

Ατομική Διπλωματική Εργασία

ΨΗΦΙΟΠΟΙΗΣΗ ΕΙΚΟΝΩΝ ΜΙΚΡΟΣΚΟΠΙΟΥ

Γρηγόρης Γρηγοριάδης

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ



ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ιανουάριος 2021

ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

Ψηφιοποίηση εικόνων μικροσκοπίου

Γρηγόρης Γρηγοριάδης

Επιβλέπων Καθηγητής
Κωνσταντίνος Παττίχης

Η Ατομική Διπλωματική Εργασία υποβλήθηκε προς μερική εκπλήρωση των απαιτήσεων απόκτησης του πτυχίου Πληροφορικής του Τμήματος Πληροφορικής του Πανεπιστημίου Κύπρου

Ιανουάριος 2021

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέπον καθηγητή μου, τον κύριο Κωνσταντίνο Παττίχη, ο οποίος με καθοδήγησε με τον καλύτερο τρόπο σε αυτή την εργασία και όποτε χρειάστηκα τη βοήθειά του, αυτός μου την παρείχε απλόχερα. Επίσης θα ήθελα να ευχαριστήσω τον δόκτωρ Κυριάκο Κυριάκου, ο οποίος είχε την καλοσύνη να μου δανίσει φιλμς μικροσκοπίων που είχαν στο Ινστιτούτο, μου έδινε τις απόψεις και συμβουλές του και μου επισήμαινε διορθώσεις όποτε έκρινε εκείνος απαραίτητο.

Περίληψη

Στη διπλωματική μου εργασία, ασχολήθηκα με το θέμα της ψηφιοποίησης εικόνων μικροσκοπίου και καταμέτρησης αποστάσεων συγκεκριμένων σημείων πάνω στο κύτταρο. Το πρώτο εξάμηνο της διπλωματικής μου, ασχολήθηκα με την εύρεση μηχανήματος το οποίο θα χρησιμοποιούσα για το σκανάρισμα των εικόνων μικροσκοπίου. Το μηχάνημα θα έπρεπε να ήταν ικανό να σκανάρει τα negative films 9 mm x 7 mm σε υψηλή ποιότητα. Κάναμε μαζί με τον κύριο Κωνσταντίνο Παττίχη ο οποίος ήταν και ο υπεύθυνος καθηγητής μου σε αυτή την εργασία, διάφορες συναντήσεις με τον δόκτωρ Κυριάκο Κυριάκου του Κυπριακού Ινστιτούτου Νευρολογίας, με σκοπό την διευκρίνηση των ζητούμενων της εργασίας αυτής, αλλά και την εύρεση μηχανήματος το οποίο θα μπορούσε να παράξει τα ποθητά σκαναρίσματα. Αποφασίσαμε ότι θα έπρεπε να γράψω ένα πρόγραμμα το οποίο θα ήταν ικανό να παίρνει ένα κομμάτι από το σκαναρισμένο film, το οποίο θα επέλεγε ο χρήστης (στην περίπτωση μας ο γιατρός που θα χρησιμοποιούσε το πρόγραμμα) και το πρόγραμμα θα εφάρμοζε image segmentation, ούτως ώστε να απομονώνει ένα συγκεκριμένο κομμάτι, για το οποίο θα αναφερθούμε πιο αναλυτικά αργότερα στο report, και θα καταμετρούσε το πάχος του. Αυτό θα βοηθούσε τους γιατρούς στο να διαπιστώσουν αν ο ασθενής έχει πρόβλημα. Στη δεύτερη συνάντησή μου με τον δόκτωρ Κυριάκου, μου παρείχε διάφορα δείγματα μικροσκοπίου που θα με βοηθούσαν στην υλοποίηση του προγράμματος. Στο δεύτερο εξάμηνο της διπλωματικής μου, αφοσιώθηκα εντελώς πάνω στην υλοποίηση του προγράμματος. Είχα δοκιμάσει πάρα πολλές μεθόδους image segmentation με σκοπό να βρω ποιός αλγόριθμος θα μου πάριχε τα καλύτερα αποτελέσματα. Ακολούθως δοκίμασα εξίσου πολλούς τρόπους για καταμέτρηση του συγκεκριμένου κομματιού που θέλαμε να μετρήσουμε, για τους οποίους τρόπους θα αναφερθώ πιο αναλυτικά στη συνέχεια του report. Αργότερα, προσπάθησα να αυτοματοποιήσω την καταμέτρηση της μεμβράνης με τη χρήση του νευρωνικού δικτύου Unet το οποίο θα έκανε segment αυτόματα το σημείο στην μεμβράνη που θέλουμε να μετρήσουμε και ακολούθως αυτόματα το πρόγραμμα θα υπολόγιζε αρκετές μετρήσεις πάνω της με σκοπό να ευρεθεί ένας μέσος όρος μετρήσεων.

Περιεχόμενα

Κεφάλαιο 1	Εισαγωγή.....	7
	1.1 Επιλογή Διπλωματικής Εργασίας	7
	1.2 Συνάντηση με τον κύριο Κωνσταντίνο Παττίχη	7
	1.3 Συνάντηση με τον δόκτωρ Κυριάκο Κυριάκου	8
Κεφάλαιο 2	Σκανάρισμα films μικροσκοπίου.....	9
	2.1 Εύρεση μηχανήματος για σκανάρισμα	9
	2.2 Απόκτηση δειγμάτων για σκανάρισμα	11
	2.3 Δοκιμές Σκαναρισμάτων	12
	2.4 Τελικό Σκανάρισμα	13
	2.5 Αρχικές Μετρήσεις	13
Κεφάλαιο 3	Image Segmentation.....	19
	3.1 Εύρεση αλγόριθμου για segmentation	19
	3.2 Canny Edges segmentation	20
	3.3 K-means segmentation	21
	3.4 Chan Vese segmentation	22
	3.5 Επιλογή αλγόριθμου	23
Κεφάλαιο 4	Πρώτη έκδοση του προγράμματος.....	24
	4.1 GUI για αποκοπή κομματιού από το film	24
	4.2 Εφαρμογή Chan Vese	25
	4.3 Αρχικές δοκιμές μετρήσεων	26
	4.4 Εύρεση σκελετού εικόνας	27
	4.5 Εμφάνιση γραμμών για μετρήσεις	29
Κεφάλαιο 5	Τελική έκδοση του προγράμματος.....	30
	5.1 Χειροκίνητη επιλογή σημείου για μετρήσεις	30
	5.2 Εύρεση σκελετού και σημείων	32
	5.3 Κάθετες Γραμμές πάνω στο σκελετό	33

5.4 Μετατροπή μετρήσεων σε πραγματική κλίμακα	34
5.5 Αποτελέσματα	36
5.6 Εκτίμηση σφάλματος με υπάρχοντες μετρήσεις	64
Κεφάλαιο 6 Segmentation με τη χρήση Unet.....	72
6.1 Τι είναι το Unet και πώς λειτουργεί	72
6.2 Δημιουργία δικού μας μοντέλου	74
6.3 Αποτελέσματα μετά το training του δικτύου	76
6.4 Ενσωμάτωση του μοντέλου στο σύστημα μας	77
Κεφάλαιο 7 Συμπεράσματα.....	79
7.1 Αλγόριθμοι για Image Segmentation	79
7.2 Unet και machine learning	80
7.3 Τελική αξιολόγηση του προγράμματός μας	80
7.4 Future work	81
Π α ρ ά ρ τ η μ α 1.....	82
Π α ρ ά ρ τ η μ α 2.....	86
Β ι β λ ι ο γ ρ α φ ί α	93

Κεφάλαιο 1

Εισαγωγή

1.1 Επιλογή Διπλωματικής Εργασίας	7
1.2 Συνάντηση με τον κύριο Κωνσταντίνο Παττίχη	7
1.3 Συνάντηση με τον δόκτωρ Κυριάκο Κυριάκου	8

1.1 Επιλογή Διπλωματικής Εργασίας

Στο τελευταίο έτος της φοίτησής μας στο Πανεπιστήμιο Κύπρου, καλεστήκαμε να επιλέξουμε θέματα για διπλωματική εργασία, η οποία είναι η εργασία που θα συνοψίζει όλη μας την πορεία στο πανεπιστήμιο. Πήρα τη λίστα με τις επιλογές των θεμάτων που είχαμε να επιλέξουμε και το θέμα που μου κίνησε περισσότερο το ενδιαφέρον ήταν η ψηφιοποίηση εικόνων μικροσκοπίου.

Ο λόγος που επέλεξα αυτό το θέμα για την εργασία μου είναι γιατί είχα ασχοληθεί με ψηφιοποίηση στο παρελθόν, όταν βοηθούσα τον πατέρα μου στη δουλειά του να μετατρέψει analog κασέτες σε digital, έτσι είχα μια γενική γνώση περί ψηφιοποίησης. Επίσης, όταν είδα ότι ο καθηγητής ο οποίος θα με καθοδηγούσε σε αυτή την εργασία ήταν ο κύριος Παττίχης είμουνα σίγουρος ότι θα είχα επαρκή βοήθεια και σωστή καθοδήγηση γνωρίζοντας τον από μαθήματα που μου δίδαξε στο παρελθόν.

1.2 Συνάντηση με τον κύριο Κωνσταντίνο Παττίχη

Μετά την επιλογή του θέματος της διπλωματικής μου εργασίας, έκανα μια συνάντηση με τον κύριο Παττίχη για να μάθω περαιτέρω λεπτομέρειες περί του θέματος, καθώς και το πως θα ξεκινούσαμε. Μου εξήγησε ότι θα συνεργαζόμασταν με το Ινστιτούτο Νευρολογίας και Γενετικής Κύπρου και πιο συγκεκριμένα με τον δόκτωρ Κυριάκο Κυριάκου ο οποίος είναι ο

είναι ο Ιδρυτής και Επικεφαλής του Department of Electron Microscopy/Molecular Pathology, και Καθηγητής, και Πρύτανης του Cyprus School of Molecular Medicine.

Επίσης ο κύριος Παττίχης μου υπέδειξε την σημαντικότητα της εύρεσης μηχανήματος το οποίο θα μας βοηθούσε να σκανάρουμε τα films του μικροσκοπίου σε υψηλή ευκρίνεια, πράγμα που θα ήταν θέμα συζήτησης και σε συνάντηση με τον δόκτωρ Κυριάκου την οποία και είχαμε προγραμματίσει.

1.3 Συνάντηση με τον δόκτωρ Κυριάκο Κυριάκου

Μετά την συνάντησή μου με τον κύριο Παττίχη κανονίσαμε συνάντηση στο Ινστιτούτο Νευρολογίας και Γενετικής Κύπρου μαζί με τον δόκτωρ Κυριάκου για να καταλήξουμε στα ζητούμενα της εργασίας μας, καθώς και σε ένα μηχανήμα το οποίο θα μπορούσε να σκανάρει τα films με τις συγκεκριμένες διαστάσεις τους (90 mm x 65 mm) και σε υψηλή ευκρίνεια. Μετά το ψάξιμο που κάναε online καταλήξαμε σε διάφορα μηχανήματα τα οποία κλίθηκα να μελετήσω και στο σπίτι για να δω αν πληρούν τις προδιαγραφές μας και είναι στα όρια του budget μας.

Όσο αφορά το θέμα της διπλωματικής, ο δόκτωρ Κυριάκου μου εξήγησε πως μετά το σκανάρισμα, οι γιατροί εξετάζουν το πάχος της μεμβράνης γύρω από το κύτταρο του μυϊκού ιστού με σκοπό να εξακριβώσουν αν ο ασθενής έχει πρόβλημα ή όχι. Η δουλεία μου ήταν να γράψω ένα πρόγραμμα το οποίο θα έκανε την διαδικασία αυτή πιο εύκολη. Πιο κάτω φαίνεται μια εικόνα που δείχνει ακριβώς ποιά σημεία καλούμασταν να μετρήσουμε πάνω στη μεμβράνη.

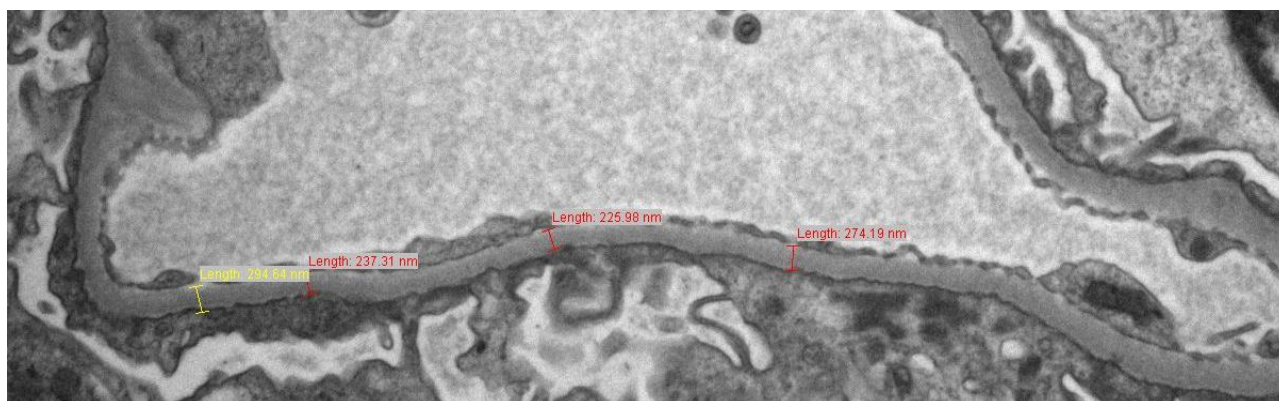


Figure 1.1 Παράδειγμα Μετρήσεων

Κεφάλαιο 2

Σκανάρισμα films μικροσκοπίου

2.1 Εύρεση μηχανήματος για σκανάρισμα	9
2.2 Απόκτηση δειγμάτων για σκανάρισμα	10
2.3 Δοκιμές Σκαναρισμάτων	12
2.4 Τελικό σκανάρισμα	13
2.5 Μετρήσεις	13

2.1 Εύρεση μηχανήματος για σκανάρισμα

Μετά τη συνάντησή μου με τον δόκτωρ Κυριάκου, καταλήξαμε σε διάφορα μηχανήματα που ήταν υποψήφια για παραγγελία. Όταν πήγα σπίτι τα μελέτησα ακόμη περισσότερο και διαπίστωσα πως δεν μπορούσαν όλα να σκανάρουν τα φίλμς των διαστάσεών μας, ζήτημα το οποίο ήταν και το πιο σημαντικό. Για καλή μου τύχη μίλησα στον πατέρα μου, ο οποίος ασχολείται με παρόμοια θέματα, και μου είπε πως είχε στην κατοχή του ένα film scanner το οποίο πληρούσε όλες τις προδιαγραφές που χρειαζόμασταν. Το film scanner αυτό λεγόταν Epson Perfection V550 Photo και οι προδιαγραφές του φαίνονται πιο κάτω.



Figure 2.1 Epson Perfection V550 Photo



Figure 2.2 Εσωτερικό του Scanner

Tech Specs

Product features and specifications are subject to change without prior notice.

SKU: B11B210303

EXPAND ALL

Technology

Scanner Type	Flatbed Scanner
Optical Resolution	6,400 DPI (Horizontal x Vertical)
Optical Resolution	Main 6,400 DPI x Sub 9,600 DPI
Optical Density	3.4 Dmax
Maximum document size	216 mm x 297 mm (Horizontal x Vertical)
Paper Formats	A4 (21.0x29.7 cm)
Scanning color depth	Input: 48 Bits Color , Output: 48 Bits Color
Category	High Resolution

Scanner

Optical Sensor	Matrix CCD
Light Source	White LED, IR LED with ReadyScan LED Technology
Scanning Method	Fixed documents and moving carriage
Output Resolution	50~6400 (1 dpi step), 9600, 12800 DPI

Figure 2.3 Epson Perfection V550 Photo Specifications

2.2 Απόκτηση δειγμάτων για σκανάρισμα

Αφού είχα στην κατοχή μου το μηχάνημα, έκανα μια συνάντηση με τον δόκτωρ Κυριάκου για να μου δώσει διάφορα φίλμς μικροσκοπίου τα οποία θα χρησιμοποιούσα για να κάνω τις δοκιμές μου. Ο δόκτωρ, μου παρείχε ένα φάκελο ο οποίος περιείχε όχι μόνο τα δείγματα που θα σκάναρα, αλλά ως επίσης διάφορα έγγραφα όσον αφορά τη μεμβράνη των κυττάρων που θα μετρήσουμε, καθώς και τα εκτυπωμένα φίλμς στα οποία μου είχε σημαδέψει τα σημεία πάνω στην μεμβράνη που θα μετρούσα όπως φαίνεται πιο κάτω.

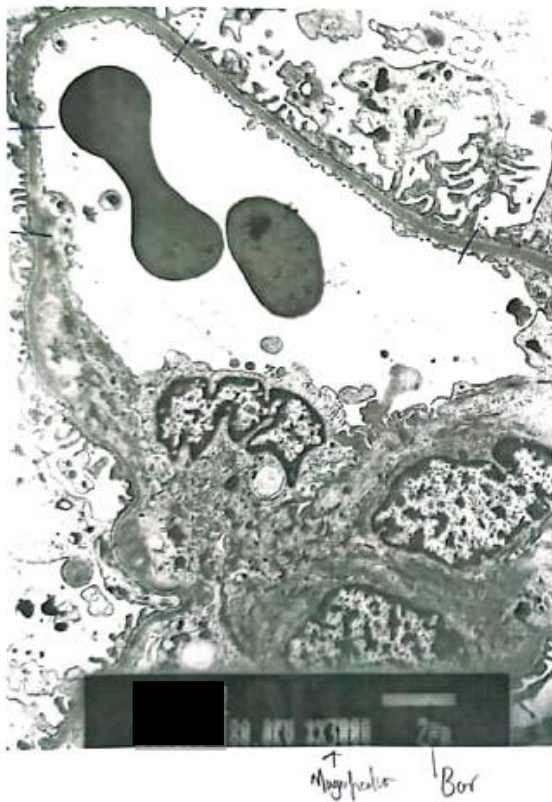


Figure 2.4 Σκαναρισμένο film μικροσκοπίου

Η πιο πάνω εικόνα απεικονίζει ένα από τα δείγματα. Πάνω φαίνονται με στυλό οι περιοχές στη μεμβράνη που καλόμουν να μετρήσω, στο κάτω μέρος υπάρχει το magnification (x3000) και το scale σε micrometers (2μm).

Με αυτό τον τρόπο είχαν αναπαρασταθεί και τα υπόλοιπα από τα 12 δείγματα που μου έδωσε ο δόκτωρ.

2.3 Δοκιμές Σκαναρισμάτων

Με το που πήρα τον φάκελο που μου έδωσε ο δόκτωρ Κυριάκου ξεκίνησα αμέσως να κάνω διάφορα πειράματα σκαναρισμάτων. Πήρα ένα φίλμ και το σκάναρα 6 φορές, με διαφορετικές διαστάσεις, διαφορετικό resolution και κάποιες φορές με grayscale. Ο πιο κάτω πίνακας περιέχει όλα τα σκαναρίσματα που κάναμε για τα films.

Test Number	Dimensions	Resolution	Grayscale	Size
Test 1	2570 x 3535 pixels	1200 x 1200 ppi	Yes	5.80 MB
Test 2	2534 x 3550 pixels	1200 x 1200 ppi	No	9.80 MB
Test 3	11415 x 15800 pixels	2000 x 2000 ppi	No	125 MB
Test 4	11415 x 15800 pixels	2000 x 2000 ppi	Yes	65.9 MB
Test 5	14166 x 19021 pixels (A4)	2000 x 2000 ppi	No	177 MB
Test 6	14166 x 19021 pixels (A4)	2000 x 2000 ppi	Yes	96.6 MB

Table 2.1 Δοκιμές σκαναρίσματος για ένα δείγμα

Αρχικά επιλέχθηκε από τον δόκτωρ Κυριάκου το Test 6 το οποίο είχε μέγεθος A4, όμως το μέγεθος του αρχείου είναι αχρείαστα μεγάλο και η διαδικασία σκαναρίσματός του πολύ χρονοβόρα. Έτσι επιλέχθηκε το Test 4 το οποίο ήταν η δεύτερη επιλογή του δόκτωρ Κυριάκου.

Στη συνέχεια έκανα μια συνάντηση με τον κύριο Παττίχη και μου είπε πως θα ήταν καλά να δοκιμάσουμε τα όρια του μηχανήματος όσο αφορά το resolution και έτσι δοκίμασα και σκάναρα το φίλμ για μεγαλύτερα resolutions που φαίνονται στον πίνακα πιο κάτω.

Test Number	Dimensions	Resolution	Grayscale	Size
Test 7	11415 x 17865 pixels	3200 x 3200 ppi	Yes	194 MB
Test 8	14136 x 22222 pixels	6400 x 6400 ppi	Yes	299 MB
Test 9	14120 x 22081 pixels	8000 x 8000 ppi	Yes	297 MB

Table 2.2 Δοκιμές σκαναρίσματος για ένα δείγμα

Με βάση τον πιο πάνω πίνακα αποφασίσαμε να συνεχίσουμε με τις διαστάσεις και το resolution του Test 9.

2.4 Τελικό Σκανάρισμα

Πιο κάτω φαίνεται ο τελικός πίνακας με όλα τα σκαναρισμένα φίλμς για τα dimensions και resolution που προαναφέρθηκαν.

Name	Dimensions	Resolution	Grayscale	Size
Sample 1	13190 x 22181	8000 x 8000 ppi	Yes	294 MB
Sample 2	13939 x 22199	8000 x 8000 ppi	Yes	295 MB
Sample 3	13939 x 22199	8000 x 8000 ppi	Yes	295 MB
Sample 4	13911 x 22036	8000 x 8000 ppi	Yes	292 MB
Sample 5	14248 x 22021	8000 x 8000 ppi	Yes	299 MB
Sample 6	13911 x 21176	8000 x 8000 ppi	Yes	280 MB
Sample 7	14181 x 21621	8000 x 8000 ppi	Yes	292 MB
Sample 8	14181 x 21765	8000 x 8000 ppi	Yes	294 MB
Sample 9	14120 x 22081	8000 x 8000 ppi	Yes	297 MB
Sample 10	14119 x 21696	8000 x 8000 ppi	Yes	292 MB
Sample 11	14261 x 22105	8000 x 8000 ppi	Yes	300 MB
Sample 12	14119 x 21489	8000 x 8000 ppi	Yes	289 MB

Table 2.3 Τελικά σκαναρίσματα για όλα τα δείγματα

2.5 Αρχικές Μετρήσεις

Μετά το σκανάρισμα των φίλμς που μου παρείχε ο δόκτωρ Κυριάκου, μετά από επεξεργασία των εικονώ στο Photoshop σημάδεψα τα σημεία που έπρεπε να μετρήσω με μια κίτρινη γραμμή. Επίσης πήρα την προτοβουλία να υπολογίσα και σημειώσα την απόσταση κάθε κίτρινης γραμμής, μετατρέποντας τα pixels σε mm. Πιο κάτω ακολουθεί κάθε εικόνα με τις μετρήσεις της.

Sample 1:

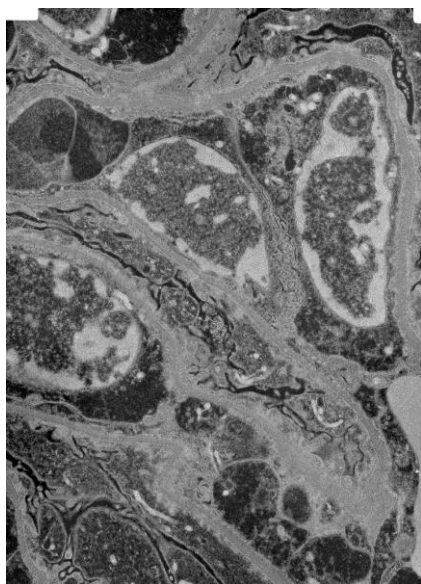


Figure 2.5.1 Σκανάρισμα πρώτου δείγματος

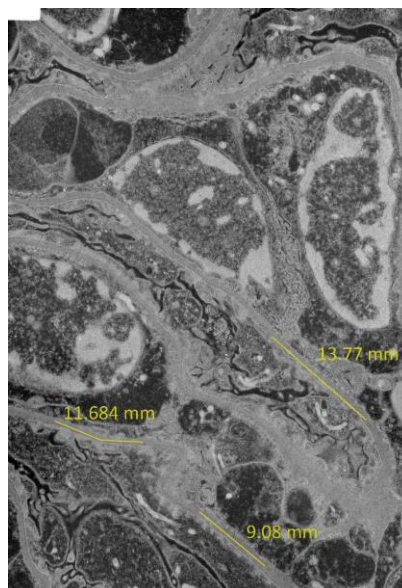


Figure 2.5.2 Περιοχές για μετρήσεις

Sample 2:

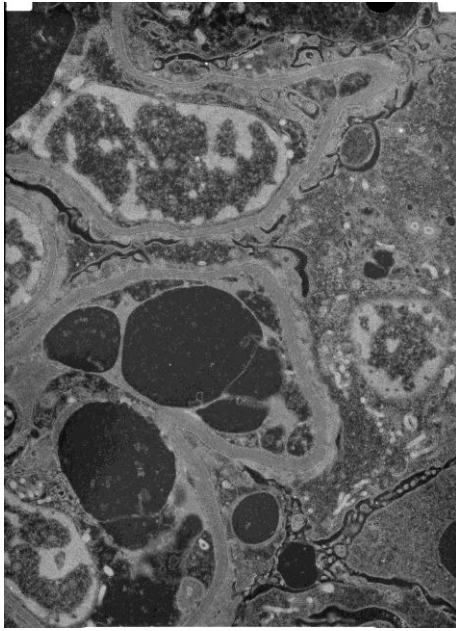


Figure 2.6.1 Σκανάρισμα δεύτερον δείγματος

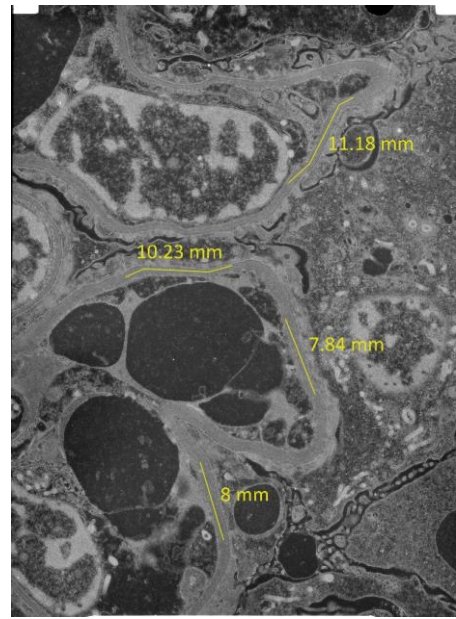


Figure 2.6.2 Περιοχές για μετρήσεις

Sample 3:

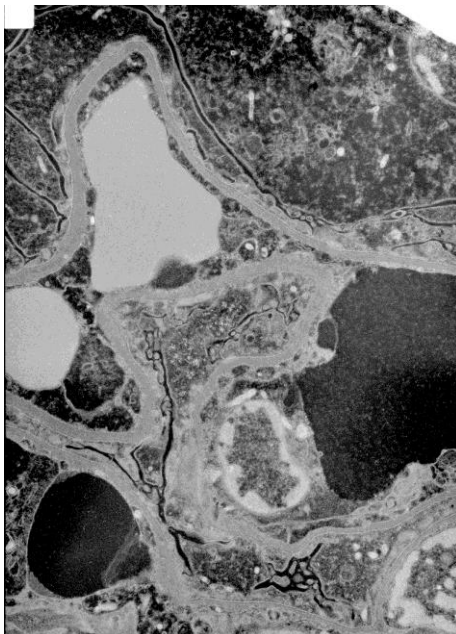


Figure 2.7.1 Σκανάρισμα τρίτον δείγματος

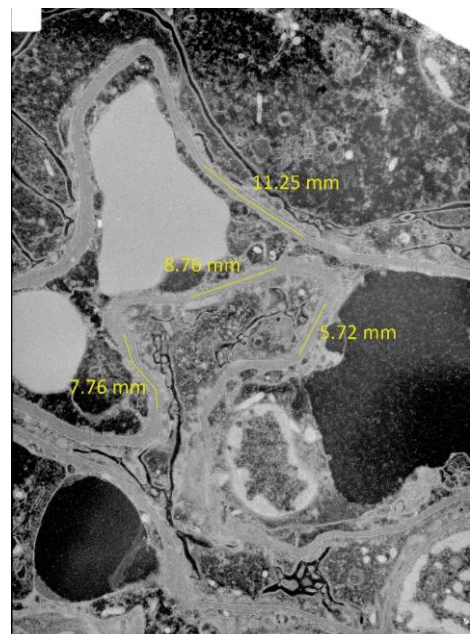


Figure 2.7.2 Περιοχές για μετρήσεις

Sample 4:

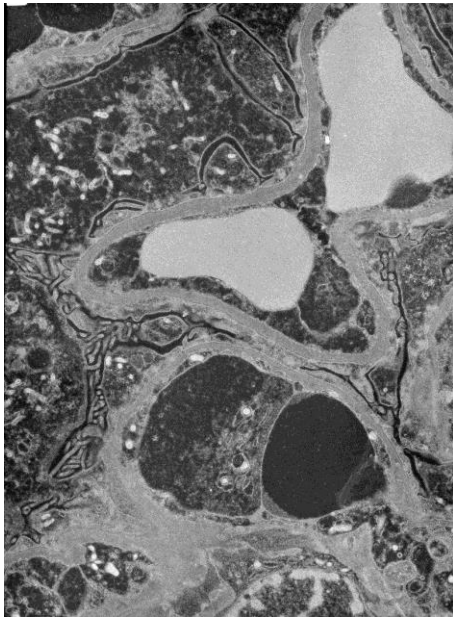


Figure 2.8.1 Σκανάρισμα τέταρτου δείγματος

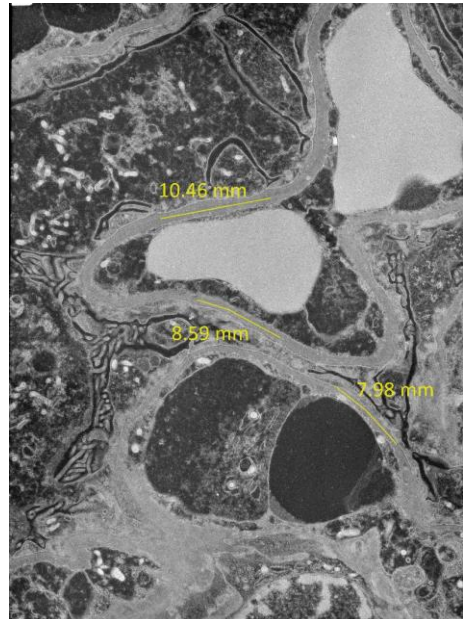


Figure 2.8.2 Περιοχές για μετρήσεις

Sample 5:

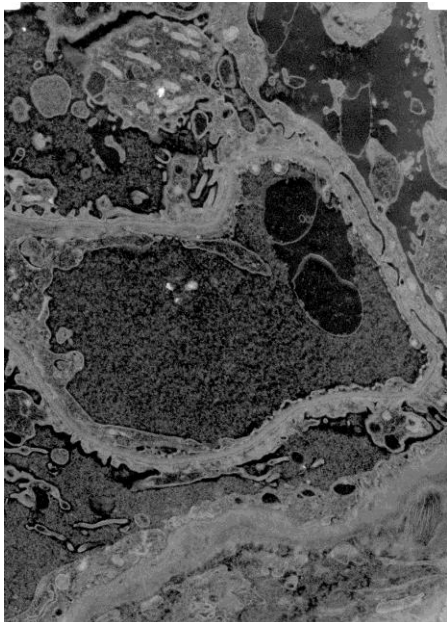


Figure 2.9.1 Σκανάρισμα πέμπτου δείγματος

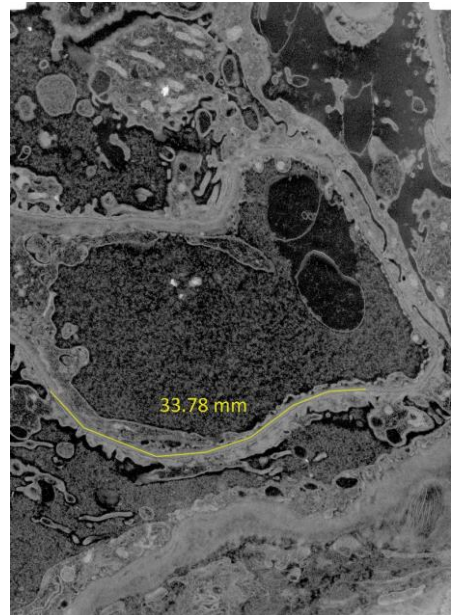


Figure 2.9.2 Περιοχές για μετρήσεις

Sample 6:

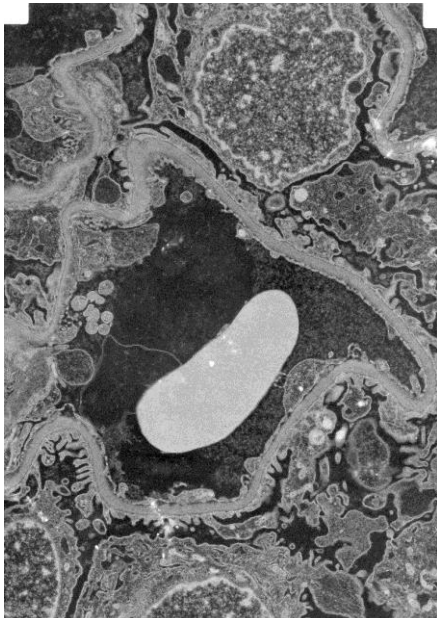


Figure 2.10.1 Σκανάρισμα έκτου δείγματος

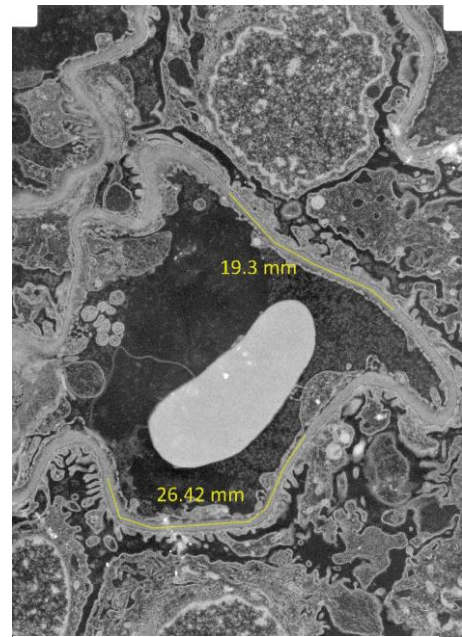


Figure 2.10.2 Περιοχές για μετρήσεις

Sample 7:

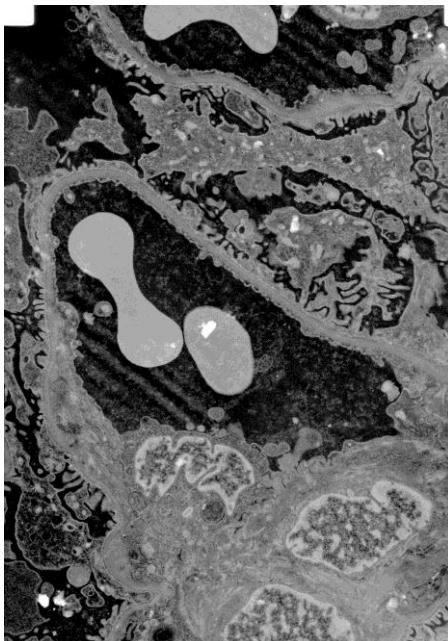


Figure 2.11.1 Σκανάρισμα έβδομου δείγματος

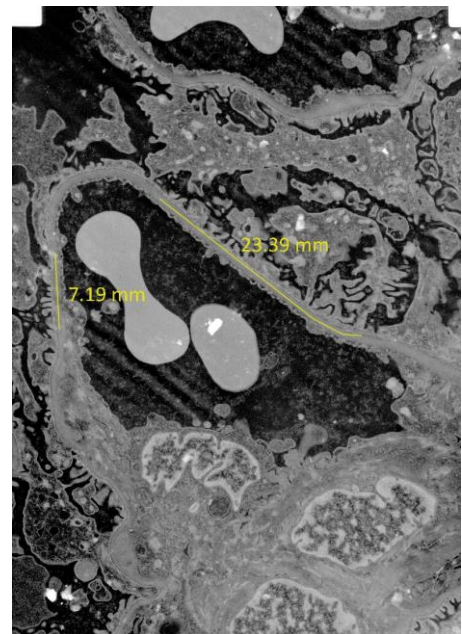


Figure 2.11.2 Περιοχές για μετρήσεις

Sample 8:

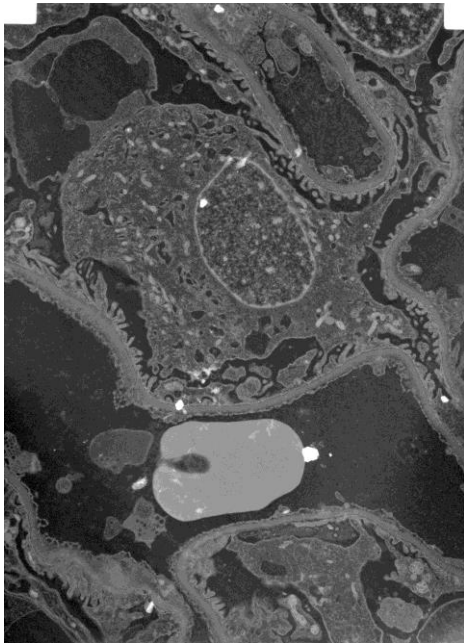


Figure 2.12.1 Σκανάρισμα όγδοου δείγματος

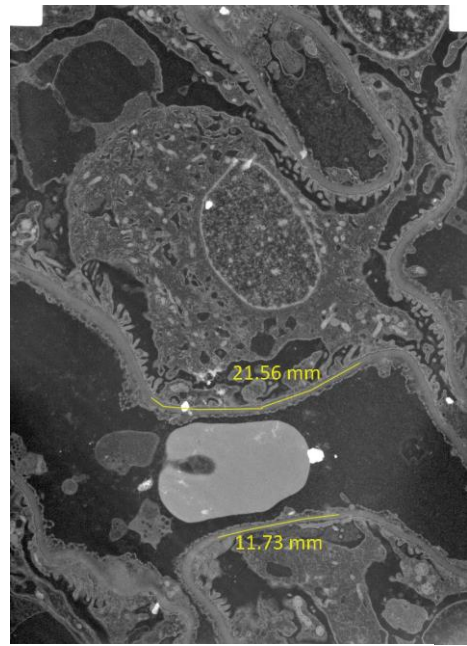


Figure 2.12.2 Περιοχές για μετρήσεις

Sample 9:

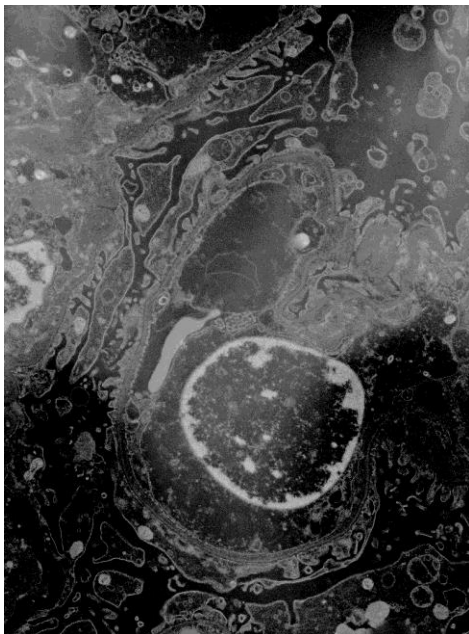


Figure 2.13.1 Σκανάρισμα ένατου δείγματος

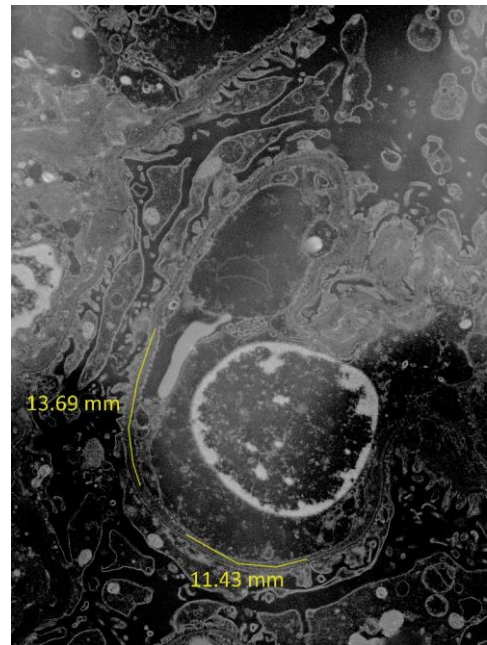


Figure 2.13.2 Περιοχές για μετρήσεις

Sample 10:

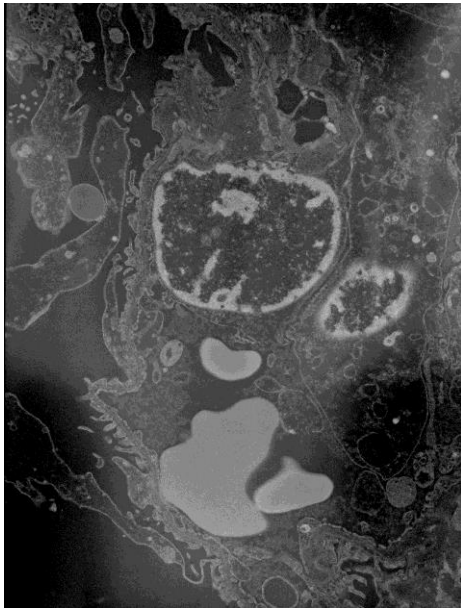


Figure 2.14.1 Σκανάρισμα δέκατον δείγματος

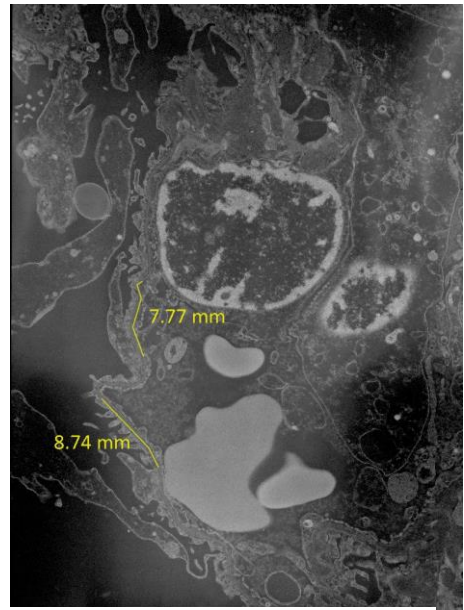


Figure 2.14.2 Περιοχές για μετρήσεις

Sample 11:

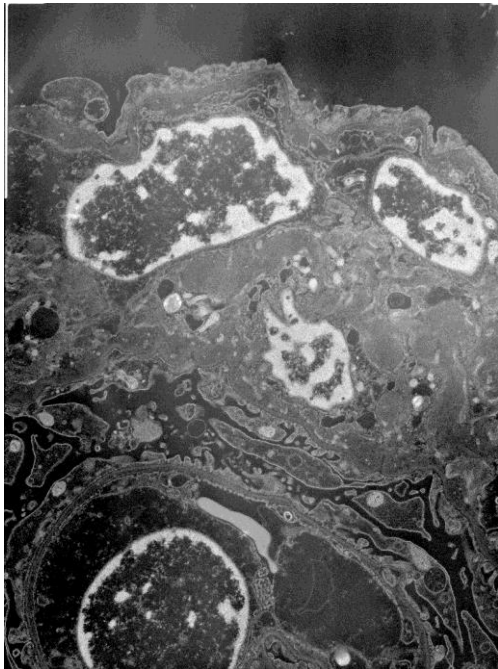


Figure 2.15.1 Σκανάρισμα ενδέκατον δείγματος

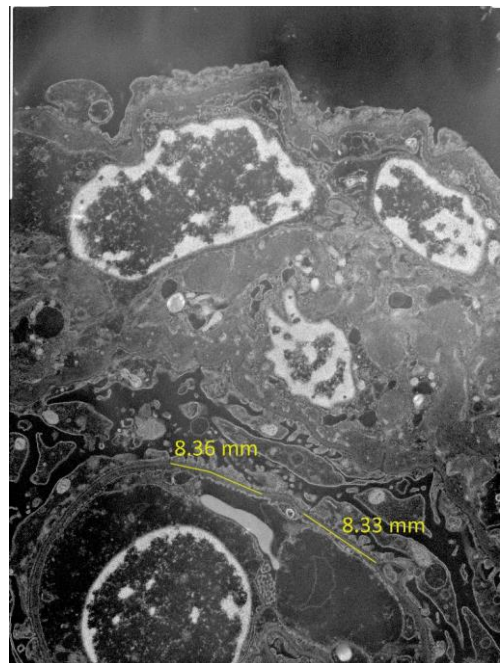


Figure 2.15.2 Περιοχές για μετρήσεις

Κεφάλαιο 3

Image Segmentation

3.1 Εύρεση αλγόριθμου για segmentation	19
3.2 Canny Edges segmentation	20
3.3 Kmeans segmentation	21
3.4 Chan Vese segmentation	22
3.5 Επιλογή αλγορίθμου	23

3.1 Εύρεση αλγόριθμου για segmentation

Μετά από τα σκαναρίσματα των films, αποφάσισα να ξεκινήσω να γράφω το πρόγραμμα το οποίο θα έκανε τις μετρήσεις που θέλαμε για το πάχος της μεμβράνης με ακρίβεια. Το πρόγραμμα αυτό γράφτηκε στη γλώσσα Python γιατί η Python μας δίνει πρόσβαση σε πάρα πολλές βιβλιοθήκες που θα με βοηθούσαν στην υλοποίηση διαφόρων λειτουργιών που είχα υπόψη.

Το πρώτο βήμα για την υλοποίηση του προγράμματος, όπως μου εξήγησε ο κύριος Παττίχης ήταν να βρούμε διάφορους αλγόριθμους οι οποίοι θα μας βοηθούσαν να κάνουμε segment την εικόνα μας στο σημείο όπου θα γίνονται οι μετρήσεις μας. Ο καθηγητής μου μου πρότινε να προστρέξω στους αλγόριθμους canny, snakes supervised και snakes unsupervised. Αρχικά δοκίμασα τον αλγόριθμο snakes supervised με τον οποίο επιλέεις το κομμάτι που θέλεις να κάνεις segment με τη χρήση coordinates, χωρίς καμία επιτυχία αφού το film του μικροσκοπίου είναι πολύ περίπλοκο για να δουλέψει ο αλγόριθμος.

3.2 Canny Edges Segmentation

Μετά από τις ανεπιτυχείς προσπάθειές μου να κάνω το supervised segmentation να δουλέψει, έστρεψα την προσοχή μου στον αλγόριθμο canny edges. Το Canny edges detection είναι μια τεχνική για την εξαγωγή χρήσιμων δομικών πληροφοριών από διαφορετικά αντικείμενα μέσα σε εικόνες και τη δραματική μείωση της ποσότητας των δεδομένων που πρόκειται να υποβληθούν σε επεξεργασία.

Ο αλγόριθμος αυτός ακολουθεί 5 διαφορετικά βήματα:

1. Εφαρμόζεται πρώτα το φίλτρο Gaussian για να εξομαλυνθεί η εικόνα και να αφαιρεθεί ο θόρυβος
2. Μετά βρίσκει τα intensity gradients της εικόνας
3. Εφαρμόζει non-maximum suppression για να μην πάρει ψευδείς αποκρίσεις στην ανίχνευση άκρων
4. Εφαρμόζει double thresholding για να προσδιορίσει πιθανές ακμές
5. Τέλος, κάνει suppress όλες τις ακμές που δεν ενώνονται με κύριες ακμές.

Το Canny edges detection μπορούσε να καθορίσει ακμές πολύ γρήγορα και σε ολόκληρο το film, χωρίς να δυσκολεύεται. Έτσι το εφάρμοσα για ένα συγκεκριμένο κομμάτι στο film που φαίνεται πιο κάτω, το οποίο ήταν σε αισθητά μειωμένες διαστάσεις και resolution με το κανονικό για να μπορώ να παίρνω αποτελέσματα πιο γρήγορα.

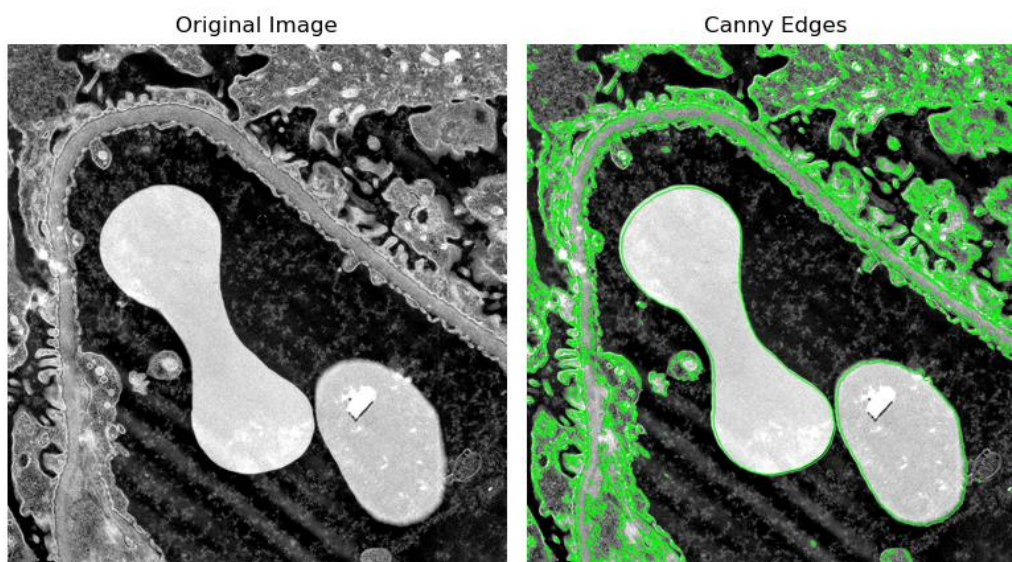


Figure 3.1 Canny edges segmentation

Το ίδιο κομμάτι του film χρησιμοποιήθηκε και για τους υπόλοιπους αλγόριθμους που χρησιμοποιήθηκαν μετά.

3.3 K-means Segmentation

Στη συνέχεια μελέτησα διάφορους αλγόριθμους για unsupervised segmentation και βρήκα τον αλγόριθμο K-means. Ο αλγόριθμος αυτός, ονομάζεται αλλιώς και k-means clustering επειδή χωρίζει τα χαρακτηριστικά της εικόνας σε περιοχές. Βασικά παίρνει κομμάτια της εικόνας που έχουν παρόμοιο χρώμα για παράδειγμα και τα οργανώνει σε clusters. Το κάθε cluster έχει το δικό του χρώμα. Το k-means clustering είναι εύκολο στην εφαρμογή ακόμη και σε μεγάλα σύνολα δεδομένων όπως τα films μας. Έχει χρησιμοποιηθεί με επιτυχία στο market segmentation, στο computer vision, στην αστρονομία και σε πολλούς άλλους τομείς. Συχνά χρησιμοποιείται ως βήμα προεπεξεργασίας για άλλους αλγόριθμους, για παράδειγμα για την εύρεση μιας αρχικής διαμόρφωσης.

Εφάρμοσα αυτό τον αλγοριθμο για το ίδιο κομμάτι του film που χρησιμοποίησα και στο canny edges detection και το αποτέλεσμα φαίνεται πιο κάτω.

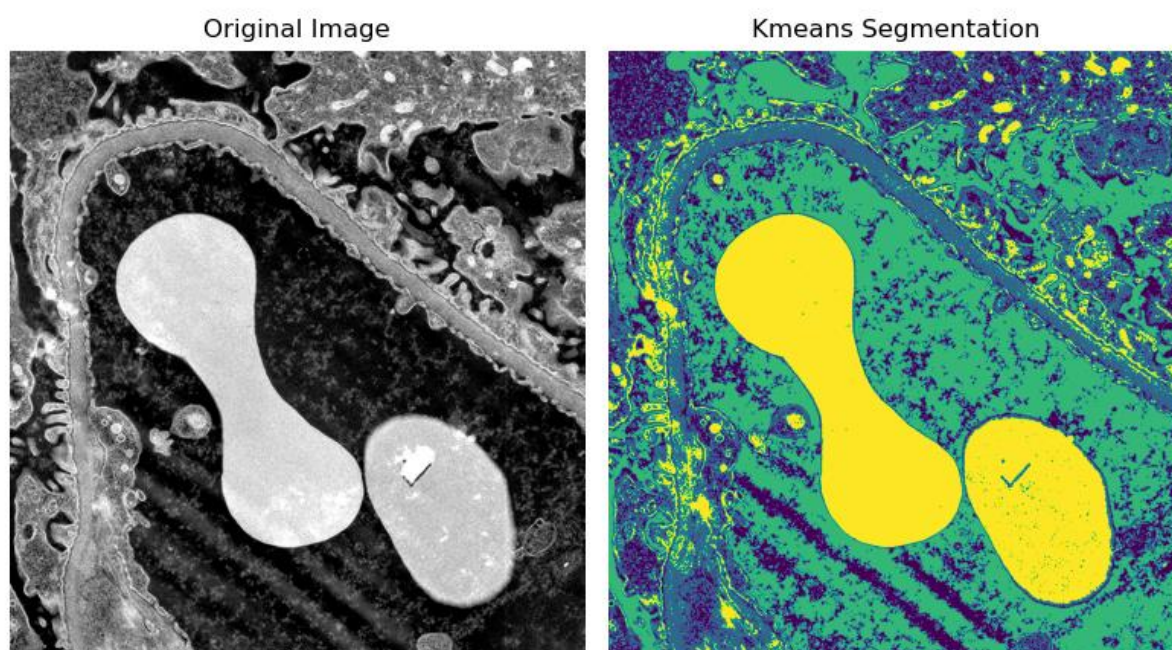


Figure 3.2 K-means segmentation

Όπως παρατηρούμε δεν είναι όσο αποτελεσματικός όσο είναι ο Canny edges detection για τη δική μας περίπτωση.

3.4 Chan Vese Segmentation

Ένας άλλος αλγόριθμος για unsupervised segmentation που μελέτησα, ήταν ο αλγόριθμος Chan Vese. Το μοντέλο Chan-Vese για active contours είναι μια ισχυρή και ευέλικτη μέθοδος που μπορεί να τμηματοποιήσει πολλών ειδών εικόνων, συμπεριλαμβανομένων ορισμένων που θα ήταν αρκετά δύσκολο να διαχωριστούν με τη μέθοδο «κλασικής» τμηματοποίησης, δηλαδή χρησιμοποιώντας μεθόδους κατωφλίου ή βαθμίδωσης, όπως είναι και η περίπτωση μας. Αυτό το μοντέλο βασίζεται στο Mumford-Shah για τμηματοποίηση και χρησιμοποιείται ευρέως στον τομέα της ιατρικής απεικόνισης, ειδικά για την κατάτμηση του εγκεφάλου, της καρδιάς και της τραχείας. Το μοντέλο βασίζεται σε ένα πρόβλημα ελαχιστοποίησης της ενέργειας, το οποίο μπορεί να αναδιατυπωθεί στη διαμόρφωση καθορισμένων επιπέδων, οδηγώντας σε έναν ευκολότερο τρόπο επίλυσης του προβλήματος.

Χρησιμοποιήθηκε το ίδιο κομμάτι του film που χρησιμοποιήθηκε και για τις προηγούμενες μεθόδους segmentation και το αποτέλεσμα παρουσιάζεται πιο κάτω.

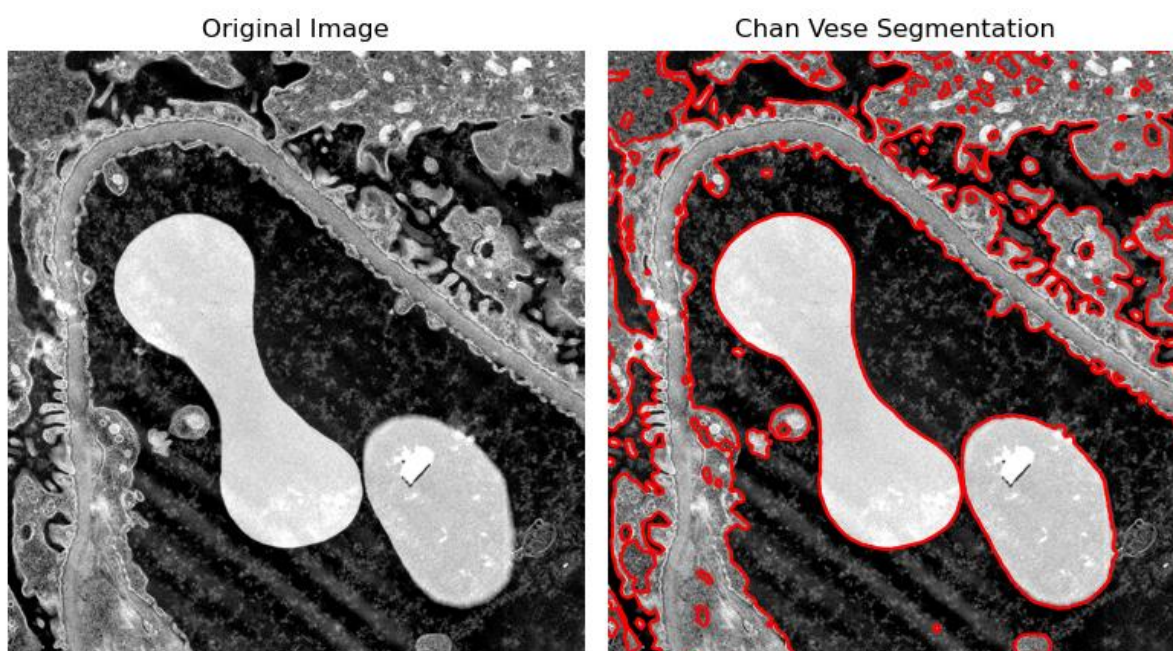


Figure 3.3 Chan Vese segmentation

Όπως βλέπουμε ο αλγόριθμος Chan Vese είναι ο πιο αποτελεσματικός από τους άλλους 2 για την περίπτωση μας. Είναι πιο ακριβής από τον Canny edges και δεν παίρνει τόσες άχρηστες πληροφορίες όσες εκείνος.

3.5 Επιλογή αλγορίθμου

Σε μετέπειτα στάδιο έγραψα ένα πρόγραμμα το οποίο εκτελούσε και τους 3 αλγόριθμους και παρουσίαζε όλα τα αποτελέσματα σε ένα plot για να τα δείξω του κύριου Παττίχη, ο οποίος συμφώνησε ότι ο Chan Vese αλγόριθμος ήταν ο καλύτερος από τους 3 και μου πρότεινε να συνεχίσω με βάση εκείνο. Το plot του προγράμματος ήταν το ακόλουθο.

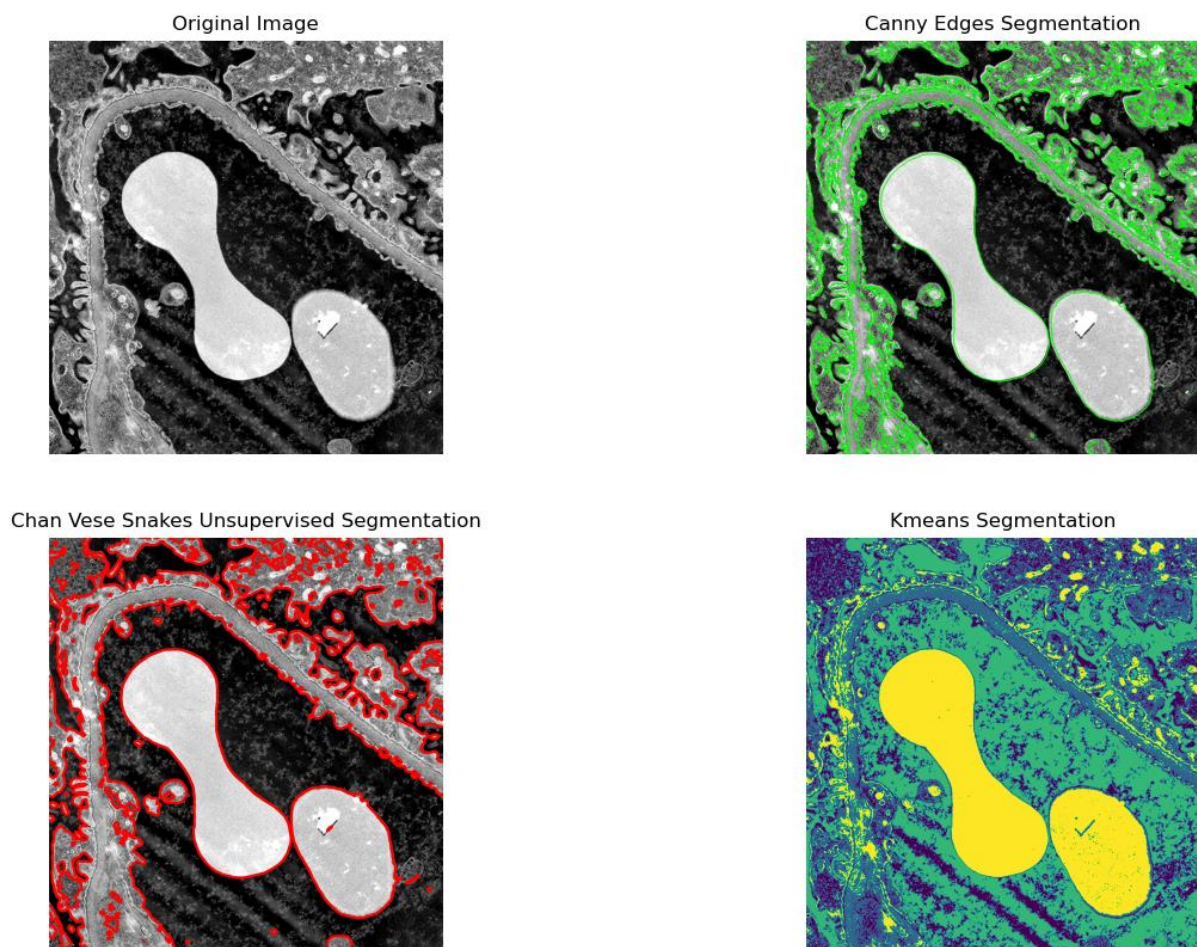


Figure 3.4 Plot με όλους τους αλγόριθμους που μελετήθηκαν για image segmentation

Έτσι επιλέχθηκε ο αλγόριθμος Chan Vese για τη συνέχεια.

Κεφάλαιο 4

Πρώτη έκδοση του προγράμματος

4.1 GUI για αποκοπή κομματιού από το film	24
4.2 Εφαρμογή Chan Vese	25
4.3 Αρχικές δοκιμές μετρήσεων	26
4.4 Εύρεση σκελετού εικόνας	28
4.5 Εμφάνιση γραμμών για μετρήσεις	29

4.1 GUI για αποκοπή κομματιού από το film

Μετά την επιλογή του chan vese αλγόριθμου, σκέφτηκα να δημιουργήσω ένα GUI το οποίο θα επέτρεπε στον χρήστη να επιλέξει το σημείο στο film που αυτός θα ήθελε να μετρήσει, έτσι με τη βοήθεια της βιβλιοθήκης matplotlib αναπαρηστούσα ολόκληρο το film στο plot και ο χρήστης με τη χρήση του mouse μπορούσε να επιλέξει την περιοχή που θέλει να μετρήσει με click - drag - release όπως φαίνεται πιο κάτω.

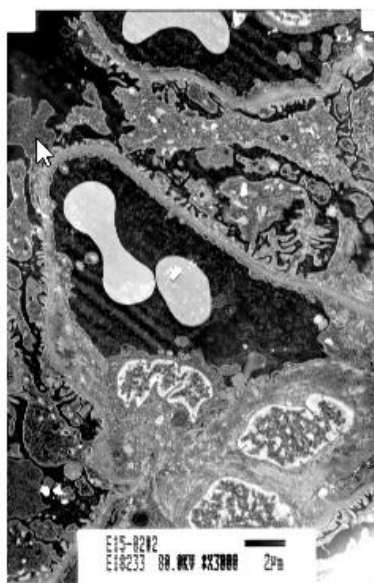


Figure 4.1 Παρουσίαση του film από το πρόγραμμα

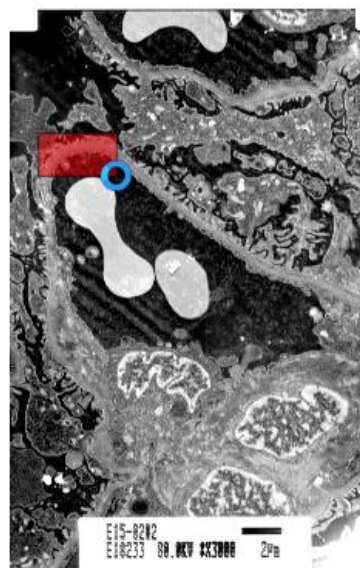


Figure 4.2 Επιλογή κομματιού από το χρήστη

Ο τρόπος που γίνεται η αποκοπή κομματιού είναι με τη χρήση της εντολής RectangleSelector από τα widgets της βιβλιοθήκης matplotlib.

4.2 Εφαρμογή Chan Vese

Αφού επιλέξαμε το κομμάτι στο οποίο θέλουμε να πάρουμε μετρήσεις, αποκόπτεται από το υπόλοιπο film και εκτελείται πάνω του ο αλγόριθμος chan vese. Με το που εκτελείται ο chan vese, παρουσιάζεται σε ένα plot το αποκομμένο κομμάτι με το contour area του segmentation και δίπλα η μάσκα του την οποία θα χρησιμοποιήσουμε αργότερα για να αφαιρέσουμε τα σημεία έξω από το contour area του Chan vese. Το plot είναι το ακόλουθο.

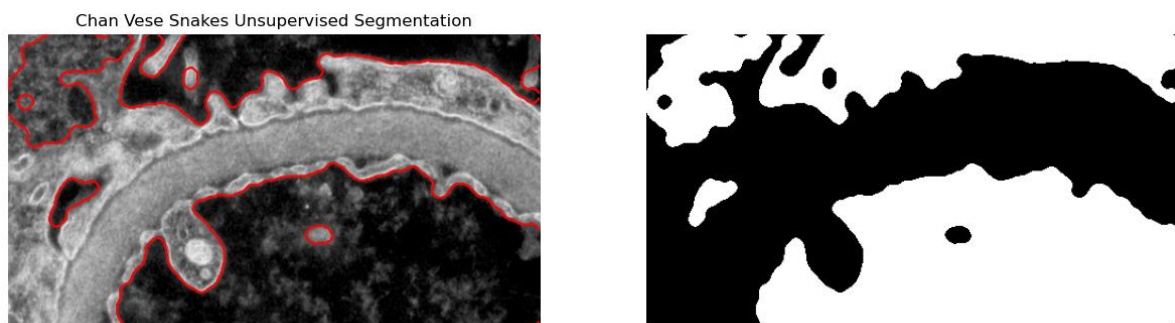


Figure 4.3 Εφαρμογή του Chan Vese στο επιλεγμένο κομμάτι

Ακολούθως, για να μπορέσει το πρόγραμμα να κατανοήσει πιο κομμάτι θα αφαιρεθεί μετά το segmentation, εκτελείται ένας έλεγχος ο οποίος συγκρίνει τα άσπρα και τα μαύρα pixels στην εικόνα της μάσκας. Αν τα μαύρα είναι περισσότερα από τα άσπρα τότε αφαιρούνται από την αρχική εικόνα τα pixels τα οποία έχουν τα ίδια coordinates με τα άσπρα στην μάσκα και το αντίθετο. Όταν γίνει αυτό τότε παρουσιάζεται το τελικό αποκομμένο κομμάτι στην οθόνη όπως φαίνεται πιο κάτω.



Figure 4.4 Αφαίρεση περιοχής εκτός του Chan Vese

Ο λόγος που γίνεται αυτή η διαδικασία είναι για να μπορέσουμε σε μετέπειτα στάδιο να πάρουμε το σκελετό της εικόνας που θα μας βοηθήσει να πάρουμε τις μετρήσεις μας.

4.3 Αρχικές δοκιμές μετρήσεων

Πρωτού επιχειρήσω να πιάσω αυτόματα τις γραμμές οι οποίες θα μετρήσουν το πάχος της μεμβράνης, χρειάστηκε να γράψω ένα μικρό κώδικα τον οποίο θα χρησιμοποιούσα για να μετατρέψω την απόσταση μιας γραμμής του αποκομμένου κομματιού στην πραγματική κλίμακα του film.

Για να μπορέσω να φέρω το ζητούμενο αυτό εις πέρας, θα χρειαζομουν τις πραγματικές διαστάσεις του film που ήταν 90 mm x 60 mm, τα dimensions του film που τα βρήκα με την εντολή shape και τα dimensions του αποκομμένου κομματιού που πήραμε. Για να μετατρέψω το αποκομμένο κομμάτι στις πραγματικές του διαστάσεις στο film χρησιμοποίησα τις εξισώσεις: $\text{new_real_x} = \text{new_x} * \text{real_x} / \text{og_x}$ και $\text{new_real_y} = \text{new_y} * \text{real_y} / \text{og_y}$ όπου το new_x είναι ίσο με το πλάτος του αποκομμένου κομματιού σε pixels με τη χρήση της εντολής shape[1], το real_x είναι ίσο με το πραγματικό πλάτος του film, δηλαδή 60 mm, το og_x είναι το πλάτος του film σε pixels που πήραμε και πάλι από την εντολή shape[1] και η εξίσωση μας δίνει το πραγματικό πλάτος του αποκομμένου κομματιού σε mm. Η δεύτερη εξίσωση κάνει την ίδια δουλειά με την πρώτη, απλά για το ύψος του κομματιού αυτή τη φορά.

Στη συνέχεια υλοποίησα μια λειτουργία η οποία μας δίνει τη δυνατότητα να επιλέξουμε δύο σημεία πάνω στη μεμβράνη και το πρόγραμμα τραβάει μεταξύ τους μια γραμμή και υπολογίζει την απόστασή μεταξύ τους. Αυτό το επιτυγχάνει με τη χρήση της ίδιας εξίσωσης που αναφέραμε πιο πάνω, απλά αυτή τη φορά υπολογίζουμε που βρίσκεται το κάθε σημείο πάνω στο film σε πραγματική κλίμακα. Οι εξισώσεις είναι οι εξής:

$\text{Point}[0] * \text{RealDim}[0] / \text{ImageDim}[0]$ και $\text{Point}[1] * \text{RealDim}[1] / \text{ImageDim}[1]$ όπου Point είναι το σημείο μας, RealDim είναι οι διαστάσεις του αποκομμένου κομματιού σε pixels που πήραμε από την εντολή shape και ImageDim είναι οι διαστάσεις του αποκομμένου κομματιού σε πραγματική κλίμακα όπως υπολογίσαμε πιο πάνω. Η πρώτη εξίσωση επιστρέφει το y coordinate του σημείου ενώ η δεύτερη επιστρέφει το x coordinate στην πραγματική κλίμακα του film.

Αφού βρήκαμε τα σημεία στην πραγματική κλίμακά τους, τότε υπολογίζουμε την Ευκλείδεια απόσταση ($\text{dist}((x, y), (a, b)) = \sqrt{(x - a)^2 + (y - b)^2}$) μεταξύ τους και αναγράφουμε την απόστασή τους πάνω στην εικόνα.

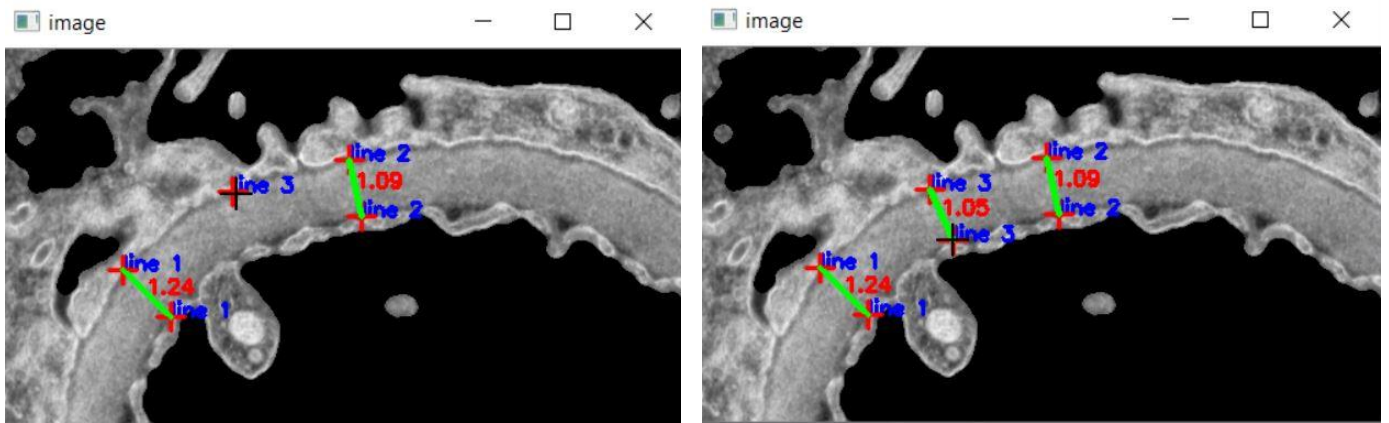


Figure 4.5.1 και Figure 4.5.2 Επιλογή αποστάσεων και υπολογισμός τους σε mm

Πιο πάνω φαίνεται η διαδικασία με την οποία επιλέγουμε τα σημεία της μεμβράνης και πάνω στην γραμμή φαίνεται η απόστασή τους σε mm όπως είναι και στο film όταν τα μετρήσουμε με ρίγα, αυτό θα μας βοηθήσει να την μετατρέψουμε αργότερα στην πραγματική τους κλίμακα με βάση την μεγένθυση του film, η οποία αναγράφεται πάνω.

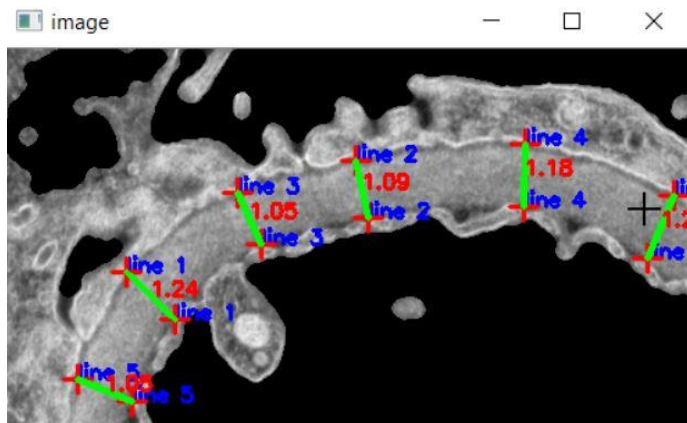


Figure 4.6 Παρουσίαση αποστάσεων στο film

Στην πάνω εικόνα φαίνονται όλες οι γραμμές που ειλέξαμε για αυτό το παράδειγμα και η κάτω εικόνα δείχνει το console του προγράμματος, όπου αναγράφεται το μήκος κάθε γραμμής και κάθε φορά υπολογίζεται ο μέσος όρος των γραμμών που επιλέχθηκαν σε mm.

```
distance for line 1 is 1.24
Average length: 1.24
distance for line 2 is 1.09
Average length: 1.165
distance for line 3 is 1.05
Average length: 1.1266666666666667
distance for line 4 is 1.18
Average length: 1.14
distance for line 5 is 1.05
Average length: 1.1219999999999999
distance for line 6 is 1.27
Average length: 1.1466666666666665
```

Figure 4.7 Παρουσίαση αποστάσεων και υπολογισμός μέσου όρου

4.4 Εύρεση σκελετού εικόνας

Όπως είχα αναφέρει και στο υποκεφάλαιο 4.2, ο λόγος που χρησιμοποιώ τον αλγόριθμο `chan_vese` και αφαιρώ το `background` της εικόνας είναι για να μπορέσω να βρώ τον σκελετό του σημείου στο οποίο θέλουμε να πάρουμε τις μετρήσεις μας. Αυτό το επιτυγχάνω με την χρήση της βιβλιοθήκης `fil_finder`.

Ο σκοπός που μπαίνω στη διαδικασία εύρεσης του σκελετού είναι για να ξέρω αργότερα όταν θα τραβάω γραμμές από τα σημεία στην περίμετρο της μεμβράνης, αν οι γραμμές περνάνε από το κέντρο του στοιχείου, πράγμα που θα εξηγηθεί πιο αναλυτικά στο επόμενο υποκεφάλαιο. Για σκοπούς παρουσίασης επιλέχθηκε άλλο κομμάτι από το `film` το οποίο είχε τον καλύτερο δυνατό σκελετό όπως φαίνεται πιο κάτω:

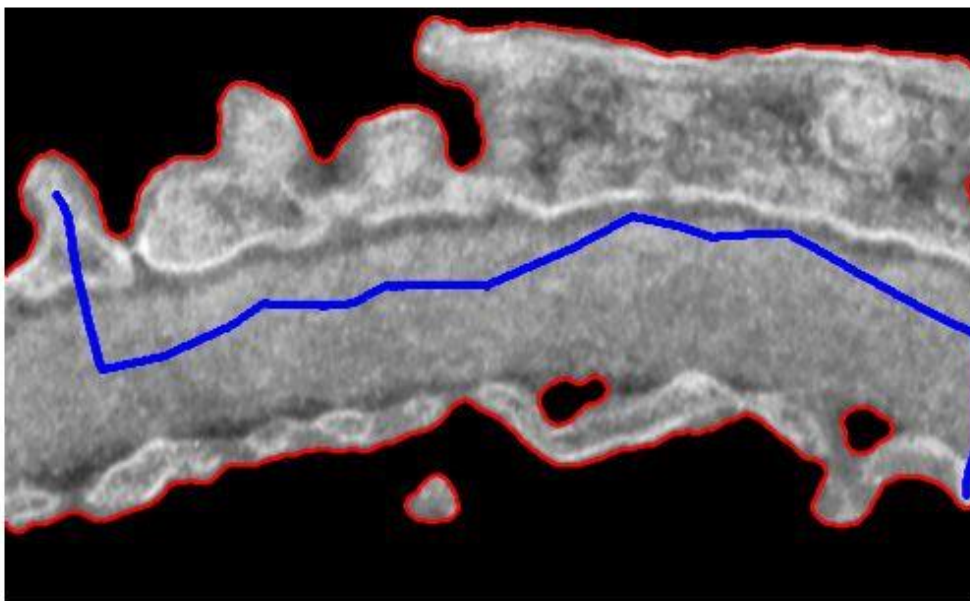


Figure 4.8 Σκελετός του κομματιού

Όπως βλέπουμε οι σκελετοί μας δεν μπορούν να είναι πάρα πολύ ακριβείς γιατί έχουμε πολλά εξογκώματα στην μεμβράνη, εν πάση περιπτώσει ο σκελετός αυτός θα μας βοηθήσει να πάρουμε τις γραμμές που θέλουμε.

4.5 Εμφάνιση γραμμών για μετρήσεις

Μετά την εύρεση του σκελετού, χρησιμοποίησα το contour area που πήρα από τον chan vese για να πάρω διάφορα τυχέα σημεία πάνω στην περίμετρό του, από τα οποία αργότερα θα τραβούσα τις γραμμές μου. Για σκοπούς ακρίβειας επιλέχθηκαν 15 τυχέα σημεία πάνω στην περίμετρο όπως φαίνεται πιο κάτω.

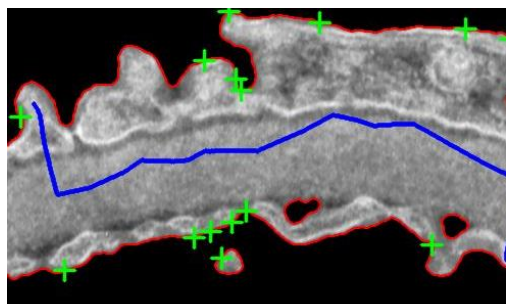


Figure 4.9 Τυχαία σημεία πάνω στη μεμβράνη

Μετά την επιλογή των σημείων, ξεκινούσα από κάθε ένα τους και τραβούσα όσο το δυνατό περισσότερες γραμμές μπορούσα ελέγχοντας ότι περνάνε από το σκελετό που βρήκαμε πιο πριν και ότι ενώνονται ξανά με την περίμετρο. Μετά την εύρεση των γραμμών αυτών, υπολόγιζα ποιά γραμμή από αυτές ήταν η μικρότερη, αυτό γιατί αν μια γραμμή ξεκινάει από ένα συγκεκριμένο σημείο, περνάει από το σκελετό, φθάνει σε άλλο σημείο της περιμέτρου και είναι η μικρότερη, τότε το πιο πιθανό να είναι η απόσταση που θέλουμε να πάρουμε. Η γραμμή που θα είχε το πιο μικρό μήκος βαφόταν με πράσινο χρώμα για να ξεχωρίζει από τις υπόλοιπες.

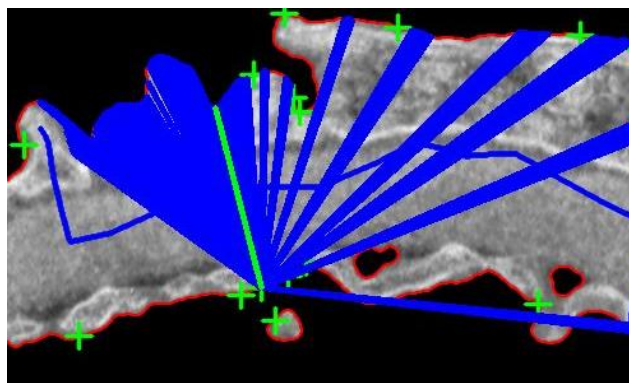
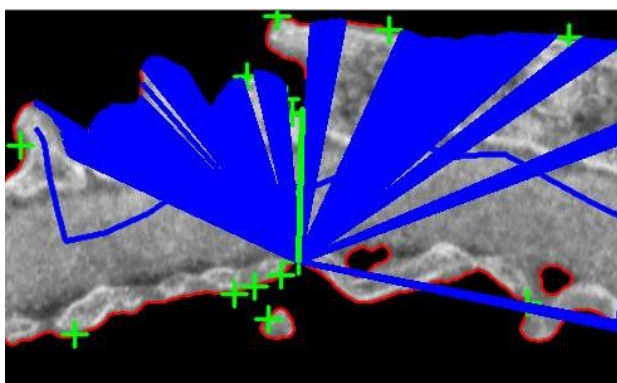


Figure 4.10.1 και Figure 4.10.2 Παραδείγματα γραμμών από τα τυχαία σημεία

Αυτό όμως αποδείχτηκε ότι δεν ήταν αρκετά ακριβή για την περίπτωσή μας, λόγω των εξογκωμάτων της μεμβράνης. Έτσι μετά από εισήγηση του κύριου Παττίχη αποφασίσαμε να μην χρησιμοποιήσουμε τον chan vese για το segmentation εφόσον δεν μπορεί να απομονώσει το σημείο της μεμβράνης που θέλουμε και να συνεχίσουμε αφήνοντας τον χρήστη να επιλέγει εκείνος την περιοχή του contour area με τη βοήθεια σημείων που θα επιλέγει με το mouse.

Κεφάλαιο 5

Τελική έκδοση του προγράμματος

5.1 Χειροκίνητη επιλογή σημείου για μετρήσεις	30
5.2 Εύρεση σκελετού και σημείων	32
5.3 Κάθετες Γραμμές πάνω στο σκελετό	33
5.4 Μετατροπή μετρήσεων σε πραγματική κλίμακα	34
5.5 Αποτελέσματα	36
5.6 Εκτίμηση σφάλματος με υπάρχοντες μετρήσεις	64

5.1 Χειροκίνητη επιλογή σημείου για μετρήσεις

Αφού το πρώτο version του προγράμματός μου με τον αλγόριθμο chan vese segmentation δεν μας έδωσε τα αναμενόμενα αποτελέσματα, συνέχισα με την δημιουργία μιας λειτουργίας η οποία θα επέτρεπε στο χρήστη να επιλέξει από μόνος του την περιοχή της μεμβράνης που θα ήθελε αυτός να μετρήσει. Αυτό το είχα πετύχει με την χρήση ενός click event το οποίο άφηνε τον χρήστη να επιλέξει τα σημεία της περιμέτρου, όπως φαίνεται στις πιο κάτω εικόνες.

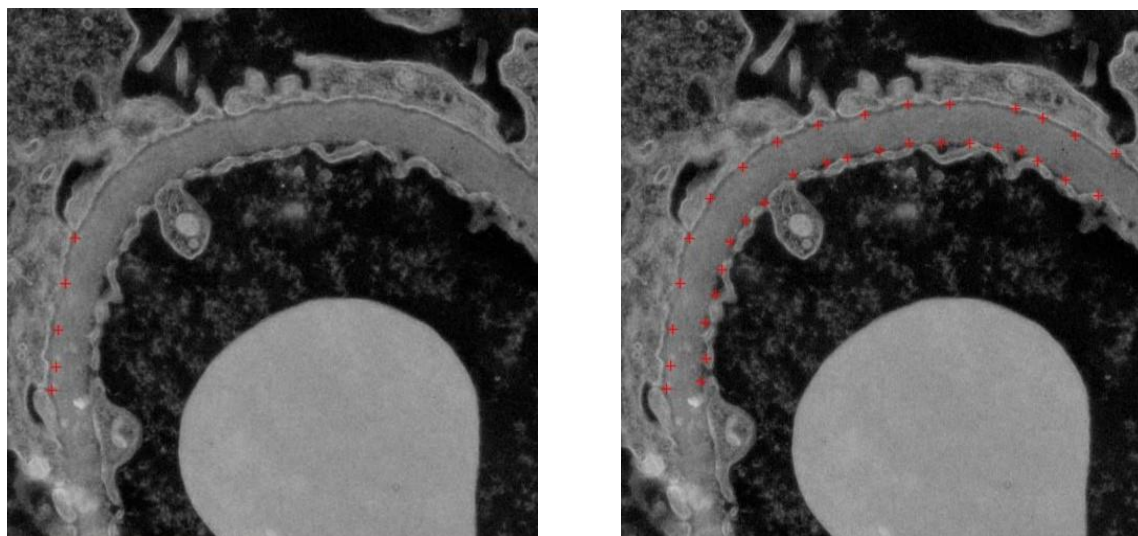


Figure 5.1.1 και Figure 5.1.2 Επιλογή περιοχής για μετρήσεις από το χρήστη

Όταν ο χρήστης επιλέγει τα σημεία, αυτά αποθηκεύονται σε ένα πίνακα ο οποίος χρησιμοποιείται από το πρόγραμμα αργότερα για να ενώσει τα σημεία αυτά με την εντολή drawContours.

Μετά την επιλογή των σημείων από τον χρήστη όπως φαίνεται στην δεύτερη εικόνα, ο χρήστης πατάει το X στο πάνω δεξιά σημείο του παραθύρου ή το κουμπί 0 και το contour area ολοκληρώνεται, υπολογίζεται και εμφανίζεται στην οθόνη όπως φαίνεται πιο κάτω.

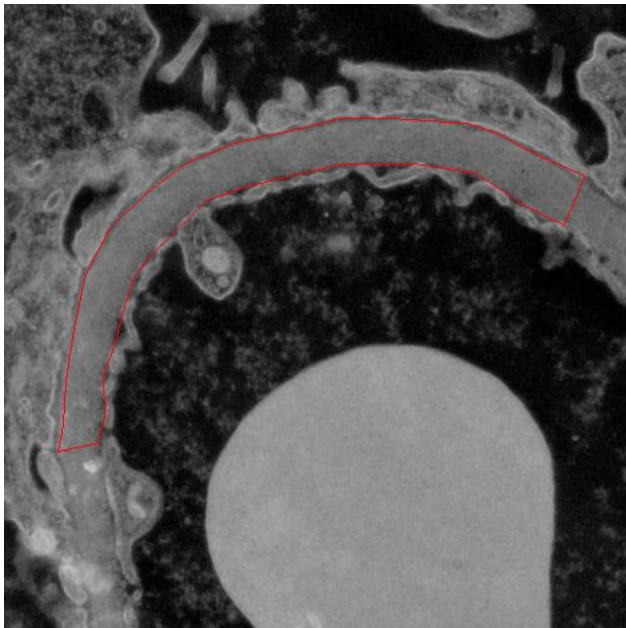


Figure 5.2 Παρουσίαση Επιλεγμένης περιοχής

Ακολούθως το εσωτερικό του contour area γεμίζεται με κόκκινο χρώμα ούτως ώστε να μπορέσουμε να αφαιρέσουμε το background του στοιχείου που θέλουμε να μετρήσουμε.

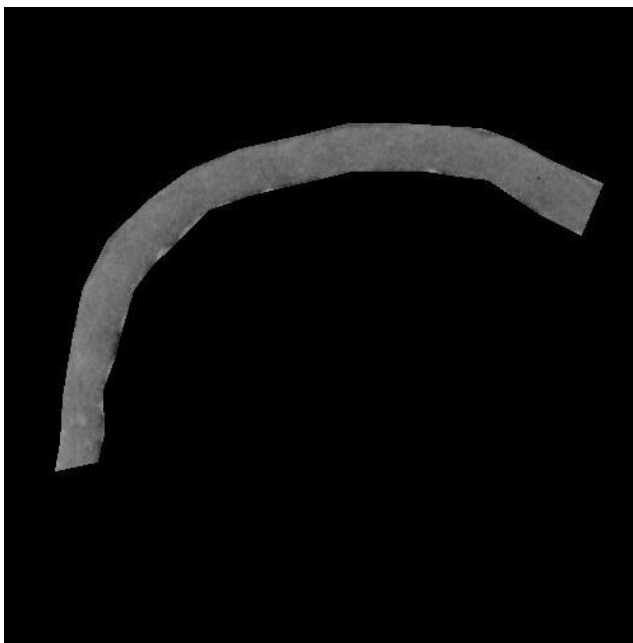


Figure 5.3 Αποκοπή περιοχής

Αυτό γίνεται για να μπορέσει μετά το πρόγραμμα να υπολογίσει το σκελετό του στοιχείου.

5.2 Εύρεση σκελετού και σημείων

Αφού αφαιρέσαμε το περίγραμμα γύρω από το contour area, το πρόγραμμά μας μπορούσε να υπολογίσει το σκελετό του στοιχείου, πάνω στον οποίο θα τραβούσα κάθετες γραμμές για να υπολογίσω το πάχος της μεμβράνης. Ο σκελετός υπολογίζεται με τον ίδιο τρόπο που προαναφέρα και στο υποκεφάλαιο 4.4, δηλαδή με τη χρήση της βιβλιοθήκης `fil_finder`. Η πιο κάτω εικόνα παρουσιάζει το σκελετό που υπολογίστηκε.

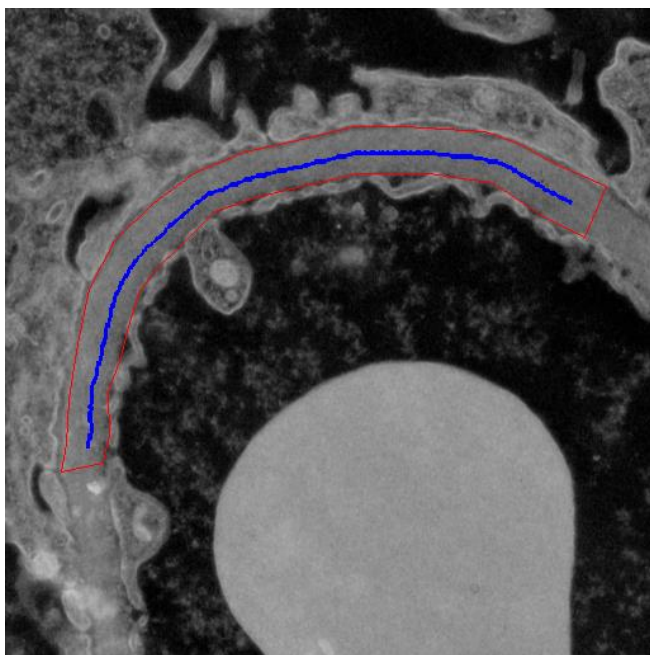


Figure 5.4 Εύρεση και παρουσίαση σκελετού της περιοχής

Μετά τον υπολογισμό του σκελετού, έπρεπε να πάρουμε αρκετά σημεία πάνω του για να μπορέσουμε να τραβήξουμε σωστές κάθετες γραμμές για ακριβή υπολογισμού του πάχους.

Αρχικά έπρεπε να βρούμε όλα τα σημεία του σκελετού. Αυτό θεώρησα ότι ήταν καλή ιδέα να το κάνω ελέγχοντας όλη την εικόνα οριζόντια για σημείο που είναι με μπλέ χρώμα (το οποίο σημαίνει ότι είναι σημείο του σκελετού). Αυτό όμως δεν δούλεψε σωστά όταν η μεμβράνη ήταν σε οριζόντια κλίση γιατί τα σημεία τα έπερνε ανακατεμένα και είχαμε πρόβλημα με τον υπολογισμό των καθέτων στη συνέχεια, αφού για να υπολογίσω την κάθετη πάνω σε ένα σημείο του σκελετού χρειάζομαι και την κλίση του σκελετού την οποία παίρνω από τα γειτονικά σημεία του σημείου που θέλω. Έτσι έπρεπε να κάνω ένα έλεγχο για να διαπιστώσω αν η κλίση του σκελετού ήταν οριζόντια (όπως είναι και στο παράδειγμά μας δηλαδή), ή κάθετη.

Πήρα αρχικά όλα τα σημεία του σκελετού όπως θα τα έπαιρνα κανονικά και έλεγα τις διαφορές του μικρότερου x coordinate με το μεγαλύτερο x coordinate και του μικρότερου y coordinate με το μεγαλύτερο y coordinate. Μετά έπαιρνα αυτές τις δύο διαφορές και έλεγα ποιά είναι η μεγαλύτερη. Αν η διαφορά των x είναι μεγαλύτερη από τη διαφορά των y τότε αυτό σημαίνει ότι ο σκελετός μας είναι οριζόντιος, ενώ αν η διαφορά των y είναι μεγαλύτερη από τη διαφορά των x τότε σημαίνει ότι ο σκελετός έχει κάθετη κλίση. Αφότου βρήκαμε την κλίση του σκελετού μπορούσαμε να πάρουμε τα σημεία του σκελετού με τη σωστή τους σειρά. Αν βρήκαμε οριζόντια κλίση τότε θα κάνουμε τον έλεγχο για τα σημεία μας κάθετα, ενώ αν βρήκαμε κάθετη κλίση ο έλεγχος θα γινόταν οριζόντια.

5.3 Κάθετες Γραμμές πάνω στο σκελετό

Μετά την εύρεση των σημείων πάνω στο σκελετό, επέλεξα 17 από εκείνα για τη δημιουργία κάθετων γραμμών. Αν το πρόγραμμα αποτύχει να πάρει καλή κάθετη γραμμή, τότε αυτή απορρίπτεται από το πρόγραμμα. Κακές κάθετες μπορούν να παρθούν αν το πρόγραμμα δεν μπορεί να βρεί τα σημεία σύγκρουσης της κάθετης γραμμής με την περίμετρο. Τέλος παρουσιάζονται οι κάθετες γραμμές πάνω στην μεμβράνη μαζί με τον αριθμό κάθε γραμμής για να μπορέσουμε αργότερα που θα υπολογιστούν οι αποστάσεις να ξέρουμε σε ποιά απόσταση αντιστοιχεί σε ποιά γραμμή. Το αποτέλεσμα της διαδικασίας αυτής παρουσιάζεται στην εικόνα πιο κάτω.

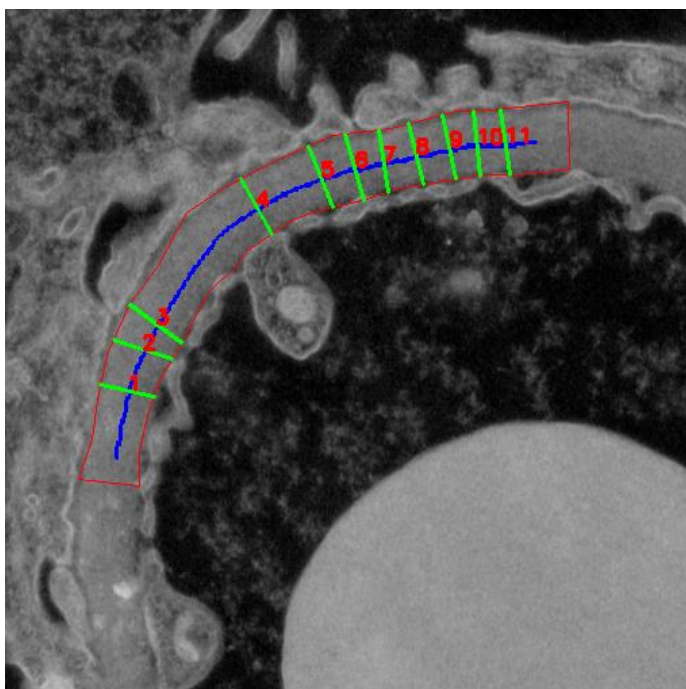


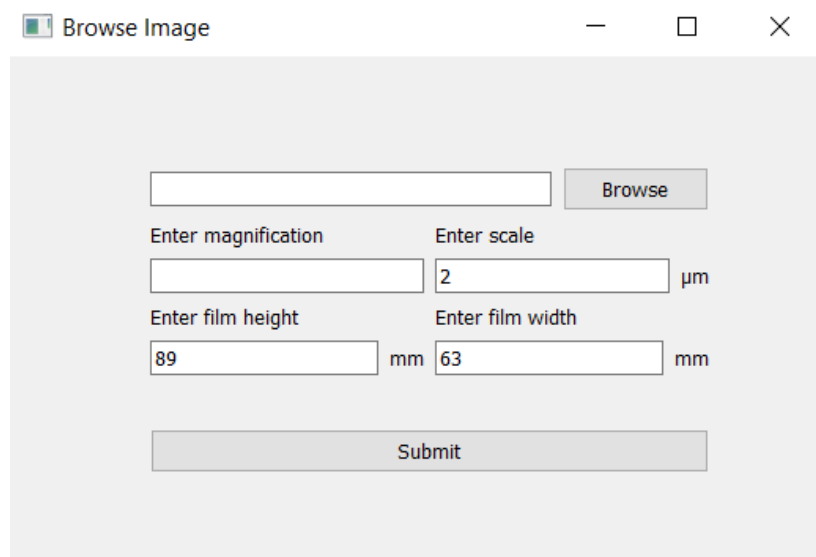
Figure 5.5 Παρουσίαση αριθμημένων κάθετων γραμμών πάνω στο σκελετό

Όπως βλέπουμε από το παράδειγμα λήφθηκαν 11 διαφορετικές αποστάσεις.

5.4 Μετατροπή μετρήσεων σε πραγματική κλίμακα

Αφού βρήκαμε τα σημεία της κάθε γραμμής τότε υπολογίσαμε την απόσταση της κάθε μιας σε mm όπως αναφέραμε στο υποκεφάλαιο 4.3. Όμως όταν παρουσίασα τα αποτελέσματα του δόκτωρ Κυριάκου σε mm αυτός μου είπε πως θα ήταν καλό να μετατρέπω τις αποστάσεις αυτές στην πραγματική τους κλίμακα σε nanometers.

Για να κάνω τη μετατροπή αυτή χρειαζόμουν τη μεγένθυση του film η οποία μπορούσε να κυμαίνεται από 3000 μέχρι και 10000. Τα films που μου έδωσε ο δόκτωρ όμως κυμαίνονταν όλα σε δύο διαφορετικές μεγενθύνσεις, 3000 και 4000. Εφόσον όμως κάθε film είχε διαφορετική μεγένθυση, δημιούργησα ένα UI το οποίο θα χρησιμοποιούσε ο χρήστης για να εισάγει σημαντικές πληροφορίες στο πρόγραμμα όπως τη μεγένθυση, την κλίμακα σε μm το ύψος και το πλάτος του film σε mm και την εικόνα του film που θα χρησιμοποιούσαμε για να πάρουμε μετρήσεις. Το UI αυτό φαίνεται πιο κάτω.



The screenshot shows a web browser window with the title "Browse Image". Inside the browser, there is a form with the following fields and labels:

- A text input field at the top.
- A "Browse" button next to the top input field.
- A label "Enter magnification" above a text input field.
- A label "Enter scale" above a text input field containing the value "2".
- A unit "μm" to the right of the "Enter scale" input field.
- A label "Enter film height" above a text input field containing the value "89".
- A unit "mm" to the right of the "Enter film height" input field.
- A label "Enter film width" above a text input field containing the value "63".
- A unit "mm" to the right of the "Enter film width" input field.
- A "Submit" button at the bottom of the form.

Figure 5.6 UI με το οποίο εισάγει ο χρήστης πληροφορίες

Όπως βλέπουμε στην πιο πάνω εικόνα στοιχεία όπως το ύψος, το πλάτος και η κλίμακα του film είναι δοσμένα από το πρόγραμμα γιατί συνήθως είναι σταθερά σε κάθε film.

Για να μετατρέψω τις αποστάσεις στην πραγματική τους κλίμακα χρησιμοποίησα την εξίσωση: $\text{distance} = \text{distance} * 1000 / \text{magnification} * 1000$, όπου παίρνω την απόσταση που υπολογίσαμε σε mm, την μετατρέπω σε microns επειδή η κλίμακα είναι πάντα σε microns, διαιρώ δια τη μεγένθυση και μετά πολλαπλασιάζω επί 1000 για να μετατρέψω τα microns σε nanometers.

Για να μπορέσω να δείξω στο χρήστη τα αποτελέσματα των μετρήσεων μου, δημιούργησα ένα message box το οποίο έδειχνε τα εξής:

- Τον αριθμό των καθέτων γραμμών που βρήκαμε
- Τον αριθμό κάθε γραμμής μαζί με την απόστασή της σε nanometers
- Την μικρότερη απόσταση σε nanometers
- Την μεγαλύτερη απόσταση σε nanometers
- Το μέσο όρο των αποστάσεων σε nanometers
- Το standard deviation των αποστάσεων σε nanometers
- Τη μέση τιμή των αποστάσεων σε nanometers
- Το πρώτο και τρίτο τεταρτημόριο (first and third quartiles) σε nanometers
- Το interquartile σε nanometers

Το message box αυτό φαίνεται πιο κάτω για τα αποτελέσματα του παραδείγματός μας με μεγένθυση 3000.

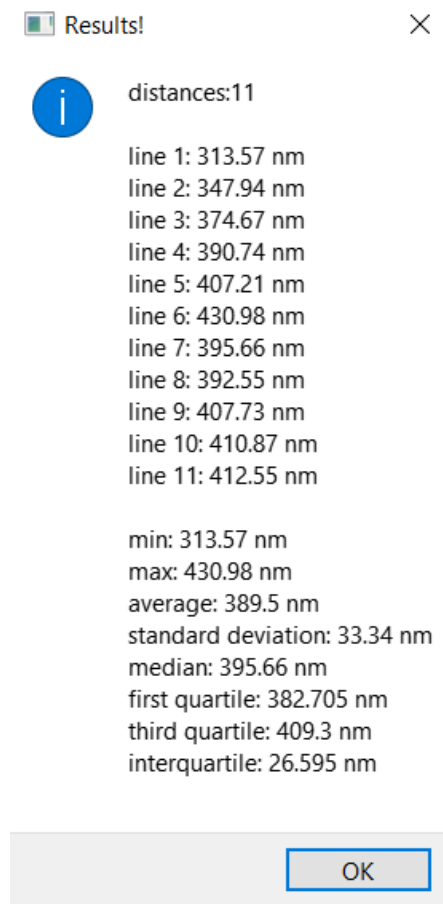


Figure 5.7.1 Παρουσίαση μετρήσεων

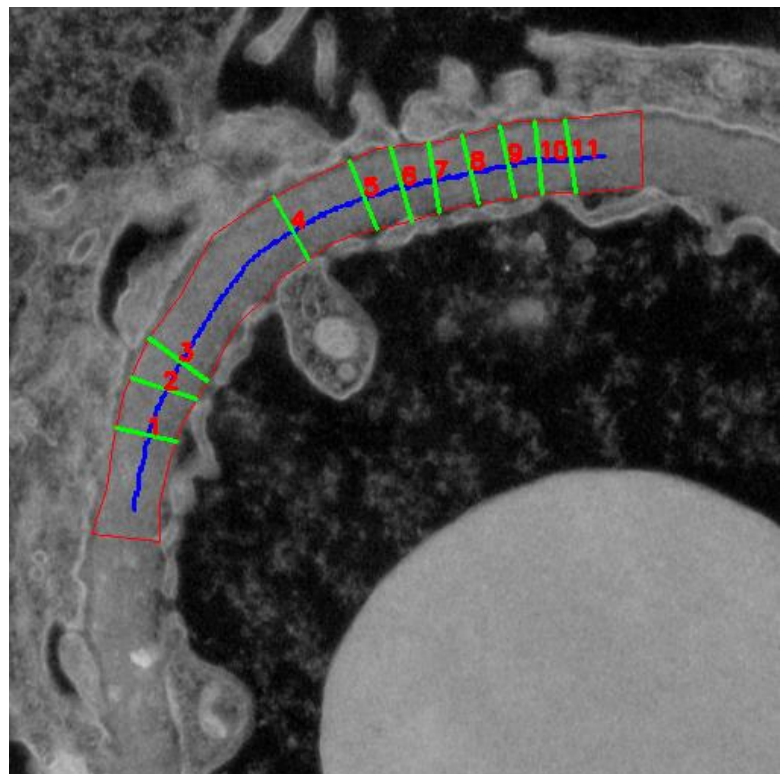


Figure 5.7.2 Παρουσίαση αριθμημένων γραμμών

5.4 Τελικά αποτελέσματα

Πιο κάτω ακολουθούν τα αποτελέσματα των μετρήσεων για κάθε δείγμα που μου έδωσε ο δόκτωρ Κυριάκου.

Sample 1:

Part 1:

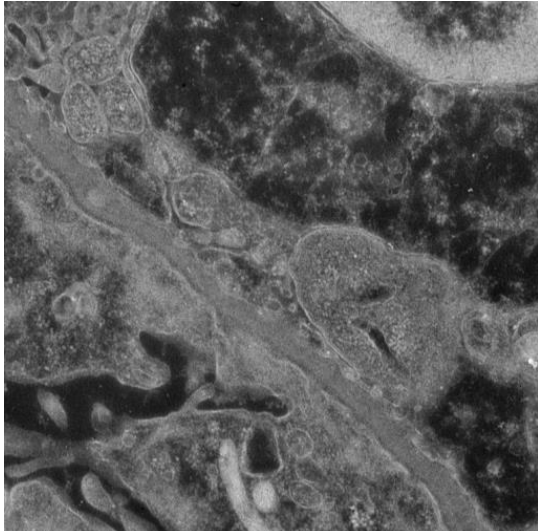


Figure 5.8.1 Κομμάτι της μεμβράνης

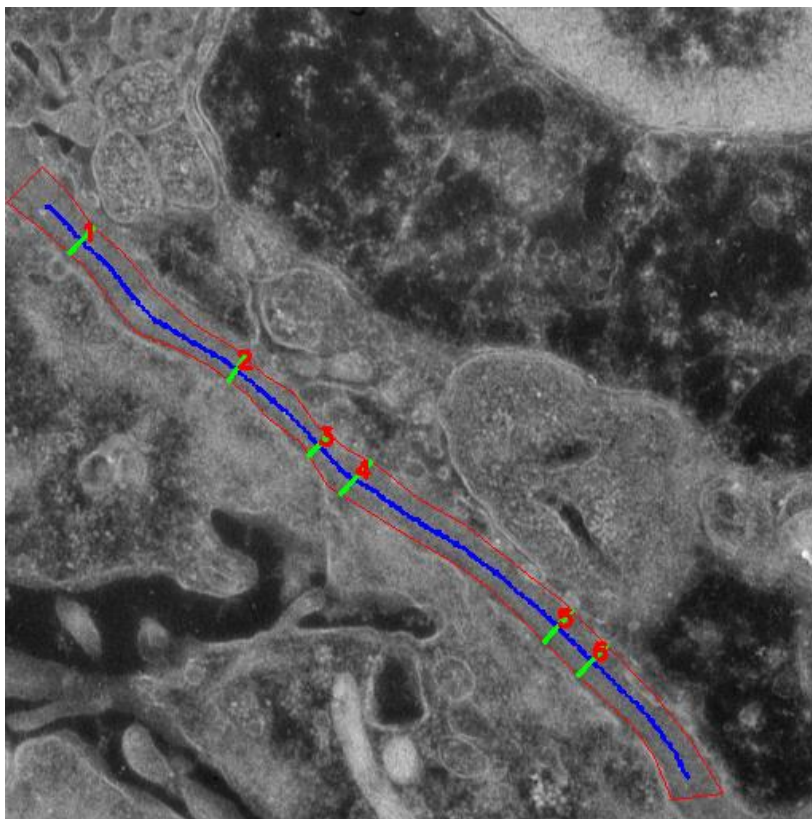


Figure 5.8.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	169.36 nm
2	141.44 nm
3	141.51 nm
4	212.89 nm
5	203.68 nm
6	202.0 nm

Table 5.1.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
141.44 nm	212.89 nm	178.48 nm	32.231 nm	185.68 nm	155.435 nm	202.84 nm	47.405 nm

Table 5.1.2 Ολικές εκτιμήσεις

Sample 1:
Part 2:

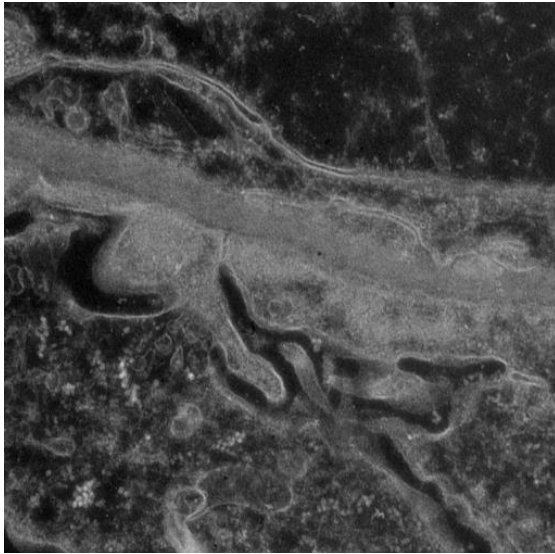


Figure 5.9.1 Κομμάτι της μεμβράνης

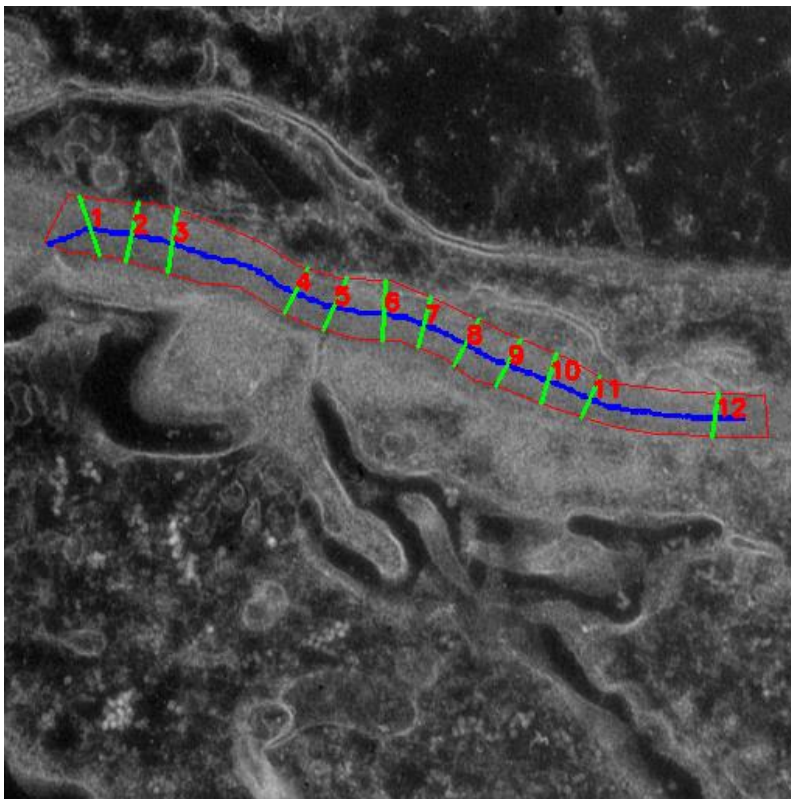


Figure 5.9.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	263.54 nm
2	256.3 nm
3	274.03 nm
4	212.1 nm
5	236.38 nm
6	258.7 nm
7	217.38 nm
8	220.65 nm
9	220.65 nm
10	220.32 nm
11	186.65 nm
12	187.07 nm

Table 5.2.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
186.65 nm	274.03 nm	229.48 nm	28.742 nm	220.65 nm	214.74 nm	257.5 nm	42.76 nm

Table 5.2.2 Ολικές εκτιμήσεις

Sample 1:
Part 3:

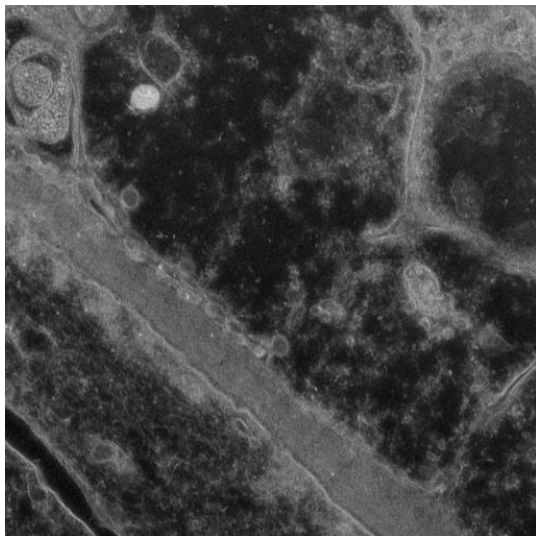


Figure 5.10.1 Κομμάτι της μεμβράνης

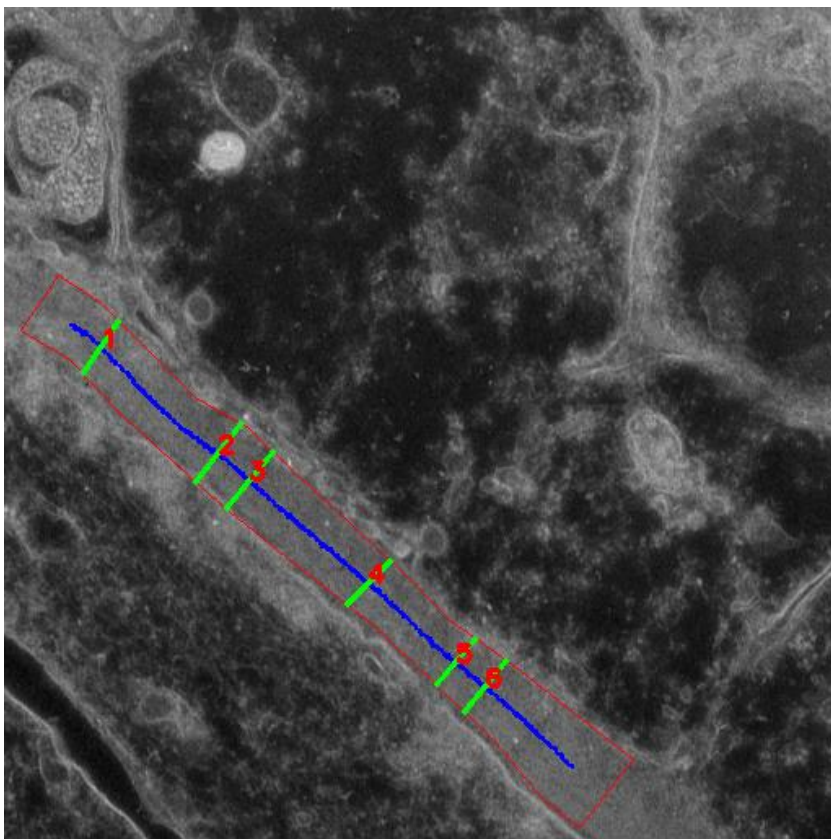


Figure 5.10.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	292.39 nm
2	341.51 nm
3	339.34 nm
4	282.43 nm
5	274.87 nm
6	308.11 nm

Table 5.3.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
274.87 nm	341.51 nm	306.44 nm	28.585 nm	300.25 nm	287.41 nm	323.725 nm	36.315 nm

Table 5.3.2 Ολικές εκτιμήσεις

Sample 2:
Part 1:

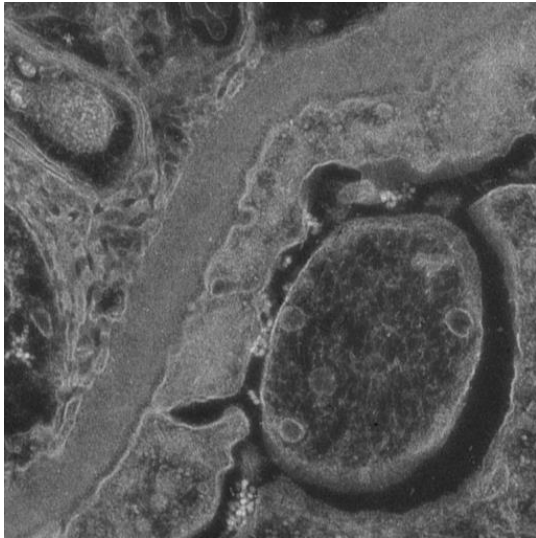


Figure 5.11.1 Κομμάτι της μεμβράνης

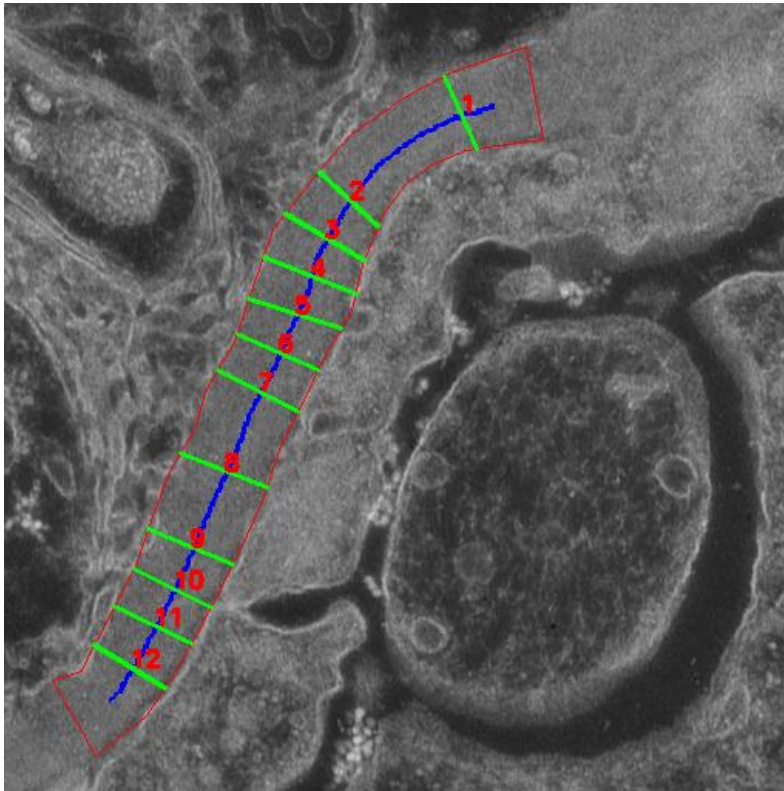


Figure 5.11.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	296.17 nm
2	288.25 nm
3	324.45 nm
4	350.64 nm
5	340.06 nm
6	309.23 nm
7	320.03 nm
8	326.34 nm
9	323.79 nm
10	305 nm
11	302.26 nm
12	295.64 nm

Table 5.4.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
288.25 nm	350.64 nm	315.16 nm	18.971 nm	314.63 nm	299.215 nm	325.395 nm	26.18 nm

Table 5.4.2 Ολικές εκτιμήσεις

Sample 2:

Part 2:

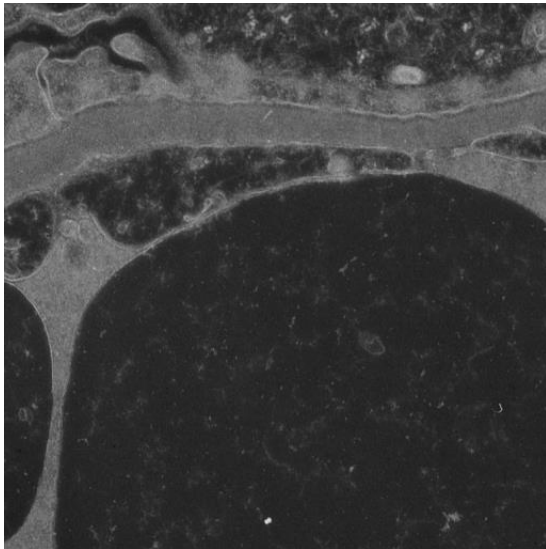


Figure 5.12.1 Κομμάτι της μεμβράνης

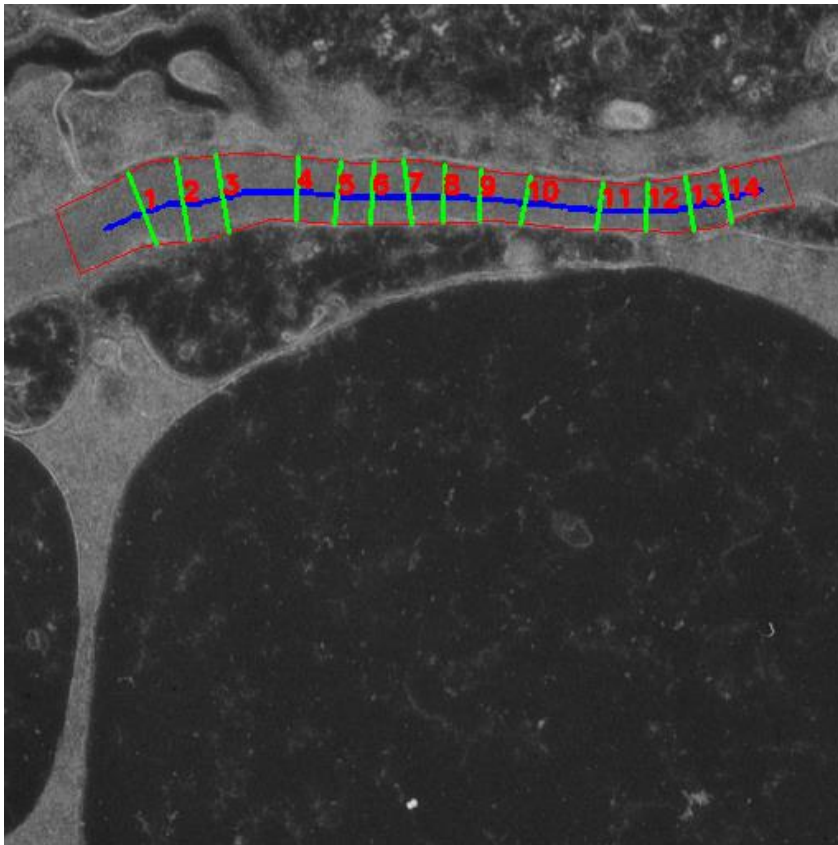


Figure 5.12.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	385.84 nm
2	417.86 nm
3	401.19 nm
4	328.99 nm
5	321.96 nm
6	320.84 nm
7	331.18 nm
8	295.27 nm
9	261.55 nm
10	248.95 nm
11	238.16 nm
12	244.69 nm
13	280.99 nm
14	273.85 nm

Table 5.5.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
238.16 nm	417.86 nm	310.81 nm	58.591 nm	308.05 nm	267.7 nm	330.085 nm	62.385 nm

Table 5.5.2 Ολικές εκτιμήσεις

Sample 2:
Part 3:

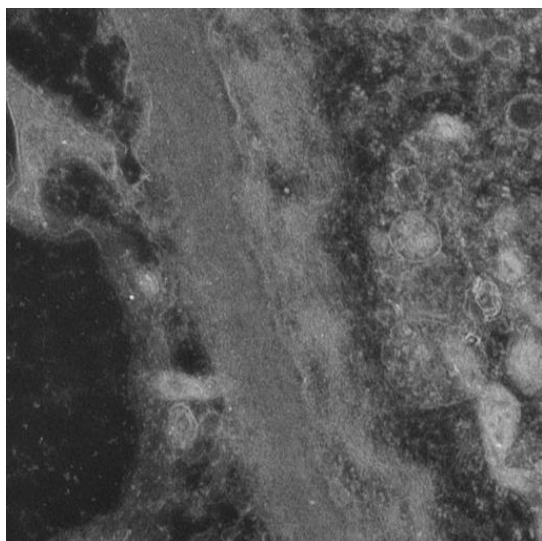


Figure 5.13.1 Κομμάτι της μεμβράνης

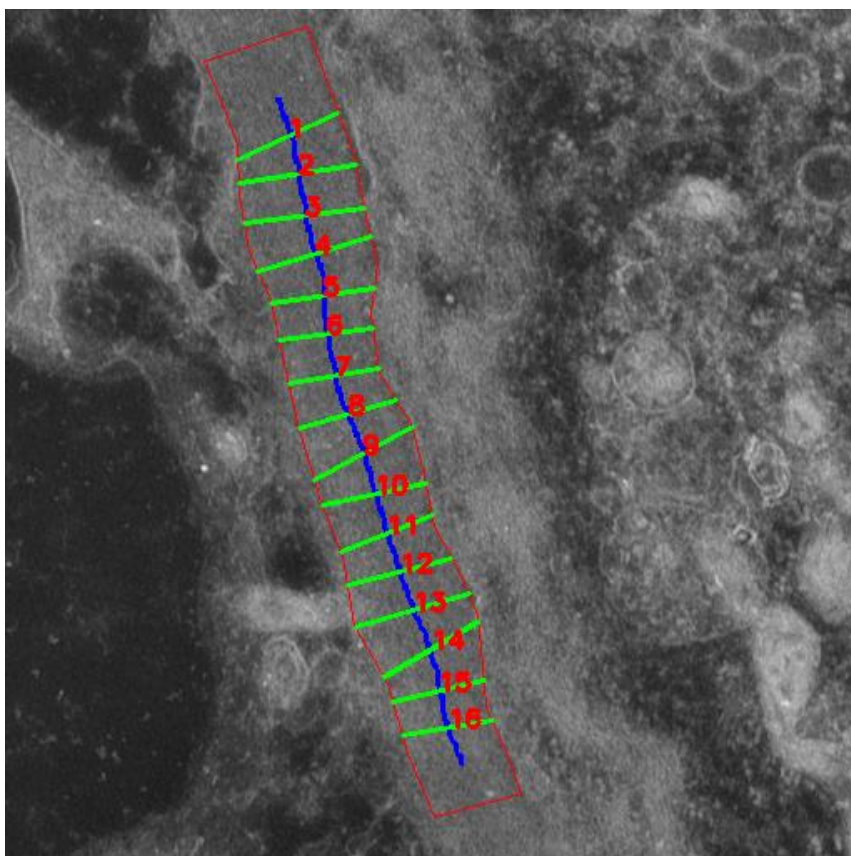


Figure 5.13.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	349.07 nm
2	368.41 nm
3	376.95 nm
4	372.14 nm
5	325.62 nm
6	293.18 nm
7	284.7 nm
8	315.81 nm
9	355.37 nm
10	329.9 nm
11	311.16 nm
12	334.07 nm
13	370.38 nm
14	345.07 nm
15	294.6 nm
16	284.7 nm

Table 5.6.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
284.7 nm	376.95 nm	331.95 nm	32.213 nm	331.985 nm	302.88 nm	361.89 nm	59.01 nm

Table 5.6.2 Ολικές εκτιμήσεις

Sample 2:
Part 4:

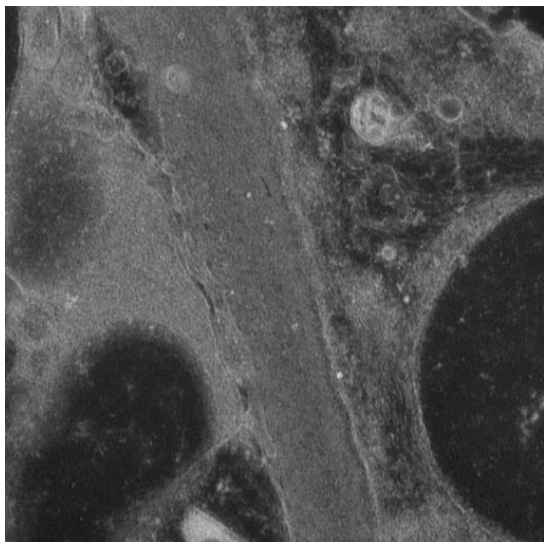


Figure 5.14.1 Κομμάτι της μεμβράνης

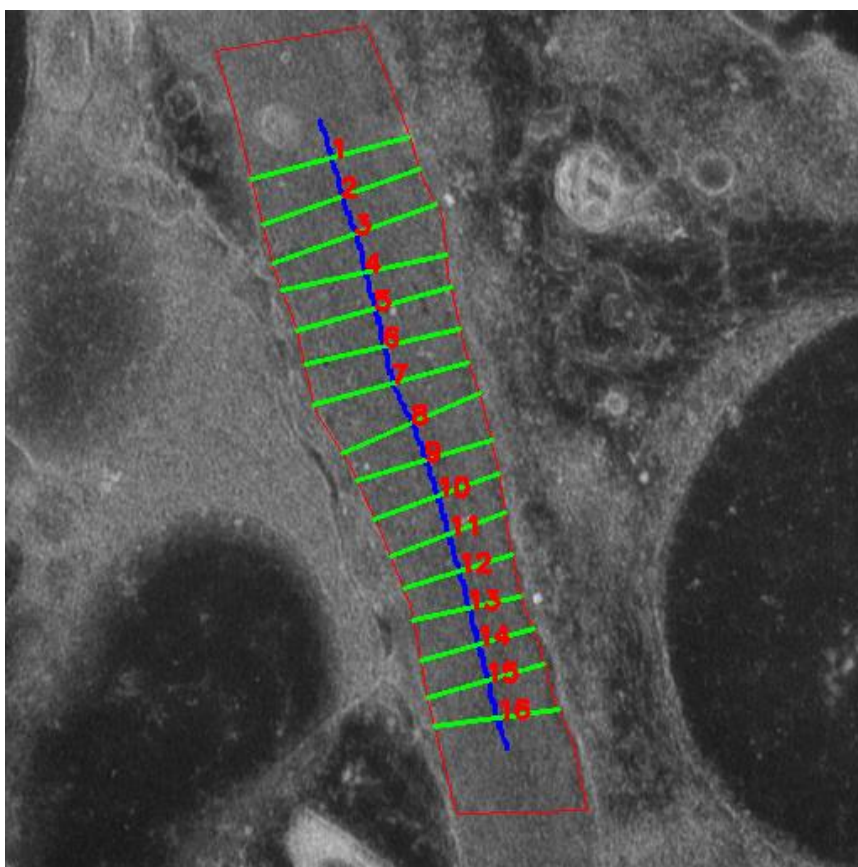


Figure 5.14.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	457.43 nm
2	466 nm
3	487.11 nm
4	470.39 nm
5	449.93 nm
6	443.12 nm
7	448.46 nm
8	424.79 nm
9	393.38 nm
10	372.17 nm
11	344.27 nm
12	315.99 nm
13	307.53 nm
14	327.76 nm
15	338.2 nm
16	349.64 nm

Table 5.7.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
307.53 nm	487.11 nm	399.76 nm	62.513 nm	409.085 nm	341.235 nm	453.68 nm	112.445 nm

Table 5.7.2 Ολικές εκτιμήσεις

Sample 3: Part 1:



Figure 5.15.1 Κομμάτι της μεμβράνης

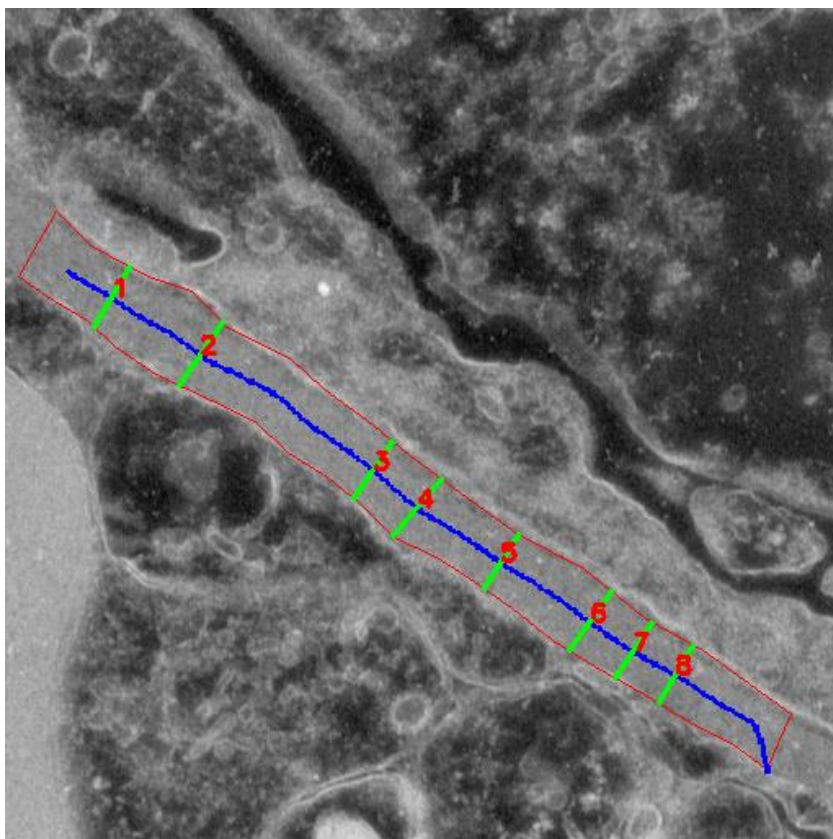


Figure 5.15.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	257.97 nm
2	282.68 nm
3	253 nm
4	272.25 nm
5	233.85 nm
6	264.19 nm
7	244.84 nm
8	244.53 nm

Table 5.8.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
233.85 nm	282.68 nm	256.66 nm	16.031 nm	255.485 nm	244.685 nm	268.22 nm	23.535 nm

Table 5.8.2 Ολικές εκτιμήσεις

Sample 3:
Part 2:

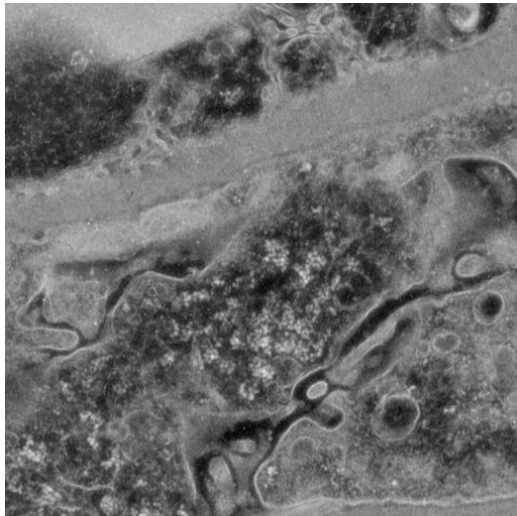


Figure 5.16.1 Κομμάτι της μεμβράνης

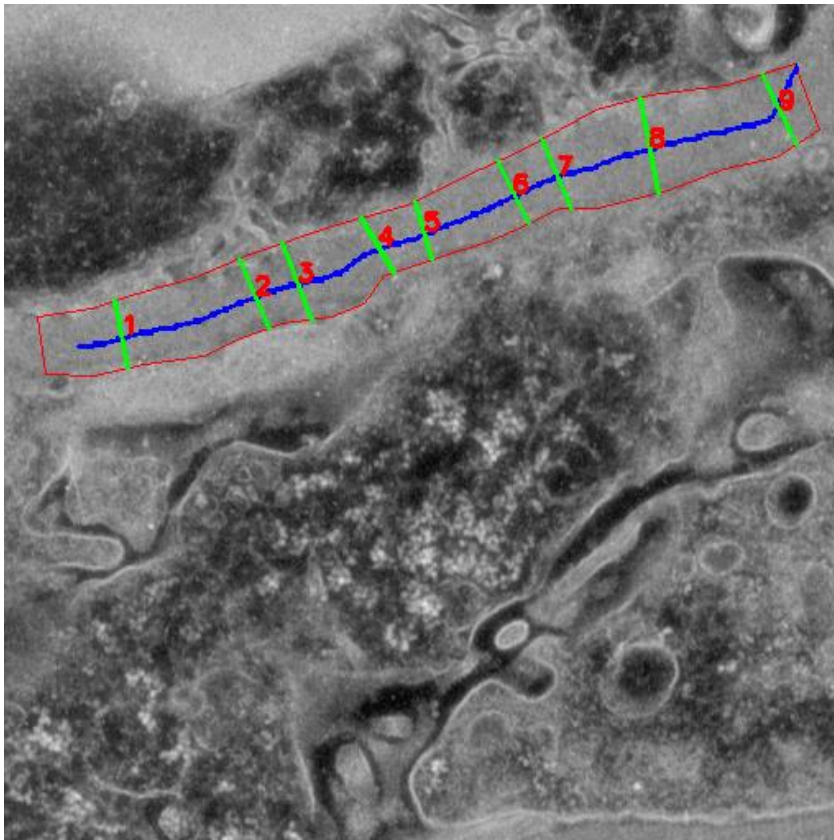


Figure 5.16.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	307.36 nm
2	332.81 nm
3	367.43 nm
4	278.26 nm
5	266.95 nm
6	308.22 nm
7	332.81 nm
8	434.44 nm
9	347.4 nm

Table 5.9.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
266.95 nm	434.44 nm	330.63 nm	50.328 nm	332.81 nm	307.36 nm	347.4 nm	40.04 nm

Table 5.9.2 Ολικές εκτιμήσεις

Sample 3:
Part 3:

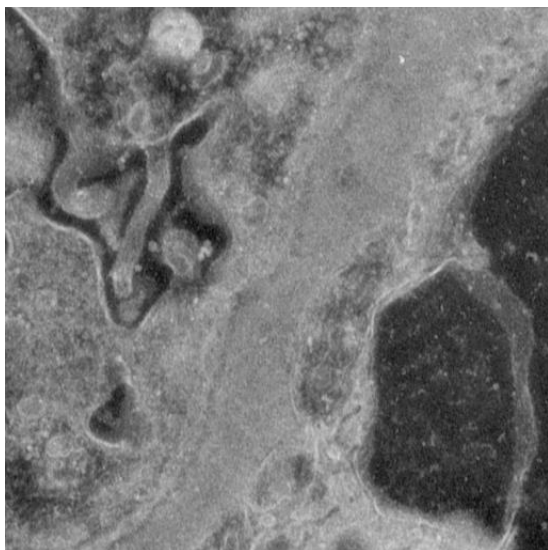


Figure 5.17.1 Κομμάτι της μεμβράνης

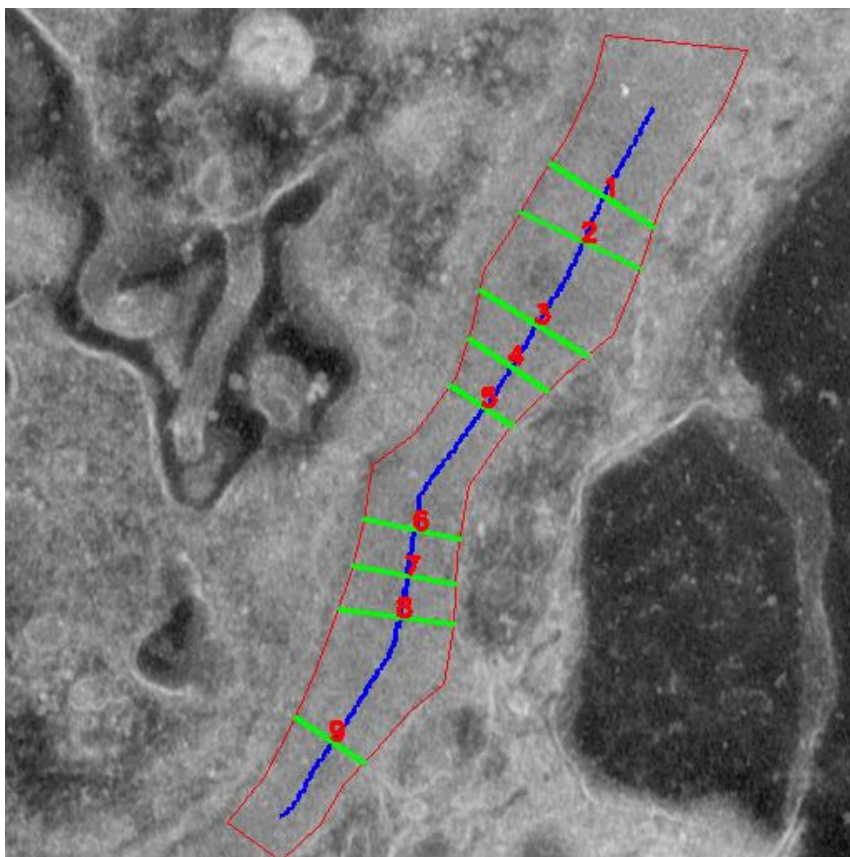


Figure 5.17.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	310.93 nm
2	337.46 nm
3	320.39 nm
4	241.44 nm
5	188.99 nm
6	247.51 nm
7	262.86 nm
8	290.07 nm
9	210.5 nm

Table 5.10.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
188.99 nm	337.46 nm	267.79 nm	50.744 nm	262.86 nm	241.44 nm	310.93 nm	69.49 nm

Table 5.10.2 Ολικές εκτιμήσεις

Sample 3:
Part 4:

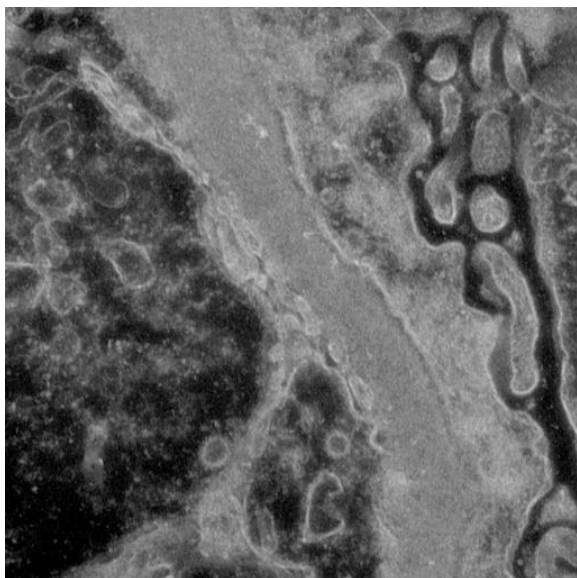


Figure 5.18.1 Κομμάτι της μεμβράνης

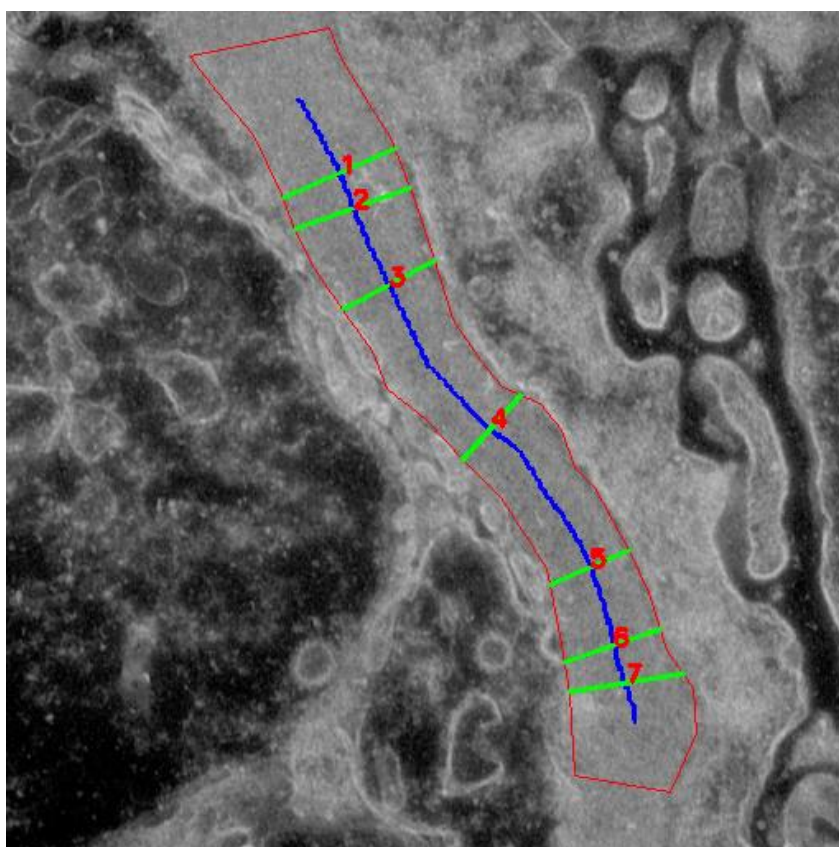


Figure 5.18.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	383.31 nm
2	381.45 nm
3	329.57 nm
4	280.48 nm
5	270.14 nm
6	314.63 nm
7	355.4 nm

Table 5.11.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
270.14 nm	383.31 nm	330.71 nm	45.465 nm	329.57 nm	297.555 nm	368.425 nm	70.87 nm

Table 5.11.2 Ολικές εκτιμήσεις

Sample 4:
Part 1:

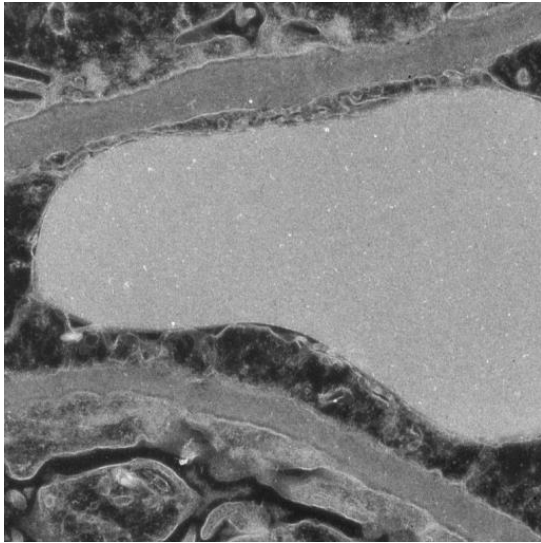


Figure 5.19.1 Κομμάτι της μεμβράνης

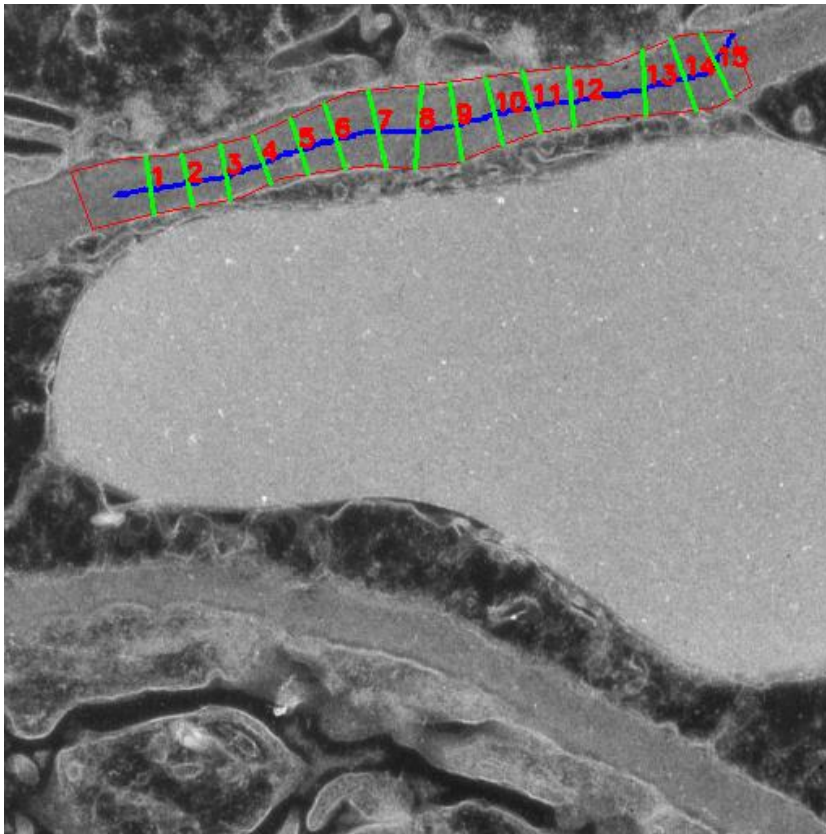


Figure 5.19.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	375.14 nm
2	346.95 nm
3	366.08 nm
4	326.67 nm
5	377.2 nm
6	443.93 nm
7	511.04 nm
8	553.92 nm
9	516.08 nm
10	448.94 nm
11	418.08 nm
12	384.54 nm
13	425.47 nm
14	493.43 nm
15	454.79 nm

Table 5.12.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
326.67 nm	553.92 nm	429.48 nm	67.722 nm	425.47 nm	376.17 nm	474.11 nm	97.94 nm

Table 5.12.2 Ολικές εκτιμήσεις

Sample 4:
Part 2:

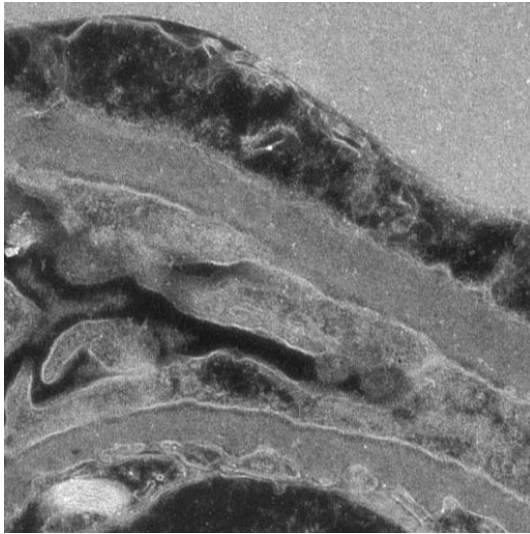


Figure 5.20.1 Κομμάτι της μεμβράνης

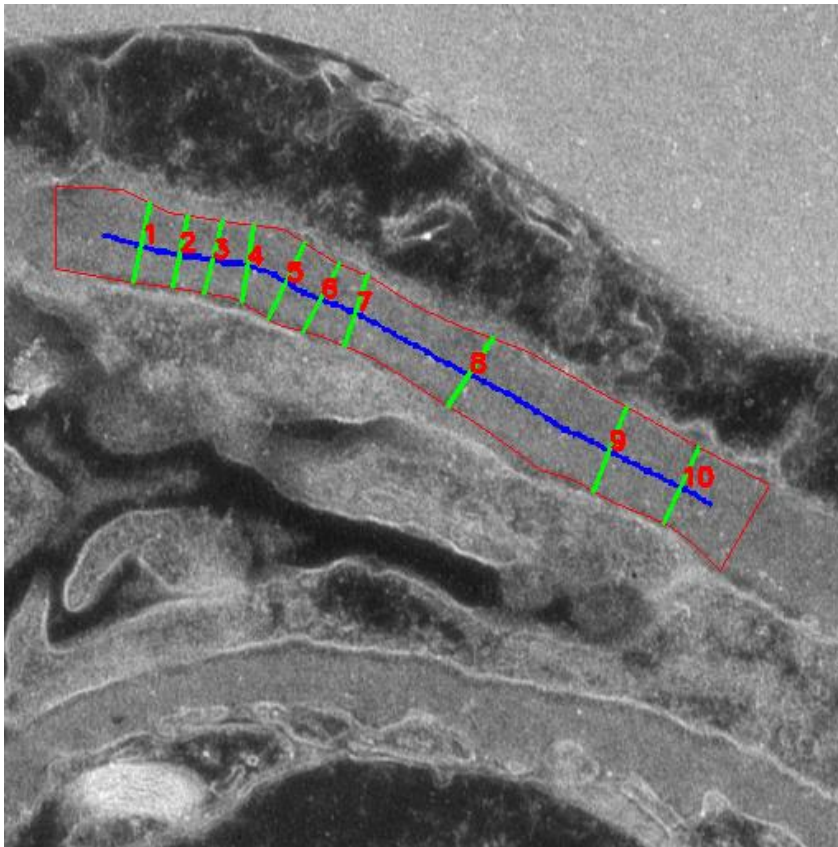


Figure 5.20.2 Γραμμές πάνω στο κομμάτι

Line n°	Distances:
1	300.27 nm
2	269.04 nm
3	277.42 nm
4	285.33 nm
5	309.11 nm
6	281.32 nm
7	275.89 nm
8	295.13 nm
9	333.34 nm
10	306.84 nm

Table 5.13.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
269.04 nm	333.34 nm	293.37 nm	19.548 nm	290.23 nm	279.37 nm	303.555 nm	24.185 nm

Table 5.13.2 Ολικές εκτιμήσεις

Sample 4:
Part 3:

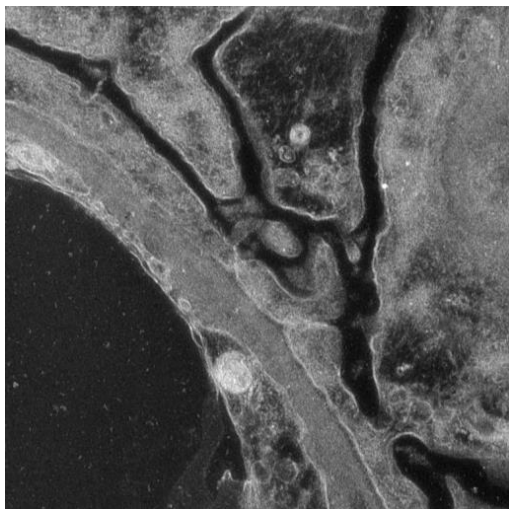


Figure 5.21.1 Κομμάτι της μεμβράνης

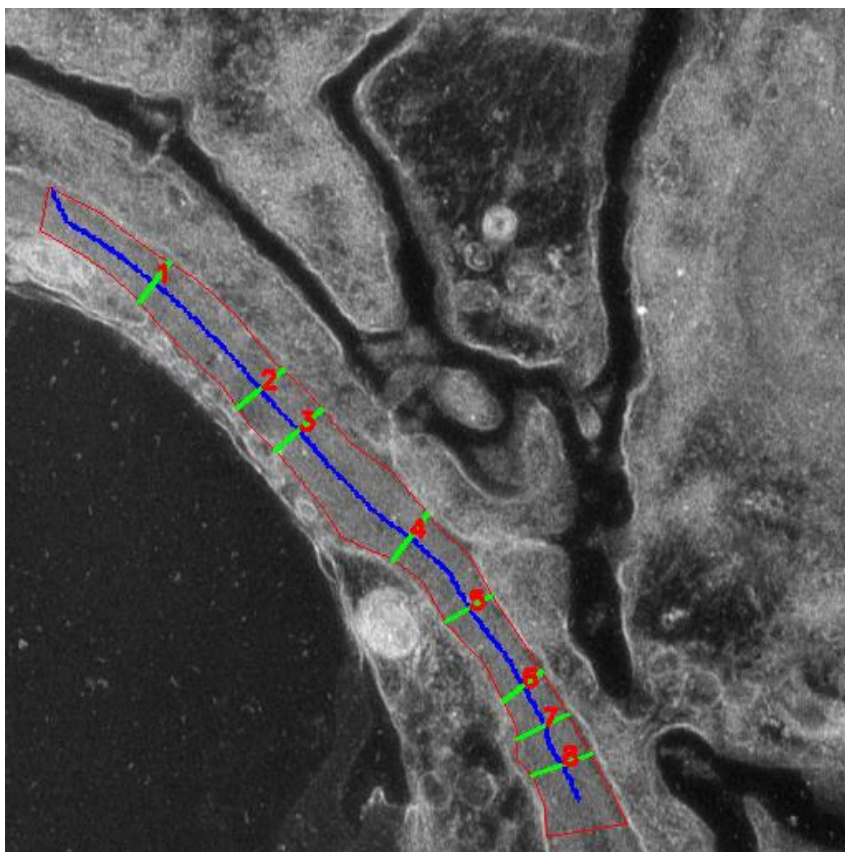


Figure 5.21.2 Γραμμές πάνω στο κομμάτι

Line n°	Distances:
1	192.18 nm
2	228.07 nm
3	236.99 nm
4	224.21 nm
5	200.57 nm
6	183.51 nm
7	213.25 nm
8	235.63 nm

Table 5.14.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
183.51 nm	236.99 nm	214.3 nm	20.3 nm	218.73 nm	196.375 nm	231.85 nm	35.475 nm

Table 5.14.2 Ολικές εκτιμήσεις

Sample 5:

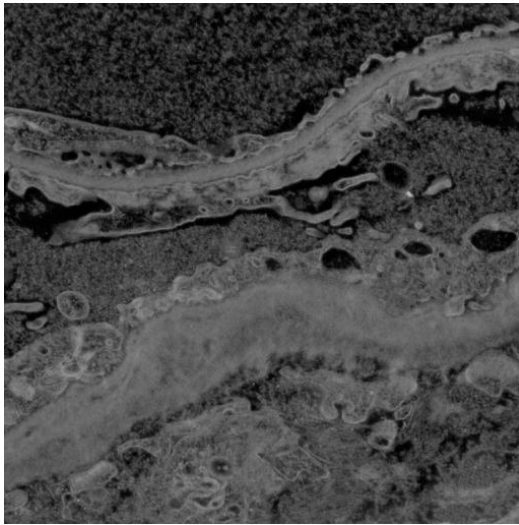


Figure 5.22.1 Κομμάτι της μεμβράνης

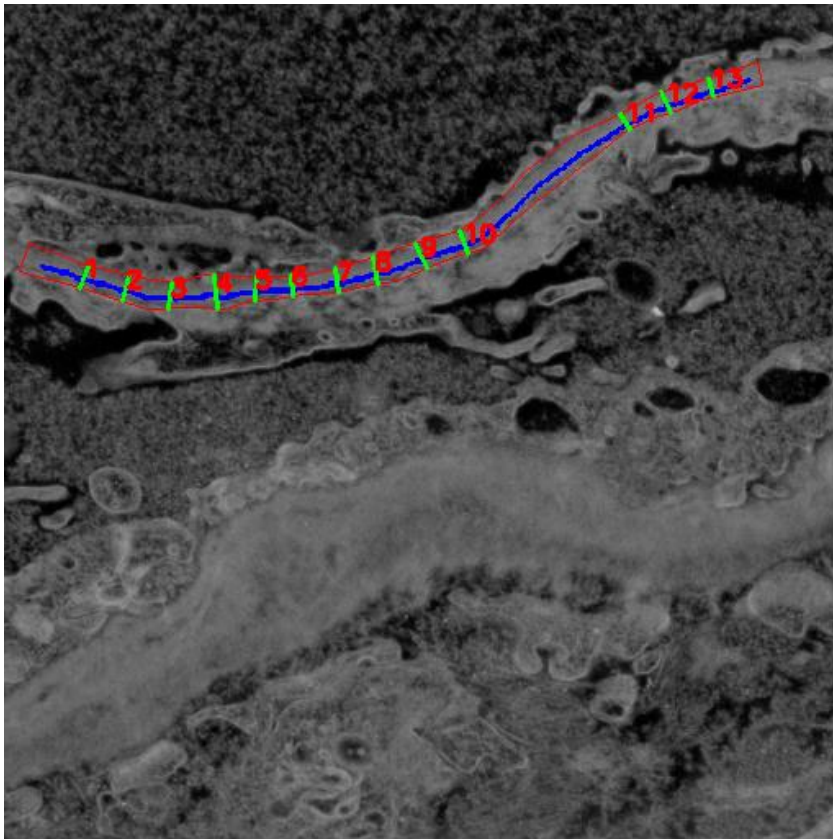


Figure 5.22.2 Γραμμές πάνω στο κομμάτι

Line n°	Distances:
1	312.26 nm
2	372.6 nm
3	366.01 nm
4	479.16 nm
5	341.64 nm
6	296.26 nm
7	343.42 nm
8	391.22 nm
9	369.04 nm
10	333.86 nm
11	237.65 nm
12	319.33 nm
13	257.3 nm

Table 5.15.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
237.65 nm	479.16 nm	339.98 nm	61.27 nm	341.64 nm	312.26 nm	369.04 nm	56.78 nm

Table 5.15.2 Ολικές εκτιμήσεις

Sample 6:
Part 1:

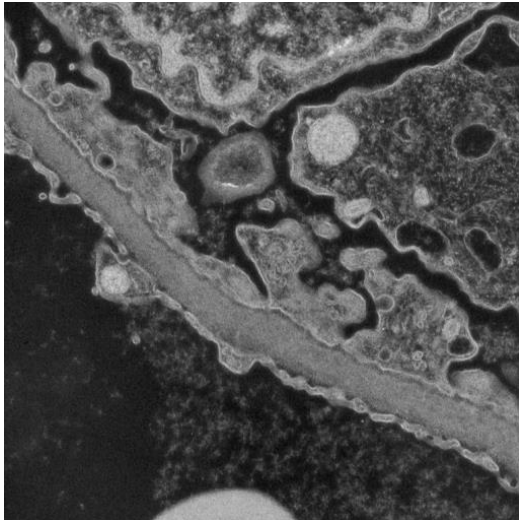


Figure 5.23.1 Κομμάτι της μεμβράνης

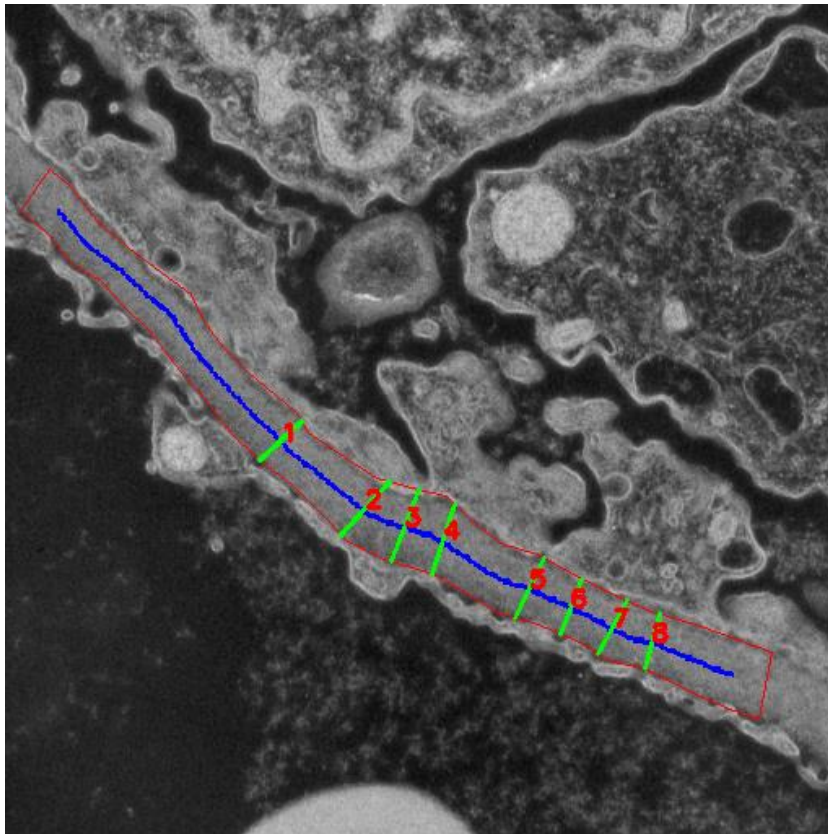


Figure 5.23.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	329.43 nm
2	419.57 nm
3	460.34 nm
4	442.97 nm
5	398.83 nm
6	355.33 nm
7	362.83 nm
8	345.65 nm

Table 5.16.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
329.43 nm	460.34 nm	389.37 nm	48.224 nm	380.83 nm	350.49 nm	431.27 nm	80.78 nm

Table 5.16.2 Ολικές εκτιμήσεις

Sample 6:

Part 2:

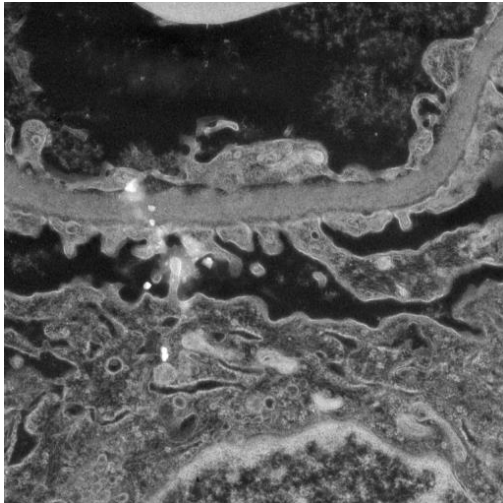


Figure 5.24.1 Κομμάτι της μεμβράνης

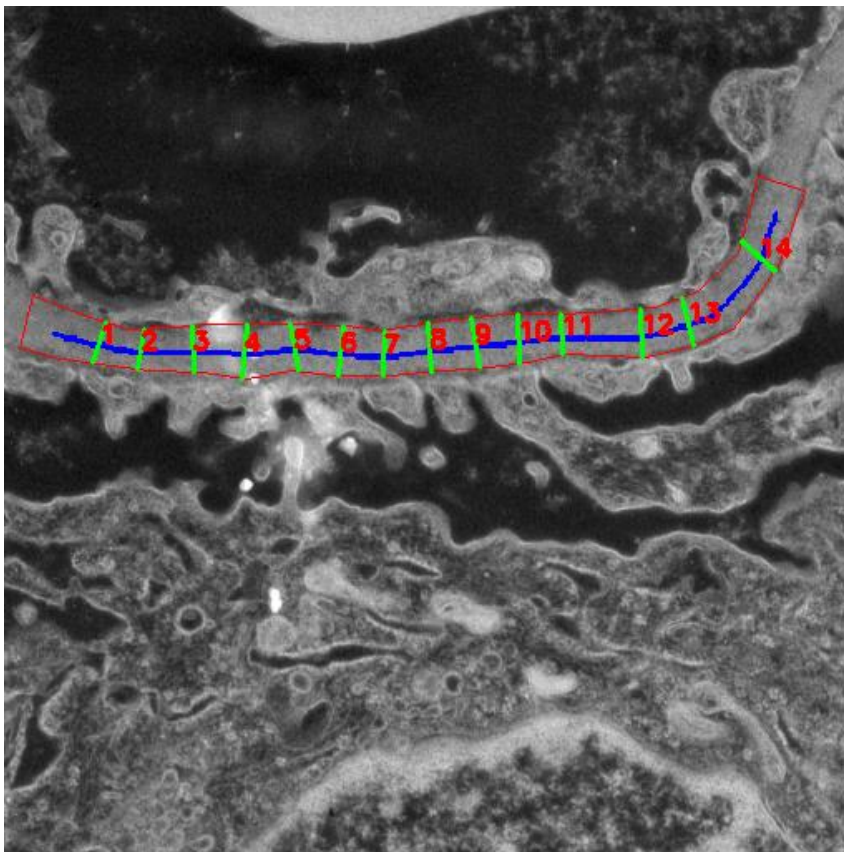


Figure 5.24.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	276.65 nm
2	268.18 nm
3	309.66 nm
4	365.96 nm
5	313.37 nm
6	332.92 nm
7	298.62 nm
8	332.2 nm
9	335.22 nm
10	331.76 nm
11	265.47 nm
12	320.71 nm
13	327.83 nm
14	279.37 nm

Table 5.17.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
265.47 nm	365.96 nm	311.28 nm	29.861 nm	317.04 nm	288.995 nm	331.98 nm	42.985 nm

Table 5.17.2 Ολικές εκτιμήσεις

Sample 7:

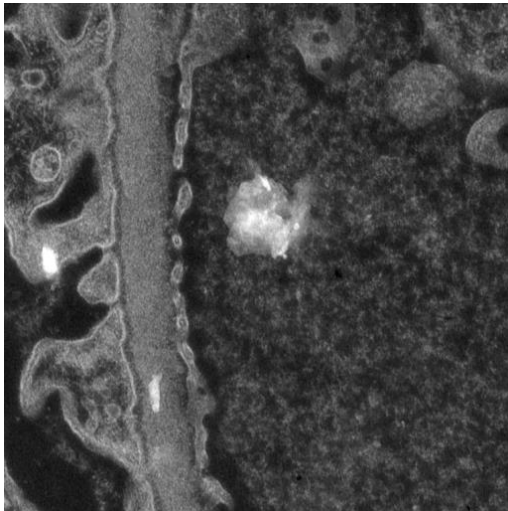


Figure 5.25.1 Κομμάτι της μεμβράνης

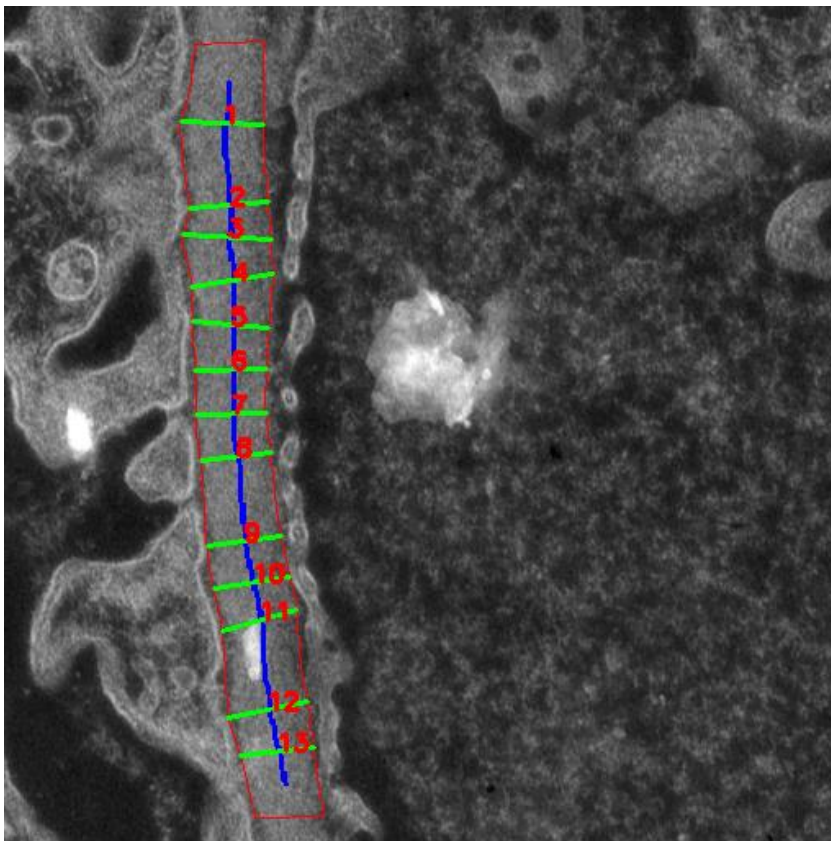


Figure 5.25.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	262.5 nm
2	252.85 nm
3	283.77 nm
4	266.47 nm
5	242.41 nm
6	230.84 nm
7	225.6 nm
8	227.45 nm
9	238.66 nm
10	244.78 nm
11	248.21 nm
12	263.7 nm
13	237.86 nm

Table 5.18.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
225.6 nm	283.77 nm	248.08 nm	17.202 nm	244.78 nm	237.86 nm	262.5 nm	24.64 nm

Table 5.18.2 Ολικές εκτιμήσεις

Sample 8:
Part 1:

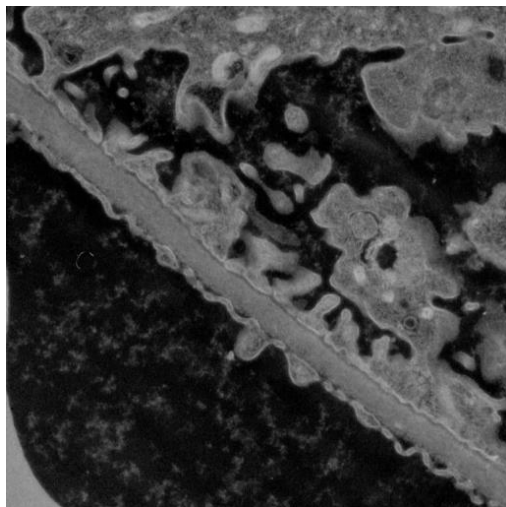


Figure 5.26.1 Κομμάτι της μεμβράνης

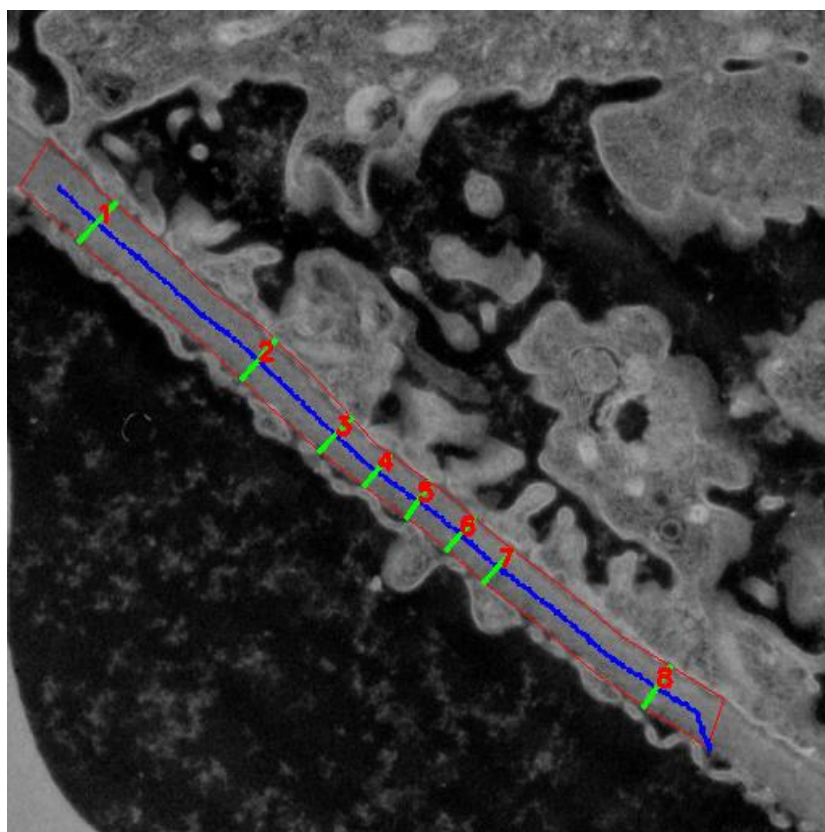


Figure 5.26.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	409.01 nm
2	394.39 nm
3	356.94 nm
4	307.83 nm
5	333.57 nm
6	325.13 nm
7	332.24 nm
8	378.78 nm

Table 5.19.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
307.83 nm	409.01 nm	354.74 nm	36.119 nm	345.255 nm	328.685 nm	386.585 nm	57.9 nm

Table 5.19.2 Ολικές εκτιμήσεις

Sample 8:
Part 2:

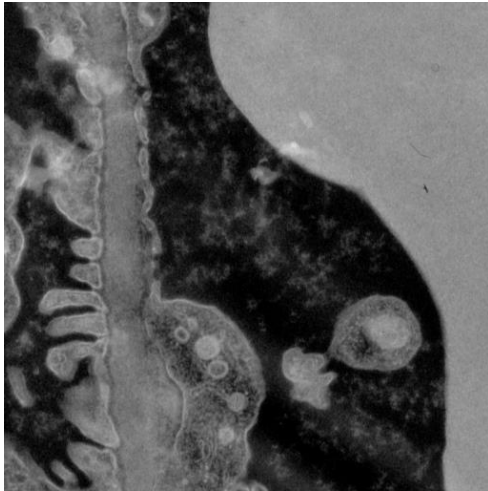


Figure 5.27.1 Κομμάτι της μεμβράνης

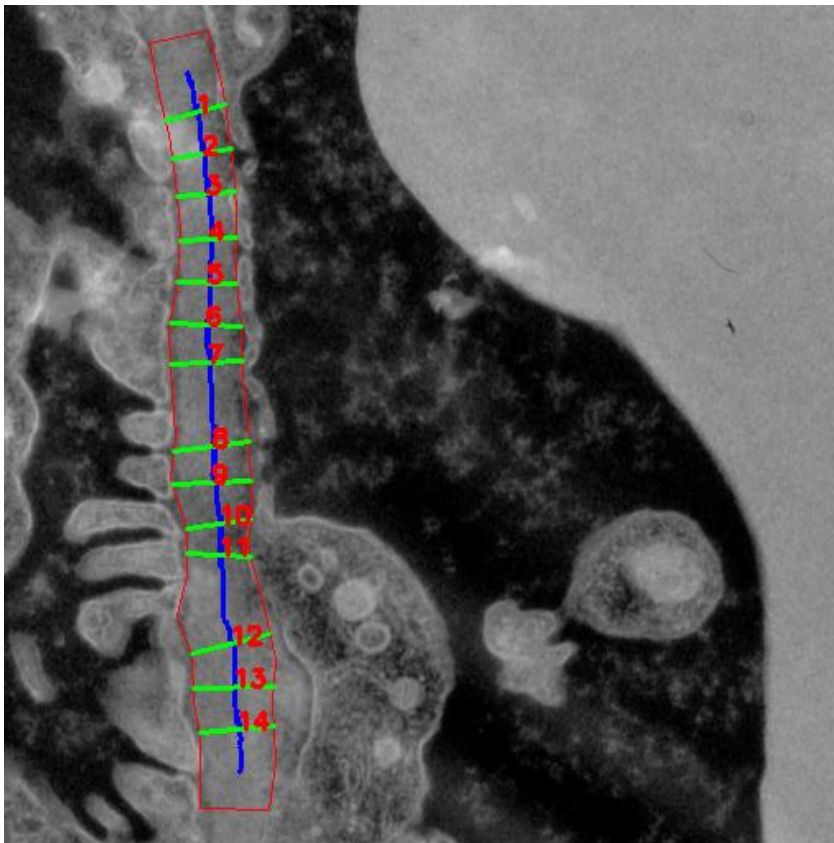


Figure 5.27.2 Γραμμές πάνω στο κομμάτι

Line n°	Distances:
1	309.53 nm
2	300.64 nm
3	288.64 nm
4	279.72 nm
5	295.67 nm
6	353.47 nm
7	361.67 nm
8	381.64 nm
9	394.47 nm
10	324.88 nm
11	320.68 nm
12	401.33 nm
13	402.35 nm
14	371.24 nm

Table 5.20.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
279.72 nm	402.35 nm	341.85 nm	44.067 nm	339.175 nm	305.085 nm	376.44 nm	71.355 nm

Table 5.20.2 Ολικές εκτιμήσεις

Sample 9:

Part 1:

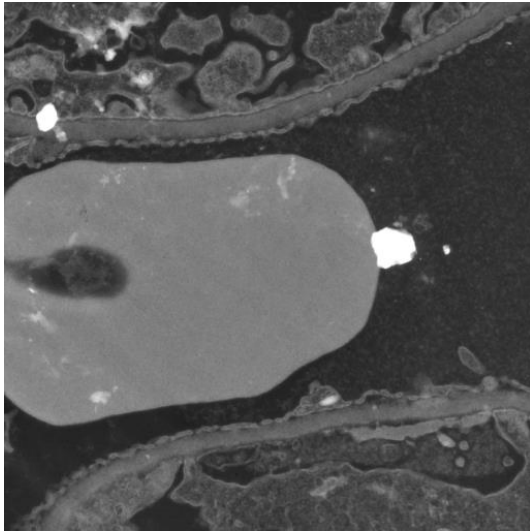


Figure 5.28.1 Κομμάτι της μεμβράνης

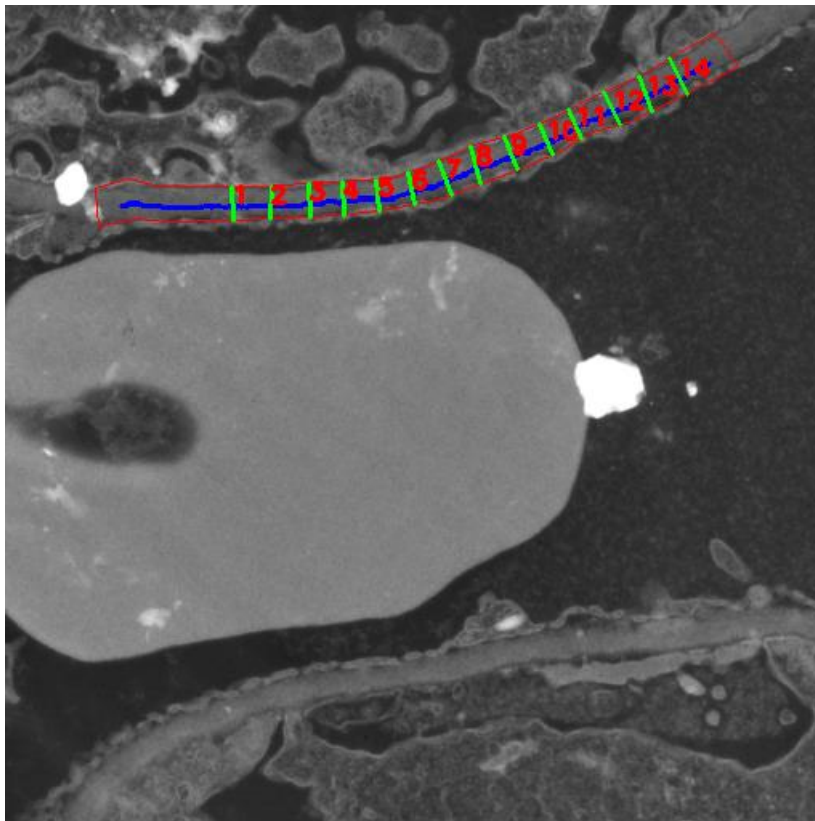


Figure 5.28.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	372.32 nm
2	354.62 nm
3	372.32 nm
4	372.32 nm
5	338.1 nm
6	360.13 nm
7	393.87 nm
8	418.81 nm
9	377.19 nm
10	360.61 nm
11	344.13 nm
12	389.5 nm
13	410.64 nm
14	428.45 nm

Table 5.21.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
338.1 nm	428.45 nm	378.07 nm	27.267 nm	372.32 nm	360.37 nm	391.685 nm	31.315 nm

Table 5.21.2 Ολικές εκτιμήσεις

Sample 9:

Part 2:

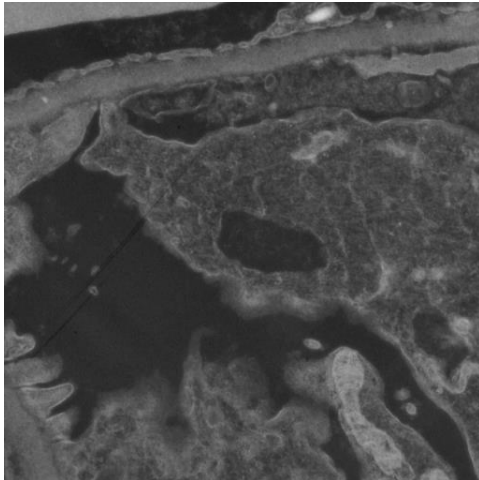


Figure 5.29.1 Κομμάτι της μεμβράνης

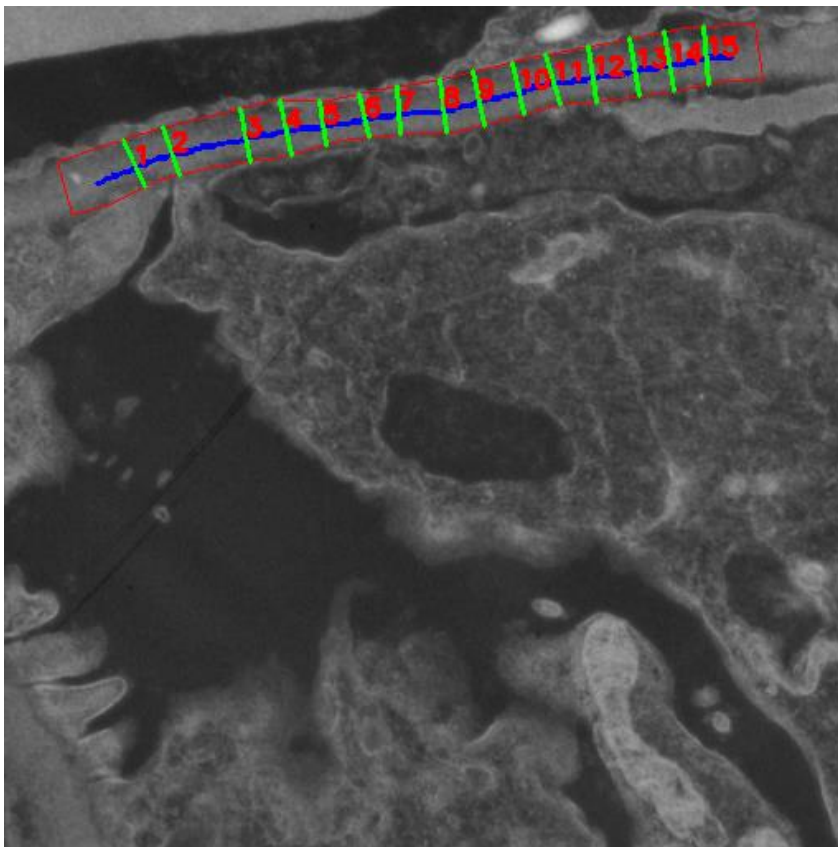


Figure 5.29.2 Γραμμές πάνω στο κομμάτι

Line n°	Distances:
1	321.24 nm
2	321.13 nm
3	342.88 nm
4	355.01 nm
5	287.75 nm
6	280.63 nm
7	306.88 nm
8	340.15 nm
9	355.01 nm
10	334.3 nm
11	328.53 nm
12	362.31 nm
13	374.15 nm
14	384.59 nm
15	370.77 nm

Table 5.22.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
280.63 nm	384.59 nm	337.69 nm	30.708 nm	340.15 nm	321.185 nm	358.66 nm	37.475 nm

Table 5.22.2 Ολικές εκτιμήσεις

Sample 10:

Part 1:

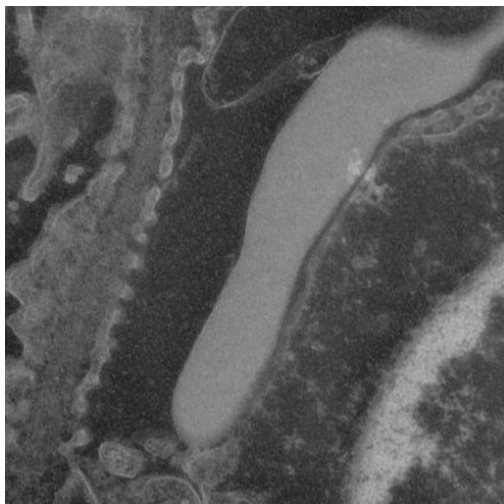


Figure 5.30.1 Κομμάτι της μεμβράνης



Figure 5.30.2 Γραμμές πάνω στο κομμάτι

Line n°	Distances:
1	143.9 nm
2	150.37 nm
3	166.85 nm
4	135.82 nm
5	157.67 nm
6	160.75 nm
7	170.55 nm
8	220.72 nm
9	223.74 nm
10	238.36 nm
11	240.86 nm
12	226.08 nm

Table 5.23.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
135.82 nm	240.86 nm	186.31 nm	39.991 nm	168.7 nm	154.02 nm	224.91 nm	70.89 nm

Table 5.23.2 Ολικές εκτιμήσεις

Sample 10:
Part 2:

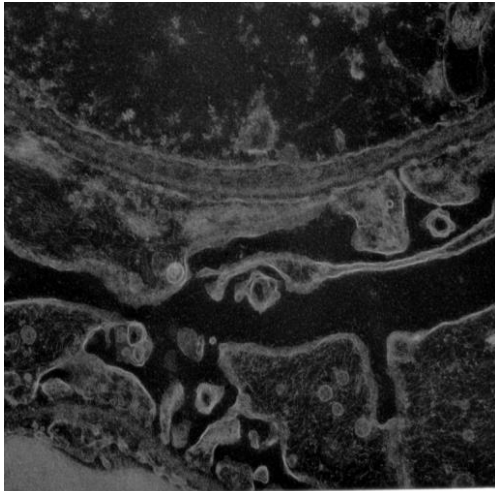


Figure 5.31.1 Κομμάτι της μεμβράνης

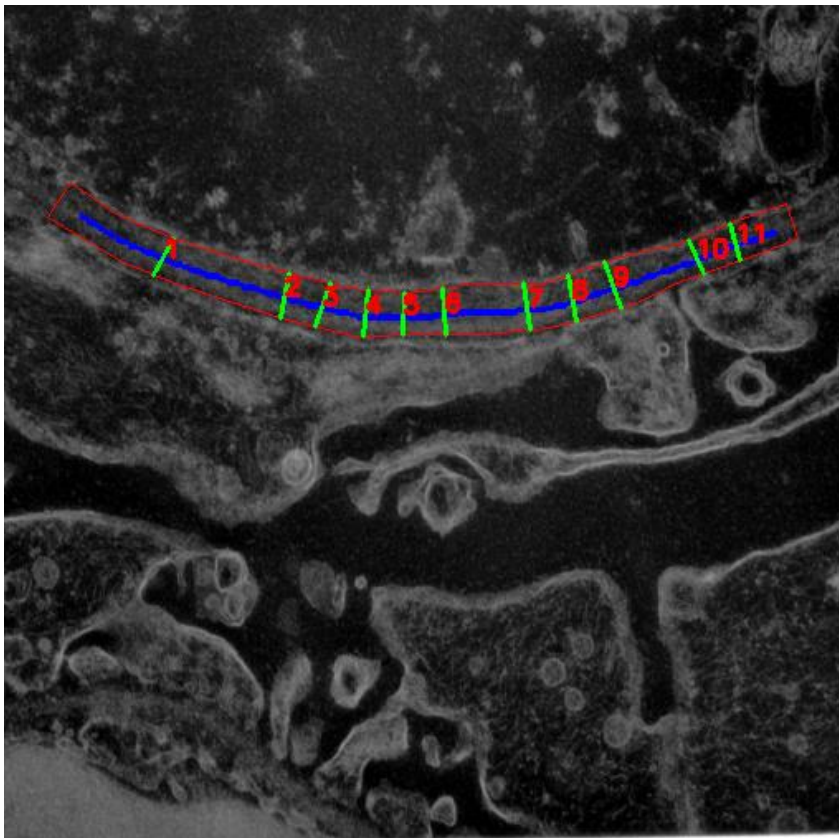


Figure 5.31.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	227.21 nm
2	252.43 nm
3	251.63 nm
4	259.27 nm
5	230.52 nm
6	258.57 nm
7	260.26 nm
8	261.52 nm
9	271.8 nm
10	199.71 nm
11	209.11 nm

Table 5.24.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
199.71 nm	271.8 nm	243.82 nm	23.582 nm	252.43 nm	228.865 nm	259.765 nm	30.9 nm

Table 5.24.2 Ολικές εκτιμήσεις

Sample 11:
Part 1:

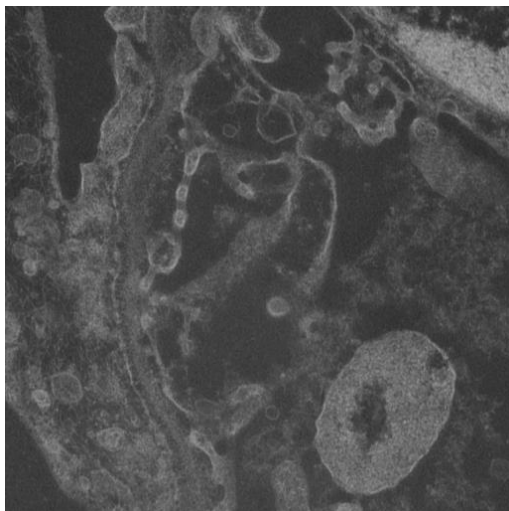


Figure 5.32.1 Κομμάτι της μεμβράνης

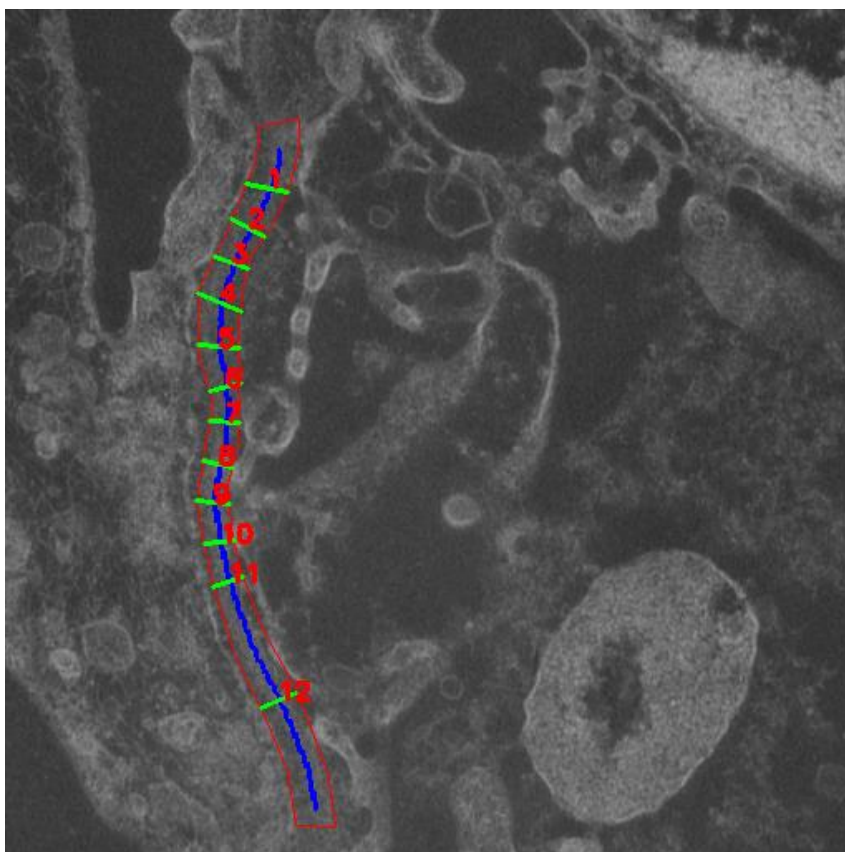


Figure 5.32.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	132.68 nm
2	120.66 nm
3	110.91 nm
4	148.28 nm
5	130.18 nm
6	102.46 nm
7	93.55 nm
8	96.03 nm
9	104.36 nm
10	94.05 nm
11	99.24 nm
12	120.03 nm

Table 5.25.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
93.55 nm	148.28 nm	112.7 nm	17.682 nm	107.63 nm	97.635 nm	125.42 nm	27.785 nm

Table 5.25.2 Ολικές εκτιμήσεις

Sample 11:
Part 2:

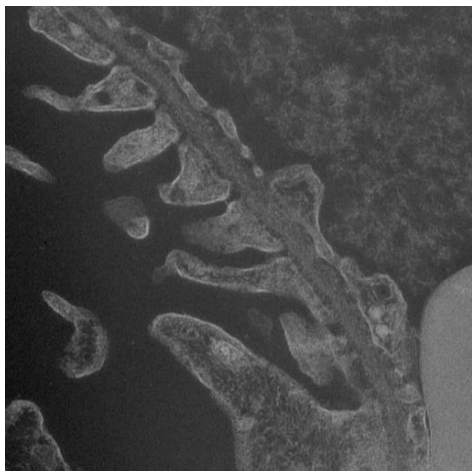


Figure 5.33.1 Κομμάτι της μεμβράνης

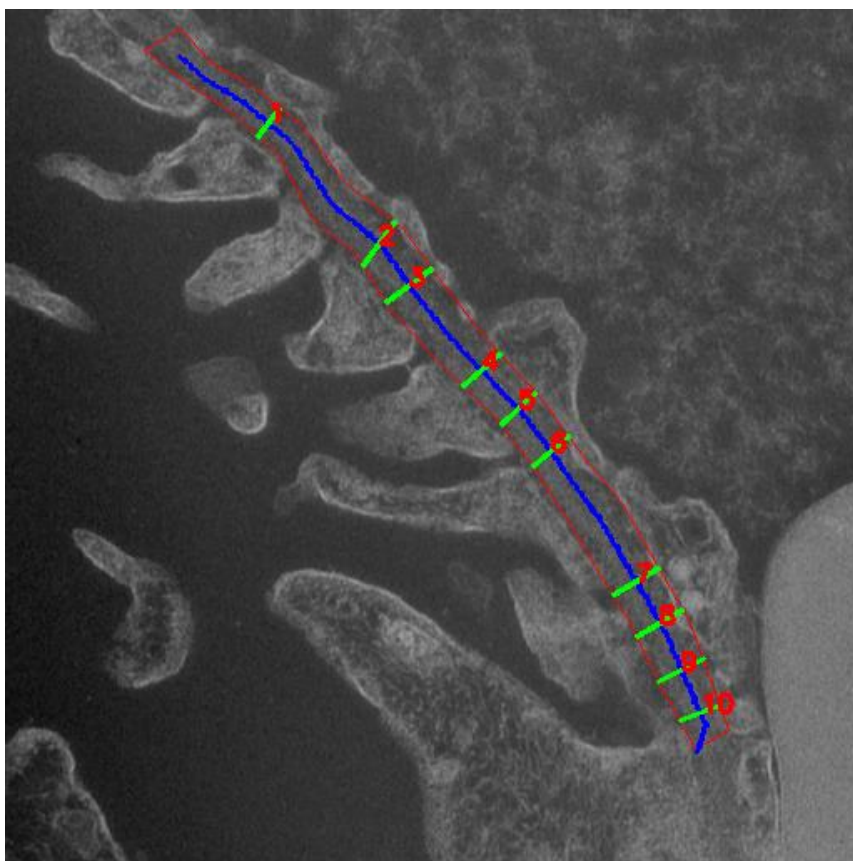


Figure 5.33.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	116.65 nm
2	175.8 nm
3	179.33 nm
4	156.02 nm
5	148.6 nm
6	148.41 nm
7	166.26 nm
8	166.26 nm
9	160.51 nm
10	134.32 nm

Table 5.26.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
116.65 nm	179.33 nm	155.22 nm	19.125 nm	158.265 nm	148.505 nm	166.26 nm	17.755 nm

Table 5.26.2 Ολικές εκτιμήσεις

Sample 12:

Part 1:

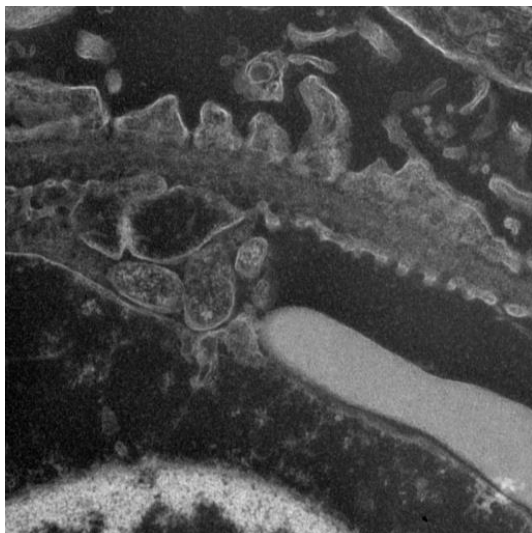


Figure 5.34.1 Κομμάτι της μεμβράνης

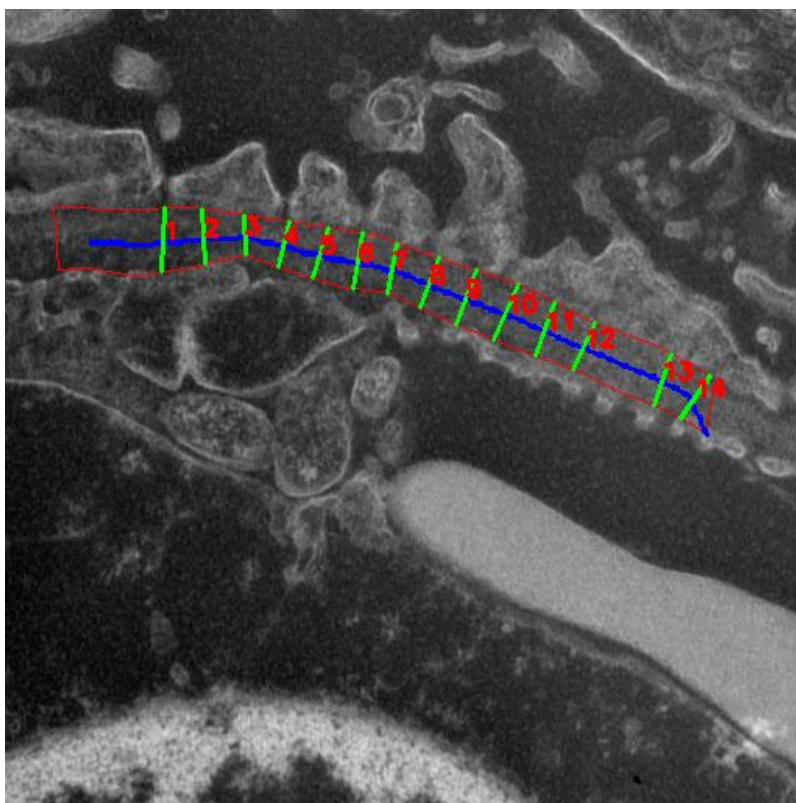


Figure 5.34.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	266.9 nm
2	227.93 nm
3	162.69 nm
4	193.19 nm
5	216.54 nm
6	236.85 nm
7	211.18 nm
8	220.15 nm
9	240.55 nm
10	252.95 nm
11	232.48 nm
12	212.33 nm
13	224.48 nm
14	209.2 nm

Table 5.27.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
162.69 nm	266.9 nm	221.96 nm	25.488 nm	222.315 nm	211.755 nm	234.665 nm	22.91 nm

Table 5.27.2 Ολικές εκτιμήσεις

Sample 12:
Part 2:

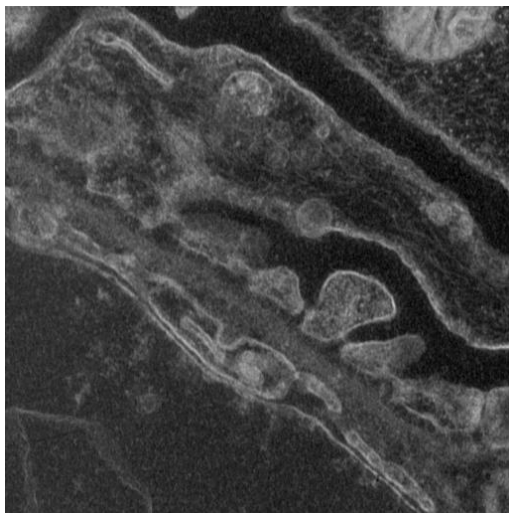


Figure 5.35.1 Κομμάτι της μεμβράνης

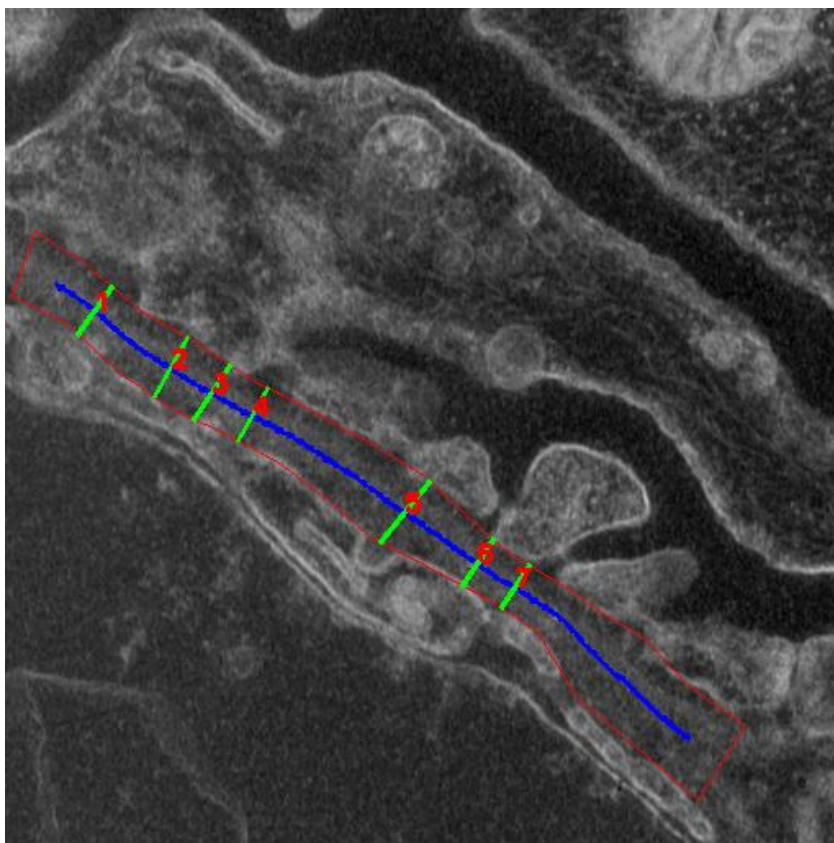


Figure 5.35.2 Γραμμές πάνω στο κομμάτι

Line n ^o	Distances:
1	164.05 nm
2	190.32 nm
3	182.25 nm
4	169.62 nm
5	217.32 nm
6	159.41 nm
7	142.84 nm

Table 5.28.1 Μετρήσεις κάθε γραμμής

Min	Max	Average	Std. Dev.	Median	1 st Quartile	3 rd Quartile	Interquartile
142.84 nm	217.32 nm	175.12 nm	24.167 nm	169.62 nm	161.73 nm	186.285 nm	24.555 nm

Table 5.28.2 Ολικές εκτιμήσεις

5.6 Εκτίμηση σφάλματος με υπάρχοντες μετρήσεις

Για σκοπούς εκτίμησης της ακρίβειας των μετρήσεών μας, έπρεπε να συγκρίνω τις μετρήσεις μου με μετρήσεις του δόκτωρ Κυριάκου. Έτσι μου έστειλε κάποια δείγματα τα οποία είχαν πάνω τις μετρήσεις των γιατρών και έτσι εγώ πήρα τις δικές μου μετρήσεις και τις σύγκρινα με εκείνες.

Sample 1

Μετρήσεις δόκτωρ Κυριάκου
173.87 nm
145.75 nm
145.75 nm

Table 5.29.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 155.12 nm

Line	Δικές μας μετρήσεις
1	144.93 nm
2	149.91 nm
3	134.45 nm
4	121.98 nm
5	105.51 nm
6	129.75 nm
7	125.04 nm
8	137.14 nm
9	149.39 nm
10	140.67 nm
11	153.78 nm
12	164.42 nm
13	175.93 nm
14	179.77 nm
15	176.26 nm
16	183.64 nm

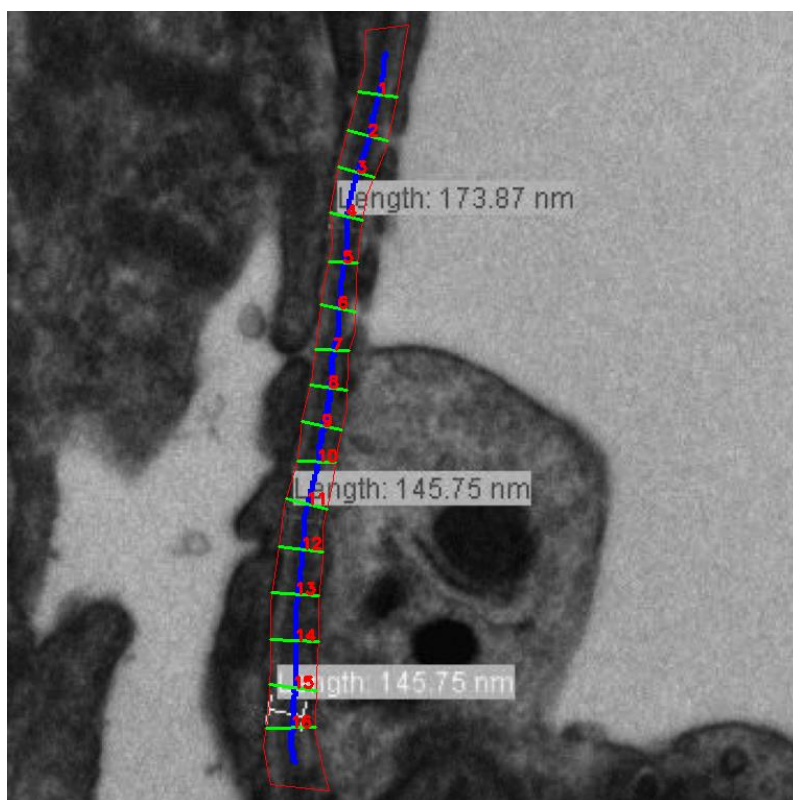


Figure 5.36 Γραμμές πάνω στο κομμάτι

Table 5.29.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 148.29 nm

Average Difference: 6.83 nm

Average Error: 4.4%

Sample 2

Μετρήσεις δόκτωρ Κυριάκου
319.44 nm
311.53 nm
335.66 nm

Table 5.30.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 322.21 nm

Line	Δικές μας μετρήσεις
1	308.92 nm
2	305.04 nm
3	324.87 nm
4	350.06 nm
5	327.3 nm
6	331.61 nm
7	306.3 nm
8	290.99 nm
9	291.45 nm
10	316.5 nm
11	315.25 nm

Table 5.30.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 315.3 nm

Average Difference: 6.91 nm

Average Error: 2.1%

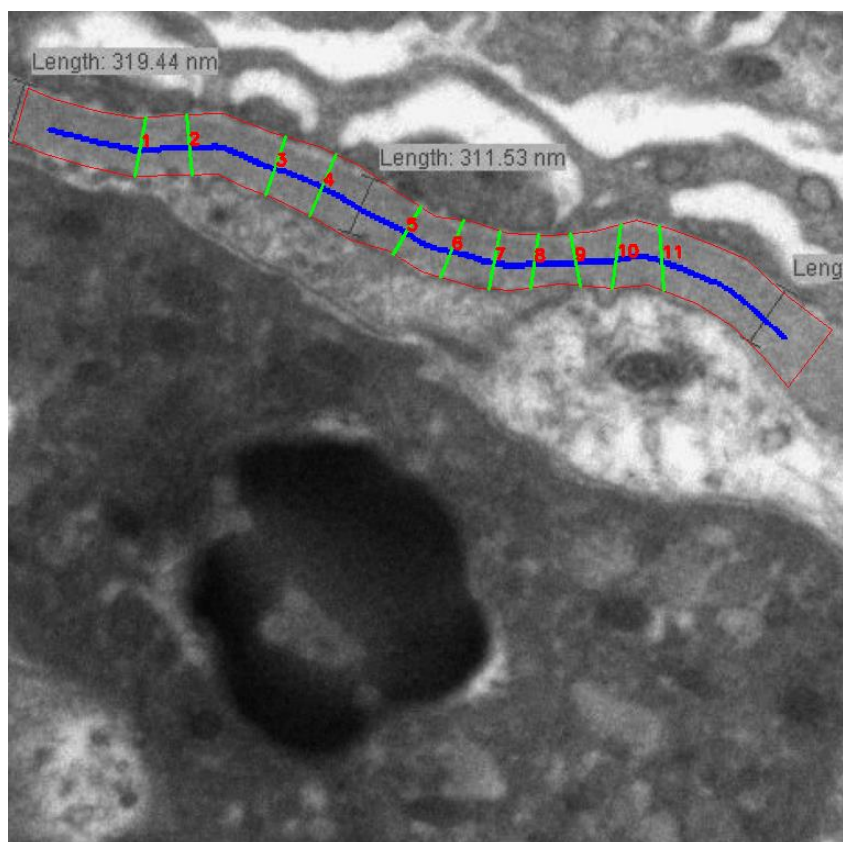


Figure 5.37 Γραμμές πάνω στο κομμάτι

Sample 3

Μετρήσεις δόκτωρ Κυριάκου
327.44 nm
391.12 nm

Table 5.31.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 359.28 nm

Line	Δικές μας μετρήσεις
1	343.86 nm
2	366.82 nm
3	379.29 nm
4	370.01 nm
5	369.3 nm
6	358.93 nm
7	349.71 nm
8	312.25 nm
9	312.25 nm
10	331.18 nm
11	315.52 nm
12	320.95 nm
13	327.33 nm
14	339.21 nm
15	352.48 nm
16	347.25 nm
17	337.38 nm
18	333.76 nm

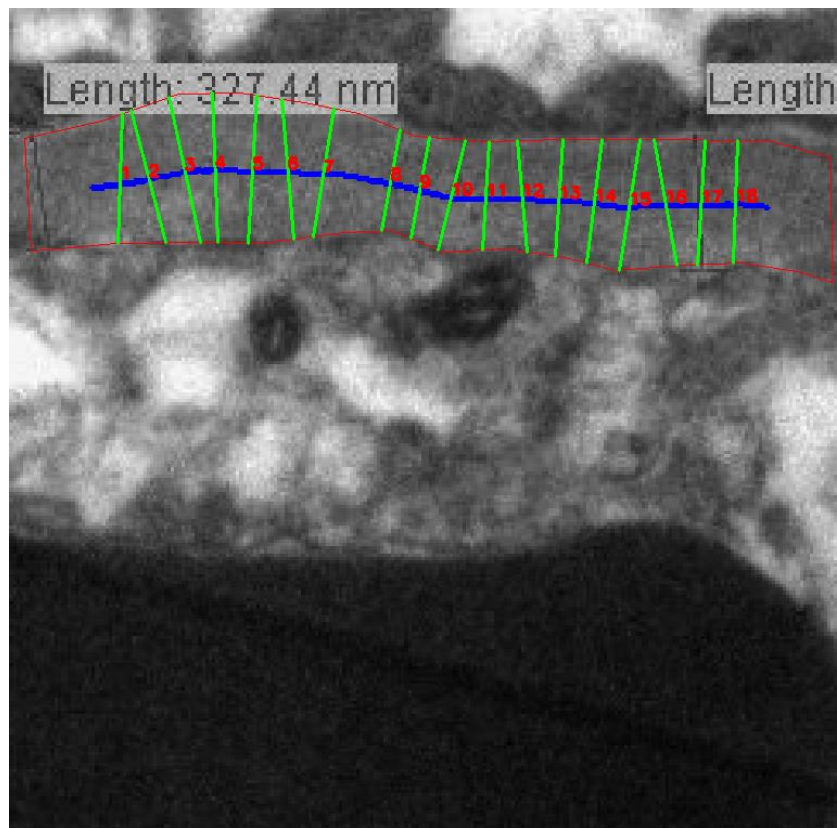


Figure 5.38 Γραμμές πάνω στο κομμάτι

Table 5.31.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 342.64 nm

Average Difference: 16.64 nm

Average Error: 4.6%

Sample 4

Μετρήσεις δόκτωρ Κυριάκου
325.52 nm
320.15 nm
306.24 nm

Table 5.32.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 317.30 nm

Line	Δικές μας μετρήσεις
1	304.84 nm
2	296.21 nm
3	247.13 nm
4	296.69 nm
5	307.44 nm
6	282.02 nm
7	330.04 nm
8	316.66 nm
9	317.71 nm
10	338.65 nm

Table 5.32.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 303.74 nm

Average Difference: 13.56 nm

Average Error: 4.3%

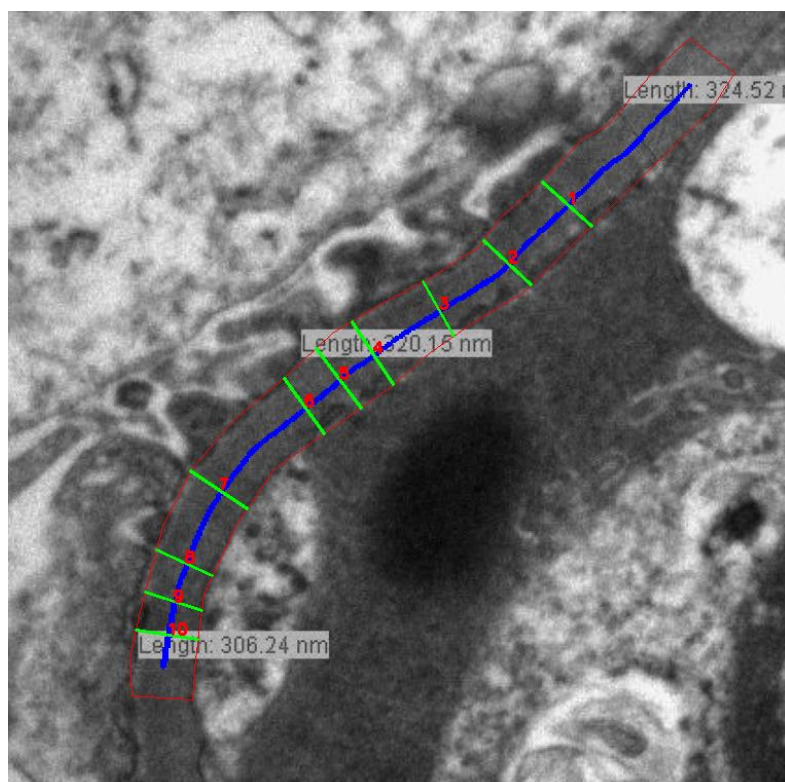


Figure 5.39 Γραμμές πάνω στο κομμάτι

Sample 5

Μετρήσεις δόκτωρ Κυριάκου
251.55 nm
301.31 nm
257.44 nm
251.55 nm

Table 5.33.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 265.45 nm

Line	Δικές μας μετρήσεις
1	304.84 nm
2	296.21 nm
3	247.13 nm
4	296.69 nm
5	307.44 nm
6	282.02 nm
7	330.04 nm
8	316.66 nm

Table 5.33.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 260.34 nm

Average Difference: 5.11 nm

Average Error: 1.9%

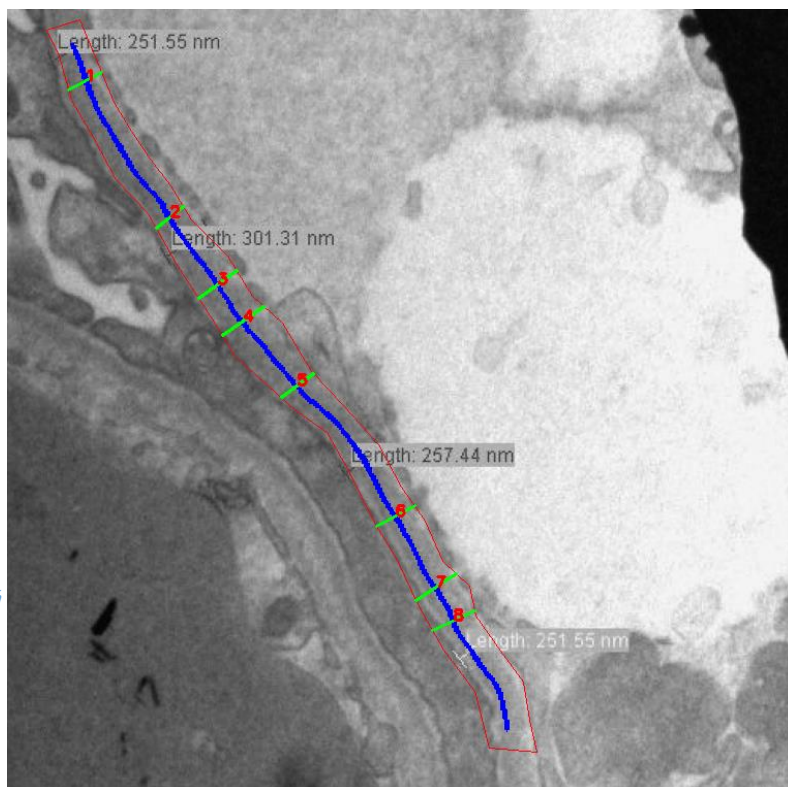


Figure 5.40 Γραμμές πάνω στο κομμάτι

Sample 6

Μετρήσεις δόκτωρ Κυριάκου
157.72 nm
166.44 nm
183.28 nm

Table 5.34.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 169.15 nm

Line	Δικές μας μετρήσεις
1	193.61 nm
2	188.75 nm
3	176.34 nm
4	149.49 nm
5	124.47 nm
6	127.64 nm
7	158.27 nm
8	198.97 nm
9	213.22 nm
10	211.99 nm
11	183.21 nm
12	192.82 nm
13	195.09 nm

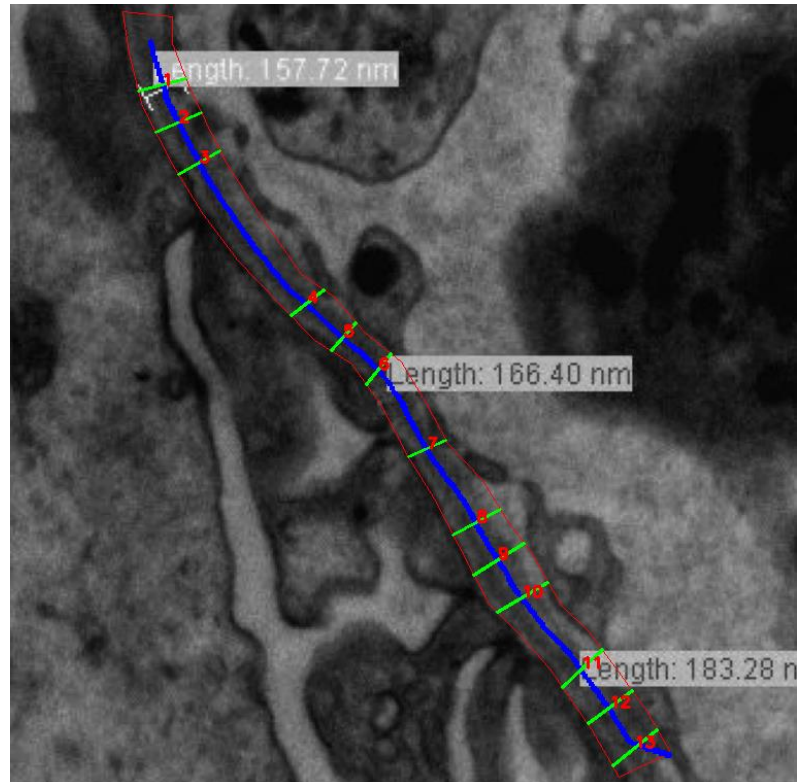


Table 5.34.2 Δικές μας μετρήσεις για κάθε γραμμή Figure 5.41 Γραμμές πάνω στο κομμάτι

Average: 177.99 nm

Average Difference: 8.84 nm

Average Error: 5.2%

Sample 7

Μετρήσεις δόκτωρ Κυριάκου
191.51 nm
202.12 nm

Table 5.35.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 196.82 nm

Line	Δικές μας μετρήσεις
1	205.19 nm
2	183.39 nm
3	169.97 nm
4	174.79 nm
5	196.29 nm
6	214.77 nm
7	230.09 nm
8	240.18 nm
9	231.3 nm
10	224.4 nm
11	210.4 nm
12	203.25 nm
13	206.72 nm
14	214.21 nm
15	215.28 nm
16	206.72 nm
17	204.51 nm
18	200.4 nm

Table 5.35.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 207.33 nm

Average Difference: 10.51 nm

Average Error: 5.3%

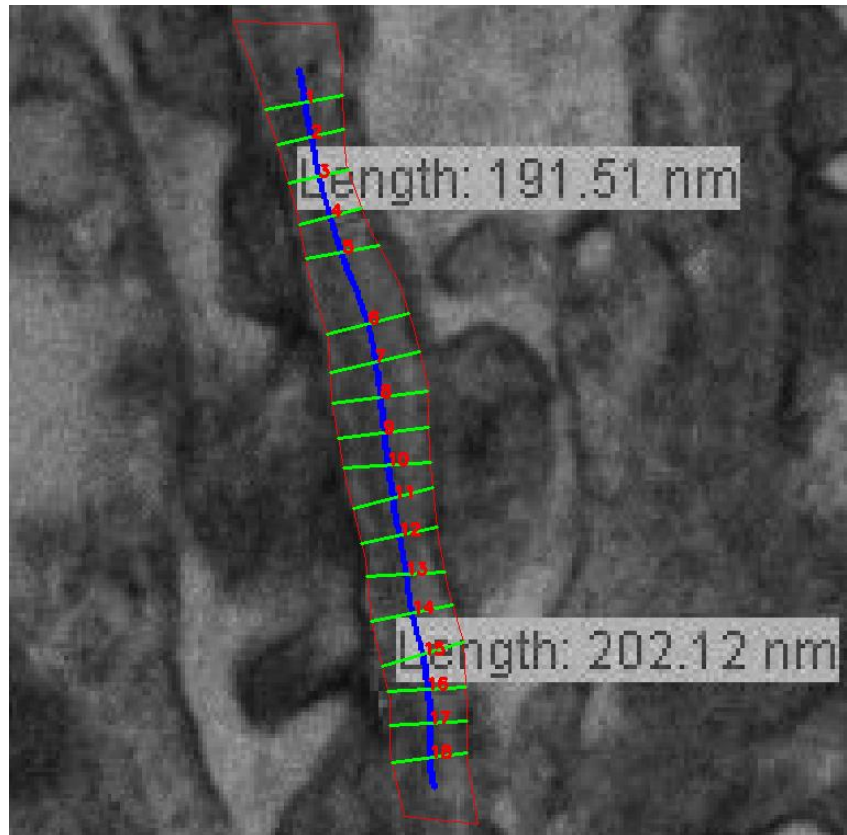


Figure 5.42 Γραμμές πάνω στο κομμάτι

Sample 8

Μετρήσεις δόκτωρ Κυριάκου
273.93 nm
200.63 nm

Table 5.36.1 Μετρήσεις του δόκτωρ Κυριάκου

Average: 237.28 nm

Line	Δικές μας μετρήσεις
1	217.98 nm
2	215.29 nm
3	223.05 nm
4	237.97 nm
5	238.14 nm
6	270.6 nm
7	249.7 nm
8	261.61 nm
9	263.34 nm
10	226.88 nm
11	229.87 nm
12	223.35 nm
13	227.35 nm
14	215.56 nm
15	201.46 nm
16	243.31 nm

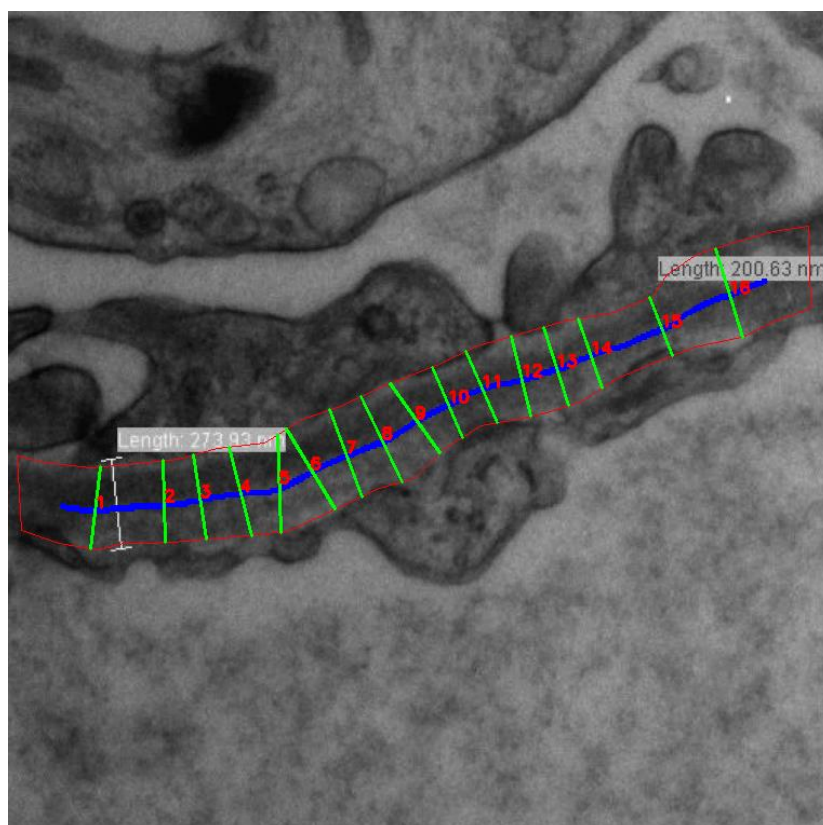


Figure 5.43 Γραμμές πάνω στο κομμάτι

Table 5.36.2 Δικές μας μετρήσεις για κάθε γραμμή

Average: 234.09 nm

Average Difference: 3.19 nm

Average Error: 1.3%

Όπως παρατηρούμε από τις πιο πάνω συγκρίσεις μας το σφάλμα των μετρήσεων μας είναι ελάχιστο, πράγμα που αποδεικνύει την ακρίβεια τους.

Κεφάλαιο 6

Segmentation με τη χρήση Unet

6.1 Τι είναι το Unet και πώς λειτουργεί	72
6.2 Δημιουργία δικού μας μοντέλου	74
6.3 Αποτελέσματα μετά το training του δικτύου	76
6.4 Ενσωμάτωση του μοντέλου στο σύστημα μας	77

6.1 Τι είναι το Unet και πώς λειτουργεί

Μετά την ανάπτυξη του προγράμματος αυτού, έκανα μια συνάντηση με τον κύριο Παττίχη για να του δείξω την πρόοδό μου και να μάθω πως επρόκειτο να προχωρίσω με την εργασία αυτή. Ο κύριος Παττίχης μου είπε πως καλό θα ήταν η έυρεση της περιοχής της μεμβράνης η οποία θα μετρηθεί να γίνεται αυτόματα, χωρίς την παρέμβαση του χρήστη και μου σύστησε να δοκιμάσω το Unet.

Το Unet είναι ένα νευρωνικό δίκτυο το οποίο ασχολείται με το Bio Medical Image Segmentation και ονομάζεται έτσι επειδή η αρχιτεκτονική του έχει σχήμα U. Η αρχιτεκτονική του δικτύου αυτού περιέχει δύο διαδρομές. Η πρώτη διαδρομή ονομάζεται contraction, ή αλλιώς encoder και χρησιμοποιείται για την καταγραφή του context της εικόνας. Ο encoder είναι απλώς μια παραδοσιακή στοίβα από convolutional layers και pooling layers. Η δεύτερη διαδρομή ονομάζεται expanding path ή αλλιώς decoder και χρησιμοποιείται για να κάνει ακριβή localization χρησιμοποιώντας μεταφερόμενα convolutions. Έτσι, είναι ένα end to end fully convolutional network (FCN) αφού περιέχει μόνο Convolutional layers και δεν περιέχει κανένα πυκνό layer και αυτό το καθιστά ικανό να μπορεί να δεχτεί εικόνα οποιουδήποτε μεγέθους. Η εικόνα στην επόμενη σελίδα αναπαριστά το Unet που έχει περιγραφεί.

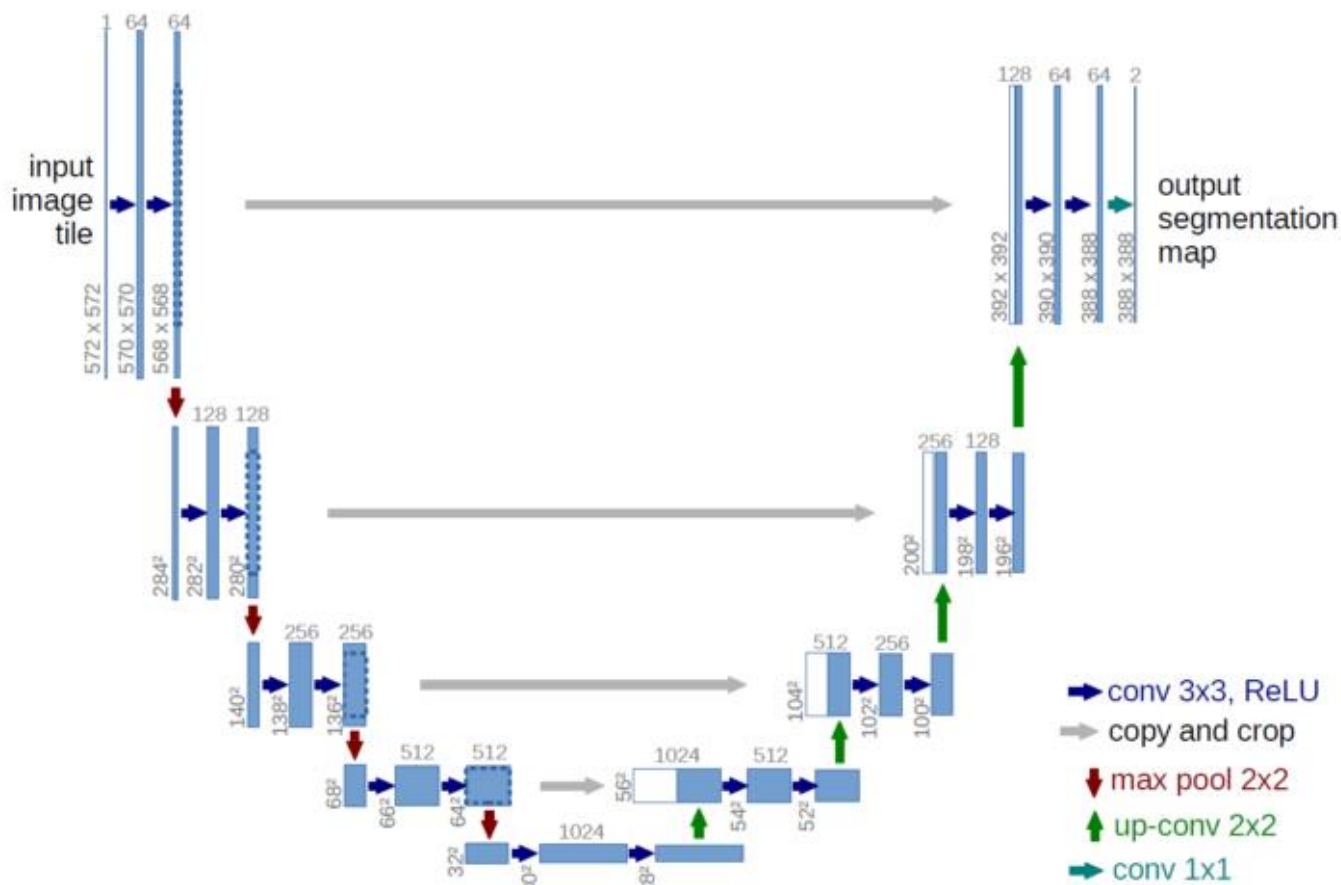


Figure 6.1 Αρχιτεκτονική μοντέλου Unet

Να αναφερθεί επίσης ότι για την δική μας περίπτωση θα χρησιμοποιήσουμε σαν μέγεθος input εικόνας 128x128x3 και όχι 572x572x3 που δείχνει το παράδειγμα.

6.2 Δημιουργία δικού μας μοντέλου

Το Unet που κάναμε implement είναι το ίδιο με το σχήμα στην εικόνα του υποκεφαλαίου 6.1 με την διαφορά να υπάρχει μόνο στα μεγέθη. Αρχικά ξεκινούμε με input εικόνες μεγέθους $128 \times 128 \times 3$, και ακολουθώντας την διαδρομή του encode το μέγεθος μετατρέπεται σε $64 \times 64 \times 16$, μετά σε μέγεθος $32 \times 32 \times 32$, $16 \times 16 \times 64$, $8 \times 8 \times 128$ και $8 \times 8 \times 256$. Ακολουθώντας κατά την διαδρομή του decode, το μέγεθος θα αυξηθεί από $8 \times 8 \times 256$ σε $16 \times 16 \times 128$, $32 \times 32 \times 64$, $64 \times 64 \times 32$, $128 \times 128 \times 16$ και το τελικό μας output είναι $128 \times 128 \times 1$. Δημιούργησα επίσης, δύο φακέλους οι οποίοι περιέχουν, ο ένας τα training data και ο άλλος τα testing data. Για να παράξω αυτά τα δεδομένα, χρειάστηκε να αλλάξω τον τρόπο με τον οποίο αποθήκευα το αποκομμένο κομμάτι από το crop του χρήστη στο πρόγραμμά μου. Το κομμάτι αυτό έπρεπε να ήταν τετράγωνο για να μπορώ αργότερα να αλλάξω το μέγεθός του και να το κάνω 128×128 . Έτσι απλά όταν ο χρήστης επέλεγε την περιοχή του film που θέλει να απομονώσει χρησιμοποιώντας την πιο μακριά πλευρά του rectangle που χρησιμοποιούσε και εξίσωνα κάθε πλευρά με εκείνη, έτσι έπαιρνα τετραγωνικό κομμάτι αλλά επίσης και μεγαλύτερο από αυτό που επέλεγε ο χρήστης. Εκτέλεσα το πρόγραμμα αρκετές φορές με σκοπό να αποθηκεύσω όσο το δυνατόν περισσότερα κομμάτια μπορούσα για να χρησιμοποιήσω τόσο για το train set, όσο και για το test set. Δυστυχώς ήταν αδύνατη η συλλογή μεγάλου αριθμού κομματιών που να ήταν ευδιάκριτα γιατί τα περισσότερα films που είχα στην κατοχή μου ήταν είτε σκούρα είτε δεν φαινόταν καλά τα σημεία στα οποία διαχωρίζεται η μεμβράνη από τα εξωγόμενα. Για αυτό το λόγο τα δείγματα που είχα για να δώσω στο νευρωνικό δίκτυο ήταν 14, από τα οποία τα 9 άνηκαν στο train set και τα υπόλοιπα 5 στο test set.

Για να μπορώ όμως να κάνω train το δίκτυο, χρειαζόμουν και κάποιο είδος μάσκας για το κάθε δείγμα ούτως ώστε το Unet να μπορεί να έχει ένα expected output. Έτσι, μετά τον διαχωρισμό των κομματιών αυτών από το film τους, επέλεγα ένα ένα στα films την περιοχή της μεμβράνης που ήθελα να κάνω segment, με το χειροκίνητο τρόπο που ανέφερα στο υποκεφάλαιο 5.1 και απόκοπτα το κομμάτι εκείνο από το δείγμα. Ακολουθώντας δημιούργησα μια άλλη εικόνα η οποία θα είχε μόνο τα χρώματα άσπρο και μαύρο και γέμισα με άσπρο χρώμα τα pixels στα οποία είχαμε το αποκομμένο κομμάτι. Μετά έκανα resize και τις δύο εικόνες, δηλαδή του δείγματος και της μάσκας του, σε διαστάσεις 128×128 . Συνέχισα με τον ίδιο τρόπο για όλα τα 9 δείγματα που θα απάρτιζαν το train set του δικτύου και μερικά από τα αποτελέσματα αυτής της διαδικασίας φαίνονται πιο κάτω.

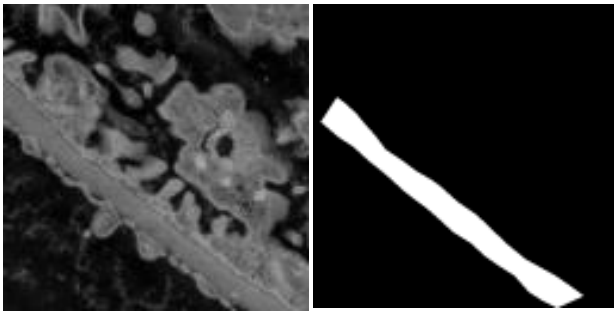


Figure 6.2 Δείγμα μαζί με την μάσκα του

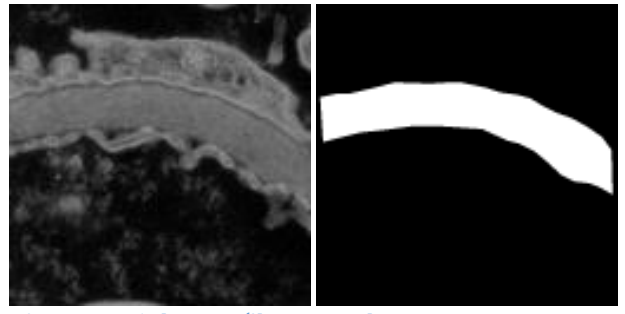


Figure 6.3 Δείγμα μαζί με την μάσκα του

Στα αριστερά φαίνονται τα δείγματα και στα δεξιά τους οι μάσκες τους οι οποίες θα δίνουν στο Unet ένα expected output για το κάθε ένα τους. Αυτά τα δύο δείγματα ανήκουν στο train set.

Μετά την δημιουργία των sets αυτών, τα χώρισα σε φακελούς ούτως ώστε το πρόγραμμα να μπορεί να διαβάζει όλες τις εικόνες μαζί με τις μάσκες τους και να διαχωρίζει τα δεδομένα σε training και testing.

Αφού τα δείγματα που μπορούσαμε να δώσουμε μέσα στο δίκτυο για να εκπαιδευτεί ήταν πάρα πολύ λίγα, μετά από καθοδήγηση του κύριου Παττίχη χρησιμοποίησα τη μέθοδο του data augmentation για το training set μου και με τη χρήση των υπάρχοντων δειγμάτων παράξα 735 δείγματα. Αυτά τα δείγματα ήταν βασικά διάφορες παραλλαγές των 9 δειγμάτων που είχα αρχικά για την εκπαίδευση του δικτύου. Αυτές οι παραλλαγές δημιουργούνταν με περιστροφές των δειγμάτων, αποκοπές, μεγενθύνσεις και αντανakλάσεις.

Μετά την υλοποίηση του δικτύου με τον τρόπο που ανέφερα στην αρχή του υποκεφαλαίου, έπρεπε να εκπαιδευτεί το μοντέλο και να αποθηκευτεί. Αυτό επιτεύχθηκε με την βοήθεια της εντολής `model.save('model_for_segmentation.h5')` και εκπαιδεύτηκε με την εντολή `results = model.fit( X_train, Y_train, validation_split=0.1, batch_size=16, epochs=50)`.

Η πρώτη εντολή ορίζει το path όπου θα αποθηκευτεί το μοντέλο και το αποθηκεύει και η δεύτερη εντολή, χρησιμοποιεί το train set και εκτελεί την εκπαίδευση για 50 εποχές. Το X_train είναι τα δείγματα ενώ το Y_train είναι οι μάσκες των δειγμάτων. Τέλος ακολουθούσαν οι εντολές predict με τις οποίες ελέγχουμε τα αποτελέσματα μας πάνω στο train set και στο test set.

6.3 Αποτελέσματα μετά το training του δικτύου

Σε αυτό το σημείο τρέξαμε το δίκτυό μας για τα δεδομένα που πήραμε για 50 εποχές και εμφανίζαμε κατά τη διάρκεια του training τα αποτελέσματα της κάθε εποχής.

```
Epoch 1/50
42/42 [=====] - 66s 2s/step - loss: 0.5221 - accuracy: 0.7968 - val_loss: 0.2523 - val_accuracy: 0.8950
Epoch 2/50
42/42 [=====] - 73s 2s/step - loss: 0.2573 - accuracy: 0.8643 - val_loss: 0.2099 - val_accuracy: 0.8950
Epoch 3/50
42/42 [=====] - 73s 2s/step - loss: 0.2241 - accuracy: 0.8760 - val_loss: 0.1407 - val_accuracy: 0.9381
```

Figure 6.4 Διαδικασία της εκπαίδευσης του δικτύου

Παρατηρούμε ότι με κάθε εποχή το accuracy μας βελτιώνεται. Μετά το Training του δικτύου ακολούθησαν τα αποτελέσματα των προβλέψεων που πήραμε για ένα δείγμα του train set και ένα του test set με τη χρήση της εντολής `model.predict(img, verbose = 1)`. Σε αυτό το σημείο για σκοπούς validation κάναμε load το εκπαιδευμένο μοντέλο και το ελέγξαμε πάνω σε μια εικόνα που περιεχόταν στο training και μια άλλη εικόνα εκτός του train set.

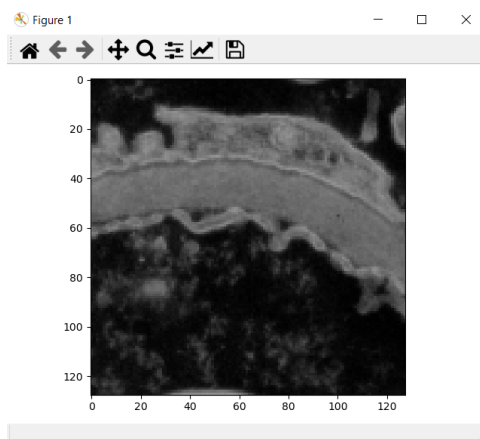


Figure 6.5.1 Παράδειγμα δείγματος από training set

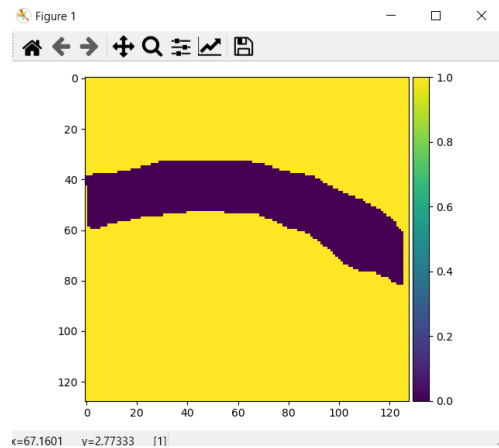


Figure 6.5.2 Η πρόβλεψη της segmented περιοχής

Σε αυτό το δείγμα του train set παρατηρούμε ότι το αποτέλεσμα του segmentation είναι πολύ σωστό

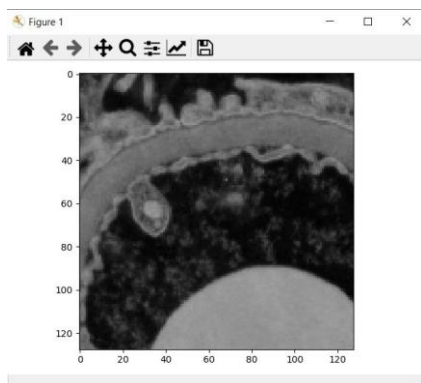


Figure 6.6.1 Παράδειγμα δείγματος εκτός του training set

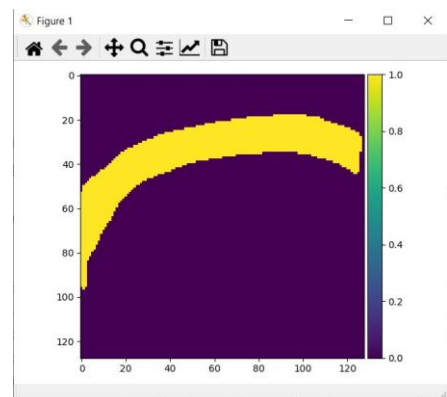


Figure 6.6.2 Η πρόβλεψη της segmented περιοχής

Το πιο πάνω αποτέλεσμα είναι η πρόβλεψη της εικόνας που δεν ανήκει στο training set. Παρατηρούμε ότι βρήκε την περιοχή με πολύ υψηλή ακρίβεια.

6.4 Ενσωμάτωση του μοντέλου στο σύστημα μας

Μετά την εκπαίδευση του δικτύου, μένει να το δοκιμάσουμε στο πρόγραμμά μας. Έτσι μετά την επιλογή του κομματιού που θέλει να εστιάσει ο χρήστης, τετραγωνίζαμε το κομμάτι χρησιμοποιώντας την μεγαλύτερη πλευρά της επιλογής του χρήστη ούτως ώστε να μην αποκόψουμε πληροφορίες που ενδεχομένως να θέλει ο χρήστης και αλλάζαμε τις διαστάσεις του κομματιού σε 128x128 για να μπορεί να το δεχτεί το Unet. Ακολούθως καλείται η μέθοδος prediction που περιέχεται στο Unet.py και επιστρέφεται πίσω η μάσκα της μεμβράνης και με βάση αυτή υπολογίζεται και το contour area όπως φαίνεται πιο κάτω.

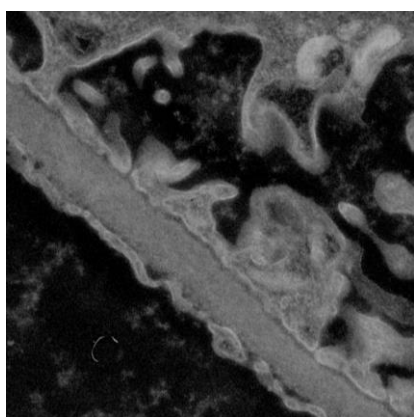


Figure 6.7.1 Αποκομμένο κομμάτι



Figure 6.7.2 Πρόβλεψη Unet

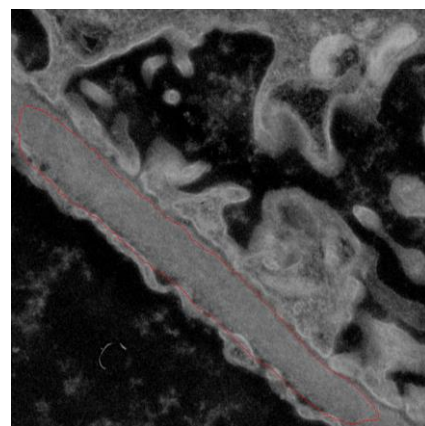
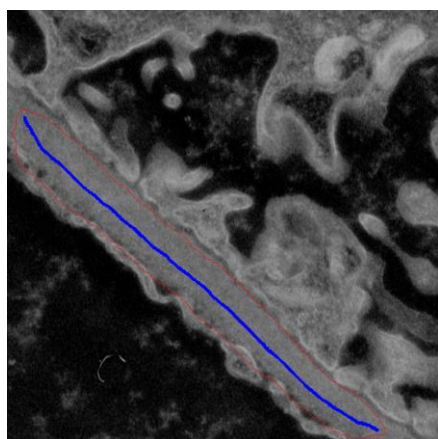


Figure 6.7.3 Contour Area

Όπως βλέπουμε το πρόγραμμα υπολογίζει με πάρα πολύ ακρίβεια το κομμάτι της μεμβράνης και ακολούθως υπολογίζεται ο σκελετός και οι κάθετες γραμμές μαζί με τα αποτελέσματα των μετρήσεων.



ομματιού

Figure 6.8.2 Κάθετες γραμμές στο σκελετό

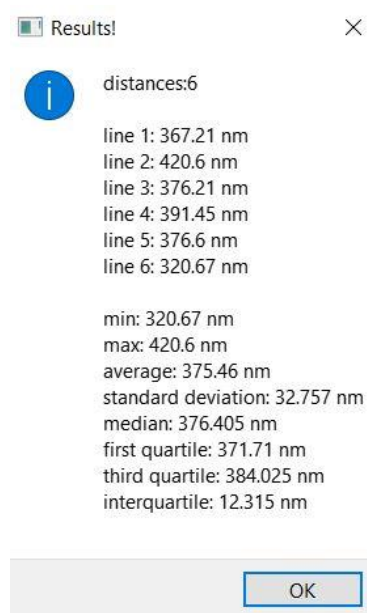
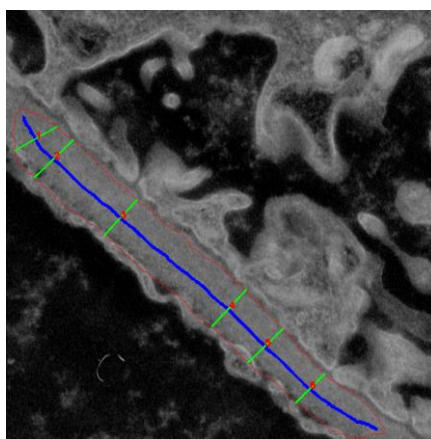


Figure 6.8.3 Αποτελέσματα μετρήσεων

Όμως υπάρχουν και κάποιες περιπτώσεις στις οποίες το πρόγραμμα αποτυγχάνει να βρεί την μεμβράνη όπως το παράδειγμα πιο κάτω.

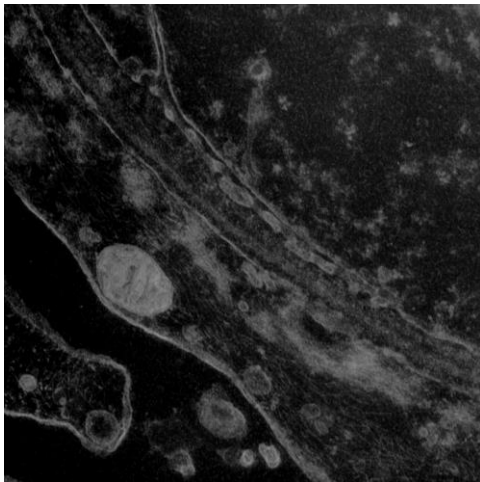


Figure 6.9.1 Αποκομμένο κομμάτι

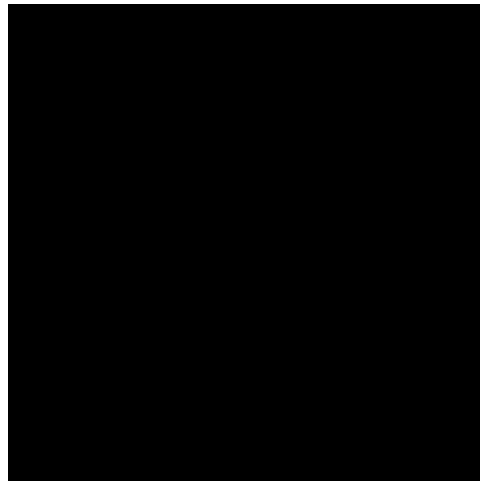


Figure 6.9.2 Πρόβλεψη Unet

Αυτό συμβαίνει επειδή η μεμβράνη αυτής της εικόνας είναι πολύ δυσδιάκριτη και το Unet αποτυγχάνει να την βρεί.

Άρα συμπεραίνουμε ότι αφού το πρόγραμμα αυτό αποτυγχάνει για πολλά δυσδιάκριτα παραδείγματα αλλά είναι πολύ ακριβή για ευδιάκριτα παραδείγματα, θα ήταν καλό να υπάρχουν δύο προγράμματα, το ένα να λειτουργεί όπως αναφέραμε στο κεφάλαιο 5 και το άλλο να κάνει χρήση του Unet.

Κεφάλαιο 7

Συμπεράσματα

7.1 Αλγόριθμοι για Image Segmentation	79
7.2 Unet και machine learning	80
7.3 Τελική αξιολόγηση του προγράμματός μας	80
7.4 Future work	81

7.1 Αλγόριθμοι για Image Segmentation

Σε αυτή τη διπλωματική εργασία αποκόμισα πολλές γνώσεις σχετικά με το image segmentation. Μελέτησα πάρα πολλά είδη supervised και unsupervised segmentation, όπως είναι οι αλγόριθμοι canny edges, chan vese και kmeans. Δυστυχώς δεν καταφέραμε να χρησιμοποιήσουμε κάποιον από αυτούς στην εργασία αυτή εξ αιτίας των περιορισμών του κάθε ενός. Ο Canny edges έπαιρνε περισσότερες λεπτομέρειες από εκείνες που χρειαζόμασταν και δεν μπορούσαμε να απομονώσουμε το κομμάτι που θέλαμε. Ο kmeans έκανε ομαδοποίηση των χρωμάτων της ικονας το οποίο δεν μας βοηθούσε σχεδόν καθόλου. Ο chan vese ήταν ο καλύτερος από τους τρεις που μελέτησα για την περίπτωση μας καθώς έφτανε πιο κοντά στο σημείο που χρειαζόματαν να μετρήσουμε χωρίς να παίρνει τα αχρηαστα κομμάτια που έπαιρνε ο canny. Δυστυχώς δεν μπορούσα να χρησιμοποιήσω κάποιον από αυτούς τους αλγορίθμους, αφού όλοι τους έπαιρναν το κομμάτι γύρω από την μεμβράνη το οποίο θέλαμε να αποφύγουμε να μετρήσουμε.

7.2 Unet και machine learning

Με image segmentation ασχολήθηκα και μέσω του νευρωνικού δικτύου Unet σε συνδυασμό με μηχανική μάθηση. Αποκόμισα πολλές γνώσεις όσο αφορά την αρχιτεκτονική του δικτύου και πως δουλεύει και πως πρέπει να διαχειριζόμαστε ένα μοντέλο ενός δικτύου σαν κι αυτό. Το Unet είναι ένα πολύ καλό εργαλείο για Image segmentation σε Bio Medical Images, αλλά χρειάζεται αρκετά παραδείγματα δειγμάτων με την μάσκα τους, πράγμα που δυστυχώς δεν είχαμε στην κατοχή μας, έτσι χρησιμοποιήσαμε τη μέθοδο του augmentation για να παράξουμε περισσότερα δείγματα με τη χρήση αυτών που έχουμε ήδη. Αφού το Unet όμως αποτυγχάνει να βρει τη μεμβράνη για παραδείγματα τα οποία είναι δυσδιάκριτα, κατά την άποψη μου είναι προτιμότερο ο χρήστης να επιλέγει με σημεία την περιοχή που θέλει να μετρήσει, ζήτημα που παίρνει λίγα δευτερόλεπτα και όχι πολύ κόπο, παρά να του δίνει το πρόγραμμα λάθος μετρήσεις.

7.3 Τελική αξιολόγηση του προγράμματός μας

Σε γενικές γραμμές για το πρόγραμμα είχαν γίνει πολλές μελέτες σχετικά με το image segmentation, την απόκτηση μετρήσεων και την μετατροπή τους στην πραγματική τους κλίμακα, τα contour areas και τη λήψη σκελετού εικόνας.

Κάποια από τα πιο πάνω σημεία τα οποία είχα μελετήσει δεν χρησιμοποιήθηκαν στο τελικό version του προγράμματος, αλλά δεν πάνουν να απαρτίζουν ένα σημαντικό κομμάτι αυτής της εργασίας. Σε γενικές γραμμές το πρόγραμμα είναι εύκολο στη χρήση, αρχικά ο χρήστης καθοδηγείται στην επιλογή του film, μαζί και με άλλες σημαντικές πληροφορίες που το αφορούν. Στη συνέχεια επιλέγει το κομμάτι στο film που θέλει να επικεντρωθεί. Ακολουθώς το πρόγραμμα του εμφανίζει το αποκομμένο κομμάτι και καλείται αυτός να επιλέξει την περιοχή στην οποία θα πάρει μετρήσεις, ή αν χρησιμοποιηθεί η δεύτερη εκδοχή του προγράμματος μας που κάνει χρήση του Unet τότε το πρόγραμμα θα του επιστρέψει την περιοχή της μεμβράνης αυτόματα. Τέλος, το πρόγραμμα του εμφανίζει διάφορες μετρήσεις πάνω σε εκείνη την περιοχή που επέλεξε μαζί με το minimum τους, το maximum, τον μέσο όρο, το standard deviation, την μέση τιμή, το πρώτο και τρίτο τεταρτημόριό τους και το interquartile των μετρήσεων. Επίσης να αναφέρουμε ότι οι τιμές είναι σε νανόμετρα και είναι με υψηλή ακρίβεια.

7.4 Future Work

Σε μελλοντικό στάδιο το πρόγραμμα μπορεί να είναι ικανό να παίρνει αυτόματα την περιοχή για τις μετρήσεις μετά την αποκοπή κομματιού από το film και σε δυσδιάκριτα film. Αυτό θα γίνει με το ήδη έτοιμο μοντέλο Unet που έφτιαξα. Με περισσότερες εικόνες δειγμάτων για το training και το testing το unet θα έχει λάβει επαρκή υλικό για να κάνει σωστά την εκπαίδευσή του. Έτσι θα είναι ικανό να παρουσιάζει στο χρήστη την περιοχή των μετρήσεων αυτόματα ακόμη και αν δεν είναι τόσο ευδιάκριτη.

Ακόμη μια εφαρμογή που θα ήταν καλό να γίνει είναι η αναγνώριση του magnification του film, χωρίς να χρειάζεται ο χρήστης να το εισχωρήσει στο πρόγραμμα. Αυτό θα επιτευχθεί με άλλο νευρωνικό δίκτυο το οποίο θα κάνει αναγνώριση χαρακτήρων πάνω σε εικόνα και θα επιστρέφει τον αριθμό της μεγένθυσης στο πρόγραμμα.

Παράρτημα 1

Περιγραφή συστήματος

Με το πάτημα του πιο κάτω συνδέσμου θα μεταφερθείτε στο GitHub, όπου έχει αναρτηθεί όλο το σύστημα https://github.com/ggrego04/electron_microscopy_digitisation.

final.py

Αυτό το αρχείο, είναι η πρώτη εκδοχή του συστήματος με τη χειροκίνητη επιλογή της μεμβράνης. Αρχικά εμφανίζει στο χρήστη ένα UI, το οποίο το δημιούργησα με τη χρήση του framework Qt Designer, και ζητά από το χρήστη να φορτώσει το film το οποίο θέλει να επεξεργαστεί. Με το πάτημα του Browse κουμπιού, ανήγει ένα παράθυρο περιήγησης στον υπολογιστή του χρήστη και αυτός επιλέγει το αρχείο και πατάει OK. Ακολούθως μετά τη φόρτωση του αρχείου, ο χρήστης θα πρέπει να δώσει στο UI τις διαστάσεις του film σε mm καθώς και τη μεγένθυση του film που αναγράφεται πάνω σε αυτό.

Όταν ο χρήστης πατήσει OK, τότε το πρόγραμμα φορτώνει την εικόνα και την παρουσιάζει σε plot, με τη βοήθεια της βιβλιοθήκης matplotlib. Σε αυτό το σημείο ο χρήστης επιλέγει με το mouse του το σημείο που θέλει να εστιαστεί μέσω drag and release. Εδώ χρησιμοποιήθηκε το RectangleSelector widget του matplotlib. Μετά την επιλογή, το πρόγραμμα τετραγωνίζει το κομμάτι αυτό χρησιμοποιώντας την πιο μεγάλη πλευρά του ορθογωνίου της επιλογής του χρήστη. Αυτό παίρνει πιο μεγάλο σημείο από αυτό που επέλεξε ο χρήστης και έτσι δεν αποκόβει το σημείο που θέλει να μελετήσει. Τέλος αποθηκεύεται το κομμάτι σαν εικόνα και υπολογίζονται οι διαστάσεις του σε mm όπως αναφέρω στο υποκεφάλαιο 4.3.

Μετά την αποκοπή του κομματιού, παρουσιάζεται το κομμάτι και ζητάει από το χρήστη να επιλέξει με το mouse του την περιοχή της μεμβράνης. Αυτό επιτυγχάνεται κάνοντας clicks στα σημεία της περιφέρειας της μεμβράνης, οι συντεταγμένες των οποίων αποθηκεύονται σε μια λίστα. Ακολούθως, με τη χρήση της εντολής cv2.drawContours και των σημείων της λίστας, το πρόγραμμα δημιουργεί το περίγραμμα της μεμβράνης. Στη συνέχεια γίνεται αντίγραφο της πιο πάνω εικόνας και με την εντολή cv2.fillPoly γεμίζεται το εσωτερικό της περιοχής με κόκκινο χρώμα. Αυτό γίνεται για να μπορούμε να αποκόψουμε το περίγραμμα, το οποίο επιτυγχάνεται βάφοντας με μαύρο χρώμα όποια σημεία δεν είναι κόκκινα στο αντίγραφο.

Ο λόγος που αποκόψαμε το περίγραμμα της περιοχής είναι για να μπορέσουμε να υπολογίσουμε το σκελετό της και αυτό επιτυγχάνεται με τη χρήση της βιβλιοθήκης FilFinder. Μετά τον υπολογισμό του σκελετού, παρουσιάζεται το κομμάτι με την περιοχή που επέλεξε ο χρήστης, όπου με κόκκινο χρώμα απεικονίζεται το περίγραμμα της περιοχής και με μπλέ χρώμα εμφανίζεται ο σκελετός της.

Μετά τον υπολογισμό του σκελετού καλούμαστε να βρούμε αρκετές κάθετες γραμμές πάνω του οι οποίες θα είναι και οι μετρήσεις μας. Έτσι, έπρεπε να βρούμε αρκετά σημεία πάνω στο σκελετό από τα οποία θα τραβούσαμε τις γραμμές. Για να το πετύχουμε αυτό έπρεπε πρώτα να βρούμε την κλίση του σκελετού με σκοπό να πάρουμε τα σημεία με τη σωστή σειρά αφού για να υπολογιστεί μια κάθετη σε ένα σημείο, πρέπει να γνωρίζουμε και το προηγούμενο του. Αυτό το πετυχαίνουμε με το να ξέρουμε όλα τα σημεία του σκελετού και αφού το FilFinder δεν μας επέστρεφε τα σημεία του σκελετού, εμείς έπρεπε να το κάνουμε χειροκίνητα. Ψάξαμε λοιπόν και βρήκαμε όλα τα μπλέ σημεία πάνω στην εικόνα μας. Μετά βρήκαμε την πιο μεγάλη y-coordinate και την πιο μικρή y-coordinate και τις αφαιρέσαμε, βρίσκοντας έτσι το μήκος του σκελετού στο y-axis. Επαναλάβαμε την ίδια διαδικασία και για τα x-coordinates βρίσκοντας το πλάτος του σκελετού στο x-axis. Μετά συγκρίναμε τις δύο διαφορές βρίσκοντας τη μεγαλύτερη. Αν η μεγαλύτερη ήταν η y, τότε η κλίση του σκελετού είναι κάθετη, αν η μεγαλύτερη ήταν η x τότε η κλίση του σκελετού είναι οριζόντια. Αφού διευκρινήσαμε ποιά είναι η κλίση του σκελετού, μένει να πάρουμε ένα αριθμό σημείων πάνω σε αυτόν από τα οποία θα πάρουμε τις γραμμές μας. Αν η κλίση του σκελετού είναι κάθετη, τότε θα κάνουμε τον έλεγχο της εικόνας για τα σημεία οριζόντια ή αλλιώς κάθετα, παίρνοντας έτσι περίπου 20 σημεία με τη σειρά πάνω στο σκελετό.

Μετά τον υπολογισμό των σημείων υπολογίστηκαν οι κάθετες μια μια με την χρήση της μεθόδου getPerpCoord η οποία τραβούσε τις κάθετες πάνω στο σημείο του σκελετού αλλά αυτές έβγαιναν εκτός της περιοχής που μετρούσαμε, έτσι έπρεπε να βρούμε τα σημεία του intersection με την κόκκινη γραμμή της περιοχής. Για να το πετύχουμε αυτό πήραμε μια μια τις κάθετες γραμμές, βρήκαμε ποιά σημεία του contour area αλλάζαν χρώμα και γίνονταν πράσινα (αφού οι κάθετες γραμμές που υπολογίζονταν απεικονίζονταν με πράσινο χρώμα) και τα αποθηκεύαμε. Μετά πήραμε αυτά τα σημεία και χρησιμοποιήσαμε την εντολή cv2.line για να απεικονίσουμε τη γραμμή μεταξύ των δύο σημείων. Μαζί με την απεικόνιση, υπολογίσαμε την απόσταση τους με τον τρόπο που ανέφερα στο υποκεφάλαιο 5.4.

Αφού είχαμε στην κατοχή μας την απόσταση της κάθε γραμμής, έμεινε να τις εμφανίζουμε με κάποιο τρόπο στο χρήστη, μαζί και με κάποιες ολικές μετρήσεις όπως για παράδειγμα των μέσο όρο. Οι ολικές μετρήσεις που υπολογίστηκαν ήταν η *minimum* απόσταση, η *maximum* απόσταση, ο μέσος όρος, το *standard deviation*, η μέση τιμή, το πρώτο τεταρτημόριο, το τρίτο τεταρτημόριο και το *interquartile* που είναι η διαφορά των δύο τεταρτημορίων. Αποφάσισα ότι θα ήταν καλή ιδέα να χρησιμοποιήσω ένα *QMessageBox* το οποίο θα τα παρουσίαζε και έτσι πέρασα ένα *string* μέσα το οποίο περιείχε τον αριθμό των αποστάσεων που υπολογίστηκαν, την απόσταση της κάθε γραμμής και τις ολικές μετρήσεις που ανάφερα πιο πάνω.

Unet_training.py

Αυτό το πρόγραμμα υλοποιεί το νευρωνικό δίκτυο *Unet* το οποίο μας βοηθά στην αυτόματη επιλογή της μεμβράνης μετά την επιλογή του κομματιού από το χρήστη και χρησιμοποιείται στη δεύτερη εκδοχή του προγράμματός μας *final_v2.py*.

Αρχικά πήραμε με τη χρήση του *final.py* δείγματα μαζί με την μάσκα τους για να δώσουμε στο *Unet* με σκοπό να εκπαιδευτεί. Δημιουργήσαμε ένα φάκελο με *train* δεδομένα ο οποίος περιείχε ένα φάκελο για κάθε δείγμα ο οποίος με τη σειρά του περιείχε άλλους δύο φακέλους. Ένας περιείχε το δείγμα και ο άλλος τη μάσκα του.

Το πρόγραμμα ξεκινά συλλέγοντας από τους φακέλους τα δείγματα, τα οποία αποθηκεύει σε ένα *array*, το ίδιο γίνεται και για τις μάσκες. Ακολούθως δημιουργείται το νευρωνικό δίκτυο με τον τρόπο που ανάφερα στο υποκεφάλαιο 6.2 και με την εντολή `results = model.fit(X_train, Y_train, validation_split=0.1, batch_size=16, epochs=50)` το πρόγραμμα εκπαιδεύτηκε για 50 εποχές με τα *training samples* που του έχουμε δώσει. Στη συνέχεια, αποθηκεύσαμε το εκπαιδευμένο μοντέλο για να μπορεί να το χρησιμοποιήσει αργότερα το σύστημά μας με την εντολή `model.save('model_for_segmentation.h5')`. Αργότερα για σκοπούς *validation*, κάναμε *load* το εκπαιδευμένο μοντέλο με την εντολή `model = tf.keras.models.load_model("model_for_segmentation.h5")` και χρησιμοποιήσαμε ένα *test_image* για να εξετάσουμε την ακρίβεια του μοντέλου μας. Κάναμε *predict* με τη χρήση της μεθόδου *prediction* και η μάσκα της *segmented* εικόνας εμφανίζεται στην οθόνη.

Unet.py

Αυτό είναι το πρόγραμμα από το οποίο καλεί τη μέθοδο prediction το final_v2.py για να πάρει το segmented μέρος της μεμβράνης. Μέσα περιέχει το μοντέλο Unet που χρησιμοποιήθηκε και στο Unet_training.py και τη μέθοδο prediction η οποία παίρνει την εικόνα και επιστρέφει πίσω τη μάσκα.

augment.py

Αυτό το πρόγραμμα παίρνει όλα τα δείγματα που έχουμε για την εκπαίδευση του δικτύου και κάνει διάφορες παραλλαγές τους, δηλαδή crop, rotate, mirror και zoom. Και τα αποθηκεύει στο φάκελο train σαν διαφορετικά δείγματα.

final_v2.py

Αυτή είναι η δεύτερη εκδοχή του προγράμματός μας η οποία υπολογίζει την περιοχή της μεμβράνης που θα μετρηθεί χωρίς να χρειάζεται ο χρήστης να επέμβει. Κάνει ακριβώς ότι κάνει το final.py, με την διαφορά ότι μετά την επιλογή του κομματιού από το χρήστη, αντί αυτός να χρειάζεται να επιλέξει με το mouse του την περιοχή της μεμβράνης, το πρόγραμμα παίρνει το κομμάτι, το κάνει resize σε διαστάσεις 128x128 και με τη χρήση της μεθόδου prediction από το Unet.py παίρνει την περιοχή της μεμβράνης σε μορφή μάσκας. Μετά συνεχίζεται η ίδια διαδικασία με το final.py όπου αφαιρείται το περίγραμμα της μεμβράνης, υπολογίζεται ο σκελετός, παίρνουμε και μετρούμε τις κάθετες αποστάσεις πάνω στο σκελετό και μετά εμφανίζουμε τα αποτελέσματα στο χρήστη.

image_segmentation_algorithms.py

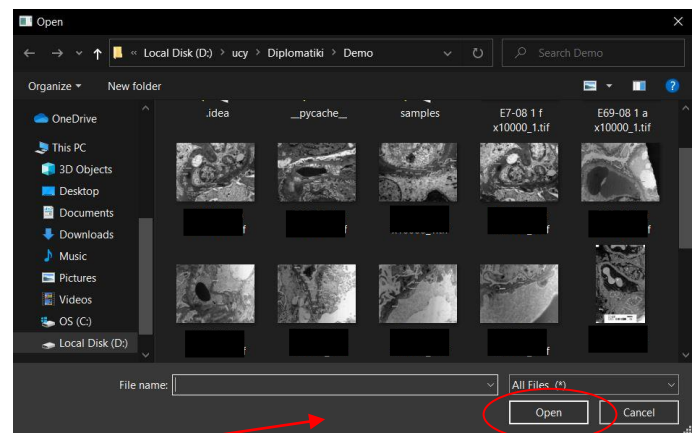
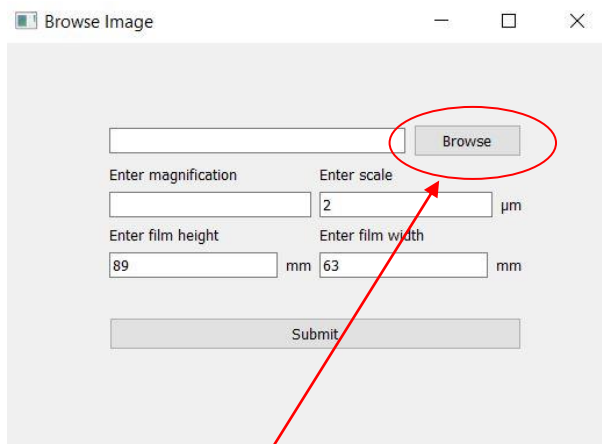
Αυτό το πρόγραμμα κάνει εφαρμογή των τριών αλγορίθμων που μελετήθηκαν και αναφέρθηκαν στο κεφάλαιο 3. Διαβάζεται από το πρόγραμμα μια εικόνα και πάνω της εφαρμόζεται ο αλγόριθμος canny edges, ο αλγόριθμος k-means και ο αλγόριθμος chan vese. Τέλος, παρουσιάζονται τα αποτελέσματα των τριών αλγορίθμων σε ένα plot.

Παράρτημα 2

Manual χρήσης των δύο συστημάτων

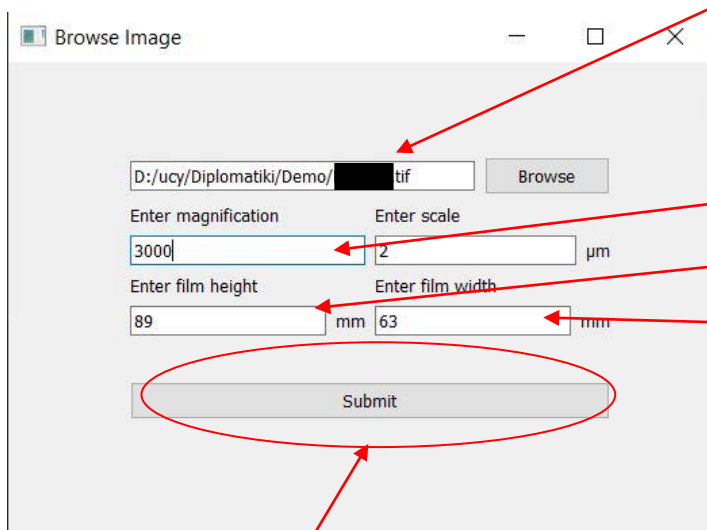
final.py

Με το ξεκίνημα του προγράμματος, το πρώτο παράθυρο που ανήγει είναι το Browse image, όπου ο χρήστης θα επιλέξει την εικόνα που θέλει να μελετήσει.



Με το πάτημα του κουμπιού Browse ανήγει το παράθυρο περιήγησης εικόνας

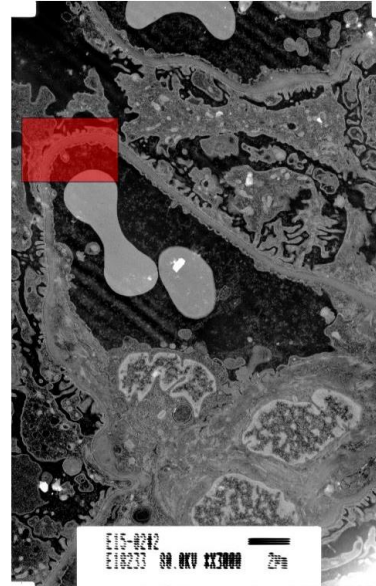
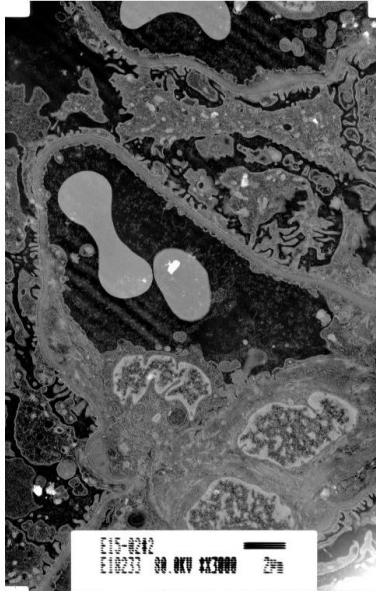
Επιλέγοντας την εικόνα και πατώντας Open φορτώνεται η εικόνα



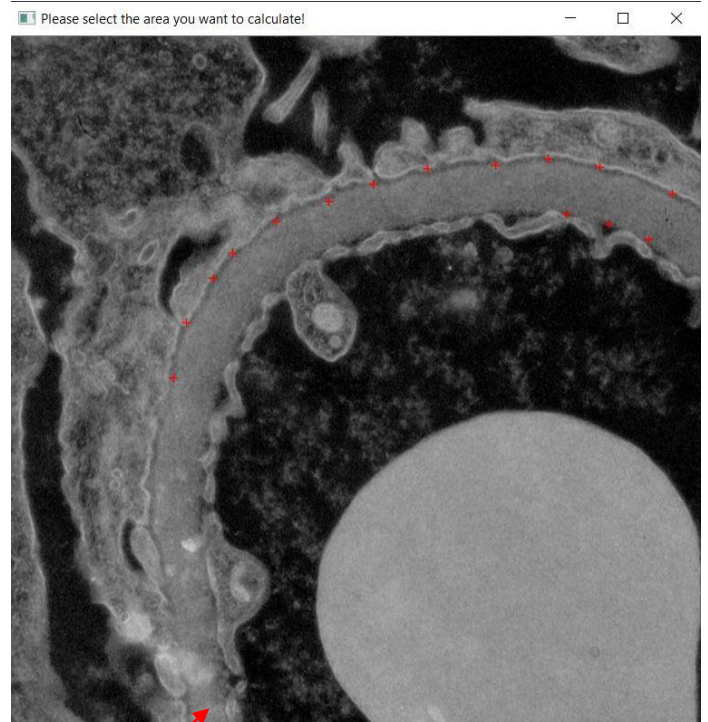
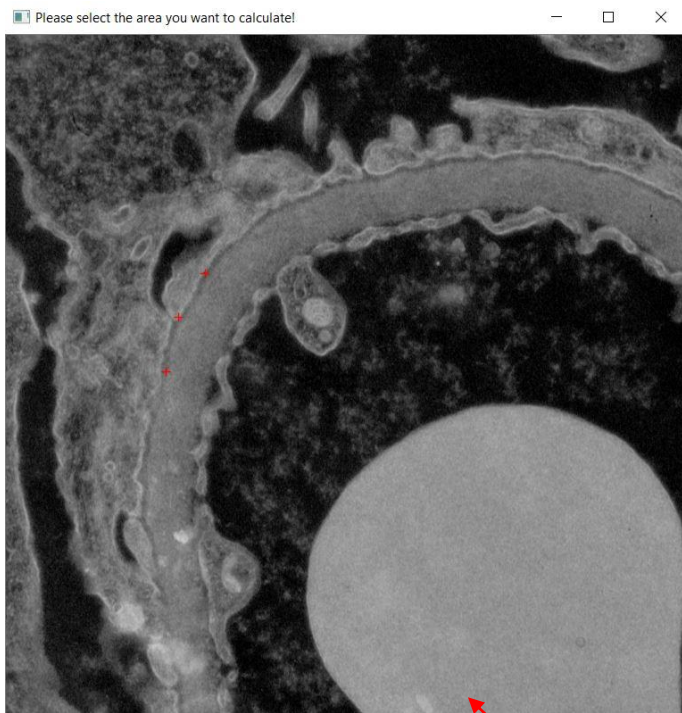
Ο χρήστης πρέπει να δώσει το magnification του film, το μήκος και το πλάτος είναι δοσμένα από το πρόγραμμα αλλά μπορεί ο ίδιος να τα αλλάξει

Πατώντας Submit το πρόγραμμα προχωρεί στην επόμενη φάση

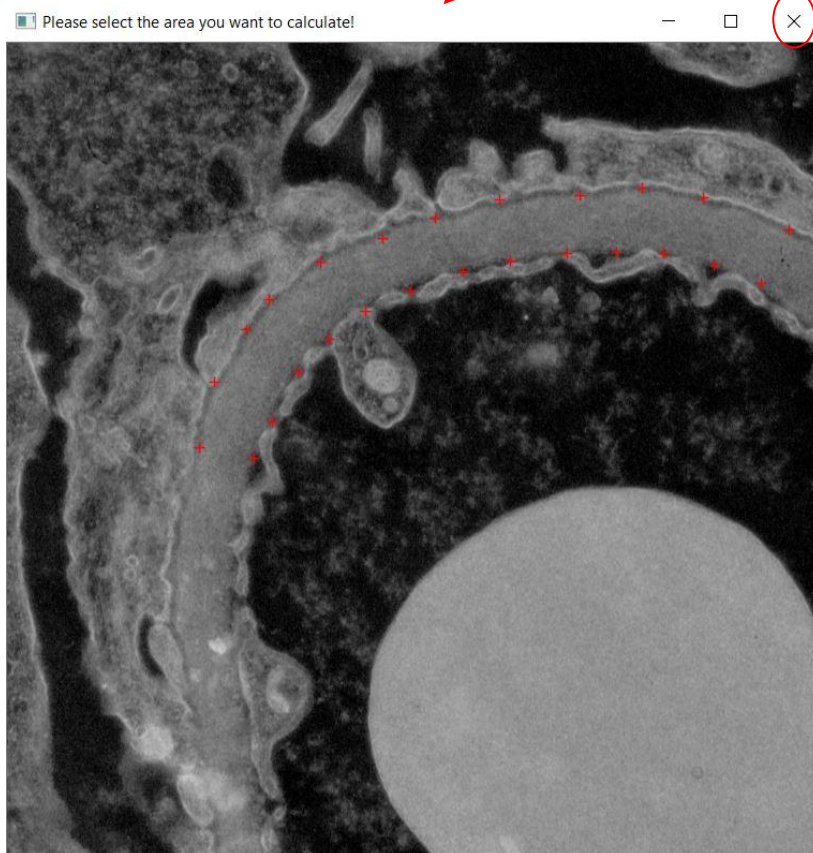
Το σύστημα εμφανίζει στο χρήστη την εικόνα που φορτώθηκε και αυτός με τη σειρά του θα πρέπει να επιλέξει το σημείο τοθ film το οποίο θέλει να εστιάσει με τη χρήση του mouse του κάνοντας click, drag και release.



Το πρόγραμμα εμφανίζει στο χρήστη το αποκομμένο κομμάτι που επέλεξε και του ζητά να επιλέξει με πολλά σημεία και τη χρήση του mouse (με clicks).

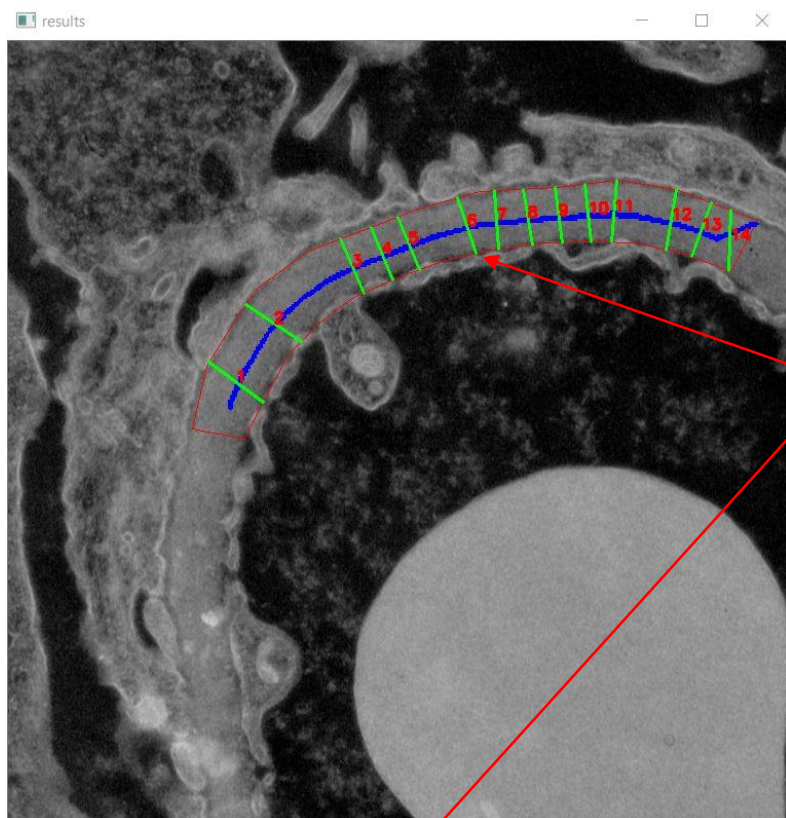


Ο χρήστης επιλέγει την περιοχή της μεμβράνης όπως φαίνεται στις εικόνες.



Με το πάτημα του κουμπιού X ο χρήστης λέει στο σύστημα ότι τελείωσε με την επιλογή της περιοχής.

Το σύστημα εμφανίζει τα πιο κάτω αποτελέσματα.



Κάθε κάθετη γραμμή έχει και τον αριθμό της ο οποίος θα επιτρέπει στο χρήστη να γνωρίζει την απόσταση κάθε γραμμής που αναγράφεται στο message box κάτω

Results!



distances:14

line 1: 538.43 nm
line 2: 530.21 nm
line 3: 506.36 nm
line 4: 488.47 nm
line 5: 488.47 nm
line 6: 502.79 nm
line 7: 508.3 nm
line 8: 490.31 nm
line 9: 488.63 nm
line 10: 502.18 nm
line 11: 530.13 nm
line 12: 526.6 nm
line 13: 491.04 nm
line 14: 521.99 nm

min: 488.47 nm
max: 538.43 nm
average: 508.14 nm
standard deviation: 18.087 nm
median: 504.5750000000005 nm
first quartile: 490.675 nm
third quartile: 524.295 nm
interquartile: 33.62 nm

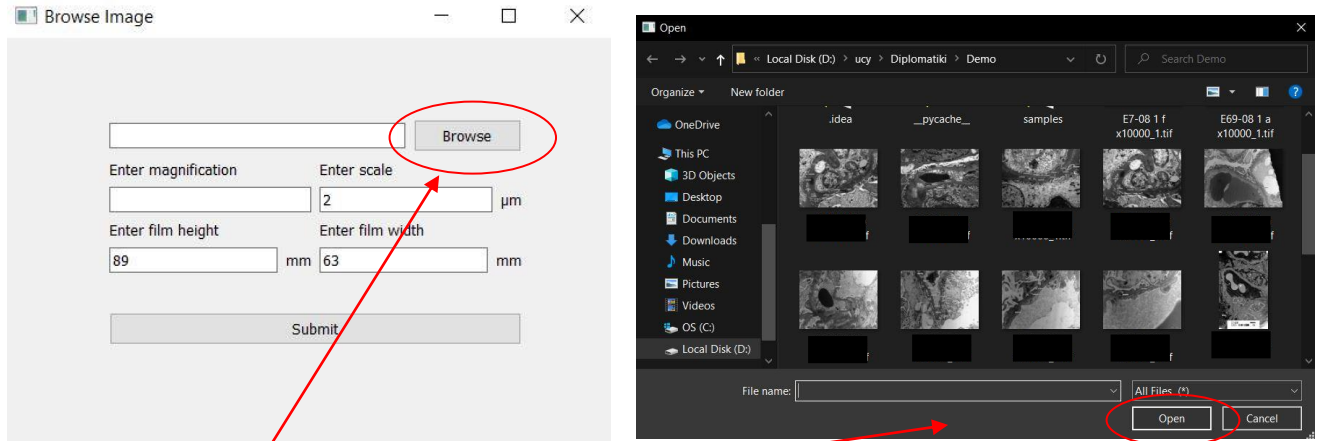
Το παράθυρο Results περιέχει το μήκος κάθε γραμμής σε nm μαζί με τη μικρότερη απόσταση, τη μεγαλύτερη απόσταση, το μέσο όρο, το standard deviation, τη μέση τιμή, το πρώτο και το τρίτο τεταρτημόριο των τιμών καθώς και την διαφορά τους (interquartile)

OK

Πατώντας OK ο χρήστης τερματίζει το πρόγραμμα

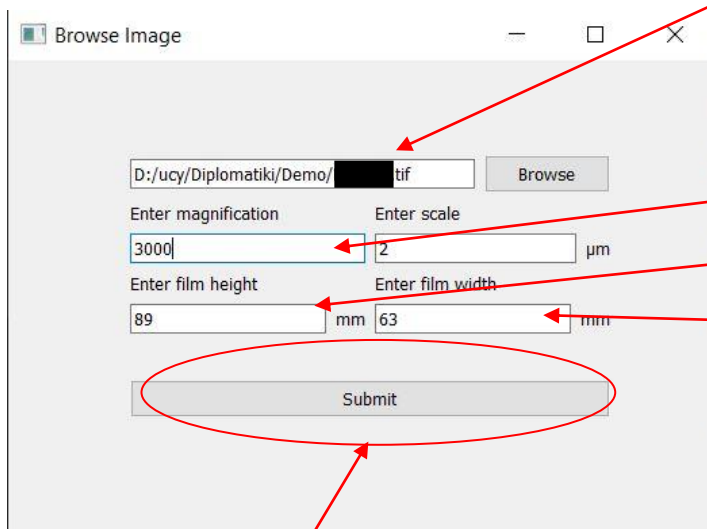
final v2.py

Με το ξεκίνημα του προγράμματος, το πρώτο παράθυρο που ανήγει είναι το Browse image, όπου ο χρήστης θα επιλέξει την εικόνα που θέλει να μελετήσει.



Με το πάτημα του κουμπιού Browse ανήγει το παράθυρο περιήγησης εικόνας

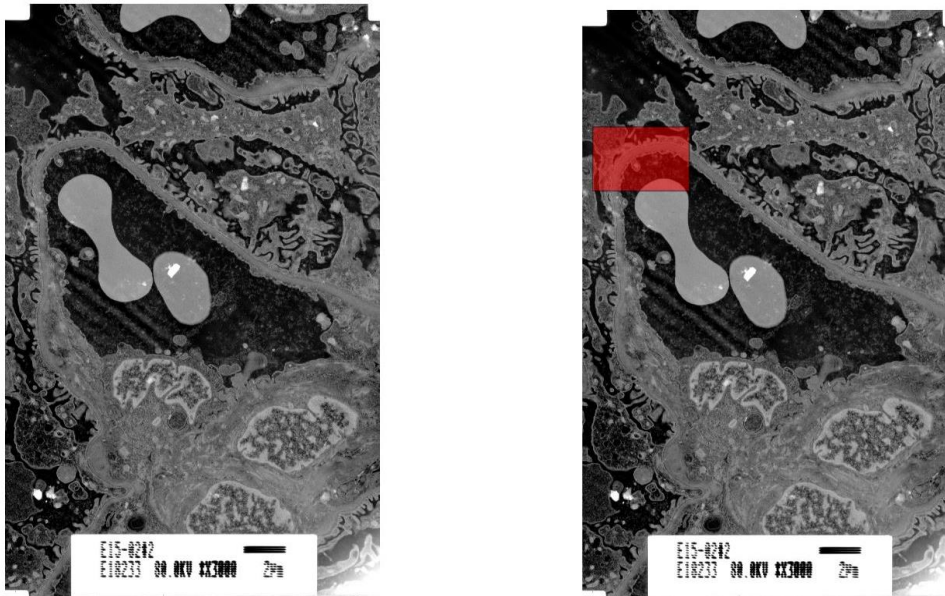
Επιλέγοντας την εικόνα και πατώντας Open φορτώνεται η εικόνα



Ο χρήστης πρέπει να δώσει το magnification του film, το μήκος και το πλάτος είναι δοσμένα από το πρόγραμμα αλλά μπορεί ο ίδιος να τα αλλάξει

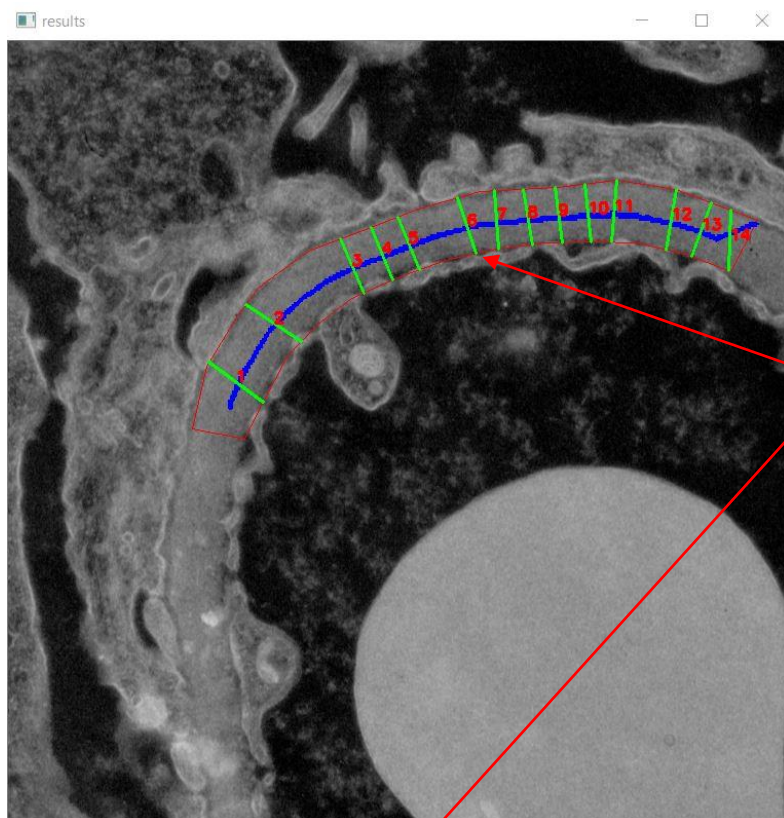
Πατώντας Submit το πρόγραμμα προχωρεί στην επόμενη φάση

Το σύστημα εμφανίζει στο χρήστη την εικόνα που φορτώθηκε και αυτός με τη σειρά του θα πρέπει να επιλέξει το σημείο τοθ film το οποίο θέλει να εστιάσει με τη χρήση του mouse του κάνοντας click, drag και release.

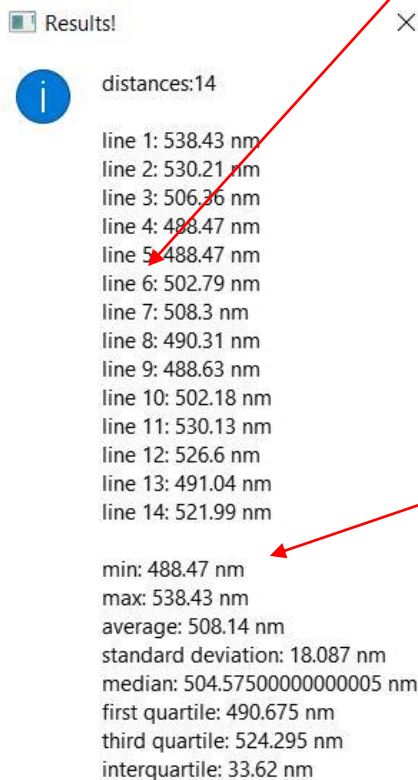


Μετά από αυτό το σημείο δεν χρειάζεται περεταίρω παρέμβαση από το χρήστη.

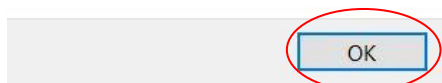
Το σύστημα εμφανίζει τα πιο κάτω αποτελέσματα.



Κάθε κάθετη γραμμή έχει και τον αριθμό της ο οποίος θα επιτρέπει στο χρήστη να γνωρίζει την απόσταση κάθε γραμμής που αναγράφεται στο message box κάτω



Το παράθυρο Results περιέχει το μήκος κάθε γραμμής σε nm μαζί με τη μικρότερη απόσταση, τη μεγαλύτερη απόσταση, το μέσο όρο, το standard deviation, τη μέση τιμή, το πρώτο και το τρίτο τεταρτημόριο των τιμών καθώς και την διαφορά τους (interquartile)



Πατώντας OK ο χρήστης τερματίζει το πρόγραμμα

Βιβλιογραφία

- [1] Epson Perfection V550 Photo Specifications
<https://www.epson.eu/products/scanners/consumer-scanners/perfection-v550-photo#specifications>

- [2] Unet
<https://towardsdatascience.com/understanding-semantic-segmentation-with-unet-6be4f42d4b47>
https://github.com/bnsreenu/python_for_microscopists
<https://medium.com/@pallawi.ds/semantic-segmentation-with-u-net-train-and-test-on-your-custom-data-in-keras-39e4f972ec89>

- [3] Image Segmentation
https://docs.opencv.org/master/d3/db4/tutorial_py_watershed.html
<https://circuitdigest.com/tutorial/image-segmentation-using-opencv>
<https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/>
https://scikit-image.org/docs/dev/auto_examples/segmentation/plot_chan_vese.html

- [4] FilFinder για εύρεση σκελετού εικόνας
<https://fil-finder.readthedocs.io/en/latest/tutorial.html>
https://scikit-image.org/docs/dev/auto_examples/edges/plot_skeleton.html

- [5] Rectangle Selector Tool
https://matplotlib.org/3.3.3/gallery/widgets/rectangle_selector.html
https://het.as.utexas.edu/HET/Software/Matplotlib/examples/widgets/rectangle_selector.html

- [6] Click Events πάνω σε plot
<https://plotly.com/python/click-events/>

- [7] Formula μεγένθυσης πάνω στο film
<https://www.bbc.co.uk/bitesize/guides/z3vypbk/revision/3>