

Πανεπιστήμιο Δυτικής Μακεδονίας

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ & ΜΗΧΑΝΙΚΩΝ

ΥΠΟΛΟΓΙΣΤΩΝ

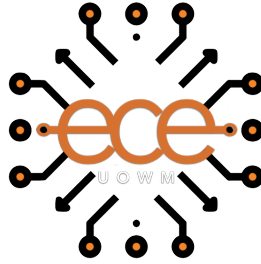
ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Σχεδίαση και υλοποίηση βοηθού
ρομποτικού βραχίονα με φωνητική
αλληλεπίδραση και χρήση μηχανικής
όρασης

Νάντζιος Γεώργιος

Επιβλέπων Καθηγητής : Δρ. Μηνάς Δασυγένης
Εργαστήριο Ρομποτικής Ενσωματωμένων και Ολοκληρωμένων
Συστημάτων

29 Ιουνίου 2021



University of Western Macedonia

FACULTY OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

DIPLOMA THESIS

Design and implementation of a robotic
arm assistant with voice interaction
using machine vision

Nantzios Georgios

Supervisor: Dr.Minas Dasygenis

Laboratory of Robotics Embedded and Integrated Systems

29 June 2021

Περίληψη

Στη σημερινή εποχή με τη συνεχόμενη ανάπτυξη της τεχνολογίας και της επιστήμης των υπολογιστών έχει δημιουργηθεί το απαραίτητο τεχνολογικό υπόβαθρο για την εξέλιξη και υλοποίηση ακόμα και των πιο περίπλοκων μηχανισμών. Η ψηφιοποίηση σχεδόν όλων σε συνδυασμό με την πρόοδο της ρομποτικής, υπόσχεται ένα μέλλον όπου η πρόσβαση σε μηχανήματα υψηλής τεχνολογίας θα είναι διαθέσιμη για όλους. Τα ρομπότ γίνονται όλο και πιο ικανά λόγω της δυνατότητας τους να εκτελούν πιο πολύπλοκους υπολογισμούς, να ενσωματώνουν ανεπτυγμένες τεχνολογίες μηχανικής μάθησης και όρασης καθώς και να αλληλοεπιδρούν με φυσικότητα με τον άνθρωπο μέσω όλο και πιο πλούσιων αισθητήρων και τελικών εργαλείων δράσης.

Λαμβάνοντας υπόψιν τη σημαντικότητα ενασχόλησης και υλοποίησης ενός τέτοιου ρομποτικού συστήματος, η παρούσα διπλωματική εργασία παρουσιάζει τον σχεδιασμό και την υλοποίηση ενός ολοκληρωμένου, αλληλοεπιδρώμενου φωνητικά, ρομποτικού βραχίονα μεταβίβασης αντικειμένων στο χρήστη. Το ρομποτικό σύστημα βασίζεται στο ρομποτικό βραχίονα Niryo-One, όπου στη περιοχή του καρπού έχει τοποθετηθεί usb camera. Τα αντικείμενα αναγνωρίζονται ως προς το χρώμα, το σχήμα και κάποια ως προς το είδος τους. Παράλληλα, δύναται η λειτουργία ανάγνωσης QR codes, όπου με την εκτύπωση και τοποθέτηση αυτών στα αντικείμενα, μπορούν να δοθούν προσωπικές ονομασίες προσφώνησης σε αυτά.

Το εν λόγω ρομποτικό σύστημα, λόγω των διαφορετικών λειτουργιών αναγνώρισης και εξατομίκευσης που προσφέρει μπορεί να χρησιμοποιηθεί σε διάφορους τομείς όπως π.χ. στη βιομηχανία, στην ιατρική περίθαλψη, στο οικιακό περιβάλλον και σε άλλους.

Λέξεις κλειδιά: Ρομποτική, Ρομποτικός βραχίονας, Niryo-One, Μηχανική Όραση, Βαθιά Μάθηση, YOLO, OpenCV, Φωνητική Αναγνώριση, Google API, Ρομπότ Βοηθός

Abstract

With the continuous growth of technology and computer science, the necessary technological background has been created for the evolution and implementation of even the most complex mechanisms. The digitization of practically everything, coupled with advances in robotics, promises a future where access to high-tech machines will be available for all. Robots are becoming increasingly capable due to their ability to execute more complex computations, integrate advanced machine learning and vision technologies, and interact naturally with humans through increasingly richer sensors and end-effectors.

Considering the importance of occupation and implementation of such a robotic system, the present diploma thesis aims to design and implement a comprehensive voice interaction robotic arm to deliver objects to the user. The robotic system is based on a Niryo-One robotic arm where a USB camera has been installed in the area of the end-effector. The objects are recognized based on the colour, shape, and some of them in terms of their type. At the same time, the QR code reading function can be used to print and place them on the items in order to call them with a personal preference name.

This robotic system, due to the variety of recognition and customization functions it offers, can be used in various fields such as industry, medical care, home environment and others.

Keywords: Robotics, Robotic arm, Niryo-One, Machine Vision, Deep Learning, YOLO, OpenCV, Speech Recognition, Google API, Speech to Text, Raspberry Pi, Assistance robot

Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν.1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο «Σχεδίαση και υλοποίηση βοηθού ρομποτικού βραχίονα με φωνητική αλληλεπίδραση και χρήση μηχανικής όρασης» καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Μηνά Δασυγένη αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή. Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα. Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και μόνο.

Copyright (C) Νάντζιος Γεώργιος & Δασυγένης Μηνάς, 2021, Κοζάνη

Υπογραφή Φοιτητή

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τους ανθρώπους που συνέβαλλαν ενεργά στη διεκπεραίωση της παρούσας διπλωματικής. Στη συνέχεια, θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή Δρ. Μηνά Δασυγένη για την εμπιστοσύνη και την στήριξη που μου έδειξε καθ' όλη τη διάρκεια της συνεργασίας μας, τόσο στα πλαίσια εκπόνησης της διπλωματικής όσο και συνολικά στα έτη φοίτησης μου.

Ένα εγκάρδιο ευχαριστώ ανήκει στους φίλους για την αμέριστη κατανόηση, συμπαράσταση και στήριξη που επέδειξαν ιδιαίτερα τα τελευταία χρόνια.

Τέλος, το μεγαλύτερο ευχαριστώ το οφείλω στους γονείς μου Άγγελο και Χριστίνα καθώς και στον αδελφό μου Ισίδωρο για την στήριξη, τις θυσίες και την αγάπη τους όλα αυτά τα χρόνια, δίνοντας μου κουράγιο να προχωρώ και να ξεπερνώ κάθε εμπόδιο για να φτάσω στον στόχο μου.

Περιεχόμενα

1	Εισαγωγή	15
1.1	Ορισμός του Προβλήματος	15
1.2	Ιστορία της Ρομποτικής	16
1.3	Η ρομποτική ως συνέχεια της Τεχνολογικής Επανάστασης	18
1.4	Τι είναι το ρομπότ;	19
1.5	Η ρομποτική στη σημερινή εποχή	20
1.6	Αντικείμενο Διπλωματικής Εργασίας	22
1.7	Κίνητρο Διεξαγωγής Εργασίας	23
1.8	Συμβολή στην επιστημονική κοινότητα	25
1.9	Παρόμοια Ερευνητικά Έργα	25
1.10	Περιγραφή Κεφαλαίων	29
2	Θεωρητικό Υπόβαθρο	30
2.1	Προγραμματισμός Όρασης Μηχανής	30
2.1.1	Python	30
2.1.2	PyCharm (IDE)	32
2.1.3	Notepad++	33
2.1.4	Niryo-One Studio	33
2.2	Όραση Μηχανής	34
2.2.1	Ορισμός της έννοιας όραση μηχανής	34
2.2.2	Βιβλιοθήκη OpenCV	37
2.2.3	Λογισμικό ZBar	39
2.3	Τεχνικές Βαθιάς Μηχανικής Μάθησης και Αναγνώρισης Αντικειμένων	40
2.3.1	Εισαγωγή στη Τεχνητή Νοημοσύνη	40
2.3.2	Μηχανική μάθηση	40
2.3.3	Τεχνητά Νευρωνικά Δίκτυα	43

2.3.4	Βαθιά Μάθηση	49
2.3.5	Συνελικτικά Νευρωνικά Δίκτυα	49
2.3.6	Μοντέλα Ανίχνευσης Αντικειμένων	56
2.4	Αναγνώριση φωνής	71
2.4.1	Τρόπος λειτουργίας	71
2.4.2	Google voice Recognition API	73
2.5	Αναπαραγωγή φωνής	74
2.5.1	Τρόπος Λειτουργίας	75
2.5.2	Βιβλιοθήκη pyttsx3	77
2.6	Αρχιτεκτονική Ρομποτικού Βραχίονα	77
2.6.1	Είδη Αρθρώσεων	79
2.6.2	Βαθμοί ελευθερίας και κινητικότητας	81
2.6.3	Χώρος Εργασίας	82
2.6.4	Κινηματική Ανάλυση	83
2.6.5	Περιγραφή βραχίονα Niryo-One	90
2.6.6	Τεχνικά χαρακτηριστικά	93
2.7	Σύνοψη κεφαλαίου	96
3	Σχεδίαση και υλοποίηση λογισμικού	97
3.1	Γενική περιγραφή λειτουργίας συστήματος	97
3.2	Όραση μηχανής	102
3.2.1	Φίλτρα εικόνας	104
3.2.2	Εντοπισμός και αναγνώριση σχήματος και χρώματος αντικειμένων	109
3.2.3	Εντοπισμός και αναγνώριση των αντικειμένων με τον αλγόριθμο Yolo	111
3.2.4	Εντοπισμός και αποκωδικοποίηση QR codes	113
3.2.5	Εντοπισμός περιγράμματος αντικειμένων	114
3.2.6	Παραδείγματα εντοπισμού και αναγνώρισης αντικειμένων σε διάφορες συνθήκες φωτισμού	116
3.3	Φωνητικές εντολές και αλληλεπίδραση	122
3.3.1	Αναγνώριση φωνητικών εντολών	123
3.3.2	Φωνητική απάντηση	125

3.4	Εκτέλεση κινήσεων	125
3.4.1	Niryo-One Python API	126
3.4.2	Μετατροπή συντεταγμένων μηχανικής όρασης σε συντεταγμένες χώρου	126
3.5	Μεταφορά δεδομένων μεταξύ σεναρίων κώδικα	131
3.6	Σύνοψη μετρικών κώδικα	133
3.7	Συμβάσεις λογισμικού μέρους	135
3.8	Σύνοψη κεφαλαίου	137
4	Πειραματική Διαδικασία	138
4.1	Ορισμός πειραμάτων	138
4.2	Εκτέλεση πειραμάτων	141
4.3	Αποτελέσματα και συζήτηση	149
5	Συμπεράσματα και μελλοντικές επεκτάσεις	153
5.1	Σύνοψη και συμπεράσματα	154
5.2	Μοντέλο ανάλυσης συστήματος (S.W.O.T)	156
5.3	Προβλήματα που προέκυψαν κατά την εκπόνηση	157
5.4	Μελλοντικές επεκτάσεις	160

Κατάλογος σχημάτων

2.1 Το ολοκληρωμένο αναπτυξιακό περιβάλλον Pycharm.	33
2.2 Οι αριθμοί που αντιλαμβάνεται το σύστημα αντιστοιχούν στην χρωματική ένταση των επιμέρους εικονοστοιχείων	36
2.3 Σύνδεση του πεδίου της όρασης μηχανής με άλλα επιστημονικά πεδία	37
2.4 Σύνδεση του πεδίου της όρασης μηχανής με άλλα επιστημονικά πεδία	38
2.5 Διάγραμμα Venn των διαφόρων κατηγοριών μηχανικής μάθησης	42
2.6 Βιολογικός νευρώνας	44
2.7 Σιγμοειδή Συνάρτηση	45
2.8 Συνάρτηση tanh	45
2.9 Συνάρτηση ReLU	46
2.10 Συνάρτηση Leaky ReLU	47
2.11 Απλό μοντέλο NN με ένα κρυφό επίπεδο - Perceptron	48
2.12 Τρισδιάστατη κατανομή εικόνας	51
2.13 Διαδικασία συνέλιξης	52
2.14 Εφαρμογή Φίλτρων	53
2.15 Μέθοδοι Συγκεντρωτικού Επιπέδου	55
2.16 Συνελικτικό Δίκτυο	56
2.17 Μοντέλο Darknet-53	61
2.18 Prediction Feature Map	64
2.19 Αγκυρωμένα πλαίσια	65
2.20 Σχηματική απεικόνιση υπολογισμού των Bounding Boxes	67
2.21 Σχηματική απεικόνιση του Intersection over Union	69
2.22 Επίδραση της Non-max suppression	70
2.23 Επίπεδα επεξεργασίας σήματος φωνής	73
2.24 Στάδια σύνθεσης ομιλίας	76
2.25 Τυπική μορφή βιομηχανικού βραχίονα (Cincinnati Milacron T3)	78

2.26 Τα είδη των αρθρώσεων	80
2.27 Σχηματική απεικόνιση χώρου εργασίας	82
2.28 Σχηματική απεικόνιση βραχίονα 2 DOF	84
2.29 Σύστημα συντεταγμένων κατά Denavit-Hartenberg	85
2.30 Ο προσανατολισμός του στερεωμένου στο σώμα συστήματος συντεταγμένων ως προς το σταθερό στο χώρο σύστημα συντεταγμένων περιγράφεται από τρεις γωνίες	89
2.31 Γωνίες Roll Pitch Yaw	90
2.32 Γωνίες Roll Pitch Yaw	90
2.33 Ρομποτικός Βραχίονας Niryo-One	91
2.34 Ρομποτικός βραχίονας Niryo-One με χρωματισμένα μέλη	93
2.35 Μονάδα ελέγχου και επεξεργασίας	94
2.36 Σχηματική απεικόνιση περιστροφής και ονοματοδοσίας αρθρώσεων	94
2.37 Διαστάσεις συνδέσμων	95
2.38 Κάτοψη	95
2.39 Gripper 2	96
2.40 Κάτοψη Gripper 2 (σε πλήρη έκταση)	96
2.41 Κάτοψη Gripper 2 (σε σύγκλιση)	96
3.1 Διάγραμμα ροής συστήματος	100
3.2 Αρχιτεκτονική συστήματος	101
3.3 Διάγραμμα ροής μηχανικής όρασης	103
3.4 Φίλτρο διάβρωσης εικόνας	106
3.5 Φίλτρο διαστολής εικόνας	106
3.6 Φίλτρο κλεισίματος εικόνας	107
3.7 Γκαουσιανό θόλωμα	108
3.8 Φίλτρο κατωφλίωσης	109
3.9 Συνθήκες καλού φωτισμού	118
3.10 Συνθήκες κακού φωτισμού	118
3.11 Συνθήκες κακού φωτισμού	119
3.12 Συνθήκες χαμηλού φωτισμού	120
3.13 Συνθήκες φυσικού φωτισμού	120
3.14 Συνθήκες καλού φωτισμού	121

3.15 Εντοπισμός και αναγνώριση QR code	121
3.16 Κάδρο λήψης κάμερας	127
3.17 Θέση αναμονής ή θέση εκκίνησης	129
3.18 Περιοχές αναζήτησης	130
3.19 Σημάδια για τον προσδιορισμό των ακραίων σημείων χώρου	130
3.20 Αποτελέσματα κυκλωματικής πολυπλοκότητας	134
3.21 Αποχρώσεις του άσπρου	136
4.1 Αρχική κατάσταση έναρξης πειραμάτων	141
4.2 Κάτοψη αρχικής κατάστασης	142
4.3 Κάτοψη θέσης Α	143
4.4 Κάτοψη θέσης Β	143
4.5 Κάτοψη θέσης Γ	144
4.6 Κάτοψη θέσης Δ	144
4.7 Κάτοψη θέσης Ε	145
4.8 Πιάσιμο του αντικειμένου	146
4.9 Πιάσιμο του αντικειμένου (μεγέθυνση)	146
4.10 Παράδοση αντικειμένου	147
4.11 Παράδοση αντικειμένου (μεγέθυνση)	147
4.12 Αναγνώριση χεριού	148
4.13 Ροή εκτέλεσης του λογισμικού	148
5.1 Ανάλυσης συστήματος (S.W.O.T)	156
5.2 Τερματικό περιβάλλον για την αλλαγή του ID	158
5.3 Σπασμένο γρανάζι	159
5.4 Σπασμένο γρανάζι (μεγέθυνση)	159

Κατάλογος πινάκων

3.1	Μετρικές αρχείων κώδικα.	134
3.2	Μετρικές αρχείων κώδικα.	135
4.1	Σχεδιασμός πειραμάτων.	140
4.2	Αποτελέσματα πειραμάτων στο φορητό υπολογιστή	149
4.3	Αποτελέσματα πειραμάτων στο Raspberry Pi 4	150

Κατάλογος εξισώσεων

2.1	Σιγμοειδή	45
2.2	Υπερβολική Εφαπτομένη	45
2.3	ReLU	46
2.4	Leaky ReLU	47
2.5	Εξόδος νευρώνα	47

Κεφάλαιο 1

Εισαγωγή

1.1 Ορισμός του Προβλήματος

Η ραγδαία εξέλιξη των τεχνολογιών πληροφορικής τόσο στον τομέα του υλικού με τη συνεχόμενη μείωση του κόστους κατασκευής των ενσωματωμένων συστημάτων όσο και στον τομέα του λογισμικού, έδωσε τη δυνατότητα στους μηχανικούς και επιστήμονες να αναπτύσσουν προηγμένα ρομποτικά συστήματα σε χαμηλό κόστος σε σχέση με τα προηγούμενα χρόνια [1], [2]. Τεχνολογίες όπως η μηχανική όραση και η τεχνητή νοημοσύνη, έχουν δώσει την δυνατότητα εφαρμογής της ρομποτικής σε διάφορους τομείς της καθημερινής ζωής των ανθρώπων όπως στην κοινωνική μέριμνα, στην φροντίδα των ηλικιωμένων και ΑμΕΑ, στην υγεία κ.α. [3], [4]. Η τεχνολογική αυτή εξέλιξη έχει ως αποτέλεσμα μία συνεχής και αυξανόμενη επικοινωνία και κοινωνική αλληλεπίδραση των ανθρώπων με τα ρομπότ. Εστιάζοντας στα ρομποτικά συστήματα που βρίσκονται σε επικοινωνιακή αλληλεπίδραση με τον άνθρωπο, σε ρόλους ρομπότ βοηθών, παρατηρούμε ότι έχουν γίνει αρκετές προσπάθειες κάλυψης των διάφορων αναγκών που προκύπτουν εξυπηρετώντας τους εκάστοτε σκοπούς [5], [6]. Είναι όμως φανερό ότι στη πλειονότητα αυτών είναι ακόμα δύσκολη η απόκτηση τους λόγω κόστους και δεν είναι πάντα εύκολο να προσαρμοστούν στις μεταβλητές ανάγκες του εκάστοτε χρήστη.

Λαμβάνοντας υπόψιν όλους τους παραπάνω παράγοντες αναπτύχθηκε ο εν λόγω βοηθός ρομποτικός βραχίονας. Στόχος του συστήματος αυτού, όπως περιγράφεται και παρακάτω είναι η παροχή βοήθειας μέσω της μεταβίβασης αντικειμένων στο χρήστη με τρόπο εφάμιλλο και οικείο στην αλληλεπίδραση με τον άνθρωπο.

1.2 Ιστορία της Ρομποτικής

Η ιστορία της ρομποτικής είναι συνυφασμένη με την άνοδο της τεχνολογία, της επιστήμης και την βασική αρχή της προόδου, γεγονότα που καθόρισαν το κοινωνικό γίγνεσθαι τους τελευταίους αιώνες.

Η ιστορία ξεκινά να γράφεται ήδη από τα αρχαία χρόνια και την ελληνική μυθολογία. Σύμφωνα με το Απολλόδωρο, ο Τάλως ήταν ένα μυθικό γιγάντιο χάλκινο δημιούργημα του θεού Ήφαιστου, το οποίο δόθηκε ως δώρο στον Βασιλιά Μίνωα για να προστατεύει την Κρήτη από τους εχθρούς της. Οι πρώτες κατασκευές ξεκίνησαν να δημιουργούνται από το 400 π.Χ. , με την αρχή να γίνεται από τον μαθηματικό Αρχύτα τον Ταραντινό (428-347 π.Χ.), ο οποίος φημολογείται πως κατασκεύασε μια ιπτάμενη μηχανή. Η ‘περιστερά’ ή ‘πετομηχανή’ όπως ονομάστηκε, χρησιμοποιούσε τον ατμό ως κινητήρια δύναμη και μπορούσε να διανύσει μέχρι και 200 μέτρα [7]. Αργότερα στο διάστημα (1452-1519 μ.Χ.), ο Ιταλός πολυμαθής Leonardo da Vinci σχεδιάζει και (πιθανόν κατασκευάζει) ανθρωποειδή ρομπότ με πανοπλία έχοντας τη δυνατότητα να κινεί τα χέρια και το κεφάλι του και να ανασηκώνεται. Τον 18 αιώνα ο Jacques de Vaucanson ξεκινάει την κατασκευή ρομπότ [8]. Η πιο διάσημη κατασκευή του ήταν μια ρομποτική πάπια που είχε τη δυνατότητα να κινείται, να κουνάει τα φτερά της και να τρώει σπόρους. Η κατασκευή αυτή, αποτελούσε ένα παράδειγμα στα πλαίσια της μελέτης του Vaucanson στον τομέα της λεγόμενης ‘κινούμενης ανατομίας’, δηλαδή στην μοντελοποίηση της ανατομίας ανθρώπων και ζώων με χρήση μηχανισμών. Το 1898, ο Σέρβος εφευρέτης και μηχανικός Nikola Tesla παρουσιάζει το πρώτο τηλεχειριζόμενο πλοίο.

Η έννοια «ρομπότ» εμφανίστηκε και χρησιμοποιήθηκε το 1921 από τον Τσέχο συγγραφέα, δραματογράφο και κριτικό Karel Čapek, ο οποίος χρησιμοποίησε αυτό τον όρο στο θεατρικό έργο του R.U.R. (Rossum’s Universal Robots). Αυτός ο όρος επινοήθηκε από την τσέχικη λέξη «robota», η οποία έχει τη σημασία της καταναγκαστικής εργασίας. Στο έργο του ο συγγραφέας φαντάζεται ένα αυτόματο μηχανικό κατασκεύασμα το “Rossum” όπου στο τέλος στρέφεται τελικά εναντίον της ανθρωπότητας. Μέσω αυτού σατιρίζεται η εξάρτηση της κοινωνίας από τους μηχανικούς

εργάτες (ρομπότ) της τεχνολογικής εξέλιξης οι οποίοι τελικά εξοντώνουν τους δημιουργούς τους.

Το 1942, ο Issac Asimov συγγραφέας επιστημονικής φαντασίας στο βιβλίο «I, Robot», [9] εισαγάγει τη λέξη Robotics και περιγράφει το ρομπότ ως μια μεταλλική με γρανάζια ανθρωποειδή κατασκευή απαλλαγμένη από συναισθήματα και προσωπικότητα. Η μεγαλύτερη όμως συνεισφορά του Asimov, πέρα από την εισαγωγή της έννοιας της Ρομποτικής, είναι η δημιουργία των τριών βασικών νόμων των ρομπότ:

- Νόμος 1: Ένα ρομπότ δεν πρέπει ποτέ να βλάψει ένα άνθρωπο ή λόγω αδράνειας του να αφήσει ένα άνθρωπο να πάθει κακό.
- Νόμος 2: Ένα ρομπότ πρέπει πάντοτε να υπακούει τις εντολές που δίνονται από τους ανθρώπους εκτός και αν συγκρούονται με ένα ανώτερο νόμο.
- Νόμος 3: Ένα ρομπότ πρέπει να προστατέψει την ύπαρξη του εκτός και αν αυτό συγκρούεται με ένα ανώτερο νόμο.

Τα πρώτα σύγχρονα ρομπότ με τη μορφή βραχίονα κάνουν την εμφάνισή τους τις δεκαετίες του 1940 και 1950 όπου προορίζονται για αυτοματοποιημένη βιομηχανική χρήση. Το 1966 παρουσιάζεται από το Ινστιτούτο Έρευνας του πανεπιστημίου του Stanford, το Shakey, το πρώτο κινούμενο ρομπότ με δυνατότητες Τεχνητής Νοημοσύνης [10] . Οι λειτουργίες που μπορούσε να εκτελέσει περιορίζονταν στον οπτικό εντοπισμό ενός αντικειμένου, την μετάβαση και την αλληλεπίδραση με αυτό όπως για παράδειγμα να το σπρώξει.

Τα τελευταία χρόνια, η επιστήμη της ρομποτικής εμπλέκεται σε μια πληθώρα εφαρμογών η ανάπτυξη των οποίων θα συνεχίσει να επεκτείνεται με ολοένα νέα είδη ρομπότ κατάλληλα για τη βιομηχανία, την επιστήμη και την καθημερινή ζωή του ανθρώπου, καθιστώντας την αναπόσπαστο κομμάτι στην εξέλιξη της ανθρωπότητας.

1.3 Η ρομποτική ως συνέχεια της Τεχνολογικής Επανάστασης

Ο 19ος αιώνας σηματοδότησε την επιτάχυνση και την καθολική υιοθέτηση των βιομηχανικών διεργασιών. Η αρχή του αιώνα, λόγω της Βιομηχανικής Επανάστασης, βρισκόταν σε μια διαδικασία προσαρμογής και μέχρι το τέλος είχε αναπτυχθεί το αυτοκίνητο και μόλις επρόκειτο να εμφανιστούν τα πρώτα αεροσκάφη. Το αντίκτυπο στη ζωή των ανθρώπων ήταν τεράστιο. Επήλθε διαμόρφωση και προσαρμογή στους κοινωνικούς και οικονομικούς κανόνες που διέπουν την υγειονομική περίθαλψη, την κατασκευή, τα μέτρα ασφαλείας, το εργασιακό περιβάλλον και τη ζωή στο σπίτι. Τον 20ο αιώνα, αυτή η επανάσταση επαναλήφθηκε με την μορφή Τεχνολογική, αλλά αυτή τη φορά με πιο γρήγορους ρυθμούς. Οι νέοι ορίζοντες που διαμορφώθηκαν από τους εξελισσόμενους τομείς της ηλεκτρονικής, των τηλεπικοινωνιών, του αυτοματισμού και της κοστολόγησης, ήταν οι κινητήριες δυνάμεις ώστε η τεχνολογία να μεταφερθεί πιο άμεσα από το εργαστήριο και τα ερευνητικά ιδρύματα στο σπίτι. Στις αρχές του 1990, οι πρώτοι υπολογιστές αρχίζουν να μπαίνουν στην καθημερινότητα ενώ ήταν σχεδόν άγνωστοι πριν από εκατό χρόνια.

Φθάνοντας στο παρόν, βρισκόμαστε στο επίκεντρο μιας νέας τεχνολογικής ανάπτυξης ίσης σημασίας: της Ρομποτικής Επανάστασης. Αυτή η επανάσταση θα τοποθετήσει τον 21ο αιώνα σε περίοπτη θέση στην ιστορία. Η επανάσταση αυτή, θα επηρεάσει καθοριστικά τις ζωές μας και τις ζωές των μελλοντικών γενεών.

Η ανάπτυξη της ρομποτικής, δημιούργησε ένα ευρύ φάσμα εφαρμογών, από τις πιο καθημερινές και πλέον δεδομένες έως τις πιο εξειδικευμένες και πολυσύνθετες. Θα επιτρέψουν την εκτέλεση παραγωγικών δραστηριοτήτων και καθηκόντων που δεν μπορούν να εκτελεστούν από ανθρώπους, όπως είναι η ανάλυση, ο έλεγχος και η επεξεργασία μαζικών δεδομένων. Ως ιδιαίτερα σημαντικούς τομείς μπορούμε να αναφέρουμε την ιατρική σε εφαρμογές χειρουργικών επεμβάσεων για σταθερές και λεπτεπίλεπτες κινήσεις, στην Εθνική Άμυνα και ασφάλεια σε εφαρμογές αντιμετώπισης επικίνδυνων συμβάντων, εκτάκτων καταστάσεων και πρόληψης. Επιπρόσθετα, στη βιομηχανία, σε τομείς συναρμολόγησης, συγκόλλησης σε απαγορευτικές για τον άνθρωπο συνθήκες, στη γεωργία, εισάγοντας νέες διαδικασίες σε όλα τα στά-

δια παραγωγής. Ακόμα, στον κατασκευαστικό τομέα σε εργοτάξια, στη διαδικασία οικοδόμησης και κατασκευής με χρήση ρομποτικών βραχιόνων για 3D εκτύπωση ικανών να μειώσουν το χρόνο έως και 4 φορές σε σύγκριση με επαγγελματίες οικοδόμους σύμφωνα με τη Fastbrick Robotics Αυστραλίας [11]. Επιπλέον, υπό τις παρούσες συνθήκες γήρανσης του πληθυσμού, τα ρομπότ αποτελούν μια λύση για το πρόβλημα της αυξανόμενης έλλειψης εργατικού δυναμικού για την εκτέλεση χειρωνακτικών εργασιών [12].

Η στρατηγική της ΕΕ στον τομέα της ρομποτικής για το 2020 [13] περιγράφει τις τρέχουσες εξελίξεις ως εξής:

«Η ρομποτική τεχνολογία θα κυριαρχήσει την επόμενη δεκαετία. Θα επηρεάσει κάθε πτυχή της εργασιακής και της προσωπικής ζωής. Η ρομποτική μπορεί να μεταβάλει τη ζωή μας και τις εργασιακές μας πρακτικές, να αυξήσει το επίπεδο της αποτελεσματικότητας και της ασφάλειας, να βελτιώσει την ποιότητα των υπηρεσιών και να δημιουργήσει θέσεις εργασίας. Ο αντίκτυπός της θα αυξάνεται με την πάροδο του χρόνου, όπως θα αυξάνεται και η διαδραστικότητα μεταξύ ρομπότ και ανθρώπων».

1.4 Τι είναι το ρομπότ;

Προσπαθώντας να δώσουμε ένα ορισμό για το ρομπότ η δόκτωρ Σοφία Δρακάκη στο άρθρο της, «*Τι ακριβώς είναι η ρομποτική*» [14] αναφέρει:

«*Το Ρομπότ είναι μία κατασκευή, που μπορεί να εκτελεί προγραμματισμένες εργασίες, αλλά σε αντίθεση με τις απλές μηχανές ακολουθεί τη μέθοδο: αντιλαμβάνομαι, σκέπτομαι, ενεργώ. Το Ρομπότ καθοδηγείται από ένα ηλεκτρονικό κύκλωμα και από κάποιο πρόγραμμα υπολογιστή και αποτελείται από δομικά μέρη (δοκοί, γρανάζια, ρόδες ..), ηλεκτρονικά (μονάδα εγκεφάλου...), αισθητήρες (χρώματος, φωτός, κίνησης, απόστασης.....) και κινητήρες. Συνήθως χρησιμοποιείται για την εκτέλεση επικίνδυνων ή δύσκολων (για ανθρώπους) εργασιών ή για να αυτοματοποιήσει μονότονες επαναλαμβανόμενες εργασίες, που ωστόσο πρέπει να γίνονται με μεγάλη ταχύτητα και ακρίβεια»*

Βάση της εφαρμογής, το ρομπότ μπορεί να ταξινομηθεί είτε ως βιομηχανικό είτε ως ρομπότ παροχής υπηρεσιών [15].

- Ως βιομηχανικό ρομπότ η Διεθνής Ομοσπονδία Ρομποτικής ορίζει έναν «αυτόματα ελεγχόμενο, επαναπρογραμματιζόμενο, πολλαπλών χρήσεων βραχίονα, προγραμματιζόμενο σε τρεις ή περισσότερους άξονες, που μπορεί να είναι είτε σταθερός είτε κινητός για χρήση σε βιομηχανικές εφαρμογές αυτοματισμού». (ορισμός που περιέχεται στο πρότυπο [ISO 8373:1994])
- Τα ρομπότ παροχής υπηρεσιών έχουν ως σκοπό να υποστηρίζουν, να κρατούν συντροφιά και να περιθάλπουν ανθρώπους. Ενεργούν με τη βασική ευφυΐα που απαιτείται για την εκτέλεση των καθηκόντων που τους ανατίθενται. Τα συγκεκριμένα ρομπότ ταξινομούνται σε τρεις κατηγορίες: Τα ρομπότ της κατηγορίας 1, αντικαθιστούν τους ανθρώπους στην εκτέλεση εργασιών σε δυσμενή περιβάλλοντα εργασίας, καθώς και σε κοπιαστικές δραστηριότητες. Τα ρομπότ της κατηγορίας 2, επικεντρώνονται στο να κάνουν πιο άνετη τη ζωή των ανθρώπων, παρέχοντας υπηρεσίες ψυχαγωγίας, βοήθειας προς τους ηλικιωμένους, περίθαλψης ασθενών και συνεργατικής εργασίας. Τα ρομπότ της κατηγορίας 3, εκτελούν εργασίες σε ανθρώπους, όπως π.χ. τα ιατρικά ρομπότ που χρησιμοποιούνται για χειρουργικές επεμβάσεις, διαγνωστικούς σκοπούς, θεραπείες για αποκατάσταση

1.5 Η ρομποτική στη σημερινή εποχή

Ο άνθρωπος έχοντας αντιληφθεί τη δύναμη της μάθησης και της νόησης, προσπάθησε να δημιουργήσει ρομπότ με τις δεξιότητες αυτές. Έτσι, κατά τη δεκαετία του 1980, δημιουργούνται και αναπτύσσονται οι κλάδοι της τεχνητής νοημοσύνης και μηχανικής μάθησης που συνέβαλαν στην αύξηση του αριθμού των κλάδων στους οποίους μπορούσε να εφαρμοστεί η ρομποτική. Πλέον, δεν ήταν μια απλή κατασκευή και εκτέλεση μιας προκαθορισμένης αλληλουχίας κινήσεων και αντιδράσεων αλλά πλέον είναι εφοδιασμένη με ικανότητα μάθησης, δυνατότητα κατανόησης της φυσικής γλώσσας, ικανότητα αξιολόγησης στοιχείων, λήψης αποφάσεων κ.ά.

Η κοινωνία προσπαθεί να προσαρμοστεί στις τάσεις αυτοματοποίησης που παρουσιάζονται. Η σταδιακή αύξηση της παρουσίας της ρομποτικής και κατ' επέκταση των ρομπότ στην καθημερινότητα μας, τόσο στο οικιακό περιβάλλον όσο και στην εργασία, έχει δημιουργήσει νέες βάσεις στις σχέσεις του ανθρώπου με τα ρομπότ, αντιμετωπίζοντας τα ως εργαλεία που θα τον βοηθούν ακόμα και θα τον αποφορτίσουν από επαναλαμβανόμενες ενέργειες όταν υπάρχει ανάγκη. Πλέον ο άνθρωπος έχει τη δυνατότητα να αλληλοεπιδράσει μέσω της φυσικής γλώσσας με τα ρομπότ. Αυτή η πιο άμεση επαφή καταλύει την ψυχρή έως τώρα αλληλεπίδραση μέσω κουμπιών και μοχλών ακόμα και μέσω κώδικα και εισάγει τη μορφή αλληλεπίδρασης που έχει συνηθίσει ο άνθρωπος να χρησιμοποιεί.

Η ευρεία εξάπλωση των καινοτομιών της ρομποτικής δεν θα μπορούσε να μην συνεπάγεται με επιπτώσεις στο μέλλον της εργασίας. Σύμφωνα με πληροφορίες του βρετανικού γραφείου Oxford Economics [16], οι θέσεις εργασίας που θα επηρεαστούν περισσότερο είναι αυτές που απαιτούν τα λιγότερα προσόντα. Δεν είναι λίγοι αναλυτές που υποστηρίζουν ότι αν η αυτοματοποίηση έχει συμβάλει στη δημιουργία περισσότερων θέσεων εργασίας από αυτές που καταστρέφονται, τότε αυτή δημιούργησε επίσης ένα ρήγμα ανάμεσα στις εργασίες υψηλότερης και χαμηλότερης εξειδίκευσης, αφήνοντας πολλούς εργαζόμενους στο περιθώριο. Από την άλλη πλευρά, οι δουλειές που απαιτούν ενσυναίσθηση, δημιουργικότητα ή κοινωνική ευφυΐα είναι πιθανό να συνεχίσουν να εκτελούνται αποκλειστικά από τους ανθρώπους. Παράλληλα, δεν θα λείπουν οι δυσκολίες από μεγάλο πλήθος ατόμων να εμπιστευτούν και να συνεργαστούν με τα ρομπότ. Αυτό πηγάζει από τον φόβο για το άγνωστο, το μη ελέγξιμο και την απρόσωπη φύση της αλληλεπίδρασης [17], [18].

1.6 Αντικείμενο Διπλωματικής Εργασίας

Η παρούσα διπλωματική εργασία, αφορά την ανάπτυξη ενός ρομποτικού βραχίονα φωνητικής αλληλεπίδρασης, με στόχο την ανίχνευση και αναγνώριση αντικειμένων, την συλλογή και παράδοση αυτών στον χρήστη. Αυτό το εγχείρημα έχει ως στόχο να συνδυάσει την όραση μηχανής και τη φωνητική αλληλεπίδραση με την ανάπτυξη ενός ρομποτικού συστήματος που θα ενεργεί βάση των εντολών που θα δέχεται από τον χρήστη. Η ιδιαιτερότητα έγκειται αφενός στην άμεση και φιλική προς στο χρήστη επικοινωνία και αφετέρου στο ευρύτατο πλήθος αναγνώρισης αντικειμένων που δύναται να εντοπίσει.

Το ρομποτικό αυτό σύστημα, βασίζεται στον ρομποτικό βραχίονα Niryo-One με την προσθήκη USB κάμερας (Logitech C922) στη περιοχή του τελικού εργαλείου δράσης (end effector) η οποία βρίσκεται τοποθετημένη σε βάση που έχει σχεδιαστεί ειδικά για αυτή. Προκειμένου να υλοποιηθούν οι στόχοι που έχουν τεθεί για το εν λόγω σύστημα, αναπτύχθηκε κατάλληλο λογισμικό το οποίο υποστηρίζει την αναγνώριση του σχήματος και χρώματος των αντικειμένων καθώς και την αναγνώριση του είδους για όσα αντικείμενα έχει εκπαιδευτεί το σύστημα. Ταυτόχρονα είναι δυνατός ο εντοπισμός και η αναγνώριση QR code. Με τη λειτουργία ανίχνευσης των QR code, επιτυγχάνεται κατά πρώτων η διεύρυνση του πλήθους των αντικειμένων που μπορούν να εντοπιστούν και κατά δεύτερον η προσαρμοστικότητα του εντοπισμού και της αναγνώρισης στις ανάγκες του χρήστη. Το μόνο που απαιτείται είναι ο χρήστης να εκτυπώσει τα QR code και να τα τοποθετήσει στα αντικείμενα που επιθυμεί. Η θέση όλων των αντικειμένων εκφράζεται με συντεταγμένες στον x, y άξονα που πρόκειται για τις σχετικές θέσεις στον ωφέλιμο χώρο του ρομπότ σε σχέση με την θέση της κάμερας. Κατά την έναρξη λειτουργίας, ο χρήστης ενημερώνεται μέσω ενός ηχητικού μηνύματος από τον ρομποτικό βραχίονα για τις διαθέσιμες γλώσσες που μπορεί να επιλέξει για τη συνέχεια της επικοινωνίας. Προς το παρόν υποστηρίζεται η Ελληνική και η Αγγλική. Αφού επιλεχθεί η γλώσσα, το σύστημα ενημερώνει το χρήστη για την διαθεσιμότητα του και περιμένει την επιλογή λειτουργίας που θα του αναθέσει για εκτέλεση ο χρήστης.

Υπάρχουν διαθέσιμες δύο επιλογές λειτουργίας. Στη πρώτη ο χρήστης μπορεί να ζητήσει απευθείας να του παραδοθεί ένα αντικείμενο που επιθυμεί και στη δεύτερη το ρομποτικό σύστημα ελέγχει τον ωφέλιμο χώρο για να εντοπίσει τα διαθέσιμα αντικείμενα που υπάρχουν να τα κατονομάσει στον χρήστη και έπειτα ο χρήστης να επιλέξει ποιο ή ποια επιθυμεί να του παραδοθούν. Κατά τη διαδικασία προσέγγισης και ανασήκωσης του αντικειμένου έχει προηγηθεί ο υπολογισμός της θέσης, της κλίσης και της γωνίας στην οποία βρίσκεται ώστε να επιτευχθεί η βέλτιστη θέση του επενεργητή πέρατος για την ολοκλήρωση της ενέργειας.

Αξίζει να αναφερθεί ότι, ο ρομποτικός βραχίονας αρχικά θα προσπαθήσει να απελευθερώσει το αντικείμενο στο χέρι του χρήστη, εάν εντοπιστεί, αλλιώς θα το παραδώσει σε ένα προκαθορισμένο σημείο που έχει οριστεί. Μετά το τέλος της παράδοσης ο χρήστης έχει τη δυνατότητα είτε να προχωρήσει στην επόμενη ενέργεια είτε να ζητήσει από το ρομποτικό σύστημα να απενεργοποιηθεί.

Το σύστημα έχει αναπτυχθεί ώστε να λειτουργεί είτε με χρήση σταθερού ή φορητού υπολογιστή είτε με Raspberry Pi 4. Η κύρια διαφορά των υπολογιστικών συστημάτων που χρησιμοποιούνται βρίσκεται στην επεξεργαστική ισχύ τους και κατ' επέκταση στο χρόνο εντοπισμού και απόκρισης.

1.7 Κίνητρο Διεξαγωγής Εργασίας

Λαμβάνοντας υπόψιν το βαθμό που επηρεάζει η είσοδος της ρομποτικής στη σύγχρονη κοινωνία, γίνεται αντιληπτή η σημαντικότητα εξέλιξης και κατά προέκταση η ενασχόληση με αυτόν τον κλάδο. Αναμφισβήτητα, αποτελεί πρόκληση η μελέτη ενός συστήματος που μπορεί να συνδυάσει πολλές από τις τεχνολογίες αιχμής και να βοηθήσει τον άνθρωπο αποτελεσματικά στην καθημερινότητα του.

Παράλληλα με αυτό, οι ποικίλες εφαρμογές που μπορεί να έχει το παρόν ρομποτικό σύστημα ως ρομπότ σταθερής βάσης αποτέλεσε κίνητρο και πηγή έμπνευσης για την ανάπτυξη των χαρακτηριστικών και ιδιοτήτων του. Ενδεικτικά μπορούν να αναφερθούν ορισμένοι τομείς που θα φαινόταν χρήσιμος ο ρομποτικός βραχίονας.

Αρχικά, στον τομέα της Υγείας τόσο στο κομμάτι των χειρουργικών επεμβάσεων όσο και στη περίθαλψη ασθενών. Γνωρίζουμε ότι, η χειρουργική ομάδα κατά τη διαδικασία επεμβάσεων είναι προσηλωμένη στο έργο της και ιδιαίτερα ο υπεύθυνος ιατρός χειρουργός. Η εφαρμογή του ρομποτικού βραχίονα σε επικουρική θέση θα μπορούσε να βοηθήσει στην ορθή μεταβίβαση των χειρουργικών εργαλείων στα μέλη της ομάδας όπου απρόσκοπτα θα συνεχίζουν την δουλειά τους. Όσο αναφορά τη περίθαλψη, η τοποθέτηση του ρομποτικού βραχίονα στις κλίνες των ασθενών μπορεί να βοηθήσει στην παροχή των φαρμάκων και αντικειμένων στον ασθενή χωρίς την παρουσία νοσηλευτικού προσωπικού. Να επισημανθεί ότι τα ρομπότ δεν μπορούν επ' ουδενί να συγκριθούν με τους πραγματικούς φροντιστές και δεν είναι σε θέση να τους αντικαταστήσουν στο εγγύς μέλλον.

Στον τομέα της βιομηχανίας, μία ενδεχόμενη χρήση του ρομποτικού βραχίονα θα ήταν η τοποθέτηση του σε γραμμή παραγωγής όπου θα δύναται η λειτουργία διαλογής βάση σχήματος χρώματος ή ακόμα και ετικέτας QR code και τοποθέτησης των προϊόντων σε συσκευασίες. Οι ίδιες λειτουργίες θα μπορούσαν να φανούν χρήσιμες και σε χώρους όπως αποθήκες τροφίμων και βιβλιοθήκες όπου υπάρχει ανάγκη σε ταξινόμηση και οργάνωση αντικειμένων, διαδικασίες οι οποίες σε μεγάλο βαθμό πραγματοποιούνται από τον άνθρωπο.

Σημαντική είναι η βοήθεια που μπορεί να προσφέρει στο οικιακό περιβάλλον. Ορισμένες από τις καθημερινές εργασίες του νοικοκυριού μπορούν να εκτελεστούν σε συνεργασία με τον ρομποτικό βραχίονα όπως είναι το μαγείρεμα, η ταξινόμηση των πιάτων και των ρούχων κ.α.

Ιδιαίτερη σημασίας είναι η βοήθεια που μπορεί να παρέχει σε ΑμΕΑ και ηλικιωμένους. Ο στόχος είναι τα άτομα αυτά να μπορούν να υπάρξουν αυτόνομα χωρίς την ανάγκη για συνεχόμενη παρουσία φροντιστή. Με τη βοήθεια του ρομπότ, άτομα με προβλήματα όρασης ή κινητικά προβλήματα θα έχουν τη δυνατότητα μέσω των φωνητικών εντολών να ζητούν τη μεταβίβαση αντικείμενων και την εκτέλεση ενεργειών.

Το μεγαλύτερο κίνητρο στη σύλληψη της ιδέας και τελικά στην υλοποίηση της αποτέλεσε η επιθυμία μου να εμβαθύνω και να εμπλουτίσω τις γνώσεις μου στους τομείς της μηχανικής όρασης και της ρομποτικής. Η ανάπτυξη λογισμικού που μπορεί να δώσει την αίσθηση της όρασης σε ένα μηχάνημα με τρόπο τέτοιο ώστε να βοηθά χρήστες διαφόρων αναγκών και συνθηκών αποτέλεσε απαρχή ζήλου για τη βέλτιστη λειτουργία.

1.8 Συμβολή στην επιστημονική κοινότητα

Στο πλαίσιο εκπόνησης της διπλωματικής εργασίας, πραγματοποιήθηκε ερευνά που ενώνει σημαντικά κομμάτια της σύγχρονης ρομποτικής και τεχνητής νοημοσύνης. Συγκεντρώθηκαν οι απαραίτητες πληροφορίες και το τελικό αποτέλεσμα είναι ένας συνδυασμός αυτών με τεχνικές προγραμματισμού και ρομποτικής. Σημαντικό γνώρισμα αποτελεί η παρουσίαση ενός ολοκληρωμένου ρομποτικού συστήματος που βασίζεται στον ρομποτικό βραχίονα Niryo-One και έχει ως επίκεντρο την προσαρμοστικότητα και ευχρηστία στη παροχή βοήθειας στο χρήστη. Λόγω και της πρόσφατης κυκλοφορίας του Niryo-One δεν υπάρχουν γνωστές αντίστοιχες υλοποιήσεις με αυτή της παρούσας διπλωματικής. Παράλληλα, ο ταυτόχρονος συνδυασμός των λειτουργιών που υλοποιούνται στο εγχείρημα καθώς και η τεχνική που αναπτύχθηκε για την εύρεση εκτός των άλλων και της γωνίας στα αντικείμενα που έχουν αναγνωριστεί ως προς το είδος, αποτελούν καινοτομίες και συνεισφέρουν στις ήδη υπάρχουσες μεθόδους που εφαρμόζονται.

1.9 Παρόμοια Ερευνητικά Έργα

Με το πέρασμα των χρόνων παρατηρείται η σταδιακή αύξηση των κινήτρων ενασχόλησης με τους τομείς της μηχανικής όρασης και ρομποτικής για νέους επιστήμονες και ερευνητές [19]. Αυτό έχει ως αποτέλεσμα τη δημιουργία νέων εφαρμογών και τεχνικών που δίνουν λύση σε διάφορα ζητήματα που απασχολούν την επιστημονική κοινότητα. Μέχρι σήμερα, υπάρχουν αρκετά παραδείγματα εφαρμογών που έχουν αναπτυχθεί για την δημιουργία ρομποτικού βραχίονα αναγνώρισης και μεταφοράς αντικειμένων με φωνητική αλληλεπίδραση.

Η ρομποτική εφαρμογή που παρουσιάζεται στο [20] βασίζεται σε ρομποτικό βραχίονα 4 βαθμών ελευθερίας και η αναγνώριση επιτυγχάνεται για συγκεκριμένα αντικείμενα βάση των οποίων έχει γίνει η εκπαίδευση του μοντέλου. Για να επιτευχθεί αυτό έχει γίνει χρήση της αρχιτεκτονικής δικτύου προτάσεων περιοχής, Regions Proposal Network RPN με τη χρήση του R-CNN inception v2 μέσω του TensorFlow framework. Παράλληλα, η επικοινωνία με το χρήστη γίνεται μέσω φωνητικών εντολών και γραφικού περιβάλλοντος. Παρόλο που η συγκεκριμένη προσέγγιση επίλυσης του προβλήματος δίνει στο χρήστη την δυνατότητα αναγνώρισης στα αντικείμενα που χρησιμοποιεί πιο συχνά, η διαδικασία εκπαίδευσης, το ποσοστό επιτυχημένης αναγνώρισης, ο περιορισμένος ωφέλιμος χώρος καθώς και η τοποθέτηση της κάμερας στον περιβάλλοντα χώρο του ρομποτικού βραχίονα, καθιστούν το σύστημα δύσχρηστο για καθημερινή χρήση λόγω της αδυναμίας εύκολης προσαρμοστικότητας στις άμεσες αλλαγές αναγκών του χρήστη και στα διαφορετικά περιβάλλοντα χρήσης.

Ο Pulokottil κ.α. [21] προτείνουν ένα ρομποτικό σύστημα το οποίο απευθύνεται κυρίως σε ασθενείς με κινητικά προβλήματα και με προβλήματα όρασης. Πιο συγκεκριμένα, αναπτύχθηκε ρομποτική εφαρμογή στο ROS framework, η οποία βασίζεται στον βραχίονα JACO² και δεν υποστηρίζει όραση μηχανής διότι δίνει τη δυνατότητα να χειριστεί είτε με φωνητικές εντολές είτε με joystick. Η επικοινωνία με τον χρήστη γίνεται με χρήση του CMU Pocketsphinx API. Το εν λόγω εγχείρημα δοκιμάστηκε από 2 ασθενείς εθελοντές και βαθμολογήθηκε στα διαφορά τεστ που έγιναν. Βάση των αποτελεσμάτων, οι χρήστες θα χρησιμοποιούσαν το ρομποτικό σύστημα περισσότερο για κινήσεις ανασήκωσης και μετακίνησης αντικειμένων και προτίμησαν την τοποθέτηση του τόσο σε σταθερό σημείο όσο και την προσάρτηση του σε αναπηρικό καροτσάκι. Ωστόσο, για να καταφέρει ο χρήστης να εκτελέσει μια ενέργεια απαιτείται εξάσκηση καθώς θα πρέπει να καθοδηγεί βήμα προς βήμα τις κινήσεις που θα εκτελέσει ο βραχίονας, ενώ παράλληλα να εκτιμήσει την απόσταση του επενεργητή πέρατος με το αντικείμενο ώστε να δώσει την κατάλληλη φωνητική εντολή 'long' ή 'small' στο σύστημα. Έτσι, προκύπτουν πρακτικά ζητήματα δυσχρηστίας της εφαρμογής τόσο στον χειρισμό όσο και στην καθοδήγηση με τον χρήστη όπου θα πρέπει να έχει συνέχεια στραμμένη την προσοχή του στο περιβάλλον κίνησης του βραχίονα.

Στην δημοσίευση του Wang κ.α. [22] παρουσιάζεται ρομποτικός βραχίονας 6 βαθμών ελευθερίας ο οποίος βασίζεται στη χρήση μικροελεγκτή Arduino, ultrasonic αισθητήρα στη βάση του βραχίονα και κάμερας προσαρμοσμένης στη περιοχή του τελικού σημείου δράσης. Όσο αναφορά την όραση μηχανής, έχει αναπτυχθεί αλγόριθμος που εκτελείται στο Raspberry Pi για να αναγνωρίζει το σχήμα των αντικειμένων ενώ ταυτόχρονα διαχειρίζεται και την φωνητική λειτουργία που επιτυγχάνεται μέσω του Amazon Echo dot. Ιδιαίτερο ενδιαφέρον έχει ο τρόπος υπολογισμού των γωνιών κάθε άρθρωσης μιας και υπολογίζονται από νευρωνικό δίκτυο που προσομοιώθηκε στο λογισμικό MATLAB. Το δίκτυο έχει ως είσοδο την απόσταση του ρομπότ από το αντικείμενο μέσω του ultrasonic αισθητήρα και ο υπολογισμός των μοιρών περιστροφής κάθε άρθρωσης προκύπτουν με χρήση των γνωστών συναρτήσεων του αντίστροφου κινηματικού μοντέλου. Αν και τα αποτελέσματα που παρουσιάζονται φαίνεται να προσεγγίζουν τα αντικείμενα με ακρίβεια, προκύπτουν κάποια πρακτικά ζητήματα που δεν μπορούν να επιλυθούν όπως για παράδειγμα εάν το αντικείμενο που ζητηθεί να εντοπιστεί βρίσκεται πίσω από κάποιο άλλο τότε η βασική είσοδος για τον νευρωνικό δίκτυο που είναι η απόσταση από τον ultrasonic sensor θα είναι εσφαλμένη και θα οδηγήσει αναπόφευκτα σε λανθασμένο υπολογισμό των μοιρών.

Ο Tasevski κ.α. [23] παρουσιάζουν σύστημα αυτόματης αναγνώρισης ομιλίας (ASR) και σύστημα όρασης μηχανής. Τα συστήματα εφαρμόζονται στον βιομηχανικό βραχίονα ABB IRB140. Το ρομποτικό σύστημα, που εμπεριέχει τα δυο συστήματα, έχει τη δυνατότητα χειρισμού αντικειμένων τυχαία τοποθετημένων σε μια συγκεκριμένη επιφάνεια, κάνοντας αναγνώριση του σχήματος των διαστάσεων του χρώματος της θέσης και του προσανατολισμού τους. Η αναγνώριση ομιλίας βασίζεται στο σύστημα AlfaNum ASR το οποίο είναι λειτουργικό μόνο για την Σέρβικη γλώσσα ενώ το σύστημα φωνητικής απάντησης κάνει χρήση του anReader (AlfaNum) για τη Σερβική γλώσσα και της Microsoft Speech Engine για την Αγγλική. Όσο αφορά το σύστημα μηχανικής όρασης, αναπτύχθηκε σε γλώσσα προγραμματισμού C++ με χρήση της βιβλιοθήκης OpenCV 2.3.1 και χρησιμοποιήθηκαν 2 κάμερες AXIS 211 σε στερεοσκοπική λειτουργία. Το σύστημα παρόλο που φαίνεται να καλύπτει τους

απαραίτητους τομείς μιας ολοκληρωμένης λύσης εμφανίζει δυσκολίες, καθώς η επιφάνεια στην οποία βρίσκονται τοποθετημένα τα αντικείμενα είναι συγκεκριμένου μεγέθους (A4) και θα πρέπει να παραμένει ακίνητη καθ' όλη τη διάρκεια της λειτουργίας. Παράλληλα, η τοποθέτηση και λειτουργία του συστήματος απαιτεί την ρύθμιση των καμερών και του ρομποτικού βραχίονα στο χώρο κάτι το οποίο δεν μπορεί να πραγματοποιηθεί από το μέσο χρήστη λόγω αυξημένης πολυπλοκότητας.

Ο Bandara κ.α. [24] παρουσιάζουν ένα ρομποτικό σύστημα βοηθού με δυνατότητα εκτός της φωνητικής λειτουργίας και αναγνώρισης χειρονομιών. Το εν λόγω σύστημα, εκτελείται από βραχίονα 6 βαθμών ελευθέριας και έχει τις δυνατότητες ανασήκωσης και τοποθέτησης αντικειμένων στο χώρο. Η αναγνώριση των σχημάτων και των χρωμάτων γίνεται με Kinect V2 αισθητήρα σε συνεργασία με την εφαρμογή που αναπτύχθηκε σε C#.Net με αλγόριθμους από Aforge.Net. Για την φωνητική λειτουργία χρησιμοποιήθηκε η πλατφόρμα Microsoft Speech αναγνωρίζοντας την Αγγλική γλώσσα. Οι εντολές ακολουθούν συγκεκριμένο λεξικό και ο υπολογισμός των γωνιών των αρθρώσεων προκύπτει με χρήση συναρτήσεων αντίστροφου κινηματικού μοντέλου που υλοποιείται στη MATLAB. Σημαντικό μειονέκτημα βάσει των αποτελεσμάτων θεωρείται το ποσοστό αστοχίας εκτέλεσης των ενεργειών το οποίο προκύπτει από λάθη συντεταγμένων σε κάθε άξονα καθώς λείπει η δυνατότητα ανάδρασης από τα μοτέρ των αρθρώσεων.

Το έργο που παρουσιάζεται στην παρούσα διπλωματική εργασία διαφέρει από τα προαναφερθέντα καθώς συνδυάζει τρία διαφορετικά είδη αναγνώρισης αντικειμένων: 1. Σχήματος χρώματος 2. Είδους 3. QR code ετικετών και δίνει μεγάλη ευελιξία στα αντικείμενα που μπορούν να αναγνωριστούν. Παράλληλα, η επικοινωνία επιτυγχάνεται αμφίδρομα με υποστήριξη και της Αγγλικής και της Ελληνικής γλώσσας και η διαδικασία εγκατάστασης και λειτουργίας δεν απαιτεί ρυθμίσεις από το χρήστη. Επιπλέον, υπάρχει η δυνατότητα εκτέλεσης του βασικού προγράμματος αναγνώρισης αντικειμένων σε Raspberry Pi 4 αποφεύγοντας έτσι τη χρήση Η/Υ και καθιστώντας εύκολη τη μετακίνηση του.

1.10 Περιγραφή Κεφαλαίων

Το κείμενο διαρθρώνεται σε πέντε κεφάλαια, με κάθε ένα από αυτά να συμβάλει στην επεξήγηση και ανάλυση των επιμέρους τμημάτων που συγκρότησαν στην υλοποίηση του ρομποτικού συστήματος. Στα κεφάλαια αυτά γίνεται αναλυτική παρουσίαση, τόσο του υλικού μέρους, όσο και του λογισμικού που αναπτύχθηκε.

Αρχικά, στο πρώτο κεφάλαιο αναπτύσσονται γενικές έννοιες περί της ρομποτικής καθώς και το αντικείμενο της διπλωματικής εργασίας.

Στο δεύτερο κεφάλαιο, παρουσιάζεται το θεωρητικό υπόβαθρο των εννοιών και των μεθοδολογιών του λογισμικού και υλικού μέρους που χρησιμοποιήθηκαν. Επιπρόσθετα, παρουσιάζονται τα εργαλεία ανάπτυξης του λογισμικού και η γλώσσα προγραμματισμού.

Ακολούθως, το τρίτο κεφάλαιο αναφέρεται στο τεχνικό κομμάτι της υλοποίησης με λεπτομερή ανάλυση του τρόπου ενσωμάτωσης όλων των μεθοδολογιών και του τρόπου λειτουργίας του ρομποτικού συστήματος.

Στο τέταρτο κεφάλαιο παρουσιάζονται τα πειραματικά σενάρια που εκτελέστηκαν και τα αποτελέσματα που προέκυψαν.

Τέλος, στο πέμπτο κεφάλαιο συνοψίζονται όλα τα στοιχεία που έχουν παρουσιάσει στην διπλωματική εργασία, τα προβλήματα που προέκυψαν κατά την ανάπτυξη του συστήματος, τα τελικά συμπεράσματα και οι μελλοντικές επεκτάσεις.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

Σε αυτό το κεφάλαιο σημειώνονται οι αλγόριθμοι, οι τεχνικές, οι μεθοδολογίες και τα μοντέλα που έχουν χρησιμοποιηθεί στη διπλωματική και είναι σημαντική η παρουσίαση τους στον αναγνώστη πριν από την ανάλυση και σχεδίαση του συστήματος.

Να επισημανθεί ότι οι παρακάτω μηχανισμοί που θα αναλυθούν, δεν αποτελούν πρότυπη δουλειά και έχουν προταθεί από άλλους.

2.1 Προγραμματισμός Όρασης Μηχανής

2.1.1 Python

Η Python είναι μια διερμηνευόμενη, υψηλού επιπέδου, γενικού σκοπού και με δυναμική σημασιολογία (semantics) γλώσσα προγραμματισμού [25]. Ανήκει στην κατηγορία γλωσσών προστακτικού προγραμματισμού και υποστηρίζει τόσο την αντικειμενοστρέφεια όσο και το δομημένο προγραμματισμό. Η Python υποστηρίζει αρθρώματα (modules) και πακέτα (packages), τα οποία βοηθούν στην επαναχρησιμοποίηση και την τμηματοποίηση κώδικα ενθαρρύνοντας την αναγνωσιμότητα και την εύκολη συντήρηση του. Ένα από τα πολλά υποσχόμενα οφέλη της Python είναι ότι τόσο η κύρια βιβλιοθήκη (standard library) όσο και ο διερμηνέας διατίθενται δωρεάν, σε δυαδική και πηγαίου κώδικα μορφή. Δεν υπάρχει καμία αποκλειστικότητα, καθώς η Python και όλα τα απαραίτητα εργαλεία είναι διαθέσιμα σε όλες τις μεγάλες πλατφόρμες.

Συγκεντρωτικά ανάμεσα στα κύρια χαρακτηριστικά της είναι:

- Ευκολία
 - Εκμάθησης
 - Αναγνωσιμότητας
 - Συντήρησης
- Γρήγορη Ανάπτυξη Εφαρμογών
- Διερμηνευόμενη
- Πολύ υψηλού επιπέδου δομές δεδομένων
- Επεκτασιμότητα
- Ανοιχτού Κώδικα
- Ευρεία χρήση
- Αυτόματη διαχείριση μνήμης

Με βάση τις παραπάνω ιδιότητες της Python, προκύπτει πως συγκριτικά με γλώσσες χαμηλότερου επιπέδου όπως C, C++ να καθίσταται ταχύτερη στην ανάπτυξη εφαρμογών καθώς και παράγει 3-5 φορές μικρότερα σε μέγεθος εκτελέσιμα αρχεία σε σχέση με τα αντίστοιχα σε Java. Επιπλέον, διατίθενται βιβλιοθήκες που δύναται να ενσωματώσουν τμήματα κώδικα από C και C++ καθώς και μεγάλο πλήθος έτοιμων βιβλιοθηκών εύκολα διαχειρίσιμων.

Παρά το γεγονός ότι η Python υπερτερεί σε πολλά σημεία, μειονέκτημα αποτελεί ο χρόνος εκτέλεσης των προγραμμάτων σε σχέση με μεταγλωττιζόμενες γλώσσες όπως C και C++. Αυτό έγκειται στο γεγονός ότι τα προγράμματα σε Python δεν εκτελούνται απευθείας στον επεξεργαστή καθώς δεν έχουν πρωτίστως μεταγλωττιστεί σε δυαδικό κώδικα. Ωστόσο, η ταχύτητα ανάπτυξης εφαρμογών που προσφέρει η Python αντισταθμίζει την παραπάνω χρονική καθυστέρηση.

Για τη ανάπτυξη του απαιτούμενου προγράμματος στη παρούσα υλοποίηση επιλέχθηκε η γλώσσα Python 3.7. Οι λόγοι που συνηγόρησαν σε αυτή την επιλογή είναι:

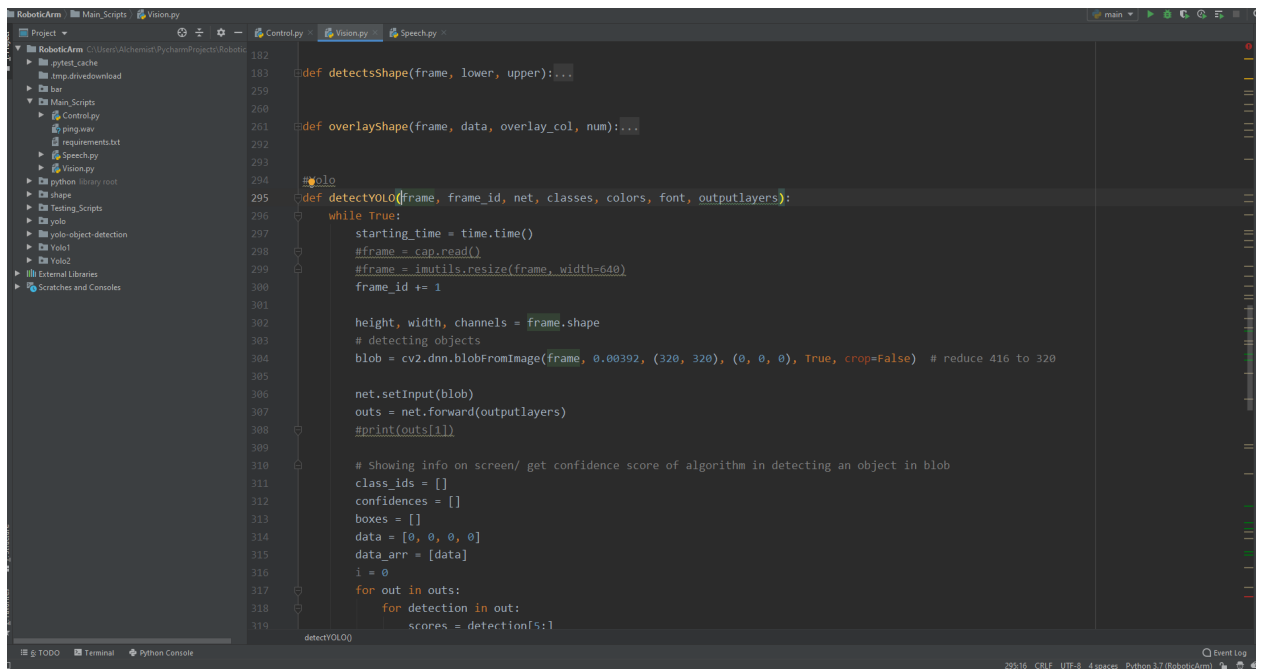
- Η πληθώρα έτοιμων πλαισίων λογισμικού πάνω σε μηχανική όραση και τεχνητή νοημοσύνη (πχ OpenCV, Darknet, κ.λπ.) για την υλοποίηση όλων των επιμέρους λειτουργιών
- Η συμβατότητα που προσφέρει στην εκτέλεση στη πλακέτα Raspberry Pi
- Όλα τα προαναφερθέντα θετικά χαρακτηριστικά και δυνατότητες που προσφέρει

2.1.2 PyCharm (IDE)

Το PyCharm [26] είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) που χρησιμοποιείται για την ανάπτυξη προγραμμάτων σε γλώσσα Python. Το εν λόγω IDE δημιουργήθηκε από τη Τσέχικη εταιρεία JetBrains και κυκλοφόρησε τον Ιούλιο του 2010. Είναι διαθέσιμο σε δύο εκδόσεις την επαγγελματική όπου είναι επί πληρωμή και την κοινοτική όπου είναι δωρεάν προσφέροντας όμως λιγότερες λειτουργίες σε σχέση με την επαγγελματική. Επίσης, αξίζει να αναφερθεί ότι το PyCharm μπορεί να εγκατασταθεί σε διαφορετικά λειτουργικά συστήματα όπως είναι τα Windows, το Linux αλλά και το Mac OS. Ορισμένες από τις δυνατότητες, που παρέχει το PyCharm IDE (με γραφικό περιβάλλον) είναι:

- Η δυνατότητα έξυπνης ολοκλήρωσης κώδικα, επιθεώρησης κώδικα, επισήμανση σφαλμάτων, γρήγορες επιδιορθώσεις, και αυτοματοποιημένες αναδιαμορφώσεις κατά τη διαδικασία ανάπτυξης
- Το ενσωματωμένο πρόγραμμα εντοπισμού σφαλμάτων Python Debugger με λειτουργία επιθεώρησης κώδικα βήμα προς βήμα
- Ευέλικτες δυνατότητες πλοήγησης έργου και κώδικα χρησιμοποιώντας εξειδικευμένες προβολές έργου, προβολές δομής αρχείων και γρήγορο μετακίνηση μεταξύ αρχείων, και κλάσεων

- Python refactoring : περιλαμβάνει ασφαλής μετονομασία και διαγραφή, μέθοδο εξαγωγής, εισαγωγής μεταβλητής γραμμής και μεθόδου



Εικόνα. 2.1: Το ολοκληρωμένο αναπτυξιακό περιβάλλον Pycharm.

2.1.3 Notepad++

Το Notepad++ [27] είναι ένας δωρεάν επεξεργαστής πηγαίου κώδικα για το λειτουργικό σύστημα των Windows το οποίο αναπτύχθηκε από τον DonHo. Αποτελεί πρόγραμμα ανοιχτού κώδικα βασισμένο στο Scintilla και είναι γραμμένο σε γλώσσα προγραμματισμού C++. Υποστηρίζει ταυτόχρονη επεξεργασία πολλαπλών αρχείων χρησιμοποιώντας καρτέλες και διαθέτει αναπαράσταση σύνταξης και αναδίπλωση κειμένου. Επίσης προσφέρει αυτόματη αλλαγή της μορφοποίησης του κειμένου σύμφωνα με τη γλώσσα προγραμματισμού που χρησιμοποιείται. Ακόμα επιτρέπει την απόκρυψη και την συμπίεση τμημάτων του κώδικα, την αυτόματη ολοκλήρωση κειμένου, την πρόβλεψη λέξεων ή ακόμα και ολόκληρων φράσεων.

2.1.4 Niryo-One Studio

Το Niryo-One Studio [28] είναι η επίσημη εφαρμογή για το έλεγχο του ρομποτικού βραχίονα Niryo-One μέσω Ηλεκτρονικού Υπολογιστή. Διατίθεται δωρεάν και είναι συμβατή με τα λειτουργικά συστήματα Windows, MacOS, Linux. Μέσω αυτής

της εφαρμογής δίνονται αρκετές επιλογές ρύθμισης, ενημέρωσης και προγραμματισμού του βραχίονα.

Μερικές από αυτές είναι:

- Αρχική βαθμονόμηση των μελών του ρομπότ (robot calibration)
- Μετακίνησης του ρομπότ είτε δίνοντας τιμές στους άξονες x, y, z, roll, pitch, yaw είτε για την κλίση των αρθρώσεων.
- Προγραμματισμός του βραχίονα μέσω του Niryo Blocks δημιουργώντας ουσιαστικά κομμάτια (blocks) εντολών ακολουθίας κινήσεων.
- Τρισδιάστατη απεικόνιση της τρέχουσας θέσης του ρομποτικού βραχίονα
- Συνεχής επίβλεψη της κατάστασης των μοτέρ προσφέροντας πληροφορίες για τη θερμοκρασία τους και παράθυρο με logs από την εκτέλεση των εντολών

2.2 Όραση Μηχανής

2.2.1 Ορισμός της έννοιας όραση μηχανής

Μηχανική όραση ή αλλιώς τεχνητή όραση, ονομάζεται ο τομέας της επιστήμης υπολογιστών και της τεχνητής νοημοσύνης ο οποίος έχει ως στόχο να αναπαράγει αλγοριθμικά την αίσθηση της όρασης. Με άλλα λόγια, στόχος αυτού του τομέα είναι ο συνδυασμός λογικής και όρασης όπως ακριβώς συμβαίνει στον άνθρωπο, μέσα από συστήματα λήψης και επεξεργασίας ψηφιακών εικόνων, παράγοντας πληροφορίες. Οι εν λόγω πληροφορίες μπορούν να αντληθούν έπειτα από επεξεργασία φωτογραφιών, βίντεο, πολυδιάστατων εικονικών μοντέλων και σαρωτών.

Η πρόοδος της τεχνολογίας έχει βοηθήσει κατά πολύ αυτό το επιστημονικό πεδίο, το οποίο έχει προσελκύσει τα τελευταία χρόνια όλο και περισσότερους επιστήμονες. Έτσι, οι εφαρμογές και τα συστήματα που προσομοιάζουν τη λειτουργία του ματιού έχουν γνωρίσει άνθιση. Ορισμένα παραδείγματα εφαρμογών τέτοιων συστημάτων είναι τα εξής:

-
- Ανίχνευση σφαλμάτων και ποιοτικός έλεγχος (π.χ. σε μια γραμμή παραγωγής)
 - Ιατρική ανάλυση εικόνας (π.χ. εντοπισμός ενός ιού ή καρκινικών κυττάρων)
 - Επίβλεψη ετικετών σε προϊόντα (π.χ. ένα βιομηχανικό ρομπότ)
 - Έλεγχος και επιτήρηση διαδικασιών (π.χ. ένα αυτόνομο ρομποτικό όχημα)
 - Αλληλεπίδραση χρηστών με υπολογιστικά συστήματα (π.χ. ως είσοδος σε μια συσκευή αλληλεπίδρασης ανθρώπου / μηχανής)
 - Καθοδήγηση ρομποτικών συστημάτων (π.χ. σε βιομηχανικό περιβάλλον ή στην έξυπνη γεωργία)
 - Επιθεώρηση συσκευασιών (π.χ. σε βιομηχανική γραμμή ταξινόμησης και συσκευασίας)

Μέσα από την επεξεργασία των δεδομένων, το σύστημα δύναται να λαμβάνει αποφάσεις και να ενημερώνει τον χρήστη για το αποτέλεσμα. Το είδος της απόφασης μπορεί να είναι κάτι απλό όπως «ο εντοπισμός ενός ατόμου ή το πλήθος των ατόμων» ή να πρόκειται για μια συνθέτη διαδικασία όπως η ανάλυση ιατρικών εξετάσεων. Σε κάθε περίπτωση για την παραγωγή ορθού αποτελέσματος οι εικόνες ή τα καρέ του video επεξεργάζονται με την εφαρμογή διάφορων φίλτρων χρωματικών μετατροπών καθώς και αλγορίθμων τεχνητής νοημοσύνης.

Σε ένα σύστημα μηχανικής όρασης προκειμένου να επεξεργαστούν τα δεδομένων από μια κάμερα απαιτείται από τον υπολογιστή να επεξεργαστεί ένα πλέγμα αριθμών όπως απεικονίζεται στην παρακάτω εικόνα



But the camera sees this:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	74	65
20	41	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

Εικόνα. 2.2: Οι αριθμοί που αντιλαμβάνεται το σύστημα αντιστοιχούν στην χρωματική ένταση των επιμέρους εικονοστοιχείων

Ο τομέας της μηχανικής όρασης μπορεί να χωριστεί σε τρία επίπεδα:

- Όραση χαμηλού επιπέδου: μετά από την λήψη και την επεξεργασία της εικόνας, εξάγονται βασικά χαρακτηριστικά όπως γωνίες, ακμές, χρώματα και άλλα
- Όραση μεσαίου επιπέδου: Σε αυτό το στάδιο χρησιμοποιούνται τα στοιχεία της όρασης χαμηλού επιπέδου με σκοπό την ανάλυση της κίνησης στην εικόνα. Στο επίπεδο αυτό, εάν κριθεί αναγκαίο, εξάγεται τρισδιάστατη κατασκευή της εικόνας αυτής.
- Όραση υψηλού επιπέδου: στο τελικό στάδιο, χρησιμοποιώντας τα δεδομένα της όρασης μεσαίου επιπέδου, εκκινούνται διεργασίες ανάλογες του αποτελέσματος που έχει παραχθεί

Ο τομέας αυτός εξελίσσεται συνεχώς, καθώς συμβαδίζει με τη ραγδαία ανάπτυξη της τεχνολογίας και τις απαιτήσεις της εποχής. Τα επιστημονικά πεδία που συμβάλλουν σήμερα στην ανάπτυξη του τομέα αυτού, είναι Τεχνητή Νοημοσύνη, η Επεξεργασία Εικόνας και η Αναγνώριση Προτύπων καθιστώντας τον αναγκαίο συνδετικό κρίκο μεταξύ πολλών επιστημών.



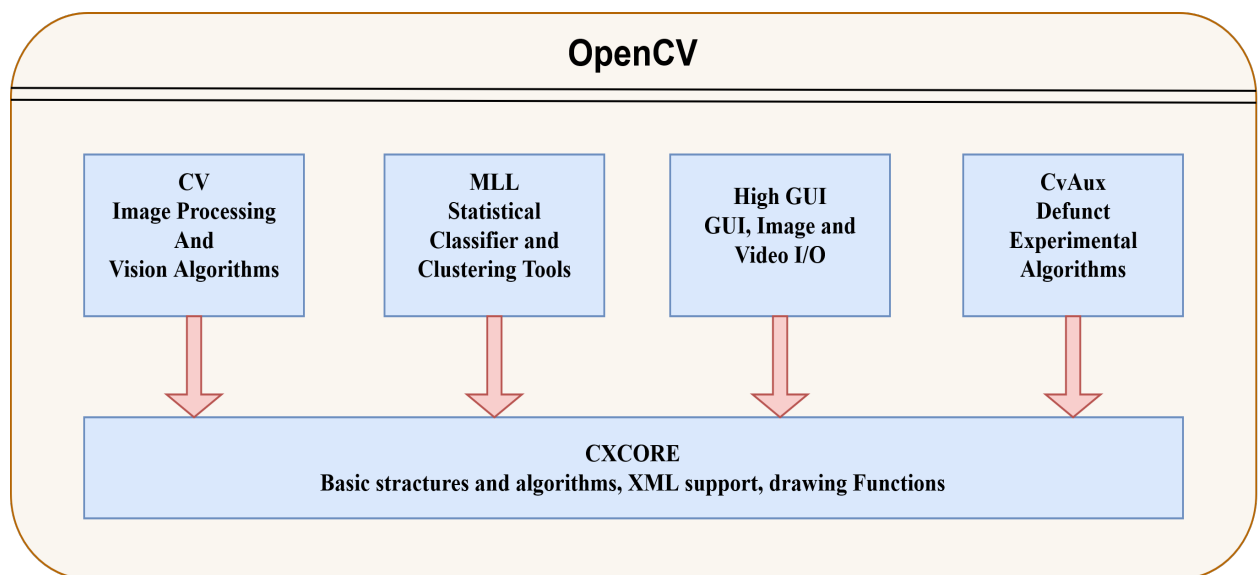
Εικόνα. 2.3: Σύνδεση του πεδίου της όρασης μηχανής με άλλα επιστημονικά πεδία

2.2.2 Βιβλιοθήκη OpenCV

Η OpenCV [29] είναι μια βιβλιοθήκη ανοικτού λογισμικού που σχεδιάστηκε για εφαρμογές μηχανικής όρασης (Computer Vision) η οποία αναπτύχθηκε αρχικά από την Intel. Έχει διεπαφές C, C++, Python, Java και υποστηρίζει Windows, Linux, Mac OS, iOS, και Android. Η βιβλιοθήκη είναι ελεύθερη για εμπορική και ερευνητική χρήση υπό την άδεια ανοικτού λογισμικού BSD. Αυτό σημαίνει ότι μπορεί να γίνει ελεύθερη χρήση όλης η μέρους της βιβλιοθήκης για την ανάπτυξη ενός εμπορικού ή ερευνητικού προϊόντος από τον οποιοδήποτε.

Η γλώσσα στην οποία είναι γραμμένη είναι η C και είναι ανεξάρτητη πλατφόρμας. Εμπεριέχει περισσότερες από 500 συναρτήσεις σε C και κλάσεις σε C++ οι οποίες

καλύπτουν πολλές διαφορετικές εφαρμογές στους τομείς της οπτικής επεξεργασίας βίντεο και εικόνας, όπως ιατρικές εφαρμογές, ασφάλειας, κλειστού κυκλώματος παρακολούθησης, ρομποτική, ρύθμισης κάμερας, διεπαφή χρήστη και υπολογιστή. Μιας και η τεχνητή όραση μαζί με την εκπαίδευση μηχανής τα τελευταία χρόνια είναι αλληλένδετες δεν θα γινόταν να μην υπήρχε και η παρουσία της δεύτερης στις διαθέσιμες βιβλιοθήκες που παρέχει η OpenCV. Η γενικής χρήσης βιβλιοθήκη εκπαίδευσης μηχανής (MLL, Machine Learning Library) στοχεύει στην αναγνώριση προτύπων μέσω στατιστικών υπολογισμών και ομαδοποίησης. Η MLL είναι πολύ χρήσιμη για έργα όρασης τα οποία αποτελούν τη βασική αποστολή του OpenCV, αλλά γενικά χρησιμοποιείται για οποιοδήποτε πρόβλημα εκπαίδευσης μηχανής. Στόχοι της βιβλιοθήκης είναι η παροχή απλών τεχνικών τεχνητής όρασης και η αποτελεσματική εκμετάλλευση των υπολογιστικών πόρων με έμφαση στις εφαρμογές πραγματικού χρόνου. Η δομή της OpenCV αποτελείται κυρίως από 5 συστατικά μέρη όπως παρουσιάζονται στην εικόνα 2.4



Εικόνα. 2.4: Σύνδεση του πεδίου της όρασης μηχανής με άλλα επιστημονικά πεδία

Αναλύοντας τα επιμέρους τμήματα που συνθέτουν τη βιβλιοθήκη OpenCV, το πρώτο δομικό μέρος με την ονομασία CV, αναπαριστά τους βασικούς αλγορίθμους για επεξεργασία εικόνας καθώς και υψηλού προγραμματιστικού επιπέδου για μηχανική όραση. Το δεύτερο, MLL αναπαριστά τη βιβλιοθήκη μηχανικής μάθησης που περιλαμβάνει εργαλεία για ομαδοποίηση και πολλούς στατικούς ταξινομητές. Το τρίτο, HighGUI, αναπαριστά τις ρουτίνες εισόδου-εξόδου συμπεριλαμβανομένου της

διεπαφής χρήστη-εφαρμογής και λειτουργίες για αποθήκευση και φόρτωση βίντεο και εικόνας. Το τέταρτο δομικό μέρος της OpenCV είναι το CvAux που περιέχει αλγορίθμους που είναι σε πειραματικό στάδιο ή έχουν εγκαταλειφθεί. Η απόληξη CXCORE περιέχει τις βασικές δομές δεδομένων, όπως το IplImage, τη βασική δομή στην OpenCV που περιγράφει το κάθε καρέ προς επεξεργασία.

Οι μεθοδολογίες που αναπτύχθηκαν στο παρακάτω κεφάλαιο έγιναν με την χρήση της έκδοσης 4.1.2 της βιβλιοθήκης.

2.2.3 Λογισμικό ZBar

Το ZBar [30] αποτελεί μια σουίτα λογισμικού ανοιχτού κώδικα η οποία σχεδιάστηκε για την ανάγνωση ετικετών γραμμωτού κώδικα (barcode) από διάφορες πηγές εισόδου όπως από ροή video, αρχεία εικόνας και από αισθητήρες έντασης φωτός. Υποστηρίζει πολλούς δημοφιλείς τύπους γραμμικών κωδικών μεταξύ των οποίων είναι EAN-13 / UPC-A, UPC-E, EAN-8, Code 128, Code 39, Interleaved 2 of 5 και QR Code.

Η σουίτα ZBar προσφέρεται είτε ως αυτόνομη εφαρμογή με γραφικό περιβάλλον και υποστήριξη γραμμής εντολών είτε ως βιβλιοθήκη γραμμένη σε γλώσσα προγραμματισμού C για χρήση σε εφαρμογές και ενσωματωμένα συστήματα.

Κάποια από τα βασικά χαρακτηριστικά που διαθέτει είναι:

- Ανεξάρτητο πλατφόρμας (Linux/Unix, Windows, iOS, embedded)
- Διαθέσιμο σε μορφή αρθώματος για (Python, Pearl, Ruby)
- Υψηλή ταχύτητα - σάρωση σε πραγματικό χρόνο από ροή βίντεο
- Μικρό αποτύπωμα μνήμης
- Μικρό μέγεθος κώδικα
- Δεν περιορίζεται στις εικόνες
- Κατάλληλο για ενσωματωμένες εφαρμογές που χρησιμοποιούν φθηνούς επεξεργαστές / υλικό

2.3 Τεχνικές Βαθιάς Μηχανικής Μάθησης και Αναγνώρισης Αντικειμένων

2.3.1 Εισαγωγή στη Τεχνητή Νοημοσύνη

Ένας από τους συνεχώς αναπτυσσόμενους κλάδους της επιστήμης της πληροφορικής, με τεράστιο ερευνητικό και πρακτικό ενδιαφέρον τα τελευταία χρόνια, αποτελεί αυτός της Τεχνητής νοημοσύνης. Διακρίνεται σε δύο υποκατηγορίες: την συμβολική και την υποσυμβολική τεχνητή νοημοσύνη. Η συμβολική κατηγορία επιδιώκει να επιλύσει προβλήματα τεχνητής νοημοσύνης χρησιμοποιώντας αλγοριθμικές διαδικασίες, δηλαδή σύμβολα και λογικούς κανόνες, ενώ η υποσυμβολική κατηγορία προσπαθεί να αναπαράγει την ανθρώπινη “ευφυΐα” μέσω αριθμητικών μοντέλων που προσπαθούν να προσομοιώσουν την λειτουργία του ανθρώπινου εγκεφάλου.

Ο John McCarthy [31], ο οποίος θεωρείται ένας από τους θεμελιωτές της τεχνητής νοημοσύνης, όρισε τον τομέα αυτόν ως «επιστήμη και μεθοδολογία της δημιουργίας νοούντων μηχανών».

Η προσφορά της Τεχνητής Νοημοσύνης στην ανθρωπότητα είναι αρκετά σημαντική, διότι η ύπαρξη όλων των καθημερινών εφαρμογών της που τυγχάνουν να θεωρούνται δεδομένες ή ελάχιστονος σημασίας οφείλονται σε αυτή.

2.3.2 Μηχανική μάθηση

Η Μηχανική Μάθηση (Machine Learning) αποτελεί ένα παρακλάδι της τεχνητής νοημοσύνης. Το όνομα αποδόθηκε από τον Arthur Samuel το 1959 [32] και βασίζεται στην καινοτόμα προσέγγιση ότι οι υπολογιστές αντί να προγραμματίζονται ντετερμινιστικά για τον τρόπο εκτέλεσης εντολών, να μπορούν από μόνοι τους να διδαχθούν πως θα πρέπει να δράσουν, ώστε να αντιμετωπίσουν προβλήματα αντίληψης στον πραγματικό κόσμο και να παίρνουν υποκειμενικές αποφάσεις.

Στόχος της αποτελεί η εξέλιξη του παραδοσιακού προγραμματισμού στο βαθμό όπου ένα πρόγραμμα μηχανικής μάθησης να δύναται να αξιοποιήσει διαφορετικούς αλγορίθμους να προσαρμόζεται και τελικά να επιλύει το εκάστοτε πρόβλημα. Για

παράδειγμα, ένα νευρωνικό δίκτυο που έχει εκπαιδευτεί να αναγνωρίζει τουλίπες σε εικόνες, δεν απαιτεί να εκπαιδευτεί εκ νέου από το μηδέν για να αναγνωρίζει και τριαντάφυλλα.

Αρκετά προβλήματα όπου η λύση τους μέχρι και πριν από μερικά χρόνια δινόταν με την ενσωμάτωση ανθρώπινης γνώσης στον κώδικα, σήμερα επιλύονται με χρήση αλγορίθμων μηχανικής μάθησης. Μερικά παραδείγματα αφορούν:

- Μηχανική Όραση (Machine Vision)
 - Εντοπισμός και αναγνώριση αντικειμένων σε εικόνες και video (Object Detection & Recognition)
- Ρομποτική (Robotics)
- Μηχανές Αναζήτησης (Search Engines)
- Ανίχνευση κυβερνοεπιθέσεων (Cyberattack detection)
- Επεξεργασία φυσικής γλώσσας (Natural Language Processing)
 - Κατανόηση της φυσικής γλώσσας και μοντελοποίηση της για χρήση από μηχανές (Natural Language Understanding and Generation)
- Παιξιμο Παιγνίων (Game playing)
- Βιοπληροφορική (Bioinformatics)

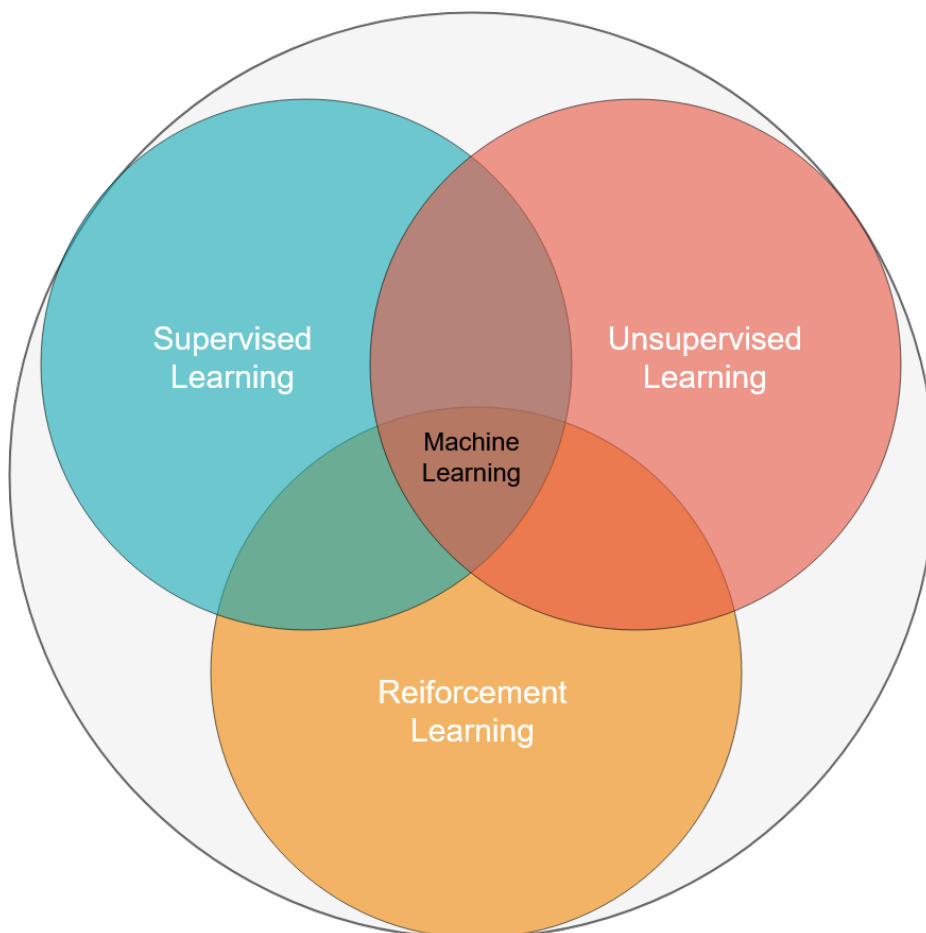
Η μηχανική μάθηση ταξινομείται σε τρεις βασικές κατηγορίες ανάλογα με την φύση των δεδομένων εισόδου και την ανατροφοδότηση σε ένα σύστημα εκμάθησης [33] :

- Μάθηση υπό επίβλεψη (Supervised Learning): Στο υπολογιστικό σύστημα δίνονται ως είσοδο δεδομένα με ετικέτες που αντιπροσωπεύουν την κατηγορία στην οποία ανήκουν. Στόχος είναι να εξαχθεί ένας γενικός κανόνας αντιστοίχισης της εισόδου με την κατάλληλη κατηγορία. Η αναγνώριση προκαθορισμένων αντικειμένων σε εικόνες είναι ένα είδους προβλήματος που ανήκει σε αυτή την κατηγορία.

- Μάθηση χωρίς επίβλεψη (Unsupervised Learning): Τα δεδομένα δεν έχουν ετικέτες (labels) αφήνοντας έτσι τον αλγόριθμο ML να μάθει από τα δεδομένα εξερευνώντας διάφορα μοτίβα.
- Εκμάθηση δια ανταμοιβής (Reinforcement Learning): Ο πράκτορας αλληλεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να εκτελέσει ένα συγκεκριμένο στόχο. Το μοντέλο πράκτορας «αμείβεται» ή «τιμωρείται» για τις αποφάσεις που πήρε προκειμένου να φτάσει στον στόχο του. Ένα παράδειγμα εφαρμογής είναι η αυτόματη πλοήγηση ενός οχήματος αποφυγής εμποδίων.

Γενικότερα, οι αλγόριθμοι ML ομαδοποιούνται και ανάλογα με την ομοιότητα τους σε σχέση με την λειτουργία που εκτελούν.

Στη Εικόνα 2.5 απεικονίζεται το διάγραμμα Venn των διαφόρων αλγοριθμικών κατηγοριών ML.



Εικόνα. 2.5: Διάγραμμα Venn των διαφόρων κατηγοριών μηχανικής μάθησης

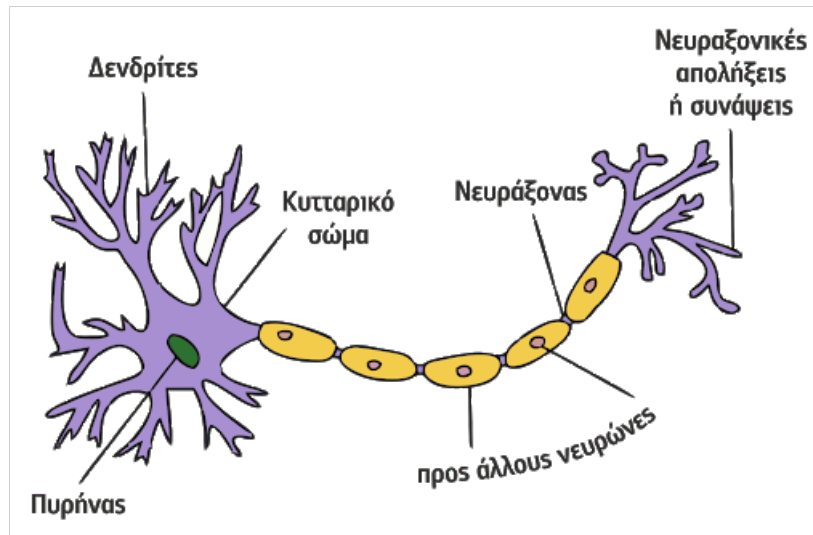
2.3.3 Τεχνητά Νευρωνικά Δίκτυα

Η μηχανική μάθηση έχει στη διάθεση της διάφορους τρόπους και τεχνικές για να επιλύει προβλήματα. Ένας από αυτούς είναι τα τεχνητά νευρωνικά δίκτυα, artificial neural networks (ANN). Αποτελούνται από συστήματα όπου επεξεργάζονται δεδομένα τα οποία εξαρτώνται από ένα πλήθος τεχνητών νευρώνων οργανωμένων σε δομές βασισμένες σε αυτές του ανθρώπινου εγκεφάλου. Η δομή του συστήματος αποτελείται από τρία επίπεδα. Το πρώτο είναι το επίπεδο εισόδου (input layer), όπου χρησιμοποιείται για την εισαγωγή δεδομένων, το επόμενο επίπεδο ονομάζεται κρυφό επίπεδο (hidden layer) και αποτελείται από νευρώνες κατανεμημένους σε διάφορα επίπεδα επιτελώντας ξεχωριστές λειτουργίες και τέλος υπάρχει το επίπεδο εξόδου (output layer). Η αύξηση επιπέδων μπορεί μεν να επιφέρει μεγαλύτερη ακρίβεια αλλά παράλληλα να αυξήσει την πολυπλοκότητα σε τέτοιο βαθμό που να καθιστά το σύστημα μη αποδοτικό στην πράξη.

Η μοντελοποίηση του βιολογικού νευρώνα με το μοντέλο Perceptron [34], [35] από τον Frank Rosenblatt το 1957 θεωρείται ίσως τη σπουδαιότερη συνεισφορά στο χώρο των νευρωνικών δικτύων. Το μοντέλο νευρώνα Perceptron είναι εμπνευσμένο από το βιολογικό οπτικό σύστημα.

Η είσοδος σε κάθε νευρώνα προέρχεται από ένα ή πολλούς νευρώνες μέσω των δενδριτών. Το σήμα που έλαβε ο νευρώνας επεξεργάζεται και το αποτέλεσμα που προκύπτει μεταφέρεται στον νευράξονα όπου μέσω των συνάψεων μεταβιβάζεται στους άλλους νευρώνες. Οι νευρώνες μπορούν να βρίσκονται σε μία από τις εξής δύο καταστάσεις, να πυροδοτούν ή όχι τους επόμενους συνδεδεμένους νευρώνες, γεγονός που μεταφράζεται σε 1 και 0 αντίστοιχα. Πιο συγκεκριμένα, για να υπάρξει έξοδος θα πρέπει να έχει λάβει μια τιμή μεγαλύτερη από αυτή που ορίζει η συνάρτηση ενεργοποίησης, η οποία βρίσκεται στο πυρήνα κάθε νευρώνα. Για παράδειγμα, η είσοδος μπορεί να αποτελεί την χρωματική κωδικοποίηση ενός pixel στο χρωματικό μοντέλο RGB μια εικόνας, η οποία μπαίνει ως είσοδος στην συνάρτηση ενεργοποίησης μαζί με άλλα pixel και υπολογίζονται τα αντίστοιχα βάρη τους. Αν ο νευρώνας δεχθεί συνολικό σήμα μεγαλύτερο από το σταθμικό μέσο της συνάρτησης ενεργοποίησης ή έχει ορισθεί μια σταθερά πόλωσης (bias), τότε αντιδρά και

ενεργοποιείται στέλνοντας το κατάλληλο σήμα στην επόμενη σύναψη. Αναλόγως της εκπαίδευσης και του επιθυμητού αποτελεσμάτων οι τιμές βάρους των νευρώνων αλλάζουν δυναμικά. Συνεπώς, διαπιστώνεται πως η γενική λειτουργία των βιολογικών νευρώνων ταυτίζεται με αυτή των ANN.

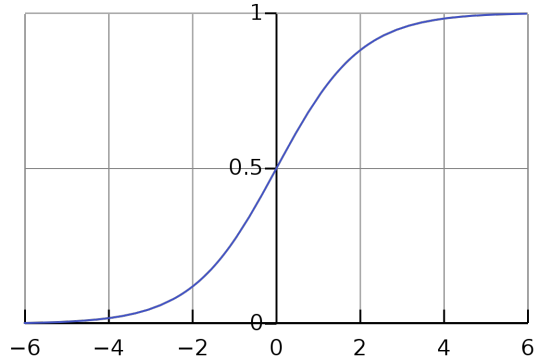


Εικόνα. 2.6: Βιολογικός νευρώνας

Συμπερασματικά το μοντέλο νευρώνα Perceptron δέχεται πληροφορία που μεταφέρεται από τις νευραξονικές απολήξεις (x_i), προτού μεταφερθεί στους δενδρίτες των επόμενων νευρώνων και αλληλεπιδρά πολλαπλασιαστικά με τις συνάψεις ($w_i * x_i$). Οι παράγοντες πολλαπλασιασμού w_n ονομάζονται βάρη και αποτελούν τις μεταβλητές παραμέτρους ενός νευρώνα. Η τιμή των παραμέτρων αυτών ελέγχει την επίδραση μεταξύ των νευρώνων. Η είσοδος bias (b) (πόλωση) παίρνει τιμές $(-1,1)$ και χρησιμοποιείται για τον προσδιορισμό της ενεργοποίησης του νευρώνα. Κάθε νευρώνας έχει έναν όρο bias, ο οποίος αλλάζει την τιμή του μαζί με τα υπόλοιπα βάρη. Η συνάρτηση ενεργοποίησης (activation function) f αποτελεί ένα γραμμικό φίλτρο που ελέγχει την ροή της πληροφορίας στους συνδεδεμένους νευρώνες και προσδίδει ευελιξία και ικανότητα εκτίμησης όσο αφορά πολύπλοκες μη γραμμικές σχέσεις στα δεδομένα εισόδου. Παρακάτω παρατίθενται μερικές ευρέως χρησιμοποιούμενες συναρτήσεις ενεργοποίησης.

- Σιγμοειδής (Sigmoid)

$$f(x) = \frac{1}{1 + e^{-x}} \quad (2.1)$$

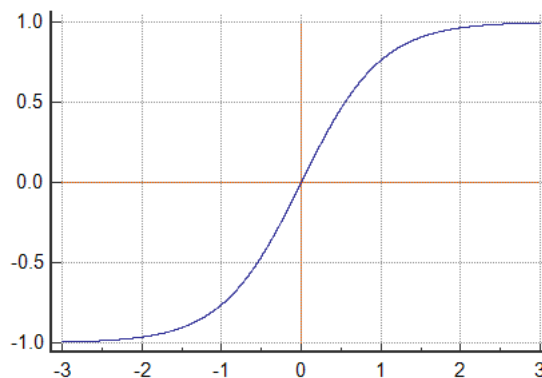


Εικόνα. 2.7: Σιγμοειδή Συνάρτηση

Η συνάρτηση αυτή παίρνει ως είσοδο έναν πραγματικό αριθμό και τον κανονικοποιεί στο διάστημα $[0, 1]$. Παρά το γεγονός ότι στην αρχή της ανάπτυξης νευρωνικών δικτύων χρησιμοποιούταν εκτενώς, εμφανίζει μερικά μειονεκτήματα. Η παράγωγός της σε ακραίες τιμές τείνει να μηδενιστεί οπότε περιορίζει τη διαδικασία μάθησης. Επιπρόσθετα, δεν έχει συμμετρικότητα ως προς το 0 και το υπολογιστικό κόστος αυξάνεται όταν απαιτηθεί η ύψωσή της σε δύναμη [36].

- Υπερβολική Εφαπτομένη (tanh)

$$f(x) = \frac{(e^x - e^{-x})}{(e^x + e^{-x})} \quad (2.2)$$

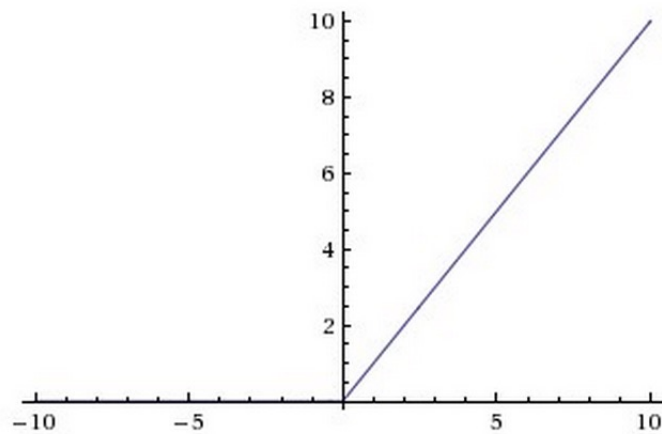


Εικόνα. 2.8: Συνάρτηση tanh

Η \tanh είναι συμμετρική ως προς το 0 κανονικοποιείται στο διάστημα $[-1, 1]$ και εμφανίζει το ίδιο πρόβλημα με την σιγμοειδή στις ακραίες τιμές παραγώγου.

- Ανορθωμένη Γραμμική Μονάδα (ReLU)

$$f(x) = \begin{cases} 0 & \text{για } x \leq 0 \\ x & \text{για } x > 0 \end{cases} = \max\{0, x\} = x1_{x>0} \quad (2.3)$$

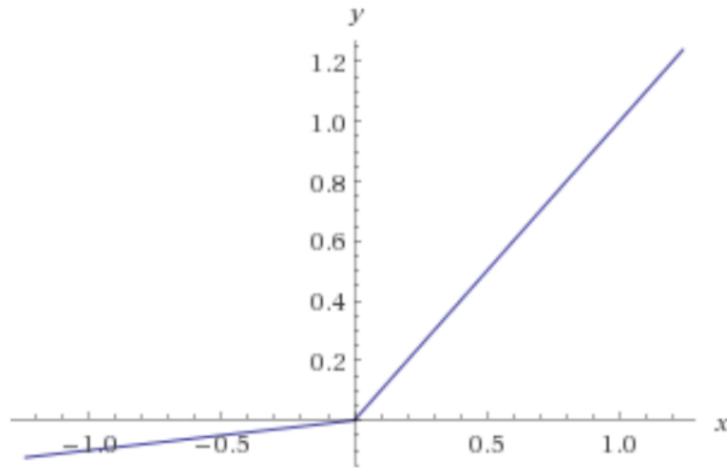


Εικόνα. 2.9: Συνάρτηση ReLU

Η ανορθωμένη γραμμική συνάρτηση φέρει αναβαθμίσεις σε σύγκριση με τις προηγούμενες [37]. Παρόλο που δεν είναι συμμετρική, δεν εμφανίζει το πρόβλημα στις θετικές ακραίες τιμές και διατηρεί τις επιθυμητές ιδιότητες ενώ η απλότητα που έχει βοηθά στη ομαλή εκπαίδευση ενός συστήματος. Παρουσιάζει βέβαια πιο έντονο πρόβλημα σε σχέση με τις προηγούμενες συναρτήσεις στις αρνητικές τιμές καθώς η τιμή για αρνητικές τιμές είναι μόνιμα μηδέν. Αυτό έχει σαν αποτέλεσμα αν ένας νευρώνας στη διαδικασία εκπαίδευσης εμφανίσει αρνητική τιμή να παραμείνει ανενεργός μέχρι το πέρας αυτής.

- Διακεκομμένη Ανορθωμένη Γραμμική Μονάδα (Leaky ReLU)

$$f(x) = \begin{cases} 0.01x & \text{για } x < 0 \\ x & \text{για } x \geq 0 \end{cases} \quad (2.4)$$

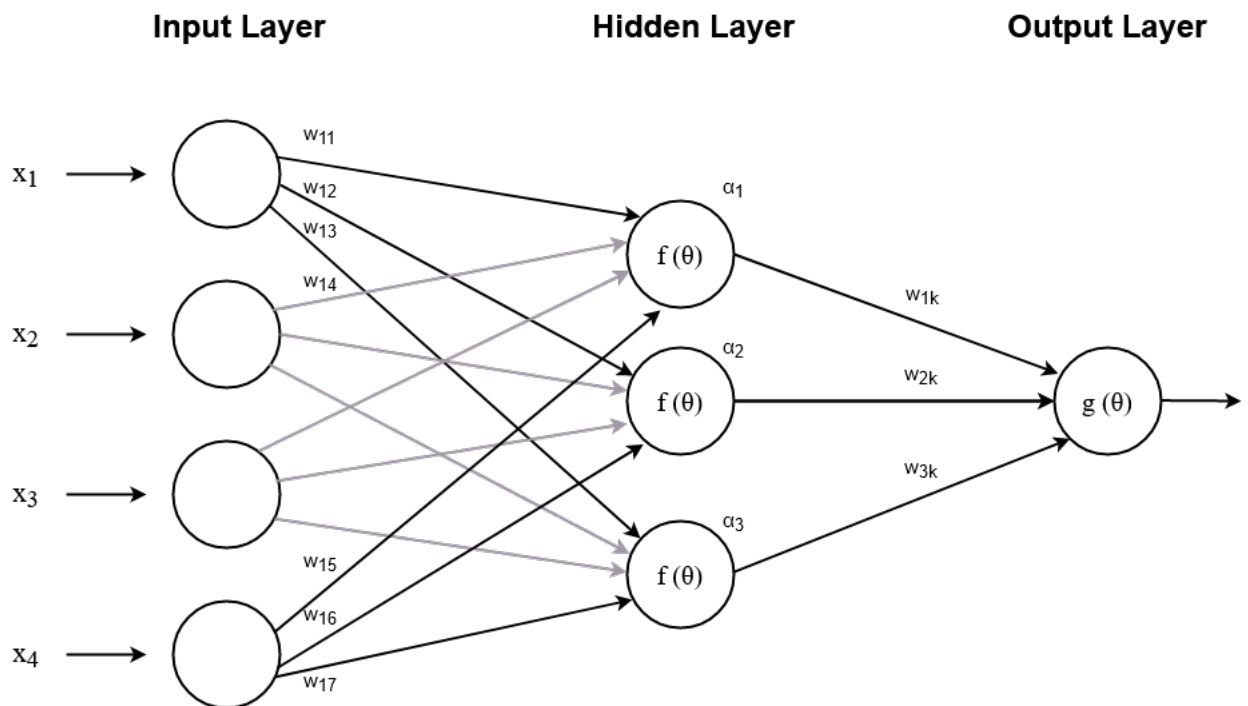


Εικόνα. 2.10: Συνάρτηση Leaky ReLU

Η εξελιγμένη αυτή συνάρτηση ενεργοποίησης διατηρεί όλα τα πλεονεκτήματα της ReLU ενώ ταυτόχρονα επιλύει και το βασικό πρόβλημα που αντιμετώπιζε για αρνητικές τιμές. Αποτελεί πλέον βασικό εργαλείο στα προβλήματα μηχανικής μάθησης.

Η έξοδος από ένα νευρώνα υπολογίζεται από τη σχέση:

$$\alpha = f\left(\sum_{i=0}^N w_i x_i + b\right) \quad (2.5)$$



Εικόνα. 2.11: Απλό μοντέλο NN με ένα κρυφό επίπεδο - Perceptron

Όταν οι νευρώνες ενός δικτύου είναι όλοι συνδεδεμένοι μεταξύ τότε χαρακτηρίζεται ως πλήρως συνδεδεμένο αλλιώς είναι μερικώς συνδεδεμένο. Στη περίπτωση όπου δεν υπάρχει σύνδεση μεταξύ νευρώνων προηγούμενου επιπέδου, τα δίκτυα φέρουν πρόσθια τροφοδότηση, ενώ στη περίπτωση συνδέσεων μεταξύ νευρώνων ίδιου επιπέδου, χαρακτηρίζεται με ανατροφοδότηση.

Βασικές λειτουργίες των ANN είναι η λειτουργία της μάθησης όπου κατά την διαδικασία της εκπαίδευσης, χρησιμοποιείται για να τροποποιήσει την τιμή των βαρών του δικτύου και η λειτουργία ανάκλησης (recall) όπου υπολογίζει ένα διάνυσμα εξόδου για ένα συγκεκριμένο διάνυσμα εισόδου και τιμές βαρών.

Ακόμα, τα ANN μπορούν να εμφανίσουν φαινόμενα υποπροσαρμογής ή ατελούς μάθησης (underfitting) ή υπερπροσαρμογής (overfitting). Όταν το δίκτυο είναι αρκετά απλό τότε μπορεί να αποτύχει να μοντελοποιήσει τα δεδομένα εκπαίδευσης και έτσι να οδηγηθεί σε υποπροσαρμογή. Σε άλλη περίπτωση, ένα πολύπλοκο νευρωνικό δίκτυο μπορεί να οδηγήσει σε υπερβολική μοντελοποίηση των δεδομένων εκπαίδευσης καθώς και του θορύβου που μπορεί να υπάρχει σε αυτά. Συνέπεια αυτού είναι η εμφάνιση υπερπροσαρμογής όπου προβλέπει ορθά τα δεδομένα εκ-

παίδευσης ωστόσο παράγει λάθος προβλέψεις για τα επόμενα δεδομένα εισόδου.

Τα ANN είναι ιδιαίτερα χρήσιμα σε προβλήματα που δεν μπορούν να γίνουν προβλέψεις όπως προβλήματα σε πολλές ανθρώπινες δραστηριότητες που σχετίζονται με την ταξινόμηση (classification), αναγνώριση (recognition), αποτίμηση (assessment) και πρόβλεψη (prediction).

2.3.4 Βαθιά Μάθηση

Με την άνοδο της τεχνολογίας, την έρευνα και την προσθήκη νέων τεχνικών, κάνει την εμφάνισή της μια νέα μορφή δικτύων αυτή των βαθύς νευρωνικών δικτύων (DNN) και οι μέθοδοι εκπαίδευσής τους ονομάζονται τεχνικές βαθιάς μάθησης (deep learning) [38]. Σε αντίθεση με τα απλά νευρωνικά δίκτυα, στη βαθιά μάθηση έχουμε πολύ περισσότερα κρυφά επίπεδα που αποσκοπούν στην εξαγωγή συμπερασμάτων με μεγαλύτερη ακρίβεια καθώς και εξαγωγή χαρακτηριστικών υψηλού επιπέδου. Τα δίκτυα συνήθως λαμβάνουν πολυδιάστατα δεδομένα και σε συνδυασμό με την ύπαρξη πολλών επιπέδων, η επεξεργασία αυτών καθίσταται εξαιρετικά χρονοβόρα. Παρόλα αυτά με την εμφάνιση πολυπύρηνων GPUs έχει επιτευχθεί η δραστική μείωση του χρόνου εκπαίδευσης τους.

2.3.5 Συνελικτικά Νευρωνικά Δίκτυα

Στις προηγούμενες ενότητες έγινε αναφορά στα νευρωνικά δίκτυα και την γενικότερη λειτουργία τους. Σε αυτή την παράγραφο γίνεται αναφορά σε συγκεκριμένα μοντέλα πολυεπίπεδων νευρωνικών δικτύων και πιο συγκεκριμένα στα νευρωνικά δίκτυα συνέλιξης.

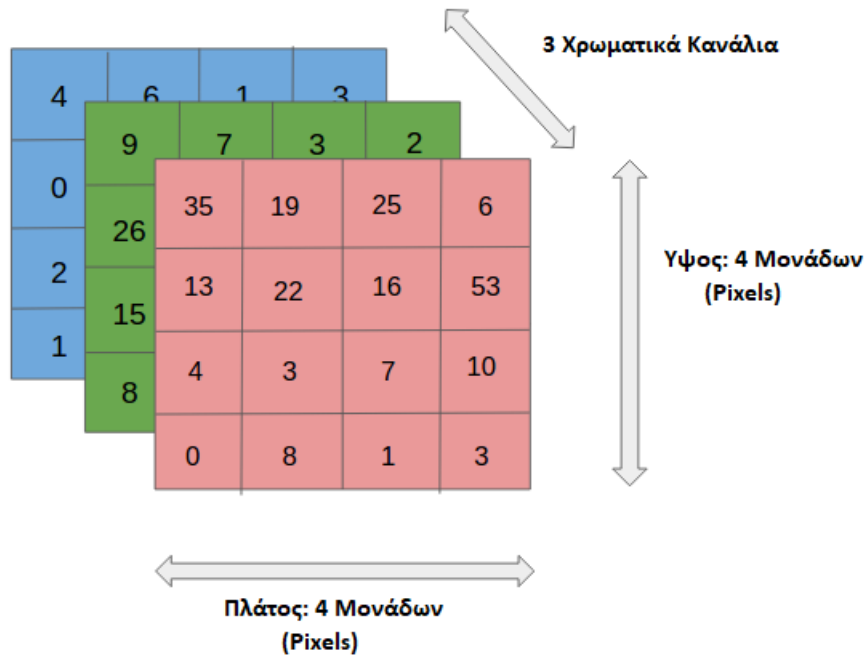
Αφού εξεταστεί η γενικότερη λειτουργία των νευρωνικών δικτύων, σε αυτή τη παράγραφο γίνεται αναφορά στα νευρωνικά δίκτυα συνέλιξης. Αυτά ανήκουν στην κατηγορία των πολυεπίπεδων νευρωνικών δικτύων και κύρια εφαρμογή αυτών των δικτύων είναι στα προβλήματα εντοπισμού αντικειμένων σε εικόνες. Οι αλγόριθμοι νευρωνικών δικτύων δεν ανακαλύφθηκαν πρόσφατα. Ήδη από την δεκαετία του 1940 είχε ξεκινήσει η έρευνα σε μαθηματικά μοντέλα νευρώνων και αλγορίθμων

εκπαίδευσής. Την δεκαετία του 1990 χρησιμοποιήθηκαν για τη λύση του προβλήματος της αναγνώρισης εικόνων. Τα τελευταία χρόνια το ενδιαφέρον της επιστημονικής κοινότητας έχει στραφεί έντονα στα νευρωνικά δίκτυα λόγω της προόδου στον σχεδιασμό υλικού, της μετάβασης σε ολοένα και μικρότερες λιθογραφίες ολοκλήρωσης επεξεργαστών, της ανάπτυξης νέων μηχανισμών παραλληλοποίησης, της αύξησης του μέγεθος της μνήμης, καθώς και της αφθονίας των διαθέσιμων δεδομένων. Στον τομέα της μηχανικής όρασης έχει σημειωθεί σημαντική πρόοδος στους αλγορίθμους που χρησιμοποιούν CNN και πλέον μέθοδοι όπως YOLO [38] και Faster R-CNN [39] βρίσκονται στο επίκεντρο.

Ο τρόπος λειτουργίας δεν διαφέρει με αυτόν που παρουσιάστηκε προηγουμένως, δηλαδή αποτελούνται από πολλά επίπεδα, όπου το κάθε επίπεδο αποτελείται από έναν αριθμό νευρώνων οι οποίοι έχουν σαν παραμέτρους εκμάθησης τα βάρη τους w_j^i την τιμή πόλωσης b^i . Το σήμα εισόδου εισέρχεται σε κάθε νευρώνα εκτελείται μία πράξη εσωτερικού γινομένου και μπορεί να εφαρμοστεί στο αποτέλεσμα μία μη γραμμική συνάρτηση. Το τελευταίο επίπεδο των CNN είναι πλήρες συνδεδεμένο και έχει μία συνάρτηση σφάλματος. Η διαφορά μεταξύ των μοντέλων CNN και των ANN έγκειται στο γεγονός ότι τα δεδομένα εισόδου των CNN είναι εικόνες.

Στα CNN η τελική πληροφορία δομείται από μικρότερα τμήματα πληροφορίας τα οποία ενώνονται σταδιακά σε κάθε επίπεδο. Για παράδειγμα, σε ένα μοντέλο CNN το πρώτο επίπεδο προσπαθεί να εντοπίσει ακμές, το δεύτερο επίπεδο παίρνει την πληροφορία των ακμών και προσπαθεί να εντοπίσει περιγράμματα κ.λπ.

Εάν υποθέσουμε ότι θέλουμε να εισάγουμε μια εικόνα σε ένα δίκτυο CNN, το οποίο έχει οριστεί να δέχεται εικόνες ανάλυσης 32x32 και έχοντας υπόψιν ότι κάθε εικονοστοιχείο που υπάρχει στην εικόνα χαρακτηρίζεται από 3 τιμές στο χρωματικό μοντέλο RGB, τότε το επίπεδο εισόδου θα έχει διαστάσεις 32x32x3 εισάγοντας έτσι την έννοια του βάθους.

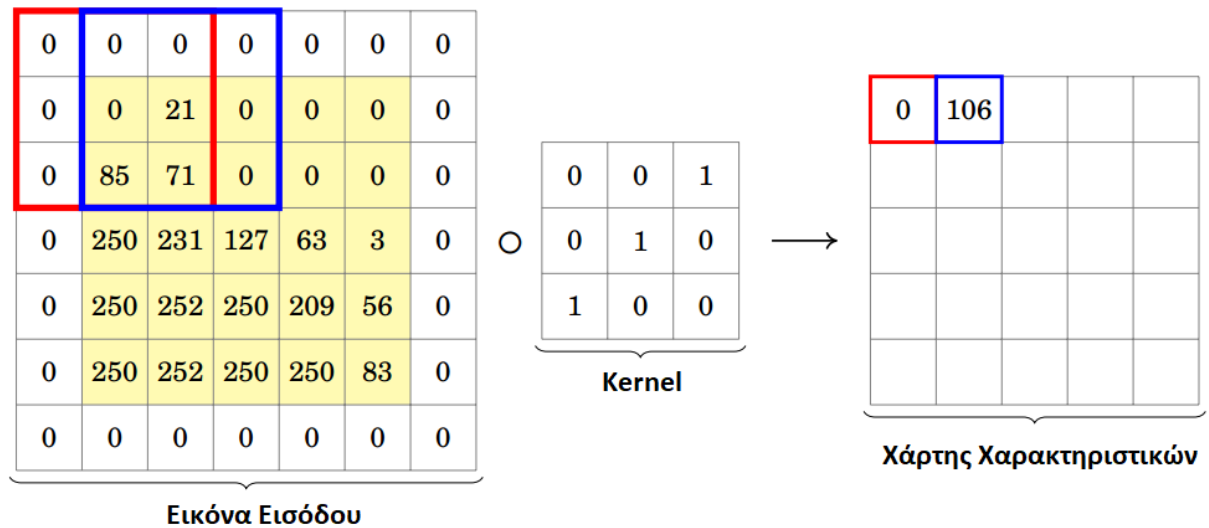


Εικόνα. 2.12: Τρισδιάστατη κατανομή εικόνας

Τα 3 βασικά στρώματα ενός συνελικτικού δικτύου αναλύονται παρακάτω.

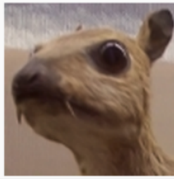

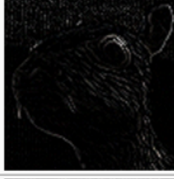

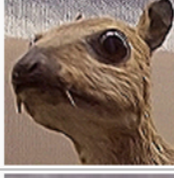
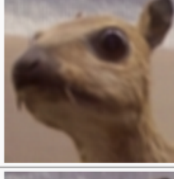
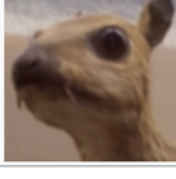
Συνελικτικό επίπεδο

Το συνελικτικό επίπεδο αποτελεί τη βάση του CNN και εκτελεί τις βασικές λειτουργίες της εκπαίδευσης και είναι υπεύθυνο για την εξαγωγή των βασικών χαρακτηριστικών. Πριν όμως γίνει αναφορά για τη διαδικασία εξαγωγής θα πρέπει πρώτα να γίνει κατανοητή η έννοια του συνελικτικού φίλτρου ή kernel. Το φίλτρο αποτελεί ένα τετραγωνικό πίνακα μικρότερο σε σχέση με την είσοδο και με βάθος ίσο με την εικόνα εισόδου. Το φίλτρο διατρέχει διαδοχικά την εικόνα εισόδου για κάθε κανάλι χρώματος και τα αποτελέσματα των πολλαπλασιασμών των δύο πινάκων συνθέτουν τον χάρτη χαρακτηριστικών [40].



Εικόνα. 2.13: Διαδικασία συνέλιξης

Η συνέλιξη μιας εικόνας με διαφορετικά φίλτρα μπορεί να εξάγει χαρακτηριστικά όπως ανίχνευση άκρων, θόλωση κ.λπ. Το παρακάτω παράδειγμα δείχνει διάφορες εικόνες συνέλιξης μετά την εφαρμογή διαφορετικών τύπων φίλτρων (kernels):

Operation	Filter	Convolved Image
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

Εικόνα. 2.14: Εφαρμογή Φίλτρων

Ο όγκος δεδομένων που θα εξαχθούν από ένα συνελικτικό επίπεδο εξαρτάται από πολλούς παράγοντες και κάποιοι από αυτούς είναι το μέγεθος και ο αριθμός των φίλτρων (depth). Η ύπαρξη πολλών διαφορετικών φίλτρων, παρόλο που κοστίζει υπολογιστικά, μεγαλώνει τον όγκο εξόδου και συμβάλλει στην εξαγωγή περισσότερων χαρακτηριστικών. Το βήμα (stride) είναι ο αριθμός των μετατοπίσεων pixel πάνω στον πίνακα εισόδου. Όταν το βήμα είναι 1, τότε μεταφέρουμε τα φίλτρα σε 1 pixel κάθε φορά. Όταν το βήμα είναι 2 τότε μετακινούμε τα φίλτρα σε 2 pixel

κάθε φορά και ούτω καθεξής. Μικρή ολίσθηση οδηγεί σε μεγαλύτερη έξοδο και αντίστροφα. Επίσης, σε ορισμένες περιπτώσεις, το φίλτρο δεν ταιριάζει τέλεια στην εικόνα εισόδου και για να μη χαθούν τυχόν χαρακτηριστικά που βρίσκονται κοντά στο πλαίσιο της εικόνας, είτε εφαρμόζεται γέμισμα (padding), συνήθως με μηδενικά (zero padding) στο περίγραμμα της, είτε αγνοείται αυτό το σημείο της εικόνας όπου το φίλτρο δεν ταιριάζει τέλεια. Αυτό καλείται έγκυρο padding (valid padding) στο οποίο κρατείται μόνο το ωφέλιμο σημείο της εικόνας. Τόσο το padding, όσο και το depth και το stride αποτελούν υπερπαραμέτρους του δικτύου.

Συνοψίζοντας τη διαδικασία που αναλύθηκε δοθέντος μιας εισόδου με μέγεθος $W_1 \times H_1 \times D_1$ εφαρμόζοντας N αριθμό φίλτρων μεγέθους $K \times K \times D_1$ με βήμα μετατόπισης (stride) S και γέμισμα (padding) P , ο παραγόμενος όγκος θα έχει μέγεθος $W_2 \times H_2 \times D_2$.

$$W_2 = \frac{W_1 - K + 2P}{S + 1} \quad (2.6)$$

$$H_2 = \frac{H_1 - K + 2P}{S + 1} \quad (2.7)$$

$$D_2 = K \quad (2.8)$$

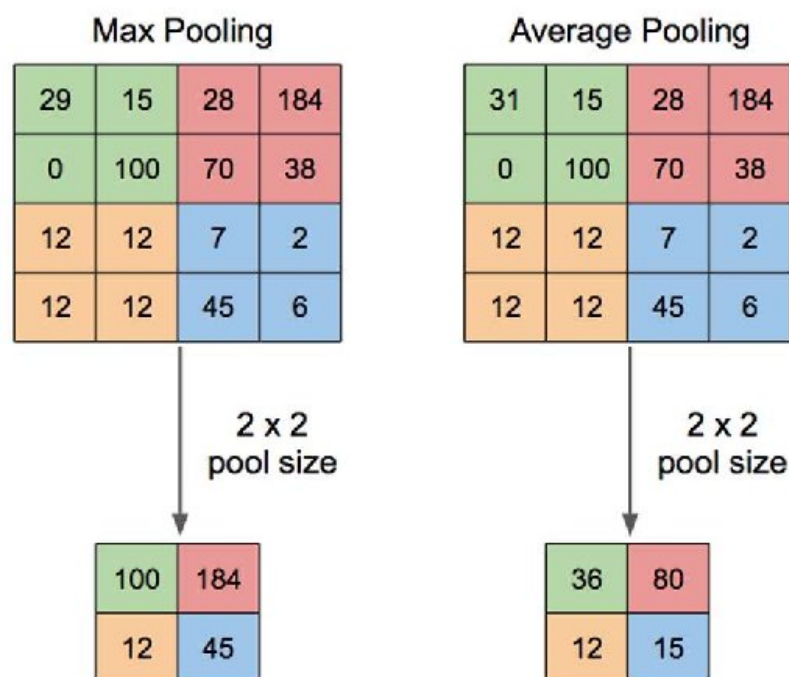
Στο επόμενο στάδιο της διαδικασίας οι χάρτες χαρακτηριστικών περνάνε από τις συναρτήσεις ενεργοποίησης. Στα συνελκτικά νευρωνικά δίκτυα χρησιμοποιούνται οι ReLU και η παραλλαγή της Leaky ReLU.

Επίπεδο Συγκέντρωσης (Pooling Layer)

Το επίπεδο συγκέντρωσης τοποθετείται μετά το επίπεδο συνέλιξης. Η κύρια χρησιμότητά του έγκειται στη μείωση των διαστάσεων του όγκου δεδομένων και κατά επέκταση των παραμέτρων και της πολυπλοκότητας του για το επόμενο επίπεδο συνέλιξης. Δεν επηρεάζει τη διάσταση βάθους του όγκου των δεδομένων. Η λειτουργία που εκτελείται από αυτό το επίπεδο ονομάζεται επίσης «down sampling» (μείωση δειγματοληψίας), καθώς η μείωση του μεγέθους οδηγεί και σε απώλεια πληροφοριών. Ωστόσο, μια τέτοια απώλεια είναι επωφελής για το δίκτυο για δύο λόγους:

- Η μείωση του μεγέθους οδηγεί σε μικρότερα υπολογιστικά κόστη για τα επερχόμενα επίπεδα του δικτύου
- Λειτουργεί κατά της υπερπροσαρμογής

Όπως και η λειτουργία της συνέλιξης που εκτελείται παραπάνω, το επίπεδο συγκέντρωσης χρησιμοποιεί ένα συρόμενο παράθυρο με το οποίο κινείται διαδοχικά στην είσοδο μετατρέποντας τις τιμές σε αντιπροσωπευτικές. Ο μετασχηματισμός πραγματοποιείται είτε με τη λήψη της μέγιστης τιμής από τις τιμές που παρατηρούνται στο παράθυρο με τη μέθοδο «max pooling», είτε με τη λήψη του μέσου όρου των τιμών «average pooling». Η μέγιστη συγκέντρωση προτιμάται έναντι άλλων τεχνικών λόγω των καλύτερων χαρακτηριστικών της απόδοσης. Για παράδειγμα, εάν η είσοδος είναι όγκος μεγέθους $4 \times 4 \times 3$ και το συρόμενο παράθυρο έχει μέγεθος 2×2 , τότε για κάθε κανάλι χρώματος, οι τιμές θα υποβληθούν σε δειγματοληψία προς την αντιπροσωπευτική μέγιστη τιμή τους εάν εκτελεστεί η λειτουργία μέγιστης συγκέντρωσης. Το επίπεδο συγκέντρωσης είναι προαιρετικό στα CNN και πολλές αρχιτεκτονικές με τη χρήση συνεχόμενων συνελικτικών επιπέδων έχουν αποκτήσει μεγαλύτερη απήχηση καθώς σε ορισμένα προβλήματα μηχανικής μάθησης είναι αποτελεσματικότερες.

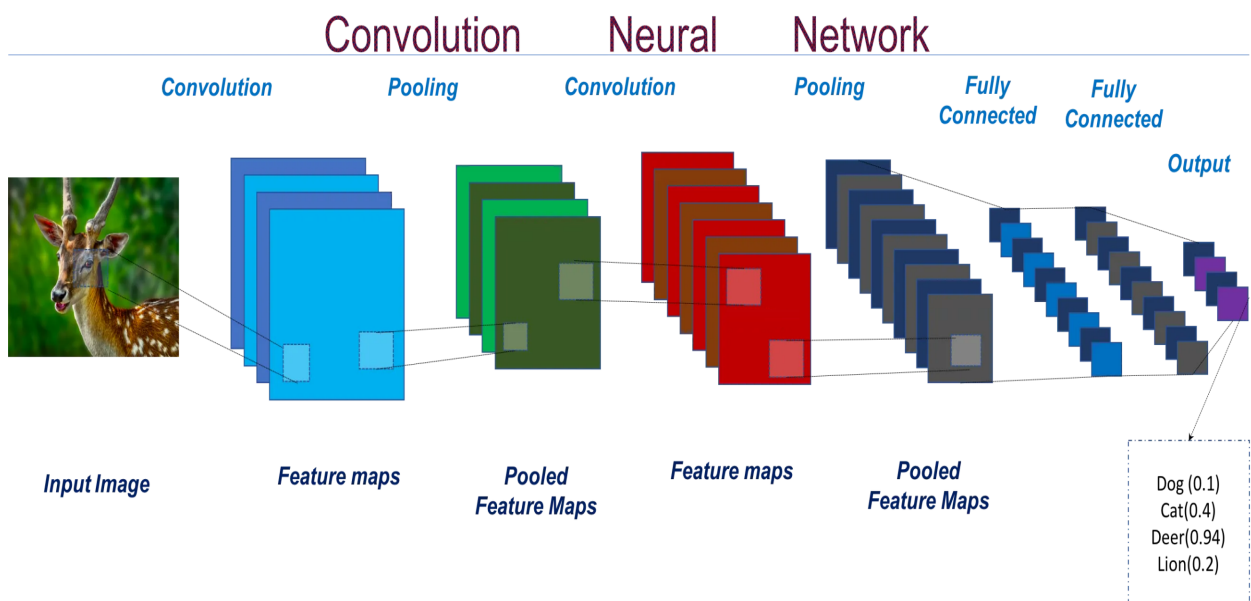


Εικόνα. 2.15: Μέθοδοι Συγκεντρωτικού Επιπέδου

Πλήρως Συνδεδεμένο Επίπεδο (Fully-Connected Layer)

Η προσθήκη ενός πλήρως συνδεδεμένου επιπέδου (Fully Connected Layer) είναι ένας (συνήθως) φθηνός υπολογιστικά τρόπος εκμάθησης μη γραμμικών συνδυασμών υψηλού επιπέδου χαρακτηριστικών όπως προκύψαν από την έξοδο του συνελικτικού επιπέδου. Σε αυτό το επίπεδο, κάθε νευρώνας του προηγούμενου επιπέδου συνδέεται με όλους τους νευρώνες του επόμενου επιπέδου. Στόχος του επιπέδου αυτού είναι η κατηγοριοποίηση των εξόδων από τα προηγούμενα επίπεδα σε κλάσεις και πιθανώς συνοδευόμενο με πιθανότητα επιτυχίας.

Συνοψίζοντας παρατηρούμε ότι σε ένα συνελικτικό δίκτυο διενεργείται η εξαγωγή χαρακτηριστικών της εικόνας και η κατηγοριοποίηση αυτής.



Εικόνα. 2.16: Συνελικτικό Δίκτυο

2.3.6 Μοντέλα Ανίχνευσης Αντικειμένων

Έχοντας εξετάσει τα επιμέρους δομικά επίπεδα των συνελικτικών δικτύων, σε αυτήν την ενότητα θα εξετάσουμε βασικούς αλγορίθμους βαθιάς μηχανικής μάθησης που χρησιμοποιούνται για την ανίχνευση αντικειμένων [41].

Οι μέθοδοι ανίχνευσης αντικειμένων έχουν αναπτυχθεί είτε σε προσεγγίσεις βα-

σισμένες στη μηχανική μάθηση είτε στη βαθιά μάθηση. Όσο αφορά τη μηχανική μάθηση, αρχικά ορίζονται τα χαρακτηριστικά και στη συνέχεια χρησιμοποιείται μια τεχνική όπως η μηχανή φορέα υποστήριξης (SVM) για την ταξινόμηση. Στην περίπτωση της βαθιάς μάθησης, οι τεχνικές ανιχνεύουν για αντικείμενα σε όλο το εύρος της εικόνας χωρίς να καθορίζουν συγκεκριμένα χαρακτηριστικά βασιζόμενες σε συνελκτικά νευρωνικά δίκτυα.

Ένα από τα κύρια προβλήματα της όρασης μηχανής είναι η ανίχνευση αντικειμένων, η οποία είναι στενά συνδεδεμένη με την ταξινόμηση των αντικειμένων σε κλάσεις ούτως ώστε να εξάγεται η πληροφορία ως προς το είδος του αντικειμένου που αναγνωρίστηκε. Βασικές λειτουργίες αποτελούν ο εντοπισμός αντικειμένων (object detection/ localization) δηλαδή ο προσδιορισμός της θέσης των αντικειμένων σε μια δεδομένη εικόνα καθώς και η ταξινόμηση αυτών (object recognition) στις κατηγορίες που ανήκουν.

Οι σημαντικότεροι αλγόριθμοι βαθιάς μάθησης που χρησιμοποιούνται στην ανίχνευση αντικειμένων είναι οι εξής:

- Αλγόριθμοι Προτάσεων Περιοχής (R-CNN, Fast R-CNN, Faster R-CNN, Mask R-CNN)
- Ανιχνευτής Πολλαπλών Θυρίδων μιας Λήψης (Single Shot MultiboxDetector-SSD)
- You Only Look Once (YOLO)

Αλγόριθμοι όπως Faster R-CNN, Mask R-CNN και YOLO αποτελούν τις νεότερες προσεγγίσεις που συνδυάζουν τα συνελκτικά νευρωνικά δίκτυα με έξυπνη σχεδίαση και αποτελεσματικότητα, αυξάνοντας τη ταχύτητα αναγνώρισης όταν εκτελούνται σε ισχυρές GPUs [42].

R-CNN

Σε αυτό το σημείο θα γίνει μία σύντομη αναφορά στο R-CNN (Regions with CNN), μία μέθοδο εντοπισμού αντικειμένων σε εικόνες η οποία παρουσιάστηκε το 2014 από τους Girshick et al [43] και κατάφερε να πετύχει υψηλή απόδοση ανίχνευσης αντικειμένων σε σχέση με άλλες μεθόδους κατά την εποχή δημιουργίας του. Έπειτα, θα γίνει ανάλυση στον τρόπο λειτουργίας του δικτύου YOLO v3 [44], το οποίο και χρησιμοποιήθηκε στη διπλωματική εργασία.

Η ονομασία της μεθόδου R-CNN, έγκειται στο γεγονός ότι η αναγνώριση της εικόνας γίνεται με χρήση CNN σε περιοχές της εικόνας. Πιο συγκεκριμένα, εκτελείται επιλεκτική αναζήτηση για να εξαχθούν περίπου 2000 περιοχές από την εικόνα οι οποίες ονομάζονται προτάσεις περιοχής (region proposal). Επομένως, αντί να γίνει ταξινόμηση ενός τεράστιου αριθμού περιοχών ή η σάρωση ολόκληρης της εικόνας, ο αλγόριθμος εργάζεται με 2000 περιοχές.

Μπορούμε να χωρίσουμε τη μέθοδο σε τρία στάδια επεξεργασίας. Το πρώτο στάδιο, είναι η δημιουργία προτάσεων περιοχής, δηλαδή η υπόθεση πως κάποιες περιοχές έχουν υψηλή πιθανότητα να περιέχουν κάποιο αντικείμενο, ανεξάρτητα από την κατηγορία στην οποία μπορεί αυτό να ανήκει. Προκειμένου να προταθούν οι διάφορες περιοχές, συχνά γίνεται χρήση του αλγορίθμου επιλεκτικής αναζήτησης. Ο αλγόριθμος αυτός, δημιουργεί τμήματα της εικόνας που θα μπορούσαν να ανήκουν σε ένα αντικείμενο, βασισμένα στο χρώμα, την υφή, το μέγεθος και το σχήμα, και συνδυάζει παρόμοιες περιοχές προκειμένου να σχηματίσουν αντικείμενα. Με αυτόν τον τρόπο δημιουργούνται προτάσεις αντικειμένων με διαφορετικές κλίμακες.

Στη συνέχεια, το συνελκτικό νευρωνικό δίκτυο, του οποίου τα τελευταία επίπεδα είναι πλήρως συνδεδεμένα, αφορά την εξαγωγή χαρακτηριστικών από τις προτάσεις περιοχής. Το δίκτυο πρέπει να ρυθμιστεί με ακρίβεια για να εκπαιδευτεί αφενός ως προς τα οπτικά χαρακτηριστικά των νέων τύπων εικόνων (παραμορφωμένες προτάσεις περιοχής) και αφετέρου ως προς ειδικές κατηγορίες στόχων του μικρότερου συνόλου δεδομένων για την διαδικασία ανίχνευσης. Όλες οι προτάσεις

της περιοχής που δημιουργούνται από κάθε εικόνα, ως αποτέλεσμα χρήσης του αλγορίθμου επιλεκτικής αναζήτησης, μετατρέπονται σε εισόδους σταθερού μεγέθους, ανεξάρτητα από το πραγματικό μέγεθος ή την αναλογία διαστάσεων, προκειμένου να τελειοποιηθεί το CNN. Μια πρόσθετη παράμετρος p χρησιμοποιείται για να υποδείξει την ποσότητα της πιθανής διαστολής του αρχικού πλαισίου οριοθέτησης ώστε να συμπεριλάβει κάποιο πλαίσιο γύρω από την περιοχή αυτήν. Το κάθε πλαίσιο λαμβάνει μία ετικέτα ταξινόμησης, ανάλογα με το ποσοστό ταύτισης με κάποιο αντικείμενο. Σε περίπτωση που το ποσοστό αυτό είναι χαμηλότερο από κάποιο κατώφλι (threshold), τότε το πλαίσιο αντιμετωπίζεται ως φόντο.

Στο τέλος του πρώτου σταδίου, το δίκτυο εκπαιδεύεται με ρυθμό εκμάθησης αρκετά μικρότερο από εκείνον στην αρχική εκπαίδευση. Σε κάθε επανάληψη δοκιμάζεται ένας αριθμός παραθύρων τα οποία είναι θετικά σε όλες τις κατηγορίες και μερικά τα οποία ανήκουν στην τάξη του φόντου, για να διασφαλιστεί ότι υπάρχει επαρκής αναπαράσταση από τις θετικές κατηγορίες κατά τη διάρκεια της εκπαίδευσης.

Τα κύριο πρόβλημα του αλγορίθμου έγκειται στον εκτεταμένο χρόνο εκπαίδευσης, λόγω της ανάγκης για ταξινόμηση 2000 περιοχών ανά εικόνα. Αυτό οδηγεί στην αδυναμία εκτέλεσης σε πραγματικό χρόνο μιας και κάθε εικόνα χρειάζεται αρκετά δευτερόλεπτα για να ταξινομηθεί. Τέλος, λόγω της στατικής φύσης του αλγορίθμου, μπορεί να οδηγήσει σε λάθος προτάσεις περιοχών λόγω και της απουσίας περαιτέρω εκπαίδευσης.

Το 2015 ο R. Girshick και στην συνέχεια οι R. Shaoqing et al. πρότειναν βελτιώσεις του υπάρχοντος μοντέλου με τους αλγόριθμους Fast R-CNN [33] και Faster R-CNN [23] αντίστοιχα. Τα προβλήματα που αντιμετωπίζουν αυτοί οι αλγόριθμοι είναι το γεγονός ότι οι υπολογισμοί χαρακτηριστικών πραγματοποιούνται επανειλημμένα για κάθε προτεινόμενη περιοχή της εικόνας και η χαμηλή ταχύτητα εκτέλεσης των αλγορίθμων πρότασης περιοχών. Για το πρώτο πρόβλημα το Fast R-CNN υπολογίζει μία εικόνα χαρακτηριστικών για ολόκληρη την εικόνα και στην συνέχεια επεξεργάζεται τις προτεινόμενες περιοχές αυτής της εικόνας, ενώ ταυτόχρονα προ-

τείνει μία καθολική διαδικασία εκπαίδευσης. Με τον τρόπο αυτόν, το υπολογιστικό κόστος υπολογισμού χαρακτηριστικών διαμοιράζεται μεταξύ των περιοχών. Όσον αφορά το Faster R-CNN, αυτό αντικαθιστά τους αλγορίθμους πρότασης περιοχών με ένα μικρό CNN το οποίο ονομάζεται Δίκτυο πρότασης περιοχών και υπολογίζει τις υποψήφιες προς αναγνώριση περιοχές σαφώς ταχύτερα.

Yolo

Ο YOLO παίρνει το όνομά του από την φράση You Only Look Once (Κοιτάς Μόνο Μία Φορά) λόγω του γεγονότος ότι χρησιμοποιεί ένα πλήρες συνδεδεμένο νευρωνικό δίκτυο που με ένα πέρασμα ελέγχει την εικόνα για αντικείμενα, στα οποία έχει εκπαιδευτεί να αναγνωρίζει. Ο Yolo σε αντίθεση με τα άλλα συστήματα, δεν χρησιμοποιεί τις περιοχές ενδιαφέροντος ώστε να τις τροφοδοτήσει σε ένα συνελικτικό δίκτυο, αλλά τεμαχίζει την εικόνα σε S κελιά (cells), που στο σύνολό τους δημιουργούν ένα πλέγμα (grid), με $S \times S$ κελιά τα οποία χρησιμοποιούνται κατά την διαδικασία της εκπαίδευσης. Ουσιαστικά, αντιμετωπίζει την ανίχνευση σαν μια διαδικασία παλινδρόμησης σε χωρικά διαχωρισμένα πλαίσια οριοθέτησης (bounding boxes) και σχετικές πιθανότητες κλάσης. Οι ικανότητες του επιπλέον της ταχύτητας και της ακρίβειας, είναι και ο ταυτόχρονος εντοπισμός πολλών διαφορετικών αντικειμένων σε μία εικόνα και ο εντοπισμός τους στο χώρο.

Ο YOLO v3 είναι μια βελτίωση σε σχέση με τα προηγούμενα δίκτυα ανίχνευσης YOLO. Σε σύγκριση με προηγούμενες εκδόσεις, διαθέτει ανίχνευση πολλαπλής κλίμακας, ισχυρότερο δίκτυο εξαγωγής χαρακτηριστικών και ορισμένες αλλαγές στη λειτουργία απώλειας.

Ένα Πλήρες Συνελικτικό Νευρωνικό Δίκτυο

Ο YOLO v3 χρησιμοποιεί μόνο συνελικτικά στρώματα, καθιστώντας τον ένα πλήρως συνδεδεμένο συνελικτικό δίκτυο. Οι συγγραφείς στη δημοσίευση του αλγορίθμου παρουσιάζουν μια νέα, βαθύτερη αρχιτεκτονική του εξαγωγέα χαρακτηριστικών που ονομάζεται Darknet-53. Όπως υποδηλώνει το όνομά του, το Darknet-53

δανείζεται αυτήν την ιδέα από τη πρώτη έκδοση του αλγόριθμου YOLO v1, όπου ο εξαγωγέας Darknet-19 έκανε χρήση 19 επίπεδων και την επεκτείνει με επιτυχία σε 53 επίπεδα. Όπως παρουσιάζεται στο παρακάτω σχήμα περιέχει 53 συνελκτικά επίπεδα, το καθένα ακολουθούμενο από το επίπεδο ομαλοποίησης παρτίδων (batch normalization) και τη συνάρτηση ενεργοποίησης Leaky ReLU. Δεν χρησιμοποιείται κάποιου είδους ομαδοποίηση (pooling) και ένα συνελκτικό επίπεδο με βήμα (stride) 2 χρησιμοποιείται για την μείωση δειγματοληψίας των χαρτών χαρακτηριστικών. Αυτό βοηθά στην πρόληψη της απώλειας χαρακτηριστικών χαμηλού επιπέδου που συχνά αποδίδονται στην ομαδοποίηση.

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1x	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2x	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8x	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8x	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4x	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8
	Avgpool		Global	
	Connected		1000	
	Softmax			

Εικόνα. 2.17: Μοντέλο Darknet-53

Ο αλγόριθμός δεν απαιτεί κάποιο συγκριμένο μέγεθος στις εικόνες εισόδου, ωστόσο είναι προτιμότερο να χρησιμοποιηθεί ένα σταθερό μέγεθος εισόδου για την αύξηση της ταχύτητας επεξεργασίας αλλά και την αποφυγή διαφόρων προβλημάτων που ενδέχεται να προκύψουν κατά την εκτέλεση. Όταν οι εικόνες επεξεργάζονται σε παρτίδες μπορούν να επεξεργαστούν παράλληλα από την GPU, γεγονός που θα μειώσει σημαντικά το χρόνο εκτέλεσης. Το δίκτυο υποβαθμίζει (downsample) την εικόνα με τον παράγοντα stride του δικτύου. Αν παραδείγματος χάριν ένα δίκτυο έχει stride ίσο με 32 pixels, για μία εικόνα μεγέθους 416 x 416 θα προκύψει έξοδος μεγέθους 13 x 13.

Η Είσοδος του Συστήματος

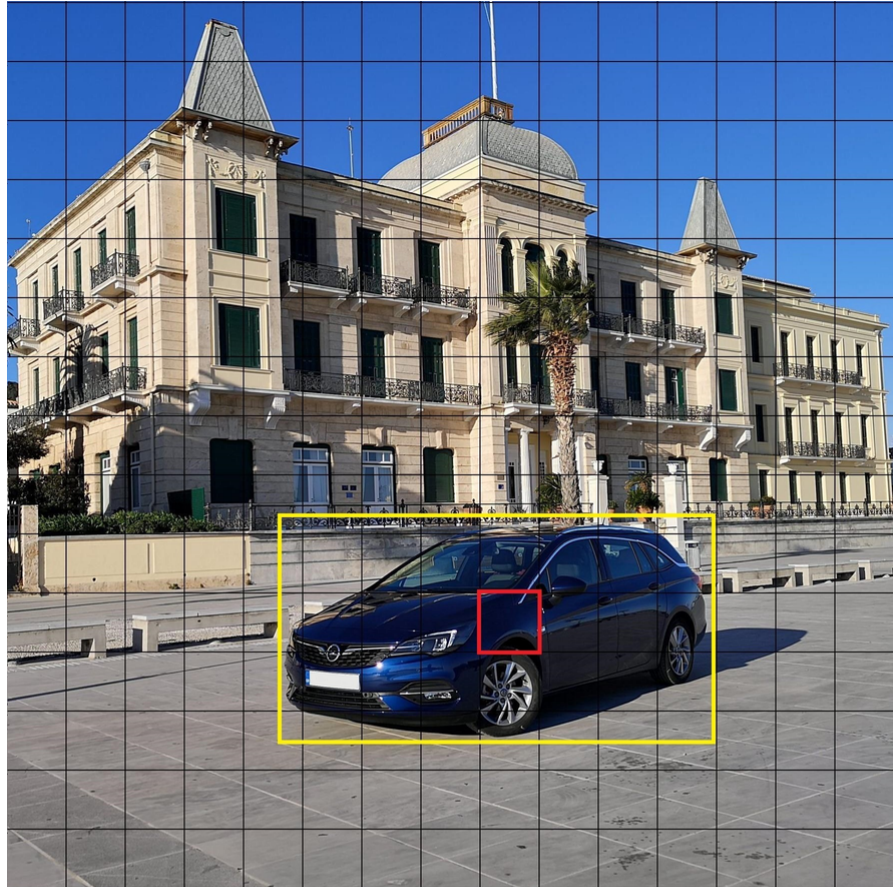
Η είσοδος στο σύστημα είναι τρισδιάστατοι πίνακες εικόνων σε παρτίδες με μέγεθος (m, 416, 416, 3) και η έξοδος λίστα κουτιών οριοθέτησης μαζί με τις αναγνωρισμένες κλάσεις. Κάθε πλαίσιο οριοθέτησης αντιπροσωπεύεται από 6 αριθμούς (pc, bx, by, bh, bw, c).

Με τον ίδιο τρόπο όπως σε όλους τους αλγόριθμους εντοπισμού αντικειμένων τα χαρακτηριστικά όπου το συνελικτικό δίκτυο έμαθε από τα συνελικτικά στρώματα μεταφέρονται σε έναν ταξινομητή (classifier) / παλινδρομητή (regressor) που κάνει την πρόβλεψη ανίχνευσης (συντεταγμένες των ορίων οριοθέτησης, την ετικέτα κλάσης, κ.λπ.).

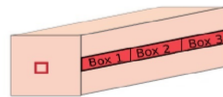
Στο YOLO, η πρόβλεψη γίνεται χρησιμοποιώντας ένα συνελικτικό επίπεδο που χρησιμοποιεί 1×1 συνελίξεις. Έτσι, το πρώτο που εξάγεται είναι ένας χάρτης χαρακτηριστικών. Δεδομένου ότι έχουν χρησιμοποιηθεί 1×1 συνελίξεις, το μέγεθος του χάρτη προβλέψεων (prediction map) είναι ακριβώς το μέγεθος του χάρτη δυνατοτήτων. Στο YOLO v3, ο τρόπος με τον οποίο αποδίδεται αυτός ο χάρτης πρόβλεψης είναι ότι κάθε κελί μπορεί να προβλέψει έναν καθορισμένο αριθμό πλαισίων οριοθέτησης (bounding box).

Εάν εξεταστεί η περίπτωση που υπάρχουν ($B \times (5 + C)$) εισαγωγές στο χάρτη χαρακτηριστικών (με το B να αντιπροσωπεύει τον αριθμό των πλαισίων οριοθέτησης που μπορεί να προβλέψει κάθε κελί), κάθε ένα από αυτά τα κουτιά οριοθέτησης B μπορεί να ειδικεύεται στην ανίχνευση ενός συγκεκριμένου είδους αντικειμένου. Κάθε ένα από τα κουτιά οριοθέτησης έχει $(5 + C)$ ιδιότητες, οι οποίες περιγράφουν τις συντεταγμένες κέντρου του, τις διαστάσεις, τη βαθμολογία αντικειμενικότητας και τη τάξη C εμπιστοσύνης για την πρόβλεψη. Ο αλγόριθμος YOLO v3 προβλέπει 3 κουτιά οριοθέτησης για κάθε κελί. Αυτό σημαίνει ότι κάθε κελί που ανήκει στο χάρτη χαρακτηριστικών μπορεί να προβλέπει ένα αντικείμενο μέσω ενός από τα οριοθετημένα πλαίσια εάν το κέντρο του αντικειμένου βρίσκεται στην περιοχή αυτού του κελιού. Αυτό συνδέεται με το πως έχει εκπαιδευτεί ο YOLO v3 όπου μόνο ένα πλαίσιο οριοθέτησης είναι υπεύθυνο για την ανίχνευση οποιουδήποτε αντικειμένου.

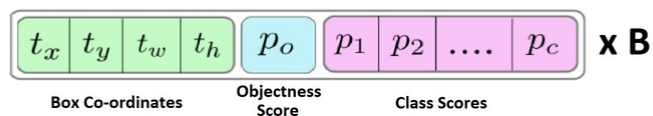
Αρχικά για να μπορέσει να εξακριβωθεί σε ποιο κελί ανήκει αυτό το πλαίσιο οριοθέτησης διαιρείται η εικόνα εισόδου σε ένα πλέγμα διαστάσεων ίσο με εκείνο του τελικού χάρτη χαρακτηριστικών. Στο παρακάτω παράδειγμα πρόκειται για εικόνα εισόδου διαστάσεων 640×640 και το stride του δικτύου είναι 42. Εφαρμόζοντας τις εξισώσεις 2.6, 2.7, 2.8 που αναλύθηκε πριν προκύπτει πίνακας χαρακτηριστικών μεγέθους 15×15 άρα και η εικόνα θα χωριστεί σε 15×15 κελιά.



Prediction Feature Map



Attributes of a bounding box

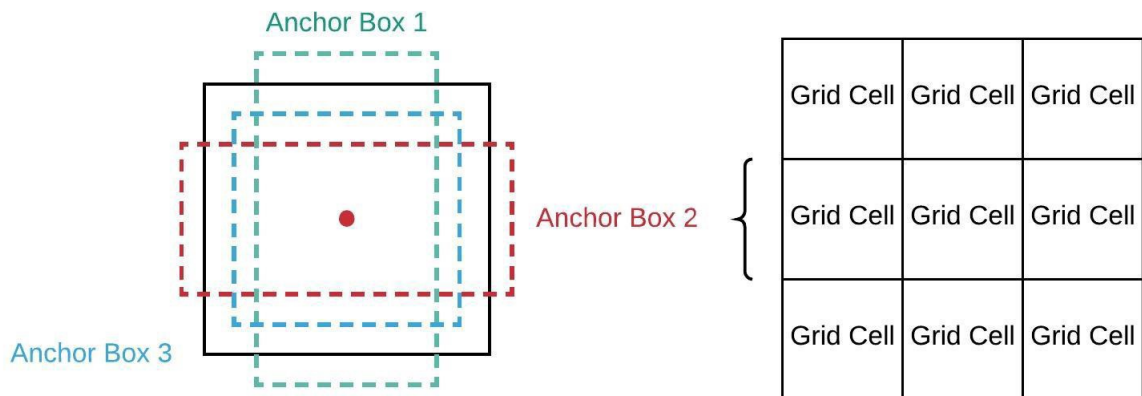


Εικόνα. 2.18: Prediction Feature Map

Στη συνέχεια, επιλέγεται το κελί από την εικόνα εισαγωγής που βρίσκεται στο κέντρο να είναι υπεύθυνο για την πρόβλεψη του αντικειμένου. Στην εικόνα, είναι 9ο κελί στην 11η σειρά στο πλέγμα που σημειώνεται με κόκκινο περίγραμμα και είναι υπεύθυνο για την ανίχνευση του αυτοκινήτου. Το κελί αυτό έχει τη δυνατότητα να προβλέψει 3 πλαίσια οριοθέτησης για το αντικείμενο. Προκειμένου να βρεθεί σε ποιο από αυτά τα 3 πλαίσια θα ανατεθεί η ετικέτα εντοπισμού θα πρέπει να αναλυθούν τα αγκυρωμένα πλαίσια. Με λίγα λόγια η διαδικασία που προηγήθηκε, ο διαχωρισμός δηλαδή της εικόνας σε κελιά, εξυπηρετεί στο να γίνει κατανοητό ποιο

κελί είναι υπεύθυνο για την ανίχνευση του αντικειμένου.

Αγκυρωμένα πλαίσια (Anchor boxes)



Εικόνα. 2.19: Αγκυρωμένα πλαίσια

Θα ήταν σίγουρα πιο εύκολο να υπολογιστούν εξ αρχής οι διαστάσεις του πλαισίου οριοθέτησης αλλά λόγω του γεγονότος ότι, αυτό μπορεί να οδηγήσει σε ασταθή διαβάθμιση κατά τη διάρκεια της προπόνησης, δεν εφαρμόζεται. Αντ' αυτού, οι περισσότεροι από τους σύγχρονους αλγόριθμους εντοπισμού αντικειμένων προβλέπουν μετασχηματισμούς λογαριθμικού χώρου ή προκαθορισμένα και προεπιλεγμένα πλαίσια οριοθέτησης που ονομάζονται αγκυρωμένα πλαίσια. Στη συνέχεια, αυτοί οι μετασχηματισμοί εφαρμόζονται στα αγκυρωμένα πλαίσια για να ληφθεί η πρόβλεψη.

Τα αγκυρωμένα πλαίσια είναι ένα είδος πλαισίου οριοθέτησης, τα οποία υπολογίστηκαν στο COCO dataset χρησιμοποιώντας k-means συσταδοποίηση. Το πλάτος και το ύψος του κουτιού προκύπτει ως μετατόπιση από το κέντρο ενός αντικειμένου (centroids) των συστάδων. Οι κεντρικές συντεταγμένες του πλαισίου σε σχέση με τη θέση της εφαρμογής φίλτρου προβλέπονται χρησιμοποιώντας τη σιγμοειδή συνάρτηση.

Ο παρακάτω τύπος περιγράφει τον τρόπο μετατροπής της εξόδου δικτύου για τη λήψη προβλέψεων πλαισίων οριοθέτησης:

$$b_x = \sigma(t_x) + c_x \quad (2.9)$$

$$b_y = \sigma(t_y) + c_y \quad (2.10)$$

$$b_w = p_w e^{t_w} \quad (2.11)$$

$$b_h = p_h e^{t_h} \quad (2.12)$$

Οι μεταβλητές b_x, b_y, b_h, b_w, b_h οι συντεταγμένες κέντρου x, y , δηλαδή το πλάτος και ύψος της πρόβλεψης. Οι μεταβλητές t_x, t_y, t_w, t_h το αποτέλεσμα εξόδου του δικτύου, οι c_x και c_y είναι οι πάνω-αριστερές συντεταγμένες του πλέγματος και τέλος οι p_w και p_h είναι οι διαστάσεις του αγκυρωμένου πλαισίου.

Κεντρικές Συντεταγμένες (Centroid)

Το «centroid» όπως μαρτυρά και το όνομά του είναι το κέντρο ενός αντικειμένου. Είναι το αποτέλεσμα μιας σιγμοειδούς συνάρτησης με είσοδο τις συντεταγμένες κέντρου του αντικειμένου b_x και b_y και έξοδο στο διάστημα 0 έως 1. Αυτό συμβαίνει διότι ο YOLO v3 δεν προβλέπει τις απόλυτες συντεταγμένες του κέντρου του πλαισίου οριοθέτησης αλλά τις μετατοπίσεις που:

- Είναι ανάλογες με την επάνω αριστερή γωνία του κελιού πλέγματος που προβλέπει το αντικείμενο
- Ομαλοποιούνται από τις διαστάσεις του κελιού από το χάρτη χαρακτηριστικών, το οποίο είναι 1

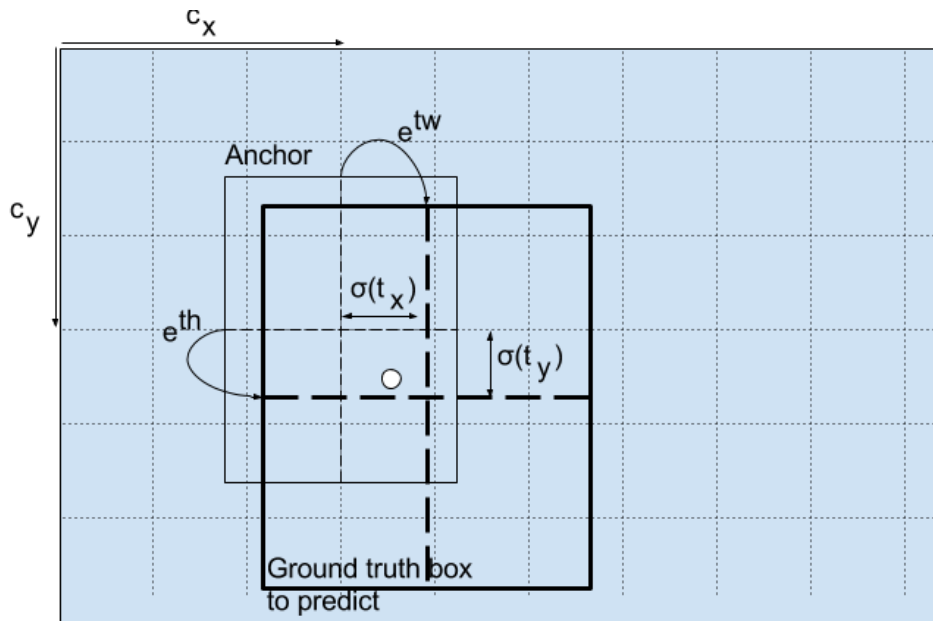
Για παράδειγμα, μπορεί να εξεταστεί η περίπτωση της παραπάνω εικόνας του αυτοκινήτου. Εάν οι συντεταγμένες πρόβλεψης για το κέντρο είναι (0.4, 0.7), τότε αυτό σημαίνει ότι το κέντρο βρίσκεται στο (8.4, 10.7) στο χάρτη χαρακτηριστικών 15 x 15. (Δεδομένου ότι οι άνω αριστερά συντεταγμένες του κόκκινου κελιού είναι

(8,10)).

Το πρόβλημα προκύπτει εάν οι εκτιμώμενες συντεταγμένες x και y είναι μεγαλύτερες από 1, για παράδειγμα (1.2, 0.7). Αυτό σημαίνει ότι το κέντρο βρίσκεται στο (9.2, 10.7), δηλαδή στο 10ο κελί στην 11η σειρά ακριβώς στο δίπλα από το κόκκινο κελί. Έτσι, για να λυθεί αυτό το πρόβλημα, η έξοδος περνά μέσω της σιγμοειδής συνάρτησης, η οποία συμπιέζει την έξοδο από 0 έως 1, διατηρώντας αποτελεσματικά το κέντρο στο κελί που προβλέφθηκε.

Διαστάσεις του κουτιού οριοθέτησης

Οι διαστάσεις του πλαισίου οριοθέτησης προβλέπονται εφαρμόζοντας λογαριθμικό μετασχηματισμό στην έξοδο και μετά πολλαπλασιάζοντας με τις διαστάσεις του αγκυρωμένου πλαισίου.



Εικόνα. 2.20: Σχηματική απεικόνιση υπολογισμού των Bounding Boxes

Εδώ οι προβλέψεις, bw και bh ομαλοποιούνται από το ύψος και το πλάτος της εικόνας εισόδου. Έτσι, εάν οι προβλέψεις bw, bh το πλαίσιο που περιέχει το αυτοκίνητο είναι (0.3, 0.8), τότε το πραγματικό πλάτος και ύψος στο χάρτη χαρακτηριστικών 15×15 είναι $(15 \times 0.3, 15 \times 0.8)$.

Βαθμολογία αντικειμενικότητας και εμπιστοσύνη τάξης (Objectness Score και Class Confidences)

Ο βαθμός αντικειμενικότητας αντιπροσωπεύει την πιθανότητα ότι ένα αντικείμενο βρίσκεται μέσα σε ένα πλαίσιο οριοθέτησης. Η μέγιστη τιμή που μπορεί να πάρει είναι 1 και αυτό θα συμβεί για το κόκκινο κελί και τα γειτονικά του ενώ η ελάχιστη είναι το μηδέν και θα ισχύει για τα κελιά που βρίσκονται στις γωνίες του πλέγματος. Η βαθμολογία αντικειμενικότητας περνά επίσης από μία σιγμοειδή συνάρτηση, καθώς πρέπει να ερμηνευτεί ως πιθανότητα.

Οι τάξεις εμπιστοσύνης, αντιπροσωπεύουν τις πιθανότητες του αντικειμένου που ανιχνεύθηκε να ανήκει σε μια συγκεκριμένη τάξη (αυτοκίνητο, ποδήλατο, κινητό τηλέφωνο, κουτάλι, μαχαίρι).

Προβλέψεις σε διαφορετικές κλίμακες

Ο YOLO v3 κάνει προβλέψεις σε 3 διαφορετικές κλίμακες. Το επίπεδο ανίχνευσης χρησιμοποιείται σε χάρτες χαρακτηριστικών τριών διαφορετικών μεγεθών, με βήματα 32, 16 και 8 αντίστοιχα. Αυτό σημαίνει ότι γίνονται ανιχνεύσεις σε κάθε μία από τις κλίμακες αν για παράδειγμα η εικόνα εισαγωγής έχει μέγεθος 416 x 416 τότε θα εκτελεστεί η διαδικασία εντοπισμού για τις κλίμακες 13 x 13, 26 x 26 και 52 x 52.

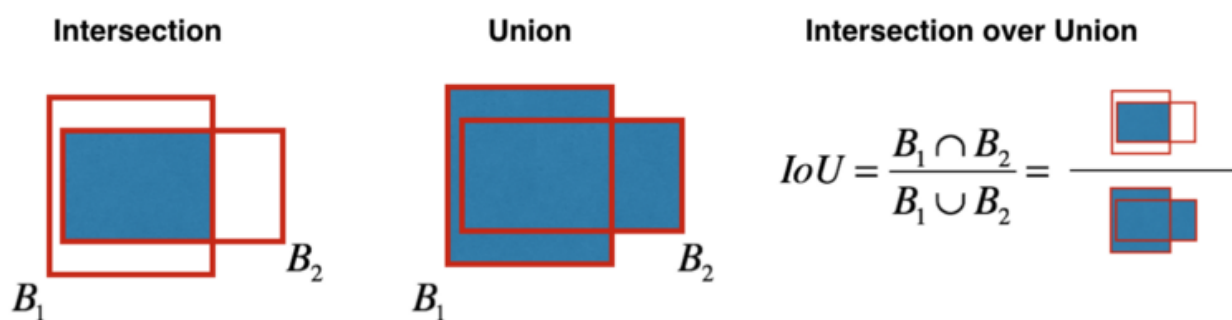
Το δίκτυο μειώνει τη δειγματοληψία στην εικόνα εισόδου μέχρι το πρώτο επίπεδο ανίχνευσης, όπου εκτελείται ανίχνευση χρησιμοποιώντας τους χάρτες χαρακτηριστικών ενός επιπέδου με stride ίσο με 32. Ακολούθως, τα επόμενα επίπεδα υπερδειγματοληπτούν με παράγοντα 2 και συνδυάζονται με τους χάρτες χαρακτηριστικών προηγούμενων επιπέδων έχοντας το ίδιο μέγεθος. Μια άλλη ανίχνευση πραγματοποιείται στο επίπεδο με το stride 16. Επαναλαμβάνεται η ίδια διαδικασία υπερδειγματοληψίας και η τελική ανίχνευση γίνεται στο επίπεδο με stride 8. Σε κάθε κλίμακα, κάθε κελί προβλέπει 3 πλαίσια οριοθέτησης χρησιμοποιώντας 3 αγκυρωμένα πλαίσια κάνοντας τον συνολικό αριθμό αγκυρωμένων πλαισίων που χρησιμοποιήθηκαν σε 9.

Οι συγγραφείς του YOLO v3 αναφέρουν ότι η υπερδειγματοληψία μπορεί να βοηθήσει το δίκτυο να μάθει λεπτομερείς λειτουργίες που είναι καθοριστικές για τον εντοπισμό μικρών αντικειμένων σε αντίθεση με τις προηγούμενες εκδόσεις.

Επεξεργασία εξόδου

Για την μείωση των οριοθετημένων πλαισίων που έχουν δημιουργηθεί από κάθε κλίμακα και την ύπαρξη μόνο αυτών που περικλείουν τα αναγνωρισμένα αντικείμενα, πρώτα, φιλτράρονται τα πλαίσια με βάση τη βαθμολογία αντικειμενικότητάς τους. Τα πλαίσια με βαθμολογίες κάτω από ένα όριο (για παράδειγμα κάτω από 0.5) αγνοούνται. Στη συνέχεια, η μη μέγιστη καταστολή (NMS) σκοπεύει να εξουδετερώσει το πρόβλημα πολλαπλών ανιχνεύσεων του ίδιου αντικειμένου. Ακολούθως, ένα δεύτερο φίλτρο για την επιλογή των σωστών πλαισίων ονομάζεται NMS εφαρμόζεται. Το NMS χρησιμοποιεί την συνάρτηση που ονομάζεται «Διασταύρωση μέσω Ένωσης» ή IoU.

Στην πράξη ορίζεται ένα πλαίσιο και επιλέγεται να χρησιμοποιηθούν οι δύο γωνίες του (πάνω αριστερά και κάτω δεξιά): $(x1, y1, x2, y2)$ αντί να χρησιμοποιηθεί το σημείο στο κέντρο του.



Εικόνα. 2.21: Σχηματική απεικόνιση του Intersection over Union

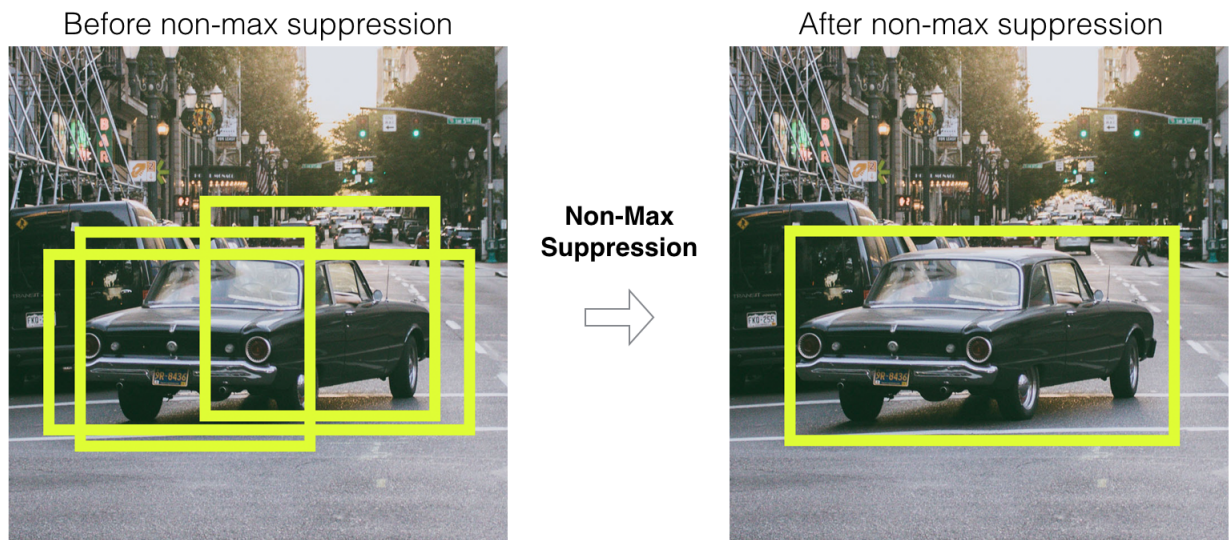
Για τον υπολογισμό της επιφάνειας ενός ορθογωνίου πολλαπλασιάζεται το ύψος του $(y2 - y1)$ με το πλάτος του $(x2 - x1)$ και για την εύρεση των συντεταγμένων $(x1, y1, x2, y2)$ της τομής δύο κουτιών υπολογίζονται:

- x_{i1} = μέγιστο των συντεταγμένων x_1 των δύο πλαισίων
- y_{i1} = μέγιστο των συντεταγμένων y_1 των δύο πλαισίων
- x_{i2} = ελάχιστο των συντεταγμένων x_2 των δύο πλαισίων
- y_{i2} = ελάχιστο των συντεταγμένων y_2 των δύο πλαισίων

Για να εφαρμοστεί η μη μέγιστη καταστολή, τα βασικά βήματα είναι:

1. Επιλογή του πλαισίου που έχει την υψηλότερη βαθμολογία
2. Υπολογισμός της επικάλυψής του με όλα τα άλλα πλαίσια και αφαίρεση των πλαισίων που επικαλύπτουν περισσότερο μέσω του `iou_threshold`
3. Επιστροφή στο βήμα 1 και επανάληψη έως ότου δεν υπάρχουν περισσότερα πλαίσια με χαμηλότερη βαθμολογία από το τρέχον επιλεγμένο πλαίσιο

Αυτά τα βήματα θα αφαιρέσουν όλα τα πλαίσια που έχουν μεγάλη επικάλυψη με τα επιλεγμένα πλαίσια. Απομένουν μόνο τα «καλύτερα» πλαίσια:



Εικόνα. 2.22: Επίδραση της Non-max suppression

2.4 Αναγνώριση φωνής

Με τον όρο αναγνώριση φωνής [45] αναφερόμαστε στην ικανότητα του συστήματος να συλλάβει τον προφορικό λόγο του χρήστη και να μετατρέψει σε κείμενο τα ηχητικά κύματα της φωνής.

Η προσπάθεια για μετατροπή του προφορικού λόγου, ως κυρίαρχο μέσω της ανθρώπινης επικοινωνίας, σε γραπτό λόγο από τις μηχανές είναι εγχείρημα που απασχολεί την επιστήμη πριν ακόμα την δεκαετία του 70'. Ξεκινώντας από τα εργαστήρια του Bell όπου δημιουργείται μια μηχανή με όνομα Audrey (1952) η οποία μπορεί να αναγνωρίσει ποιος αριθμός από το 1 έως το 9 εκφωνήθηκε στο μικρόφωνο της, φτάνουμε στο σήμερα όπου ηλεκτρονικοί βοηθοί όπως η Google Assistant και η Alexa μπορούν μόνο μέσω φωνητικών εντολών να χειριστούν κινητές συσκευές και online site πωλήσεων.

Για να επιτευχθεί ο στόχος ενός συστήματος αναγνώρισης ομιλίας, δηλαδή η ορθή μετατροπή του λόγου σε κείμενο, απαιτείται να ξεπεραστούν σημαντικά εμπόδια που αφορούν τον ανθρώπινο προφορικό λόγο όπως τα διαφορούμενα νοήματα στις λέξεις και η προφορά της ίδιας λέξης από διαφορετικούς χρήστες καθώς και προβλήματα που αφορούν το περιβάλλον θορύβου της ομιλίας και το μέγεθος του λεξιλογίου που γνωρίζει η μηχανή.

2.4.1 Τρόπος λειτουργίας

Οι απαιτήσεις μια μηχανής αναγνώρισης ομιλίας, όπως αναφέρθηκαν και παραπάνω, κάνουν αντιληπτή την ανάγκη για ακριβείς αλγορίθμους και τεχνικές λειτουργίας ώστε η παραγωγή του τελικού γραπτού κειμένου να είναι όμοια τόσο γραμματικά όσο και λεξιλογικά με την ομιλία του χρήστη στην είσοδο.

Το πρώτο στάδιο λειτουργίας αφορά τη ψηφιοποίηση της ομιλίας. Αυτό περιλαμβάνει την δειγματοληψία, δηλαδή τη λήψη των κυμάτων της φωνής μέσω μικροφώνου, την κβαντοποίηση και την κωδικοποίηση του ήχου, δηλαδή την μετατροπή του

σε bits.

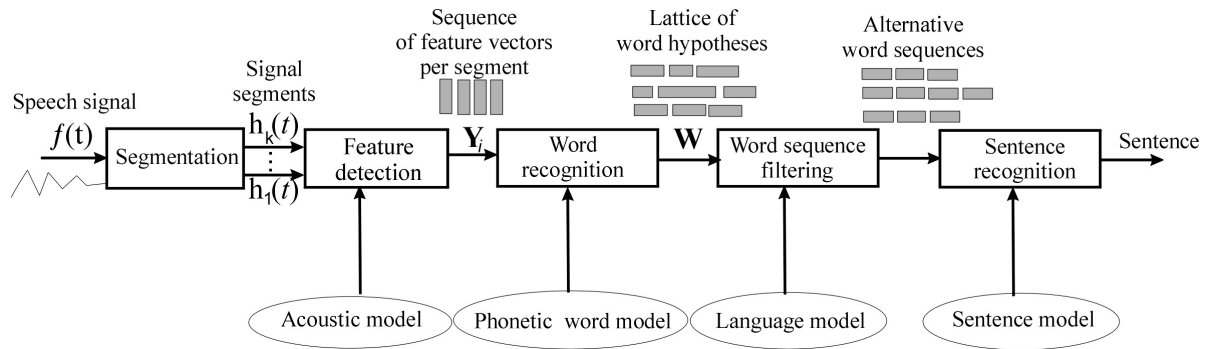
Το δεύτερο στάδιο αφορά την επεξεργασία του σήματος όπου περιλαμβάνονται διαδικασίες όπως η απομάκρυνση του θορύβου, η προσαρμογή της ταχύτητας ανάλογα με την μηχανή και η ομαλοποίηση της έντασης του ήχου. Για της επίτευξη αυτών χρησιμοποιούνται εργαλεία όπως το ML Frequency Cepstral Coefficients (MFCC) όπου εφαρμόζουν φίλτρα πάνω στα πλαίσια (frames) του σήματος κρατώντας την χρήσιμη πληροφορία και απορρίπτοντας τις υπόλοιπες.

Στο τρίτο στάδιο, γίνεται ο διαχωρισμός του ήχου σε φωνήματα, δηλαδή στο κώδικα από bits που έχει σχηματιστεί αποδίδονται τα στοιχεία της γλώσσας. Η διαδικασία αυτή αποτελεί την ακουστική ανάλυση και διαφέρει ανάλογα την γλώσσα του ομιλητή. Για παράδειγμα στην ελληνική γλώσσα τα φωνήματα που οφείλει να ξέρει η μηχανή είναι 24 συν τους δίφθογγους.

Για την αναγνώριση των φωνημάτων χρησιμοποιούνται ακουστικά μοντέλα όπως τα κρυφά μοντέλα Markov (Hidden Markov Models – HMM) [46] τα οποία αναπαριστούν τα φωνήματα της λέξης σε αλυσίδες και δημιουργούν όλους τους πιθανούς συνδυασμούς που μπορούν να σχηματίσουν λέξεις. Με τον τρόπο αυτό πραγματοποιείται η γλωσσική ερμηνεία η οποία όμως χρειάζεται να γνωρίζει την πιθανότερη λέξη, από το σύνολο αυτών που σχηματίζονται, την οποία χρησιμοποίησε ο ομιλητής.

Για την ορθή επιλογή της λέξης, από τα μοντέλα Markov, που θα προκύψει τελικά στο κείμενο του υπολογιστή, απαιτείται το σημαντικότερο στάδιο της λειτουργίας που είναι η αναγνώριση και η εκπαίδευση της μηχανής. Σε αυτό το στάδιο η μηχανή, πριν τεθεί σε χρήση, έρχεται σε επαφή με εκατομμύρια προτάσεις την γλώσσας που μετατρέπει, με σκοπό να γνωρίζει τις πιθανές χρήσης κάθε λέξης και να μπορεί να επιλεγεί την πιο πιθανή σειρά των φωνημάτων και των λέξεων από τα συμφραζόμενα της πρότασης που έχει πάρει στην δειγματοληψία.

Στο τελικό στάδιο, τα bits που έχει συγκεντρώσει ο υπολογιστής από τις παραπάνω διαδικασίες, κωδικοποιούνται με την χρήση του ASCII κώδικα και παρουσιάζονται στην χρήση ως γραπτός λόγος.



Εικόνα. 2.23: Επίπεδα επεξεργασίας σήματος φωνής

2.4.2 Google voice Recognition API

Η αμεσότητα που δημιουργεί στον άνθρωπο η χρήση της ομιλίας και η κατανόηση του λόγου του από τον υπολογιστή έχει κινήσει το ενδιαφέρον των μεγαλύτερων εταιριών του κόσμου. Η Google, η IBM και η Microsoft αποτελούν ορισμένες από τις εταιρίες όπου θέλησαν να ενσωματώσουν τα συστήματα αναγνώρισης ομιλίας στις παροχές τους με σκοπό να κάνουν ακόμα πιο ευέλικτη την χρήση των εφαρμογών τους και να καλύψουν τις αυξανόμενες απαιτήσεις των χρηστών.

Η Google το 2016, κάνει διαθέσιμη σε όλους τους χρήστες της την δυνατότητα του Google voice Recognition API (Application Programming Interface) [47] όπου μέσω αυτού ο χρήστης μπορεί προφορικά να αιτηθεί διαδικασίες στις εφαρμογές της google και εκείνες να κατανοήσουν το αίτημα του.

Εάν αναλογιστούμε τις παροχές της google που υπάρχουν στα κινητά μας τηλεφωνά όπως είναι οι χάρτες της Google, το Gmail, ο browser και το Google Assistance αντιλαμβανόμαστε αμέσως πόσες λειτουργίες μπορούν πλέον να πραγματοποιηθούν χωρίς να χρειαστεί να ακουμπήσουμε την συσκευή μας και να πληκτρολογήσουμε την αναζήτηση μας ή την τοποθεσία προορισμού μας.

Πλέον, το Google voice Recognition API αποτελεί ένα από τα πιο προηγμένα συστήματα speech to text όπου υποστηρίζονται 125 γλώσσες και μορφές API είτε με ανάκτηση αρχείων ήχου είτε με άμεση ομιλία.

Στην πρώτη μορφή, το αρχείο ήχου θα πρέπει να έχει διάρκεια 1 λεπτό και ο χρήστης μπορεί να το έχει αποθηκευμένο στην συσκευή του σαν αρχείο ήχου και να το ανακτήσει όταν χρειαστεί η μετατροπή του σε κείμενο. Σε περίπτωση που το αρχείο είναι αρκετά μεγάλο δίνεται η δυνατότητα αποθήκευσης στο google Cloud και ανάκτηση από εκεί.

Στην δεύτερη μορφή η οποία και χρησιμοποιήθηκε, η ροή δεδομένων είναι άμεση αφού ο χρήστης ενεργοποιώντας το μικρόφωνο μέσω της αντίστοιχης εφαρμογής google που χρησιμοποιεί μπορεί να γνωστοποιήσει το αίτημα που θέλει να καταγράψει το σύστημα.

Η ίδια η google ένα χρόνο μετά το άνοιγμα του Google voice Recognition API στο ευρύ κοινό, δήλωσε ότι χιλιάδες χρήστες το χρησιμοποιήσαν έχοντας πρόσβαση σε περισσότερες παροχές και αυτό αποτέλεσε κίνητρο για την ίδια την εταιρεία να προσθέσει νέα εργαλεία στις παροχές της.

2.5 Αναπαραγωγή φωνής

Ένα σύστημα εκφώνησης κειμένου (text-to-speech,TTS) [48] αποτελεί τον μηχανισμό εκείνο οπου αναλαμβάνει την διαδικασία μετατροπής μιας ακολουθίας αλφαβητικών συμβολών στο αντίστοιχο ακουστικό σήμα ομιλίας.

Η θεμελιώδης διαφορά μεταξύ των συστημάτων TTS και μηχανών ομιλίας όπως π.χ. το ραδιόφωνο έγκειται στο γεγονός ότι στην περίπτωση των TTS απαιτούμε η μηχανή να εκφωνήσει αυτόματα οποιαδήποτε καινούργια πρόταση της δοθεί. Επιπλέον, η βασική διαφορά των TTS συστημάτων από άλλα συστήματα σύνθεσης ομιλίας όπως τα voice response, είναι πως στην είσοδο του γραπτού λόγου τα TTS συστήματα αποθηκεύουν το λεξιλόγιο και την σύνταξη χωρίς περιορισμό, πραγ-

ματοποιώντας γλωσσολογική ανάλυση εις βάθος για την εξαγωγή αποτελέσματος ομιλίας.

Στόχος αυτών των συστημάτων είναι η ποιότητα του αποτελέσματος να καλύπτει την φυσικότητα του προφορικού λόγου και η μηχανή να εκφωνεί οποιαδήποτε σειρά λέξεων και φωνημάτων ζητηθεί χωρίς να αλλοιώνεται το νόημα της πρότασης και η σημασία των λέξεων.

2.5.1 Τρόπος Λειτουργίας

Το γενικό μοντέλο ενός TTS συστήματος απαρτίζεται από το front-end και το back-end τα οποία εκτελούν την γλωσσική ανάλυση του κειμένου και την επεξεργασία του σήματος αντίστοιχα [49].

Πιο συγκεκριμένα, το front-end, αφού έχει εισέλθει το κείμενο στο σύστημα εκτελεί τις πρώτες βασικές αναλύσεις της γλώσσας με σκοπό να απλοποιηθεί η μετέπειτα φωνητική μετατροπή. Σε πρώτη φάση, το κείμενο της εισόδου γράφεται αναλυτικά και ολογράφως συμπεριλαμβάνοντας αριθμούς και σημεία στίξης. Στην συνέχεια κάθε γράμμα αντιστοιχίζεται σε ένα αρχείο ήχου που το αντιπροσωπεύει για την παραγωγή της ηχητικής πληροφορίας.

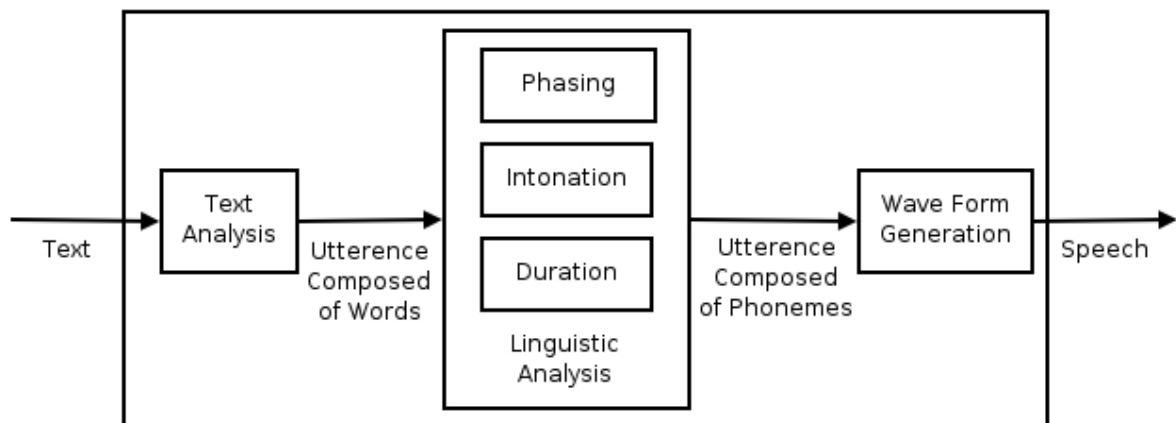
Το back-end, γνωστό και ως συνθέτης, αναλαμβάνει την ακρόαση του κειμένου αφού πρώτα έχει αναπαραστήσει με το εκάστοτε μοντέλο χρήσης της μηχανής τα φωνήματα σε ήχους. Επιπλέον λειτουργία του back-end, είναι να αποδώσει στον ήχο προσωδία, δηλαδή να αποφασίσει για τον τόνο της φωνής που θα ακουστεί το ύφος και την διάρκεια των γραμμάτων.

Οι βασικές κατηγορίες των τεχνολογιών που επικρατούν για τους συνθέτες των TTS συστημάτων διαφοροποιούνται στον τρόπο που επιλεγεί το σύστημα να θέσει σε σειρά τα φωνήματα δημιουργώντας έτσι το αρχείο ήχου που θα εκφωνηθεί.

Στην σύνθεση ομιλίας σε συνένωση μονάδων, το μοντέλο βασίζεται σε προηχογραφημένα κείμενα από πραγματικούς ομιλητές τα οποία τμηματίζονται ανάλογα με το γραπτό λόγο που δόθηκε και ανασυγκολλούνται με νέα σειρά και νόημα αντίστοιχο της εισόδου. Το σημαντικό σε αυτό τον τρόπο σύνθεσης είναι η ανάγκη για ομαλή μετάβαση των λέξεων και η αποφυγή παραμορφωμένης ομιλίας λόγω της μετέπειτα συγκόλλησης.

Στην σύνθεση ομιλίας βασισμένη σε κανόνες, το μοντέλο μελετά τα ανθρωπινά όργανα που συμβάλουν στην παραγωγή φωνής και τα αντιστοιχεί σε μηχανικά συστήματα. Με αυτόν τον τρόπο η στοματική κοιλότητα αντικαθίσταται από χρονικά μεταβαλλόμενα ψηφιακά φίλτρα όπου διεγείρονται με μεταβολές πίεσης του αέρα ακριβώς όπως οι φωνητικές χορδές. Τα μοντέλα αυτά, αν και υπερσχύουν ποιότητα έναντι της συνένωσης μονάδων παράγουν μηχανικό ήχο υστερώντας έτσι στην φυσικότητα που απαιτεί η ομιλία.

Το τελευταίο μοντέλο, η αρθρωτική σύνθεση αποτελεί την προέκταση της ομιλίας βασισμένη σε κανόνες όπου επί του παρόντος θεωρείται ακόμα εργαλείο προς έρευνα και σκοπός του είναι η ο μοντελοποίηση και δημιουργία ενός φωνητικού καναλιού συνάρτηση μηχανικών προσομοιώσεων όλων των φωνητικών οργάνων που μετέχουν στην ομιλία.



Εικόνα. 2.24: Στάδια σύνθεσης ομιλίας

2.5.2 Βιβλιοθήκη pyttsx3

Η pyttsx3 [50] αποτελεί μια βιβλιοθήκη μετατροπής (κείμενο-σε-ομιλία) , ανεπτυγμένη σε Python. Σε αντίθεση με άλλες βιβλιοθήκες λειτουργεί χωρίς την απαίτηση σύνδεσης στο internet καθώς εκμεταλλεύεται τους ήδη εγκατεστημένους εκφωνητές του λειτουργικού συστήματος. Ουσιαστικά πρόκειται για μια συλλογή από μηχανές μετατροπής κειμένου ομιλίας όπως:

- SAPI5 για Windows XP, Windows Vista και Windows 8,8.1, 10
- NSSpeechSynthesizer για Mac OS X 10.5 (Leopard) και 10.6 (Snow Leopard)
- espeak για Ubuntu Desktop Edition 8.10 (Intrepid), 9.04 (Jaunty), και 9.10 (Karmic)

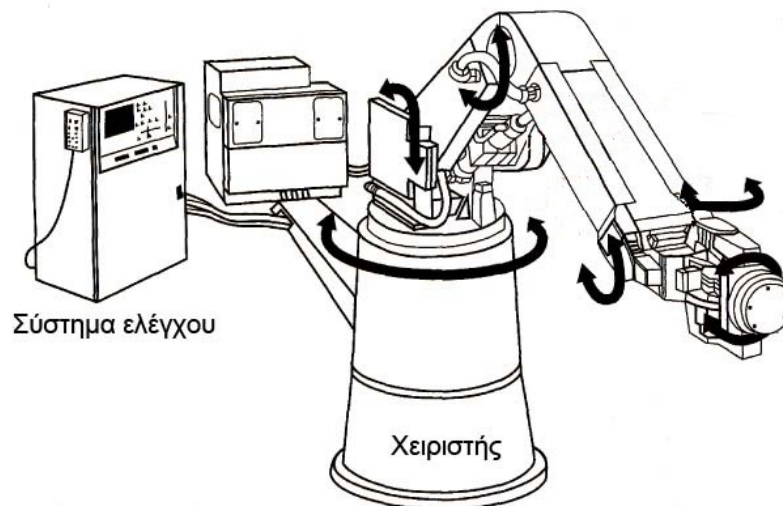
Ανάλογα με το Λειτουργικό Σύστημα στο οποίο εκτελείται, επιλέγεται και η αντίστοιχη συμβατή μηχανή εκφώνησης. Μέσω της βιβλιοθήκη δίνεται η δυνατότητα ο χρήστης να επιλέξει οποία φωνή επιθυμεί ανάμεσα από τις εγκατεστημένες στο σύστημα του φωνές καθώς επίσης να ρυθμίσει την ένταση αναπαραγωγής την ταχύτητα εκφώνησης και τέλος να αποθηκεύσει τη φωνητική εντολή ως αρχείο ήχου.

2.6 Αρχιτεκτονική Ρομποτικού Βραχίονα

Ο ρομποτικός βραχίονας έκανε την εμφάνιση του τη δεκαετία του 1950, όπου με τη σταδιακή ανάπτυξή του όλα αυτά τα χρόνια, κατάφερε να διεισδύσει σε πολλούς τομείς της καθημερινότητας και ιδιαίτερα της βιομηχανίας [51], [52].

Κάποια από τα χαρακτηριστικά, που διαφοροποιούν τους βραχίονες του σήμερα είναι ο βαθμό ελευθερίας, η φέρουσα ικανότητα, η ταχύτητα, η επιτάχυνση, η ακρίβεια κλπ. Αλλά είτε ως μικρά «ρομπότ πάγκου», είτε ως μεγαλύτερα είναι πολύ σημαντικά ως προς την αποτελεσματικότητα της λειτουργίας των διαφόρων μονάδων. Βασικές εφαρμογές της ρομποτικής αποτελούν η βαφή, η συγκόλληση, η συναρμολόγηση, ο ποιοτικός έλεγχος κ.α. Όλα τα παραπάνω, εκτελούνται με ταχύτητα, ακρίβεια και υψηλή αντοχή.

Τα βασικά δομικά στοιχεία, που απαρτίζουν ένα σύγχρονο ρομποτικό βραχίονα είναι η βάση, οι αρθρώσεις, οι σύνδεσμοι, οι ενεργοποιητές και το τελικό εργαλείο δράσης. Η βάση τοποθετείται στον χώρο εργασίας του ρομποτικού βραχίονα και πάνω σε αυτή συνδέονται διαδοχικά τα υπόλοιπα στοιχεία. Οι σύνδεσμοι αποτελούν τα στερεά σώματα του βραχίονα και οι αρθρώσεις τούς μηχανισμούς κίνησης μεταξύ των συνδέσμων. Τέλος, οι ενεργοποιητές είναι υπεύθυνοι για την μηχανική κίνηση του βραχίονα, όπου στην περίπτωση ενός ηλεκτρικού βραχίονα είναι ηλεκτρικοί κινητήρες, όπως για παράδειγμα σερβοκινητήρες. Ακολουθώς, μια ομάδα αρθρώσεων που ονομάζεται «καρπός» είναι υπεύθυνη για την κίνηση του τελικού εργαλείου δράσης. Ο βαθμός ελευθερίας ενός ρομποτικού βραχίονα ισούται με τον αριθμό των αρθρώσεων που διαθέτει. Στον καρπό μπορούν να προσαρμοστούν τα εργαλεία που απαιτούνται, ανάλογα τη φύση της εργασία που εκτελεί ο βραχίονας (π.χ. εργαλείο ηλεκτροσυγκόλλησης, βαφής ή αρπαγής). Η αρπάγη μπορεί να εκτελεστεί με διάφορες μορφές όπως για παράδειγμα με χρήση ηλεκτρομαγνήτη, βεντούζας αναρρόφησης ή δαγκάνας κ.α.



Εικόνα. 2.25: Τυπική μορφή βιομηχανικού βραχίονα (Cincinnati MilacronT3)

Τα μέλη του βραχίονα αριθμούνται. Στην ακίνητη βάση του αντιστοιχεί ο αριθμός 0. Το πρώτο κινούμενο μέλος έχει τον αριθμό 1. Η αρίθμηση συνεχίζεται μ' αυτή τη λογική, μέχρι το ελεύθερο άκρο του βραχίονα, που φέρει τον αριθμό n. Για την τοποθέτηση ενός αποληκτῆριου επενεργητή στον χώρο, απαιτούνται κατ' ελάχιστο έξι αρθρώσεις. Ένας τυπικός βραχίονας έχει πέντε ή έξι αρθρώσεις. Μερικά ρομπότ δεν είναι τόσο απλά όσο μια απλή κινηματική αλυσίδα, δηλαδή έχουν συνδέσμους σε σχήμα παραλληλογράμμου ή άλλες κλειστές κινηματικές δομές.

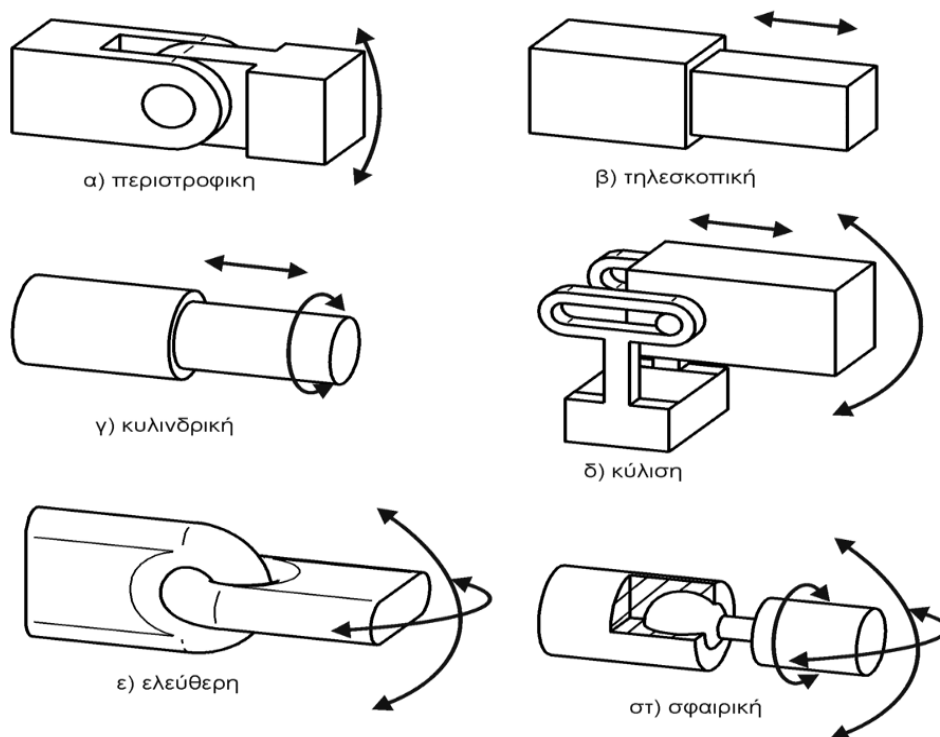
2.6.1 Είδη Αρθρώσεων

Όπως έχει ήδη σημειωθεί, ένα από τα βασικά στοιχεία ενός ρομποτικού βραχίονα αποτελούν οι σύνδεσμοι, δηλαδή διαδοχικά συνδεδεμένα στερεά σώματα. Οι σύνδεσμοι στο σύνολο τους, δημιουργούν μια κινηματική αλυσίδα, συνδεδεμένοι μεταξύ τους ανά δύο με αρθρώσεις. Οι βραχίονες δυο αρθρώσεων έχουν την δυνατότητα κίνησης στο δυσδιάστατο χώρο, ενώ οι βραχίονες με τρεις αρθρώσεις κινούνται στον τρισδιάστατο. Η προσθήκη επιπλέον αρθρώσεων στον βραχίονα αφορά τον προσανατολισμό του αντικειμένου στο χώρο, ενώ ταυτόχρονα διατηρείται η τρισδιάστατη κίνηση. Οι αρθρώσεις μπορούν να κατηγοριοποιηθούν σε στροφικές και πρισματικές (τηλεσκοπικές). Η ύπαρξη και άλλων ειδών αρθρώσεων βασίζεται στον συνδυασμό των παραπάνω κατηγοριών. Ενδεικτικά είδη είναι οι κυλινδρικές, οι ελεύθερες, οι σφαιρικές και οι αρθρώσεις κύλισης.

- Η περιστροφική άρθρωση είναι άρθρωση που επιτρέπει σχετική στροφή μεταξύ δύο γειτονικών συνδέσμων. Δίνει ένα βαθμό ελευθερίας, αφήνοντας το σώμα να περιστραφεί σε ένα επίπεδο και αποκόπτει κάθε άλλη δυνατότητα κίνησης
- Η πρισματική (ή τηλεσκοπική) άρθρωση είναι άρθρωση που επιτρέπει σχετική μετατόπιση (σε ευθεία γραμμή) μεταξύ δύο γειτονικών συνδέσμων. Δίνει και αυτή ένα βαθμό ελευθερίας, αφήνοντας το σώμα να μετατοπίζεται στη διεύθυνση ενός από τους άξονες και αποκόπτει κάθε άλλη δυνατότητα κίνησης
- Σύνθετες αρθρώσεις είναι αυτές που αναλύονται γεωμετρικά σε υπέρθεση δύο ή περισσότερων από τις βασικές αρθρώσεις (στροφική και πρισματική) από έναν βαθμό κινητικότητας

- Η κυλινδρική άρθρωση δίνει δύο βαθμούς ελευθερίας, αφήνοντας μία μεταφορική κίνηση στη διεύθυνση ενός άξονα και μία περιστροφική γύρω από τον άξονα αυτό
- Η άρθρωση της κύλισης δίνει και αυτή δύο βαθμούς ελευθερίας, δηλαδή μία μεταφορική και μία περιστροφική κίνηση, αλλά σε αυτή την περίπτωση ο άξονας της περιστροφικής κίνησης είναι κάθετος στη διεύθυνση του άξονα που πραγματοποιείται η μεταφορική κίνηση.
- Η ελεύθερη άρθρωση δίνει δύο βαθμούς ελευθερίας, αφήνοντας δύο περιστροφικές κινήσεις και εμποδίζοντας όλες τις υπόλοιπες
- Η σφαιρική άρθρωση δίνει τρεις βαθμούς ελευθερίας, αφήνοντας και τις τρεις περιστροφικές κινήσεις ελεύθερες και εμποδίζοντας όλες τις μεταφορικές

Τα σχήματα που ακολουθούν αναπαριστούν τα είδη των αρθρώσεων.



Εικόνα. 2.26: Τα είδη των αρθρώσεων

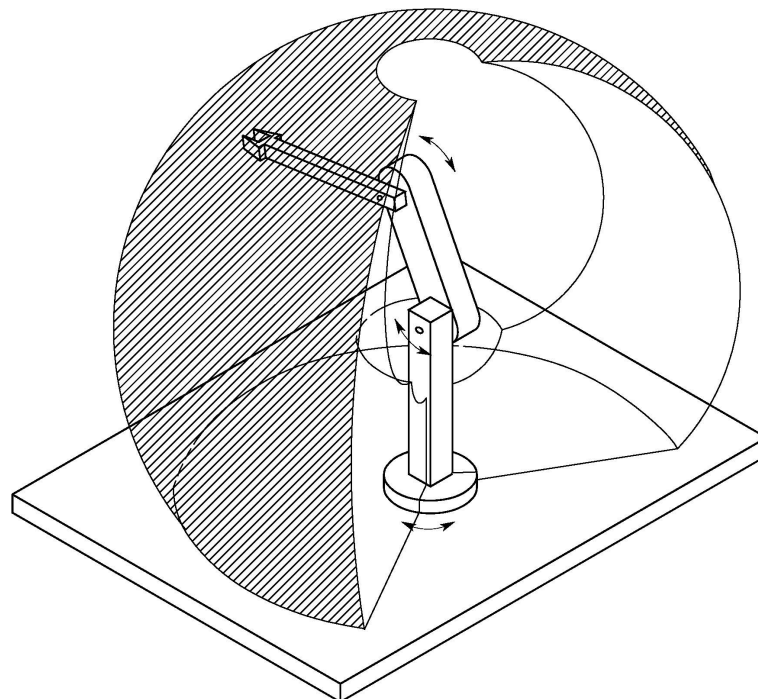
2.6.2 Βαθμοί ελευθερίας και κινητικότητα

Ως βαθμός ελευθερίας (Degrees Of Freedom) (DOF) ορίζεται ο αριθμός των ανεξάρτητων μεταβλητών θέσεων ενός ρομποτικού βραχίονα. Ο βαθμός ελευθερίας για κάθε σχηματισμό θα πρέπει να είναι γνωστός και μονοσήμαντα ορισμένος, έτσι ώστε να είναι εφικτός ο προσδιορισμός της θέσης των τμημάτων που τη συνιστούν. Πρόκειται για μια γενική αρχή, που εφαρμόζεται σε όλους τους μηχανισμούς. Ένα αντικείμενο λέγεται ότι έχει n DOF εάν η διαμόρφωση του μπορεί να καθοριστεί στο ελάχιστο από τις παραμέτρους n . Κατά συνέπεια, ο αριθμός DOF είναι ίσος με τη διάσταση του χώρου διαμόρφωσης. Για ένα ρομποτικό χειριστή, ο αριθμός των αρθρώσεων καθορίζει τον αριθμό DOF. Ορισμένες εφαρμογές, όπως η προσέγγιση γύρω η πίσω από εμπόδια, μπορούν να απαιτήσουν περισσότερους από έξι DOF. Ένας χειριστής που έχει περισσότερους από έξι DOF αναφέρεται ως κινηματικά περιττός.

Σε ένα ρομποτικό βραχίονα, επιπλέον των βαθμών ελευθερίας, υπάρχουν και οι βαθμοί κινητικότητας. Οι βαθμοί κινητικότητας ταυτίζονται σε πλήθος με τις αρθρώσεις του ρομπότ, ενώ το είδος των εργασιών που πραγματοποιεί το ρομπότ καθορίζουν το βαθμό ελευθερίας του. Κατά συνέπεια, δεν είναι απαραίτητο να ταυτίζεται ο βαθμός ελευθερίας με το βαθμό κινητικότητας, σε όλες τις εργασίες. Ωστόσο, υπάρχουν και διεργασίες όπου το πλήθος των βαθμών είναι ίσο μεταξύ τους, για παράδειγμα ένα άκαμπτο αντικείμενο σε τρισδιάστατο χώρο έχει έξι DOF, τρεις για τον προσδιορισμό θέσης και τρεις για τον προσανατολισμό. Επομένως, ένας χειριστής πρέπει τυπικά να κατέχει τουλάχιστον έξι ανεξάρτητους DOF, δηλαδή να αποτελείται από έξι αρθρώσεις και κατά συνέπεια ίσους βαθμούς κινητικότητας. Με λιγότερους από έξι DOF ο βραχίονας δε μπορεί να φθάσει κάθε σημείο στο χώρο εργασίας του με τυχαίο προσανατολισμό.

2.6.3 Χώρος Εργασίας

Ο χώρος εργασίας ενός χειριστή είναι ο συνολικός χώρος σάρωσης από το τελικό εργαλείο δράσης καθώς ο χειριστής εκτελεί όλες τις δυνατές κινήσεις. Ο χώρος εργασίας περιορίζεται από τη γεωμετρία του χειριστή τόσο όσο είναι οι μηχανικοί περιορισμοί στις αρθρώσεις και το μέγεθος των συνδέσμων. Για παράδειγμα, μια περιστροφική άρθρωση μπορεί να οριοθετεί σε λιγότερο από μια πλήρη κίνηση 360°. Επίσης, ο χώρος εργασίας ενός ρομπότ διακρίνεται στον προσβάσιμο και στον επιδέξιο χώρο εργασίας. Προσβάσιμος χώρος εργασίας είναι ο χώρος όπου το τελικό εργαλείο δράσης του βραχίονα μπορεί να έχει πρόσβαση κατά έναν τουλάχιστον προσανατολισμό της κινηματικής αλυσίδας. Ενώ ο επιδέξιος χώρος εργασίας, είναι υποσύνολο του προσβάσιμου και αποτελεί το γεωμετρικό τόπο που είναι προσβάσιμος από το τελικό εργαλείο δράσης, σε οποιοδήποτε σημείο της κινηματικής αλυσίδας. Αξίζει να σημειωθεί ότι, ο επιδέξιος χώρος εργασίας υφίσταται μόνο σε χειριστές από 3 βαθμούς κινητικότητας ή ελευθερίας και πάνω, οι οποίοι μπορούν να έχουν διαφορετικούς προσανατολισμούς.



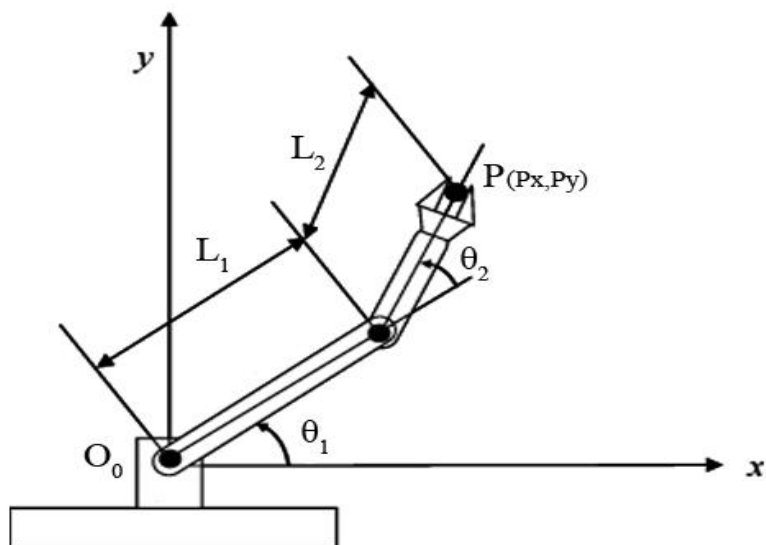
Εικόνα. 2.27: Σχηματική απεικόνιση χώρου εργασίας

2.6.4 Κινηματική Ανάλυση

Η κινηματική ασχολείται με την αναλυτική μελέτη της κίνησης του ρομπότ, με γεωμετρική προσέγγιση στο χώρο, χωρίς να μελετά τις αίτιες που ευθύνονται για αυτήν. Τα δυο βασικά προβλήματα που θα μελετηθούν παρακάτω και αφορούν την κινηματική των βραχιόνων είναι το ευθύ και το αντίστροφο κινηματικό πρόβλημα. Οποιαδήποτε διαμόρφωση των μελών του αρθρωτού βραχίονα στο χώρο, παράγει το σημείο θέσης και τον προσανατολισμό του. Τα δυο αυτά μεγέθη, μαζί με τις κινήσεις των αρθρώσεων, αποτελούν τα μετρήσιμα και τα ελέγξιμα μεγέθη αντίστοιχα που πρέπει να προσδιορισθούν. Στον προσδιορισμό της θέσης αντιστοιχεί το σημείο που βρίσκεται το άκρο του βραχίονα και για τον προσδιορισμό του προσανατολισμού χρειάζεται η κατεύθυνση του τελευταίου συνδέσμου ή του τελικού εργαλείου δράσης. Όπως ορίστηκαν οι έννοιες, διαπιστώνεται πως η θέση μπορεί να προσδιοριστεί εύκολα σε οποιαδήποτε από τις τρεις διαστάσεις, ενώ αντίθετα ο προσανατολισμός και η περιστροφή περιπλέκονται.

Ευθύ κινηματικό πρόβλημα

Το ευθύ κινηματικό πρόβλημα είναι υπεύθυνο για τον καθορισμό της θέσης και του προσανατολισμού του τελικού εργαλείου δράσης, έχοντας ως μεταβλητές, για τον προσδιορισμό αυτό, τις κινήσεις, μηχανικούς περιορισμούς των αρθρώσεων και το μέγεθος των συνδέσμων. Ένα παράδειγμα κίνησης, που θα μπορούσε να δοθεί ως δεδομένο στο κινηματικό πρόβλημα, είναι οι γωνίες που σχηματίζουν οι περιστροφικές αρθρώσεις του βραχίονα και το μήκος των συνδέσμων, όπως παρουσιάζονται στον βραχίονα 2DOF του παρακάτω σχήματος.



Εικόνα. 2.28: Σχηματική απεικόνιση βραχίονα 2 DOF

Συνεπώς, οι σχέσεις που προκύπτουν για την ευθεία κινηματική ανάλυση του παραπάνω βραχίονα είναι οι εξής:

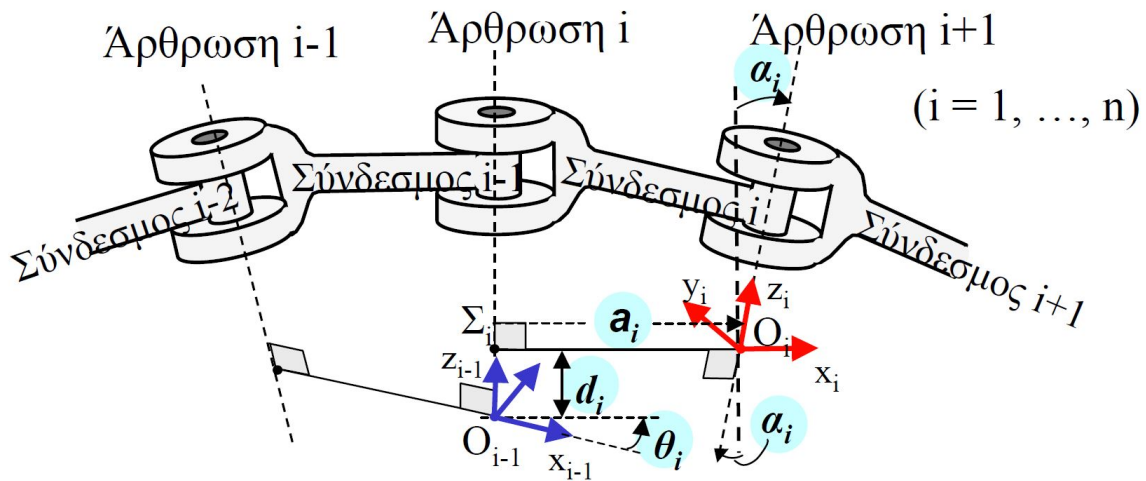
$$P_x = L_1 \cos \theta_1 + L_2 \cos (\theta_1 + \theta_2) \quad (2.13)$$

$$P_y = L_1 \sin \theta_1 + L_2 \sin (\theta_1 + \theta_2) \quad (2.14)$$

Σύμφωνα με όσα έχουν αναπτυχθεί παραπάνω, η δομή του βραχίονα περιλαμβάνει διαδοχικούς συνδέσμους συνδεδεμένους με αρθρώσεις από την βάση του έως το τελικό εργαλείο δράσης του ρομπότ. Για την επίλυση του κινηματικού προβλήματος, ένα από τα κύρια κινηματικά μοντέλα που χρησιμοποιείται είναι η μέθοδος Denavit και Hartenberg [53]. Στο μοντέλο αυτό, γίνεται χρήση τεσσάρων παραμέτρων για τον ορισμό του κινηματικού προβλήματος για ένα ρομπότ με n αρθρώσεις. Η εφαρμογή ενός συστήματος συντεταγμένων για κάθε άρθρωση, καθορίζει τις D-H παραμέτρους και εκτελείται ο αλγόριθμος που παρουσιάζεται παρακάτω. Η παραπάνω διαδικασία, καταλήγει στον προσδιορισμό της θέσης και του προσανατολισμού του τελικού εργαλείου δράσης ως προς το ακίνητο σύστημα συντεταγμένων.

Αλγόριθμος Denavit – Hartenberg

Σύμφωνα με τον αλγόριθμο Denavit - Hartenberg (D-H), σε κάθε άρθρωση του βραχίονα αντιστοιχεί ένα σύστημα ορθοκανονικών και δεξιόστροφων συντεταγμένων. Η μέθοδος αυτή είναι σημαντική για τον καθορισμό του σημείου θέσης και των προσανατολισμό δύο διαδοχικών συνδέσμων.



Εικόνα. 2.29: Σύστημα συντεταγμένων κατά Denavit-Hartenberg

Στη συνέχεια, ακολουθεί ο προσδιορισμός των τεσσάρων παραμέτρων της σύμβασης D-H, οι οποίες είναι:

- θ_i η γωνία μεταξύ του άξονα x_{i-1} και της κοινής καθέτου $\Sigma_i O_i$
- d_i η απόσταση O_{i-1} και Σ_i (μετατόπιση κατά μήκος του z_{i-1} – άρθρωση i)
- a_i το μήκος της κοινής καθέτου $\Sigma_i O_i$ (άρθρωση $i \leftrightarrow$ άρθρωση $i+1$)
- α_i η γωνία μεταξύ του άξονα z_{i-1} και z_i (στροφή γύρω από τον άξονα x_i)

Εφόσον έχουν ορισθεί τα πλαίσια των αρθρώσεων και οι παράμετροι της σύμβασης D-H, ακολουθώντας τα παρακάτω βήματα, μπορούν να ορισθούν η θέση και ο προσανατολισμός του πλαισίου i ως προς το πλαίσιο $i-1$.

- **Βήμα 1:** Περιστροφή του πλαισίου $i-1$ γύρω από τον άξονα Z_{i+1} κατά γωνία θ_i
- **Βήμα 2:** Μετατόπιση d_i του πλαισίου $i-1$ κατά μήκος του άξονα Z_{i-1}

- **Βήμα 3:** Μετατόπιση a_i (μήκος της κοινής καθέτου) κατά το νέο (στραφέντα) άξονα x_{i-1} (που τώρα συμπίπτει με τον x_i)
- **Βήμα 4:** Περιστροφή γύρω από τον άξονα x_i κατά γωνία α_i

Συνεπώς από το βήμα 1 και βήμα 2 προκύπτει η μήτρα μετασχηματισμού:

$$A_{\Sigma_i}^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.15)$$

Από το βήμα 3 και 4 προκύπτει:

$$A_i^{\Sigma_i} = \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & \cos a_i & -\sin a_i & 0 \\ 0 & \sin a_i & \cos a_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.16)$$

Συνολικά, η μήτρα μετασχηματισμού η οποία δίνει τη σχετική θέση και τον προσανατολισμό του πλαισίου I ως προς το πλαίσιο i-1, είναι η ακόλουθη:

$$A_i^{i-1} = A_{\Sigma_i}^{i-1} * A_i^{\Sigma_i} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos a_i & \sin\theta_i \sin a_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos a_i & -\cos\theta_i \sin a_i & a_i \sin\theta_i \\ 0 & \sin a_i & \cos a_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.17)$$

Στον παραπάνω πίνακα, κάθε σειρά περιέχει όλες τις απαραίτητες παραμέτρους για την δημιουργία των ομογενών μετασχηματισμών, μεταξύ των πλαισίων των γειτονικών συνδέσμων

- $q_i = \theta_i$ για περιστροφική άρθρωση
- $q_i = d_i$ για πρισματική άρθρωση
- a_i και α_i ορίζονται από τη γεωμετρία του συνδέσμου και είναι σταθερές

Σύνδεσμος i	a_i	α_i	d_i	θ_i
1	a_1	0	0	q_1
2	a_2	-90°	0	q_2
3	0	0	d_3	q_3

Αντίστροφο κινηματικό πρόβλημα

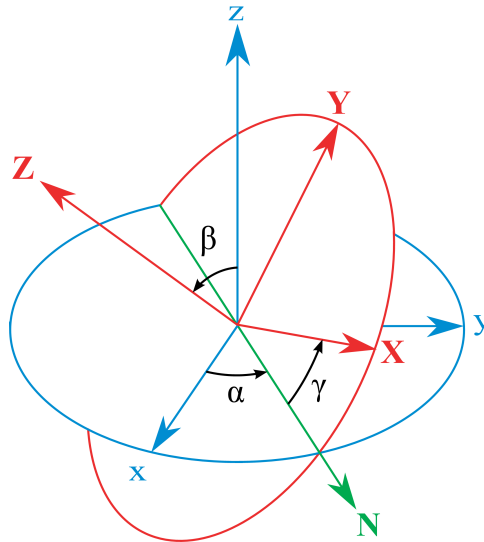
Στην ρομποτική όμως το βάρος πέφτει στην αντίστροφη κινηματική και όχι στην ευθύγραμμη. Το αντίστροφο κινηματικό πρόβλημα των σειριακών χειριστών μελετάται για πολλές δεκαετίες και είναι απαραίτητο στον έλεγχο των χειριστών λόγο ότι μας ενδιαφέρει να δίνουμε τη θέση και το προσανατολισμό του σημείου στο οποίο θέλουμε να πάει το τελικό εργαλείο δράσης και να υπολογίζονται οι κλίσεις του κάθε μέλους του βραχίονα. Η επίλυση του αντίστροφου κινηματικού προβλήματος είναι υπολογιστικά κοστοβόρα και γενικά απαιτείται αρκετός χρόνος στον έλεγχο των χειριστών σε πραγματικό χρόνο. Αρχικά πρέπει να βρεθεί ο χώρος εργασίας, δηλαδή να οριστούν τα όρια κλίσης του κάθε μέλους του βραχίονα για τη σωστή λειτουργία του και στη συνέχεια να βρεθούν όλοι οι συνδυασμοί λύσεων. Ένας βραχίονας έχει τη δυνατότητα να προσεγγίσει το τελικό σημείο με διαφορετικούς συνδυασμούς και συνήθως όσο περισσότερα μέλη υπάρχουν τόσο περισσότεροι είναι οι συνδυασμοί. Τέλος η σωστή επιλογή συνδυασμού ορίζεται από τον μηχανικό και συνήθως επιλέγεται ο συνδυασμός που υλοποιείται με τη λιγότερη δυνατή κίνηση του βραχίονα. Υπάρχουν δύο προσεγγίσεις λύσεων η γεωμετρική που χωρίζει το βραχίονα σε μικρά γεωμετρικά σχήματα και προσπαθεί να βρει εξισώσεις σχετικά με τις γωνίες των μελών του με τη βοήθεια τριγωνομετρίας και επιπεδομετρίας και η αλγεβρική που μετασχηματίζει τις εξισώσεις της ευθύγραμμης κινηματικής με τέτοιο τρόπο που να έχουμε σαν αγνώστους τις γωνίες των μελών του βραχίονα.

Μηχανισμός Προσανατολισμού

Οι περιστροφές ορίζονται από μία ή περισσότερες διαδοχικές περιστροφές γύρω από έναν από τους τρεις ορθογωνικούς άξονες ενός πλαισίου συντεταγμένων. Ο Ελβετός μαθηματικός Leonhard Euler (1707 - 1783) [54] έδωσε το θεώρημα περιστροφής του Euler το οποίο λέει ότι οποιαδήποτε περιστροφή μπορεί να αναλυθεί σε τρεις επιμέρους περιστροφές ως προς τους τρεις άξονες συντεταγμένων, με την προϋπόθεση ότι δεν θα γίνουν δύο συνεχόμενες περιστροφές ως προς τον ίδιο άξονα. Με άλλα λόγια, εάν έχουμε μία αυθαίρετη περιστροφή αυτή θα μπορούσε να αναλυθεί σε μία περιστροφή ως προς τον άξονα z , μία περιστροφή ως προς τον άξονα y και μία περιστροφή ως προς τον άξονα x .

Υπάρχουν δώδεκα σύνολα τριών διαδοχικών περιστροφών που ικανοποιούν την απαίτηση του Euler, χωρίς δύο διαδοχικές περιστροφές να έχουν τον ίδιο άξονα (XYX, XZX, YXY, YZY, ZXZ, ZYZ, XYZ, XZY, YZX, YXZ, ZXY, ZYX). Όλες αυτές οι ακολουθίες μπορούν να αναφέρονται σε γωνίες Euler, αλλά στην πράξη, μόνο δύο χρησιμοποιούνται ευρέως.

Η ακολουθία περιστροφής **ZYZ** είναι η εφαρμοζόμενη ακολουθία περιστροφής όταν αναφέρονται στις γωνίες Euler. Χρησιμοποιείται συχνά στους τομείς της αεροναυτικής και του μηχανικού σχεδιασμού. Επίσης η ακολουθία περιστροφής **XYZ** είναι εξίσου γνωστή και ονομάζεται *roll - pitch - yaw* ή απλά RPY . Η ακολουθία χρησιμοποιείται κυρίως για να περιγράψει τη στάση των κινούμενων μηχανών όπως πλοία, αεροσκάφη, αυτοκίνητα και ρομπότ.



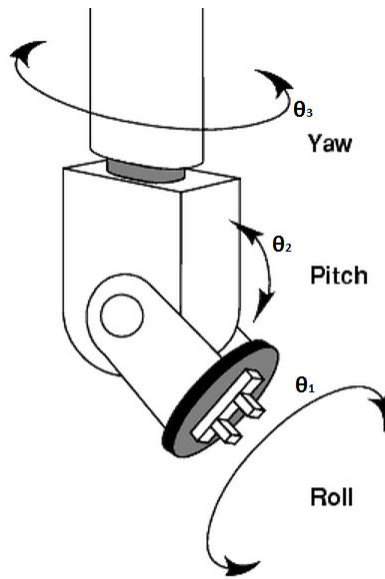
Εικόνα. 2.30: Ο προσανατολισμός του στερεωμένου στο σώμα συστήματος συντεταγμένων ως προς το σταθερό στο χώρο σύστημα συντεταγμένων περιγράφεται από τρεις γωνίες

Ακολουθώς θα αναλυθούν οι περιστροφές του καρπού του ρομπότ χρησιμοποιώντας την ακολουθία RPY

Ο άξονας X θεωρείται ότι είναι η μπροστινή κατεύθυνση του μηχανήματος. Ο άξονας Z συνήθως προσανατολίζεται προς την ανοδική κατεύθυνση. Ωστόσο, με τα αεροσκάφη ο άξονας Z συχνά βρίσκεται προς τα κάτω.

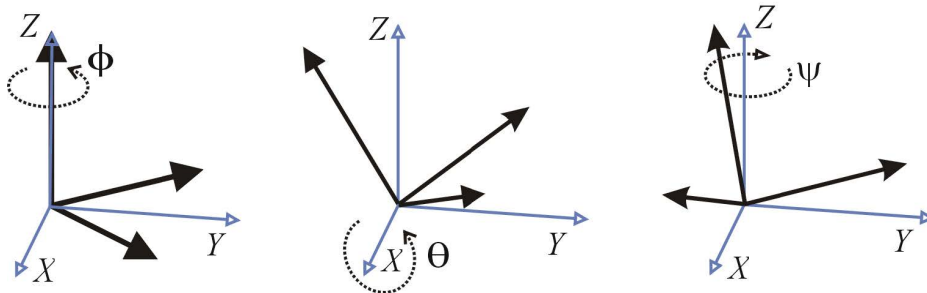
- Η περιστροφή γύρω από τον άξονα X ονομάζεται *roll*
- Η περιστροφή γύρω από τον άξονα Y ονομάζεται *pitch*
- Η περιστροφή γύρω από τον άξονα Z ονομάζεται *yaw*

Στην εικόνα 2.32 περιγράφεται ο προσανατολισμός με τη βοήθεια των γωνιών RPY. Οι πρώτες δύο γωνίες θ_1 (roll) και θ_2 (pitch) είναι ίδιες με τις πρώτες δύο γωνίες Euler, ενώ η τρίτη γωνία περιστροφής θ_3 (yaw) είναι γύρω από τον τρέχοντα άξονα z.



Εικόνα. 2.31: Γωνίες Roll Pitch Yaw

Ο προσανατολισμός στις δύο παραπάνω περιπτώσεις γίνεται με διαδοχικές περιστροφές ως προς τους νέους περιστραμμένους άξονες . Ωστόσο, ο προσανατολισμός μπορεί να δοθεί και με διαδοχικές περιστροφές ως προς ένα σταθερό σύστημα, όπως είναι στην περίπτωση του βιομηχανικού ρομπότ το σύστημα της βάσης.



Εικόνα. 2.32: Γωνίες Roll Pitch Yaw

2.6.5 Περιγραφή βραχίονα Niryo-One

Ο ρομποτικός βραχίονας που χρησιμοποιήθηκε για τις ανάγκες τις παρούσας διπλωματικής εργασίας είναι το Niryo-One της Γαλλικής εταιρείας Niryo [55]. Ο βραχίονας βρίσκεται στο Εργαστήριο Ρομποτικής Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας με έδρα τη Κοζάνη. Το Niryo-One αποτελεί

το πρώτο και το μοναδικό 3D εκτυπωμένο ρομποτικό βραχίονα 6 βαθμών ελευθερίας που προσφέρει η εταιρεία. Το project δημιουργίας του Niryo-One προτάθηκε στο ευρύ κοινό στο Kickstarter το 2017 και κέρδισε 153 συνεισφέροντες συγκεντρώνοντας \$80.756 από τα \$ 20.000 που απαιτούνται για την εκκίνηση του συλλογικού και εκπαιδευτικού ρομπότ. Τα αρχεία STL για την εκτύπωση των μελών του ρομπότ και ο πηγαίος κώδικας διατίθενται ελεύθερα στο κοινό. Η κοινοτική πτυχή αυτού του ρομπότ είναι ένα ουσιαστικό μέρος της εταιρείας, βασικός σκοπός της οποίας είναι να κάνει τη ρομποτική πιο προσβάσιμη, προσφέροντας χαμηλού κόστους και εύκολα στη χρήση ρομπότ. Ακόμα, επιδιώκει να δημιουργήσει ένα πλήρες σύνολο υπηρεσιών γύρω από το Niryo-One και μια κοινότητα γύρω από έργα ανοιχτού κώδικα. Το συγκεκριμένο μοντέλο προορίζεται για εκπαιδευτικούς σκοπούς, εισαγωγή στο industry 4.0, επαγγελματική εξάσκηση καθώς και σε ερευνητικά εργαστήρια.



Εικόνα. 2.33: Ρομποτικός Βραχίονας Niryo-One

Υπάρχουν πολλοί τρόποι προγραμματισμού του Nirygo-One (από υψηλό σε χαμηλό επίπεδο):

- Προγραμματισμός του ρομπότ με τη λειτουργία εκμάθησης στην οποία ο χρήστης μπορεί να μετακινεί το ρομπότ σε όποια θέση ή θέσεις επιθυμεί και να επιλέγει ποιες θα εκτελέσει το ρομπότ. Ακόμα προσφέρεται η δυνατότητα οπτικού προγραμματισμού (παρόμοιο με Scratch) κάνοντας το προσβάσιμο σε άτομα χωρίς προγραμματιστικές γνώσεις
- Χρήση χειριστηρίου Xbox για την απευθείας κίνηση του ρομπότ
- Προγραμματισμός του ρομπότ σε γλώσσα Python κάνοντας χρήση του Python API που προσφέρεται
- Σύνδεση του ρομπότ μέσω των ψηφιακών διεπαφών στο πίσω μέρος για επικοινωνία με άλλες συσκευές όπως Arduino και Raspberry Pi
- Τέλος διατίθεται ελεύθερα ο πηγαίος κώδικας ROS για την τροποποίηση και παραμετροποίηση του στις ανάγκες του χρήστη χρησιμοποιώντας Python ή C++

2.6.6 Τεχνικά χαρακτηριστικά

Το Niryo-One μπορεί να κινηθεί στα επίπεδα x , y και z , καθώς και σε τρεις άξονες τοποθέτησης του καρπού χρησιμοποιώντας κινήσεις roll, pitch και yaw. Αυτό σημαίνει ότι έχει 6 βαθμούς ελευθερίας.

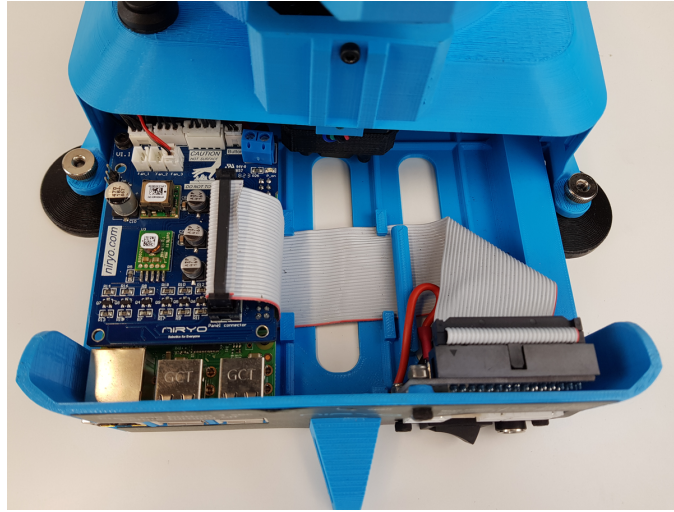
Ο ρομποτικός βραχίονας μπορεί να χωριστεί σε 7 κομμάτια:

1	Βάση (Base)	Πορτοκαλί
2	Όμος (Shoulder)	Κίτρινο
3	Μπράτσο (Arm)	Πράσινο
4	Αγκώνας (Elbow)	Τιρκουάζ
5	Πήχης (Forearm)	Μπλε
6	Καρπός (Wrist)	Μωβ
7	Χέρι (Hand)	Κόκκινο



Εικόνα. 2.34: Ρομποτικός βραχίονας Niryo-One με χρωματισμένα μέλη

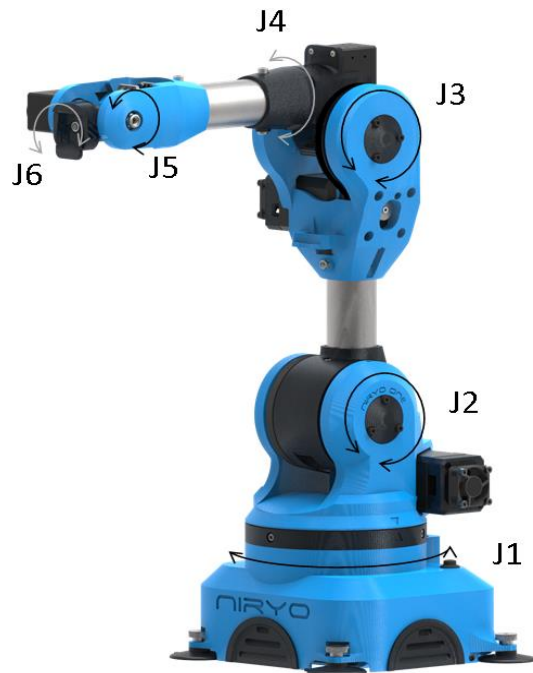
Στη περιοχή της βάσης ενσωματώνεται η μονάδα ελέγχου και επεξεργασίας του βραχίονα η οποία αποτελείται από το Raspberry Pi 3 και ενός επεκτάσιμου shield που είναι υπεύθυνο για την οδήγηση των ενεργοποιητών.



Εικόνα. 2.35: Μονάδα ελέγχου και επεξεργασίας

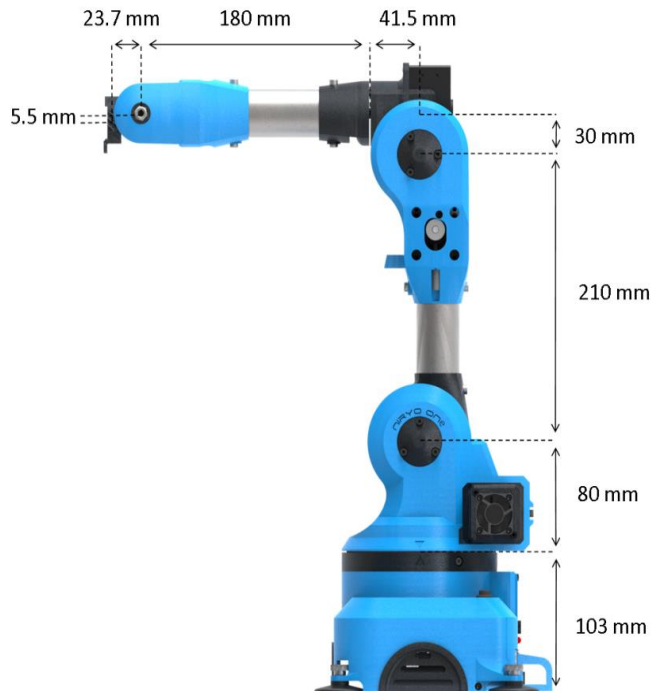
Για την κίνηση των αρθρώσεων χρησιμοποιούνται ειδικοί κινητήρες. Συγκεκριμένα αποτελείται από 3 NiryoSteppers, 2 Dynamixel XL – 430 (σερβοκινητήρας) και 1 Dynamixel XL – 320 (σερβοκινητήρας).

	Ελάχιστο	Μέγιστο
J1	-175°	175°
J2	-90°	36.7°
J3	-80°	90°
J4	-175°	175°
J5	-100°	110°
J6	-145°	147.5°

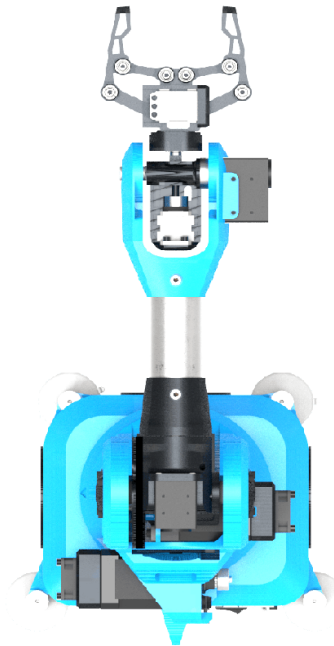


Εικόνα. 2.36: Σχηματική απεικόνιση περιστροφής και ονοματοδοσίας αρθρώσεων

Οι διαστάσεις του βραχίονα είναι κατάλληλες για εκτέλεση εργασιών σε μικρή απόσταση.



Εικόνα. 2.37: Διαστάσεις συνδέσμων



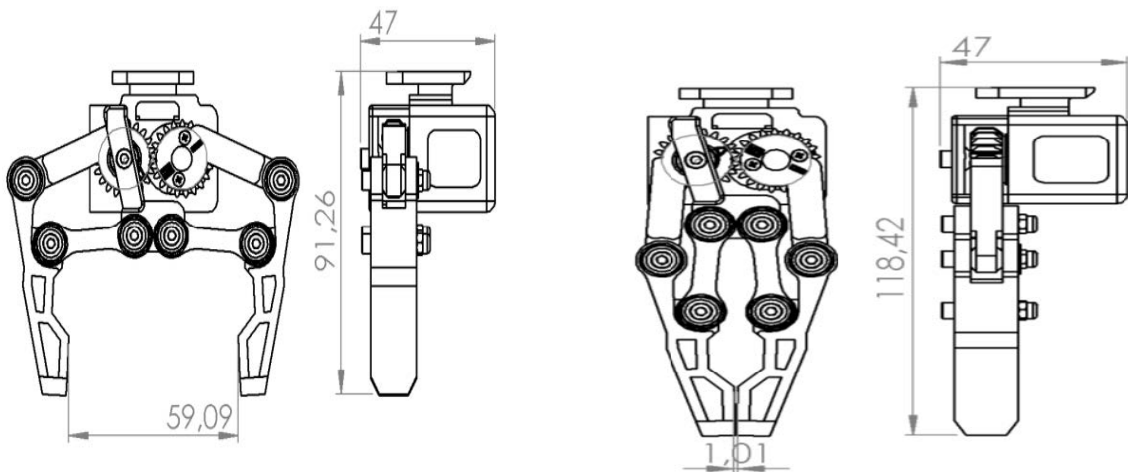
Εικόνα. 2.38: Κάτοψη

Βάρος	3.3kg
Μέγιστη απόσταση αρπαγής	440mm

Όσο αφορά τη περιοχή του καρπού η εταιρεία προσφέρει αρκετές επιλογές προσάρτησης τελικών εργαλείων δράσης. Οι επιλογές συνθέτονται από τρία διαφορετικά εργαλεία αρπαγής GRIPPER, ένα ηλεκτρομαγνήτη καθώς και μία βεντούζας αναρρόφησης. Για τους σκοπούς της παρούσας διπλωματικής εργασίας επιλέχθηκε να προσαρμοστεί το GRIPPER 2 καθώς τόσο ο μηχανισμός σύγκλισης των σιαγόνων όσο και το άνοιγμα τους επιτρέπουν την βέλτιστη σε μέγεθος και εφαρμογή γραπμά καθημερινών αντικειμένων.



Εικόνα. 2.39: Gripper 2



Εικόνα. 2.40: Κάτοψη Gripper 2 (σε πλήρη έκταση)

Εικόνα. 2.41: Κάτοψη Gripper 2 (σε σύγκλιση)

2.7 Σύνοψη κεφαλαίου

Στο συγκεκριμένο κεφάλαιο παρουσιάστηκαν και αναλύθηκαν οι τεχνολογίες όλων των λειτουργιών που αναπτύχθηκαν στο υλικό και λογισμικό μέρος της παρούσας διπλωματικής εργασίας. Ακολουθεί το επόμενο κεφάλαιο στο οποίο παρουσιάζεται η σχεδίαση του συστήματος και ο τρόπος υλοποίησης και ενσωμάτωσης των τεχνολογιών.

Κεφάλαιο 3

Σχεδίαση και υλοποίηση λογισμικού

Στο παρόν κεφάλαιο γίνεται διεξοδική αναφορά στην αρχιτεκτονική, τον τρόπο λειτουργίας του προγράμματος καθώς και στους αλγόριθμους και τις τεχνικές που αναπτύχθηκαν σε κάθε τμήμα του ξεχωριστά. Στόχος του προγράμματος είναι η ανίχνευση και ο εντοπισμός αντικειμένων βάσει του χρώματος, του σχήματος, του είδους καθώς και της ετικέτας QR code, ώστε να συλλεχθούν και να παραδοθούν από το ρομποτικό βραχίονα στον χρήστη μέσω φωνητικής αλληλεπίδρασης.

3.1 Γενική περιγραφή λειτουργίας συστήματος

Ξεκινώντας με τη περιγραφή των βημάτων του συστήματος, αρχικά εκφωνείται στα Ελληνικά και ακολούθως στα Αγγλικά ηχητικό μήνυμα που ενημερώνει τον χρήστη για τις διαθέσιμες γλώσσες που μπορεί να επιλέξει για την μετέπειτα φωνητική αλληλεπίδραση. Ταυτόχρονα με την προαναφερθείσα ενέργεια εκτελείται αυτόματη βαθμονόμηση του ρομποτικού βραχίονα η οποία καταλήγει σε μία προκαθορισμένη θέση ετοιμότητας. Σε αυτό το σημείο ο χρήστης μπορεί να επιλέξει να απαντήσει είτε στα Ελληνικά λέγοντας «Ελληνικά» ή «Αγγλικά», είτε στα Αγγλικά λέγοντας «Greek» ή «English». Το σύστημα αφού έχει αναγνωρίσει επιτυχώς τη γλώσσα επικοινωνίας, ανταποκρίνεται εκφωνώντας την επιλογή προς επιβεβαίωση. Στη περίπτωση που δεν έχει επιτευχθεί επιτυχής αναγνώριση μετά από τρεις προσπάθειες, επιλέγεται ως προκαθορισμένη γλώσσα επικοινωνίας τα Ελληνικά. Στη συνέχεια, εκφωνείται μήνυμα που ενημερώνει το χρήστη για τις διαθέσιμες λειτουργίες που προσφέρονται και τις κατάλληλες λέξεις που θα πρέπει να εκφωνηθούν για την ενεργοποίηση τους. Οι λειτουργίες αυτές είναι η αναζήτηση και η απευθείας παράδοση αντικειμένου. Για την ενεργοποίηση της πρώτης ο χρήστης θα πρέπει να

εκφωνήσει τη λέξη «ψάξε» ή «search» ενώ για την δεύτερη θα πρέπει να κατονομάσει το αντικείμενο που επιθυμεί να του παραδοθεί.

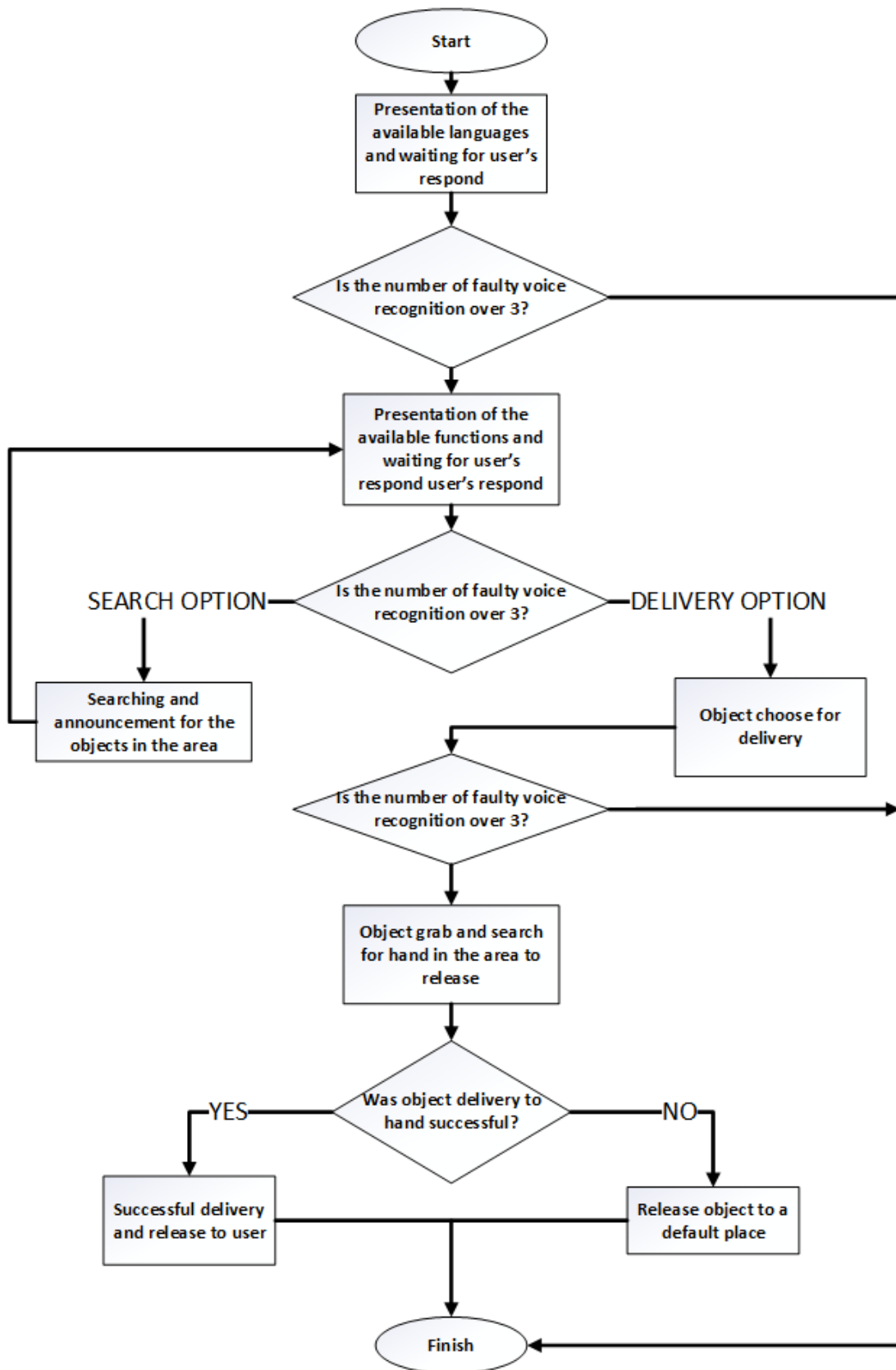
Πιο συγκεκριμένα, κατά την ενεργοποίηση της δεύτερης λειτουργίας δίνεται μια ελευθερία σύνθεσης της πρότασης αρκεί να εμπεριέχονται σε αυτή οι λέξεις «πιάσε» ή «φέρε» για την Ελληνική γλώσσα και «pickup» ή «take» για την Αγγλική. Στην πρόταση οι προαναφερθέντες λέξεις θα πρέπει να συνοδεύονται με το χρώμα και το σχήμα του αντικειμένου στην περίπτωση που ο χρήστης επιλέξει αυτού του είδους την αναγνώριση. Διαφορετικά με το όνομα του αντικειμένου, εάν αυτό υπάρχει στη λίστα των αντικειμένων που αναγνωρίζονται ως προς το είδος του ή με το όνομα που έχει ο ίδιος δώσει να αναγνωρίζεται από την QR code ετικέτα. Παραδείγματα εκφώνησης για την ενεργοποίηση της λειτουργίας απευθείας παράδοσης στα ελληνικά είναι «Φέρε μου τον κόκκινο κύκλο» ή «Πιάσε το κουτάλι» ενώ στα Αγγλικά «Pick up my pills» ή «Take the remote». Να σημειωθεί ότι ο βραχίονας από την αρχή της εκτέλεσης του προγράμματος βρίσκεται σε προκαθορισμένη θέση ετοιμότητας έχοντας στραμμένη την κάμερα στη περιοχή που βρίσκεται ακριβώς μπροστά του.

Στην περίπτωση που επιλεγθεί η λειτουργία της αναζήτησης, ο βραχίονας ξεκινά να εκτελεί τη σάρωση για αντικείμενα που βρίσκονται περιμετρικά του μεταβαίνοντας σε 5 προκαθορισμένες θέσεις ώστε να καλυφθεί όλο το εύρος του χώρου εργασίας. Κατά τη σάρωση, για όλα τα αντικείμενα που αναγνωρίζονται επιτυχώς με τουλάχιστον έναν από τους τρόπους εντοπισμού καταγράφονται οι θέσεις και τα ονόματά τους. Στην περίπτωση που ανήκουν στη λίστα των αντικειμένων που αναγνωρίζονται από τον αλγόριθμο Yolo ή των αντικειμένων που φέρουν QR code, καταγράφεται το αντίστοιχο όνομα ενώ στη περίπτωση που η αναγνώριση γίνεται βάση του σχήματος και του χρώματος καταγράφεται το σχήμα και το χρώμα. Με το τέλος της σάρωσης εκφωνείται μήνυμα που ενημερώνει το χρήστη για το τερματισμό της διαδικασίας και στη συνέχεια εκφωνούνται τα ονόματα των αντικειμένων με τη σειρά που εντοπίστηκαν. Ο χρήστης έχει τη δυνατότητα σε αυτό το σημείο να ζητήσει να επαναληφθεί η διαδικασία ή να προβεί στην εκφώνηση του αντικειμένου που επιθυμεί να του παραδοθεί.

Σε άλλη περίπτωση, που ο χρήστης επιλέξει τη διαδικασία απευθείας παράδοσης και έχει εκφωνήσει το αντικείμενο που επιθυμεί με το όνομά του, ο ρομποτικός βραχίονας ξεκινά να αναζητήσει αυτό το αντικείμενο στη περιοχή γύρω του ακολουθώντας την ίδια διαδικασία που περιεγράφηκε προηγουμένως. Εάν δεν βρεθεί το αντικείμενο, τότε ενημερώνεται ο χρήστης με ανάλογο φωνητικό μήνυμα για την αποτυχία εύρεσής του.

Όταν το αντικείμενο που έχει ζητήσει ο χρήστης να του μεταβιβασθεί έχει εντοπιστεί επιτυχώς, τότε εκτελείται διαδικασία προσέγγισης και ανασήκωσης του αντικειμένου. Ο βραχίονας αφού έχει επιτυχώς ανασηκώσει το αντικείμενο ξεκινά να εκτελεί λειτουργία αναζήτησης εντοπισμού του χεριού του χρήστη, ώστε να το απελευθερώσει στο χέρι του. Η λειτουργία αναζήτησης, όπως και οι προηγούμενες, παραμένει ίδια μόνο που αυτή τη φορά η αναζήτηση σταματά όταν εντοπιστεί το χέρι του χρήστη. Μόλις το εντοπίσει, προσεγγίζει την παλάμη και απελευθερώνει το αντικείμενο. Σε άλλη περίπτωση που δεν εντοπιστεί κάποιο χέρι ο βραχίονας μεταβαίνει σε ένα προκαθορισμένο σημείο και απελευθερώνει εκεί το αντικείμενο. Όταν παραδοθεί το αντικείμενο, εκφωνείται το ανάλογο ηχητικό μήνυμα που ενημερώνει το χρήστη για το αποτέλεσμα της διαδικασίας παράδοσης. Στο τέλος αυτής, ο χρήστης έχει τη δυνατότητα να επιλέξει μια από τις διαθέσιμες λειτουργίες που προσφέρονται ή να τερματίσει τη διαδικασία εκφωνώντας τη λέξη «σταμάτα» ή «stop». Να σημειωθεί ότι υπάρχει η δυνατότητα να τερματιστεί η εκτέλεση του προγράμματος όταν το σύστημα αναμένει φωνητική εντολή από το χρήστη.

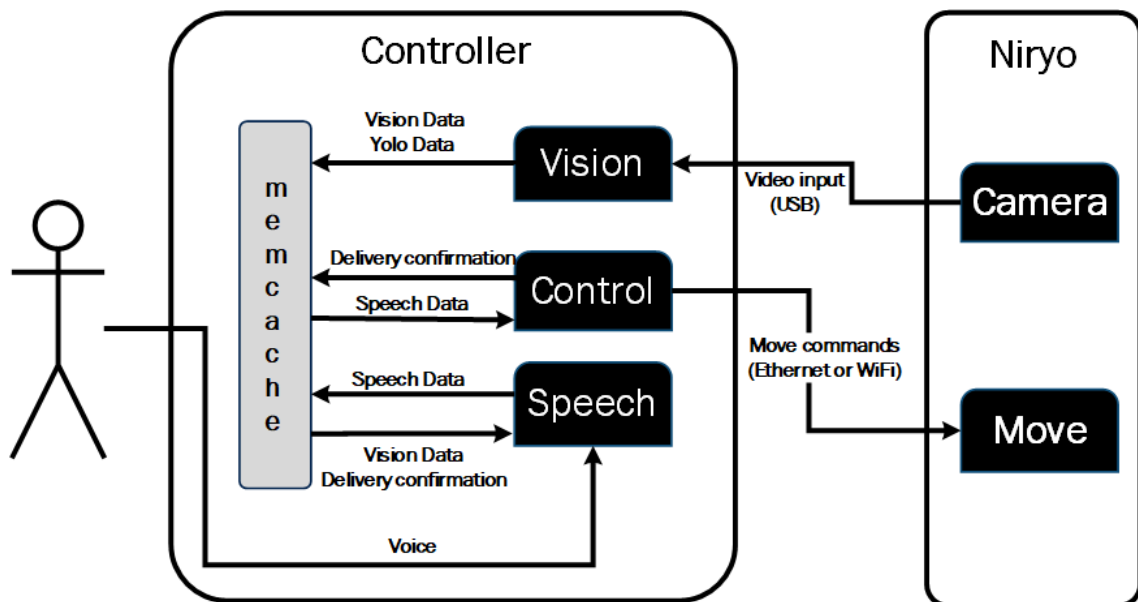
Για να γίνει πιο κατανοητός ο τρόπος με τον οποίο λειτουργεί το σύστημα στο σύνολο του, αρκεί κανείς να δει το παρακάτω διάγραμμα ροής.



Εικόνα. 3.1: Διάγραμμα ροής συστήματος

Για την δυνατόν βέλτιστη επίτευξη των αναγκών του συστήματος και λόγω του γεγονότος ότι οι διεργασίες θα πρέπει να εκτελούνται ασύγχρονα, το λογισμικό που αναπτύχθηκε έχει οργανωθεί σε τρία διαφορετικά τμήματα κώδικα. Το κάθε ένα από αυτά εκτελεί μια ξεχωριστή λειτουργία και ο συνδυασμός αυτών παράγει το τελικό αποτέλεσμα. Πιο συγκεκριμένα, τα βασικά στοιχεία που περιλαμβάνει το παρόν σύστημα μπορούν να διαιρεθούν σε τρία επιμέρους κομμάτια, της μηχανικής όρασης (Vision), της φωνητικής αλληλεπίδρασης (Speech) και τέλος το κομμάτι που είναι υπεύθυνο για την διαχείριση των κινήσεων του βραχίονα (Control). Πρέπει να υπογραμμισθεί ότι όσα περιγράφονται σε αυτό το κεφάλαιο, έχουν προκύψει ύστερα από ενδελεχή έρευνα και συνεχόμενους πειραματισμούς για την επίτευξη των βέλτιστων επιδόσεων και δεν αποτελούν έτοιμο προγραμματιστικό μέρος καμιάς εργασίας.

Στο παρακάτω διάγραμμα αρχιτεκτονικής απεικονίζονται τα σενάρια κώδικα με τις διεργασίες που εκτελούνται σε αυτές, σε συνδυασμό με τις ροές ανταλλαγής δεδομένων και πληροφοριών του συστήματος.



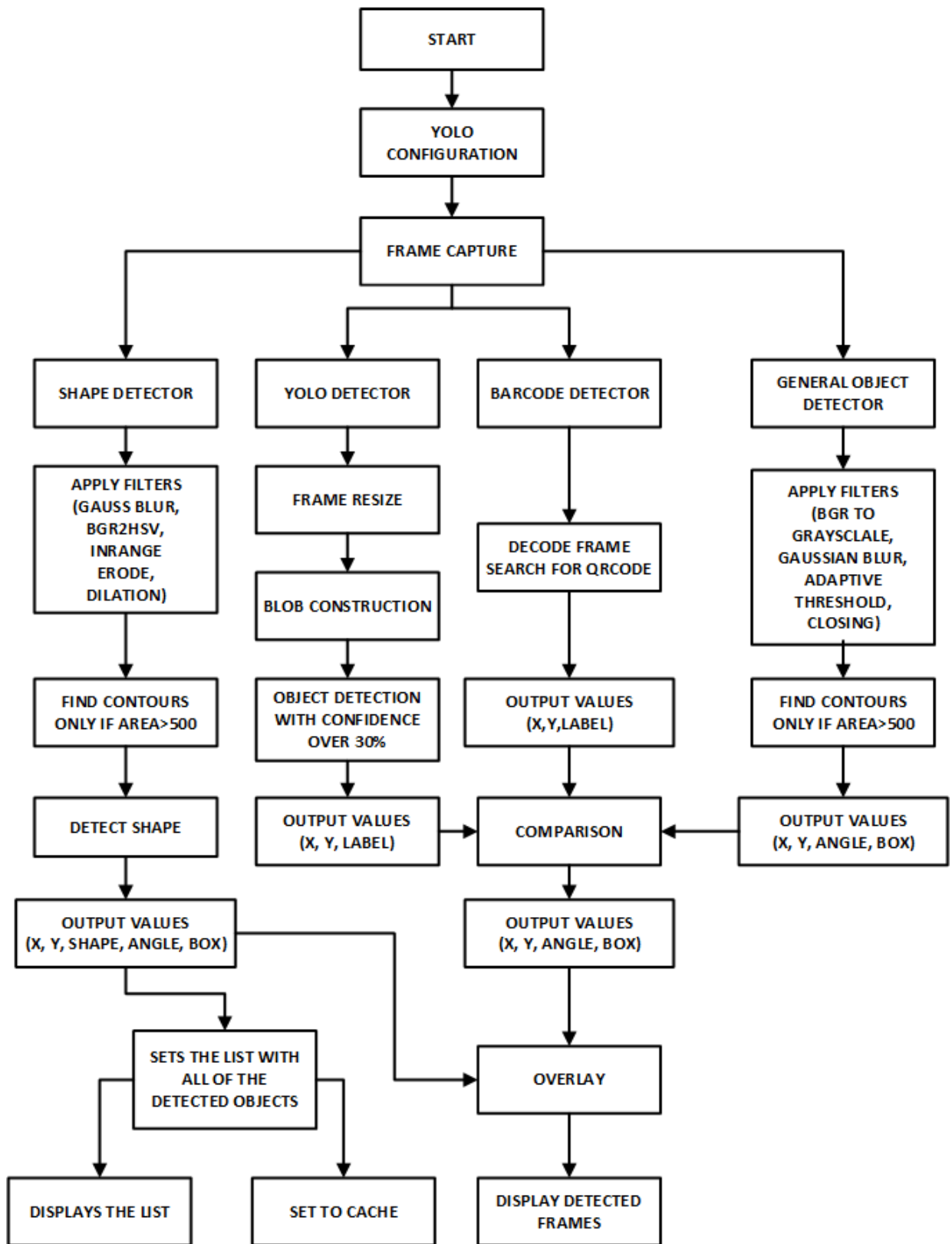
Εικόνα. 3.2: Αρχιτεκτονική συστήματος

Στις επόμενες ενότητες αναλύονται διεξοδικά οι διεργασίες που εκτελούνται σε κάθε τμήμα του κώδικα καθώς και ο τρόπος με τον οποίο ανταλλάσσουν δεδομένα και πληροφορίες.

3.2 Όραση μηχανής

Σε αυτή την ενότητα θα παρουσιαστεί και θα αναλυθεί λεπτομερώς η πιο σημαντική λειτουργία του συστήματος, αυτή της όρασης μηχανής. Η συνεχόμενη επεξεργασία των καρέ του video εισόδου σε συνδυασμό με την ορθή αναγνώριση και εντοπισμό των αντικειμένων, την ορίζουν ως μία νευραλγικής σημασίας λειτουργία του συστήματος. Το γεγονός αυτό οδήγησε να αποδειχθεί ιδιαίτερα απαιτητική τόσο στον σχεδιασμό, την παραμετροποίηση και την υλοποίηση, όσο και στις απαιτήσεις για επεξεργαστική ισχύς.

Προκειμένου να υλοποιηθεί η εν λόγω λειτουργία αναπτύχθηκε σενάριο κώδικα (Vision) στο οποίο ενσωματώνονται και εκτελούνται όλοι οι διαθέσιμοι τρόποι εντοπισμού και αναγνώρισης αντικειμένων. Στο σενάριο αυτό η είσοδος προέρχεται από τη usb κάμερα που βρίσκεται προσαρτημένη στη περιοχή του τελικού εργαλείου δράσης, σε μορφή ροής video. Στο κυρίως τμήμα εκτελείται το σύνολο των ειδών εντοπισμού από τις αντίστοιχες συναρτήσεις και η εισαγωγή των χαρακτηριστικών όλων των εντοπισθέντων αντικειμένων (ετικέτα, συντεταγμένες, μοίρες γωνίας) σε μία λίστα. Αυτή η λίστα ανανεώνεται στο σύνολο της σε ζωντανό χρόνο, έτσι ώστε τα αντικείμενα που υπάρχουν σε αυτή να ανταποκρίνονται στα τρέχοντα αντικείμενα που βρίσκεται στραμμένη η κάμερα. Το γεγονός αυτό φαίνεται ιδιαίτερα χρήσιμο για δύο λόγους, πρώτων διότι δίνεται η δυνατότητα ενημέρωσης για κάθε χρονική στιγμή των αντικειμένων που εντοπίζονται και δεύτερον αποτελούν τμήμα των δεδομένων που διαμοιράζεται με τα υπόλοιπα σενάρια κώδικα. Τέλος, έξοδο του σεναρίου αποτελεί η εμφάνιση ισάριθμων παραθύρων με τα είδη εντοπισμού, όπου στο κάθε ένα από αυτά το αντικείμενο που έχει εντοπιστεί περικλείεται σε ένα χρωματιστό πλαίσιο εμφανίζοντας όλα τα χαρακτηριστικά του (ετικέτα, συντεταγμένες, μοίρες γωνίας).



Εικόνα. 3.3: Διάγραμμα ροής μηχανικής όρασης

3.2.1 Φίλτρα εικόνας

Η ανίχνευση ενός αντικειμένου σε ένα χώρο αποτελεί μία από τις σημαντικότερες προκλήσεις που συναντώνται κατά την ανάπτυξη ενός προγράμματος στον τομέα της μηχανικής όρασης. Η ανίχνευση στατικών αντικειμένων αποτελεί προϊόν κατάτμησης και μορφολογικών χειρισμών μίας εικόνας. Κάθε φορά που αναπτύσσεται ένα πρόγραμμα ανίχνευσης ενός κινούμενου ή στατικού αντικειμένου σε μια εικόνα, θα πρέπει να λαμβάνονται υπόψη οι παράγοντες του περιβάλλοντος που μπορούν να αλλοιώσουν τα στοιχεία της εικόνας και να κάνουν πιο δύσκολη τη διαδικασία παραγωγής ορθών αποτελεσμάτων. Αξίζει να σημειωθεί ότι κάθε εφαρμογή ανίχνευσης που χρησιμοποιεί εικόνες ή βίντεο ως είσοδο προς επεξεργασία, έχει διαφορετικές ανάγκες και χρήζει διαφορετικής μεταχείρισης και παραμετροποίησης.

Προκειμένου όμως, να εξαλειφθούν τυχόν οπτικές αλλοιώσεις που δύναται να εμφανιστούν κατά τη διαδικασία της δημιουργία, επεξεργασίας και αποθήκευσης μιας ψηφιακής εικόνας εφαρμόζονται τα κατάλληλα φίλτρα. Οι αλλοιώσεις αυτές, υποβαθμίζουν την ποιότητα της εικόνας και συνήθως οφείλονται σε διάφορα είδη θορύβου που προέρχονται από:

- Το φως (ένταση, θερμοκρασία, γωνία φωτισμού)
- Τη θέση του αντικειμένου
- Το υλικό κατασκευής του αντικειμένου
- Τη λανθασμένη εστίαση της κάμερας

Εάν παραμείνουν, το αποτέλεσμα του εντοπισμού και της αναγνώρισης είτε θα είναι λανθασμένο είτε θα έχει αποτύχει εντελώς. Σκοπός των φίλτρων είναι η απομάκρυνση όσο το δυνατόν περισσότερου θορύβου και η ανάδειξη των χαρακτηριστικών που είναι χρήσιμα για την απομόνωση του αντικειμένου από το παρασκήνιο.

Οι τεχνικές βελτιστοποίησης που χρησιμοποιούνται για την απομάκρυνση των αλλοιώσεων, αποτελούν κατηγορία των μορφολογικών χειριστών (morphological operators). Στην κατηγορία αυτή ανήκουν: η διάβρωση (erosion), το φίλτρο διαστολής (dilation), η διαβάθμιση (gradient), το άνοιγμα (opening), το κλείσιμο (closing)

κ.α. Πριν όμως την εφαρμογή των μορφολογικών χειριστών, εφαρμόζονται φίλτρα εξομάλυνση της εικόνας όπως η κατωφλίωση (thresholding), το Γκαουσιανό θόλωμα (Gaussian blur) κ.α. Εκτός από την απομάκρυνση του θορύβου σκοπός των τεχνικών αυτών είναι η βελτιστοποίηση της οπτικής εμφάνισης μιας εικόνας, έτσι όπως την αντιλαμβάνεται το ανθρώπινο μάτι. Επιπροσθέτως, γεμίζουν τα κενά της εικόνας με σκοπό να μην υπάρχουν «λευκά» διαστήματα ανάμεσα στα pixel που αποτελούν το περίγραμμά (contour) τους.

Οι παρακάτω τεχνικές που αναλύονται αποτελούν μέρος των εργαλείων που χρησιμοποιήθηκαν στη παρούσα εργασία. Συγκεκριμένα, εφαρμόστηκαν στη λειτουργία εντοπισμού αντικειμένων βάση του χρώματος και του σχήματος καθώς και για τον εντοπισμό γενικού τύπου αντικειμένων. Η λειτουργία εντοπισμού γενικού τύπου αντικειμένων, αναπτύχθηκε έτσι ώστε να είναι δυνατή η εύρεση της γωνίας θέσης των αντικειμένων που ανήκουν στην λίστα που αναγνωρίζονται από τον αλγόριθμο Yolo, καθώς και των αντικειμένων που φέρουν ετικέτα QR code.

Φίλτρο διάβρωσης εικόνας (erosion)

Η διάβρωση είναι ένας από τους δύο βασικούς τελεστές στον τομέα της μορφολογικών χειριστών, ο άλλος είναι η διαστολή. Συνήθως εφαρμόζεται σε δυαδικές εικόνες, αλλά υπάρχουν εκδόσεις που λειτουργούν σε εικόνες κλίμακας του γκρι.

Η βασική ιδέα της διάβρωσης είναι η αφαίρεση των νέων στοιχείων που εισάγονται στην εικόνα και ανήκουν σε άλλα αντικείμενα, κρατώντας πάντα το λευκό σε πρώτο πλάνο. Ουσιαστικά, διαβρώνει τα όρια του αντικειμένου προσκηνίου προσπαθώντας πάντα να διατηρήσει το προσκηνίο σε λευκό. Αρχικά χρειάζεται δύο εισόδους, την αρχική εικόνα και δομικό στοιχείο ή πυρήνας (kernel). Έπειτα ο πυρήνας σαρώνει την εικόνα pixel προς pixel και εκτελείται η ακόλουθη μεταβολή τιμών:

Ένα pixel στην αρχική εικόνα είτε φέρει την τιμή 1 είτε 0, θα πάρει την τιμή 1 μόνο εάν όλα τα pixels κάτω από το kernel είναι 1, αλλιώς γίνεται μηδέν (διαβρώνεται).

Κατά την παραπάνω διαδικασία, όλα τα pixels κοντά στο σύνορο θα απορρίπτονται ανάλογα με το μέγεθος του πυρήνα. Έτσι, το πάχος ή το μέγεθος του αντικειμένου στο προσκήνιο μειώνεται, η αλλιώς μειώνεται η λευκή περιοχή στην εικόνα. Η διαδικασία αυτή είναι χρήσιμη για την αφαίρεση μικρής έντασης θορύβου που εμφανίζεται σε μορφή λευκών κηλίδων.



Εικόνα. 3.4: Φίλτρο διάβρωσης εικόνας

Φίλτρο διαστολής εικόνας (dilation)

Η τεχνική αυτή είναι η αντίστροφη της διάβρωσης. Εδώ ένα εικονοστοιχείο έχει την τιμή 1 μόνο στην περίπτωση που τουλάχιστον ένα pixel στην περιοχή του πυρήνα έχει την τιμή 1. Έτσι αυξάνεται η λευκή περιοχή στην εικόνα ή το μέγεθος του αντικειμένου προσκηνίου. Στις περισσότερες περιπτώσεις χρησιμοποιείται πρώτα η τεχνική της διάβρωσης και μετά της διαστολής. Αυτό συμβαίνει διότι η διάβρωση θα καταφέρει να αφαιρέσει το θόρυβο αλλά ταυτόχρονα θα μικραίνει το μέγεθος του αντικειμένου. Για να εξισοροποιηθεί αυτό, χρησιμοποιείται η τεχνική της διαστολής για να αυξηθεί το αντικείμενο και να ενωθούν τυχόν ασυνεχή τμήματά του.



Εικόνα. 3.5: Φίλτρο διαστολής εικόνας

Φίλτρο κλεισίματος εικόνας (closing)

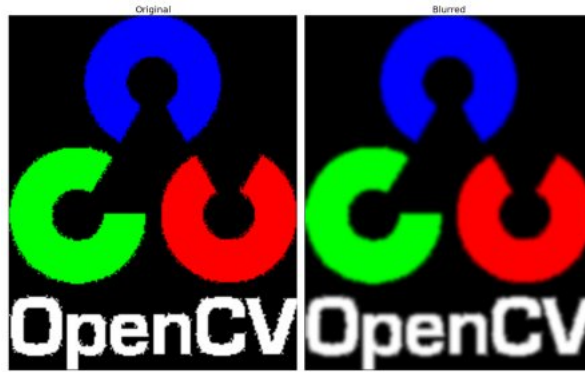
Το φίλτρο κλεισίματος είναι χρήσιμο στο κλείσιμο μικρών οπών μέσα σε αντικείμενα προσκηνίου ή μικρών μαύρων σημείων στο αντικείμενο. Φαίνεται ιδιαίτερα χρήσιμο όταν οι συνθήκες φωτισμού είναι χαμηλές και δημιουργείται ψηφιακός θόρυβος από τις συσκευές καταγραφής, καθώς και στην περίπτωση που δημιουργείται σκιά των αντικειμένων λόγω κακής γωνίας φωτισμού.



Εικόνα. 3.6: Φίλτρο κλεισίματος εικόνας

Γκαουσιανό θόλωμα (Gaussian blur)

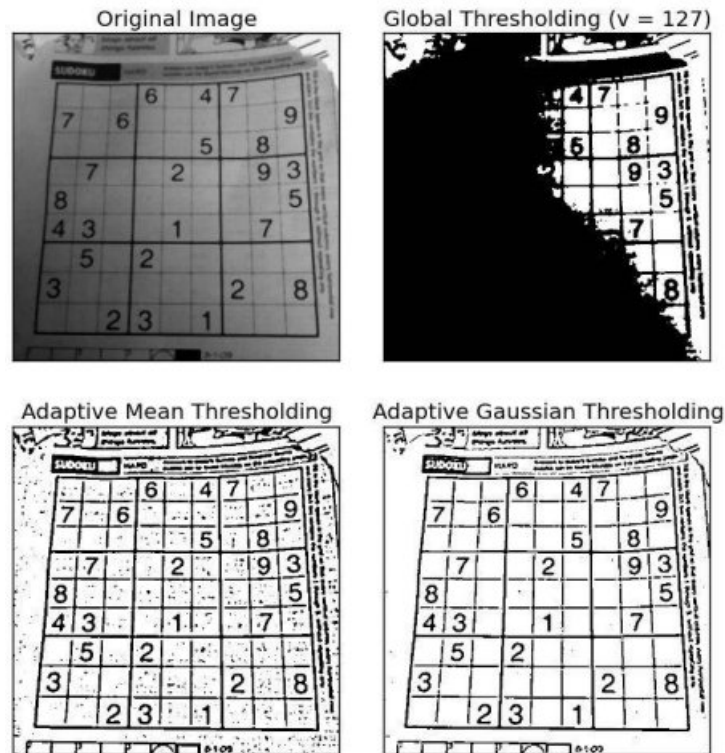
Σε αυτήν τη μέθοδο, αντί για ένα φίλτρο πλαισίου, χρησιμοποιείται ένα Γκαουσιανό kernel. Αρχικά θα πρέπει να καθοριστεί το πλάτος και το ύψος του kernel που πρέπει να είναι θετικό και περιττό. Πρέπει επίσης να καθοριστεί η τυπική απόκλιση στις κατευθύνσεις x και y , σ_X και σ_Y αντίστοιχα. Εάν έχει οριστεί μόνο το σ_X , το σ_Y λαμβάνεται το ίδιο με το σ_X . Εάν και τα δύο δοθούν ως μηδενικά, υπολογίζονται από το μέγεθος του πυρήνα. Το θόλωμα Γκάους είναι πολύ αποτελεσματικό στην απομάκρυνση του θορύβου Γκάους από μια εικόνα και αποτελεί από τα πρώτα βήματα εφαρμογής φίλτρου σε μια εικόνα προς επεξεργασία, έτσι ώστε να εφαρμοστούν καλύτερα οι μορφολογικοί χειρισμοί που θα ακολουθήσουν.



Εικόνα. 3.7: Γκαουσιανό θόλωμα

Thresholding (κατωφλίωση)

Πρόκειται για τον καθορισμό μιας κλίμακας τιμών φωτεινότητας (κατώφλι) των εικονοστοιχείων στην εικόνα προς επεξεργασία. Για κάθε εικονοστοιχείο, εφαρμόζεται η ίδια τιμή κατωφλίου. Εάν η τιμή φωτεινότητας των εικονοστοιχείων είναι μικρότερη από το κατώφλι, τότε ορίζεται σε 0, διαφορετικά ορίζεται σε μια μέγιστη τιμή. Ένας τρόπος επιλογής του threshold είναι με οπτική επιθεώρηση του ιστογράμματος της εικόνας. Μια άλλη μέθοδος επιλογής τιμής του threshold είναι μέσω δοκιμής και σφάλματος. Στην περίπτωση αυτή επιλέγονται διάφορες τιμές του threshold έως ότου κάποια παράγει καλά αποτελέσματα με βάση δεδομένα κριτήρια του παρατηρητή. Ωστόσο, αυτό μπορεί να μην είναι θεμιτό σε όλες τις περιπτώσεις, π.χ. εάν μια εικόνα έχει διαφορετικές συνθήκες φωτισμού σε διαφορετικές περιοχές. Σε αυτήν την περίπτωση, το προσαρμοστικό (adaptive) κατώφλι μπορεί να βοηθήσει. Εδώ, ο αλγόριθμος καθορίζει το κατώτατο όριο για ένα pixel με βάση μια μικρή περιοχή γύρω από αυτό. Προκύπτουν λοιπόν διαφορετικά όρια για διαφορετικές περιοχές της ίδιας εικόνας που δίνουν καλύτερα αποτελέσματα για εικόνες με διαφορετικό φωτισμό.



Εικόνα. 3.8: Φίλτρο κατωφλίωσης

3.2.2 Εντοπισμός και αναγνώριση σχήματος και χρώματος αντικειμένων

Η λειτουργία εντοπισμού του σχήματος και χρώματος αντικειμένων εκτελείται πρώτη κατά σειρά στο σενάριο κώδικα της μηχανικής όρασης. Αρχικά να υπογραμμιστεί ότι ο αλγόριθμος που αναπτύχθηκε γι' αυτήν τη λειτουργία μπορεί να τροποποιηθεί ώστε να αναγνωρίζει οποιοδήποτε χρώμα. Στην παρούσα υλοποίηση επιλέχθηκε να ανιχνεύονται 3 χρώματα το κόκκινο, το πράσινο και το μπλε και όσο αφορά το σχήμα να ανιχνεύονται το τετράγωνο, το ορθογώνιο, το τρίγωνο, το πεντάγωνο και ο κύκλος. Η συνάρτηση η οποία υλοποιεί τον εν λόγω αλγόριθμο, δέχεται ως είσοδο το τρέχον στιγμιότυπο από την κάμερα, το οποίο είναι μια εικόνα διαστάσεων 640 x 480 pixels και τις ανώτερες και κατώτερες τιμές ορίων του χρώματος που πρόκειται να ανιχνευθεί. Το πρώτο βήμα που εκτελείται είναι η εφαρμογή Γκαουσιανού φίλτρου θολώματος με χρήση kernel διαστάσεων 11 x 11. Η τιμή του kernel προέκυψε πειραματικά διότι απέδωσε καλύτερο οπτικό αποτέλεσμα συναρτήσει και της κάμερας USB που χρησιμοποιήθηκε. Η επιλογή του Γκαουσιανού θολώματος έγινε βάση του αποτελέσματος που αποφέρει σε συνδυασμό με τους μορφολογικούς χειρισμούς που ακολουθούν. Έπειτα η εικόνα μετατρέπεται από το χρωματικό χώρο

BGR στον χρωματικό χώρο HSV (hue, saturation, value), έτσι ώστε στη συνέχεια να απομονωθεί το επιθυμητό χρώμα σύμφωνα με τις παραμέτρους εισόδου που εισήχθησαν. Ο χρωματικός χώρος HSV προτιμάται σε περιπτώσεις εντοπισμού χρώματος διότι δίνεται η δυνατότητα να οριστεί ένα αποδεκτό εύρος απόχρωσης του χρώματος εντοπισμού. Ακολούθως εφαρμόζονται μορφολογικοί χειρισμοί ξεκινώντας με το φίλτρο διάβρωσης να εφαρμόζεται με διπλό πέρασμα και έπειτα το φίλτρο διαστολής. Αφού ολοκληρωθούν και οι μορφολογικοί χειρισμοί το οπτικό αποτέλεσμα που έχει προκύψει στην εικόνα είναι η εμφάνιση με άσπρο χρώμα μόνο των αντικειμένων που έχουν το χρώμα που ανιχνεύεται. Έχοντας πετύχει την εμφάνιση μόνο των επιθυμητών αντικειμένων, σειρά έχει η εύρεση των περιγραμμάτων. Η εύρεση των περιγραμμάτων έχει ως αποτέλεσμα την δημιουργία μιας λίστας από σημεία διάστατου χώρου που περικλείουν τα αντικείμενα. Ουσιαστικά πρόκειται για την καμπύλη που ενώνει όλα τα συνεχή σημεία κατά μήκος των ορίων διαχωρισμού του αντικειμένου. Από τα σημεία αυτά επιλέγονται εκείνα που περικλείουν περιοχές μεγαλύτερες από έναν αριθμό pixel, τα οποία ανήκουν σε ένα αντικείμενο. Η διάκριση αυτή γίνεται με σκοπό να απορριφθούν πολύ μικρές περιοχές που μπορεί να εμφανίζονται λόγω κάποια σκιάς που θα έχει δημιουργηθεί από κακό φωτισμό. Για τις περιοχές αυτές σχεδιάζεται ένα πλαίσιο που τις περικλείει και ένα ορθογώνιο οριοθέτησης στην ελάχιστη επιφάνειά τους με αποτέλεσμα να υπολογίζεται η γωνία περιστροφής του αντικειμένου. Στα πλαίσια που έχουν σχηματιστεί υπολογίζεται το κέντρο μάζας του αντικειμένου το οποίο θα είναι και οι τελικές συντεταγμένες θέσης του στην εικόνα. Οι τιμές των συντεταγμένων αντικειμένου φανερώνουν εκείνο το pixel στην εικόνα που συμπίπτει στο κέντρο του αντικειμένου.

Για την εύρεση του σχήματος χρησιμοποιείται η λίστα περιγραμμάτων που έχει δημιουργηθεί από πριν. Υπολογίζεται πρώτα η περίμετρος του περιγράμματος και έπειτα μέσω μιας συνάρτησης γίνεται προσέγγιση του αριθμού των ευθύγραμμων τμημάτων που σχηματίζεται από τα σημεία της λίστας περιγραμμάτων. Η προσέγγιση αυτή βασίζεται στην υπόθεση ότι μια καμπύλη μπορεί να προσεγγιστεί από μικρότερα ευθύγραμμα τμήματα. Είναι σημαντικό να γίνει αντιληπτό ότι η λίστα περιγραμμάτων αποτελείται από μια λίστα κορυφών. Ακολούθως ελέγχεται ο αριθμός των καταχωρήσεων σε αυτήν τη λίστα για να προσδιοριστεί το σχήμα ενός αντικει-

μένου. Για παράδειγμα, εάν το κατά προσέγγιση περίγραμμα έχει τρεις κορυφές, τότε πρέπει να είναι ένα τρίγωνο. Εάν ένα περίγραμμα έχει τέσσερις κορυφές, τότε πρέπει να είναι είτε τετράγωνο είτε ορθογώνιο. Για να προσδιοριστεί ποιό από τα δύο είναι, υπολογίζεται η αναλογία διαστάσεων του σχήματος, που είναι το πλάτος του πλαισίου οριοθέτησης του περιγράμματος διαιρούμενο με το ύψος. Εάν ο λόγος διαστάσεων είναι περίπου 1.0 τότε είναι τετράγωνο (αφού όλες οι πλευρές έχουν περίπου ίσο μήκος) αλλιώς το σχήμα είναι ορθογώνιο. Εάν ένα περίγραμμα έχει πέντε κορυφές, τότε θεωρείται ως πεντάγωνο. Διαφορετικά, με τη διαδικασία της εις άτοπον επαγωγής το σχήμα που εξετάζεται είναι ένας κύκλος. Στο τέλος της διαδικασίας στο παράθυρο εντοπισμού των αντικειμένων σχεδιάζεται το πλαίσιο οριοθέτησης συνοδευόμενο από το όνομα του σχήματος, τις συντεταγμένες κέντρου και τη γωνία που σχηματίζεται.

3.2.3 Εντοπισμός και αναγνώριση των αντικειμένων με τον αλγόριθμο Yolo

Η λειτουργία που ακολουθεί στον σενάριο του κώδικα μηχανικής όρασης είναι η αναγνώριση του είδους των αντικειμένων. Όπως έχει προαναφερθεί αυτό επιτυγχάνεται με τη χρήση του αλγόριθμου αναγνώρισης αντικειμένων YOLO v3. Για την εκτέλεσή του δεν απαιτείται κάποιου είδους εγκατάσταση του αλγόριθμου στο σύστημα παρόλα αυτά, θα πρέπει να υπάρχει ένα framework στο οποίο θα εκτελεστεί.

Μερικά από τα ευρύτερα χρησιμοποιηθέντα frameworks που είναι συμβατά με τον αλγόριθμό είναι το Darknet, το Darflow και η OpenCV. Στη παρούσα υλοποίηση, επιλέχθηκε να χρησιμοποιηθεί η OpenCV. Η επιλογή αυτή προέκυψε διότι, η χρήση της είναι απαραίτητη από όλες τις υπόλοιπες λειτουργίες εντοπισμού και αναγνώρισης αντικειμένων που έχουν αναπτυχθεί. Παράλληλα, καλύπτει επαρκώς τις απαιτήσεις του συστήματος όσο αφορά το χρόνο αναγνώρισης αντικειμένων και τέλος υποστηρίζει απευθείας τη Python χωρίς τη ανάγκη ενδιάμεσων Python ports. Έτσι, επιτυγχάνεται η ύπαρξη ενός καθολικού framework για το σενάριο του κώδικα μηχανικής όρασης αποφεύγοντας περαιτέρω εγκατάσταση νέων frameworks αποκλειστικού σκοπού.

Προκειμένου να εκτελεστεί ο αλγόριθμός απαιτείται αρχικά να φορτωθούν τρία αρχεία. Τα αρχεία αυτά είναι το `yolov3.weights`, το `yolov3.cfg` και το `coco.names`. Το

αρχείο **yolov3.weights** αποτελεί το προ-εκπαιδευμένο μοντέλο στο COCO dataset και ουσιαστικά είναι ο πυρήνας του αλγορίθμου YOLO v3. Στο αρχείο **yolov3.cfg** ενσωματώνονται όλες οι απαραίτητες ρυθμίσεις για τον αλγόριθμο και στο αρχείο **coco.names** περιέχονται τα 80 διαφορετικά ονόματα κλάσεων που χρησιμοποιούνται στο COCO dataset. Για τη φόρτωση των αρχείων **yolov3.weights** και **yolov3.cfg** γίνεται χρήση του **dnn module** της OpenCV. Έχει οριστεί ως backend το DNN back-end από την OpenCV και ως μονάδα εκτέλεσης του αλγορίθμου η CPU. Να αναφερθεί ότι θα ήταν δυνατή η εκτέλεση του αλγορίθμου στη GPU εάν δινόταν ως όρισμα στόχου εκτέλεσης το **cv.dnn.DNN_TARGET_OPENCL**. Ωστόσο, η τρέχουσα έκδοση OpenCV έχει δοκιμαστεί μόνο με GPU της Intel και θα αλλάξει αυτόματα σε CPU, εάν δεν εντοπιστεί GPU της Intel. Σε επόμενο βήμα καθορίζονται τα επίπεδα εξόδου που είναι υπεύθυνα για τον προσδιορισμό του αντικειμένου που εντοπίζεται χρησιμοποιώντας τις συναρτήσεις **net.getUnconnectedOutLayers()** και **net.getLayerNames()**.

Αφού έχουν εκτελεστεί όλες οι απαραίτητες εντολές παραμετροποίησης, ακολουθεί η μετατροπή κάθε καρέ εικόνας από τη ροή video σε blob μέσω της συνάρτησης **blobFromImage()** για το νευρωνικό δίκτυο. Το blob είναι μια συλλογή από frames με τις ίδιες διαστάσεις (ύψος, πλάτος) και το ίδιο βάθος (αριθμών χρωματικών καναλιών) που έχουν προ-επεξεργαστεί με τον ίδιο τρόπο. Σε αυτήν τη διαδικασία, κλιμακώνονται οι τιμές εικονοστοιχείων στο εύρος τιμών 0 έως 1 χρησιμοποιώντας συντελεστή κλίμακας 0.00392 (1/255). Επίσης γίνεται υποβάθμιση του μεγέθους της εικόνας σε (320x320) χωρίς περικοπή, το οποίο θα επιτρέψει αύξηση της ταχύτητας αναγνώρισης με σχετική απώλεια ακρίβειας.

Για την αναγνώριση των αντικειμένων εκτελείται η συνάρτηση **forward()**. Ως είσοδος απαιτείται το τελικό επίπεδο το οποίο θα διατρεχθεί από ολόκληρο το δίκτυο. Προκειμένου να προσδιοριστεί το τελευταίο επίπεδο του δικτύου χρησιμοποιείται η συνάρτηση **getUnconnectedOutLayers()** που δίνει τα ονόματα των μη συνδεδεμένων επιπέδων εξόδου, τα οποία είναι ουσιαστικά τα τελευταία επίπεδα του δικτύου. Στη συνέχεια, εκτελείται πέρασμα εμπρός (forward pass) του δικτύου για να ληφθεί έξοδος από τα επίπεδα εξόδου.

Η συνάρτηση **forward()** επιστρέφει τα αποτελέσματα των αναγνωρίσεων με μορφή ένθετης λίστας και περιέχει πληροφορίες για όλα τα αντικείμενα που εντοπίστηκαν. Πιο συγκεκριμένα, περιλαμβάνει τις συντεταγμένες x και y του κέντρου του αντικειμένου που ανιχνεύτηκε, το ύψος και πλάτος του πλαισίου οριοθέτησης, το ποσοστό εμπιστοσύνης και τις βαθμολογίες για όλες τις κλάσεις αντικειμένων που αναφέρονται στο **coco.names**.

Σε αυτό το σημείο η ανίχνευση έχει ολοκληρωθεί και το μόνο που μένει είναι η εμφάνιση των αποτελεσμάτων στην οθόνη.

Στη συνέχεια διατρέχεται η λίστα αποτελεσμάτων έχοντας ως κριτήριο επιλογής τα πλαίσια οριοθέτησης με ποσοστό εμπιστοσύνης άνω του 30%. Αν αυξηθεί αυτό το ποσοστό σημαίνει ότι υπάρχει μεγαλύτερη ακρίβεια στην ανίχνευση, αλλά πιθανόν λιγότερος αριθμός αντικειμένων και αντίστροφα.

Τα κουτιά με εμπιστοσύνη ίση ή μεγαλύτερη από το όριο εμπιστοσύνης που τέθηκε υπόκεινται σε **Non Maximum Suppression (NMS)**. Παρόλο που παραλείφθηκαν τα πλαίσια περιορισμένης εμπιστοσύνης, υπάρχει πιθανότητα να υπάρχουν διπλές ανιχνεύσεις, αλληλεπικαλυπτόμενα πλαίσια οριοθέτησης γύρω από ένα αντικείμενο, περιττά ή ξένα πλαίσια οριοθέτησης. Για να επιδιορθωθούν όλα αυτά τα ζητήματα εφαρμόζεται **NMS**.

Τέλος, στο παράθυρο εντοπισμού των αντικειμένων σχεδιάζεται το πλαίσιο οριοθέτησης ετικέτας κατηγορίας, η βαθμολογία εμπιστοσύνης, οι συντεταγμένες x , y του κέντρου, και ο αριθμός των frames (καρέ) που επεξεργάζεται ο αλγόριθμος στη χρονική διάρκεια ενός δευτερολέπτου (frames per second).

3.2.4 Εντοπισμός και αποκωδικοποίηση QR codes

Η επιλογή για τη χρήση QR codes (2D barcodes) έγινε βάση των πλεονεκτημάτων που προσφέρουν έναντι των γραμμωτών barcodes. Πιο συγκεκριμένα, καταλαμβάνουν μικρότερο χώρο στη επιφάνεια του αντικειμένου καταφέροντας ταυτόχρονα

να μεταφέρουν 80 φορές μεγαλύτερο όγκο και είδος πληροφορίας σε σχέση με τα γραμμωτά barcodes. Επιπρόσθετα, διαθέτουν χαμηλό ποσοστό εσφαλμένης ανάγνωσης και δυνατότητα ανάγνωσης από οποιαδήποτε γωνία και θέση της κάμερας. Ο συνδυασμός όλων αυτών των πλεονεκτημάτων καταφέρνει να εξυπηρετήσει πλήρως τις περιπτώσεις και τους σκοπούς χρήσης της παρούσας υλοποίησης.

Ο εντοπισμός των QR codes βασίζεται στη βιβλιοθήκη Zbar κάνοντας χρήση του Python αρθρώματος που διατίθεται. Για την εκκίνηση της λειτουργίας, εισάγεται ως είσοδος στη συνάρτηση αποκωδικοποίησης `decode()` η ροή βίντεο από την κάμερα. Ως έξοδο η συνάρτηση επιστρέφει ένα πίνακα με τα εντοπισθέντα QR codes. Το γεγονός αυτό επιτρέπει τον εντοπισμό πολλαπλών QR codes για κάθε frame που εισάγεται. Σε επόμενο βήμα διατρέχεται αυτός ο πίνακας έτσι ώστε να εξαχθούν για όλα τα QR codes το πεδίο του ονόματος και τα 4 σημεία στο χώρο που το περιλαμβάνουν. Για την εξαγωγή των σημείων χρησιμοποιείται το αντικείμενο `.rect` και για το πεδίο του ονόματος η συνάρτηση `.decode("utf-8")` που μετατρέπει τον πίνακα byte σε αλφαριθμητικό. Στο τέλος της διαδικασίας στο παράθυρο εντοπισμού των αντικειμένων σχεδιάζεται το πλαίσιο οριοθέτησης με την ετικέτα ονόματος και τις συντεταγμένες κέντρου του QR code.

3.2.5 Εντοπισμός περιγράμματος αντικειμένων

Η λειτουργία εντοπισμού περιγράμματος αντικειμένων αποσκοπεί στον εντοπισμό του περιγράμματος όλων των αντικείμενων που υπάρχουν στο χώρο. Αυτό το είδος εντοπισμού αναπτύχθηκε επικουρικά για τις λειτουργίες εντοπισμού μέσω του αλγορίθμου Yolo και εντοπισμού των QR codes.

Κάθε αντικείμενο που εντοπίζεται επιτυχώς από τις διαθέσιμες λειτουργίες, θα πρέπει να συνοδεύεται από τρία βασικά χαρακτηριστικά, την ετικέτα ονόματος, τις συντεταγμένες του κέντρου και τη γωνία που σχηματίζει σε σχέση με τη εκάστοτε θέση της κάμερας. Τα χαρακτηριστικά της ετικέτας και των συντεταγμένων εξάγονται από όλες τις λειτουργίες εντοπισμού. Το χαρακτηριστικό της γωνίας όμως δεν μπορεί να εξαχθεί ούτε από τον αλγόριθμό Yolo ούτε από τη βιβλιοθήκη Zbar. Δεν

θα ήταν δυνατό να παραλειφθεί αυτό το δεδομένο, καθώς είναι απαραίτητο για την σωστή τοποθέτηση του τελικού εργαλείου δράσης του βραχίονα. Για το λόγο αυτό αναπτύχθηκε η λειτουργία εντοπισμού περιγράμματος αντικειμένων.

Για την υλοποίηση αυτού του είδους εντοπισμού αναπτύχθηκε κατάλληλη συνάρτηση η οποία δέχεται ως είσοδο το τρέχον στιγμιότυπο από την κάμερα. Αρχικά, εφαρμόζεται φίλτρο μετατροπής της εικόνας σε κλίμακα του γκρι και Γκαουσιανού θολώματος με χρήση kernel 13×13 . Η εφαρμογή αυτών των φίλτρων είναι απαραίτητη και προ-απαιτείται για την σωστή λειτουργία των επομένων διεργασιών που θα ακολουθήσουν. Ακολούθως, εφαρμόζονται μορφολογικοί χειρισμοί προσαρμοστικού κατωφλίου και φίλτρου κλεισίματος με διπλό πέρασμα. Με τη εφαρμογή αυτών των φίλτρων επιτυγχάνεται η οπτική απομόνωση των αντικειμένων για διάφορες συνθήκες φωτισμού. Το οπτικό αποτέλεσμα που προκύπτει είναι ο χρωματισμός σε κόκκινο της επιφάνειας των αντικειμένων. Έχοντας επιτευχθεί η απομόνωση όλων των αντικειμένων, σειρά έχει η εύρεση των περιγραμμάτων. Για την εύρεση αυτών ακολουθούνται τα ίδια βήματα που περιγράφηκαν στην ενότητα λειτουργίας εντοπισμού και αναγνώρισης σχήματος και χρώματος αντικειμένων. Αφού υπολογιστεί το περίγραμμα και το ορθογώνιο οριοθέτησης της ελάχιστης επιφάνειάς τους στα πλαίσια που έχουν σχηματιστεί, υπολογίζεται το κέντρο μάζας του αντικειμένου. Μέσω αυτού μπορούν εύκολα να υπολογιστούν και οι τελικές συντεταγμένες θέσης του στην εικόνα. Η συνάρτηση επιστρέφει τις συντεταγμένες κέντρου, τις διαστάσεις του πλαισίου οριοθέτησης και τη γωνία που σχηματίζεται.

Σε αυτό το σημείο να υπογραμμισθεί ότι για να θεωρηθεί ότι ένα αντικείμενο είναι επιτυχώς αναγνωρισμένο θα πρέπει να εντοπιστεί ταυτόχρονα και από τη λειτουργία εντοπισμού περιγράμματος αντικειμένων και από την εκάστοτε λειτουργία εντοπισμού που προορίζεται να αναγνωρισθεί.

Η συσχέτιση ενός αντικειμένου με το χαρακτηριστικό της γωνίας που σχηματίζει υλοποιείται μέσω μιας συνάρτησης. Η εν προκειμένω συνάρτηση καλείται με σταθερό όρισμα τη λίστα εντοπισθέντων αντικειμένων από τη λειτουργία εντοπισμού περιγράμματος αντικειμένων και μία εκ των λιστών που έχουν προκύψει από τις

λειτουργίες εντοπισμού μέσω του αλγορίθμου Yolo και εντοπισμού των QR codes.

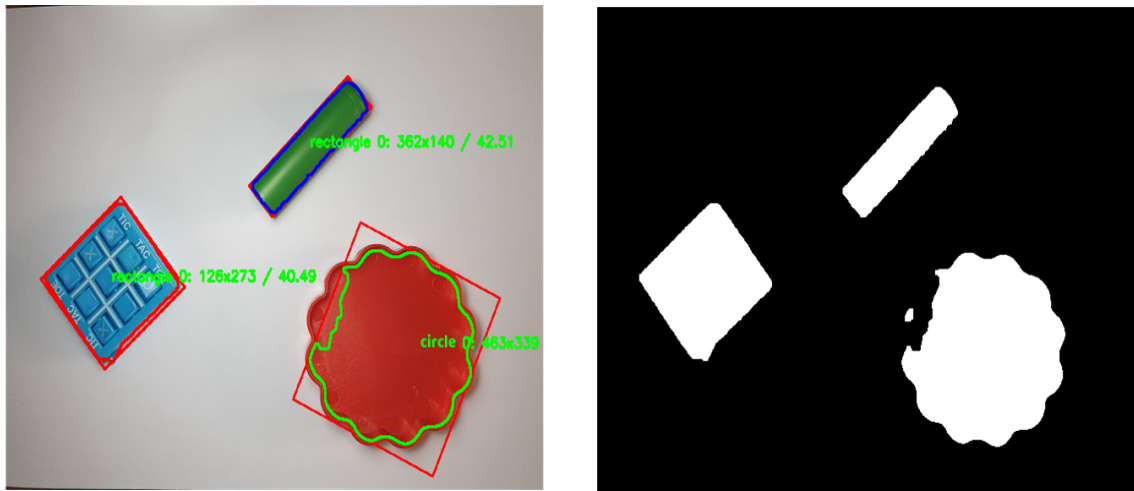
Η συσχέτιση επιτυγχάνεται μέσω του κοινού χαρακτηριστικού που έχει εξαχθεί από όλες τις λειτουργίες εντοπισμού, αυτού των συντεταγμένων (x,y) του κέντρου. Όταν εντοπισθούν αντικείμενα που τα κέντρα τους έχουν μικρή απόκλιση μεταξύ τους, τότε θεωρείται ότι πρόκειται για το ίδιο αντικείμενο. Να σημειωθεί ότι οι συντεταγμένες κέντρου που επιστρέφει ο αλγόριθμος YOLO αποτελούν το κέντρο του πλαισίου οριοθέτησης ενώ οι συντεταγμένες κέντρου που επιστρέφει η μέθοδος εντοπισμού περιγράμματος αντικειμένων αποτελούν το κέντρο βάρους των αντικειμένων. Οι συντεταγμένες αυτές δεν έχουν μεγάλη απόκλιση μεταξύ τους γι' αυτό και μέσω μιας τιμής κατωφλίου που εφαρμόζεται κατά τη σύγκριση εξάγεται αν αναφέρεται στο ίδιο αντικείμενο. Αντίστοιχα και η μέθοδος εντοπισμού QR code επιστρέφει τις συντεταγμένες κέντρου του QR code το οποίο μπορεί να έχει τοποθετηθεί σε οποιαδήποτε σημείο στην επιφάνεια του αντικειμένου. Εφαρμόζοντας πάλι μια τιμή κατωφλίου γίνεται η σύγκριση και εξάγεται αν αναφέρεται στο ίδιο αντικείμενο. Τέλος, η συνάρτηση επιστρέφει μια νέα λίστα με την ετικέτα ονόματος, τις συντεταγμένες κέντρου από τη μέθοδο εντοπισμού περιγράμματος και τη γωνία που σχηματίζει το αντικείμενο.

3.2.6 Παραδείγματα εντοπισμού και αναγνώρισης αντικειμένων σε διάφορες συνθήκες φωτισμού

Γίνεται εύκολα αντιληπτό ότι σε εφαρμογές που βασίζονται στη λήψη δεδομένων από οπτική καταγραφή είναι ιδιαίτερα σημαντικό να καταφέρνουν να εξάγουν αναλλοίωτο το περιεχόμενο εντοπισμού για διαφορετικές συνθήκες φωτισμού. Γι' αυτό το λόγο και στη παρούσα υλοποίηση έχει προβλεφθεί αυτός ο παράγοντας μιας και οι συνθήκες φωτισμού και το περιβάλλον λειτουργίας μπορούν να μεταβληθούν. Παρακάτω θα αναλυθούν παραδείγματα εικόνων από διαφορετικές λειτουργίες εντοπισμού και φωτισμού καθώς και θα μελετηθούν περιπτώσεις όπου το υλικό του αντικειμένου δύναται να επηρεάσει σημαντικά στην ακρίβεια εντοπισμού του.

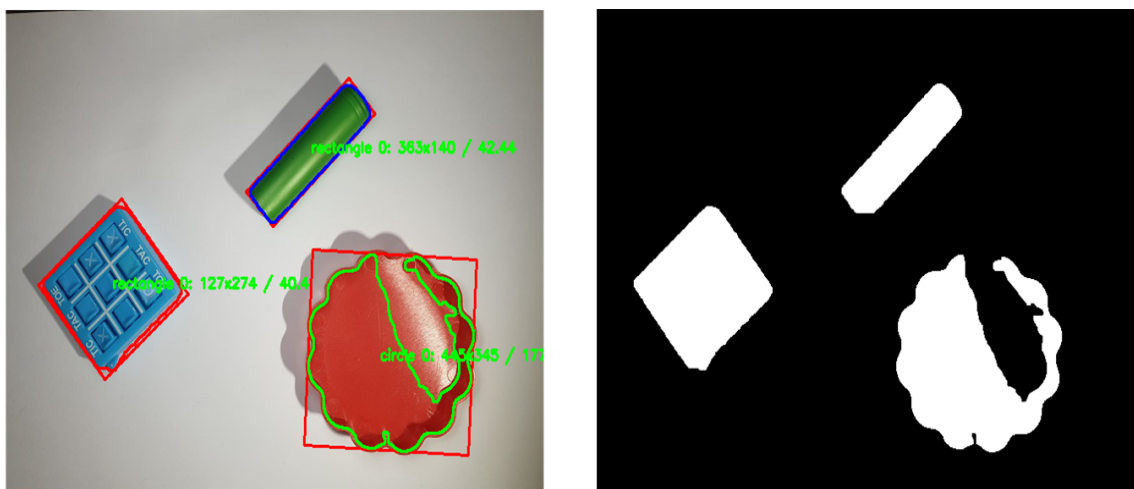
Αρχικά θα εξεταστεί η λειτουργία εντοπισμού σχήματος και χρώματος αντικειμένων σε συνθήκες καλού και κακού φωτισμού. Ως συνθήκη καλού φωτισμού θεωρείται όταν η πηγή φωτός προσπίπτει κάθετα και διαχέεται ομοιόμορφα στην ευρύτερη επιφάνεια που εντοπίζονται τα αντικείμενα, γεγονός που έχει ως αποτέλεσμα την αποφυγή δημιουργίας σκιών και αλλοίωσης της χρωματικής απεικόνισης των αντικειμένων. Στην αντίθετη περίπτωση ως συνθήκη κακού φωτισμού θεωρείται όταν η πηγή φωτός προσπίπτει με γωνίας στην επιφάνεια εντοπισμού δημιουργώντας έντονες σκιές, χρωματικές αλλοιώσεις και αντανακλάσεις στην επιφάνεια των αντικειμένων.

Στην Εικόνα 3.9 παρατίθενται δύο στιγμιότυπα από τη ροή video της κάμερας. Στο πρώτο απεικονίζεται ο αλγόριθμος εντοπισμού του σχήματος και στο δεύτερο εφαρμόζεται ο αλγόριθμος εντοπισμού του χρώματος. Παρατηρείται ότι στα αντικείμενα έχει λειτουργήσει ορθά ο εντοπισμός τους τόσο στο χρώμα όσο και κατά προσέγγιση στο σχήμα τους. Δεν υπάρχουν σκιές και έτσι ο αλγόριθμος έχει καταφέρει να εφαρμόσει επακριβώς το πλαίσιο οριοθέτησης. Βέβαια στην περίπτωση του κόκκινου αντικειμένου λόγω και του υλικού κατασκευής του, εμφάνισε μια μικρή χρωματική αλλοίωση. Αυτό φυσικά δεν αποτελεί πρόβλημα καθώς το κέντρο του αντικειμένου δεν διαφοροποιείται αρκετά ώστε να προκαλέσει μετατόπιση της ακριβής θέσης του.



Εικόνα. 3.9: Συνθήκες καλού φωτισμού

Ακολούθως στην Εικόνα 3.10 μελετώνται τα ίδια αντικείμενα με διαφορετικές συνθήκες φωτισμού. Σε αυτή τη περίπτωση εμφανίζονται έντονες σκιές και αντανάκλαση στο κόκκινο αντικείμενο. Η αντανάκλαση που εμφανίζεται έχει ως αποτέλεσμα για εκείνο το σημείο του αντικειμένου να εμφανίζεται ως απόχρωση του άσπρου χρώματος. Έτσι ο αλγόριθμος εντοπισμού του χρώματος εντοπίζει άσπρο και όχι κόκκινο κάτι που θεωρείται ως μέρος του χρώματος παρασκήνιου. Για το λόγο αυτό αποκλείει την άσπρη περιοχή από την επιφάνεια του αντικειμένου και τη θεωρεί παρασκήνιο. Παρόλα όμως αυτής της αλλοίωσης ο αλγόριθμός έχει καταφέρει να φιλτράρει και να αποβάλει τις σκιές και τις αντανακλάσεις έχοντας εφαρμόσει ορθά το πλαίσιο οριοθέτησης στα περιγράμματα των αντικειμένων.



Εικόνα. 3.10: Συνθήκες κακού φωτισμού

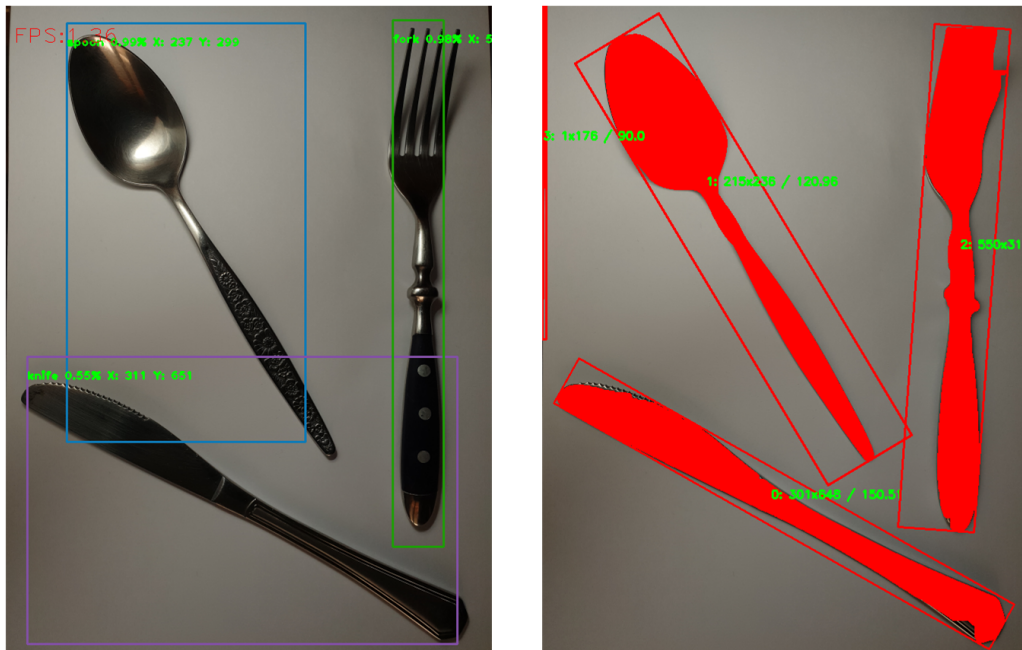
Ενδιαφέρον παρουσιάζουν οι περιπτώσεις όπου το υλικό των αντικειμένων δύναται να μεγεθύνει τα προβλήματα που δημιουργεί ο κακός φωτισμός. Τα αντικείμενα αυτά είναι συνήθως τα μεταλλικά ή εκείνα με γυαλιστερή επιφάνεια διότι το φως που προσπίπτει στην επιφάνεια τους θα δημιουργεί έντονη αντανάκλαση και χρωματική αλλοίωση. Στις παρακάτω περιπτώσεις θα εξεταστούν μεταλλικά καθημερινά σκεύη (πιρούνι, κουτάλι, μαχαίρι) με τη μέθοδο αναγνώρισης αντικειμένων μέσω του αλγορίθμου Yolo σε διαφορετικές συνθήκες φωτισμού. Οι συνθήκες στις οποίες θα μελετηθούν είναι κακού, χαμηλού, φυσικού και καλού φωτισμού.

Στην Εικόνα 3.11 εφαρμόζονται συνθήκες κακού φωτισμού. Παρατηρείται ότι παρά την ύπαρξη ανακλάσεων και έντονων σκιάσεων και οι δύο μέθοδοι εντοπισμού των αντικειμένων έχουν εξάγει ορθά αποτελέσματα με σωστές τοποθετήσεις για το κέντρο βάρους των αντικειμένων. Ο αλγόριθμος YOLO έχει καταφέρει να αναγνωρίσει σχεδόν στο 100% το πιρούνι και το κουτάλι και στο 75% το μαχαίρι που έχει και τις πιο έντονες χρωματικές αλλοιώσεις. Αντίστοιχα η μέθοδος εντοπισμού περιγράμματος αντικειμένων με μικρή απόκλιση λόγω των σκιών (κουτάλι, πιρούνι) έχει καταφέρει να εντοπίσει το ακριβές περίγραμμα από όλα τα αντικείμενα επιτυχώς.

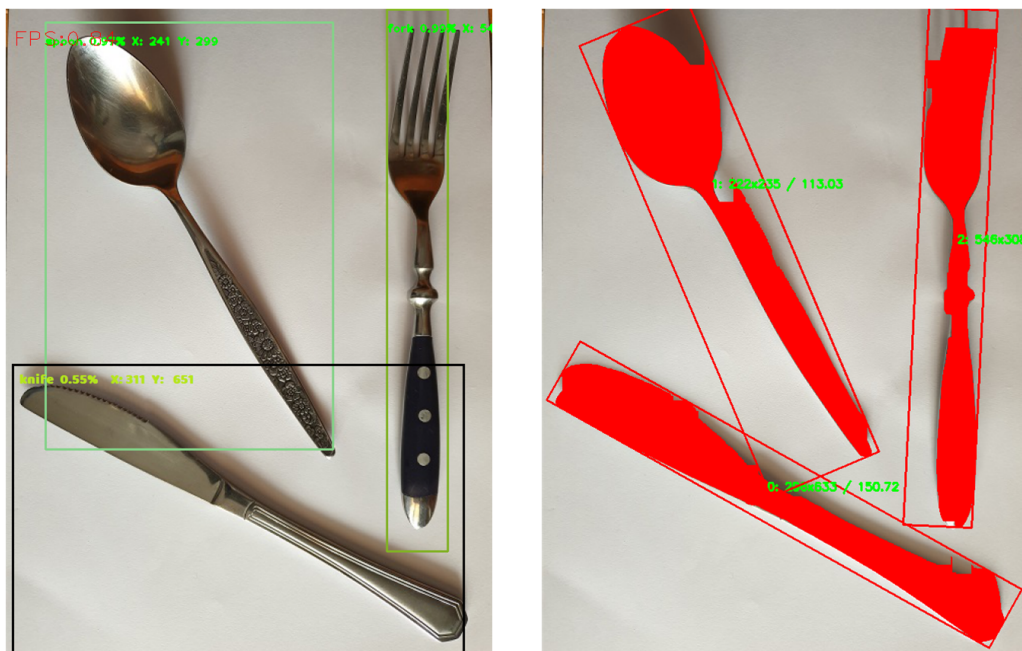


Εικόνα. 3.11: Συνθήκες κακού φωτισμού

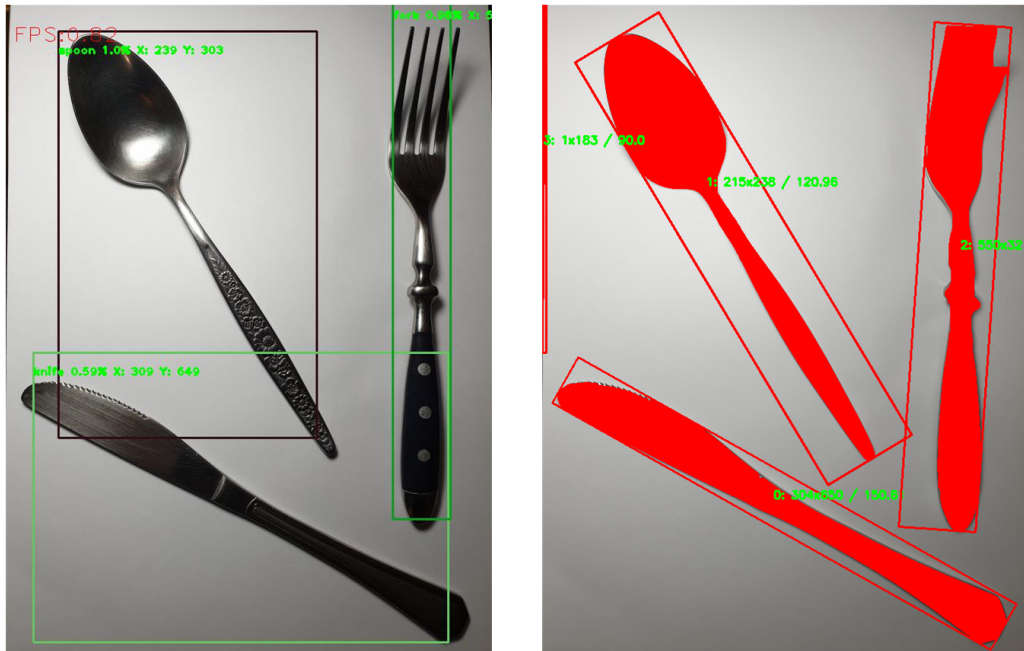
Στις Εικόνες 3.12, 3.13, 3.14 παρατηρείται ότι και οι δύο μέθοδοι έχουν καταφέρει να εντοπίσουν τα αντικείμενα επιτυχώς και η μέθοδος εντοπισμού περιγραμμάτων να σχεδιάσει με μεγάλη ακρίβεια το περίγραμμα των αντικείμεμων, δίνοντας έτσι μεγάλη ακρίβειά για την θέση του αντικειμένου και το κέντρο βάρους που αποτελεί το σημείο ανασήκωσης.



Εικόνα. 3.12: Συνθήκες χαμηλού φωτισμού



Εικόνα. 3.13: Συνθήκες φυσικού φωτισμού



Εικόνα. 3.14: Συνθήκες καλού φωτισμού

Τέλος, στην Εικόνα 3.15 παρατίθεται στιγμιότυπα από τη λειτουργία εντοπισμού QR codes και εντοπισμού περιγράμματος αντικειμένων. Παρατηρείται ότι και οι δύο μέθοδοι έχουν εξάγει ορθά αποτελέσματα τόσο στον εντοπισμό και διάβασμα των QR codes όσο και των περιγραμμάτων των αντικειμένων.



Εικόνα. 3.15: Εντοπισμός και αναγνώριση QR code

3.3 Φωνητικές εντολές και αλληλεπίδραση

Η φωνητική αλληλεπίδραση είναι ο συνδυαστικός κρίκος μεταξύ του χρήστη και του ρομποτικού συστήματος. Αποτελεί το μόνο τρόπο με τον οποίο ο χρήστης μπορεί να μεταφέρει τις εντολές του προς το ρομποτικό σύστημα και αντίστοιχα το ρομποτικό σύστημα να ενημερώσει τον χρήστη. Σκοπός αυτής της λειτουργίας είναι αρχικά η επιτυχής αναγνώριση των φωνητικών εντολών και δεύτερον η εκφώνηση μηνυμάτων με τρόπο κατανοητό προς το χρήστη. Γίνεται αντιληπτό ότι η ορθή και αδιάκοπη λειτουργία ιδιαίτερα της φωνητικής αναγνώρισης κατέχει ζωτικής σημασίας ρόλο καθώς σε περίπτωση αποτυχίας της καθιστά εκτός λειτουργίας το ρομποτικό σύστημα και τις λειτουργίες του.

Η εν λόγω λειτουργία ενσωματώνεται στο κεντρικό σενάριο κώδικα Speech που έχει αναπτυχθεί για την εκτέλεση της βασικής ρουτίνας εκτέλεσης του ρομποτικού συστήματος. Το σενάριο κώδικα Speech αποτελεί το κεντρικό κόμβο λήψης, επεξεργασίας και ενεργοποίησης εκτέλεσης ενεργειών. Η είσοδος του σεναρίου είναι τα αποτελέσματα εξόδου του σεναρίου κώδικα όρασης μηχανής Vision και οι φωνητικές εντολές που δίνει ο χρήστης από το σενάριο κώδικα Control.

Πιο συγκεκριμένα στο σενάριο κώδικα εκτός των συναρτήσεων που υλοποιούν τις λειτουργίες φωνητικής αλληλεπίδρασης, υλοποιούνται και συναρτήσεις με συνδυαστική επεξεργασία δεδομένων. Αυτές οι συναρτήσεις είναι υπεύθυνες για την υλοποίηση της λειτουργίας αναζήτησης `search()` και της απευθείας παράδοσης αντικειμένου `check_object()`.

Η συνάρτηση `search()` καλείται όταν έχει γίνει αναγνώριση από το σύστημα της λέξης «ψάξε» ή «search» από το χρήστη. Ακολούθως δίνει κατάλληλο σήμα μέσω της memcache στο σενάριο εκτέλεσης κινήσεων Control για να ενεργοποιηθεί η λειτουργία αναζήτησης. Με την ολοκλήρωση της διαδικασίας από το βραχίονα τα αντικείμενα που έχουν εντοπιστεί ταξινομούνται σε λίστες και εκφωνούνται προς ενημέρωση του χρήστη.

Η συνάρτηση `check_object()` εκτελείται κάθε φορά που ο χρήστης κατονομάζει ένα αντικείμενο για παράδοση. Η συνάρτηση δέχεται ως είσοδο το όνομα του αντικείμενου που ζήτησε ο χρήστης και τη λίστα των αντικειμένων που έχουν εντοπιστεί. Ουσιαστικά γίνεται έλεγχος αν το αντικείμενο που ζητάει ο χρήστης υπάρχει στα αντικείμενα που έχουν εντοπιστεί από το Vision. Σε περίπτωση που το αντικείμενο υπάρχει στη λίστα τότε οι συντεταγμένες κέντρου του μεταβιβάζονται στο σενάριο Control για παράδοση, σε άλλη περίπτωση ενημερώνεται ο χρήστης μέσω ηχητικού μηνύματος για την αποτυχία εύρεσης του αντικειμένου.

3.3.1 Αναγνώριση φωνητικών εντολών

Για την υλοποίηση της φωνητικής αναγνώρισης έγινε χρήση της βιβλιοθήκης SpeechRecognition η οποία υποστηρίζει διάφορες online και offline μηχανές αναγνώρισης φωνής και API. Για τις ανάγκες της υπάρχουσας υλοποίησης επιλέχθηκε η χρήση του Google Voice Recognition API. Η επιλογή αυτή προέκυψε ύστερα από διαδικασία δοκιμής και αξιολόγησης και άλλων μηχανών αναγνώρισης ομιλίας όπως CMU Sphinx τόσο στη Αγγλική όσο και στην Ελληνική γλώσσα. Το API της Google κατάφερε να πετύχει τις περισσότερες επιτυχημένες αναγνωρίσεις στον λιγότερο χρόνο όταν ο έλεγχος έγινε για την ελληνική γλώσσα.

Οι περιπτώσεις που απαιτείται από το χρήστη να εκφωνήσει τις εντολές είναι στην αρχή εκτέλεσης του προγράμματος επιλέγοντας τη γλώσσα που επιθυμεί να επικοινωνήσει και στο σημείο που εκφωνείται η ενέργεια που θα εκτελέσει το ρομπότ. Η επίτευξη αναγνώρισης ομιλίας αποτελείται από τρία επιμέρους βήματα, την αρχικοποίηση μεταβλητών, την ηχογράφηση και αποστολή του ηχητικού και τέλος τη λήψη του κειμένου. Για την αρχικοποίηση, ορίζονται αντικείμενα των κλάσεων `Recognizer()` και `Microphone()` καθώς και οι μεταβλητές `sample_rate` και το `chunk_size`. Μέσω της μεταβλητής `sample_rate` καθορίζεται πόσο συχνά ηχογραφούνται οι τιμές για επεξεργασία και μέσω της `chunk_size` καθορίζεται ο αριθμός των bytes δεδομένων που θέλουμε να διαβάσουμε ταυτόχρονα. Προκειμένου να γίνει η ηχογράφηση ορίζεται αντικείμενο της κλάσης `Microphone()` ως πηγή και ρυθμίζεται δυναμικά το επίπεδο θορύβου περιβάλλοντος ώστε να αγνοηθεί από την

ηχογράφηση. Ακολουθώς γίνεται η ηχογράφηση του μηνύματος με χρήση της μεθόδου `listen()` και η κλήση του google API μέσω της μεθόδου `.recognize_google()` δηλώνοντας της γλώσσα που θα εντοπιστεί. Η απάντηση θα έρθει σε μορφή text για περαιτέρω επεξεργασία. Στην περίπτωση που υπάρχει καθυστέρηση απάντησης ή η υπηρεσία δεν αναγνώρισε το ηχητικό μήνυμα ο χρήστης ενημερώνεται με κατάλληλο μήνυμα. Η διαδικασία που αναλύθηκε ακολουθείται κάθε φορά που ο χρήστης δίνει μια φωνητική εντολή.

Όπως έχει αναφερθεί για να θεωρηθεί μια φωνητική εντολή έγκυρη θα πρέπει να ικανοποιεί κάποιες συμβάσεις που έχουν τεθεί. Αυτές οι συμβάσεις έχουν να κάνουν τόσο με την πληρότητα των στοιχείων σε μια εντολή παράδοσης όσο και τη χρήση συγκεκριμένων λέξεων για την λειτουργία αναζήτησης και τερματισμού. Για παράδειγμα όταν ο χρήστης ζητάει ένα αντικείμενο θα πρέπει η εντολή να περιέχει τη λέξη «φέρε» ή «πιάσε» ακολουθούμενη με το χρώμα και το σχήμα αν ζητάει κάποιο αντικείμενο με βάση τον εντοπισμό αυτής της λειτουργίας ή το όνομα του αντικειμένου για τις περιπτώσεις αναγνώρισης με Yolo και QR code. Αντίστοιχα για τη λειτουργία αναζήτησης θα πρέπει να χρησιμοποιεί τη λέξη «ψάξε» και για τη λειτουργία τερματισμού τη λέξη «σταμάτησε». Με τον ίδιο τρόπο ισχύει και για την αγγλική γλώσσα.

Προκειμένου να υλοποιηθεί αυτός ο έλεγχος εγκυρότητας έχουν αναπτυχθεί συναρτήσεις στις οποίες το κείμενο που έχει επιστραφεί από την αναγνώριση ομιλίας επεξεργάζεται ώστε να εξαχθούν μόνο οι λέξεις που ικανοποιούν αυτούς τους κανόνες. Όλες οι αποδέκτες λέξεις του συστήματος έχουν εισαχθεί σε λίστες ανά κατηγορία. Πιο συγκριμένα η συνάρτηση `close_matches()` δέχεται ως είσοδο το κείμενο που έχει επιστραφεί από την αναγνώριση ομιλίας και μία εκ των λιστών με τις αποδεκτές λέξεις. Η συνάρτηση ελέγχει κάθε λέξη από το επιστρεφόμενο κείμενο με κάθε λέξη από τη λίστα. Η λέξη με την οποία μοιάζει περισσότερο επιστρέφεται ως την αναγνωριζόμενη λέξη. Κατ' αυτόν τρόπο δίνοντας ως όρισμα διαφορετικές λίστες με αποδεκτές λέξεις ελέγχεται μέσω της βασικής ρουτίνας ελέγχου που υλοποιείται από τη συνάρτηση `run()` αν η φωνητική εντολή εκπληρεί ή όχι τις συμβάσεις πληρότητας ώστε στο τέλος να εξαχθεί η τελική εντολή.

3.3.2 Φωνητική απάντηση

Η λειτουργία της φωνητικής απάντησης χρησιμοποιείται κάθε φορά που το σύστημα θέλει να μεταφέρει μηνύματα στο χρήστη. Ο χρήστης ενημερώνεται διαρκώς με κατάλληλα μηνύματα στις περιπτώσεις δήλωσης ετοιμότητας του συστήματος για εκτέλεση εντολών καθώς και κατά την έναρξη και λήξη κάθε ενέργειας που εκτελεί το σύστημα. Ακόμα στις περιπτώσεις αποτυχίας εύρεσης αντικειμένου ή αναγνώρισης εντολής καθώς και στην περίπτωση που έχει προκύψει κάποιο σφάλμα στο σύστημα. Η λειτουργία υλοποιείται χάρις στη βιβλιοθήκη μετατροπής κειμένου σε ομιλία **pyttsx3**. Λόγω του γεγονότος ότι η ανάπτυξη του λειτουργικού μέρους έγινε σε πλατφόρμα που χρησιμοποιεί Windows 10 OS επιλέχθηκαν οι προ-εγκατεστημένες φωνές ομιλίας του λειτουργικού συστήματος τόσο για την ελληνική όσο και για την αγγλική.

Για την ενεργοποίηση της λειτουργίας καλείται η συνάρτηση **pyttsx3.init()** μέσω της οποίας γίνεται αρχικοποίηση και αναφορά στο αντικείμενο **pyttsx3.Engine** για τη ρύθμιση και διαχείριση όλων των παραμέτρων. Χρησιμοποιώντας τη συνάρτηση **engine.setProperty()** φορτώνεται η φωνή για την εκφώνηση του κειμένου και με την **engine.say()** ενεργοποιείται η εκφώνηση του κειμένου που εισήχθη ως όρισμα.

3.4 Εκτέλεση κινήσεων

Αναμφισβήτητος παράγοντας αποτελεσματικότητας της παρούσας υλοποίησης αποτελεί η ορθή εκτέλεση των εντολών από το ρομποτικό βραχίονα και τελικώς η παράδοση του εκφωνηθέντος αντικειμένου στο χέρι του χρήστη. Προϋπόθεση για την σωστή τοποθέτηση του βραχίονα στο χώρο και της κλίσης του τελικού εργαλείου δράσης αποτελούν οι συντεταγμένες κέντρου του αντικειμένου και οι μοίρες της γωνίας που σχηματίζει. Μιας και όλα τα σενάρια χρήσης περιλαμβάνουν την άμεση επαφή με τον άνθρωπο έτσι και ο τρόπος που τοποθετείται και κινείται στο χώρο θα πρέπει να χαρακτηρίζονται από ομαλότητα, ακρίβεια και ταχύτητα. Έτσι διασφαλίζεται αίσθημα εμπιστοσύνης από το χρήστη που εκφράζεται με άνεση στη μεταξύ τους αλληλεπίδραση.

Το σενάριο κώδικα που έχει αναπτυχθεί για τον υπολογισμό των απαιτούμενων κινήσεων και τελικώς την εκτέλεση αυτών είναι το Control. Κύριες λειτουργίες που υλοποιούνται είναι η λειτουργία αναζήτησης αντικειμένων, μετατροπής συντεταγμένων κέντρου σε θέση του βραχίονα στο χώρο, προσέγγιση και ανασήκωση αντικειμένου και τέλος αποδέσμευση αυτού στο χέρι του χρήστη.

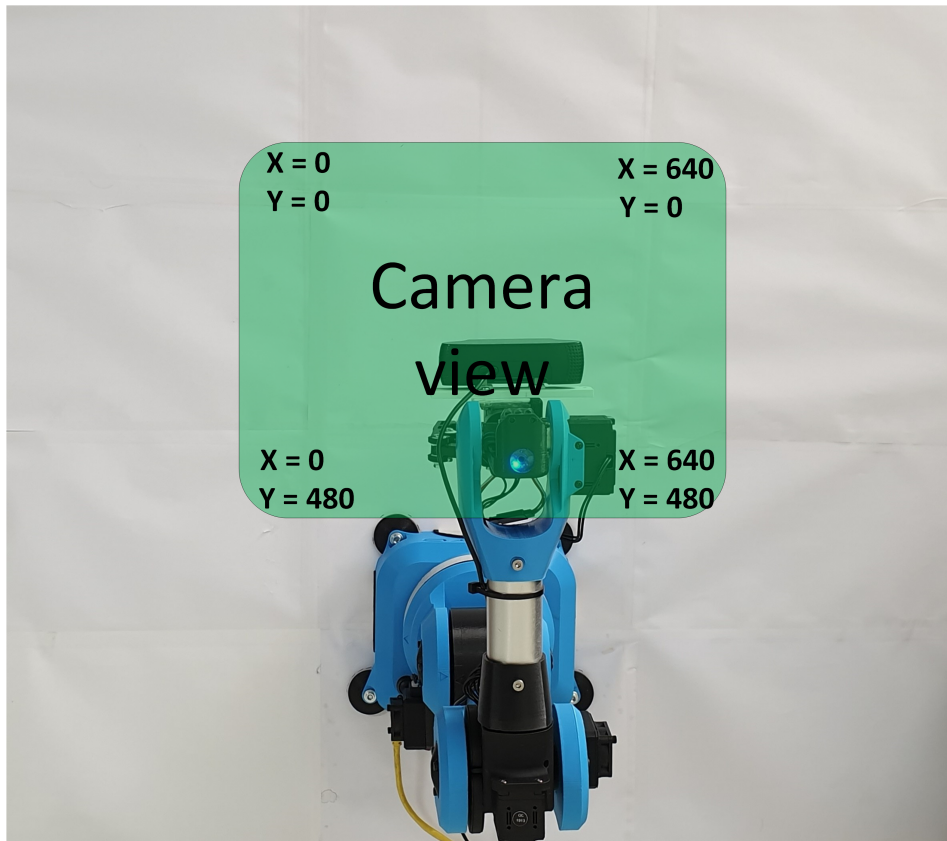
3.4.1 Niryo-One Python API

Η σύνδεση του σεναρίου Control με το ρομποτικό βραχίονα Niryo-One επιτυγχάνεται μέσω του Python API που προσφέρεται από την κατασκευάστρια εταιρεία. Όπως έχει αναφερθεί η Niryo διαθέτει ελεύθερα τόσο τα σχέδια εκτύπωσης του βραχίονα όσο και το λογισμικό που το συνοδεύει. Σε αυτό το πλαίσιο έχει δημιουργήσει Python API το οποίο επιτρέπει την ενσωμάτωσή του σε σενάριο κώδικα Python για το χειρισμό του βραχίονα. Με τη χρήση ειδικών εντολών δίνεται πρόσβαση στην παραμετροποίηση και εκτέλεση κινήσεων. Αρχικά για να επιτευχθεί η αποστολή των εντολών στο βραχίονα θα πρέπει να δημιουργηθεί μια σύνδεση. Για την σύνδεση αυτή απαιτείται ο βραχίονας και ο ελεγκτής (υπολογιστής) να βρίσκονται στο ίδιο δίκτυο. Αυτό μπορεί να επιτευχθεί είτε ασύρματα είτε ενσύρματα. Στην αρχή εκτέλεσης του σεναρίου δημιουργείται αντικείμενο τύπου **NiryoOneClient()** και μέσω αυτού συνδέεται στο ρομποτικό βραχίονα θέτοντας στη συνάρτηση **connect()** την IP του βραχίονα. Ακολούθως εκτελείται αυτόματη βαθμονόμηση που περιλαμβάνει ουσιαστικά μια ρουτίνα κίνησης του βραχίονα ώστε να βαθμονομηθούν σωστά τα κινητήρια μοτέρ και την αρχικοποίηση του τελικού εργαλείου δράσης.

3.4.2 Μετατροπή συντεταγμένων μηχανικής όρασης σε συντεταγμένες χώρου

Βασική λειτουργία του σεναρίου είναι η μετατροπή των συντεταγμένων κέντρου του αντικειμένου στην κατάλληλη θέση (pose) που θα πρέπει να πάρει ο βραχίονας. Το σενάριο κώδικα Control δέχεται τις εντολές προς εκτέλεση από το σενάριο κώδικα Speech. Οι συντεταγμένες κέντρου αποτελούνται από τιμές που κυμαίνονται από 0 έως 640 για τον οριζόντιο άξονα και από 0 έως 480 για τον κάθετο άξονα. Οι τιμές αυτές αναφέρονται στον αριθμό των pixel που υπάρχουν στο παράθυρο που «βλέπει» η κάμερα μιας και γίνεται υποβάθμιση της ανάλυσης εισόδου σε 640x480.

Έτσι λοιπόν αν ένα αντικείμενο για παράδειγμα έχει την τιμή 120x340 τότε σημαίνει ότι το κέντρο του βρίσκεται στο pixel με θέση 120 στον οριζόντιο άξονα και 340 στον κάθετο. Στην εικόνα που ακολουθεί απεικονίζεται το «κάδρο» λήψης της κάμερας στο χώρο και οι τιμές των pixel στις γωνίες.



Εικόνα. 3.16: Κάδρο λήψης κάμερας

Η θέση του βραχίονα μπορεί να οριστεί με δύο τρόπους, είτε με τη γωνία περιστροφής για κάθε κινητήρα άρθρωσης είτε με τιμές για τα επίπεδα x , y , z που ορίζουν τη θέση του τελικού εργαλείου δράσης στο χώρο και roll, pitch, yaw για τον προσανατολισμό τοποθέτησης του καρπού. Το σύνολο των τιμών του δεύτερου τρόπου ονομάζεται αλλιώς και πόζα. Για τους σκοπούς της παρούσας υλοποίησης έγινε χρήση του χειρισμού μέσω των τιμών x , y , z , roll, pitch, yaw. Πριν όμως αναλυθεί ο τρόπος μέσω του οποίου έγινε η συσχέτιση των σημείων από το κάδρο που «βλέπει» η κάμερα σε πραγματικό σημείο στο χώρο θα γίνει αναφορά στη θέση αναμονής και στον τρόπο που εκτελεί αναζήτηση ο βραχίονας.

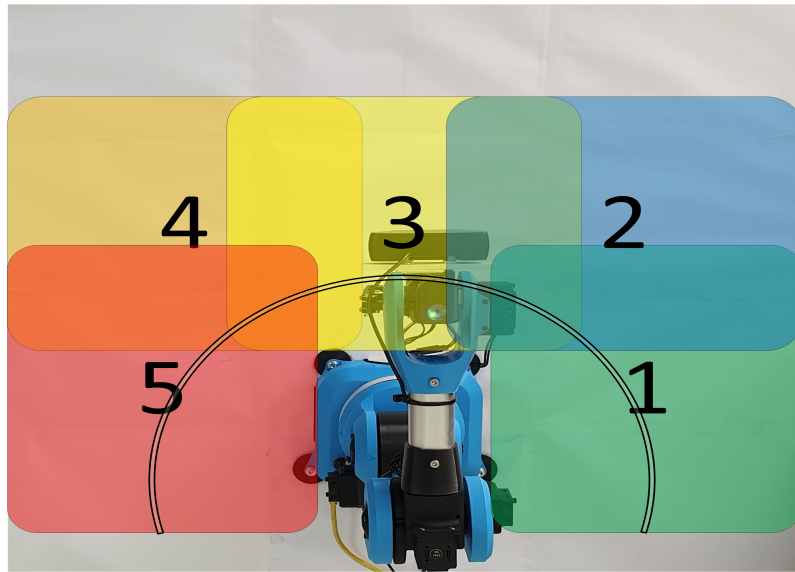
Για τον βραχίονα έχει οριστεί μια θέση ως θέση αναμονής ή ως θέση εκκίνησης. Σε αυτή τη θέση ο βραχίονας βρίσκεται ανυψωμένος με προσανατολισμό προς τα εμπρός και ο πήχης του βρίσκεται παράλληλα με το οριζόντιο επίπεδο. Το τελικό εργαλείο δράσης βρίσκεται στραμμένο προς τα κάτω έτσι ώστε η κάμερα να καταγράφει τα αντικείμενα που βρίσκονται στη περιβάλλουσα περιοχή. Η θέση αυτή έχει επιλεγεί καθώς η κάμερα βρίσκεται σε πλήρη καθετότητα και σε ικανοποιητικό σημείο για να αναγνωρίζει και να περιλαμβάνει αρκετά αντικείμενα στο «κάδρο» λήψης της.

Η καθετότητα αποτελεί σημαντικό παράγοντα για διάφορους λόγους. Αρχικά επιτυγχάνεται συνέπεια στην καταγραφή των σημείων και αποφυγή περεταίρω τροποποίησης σε περίπτωση αντικατάστασης της κάμερας. Ακόμα εξαλείφεται ο παράγοντας του βάθους στην εικόνα γεγονός που θα απαιτούσε περαιτέρω επεξεργασία και χρήση στεροσκοπικής κάμερας το οποίο δεν αποτελεί κομμάτι έρευνας στην παρούσα διπλωματική εργασία.



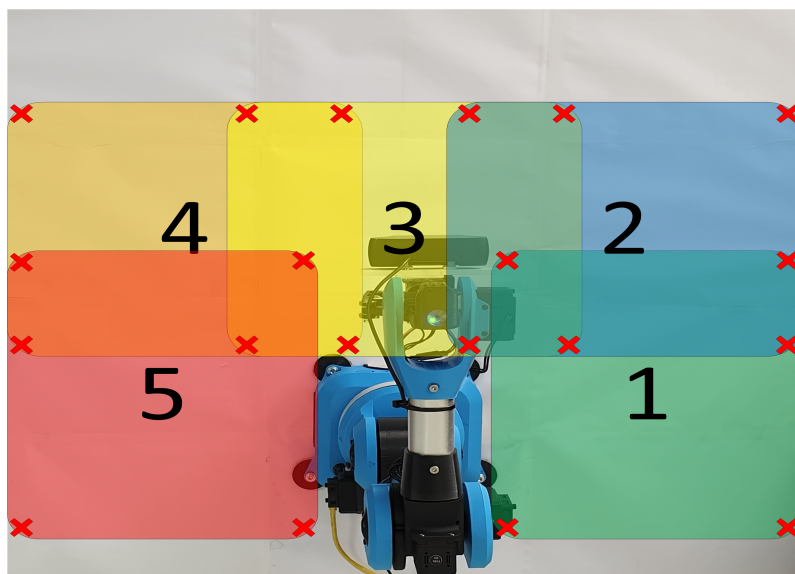
Εικόνα. 3.17: Θέση αναμονής ή θέση εκκίνησης

Η λειτουργία της αναζήτησης έχει οριστεί να καλύπτει την περιοχή περιμετρικά του βραχίονα σε συνολικό εύρος 220° . Για την κάλυψη αυτού το χώρου έχει διαιρεθεί σε πέντε τμήματα και έχουν οριστεί οι αντίστοιχες θέσεις. Κατά τη εκτέλεση της αναζήτησης ο βραχίονας διατηρεί τη θέση αναμονής καθώς περιστρέφεται αριστερόστροφα ο σύνδεσμος **j1** της βάσης.



Εικόνα. 3.18: Περιοχές αναζήτησης

Για να συσχετιστούν τα σημεία από το «κάδρο» εικόνας της κάμερας με τα σημεία που ανταποκρίνονται στο χώρο απαιτείται αρχικά να οριοθετηθεί η κίνηση του τελικού εργαλείου δράσης μέσα σε καθένα από αυτά «κάδρα». Για να επιτευχθεί αυτό χρησιμοποιήθηκε η δυνατότητα «ζωντανής» ενημέρωση των τιμών πόζας για κάθε θέση που βρίσκεται ο βραχίονας μέσω του API. Σημειώνοντας σημάδια στο χώρο που συμπίπτουν με τις γωνίες των 5 κάδρων λήψεως και τοποθετώντας το τελικό εργαλείο δράσης σε αυτά επιτεύχθηκε η καταγραφή όλων των ακραίων σημείων χώρου που μπορεί να κινείται ο βραχίονας. Στην εικόνα που ακολουθεί παριστάνεται η διαδικασία που αναλύθηκε.



Εικόνα. 3.19: Σημάδια για τον προσδιορισμό των ακραίων σημείων χώρου

Στο σενάριο κώδικα Control η μετατροπή των συντεταγμένων κέντρου σε πόζα για το βραχίονα υλοποιείται απο συνάρτηση **pose_calculation()**. Η συνάρτηση δέχεται ως είσοδο τις συντεταγμένες κέντρου, τη γωνία και τον αριθμό της θέσης που ανήκει το αντικείμενο. Αρχικά μέσω της μεθόδου **interp1d()** γίνεται αντιστοίχιση του εύρους τιμών των φυσικών ορίων με το εύρος τιμών του κάδρου λήψης, έτσι ώστε εισάγοντας τις συντεταγμένες κέντρου να προκύψουν οι αντίστοιχες τιμές στο φυσικό χώρο. Όσο αφορά τον προσανατολισμό του τελικού εργαλείου δράσης στη προσέγγιση και πιάσιμο του αντικειμένου, αυτός είναι κάθετος της άνω πλευράς του αντικειμένου και παραμένει σταθερός σε όλες τις περιπτώσεις. Η περιστροφή του τελικού εργαλείου δράσης εξαρτάται από τη γωνία του αντικειμένου και υπολογίζεται στη συνάρτηση **deg_calc()**. Ως έξοδο της **pose_calculation()** επιστρέφονται όλες τις τιμές της πόζας έτοιμες για εκτέλεση.

Οι λειτουργίες αναζήτησης αντικειμένων και αναζήτησης του χρήστη υλοποιούνται από τις συναρτήσεις **search_object()** και **search_user()** αντίστοιχα. Έχουν πανομοιότυπη λειτουργία και περιλαμβάνουν την εκτέλεση ρουτίνας μετάβασης μεταξύ των προκαθορισμένων θέσεων αναζήτησης. Κατά τη διάρκεια αυτής συλλέγονται οι απαραίτητες πληροφορίες από το σενάριο κώδικα Vision, ώστε με το πέρας να επιστραφούν μέσω memcache τα αναγνωρισμένα αντικείμενα στο σενάριο κώδικα Speech για ανάγνωση και επεξεργασία.

Η εκτέλεση της εκάστοτε πόζας του βραχίονα εκτελείται από τη συνάρτηση **catch_object()** όταν πρόκειται για προσέγγιση και ανασήκωση αντικειμένου και από τη **release_object()** για την απελευθέρωση του αντικειμένου. Η συνάρτηση που εκτελείται για το άνοιγμα και κλείσιμο της δαγκάνας είναι η **nc.open_gripper()** και για την εκτέλεση της πόζας η **nc.move_pose()**.

3.5 Μεταφορά δεδομένων μεταξύ σεναρίων κώδικα

Η αρχιτεκτονική βάση της οποίας αναπτύχθηκε το λογισμικό μέρος, προβλέπει τη δημιουργία ενός σεναρίου κώδικα για κάθε λειτουργία και εν συνεχεία τη μεταξύ τους επικοινωνία για τον διαμοιρασμό των δεδομένων. Ο συνδυαστικός κρίκος μεταξύ των τριών σεναρίων κώδικα (Control, Vision, Speech) είναι το καταναμεμημένο

σύστημα αποθήκευσης μνήμης, memcache.

Η memcache διανέμεται δωρεάν, είναι ανοιχτού κώδικα, υψηλής απόδοσης κατανεμημένο σύστημα αποθήκευσης αντικειμένων μνήμης γενικής φύσης. Κυρίως προορίζεται για χρήση στην επιτάχυνση δυναμικών εφαρμογών ιστού, μειώνοντας το φορτίο της βάσης δεδομένων. Βασίζεται στην αποθήκευση με τιμή κλειδιού για μικρά κομμάτια αυθαίρετων δεδομένων (συμβολοσειρές, αντικείμενα) από αποτελέσματα κλήσεων βάσης δεδομένων, τα οποία είναι διαθέσιμα για τις πιο δημοφιλείς γλώσσες. Παρέχει ένα τεράστιο λεξικό διαθέσιμο στο δίκτυο το οποίο διαφοροποιείται από ένα κλασικό Python λεξικό, στα εξής:

- Τα κλειδιά και οι τιμές πρέπει να είναι bytes
- Τα κλειδιά και οι τιμές διαγράφονται αυτόματα μετά από ένα χρονικό διάστημα

Οι clients (πελάτες) της memcache επικοινωνούν με διακομιστή μέσω συνδέσεων TCP. Ο διακομιστής μνήμης memcache που εκτελείται ακούει σε μια διαμορφώσιμη θύρα και οι πελάτες που συνδέονται σε αυτήν τη θύρα, στέλνουν και λαμβάνουν δεδομένα στο διακομιστή.

Στην παρούσα υλοποίηση έγινε η εγκατάσταση της βιβλιοθήκης Python **pymemcache_client** και του memcache server ο οποίος τρέχει τοπικά στο σύστημα. Σε κάθε σενάριο κώδικα προκειμένου να συνδεθούν με τον server εγκαθιδρύουν μια σύνδεση client με τη μέθοδο **base.Client()**. Αφού γίνει αυτό δυο σημαντικές μέθοδοι που χρησιμοποιούνται στη memcache είναι η μέθοδος **set()** και **get()**. Στη πρώτη περίπτωση για την εισαγωγή δεδομένων στο server απαιτείται πρώτα η μετατροπή του στοιχείου σε JSON string μέσω της μεθόδου **json.dumps()** και ακολούθως η μεταφορά του χρησιμοποιώντας τη **client.set('key', data_str)**. Το όρισμα key είναι το χαρακτηριστικό όνομα που θα συνοδεύει τα δεδομένα που αποστάλθηκαν και θα επιτρέπει να ληφθούν από άλλους client. Κατά αντιστοιχία για την λήψη των δεδομένων χρησιμοποιείται αρχικά η μέθοδος **client.get('key')** και εν συνέχεια για την μετατροπή του JSON στην αρχική μορφή των δεδομένων η μέθοδος **json.loads()**. Στα σενάρια κώδικα το σύνολο των πληροφοριών που συνήθως δομούνται σε λίστες είναι απαραίτητο να μεταφέρονται και να λαμβάνονται με τη μεγαλύτερη ταχύτητα και

τις λιγότερο δυνατόν περιπτώσεις αποτυχίας. Χρησιμοποιείται εκτός των χρηστικών πληροφοριών, όπως είναι λίστες με τα αναγνωρισμένα αντικείμενα, φωνητική εντολή ή λίστα με το αντικείμενο για παράδοση και σε περιπτώσεις συγχρονισμού εκτέλεσης μεταξύ των σεναρίων χρησιμοποιώντας το ως σύστημα σηματοδότησης.

3.6 Σύνοψη μετρικών κώδικα

Σε αυτή την παράγραφο θα αναφερθούν συνοπτικά δεδομένα σχετικά με τον πηγαίο κώδικα της εργασίας. Το σύστημα αποτελείται, τόσο από υλικό, όσο και από λογισμικό. Για την ορθότερη παρουσίαση των δυνατοτήτων και της ποιότητας του, πραγματοποιήθηκε ένα σύνολο δοκιμών και μετρήσεων.

Το λογισμικό του συστήματος αποτελείται από 3 αρχεία πηγαίου κώδικα. Στους πίνακες που ακολουθούν παρουσιάζονται τα χαρακτηριστικά του κάθε σεναρίου ξεχωριστά και στο τέλος συνολικά.

Σενάρια κώδικα	Γραμμές κώδικα	Μέγεθος(bytes)	Συναρτήσεις	Βιβλιοθήκες
Speech	538	22.777	13	11
Vision	553	24.576	12	11
Control	332	21.675	9	6
Σύνολο	1423	469028	34	19

Πίνακας 3.1: Μετρικές αρχείων κώδικα.

Η κυκλωματική πολυπλοκότητα είναι μια μέτρηση της πολυπλοκότητας του κώδικα που προτείνει ο McCabe και θεωρείται συχνά ως ένας αριθμός που επιτρέπει να προσδιοριστεί η πολυπλοκότητα ενός προγράμματος. Όσο πιο περίπλοκος είναι ο κώδικας, τόσο περισσότερος χρόνος και πόροι θα χρειαστούν για να αναπτυχθεί, να διατηρηθεί και τόσο πιο πιθανό είναι να έχει σφάλματα έναντι κώδικα με χαμηλότερη βαθμολογία πολυπλοκότητας. Υψηλή πολυπλοκότητα δείχνει περισσότερες διαδρομές εντός του κώδικα. Ένα σκορ 10 ή λιγότερο θεωρείται αρκετά απλό και εύκολο στη συντήρηση, το 11-20 υποδεικνύει ένα κάπως περίπλοκο πρόγραμμα, το 21-50 είναι ένα πολύ περίπλοκο πρόγραμμα και οτιδήποτε άνω των 50 είναι τόσο περίπλοκο που δεν είναι δοκιμασμένο. Στα παρακάτω στιγμιότυπο παρατίθενται τα αποτελέσματα από τον έλεγχο κυκλωματικής πολυπλοκότητας McCabe.

```
(RoboticArm) C:\Users\Alchemist\PycharmProjects\RoboticArm\Main_Scripts>python -m mccabe --min 5 Control.py
111:0: 'catch_object' 5
146:0: 'find_to_release_object' 9
191:0: 'pose_calculation' 7
276:0: 'main' 10

(RoboticArm) C:\Users\Alchemist\PycharmProjects\RoboticArm\Main_Scripts>python -m mccabe --min 5 Vision.py
69:0: 'detectGeneral' 5
183:0: 'detectsShape' 5
295:0: 'detectYOLO' 7
368:0: 'comparison' 5
423:0: 'main' 12

(RoboticArm) C:\Users\Alchemist\PycharmProjects\RoboticArm\Main_Scripts>python -m mccabe --min 5 Speech.py
88:0: 'search' 13
209:0: 'language_detect' 10
264:0: 'detect' 5
322:0: 'run' 13
410:0: 'check_object' 7
430:0: 'main' 15
```

Εικόνα. 3.20: Αποτελέσματα κυκλωματικής πολυπλοκότητας

Όσο αφορά τον τομέα του υλικού, στον πίνακα 3.2 παρουσιάζονται τα μετρικά

χρόνων απόκρισης του βραχίονα που παραμένουν αμετάβλητα ανεξαρτήτως πλατφόρμας εκτέλεσης και αντικειμένου.

Χρόνος διεκπεραίωσης λειτουργίας αναζήτησης αντικειμένων (sec)	15
Χρόνος διεκπεραίωσης λειτουργίας αναζήτησης χρήστη (sec)	3-15
Χρόνος προσέγγισης και αρπαγής (sec)	7
Χρόνος παράδοσης (sec)	5

Πίνακας 3.2: Μετρικές αρχείων κώδικα.

3.7 Συμβάσεις λογισμικού μέρους

Κατά τον σχεδιασμό της αρχιτεκτονικής και την ανάπτυξη του λογισμικού μέρους χρειάστηκε να γίνουν κάποιες συμβάσεις για την ορθή εκτέλεση. Οι συμβάσεις αυτές αναφέρονται στα επίπεδα φωτεινότητας και θερμοκρασίας του φωτός, τη χρωματική απόχρωση της επιφάνειας εργασίας, το ύψος των αντικείμενων και τέλος την σύνδεση στο διαδίκτυο. Αρχικά το επίπεδο φωτεινότητας του χώρου λειτουργίας θα πρέπει να είναι τουλάχιστον 20 lux. Κάτω από αυτή την ένταση φωτός ο χώρος είναι αρκετά σκοτεινός και προκαλεί αδυναμία απεικόνισης και κατά προέκταση αναγνώριση των αντικειμένων. Επίσης οι ιδανικές συνθήκες για τη θερμοκρασία του φωτός είναι από 2400K με 6500K (Kelvin). Στο διάστημα αυτό το φως βρίσκεται σε λευκές αποχρώσεις οι οποίες δεν επηρεάζουν την απεικόνιση του χρώματος των αντικειμένων. Ακόμα σημαντικός παράγοντας για την αναγνώριση των αντικειμένων είναι η χρωματική απόχρωση της επιφάνειας εργασίας ή αλλιώς το χρώμα του background. Όπως έχει ήδη αναφερθεί στην Ενότητα 3.2 "Όραση μηχανής" προκειμένου να απομονωθούν τα αντικείμενα τόσο χρωματικά όσο και σχηματικά έχει οριστεί ως σύμβαση το χρώμα του background να είναι στις αποχρώσεις του άσπρου. Έτσι όποιο σημείο στην επιφάνεια εργασίας είναι άσπρο δεν θεωρείται από το πρόγραμμα αντικείμενο αλλά παρασκήνιο. Ενδεικτικά τα χρώματα που υποστηρίζονται απεικονίζονται στην εικόνα 3.21.

HTML color name	Sample		Hex triplet
	(rendered by name)	(rendered by hex triplet)	
White			#FFFFFF
Whitesmoke			#F5F5F5
Snow			#FFFAFA
Honeydew			#F0FFF0
Mintcream			#F5FFFA
Azure			#F0FFFF
Aliceblue			#F0F8FF
Ghostwhite			#F8F8FF
Seashell			#FFF5EE
Beige			#F5F5DC
Oldlace			#FDF5E6
Floralwhite			#FFFAF0
Ivory			#FFFFF0
Antiquewhite			#FAEBD7
Linen			#FAF0E6
Lavenderblush			#FFF0F5

Εικόνα. 3.21: Αποχρώσεις του άσπρου

Η σύμβαση για το μέγιστο μήκος της μίας πλευράς των αντικειμένων έχει γίνει λόγω του μήκους ανοίγματος του τελικού εργαλείου δράσης. Το Gripper 2 που γίνεται χρήση στην υλοποίηση, καταφέρνει μέγιστο άνοιγμα στα 6cm κάτι που περιορίζει τη μία πλευρά του αντικειμένου να είναι το πολύ 5.9cm. Ακόμα, το γεγονός πως δεν γίνεται χρήση στερεοσκοπικής κάμερας και κατά προέκταση υπολογισμός του βάθους καθιστά το σύστημα αδύνατο να υπολογίζει το ύψος των αντικειμένων. Έτσι η σύμβαση που έχει γίνει είναι ότι το μέγιστο ύψος των υποστηριζόμενων αντικειμένων να είναι το πολύ 6.5cm. Το ύψος αυτό προέκυψε βάση των διαστάσεων του τελικού εργαλείου δράσης. Τέλος, κατά τη λειτουργία του συστήματος απαιτείται σύνδεση στο διαδίκτυο. Η σύμβαση αυτή απαιτείται για την λειτουργία της αναγνώρισης φωνής, μιας και το σύστημα χρησιμοποιεί σύστημα αναγνώρισης σε ζωντανό χρόνο μέσω Google.

3.8 Σύνοψη κεφαλαίου

Με την ανάλυση των λειτουργιών των σεναρίων Vision, Speech, Control έφθασε στο πέρας το τρίτο κεφάλαιο. Συνοψίζοντας στο παρόν κεφάλαιο αναλύθηκε η συνολική λειτουργία του λογισμικού του συστήματος, καθώς και των επιμέρους υποσυστημάτων. Ακολουθεί το τέταρτο κεφάλαιο του συγγράμματος, στο οποίο ορίζονται και παρουσιάζονται τα πειράματα που διεξήχθησαν και τα μετρικά απόδοσης του συστήματος. Παράλληλα αναλύονται τα αποτελέσματα και γίνεται συζήτηση επ' αυτών.

Κεφάλαιο 4

Πειραματική Διαδικασία

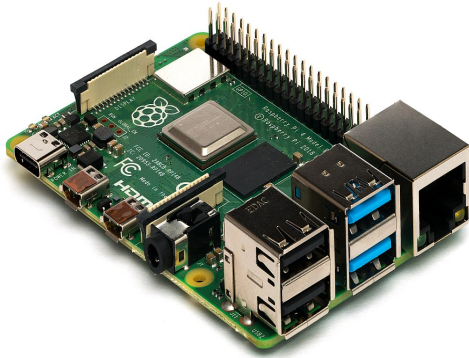
Σε αυτό το κεφάλαιο η παρούσα υλοποίηση θα δοκιμαστεί σε πραγματικά σενάρια χρήσης. Αρχικά θα οριστούν τα πειράματα που έλαβαν μέρος και τα χαρακτηριστικά των συστημάτων που εκτελέστηκαν. Ακολούθως, θα παρουσιαστούν φωτογραφίες με στιγμιότυπα από την εκτέλεση των αλγορίθμων αναγνώρισης αντικειμένων καθώς και την αρπαγή και παράδοση αυτών στο χρήστη. Τέλος, θα παρατεθούν τα αποτελέσματα των μετρήσεων και ο σχολιασμός αυτών.

4.1 Ορισμός πειραμάτων


Στη πειραματική διαδικασία που υλοποιήθηκε, εξετάστηκε η ακρίβεια, η συνέπεια και η απόκριση του συστήματος για δύο διαφορετικά συστήματα. Βασικό στοιχείο μελέτης και σύγκρισης είναι αν καθίσταται δυνατό και σε ποιό βαθμό να καλύψει τις απαιτήσεις επεξεργαστικών πόρων του συστήματος ένα ενσωματωμένο σύστημα. Η σύγκριση έγινε μεταξύ ενός συμβατικών επιδόσεων φορητό υπολογιστή και της πλακέτας επεξεργασίας Raspberry Pi 4.

Rpi4

Χαρακτηριστικά



- SoC: Broadcom BCM2711B0 quad-core A72 (ARMv8-A) 64-bit @ 1.5GHz
- GPU: Broadcom VideoCore VI
- Networking: 2.4 GHz and 5 GHz 802.11b/g/n/ac wireless LAN
- RAM: 4GB LPDDR4 SDRAM
- Bluetooth: Bluetooth 5.0, Bluetooth Low Energy (BLE)
- Storage: microSD
- Ports: 2 × micro-HDMI 2.0, 3.5 mm analogue audio-video jack, 2 × USB 2.0, 2 × USB 3.0, Gigabit Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)
- Operating System: Raspbian
- Dimensions: 88 mm × 58 mm × 19.5 mm, 46 g

Laptop	Χαρακτηριστικά
	<ul style="list-style-type: none"> • CPU: Intel Core i5-7200U dual core 2.5GHz (up to 3.1GHz) • GPU: Nvidia GeForce MX150 2GB GDDR5 • Memory: 8GB DDR4 • Storage: 2x 256GB Pcle SSD • Operating System: Windows 10 Pro 64-bit

Για τη πειραματική διαδικασία αναπτύχθηκαν 4 διαφορετικά σενάρια αλληλεπίδρασης με το ρομποτικό σύστημα. Ο συνδυασμός των 4 σεναρίων προβλέπουν στην εφαρμογή όλων των διαθέσιμων δυνατοτήτων και λειτουργιών που προσφέρει το σύστημα.

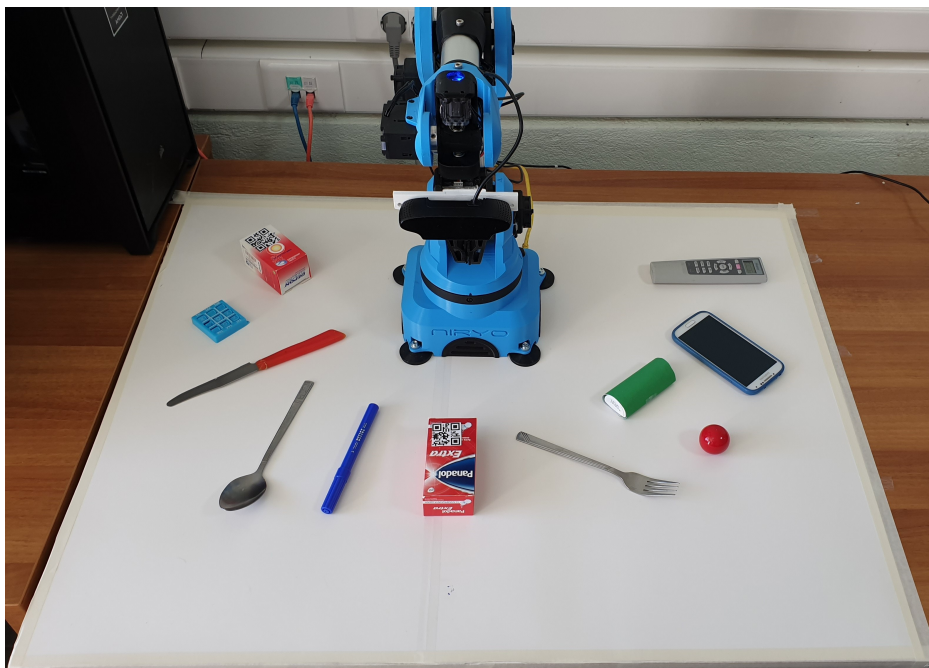
	Πείραμα 1	Πείραμα 2	Πείραμα 3	Πείραμα 4
Γλώσσα αλληλεπίδρασης:	Ελληνικά	Αγγλικά	Ελληνικά	Αγγλικά
Φωνή χρήστη:	Ανδρική	Ανδρική	Γυναικεία	Γυναικεία
Λειτουργία αναζήτησης:	Ναι	Όχι	Όχι	Ναι
Τρόπος παράδοσης:	Χέρι χρήστη	Χέρι χρήστη	Προκαθορισμένο σημείο	Χέρι χρήστη
Λειτουργία αναγνώρισης αντικειμένου:	Yolo	Yolo	Shape & Color	QR code

Πίνακας 4.1: Σχεδιασμός πειραμάτων.

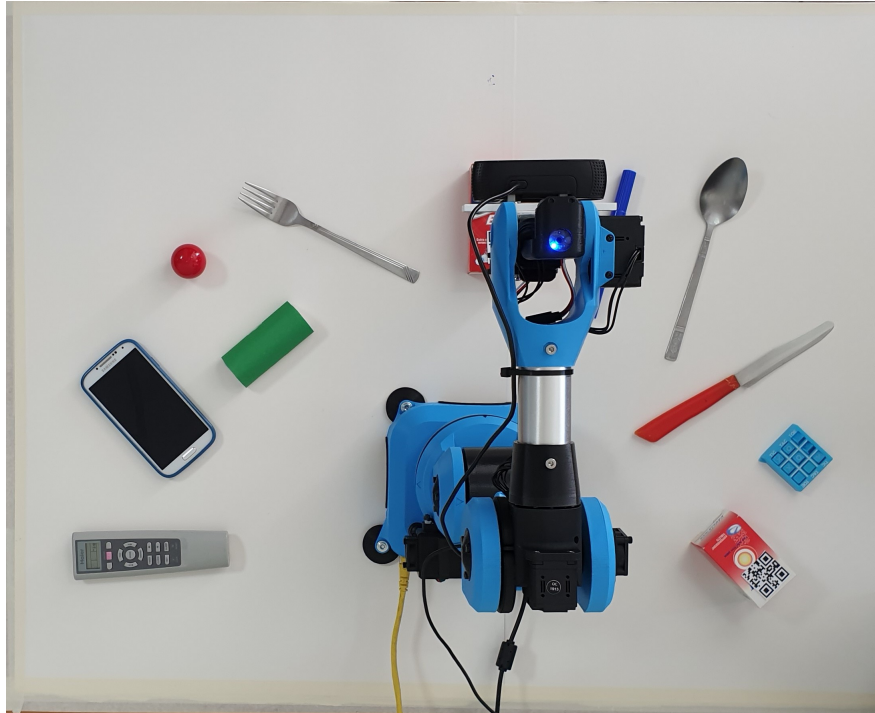
Κατά τη διάρκεια των πειραμάτων τα επίπεδα φωτεινότητας του χώρου ήταν στο αποδεκτό εύρος καλής λειτουργίας όπως και τα επίπεδα θορύβου περιβάλλοντος, ώστε να μην επηρεάσουν τα αποτελέσματα.

4.2 Εκτέλεση πειραμάτων

Η διεξαγωγή των πειραμάτων έγινε στο χώρο του Εργαστηρίου Ρομποτικής Ενσωματωμένων και Ολοκληρωμένων Συστημάτων του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας. Διαμορφώθηκε μια ειδική περιοχή για να καλύπτει τις απαιτήσεις του χώρου εργασίας, του φωτισμού, και του χρώματος παρασκήνιου των αντικειμένων. Στην περιβάλλουσα περιοχή του βραχίονα τοποθετήθηκαν αντικείμενα ικανά να εντοπιστούν από τις διαθέσιμες λειτουργίες αναγνώρισης. Το σύνολο των αντικειμένων που τοποθετήθηκαν ήταν 11 εκ των οποίων τα 5 αναγνωρίζονται από τον αλγόριθμο Yolo, τα 4 με τη λειτουργία αναγνώρισης σχήματος και χρώματος και τα 2 με την αναγνώριση QR code. Στην εικόνα παρουσιάζεται η περιοχή που έχει τοποθετηθεί ο βραχίονας και τα αντικείμενα στο χώρο.

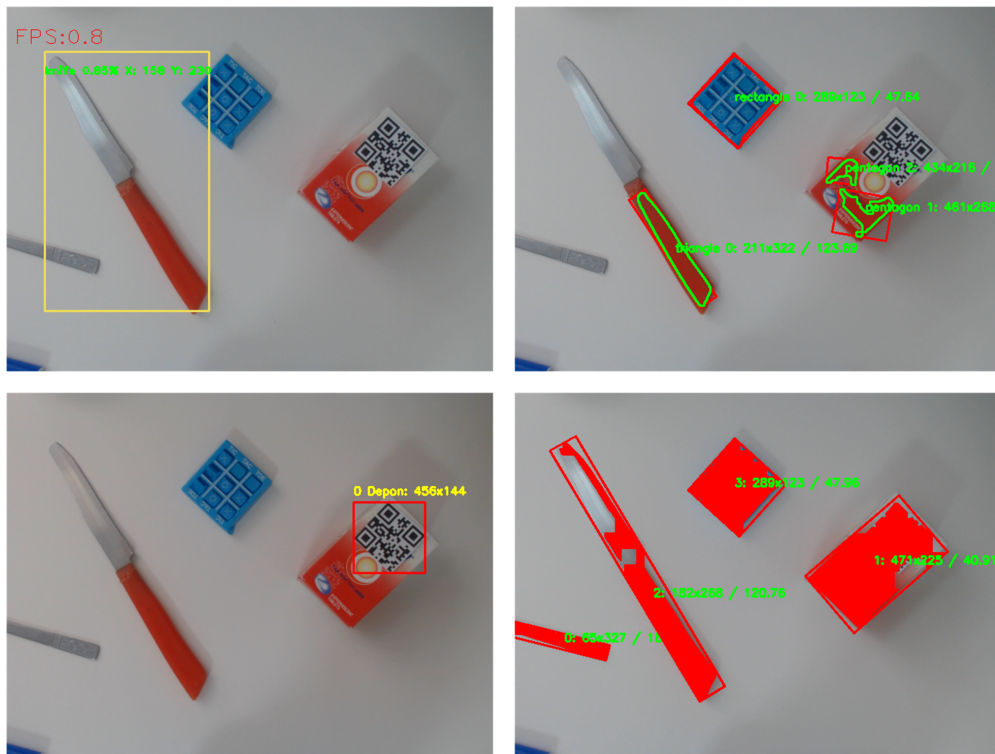


Εικόνα. 4.1: Αρχική κατάσταση έναρξης πειραμάτων



Εικόνα. 4.2: Κάτοψη αρχικής κατάστασης

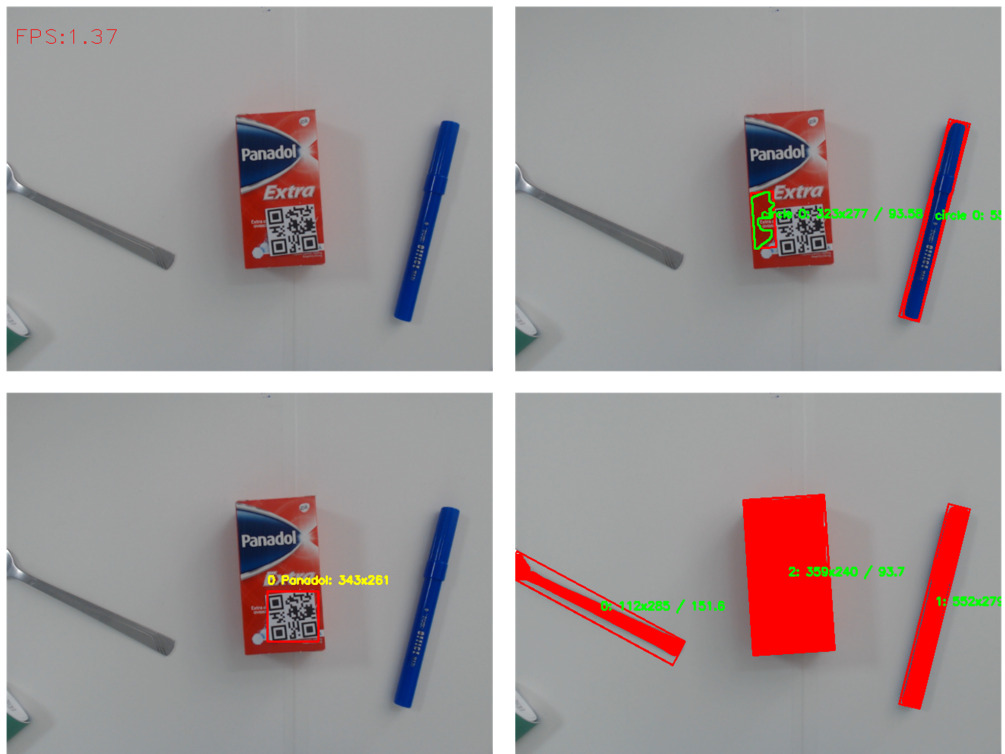
Στις επόμενες εικόνες θα παρουσιαστούν βήμα προς βήμα όλα τα στάδια κατά την εκτέλεση του πειράματος 4 από τον φορητό υπολογιστή. Αρχικά θα παρουσιαστούν τα στιγμιότυπα καταγραφής από όλες τις λειτουργίες αναγνώρισης αντικειμένων για κάθε θέση της λειτουργίας αναζήτησης.



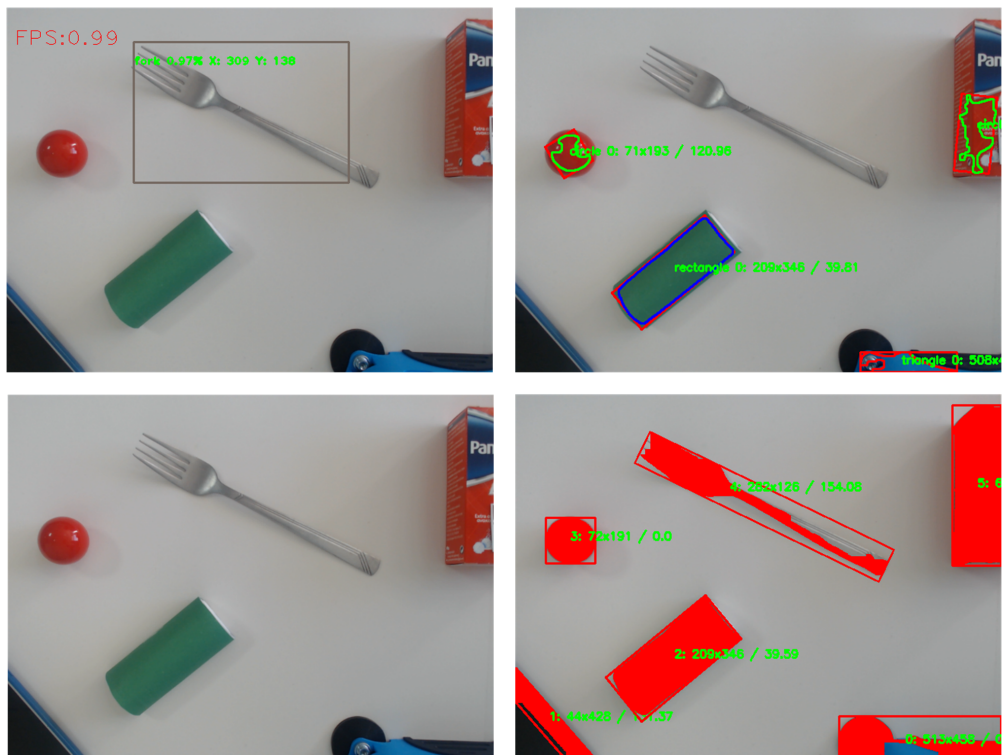
Εικόνα. 4.3: Κάτοψη θέσης Α



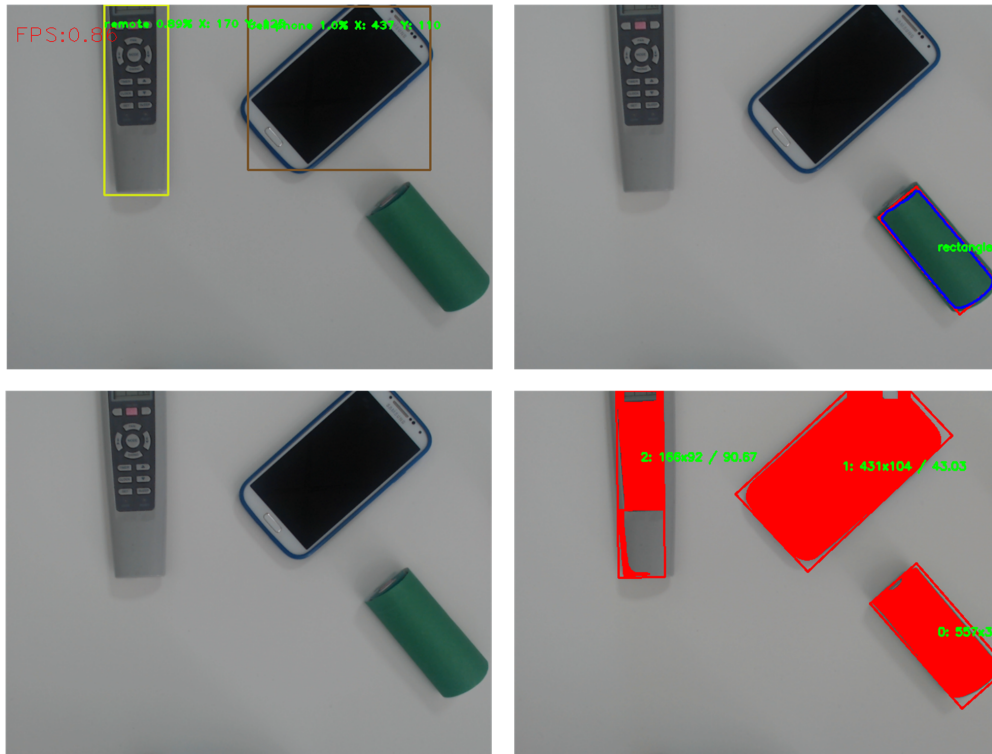
Εικόνα. 4.4: Κάτοψη θέσης Β



Εικόνα. 4.5: Κάτοψη θέσης Γ

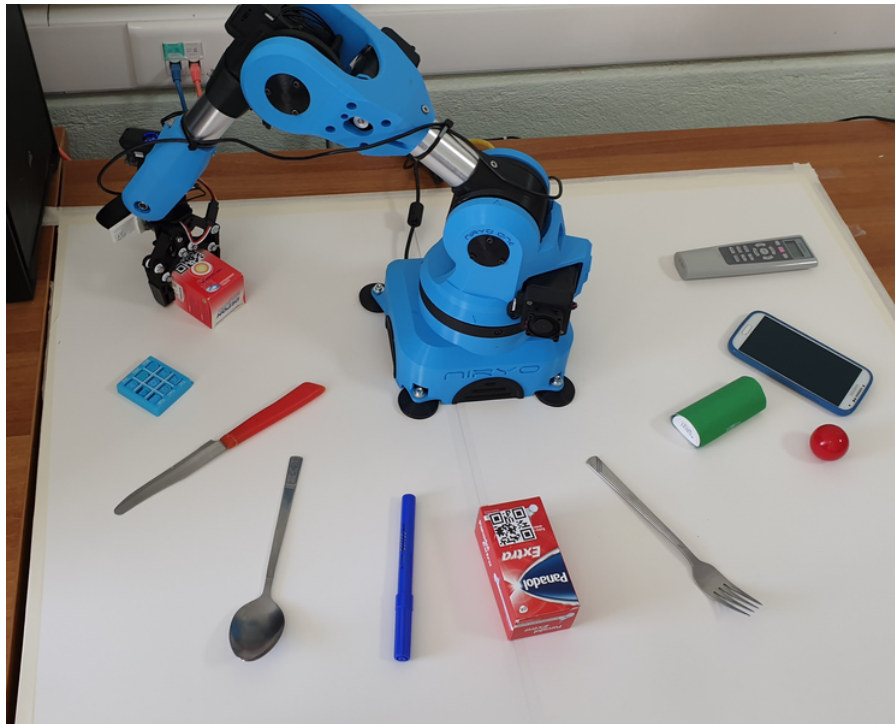


Εικόνα. 4.6: Κάτοψη θέσης Δ

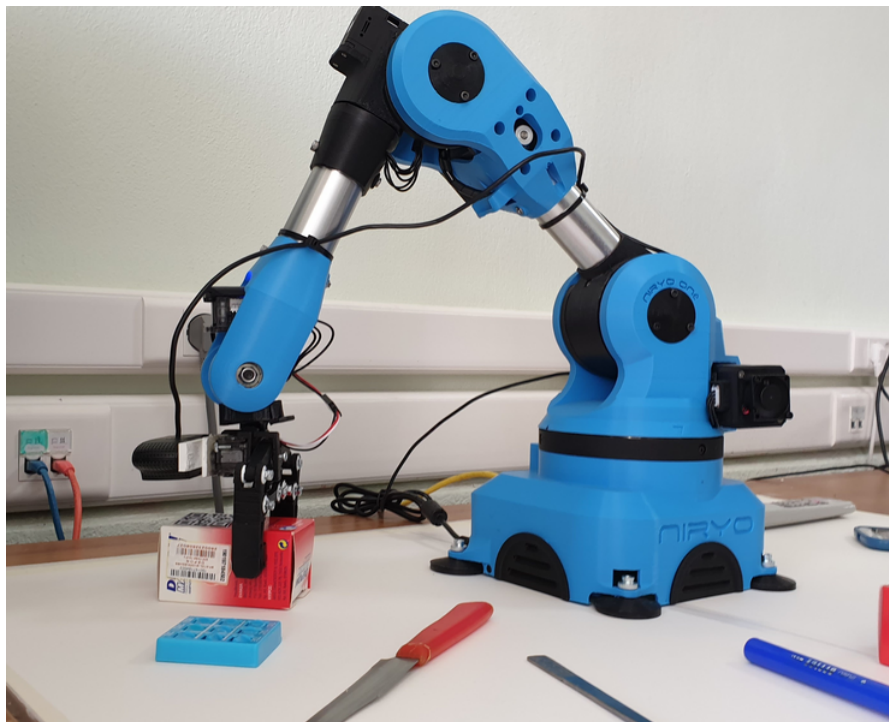


Εικόνα. 4.7: Κάτοψη θέσης E

Για το πείραμα 4 επιλέχθηκε αντικείμενο βάση της αναγνώρισης μέσω QR codes. Το αντικείμενο αυτό, όπως απεικονίζεται στις εικόνες 4.9, 4.8 να ανασηκώνεται από το βραχίονα είναι το αντικείμενο με όνομα ετικέτας «Depon».

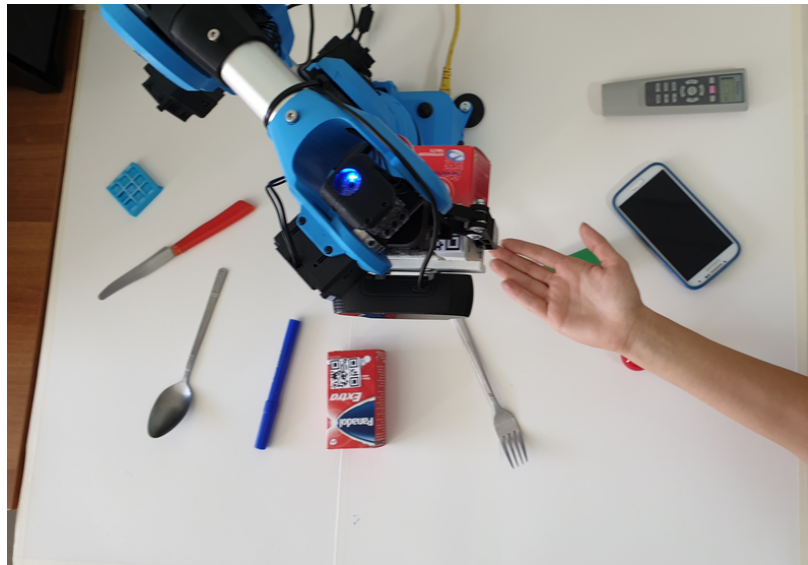


Εικόνα. 4.8: Πιάσιμο του αντικειμένου

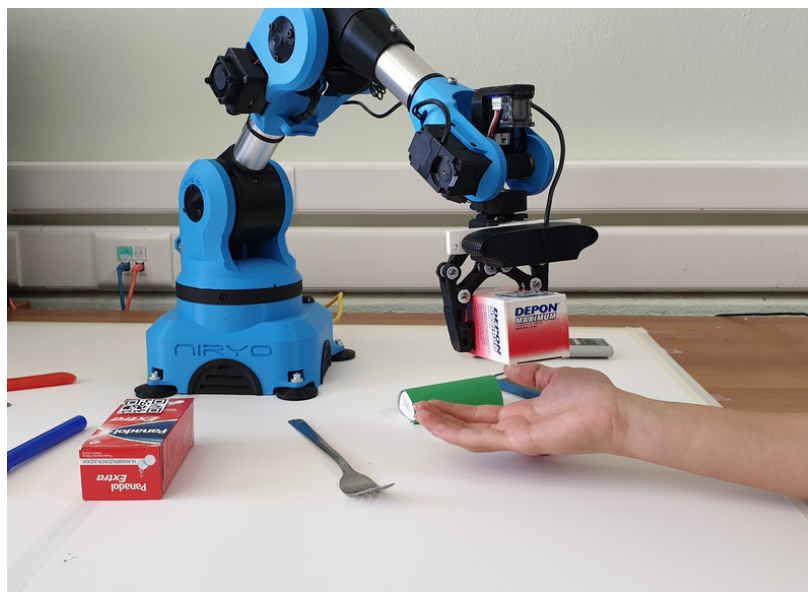


Εικόνα. 4.9: Πιάσιμο του αντικειμένου (μεγέθυνση)

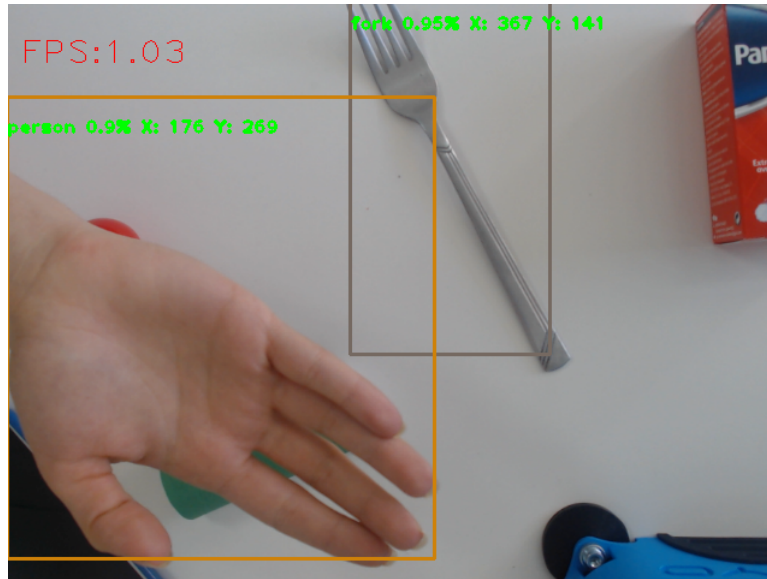
Ακολούθως ο βραχίονας θα αναζητήσει το χέρι του χρήστη για να το παραδώσει.



Εικόνα. 4.10: Παράδοση αντικειμένου



Εικόνα. 4.11: Παράδοση αντικειμένου (μεγέθυνση)



Εικόνα. 4.12: Αναγνώριση χεριού

Η ροή εκτέλεσης του λογισμικού κατά την εκτέλεση του πειράματος 4 παρατίθεται στο στιγμιότυπο εικόνας

```
Which language do you want to use? For English, please say English, for Greek, say Greek
Ποιά γλώσσα θέλετε να χρησιμοποιήσετε; Για την Αγγλική, πείτε Αγγλικά, για την Ελληνική, πείτε Ελληνικά
Please say the language you prefer
english
You said : english
Please say what do you want
search
You said : search
Searching operation has been started succesfully
Searching operation has been finished. The following items have been found
Depon, blue rectangle, knife, spoon, blue circle, panadol, red circle, green rectangle, fork, remote, cellphone
Please say what do you want
can you pick up depon
action: pickup
object: depon
Barcode: depon mapped at x:456 y:144 deg:40.91 frame:1
Pick up operation has been started
Searching for person
Searching operation has been finished. person mapped at x:176 y:269 frame:4
Delivery operation has been started
Successful delivered to person
Please say what do you want
stop
You said : stop
Program finished successfully
Process finished with exit code 0
```

Εικόνα. 4.13: Ροή εκτέλεσης του λογισμικού

4.3 Αποτελέσματα και συζήτηση

Τα μετρικά που επιλέχθηκαν να καταγραφούν σχετίζονται με παραμέτρους που αναδεικνύουν τις ειδοποιούς διαφορές μεταξύ των δύο συστημάτων οι οποίες επηρεάζουν την απόδοση στην εκτέλεση του προγράμματος. Τα αποτελέσματα των πειραμάτων από τα δύο συστήματα παρουσιάζονται στους πίνακες 4.2 και 4.3.

	Laptop			
	Πείραμα 1	Πείραμα 2	Πείραμα 3	Πείραμα 4
Μέση τιμή fps κατά την εκτέλεση του Yolo	1,51	1,45	1,42	1,38
Αποτυχημένες προσπάθειες αναγνώριση φωνητικής εντολής	0	1	1	0
Αριθμός λανθασμένων ή μη αναγνωρισμένων αντικειμένων από σύνολο 11 αντικειμένων	1	0	1	0
Συνολικός χρόνος αλληλεπίδρασης	90	53	32	92

Πίνακας 4.2: Αποτελέσματα πειραμάτων στο φορητό υπολογιστή

	Raspberry Pi 4			
	Πείραμα 1	Πείραμα 2	Πείραμα 3	Πείραμα 4
Τιμή fps κατά την εκτέλεση του Yolo	0,46	0,42	0,36	0,38
Αποτυχημένες προσπάθειες αναγνώριση φωνητικής εντολής	0	1	2	0
Αριθμός λανθασμένων ή μη αναγνωρισμένων αντικειμένων από σύνολο 11 αντικειμένων	2	2	3	2
Συνολικός χρόνος αλληλεπίδρασης	104	58	42	101

Πίνακας 4.3: Αποτελέσματα πειραμάτων στο Raspberry Pi 4

Κατά τη μελέτη του πλήθους fps που κατάφερε το σύστημα να επεξεργαστεί στην εκτέλεση του αλγόριθμου Yolo, παρατηρούνται μικρές τιμές και στα δύο συστήματα. Αυτό μπορεί να αποδοθεί στο γεγονός ότι ο αλγόριθμός εκτελείται από τον επεξεργαστή του συστήματος και όχι από τη κάρτα γραφικών, γεγονός που μειώνει αισθητά την απόδοση του. Υπάρχει αισθητή γενικευμένη μείωση αυτού του πλήθους στην πλακέτα επεξεργασίας Raspberry Pi 4 το οποίο έγκειται στο γεγονός πως ο επεξεργαστής που φέρει είναι σαφώς χαμηλότερων επιδόσεων από ένα φορητό υπολογιστή. Παράλληλα παρατηρείται σταδιακή μείωση των καρτέ με το πέρας των πειραμάτων και στα δύο συστήματα. Το γεγονός αυτό δύναται να οφείλεται στην παρατεταμένη υψηλή χρήση του CPU το οποίο οδηγεί σε υψηλή θερμοκρασία και την μείωση ισχύος.

Το μετρικό αποτυχημένων αναγνωρίσεων φωνητικής εντολής προκύπτει κάθε φορά που δεν αναγνωριστεί επιτυχώς η λέξη ή υπάρχει αδυναμία αναγνώρισης από το API. Τα αποτελέσματα στο άθροισμα των τιμών μεταξύ των δύο συστημάτων διαφέρει κατά 1. Παρατηρείται συνέπεια σε λανθασμένη αναγνώριση για το πείραμα 3 και από τα δύο συστήματα. Πιθανό αίτιο είναι ότι στο πείραμα 3 το

αντικείμενο για παράδοσή πρέπει να χαρακτηριστεί από το χρώμα και από το σχήμα του, γεγονός που αυξάνει τις πιθανότητες για αποτυχημένη αναγνώριση. Ακόμα μπορεί να αποτελεί τυχαίο γεγονός που έγκειται στο δίκτυο και στα αιτήματα που δέχεται το API της google ή ακόμα από κάποιο ήχο περιβάλλοντος που εμπόδισε τη σωστή εγγραφή της εντολής. Στο σύνολο όμως των αποτελεσμάτων δεν παρατηρήθηκε συνεχόμενη αδυναμία αναγνώρισης.

Ο αριθμός μη αναγνωρισμένων αντικειμένων αναφέρεται στο πλήθος των αντικειμένων που δεν αναγνωρίστηκαν επιτυχώς από το πρόγραμμα. Σε σύγκριση με τον φορητό υπολογιστή στο Raspberry Pi 4 υπάρχει εμφανώς αυξημένο ποσοστό με μέσο όρο ποσοστού 20%. Το γεγονός αυτό προκύπτει ως επακόλουθο των λίγων fps που επεξεργάζεται το σύστημα. Ως αποτέλεσμα είναι να μην προλαβαίνει να αναγνωρίσει τα αντικείμενα ή να αναγνωρίζονται λανθασμένα. Επίσης ενιαίος παράγοντας μη αναγνώρισης αντικειμένου διατελεί το focus της κάμερας, το οποίο μπορεί να θολώσει το αντικείμενο και να προκαλέσει αδυναμία αναγνώρισης.

Ο συνολικός χρόνος αλληλεπίδρασης αναφέρεται στο χρόνο που έχει παρέλθει από την αρχή εκτέλεσης του προγράμματος μέχρι τη στιγμή της παράδοσης του αντικειμένου. Κατ' αντιστοιχία τα πειράματα που έχουν εκτελεστεί από τον φορητό υπολογιστή έχουν λιγότερο χρόνο. Επίσης όπως είναι φυσιολογικό τα πειράματα που δεν περιλαμβάνουν λειτουργία αναζήτησης και την παράδοση του αντικειμένου στον χρήστη εκτελούνται σε λιγότερο χρόνο.

Τα αποτελέσματα των μετρικών όσο αφορά την επεξεργαστική πλακέτα Raspberry Pi 4 θα μπορούσαν να χαρακτηριστούν αρκετά ικανοποιητικά δεδομένων των επιδόσεων. Η επεξεργαστική κατωτερότητα του Raspberry Pi 4 δεν φάνηκε έντονα στα αποτελέσματα των μετρικών καταφέροντας έτσι μπορεί να ανταγωνιστεί τις επιδόσεις ενός φορητού υπολογιστή με κάποιες μικρές παραχωρήσεις όσο αφορά την αξιοπιστία.

Στο σύνολο των αποτελεσμάτων το σύστημα που αναπτύχθηκε ανταπεξήλθε των προσδοκιών και τα ποσοστά απόδοσής ήταν αρκετά υψηλά κυρίως στην εκτέλεση από το φορητό υπολογιστή. Οι αλγόριθμοι και οι μέθοδοι που αναπτύχθηκαν για την αναγνώριση αντικειμένων, οι παραμετροποιήσεις στην αναγνώριση της ομιλίας καθώς και το σύνολο των τεχνικών που εφαρμόστηκαν κατά την ανάπτυξη του προγράμματος συνέβαλαν στη ομαλή και βέλτιστη απόδοση του συστήματος.

Κεφάλαιο 5

Συμπεράσματα και μελλοντικές επεκτάσεις

Η παρούσα διπλωματική εργασία πραγματοποιείται την υλοποίηση ενός βοηθού ρομποτικού βραχίονα φωνητικής αλληλεπίδρασης, χρησιμοποιώντας όραση μηχανής και αλγόριθμο βαθιάς μάθησης για την αναγνώριση των αντικειμένων.

Η εκπόνηση της παρούσας διπλωματικής διαμοιράστηκε σε 5 στάδια. Το πρώτο στάδιο ήταν η ανάπτυξη του λογισμικού μέρους του συστήματος σε γλώσσα προγραμματισμού Python για την ευρέως εκτέλεση από τα περισσότερα λειτουργικά συστήματα. Η ανάπτυξη αφορούσε κυρίως το κομμάτι της μηχανικής μάθησης και της φωνητικής αλληλεπίδρασης. Το δεύτερο στάδιο ήταν η ανάπτυξη του σενάριου κώδικα για τη εκτέλεση των κινήσεων (Control) από το ρομποτικό βραχίονα. Καθ' όλη τη διαδικασία ανάπτυξης υπήρχε άμεση δοκιμή του κώδικα από το βραχίονά Niryo-One για τη καλύτερη παραμετροποίηση. Στο τρίτο στάδιο βρίσκεται η μεταφορά του προγράμματος στο Raspberry Pi 4 και αντιμετώπιση των προβλημάτων που προέκυψαν. Στο τέταρτο στάδιο ανήκουν τα πειράματα και δοκιμή του συστήματος σε πραγματικές συνθήκες για την βελτίωση της εμπειρίας αλληλεπίδρασης με τον χρήστη. Στο τελευταίο στάδιο έγινε η συγγραφή της διπλωματικής και η λήψη των φωτογραφιών παρουσίασης του ρομποτικού βραχίονα.

5.1 Σύνοψη και συμπεράσματα

Το ρομποτικό σύστημα που υλοποιείται στην εν λόγω διπλωματική εργασία δημιουργήθηκε με σκοπό την παροχή βοήθειας, μεταβιβάζοντας αντικείμενα στο χρήστη. Η υλοποίηση βασίστηκε στον ρομποτικό βραχίονα Niryo-One με τον χειρισμό να γίνεται μέσω φωνητικών εντολών από το χρήστη, εκφωνώντας το αντικείμενο που επιθυμεί να του μεταβιβαστεί. Η αναγνώριση των αντικειμένων γίνεται από το πρόγραμμα αναγνώρισης που έχει αναπτυχθεί μέσω της USB κάμερας που φέρει προσαρτημένη ο βραχίονας στην περιοχή του τελικού εργαλείου δράσης. Για την αναγνώριση των αντικειμένων έχουν αναπτυχθεί 3 διαφορετικοί μέθοδοι. Οι μέθοδοι αυτοί περιλαμβάνουν την αναγνώριση του σχήματος και του χρώματος ενός αντικειμένου μέσω κατάτμησης και μορφολογικών χειρισμών, το είδος του αντικειμένου μέσω του αλγόριθμου αναγνώρισης αντικειμένων Yolo και τέλος της ετικέτας QR code που φέρουν. Με τη μέθοδο αναγνώρισης QR code ανοίγει ένα παράθυρο εντοπισμού για οποιοδήποτε αντικείμενο και εξατομίκευσης του ονόματος προσφώνησης βάση των προσωπικών προτιμήσεων.

Διατίθενται δύο επιλογές λειτουργίας, της αναζήτησης και της απευθείας παράδοσης. Στη πρώτη ο ρομποτικός βραχίονας ελέγχει το χώρο εργασίας για τον εντοπισμό των διαθέσιμων αντικειμένων και στη συνέχεια τα αποτελέσματα εκφωνούνται στον χρήστη, ενώ στη δεύτερη ο χρήστης μπορεί να ζητήσει απευθείας ένα αντικείμενο για παράδοση. Η παράδοση του αντικειμένου γίνεται στο χέρι του χρήστη εφόσον εντοπιστεί, αλλιώς σε ένα προκαθορισμένο σημείο.

Η ιδιαιτερότητα του συστήματος εντοπίζεται αρχικά στην εύκολη και άμεση επικοινωνία με το χρήστη και στο ευρύτατο πλήθος αναγνώρισης αντικειμένων που δύναται να εντοπιστούν.

Πέρα από τα πλεονεκτήματα που έχει η παρούσα διπλωματική, υπάρχει κι ένα ερευνητικό υπόβαθρο πίσω από την ανάπτυξη της αναγνώρισης αντικειμένων. Για την αναγνώριση του είδους των αντικειμένων επιλέχθηκε αλγόριθμος αναγνώρισης αντικειμένων YOLO, παρόλα αυτά ο αλγόριθμος δεν εξετάζει τη γωνία του αντι-

κειμένου στο χώρο, χαρακτηριστικό απαραίτητο για τις ανάγκες της εφαρμογής. Για την κάλυψη αυτής της ανάγκης αναπτύχθηκε αλγόριθμος εντοπισμού εύρεσης της γωνίας θέσης των αντικειμένων που ανήκουν στην λίστα που αναγνωρίζονται από τον αλγόριθμο Yolo καθώς και των αντικειμένων που φέρουν ετικέτα QR code, επιφέροντας μια προέκταση που δεν είχε υλοποιηθεί.

Επιπροσθέτως, δοκιμές έχουν γίνει και για τις τιμές του κατωφλιού σε όλα τα σημεία που αυτό έχει χρησιμοποιηθεί. Οι τιμές αυτές ποικίλουν από χώρο σε χώρο λόγω έντασης του φωτός και εισαγωγής θορύβου στην εικόνα, γι' αυτό χρειάστηκε αρκετός καιρός και πειράματα μέχρι να βρεθούν οι τελικές τιμές για την κάθε εφαρμογή. Για τον υπολογισμό των τιμών αυτών έχουν ληφθεί υπόψη και οι διαφορές στα χαρακτηριστικά της κάθε κάμερας που χρησιμοποιείται.

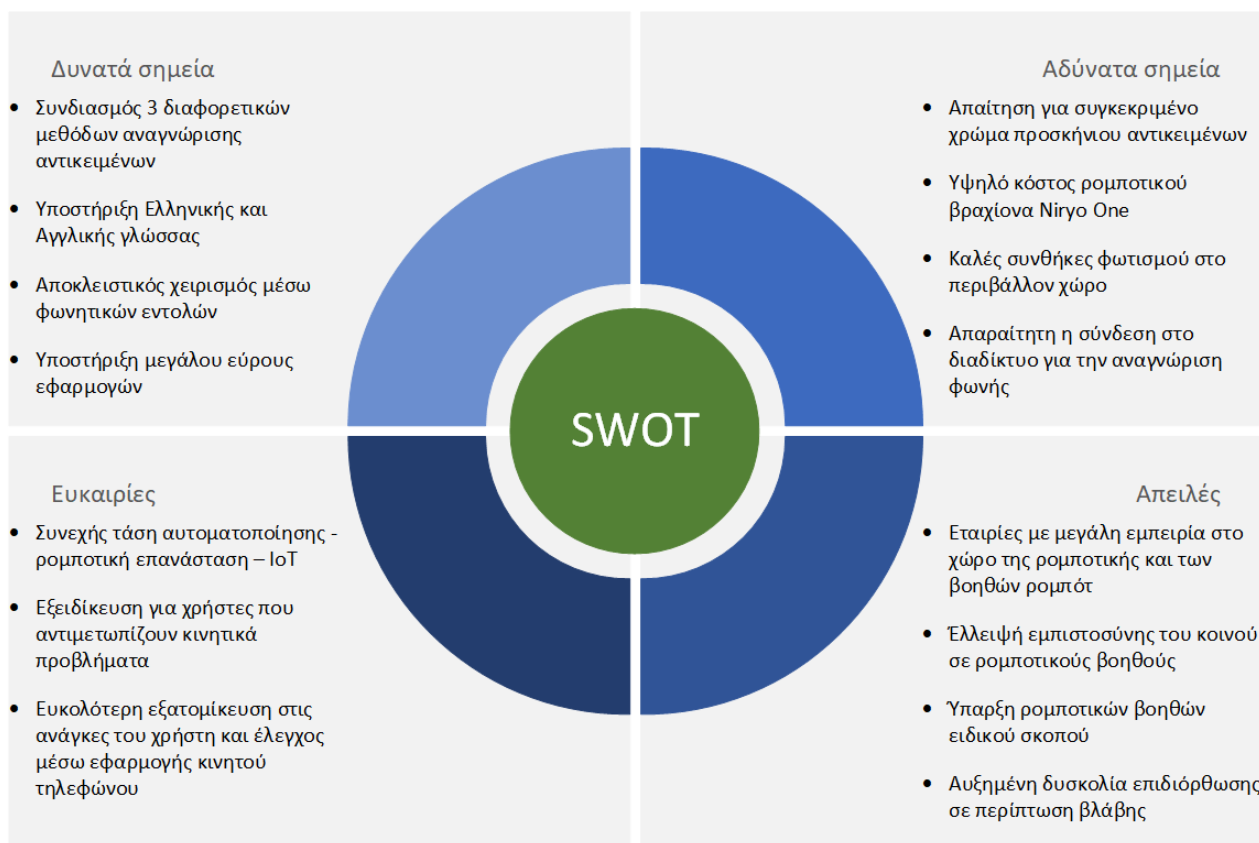
Παράλληλα, κατά τη διαδικασία των πειραμάτων εξετάστηκε η απόδοση και η αποτελεσματικότητα του προγράμματος σε δύο διαφορετικά υπολογιστικά συστήματα. Η σύγκριση μεταξύ ενός φορητού υπολογιστή και μίας ευρέως διαδεδομένης πλακέτας επεξεργασίας όπως το Raspberry Pi 4 και κατά προέκταση η βελτιστοποίηση σε αυτή αποφέρει ερευνητικές προεκτάσεις για τη δυνατόν βέλτιστή αξιοποίηση των δυνατοτήτων του ενσωματωμένου συστήματος.

Τα αποτελέσματα των πειραμάτων που διεξήχθησαν επιβεβαίωσαν την ορθή λειτουργία του συστήματος με το ποσοστό επιτυχημένης αναγνώρισης αντικειμένων από τον φορητό υπολογιστή να είναι στο 95,45% ενώ για το Raspberry Pi 4 να ανέρχεται στο 79,54%.

Εν κατακλείδι, το παρόν ρομποτικό σύστημα αποτελεί μια καινοτόμα προσέγγιση ενός βοηθού που ανταποκρίνεται στις ανάγκες της σημερινής εποχής με πολλές εφαρμογές συνδυάζοντας τεχνολογίες και μεθόδους με έντονο ερευνητικό ενδιαφέρον για τα επόμενα χρόνια.

5.2 Μοντέλο ανάλυσης συστήματος (S.W.O.T)

Η ανάλυση SWOT είναι αποτελεσματική τεχνική ανάλυσης των δυνατών σημείων, των αδυναμιών, των ευκαιριών και των απειλών που έχει το σύστημα το οποίο έχει αναπτυχθεί. Η ονομασία της είναι ακρωνύμιο των αντίστοιχων αγγλικών λέξεων (Strengths, Weaknesses, Opportunities, Threats). Η ανάλυση των χαρακτηριστικών βοηθά αρχικά στον καθορισμό των απαιτήσεων του συστήματος και στην μετέπειτα βελτίωσή του. Κάθε σύστημα έχει εσωτερικούς (δυνατά και αδύναμα σημεία του συστήματος) και εξωτερικούς (ευκαιρίες και απειλές) παράγοντες, δηλαδή θετικά και αρνητικά χαρακτηριστικά τα οποία επηρεάζουν την εξέλιξη του και ίσως, σε κάποιες περιπτώσεις, μειώνουν τη λειτουργικότητά του.



Εικόνα. 5.1: Ανάλυση συστήματος (S.W.O.T)

5.3 Προβλήματα που προέκυψαν κατά την εκπόνηση

Κατά τη διάρκεια εκπόνησης της διπλωματικής εργασίας προέκυψαν διάφορων ειδών ζητήματα τόσο σε επίπεδο υλικού, όσο και λογισμικού τα οποία αποτέλεσαν πρόκληση ως προς τον τρόπο προσέγγισης και επίλυσης τους.

Το πρώτο πρόβλημα που έπρεπε να ξεπεραστεί είναι ο θόρυβος των εικόνων που λαμβάνονται από την κάμερα και η αλλοίωση του εντοπισμού από τις αντανακλάσεις του φωτός σε μεταλλικά αντικείμενα ή αντικείμενα με γυαλιστερή επιφάνεια. Ο θόρυβος προερχόταν είτε από την ένταση του φωτός στο χώρο ή από την επεξεργασία της εικόνας ενώ οι αντανακλάσεις εμφανίζονταν αναπόφευκτα λόγω του υλικού κατασκευής των αντικειμένων.

Αρχικά, για να απομακρυνθεί ο θόρυβος αυτός εφαρμόστηκαν κατάλληλα φίλτρα και τεχνικές με τις οποίες εξομαλύνεται η εικόνα και εξάγονται τα προσδοκώμενα δεδομένα. Επίσης, για να βρεθεί το κατάλληλο κατώφλι για τη μείωση του θορύβου, έγινε δοκιμή διαφόρων τιμών ώστε να βρεθούν οι κατάλληλες γενικευμένες, που μπορούν να εφαρμοστούν σε διαφορετικές συνθήκες φωτισμού. Για την απομάκρυνση των αντανακλάσεων και της αλλοίωσης του περιγράμματος, εφαρμόστηκαν κατάλληλοι παράμετροι στις συναρτήσεις επεξεργασίας και πλήθος δοκιμών. Το αποτέλεσμα εγγυάται την ορθό σχεδιασμό των περιγραμμάτων ακόμα και σε συνθήκες έντονου φωτισμού.

Πρέπει να σημειωθεί ότι τα προβλήματα που προκύπτουν από το φως μπορεί να δημιουργήσουν τεράστια προβλήματα κατά την εκτέλεση προγραμμάτων που βασίζονται στην επεξεργασία εικόνων.

Ακόμα μια πρόκληση αποτέλεσε η ανάγκη για αλλαγή του εργαλείου αρπαγής. Μιας και ο ρομποτικός βραχίονας είχε ήδη τοποθετημένο το Gripper 1 και όχι το Gripper 2 που επιλέχθηκε να χρησιμοποιηθεί, απαιτήθηκε η εκτύπωση και η συναρμολόγηση του. Πέραν των δεδομένων αυτών ενεργειών αναγκαίο ήταν να γίνουν κάποιες αλλαγές σε configuration αρχεία που έτρεχαν στη ενσωματωμένη

πλακέτα Raspberry Pi του Niryo-One. Οι αλλαγές αναφερόντουσαν στο ID του εργαλείου που χρησιμοποιεί ο βραχίονας. Μιας και κάθε εξάρτημα του βραχίονα χαρακτηρίζεται από ένα μοναδικό ID προκειμένου να εφαρμοστούν κατάλληλες ρυθμίσεις για το καθένα θα έπρεπε για την αλλαγή από το Gripper 1 στο Gripper 2 να γίνει η αλλαγή του ID. Για την αλλαγή χρειάστηκε έρευνα στο διαδίκτυο για λήψη πληροφοριών μιας και η εταιρεία δεν παρείχε κάποιον οδηγό με τα απαραίτητα βήματα. Η αλλαγή έγινε με σύνδεση ssh στο Raspberry Pi και επεξεργασία των κατάλληλων αρχείων όπως φαίνεται και στο παρακάτω στιγμιότυπο.

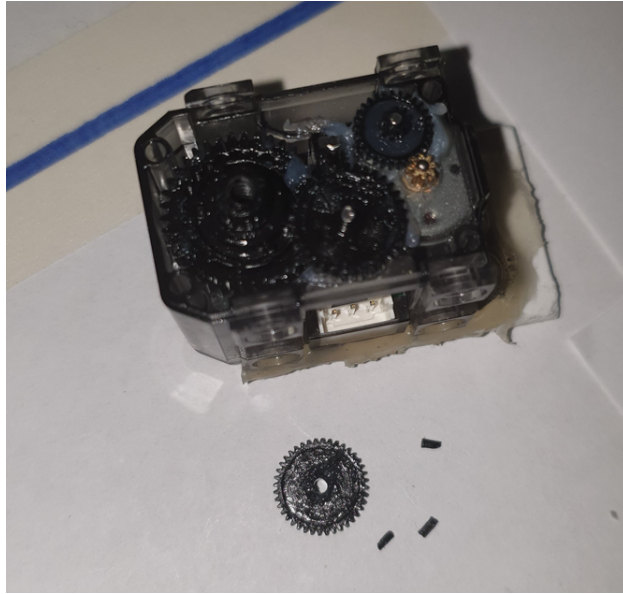
```
##
#
# List of all required and authorized motors (motors ids) for Niryo One V2
#
# (please read carefully before editing anything)
#
# You should not need to edit this file, unless
# - you want to debug on some detached motors
# - you are developing new features
# - you really know what you are doing
#
#
can_required_motors: # Axis 1-3 of Niryo One (stepper 1 -> id 1, stepper 2 -> id 2, ...)
# Edit only for debug purposes (ex : you want to test some motors separately and detached from the robot)
# --> Commented ids will make associated motor disable (and thus not trigger an error if not connected)
# Before editing, please be sure that you know what you're doing
- 1 # Axis 1 enabled if not commented
- 2 # Axis 2 enabled if not commented
- 3 # Axis 3 enabled if not commented

dxi_required_motors: # axis 4, 5 and 6 of Niryo One.
# Edit only for debug purposes (ex : you want to test some motors separately and detached from the robot)
# --> Commented ids will make associated motor disable (and thus not trigger an error if not connected)
# Before editing, please be sure that you know what you're doing
- 2 # -> id of Axis 4
- 3 # -> id of Axis 5
- 6 # -> id of Axis 6

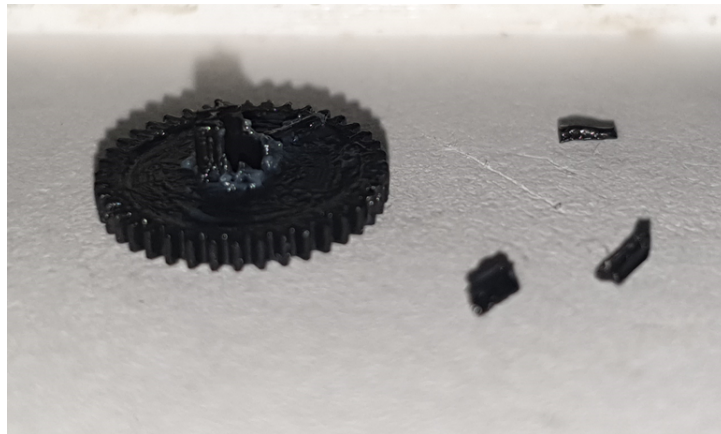
dxi_authorized_motors: # here include all Dynamixel tools that can possibly be attached to Niryo One
- 2 # -> id of Axis 4
- 3 # -> id of Axis 5
- 6 # -> id of Axis 6
- 11 # id of Gripper 1
- 12 # id of Gripper 2
- 13 # id of Gripper 3
- 31 # id of Vacuum Pump 1
```

Εικόνα. 5.2: Τερματικό περιβάλλον για την αλλαγή του ID

Ένα πρόβλημα που θα έθετε σε αναβολή της διαδικασίας εκτέλεσης πειραμάτων, ήταν το σπάσιμο ενός γραναζιού από τον κινητήρα servo που είναι υπεύθυνος για το άνοιγμα και κλείσιμο του εργαλείου αρπαγής. Πιθανότατα οι εκτενείς δοκιμές και το βάρος κάποιων αντικειμένων να προκάλεσαν το σπάσιμο ενός κομματιού από το διπλό γρανάζι κίνησης του κινητήρα. Όπως φαίνεται και στις παρακάτω εικόνες το σπάσιμο ήταν σε μεγάλο μέρος, γεγονός που έθετε εκτός λειτουργίας το εργαλείο αρπαγής.



Εικόνα. 5.3: Σπασμένο γρανάζι



Εικόνα. 5.4: Σπασμένο γρανάζι (μεγέθυνση)

Για την επίλυση χρειάστηκε να αφαιρεθεί το αντίστοιχο κομμάτι που έσπασε από άλλο γρανάζι και η προσαρμογή του στο σπασμένο. Η μεταποίηση αυτή είχε ως αποτέλεσμα εκτός της επισκευής αλλά και την αναβάθμιση του κινητήρα. Η αναβάθμιση έγινε σε επίπεδο αντοχής βάρους καθώς το τμήμα που τοποθετήθηκε ήταν μεταλλικό σε σχέση με το πλαστικό που προϋπήρχε.

5.4 Μελλοντικές επεκτάσεις

Κατά την εκπόνηση της διπλωματικής προέκυψαν διάφορες σκέψεις και αναζητήσεις σχετικά με τη βελτιστοποίηση και την αύξηση αξιοπιστίας του συστήματος. Οι επεκτάσεις αναφέρονται τόσο στο τομέα του υλικού όσο και του λογισμικού.

Αναφορικά με το τομέα του υλικού, για την αντιμετώπιση ανεπαρκούς φωτισμού του χώρου, ο οποίος αναπόφευκτα οδηγεί σε αδυναμία αναγνώρισης των αντικειμένων, η προσθήκη μιας πηγής φωτός LED στο τελικό εργαλείο δράσης του βραχίονα θα μπορούσε να επιλύσει το συγκεκριμένο πρόβλημα. Μια ακόμη επέκταση θα μπορούσε να αφορά την ενσωμάτωση ενός μαγνητικού διακόπτη στο τελικό εργαλείο δράσης, για την ενημέρωση του συστήματος σε περίπτωση που η δαγκάνα αποτύγχανε να πιάσει το αντικείμενο ή εμφάνιζε δυσλειτουργία. Με την ενημέρωση αυτή το σύστημα θα είχε τη δυνατότητα να προβεί σε επανάληψη εκτέλεσης της διαδικασίας αρπάγης. Ενδιαφέρουσα υλοποίηση θα αποτελούσε η αντικατάσταση της USB κάμερας με μία στεροσκοπική ή προσθήκη ενός ultrasonic αισθητήρα έτσι ώστε να εντοπίζεται το ύψος των αντικειμένων και κατ' προέκταση να υπολογίζεται η κατάλληλη απόσταση τοποθέτησης της δαγκάνας από το επίπεδο κατά τη διαδικασία αρπαγής.

Τα αποτελέσματα των δοκιμών που διεξήχθησαν αποτελούν κίνητρο για την εκτέλεση του προγράμματος σε πλακέτες που ειδικεύονται στην εκτέλεση αλγορίθμων νευρωνικών δικτύων, ταξινόμησης εικόνας, ανίχνευση αντικειμένων και επεξεργασίας λόγου. Μια τέτοια πλακέτα αποτελεί η Nvidia Jetson Nano η οποία διατηρεί το μικρό μέγεθος όπως το Raspberry Pi 4 και ταυτόχρονα επιφέρει την κατακόρυφη αύξηση των καρέ επεξεργασίας και του ποσοστού αναγνώρισης.

Ενδιαφέρουσα υλοποίηση θα αποτελούσε η ενσωμάτωση αναγνώρισης περισσότερων χρωμάτων ή η δυνατόν εκτίμηση του προγράμματος για το χρώμα των αντικειμένων, καθώς και η αναγνώριση τους σε οποιοδήποτε χρωματική απόχρωση φόντου του περιβάλλοντα χώρου.

Τέλος, επέκταση με ιδιαίτερη σημασία στην προσθήκη χρηστικών λειτουργιών, είναι η ανάπτυξη εφαρμογής για smartphone η οποία θα επιτρέπει τον χρήστη να έχει οπτική απεικόνιση του χώρου σε ζωντανό χρόνο ώστε να χειρίζεται απομακρυσμένα τις κινήσεις του βραχίονα ή να αλληλεπιδρά μαζί του μέσω φωνητικών εντολών.

Βιβλιογραφία

- [1] UNITED NATIONS CONFERENCE ON TRADE AND DEVELOPMENT, *Impact of rapid technological change on sustainable development*. Place of publication not identified: UNITED NATIONS, 2020. OCLC: 1145601349.
- [2] E. Garcia, M. Jimenez, P. De Santos, and M. Armada, “The evolution of robotics research,” *IEEE Robotics & Automation Magazine*, vol. 14, pp. 90–103, Mar. 2007.
- [3] J. Andreu-Perez, F. Deligianni, D. Ravi, and G.-Z. Yang, “Artificial Intelligence and Robotics,” *arXiv:1803.10813 [cs]*, Mar. 2018. arXiv: 1803.10813.
- [4] Q. Zhao, D. Tu, S. Xu, H. Shao, and Q. Meng, “Natural human-robot interaction for elderly and disabled healthcare application,” in *2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, (Belfast, United Kingdom), pp. 39–44, IEEE, Nov. 2014.
- [5] Y. Kobayashi, M. Gyoda, T. Tabata, Y. Kuno, K. Yamazaki, M. Shibuya, and Y. Seki, “Assisted-care robot dealing with multiple requests in multi-party settings,” in *Proceedings of the 6th international conference on Human-robot interaction - HRI '11*, (Lausanne, Switzerland), p. 167, ACM Press, 2011.
- [6] M. A. Goodrich and A. C. Schultz, “Human-Robot Interaction: A Survey,” *Foundations and Trends® in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007.
- [7] C. Huffman, “Archytas,” in *The Stanford Encyclopedia of Philosophy* (E. N. Zalta, ed.), Metaphysics Research Lab, Stanford University, summer 2018 ed., 2018.
- [8] J. Riskin, “The Defecating Duck, or, the Ambiguous Origins of Artificial Life,” *Critical Inquiry*, vol. 29, pp. 599–633, June 2003.
- [9] I. ASIMOV, *I, ROBOT*. Place of publication not identified: HARPERCOLLINS, 2018. OCLC: 1002122863.
- [10] B. Kuipers, E. A. Feigenbaum, P. E. Hart, and N. J. Nilsson, “Shakey: From Conception to History,” *AI Magazine*, vol. 38, p. 88, Mar. 2017.
- [11] Tess, “This brick-laying 3d printer robot can build a house four times faster than a human bricklayer,” <https://www.3ders.org/articles/20160728-this-brick-laying-3d-printer-robot-can-build-a-house-four-times-faster-than-a-human-bricklayer.html>. Sept. 2020.
- [12] J. Hawksworth and R. Berriman, “Will robots really steal our jobs? An international analysis of the potential longterm impact of automation.” Sept. 2020.

-
- [13] D. J. Kaivo-oja, “A review on the future of work: Robotics,” tech. rep., EU-OSHA.
- [14] Δρακάκη, “Τι ακριβώς είναι η Ρομποτική,” <http://www.athinodromio.gr>. Sept 2020.
- [15] H. H. Poole, “Types of Robots,” in *Fundamentals of Robotics Engineering* (H. H. Poole, ed.), pp. 27–51, Dordrecht: Springer Netherlands, 1989.
- [16] E. Oxford, “Report - how robots change the world,” <https://cdn2.hubspot.net>. Feb 2021.
- [17] M. Szollosy, “Why Are We Afraid of Robots? The Role of Projection in the Popular Conception of Robots,” in *Beyond Artificial Intelligence* (J. Romportl, E. Zackova, and J. Kelemen, eds.), vol. 9, pp. 121–131, Cham: Springer International Publishing, 2015. Series Title: Topics in Intelligent Engineering and Informatics.
- [18] Y. Liang and S. A. Lee, “Fear of Autonomous Robots and Artificial Intelligence: Evidence from National Representative Data with Probability Sampling,” *International Journal of Social Robotics*, vol. 9, pp. 379–384, June 2017.
- [19] R. Ojstersek and M. Veber, “Intrinsic Motivation of Students Learning Robotics in Mechatronics Education,” <https://www.researchgate.net>.
- [20] N. Nandan and K. Thippeswamy, “A Tensorflow Based Robotic Arm,” in *2018 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, pp. 326–330, Dec. 2018.
- [21] T. B. Pulikottil, M. Caimmi, M. G. Dangelo, E. Biffi, S. Pellegrinelli, and L. M. Tosatti, “A Voice Control System for Assistive Robotic Arms: Preliminary Usability Tests on Patients,” in *2018 7th IEEE International Conference on Biomedical Robotics and Biomechanics (Biorob)*, (Enschede), pp. 167–172, IEEE, Aug. 2018.
- [22] Z. Wang, D. Chen, and P. Xiao, “Design of a Voice Control 6DoF Grasping Robotic arm Based on Ultrasonic Sensor, Computer Vision and Alexa Voice Assistance,” in *2019 10th International Conference on Information Technology in Medicine and Education (ITME)*, (Qingdao, China), pp. 649–654, IEEE, Aug. 2019.
- [23] J. Tasevski, M. Nikolic, and D. Miskovic, “Integration of an industrial robot with the systems for image and voice recognition,” *Serbian Journal of Electrical Engineering*, vol. 10, no. 1, pp. 219–230, 2013.
- [24] H. Bandara, M. Edirisighe, B. Balasooriya, and A. Jayasekara, “Development of an interactive service robot arm for object manipulation,” in *2017 IEEE International Conference on Industrial and Information Systems (ICIIS)*, pp. 1–6, Dec. 2017.
- [25] “Welcome to python.org,” <https://www.python.org/>. Sept. 2020.
- [26] “Pycharm: the python ide for professional developers by jetbrains,” <https://www.jetbrains.com/pycharm/>. Nov. 2020.
- [27] “Notepad++,” <https://notepad-plus-plus.org/>. Nov. 2020.

-
- [28] “Niryo One – An accessible educational 6 axis robotic arm, just for you – Niryo,” <https://niryo.com/niryo-one/>. Nov. 2020.
- [29] “OpenCV,” <https://opencv.org/>. Nov.2020.
- [30] “ZBar bar code reader,” <http://zbar.sourceforge.net/>. Dec. 2020.
- [31] J. Benitez, J. Castro, and I. Requena, “Are artificial neural networks black boxes?,” *IEEE Transactions on Neural Networks*, vol. 8, pp. 1156–1164, Sept. 1997.
- [32] A. L. Samuel, “Some Studies in Machine Learning Using the Game of Checkers,” *IBM Journal of Research and Development*, vol. 3, pp. 210–229, July 1959.
- [33] S. J. Russell, P. Norvig, and E. Davis, *Artificial intelligence: a modern approach*. Prentice Hall series in artificial intelligence, Upper Saddle River: Prentice Hall, 3rd ed ed., 2010.
- [34] F. Rosenblatt, “The Perceptron, a Perceiving and Recognizing Automaton,” *01/1957*. Αδημοσίευτη ερευνητική εργασία.
- [35] . F. Rosenblatt, “The Perceptron: a Probabilistic Model for Information Storage and Organization In the Brain,” 1958. Αδημοσίευτη ερευνητική εργασία.
- [36] J. Han and C. Moraga, “The influence of the sigmoid function parameters on the speed of backpropagation learning,” in *From Natural to Artificial Neural Computation* (G. Goos, J. Hartmanis, J. Leeuwen, J. Mira, and F. Sandoval, eds.), vol. 930, pp. 195–201, Berlin, Heidelberg: Springer Berlin Heidelberg, 1995. Series Title: Lecture Notes in Computer Science.
- [37] C. C. Aggarwal, “Restricted Boltzmann Machines,” in *Neural Networks and Deep Learning*, pp. 235–270, Cham: Springer International Publishing, 2018.
- [38] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (Las Vegas, NV, USA), pp. 779–788, IEEE, June 2016.
- [39] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, pp. 1137–1149, June 2017.
- [40] I. Wallach, M. Dzamba, and A. Heifets, “AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery,” *arXiv:1510.02855 [cs, q-bio, stat]*, Oct. 2015. arXiv: 1510.02855.
- [41] K. Desai, S. Parikh, K. Patel, P. Bide, and S. Ghane, “Survey of Object Detection Algorithms and Techniques,” in *Cybernetics, Cognition and Machine Learning Applications* (V. K. Gunjan, P. N. Suganthan, J. Haase, A. Kumar, and B. Raman, eds.), Algorithms for Intelligent Systems, (Singapore), pp. 247–257, Springer, 2020.
- [42] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, “Object Detection With Deep Learning: A Review,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 3212–3232, Nov. 2019.

-
- [43] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” *arXiv:1311.2524 [cs]*, Oct. 2014. arXiv: 1311.2524.
- [44] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” p. 6.
- [45] R. L. Klevans and R. D. Rodman, *Voice Recognition*. USA: Artech House, Inc., 1st ed., 1997.
- [46] B.-H. Juang, S. E. Levinson, L. R. Rabiner, and M. M. Sondhi, “Hidden Markov model speech recognition arrangement,” Nov. 1988.
- [47] “Speech-to-Text: Automatic Speech Recognition,” <https://cloud.google.com/speech-to-text?hl=el>. Dec. 2020.
- [48] P. Taylor, *Text-to-Speech Synthesis*. Cambridge University Press, Feb. 2009.
- [49] T. Dutoit, *An Introduction to Text-to-Speech Synthesis*. Springer Science & Business Media, Apr. 1997. Google-Books-ID: xjXU3FujWbwC.
- [50] “Using pytt3 — pytt3 2.6 documentation,” <https://pytt3.readthedocs.io/en/latest/engine.html>. Dec. 2020.
- [51] D. Ardayfio, *Fundamentals of Robotics*. CRC Press, July 2020. Google-Books-ID: k1LzDwAAQBAJ.
- [52] J. J. Craig, *Introduction to robotics: mechanics and control*. Upper Saddle River, N.J: Pearson/Prentice Hall, 3rd ed ed., 2005.
- [53] H. Lipkin, “A Note on Denavit-Hartenberg Notation in Robotics,” pp. 921–926, American Society of Mechanical Engineers Digital Collection, June 2008.
- [54] U. Hartmann, *Aktuelle Methoden der Laser- und Medizinphysik: Tagungsband der 2. Remagener Physiktage 2004, RheinAhrCampus Remagen, 29. September bis 1. Oktober 2004*. Margret Schneider, 2005.
- [55] “Niryo one – an accessible educational 6 axis robotic arm, just for you – niryo,” <https://niryo.com/niryo-one/>. Nov.2020.