



ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ – ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΒΙΟΪΑΤΡΙΚΗΣ

**ΔΗΜΙΟΥΡΓΙΑ ΚΑΙ ΑΝΑΠΤΥΞΗ
ΚΩΔΙΚΑ ΣΕ ΓΛΩΣΣΑ ΡΥΘΜΟΝ ΓΙΑ
ΤΟΝ ΧΕΙΡΙΣΜΟ ΡΟΜΠΟΤΙΚΟΥ
ΒΡΑΧΙΟΝΑ ΜΕ ΤΗΝ ΧΡΗΣΗ WEB-
CAMERA**

ΘΩΜΟΠΟΥΛΟΣ ΔΙΟΝΥΣΙΟΣ

Αριθμός Μητρώου: **48014027**

Επιβλέπων Καθηγητής
Παντελής Ασβεστάς, Αναπληρωτής Καθηγητής

Αθήνα 21/10/2021

Η Τριμελής Εξεταστική Επιτροπή

Ο Επιβλέπων Καθηγητής

Παντελεήμων Ασβεστάς

Αναπληρωτής Καθηγητής

Δημήτριος Γκλώτσος

Αναπληρωτής Καθηγητής

Σπυρίδων Κωστόπουλος

Επίκουρος Καθηγητής

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΠΤΥΧΙΑΚΗΣ/ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Ο υπογράφων **ΘΩΜΟΠΟΥΛΟΣ ΔΙΟΝΥΣΙΟΣ** του **ΒΑΣΙΛΕΙΟΥ**, με αριθμό μητρώου **48014027** φοιτητής του Τμήματος **ΜΗΧΑΝΙΚΩΝ ΒΙΟΙΑΤΡΙΚΗΣ** της Σχολής **ΜΗΧΑΝΙΚΩΝ** του Πανεπιστημίου Δυτικής Αττικής, δηλώνω υπεύθυνα ότι:

«Είμαι συγγραφέας αυτής της πτυχιακής/διπλωματικής εργασίας και κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου/διπλώματός μου».

Ημερομηνία

Ο/Η Δηλών/ούσα

21-10-2021



Περίληψη

Η παρούσα διπλωματική εργασία αποτελεί μία ιδέα ένωσης της κοινής ψηφιακής τεχνολογίας, με έναν ρομποτικό βραχίονα για την επίτευξη στόχων πάνω στον χώρο της Βιοϊατρικής Τεχνολογίας. Η τεχνολογία πλέον ακμάζει καθώς έχουμε πλέον τα μέσα να την χειριστούμε με πολλούς τρόπους. Αυτό μας εξυπηρετεί στο να πραγματοποιούμε τις ιδέες μας αλλά και να κάνουμε την ζωή μας ακόμη πιο εύκολη με λιγότερους κινδύνους.

Η ιδέα της παρούσας διπλωματικής ξεκίνησε από την αναζήτηση μου για εργασία πάνω στον χώρο της Βιοϊατρικής τεχνολογίας. Σε μία περίοδο παγκόσμιας επιδημιολογικής κρίσης του ιού Sars Cov 2 ή αλλιώς ευρέως γνωστός ως Covid-19, οι δουλειές στέρευαν και ήταν δύσκολο να απορροφηθώ. Κάθε περίπτωση όμως έχει και τα θετικά της μηνύματα. Για εμένα, ήταν πως πέρασα από διάφορες έμπρακτες συνεντεύξεις, μία από τις οποίες ήταν η αφαίρεση του θυροειδή αδένου από έναν ασθενή. Στο χειρουργείο εντόπισα διάφορες διαδικασίες προκειμένου να επιτευχθεί αυτός ο στόχος και έτσι σκέφτηκα το πόσο χρήσιμο θα ήταν να υπήρχε ένας αποστειρωμένος ρομποτικός βραχίονας υψηλής ακριβείας για την ασφαλή επιλογή και τοποθέτηση του αφαιρεμένου αδένου σε ένα ασφαλές σημείο για την σωστή του βιοψία. Όλα αυτά μπορούν να συμβούν όταν έχουμε στην κατοχή μας έναν ρομποτικό βραχίονα που έχει τουλάχιστον τρεις σερβοκινητήρες ακριβείας και γάντζο για την επιλογή του αντικειμένου στον χώρο, έναν υπολογιστή και μία κάμερα. Έτσι λοιπόν, η παρούσα διπλωματική άσκηση είναι μία δημιουργία ενός κώδικα μέσω της γλώσσας προγραμματισμού Python για την εύρυθμη λειτουργία του ρομποτικού βραχίονα σύμφωνα με τις ανάγκες του χειριστή. Ο χειριστής καλείται να επιλέξει το αντικείμενο ενδιαφέροντος του από το πεδίο όρασης (Field Of View | FOV) που έχει από την κάμερα και την επιβεβαίωση του προκειμένου να επιλεγεί από τον ρομποτικό βραχίονα και να μεταφερθεί με ασφάλεια στο σημείο που έχει προκαθοριστεί σαν σημείο ασφάλειας.

Η διαδικασία αυτή βρίσκεται σε πρώιμο στάδιο για την παρούσα φάση αλλά «καλλιεργείται» στην βασική ιδέα της αυτοματοποίησης και της αποφυγής των ανεπιθύμητων ενεργειών του ανθρώπινου παράγοντα. Στην ανάπτυξη του κώδικα, θα ήταν χρήσιμη η ανάδραση του κώδικα ως προς την επιλογή, ο αυτόματος εντοπισμός του αντικειμένου με την κατάλληλη ένδειξη του στην εικόνα και πολλά ακόμα. Η τεχνολογία είναι ένα μέσο το οποίο μπορεί να μας δώσει πολλά υπέρ αν την χρησιμοποιήσουμε κατάλληλα.

Abstract

This dissertation is an idea of combining common digital technology, with a robotic arm to achieve goals in the field of Biomedical Technology. Technology is now thriving as we now have the means to handle it in many ways. This helps us to apply our ideas but also to make our life even easier with less risks.

The idea of this diploma started from my search for a job in the field of Biomedical technology. At a time of global epidemic crisis of the Sars Cov 2 virus or otherwise known as Covid-19, jobs were running low and it was difficult for me to absorb. But each case has its positive messages. For me, it was that I went through various practical interviews, one of which was the removal of the thyroid gland from a patient. In the operating room I found various procedures to achieve this goal and so I thought how useful it would be to have a high precision sterile robotic arm to safely select and place the removed gland in a safe place for proper biopsy. All this can happen when we have a robotic arm that has at least three precision servomotors and a hook for selecting the object in space, a computer and a camera. Thus, this dissertation is a creation of a code through the Python programming language for the smooth operation of the robotic arm according to the needs of the operator. The operator is asked to select the object of interest from the field of view (FOV) he has from the camera and his confirmation in order to be selected by the robotic arm and to be transported safely to the point designated as a safety point.

This process is at an early stage for the present phase but is "cultivated" in the basic idea of automation and the avoidance of side effects of the human agent. In the development of the code, the feedback of the code in terms of selection, the automatic detection of the object with its appropriate indication in the image and much more would be useful. Technology is a tool that can give us a lot of benefits if we use it properly.

Ευχαριστίες:

Στα πλαίσια της ανάπτυξης της παρούσας διπλωματικής εργασίας, μου δόθηκε ο απαραίτητος εξοπλισμός ώστε να εκτελεσθεί το έργο μου. Συγκεκριμένα, μου δόθηκε ο ρομποτικός βραχίονας «Mover6» της εταιρείας «CommonPlace Robotics GmbH», το οποίο αποτελεί περιουσία του Πανεπιστημίου Δυτικής Αττικής (Π.Α.Δ.Α.) και συγκεκριμένα του τμήματος Μηχανικών Βιοϊατρικής. Επίσης, θα ήθελα να ευχαριστήσω το Πανεπιστήμιο, το τμήμα των Μηχανικών Βιοϊατρικής καθώς και τα μέλη τα οποία το απαρτίζουν καθώς μου παραχωρήθηκε ο απαραίτητος χώρος και οι απαραίτητες τεχνολογικές συσκευές προκειμένου να καταφέρω να προχωρήσω την διπλωματική μου εργασία με όλα τα απαραίτητα μέσα τα οποία χρειάζονται.

Περιεχόμενα

Περίληψη.....	4
Abstract	5
Εισαγωγή.....	10
Σκοπός της διπλωματικής εργασίας	10
1 Ιστορικό υπόβαθρο των ρομποτικών συστημάτων	12
1.1 - Η πρώτη γενιά ρομποτικών συστημάτων.....	12
1.2 - Η δεύτερη γενιά ρομποτικών συστημάτων	13
1.3 - Η τρίτη γενιά ρομποτικών συστημάτων.....	15
2 Ρομποτική στην ιατρική	16
2.1 - Η αρχή των ρομποτικών συστημάτων στην υγεία	16
2.2 - Ρομποτική στον χώρο της ορθοπαιδικής	18
2.3 - Επεμβατικές ή Μη Επεμβατικές χειρουργικές επεμβάσεις.....	19
2.4 - Εφαρμογές της ρομποτικής στην νευροχειρουργική.....	22
2.5 - Εφαρμογές της ρομποτικής στην ακτινοθεραπεία	23
2.6 - Αποκατάσταση.....	24
3 Έλεγχος οπτικής οδήγησης ρομπότ.....	26
4 Γιατί ρομποτική στον ιατρικό χώρο;	30
4.1 - Βασικές προδιαγραφές των ρομπότ για ιατρική χρήση	30
Ειδικό μέρος – Σχεδιασμός & Εκτέλεση.....	32
1 Ρομποτικό σύστημα Mover 6 της Commonplace Robotics GmbH.....	32
1.1 Εγκατάσταση και καλωδίωση	32
1.2 Ρομποτικό σύστημα.....	37
1.3 Τελικός επενεργητής	39
1.4 Προγραμματιστικό περιβάλλον CPRog	40
1.5 – Ένωση ψηφιακού και πραγματικού κόσμου	50
2. Προγραμματισμός και δημιουργία script	61
1. Προγραμματιστικό περιβάλλον PyCharm και ασφαλής εγκατάσταση	61
2. Κώδικας της παρούσας διπλωματικής εργασίας.....	69
3. Αποτελέσματα και Συμπεράσματα.....	79
Βιβλιογραφία.....	80

Εικόνα 1 - Αυτοματοποιημένο ρομπότ οικιακού καθαρισμού.....	10
Εικόνα 2 - Αυτοματοποιημένοι ρομποτικοί βραχίονες για βιομηχανικούς σκοπούς.....	11
Εικόνα 3 - Ρομπότ Unimate της Unimation	13
Εικόνα 4 - Ρομπότ WABOT-1 (αριστερά), Ρομπότ WABOT-2 (δεξιά).....	14
Εικόνα 5 - Δύο πρωτότυπα ρομποτικά συστήματα τύπου SCARA του καθηγητή Hiroshi Makino.....	14
Εικόνα 6 – Ρομποτικό όχημα Sojourner της NASA (Πηγή: [5])	15
Εικόνα 7 - Honda Asimo robot (last version) (Πηγή: [6])	15
Εικόνα 8 - Σύγχρονο ρομποτικό χειρουργείο.....	16
Εικόνα 9 - Ρομποτικό σύστημα Arhtrobot (Πηγή: [7]).....	17
Εικόνα 10 - Ρομποτικό σύστημα κατά την πρώτη χειρουργική επέμβαση	17
Εικόνα 11 - TSolution-One	18
Εικόνα 12 - TPLAN 3D Planning Workstation (Workstation για την δημιουργία του 3D μοντέλου της περιοχής ενδιαφέροντος).....	19
Εικόνα 13 - Σύστημα ZEUS της Computer Motion: Χειρισμός διαδικασίας από τον χειρουργό (αριστερή διάταξη) , ρομποτικοί βραχίονες (δεξιά διάταξη)	20
Εικόνα 14 - Ρομποτικό Σύστημα da Vinci	20
Εικόνα 15 - Telelap ALF-X.....	21
Εικόνα 16 - Flex Robot System by Medrobotics (Αριστερά: η ολική διάταξη του ρομποτικού συστήματος, Δεξιά: Ο πολυαρθρωτός σωλήνας εισχώρησης στον ασθενή).....	22
Εικόνα 17 - Συμβατική νευροχειρουργική (αριστερά) Ρομποτική Νευροχειρουργική (Δεξιά) - (Πηγή: Renishaw).....	23
Εικόνα 18 - CyberKnife (Πηγή Wikipedia)	24
Εικόνα 19 - Hocoma Lokomat® σύστημα αποκατάστασης κάτω άκρων.....	25
Εικόνα 20 - ABB 3D inspection robot (Πηγή: [22])	28
Εικόνα 21 - Καλωδίωση ρομποτικού συστήματος.....	32
Εικόνα 22 - Συνδεσμολογία (Τροφοδοτικό).....	33
Εικόνα 23 - Συνδεσμολογία (Κουμπί άμεσης διακοπής λειτουργίας).....	34
Εικόνα 24 - Συνδεσμολογία (Καλώδιο προς υπολογιστή)	35
Εικόνα 25 - Ειδικό καλώδιο επικοινωνίας	36
Εικόνα 26 - Σύνδεση ειδικού μετατροπέα με το ρομποτικό σύστημα (αριστερά) και σύνδεση με USB θύρα στον υπολογιστή (Δεξιά)	36
Εικόνα 27 - Εκπαιδευτικός ρομποτικός βραχίονας Mover 6 της εταιρείας Commonplace Robotics (Με μπλε επισημαίνονται οι απλοί κινητήρες servo και με κόκκινο χρώμα επισημαίνονται οι έξυπνοι κινητήρες servo)	37
Εικόνα 28 - CPRog program, Robot Control Environment	38
Εικόνα 29 - Τελικός επενεργητής του Mover 6 (gripper)	39
Εικόνα 30 - Οδηγός εγκατάστασης προγράμματος CPRog	40
Εικόνα 31 - Εικονίδιο CPRog	41
Εικόνα 32 - Προγραμματιστικό περιβάλλον του CPRog	41
Εικόνα 33 - CPRog (Go to file).....	42
Εικόνα 34 - CPRog (Robotic System options).....	42
Εικόνα 35 - CPRog (Activity troubleshooting).....	43
Εικόνα 36 - CPRog (Motion)	44

Εικόνα 37 - CPRog (Motion interface explanation).....	44
Εικόνα 38 - CPRog (Motion Type)	45
Εικόνα 39 - CPRog (Replay Mode)	45
Εικόνα 40 - CPRog (New program script)	46
Εικόνα 41 - CPRog (Log messages).....	46
Εικόνα 42 - CPRog (Infocenter).....	47
Εικόνα 43 - CPRog (Jogging)	47
Εικόνα 44 - CPRog (Input/Output)	47
Εικόνα 45 - CPRog (Programs & Variables)	47
Εικόνα 46 - CPRog (Camera).....	48
Εικόνα 47 - CPRog (Scene).....	48
Εικόνα 48 - CPRog (Scene explanation).....	49
Εικόνα 49 - Πραγματικοί άξονες (Μπλέ άξονας: X (μήκος), Πράσινος άξονας: Y (ύψος), Κόκκινος άξονας: Z(πλάτος).....	51
Εικόνα 50 - Πρότυπο αξόνων του προγράμματος CPRog	52
Εικόνα 51 - Χώρος εργασίας ανθρωπομορφικού βραχίονα	53
Εικόνα 52 - Υπολογισμοί για τον X άξονα (απομάκρυνσης)	55
Εικόνα 53 - Γεωμετρικός χώρος εργασίας	56
Εικόνα 54 - Παράδειγμα στον γεωμετρικό χώρο εργασίας	57
Εικόνα 55.....	59
Εικόνα 56 - Python logo.....	61
Εικόνα 57 - Πρώτη επαφή με το PyCharm	63
Εικόνα 58 - Δημιουργώντας νέο project στο PyCharm.....	63
Εικόνα 59 - Προγραμματιστικό περιβάλλον του PyCharm.....	64
Εικόνα 60 - PyCharm (File, New)	65
Εικόνα 61 - PyCharm (File, New, Python File)	65
Εικόνα 62 - PyCharm (File, New, Python File, Name).....	66
Εικόνα 63 - PyCharm (New File Created!)	66
Εικόνα 64 - Γράφουμε την εντολή στο πεδίο προγραμματισμού.....	67
Εικόνα 65 - Εκτελούμε την πρώτη φορά με Run - Run	67
Εικόνα 66 - Επιλέγουμε να εκτελεστεί η εντολή μέσα στο αρχείο όπου γράψαμε τον κώδικα (Diplomatiki Ergasia στην συγκεκριμένη περίπτωση).....	68
Εικόνα 67 - Το αποτέλεσμα μας έχει εμφανιστεί (εκτυπωθεί) στο κάτω μέρος του προγράμματος μας.....	68
Table 1 Εισαγωγή του κώδικα.....	69
Table 2 - Ορισμός εντολών εντοπισμού αντικειμένου	70
Table 3 Ορισμός ελάχιστων τιμών και διαδικασία βαθμονόμησης υψηλότερων τιμών.	70
Table 4 Ορισμός ψηφιακών και πραγματικών τιμών	71
Table 5 Βασική εντολή του κώδικά μας για την διαδικασία του εντοπισμού.....	72
Table 6 Διαδικασία μετατροπής ψηφιακών τιμών σε πραγματικές τιμές	73

Εισαγωγή

Σκοπός της διπλωματικής εργασίας

Η ρομποτική αποτελεί ένα ευρέως αναπτυσσόμενο αντικείμενο που παίρνει μέρος τόσο σε απλές και καθημερινές διαδικασίες όσο σε σημαντικές και ευμεγέθους βιομηχανίες με σκοπό την παραγωγή αγαθών που προορίζονται στο καταναλωτικό κοινό. Η ρομποτική (Robotics) ασχολείται με την μελέτη εκείνων των μηχανών που μπορούν να αντικαταστήσουν τον άνθρωπο στην εκτέλεση μίας εργασίας, όσον αφορά τόσο φυσικές δραστηριότητες όσο και λήψεις αποφάσεων (Πηγή: [1]). Πλέον, στην καθημερινότητά μας, είναι εύκολο να διακρίνουμε διατάξεις σαν αυτές. Συγκεκριμένα, σε μία μέση κατοικία, θα εντοπίσουμε συσκευές όπως το πλυντήριο ρούχων ή το πλυντήριο πιάτων, τον υπολογιστή ή το κινητό τηλέφωνο καθώς και τις αυτόματες και αυτοκινούμενες συσκευές καθαρισμού του δαπέδου του σπιτιού (Εικόνα 1). Διατάξεις σαν αυτές αποτελούν καθημερινές συσκευές ρομποτικού περιεχομένου με σκοπό την οικιακή τους χρήση για την διευκόλυνση του ανθρώπου.



Εικόνα 1 - Αυτοματοποιημένο ρομπότ οικιακού καθαρισμού

Αντιθέτως, υπάρχουν ρομποτικά συστήματα τα οποία δεν συναντώνται σε μία μέση κατοικία αλλά σε μεγάλα εργοστάσια ή βιομηχανίες. Τέτοια παραδείγματα αποτελούν τα εργοστάσια παραγωγής αυτοκινήτων και λοιπών οχημάτων (Εικόνα 2), εργοστάσια παραγωγής τροφίμων, ρούχων, είδη πρώτης ανάγκης, καθημερινών αντικειμένων, φαρμακευτικού εξοπλισμού κ.α. .



Εικόνα 2 - Αυτοματοποιημένοι ρομποτικοί βραχίονες για βιομηχανικούς σκοπούς

Όμως, η χρήση των ρομποτικών συστημάτων δεν σταματάει εκεί. Ρομποτικά συστήματα μπορούν να συναντηθούν και σε χώρους της ιατρικής τα οποία έχουν ενισχύσει σημαντικά την βεβαιότητα των χειρουργείων αλλά και την αποτελεσματικότητά τους.

Αυτός ο λόγος είναι και η αιτία της ανάπτυξης της παρούσας διπλωματικής εργασίας, η περαιτέρω ανάπτυξη ενός έτοιμου ρομποτικού βραχίονα της γερμανικής εταιρείας “[CommonPlace Robotics GmbH](#)” με πρόσθεση το χαρακτηριστικό της ρομποτικής όρασης με σκοπό την διευκόλυνση διαδικασιών σε έναν ιατρικό χώρο όπως το χειρουργείο. Συγκεκριμένα, θα ασχοληθούμε με τον ρομποτικό βραχίονα “[Mover6](#)” ο οποίος - σύμφωνα με την επίσημη ιστοσελίδα του βραχίονα - προσδιορίζεται για εκπαιδευτικούς σκοπούς, είναι ένας ρομποτικός βραχίονας που αποτελείται από 6 κινηματικούς άξονες που με αυτόν τον τρόπο επιτρέπει στον τελικό επενεργητή (λαβή) να κινείται και στους έξι (6) βαθμούς ελευθερίας. Συνδυάζοντας την φυσική, τα μαθηματικά και την επιστήμη των υπολογιστών μπορούμε να τον αξιοποιήσουμε σε διάφορες εφαρμογές. Η ακτίνα κίνησής του αγγίζει τα 60 εκατοστά (cm) και μπορεί να ανυψώσει αντικείμενα με μέγιστο βάρος τα 400 γραμμάρια (g).

Στη συγκεκριμένη διπλωματική, το ρομποτικό σύστημα βρίσκεται στην κατοχή του τμήματος Μηχανικών Βιοϊατρικής, και η ανάπτυξή του θα γίνεται με την φυσική παρουσία του φοιτητή που έχει αναλάβει την διπλωματική εργασία στον χώρο του πανεπιστημίου. Συγκεκριμένα, ο φοιτητής καλείται να το συναρμολογήσει από την αρχή, να το προσαρμόσει στον χώρο σύμφωνα με τις οδηγίες του κατασκευαστή τηρώντας τα ασφαλιστικά μέτρα τα οποία εκείνος έχει ορίσει στο επίσημο εγχειρίδιο του ρομποτικού συστήματος, να εκπαιδευτεί μέσω έρευνας και εφαρμογών πάνω στο λογισμικό (Software) του συγκεκριμένου μηχανήματος, να πραγματοποιήσει την απαραίτητη έρευνα της συμβατότητας των υλικών που θα θελήσουμε να προσαρμοστούν στον ρομποτικό βραχίονα και να καταφύγουμε στο τελικό συμπέρασμα της διπλωματικής.

Συγκεκριμένα, με την χρήση του συγκεκριμένου ρομποτικού συστήματος, θα προσπαθήσουμε να εντάξουμε την ρομποτική όραση μέσω μίας κάμερας που θα συνδέσουμε στον ρομποτικό βραχίονα ή τον υπολογιστή ώστε με την ρομποτική όραση να δημιουργήσουμε την ικανότητα του συστήματος να αναγνωρίζει πρότυπα τα οποία θέλουμε να αναγνωρίσει, να τα συλλέξει και να τα τοποθετεί σε συγκεκριμένα σημεία. Τέτοια αντικείμενα, μπορούν να αποτελέσουν τα χρησιμοποιημένα χειρουργικά εργαλεία τα οποία θα συλλέγονται και θα τοποθετούνται σε κλίβανο αποστείρωσης για την επόμενη υπόθεση στο χειρουργείο.

1 Ιστορικό υπόβαθρο των ρομποτικών συστημάτων

1.1 - Η πρώτη γενιά ρομποτικών συστημάτων

Θα έλεγε κανείς πως η ρομποτική είναι ένας τομέας όπου οι ρίζες του είναι αρκετά σύγχρονες και μικρές. Αντιθέτως όμως, η ρομποτική έχει βαθιές πολιτιστικές ρίζες. Συγκεκριμένα, παρατηρούμε μία επιρροή από την αρχαία Ελληνική Μυθολογία και συγκεκριμένα στον Θρύλο του Τιτάνα Προμηθέα, του χάλκινου σκλάβου που σφυρηλάτησε ο Ήφαιστος, είναι μία αρχή η οποία τροφοδότησε την ιδέα του Φράνκενσταϊν στη σύγχρονη εποχή.

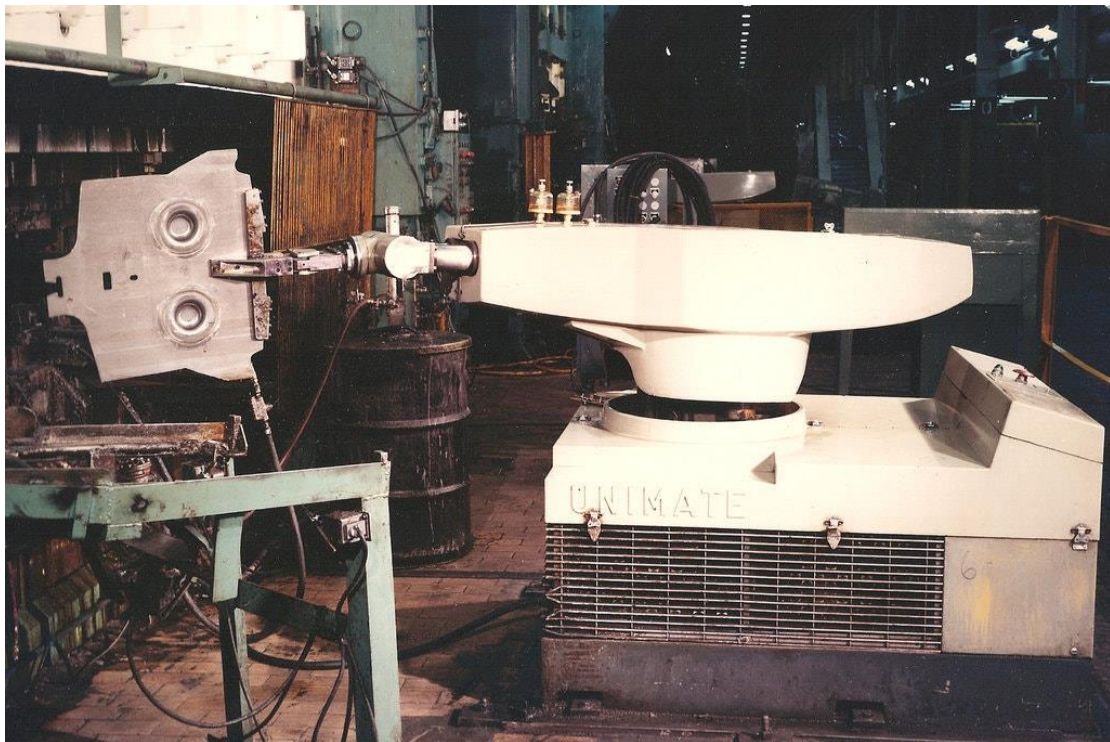
Όπως αναφέραμε και σε προηγούμενο κεφάλαιο, η ρομποτική ασχολείται με την μελέτη και ανάπτυξη μηχανών που μπορούν να αντικαταστήσουν τον άνθρωπο στην εκτέλεση μίας εργασίας, τόσο σε φυσικές δραστηριότητες όσο και στην λήψη των αποφάσεων. Ανέκαθεν, μία από τις μεγαλύτερες φιλοδοξίες του ανθρώπου, ήταν να δώσει ζωή στα αντικείμενα τα οποία έφτιαχνε. Κατά το 1920 μ.Χ. , ο Τσέχος συγγραφέας *Karel Capek*, επηρεασμένος από την Βιομηχανική εποχή, έγραψε το θεατρικό του έργο «Τα παγκόσμια ρομπότ του Rossum» (Rossum's Universal Robots -R.U.R.). Αυτή ήταν η αφορμή της έκδοσης της λέξης ρομπότ (robot) η οποία παράγεται από τον όρο “robota” που σημαίνει «εξαναγκασμένη εργασία» σε Σλαβικές γλώσσες. Πρόκειται για ένα διήγημα επιστημονικής φαντασίας το οποίο απεικονίζει τις τάσεις των ρομπότ να στρέφονται ενάντια στον άνθρωπο.

Πέραν όμως αυτής της οργανικής παρουσίας των ρομπότ, άρχισε η εικόνα του ρομπότ ως μηχανικό σύστημα. Η ιδέα άρχισε το 1940 μ.Χ. από τον γνωστό Ρώσο συγγραφέα *Isaac Asimov*, ο οποίος σκέφτηκε το ρομπότ ως ένα αντικείμενο σε σχήμα ανθρώπου αλλά χωρίς την ικανότητα της εκδήλωσης ή ύπαρξης συναισθημάτων. Ο όρος «ρομποτική» παρουσιάστηκε από τον Asimov ως εκείνη την επιστήμη που μελετάει τα ρομπότ βασιζόμενη σε τρεις βασικούς νόμους. Αυτοί είναι:

- Ένα ρομπότ δεν μπορεί να τραυματίσει ή, μέσω αδράνειας, να βλάψει ένα ανθρώπινο όν.
- Ένα ρομπότ επιβάλλεται να ακούει στις εντολές του ανθρώπου, εκτός εάν αυτές οι εντολές έρχονται σε αντίθεση με τον πρώτο κανόνα.
- Ένα ρομπότ θα πρέπει να προστατεύει την ίδια του την ύπαρξη, χωρίς όμως αυτό να σημαίνει πως μπορεί να προβεί τους δύο κανόνες που αναφέρθηκαν προηγουμένως.

Αυτοί αποτελούν τους τρεις βασικούς νόμους συμπεριφοράς που ένα ρομπότ θα πρέπει να ακολουθεί προκειμένου να λειτουργεί με ασφάλεια επιτυγχάνοντας τις ανάγκες του ανθρώπου χωρίς να τον βλάψει.

Το 1954 μ.Χ., ο Αμερικανός εφευρέτης *George Charles Devol Jr.*, αιτήθηκε κατοχύρωση ευρεσιτεχνίας ως πατέντα, για το πρώτο προγραμματιζόμενο ρομπότ. Η αίτηση του έγινε δεκτή το 1961 και έναν χρόνο αργότερα, σε συνεργασία με τον *Joseph Engelberger* δημιούργησαν την πρώτη εταιρεία ρομποτικής, την “Unimation”. Η εταιρεία αυτή δημιούργησε το πρώτο ρομπότ το οποίο το χρησιμοποίησαν στην εταιρεία General Motors για την παραγωγή, το “Unimate” (Εικόνα 3). Η χρήση του στο εργοστάσιο, ήταν η μεταφορά μετάλλου σε υψηλή θερμοκρασία και την τοποθέτησή του σε ψυκτικό υγρό. Η επιτυχία του κρίθηκε και από την μετέπειτα χρήση του και από άλλα εργοστάσια. Από αυτό το ρομπότ και έπειτα, η ίδια η λέξη ρομπότ αρχίζει και παίρνει μία διαφορετική όψη από αυτή της ανθρώπινης.

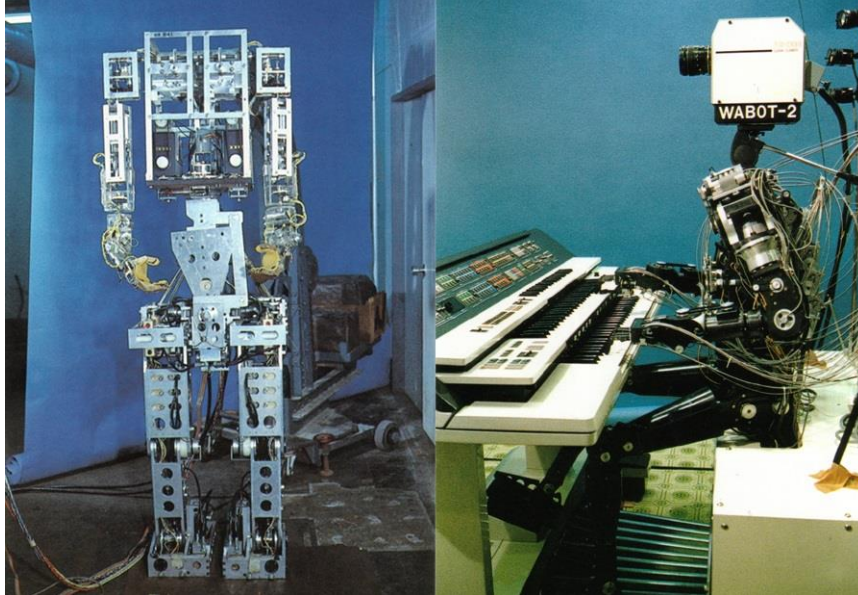


Εικόνα 3 - Ρομπότ Unimate της Unimation

1.2 - Η δεύτερη γενιά ρομποτικών συστημάτων

Το 1970, με την ένωση τεσσάρων εργαστηρίων του Πανεπιστημίου Επιστήμης και Μηχανικής της Waseda (School of Science & Engineering of Waseda University) άρχισε η κατασκευή του «Βιομηχανικού γκρούπ» (The Bio-Engineering group) με σκοπό την δημιουργία ενός ανθρωπόμορφου ρομπότ, το WABOT-1 (WASeda roBOT). Το έτος παραγωγής του WABOT-1, έγινε το 1973. Το ρομπότ αυτό εκτελούσε κινήσεις όμοιες με αυτές του ανθρώπου, μπορούσε να επικοινωνήσει με έναν άνθρωπο μέσω της ανθρώπινης ομιλίας στην ιαπωνική γλώσσα και να μετρήσει αποστάσεις και κατευθύνσεις. Έπειτα, ακολούθησε κι

άλλο ένα παρόμοιο αλλά εξελιγμένο μοντέλο της WABOT, το WABOT-2 (Εικόνα 4). Εξίσου ανθρωπόμορφο αλλά πιο εξελιγμένο από την άποψη της ρομποτικής όρασης, της επεξεργασίας αυτής και της επαφής του με την μουσική (Πηγή: [2]). Το παρόν δεν αποτελεί ρομπότ βιομηχανικού χαρακτήρα αλλά είναι ένα από τα ρομπότ που εισάγουν πολλές πληροφορίες για μελλοντικά σχέδια και ανάγκες του ανθρώπου.



Εικόνα 4 - Ρομπότ WABOT-1 (αριστερά), Ρομπότ WABOT-2 (δεξιά)

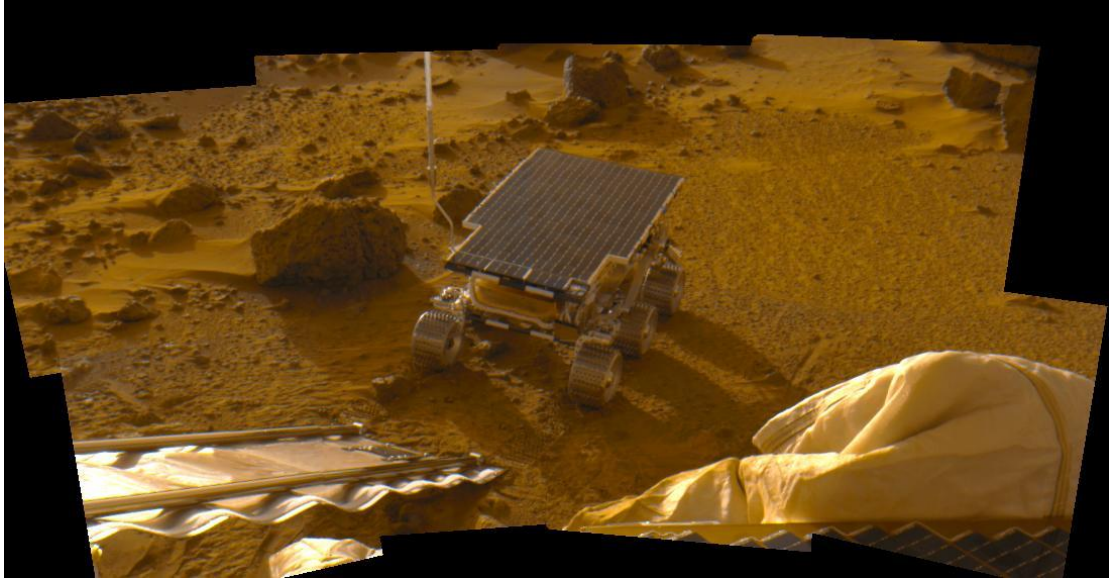
Ακόμη, το έτος 1978 στο γραφείο του καθηγητή Hiroshi Makino στο Πανεπιστήμιο της Yamanashi της Ιαπωνίας (Yamanashi University in Japan), δημιουργήθηκε ο πρώτος ρομποτικός βραχίονας τύπου SCARA (Selective Compliance Assembly Robot Arm). Ο συγκεκριμένος ρομποτικός βραχίονας είχε την ικανότητα να κινηθεί στον χώρο επιλεκτικά εκμεταλλευόμενος τους τρεις άξονες κίνησης (X, Y και Z άξονες) (Πηγές: [3,4]).



Εικόνα 5 - Δύο πρωτότυπα ρομποτικά συστήματα τύπου SCARA του καθηγητή Hiroshi Makino

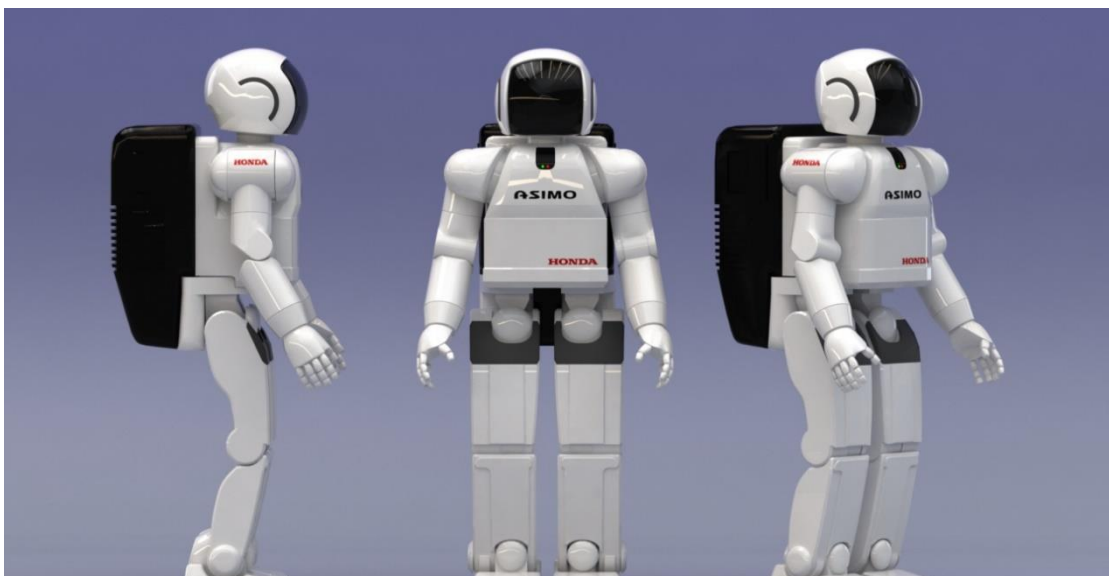
1.3 - Η τρίτη γενιά ρομποτικών συστημάτων

Η ρομποτική έως εδώ έδειξε τις ικανότητές με την παρουσία της στις βιομηχανικές εγκαταστάσεις ανεβάζοντας την παραγωγή, χαμηλώνοντας το τελικό κόστος στον καταναλωτή και δημιουργώντας ένα πολύ πιο ασφαλές κλίμα εργασίας για τον ανθρώπινο παράγοντα. Όμως, τα πλεονεκτήματα αυτά κέντρισαν το ενδιαφέρον και άλλων τομέων. Το 1997 δημιουργήθηκε το ρομποτικό όχημα «Sojourner» (Εικόνα 6) το οποίο το χειρίστηκε η NASA στην επιχείρησή της για αναγνώριση του πλανήτη Άρη στις 4 Ιουλίου του 1997 με μεγάλη επιτυχία (Πηγή: [5]).



Εικόνα 6 – Ρομποτικό όχημα Sojourner της NASA (Πηγή: [5])

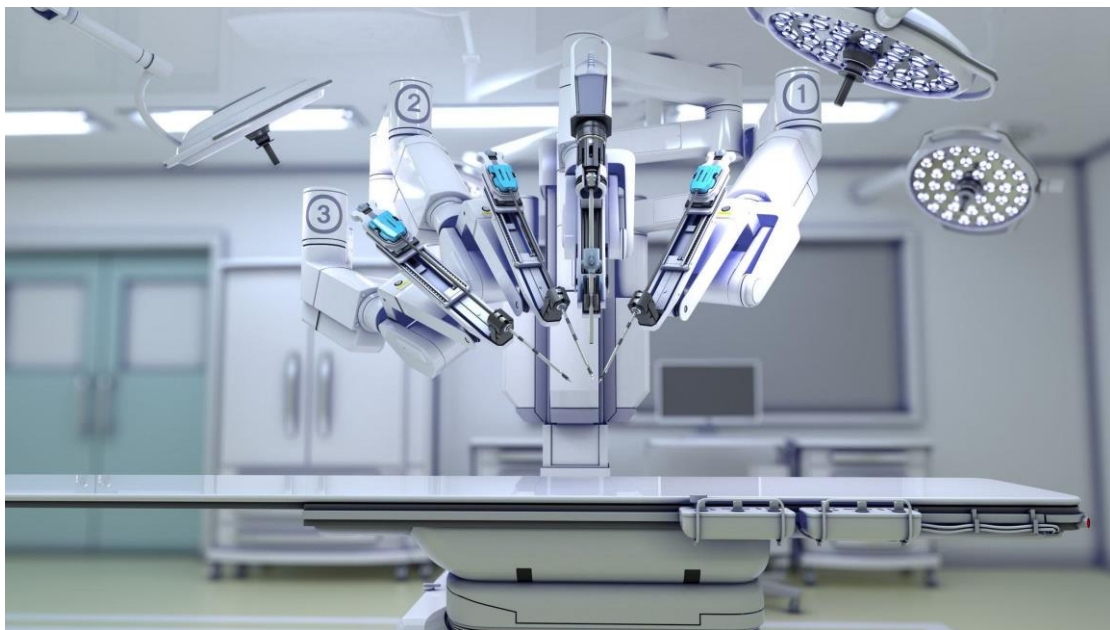
Ακόμη, το 2000 παρουσιάστηκε από την γνωστή εταιρεία Honda, το ρομπότ ASIMO (Εικόνα 7). Το ρομπότ αυτό ήταν ένα εξίσου ανθρωπόμορφο ρομπότ το οποίο όμως άγγιζε ακόμη περισσότερο την ανθρώπινη όψη όσο την κίνηση και την λειτουργία (Πηγή: [6])



Εικόνα 7 - Honda Asimo robot (last version) (Πηγή: [6])

2 Ρομποτική στην ιατρική

Η ρομποτική στην ιατρική αναπτύχθηκε αργότερα έναντι της βιομηχανικής ρομποτικής. Ο λόγος είναι πως στην ιατρική, θα χρειάζεται να υπάρχει εμπιστοσύνη προς το μηχάνημα προκειμένου να αναλάβει μία διαδικασία, να τηρεί κάποιες βασικές προϋποθέσεις υγείας, ασφάλειας και λειτουργίας. Οι πρώτες απόπειρες δημιουργίας ενός ρομποτικού συστήματος για την πραγματοποίηση ενός χειρουργείου, τέθηκε σαν ιδέα την δεκαετία του '70 καθώς αποτέλεσε σημαντική λύση για τον απομακρυσμένο έλεγχο ενός χειρουργείου σε περιπτώσεις όπου ο γιατρός-χειρουργός δεν δύναται να βρεθεί δια ζώσης ή ακόμα και σαν βοήθεια μεταξύ των γιατρών. Η ιδέα αυτή οδήγησε την δημιουργία ενός χειρουργείου απομακρυσμένου ελέγχου εντός του διαστημοπλοίου ώστε σε περίπτωση που ένας αστροναύτης τίθεται σε κίνδυνο η σωματική του υγεία, να υπάρχει η δυνατότητα να πραγματοποιηθεί το συγκεκριμένο χειρουργείο αποτελεσματικά. Δυστυχώς, οι δυσκολίες που εμφανιζόντουσαν ως προς την επικοινωνία Γης – Σταθμού ήταν αρκετές, δυσκολεύοντας έτσι το έργο αυτό. Όμως, η ιδέα αυτή δεν χάθηκε αλλά αναπτύχθηκε για κοντινότερες αποστάσεις. Τέτοια μοντέλα κάνανε την εμφάνισή τους στις αρχές της δεκαετίας του '80.



Εικόνα 8 - Σύγχρονο ρομποτικό χειρουργείο

2.1 - Η αρχή των ρομποτικών συστημάτων στην υγεία

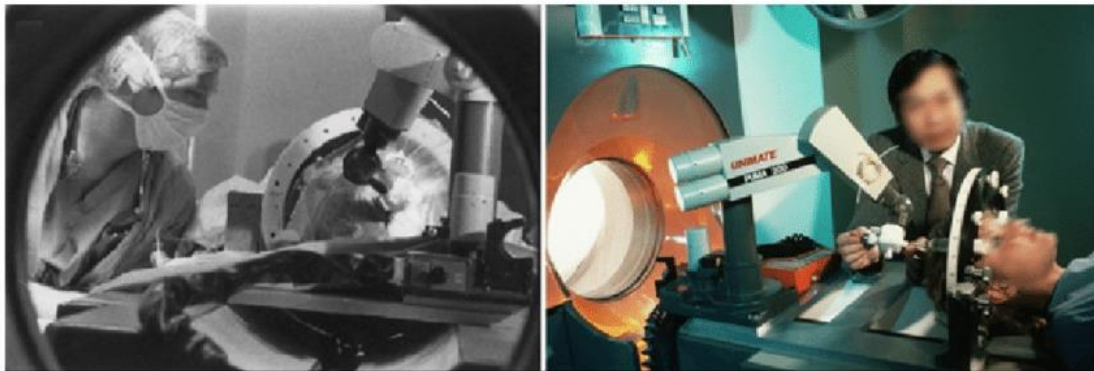
Το πρώτο μηχάνημα ρομπότ που δημιουργήθηκε για τον ιατρικό χώρο, είναι το Arhtrobot (Εικόνα 9). Το σύστημα αυτό αναπτύχθηκε για την υποστήριξη χειρουργικών ορθοπεδικών επεμβάσεων. Το σύστημα αυτό προτάθηκε από τον James McEwen, (Καναδός χειρουργός, εφευρέτης και διευθυντής του τμήματος Μηχανικών Βιοϊατρικής στο γενικό νοσοκομείο του Βανκούβερ, του Καναδά) στον μηχανικό του Geof Auchinleck. Η κατασκευή ξεκίνησε το 1984 καθώς και η πρώτη επέμβαση που πραγματοποιήθηκε με τη χρήση του

συγκεκριμένου ρομπότ, έγινε στις 12 Μαΐου του 1985 στο νοσοκομείο της Βανκούβερ (Πηγή: [7]).



Εικόνα 9 - Ρομποτικό σύστημα Arhtrobot (Πηγή: [7])

Έπειτα, το 1985 η εταιρεία Unimation έκανε μία μετατροπή στο στην έκδοση του βιομηχανικού ρομπότ, το “Puma 200” (Programmable Universal Manipulation Arm) (Εικόνα 10) με σκοπό την υποβοήθηση βιοψίας του εγκεφάλου. Το πλάνο του χειρουργείου είχε την εξής διαδικασία: πρώτα πραγματοποιούνταν λήψη εικόνων από αξονικό τομογράφο (CT) με χρήση κατάλληλου πλαισίου. Με τα αποτελέσματα της αξονικής τομογραφίας, γίνεται ο εντοπισμός των συντεταγμένων του σημείου όπου πρόκειται να γίνει η βιοψία. Έτσι, το ρομπότ χρησιμοποιεί τις συντεταγμένες αυτές ώστε να δείξει την κατεύθυνση στην οποία έπρεπε να γίνει η τομή στο κρανίο για να εισχωρούσε το όργανο για τη βιοψία (Πηγή: [8]).



Εικόνα 10 - Ρομποτικό σύστημα κατά την πρώτη χειρουργική επέμβαση

2.2 - Ρομποτική στον χώρο της ορθοπεδικής

Η ρομποτική άρχισε όλο και περισσότερο να μεγαλώνει τις διαστάσεις της στην ιατρική. Συγκεκριμένα, το έτος 1992 η ρομποτική έκανε ένα αρκετά ολοκληρωμένο βήμα πάνω στην ιατρική και συγκεκριμένα στην ορθοπεδική, όπου ήταν η πρώτη φορά που εισήγαγαν ένα ολοκληρωμένο ρομποτικό ορθοπεδικό χειρουργικό σύστημα, το “ROBODOC”. Την παρουσίαση την έκανε η αμερικάνικη εταιρεία “Integrated Surgical Systems” (Πηγή: [9]). Το έτος 2014, η εταιρεία άλλαξε την ονομασία του μηχανήματος από Robodoc σε TSolution-One (Εικόνα 11).



Εικόνα 11 - TSolution-One

Το συγκεκριμένο μηχανήμα χρησιμοποιείται για την τοποθέτηση εμφυτευμάτων σε ισχίο και στο γόνατο. Στην τωρινή του έκδοση, η χρήση του περιλαμβάνει τρία στάδια. Αυτά είναι:

- Λήψη αξονικής τομογραφίας (προ εγχειρητική) για την δημιουργία τρισδιάστατου μοντέλου της περιοχής ενδιαφέροντος με την βοήθεια ενός workstation (TPLAN 3D Planning Workstation – Εικόνα 12, Πηγή: [10]).
- Στη συνέχεια πραγματοποιείται η μελέτη και επεξεργασία του μοντέλου αυτού από τον χειρουργό για την επιλογή του κατάλληλου εμφυτεύματος για την κατάλληλη τοποθέτηση και ευθυγράμμισή του.

- Στο τέλος, το ρομποτικό σύστημα αναλαμβάνει την αφαίρεση του τμήματος του οστού που μας ενδιαφέρει και την πρόσθεση του νέου οστού-εμφυτεύματος.



Εικόνα 12 - TPLAN 3D Planning Workstation (Workstation για την δημιουργία του 3D μοντέλου της περιοχής ενδιαφέροντος)

2.3 - Επεμβατικές ή Μη Επεμβατικές χειρουργικές επεμβάσεις

Οι επεμβατικές χειρουργικές επεμβάσεις με την χρήση συστημάτων ρομπότ είναι αρκετά αξιόπιστες, λιγότερο οδυνηρές, δεν δημιουργούν μεγάλες τομές μειώνοντας έτσι κατά πολύ τον χρόνο επούλωσης και περιορίζουν τον κίνδυνο μόλυνσης. Αυτές οι επεμβάσεις λέγονται **Ελάχιστα Επεμβατικές Επεμβάσεις (Minimally Invasive Procedures)**. Μέσω μίας ειδικής κάμερας (ενδοσκοπίου) απεικονίζεται το εσωτερικό του ανθρώπινου σώματος και των ειδικών εργαλείων (αρθροσκοπικών για αρθρώσεις ή σπονδυλική στήλη , λαμπαροσκοπικών για κοιλιά ή λεκάνη).

Η αρχή έγινε από μερικά διαφορετικά ρομποτικά συστήματα. Ένα όμως εκ των οποίων που χρησιμοποιήθηκε στον χώρο του χειρουργείου αλλά και σε τέτοιου είδους επεμβάσεις, είναι το μηχάνημα Zeus της εταιρείας Computer Motion (Εικόνα 13). Η ρομποτική αυτή διάταξη αποτελούταν από τρεις βασικούς βραχίονες, έναν για την εικόνα και δύο εκτελεστικούς βραχίονες, δηλαδή εκείνοι που θα πραγματοποιήσουν το χειρουργείο. Ο έλεγχος του μηχανήματος πραγματοποιούταν μέσω κοντινού, στον χώρο, συστήματος από τον γιατρό που είχε αναλάβει το συγκεκριμένο χειρουργείο. Η ακρίβεια του χειρουργείου ήταν αρκετά μεγάλη καθώς για κινήσεις εκατοστών του γιατρού, αντιστοιχούσε σε κινήσεις χιλιοστών του ρομπότ. Ένα αρνητικό που θα μπορούσε να αποδοθεί στο συγκεκριμένο σύστημα, θα μπορούσε να θεωρηθεί η απώλεια αίσθησης του χειρουργικού βάρους από τον γιατρό καθώς από την οθόνη παρατήρησης της πορείας του χειρουργείου, δεν δύναται η δυνατότητα κατανόησης του βάρους. Αντιθέτως, σε μία απλή χειρουργική επέμβαση χωρίς την χρήση ενός ρομποτικού συστήματος, ο γιατρός έχει την δυνατότητα να αντιλαμβάνεται άμεσα το βάθος της επέμβασης. Μία εφαρμογή που έδωσε εξαιρετική σημασία στο Zeus ήταν η εξ αποστάσεως επέμβαση αφαίρεσης χοληδόχου κύστης, με τον χειριστή χειρουργό να βρίσκεται στην Νέα Υόρκη των Ηνωμένων Πολιτειών ενώ η ασθενής βρισκόταν στο Στρασβούργο της Γαλλίας (Πηγή: [10]) .



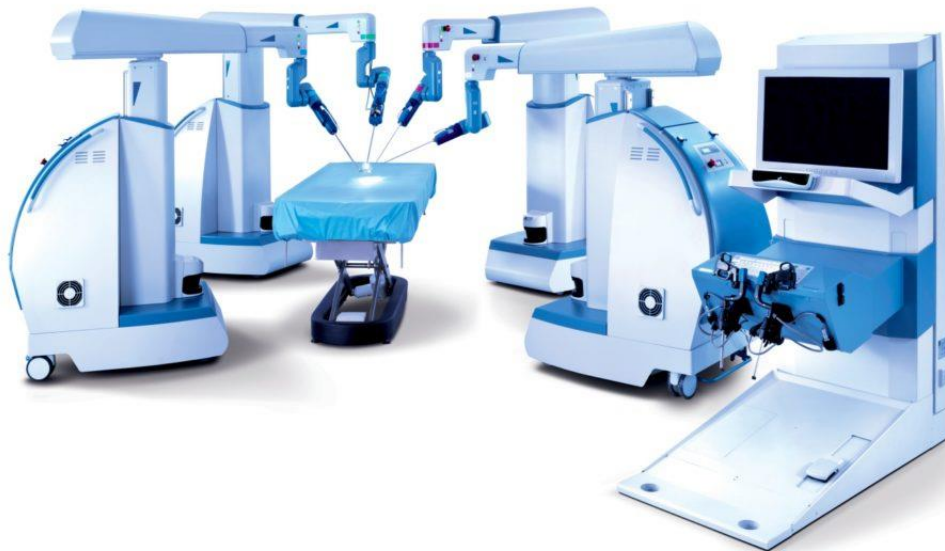
Εικόνα 13 - Σύστημα ZEUS της Computer Motion: Χειρισμός διαδικασίας από τον χειρουργό (αριστερή διάταξη) , ρομποτικοί βραχίονες (δεξιά διάταξη)

Κατά το έτος όμως του 2003, η εταιρεία του Zeus, η Computer Motion εξαγοράστηκε από την μεγαλύτερή της εταιρεία, την Surgical Intuitive. Έτσι, διακόπηκε η παραγωγή του συστήματος Zeus αλλά δημιουργήθηκε μία νέα, καλύτερή του έκδοση, το γνωστό μας πλέον σύστημα, da Vinci (Εικόνα 14) . Σε αντίθεση με το Zeus, το da Vinci αποτελεί ένα τροχήλατο μηχάνημα όπου στην πρώτη του έκδοση ξεκίνησε με τρεις βραχίονες ενώ μετά έγιναν τέσσερις. Μία πολύ βασική διαφορά η οποία συναντάται μεταξύ του συστήματος Zeus και da Vinci, είναι πως η τελευταία είναι εξοπλισμένη με μία τρισδιάστατη στερεοσκοπική κάμερα με δυνατότητα μεγέθυνσης x10, ελεγχόμενη από τον χειρουργό κερδίζοντας έτσι την δυνατότητα μίας σταθερής πλοήγησης στο σώμα του ασθενή. Η χρήση του συστήματος αυτού συναντάται σε γυναικολογικές επεμβάσεις, σε χειρουργικές επεμβάσεις αφαίρεσης του προστάτη και σε επιδιορθώσεις καρδιακής βαλβίδας.



Εικόνα 14 - Ρομποτικό Σύστημα da Vinci

Σε ανταγωνισμό με το μοντέλο da Vinci της Surgical Intuitive, βρίσκουμε το ρομποτικό σύστημα *Telelap ALF-X* (Εικόνα 15) της Ιταλικής εταιρείας *Sofar*. Έπειτα, η εταιρεία *Sofar* εξαγοράστηκε από την εταιρεία *TransEnterix* το έτος 2015, αλλάζοντας έτσι την ονομασία του συστήματος σε *Senhance Surgical Robotic System*. Το σύστημα αυτό, έναντι του da Vinci, έχει δύο βασικά καινοτόμα χαρακτηριστικά. Το πρώτο είναι πως παρέχει «ανάδραση αφής» (haptic feedback) δηλαδή, η υπόδειξη της δύναμης που ασκείται από τον χειρουργό όταν χρησιμοποιεί το χειρουργικό εργαλείο για την επέμβασή του στο ανατομικό όργανο. Το δεύτερο καινοτόμο χαρακτηριστικό, είναι η «παρακολούθηση ματιού» (eye tracking) η οποία προσαρμόζει την εικόνα στην οθόνη ανάλογα με τη θέση οφθαλμού του χειρουργού. Έτσι, όσο ο γιατρός πλησιάζει στην οθόνη, αντίστοιχα το σύστημα μεγεθύνει την εικόνα (Πηγή: [11]). Το εν λόγω μηχάνημα, όπως φαίνεται και στην παρακάτω εικόνα, αποτελείται από τροχήλατους βραχίονες, την τράπεζα εξέτασης και το σημείο χειρισμού στο οποίο βρίσκεται ο αρμόδιος χειρουργός.



Εικόνα 15 - Telelap ALF-X

Αντιθέτως με τα ελαχίστως επεμβατικά ρομποτικά συστήματα, τα μη επεμβατικά δεν έχουν την ανάγκη ανάπτυξης τομών για να προσεγγίσουν την περιοχή ενδιαφέροντος. Συγκεκριμένα, χρησιμοποιούν υπάρχοντες οδούς που βοηθάνε στην πλοήγησή τους για να φτάσουν έως εκεί. Ένα τέτοιο παράδειγμα αποτελεί το ρομποτικό σύστημα *Flex* της εταιρείας *Medrobotics*. Η πρόσβασή του στον ανθρώπινο οργανισμό, πραγματοποιείται από κοιλότητες όπως την μύτη, το στόμα, το ορθό κ.λπ. Η υλοποίηση του γίνεται με την χρήση ενός πολυαρθρωτού σωλήνα ο οποίος φέρει στο τελικό του σημείο (τελικός επενεργητής) μία κάμερα για να επιτυγχάνεται η εικόνα του στον χειριστή/χειρουργό και τα απαραίτητα χειρουργικά εργαλεία (Πηγή: [12]).



Εικόνα 16 - Flex Robot System by Medrobotics (Αριστερά: η ολική διάταξη του ρομποτικού συστήματος, Δεξιά: Ο πολυαρθρωτός σωλήνας εισχώρησης στον ασθενή)

2.4 - Εφαρμογές της ρομποτικής στην νευροχειρουργική

Στην περίπτωση της νευροχειρουργικής, το βασικότερο «εργαλείο» του ιατρού δεν αποτελεί κάποιο αντικείμενο που θα έχει μπροστά του. Αντιθέτως, η βασικότερη ανάγκη για τέτοια χειρουργεία, αποτελεί η ακρίβεια. Στην περίπτωση επεμβάσεων σε ασθενείς με Πάρκινσον, ιδιοπαθή τρόμο (κίνηση άκρων χωρίς την εντολή μας), σε ασθενείς με δυσκινησίες καθώς και σε ψυχιατρικά νοσήματα (ιδεοψυχαναγκαστική διαταραχή, ψυχώσεις, κυκλοθυμία, κατάθλιψη και επιληψία) , η ακρίβεια τους πρέπει να είναι ιδιαίτερα υψηλή καθώς κατά τη διαδικασία ανοίγονται οπές στο κρανίο ώστε να τοποθετηθούν τα κατάλληλα ηλεκτρόδια για να παρέχουν τις κατάλληλες αλλαγές που επιθυμούμε να πραγματοποιηθούν στις συγκεκριμένες περιοχές του εγκεφάλου. Οι επεμβάσεις αυτές διαρκούν αρκετές ώρες και το αποτέλεσμά τους είναι κρίσιμο. Αυτές οι επεμβάσεις είναι οι στερεοτακτικές νευροχειρουργικές όπου τα χειρουργικά ρομπότ αυτού του τομέα αναλαμβάνουν την «Εν βάθος εγκεφαλική διέγερση» (Deep Brain Stimulation).

Τέτοια μηχανήματα έχουν αναπτυχθεί με τον καιρό για την καλύτερη αποτελεσματικότητα των χειρουργείων αυτών. Ένα τέτοιο γνωστό σύστημα αποτελεί το “*NeuroMate*” της εταιρείας “*Renishaw-Mayfield*” (Εικόνα 17). Για την επέμβαση, πραγματοποιούνται τα παρακάτω στάδια. Σε προ εγχειρητικό στάδιο, ο ιατρός παίρνει μία εικόνα του ασθενή από μαγνητικό τομογράφο στην περιοχή του κεφαλιού (MRI). Τα αποτελέσματα τα περνάει σε ένα εξειδικευμένο λογισμικό μέσω του οποίου ο ιατρός κάνει το πλάνο του χειρουργείου. Σε αυτό το στάδιο, ο γιατρός δύναται να επιλέξει το σημείο όπου θα γίνει η οπή και τα σημεία στα οποία θα πραγματοποιηθεί η διέγερση του εγκεφάλου. Έπειτα, κατά την προετοιμασία του χειρουργείου αφού ο ασθενής βρίσκεται στην θέση όπου θα

χειρουργηθεί, πραγματοποιείται η λήψη μίας εικόνας από έναν αξονικό τομογράφο (CT) η οποία συνδυάζεται με την μαγνητική τομογραφία που έχει προηγηθεί. Έτσι, η συνδυασμένη αυτή εικόνα βοηθάει το ρομποτικό σύστημα να ευθυγραμμιστεί στον χώρο και να πραγματοποιήσει την επέμβαση με τις επιλογές του χειρουργού (σημείο οπών, σημεία διέγερσης του εγκεφάλου κ.α.). Αφού το στάδιο αυτό έχει ολοκληρωθεί, η διαδικασία ξεκινάει με μία δερματική τομή στο κρανίο του ασθενούς και έπειτα το άνοιγμα οπών στο κρανίο για την επέμβαση του. Τοποθετείται στον ρομποτικό βραχίονα ένα ειδικό σύρμα το οποίο φαίνεται στην αξονική τομογραφία. Αυτό το σύρμα αποτελεί το μέσο με το οποίο ανοίγεται ο «δρόμος» για την τοποθέτηση των ηλεκτροδίων. Κατά την όδευση του προς το σημείο, πραγματοποιείται έλεγχος για το εάν η υπάρχει απόκλιση της πραγματικής τροχιάς με την θεωρητική που δημιούργησε ο χειρουργός στο στάδιο της μαγνητικής. Σε περίπτωση που υπάρχει σημαντική απόκλιση, αυτομάτως διορθώνεται. Όταν το σύρμα φτάσει στο σημείο-στόχο του εγκεφάλου που έχει οριστεί από τον χειρουργό, επιβεβαιώνεται το σημείο της τροχιάς, αφαιρείται το νήμα και τοποθετούνται τα ειδικά ηλεκτρόδια για να πραγματοποιηθεί το απαραίτητο ερέθισμα. Αυτή η διαδικασία είναι η διαδικασία για κάθε σημείο που πρέπει να γίνει η συγκεκριμένη επέμβαση.



Εικόνα 17 - Συμβατική νευροχειρουργική (αριστερά) Ρομποτική Νευροχειρουργική (Δεξιά) - (Πηγή: Renishaw)

2.5 - Εφαρμογές της ρομποτικής στην ακτινοθεραπεία

Ένα μεγάλο πρόβλημα της ιατρικής, ιδιαίτερα τα τελευταία χρόνια, δείχνει να είναι η αντιμετώπιση του καρκίνου σε καρκινοπαθείς ασθενείς. Η αντιμετώπιση του καρκίνου έρχεται από τις αρχές του 1970 μετά από την μελέτη του. Σήμερα, ο κλάδος της ακτινοθεραπείας έχει ακμάσει φέρνοντας έτσι πιο αποτελεσματικά συστήματα για την θεραπεία του. Συγκεκριμένα, το πιο γνωστό ρομποτικό σύστημα ακτινοθεραπείας είναι το *CyberKnife* (Εικόνα 18) που κατασκευάζεται από την εταιρεία *Accuray Incorporated*. Το συγκεκριμένο σύστημα αναπτύχθηκε από Φυσικούς και Μηχανικούς του πανεπιστημίου του Στάνφορντ της Αμερικής το 1999 όπου έγινε αποδεκτής από το U.S. Food and Drug Administration (FDA). Το σύστημα δόθηκε σε λειτουργία το 2001 και έχει θεραπεύσει περισσότερους από 20,000 ασθενείς παγκοσμίως (Πηγή: [13]).



Εικόνα 18 - CyberKnife (Πηγή Wikipedia)

Το σύστημα του CyberKnife είναι μία αρθρωτή ρομποτική διάταξη η οποία φέρει στο τελικό το μέρος έναν γραμμικό επιταχυντή (linear accelerator – linac) ο οποίος κοιτάζει στον τελικό στόχο που έχει προγραμματιστεί για να εξουδετερώσει. Ο γραμμικός του επιταχυντής δημιουργεί ακτίνες X μεγάλης ενέργειας ή φωτόνια για να πραγματοποιηθεί η θεραπεία. Για την καλύτερη του ακρίβεια αλλά και χρησιμότητα, το σύστημα παρέχει και την δυνατότητα της ζωντανής μετάδοσης εικόνας καθώς και την μέτρηση της δόσης που δέχεται ο ασθενής από αρκετές γωνίες για να έχουμε τα βέλτιστα αποτελέσματα. Το σύστημα αυτό είναι μη-επεμβατικό και η αρμοδιότητα του είναι στην εξόντωση καρκινικών και μη, όγκων που βρίσκονται στο σώμα του ασθενή. Συγκεκριμένα, οι περιοχές που καλύπτει μπορεί να είναι ο προστάτης, τα πνευμόνια, ο εγκέφαλος, η σπονδυλική στήλη, το κεφάλι και ο αυχένας – λαιμός, το συκώτι, το πάγκρεας, τα νεφρά αλλά μπορεί να χρησιμοποιηθεί και σε ασθενείς με περίπλοκη διάταξη του καρκίνου στο σώμα τους. Οι θεραπείες που χρειάζονται συνήθως από το σύστημα αυτό χωρίζονται σε μία έως πέντε συνεδρίες (Πηγή: [14]).

2.6 - Αποκατάσταση

Πολλές φορές, ασθενείς μετά από τραυματισμούς (αυτοκινητιστικά ατυχήματα ή άλλες περιπτώσεις), νευρολογικές παθήσεις, εγκεφαλικά επεισόδια ή οτιδήποτε, πάσχουν από δυσκολίες στην κίνησή τους. Η φυσική επιδιόρθωση αυτών των ατόμων είναι δύσκολη καθώς θα πρέπει να υπάρξει ειδικός χειρισμός για να επανέλθει ξανά ο οργανισμός του στην καλύτερη κατάσταση του, ιδανικά στην πρωτεύουσα κατάστασή του οργανισμού πριν την κατάσταση αλλαγής. Το αποτέλεσμα ενός εγκεφαλικού επεισοδίου μπορεί να προκαλέσει σοβαρή βλάβη σε ένα μέρος του σώματος (αριστερό ή δεξιό ημισφαίριο του εγκεφάλου) με επίπτωση την δυσκολία ή και αδυναμία του ατόμου να κινηθεί. Η επαναφορά του ασθενή στην φυσιολογική του κατάσταση γίνεται με την βοήθεια των φυσιοθεραπευτών. Η επιστήμη της φυσιοθεραπείας έχει ακριβώς αυτόν τον σκοπό, την αποκατάσταση των κινήσεων ενός ασθενή έπειτα από

κάποιο επεισόδιο το οποίο του προκάλεσε αποδεδειγμένη δυσκολία στην κίνησή του ή σε κάποια άλλη διαδικασία. Δυστυχώς, κατά τη διαδικασία της φυσιοθεραπείας, ο όγκος του ασθενή διαχειρίζεται πολλές φορές από τον ίδιο τον φυσιοθεραπευτή, ένα γεγονός το οποίο επιβαρύνει την επίδοση του με την συνέχεια της ημέρας. Για αυτόν τον σκοπό, έχουν αναπτυχθεί ρομποτικά συστήματα *Αποκατάστασης (Rehabilitation)*. Είναι δηλαδή ένας τρόπος, ή ένα μέσο, το οποίο μπορεί να προσφέρει σωστές τοποθετήσεις του ασθενή στο σημείο που έχει το πρόβλημα, να τον κρατήσει όρθιο ή και να υποβοηθήσει την προσπάθεια των κινήσεων του προκειμένου να βοηθηθεί και να επανέλθει.

Τα ρομποτικά αυτά συστήματα υποβοήθησης του ασθενή για την αποκατάστασή του, ανήκουν στην κατηγορία των **εξωσκελετών**. Η ονομασία αυτή δίδεται χάριν στην διάταξή του που ακολουθεί ακριβώς την οστική διάταξη των άνω ή των κάτω άκρων. Αυτά τα συστήματα, είναι προγραμματισμένα να αποδίδουν συγκεκριμένες δυνάμεις, αντιστάσεις ή κινήσεις ανάλογα με την αποκατάσταση που έχει επιλεγεί. Παραδείγματα αποκατάστασης των άνω άκρων είναι τα εξής ρομποτικά συστήματα:

- **MIT-MANUS** της Interactive Motion Technologies, Cambridge, MA (Πηγή:[15])
- **ARMIN IV** (Πηγή: [16])
- **MGA** (Πηγή: [17])
- **CADEN-7** (Πηγή: [18])
- **SRE**
- **ABLE**
- **RUPET IV**

Ακόμη, πέρα από την αποκατάσταση των άνω άκρων, τα ρομποτικά συστήματα χρησιμοποιούνται για την αποκατάσταση των κάτω άκρων επίσης. Ένα τέτοιο παράδειγμα αποτελεί το σύστημα “*Lokomat*” της Ελβετικής εταιρείας “*Hocoma*” το οποίο ενισχύει κάθε προσπάθεια του ασθενή με τις δυσλειτουργίες των κάτω άκρων με σκοπό την «αναγέννηση» της δύναμης τους (Πηγή: [19]).



Εικόνα 19 - Hocoma Lokomat® σύστημα αποκατάστασης κάτω άκρων

Οι περιπτώσεις που αναθέτονται στα ρομποτικά αυτά συστήματα για την επιδιόρθωση των κάτω άκρων, βασίζονται περισσότερο σε περιπτώσεις αποκατάστασης του αστραγάλου του ασθενή ή σε περίπτωση κάποιας ανεπιθύμητης αλλαγής στην βάδιση έπειτα από κάποιο επεισόδιο ή νευρολογική διαταραχή. Στην περίπτωση του αστραγάλου, έχουμε δύο βασικές κατηγορίες ταξινόμησης της αποκατάστασης, τα **φορετά** (Wearables) και οι **ρομποτικές πλατφόρμες** (Robotic Platforms). Στην πρώτη περίπτωση, η μορφή τους είναι ένα εξωσκελετικό ρομποτικό σύστημα το οποίο πραγματοποιεί κατάλληλες κινήσεις για την επιδιόρθωση της άρθρωσης του αστραγάλου. Στην δεύτερη περίπτωση συναντάμε ειδικές ρομποτικές πλατφόρμες στις οποίες τοποθετείται το πόδι του ασθενούς. Έτσι, μέσω συγκεκριμένων, κυκλικών κινήσεων εντός της επιτρεπτής γωνίας της άρθρωσης, πραγματοποιείται η αποκατάσταση βλαβών (Πηγή:[20]).

3 Έλεγχος οπτικής οδήγησης ρομπότ

Τα ρομποτικά συστήματα μπορούν να εκτελούν λίγες ή αρκετές ενέργειες με διάφορους τρόπους. Συγκεκριμένα, μπορούν να προγραμματιστούν να εκτελούν ακριβείς κινήσεις στον χώρο μέσω του προγραμματισμού τους σύμφωνα με την ανάγκη της υπηρεσίας που χρειάζεται να εκτελεσθεί αλλά και την διάθεση του χώρου. Για παράδειγμα, ένα ρομποτικό σύστημα το οποίο τοποθετεί την πόρτα του αυτοκινήτου στο πλαίσιο του οχήματος, είναι ένα σύστημα προγραμματισμένο να πραγματοποιεί συγκεκριμένες κινήσεις κατά την διάρκεια της παραγωγής των οχημάτων. Ένα ρομποτικό σύστημα το οποίο δέχεται διάφορες εξετάσεις αίματος και τις ταξινομεί, είναι προγραμματισμένο να εκτελεί αυτή τη κίνηση με την βοήθεια κατάλληλων αισθητήρων.

Αντιθέτως, στον χώρο της ρομποτικής, υπάρχουν συστήματα με ένα πολύ βασικό χαρακτηριστικό που τοποθετείται για την πραγματοποίηση άλλων σκοπών. Αυτό το χαρακτηριστικό, είναι η **ρομποτική όραση**. Η ρομποτική όραση είναι ένας τρόπος με τον οποίον το ρομποτικό σύστημα μπορεί να συλλέξει πληροφορίες του περιβάλλοντος του, του χώρου ενδιαφέροντος αλλά και αντικειμένων ή καταστάσεων που θέλει να αποθηκεύσει για να εκτελέσει μία ενέργεια. Συγκεκριμένα, αναφερόμαστε στην αναγνώριση προτύπων με βάση την ρομποτική όραση. Τέτοια συστήματα, χρησιμοποιούν την «όρασή» τους για να αναγνωρίσουν αντικείμενα στον χώρο τα οποία, μέσω κατάλληλου προγραμματισμού, ανήκουν στο ενδιαφέρον της συνολικής εκτέλεσης της εργασίας. Για παράδειγμα, ένα απλοϊκό ρομποτικό σύστημα το οποίο έχει προγραμματιστεί να διαχωρίζει αντικείμενα με συγκεκριμένα γεωμετρικά χαρακτηριστικά (τετράγωνα κουτιά, σφαιροειδή αντικείμενα), χρειάζεται την ικανότητα της ρομποτικής όρασης προκειμένου να αναζητήσει τα αντικείμενα αυτά στον χώρο, να τα εντοπίσει και να επιστημάνει τις συντεταγμένες τους με βάση την αρχική του στην οποία έχει βαθμονομηθεί, να εκτελέσει την βέλτιστη και ασφαλέστερη διαδρομή για την απόκτησή του και τέλος, την επιστροφή ή τοποθέτηση του στο κατάλληλο πεδίο που έχει οριστεί από τον προγραμματισμό του. Ακόμη, η ρομποτική όραση δεν απευθύνεται μόνο στην αναγνώριση προτύπων σύμφωνα με τα γεωμετρικά χαρακτηριστικά του αντικειμένου καθώς είναι δυνατή η αναγνώριση των αντικειμένων με βάση περισσότερων ιδιοτήτων. Τέτοιες ιδιότητες αποτελούν το χρώμα, η θερμοκρασία, το μέγεθος αλλά και άλλα χαρακτηριστικά πιο σύνθετα. Ένα βασικό και καθημερινό παράδειγμα ρομποτικής όρασης και κρίσης, είναι η

αναγνώριση προσώπου των κινητών. Η διαδικασία αυτή προϋποθέτει την όραση του συστήματος του τηλεφώνου (κάμερα), την δημιουργία ενός προτύπου σημείων σύμφωνα με τα χαρακτηριστικά του ιδιοκτήτη του τηλεφώνου, την αποθήκευση του συγκεκριμένου προτύπου σε έναν ασφαλή χώρο στο κινητό και την τυχόν βαθμονόμηση του προτύπου σε περίπτωση αλλαγών. Έτσι, με το άνοιγμα του τηλεφώνου, η κάμερα αυτομάτως ελέγχει το πρόσωπο του χειριστή εκείνης της στιγμής βάζοντας όρια ενδιαφέροντος γύρω από το πρόσωπο του για να συγκρίνει τα πρότυπα που «βλέπει». Μετράει τις αποκλίσεις μεταξύ του αποθηκευμένου προτύπου του χρήστη και έτσι «κρίνει» εάν ο χειριστής είναι ο ιδιοκτήτης. Στην περίπτωση που είναι, το κινητό ανοίγει ενώ παραμένει κλειστό στη περίπτωση που ο χειριστής δεν είναι ο ιδιοκτήτης.

Η ρομποτική όραση εντάχθηκε στον κόσμο μας έπειτα από τις υψηλές απαιτήσεις του ανθρώπου στην αύξηση της ταχύτητας της παραγωγής αγαθών και υπηρεσιών με την ταυτόχρονη μείωση των αποκλίσεων που μπορούν να πραγματοποιηθούν κατά την διαδικασία αυτή. Αυτή η ανάγκη έφερε την δημιουργία της ρομποτικής όρασης στις δεκαετίες του 1980 με 1990. Η εικόνα αναφέρεται σε πληροφορίες δύο διαστάσεων (2D - 2 Dimensional) όπου μία κάμερα τοποθετημένη επάνω στο ρομποτικό σύστημα (κινούμενη αλλά με πάντα την ίδια γωνία όψης) ή τοποθετημένη σε σταθερό σημείο (στατική θέση) και συλλέγει πληροφορίες του χώρου και των αντικειμένων σχετικά με τα προϊόντα της παραγωγής. Έτσι, δημιουργείται ένα αλγόριθμος ο οποίος βοηθάει στην αναγνώριση και την αξιολόγηση του προϊόντος ανάλογα το αρχικό με το οποίο έχει εκπαιδευτεί. Για παράδειγμα, στην περίπτωση μίας εταιρείας παραγωγής κλειδιών, κατά την συλλογή των κλειδιών στις θήκες τους με προορισμό την πώληση, τοποθετούνται όσα κλειδιά είναι όμοια με αυτό που έχει ζητήσει ο κατασκευαστής ενώ τα υπόλοιπα που έχουν είτε μικρές είτε σοβαρές αποκλίσεις με το αρχικό, απορρίπτονται (Πηγή: [21]). Τέτοια ρομποτικά συστήματα έχουν περισσότερο αναπτυχθεί στους τομείς της παραγωγής καθώς έχουν συνεισφέρει σημαντικά στην αύξηση του ρυθμού παραγωγής, της μείωσης του λάθους από ανθρώπινο παράγοντα ιδιαίτερα λόγω απειρίας αλλά και στην ακρίβεια της ποιότητας των αγαθών. Τέτοια ρομποτικά συστήματα δημιουργούνται από εταιρείες όπως η *FANUC*, *YASKAWA*, *ABB*, *KUKA* και λοιπά. Πέραν από ρομποτική όραση σε δύο διαστάσεις, έχει αρχίσει να αναπτύσσεται εξίσου και η ρομποτική όραση σε τρεις διαστάσεις (3D – 3 Dimensional) σε αυτούς τους χώρους, όπως για παράδειγμα το αυτοματοποιημένο σύστημα ρομπότ το οποίο καλείται να επιθεωρήσει προϊόντα παραγωγής στους τρεις του άξονες. Το σύστημα αυτό ανήκει στην εταιρεία *ABB*, πρόκειται για ένα σύστημα το οποίο είναι του έτους 2020 με πολλές δυνατότητες (Εικόνα 20).



Εικόνα 20 - ABB 3D inspection robot (Πηγή: [22])

Η κάμερες αποτελούνται από συγκεκριμένα στοιχεία και το πιο βασικό είναι μία συστοιχία αισθητήρων με την οποία «αποτυπώνεται» η εικόνα. Οι αισθητήρες αυτοί είναι φτιαγμένοι με τέτοιο τρόπο ώστε να δημιουργούν τάση στις εξόδους του όταν το φως προσπέσει επάνω τους. Έτσι, αυτή η αναλογική τάση μετατρέπεται σε μία ψηφιακή, αντίστοιχη της έντασης του φωτός ή άλλου χαρακτηριστικού. Αυτοί οι αισθητήρες τοποθετούνται ο ένας δίπλα στον άλλον ώστε να φτιάξουν την εικόνα. Κάθε αισθητήρας αντιστοιχεί σε ένα pixel (picture element) της εικόνας. Οι πιο βασικές διατάξεις σαν αυτές είναι:

- CCD (Charge coupled device – διάταξη ζεύγους φορτίου)
- APS (active pixel sensor – αισθητήρας ενεργού pixel)

Οι CCD διατάξεις είναι αρκετά αξιόπιστες και ακριβείς διατάξεις με πολύ καλή τελική εικόνα. Το αρνητικό τους είναι το κόστος τους καθώς αποτελείται από πιο σύνθετα υλικά. Παρόλα αυτά, οι CCD διατάξεις χρησιμοποιούνται σε επαγγελματικούς εξοπλισμούς αλλά και σε ιατρικές ή επιστημονικές εφαρμογές. Αντιθέτως με τις CCD, οι διατάξεις APS αποτελούν μία πιο οικονομική λύση και χρησιμοποιούνται κυρίως από απλά προϊόντα όπως τις κάμερες των υπολογιστών ή των κινητών τηλεφώνων.

Οι εικόνες που παίρνουν αυτές οι διατάξεις μπορούν να αναλυθούν σε δυαδικές (binary images) όπου υπάρχουν δύο βασικά χρώματα (λευκό και μαύρο), μονόχρωμες εικόνες ή γκριζας κλίμακας (grayscale images) όπου υπάρχουν 256 διαφορετικές αποχρώσεις του γκρι σαν πληροφορία της εικόνας, εικόνες πλήρους χρώματος (full color images) ή RGB εικόνες όπου το αποτέλεσμα τους προκύπτει από την ανάμειξη των τριών βασικών χρωμάτων

(Κόκκινο-Πράσινο-Μπλε) και τέλος, έγχρωμες εικόνες με παλέτα οι οποίες περιλαμβάνουν συγκεκριμένα χρώματα.

Στη συνέχεια, εμείς εκμεταλλευόμαστε την εικόνα στον υπολογιστή. Μία εικόνα είναι ένα σύνολο αριθμητικών τιμών όπου κάθε τιμή αντιπροσωπεύει μία πληροφορία ανά εικονοστοιχείο (pixel). Έτσι, έχουμε στην κατοχή μας έναν πίνακα με γραμμές και στήλες ίσες με την ανάλυση της εικόνας. Για παράδειγμα, μία εικόνα 500X500 αντιστοιχεί σε 500 γραμμές και 500 στήλες πληροφοριών ανά εικονοστοιχείο όπου σε κάθε σημείο γραμμής και στήλης αντιστοιχεί σε έναν τόνο του γκρι.

Στην συγκεκριμένη περίπτωση, θα προσπαθούμε να εφαρμόσουμε τον εντοπισμό του αντικειμένου ενδιαφέροντος μέσω της χρήσης μίας κάμερας. Για να το επιτύχουμε αυτό, θα πρέπει να εφαρμόσουμε ειδικές μεθόδους ανάλυσης εικόνας ώστε να δώσουμε τις απαραίτητες πληροφορίες στον ρομποτικό βραχίονα να μεταβεί στο σημείο αυτό και να συλλέξει το αντικείμενο που θέλουμε. Για αυτή την διαδικασία υπάρχουν διάφορες τεχνικές. Αυτές είναι η κατωφλίωση, αφαίρεση χρωματικών συνιστωσών, αφαίρεση υποβάθρου, ανάπτυξη περιοχής και ομαδοποίηση k-Μέσων. Η τεχνική της «κατωφλίωσης» χρησιμοποιείται για μονόχρωμες εικόνες όπου τα αντικείμενα που υπάρχουν στον χώρο θα πρέπει να έχουν χρωματικές διαφορές για την ευκολότερη αναγνώρισή τους. Έτσι, επιλέγοντας μία τιμή κατωφλιού, δημιουργούμε ένα μαύρο υπόβαθρο στην περιοχή μη ενδιαφέροντος και λευκό στο αντικείμενο ενδιαφέροντος. Στην «τεχνική αφαίρεσης χρωματικών συνιστωσών», εντοπίζουμε ένα αντικείμενο συγκεκριμένου χρώματος. Για παράδειγμα, σε μία εικόνα όπου υπάρχουν διάφορα χρωματιστά αντικείμενα, επιλέγουμε να εντοπίσουμε μόνο το πράσινο αντικείμενο. Έτσι, στο σύνολο της εικόνας αφαιρούμε όλους τους υπόλοιπους περιττούς χρωματισμούς (κόκκινο και μπλε) και έχουμε το αντικείμενο που θέλουμε. Σε περίπτωση όμως που δεν γνωρίζουμε το χρώμα του αντικειμένου και δεν είναι από τις αρχικές μας πληροφορίες, τότε χρησιμοποιούμε την τεχνική «αφαίρεση υποβάθρου». Έτσι, έχοντας μία εικόνα πρότυπο του υποβάθρου, μπορούμε να ανιχνεύσουμε οτιδήποτε εισέρχεται στον χώρο αυτόν απλά αφαιρώντας τις τιμές του υποβάθρου από την νέα εικόνα. Στην περίπτωση όπου θέλουμε να έχει ο χρήστης την ικανότητα της επιλογής ενός αντικειμένου, εφαρμόζουμε την διαδικασία της «ανάπτυξης περιοχής» (region growing), μία διαδικασία με την οποία ο χρήστης επιλέγει ένα επιθυμητό εικονοστοιχείο (pixel) σαν αρχικό και το ορίζει ως πρότυπο σύμφωνα με το οποίο πλέον ο υπολογιστής θα υπολογίσει τα γειτονικά του στοιχεία στο κατά πόσο είναι 'κοντινά' στην τιμή του προτύπου. Αυτή η διαδικασία επαναλαμβάνεται συνεχώς έως ότου οριοθετηθεί το αντικείμενο που επέλεξε ο χρήστης με άσπρο χρώμα και με μαύρο η περιοχή μη ενδιαφέροντος. Τέλος, η τεχνική της «ομαδοποίησης k-μέσων» (k-Means Clustering) αναφέρεται στην ομαδοποίηση ενός ή και παραπάνω αντικειμένων σε έναν χώρο. Για παράδειγμα, έχοντας ένα μονόχρωμο υπόβαθρο (background) στην εικόνα μας και δύο αντικείμενα διαφορετικού χρώματος, επιλέγοντας για $k=2$, η διαδικασία μας οριοθετεί τα δύο διαφορετικά αντικείμενα δείχνοντάς μας λευκά και μαύρη την περιοχή μη ενδιαφέροντος.

4 Γιατί ρομποτική στον ιατρικό χώρο;

Πλέον, είναι εύκολο να κατανοήσουμε πως η χρήση εξειδικευμένων ρομποτικών συστημάτων στον ιατρικό χώρο, μας επιφέρει κάποια βασικά οφέλη τα οποία αναζητούμε για την επίτευξη των καλύτερων αποτελεσμάτων. Η στρέψη του ανθρώπου στα συστήματα αυτά έγινε άλλωστε εξαιτίας αυτού του κριτηρίου, της ανάγκης για καλύτερα αποτελέσματα μέσω ασφαλέστερων διαδικασιών αφαιρώντας όλο και πιο πολύ τις πιθανότητες κινδύνου για τον ασθενή αλλά και για το προσωπικό επίσης. Έχει λοιπόν παρατηρηθεί πως η χρήση ρομποτικών συστημάτων στον ιατρικό χώρο, ειδικά στον τομέα της χειρουργικής ιατρικής, μας έχει επιφέρει τα εξής πλεονεκτήματα:

- **Ακρίβεια κίνησης και τοποθέτησης του εργαλείου**
- **Καλύτερη λεπτομέρεια στην κίνηση**
- **Σταθεροποίηση κίνησης**
- **Ευελιξία σύνθετων κινήσεων**
- **Ταυτόχρονη χρήση δεδομένων από διάφορες πηγές του ίδιου ασθενή**

Τα πλεονεκτήματα αυτά αποτελούν σημαντική αξία για την είσοδο και την παραμονή των ρομποτικών συστημάτων στον τομέα της υγείας. Με την καλύτερη λεπτομέρεια στην κίνηση του, ένα ρομπότ είναι ικανό να μεταφράσει τις εκατοστιαίες κινήσεις του ιατρού σε κινήσεις χιλιοστών στον ασθενή. Η σταθεροποίηση της κίνησης είναι ιδιαίτερα σημαντική ώστε να αποφεύγονται τα λάθη κατά την διαδικασία των χειρουργείων καθώς η ευελιξία της κίνησης δημιουργεί μία φιλικότητα ως προς τον χρήστη (ιατρό) κατά τον χειρισμό του καθώς η σύνθεση του κίνηση μπορεί να επιτευχθεί καλύτερα από τα ανθρώπινα χέρια. Τέλος, αξίζει να σημειωθεί πως ένα υπέρ της χρήσης των ρομποτικών συστημάτων, είναι η δυνατότητα συνδυασμού περισσότερων πληροφοριών για τον ασθενή. Για παράδειγμα, ένας ασθενής ο οποίος χειρουργείται στην κάτω κοιλία με την χρήση ενός ρομπότ, θα διατρέχει λιγότερο κίνδυνο στην περίπτωση που έχει πραγματοποιηθεί μία διαγνωστική εξέταση (π.χ. Αξονική τομογραφία κάτω κοιλίας). Ο λόγος είναι επειδή το σύστημα θα μπορεί να βλέπει την θέση του βασικού προβλήματος (π.χ. κακοήθης όγκος) ανάμεσα σε όργανα τα οποία δεν θα πρέπει να πειραχτούν από την επέμβαση του ιατρού. Έτσι, σε χειρουργεία με ανάγκη για μεγάλη λεπτομέρεια στην τομή και την αφαίρεση όγκων, η δυνατότητα αυτή βοηθάει άμεσα στην καταπολέμηση του κινδύνου να τραυματιστούν γειτονικά όργανα, αγγεία και οτιδήποτε άλλο υγιές.

4.1 - Βασικές προδιαγραφές των ρομπότ για ιατρική χρήση

Στην συγκεκριμένη περίπτωση που ασχολούμαστε με τα ρομπότ για την ιατρική χρήση, οι βασικές προδιαγραφές που θα πρέπει να έχουν απαραίτητα είναι οι εξής:

Δ) Ασφάλεια στον χώρο εργασίας (ασθενής και ιατρικό προσωπικό): Με την ασφάλεια στον χώρο εργασίας εννοούμε την αξιοπιστία του συστήματος ως προς την ασφάλεια της υγείας του ασθενή και του ιατρικού προσωπικού. Καθώς με τη χρήση του ρομποτικού συστήματος πραγματοποιείται η επαφή του συστήματος με τον ασθενή, θα πρέπει να υπάρχει η βασική προδιαγραφή που να ορίζει πως το σύστημα δεν είναι ικανό να δημιουργήσει πρόβλημα υγείας

στις παραπάνω ομάδες με κάποιον τρόπο. Αξίζει να σημειωθεί πως τα ατυχήματα στον ιατρικό τομέα, δημιουργούν μεγαλειώδη προβλήματα τόσο στον χειριστή-ιατρό που ήταν υπεύθυνος εκείνη την ώρα καθώς και στην κατασκευάστρια εταιρεία η οποία ενέκρινε την παραγωγή του συγκεκριμένου συστήματος.

II) **Δυνατότητα αποστείρωσης** : Η δυνατότητα αποστείρωσης αναφέρεται στην ικανότητα των υλικών που διαθέτει το ρομποτικό σύστημα στην πλήρη αποστείρωση του από τις χρήσεις προηγούμενων ασθενών. Ένα μέτρο ασφάλειας κατά την διαδικασία ενός ιατρικού ελέγχου ή κάποιας πιο εξειδικευμένης εφαρμογής είναι η ασφάλεια της περίπτωσης τυχόν μόλυνσης του ασθενή από τον ιατρικό εξοπλισμό. Για παράδειγμα, στη περίπτωση ενός χειρουργείου, ένα περιστατικό το οποίο μολύνθηκε σοβαρά κατά την επέμβαση ενός ρομποτικού συστήματος λόγω κακού καθαρισμού των μελών του, θα αποτελούσε πρόβλημα ζωτικής σημασίας για την ύπαρξη των συστημάτων αυτών στον ιατρικό χώρο.

III) **Χώρος λειτουργίας** : Όπως κάθε σύστημα στον ιατρικό χώρο, θα πρέπει να τηρεί κάποιες βασικές προδιαγραφές για την εγκατάσταση του και την χρήση του στον χώρο του. Για παράδειγμα, βαριά μηχανήματα του ιατρικού χώρου (π.χ. Μαγνητικός τομογράφος) τοποθετούνται σε ειδικό χώρο με μη μαγνητιζόμενα υλικά, σε χαμηλούς ορόφους και με προσοχή στον σχεδιασμό των διπλανών χώρων. Έτσι, τα ρομποτικά συστήματα έχουν βασικές προϋποθέσεις στον χώρο λειτουργίας τους για να είναι ικανά και πλήρως λειτουργικά. Αυτές οι προδιαγραφές είναι οι εξής:

- **Μη ύπαρξη φυσικών εμποδίων στον χώρο λειτουργίας του** (τραπέζια, καρέκλες, υάλινα αντικείμενα, σταθερά αντικείμενα κ.α.)
- **Η μεταφορά του συστήματος να είναι εύκολη σε περίπτωση μεταφερόμενου συστήματος.** Δεν συμβαίνει το ίδιο σε σταθερές μονάδες ρομποτικών επεμβάσεων.
- **Ηλεκτρομαγνητική συμβατότητα με τον υπόλοιπο ιατρικό εξοπλισμό.**
- **Να μπορεί να λειτουργήσει χωρίς την παρουσία εξειδικευμένου τεχνικού προσωπικού.**

Με τις τρεις αυτές βασικές προδιαγραφές επιτυγχάνεται η ασφαλής χρήση των συστημάτων αυτών στον ιατρικό χώρο χωρίς να δημιουργούν πιθανούς κινδύνους κατά την χρήση τους.

Ειδικό μέρος – Σχεδιασμός & Εκτέλεση

1 Ρομποτικό σύστημα Mover 6 της Commonplace Robotics GmbH

1.1 Εγκατάσταση και καλωδίωση

Στα πρώτα βήματα της εγκατάστασης του ρομποτικού βραχίονα στον χώρο, θα έρθουμε σε επαφή με την απαραίτητη καλωδίωση που χρειάζεται για την παροχή τάσης, προστασίας και επικοινωνίας με το ρομποτικό σύστημα και τον χρήστη. Σύμφωνα με το εγχειρίδιο του προϊόντος, η καλωδίωση πρέπει να συνδεθεί με τον παρακάτω τρόπο που φαίνεται στην εικόνα (Εικόνα 21).

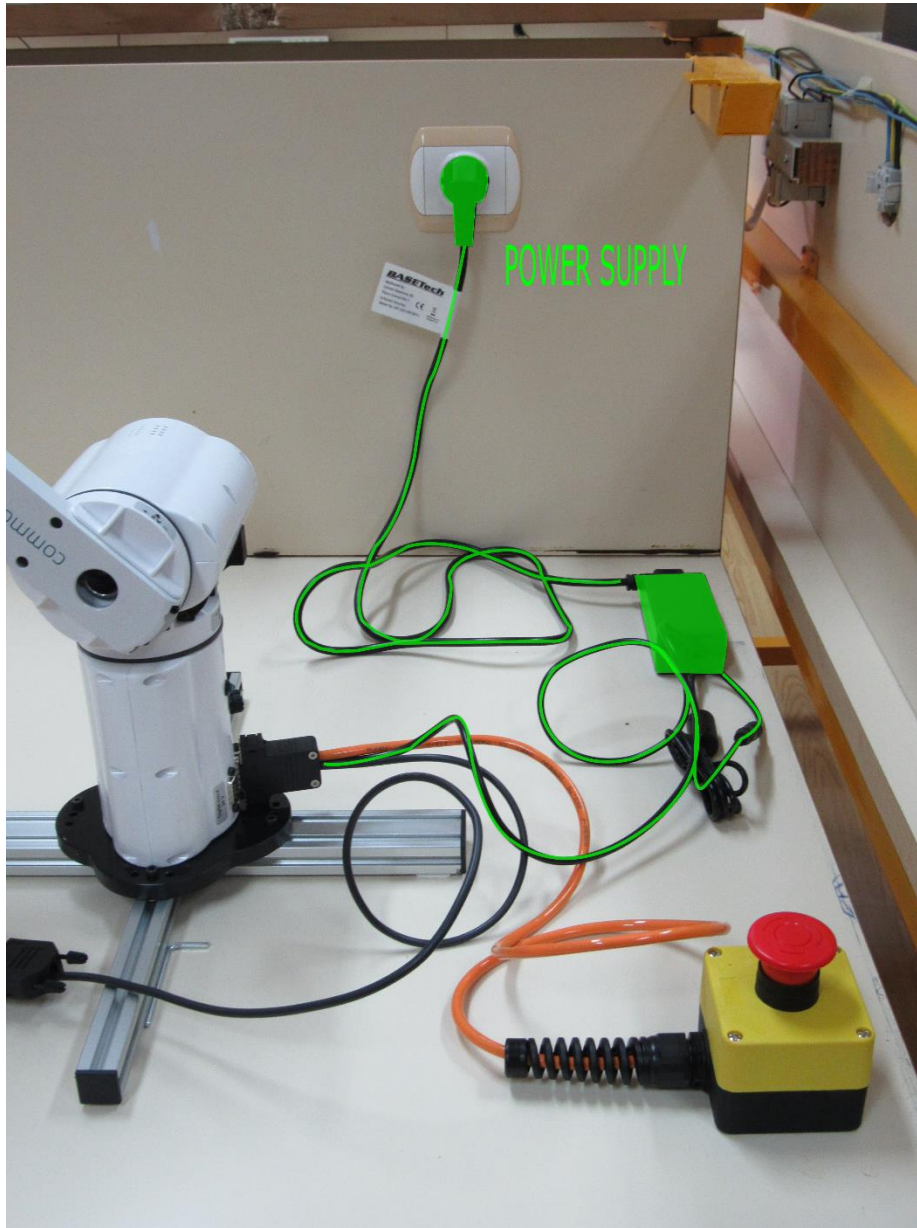


Εικόνα 21 - Καλωδίωση ρομποτικού συστήματος

Έπειτα, αφού πραγματοποιήσουμε την παραπάνω συνδεσμολογία, συνδέουμε το βασικό βύσμα στην βάση του ρομπότ, το βύσμα USB στον υπολογιστή και το βύσμα παροχής τάσης στο δίκτυο του ρεύματος. Έτσι, έχουμε πραγματοποιήσει ασφαλή τρόπο εγκατάστασης για τα ρεύματα που θα δεχθεί το ρομποτικό μας σύστημα. Σε περίπτωση όπου το ρομποτικό μας σύστημα δεν λειτουργεί παρά την σωστή εγκατάσταση των μερών του, κάνουμε τις εξής ενέργειες:

- **Ελέγχουμε τη τάση του δικτύου να είναι ενεργοποιημένη και να παίρνουμε τάση στην πρίζα.** Σε περίπτωση που αυτό δεν ισχύει, ελέγχουμε τον πίνακα ρευμάτων του δωματίου. Σε περίπτωση που όλα λειτουργούν σωστά, αλλάζουμε πρίζα και δοκιμάζουμε ξανά.
- **Ελέγχουμε το LED του ρομπότ (πίσω κοντά στη βάση του) να είναι αναμμένο.** Στην περίπτωση που δεν είναι αναμμένο τότε πρέπει να ελέγξουμε το **κουμπί πανικού** να είναι στην OFF του κατάσταση (δηλαδή, να το έχουμε γυρίσει αριστερόστροφα ώστε να έχει προεκταθεί προς τα επάνω). Στην περίπτωση που το LED συνεχίζει να μην ανάβει, θα πρέπει να εξετάσουμε την κατάσταση της **ασφάλειας** που έχει το ρομπότ στην βάση του. Αυτή είναι ασφάλεια των 2.5 Ampere. Στην περίπτωση που δεν ισχύει κάτι από τα παραπάνω, θα πρέπει να εξετάσουμε το **τροφοδοτικό** ώστε να δούμε ότι είναι σωστά συνδεδεμένο και πως δίνει τάση 12 Volt DC.

Έτσι, συνδέοντας τα επιμέρους βύσματα στον ρομποτικό βραχίονα, θα έχουμε τα εξής αποτελέσματα.



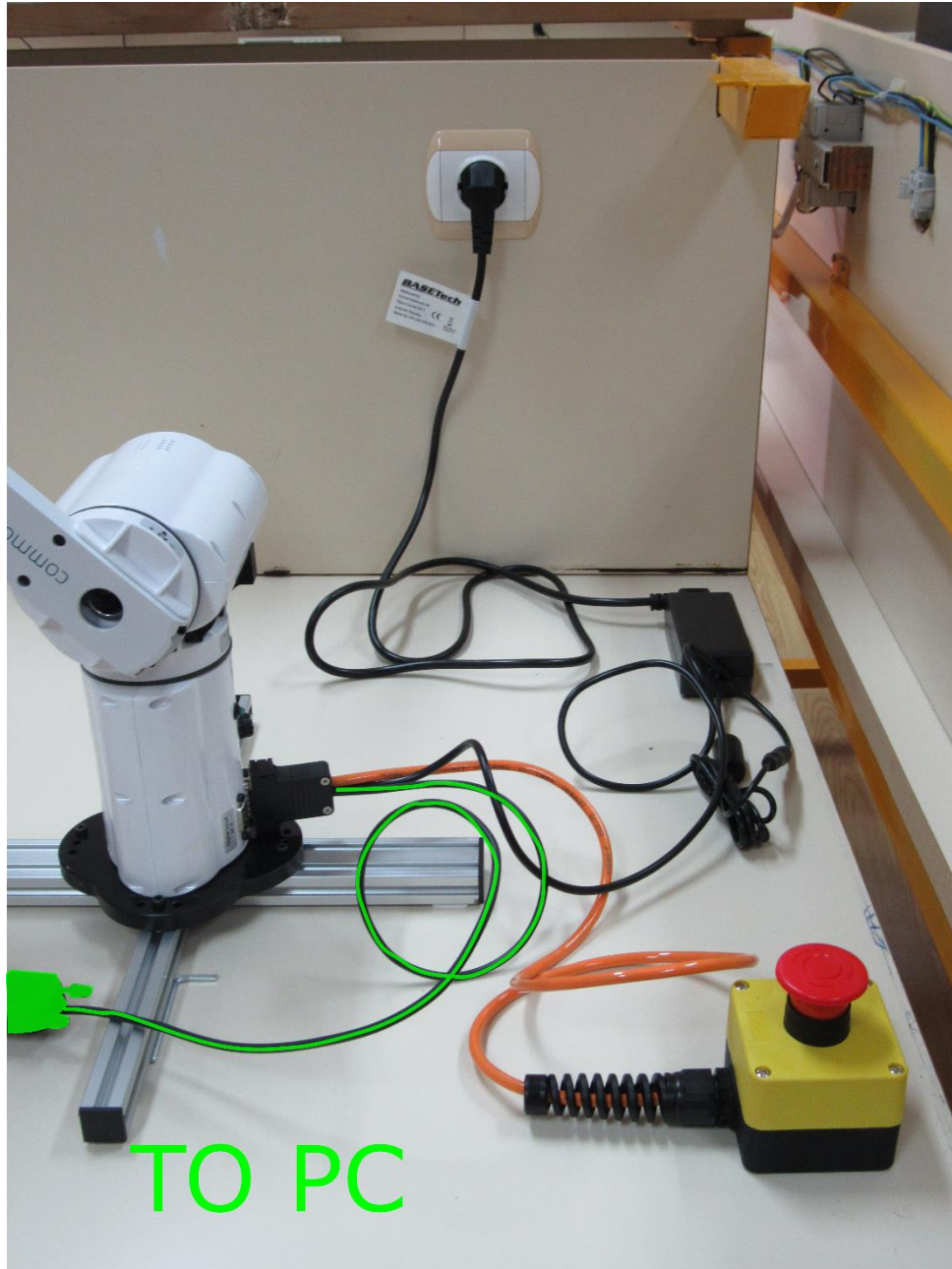
Εικόνα 22 - Συνδεσμολογία (Τροφοδοτικό)

Στην Εικόνα 22 απεικονίζεται η βασική συνδεσμολογία των επιμέρους τμημάτων του. Με πράσινο χρώμα, στιγματίζουμε το δίκτυο του τροφοδοτικού του ρομποτικού συστήματος. Όπως αναφέρθηκε και προηγουμένως, πρόκειται για ένα τροφοδοτικό των 12 Volt DC με μέγιστο ρεύμα τα 5 Ampere. Το τελικό μέρος του τροφοδοτικού καταλήγει σε ένα κοινό βύσμα όπου συναντάμε το καλώδιο πανικού και το καλώδιο επικοινωνίας του ρομποτικού συστήματος με τον υπολογιστή. Αυτό το βύσμα τοποθετείται με προσεκτικές κινήσεις στο πίσω μέρος του ρομποτικού βραχίονα.



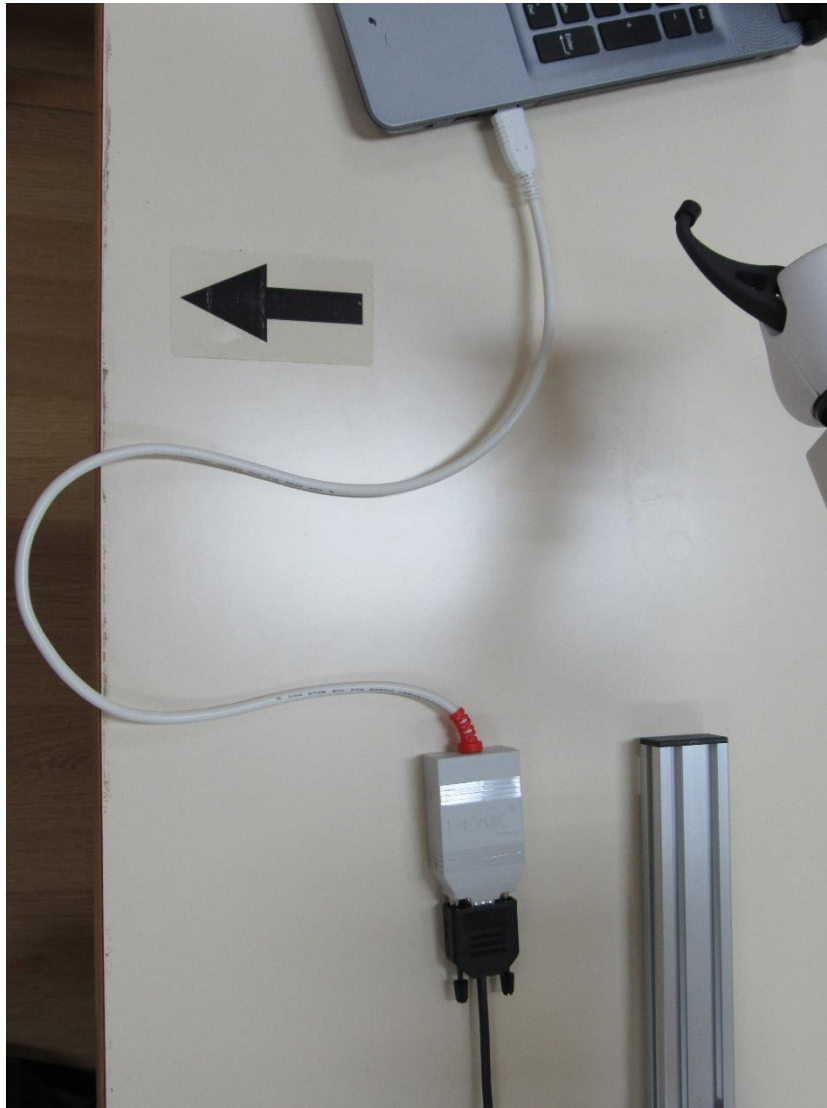
Εικόνα 23 - Συνδεσμολογία (Κουμπί άμεσης διακοπής λειτουργίας)

Στην Εικόνα 23 απεικονίζεται το σύστημα της άμεσης διακοπής λειτουργίας το οποίο συνδέεται και αυτό ταυτόχρονα σε ένα βύσμα μαζί με το τροφοδοτικό λειτουργίας αλλά και το καλώδιο επικοινωνίας με τον υπολογιστή μας. Το κουμπί αυτό είναι σε κατάσταση OFF/0 όταν το κουμπί είναι γυρισμένο αριστερόστροφα στην υψηλή του θέση. Σε εκείνη την θέση επιτρέπει τη διέλευση ρεύματος ως προς το ρομποτικό σύστημα. Στη περίπτωση που πατηθεί στην θέση ON/1, τότε διακόπτεται ακαριαία την λειτουργία του ρομποτικού βραχίονα. Αυτό το χρησιμοποιούμε για θέματα ασφαλείας των ανθρώπων που τυχόν να μπηκαν στη περιοχή λειτουργίας του ρομπότ ή ακόμη και για να προφυλάξουμε το ρομπότ από λάθος κινήσεις.



Εικόνα 24 - Συνδεσμολογία (Καλώδιο προς υπολογιστή)

Τέλος, στην Εικόνα 24, επισημαίνεται με πράσινο χρώμα το καλώδιο το οποίο βγαίνει από το κεντρικό βύσμα και καταλήγει στην ένωση ενός ειδικού προσαρμογέα για την επικοινωνία του ρομποτικού συστήματος με τον υπολογιστή. Από εκεί, χρησιμοποιούμε το παρακάτω καλώδιο, το λευκό, το οποίο συνδέεται μεταξύ του ρομποτικού βραχίονα και του υπολογιστή. Το συγκεκριμένο καλώδιο είναι ειδικό καλώδιο και το κόστος του ανέρχεται στα 185€ (Αναφορική τιμή για το έτος 2020) από την εταιρεία.



Εικόνα 25 - Ειδικό καλώδιο επικοινωνίας



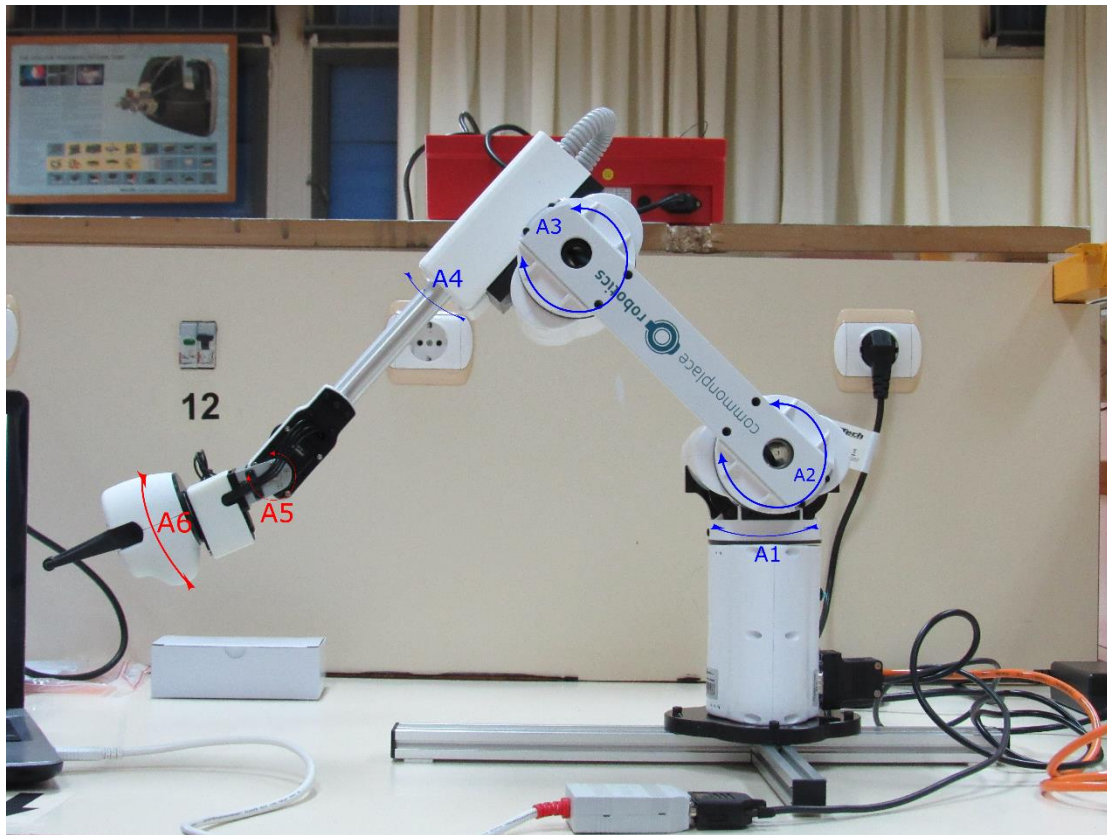
Εικόνα 26 - Σύνδεση ειδικού μετατροπέα με το ρομποτικό σύστημα (αριστερά) και σύνδεση με USB θύρα στον υπολογιστή (Δεξιά)

1.2 Ρομποτικό σύστημα

Στα πλαίσια αυτής της διπλωματικής εργασίας, ο εξοπλισμός που διαθέτουμε είναι τα εξής:

- Ρομποτικό σύστημα Mover 6 της εταιρείας Commonplace Robotics (CPR).
- Προσωπικός φορητός υπολογιστής για την επίτευξη της επικοινωνίας με το ρομποτικό σύστημα.
- Web camera για την επαφή του ρομποτικού συστήματος με το περιβάλλον και τον χώρο.

Το ρομποτικό σύστημα Mover 6 της Commonplace Robotics είναι ένας ρομποτικός βραχίονας ο οποίος προορίζεται για εκπαιδευτικούς σκοπούς και όχι ιατρικούς ή μαζικής παραγωγής. Η τελευταία του έκδοση είναι του έτους 2016. Το σύστημα αποτελείται από έξι (6) servo κινητήρες, τέσσερις (4) servo ενώσεις με κινητήρα γραναζιού και δύο (2) έξυπνους κινητήρες servo (Smart Servos) (Εικόνα 27).

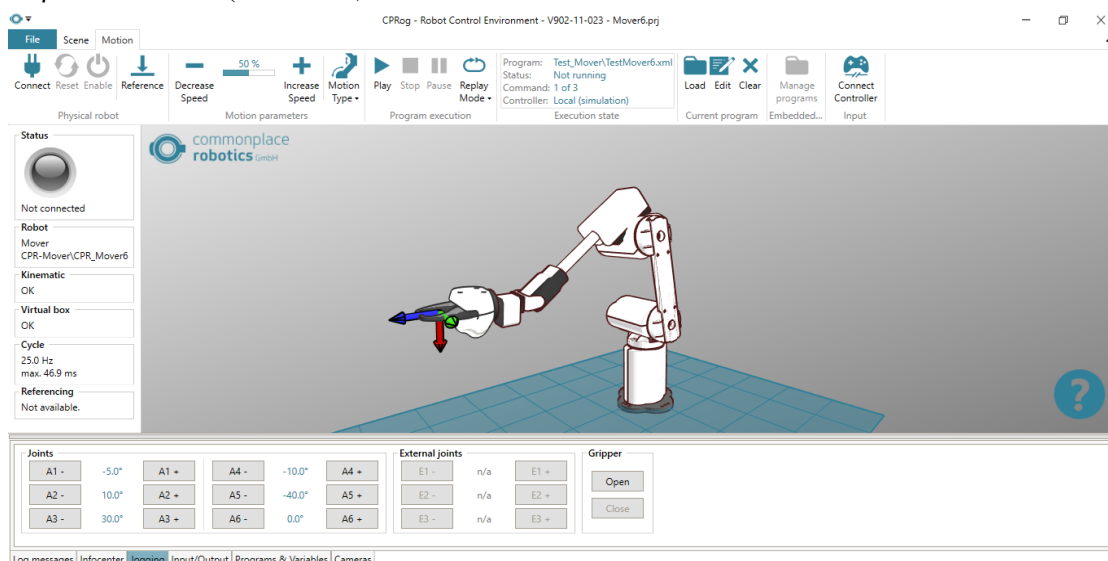


Εικόνα 27 - Εκπαιδευτικός ρομποτικός βραχίονας Mover 6 της εταιρείας Commonplace Robotics (Με μπλε επισημαίνονται οι απλοί κινητήρες servo και με κόκκινο χρώμα επισημαίνονται οι έξυπνοι κινητήρες servo)

Σε αυτό το σημείο θα ήταν χρήσιμο να επισημάνουμε τις διαφορές των δύο αυτών τύπων κινητήρων servo και γιατί βρίσκονται στην θέση που βρίσκονται. Οι απλοί και γνωστοί κινητήρες servo βρίσκονται σε διάφορες εφαρμογές της απλής έως της σύνθετης ρομποτικής. Οι κινητήρες servo δίνουν σημαντικά πλεονεκτήματα έναντι των απλών ηλεκτρικών μοτέρ καθώς η κίνηση τους είναι πολύ πιο ελεγχόμενη από τον χρήστη. Έτσι, μπορούμε να

επιτύχουμε ακόμα καλύτερα αποτελέσματα σε σχέση με πριν. Στην περίπτωση των απλών σερβο κινητήρων, η αλλαγή της θέσης τους (περιστροφή από την εφαρμογή τάσης) πραγματοποιείται με την χρήση των «Παλμών Διαμορφωμένου Πλάτους» [Pulse Width Modulated Signal (PWM)]. Η έννοια του ηλεκτρικού παλμού βασίζεται στην εφαρμογή ρεύματος συγκεκριμένης τιμής (ταχύτητα περιστροφής του κινητήρα) για συγκεκριμένο χρόνο/παλμό (διάρκεια της κίνησης για την επίτευξη της επιθυμητής θέσης). Όπως καταλαβαίνουμε από τα παραπάνω, ο τρόπος αυτός δεν αποτελεί έναν σύνθετο τρόπο λειτουργίας για τους κινητήρες, αντιθέτως είναι απλοϊκός και εύκολος. Το αρνητικό της συγκεκριμένης λειτουργίας είναι πως δεν μπορούμε με κάποιον τρόπο να εντοπίσουμε την ακριβή θέση που έχει μεταφερθεί ο συγκεκριμένος σερβο κινητήρας. Στην πράξη, δεν θα γνωρίζουμε τις ακριβείς του συντεταγμένες ώστε να τον αξιοποιήσουμε κατάλληλα σε μικρο-εφαρμογές. Αντιθέτως, οι «έξυπνοι σερβο κινητήρες» έχουν διαφορετικό τρόπο λειτουργίας ο οποίος μας βοηθάει εξαιρετικά στον εντοπισμό της θέσης τους. Η λειτουργία τους αλλάζει και δεν βασίζεται στην εφαρμογή παλμών διαμορφωμένου πλάτους. Πλέον, η επικοινωνία με τον υπολογιστή πραγματοποιείται σειριακά, το οποίο βοηθάει τον υπολογιστή να αναγνωρίζει την θέση του κάθε στιγμή όπου αυτός μεταβάλλει την θέση του. Τα υπέρ της χρήσης των έξυπνων αυτών κινητήρων είναι σίγουρα η δυνατότητα της ανάδρασης της θέσης του (feedback), η ακρίβεια που μπορεί να παρέχει στον χρήστη όταν πρόκειται για σημαντικές διαδικασίες αλλά επίσης το γεγονός πως μπορεί να χειριστεί παραπάνω από 180° (μοίρες). Βεβαίως, κάθε εφαρμογή έχει και την αρνητική του πλευρά η οποία στην προκειμένη περίπτωση είναι πως η επικοινωνία των έξυπνων κινητήρων διαφέρει όταν αλλάζουν οι κατασκευάστριες εταιρείες αλλά και πως είναι μία σχετικά νέα εφεύρεση στον χώρο της ρομποτικής, γεγονός που δεν βοηθάει ακόμα στην ανάπτυξη εμπιστοσύνης ως προς αυτά τα συστήματα.

Με την κατάλληλη χρήση των κινητήρων αυτών, θα είμαστε ικανοί να πραγματοποιήσουμε την παρούσα διπλωματική. Η διαχείριση του ρομποτικού βραχίονα αλλά και η επικοινωνία με τον ηλεκτρονικό υπολογιστή, πραγματοποιείται με τη χρήση ειδικού προγράμματος που έχει αναπτύξει η παρούσα εταιρεία. Το πρόγραμμά μας είναι φαίνεται στην παρακάτω εικόνα (Εικόνα 28):



Εικόνα 28 - CPRog program, Robot Control Environment

1.3 Τελικός επενεργητής

Ο τελικός επενεργητής (Εικόνα 29), είναι το τελικό σημείο ενός ρομποτικού συστήματος. Συγκεκριμένα, ο τελικός επενεργητής του ρομποτικού βραχίονα Mover 6 είναι το gripper (γάντζος ή). Ο τελικός επενεργητής στο συγκεκριμένο σύστημα ελέγχεται από δύο (2) έξυπνους κινητήρες servo, τους A5 και A6 (Εικόνα 27). Αυτοί είναι υπεύθυνοι να προσδιορίσουν την τελική θέση του γάντζου προκειμένου να εκτελέσει την ενέργεια που θέλουμε. Εφόσον η δαγκάνα του gripper είναι σχεδιασμένη με servo κινητήρα, μας δίνει μία βασική δυνατότητα της επιλογής του ανοίγματος που επιθυμούμε (aperture). Έτσι, ο χρήστης επιλέγει το πόσο ανοιχτή ή κλειστή επιθυμεί να είναι η δαγκάνα κατά το πρόγραμμα κίνησης του. Το μέγιστο επιτρεπτό βάρος που δύναται να σηκώσει ο γάντζος υπολογίζεται στα 400g. Αυτό βασίζεται στην προστασία των κινητήρων servo κατά την εφαρμογή της δύναμης.



Εικόνα 29 - Τελικός επενεργητής του Mover 6 (gripper)

Ο τελικός επενεργητής χειρίζεται από το πρόγραμμα στην επιλογή “gripper”. Επίσης, μπορούμε να τον ρυθμίσουμε πολύ πιο αναλυτικά ως προς το άνοιγμά του στην δημιουργία νέου προγράμματος (script). Εκεί, μπορούμε να ρυθμίσουμε το άνοιγμα (aperture) επί τις εκατό (%).

1.4 Προγραμματιστικό περιβάλλον CPRog

Όπως κάθε προγραμματιζόμενο ρομποτικό σύστημα, έτσι και το σύστημα Mover 6 της CommonPlace Robotics GmbH παρέχει ένα δικό του πρόγραμμα για την διαχείριση και τον προγραμματισμό του. Συγκεκριμένα, το πρόγραμμα αυτό λέγεται «CPRog – Robot Control Environment» και είναι ειδικά ανεπτυγμένο από την εταιρεία για τα δικά της προϊόντα ρομποτικών συστημάτων. Το πρόγραμμα αυτό το εγκαθιστάμε με τρεις τρόπους:

- Με την χρήση του CD που έρχεται μαζί με την αγορά του ρομποτικού συστήματος από την εταιρεία
- Με την επίσκεψη μας στην σελίδα της εταιρείας στον παρακάτω σύνδεσμο: (https://cpr-robots.com/download/CPRog/Installer_CPRog_V11-023.exe)
- Με το που συνδέσουμε το καλώδιο στον υπολογιστή εφόσον και αν η σύνδεση του ρομποτικού συστήματος είναι σωστή (βλέπε παράγραφο 1.1 Εγκατάσταση και καλωδίωση).

Αφού ανοίξει το απαραίτητο παράθυρο της εγκατάστασης, θα δούμε και θα ακολουθήσουμε τις παρακάτω οδηγίες:

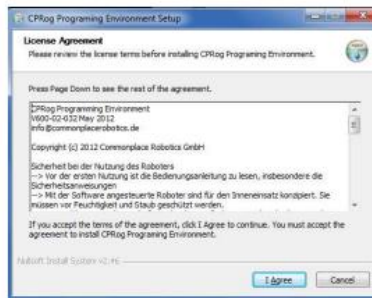
3.3 Installation of the CPRog Software



Insert the CPRog CD into the drive.

Depending on your systems configuration the CD menu will open automatically, or you have to start it manually:
D:\astorun\astorun.exe

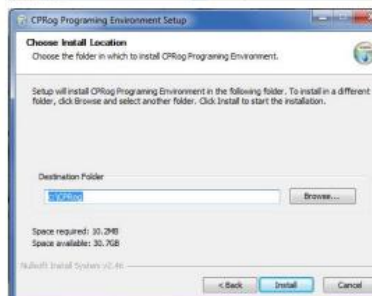
Choose the first button „CPRog Installation“



Maybe you have to allow changes to be made on your system.

After the installer has started you need to choose between English and German as language.

Then you need to confirm the license agreement.

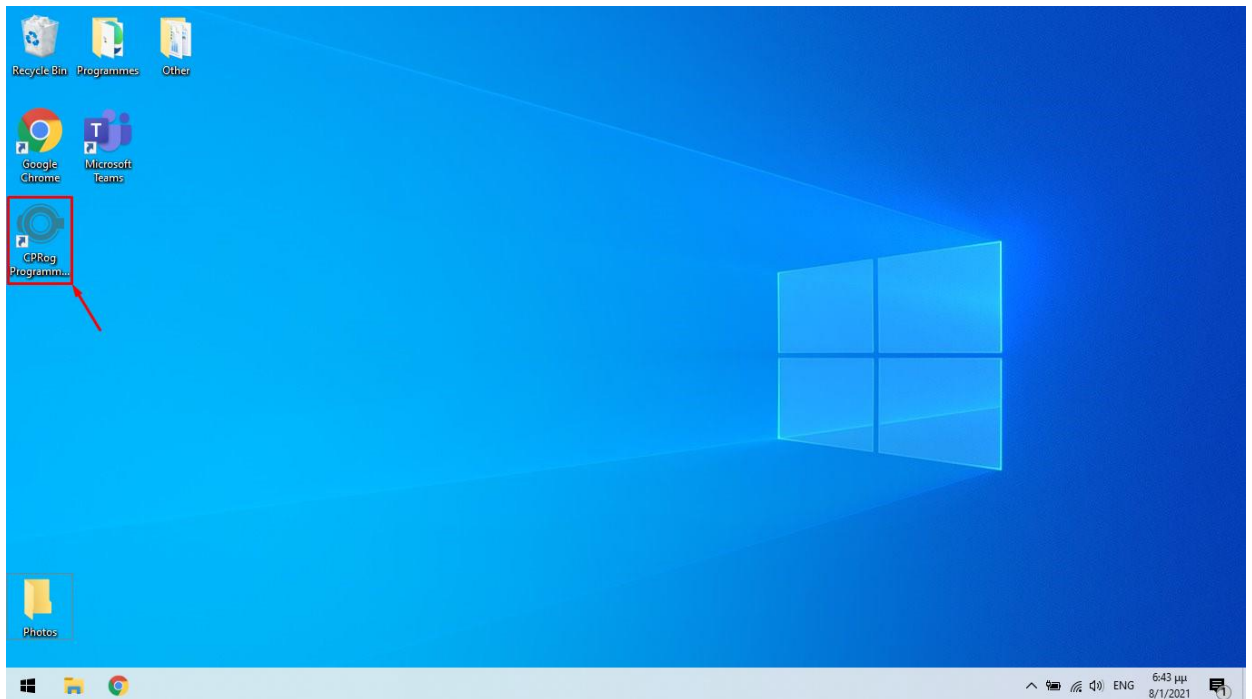


In the next step you can choose where to install CPRog. The recommended directory is
c : \CPRog.

When installing in a Windows program directory like
c : \Programme it is possible that CPRog can only be started as administrator.

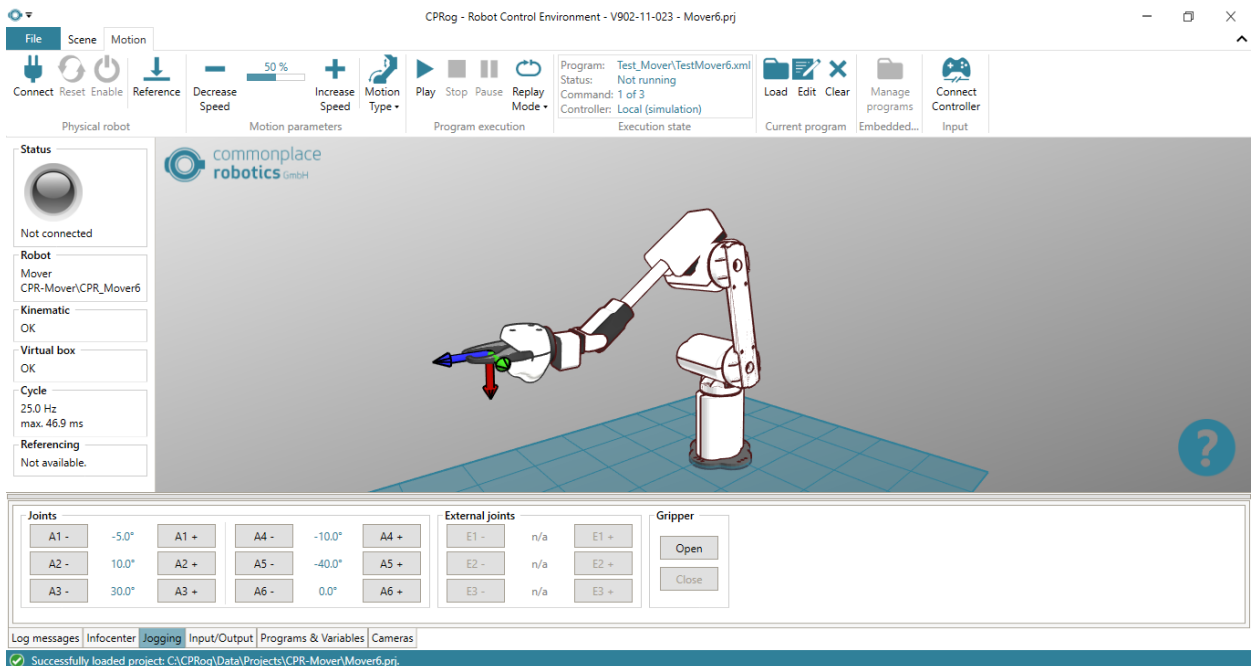
Εικόνα 30 - Οδηγός εγκατάστασης προγράμματος CPRog

Από εκεί και πέρα, αφού η εγκατάσταση πραγματοποιηθεί σωστά, θα μας εμφανιστεί το εικονίδιο του προγράμματος στην επιφάνεια εργασίας του υπολογιστή μας. Για να ανοίξουμε το πρόγραμμα, επιλέγουμε και ανοίγουμε το παρακάτω εικονίδιο (Εικόνα 31):



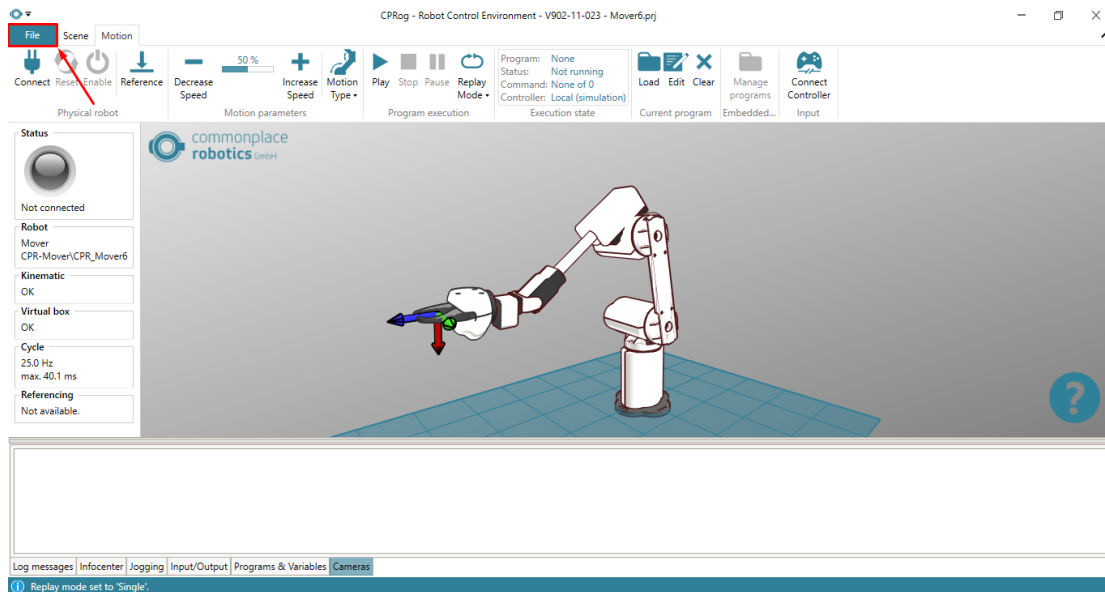
Εικόνα 31 - Εικονίδιο CPRog

Στη συνέχεια, περιμένουμε να φορτώσει το πρόγραμμα και να ανοίξει. Αφού ανοίξει, θα υποδεχτούμε στην οθόνη μας το παρακάτω προγραμματιστικό περιβάλλον (interface) για την διαχείριση του ρομποτικού μας συστήματος (Εικόνα 32):

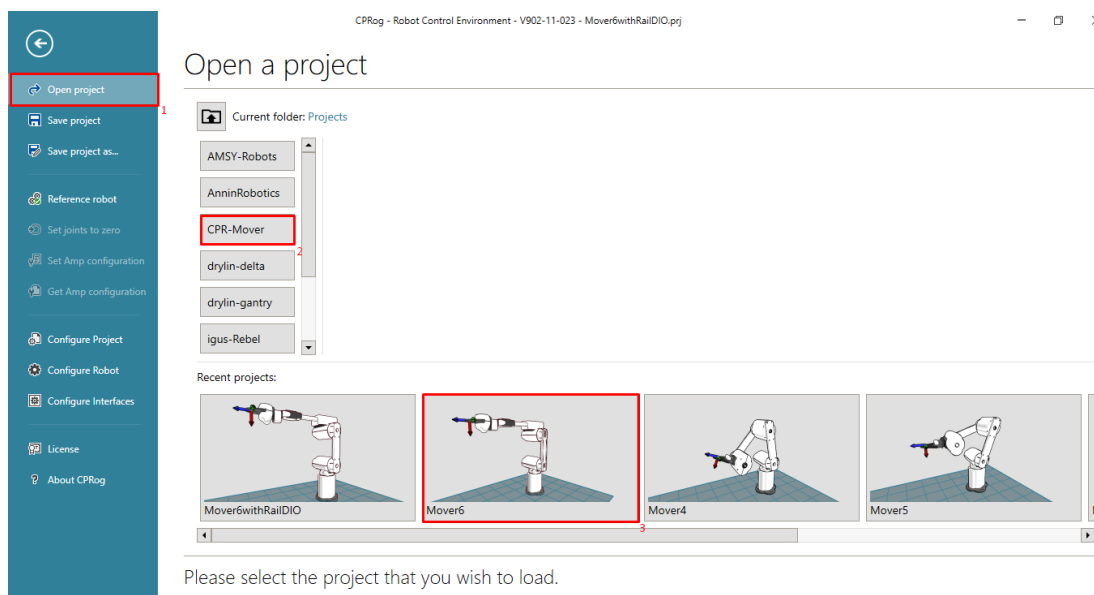


Εικόνα 32 - Προγραμματιστικό περιβάλλον του CPRog

Προκειμένου να αρχίσουμε την διαδικασία προγραμματισμού ή διαχείρισης του ρομποτικού μας συστήματος, θα πρέπει να ακολουθηθούν τα βήματα που επιδεικνύονται στις παρακάτω εικόνες:



Εικόνα 33 - CPRog (Go to file)

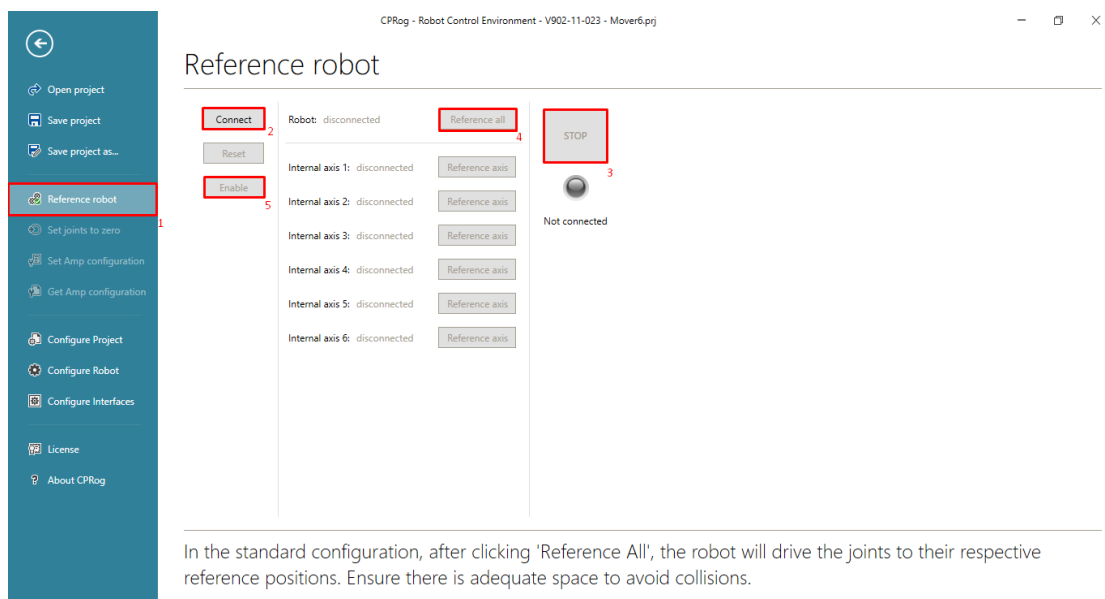


Εικόνα 34 - CPRog (Robotic System options)

Ακολουθώντας τα παραπάνω βήματα, θα έρθουμε σε επαφή με το πρόγραμμα για τον συγκεκριμένο ρομποτικό βραχίονα. Η εικόνα που θα έχουμε θα είναι ακριβώς η ίδια με αυτή που έχουμε στην Εικόνα 32. Με συνδεδεμένο το ρομποτικό σύστημα, παρατηρούμε αριστερά πως η ένδειξη φαίνεται να είναι κόκκινη. Η ένδειξη αυτή έχει τρεις καταστάσεις:

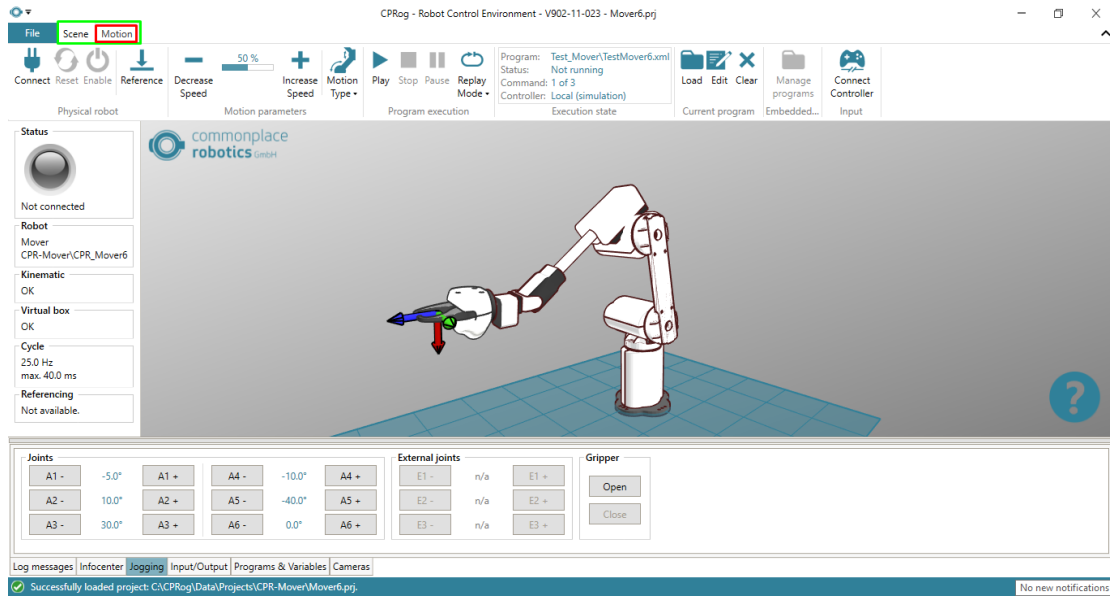
- **Γκρι χρώμα**, σημαίνει πως ο βραχίονας δεν βρίσκεται από το πρόγραμμα (πιθανών να μην είναι συνδεδεμένος σωστά ή και καθόλου)
- **Κόκκινο χρώμα**, σημαίνει πως ο βραχίονας είναι συνδεδεμένος με τον υπολογιστή αλλά δεν είναι έτοιμος να χρησιμοποιηθεί (δεν τον έχουμε ενεργοποιήσει ακόμα ή είναι πατημένο το Panic Button), και
- **Πράσινο χρώμα**, το οποίο σημαίνει πως ο ρομποτικός βραχίονας είναι έτοιμος για χρήση και είναι σωστά συνδεδεμένος και εγκατεστημένος ως προς τις καλωδιώσεις του.

Πλέον, ήρθε η ώρα να εξηγηθούν βασικότερες λειτουργίες του προγράμματος ως προς την χρήση με σκοπό την εξοικείωση του χρήστη/χειριστή με αυτό. Συγκεκριμένα, προκειμένου να ενεργοποιήσουμε το ρομποτικό σύστημα, αυτό το κάνουμε με το κουπί “**Enable**”. Στην περίπτωση όμως που το κουπί αυτό δεν είναι διαθέσιμο, ακολουθούμε την παρακάτω διαδικασία που φαίνεται στην εικόνα (Εικόνα 35):



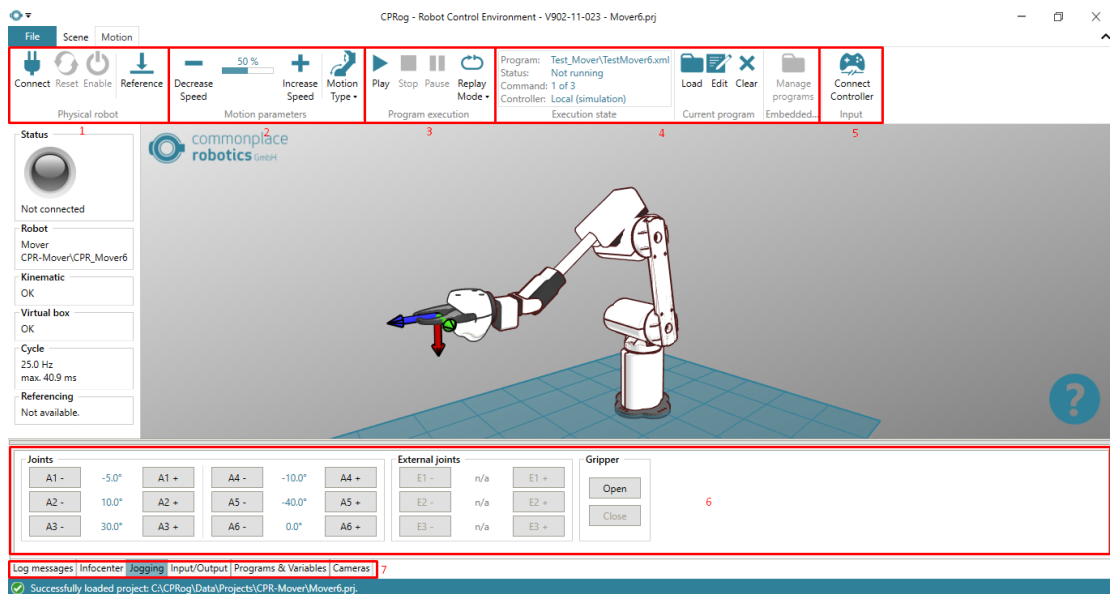
Εικόνα 35 - CPRog (Activity troubleshooting)

Το πρόγραμμα CPRog έχει δύο βασικές καρτέλες στην επιφάνεια του. Συγκεκριμένα, η πρώτη επιφάνεια που μας επιτρέπει τον χειρισμό του ρομποτικού βραχίονα είναι η καρτέλα “**Motion**” (Εικόνα 36).



Εικόνα 36 - CPRog (Motion)

Στην καρτέλα αυτή μπορούμε να μετακινήσουμε τον ρομποτικό βραχίονα, να ελέγχουμε τη «δαγκάνα» του τελικού επενεργητή, έχουμε πρόσβαση σε ρυθμίσεις ταχύτητας της κίνησης, να φορτώσουμε έτοιμα προγράμματα ή να δημιουργήσουμε εμείς νέα και αρκετά ακόμα (Εικόνα 37).

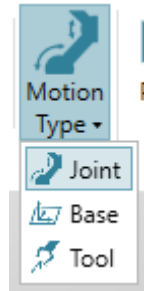


Εικόνα 37 - CPRog (Motion interface explanation)

Συγκεκριμένα, παρατηρούμε τις εξής λειτουργίες:

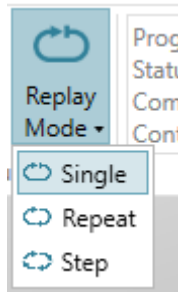
- **Περιοχή αρχικών καταστάσεων (1)**, όπου εκεί μπορούμε να πραγματοποιήσουμε νέα σύνδεση στον υπολογιστή με τον ρομποτικό μας βραχίονα, την επαναφορά των επιλογών του, την ενεργοποίηση του (πράσινη ένδειξη) και τον ορισμό των αρχικών σημείων του ρομποτικού συστήματος.

- **Περιοχή επιλογής ταχύτητας κίνησης και τρόπου χειρισμού (2)** , όπου εκεί επιλέγουμε την ταχύτητα με την οποία επιθυμούμε να πραγματοποιηθεί η κίνηση που έχουμε σκοπό να κάνουμε αλλά και τον τρόπο με τον οποίο θα χειριζόμαστε την κίνηση (Motion type) του ρομποτικού συστήματος (Εικόνα 38).



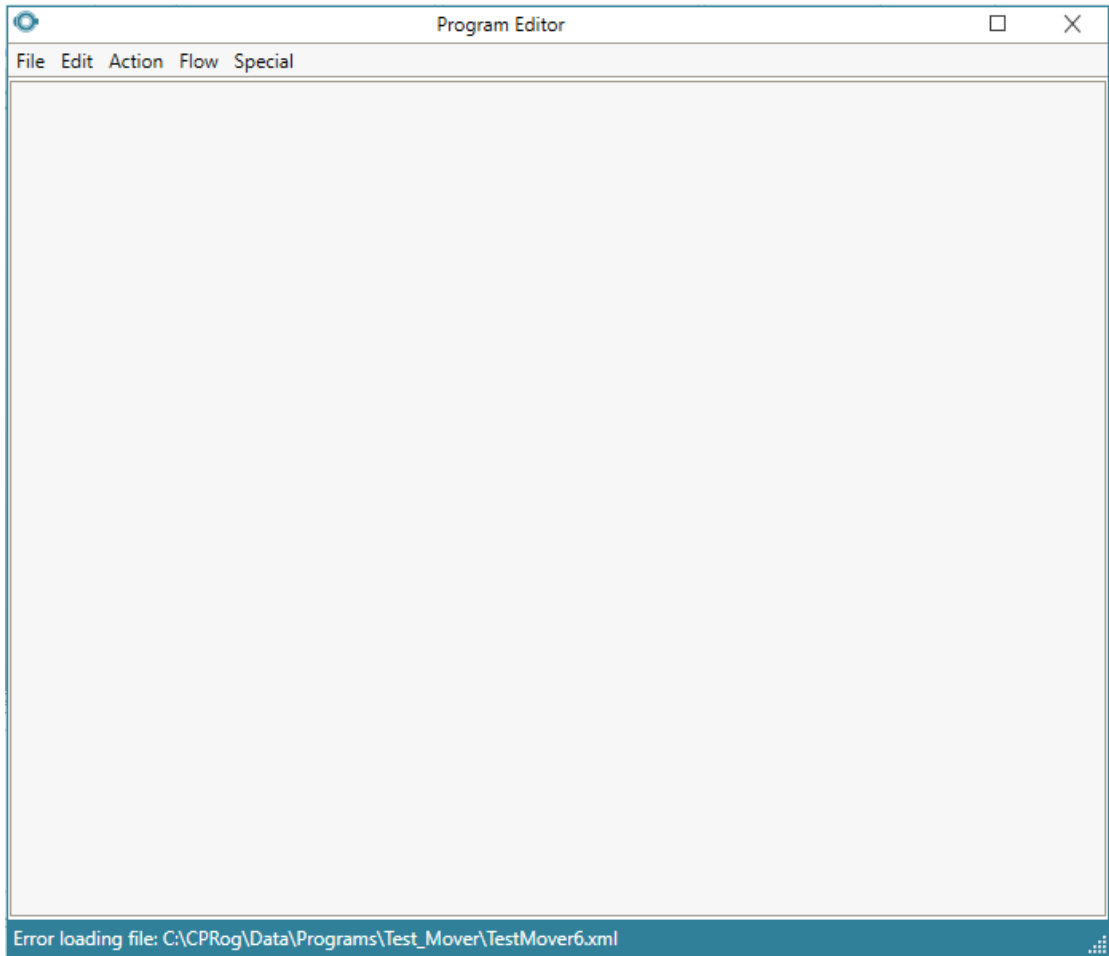
Εικόνα 38 - CPRog (Motion Type)

- **Περιοχή επιλογής έναρξης, διακοπής, παύσης και συνεχόμενης, τμηματικής ή επαναλαμβανόμενης κίνησης (Εικόνα 39) (3)** ,



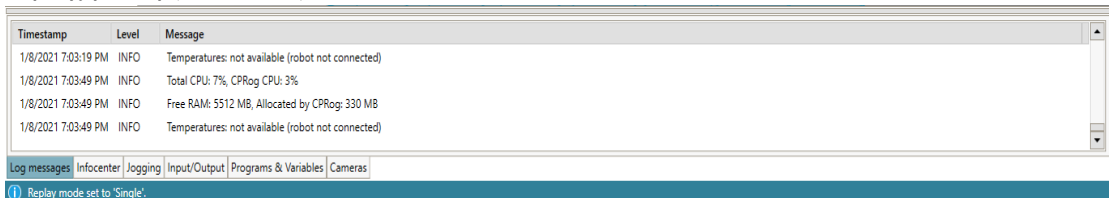
Εικόνα 39 - CPRog (Replay Mode)

- **Περιοχή φόρτωσης, επιλογής ή δημιουργίας νέου προγράμματος (4)** , όπου εδώ μπορούμε να επιλέξουμε ένα από τα έτοιμα προγράμματα που έχει ήδη μέσα το πρόγραμμα μας. Επίσης, υπάρχει η δυνατότητα δημιουργίας νέου προγράμματος (script) με το οποίο θα έχουμε την δυνατότητα να καλύψουμε τις δικές μας ανάγκες. Το προγραμματιστικό περιβάλλον δεν αποτελείται από κώδικα αλλά από έτοιμες ρυθμίσεις πάνω σε έτοιμες και βασικές κινήσεις που μας παρέχει το πρόγραμμα (Εικόνα 40).



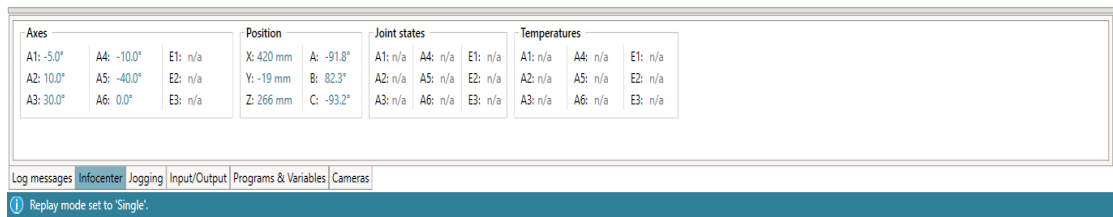
Εικόνα 40 - CPRog (New program script)

- **Περιοχή επιλογής χειριστηρίου (Controller) (5)**
- **Βασική περιοχή ελέγχου και χειρισμού του ολικού συστήματος από τον χρήστη (6) , και**
- **Περιοχή επιλογής δυνατοτήτων (7) που θέλει ο χρήστης να παρακολουθεί στην περιοχή 6 . Συγκεκριμένα στην περιοχή αυτή θα συναντήσουμε επιλογές όπως το “Log Messages” το οποίο μας επιτρέπει να έχουμε επαφή με την λειτουργία του ρομποτικού συστήματος κάθε στιγμή στην οποία αυτό εκτελεί μία κίνηση. Για παράδειγμα, στην περίπτωση που το ρομποτικό μας σύστημα εκτελέσει μία κίνηση και βρει εμπόδιο κατά την πορεία της κίνησης του, αυτή θα φανεί σαν μήνυμα στην περιοχή αυτή (Εικόνα 41).**



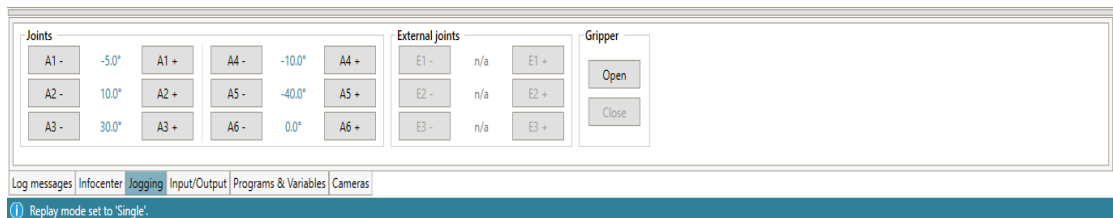
Εικόνα 41 - CPRog (Log messages)

Στην περιοχή “**Infocenter**” μπορούμε να δούμε κάθε στιγμή τις τιμές της κάθε άρθρωσης του ρομποτικού βραχίονα καθώς και άλλες λεπτομέρειες. Οι τιμές αυτές είναι αναφορικές και δεν δέχονται κάποια μεταβολή από τον χρήστη (Εικόνα 42).



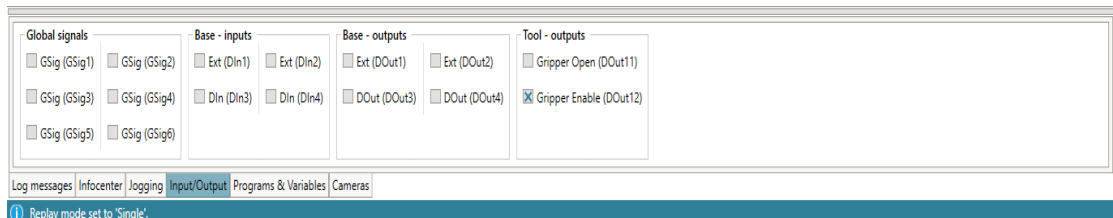
Εικόνα 42 - CPRog (Infocenter)

Στην περίπτωση που επιθυμούμε να πραγματοποιήσουμε κάποια ή κάποιες ενέργειες με τη χρήση των επιλογών στο ποντίκι του υπολογιστή, αυτή επιτυγχάνεται στην περιοχή του “**Jogging**”. Στη συγκεκριμένη περιοχή μας επιτρέπεται να ελέγξουμε την κίνηση του ρομποτικού συστήματος τμηματικά (Εικόνα 43).



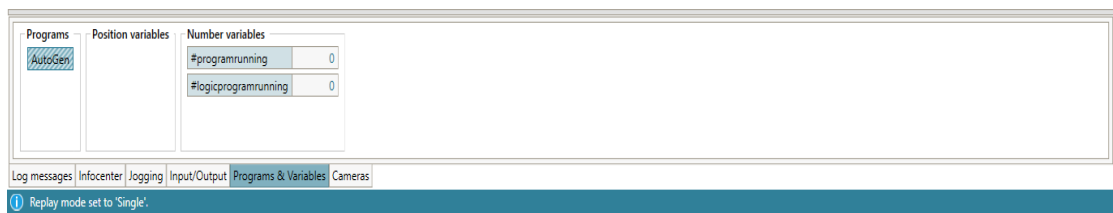
Εικόνα 43 - CPRog (Jogging)

Μετά έχουμε την περιοχή “**Input/Output**” όπου εκεί μας δίνεται η δυνατότητα επιλογής κάποιων ειδικών και συγκεκριμένων παραμέτρων που υπάρχουν στον ρομποτικό βραχίονα (Εικόνα 44).



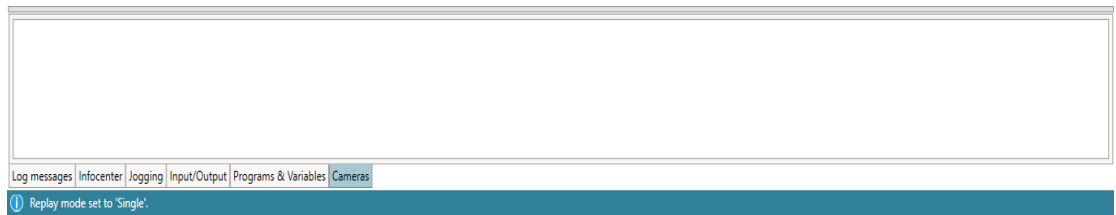
Εικόνα 44 - CPRog (Input/Output)

Η επόμενη καρτέλα αναφέρεται σε μία ενδεικτική περιοχή όπου μπορούμε να διακρίνουμε τον φόρτο του προγράμματος κατά την εκτέλεση του αλλά και διάφορες άλλες σχετικές ενδείξεις που μπορούν να μας φανούν χρήσιμες (Εικόνα 44).



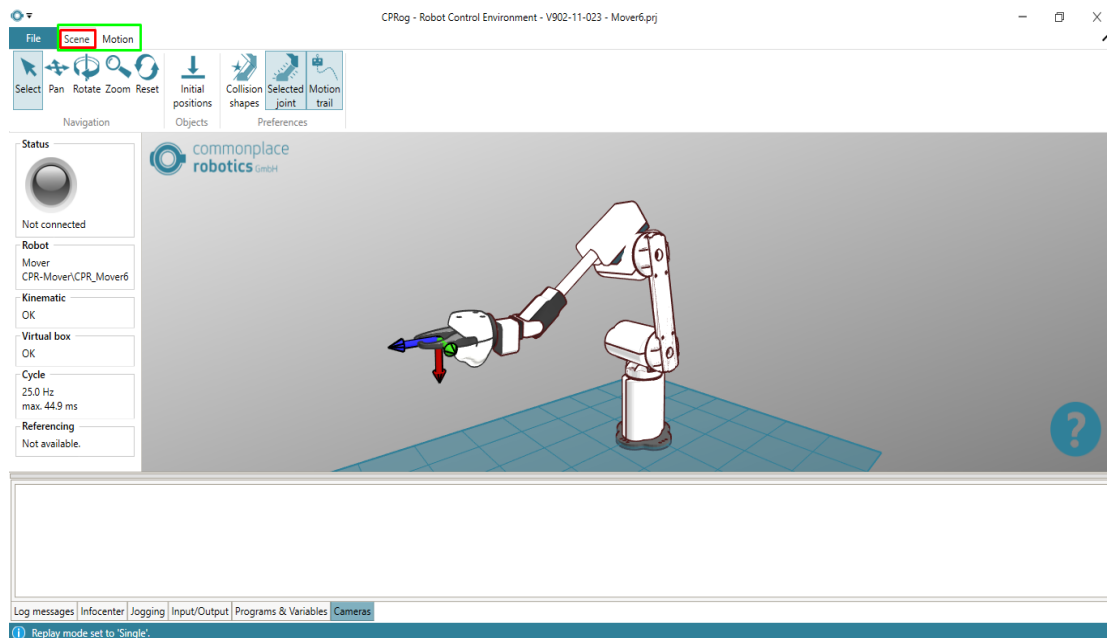
Εικόνα 45 - CPRog (Programs & Variables)

Τέλος, διακρίνουμε την τελευταία επιλογή που έχουμε, την περιοχή της κάμερα στην οποία φαίνονται ρυθμίσεις για την κάμερα σε περίπτωση που χρησιμοποιούμε μία για κάποιον σκοπό (Εικόνα 46).



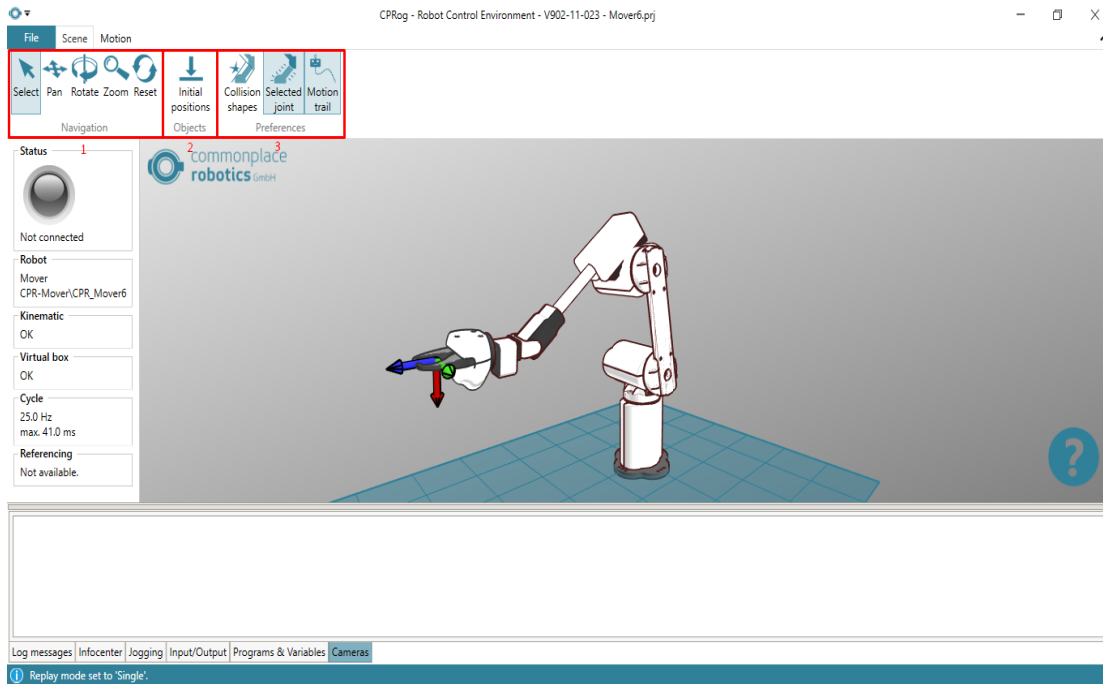
Εικόνα 46 - CPRog (Camera)

Η επόμενη καρτέλα αναφέρεται στην κίνηση στο περιβάλλον του ρομποτικού συστήματος ως προς την όψη του χειριστή (Point Of View – POV). Συγκεκριμένα, η περιοχή “Scene” μας δίνει την δυνατότητα να μπορούμε να επιλέξουμε εμείς το σημείο από το οποίο θέλουμε εμείς να παρακολουθούμε τον βραχίονα κατά την κίνηση του στο προγραμματιστικό περιβάλλον του (Εικόνα 47).



Εικόνα 47 - CPRog (Scene)

Στην περιοχή της καρτέλας “Scene” διακρίνουμε διάφορες επιλογές για την ρύθμιση της γωνίας προβολής της αναπαράστασης του ρομποτικού συστήματος. Συγκεκριμένα, τα παρατηρούμε στην παρακάτω εικόνα (Εικόνα 48):



Εικόνα 48 - CPRog (Scene explanation)

Εκεί, μπορούμε να διακρίνουμε τις εξής επιλογές και περιοχές:

- **Περιοχή ρύθμισης γωνίας θέασης της αναπαράστασης (1)** , όπου εδώ μπορούμε να επιλέξουμε, να μετακινήσουμε, να στρέψουμε, να μεγεθύνουμε θετικά ή αρνητικά το περιβάλλον ή και να το φέρουμε στην αρχική του θέση.
- **Περιοχή ανάθεσης θέσεων των αντικειμένων (2)** , όπου στην συγκεκριμένη επιλογή μπορούμε να επιλέξουμε την τοποθεσία στην οποία θα υπάρχουν κάποια αντικείμενα τα οποία μας ενδιαφέρουν. Έτσι, με αυτή την επιλογή, τα ορίζουμε τόσο σαν αντικείμενο όσο και σαν θέση.
- **Περιοχή επιλογής τρόπων προβολής και απεικόνισης του ρομποτικού βραχίονα (3)** , όπου εδώ μπορούμε να επιλέξουμε τον τρόπο απεικόνισης του ρομποτικού βραχίονα στο προγραμματιστικό του περιβάλλον.

Αξίζει να σημειωθεί σε αυτό το σημείο ότι η περιοχή **6** και **7** της εικόνας 37, δεν αλλάζει στην περίπτωση που αλλάζουμε τις καρτέλες μεταξύ τους. Αυτό σημαίνει πως οι περιοχές αυτές δεν ανήκουν σε κάποια από τις δύο καρτέλες και πως κατά την διάρκεια του χειρισμού του ρομποτικού βραχίονα μέσω του προγράμματος, θα είμαστε σε θέση πάντα να έχουμε επαφή με αυτές τις περιοχές.

1.5 – Ένωση ψηφιακού και πραγματικού κόσμου

Η ενότητα αυτή αποτελεί ένα ζωτικό μέρος της διπλωματικής αυτής. Η εκτέλεση μίας ιδέας σαν αυτή, προϋποθέτει την ύπαρξη ενός πλάνου. Στην συγκεκριμένη περίπτωση, προσπαθούμε να δημιουργήσουμε έναν τρόπο επικοινωνίας ανάμεσα σε εμάς (ιδέα) και τον ρομποτικό βραχίονα (εκτελεστικό όργανο). Έτσι, η αρχική ιδέα φαίνεται να είναι η εξής:

ΙΔΕΑ → ΜΗΧΑΝΙΣΜΟΣ → ΑΠΟΤΕΛΕΣΜΑ

Βέβαια, αυτό αποτελεί μία απλούστευση της πραγματικής ιδέας. Κατά την περίπτωση που τοποθετούμε συστήματα όπως μία κάμερα, αυτομάτως εντάσσουμε στην πράξη και τον χώρο. Ο πραγματικός χώρος, αποτελείται από τρεις βασικούς άξονες, τον άξονα του ύψους (Άξονας Y | Y Axes), τον άξονα του μήκους (Άξονας X | X Axes) και τέλος τον άξονα πλάτους (Άξονας Z | Z Axes). Έτσι, η προηγούμενη ιδέα αρχίζει να αλλάζει προσεγγίζοντας ακόμα περισσότερο τις πραγματικές συνθήκες. Οπότε, η διαδικασία αρχίζει και γίνεται ως εξής:

ΕΠΙΘΥΜΙΑ → ΧΩΡΟΣ → ΜΗΧΑΝΙΣΜΟΣ → ΑΠΟΤΕΛΕΣΜΑ

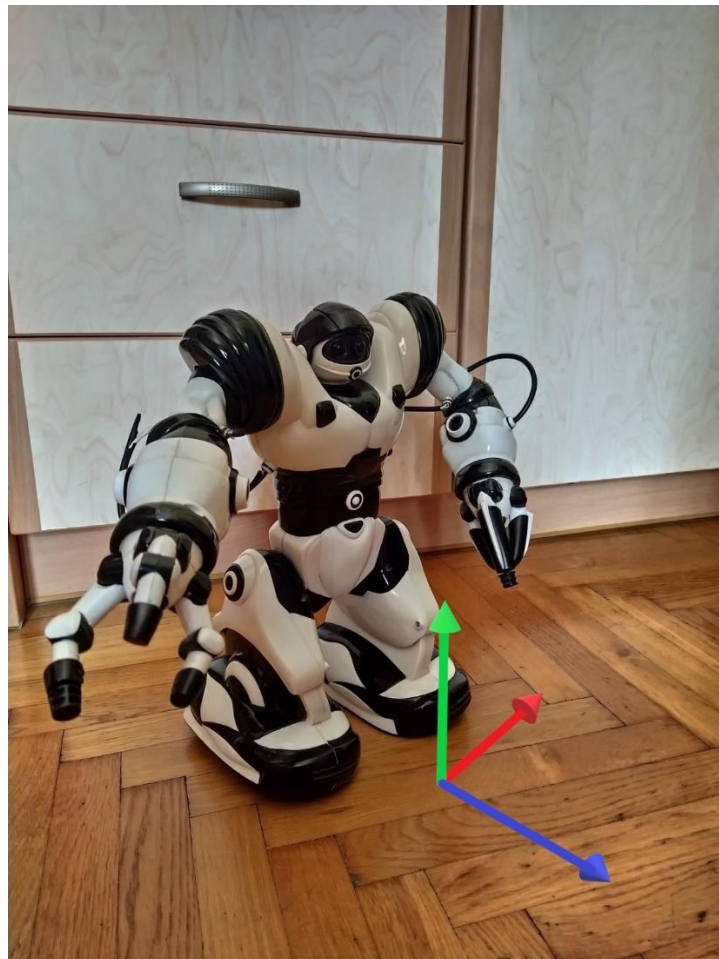
Η διαδικασία άλλαξε και χωρίζεται στις εξής κατηγορίες:

- **ΕΠΙΘΥΜΙΑ**, θεωρούμε την επιλογή του χειριστή στην επιλογή συγκεκριμένου αντικειμένου από τον χώρο
- **ΧΩΡΟΣ**, θεωρούμε την ύπαρξη του φυσικού περιβάλλοντος ο οποίος βρίσκεται εντός του πεδίου ενδιαφέροντος και ικανοτήτων του ρομποτικού μας βραχίονα
- **ΜΗΧΑΝΙΣΜΟΣ**, θεωρούμε τον κώδικα ο οποίος αποτελεί την μετασχηματιστική διαδικασία ώστε να ικανοποιηθεί η επιθυμία του χρήστη η οποία έγινε με κριτήριο το περιεχόμενο του χώρου για να μας αποδοθεί το επιθυμητό **αποτέλεσμα**, και
- **ΑΠΟΤΕΛΕΣΜΑ**, θεωρούμε την επιτυχή πραγματοποίηση της επιθυμίας του χρήστη μέσω της διαδικασίας του **μηχανισμού** (κώδικα) που δημιουργήσαμε.

Η ιδιαιτερότητα της διαδικασίας αυτής, είναι πως η ένωση του ψηφιακού (κάμερα) με τον πραγματικό χώρο (περιβάλλον) δημιουργεί κάποιες προϋποθέσεις «μετάφρασης» από την μία πλευρά, στην άλλη. Με απλούστερα λόγια, θα πρέπει να μεταφράσουμε με κάποιον τρόπο την τοποθεσία του αντικειμένου που μας ενδιαφέρει και που εντοπίζουμε μέσω της εικόνας της κάμερας στο πρόγραμμα ώστε να γίνει ένας σωστός μετασχηματισμός των εντολών του προγράμματος. Για παράδειγμα, παρατηρούμε τον ρομποτικό βραχίονα όπου κάθε του άρθρωση βρίσκεται στις 0° ως προς τα πρότυπα του προγράμματος CPRog. Στην πραγματικότητα όμως, ως προς τη δική μας οπτική γωνία, δεν βρίσκονται στις 0° κάθε του άρθρωση. Ας υποθέσουμε πως υπάρχει ένα αντικείμενο στον χώρο με συντεταγμένες $\mathbf{A}(X,Y,Z)$ και επιθυμούμε να μεταφέρουμε τον ρομποτικό βραχίονα στο σημείο με τις συντεταγμένες αυτές. Ως προς την πραγματικότητα, εάν τοποθετούσαμε τις αρθρώσεις γεωμετρικά με τέτοιο τρόπο ώστε να φτάσει το σημείο A, οι τιμές που θα βρίσκαμε ανά άρθρωση δεν θα συμπίπτουν με τις τιμές που φαίνονται στο πρόγραμμά μας. Έτσι, θα πρέπει να βαθμονομήσουμε το πρόγραμμα με τις συντεταγμένες του πραγματικού περιβάλλοντος. Αυτή η διαδικασία πραγματοποιείται παίρνοντας γνωστές αποστάσεις, σημειώνοντας τις τιμές του βραχίονα και δημιουργώντας εξισώσεις που μπορούν να ικανοποιήσουν αυτές τις κινήσεις.

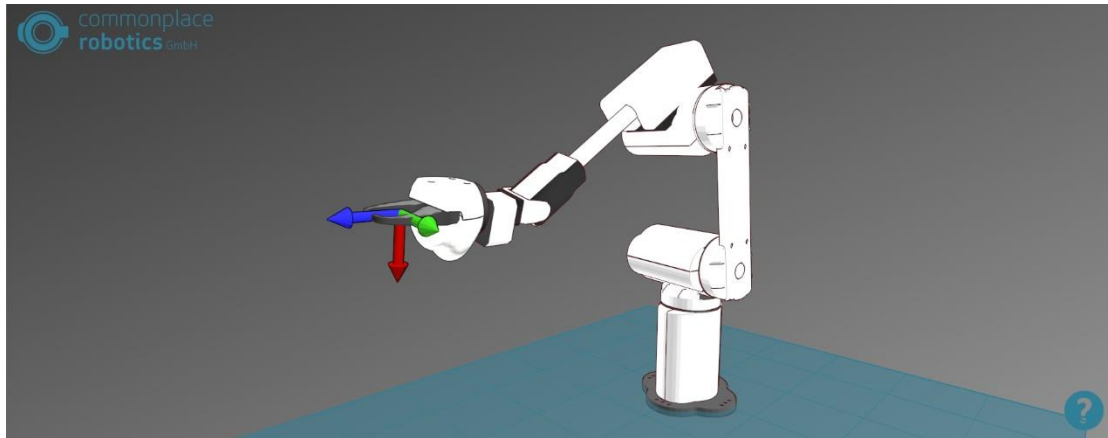
Στη συγκεκριμένη περίπτωση, θα γνωρίζουμε το ύψος το οποίο θα κινείται ο ρομποτικός βραχίονας αλλά και το πόσο θα πρέπει να «βυθίζεται» ανάλογα το μήκος που έχει διανύσει ώστε να λάβει το αντικείμενο με τον τελικό του επενεργητή. Όλες όμως αυτές οι διαδικασίες θα πρέπει να ελεγχθούν, να καταγραφούν και να προσαρμοστούν ανάλογα με τις ανάγκες του χώρου και του προγράμματος. Η διαδικασία θα ήταν πολύ πιο εύκολη στην περίπτωση που το ρομποτικό σύστημα πραγματοποιούσε ίδιες κινήσεις κάθε φορά. Για παράδειγμα, στην περίπτωση που θέλαμε να τοποθετήσουμε μία κάμερα στο άκρο του ρομποτικού βραχίονα με την οποία να διάβαζε το QR Code των αντικειμένων που κινούνται με σταθερή ταχύτητα επάνω σε μία βιομηχανική πλατφόρμα ή ακόμα να πραγματοποιούσε ποιοτικό έλεγχο του αντικειμένου, η κίνηση του βραχίονα θα βασιζόταν στις ίδιες μοίρες και τις ίδιες τοποθεσίες στον χώρο. Έτσι, θα δημιουργούσαμε μία επαναλαμβανόμενη κίνηση που θα την ακολουθούσε εις βάθος χρόνου. Στην συγκεκριμένη περίπτωση όμως, καθώς ο χρήστης έχει πλέον την επιλογή ενός αντικειμένου από τον χώρο, η διαδικασία αυτή γίνεται πιο σύνθετη και απαιτεί διαφορετικούς τρόπους μεταχείρισης.

Αν θεωρήσουμε τους άξονες του περιβάλλοντος μας ως Y το ύψος, X το μήκος και Z το πλάτος, θα πρέπει να τους βαθμονομήσουμε ανάλογα με τους πρότυπους άξονες του προγράμματος CPRog. Συγκεκριμένα, όπως φαίνεται στη παρακάτω εικόνα (Εικόνα 49):



Εικόνα 49 - Πραγματικοί άξονες (Μπλέ άξονας: X (μήκος), Πράσινος άξονας: Y (ύψος), Κόκκινος άξονας: Z (πλάτος))

Όπως μπορούμε να παρατηρήσουμε στην εικόνα 49, υπάρχει το σύστημα αξόνων το οποίο χρησιμοποιούμε για κάθε χαρακτηρισμό ενός αντικειμένου στον χώρο. Συγκεκριμένα, όπως αναγράφεται και στην περιγραφή της εικόνας, ο χρωματισμένος με μπλε χρώμα άξονας αντιπροσωπεύει τον άξονα X (άξονας του μήκους), ο κάθετος άξονας που είναι πράσινος αντιπροσωπεύει τον άξονα Y (άξονας του ύψους) και τέλος, ο κάθετος στους προηγούμενους δύο άξονας χρωματισμένος με κόκκινο χρώμα, αντιπροσωπεύει τον άξονα Z (άξονας του πλάτους). Αντίστοιχα, οι άξονες του προγράμματος CPRog έχουν την εξής ιδιαιτερότητα σε σχέση με τους πραγματικούς (Εικόνα 50):



Εικόνα 50 - Πρότυπο αξόνων του προγράμματος CPRog

Σύμφωνα με το πρόγραμμα και την κίνηση του, ο χρωματισμένος με μπλε χρώμα είναι ο άξονας X, ο χρωματισμένος με πράσινο χρώμα άξονας είναι ο άξονας Y και τέλος ο χρωματισμένος με κόκκινο χρώμα άξονας, είναι ο άξονας Z. Η διαφορά όμως της πραγματικότητας με το σύστημα αξόνων που παρατηρούμε στο πρόγραμμα, είναι πως υπάρχει μία αλλαγή ως προς τον άξονα X. Συγκεκριμένα, πρόκειται για μία αλλαγή των αξόνων (-90°) με σταθερό τον άξονα X κατά ένα τέταρτο του κύκλου. Έτσι, η αντιστοιχία του πραγματικού κόσμου με τον προγραμματιστικό χώρο του βραχίονα είναι ο εξής:

- Ο άξονας X του πραγματικού χώρου είναι ο άξονας X του προγράμματος CPRog
- Ο άξονας Y του πραγματικού χώρου είναι ο άξονας Z του προγράμματος CPRog
- Ο άξονας Z του πραγματικού χώρου είναι ο άξονας Y του προγράμματος CPRog

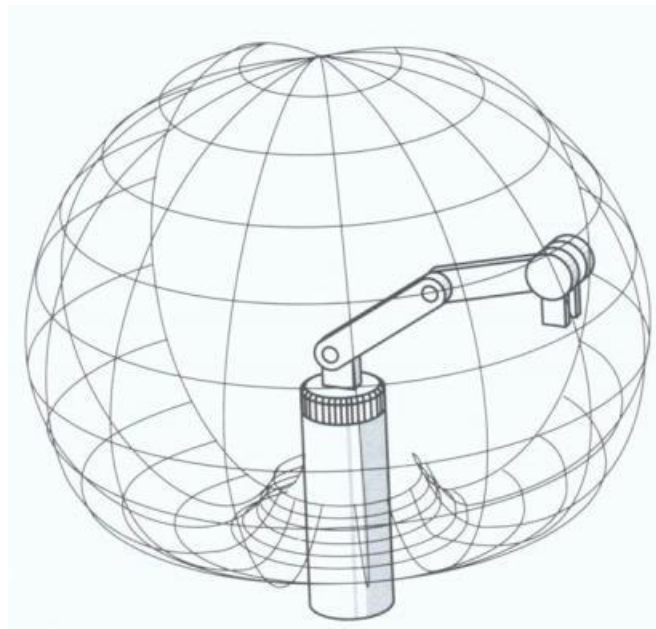
Αυτό ήταν το πρώτο βήμα της βαθμονόμησης μας, η βαθμονόμηση των αξόνων ως προς το οπτικό πεδίο, δηλαδή την κατανόηση και την συμφωνία μας για την κίνηση του βραχίονα στον χώρο σύμφωνα με το πρότυπο που εμείς θέλουμε. Όμως, η βαθμονόμηση δεν τελειώνει εδώ. Εφόσον βρήκαμε την μετάφραση του χώρου ως προς τις κατευθύνσεις, πλέον θα πρέπει να δώσουμε έναν τρόπο μετάφρασης των αποστάσεων του πραγματικού χώρου για να βοηθηθεί ο ρομποτικός βραχίονας και να τις κατανοήσει. Ένα παράδειγμα στο δικό μας πρόβλημα, είναι η μαθηματική διαδικασία που θα πρέπει να αποδοθεί στον κώδικα προκειμένου ο βραχίονας να κατανοήσει την τοποθεσία του αντικειμένου. Έτσι, εάν ένα αντικείμενο βρίσκεται στο σημείο A που αναφέραμε παραπάνω $[A(X,Y,Z)]$ τότε θα πρέπει να φτιάξουμε τρεις μαθηματικούς τύπους που θα υπολογίζουν την απομάκρυνση (X), την περιστροφή (Y) και τέλος το ύψος (Z) του αντικειμένου. Κάθε μας πληροφορία θα έρχεται από κάποιες αριθμητικές συντεταγμένες οι οποίες θα προέρχονται από την χρήση της κάμερας μέσω ειδικής βιβλιοθήκης του PyCharm. Αυτοί οι αριθμοί θα πρέπει να προσαρμοστούν κατάλληλα με τέτοιο τρόπο

ώστε να αποδοθεί μία «μετάφραση» στο σύστημα για την ένωση του φυσικού με του ψηφιακού κόσμου. Αυτή η διαδικασία πραγματοποιείται μέσω της χρήσης του προγράμματος CPRog και του γνωστού Microsoft Office Excel. Ο λόγος που θα χρησιμοποιήσουμε το παραπάνω πρόγραμμα, είναι επειδή μέσω εκείνου θα μπορούσαμε εύκολα να δημιουργήσουμε συναρτήσεις για τον υπολογισμό των αποστάσεων. Η συνάρτηση θα είναι της μορφής $y = ax \pm b$ όπου y θα είναι το αποτέλεσμα μας για την κίνηση σε κάθε άξονα και το x θα είναι η πληροφορία που θα μας δίνεται από την ίδια την κάμερα. Οπότε εδώ προκύπτουν τα εξής ζητήματα:

- Πρώτον, θα πρέπει να δημιουργήσουμε τρεις (3) συναρτήσεις, μία για κάθε άξονα έχοντας λάβει υπόψιν τις ακραίες τιμές που θα έχουμε για την αναγνώριση και λήψη του αντικειμένου, και
- Δεύτερον, θα πρέπει να βαθμονομήσουμε εξίσου και τις αριθμητικές πληροφορίες που θα μας δίνονται από την κάμερα ώστε να ταιριάζουν με το αποτέλεσμα που θα λαμβάνουμε από την συνάρτηση και το πρόγραμμα του ρομποτικού βραχίονα.

Η πρώτη διαδικασία θα πραγματοποιηθεί με την βοήθεια του Microsoft Office Excel παίρνοντας τις ακραίες τιμές ανά άξονα. Συγκεκριμένα, τους πρώτους τρεις κινητήρες (κινητήρες θέσεως) τους ρυθμίζουμε από τις αρθρώσεις A1, A2 και A3 του προγράμματος. Ο κινητήρας A1 ευθύνεται για την περιστροφή και οι κινητήρες A2 και A3 ευθύνονται ταυτόχρονα για την απομάκρυνση του από την βάση, δηλαδή την απόσταση καθώς και το ύψος.

Έτσι, φτιάχνουμε δύο συναρτήσεις οι οποίες θα χειρίζονται τις αρθρώσεις A2 και A3 του ρομποτικού βραχίονα με σκοπό την απομάκρυνση του βραχίονα από την βάση. Η κίνηση του ρομποτικού βραχίονα στον χώρο θα είναι σφαιρική δηλαδή, ο χώρος στον οποίο ο βραχίονας θα μπορεί να κινηθεί εάν αυτός είναι αυτός που φαίνεται στην παρακάτω εικόνα (Εικόνα 51):

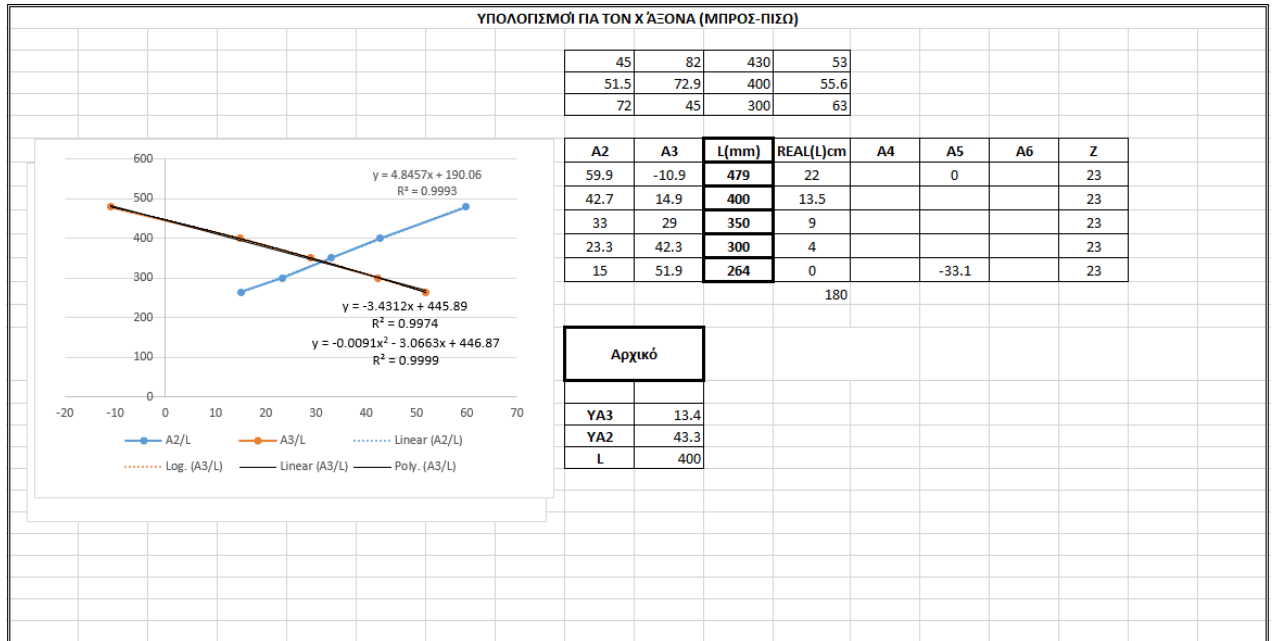


Εικόνα 51 - Χώρος εργασίας ανθρωπομορφικού βραχίονα

Στην περίπτωση μας όμως, εμείς θα χρησιμοποιήσουμε συγκεκριμένο χώρο, σε συγκεκριμένο ύψος (για όσα αντικείμενα είναι στο ύψος του τραπέζιού στο οποίο βρίσκεται ο ρομποτικός βραχίονας. Στην παρούσα εργασία δεν θα μεταβάλλεται το ύψος του ρομποτικού βραχίονα καθώς αυτό προϋποθέτει την ύπαρξη δεύτερης κάμερας κάθετη στην διεύθυνση της πρώτης και υπάρχουσας κάμερας. Άρα εμείς, θα έχουμε την επιλογή σε έναν χώρο ο οποίος θα είναι κυκλικός με ελάχιστη απόσταση αντικειμένου (ακτίνα) τα 264 mm (26.4 cm) και μέγιστη απόσταση αντικειμένου (ακτίνα) τα 479 mm (47.9 cm). Για να πάρουμε τις εξισώσεις κάθε άρθρωσης, θα πρέπει να σκεφτούμε ποιο είναι το επιθυμητό ύψος που θεωρούμε πως θα ικανοποιεί τον σκοπό μας. Προσωπικά, θεωρώ πως τα 23 mm είναι ικανά για την διαχείριση των αντικειμένων στον χώρο εργασίας μας. Έχοντας την μεταβλητή του ύψους ως σταθερή (άξονας Z του προγράμματος), έχοντας την μεταβλητή της στρέψης του βραχίονα σταθερή (άξονας Y του προγράμματος) μπορούμε να βγάλουμε τις εξισώσεις μοιρών των αρθρώσεων A2 και A3. Η διαδικασία είναι η παρακάτω:

- Μετακινούμε τον βραχίονα στην θέση $L=264$ mm με τέτοιο τρόπο ώστε το ύψος να έχει την τιμή 23 mm. Καταγράφουμε στο excel σε πίνακα τα στοιχεία που μας δίνονται από το Software του CPRog για τις δύο αρθρώσεις.
- Μετακινούμε τον βραχίονα στην θέση $L=300$ mm με ύψος 23 mm. Καταγράφουμε ξανά στον πίνακα τα στοιχεία των αρθρώσεων A2 και A3.
- Μετακινούμε τον βραχίονα στην θέση $L=350$ mm με ύψος 23 mm. Καταγράφουμε ξανά στον πίνακα τα στοιχεία των αρθρώσεων A2 και A3.
- Μετακινούμε τον βραχίονα στην θέση $L=400$ mm με ύψος 23 mm. Καταγράφουμε ξανά στον πίνακα τα στοιχεία των αρθρώσεων A2 και A3.
- Μετακινούμε τον βραχίονα στην θέση $L=479$ mm με ύψος 23 mm. Καταγράφουμε ξανά στον πίνακα τα στοιχεία των αρθρώσεων A2 και A3.

Αφού έχουμε καταγράψει αυτές τις μεταβλητές, είμαστε έτοιμοι να δημιουργήσουμε τις κατάλληλες εξισώσεις που μας δείχνουν την κίνηση των αρθρώσεων ανάλογα την απομάκρυνση τους από το ελάχιστο προς το μέγιστο σημείο επιλογής του αντικειμένου. Έτσι, καταλήγουμε στα παρακάτω στοιχεία (Εικόνα 52):



Εικόνα 52 - Υπολογισμοί για τον Χ άξονα (απομάκρυνσης)

Άρα, έχουμε τις παρακάτω εξισώσεις που μας προσδιορίζουν και την κίνηση της κάθε άρθρωσης που ευθύνεται για την απομάκρυνση. Συγκεκριμένα, για την άρθρωση A2 έχουμε την παρακάτω εξίσωση (1):

$$y(A2) = 4.8457x_2 + 190.06$$

Όπου $y(A2)$ η απομάκρυνση σε mm και x οι μοίρες της άρθρωσης στο CPRog. Εμείς όμως θέλουμε τις μοίρες και όχι την απομάκρυνση καθώς αυτή θα την γνωρίζουμε αργότερα από τον συσχετισμό των ψηφιακών τιμών `bbox[1]` και `bbox[2]` με την πραγματική απόσταση. Άρα, λύνουμε την παραπάνω εξίσωση ως προς x . Έτσι έχουμε την εξίσωση (2):

$$x_2 = \frac{(y(A2) - 190.06)}{4.8457}$$

Οπότε, με την εξίσωση (2) μπορούμε να βρούμε την αλλαγή της άρθρωσης A2 στην απόσταση του αντικειμένου. Ακριβώς την ίδια διαδικασία πρέπει να κάνουμε για την εύρεση της κίνησης της άρθρωσης A3. Η A3 άρθρωση ακολουθεί την εξίσωση (3) που φαίνεται παρακάτω:

$$y(A3) = -3.4312x_3 + 445.89$$

Όμως, ακριβώς και όπως στην περίπτωση της άρθρωσης A2, εμείς θέλουμε να βρούμε την εξίσωση μεταβολής της γωνίας της άρθρωσης A3. Έτσι, θα λύσουμε ως προς x και θα αντικαθιστούμε το $y(A3)$ με την ψηφιακή τιμή που θα μας δίνεται. Άρα έχουμε την εξίσωση (4):

$$x_3 = - \frac{(y(A3) - 445.89)}{3.4312}$$

Για την εύρεση της γωνίας που επιθυμούμε να στρέψει ο ρομποτικός βραχίονας προκειμένου να συναντήσει το αντικείμενο που μας ενδιαφέρει, θα πρέπει να το υπολογίσουμε τελείως διαφορετικά. Συγκεκριμένα, θα πρέπει να το προσεγγίσουμε γεωμετρικά και όχι εμπειρικά σύμφωνα με παρατήρηση τιμών και την καταγραφή τους. Η βασική ιδέα είναι η εξής:



Εικόνα 53 - Γεωμετρικός χώρος εργασίας

Στην Εικόνα 53 μπορούμε να διακρίνουμε τον γεωμετρικό χώρο εργασίας μας, δηλαδή, τα υπολογισμένα και προσεγγίσιμα σημεία στα οποία μπορεί να βρίσκεται ένα αντικείμενο το οποίο μας ενδιαφέρει να επιλέξουμε με τον ρομποτικό μας βραχίονα. Θυμίζουμε πως η ελάχιστη απόσταση σε μήκος για την επιλογή ενός αντικειμένου είναι τα 25,4 εκατοστά (cm) ή αλλιώς η εσωτερική ακτίνα MIN και ορίζουμε ως μέγιστη απόσταση επιλογής του αντικειμένου που μας ενδιαφέρει τα 47,9 εκατοστά (cm) ή αλλιώς η εξωτερική ακτίνα MAX. Ο χώρος που φαίνεται εάν τον φανταστούμε σαν έναν τετράγωνο χώρο που καλύπτει τα εσωτερικά ζωγραφισμένα στοιχεία, είναι ένας χώρος από τον οποίο θα παίρνουμε ψηφιακές τιμές, τις συντεταγμένες μας. Συγκεκριμένα, κάτω αριστερά θα είχαμε την τιμή $(bbox[0], bbox[1]) = (0,0)$ δηλαδή, οι ελάχιστες και πάνω δεξιά θα έχουμε $(bbox[0], bbox[1]) = (Max, Max)$. Οι τιμές αυτές που ορίζονται ως max ορίζονται μόνο κατόπιν ρύθμισης του επιθυμητού ύψους και κλίσης της κάμερας. Οπότε αυτά αποτελούν και τα στοιχεία τα οποία θα πρέπει να λάβουμε υπόψιν μας προκειμένου να κάνουμε μία βαθμονόμηση των τιμών του πραγματικού χώρου με αυτόν του ψηφιακού. Με άλλα λόγια, θα πρέπει κάπως να μεταφράσουμε στο σύστημα τις πραγματικές αποστάσεις με τις ψηφιακές μονάδες.

Πλέον, το μόνο που μας μένει είναι να προσδιορίσουμε την τιμή της γωνίας μας με την χρήση της τριγωνομετρίας. Για αυτόν τον λόγο, θα χρησιμοποιήσουμε ένα σχετικό παράδειγμα για να δούμε τον μηχανισμό πως δουλεύει:



Εικόνα 54 - Παράδειγμα στον γεωμετρικό χώρο εργασίας

Όπως μπορούμε να διακρίνουμε στην Εικόνα 54, εάν επιλέξουμε μία τιμή η οποία μπορεί να έχει μήκος οποιοδήποτε μεταξύ $MIN \leq C \leq MAX$ τότε θα μπορεί να επιλεγεί από τον ρομποτικό μας βραχίονα. Συγκεκριμένα, για το σημείο C με συντεταγμένες τις τιμές $bbox[0]$ και $bbox[1]$ που για συντομία στο παράδειγμά μας θα τις ονομάσουμε (x, y) αντίστοιχα, παρατηρούμε πως σχηματίζεται ένα τρίγωνο πάνω στο οποίο μπορούμε να βρούμε τα στοιχεία που μας απασχολούν. Συγκεκριμένα, εμάς μας ενδιαφέρει η **απόσταση c** και η **γωνία B**. Υποθέτουμε πως έχουμε βαθμονομημένα τα στοιχεία που αναφέραμε παραπάνω, δηλαδή την απόσταση του πραγματικού χώρου με αυτές των ψηφιακών μας τιμών. Έχοντας κάνει αυτή τη βαθμονόμηση, το λογισμικό θα μπορεί να αναγνωρίσει το μήκος της απόστασης C. Έτσι, με το πυθαγόρειο θεώρημα, έχουμε:

$$c^2 = a^2 + b^2$$

Από την συγκεκριμένη εξίσωση βρίσκουμε την απόσταση c, δηλαδή την απόσταση από το κέντρο του ρομποτικού βραχίονα. Αυτή θα χρησιμοποιήσουμε για την προέκταση του ρομποτικού βραχίονα όπως δείξαμε στην παραπάνω παράγραφο. Ύστερα, μας ενδιαφέρει να μάθουμε την γωνία B. Συγκεκριμένα, σύμφωνα με το παραπάνω σχήμα, με ενδιαφέρει να βρεθεί η γωνία CB(MAX). Η γωνία όμως αυτή είναι ίση με την γωνία A του τριγώνου ABC. Αν ορίσουμε την ευθεία CC και την ευθεία MIN MAX, πρόκειται για δύο ευθείες οι οποίες είναι παράλληλες. Τις ευθείες αυτές τις ενώνει μία άλλη ευθεία, η AB. Άρα, η γωνία A είναι ίση με την γωνία B (την εξωτερική του τριγώνου) ως εντός – εντός εναλλάξ. Αυτή η γωνία είναι η γωνία ενδιαφέροντος. Σύμφωνα με το πρόγραμμα CPRog, το αριστερό τεταρτημόριο (Τεταρτημόριο II) δέχεται τιμές από $90 - 0^\circ$ ενώ για το δεξί τεταρτημόριο (Τεταρτημόριο I) παίρνει τιμές από $0 - (-90)^\circ$. Άρα γνωρίζοντας τις πλευρές του τριγώνου, μπορούμε να βρούμε την γωνία από τον παρακάτω τύπο:

$$\sin^{-1}\left(\frac{a}{c}\right)$$

Η γωνία αυτή μας είναι ιδιαίτερα χρήσιμη καθώς αντιπροσωπεύει ορθόδοξα την γωνία Β που φαίνεται στην εικόνα (CB(MAX)). Έτσι, η τιμή της άρθρωσης A1 ισούται με την παραπάνω εξίσωση.

Έχοντας στην κατοχή μας την γωνία περιστροφής του ρομποτικού βραχίονα (παράμετρος A1) αλλά και την απόσταση που υπάρχει μεταξύ του αντικειμένου και του συστήματος αναφοράς του ρομποτικού βραχίονα (απόσταση C, συνδυασμός κίνησης των αρθρώσεων A2 και A3), μπορούμε να μεταβούμε στην επόμενη διαδικασία. Πλέον, συνεχίζουμε στην βαθμονόμηση του πραγματικού κόσμου με αυτόν του ψηφιακού. Οι τιμές που δεχόμαστε από τα στοιχεία bbox[0] και bbox[1] είναι ψηφιακές τιμές που αντιπροσωπεύουν τον χώρο που βλέπει η κάμερα. Μπορούμε να αντιληφθούμε όμως πως αυτές οι τιμές δεν είναι αντιπροσωπευτικές με το πραγματικό μήκος που έχουμε. Για παράδειγμα, μπορούμε να έχουμε σαν τιμή στην μεταβλητή bbox[0] = 40 . Αυτή δεν σημαίνει πως είναι 40 mm/cm/m μακριά από το σύστημα αναφοράς του ρομποτικού βραχίονα αλλά ούτε και 40° γωνία από την ευθεία του ρομποτικού βραχίονα. Έτσι, απευθείας έρχεται η ανάγκη της βαθμονόμησης αυτών των μονάδων για την καλύτερη διευκόλυνση μας. Όπως θα παρατηρήσετε στην εικόνα 52, οι τιμές που έχουν ληφθεί για την δημιουργία μίας εξίσωσης κίνησης των αρθρώσεων είναι βασισμένες στην πραγματική απόσταση της κλίμακας των χιλιοστών (mm). Επίσης, θα πρέπει να λάβουμε υπόψιν και τον περιορισμό κινήσεως τόσο στο ελάχιστο μήκος και κάτω όσο και το μέγιστο και άνω. Δηλαδή, ένα αντικείμενο δεν θα μπορεί να επιλεγθεί από τον ρομποτικό βραχίονα για μήκη κάτω των 264(mm) και άνω των 479(mm). Με την κατάλληλη διαμόρφωση της εξίσωσης, μπορούμε να μικρύνουμε την ελάχιστη απόσταση κατά πολύ αλλά αυτό δεν ήταν θεμιτό στην συγκεκριμένη διπλωματική άσκηση καθώς δεν προγραμματίζουμε τις αρθρώσεις A4,A5 και A6.

Η βαθμονόμηση των ψηφιακών τιμών με τον πραγματικό χώρο θα γίνει πρακτικά με την βοήθεια της κάμερας, του υπολογιστή (χρήση επεξεργαστικού προγράμματος excel) και την βοήθεια ενός αντικειμένου προτύπου (αποφεύγουμε την αλλαγή διαστάσεων των αντικειμένων σε περιπτώσεις βαθμονόμησης για την δημιουργία ενός προτύπου γιατί έχουμε μεγάλη διασπορά των στοιχείων μας). Οπότε, καταγράφουμε την θέση του αντικειμένου σύμφωνα με τις συντεταγμένες που μας δίνονται από την κάμερα στον υπολογιστή μας. Η καταγραφή θα φαίνεται στον παρακάτω πίνακα έπειτα από τις παρακάτω εικόνες που θα δείχνουν την διαδικασία:

Δημιουργούμε έναν απλοϊκό πίνακα με τις αποστάσεις που έχουμε μετρήσει τόσο σαν εκατοστά στον χώρο εργασίας μας αλλά τόσο και στις τιμές της κάμερας που έχουμε παρατηρήσει. Για τις τιμές της κάμερας, σύμφωνα με το λογισμικό του προγράμματος της Phycharm, οι ελάχιστες τιμές βρίσκονται στην κάτω αριστερά γωνία και οι μέγιστες τιμές της φαίνονται στην πάνω δεξιά γωνία. Έτσι, σημειώνουμε τις αντίστοιχες τιμές μας για να έχουμε έναν μετατροπέα της ψηφιακής με την πραγματική απόσταση:

	bbox[0]	bbox[1]	Length (mm)
MIN	0	0	0
MAX	$bbox[0]_{MAX}$	$bbox[1]_{MAX}$	479

Ο λόγος όπου δεν έχουμε τις μέγιστες τιμές μας σαν κάτι σταθερό, είναι επειδή η τοποθέτηση της κάμερας έχει δυνατότητα μετακίνησης. Για παράδειγμα, μπορούμε να αποδώσουμε μεγαλύτερο ύψος στην κάμερα ή να την μετακινήσουμε πίσω και μπροστά. Ακόμη, μπορούμε να αλλάξουμε και τη γωνία τοποθέτησης της κάμερας. Όλες αυτές οι μεταβολές, ακόμη και οι πιο μικρές, επηρεάζουν άμεσα τις μετρήσεις των μέγιστων σημείων. Αυτός είναι και ο λόγος όπου στο πρόγραμμα έχει προγραμματιστεί ερώτηση προς τον χρήστη για να ορίσει τις μέγιστες τιμές του. Ας θυμηθούμε όμως τι βλέπει η κάμερα μας με την παρακάτω εικόνα.



Εικόνα 55

Στην παρούσα φάση, μας ενδιαφέρει να βρούμε την αντιστοίχιση των ψηφιακών τιμών με αυτές του χώρου εργασίας για να υπολογίσουμε τις αποστάσεις των a και b για να βρούμε την απόσταση c που μας ενδιαφέρει. Η αντιστοίχιση προκύπτει από την απλή μέθοδο των τριών και έχουμε:

- Στη περίπτωση του $bbox[0]$: Για MAX αντιστοιχεί στα 479(mm) , στα $bbox[0]$ πόσα (mm) αντιστοιχούν;

$$b = \frac{bbox[0] * 479 (mm)}{MAX}$$

- Στην περίπτωση του $bbox[1]$: Για MAX αντιστοιχεί στα 479 (mm), στα $bbox[1]$ πόσα (mm) αντιστοιχούν;

$$a = \frac{bbox[1] * 479 (mm)}{MAX}$$

Πλέον, έχοντας στην κατοχή μας τις τιμές αυτές, μπορούμε να υπολογίσουμε μέσω του Πυθαγορείου Θεωρήματος το αποτέλεσμα της πραγματικής απόστασης c αλλά ταυτόχρονα σύμφωνα με τις παραπάνω συναρτήσεις που βγάλαμε, βρίσκουμε και την στρέψη του ρομποτικού βραχίονα σε μοίρες. Καθώς όμως βρισκόμαστε σε ένα κομβικό σημείο της λήξης των μετρήσεων, θα πρέπει να παρατηρήσουμε ότι ο γεωμετρικός μας χώρος, πέραν της μέγιστης του απόστασης έχει και ελάχιστη απόσταση η οποία δεν είναι το μηδέν. Για αυτόν τον λόγο, θα πρέπει να δημιουργηθεί μία συνθήκη AN η οποία αφού έχει υπολογιστεί το c θα πρέπει να το συγκρίνει εάν είναι εντός των ορίων του. Στην περίπτωση που είναι, η διαδικασία του ρομποτικού βραχίονα θα λειτουργήσει φυσιολογικά με σκοπό την επιλογή του αντικειμένου από τον χώρο εργασίας. Αντιθέτως, εάν η διαδρομή είναι πάνω από την μέγιστη ή μικρότερη από την ελάχιστη, θα πρέπει να αναφέρεται σχετικό μήνυμα στον χρήστη από το πρόγραμμα για να ειδοποιείται κατάλληλα.

2. Προγραμματισμός και δημιουργία script

Για την εκτέλεση της λειτουργίας που επιθυμούμε να δημιουργήσουμε, θα χρειαστεί να δημιουργήσουμε ένα δικό μας σενάριο (script) μέσω ειδικού προγραμματιστικού περιβάλλοντος. Ο χειρισμός του ρομποτικού βραχίονα είναι αρκετά εύκολος με την χρήση του προγραμματιστικού του περιβάλλοντος CPRog και μπορούμε εύκολα να δημιουργήσουμε απλές, επαναλαμβανόμενες και ειδικές κινήσεις. Στις απλές κινήσεις περιλαμβάνουμε κινήσεις όπως η μετακίνηση από ένα σημείο Α σε ένα σημείο Β στον χώρο, είτε γραμμικά είτε με επιθυμητές αλλαγές στις γωνίες των κινητήρων του. Στις επαναλαμβανόμενες κινήσεις θα περιλάβουμε κινήσεις όπου θα υπάρχει συνέχεια εις βάθος χρόνου. Ένα απλό παράδειγμα θα ήταν ο προγραμματισμός κινήσεως του βραχίονα να μεταβαίνει στο σημείο Α και να παραλαμβάνει ένα αντικείμενο με την δαγκάνα (gripper) του από τον διάδρομο παραγωγής για την τοποθέτηση του σε ένα σημείο Β που βρίσκεται ο διάδρομος ελέγχου του αντικειμένου. Στις ειδικές κινήσεις συγκαταλέγονται περιπτώσεις με προγραμματιστικές προϋποθέσεις και εντολές. Αυτές είναι οι διαδικασίες αν...τότε (if), όσο... (while) και άλλα ακόμα περισσότερα. Στην περίπτωση όμως που επιθυμούμε να εκτελέσουμε μία ιδέα την οποία το προγραμματιστικό περιβάλλον του ρομποτικού βραχίονα δεν δύναται να τη πραγματοποιήσει, τότε καλούμαστε να την δημιουργήσουμε με έμμεσους τρόπους μέσω του προγραμματισμού. Στην συγκεκριμένη διπλωματική εργασία, ο προγραμματισμός θα υλοποιηθεί με την γλώσσα προγραμματισμού **Python** και την χρήση του δωρεάν προγράμματος **PyCharm**. Η επιλογή της συγκεκριμένης γλώσσας έγινε με κριτήρια της ευκολίας της ως προς την εκμάθηση και την χρήση της αλλά και την ικανότητα χρήσης ειδικών βιβλιοθηκών που βοηθάνε στην υλοποίηση της ιδέας μας.

1. Προγραμματιστικό περιβάλλον PyCharm και ασφαλής εγκατάσταση

Όπως αναφέραμε παραπάνω, στη συγκεκριμένη διπλωματική θα χρησιμοποιηθεί η python σαν γλώσσα προγραμματισμού προκειμένου να υλοποιήσουμε τις απαιτήσεις μας.



Εικόνα 56 - Python logo

Η python πρόκειται για μία διερμηνευόμενη, γενικού σκοπού και υψηλού επιπέδου, γλώσσα προγραμματισμού. Αναλυτικά, αυτά αναφέρονται στο ότι η γλώσσα αυτή διερμηνεύεται αυτόματα από τις εντολές που γράφουμε στη γλώσσα μηχανής, η χρήση της

είναι γενική και δεν πρόκειται για μία ειδικά σχεδιασμένη γλώσσα για πολύ ειδικούς σκοπούς και τέλος, πρόκειται για μία γλώσσα η οποία επιτρέπει την μεταφορά της σε κάθε άλλο υπολογιστή. Η python πρόκειται για μία προγραμματιστική γλώσσα η οποία έχει ως σκοπό την δημιουργία ενός μικρότερου και πιο συμπυκνωμένου κώδικα όπου ταυτόχρονα θα είναι το ίδιο λειτουργικός με άλλες προγραμματιστικές γλώσσες όπως αυτή της C++ ή της Java. Ένα εξαιρετικά θετικό προσόν της python είναι η λειτουργία της με τις βιβλιοθήκες της καθώς αυτές επιτρέπουν εύκολα λειτουργίες που επιθυμεί ο χρήστης. Ταυτόχρονα, ένα μειονέκτημα της είναι η ταχύτητά της λόγω του διερμηνέα. Καθώς άλλες γλώσσες μεταφράζονται απευθείας σε γλώσσα μηχανής, η συγκεκριμένη θα πρέπει να διερμηνευτεί με έναν τρόπο ώστε να μεταφραστεί σε γλώσσα μηχανής. Η διαδικασία αυτή παίρνει παραπάνω χρόνο καθώς η μετάφραση γίνεται έμμεσα και όχι άμεσα.

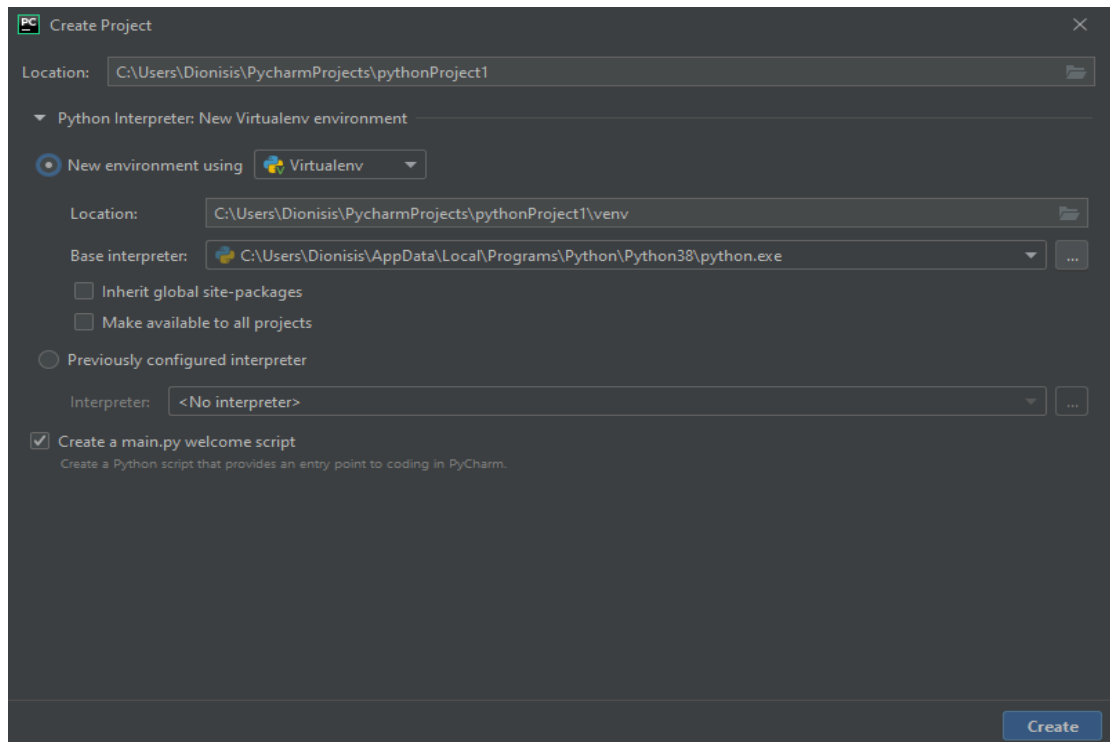
Η python μπορεί να διατυπωθεί ακόμη και σε ένα απλό αρχείο notepad. Ο λόγος όμως που προτιμάμε να την διατυπώσουμε σε ειδικό πρόγραμμα, είναι οι ευκολίες που πρόκειται να μας διαθέσει κατά την διάρκεια της δημιουργίας, ελέγχου αλλά και δοκιμής του κώδικα. Επίσης, ένας σημαντικός λόγος της χρήσης προγράμματος είναι για την εύκολη διαχείριση των βιβλιοθηκών που μας παρέχει η Python. Έτσι, θα χρησιμοποιήσουμε το πρόγραμμα “PyCharm”. Πρώτα θα πρέπει να κατεβάσουμε την γλώσσα προγραμματισμού μας για να έχει ο υπολογιστής την απαραίτητη βάση. Συγκεκριμένα, θα πρέπει να μεταβούμε στην επίσημη ιστοσελίδα της Python (www.python.org/downloads/) και να κατεβάσουμε την τελευταία έκδοση της Python που διατίθεται άνευ πληρωμής για το λειτουργικό του υπολογιστή μας. Εκτελούμε το αρχείο που κατέβηκε στον υπολογιστή μας και κάνουμε τα απαραίτητα βήματα που απαιτούνται για την σωστή εγκατάστασή του (οι οδηγίες θα δίνονται από τον οδηγό εγκατάστασης του ίδιου του προγράμματος). Έπειτα, θα πρέπει να κατεβάσουμε την πλατφόρμα με την οποία θα προγραμματίζουμε. Αυτή η πλατφόρμα, θα είναι το PyCharm. Το PyCharm το κατεβάζουμε στον υπολογιστή μας μέσω της ηλεκτρονικής σελίδας “JetBrains” στην ιστοσελίδα (www.jetbrains.com/pycharm/) και βρίσκουμε το απαραίτητο αρχείο για το λειτουργικό του υπολογιστή μας. Αφού το κατεβάσουμε, εκτελούμε το αρχείο και κάνουμε ότι είναι απαραίτητο για να το προετοιμάσουμε για χρήση.

Εφόσον η διαδικασία προχωρήσει και εκτελεστεί όπως πρέπει, εκτελούμε το πρόγραμμα για να δούμε το περιβάλλον του. Αρχικά, το πρόγραμμα θα μας δώσει κάποιες επιλογές από τις οποίες θα επιλέξουμε την “**Create New Project**” (Εικόνα 50).



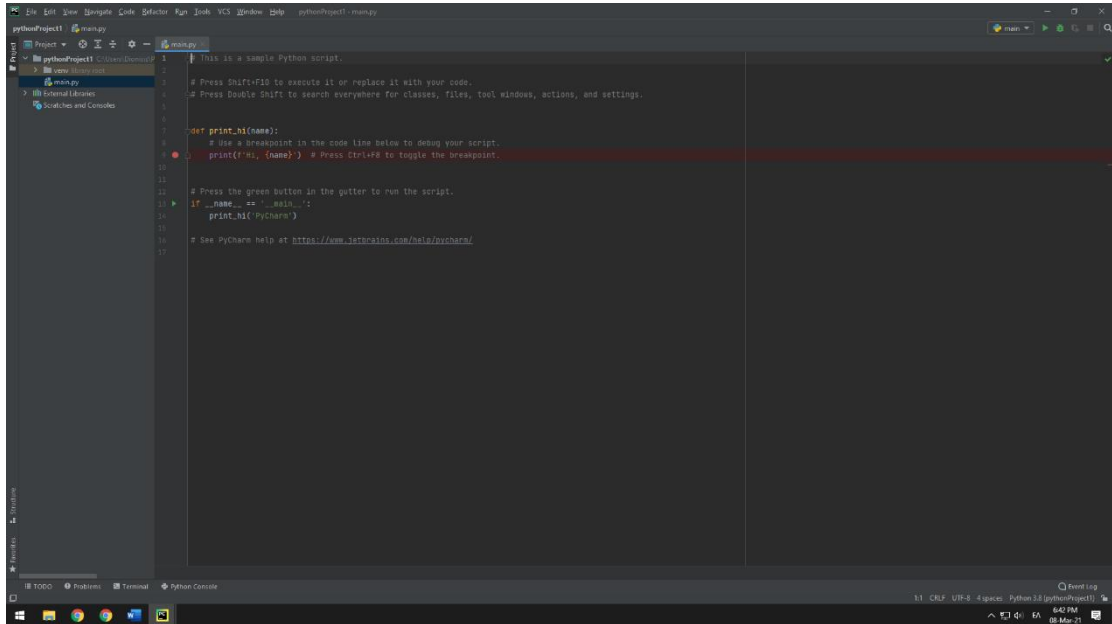
Εικόνα 57 - Πρώτη επαφή με το PyCharm

Έπειτα, το πρόγραμμα θα μας δώσει την επιλογή να δώσουμε κάποιες πληροφορίες για το project αυτό όπως την ονομασία του, τον μεταφραστή κ.α. . Περισσότερες πληροφορίες δίνονται στην παρακάτω εικόνα (Εικόνα 51):



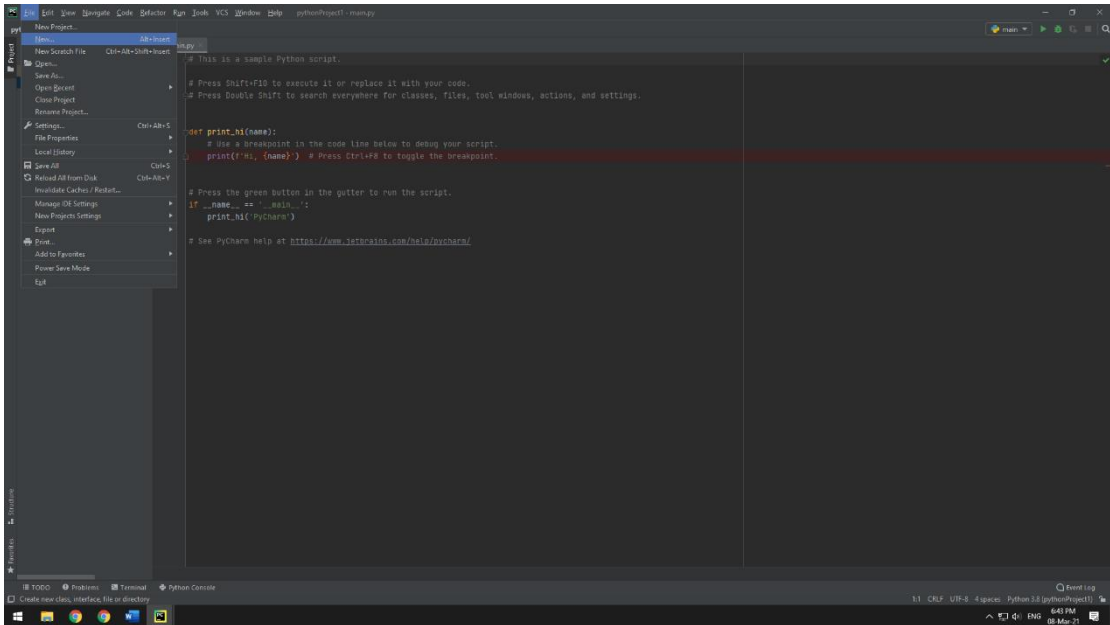
Εικόνα 58 - Δημιουργώντας νέο project στο PyCharm

Αφού κάνουμε τα απαραίτητα που φαίνονται στις παραπάνω εικόνες, το πρόγραμμα θα μας εμφανίσει το περιβάλλον προγραμματισμού του (Εικόνα 52). Εκεί, θα μπορέσουμε να δημιουργήσουμε τον θεμιτό μας κώδικα ο οποίος θα μας επιτρέψει την επίτευξη του στόχου μας για την συγκεκριμένη διπλωματική εργασία. Τυπικά, όπως θα φαίνεται στην παρακάτω εικόνα (Εικόνα 52), θα εμφανίζεται αυτόματα ένα αρχείο ως “main.py” το οποίο πρόκειται για ένα προγραμματιστικό αρχείο python. Εκεί, μπορούμε να δημιουργήσουμε τον κώδικα χρησιμοποιώντας τις κατάλληλες εντολές.

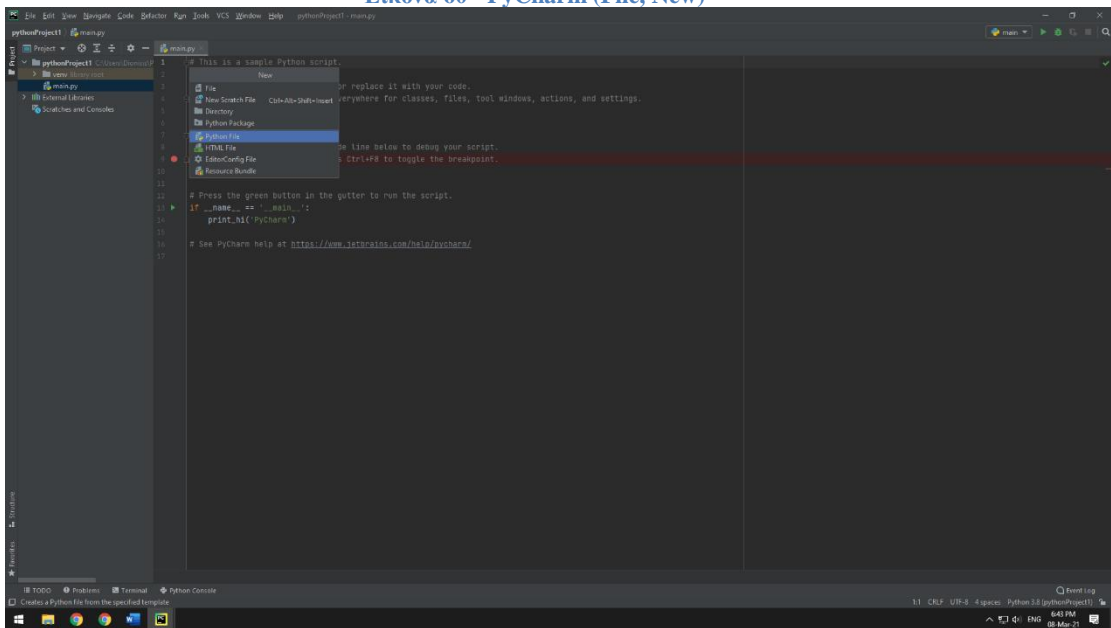


Εικόνα 59 - Προγραμματιστικό περιβάλλον του PyCharm

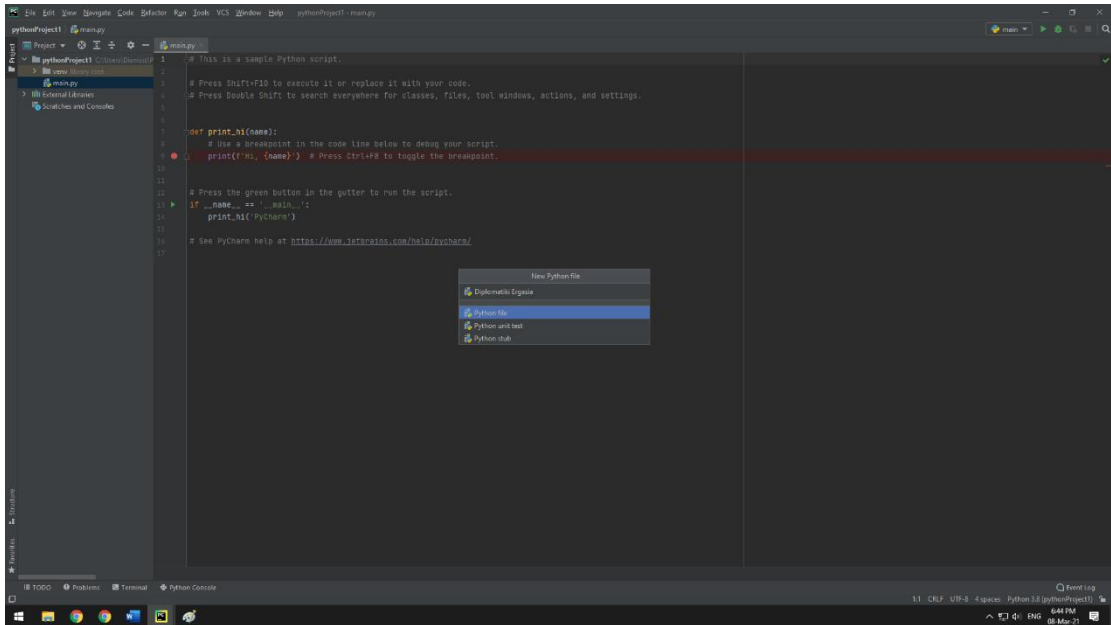
Μπορούμε να διαγράψουμε το περιεχόμενο του συγκεκριμένου αρχείου και να δημιουργήσουμε τον κώδικά μας σε αυτό το πεδίο. Όμως, εμείς θα δημιουργήσουμε ένα δικό μας αρχείο εκτελώντας τις παρακάτω ενέργειες (Εικόνα 53 – Εικόνα 56):



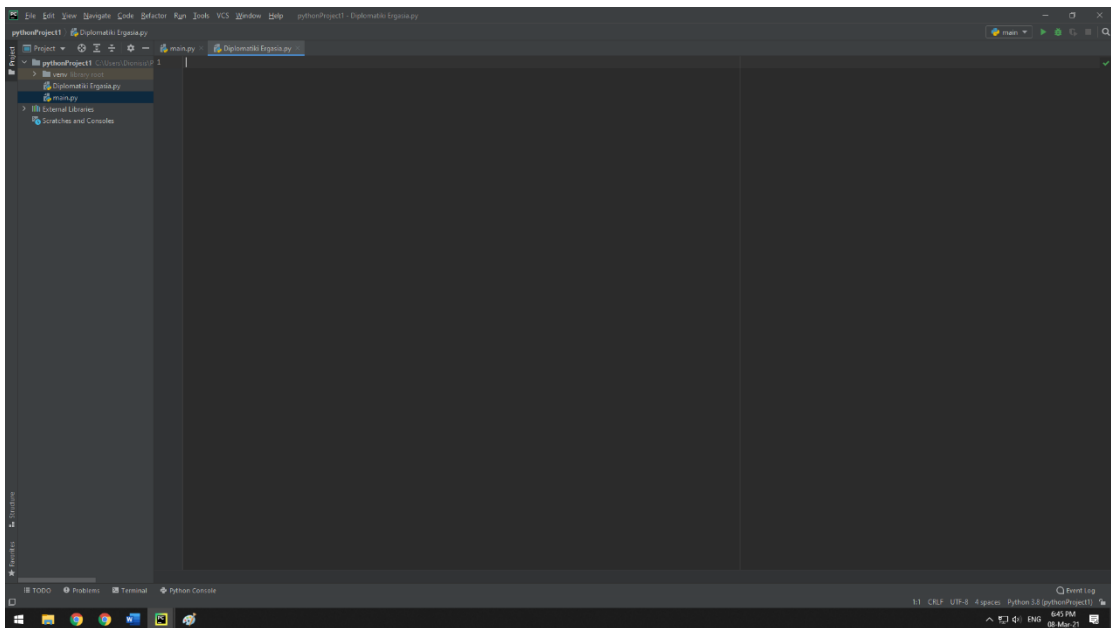
Εικόνα 60 - PyCharm (File, New)



Εικόνα 61 - PyCharm (File, New, Python File)

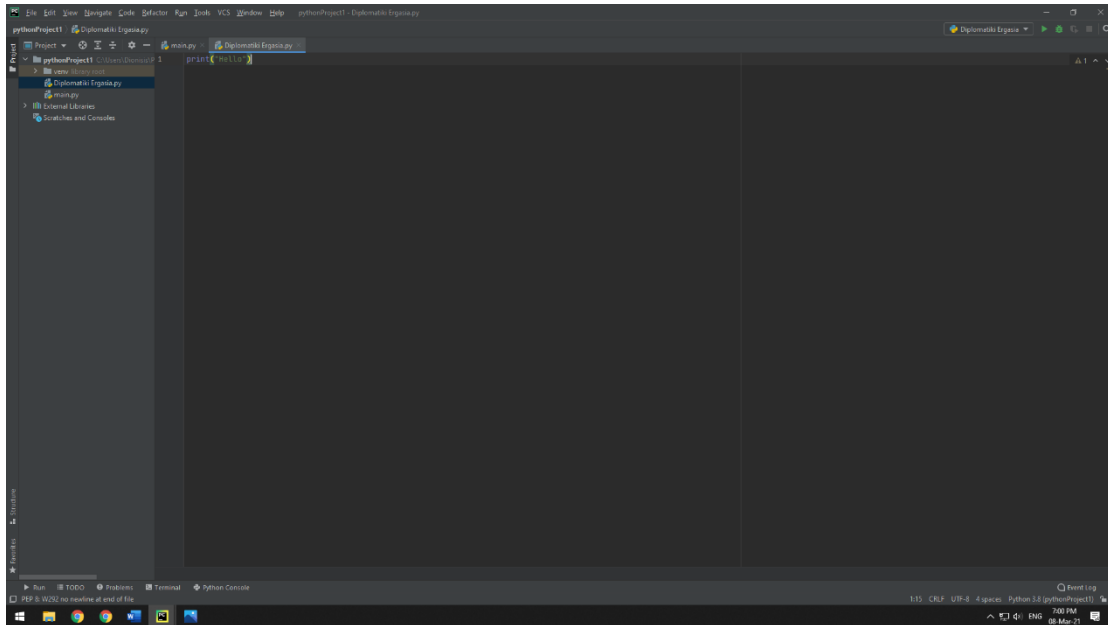


Εικόνα 62 - PyCharm (File, New, Python File, Name)

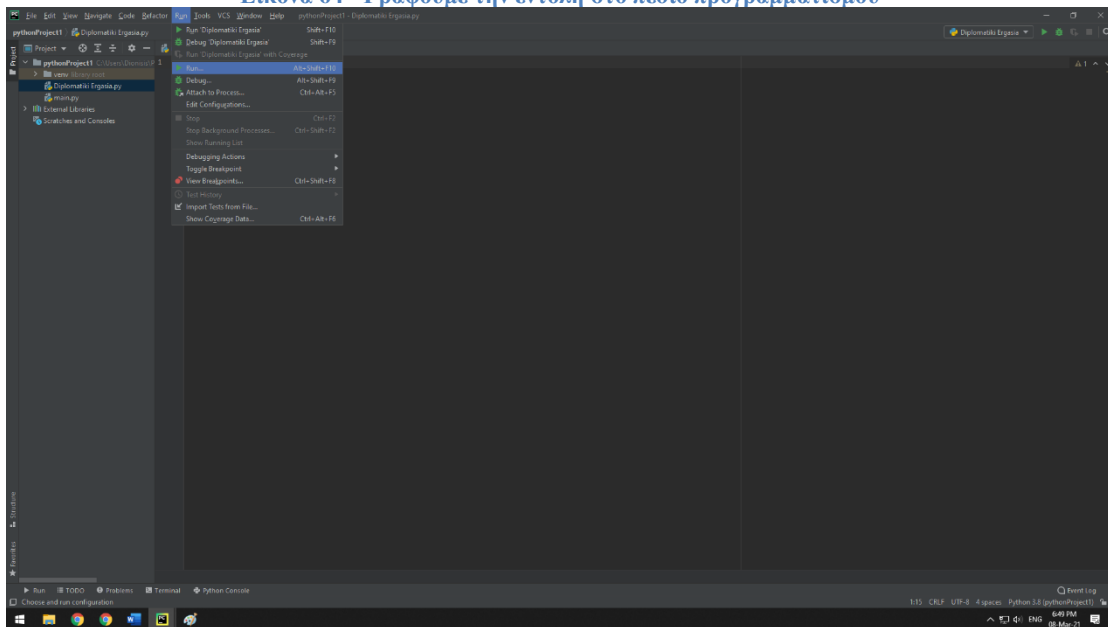


Εικόνα 63 - PyCharm (New File Created!)

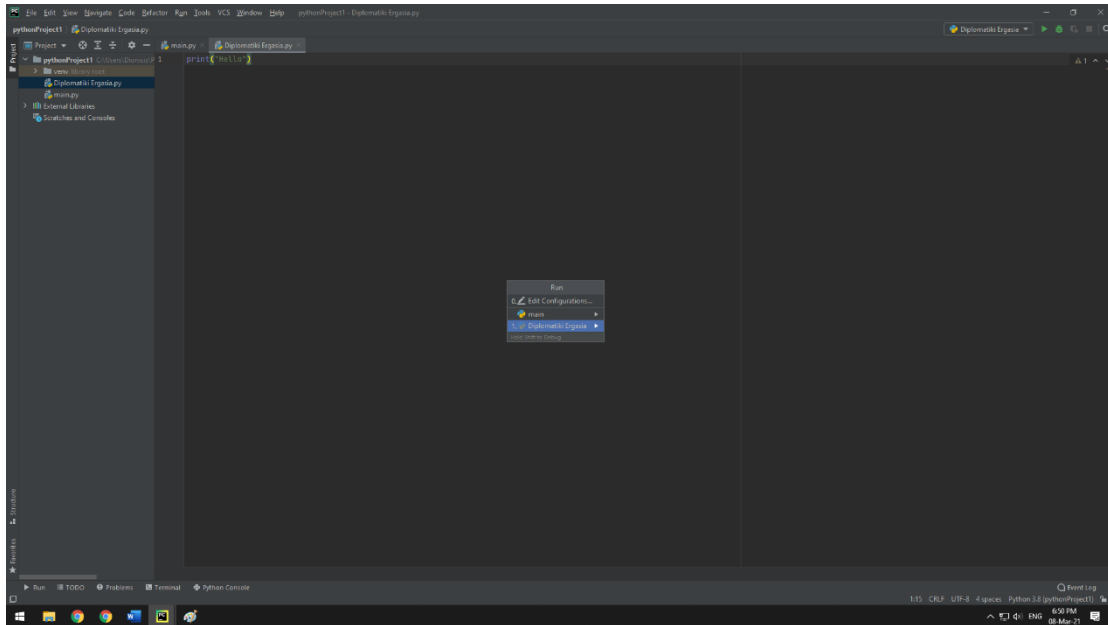
Πλέον έχουμε δημιουργήσει το νέο αρχείο στο οποίο θα μπορούμε να δημιουργήσουμε και να εκτελέσουμε τον κώδικα μας. Στην συγκεκριμένη περίπτωση, το αρχείο αυτό ονομάστηκε “Diplomatiki Ergasia” για ενδεικτικούς και εκπαιδευτικούς σκοπούς. Ενδεικτικά, μπορούμε να εκτελέσουμε ένα παράδειγμα το οποίο θα φανεί στις παρακάτω εικόνες. Το παράδειγμα αυτό βασίζεται στην εκτέλεση εντολής “print” του κώδικα της Python στο πρόγραμμα. Με απλούστερα λόγια, η εντολή αυτή μας «εμφανίζει» το εσωτερικό αποτέλεσμα της. Για το συγκεκριμένο παράδειγμα, θα εκτελέσουμε την εντολή [print(“Hello”)]. Τον τρόπο εκτέλεσης και το αποτέλεσμα θα τα δείτε στις παρακάτω εικόνες (Εικόνα 57 – Εικόνα 60).



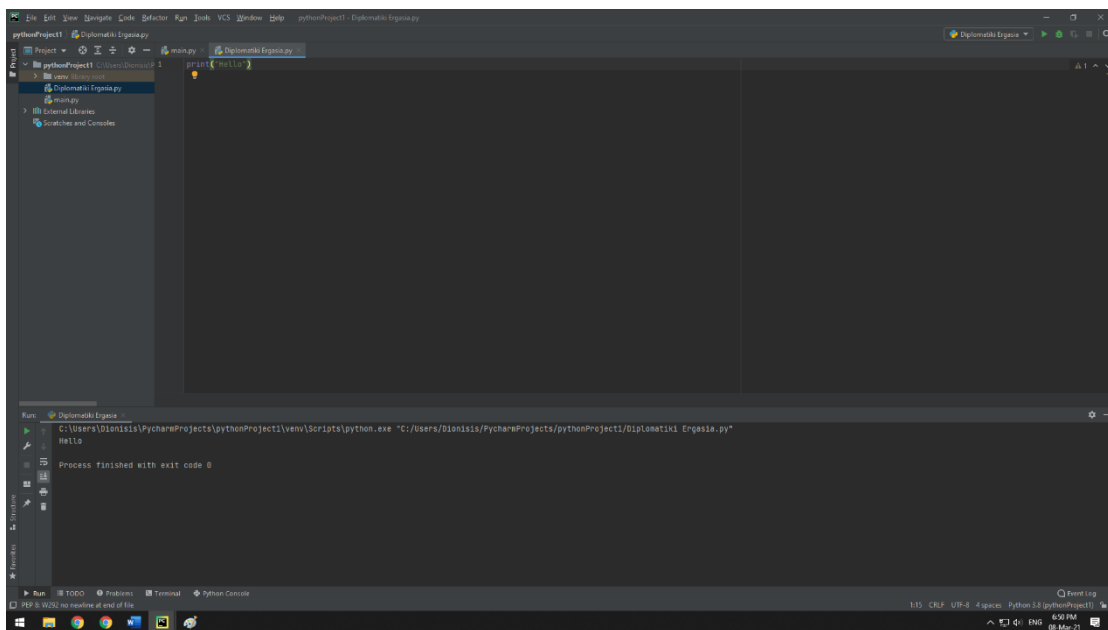
Εικόνα 64 - Γράφουμε την εντολή στο πεδίο προγραμματισμού



Εικόνα 65 - Εκτελούμε την πρώτη φορά με Run - Run



Εικόνα 66 - Επιλέγουμε να εκτελεστεί η εντολή μέσα στο αρχείο όπου γράψαμε τον κώδικα (Diplomatiki Ergasia στην συγκεκριμένη περίπτωση)



Εικόνα 67 - Το αποτέλεσμά μας έχει εμφανιστεί (εκτυπωθεί) στο κάτω μέρος του προγράμματος μας

Τα βήματα αυτά καλύπτουν τις βασικές ενέργειες που θα πρέπει να πραγματοποιηθούν ώστε να ξεκινήσουμε το παραπάνω πρόγραμμα και να δημιουργήσουμε τον κώδικα μας. Η Python είναι μία γλώσσα προγραμματισμού η οποία θεωρώ πως είναι πολύ πιο προσιτή στην εκμάθησή της σε ένα άτομο που έχει μέτρια γνώση στην δημιουργία κώδικα σε MATLAB. Οι εντολές αλλά και το περιεχόμενο της γλώσσας προγραμματισμού της Python, δεν απέχει σημαντικά πολύ από αυτή του MATLAB. Αφού πραγματοποιηθούν τα παραπάνω βήματα, είμαστε έτοιμοι να προχωρήσουμε στην δημιουργία του επιθυμητού μας κώδικα.

2. Κώδικας της παρούσας διπλωματικής εργασίας

Στην παρούσα ενότητα θα παρουσιαστεί ο κώδικας ο οποίος θα τρέχει για την διαδικασία που θέλουμε να εκτελέσουμε. Συγκεκριμένα, ο σκοπός μας είναι να δημιουργήσουμε την ικανότητα στον χρήστη / χειριστή να μπορεί να επιλέγει το αντικείμενο του ενδιαφέροντος του προκειμένου να το επιλέγει ο ρομποτικός βραχίονας και να το μεταφέρει σε ασφαλή σημείο. Όπως έχουμε αναφέρει ήδη σε προηγούμενη ενότητα, η προγραμματιστική γλώσσα της Python είναι αρκετά ευέλικτη σε αυτό το σημείο καθώς με την χρήση των κατάλληλων εντολών από τη βιβλιοθήκη της, μπορούμε να ενσωματώσουμε τις ιδέες μας και να τις κάνουμε πράξη. Για παράδειγμα, η χρήση της web camera μέσω του κώδικα της Python πραγματοποιείται με την κλήση της εντολής “`import cv2`” από τη βιβλιοθήκη. Με την εντολή αυτή, ο κώδικας καταλαβαίνει πως θα πρέπει να λάβει στοιχεία από την κάμερα του υπολογιστή όπου είναι συνδεδεμένη και σε αυτόν. Έπειτα, ορίζουμε την διαδικασία αυτή με την εξής εντολή “`cap = cv2.VideoCapture(0)`” . Μετά, δημιουργούμε την διαδικασία του εντοπισμού / tracking του αντικειμένου. Υπάρχουν δύο εντολές με τις οποίες μπορούμε να πραγματοποιήσουμε την ιδέα μας οι οποίες διαφέρουν μεταξύ τους σε κάποια χαρακτηριστικά. Πρώτα έχουμε την “`tracker = cv2.legacy_TrackerMOSSE.create()`” και έπειτα έχουμε την “`tracker = cv2.TrackerCSRT_create()`” . Η διαφορά μεταξύ αυτών των δύο εντολών είναι στα χαρακτηριστικά της απόκρισης τους αλλά και της αποτελεσματικότητάς τους. Για παράδειγμα, η πρώτη εντολή (η οποία χρησιμοποιείται στον κώδικά μας) παρέχει πολλά Frames Per Second (FPS) δίνοντάς μας έτσι την δυνατότητα να μην μπερδεύεται το πρόγραμμα κατά την επιλογή του αντικειμένου. Όμως, χάνουμε στην αποτελεσματικότητα της διατήρησης του Region Of Interest (ROI) καθώς εάν μετακινήσουμε ανορθόδοξα το αντικείμενό μας, χάνεται το ROI από την οθόνη μας και χάνει ο κώδικας το αντικείμενο. Αντιθέτως, η δεύτερη εντολή υστερεί στα FPS καθώς είναι κατά δύο φορές λιγότερα από την πρώτη αλλά λόγω αυτού, κρατάει καλύτερα το αντικείμενο σε ανίχνευση. Ο λόγος που επέλεξα την πρώτη εντολή αντί για την δεύτερη, είναι πως τα αντικείμενα που θα παρουσιάσουμε στην διπλωματική αυτή είναι στατικά και δεν κινούνται καθόλου.

Ακόμη, στην εισαγωγή του προγράμματος έχουν δημιουργηθεί κατάλληλα μηνύματα για την καλύτερη επικοινωνία και εκμάθηση του προγράμματος στον χρήστη. Στην αρχή, ο κώδικας εκπαιδεύει τον χρήστη τόσο στο τι κάνει ο κώδικας όσο και στον τρόπο λειτουργίας του.

```
import cv2

print('-'*40)
info=input('You will now use the tracking mode. Make a square in
your screen that covers the object of your\n'
           'interest. If the tracking process is lost, or if you
have selected the object and you are sure,\n'
           'please press the button -q- to continue. Once the
camera is open, select the ROI and press ENTER \n'
           'or SPACE. To start the process, type -ok-: ')
print('-'*40)

#-----
```

Table 1 Εισαγωγή του κώδικα

Αργότερα από αυτό το στάδιο, ο κώδικας αρχίζει και «χτίζεται» με τις απαραίτητες πληροφορίες για την εύρυθμη λειτουργία του πάνω στις ανάγκες μας. Εδώ, ορίζονται οι εντολές του εντοπισμού / tracking και διάφορων άλλων χαρακτηριστικών τους:

```
#-----
cap = cv2.VideoCapture(0)

tracker = cv2.legacy_TrackerMOSSE.create()
# tracker = cv2.TrackerCSRT_create()
success, img = cap.read()
bbox = cv2.selectROI("Tracking", img, False)
tracker.init(img, bbox)

def drawBox(img, bbox):
    x, y, w, h =int(bbox[0]),int(bbox[1]),int(bbox[2]),int(bbox[3])
    cv2.rectangle(img, (x,y), ((x+w), (y+h)), (0,0,255), 3, 1)
    cv2.putText(img, "Tracking", (75, 75),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

#-----
```

Table 2 - Ορισμός εντολών εντοπισμού αντικειμένου

Οι τιμές που θα φαίνονται καθώς τρέχουμε τον κώδικα θα είναι οι `bbox[0]` & `bbox[1]`. Αυτές οι τιμές θα έχουν ως ελάχιστη τιμή το 0 και μέγιστη μία καθορισμένη τιμή από τον χρήστη ή μέσω μίας διαδικασίας βαθμονόμησης η οποία ακόμα χτίζεται και είναι στα αρχικά της βήματα. Ανάλογα με το Field Of View (FOV), η κάμερα από ψηλά θα κοιτάζει προς τα κάτω. Έτσι, για κάτω αριστερά γωνία έχουμε `bbox[0]=bbox[1]=0` και για πάνω δεξιά γωνία έχουμε το `bbox[0]=de_max` και `bbox[1]=dr_max`.

```
#-----
de_min=0
dr_min=0
#-----
#Calibration question
calib=input('Do you have steady camera or no? Please, give me a yes or no: ')

if calib=='yes' or 'Yes' or 'YES':
    # min_a=input('Give me the minimum value of Y axis: ')
    de_max=input('Give me the maximum value of Y axis: ')
    # min_b=input('Give me the minimum value of X axis: ')
    dr_max=input('Give me the maximum value of X axis: ')
else:
    de_max=
    dr_max=

#-----
```

Table 3 Ορισμός ελάχιστων τιμών και διαδικασία βαθμονόμησης υψηλότερων τιμών.

Λίγο πριν ξεκινήσουμε με την βασική διαδικασία που πραγματοποιείται σε εντολή while, μία εντολή η οποία συνεχίζει μέχρι να οριστεί μία πολύ συγκεκριμένη συνθήκη, πρέπει να ορίσουμε τις ψηφιακές και τις αναλογικές τιμές. Καθώς όμως οι καταστάσεις αλλάζουν, ενδέχεται ένα κομμάτι του προγραμματισμού του κώδικα να βρίσκεται σαν σχόλιο. Εκείνες οι σειρές δεν λαμβάνονται υπόψιν από τον κώδικα καθώς δημιουργούνται νέες εντολές παρακάτω.

```
#-----
# #                               Setting the digital scale
# # de = digital extension , Αναφερόμαστε στις A2 και A3 αρθρώσεις,
# # de_min η κάτω τιμή που βλέπουμε στην κάμερα , de_max η ανώτερη
# # τιμή που βλέπουμε στην προέκταση από
# # την κάμερα
# de_min= 0
# de_max= max_a
# # dr = digital round , Αναφερόμαστε στην A1 άρθρωση και τη στρέψη
# # της
# dr_min= 0
# dr_max= max_b

#                               Setting the real scale (mm)
# Οι τιμές αυτές αναφέρονται στις ελάχιστες και μέγιστες αποστάσεις
# όπου φτάνει ο ρομποτικός βραχίονας.
re_min=264
re_max=479
#-----
```

Table 4 Ορισμός ψηφιακών και πραγματικών τιμών

Έτσι, μπαίνουμε στην καρδιά του προγράμματος μας καθώς αποτελεί το σημείο όπου ο χρήστης πλέον έχει εικόνα από την κάμερα αλλά ταυτόχρονα και ο υπολογιστής δέχεται πληροφορίες μέσω αυτής. Έτσι, με μία συνθήκη while και μέσα της συνθήκες if, καταφέρνουμε να δημιουργήσουμε το αποτέλεσμα που θέλουμε. Στην πραγματικότητα, ΌΣΟ διακρίνει το αντικείμενο μέσω του αλγορίθμου, αναφέρει τις τιμές και τις αποθηκεύει σε xr & yr αντίστοιχα. ΑΝ όμως το αντικείμενο χαθεί από το FOV της κάμερας, δείχνει το μήνυμα “Lost the Object”. Έπειτα, ρυθμίζονται κάποιες άλλες παράμετροι όπως το να απεικονίζεται ο ρυθμός των FPS που έχει ο αλγόριθμος εντοπισμού του αντικειμένου μας, το χρώμα των γραμμάτων καθώς και της θέσης αυτών των πληροφοριών.

```

#-----
while True:
    timer = cv2.getTickCount()
    success, img = cap.read()

    success, bbox = tracker.update(img)
    print(bbox[0],bbox[1])
    # if calib=='no':
    #     if (bbox[0]) < min_a:
    #         min_a=bbox[0]
    #     if (bbox[0]) > max_a:
    #         max_a=(bbox[0])
    #     if (bbox[1]) < min_b:
    #         min_b=bbox[1]
    #     if (bbox[1]) > max_b:
    #         max_b=(bbox[1])

    if success:
        drawBox(img,bbox)
        xr = str(bbox[0])
        yr = str(bbox[1])

    else:
        cv2.putText(img, "Lost the Object", (75, 75),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

    fps = cv2.getTickFrequency()/(cv2.getTickCount()-timer)
cv2.putText(img, str(int(fps)), (75,50), cv2.FONT_HERSHEY_SIMPLEX, 0.7,
(0,0,255), 2)
    cv2.imshow("Traking",img)

    if cv2.waitKey(1) & 0xff == ord('q'):
        break

#-----

```

Table 5 Βασική εντολή του κώδικά μας για την διαδικασία του εντοπισμού.

Σαν τελευταίο σκαλί πριν την ολοκλήρωση του κώδικά μας θα ήταν να μεταφράσουμε τις τιμές που μας δόθηκαν παραπάνω. Ο υπολογιστής μας πλέον «κοιτάζει» ψηφιακά και οι τιμές αυτές δεν αντιπροσωπεύουν την πραγματικότητα. Έτσι, θα πρέπει εμείς να μεταφράσουμε αυτές τις τιμές ώστε να μπορούμε να τις διαχειριστούμε για το πρόγραμμα μας. Για να γίνουμε πιο σαφείς, θα πρέπει να προσδιορίσουμε και να αντιστοιχίσουμε τις τιμές *xr* & *yr* πάνω στον πραγματικό γεωμετρικό χώρο σε χιλιοστά (mm). Έτσι, ακολουθεί η παρακάτω διαδικασία:


```
#-----
#
# Lets do the math!
#
# x= (xr*479)/(de_max)
# ya2=(x-445.89)/(-3.4312)
# ya3=(x-190.06)/(4.8457)
#
# Μετατροπή από Ψηφιακές σε Αναλογικές
τιμές ( c & a του τριγώνου)
b=(xr*479)/de_max
a=(yr*479)/dr_max
#
# Βρίσκουμε τις συντεταγμένες του
αντικειμένου μέσω της τριγωνομετρίας
c=(float)(( a^2 + b^2 )^(0.5))
#
# Βρίσκουμε τις μοίρες των αρθρώσεων A2
και A3
A2 = ((c - 190.06)/4.8457)
A3 = -((c-445.89)/3.4312)
#
# I see triangles!
# Σειρά έχει να βρούμε την περιστροφή της άρθρωσης A1 και σε
συνδυασμό με αυτό μπορούμε να βρούμε
# και την απόσταση του αντικειμένου από το κέντρο του βραχίονα.
dmid=dr_max/2
if yr<dmid:
    c=dmid-yr
    pos="L"
if yr==dmid:
    c=0
    A1=0
if yr>dmid:
    c=dmid-(dr_max-yr)
    pos="R"
r=1/(sin(a/b))
#-----
```

Table 6 Διαδικασία μετατροπής ψηφιακών τιμών σε πραγματικές τιμές

Στο τελικό σκαλί του κώδικά μας συναντάμε την διαδικασία δημιουργίας κατάλληλου αρχείου για την ανάγνωσή του από το πρόγραμμα CPRog. Το πρόγραμμα αυτό διαβάζει αρχεία xml οπότε εμείς πρέπει να το δημιουργήσουμε και να το «γράψουμε» με τον τρόπο όπου εκείνο το διαβάζει. Τα xml αρχεία μπορούν να διαβαστούν από το γνωστό notepad ή ακόμα καλύτερα από το notepad++ το οποίο μας δείχνει και την δομή (structure) του κώδικα στον οποίο το πρόγραμμα CPRog υπακούει.

Τέλος, προσαρμόζουμε την παραπάνω μήτρα του κώδικα σύμφωνα με τις καταστάσεις που χρειαζόμαστε και κάνουμε τις απαραίτητες αλλαγές στις τιμές. Ο τελικός κώδικας της διπλωματικής αυτής, είναι ο παρακάτω:

```

import math
import cv2

print('-' * 40)
info = input('You will now use the tracking mode. Make a square in
your screen that covers the object of your\n'
            'interest. If the tracking process is lost, or if you
have selected the object and you are sure,\n'
            'please press the button -q- to continue. Once the
camera is open, select the ROI and press ENTER \n'
            'or SPACE. To start the process, press enter ')
print('-' * 40)

# -----
-----

cap = cv2.VideoCapture(1)

tracker = cv2.legacy_TrackerMOSSE.create()
# tracker = cv2.TrackerCSRT_create()
success, img = cap.read()
bbox = cv2.selectROI("Tracking", img, False)
tracker.init(img, bbox)

def drawBox(img, bbox):
    x, y, w, h = int(bbox[0]), int(bbox[1]), int(bbox[2]),
int(bbox[3])
    cv2.rectangle(img, (x, y), ((x + w), (y + h)), (0, 0, 255), 3, 1)
    cv2.putText(img, "Tracking", (75, 75), cv2.FONT_HERSHEY_SIMPLEX,
0.7, (0, 0, 255), 2)

# -----
-----

demin = int(0)
drmin = int(0)
demax = 280
drmax = 400
# -----
-----

# Calibration question
# calib=input('Do you have steady camera or no? Please, give me a yes
or no: ')

# if calib=='yes' or 'Yes' or 'YES':
#     # min_a=input('Give me the minimum value of Y axis: ')
#     demax=input('Give me the maximum value of Y axis: ')
#     # min_b=input('Give me the minimum value of X axis: ')
#     drmax=input('Give me the maximum value of X axis: ')
# else:
# -----
-----

# #                               Setting the digital scale
# # de = digital extension , Αναφερόμαστε στις A2 και A3 αρθρώσεις,
# # de_min η κάτω τιμή που βλέπουμε στην κάμερα , de_max η ανώτερη
τιμή που βλέπουμε στην προέκταση από
# # την κάμερα
# de_min= 0
# de_max= max_a

```

```

# # dr = digital round , Αναφερόμαστε στην A1 άρθρωση και τη στρέψη
της
# dr_min= 0
# dr_max= max_b

#                                     Setting the real scale (mm)
# Οι τιμές αυτές αναφέρονται στις ελάχιστες και μέγιστες αποστάσεις
όπου φτάνει ο ρομποτικός βραχίονας.
remin = int(264)
remax = int(479)
# -----
# -----
while True:
    timer = cv2.getTickCount()
    success, img = cap.read()

    success, bbox = tracker.update(img)
    print(bbox[0], bbox[1])

    #bbox0: Round axis (XR)
    #bbox1: Distance axis (YR_

    # if calib=='no':
    #     if (bbox[0]) < min_a:
    #         min_a=bbox[0]
    #     if (bbox[0]) > max_a:
    #         max_a=(bbox[0])
    #     if (bbox[1]) < min_b:
    #         min_b=bbox[1]
    #     if (bbox[1]) > max_b:
    #         max_b=(bbox[1])

    if success:
        drawBox(img, bbox)
        xr = int(bbox[0])
        yr = int(bbox[1])

    else:
        cv2.putText(img, "Lost the Object", (75, 75),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)

        fps = cv2.getTickFrequency() / (cv2.getTickCount() - timer)
        cv2.putText(img, str(int(fps)), (75, 50),
cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
        cv2.imshow("Traking", img)

        if cv2.waitKey(1) & 0xff == ord('q'):
            break

# -----
# -----
#                                     Lets do the math!

# x= (xr*479)/(de_max)
# ya2=(x-445.89)/(-3.4312)
# ya3=(x-190.06)/(4.8457)

```



```

-----
# -----
-----

print('Minimum and maximum values')
print('-' * 40)
print(a, b, c, r, rr , A2, A3)
# print(min_b,max_b)
print('-' * 40)
with open('dionTho.xml', 'w') as f:
    # xr = str(float(xr) - 50)
    # yr = str(float(yr) - 100)
    print(xr, yr, pos)
    nr1 = str(1)
    nr2 = str(2)
    nr3 = str(3)
    nr4 = str(4)
    nr5 = str(5)
    nr6 = str(6)
    nr7= str(7)
    nr8=str(8)
    nr9=str(9)
    nr10=str(10)
    time=str(3)
    vrs = str(1.0)
    en = str(8)
    a22 = str(A22)
    a33 = str(A33)
    a2 = str(A2)
    a3 = str(A3)
    a4 = str(0)
    a5 = str(0)
    a6 = str(0)
    e1 = str(0)
    e2 = str(0)
    e3 = str(0)
    vP = str(50)
    acc = str(50)
    smooth = str(20)
    r = str(r)
    L1=str(90)
    L2=str(40)
    L3=str(0)
    L=str(0)
    K1=str(90)
    K2=str(20)
    K3=str(8)
    gr1=str(100)
    gr2=str(0)

    # where = input ("Που θες να φτάσεις; ")
    f.write(
        '<?xml version="' + vrs + '" encoding="' + 'utf-' + en +
        '"?>\n<!-- values in mm and degree -->\n<Program>\n<Header  />\n'

        '<Gripper Nr="' + nr1 + '" Position="' + gr1 + '" Descr="" />\n'
        '<Joint
Nr="' + nr2 + '" a1="' + L1 + '" a2="' + L2 + '" a3="' + L3 + '"
a4="' + L + '" a5="' + L + '" a6="' + L + '" e1="' + e1 + '" e2="' +

```

```

e2 + '" e3="' + e3 + '" velPercent="' + vP + '" acc="' + acc + '"
smooth="' + smooth + '" AbortCondition="false" Descr="" />\n'
'<Joint
Nr="' + nr3 + '" a1="' + r + '" a2="' + a22 + '" a3="' + a33 + '"
a4="' + L + '" a5="' + L + '" a6="' + L + '" e1="' + e1 + '" e2="' +
e2 + '" e3="' + e3 + '" velPercent="' + vP + '" acc="' + acc + '"
smooth="' + smooth + '" AbortCondition="false" Descr="" />\n'
'<Wait
Nr="' + nr4 + '" Type="Time" Seconds="' + time + '" Descr="" />\n'
'<Joint
Nr="' + nr5 + '" a1="' + r + '" a2="' + a2 + '" a3="' + a3 + '" a4="'
+ a4 + '" a5="' + a5 + '" a6="' + a6 + '" e1="' + e1 + '" e2="' + e2
+ '" e3="' + e3 + '" velPercent="' + vP + '" acc="' + acc + '"
smooth="' + smooth + '" AbortCondition="false" Descr="" />\n'
'<Wait Nr="' + nr6 + '" Type="Time" Seconds="' + time + '" Descr=""
/>\n'
'<Gripper Nr="' + nr7 + '" Position="' + gr2 + '" Descr="" />\n'
'<Wait Nr="' + nr8 + '" Type="Time" Seconds="' + time + '" Descr=""
/>\n'
'<Joint Nr="' + nr9 + '" a1="' + K1 + '" a2="' + K2 + '" a3="' + K3 +
'" a4="' + L + '" a5="' + L + '" a6="' + L + '" e1="' + e1 + '" e2="'
+ e2 + '" e3="' + e3 + '" velPercent="' + vP + '" acc="' + acc + '"
smooth="' + smooth + '" AbortCondition="false" Descr="" />\n'
'<Gripper Nr="' + nr10 + '" Position="' + gr1 + '" Descr="" />\n'
'</Program>')

```

3. Αποτελέσματα και Συμπεράσματα

Σύμφωνα με την πρόοδο της παρούσας διπλωματικής, η ανάπτυξη ενός αντίστοιχου προγράμματος για την επίτευξη ενός τέτοιου σκοπού αποτελεί μία δύσκολη διαδικασία η οποία χρειάζεται πολύ χρόνο και απασχόληση. Η επίτευξη του είναι μία διαδικασία η οποία θέλει πολλές προγραμματιστικές γνώσεις, γνώσεις μαθηματικών, τριγωνομετρίας και πολλών άλλων τομέων. Ο συγκεκριμένος κώδικας έχει καλή αποτελεσματικότητα καθώς μπορεί και βρίσκει τα αντικείμενα που θέτουμε προς ανίχνευση και να τα επιλέξει όσο είναι εντός του χώρου που μπορεί να τα διαχειριστεί. Θα ήταν χρήσιμο στις επόμενες εκδόσεις του κώδικα αυτού να έχει μία διαφορετική προσέγγιση σε κάποιες παραμέτρους όπως την επιφάνεια του ROI σχετικά με το αντικείμενο που επιλέγουμε, την απόσταση της κάμερας και του πεδίου που μας ενδιαφέρει καθώς και της διαδικασίας μεταφοράς του αρχείου που δημιουργείται από τον κώδικα στον φάκελο του προγράμματος του ρομποτικού βραχίονα. Θα ήταν αρκετά βοηθητικό να συμβαίνει από μόνο του αυτό σαν διαδικασία αυτοματοποιημένα. Πέραν αυτών, ο κώδικας αυτός είναι ικανός να επιλέξει τα αντικείμενα που θέτει ο χρήστης και να τα μεταφέρει σε ένα «ασφαλή σημείο» το οποίο έχει οριστεί από τον ίδιο τον χρήστη.

Βιβλιογραφία

1. Siciliano, B. Lorenzo, S. Luigi, V. Giuseppe, O. (2013), *Ρομποτική Μοντελοποίηση, Σχεδιασμός και Έλεγχος*, Φούντας ISBN 13: 978960330748-8
2. Univesity of Waseda, “WABOT WAseda roBOT”, Ηλεκτρονική πηγή: http://www.humanoid.waseda.ac.jp/booklet/kato_2.html
3. Dan Feng, 24/4/2015 , “Performing SCARA Robots”, Ηλεκτρονική πηγή: <http://www.interactivearchitecture.org/performing-scara-robots.html>
4. Scientific Research Publishing, A. Gasoaretto, L. Scalera, 15/02/2019, Ηλεκτρονική πηγή: file:///E:/Download/A_Brief_History_of_Industrial_Robotics_in_the_20th.pdf
5. Nasa , “Nasa’s First Rover on the Red Planet” , July 3 2019 , Yvette Smith , Ηλεκτρονική πηγή: <https://www.nasa.gov/image-feature/nasas-first-rover-on-the-red-planet>
6. Honda (The power of dreams), Ηλεκτρονική πηγή: <https://global.honda/innovation/robotics/ASIMO.html>
7. elDiario.es , “This was Aarthrobot, the grandfather of surgical robots who got lost in a moving house” , Jose Manuel Blanco , June 16, 2017 – 14:09 , Ηλεκτρονική πηγή: https://www.eldiario.es/hojaderouter/tecnologia/arthrobot-geof-auchinleck-robot-cirujano-brian-day-james-mcewen-canada_1_3339049.html
8. ResearchGate, “Origins of surgical robotics: From space to the operating room”, Arpad takacs, Unre J. Rudaw , Denes Akos Nagy, Tamas Haidegger , January 2016 , Ηλεκτρονική πηγή: file:///E:/Download/Takacs_Nagy_Rudas_Haidegger_65.pdf
9. Cision PR Newswire, Sep. 27, 2017, 18:45 ET, “TSolution One® Surgical System Hits Double Digits In Total Knee Arthroplasty Cases in Korea”, Ηλεκτρονική Πηγή: <https://www.prnewswire.com/news-releases/tsolution-one-surgical-system-hits-double-digits-in-total-knee-arthroplasty-cases-in-korea-300527142.html>
10. WBS, “The ZEUS robotic system: experimental and clinical applications”, J. Marescaux MD, F. Rubino MD, “Remote Surgery” pp. 8-9 , Ηλεκτρονική πηγή: [Remote surgery - Zeus](#) .
11. ResearchGate , “Telelap Alf-X: A novel Telesurgical Surgical System for the first 21st Century”, M. Buscarini , M. Stark, Emilio R. Morales, December 2012 , Ηλεκτρονική πηγή: [file:///E:/Download/Telelap Alf-X A novel Telesurgical System for the .pdf](file:///E:/Download/Telelap_Alf-X_A_novel_Telesurgical_System_for_the_.pdf)
12. Medrobotics, Flex® Robotic System: Expanding the reach of surgery®, Ηλεκτρονική πηγή: <https://medrobotics.com/gateway/flex-robotic-system/?c=INTL>
13. Genesis Healthcare, “History of CyberKnife”. Ηλεκτρονική πηγή: <http://www.mygenesishealth.com/treatment-options/genesis-cyberknife/history-of-cyberknife.html>
14. CyberKnife®Accuray®, “CyberKnife: How It Works”, Ηλεκτρονική πηγή: <https://cyberknife.com/cyberknife-how-it-works/>
15. Researchgate, “Advances In the understanding and Treatment of Stroke Impairment Using Robotic Devices” , J. Hidler, D. Nichols, M. Pelliccio, K. Brady, Feb 2005, Ηλεκτρονική πηγή: https://www.researchgate.net/figure/MIT-MANUS-Interactive-Motion-Technologies-Cambridge-MA_fig1_7800973
16. Researchgate, “The effectiveness of robotic training depends on motor task characteristics”, L. M. Crespo, N. Rappo, R. Riener, Dec 2017 , Ηλεκτρονική πηγή:

- https://www.researchgate.net/figure/ARMin-IV-is-a-7-DoF-robotic-device-designed-for-upper-limb-rehabilitation-Nef-et-al_fig1_320227548
17. Researchgate, “A brief Review on Robotic Exoskeletons for Upper Extremity Rehabilitation to find the Gap between Research Porotype and Commercial Type”, Md Rasedul Islam, C. Spiewak, M. Ranham, r. Fareh, Jan 2017 ,Ηλεκτρονική πηγή: https://www.researchgate.net/figure/Rehabilitation-type-device-MGA-exoskeleton-26_fig3_322858480
 18. IEEE/ASME Transactions on Mechatronics, Vol. 12, No. 4, “Upper-Limb Powered Exoskeleton Design” , J. C. Perry, J. Rosen, S. Burns, August 2007, http://www.columbia.edu/~njr2121/JP_13.pdf
 19. Hocoma official site, Lokomat®, Ηλεκτρονική πηγή: <https://www.hocoma.com/solutions/lokomat-2/>
 20. U.S. National Library of Medicine - National Institutes of Health, “Robotics in Lower-Limb” Rehabilitation after stroke” , X. Zhang, Z. Yue , J. Wang , 2017 Jun 8 , Ηλεκτρονική πηγή: <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5480018/>
 21. RobotWorx, Vision Robots, Ηλεκτρονική πηγή: <https://www.robots.com/applications/vision>
 22. ABB , “ABB’s new 3D inspection robot cell makes quality control testing ten times faster” , Switcherland 2020-07-28 , Ηλεκτρονική πηγή: <https://new.abb.com/news/detail/65732/abbs-new-3d-inspection-robot-cell-makes-quality-control-testing-ten-times-faster>
 - 23.