



Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Μηχανολόγων Μηχανικών

Τομέας Τεχνολογίας των Κατεργασιών

**Σύστημα Αυτόματης Παρακολούθησης και Ρύθμισης
Διάταξης Έγχυσης Ρητίνης με τη Βοήθεια Κενού για την
Παραγωγή Επίπεδων Τεμαχίων από Σύνθετα Υλικά**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΣΤΕΦΑΝΟΣ ΛΕΚΑΝΙΔΗΣ

Επιβλέπων: Γ.Χ.ΒΟΣΝΙΑΚΟΣ

Καθηγητής Ε.Μ.Π.

Αθήνα, Δεκέμβριος 2018

*Στην μνήμη του πατέρα μου,
Γιώργου*

Ευχαριστίες

Στο σημείο αυτό, τελειώνοντας την εκπόνηση της διπλωματικής μου εργασίας και κλείνοντας ένα μεγάλο και σημαντικό κεφάλαιο για μένα, αυτό των σπουδών στην σχολή των Μηχανολόγων Μηχανικών, θα ήθελα να ευχαριστήσω τον καθηγητή κ. Γ.Χ. Βοσνιάκο για την ευκαιρία που μου έδωσε να ασχοληθώ με ένα τόσο ενδιαφέρον και πολύπλευρο από άποψη γνώσεων θέμα, δίνοντάς μου την δυνατότητα εκτός από το να διευρύνω τις γνώσεις μου σε διάφορους τομείς, να μάθω να τις συνδυάζω με τέτοιον τρόπο ώστε να δημιουργηθεί ένα αρμονικό αποτέλεσμα. Θα ήθελα επίσης να τον ευχαριστήσω για την άψογη συνεργασία που είχαμε, καθώς και για την άμεση ανταπόκρισή του για οποιοδήποτε ζήτημα προέκυπτε κατά την διάρκεια εκπόνησης της διπλωματικής.

Επίσης, θα ήθελα να ευχαριστήσω τον κ. Ν. Μελισσά και τον κ. Κ. Κερασιώτη για την σημαντική τους βοήθεια σχετικά με τον εξοπλισμό του εργαστηρίου και τον απόφοιτο Π. Δημακόπουλο για την ενημέρωση που μου παρείχε σχετικά με την κατεργασία VARI.

Τέλος δεν θα μπορούσα να παραλείψω να ευχαριστήσω την οικογένειά μου για την αμέριστη υποστήριξή της όλα αυτά τα χρόνια.

Περίληψη

Ο σκοπός της παρούσας διπλωματικής εργασίας είναι η ανάπτυξη ενός συστήματος παρακολούθησης και ρύθμισης διάταξης της έγχυσης της ρητίνης για την παραγωγή τεμαχίων από σύνθετα υλικά με τη βοήθεια κενού, χρησιμοποιώντας την μέθοδο Vacuum Assisted Resin Infusion (VARI).

Για την πραγματοποίηση του συστήματος αυτού χρησιμοποιήθηκε μια συμβατική κάμερα διαδικτύου για την λήψη στιγμιότυπων της ροής της ρητίνης και το προγραμματιστικό περιβάλλον MATLAB για την ανάπτυξη του κατάλληλου κώδικα για τον εντοπισμό του μετώπου της ροής της ρητίνης, σχεδιάστηκε και υλοποιήθηκε το υδραυλικό /πνευματικό σύστημα για την έγχυση της ρητίνης στο καλούπι και τέλος αναπτύχθηκε ο κώδικας και το ηλεκτρονικό κύκλωμα που είναι απαραίτητα για τον έλεγχο του υδραυλικού/πνευματικού συστήματος μέσω του μικροελεγκτή Arduino.

Στην αρχή της εργασίας, πραγματοποιείται μια παρουσίαση για την κατεργασία σύνθετων υλικών VARI. Στην συνέχεια παρουσιάζονται μέθοδοι που έχουν αναπτυχθεί για τον εντοπισμό του μετώπου της ροής της ρητίνης. Επιπλέον πραγματοποιείται μια θεωρητική ανασκόπηση για την επεξεργασία εικόνας.

Στα επόμενα κεφάλαια περιγράφεται η μέθοδος εντοπισμού του μετώπου της ροής της ρητίνης και του ελέγχου αυτής μέσω των βαλβίδων και του μικροελεγκτή Arduino.

Τέλος, παρουσιάζονται τα αποτελέσματα και τα συμπεράσματα του συστήματος παρακολούθησης και ρύθμισης διάταξης της έγχυσης της ρητίνης που αναπτύχθηκε στην παρούσα διπλωματική εργασία.

Λέξεις Κλειδιά: Σύνθετα Υλικά, Διαδικασία Έγχυσης Ρητίνης Με Τη Βοήθεια Κενού, Υπολογιστική Όραση, Μικροελεγκτής Arduino, Αυτόματος Έλεγχος, Βαλβίδες

Abstract

The aim of this final year thesis is to develop a system for monitoring and regulating the injection of the resin for the production of composite materials by vacuum using the VARI method.

To implement this system a conventional webcam was used to capture the flow of the resin and the MATLAB programming environment to develop the appropriate code to detect the resin flow front, the hydraulic/pneumatic system for the infusion of the resin was designed and implemented and finally, the code and the electronic circuit were developed which are necessary to control the hydraulic/pneumatic system through the Arduino microcontroller.

At the beginning of this dissertation, a presentation is made regarding the processing of composite materials via VARI method. Subsequently the methods which have been developed to detect the resin flow are shown. In addition, a theoretical review of image processing is presented.

In the following chapters the method of detecting the flow of the resin and its control through the valves and the Arduino microcontroller is described.

Last but not least, the results and conclusions of the system for monitoring and regulating the resin infusion set up in this dissertation are presented.

Keywords: Composite Materials, Vacuum Assisted Resin Infusion, Computer Vision, Arduino Microcontroller, Automatic Control, Valves

Πίνακας Περιεχομένων

ΚΕΦΑΛΑΙΟ 1 Περιγραφή κατεργασίας VARI	16
1.1 Εισαγωγή VARI	16
1.2 Μέθοδος VARI	17
1.2.1 Εισαγωγή.....	17
1.2.2 Περιγραφή μεθόδου VARI	18
1.2.3 VARI Με Χρήση Διπλής Σακούλας	18
1.2.4 Νόμος Darcy	19
1.2.5 Πιλοτική διάταξη VARI Εργαστηρίου	20
ΚΕΦΑΛΑΙΟ 2 Αυτοματοποίηση VARI.....	22
2.1 Εισαγωγή.....	22
2.2 Αισθητήρες Εντοπισμού Μετώπου Ροής.....	22
ΚΕΦΑΛΑΙΟ 3 Υπολογιστική Όραση.	31
3.1 Εισαγωγή.....	31
3.2 Ψηφιακή Εικόνα	31
3.2.1 Διαδική Εικόνα	31
3.2.2 Εικόνα Γκρι Απόχρωσης	31
3.2.3 Έγχρωμη Εικόνα.....	31
3.2.4 Ενδεικτική Εικόνα	32
3.3 Συστήματα Συντεταγμένων.....	32
3.3.1 Συντεταγμένες Εικονοστοιχείων	32
3.3.2 Χωρικές Συντεταγμένες.....	33
3.4 Βελτίωση Εικόνας.....	34
3.4.1 Εισαγωγή.....	34
3.4.2 Ιστόγραμμα Εικόνας	34
3.4.3 Τμηματοποίηση Εικόνας Με Τεχνικές Κατωφλίωσης	34

3.4.4 Συνδεσιμότητα (Connectivity)	36
3.4.5 Φίλτρα Εικόνας.....	36
3.5 Ανίχνευση Ακμών	41
3.5.1 Εισαγωγή.....	41
3.5.2 Ανίχνευση Ακμών Πρώτης Παραγώγου	42
3.5.3 Ανίχνευση ακμών Δεύτερης Παραγώγου	44
3.6 Μορφολογικές λειτουργίες	47
3.6.1 Εισαγωγή.....	47
3.6.2 Δυαδικές Εικόνες - Προσκήνιο/Παρασκήνιο	47
3.6.3 Στοιχείο Δόμησης (Structuring Element)	47
3.6.4 Διαστολή-Διάβρωση (Dilation-Erosion).....	48
3.6.5 Αποσύνθεση Στοιχείου Δόμησης.....	48
3.6.6 Συνδυασμός Μορφολογικών Λειτουργιών	49
3.6.7 Εξαγωγή Και Επισήμανση Συνδεδεμένων Στοιχείων.....	50
ΚΕΦΑΛΑΙΟ 4 Εντοπισμός του Μετώπου Ροής.....	52
4.1 Εισαγωγή.....	52
4.1.1 MATLAB ως Γλώσσα Προγραμματισμού	52
4.1.2 Εργαλειοθήκη Επεξεργασίας Εικόνας (Image Processing Toolbox)	52
4.2 Προ-Επεξεργασία Εικόνας	53
4.2.1 Εισαγωγή.....	53
4.2.2 Περικοπή (crop).....	53
4.2.3 Φιλτράρισμα εικόνας χρησιμοποιώντας το φίλτρο Gauss	55
4.2.4 Μετατροπή Εικόνας σε Απόχρωση του Γκρι και Δυαδική	55
4.2.5 Βελτίωση Εικόνας Χρησιμοποιώντας Μορφολογικές Λειτουργίες.....	57
4.2.6 Χρήση Αλγόριθμου Ανίχνευσης Ακμών	59
4.2.7 Κώδικας MATLAB.....	61
4.3 Εύρεση Θέσης του Μετώπου Ροής της ρητίνης.....	62
4.3.1 Εισαγωγή.....	62

4.3.2	Εύρεση Συντεταγμένων	62
4.3.3	Μορφολογικής Λειτουργία Διαστολής	63
4.3.4	Εξαγωγή Συντεταγμένων.....	64
4.3.5	Ορισμός Σημείων στο Μέτωπο Ροής	64
4.3.6	Κώδικας MATLAB.....	65
4.4	Επεξεργασία Υφάσματος προς Εμποτισμό	67
4.4.1	Εισαγωγή.....	67
4.4.2	Οριοθέτηση Περιοχής Ενδιαφέροντος	67
4.4.3	Ανίχνευση του Σχήματος του Υφάσματος προς Εμποτισμό	68
4.4.4	Εύρεση Συντεταγμένων Εικονοστοιχείων του Υφάσματος.....	73
4.4.5	Αλγόριθμος Γεμίματος Περιοχής στο Ύφασμα προς Εμποτισμό	74
4.4.6	Αλλαγή Τιμής Εικονοστοιχείων Υφάσματος προς Εμποτισμό	75
4.4.7	Δημιουργία Σημείων Ελέγχου για τον Έλεγχο Ροής της Ρητίνης.....	76
4.4.9	Κώδικας MATLAB.....	84
4.5	Ενσωμάτωση Μετώπου Ροής και Υφάσματος σε μια Εικόνα.....	89
4.5.1	Εισαγωγή.....	89
4.5.2	Δημιουργία Εικόνας.....	89
4.5.3	Μετατροπή Σε Έγχρωμη Εικόνα	90
4.5.4	Εμφάνιση Της Labelcol1 Στην Διάρκεια Της VARI	90
4.5.5	Εντοπισμός Εικονοστοιχείων Μετώπου Ροής Στο Ύφασμα	93
4.5.6	Κώδικας MATLAB.....	93
	ΚΕΦΑΛΑΙΟ 5 Έλεγχος Διαδικασίας VARI	95
5.1	Εισαγωγή.....	95
5.2	Βασική Προετοιμασία Προγράμματος	95
5.2.1	Κλείσιμο Ανοιχτών Παραθύρων στη MATLAB.....	95
5.2.2	Διαγραφή Μεταβλητών	95
5.2.3	Αναγνώριση Κάμερας και Μικροελεγκτή Arduino από την MATLAB.....	95
5.2.4	Έλεγχος Λειτουργίας Βαλβίδων.....	97

5.3 Επεξεργασία Υφάσματος προς Εμποτισμό	97
5.3.1 Εκτέλεση Υποπρογράμματος vkalouri.m.....	97
5.3.2 Αλλαγή Τιμής Εικονοστοιχείων Υφάσματος Εμποτισμού	97
5.3.3 Υπολογισμός Αριθμού Εικονοστοιχείων Υφάσματος Προς Εμποτισμό	98
5.3.4 Διακοπή Εκτέλεσης Προγράμματος	98
5.4 Βασικοί Υπολογισμοί – Κλήση Υποπρογραμμάτων για Κάθε Στιγμιότυπο	98
5.4.1 Επαναληπτική Διαδικασία	98
5.4.2 Στιγμιότυπο από την Κάμερα	99
5.4.3 Εκτέλεση Υποπρογράμματος vgrammh2roes.m	99
5.4.4 Εκτέλεση Υποπρογράμματος vroh.m.....	99
5.4.5 Υπολογισμός Περιοχής Εμποτισμένου Υφάσματος Με Ρητίνη.....	99
5.4.6 Υπολογισμός Ποσοστού Εμποτισμού Υφάσματος	101
5.4.7 Εκτέλεση Υποπρογράμματος arduinno.m.....	103
5.5 Κώδικας MATLAB.....	103
5.5.1 Κώδικας tworoesvideo.m.....	103
5.5.2 Κώδικας impdilation.m	105
ΚΕΦΑΛΑΙΟ 6 Έλεγχος έγχυσης ρητίνης	106
6.1 Εισαγωγή.....	106
6.2 Arduino	106
6.2.1 Πλακέτα Arduino	106
6.2.2 Γλώσσα προγραμματισμού Arduino.....	110
6.3 Μέθοδος Ελέγχου Ροής Ρητίνης	112
6.3.1 Εισαγωγή.....	112
6.3.2 Έλεγχος Παραμέτρων.....	112
6.3.3 Έλεγχος Περίπτωσης μη Επιτυχούς Κενού ή έμφραξης Θύρας Αναρρόφησης ...	115
6.3.4 Δήλωση Μεταβλητών Τύπου persistent.....	117
6.4 Κώδικας MATLAB.....	118
ΚΕΦΑΛΑΙΟ 7 Ηλεκτρονικό Κύκλωμα Ελέγχου των Βαλβίδων	122

7.1 Εισαγωγή.....	122
7.2 Στοιχεία Ηλεκτρονικού Κυκλώματος.....	123
ΚΕΦΑΛΑΙΟ 8 Υδραυλικό – Πνευματικό Κύκλωμα για Έλεγχο Έγχυσης Ρητίνης.....	128
8.1 Γενική Περιγραφή Βαλβίδων	128
8.2 Επιλογή Βαλβίδας για Έγχυση Ρητίνης.....	130
8.2.1 Αναγκαία Χαρακτηριστικά	130
8.2.2 Ιδανικές Βαλβίδες Έγχυσης Ρητίνης Υπό Κενό	131
8.2.3 Τελική Επιλογή Βαλβίδας.....	133
8.3 Σύνδεση Υδραυλικών – Πνευματικών Στοιχείων	134
8.3.1 Ηλεκτροβαλβίδα Ελέγχου	134
8.4 Εξαρτήματα.....	135
ΚΕΦΑΛΑΙΟ 9 Αυτοματοποιημένη Κατεργασία VARI	140
9.1 Εισαγωγή.....	140
9.2 Προετοιμασία εξοπλισμού	140
9.3 Έναρξη Διεργασίας	141
9.4 Καθαρισμός Βαλβίδων	144
ΚΕΦΑΛΑΙΟ 10 Συμπεράσματα και Επεκτάσεις.....	145
10.1 Συμπεράσματα	145
10.2 Μελλοντικές Επεκτάσεις.....	146

Κατάλογος Πινάκων

<i>Πίνακας 1: Εξοπλισμός VARI Μονάδας Σύνθετων Υλικών του Εργαστηρίου Τεχνολογίας Των Κατεργασιών.....</i>	<i>21</i>
<i>Πίνακας 2: Χαρακτηριστικά Arduino Uno</i>	<i>110</i>
<i>Πίνακας 3: Αντίδραση Υλικών.....</i>	<i>133</i>
<i>Πίνακας 4: Εξαρτήματα Υδρλικού Πνευματικού Κυκλώματος</i>	<i>136</i>

Κατάλογος Εικόνων

Εικόνα 1: VARI.....	17
Εικόνα 2: RTM.....	17
Εικόνα 3:VARI με χρήση Διπλής Σακούλας.....	19
Εικόνα 4: Σύστημα Συντεταγμένων Εικονοστοιχείων.....	33
Εικόνα 5: Σύστημα Χωρικών Συντεταγμένων.....	33
Εικόνα 6: Συνδεσιμότητα 4 και 8 στοιχείων.....	36
Εικόνα 7: Ο μηχανισμός φιλτραρίσματος εικόνας με 3x3 πυρήνα φίλτρου.....	38
Εικόνα 8: Φίλτρο μεσαίας τιμής.....	40
Εικόνα 9: Φίλτρο Gauss με πυρήνα και τυπική απόκλιση, 3x3 $\sigma=1$, 11x11 $\sigma=2$ και 21x21 $\sigma=4$ αντίστοιχα.....	41
Εικόνα 10: Είδη ακμών, (a) Βηματική, (b) κλιμακωτή, (c) γραμμική, (d) οροφής.....	42
Εικόνα 11: Παράδειγμα στοιχείου διαμόρφωσης (Structuring element).....	48
Εικόνα 12: Αρχικό στοιχείο δόμησης και τελικό στοιχείο δόμησης ύστερα από αποσύνθεση.....	49
Εικόνα 13: Αρχική Εικόνα (από κάμερα).....	54
Εικόνα 14: Τελική εικόνα μετά από imcrop.....	54
Εικόνα 15: Φίλτρο Gauss.....	55
Εικόνα 16: εικόνα σε απόχρωση του γκρι.....	56
Εικόνα 17: Αύξηση αντίθεσης με χρήση της imadjust.....	56
Εικόνα 18: Δυαδική εικόνα με χρήση της imbinarize, και Otsu's threshold.....	57
Εικόνα 19:Εικόνα με αλλαγή παρασκηνίου / προσκηνίου.....	58
Εικόνα 20: Στοιχείο Δόμησης, σχήματος Δίσκου(disc), ακτίνας 10 εικονοστοιχείων.....	59
Εικόνα 21: Εικόνα μετά την εφαρμογή του αλγόριθμου για την πλήρωση κενών imclose ...	59
Εικόνα 22: Ανίχνευση Μετώπου Ροής Με Χρήση Αλγόριθμου Sobel.....	60
Εικόνα 23: Εικόνα μετά την χρήση μορφολογικού αλγόριθμου ανοίγματος bwareaopen....	60
Εικόνα 24: Εφαρμογή του μορφολογικού αλγορίθμου Διαστολής imdilate.....	64
Εικόνα 26: Αρχική Εικόνα (από κάμερα):.....	69
Εικόνα 25: Τελική εικόνα μετά από imcrop.....	69
Εικόνα 27: Φίλτρο Gauss.....	70
Εικόνα 28: Εικόνα σε απόχρωση του γκρι.....	70
Εικόνα 29: Εικόνα με αυξημένη αντίθεση μέσω της imadjust.....	71
Εικόνα 30: Δυαδική εικόνα χρησιμοποιώντας την imbinarize και Otsus's threshold.....	71
Εικόνα 31: Αλλαγή Παρασκηνίου/ Προσκηνίου.....	72

Εικόνα 32 : Εικόνα μετά την εφαρμογή του αλγορίθμου imclose	72
Εικόνα 33: Ανίχνευση ορίων υφάσματος προς εμποτισμό με χρήση του αλγορίθμου του Sobel.....	73
Εικόνα 34: Εικόνα μετά τη χρήση της μορφολογικής λειτουργίας ανοίγματος bwareaopen	73
Εικόνα 35: Ενίσχυση περιγράμματος του υφάσματος με χρήση της boundarymask	75
Εικόνα 36: Επιφάνεια υφάσματος με χρήση της imfill	75
Εικόνα 37: Διαίρεση υφάσματος στην οριζόντια διεύθυνση.3, 6, 9, 12, οι τιμές των εικονοστοιχείων στο κάθε διάστημα	78
Εικόνα 38: Διαίρεση του υφάσματος σε 18 ξεχωριστά τμήματα. Οι αριθμοί είναι η τιμή των εικονοστοιχείων.....	80
Εικόνα 39: Διαίρεση του υφάσματος σε 18 ξεχωριστά τμήματα. Οι αριθμοί είναι η τιμή των εικονοστοιχείων.....	82
Εικόνα 40: Όριο MaXy για τον έλεγχο της ροής της ρητίνης.....	84
Εικόνα 41: Τα δύο εντοπισμένα μέτωπα ροής , και το ύφασμα που δεν έχει εμποτιστεί	91
Εικόνα 42: Τα δύο εντοπισμένα μέτωπα ροής, μέρος του υφάσματος έχει εμποτιστεί με ρητίνη από την είσοδο B	91
Εικόνα 43: Τα δύο εντοπισμένα μέτωπα ροής, μέρος του υφάσματος έχει εμποτιστεί με ρητίνη από την είσοδο A και την είσοδο B	91
Εικόνα 44: Τα δύο εντοπισμένα μέτωπα της ροής πριν ενωθούν και γίνουν ένα	92
Εικόνα 45: Τα δύο μέτωπα ροής έχουν ενωθεί, έχουμε ένα ενιαίο μέτωπο ροής	92
Εικόνα 46: Το μέτωπο ροής της ρητίνης σε μεταγενέστερο χρόνο, με μεγαλύτερο μέρος του υφάσματος εμποτισμένο	92
Εικόνα 47: Object Arduino.....	96
Εικόνα 48: Object webcam	96
Εικόνα 49: Πλακέτα Arduino UNO	107
Εικόνα 50: Ηλεκτρονικό κύκλωμα	122
Εικόνα 51: Ηλεκτρονικό σχέδιο κυκλώματος για τον έλεγχο των ηλεκτροβαλβίδων	122
Εικόνα 52: Μέρη βαλβίδας (plug valve).....	128
Εικόνα 53: Γωνιακή βαλβίδα εμβόλου.....	132
Εικόνα 54: Ομοαξονική βαλβίδα (πνευματική).....	132
Εικόνα 55: Σφαιρική Βαλβίδα.....	132
Εικόνα 56: Πενταοδική Βαλβίδα Αέρος 2 Θέσεων	134
Εικόνα 57: Υδραυλικό-Πνευματικό σχέδιο.....	135
Εικόνα 58: Υδραυλικό-Πνευματικό σχέδιο.....	135

Εικόνα 59: Εγκατάσταση Αυτοματοποιημένης Κατεργασίας VARI	140
Εικόνα 60: Ρυθμιστής Πίεσης Αέρα Βαλβίδων	141
Εικόνα 61: Επιβεβαίωση Καλής Λειτουργίας Βαλβίδων/ Οριοθέτηση Περιοχής Ενδιαφέροντος.....	141
Εικόνα 62: Σημεία Ελέγχου, Μέγιστο Όριο Εντοπισμένου Υφάσματος / Αναμονή Εντολής Χρήστη Για Συνέχεια	142
Εικόνα 63: Τα Υλικά που τοποθετούνται με διαδοχική σειρά/ πίεση κενού	143
Εικόνα 64: Διάταξη Αυτοματοποιημένης VARI Σε Αναμονή	143
Εικόνα 65: Στιγμιότυπο Διαδικασίας Αυτοματοποιημένης VARI	143
Εικόνα 66: Καθαρισμός Βαλβίδων Από Την Ρητίνη	144

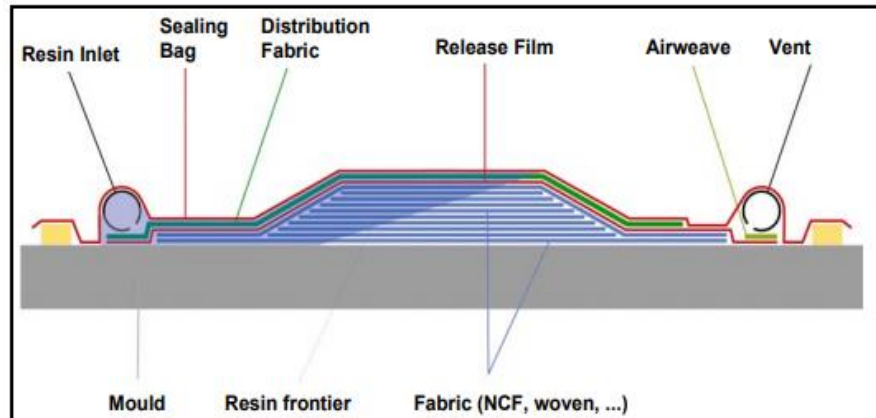
ΚΕΦΑΛΑΙΟ 1 Περιγραφή κατεργασίας VARI

1.1 Εισαγωγή

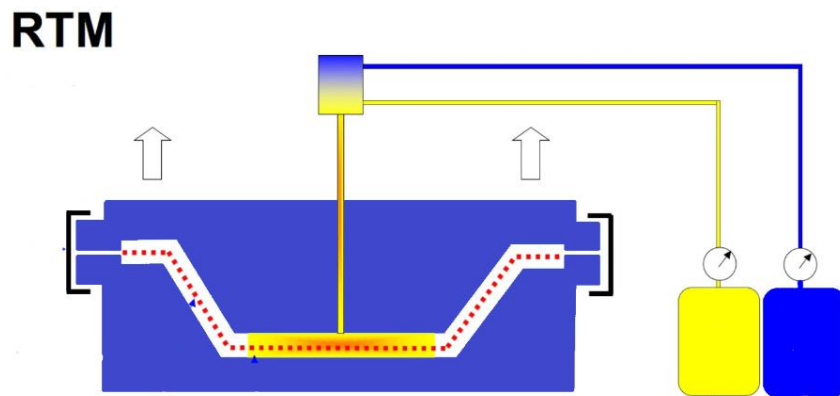
Στις μέρες μας, η χρήση σύνθετων υλικών σε διάφορες κατασκευαστικές εφαρμογές έχει αυξηθεί, καθώς πολλές από τις τεχνολογίες απαιτούν υλικά με ασυνήθιστους συνδυασμούς ιδιοτήτων, οι οποίες δεν μπορούν να επιτευχθούν με τη χρήση των συμβατικών υλικών, όπως των μετάλλων, των κεραμικών και των πολυμερών υλικών. Αρκεί να σκεφτούμε ότι για την κατασκευή ενός αεροσκάφους το ιδανικό υλικό που απαιτείται, πρέπει να έχει χαμηλή πυκνότητα αλλά ταυτόχρονα να έχει αντοχή στα κρουστικά φορτία, στην διάβρωση, στις υψηλές θερμοκρασίες καθώς και να έχει μεγάλη δυσκαμψία, ιδιότητες πολλές από τις οποίες δεν συνυπάρχουν ταυτόχρονα στα συμβατικά κράματα. Μερικά από τα πλεονεκτήματα των σύνθετων υλικών είναι η βελτίωση της δυσθραυστότητας, η αύξηση της σκληρότητας, καθώς και της θερμοκρασιακής και χημικής αντίστασης.

Υπάρχουν πολλές κατασκευαστικές μέθοδοι για την διαμόρφωση κομματιών από σύνθετα υλικά. Οι περισσότερες από αυτές χρησιμοποιούν ως μήτρα, πολυμερές, ενισχυμένο με ίνες γυαλιού, αραμιδίου ή ανθρακονήματος. Οι τεχνικές έγχυσης ρητίνης με τη χρήση υψηλής πίεσης ή με τη δημιουργία κενού, έχουν γίνει δημοφιλείς στην παραγωγή τέτοιων σύνθετων υλικών, καθώς το κόστος παραγωγής είναι αρκετά μικρό, με ποιοτικό αποτέλεσμα. Στη διεθνή βιβλιογραφία αυτές οι τεχνικές αναφέρονται ως VARTM (Vacuum Assisted Resin Transfer Moulding), VARIM (Vacuum Assisted Resin Infusion Moulding), SCRIMP (Seemann Composites Resin Infusion Moulding Process), VARI (Vacuum Assisted Resin Infusion process) καθώς και άλλες. Όλες οι τεχνικές έχουν ουσιαστικά την ίδια τεχνολογία και χρησιμοποιούν την μέθοδο εμποτισμού ξηρής ενίσχυσης με θερμοσκληρυνόμενη ρητίνη που κινείται υπό κενό.

Μια εναλλακτική τεχνική με την ίδια φιλοσοφία αλλά με κάποιες διαφορές είναι αυτή της RTM (Resin Transfer Moulding). Οι σημαντικότερες διαφορές είναι ότι στην RTM, η ρητίνη αντλείται υπό χαμηλή έως μέτρια πίεση μέσω των διόδων έγχυσης, ακολουθώντας προκαθορισμένες διαδρομές μέσω της διαμόρφωσης που υπάρχει στο καλούπι, ενώ στην VRTM η ρητίνη εκχέεται λόγω του κενού που υπάρχει στο καλούπι. Επιπλέον στην RTM το άνω μέρος του καλουπιού είναι μεταλλική πλάκα, ενώ στην VRTM χρησιμοποιείται σφραγιστική μεμβράνη. Γενικά η RTM προτείνεται για μικρά κομμάτια με πολύπλοκο καλούπι. Η μέθοδος VRTM θα περιγραφεί αναλυτικότερα αμέσως παρακάτω.



Εικόνα 1: VARI



Εικόνα 2: RTM

1.2 Μέθοδος VARI

1.2.1 Εισαγωγή

Η VARI είναι μια κατασκευαστική μέθοδος παραγωγής σύνθετων προϊόντων κλειστής διαμόρφωσης, που δεν χρησιμοποιεί αυτόκαυστο κλίβανο (Out Of Autoclave), επιτυγχάνοντας την ίδια ποιότητα με αυτό των autoclave, αλλά με διαφορετική διαδικασία. Η Spirit Aerosystems, ο μεγαλύτερος κατασκευαστής αεροσκαφών πρώτου επιπέδου η οποία μεταξύ άλλων κατασκευάζει σημαντικά κομμάτια αεροσκαφών της Boeing συμπεριλαμβανομένης και της ατράκτου του 737, υποστηρίζει ότι θα μπορούσε να παράγει αεροσκάφη και αεροπλάνα 40% ταχύτερα και με πολύ μικρότερο κόστος παρακάμπτοντας την διαδικασία του αυτόκαυστου κλίβανου για την παραγωγή προϊόντων από σύνθετα υλικά.

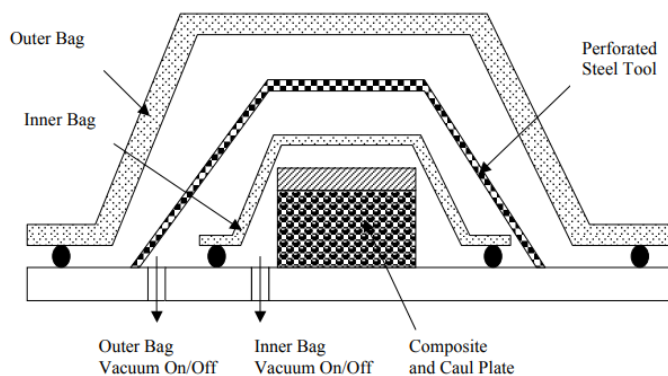
1.2.2 Περιγραφή μεθόδου

Η διαδικασία της μεθόδου VARI έχει ως εξής: Τοποθετείται το στεγνό ύφασμα προς εμποτισμό στο ανοικτό καλούπι. Στην συνέχεια τοποθετούνται κάποια υλικά τα οποία βοηθούν στην καλύτερη ροή της ρητίνης, καθώς και τη διέξοδο των πτητικών προϊόντων που έχουν δημιουργηθεί από την εξώθερμη αντίδραση της ρητίνης. Στη συνέχεια πάνω από το καλούπι τοποθετείται μια ειδική πλαστική μεμβράνη έτσι ώστε να σφραγιστεί καλά. Η τοποθέτηση της σφραγιστικής μεμβράνης γίνεται χρησιμοποιώντας ειδική ταινία στεγανοποίησης στα άκρα του καλουπιού. Το καλούπι είναι συνδεδεμένο με μια πηγή ρητίνης και με μια αντλία κενού. Χάρη στο κενό που έχει δημιουργηθεί στο καλούπι, η ρητίνη η οποία έχει αναμιχθεί με τον σκληρυντή διαχέεται στις ενισχυτικές ίνες. Η διαδικασία ολοκληρώνεται όταν ολόκληρο το ύφασμα έχει εμποτιστεί πλήρως. Κρίσιμη λεπτομέρεια, είναι να έχει διασφαλιστεί η στεγανότητα όλης της διάταξης, διότι με ύπαρξη διαρροής κενού, το πορώδες της ρητίνης δεν θα είναι μηδενικό, καθώς θα έχουν δημιουργηθεί φυσαλίδες αέρα, με συνέπεια η τελική αντοχή του τεμαχίου να μην είναι στα επιθυμητά επίπεδα. Με τη μέθοδο VARI το ποσοστό κατά όγκο των ενισχυτικών ινών στο τελικό τεμάχιο μπορεί να φτάσει έως και 70%.

1.2.3 VARI με Χρήση Διπλής Σακούλας

Μια εναλλακτική τεχνική της VARI, είναι η χρήση δύο πλαστικών μεμβρανών στεγανοποίησης (double vacuum bag). Στην συγκεκριμένη τεχνική αφού τοποθετηθεί το ύφασμα και τα παρελκόμενα του, τοποθετείται η πρώτη πλαστική μεμβράνη σφράγισης, η εσωτερική μεμβράνη, η οποία είναι σφραγισμένη γύρω από το καλούπι. Μέσα στην σφραγιστική μεμβράνη συνδέεται η αντλία κενού, καθώς και η δίοδος της πηγής της ρητίνης. Μέχρι εδώ η διαδικασία είναι ίδια με αυτή της VARI. Στη συνέχεια τοποθετείται η δεύτερη μεμβράνη σφράγισης, η εξωτερική μεμβράνη. Η εξωτερική μεμβράνη σφραγίζεται με τον ίδιο τρόπο όπως η πρώτη και συνδέεται με μια ξεχωριστή αντλία κενού. Η δίοδος της αντλίας κενού, βρίσκεται ανάμεσα στην εσωτερική και την εξωτερική μεμβράνη. Πριν τη συναρμολόγηση της εξωτερικής σφραγιστικής μεμβράνης, τοποθετείται ένα διάτρητο άκαμπτο εξάρτημα, έξω από την περίμετρο της εσωτερικής σφραγιστικής μεμβράνης, το οποίο είναι ανθεκτικό στην διαφορά πίεσης. Κατά τη διάρκεια της έγχυσης, στην εξωτερική μεμβράνη έχουμε πλήρες κενό (30 Hg), ενώ στην εσωτερική μεμβράνη έχουμε μικρότερο επίπεδο κενού (28 Hg). Λόγω της διαφοράς πίεσης η εσωτερική σακούλα φουσκώνει και πιέζει το διάτρητο άκαμπτο εργαλείο. Έτσι, τα πτητικά προϊόντα λόγω της αντίδρασης της ρητίνης με τον σκληρυντή έχουν διέξοδο και δεν δημιουργούνται κενά στο πολυστρωματικό

υλικό. Προς το τέλος της διαδικασίας εφαρμόζεται πλήρες κενό στην εσωτερική σακούλα, και οι στρώσεις του πλέγματος συνενώνονται. Η μέθοδος της διπλής σακούλας αναπτύχθηκε από τους μηχανικούς της NASA [1].



Εικόνα 3: VARI με χρήση Διπλής Σακούλας

1.2.4 Νόμος Darcy

Η ροή της ρητίνης μέσω του ξηρού υφάσματος μπορεί να περιγραφεί από τον νόμο του Darcy, για ροή νευτώνειου υγρού μέσω ενός πορώδους υλικού. Η ταχύτητα του μετώπου της ροής μπορεί να υπολογιστεί από τον τύπο:

$$v = \frac{K \cdot \Delta p}{\eta(1-\varphi) \cdot s} \quad (1.1)$$

όπου v η ταχύτητα του ρευστού σε m/s, K η διαπερατότητα του πορώδους υλικού σε m^2 , Δp η διαφορά πίεσης της εισόδου της ρητίνης με το μέτωπό της σε Pa, η το δυναμικό ιξώδες της ρητίνης με μονάδες Pa·s, φ το κλάσμα όγκου των ινών, και s η απόσταση της εισόδου της ρητίνης από το μέτωπό της σε m.

Η εξίσωση (1.1) εκφρασμένη σε ολοκλήρωμα, έχοντας αντικαταστήσει την ταχύτητα v με $\frac{ds}{dt}$ γράφεται ως εξής:

$$\int_0^{x_f} s ds = \int_0^{t_f} \frac{K \cdot \Delta p}{\eta(1-\varphi)} dt \rightarrow \frac{1}{2} x_f^2 = \frac{K \cdot \Delta p}{\eta(1-\varphi)} t_f \quad (1.2)$$

Από την παραπάνω έκφραση μπορούμε να δούμε, πως η μέγιστη εφικτή απόσταση εξαρτάται από τον χρόνο έγχυσης. Με δεδομένη τη διαπερατότητα του υφάσματος, το ιξώδες της ρητίνης και τη μέγιστη διαφορά πίεσης των 1000 hPa, η επιτυγχανόμενη απόσταση δεν είναι επαρκής για έγχυση σε μεγάλα τεμάχια. Συνήθως, οι μεγάλες κατασκευές έχουν πολύ μικρή αναλογία πάχους προς επιφάνεια, επομένως η προτεινόμενη κατεύθυνση ροής της ρητίνης για μια διαδικασία υποβοηθούμενη από κενό είναι στην κατεύθυνση του πάχους του υφάσματος, η οποία είναι μια από τις βασικές αρχές της

διαδικασίας VARI. Ο εμποτισμός του υφάσματος κατά την κατεύθυνση του πάχους, πραγματοποιείται με συγκεκριμένο ύφασμα διανομής ρητίνης στην επιφάνεια επίστρωσης. Η ποιότητα της διαδικασίας έγχυσης επηρεάζεται από πολλές παραμέτρους. Μεταβαλλόμενες παράμετροι είναι η θερμοκρασία και η πίεση. Από τον νόμο του Darcy παρατηρούμε ότι η ταχύτητα της ροής της ρητίνης είναι αντιστρόφως ανάλογη προς το ιξώδες αυτής. Επομένως η θερμοκρασία παίζει σημαντικό ρόλο, αφού επηρεάζει το ιξώδες της ρητίνης, άρα και την ταχύτητά της. Όσο αφορά την πίεση, πρέπει να γίνει αντιληπτό ότι αυτή είναι η κινητήρια δύναμη της έγχυσης της ρητίνης. Αν η ροή της ρητίνης είναι πολύ γρήγορη για να επιτευχθεί πλήρης επικάλυψη του υφάσματος, η μείωση του πλήρους κενού μειώνει την ταχύτητα της ροής, όμως η μεμβράνη κενού θα χαλάρωνε, κάτι το οποίο δεν είναι επιθυμητό. Εναλλακτικά, μπορούμε να περιορίσουμε της ροή της ρητίνης. Επίσης, η πίεση έγχυσης δεν πρέπει να πέσει κάτω από την πίεση βρασμού της ρητίνης, για την αποφυγή δημιουργίας φυσαλίδων, οι οποίες δεν θα μπορέσουν να απομακρυνθούν με τη αύξηση της πίεσης.

1.2.5 Πιλοτική διάταξη VARI Εργαστηρίου



Στη μονάδα σύνθετων υλικών του Εργαστηρίου Τεχνολογίας των Κατεργασιών της Σχολής Μηχανολόγων Μηχανικών, ο βασικός εξοπλισμός που χρησιμοποιείται είναι:

- Αντλία κενού, η οποία δημιουργεί το κενό μέσα στο καλούπι για τον εμποτισμό του υφάσματος με ρητίνη.
- Κάδος υποπίεσης για την κατακράτηση της πλεονάζουσας ρητίνης. Ο κάδος υποπίεσης παρεμβάλλεται μεταξύ της αντλίας κενού και του καλουπιού, ώστε να προστατεύει την αντλία κενού από απρόσμενη διείσδυση της ρητίνης μέσα σε αυτήν κατά την αναρρόφηση, όταν ολοκληρωθεί η κάλυψη του υφάσματος με ρητίνη και δεν προλάβουμε να σταματήσουμε τη παροχή.
- Μανόμετρο ακριβείας, για την μέτρηση της πίεσης κενού.

Τα χαρακτηριστικά του παραπάνω εξοπλισμού VARI περιγράφονται στον παρακάτω Πίνακα.

Ο καλούπι και τα παρελκόμενα του, όπως και τα υφάσματα εξαρτώνται από το προς κατασκευή τεμάχιο.

Πίνακας 1: Εξοπλισμός VARI Μονάδας Σύνθετων Υλικών του Εργαστηρίου Τεχνολογίας Των Κατεργασιών

α/α	Είδος	Χαρακτηριστικά	Εικόνα																				
1	Αντλία Κενού RC.8D	<table border="1"> <tr> <td>Κωδικός SKU</td> <td>9602021</td> </tr> <tr> <td>Κατασκευαστής</td> <td>DVP</td> </tr> <tr> <td>Αριθμός Σταδίων</td> <td>2</td> </tr> <tr> <td>Ισχύς (KW)</td> <td>0.37</td> </tr> <tr> <td>Ονομαστική Πίεση (mbar)</td> <td>0.01</td> </tr> <tr> <td>Ταχύτητα Αντλησης (m³/h)</td> <td>9.5</td> </tr> <tr> <td>Ταχύτητα Κινητήρα</td> <td>1700</td> </tr> <tr> <td>Αυτονομία Λαδιού (ml)</td> <td>350</td> </tr> <tr> <td>Διαστάσεις (mm)</td> <td>390 x 144 x 244</td> </tr> <tr> <td>Βάρος (kg)</td> <td>11.2</td> </tr> </table>	Κωδικός SKU	9602021	Κατασκευαστής	DVP	Αριθμός Σταδίων	2	Ισχύς (KW)	0.37	Ονομαστική Πίεση (mbar)	0.01	Ταχύτητα Αντλησης (m ³ /h)	9.5	Ταχύτητα Κινητήρα	1700	Αυτονομία Λαδιού (ml)	350	Διαστάσεις (mm)	390 x 144 x 244	Βάρος (kg)	11.2	
Κωδικός SKU	9602021																						
Κατασκευαστής	DVP																						
Αριθμός Σταδίων	2																						
Ισχύς (KW)	0.37																						
Ονομαστική Πίεση (mbar)	0.01																						
Ταχύτητα Αντλησης (m ³ /h)	9.5																						
Ταχύτητα Κινητήρα	1700																						
Αυτονομία Λαδιού (ml)	350																						
Διαστάσεις (mm)	390 x 144 x 244																						
Βάρος (kg)	11.2																						
2	Κάδος Υποπίεσης Κατακράτησης Ρητίνης	<table border="1"> <tr> <td>Κωδικός SKU</td> <td>RB 451</td> </tr> <tr> <td>Κατασκευαστής</td> <td>AIRTECH</td> </tr> <tr> <td>Χωρητικότητα Κάδου (lt)</td> <td>2</td> </tr> <tr> <td>Μέγιστη Θερμοκρασία Ρητίνης (°C)</td> <td>50</td> </tr> </table>	Κωδικός SKU	RB 451	Κατασκευαστής	AIRTECH	Χωρητικότητα Κάδου (lt)	2	Μέγιστη Θερμοκρασία Ρητίνης (°C)	50													
Κωδικός SKU	RB 451																						
Κατασκευαστής	AIRTECH																						
Χωρητικότητα Κάδου (lt)	2																						
Μέγιστη Θερμοκρασία Ρητίνης (°C)	50																						
3	Μανόμετρο Ακριβείας	<table border="1"> <tr> <td>Κωδικός SKU</td> <td>VAC-GAUGE 30</td> </tr> <tr> <td>Κατασκευαστής</td> <td>AIRTECH</td> </tr> <tr> <td>Εύρος (bar)</td> <td>0-(-1)</td> </tr> <tr> <td>Διαβάθμιση (bar)</td> <td>-0.02</td> </tr> <tr> <td>Σπείρωμα (inch)</td> <td>1/4</td> </tr> </table>	Κωδικός SKU	VAC-GAUGE 30	Κατασκευαστής	AIRTECH	Εύρος (bar)	0-(-1)	Διαβάθμιση (bar)	-0.02	Σπείρωμα (inch)	1/4											
Κωδικός SKU	VAC-GAUGE 30																						
Κατασκευαστής	AIRTECH																						
Εύρος (bar)	0-(-1)																						
Διαβάθμιση (bar)	-0.02																						
Σπείρωμα (inch)	1/4																						

ΚΕΦΑΛΑΙΟ 2 Αυτοματοποίηση VARI

2.1 Εισαγωγή

Στις κατεργασίες VARI η γνώση του μετώπου ροής είναι πολύ σημαντική, ειδικά όταν έχουμε να κάνουμε με μεγάλες κατασκευές όπου το πάχος μπορεί αλλάζει. Ένα κρίσιμο θέμα της για την επιτυχία της κατεργασίας VARI είναι να εξασφαλίσουμε ότι η ρητίνη θα διαβρέξει όλη την επιφάνεια του υφάσματος. Έτσι έχουν αναπτυχθεί κάποιες τεχνικές, με τις οποίες μπορεί να εντοπιστεί το μέτωπο της ροής της ρητίνης, έτσι ώστε να εξάγουμε συμπεράσματα για τον τρόπο που αναπτύσσεται το μέτωπο, και με βάση αυτό να βελτιστοποιήσουμε την διαδικασία έγχυσης, με καλύτερο σχεδιασμό των θέσεων των διόδων έγχυσης της ρητίνης, ή της διόδου αναρρόφησης καθώς και να ελέγξουμε την διαδικασία έγχυσης ρητίνης. Οι τεχνικές αυτές χρησιμοποιούν ειδικούς αισθητήρες για τον εντοπισμό του μετώπου της ρητίνης, οι οποίοι θα περιγραφούν παρακάτω.

2.2 Αισθητήρες Εντοπισμού Μετώπου Ροής

Αισθητήρες πίεσης

Υπάρχουν δύο ειδών αισθητήρες: πιεζοηλεκτρικοί και χωρητικοί, οι οποίοι ενσωματώνονται στα φύλλα των σύνθετων ινών και μπορούν να μετρήσουν την πίεση σε πραγματικό χρόνο. Μια αλλαγή πίεσης δείχνει την αρχή έγχυσης της ρητίνης. Οι πιεζοηλεκτρικοί αισθητήρες είναι σε θέση να παρακολουθήσουν την πίεση κενού στην εγκατάσταση των σύνθετων πολυστρωματικών ινών, καθώς και την άφιξη της ρητίνης σε αυτούς. Επιπλέον, μπορεί να μετρηθεί η αύξηση της πίεσης της ρητίνης λόγω έμφραξης της εισόδου του αισθητήρα από τη ρητίνη. Αντίθετα οι χωρητικοί αισθητήρες δεν έχουν αυτή την ικανότητα [2].

Η λειτουργία της συγκεκριμένης τεχνικής για την εύρεση του μετώπου ροής είναι η εξής. Η δεξαμενή ρητίνης είναι τοποθετημένη σε συνθήκες ατμοσφαιρικής πίεσης. Όταν η ρητίνη διαβρέχει το ύφασμα των σύνθετων ινών, τότε παρατηρείται σταδιακή αύξηση της πίεσης στο σημείο που εμποτίζεται. Ο αισθητήρας που βρίσκεται πιο κοντά στο σημείο εμποτισμού εμφανίζει ταχύτερη αύξηση της πίεσης σε σχέση με τους άλλους. Όταν η ρητίνη εμποτίσει τον αισθητήρα τότε η πίεση αυξάνεται απότομα [3]. Έτσι γνωρίζοντας την πραγματική πίεση από τους αισθητήρες και χρησιμοποιώντας την εξίσωση του Darcy για ασυμπίεστο ρευστό:

$$(\nabla \cdot v = 0 \Rightarrow \nabla \cdot \left(\frac{K \cdot \Delta p}{\eta(1-\phi) \cdot s} \right) = 0 \quad (2.1)$$

με υπολογιστικές επαναλήψεις βρίσκουμε την πραγματική θέση του μετώπου της ροής [4]. Οι αισθητήρες πίεσης έχουν υψηλό λόγο σήματος προς θόρυβο (SNR), αλλά μειονεκτούν στην τοπική ενίσχυση, και παρεμβάλλονται στη ροή της ρητίνης.

Αισθητήρες Αγωγιμότητας

Οι αισθητήρες αγωγιμότητας είναι ένα πλέγμα αγώγιμων νημάτων το οποίο προσαρμόζεται στο καλούπι. Καθώς αρχίζει η έγχυση της ρητίνης, τα κενά μεταξύ των επιπέδων νημάτων είναι γεμάτα με την αγώγιμη ρητίνη, η οποία δημιουργεί ένα κλειστό ηλεκτρικό κύκλωμα. Τα όργανα που είναι συνδεδεμένα ανιχνεύουν το σήμα και υπολογίζουν την τοποθεσία της ρητίνης, καθώς και την κατάσταση θεραπείας της (curing).

Συνήθως χρησιμοποιούνται δύο ειδών αισθητήρες, μικρές βέργες από άνθρακα καθώς και σύρματα χαλκού μονωμένα με γυαλί. Ο αισθητήρας από ανθρακόνημα έχει το πλεονέκτημα ότι είναι συμβατός με το μακροσκοπικό περιβάλλον των ελασμάτων και ιδανικός για την απεικόνιση ροής της ρητίνης μέσω του καλουπιού. Όσον αφορά την παρακολούθηση της θεραπείας, λόγω της σχετικής αντίστασης που δημιουργείται, τα δεδομένα τάσης /χρόνου μπορεί να έχουν επηρεαστεί αρνητικά επομένως τα αποτελέσματα για την σκλήρυνση της ρητίνης πιθανόν να είναι λανθασμένα. Αντιθέτως οι αισθητήρες χαλκού έχουν σχεδόν μηδενική αντίσταση στη ροή της ρητίνης, επομένως δεν επηρεάζονται τα δεδομένα τάσης/χρόνου, και τα συμπεράσματα για τα στάδια σκλήρυνσης της ρητίνης είναι αξιόπιστα.

Οι αισθητήρες αγωγιμότητας, μπορούν να εντοπίζουν την παραμικρή αλλαγή σήματος, σε πολλαπλά σημεία στο καλούπι, αλλά μειονεκτούν στο ότι έχουν χαμηλό λόγο σήματος προς θόρυβο. Επίσης όπως είναι λογικό, οι αισθητήρες αγωγιμότητας δεν μπορούν να χρησιμοποιηθούν σε αγώγιμα μέσα όπως τα ανθρακονήματα [5][6].

Γραμμικοί Αισθητήρες Συνεχούς Ρεύματος (Linear Direct Current Sensing)

Οι γραμμικοί αισθητήρες συνεχούς ρεύματος έχουν περίπου την ίδια λογική με τους αισθητήρες αγωγιμότητας. Η μεγάλη διαφορά είναι ότι μπορεί να εντοπιστεί το μέτωπο της ροής σε όλο το μήκος των δύο αγωγών που τοποθετούνται και όχι μόνο σε διακριτά σημεία. Η φυσική αρχή των LDC αισθητήρων είναι βασισμένη στην εξάρτηση που υπάρχει μεταξύ της τάσης εξόδου και του μετώπου της ροής. Οι δύο αγωγοί τοποθετούνται παράλληλα μεταξύ τους κατά μήκος του καλουπιού. Η τάση εξόδου δίνεται από την σχέση:

$$V_{out}(t) = \frac{R_f}{R_{in}} P V_{in} \quad (2.2)$$

όπου η αντίσταση R_{in} μεταξύ των δύο αγωγών υπολογίζεται από τη σχέση:

$$R_{in} = \rho \frac{D}{hx_f(t)(1-\Phi)} \quad (2.3)$$

όπου ρ η ειδική αντίσταση της ρητίνης, D η απόσταση μεταξύ των δύο αγωγών, h το ύψος των δύο αγωγών, Φ το ποσοστό των ινών κατά όγκο και $x_f(t)$ η θέση του μετώπου της ροής. R_f μεταβλητή αντίσταση του κυκλώματος, P συντελεστής ενίσχυσης του κυκλώματος. Επομένως η θέση του μετώπου της ροής δίνεται από τη σχέση [7]:

$$x_f(t) = V_{out}(t) \frac{\rho D}{R_f P h (1-\Phi) V_{in}} \quad (2.4)$$

Στο καλούπι μπορούν να τοποθετηθούν περισσότερα ζεύγη αγωγών, έτσι ώστε να έχουμε μεγαλύτερη λεπτομέρεια για το προφίλ του μετώπου της ρητίνης. Η χρήση των αισθητήρων αυτών έχει αρκετά πλεονεκτήματα, όπως το ότι δεν εμποδίζουν την ροή της ρητίνης και το κόστος τους είναι χαμηλότερο από αυτό άλλων μεθόδων όπως αυτή των οπτικών ινών. Παρόλα αυτά δεν είναι εύκολο να χρησιμοποιηθούν σε πολύπλοκα καλούπια. Επίσης, όπως συμβαίνει και με τους αισθητήρες αγωγιμότητας, οι LDC αισθητήρες δεν μπορούν να χρησιμοποιηθούν σε αγωγίμα μέσα όπως τα ανθρακονήματα. Τέλος, λόγω του ότι η ειδική αντίσταση της ρητίνης είναι συνάρτηση του χρόνου, της θερμοκρασίας και της θέσης της, πρέπει να προηγηθούν αρκετές δοκιμές ώστε να γίνει σωστή παραμετροποίηση του ηλεκτρικού κυκλώματος [7].

Διηλεκτρικοί Αισθητήρες

Η εφαρμογή των διηλεκτρικών αισθητήρων είναι αρκετά εύκολη, αλλά θέτει το πρόβλημα της εντοπισμένης ανίχνευσης και επεμβατικότητας. Ο διηλεκτρικός αισθητήρας αποτελείται από δύο παράλληλα επίπεδα ηλεκτρόδια χαλκού ενσωματωμένα ανάμεσα σε δύο στρώσεις μιας πολυμερούς μεμβράνης. Ένα πολύ λεπτό στρώμα κόλλας υπάρχει μεταξύ των ηλεκτροδίων και του πολυμερούς φιλμ και ανάμεσα στα δύο στρώματα της πολυμερούς μεμβράνης. Ανάμεσα στα δύο ηλεκτρόδια υπάρχει παγιδευμένος αέρας. Η μία πλευρά του αισθητήρα είναι καλυμμένη από μεταλλικό φύλλο. Συνεπώς, όταν εφαρμόζεται μια τάση μεταξύ των δύο ηλεκτροδίων, δημιουργείται ηλεκτρικό πεδίο μόνο από την μία πλευρά του αισθητήρα. Όταν αρχίσει η έγχυση της ρητίνης, ο αέρας που είναι ανάμεσα στα δύο ηλεκτρόδια αντικαθίσταται από αυτήν, και η ρητίνη συμπεριφέρεται ως διηλεκτρικό μέσο, με αποτέλεσμα να αλλάζει η χωρητικότητα του πυκνωτή. Ο αισθητήρας τοποθετείται παράλληλα στην αναμενόμενη κατεύθυνση της ροής. Η ανάλυση της ηλεκτρικής απόκρισης μπορεί να γίνει χρησιμοποιώντας ένα ηλεκτρικό κύκλωμα ισοδύναμο με τον αισθητήρα.

Τα επιμέρους τμήματα του αισθητήρα που αναφέρθηκαν παραπάνω συμβάλλουν το καθένα ξεχωριστά ως σύνθετες αντιστάσεις. Το πολυμερικό φιλμ είναι συνδεδεμένο σε σειρά με

την κόλλα, και όλο το στοιχείο αυτό παράλληλα με τον εγκλωβισμένο αέρα. Ο αισθητήρας μπορεί να θεωρηθεί ότι αποτελείται από παράλληλα ηλεκτρικά κυκλώματα εμποτισμένων από ρητίνη και στεγνών περιοχών, και το κάθε επιμέρους κύκλωμα να αναλυθεί ως κύκλωμα με στοιχεία σε σειρά. Έτσι η αλλαγή της χωρητικότητας εκφραζόμενη ως ηλεκτρική τάση δίνει τη θέση του μετώπου της ρητίνης. Η εξίσωση που προκύπτει από την ανάλυση του κυκλώματος είναι:

$$Z_{sensor} = \frac{Z_s^2 + Z_s Z_w + Z_s Z_d + Z_d Z_w}{l z_s + l_w z_d + l z_w - l_w z_w} \quad (2.5)$$

όπου Z_s είναι σύνθετη αντίσταση που προκύπτει από το πολυμερικό φιλμ, την κόλλα και τον αέρα, $z_s = \left(\frac{1}{z_f + z_g} + \frac{1}{z_a}\right)^{-1}$, l το συνολικό μήκος του αισθητήρα, και l_w το μήκος της εμποτισμένης περιοχής.

Επιπλέον μετά το πέρας της έγχυσης της ρητίνης, με τους διηλεκτρικούς αισθητήρες μπορούμε να παρακολουθήσουμε μακροσκοπικά την εξέλιξη της σκλήρυνσης της ρητίνης [8].

Αισθητήρας Τάσης

Οι αισθητήρες τάσης χρησιμοποιούνται για την εύρεση της ροής της ρητίνης, αλλά και για την παρακολούθηση της εξέλιξης της σκλήρυνσης της. Υπάρχουν δυο ειδών αισθητήρες τάσεις, οι σημειακοί και οι γραμμικοί [9]. Το ηλεκτρικό κύκλωμα που χρησιμοποιείται και στα δύο είδη αισθητήρων είναι ίδιο και έχει δύο αντιστάσεις. Η πρώτη αντίσταση χρησιμεύει ως διαιρέτης τάσης ενώ η δεύτερη, είναι η αντίσταση που δημιουργείται από το πολυμερικό μέσο μεταξύ των 2 ηλεκτροδίων του κυκλώματος. Η αρχική τάση μεταξύ των δύο σημείων λόγω της επαφής με το ύφασμα εξαρτάται από το είδος του υφάσματος που χρησιμοποιούμε, πχ για το υαλόνημα η τάση είναι μηδενική. Όταν η ρητίνη αρχίσει να διαχέεται, τότε το κύκλωμα κλείνει και η τάση εξόδου αυξάνει.

Στους αισθητήρες σημειακής τάσης, τα ηλεκτρόδια είναι πολύ κοντά το ένα με το άλλο και η ρητίνη ακουμπάει την άκρη αυτών. Αντίθετα στους γραμμικούς αισθητήρες τάσης, τα ηλεκτρόδια είναι ένα ζεύγος παράλληλων γυμνών καλωδίων, και η ρητίνη καλύπτει το μήκος αυτών. Αν και οι γραμμικοί αισθητήρες δίνουν αρκετή πληροφόρηση για το μέτωπο της ρητίνης, χάνουν στο ότι μπορούν να εντοπίσουν το ποσοστό κάλυψης της ρητίνης και όχι τα τμήματα που είναι εμποτισμένα, με αποτέλεσμα, αν υπάρχουν πολλαπλά σημεία επικάλυψης της ρητίνης, να δώσει λανθασμένη πληροφόρηση για τη θέση του μετώπου. Επίσης μπορούν μεν να χρησιμοποιηθούν μια φορά, αλλά και το κόστος κατασκευής τους είναι πολύ μικρό. Για την αντιμετώπιση της μη σωστής πληροφορίας για τη θέση της

ρητίνης, έχουν αναπτυχθεί οι περιτυλιγμένοι γραμμικοί αισθητήρες όπου τα ηλεκτρόδια είναι τοποθετημένα σε σπειροειδή παράλληλη διάταξη. Παρόλα αυτά και αυτοί δεν μπορούν χρησιμοποιηθούν πάνω από μια φορά καθώς το κύκλωμα είναι τυπωμένο, και οι επαφές είναι πολύ λεπτές. Οι αισθητήρες σημειακής τάσης, ξεπερνούν τα παραπάνω προβλήματα. Αποτελούνται από έναν χάλκινο σωλήνα και μια χάλκινη ράβδο ομόκεντρα τοποθετημένη και λειτουργούν ως δύο ξεχωριστά ηλεκτρόδια. Δύο πλαστικοί σωλήνες κατασκευασμένοι από πολυοξυμεθυλένιο διαχωρίζουν τον χάλκινο σωλήνα από την χάλκινη ράβδο. Το μέγεθος των αισθητήρων αυτών είναι αρκετά μικρό: 9.5 mm.

Στη διάταξη χρησιμοποιούνται πολλοί σημειακοί αισθητήρες τάσης για την καλύτερη εύρεση του μετώπου τα ρητίνης. Ένα μειονέκτημα του συγκεκριμένου αισθητήρα είναι ότι είναι ευαίσθητος στον ηλεκτρικό θόρυβο, και η τάση του αισθητήρα είναι δύσκολο να εντοπιστεί από το σύστημα πρόσκτησης δεδομένων, εάν η ρητίνη έχει χαμηλή αγωγιμότητα. Για αυτό τον λόγο χρησιμοποιείται ένα κύκλωμα ενίσχυσης το οποίο αποτελείται από ένα τελεστικό ενισχυτή απομόνωσης (buffer) όπως ο OPA655P, έναν τελεστικό τελεστή ενίσχυσης, όπως ο LM358P, δύο πυκνωτές των 1 Mf και ένα ποτενσιόμετρο. Με την κατάλληλη συνδεσμολογία, το σήμα ενισχύεται [10].

Αισθητήρες Ανακλαστικότητας στον Χρόνο (Time domain Reflectometry sensors)

Ο E-TDR αισθητήρας μετρά τις αντανακλάσεις ενός σήματος κατά μήκος του καλοπιού. Για να μετρήσει αυτές τις αντανακλάσεις, ένας πομπός θα μεταδώσει ένα σήμα συγκεκριμένης συχνότητας με κατεύθυνση το επίπεδο ροής της ρητίνης, και θα ακούσει τις αντανακλάσεις. Ο E-TDR αισθητήρας αναλύει τις μεταβολές στην απόκριση της γραμμής μετάδοσης κατά τη διάρκεια της ροής της ρητίνης.

Η γραμμή μετάδοσης τοποθετείται ανάμεσα στο καλούπι και στο σύνθετο ύφασμα. Αυτή αποτελείται από μια συνεχή δομή αντιστάσεων, επαγωγών, πυκνωτών και αγωγών. Όταν το μέτωπο της ρητίνης βρέξει τον αισθητήρα, οι διηλεκτρικές ιδιότητες του αλλάζουν, προκαλώντας ανάκλαση του ηλεκτρικού σήματος. Το σήμα αξιολογείται στο πεδίο του χρόνου με τη χαρτογράφηση των περιοχών της ρητίνης στον αισθητήρα. Ο αισθητήρας E-TDR στέλνει ένα σήμα τάσης ανά picosecond στη γραμμή μετάδοσης, και μετρά την τάση ανάκλασης, ύστερα υπολογίζεται ο συντελεστής ανάκλασης τάσης:

$$\rho = \frac{E_r}{E_i} = \frac{Z_d - Z_0}{Z_d + Z_0} \quad (2.6)$$

όπου E_i η αρχική τάση, E_r η αντανακλώμενη τάση, Z_0 και Z_d οι σύνθετες αντιστάσεις πριν και μετά την έγχυση ρητίνης αντίστοιχα. Για τον υπολογισμό της θέσης της ρητίνης

χρησιμοποιείται ένας αλγόριθμος κατωφλιού. Υπολογίζεται η ταχύτητα του ηλεκτρομαγνητικού κύματος που ανακλάται, πολλαπλασιάζεται με τον χρόνο και βρίσκεται η ακριβής θέση του μετώπου της ρητίνης. Η συχνότητα του σήματος που εκπέμπεται είναι στην περιοχή των μικροκυμάτων. Τα πλεονεκτήματα των E-TDR αισθητήρων είναι ότι μπορούν να εντοπίσουν την συμπεριφορά της ρητίνης με ευκρίνεια 3mm σε ένα μήκος γραμμής μετάδοσης αρκετών μέτρων, είναι αξιόπιστοι για τον εντοπισμό πολλαπλών θέσεων καθώς και πολλαπλών ροών. Ένα μειονέκτημα είναι ότι ο εξοπλισμός που απαιτείται είναι αρκετά δαπανηρός [11][12].

Αισθητήρες Υπερήχων

Χρησιμοποιώντας τους αισθητήρες υπερήχων δίνεται η δυνατότητα να εντοπιστεί το μέτωπο της ροής κατά την διαδικασία της έγχυσης, αλλά και η παρακολούθηση της συμπεριφοράς σκλήρυνσης της ρητίνης. Οι αισθητήρες υπερήχων έχουν το πλεονέκτημα ότι κατά την διάρκεια της κατεργασίας δεν χρειάζεται επαφή με το κομμάτι. Έτσι, το τελικό κομμάτι δεν εμπεριέχει κανέναν αισθητήρα, ούτε σημάδια από αυτούς. Αφού τοποθετηθεί ο αισθητήρας υπερήχων στο καλούπι, μπορούν να επαναληφθούν εκατοντάδες κύκλοι επεξεργασίας με τον ίδιο αισθητήρα.

Για την παραγωγή ακουστικού κύματος, χρησιμοποιείται ένα πιεζοηλεκτρικό στοιχείο. Παλμοί υψηλής τάσης διεγείρουν το πιεζοηλεκτρικό στοιχείο, προκαλώντας την ταλάντωση του στα 5 MHz. Τη στιγμή που το ηχητικό κύμα φτάσει στο όριο, μερικά από τα κύματα απορροφώνται, ενώ ένα μέρος από αυτά ανακλώνται πίσω. Όταν το ακουστικό κύμα φτάσει στο πιεζοηλεκτρικό στοιχείο, τότε παράγεται τάση, δηλ. το σήμα του αισθητήρα.

Υπάρχουν δύο μέθοδοι χρήσης υπερήχων: η άμεση ανάκλαση, και η άμεση μετάδοση. Η τεχνική της άμεσης ανάκλασης χρησιμοποιεί έναν αισθητήρα υπερήχων, ο οποίος είναι ταυτόχρονα πομπός και δέκτης. Η τεχνική της άμεσης μετάδοσης χρησιμοποιεί δύο αισθητήρες υπερήχων οι οποίοι τοποθετούνται αντιδιαμετρικά, από τους οποίους ο ένας είναι ο πομπός και ο δεύτερος ο δέκτης. Η δεύτερη τεχνική προτιμάται περισσότερο στις κατεργασίες RTM.

Αυτή η τεχνολογία έχει το πλεονέκτημα ότι δεν καταγράφονται ενοχλητικά ηχητικά κύματα λόγω της ηχούς. Η ταχύτητα του ηχητικού κύματος υπολογίζεται από τις πληροφορίες χρόνου και απόστασης που έχουν ληφθεί. Επίσης μετράται και η μεταβολή της εξασθένησης του μεταδιδόμενου σήματος [13]. Η άφιξη του μετώπου της ρητίνης εντοπίζεται από το εκπεμπόμενο σήμα των αισθητήρων υπερήχων. Επειδή μετάδοση υπερηχητικού κύματος συχνότητας 5 MHz ή και μικρότερης διαμέσου του υφάσματος δεν είναι εφικτή, ένα ανακλώμενο σήμα ίδιου πλάτους με το αρχικό θα εντοπιστεί στον αισθητήρα υπερήχων.

Κατά την διάρκεια της σκλήρυνσης της ρητίνης, η ταχύτητα του ήχου αυξάνεται και φτάνει σε μέγιστη τιμή κατά την τελική σκλήρυνση του τεμαχίου, καθώς το ιξώδες αυξάνεται. Επομένως μπορούμε να παρατηρούμε τα στάδια σκλήρυνσης της ρητίνης με τους αισθητήρες υπερήχων. Κατά το τελικό στάδιο της κατεργασίας, μέσω των αισθητήρων υπερήχων μπορούμε να υπολογίσουμε το μέτρο ελαστικότητας του κομματιού, αφού γνωρίζουμε την διαμήκη ταχύτητα του ήχου : $E = \rho \cdot c_{long}^2$ [14],[15].

Αισθητήρες Οπτικών Ινών

Η οπτική ίνα αποτελείται από τον πυρήνα ο οποίος είναι συνήθως από πυρίτιο με συντελεστή διάθλασης $n_{core} \approx 1.457$ και από τον μανδύα ο οποίος έχει πάντα λίγο μικρότερο δείκτη διάθλασης από αυτόν του πυρήνα ≈ 1.455 . Ο δείκτης διάθλασης του μανδύα είναι μικρότερος έτσι ώστε να έχουμε ολική ανάκλαση για αποφυγή απώλειας του σήματος. Εάν αφαιρεθεί τοπικά ένα μέρος του μανδύα από την οπτική ίνα (1mm), θα έχουμε ως αποτέλεσμα την εξασθένηση του μεταφερόμενου σήματος. Αυτό συμβαίνει αφού το σήμα θα εξέρχεται από τον πυρήνα στο σημείο αυτό καθώς η διαφορά των δεικτών διάθλασης είναι μεγάλη, και δεν δημιουργείται ολική ανάκλαση.

Οι εποξικές ρητίνες που χρησιμοποιούνται στις μεθόδους VARI έχουν συνήθως μεγαλύτερο δείκτη διάθλασης από αυτό του πυριτίου [6]. Αυτό ακριβώς το χαρακτηριστικό εκμεταλλευόμαστε και χρησιμοποιούμε τις οπτικές ίνες ως αισθητήρες για την εύρεση του μέτωπου της ρητίνης. Η οπτική ίνα τοποθετείται ανάμεσα στα επίπεδα των υφασμάτων. Μια πηγή laser τοποθετείται στην μια άκρη της οπτικής ίνας, ενώ στην άλλη άκρη έχει συνδεθεί φωτοδίοδος η οποία εντοπίζει το μεταφερόμενο σήμα. Κατά την διάρκεια της κατεργασίας, ένα σήμα φωτός μεταδίδεται από την μια άκρη στην άλλη μέσω της οπτικής ίνας. Η ένταση του μεταδιδόμενου σήματος είναι μεγάλη πριν η ρητίνη φτάσει στο σημείο της οπτικής ίνας όπου έχει αφαιρεθεί ο μανδύας. Όταν η ρητίνη πλησιάσει το σημείο αυτό και το επικαλύψει, τότε η ισχύς του σήματος μειώνεται. Αυτό συμβαίνει αφού ο δείκτης διάθλασης της ρητίνης είναι πολύ κοντά, αλλά ελάχιστα μεγαλύτερος από αυτόν του πυρήνα, έτσι το φως διαθλάται και το σήμα αλλοιώνεται. Με αυτόν τον τρόπο εντοπίζεται το μέτωπο της ρητίνης. Στην οπτική ίνα μπορούμε να αφαιρέσουμε σε περισσότερα από ένα σημεία τον μανδύα έτσι ώστε να χρησιμοποιηθεί σαν πολλαπλός αισθητήρας εύρεσης ροής, έτσι κάθε φορά που το μέτωπο της ροής καλύπτει τα σημεία αυτά το σήμα εξασθενεί περισσότερο και μπορούμε να εντοπίζουμε το μέτωπο της ρητίνης κάθε φορά [16].

Μια διαφορετική μέθοδος που είναι χρήσιμη σε μεγάλα καλούπια, είναι αυτή με την οποία μαζί με τις οπτικές ίνες χρησιμοποιείται φίλτρο Bragg οπτικής ίνας (Fiber Bragg grating). Οι

οπτικές ίνες με γυμνά σημεία μανδύα τοποθετούνται παράλληλα κατά μήκος του καλουπιού σε συγκεκριμένες θέσεις. Σε κάθε οπτική ίνα έχει τοποθετηθεί FBG φίλτρο με διαφορετικά κεντρικά μήκη κύματος, χρησιμοποιώντας αντάπτορες FC/APC. Όλες οι οπτικές ίνες είναι συνδεδεμένες με σύστημα FBGSLI για την ανάλυση των σημάτων. Πριν από την έγχυση της ρητίνης όλο το FBG σήμα ανιχνεύεται, καθώς η ρητίνη εκχύνεται και πλησιάζει την πρώτη οπτική ίνα, το FBG σήμα παύει να εμπεριέχει το σήμα με μήκος κύματος λ_1 [6]. Καθώς προχωράει η διαδικασία και η ρητίνη επικαλύπτει και τις άλλες οπτικές ίνες, τα σήματα με τα αντίστοιχα μήκη κύματος εξαφανίζονται.

Με τη χρήση των αισθητήρων οπτικών ινών μπορούμε να ελέγχουμε και τη διαδικασία σκλήρυνσης της ρητίνης, καθώς ο συντελεστής διάθλασης n αλλάζει, άρα και το πλάτος του μεταφερόμενου σήματος [17].

Αισθητήρες Μικροθερμοστοιχείων

Το θερμοστοιχείο αποτελείται από δύο διαφορετικού υλικού αγωγούς ενωμένους στο ένα τους άκρο. Όταν το ενωμένο άκρο θερμανθεί, τότε αναπτύσσεται μικρή διαφορά τάσης. Αν η θερμοκρασία αναφοράς διατηρείται σταθερή, η τάση είναι γραμμικά ανάλογη με τη θερμοκρασία της ένωσης των αγωγών, $v = a(T_{hot} - T_{ref})$, όπου a ο συντελεστής Seebeck. Τα θερμοστοιχεία τοποθετούνται στο κάτω μέρος του καλουπιού σε διακριτές θέσεις χωρίς να προεξέχουν περισσότερο από το επίπεδο του καλουπιού. Προτού ξεκινήσει η έγχυση της ρητίνης, έχει προηγηθεί θέρμανση της συνήθως στους 80°C για θερμοκρασία περιβάλλοντος 26°C, έτσι ώστε να υπάρχει επαρκής θερμοκρασιακή διαφορά μεταξύ ρητίνης και μεταλλικού καλουπιού [18]. Για καλύτερα αποτελέσματα έχουν προηγηθεί προσομοιώσεις σε διάφορες συνθήκες έτσι ώστε να επιλεχτεί η κατάλληλη θερμοκρασία θέρμανσης τη ρητίνης, καθώς όταν η ρητίνη έρχεται σε επαφή με το μεταλλικό καλούπι, έχουμε μια μεγάλη πτώση θερμοκρασίας ως αποτέλεσμα το θερμοστοιχείο να μην αντιληφθεί την άφιξη αυτής [19]. Σημαντικές παράμετροι για την επιτυχία του συστήματος με θερμοστοιχεία είναι οι εξής: το καλούπι να είναι από υλικό με μικρή θερμική αγωγιμότητα, ο ρυθμός ροής της ρητίνης να είναι όσο των δυνατών μεγαλύτερος, η θερμοκρασιακή διαφορά της ρητίνης με τα τοιχώματα του καλουπιού να είναι αρκετά μεγάλη και η ειδική θερμότητα της ρητίνης να είναι μεγάλη [18].

Κατά τη έγχυση της ρητίνης, όταν αυτή φτάσει σε ένα θερμοστοιχείο, δημιουργείται τάση της τάξεως των millivolts και εντοπίζεται το μέτωπο αυτής. Επίσης έχει αναπτυχθεί και αντίστροφη μέθοδος, όπου η ρητίνη παραμένει σε θερμοκρασία περιβάλλοντος και αντί για αυτή θερμαίνονται τα τοιχώματα καλουπιού. Τα θερμοστοιχεία μπορούν να χρησιμοποιηθούν και για την παρακολούθηση της σκλήρυνσης της ρητίνης, αφού η

αντίδραση είναι εξώθερμη και η θερμοκρασία αυξάνεται [10]. Μετά το πέρας της διαδικασίας σκλήρυνσης, ο ρυθμός αύξησης της θερμοκρασία μειώνεται.

Χρήση Κάμερας Και Υπολογιστικής Όρασης

Με τη χρήση κάμερας και των εργαλείων που μας προσφέρει η υπολογιστική όραση μπορούμε να εντοπίσουμε το μέτωπο της επιφανειακής ροής. Αυτή η τεχνική, η οποία υιοθετήθηκε στην παρούσα εργασία θα αναλυθεί στο επόμενο κεφάλαιο.

ΚΕΦΑΛΑΙΟ 3 Υπολογιστική Όραση.

3.1 Εισαγωγή

Για την ανίχνευση του μετώπου ροής της ρητίνης χρησιμοποιούμε μια κάμερα διαδικτύου για την λήψη στιγμιότυπων κατά τη διάρκεια της έγχυσης της ρητίνης και το περιβάλλον MATLAB με τις εργαλειοθήκες Image Processing Toolbox και Computer Vision System Toolbox, οι οποίες διαθέτουν εντολές με τις οποίες μπορούμε να επεξεργαστούμε μια εικόνα. Οι μέθοδοι που έχουμε χρησιμοποιήσει καθώς και ο κώδικας που έχουμε δημιουργήσει για την ανίχνευση του μετώπου της ροής της ρητίνης θα περιγραφεί αναλυτικά παρακάτω, αφού πρώτα κάνουμε μια εισαγωγική ανασκόπηση πάνω στην επεξεργασία εικόνας.

3.2 Ψηφιακή Εικόνα

Μια ψηφιακή εικόνα μπορεί να οριστεί ως μια δισδιάστατη συνάρτηση $f(x,y)$, όπου τα x και y είναι χωρικές επίπεδες συντεταγμένες, και το πλάτος της συνάρτησης f σε οποιοδήποτε ζεύγος (x,y) εκφράζει την ένταση της φωτεινότητας της εικόνας στο σημείο αυτό. Τα διακριτά σημεία ονομάζονται εικονοστοιχεία (pixel).

Μια ψηφιακή εικόνα μπορεί να είναι δυαδική (binary image), μονοχρωματική αποχρώσεων του γκρι (gray-level image), έγχρωμη (color image) ή ενδεικτική εικόνα (indexed image).

3.2.1 Δυαδική Εικόνα

Στην δυαδική εικόνα, κάθε εικονοστοιχείο είναι μόνο μαύρο ή άσπρο. Δεδομένου ότι υπάρχουν μόνο δύο πιθανές τιμές για κάθε εικονοστοιχείο χρειαζόμαστε μόνο ένα bit ανά εικονοστοιχείο. Στο MATLAB η δυαδική εικόνα είναι ένας λογικός πίνακας και όχι ένας πίνακας τάξης uint8 με τιμές 0 και 1.

3.2.2 Εικόνα Γκρι Απόχρωσης

Στην εικόνα γκρι απόχρωσης, κάθε εικονοστοιχείο είναι μια απόχρωση του γκρι, συνήθως από 0 που είναι το μαύρο, έως 255 που είναι το λευκό. Αυτό το εύρος σημαίνει ότι κάθε εικονοστοιχείο μπορεί να αναπαρασταθεί από οκτώ bits.

3.2.3 Έγχρωμη Εικόνα

Στην έγχρωμη εικόνα, κάθε εικονοστοιχείο έχει ένα συγκεκριμένο χρώμα το οποίο περιγράφεται από το ποσό του κόκκινου, πράσινου και μπλε. Εάν οι τιμές του κάθε

χρώματος έχουν ένα εύρος από το 0 έως το 255, οι διαφορετικοί συνδυασμοί χρωμάτων που μπορούν να υπάρξουν είναι 16.777.216. Ο αριθμός των δυαδικών ψηφίων που απαιτούνται για κάθε εικονοστοιχείο είναι 24 bits. Πρακτικά μπορεί να θεωρηθεί ότι και έγχρωμη εικόνα αποτελείται από τρεις εικόνες γκρι ανάλυσης, όπου η κάθε μια αναπαριστά την ένταση της φωτεινότητας για το κάθε χρώμα.

3.2.4 Ενδεικτική Εικόνα

Οι ενδεικτικές εικόνες έχουν δύο συνιστώσες: έναν πίνακα δεδομένων ακέραιων X και έναν πίνακα χρωματικού χάρτη. Ο πίνακας χρωματικού χάρτη είναι ένας $m \times 3$ πίνακας που περιλαμβάνει τιμές στο εύρος $[0 \ 1]$. Το μήκος του χάρτη είναι ίσο με τον αριθμό των χρωμάτων που έχουν οριστεί. Κάθε γραμμή του χάρτη καθορίζει τα κόκκινα, τα πράσινα και τα μπλε στοιχεία ενός μόνο χρώματος. Ο τύπος αυτής της εικόνας χρησιμεύει για την αποθήκευση έγχρωμης RGB εικόνας η οποία θα καταλαμβάνει λιγότερο χώρο στην μνήμη.

Οι τελικές εικόνες οι οποίες θα επεξεργαζόμαστε στο MATLAB για την ανίχνευση του μετώπου της ροής της ρητίνης θα είναι σε γκρι ή δυαδική μορφή, καθώς οι περισσότερες μέθοδοι που έχουν αναπτυχθεί για ανίχνευση ακμών/αντικειμένων είναι για εικόνες αυτών των τύπων.

Στο MATLAB η μετάβαση από τον ένα είδος εικόνας στο άλλο γίνονται πολύ εύκολα χρησιμοποιώντας ειδικές εντολές. Οι εντολές αυτές θα περιγραφούν κατά την εξήγηση του κώδικα.

3.3 Συστήματα Συντεταγμένων

Οι θέσεις σε μια ψηφιακή εικόνα μπορούν να εκφραστούν σε διάφορα συστήματα συντεταγμένων. Τα δύο κύρια συστήματα συντεταγμένων που χρησιμοποιούνται είναι:

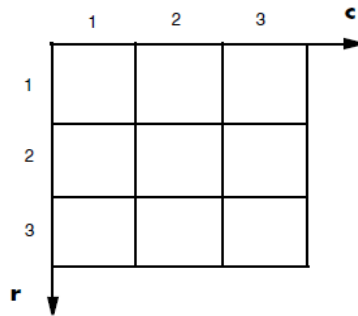
- Συντεταγμένες Εικονοστοιχείων (Pixel Coordinates)
- Χωρικές Συντεταγμένες (Spatial Coordinates)

3.3.1 Συντεταγμένες Εικονοστοιχείων

Η πιο κατάλληλη μέθοδος για την έκφραση θέσεων σε μια εικόνα είναι η χρήση συντεταγμένων των εικονοστοιχείων. Σε αυτό το σύστημα συντεταγμένων, η εικόνα αντιμετωπίζεται ως ένα πλέγμα διακεκριμένων στοιχείων, ταξινομημένων από πάνω προς τα κάτω και από αριστερά προς τα δεξιά. Δηλαδή εδώ το πρώτο στοιχείο r είναι ο αριθμός της σειράς η οποία αυξάνεται προς τα κάτω, ενώ το δεύτερο στοιχείο c είναι η στήλη η

οποία αυξάνεται προς τα δεξιά. Οι συντεταγμένες εικονοστοιχείων είναι ακέραιες τιμές οι οποίες κυμαίνονται μεταξύ του 1 και του μήκους της σειράς ή της στήλης (εικόνα).

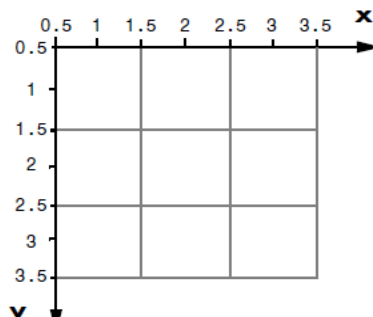
Στο MATLAB, υπάρχει αντιστοιχία ένα προς ένα μεταξύ των συντεταγμένων των εικονοστοιχείων και των συντεταγμένων του πίνακα που περιγράφει την συγκεκριμένη εικόνα. Για παράδειγμα, τα δεδομένα για το εικονοστοιχείο στην τέταρτη σειρά και τρίτη στήλη, αποθηκεύονται στο στοιχείο του πίνακα με δείκτες (4,3).



Εικόνα 4: Σύστημα Συντεταγμένων Εικονοστοιχείων

3.3.2 Χωρικές Συντεταγμένες

Στις συντεταγμένες εικονοστοιχείου, όπως έχουμε αναφέρει, κάθε εικονοστοιχείο αντιμετωπίζεται ως μια ξεχωριστή μονάδα, που αναγνωρίζεται από ένα μόνο ζεύγος συντεταγμένων ακέραιων αριθμών. Σε κάποιες εφαρμογές είναι χρήσιμο να θεωρήσουμε ένα εικονοστοιχείο ως μια τετράγωνη επιφάνεια που έχει εμβαδό. Στο χωρικό σύστημα συντεταγμένων, οι θέσεις μιας εικόνας είναι οι θέσεις ενός επιπέδου και περιγράφονται με σημεία x και y . Το σύστημα χωρικών συντεταγμένων είναι συνεχές σε αντίθεση με το σύστημα συντεταγμένων εικονοστοιχείων που είναι διακριτό.



Εικόνα 5: Σύστημα Χωρικών Συντεταγμένων

3.4 Βελτίωση Εικόνας

3.4.1 Εισαγωγή

Οι τεχνικές βελτίωσης εικόνας αναδεικνύουν τις λεπτομέρειες σε μια εικόνα που είναι σκοτεινή ή επισημαίνουν ορισμένα χαρακτηριστικά που μας ενδιαφέρουν σε μια εικόνα. Οι τεχνικές βελτίωσης περιλαμβάνουν την προσαρμογή της φωτεινότητας ή της αντίθεσης, το φιλτράρισμα, το μορφολογικό φιλτράρισμα και το θάμπωμα ή μη μιας εικόνας. Οι τεχνικές βελίωσης εικόνας τυπικά επιστρέφουν μια τροποποιημένη έκδοση της αρχικής εικόνας η οποία χρησιμεύει ως στάδιο προεπεξεργασίας για την βελτίωση των αποτελεσμάτων των τεχνικών ανάλυσης εικόνας.

3.4.2 Ιστόγραμμα Εικόνας

Το ιστόγραμμα μιας εικόνας αποχρώσεων του γκρι περιέχει σημαντικές πληροφορίες για μια εικόνα και για αυτόν το λόγο είναι ένα σημαντικό εργαλείο για την επεξεργασία εικόνας, αφού με την τροποποίηση του ιστογράμματος μπορούμε να αλλάξουμε την φωτεινότητα ή την αντίθεση μιας εικόνας.

Το ιστόγραμμα μιας εικόνας εκφράζει την κατανομή των αποχρώσεων του γκρι . Αποτελείται από ένα γράφημα που στον οριζόντιο άξονα έχει τις φωτεινότητες από 0-255 και στον κατακόρυφο άξονα το πλήθος των εικονοστοιχείων που έχουν κάθε φωτεινότητα.

Με τη βοήθεια του ιστογράμματος μπορούμε να ισορροπήσουμε την φωτεινότητα μιας εικόνας καθώς και να αυξήσουμε ή να μειώσουμε την αντίθεση. Η αντίθεση είναι πολύ σημαντική για την εύρεση ακμών σε μια εικόνα.

3.4.3 Τμηματοποίηση Εικόνας Με Τεχνικές Κατωφλίωσης

Τμηματοποίηση εικόνας (image segmentation), είναι η διαδικασία με την οποία διαχωρίζεται μια εικόνα σε κατάλληλες περιοχές που αυτή περιέχει. Αποτελεί αναγκαίο στάδιο προεπεξεργασίας για την αναγνώριση αντικειμένων σε μια εικόνα.

Η εύρεση κατωφλιών (Thresholds), είναι από τις ισχυρότερες τεχνικές για την τμηματοποίηση εικόνων της κλίμακας του γκρι, οι οποίες βασίζονται στο ιστόγραμμα της εικόνας. Στις τεχνικές αυτές ορίζεται μια συγκεκριμένη τιμή φωτεινότητας που ονομάζεται κατώφλι. Τα εικονοστοιχεία που έχουν φωτεινότητα μικρότερη ή ίση με το κατώφλι παίρνουν τιμή ίση με το 0, ενώ τα εικονοστοιχεία που έχουν μεγαλύτερη τιμή από το κατώφλι παίρνουν τιμή ίση με 255.

Σημαντικό στοιχείο για την πετυχημένη τμηματοποίηση της εικόνας είναι η σωστή επιλογή του κατωφλιού. Μια από τις καλύτερες τεχνικές εύρεσης κατωφλιού, είναι αυτή της μεθόδου Otsu. Το κριτήριο το οποίο χρησιμοποιείται στη μέθοδο του Otsu είναι η μεγιστοποίηση της διαχωριστικότητας μεταξύ των φωτεινών και σκοτεινών περιοχών. Η υλοποίησή του γίνεται ως εξής: Έστω n_i το πλήθος των εικονοστοιχείων με τιμή φωτεινότητας ίση με i . Κανονικοποιώντας το ιστόγραμμα, με βάση τις πιθανότητες εμφάνισης κάθε φωτεινότητας έχουμε:

$$p_i = \frac{n_i}{MN}, \quad \sum_{i=0}^{L-1} p_i = 1, \quad p_i \geq 0 \quad (3.1)$$

Τώρα χωρίζουμε τα εικονοστοιχεία σε δύο κλάσεις C_1 και C_2 , διαλέγοντας μια τιμή κατωφλιού k . Οι πιθανότητες κάποιο εικονοστοιχείο να ανήκει σε μια από τις δύο κλάσεις είναι:

$$P_1(k) = \sum_{i=0}^k p_i, \quad P_2(k) = \sum_{i=k+1}^{L-1} p_i = 1 - P_1(k) \quad (3.2)$$

Τα κέντρα των κλάσεων υπολογίζονται ως:

$$m_1(k) = \sum_{i=0}^k iP(i|C_1) = \sum_{i=0}^k i \frac{P(C_1|i)P(i)}{P(C_1)} = \frac{1}{P_1(k)} \sum_{i=0}^k ip_i \quad (3.3)$$

$$m_2(k) = \sum_{i=k+1}^{L-1} iP(i|C_2) = \sum_{i=k+1}^{L-1} i \frac{P(C_2|i)P(i)}{P(C_2)} = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} ip_i \quad (3.4)$$

Η μέση τιμή φωτεινότητας της εικόνας είναι :

$$m_G = \sum_{i=0}^{L-1} ip_i = P_1(k)m_1(k) + P_2(k)m_2(k) \quad (3.5)$$

Η συνολική μέση τιμή της έντασης της εικόνας (μέχρι το k) είναι:

$$m(k) = \sum_{i=0}^k ip_i \quad (3.6)$$

Η μεταβλητότητα μεταξύ των κλάσεων είναι :

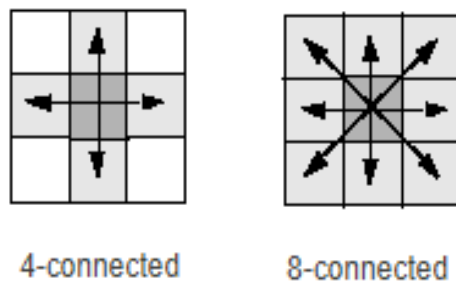
$$\sigma_B^2(k) = \frac{[m_G P_1(k) - m(k)]^2}{P_1(k)[1 - P_1(k)]} \quad (3.7)$$

Με επαναληπτική διαδικασία της παραπάνω σχέσης βρίσκουμε το k έτσι ώστε το $\sigma_B^2(k)$ να γίνεται μέγιστο.

3.4.4 Συνδεσιμότητα (Connectivity)

Ένα σημαντικό μέγεθος στις εικόνες είναι η έννοια της συνδεσιμότητας. Πολλές τεχνικές επεξεργασίας εικόνας χρησιμοποιούν την έννοια της γειτονιάς, για να ορίσουν μια τοπική περιοχή επιρροής. Σημαντικό για τον ορισμό της τοπικής γειτονιάς είναι η έννοια της συνδεσιμότητας των εικονοστοιχείων, δηλαδή η απόφαση για το ποια εικονοστοιχεία συνδέονται μεταξύ τους.

Υπάρχουν δύο είδη συνδεσιμότητας δύο διαστάσεων: τεσσάρων στοιχείων (4-connectivity) και οχτώ στοιχείων (8-connectivity). Στην συνδεσιμότητα τεσσάρων εικονοστοιχείων, αυτά συνδέονται μόνο όταν, τα άκρα τους αγγίζουν το ένα το άλλο. Αυτό σημαίνει ότι ένα ζευγάρι διπλανών εικονοστοιχείων είναι μέρος του ίδιου αντικειμένου μόνο αν και τα δύο είναι συνδεδεμένα κατά μήκος της οριζόντιας ή της κατακόρυφης κατεύθυνσης. Στην συνδεσιμότητα οχτώ εικονοστοιχείων, τα εικονοστοιχεία συνδέονται μεταξύ τους όταν έχουν κοινές άκρες ή όταν συνδέονται και διαγώνια. Η διαφορά των δύο διαφορετικών τρόπων συνδέσμων των εικονοστοιχείων φαίνεται στην παρακάτω εικόνα.



Εικόνα 6: Συνδεσιμότητα 4 και 8 στοιχείων

3.4.5 Φίλτρα Εικόνας

Ο κύριος στόχος της βελτίωσης εικόνας είναι να γίνει οπτικά αποδεκτή. Η αφαίρεση του θορύβου, η ενίσχυση των ακμών της εικόνας, το θάμπωμα της εικόνας είναι τεχνικές οι οποίες μπορούν να επιτευχθούν μέσω της διαδικασίας του φιλτραρίσματος στο χώρο (Spatial Domain Filtering), δηλαδή λαμβάνει χώρα απευθείας στα πραγματικά εικονοστοιχεία της ίδιας εικόνας. Τα φίλτρα ενεργούν σε μια εικόνα για να αλλάξουν τις τιμές των εικονοστοιχείων με κάποιο συγκεκριμένο τρόπο και ταξινομούνται σε : γραμμικά και μη γραμμικά φίλτρα. Τα γραμμικά φίλτρα είναι τα πιο συνηθισμένα.

Ανεξάρτητα από το συγκεκριμένο φίλτρο που χρησιμοποιείται, ο τρόπος που λειτουργεί είναι ακριβώς ο ίδιος. Κάθε ένα από τα εικονοστοιχεία μιας εικόνας, το οποίο λέγεται εικονοστοιχείο στόχου, επεξεργάζεται διαδοχικά. Η τιμή του εικονοστοιχείου στόχου, αντικαθίσταται από μια νέα τιμή η οποία εξαρτάται από την τιμή συγκεκριμένων εικονοστοιχείων σε μια συγκεκριμένη γειτονιά γύρω από το εικονοστοιχείο στόχου.

3.4.5.1 Γραμμικά χωρικά φίλτρα

Στα γραμμικά χωρικά φίλτρα η νέα τιμή του εικονοστοιχείου στόχου προσδιορίζεται ως μερικός γραμμικός συνδυασμός των τιμών των εικονοστοιχείων στη γειτονιά του. Οποιοσδήποτε άλλος τύπος φίλτρου είναι από τον ορισμό μη γραμμικό φίλτρο. Ο συγκεκριμένος γραμμικός συνδυασμός των γειτονικών εικονοστοιχείων που λαμβάνεται καθορίζεται από τον πυρήνα του φίλτρου ή αλλιώς μάσκα. Ο πυρήνας είναι ένας πίνακας ίδιου μεγέθους με την γειτονιά ο οποίος περιλαμβάνει τους συντελεστές βαρύτητας οι οποίοι πρόκειται να εκχωρηθούν σε κάθε ένα από τα αντίστοιχα εικονοστοιχεία στη γειτονιά του εικονοστοιχείου στόχου. Το φιλτράρισμα προχωρά με τη διαδοχική τοποθέτηση του πυρήνα έτσι ώστε η θέση του κέντρου του εικονοστοιχείου να συμπίπτει με τη θέση κάθε εικονοστοιχείου στόχου. Κάθε φορά η φιλτραρισμένη τιμή υπολογίζεται από τον συνδυασμό του επιλεγμένου συντελεστή βαρύτητας και των εικονοστοιχείων στην γειτονιά. Αυτή η διαδικασία φιλτραρίσματος μπορεί να παρουσιαστεί ως ολίσθηση του πυρήνα σε όλες τις τοποθεσίες ενδιαφέροντος της αρχικής εικόνας, πολλαπλασιάζοντας τα εικονοστοιχεία με τους αντίστοιχους συντελεστές βαρύτητας του πυρήνα, και προσθέτοντας μεταξύ τους τις τιμές αυτές. Επομένως η τιμή που έχει υπολογιστεί είναι η νέα τιμή του εικονοστοιχείου στόχου.

Οι μηχανισμοί του γραμμικού χωρικού φιλτραρίσματος στην πραγματικότητα εκφράζονται ως μια μαθηματική διαδικασία διακριτής μορφής που ονομάζεται συνέλιξη. Για τον λόγο αυτό, πολλοί πυρήνες φίλτρων περιγράφονται μερικές φορές ως πυρήνες συνέλιξης. Τη διαδικασία της συνέλιξης μεταξύ του πυρήνα και της εικόνας μπορούμε να την εκφράσουμε με την παρακάτω σχέση, όπου οι δείκτες $i=0, j=0$ αντιστοιχούν στο κεντρικό εικονοστοιχείο του πυρήνα, το οποίο έχει μέγεθος $(I_{\max}-I_{\min}+1, J_{\max}-J_{\min}+1)$

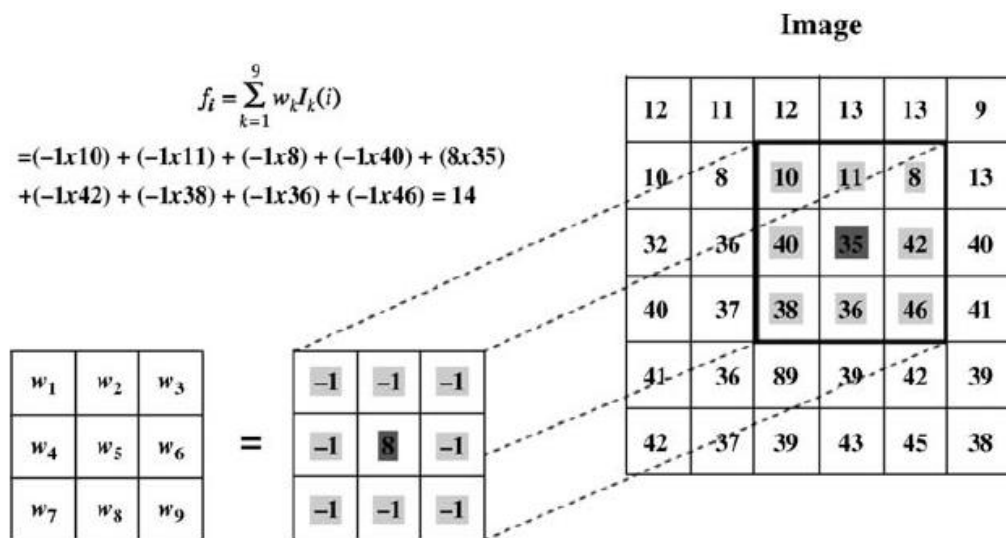
$$f(x, y) = \sum_{i=I_{\min}}^{I_{\max}} \sum_{j=J_{\min}}^{J_{\max}} w(i, j)I(x + i, y + j) \quad (3.8)$$

Σε ορισμένες περιπτώσεις μπορούμε να εφαρμόσουμε ένα γραμμικό φίλτρο σε μια επιλεγμένη περιοχή και όχι σε ολόκληρη την εικόνα, επομένως κάνουμε φιλτράρισμα βάσει περιοχής. Το φιλτράρισμα στα όρια των εικόνων δημιουργεί μια ενδιαφέρουσα κατάσταση,

αφού όταν το εικονοστοιχείο στόχου βρίσκεται κοντά στα όρια της εικόνας, ο πυρήνας επικαλύπτει την άκρη της εικόνας με αποτέλεσμα μερικά εικονοστοιχεία του πυρήνα να μην αντιστοιχούν σε κανένα εικονοστοιχείο της εικόνας. Σε αυτήν την περίπτωση έχουμε τρεις επιλογές:

- Αφήνουμε αμετάβλητα εκείνα τα εικονοστοιχεία στόχου που βρίσκονται μέσα στο όριο της εικόνας.
- Εκτελείται φιλτράρισμα μόνο σε εκείνα τα εικονοστοιχεία που βρίσκονται μέσα στο όριο.
- Συμπληρώνουμε τα εικονοστοιχεία που λείπουν από τη λειτουργία του φίλτρου αντικατοπτρίζοντας τιμές πάνω από το όριο.

Η δεύτερη και η τρίτη μέθοδος προτιμώνται περισσότερο.



Εικόνα 7: Ο μηχανισμός φιλτραρίσματος εικόνας με 3x3 πυρήνα φίλτρου

3.4.5.2 Μη Γραμμικά Χωρικά Φίλτρα

Η λειτουργία των μη γραμμικών φίλτρων έχει τον ίδιο βασικό μηχανισμό που περιγράφηκε παραπάνω για τα γραμμικά φίλτρα. Ο πυρήνας ολισθαίνει πάνω στην εικόνα με τον ίδιο τρόπο όπως στη γραμμική περίπτωση, με τη διαφορά ότι η φιλτραρισμένη τιμή θα είναι αποτέλεσμα κάποιας μη γραμμικής συνάρτησης. Για παράδειγμα, όπως πριν, θα μπορούσαμε να ορίσουμε ένα τετραγωνικό φίλτρο. Στην περίπτωση αυτή, η δράση του φίλτρου θα καθοριστεί από τρεις συντελεστές βαρύτητας που καθορίζουν την συμβολή των όρων δεύτερης, πρώτης και μηδενικής τάξης.

$$f_i = \sum_{k=1}^N w_{k1} I_k^2(i) + w_{k2} I_k(i) + w_{k3} \quad (3.9)$$

Παρακάτω θα περιγραφούν τα πιο συνηθισμένα φίλτρα.

3.4.5.3 Φίλτρο Μέσης Τιμής (Mean filtering)

Ένα από τα πιο απλά γραμμικά φίλτρα είναι το φίλτρο μέσης τιμής. Χρησιμοποιείται για την εξομάλυνση των εικόνων και τη μείωση του θορύβου σε αυτές. Η λειτουργία του φίλτρου μέσης τιμής συνίσταται στην αντικατάσταση της φωτεινότητας σε κάθε εικονοστοιχείο, με τη μέση φωτεινότητα σε μια γειτονιά του. Αυτό έχει ως αποτέλεσμα τη μείωση της μεταβλητότητας τοπικά σε κάθε εικονοστοιχείο και επομένως το θάμπωμα της εικόνας. Το φίλτρο μέσης τιμής μπορεί να θεωρηθεί ως ένα κατωδιαβατό φίλτρο το οποίο έχει ως αποτέλεσμα την απαλοιφή των στοιχείων της εικόνας που αντιστοιχούν σε υψηλές συχνότητες, δηλαδή σημεία υψηλού ενδιαφέροντος όπως τις ακμές. Η μάσκα του φίλτρου μέσης τιμής αποτελείται από ένα τετραγωνικό πίνακα συνήθως 3x3, όπου οι συντελεστές βαρύτητας έχουν τιμή ίση με 1, η οποία διαιρείται με έναν συντελεστή κλιμάκωσης ο οποίος ισούται με το άθροισμα των στοιχείων της μάσκας. Ένα τέτοιο φίλτρο φαίνεται παρακάτω:

$$h(x, y) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Όσο μεγαλύτερη είναι η γειτονιά, τόσο μεγαλύτερη εξομάλυνση επιτυγχάνεται και τόσο περισσότερες λεπτομέρειες της εικόνας εξαφανίζονται.

Μια παραλλαγή του φίλτρου μέσης τιμής είναι τα φίλτρα μέσης τιμής υπό συνθήκες. Με αυτή την μεθοδολογία, η τιμή του εικονοστοιχείου στόχου, αντικαθίσταται με τη μέση τιμή των γειτονικών του εικονοστοιχείων που ικανοποιούν κάποια συνθήκη. Τα φίλτρα αυτά επιτρέπουν την μείωση του αθροιστικού θορύβου σε εικόνες που έχουν ανά τμήματα σταθερές τιμές, χωρίς να επηρεάζονται σημαντικά οι ακμές.

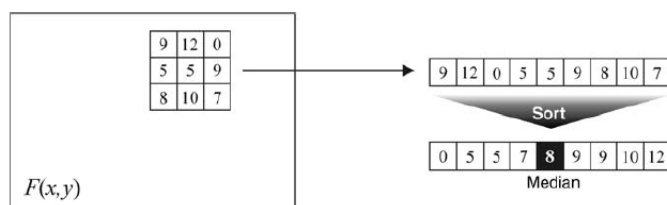
3.4.5.4 Φίλτρο Μεσαίας Τιμής (Median filtering)

Ένα άλλο φίλτρο που χρησιμοποιείται πολύ, είναι το φίλτρο μεσαίας τιμής το οποίο είναι μη γραμμικό. Το φίλτρο μεσαίας τιμής αίρει του περιορισμούς που έχουν τα φίλτρα μέσης τιμής αλλά με μεγαλύτερο υπολογιστικό κόστος, καθώς με το φίλτρο μεσαίας τιμής μπορεί να διατηρηθεί η έντονη λεπτομέρεια υψηλής συχνότητας όπως τα άκρα, ενώ ταυτόχρονα

αποφεύγεται ο θόρυβος, ιδιαίτερα ο απομονωμένος. Η τιμή median μιας γειτονιάς, δηλαδή ενός συνόλου, είναι ίση με τη μεσαία τιμή του συνόλου. Επομένως αφού ταξινομηθούν τα στοιχεία του συνόλου κατά αύξουσα τιμή, αν το πλήθος των στοιχείων είναι περιττός, η μεσαία τιμή ισούται με το μεσαίο στοιχείο του συνόλου, ενώ στην περίπτωση άρτιου πλήθους στοιχείων η μεσαία τιμή ισούται με την μέση τιμή των δύο γειτονικών μεσαίων στοιχείων. Λόγω του ότι η μεσαία τιμή προέρχεται από την ίδια γειτονιά των εικονοστοιχείων, είναι πιο ισχυρή και δεν δημιουργεί νέα μη ρεαλιστική τιμή εικονοστοιχείων. Αυτό βοηθά στην αποφυγή θολώματος των ακμών και απώλειας λεπτομέρειας της εικόνας.

Το κυριότερο μειονέκτημα της εφαρμογής του φίλτρου μεσαίας τιμής, είναι η αλλοίωση που επιφέρει σε λεπτές γραμμές και γωνίες. Αυτό το πρόβλημα μπορεί να λυθεί χρησιμοποιώντας γειτονιά κατάλληλης μορφής ανάλογα με την κατεύθυνση των γραμμών που δεν θέλουμε να εξαφανιστούν.

Το φίλτρο μεσαίας τιμής, στην πραγματικότητα είναι μια ειδική περίπτωση ενός φίλτρου γενικής κατάταξης. Τα φίλτρα γενικής κατάταξης περιλαμβάνουν τα ίδια στάδια με αυτά του φίλτρου μεσαίας τιμής που περιγράφηκε. Επομένως αφού πρώτα καθοριστεί η γειτονιά του εικονοστοιχείου στόχου, ύστερα ταξινομείται κατά αύξουσα σειρά και επιλέγουμε ποια τιμή θα είναι το νέο εικονοστοιχείο. Για παράδειγμα, αν επιλέξουμε το στοιχείο με την μικρότερη τιμή έχουμε φίλτρο ελάχιστης τιμής, αντίθετα αν επιλέξουμε το στοιχείο με την μεγαλύτερη τιμή, έχουμε φίλτρο μέγιστης τιμής.



Εικόνα 8: Φίλτρο μεσαίας τιμής

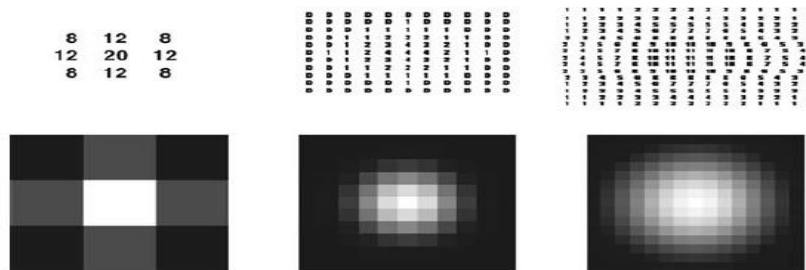
3.4.5.5 Φίλτρο Gauss

Το φίλτρο Gauss είναι από τα σημαντικότερα φίλτρα τόσο για θεωρητικούς όσο και για πρακτικούς λόγους. Χρησιμοποιώντας το φίλτρο Gauss σε μια εικόνα, την φιλτράρουμε χρησιμοποιώντας έναν πυρήνα διακριτών τιμών που προέρχεται από μια ακτινικά συμμετρική μορφή της συνεχούς συνάρτησης Gauss δύο διαστάσεων, όπως φαίνεται παρακάτω:

$$f(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (3.10)$$

Διακριτές προσεγγίσεις σε αυτή τη συνεχή συνάρτηση καθορίζονται χρησιμοποιώντας δύο ελεύθερες παραμέτρους: το επιθυμητό μέγεθος του πυρήνα (NxN) και την τιμή σ, τη τυπική απόκλιση μιας κατανομής Gauss. Το φίλτρο Gauss θεωρείται κατωδιαβατό φίλτρο όπως το φίλτρο μέσης τιμής το οποίο έχει ως αποτέλεσμα την εξομάλυνση της εικόνας. Διαφέρει από αυτόν της μέσης τιμής, καθώς ο βαθμός εξομάλυνσης ελέγχεται από την επιλογή της τιμής σ και όχι από την απόλυτη τιμή του μεγέθους του πυρήνα. Επιπλέον ο μετασχηματισμός Fourier της συνάρτησης Gauss είναι πάλι μια συνάρτηση Gauss, μια ιδιότητα που είναι σημαντική για τα φίλτρα στο πεδίο της συχνότητας.

Το φίλτρο Gauss όπως και άλλα φίλτρα είναι το πρώτο στάδιο των αλγορίθμων για την εύρεση ακμών, θέμα το οποίο θα αναλύσουμε παρακάτω, αφού είναι σημαντικό στην παρούσα εργασία, για την εύρεση του μετώπου της ροής.



Εικόνα 9: Φίλτρο Gauss με πυρήνα και τυπική απόκλιση, 3x3 σ=1, 11x11 σ=2 και 21x21 σ=4 αντίστοιχα.

3.5 Ανίχνευση Ακμών

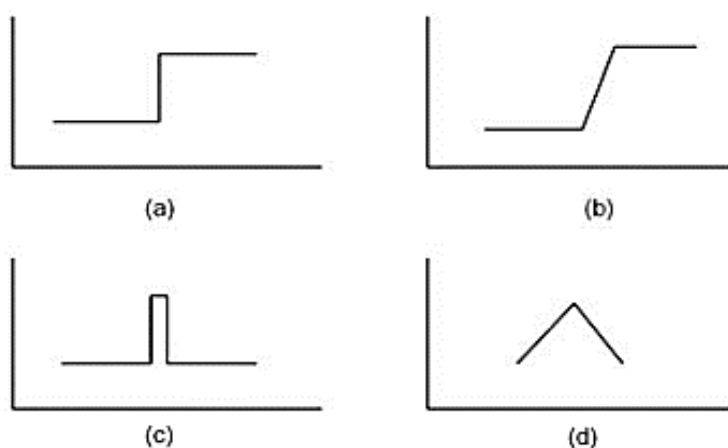
3.5.1 Εισαγωγή

Η ανίχνευση ακμών είναι μια από τις σημαντικότερες και ευρέως μελετημένες πτυχές της επεξεργασίας εικόνας. Αν μπορούμε να βρούμε το όριο ενός αντικειμένου εντοπίζοντας όλες τις ακμές του, τότε έχουμε τμηματοποιήσει την εικόνα αποτελεσματικά. Ως ακμές μπορούν να θεωρηθούν περιοχές μετάβασης έντασης μεταξύ δύο αντικειμένων. Οι ακμές χαρακτηρίζονται από συγκεκριμένα γεωμετρικά χαρακτηριστικά, όπως το πάχος, το πλάτος, η κλίση. Διαφορετικές προσεγγίσεις καταλήγουν σε διαφορετικές ανιχνεύσεις ακμών. Ανάλογα με την εφαρμογή απαιτείται αντίστοιχος αλγόριθμος. Οι περισσότερες μέθοδοι ανίχνευσης ακμών, παρουσιάζουν περιορισμένη απόδοση σε εικόνες οι οποίες μπορεί να έχουν θόρυβο ή να έχουν δημιουργηθεί σκιές λόγω μη καλού φωτισμού, επομένως για την

επιτυχία της μεθόδου ανίχνευσης ακμών είναι αναγκαία η προεπεξεργασία της εικόνας, όπως έχουμε αναφέρει πιο πάνω.

Υπάρχουν τέσσερις τύποι ακμών που προκύπτουν με βάση την μεταβολή της φωτεινότητας της εικόνας:

- Η βηματική: όπου ένταση της εικόνας αλλάζει απότομα από μια τιμή σε μια άλλη τιμή.
- Η κλιμακωτή: όταν η αλλαγή της έντασης δεν είναι απότομη, αλλά προηγείται μια μικρή απόσταση έως ότου αλλάξει στην τελική τιμή.
- Η γραμμική: όπου η ένταση της εικόνας αλλάζει ξαφνικά αλλά στη συνέχεια επιστρέφει στην αρχική τιμή μετά από ένα διάστημα σταθερής τιμής.
- Τέλος η τύπου οροφής: όπου μοιάζει με την γραμμική, αλλά η διαφορά είναι ότι επιστρέφει αμέσως στη αρχική τιμή.



Εικόνα 10: Είδη ακμών, (a) Βηματική, (b) κλιμακωτή, (c) γραμμική, (d) οροφής

Οι μέθοδοι ανίχνευσης ακμών συνήθως βασίζονται σε υπολογισμούς της πρώτης ή της δεύτερης παραγώγου κατά μήκος του προφίλ έντασης. Η πρώτη παράγωγος έχει την επιθυμητή ιδιότητα να είναι άμεσα ανάλογη με τη διαφορά έντασης κατά μήκος της ακμής. Επομένως το μέγεθος της πρώτης παραγώγου μπορεί να χρησιμοποιηθεί για την ανίχνευση ακμής σε συγκεκριμένο σημείο στην εικόνα. Η δεύτερη παράγωγος μπορεί να χρησιμοποιηθεί για τον προσδιορισμό αν ένα εικονοστοιχείο βρίσκεται στην φωτεινή ή στην σκοτεινή πλευρά μιας ακμής.

3.5.2 Ανίχνευση Ακμών Πρώτης Παραγώγου

Οι πιο απλές μέθοδοι ανίχνευσης ακμών λειτουργούν εκτιμώντας την κλίση του γκρι επιπέδου σε ένα εικονοστοιχείο, το οποίο μπορεί να υπολογιστεί με το ψηφιακό ισοδύναμο

της πρώτης παραγώγου, όπως φαίνεται παρακάτω:

$$g_x(x, y) \approx f(x + 1, y) - f(x - 1, y)$$

$$g_y(x, y) \approx f(x, y + 1) - f(x, y - 1)$$

Η 2×2 προσεγγίσεις της παραγώγου πρώτης τάξης είναι γνωστή ως μέθοδος του **Roberts** και μπορεί να περιγραφεί ως ένας πίνακας 2×2:

$$g_x = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

$$g_y = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$$

Αυτές οι κλίσεις μπορούν να υπολογιστούν μέσα σε μια γειτονιά 3×3 με χρήση της συνέλιξης, όπως φαίνεται παρακάτω:

$$g_x(x, y) = h_x * f(x, y)$$

$$g_y(x, y) = h_y * f(x, y)$$

Όπου h_x και h_y είναι οι μάσκες συνέλιξης.

Το ζεύγος πυρήνων, της μεθόδου ανίχνευση ακμών **Prewitt**, φαίνεται παρακάτω:

$$h_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix}$$

$$h_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

Ένα παρόμοιο ζεύγος το οποίο δίνει μεγαλύτερη έμφαση στα εικονοστοιχεία πάνω στους άξονες, είναι η μέθοδος ανίχνευσης ακμών Sobel, η οποία φαίνεται παρακάτω:

$$h_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$

$$h_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Παρά τις διαφορές των 3×3 μασκών των μεθόδων που παρουσιάστηκαν παραπάνω, μοιράζονται δύο ίδιες ιδιότητες. Αυτές είναι :

- Έχουν συντελεστές αντίθετων αριθμών, προκειμένου να επιτευχθεί υψηλή

απόκριση σε περιοχές εικόνας με ποικίλες εντάσεις.

- Το άθροισμα των συντελεστών είναι ίσο με το μηδέν, το οποίο σημαίνει ότι όταν εφαρμόζεται σε απολύτως ομοιογενείς περιοχές της εικόνας, το αποτέλεσμα θα είναι 0 δηλαδή μαύρο εικονοστοιχείο.

Η μέθοδος Roberts είναι γρήγορη στο υπολογισμό λόγω του μικρού μεγέθους των μασκών, αλλά είναι πολύ ευαίσθητες στον θόρυβο. Οι μέθοδοι ανίχνευσης ακμών Prewitt και Sobel προτιμώνται γενικά περισσότερο από την προσέγγιση του Roberts, διότι η κλίση δεν μετατοπίζεται κατά μισό εικονοστοιχείο και στις δύο κατευθύνσεις, και η εφαρμογή σε γειτονιά μεγαλύτερη από 3x3 δεν είναι εφικτή με τη μέθοδο Roberts.

Τα φίλτρα των Sobel και Prewitt, ανήκουν στα φίλτρα που μπορούν να διαχωριστούν γραμμικά. Αυτό σημαίνει ότι ο πυρήνας του φίλτρου μπορεί να εκφραστεί ως δύο διαφορετικά διανύσματα στήλης και γραμμής. Επομένως κατά την διεργασία φιλτραρίσματος δύο διαστάσεων μπορεί να υλοποιηθεί με διαδοχικά μονοδιάστατα φιλτραρίσματα, το πρώτο με φίλτρο γραμμής και στη συνέχεια με φίλτρο στήλης. Αυτό έχει ως αποτέλεσμα μικρότερο υπολογιστικό κόστος. Η ιδιότητα αυτή φαίνεται παρακάτω για τις μάσκες Prewitt και Sobel αντίστοιχα:

$$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} [1 \ 0 \ -1] \quad \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} [1 \ 0 \ -1]$$

3.5.3 Ανίχνευση ακμών Δεύτερης Παραγώγου

3.5.3.1 Μέθοδος με χρήση του τελεστή Laplace

Η πιο γνωστή μέθοδος εύρεσης ακμών με χρήση παραγώγων δευτέρου βαθμού, είναι η μέθοδος με χρήση του τελεστή Laplace. Στο συνεχές πεδίο, η Λαπλασιανή μιας συνάρτησης $f(x,y)$ καθορίζεται από τη σχέση :

$$\nabla^2 f(x, y) = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{3.11}$$

Σε διακριτή μορφή αντίστοιχα :

$$\nabla^2 f = f(x+1, y) + f(x-1, y) - 4f(x, y) + f(x, y+1) + f(x, y-1) \tag{3.12}$$

Η Λαπλασιανή ισούται με το μηδέν αν η $f(x,y)$ είναι μια σταθερή ή γραμμικής συνάρτηση.

Σε διαφορετική περίπτωση η Λαπλασιανή έχει διασκελισμό του μηδενός στη θέση όπου έχουμε τοπικό ακρότατο. Τα σημεία όπου έχουμε διασκελισμό του μηδενός τα θεωρούμε ως σημεία ακμών. Η τεχνική αυτή δίνει ένα μεγάλο αριθμό σημείων ακμών διότι ακόμα και μια ελάχιστη κυμάτωση της πρώτης παραγώγου είναι αρκετή για να δώσει ένα σημείο ακμής. Επομένως λόγω αυτής της ευαισθησίας είναι απαραίτητη η προεπεξεργασία της εικόνας έτσι ώστε να εξαλειφθεί ο θόρυβος που μπορεί να υπάρχει. Για τη διακριτή περίπτωση μιας εικόνας, οι μερικές παράγωγοι δεύτερης τάξης μπορούν να προσεγγισθούν από δευτέρου βαθμού διαφορές. Οι διαφορές αυτές μπορούν να υπολογισθούν με τη συνέλιξη της εικόνας με την κρουστική απόκριση ενός δισδιάστατου φίλτρου. Οι διάφορες μάσκες συνέλιξης που προκύπτουν λαμβάνοντας με διαφορετικό τρόπο τις διαφορές δευτέρου βαθμού είναι:

$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}$$

Λόγω της ευαισθησίας στον θόρυβο που έχει η μέθοδος, έχει αναπτυχθεί μια τεχνική η οποία εξαλείφει το πρόβλημα αυτό. Η τεχνική αυτή χρησιμοποιεί ένα δισδιάστατο φίλτρο Gauss σαν πρώτο στάδιο στην αρχική εικόνα, και ύστερα αφού προκύψει η Λαπλασιανή βρίσκονται τα εικονοστοιχεία για τα οποία υπάρχει διασκελισμός του μηδενός. Το φίλτρο αυτό είναι γνωστό ως φίλτρο LoG. Τα βασικά χαρακτηριστικά της τεχνικής αυτής είναι: ότι η επίδραση των απομακρυσμένων εικονοστοιχείων μειώνεται ανάλογα με τη Gaussian συνάρτηση, και η μεταβολή της τιμής της τυπικής απόκρισης δεν επιδρά στην θέση του διασκελισμού του μηδενός, αλλά καθώς η τυπική απόκλιση αυξάνει εξαλείφονται λιγότερο σημαντικές ακμές.

3.5.3.2 Αλγόριθμος Ανίχνευσης Ακμών Canny

Ο αλγόριθμος του Canny θεωρείται ένας από τους πιο ισχυρούς για την ανίχνευση και δημιουργία ακμών. Στοχεύει στην ανάπτυξη ενός ανιχνευτή ακμών που ικανοποιεί τρία βασικά κριτήρια:

- Χαμηλό ποσοστό σφάλματος. Δηλαδή ακμές που υπάρχουν πράγματι, να ανιχνεύονται με μεγάλη πιθανότητα, καθώς και σημεία που δεν ανήκουν σε ακμές να μην αναγνωρίζονται ως ακμές.
- Τα σημεία ακμής που ανιχνεύονται πρέπει να είναι καλά εντοπισμένα. Με αυτό εννοούμε, η απόσταση μεταξύ των εικονοστοιχείων της ακμής που εντοπίστηκε και η πραγματική να είναι όσο γίνεται ελάχιστη.
- Μοναδική απόκριση σε κάθε ακμή. Δηλαδή η πραγματική ακμή που υπάρχει να μη

δώσει ως απόκριση περισσότερες από μια.

Τα βήματα που ακολουθούνται κατά την εκτέλεση της μεθόδου Canny είναι:

1. Η εικόνα αρχικά εξομαλύνεται χρησιμοποιώντας ένα φίλτρο με Gaussian πυρήνα μηδενικής μέσης τιμής και καθορισμένης τυπικής απόκλισης σ . Οι τελεστές κλίσης είναι ευαίσθητοι στον θόρυβο για αυτό τον λόγο χρησιμοποιούμε το φίλτρο ώστε να μειωθεί ο θόρυβος της εικόνας. Όσο μεγαλύτερος είναι ο πυρήνας τόσο μεγαλύτερη εξομάλυνση έχουμε, αλλά οδηγούν και σε μεγαλύτερο σφάλμα στην ακριβή τοποθεσία των ακμών.
2. Βρίσκεται η ένταση της κλίσης της εικόνας. Αυτό επιτυγχάνεται χρησιμοποιώντας της μάσκες του Sobel στην οριζόντια και στη κάθετη κατεύθυνση.

$$E(x, y) = |G_x(x, y)| + |G_y(x, y)|.$$

3. Υπολογίζεται η κατεύθυνση της ακμής:

$$\theta = \tan^{-1} \frac{G_y(x, y)}{G_x(x, y)}$$

4. Ψηφιοποιείται η κατεύθυνση της ακμής. Η κατεύθυνση της ακμής είναι γνωστή, σε αυτό το βήμα, και θα την προσεγγίσουμε σε μια κατεύθυνση που μπορεί να ανιχνευτεί σε μια ψηφιακή εικόνα. Παίρνοντας ένα τυχαίο εικονοστοιχείο, η κατεύθυνση μιας ακμής που διέρχεται από αυτό το εικονοστοιχείο μπορεί να πάρει μια από τις τέσσερις πιθανές τιμές, δηλαδή 0° , 90° , 45° , και 135° . Συνεπώς προσεγγίζουμε την υπολογισμένη γωνία θ , με μια από αυτές τις τιμές που είναι κοντινότερη σε αυτήν.
5. Καταστολή των μη μέγιστων τιμών. Κατά μήκος της ακμής στην κατεύθυνσή της, καταστέλλεται οποιοδήποτε εικονοστοιχείο το οποίο θεωρείται ότι δεν ανήκει στην ακμή. Με τον τρόπο αυτό επιτυγχάνεται το φιλτράρισμα και η λέπτυνση των ακμών.
6. Κατωφλίωση υστέρησης. Η τεχνική αυτή στοχεύει στην περαιτέρω μείωση των εικονοστοιχείων των ακμών της εικόνας που έχει προκύψει ύστερα από την ολοκλήρωση των πέντε προηγούμενων βημάτων. Για τον σκοπό αυτό χρησιμοποιούνται δύο τιμές κατωφλιού T_1 και T_2 , όπου $T_1 < T_2$. Η τύχη του κάθε εικονοστοιχείου εξαρτάται από τις παρακάτω συνθήκες.

- $|E(x, y)| < T_1$, το εικονοστοιχείο απορρίπτεται.
- $|E(x, y)| > T_2$, το εικονοστοιχείο γίνεται αποδεκτό.
- $T_1 < |E(x, y)| < T_2$, το εικονοστοιχείο απορρίπτεται, εκτός αν υπάρχει μια διαδρομή από αυτό το εικονοστοιχείο σε ένα εικονοστοιχείο με μια κλίση μεγαλύτερη από T_2 . Η διαδρομή αυτή περιλαμβάνει και τις διαγώνιες

κινήσεις όπως καθορίζεται από την 8-συνδετικότητα.

Μετά το τέλος της κατωφλίωσης προκύπτει η τελική εικόνα.

Να σημειωθεί πως η τελική μορφή της εικόνας είναι σε δυαδική μορφή ανεξαρτήτως της τεχνικής που έχουμε χρησιμοποιήσει.

3.6 Μορφολογικές λειτουργίες

3.6.1 Εισαγωγή

Η έννοια της μορφολογίας είναι η μελέτη της μορφής ή της δομής. Στην επεξεργασία εικόνας χρησιμοποιούμε την μαθηματική μορφολογία ως μέσο για τον εντοπισμό και την εξαγωγή σημαντικών στοιχείων, που είναι χρήσιμα για την παρουσίαση και την περιγραφή ενός σχήματος μέσα σε μια εικόνα. Οι μορφολογικές λειτουργίες μπορούν να εφαρμοστούν, σε όλους τους τύπους εικόνας, αλλά η κύρια χρήση τους είναι στις δυαδικές εικόνες.

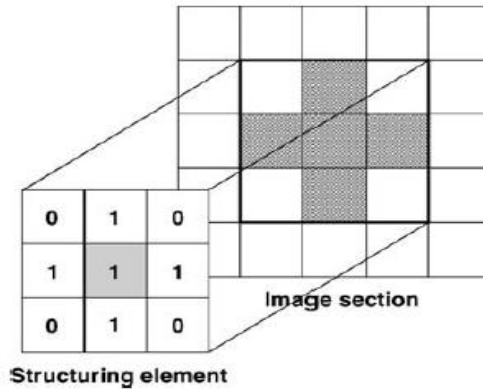
3.6.2 Δυαδικές Εικόνες - Προσκήνιο/Παρασκήνιο

Σε μια δυαδική εικόνα κάθε εικονοστοιχείο μπορεί να πάρει μια από τις δύο πιθανές λογικές τιμές 1 και 0. Η λογική τιμή 1 θεωρείται ως αληθής και η λογική τιμή 0 θεωρείται ως ψευδής. Στην επεξεργασία εικόνας, τα εικονοστοιχεία μιας δυαδικής εικόνας που έχουν τιμή ίση με 1 θεωρούμε ότι βρίσκονται στο προσκήνιο, ενώ τα εικονοστοιχεία που έχουν τιμή ίση με 0, θεωρούμε ότι βρίσκονται στο παρασκήνιο. Ένα αντικείμενο σε μια δυαδική εικόνα αποτελείται από μια ομάδα συνδεδεμένων μεταξύ τους εικονοστοιχείων.

3.6.3 Στοιχείο Δόμησης (Structuring Element)

Στην μορφολογική επεξεργασία μιας εικόνας, ο πιο κρίσιμος παράγοντας εκτός από το είδος της μορφολογικής λειτουργίας, είναι το στοιχείο δόμησης (Structuring Element). Ένα στοιχείο δόμησης είναι ένας ορθογώνιος πίνακας ο οποίος αποτελείται από εικονοστοιχεία με τιμές 1 ή 0.

Το κεντρικό εικονοστοιχείο ενός στοιχείου δόμησης αντιπροσωπεύει το εικονοστοιχείο ενδιαφέροντος, το οποίο βρίσκεται ακριβώς στο κέντρο του στοιχείου δόμησης, όταν και οι δύο διαστάσεις του είναι περιττός αριθμός, ενώ αν μια ή και οι δύο από τις διαστάσεις τους είναι ζυγός αριθμός, τότε το κεντρικό εικονοστοιχείο βρίσκεται πιο κοντά στην πάνω μεριά ή στην αριστερή μεριά για διάσταση κατά γραμμές ή στήλες αντίστοιχα. Στο MATLAB το κεντρικό εικονοστοιχείο βρίσκεται με την εντολή $\text{floor}((\text{size}(\text{nhood})+1)/2)$, όπου nhood είναι το στοιχείο δόμησης.



Εικόνα 11: Παράδειγμα στοιχείου διαμόρφωσης (Structuring element)

3.6.4 Διαστολή-Διάβρωση (Dilation-Erosion)

Οι κύριες μορφολογικές λειτουργίες είναι η διαστολή και η διάβρωση. Η διαστολή και η διάβρωση είναι σχετικές μεταξύ τους λειτουργίες αλλά δημιουργούν τελείως διαφορετικά αποτελέσματα. Η διαστολή προσθέτει εικονοστοιχεία στα όρια του αντικειμένου, δηλαδή “παχαίνει” το αντικείμενο, ενώ η διάβρωση αφαιρεί εικονοστοιχεία στα όρια του αντικειμένου, δηλαδή “λεπταίνει” το αντικείμενο.

Η κατεύθυνση και η έκταση των δύο αυτών μορφολογικών λειτουργιών, εξαρτάται από το σχήμα και το μέγεθος του στοιχείου δόμησης. Για αυτό το λόγο είναι πολύ σημαντικό η σωστή επιλογή του στοιχείου δόμησης έτσι ώστε να έχουμε τα επιθυμητά αποτελέσματα.

Μαθηματικά η λειτουργία της διαστολής περιγράφεται ως:

$$A \oplus B = \{z | (\hat{B})_z \cap A \neq \emptyset\}$$

Και η λειτουργία της διάβρωσης ως:

$$A \ominus B = \{z | (\hat{B})_z \cup A^c \neq \emptyset\}$$

όπου A είναι η εικόνα και B το στοιχείο δόμησης.

3.6.5 Αποσύνθεση Στοιχείου Δόμησης

Ένα στοιχείο διαμόρφωσης μπορεί να αναπαρασταθεί ως διαστολή δύο στοιχείων δόμησης $B = B_1 \oplus B_2$. Έτσι η εικόνα αντί να διασταλεί με το B, διαστέλεται, πρώτα με το στοιχείο διαμόρφωσης B_1 και μετά με το B_2 . Με αυτόν τον τρόπο το υπολογιστικό κόστος είναι αρκετά μικρότερο, ειδικά σε μεγάλα στοιχεία δόμησης. Έτσι αν έχουμε ένα 5x5 στοιχείο δόμησης, ο αριθμός των στοιχείων που το αποτελεί είναι 25, με την αποσύνθεση έχουμε

δύο στοιχεία διαμόρφωσης από 5 στοιχεία το καθένα. Επομένως έχουμε λιγότερες πράξεις αφού το μέγεθος των δύο καινούριων στοιχείων δόμησης είναι μικρότερο όπως φαίνεται παρακάτω. Στη MATLAB αυτή η λειτουργία γίνεται αυτόματα.

$$\begin{array}{ccccc}
 1 & 1 & 1 & 1 & 1 \\
 1 & 1 & 1 & 1 & 1 \\
 1 & 1 & \boxed{1} & 1 & 1 \\
 1 & 1 & 1 & 1 & 1 \\
 1 & 1 & 1 & 1 & 1
 \end{array}
 \oplus
 \left[\begin{array}{ccccc}
 1 & 1 & \boxed{1} & 1 & 1
 \end{array} \right]
 \oplus
 \left[\begin{array}{c}
 1 \\
 1 \\
 \boxed{1} \\
 1 \\
 1
 \end{array} \right]$$

Εικόνα 12: Αρχικό στοιχείο δόμησης και τελικό στοιχείο δόμησης ύστερα από αποσύνθεση

3.6.6 Συνδυασμός Μορφολογικών Λειτουργιών

Υπάρχουν πολλοί άλλοι τύποι μορφολογικών λειτουργιών εκτός από την διαστολή και την διάβρωση. Παρόλα αυτά οι περισσότερες από αυτές είναι συνδυασμός των δύο αυτών βασικών μορφολογικών λειτουργιών.

3.6.6.1 Άνοιγμα (Opening)

Το άνοιγμα είναι η μορφολογική λειτουργία στην οποία η διάβρωση ακολουθείται από την διαστολή. Η γενική χρησιμότητα της λειτουργίας του ανοίγματος, είναι η μετακίνηση μικρών απομονωμένων αντικειμένων από το προσκήνιο της εικόνας στο παρασκήνιο, δηλαδή η αφαίρεσή τους από την εικόνα που επεξεργαζόμαστε. Μαθηματικά η λειτουργία του ανοίγματος εκφράζεται ως:

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

3.6.6.2 Κλείσιμο (Closing)

Το κλείσιμο είναι εκείνη η μορφολογική λειτουργία στην οποία η διαστολή ακολουθείται από την διάβρωση. Η γενική χρησιμότητα της λειτουργίας του κλεισίματος είναι, το γέμισμα μικρών οπών, η συγχώνευση και το κλείσιμο μικρών κενών χωρίς να αλλάξει η μορφή του αντικειμένου. Μαθηματικά η λειτουργία του κλεισίματος εκφράζεται ως:

$$A \bullet B = (A \oplus B) \ominus B$$

3.6.6.3 Εξαγωγή Περιγράμματος

Μπορούμε να καθορίσουμε το όριο ενός αντικειμένου, αφού προηγηθεί η λειτουργία της διάβρωσης με ένα κατάλληλο μικρό στοιχείο δομής, στη συνέχεια το αποτέλεσμα που προκύπτει αφαιρείται από την αρχική εικόνα. Δηλαδή εκφρασμένη μαθηματικά έχουμε:

$$A_p = A - A \ominus B$$

3.6.6.4 Γέμισμα Περιοχής

Οι δυαδικές εικόνες συνήθως εμφανίζονται σε εφαρμογές επεξεργασίας εικόνας, ως αποτέλεσμα κατωφλιού ή άλλης διαδικασίας τμηματοποίησης μιας έγχρωμης εικόνας, ή μιας εικόνας σε κλίμακα του γκρι. Το αποτέλεσμα αυτής της μετατροπής συνήθως δεν είναι το επιθυμητό, λόγω των διακυμάνσεων του φωτισμού, με αποτέλεσμα να έχουμε τρύπες και κενά. Η πλήρωση αυτών των οπών είναι πολύ σημαντικό βήμα έτσι ώστε οι πληροφορίες που θα πάρουμε από την μορφολογική επεξεργασία να είναι αποτελεσματικές.

Η πλήρωση κενών λειτουργεί με τον αρχικό εντοπισμό ενός εικονοστοιχείου X_k , μέσα στην τρύπα, στη συνέχεια η περιοχή αυτή μεγαλώνει χρησιμοποιώντας την λειτουργία της διαστολή. Μετά από κάθε διαστολή γίνεται η λογική πράξη της σύζευξης (AND) με το λογικό συμπλήρωμα της εικόνας. Έτσι με αυτόν τον τρόπο επιτρέπουμε να μεγαλώσει η περιοχή αυτή έως ότου συναντήσει τα όρια του αντικειμένου. Η ανίχνευση του ορίου του αντικειμένου είναι απαραίτητη για την λειτουργία της πλήρωσης κενών. Μαθηματικά η λειτουργία αυτή εκφράζεται ως:

$$X_k = (X_{k-1} \oplus B) \cap A^c \quad k = 1, 2, 3, \dots$$

Όπου B είναι στο στοιχείο δόμησης.

3.6.7 Εξαγωγή Και Επισήμανση Συνδεδεμένων Στοιχείων

Το σύνολο των εικονοστοιχείων που είναι συνδεδεμένα σε ένα δεδομένο εικονοστοιχείο ονομάζεται συνδεδεμένο στοιχείο αυτού του εικονοστοιχείου ή αντικείμενο.

Μια ομάδα εικονοστοιχείων που είναι συνδεδεμένα μεταξύ τους με αυτόν τον τρόπο, διαφοροποιούνται από τα άλλα δίνοντάς τους μια μοναδική επισήμανση. Η διαδικασία αυτή οδηγεί σε μια νέα εικόνα στην οποία οι συνδεδεμένες ομάδες εικονοστοιχείων (αντικείμενα) έχουν διαφορετική τιμή από τις άλλες. Οι τιμές που παίρνουν είναι διαδοχικές ακέραιες τιμές, ξεκινώντας από το 1. Την τιμή 0 την έχουν τα εικονοστοιχεία του παρασκηνίου. Δηλαδή αν έχουμε μια εικόνα με τρία ξεχωριστά αντικείμενα, τα

εικονοστοιχεία του πρώτου έχουν όλα τιμή ίση με 1, του δεύτερου ίση με 2 και του τρίτου ίση με 3 αντίστοιχα.

Ο αλγόριθμος της μορφολογικής αυτής λειτουργίας είναι παρόμοιος με αυτόν του γεμίσματος περιοχής. Η διαδικασία εύρεσης όλων των εικονοστοιχείων ενός αντικειμένου μπορεί να επιτευχθεί χρησιμοποιώντας μια επαναληπτική διαδικασία όπου με μαθηματική μορφή εκφράζεται ως:

$$X_k = (X_{k-1} \oplus B) \cap A, \quad k = 1, 2, 3, \dots$$

Όπου A είναι η εικόνα που αποτελείται από ένα ή περισσότερα αντικείμενα και B το στοιχείο δόμησης, όπου διαφέρει ανάλογα με την συνδεσιμότητα που θέλουμε να χρησιμοποιούμε για την εύρεση των αντικειμένων.

Η επαναληπτική διαδικασία τερματίζει όταν ο αριθμός των εικονοστοιχείων που ανήκουν στο συγκεκριμένο αντικείμενο είναι ίδιος με αυτόν της προηγούμενης επανάληψης. Η ίδια επαναληπτική διαδικασία εκτελείται και για την εύρεση των εικονοστοιχείων των άλλων πιθανών αντικειμένων. Αφού βρεθούν όλα τα εικονοστοιχεία, έχουμε τερματισμό της διαδικασίας.

Στο παρακάτω παράδειγμα βλέπουμε το αποτέλεσμα της μορφολογικής λειτουργίας για την εξαγωγή και την επισήμανση των συνδεδεμένων στοιχείων. Ο πρώτος πίνακας είναι η αρχική δυαδική εικόνα στην οποία θέλουμε να βρούμε τα αντικείμενα και ο δεύτερος πίνακας είναι η εικόνα μετά την επεξεργασία. Όπως παρατηρούμε η εικόνα αποτελείται από τρία ξεχωριστά αντικείμενα.

[1	1	1	0	0	0	0	0	0	1	1	1	0	0	0	0	0
1	1	1	0	1	1	0	0	0	1	1	1	0	2	2	0	0
1	1	1	0	1	1	0	0	0	1	1	1	0	2	2	0	0
1	1	1	0	0	0	1	0	0	1	1	1	0	0	0	3	0
1	1	1	0	0	0	1	0	0	1	1	1	0	0	0	3	0
1	1	1	0	0	0	1	0	0	1	1	1	0	0	0	3	0
1	1	1	0	0	1	1	0	0	1	1	1	0	0	3	3	0
1	1	1	0	0	0	0	0]	0	1	1	1	0	0	0	0	0

ΚΕΦΑΛΑΙΟ 4 Εντοπισμός του Μετώπου Ροής σε Επίπεδα Τεμάχια δύο Εισόδων Ρητίνης

4.1 Εισαγωγή

Στη συγκεκριμένη εργασία θεωρούμε δύο εισόδους ροής ρητίνης, οι οποίες ελέγχονται από μια πνευματική βαλβίδα η κάθε μία. Η κάθε βαλβίδα είναι υπεύθυνη για τον εμποτισμό συγκεκριμένης περιοχής του υφάσματος, η οποία εξαρτάται κάθε φορά από το σχήμα και την επιφάνεια του υφάσματος προς εμποτισμό. Έτσι, αναπτύσσονται και τα αντίστοιχα μέτωπα ροής.

4.1.1 MATLAB ως Γλώσσα Προγραμματισμού

Για την ανίχνευση του μετώπου ροής της ρητίνης, αναπτύχθηκε κώδικας στη MATLAB. Η MATLAB είναι μια γλώσσα υψηλής-απόδοσης 4^{ης} γενιάς. Ενσωματώνει υπολογισμό, απεικόνιση και προγραμματισμό σε εύχρηστο περιβάλλον όπου τα προβλήματα και οι λύσεις εκφράζονται με απλές μαθηματικές εκφράσεις. Μπορεί να υποστηρίξει προγράμματα γραμμένα σε άλλες γλώσσες όπως, C, C++, Java, Fortran και Python. Η βασική δομή δεδομένων που χρησιμοποιεί η MATLAB είναι οι πίνακες, πράγμα που την καθιστά ιδανική για την επεξεργασία εικόνας, αφού και η εικόνα δεν είναι τίποτα άλλο παρά ένας δισδιάστατος ή τρισδιάστατος πίνακας, ανάλογα με τον τύπο, δυαδική/γκρι απόχρωσης και έγχρωμη αντίστοιχα.

4.1.2 Εργαλειοθήκη Επεξεργασίας Εικόνας (Image Processing Toolbox)

Τις ειδικές εντολές που χρειαζόμαστε για την πραγματοποίηση της ανίχνευσης του μετώπου ροής παρέχει η εργαλειοθήκη του MATLAB για την επεξεργασία εικόνας, Image Processing Toolbox. Η Εργαλειοθήκη Επεξεργασίας Εικόνας παρέχει ένα πλήρες σύνολο αλγορίθμων, συναρτήσεων και εφαρμογών επεξεργασίας εικόνας για:

- Ανάλυση της εικόνας, συμπεριλαμβανομένης της κατάτμησης, της μορφολογίας των στατιστικών στοιχείων και της μέτρησης αυτής.
- Βελτίωση της εικόνας, το φιλτράρισμα και την θόλωση ή μη αυτής.
- Χρήση γεωμετρικών μετασχηματισμών, ρύθμιση της έντασης της εικόνας με βάση των μεθόδων καταχώρησης
- Μετασχηματισμός εικόνας συμπεριλαμβανομένων τους FFT, DCT Radon και της μεθόδου μέσω προβολών

- Διαχείριση μεγάλων ρών εικόνων, και προβολής πολλαπλών διαστάσεων
- Εφαρμογές για την απεικόνιση εικόνων και βίντεο
- Δυνατότητα ταυτόχρονης λειτουργίας πολλών GPU
- Τέλος, υποστήριξη δημιουργίας κώδικα C

4.2 Προ-Επεξεργασία Εικόνας

4.2.1 Εισαγωγή

Για την ανίχνευση του μετώπου της ροής είναι αναγκαία η προ-επεξεργασία της εικόνας που λαμβάνουμε από την κάμερα έτσι ώστε, οι ακμές που θα βρεθούν μετά την εφαρμογή συγκεκριμένου αλγορίθμου ανίχνευσης ακμών, να αντιστοιχούν στα πραγματικά μέτωπα ροής ρητίνης. Η πιθανότητα μη επιθυμητών ακμών είναι αρκετά μεγάλη, και είναι ένα από τα προβλήματα που έπρεπε να αντιμετωπίσουμε. Ο λόγος είναι ότι η κάμερα δεν έχει άμεση οπτική επαφή με την ρητίνη, αφού στη διαδικασία της VARI εκτός από το ύφασμα τοποθετούνται κάποια ειδικά υλικά όπως το αποκολλητικό ύφασμα (Peel Ply), το αφρώδες δικτυωτό πλέγμα (Peel Ply Econostich), το διάτρητο φιλμ (Release Film), και τέλος η σακούλα κενού. Μάλιστα, ειδικά λόγω του ότι η σακούλα κενού 'τσακίζει' κατά την δημιουργία κενού μέσα στο καλούπι, ο αλγόριθμος ανίχνευσης ακμών, όπως ήταν αναμενόμενο, ανίχνευε τις τσακίσεις της σακούλας κενού ως ακμές.

Έτσι γίνεται κατανοητό πως με απλή χρήση αλγορίθμου ανίχνευσης ακμών, θα ανιχνεύαμε ακμές οι οποίες δεν μας ενδιαφέρουν. Επιπλέον υπήρχε ο κίνδυνος να μην ανιχνευτεί καθόλου το μέτωπο της ροής της ρητίνης, αφού οι ακμές του μετώπου της ροής ήταν πιο εξασθενημένες σε σύγκριση με τις ακμές που δημιουργούνταν λόγω της σακούλας κενού.

Άλλος ένας σημαντικός παράγοντας είναι ο φωτισμός, αφού ανομοιόμορφος φωτισμός υπάρχει η πιθανότητα να δημιουργεί σκιές, οι οποίες ανιχνεύονται από τους αλγορίθμους ανίχνευσης ακμών, ως ακμές. Αυτό το πρόβλημα εκτός από τον σωστό φωτισμό, μπορεί να αντιμετωπιστεί με την σωστή επεξεργασία εικόνας.

Τα στάδια επεξεργασίας της αρχικής εικόνας που λαμβάνεται από την κάμερα περιγράφονται παρακάτω.

4.2.2 Περικοπή (crop)

Η εικόνα που λαμβάνουμε από την κάμερα είναι έγχρωμη με ανάλυση 1280x720 pixels. Λόγω του ότι στην εικόνα υπάρχουν στοιχεία τα οποία δεν μας είναι χρήσιμα για την ανίχνευση του μετώπου ροής, όπως το περίγραμμα της ειδικής σφραγιστικής ταινίας για

την αεροστεγή σφράγιση της σακούλας κενού με το καλούπι, τα σπιδράλ που έχουν τοποθετηθεί για την διευκόλυνση της διάδοσης της ρητίνης, οι είσοδοι/έξοδοι ρητίνης, αλλά και το περιβάλλον γύρω από το καλούπι, περικόπτουμε την εικόνα κατάλληλα έτσι ώστε, το τελικό αποτέλεσμα να είναι εικόνα εστιασμένη σε συγκεκριμένη περιοχή ενδιαφέροντος, που εμπεριέχει το ύφασμα.

Στη MATLAB η περικοπή της αρχικής εικόνας πραγματοποιείται εκτελώντας την συνάρτηση *imcrop(image,rect)*. Χρησιμοποιώντας την *imcrop* η εικόνα που περικόπτεται έχει ορθογώνιο σχήμα. Τα ορίσματα που εισάγονται είναι: (i) η εικόνα σε μορφή πίνακα *image*, η οποία μπορεί να είναι οποιασδήποτε μορφής και (ii) οι διαστάσεις και η θέση του ορθογωνίου που θα γίνει η περικοπή *rect*, όπου *rect*=[*xmin ymin width height*]. Η επιλογή γίνεται κάθε φορά από τον χρήστη αναλόγως με το καλούπι και το ύφασμα που έχουμε. Η συνάρτηση *imcrop* επιστρέφει μια εικόνα ίδιου τύπου σε μορφή πίνακα, με διαφορετική ανάλυση, μικρότερη από την αρχική. Η εφαρμογή της *imcrop* φαίνεται παρακάτω.



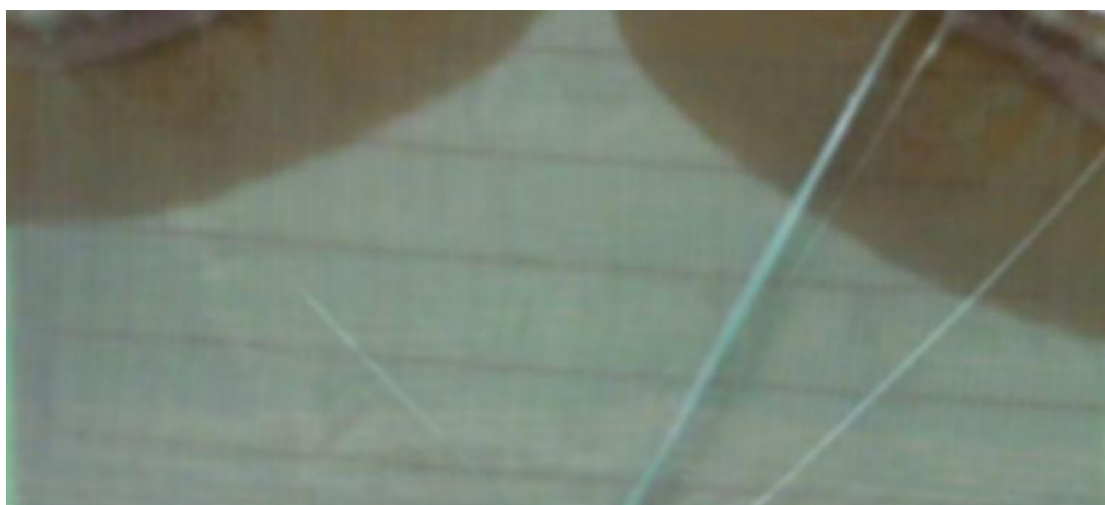
Εικόνα 13: Αρχική Εικόνα (από κάμερα)



Εικόνα 14: Τελική εικόνα μετά από *imcrop*

4.2.3 Φιλτράρισμα εικόνας χρησιμοποιώντας το φίλτρο Gauss

Στην εικόνα που έχουμε πάρει ύστερα από την εκτέλεση της `imcrop`, εφαρμόζουμε το φίλτρο Gauss. Χρησιμοποιώντας το φίλτρο Gauss, οι έντονες αλλαγές της φωτεινότητας που υπάρχουν μεταξύ των εικονοστοιχείων εξομαλύνονται. Στη MATLAB το φιλτράρισμα πραγματοποιείται χρησιμοποιώντας τη συνάρτηση `imgaussfilt(imgg,sigma)`. Τα ορίσματα που εισάγονται είναι η εικόνα σε μορφή πίνακα, και η τυπική απόκλιση σ . Η τυπική απόκλιση που έχουμε επιλέξει ύστερα από δοκιμές είναι ίση με $\sigma=2$. Το αποτέλεσμα του φίλτρου Gauss φαίνεται στην παρακάτω εικόνα.



Εικόνα 15: Φίλτρο Gauss

4.2.4 Μετατροπή Εικόνας σε Απόχρωση του Γκρι και Δυαδική

Για να εφαρμόσουμε τους αλγόριθμους ανίχνευσης ακμών για την εύρεση του μετώπου ροής της ρητίνης, πρέπει να μετατρέψουμε την εικόνα που έχουμε, από έγχρωμη σε εικόνα κλίμακας γκρι και στη συνέχεια σε δυαδική.

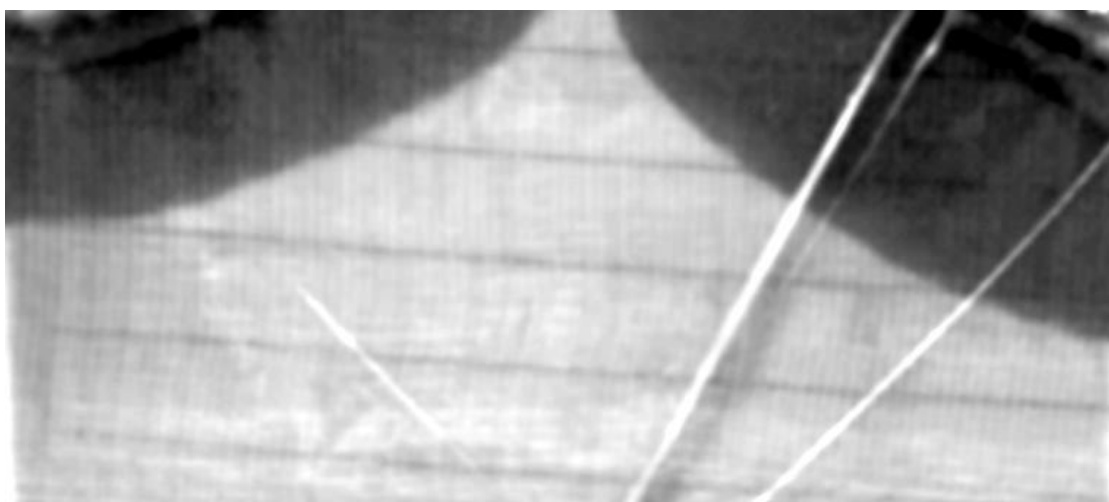
Η μετατροπή από έγχρωμη σε κλίμακα του γκρι πραγματοποιείται εξαλείφοντας τις πληροφορίες απόχρωσης (hue) και χρωματικής καθαρότητας (saturation), διατηρώντας την φωτεινότητα. Όπως έχουμε αναφέρει, η έγχρωμη εικόνα μπορεί να θεωρηθεί ότι αποτελείται από τρεις εικόνες γκρι απόχρωσης, όπου η κάθε εικόνα 'γκρι απόχρωσης' είναι υπεύθυνη για μια από τις τρεις αποχρώσεις, της κόκκινης (Red) της πράσινης (Green) και της μπλε (Blue). Η μετατροπή γίνεται αλλάζοντας την τιμή της φωτεινότητας του κάθε εικονοστοιχείου με την τιμή που υπολογίζεται και από τις τρεις τιμές φωτεινότητας των πινάκων των τριών αποχρώσεων, πολλαπλασιασμένες, η κάθε μια, με έναν συγκεκριμένο συντελεστή βαρύτητας και παίρνοντας το άθροισμά τους, $newpixel=0.2989 * R + 0.5870 * G + 0.1140 * B$. Να σημειωθεί ότι οι συντελεστές βαρύτητας μπορούν να πάρουν και άλλες

τιμές ανάλογα με την μέθοδο υπολογισμού που θέλουμε να εφαρμόσουμε. Η μετατροπή στη MATLAB γίνεται χρησιμοποιώντας την συνάρτηση `rgb2gray(Iblur)`, όπου `Iblur` είναι ο πίνακας της έγχρωμης εικόνας που θέλουμε να μετατρέψουμε. Το αποτέλεσμα της `rgb2gray` φαίνεται παρακάτω.



Εικόνα 16: εικόνα σε απόχρωση του γκρι

Αφού η εικόνα μετατράπηκε σε εικόνα απόχρωσης γκρι, θα αυξήσουμε την αντίθεση της, έτσι ώστε το μέτωπο της ροής να φαίνεται εντονότερο. Για να το επιτύχουμε αυτό θα χρησιμοποιήσουμε την συνάρτηση της MATLAB την `imadjust(gray)` όπου σαν όρισμα εισάγουμε την εικόνα απόχρωσης γκρι. Η συνάρτηση αυτή αλλάζει τις τιμές των εντάσεων των εικονοστοιχείων, με τέτοιον τρόπο, έτσι ώστε το 1% των τιμών να είναι στις υψηλές και χαμηλές εντάσεις. Έτσι η εικόνα που εξάγεται έχει αυξημένη αντίθεση. Το αποτέλεσμα της `imadjust` φαίνεται παρακάτω.



Εικόνα 17: Αύξηση αντίθεσης με χρήση της `imadjust`

Για μετατροπή της εικόνας γκρι απόχρωσης σε δυαδική, χρησιμοποιούμε την συνάρτηση της MATLAB `imbinarize(contrast,Threshold)`, όπου ως ορίσματα έχουμε την εικόνα γκρι

απόχρωσης σε μορφή πίνακα την *contrast*, και μια τιμή κατωφλιού, όπου οι τιμές που είναι πάνω από αυτήν την τιμή γίνονται ίσες με 1 και όλες οι άλλες ίσες με 0. Με αυτόν τον τρόπο δημιουργείται η δυαδική εικόνα *wb*.

Η εύρεση της τιμής του κατωφλιού που χρησιμοποιούμε για την μετατροπή της εικόνας σε δυαδική, γίνεται με τη μέθοδο του Otsu, χρησιμοποιώντας την συνάρτηση της MATLAB *graythresh(contrast)*, εισάγοντας ως όρισμα την εικόνα αποχρώσεως γκρι. Το αποτέλεσμα φαίνεται παρακάτω.



Εικόνα 18: Δυαδική εικόνα με χρήση της *imbinarize*, και *Otsu's threshold*

4.2.5 Βελτίωση Εικόνας Χρησιμοποιώντας Μορφολογικές Λειτουργίες

Στην δυαδική εικόνα που έχει προκύψει παρατηρούμε ότι το αποτέλεσμα δεν είναι το επιθυμητό, καθώς λόγω των τσακισμάτων της σακούλας κενού έχουν δημιουργηθεί κενά. Για την πλήρωση των κενών αυτών θα χρησιμοποιηθεί ο μορφολογικός αλγόριθμος γεμίσματος μιας περιοχής, βλ. Κεφάλαιο 3.

Πριν εφαρμόσουμε τον αλγόριθμο αυτό, πρέπει να αντιστρέψουμε το προσκήνιο με το παρασκήνιο, αφού παρατηρούμε ότι η ροή της ρητίνης είναι χρώματος μαύρου, δηλαδή το εικονοστοιχείο έχει τιμή ίση με 0, επομένως βρίσκεται στο παρασκήνιο. Εάν εφαρμόζαμε τον αλγόριθμο γεμίσματος περιοχής στην υπάρχουσα εικόνα, το κενό θα θεωρείτο ως ροή ρητίνης, επομένως δεν θα είχαμε το επιθυμητό αποτέλεσμα. Στη MATLAB το πραγματοποιούμε μετατρέποντας τις τιμές των εικονοστοιχείων σε λογικές τιμές, και αφαιρώντας από το 1 την τιμή του κάθε εικονοστοιχείου. Έτσι έχουμε $wb = \text{logical}(1 - bw)$, όπου *bw* ο πίνακας της αρχικής δυαδικής εικόνας και *wb* ο πίνακας της νέας δυαδικής

εικόνας με αντίθετη τιμή εικονοστοιχείων. Το αποτέλεσμα της μετατροπής αυτής φαίνεται παρακάτω.



Εικόνα 19:Εικόνα με αλλαγή παρασκηνίου / προσκηνίου

Ση συνέχεια εκτελούμε τον αλγόριθμο για την πλήρωση κενών με τη συνάρτηση *imclose(wb,se)*. Τα ορίσματα που εισάγουμε είναι η εικόνα όπου θέλουμε να εφαρμόσουμε τον αλγόριθμο γεμίσματος, δηλαδή στη περίπτωσή μας η δυαδική εικόνα που έχει προκύψει. Το δεύτερο όρισμα είναι το στοιχείο δόμησης.

Για την δημιουργία του στοιχείου δόμησης χρησιμοποιούμε τη συνάρτηση της MATLAB *strel('arbitrary',R)*. Ως πρώτο όρισμα εισάγουμε την μορφή που θέλουμε να έχει το στοιχείο δόμησης ('diamond', 'disc', 'line', 'octagon', 'rectangle', 'square', 'cube', 'cuboid', 'sphere' ή 'arbitrary' που είναι ένας πίνακας που καθορίζουμε εμείς την γειτονιά). Το δεύτερο όρισμα είναι τα βασικά γεωμετρικά του χαρακτηριστικά για την κατασκευή του στοιχείου δόμησης.

Το στοιχείο δόμησης που θα χρησιμοποιήσουμε έχει σχήμα δίσκου ('disk') ακτίνας $R=10$ εικονοστοιχείων. Αν δεν μας καλύπτει η επιλογή αυτή μπορούμε να προσαρμόσουμε το στοιχείο δόμησης ανάλογα, με διαφορετική επιλογή σχήματος και μεγέθους γειτονιάς. Ο πίνακας του στοιχείου δόμησης καθώς και το αποτέλεσμα της χρήσης της *imclose* φαίνονται παρακάτω.

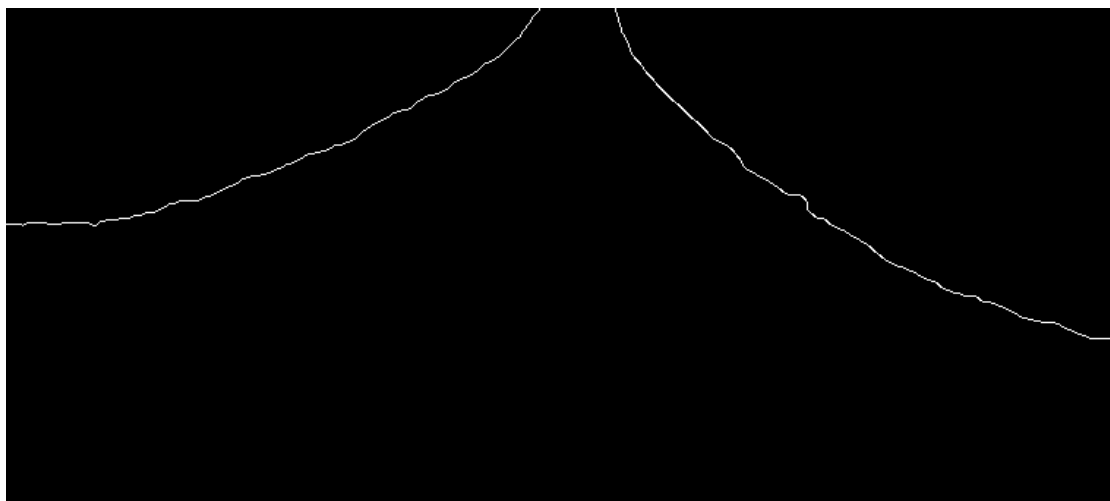


Εικόνα 22: Ανίχνευση Μετώπου Ροής Με Χρήση Αλγόριθμου Sobel

Παρατηρούμε ότι στην τελική εικόνα, έχει ανιχνευτεί και ακμή μικρού μεγέθους η οποία δεν ανήκει στο μέτωπο της ροής. Αυτό το πρόβλημα θα το αντιμετωπίσουμε χρησιμοποιώντας τον μορφολογικό αλγόριθμο του ανοίγματος, εκτελώντας την συνάρτηση της MATLAB `bwareaopen(bbww,P,conn)`. Τα ορίσματα είναι η εικόνα προς μορφοποίηση `bbww`, το μέγιστο όριο εικονοστοιχείων `P` (αν η ακμή έχει λιγότερα εικονοστοιχεία από το όριο αυτό θεωρείται ανεπιθύμητη) και το είδος της συνδεσιμότητας.

Στην περίπτωση μας μετά από δοκιμές που έγιναν και σε άλλα στάδια του μετώπου της ροής, καθώς κάθε φορά έχουμε καινούρια εικόνα με διαφορετικό μέτωπο ροής, καταλήξαμε το όριο των εικονοστοιχείων να είναι `P=120` και το είδος της συνδεσιμότητας `8` στοιχείων.

Το αποτέλεσμα της `bwareaopen` φαίνεται παρακάτω.



Εικόνα 23: Εικόνα μετά την χρήση μορφολογικού αλγόριθμου ανοίγματος `bwareaopen`

Όλα τα βήματα που αναπτύχθηκαν για τον εντοπισμό του μετώπου της ροής, εφαρμόζονται κάθε φορά στην καινούρια εικόνα που λαμβάνει η MATLAB από την κάμερα.

4.2.7 Κώδικας MATLAB

Ο κώδικας που έχει γραφτεί στη MATLAB για την υλοποίηση των παραπάνω είναι το υποπρόγραμμα `processimg2roes.m`. Οι μεταβλητές που εισάγονται είναι η αρχική εικόνα από την κάμερα σε μορφή πίνακα `image`, οι συντεταγμένες καθώς το πλάτος και το ύψος για την περικοπή της αρχικής εικόνας `xmin, ymin, width, height`. Η μεταβλητή που εξάγεται είναι η δυαδική εικόνα με εντοπισμένο το μέτωπο ροής της ρητίνης σε μορφή πίνακα.

```
%% πρόγραμμα εντοπισμού μετώπου ροής
function [ edgeopen ] =
processimg2roes(image,xmin,ymin,width,height)

% imcrop
imgg=imcrop(image,[xmin ymin width height]);
%imwrite(imgg,'image.jpg')

%% blur
Iblur = imgaussfilt(imgg,2);
%imwrite(Iblur,'Iblur.jpg')

%% μετατροπή σε gray scale
gray=rgb2gray(Iblur);
%imwrite(gray,'gray.jpg')

%% αλλάζω contrast
contrast=imadjust(gray);
%imwrite(contrast,'contrast.jpg')
% μετατροπή σε binary
level1 = graythresh(contrast);
% level=0.58
bw=imbinarize(contrast,level1);
%imwrite(bw,'bw.jpg')

%% μετατροπή bw σε wb
wb = logical(1 - bw);
%imwrite(wb,'wb.jpg')

%% γέμισμα κενών
se = strel('disk',10);
closeWB = imclose(wb,se);
% imshow(closeWB)
%imwrite(closeWB,'closeWB.jpg')

%% εύρεση edge
bbww=edge(closeWB);
%imwrite(bbww,'edge.jpg')
```

```

%% αφαίρεση μικρών ακμών bwareaopen\bwareafilt
edgeopen=bwareaopen(bbww,120);
%imwrite(edgeopen,'edgeopen.bmp');
%imshow(edgeopen);

%%%%%%%%%%
end

```

4.3 Εύρεση Θέσης του Μετώπου Ροής της ρητίνης

4.3.1 Εισαγωγή

Πλέον έχουμε μια δυαδική εικόνα με εντοπισμένο το μέτωπο ροής της ρητίνης χρησιμοποιώντας το υποπρόγραμμα `processimg2roes.m`. Σε αυτό το βήμα θα εντοπίσουμε την θέση του μετώπου ροής, δηλαδή τις συντεταγμένες των εικονοστοιχείων από τα οποία αποτελείται το εντοπισμένο μέτωπο ροής. Το σύστημα των συντεταγμένων που θα χρησιμοποιήσουμε είναι το σύστημα των εικονοστοιχείων.

Στο υποπρόγραμμα που έχουμε αναπτύξει για να επιτύχουμε τη λειτουργία αυτή, χρησιμοποιούμε μια συνθήκη για το αν έχουμε δύο ξεχωριστά μέτωπα ροής ή ένα ενιαίο. Ο λόγος είναι ότι στη διάταξή μας έχουμε δύο εισόδους ρητίνης, επομένως κατά την έναρξη της κατεργασίας VARI έχουμε δύο μέτωπα ροής. Στη συνέχεια κατά την πρόοδο του εμποτισμού, τα δύο μέτωπα ροής ενώνονται σε ένα ενιαίο μέτωπο.

4.3.2 Εύρεση Συντεταγμένων

Με τη χρήση της συνάρτησης `bwconncomp` της MATLAB, βρίσκουμε τα στοιχεία που είναι συνδεδεμένα μεταξύ τους, επομένως βρίσκουμε το πλήθος των μετώπων της ροής. Τα ορίσματα που εισάγονται είναι η δυαδική εικόνα `linenew` και το είδος της συνδεσιμότητας, στη περίπτωσή μας 8 στοιχείων. Η έξοδος της `bwconncomp` είναι τα συνδεδεμένα αντικείμενα ως πίνακας δομής (structure array) με τέσσερα πεδία:

- το πρώτο πεδίο `connectivity` αναφέρει το είδος της συνδεσιμότητας των στοιχείων
- το δεύτερο πεδίο `Imagesize` αναφέρει το μέγεθος της εικόνας,
- το τρίτο πεδίο `NumObjects` επιστρέφει τον αριθμό των συνδεδεμένων στοιχείων, δηλαδή το πλήθος των μετώπων, το οποίο χρησιμοποιούμε στη συνθήκη για τον διαχωρισμό των περιπτώσεων των μετώπων ροής
- το τέταρτο πεδίο `PixelIdxList` περιέχει πλήθος κελιών όσο είναι το πλήθος των αντικειμένων. Το κάθε κελί περιέχει τους δείκτες των εικονοστοιχείων για κάθε ξεχωριστό αντικείμενο σε μορφή διανύσματος, επομένως γνωρίζουμε και τις

συντεταγμένες του μετώπου της ροής.

Παρόμοια λειτουργία με την `bwconncomp` έχει η `bwlabel(linenew,conn)`, η οποία δημιουργεί έναν πίνακα με όνομα `label` όπου τα εικονοστοιχεία του κάθε αντικειμένου έχουν διαφορετική τιμή. Οι πιθανές τιμές είναι ακέραιες ξεκινώντας από την τιμή 1. Στην περίπτωση μας όταν έχουμε δύο μέτωπα ροής, τα εικονοστοιχεία του πρώτου μετώπου ροής έχουν τιμή ίση με 1, ενώ τα εικονοστοιχεία του δεύτερου μετώπου έχουν τιμή ίση με 2. Όταν έχουμε μοναδικό μέτωπο ροής όπως είναι αναμενόμενο έχει τιμή ίση με 1. Ο πίνακας `label` που δημιουργείται από την `bwlabel` θα χρησιμοποιηθεί σε επόμενο βήμα.

Να σημειωθεί ότι στην περίπτωση που δεν έχει ανιχνευθεί κάποιο μέτωπο ροής διότι είτε δεν έχει ξεκινήσει η έγχυση ακόμα είτε το μέτωπο της ρητίνης είναι εκτός της περιοχής ενδιαφέροντος που έχουμε ορίσει, τα αντικείμενα που εντοπίζονται από την `bwconncomp` και `bwlabel` είναι αρκετά παραπάνω από δύο. Πρόκειται για ακμές των διάφορων επιπέδων υλικών που έχουν τοποθετηθεί στο καλούπι καθώς και το ανάγλυφο επίπεδο αυτών. Αυτό δεν δημιουργεί κανένα πρόβλημα, καθώς το υποπρόγραμμα `processimg2roes.m`, είναι έτσι σχεδιασμένο ώστε όταν εκχύνεται η ρητίνη, να εντοπίζεται μόνο αυτή.

4.3.3 Μορφολογική Λειτουργία Διαστολής

Σε προκαταρκτικές δοκιμές παρατηρήθηκε ότι αρκετές φορές για κάποιο από τα μέτωπα της ροής, οι συναρτήσεις `bwconncomp` και `bwlabel` εντόπιζαν περισσότερα από ένα αντικείμενα. Ο λόγος είναι ότι δεν ικανοποιούνταν η συνθήκη γειτονιάς 8 στοιχείων, δηλαδή δεν υπήρχε κάποιο εικονοστοιχείο με τιμή ίση με 1 ακριβώς δίπλα στο σύνολο των συνδεδεμένων εικονοστοιχείων με τιμή ίση με 1, αλλά ένα εικονοστοιχείο με τιμή ίση με 0.

Για να λυθεί το πρόβλημα αυτό πριν εφαρμόσουμε τις `bwconncomp` και `bwlabel`, εφαρμόζουμε τον μορφολογικό αλγόριθμο της διαστολής (`dilation`) με στοιχείο δόμησης :

$$\begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

έτσι ώστε στην περίπτωση που υπάρχει κάποιο εικονοστοιχείο με τιμή ίση 0 ανάμεσα στα εικονοστοιχεία με τιμή 1, αυτή να αλλάξει σε 1. Στη MATLAB η διαστολή γίνεται χρησιμοποιώντας την συνάρτηση `imdilate(line,nhood)`. Πρώτο όρισμα είναι η εικόνα που θέλουμε να μορφοποιήσουμε σε μορφή πίνακα `line`, δηλαδή στην περίπτωση μας ο πίνακας `edgeopen` που έχει δημιουργηθεί στο υποπρόγραμμα `processimg2roes.m`. Δεύτερο όρισμα είναι το στοιχείο δόμησης `nhood` σε μορφή πίνακα. Το αποτέλεσμα της `imdilate` φαίνεται παρακάτω.



Εικόνα 24: Εφαρμογή του μορφολογικού αλγορίθμου Διαστολής *imdilate*

4.3.4 Εξαγωγή Συντεταγμένων

Για την εξαγωγή των συντεταγμένων των μετώπων ροής της ρητίνης χρησιμοποιούμε την εντολή της MATLAB *regionprops (ccc, properties)*, όπου *ccc* είναι ο πίνακας δομής που έχει δημιουργηθεί από την *bwconncomp*, και στο *properties* ορίζουμε το είδος της πληροφορίας που θέλουμε να πάρουμε από τα συνδεδεμένα στοιχεία. Στην περίπτωσή μας, θέλουμε τις συντεταγμένες των εικονοστοιχείων άρα ορίζουμε την ιδιότητα *'pixellist'*. Το αποτέλεσμα της *regionprops (ccc, 'pixellist')*, είναι ένας πίνακας δομής με πλήθος πεδίων το οποίο εξαρτάται από το πλήθος των αντικειμένων που έχουν βρεθεί. Στη περίπτωσή μας όταν έχουμε δύο μέτωπα ροής έχουμε δύο πεδία, ενώ όταν έχουμε ένα μέτωπο ροής έχουμε ένα πεδίο. Το κάθε πεδίο είναι ένας πίνακας $k \times 2$, όπου k το πλήθος των εικονοστοιχείων που ανήκει στο μέτωπο της ροής. Στη πρώτη στήλη έχουμε τις συντεταγμένες κατά τον άξονα x και στην δεύτερη στήλη τις συντεταγμένες κατά τον άξονα y , για κάθε εικονοστοιχείο. Για τον διαχωρισμό, εάν έχουμε δύο μέτωπα ή ένα μέτωπο ροής της ρητίνης, χρησιμοποιούμε την εντολή συνθήκης *if-elseif*. Το πλήθος των μετώπων της ροής τα γνωρίζουμε από το πεδίο *NumObjects* του πίνακα δομής *ccc*. Επομένως εάν *ccc.NumObjects==2* είναι αληθές βρίσκουμε συντεταγμένες και για τα δύο μέτωπα, αλλιώς αν *ccc.NumObjects==1*, βρίσκουμε συντεταγμένες για ένα μέτωπο ροής της ρητίνης.

4.3.5 Ορισμός Σημείων στο Μέτωπο Ροής

Σε κάθε μέτωπο ροής της ρητίνης που εντοπίζεται θα ορίσουμε τρία σημεία τα οποία θα βοηθήσουν για τον έλεγχο της έγχυσης της ρητίνης μέσω των βαλβίδων. Τα τρία αυτά σημεία θα είναι τα δύο ακραία και το μεσαίο του μετώπου της ροής, *p1oint1*, *p1oint3* και *p1oint2* αντίστοιχα. Τα ακραία σημεία *p1oint1*, *p1oint3* τα

βρίσκουμε εκτελώντας τις συναρτήσεις της MATLAB *min* και *max* για τις συντεταγμένες στον άξονα x. Ως αποτελέσματα παίρνουμε έναν πίνακα 1x2 με την ελάχιστη/μέγιστη τιμή και τον δείκτη του πίνακα που βρίσκονται οι τιμές αυτές. Γνωρίζοντας τον δείκτη του πίνακα στον x βρίσκουμε και την τιμή στον y. Για να ορίσουμε το τρίτο σημείο `p1oint3`, ο δείκτης για να βρούμε τις συντεταγμένες στον x και στον y άξονα είναι $idx2=round((idx3-idx1)/2)$, όπου `idx3` και `idx1` οι δείκτες που επιστρέφουν οι *max* και *min* αντίστοιχα. Με τη συνάρτηση `round` της MATLAB, στρογγυλοποιούμε την τιμή στον πλησιέστερο ακέραιο, αφού οι δείκτες των πινάκων είναι ακέραιοι αριθμοί.

Στη περίπτωση με τα δύο μέτωπα ροής τα σημεία του δεύτερου μετώπου `p2oint1`, `p2oint3` και `p2oint2` βρίσκονται αντίστοιχα.

4.3.6 Κώδικας MATLAB

Ο κώδικας που έχει γραφτεί στη MATLAB για την υλοποίηση των παραπάνω είναι το υπο-πρόγραμμα `vgrammh2roes.m`. Η μεταβλητή `line` που εισάγεται είναι ο πίνακας `edgeopen` που έχει δημιουργηθεί στο υπο-πρόγραμμα `processing2roes.m`. Οι μεταβλητές που εξάγονται είναι: η `linenew` η οποία έχει δημιουργηθεί από την `imdilate`, η μεταβλητή `label` που έχει δημιουργηθεί από την `bwlabel`, ο πίνακας δομής `ccc` που έχει δημιουργηθεί από την `bwconncomp`, οι συντεταγμένες των εικονοστοιχείων των δύο μετώπων ροής `x1`, `x2`, `y1`, `y2`, τέλος τα τρία σημεία που έχουν οριστεί για τα δύο μέτωπα ροής `p1oint1`, `p1oint2`, `p1oint3`, `p2oint1`, `p2oint2`, `p2oint3`

```
%% ευρεση γραμμης
function
[linenew,label,ccc,x1,x2,y1,y2,p1oint1,p1oint2,p1oint3,p2oint1
,p2oint2,p2oint3]=vgrammh2roes(line)

x1=0 ; y1=0; x2=0 ; y2=0;
p1oint1=[0 0] ;p1oint3=[0 0];p1oint2=[0 0];
p2oint1=[0 0];p2oint3=[0 0];p2oint2=[0 0];
idx1=1;idx2=1;idx3=1;

%% imdilate me pinaka nhood [0 1 0;1 1 1;0 1 0]
nhood=[0 1 0;1 1 1;0 1 0];
linenew=imdilate(line,nhood);
% %imwrite(linenew,'linenew.jpg');
% %figure,imshow(linenew)
ccc=bwconncomp(linenew);

%% Συνθήκη για το αν έχω 2 ροές ή μια ροή, αφού μετά από λίγο
θα γίνει μία άρα αλλάζει το ccc, και το label.
```

```

if ccc.NumObjects==2
% 1 τρόπος με regionprops
stats1 = regionprops(ccc,'pixellist');
pixels1=stats1(1).PixelList;
pixels2=stats1(2).PixelList;
x1=pixels1(:,1); x2=pixels2(:,1);
y1=pixels1(:,2); y2=pixels2(:,2);
% 2 τρόπος με bwlabel
label=bwlabel(linenew);
% [row1,col1]=find(label==1);
% [row2,col2]=find(label==2);
%%%%%%%%%%%% εύρεση 3 σημείων από το μέτωπο ροής, θα
%%%%%%%%%%%% χρησιμοποιηθεί για έλεγχο βαλβίδων.

%% 1 ροή
[p1x,idx1]=min(x1);
% p1y=row1(idx1);
p1oint1=[p1x y1(idx1)];
[p3x,idx3]=max(x1);
% p3y=row1(idx3);
p1oint3=[p3x y1(idx3)];
idx2=round((idx3-idx1)/2);
if idx2==0 idx2=1; end
p1oint2=[x1(idx2) y1(idx2)];

%% 2 ροή
[p2x,idx1]=min(x2);
p2oint1=[p2x y2(idx1)];
[p3x,idx3]=max(x2);
p2oint3=[p3x y2(idx3)];
idx2=round((idx3-idx1)/2);
if idx2==0 idx2=1; end
p2oint2=[x2(idx2) y2(idx2)];

elseif ccc.NumObjects==1
%1 τρόπος με regionprops
stats1 = regionprops(ccc,'pixellist');
pixels1=stats1(1).PixelList;
x1=pixels1(:,1);
y1=pixels1(:,2);
% 2 τρόπος με bwlabel
label=bwlabel(linenew);
% [row1,col1]=find(label==1);
%%%%%%%%%%%%
[p1x,idx1]=min(x1);
p1oint1=[p1x y1(idx1)];
[p3x,idx3]=max(x1);
p1oint3=[p3x y1(idx3)];
idx2=round((idx3-idx1)/2);
if idx2==0 idx2=1; end
p1oint2=[x1(idx2) y1(idx2)];

else
label=bwlabel(linenew);
end
end

```

4.4 Επεξεργασία Υφάσματος προς Εμποτισμό

4.4.1 Εισαγωγή

Στην ενότητα αυτή θα αναλυθούν τα βήματα που πραγματοποιούνται μέσω του υπο-προγράμματος που έχει δημιουργηθεί για

- Την οριοθέτηση της περιοχής ενδιαφέροντος
- την ανίχνευση του σχήματος του υφάσματος προς εμποτισμό
- την εύρεση των συντεταγμένων των εικονοστοιχείων του υφάσματος προς εμποτισμό
- την εκτέλεση του μορφολογικού αλγορίθμου γεμίσματος περιοχής στην δυαδική εικόνα που εμπεριέχει το περίγραμμα του υφάσματος προς εμποτισμό
- την δημιουργία σημείων ελέγχου, τα οποία θα χρησιμοποιηθούν για τον έλεγχο της ροής της ρητίνης από τον μικροελεγκτή Arduino

Να επισημανθεί πως οι εικόνες που παρουσιάζονται αφορούν ένα συγκεκριμένο σχήμα υφάσματος που έχει χρησιμοποιηθεί κατά τη διάρκεια των δοκιμών για την ανάπτυξη του κώδικα στη MATLAB. Παρόλα αυτά, ο κώδικας που έχει δημιουργηθεί λειτουργεί για οποιοδήποτε σχήμα υφάσματος.

4.4.2 Οριοθέτηση Περιοχής Ενδιαφέροντος

Όπως έχουμε αναφέρει στην ενότητα 4.3, ο χρήστης πρέπει να ορίσει μια περιοχή ενδιαφέροντος, έτσι ώστε η αρχική εικόνα που θα επεξεργάζεται η MATLAB κατά την διάρκεια της κατεργασίας VARI, να εμπεριέχει τα όρια του υφάσματος προς εμποτισμό και να είναι απαλλαγμένη από στοιχεία τα οποία δεν είναι χρήσιμα για την ανίχνευση του μετώπου ροής, όπως το περίγραμμα της ειδικής σφραγιστικής ταινίας για την αεροστεγή σφράγιση της σακούλας κενού με το καλούπι, τα σπирάλ που έχουν τοποθετηθεί για την διευκόλυνση της διόδου της ρητίνης, οι είσοδοι/έξοδοι ρητίνης, καθώς και το περιβάλλον γύρω από το καλούπι.

Για τον προσδιορισμό της περιοχής ενδιαφέροντος ο χρήστης ορίζει τρία σημεία στην εικόνα που έχει λάβει από την κάμερα εκτελώντας την συνάρτηση της MATLAB *snapshot(cam)*. Αυτή αποκτά το τρέχον frame από την κάμερα που έχουμε ορίσει στη MATLAB και το εκχωρεί στην μεταβλητή-πίνακα *img*. Η ανάλυση της εικόνας που εκχωρείται εξαρτάται από τις προεπιλεγμένες ρυθμίσεις της συνδεδεμένης κάμερας. Το όρισμα που εισάγουμε στην *snapshot* είναι το *object* με τα χαρακτηριστικά της κάμερας, το οποίο έχει δημιουργηθεί κατά την σύνδεση και την αναγνώριση της από τη MATLAB.

Αφού πλέον έχουμε την εικόνα υπό μορφή πίνακα, εκτελώντας την συνάρτηση της MATLAB `imshow(img)`, όπου ως όρισμα εισάγουμε την μεταβλητή `img`, εμφανίζεται η εικόνα που έχει αποκτηθεί από την `snapshot` στην οθόνη του υπολογιστή. Ζητείται από τον χρήστη να ορίσει τρία σημεία πάνω στην εικόνα που έχει εμφανιστεί. Το πρώτο σημείο είναι η αρχή της περιοχής ενδιαφέροντος, το δεύτερο σημείο είναι μετατοπισμένο κατά τον άξονα y έτσι ώστε να ορίσουμε το ύψος της περιοχής ενδιαφέροντος, και το τρίτο σημείο είναι μετατοπισμένο κατά τον άξονα x , έτσι ώστε να ορίσουμε το πλάτος της περιοχής ενδιαφέροντος.

Η επιλογή των τριών σημείων αυτών γίνεται χρησιμοποιώντας την συνάρτηση της MATLAB `ginput(n)`, η οποία μας επιτρέπει να επιλέξουμε τα σημεία αυτά με τη χρήση του ποντικιού. Το όρισμα που εισάγουμε είναι αριθμός των σημείων που θέλουμε να επιλέξουμε, στην περίπτωση μας $n=3$. Η `ginput` δημιουργεί δύο μεταβλητές διανύσματα n στοιχείων `xinitiall` και `yinitiall` που περιέχουν τις συντεταγμένες των σημείων στον άξονα x και στον άξονα y αντίστοιχα. Να σημειωθεί ότι το σύστημα των συντεταγμένων που χρησιμοποιείται εδώ και καθ' όλη την διάρκεια της διαδικασίας είναι αυτό των εικονοστοιχείων.

Στην συνέχεια από τα τρία αυτά σημεία, θα υπολογιστούν τα `xmin`, `ymin`, `width` και `height`, έτσι ώστε χρησιμοποιώντας τα ως όρισμα εισαγωγής της συνάρτησης της MATLAB `imcrop` να οριστικοποιηθεί η περιοχή ενδιαφέροντος, όπως έχουμε αναφέρει και κατά την ανάλυση της διαδικασίας για τον εντοπισμό του μετώπου της ροής. Έχουμε:

```
xmin=xinitiall(1), ymin=yinitiall(1)
```

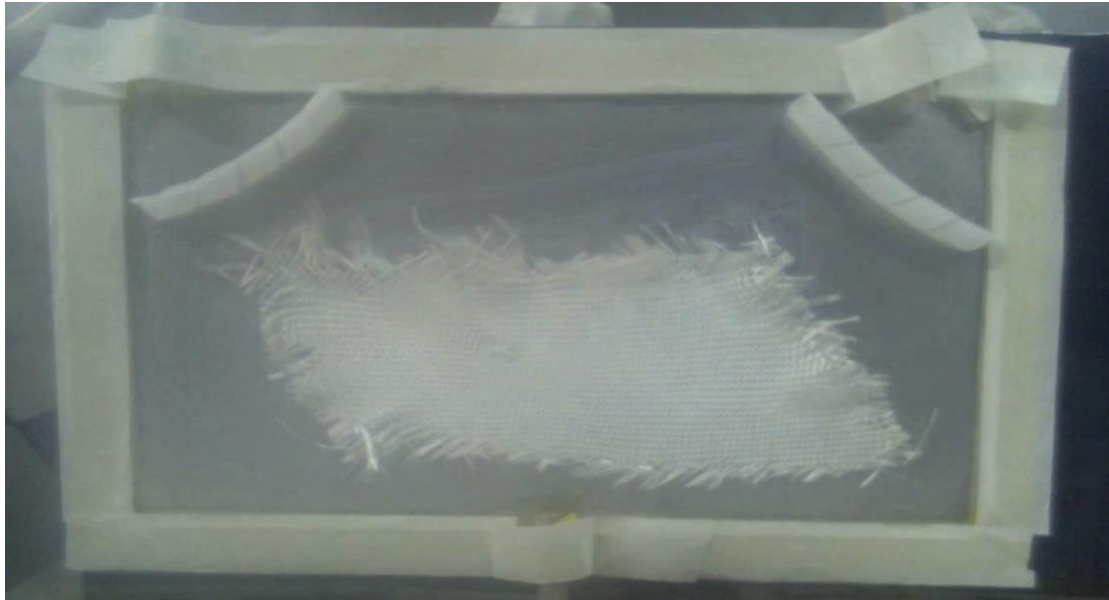
```
width= (xinitiall(3)-xinitiall(1))
```

```
height=(yinitiall(2)-yinitiall(1)).
```

4.4.3 Ανίχνευση του Σχήματος του Υφάσματος προς Εμποτισμό

Είναι σημαντικό να ανιχνεύσουμε το σχήμα του υφάσματος έτσι ώστε να γνωρίζουμε τις συντεταγμένες της τοποθεσίας του υφάσματος προς εμποτισμό, καθώς και πληροφορίες οι οποίες θα χρησιμεύσουν για τον προσδιορισμό των σημείων ελέγχουν την έγχυση της ρητίνης. Για την ανίχνευση του σχήματος του υφάσματος προς εμποτισμό, θα χρησιμοποιήσουμε τον αλγόριθμο ανίχνευσης ακμών του Sobel, που έχουμε χρησιμοποιήσει και για τον εντοπισμό του μετώπου ροής της ρητίνης. Πριν την εφαρμογή του αλγορίθμου Sobel, πρέπει η εικόνα που έχει ληφθεί από την κάμερα, να επεξεργαστεί καταλλήλως. Η διαδικασία είναι ακριβώς ίδια με αυτή που περιγράφηκε για τον εντοπισμό του μετώπου της ροής, δηλαδή :

- Περικοπή της αρχικής εικόνας εκτελώντας την συνάρτηση `imcrop(kimage,rect)`. Τα ορίσματα είναι: η εικόνα σε μορφή πίνακα `kimage`, και οι διαστάσεις και η θέση του ορθογωνίου που θα γίνει η περικοπή `rect`, όπου `rect=[xmin ymin widthheight]`. Η αρχική εικόνα και το αποτέλεσμα της `imcrop` φαίνονται παρακάτω.



Εικόνα 25: Αρχική Εικόνα (από κάμερα):



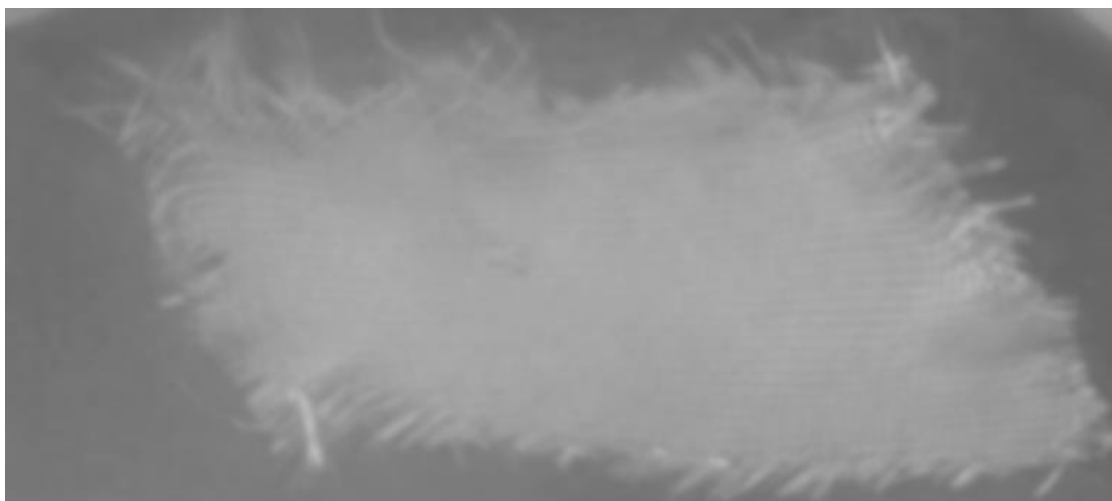
Εικόνα 26: Τελική εικόνα μετά από `imcrop`

- Εφαρμογή φίλτρου Gauss της εικόνας που δημιουργήθηκε με τη συνάρτηση `imgaussfilt(kimgg,sigma)`. Τα ορίσματα είναι η εικόνα σε μορφή πίνακα, και η τυπική απόκλιση σ , ίση με 2. Το αποτέλεσμα της `imgaussfilt` φαίνεται παρακάτω.



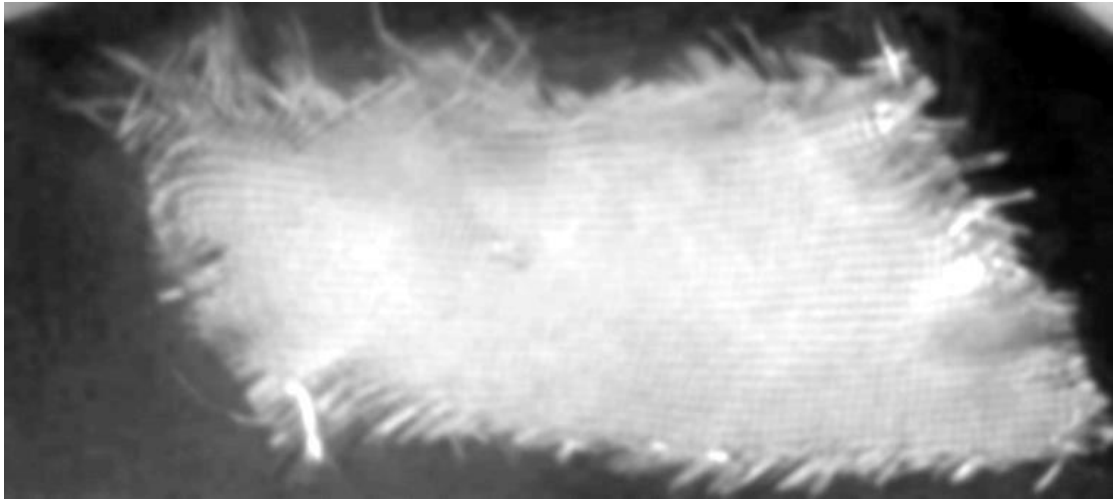
Εικόνα 27: Φίλτρο Gauss

- Μετατροπή της αρχικής εικόνας σε εικόνα γκρι ανάλυσης χρησιμοποιώντας την συνάρτηση της MATLAB `rgb2gray(klblur)`, όπου `klblur` είναι η έγχρωμη εικόνα σε μορφή πίνακα που θέλουμε να μετατρέψουμε. Το αποτέλεσμα φαίνεται παρακάτω.



Εικόνα 28: Εικόνα σε απόχρωση του γκρι

- Αλλαγή της αντίθεσης της εικόνας έτσι ώστε να ξεχωρίζει το ύφασμα, χρησιμοποιώντας την συνάρτηση της MATLAB `imadjust(kgray)` με όρισμα την εικόνα απόχρωσης γκρι `kgray`. Το αποτέλεσμα φαίνεται παρακάτω.



Εικόνα 29: Εικόνα με αυξημένη αντίθεση μέσω της *imadjust*

- Μετατροπή σε δυαδική εικόνα χρησιμοποιώντας τη συνάρτηση της MATLAB *imbinarize(kcontrast, klevel1)*, με ορίσματα την εικόνα γκρι απόχρωσης σε μορφή πίνακα *kcontrast*, και μια τιμή κατωφλιού *klevel1*. Η τιμή αυτή εντοπίζεται χρησιμοποιώντας την συνάρτηση της MATLAB *graythresh(kcontrast)* η οποία στηρίζεται στην μέθοδο του Otsu, εισάγοντας ως όρισμα την εικόνα αποχρώσεως γκρι μετά την αλλαγή της αντίθεσης. Το αποτέλεσμα φαίνεται παρακάτω.



Εικόνα 30: Δυαδική εικόνα χρησιμοποιώντας την *imbinarize* και *Otsu's threshold*

- Αντιστροφή εικονοστοιχείων προσκηνίου εικόνας σε εικονοστοιχεία παρασκηνίου και το αντίστροφο της δυαδικής εικόνας μετατρέποντας τις τιμές των εικονοστοιχείων σε λογικές τιμές, και αφαιρώντας από το 1 την τιμή του κάθε εικονοστοιχείου. Έτσι έχουμε $kwb = \text{logical}(1 - kbw)$, όπου *kwb* ο πίνακας της αρχικής δυαδικής εικόνας και *kwb* ο πίνακας της νέας δυαδική εικόνας με αντίθετη

τιμή εικονοστοιχείων . Η μετατροπή αυτή γίνεται επειδή το ύφασμα βρίσκεται στο παρασκήνιο της εικόνας, και για να μπορέσουμε να εφαρμόσουμε τις μορφολογικές λειτουργίες πρέπει να είναι στο προσκήνιο. Το αποτέλεσμα της μετατροπής αυτής φαίνεται παρακάτω.



Εικόνα 31: Αλλαγή Παρασκηνίου/ Προσκηνίου

- Πλήρωση κενών εκτελώντας την συνάρτηση της MATLAB *imclose(kwb, kse)*, όπου *kwb* η δυαδική εικόνα που δημιουργήθηκε από την παραπάνω επεξεργασία και *kse* το στοιχείο δόμησης το οποίο είναι το ίδιο που χρησιμοποιήθηκε και στην προεπεξεργασία για τον εντοπισμό του μετώπου της ροής της ρητίνης. Το αποτέλεσμα φαίνεται παρακάτω.



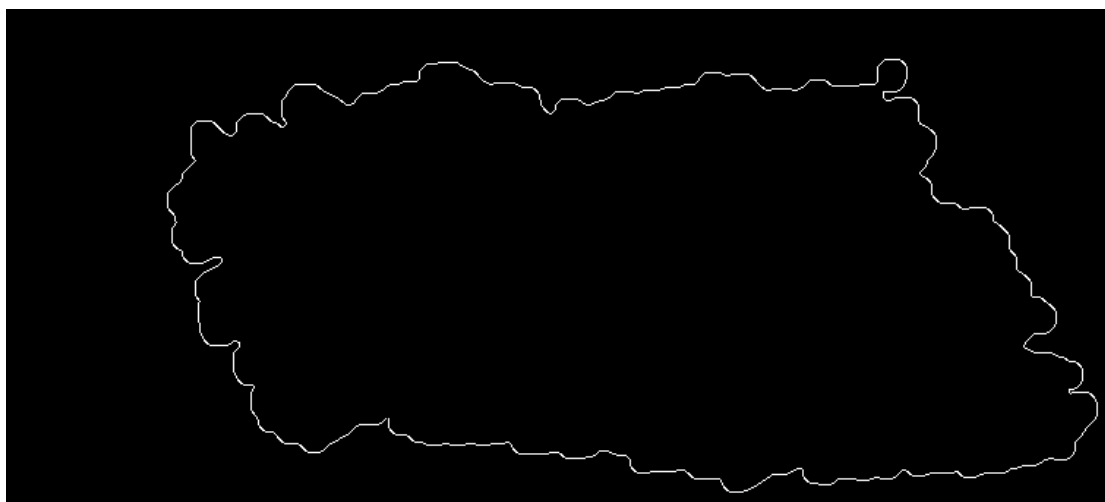
Εικόνα 32 : Εικόνα μετά την εφαρμογή του αλγορίθμου *imclose*

- Εντοπισμός των ορίων του υφάσματος με τον αλγόριθμο ανίχνευσης ακμής του Sobel, εκτελώντας την συνάρτηση της MATLAB *edge(kclose,'Sobel')*, όπου *kclose* η εικόνα σε μορφή πίνακα που δημιουργήθηκε από την εκτέλεση της *imclose*. Το αποτέλεσμα φαίνεται παρακάτω.



Εικόνα 33: Ανίχνευση ορίων υφάσματος προς εμποτισμό με χρήση του αλγορίθμου του Sobel

- Αφαίρεση μικρών οπών που τυχόν έχουν δημιουργηθεί, έτσι ώστε να μην δημιουργηθεί πρόβλημα στον εντοπισμό του υφάσματος ως μοναδικό αντικείμενο κατά τη χρήση της `bwconncomp` και `bwlabel`. Η αφαίρεση των μικρών οπών πραγματοποιείται με τη συνάρτηση της MATLAB `bwareaopen(kbbww,120)`, όπου `kbbww` η εικόνα σε μορφή πίνακα που έχει δημιουργηθεί από την εκτέλεση της συνάρτησης `edge` και 120, το κάτω όριο πλήθους εικονοστοιχείων που ορίζουν μια αποδεκτή ακμή. Το αποτέλεσμα φαίνεται παρακάτω.



Εικόνα 34: Εικόνα μετά τη χρήση της μορφολογικής λειτουργίας ανοίγματος `bwareaopen`

4.4.4 Εύρεση Συντεταγμένων Εικονοστοιχείων του Υφάσματος

Με τη χρήση της συνάρτησης `bwconncomp` της MATLAB, βρίσκουμε τα στοιχεία που είναι συνδεδεμένα μεταξύ τους. Εν προκειμένω, βρίσκουμε ως αντικείμενο αυτό του υφάσματος προς εμποτισμό. Τα ορίσματα είναι η δυαδική εικόνα υπό μορφή πίνακα `kedgeopen`, και το

είδος της συνδεσιμότητας, στην περίπτωσή μας 8 στοιχείων. Η έξοδος της `bwconncomp` όπως έχουμε αναφέρει, είναι τα συνδεδεμένα αντικείμενα ως πίνακας δομής με τέσσερα πεδία: το πλήθος των αντικειμένων, `connectivity`, `NumObjects` και `PixelIdxList`.

Για την εξαγωγή των συντεταγμένων των μετώπων ροής της ρητίνης χρησιμοποιούμε την εντολή της MATLAB `regionprops (kcc, properties)`, όπου `kcc` είναι ο πίνακας δομής που έχει δημιουργηθεί από την `bwconncomp`, και στο `properties` ορίζουμε το είδος της πληροφορίας 'pixelist', αφού θέλουμε τις συντεταγμένες των εικονοστοιχείων. Το αποτέλεσμα της `regionprops (kcc, 'pixelist')`, είναι ένας πίνακας δομής `stats2` με ένα πεδίο το οποίο αποτελείται από έναν πίνακα $k \times 2$, όπου k το πλήθος των εικονοστοιχείων που ανήκει στο μέτωπο της ροής. Στην πρώτη στήλη έχουμε τις συντεταγμένες κατά τον άξονα x και στην δεύτερη στήλη τις συντεταγμένες κατά τον άξονα y , για κάθε εικονοστοιχείο. Να σημειωθεί ότι έχουμε βρει τις συντεταγμένες των εικονοστοιχείων των ορίων του υφάσματος προς εμποτισμό.

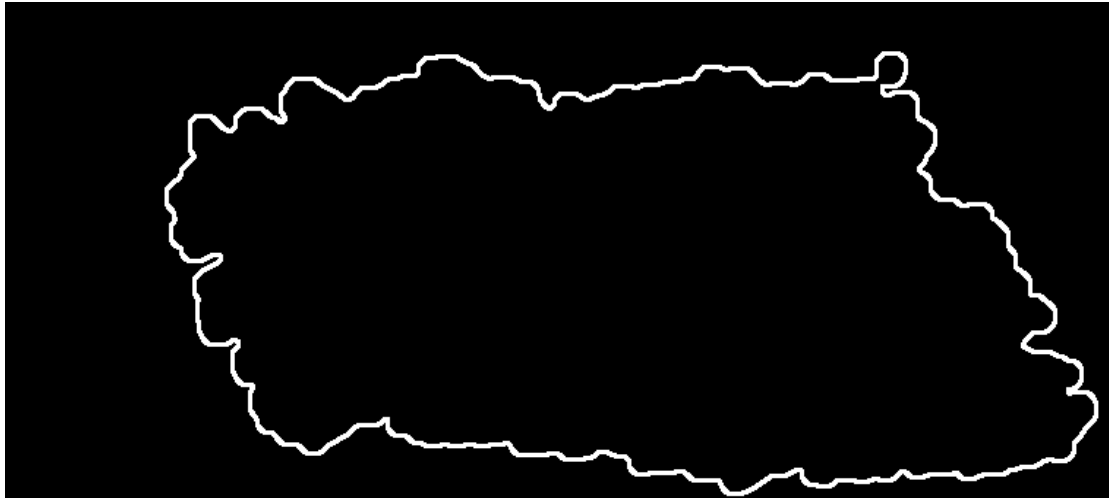
4.4.5 Αλγόριθμος Γεμίματος Περιοχής στο Ύφασμα προς Εμποτισμό

Χρησιμοποιούμε τον αλγόριθμο γεμίματος περιοχής στη δυαδική εικόνα με το περίγραμμα του υφάσματος που έχει βρεθεί, έτσι ώστε να έχουμε ολόκληρη την επιφάνειά του. Η γνώση της επιφάνειας του υφάσματος σε συνδυασμό με την γνώση του μετώπου της ροής θα μας χρησιμεύσει για τον έλεγχο της ροής της ρητίνης.

Εκτελώντας την συνάρτηση της MATLAB `imfill(mask, 'holes')`, τα εικονοστοιχεία τα οποία βρίσκονται μέσα σε ένα κλειστό όριο και είναι στο παρασκήνιο, με αλλαγή της τιμής τους από 0 σε 1, μεταφέρονται στο προσκήνιο. Με αυτόν τον τρόπο έχουμε ολόκληρη την επιφάνεια του υφάσματος προς εμποτισμό. Τα ορίσματα είναι η δυαδική εικόνα `mask`, που εμπεριέχει το περίγραμμα του υφάσματος καθώς και η ιδιότητα 'holes', έτσι ώστε οποιοδήποτε κενό υπάρχει να γεμίζει.

Πριν εκτελέσουμε τον αλγόριθμο γεμίματος περιοχής, λόγω του ότι υπάρχει περίπτωση το περίγραμμα του υφάσματος που βρέθηκε να μην είναι κλειστό σε κάποιο σημείο και να δημιουργηθεί πρόβλημα κατά την εκτέλεση της `imfill`, πρέπει να κάνουμε πιο συμπαγές το περίγραμμα του υφάσματος, δηλαδή να ενισχύσουμε του δεσμούς γειτονιάς μεταξύ των εικονοστοιχείων. Χρησιμοποιώντας την συνάρτηση της MATLAB `boundarymask(kedgeopen)`, ενισχύονται τα εικονοστοιχεία των ορίων του περιγράμματος του υφάσματος. Ως όρισμα εισάγεται η δυαδική εικόνα `kedgeopen`. Η εικόνα δυαδική εικόνα που δημιουργείται είναι ένας λογικός πίνακας. Εναλλακτικά μπορούμε να χρησιμοποιήσουμε την μορφολογική λειτουργία της διαστολής, ορίζοντας ένα στοιχείο δόμησης. Η `boundarymask` υπολογίζει

αυτόματα μια μάσκα ως στοιχείο δόμησης. Τα αποτελέσματα της *boundarymask* και της *imfill* για συγκεκριμένο σχήμα υφάσματος προς εμποτισμό φαίνονται παρακάτω.



Εικόνα 35: Ενίσχυση περιγράμματος του υφάσματος με χρήση της *boundarymask*



Εικόνα 36: Επιφάνεια υφάσματος με χρήση της *imfill*

4.4.6 Αλλαγή Τιμής Εικονοστοιχείων Υφάσματος προς Εμποτισμό

Η δυαδική εικόνα που έχει δημιουργηθεί από την εκτέλεση της *imfill* είναι ο πίνακας *kal*, όπου τα εικονοστοιχεία έχουν τιμή ίση με 0 ή 1. Θα μετατρέψουμε τα εικονοστοιχεία που έχουν τιμή ίση με 1 σε εικονοστοιχεία τιμής ίσης με 3. Η αλλαγή της τιμής εφόσον γίνει σε όλα τα εικονοστοιχεία δεν αλλάζει το περιεχόμενο και το πλήθος των αντικειμένων της εικόνας. Κάνουμε αυτή την αλλαγή στην τιμή για να ξεχωρίζει το αντικείμενο του υφάσματος από τα δύο αντικείμενα των μετώπων ροής, καθώς από την *bwlabel* έχει δημιουργηθεί ένας πίνακας όπου τα εικονοστοιχεία του μετώπου της πρώτης ροής έχουν τιμή ίση με 1 και τα εικονοστοιχεία του μετώπου της δεύτερης ροής έχουν τιμή ίση με 2. Ο πίνακας του υφάσματος και ο πίνακας των μετώπων θα προστεθούν, έτσι ώστε να έχουμε

έναν πίνακα με το ύφασμα και τα μέτωπα ροής της ρητίνης, τα οποία θα ξεχωρίζουν λόγω της διαφορετικής τιμής των εικονοστοιχείων. Αυτή η λειτουργία θα μας χρησιμεύσει παρακάτω.

Η μετατροπή αυτή γίνεται χρησιμοποιώντας την εντολή `kal(kal==1)=3`, αφού πρώτα έχουμε αλλάξει τον τύπο της μεταβλητής `kal` από λογικό σε διπλής ακρίβειας, καθώς το αποτέλεσμα της συνάρτησης `boundarymask` που χρησιμοποιήσαμε είναι λογικού τύπου. Η μετατροπή αυτή γίνεται χρησιμοποιώντας την εντολή `kal=double(kal)`.

4.4.7 Δημιουργία Σημείων Ελέγχου για τον Έλεγχο Ροής της Ρητίνης

4.4.7.1 Εισαγωγή

Για τον έλεγχο των δύο ροών της ρητίνης, πρέπει να γνωρίζουμε αν η ρητίνη έχει εμποτίσει το ύφασμα και ποια περιοχή είναι αυτή. Για να μην ελέγχονται κάθε φορά όλα τα εικονοστοιχεία του υφάσματος με αποτέλεσμα το υπολογιστικό κόστος να είναι μεγάλο, θα ορίσουμε συγκεκριμένα σημεία τα οποία θα ελέγχονται κατά τη διάρκεια της κατεργασίας.

Τις δύο περιοχές διάδοσης της ρητίνης θα τις βρούμε υπολογίζοντας το κέντρο μάζας της περιοχής του υφάσματος. Εφόσον οι γνωστές οι συντεταγμένες του κέντρου μάζας, η επιφάνεια του υφάσματος χωρίζεται σε δύο περιοχές με όριο μια νοητή γραμμή, παράλληλη στον άξονα y , η οποία περνάει από το κέντρο μάζας.

4.4.7.2 Υπολογισμός Κέντρου Μάζας

Για να υπολογιστεί το κέντρο μάζας χρησιμοποιούμε την εντολή της MATLAB `regionprops(klabeledfill,'centroid')`, όπου `klabeledfill` είναι ο πίνακας που εμπεριέχει τα εικονοστοιχεία του υφάσματος και `'centroid'` είναι η ιδιότητα που ορίζουμε στην `regionprops` για να βρούμε το κέντρο μάζας. Το αποτέλεσμα της `regionprops(klabeledfill,'centroid')` είναι ένα διάνυσμα δύο στοιχείων, των συντεταγμένων x - y του κέντρου μάζας στον άξονα.

4.4.7.3 Διαίρεση Επιφάνειας Υφάσματος

Θα διαιρέσουμε την επιφάνεια του υφάσματος σε τέσσερα τμήματα κατά την οριζόντια κατεύθυνση και σε άλλα τέσσερα τμήματα κατά την κατακόρυφη κατεύθυνση, επομένως η επιφάνεια του υφάσματος θα αποτελείται από συνολικά δεκαέξι τμήματα. Η κάθε βαλβίδα είναι υπεύθυνη από οχτώ τμήματα.

Οριζόντια Διεύθυνση

Αρχικά υπολογίζουμε τις συντεταγμένες των σημείων που θα διαιρεθεί η επιφάνεια του

υφάσματος στην οριζόντια διεύθυνση, με τα παρακάτω βήματα:

1. Υπολογίζουμε την συντεταγμένη στον άξονα του x, `firstx3` του εικονοστοιχείου της περιμέτρου του υφάσματος, το οποίο βρίσκεται πιο κοντά στην αρχή των αξόνων, χρησιμοποιώντας την συνάρτηση της MATLAB `min`. Τις συντεταγμένες `x3`, `y3` των εικονοστοιχείων της περιμέτρου του υφάσματος τις γνωρίζουμε.
2. Υπολογίζουμε την συντεταγμένη στον άξονα του x, `lastx3` του εικονοστοιχείου της περιμέτρου του υφάσματος, το οποίο βρίσκεται όσο πιο μακριά από την αρχή των αξόνων, χρησιμοποιώντας την συνάρτηση της MATLAB `max`.
3. Υπολογίζουμε τα διαστήματα στα οποία θα χωριστεί το ύφασμα στην οριζόντια κατεύθυνση. Θα υπολογιστούν δύο συνολικά διαφορετικά διαστήματα. Το πρώτο διάστημα υπολογίζεται ως εξής : `space1x=round((xcm-firstx3)/2)`, όπου `xcm` η συντεταγμένη στον άξονα του x του κέντρου μάζας. Με τη συνάρτηση `round` της MATLAB, στρογγυλοποιούμε την τιμή στον πλησιέστερο ακέραιο, αφού οι συντεταγμένες των εικονοστοιχείων είναι ακέραιοι αριθμοί. Το δεύτερο διάστημα υπολογίζεται ως εξής: `space2x=round((lastx3-xcm)/2)`.

Επομένως τα σημεία των ορίων των τμημάτων που θα χωριστεί το ύφασμα είναι:

`sx1=firstx3`

`sx2=firstx3+space1x`

`sx3=xcm`

`sx4=xcm+space2x`

`sx5=lastx3`

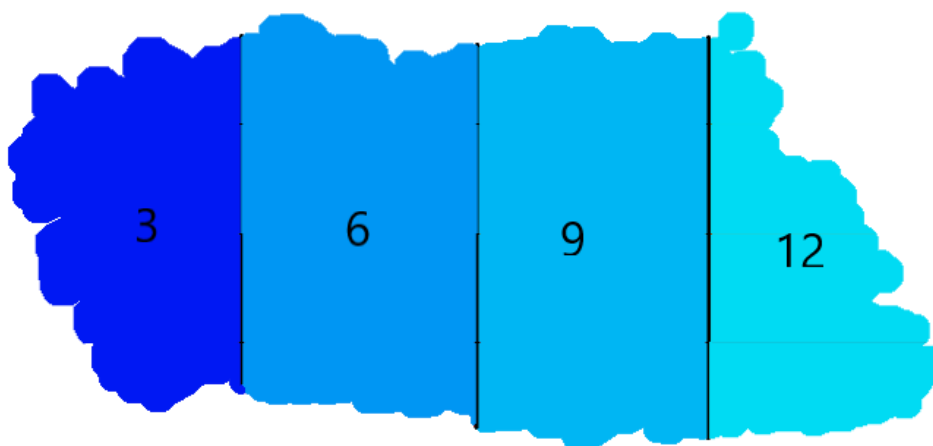
Γνωρίζοντας τις συντεταγμένες των σημείων με τα οποία θα διαιρεθεί η επιφάνεια του υφάσματος σε 4 διαφορετικά τμήματα στην οριζόντια διεύθυνση, θα αλλάξουμε για το κάθε τμήμα τις τιμές των εικονοστοιχείων. Στο πρώτο τμήμα η τιμή των εικονοστοιχείων παραμένει ίση με 3, και τα εικονοστοιχεία των υπόλοιπων τμημάτων αυξάνονται κατά 3, με τελική τιμή ίση με 12, που έχουν τα εικονοστοιχεία του τελευταίου τμήματος.

Αυτή η διαδικασία πραγματοποιείται εξετάζοντας αν η τιμή ενός εικονοστοιχείου έχει τιμή ίση με 3, `kal(xx,x)==3` ώστε να αλλάξουν μόνο τα εικονοστοιχεία του υφάσματος. Αν ικανοποιείται η συνθήκη αυτή, αναλόγως σε ποιο διάστημα βρισκόμαστε, αποδίδεται η αντίστοιχη τιμή. Για την εξέταση της παραπάνω συνθήκης στο κάθε διάστημα του υφάσματος χρησιμοποιούμε επαναληπτική διαδικασία `for`. Ο δείκτης `x` της `for`, είναι η τετμημένη του κάθε εικονοστοιχείου που θα αλλάξει τιμή. Η αρχική τιμή της `for`, είναι οι τετμημένες των αρχικών σημείων `sx2`, `sx3`, `sx4` για το κάθε διάστημα. Η τελική τιμή με την οποία θα τερματίσει η επαναληπτική διαδικασία της `for`, είναι οι τετμημένες των τελικών

σημείων $sx3$, $sx4$, $sx5$ για το κάθε διάστημα. Στο εσωτερικό της επαναληπτικής διαδικασίας `for`, υπάρχει μια δεύτερη επαναληπτική διαδικασία `for`, έτσι ώστε να αλλάξουν τιμή όλα τα εικονοστοιχεία σε όλες τις γραμμές του πίνακα, καθώς μόνο με το ένα `for`, θα αλλάξουν τα εικονοστοιχεία της πρώτης μόνο γραμμής του πίνακα που ικανοποιείται η συνθήκη. Ο δείκτης `xx` της εμβόλιμης επαναληπτικής διαδικασίας `for`, είναι η τεταγμένη του κάθε εικονοστοιχείου που θα αλλάξει τιμή. Σαν αρχική τιμή έχουμε βάλει την τεταγμένη, του εικονοστοιχείου που βρίσκεται στην αρχή των αξόνων δηλαδή ίση με 1. Η τελική τιμή με τη οποία θα τερματίσει η επαναληπτική διαδικασία `for` είναι η τιμή `lasty3`.

Να σημειωθεί ότι στο σύστημα συντεταγμένων με βάση τα εικονοστοιχεία, η τεταγμένη είναι ο δείκτης της σειράς του εικονοστοιχείου στον πίνακα. Αντίστοιχα, η τετμημένη είναι ο δείκτης της στήλης του εικονοστοιχείου στον πίνακα. Επιπλέον η αρχή των αξόνων είναι το σημείο (1,1).

Παρακάτω φαίνεται ο διαχωρισμός του υφάσματος στην οριζόντια κατεύθυνση, και οι τιμές που έχουν τα εικονοστοιχεία στο κάθε διάστημα.



Εικόνα 37: Διαίρεση υφάσματος στην οριζόντια διεύθυνση. 3, 6, 9, 12, οι τιμές των εικονοστοιχείων στο κάθε διάστημα

Κατακόρυφη Διεύθυνση

Για την διαίρεση της επιφάνειας του υφάσματος στην κατακόρυφη διεύθυνση θα υπολογίσουμε:

1. την τεταγμένη, `firsty3` του εικονοστοιχείου της περιμέτρου του υφάσματος, το οποίο βρίσκεται πιο κοντά στην αρχή των αξόνων, χρησιμοποιώντας την συνάρτηση της MATLAB `min`.
2. την τεταγμένη, `lasty3` του εικονοστοιχείου της περιμέτρου του υφάσματος, το οποίο βρίσκεται όσο πιο μακριά από την αρχή των αξόνων, χρησιμοποιώντας την συνάρτηση

της MATLAB *max*.

3. Τα διαστήματα με τα οποία θα χωριστεί το ύφασμα στην κατακόρυφη διεύθυνση. Θα υπολογιστούν δύο διαφορετικά διαστήματα. Το πρώτο υπολογίζεται ως εξής : $space1y=round((ycm-firsty3)/2)$, όπου *ycm* η τεταγμένη του κέντρου μάζας. Το δεύτερο διάστημα υπολογίζεται ως εξής: $space2y=round((lasty3-ycm)/2)$.

Τα σημεία των ορίων των τμημάτων που θα χωριστεί το ύφασμα είναι:

$sy1=firsty3$

$sy2=firsty3+space1y$

$sy3=ycm$

$sy4=ycm+space2y$

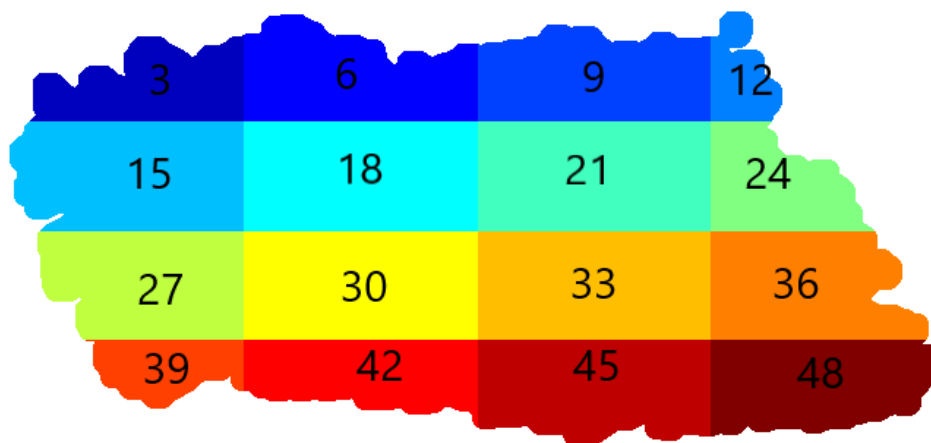
$sy5=lasty3$

Γνωρίζοντας τις συντεταγμένες των σημείων με τα οποία θα διαιρεθεί η επιφάνεια του υφάσματος σε 4 διαφορετικά τμήματα στην κατακόρυφη διεύθυνση, θα αλλάξουμε τις τιμές των εικονοστοιχείων του κάθε τμήματος. Έχει προηγηθεί αλλαγή τιμών των εικονοστοιχείων στην οριζόντια διεύθυνση. Η διαδικασία είναι η εξής: Για κάθε διάστημα της επιφάνειας του υφάσματος οι τιμές των εικονοστοιχείων αλλάζουν με αριθμητική πρόοδο με βήμα ίσο με 12. Επομένως η τιμή των εικονοστοιχείων του δεύτερου διαστήματος θα αυξηθεί κατά 12, του τρίτου κατά 24 και του τέταρτου διαστήματος κατά 36. Οι τιμές των εικονοστοιχείων του πρώτου διαστήματος δεν χρειάζονται αλλαγή.

Για την αλλαγή της τιμής των εικονοστοιχείων, η τιμή ενός εικονοστοιχείου πρέπει να είναι ίση με 3 ή 6 ή 9 ή 12 $kal(x,xx)==3 || kal(x,xx)==6 || kal(x,xx)==9 || kal(x,xx)==12$. Όπως και προηγουμένως, για την εξέταση της παραπάνω συνθήκης στο κάθε διάστημα του υφάσματος χρησιμοποιούμε επαναληπτική διαδικασία *for*. Ο δείκτης *x* της *for*, είναι η τεταγμένη του κάθε εικονοστοιχείου που θα αλλάξει η τιμή. Η αρχική τιμή της *for*, είναι οι τεταγμένες των αρχικών σημείων *sy2*, *sy3*, *sy4* για το κάθε διάστημα. Η τελική τιμή με την οποία θα τερματίσει η επαναληπτική διαδικασία *for*, είναι οι τεταγμένες των τελικών σημείων *sy3*, *sy4*, *sy5* για το κάθε διάστημα. Στο εσωτερικό της επαναληπτικής διαδικασίας *for*, υπάρχει μια δεύτερη επαναληπτική διαδικασία *for*, έτσι ώστε να αλλάξουν τιμή όλα τα εικονοστοιχεία σε όλες τις στήλες του πίνακα. Ο δείκτης *xx* της εμβόλιμης επαναληπτικής διαδικασίας *for*, είναι η τετμημένη του κάθε εικονοστοιχείου που θα αλλάξει τιμή. Σαν αρχική τιμή έχουμε βάλει την τετμημένη *x* του εικονοστοιχείου που βρίσκεται στην αρχή των αξόνων δηλαδή ίση με 1. Η τελική τιμή με την οποία θα τερματίσει η επαναληπτική

διαδικασία for είναι η τιμή lastx3.

Παρακάτω φαίνεται η διαίρεση του υφάσματος στα 16 ξεχωριστά τμήματα και οι τιμές των εικονοστοιχείων του κάθε τμήματος.



Εικόνα 38: Διαίρεση του υφάσματος σε 16 ξεχωριστά τμήματα. Οι αριθμοί είναι η τιμή των εικονοστοιχείων.

4.4.7.4 Υπολογισμός Κέντρου Μάζας για Κάθε Τμήμα

Οι τιμές των εικονοστοιχείων του κάθε τμήματος είναι διαφορετικές, με τα εικονοστοιχεία του ίδιου τμήματος να έχουν την ίδια τιμή. Αυτή η ιδιαιτερότητα μας βοηθάει ώστε η MATLAB να θεωρήσει το κάθε τμήμα ως ένα ξεχωριστό αντικείμενο. Χρησιμοποιώντας την συνάρτηση της MATLAB `regionprops(klabelfilldiv,'centroid')` την οποία έχουμε χρησιμοποιήσει και προηγουμένως, εισάγοντας ως ορίσματα: τον πίνακα `klabelfilldiv` με τις καινούριες τιμές των εικονοστοιχείων και την ιδιότητα 'centroid', βρίσκουμε τα κέντρα μάζας για κάθε τμήμα του υφάσματος. Το αποτέλεσμα της `regionprops` είναι ο πίνακας δομής `cmdiv`, με ένα διάνυσμα στήλης 1×48 . Το κάθε κελί περιέχει την τεταγμένη την τεταγμένη του κέντρου μάζας κάθε τμήματος. Να σημειωθεί πως τα κελιά τα οποία δεν είναι πολλαπλάσια του 3 είναι κενά.

4.4.7.5 Σημεία Ελέγχου

Σημεία Ελέγχου Βαλβίδας A

Τα σημεία ελέγχου για τη βαλβίδα A είναι συνολικά 9. Τα πρώτα 8 είναι τα κέντρα μάζας των τμημάτων που βρίσκονται στη πρώτη ημπεριοχή του υφάσματος. Δηλαδή τα τμήματα με τιμές εικονοστοιχείων: 3, 6, 15, 18, 27, 30, 39, 42.

Επομένως έχουμε:

```
chpointA1=round(cmdiv(3).Centroid)
```



```
chpointA2=round(cmdiv(6).Centroid)
chpointA3=round(cmdiv(15).Centroid)
chpointA4= round(cmdiv(18).Centroid)
chpointA5=round(cmdiv(27).Centroid)
chpointA6=round(cmdiv(30).Centroid)
chpointA7=round(cmdiv(39).Centroid)
chpointA8=round(cmdiv(42).Centroid)
```

Το ένατο σημείο chpointA9 είναι το ακραίο σημείο στη κατακόρυφη διεύθυνση της πρώτης ημιπεριοχής του υφάσματος halfA. Για να υπολογίσουμε το σημείο αυτό εκτελούνται τα παρακάτω βήματα:

- Υπολογίζεται το halfA εκτελώντας την εντολή: `klabelfill(:,1:xcm)`, όπου xcm η τετμημένη του κέντρου μάζας cm ολόκληρης της περιοχής του υφάσματος.
- Βρίσκονται οι συντεταγμένες των εικονοστοιχείων της halfA, εκτελώντας την εντολή: `find(halfA==3)`. Το αποτέλεσμα είναι δύο διανύσματα στήλης row, col, τα οποία είναι οι τεταγμένες και τετμημένες αντίστοιχα.
- Εκτελώντας την συνάρτηση της MATLAB `max(row)`, παίρνουμε ως αποτέλεσμα δύο μεταβλητές τις value και ind. Η value έχει την μέγιστη τιμή της τεταγμένης των εικονοστοιχείων του υφάσματος. Η μεταβλητή ind είναι ο δείκτης της θέσης του διανύσματος στήλης row με την μέγιστη τιμή. Η ind θα χρησιμοποιηθεί για να βρεθεί η τετμημένη του σημείου που έχει τεταγμένη την value, εκτελώντας την εντολή: `col(ind)`.

Επομένως το ένατο σημείο ελέγχου της βαλβίδας A είναι: `chpointA9=[col(ind),value]`

Σημεία Ελέγχου Βαλβίδας B

Παρόμοια με τα σημεία ελέγχου της Βαλβίδας A, αλλά για τα τμήματα με τιμές εικονοστοιχείων 9, 12, 21, 24, 33, 36, 45, 48 έχουμε:

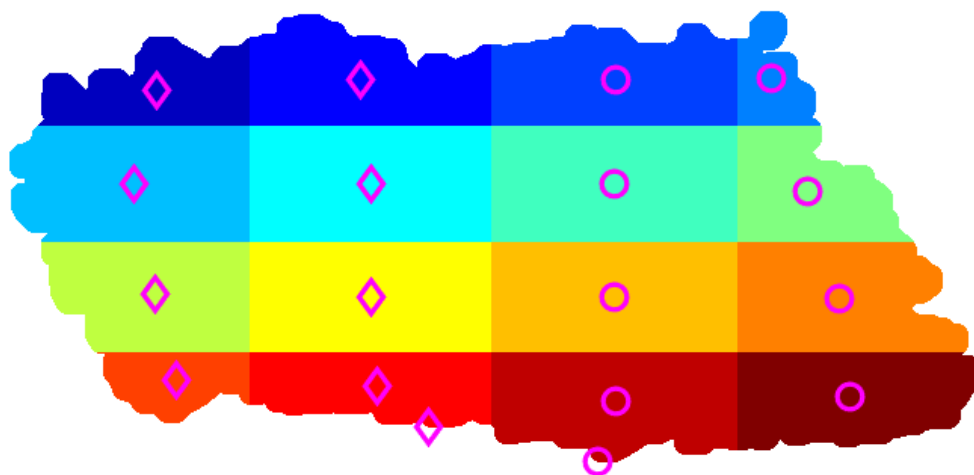
```
chpointB1=round(cmdiv(9).Centroid)
chpointB2=round(cmdiv(12).Centroid)
chpointB3=round(cmdiv(21).Centroid)
chpointB4= round(cmdiv(24).Centroid)
chpointB5=round(cmdiv(33).Centroid)
chpointB6=round(cmdiv(36).Centroid)
chpointB7=round(cmdiv(45).Centroid)
chpointB8=round(cmdiv(48).Centroid)
```

Για το ένατο σημείο `chpointB9` το οποίο είναι το ακραίο σημείο στη κατακόρυφη διεύθυνση της δεύτερης ημπεριοχής του υφάσματος `halfB` εκτελούμε τα παρακάτω βήματα:

- Υπολογίζεται το `halfB` εκτελώντας την εντολή: `klabelfill(:,xcm+1:size(klabelfill,2))`, όπου `xcm` η τεταγμένη του κέντρου μάζας `cm` ολόκληρης της περιοχής του υφάσματος. Με την συνάρτηση της MATLAB `size`, βρίσκουμε την δεύτερη διάσταση του πίνακα `klabelfill` που είναι το πλήθος των στηλών της `klabelfill`, αφού το δεύτερο όρισμα έχει τιμή 2.
- Βρίσκονται οι συντεταγμένες των εικονοστοιχείων της `halfB`, εκτελώντας την εντολή: `find(halfB==3)`. Το αποτέλεσμα είναι δύο διανύσματα στήλης `row`, `col`, τα οποία είναι οι τεταγμένες και τετμημένες αντίστοιχα.
- Παρόμοια όπως για το σημείο ελέγχου της βαλβίδας `A`, εκτελώντας την συνάρτηση της MATLAB `max(row)`, παίρνουμε ως αποτέλεσμα δύο μεταβλητές τις `value` και `ind`. Η `value` έχει την μέγιστη τιμή της τεταγμένης των εικονοστοιχείων του υφάσματος. Η μεταβλητή `ind` είναι ο δείκτης της θέσης του διανύσματος στήλης `row` με την μέγιστη τιμή. Η `ind` θα χρησιμοποιηθεί για να βρεθεί η τετμημένη του σημείου που έχει τεταγμένη την `value`, εκτελώντας την εντολή: `col(ind)`.

Επομένως το ένατο σημείο ελέγχου της βαλβίδας `B` είναι: `chpointB9=[col(ind)+xcm,value]`

Τα σημεία ελέγχου της βαλβίδας `A` και της βαλβίδας `B` φαίνονται στην παρακάτω εικόνα, όπου με ρόμβο καταδεικνύονται τα σημεία ελέγχου της βαλβίδας `A` και με κύκλο τα σημεία ελέγχου της βαλβίδας `B`.



Εικόνα 39: Διαίρεση του υφάσματος σε 18 ξεχωριστά τμήματα. Οι αριθμοί είναι η τιμή των εικονοστοιχείων.

Η εικόνα με τα σημεία ελέγχου θα εμφανίζεται στην αρχή της διαδικασίας VARI, ώστε να σιγουρευτούμε ότι η εύρεση των σημείων ελέγχου είναι επιτυχής.

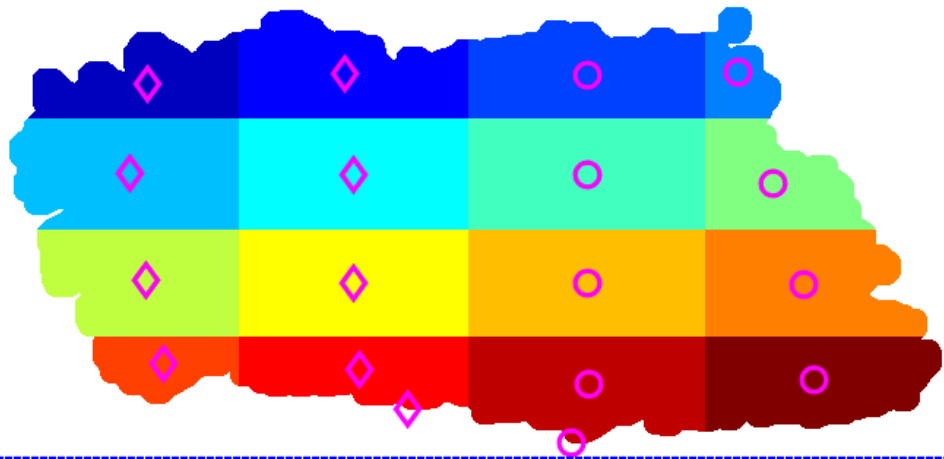
Για να εμφανιστεί η κάθε περιοχή των εικονοστοιχείων του υφάσματος που έχουν διαφορετική τιμή με διαφορετικό χρώμα, χρησιμοποιούμε την συνάρτηση της MATLAB `label2rgb(kal)`, όπου ως όρισμα εισάγουμε την μεταβλητή `kal` που είναι ο πίνακας της εικόνας του διαιρεμένου υφάσματος. Η `label2rgb` μετατρέπει έναν πίνακα με τα αντικείμενα που έχουν εντοπιστεί, σε μια έγχρωμη εικόνα, όπου το κάθε αντικείμενο έχει διαφορετικό χρώμα.

Για το σχεδιασμό των σημείων ελέγχου στην εικόνα που έχει δημιουργηθεί με την `label2rgb`, χρησιμοποιούμε την εντολή της MATLAB `plot(X,Y,LineSpec)`, όπου `X,Y` οι συντεταγμένες του σημείου που θέλουμε να σχεδιάσουμε και `LineSpec` τα χαρακτηριστικά του σημείου, όπως χρώμα, μέγεθος, σχήμα.

4.4.7.6 Υπολογισμός Ορίου $MaXy$ Για Τον Έλεγχο Της Ρητίνης

Εκτός από τα σημεία ελέγχου του υφάσματος προς εμποτισμό που έχουν υπολογιστεί για τον έλεγχο της ροής της ρητίνης, υπολογίζεται και η οριζόντια ευθεία, $y = MaXy$ η οποία χρησιμοποιείται σαν όριο. Η τιμή του $MaXy$ είναι η τεταγμένη του υφάσματος με την μεγαλύτερη τιμή `lasty3`, αυξημένη κατά τα $2/3$ της κατακόρυφης απόστασης μεταξύ του σημείου της περιοχής ενδιαφέροντος και της μεγαλύτερης τεταγμένης και της `lasty3`. Η περιοχή ενδιαφέροντος έχει ορισθεί στο υποπρόγραμμα `vkalourι.m`. Στο MATLAB εκτελείται η εντολή: `MaXy=round(lasty3+((size(kedgeopen,1)-lasty3)*2/3))`.

Για τον σχεδιασμό της ευθείας, ορίζουμε τις συντεταγμένες ενός αρχικού και ενός τελικού σημείου της ευθείας. Το αρχικό σημείο έχει τετμημένη ίση με 0 και τεταγμένη ίση με $MaXy$. Το τελικό σημείο της ευθείας έχει τετμημένη ίση με τη διάσταση της περιοχής ενδιαφέροντος στην οριζόντια διεύθυνση και τεταγμένη y ίση με $MaXy$. Επομένως, στη MATLAB δημιουργούνται δύο πίνακες γραμμής δύο στοιχείων ο καθένας, με τον πρώτο να περιέχει τις τετμημένες και ο δεύτερος τις τεταγμένες του αρχικού και τελικού σημείου της ευθείας. Στην παρακάτω εικόνα φαίνεται το όριο που έχει δημιουργηθεί, μαζί με τα σημεία ελέγχου.



Εικόνα 40: Όριο ΜαΧγ για τον έλεγχο της ροής της ρητίνης

4.4.9 Κώδικας MATLAB

Ο κώδικας που έχει γραφτεί στη MATLAB για την υλοποίηση των παραπάνω είναι το υπο-πρόγραμμα `vkaloupi.m`. Η μεταβλητή `cam` που εισάγεται το `object` με τα χαρακτηριστικά της κάμερας. Οι μεταβλητές που εξάγονται είναι τα σημεία ελέγχου για την Βαλβίδα A και την βαλβίδα B, και οι μεταβλητές `kedgeopen`, `kcc`, `klabelfill`.

```
% εύρεση καλουπιού
function [kedgeopen,kcc,klabel,klabelfill,...
chpointA1,chpointA2,chpointA3,chpointA4,chpointA5,chpointA6,ch
pointA7,chpointA8,chpointA9,...
chpointB1,chpointB2,chpointB3,chpointB4,chpointB5,chpointB6,ch
pointB7,chpointB8,chpointB9,xmin,ymin,width,height,MaxY]=vkalou
pi(cam)

img = snapshot(cam);
% imshow(img)
kimage=img;

%% Οριοθέτηση περιοχής ενδιαφέροντος
disp('επίλεξε πρώτο σημείο --> κοντά στην αρχή των αξόνων <--
( αρχή αξόνων πάνω αριστερά ) ');
disp('δευτερο σημείο -->μετατοπισμένο ως προς y <-- ');
disp('τρίτο σημείο --> μετατοπισμένο ως προς x <--');
pause(2)
imshow(kimage);
[xinitiall,yinitiall] = ginput(3)
xmin=xinitiall(1)
ymin=yinitiall(1)
width=(xinitiall(3)-xinitiall(1))
height=(yinitiall(2)-yinitiall(1))
close

%% περικοπή αρχικής εικόνας
kimgg=imcrop(kimage,[xmin ymin width height]);
```

```

%% brul
kIblur = imgaussfilt(kimgg,2);

%% μετατροπή σε gray scale
kgray=rgb2gray(kIblur);

% αλλαγή contrast
kcontrast=imadjust(kgray);

%% μετατροπή σε binary
klevell = graythresh(kcontrast);
kbw=imbinarize(kcontrast,klevell);

% μετατροπή pixel παρασκήνιου σε προσκήνιο, bw σε wb
kwb = logical(1 - kbw);

%% γέμισμα κενών
kse = strel('disk',10);
kcloseWB = imclose(kwb,kse);

%ευρεση edge
kbbww=edge(kcloseWB);

%%afairesh mikrwn akmwn bwareaopen
kedgeopen=bwareaopen(kbbww,120);
imshow(kedgeopen);

%% εύρεση object label για το ύφασμα
kcc=bwconncomp(kedgeopen);
klabel=bwlabel(kedgeopen);
klabel(klabel==1)=3; %% αντικαθιστώ όπου 1 το 3 επειδή όταν θα
% το επεξεργαστώ μαζί με τις ροές της
% ρητίνης θέλω ξεχωριστό object.

%% συντεταγμένες object 3
% 1 τρόπος
stats2 = regionprops(kcc, 'pixellist');
pixels3=stats2(1).PixelList;
x3=pixels3(:,1);
y3=pixels3(:,2);
% 2 τρόπος με bwlabel
[row3,col3]=find(klabel==3);

%% fill kaloupi object.
mask = boundarymask(kedgeopen); % υπάρχει σε ένα σημείο κενό,
και προσθέτω Pixel.
%sssee = strel('octagon',3);%%% εναλλακτικά αντί για την
παραπάνω εντολή.
kal=imfill(mask, 'holes');
kal=double(kal); % double καθώς η mask είναι λογική μεταβλητή
kal(kal==1)=3; % object label όπου έχει 1 = 3
klabelfill=kal;
% figure,imshow(kal);%%%%%%%%%%

%% κέντρο μάζας υφάσαμος

```

```

cm=regionprops(klabelfill,'centroid');
xcm=round(cm(3).Centroid(1));
ycm=round(cm(3).Centroid(2));

%% διαίρεση υφάματος οριζόντια
firstx3=min(x3);%x3(1); % εύρεση αρχής υφάματος στον άξονα x
firstx3=firstx3-1;% Παρατήρηση, αφαιρώ 1, γιατί το χ3 είναι
% για το ύφασμα πριν το fill. Η εντολή imfill
%έχει προσθέσει περιμετρικά ένα pixel.
lastx3=max(x3)
lastx3=lastx3+1; % αντίστοιχα εδώ προσθέτω 1, για τον παραπάνω
%
%
space1x=round((xcm-firstx3)/2);% διάστημα που θα χωριστεί το
%
%
space2x=round((lastx3-xcm)/2);
sx1=firstx3
sx2=firstx3+space1x
sx3=xcm
sx4=xcm+space2x
sx5=lastx3

%% διαίρεση υφάματος κάθετα
firsty3=min(y3);%x3(1); % εύρεση αρχής υφάματος στον άξονα x
firsty3=firsty3-1;
lasty3=max(y3) ;
lasty3=lasty3+1;
space1y=round((ycm-firsty3)/2);% διάστημα που θα χωριστεί το
%
%
space2y=round((lasty3-ycm)/2);
sy1=firsty3
sy2=firsty3+space1y
sy3=ycm
sy4=ycm+space2y
sy5=lasty3

%% κάθε διάστημα ξεχωριστή τιμή pixel -->x
for x= sx2:sx3 % δεύτερο διάστημα τα pixel έχουν τιμή 6
    for xx=1:lasty3
        if kal(xx,x)==3
            kal(xx,x)=6;
        end
    end
end

for x=sx3:sx4 % τρίτο διάστημα τα pixel έχουν τιμη 9
    for xx=1:lasty3
        if kal(xx,x)==3
            kal(xx,x)=9;
        end
    end
end

for x=sx4:lastx3 % τέταρτο διάστημα τα pixel έχουν τιμή 12
    for xx=1:lasty3
        if kal(xx,x)==3

```

```

        kal(xx,x)=12;
    end
end
end
% klabelfill=kal;
%% --> y
for x= sy2:sy3    % δεύτερο διάστημα τα pixel έχουν τιμή
                  % υπάρχουσα +12 κατά χ , η πρώτη σειρά έχει
                  % τιμές από προηγούμενο.
    for xx=1:lastx3
        if
            kal(x,xx)==3
|| kal(x,xx)==6|| kal(x,xx)==9|| kal(x,xx)==12
            kal(x,xx)=kal(x,xx)+12;
        end
    end
end
end

for x= sy3:sy4    % τρίτο διάστημα τα pixel έχουν τιμή +24.
    for xx=1:lastx3
        if
            kal(x,xx)==3
|| kal(x,xx)==6|| kal(x,xx)==9|| kal(x,xx)==12
            kal(x,xx)=kal(x,xx)+24;
        end
    end
end
for x= sy4:lasty3 %τέταρτο διάστημα τα pixel έχουν τιμή +36.
    for xx=1:lastx3
        if
            kal(x,xx)==3
|| kal(x,xx)==6|| kal(x,xx)==9|| kal(x,xx)==12
            kal(x,xx)=kal(x,xx)+36;
        end
    end
end
end

%% κέντρα μάζας επιμέρους διαστημάτων, θα χρησιμοποιηθούν ως
% σημεία ελέγχου
klabelfilldiv=kal;
cmdiv=regionprops(klabelfilldiv,'centroid');
% klabelfill=kal;

%% σημεία ελέγχου για βαλβίδα A (8+1)
chpointA1=round(cmdiv(3).Centroid) ;
chpointA2=round(cmdiv(6).Centroid) ;
chpointA3=round(cmdiv(15).Centroid) ;
chpointA4=round(cmdiv(18).Centroid) ;
chpointA5=round(cmdiv(27).Centroid) ;
chpointA6=round(cmdiv(30).Centroid) ;
chpointA7=round(cmdiv(39).Centroid) ;
chpointA8=round(cmdiv(42).Centroid) ;
% --> ακραίο σημείο <--
halfA=klabelfill(:,1:xcm); % από αρχή έως κ. μάζας
                             % περιοχή ευθύνης A βαλβίδας

[roW,coL]=find(halfA==3);
[value,ind]=max(roW); % ακραίο σημείο στον y,value

```

```

                                % = y, ind = index of array
chpointA9=[coL(ind),value];    % coL(ind) --> x

%% σημεία ελέγχου για βαλβίδα B (8+1)
chpointB1=round(cmdiv(9).Centroid) ;
chpointB2=round(cmdiv(12).Centroid) ;
chpointB3=round(cmdiv(21).Centroid) ;
chpointB4=round(cmdiv(24).Centroid);
chpointB5=round(cmdiv(33).Centroid) ;
chpointB6=round(cmdiv(36).Centroid);
chpointB7=round(cmdiv(45).Centroid) ;
chpointB8=round(cmdiv(48).Centroid);
% --> ακραίο σημείο <--
colklabelfill=size(klabelfill,2);
halfB=klabelfill(:,xcm+1:size(klabelfill,2)); % από κ. Μάζας
                                             %έως τέλος, περιοχή ευθύνης B
βαλβίδας
[roW,coL]=find(halfB==3);
[value,ind]=max(roW); % ακραίο σημείο
                    στον y , value = y, ind = index of array
chpointB9=[coL(ind)+xcm,value] % coL(ind) --> x, προσθέτω xcm
                    %ώστε το σημείο να είναι σημείο σύμφωνα με ολόκληρο το
                    % ύφασμα
% Υπολογισμός οριζόντιας γραμμής για τον έλεγχο της ροής της
% ρητίνης από το Arduino
MaXy=round(lasty3+((size(kedgeopen,1)-lasty3)*2/3));

% Σημεία για plot ευθείας y=MaXy
limitx=[0 size(kimgg,2)];
limity=[MaXy MaXy];

%% plot σημείων

label2rgb(kal);
imwrite(kal,'peiramakaloupi1.jpg');
figure,imshow(label2rgb(kal));
hold on;
plot(chpointA1(1),chpointA1(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA2(1),chpointA2(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA3(1),chpointA3(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA4(1),chpointA4(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA5(1),chpointA5(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA6(1),chpointA6(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA7(1),chpointA7(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA8(1),chpointA8(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointA9(1),chpointA9(2), 'md', 'LineWidth', 2,
'MarkerSize', 10);

```



```

plot(chpointB1(1),chpointB1(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB2(1),chpointB2(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB3(1),chpointB3(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB4(1),chpointB4(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB5(1),chpointB5(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB6(1),chpointB6(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB7(1),chpointB7(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB8(1),chpointB8(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(chpointB9(1),chpointB9(2), 'mo', 'LineWidth', 2,
'MarkerSize', 10);
plot(limitx, limity, 'b:', 'LineWidth', 2);

end

```

4.5 Ενσωμάτωση Μετώπου Ροής και Υφάσματος σε μια Εικόνα

4.5.1 Εισαγωγή

Τα μέτωπα ροής της ρητίνης, όπως και το ύφασμα προς εμποτισμό, έχουν ανιχνευτεί ως ξεχωριστά αντικείμενα, σε διαφορετικές εικόνες (κοινή για τα μέτωπα ροής) . Όπως γνωρίζουμε από την ανάλυση που έχει προηγηθεί, τα εικονοστοιχεία των μετώπων της ροής της ρητίνης έχουν τιμή ίση με 1 για το πρώτο μέτωπο και 2 για το δεύτερο μέτωπο ροής, όταν τα δύο μέτωπα ροής γίνονται ένα, τα εικονοστοιχεία έχουν τιμή ίση με 1. Τα εικονοστοιχεία του υφάσματος προς εμποτισμό έχουν τιμή ίση με 3.

Οι δύο ξεχωριστές εικόνες θα γίνουν μια εικόνα η οποία θα περιέχει όλα τα αντικείμενα, δηλαδή τα μέτωπα ροής της ρητίνης και το ύφασμα προς εμποτισμό. Οι εικόνες όπως έχουμε αναφέρει παρίστανται με τη χρήση των πινάκων, όπου τα στοιχεία του πίνακα είναι τα αντίστοιχα εικονοστοιχεία της εικόνας.

4.5.2 Δημιουργία Εικόνας

Εκτελώντας την πράξη της πρόσθεσης μεταξύ των δύο πινάκων οι οποίοι έχουν τις ίδιες διαστάσεις, δημιουργείται ένας πίνακας Label2 με τις ίδιες διαστάσεις που εμπεριέχει τα εικονοστοιχεία των μετώπων ροής της ρητίνης και του υφάσματος. Η εντολή που εκτελούμε στη MATLAB είναι: *Label2=label+klabelfill*, όπου η μεταβλητή label είναι ο πίνακας με τα εντοπισμένα αντικείμενα των μετώπων ροής, που έχει δημιουργηθεί από το υπο-

πρόγραμμα `vgrammh2roes.m` και η μεταβλητή `klabelfill` ο πίνακας με εντοπισμένο αντικείμενο το ύφασμα, που έχει δημιουργηθεί από το υπο-πρόγραμμα `vkaloupi.m`. Στον πίνακα `Label2` τα εικονοστοιχεία του μετώπου της ροής της ρητίνης τα οποία έχουν τις ίδιες συντεταγμένες με εικονοστοιχεία του υφάσματος προς εμπότισμό έχουν αλλάξει τιμή. Τα εικονοστοιχεία του πρώτου μετώπου ροής ή του κοινού μετώπου ροής της ρητίνης, τα οποία είχαν τις ίδιες συντεταγμένες με εικονοστοιχεία του υφάσματος έχουν πλέον τιμή ίση με 4, αντίστοιχα τα εικονοστοιχεία του δεύτερου μετώπου ροής έχουν τιμή ίση με 5.

4.5.3 Μετατροπή Σε Έγχρωμη Εικόνα

Η εικόνα `Label2` θα μετατραπεί σε έγχρωμη ώστε να ξεχωρίζουν τα εικονοστοιχεία των διαφορετικών αντικειμένων, χρησιμοποιώντας τη συνάρτηση της MATLAB `label2rgb(Label2)`, με όρισμα τη μεταβλητή `Label2`. Το αποτέλεσμα είναι μια εικόνα σε μορφή πίνακα, η `Labelcol1` όπου τα εικονοστοιχεία του μετώπου ροής και του υφάσματος παρουσιάζονται με διαφορετικό χρώμα. Να επισημανθεί πως τα εικονοστοιχεία του μετώπου τα οποία βρίσκονται στην περιοχή του υφάσματος έχουν διαφορετικό χρώμα από αυτά τα οποία είναι εκτός περιοχής υφάσματος, καθώς έχουν διαφορετική τιμή και θεωρούνται ξεχωριστά αντικείμενα.

4.5.4 Εμφάνιση Της `Labelcol1` Στην Διάρκεια Της VARI

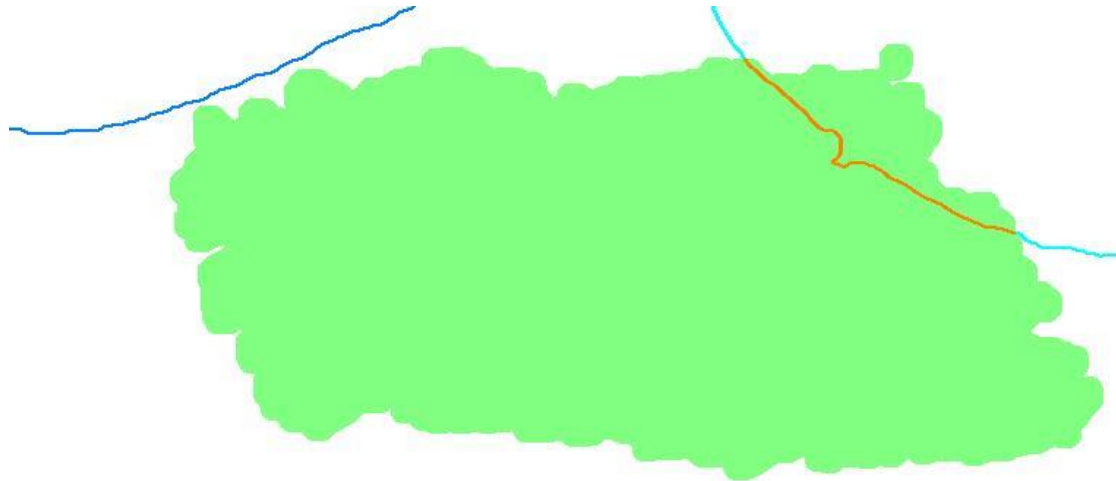
Κατά την διάρκεια της κατεργασίας VARI, η `Labelcol1` θα εμφανίζεται στην οθόνη του υπολογιστή, ώστε να βλέπουμε κάθε φορά το μέτωπο της ροής πάνω στο ύφασμα προς εμπότισμό, χρησιμοποιώντας την εντολή της MATLAB `imshow(Labelcol1)`.

Για να μειώσουμε το σχετικό υπολογιστικό κόστος, έχουμε καθορίσει στον κώδικα να εμφανίζεται ανά 25 frames, χρησιμοποιώντας συνθήκη `if` και το υπόλοιπο της διαίρεσης του αριθμού `e` της επανάληψης της υπορουτίνας με το 25 που είναι ίσο με 0 (`mod(e,25)==0`).

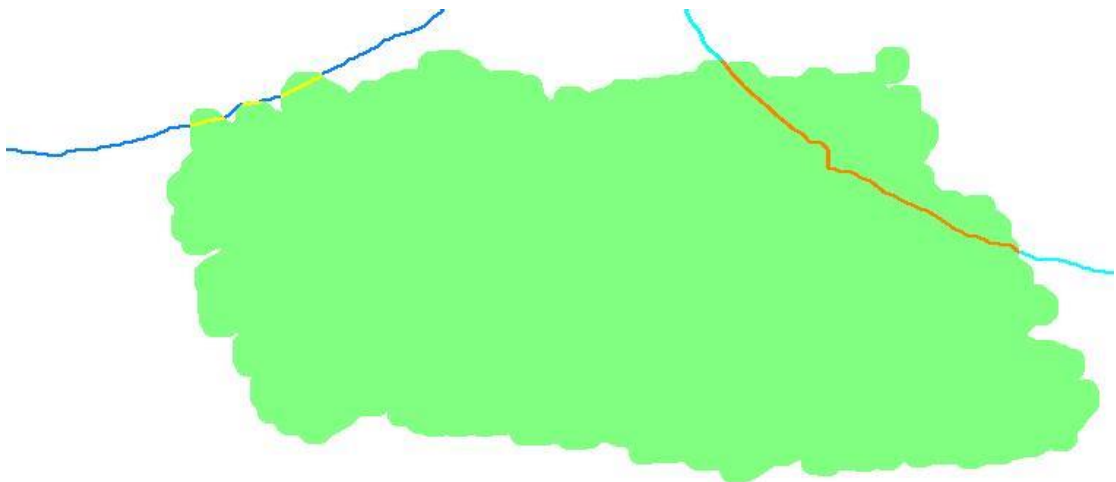
Η τιμή της `e` αυξάνεται κάθε φορά κατά 1 στην αρχή εκτέλεσης της υπορουτίνας. Να σημειωθεί ότι λόγω του ότι ο κώδικας που πραγματοποιεί την παραπάνω λειτουργία, είναι γραμμένος σε υπορουτίνα της MATLAB οι μεταβλητές δεν κρατάνε την παλιά τους τιμή, έχουμε ορίσει την μεταβλητή `e` ως μεταβλητή τύπου `persistent`. Οι μεταβλητές του τύπου αυτού κρατάνε την τιμή τους έως ότου αλλάξει. Η τιμή της `e` παραμένει, μόνο αν την καλέσει η συγκεκριμένη υπορουτίνα. Η αρχική τιμή της `e` θα οριστεί χρησιμοποιώντας την συνθήκη: `if isempty(e) e=1;end`. Στις παρακάτω εικόνες βλέπουμε τα διάφορα στάδια του μετώπου της ροής της ρητίνης.



Εικόνα 41: Τα δύο εντοπισμένα μέτωπα ροής , και το υφάσμα που δεν έχει εμποτιστεί



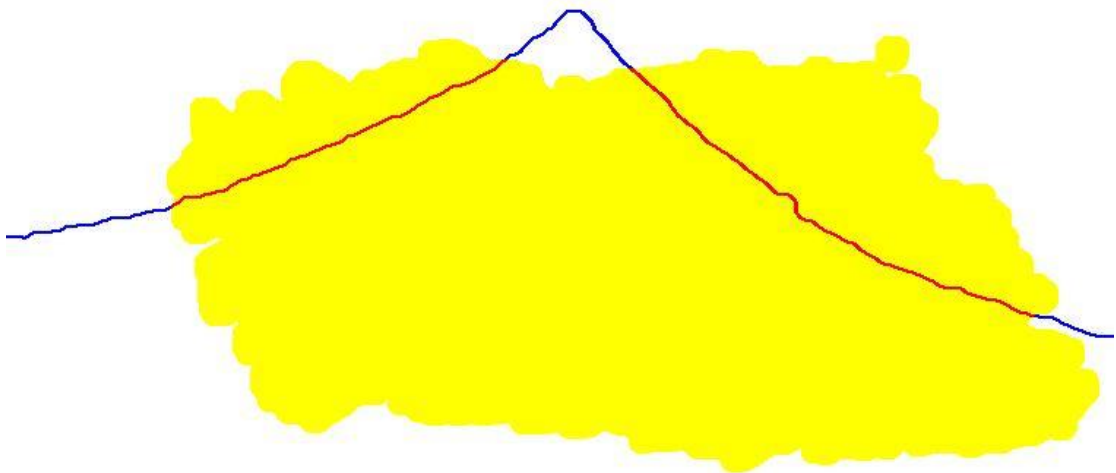
Εικόνα 42: Τα δύο εντοπισμένα μέτωπα ροής, μέρος του υφάσματος έχει εμποτιστεί με ρητίνη από την είσοδο Β



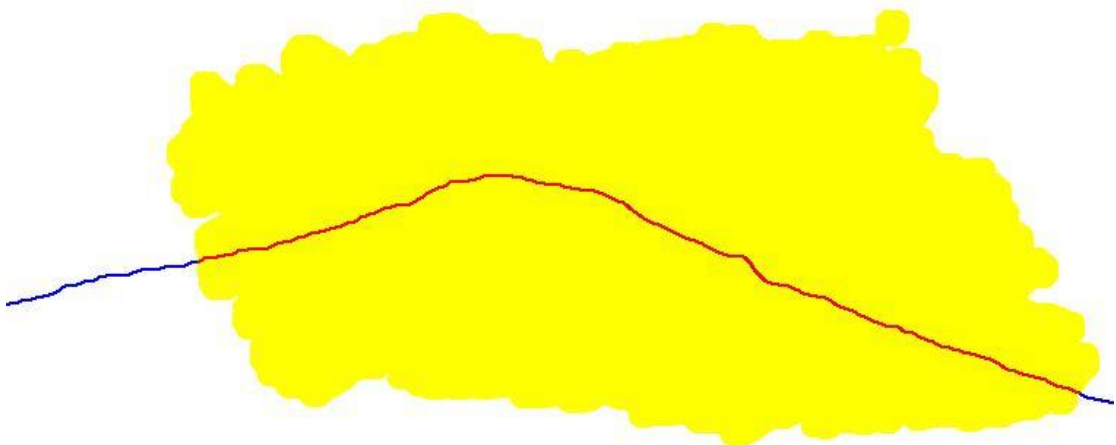
Εικόνα 43: Τα δύο εντοπισμένα μέτωπα ροής, μέρος του υφάσματος έχει εμποτιστεί με ρητίνη από την είσοδο Α και την είσοδο Β



Εικόνα 44: Τα δύο εντοπισμένα μέτωπα της ροής πριν ενωθούν και γίνουν ένα



Εικόνα 45: Τα δύο μέτωπα ροής έχουν ενωθεί, έχουμε ένα ενιαίο μέτωπο ροής



Εικόνα 46: Το μέτωπο ροής της ρητίνης σε μεταγενέστερο χρόνο, με μεγαλύτερο μέρος του υφάσματος εμποτισμένο

4.5.5 Εντοπισμός Εικονοστοιχείων Μετώπου Ροής Στο Ύφασμα

Όπως έχουμε αναφέρει, τα εικονοστοιχεία του μετώπου που βρίσκονται στην περιοχή του υφάσματος προς εμποτισμό, έχουν τιμή 4 ή 5, για την πρώτη ή την ενιαία ροή και για την δεύτερη ροή, αντίστοιχα. Για να εντοπίσουμε τις συντεταγμένες των εικονοστοιχείων αυτών, χρησιμοποιούμε την συνάρτηση της MATLAB `find (Label2==4)` και `find(Label2==5)` αντίστοιχα, εισάγοντας ως όρισμα την συνθήκη των εικονοστοιχείων που θέλουμε να βρούμε, δηλαδή τα εικονοστοιχεία με τιμές ίσες με 4 και 5. Ως αποτέλεσμα έχουμε δύο μεταβλητές για το κάθε μέτωπο, από τις οποίες η πρώτη `row6/row7` είναι η συντεταγμένη των εικονοστοιχείων του μετώπου στην κατακόρυφη διεύθυνση και η δεύτερη `col6/col7`, είναι οι συντεταγμένες των εικονοστοιχείων του μετώπου στην οριζόντια διεύθυνση. Για τον διαχωρισμό, εάν έχουμε δύο μέτωπα ή ένα μέτωπο ροής της ρητίνης, χρησιμοποιούμε την εντολή συνθήκης `if-elseif`, η οποία εξετάζει αν έχουμε δύο μέτωπα ροής ή ένα. Το πλήθος των μετώπων της ροής τα γνωρίζουμε από το πεδίο `NumObjects` του πίνακα δομής `ccc` που έχει δημιουργηθεί στο υπο-πρόγραμμα `vgrammh2roes.m`. Επομένως εάν `ccc.NumObjects==2` είναι αληθές βρίσκουμε συντεταγμένες και για τα δύο μέτωπα, αλλιώς αν `ccc.NumObjects==1`, βρίσκουμε συντεταγμένες για το ένα μέτωπο ροής της ρητίνης.

4.5.6 Κώδικας MATLAB

Ο κώδικας που έχει γραφτεί στη MATLAB για την υλοποίηση των παραπάνω είναι το υπο-πρόγραμμα `vroh.m`. Οι μεταβλητές που εισάγονται είναι: ο πίνακας `label` με τα εντοπισμένα αντικείμενα των μετώπων της ροής, ο πίνακας δομής `ccc`, ο πίνακας `klabelfill` που εμπεριέχει το ύφασμα προς εμποτισμό. Οι μεταβλητές που εξάγονται: ο πίνακας `Label2`, που αποτελείται από τα εικονοστοιχεία των μετώπων ροής και του υφάσματος προς εμποτισμό, οι συντεταγμένες των εικονοστοιχείων των μετώπων ροής `row6,row7,col6,col7`, που είναι κοινές με αυτές του υφάσματος προς εμποτισμό.

```
function
[Label2, row6, col6, row7, col7]=vroh(label, ccc, klabelfill)
persistent e
%% Αρχικές τιμές

if isempty(e) e=1;end
row6=0;col6=0;
row7=0;col7=0;

%% bwconncomp για όλα (ρητίνη και ύφασμα)
Label2=label+klabelfill;
e=e+1;
if mod(e,25)==0
Labelcol1=label2rgb(Label2);
```


ΚΕΦΑΛΑΙΟ 5 Έλεγχος Διαδικασίας VARI

5.1 Εισαγωγή

Υπεύθυνο για την παραμετροποίηση και την σειρά εκτέλεσης των υποπρογραμμάτων που έχουν δημιουργηθεί για την αυτοματοποίηση της VARI, είναι το κύριο πρόγραμμα `twooesvideo.m`, το οποίο έχει γραφτεί στην MATLAB. Εκτός από τον έλεγχο των υποπρογραμμάτων, το κύριο πρόγραμμα επιτελεί και άλλες βασικές λειτουργίες οι οποίες θα περιγραφούν αναλυτικά παρακάτω, με την σειρά εκτέλεσης τους στο πρόγραμμα.

5.2 Βασική Προετοιμασία Προγράμματος

Οι εντολές που θα αναφερθούν παρακάτω εκτελούνται μια φορά κατά την διάρκεια της αυτοματοποιημένης κατεργασίας VARI, στην αρχή αυτής.

5.2.1 Κλείσιμο Ανοιχτών Παραθύρων στη MATLAB

Πριν την έναρξη της Αυτοματοποίησης της VARI, κλείνουν όλα τα ανοιχτά παράθυρα που έχουν δημιουργηθεί στην MATLAB, λόγω εμφάνισης εικόνων ή video, από τυχόν προηγούμενη εκτέλεση προγραμμάτων, εκτελώντας την εντολή της MATLAB `close all`.

5.2.2 Διαγραφή Μεταβλητών

Αρχικά διαγράφονται οι μεταβλητές στη MATLAB από τυχόν άλλα προγράμματα, ή από προηγούμενη εκτέλεση του ίδιου προγράμματος αυτοματοποίησης VARI. Αυτό γίνεται, ώστε να ελευθερωθεί αχρειαστος χώρος στη μνήμη RAM. Επιπλέον υπάρχει η πιθανότητα, να έχουν δημιουργηθεί μεταβλητές από την εκτέλεση άλλων προγραμμάτων στην MATLAB, οι οποίες να έχουν το ίδιο όνομα, με τις μεταβλητές του προγράμματος για την αυτοματοποίηση της VARI. Η ενέργεια αυτή πραγματοποιείται χρησιμοποιώντας την εντολή της MATLAB `clear all`.

5.2.3 Αναγνώριση Κάμερας και Μικροελεγκτή Arduino από την MATLAB

Όταν συνδέσουμε την κάμερα και τον μικροελεγκτή Arduino στον υπολογιστή, για να υπάρχει επικοινωνία μεταξύ αυτών και της MATLAB πρέπει αρχικά να αναγνωριστούν και να παραμετροποιηθούν από αυτήν. Να σημειωθεί πως πρέπει να έχει εγκατασταθεί στη MATLAB το αντίστοιχο πακέτο υποστήριξης υλικού. Για την σύνδεση μέσω θύρας usb διαδικτυακής κάμερας, το αντίστοιχο πακέτο υποστήριξης που πρέπει να είναι εγκατεστημένο είναι το `usbwebcams.mlpkginstall`. Για την σύνδεση του

μικροελεγκτή Arduino το αντίστοιχο πακέτο υποστήριξης είναι το `arduinoio.mlplginstall`.

Για την αναγνώριση της κάμερας εκτελείται η εντολή της MATLAB `webcam`. Έτσι δημιουργείται ένα object της κάμερας με τις ρυθμίσεις της και τα χαρακτηριστικά της. Στην περίπτωση που υπάρχουν συνδεδεμένες παραπάνω από μια κάμερες, χρησιμοποιώντας την εντολή της MATLAB `webcamlist`, η οποία επιστρέφει μια λίστα με τα ονόματα των συνδεδεμένων καμερών, η εντολή `webcam` παίρνει ως όρισμα το όνομα της κάμερας που θέλουμε να χρησιμοποιηθεί ή τον αύξοντα αριθμό της στη λίστα.

Για την αναγνώριση του συνδεδεμένου μικροελεγκτή Arduino, εκτελείται η εντολή της MATLAB `Arduino(port, board)`, όπου ορίζεται η σειριακή θύρα `usb` που είναι συνδεδεμένο το Arduino, καθώς και ο τύπος της πλακέτας Arduino. Τα ορίσματα εισάγονται σε μορφή χαρακτήρων. Στην περίπτωσή μας το Arduino είναι συνδεδεμένο στην θύρα 5, και το Arduino που χρησιμοποιούμε είναι το Arduino uno: `arduino0=arduino('com5','uno')`.

Παρακάτω φαίνονται τα objects που δημιουργούνται στην MATLAB με τα χαρακτηριστικά της κάμερας και του Arduino.

`arduino` with properties:

```
Port: 'COM5'  
Board: 'Uno'  
AvailablePins: {'D2-D13', 'A0-A5'}  
Libraries: {'I2C', 'SPI', 'Servo'}
```

Εικόνα 47: Object Arduino

`webcam` with properties:

```
Name: 'Logitech HD Webcam C525'  
Resolution: '640x480'  
AvailableResolutions: {1x22 cell}  
Contrast: 32  
Tilt: 0  
Gain: 64  
Zoom: 1  
Brightness: 128  
Sharpness: 22  
ExposureMode: 'auto'  
Focus: 60  
Exposure: -4  
BacklightCompensation: 1  
WhiteBalance: 5500  
Pan: 0  
Saturation: 32
```

Εικόνα 48: Object webcam

5.2.4 Έλεγχος Λειτουργίας Βαλβίδων

Πριν την έναρξη της αυτοματοποιημένης κατεργασίας VARI, ελέγχονται οι βαλβίδες για την σωστή λειτουργία τους. Έτσι έχουμε ορίσει να ανοιγοκλείσουν αρκετές φορές εναλλάξ και οι δύο βαλβίδες που χρησιμοποιούνται για την έγχυση της ρητίνης, έτσι ώστε σε περίπτωση μη σωστής λειτουργίας να το παρατηρήσουμε. Για τον σκοπό αυτόν έχουμε δημιουργήσει μια επαναληπτική διαδικασία χρησιμοποιώντας την εντολή της MATLAB *for*. Στο εσωτερικό της *for* χρησιμοποιώντας την εντολή *writeDigitalPin(arduino0,'pin',value)*, ενεργοποιούνται και απενεργοποιούνται οι βαλβίδες ανάλογα με την τιμή της *value*. Χρησιμοποιώντας την εντολή *pause(1)*, η βαλβίδα παραμένει στην κατάσταση που έχει οριστεί για 1 sec.

Στο τέλος της επαναληπτικής διαδικασίας για τον έλεγχο λειτουργίας των βαλβίδων, η MATLAB εμφανίζει στην οθόνη μας το μήνυμα `'if valves operate normally, please press Enter key to continue '`, έτσι αν δεν έχουμε παρατηρήσει κάποια δυσλειτουργία στις βαλβίδες, πατάμε το Enter για να συνεχίσει η αυτοματοποιημένη λειτουργία της VARI. Για την πραγματοποίηση αυτής της λειτουργίας χρησιμοποιούμε την εντολή της MATLAB *input(prompt)*.

5.3 Επεξεργασία Υφάσματος προς Εμποτισμό

5.3.1 Εκτέλεση Υποπρογράμματος *vkaloupi.m*

Στην συνέχεια καλείται το υποπρόγραμμα *vkaloupi.m*, με το οποίο πραγματοποιείται η διαδικασία που έχει περιγραφεί για την επεξεργασία του υφάσματος προς εμποτισμό. Η εκτέλεση του *vkaloupi.m* γίνεται με την εντολή: *[output]=vkaloupi(input)*. Στην θέση του *output* ορίζονται οι μεταβλητές που εξάγονται από την εκτέλεση του υποπρογράμματος *vkaloupi.m*, και στη θέση του *input* ορίζονται οι μεταβλητές που είναι αναγκαίες για την εκτέλεση του *vkaloupi.m*. Να επισημανθεί πως το ύφασμα προς εμποτισμό είναι τοποθετημένο ήδη στο καλούπι, καθώς η φωτογραφία που θα τραβηχτεί για να την επεξεργαστεί το υποπρόγραμμα *vkaloupi.m* πρέπει να περιέχει το ύφασμα.

5.3.2 Αλλαγή Τιμής Εικονοστοιχείων Υφάσματος Εμποτισμού

Από το υποπρόγραμμα *vkaloupi.m* έχει δημιουργηθεί η μεταβλητή *klabelfill*, η οποία είναι ένας πίνακας που εμπεριέχει τα εικονοστοιχεία του υφάσματος προς εμποτισμό. Η τιμή των εικονοστοιχείων της περιοχής του υφάσματος έχει τιμή ίση με 3 όπως έχουμε περιγράψει ήδη. Θα δημιουργήσουμε τον πίνακα *labelimp* ίδιων διαστάσεων με τον *klabelfill*, με την διαφορά ότι η τιμή των εικονοστοιχείων θα είναι -3. Για την δημιουργία

του `labelimp`, εκτελείται η εντολή της MATLAB `labelimp(labelimp==3)==-3` όπου όποιο εικονοστοιχείο έχει τιμή ίση με 3 αντικαθίσταται με την τιμή -3. Η εντολή `labelimp` θα χρησιμοποιηθεί για τον υπολογισμό του ποσοστού εμποτισμού του υφάσματος καθώς και για τον έλεγχο της ροής της ρητίνης από το Arduino. Η σημασία της αλλαγής αυτής στην τιμή των εικονοστοιχείων του υφάσματος προς εμποτισμό θα φανεί παρακάτω.

5.3.3 Υπολογισμός Αριθμού Εικονοστοιχείων Υφάσματος Προς Εμποτισμό

Θα υπολογίσουμε το πλήθος των εικονοστοιχείων του υφάσματος προς εμποτισμό. Ο υπολογισμός αυτός θα χρησιμεύσει για την εξαγωγή του ποσοστού εμποτισμού του υφάσματος. Για τον υπολογισμό του αριθμού εικονοστοιχείων του υφάσματος εκτελείται η εντολή της MATLAB `nkal=numel(find(labelimp==3))`, επομένως υπολογίζεται το πλήθος των εικονοστοιχείων που ικανοποιούν την συνθήκη `labelimp==3`. Η μεταβλητή που περιέχει το πλήθος των εικονοστοιχείων του υφάσματος είναι η `nkal`.

5.3.4 Διακοπή Εκτέλεσης Προγράμματος

Η εκτέλεση του προγράμματος διακόπτεται, μέχρι να δώσουμε την εντολή να συνεχίσει, χρησιμοποιώντας την εντολή της MATLAB `input(prompt)`, με μήνυμα: 'press Enter key to continue' και πατώντας το enter, συνεχίζεται η κανονική ροή του προγράμματος.

Το πρόγραμμα διακόπτεται καθώς για να συνεχιστεί η κατεργασία της αυτοματοποιημένης VARI πρέπει να τοποθετηθούν στο καλούπι το αποκολλητικό ύφασμα (Peel Ply), το αφρώδες δικτυωτό πλέγμα (Peel Ply Econostich), το διάτρητο φιλμ (Release Film), και τέλος η σακούλα κενού. Αφού προετοιμαστεί το καλούπι, ο χρήστης πατάει enter, και συνεχίζεται η κανονική ροή του προγράμματος.

5.4 Βασικοί Υπολογισμοί – Κλήση Υποπρογραμμάτων για Κάθε Στιγμιότυπο

5.4.1 Επαναληπτική Διαδικασία

Οι εντολές που θα αναφερθούν παρακάτω βρίσκονται σε επαναληπτική διαδικασία έτσι ώστε να εκτελούνται κάθε φορά για κάθε καινούριο frame της κάμερας κατά την διάρκεια της αυτοματοποιημένης διαδικασίας VARI. Για την πραγματοποίηση της λειτουργίας αυτής χρησιμοποιούμε την εντολή της MATLAB `while . . . end`, η οποία εξετάζει αν μια συνθήκη είναι αληθής, όσο είναι αληθής, οι εντολές που βρίσκονται στο εσωτερικό της

επαναληπτικής διαδικασίας θα εκτελούνται. Η συνθήκη που έχουμε ορίσει είναι η λογική μεταβλητή `go`, η οποία είναι αληθής. Για την δημιουργία της μεταβλητής `go` με αληθή λογική τιμή, εκτελείται η εντολή `go = true`. Επομένως όσο η μεταβλητή `go` είναι αληθής οι εντολές στο εσωτερικό της `while` θα εκτελούνται. Οι εντολές που εκτελούνται περιγράφονται παρακάτω.

5.4.2 Στιγμιότυπο από την Κάμερα

Δημιουργείται ένα στιγμιότυπο από την κάμερα, εκτελώντας την εντολή της MATLAB `snapshot(cam)`, όπου `cam` είναι το `object` με τα χαρακτηριστικά της κάμερας που χρησιμοποιείται στην MATLAB. Το στιγμιότυπο αυτό το οποίο αποθηκεύεται στην μεταβλητή `videoFrame`, επεξεργάζεται καταλλήλως για τον εντοπισμό του μετώπου της ροής της ρητίνης.

5.4.3 Εκτέλεση Υποπρογράμματος `vgrammh2roes.m`

Στην συνέχεια καλείται το υποπρόγραμμα `vgrammh2roes.m`, με το οποίο πραγματοποιείται η διαδικασία για τον εντοπισμό του μετώπου της ροής της ρητίνης, καθώς και άλλες ενέργειες όπως έχουν ήδη περιγραφεί. Η εκτέλεση του `vgrammh2roes.m` γίνεται με την εντολή: `[output] = vgrammh2roes.m (input)`. Στην θέση του `output` ορίζονται οι μεταβλητές που εξάγονται από την εκτέλεση του υποπρογράμματος `vgrammh2roes.m`, και στη θέση του `input` ορίζονται οι μεταβλητές που είναι αναγκαίες για την εκτέλεση του `vgrammh2roes.m`.

5.4.4 Εκτέλεση Υποπρογράμματος `vroh.m`

Μετά το τέλος της εκτέλεσης του `vgrammh2roes.m`, εκτελείται το υποπρόγραμμα `vroh.m`. Η εκτέλεση του γίνεται με την εντολή `[output] = vroh.m(input)`, όπου στην θέση του `output` ορίζονται οι μεταβλητές που εξάγονται από την εκτέλεση του υποπρογράμματος `vroh.m`, και στη θέση του `input` ορίζονται οι μεταβλητές που είναι αναγκαίες για την εκτέλεση του `vroh.m`.

5.4.5 Υπολογισμός Περιοχής Εμποτισμένου Υφάσματος Με Ρητίνη

5.4.5.1 Εισαγωγή

Κατά την διάρκεια εκτέλεσης της αυτοματοποιημένης διαδικασίας VARI, καθώς το μέτωπο της ροής της ρητίνης προχωράει, το ύφασμα εμποτίζεται. Τα μέτωπο ροής της ρητίνης όπως και το ύφασμα προς εμποτισμό, έχουν ανιχνευτεί ως ξεχωριστά αντικείμενα. Όπως

γνωρίζουμε τα εικονοστοιχεία των μετώπων της ροής της ρητίνης έχουν τιμή ίση με 1 για το πρώτο ή το ενιαίο μέτωπο ροής και ίση με 2 για το δεύτερο μέτωπο ροής. Ο πίνακας που εμπεριέχει τα εικονοστοιχεία των μετώπων της ροής της ρητίνης είναι ο `label`. Για τα εικονοστοιχεία του υφάσματος δεν θα χρησιμοποιήσουμε πλέον τον πίνακα `klabelfill` που έχει δημιουργηθεί από το υποπρόγραμμα `vkaloupi.m`, όπου τα εικονοστοιχεία του υφάσματος έχουν τιμή ίση με 3, αλλά τον πίνακα `labelimp` που έχει δημιουργηθεί από το κύριο πρόγραμμα στην αρχή της εκτέλεσής του, όπου τα εικονοστοιχεία του υφάσματος έχουν τιμή ίση με -3.

5.4.5.2 Μέθοδος

Σε κάθε επανάληψη της επαναληπτικής διαδικασίας `while`, για κάθε `frame`, ο πίνακας `label` με τα μέτωπα της ροής της ρητίνης, προστίθεται στον πίνακα `labelimp` του υφάσματος, χρησιμοποιώντας την εντολή `labelimp=labelimp+label`. Έτσι κάθε φορά, στον πίνακα `labelimp` τα εικονοστοιχεία του μετώπου της ροής της ρητίνης τα οποία έχουν τις ίδιες συντεταγμένες με εικονοστοιχεία του υφάσματος προς εμποτισμό αλλάζουν τιμή, καθώς προστίθεται η τιμή του εικονοστοιχείου του μετώπου της ροής. Η εκτέλεση της εντολής `labelimp=labelimp+label`, γίνεται στο κύριο πρόγραμμα, επομένως η τιμή της μεταβλητής `labelimp` παραμένει και στην επόμενη εκτέλεση της επαναληπτικής διαδικασίας. Αυτό έχει ως αποτέλεσμα, τα εικονοστοιχεία του υφάσματος που έχουν αλλάξει τιμή λόγω του μετώπου της ροής της ρητίνης σε προηγούμενη επανάληψη να παραμένουν και στη τρέχουσα εκτέλεση των εντολών μέσα στην επαναληπτική διαδικασία. Επομένως τα εικονοστοιχεία των αντίστοιχων σημείων του υφάσματος που δεν έχουν εμποτιστεί, έχουν τιμή ίση με -3, ενώ τα εικονοστοιχεία των αντίστοιχων σημείων του υφάσματος που έχουν εμποτιστεί έχουν τιμή μεγαλύτερη του -3. Ο πίνακας `labelimp` που υπολογίζεται σε κάθε επανάληψη για κάθε `frame`, χρησιμοποιείται στο υποπρόγραμμα `impregnation.m` για τον υπολογισμό του ποσοστού εμποτισμού του υφάσματος από την ρητίνη, και στο υποπρόγραμμα `arduinno.m` για τον έλεγχο της ροής της ρητίνης από το Arduino.

5.4.5.3 Πιθανότητα μη Αξιόπιστου Υπολογισμού Εμποτισμένης Περιοχής

Στην περίπτωση που η ταχύτητα της ροής είναι αρκετά μεγάλη, και ο χρόνος εκτέλεσης του κώδικα αρκετά μεγάλος, υπάρχει η πιθανότητα, το μέτωπο της ροής να έχει προχωρήσει αρκετά, χωρίς να έχει επεξεργαστεί το αντίστοιχο `frame`. Αυτό θα έχει ως αποτέλεσμα, κάποια εικονοστοιχεία του υφάσματος να μην έχουν αλλάξει τιμή, ενώ τα αντίστοιχα σημεία του υφάσματος έχουν εμποτιστεί από την ρητίνη. Αυτό το γεγονός θα δημιουργήσει

πρόβλημα στον υπολογισμό του ποσοστού εμποτισμού του υφάσματος καθώς και στον έλεγχο της ρητίνης από το Arduino.

Για την αποφυγή του προβλήματος αυτού, θα χρησιμοποιήσουμε την μορφολογική λειτουργία της διαστολής στον πίνακα `linenew` ο οποίος έχει υπολογιστεί στο υποπρόγραμμα `vgrammh2goes.m` εκτελώντας την συνάρτηση της `imdilate` (`linenew,nhoodimp`), όπου ως ορίσματα εισάγουμε την εικόνα που θέλουμε να μορφοποιήσουμε και το στοιχείο δόμησης `nhoodimp`. Η μορφή του στοιχείου δόμησης πρέπει να είναι τέτοια ώστε να προσθέτει εικονοστοιχεία μόνο στην πίσω μεριά του μετώπου της ροής, επομένως τα εικονοστοιχεία του στοιχείου δόμησης τα οποία βρίσκονται κάτω από το κεντρικό εικονοστοιχείο, είναι όλα μηδενικά. Το στοιχείο δόμησης που έχουμε δημιουργήσει για αυτήν την περίπτωση είναι :

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

Στη συνέχεια εκτελείται η συνάρτηση της MATLAB `bwlabel`, ώστε να εντοπιστούν τα μέτωπα ροής ως ξεχωριστά αντικείμενα, το αποτέλεσμα της `bwlabel` είναι ο πίνακας `labelimpdi`, ο οποίος θα χρησιμοποιείται στην θέση του `label`. Οι παραπάνω εντολές εκτελούνται στο υποπρόγραμμα `impdilation.m`

5.4.6 Υπολογισμός Ποσοστού Εμποτισμού Υφάσματος

5.4.6.1 Εισαγωγή

Κατά την διάρκεια της αυτοματοποιημένης κατεργασίας VARI, υπολογίζεται το ποσοστό εμποτισμού `imp` έτσι ώστε να γνωρίζουμε πόση επιφάνεια του υφάσματος προς εμποτισμό έχει εμποτιστεί με ρητίνη. Το ποσοστό εμποτισμού `imp` εκτός από το να ενημερώνεται ο χρήστης για την εξέλιξη της διαδικασίας, είναι βασικός παράγοντας για τον έλεγχο της ροής της ρητίνης από το Arduino.

5.4.6.2 Μέθοδος

Ο υπολογισμός του ποσοστού εμποτισμού στηρίζεται στην ανίχνευση των εικονοστοιχείων εκείνων του υφάσματος προς εμποτισμό σε κάθε `frame`, που δεν έχει αλλάξει η τιμή τους μετά την έκχυση της ρητίνης. Τα βήματα που πραγματοποιούνται είναι τα εξής :

1. Υπολογίζεται ο αριθμός των εικονοστοιχείων του υφάσματος πριν τον εμποτισμό. Το πλήθος των εικονοστοιχείων n_{kal} του υφάσματος τον έχουμε υπολογίσει στην αρχή του προγράμματος.
2. Υπολογίζεται ο αριθμός των εικονοστοιχείων εκείνων του υφάσματος που κατά την διάρκεια της έγχυσης της ρητίνης έχουν μείνει αναλλοίωτα, δηλαδή η τιμή τους παραμένει -3. Για τον υπολογισμό των εικονοστοιχείων αυτών χρησιμοποιείται η εντολή της MATLAB `nn=numel(find(labelimp==-3))`, επομένως υπολογίζεται το πλήθος των εικονοστοιχείων αυτών που ικανοποιούν την συνθήκη `labelimp==-3`. Να σημειωθεί πως η `labelimp` αλλάζει σε κάθε επανάληψη, αφού καθώς το μέτωπο της ροής προχωράει, τα αντίστοιχα εικονοστοιχεία των σημείων του υφάσματος που έχουν εμποτιστεί έχουν τιμή διάφορη του -3. Η μεταβλητή που περιέχει το πλήθος των εικονοστοιχείων του υφάσματος που έχουν μείνει αναλλοίωτα κατά την διάρκεια του εμποτισμού σε κάθε επανάληψη, είναι η nn .
3. Σε κάθε επανάληψη υπολογίζεται το ποσοστό των εικονοστοιχείων του υφάσματος που έχουν μείνει αναλλοίωτα, επομένως εκτελείται η εντολή `imp=nn*100/nkal`.
4. Το παραπάνω ποσοστό που έχει βρεθεί αφαιρείται από το 100, έτσι ώστε να έχουμε το ποσοστό των εικονοστοιχείων του υφάσματος που έχουν αλλάξει τιμή, δηλαδή που τα αντίστοιχα σημεία του υφάσματος έχουν εμποτιστεί. Στη MATLAB εκτελείται η εντολή `imp=round((100-imp),2)`. Έτσι έχουμε ως αποτέλεσμα έναν αριθμό στρογγυλοποιημένο στο δεύτερο δεκαδικό ψηφίο χρησιμοποιώντας τη συνάρτηση της MATLAB `round` και εισάγοντας ως όρισμα εκτός από τον αριθμό που θα στρογγυλοποιηθεί, την τιμή που είναι ίση με 2.

Οι εντολές της MATLAB για τον υπολογισμό του `imp`, εκτελούνται στο υποπρόγραμμα `impregnation.m`, το οποίο καλείται από το κύριο πρόγραμμα σε κάθε επανάληψη.

Ο υπολογισμός του ποσοστού εμποτισμού πραγματοποιείται μόνο στην περίπτωση που έχουν εντοπιστεί τα μέτωπα ροής, και δεν έχει εμποτιστεί πλήρως το ύφασμα. Χρησιμοποιώντας την εντολή της MATLAB `if`, ελέγχοντας την συνθήκη `ccc.NumObjects <=2 && nkal~=nn && ccc.NumObjects ~=0`, αν είναι αληθής εκτελείται το τμήμα του κώδικα για τον υπολογισμό του `imp`.

5.4.6.3 Σημασία Αρνητικής Αρχικής Τιμής Εικονοστοιχείων Υφάσματος

Όπως έχουμε ήδη αναφέρει, στην αρχή του προγράμματος, πριν εκτελεστούν οι εντολές που βρίσκονται μέσα στην επαναληπτική διαδικασία, δημιουργείται ο πίνακας `labelimp`,

στον οποίο η τιμή των εικονοστοιχείων του υφάσματος πριν την έγχυση της ρητίνης είναι ίση με -3.

Γνωρίζουμε ότι τα εικονοστοιχεία των μετώπων της ροής της ρητίνης έχουν τιμή ίση με 1 για το πρώτο ή το ενιαίο μέτωπο ροής και ίση με 2 για το δεύτερο μέτωπο ροής. Σε κάθε επαναληπτική διαδικασία τα μέτωπα ροής προστίθενται στον πίνακα `labelimp` στον οποίο η τιμή των εικονοστοιχείων όπου περνάει το μέτωπο αυξάνεται είτε αυτό ανήκει στην περιοχή του υφάσματος, είτε έξω από αυτήν. Επομένως υπάρχει η περίπτωση στο ίδιο εικονοστοιχείο της `labelimp` να έχει προστεθεί η τιμή του εικονοστοιχείου του μετώπου της ροής περισσότερες φορές από μία, καθώς το μέτωπο της ροής μπορεί να μην έχει αλλάξει θέση, με αποτέλεσμα να δημιουργηθεί εικονοστοιχείο το οποίο να έχει τιμή ίση με 3 και δεν ανήκει στην περιοχή του υφάσματος. Για τον λόγο αυτόν δεν χρησιμοποιούμε τον πίνακα `klabelfill` στον οποίον τα εικονοστοιχεία του υφάσματος έχουν τιμή ίση με 3.

Ακόμη και αν επιλεγόταν άλλη τιμή για το ύφασμα, π.χ. θετική ακέραια, υπάρχει η πιθανότητα μετά από πολλές επαναλήψεις να δημιουργηθεί ο αριθμός αυτός λόγω των τιμών των μετώπων της ροής που προστίθενται κάθε φορά. Επομένως η πιο αξιόπιστη λύση είναι η τιμή των εικονοστοιχείων να είναι αρνητική. Δεν υπάρχει περίπτωση να δημιουργηθεί εικονοστοιχείο με τιμή ίση ή μικρότερη από την αρνητική τιμή, αφού κάθε φορά στον πίνακα `labelimp` προστίθεται ο πίνακας `label` ο οποίος περιέχει μη αρνητικούς.

5.4.7 Εκτέλεση Υποπρογράμματος `arduinno.m`

Στο τέλος του κύριου προγράμματος καλείται το υποπρόγραμμα `arduinno.m` για τον έλεγχο της ροής της ρητίνης. Η εκτέλεση του γίνεται με την εντολή `[]= arduinno.m(input)`, όπου στη θέση του `input` ορίζονται οι μεταβλητές που είναι αναγκαίες για την εκτέλεση του `arduinno.m`.

5.5 Κώδικας MATLAB

5.5.1 Κώδικας `twooesvideo.m`

Ο κώδικας του κύριου προγράμματος της MATLAB `twooesvideo.m` είναι:

```
function twooesvideo()  
close all  
clear all  
  
nn=0;  
nkal=0;  
imp=0;  
%% set camera
```

```

cam=webcam(1)
%% set arduino
arduino=arduino('com5','uno')

%% έλεγχος βαλβίδων
for nval=1:5
writeDigitalPin(arduino,'D10',1)
pause(1)
writeDigitalPin(arduino,'D10',0)
writeDigitalPin(arduino,'D8',1)
pause(1)
writeDigitalPin(arduino,'D8',0)
end
%% επιβεβαίωση κανονικής λειτουργίας
prompt = 'if valves operate normally, please press Enter key
to continue ';
input(prompt)
% -----
[kedgeopen,kcc,klabel,klabelfill,chpointA1,chpointA2,chpointA3
,
chpointA4,chpointA5, ...
chpointA6,chpointA7,chpointA8,chpointA9,chpointB1, ...
chpointB2,chpointB3,chpointB4,chpointB5,chpointB6, ...
chpointB7,chpointB8,chpointB9,xmin,ymin,width,height]=vkaloupi
(cam);
MaXy=size(kedgeopen,1);
MaXy=lasty3+((size(kedgeopen,1)-lasty3)*2/3)
%% Αλλαγή τιμής εικονοστοιχείων υφάσματος
labelimp=klabelfill;
labelimp(labelimp==3)=-3;% το αντικαθιστώ με -3 διότι δεν
%μπορεί να δημιουργηθεί αρνητικός αριθμός από την πρόσθεση
%της ρητίνης επομένως όταν είναι -3 η ρητίνη δεν έχει
%εμποτίσει το σημείο αυτό.

nkal=numel(find(labelimp==-3));% number of pixels για το
%ύφασμα, το χρησιμοποιώ για
%ποσοστό εμποτισμού

%% διακοπή μέχρι η διαδικασία να είναι έτοιμη
prompt = 'press Enter key to continue ';
input(prompt)
%%%
close all;
go=true;
while go

videoFrame=snapshot(cam);
[edgeopen]=processing2roes(videoFrame,xmin,ymin,width,height);
[linenew,label,ccc,x1,x2,y1,y2,ploint1,ploint2,ploint3, ...
p2oint1,p2oint2,p2oint3]=vgrammh2roes(edgeopen);
% ccc.NumObjects
[Label2,row6,col6,row7,col7]=vrroh(label,ccc,klabelfill);

%% εύρεση ποσοστού εμποτισμού

```



```

if ccc.NumObjects <=2  && nkal~=nn && ccc.NumObjects ~=0

%%-----
-
% στην περίπτωση που λόγω ταχύτητας ροής και καθυστέρησης
% προγράμματος το καλούπι που έχει εμποτιστεί δεν φαίνεται,
για να
% μην έχουμε λανθασμένο ποσοστό           %εμποτισμού κάνουμε
dilation
%προς τα πίσω.

[labelimpdi]=impdilation(linenew);
labelimp=labelimp+labelimpdi;

%% - - - - -
labelimp=labelimp+label; %στην περίπτωση που εκτελούνται οι
                        % παραπάνω εντολές, η εντολή αυτή δεν
                        % εκτελείται
[imp]=impregnation(labelimp,nkal);
end
%% control arduino
arduino(arduino0,y1,y2,ccc,imp,labelimp,MaXy,chpointA1,chpoint
A2,chpointA3,chpointA4,chpointA5,chpointA6,chpointA7,chpointA8
,chpointA9,...

chpointB1,chpointB2,chpointB3,chpointB4,chpointB5,chpointB6,ch
pointB7,chpointB8,chpointB9,...
    ppoint2,p2oint2)
    ppoint1,ppoint3,p2oint1,p2oint3,
shm61,shm63,shm62,shm71,shm73,shm72)
end

end

```

5.5.2 Κώδικας impdilation.m

Ο κώδικας του υποπρογράμματος impdilation.m είναι:

```

function [labelimpdi]=impdilation(linenew)

nhoodimp=[0 1 0;1 1 1;1 1 1;1 1 1;1 1 1;0 0 0;0 0 0;0 0 0;0 0
0;0 0 0]; %
linenewimp=imdilate(linenew,nhoodimp);

labelimpdi=bwlabel(linenewimp);
end

```

ΚΕΦΑΛΑΙΟ 6 Έλεγχος έγχυσης ρητίνης

6.1 Εισαγωγή

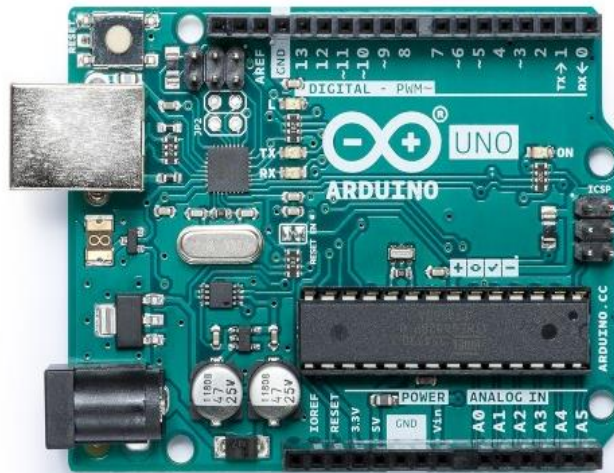
Για τον έλεγχο της έγχυσης της ρητίνης χρησιμοποιείται ο μικροελεγκτής Arduino. Μέσω του Arduino, οι δύο πνευματικές βαλβίδες που χρησιμοποιούνται για την έγχυση της ρητίνης από τις δύο διαφορετικές εισόδους θα ενεργοποιούνται ή θα απενεργοποιούνται, ελέγχοντας κάθε φορά συγκεκριμένες συνθήκες. Οι συνθήκες που έχουν ορισθεί καθώς και ο κώδικας που έχει γραφτεί στη MATLAB για την πραγματοποίηση του ελέγχου της ροής της ρητίνης, θα αναλυθούν παρακάτω, αφού γίνει μια παρουσίαση του μικροελεγκτή.

6.2 Arduino

Το Arduino είναι μια ανοιχτού κώδικα πλατφόρμα πρωτοτυποποίησης ηλεκτρονικών βασισμένη σε ευέλικτο και εύκολο στη χρήση hardware και software. Μπορεί να χρησιμοποιηθεί για την ανάπτυξη αυτόνομων διαδραστικών αντικειμένων ή μπορεί να συνδεθεί κατευθείαν στον υπολογιστή και η λειτουργία του να ελέγχεται από αυτόν. Αποτελείται από δύο κύρια μέρη: την πλακέτα πάνω στην οποία εργαζόμαστε και συνδέουμε τα διάφορα ηλεκτρονικά στοιχεία για την υλοποίηση των επιθυμητών εργασιών, και από το Arduino IDE, ένα πρόγραμμα που εκτελείται στον υπολογιστή και δημιουργεί τον κώδικα που χρειάζεται για να λειτουργήσει η πλακέτα και να εκτελέσει τις επιθυμητές λειτουργίες.

6.2.1 Πλακέτα Arduino

Σημαντικό μέρος της μεθόδου υλικού ανοιχτού κώδικα είναι να παρέχει πολλές επιλογές στους προγραμματιστές για την υλοποίηση των εφαρμογών τους. Έτσι και για το Arduino έχουν κατασκευαστεί αρκετές διαφορετικές πλακέτες οι οποίες επιτρέπουν την υλοποίηση διαφορετικών εφαρμογών. Οι σημαντικότερες είναι οι Arduino Uno, Arduino Due, Arduino Leonardo, Arduino Mega, Arduino Micro, Arduino Esplora, Arduino Yun, Arduino Ethernet, LilyPad Arduino. Στην παρούσα εργασία έχει χρησιμοποιηθεί το Arduino Uno, το οποίο είναι η πιο διαδεδομένη πλακέτα Arduino. Η περιγραφή της πλακέτας Arduino Uno θα γίνει παρακάτω.



Εικόνα 49: Πλακέτα Arduino UNO

Μικροελεγκτής ATmega328

Η πλακέτα του Arduino Uno βασίζεται στον ATmega328, έναν 8-bit RISC μικροελεγκτή, ο οποίος χρονίζεται στα 16MHz. Ο ATmega328 διαθέτει ενσωματωμένη μνήμη τριών τύπων.

- 2Kb μνήμης SRAM, η οποία είναι η ωφέλιμη μνήμη που μπορεί να χρησιμοποιήσει το πρόγραμμά μας για την αποθήκευση μεταβλητών και πινάκων. Η μνήμη SRAM επειδή είναι τύπου RAM, όταν πάψει να διαρρέεται από ρεύμα ή γίνει επανεκκίνηση στον μικροελεγκτή, χάνει όλα τα δεδομένα της.
- 1Kb μνήμης EEPROM, η οποία μπορεί να χρησιμοποιηθεί για εγγραφή/ανάγνωση δεδομένων από το πρόγραμμά μας κατά τον χρόνο εκτέλεσής του. Σε αντίθεση με την SRAM, η EEPROM δεν χάνει το περιεχόμενό της σε περίπτωση απώλειας της τροφοδοσίας ή επανεκκίνησης του μικροελεγκτή.
- 32Kb μνήμης Flash, από τα οποία τα 2Kb χρησιμοποιούνται από το firmware του Arduino το οποίο υπάρχει εκεί από την αγορά του. Το firmware αυτό, το οποίο στην ορολογία του Arduino ονομάζεται Bootloader είναι αναγκαίο για την σύνδεση του με τον Η/Υ και για την μεταφορά του προγράμματός στο Arduino μέσω της θύρας USB, χωρίς να χρειάζεται εξωτερικός hardware programmer. Τα υπόλοιπα 30Kb χρησιμοποιούνται για την αποθήκευση του προγράμματος, αφού πρώτα μεταγλωττιστεί στον Η/Υ. Η μνήμη Flash όπως και η EEPROM δεν χάνει το περιεχόμενό της ύστερα από απώλεια ρεύματος ή επανεκκίνησης του μικροελεγκτή. Η μνήμη Flash υπό κανονικές συνθήκες, δεν προορίζεται για χρήση από το πρόγραμμά μας κατά την διάρκεια εκτέλεσης αυτού. Παρόλα αυτά λόγω της μικρής συνολικής μνήμης που είναι

διαθέσιμη για την αποθήκευση μεταβλητών, πινάκων και άλλων (2Kb SRAM + 1Kb EEPROM), διατίθεται μια βιβλιοθήκη στο Arduino που επιτρέπει την χρήση του χώρου της μνήμης Flash που έχει περισσέψει ύστερα από την μεταγλώττιση του προγράμματός μας.

Είσοδοι/Εξοδοι Arduino

Η πλακέτα του Arduino παρέχει εισόδους και εξόδους που προορίζονται για διαφορετική χρήση.

- Αρχικά διαθέτει σειριακή επικοινωνία την οποία προωθεί μέσα από έναν ελεγκτή Serial-over-USB ώστε να συνδέεται με τον υπολογιστή μέσω USB. Η σύνδεση αυτή χρησιμοποιείται για την μεταφορά του προγράμματος στο Arduino και γενικά για την αμφίδρομη επικοινωνία μεταξύ του υπολογιστή και του Arduino κατά την διάρκεια εκτέλεσης του προγράμματος.
- Διαθέτει 14 pin, τα οποία είναι αριθμημένα από 0 έως 13 και μπορούν να λειτουργήσουν είτε ως ψηφιακές εισοδοι είτε ως ψηφιακές έξοδοι. Η τάση λειτουργίας τους είναι 5V και το μέγιστο ρεύμα που μπορεί να δεχτεί το κάθε pin είναι 40mA. Μερικά από τα pin που αναφέρθηκαν έχουν και άλλες λειτουργίες, οι οποίες θα αναφερθούν παρακάτω.
- Τα pin 0 και 1 λειτουργούν ως RX και TX αντίστοιχα της σειριακής επικοινωνίας, όταν αυτά ενεργοποιηθούν. Ο RX δέχεται δεδομένα από εξωτερικές συσκευές, για παράδειγμα από μια δεύτερη πλακέτα Arduino. Και ο TX στέλνει δεδομένα από το Arduino σε κάποια άλλη συσκευή.
- Τα pin 2 και 3 λειτουργούν και ως εξωτερικοί διακόπτες (interrupt), δηλαδή μπορούν να διακόψουν την κανονική ροή του προγράμματος όταν υπάρξει αλλαγή στην κατάσταση τους, από μηδενική τάση σε υψηλή τάση ή και το αντίθετο και να εκτελεστεί μια σειρά από εντολές.
- Τα pin 3, 5, 6, 9, 10 και 11 μπορούν να χρησιμοποιηθούν ως αναλογικές έξοδοι με το σύστημα PWM (Pulse Width Modulation). Το PWM δεν είναι πραγματικά αναλογικό σύστημα, αλλά προσομοιώνει ένα αναλογικό σήμα εναλλάσσοντας τον ψηφιακό παλμό με μεγάλη συχνότητα και για ίσους χρόνους μεταξύ 0 V και 5 V.
- Διαθέτει 6 pin με την σήμανση “ analog in “ αριθμημένα από το 0 έως το 5, τα οποία λειτουργούν ως αναλογικές εισοδοι χρησιμοποιώντας τον αναλογικό-ψηφιακό μετατροπέα (ADC) που είναι ενσωματωμένος στον μικροελεγκτή. Η ανάλυση του ADC είναι 10-bit, που σημαίνει ότι η ελάχιστη τιμή που μπορεί να πάρει είναι 0 και η

μέγιστη τιμή 1023. Η τάση αναφοράς μπορεί να ρυθμιστεί με μια εντολή στα 1.1 V ή μεταξύ 2 V και 5 V τροφοδοτώντας εξωτερικά το pin με την ένδειξη AREF. Τα pin αυτά, εκτός από αναλογικές είσοδοι μπορούν να μετατραπούν και σε ψηφιακά pin εισόδου/εξόδου εκτελώντας την κατάλληλη εντολή μέσα από το πρόγραμμα. Στην περίπτωση αυτή μετονομάζονται σε pin 14-19.

Τροφοδοσία Arduino Uno

Το Arduino μπορεί να τροφοδοτηθεί με ρεύμα από τον υπολογιστή μέσω της σύνδεσης USB, είτε από εξωτερική τροφοδοσία που παρέχεται μέσω της σύνδεσης εξωτερικής πηγής στην υποδοχή των 2.1mm ή στους ακροδέκτες εξωτερικής τροφοδοσίας Vin και GND. Η εξωτερική τροφοδοσία για την ομαλή λειτουργία του Arduino πρέπει να είναι από 7 V έως 12 V. Στην πλακέτα του Arduino υπάρχουν 6 ακροδέκτες που έχουν σχέση με την τροφοδοσία του Arduino. Αυτοί είναι :

- **Vin:** εκεί συνδέεται ο θετικός πόλος της εξωτερικής πηγής τάσης και σε συνδυασμό με τον ακροδέκτη γείωσης δίπλα από αυτό, λειτουργεί ως μέθοδος εξωτερικής τροφοδοσίας. Στην περίπτωση που έχουμε συνδέσει εξωτερική πηγή τάσης μέσω της υποδοχής 2.1mm , το pin Vin μπορεί να χρησιμοποιηθεί για την τροφοδότηση εξαρτημάτων με τάση από την εξωτερική τροφοδοσία, πριν περάσει από τον regulator τάσης του Arduino και η τάση γίνει 5 V.
- **5 V:** Το pin αυτό μπορεί να τροφοδοτήσει τα ηλεκτρονικά εξαρτήματα με τάση 5 V. Προέρχεται είτε από την εξωτερική τροφοδοσία είτε από την θύρα USB.
- **3.3 V:** Το pin αυτό μπορεί να τροφοδοτήσει τα ηλεκτρονικά εξαρτήματα με τάση 3.3 V. Προέρχεται μόνο από την τάση που παράγεται από τον ελεγκτή Serial-over-USB.
- **RESET:** Το pin αυτό, αν γειωθεί σε οποιοδήποτε από τα pin γείωσης (GND), προκαλεί επανεκκίνηση του μικροελεγκτή.
- **GND:** γείωση.
- **IOREF:** Το pin αυτό παρέχει την τάση αναφοράς για την λειτουργία του μικροελεγκτή.

Σύνδεση Arduino με εξωτερικές συσκευές

Ο βασικός τρόπος επικοινωνίας μεταξύ των συσκευών Arduino είναι η σειριακή επικοινωνία. Η σειριακή επικοινωνία απαιτεί μικρό αριθμό συνδέσεων μεταξύ των συσκευών σε σχέση με άλλα πρωτόκολλα και παρέχει έναν απλό τρόπο μεταφοράς δεδομένων μεταξύ αυτών των διαφορετικών συσκευών.

Η πλακέτα του Arduino υποστηρίζει τρεις διαφορετικούς τρόπους επικοινωνίας:

- Η βασική σειριακή που γίνεται μέσω θύρας USB ή μέσω των pin RX, TX.
- Η σειριακή περιφερειακή διασύνδεση (θύρα SPI)
- I²C θύρα

Για τα παραπάνω πρωτόκολλα σύνδεσης υπάρχουν ειδικές βιβλιοθήκες.

Πίνακας 2: Χαρακτηριστικά Arduino Uno

Microcontroller	ATmega328P
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

6.2.2 Γλώσσα προγραμματισμού Arduino

6.2.2.1 Εισαγωγή

Για την διαχείριση του Arduino από τον υπολογιστή, παρέχεται το πρόγραμμα Arduino IDE, (ανοιχτό λογισμικό). Η γλώσσα η οποία υποστηρίζει είναι η λεγόμενη Wiring η οποία είναι μια παραλλαγή της C/C++ για μικροελεγκτές AVR όπως ο ATmega. Επομένως μπορούν να χρησιμοποιηθούν αυτούσιες οι βασικές εντολές, συναρτήσεις της C/C++ καθώς και οι ίδιοι τύποι δεδομένων και τελεστών. Εκτός από αυτές τις εντολές έχουν δημιουργηθεί κάποιες

ειδικές εντολές οι οποίες βοηθούν την διαχείριση του hardware του Arduino. Στην παρούσα εργασία, δεν χρησιμοποιείται το Arduino IDE, αλλά χρησιμοποιείται το MATLAB για τον έλεγχο του Arduino.

6.2.2.2 Προγραμματισμός Arduino μέσω MATLAB

Χρησιμοποιώντας το πακέτο υποστήριξης του MATLAB για Arduino, μπορούμε να χρησιμοποιήσουμε το MATLAB για να επικοινωνήσουμε διαδραστικά με την πλακέτα του Arduino. Το πακέτο υποστήριξης της πλακέτας Arduino επιτρέπει:

- λήψη δεδομένων από τους αναλογικούς ή ψηφιακούς αισθητήρες που είναι συνδεδεμένοι στην πλακέτα του Arduino.
- έλεγχο άλλων συσκευών χρησιμοποιώντας τις ψηφιακές ή τις PWM εξόδους
- έλεγχο DC κινητήρων, servo, βηματικών κινητήρων καθώς και άλλων εξαρτημάτων.
- πρόσβαση σε άλλες συσκευές ή αισθητήρες χρησιμοποιώντας τα I²C ή SPI πρωτόκολλα επικοινωνίας.
- σύνδεση του Arduino μέσω USB ή ασύρματα
- ενσωμάτωση έτοιμων βιβλιοθηκών ή δημιουργία καινούριων για την υλοποίηση του προγράμματός μας και τον έλεγχο του Arduino.

Το πρόγραμμα που χρειάζεται για την υλοποίηση της εφαρμογής μας με το Arduino γράφεται στο editor του MATLAB . Επειδή το MATLAB είναι μια γλώσσα υψηλού επιπέδου, μπορούμε να δούμε τα αποτελέσματα του κώδικα χωρίς να κάνουμε compiling. Όπως και στο Arduino IDE που μπορεί να χρησιμοποιηθούν αυτούσιες οι βασικές εντολές, συναρτήσεις της C/C++ καθώς και οι ίδιοι τύποι δεδομένων και τελεστών, στο MATLAB ισχύει αντίστοιχα το ίδιο. Επιπλέον έχουν αναπτυχθεί ειδικές εντολές στο MATLAB για την διαχείριση της πλακέτας του Arduino. Οι βασικότερες είναι:

- **Arduino:** ανιχνεύει σε ποια σειριακή θύρα του υπολογιστή είναι συνδεδεμένη η πλακέτα του Arduino και δημιουργεί ένα object για να μπορέσουμε να την προγραμματίσουμε. Επιπλέον μέσω αυτής της εντολής φορτώνονται οι βασικές βιβλιοθήκες του MATLAB για το Arduino (I2C, 'Servo', 'SPI') ή βιβλιοθήκες που έχουν δημιουργηθεί από εμάς ή άλλους χρήστες.
- **configurePin:** ρυθμίζει τη λειτουργία ενός pin ανάλογα με την παράμετρο που θα του ορίσουμε. Για παράδειγμα μέσω των AnalogInput, DigitalInput/ DigitalOutput ορίζουμε αν το συγκεκριμένο pin είναι αναλογική ή ψηφιακή είσοδος/έξοδος.
- **readDigitalPin/ writeDigitalPin:** το αντίστοιχο pin διαβάζει ή στέλνει δεδομένα μέσω ψηφιακού σήματος. Οι τιμές που παίρνουν τα pin ορίζονται ως λογικές, επομένως είναι

0 για false και 1 για true.

- **writePWMVoltage/ writePWMDutyCycle:** δημιουργούν PWM σήμα με συγκεκριμένη τάση ή συγκεκριμένη συχνότητα σε ένα συγκεκριμένο pin.

Υπάρχουν πολλές άλλες εντολές που έχουμε την δυνατότητα να χρησιμοποιήσουμε ανάλογα με την εφαρμογή που θέλουμε να υλοποιήσουμε (σύνδεση εξαρτημάτων μέσω των πρωτόκολλων επικοινωνίας I²C ή SPI ,έλεγχος servo κτλπ).

6.3 Μέθοδος Ελέγχου Ροής Ρητίνης

6.3.1 Εισαγωγή

Για τον εμποτισμό του υφάσματος, η ρητίνη εκχύεται από δύο βαλβίδες οι οποίες ελέγχονται από τον μικροελεγκτή. Καθώς το μέτωπο της ροής της ρητίνης προχωράει, το ύφασμα εμποτίζεται. Τα εικονοστοιχεία που αντιστοιχούν στα σημεία του υφάσματος που έχουν εμποτιστεί, αλλάζουν τιμή, καθώς προστίθεται η τιμή του εικονοστοιχείου του μετώπου της ροής, στην τιμή του αντίστοιχου εικονοστοιχείου του υφάσματος. Η τιμή των εικονοστοιχείων του υφάσματος πριν τον εμποτισμό είναι ίση με -3, ενώ οι τιμές των εικονοστοιχείων των δύο μετώπων της ροής είναι 1 και 2 για το κάθε μέτωπο ή 1 για το ενιαίο μέτωπο ροής. Επομένως όταν το ύφασμα εμποτίζεται, η τιμή του εικονοστοιχείου του υφάσματος που έχει εμποτιστεί είναι ακέραια τιμή μεγαλύτερη του -3. Χρησιμοποιώντας την μέθοδο αυτήν, γίνεται αντιληπτό ποιό μέρος του υφάσματος έχει εμποτιστεί με ρητίνη. Ο πίνακας που εμπεριέχει τα εικονοστοιχεία του υφάσματος με τιμή ίση με -3 είναι η μεταβλητή labelimp.

6.3.2 Έλεγχος Παραμέτρων

Όπως έχουμε περιγράψει, η κάθε βαλβίδα είναι υπεύθυνη για τον εμποτισμό συγκεκριμένης περιοχής του υφάσματος , η οποία έχει υπολογιστεί από το υποπρόγραμμα vkalouri.m. Στην κάθε περιοχή του υφάσματος έχουν δημιουργηθεί 9 σημεία ελέγχου από το υποπρόγραμμα vkalouri.m. Τα σημεία ελέγχου είναι συγκεκριμένα εικονοστοιχεία του υφάσματος τα οποία πριν τον εμποτισμό του υφάσματος με ρητίνη έχουν την ίδια συγκεκριμένη τιμή, η οποία είναι ίση με -3. Καθώς η ρητίνη εμποτίζει το ύφασμα, αν το μέτωπο της ροής της ρητίνης έχει εμποτίσει κάποιο ή κάποια από τα σημεία ελέγχου, τότε η τιμή του σημείου ελέγχου που έχει εμποτιστεί αλλάζει τιμή.

Εκτός από τον έλεγχο των σημείων ελέγχου για αλλαγή στην τιμή τους, υπάρχουν άλλες δύο παράμετροι για τον έλεγχο των βαλβίδων:

- Η σχέση της τεταγμένης ενός από των τριών σημείων των μετώπων ροής, σε σχέση

με την τεταγμένη του σημείου MaXy.

- Το ποσοστό εμποτισμού impr του υφάσματος

6.3.2.1 Ενεργοποίηση/Απενεργοποίηση Βαλβίδων

Συνθήκες

Για τον έλεγχο της κατάστασης των βαλβίδων, δηλαδή για τον αν θα είναι ενεργοποιημένες ή απενεργοποιημένες, πρέπει να ικανοποιούνται κάποιες συγκεκριμένες συνθήκες, ώστε να δοθεί το αντίστοιχο σήμα στις βαλβίδες από το μικροελεγκτή. Για την εξέταση των συνθηκών αυτών χρησιμοποιούμε την εντολή *if/elseif* της MATLAB.

Αρχικά, εξετάζονται οι τιμές των 8 πρώτων σημείων ελέγχου για την αντίστοιχη βαλβίδα. Αν έστω και ένα σημείο ελέγχου έχει την αρχική τιμή, δηλαδή -3, η αντίστοιχη βαλβίδα είναι ενεργοποιημένη. Μεταξύ του ελέγχου των σημείων ελέγχου, χρησιμοποιούμε την λογική πράξη της διάζευξης(*or*).

Επομένως η συνθήκη για την βαλβίδα A είναι:

```
labelimp(chpointA1(2),chpointA1(1))==-3  
||labelimp(chpointA2(2),chpointA2(1))==-  
3||labelimp(chpointA3(2),chpointA3(1))==-3 ||...  
labelimp(chpointA4(2),chpointA4(1))==-  
3||labelimp(chpointA5(2),chpointA5(1))==-  
3||labelimp(chpointA6(2),chpointA6(1))==-3 ...  
labelimp(chpointA6(2),chpointA6(1))==-  
3||labelimp(chpointA7(2),chpointA7(1))==-  
3||labelimp(chpointA8(2),chpointA8(1))==-3
```

και για την βαλβίδα B:

```
labelimp(chpointB1(2),chpointB1(1))==-3  
||labelimp(chpointB2(2),chpointB2(1))==-  
3||labelimp(chpointB3(2),chpointB3(1))==-3 ||...  
labelimp(chpointB4(2),chpointB4(1))==-  
3||labelimp(chpointB5(2),chpointB5(1))==-  
3||labelimp(chpointB6(2),chpointB6(1))==-3 ...  
labelimp(chpointB6(2),chpointB6(1))==-  
3||labelimp(chpointB7(2),chpointB7(1))==-  
3||labelimp(chpointB8(2),chpointB8(1))==-3
```

Στην περίπτωση που όλα τα σημεία ελέγχου της αντίστοιχης βαλβίδας έχουν αλλάξει τιμή,

το οποίο σημαίνει ότι η ρητίνη έχει εμποτίσει τα αντίστοιχα σημεία του υφάσματος , ελέγχεται η συνθήκη του *elseif* έτσι ώστε η αντίστοιχη βαλβίδα να απενεργοποιηθεί. Αν οι συνθήκες του *elseif* ικανοποιούνται τότε η βαλβίδα θα απενεργοποιηθεί μετά από 5 επαναλήψεις, για μεγαλύτερη ασφάλεια έχοντας δημιουργήσει μια επαναληπτική διαδικασία χρησιμοποιώντας την εντολή *if*, και τις μεταβλητές *delayA* και *delayB* για την κάθε βαλβίδα οι οποίες αυξάνουν την τιμή τους κατά μία μονάδα για κάθε επανάληψη. Όταν η αντίστοιχη μεταβλητή έχει τιμή ίση με 5, τότε ικανοποιείται η συνθήκη του *if*. Για την απενεργοποίηση της αντίστοιχης βαλβίδας πρέπει να ικανοποιείται μια από τις παρακάτω δύο συνθήκες:

- το ένατο σημείο ελέγχου, *chpointA9/ chpointB9* το οποίο είναι το ακραίο σημείο ελέγχου της αντίστοιχης ημιπεριοχής του υφάσματος να έχει αλλάξει τιμή, δηλαδή να είναι διάφορη του -3 και ταυτόχρονα η τιμή της τεταγμένης ενός εκ των τριών σημείων των μετώπων ροής *p1oint1*, *p1oint3* και *p1oint2* για την πρώτη ροή ή την ενιαία, *p2oint1*, *p2oint3* και *p2oint2* για την δεύτερη ροή αντίστοιχα, να είναι μεγαλύτερη από την τεταγμένη του σημείου *MaXy*. Μεταξύ των δύο συνθηκών χρησιμοποιούμε την λογική πράξη της σύζευξης (*and*). Λόγω του ότι κατά την διάρκεια της VARI τα δύο μέτωπα της ροής της ρητίνης θα ενωθούν, και θα έχουμε ένα μέτωπο ροής, στην βαλβίδα B εκτός από τα σημεία *p2oint1*, *p2oint3* και *p2oint2*, ελέγχονται και τα σημεία *p1oint1*, *p1oint3* και *p1oint2* για την περίπτωση του ενός μετώπου ροής, καθώς όταν τα δύο μέτωπα της ροής ενωθούν, παύουν να υπολογίζονται τα σημεία *p2oint1*, *p2oint3* και *p2oint2*. Μεταξύ των σημείων για την ενιαία ροή και την δεύτερη ροή, χρησιμοποιείται η λογική πράξη της διάζευξης (*or*).
- Το ποσοστό εμποτισμού του υφάσματος *imp* να είναι 100%, δηλαδή το ύφασμα να έχει εμποτιστεί σε ολόκληρη την επιφάνειά του.

Επομένως οι συνθήκες για την βαλβίδα A είναι:

```
labelimp(chpointA9(2),chpointA9(1))~=-3 &&
(p1oint1(1)>=MaXy||p1oint2(1)>=MaXy||p1oint3(1)>=MaXy)||imp=100.00
```

και για την βαλβίδα B:

```
labelimp(chpointB9(2),chpointB9(1))~=-3 &&
(p2oint1(1)>=MaXy||p2oint2(1)>=MaXy||p2oint3(1)>=MaXy)||imp>=100.00
```

6.3.2.2 Εντολές Arduino

Για την ενεργοποίηση και την απενεργοποίηση των βαλβίδων εκτελείται η εντολή *writeDigitalPin(arduino, 'pin',value)*, όπου *arduino0* είναι το object με τα χαρακτηριστικά του Arduino, *pin* είναι ο αριθμός της εξόδου του Arduino που ελέγχει την κάθε βαλβίδα (D10

και D8 για την βαλβίδα A και την βαλβίδα B αντίστοιχα), και value η τιμή που θα πάρει το αντίστοιχο pin (0 για απενεργοποίηση-τάση 0 V και 1 για ενεργοποίηση -τάση 5V). Κάθε φορά που ενεργοποιείται η απενεργοποιείται κάποια από τις δύο βαλβίδες, εμφανίζεται το αντίστοιχο μήνυμα στην οθόνη του υπολογιστή, χρησιμοποιώντας την εντολή της MATLAB *disp 'μήνυμα'*.

6.3.2.3 Έλεγχος Προηγούμενης Κατάστασης Βαλβίδας

Κάθε φορά που ικανοποιούνται οι κατάλληλες συνθήκες για την ενεργοποίηση ή την απενεργοποίηση της αντίστοιχης βαλβίδας, πριν δοθεί η ανάλογη εντολή από το Arduino, ελέγχεται η προηγούμενη κατάσταση της αντίστοιχης βαλβίδας. Επομένως αν η βαλβίδα είναι ήδη ενεργοποιημένη δεν θα εκτελεστεί η εντολή για την ενεργοποίησή της, αντίστοιχα όταν η βαλβίδα είναι ήδη απενεργοποιημένη, δεν θα εκτελεστεί η εντολή για την απενεργοποίησή της. Ο λόγος για τον οποίο γίνεται ο συγκεκριμένος έλεγχος είναι ότι σε δοκιμές που έγιναν κατά την διάρκεια της κατεργασίας, παρατηρήθηκε με την βοήθεια του profile της MATLAB πως ο συνολικός χρόνος εκτέλεσης της εντολής writeDigitalPin, είναι αρκετά μεγάλος με αποτέλεσμα το πρόγραμμα να καθυστερεί. Το αποτέλεσμα της καθυστέρησης αυτής ήταν το μέτωπο της ροής να έχει προχωρήσει αρκετά, χωρίς να έχει επεξεργαστεί από την MATLAB το αντίστοιχο frame, κάτι το οποίο θα δημιουργούσε αβεβαιότητα ως προς την απόκριση της βαλβίδας ανάλογα με τη θέση του μετώπου της ροής της ρητίνης.

Για την πραγματοποίηση της παραπάνω λειτουργίας έχουν δημιουργηθεί δύο μεταβλητές για την κάθε βαλβίδα: A και preValueA για την βαλβίδα A, B και preValueB για την βαλβίδα B. Οι μεταβλητές A και B, όταν ενεργοποιείται η αντίστοιχη βαλβίδα παίρνουν τιμή ίση με 1, και όταν απενεργοποιηθεί η αντίστοιχη βαλβίδα, παίρνουν τιμή ίση με 0. Οι μεταβλητές preValueA και preValueB, έχουν ως τιμή την αντίστοιχη τιμή των A και B της προηγούμενης επανάληψης. Στο τέλος κάθε επανάληψης αλλάζει η τιμή των preValueA και preValueB με τις αντίστοιχες τιμές των A και B.

6.3.3 Έλεγχος Περίπτωσης μη Επιτυχούς Κενού ή έμφραξης Θύρας Αναρρόφησης

6.3.3.1 Εισαγωγή

Κατά τη διάρκεια της κατεργασίας VARI, υπάρχει περίπτωση το μέτωπο ροής της ρητίνης να έχει μείνει στάσιμο, με αποτέλεσμα να μην εξελίσσεται ο εμποτισμός του υφάσματος όπως πρέπει. Οι λόγοι που μπορεί να συμβαίνει αυτό εκτός από τον προφανέστατο να είναι

κλειστές οι βαλβίδες, είναι οι εξής δύο:

1. Υπάρχει περίπτωση να έχει μειωθεί το κενό που είχε δημιουργηθεί στην σακούλα κενού με αποτέλεσμα η διαφορά πίεσης της εισόδου της ρητίνης και της εισόδου αναρρόφησης να μην είναι αρκετή για την έκχυση της ρητίνης. Οι πιθανότερες αιτίες για την μείωση του κενού είναι:
 - Να έχει σκιστεί η σακούλα κενού
 - Η σφραγιστική ταινία να μην κάνει σωστή επαφή με το καλούπι
 - Η ένωση του σωλήνα αναρρόφησης με την αντλία κενού ή με τον παγιδευτή ρητίνης που παρεμβάλλεται μεταξύ του καλουπιού και της αντλίας κενού να μην είναι επιτυχής
 - Να έχει εμφανιστεί τεχνικό πρόβλημα στην αντλία κενού.
2. Υπάρχει περίπτωση η είσοδος αναρρόφησης ρητίνης να έχει φραχθεί από την σακούλα κενού, με αποτέλεσμα η ρητίνη να μένει στάσιμη.

6.3.3.2 Μέθοδος Εντοπισμού Στάσιμου Μετώπου

Στην περίπτωση που λόγω προβληματικού κενού ή λόγω φραγής της εισόδου αναρρόφησης, το μέτωπο της ρητίνης έχει μείνει στάσιμο, η τιμή του ποσοστού εμποτισμού ρ_1 καθώς και οι συντεταγμένες των εικονοστοιχείων του μετώπου ροής δεν αλλάζουν. Εάν οι τιμές αυτές έχουν μείνει ίδιες τρεις συνεχόμενες φορές κατά την διάρκεια της κατεργασίας, τότε συμπεραίνουμε ότι το μέτωπο της ροής της ρητίνης έχει μείνει στάσιμο, και ενεργοποιείται το κόκκινο led στο κύκλωμα του μικροελεγκτή για να μας προειδοποιήσει. Το εικονοστοιχείο που χρησιμοποιείται για την σύγκριση, είναι σημείο ελέγχου $p1_{point2}$ του πρώτου ή του ενιαίου μετώπου της ροής και το σημείο ελέγχου $p2_{point2}$ για το μέτωπο της δεύτερης ροής της ρητίνης. Τα σημεία ελέγχου μετώπου $p1_{point2}$ και $p2_{point2}$ έχουν υπολογιστεί στο υπο-πρόγραμμα `vgrammh2roes.m`.

Για την πραγματοποίηση της παραπάνω μεθόδου στη MATLAB, αρχικά εκτελείται η εντολή `if/elseif` της MATLAB εξετάζοντας αν έχουμε δύο μέτωπα ροής ή ένα. Αν έχουμε δύο μέτωπα ροής, έχουμε και δύο σημεία ελέγχου μετώπων. Η μεταβλητή που εξετάζεται είναι η `ccc.NumObjects`, επομένως αν η `ccc.NumObjects==2` είναι αληθής, έχουμε δύο μέτωπα ροής, ενώ αν `ccc.NumObjects==1` είναι αληθής έχουμε ένα μέτωπο ροής. Στη συνέχεια χρησιμοποιώντας εμβόλιμη εντολή `if` στο `if/elseif`, ελέγχεται αν οι τρέχουσες συντεταγμένες των σημείων ελέγχου των μετώπων $p1_{point2}$ και $p2_{point2}$ για δύο μέτωπα ροής ή $p1_{point2}$ για το ένα μέτωπο ροής είναι ίδιες με τις συντεταγμένες των σημείων ελέγχου των μετώπων της δεύτερης κατά σειρά προηγούμενης επανάληψης, `previous3p1point2` και

previous3p2oint2 αντίστοιχα για δύο μέτωπα ροής ή previous3p1oint2 για ένα μέτωπο ροής. Επίσης ελέγχεται αν η τρέχουσα τιμή του ποσοστού εμποτισμού imp είναι ίδια με την τιμή του ποσοστού εμποτισμού της δεύτερης κατά σειρά προηγούμενης επανάληψης previous3imp. Αν ταυτόχρονα οι δύο αυτές συνθήκες είναι αληθείς τότε το μέτωπο της ροής της ρητίνης είναι στάσιμο. Να σημειωθεί πως για να είναι αληθής η κάθε συνθήκη πρέπει οι βαλβίδες να είναι ενεργοποιημένες.

Επομένως οι συνθήκες που εξετάζονται στη MATLAB για δύο μέτωπα ροής είναι:

```
imp==previous3imp && (A==1||B==1) &&  
(isequal(p1oint2,previous3p1oint2) && A==1) &&  
(isequal(p2oint2,previous3p2oint2) && B==1)
```

για το ένα μέτωπο ροής:

```
imp==previous3imp && (A==1||B==1) &&  
(isequal(p1oint2,previous3p1oint2) && ( A==1||B==1))
```

6.3.3.3 Προειδοποίηση μη Επιτυχούς Κενού ή Σφράγισης Θύρας Αναρρόφησης

Στην περίπτωση που το μέτωπο της ροής έχει μείνει στάσιμο για τρία συνεχόμενα στη σειρά frames, τότε ενεργοποιείται ένα κόκκινο Led στο κύκλωμα που έχει κατασκευαστεί. Στην περίπτωση που το μέτωπο της ροής δεν είναι στάσιμο πια τότε απενεργοποιείται. Για την ενεργοποίηση και την απενεργοποίηση του Led εκτελείται η εντολή `writeDigitalPin(arduino0,'pin',value)`, όπου `arduino0` είναι το object με τα χαρακτηριστικά του Arduino, `pin` είναι ο αριθμός της εξόδου του Arduino που ελέγχει το Led, στη συγκεκριμένη περίπτωση το D11 και `value` η τιμή που θα πάρει το αντίστοιχο pin, 0 για απενεργοποίηση και 1 για ενεργοποίηση.

6.3.3.4 Έλεγχος Προηγούμενης Κατάστασης Led

Όπως έγινε και για τον έλεγχο των δύο βαλβίδων για την μείωση του υπολογιστικού κόστους του προγράμματος, πριν δοθεί η ανάλογη εντολή από το Arduino, ελέγχεται η προηγούμενη κατάσταση του Led. Επομένως αν το Led είναι ήδη ενεργοποιημένο δεν θα εκτελεστεί η εντολή για την ενεργοποίησή του, αντίστοιχα όταν το Led είναι ήδη απενεργοποιημένο, δεν θα εκτελεστεί η εντολή για την απενεργοποίησή του. Για την πραγματοποίηση της παραπάνω λειτουργίας έχει δημιουργηθεί μια μεταβλητή η `prevalueD11`.

6.3.4 Δήλωση Μεταβλητών Τύπου persistent

Ο κώδικας που πραγματοποιεί τις παραπάνω λειτουργίες, είναι γραμμένος σε υπορουτίνα

της MATLAB, επομένως οι μεταβλητές δεν κρατάνε την παλιά τους τιμή. Για τον λόγο αυτόν όπως κάναμε και στο υποπρόγραμμα `nrho.m`, οι μεταβλητές `previousimp`, `previous3imp`, `previous3p1oint2`, `previousp1oint2`, `previous3p2oint2`, `previousp2oint2`, `preValueA`, `preValueB`, `A`, `B` και `prevalueD11` δηλώνονται ως τύπου `persistent`. Οι αρχικές τιμές των παραπάνω μεταβλητών τύπου `persistent` θα οριστούν χρησιμοποιώντας την εντολή της `if` ελέγχοντας την συνθήκη `isempty()`, όπου μέσα στις παρενθέσεις βάζουμε την αντίστοιχη μεταβλητή, έτσι αν είναι κενή η μεταβλητή παίρνει την αρχική της τιμή. Επομένως έχουμε αντίστοιχα:

```
if isempty(A) || isempty(B) A=0; B=0; end
if isempty( prevalueD11) prevalueD11=0; end
if isempty(preValueA) || isempty(preValueB)
    preValueA=A; preValueB=B; end
if isempty(previous3imp) previous3imp=inf; end
if isempty(previousimp) previousimp=inf; end
if isempty(previous3p1oint2) previous3p1oint2=inf; end
if isempty(previousp1oint2) previousp1oint2=inf; end

if isempty(previous3p2oint2) previous3p2oint2=inf; end
if isempty(previousp2oint2) previousp2oint2=inf; end6.4
```

6.4 Κώδικας MATLAB

Ο κώδικας που έχει γραφτεί στη MATLAB για την υλοποίηση των παραπάνω είναι το υποπρόγραμμα `arduinno.m`. Οι μεταβλητές που εισάγονται είναι: η μεταβλητή `arduin0` που είναι το `object` με τα χαρακτηριστικά του Arduino, τα σημεία ελέγχου `p1oint1`, `p1oint3` και `p1oint2` για το πρώτο ή το ενιαίο μέτωπο ροής, `p2oint1`, `p2oint3` και `p2oint2` για το δεύτερο μέτωπο ροής, ο πίνακας δομής `ccc`, το ποσοστό εμποτισμού του υφάσματος `imp`, ο πίνακας με τα εικονοστοιχεία του υφάσματος `labelimp`, το σημείο `MaXy`, τα σημεία ελέγχου του υφάσματος προς εμποτισμό, `chpointA1`, `chpointA2`, `chpointA3`, `chpointA4`, `chpointA5`, `chpointA6`, `chpointA7`, `chpointA8`, `chpointA9`, `chpointB1`, `chpointB2`, `chpointB3`, `chpointB4`, `chpointB5`, `chpointB6`, `chpointB7`, `chpointB8`, `chpointB9`.

```
%% arduino

function arduinno(arduin0, p1oint1,p1oint3,p1oint2, ...
    p1oint1,p1oint3,p1oint2,ccc,imp,labelimp,MaXy,chpointA1, ...
    chpointA2,chpointA3,chpointA4,chpointA5,chpointA6, ...
    chpointA7,chpointA8,chpointA9,chpointB1,chpointB2, ...
    chpointB3,chpointB4,chpointB5,chpointB6,chpointB7, ...
    chpointB8,chpointB9)

persistent previousimp previous3imp previous3p1oint2 previousp1oint2
previous3p2oint2 previousp2oint2 preValueA preValueB A B ...
prevalueD11 delayA delayB

if isempty(A) || isempty(B) A=0; B=0; end
if isempty( prevalueD11) prevalueD11=0; end
```

```

if isempty(preValueA) || isempty(preValueB)
    preValueA=A;
    preValueB=B;
end
if isempty(preValueA) || isempty(preValueB)
    preValueA=A;
    preValueB=B;
end

% ----->  έλεγχος Βαλβίδας Α <-----

if                                labelimp(chpointA1(2),chpointA1(1))==-3
||labelimp(chpointA2(2),chpointA2(1))==-
3||labelimp(chpointA3(2),chpointA3(1))==-3 ||...
    labelimp(chpointA4(2),chpointA4(1))==-
3||labelimp(chpointA5(2),chpointA5(1))==-
3||labelimp(chpointA6(2),chpointA6(1))==-3 ...
    labelimp(chpointA6(2),chpointA6(1))==-
3||labelimp(chpointA7(2),chpointA7(1))==-
3||labelimp(chpointA8(2),chpointA8(1))==-3;
    if preValueA==0
writeDigitalPin(arduino,'D10',1)
A=1;
disp('Valve A is -on-')
preValueA =A;
    end
elseif                                labelimp(chpointA9(2),chpointA9(1))~=-3 ...
&&(p1oint1(1)>=MaXy||p1oint2(1)>=MaXy||p1oint3(1)>=MaXy)||imp>=100.00

    if preValueA==1
        delayA=delayA+1
    if delayA==5
        writeDigitalPin(arduino,'D10',0)
        A=0;
        disp('Valve A is -OFF-')
        preValueA =A;
        delayA=0
    end
end

%----->  έλεγχος Βαλβίδας Β <-----

If                                labelimp(chpointB1(2),chpointB1(1))==-3
||labelimp(chpointB2(2),chpointB2(1))==-
3||labelimp(chpointB3(2),chpointB3(1))==-3 ||...
    labelimp(chpointB4(2),chpointB4(1))==-
3||labelimp(chpointB5(2),chpointB5(1))==-
3||labelimp(chpointB6(2),chpointB6(1))==-3 ...
    labelimp(chpointB6(2),chpointB6(1))==-
3||labelimp(chpointB7(2),chpointB7(1))==-
3||labelimp(chpointB8(2),chpointB8(1))==-3;
if preValueB==0
writeDigitalPin(arduino,'D8',1)
B=1;
disp('Valve B is -on-')
preValueB=B;
end
elseif                                labelimp(chpointB9(2),chpointB9(1))~=-3    &&    ...
(p2oint1(1)>=MaXy||p2oint2(1)>=MaXy||p2oint3(1)>=MaXy)||...
p1oint1(1)>=MaXy||p1oint2(1)>=MaXy||p1oint3(1)>=MaXy)||...

```

```

imp>=100.00

if preValueB==1
    delayB=delayB+1
    if delayB==5
        writeDigitalPin(arduino,'D8',0)
        B=0;
        disp('Valve B is - OFF -')
        preValueB=B;
        delayB=0
    end
end

%% ----- περίπτωση μη επιτυχούς κενού -----
% συγκρίνει την υπάρχουσα τιμή του ποσοστού εμποτισμού, και %ενός
σημείου του μετώπου της ρητίνης, με τις προ προηγούμενες %τιμές.

% Αρχικές τιμές
if isempty(previous3imp) previous3imp=inf; end
if isempty(previousimp) previousimp=inf; end
if isempty(previous3p1oint2) previous3p1oint2=inf; end
if isempty(previousp1oint2) previousp1oint2=inf; end
if isempty(previous3p2oint2) previous3p2oint2=inf; end
if isempty(previousp2oint2) previousp2oint2=inf; end

if ccc.NumObjects==2

    if      imp==previous3imp      &&      (A==1||B==1)      &&
(isequal(p1oint2,previous3p1oint2)      &&      A==1) &&
(isequal(p2oint2,previous3p2oint2) && B==1)
        if prevalueD11==0
            writeDigitalPin(arduino,'D11',1);
            prevalueD11=1;
        end
    else
        if prevalueD11==1
            writeDigitalPin(arduino,'D11',0);
            prevalue=0;
        end
    end

elseif ccc.NumObjects==1

    if      imp==previous3imp      &&      (A==1||B==1)      &&
(isequal(p1oint2,previous3p1oint2) &&( A==1||B==1))
        if prevalueD11==0
            writeDigitalPin(arduino,'D11',1);
            prevalueD11=1;
        end
    else
        if prevalueD11==1
            writeDigitalPin(arduino,'D11',0);
            prevalueD11=0;
        end
    end

end
previous3imp=previousimp;
previousimp=imp;

```

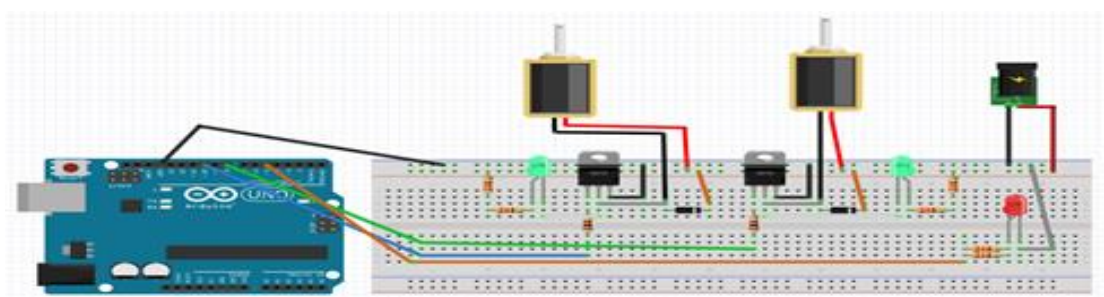


```
previous3point2=previouspoint2;  
previouspoint2=point2;  
previous3p2oint2=previousp2oint2;  
previousp2oint2=p2oint2;  
end
```

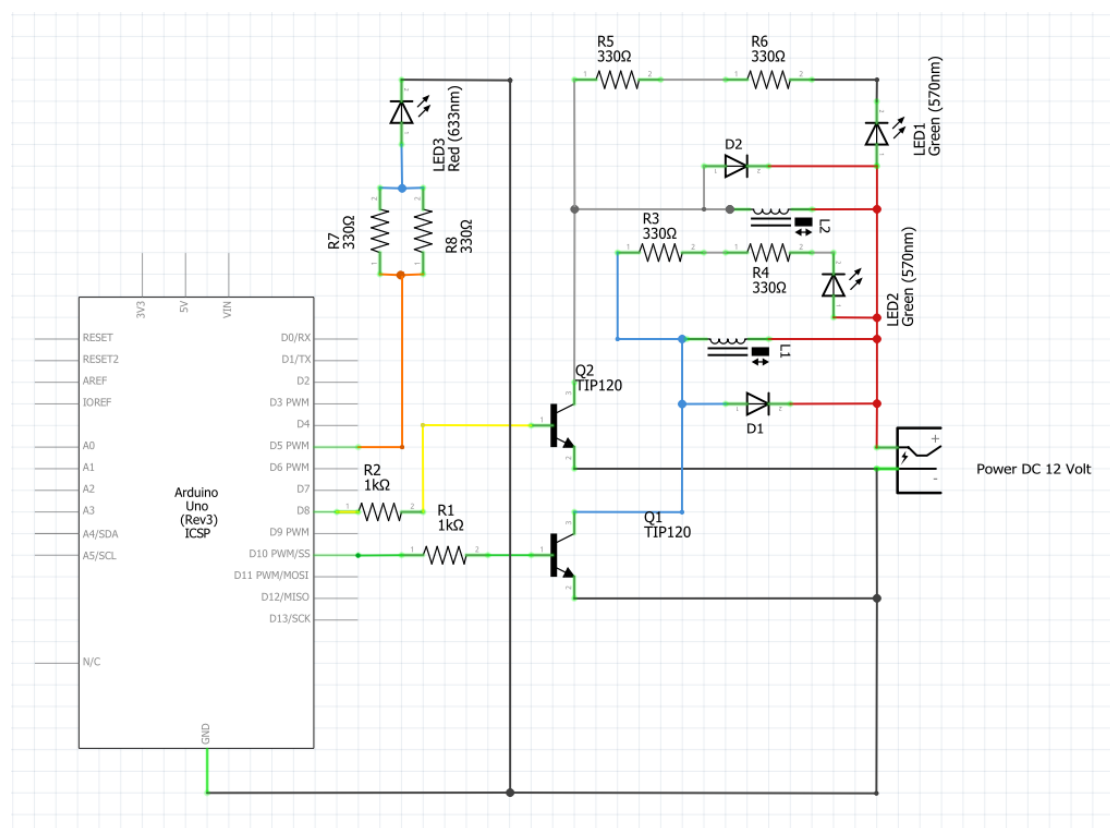
ΚΕΦΑΛΑΙΟ 7 Ηλεκτρονικό Κύκλωμα Ελέγχου των Βαλβίδων

7.1 Εισαγωγή

Ο έλεγχος των δύο ηλεκτροβαλβίδων γίνεται μέσω του μικροεπεξεργαστή Arduino Uno. Το ηλεκτρονικό κύκλωμα το οποίο κατασκευάστηκε αποτελείται από: έναν μετασχηματιστή AC-DC, δύο darlington ηρη τρανζίστορ TIP 120, δύο διόδους 1N4004, δύο αντιστάσεις των 1 ΚΩ, 4 αντιστάσεις των 330 Ω, τρία LED και αγωγούς, για την σύνδεση των στοιχείων μεταξύ τους και με τον μικροεπεξεργαστή Arduino Uno.



Εικόνα 50: Ηλεκτρονικό κύκλωμα



Εικόνα 51: Ηλεκτρονικό σχέδιο κυκλώματος για τον έλεγχο των ηλεκτροβαλβίδων

7.2 Στοιχεία Ηλεκτρονικού Κυκλώματος

Μετασχηματιστής AC-DC

Η μέγιστη τάση που μπορεί να τροφοδοτήσει το Arduino τα εξαρτήματα που είναι συνδεδεμένα σε αυτό, είναι 5 Volts. Επιπλέον η μέγιστη ένταση ρεύματος που μπορεί να παρέχει σε αυτά με ασφάλεια η κάθε έξοδος του Arduino είναι 20 mA, με μέγιστη συνολική ένταση ρεύματος όλων των εξόδων 150 mA [20]. Συνεπώς, χρησιμοποιούμε εξωτερική τροφοδοσία για την σύνδεση των εξαρτημάτων που απαιτούν μεγαλύτερη τάση καθώς και μεγαλύτερη ένταση ρεύματος.

Ο μετασχηματιστής που χρησιμοποιούμε στην παρούσα εργασία μετασχηματίζει την εναλλασσόμενη τάση των 220 V του ηλεκτρικού δικτύου σε σταθερή τάση των 12 V. Ο μετασχηματιστής είναι απαραίτητος για την λειτουργία των ηλεκτροβαλβίδων που χρησιμοποιούμε στο υδραυλικό-πνευματικό σύστημα για την έγχυση της ρητίνης, καθώς τα πηνία των ηλεκτροβαλβίδων λειτουργούν με σταθερή τάση 12 V και το ρεύμα που καταναλώνουν είναι 250mA.

Τρανζίστορ BJT ηρη

Τα τρανζίστορ χρησιμοποιούνται σε πολλές εφαρμογές στα ηλεκτρικά κυκλώματα. Τα είδη των τρανζίστορ είναι : το διπολικό τρανζίστορ επαφής – BJT, το τρανζίστορ επίδρασης πεδίου-FET, και το τρανζίστορ επαφής –JFET. Οι συχνότερες εφαρμογές των τρανζίστορ στα ηλεκτρικά κυκλώματα, είναι ως ενισχυτές σήματος ή ως διακόπτες.

Στο ηλεκτρικό κύκλωμα που έχουμε σχεδιάσει, χρησιμοποιούμε το τρανζίστορ, ως διακόπτη, για τον έλεγχο των ηλεκτροβαλβίδων. Το τρανζίστορ που έχουμε χρησιμοποιήσει είναι το TIP 120, το οποίο είναι Darlington τύπου ηρη. Τα τρανζίστορ Darlington λειτουργούν ως BJT τρανζίστορ. Αποτελούνται από δύο ηρη ή ηρη BJT τρανζίστορ (στην περίπτωση μας ηρη), τα οποία είναι συνδεδεμένα μεταξύ τους, έτσι ώστε το ρεύμα του εκπομπού του πρώτου BJT να είναι το ρεύμα εισόδου της βάσης του δεύτερου BJT τρανζίστορ. Η διαφορά ενός τρανζίστορ Darlington και ενός τρανζίστορ BJT είναι ότι λόγω της διάταξης του, το κέρδος της έντασης του ρεύματος (h_{FE}), είναι 1000, όταν στα απλά BJT τρανζίστορ είναι από 50 έως 200 κατά μέγιστο. Αυτό σημαίνει ότι το ρεύμα συλλέκτη εκπομπού μπορεί να είναι 1000 φορές μεγαλύτερο από το ρεύμα της βάσης, έτσι με ένα σήμα πολύ μικρής έντασης μπορούμε να ελέγξουμε μεγάλες ποσότητες ρεύματος. Το μειονέκτημά του όμως είναι ότι η τάση κατωφλιού, δηλαδή η τάση βάσης- εκπομπού V_{BE} , όπου πάνω από αυτή βρισκόμαστε στη περιοχή κορεσμού και διαρρέει ρεύμα από τον συλλέκτη στον εκπομπό, είναι διπλάσια από αυτή του BJT τρανζίστορ δηλαδή, $0,7 \times 2 = 1,4V$,

αφού έχουμε πτώση τάσης στη δίοδο βάσης εκπομπού της τάξεως των 0,7V στο κάθε ένα ξεχωριστό BJT τρανζίστορ.

Η σύνδεση που έχουμε κάνει είναι τύπου low-side switch [21], δηλαδή, το τρανζίστορ TIP120 είναι στην πλευρά της γείωσης. Έχουμε συνδέσει την βάση (B) με μια έξοδο του Arduino, έτσι ώστε να λαμβάνει το σήμα για την ενεργοποίηση της ηλεκτροβαλβίδας, τον συλλέκτη (C) με την γείωση του ηλεκτρικού κυκλώματος και τον εκπομπό (E) με τον ουδέτερο του πηνίου της ηλεκτροβαλβίδας. Στο πηνίο δεν παίζει σημαντικό ρόλο ποια επαφή θα συνδέσουμε στη γείωση, και ποια στον θετικό πόλο.

Δίοδος Ανόρθωσης

Οι δίοδοι είναι ημιαγωγά στοιχεία που αποτελούνται από ημιαγωγά υλικά τύπου n και p. Το υλικό τύπου n παράγει τα ελεύθερα ηλεκτρόνια, ενώ το υλικό τύπου p είναι οι οπές όπου συλλέγει τα ελεύθερα ηλεκτρόνια. Το χαρακτηριστικό της διόδου, είναι ότι άγει μόνο από την μία φορά, με άλλα λόγια συμπεριφέρεται ως αγωγός ή ως μονωτής ανάλογα με την φορά της τάσης.

Οι τύποι των διόδων είναι : οι ημιαγωγοί δίοδοι, οι δίοδοι ανόρθωσης οι οποίες επιτρέπουν μεγαλύτερη ένταση ρεύματος σε σχέση με τις δίοδους ημιαγωγών, οι δίοδοι εκπομπής φωτός, τα γνωστά μας LED, στις οποίες όπως και στις δίοδους ημιαγωγών, το ρεύμα διαρρέει μόνο από την μία κατεύθυνση, αλλά εκπέμπουν και φως, οι schottky δίοδοι, οι οποίες αντί για ημιαγωγό τύπου P χρησιμοποιεί μέταλλο για την σύνδεση με τον ημιαγωγό τύπου N, με αποτέλεσμα την γρήγορη μετάβαση από ορθή σε ανάστροφη πόλωση και το αντίθετο, οι δίοδοι Zener όπου λειτουργούν και στην περιοχή διάσπασης χωρίς να αυτοκαταστραφούν, και τέλος οι φωτοδίοδοι οι οποίες παράγουν ηλεκτρικό ρεύμα από την ενέργεια των φωτονίων. Οι δίοδοι χρησιμοποιούνται στα περισσότερα ηλεκτρικά κυκλώματα. Χρησιμοποιούνται ως ανορθωτές για την ελεγχόμενη διέλευση του ρεύματος στα αναλογικά ηλεκτρονικά κυκλώματα, ως ψαλιδιστές σημάτων στα ψηφιακά ηλεκτρονικά κυκλώματα, ως πολλαπλασιαστές της τάσης για την ενίσχυση της ισχύος και τέλος για την προστασία των ηλεκτρικών κυκλωμάτων είτε σε περίπτωση διέλευσης ρεύματος αντίθετης φοράς από αυτή της κανονικής φοράς, είτε λόγω δημιουργίας ξαφνικής τάσης μεγάλου πλάτους. Οι δίοδοι που χρησιμοποιούνται για την προστασία του ηλεκτρικού κυκλώματος από μεγάλες τάσεις αντίθετης πολικότητας, ονομάζονται Flyback ή Snubber δίοδοι [22].

Στο ηλεκτρικό κύκλωμα που έχουμε σχεδιάσει χρησιμοποιούμε την 1N4004 δίοδο ανόρθωσης για την προστασία του ηλεκτρικού κυκλώματος στην περίπτωση δημιουργίας τάσης μεγάλου πλάτους αντίθετης πολικότητας, που θα είχε ως αποτέλεσμα την πιθανή

καταστροφή του τροφοδοτικού, στοιχείων του ηλεκτρικού κυκλώματος όπως τα τρανζίστορ που η πολικότητα τους έχει μεγάλη σημασία, καθώς και πιθανότητα καταστροφής του Arduino.

Ο λόγος που μπορεί να δημιουργηθεί τάση αντίθετης φοράς μεγάλου πλάτους, είναι ότι στο ηλεκτρικό κύκλωμα έχουμε συνδέσει τα πηνία των ηλεκτροβαλβίδων. Όταν το πηνίο σταματήσει να διαρρέεται από ρεύμα, γνωρίζουμε από την θεωρία του ηλεκτρομαγνητισμού και τον κανόνα του Lenz ότι στο πηνίο θα δημιουργηθεί τάση λόγω αυτεπαγωγής, με φορά η οποία θα αντιτίθεται στο αίτιο που την προκάλεσε, έτσι ώστε να εμποδίσει το ρεύμα να μηδενιστεί. Με την διόδο στο κύκλωμα, το ρεύμα που δημιουργείται από το πηνίο λόγω αυτεπαγωγής, διοχετεύεται σε αυτή, χωρίς να προκαλείται ζημιά σε άλλα εξαρτήματα του ηλεκτρικού κυκλώματος.

Η διόδος συνδέεται έτσι ώστε η άνοδος της να είναι συνδεδεμένη με τον αρνητικό πόλο του πηνίου και η κάθοδος της με τον θετικό του πόλο.

Σημειωτέο ότι χρησιμοποιούμε διόδο ανόρθωσης και όχι διόδο ημιαγωγών καθώς μπορεί να λειτουργήσει σε ρεύματα αρκετά μεγάλης έντασης, όπως έχει αναφερθεί.

Ωμικές Αντιστάσεις

Στο ηλεκτρικό κύκλωμα που έχουμε κατασκευάσει, χρησιμοποιούμε αντιστάσεις για την σύνδεση του τρανζίστορ TIP120 με το Arduino, και για την σύνδεση των Led στην τροφοδοσία.

Στο τρανζίστορ παρεμβάλουμε την αντίσταση μεταξύ της βάσης και της εξόδου του Arduino. Ο λόγος που τοποθετούμε την αντίσταση αυτή, είναι για την προστασία του ίδιου του τρανζίστορ από την διέλευση μεγάλης έντασης ρεύματος, που θα είχε ως αποτέλεσμα την καταστροφή του. Επίσης η αντίσταση αυτή προστατεύει και το Arduino καθώς υπάρχει περιορισμός στο μέγιστο ρεύμα που μπορεί να παρέχει η κάθε έξοδος (40mA)[20].

Έτσι, αν, για κάποιον λόγο, οι απαιτήσεις του ηλεκτρικού κυκλώματος για ρεύμα είναι αυξημένες, ή σε περίπτωση βραχυκυκλώματος, το Arduino δεν θα μπορέσει να ανταποκριθεί και οι πιθανότητες καταστροφής του μικροεπεξεργαστή ATmega του Arduino είναι πολύ μεγάλες. Αλλά ακόμα και αν δεν καταστραφεί, στην περίπτωση που χρησιμοποιούμε περισσότερες από μια εξόδους στο Arduino, όλο το ρεύμα θα διοχετεύεται σε μια έξοδο, με αποτέλεσμα τα άλλα εξαρτήματα που είναι συνδεδεμένα στις άλλες εξόδους να μην λειτουργούν.

Το ρεύμα που διαρρέει το πηνίο της ηλεκτροβαλβίδας όταν βρίσκεται σε διέγερση είναι 250

mA, άρα το ρεύμα του συλλέκτη που έχουμε είναι, $I_C=250\text{mA}$. Από το datasheet του τρανζίστορ βλέπουμε ότι το κέρδος h_{FE} του τρανζίστορ TIP120 σε κορεσμό (saturation) είναι: $h_{FE}(\text{sat})= 250$, έτσι το ρεύμα της βάσης που χρειάζεται ώστε να έχουμε ρεύμα συλλέκτη $I_C=250\text{mA}$, είναι:

$$I_B = \frac{I_C}{250} = \frac{0.250}{250} = 0.001\text{A} \quad (7.1)$$

Η τάση βάσης-εκπομπού V_{BE} όταν διαρρέεται από ρεύμα είναι: $V_{BE}=1.4\text{V}$. Η αντίσταση δίνεται ως: $R_{base} = \frac{V_{OH}-V_{BE}}{I'_B}$ (7.2)

όπου V_{OH} η τάση της εξόδου του Arduino, $I'_B = 3I_B$, (έχουμε πολλαπλασιάσει το I_B με συντελεστή ασφαλείας, έτσι ώστε να είμαστε σίγουρα στη περιοχή κορεσμού. Επομένως έχουμε:

$$R_{base} = \frac{V_{OH}-V_{BE}}{I'_B} = \frac{5-1.4}{0.003} = 1200 \Omega \quad (7.2\alpha)$$

Το μέγιστο ρεύμα που μπορεί να δώσει μια έξοδος του Arduino μέσα σε ασφαλή όρια όπως έχουμε αναφέρει είναι 20 mA. Άρα η ελάχιστη αντίσταση που χρειαζόμαστε για την προστασία του Arduino είναι:

$$R_{min} = \frac{V_{OH}}{I_{max}} = \frac{5}{0.02} = 250 \Omega \quad (7.3)$$

Επομένως η αντίσταση που θα επιλέξουμε πρέπει να είναι:

$$250 \Omega < R < 1200 \Omega \quad (7.4)$$

Στο ηλεκτρικό κύκλωμα που έχουμε σχεδιάσει επιλέχθηκε αντίσταση 1 kΩ.

Για να αποφευχθεί η καταστροφή των LED λόγω μεγάλου ρεύματος πρέπει να τοποθετηθεί σε σειρά μαζί με το LED και μια αντίσταση. Βλέπουμε από το datasheet του LED, ότι το μέγιστο ορθό ρεύμα I_F είναι $I_F = 20\text{mA}$ και το δυναμικό παρεμπόδισης $V_{LED} = 2.2\text{V}$. Επομένως η ελάχιστη αντίσταση που χρειάζεται για την αποφυγή καταστροφής του LED βρίσκεται από τον νόμο του Ohm:

$$R_S = \frac{V_{in}-V_{LED}}{I_F} \quad (7.4)$$

Για τα LED που είναι συνδεδεμένα με την εξωτερική τάση των 12 V έχουμε:

$$R_{S12} = \frac{V_{in}-V_{LED}}{I_F} = \frac{12-2.2}{0.02} = 490\Omega \quad (7.4\alpha)$$

ενώ για αυτά που είναι συνδεδεμένα με την τάση του Arduino (5V) έχουμε:

$$R_{S5} = \frac{V_{in}-V_{LED}}{I_F} = \frac{5-2.2}{0.02} = 140\Omega \quad (7.4\beta)$$

Στο ηλεκτρικό κύκλωμα που έχουμε σχεδιάσει, για τα LED σύνδεσης στα 12 V έχουμε βάλει δύο αντιστάσεις των 330 Ω σε σειρά, ώστε να έχουμε συνολική αντίσταση 660 Ω η οποία είναι αρκετή για την προστασία του. Για τα LED σύνδεσης στα 5V έχουμε τοποθετήσει δύο αντιστάσεις 330Ω σε παράλληλη σύνδεση, έτσι η συνολική αντίσταση είναι 165Ω.

LED

Με ορθή πόλωση, τα ελεύθερα ηλεκτρόνια και οι οπές που είναι κοντά στην επαφή των δύο υλικών τύπου p και n, αλληλοεξουδετερώνονται, με αποτέλεσμα καθώς τα ελεύθερα ηλεκτρόνια πέφτουν από τα υψηλότερα ενεργειακά επίπεδα σε χαμηλότερα εκπέμπονται φωτόνια. Ανάλογα με τα υλικά που χρησιμοποιούνται αλλάζουν τα ενεργειακά επίπεδα, επομένως και το φάσμα του φωτός που εκπέμπεται.

Στο ηλεκτρικό κύκλωμα που έχουμε σχεδιάσει, έχουμε τοποθετήσει δύο LED πράσινου φωτός τα οποία εκπέμπουν όταν είναι ενεργοποιημένες οι ηλεκτροβαλβίδες, και ένα LED κόκκινου φωτός το οποίο εκπέμπει όταν υπάρχει σφάλμα στο σύστημά μας.

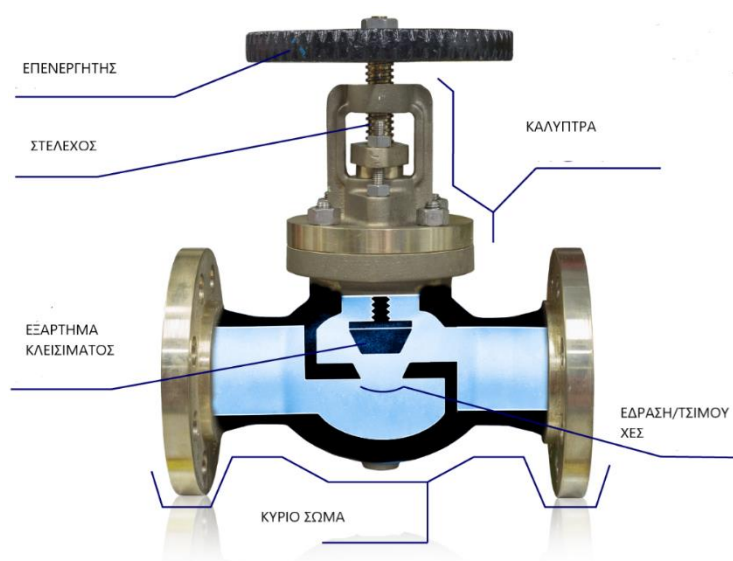
ΚΕΦΑΛΑΙΟ 8 Υδραυλικό – Πνευματικό Κύκλωμα για Έλεγχο Έγχυσης Ρητίνης

8.1 Γενική Περιγραφή Βαλβίδων

Για τον έλεγχο της ρητίνης στην διαδικασία της VARI χρησιμοποιούμε βαλβίδες για την κάθε είσοδο. Αυτές, καθώς και το υδραυλικό –πνευματικό σύστημα για τον έλεγχο της ρητίνης θα περιγραφούν παρακάτω, αφού πρώτα κάνουμε μια εισαγωγή στις βαλβίδες.

Οι βαλβίδες είναι μηχανικές διατάξεις που ρυθμίζουν την ροή υγρών, κοκκωδών υλικών ή αερίων, ανοίγοντας, κλείνοντας μερικώς ή ολικώς μια δίοδο.

Οι κυριότερες εφαρμογές βαλβίδων είναι: Ρύθμιση παροχής του ρευστού, Ρύθμιση της πίεσης του ρευστού, Έλεγχος της κατεύθυνσης της ροής, Προστασία υδραυλικού/πνευματικού κυκλώματος από την υψηλή πίεση του ρευστού, Προστασία υδραυλικού/πνευματικού συστήματος από αντίστροφη ροή του ρευστού (ανεπίστροφες βαλβίδες). Μια βαλβίδα αποτελείται από πολλά μέρη, όπως φαίνεται στην παρακάτω εικόνα.



Εικόνα 52: Μέρη βαλβίδας (plug valve)

Τα κυριότερα μέρη είναι :

Σώμα ή κέλυφος της βαλβίδας (Valve Body). Είναι το κύριο στοιχείο μιας διάταξης βαλβίδας, καθώς συγκρατεί όλα τα συναρμολογούμενα μέρη μαζί. Δια μέσου αυτού περνάει η ροή των υγρών /αερίων και δέχεται τις πιέσεις του ρευστού. Τα άκρα του είναι έτσι σχεδιασμένα, ώστε να συνδέονται οι σωληνώσεις εισόδου και εξόδου. Το υλικό από το

οποίο κατασκευάζεται εξαρτάται από τη χρήση που προορίζεται η βαλβίδα. Μπορεί να είναι μεταλλικό (ορείχαλκος, μπρούτζο χυτοσίδηρος, ανοξείδωτος σίδηρος) ή πλαστικό (PVC,PTFE,PP) για μικρές πιέσεις ρευστού.

Καλύπτρα (Valve Bonnet). Η καλύπτρα λειτουργεί ως καπάκι του κελύφους της βαλβίδας, καθώς αφού τοποθετηθούν όλα τα συναρμολογούμενα μέρη της βαλβίδας στο κέλυφος, τοποθετείται η καλύπτρα έτσι ώστε να τα συγκρατεί ενωμένα και σφικτά. Όπως και το κέλυφος και η καλύπτρα δέχεται την πίεση του ρευστού. Συνήθως είναι από το ίδιο υλικό που είναι και το κύριο σώμα της βαλβίδας. Πολλές βαλβίδες, όπως οι βαλβίδες κώνου αλλά και οι βαλβίδες μπίλιας, δεν έχουν καλύπτρα αφού το κέλυφος είναι διαιρετό.

Εξάρτημα κλεισίματος (Disk). Είναι το κομμάτι εκείνο τη βαλβίδας που επιτρέπει η εμποδίζει την ροή του ρευστού ανάλογα με την θέση του. Ανάλογα με το είδος της βαλβίδας, το εξάρτημα κλεισίματος είναι διαφορετικό, για παράδειγμα στις βαλβίδες μπίλιας είναι μπίλια, στις βαλβίδες πεταλούδας είναι δίσκος. Η επιφάνεια του εξαρτήματος κλεισίματος πρέπει να είναι τόσο καλή έτσι ώστε η τριβή με το ρευστό να είναι όσο γίνεται μικρότερη.

Έδραση (Seat). Η έδραση είναι η επιφάνεια που εδράζεται το εξάρτημα κλεισίματος. Μεταξύ του εξαρτήματος κλεισίματος και της έδρασης υπάρχει η τσιμούχα συνήθως o-ring, η οποία εμποδίζει την διαρροή ρευστού όταν η βαλβίδα είναι σε θέση off. Οι βαλβίδες μπορεί να έχουν μια έδραση όπως οι πλανητικές βαλβίδες ή δύο εδράσεις, όπως οι βαλβίδες μπίλιας, μια ανάντη και μια κατάντη της μπίλιας. Τα υλικά από τα οποία κατασκευάζονται, είναι PTFE-Teflon,NBR,EPDM και FKM.

Στέλεχος (Stem). Είναι το κομμάτι της βαλβίδας που είναι υπεύθυνο για τον έλεγχο του εξαρτήματος κλεισίματος. Το στέλεχος ενώνει τον επενεργητή της βαλβίδας με το εξάρτημα κλεισίματος. Η κίνησή του είναι γραμμική ή κυκλική.

Επενεργητής (Actuators). Ο επενεργητής είναι μηχανισμός υπεύθυνος για το άνοιγμα ή το κλείσιμο της βαλβίδας, ανάλογα με την εντολή που θα του δοθεί. Προσαρμόζεται στο στέλεχος της βαλβίδας.

Οι επενεργητές διακρίνονται σε *χειροκίνητους* και *αυτόματους*. Ο μηχανισμός του χειροκίνητου επενεργητή ανάλογα με το είδος της βαλβίδας, αποτελείται από λεβιέ ή κοχλία. Όταν ασκηθεί γραμμική δύναμη στον πρώτο ή ροπή στον δεύτερο τότε η βαλβίδα ανοίγει ή κλείνει ανάλογα. Οι αυτόματοι επενεργητές διακρίνονται σε:

- Ηλεκτρικούς επενεργητές (Electric/Motor Actuators). Ο ηλεκτρικός επενεργητής

αποτελείται από έναν ηλεκτροκινητήρα. Ανάλογα με το ηλεκτρικό σήμα που λάβει, το σύστημα των γραναζιών ρυθμίζεται έτσι ώστε η βαλβίδα να ενεργοποιηθεί ή να απενεργοποιηθεί. Ένα βασικό πλεονέκτημα των ηλεκτρικών επενεργητών είναι ότι, η βαλβίδα εκτός από τις θέσεις ON και OFF, μπορεί να πάρει και ενδιάμεσες.

- Πνευματικούς Επενεργητές (Air/Pneumatic Actuators). Οι πνευματικοί επενεργητές χρησιμοποιούν πεπιεσμένο αέρα έτσι ώστε να ελέγξουν την βαλβίδα. Η δύναμη του πεπιεσμένου αέρα ασκείται πάνω σε διάφραγμα ή σε έμβολο, τα οποία με τη σειρά τους επιδρούν στο στέλεχος της βαλβίδας. Υπάρχουν δύο ειδών πνευματικοί επενεργητές: μονής και διπλής ενέργειας. Στους επενεργητές μονής ενέργειας προκειμένου να επιστρέψει η βαλβίδα στην αρχική κατάσταση, χρησιμοποιείται ελατήριο. Στους επενεργητές διπλής ενέργειας, ανάλογα με την κατεύθυνση της ροής του αέρα, ενεργοποιείται ή απενεργοποιείται η βαλβίδα. Ένα βασικό πλεονέκτημα των πνευματικών επενεργητών είναι ότι ο χρόνος που η βαλβίδα χρειάζεται για να μεταβεί από την θέση ON στη θέση OFF και το αντίστροφο είναι πάρα πολύ μικρός, δηλαδή έχει πολύ γρήγορη απόκριση.
- Υδραυλικούς επενεργητές (Hydraulic Actuators). Η λειτουργία των υδραυλικών επενεργητών είναι ίδια με αυτήν των πνευματικών, με τη διαφορά ότι αντί για πεπιεσμένο αέρα εκμεταλλεύονται την πίεση του υδραυλικού υγρού, που μπορεί να είναι νερό ή λάδι.
- Επενεργητές πηνίου (Solenoid Actuators). Πρόκειται για ηλεκτρομηχανικό εξοπλισμό ο οποίος εκμεταλλεύεται το μαγνητικό πεδίο που δημιουργείται λόγω του πηνίου όταν διαρρέεται από ρεύμα. Δύο μέθοδοι ενεργοποίησης χρησιμοποιούνται στις βαλβίδες με επενεργητές πηνίου: άμεσης λειτουργίας (direct acting) και λειτουργίας πιλότου (pilot operated). Μια σημαντική διαφορά μεταξύ των δύο αυτών μεθόδων είναι ότι οι βαλβίδες άμεσης λειτουργίας μπορούν να λειτουργήσουν και σε **συνθήκες κενού**, ενώ αντίθετα οι βαλβίδες πιλότου χρειάζονται διαφορά πίεσης μεγαλύτερη των 0,05 MPa προκειμένου να λειτουργήσουν [23].

8.2 Επιλογή Βαλβίδας για Έγχυση Ρητίνης

8.2.1 Αναγκαία Χαρακτηριστικά

Λόγω της ποικιλομορφίας των συστημάτων, των υγρών και των περιβαλλοντικών συνθηκών στα οποία πρέπει να λειτουργούν οι βαλβίδες, έχουν αναπτυχθεί πολλοί τύποι βαλβίδων. Μερικά παραδείγματα κοινών τύπων βαλβίδων είναι: οι σφαιρικές βαλβίδες, οι βαλβίδες

πεταλούδας, οι κωνικές βαλβίδες, οι βαλβίδες πύλης, οι βαλβίδες διαφράγματος, οι βαλβίδες στραγγαλισμού, βαλβίδες γωνίας, πλανητικές βαλβίδες καθώς και άλλες.

Στην παρούσα εργασία θέλουμε να χρησιμοποιήσουμε την βαλβίδα για τον άμεσο έλεγχο της ρητίνης σε συνθήκες κενού. Επομένως κατανοούμε ότι η βαλβίδα που πρέπει να επιλέξουμε πρέπει να έχει τα εξής χαρακτηριστικά:

- όσο των δυνατών μικρότερη πτώση πίεσης κατά τη διέλευση του ρευστού, αφού δεν έχουμε μεγάλη διαφορά πίεσης μεταξύ της εισόδου και της εξόδου της ρητίνης. Η διαφορά πίεσης που έχουμε είναι περίπου 0,8 bar.
- αποφυγή στενών κοιλοτήτων, καθώς το ιξώδες της ρητίνης που χρησιμοποιούμε είναι μεγάλο, περίπου 325-375 mPa.s (25° c) και σε περίπτωση διέλευσης της ρητίνης σε στενές κοιλότητες θα είχαμε μεγάλη πτώση της ταχύτητας ροής.
- μικρό χρόνο απόκρισης μετάβασης μεταξύ των δύο ακραίων θέσεων της βαλβίδας.

8.2.2 Ιδανικές Βαλβίδες Έγχυσης Ρητίνης Υπό Κενό

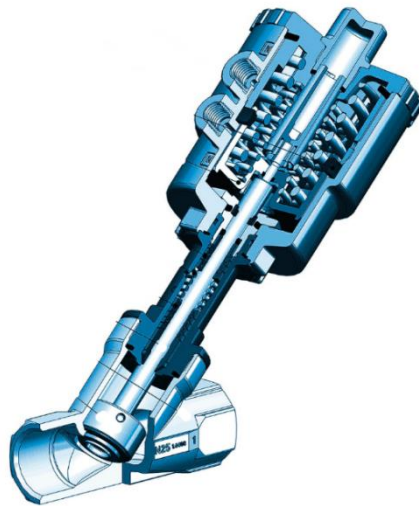
Από τους παραπάνω περιορισμούς προκύπτει ότι κατάλληλες βαλβίδες για τον έλεγχο της ρητίνης υπό συνθήκες κενού είναι: οι γωνιακές βαλβίδες έδρασης εμβόλου (angle seat piston valve), οι ομοαξονικές βαλβίδες (coaxial valve) και οι σφαιρικές βαλβίδες (ball valve), καθώς ικανοποιούν και τα τρία παραπάνω χαρακτηριστικά. Αρχικά θα γίνει περιγραφή των βαλβίδων αυτών, και τελικά θα αιτιολογηθεί η επιλογή μας.

Γωνιακή Βαλβίδα Εμβόλου: Είναι κατάλληλη για εφαρμογές κενού και για παχύρευστα υγρά καθώς έχει καθαρή διαδρομή ροής με σχεδόν μηδενική πτώση πίεσης. Η ροή ελέγχεται από έμβολο υπό γωνία. Ο έλεγχος του εμβόλου γίνεται με πνευματικό επενεργητή ο οποίος είναι ενσωματωμένος στην βαλβίδα (εικόνα 53).

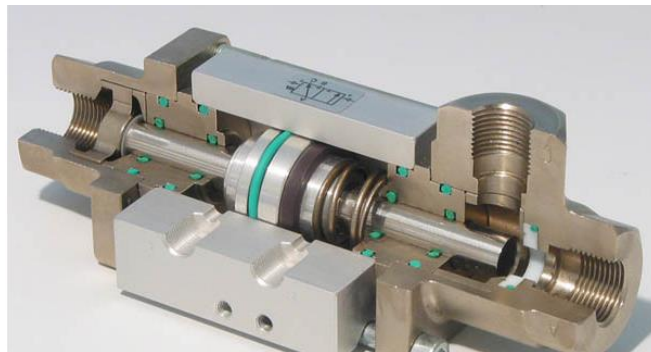
Ομοαξονική Βαλβίδα: Λειτουργεί χρησιμοποιώντας ένα εσωτερικό, πλευρικά κινητό, κούλο σωλήνα, ο οποίος ελέγχει τη ροή που περνάει μέσα από τη βαλβίδα. Όταν ενεργοποιηθεί η βαλβίδα, ο σωλήνας αυτός μετακινείται αξονικά, έτσι ώστε να απομακρυνθεί από την έδραση και το ρευστό να μπορεί να περάσει. Λόγω του σχεδιασμού της, η πτώση της πίεσης είναι σχεδόν μηδενική. Η ενεργοποίηση της ομοαξονικής βαλβίδας μπορεί να γίνει είτε με πνευματικό επενεργητή, είτε με επενεργητή πηνίου, οι οποίοι σε κάθε περίπτωση είναι ενσωματωμένοι στη βαλβίδα (εικόνα 54).

Σφαιρική Βαλβίδα: Είναι ένα είδος βαλβίδας στρέψης, η οποία χρησιμοποιεί μια κούφια, διάτρητη και περιστρεφόμενη σφαίρα για τον έλεγχο της ροής. Η σφαίρα κινείται κατά 90 μοίρες όταν ενεργοποιείται. Όταν η σφαιρική βαλβίδα είναι ανοικτή, η διαδρομή της ροής

είναι τελείως γραμμική, χωρίς εμπόδια κατά το μήκος της ροής, με αποτέλεσμα να έχουμε σχεδόν μηδενική πτώση πίεσης, οπότε καθίσταται ιδανική για εφαρμογές κενού και για ρευστά με μεγάλο ιξώδες. Οι επενεργητές που θα προσαρμοστούν στην σφαιρική βαλβίδα για τον έλεγχο της είναι είτε ηλεκτρικοί, είτε πνευματικοί (εικόνα 55).



Εικόνα 53: Γωνιακή βαλβίδα εμβόλου



Εικόνα 54: Ομοαξονική βαλβίδα (πνευματική)



Εικόνα 55: Σφαιρική βαλβίδα

8.2.3 Τελική Επιλογή Βαλβίδας

8.2.3.1 Σφαιρική Βαλβίδα

Στην παρούσα εργασία επιλέχτηκε σφαιρική βαλβίδα, για τους εξής λόγους:

- Έχει μεγαλύτερο συντελεστή ροής C_v σε σχέση με τις ομοαξονικές βαλβίδες και τις γωνιακές βαλβίδες εμβόλου [24].
- Είναι πολύ πιο εύκολη στην συντήρηση και τον καθαρισμό των τμημάτων από όπου περνάει το ρευστό. Αυτό είναι πολύ σημαντικό για την εφαρμογή μας, καθώς στο τέλος της κατεργασίας VARI, η βαλβίδα πρέπει να καθαριστεί πάρα πολύ καλά, έτσι ώστε να μη στερεοποιηθούν υπολείμματα ρητίνης στο εσωτερικό της βαλβίδας.

8.2.3.2 Επενεργητής Σφαιρικής Βαλβίδας

Στην παρούσα εργασία επιλέχτηκε πνευματικός επενεργητής. Ο βασικότερος λόγος για την επιλογή του υδραυλικού επενεργητή, είναι ότι η απόκρισή του είναι πολύ πιο γρήγορη σε σχέση με τον ηλεκτρικό επενεργητή. Ειδικότερα, περιστροφή 90 μοιρών της σφαιρικής βαλβίδας γίνεται σε περίπου 0,5 sec. Να επισημανθεί ότι ο χρόνος μετάβασης μεταξύ των δύο ακραίων καταστάσεων εξαρτάται από την πίεση του αέρα που εφαρμόζεται. Αντίστοιχα, για τον ηλεκτρικό επενεργητή ο χρόνος δεν είναι λιγότερος από 6 sec.

8.2.3.3 Επιλογή Υλικού Κατασκευής Σφαιρικής Βαλβίδας

Στην παρούσα εργασία το ρευστό το οποίο θα διαρρέει την βαλβίδα είναι εποξική ρητίνη (epoxy resin). Επιπλέον για τον καθαρισμό της βαλβίδας από την ρητίνη χρησιμοποιείται Acetone.

Πίνακας 3: Αντίδραση Υλικών

Category Description	PP	316SS	Carbon	Ceramic	PTFE	EPDM	FKM	NBR
A No known effect								
B Some effect; evaluate with caution								
C Moderate to Severe effect; evaluation not recommended								
D Unknown								
Acetone	B	A	A	A	A	A	C	C
Nitrotoluene	D	D	D	D	D	A	C	C
Epoxy Resins	D	D	D	D	D	A	C	D

Για το κέλυφος και τη σφαίρα της βαλβίδας μπορούμε να επιλέξουμε οποιοδήποτε μεταλλικό υλικό, εν προκειμένω ορείχαλκο. Για την επιλογή του υλικού των τσιμουχών, με βάση τον παρακάτω πίνακα, αποφεύγεται το FKM και το NBR, οπότε επιλέξαμε PTFE.

8.3 Σύνδεση Υδραυλικών – Πνευματικών Στοιχείων

Στο σύστημά μας για την κατεργασία VARI, έχουμε δύο εισόδους για την εισαγωγή ρητίνης και μια έξοδο για την σύνδεση με την αντλία κενού. Επομένως χρησιμοποιούμε δύο σφαιρικές πνευματικές βαλβίδες.

8.3.1 Ηλεκτροβαλβίδα Ελέγχου

Για τον έλεγχο του πνευματικού επενεργητή της σφαιρικής βαλβίδας, χρησιμοποιούμε κατευθυντική βαλβίδα ελέγχου. Οι κατευθυντικές βαλβίδες ελέγχου καθορίζουν την διεύθυνση της ροής του πεπιεσμένου αέρα για τον έλεγχο της θέσης λειτουργίας του επενεργητή. Ανάλογα με την κατεύθυνση της ροής, η πνευματική σφαιρική βαλβίδα ανοίγει ή κλείνει αντίστοιχα. Η ενεργοποίηση της βαλβίδας μπορεί να γίνει:

- Χειροκίνητα
- Με εντολή αέρα
- Με ηλεκτρικό σήμα σε πηνίο

Στην παρούσα διπλωματική εργασία χρησιμοποιούμε ηλεκτροβαλβίδα ελέγχου, καθώς επιθυμούμε αυτόματο έλεγχο στην ηλεκτροβαλβίδα με χρήση μικροελεγκτή.

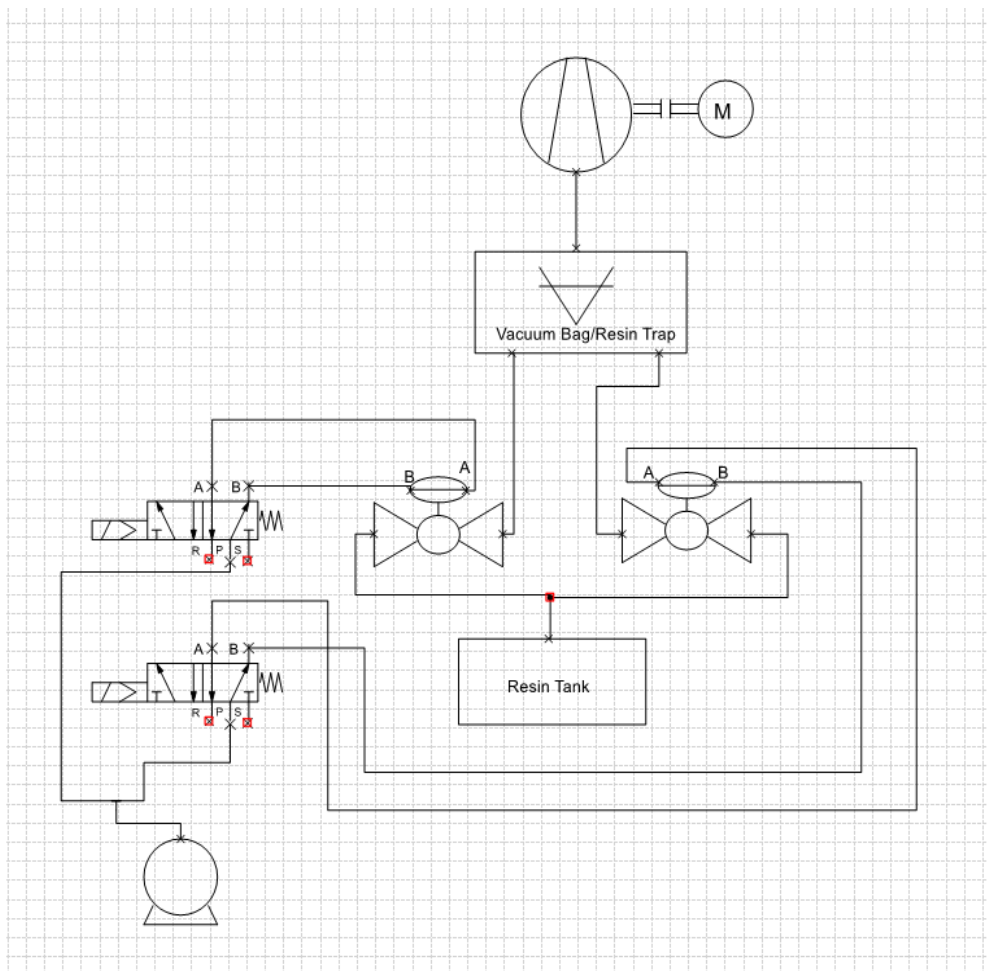
Οι επενεργητές των δύο σφαιρικών βαλβίδων που χρησιμοποιούμε είναι διπλής ενέργειας επομένως θα χρησιμοποιήσουμε δύο πενταδικές ηλεκτροβαλβίδες αέρος δύο θέσεων (5/2), για τον έλεγχο της κάθε πνευματικής βαλβίδας (εικόνα 56).



Εικόνα 56: Πενταοδική Βαλβίδα Αέρος 2 Θέσεων

Η πενταοδική ηλεκτροβαλβίδα, έχει δύο πόρτες εξόδου (A,B), μια πόρτα εισόδου(P) και δύο πόρτες εκτόνωσης (R,S). Η κάθε μία από τις πόρτες εξόδου (A,B), συνδέεται στις δύο πόρτες εισόδου (A,B) του πνευματικού επενεργητή. Ανάλογα με το συνδυασμό σύνδεσης, αλλάζει η αρχική κατάσταση της πνευματικής βαλβίδας. Στην περίπτωσή μας, όταν δεν δίνεται

εντολή στην ηλεκτροβαλβίδα, θέλουμε η πνευματική βαλβίδα να είναι κλειστή. Στην πόρτα εισόδου (P) συνδέουμε την παροχή του πεπιεσμένου αέρα. Οι δύο πόρτες εκτόνωσης (R,S) μένουν ελεύθερες έτσι ώστε ο πεπιεσμένος αέρας να εκτονώνεται στο περιβάλλον, καθώς όταν θέλουμε να αλλάξουμε την κατάσταση του επενεργητή, ο αέρας είναι εγκλωβισμένος μέσα σε αυτό, και πρέπει να εκτονωθεί. Το πνευματικό-υδραυλικό σχέδιο φαίνεται παρακάτω (Εικόνα 54).






Εικόνα 54: Υδραυλικό-Πνευματικό Σχέδιο

8.4 Εξαρτήματα

Για την πραγματοποίηση του ελέγχου της έγχυσης ρητίνης, οι βαλβίδες που χρησιμοποιήθηκαν καθώς και τα παρελκόμενα εξαρτήματα για την σύνδεση μεταξύ τους, και με την παροχή του πεπιεσμένου αέρα περιγράφονται στον παρακάτω πίνακα.

Πίνακας 4: Εξαρτήματα Υδραυλικού Πνευματικού Κυκλώματος

α/α	Είδος	Χαρακτηριστικά	Εικόνα																																
1	Πνευματική Βάνα Δίοδη Ball Valve	<table border="1"> <tr> <td>Κωδικός SKU</td> <td>VD100200</td> </tr> <tr> <td>Κατασκευαστής</td> <td>Alpharompe</td> </tr> <tr> <td>Λειτουργία</td> <td>2/2</td> </tr> <tr> <td>Σπείρωμα εισόδου-εξόδου</td> <td>G1/2"</td> </tr> <tr> <td>Διάμετρος εμβόλου (mm)</td> <td>32</td> </tr> <tr> <td>Μεγ.πίεση εμβόλου (bar)</td> <td>8</td> </tr> <tr> <td>Δύναμη εμβόλου σε 6 bar (Nm)</td> <td>7.6</td> </tr> <tr> <td>Βάση σύνδεσης</td> <td>ISO 5211 / DIN 3337</td> </tr> <tr> <td>Υλικό βάνας</td> <td>Επινικελαωμένος ορείχαλκος</td> </tr> <tr> <td>O-ring βάνας</td> <td>Viton</td> </tr> <tr> <td>Οδηγός βάνας</td> <td>PTFE</td> </tr> <tr> <td>Θερμοκρασία λειτουργίας (°C)</td> <td>-20 έως +80</td> </tr> <tr> <td>Υλικό πιστονιού εμβόλου</td> <td>Ανοδιωμένο αλουμίνιο</td> </tr> <tr> <td>Μήκος (mm)</td> <td>117</td> </tr> <tr> <td>Ύψος (mm)</td> <td>101</td> </tr> <tr> <td>Πλάτος (mm)</td> <td>45</td> </tr> </table>	Κωδικός SKU	VD100200	Κατασκευαστής	Alpharompe	Λειτουργία	2/2	Σπείρωμα εισόδου-εξόδου	G1/2"	Διάμετρος εμβόλου (mm)	32	Μεγ.πίεση εμβόλου (bar)	8	Δύναμη εμβόλου σε 6 bar (Nm)	7.6	Βάση σύνδεσης	ISO 5211 / DIN 3337	Υλικό βάνας	Επινικελαωμένος ορείχαλκος	O-ring βάνας	Viton	Οδηγός βάνας	PTFE	Θερμοκρασία λειτουργίας (°C)	-20 έως +80	Υλικό πιστονιού εμβόλου	Ανοδιωμένο αλουμίνιο	Μήκος (mm)	117	Ύψος (mm)	101	Πλάτος (mm)	45	
Κωδικός SKU	VD100200																																		
Κατασκευαστής	Alpharompe																																		
Λειτουργία	2/2																																		
Σπείρωμα εισόδου-εξόδου	G1/2"																																		
Διάμετρος εμβόλου (mm)	32																																		
Μεγ.πίεση εμβόλου (bar)	8																																		
Δύναμη εμβόλου σε 6 bar (Nm)	7.6																																		
Βάση σύνδεσης	ISO 5211 / DIN 3337																																		
Υλικό βάνας	Επινικελαωμένος ορείχαλκος																																		
O-ring βάνας	Viton																																		
Οδηγός βάνας	PTFE																																		
Θερμοκρασία λειτουργίας (°C)	-20 έως +80																																		
Υλικό πιστονιού εμβόλου	Ανοδιωμένο αλουμίνιο																																		
Μήκος (mm)	117																																		
Ύψος (mm)	101																																		
Πλάτος (mm)	45																																		
2	Πνευματική Βάνα Δίοδη Ball Valve	<table border="1"> <tr> <td>Κωδικός SKU</td> <td>411140U030</td> </tr> <tr> <td>Κατασκευαστής</td> <td>EVIAN</td> </tr> <tr> <td>Λειτουργία</td> <td>2/2</td> </tr> <tr> <td>Σπείρωμα εισόδου-εξόδου</td> <td>G1/2"</td> </tr> <tr> <td>Διάμετρος εμβόλου (mm)</td> <td>32</td> </tr> <tr> <td>Μεγ.πίεση εμβόλου (bar)</td> <td>8</td> </tr> <tr> <td>Δύναμη εμβόλου σε 6 bar (Nm)</td> <td>7.6</td> </tr> <tr> <td>Βάση σύνδεσης</td> <td>ISO 5211 / DIN 3337</td> </tr> <tr> <td>Υλικό βάνας</td> <td>Επινικελαωμένος ορείχαλκος</td> </tr> <tr> <td>O-ring βάνας</td> <td>PTFE</td> </tr> <tr> <td>Οδηγός βάνας</td> <td>PTFE</td> </tr> <tr> <td>Θερμοκρασία λειτουργίας (°C)</td> <td>-20 έως +80</td> </tr> <tr> <td>Υλικό πιστονιού εμβόλου</td> <td>Ανοδιωμένο αλουμίνιο</td> </tr> <tr> <td>Μήκος (mm)</td> <td>119</td> </tr> <tr> <td>Ύψος (mm)</td> <td>101</td> </tr> <tr> <td>Πλάτος (mm)</td> <td>67</td> </tr> </table>	Κωδικός SKU	411140U030	Κατασκευαστής	EVIAN	Λειτουργία	2/2	Σπείρωμα εισόδου-εξόδου	G1/2"	Διάμετρος εμβόλου (mm)	32	Μεγ.πίεση εμβόλου (bar)	8	Δύναμη εμβόλου σε 6 bar (Nm)	7.6	Βάση σύνδεσης	ISO 5211 / DIN 3337	Υλικό βάνας	Επινικελαωμένος ορείχαλκος	O-ring βάνας	PTFE	Οδηγός βάνας	PTFE	Θερμοκρασία λειτουργίας (°C)	-20 έως +80	Υλικό πιστονιού εμβόλου	Ανοδιωμένο αλουμίνιο	Μήκος (mm)	119	Ύψος (mm)	101	Πλάτος (mm)	67	
Κωδικός SKU	411140U030																																		
Κατασκευαστής	EVIAN																																		
Λειτουργία	2/2																																		
Σπείρωμα εισόδου-εξόδου	G1/2"																																		
Διάμετρος εμβόλου (mm)	32																																		
Μεγ.πίεση εμβόλου (bar)	8																																		
Δύναμη εμβόλου σε 6 bar (Nm)	7.6																																		
Βάση σύνδεσης	ISO 5211 / DIN 3337																																		
Υλικό βάνας	Επινικελαωμένος ορείχαλκος																																		
O-ring βάνας	PTFE																																		
Οδηγός βάνας	PTFE																																		
Θερμοκρασία λειτουργίας (°C)	-20 έως +80																																		
Υλικό πιστονιού εμβόλου	Ανοδιωμένο αλουμίνιο																																		
Μήκος (mm)	119																																		
Ύψος (mm)	101																																		
Πλάτος (mm)	67																																		

3	Ηλεκτρο- βαλβίδα αέρος	<table border="1"> <tbody> <tr> <td>Κωδικός SKU</td> <td>002-130-0010</td> </tr> <tr> <td>Κατασκευαστής</td> <td>Airblock</td> </tr> <tr> <td>Μεγ.Πίεση (bar)</td> <td>10</td> </tr> <tr> <td>Θερμοκρασία περιβάλλοντος (°C)</td> <td>-10 έως +60</td> </tr> <tr> <td>Σπείρωμα εισόδου-εξόδου</td> <td>G1/8"</td> </tr> <tr> <td>Υλικό σώματος</td> <td>Αλουμίνιο</td> </tr> <tr> <td>Λειτουργία</td> <td>5/2</td> </tr> <tr> <td>Διάμετρος πηνείου (mm)</td> <td>9</td> </tr> <tr> <td>Συνδεσμολογία πηνείου</td> <td>Με φινις DIN 46244</td> </tr> <tr> <td>Τάση πηνείου (V)</td> <td>12 DC</td> </tr> <tr> <td>Θερμοκρασία αέρα (°C)</td> <td>+5 έως +60</td> </tr> <tr> <td>Επιφάνεια παροχής (mm²)</td> <td>14</td> </tr> </tbody> </table>	Κωδικός SKU	002-130-0010	Κατασκευαστής	Airblock	Μεγ.Πίεση (bar)	10	Θερμοκρασία περιβάλλοντος (°C)	-10 έως +60	Σπείρωμα εισόδου-εξόδου	G1/8"	Υλικό σώματος	Αλουμίνιο	Λειτουργία	5/2	Διάμετρος πηνείου (mm)	9	Συνδεσμολογία πηνείου	Με φινις DIN 46244	Τάση πηνείου (V)	12 DC	Θερμοκρασία αέρα (°C)	+5 έως +60	Επιφάνεια παροχής (mm ²)	14	
Κωδικός SKU	002-130-0010																										
Κατασκευαστής	Airblock																										
Μεγ.Πίεση (bar)	10																										
Θερμοκρασία περιβάλλοντος (°C)	-10 έως +60																										
Σπείρωμα εισόδου-εξόδου	G1/8"																										
Υλικό σώματος	Αλουμίνιο																										
Λειτουργία	5/2																										
Διάμετρος πηνείου (mm)	9																										
Συνδεσμολογία πηνείου	Με φινις DIN 46244																										
Τάση πηνείου (V)	12 DC																										
Θερμοκρασία αέρα (°C)	+5 έως +60																										
Επιφάνεια παροχής (mm ²)	14																										
4	Ηλεκτρο- βαλβίδα αέρος τύπου NAMUR	<table border="1"> <tbody> <tr> <td>Κωδικός SKU</td> <td>EVNM55LML00M</td> </tr> <tr> <td>Κατασκευαστής</td> <td>AIRCOMP</td> </tr> <tr> <td>Μεγ.Πίεση (bar)</td> <td>10</td> </tr> <tr> <td>Θερμοκρασία περιβάλλοντος (°C)</td> <td>-10 έως +60</td> </tr> <tr> <td>Σπείρωμα εισόδου-εξόδου</td> <td>G1/8"</td> </tr> <tr> <td>Υλικό σώματος</td> <td>Αλουμίνιο</td> </tr> <tr> <td>Λειτουργία</td> <td>5/2</td> </tr> <tr> <td>Διάμετρος πηνείου (mm)</td> <td>9</td> </tr> <tr> <td>Συνδεσμολογία πηνείου</td> <td>Με φινις DIN 46244</td> </tr> <tr> <td>Τάση πηνείου (V)</td> <td>12 DC</td> </tr> <tr> <td>Θερμοκρασία αέρα (°C)</td> <td>+5 έως +60</td> </tr> <tr> <td>Επιφάνεια παροχής (mm²)</td> <td>14</td> </tr> </tbody> </table>	Κωδικός SKU	EVNM55LML00M	Κατασκευαστής	AIRCOMP	Μεγ.Πίεση (bar)	10	Θερμοκρασία περιβάλλοντος (°C)	-10 έως +60	Σπείρωμα εισόδου-εξόδου	G1/8"	Υλικό σώματος	Αλουμίνιο	Λειτουργία	5/2	Διάμετρος πηνείου (mm)	9	Συνδεσμολογία πηνείου	Με φινις DIN 46244	Τάση πηνείου (V)	12 DC	Θερμοκρασία αέρα (°C)	+5 έως +60	Επιφάνεια παροχής (mm ²)	14	
Κωδικός SKU	EVNM55LML00M																										
Κατασκευαστής	AIRCOMP																										
Μεγ.Πίεση (bar)	10																										
Θερμοκρασία περιβάλλοντος (°C)	-10 έως +60																										
Σπείρωμα εισόδου-εξόδου	G1/8"																										
Υλικό σώματος	Αλουμίνιο																										
Λειτουργία	5/2																										
Διάμετρος πηνείου (mm)	9																										
Συνδεσμολογία πηνείου	Με φινις DIN 46244																										
Τάση πηνείου (V)	12 DC																										
Θερμοκρασία αέρα (°C)	+5 έως +60																										
Επιφάνεια παροχής (mm ²)	14																										
5	Αυτόματο Ρακόρ T Με Σπείρωμα Στη Μέση	<table border="1"> <tbody> <tr> <td>Κωδικός SKU</td> <td>27216 00 008</td> </tr> <tr> <td>Κατασκευαστής</td> <td>AIGNEP</td> </tr> <tr> <td>Μέγιστη Πίεση (bar)</td> <td>20</td> </tr> <tr> <td>Θερμοκρασία περιβάλλοντος (°C)</td> <td>-20 έως +80</td> </tr> <tr> <td>Διατομή Σωλήνα (mm)</td> <td>6</td> </tr> <tr> <td>Διατομή Σπειρώματος (inch)</td> <td>1/8</td> </tr> <tr> <td>Υλικό</td> <td>Επινικελωμένος Ορείχαλκος</td> </tr> </tbody> </table>	Κωδικός SKU	27216 00 008	Κατασκευαστής	AIGNEP	Μέγιστη Πίεση (bar)	20	Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80	Διατομή Σωλήνα (mm)	6	Διατομή Σπειρώματος (inch)	1/8	Υλικό	Επινικελωμένος Ορείχαλκος											
Κωδικός SKU	27216 00 008																										
Κατασκευαστής	AIGNEP																										
Μέγιστη Πίεση (bar)	20																										
Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80																										
Διατομή Σωλήνα (mm)	6																										
Διατομή Σπειρώματος (inch)	1/8																										
Υλικό	Επινικελωμένος Ορείχαλκος																										

6	Ρακόρ Κωνικό Ευθύ Με Θηλυκό Σπειρώμα	<table border="1"> <tr><td>Κωδικός SKU</td><td>03050 00 01 BS NB</td></tr> <tr><td>Κατασκευαστής</td><td>AIGNEP</td></tr> <tr><td>Μέγιστη Πίεση (bar)</td><td>20</td></tr> <tr><td>Θερμοκρασία περιβάλλοντος (°C)</td><td>-20 έως +80</td></tr> <tr><td>Διατομή Σωλήνα (mm)</td><td>6</td></tr> <tr><td>Διατομή Σπειρώματος (inch)</td><td>1/8</td></tr> <tr><td>Υλικό</td><td>Επινικελωμένος Ορείχαλκος</td></tr> </table>	Κωδικός SKU	03050 00 01 BS NB	Κατασκευαστής	AIGNEP	Μέγιστη Πίεση (bar)	20	Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80	Διατομή Σωλήνα (mm)	6	Διατομή Σπειρώματος (inch)	1/8	Υλικό	Επινικελωμένος Ορείχαλκος			
Κωδικός SKU	03050 00 01 BS NB																		
Κατασκευαστής	AIGNEP																		
Μέγιστη Πίεση (bar)	20																		
Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80																		
Διατομή Σωλήνα (mm)	6																		
Διατομή Σπειρώματος (inch)	1/8																		
Υλικό	Επινικελωμένος Ορείχαλκος																		
7	Ευθύ Αυτόματο Ρακόρ (x 8)	<table border="1"> <tr><td>Κωδικός SKU</td><td>57000 00 002</td></tr> <tr><td>Κατασκευαστής</td><td>AIGNEP</td></tr> <tr><td>Μέγιστη Πίεση (bar)</td><td>20</td></tr> <tr><td>Θερμοκρασία περιβάλλοντος (°C)</td><td>-20 έως +80</td></tr> <tr><td>Διατομή Σωλήνα (mm)</td><td>6</td></tr> <tr><td>Διατομή Σπειρώματος (inch)</td><td>1/8</td></tr> <tr><td>Υλικό</td><td>Επινικελωμένος Ορείχαλκος</td></tr> </table>	Κωδικός SKU	57000 00 002	Κατασκευαστής	AIGNEP	Μέγιστη Πίεση (bar)	20	Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80	Διατομή Σωλήνα (mm)	6	Διατομή Σπειρώματος (inch)	1/8	Υλικό	Επινικελωμένος Ορείχαλκος			
Κωδικός SKU	57000 00 002																		
Κατασκευαστής	AIGNEP																		
Μέγιστη Πίεση (bar)	20																		
Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80																		
Διατομή Σωλήνα (mm)	6																		
Διατομή Σπειρώματος (inch)	1/8																		
Υλικό	Επινικελωμένος Ορείχαλκος																		
8	Ίσιο Βιδωτό Ρακόρ (x4)	<table border="1"> <tr><td>Κωδικός SKU</td><td>01020 00 014</td></tr> <tr><td>Κατασκευαστής</td><td>AIGNEP</td></tr> <tr><td>Μέγιστη Πίεση (bar)</td><td>20</td></tr> <tr><td>Θερμοκρασία περιβάλλοντος (°C)</td><td>-20 έως +80</td></tr> <tr><td>Διατομή Σωλήνα out/in (mm)</td><td>10/8</td></tr> <tr><td>Διατομή Σπειρώματος (inch)</td><td>1/8</td></tr> <tr><td>Υλικό</td><td>Επινικελωμένος Ορείχαλκος</td></tr> </table>	Κωδικός SKU	01020 00 014	Κατασκευαστής	AIGNEP	Μέγιστη Πίεση (bar)	20	Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80	Διατομή Σωλήνα out/in (mm)	10/8	Διατομή Σπειρώματος (inch)	1/8	Υλικό	Επινικελωμένος Ορείχαλκος			
Κωδικός SKU	01020 00 014																		
Κατασκευαστής	AIGNEP																		
Μέγιστη Πίεση (bar)	20																		
Θερμοκρασία περιβάλλοντος (°C)	-20 έως +80																		
Διατομή Σωλήνα out/in (mm)	10/8																		
Διατομή Σπειρώματος (inch)	1/8																		
Υλικό	Επινικελωμένος Ορείχαλκος																		
9	Λάστιχο Αέρος	<table border="1"> <tr><td>Κωδικός SKU</td><td>70227.02150.59007</td></tr> <tr><td>Κατασκευαστής</td><td>Fitt</td></tr> <tr><td>Εξωτερική διάμετρος (mm)</td><td>11</td></tr> <tr><td>Εσωτερική διάμετρος (mm)</td><td>6</td></tr> <tr><td>Πίεση λειτουργίας (bar)</td><td>20</td></tr> <tr><td>Μεγ.Πίεση θραύσης (bar)</td><td>60</td></tr> <tr><td>Θερμοκρασία λειτουργίας (°C)</td><td>-10 έως +50</td></tr> <tr><td>Βάρος (gr/mt)</td><td>87</td></tr> </table>	Κωδικός SKU	70227.02150.59007	Κατασκευαστής	Fitt	Εξωτερική διάμετρος (mm)	11	Εσωτερική διάμετρος (mm)	6	Πίεση λειτουργίας (bar)	20	Μεγ.Πίεση θραύσης (bar)	60	Θερμοκρασία λειτουργίας (°C)	-10 έως +50	Βάρος (gr/mt)	87	
Κωδικός SKU	70227.02150.59007																		
Κατασκευαστής	Fitt																		
Εξωτερική διάμετρος (mm)	11																		
Εσωτερική διάμετρος (mm)	6																		
Πίεση λειτουργίας (bar)	20																		
Μεγ.Πίεση θραύσης (bar)	60																		
Θερμοκρασία λειτουργίας (°C)	-10 έως +50																		
Βάρος (gr/mt)	87																		

10	Σωληνάκι Αέρος	Κωδικός SKU	TPU 4/6	
		Κατασκευαστής	Airblock	
		Εξωτερική διάμετρος (mm)	6	
		Εσωτερική διάμετρος (mm)	4	
		Πίεση λειτουργίας (bar)	13	
		Μεγ.Πίεση θραύσης (bar)	40	
		Θερμοκρασία λειτουργίας (°C)	-40 έως + 60	
		Βάρος (gr/mt)	19.5	

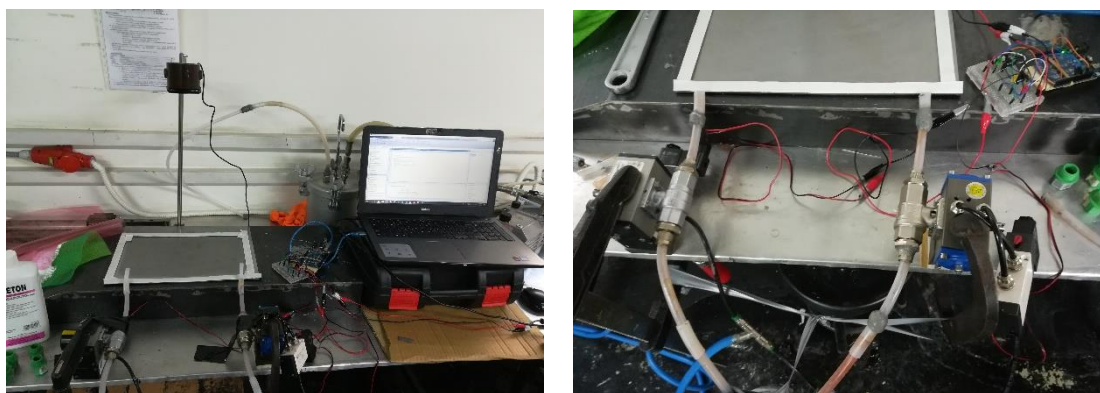
ΚΕΦΑΛΑΙΟ 9 Αυτοματοποιημένη Διεργασία VARI

9.1 Εισαγωγή

Στο κεφάλαιο αυτό θα περιγραφούν τα βήματα που εκτελούνται για την αυτοματοποιημένη κατεργασία VARI, καθώς και η συνολική διάταξη του εξοπλισμού, όπως έχει εγκατασταθεί στη μονάδα σύνθετων υλικών του Εργαστηρίου Τεχνολογίας των Κατεργασιών της Σχολής Μηχανολόγων Μηχανικών.

9.2 Προετοιμασία εξοπλισμού

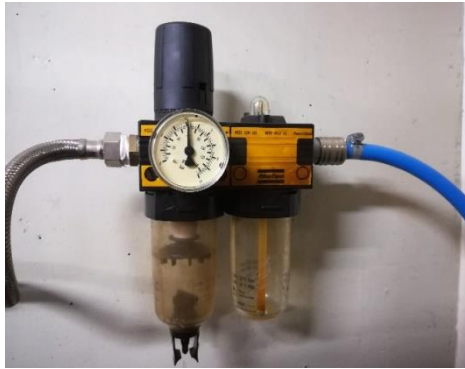
Αρχικά για την υλοποίηση της αυτοματοποιημένης κατεργασίας VARI, πρέπει να προετοιμάσουμε τον εξοπλισμό, δηλαδή τις βαλβίδες, το Arduino, την κάμερα και το καλούπι στο οποίο θα πραγματοποιηθεί η κατεργασία VARI. Να σημειωθεί πως το καλούπι που χρησιμοποιούμε είναι επίπεδη επιφάνεια διαστάσεων 350 x 250 mm από αλουμίνιο. Η κάμερα για την καταγραφή του μετώπου που χρησιμοποιήθηκε είναι η Logitech C525 με μέγιστη ανάλυση 720p (1280 x 720 pixels) και 30 frames/sec, η οποία είναι στερεωμένη σε ειδική βάση για ρύθμιση της θέσης της. Η εγκατάσταση φαίνεται παρακάτω.



Εικόνα 59: Εγκατάσταση Αυτοματοποιημένης Κατεργασίας VARI

Μέσω της MATLAB εκτελώντας τις εντολές `cam=webcam` και `preview (cam)`, έτσι ώστε να έχουμε εικόνα από την κάμερα, ρυθμίζουμε κατάλληλα την θέση της κάμερας έτσι ώστε να εστιάζει σε όλο το καλούπι.

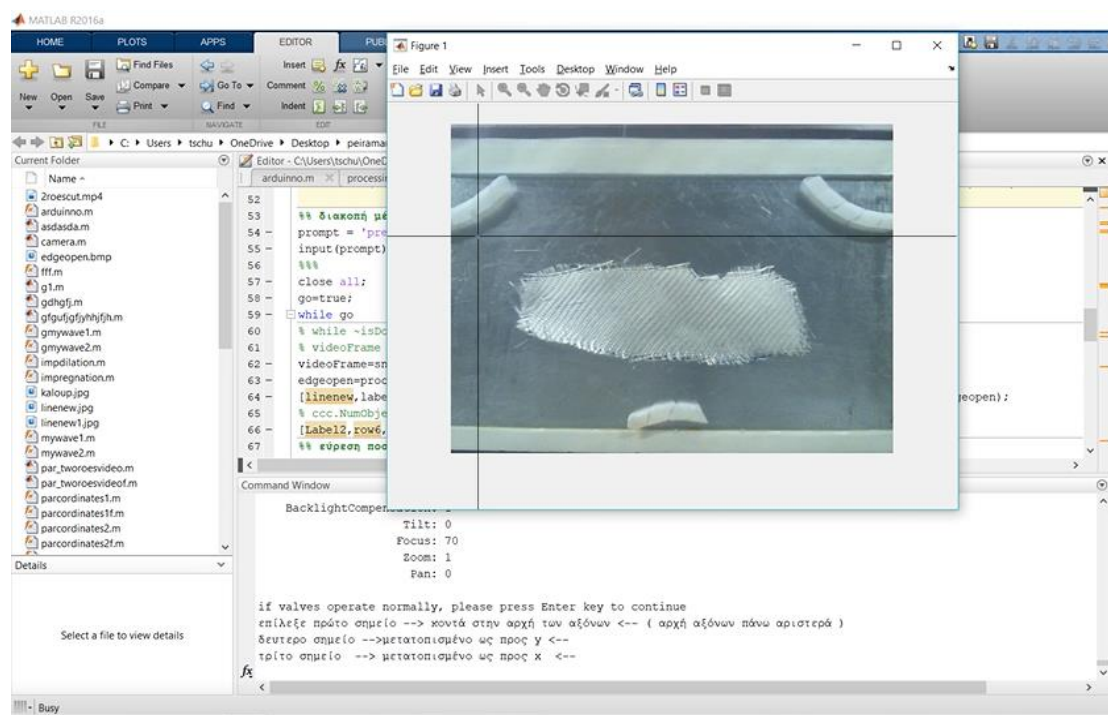
Η πίεση του αέρα πρέπει να είναι μεγαλύτερη από 3.5 bar, ώστε οι βαλβίδες να λειτουργούν σωστά. Στο κύκλωμα πεπιεσμένου αέρα του Εργαστηρίου, αυτό γίνεται με ρυθμιστή πίεση, όπως φαίνεται παρακάτω.



Εικόνα 60: Ρυθμιστής Πίεσης Αέρα Βαλβίδων

9.3 Έναρξη Διεργασίας

Αρχικά τοποθετούμε στο καλούπι το ύφασμα που θα εμποτισθεί με ρητίνη. Το τοποθετούμε έτσι ώστε να μην είναι κοντά στα σπινάλια τα οποία βοηθάνε στην καλύτερη ροή της ρητίνης. Αφού τοποθετήσουμε το ύφασμα, εκτελούμε το πρόγραμμα `tworoesvideo.m`. Στο παράθυρο εντολών της MATLAB εμφανίζονται οι παραμετροποιήσεις της κάμερας και του Arduino από την MATLAB. Αφού ανοιγοκλείσουν πέντε φορές οι βαλβίδες, θα εμφανιστεί μήνυμα για επιβεβαίωση από τον χρήστη καλής λειτουργίας αυτών, με Enter. Στη συνέχεια εμφανίζεται στη οθόνη του υπολογιστή το ύφασμα προς εμποτισμό, με τις οδηγίες που πρέπει να ακολουθηθούν προκειμένου να επιλέξουμε τα τρία σημεία για την οριοθέτηση της περιοχής ενδιαφέροντος, όπως φαίνεται στην παρακάτω εικόνα.

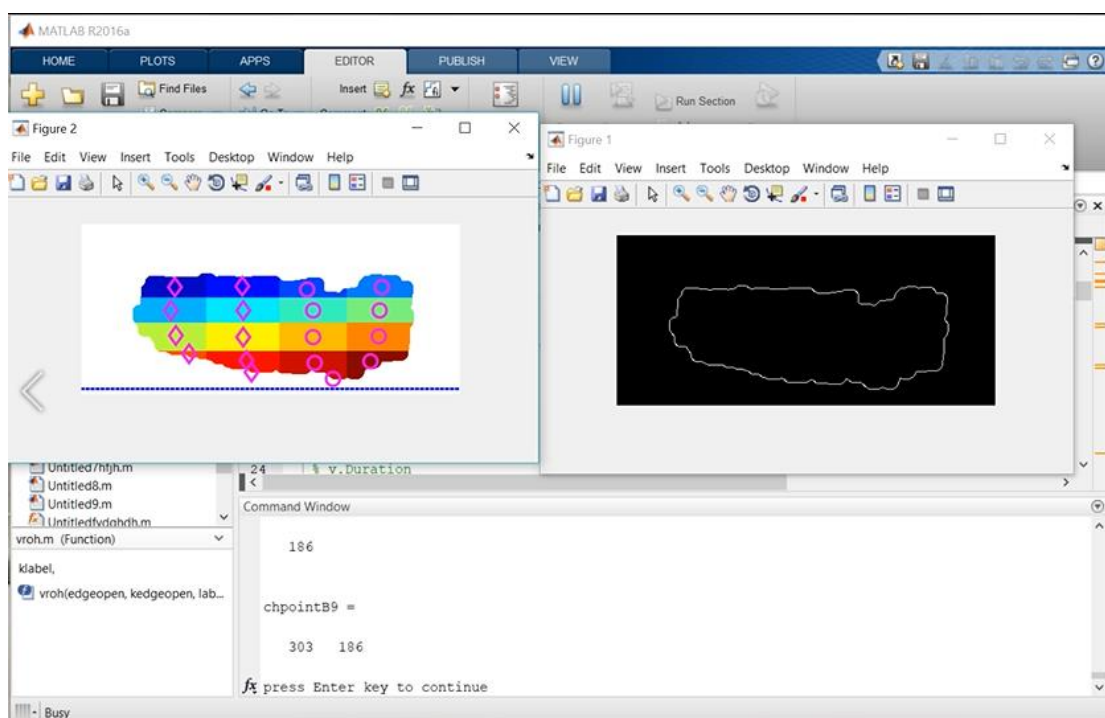


Εικόνα 61: Επιβεβαίωση Καλής Λειτουργίας Βαλβίδων/ Οριοθέτηση Περιοχής Ενδιαφέροντος

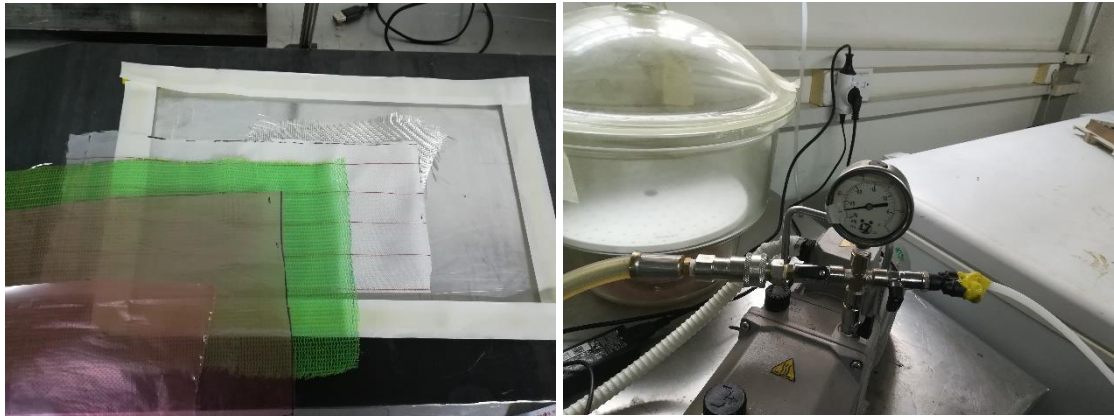
Επιθυμητό είναι το δεύτερο σημείο να επιλεχτεί όσο γίνεται πιο μακριά από τα όρια του υφάσματος, έτσι ώστε να υπάρχει μια μικρή απόσταση μεταξύ του πάνω ορίου, του υφάσματος και της περιοχής ενδιαφέροντος αντίστοιχα, ώστε να είναι αποδοτικότερος ο αλγόριθμος εύρεσης ακμών.

Αφού επιλεχτεί η περιοχή ενδιαφέροντος, ανιχνεύεται το περίγραμμα του υφάσματος, και εντοπίζονται τα σημεία ελέγχου εμποτισμού της ρητίνης του υφάσματος και το μέγιστο όριο. Σε αυτό το διάστημα έχει σταματήσει η εκτέλεση του προγράμματος έτσι ώστε να προετοιμάσουμε το καλούπι για την διαδικασία VARI, τοποθετώντας τα διάφορα υλικά πάνω στο ύφασμα (αποκολλητικό ύφασμα, αφρώδες δικτυωτό πλέγμα, διάτρητο φιλμ), και τέλος τη σακούλα κενού σφραγίζοντάς την καλά. Επιπλέον στο δοχείο ρητίνης αναμιγνύουμε την κατάλληλη ποσότητα εποξικής ρητίνης με τον σκληρυντή. Αφού ελέγξουμε το μανόμετρο κενού της αντλίας κενού, ώστε η ένδειξη του να είναι γύρω στα 90 kPa για την δημιουργία ικανοποιητικού κενού, πατάμε Enter ώστε να συνεχιστεί η διαδικασία.

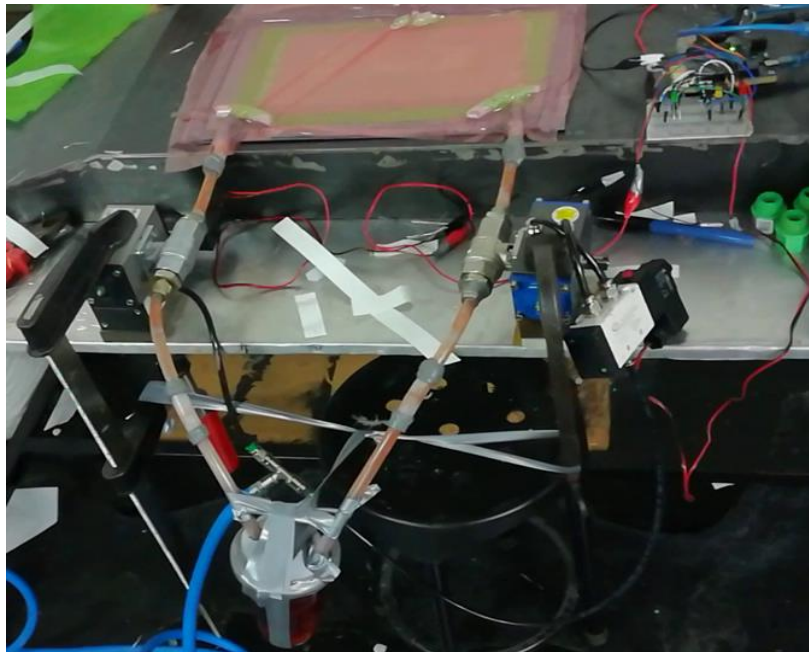
Οι διαστάσεις των υλικών που χρησιμοποιήθηκαν σε mm είναι: Υαλούφασμα 20 x 7.5, αποκολλητικό ύφασμα 26.5 x 12.5, πλέγμα 30 x 16, διάτρητο φιλμ 26.5 x 15.5, σακούλα κενού 45 x 35. Η εποξική ρητίνη που χρησιμοποιήθηκε είναι η R481/H50 με σκληρυντή 172012 της Fibermax με αναλογία 20%κ.β. Η συνολική ποσότητα που χρησιμοποιούμε είναι 144 g.



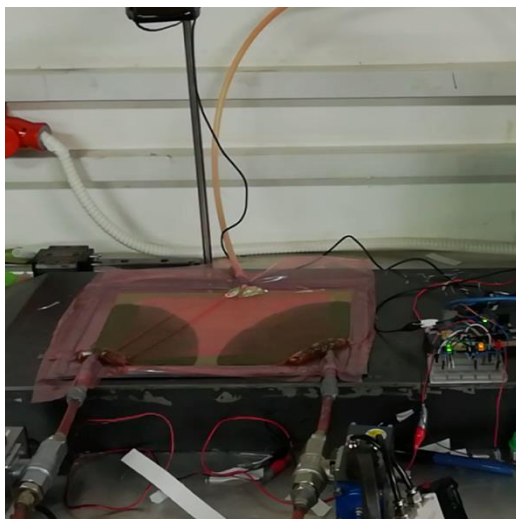
Εικόνα 62: Σημεία Ελέγχου, Μέγιστο Όριο Εντοπισμένου Υφάσματος / Αναμονή Εντολής Χρήστη Για Συνέχεια



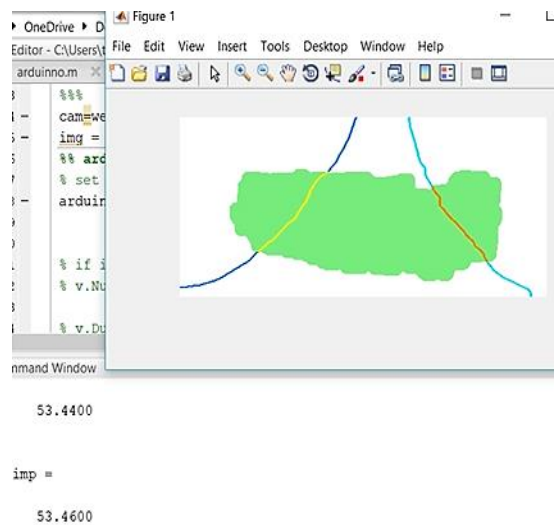
Εικόνα 63: Τα Υλικά που τοποθετούνται με διαδοχική σειρά/ πίεση κενού



Εικόνα 64: Διάταξη Αυτοματοποιημένης VARI Σε Αναμονή



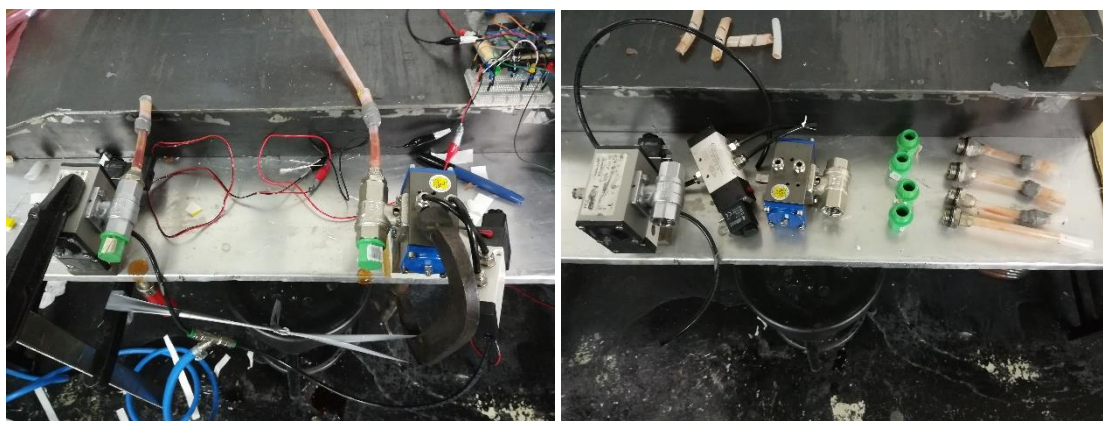
Εικόνα 65: Στιγμιότυπο Διαδικασίας Αυτοματοποιημένης VARI



Στην οθόνη του υπολογιστή εμφανίζεται το μέτωπο της ρητίνης και στο παράθυρο εντολών της MATLAB το ποσοστό εμποτισμού και ενημέρωση για τον αν οι βαλβίδες ανοιγοκλείνουν. Στην παραπάνω εικόνα βλέπουμε ένα στιγμιότυπο της διαδικασίας.

9.4 Καθαρισμός Βαλβίδων

Στο τέλος της διαδικασίας, αφού αποσυνδέσουμε το καλούπι με το εμποτισμένο πλέον ύφασμα, σημαντική διαδικασία είναι ο καθαρισμός των βαλβίδων και των εξαρτημάτων τους από την ρητίνη. Αυτή η διαδικασία εξαρτάται από την ρητίνη που χρησιμοποιούμε, αλλά καλό είναι να γίνει μέσα στο πρώτο εικοσάλεπτο για αποφυγή προβλημάτων. Η διαδικασία είναι η εξής: Ξεβιδώνουμε τα ρακόρ εισαγωγής και εξαγωγής της ρητίνης τοποθετώντας τα σε ένα δοχείο με ασετόν, βιδώνουμε τις τάπες στεγανότητας στις εισόδους της ρητίνης της κάθε βαλβίδας, έχοντας ανοιχτές τις βαλβίδες τοποθετούμε τον σωλήνα αναρρόφησης από την αντλία κενού στην έξοδο της κάθε βαλβίδας, ώστε να απομακρυνθεί η μεγάλη ποσότητα της ρητίνης. Στην συνέχεια γεμίζουμε τις βαλβίδες με ασετόν βιδώνοντας τις τάπες στεγανότητας. Ανοιγοκλείνουμε τις βαλβίδες αρκετές φορές και ξεβιδώνουμε τις τάπες για να φύγει η διαλυμένη ρητίνη με ασετόν. Επαναλαμβάνουμε την διαδικασία έως ότου δεν υπάρχουν υπολείμματα ρητίνης στο εσωτερικό των βαλβίδων.



Εικόνα 66: Καθαρισμός Βαλβίδων Από Την Ρητίνη

ΚΕΦΑΛΑΙΟ 10 Συμπεράσματα και Επεκτάσεις

10.1 Συμπεράσματα

Στη παρούσα διπλωματική η υλοποίηση του συστήματος παρακολούθησης και ρύθμισης διάταξης έγχυσης ρητίνης με τη βοήθεια κενού της κατεργασίας VARI στέφθηκε από απόλυτη επιτυχία. Τα βασικά σημεία για την υλοποίηση του συστήματος ήταν:

- Η ανάπτυξη αξιόπιστου κώδικα στην MATLAB για τον εντοπισμό του μετώπου της ρητίνης.
- Η ανάπτυξη αξιόπιστου κώδικα στην MATLAB για τον καθορισμό των κατάλληλων συνθηκών που πρέπει να ελέγχονται για την έγχυση της ρητίνης από τις βαλβίδες.
- Η ορθή επιλογή των βαλβίδων, καθώς χρησιμοποιείται κενό για την ροή της ρητίνης, και το ιξώδες του ρευστού που διαρρέει τις βαλβίδες είναι αρκετά υψηλό.

Έγινε προσπάθεια ώστε η αυτοματοποιημένη διαδικασία να μπορεί να ανταπεξέλθει σε φάσμα εφαρμογών. Παρόλα αυτά η κάθε εφαρμογή έχει τις ιδιαιτερότητές της, και με βάση αυτές πρέπει να διαμορφωθεί ο κώδικας κατάλληλα ώστε να έχουμε το επιθυμητό αποτέλεσμα.

Ο μέσος χρόνος εκτέλεσης ενός ολόκληρου κύκλου του κώδικα είναι 0.11 sec, σε υπολογιστή με τα εξής χαρακτηριστικά : επεξεργαστής Intel Core i7-7500U, μνήμη RAM DDR4 8 GB, κάρτα γραφικών AMD R7 445 αυτόνομης μνήμης 4 GB, λειτουργικό σύστημα Window 10 64bit. Η έκδοση της MATLAB που εκτελέστηκε ο κώδικας είναι η R2016a. Δεν παρατηρήθηκε σημαντική καθυστέρηση μεταξύ του frame που επεξεργάζεται ο πρόγραμμα σε σχέση με την θέση του μετώπου της ροής εκείνη τη στιγμή, τέτοια ώστε να είναι ικανή να δημιουργήσει πρόβλημα στον έλεγχο της ρητίνης.

Στην περίπτωση που η καθυστέρηση είναι μη αποδεκτή, αυτή μπορεί να μειωθεί ως ένα βαθμό μέσω του κώδικα που έχει γραφτεί. Παρόλα αυτά η MATLAB μέσω της εργαλειοθήκης του παράλληλου υπολογισμού (Parallel Computing Toolbox) δίνει την δυνατότητα να τρέξουμε το πρόγραμμα εξ ολοκλήρου σε κάρτα γραφικών αρχιτεκτονικής CUDA, με αποτέλεσμα να μειώνεται αισθητά το υπολογιστικό κόστος.

10.2 Μελλοντικές Επεκτάσεις

Το σύστημα θα μπορούσε να επεκταθεί χρησιμοποιώντας παράλληλα ειδικούς αισθητήρες όπως αισθητήρες υπερήχων, οπτικών ινών, μικρο-θερμοστοιχεία και άλλους, με τους οποίους θα παρακολουθείται η διαδικασία της σκλήρυνσης της ρητίνης.

Επιπλέον θα μπορούσαν να χρησιμοποιηθούν περισσότερες κάμερες και βαλβίδες είτε ως κύριες είτε ως βοηθητικές, για τον ταυτόχρονο εντοπισμό των μετώπων ροής της ρητίνης και τον έλεγχο της έγχυσης της ρητίνης για καλούπια τα οποία είναι αρκετά μεγάλα ή/και πολύπλοκα.

Βιβλιογραφία

- [1] T. H. Hou and B. J. Jensen, "Evaluation of Double-Vacuum-Bag Process For Composite Fabrication."
- [2] M. Moghaddam, A. Breede, C. Brauner, and W. Lang, "Embedding Piezoresistive Pressure Sensors to Obtain Online Pressure Profiles Inside Fiber Composite Laminates," *Sensors*, vol. 15, no. 4, pp. 7499–7511, Mar. 2015.
- [3] M. K. Moghaddam *et al.*, "Piezoresistive Pressure Sensors for Resin Flow Monitoring in Carbon Fibre-Reinforced Composite," *Proceedings*, vol. 1, no. 4, p. 339, Aug. 2017.
- [4] C. Di Fratta, F. Klunker, and P. Ermanni, "A methodology for flow-front estimation in LCM processes based on pressure sensors," *Compos. Part A Appl. Sci. Manuf.*, vol. 47, pp. 1–11, Apr. 2013.
- [5] U. K. Vaidya, N. C. Jadhav, M. V Hosur, J. W. Gillespie, and B. K. Fink, "Assessment of flow and cure monitoring using direct current and alternating current sensing in vacuum-assisted resin transfer molding," *Smart Mater. Struct.*, vol. 9, no. 6, pp. 727–736, Dec. 2000.
- [6] N. Gupta and R. Sundaram, "Fiber optic sensors for monitoring flow in vacuum enhanced resin infusion technology (VERITY) process," *Compos. Part A Appl. Sci. Manuf.*, vol. 40, no. 8, pp. 1065–1070, Aug. 2009.
- [7] T. Luthy and P. Ermanni, "Linear direct current sensing system for flow monitoring in Liquid Composite Moulding," *Compos. Part A Appl. Sci. Manuf.*, vol. 33, no. 3, pp. 385–397, Mar. 2002.
- [8] A. A. Skordos and I. K. Partridge, "A dielectric sensor for measuring flow in resin transfer moulding," 2000.
- [9] P. Barooah, J. Q. Sun, P. Barooah, and B. Berker, "Lineal Sensors for Liquid Injection Molding of Advanced Composite Materials Study on Vibration and Control of high-dimensional uncertain nonlinear dynamical systems based on discretized state space methods View project Predicting Surgical Outcomes In Rectus Transfer Surgeries View project Lineal Sensors for Liquid Injection Molding of Advanced Composite Materials," 2003.
- [10] M. Danisman, G. Tuncol, A. Kaynar, and E. M. Sozer, "Monitoring of resin flow in the resin transfer molding (RTM) process using point-voltage sensors," *Compos. Sci. Technol.*, vol. 67, no. 3–4, pp. 367–379, Mar. 2007.
- [11] A. Dominauskas, D. Heider, and J. W. Gillespie, "Electric time-domain reflectometry sensor for online flow sensing in liquid composite molding processing," *Compos. Part A Appl. Sci. Manuf.*, vol. 34, no. 1, pp. 67–74, Jan. 2003.
- [12] A. Dominauskas, D. Heider, and J. W. Gillespie, "Electric time-domain reflectometry distributed flow sensor," *Compos. Part A Appl. Sci. Manuf.*, vol. 38, no. 1, pp. 138–146, Jan. 2007.
- [13] E. Schmachtenberg, J. Schulte zur Heide, and J. Töpker, "Application of ultrasonics for the process control of Resin Transfer Moulding (RTM)," *Polym. Test.*, vol. 24, no. 3, pp. 330–338, May 2005.
- [14] D. D. Shepard and K. R. Smith, "Ultrasonic cure monitoring of advanced composites," *Sens. Rev.*, vol. 19, no. 3, pp. 187–194, 1999.
- [15] "On-line Process Monitoring of Thermosets by Ultrasonic Methods." [Online]. Available: <https://www.ndt.net/article/ecndt98/material/159/159.htm>. [Accessed: 02-Oct-2018].
- [16] S. T. Lim and W. Il Lee, "An analysis of the three-dimensional resin-transfer mold filling process," *Compos. Sci. Technol.*, vol. 60, no. 7, pp. 961–975, May 2000.
- [17] C. Lekakou, S. Cook, Y. Deng, T. W. Ang, and G. T. Reed, "Optical fibre sensor for

- monitoring flow and resin curing in composites manufacturing,” *Compos. Part A Appl. Sci. Manuf.*, vol. 37, no. 6, pp. 934–938, Jun. 2006.
- [18] G. Tuncol, M. Danisman, A. Kaynar, and E. M. Sozer, “Constraints on monitoring resin flow in the resin transfer molding (RTM) process by using thermocouple sensors,” *Compos. Part A Appl. Sci. Manuf.*, vol. 38, no. 5, pp. 1363–1386, May 2007.
- [19] A. Aktas, S. W. Boyd, and R. A. Shenoi, “ECCM15-15 TH EUROPEAN CONFERENCE MEASUREMENT OF PERMEABILITY AND CURE USING THERMOCOUPLES IN THE VACUUM ASSISTED RESIN INFUSION PROCESS TO AID SIMULATION.”
- [20] “Arduino Playground - ArduinoPinCurrentLimitations.” [Online]. Available: <https://playground.arduino.cc/Main/ArduinoPinCurrentLimitations>. [Accessed: 10-Oct-2018].
- [21] “Transistors - learn.sparkfun.com.” [Online]. Available: <https://learn.sparkfun.com/tutorials/transistors/applications-i-switches>. [Accessed: 12-Oct-2018].
- [22] “Diodes - learn.sparkfun.com.” [Online]. Available: <https://learn.sparkfun.com/tutorials/diodes/all>. [Accessed: 12-Oct-2018].
- [23] “Direct Acting vs. Pilot Operated Solenoid Valve | ATO.com.” [Online]. Available: <https://www.ato.com/direct-acting-vs-pilot-operated-solenoid-valve>. [Accessed: 18-Oct-2018].
- [24] “Coaxial Valves vs. Ball Valves for On/Off Applications - Assured Automation.” [Online]. Available: <https://assuredautomation.com/news-and-training/coaxial-valves-vs-ball-valves-for-on-off/>. [Accessed: 19-Oct-2018].