίνεται παίρνοντας ποσοστιαία το  $\frac{1}{4}$  του block χωρίς επικάλυψη. Επειδή σε πολλές εικόνες υπήρχε πρόβλημα με τις οριακές τιμές, για να χωρέσει το block την διαίρεση χωρίς το παράθυρο να ξεφεύγει απ' τα όρια της εικόνας, μετακινήσαμε τα sub blocks ελάχιστα το ένα μέσα στο άλλο.

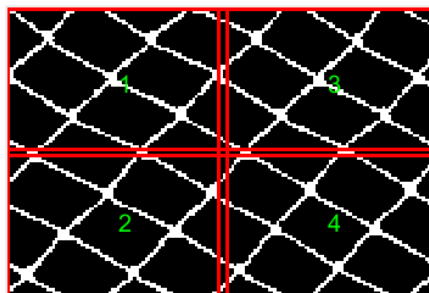
Ο καταμερισμός της εικόνας σε όσο το δυνατόν μικρότερα μέρη γίνεται για να μπορέσουμε να επεξεργαστούμε τις εικόνες μας σε μικρό χρονικό διάστημα μέσω της μεθόδου του παραλληλισμού.

Η επιλογή του μεγέθους του παραθύρου, λόγω των υποθαλάσσιων συνθηκών, γίνεται βάσει την λήψη που έχει η εικόνα, την ανάλυση της, την εστιακή απόσταση κτλ. Παρόλα αυτά θεωρούμε πως το παράθυρο στο τελευταίο στάδιο της διαίρεσης θα πρέπει να περιλαμβάνεται ένα αρκετά μεγάλο μέρος της εικόνας, ικανό για την συναγωγή λογικών αποτελεσμάτων μέσω της έντασης του. Το μέγεθος αυτό καθορίστηκε μέσω παρατηρήσεων σε μέγεθος διπλάσιο απ' όσο είναι το τετράγωνο που δημιουργεί το πλέγμα του δικτύου (net cell).

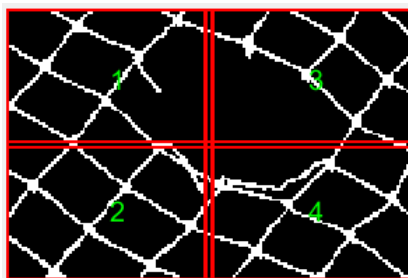
Αυτό το βλέπουμε και στις παρακάτω εικόνες, στην πρώτη παρουσιάζουμε κάποιο block χωρίς τρύπα, όπου με κόκκινο χρώμα απεικονίζεται το παράθυρο του sub block, στην δεύτερη εικόνα παρουσιάζουμε ολοκληρωμένα την διαίρεση ενός block σε sub blocks, ενώ τέλος στην τρίτη εικόνα δείχνουμε πως απεικονίζεται μια τρύπα μετά την διαίρεση του block.



(a)



(b)



(c)

#### Εικόνα 4.7 Sub block

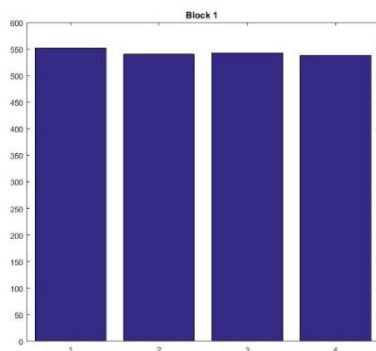
- (a) Απεικόνιση sub block. Το μέγεθος του παραθύρου είναι  $\frac{1}{4}$  του block.
- (b) Sub Block χωρίς τρύπα στα δίκτυα.
- (c) Sub Block με τρύπα στα δίκτυα

Στην συνέχεια μπορούμε να εντοπίσουμε την ένταση σε κάθε sub block. Για τον εντοπισμό ελέγχουμε σε κάθε ένα παράθυρο ποιο είναι το ποσοστό λευκών ριχέλ που περιέχει. Στο επίπεδο του αλγορίθμου αναζητούμε σε κάθε έναν πίνακα την ποσότητα των άσων που περιέχει.

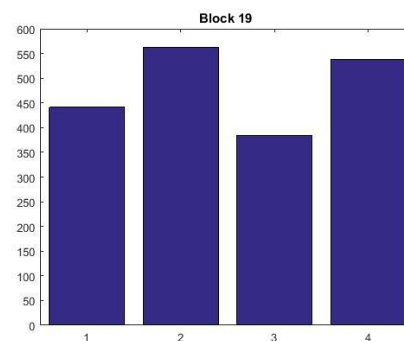
Στα παράθυρα όπου το πλέγμα δεν είναι κομμένο το ποσοστό άσων αναμένουμε να είναι μεγάλο, ενώ στα παράθυρα όπου υπάρχουν ασυνέχειες το ποσοστό της έντασης θα είναι μειωμένο.

Στις παρακάτω εικόνες παρουσιάζουμε σε μορφή ιστογράμματος, το ποσοστό της έντασης των δύο παραπάνω εικόνων. Παρατηρούμε ότι τα αποτελέσματα είναι ανάλογα των προβλέψεων μας.

Ο xχ' άξονας δηλώνει το εκάστοτε παράθυρο ενώ ο γγ' δηλώνει το ποσοστό άσων που περιέχει το παράθυρο.



(a)

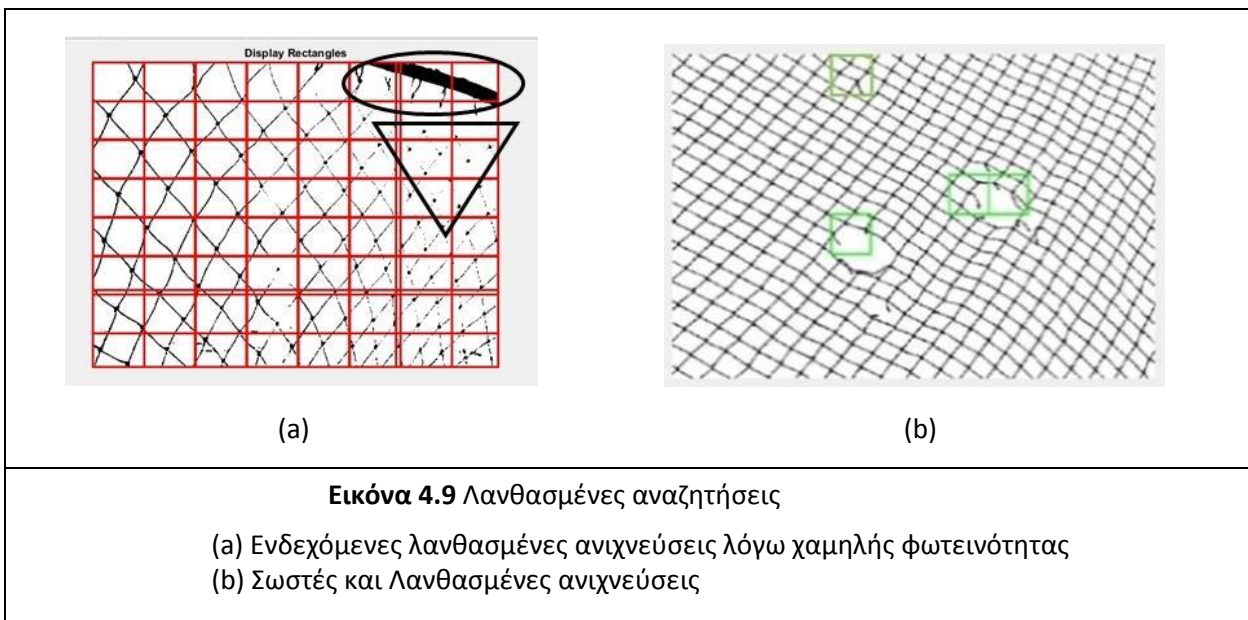


(b)

#### Εικόνα 4.8 Ιστόγραμμα με/χωρίς τρύπα

- (a) Ιστόγραμμα block χωρίς τρύπα (b) Ιστόγραμμα block με τρύπα

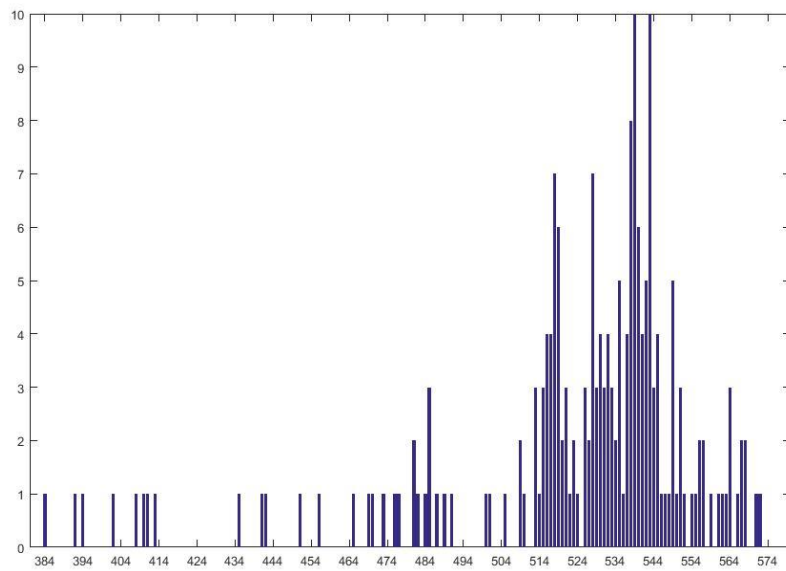
Παρόλα αυτά, η δυνατότητα καταγραφής της έντασης διαφόρων σημείων της εικόνας δεν μας δίνει ξεκάθαρο αποτέλεσμα για τη δομή του δικτυού. Όπως έχουμε ξανατονίσει, οι υποθαλάσσιες συνθήκες είναι περίπλοκες, έτσι τα αποτελέσματα που θα λάβουμε ενδεχομένως να είναι παραπλανητικά πολλές φορές. Υπάρχει η πιθανότητα ανίχνευσης από τον αλγόριθμο μιας ασυνέχειας πάνω στην εικόνα λόγω χαμηλής φωτεινότητας, είτε λόγω της κλίσης της κάμερας. Στην αριστερή εικόνα εντοπίζονται ασυνέχειες στην περιοχή με το τρίγωνο λόγω χαμηλής φωτεινότητας ενώ με το οβάλ σχήμα εντοπίζονται αντικείμενα τα οποία δεν ανήκουν πάνω στο πλέγμα. Το αποτέλεσμα αυτό προκύπτει λόγω χαμηλής κλίσης της κάμερας λήψης. Στη δεξιά εικόνα εντοπίζεται λανθασμένα τρύπα στο πάνω μέρος της εικόνας λόγω της κυμάτωσης που έχει το δίχτυ, κάτι που δημιουργεί μεγαλύτερη επιφάνεια σε εκείνο το σημείο και άρα μεγαλύτερη συγκέντρωση άσπρων pixel (άσων) πάνω στο παράθυρο του sub block.



Για το λόγο αυτό και με στόχο να ελαττώσουμε την πιθανότητα λανθασμένης ανίχνευσης, χρησιμοποιήσαμε την μέθοδο της αθροιστικής κατανομής των εντάσεων ώστε να εντοπίσουμε ένα καλύτερο threshold για την εικόνα μας.

#### 4.1.4 Υπολογισμός αθροιστικής συνάρτησης (cumulative sum)

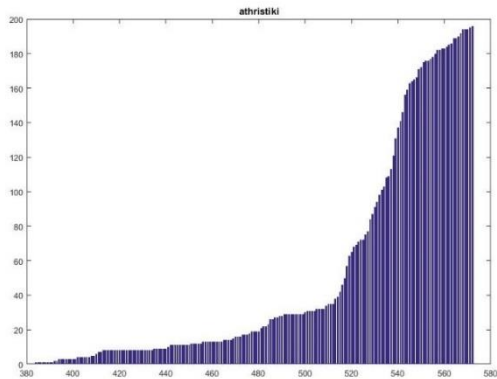
Με την βοήθεια ενός ιστογράμματος απεικονίζουμε όλες τις τιμές των εντάσεων που περιέχονται στην εικόνα. Ο άξονας  $x'$  αντιστοιχεί στις τιμές των εντάσεων κάθε ενός sub block, ενώ ο  $y'$  απεικονίζει την πιθανότητα εμφάνισης κάθε τιμής της έντασης. Πόσες φορές δηλαδή εμφανίστηκε η τιμή της έντασης πάνω στην εικόνα. Παρατηρούμε ότι η πλειοψηφία των τιμών έχει μεγάλη τιμή (βρίσκεται δεξιά της κατανομής) και εμφανίζεται και πολλές φορές. Ενώ στα αριστερά έχουμε τις τιμές με μικρή ένταση και με μικρή πυκνότητα εμφάνισης, οι τιμές αυτές αντιστοιχούν στις τρύπες της εικόνας.



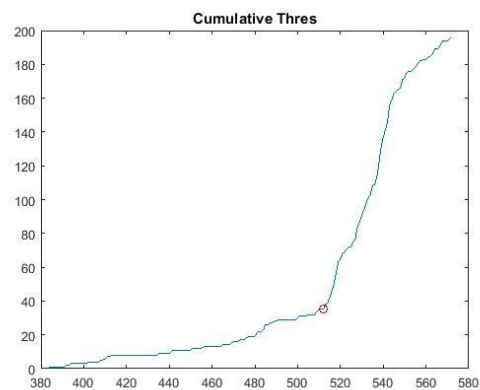
**Εικόνα 4.10** Καταμερισμός της έντασης της εικόνας

Στη συνέχεια εφαρμόζουμε αθροιστική συνάρτηση κατανομής (cdf) πάνω σε αυτές τις τιμές χρησιμοποιώντας στην συνάρτηση `cumsum` της βιβλιοθήκης της MATLAB. Τέλος αναζητούμε πάνω στην αθροιστική συνάρτηση (cdf) το σημείο με την μεγαλύτερη αλλαγή κλίση της καμπύλης. Το σημείο αυτό ονομάζεται γόνατο (knee curve) της συνάρτησης.

Τα αποτελέσματα είναι τα εξής:



(a)



(b)

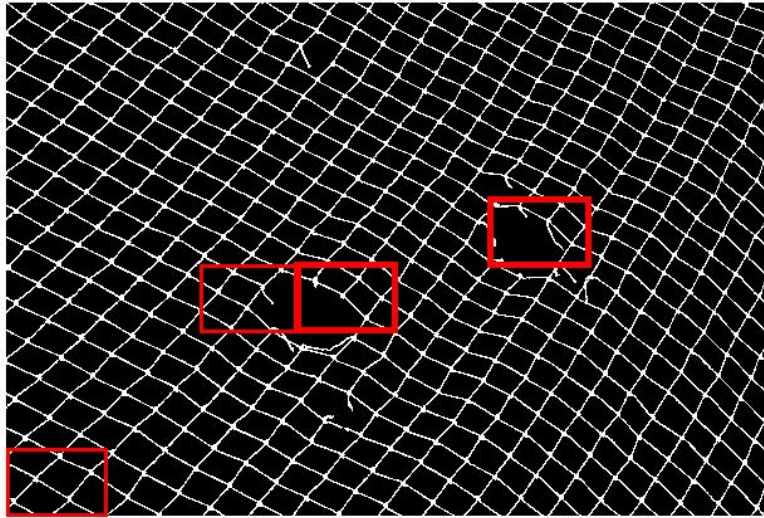
**Εικόνα 4.11** Αθροιστική/ Threshold

(a) Αθροιστική συνάρτηση κατανομής      (b) Το κόκκινο σημείο αποτελεί το knee curve (threshold)



#### 4.1.5 Εντοπισμός πιθανών τρυπών

Τα αποτελέσματα της εικόνας παρουσιάζονται παρακάτω. Παρατηρούμε πως εντοπίζονται οι τρύπες σε σχετικά ικανοποιητικό βαθμό, αλλά επιπλέον εντοπίζεται εσφαλμένα και κάποιο σημείο στο άκρο της εικόνας. Αυτό έχει να κάνει με τις συνθήκες λήψης της εικόνας, όπως προ είπαμε. Λόγω κυμάτωσης του νερού στο συγκεκριμένο σημείο έχει απλώσει το δίχτυ, αυξάνοντας το εμβαδόν του πλέγματος και άρα προσθέτοντας επιπλέον άσπρα pixel στο σημείο.



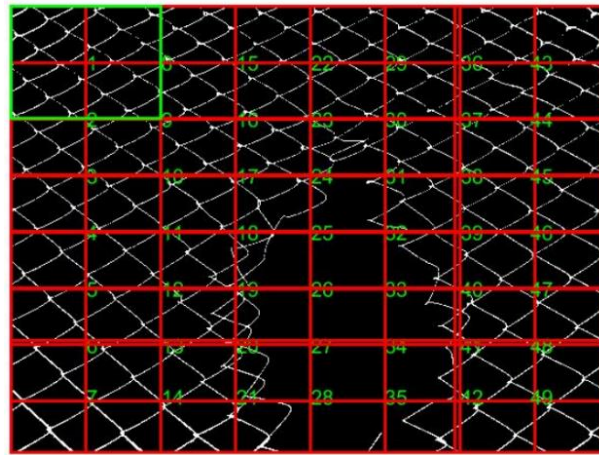
**Εικόνα 4.12** Τελικό αποτέλεσμα εντοπισμού

## 4.1.6 Αποτελέσματα

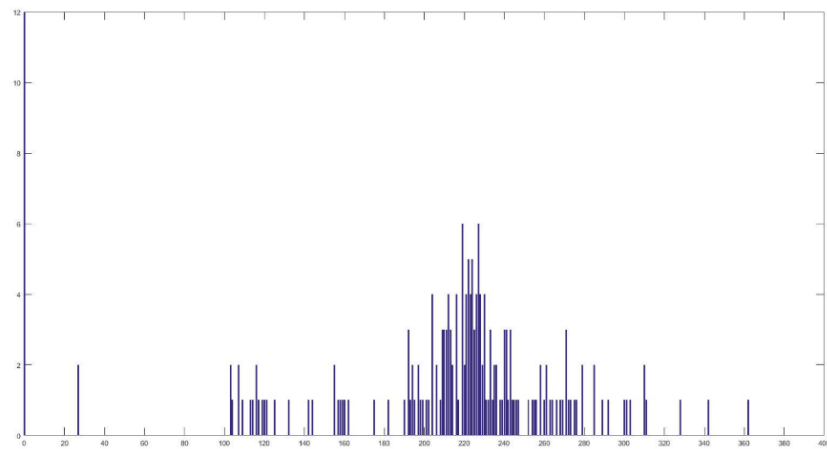
### *Τα αποτελέσματα για την δεύτερη εικόνα*

Στο δεύτερο παράδειγμα τα δίκτυα έχουν μεγαλύτερο μέγεθος, ενώ υπάρχει μια μεγάλη τρύπα στο μέσον της εικόνας. Τα αποτελέσματα είναι τα εξής:

Στην Εικόνα 4.13 (a) απεικονίζεται η κατάτμηση της εικόνας σε παράθυρα, όπου με πράσινο δείχνουμε το αρχικό τετράγωνο καταμερισμού. Στη Εικόνα 4.13 (b) φαίνεται το ιστόγραμμα των εντάσεων, στην Εικόνα 4.14 (a) και (b) φαίνονται η αθροιστική συνάρτηση μαζί με το threshold της καμπύλης. Τέλος στην 4.14 (c) εμφανίζουμε τα sub blocks στα οποία υπάρχει τρύπα.



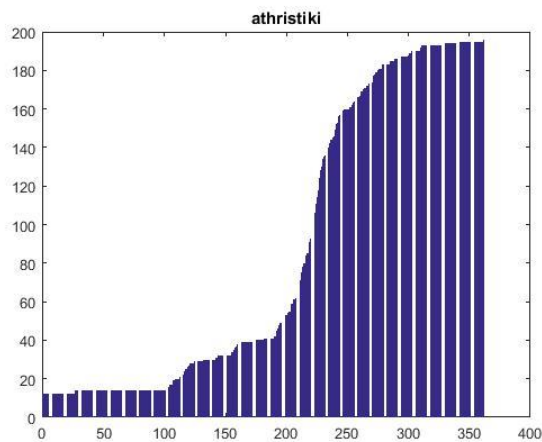
(a)



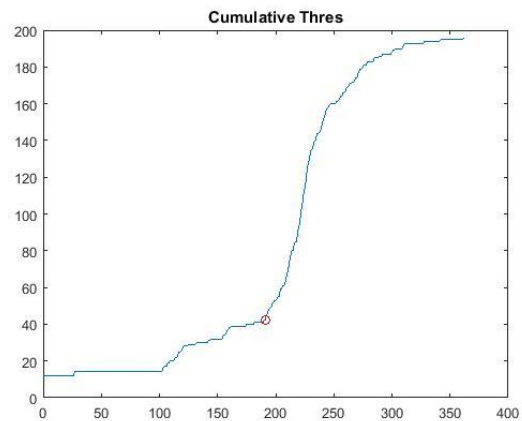
(b)



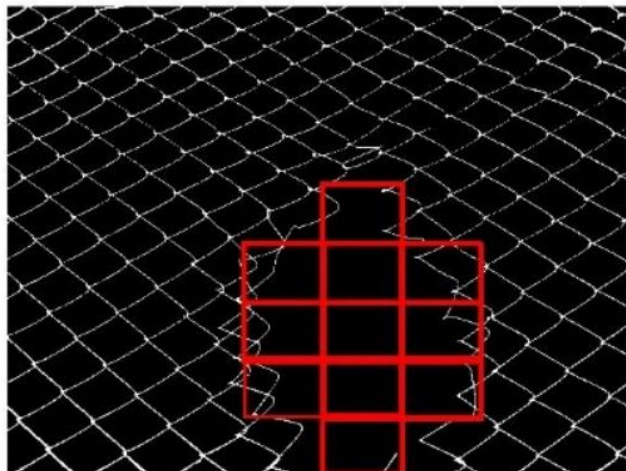
**Εικόνα 4.13** Παράδειγμα 2: Καταμερισμός block δεύτερης εικόνας/ Ιστόγραμμα έντασης  
 (a) Καταμερισμός σε block στην δεύτερη εικόνα      (b) Ιστόγραμμα εντάσεως εικόνας



(a)



(b)



(c)

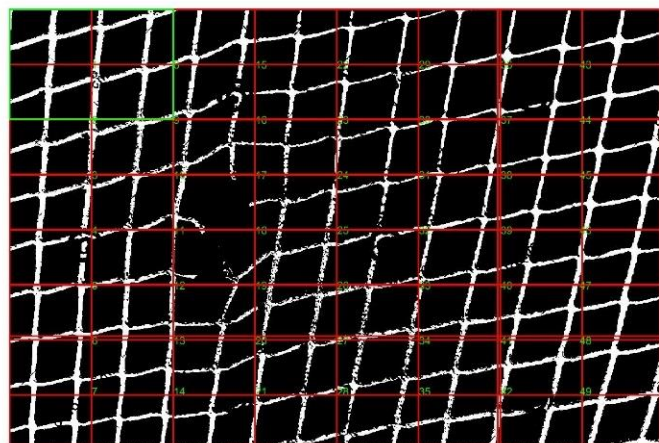
**Εικόνα 4.14** Παράδειγμα 2: Αθροιστική συνάρτηση / Threshold/ Τελικό αποτέ-  
 (a) Cumulative sum της δεύτερης εικόνας      (b) Knee curve, το threshold με κόκκινο  
 (c) Τελικό αποτέλεσμα

---

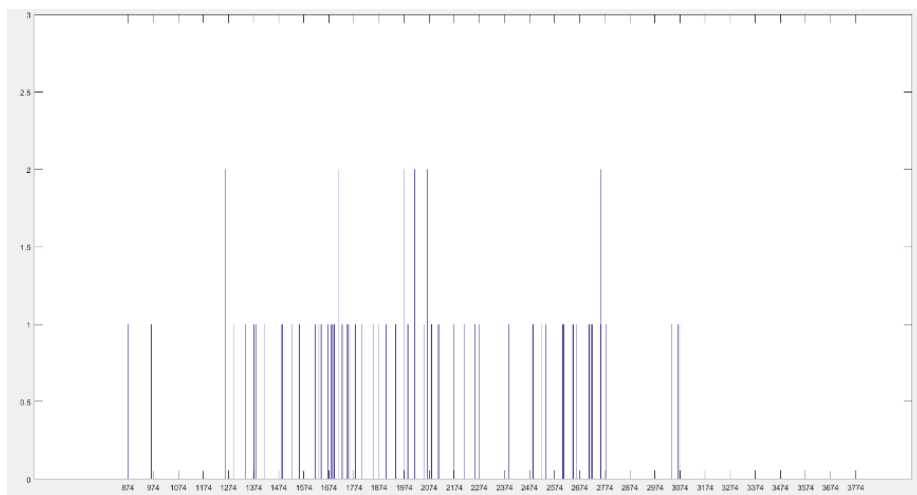
*Τα αποτελέσματα για την τρίτη εικόνα:*

---

Στην Εικόνα 4.15 (a) φαίνεται όπως και πριν ο καταμερισμός της εικόνας, στην Εικόνα 4.15 (b) το ιστόγραμμα των εντάσεων, στην Εικόνα 4.16 (a) και (b) η αθροιστική συνάρτηση μαζί με το threshold, ενώ στην Εικόνα 4.16 (c) εμφανίζονται τα sub blocks τα οποία περιλαμβάνουν ασυνέχειες.



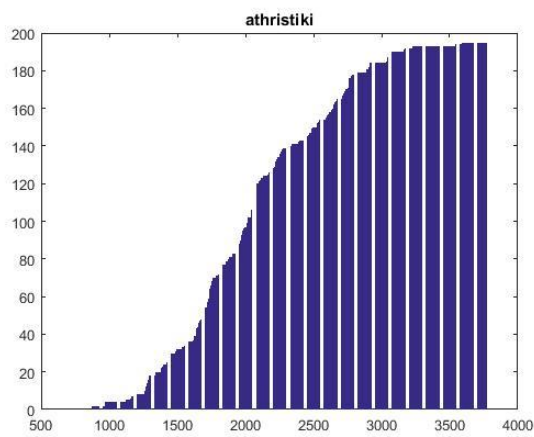
(a)



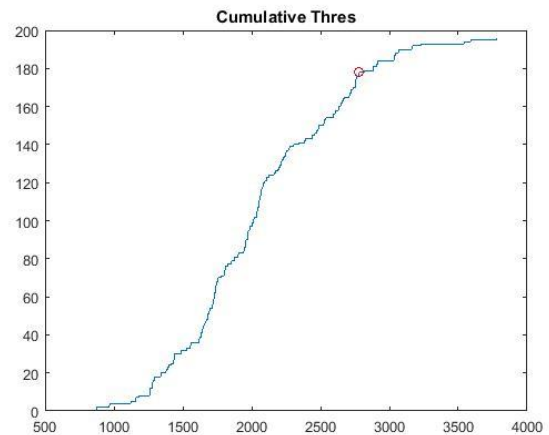
(b)

**Εικόνα 4.15** Παράδειγμα 3: Καταμερισμός εικόνας/ Ιστόγραμμα έντασης

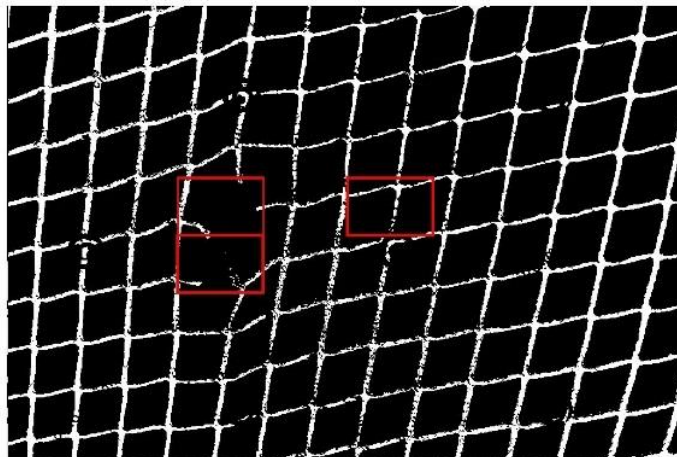
(a) Καταμερισμός block για την τρίτη εικόνα    (b) Ιστόγραμμα εντάσεως εικόνας



(a)



(b)



(c)

**Εικόνα 4.16** Παράδειγμα 3: Αθροιστική συνάρτηση/ Threshold/ Τελικό αποτέλεσμα

(a) Cumulative sum της τρίτης εικόνα    (b) Knee curve, το threshold με κόκκινο

(c) Τελικό αποτέλεσμα

Παρατηρούμε ότι στο αποτέλεσμα του δεύτερου παραδείγματος, η τρύπα εντοπίζεται ολόκληρη. Στο αποτέλεσμα του τρίτου παραδείγματος επίσης εντοπίζεται η ασυνέχεια στο δίκτυ μαζί όμως με ένα

ακόμη εσφαλμένο σημείο. Αυτό συμβαίνει κυρίως λόγω του τεντώματος που υφίσταται (στο σημείο αυτό) το πλέγμα καθώς και λόγω της κακής φωτεινότητας.

Με το πέρας της κατωφλίωσης κάποια σημεία του δικτυού ενδεχομένως να εξασθενήσουν όταν δεν υπάρχουν μεγάλες διακυμάνσεις στη φωτεινότητα της εικόνας. Αποτέλεσμα αυτού του «φαγώματος» είναι η ανισομερής εμφάνιση περισσότερων μαύρων απ' ότι λευκών pixels, και άρα εν τέλει η επιλογή του σημείου ως τρύπα από τον αλγόριθμο.

Για να ξεπεράσουμε προβλήματα όπως τα παραπάνω θα χρειαζόμασταν επιπλέον φίλτρα εντοπισμού και αποκατάστασης συνόρων ή/και αναζήτηση εναλλακτικού προσαρμοσμένου κατωφλίου πάνω στην εικόνα.

Η παρούσα μέθοδος μας δίνει τη δυνατότητα μιας πολλαπλής οπτικής απεικόνισης σημείων του πλέγματος. Παρόλα αυτά όπως αναφέραμε παραπάνω, είναι επιρρεπής σε λάθη λόγω της ανισομερούς κατανομής των ιστογραμμάτων έντασης.

Στα προβλήματα που δεν μπορεί να δώσει επαρκή λύση η πρώτη μέθοδος θεωρούμε πως απαντάει η δεύτερη. Σε αυτήν εστιάζουν στην τοπική επεξεργασία της εικόνας, βασιζόμενοι στη γραμμική απόσταση σημείων, δηλαδή σημείων που ενδεχομένως να παρουσιάζουν (εσφαλμένα) στοιχεία ασυνέχειας μέσω της τοπικής ανάλυσης.

## 4.2 Εισαγωγή - Local process

Στο δεύτερο μέρος του κεφαλαίου θα αναλύσουμε τη μέθοδο της τοπικής επεξεργασίας (local processing) για τον εντοπισμό των τρυπών πάνω σε δίκτυα. Κύριος σκοπός αυτής της μεθόδου είναι η εξαγωγή των γραμμών του πλέγματος με βάση τις ακμές (edges) που περιέχονται πάνω σε μια εικόνα μέσω του μετασχηματισμού Hough.

Ο μετασχηματισμός Hough είναι μια τεχνική διαδικασία όπου εξάγουμε μοτίβα μέσα από σύνολα σημείων. Σκοπός αυτής της μεθόδου είναι η δημιουργία ενός παράλληλου πλέγματος ευθειών πάνω από τον σύνολο των δικτύων της εικόνας. Αφού το επιτύχουμε αυτό, υπολογίζουμε την Ευκλείδεια απόσταση μεταξύ πραγματικού και εικονικού δικτύου. Αν τα αποτελέσματα των αποστάσεων είναι μεγάλα τότε δηλώνουμε την ύπαρξη τρυπών στα δίκτυα και χρωματίζουμε τα σημεία όπου το δίκτυο είναι σκισμένο.

- Η επεξεργασία της εικόνας και η ανίχνευσης ακμών.
- Η εφαρμογή Hough transform και ο εντοπισμός ευθειών.
- Η ανίχνευση μέγιστων ακμών και ο εντοπισμός των κύριων κατευθύνσεων του δικτύου.
- Ο εντοπισμός μέσης απόστασης δικτύου.
- Ο εντοπισμός πιθανών τρυπών.

Για την υλοποίηση της παραπάνω μεθόδου δημιουργήσαμε επτά συναρτήσεις στο πρόγραμμα της MATLAB. Οι ενέργειες παρουσιάζονται ευσύννοπτα παρακάτω, ενώ ακολουθούν τα διαγράμματα ροής της μεθόδου στην Εικόνα 4.17 και του αλγορίθμου στην Εικόνα 4.18:

Την κύρια συνάρτηση μας (main) η οποία διαβάζει την εικόνα μας, την επεξεργάζεται και εντοπίζει τις γραμμές πάνω σε αυτήν μέσω της συνάρτησης του μετασχηματισμού Hough.

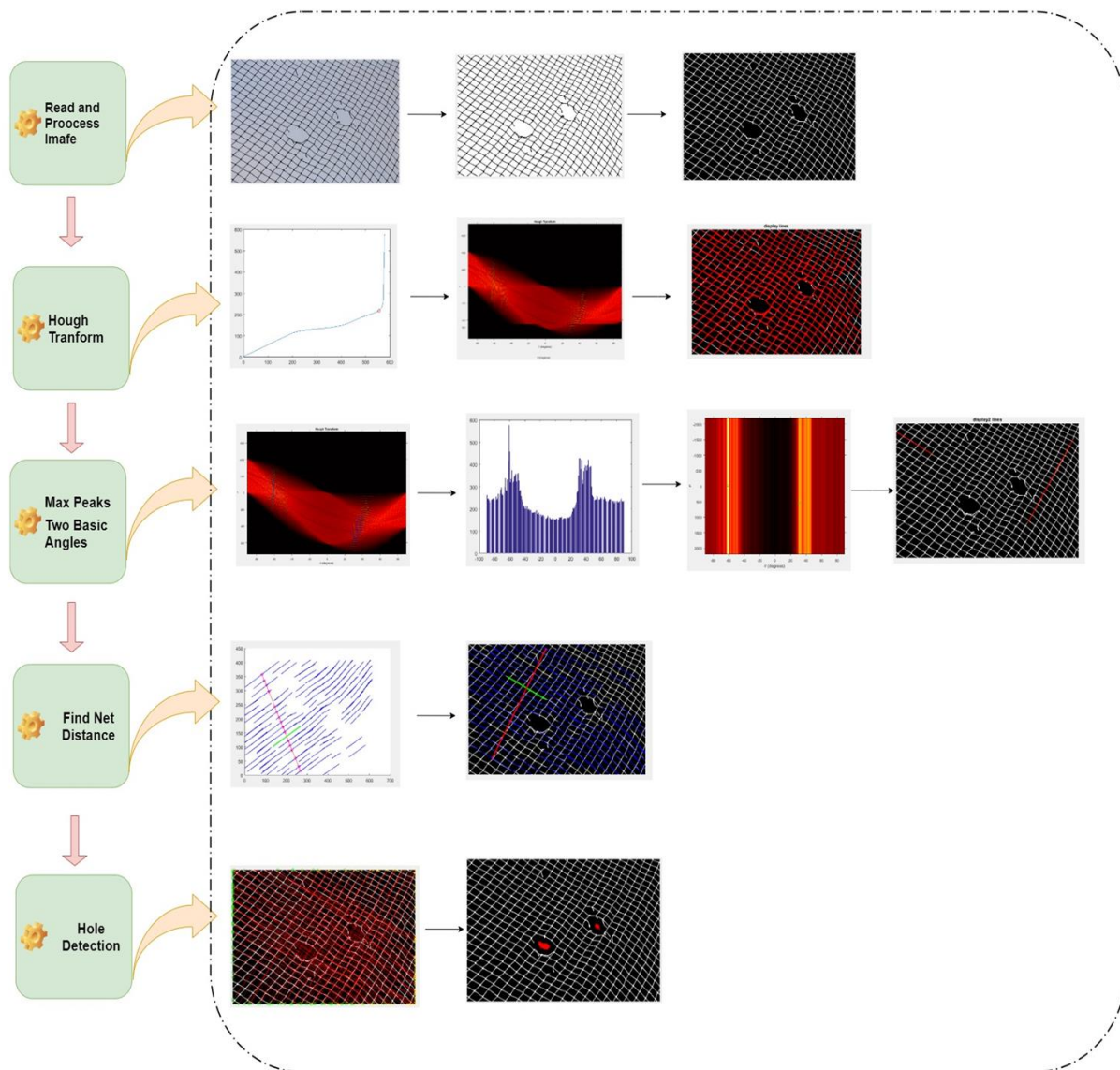
Στην συνέχεια με βάση τις τιμές των ψήφων κάθε ενός ευθυγράμμου τμήματος στον κβαντισμένο αθροιστικό πίνακα του παραμετρικού χώρου (Hough space) δημιουργούμε μια αθροιστική συνάρτηση, στην οποία αναζητούμε το όριο κλίσης πάνω σε αυτή. Αφού το εντοπίσουμε εφαρμόζουμε το κατώφλι πάνω στις ακμές (peaks) του παραμετρικού χώρου (Hough space) ώστε να επιλέξουμε τα αντίστοιχα ευθύγραμμα τμήματα. Οι εν λόγω ευθείες αντιστοιχούν στα σημεία (peaks) με τις υψηλότερες τιμές ψήφου.

Έπειτα αναζητούμε τις κατευθύνσεις των παραπάνω ευθυγράμμων τμημάτων. Για να το πετύχουμε αυτό ψάχνουμε το ζεύγος  $(\theta, \rho)$  στον παραμετρικό χώρο τω δύο peaks με τον υψηλότερο αριθμό ψήφων (max peaks). Μόλις εντοπιστούν αυτομάτως θα γνωρίζουμε και τις δύο κύριες κατευθύνσεις που θα έχει το πλέγμα μας.

Μετά τον εντοπισμό των κύριων κατευθύνσεων μεταβαίνουμε στην συνάρτηση η οποία θα μας δώσει την απόσταση που έχει το πλέγμα του δικτύου της εικόνας.

Τέλος έχοντας πλέον γνώση και της κατεύθυνσης του πλέγματος αλλά και της απόστασης του δικτύου, περνάμε στη συνάρτηση υλοποίησης του εικονικού πλέγματος. Απ' όπου θα αναζητήσουμε την ευκλείδεια απόσταση μεταξύ πραγματικού και εικονικού πλέγματος. Τα σημεία τα οποία παρουσιάζουν μεγάλες τιμές απόκλισης από την μέση απόσταση, χρωματίζονται ως ασυνέχειες

*Block diagram της διαδικασίας:*

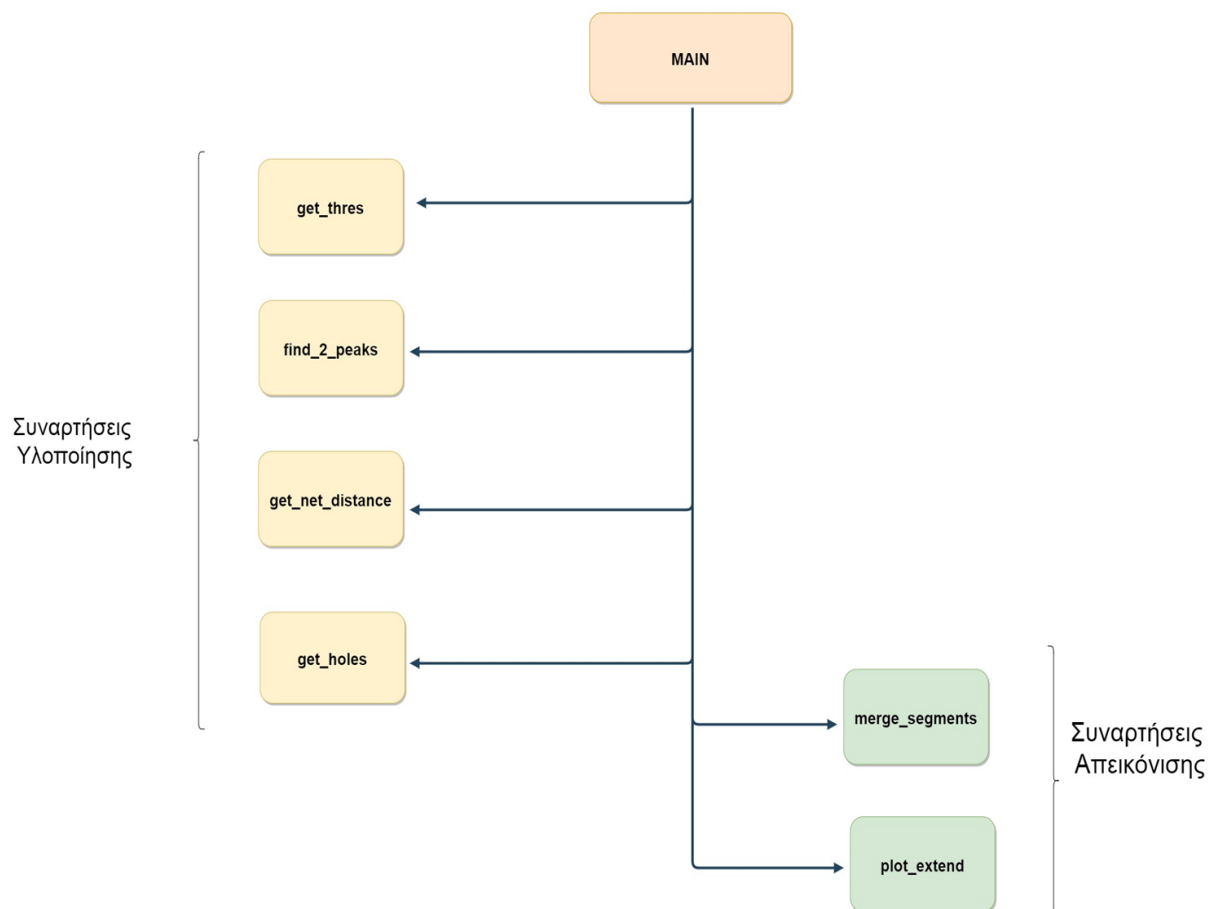


**Εικόνα 4.17** Block Diagram δεύτερης μεθόδου

---

*Block diagram των συναρτήσεων:*

---

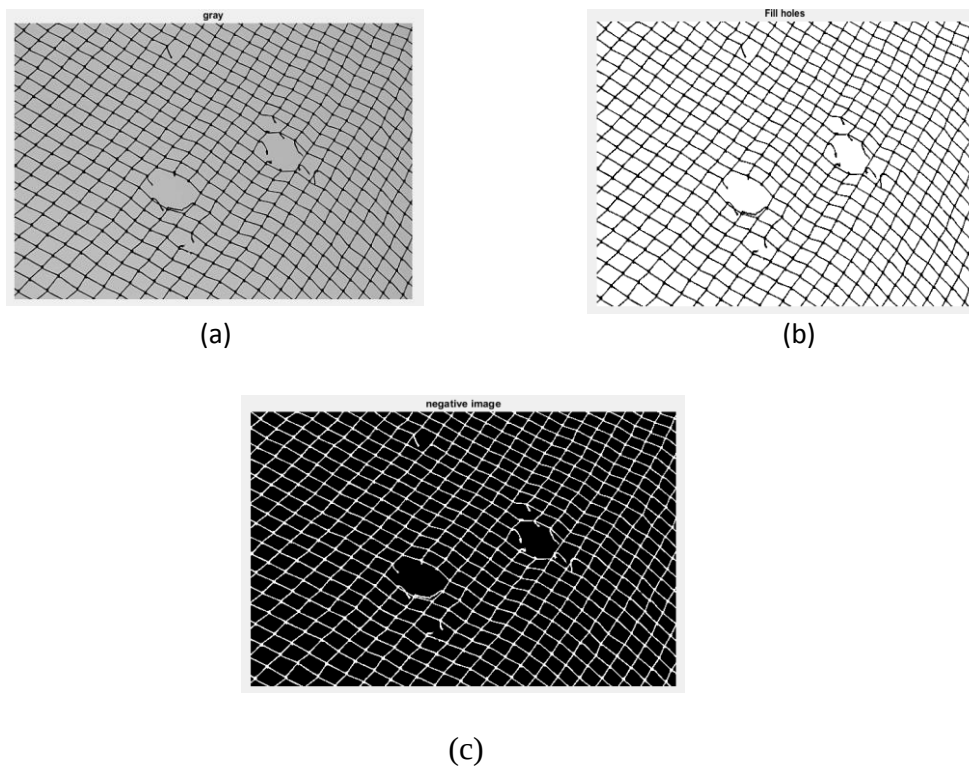


**Εικόνα 4.18** Block Diagram συναρτήσεων δεύτερης μεθόδου

#### 4.2.1 Διαδικασία ανίχνευσης ακμών

Η διαδικασία ανίχνευσης ακμών γίνεται με τον ίδιο τρόπο όπως και στην προηγούμενη μέθοδο (μέθοδος Otsu και δημιουργία συνεκτικότητας στα συνορεύοντα των pixel).

Τα αποτελέσματα φαίνονται στην Εικόνα 4.5, η οποία ξανά παρατίθεται παρακάτω:



**Εικόνα 4.5** Gray / Binary / Invert image  
(a) Gray Image    (b) Binary Image    (c) Inverse Binary Image



### 4.2.2 Εφαρμογή Hough transform και εντοπισμός ευθειών

Αφού έχουμε επεξεργαστεί την εικόνα μας, το επόμενο βήμα είναι να εντοπίσουμε τις ευθείες από δίκτυα που υπάρχουν πάνω σε αυτήν.

Χρησιμοποιώντας την τεχνική του μετασχηματισμού Hough, όπου προσπαθούμε να πετύχουμε δύο πράγματα:

- a) Τον εντοπισμό όλων των γραμμών που μπορεί να υπάρχουν στα δίκτυα της εικόνα μας, καθώς και
- b) Τον εντοπισμό των κύριων ευθειών αυτής (τις οποίες ονομάζουμε strong lines).

Τα αποτελέσματα φαίνονται στην Εικόνα 4.19, όπου στην (a) παρουσιάζονται με πράσινο χρώμα τα peaks πάνω στον παραμετρικό χώρο (Hough space), ενώ στην (b) παρουσιάζονται οι ευθείες, οι οποίες αντιστοιχούν στα προαναφερθέντα peak.

Αναλυτικότερα:

#### 4.2.2.1 Εντοπισμός όλων των ευθειών:

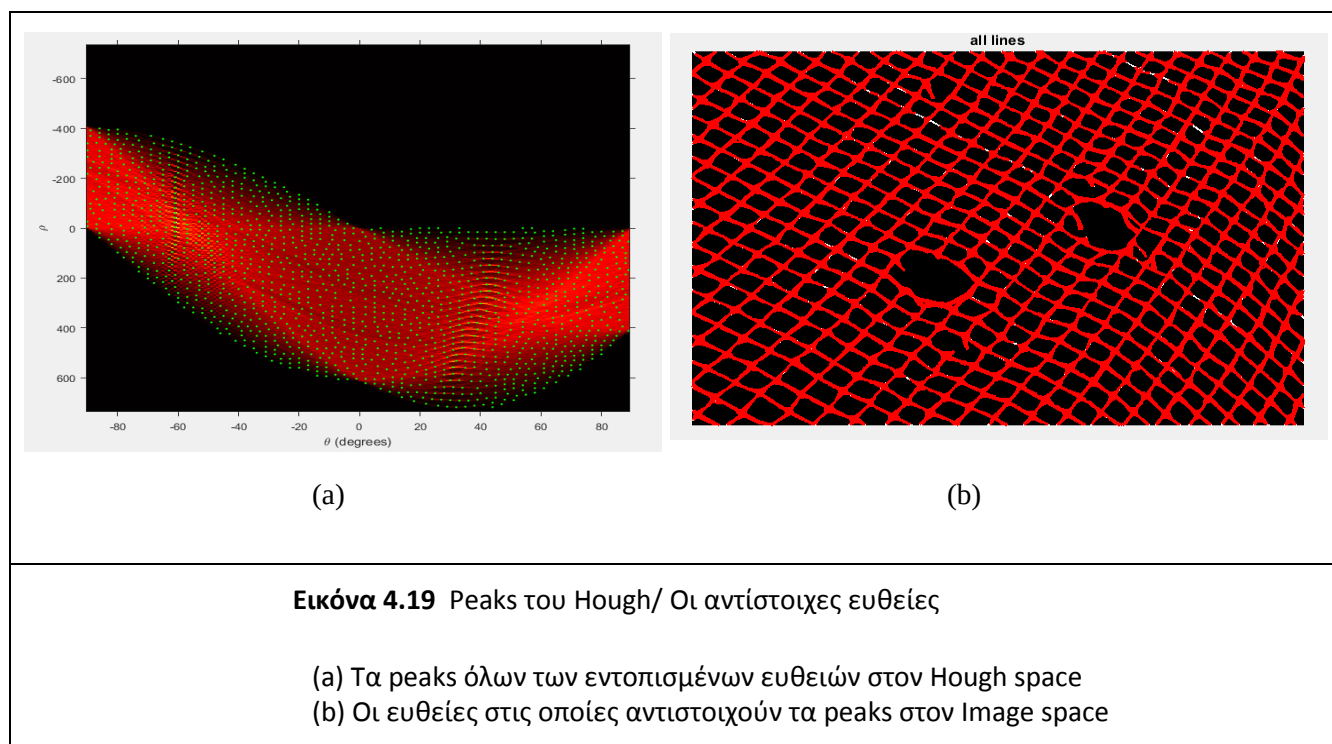
Για την υλοποίηση της συνάρτησης Hough θέσαμε τα όρια γωνίας theta ίσο με  $-90 \leq \theta \leq 90$ .

Το αποτέλεσμα που πήραμε απ' την συνάρτηση είναι ο κβαντισμένος πίνακας του Hough (ή αλλιώς accumulator bin) με μέγεθος  $D = \sqrt{rows^2 + cols^2}$  όπου rows και cols, οι γραμμές και οι στήλες της εικόνας μας.

Ο άξονας xχ' του αθροιστικού πίνακα έχει μέγεθος 180 στοιχείων, ενώ ο γγ' έχει τιμή round(D). Όπου round(D) ορίζεται ο πλησιέστερος ακαριαίος αριθμός της τιμής D.

Κατά αναλογία του παραμετρικού χώρου, ο xχ' άξονας αντιστοιχεί στις γωνιές theta οι οποίες παίρνουν τιμές από  $[-90 : 90]$  ενώ ο γγ' άξονας αντιστοιχίζεται με τις τιμές του rho οι οποίες έχουν εύρος  $[-round(D) : +round(D)]$ .

Το τελικό αποτέλεσμα που παίρνουμε είναι ένας πίνακας ο οποίος περιέχει το σημείο εκκίνησης και τερματισμού της κάθε ευθείας, την κλίση της (γωνία  $\theta$ ) καθώς και την τιμή του  $\rho$  (απόσταση απ' την αρχή των αξόνων). Η απεικόνιση των ευθειών στον παραμετρικό χώρο φαίνεται στην εικόνα 4.19 (a), ενώ οι αντίστοιχες εντοπισμένες ευθείες στην 4.19 (b).



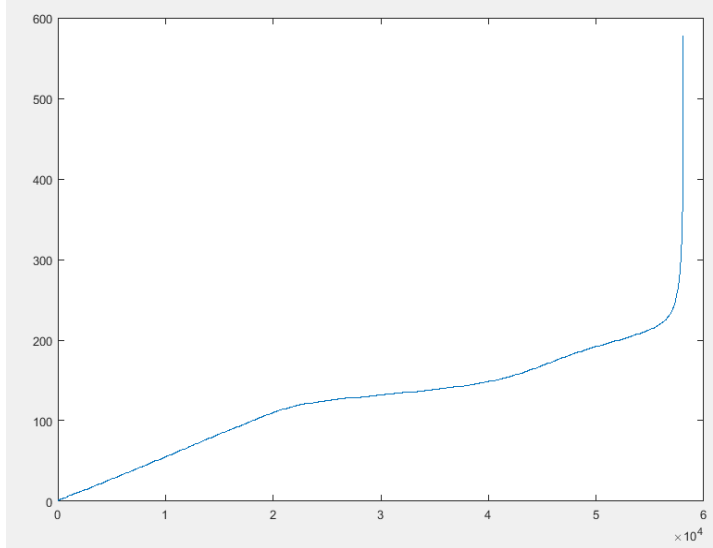
Κάθε μια ημιτονοειδής γραμμή στον παραμετρικό χώρο απεικονίζει όλες τις πιθανές ευθείες που περνάνε από ένα σημείο  $(x,y)$  της εικόνας. Ενώ αντίστοιχα κάθε σημείο (peak) της ημιτονοειδούς καμπύλης με ζεύγος  $(\theta, \rho)$  αντιστοιχεί σε ένα εύρος ευθειών στο επίπεδο της εικόνας. Τα σημεία διασταύρωσης των καμπυλών αντιστοιχούν στις ευθείες οι οποίες έχουν αθροιστικά αρκετά μεγάλο αριθμό ψήφων στον αθροιστικό πίνακα (accumulator bin).

Παρατηρούμε πως οι διασταυρωμένες ημιτονοειδείς καμπύλες επικεντρώνονται γύρω από δύο σημεία, κοντά στις  $-60^\circ$  και στις  $30^\circ$  μοίρες αντίστοιχα. Με αυτό το δεδομένο μπορούμε να βγάλουμε το συμπέρασμα πως σε αυτά τα peaks είναι υποψήφια να είναι φορείς “δυνατών” γραμμών - γραμμών από δίκτυα.

#### 4.2.2.2 Εντοπισμός κύριων ευθειών (strong lines):

Μετά τον εντοπισμό όλων των ευθειών της εικόνας μας, στόχος μας είναι να κρατήσουμε μόνο τις κύριες γραμμές (strong lines), εκείνες δηλαδή που έχουν αρκετούς ψήφους στον αθροιστικό πίνακα (accumulator bin) και να αφαιρέσουμε τις “αδύνατες” οι οποίες μπορεί να βρίσκονται σε κάποιο οριακό σημείο της εικόνας. Αυτό θα μας βοηθήσει στην μοντελοποίηση του εικονιζόμενου δικτύου, και άρα την εύρεση τρυπών πάνω σε αυτό.

Επειδή θέλουμε να αυτοματοποιήσουμε την εύρεση τρυπών, χρειαζόμαστε αρχικά ένα τρόπο με τον οποίο θα βρίσκουμε ποια είναι τα μέγιστα στον αθροιστικό πίνακα. Γι’ αυτό τον λόγο ταξινομούμε σε αύξουσα σειρά τις τιμές που υπήρχαν στον πίνακα για τιμές μεγαλύτερες του μηδενός. Το αποτέλεσμα φαίνεται στην εικόνα 4.20.



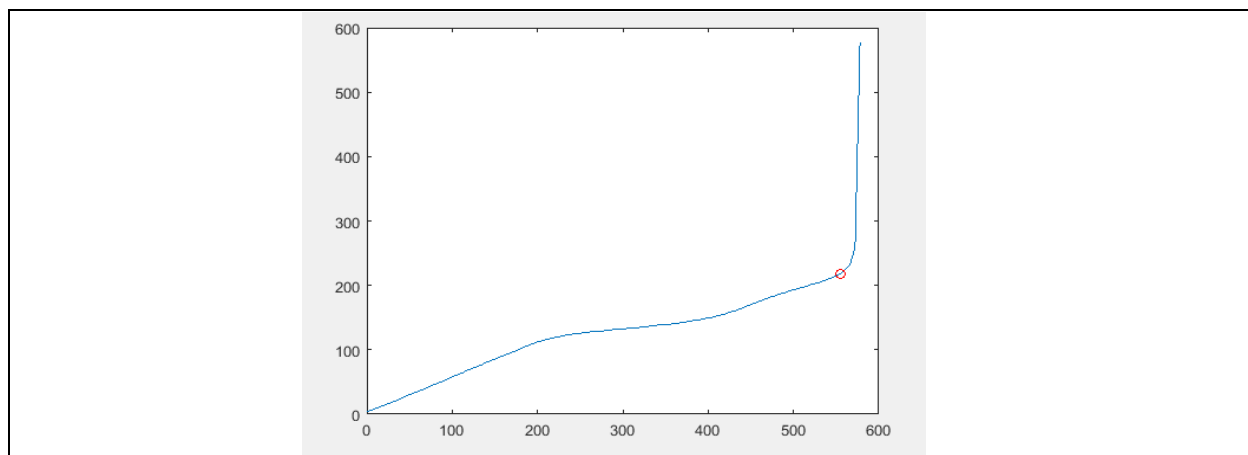
**Εικόνα 4.20** Κατανομή όλων των peaks με τιμή μεγαλύτερη του μηδενός

Έπειτα δημιουργούμε μια συνάρτηση η οποία δέχεται σαν όρισμα τον παραπάνω πίνακα, ενώ έχει σαν σκοπό τον εντοπισμό του ορίου κλίσης της παραπάνω αθροιστικής συνάρτησης. Σκοπός μας είναι να απομονώσουμε τα σημεία της καμπύλης που έχουν μεγάλες τιμές.

Για καμπύλη της παραπάνω μορφής, ψάχνουμε το σημείο όπου η άνοδος είναι πολύ «απότομη». Αυτό μπορεί να γίνει παίρνοντας τα σημεία, όπου η παράγωγος ξεπερνά κάποια τιμή. Επιλέγουμε ως όριο, το σημείο όπου η κλίση γίνεται  $45^\circ$ , άρα  $\frac{dx}{dy} = 1$ . Οπτικά, η καμπύλη φαίνεται όπως στο παραπάνω σχήμα όταν τη σχεδιάζουμε, αλλά για να είναι σωστή η εύρεση του σημείου με κλίση 1, θα πρέπει οι δύο άξονες να έχουν την ίδια κλίμακα (αν π.χ. έχουμε 1000 σημεία, ο οριζόντιος άξονας έχει τιμές 1 ως 1000).

Οπότε, επειδή οι τιμές είναι πολλές παίρνουμε 100 ενδεικτικά σημεία της καμπύλης (τιμές Y). Δημιουργούμε στη συνέχεια για τον X άξονα 100 τιμές με εύρος από 0 ως  $\max(Y)$ . Τέλος, υπολογίζουμε την παράγωγο (διαφορά του κάθε σημείου με το προηγούμενο διαιρεμένη με το  $dx$ ).

Το αποτέλεσμα που παίρνουμε είναι τα σημεία με κλίση πάνω από 1 και κρατάμε το πρώτο από αυτά ως το όριο.

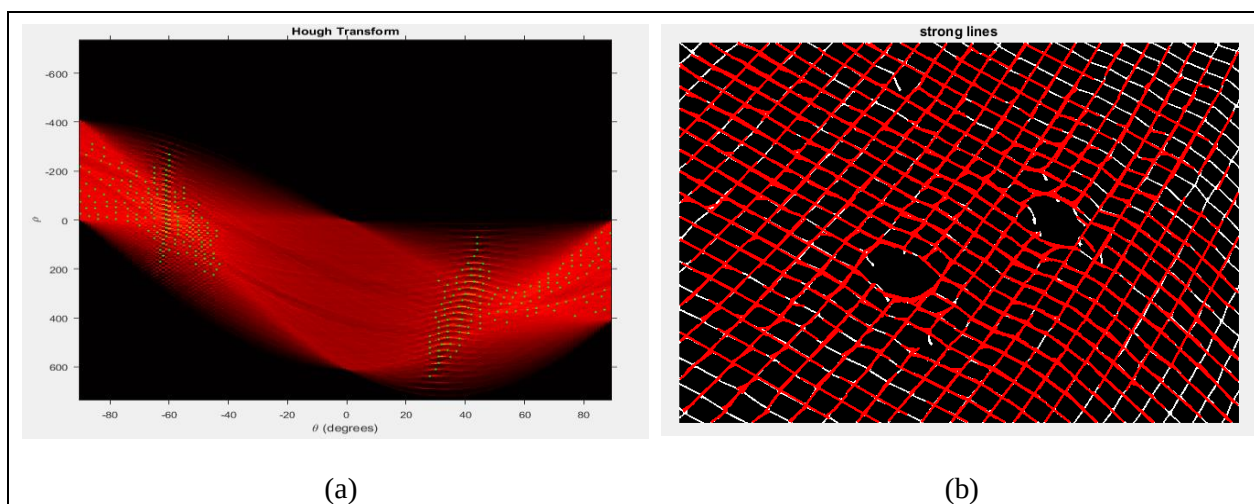


**Εικόνα 4.21** Threshold κατανομή

Με το κόκκινο κύκλο φαίνεται το σημείο κλίσης ο οποίο αντιστοιχεί στο κατώφλι (threshold)

Με αυτόν τον τρόπο βρίσκουμε αυτοματοποιημένα ποιο είναι το όριο (αριθμός ψήφων ενός κελιού στον αθροιστικό πίνακα) πάνω απ' το οποίο θα αναζητούμε τα strong lines στην εικόνα μας.

Τα σημεία (peaks) στον παραμετρικό χώρο, τα οποία ικανοποιούν το όριο που βρήκαμε παραπάνω απεικονίζονται στην Εικόνα 4.22 (a).



(a)

(b)

**Εικόνα 4.22** Peaks των strong lines/ Αντίστοιχες γραμμές

(a) Με πράσινο φαίνονται τα peaks των γραμμών (strong lines) που ικανοποιούν την τιμή του threshold στον Hough space

(b) Οι γραμμές στις οποίες αντιστοιχούν αυτά τα peaks στον χώρο της εικόνας (Image space)

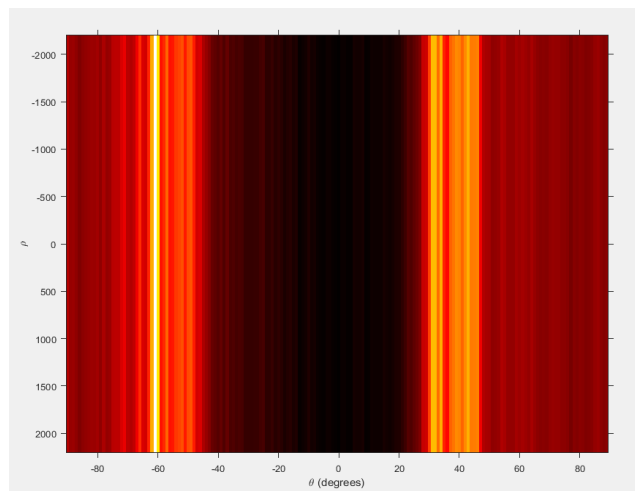
### 4.2.3 Max Peaks (Εντοπισμός Κύριων Κατευθύνσεων)

Αφού εντοπίσαμε λοιπόν όλα τις κύριες ευθείες της εικόνας μας (strong lines), στη συνέχεια αναζητούμε τις δύο κύριες κατευθύνσεις που μπορεί να έχουν αυτές. Σε μια εικόνα διχτυών, το πλέγμα συνηθίζεται να έχει δύο κύριες κατευθύνσεις, είτε κάθετα-οριζόντια, είτε λοξός αριστερά - δεξιά.

Την επιλογή της κατεύθυνσης την εντοπίζουμε σε περιοχές με έντονες χρωματικές διαφοροποιήσεις στον παραμετρικό χώρο, αυτό συμβαίνει λόγω του ότι εκεί διασταυρώνονται οι περισσότερες ημιτονοειδείς γραμμές και άρα αυτά τα σημεία θα περιέχουν μεγάλο αριθμό ψήφων στο αθροιστικό πίνακα. Κάτι που μας δηλώνει την ύπαρξη διχτυού στην εικόνα.

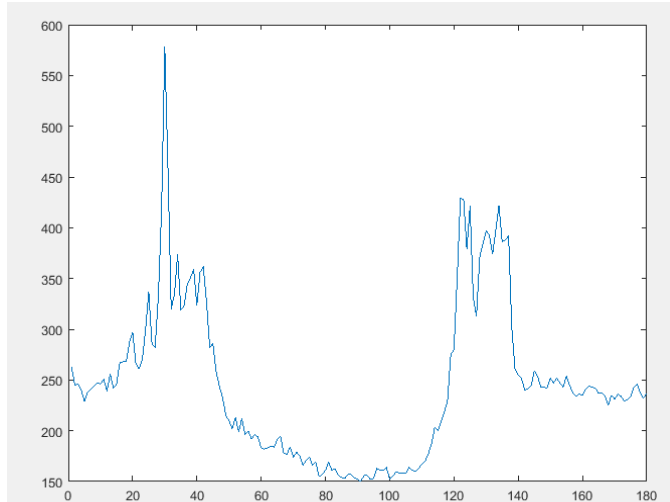
Εκκινούμε λοιπόν, αναζητώντας τις μέγιστες τιμές σε κάθε στήλη του αθροιστικού πίνακα. Δηλαδή αναζητούμε για κάθε μια τιμή του άξονα  $x'$  (τιμές του  $\theta$ ) βρίσκουμε την αντίστοιχη  $yy'$  max (μέγιστη τιμή του  $\rho$ ). Προβάλλουμε δηλαδή τις μέγιστες τιμές του  $\rho$ , ως προς τον οριζόντιο άξονα.

Οι τιμές αυτές απεικονίζονται στο παρακάτω διάγραμμα του παραμετρικού χώρου, ως προς  $\theta$  ( $x'$ ) και ως προς  $\rho$  ( $y'$ ).



**Εικόνα 4.23** Τα max peaks του accumulator bin  
Τα max peaks κάθε γωνίας ( στο εύρος [-90 : 89] ) του accumulator bin

Αν απεικονίσουμε τις ίδιες τιμές σε ένα πιο απλό διάγραμμα κατανομής παρατηρούμε ότι σε ορισμένες περιπτώσεις η μέγιστη τιμή του πίνακα είναι εμφανής (αριστερή κατανομή) ενώ σε άλλες όχι (δεξιά κατανομή). Προσπαθήσαμε λοιπόν να φτιάξουμε μια συνάρτηση η οποία θα αποφασίζει κάθε φορά μέσα από ένα εύρος peaks με γειτονικές τιμές την επιλογή των κύριων γωνιών.



**Εικόνα 4.24** Γράφημα των τιμών που παίρνουν τα max peaks

Σαν όρισμα δέχεται τον πίνακα των σημείων (peaks) από την παραπάνω εικόνα, σημεία που αντιστοιχούν στις κύριες κατευθύνσεις μας (strong lines).

Στη συνέχεια προσπαθήσαμε με βάση την εκάστοτε κατανομή να δημιουργήσουμε δύο ομάδες από peaks, τα οποία θα αποθηκεύονται σε δύο αντίστοιχους πίνακες.

Πρώτα αναζητούμε από τον αθροιστικό πίνακα (accumulator bin) το σημείο με τον μεγαλύτερο αριθμό ψήφων. Έπειτα ψάχνουμε γειτονικά σημεία (peaks) που ικανοποιούν τις εξής συνθήκες ως προς το max peak:

α)  $-10 \leq \theta \leq 10$  και

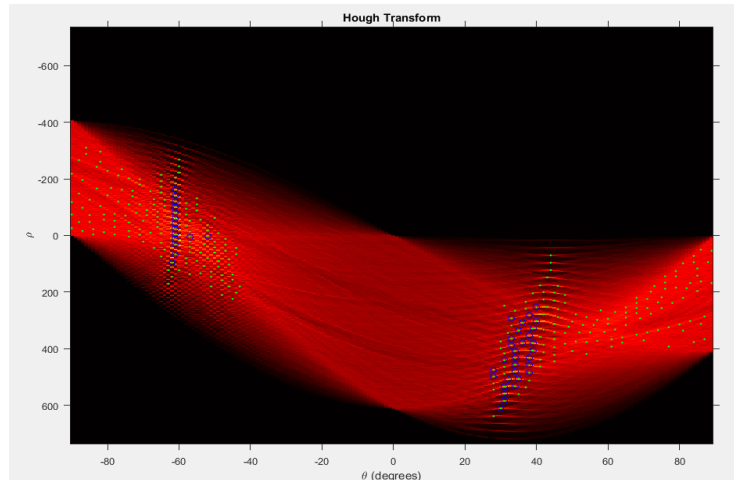
β) κάθε γειτονικό σημείο που θα έχει τιμή (τιμή του rho) μεγαλύτερη ή ίση· απ' το 60% του max peak

Με αυτόν τον τρόπο προσπαθούμε να αναζητήσουμε γειτονικά σημεία (πχ. της αριστερής κατανομής) σε ένα εύρος  $20^\circ$  μοιρών τα οποία έχουν αριθμό ψήφων στο αθροιστικό πίνακα 60% του max peak value.

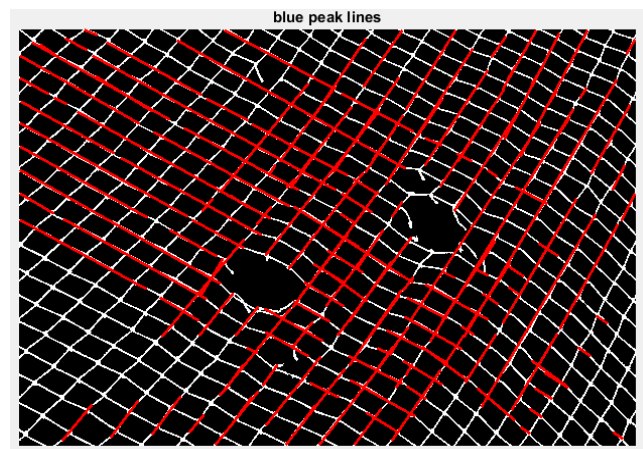
Αφού εντοπιστούν τα σημεία τα αποθηκεύομαι σε έναν πίνακα (p1), και τέλος βρίσκουμε τον μέσο όρο της κλίσης τους ως προς την εικόνα.

Στη συνέχεια αναζητούμε peak τα οποία θα έχουν διαφορά κλίσης τουλάχιστον  $30^\circ$  μοίρες από τον μέσο όρο του πίνακα p1. Διασφαλίζοντας έτσι ότι, τα επόμενα αναζητούμενα peaks θα βρίσκεται στην

αντίθετη κατεύθυνση από εκείνα του πρώτου πίνακα. Και επαναλαμβάνουμε την ως άνω διαδικασία αποθηκεύοντας τα αποτελέσματα σε έναν άλλο πίνακα (p2). Οι δύο ομάδες από peaks φαίνονται στην παρακάτω εικόνα:



(a)



(b)

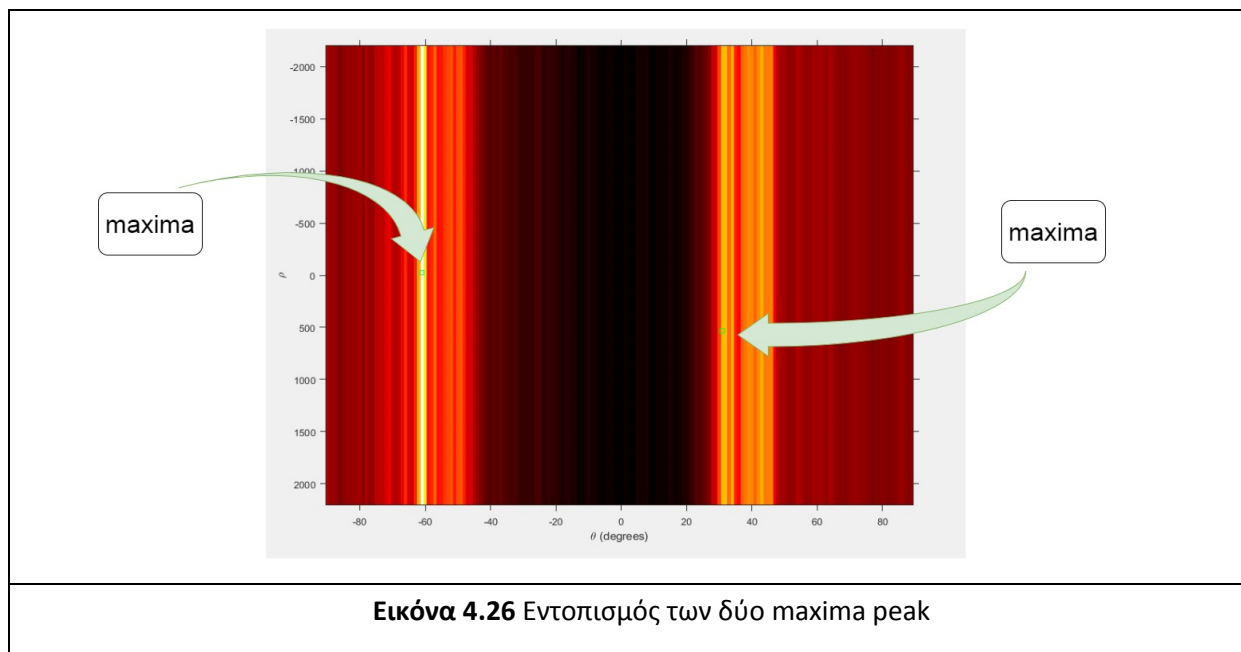
**Εικόνα 4.25** Blue peaks/ Αντίστοιχες ευθείες

(a) Με πράσινο χρώμα είναι τα peaks των strong lines με μπλε χρώμα είναι τα peaks που ικανοποιούν την παραπάνω συνθήκη στον Hough space

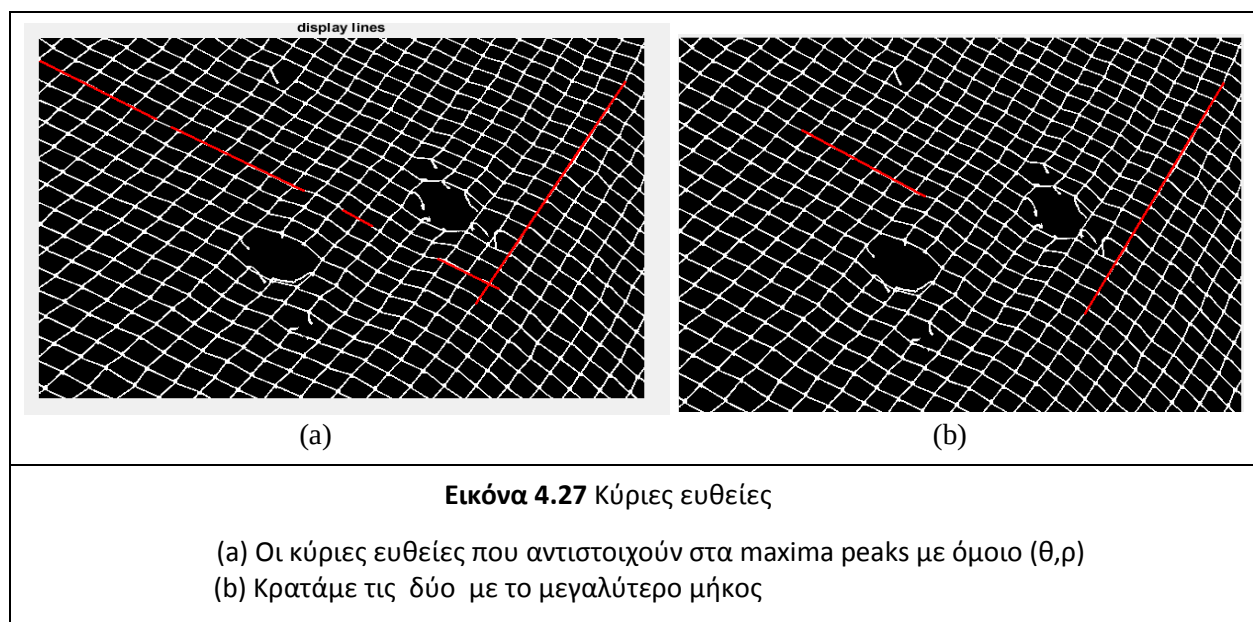
(b) Οι γραμμές αντιστοιχούν στα μπλε peaks στον Image space

Τέλος, αφού έχουμε βρει τις δύο ομάδες που μας ικανοποιούν, επιλέγουμε τα δύο peak (ένα από κάθε ομάδα) με την μεγαλύτερη τιμή ψήφων. Κάθε peak θα αντιστοιχεί και σε μια κύρια ευθεία της εικόνας. Οι κλίσεις (theta) των δύο αντίρροπων ευθειών θα εκπροσωπούν τις δύο κύριες κατευθύνσεις

που θα έχει το πλέγμα μας. Τα δύο σημεία πάνω στον παραμετρικό χώρο με τις μέγιστες τιμές απεικονίζονται στην Εικόνα 4.26, ενώ η αντιστοιχία των σημείων στον χώρο της εικόνα απεικονίζεται στην Εικόνα 4.27:



Παρατηρούμε ότι για την μια κατεύθυνση υπάρχουν τέσσερις ευθύγραμμα τμήματα που μας ικανοποιούν, με κοινό ζεύγος  $(\theta, \rho)$ . Γι' αυτό κρατάμε ένα από αυτά, μιας και δεν δημιουργεί κάποια διαφορά στο τελικό αποτέλεσμα. Άρα τελικώς οι δύο κύριες ευθείες μας απεικονίζονται στην Εικόνα 4.27 (b). Οι ευθείες αυτές αντιστοιχούν στις δύο κύριες κατευθύνσεις που έχει το δίκτυ της εικόνας μας.





#### 4.2.4 Εντοπισμός μέσης απόστασης διχτυών

Αφού έχουμε βρει τις δύο κύριες κατευθύνσεις των ευθειών, σκοπός μας πλέον είναι να βρούμε το μήκος του “πλέγματος” του διχτυού. Αυτό μπορεί να βρεθεί απ’ την μέση απόσταση μεταξύ παράλληλων γραμμών που σχηματίζουν το πλέγμα.

Αρχικά βρίσκουμε τον μέσο όρο της κλίσης των δύο κύριων ευθειών. Έπειτα με βάση αυτή την τιμή χωρίσαμε όλες τις ευθείες που υπάρχουν στην εικόνα σε δύο μεγάλες ομάδες (πχ κάθετες – οριζόντιες).

Όπου αντικαθιστούμε το ζεύγος  $(\theta, \rho)$  σε κάθε ευθεία με το αντίστοιχο  $(\theta, \rho)$  της ομάδας στην οποία ανήκει.

Έτσι δημιουργούμε έναν νέο πίνακα με μονάχα δύο ζεύγη  $(\theta, \rho)$ , αυτά των κύριων ευθειών.

Στη συνέχεια εκχωρούμε τον παραπάνω πίνακα στην συνάρτηση `get_net_distance`, στην οποία υλοποιείται η υπόλοιπη διαδικασία. Η εύρεση δηλαδή της μέσης απόστασης που έχουν τα δίχτυα.

Εκκινούμε υπολογίζοντας για κάθε μια γραμμή της εικόνας τους συντελεστές  $a, b$ , ενώ παράλληλα αποθηκεύουμε και το μήκος του κάθε ευθυγράμμου τμήματος.

Ισχύει η εξίσωση:

$$y = a \cdot x + b \text{ με}$$

$$a = \frac{y_2 - y_1}{x_2 - x_1} \text{ και}$$

$$b = y_1 - a \cdot x_1 \text{ ή } b = y_2 - a \cdot x_2.$$

Όπου  $p_1: (x_1, y_1)$  και  $p_2: (x_2, y_2)$  ορίζουμε τα σημεία αρχής και τέλους του ευθυγράμμου τμήματος.

Στη συνέχεια κρατάμε μονάχα όσες ευθείες έχουν μήκος μεγαλύτερο από τον μέσο όρο, καθώς παρατηρούμε ότι τα μικρά τμήματα εντοπίζονται σε διάσπαρτα σημεία και δεν είναι παράλληλα, σε αντίθεση με τα μεγάλα που είναι πάντα παράλληλα.

Τέλος χωρίζουμε τις ευθείες του πίνακα σε δύο ομάδες με βάση τους δύο κύριους συντελεστές κλίσης (γωνίας) που έχουμε εντοπίσει.

Μετά, επαναλαμβάνουμε την παρακάτω διαδικασία για κάθε ομάδα γραμμών και για κάθε γραμμή της ομάδας ξεχωριστά:

Σε μια ευθεία βρίσκουμε το μέσο της και φέρουμε σε αυτό το σημείο μια κάθετη γραμμή.

$$\text{Μέσο σημείο του ευθύγραμμου τμήματος : } p_0 = \frac{p_1 + p_2}{2}$$

Η κάθετη έχει συντελεστή  $a' = \frac{1}{a}$ , ενώ το  $b$  βρίσκεται βάζοντας στον τύπο τις

$$\text{συντεταγμένες του } p_0: b' = y_0 - a \cdot x_0.$$

Μετά, βρίσκουμε τα σημεία τομής της κάθετης με τα υπόλοιπα παράλληλα ευθύγραμμα τμήματα της ομάδας.

Αν η κάθετη γραμμή που φέραμε δεν τέμνει κάποια άλλη ευθεία, δεν την επεκτείνουμε, γιατί πιθανόν τα γειτονικά ευθύγραμμα τμήματα βρίσκεται μακριά και δεν είναι παράλληλα με το αρχικό τμήμα. Στη συνέχεια βρίσκουμε τις αποστάσεις μεταξύ των τεμνόμενων γραμμών.

Επειδή οι αποστάσεις διαφέρουν λίγο μεταξύ τους, κάνουμε μια στρογγυλοποίηση στην κοντινότερη ακέραια τιμή που είναι πολλαπλάσιο του 5 (θεωρούμε ότι το πλέγμα είναι τουλάχιστον 5 pixel).

Στο τέλος κρατάμε κάθε φορά την πιο συχνή τιμή (απόστασης) ανάμεσα στις τεμνόμενες ευθείες.

Αφού η διαδικασία επαναληφθεί για όλα τα ευθύγραμμα τμήματα και των δύο ομάδων.

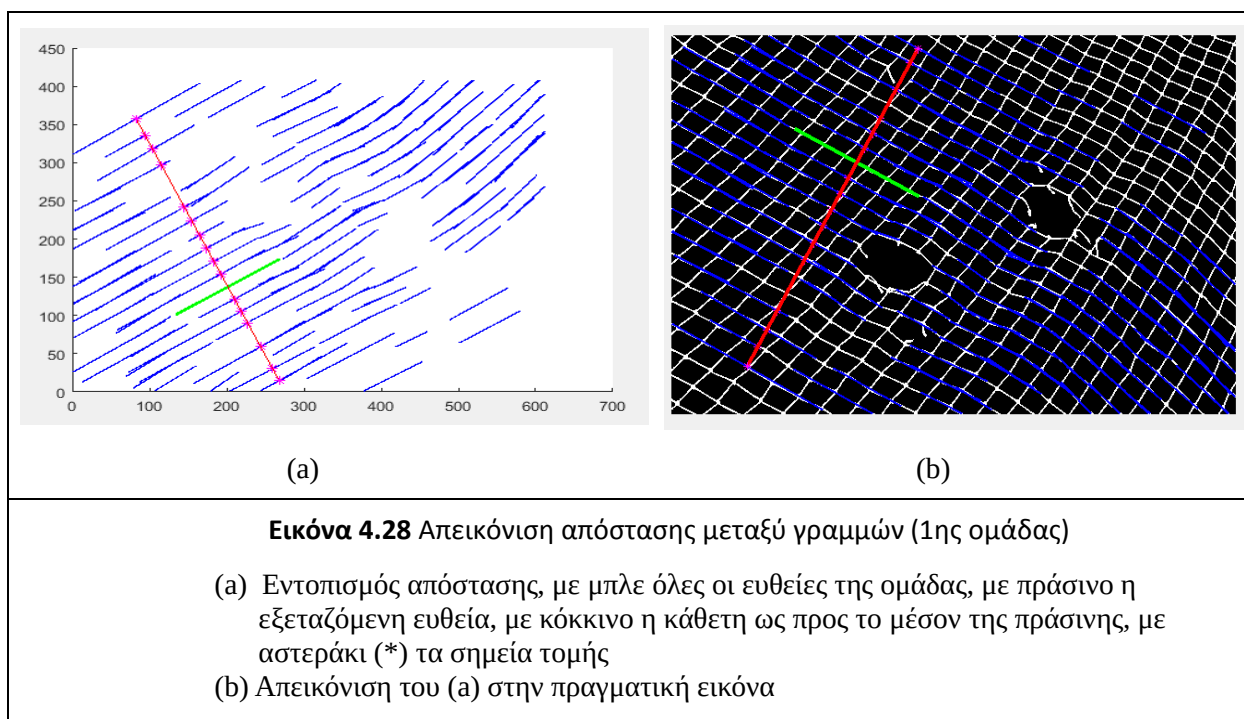
Επιστρέφουμε τον μέσο όρο των δύο ομάδων ( $\frac{dist1+dist2}{2}$ ) ως τη μέση απόσταση μεταξύ παράλληλων γραμμών.

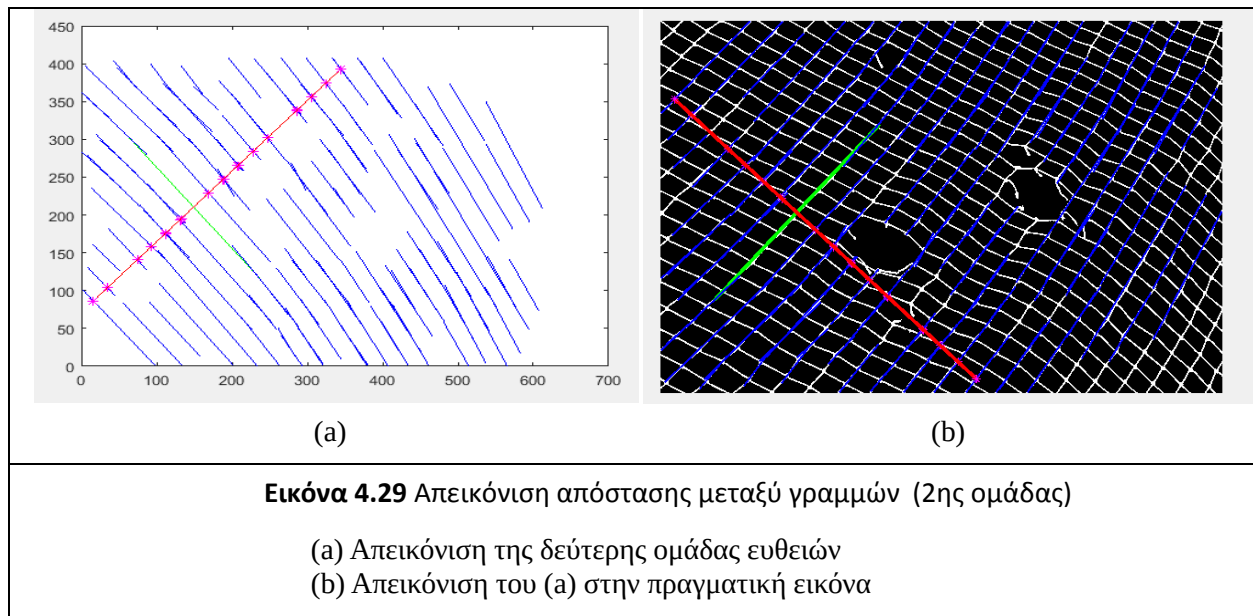
Δηλαδή το μήκος του πλέγματος.

Στα παρακάτω παραδείγματα (Εικόνες 4.28 και 4.29) δείχνουμε δύο ενδεικτικά ευθύγραμμα τμήματα των δύο κύριων κατευθύνσεων του πλέγματος. Στα αριστερά απεικονίζουμε τις ευθείες σε ένα plot ενώ στα δεξιά πως αυτές προβάλλονται πάνω στην εικόνα.

Έστω με πράσινο χρώμα ένα ευθύγραμμο τμήμα του πίνακα που εξετάζουμε, φέρουμε σε αυτό μια κάθετη (κόκκινο χρώμα) στο μέσον του, η οποία τέμνει κάποιες γειτονικές ευθείες (μπλε). Τα σημεία τομής παρουσιάζονται με το χρώμα magenta.

Παρατηρούμε ότι η πιο συχνή απόσταση των σημείων τομής είναι αυτή μεταξύ παραλλήλων.





Τέλος παρουσιάζουμε κάποια διαγράμματα που έχουν σχέση με τις αποστάσεις των γραμμών στην εικόνα.

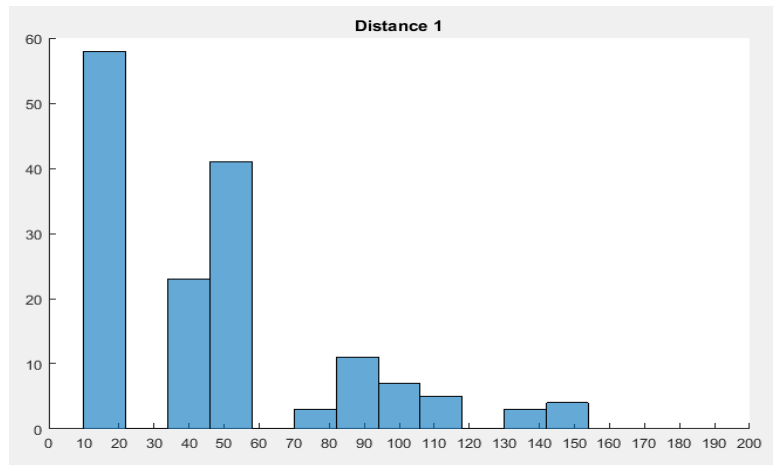
Στα αριστερά διαγράμματα με το ανοικτό μπλε βλέπουμε την κατανομή των τιμών στις δύο ομάδες ευθειών.

Είναι εμφανές πως αθροιστικά οι περισσότερες τιμές των αποστάσεων παίρνουν μικρές τιμές, ενώ ελάχιστες είναι οι τιμές των αποστάσεων με μεγάλη τιμή (η οποία δηλώνει και την ύπαρξη τρύπας), ενώ με τα σκούρα μπλε διαγράμματα απεικονίζουμε αναλυτικά την τιμή που έχει η απόστασης για κάθε ένα ευθύγραμμο τμήμα.

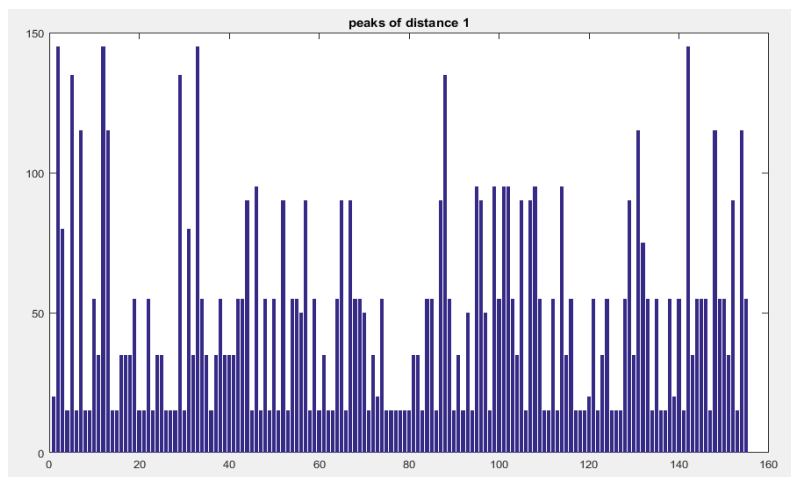
---

Διαγράμματα 1ης ομάδας ευθειών

---



(a)



(b)

**Εικόνα 4.30** Ιστογράμματα 1ης ομάδας ευθειών

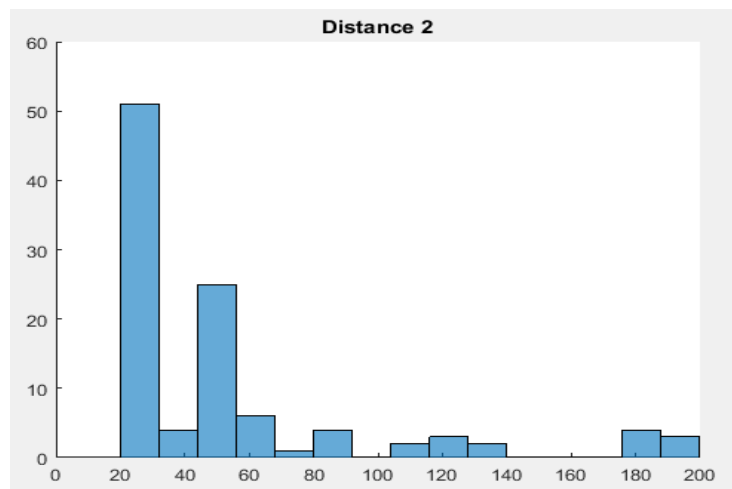
(a) Διάγραμμα κατανομής αποστάσεων, ο άξονας  $x'$  αντιστοιχεί στην τιμή της απόστασης ο  $y'$  αντιστοιχίζεται στον πλήθος των γραμμών που έχουν αυτή την τιμή

(b) Αναλυτικό διάγραμμα της τιμής της απόστασης για κάθε ευθύγραμμο τμήμα, ο  $x'$  άξονας αντιστοιχίζεται στον αριθμό των ευθειών που ανήκουν στην πρώτη ομάδα, ο  $y'$  άξονας αντιστοιχίζεται στην τιμή της απόστασης

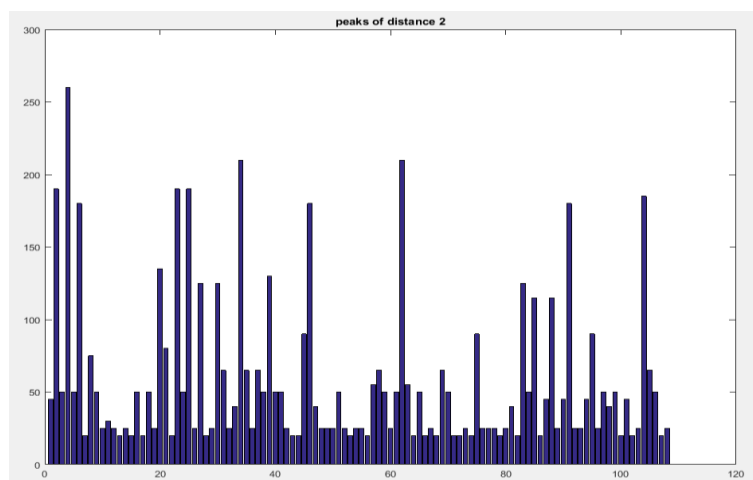
---

Διαγράμματα 2ης ομάδας ευθειών:

---



(a)



(b)

**Εικόνα 4.31** Ιστογράμματα 2ης ομάδας ευθειών

(a) Διάγραμμα κατανομής αποστάσεων, ο άξονας  $x'$  αντιστοιχεί στην τιμή της απόστασης ο  $yy'$  αντιστοιχίζεται στον πλήθος των γραμμών που έχουν αυτή την τιμή

(b) Αναλυτικό διάγραμμα της τιμής της απόστασης για κάθε ευθύγραμμο τμήμα, ο  $x'$  άξονας αντιστοιχίζεται στον αριθμό των ευθειών που ανήκουν στην δεύτερη ομάδα, ο  $yy'$  άξονας αντιστοιχίζεται στην τιμή της απόστασης

Συμπερασματικά από τα σχήματα 4.30 (a) και 4.31 (a) παρατηρούμε ότι η πλειοψηφία των τιμών των αποστάσεων συγκεντρώνεται στην αριστερή πλευρά των ιστογραμμάτων, παίρνοντας μικρή τιμή.

Για την ομάδα ευθειών 1 (Εικόνα 4.30 (a)) πάνω από 50 ευθύγραμμα τμήματα παίρνουν τιμή μέσα στο εύρος [10-20] ενώ τα υπόλοιπα κυμαίνονται σε τιμές έως και 60. Μια μικρή μειοψηφία είναι αυτή που συγκεντρώνει τιμές αρκετά μεγάλες, σε εύρος [140-150].

Αντίστοιχα και στην ομάδα ευθειών 2 (Εικόνα 4.31 (a)) η πλειοψηφία των τιμών των αποστάσεων συγκεντρώνεται σε τιμές από [20-60], ενώ μια μικρή μειοψηφία ευθυγράμμων τμημάτων συγκεντρώνει τιμές αρκετά μεγάλες [180-200].

Συμπερασματικά μπορούμε να πούμε πως η μεθοδολογία εντοπισμού των αποστάσεων επιβεβαιώνει την ίδια την εικόνα. Στο πλέγμα της εικόνας, η πλειοψηφία των αποστάσεων του δικτυού έχει σχετικά παρόμοια (μικρή) τιμή, ενώ υπάρχουν συγκεκριμένα σημεία (ασυνέχειες-τρύπες) τα οποία παίρνουν υψηλές τιμές λόγω της απόστασης από το εικονικό πλέγμα που δημιουργήσαμε.

#### 4.2.5 Εντοπισμός πιθανών τρυπών

Αν έχουμε βρει τις κατάλληλες γραμμές και το μέγεθος του πλέγματος (`hole_threshold`), η εύρεση των τρυπών είναι πλέον εύκολη.

Η διαδικασία υλοποιείται μέσα σε μια συνάρτηση με όνομα `get_holes` και έχει ως εξής:

Διατρέχουμε όλες τις ευθείες της εικόνας μας και αποθηκεύουμε το μήκος της κάθε μιας σε ένα πίνακα.

Αφαιρούμε όσες έχουν μικρό μέγεθος καθώς τα μικρά ευθύγραμμα τμήματα στην πλειοψηφία τους δεν είναι παράλληλα με τα δίκτυα (μπορεί να είναι κομμένες άκρες του δικτυού ή θόρυβος).

Το όριο που θέτουμε για τον διαχωρισμό είναι το  $\text{length} < 2 \cdot \text{hole\_threshold}$ , όπου `hole_threshold` είναι το μέγεθος του πλέγματος που βρήκαμε στο κεφάλαιο 4.2.4. Θεωρούμε δηλαδή ότι μια τρύπα θα έχει τουλάχιστον διπλάσιο μέγεθος από την απόσταση δύο ευθειών του πλέγματος.

Στη συνέχεια βρίσκουμε όπως και πριν τους συντελεστές της κάθε ευθείας και προεκτείνουμε κάθε μια μέχρι το τέλος της εικόνας.

Αν μια εικόνα έχει μέγεθος [N,M] θέτουμε  $x=1:N$  και υπολογίζουμε το  $y$  για τιμές εντός  $1 < y < M$  (για να μην βγουν οι ευθείες εκτός εικόνας) :

$$y = a \cdot x + b$$

$$a = \frac{p2(2) - p1(2)}{p2(1) - p1(1)}$$

$$b = p1(2) - a \cdot p1(1);$$

Όπου  $p1: (x1,y1)$  και  $p2: (x2,y2)$  αρχής και τέλους του ευθυγράμμου τμήματος.

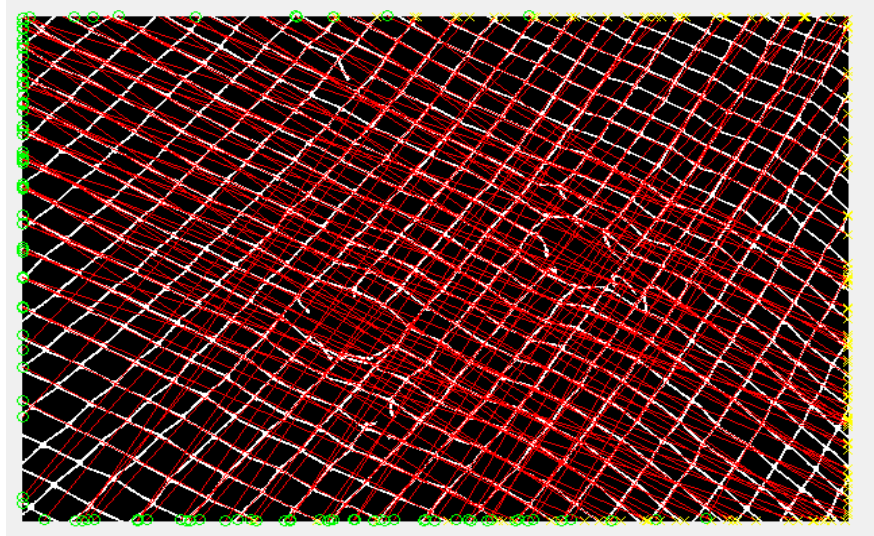
Δεδομένου ότι η εικόνα μας είναι ασπρόμαυρη (binary) με το προσκήνιο (δίχτυα) να έχει τιμή 1 και το φόντο τιμή 0, κάθε άσπρο pixel θα αντιστοιχίζεται με κάποιο σημείο του δικτυού της εικόνας. Απο-  
θηκεύουμε λοιπόν τα σημεία της εικόνας με τιμή διάφορη του μηδενός σε έναν πίνακα  $[X,Y]$ .

Έπειτα υπολογίζουμε την απόσταση κάθε σημείου  $(x,y)$  μιας ευθείας του πλέγματος από τα σημεία  $(X,Y)$  του δικτυού της εικόνας.

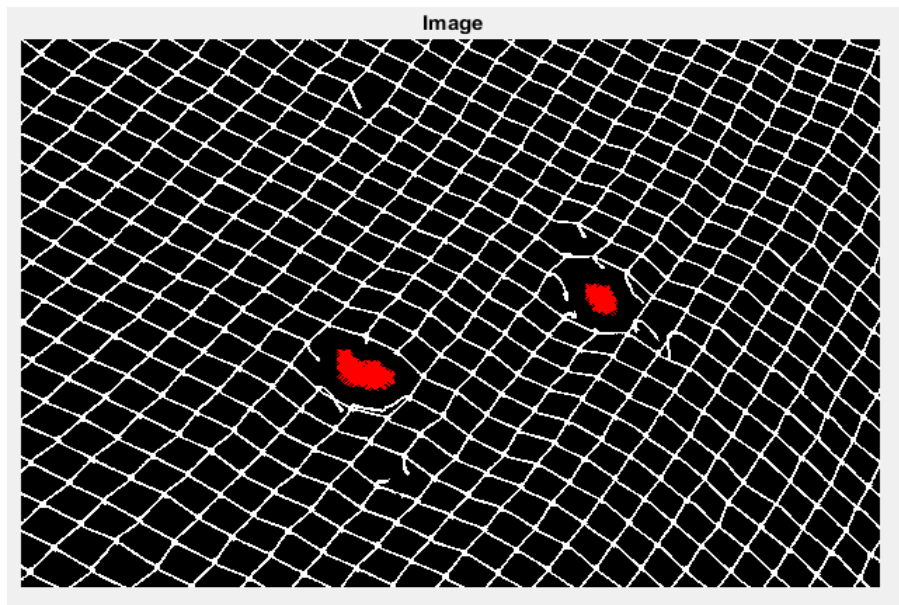
Στο τέλος κρατάμε για κάθε σημείο της ευθείας την ελάχιστη απόσταση από το δίχτυ.

Χαρακτηρίζουμε ως τρύπες όσα σημεία της ευθείας απέχουν πάνω από το όριο `hole_threshold`.

Στην εικόνα 4.32 παρουσιάζεται το πλέγμα που έχουμε δημιουργήσει ενώ στην 4.33 οι τρύπες που εντόπισε ο αλγόριθμος.



**Εικόνα 4.32** Το πλέγμα που δημιουργήσαμε



**Εικόνα 4.33** Εντοπισμός τρυπών



## 4.2.6 Αποτελέσματα

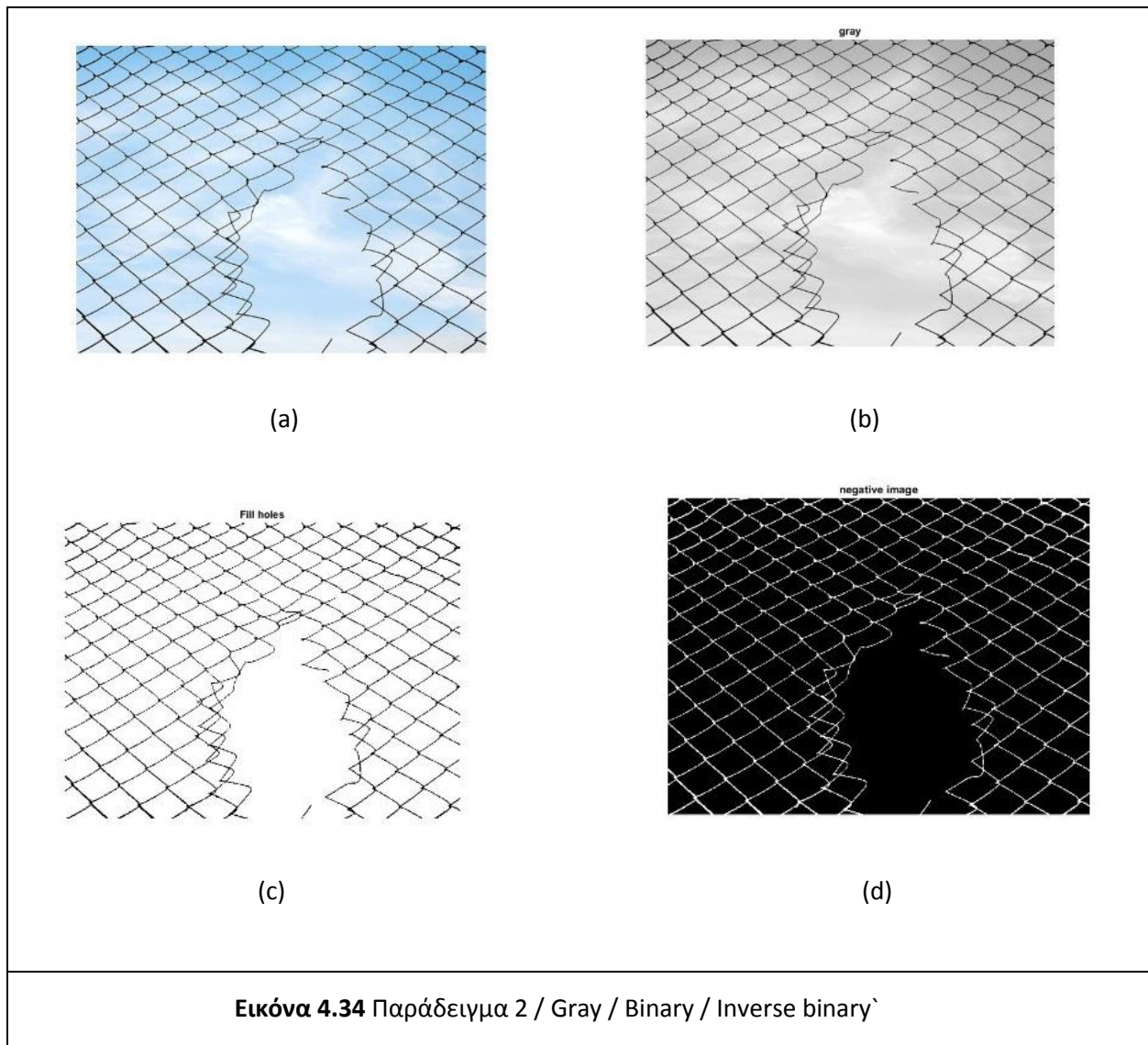
### 4.2.6.1 Τα αποτελέσματα για την δεύτερη εικόνα:

---

#### *Τα αποτελέσματα της εικόνας στο πρώτο στάδιο*

---

Παρακάτω παρουσιάζονται οι εικόνες από την αρχική ανάγνωση στην gray μετάβασή, από εκεί στην binary και τέλος στην αντιστροφή της τελευταίας.

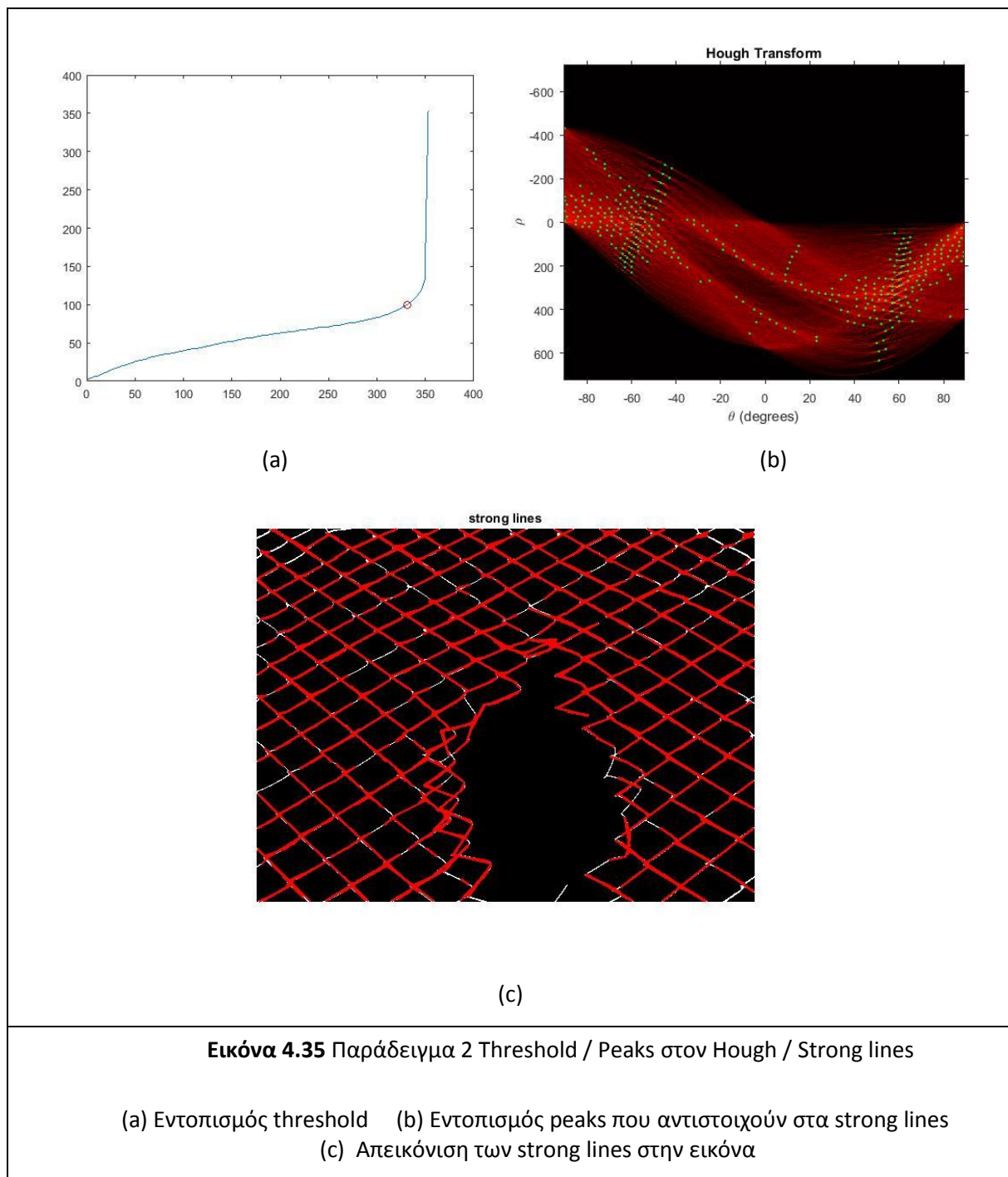


---

*Τα αποτελέσματα στο δεύτερο στάδιο*

---

Η εικόνα 4.35 (a) δείχνει το κατώφλι του οποίου η τιμή είναι 100, η εικόνα 4.35 (b) δείχνει τα αποτελέσματα του Hough transform, και η 4.35 (c) τα strong lines που εντοπίστηκαν με βάση το παραπάνω όριο.



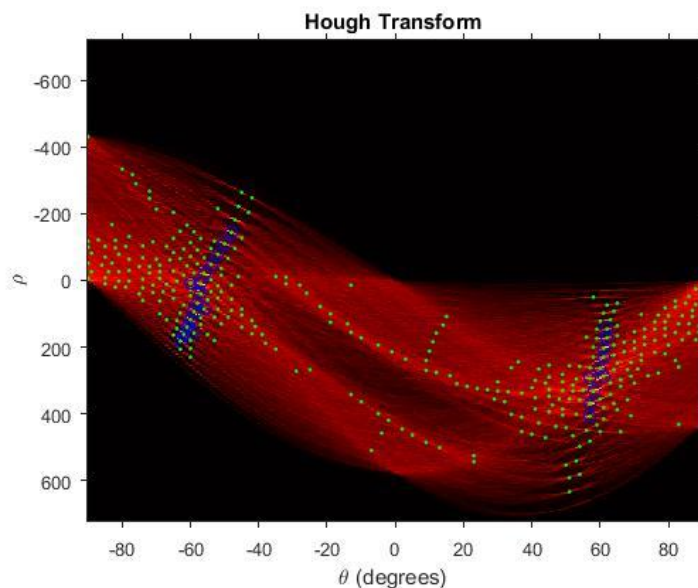
---

### Τα αποτελέσματα στο τρίτο στάδιο

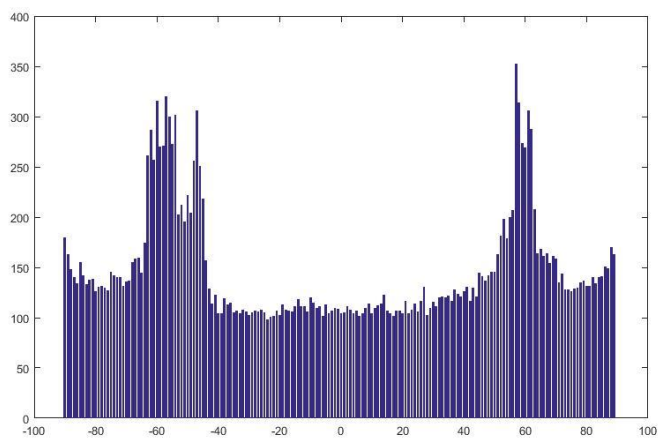
---

Η εικόνα 4.36 (a) δείχνει τα max peaks, παρατηρούμε ότι οι δύο κύριες κατευθύνσεις βρίσκονται κοντά στις -60 και 60 μοίρες.

Στην εικόνα 4.36 (b) παρουσιάζουμε το ιστόγραμμα της κατανομής ( $\theta, \rho$ ), όπου παρατηρούμε στις αρνητικές μοίρες το εύρος των max peak είναι αρκετά ευρύ (φτάνει μέχρι το -50) ενώ στις θετικές μοίρες τα max peak συγκεντρώνονται γύρω από τις 60 μοίρες.



(a)

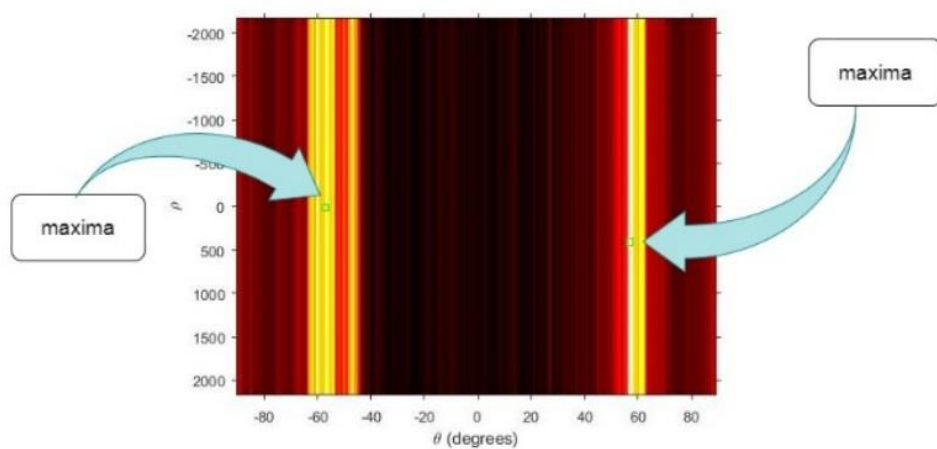


(b)

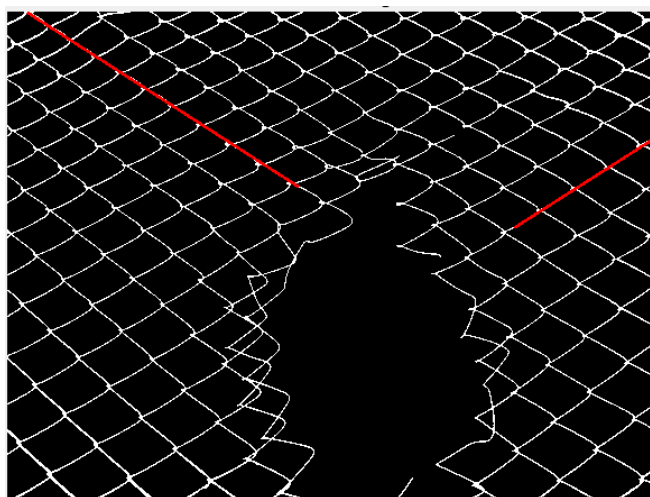
**Εικόνα 4.36** Παράδειγμα 2 Blue Peaks / Ιστόγραμμα max peaks

- (a) Εντοπισμός των max peaks
- (b) Διάγραμμα των max peaks, όσα έχουν μπλε χρώμα στην (a)

Στην εικόνα 4.37 (a) παρουσιάζονται οι δύο κύριες γωνίες: Για τα αρνητικά  $(\theta, \rho) = (-57, 10)$  ενώ για τα θετικά  $(\theta, \rho) = (57, 408)$ . Η εικόνα 4.37 (b) δείχνει τις κύριες ευθείες του δικτυού οι οποίες αντιστοιχούν στα παραπάνω ζεύγη  $(\theta, \rho)$ .



(a)

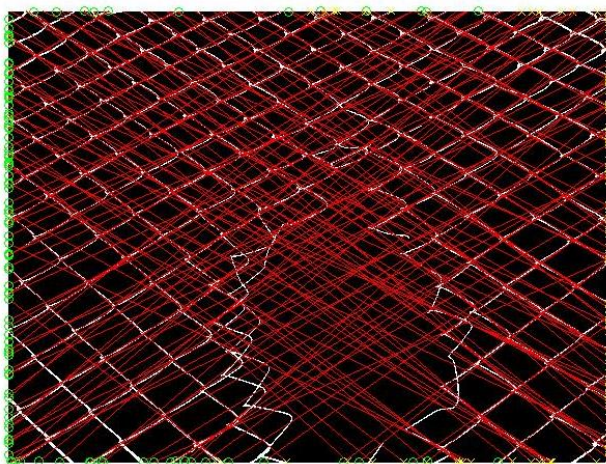


(b)

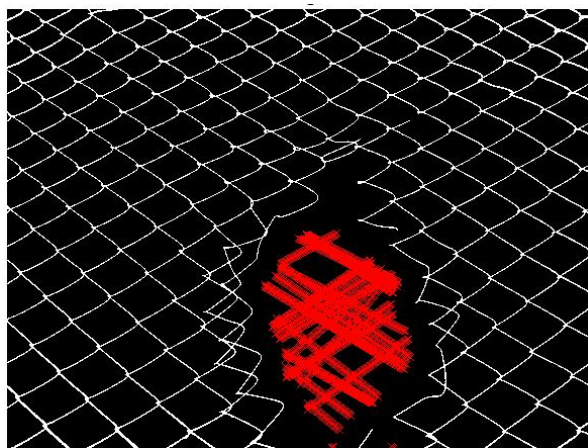
**Εικόνα 4.37** Παράδειγμα 2 Maxima / Κύριες κατευθύνσεις

(a) Απεικόνιση των δύο maxima (b) Οι γραμμές κύριων κατευθύνσεων

Στην εικόνα 4.38 (a) παρουσιάζεται το πλέγμα που υλοποιήσαμε με τιμή μέσης απόστασης δικτύου ίση με 30. Τέλος στην εικόνα 4.38 (b) παρουσιάζεται η εντοπισμένη τρύπα.



(a)



(b)

**Εικόνα 4.38** Παράδειγμα 2 πλέγμα / Εντοπισμός τρύπας

(a) Το πλέγμα που δημιουργήσαμε    (b) Η εντοπισμένη τρύπα



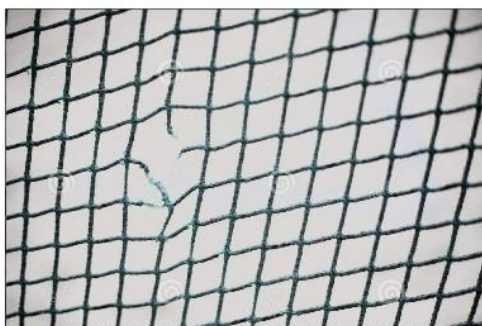
#### 4.2.6.2 Τα αποτελέσματα για την τρίτη εικόνα

---

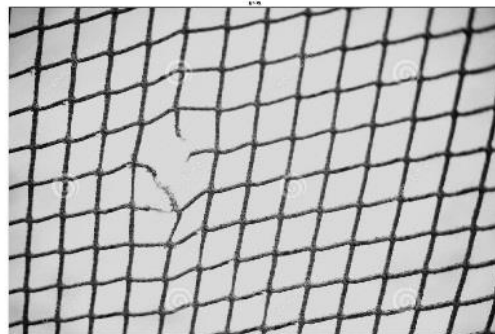
##### *Τα αποτελέσματα της εικόνας στο πρώτο στάδιο*

---

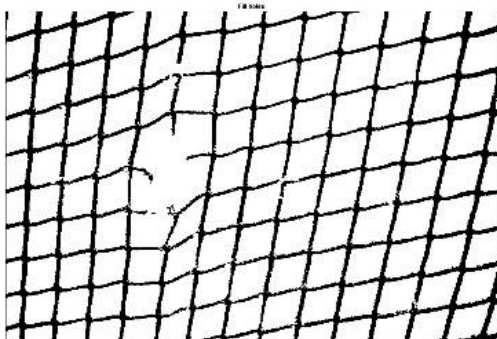
Παρακάτω παρουσιάζονται οι εικόνες από την αρχική ανάγνωση στην gray μετάβασή, από εκεί στην binary και τέλος στην αντιστροφή της τελευταίας.



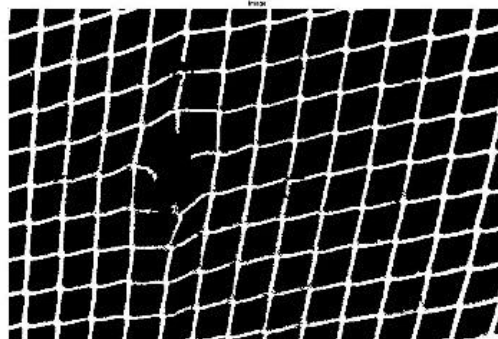
(a)



(b)



(c)



(d)

**Εικόνα 4.39** Παράδειγμα 3 / Gray / Binary / Inverse binary

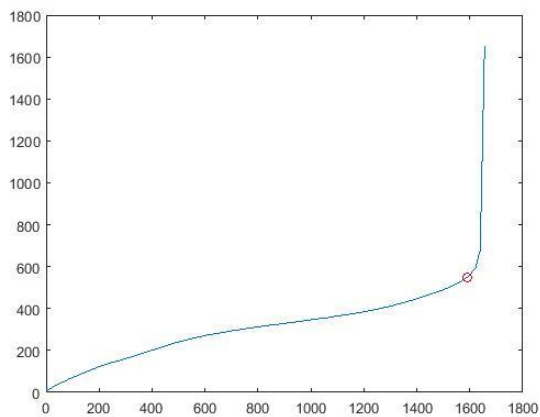
(a) Αρχική εικόνα   (b) Gray εικόνα   (c) Binary εικόνα   (d) Αντεστραμμένη binary

---

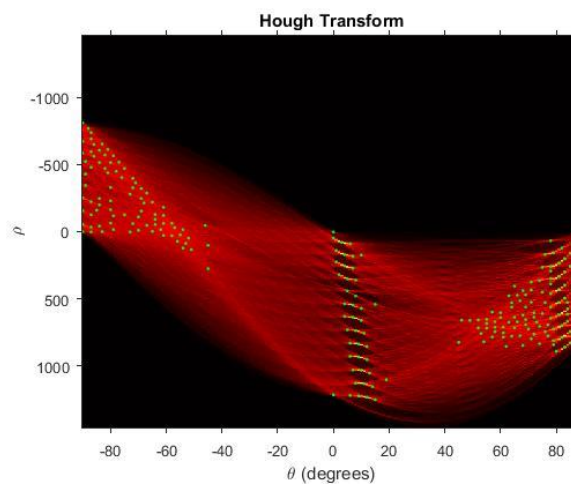
### Τα αποτελέσματα στο δεύτερο στάδιο

---

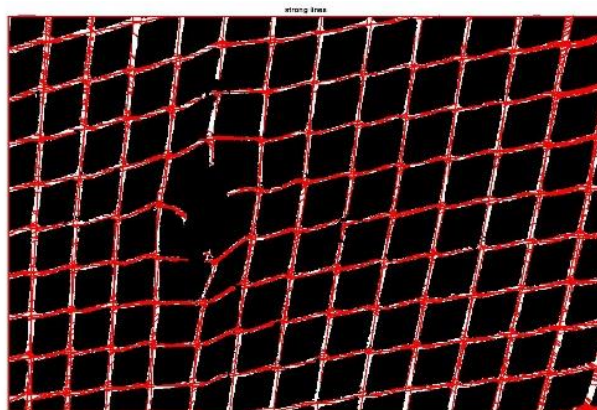
Στην εικόνα 4.40 (a) φαίνεται το κατώφλι με κόκκινο χρώμα το οποίο έχει τιμή ίση με 549, στην εικόνα 4.40 (b) απεικονίζεται το αποτέλεσμα του Hough transform και στην 4.40 (c) τα strong lines. Ευθείες που αντιστοιχούν στα παραπάνω peaks.



(a)



(b)



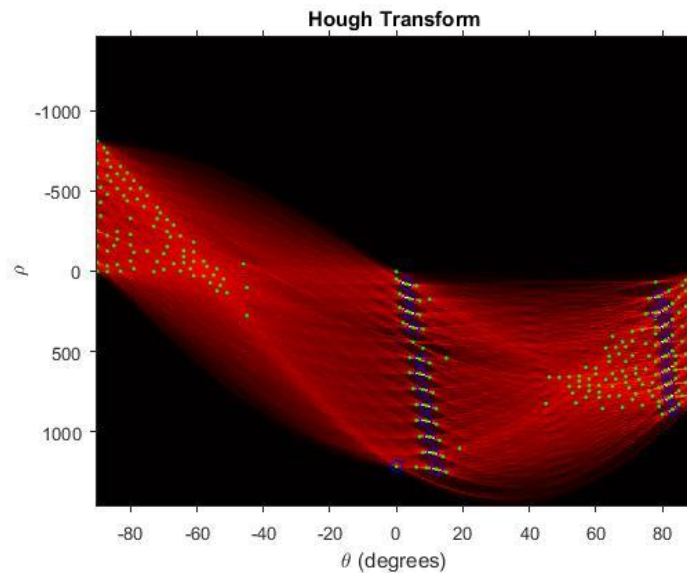
(c)



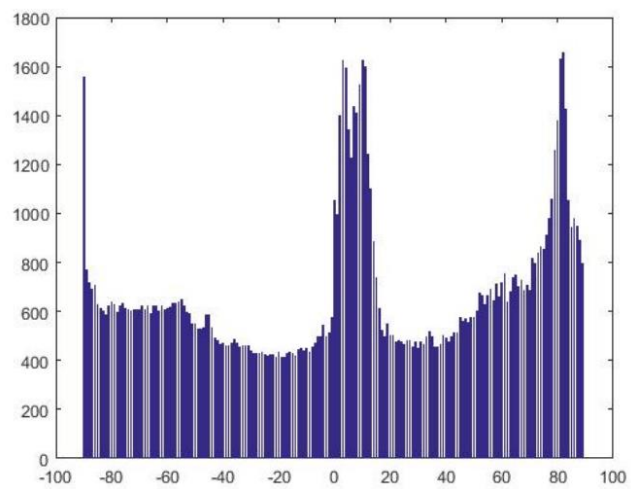
**Εικόνα 4.40** Παράδειγμα 3 Threshold / Max peaks / Αντίστοιχες ευθείες

(a) Εντοπισμός threshold (b) Τα peaks με πράσινο χρώμα των strong lines

(c) Η απεικόνιση των strong lines στην εικόνα\



(a)



(b)

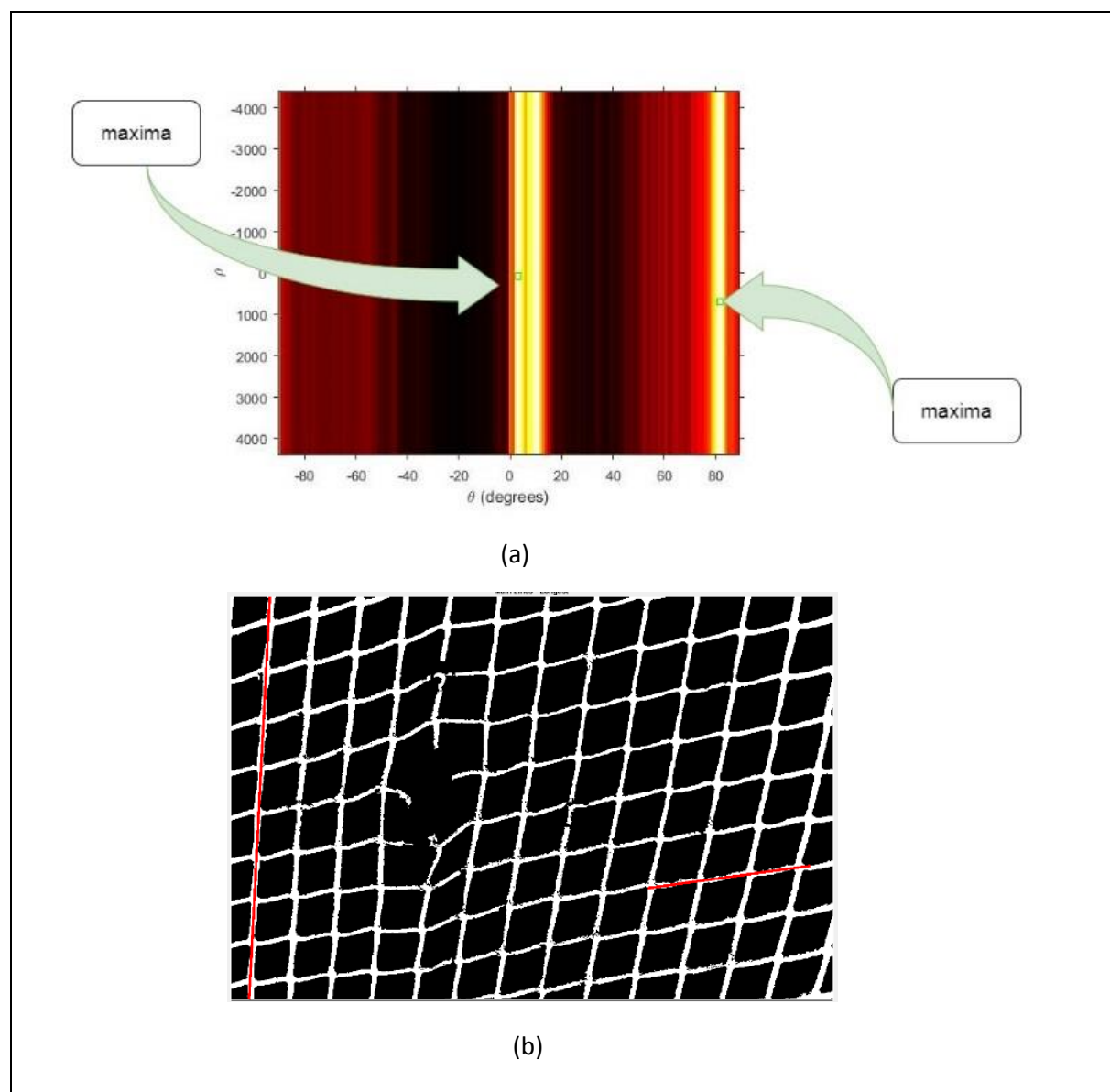
**Εικόνα 4.41** Παράδειγμα 3 Blue peaks / Ιστόγραμμα max peaks

(a) Εντοπισμός των max peaks

(b) Διάγραμμα των max peaks, όσα έχουν μπλε χρώμα στην (a)

Η εικόνα 4.41 (a) δείχνει τα μέγιστα σημεία του παραμετρικού χώρου (max peaks). Παρατηρούμε ότι και οι δύο κύριες κατευθύνσεις βρίσκονται στον θετικό άξονα, η μια κοντά στις 10 μοίρες και η άλλη κοντά στις 90 μοίρες. Αυτό συμβαίνει για το λόγο ότι οι κατευθύνσεις του πλέγματος δεν είναι διαγώνιες αλλά κάθετες. Στην εικόνα 4.41 (b) παρατηρείται το ιστόγραμμα  $(\theta, \rho)$  των max peaks. Παρατηρούμε τα κύρια peaks να κατανέμονται στον θετικό άξονα με κάποια ελάχιστα να βρίσκονται στον αρνητικό  $(-90^\circ)$ .

Στην εικόνα 4.42 (a) παρουσιάζονται τις δύο κύριες γωνίες: Η πρώτη  $(\theta, \rho) = (3, 80)$  ενώ η δεύτερη  $(\theta, \rho) = (82, 700)$ . Η εικόνα 4.37 (b) δείχνει τις κύριες ευθείες του διχτυού οι οποίες αντιστοιχούν στα παραπάνω ζεύγη  $(\theta, \rho)$ .

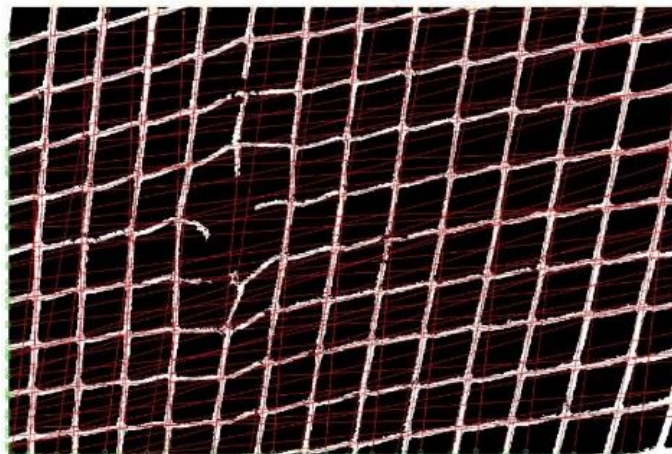


Εικόνα 4.42 Παράδειγμα 3 Maxima / Κύριες κατευθύνσεις

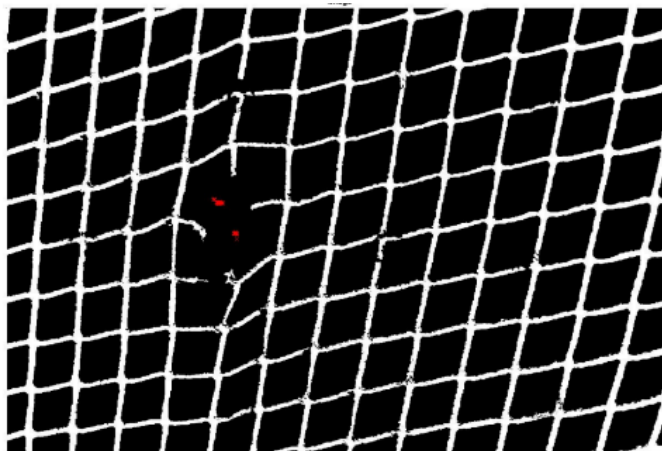
(a) Απεικόνιση των δύο maxima

(b) Οι γραμμές κύριων κατευθύνσεων

Στην εικόνα 4.43 (a) παρουσιάζουμε το πλέγμα που έχουμε δημιουργήσει με τιμή μέσης απόστασης δικτυού ίση με 85 , ενώ στην 4.43 (b) το τελικό αποτέλεσμα με την εντοπισμένη τρύπα.



(a)



(b)

**Εικόνα 4.43** Παράδειγμα 3 Πλέγμα / Εντοπισμός τρύπας

(a) Το πλέγμα που δημιουργήσαμε      (b) Η εντοπισμένη τρύπα

# Κεφάλαιο 5 : Συμπεράσματα και μελλ/κή έρευνα

## 5.1 Σύνοψη - Συμπεράσματα

Στην παρούσα διπλωματική παρουσιάσαμε ένα συνδυαστικό τρόπο ανίχνευσης ασυνεχειών πάνω σε δομές δικτύων ιχθυοκαλλιέργειας. Η αναζήτηση έγινε μέσω δύο μεθόδων διαφορετικής θεωρητικής προσέγγισης, όπου προσπαθήσαμε να εξάγουμε και να απεικονίσουμε τα εσφαλμένα σημεία πάνω στην εικόνα.

Σαν πρώτο βήμα, προσπαθήσαμε να απομονώσουμε τις δομές (foreground) από το υπόλοιπο κάδρο της εικόνας (background). Ο διαχωρισμός έγινε μέσω απλών μεθόδων φιλτραρίσματος καταλήγοντας στο binary αποτέλεσμα που θέλαμε.

Στην συνέχεια εφαρμόσαμε σαν πρώτη μέθοδο μια ολική (global) προσέγγιση πάνω στα αντικείμενα της εικόνας. Αρχικά η εικόνα τμηματοποιήθηκε σε πολλαπλά μέρη, μέσα στα οποία εξετάσαμε το μοτίβο των αντικειμένων που υπάρχουν. Ο διαχωρισμός των δομών σε εσφαλμένες ή σωστές γίνεται βάση της συνολικής κατανομής των εντάσεων.

Με τον καταμερισμό της εικόνας σε sub blocks μπορέσαμε να επεξεργαστούμε τις δομές σε μικρότερο χρονικό διάστημα, λόγω παραλληλισμού, προσφέροντας παράλληλα σχετικά μικρή πολυπλοκότητα.

Με τη μέθοδο αυτή πετυχαίνουμε μια γενική ανίχνευση των σημείων της εικόνας στα οποία ενδεχομένως να υπάρχει ασυνέχεια πάνω στο δίκτυ.

Στη δεύτερη μέθοδο προσεγγίζουμε τα αντικείμενα της εικόνας με έναν πιο λεπτομερή τρόπο, μοντελοποιώντας τις γραμμές του δικτύου σε ένα πλέγμα παράλληλων ευθειών και αναζητώντας την απόσταση μεταξύ των δύο. Η όλη διαδικασία πραγματοποιήθηκε με τη βοήθεια της συνάρτησης του μετασχηματισμού Hough.

Στη συγκεκριμένη μέθοδο, η αναζήτηση σφαλμάτων στα δίκτυα έγινε πιο συγκεκριμένη, δίνοντας μας ακριβή αποτελέσματα. Στα μειονεκτήματα της μεθόδου μπορούμε να περιλάβουμε την αύξηση του κόστους πολυπλοκότητας και τον χρόνο ανίχνευσης.

Με τις πειραματικές υποθαλάσσιες εικόνες που επιλέξαμε, δοκιμάσαμε τις μεθόδους μας στην αναζήτηση δομών διαφορετικού σχήματος και μεγέθους. Επιπλέον οι παραμορφώσεις που δημιουργούνται πάνω στο πλέγμα λόγω κυμάτωσης μας προσφέρουν μια πλειάδα από εναλλακτικά μοτίβο πάνω στο δίκτυ (ανεστραμμένο, τεντωμένο κλπ.) για να εξετάσουμε τα αποτελέσματα μας.

Τα αποτελέσματα που εξάγαμε και από τις δύο μεθόδους ήταν ικανοποιητικά. Η πρώτη μέθοδος μας έδωσε σε μεγάλο βαθμό τις περιοχές στις οποίες υπάρχει ασυνέχεια στο πλέγμα, χωρίς όμως να αποφεύγει εσφαλμένες εντοπίσεις. Οι λανθασμένες εντοπίσεις θεωρούμε πως έγιναν για δύο κυρίως λόγους. Πρωτίστως λόγω της κακής αρχικής επεξεργασίας της εικόνας. Όπως προ είπαμε και στα προηγούμενα κεφάλαια, η αρχική επεξεργασία δεν ήταν ο κάθε αυτός σκοπός της εργασίας μας. Έτσι αρκεστήκαμε σε απλά μοντέλα φιλτραρίσματος, τα οποία μας έδιναν απ' τη μια ευδιάκριτα αποτέλεσμα στις περισσότερες εικόνες, χωρίς όμως να αποφεύγονται εξασθενήσεις πάνω στο δίκτυ. Θεωρούμε πως οι λανθασμένες ανιχνεύσεις στα παραδείγματα 1 & 3 οφείλονται στις στρεβλώσεις του δικτύου (stretching) λόγω της κυμάτωσης του πλέγματος. Κάτι που οδηγεί σε έναν άνισο καταμερισμό του αριθμού των pixel και άρα την ποσότητα της έντασης στα διάφορα τμήματα (παράθυρα) της εικόνας.

Τα αποτελέσματα της δεύτερης μεθόδου θεωρούνται πλήρως επιτυχημένα.

Εν τέλη θεωρούμε πως ο συνδυασμός των δύο μεθόδων μπορεί κινηθεί σε δύο κατευθύνσεις. Σε μια παράλληλη και σε μια σειριακή υλοποίηση. Στην πρώτη, μπορούμε να υλοποιήσουμε τις δύο μεθόδους παράλληλα όπως υλοποιήθηκαν στην παρούσα διπλωματική, εξάγοντας και συγκρίνοντας τα αποτελέσματα στο τέλος (παίρνοντας υπ' όψη τις λανθασμένες ανιχνεύσεις). Στη σειριακή υλοποίηση, μπορούμε να συνδυάσουμε ολική και τοπική επεξεργασία με ένα γραμμικό τρόπο.

Υλοποιούμε αρχικά την ολική επεξεργασία αναζητώντας σε πρώτο επίπεδο, μέσω των εντάσεων, τα σημεία (blocks / sub blocks) που μπορεί να υπάρχουν ασυνέχειες πάνω στην εικόνα. Στην συνέχεια, εφαρμόζουμε μια τοπική - λεπτομερή - αναζήτηση πάνω στα δίκτυα των συγκεκριμένων blocks μέσω του local process της δεύτερης μεθόδου. Η συνδυαστική μέθοδος αυτή μπορεί να μας προσφέρει άριστα αποτελέσματα σε ικανοποιητικό βαθμό αναλογικά με την πολυπλοκότητα και τον χρόνο εκτέλεσης. Και αυτό συμβαίνει για τον λόγο ότι το local process (δεύτερη μέθοδος) το οποίο καθορίζει εν τέλη την ύπαρξη ή μη ασυνέχειας δεν υλοποιείται σε όλο το μήκος της εικόνας, αλλά σε πολύ συγκεκριμένα σημεία, τα οποία έχουν εξαχθεί μέσω της στατιστικής ανάλυσης της ολικής επεξεργασίας (πρώτη μέθοδος). Μειώνοντας αισθητά την πολυπλοκότητα και τον χρόνο εκτέλεσης.

Η διαφορά μεταξύ παράλληλης και σειριακής υλοποίησης έγκειται στο γεγονός ότι, η πρώτη επικεντρώνεται περισσότερο στην δυναμικότητα της ανίχνευσης των αντικειμένων μέσω δύο ανεξάρτητων μεθόδων. Ενώ εναλλακτικά, η σειριακή υλοποίηση δίνει έμφαση στον λεπτομερή απεικόνιση της ασυνέχειας (τρύπας) καθώς και στις χρονικές απαιτήσεις του αλγορίθμου ανίχνευσης.

Στην παρούσα μελέτη υλοποιήσαμε ανεξάρτητα τα δύο σχήματα, ενώ στο τέλος περάσαμε στον παράλληλο συνδυασμό τους. Η σειριακή υλοποίηση αποτελεί μια φυσική συνέχεια της παρούσας ανάλυσης καθιστώντας την πρόκληση για μελλοντική έρευνα μιας και οδηγεί σε βελτιώσεις τόσο στην ανίχνευση όσο και στον εντοπισμό των δομών του πλέγματος.



## 5.2 Προτάσεις για μελλοντική έρευνα

Αρχίζοντας από το τέλος, σίγουρα κύριο μελλοντικό μέλημα μας είναι η υλοποίηση της συνδυαστικής (σειριακής) global και local διαδικασίας όπως αναφέρθηκε παραπάνω, μιας και οι μέθοδοι μπορεί να υλοποιηθούν (αυτόνομα) αλλά δεν προχωρήσαμε σε μια συνδυαστική υλοποίηση. Η εφαρμογή μπορεί να γίνει αρχικά στις τρεις ενδεικτικές εικόνες που παρουσιάσαμε στα παραδείγματα και στη συνέχεια να περάσουμε σε μια εφαρμογή της σειριακής υλοποίησης σε εικόνες παρμένες από κάποιο ROV ή AUV εκεί όπου θα έχουμε εικόνες με κοινά χαρακτηριστικά (φωτεινότητα, απόσταση δικτυού κλπ).

Επιπλέον περιθώριο βελτίωσης σίγουρα χρήζει η πρώτη φάση των αλγορίθμων, μιας και με μια καλύτερη pre-process διαδικασία, φιλτράροντας καλύτερα τον θόρυβο και βελτιώνοντας τις συνθήκες φωτεινότητας που επικρατούν μέσα σε αυτή θεωρούμε πως μπορούμε να εξάγουμε καλύτερα αποτελέσματα και στις δύο μεθόδους.

Ακόμη σημαντικό σημείο είναι η πλήρης αυτοματοποίηση και των δύο αλγορίθμων ανεξαρτήτως συνθηκών που επικρατούν στον υποθαλάσσιο χώρο.

Δεδομένου του γεγονότος ότι χρησιμοποιήθηκαν τρεις διαφορετικού μεγέθους και έντασης εικόνες χρειάστηκε κάθε φορά να προσαρμόζουμε την τιμή του εντοπισμένου threshold στις συνθήκες της εκάστοτε εικόνας. Αυτό συμβαίνει για τους λόγους που έχουμε αναφέρει παραπάνω (απλή pre-process διαδικασία και άνισος καταμερισμός της έντασης λόγω τεντώματος του δικτυού).

Στην πρώτη μέθοδο τα threshold που επιλέγονται (αυτόματα) μαζί με τις ασυνέχειες του δικτυού εντοπίζονται και επιπλέον εσφαλμένα σημεία. Έτσι χρειάστηκε να τοποθετήσουμε στην τιμή του ορίου μια ελαφρώς χαμηλότερη τιμή (manually) από την εντοπισθείσα ώστε να πάρουμε το αποτέλεσμα που θέλουμε.

Στην δεύτερη μέθοδο η προσαρμογή πάνω στο εντοπισθέν όριο έγινε στην τιμή της απόστασης του δικτυού. Αναλόγως την εικόνα που χρησιμοποιούμε η απόσταση των δικτύων αλλάζει (αλλού είναι μεγαλύτερη, αλλού μικρότερη). Έτσι πήραμε κάθε φορά ποσοστιαία την τιμή του εντοπισθέντος ορίου. Για τα παραδείγματα 1 & 2 πήραμε το 80% της τιμής ( $0.8 * \text{threshold}$ ) ενώ για το παράδειγμα 3 πήραμε το  $0.6 * \text{threshold}$  της απόστασης.

Επίσης και οι δύο μέθοδοι μας υλοποιήθηκαν σε στατικές εικόνες. Είναι σημαντικό να τρέξουμε τους αλγορίθμους σε real-time βίντεο, με σκοπό να δούμε πως ανταποκρίνονται στα διαδοχικά frames καθώς και στις πραγματικές υποθαλάσσιες συνθήκες.

Τέλος θα πρέπει να ελέγχουμε πως οι δύο προσεγγίσεις μας θα ενσωματωθούν σε hardware εφαρμογές με σκοπό να χρησιμοποιηθούν από υποθαλάσσια οχήματα ROV και AUV και να μελετηθούν τα προβλήματα που θα εντοπισθούν καθώς και οι μορφές επίλυσής τους.

# Βιβλιογραφία

- [1] Συνέδριο Ιχθυοκαλλιέργειας 2019 (<https://aquaculture.gr>)
- [2] Metian, M., Troell, M., Christensen, V., Steenbeek, J., & Pouil, S. (2020). Mapping diversity of species in global aquaculture. *Reviews in Aquaculture*, 12(2), 1090-1100.
- [3] Aquaculture operations in floating HDPE cages (A field handbook)
- [4] Karimanzira, D., Jacobi, M., Pfützenreuter, T., Rauschenbach, T., Eichhorn, M., Taubert, R., & Ament, C. (2014). First testing of an AUV mission planning and guidance system for water quality monitoring and fish behavior observation in net cage fish farming. *Information Processing in Agriculture*, 1(2), 131-140.
- [5] Borović, B., Vasiljević, A., & Kuljača, O. (2011, January). Potentials of Using Underwater Robotics for Fishing and Fish Farming. In *DEMAT-Development and Evaluation of Maritime Technologies*.
- [6] Osen, O. L., Sandvik, R. I., Rogne, V., & Zhang, H. (2017, September). A novel low cost roV for aquaculture application. In *OCEANS 2017-Anchorage* (pp. 1-7). IEEE.
- [7] Chalkiadakis, V., Papandroulakis, N., Livanos, G., Moirogiorgou, K., Giakos, G., & Zervakis, M. (2017, October). Designing a small-sized autonomous underwater vehicle architecture for regular periodic fish-cage net inspection. In *2017 IEEE International Conference on Imaging Systems and Techniques (IST)* (pp. 1-6). IEEE.
- [8] <https://ir.lib.uth.gr/bitstream/handle/11615/45246/10908.pdf?sequence=1>
- [9] Bao, J., Li, D., Qiao, X., & Rauschenbach, T. (2020). Integrated navigation for autonomous underwater vehicles in aquaculture: A review. *Information Processing in Agriculture*, 7(1), 139-151.
- [10] Karimanzira, D., Jacobi, M., Pfützenreuter, T., Rauschenbach, T., Eichhorn, M., Taubert, R., & Ament, C. (2014). First testing of an AUV mission planning and guidance system for water quality monitoring and fish behavior observation in net cage fish farming. *Information Processing in Agriculture*, 1(2), 131-140.
- [11] Tao, Q., Huang, K., Qin, C., Guo, B., Lam, R., & Zhang, F. (2018, October). Omnidirectional Surface Vehicle for Fish Cage Inspection. In *OCEANS 2018 MTS/IEEE Charleston* (pp. 1-6). IEEE.
- [12] Wei, C., Wang, J., & Chen, G. (2019, November). Denoising and contrast enhancement fusion based on white balance for underwater images. In *2019 International Conference on Image and Video Processing, and Artificial Intelligence* (Vol. 11321, p. 113210E). International Society for Optics and Photonics.
- [13] Chen, Shu, Dongdong Wu et al., 'An Algorithm for Image Restoration Based on Underwater Video Series', 2019 International Conference on Modeling, Simulation, Optimization and Numerical Techniques (SMONT 2019). Atlantis Press, 2019
- [14] Siddiqui, S. A., Salman, A., Malik, M. I., Shafait, F., Mian, A., Shortis, M. R., & Harvey, E. S. (2018). Automatic fish species classification in underwater videos: exploiting pre-trained deep neural network models to compensate for limited labelled data. *ICES Journal of Marine Science*, 75(1), 374-389.

- [15] Shafait, F., Harvey, E. S., Shortis, M. R., Mian, A., Ravanbakhsh, M., Seager, J. W., ... & Edgington, D. R. (2017). Towards automating underwater measurement of fish length: a comparison of semi-automatic and manual stereo-video measurements. *ICES Journal of Marine Science*, 74(6), 1690-1701.
- [16] Hassaballah, M., Abdelmgeid, A. A., & Alshazly, H. A. (2016). Image features detection, description and matching. In *Image Feature Detectors and Descriptors* (pp. 11-45). Springer, Cham.
- [17] Tuytelaars, Tinne, and Krystian Mikolajczyk. *Local invariant feature detectors: a survey*. Now Publishers Inc, 2008.
- [18] Zhang, S., Tian, Q., Huang, Q., Gao, W., & Rui, Y. (2014). USB: Ultrashort binary descriptor for fast visual matching and retrieval. *IEEE transactions on image processing*, 23(8), 3671-3683.
- [19] Materka, A., & Strzelecki, M. (1998). Texture analysis methods—a review. *Technical university of lodz, institute of electronics, COST B11 report, Brussels*, 10(1.97), 4968.
- [20] Armi, L., & Fekri-Ershad, S. (2019). Texture image analysis and texture classification methods-A review. *arXiv preprint arXiv:1904.06554*.
- [21] Leung, T., & Malik, J. (1996, April). Detecting, localizing and grouping repeated scene elements from an image. In *European Conference on Computer Vision* (pp. 546-555). Springer, Berlin, Heidelberg.
- [22] Gui, Y., & Ma, L. (2010). Periodic pattern of texture analysis and synthesis based on texels distribution. *The Visual Computer*, 26(6-8), 951-964.
- [23] Gui, Y., Chen, M., Ma, L., & Chen, Z. (2011, May). Texel based regular and near-regular texture characterization. In *2011 International Conference on Multimedia and Signal Processing* (Vol. 1, pp. 266-270). IEEE.
- [24] Liu, Y., Lin, W. C., & Hays, J. (2004). Near-regular texture analysis and manipulation. *ACM Transactions on Graphics (TOG)*, 23(3), 368-376.
- [25] An adaptive method of damage detection for fishing nets based on image processing technology
- [25] Zhao, Y. P., Niu, L. J., Du, H., & Bi, C. W. (2020). An adaptive method of damage detection for fishing nets based on image processing technology. *Aquacultural Engineering*, 102071.
- [26] Petrov, A., & Popov, A. (2020). Overview of the application of computer vision technology in fish farming. In *E3S Web of Conferences* (Vol. 175, p. 02015). EDP Sciences.
- [27] Jakobsen, R. A. H. (2011). *Automatic Inspection of Cage Integrity with Underwater Vehicle* (Master's thesis, Institutt for teknisk kybernetikk).
- [28] Bjørlo-Larsen, V., & Skeide, E. S. (2019). *Damage detection in fish farming nets using video analysis* (Bachelor's thesis, NTNU).
- [29] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 779-788).
- [30] Tao, Q., Huang, K., Qin, C., Guo, B., Lam, R., & Zhang, F. (2018, October). Omnidirectional Surface Vehicle for Fish Cage Inspection. In *OCEANS 2018 MTS/IEEE Charleston* (pp. 1-6). IEEE.
- [31] <https://justinyuzheng.files.wordpress.com/2018/12/final-report-gtmur.pdf>

- [32] Garcia, R., Nicosevici, T., & Cufi, X. (2002, October). On the way to solve lighting problems in underwater imaging. In *OCEANS'02 MTS/IEEE* (Vol. 2, pp. 1018-1024). IEEE.
- [33] [wikipedia.org/wiki/Knee\\_of\\_a\\_curve](https://en.wikipedia.org/wiki/Knee_of_a_curve)
- [34] [https://en.wikipedia.org/wiki/Knee\\_of\\_a\\_curve](https://en.wikipedia.org/wiki/Knee_of_a_curve)
- [35] [http://nana.lecturer.pens.ac.id/index\\_files/referensi/computer\\_vision/Computer%20Vision.pdf](http://nana.lecturer.pens.ac.id/index_files/referensi/computer_vision/Computer%20Vision.pdf)
- [36] Duda, R. O., & Hart, P. E. (1972). Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15(1), 11-15.
- [37] [http://www.cs.unc.edu/~lazebnik/spring10/lec09\\_hough.pdf](http://www.cs.unc.edu/~lazebnik/spring10/lec09_hough.pdf)
- [38] 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)
- [39] [https://en.wikipedia.org/wiki/Hough\\_transform#/media/File:R\\_theta\\_line.GIF](https://en.wikipedia.org/wiki/Hough_transform#/media/File:R_theta_line.GIF)
- [40] <https://www.mathworks.com/help/images/ref/imfill.html>
- [41] Deng, X., Wang, H., & Liu, X. (2019). Underwater image enhancement based on removing light source color and dehazing. *IEEE Access*, 7, 114297-114309.
- [42] Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE transactions on systems, man, and cybernetics*, 9(1), 62-66.