



ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΗΛΕΚΤΡΟΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Διπλωματική Εργασία

**Εκτίμηση κατανάλωσης ενέργειας
ηλεκτροκίνητων οχημάτων μέσω εναέριας
παρατήρησης της κυκλοφορίας**

Κωνσταντίνος Λεοντσίνης

Εξεταστική Επιτροπή:

Αναπληρωτής Καθηγητής Κανέλλος Φώτιος (Επιβλέπων)

Καθηγητής Σταυρακάκης Γεώργιος

Αναπληρωτής Καθηγητής Παρτσινέβελος Παναγιώτης

Χανιά 2022

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω θερμά τον κύριο Παναγιώτη Παρτσινέβελο για την βοήθεια του στον τομέα της υπολογιστικής όρασης και της επεξεργασίας εικόνας, καθώς οι γνώσεις μου σε αυτά τα θέματα ήταν αρκετά περιορισμένες. Επίσης, θα ήθελα να ευχαριστήσω τον κύριο Φώτιο Κανέλλο για την καθοδήγηση και τις συμβουλές που μου παρείχε σε όλη τη διάρκεια εκπόνησης της διπλωματικής εργασίας. Επιπλέον, θα ήθελα να ευχαριστήσω την συμφοιτήτρια Χαρά Τζάλα για την υποστήριξή της κατά την διάρκεια εκπόνησης της διπλωματικής εργασίας, καθώς και τον συμφοιτητή Παντελή Νικολακάκη για τα φοιτητικά χρόνια που περάσαμε στα Χανιά. Ολοκληρώνοντας, θα ήθελα να ευχαριστήσω τους φίλους και την οικογένειά μου για την υποστήριξη που μου έδειχναν καθ' όλη την διάρκεια των ακαδημαϊκών σπουδών μου.

Περίληψη

Η έξαρση του φαινομένου του θερμοκηπίου και κατ' επέκταση η κλιματική αλλαγή, που λαμβάνει χώρα τα τελευταία χρόνια, οφείλεται στους αέριους ρύπους που εκλύονται στην ατμόσφαιρα από την καύση ορυκτών πόρων όπως είναι το πετρέλαιο και τα παράγωγά του. Η χρήση των παραπάνω πόρων γίνεται κατά κύριο λόγο από τα αυτοκίνητα λόγω των μηχανών εσωτερικής καύσης που δέχονται κυρίως ως καύσιμο τη βενζίνη και το diesel. Η επιτακτική ανάγκη αλλαγής των κινητήρων χωρίς να παράγονται αέριοι ρύποι ώθησε τις αυτοκινητοβιομηχανίες στην παραγωγή αμιγώς ηλεκτρικών οχημάτων, τα οποία λειτουργούν με ηλεκτρικούς κινητήρες και τροφοδοτούνται από μπαταρίες.

Το πιο σημαντικό μειονέκτημα των ηλεκτρικών αυτοκινήτων έγκειται στην αυξημένη κατανάλωση ισχύος της μπαταρίας, που διαθέτουν, όταν κινούνται σε μεγάλες ταχύτητες. Επομένως, για την ομαλή αλλαγή του στόλου αυτοκινήτων της χώρας μας είναι ανάγκη η εκτίμηση των ταχυτήτων που αναπτύσσονται από τα αυτοκίνητα κατά μήκος των αυτοκινητόδρομων και των οδών ταχείας κυκλοφορίας, με σκοπό την σωστή τοποθέτηση σταθμών φόρτισης κατά μήκος αυτών.

Στην παρούσα διπλωματική εργασία αναλύεται μία μέθοδος καταγραφής των ταχυτήτων που αναπτύσσονται κατά μήκος του Βόρειου Οδικού Άξονα της Κρήτης (BOAK) από τα αυτοκίνητα. Αρχικά, η καταγραφή των βίντεο πραγματοποιήθηκε σε τρία σημεία κατά μήκος του Βόρειου Οδικού Άξονα της Κρήτης περιφερειακά των Χανίων και σε απόσταση 7 χιλιομέτρων το ένα από το άλλο. Η λήψη των πλάνων υλοποιήθηκε με την βοήθεια ενός πολιτικού μη επανδρωμένου αεροσκάφους, με στόχο την αποφυγή πιθανών παρεμβολών που εμφανίζονται μπροστά από την κάμερα. Στην συνέχεια, με την αξιοποίηση του αλγορίθμου ανίχνευσης αντικειμένων YOLO (You Only Look Once) και της μεθόδου Mixture of Gaussian 2 (MOG2), που παρέχεται από την βιβλιοθήκη OpenCV, αναγνωρίζονται τα αυτοκίνητα και καταγράφονται οι ταχύτητες που αναπτύσσουν. Η αξιοποίηση του αλγορίθμου YOLO πραγματοποιείται με σκοπό την καταγραφή μόνο των ταχυτήτων των αυτοκινήτων και όχι όλων των οχημάτων, όπως των φορτηγών και των λεωφορείων.

Στη συνέχεια, οι ταχύτητες, που βρέθηκαν, χρησιμοποιούνται για την εκτίμηση της ενεργειακής κατανάλωσης τεσσάρων αμιγώς ηλεκτρικών αυτοκινήτων, που καλύπτουν τις διαδρομές Χανιά-Ρέθυμνο και Χανιά-Ηράκλειο. Ολοκληρώνοντας, με την αξιοποίηση ενός συστήματος ασαφούς λογικής εξάγεται η πιθανότητα ο οδηγός ενός ηλεκτρικού αυτοκινήτου να αναζητάει σταθμό φόρτισης κατά μήκος του Βόρειου Οδικού Άξονα της Κρήτης.

Συνοψίζοντας, στόχος της παρούσας διπλωματικής εργασίας αποτελεί η σύσταση τοποθέτησης σταθμών φόρτισης κατά μήκος του BOAK μεταξύ των Χανίων και του Ηρακλείου, με βάση τα αποτελέσματα που αποδίδει το ασαφές σύστημα, που προαναφέρεται.

Abstract

The enhanced greenhouse effect and climate change, which has taken place in recent years, are due to the gaseous pollutants, which are released into the atmosphere from the burning of mineral resources, such as oil and its derivatives. The usage of the above resources is mainly made by the cars because of the internal combustion engines, which mostly accept gasoline and diesel as fuel. The urgent need of changing the engine's car, without polluting the environment, has motivated the car industries to design and produce electric cars, that uses an electric motor and powers from batteries.

The most important disadvantage of the electric cars is the increased power consumption of the battery when traveling at high speeds. Therefore, for the smooth transition of the car's fleet, it is essential to estimate cars' speeds on highways and expressways, in order to properly place charging stations along them.

In the present diploma thesis, a speed recording method is analyzed by the cars, that are traveling along the Northern Road Axis of Crete. First of all, the video recording of the cars took place at three points along the Northern Road Axis of Crete around the city of Chania and at a distance of 7 kilometers from each other. Moreover, the videos were record with a civilian drone, in order to avoid possible interference in front of the camera. Furthermore, the identification of the cars and estimation of their speed is implemented by the object detection algorithm YOLO (You Only Look Once) and the Mixture of Gaussian 2 (MOG2) method, which is provided by the OpenCV library. YOLO algorithm is used to record only the speeds of cars and not all vehicles, such as trucks and buses.

Subsequently, the speeds are used to estimate the power consumption of four electric cars, that cover the routes Chania-Rethymno and Chania-Hraklion. In conclusion, a fuzzy logic system is used to derive the possibility a driver of an electric car is searching for a charging station along the Northern Road Axis of Crete.

In summary, the aim of this thesis is a recommendation about the placement of the charging stations on the highway from Chania to Hraklion, which is based upon the fuzzy logic system's results.

Περιεχόμενα

Περίληψη.....	3
Abstract	4
Λίστα Εικόνων	7
Κεφάλαιο 1^ο	10
1.1 Εισαγωγή	10
1.2 Στόχος Διπλωματικής Εργασίας.....	11
1.3 Μεθοδολογική Προσέγγιση.....	11
1.4 Εφαρμογές της διπλωματικής εργασίας	11
1.5 Δομή Διπλωματικής Εργασίας.....	12
Κεφάλαιο 2^ο	13
2.1 Εισαγωγή	13
2.2 Τι είναι η μηχανική μάθηση.....	13
2.4 Μέθοδοι Μηχανικής Μάθησης.....	15
2.5 Συνελκτικά Νευρωνικά Δίκτυα [12].....	19
2.5.1 Εισαγωγή	19
2.5.2 Convolution Layer	19
2.5.3 Pooling Layer	20
2.5.4 Fully Connected Layer	21
2.6 Τεχνικές αλγορίθμων ανίχνευσης αντικειμένων σε εικόνα	21
2.6.1 Intersection over Union (IoU) [13]	21
2.6.2 Non-Maximum Suppression (NMS) [14]	22
2.7 Μέθοδος αξιολόγησης αλγορίθμων ανίχνευσης αντικειμένων [15]	23
2.8 Αλγόριθμοι ανίχνευσης αντικειμένων	25
2.8.1 Εισαγωγή	25
2.8.2 Αλγόριθμος R-CNN [17].....	26
2.8.3 Αλγόριθμος Fast R-CNN [18]	27
2.8.4 Αλγόριθμος Faster R-CNN [19]	28
2.8.5 Αλγόριθμος Yolo	29
2.9 Σχετικές έρευνες καταγραφής αυτοκινήτων και των ταχυτήτων τους σε βίντεο.....	34
2.10 Δυναμικές Εξισώσεις Οχημάτων	36
Κεφάλαιο 3^ο	39
3.1 Εισαγωγή	39
3.2 Μη επανδρωμένα αεροσκάφη	39
3.2.1 Ιστορική αναδρομή	39
3.2.2 DJI Phantom 4 Pro 2.0	40
3.3 Εργαλεία	41
3.3.1 PYTHON	41
3.3.2 Βιβλιοθήκη OPENCV.....	42

3.3.3 CUDA	43
3.4 Μεθοδολογία ανίχνευσης ταχύτητας και συνόλου αυτοκινήτων	44
3.5 Προβλήματα	49
3.6 Επίλυση προβλημάτων	50
3.6.1 Προτεινόμενες λύσεις.....	50
3.6.2 Μέθοδοι ανίχνευσης κίνησης	51
3.6.3 Επιλογή κατάλληλης μεθόδου	55
3.7 Αποτελέσματα ταχυτήτων.....	58
Κεφάλαιο 4^ο	60
4.1 Εισαγωγή	60
4.2 OCTAVE	60
4.3 Μεθοδολογία ενεργειακής αξιολόγησης	62
4.3.1 Ηλεκτρικά αυτοκίνητα	62
4.3.2 Ωριαίος κυκλοφοριακός φόρτος	62
4.4.3 Αξιοποίηση εργαλείου Octave	69
Κεφάλαιο 5^ο	79
5.1 Αποτελέσματα διαδρομής Χανιά-Ρέθυμνο	79
5.2 Αποτελέσματα διαδρομής Χανιά-Ηράκλειο	84
5.3 Αποτελέσματα διαδρομής Ρέθυμνο-Ηράκλειο	89
Κεφάλαιο 6^ο	94
6.1 Μελλοντικές Βελτιώσεις	94
Βιβλιογραφία.....	95

Λίστα Εικόνων

Εικόνα 1:Είδη μηχανικής μάθησης	14
Εικόνα 2:Γραμμική παλινδρόμηση (linear regression)	16
Εικόνα 3: Παράδειγμα ομαδοποίησης δεδομένων	17
Εικόνα 4: Παράδειγμα απλού νευρωνικού δικτύου	18
Εικόνα 5: Χρήση συνελκτικού στρώματος	20
Εικόνα 6: Εφαρμογή τεχνικής Padding.....	20
Εικόνα 7: Παράδειγμα max Pooling.....	21
Εικόνα 8: Η μέθοδος Intersection over Union.....	22
Εικόνα 9: Παράδειγμα μεθόδου Intersection over Union	22
Εικόνα 10: Παράδειγμα μεθόδου Non-Maximum Suppression	23
Εικόνα 11: Παράδειγμα γραφήματος precision-recall	24
Εικόνα 12: Δημοσιεύσεις αλγορίθμων ανίχνευσης αντικειμένων.....	25
Εικόνα 13: Παράδειγμα αλγορίθμου R-CNN.....	27
Εικόνα 14: Παράδειγμα αλγορίθμου Fast R-CNN.....	28
Εικόνα 15: Παράδειγμα αλγορίθμου Faster R-CNN.....	29
Εικόνα 16:Αρχιτεκτονική δικτύου YOLO	30
Εικόνα 17: Πίνακας εξόδου του εκάστοτε κελιού.....	31
Εικόνα 18: Εφαρμογή αλγορίθμου YOLO.....	31
Εικόνα 19: Χρήση anchor boxes	32
Εικόνα 20: Στρατιωτικό μη επανδρωμένο αεροσκάφος.....	40
Εικόνα 21: DJI Phantom 4 Pro 2.0	41
Εικόνα 22: Το λογότυπο της Python	42
Εικόνα 23: Το λογότυπο της βιβλιοθήκης OpenCV.....	43
Εικόνα 24: Κλάσεις του αρχείου coco.names	44
Εικόνα 25: Παραδείγματα του αρχείου yolov4.cfg.....	45
Εικόνα 26: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 320x320.....	46
Εικόνα 27: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 416x416.....	46
Εικόνα 28: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 512x512.....	46
Εικόνα 29: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 608x608.....	47
Εικόνα 30: Μέτρηση της απόστασης με την βοήθεια του Google Maps	48
Εικόνα 31: Η απόσταση (μεταξύ των δύο γαλάζιων γραμμών) όπως φαίνεται από το βίντεο του drone	49
Εικόνα 32: Παράδειγμα εφαρμογής της παραπάνω τεχνικής.....	51
Εικόνα 33: Εικόνα που εισάγεται στον αλγόριθμο YOLO	51
Εικόνα 34: Η μέθοδος Background Subtraction.....	52
Εικόνα 35: Αρχική εικόνα	52
Εικόνα 36: Αποτελέσματα μεθόδου MOG.....	53
Εικόνα 37: Αποτέλεσμα μεθόδου GMG	54
Εικόνα 38: Αποτέλεσμα μεθόδου MOG2	55
Εικόνα 39: Αρχική εικόνα	56
Εικόνα 40: Αποτέλεσμα μεθόδου MOG2 χωρίς τις τεχνικές erode και dilation.....	56
Εικόνα 41: Αποτέλεσμα πλαισίου οριοθέτησης χωρίς την χρήση των τεχνικών erode και dilation.....	56
Εικόνα 42: Αποτέλεσμα μεθόδου MOG2 με τις τεχνικές erode και dilation	57
Εικόνα 43: Αποτέλεσμα πλαισίου οριοθέτησης με τις τεχνικές erode και dilation	57
Εικόνα 44: Τα τρία σημεία που καταγράφηκαν τα βίντεο	58

Εικόνα 45: Διάγραμμα του αριθμού των αυτοκινήτων με την ίδια ταχύτητα	59
Εικόνα 46: Το λογότυπο του εργαλείου Octave.....	61
Εικόνα 47: Σταθμοί καταγραφής κινούμενων οχημάτων.....	63
Εικόνα 48: Μετρήσεις οχημάτων στον BOAK για την θερινή περίοδο.....	63
Εικόνα 49: Μετρήσεις οχημάτων στον BOAK για την χειμερινή περίοδο	64
Εικόνα 50: Μέση ημερήσια κυκλοφορία και στις 2 κατευθύνσεις ανά σταθμό και μήνα	64
Εικόνα 51: Η μεταβλητή A αντιπροσωπεύει τον αριθμό των οχημάτων που πραγματοποιούν τη διαδρομή Χανιά-Ρέθυμνο	66
Εικόνα 52: Η μεταβλητή R αντιπροσωπεύει τον αριθμό των οχημάτων που πραγματοποιούν τη διαδρομή Ρέθυμνο-Ηράκλειο	66
Εικόνα 53: Το αποτέλεσμα της αφαίρεσης RS2-A αντιπροσωπεύει τον αριθμό των οχημάτων που πραγματοποιούν τη διαδρομή Χανιά-Ηράκλειο.....	67
Εικόνα 54: Κατανομή αρχικού ποσοστού φόρτισης της μπαταρίας για τα αυτοκίνητα της διαδρομής Χανιά-Ρέθυμνο	70
Εικόνα 55: Κατανομή αρχικού ποσοστού φόρτισης της μπαταρίας για τα αυτοκίνητα της διαδρομής Χανιά-Ηράκλειο	71
Εικόνα 56: Κατανομή αρχικού ποσοστού φόρτισης της μπαταρίας για τα αυτοκίνητα της διαδρομής Ρέθυμνο-Ηράκλειο.....	71
Εικόνα 57: Κατανομή ποσοστού άγχους του οδηγού για το ποσοστό φόρτισης της μπαταρίας του αυτοκινήτου	72
Εικόνα 58: Ποσοστό άγχους ως είσοδο στο σύστημα fuzzy	74
Εικόνα 59: Ποσοστό φόρτισης της μπαταρίας ως είσοδο στο σύστημα fuzzy	75
Εικόνα 60: Έξοδος συστήματος fuzzy	76
Εικόνα 61: Η έξοδος του συστήματος fuzzy συναρτήσει των δύο εισόδων	77
Εικόνα 62: Αποτελέσματα διαδρομής Χανιά-Ρέθυμνο	79
Εικόνα 63: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Audi, διαδρομή Χανιά-Ρέθυμνο).....	80
Εικόνα 64: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Audi, διαδρομή Χανιά-Ρέθυμνο).....	80
Εικόνα 65: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Fiat, διαδρομή Χανιά-Ρέθυμνο)	81
Εικόνα 66: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Fiat, διαδρομή Χανιά-Ρέθυμνο)	81
Εικόνα 67: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Nissan, διαδρομή Χανιά-Ρέθυμνο).....	82
Εικόνα 68: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Nissan, διαδρομή Χανιά-Ρέθυμνο).....	82
Εικόνα 69: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Mazda, διαδρομή Χανιά-Ρέθυμνο).....	83
Εικόνα 70: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Mazda, διαδρομή Χανιά-Ρέθυμνο).....	83
Εικόνα 71: Αποτελέσματα διαδρομής Χανιά-Ηράκλειο	84
Εικόνα 72: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Audi, διαδρομή Χανιά-Ηράκλειο).....	85
Εικόνα 73: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Audi, διαδρομή Χανιά-Ηράκλειο).....	85
Εικόνα 74: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Fiat, διαδρομή Χανιά-Ηράκλειο)	86
Εικόνα 75: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Fiat, διαδρομή Χανιά-Ηράκλειο)	86
Εικόνα 76: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης	

σταθμού φόρτισης(Nissan, διαδρομή Χανιά-Ηράκλειο).....	87
Εικόνα 77: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Nissan, διαδρομή Χανιά-Ηράκλειο).....	87
Εικόνα 78: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Mazda, διαδρομή Χανιά-Ηράκλειο).....	88
Εικόνα 79: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Mazda, διαδρομή Χανιά-Ηράκλειο).....	88
Εικόνα 80: Αποτελέσματα διαδρομής Ρέθυμνο-Ηράκλειο	89
Εικόνα 81: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Audi, διαδρομή Ρέθυμνο-Ηράκλειο).....	90
Εικόνα 82: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Audi, διαδρομή Ρέθυμνο-Ηράκλειο).....	90
Εικόνα 83: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Fiat, διαδρομή Ρέθυμνο-Ηράκλειο).....	91
Εικόνα 84: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Fiat, διαδρομή Ρέθυμνο-Ηράκλειο).....	91
Εικόνα 85: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Nissan, διαδρομή Ρέθυμνο-Ηράκλειο).....	92
Εικόνα 86: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Nissan, διαδρομή Ρέθυμνο-Ηράκλειο).....	92
Εικόνα 87: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Mazda, διαδρομή Ρέθυμνο-Ηράκλειο).....	93
Εικόνα 88: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Mazda, διαδρομή Ρέθυμνο-Ηράκλειο).....	93

Κεφάλαιο 1^ο

1.1 Εισαγωγή

Η ιδέα της ηλεκτροκίνησης δεν αποτελεί καινούρια εφεύρεση των αυτοκινητοβιομηχανιών. Κατά την πρώτη, μάλιστα, περίοδο της αυτοκινητοβιομηχανίας έως περίπου το 1920 τα ηλεκτρικά οχήματα ανταγωνίζονταν σε μεγάλο βαθμό τα οχήματα που χρησιμοποιούσαν μηχανές εσωτερικής καύσης ως πολυτελή οχήματα που προορίζονταν για αστική χρήση. Επιπλέον, ήταν ιδιαίτερα δημοφιλή για την ησυχία και το πολύ χαμηλό κόστος συντήρησης που είχαν. Παρόλα αυτά, τρεις κύριοι λόγοι οδήγησαν τις αυτοκινητοβιομηχανίες να στραφούν στην παραγωγή οχημάτων που χρησιμοποιούν ως καύσιμο παράγωγα του πετρελαίου.

Κατ' αρχάς, η αυτονομία που είχαν τα ηλεκτρικά οχήματα ήταν πολύ μικρή, καθώς οι μπαταρίες που είχαν σχεδιαστεί εκείνη την εποχή δεν ήταν ικανές να επιτρέψουν την κάλυψη μεγάλων αποστάσεων, αφού συνήθως η αυτονομία τους έφτανε κατά μέσο όρο τα 50 χιλιόμετρα. Επίσης, η μαζική παραγωγή μη ηλεκτρικών οχημάτων από τον Henry Ford, που είχε ως αποτέλεσμα την δραματική μείωση στα κόστη συντήρησής τους, αλλά και η ανακάλυψη μεγάλων κοιτασμάτων πετρελαίου στις Ηνωμένες Πολιτείες της Αμερικής ήταν από τους λόγους που οδήγησε τις αυτοκινητοβιομηχανίες να στραφούν αποκλειστικά στην παραγωγή οχημάτων με μηχανή εσωτερικής καύσης. [1]

Ωστόσο, με το πέρασμα των ετών η αύξηση της κρίσης της κλιματικής αλλαγής, η μείωση των κοιτασμάτων πετρελαίου και η ανάπτυξη της τεχνολογίας των μπαταριών ώθησε τις αυτοκινητοβιομηχανίες στη σχεδίαση και κατασκευή υβριδικών και αμιγώς ηλεκτρικών οχημάτων.

Ένα υβριδικό όχημα αποτελείται από έναν ηλεκτρικό κινητήρα, ο οποίος τροφοδοτείται από μία συστοιχία μπαταριών, και ένα κινητήρα εσωτερικής καύσης, ο οποίος τροφοδοτείται με βενζίνη αντίστοιχα. Ο ηλεκτρικός κινητήρας χρησιμοποιείται για μικρές ταχύτητες, ενώ ο κινητήρας εσωτερικής καύσης ενεργοποιείται σε μεγαλύτερες ταχύτητες που αναπτύσσει το όχημα. Επιπλέον, ο κινητήρας εσωτερικής καύσης ενεργοποιείται σε περίπτωση που οι μπαταρίες έχουν χάσει μεγάλο ποσοστό ισχύος με σκοπό την επαναφόρτισή τους. Με αυτόν τον τρόπο, επιτυγχάνεται πολύ μεγάλη αυτονομία σε συνδυασμό με τις πολύ μικρές εκπομπές ρύπων στην ατμόσφαιρα.

Από την άλλη πλευρά, τα αμιγώς ηλεκτρικά οχήματα χρησιμοποιούν μόνο έναν ηλεκτρικό κινητήρα, ο οποίος τροφοδοτείται από μία πολύ μεγαλύτερη συστοιχία μπαταριών σε σχέση με τα υβριδικά οχήματα. Παρόλα αυτά, το πρόβλημα του παρελθόντος με την αυτονομία που πετυχαίνουν οι μπαταρίες δεν έχει επιλυθεί. Τα περισσότερα ηλεκτρικά οχήματα έχουν αυτονομία περίπου στα 400 χιλιόμετρα, σε αντίθεση με τα συμβατικά οχήματα βενζίνης που έχουν περίπου στα 600 χιλιόμετρα και τα οχήματα πετρελαίου που έχουν ακόμη περισσότερη.

Επιπλέον, η αυτονομία των οχημάτων επηρεάζεται σε μεγάλο βαθμό από τον τρόπο οδήγησης από τον εκάστοτε οδηγό και από τη περιοχή που κινείται το όχημα. Για παράδειγμα, τα οχήματα που χρησιμοποιούν καύσιμα βενζίνης και πετρελαίου καταναλώνουν περισσότερο καύσιμο όταν κινούνται εντός της πόλης, λόγω της συχνής στάσης και επιτάχυνσης που πραγματοποιούν. Από την άλλη πλευρά, η κατανάλωση καυσίμου στις εθνικές οδούς και στους αυτοκινητόδρομους είναι λιγότερη, εφόσον ο κινητήρας λειτουργεί σε σταθερό ρυθμό χωρίς απότομες

επιταχύνσεις. Αντίθετα, η κατανάλωση ισχύος των ηλεκτρικών οχημάτων είναι λιγότερη στους δρόμους της πόλης εξαιτίας της απενεργοποίησης του κινητήρα σε περιπτώσεις στάσεων, ενώ είναι πολύ μεγαλύτερη σε εθνικές οδούς και αυτοκινητόδρομους λόγω της συνεχής χρήσης του κινητήρα.

1.2 Στόχος Διπλωματικής Εργασίας

Τα ηλεκτρικά αυτοκίνητα αναμένεται να κατακλίσουν τους δρόμους τα επόμενα χρόνια εξαιτίας της συμβολής τους στην προστασία του περιβάλλοντος, στην ευκολία της χρήσης τους και στην φθηνή ανατροφοδότησή τους σε σχέση με τα συμβατικά αυτοκίνητα. Ωστόσο, ένα πρόβλημα που αντιμετωπίζουν τα ηλεκτρικά αυτοκίνητα στην χώρα μας είναι η επαναφόρτιση των μπαταριών τους. Στις μεγαλουπόλεις το πρόβλημα αυτό αρχίζει να λύνεται, με την κατασκευή και τοποθέτηση όλο και περισσότερων σταθμών φόρτισης. Παρόλα αυτά, το πρόβλημα εντείνεται στις εθνικές οδούς όπου η τοποθέτηση των σταθμών φόρτισης πρέπει να γίνει με μεγάλη ακρίβεια αφού η αυτονομία των ηλεκτρικών αυτοκινήτων με την σημερινή τεχνολογία δεν επιτρέπει την τυχαία τοποθέτηση τους.

Στόχος, λοιπόν, της παρούσας διπλωματικής εργασίας είναι η εκτίμηση της κατανάλωσης ενέργειας ηλεκτρικών αυτοκινήτων σε αυτοκινητόδρομους και εθνικές οδούς και πιο συγκεκριμένα για τον Βόρειο Οδικό Άξονα της Κρήτης (BOAK), ώστε να κατασκευαστούν σταθμοί φόρτισης σε καίρια σημεία κατά μήκος του.

1.3 Μεθοδολογική Προσέγγιση

Η εκτίμηση της κατανάλωσης ενέργειας πραγματοποιήθηκε με βάση τις πραγματικές ταχύτητες που αναπτύσσονται από τα αυτοκίνητα που κινούνται στον Βόρειο Οδικό Άξονα της Κρήτης περιφερειακά των Χανίων. Εφαρμόζοντας τις μαθηματικές σχέσεις, οι οποίες αναλύονται σε επόμενο κεφάλαιο, σε συνδυασμό με τις παραπάνω ταχύτητες υπολογίζεται η ισχύς που καταναλώνεται από κάθε αυτοκίνητο. Αποτέλεσμα των παραπάνω είναι η πραγματοποίηση υποθέσεων για κάποια μοντέλα ηλεκτρικών αυτοκινήτων που υπάρχουν ήδη στην αγορά και δύναται να βρεθεί η απόσταση που χρειάζεται να έχει ο ένας σταθμός φόρτισης από τον άλλο κατά μήκος του BOAK.

Η εύρεση της ταχύτητας των αυτοκινήτων δεν πραγματοποιείται σε πραγματικό χρόνο, αλλά ετεροχρονισμένα μέσω βίντεο. Η λήψη των βίντεο γίνεται με την βοήθεια ενός μη επανδρωμένου αεροσκάφους σε τρία σημεία του BOAK περιφερειακά των Χανίων και σε απόσταση 7 χιλιομέτρων το ένα από το άλλο.

Οι ταχύτητες των αυτοκινήτων υπολογίζονται μέσα από την διαδικασία επεξεργασίας του εκάστοτε βίντεο. Τέλος, χρησιμοποιείται ένας αλγόριθμος ανίχνευσης αντικειμένων, ο οποίος συμβάλει στην αναγνώριση μόνο των αυτοκινήτων και όχι άλλων οχημάτων, όπως των μηχανών και των φορτηγών, για την καλύτερη προσέγγιση των αποτελεσμάτων.

1.4 Εφαρμογές της διπλωματικής εργασίας

Για την ομαλή αλλαγή του στόλου των αυτοκινήτων στην χώρα μας σε ηλεκτρικά αυτοκίνητα απαιτείται η τοποθέτηση σταθμών φόρτισης κατά μήκος των εθνικών

οδών από τις εταιρίες ηλεκτρισμού αλλά και εταιρίες που προμηθεύουν καύσιμα στα συμβατικά αυτοκίνητα και επιθυμούν να επενδύσουν σε ένα τέτοιο μεγάλο έργο. Επιπλέον, δεν μπορεί να αποτελέσει ζημιογόνα επένδυση γιατί όλες οι αυτοκινητοβιομηχανίες προσπαθούν να παράγουν όλο και περισσότερα ηλεκτρικά αυτοκίνητα και να περιορίσουν την παραγωγή των συμβατικών.

Η μεθοδολογία που αναπτύσσεται στην παρούσα διπλωματική εργασία μπορεί να χρησιμοποιηθεί από τις εταιρίες που αναφέρονται στην προηγούμενη παράγραφο για την μελέτη και σωστή τοποθέτηση σταθμών φόρτισης κατά μήκος του Βόρειου Οδικού Άξονα της Κρήτης. Επιπρόσθετα, δίνεται η ευκαιρία στις εταιρίες υγρών καυσίμων να επεκτείνουν τις δραστηριότητες τους στον τομέα παροχής ρεύματος στα ηλεκτρικά αυτοκίνητα. Η μεθοδολογία της μελέτης τοποθέτησης σταθμών φόρτισης κατά μήκος των εθνικών δρόμων πέρα του ΒΟΑΚ δύναται να γενικευτεί για την γρηγορότερη ένταξη της χώρας μας στην πρακτική χρήση της τεχνολογίας της ηλεκτροκίνησης.

1.5 Δομή Διπλωματικής Εργασίας

Στο δεύτερο κεφάλαιο αναλύεται η θεωρία της μηχανικής μάθησης και των διαφόρων μεθόδων που χρησιμοποιούνται. Επίσης, αναφέρονται διάφοροι αλγόριθμοι ανίχνευσης αντικειμένων, καθώς και οι δυναμικές εξισώσεις των ηλεκτρικών αυτοκινήτων που έχουν ως αποτέλεσμα την κατανάλωση της ισχύος της μπαταρίας δεδομένου μίας ταχύτητας.

Από την άλλη πλευρά, στο τρίτο κεφάλαιο αναφέρεται το είδος του drone που χρησιμοποιήθηκε για την καταγραφή των βίντεο, ενώ επίσης αναφέρεται η γλώσσα προγραμματισμού και η βιβλιοθήκη, που χρησιμοποιήθηκε. Επιπλέον, αναλύονται ο αλγόριθμος που επιλέχθηκε για την αναγνώριση των αυτοκινήτων και διάφορα μέσα που αξιοποιούνται για την καταγραφή της ταχύτητας των αυτοκινήτων.

Στο τέταρτο κεφάλαιο παρουσιάζεται το εργαλείο και η μεθοδολογία που αξιοποιήθηκε για την ενεργειακή αξιολόγηση των ηλεκτρικών αυτοκινήτων. Τέλος, εξηγείται το σύστημα ασαφούς λογικής που χρησιμοποιείται για την ένδειξη του χιλιομέτρου αναζήτησης ενός σταθμού φόρτισης από έναν οδηγό.

Ολοκληρώνοντας, στο πέμπτο κεφάλαιο αναφέρονται μελλοντικές διορθώσεις που μπορούν να πραγματοποιηθούν και να βελτιώσουν την μεθοδολογία της παρούσας διπλωματικής εργασίας.

Κεφάλαιο 2^ο

2.1 Εισαγωγή

Σε αυτό το κεφάλαιο αναλύεται το θεωρητικό υπόβαθρο, που χρησιμοποιείται στην παρούσα διπλωματική εργασία για την βέλτιστη επίτευξη των αποτελεσμάτων. Αρχικά, εξηγείται η θεωρία της μηχανικής μάθησης και των διαφόρων μεθόδων της, που εφαρμόζονται στην υπολογιστική όραση, για την καλύτερη ανίχνευση των αυτοκινήτων στον BOAK. Στο τέλος του κεφαλαίου αναφέρονται οι δυναμικές εξισώσεις των ηλεκτρικών οχημάτων, που αναπτύχθηκαν από τους Mehrdad Ehsani, Yimin Gao, Ali Emadi, και από τους οποίους υπολογίζεται η ισχύς που καταναλώνει ένα ηλεκτρικό αυτοκίνητο με μία μη μεταβαλλόμενη ταχύτητα.

2.2 Τι είναι η μηχανική μάθηση

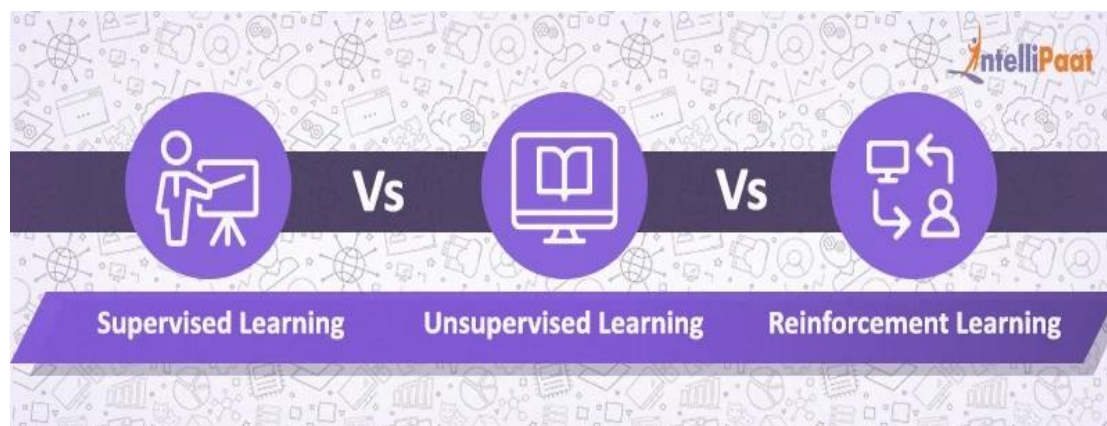
Σύμφωνα με τον Tom Mitchell ένα πρόγραμμα υπολογιστή λέμε ότι μαθαίνει από την εμπειρία E αναφορικά με μια εργασία T και ένα μέτρο απόδοσης P , εάν η απόδοση του T , όπως μετράτε από το P , βελτιώνεται με την εμπειρία E . Ουσιαστικά, μηχανική μάθηση αποτελεί η μελέτη και κατασκευή αλγορίθμων υπολογιστών ικανών να δέχονται δεδομένα, να αναγνωρίζουν μοτίβα σχετικά με αυτά και να βελτιώνονται αυτόματα από τα λάθη τους. Έτσι, δεν χρειάζεται να προγραμματίζεται ο υπολογιστής βήμα-βήμα όπως συνέβαινε τόσα χρόνια, αλλά να προγραμματίζεται με τέτοιο τρόπο ώστε να μαθαίνει από τη δοκιμή, το λάθος και την εμπειρία όπως ο άνθρωπος. [2]

Επιπλέον, η μηχανική μάθηση θεωρείται ένας κλάδος της τεχνητής νοημοσύνης (Artificial Intelligence-AI). Μάλιστα, οι αλγόριθμοι μηχανικής μάθησης χρησιμοποιούνται ευρέως στην υπολογιστική όραση (Computer Vision), αφού τα δεδομένα που δύναται να επεξεργαστούν είναι εικόνες και βίντεο όλων των ειδών. Με την εισαγωγή, λοιπόν, των δεδομένων αυτών ο αλγόριθμος αρχίζει να καταλαβαίνει μοτίβα και να κάνει προβλέψεις με βάση αυτά τα μοτίβα, όπως να καταλαβαίνει αν σε μία εικόνα βρίσκεται αυτοκίνητο ή ποδήλατο.

Χαρακτηριστικό παράδειγμα μηχανικής μάθησης που συναντάτε είναι η αναγνώριση των ανεπιθύμητων e-mail (spam e-mail) από διάφορες εταιρίες όπως η Google και η Yahoo. Ο συγκεκριμένος αλγόριθμος έχει εκπαιδευτεί κατάλληλα, ώστε να αναγνωρίζει εάν πρόκειται για ένα spam e-mail από τυχόν εκφράσεις που περιλαμβάνονται σε αυτά. Επίσης, είναι άξιο αναφοράς πως εάν στο μέλλον δεν χρησιμοποιούνται οι συγκεκριμένες εκφράσεις ο αλγόριθμος μπορεί να προσαρμοστεί κατάλληλα και να τα αναγνωρίσει χωρίς ανθρώπινη παρέμβαση. Επίσης, ένα άλλο παράδειγμα αποτελεί η αναζήτηση αγοράς προϊόντων στο διαδίκτυο που έχει σαν αποτέλεσμα να δημιουργούνται κατάλληλες διαφημίσεις για τους καταναλωτές στα μέσα κοινωνικής τους δικτύωσης. [3]

2.3 Είδη μηχανικής μάθησης

Οι τρόποι μάθησης των αλγορίθμων μηχανικής μάθησης είναι κυρίως τέσσερις και μπορούν να παρομοιαστούν με αυτούς που μαθαίνει ένας άνθρωπος, αφού υπάρχει ο τρόπος της επιβλεπόμενης μάθησης, της μη επιβλεπόμενης, της ενισχυτικής μάθησης, ενώ συχνά χρησιμοποιείται και μια υποκατηγορία της επιβλεπόμενης και της μη επιβλεπόμενης, η οποία ονομάζεται ημι-επιβλεπόμενη μάθηση.



Εικόνα 1:Είδη μηχανικής μάθησης

(Πηγή: <https://intellipaat.com/blog/supervised-learning-vs-unsupervised-learning-vs-reinforcement-learning/>)

- **Επιβλεπόμενη μάθηση (Supervised learning) [4]**

Πιο αναλυτικά, στην επιβλεπόμενη μάθηση (Supervised learning) ο αλγόριθμος δέχεται διάφορα δεδομένα ως είσοδο καθώς και τα επιθυμητά αποτελέσματα από έναν επιστήμονα δεδομένων ή αναλυτή δεδομένων και σκοπός του είναι να μάθει ένα γενικό μοτίβο προκειμένου να αντιστοιχεί τις εισόδους με τα αποτελέσματα. Επιπρόσθετα, ο άνθρωπος οφείλει να εισάγει τα χαρακτηριστικά που διαφοροποιούν τα δεδομένα για την καλύτερη ανάπτυξη των προβλέψεων. Όταν ολοκληρωθεί η εκπαίδευση εισάγονται νέα δεδομένα στον αλγόριθμο και μέσω της εκπαίδευσής του πραγματοποιεί τις επιθυμητές προβλέψεις. Μερικές από τις μεθόδους που χρησιμοποιούνται στην επιβλεπόμενη μάθηση είναι τα νευρωνικά δίκτυα, η γραμμική και μη γραμμική παλινδρόμηση (linear, non linear regression), τα δίκτυα Bayes, τα δέντρα αποφάσεων και η ταξινόμηση (classification).

Παραδείγματος χάρη για την κατασκευή ενός αλγορίθμου αναγνώρισης δύο κλάσεων, όπως αυτοκινήτων και ποδηλάτων, ο χρήστης για την εκπαίδευση του αλγορίθμου εισάγει δεδομένα εικόνων διαφόρων αυτοκινήτων και ποδηλάτων. Παράλληλα, αναφέρει ποια κλάση απεικονίζεται σε κάθε εικόνα, αλλά και τα χαρακτηριστικά αυτής, όπως το σχήμα και το μέγεθος. Με το τέλος της εκπαίδευσης και για την δοκιμή του αλγορίθμου εισάγονται νέες εικόνες αυτοκινήτων και ποδηλάτων, με σκοπό την σωστή πρόβλεψη της κάθε κλάσης στην εκάστοτε εικόνα.

- **Μη επιβλεπόμενη μάθηση (Unsupervised learning) [5]**

Από την άλλη πλευρά, στην μη επιβλεπόμενη μάθηση (Unsupervised learning) η εκπαίδευση βασίζεται στην ομαδοποίηση και ανάλυση των δεδομένων εισόδου. Με την μέθοδο αυτή οι αλγόριθμοι ανακαλύπτουν κρυφά μοτίβα και πακέτα δεδομένων που μοιράζονται αντίστοιχα χαρακτηριστικά χωρίς την παρέμβαση του ανθρώπου.

Η ικανότητα της μη επιβλεπόμενης μάθησης στην ανακάλυψη ομοιοτήτων και διαφορών στα δεδομένα την καθιστά ιδανική στην αντιμετώπιση προβλημάτων, όπως στην διερευνητική ανάλυση δεδομένων, στις στρατηγικές διασταυρούμενων πωλήσεων και στην τμηματοποίηση πελατών. Μερικές από τις μεθόδους που χρησιμοποιούνται στην μη επιβλεπόμενη μάθηση είναι η K-means ομαδοποίηση (K-means clustering) και τα νευρωνικά δίκτυα.

Αναφορικά με το παραπάνω παράδειγμα ο αλγόριθμος της μη επιβλεπόμενης μάθησης αντιλαμβάνεται μέσω μοτίβων την διαφορά μεταξύ ενός αυτοκινήτου και ενός ποδηλάτου, χωρίς όμως να είναι εφικτός ο προσδιορισμός του αντικειμένου που απεικονίζεται στην εικόνα ως αυτοκίνητο ή ως ποδήλατο.

- **Ενισχυτική μάθηση (Reinforcement learning) [6]**

Η ενισχυτική μάθηση (Reinforcement learning) είναι παρόμοια με την επιβλεπόμενη μάθηση, αλλά ο αλγόριθμος δεν εκπαιδεύεται από ένα σύνολο δεδομένων. Παρόλα αυτά, αλληλεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να επιτευχθεί ένας στόχος και εκπαιδεύεται μέσω των δοκιμών και των λαθών που πραγματοποιεί. Για την επίτευξη του στόχου ο αλγόριθμος δέχεται ανταμοιβές ή ποινές αναλόγως τις πράξεις του με σκοπό του να είναι η ελαχιστοποίηση των ποινών. Αν και οι ποινές και οι ανταμοιβές εξαρτώνται από τον άνθρωπο, ο στόχος δεν είναι γνωστός και δεν παρέχεται καμία βοήθεια ως προς την επίτευξή του. Ουσιαστικά, ξεκινάει από το μηδέν χωρίς καμία βοήθεια και με διάφορες τεχνικές προσπαθεί να ολοκληρώσει το στόχο του. Ένα χαρακτηριστικό παράδειγμα του τρόπου αυτού είναι η χρήση του στην υποβοήθηση της οδήγησης, στην πλήρη αυτοματοποιημένη οδήγηση, αλλά και στην εκμάθηση ενός παιχνιδιού με αντίπαλο τον άνθρωπο.

- **Ημι-επιβλεπόμενη μάθηση (Semi-supervised learning) [7]**

Τέλος, μεταξύ της επιβλεπόμενης και της μη επιβλεπόμενης μάθησης βρίσκεται μία υποκατηγορία που ονομάζεται ημι-επιβλεπόμενη μάθηση. Με τον τρόπο αυτό ο άνθρωπος κατά την εκπαίδευσή της ημι-επιβλεπόμενη μάθησης εισάγει δεδομένα χωρίς όμως να εισάγει όλα τα επιθυμητά αποτελέσματα όλων των δεδομένων.

2.4 Μέθοδοι Μηχανικής Μάθησης

Όπως αναφέρεται παραπάνω όλα τα είδη αλγορίθμων μηχανικής μάθησης εκτελούνται με διάφορες μεθόδους. Παρακάτω αναλύονται τα πιο χρησιμοποιούμενα μοντέλα μηχανικής μάθησης:

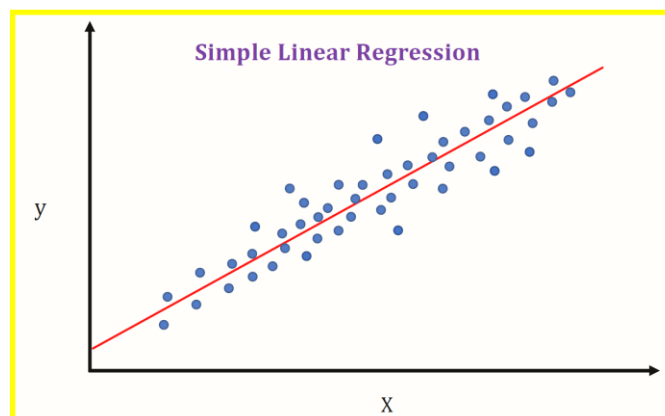
- **Παλινδρόμηση (Regression) [8]**

Η μέθοδος της παλινδρόμησης χρησιμοποιείται στην εκπαίδευση αλγορίθμων επιβλεπόμενης μάθησης. Ο στόχος της παλινδρόμησης είναι η πρόβλεψη αριθμητικών τιμών χρησιμοποιώντας δεδομένα του παρελθόντος. Για παράδειγμα, η πρόβλεψη τιμής μιας κατοικίας βασισμένη στις τιμές άλλων κατοικιών με παρόμοια χαρακτηριστικά, όπως τα τετραγωνικά μέτρα.

Η γραμμική παλινδρόμηση αποτελεί την πιο απλή και πιο βασική μέθοδο παλινδρόμησης, αφού εκφράζεται από τον μαθηματικό τύπο $y = m * x + b$. Η εκπαίδευση πραγματοποιείται με πολλά ζεύγη δεδομένων x, y υπολογίζοντας τη θέση

και τη κλίση μιας γραμμής που ελαχιστοποιεί τη συνολική απόσταση μεταξύ όλων των δεδομένων και της γραμμής. Στην πραγματικότητα, υπολογίζεται η κλίση m και η σταθερά b για μια γραμμή που προσεγγίζει καλύτερα τις παρατηρήσεις στα δεδομένα.

Τέλος, αξίζει να αναφερθεί πως εκτός από την γραμμική παλινδρόμηση υπάρχει και η λογιστική παλινδρόμηση, στην οποία η έξοδος είναι διακριτής μορφής και εφαρμόζεται ο μαθηματικός τύπος $f(x) = 1/(1+e^{-(x)})$.



Εικόνα 2:Γραμμική παλινδρόμηση (linear regression)

(Πηγή: <https://medium.com/@dhaval.sony.504/everything-in-short-about-simple-linear-regression-633fc9f8dd65>)

• Ταξινόμηση (Classification) [9]

Ακόμη μία μέθοδος που εφαρμόζεται στην επιβλεπόμενη μάθηση είναι η ταξινόμηση, η οποία χρησιμοποιείται στην αναγνώριση των κλάσεων (δύο ή περισσότερες) σε νέα δεδομένα με βάση τα δεδομένα εκπαίδευσης. Σε αντίθεση με την παλινδρόμηση η τιμή εξόδου της ταξινόμησης είναι μια κλάση και όχι μια αριθμητική τιμή. Για παράδειγμα, η μέθοδος της ταξινόμησης εφαρμόζεται στην εκτίμηση της ύπαρξης ενός αυτοκινήτου ή ενός φορτηγού σε μια εικόνα. Στο συγκεκριμένο παράδειγμα το αποτέλεσμα μπορεί να είναι είτε ότι υπάρχει αυτοκίνητο στην εικόνα, είτε ότι υπάρχει φορτηγό, είτε ότι δεν υπάρχει τίποτα από τα δύο.

Επιπλέον, υπάρχουν δύο τεχνικές μάθησης ενός αλγορίθμου ταξινόμησης, οι οποίοι ονομάζονται Μάθηση Lazy και Μάθηση Eager.

Με την τεχνική της Μάθησης Lazy στην αρχή αποθηκεύονται τα δεδομένα εκπαίδευσης και αναμένονται τα δεδομένα δοκιμής. Στην συνέχεια, η ταξινόμηση υλοποιείται με βάση τα πιο σχετικά μεταξύ τους δεδομένα που αποθηκεύονται στο σύνολο των δεδομένων εκπαίδευσης. Σε σύγκριση με την τεχνική Μάθησης Eager, η Μάθηση Lazy χρειάζεται λιγότερο χρόνο εκπαίδευσης, αλλά περισσότερο χρόνο πρόβλεψης των δεδομένων δοκιμής.

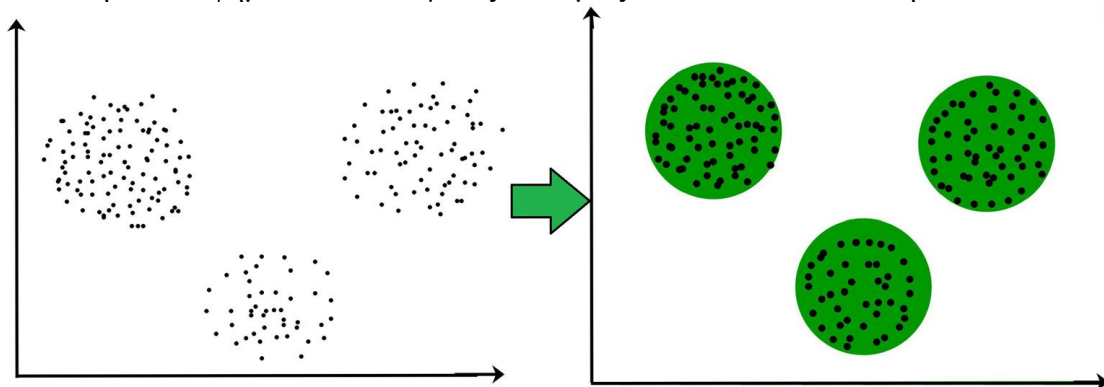
Αντίθετα, στην τεχνική Μάθησης Eager κατασκευάζουν ένα μοντέλο ταξινόμησης με βάση τα δεδομένα εκπαίδευσης πριν λάβουν τα δεδομένα δοκιμής. Επομένως, σε αντίθεση με την Μάθηση Lazy, η Eager χρειάζεται περισσότερο χρόνο εκπαίδευσης και λιγότερο χρόνο για την πρόβλεψη των δεδομένων δοκιμής.

- **Ομαδοποίηση (Clustering) [10]**

Εν αντιθέσει με τις προηγούμενες μεθόδους η ομαδοποίηση χρησιμοποιείται στους αλγόριθμους της μη επιβλεπόμενης μάθησης, αφού κατά την εκπαίδευση ο αλγόριθμος δεν γνωρίζει τα επιθυμητά αποτελέσματα. Ομαδοποίηση ονομάζεται η διαδικασία κατά την οποία κατηγοριοποιούνται τα δεδομένα σε έναν αριθμό ομάδων, έτσι ώστε τα δεδομένα μιας ομάδας να έχουν όμοια χαρακτηριστικά μεταξύ τους και διαφορετικά χαρακτηριστικά με τις υπόλοιπες ομάδες. Στην ουσία, γίνεται μια συλλογή δεδομένων ανάλογα με τις ομοιότητες και τις διαφορές που εμφανίζονται μεταξύ τους.

Για παράδειγμα, στην παρακάτω εικόνα κάθε στοιχείο που απεικονίζεται αποτελεί ένα δεδομένο. Παρατηρείται πως υπάρχουν τρεις ομάδες που τα στοιχεία τους έχουν παρόμοια χαρακτηριστικά. Τέλος, αξίζει να αναφερθεί πως οι ομάδες δεν χρειάζεται να είναι σε κυκλική μορφή, όπως φαίνεται στην εικόνα.

Χαρακτηριστικά παραδείγματα που χρησιμοποιούνται στην καθημερινότητα είναι η εύρεση επικίνδυνων περιοχών σε περιπτώσεις σεισμού και η προβολή κατάλληλων διαφημίσεων ανάλογα τις επιθυμίες του κάθε καταναλωτή.



Εικόνα 3: Παράδειγμα ομαδοποίησης δεδομένων
(Πηγή: <https://www.geeksforgeeks.org/clustering-in-machine-learning/>)

- **Νευρωνικά Δίκτυα (Neural Networks) [11]**

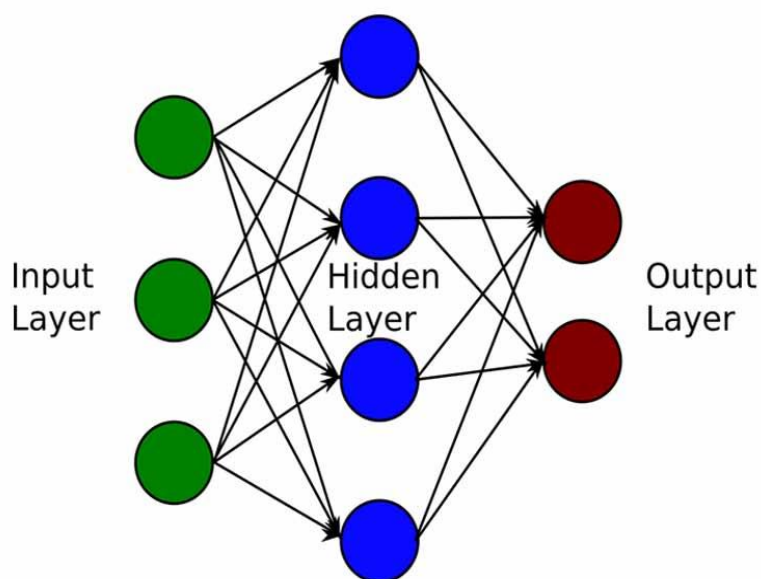
Τα νευρωνικά δίκτυα αποτελούν μέθοδο της επιβλεπόμενης μάθησης. Έχουν εμπνευστεί και σχεδιαστεί με βάση τον ανθρώπινο εγκέφαλο, ο οποίος αποτελείται από πολλούς νευρώνες που επικοινωνούν μεταξύ τους. Τα τεχνητά νευρωνικά δίκτυα αποτελούνται από το στρώμα εισόδου (input layer) τα οποία συνδέονται με τα κρυφά στρώματα (hidden layers) τα οποία με τη σειρά τους συνδέονται με το στρώμα εξόδου (output layers). Για την σωστή πρόβλεψη τα κρυφά στρώματα, τα οποία μπορεί να είναι πολλά ή λίγα αναλόγως το πρόβλημα, τροφοδοτούνται από λειτουργίες που συμβάλουν στην ενεργοποίηση και απενεργοποίηση τους.

Για παράδειγμα, για την ανίχνευση ενός αντικειμένου σε μία εικόνα όλα τα pixels της εικόνας αποτελούν τους νευρώνες του στρώματος εισόδου. Οι νευρώνες του ενός στρώματος συνδέονται με τους νευρώνες του επόμενου στρώματος μέσω των καναλιών, που τους ανατίθεται μια αριθμητική τιμή ονομαζόμενη και ως βάρος (weight). Επομένως, οι τιμές των νευρώνων εισόδου πολλαπλασιάζονται με τα βάρη του εκάστοτε καναλιού και το άθροισμα αυτών στέλνεται ως είσοδος στους νευρώνες του επόμενου κρυφού στρώματος.

Κάθε νευρώνας των κρυφών στρώματων συσχετίζεται με μία διαφορετική αριθμητική τιμή που ονομάζεται bias και η οποία προστίθεται στο άθροισμα που στάλθηκε ως είσοδος. Στη συνέχεια, η τιμή που προκύπτει και αποτελεί μία πιθανότητα ελέγχεται από τη συνάρτηση κατωφλίου (threshold function) ή αλλιώς συνάρτηση ενεργοποίησης (activation function). Συνήθως η συνάρτηση που χρησιμοποιείται είναι η sigmoid ή η ReLU (Rectified Linear Unit). Το αποτέλεσμα της συνάρτησης καθορίζει την ενεργοποίηση ή απενεργοποίηση του συγκεκριμένου νευρώνα. Η διαδικασία επαναλαμβάνεται μέχρι να φτάσει στο τελευταίο στρώμα εξόδου και να γίνει η πρόβλεψη του αντικειμένου. Η μέθοδος που εξηγείται παραπάνω ονομάζεται Forward Propagation.

Σε περίπτωση λάθους πρόβλεψης χρησιμοποιείται η μέθοδος Back Propagation. Η προβλεπόμενη έξοδος συγκρίνεται με την πραγματική έξοδο για την εκτίμηση του σφάλματος και στην συνέχεια τροφοδοτείται το σύστημα για την αναδιαμόρφωση των βαρέων (weights). Η διαδικασία επαναλαμβάνεται μέχρι το μέγεθος του σφάλματος να ελαχιστοποιηθεί.

Παρόλα αυτά, για την ανίχνευση ενός αντικειμένου σε μία εικόνα δεν χρησιμοποιείται ένα απλό τεχνητό νευρωνικό δίκτυο λόγω του μεγέθους των υπολογισμών που χρειάζονται. Αντίθετα, χρησιμοποιείται ένα συνελκτικό νευρωνικό δίκτυο (Convolution Neural Network – CNN), το οποίο δομείται από τα στρώματα ενός απλού νευρωνικού δικτύου σε συνδυασμό με διάφορες τεχνικές που υλοποιούνται πριν από αυτό με σκοπό τη μείωση του αριθμού των υπολογισμών που απαιτούνται για την ανίχνευση των αντικειμένων.



Εικόνα 4: Παράδειγμα απλού νευρωνικού δικτύου
(Πηγή: <https://www.techleer.com/articles/242-backpropagation-a-supervised-learning-neural-network-method/>)

2.5 Συνελκτικά Νευρωνικά Δίκτυα [12]

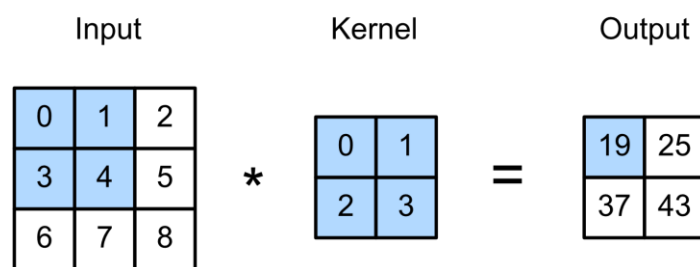
2.5.1 Εισαγωγή

Μία εικόνα αποτελείται από μία συστοιχία πινάκων, οι οποίοι σε κάθε στοιχείο (pixel) τους έχουν μία αριθμητική ακέραια τιμή. Για παράδειγμα, μια ασπρόμαυρη εικόνα αποτελείται από έναν πίνακα, που τα στοιχεία του έχουν μία τιμή μεταξύ 0-255, όπου το μηδέν αναπαριστά το μαύρο και το 255 το άσπρο pixel. Αντίθετα, μία τυπική εικόνα με χρώματα αποτελείται από τρεις πίνακες, όπου ο ένας πίνακας αναπαριστά το χρώμα κόκκινο, ο δεύτερος το χρώμα μπλε και ο τρίτος το χρώμα πράσινο, ενώ οι τιμές που βρίσκονται σε κάθε πίνακα είναι από 0 έως 255. Έτσι, όταν ένα pixel μίας εικόνας είναι κόκκινο τότε ο πίνακας του κόκκινου χρώματος έχει τιμή 255 στο συγκεκριμένο στοιχείο, ενώ οι υπολειπόμενοι πίνακες έχουν τιμή 0 στο αντίστοιχο στοιχείο.

Επομένως, για την καλύτερη ανίχνευση αντικειμένων σε μία εικόνα χρησιμοποιείται ένα συνελκτικό νευρωνικό δίκτυο (Convolution Neural Network – CNN), καθώς αποτελεί μία μέθοδο της βαθιάς μάθησης (deep learning), η οποία χρησιμοποιείται στην επεξεργασία δεδομένων που ακολουθούν ένα μοτίβο. Έχουν σχεδιαστεί έτσι ώστε να προσαρμόζονται αυτόματα και να μαθαίνουν τα χαρακτηριστικά των μοτίβων αυτών. Τα CNN αποτελούνται από τρεις τύπους στρώματων (layers) τα συνελκτικά στρώματα, τα συγκεντρωτικά στρώματα και τα πλήρως συνδεδεμένα στρώματα. Τα πρώτα δύο αναγνωρίζουν τα χαρακτηριστικά μίας εικόνας, ενώ τα πλήρως συνδεδεμένα στρώματα χρησιμοποιούν τα χαρακτηριστικά αυτά για την ανίχνευση του αντικειμένου.

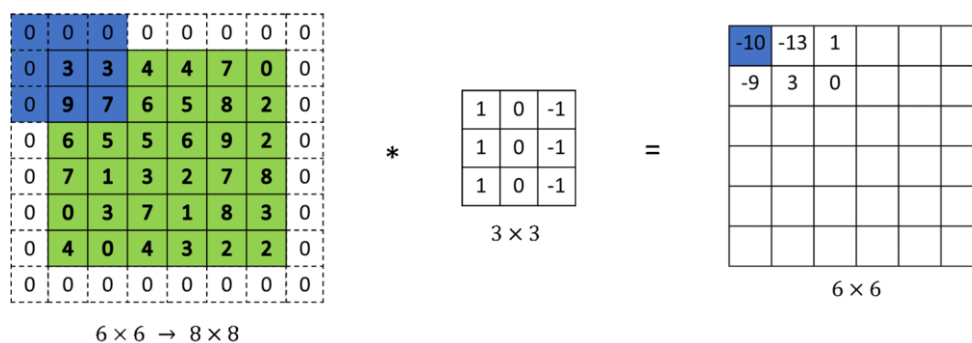
2.5.2 Convolution Layer

Πιο συγκεκριμένα, το συνελκτικό στρώμα είναι ένας τύπος γραμμικής λειτουργίας που χρησιμοποιείται για την εξαγωγή χαρακτηριστικών, όπου ένας πίνακας αριθμών, που ονομάζεται kernel ή φίλτρο, εφαρμόζεται στην εικόνα, η οποία όπως αναλύεται παραπάνω αποτελείται από έναν ή περισσότερους πίνακες ακέραιων αριθμών. Ο πίνακας kernel είναι πολύ μικρότερος σε σχέση με το μέγεθος της εικόνας, με το πιο σύνηθες μέγεθος να είναι 3x3, ενώ μπορεί να είναι 5x5 ή 7x7. Το μέγεθος και οι τιμές που περιλαμβάνει εξαρτώνται από το εκάστοτε χαρακτηριστικό που επιθυμείται να αναγνωριστεί. Το φίλτρο ξεκινά την αναγνώριση των χαρακτηριστικών από την αρχή της εικόνας και στην συνέχεια μετατοπίζεται κατά ένα ή περισσότερα pixel ώστε να καλύψει όλη την εικόνα. Ουσιαστικά, όταν εφαρμόζεται ένα φίλτρο στην εικόνα αναπαράγεται ένας καινούριος πίνακας που ονομάζεται πίνακας χαρακτηριστικών (feature map). Ο καινούριος πίνακας προκύπτει από το άθροισμα του πολλαπλασιασμού των στοιχείων του φίλτρου με των αντίστοιχων της εικόνας, όπως φαίνεται στο παρακάτω απλό παράδειγμα. Σε περίπτωση που η εικόνα είναι με χρώματα RGB δηλαδή αποτελείται από τρεις πίνακες τότε και το φίλτρο θα αποτελείται από τρεις πίνακες.



Εικόνα 5: Χρήση συνελικτικού στρώματος
(Πηγή: <https://programmer.ink/think/image-convolution-operation.html>)

Παρόλα αυτά, η λειτουργία που περιγράφεται παραπάνω δεν επιτρέπει στο κέντρο του φίλτρου να επικαλύπτει τα εξωτερικά στοιχεία των πινάκων της εικόνας, αλλά επίσης μειώνεται το ύψος και το πλάτος του πίνακα χαρακτηριστικών σε σχέση με την εικόνα εισόδου. Η τεχνική του Padding αντιμετωπίζει το πρόβλημα αυτό προσθέτοντας καινούριες γραμμές και στήλες μηδενικών τιμών σε κάθε πλευρά της εικόνας εισόδου. Με αυτόν τον τρόπο, επιτυγχάνεται η κάλυψη του εξωτερικού στοιχείου της εικόνας από το κεντρικό στοιχείο των φίλτρων, αλλά και δεν διαφέρει το μέγεθος του πίνακα χαρακτηριστικών από το μέγεθος της εικόνας εισόδου.



Εικόνα 6: Εφαρμογή τεχνικής Padding
(Πηγή: <https://datahacker.rs/what-is-padding-cnn/>)

2.5.3 Pooling Layer

Από την άλλη πλευρά, τα συγκεντρωτικά στρώματα (pooling layers) χρησιμοποιούνται για την μείωση του μεγέθους των πινάκων χαρακτηριστικών, με σκοπό να ελαχιστοποιηθούν οι υπολογισμοί των βαρέων των πλήρως συνδεδεμένων δικτύων. Ουσιαστικά, πρόκειται για ακόμη ένα φίλτρο με το μέγεθός του να είναι συνήθως 2x2, το οποίο αρχίζει από την αρχή του πίνακα χαρακτηριστικών και εξάγει σε ένα νέο πίνακα τη μέγιστη τιμή που βρίσκονται στα στοιχεία αυτά και συνεχίζει με βήμα (stride) 2 μέχρι να καλύψει όλο τον πίνακα. Η μέθοδος αυτή ονομάζεται max pooling, ενώ υπάρχει και η μέθοδος average pooling, η οποία εξάγει τον μέσο όρο των τιμών που βρίσκονται στον πίνακα χαρακτηριστικών.

Image Matrix				Max Pool	
2	1	3	1	2	4
1	0	1	4	7	9
0	6	9	5		
7	1	4	1		

Εικόνα 7: Παράδειγμα max Pooling

(Πηγή: <https://austingwalters.com/convolutional-neural-networks-cnn-to-classify-sentences/max-pooling/>)

2.5.4 Fully Connected Layer

Στη συνέχεια, ο πίνακας χαρακτηριστικών μετατρέπεται σε έναν μονοδιάστατο πίνακα και εισάγεται σε ένα πλήρως συνδεδεμένο νευρωνικό δίκτυο, το οποίο είναι ίδιο με το νευρωνικό δίκτυο που αναλύεται παραπάνω με τη διαφορά πως όλοι οι νευρώνες του ενός στρώματος συνδέονται με όλους τους νευρώνες του επόμενου στρώματος. Αν υποθέσουμε ότι το νευρωνικό δίκτυο έχει εκπαιδευτεί κατάλληλα τότε και τα βάρη (weights) του έχουν τις κατάλληλες τιμές, ώστε να αναγνωριστούν τα αντικείμενα σε μια εικόνα από το τελευταίο στρώμα του με τη βοήθεια της κατάλληλης συνάρτησης ενεργοποίησης.

Η συνάρτηση ενεργοποίησης που χρησιμοποιείται στο τελευταίο στρώμα διαφέρει από εκείνες στα προηγούμενα στρώματα. Πιο συγκεκριμένα, η συνάρτηση που ενδείκνυται για την αναγνώριση των αντικειμένων ονομάζεται softmax function, η οποία μετατρέπει τις τιμές του τελευταίου στρώματος σε πιθανότητες για κάθε κλάση που μπορεί να αναγνωριστεί, όπου κάθε τιμή βρίσκεται μεταξύ 0 και 1 και όλες οι τιμές αθροίζουν στο 1. Έτσι, η μεγαλύτερη τιμή αντικατοπτρίζει την κλάση που έχει αναγνωριστεί στην εικόνα.

Για παράδειγμα, αν ένα CNN αναγνωρίζει τρεις κλάσεις, όπως αυτοκίνητο, άνθρωπο και ποδήλατο και δοθεί μία εικόνα που απεικονίζει ένα αυτοκίνητο, τότε η συνάρτηση softmax θα μετατρέψει τις τιμές του τελευταίου στρώματος σε πιθανότητες που ανταποκρίνονται στην ύπαρξη του αυτοκινήτου στην εικόνα. Με αυτόν τον τρόπο, οι τιμές των τριών νευρώνων του τελευταίου στρώματος θα είναι 0.95 για την κλάση του αυτοκινήτου, 0.01 για την κλάση του ανθρώπου και 0,04 για την κλάση του ποδηλάτου.

2.6 Τεχνικές αλγορίθμων ανίχνευσης αντικειμένων σε εικόνα

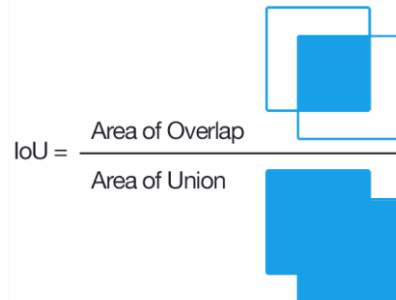
Παρακάτω αναλύονται δύο τεχνικές που χρησιμοποιούνται τόσο κατά την εκπαίδευση αλγορίθμων ανίχνευσης αντικειμένων όσο και κατά την εφαρμογή των αλγορίθμων αυτών σε εικόνες για την ανίχνευση των αντικειμένων.

2.6.1 Intersection over Union (IoU) [13]

Οι αλγόριθμοι ανίχνευσης αντικειμένων σε εικόνες πρέπει όχι μόνο να απαντούν στο ερώτημα αν βρίσκεται ένα αντικείμενο στην εικόνα αλλά και να σχεδιάζουν ένα πλαίσιο οριοθέτησης (bounding box) γύρω από το αντικείμενο που ανιχνεύθηκε.

Η μέθοδος του Intersection over Union εφαρμόζεται κυρίως κατά την εκπαίδευση του αλγορίθμου και χρησιμοποιείται στην ποσοτικοποίηση της ποιότητας της

πρόβλεψης του πλαισίου. Πρακτικά, συγκρίνεται το προβλεπόμενο πλαίσιο σε σχέση με το πραγματικό πλαίσιο (ground truth) που περιλαμβάνει το αντικείμενο. Ειδικότερα, το IoU υπολογίζει τον λόγο της τομής των δύο πλαισίων προς την ένωση αυτών, όπως φαίνεται στο παρακάτω παράδειγμα.

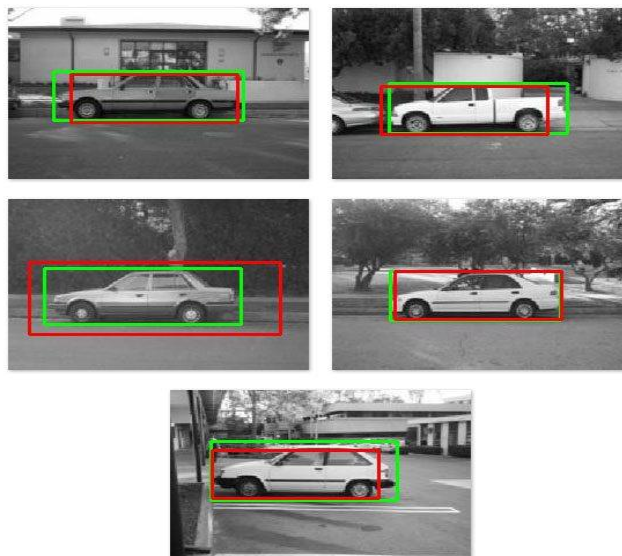


Εικόνα 8: Η μέθοδος Intersection over Union

(Πηγή: https://commons.wikimedia.org/wiki/File:Intersection_over_Union_-_visual_equation.png)

Αν το αποτέλεσμα που προκύπτει είναι μεγαλύτερο ή ίσο του 0.5 τότε ο αλγόριθμος πραγματοποίησε καλή πρόβλεψη. Σε διαφορετική περίπτωση ο αλγόριθμος εκτελείται ξανά μέχρι να προβλέψει ένα σωστό πλαίσιο.

Για παράδειγμα, στις παρακάτω εικόνες το πράσινο πλαίσιο είναι αυτό που περιβάλλει τα αυτοκίνητα, ενώ το κόκκινο αποτελεί την πρόβλεψη του αλγορίθμου.



Εικόνα 9: Παράδειγμα μεθόδου Intersection over Union

(Πηγή: <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>)

2.6.2 Non-Maximum Suppression (NMS) [14]

Ένα άλλο πρόβλημα που αντιμετωπίζουν οι αλγόριθμοι ανίχνευσης αντικειμένων είναι η ύπαρξη πολλών προβλέψεων με μικρές διαφορές, οι οποίες αντιστοιχούν στο ίδιο αντικείμενο. Η μέθοδος του Non-Maximum Suppression χρησιμοποιείται για να συγχωνεύσει τα αλληλεπικαλυπτόμενα πλαίσια.

Αρχικά, εντοπίζεται το πλαίσιο που κατέχει την μεγαλύτερη πιθανότητα πρόβλεψης για την ύπαρξη ενός αντικειμένου στην εικόνα. Στη συνέχεια, συγκρίνεται με όλα τα άλλα πλαίσια με μικρότερη πιθανότητα και απορρίπτεται όσα έχουν επικάλυψη IoU μεγαλύτερη από μία προκαθορισμένη τιμή.



Εικόνα 10: Παράδειγμα μεθόδου Non-Maximum Suppression
(Πηγή: https://www.researchgate.net/figure/Non-Maximal-Suppression_fig5_345061606)

2.7 Μέθοδος αξιολόγησης αλγορίθμων ανίχνευσης αντικειμένων [15]

Ένα συνελκτικό νευρωνικό δίκτυο πρέπει να αξιολογείται για την απόδοσή του να ανιχνεύει σωστά τα αντικείμενα για τα οποία εκπαιδεύτηκε να αναγνωρίζει. Η πιο δημοφιλής μέθοδος αξιολόγησης των συνελκτικών νευρωνικών δικτύων για προβλήματα ανίχνευσης αντικειμένων σε μία εικόνα είναι το ποσοστό Mean Average Precision. Η μέθοδος αυτή βασίζεται σε δύο μονάδες μέτρησης, οι οποίες είναι η Precision και η Recall, όπως φαίνεται παρακάτω.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

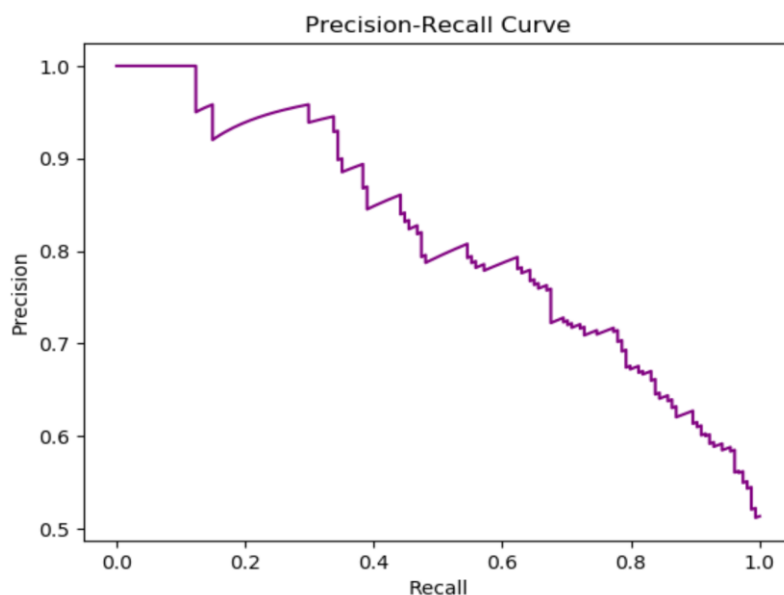
True Positive (TP) αποτελούν οι προβλέψεις που αναγνώρισαν ένα αντικείμενο, το οποίο βρίσκεται στην εικόνα.

False Positive (FP) αποτελούν οι προβλέψεις που αναγνώρισαν ένα αντικείμενο, το οποίο όμως δεν βρίσκεται στην εικόνα.

False Negative (FN) αποτελούν οι προβλέψεις που δεν αναγνώρισαν ένα αντικείμενο, το οποίο όμως βρίσκεται στην εικόνα.

True Negative (TN) αποτελούν οι προβλέψεις που δεν αναγνώρισαν ένα αντικείμενο, το οποίο δεν βρίσκεται στην εικόνα.

Η εξίσωση της Precision αποτελεί έναν τρόπο μέτρησης της ακρίβειας της πρόβλεψης του συνελκτικού νευρωνικού δικτύου στην κατηγοριοποίηση των αντικειμένων και υπολογίζεται ως το ποσοστό των σωστών προβλέψεων προς το άθροισμα των σωστών και λανθασμένων προβλέψεων. Η εξίσωση της Recall, από την άλλη πλευρά, αποτελεί έναν τρόπο μέτρησης του ποσοστού των αντικειμένων που βρέθηκαν και υπολογίζεται ως το ποσοστό των σωστών προβλέψεων προς το άθροισμα των σωστών προβλέψεων και των προβλέψεων που δεν ανίχνευσαν σωστά την ύπαρξη ενός αντικειμένου που βρισκόταν στην εικόνα. Στην συνέχεια, οι τιμές των precision και recall τοποθετούνται σε ένα δυσδιάστατο άξονα και σχηματίζεται ένα διάγραμμα όπως το παρακάτω.



Εικόνα 11: Παράδειγμα γραφήματος precision-recall
(Πηγή: <https://www.statology.org/precision-recall-curve-python/>)

Το εμβαδόν του γραφήματος αποτελεί την τιμή του Average Precision (AP) και υπολογίζεται από τον παρακάτω μαθηματικό τύπο.

$$AP = \int_0^1 p(r)dr$$

Επιπρόσθετα, ένας ακόμη τρόπος αξιολόγησης των αλγορίθμων ανίχνευσης αντικειμένων είναι το IoU μεταξύ των πλαισίων οριοθέτησης που δημιουργούνται από τον αλγόριθμο και των πραγματικών πλαισίων οριοθέτησης ενός αντικειμένου σε μία εικόνα. Το αποτέλεσμα αποτελεί το ποσοστό επικάλυψης των δύο προαναφερθέντων πλαισίων οριοθέτησης, οπότε οι τιμές του βρίσκονται μεταξύ του 0 και του 1. Ουσιαστικά, ο λόγος που χρησιμοποιείται η μέθοδος του IoU είναι η καθιέρωση μιας πρόβλεψης πλαισίου οριοθέτησης ως σωστής ή λανθασμένης αναλόγως αν το ποσοστό του αποτελέσματος βρίσκεται πάνω ή κάτω από μία τιμή κατωφλίου (threshold). Συνήθως η τιμή του κατωφλίου είναι 0.5 (AP@0.5).

Επιπλέον, για την εύρεση του ποσοστού επιτυχίας ενός αλγορίθμου ανίχνευσης αντικειμένων απαιτείται ο υπολογισμός του μέσου όρου όλων των AP, δηλαδή όλων των κλάσεων, που επιθυμείται ένας αλγόριθμος να ανιχνεύσει. Ο μαθηματικός τύπος που υπολογίζει το mean Average Precision (mAP) είναι ο παρακάτω, όπου N είναι ο αριθμός των κλάσεων.

$$\text{mAP} = \frac{1}{N} \sum_{i=1}^N \text{AP}_i$$

Τέλος, συνηθίζεται να γίνεται ο υπολογισμός του mAP εφαρμόζοντας διάφορες τιμές κατωφλίου. Με αυτόν τον τρόπο, σχηματίζεται μια γενικότερη άποψη σχετικά με την επιτυχία ενός αλγορίθμου να αναγνωρίσει και να πλαισιώσει ένα αντικείμενο.

2.8 Αλγόριθμοι ανίχνευσης αντικειμένων

2.8.1 Εισαγωγή

Η ανίχνευση αντικειμένων (object detection) αποτελεί ένα κύριο κεφάλαιο στην υπολογιστική όραση (computer vision), το οποίο απασχολεί τους επιστήμονες αρκετά χρόνια. Μάλιστα, από τα τέλη του περασμένου αιώνα δημοσιεύονται άρθρα σε επιστημονικά περιοδικά, ωστόσο η υπολογιστικής ισχύ της εποχής εκείνης απέτρεψαν την περαιτέρω ενασχόληση επί του θέματος.



Εικόνα 12: Δημοσιεύσεις αλγορίθμων ανίχνευσης αντικειμένων
(Πηγή: <https://arxiv.org/pdf/1905.05055.pdf>)

Όπως παρατηρείται από την παραπάνω εικόνα οι δημοσιεύσεις αυξήθηκαν ραγδαία από τα τέλη τις δεκαετίας του 2000, δηλαδή την εποχή που οι επεξεργαστές και οι κάρτες γραφικών των υπολογιστών είχαν εξελιχθεί σε τέτοιο βαθμό που επέτρεπαν τις δοκιμές για την εύρεση του κατάλληλου αλγορίθμου ανίχνευσης αντικειμένων. Επίσης, τα τελευταία χρόνια η ραγδαία ανάπτυξη των τεχνικών βαθιάς μάθησης (deep learning), τα οποία χρησιμοποιούνται από αλγορίθμους που αναπτύσσονται ακόμη και σήμερα, οδήγησαν σε αξιοσημείωτα και καινοτόμα αποτελέσματα. Την σημερινή εποχή, μάλιστα, τέτοιοι αλγόριθμοι αποτελούν αναπόσπαστο κομμάτι πολλών καθημερινών εφαρμογών, όπως η αυτόνομη οδήγηση

και η ανίχνευση αντικειμένων σε βίντεο από κάμερες ασφαλείας με απώτερο σκοπό την παρακολούθηση των αντικειμένων αυτών. [16]

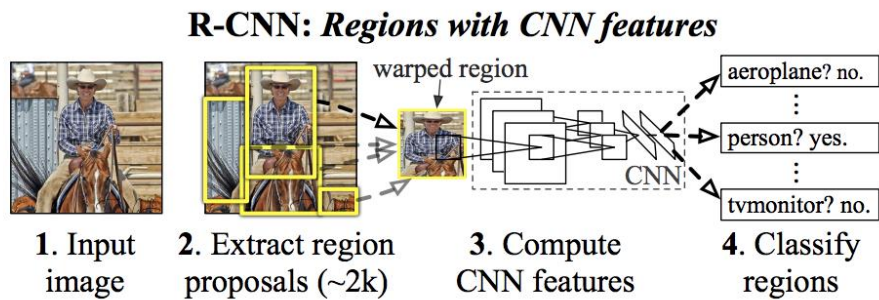
Στην πραγματικότητα, το πρόβλημα που προσπαθούν να λύσουν οι συγκεκριμένοι αλγόριθμοι δεν είναι μόνο η απάντηση στο ερώτημα ύπαρξης ενός αντικειμένου στην εικόνα αλλά και η τοποθέτηση ενός πλαισίου οριοθέτησης (bounding box) γύρω από αυτό. Στη συνέχεια αναλύονται διάφοροι αλγόριθμοι ανίχνευσης αντικειμένων, που αναπτύχθηκαν τα τελευταία χρόνια.

2.8.2 Αλγόριθμος R-CNN [17]

Μία από τις πρώτες καινοτόμες χρήσης των συνελκτικών νευρωνικών δικτύων σε αλγόριθμο ανίχνευσης αντικειμένων είναι το λεγόμενο Regions with Convolution Neural Network ή εν συντομία R-CNN. Ο αλγόριθμος δημοσιεύθηκε από τους Ross Girshick, Jeff Donahue, Trevor Darrell και Jitendra Malik το 2013 και έφερε επανάσταση στην υπολογιστική όραση (computer vision), αφού κατάφερε να πετύχει υψηλή απόδοση και εκπληκτικά αποτελέσματα στην ανίχνευση των αντικειμένων σε μια εικόνα την χρονιά εκείνη.

Η αρχή λειτουργίας του έγκειται στην δημιουργία προτάσεων περιοχής (region proposals), οι οποίες στη συνέχεια εισάγονται σε ένα συνελκτικό νευρωνικό δίκτυο για την κατηγοριοποίηση του αντικειμένου που απεικονίζεται στην εκάστοτε περιοχή. Η δημιουργία προτάσεων περιοχής υλοποιείται με τη χρήση του αλγορίθμου επιλεκτικής αναζήτησης (selective search algorithm), από τον οποίο εξάγονται μόνο 2000 περιοχές σε κάθε εικόνα. Η επιλεκτική αναζήτηση των περιοχών δεν συμβαίνει τυχαία, αλλά βασίζεται στο χρώμα, στο μέγεθος και στο σχήμα, με σκοπό να συνδυαστούν παρόμοιες περιοχές που μπορεί να σχηματίζουν ένα αντικείμενο. Με αυτόν τον τρόπο οι προτάσεις περιοχών, που δημιουργούνται, είναι διαφόρων μεγεθών. Στη συνέχεια, κάθε περιοχή μετατρέπεται σε ένα σταθερό μέγεθος και εισάγεται στο συνελκτικό νευρωνικό δίκτυο για την εξαγωγή πινάκων χαρακτηριστικών. Κατόπιν, με τη βοήθεια της μεθόδου Support Vector Machine, η οποία εκπαιδεύεται ανεξάρτητα για κάθε κλάση, εξάγεται η πιθανότητα παρουσίας ενός αντικειμένου στην περιοχή αυτή. Η μέθοδος του Support Vector Machine έγκειται στην κατηγορία της επιβλεπόμενης μηχανικής μάθησης και χρησιμοποιείται κυρίως σε προβλήματα τύπου κατηγοριοποίησης, αλλά και πιο σπάνια σε προβλήματα τύπου ταξινόμησης.

Παρά τα εξαιρετικά αποτελέσματα του αλγορίθμου R-CNN υπάρχουν πολλά μειονεκτήματα στην μεθοδολογία που χρησιμοποιεί. Πρώτα από όλα, η ύπαρξη 2000 προτάσεων περιοχών και η μη ταυτόχρονη εκπαίδευση του συνελκτικού νευρωνικού δικτύου και της μεθόδου Support Vector Machine (SVM) έχουν ως αποτέλεσμα να απαιτείται αρκετός χρόνος για την εκπαίδευση του αλγορίθμου. Επίσης, η ανίχνευση των αντικειμένων σε μία εικόνα με την χρήση του συγκεκριμένου αλγορίθμου πραγματοποιείται σε 47 δευτερόλεπτα, που έχει ως συνέπεια να μην επιτρέπεται η χρήση του σε προβλήματα πραγματικού χρόνου. Επιπλέον, ο αλγόριθμος επιλεκτικής αναζήτησης δεν εκπαιδεύεται, που σημαίνει πως υπάρχει πιθανότητα να μην προτείνει τις καταλληλότερες περιοχές για ανίχνευση αντικειμένων. Παρόλα αυτά, έχει υψηλό ποσοστό 66% mAP στο VOC 2007 dataset.



Εικόνα 13: Παράδειγμα αλγορίθμου R-CNN
(Πηγή: <https://arxiv.org/abs/1311.2524>)

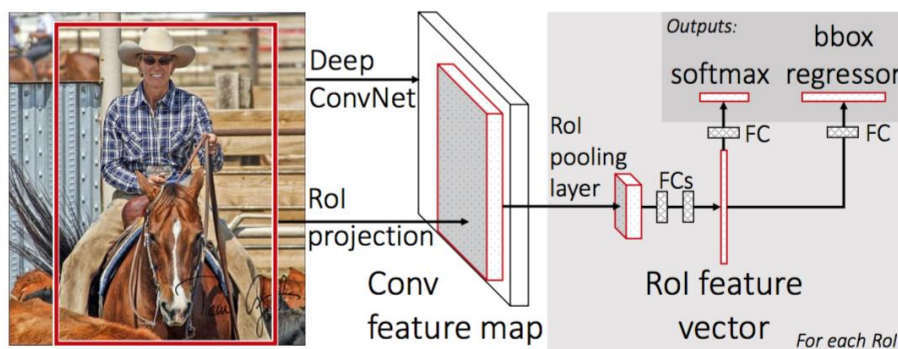
2.8.3 Αλγόριθμος Fast R-CNN [18]

Ένας από τους δημιουργούς του αλγορίθμου R-CNN Ross Girshick δημοσίευσε δύο χρόνια αργότερα, τον Απρίλιο του 2015 μία καλύτερη έκδοση ονομάζοντάς την Fast R-CNN καταφέροντας να επιλύσει πολλά από τα μειονεκτήματα του R-CNN.

Ο αλγόριθμος Fast R-CNN δέχεται ως είσοδο ολόκληρη την εικόνα και ένα σύνολο από προτάσεις περιοχής με την βοήθεια του αλγορίθμου επιλεκτικής αναζήτησης. Αρχικά, το συνελκτικό νευρωνικό δίκτυο επεξεργάζεται την εικόνα με πολλά συνελκτικά και συγκεντρωτικά στρώματα, τα οποία χρησιμοποιούν τη μέθοδο max pooling, με σκοπό την παραγωγή των χαρτών χαρακτηριστικών. Στη συνέχεια, για κάθε περιοχή, που προτάθηκε από τον αλγόριθμο επιλεκτικής αναζήτησης, εξάγεται ένα προκαθορισμένου μεγέθους διάνυσμα χαρακτηριστικών από τον χάρτη χαρακτηριστικών με την βοήθεια ενός συγκεντρωτικού στρώματος, που ονομάζεται region of interest (RoI) pooling layer. Κάθε διάνυσμα χαρακτηριστικών τροφοδοτείται σε μια ακολουθία πλήρως συνδεδεμένων στρωμάτων, τα οποία καταλήγουν σε δύο εξωτερικά στρώματα. Το πρώτο εξωτερικό στρώμα χρησιμοποιεί την συνάρτηση softmax για την κατηγοριοποίηση του αντικειμένου, ενώ το δεύτερο στρώμα για την εξαγωγή τεσσάρων πραγματικών αριθμών που αντικατοπτρίζουν την πλαισίωση του αντικειμένου στην εικόνα.

Όσον αναφορά το RoI συγκεντρωτικό στρώμα, η αρχή λειτουργίας του έγκειται στην χρήση του max pooling για την μετατροπή των χαρακτηριστικών οποιασδήποτε περιοχής ενδιαφέροντος (region of interest) σε ένα μικρό χάρτη χαρακτηριστικών με προκαθορισμένο μέγεθος. Κάθε περιοχή ορίζεται από μία πλειάδα (r, c, h, w) , όπου το r, c καθορίζει την επάνω αριστερή γωνία στην εικόνα, ενώ τα h, w το ύψος και το πλάτος αντίστοιχα.

Τα παραπάνω είχαν ως αποτέλεσμα την δραματική μείωση του χρόνου εκπαίδευσης και ανίχνευσης των αντικειμένων σε μία εικόνα. Ο αλγόριθμος Fast R-CNN είναι κλάσεις ταχύτερος από τον R-CNN, γιατί δεν τροφοδοτούνται 2000 περιοχές για κάθε εικόνα στο συνελκτικό νευρωνικό δίκτυο. Αντιθέτως, ολόκληρη η εικόνα εισάγεται σε αυτό, οπότε η συνέλιξη πραγματοποιείται μόνο μία φορά. Με αυτόν τον τρόπο, ο χρόνος εκπαίδευσης είναι 9 φορές πιο γρήγορος από τον αλγόριθμο R-CNN, ενώ ο χρόνος ανίχνευσης των αντικειμένων σε μία εικόνα πραγματοποιείται σε 2 δευτερόλεπτα έναντι των 47 δευτερολέπτων του αλγορίθμου R-CNN. Ολοκληρώνοντας, το ποσοστό του Fast R-CNN είναι στο 66.9% mAP στο VOC 2007 dataset, ποσοστό υψηλότερο έναντι του 66% του R-CNN.



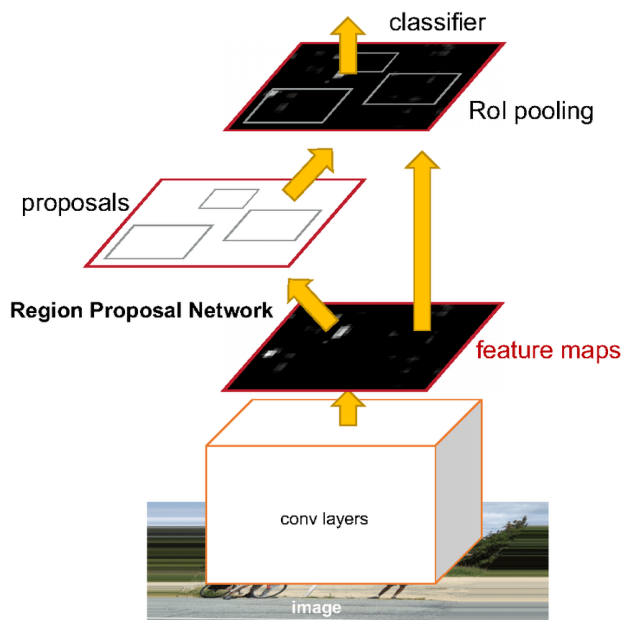
Εικόνα 14: Παράδειγμα αλγορίθμου Fast R-CNN
(Πηγή: <https://arxiv.org/abs/1504.08083>)

2.8.4 Αλγόριθμος Faster R-CNN [19]

Οι παραπάνω δύο αλγόριθμοι χρησιμοποιούν την επιλεκτική αναζήτηση ως μέσο για την εξαγωγή προτάσεων περιοχών, που μπορεί περιλαμβάνουν ένα αντικείμενο. Παρόλα αυτά, η μέθοδος της επιλεκτικής αναζήτησης αποτελεί μία αργή και χρονοβόρα διαδικασία, που επηρεάζει δραματικά την ταχύτητα και την απόδοση του δικτύου. Για τον παραπάνω λόγο, τον Ιούνιο του 2015 δημοσιεύθηκε ένας καλύτερος αλγόριθμος ανίχνευσης αντικειμένων από τους Shaoqing Ren, Kaiming He, Ross Girshick και Jian Sun, που ονομάζεται Faster R-CNN. Η βασική ιδέα είναι η χρήση ενός δικτύου πρότασης περιοχών (Regional Proposed Network) έναντι της μεθόδου επιλεκτικής αναζήτησης, που χρησιμοποιείται στους αλγορίθμους R-CNN και Fast R-CNN αντίστοιχα.

Ο αλγόριθμος Faster R-CNN συγκροτείται από δύο ενότητες. Η πρώτη ενότητα αποτελεί ένα πλήρως συνδεδεμένο συνελκτικό νευρωνικό δίκτυο (RPN), που χρησιμοποιείται για την πρόταση περιοχών, ενώ η δεύτερη ενότητα αποτελεί τον αλγόριθμο Fast R-CNN που αξιοποιεί τις περιοχές που έχουν προταθεί. Ουσιαστικά, όπως στον αλγόριθμο Fast R-CNN, έτσι και στον αλγόριθμο Faster R-CNN, η εικόνα εισάγεται σε ένα συνελκτικό νευρωνικό δίκτυο από τον οποίο εξάγεται ένας χάρτης χαρακτηριστικών. Σε αντίθεση με τον Fast R-CNN, στους χάρτες χαρακτηριστικών που παράγονται από το συνελκτικό νευρωνικό δίκτυο του Faster R-CNN δεν χρησιμοποιείται η επιλεκτική αναζήτηση για την πρόταση των περιοχών που προορίζονται για ανίχνευση ενός αντικειμένου, αλλά ένα δίκτυο RPN. Το δίκτυο RPN δέχεται ως είσοδο τον χάρτη χαρακτηριστικών του συνελκτικού νευρωνικού δικτύου και εξάγει πλαίσια αγκύρωσης (anchor boxes), ως προτάσεις περιοχών, μέσω ενός συρόμενου παραθύρου (sliding window), που εφαρμόζεται στον χάρτη χαρακτηριστικών. Για κάθε συρόμενο παράθυρο, το δίκτυο δημιουργεί συνολικά 9 πλαίσια αγκύρωσης με τα μεγέθη τους να είναι 128x128, 256x256 και 512x512, ενώ οι αναλογίες τους είναι 1:1, 1:2 και 2:1 για κάθε ένα από αυτά. Στην συνέχεια, με τη χρήση του RoI συγκεντρωτικού στρώματος αναδιαμορφώνονται οι προβλεπόμενες προτάσεις περιοχών και τροφοδοτούνται σε μια ακολουθία πλήρως συνδεδεμένων στρωμάτων, με σκοπό την αναγνώριση και πλαισίωση του αντικειμένου στην εικόνα.

Με την χρήση του δικτύου RPN έναντι της επιλεκτικής αναζήτησης το Faster R-CNN αποδείχθηκε ταχύτερο και με καλύτερα αποτελέσματα από το Fast R-CNN. Η ανίχνευση των αντικειμένων σε μία εικόνα πραγματοποιείται σε μόλις 0.2 δευτερόλεπτα έναντι των 2 δευτερολέπτων του Fast R-CNN. Ολοκληρώνοντας, το ποσοστό του Faster R-CNN 66.9 mAP είναι το ίδιο με του Fast R-CNN στο VOC 2007 dataset.



Εικόνα 15: Παράδειγμα αλγορίθμου Faster R-CNN
(Πηγή: <https://arxiv.org/abs/1506.01497>)

2.8.5 Αλγόριθμος Yolo

Μια διαφορετική προσέγγιση, η οποία παραλείπει τη μέθοδο πρότασης περιοχών και εκτελεί τον εντοπισμό και την κατηγοριοποίηση των αντικειμένων σε μία εικόνα είναι ο αλγόριθμος YOLO. YOLO σημαίνει You Only Look Once, δηλαδή κοιτάς μόνο μία φορά, και ο τρόπος λειτουργίας του συνάπτει με το όνομά του. Ουσιαστικά, ένα ενιαίο συνελκτικό νευρωνικό δίκτυο, που βασίζεται στην παλινδρόμηση, ανιχνεύει αντικείμενα σαρώνοντας την εικόνα μόνο μία φορά, εξού και το όνομά του.

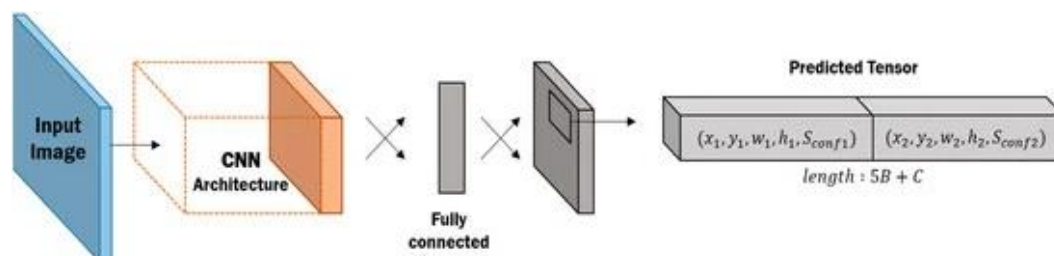
Η πρώτη έκδοση του αλγορίθμου YOLO δημοσιεύθηκε από τους Joseph Redmon, Santosh Divvala, Ross Girshick και Ali Farhadi το 2015, φέρνοντας επανάσταση στον τομέα της υπολογιστικής όρασης (computer vision), αφού ήταν εφικτή η ανίχνευση αντικειμένων σε πραγματικό χρόνο. Επιπλέον, τον επόμενο χρόνο ο Joseph Redmon και ο Ali Farhadi δημοσίευσαν την δεύτερη έκδοση του YOLO (γνωστή και ως YOLO9000), ενώ το 2018 την τρίτη βελτιώνοντας έτσι τον αλγόριθμο τόσο στην αύξηση της ακρίβειας στην αναγνώριση των αντικειμένων όσο και στην ταχύτητα. Η τέταρτη έκδοση του YOLO δημοσιεύθηκε το 2020 από μία διαφορετική ομάδα επιστημών και πιο συγκεκριμένα από τους Alexey Bochkovskiy, Chien-Yao Wang και Hong-Yuan Mark Liao, οι οποίοι χρησιμοποίησαν διάφορες τεχνικές για την καλύτερη ανίχνευση των αντικειμένων χωρίς να επηρεάζεται η ταχύτητα. Οι διαφορές όλων των εκδόσεων του YOLO βρίσκονται στα συνελκτικά

νευρωνικά δίκτυα και στις μεθόδους που χρησιμοποιούν σε αυτά και όχι στην γενική μεθοδολογία.

Η πρώτη έκδοση του YOLO εκπαιδεύτηκε με βάση το Pascal VOC dataset που περιλαμβάνει 20 κλάσεις αντικειμένων, ενώ η τρίτη και η τέταρτη έκδοση έχουν εκπαιδευτεί με βάση το COCO (Common Objects in Context) dataset της Microsoft, το οποίο περιλαμβάνει 80 κλάσεις καθημερινών αντικειμένων. Η δεύτερη έκδοση, ωστόσο, έχει εκπαιδευτεί και με τα δύο dataset. Επιπλέον, αξίζει να σημειωθεί ότι στο COCO dataset υπάρχουν 2.5 εκατομμύρια επισημασμένες αναφορές των αντικειμένων σε 328000 εικόνες, γεγονός που διευκολύνει την εκπαίδευση του YOLO στην αναγνώριση των συγκεκριμένων αντικειμένων.

Η αρχιτεκτονική του αλγορίθμου είναι εκπληκτικά γρήγορη από την πρώτη έκδοσή της, δεδομένου ότι μπορεί να ανιχνεύει αντικείμενα στα 45 καρέ το δευτερόλεπτο, για αυτό το λόγο χρησιμοποιείται σε προβλήματα αναγνώρισης αντικειμένων σε πραγματικό χρόνο. Παράλληλα, ο αλγόριθμος YOLO έχει πολύ μικρότερες πιθανότητες να εμφανίσει λάθος πρόβλεψη στο παρασκήνιο μιας εικόνας σε σχέση με τους αλγορίθμους που αναπτύσσονται παραπάνω. Ωστόσο, ένα μειονέκτημα του YOLO παρουσιάζεται κατά την ανίχνευση μικρών αντικειμένων είτε λόγω της φύσης τους είτε λόγω κακής λήψης της εικόνας από μακρινή απόσταση.

Η αρχή λειτουργίας του διαφέρει ολοκληρωτικά σε σχέση με τους προηγούμενους αλγορίθμους. Το YOLO δέχεται την εικόνα ως είσοδο και την διαιρεί σε ένα πλέγμα $S \times S$ κελιών. Αν το μέσο ενός αντικειμένου βρίσκεται σε ένα κελί, τότε το κελί αυτό ευθύνεται για την ανίχνευση του συγκεκριμένου αντικειμένου. Πιο συγκεκριμένα, ένα συνελκτικό νευρωνικό δίκτυο εξάγει πίνακες χαρακτηριστικών και με τη χρήση μιας συνάρτησης ενεργοποίησης εξάγεται μέσω των πλήρως συνδεδεμένων στρωμάτων ένας πίνακας που περιλαμβάνει τόσο τις πιθανότητες της κάθε κλάσης όσο και τις συντεταγμένες του πλαισίου, για το κάθε κελί, όπως φαίνεται στην παρακάτω εικόνα. [20]



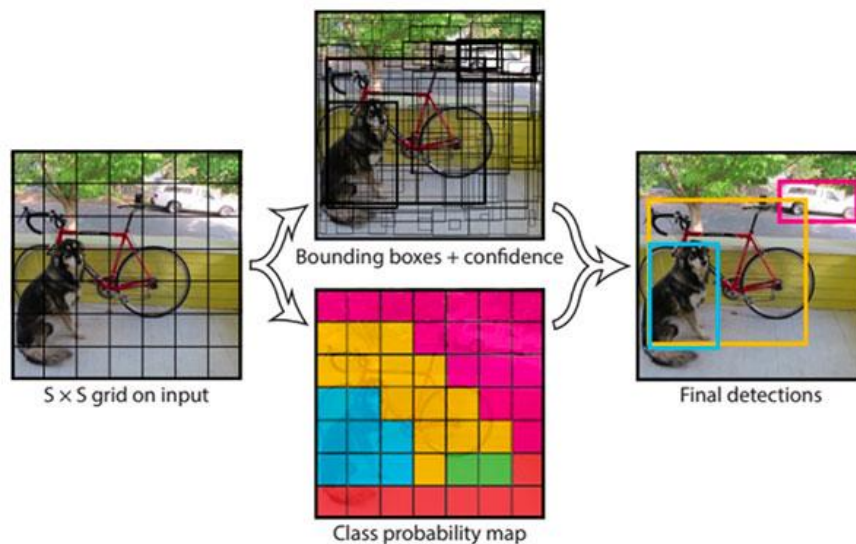
Εικόνα 16: Αρχιτεκτονική δικτύου YOLO
(Πηγή: <https://www.mdpi.com/2076-3417/10/2/612/htm>)

Επομένως, ο πίνακας εξόδου έχει μέγεθος $S \times S \times (B \times (5 + C))$, όπου το S είναι ίδιο με τον αριθμό που διαιρεί την εικόνα εισόδου σε κελιά, B είναι το σύνολο των αντικειμένων που ανιχνεύτηκαν από το εκάστοτε κελί. Ο αριθμός 5 προκύπτει από τη πιθανότητα να βρίσκεται το κέντρο ενός αντικείμενο στο συγκεκριμένο κελί και τις συντεταγμένες του πλαισίου οριοθέτησης, δηλαδή τις συντεταγμένες του κέντρου (x, y) στο συγκεκριμένο κελί, το ύψος και το πλάτος του πλαισίου που επεκτείνονται και σε άλλα κελιά. Τέλος, το C είναι ο αριθμός των κλάσεων των αντικειμένων που δύναται να ανιχνεύσει ο αλγόριθμος. Συνεπώς, ο πίνακας εξόδου του κάθε κελιού έχει την παρακάτω μορφή.

y =	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

Εικόνα 17: Πίνακας εξόδου του εκάστοτε κελιού
(Πηγή: <https://medium.com/machine-learning-bites/deeplearning-series-object-detection-and-localization-yolo-algorithm-r-cnn-71d4dfd07d5f>)

Στην παραπάνω εικόνα το Pc απεικονίζει τη πιθανότητα να βρίσκεται ένα αντικείμενο στο κελί, τα bx, by είναι οι συντεταγμένες του κέντρου στο συγκεκριμένο κελί. Παράλληλα, τα bw και bh είναι το πλάτος και το ύψος του πλαισίου που επεκτείνονται και πέρα του ενός κελιού, ενώ τα C1, C2, C3 είναι η πιθανότητα της κάθε κλάσης με τις τιμές τους να βρίσκονται μεταξύ του 0 και 1.



Εικόνα 18: Εφαρμογή αλγορίθμου YOLO
(Πηγή: <https://arxiv.org/pdf/1506.02640.pdf>)

Για παράδειγμα, ας υποθέσουμε ότι ο αλγόριθμος μπορεί να ανιχνεύσει τρεις κλάσεις αυτοκίνητα (C1), ποδήλατα(C2) και σκυλιά(C3), όπως στο παράδειγμα της παραπάνω εικόνας. Τότε οι πίνακες των κελιών που ευθύνονται για την ανίχνευση και πλαισίωση του σκυλιού έχουν τιμή 1 ως Pc, τα bx, by, bw, bh εξαρτώνται από το εκάστοτε κελί, ενώ τα C1 και C2 έχουν τιμές που πλησιάζουν το 0, σε αντίθεση με το C3 που είναι η κλάση του σκυλιού και έχει τιμή πολύ κοντά στο 1. Αντίστοιχα, οι πίνακες για τα κελιά που ανιχνεύουν το αυτοκίνητο και το ποδήλατο έχουν παρόμοιες τιμές με την διαφορά ότι το C1 έχει τιμή κοντά στο 1 για τα κελιά που ανιχνεύουν αυτοκίνητο και το C2 για τα κελιά που ανιχνεύουν ποδήλατο. Αντίθετα, οι πίνακες των κελιών που δεν αναγνωρίζουν ένα αντικείμενο έχουν τιμή 0 στο Pc και τα υπόλοιπα στοιχεία του πίνακα αναφέρονται ως don't care.

Επιπροσθέτως, υπάρχει πιθανότητα όχι μόνο ένα κελί να θεωρεί πως έχει το κέντρο ενός αντικειμένου αλλά περισσότερα, εξαιτίας της ανάλυσης της εικόνας ή κακής επιλογής τιμής του S . Σε αυτήν την περίπτωση χρησιμοποιείται η μέθοδος του Non-Maximum Suppression, που εξηγείται παραπάνω.

Αξίζει, βέβαια, να σημειωθεί πως αρχικά αν το κέντρο δύο αντικειμένων βρισκόταν σε ένα κελί υπήρχε πρόβλημα, επειδή ο πίνακας εξόδου περιλαμβάνει τις πληροφορίες για ένα μόνο αντικείμενο. Επομένως, το B στον τύπο $S \times S \times (B \times (5+C))$ έχει την τιμή 1. Μία πιθανή λύση θα ήταν να αυξηθεί ο αριθμός των κελιών, δηλαδή η τιμή S , έτσι ώστε να υπάρχει μικρότερη πιθανότητα να βρίσκεται το κέντρο δύο αντικειμένων στο ίδιο κελί. Έτσι όμως, το πρόβλημα απλά περιορίζεται και δεν επιλύεται ολοκληρωτικά, ώστε να μην υπάρχει κανένα περιθώριο λάθους.

Την οριστική λύση στο πρόβλημα δίνει η μέθοδος των πλαισίων αγκύρωσης (anchor boxes), που χρησιμοποιείται από την δεύτερη έκδοση του YOLO. Ουσιαστικά, πρόκειται για προκαθορισμένα πλαίσια οριοθέτησης τα οποία έχουν διαφορετικό μέγεθος το ένα από το άλλο, βρίσκονται σε κάθε κελί και κάθε ένα ευθύνεται για την ανίχνευση των αντικειμένων που το κέντρο τους βρίσκεται στο ίδιο κελί.



Εικόνα 19: Χρήση anchor boxes

(Πηγή: <https://www.codestudyblog.com/sfb20b2/0305131817.html>)

Για παράδειγμα, η παραπάνω εικόνα έχει χωριστεί σε ένα πλέγμα 3×3 με συνολικά 9 κελιά. Όπως παρατηρείται το κέντρο των δύο αντικειμένων βρίσκεται στο ίδιο κελί, οπότε ο αλγόριθμος χωρίς την χρήση των anchor boxes δεν θα λειτουργούσε σωστά. Ας υποθέσουμε ότι οι κλάσεις αναγνώρισης είναι τρεις και πιο συγκεκριμένα είναι τα αυτοκίνητα, οι άνθρωποι και τα φορτηγά και ότι τα anchor boxes είναι δύο για κάθε κελί, με το πρώτο να επεκτείνεται οριζόντια και το άλλο κάθετα. Στην προκειμένη περίπτωση το B του τύπου $S \times S \times (B \times (5+C))$ θα ισούται με 2, αφού δύο είναι τα πλαίσια αγκύρωσης που μπορεί να υπάρχουν για κάθε κελί. Ο πίνακας εξόδου, λοιπόν, του κελιού που περιλαμβάνει το κέντρο των αντικειμένων είναι ο ακόλουθος:

y =	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3
	pc
	bx
	by
	bh
	bw
	c1
	c2
	c3

όπου τα πρώτα οχτώ στοιχεία περιλαμβάνουν τα δεδομένα για την ανίχνευση του αυτοκινήτου, ενώ τα υπόλοιπα για την ανίχνευση του ανθρώπου. Τέλος, σε περίπτωση που βρίσκεται μόνο το κέντρο του αυτοκινήτου στο κελί τότε ο πίνακας εξόδου του θα είχε το ίδιο μέγεθος, με τη διαφορά ότι το Pc της δεύτερης οχτάδας που αντιστοιχεί στο δεύτερο anchor box θα είχε τιμή 0, εφόσον δεν βρέθηκε ένα αντικείμενο που να εμφανίζεται σε κατακόρυφη όψη.

Όπως έχει αναφερθεί, οι διαφορές των εκδόσεων του YOLO δεν βρίσκονται στην μεθοδολογία που ακολουθούν, αλλά στο συνελκτικό νευρωνικό δίκτυο και στις μεθόδους που εφαρμόζονται σε αυτό.

Πιο συγκεκριμένα, στην πρώτη έκδοση του YOLO η εικόνα εισόδου έχει προκαθορισμένο μέγεθος του 448x448x3, όπου το 448 αποτελεί το ύψος και το πλάτος της εικόνας, ενώ το 3 τον αριθμό των καναλιών των χρωμάτων (Red, Green, Blue). Παράλληλα, η εκπαίδευση του αλγορίθμου υλοποιείται με εικόνες μεγέθους 224x224. Επιπρόσθετα, οι ερευνητές χρησιμοποίησαν την τιμή $S = 7$ για την διαίρεση της εικόνας σε ένα πλέγμα 7x7 και $B = 2$, ενώ με την χρήση του Pascal VOC dataset κατά την εκπαίδευση το C, δηλαδή οι κλάσεις, ισοδυναμούσε με 20. Επιπλέον, το συνελκτικό νευρωνικό δίκτυο ήταν εμπνευσμένο από το GoogleNet, με αποτέλεσμα να αποτελείται από 24 συνελκτικά στρώματα (convolution layers), 4 μέγιστης συγκέντρωσης στρώματα (max pooling layers) και 2 πλήρως συνδεδεμένα στρώματα (fully connected layers). Τέλος, η συνάρτηση ενεργοποίησης για όλα τα στρώματα είναι η ReLU, εκτός από το τελευταίο που χρησιμοποιεί την γραμμική (Linear Activation Function). Έτσι λοιπόν, η απόδοση της πρώτης έκδοσης έχει ποσοστό 63.4% mAP στα 45 καρέ ανά δευτερόλεπτο στο VOC 2007 dataset. [21]

Σε αντίθεση με την πρώτη έκδοση, η δεύτερη έκδοση του YOLO χρησιμοποιεί ένα συνελκτικό νευρωνικό δίκτυο με 19 συνελκτικά στρώματα και 5 max pooling στρώματα που ονομάζεται Darknet-19, έναντι των 24 συνελκτικών και 4 max pooling στρωμάτων αντίστοιχα. Όπως έχει αναφερθεί στην δεύτερη έκδοση τα πλαίσια οριοθέτησης των αντικειμένων προβλέπονται με τη χρήση των πλαισίων αγκύρωσης, σε αντίθεση με την πρώτη έκδοση όπου προβλέπονται μέσω των πλήρως συνδεδεμένων στρωμάτων. Επίσης, το μέγεθος της εικόνας εισόδου μειώνεται από 448 σε 416 για την εκτέλεση του αλγορίθμου στο VOC 2007 dataset, με σκοπό να μην υπάρχει περιττός αριθμός στα μεγέθη των χαρτών χαρακτηριστικών που παράγει το συνελκτικό νευρωνικό δίκτυο. Με αυτόν τον τρόπο η απόδοση της δεύτερης έκδοσης έχει ποσοστό 76.8% mAP στα 67 καρέ το δευτερόλεπτο. [22]

Από την άλλη πλευρά, η τρίτη έκδοση του αλγορίθμου YOLO χρησιμοποιεί το συνελκτικό νευρωνικό δίκτυο Darknet-53, αποτελείται από 53 συνελκτικά στρώματα και συνολικά 107 στρώματα, ενώ οι συναρτήσεις ενεργοποίησης αποτελούνται από τις Leaky και ReLU. Τα πλαίσια αγκύρωσης είναι 9 και έχουν τις εξής τιμές για κάθε κελί: (10×13) , (16×30) , (33×23) , (30×61) , (62×45) , (59×119) , (116×90) , (156×198) , (373×326) . Το μέγεθος της εικόνας που εισάγεται στον αλγόριθμο είναι 320×320 , 416×416 και 608×608 . Το ποσοστό απόδοσης για τις εικόνες μεγέθους 320×320 είναι 51.5% mAP στα 45 fps, ενώ για τις εικόνες μεγέθους 416×416 είναι 55.3% mAP στα 34 fps και τέλος για τις εικόνες μεγέθους 608×608 είναι 57.9% mAP στα 19 fps. Ολοκληρώνοντας, αξίζει να αναφερθεί πως τα παραπάνω αποτελέσματα προκύπτουν με την χρήση της κάρτας γραφικών Titan X της Nvidia. [23]

Τέλος, η τέταρτη έκδοση του αλγορίθμου YOLO χρησιμοποιεί το συνελκτικό νευρωνικό δίκτυο CSPDarknet-53, που περιέχει συνολικά 162 στρώματα, και με την βοήθεια των τεχνικών Bag of Frebies, που χρησιμοποιείται κατά την εκπαίδευση, και Bag of Specials, που χρησιμοποιείται κατά την εκτέλεση του αλγορίθμου να οδηγούν σε καλύτερα αποτελέσματα. Ο αριθμός και τα μεγέθη των πλαισίων αγκύρωσης της τέταρτης έκδοσης του αλγορίθμου YOLO είναι ίδια με της τρίτης έκδοσης. Το μέγεθος της εικόνας που εισάγεται στο δίκτυο μπορεί να είναι 320×320 με ποσοστό απόδοσης 60% mAP στα 63fps, 416×416 με ποσοστό απόδοσης 62.8% mAP στα 55 fps, 512×512 με ποσοστό απόδοσης 64.9% mAP στα 45 fps και τέλος 608×608 με ποσοστό απόδοσης 65.7% mAP στα 34 fps. Επιπρόσθετα, αξίζει να αναφερθεί πως σύμφωνα με τους ερευνητές της τέταρτης έκδοσης του YOLO τα παραπάνω αποτελέσματα προκύπτουν με την χρήση της κάρτας γραφικών RTX 2070 της Nvidia. Η διαίρεση της εικόνας εισόδου σε ένα πλέγμα κελιών εξαρτάται από το μέγεθος της εικόνας. Για παράδειγμα, εάν το μέγεθος της εικόνας είναι 416×416 , τότε το πλέγμα κελιών θα είναι 13×13 . Γενικά, όσο πιο μεγάλο ή μικρό είναι το μέγεθος της εικόνας εισόδου τόσο μεγαλύτερο ή μικρότερο αντίστοιχα είναι και το πλέγμα κελιών. Σε επίπεδο εφαρμογής τα μεγέθη της εικόνας εισόδου μπορούν να είναι από 228×228 έως 1024×1024 με μια αύξηση 32×32 ανά μέγεθος, ενώ το πλάτος και ύψος του πλέγματος των κελιών ποικίλει μεταξύ 9 και 32. Στην παρούσα διπλωματική εργασία χρησιμοποιείται η τέταρτη έκδοση του αλγορίθμου YOLO. [24]

2.9 Σχετικές έρευνες καταγραφής αυτοκινήτων και των ταχυτήτων τους σε βίντεο

Επί του παρόντος, η ανίχνευση των οχημάτων με βάση την υπολογιστική όραση σε βίντεο χωρίζεται σε μεθόδους μηχανικής όρασης (machine vision) και σε σύνθετες μεθόδους βαθιάς μάθησης (deep learning). Οι μέθοδοι μηχανικής όρασης χρησιμοποιούν την κίνηση ενός οχήματος για να το διαχωρίσουν από το φόντο. Οι μέθοδοι αυτοί χωρίζονται σε τρεις βασικές κατηγορίες, οι οποίες είναι η μέθοδος της αφαίρεσης φόντου (background subtraction), η μέθοδος που αξιοποιεί την διαφορά στα pixel μεταξύ διαδοχικών frame και η μέθοδος που χρησιμοποιεί το optical flow. Από την άλλη πλευρά, οι μέθοδοι βαθιάς μάθησης συγκροτούνται από τα συνελκτικά νευρωνικά δίκτυα που εξηγούνται παραπάνω και τα οποία έχουν επιφέρει πολλά καινοτόμα επιτεύγματα στον τομέα της αναγνώρισης των οχημάτων. Παρακάτω αναφέρονται διάφορα παραδείγματα ερευνών που έχουν δημοσιευθεί τα τελευταία χρόνια σε θέματα είτε καταγραφής των οχημάτων είτε των ταχυτήτων που αναπτύσσουν σε αστικούς και εθνικούς δρόμους είτε και των δύο.

Το 2015 δημοσιεύθηκε το paper "Fast multiclass vehicle detection on aerial images" από τους Kang Liu και Gellert Mattyus. Οι συγγραφείς χρησιμοποιούν μη επανδρωμένο αεροσκάφος για την καταγραφή των δρόμων. Ο εντοπισμός των πλαισίων οριοθέτησης των οχημάτων πραγματοποιείται με την μέθοδο του συρόμενου παραθύρου (sliding window), ενώ η κατηγοριοποίηση των οχημάτων υλοποιείται με την μέθοδο Histogram of Oriented Gradient (HOG). [25]

Το 2018 δημοσιεύθηκε το paper "Automatic Vehicle Speed Estimation Method for Unmanned Aerial Vehicle Images" από τους Hao Long, Yi-Nung Chung και Jun-de Li. Η συγκεκριμένη μελέτη παρουσιάζει μία λύση στο πρόβλημα της ανίχνευσης της ταχύτητας των οχημάτων σε βίντεο που έχουν καταγραφεί με την βοήθεια μη επανδρωμένων αεροσκαφών. Αρχικά, η μέθοδος HSV (hue, saturation, value) αξιοποιείται για την μείωση της εναλλαγής του φωτισμού και για την αφαίρεση των σκιών που υπάρχουν λόγω του ύψους που καταγράφεται το βίντεο. Η ανίχνευση της κίνησης βασίζεται στην διαφορά των διαδοχικών frame για να διαχωρίσει τα κινούμενα αντικείμενα από το φόντο. Τέλος, με την εύρεση του κέντρου του οχήματος και της απόστασης που διανύει κατά μήκος της εικόνας του βίντεο εκφρασμένη σε pixel εκτιμάται η ταχύτητα που αναπτύσσουν τα οχήματα. [26]

Επίσης, τον ίδιο χρόνο δημοσιεύτηκε το paper "A Video based Vehicle Detection, Counting and Classification System" από τους Sheeraz Memon, Sania Bhatti, Liaquat A. Thebo, Mir Muhammad B. Talpur και Mohsin A. Me. Η αρχή λειτουργίας του βασίζεται στην χρήση του αλγορίθμου αφαίρεσης φόντου (background subtraction) Gaussian Mixture Model (GMM) για την ανίχνευση της κίνησης των οχημάτων και την καταγραφή του συνολικού αριθμού οχημάτων. Η κατηγοριοποίηση των οχημάτων πραγματοποιείται με την βοήθεια της μεθόδου Support Vector Machine (SVM). Τέλος, η καταγραφή των βίντεο πραγματοποιείται από στατικές κάμερες ασφαλείας που βρίσκονται στους δρόμους. [27]

Το 2019 δημοσιεύθηκε το paper "Video-Based Vehicle Counting System for Urban Roads in Nigeria Using Yolo and DCF-CSR Algorithms" από τους Aderonke A. Oni και Nicholas Kajoh. Το περιεχόμενο της παραπάνω δημοσίευσης παρέχει ακριβή δεδομένα καταγραφής της κίνησης των αυτοκινήτων στους αστικούς δρόμους της Νιγηρίας. Ο αλγόριθμος που χρησιμοποιείται για την αναγνώριση των οχημάτων είναι η τρίτη έκδοση του YOLO, ενώ το σύστημα παρακολούθησης συνιστά το DCF-CSR (Discriminative Correlation Filter with Channel and Spatial Reliability). Τέλος, σύμφωνα με τους συγγραφείς ο καλύτερος τρόπος εφαρμογής της παραπάνω μεθόδου είναι από βίντεο που καταγράφουν οι κάμερες ασφαλείας που υπάρχουν στους δρόμους. [28]

Αντίστοιχα τον ίδιο χρόνο δημοσιεύθηκε το paper "Vision-based vehicle detection and counting system using deep learning in highway scenes" από τους Huansheng Song, Haoxiang Liang, Huaiyu Li, Zhe Dai και Xu Yun. Στην συγκεκριμένη έρευνα οι συγγραφείς παρουσίασαν ένα καινούριο dataset, το οποίο απευθύνεται για αναγνώριση των οχημάτων σε εθνικές οδούς, ενώ παράλληλα αξιοποιούν την τρίτη έκδοση του αλγορίθμου YOLO για την αναγνώριση των οχημάτων. Η καταγραφή των βίντεο υλοποιείται από τις κάμερες ασφαλείας που υπάρχουν στις εθνικές οδούς. Επίσης, η αρχή λειτουργίας του βασίζεται στην διαίρεση της εικόνας σε επιμέρους τμήματα. Πιο συγκεκριμένα, αρχικά η επιφάνεια του δρόμου αποσπάται και χωρίζεται σε δύο περιοχές. Στην συνέχεια, κάθε περιοχή στέλνεται στον αλγόριθμο YOLO για την αναγνώριση του τύπου και της τοποθεσίας

των οχημάτων. Τέλος, οι τροχιά (trajectory) του κάθε οχήματος λαμβάνεται από τον αλγόριθμο ORB για την καταμέτρηση των οχημάτων. [29]

Επιπλέον, το 2019 δημοσιεύθηκε το paper "Speed Estimation of Multiple Moving Objects from a Moving UAV Platform" από τους Debojit Biswas, Hongbo Su, Chengyi Wang και Aleksandar Stevanovic. Στην συγκεκριμένη μελέτη, παρουσιάστηκε ένα σύστημα ανίχνευσης της ταχύτητας πολλαπλών αντικειμένων μέσω εναέριας παρακολούθησης από ένα κινούμενο μη επανδρωμένο αεροσκάφος. Πιο αναλυτικά, για την αναγνώριση των αντικειμένων εφαρμόστηκε ο αλγόριθμος Faster R-CNN, ενώ ένα φίλτρο CSRT (channel and spatial reliability tracking) χρησιμοποιήθηκε για την παρακολούθηση των αντικειμένων. Παράλληλα, εφαρμόζεται η μέθοδος της ευθυγράμμισης με βάση τα χαρακτηριστικά της εικόνας για την εύρεση της κατάλληλης τοποθεσίας των αντικειμένων στην εικόνα. Τέλος, εκτελείται η μέθοδος SSIM (structural similarity index measurement) για τον έλεγχο της ομοιότητας μεταξύ του πραγματικού frame και του frame που πραγματοποιείται αναγνώριση των αντικειμένων. Ο έλεγχος αυτός είναι απαραίτητος εξαιτίας της κίνησης του drone και των νέων αντικειμένων που εισέρχονται στο πλάνο που καταγράφεται. [30]

Το 2020 δημοσιεύθηκε το paper "Real-time monitoring of traffic parameters" από τους Kirill Khazukov, Vladimir Shepelev, Tatiana Karpeta, Salavat Shabiev, Ivan Slobodin, Irakli Charbadze και Irina Alferova. Σκοπός της συγκεκριμένης μελέτης είναι η ανάπτυξη ενός συστήματος συλλογής δεδομένων της κυκλοφοριακής κίνησης σε πραγματικό χρόνο από σταθερές κάμερες ασφαλείας, όπως η ένταση της κυκλοφοριακής συμφόρησης, οι κινήσεις των οχημάτων και η ταχύτητα που αναπτύσσουν. Η λύση, που προτείνεται, βασίζεται στην τρίτη έκδοση του αλγορίθμου YOLO και στο σύστημα παρακολούθησης SORT (Simple Online and Real-Time tracking algorithm). Για την αναγνώριση των οχημάτων οι συγγραφείς παρουσίασαν ένα καινούριο dataset που περιλαμβάνει 6000 εικόνες με 4,3 εκατομμύρια οχήματα. Τέλος, για την εύρεση της ταχύτητας των οχημάτων χρησιμοποιήθηκε μία μέθοδος μετατροπής των συντεταγμένων της εικόνας σε γεωγραφικές συντεταγμένες. [31]

Στην παρούσα διπλωματική εργασία χρησιμοποιείται ένα μη επανδρωμένο αεροσκάφος για την καταγραφή των οχημάτων, ενώ για την αναγνώριση των αυτοκινήτων χρησιμοποιήθηκε η τέταρτη έκδοση του αλγορίθμου YOLO. Επιπλέον, τα δεδομένα, που προκύπτουν για το πλαίσιο οριοθέτησης του εκάστοτε αυτοκινήτου, αξιοποιούνται τόσο για την εύρεση του συνόλου των αυτοκινήτων που κινούνται στις περιοχές του BOAK που κατέγραψε το drone, όσο και για την εύρεση των ταχυτήτων που αναπτύσσουν κατά μήκος του. Παράλληλα, το dataset, που χρησιμοποιήθηκε, είναι το COCO της Microsoft που περιλαμβάνει 80 κλάσεις καθημερινών αντικειμένων. Ωστόσο, μόνο με την χρήση του αλγορίθμου YOLO παρατηρήθηκαν διάφορα προβλήματα, τα οποία καταλήγουν σε λανθασμένα αποτελέσματα. Για την επίλυση των προβλημάτων, χρησιμοποιήθηκε μία μέθοδος ανίχνευσης κίνησης σε συνδυασμό με τον αλγόριθμο YOLO για την αναγνώριση των αυτοκινήτων, η οποία εξηγείται στο κεφάλαιο 3.

2.10 Δυναμικές Εξισώσεις Οχημάτων

Το 2010 οι επιστήμονες Mehrdad Ehsani, Yimin Gao και Ali Emadi δημοσίευσαν μία μελέτη [32] κατά την οποία είχαν αναπτύξει τις παρακάτω εξισώσεις, που αποσκοπούν στην εύρεση της κατανάλωσης της ισχύος σε ένα αμιγώς ηλεκτρικό όχημα.

$$F_{tr} = F_i + F_{ad} + F_{roll} + F_g \quad (1)$$

$$F_{ad} = 0.5 \times \rho \times A_f \times C_d \times (V - V_m)^2 \quad (2)$$

$$F_{roll} = C_{rr} \times m_v \times \cos(\theta) \times g \quad (3)$$

$$F_g = m_v \times g \times \sin(\theta) \quad (4)$$

$$F_i = m_v \times a \quad (5)$$

όπου το F_{tr} δηλώνει την ελκτική δύναμη (Tractive Force), το F_i την αδρανειακή δύναμη (Inertial Force), το F_{ad} την αεροδυναμική δύναμη (Aerodynamic Drag Force), το F_{roll} την δύναμη της αντίστασης κύλισης (Rolling Resistance Force) και το F_g την δύναμη του βάρους του οχήματος σε σχέση με την κλίση του οδοστρώματος (Grading Resistance Force) αντίστοιχα.

Στην εύρεση της αεροδυναμικής δύναμης το ρ αποτελεί την πυκνότητα του αέρα (air density), η οποία εξαρτάται από την θερμοκρασία του περιβάλλοντος όπως φαίνεται από τον παρακάτω πίνακα.

Θερμοκρασία (°C)	Πυκνότητα του αέρα (kg/m ³)
0	1.292
5	1.268
10	1.246
15	1.225
20	1.204
25	1.184
30	1.164
40	1.127

Πίνακας 1: Σχέση μεταξύ πυκνότητας του αέρα και θερμοκρασίας περιβάλλοντος (Πηγή: https://en.wikipedia.org/wiki/Density_of_air)

Στην συνέχεια, το A_f αποτελεί την μπροστινή περιφέρεια του οχήματος εκφρασμένη σε m², ενώ το C_d τον συντελεστή οπισθέλκουσας του αέρα (air drag coefficient), που έχει το όχημα. Παράλληλα, το V είναι η ταχύτητα του οχήματος εκφρασμένη σε m/s, ενώ το V_m είναι η ταχύτητα του αέρα εκφρασμένη και αυτή σε m/s, που είτε ωθεί το όχημα προς την κατεύθυνση που κινείται, οπότε έχει αρνητικό πρόσημο, είτε του αντιστέκεται, οπότε έχει θετικό πρόσημο.

Από την άλλη πλευρά, στην εύρεση της δύναμης της αντίστασης της κύλισης το C_{rr} αποτελεί τον συντελεστή αντίστασης κύλισης (Road Rolling Resistance Coefficient) του οχήματος, ο οποίος εξαρτάται από το μέρος που κινείται το όχημα, αλλά και από τον τύπο των ελαστικών που χρησιμοποιεί. Στον παρακάτω πίνακα αναγράφονται οι τιμές που μπορεί να έχει ο συντελεστής αντίστασης κύλισης αναλόγως τον τύπο του ελαστικού που χρησιμοποιεί το εκάστοτε αυτοκίνητο και το είδος της επιφάνειας που κινείται.

Συντελεστής αντίστασης κύλισης	Επιφάνεια
0.01-0.015	Καινούρια άσφαλτο

0.02	Δρόμος με χαλίκια ή παλιά άσφαλτος
0.04 - 0.08	Συμπαγή άμμος
0.2 - 0.4	Μη συμπαγή άμμος

Πίνακας 2: Τιμές συντελεστή αντίστασης κύλισης

(Πηγή: https://www.engineeringtoolbox.com/rolling-friction-resistance-d_1303.html?fbclid=IwAR3TbIDo5nJwYhFH1jrQP482KnBJRvm0K0fslXtonD0LTIwasfIsxsLRyJY)

Ακολουθως, το m_v είναι το βάρος του οχήματος εκφρασμένο σε κιλά, το g η επιτάχυνση της βαρύτητας και το θ είναι η κλίση του οδοστρώματος, πάνω στο οποίο κινείται το όχημα.

Τα παραπάνω τρία στοιχεία χρησιμοποιούνται και για την εύρεση της δύναμης του βάρους του οχήματος σε σχέση με την κλίση του οδοστρώματος. Τέλος, το a στην εξίσωση της αδρανειακής δύναμης αποτελεί την επιτάχυνση του οχήματος εκφρασμένη σε m/s^2 .

Έτσι, το άθροισμα των παραπάνω δυνάμεων ισοδυναμεί με την ελκτική δύναμη από την οποία υπολογίζεται η ισχύς που καταναλώνεται από ένα όχημα με μία δεδομένη ταχύτητα. Πιο συγκεκριμένα, το γινόμενο της ελκτικής δύναμης με την ταχύτητα του οχήματος εκφρασμένη σε m/s ισούται με την ισχύ που καταναλώνεται, όπως φαίνεται και από την παρακάτω εξίσωση.

$$P = F_{tr} \cdot V \quad (6)$$

Γνωρίζοντας, λοιπόν, την ισχύ που καταναλώνεται για μία ταχύτητα και τον χρόνο που απαιτείται για να φτάσει ένα αυτοκίνητο στον προορισμό του, δεδομένου της ταχύτητας που έχει, υπολογίζεται το ποσοστό φόρτισης της μπαταρίας που απομένει στο αυτοκίνητο.

Κεφάλαιο 3^ο

3.1 Εισαγωγή

Σε αυτό το κεφάλαιο αναλύονται τα μέσα και η μεθοδολογία, που εφαρμόστηκαν για την εύρεση των ταχυτήτων των αυτοκινήτων στον Βόρειο Οδικό Άξονα της Κρήτης. Πιο συγκεκριμένα, αρχικά εξηγείται η γενικότερη χρήση των μη επανδρωμένων αεροσκαφών, καθώς και το συγκεκριμένο μοντέλο που χρησιμοποιήθηκε για την καταγραφή του συνόλου των αυτοκινήτων που διέρχονται από τον BOAK και της ταχύτητας που αναπτύσσουν κατά μήκος αυτού. Στην συνέχεια, αναφέρεται η γλώσσα προγραμματισμού και η βιβλιοθήκη, που επιλέχθηκε για την ανάλυση των βίντεο που κατέγραψε το μη επανδρωμένο αεροσκάφος. Τέλος, αναπτύσσεται η μεθοδολογία βάση της οποίας εκτιμάται η ταχύτητα με την οποία κινούνται τα αυτοκίνητα κατά μήκος του BOAK.

3.2 Μη επανδρωμένα αεροσκάφη

3.2.1 Ιστορική αναδρομή

Μη επανδρωμένο αεροσκάφος (Unmanned Aerial Vehicle - UAV ή drone) ονομάζεται το κάθε είδους ιπτάμενο όχημα το οποίο δεν έχει πλήρωμα στην άτρακτό του και είτε χειρίζεται από απόσταση είτε πραγματοποιεί κάποιο προκαθορισμένο σχέδιο πτήσης χωρίς ανθρώπινη διαχείριση. Χρονολογείται πως τα πρώτα UAV αναπτύχθηκαν για στρατιωτικούς σκοπούς κατά τον πρώτο παγκόσμιο πόλεμο από την Αγγλία και τις ΗΠΑ, όμως παρά τις επιτυχείς δοκιμές δεν χρησιμοποιήθηκαν στη διάρκεια του πολέμου.

Κατά την διάρκεια του μεσοπολέμου οι έρευνες και οι δοκιμές συνεχίστηκαν με αποτέλεσμα να παραχθούν UAV που χρησιμοποιήθηκαν στον στρατό ως εναέριοι στόχοι για την εκπαίδευση των στρατιωτών κατά την διάρκεια του δεύτερου παγκοσμίου πολέμου. Η πρώτη πολυάριθμη καταγεγραμμένη χρήση drone χρονολογείται στον πόλεμο του Βιετνάμ ως ένα μέσο καταγραφής των περιοχών από τις ΗΠΑ χωρίς την απειλή απώλειας ζωής σε περίπτωση ρίψης αεροσκάφους με προσωπικό. Παράλληλα, πολλές χώρες ακολούθησαν το παράδειγμα των ΗΠΑ και της Αγγλίας αναπτύσσοντας τα δικά τους μη επανδρωμένα αεροσκάφη.

Με την πάροδο όμως των χρόνων η τεχνολογία των drone κατάφερε να εμπορευματοποιηθεί και να χρησιμοποιηθεί στην έρευνα, στην καταγραφή κλιματικών αλλαγών, στην γεωργία, στην φωτογραφία καθώς και σε πολλούς άλλους κλάδους της καθημερινότητας. Παρόλα αυτά, η κατασκευή ενός drone που προορίζεται για στρατιωτικούς σκοπούς δεν έχει καμία σχέση στην κατασκευή και στην μορφολογία του με το drone που χρησιμοποιείται για τους παραπάνω σκοπούς. Τα πρώτα έχουν συνήθως σταθερές πτέρυγες και μοιάζουν με μικρά αεροπλάνα, ενώ τα δεύτερα αποτελούνται από έλικες. Ο λόγος που υπάρχει τέτοια μεγάλη διαφορά στην μορφολογία είναι η ταχύτητα που πρέπει να έχει και η ικανότητα του να μεταφέρει βαρύ οπλισμό ένα UAV στρατιωτικής χρήσης. Από την άλλη πλευρά, τα UAV που χρησιμοποιούνται στην καθημερινότητά μας αποτελούνται από μία ή παραπάνω κάμερες και από διάφορους αισθητήρες αναλόγως την χρήση. [33]



Εικόνα 20: Στρατιωτικό μη επανδρωμένο αεροσκάφος
(Πηγή: <https://www.nextgov.com/cio-briefing/2015/04/heres-how-you-hack-drone/111229/>)

3.2.2 DJI Phantom 4 Pro 2.0

Το drone που χρησιμοποιήθηκε για την καταγραφή των αυτοκινήτων ονομάζεται Phantom 4 Pro 2.0 της εταιρίας DJI. Πρόκειται για μοντέλο του 2018, το οποίο αποτελείται από 4 έλικες (quadcopter) με το βάρος του να μην ξεπερνάει τα 1,3 κιλά συμπεριλαμβανομένου της μπαταρίας και των ελίκων. [34]

Ο τύπος της επαναφορτιζόμενης μπαταρίας, που χρησιμοποιεί, είναι LiPo 4S χωρητικότητας 5870mAh και τάσης 15.7V. Η ενέργεια που προσφέρει στο μη επανδρωμένο είναι 89.2Wh, ενώ το βάρος της είναι 468 γραμμάρια. Τα χαρακτηριστικά αυτής της μπαταρίας αποδίδουν περίπου 30 λεπτά μέγιστη ώρα πτήσης με δυνατότητα καταγραφής βίντεο.

Ο χειρισμός του drone γίνεται μέσω τηλεχειριστηρίου με συχνότητα λειτουργίας στα 2400-2438GHz και 5725-5850GHz. Η μπαταρία που χρησιμοποιεί είναι τύπου LiPo 2S των 6000mAh, ενώ το ίδιο λειτουργεί στα 7.4V με κατανάλωση ρεύματος στα 1.2A. Τέλος, η μέγιστη απόσταση μετάδοσης χωρίς εμπόδια είναι τα 6 χιλιόμετρα.

Η κάμερα που μεταφέρει το drone διαθέτει έναν αισθητήρα τύπου CMOS της μίας ίντσας, με δυνατότητα λήψης φωτογραφιών στα 20MPixels. Ο φακός της κάμερας κυμαίνεται από τα 8.8mm έως τα 24mm με διάφραγμα από f/2.8 μέχρι f/11. Το εύρος του ISO ποικίλει τόσο στη καταγραφή βίντεο όσο και στην λήψη φωτογραφιών με τις ταχύτητες του μηχανικού κλείστρου να βρίσκονται στα 8-1/2000 second, ενώ του ηλεκτρονικού κλείστρου στα 8-1/8000 second. Επίσης, η ανάλυση των βίντεο, που καταγράφονται, πραγματοποιείται σε HD (1280x720), FHD (1920x1080), 2.7K (2720x1530) και 4K (3840x2160) στα 25, 30 και 60 καρέ ανά δευτερόλεπτο. Η ικανότητα καταγραφής βίντεο στα 60 καρέ ανά δευτερόλεπτο αποτελεί ένα πολύ σημαντικό προσόν για μια κάμερα που χρησιμοποιείται στην παρούσα διπλωματική εργασία, γιατί ο υπολογισμός της ταχύτητας των αυτοκινήτων πραγματοποιείται με μεγαλύτερη ακρίβεια. Παράλληλα, η δυνατότητα καταγραφής βίντεο σε τόσο υψηλές αναλύσεις, όπως η 4K διευκολύνει τον αλγόριθμο στην αναγνώριση όλων των οχημάτων που κινούνται στον BOAK.

Επιπλέον, ο συνδυασμός των ηλεκτρικών κινητήρων με τους έλικες που διαθέτει, δίνουν τη δυνατότητα να αναπτύσσει οριζόντιες ταχύτητες έως 72 km/h. Παράλληλα, η μέγιστη ταχύτητα ανύψωσης δεν ξεπερνάει τα 6 m/s, ενώ η μέγιστη ταχύτητα καθόδου τα 4 m/s. Συμπληρωματικά, η μέγιστη ταχύτητα του αέρα στην οποία δύναται να προβάλλει αντίσταση φτάνει μέχρι τα 10 m/s, που σημαίνει ότι αντιστέκεται σε ανέμους εντάσεως μέχρι και 5 μποφόρ, με αποτέλεσμα τα βίντεο που καταγράφονται να φαίνονται πολύ σταθερά.

Τέλος, το εύρος της θερμοκρασίας που επιτρέπει την σωστή λειτουργία του drone είναι μεταξύ των 0°C και 40°C.



Εικόνα 21: DJI Phantom 4 Pro 2.0
(Πηγή: <https://www.dji.com/gr/phantom-4-pro-v2>)

3.3 Εργαλεία

3.3.1 PYTHON

Η γλώσσα προγραμματισμού που επιλέχθηκε για την επίλυση του προβλήματος της ανίχνευσης και καταγραφής του συνόλου των αυτοκινήτων, καθώς και των ταχυτήτων που αναπτύσσουν, είναι η Python. Η Python αναπτύχθηκε από τον Ολλανδό Guido van Rossum και δημοσιεύτηκε για πρώτη φορά το 1991. Ουσιαστικά, αποτελεί μία αντικειμενοστραφή και υψηλού επιπέδου γλώσσα προγραμματισμού, που διαφέρει από τις υπόλοιπες γλώσσες προγραμματισμού που υπάρχουν.

Αρχικά, η Python χαρακτηρίζεται απλή και χωρίς πολλές συντακτικές ιδιαιτερότητες διευκολύνοντας με αυτόν τον τρόπο την κατανόησή των αλγορίθμων που αναπτύσσονται. Χαρακτηριστικό είναι ότι το συντακτικό της επιτρέπει την έκφραση εννοιών σε λιγότερες γραμμές κώδικα από ότι θα ήταν δυνατό σε άλλες γλώσσες προγραμματισμού όπως η C++ ή η Java. Αυτό επιτυγχάνεται με την χρήση των πολυάριθμων βιβλιοθηκών που διαθέτει. Κάθε βιβλιοθήκη περιλαμβάνει έναν μεγάλο αριθμό από συναρτήσεις και μεθόδους, που μπορούν να χρησιμοποιηθούν και να μειώσουν το σύνολο των γραμμών του κώδικα. Επιπλέον, αξίζει να σημειωθεί ότι η Python αναπτύσσεται ως ανοιχτό λογισμικό (open source) και η διαχείρισή της ελέγχεται από τον μη κερδοσκοπικό οργανισμό Python Software Foundation. Ο

κώδικας διανέμεται με την άδεια Python Software Foundation License, η οποία είναι συμβατή με την GPL (General Public License).

Επιπλέον, αξίζει να αναφερθεί πως η αποθήκευση των δεδομένων σε έναν κώδικα Python γίνεται δυναμικά, δηλαδή δεν χρειάζεται η αναφορά των τύπων των δεδομένων κατά την δήλωση της μεταβλητής. Ολοκληρώνοντας, οι διερμηνευτές της Python είναι διαθέσιμοι για εγκατάσταση σε όλα τα γνωστά λειτουργικά συστήματα, όπως Windows, Linux και MacOS, επιτρέποντας στην Python την εκτέλεση του κώδικα σε όλα τα παραπάνω χωρίς να υπάρχουν αλλαγές στα αποτελέσματα που παράγει. [35]



Εικόνα 22: Το λογότυπο της Python
(Πηγή: <https://el.wikipedia.org/wiki/Python>)

3.3.2 Βιβλιοθήκη OPENCV

Τα τελευταία χρόνια η ανάπτυξη της επιστήμης της μηχανικής μάθησης και της υπολογιστικής όρασης ώθησε τους προγραμματιστές στην δημιουργία βιβλιοθηκών, που βοηθούν στην επίλυση των προβλημάτων. Η βιβλιοθήκη Opencv (Open Source Computer Vision Library) αποτελεί την κύρια επιλογή τόσο για προβλήματα υπολογιστικής όρασης, όσο και για προβλήματα μηχανικής μάθησης. Αποτελεί βιβλιοθήκη ανοιχτού κώδικα και κυκλοφόρησε για πρώτη φορά το 1999 από την εταιρία Intel και έχει γραφτεί στη γλώσσα προγραμματισμού C++, γεγονός που την καθιστά ικανή να εκμεταλλευτεί την πολυπύρηνη επεξεργασία από τον επεξεργαστή του υπολογιστή. Επιπρόσθετα, η BSD άδεια της Opencv διευκολύνει τις εταιρίες προγραμματισμού και κατ' επέκταση όλους τους προγραμματιστές στην χρήση και τροποποίηση του κώδικα. Η βιβλιοθήκη διαθέτει περισσότερους από 2500 βελτιστοποιημένους αλγορίθμους, οι οποίοι περιλαμβάνουν ένα ολοκληρωμένο σύνολο τόσο κλασικών όσο και σύγχρονων αλγορίθμων υπολογιστικής όρασης και μηχανικής μάθησης. Παραδείγματα των αλγορίθμων αυτών αποτελούν οι εφαρμογές αναγνώρισης προσώπων, η ανίχνευση αντικειμένων, η ανίχνευση κίνησης σε βίντεο και πολλά άλλα. Όπως έχει αναφερθεί η βιβλιοθήκη έχει γραφτεί στην γλώσσα προγραμματισμού C++, αλλά μπορεί να αξιοποιηθεί και από άλλες γλώσσες προγραμματισμού όπως η Python, η Java και η Matlab, ενώ παράλληλα υποστηρίζεται από όλα τα διαθέσιμα λειτουργικά συστήματα όπως Windows, Linux, MacOS και Android. Τέλος, αξίζει να σημειωθεί πως έχουν αναπτυχθεί οι διεπαφές CUDA και OPENCL, με σκοπό ο κώδικας να εκτελείται μέσω της κάρτας γραφικών

(GPU), δεδομένου ότι η κάρτα γραφικών είναι πολύ πιο γρήγορη από έναν επεξεργαστή. [36]



Εικόνα 23: Το λογότυπο της βιβλιοθήκης OpenCV
(Πηγή: <https://en.wikipedia.org/wiki/OpenCV>)

3.3.3 CUDA

Τον Νοέμβριο του 2006 η εταιρία NVIDIA παρουσίασε το εργαλείο CUDA. Πρόκειται για μια γενικής χρήσης πλατφόρμα, που αξιοποιεί την παράλληλη υπολογιστική μηχανή των καρτών γραφικών (GPUs) της NVIDIA με σκοπό την επίλυση πολλών σύνθετων υπολογιστικών προβλημάτων με έναν πιο αποτελεσματικό τρόπο από ένα κεντρικό επεξεργαστή (CPU).

Ο λόγος για την ανάπτυξη του εργαλείου CUDA είναι η μεγάλη διαφορά της απόδοσης και διαχείρισης μνήμης της κάρτας γραφικών σε σχέση με έναν επεξεργαστή, χωρίς να παρουσιάζεται μια ουσιαστική διαφορά στο κόστος και στην ισχύ που χρειάζονται. Πολλές εφαρμογές που αναπτύσσονται, αξιοποιούν τις υψηλότερες δυνατότητες αυτές για να εκτελούνται σε μία κάρτα γραφικών έναντι σε έναν επεξεργαστή. Χαρακτηριστικά παραδείγματα αποτελούν ο γραφικός σχεδιασμός (graphic design), η νευροεπιστήμη, η ανάπτυξη παιχνιδιών (game development), η μηχανική μάθηση και η τεχνητή νοημοσύνη.

Οι κύριες διαφορές στις δυνατότητες μεταξύ μίας κάρτας γραφικών και ενός κεντρικού επεξεργαστή βρίσκονται στον τρόπο που έχει σχεδιαστεί το κάθε ένα. Από την μία πλευρά, ένας κεντρικός επεξεργαστής έχει σχεδιαστεί για την εκτέλεση μίας ακολουθίας λειτουργιών, που ονομάζονται threads, όσο το δυνατόν πιο γρήγορα με την δυνατότητα εκτέλεσης κάποιων δεκάδων από αυτές σε παραλληλία. Αντίθετα, οι κάρτες γραφικών έχουν σχεδιαστεί ώστε οι λειτουργίες αυτές να μην εκτελούνται κατά δεκάδες σε παραλληλία, αλλά κατά χιλιάδες. Ουσιαστικά, οι κάρτες γραφικών εξειδικεύονται σε πολλούς παράλληλους υπολογισμούς, για αυτό και έχουν σχεδιαστεί έτσι ώστε περισσότερα τρανζίστορ να ασχολούνται με την επεξεργασία δεδομένων και όχι στην προσωρινή αποθήκευση αυτών (caching) όπως πραγματοποιούν οι κεντρικοί επεξεργαστές.

Στην παρούσα διπλωματική εργασία η εκτέλεση του αλγορίθμου YOLO υλοποιείται από την κάρτα γραφικών NVIDIA GTX 1060 3GB με την αξιοποίηση του εργαλείου CUDA για τους λόγους που αναφέρθηκαν παραπάνω. [37]

3.4 Μεθοδολογία ανίχνευσης ταχύτητας και συνόλου αυτοκινήτων

Για την ανίχνευση των οχημάτων που διασχίζουν τον Βόρειο Οδικό Άξονα της Κρήτης και για την καταγραφή της ταχύτητάς των αυτοκινήτων επιλέχθηκε η γλώσσα προγραμματισμού Python, ενώ ο αλγόριθμος που αναγνωρίζει τα οχήματα είναι η τέταρτη έκδοση του YOLO (Yolov4).

Τα αρχεία που απαιτούνται για την εκτέλεση του YOLO είναι τα coco.names, yolov4.cfg και yolov4.weights.

Το αρχείο coco.names περιλαμβάνει τα ονόματα των 80 κλάσεων, που δύναται να ανιχνεύσει ο αλγόριθμος YOLO, τα οποία και απεικονίζονται παρακάτω. [38]

1	person	21	elephant	41	wine glass	61	diningtable
2	bicycle	22	bear	42	cup	62	toilet
3	car	23	zebra	43	fork	63	tvmonitor
4	motorbike	24	giraffe	44	knife	64	laptop
5	aeroplane	25	backpack	45	spoon	65	mouse
6	bus	26	umbrella	46	bowl	66	remote
7	train	27	handbag	47	banana	67	keyboard
8	truck	28	tie	48	apple	68	cell phone
9	boat	29	suitcase	49	sandwich	69	microwave
10	traffic light	30	frisbee	50	orange	70	oven
11	fire hydrant	31	skis	51	broccoli	71	toaster
12	stop sign	32	snowboard	52	carrot	72	sink
13	parking meter	33	sports ball	53	hot dog	73	refrigerator
14	bench	34	kite	54	pizza	74	book
15	bird	35	baseball bat	55	donut	75	clock
16	cat	36	baseball glove	56	cake	76	vase
17	dog	37	skateboard	57	chair	77	scissors
18	horse	38	surfboard	58	sofa	78	teddy bear
19	sheep	39	tennis racket	59	pottedplant	79	hair drier
20	cow	40	bottle	60	bed	80	toothbrush

Εικόνα 24: Κλάσεις του αρχείου coco.names

Από την άλλη πλευρά, το αρχείο yolov4.cfg περιέχει τις ρυθμίσεις των παραμέτρων της τέταρτης έκδοσης του YOLO. Πιο συγκεκριμένα, η μεταβλητή batch αναφέρεται στον αριθμό των εικόνων που επεξεργάζονται σε κάθε επανάληψη. Το subdivisions αποτελεί τον αριθμό των μικρών επαναλήψεων σε μια επανάληψη (batch), με σκοπό η κάρτα γραφικών να επεξεργαστεί τα δείγματα των μικρών επαναλήψεων μία φορά. Το width και το height αποτελούν τα μεγέθη του πλάτους και του ύψους που πρέπει να μεταβληθούν για την εικόνα εισόδου κατά την διάρκεια της εκπαίδευσης και εκτέλεσης του αλγορίθμου. Η παράμετρος channels αποτελεί τον αριθμό των καναλιών της εικόνας εισόδου που πρέπει να μεταβληθεί κατά την διάρκεια εκπαίδευσης και εκτέλεσης του αλγορίθμου. [39]

Κατά την διάρκεια της εκπαίδευσης του αλγορίθμου χρησιμοποιούνται οι παράμετροι learning_rate, momentum, decay, saturation, exposure και hue. Οι παράμετροι momentum και decay χρησιμοποιούνται για την βελτιστοποίηση του αρχείου των βαρών, που χρησιμοποιείται κατά την εκτέλεση του αλγορίθμου. Αντίθετα, οι μεταβλητές saturation, exposure και hue μεταβάλλουν τυχαία τα χαρακτηριστικά των χρωμάτων της εικόνας εισόδου για βέλτιστα αποτελέσματα, ενώ η μεταβλητή learning_rate αποτελεί την παράμετρο εκμάθησης. [40]

Στη δομή των συνελκτικών νευρωνικών δικτύων περιλαμβάνεται η μεταβλητή batch-normalization, η οποία ενεργοποιεί ή απενεργοποιεί το batch-normalization, το οποίο αποτελεί μια μέθοδο επιτάχυνσης του νευρωνικού δικτύου μέσω της

κανονικοποίησης των στρωμάτων εισόδου. Για τον έλεγχο των φίλτρων, που εφαρμόζονται από το συνελκτικό νευρωνικό δίκτυο, υπάρχουν οι παράμετροι `filters` και `size`, που δηλώνουν τον αριθμό και το μέγεθος των φίλτρων αντίστοιχα. Η μεταβλητή `stride` δηλώνει τον αριθμό των βημάτων που μετατοπίζονται τα φίλτρα στην εικόνα εισόδου, με σκοπό την ολοκληρωτική κάλυψη της εικόνας. Ολοκληρώνοντας, η παράμετρος `pad` αποτελεί το σύνολο των γραμμών και στηλών που περιβάλλουν την εικόνα (`Padding`).

Στις τελευταίες γραμμές του αρχείου `yolov4.cfg` βρίσκονται διάφορα χαρακτηριστικά του αλγορίθμου Yolov4. Πιο συγκεκριμένα, αρχικά αναφέρεται το σύνολο και τα μεγέθη των πλαισίων αγκύρωσης, που χρησιμοποιούνται από τον αλγόριθμο από τις μεταβλητές `num` και `anchors` αντίστοιχα, ενώ η μεταβλητή `class` δηλώνει το σύνολο των κλάσεων, που ανιχνεύονται από τον αλγόριθμο.

Ακολουθούν μερικά παραδείγματα από το αρχείο `yolov4.cfg`.

```
[net]
batch=64
subdivisions=8
# Training
#width=512
#height=512
width=608
height=608
channels=3
momentum=0.949
decay=0.0005
angle=0
saturation = 1.5
exposure = 1.5
hue=.1

[convolutional]
batch_normalize=1
filters=32
size=3
stride=1
pad=1
activation=mish

[convolutional]
batch_normalize=1
filters=64
size=3
stride=2
pad=1
activation=mish
```

```
[yolo]
mask = 6,7,8
anchors = 12, 16, 19, 36, 40, 28, 36, 75, 76, 55, 72, 146, 142, 110, 192, 243, 459, 401
classes=80
num=9
jitter=.3
ignore_thresh = .7
truth_thresh = 1
random=1
scale_x_y = 1.05
iou_thresh=0.213
cls_normalizer=1.0
iou_normalizer=0.07
iou_loss=ciou
nms_kind=greedynms
beta_nms=0.6
```

Εικόνα 25: Παραδείγματα του αρχείου `yolov4.cfg`

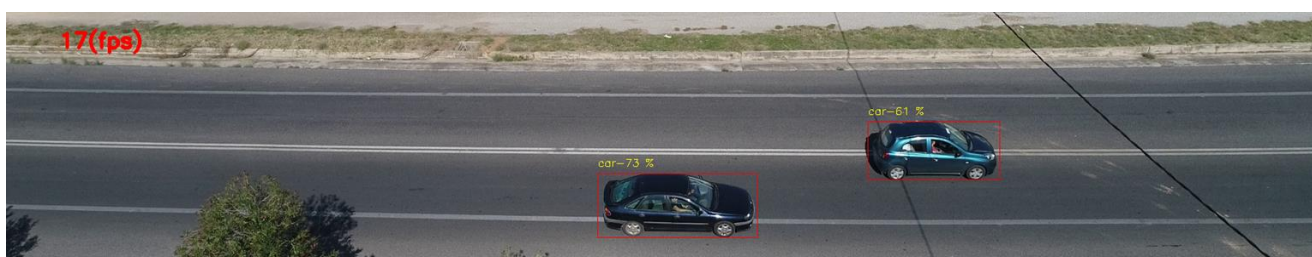
Τέλος, το αρχείο `yolov4.weights` περιλαμβάνει τα εκπαιδευμένα βάρη, τα οποία αξιοποιεί το συνελκτικό νευρωνικό δίκτυο για την ανίχνευση των επιθυμητών αντικειμένων με το συνολικό του μέγεθός να είναι 245.7 MB. Τα παραπάνω τρία αρχεία αξιοποιούνται από μεθόδους της βιβλιοθήκης `OpenCV` για την αρχικοποίηση του δικτύου.

Επιπρόσθετα, με την βοήθεια διαφορετικών μεθόδων της βιβλιοθήκης `OpenCV`

δίνεται εντολή για την ενεργοποίηση του εργαλείου Cuda, με σκοπό ο αλγόριθμος YOLO να εκτελείται στην κάρτα γραφικών και όχι στον κεντρικό επεξεργαστή. Η παραπάνω διαδικασία έχει ως αποτέλεσμα την ταχύτερη εκτέλεση του αλγορίθμου, όπως έχει αναλυθεί στο κεφάλαιο 3.3.3.

Είναι ευρέως γνωστό πως ένα βίντεο αποτελείται από μία συνεχή ακολουθία εικόνων. Η ανίχνευση των αντικειμένων και κατ' επέκταση η εκτέλεση του αλγορίθμου YOLO σε ένα βίντεο πραγματοποιείται σε κάθε εικόνα, που υπάρχει.

Αναλυτικότερα, κάθε καρέ εικόνων μετατρέπεται σε ένα blob (Binary Large Object), το οποίο αποτελεί τη συλλογή των δυαδικών δεδομένων της εικόνας αποθηκευμένα σε μία οντότητα, και εισάγεται στο δίκτυο, που έχει δημιουργηθεί. Το μέγεθος της εικόνας, που εισάγεται στο δίκτυο, επηρεάζει, τόσο την ταχύτητα εκτέλεσης του αλγορίθμου, όσο και την απόδοση της πρόβλεψης των αντικειμένων που υπάρχουν σε κάθε εικόνα. Η τέταρτη έκδοση του αλγορίθμου YOLO επιτρέπει την είσοδο εικόνων με ανάλυση 320x320, 416x416, 512x512 και 608x608. Ακολουθούν στιγμιότυπα, τα οποία απεικονίζουν την εκτέλεση του αλγορίθμου με είσοδο τα μεγέθη που αναφέρονται παραπάνω.



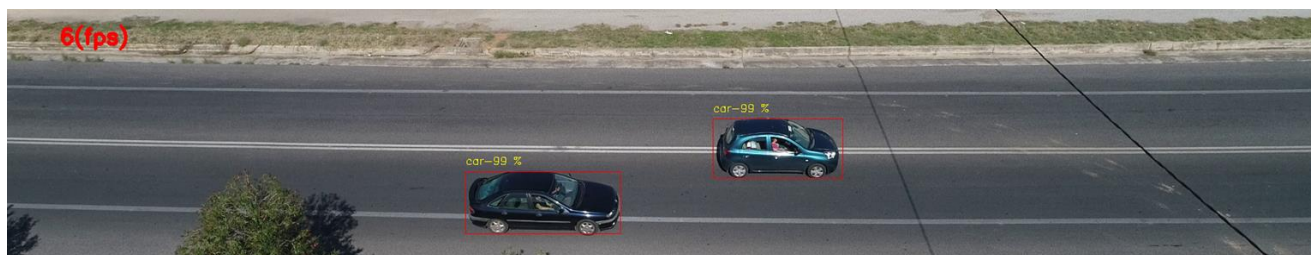
Εικόνα 26: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 320x320



Εικόνα 27: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 416x416



Εικόνα 28: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 512x512



Εικόνα 29: Εφαρμογή αλγορίθμου YOLO με ανάλυση εικόνας 608x608

Ανάλυση εικόνας	Πιθανότητα Πρόβλεψης	Ταχύτητα
320x320	61%, 73%	17(fps)
416x416	92%, 83%	13(fps)
512x512	88%, 95%	9(fps)
608x608	99%, 99%	6(fps)

Πίνακας 3: Αποτελέσματα εκτέλεσης του αλγορίθμου με διαφορετική ανάλυση της εικόνας εισόδου

Παρατηρείται πως με την αύξηση της ανάλυσης της εικόνας εισόδου η ταχύτητα μειώνεται σημαντικά, ενώ η απόδοση της ανίχνευσης των αυτοκινήτων αυξάνεται. Η επιλογή της κατάλληλης ανάλυσης της εικόνας εξαρτάται από την απόδοση του αλγορίθμου σε κάθε μία από αυτές. Στο συγκεκριμένο βίντεο η επιλογή της ανάλυσης 416x416 αποδίδει ικανοποιητικά αποτελέσματα.

Στην συνέχεια, πραγματοποιείται η διαδικασία της προώθησης προς τα τελευταία στρώματα του νευρωνικού δικτύου, όπου βρίσκονται οι πιθανότητες πρόβλεψης της κάθε κλάσης και τα δεδομένα για την κατασκευή των πλαισίων οριοθέτησης. Για την απαλοιφή των κλάσεων με μικρή πιθανότητα πρόβλεψης τίθεται η παράμετρος της ελάχιστης πιθανότητας στη τιμή 50%. Με την εφαρμογή της παραμέτρου αφαιρούνται τα δεδομένα της κάθε κλάσης, που εμφανίζουν πιθανότητα πρόβλεψης μικρότερη του 50%, από την λίστα ανιχνεύσεων.

Είναι γεγονός πως ο αλγόριθμος YOLO προβλέπει την ύπαρξη ενός αντικειμένου παραπάνω από μία φορά, με αποτέλεσμα να αποθηκεύονται και τα δεδομένα πλαισίωσης του ίδιου αντικειμένου πολλαπλές φορές. Για την επίλυση του προβλήματος χρησιμοποιείται η μέθοδος Non-Maximum Suppression της βιβλιοθήκης OpenCV. Αποτέλεσμα των παραπάνω είναι η δημιουργία δύο πινάκων, που περιέχουν τα ονόματα των κλάσεων και τα δεδομένα για τα πλαίσια οριοθέτησης του εκάστοτε αντικειμένου αντίστοιχα. Οι πίνακες αξιοποιούνται στην σχεδίαση των πλαισίων οριοθέτησης για κάθε αντικείμενο που ανιχνεύεται από τον αλγόριθμο.

Για την καταγραφή της ταχύτητας των αυτοκινήτων που διέρχονται από τον BOAK απαιτείται η καταχώρηση ενός συγκεκριμένου αριθμού (id) για κάθε αυτοκίνητο. Ως αποτέλεσμα όταν τελειώσει η επεξεργασία του βίντεο υπάρχουν πληροφορίες τόσο για τις ταχύτητες που αναπτύσσονται κατά μήκος του BOAK, όσο και για τον αριθμό των αυτοκινήτων που κινούνται.

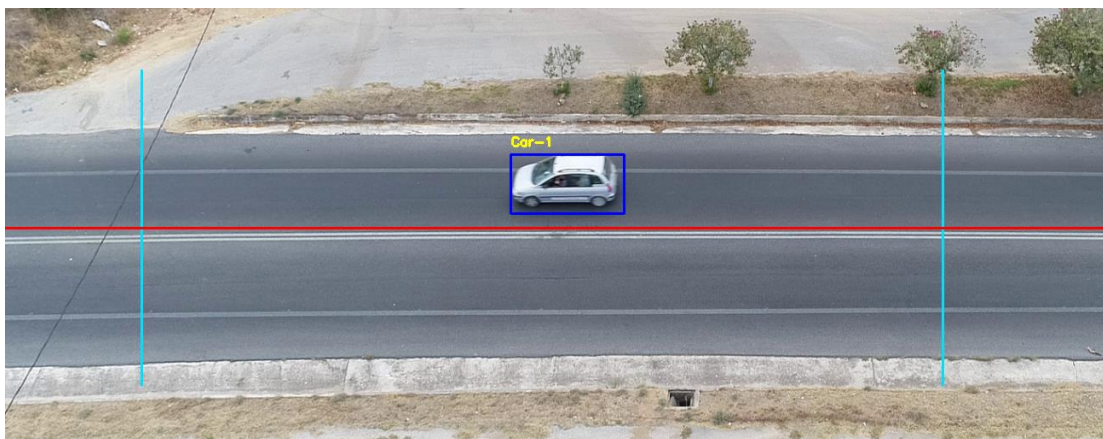
Η ανάθεση ενός συγκεκριμένου id για κάθε αυτοκίνητο είναι απλή και έχει άμεση σχέση με τους δύο πίνακες, που παράγονται από τον αλγόριθμο YOLO και περιλαμβάνουν τα ονόματα των κλάσεων που ανιχνεύονται και τα δεδομένα των πλαισίων οριοθέτησής τους. Τα δεδομένα του τελευταίου πίνακα αποτελούν οι συντεταγμένες του άνω αριστερά άκρου του πλαισίου οριοθέτησης σε σχέση με την

εικόνα, καθώς και το ύψος και το πλάτος του πλαισίου αυτού. Στην αρχική εμφάνιση και πρόβλεψη ενός αυτοκινήτου αποθηκεύονται οι συντεταγμένες του κέντρου του πλαισίου οριοθέτησης και το αντίστοιχο id σε ένα λεξικό (dictionary). Στο επόμενο frame που παρουσιάζεται γίνεται έλεγχος για την απόσταση που υπάρχει μεταξύ των αυτοκινήτων του προηγούμενου frame και του παρόντος. Αν η απόσταση είναι κάτω από ένα όριο τότε το λεξικό ενημερώνεται και αλλάζει τις συντεταγμένες του κέντρου σύμφωνα με το παρόν frame.

Από την άλλη πλευρά, ο υπολογισμός της ταχύτητας των αυτοκινήτων υλοποιείται με μία πιο σύνθετη μέθοδο. Αρχικά, με την βοήθεια του google maps υπολογίζεται η απόσταση, που καλύπτει ένα αυτοκίνητο στην περιοχή που καταγράφεται το βίντεο από το μη επανδρωμένο αεροσκάφος, όπως φαίνεται στην εικόνα 31. Στην συνέχεια, υπολογίζεται το σύνολο των frame, που χρειάζεται το αυτοκίνητο για να καλύψει την συγκεκριμένη απόσταση. Γνωρίζοντας, λοιπόν, τον σύνολο των εικόνων (frames) ανά δευτερόλεπτο (fps), που καταγράφεται το βίντεο, υπολογίζεται η ταχύτητα των αυτοκινήτων. Πιο αναλυτικά, η διαίρεση του συνόλου των frame, που χρειάζεται από το αυτοκίνητο για να καλύψει την συγκεκριμένη απόσταση, με τον συνολικό αριθμό εικόνων ανά δευτερόλεπτο της καταγραφής του βίντεο, έχει ως αποτέλεσμα την συνολική χρονική διάρκεια, που χρειάστηκε το αυτοκίνητο για να καλύψει την συγκεκριμένη απόσταση, σε δευτερόλεπτα. Τέλος, η διαίρεση της απόστασης με την παραπάνω χρονική διάρκεια έχει ως αποτέλεσμα την εύρεση της ταχύτητας, που αναπτύσσει το αυτοκίνητο στην συγκεκριμένη περιοχή, σε μέτρα ανά δευτερόλεπτο (m/s). Για την μετατροπή της τιμής της ταχύτητας από μέτρα ανά δευτερόλεπτο σε χιλιόμετρα ανά ώρα (km/h) απαιτείται ο πολλαπλασιασμός με την τιμή 3,6. Η παραπάνω διαδικασία πραγματοποιείται για κάθε id.



Εικόνα 30: Μέτρηση της απόστασης με την βοήθεια του Google Maps



Εικόνα 31: Η απόσταση (μεταξύ των δύο γαλάζιων γραμμών) όπως φαίνεται από το βίντεο του drone

Για παράδειγμα, το σύνολο των frame, που χρειάστηκε το αυτοκίνητο που απεικονίζεται στην εικόνα 32 για να καλύψει την απόσταση από τη μία γαλάζια γραμμή έως την άλλη, είναι 110. Ο αριθμός των εικόνων ανά δευτερόλεπτο, που καταγράφεται το βίντεο, είναι 60 fps. Το αποτέλεσμα της διαίρεσης των 110 frame προς τα 60 fps είναι 1,83 δευτερόλεπτα. Επιπλέον, όπως φαίνεται από την εικόνα 31 η απόσταση, που διανύει το αυτοκίνητο είναι 31 μέτρα. Επομένως, η ταχύτητα, που αναπτύσσει το συγκεκριμένο αυτοκίνητο σε μέτρα ανά δευτερόλεπτο (m/s) ορίζεται από το αποτέλεσμα της διαίρεσης της απόστασης προς τον χρόνο. Η τιμή της ταχύτητας που προκύπτει με τα συγκεκριμένα δεδομένα είναι 16,9 m/s και η μετατροπή της σε χιλιόμετρα ανά ώρα προκύπτει από τον πολλαπλασιασμό της παραπάνω τιμής με το 3,6. Το τελικό αποτέλεσμα είναι 60 km/h.

3.5 Προβλήματα

Κατά την χρήση της παραπάνω μεθοδολογίας στα βίντεο, που κατέγραψε το μη επανδρωμένο αεροσκάφος, παρατηρήθηκαν διάφορα προβλήματα. Πρώτα απ' όλα, η ανάλυση της εικόνας εισόδου 416x416 έχει ως αποτέλεσμα η ταχύτητα εκτέλεσης του αλγορίθμου στο βίντεο να πραγματοποιείται με 13 εικόνες το δευτερόλεπτο(fps) ανεξαρτήτως της ταχύτητας εγγραφής του βίντεο από το drone. Παραδείγματος χάρη, ένα βίντεο των 5 λεπτών με ταχύτητα εγγραφής τα 60 fps έχει συνολικά 18000 frame. Ο χρόνος που απαιτείται για την συνολική εκτέλεση του αλγορίθμου στο βίντεο είναι περίπου 23 λεπτά. Με την αλλαγή της ανάλυσης της εικόνας εισόδου σε 320x320 ο συνολικός χρόνος εκτέλεσης του αλγορίθμου μειώνεται στα 17 λεπτά περίπου. Το κόστος όμως της χρήσης της ανάλυσης 320x320 είναι η αστοχία της πρόβλεψης των αυτοκινήτων σε διάφορα διαστήματα του βίντεο.

Επίσης, παρατηρήθηκε πως με την χρήση της ανάλυσης της εικόνας εισόδου 416x416 υπάρχουν μερικά σημεία στη διάρκεια του βίντεο, στα οποία ο αλγόριθμος αδυνατούσε να προβλέψει την ύπαρξη κάποιων αυτοκινήτων. Το συγκεκριμένο πρόβλημα έχει ως αποτέλεσμα την λανθασμένη εκτίμηση τόσο του αριθμού των αυτοκινήτων που κινούνται στον BOAK, όσο και της ταχύτητας αυτών. Το πρόβλημα

επιλύεται με την αύξηση της ανάλυσης της εικόνας σε 512x512 και σε 608x608 για ακόμη καλύτερα αποτελέσματα στην πρόβλεψη των αντικειμένων. Παρόλα αυτά, οι συγκεκριμένες αναλύσεις της εικόνας εισόδου έχουν ένα μεγάλο μειονέκτημα. Όπως έχει αναφερθεί, η ταχύτητα εκτέλεσης του αλγορίθμου YOLO σε βίντεο με ανάλυση εικόνας εισόδου 512x512 είναι τα 9 fps, ενώ με ανάλυση 608x608 είναι τα 6 fps. Το μειονέκτημα, λοιπόν, των παραπάνω είναι η πολύ μεγάλη χρονική διάρκεια που απαιτείται για την πλήρη επεξεργασία ενός βίντεο. Για παράδειγμα, για ένα βίντεο των 5 λεπτών με ταχύτητα εγγραφής τα 60 fps ο συνολικός χρόνος που χρειάζεται για την εκτέλεση του αλγορίθμου με ανάλυση της εικόνας εισόδου 512x512 είναι περίπου 33 λεπτά, ενώ για την ανάλυση 608x608 είναι περίπου 50 λεπτά.

3.6 Επίλυση προβλημάτων

3.6.1 Προτεινόμενες λύσεις

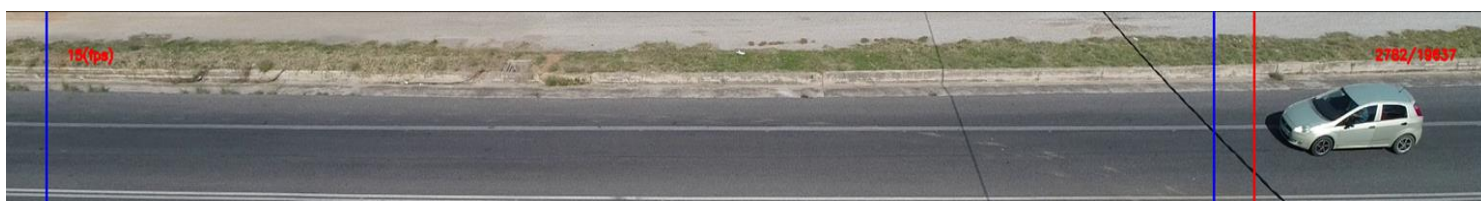
Το σημαντικότερο από τα προβλήματα, που αναφέρονται παραπάνω, είναι η χρονική διάρκεια επεξεργασίας των βίντεο για τις αναλύσεις 512x512 και 608x608. Όπως έχει ήδη αναφερθεί, η κάρτα γραφικών, η οποία χρησιμοποιείται στην παρούσα διπλωματική εργασία, είναι η NVIDIA GTX 1060 3GB. Η δεδομένη κάρτα γραφικών έχει πολύ μικρότερη ισχύ σε σχέση με την κάρτα γραφικών, που εξετάστηκε η τέταρτη έκδοση του YOLO, η οποία είναι η NVIDIA RTX 2070. Επομένως, μία ενδεικτική λύση είναι η χρήση της κάρτας γραφικών NVIDIA RTX 2070 με την ανάλυση της εικόνας εισόδου στα 608x608. Με αυτόν τον τρόπο, οι προβλέψεις των οχημάτων κατά μήκος του BOAK πραγματοποιούνται σωστά χωρίς διακοπές κατά την διάρκεια του βίντεο, ενώ η ταχύτητα εκτέλεσης του αλγορίθμου είναι τα 34 fps. Παρόλα αυτά, η εναλλακτική αυτή λύση απορρίπτεται λόγω του ακριβού κόστους αγοράς της συγκεκριμένης κάρτας γραφικών σε συνδυασμό με την δυσκολία εύρεσής της στην εγχώρια αγορά.

Η λύση, που παρουσιάζεται και επιλέγεται για την παρούσα διπλωματική εργασία αξιοποιεί τον αλγόριθμο YOLO και τις μεθόδους ανίχνευσης κίνησης της βιβλιοθήκης OpenCV. Πιο αναλυτικά, η βιβλιοθήκη OpenCV περιλαμβάνει διάφορες μεθόδους ανίχνευσης κίνησης αντικειμένων σε βίντεο, οι οποίες χρησιμοποιούν τεχνικές με σκοπό την αφαίρεση του παρασκηνίου και την ανάδειξη των αντικειμένων που κινούνται (background subtraction) με την χρήση φίλτρων.

Η εφαρμογή της μεθόδου ανίχνευσης κίνησης πραγματοποιείται καθ' όλη τη διάρκεια του βίντεο και σε ολόκληρη την εικόνα που καλύπτει το δρόμο. Όταν ένα όχημα εμφανίζεται στην αρχή του δρόμου, τότε ανιχνεύεται η κίνησή του και αναγνωρίζονται οι ακριβείς συντεταγμένες του σε σχέση με τη γενική εικόνα για την τοποθέτηση πλαισίου οριοθέτησης γύρω από αυτό. Στην συνέχεια, με την βοήθεια των συντεταγμένων, αποκόπτεται η εικόνα που περιλαμβάνει μόνο το όχημα που εντοπίστηκε και στέλνεται προς αναγνώριση στον αλγόριθμο YOLO. Αν το όχημα ανιχνεύεται ως αυτοκίνητο τότε ένας μετρητής αυξάνεται κατά μία μονάδα. Παράλληλα, αυξάνεται με τον ίδιο ρυθμό ένας διαφορετικός μετρητής, ο οποίος υπολογίζει τον συνολικό αριθμό επαναλήψεων, που στάλθηκε προς αναγνώριση το συγκεκριμένο όχημα. Τα δεδομένα του πλαισίου οριοθέτησης και των μετρητών αποθηκεύονται στο λεξικό της Python. Η διαδικασία αυτή επαναλαμβάνεται μέχρι το όχημα να διανύσει μία μικρή απόσταση. Στο τέλος της απόστασης, υπολογίζεται το

ποσοστό εμφάνισης αυτοκινήτου και στην περίπτωση που είναι πάνω από 50% τότε η καταγραφή των δεδομένων του οχήματος συνεχίζεται για την εύρεση της ταχύτητάς του. Σε διαφορετική περίπτωση διαγράφονται τα δεδομένα του οχήματος.

Για παράδειγμα, στην παρακάτω εικόνα το όχημα στέλνεται για αναγνώριση στον αλγόριθμο YOLO μέχρι να περάσει την κόκκινη γραμμή που απεικονίζεται στην εικόνα. Όπως αναφέρεται παραπάνω, καθ' όλη τη διάρκεια που το όχημα βρίσκεται μεταξύ της αρχής της εικόνας και της κόκκινης γραμμής, ένας μετρητής αυξάνεται κατά μία μονάδα σε περίπτωση που το όχημα αναγνωρίστηκε ως αυτοκίνητο. Επίσης, ένας διαφορετικός μετρητής αυξάνεται κατά μία μονάδα, όποτε στέλνεται το συγκεκριμένο όχημα προς αναγνώριση. Στο συγκεκριμένο παράδειγμα, το όχημα στάλθηκε προς αναγνώριση στον αλγόριθμο YOLO 19 φορές και αναγνωρίστηκε ως αυτοκίνητο τις 19, με το ποσοστό εμφάνισης αυτοκινήτου να είναι 100%. Στην συνέχεια, μεταξύ των μπλε γραμμών καταγράφεται η ταχύτητα του αυτοκινήτου με την ίδια τεχνική που έχει αναλυθεί στο κεφάλαιο 3.4 με τα δεδομένα του πλαισίου οριοθέτησης να παρέχονται από την μέθοδο ανίχνευσης κίνησης. Σε περίπτωση που το ποσοστό εμφάνισης αυτοκινήτου είναι κάτω από 50%, τότε τα δεδομένα που προκύπτουν από την μέθοδο ανίχνευσης κίνησης για το πλαίσιο οριοθέτησης χαρακτηρίζονται ως don't care και δεν αξιοποιούνται.



Εικόνα 32: Παράδειγμα εφαρμογής της παραπάνω τεχνικής

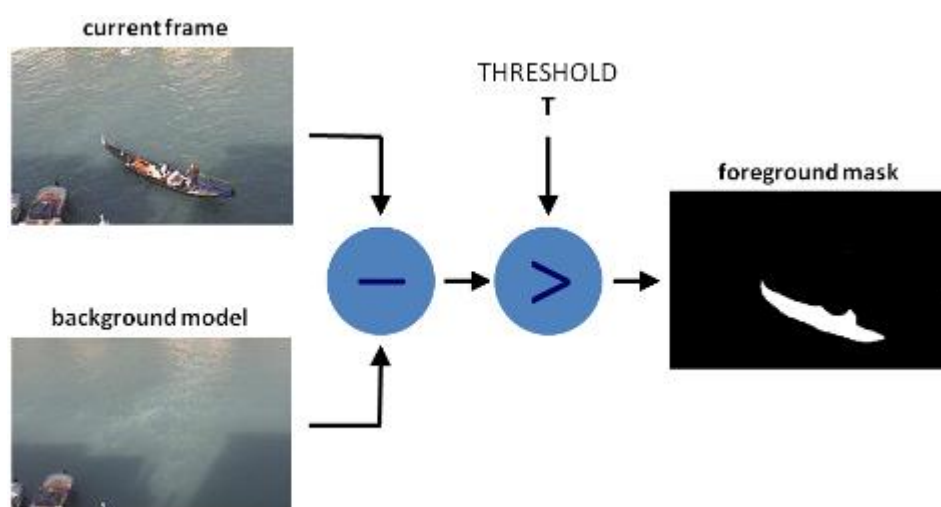


Εικόνα 33: Εικόνα που εισάγεται στον αλγόριθμο YOLO

3.6.2 Μέθοδοι ανίχνευσης κίνησης

Η μέθοδος του background subtraction αποτελεί μία ευρέως γνωστή τεχνική για την υπολογιστική όραση, όπου το προσκήνιο των εικόνων ενός βίντεο αποκόπτεται για περαιτέρω επεξεργασία. Για να χρησιμοποιηθούν οι μέθοδοι απαιτείται η κάμερα που καταγράφει το βίντεο να είναι αρκετά σταθερή και να μην κινείται. Οι τρεις μέθοδοι, που εξετάζονται, ονομάζονται MOG, MOG2 και GMG. Η έξοδος, που παράγεται από τις παραπάνω μεθόδους για κάθε εικόνα ενός βίντεο, είναι μία δυαδική εικόνα, η οποία παρουσιάζει το παρασκήνιο και τα αντικείμενα που κινούνται. Πιο συγκεκριμένα, το παρασκήνιο παρουσιάζεται από pixels των οποίων

το χρώμα είναι μαύρο, δηλαδή τους αναθέτεται η τιμή 0, και το προσκίγνιο από pixels των οποίων το χρώμα είναι άσπρο, δηλαδή με τιμή 1, όπως φαίνεται στην παρακάτω εικόνα. [41]



Εικόνα 34: Η μέθοδος Background Subtraction

(Πηγή: https://docs.opencv.org/4.x/d1/dc5/tutorial_background_subtraction.html)

Παρακάτω απεικονίζεται το στιγμιότυπο ενός βίντεο, το οποίο καταγράφει τις κινήσεις των ανθρώπων, στην αρχική του μορφή. Στην συνέχεια κατά την εξήγηση των μεθόδων απεικονίζονται τα αποτελέσματα τους σε σχέση με την αρχική εικόνα.



Εικόνα 35: Αρχική εικόνα

(Πηγή: https://docs.opencv.org/4.x/d8/d38/tutorial_bgsegm_bg_subtraction.html)

- **MOG (Mixture of Gaussians)**

Η συγκεκριμένη μέθοδος αποτελεί έναν Gaussian Mixture-based αλγόριθμο και δημοσιεύτηκε για πρώτη φορά στο paper "An improved adaptive background mixture model for real-time tracking with shadow detection" το 2001 από τους P. KadewTraKuPong και R. Bowden. Χρησιμοποιεί ένα σύνολο από k Gaussian

κατανομές για να καθορίσει κάθε pixel του φόντου, όπου οι τιμές του k κυμαίνονται μεταξύ του 3 και του 5. Σύμφωνα με τους ερευνητές οι διάφορες κατανομές αντιπροσωπεύουν τα διαφορετικά χρώματα του προσκηνίου και του φόντου. Τα βάρη των κατανομών είναι ανάλογα της χρονικής διάρκειας που κάθε χρώμα παραμένει σε ένα pixel. Επομένως, όταν το βάρος της κατανομής ενός pixel είναι χαμηλό, τότε το pixel θεωρείται προσκήνιο. [42]

Στην Python για να χρησιμοποιηθεί η μέθοδος απαιτείται η ενεργοποίηση της με την συνάρτηση `cv2.bgsegm.createBackgroundSubtractorMOG()`, ενώ για την λήψη του φίλτρου προσκηνίου χρησιμοποιείται η συνάρτηση `backgroundsubtractor.apply()`. Αξίζει να σημειωθεί πως, η συνάρτηση ενεργοποίησης δέχεται τέσσερις παραμέτρους ως είσοδο, με σκοπό την ρύθμιση των αποτελεσμάτων της.

Πιο αναλυτικά, η παράμετρος `history` καθορίζει τον αριθμό των frame, που χρησιμοποιούνται για την ρύθμιση των βαρών του μοντέλου, σε όλη την διάρκεια επεξεργασίας του βίντεο. Οι χαμηλές τιμές καταλήγουν σε αυξημένη ευαισθησία στις ξαφνικές αλλαγές της φωτεινότητας των pixel. Επιπλέον, η παράμετρος `nmixtures` υποδηλώνει τον συνολικό αριθμό των Gaussian κατανομών, που χρησιμοποιούνται στο βίντεο. Οι υψηλές τιμές αυξάνουν σημαντικά τον χρόνο επεξεργασίας του εκάστοτε βίντεο. Επιπρόσθετα, η παράμετρος `backgroundRatio` ορίζει το κατώφλι (`threshold`) του βάρους για την διαφοροποίηση των pixel του προσκηνίου και του φόντου. Οι χαμηλές τιμές ενδέχεται να οδηγήσει σε λανθασμένα αποτελέσματα. Τέλος, η παράμετρος `noiseSigma` καθορίζει τον θόρυβο, που εμφανίζεται στα αποτελέσματα. Όσο μικρότερη είναι η τιμή του, τόσο περισσότερες είναι οι πιθανότητες να εμφανιστεί θόρυβος, ενώ όσο μεγαλύτερη είναι η τιμή του η δημιουργία θορύβου περιορίζεται.



Εικόνα 36: Αποτελέσματα μεθόδου MOG

(Πηγή: https://docs.opencv.org/4.x/d8/d38/tutorial_bgsegm_bg_subtraction.html)

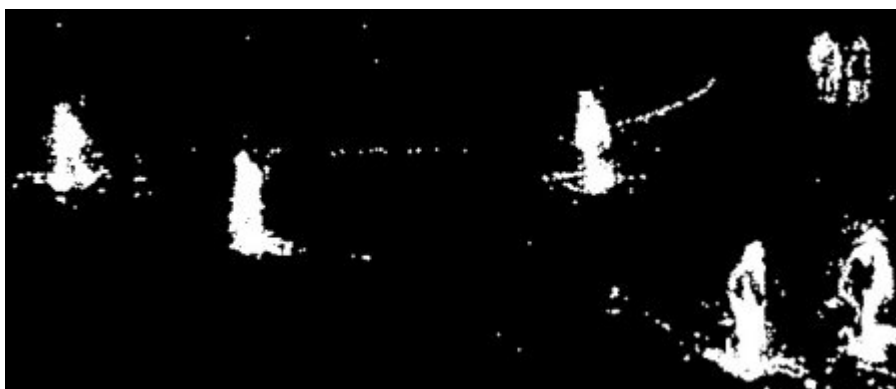
- **GMG**

Η συγκεκριμένη μέθοδος δημοσιεύτηκε για πρώτη φορά από τους Andrew B. Godbehere, Akihiro Matsukawa και Ken Goldberg το 2012 στο paper "Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation" και προσπαθεί να μοντελοποιήσει το παρασκήνιο των frame ενός

βίντεο με τη χρήση φίλτρων Kalman και της Bayesian κατάτμησης. Κατά το πρώτο στάδιο της μεθόδου σε κάθε pixel αναθέεται μία τιμή, η οποία εξαρτάται από το χρονικό διάστημα που ένα χρώμα παραμένει στο ίδιο pixel. Σε κάθε εικόνα του βίντεο παρατηρείται το χρώμα κάθε pixel ενημερώνοντας, παράλληλα, την τιμή που έχει ανατεθεί. Τα pixel των οποίων το χρώμα παραμένει αμετάβλητο για ένα δεδομένο χρονικό διάστημα θεωρούνται παρασκήνιο. Αντίθετα, στο δεύτερο στάδιο ελέγχονται τα pixel του προσκήνιου, με σκοπό την μείωση του θορύβου. [43]

Στην Python για να χρησιμοποιηθεί η μέθοδος απαιτείται η ενεργοποίηση της με την συνάρτηση `cv2.bgsegm.createBackgroundSubtractorGMG()`, ενώ για την λήψη του φίλτρου προσκήνιου χρησιμοποιείται η συνάρτηση `backgroundsubtractor.apply()`. Αξίζει να σημειωθεί πως, η συνάρτηση ενεργοποίησης δέχεται δύο παραμέτρους ως είσοδο για την ρύθμιση των αποτελεσμάτων της.

Πιο συγκεκριμένα, η παράμετρος `initializationFrames` δηλώνει τον αριθμό των frame που χρειάζονται για την ενεργοποίηση της μεθόδου, με αποτέλεσμα κατά την αρχικοποίηση οι εικόνες, που απεικονίζουν το προσκήνιο να είναι πάντα μαύρες. Όσο, λοιπόν, περισσότερα frame χρησιμοποιούνται για την αρχικοποίηση του μοντέλου τόσο λιγότερο θόρυβο εμφανίζει. Επιπλέον, η παράμετρος `decisionThreshold` καθορίζει την τιμή του κατωφλίου, από την οποία τα pixel κατηγοριοποιούνται ως προσκήνιο ή παρασκήνιο. Οι υψηλές τιμές έχουν ως αποτέλεσμα την μη πλήρη ανίχνευση των αντικειμένων που κινούνται.



Εικόνα 37: Αποτέλεσμα μεθόδου GMG

(Πηγή: https://docs.opencv.org/4.x/d8/d38/tutorial_bgsegm_bg_subtraction.html)

- **MOG2 (Mixture of Gaussians 2)**

Η δεύτερη έκδοση της μεθόδου Mixture of Gaussians βασίζεται στα paper "Improved adaptive Gaussian mixture model for background subtraction" και "Efficient adaptive density estimation per image pixel for the task of background subtraction", όπου το πρώτο δημοσιεύθηκε από τον Z. Zivkovic το 2004 και το δεύτερο από τον Z. Zivkovic σε συνεργασία με τον Van Der Heijden το 2006. Ο κύριος στόχος της μεθόδου MOG2 είναι η επίλυση των περιορισμών, που παρουσιάζονται στην πρώτη έκδοση, δηλαδή στο πρόβλημα του αριθμού των κατανομών που εφαρμόζονται. Ουσιαστικά, η συγκεκριμένη μέθοδος επιλέγει τον κατάλληλο αριθμό των Gaussian κατανομών για κάθε pixel, επιτυγχάνοντας έτσι μια καλύτερη αναπαράσταση της πολυπλοκότητας των χρωμάτων σε κάθε frame. [44]

Στην Python για να χρησιμοποιηθεί η συγκεκριμένη μέθοδος απαιτείται η ενεργοποίηση της με την συνάρτηση `cv2.createBackgroundSubtractorMOG2()`, ενώ για την λήψη του φίλτρου του προσκηνίου χρησιμοποιείται η συνάρτηση `backgroundsubtractor.apply()`. Αξίζει να σημειωθεί πως, η συνάρτηση ενεργοποίησης της μεθόδου δέχεται τρεις παραμέτρους ως είσοδο για την ρύθμιση των αποτελεσμάτων της.

Αναλυτικότερα, η παράμετρος `history` υποδηλώνει τον αριθμό των frame που χρησιμοποιούνται για την μοντελοποίηση του φόντου. Επιπλέον, η παράμετρος `varThreshold` συγκρίνει τις τιμές του βάρους των pixel σε ένα frame με βάση τις τιμές του μοντέλου, που έχει κατασκευαστεί. Οι χαμηλές τιμές της συγκεκριμένης παραμέτρου τείνουν να ανιχνεύουν αντικείμενα που δεν κινούνται. Ολοκληρώνοντας, η παράμετρος `detectShadows` ενεργοποιεί ή απενεργοποιεί την ανίχνευση της σκιάς των αντικειμένων που κινούνται και την εμφανίζει σε γκρι χρώμα. Με την ενεργοποίηση της παραπάνω παραμέτρου ο χρόνος επεξεργασίας αυξάνεται.



Εικόνα 38: Αποτέλεσμα μεθόδου MOG2

(Πηγή: https://docs.opencv.org/4.x/d1/dc5/tutorial_background_subtraction.html)

3.6.3 Επιλογή κατάλληλης μεθόδου

Σύμφωνα με το paper "A Comparison between Background Modelling Methods for Vehicle Segmentation in Highway Traffic Videos" [45], που δημοσιεύθηκε το 2018 από τους L. A. Marcomini και A. L. Cunha η καλύτερη μέθοδος για την ανίχνευση κίνησης των οχημάτων σε ένα βίντεο είναι η MOG2. Ωστόσο, κατά την εφαρμογή της μεθόδου MOG2 παρατηρήθηκε η εμφάνιση θορύβου στις εικόνες εξόδου, με αποτέλεσμα τα πλαίσια οριοθέτησης των κινούμενων οχημάτων να είναι μεγαλύτερα από τα πραγματικά, έχοντας σαν συνέπεια την λανθασμένη εκτίμηση της ταχύτητας των αυτοκινήτων.

Για την επίλυση του προβλήματος αξιοποιούνται οι μέθοδοι `erode` και `dilation` της βιβλιοθήκης `OpenCV` [46]. Όπως έχει αναφερθεί οι εικόνες εξόδου της μεθόδου MOG2 αποτελούνται από μαύρα και άσπρα pixel, στα οποία αναθέτονται η τιμή 0

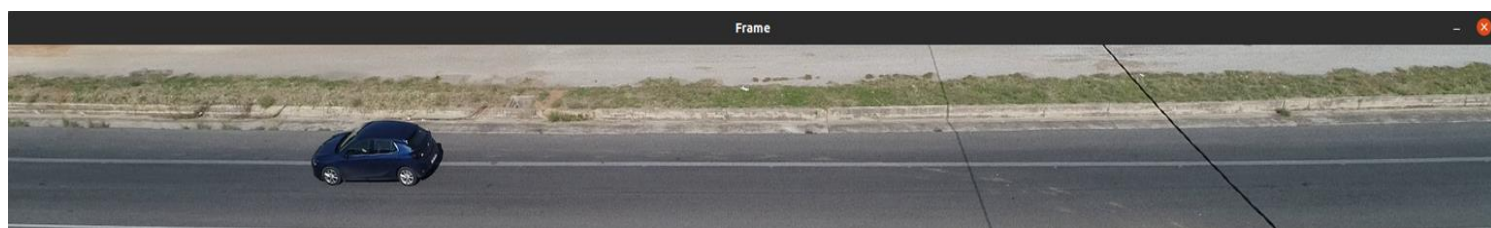
και 1 αντίστοιχα. Ουσιαστικά, οι εικόνες εξόδου αποτελούν πίνακες των οποίων τα στοιχεία δέχονται την τιμή 0 ή 1.

Στην μέθοδο erode ένας πίνακας kernel περιτού μεγέθους (3x3, 5x5, 7x7 κλπ) ελέγχει τα pixel της εικόνα. Το αποτέλεσμα που προκύπτει εξαρτάται από το μέγεθος του πίνακα kernel και από το σύνολο των άσπρων pixel. Πιο συγκεκριμένα, ένα pixel θεωρείται άσπρο αν όλα τα pixel που εξετάζονται από τον πίνακα kernel είναι άσπρα, δηλαδή έχουν τιμή 1, σε διαφορετική περίπτωση το pixel γίνεται μαύρο. Αποτέλεσμα των παραπάνω, είναι η οριστική αφαίρεση του θορύβου που υπάρχει από την μέθοδο MOG2. Παρόλα αυτά, με την χρήση της συγκεκριμένης τεχνικής μειώνεται και το μέγεθος των οχημάτων που ανιχνεύονται.

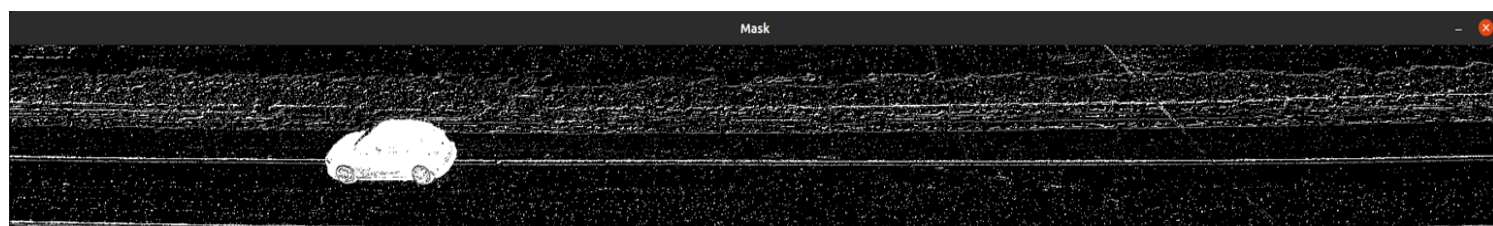
Για την αντιμετώπιση του παραπάνω προβλήματος αξιοποιείται η τεχνική του dilation. Η αρχή λειτουργίας του είναι η ακριβώς αντίθετη από την τεχνική erode. Ουσιαστικά, στο τελικό αποτέλεσμα ένα pixel θεωρείται άσπρο αν τουλάχιστον ένα από τα pixel που εξετάζονται από τον πίνακα kernel είναι άσπρο.

Στην παρούσα διπλωματική εργασία το μέγεθος του πίνακα kernel που χρησιμοποιείται για την τεχνική erode έχει μέγεθος (5,5). Από την άλλη πλευρά, το μέγεθος του πίνακα kernel που χρησιμοποιείται στην τεχνική dilation έχει μέγεθος (9,9).

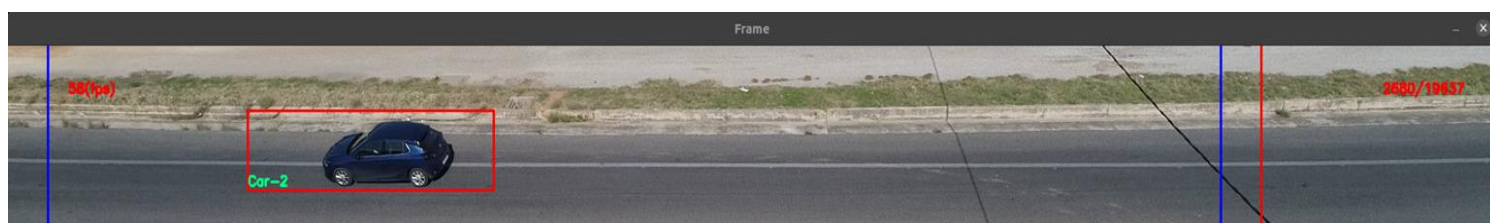
Με την χρήση των παραπάνω τεχνικών αφαιρείται ο θόρυβος, που παρουσιάζεται, χωρίς να επηρεάζεται σημαντικά το μέγεθος των οχημάτων που ανιχνεύονται λόγω της κίνησής τους, όπως απεικονίζεται στις παρακάτω εικόνες.



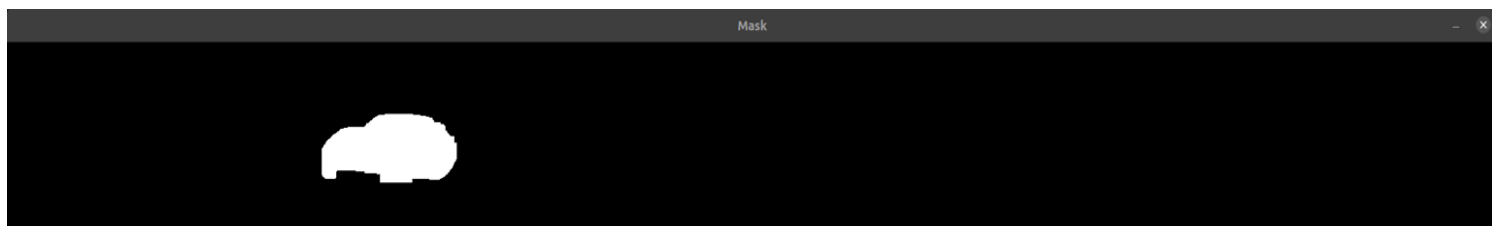
Εικόνα 39: Αρχική εικόνα



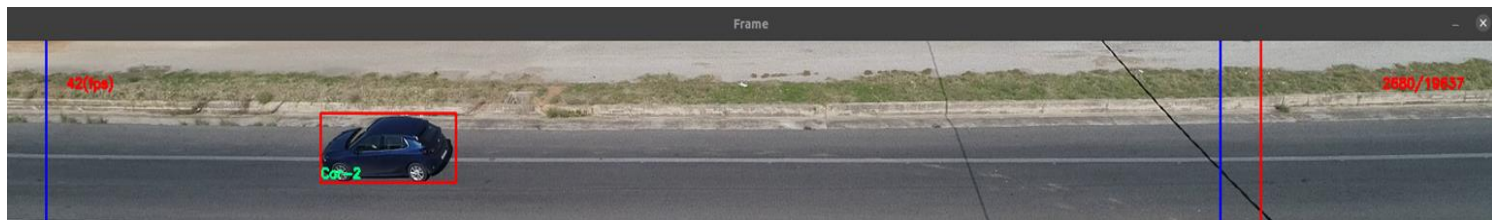
Εικόνα 40: Αποτέλεσμα μεθόδου MOG2 χωρίς τις τεχνικές erode και dilation



Εικόνα 41: Αποτέλεσμα πλαισίου οριοθέτησης χωρίς την χρήση των τεχνικών erode και dilation

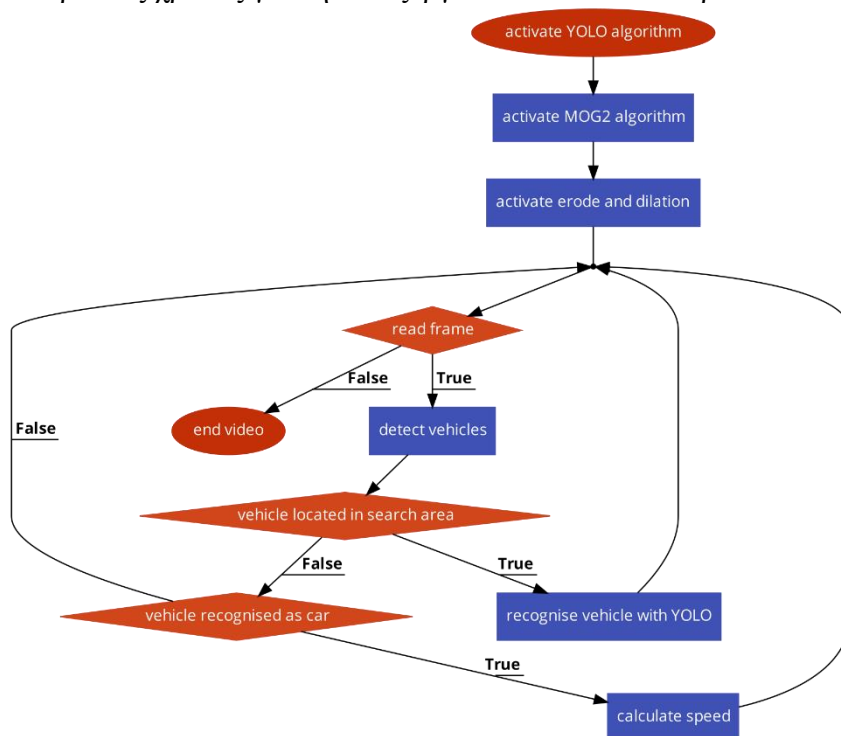


Εικόνα 42: Αποτέλεσμα μεθόδου MOG2 με τις τεχνικές erode και dilation



Εικόνα 43: Αποτέλεσμα πλαισίου οριοθέτησης με τις τεχνικές erode και dilation

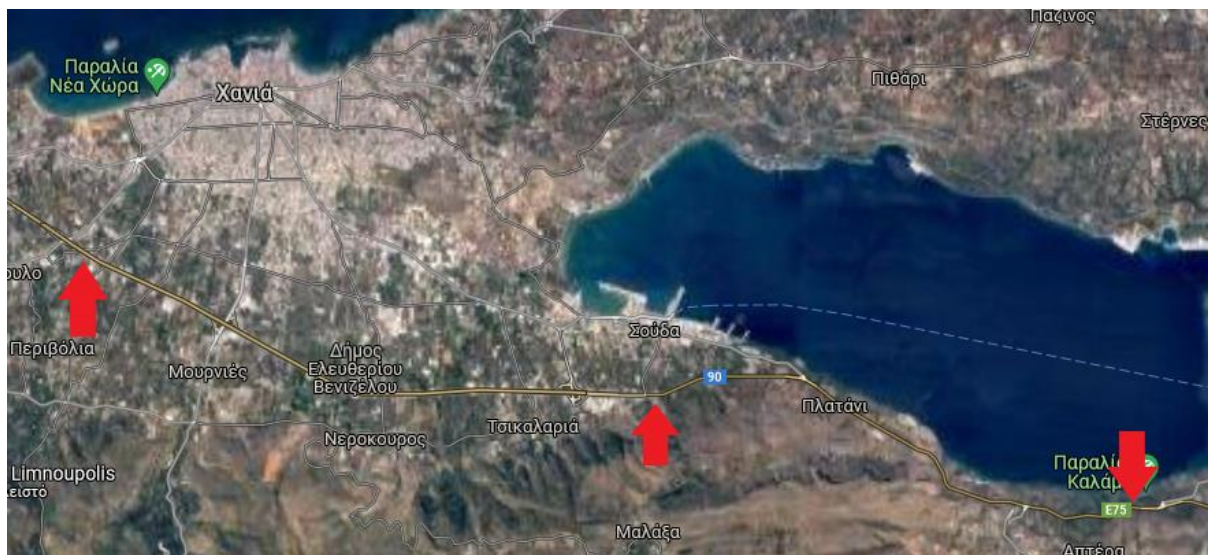
Αξίζει να σημειωθεί πως, η ταχύτητα επεξεργασίας του βίντεο με την χρήση των μεθόδων erode, dilation και MOG2 είναι 40 fps όταν τα οχήματα δεν βρίσκονται στην περιοχή αναγνώρισής τους. Αντίθετα, κατά την αναγνώριση των οχημάτων από τον αλγόριθμο YOLO με την ανάλυση της εικόνας εισόδου να είναι 320x320 η ταχύτητα επεξεργασίας μειώνεται στα 15 fps, ενώ σε μεμονωμένες περιπτώσεις που βρίσκονται δύο οχήματα στην περιοχή αναγνώρισης η ταχύτητα επεξεργασίας μειώνεται περαιτέρω στα 9 fps. Τα αποτελέσματα αυτά είναι πολύ ικανοποιητικά, καθώς δεν απαιτείται αρκετός χρόνος για την επεξεργασία του εκάστοτε βίντεο.



Διάγραμμα 1: Διάγραμμα ροής της μεθοδολογίας καταγραφής των ταχυτήτων

3.7 Αποτελέσματα ταχυτήτων

Ο αριθμός των βίντεο, που επεξεργάστηκαν για την εύρεση του συνόλου των αυτοκινήτων που κινούνται στον ΒΟΑΚ και των ταχυτήτων που αναπτύσσουν κατά μήκος του, είναι 8. Η χρονική διάρκεια του εκάστοτε βίντεο διαφέρει λόγω του περιορισμού της διάρκειας πτήσης του drone, με τα περισσότερα βίντεο να έχουν χρονική διάρκεια 5 λεπτών. Όπως έχει αναφερθεί, οι περιοχές στις οποίες καταγράφηκαν τα βίντεο είναι τρεις και είναι περιφερειακά των Χανίων με απόσταση 7 χιλιομέτρων η μία περιοχή από την άλλη, όπως φαίνεται από την παρακάτω εικόνα.



Εικόνα 44: Τα τρία σημεία που καταγράφηκαν τα βίντεο
(Πηγή: <https://www.google.gr/maps>)

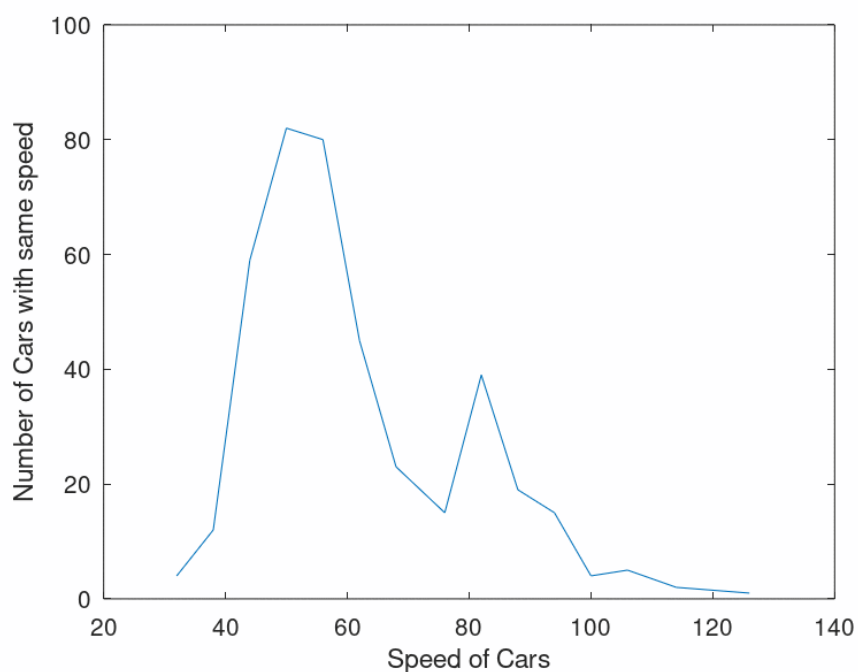
Ο συνολικός αριθμός των αυτοκινήτων, που καταγράφηκαν από όλα τα βίντεο, είναι 405, ενώ οι ταχύτητες που αναπτύσσουν διαφέρουν σε μεγάλο βαθμό εξαιτίας των εξόδων που υπάρχουν στις συγκεκριμένες περιοχές του ΒΟΑΚ, που έχουν ως αποτέλεσμα την μείωση της ταχύτητας των αυτοκινήτων, αλλά και της κίνησης που παρουσιάζεται στις περιοχές αυτές. Πιο αναλυτικά, ο παρακάτω πίνακας απεικονίζει τον αριθμό των αυτοκινήτων που έχουν κατά 5χλμ/ώρα την ίδια ταχύτητα.

Αριθμός Αυτοκινήτων	Ταχύτητα
4	32
12	38
59	44
82	50
80	56
45	62
23	68
15	76
39	82
19	88

15	94
4	100
5	106
2	114
1	126

Πίνακας 3: Αριθμός αυτοκινήτων με την ίδια ταχύτητα (κατά 5χλμ/ώρα)

Παρακάτω απεικονίζεται το διάγραμμα που προκύπτει από τον αριθμό των αυτοκινήτων που έχουν την ίδια ταχύτητα σε σχέση με τις ταχύτητες που τους αναλογούν.



Εικόνα 45: Διάγραμμα του αριθμού των αυτοκινήτων με την ίδια ταχύτητα

Κεφάλαιο 4^ο

4.1 Εισαγωγή

Σε αυτό το κεφάλαιο παρουσιάζεται το εργαλείο και η μεθοδολογία, που χρησιμοποιήθηκαν για την ενεργειακή αξιολόγηση των ταχυτήτων σε τέσσερα αμιγώς ηλεκτρικά αυτοκίνητα, τα οποία υπάρχουν ήδη στην αγορά και είναι ελεύθερα προς τους καταναλωτές. Επίσης, παρουσιάζονται τα αποτελέσματα της ενεργειακής αξιολόγησης, τα οποία αποτελούν την χιλιομετρική απόσταση από την εκκίνηση κατά την οποία ο οδηγός ενός ηλεκτρικού αυτοκινήτου αναζητάει σταθμό φόρτισης για τις μπαταρίες του αυτοκινήτου του για κάθε ώρα της ημέρας. Τέλος, με σκοπό τα αποτελέσματα να ανταποκρίνονται περισσότερο στην πραγματικότητα, κατασκευάζεται ένα σύστημα ασαφούς λογικής (Fuzzy Inference System), το οποίο εξάγει την πιθανότητα ένας οδηγός αμιγούς ηλεκτρικού αυτοκινήτου να αναζητάει σταθμό φόρτισης δεδομένου του ποσοστού φόρτισης της μπαταρίας και του άγχους που έχει ο εκάστοτε οδηγός για αυτό.

4.2 OCTAVE

Η ενεργειακή αξιολόγηση των ταχυτήτων, που αναπτύσσονται κατά μήκος του Βόρειου Οδικού Άξονα της Κρήτης, σε τέσσερα μοντέλα αμιγώς ηλεκτροκίνητων αυτοκινήτων πραγματοποιείται με την βοήθεια του εργαλείου Octave.

Το Octave αποτελεί μία υψηλού επιπέδου γλώσσα, που διαθέτει μία διαδραστική πλατφόρμα, η οποία έχει σχεδιαστεί για την ανάλυση και επίλυση σύνθετων υπολογιστικών και μαθηματικών προβλημάτων σε ταχύτερο χρόνο από τις κοινές γλώσσες προγραμματισμού, όπως η C, η C++ και η Fortan. Ουσιαστικά, αποτελεί ένα ολοκληρωμένο σύστημα ανάλυσης δεδομένων και ανάπτυξης επιστημονικών εφαρμογών. Επιπλέον, λόγω της ευχρηστίας της γλώσσας, που χρησιμοποιεί, το Octave προσφέρει μια ποιοτική γραφική απεικόνιση δεδομένων και αποτελεσμάτων, μέσω του περιβάλλοντός της, όπου όλα τα προβλήματα και οι λύσεις εκφράζονται με έναν απλό και μαθηματικό τρόπο. Μερικές από τις τυπικές χρήσεις του Octave είναι οι παρακάτω.

- Μαθηματικά και υπολογισμοί
- Ανάπτυξη αλγορίθμων
- Απόκτηση δεδομένων
- Μοντελοποίηση και προσομοίωση
- Ανάλυση δεδομένων
- Επιστημονικά και τεχνητά διαγράμματα και γραφήματα
- Ανάπτυξη εφαρμογών, συμπεριλαμβανομένου της δημιουργίας γραφικών διεπαφών χρήστη

Το Octave αποτελεί open-source, το οποίο σημαίνει πως είναι διαθέσιμο προς όλους και δεν χρειάζεται συνδρομή όπως απαιτείται από το Matlab. Ουσιαστικά, το

Octave συνιστά ένα διαδραστικό σύστημα, όπου τα βασικά του δεδομένα αποτελούν οι πίνακες, στους οποίους δεν απαιτείται η αρχική ρύθμιση των διαστάσεών τους. Η ευχέρεια αυτή επιτρέπει την επίλυση σχεδόν όλων των επιστημονικών και τεχνικών υπολογιστικών προβλημάτων, ιδιαίτερα όσων διατυπώνονται συναρτήσει διανυσμάτων και πινάκων.

Το Octave αρχικά αναπτύχθηκε για να προσφέρει εύκολη πρόσβαση στο λογισμικό μαθηματικής και αριθμητικής ανάλυσης, που αναπτύχθηκε στα πλαίσια των μεγάλων προγραμμάτων LINPACK (γραμμική άλγεβρα, γραμμικά συστήματα) και EISPACK (ιδιοτιμές, ιδιοδιανίσματα, φασματικά αναπτύγματα πινάκων). Τον τελευταίο καιρό, το Octave βασίζεται στην εξέλιξη των παραπάνω προγραμμάτων, δηλαδή στις βιβλιοθήκες LAPACK και BLAS, οι οποίες αποτελούν το τελευταία τεχνολογικό επίτευγμα σε θέματα χειρισμού και ανάλυσης πινάκων.

Το Octave εξελίσσεται με το πέρασμα των χρόνων με την συμβολή των χρηστών και των προγραμματιστών της σε θέματα ανάπτυξης διάφορων αλγορίθμων που διευκολύνουν την ανάλυση δεδομένων εξαιτίας της δυνατότητάς του να είναι open-source. Σε πολλά ακαδημαϊκά περιβάλλοντα ανά τον κόσμο το Octave αποτελεί ένα βασικό εκπαιδευτικό εργαλείο για εισαγωγικά και προχωρημένα μαθήματα σε τομείς των μαθηματικών, της μηχανικής και της επιστήμης γενικότερα. Στην βιομηχανία, από την άλλη πλευρά, χρησιμοποιείται η Matlab εξαιτίας του περιβάλλοντος εργασίας και των γραφικών δυνατοτήτων του είναι καλύτερο εργαλείο επιλογής για υψηλή παραγωγικότητα σε τομείς έρευνας, ανάλυσης και ανάπτυξης εφαρμογών από το Octave.

Επιπρόσθετα, το Octave διαθέτει ένα σύνολο πρόσθετων, εξειδικευμένων λύσεων που ονομάζονται toolboxes (εργαλειοθήκες). Τα toolboxes αποτελούν ένα σύνολο συναρτήσεων, οι οποίες επεκτείνουν τις δυνατότητες του Octave για επίλυση πιο σύνθετων και περίπλοκων προβλημάτων ενός πιο συγκεκριμένου τύπου. Χαρακτηριστικά παραδείγματα αποτελούν τα προβλήματα επεξεργασίας σήματος, έλεγχου συστημάτων, νευρωνικών δικτύων, προσομοίωσης δεδομένων και πολλά άλλα. [47]



Εικόνα 46: Το λογότυπο του εργαλείου Octave
(Πηγή: <https://ubunlog.com/ha/gnu-octave-bude-tushen-madadin-matlab/>)

4.3 Μεθοδολογία ενεργειακής αξιολόγησης

4.3.1 Ηλεκτρικά αυτοκίνητα

Αρχικά, οι διαδρομές, που προτείνονται στην παρούσα διπλωματική εργασία, είναι από τα Χανιά μέχρι το Ρέθυμνο, με συνολική απόσταση 63.9 χιλιόμετρα, από τα Χανιά μέχρι το Ηράκλειο, με συνολική απόσταση 142 χιλιόμετρα και τέλος από το Ρέθυμνο μέχρι το Ηράκλειο με συνολική απόσταση 78.1 χιλιόμετρα αντίστοιχα.

Επιπλέον, τα τέσσερα αυτοκίνητα που επιλέχθηκαν για την παρούσα διπλωματική εργασία αποτελούν τα Audi Q4 e-tron 35, Fiat 500e Hatchback, Nissan Leaf και το Mazda MX-30.

Πιο αναλυτικά, το Audi Q4 e-tron 35 διαθέτει μπαταρία των 55kWh, ενώ η διαθέσιμη χωρητικότητα της μπαταρίας για οδήγηση είναι 52kWh με αυτονομία στα 285 χιλιόμετρα. Επιπρόσθετα, το συνολικό του βάρος ως αυτοκίνητο είναι 1965 κιλά και το μέγιστο βάρος που δύναται να μεταφέρει είναι 585 κιλά. Ο συντελεστής οπισθέλκουσας του συγκεκριμένου αυτοκινήτου έχει τιμή 0,26 και η μπροστινή επιφάνεια του αυτοκινήτου είναι συνολικά 3,0437 m². [48]

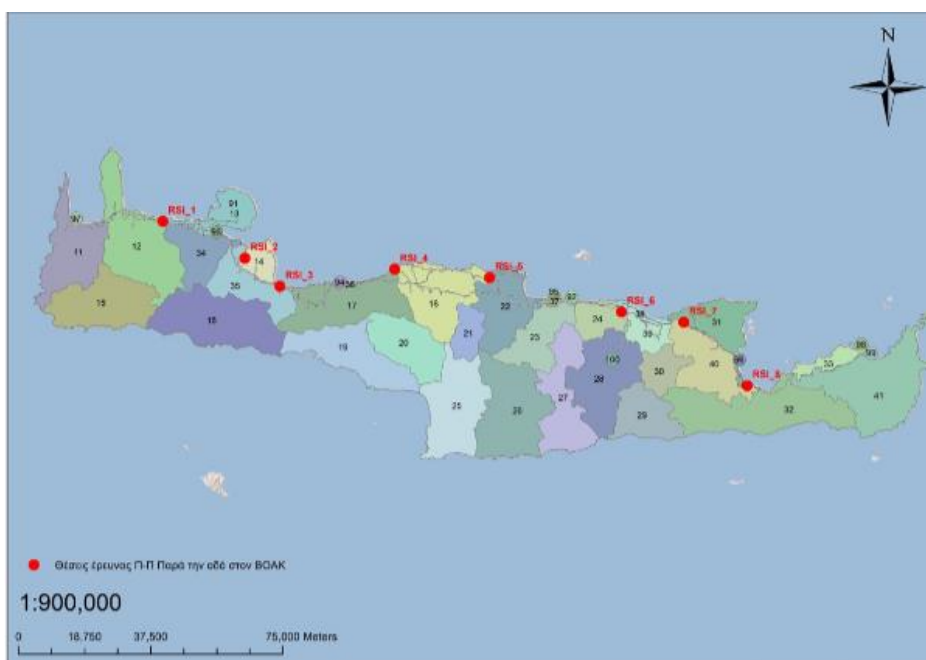
Αντίστοιχα, το Fiat 500e Hatchback διαθέτει μπαταρία των 42kWh με την συνολική χωρητικότητα της μπαταρίας, η οποία προορίζεται για οδήγηση, να είναι 37,3kWh με αυτονομία στα 235 χιλιόμετρα. Επίσης, το συνολικό του βάρος είναι 1365 κιλά και το μέγιστο βάρος που μπορεί να μεταφέρει είναι 400 κιλά. Ο συντελεστής οπισθέλκουσας του Fiat 500e έχει τιμή 0,311 και η μπροστινή επιφάνεια του αυτοκινήτου είναι 2,5699 m². [49]

Από την άλλη πλευρά, το Nissan Leaf διαθέτει μπαταρία συνολικής χωρητικότητας 40kWh εκ των οποίων οι 37kWh χρησιμοποιούνται για την οδήγηση με αυτονομία στα 225 χιλιόμετρα. Το συνολικό βάρος του Nissan Leaf ανέρχεται στα 1580 κιλά, ενώ το μέγιστο φορτίο που μπορεί να μεταφέρει δεν ξεπερνάει τα 490 κιλά. Ο συντελεστής οπισθέλκουσας του Nissan Leaf έχει τιμή 0,32 και η μπροστινή επιφάνεια του αυτοκινήτου είναι 2,7356 m². [50]

Ολοκληρώνοντας, το Mazda MX-30 διαθέτει μπαταρία συνολικής χωρητικότητας 35,5kWh εκ των οποίων οι 30kWh αξιοποιούνται για την οδήγηση με αυτονομία στα 170 χιλιόμετρα. Επιπλέον, το συνολικό βάρος του συγκεκριμένου αυτοκινήτου είναι 1720 κιλά, ενώ το μέγιστο φορτίο που δύναται να μεταφέρει είναι 474 κιλά. Τέλος, ο συντελεστής οπισθέλκουσας του Mazda MX-30 έχει τιμή 0,29 και η μπροστινή επιφάνεια του αυτοκινήτου είναι 2,8182 m². [51]

4.3.2 Ωριαίος κυκλοφοριακός φόρτος

Τα στοιχεία του ωριαίου κυκλοφοριακού φόρτου, που αναφέρονται παρακάτω, προκύπτουν από έρευνα του υπουργείου υποδομών και συγκοινωνιών για τον BOAK για το έτος 2018 και χρησιμοποιούνται μετά από επικοινωνία με την διεύθυνση Δ16 του συγκεκριμένου υπουργείου. Πιο συγκεκριμένα, τα αυτογραφικά μηχανήματα που καταμετρούν τα οχήματα που κινούνται σε μία περιοχή εγκαταστάθηκαν σε 8 σημεία κατά μήκος του BOAK όπως απεικονίζεται στην παρακάτω εικόνα. Τα ονόματα των σταθμών από τα αριστερά προς τα δεξιά είναι τα εξής: RS-1, RS-2, RS-3, RS-4, RS-5, RS-6, RS-7, RS-8.



Εικόνα 47: Σταθμοί καταγραφής κινούμενων οχημάτων

Επιπλέον, στους πίνακες που ακολουθούν παρουσιάζεται το άθροισμα των κυκλοφοριακών φόρτων για τους συγκεκριμένους σταθμούς έρευνας ανά ώρα και ανά κατεύθυνση για την θερινή και χειμερινή περίοδο.

ΩΡΑ	Προς Ανατολή	Προς Δύση	Σύνολο
00:00-01:00	1.171	1.099	2.270
01:00-02:00	718	697	1.416
02:00-03:00	398	387	785
03:00-04:00	266	241	507
04:00-05:00	280	271	551
05:00-06:00	485	476	961
06:00-07:00	1.327	1.169	2.496
07:00-08:00	2.443	2.220	4.662
08:00-09:00	3.092	3.061	6.153
09:00-10:00	3.597	3.954	7.550
10:00-11:00	4.063	4.523	8.586
11:00-12:00	4.217	4.608	8.825
12:00-13:00	3.851	4.046	7.897
13:00-14:00	3.488	3.559	7.047
14:00-15:00	3.520	3.407	6.926
15:00-16:00	3.688	3.705	7.393
16:00-17:00	3.699	3.640	7.339
17:00-18:00	3.863	3.696	7.559
18:00-19:00	4.080	3.714	7.794
19:00-20:00	4.057	3.410	7.466
20:00-21:00	3.311	2.829	6.140
21:00-22:00	2.655	2.259	4.914
22:00-23:00	2.014	1.870	3.884
23:00-24:00	1.649	1.503	3.152
ΣΥΝΟΛΟ	61.931	60.341	122.272

Εικόνα 48: Μετρήσεις οχημάτων στον ΒΟΑΚ για την θερινή περίοδο

ΩΡΑ	Προς Ανατολή	Προς Δύση	Σύνολο
00:00-01:00	240	200	441
01:00-02:00	111	122	233
02:00-03:00	71	67	138
03:00-04:00	60	67	128
04:00-05:00	85	106	191
05:00-06:00	284	240	523
06:00-07:00	765	621	1.386
07:00-08:00	1.700	1.711	3.411
08:00-09:00	2.235	2.024	4.259
09:00-10:00	2.109	1.971	4.080
10:00-11:00	1.874	1.838	3.712
11:00-12:00	1.779	1.647	3.426
12:00-13:00	1.782	1.659	3.441
13:00-14:00	1.811	1.763	3.574
14:00-15:00	1.966	1.881	3.847
15:00-16:00	2.117	2.057	4.175
16:00-17:00	2.057	2.012	4.069
17:00-18:00	1.875	1.830	3.704
18:00-19:00	1.535	1.514	3.048
19:00-20:00	1.172	1.207	2.380
20:00-21:00	974	924	1.898
21:00-22:00	829	776	1.605
22:00-23:00	585	522	1.107
23:00-24:00	395	338	733
ΣΥΝΟΛΟ	28.411	27.099	55.509

Εικόνα 49: Μετρήσεις οχημάτων στον ΒΟΑΚ για την χειμερινή περίοδο

Στον παρακάτω πίνακα απεικονίζεται η μέση ημερήσια κυκλοφορία και για τις δύο κατευθύνσεις ανά σταθμό και μήνα μετρήσεων.

Θέση Έρευνας	Ιούλιος 2018	Αύγουστος 2018	Σεπτέμβριος 2018	Νοέμβριος 2018
ATC-1 (RSI-1)	22.244	22.741	18.892	11.500
ATC-2A (RSI-2)	17.298	18.001	14.980	7.990
ATC-3 (RSI-3)	17.492	18.107	15.164	7.136
ATC-4 (RSI-4)	18.335	18.218	16.438	8.069
ATC-5 (RSI-5)	14.432	14.492	12.681	5.804
ATC-6 (RSI-6)	22.102	22.652	19.694	9.772
ATC-7 (RSI-7)	13.890	14.235	12.317	5.596
ATC-8 (RSI-8)	8.030	8.489	6.896	3.479

Εικόνα 50: Μέση ημερήσια κυκλοφορία και στις 2 κατευθύνσεις ανά σταθμό και μήνα

Ωστόσο, στην παρούσα διπλωματική εργασία δεν χρειάζονται τα δεδομένα και των 8 σταθμών μέτρησης αλλά μόνο δύο και πιο συγκεκριμένα οι RS-2 και RS-4 γιατί εξετάζονται οι διαδρομές Χανιά-Ρέθυμνο και Χανιά-Ηράκλειο. Για τον υπολογισμό του αριθμού των οχημάτων, που κινούνται σε κάθε σταθμό, απαιτείται η εύρεση του ποσοστού που έχει ο κάθε σταθμός σε σχέση με το σύνολο των οχημάτων. Για την θερινή περίοδο λαμβάνεται ως μήνας αναφοράς ο Αύγουστος ενώ η χειμερινή περίοδος δεν λαμβάνεται υπόψιν εξαιτίας της τεράστιας διαφοράς του

αριθμού των οχημάτων με την θερινή περίοδο. Για παράδειγμα, από τον σταθμό RS-2 περνάνε 18001 οχήματα, ενώ μέσα στην μέρα περνάνε συνολικά 136935 από όλους τους σταθμούς. Επομένως, από τον σταθμό RS-2 διέρχεται το 13,1% των συνολικών οχημάτων. Για τον υπολογισμό των οχημάτων που διέρχονται από τον σταθμό κάθε ώρα απαιτείται ο πολλαπλασιασμός του 13,1% με το σύνολο των οχημάτων που κινούνται εκείνη την ώρα προς μία κατεύθυνση.

Στους παρακάτω πίνακες αναφέρεται ο αριθμός των οχημάτων που κινούνται στους σταθμούς RS-2 και RS-4.

Ωρα	Προς Ανατολή
00:00-01:00	153
01:00-02:00	94
02:00-03:00	52
03:00-04:00	35
04:00-05:00	37
05:00-06:00	63
06:00-07:00	173
07:00-08:00	320
08:00-09:00	405
09:00-10:00	471
10:00-11:00	532
11:00-12:00	552
12:00-13:00	504
13:00-14:00	457
14:00-15:00	461
15:00-16:00	483
16:00-17:00	484
17:00-18:00	506
18:00-19:00	534
19:00-20:00	531
20:00-21:00	433
21:00-22:00	347
22:00-23:00	263
23:00-24:00	216

Πίνακας 4: Ωριαίος κυκλοφοριακός φόρτος για τον σταθμό RS-2

Ωρα	Προς Ανατολή
00:00-01:00	155
01:00-02:00	95
02:00-03:00	52
03:00-04:00	35
04:00-05:00	37
05:00-06:00	64
06:00-07:00	176
07:00-08:00	324
08:00-09:00	411

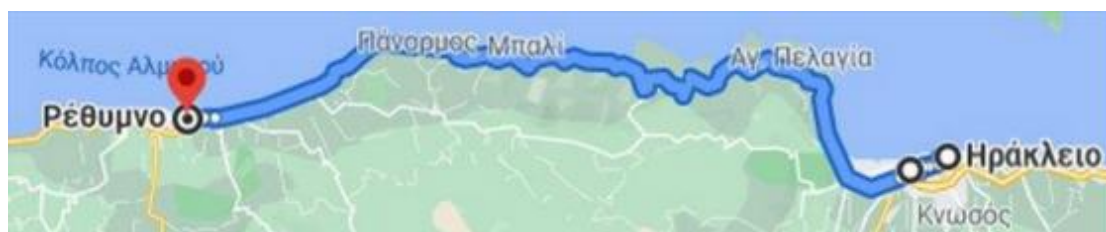
09:00-10:00	478
10:00-11:00	540
11:00-12:00	560
12:00-13:00	512
13:00-14:00	463
14:00-15:00	468
15:00-16:00	490
16:00-17:00	491
17:00-18:00	513
18:00-19:00	542
19:00-20:00	539
20:00-21:00	440
21:00-22:00	353
22:00-23:00	267
23:00-24:00	219

Πίνακας 5: Ωριαίος κυκλοφοριακός φόρτος για τον σταθμό RS-4

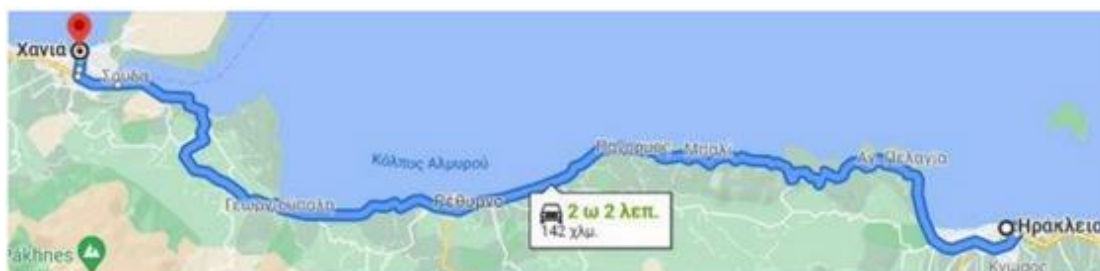
Για τον υπολογισμό των συνολικών οχημάτων, που πραγματοποιούν την διαδρομή Χανιά-Ρέθυμνο κάθε ώρα, απαιτείται η λύση της εξίσωσης $RS2-A+R=RS4$. Πιο συγκεκριμένα, το RS2 αποτελεί τον αριθμό των οχημάτων που κινούνται στον σταθμό RS-2, δηλαδή τον αριθμό των οχημάτων που φεύγουν από τα Χανιά, το A αποτελεί τον αριθμό των οχημάτων που πραγματοποιούν την διαδρομή Χανιά-Ρέθυμνο, το R αποτελεί τον αριθμό των οχημάτων που φεύγουν από το Ρέθυμνο για την πόλη του Ηρακλείου και τέλος το RS4 αποτελεί τον αριθμό των οχημάτων που διέρχονται από τον σταθμό RS-4.



Εικόνα 51: Η μεταβλητή A αντιπροσωπεύει τον αριθμό των οχημάτων που πραγματοποιούν τη διαδρομή Χανιά-Ρέθυμνο



Εικόνα 52: Η μεταβλητή R αντιπροσωπεύει τον αριθμό των οχημάτων που πραγματοποιούν τη διαδρομή Ρέθυμνο-Ηράκλειο



Εικόνα 53: Το αποτέλεσμα της αφαίρεσης RS2-A αντιπροσωπεύει τον αριθμό των οχημάτων που πραγματοποιούν τη διαδρομή Χανιά-Ηράκλειο

Το R θεωρείται πως έχει την μισή τιμή του συνόλου των αυτοκινήτων που πραγματοποιούν την διαδρομή Χανιά-Ηράκλειο. Σύμφωνα με τα παραπάνω προκύπτει ο παρακάτω πίνακας, που απεικονίζει τον αριθμό των οχημάτων που πραγματοποιούν την διαδρομή Χανιά-Ρέθυμνο.

Ώρα	Προς Ανατολή
00:00-01:00	50
01:00-02:00	31
02:00-03:00	18
03:00-04:00	12
04:00-05:00	13
05:00-06:00	21
06:00-07:00	56
07:00-08:00	104
08:00-09:00	131
09:00-10:00	153
10:00-11:00	172
11:00-12:00	179
12:00-13:00	163
13:00-14:00	149
14:00-15:00	149
15:00-16:00	157
16:00-17:00	157
17:00-18:00	164
18:00-19:00	173
19:00-20:00	172
20:00-21:00	140
21:00-22:00	112
22:00-23:00	85
23:00-24:00	70

Πίνακας 6: Ωριαίος κυκλοφοριακός φόρτος για την διαδρομή Χανιά-Ρέθυμνο

Από την άλλη πλευρά, ο αριθμός των οχημάτων για τους οποίους πραγματοποιείται ενεργειακή αξιολόγηση για την διαδρομή Χανιά-Ηράκλειο, προκύπτει από τον

υπολογισμό της αφαίρεσης του αριθμού των οχημάτων που κινούνται στον σταθμό RS-2 και των οχημάτων που μένουν στο Ρέθυμνο. Επομένως, προκύπτει ο παρακάτω πίνακας.

Ωρα	Προς Ανατολή
00:00-01:00	103
01:00-02:00	63
02:00-03:00	34
03:00-04:00	23
04:00-05:00	24
05:00-06:00	42
06:00-07:00	117
07:00-08:00	216
08:00-09:00	274
09:00-10:00	318
10:00-11:00	360
11:00-12:00	373
12:00-13:00	341
13:00-14:00	308
14:00-15:00	312
15:00-16:00	326
16:00-17:00	327
17:00-18:00	342
18:00-19:00	361
19:00-20:00	359
20:00-21:00	293
21:00-22:00	235
22:00-23:00	178
23:00-24:00	146

Πίνακας 7: Ωριαίος κυκλοφοριακός φόρτος για την διαδρομή Χανιά-Ηράκλειο

Τέλος, ο αριθμός των οχημάτων για τους οποίους πραγματοποιείται ενεργειακή αξιολόγηση για την διαδρομή Ρέθυμνο-Ηράκλειο, προκύπτει από τον υπολογισμό της εξίσωσης $R = (RS2-A)/2$, δεδομένου ότι υπάρχει η παραδοχή ότι τα οχήματα που φεύγουν από το Ρέθυμνο είναι τα μισά σε σχέση με αυτά που φεύγουν από τα Χανιά με προορισμό το Ηράκλειο. Σύμφωνα με τα παραπάνω προκύπτει ο παρακάτω πίνακας.

Ωρα	Προς Ανατολή
00:00-01:00	52
01:00-02:00	32
02:00-03:00	18
03:00-04:00	12
04:00-05:00	13
05:00-06:00	22
06:00-07:00	59
07:00-08:00	108
08:00-09:00	137
09:00-10:00	160

10:00-11:00	180
11:00-12:00	187
12:00-13:00	171
13:00-14:00	155
14:00-15:00	156
15:00-16:00	164
16:00-17:00	164
17:00-18:00	171
18:00-19:00	181
19:00-20:00	180
20:00-21:00	147
21:00-22:00	118
22:00-23:00	89
23:00-24:00	73

Πίνακας 8: Ωριαίος κυκλοφοριακός φόρτος για την διαδρομή Ρέθυμνο-Ηράκλειο

4.4.3 Αξιοποίηση εργαλείου Octave

Για την δημιουργία ενός πίνακα που περιλαμβάνει τις ταχύτητες κάθε ώρας αξιοποιείται η συνάρτηση `gendist`. Η συνάρτηση `gendist` δημοσιεύτηκε από τους Tristan Ursell και Derek O'Connor το 2012 στο forum της Matlab, αλλά μπορεί να αξιοποιηθεί και στο Octave. Η αρχή λειτουργίας του έγκειται στην παραγωγή τυχαίων αριθμών σύμφωνα με μία διακριτή κατανομή πιθανότητας. Πιο συγκεκριμένα, η συνάρτηση `gendist(P,N,M)` δέχεται ως είσοδο έναν πίνακα P , ο οποίος περιλαμβάνει τιμές που σχηματίζουν μία διακριτή κατανομή πιθανότητας για κάθε θέση του P . Η συνάρτηση εξάγει έναν $N \times M$ πίνακα, που περιλαμβάνει ακέραιους αριθμούς που αντιστοιχούν στις θέσεις του P και οι οποίοι επιλέγονται τυχαία με βάση την κατανομή. Αξίζει να αναφερθεί πως, ο πίνακας P θα κανονικοποιηθεί σε περίπτωση που δεν είναι ήδη κανονικοποιημένος και πως τα N και M πρέπει να έχουν τιμές μεγαλύτερες ή ίσες του 1. [52]

Για παράδειγμα, αν το P έχει τιμές $[0.2 \ 0.4 \ 0.4]$ (Σημείωση: το άθροισμα των τιμών του P ισοδυναμούν με 1) τότε το αποτέλεσμα της συνάρτησης `gendist` έχει τιμές μόνο 1, 2 και 3, οι οποίες αντιστοιχούν στις θέσεις του P . Αν για παράδειγμα η συνάρτηση κληθεί ως $T = \text{gendist}(P,1,10)$ τότε το T θα περιέχει κατά μέσο όρο δύο φορές το 1, τέσσερις φορές το 2 και τέσσερις φορές το 3, σύμφωνα με τις τιμές του P , με ένα πιθανό αποτέλεσμα να είναι $T = [2 \ 2 \ 2 \ 3 \ 2 \ 3 \ 1 \ 3 \ 1 \ 3]$. Για την δημιουργία ενός πίνακα X με μέγεθος 1×10 (μέγεθος T), που περιλαμβάνει τα στοιχεία του πίνακα P αρκεί η ανάθεση με την εντολή $X = P(T)$.

Στην παρούσα διπλωματική εργασία δημιουργείται ένας πίνακας που περιλαμβάνει την πιθανότητα εμφάνισης της εκάστοτε ταχύτητας που έχει βρεθεί από τα βίντεο, όπως φαίνεται στον παρακάτω πίνακα.

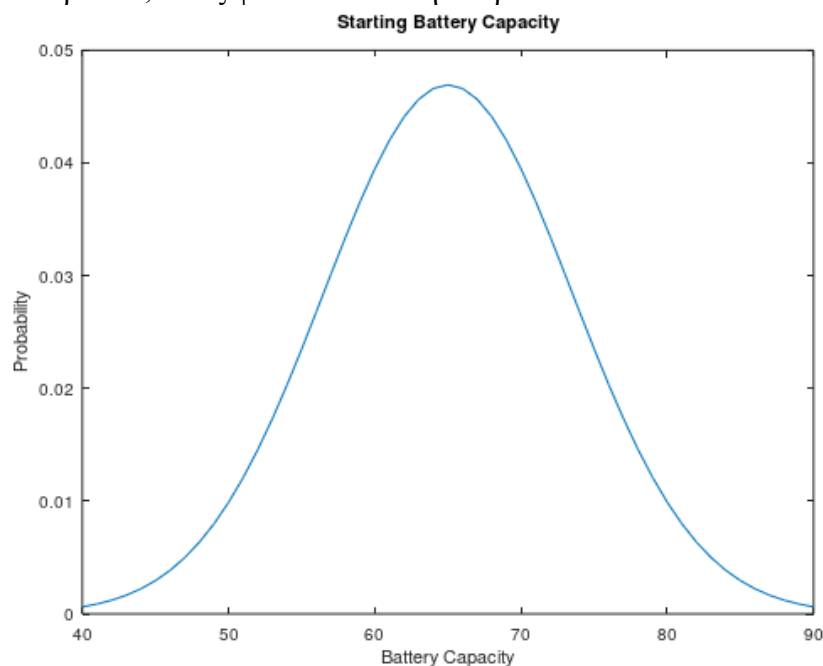
Πιθανότητα Εμφάνισης	Ταχύτητα
0.9%	32
2.9%	38
14.56%	44
20.24%	50
19.75 %	56

11.11%	62
5.67%	68
3.7%	76
9.62%	82
4.69%	88
3.7%	94
0.98%	100
1.23%	106
0.49%	114
0.24%	126

Πίνακας 9: Πιθανότητα εμφάνισης της εκάστοτε ταχύτητας

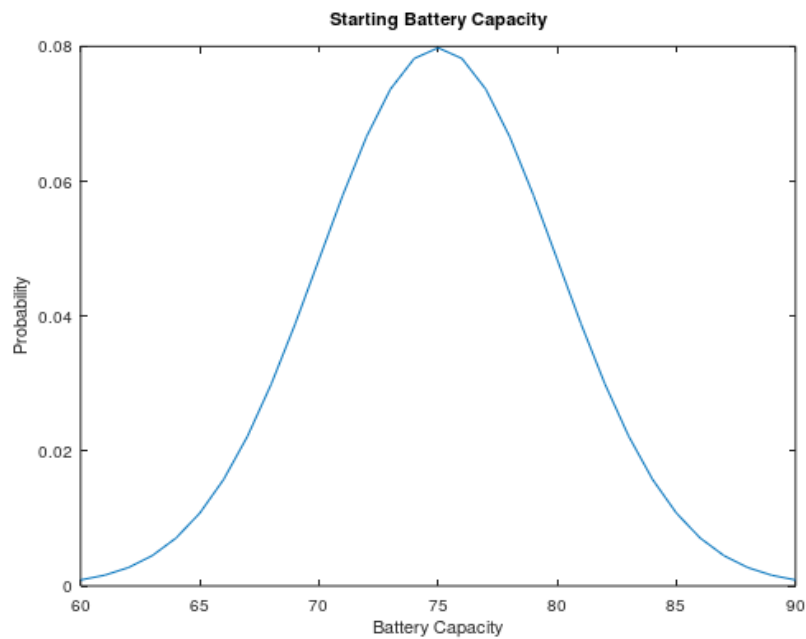
Με την χρήση της συνάρτησης gendist, λοιπόν, παράγονται πίνακες, που περιλαμβάνουν τυχαίες ταχύτητες για όλες τις ώρες της ημέρας και για κάθε ηλεκτρικό αυτοκίνητο. Το μέγεθος των πινάκων αυτών εξαρτάται από τον αριθμό του συνόλου των αυτοκινήτων που κινούνται την εκάστοτε ώρα.

Στην συνέχεια, γίνεται μία υπόθεση ότι όλα τα αυτοκίνητα οποιαδήποτε ώρας και με οποιοδήποτε προορισμό ξεκινάνε από τα Χανιά με ένα τυχαίο ποσοστό φόρτισης της μπαταρίας τους. Η επιλογή της τιμής, που ανατίθεται σε κάθε αυτοκίνητο, προκύπτει από μία γκαουσιανή κατανομή, η οποία είναι διαφορετική για κάθε προορισμό. Πιο συγκεκριμένα, για την διαδρομή Χανιά-Ρέθυμνο οι τιμές του αρχικού ποσοστού φόρτισης της μπαταρίας κυμαίνονται μεταξύ 40% και 90% με μέση τιμή το 65% και διασπορά 8.5, όπως φαίνεται από την παρακάτω εικόνα.



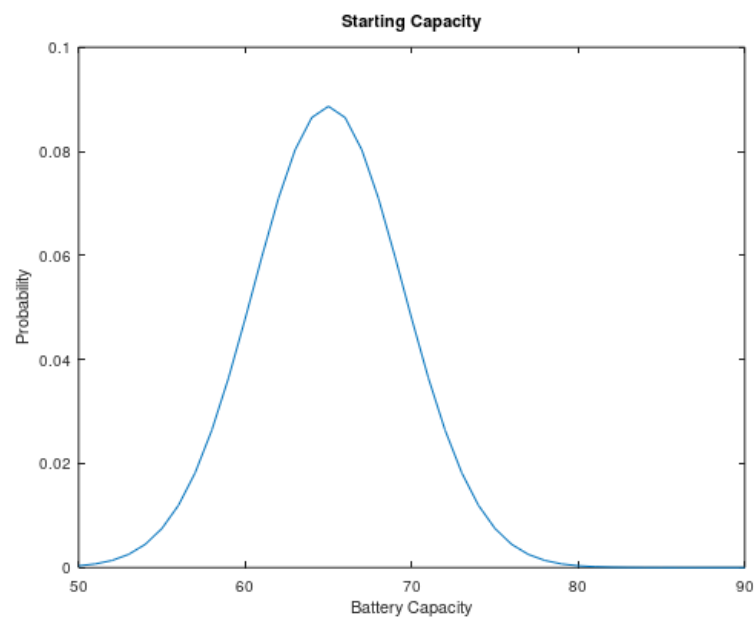
Εικόνα 54: Κατανομή αρχικού ποσοστού φόρτισης της μπαταρίας για τα αυτοκίνητα της διαδρομής Χανιά-Ρέθυμνο

Από την άλλη πλευρά, για την διαδρομή Χανιά-Ηράκλειο οι τιμές του αρχικού ποσοστού φόρτισης της μπαταρίας κυμαίνονται μεταξύ 60% και 90% με μέση τιμή 75% και διασπορά 5, όπως φαίνεται από την παρακάτω εικόνα.



Εικόνα 55: Κατανομή αρχικού ποσοστού φόρτισης της μπαταρίας για τα αυτοκίνητα της διαδρομής Χανιά-Ηράκλειο

Τέλος, για την διαδρομή Ρέθυμνο-Ηράκλειο οι τιμές του αρχικού ποσοστού φόρτισης της μπαταρίας κυμαίνονται μεταξύ 50% και 90% με μέση τιμή 65% και διασπορά 4.5, όπως φαίνεται από την παρακάτω εικόνα.



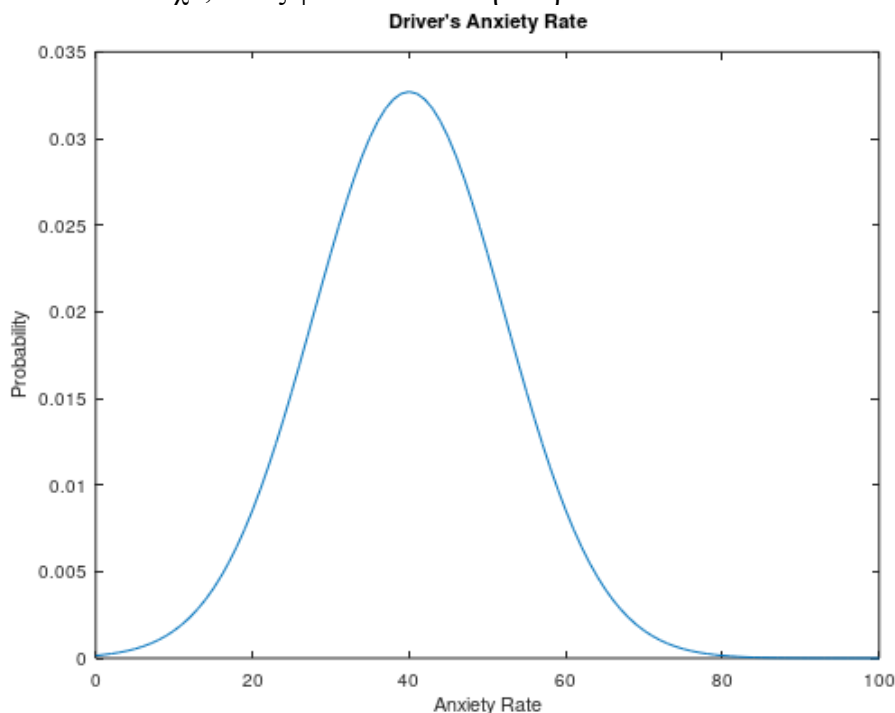
Εικόνα 56: Κατανομή αρχικού ποσοστού φόρτισης της μπαταρίας για τα αυτοκίνητα της διαδρομής Ρέθυμνο-Ηράκλειο

Ο λόγος, για τον οποίο υπάρχει μια διαφορά ως προς τις τιμές του ποσοστού φόρτισης της μπαταρίας, βρίσκεται στην μεγάλη χιλιομετρική διαφορά των διαδρομών. Έτσι λοιπόν, τα αυτοκίνητα που διανύουν την διαδρομή Χανιά-Ηράκλειο

εκκινούν με μεγαλύτερο ποσοστό φόρτισης της μπαταρίας τους από τα αυτοκίνητα που διανύουν τις διαδρομές Χανιά-Ρέθυμνο και Ρέθυμνο-Ηράκλειο.

Με την βοήθεια της συνάρτησης `gendist` δημιουργούνται πίνακες, οι οποίοι αναλογούν για κάθε ώρα της ημέρας και για κάθε αυτοκίνητο. Το περιεχόμενο των συγκεκριμένων πινάκων περιλαμβάνει τυχαίες τιμές που αναλογούν στο αρχικό ποσοστό φόρτισης της μπαταρίας, σύμφωνα με την γκαουσιανή κατανομή που αναφέρεται παραπάνω για κάθε προορισμό. Το μέγεθος των πινάκων αυτών εξαρτάται από τον αριθμό του συνόλου των αυτοκινήτων που κινούνται την εκάστοτε ώρα.

Ακολουθώς, για κάθε οδηγό όλων των αυτοκινήτων, για κάθε ώρα της ημέρας και για οποιοδήποτε προορισμό, ανατίθεται μία τιμή, που αναλογεί στο ποσοστό του άγχους που έχει ο εκάστοτε οδηγός για το ποσοστό φόρτισης της μπαταρίας του αυτοκινήτου που οδηγεί. Η επιλογή της τιμής, που ανατίθεται σε κάθε οδηγό, προκύπτει από μία γκαουσιανή κατανομή και κυμαίνονται μεταξύ 0% και 100%. Το 0% αναλογεί σε μηδενικό άγχος του οδηγού όσον αφορά το ποσοστό φόρτισης της μπαταρίας του αυτοκινήτου, ενώ αντίθετα το 100% αναλογεί σε πάρα πολύ μεγάλο άγχος. Επιπλέον, η μέση τιμή, που προτείνεται στην παρούσα διπλωματική εργασία για την συγκεκριμένη γκαουσιανή κατανομή, είναι 40%, ενώ η διασπορά λαμβάνει την τιμή 12.2 αντίστοιχα, όπως φαίνεται από την παρακάτω εικόνα.



Εικόνα 57: Κατανομή ποσοστού άγχους του οδηγού για το ποσοστό φόρτισης της μπαταρίας του αυτοκινήτου

Η δημιουργία των πινάκων, που περιλαμβάνουν το ποσοστό του άγχους των οδηγών για κάθε ώρα της ημέρας και για όλα τα αυτοκίνητα, υλοποιείται με την βοήθεια της συνάρτησης `gendist`. Το μέγεθος των πινάκων αυτών εξαρτάται από τον αριθμό του συνόλου των αυτοκινήτων που κινούνται την εκάστοτε ώρα.

Όπως έχει αναφερθεί ήδη, χρησιμοποιείται ένα σύστημα ασαφούς λογικής (Fuzzy Inference System), το οποίο εξάγει την πιθανότητα αναζήτησης ενός σταθμού φόρτισης από τον οδηγό ενός ηλεκτρικού αυτοκινήτου.

Πρωτίστως, τα χαρακτηριστικά του συστήματος fuzzy παραμένουν τα προκαθορισμένα (default), που προτείνει το Octave. Πιο αναλυτικά, ο τύπος του συστήματος, που αξιοποιείται, είναι ο mamdani, ο οποίος πήρε το όνομά του από τον Ebrahim Mamdani, που ήταν από τους πρώτους που ασχολήθηκε με την ασαφή λογική. Ουσιαστικά, πρόκειται για ένα σύστημα που έχει την μορφή “if x is A then y is B” με τις εξόδους του να αποτελούν ασαφή σύνολα. Στην συνέχεια, οι μέθοδοι and_method και or_method χρησιμοποιούνται στους κανόνες που συντάσσονται και περιγράφονται παρακάτω. Αναλυτικότερα, αν ένας κανόνας έχει παραπάνω από μία είσοδο τότε με την βοήθεια των τελεστών AND και OR προκύπτει ένας αριθμός που αντιπροσωπεύει το αποτέλεσμα της υπόθεσης του κανόνα. Στην περίπτωση που εφαρμόζεται ο τελεστής AND και η επιλογή της μεθόδου and_method είναι η ‘min’, τότε ο αριθμός που αποδίδεται είναι ο μικρότερος από τις τιμές εισόδου, που εκφράζουν τον κανόνα. Από την άλλη πλευρά, αν εφαρμόζεται ο τελεστής OR και η επιλογή της μεθόδου or_method είναι η ‘max’, τότε ο αριθμός που αποδίδεται είναι ο μεγαλύτερος από τις τιμές εισόδου, που εκφράζουν τον κανόνα. Επίσης, η μέθοδος imp_method (Implication Method) αξιοποιείται για τον υπολογισμό του επακόλουθου ασαφούς συνόλου. Η προκαθορισμένη επιλογή, που προτείνεται για το σύστημα mamdani, είναι η ‘min’. Επιπλέον, η μέθοδος agg_method (Aggregation Method) αποτελεί την διαδικασία κατά την οποία τα ασαφή σύνολα που αντιπροσωπεύουν τις εξόδους κάθε κανόνα συνδυάζονται σε ένα ενιαίο ασαφές σύνολο. Η προκαθορισμένη επιλογή, που προτείνεται για το σύστημα mamdani, είναι η ‘max’. Ολοκληρώνοντας, η μέθοδος defuzz_method (Defuzzification method) αποφασίζει την τιμή εξόδου του συστήματος δεδομένου ενός συνόλου, που προέρχεται από την μέθοδο agg_method. Η προκαθορισμένη επιλογή, που προτείνεται για το σύστημα mamdani, είναι η ‘centroid’. [53]

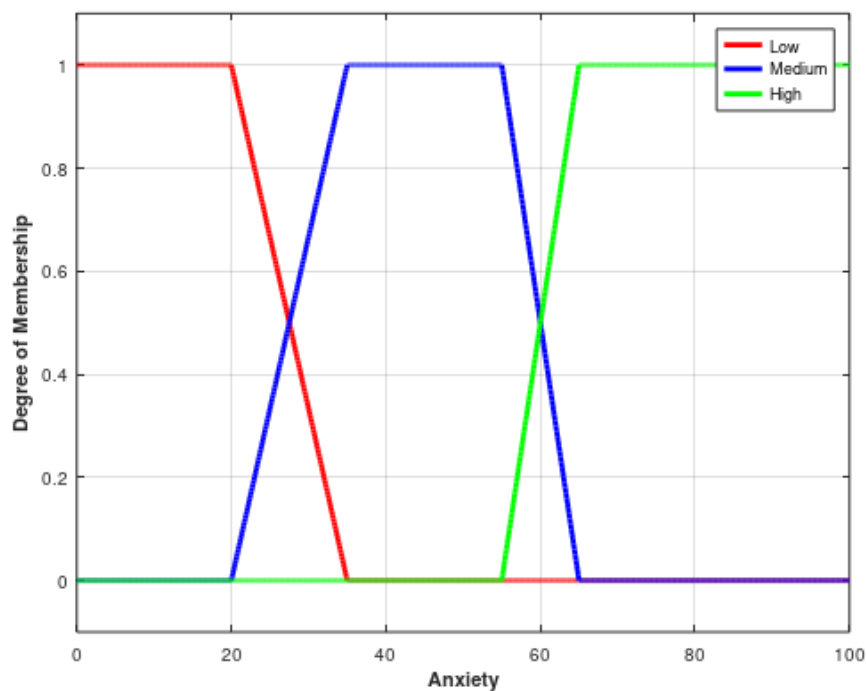
Οι είσοδοι του συστήματος αποτελούνται από το ποσοστό άγχους του εκάστοτε οδηγού, το οποίο παραμένει σταθερό για όλη τη διάρκεια της διαδρομής, και το ποσοστό φόρτισης της μπαταρίας του αυτοκινήτου που οδηγεί, το οποίο όπως είναι αναμενόμενο μειώνεται.

Πιο αναλυτικά, το ποσοστό του άγχους χωρίζεται σε τρία επίπεδα, τα οποία περιγράφονται από τον παρακάτω πίνακα.

Ποσοστό άγχους	Χαρακτηρισμός επιπέδων
0%-35%	Low
20%-65%	Medium
55%-100%	High

Πίνακας 10: Τιμές και χαρακτηρισμοί των επιπέδων του ποσοστού άγχους

Η συνάρτηση μέλους (membership function), που επιλέγεται στην παρούσα διπλωματική εργασία για την συγκεκριμένη είσοδο, αποτελείται από την trapmf, η οποία μετατρέπει τις τιμές του κάθε επιπέδου σε μορφή τραπεζίου, όπως φαίνεται από την παρακάτω εικόνα.



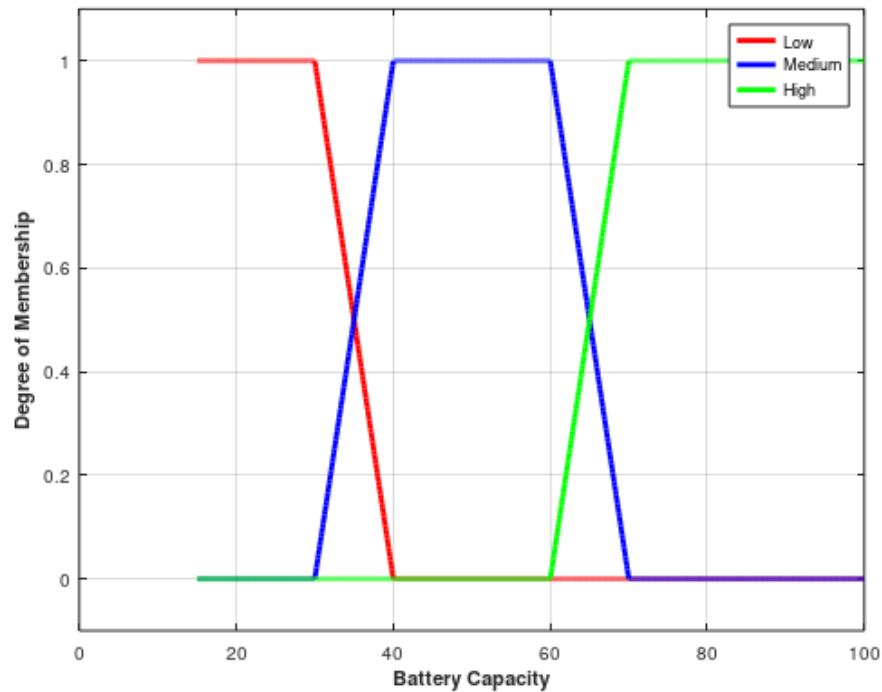
Εικόνα 58: Ποσοστό άγχους ως είσοδο στο σύστημα fuzzy

Αντίστοιχα, η δεύτερη είσοδος του συστήματος fuzzy, που αποτελεί το ποσοστό φόρτισης της μπαταρίας του αυτοκινήτου, χωρίζεται σε τρία επίπεδα, τα οποία παρουσιάζονται στον παρακάτω πίνακα.

Ποσοστό φόρτισης της μπαταρίας	Χαρακτηρισμός επιπέδων
15%-40%	Low
30%-70%	Medium
60%-100%	High

Πίνακας 11: Τιμές και χαρακτηρισμοί των επιπέδων του ποσοστού φόρτισης της μπαταρίας των αυτοκινήτων

Επιπλέον, η συνάρτηση μέλους (membership function), που επιλέγεται για την συγκεκριμένη είσοδο, είναι εκ νέου η trapmf, η οποία μετατρέπει τις τιμές του εκάστοτε επιπέδου σε μορφή τραπεζίου, όπως απεικονίζεται στην παρακάτω εικόνα.



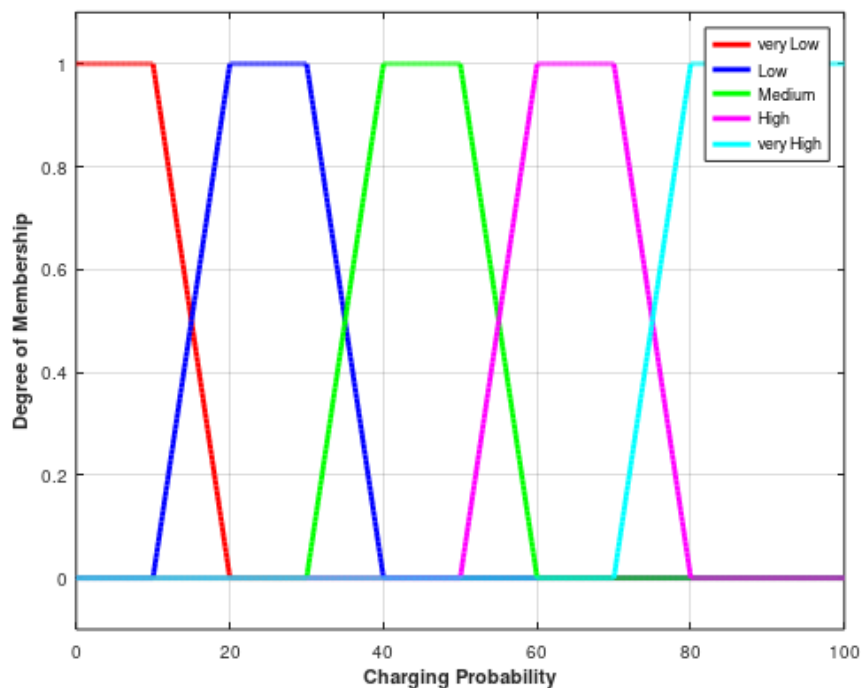
Εικόνα 59: Ποσοστό φόρτισης της μπαταρίας ως είσοδο στο σύστημα fuzzy

Ολοκληρώνοντας, η έξοδος του συστήματος fuzzy, η οποία αποτελεί την πιθανότητα ένας οδηγός να αναζητάει σταθμό φόρτισης για την μπαταρία του αυτοκινήτου του, χωρίζεται σε πέντε επίπεδα, όπως φαίνεται από τον παρακάτω πίνακα.

Πιθανότητα αναζήτησης σταθμού φόρτισης	Χαρακτηρισμός επιπέδων
0%-20%	Very Low
10%-40%	Low
30%-60%	Medium
50%-80%	High
70%-100%	Very High

Πίνακας 12: Τιμές και χαρακτηρισμοί των επιπέδων αναζήτησης σταθμού φόρτισης

Επιπρόσθετα, η συνάρτηση μέλους (membership function), που επιλέγεται για την έξοδο του συστήματος fuzzy, είναι επίσης η trapmf, η οποία μετατρέπει τις τιμές του εκάστοτε επιπέδου σε μορφή τραπεζίου, όπως απεικονίζεται στην παρακάτω εικόνα.



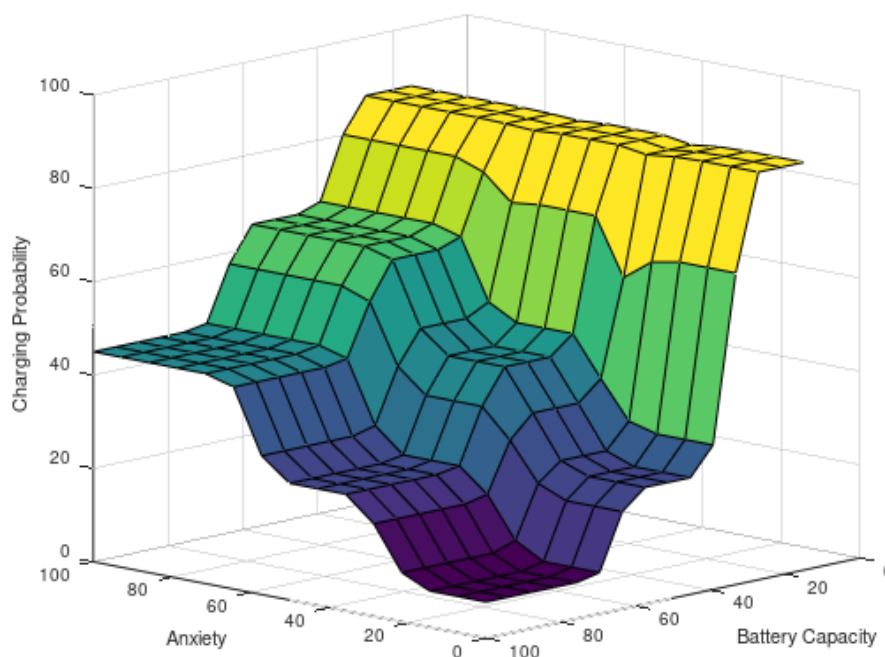
Εικόνα 60: Έξοδος συστήματος fuzzy

Επιπλέον, για την σωστή λειτουργία του συστήματος απαιτείται η σύσταση κανόνων για όλα τα επίπεδα των δύο εισόδων, που αντιστοιχούν σε επίπεδα της εξόδου. Η ύπαρξη τριών επιπέδων στις δύο εισόδους αντιστοιχεί σε τουλάχιστον εννέα κανόνες, που συνδέονται με τα πέντε επίπεδα της εξόδου. Επιπρόσθετα, δίνεται η δυνατότητα αλλαγής της τιμής του βάρους του εκάστοτε κανόνα, με τιμές που κυμαίνονται μεταξύ του 0 και 1, για την περαιτέρω έμφασή τους στο σύστημα αναλόγως της τιμής που δέχονται. Στην παρούσα διπλωματική εργασία επιλέχθηκε η ίδια τιμή για τα βάρη κάθε κανόνα, η οποία είναι το 1, ενώ επίσης συντάχθηκαν εννέα κανόνες, οι οποίοι είναι οι εξής:

1. If (Anxiety is Low) and (Battery Capacity is Low), then (Charging Probability is very High) (1)
2. If (Anxiety is Low) and (Battery Capacity is Medium), then (Charging Probability is Low) (1)
3. If (Anxiety is Low) and (Battery Capacity is High), then (Charging Probability is very Low) (1)
4. If (Anxiety is Medium) and (Battery Capacity is Low), then (Charging Probability is very High) (1)
5. If (Anxiety is Medium) and (Battery Capacity is Medium), then (Charging Probability is Medium) (1)
6. If (Anxiety is Medium) and (Battery Capacity is High), then (Charging Probability is Low) (1)
7. If (Anxiety is High) and (Battery Capacity is Low), then (Charging Probability is very High) (1)
8. If (Anxiety is High) and (Battery Capacity is Medium), then (Charging Probability is High) (1)

9. If (Anxiety is High) and (Battery Capacity is High), then (Charging Probability is Medium) (1)

Σύμφωνα, λοιπόν, με τα δεδομένα των εισόδων, τα δεδομένα της εξόδου, αλλά και τους κανόνες, που αναγράφονται παραπάνω, το σύστημα fuzzy δημιουργεί μία σχέση μεταξύ των τιμών εισόδου και εξόδου. Στην παρακάτω εικόνα απεικονίζεται η έξοδος του συστήματος fuzzy σε συνάρτηση με το ποσοστό του άγχους και το ποσοστό φόρτισης της μπαταρίας.

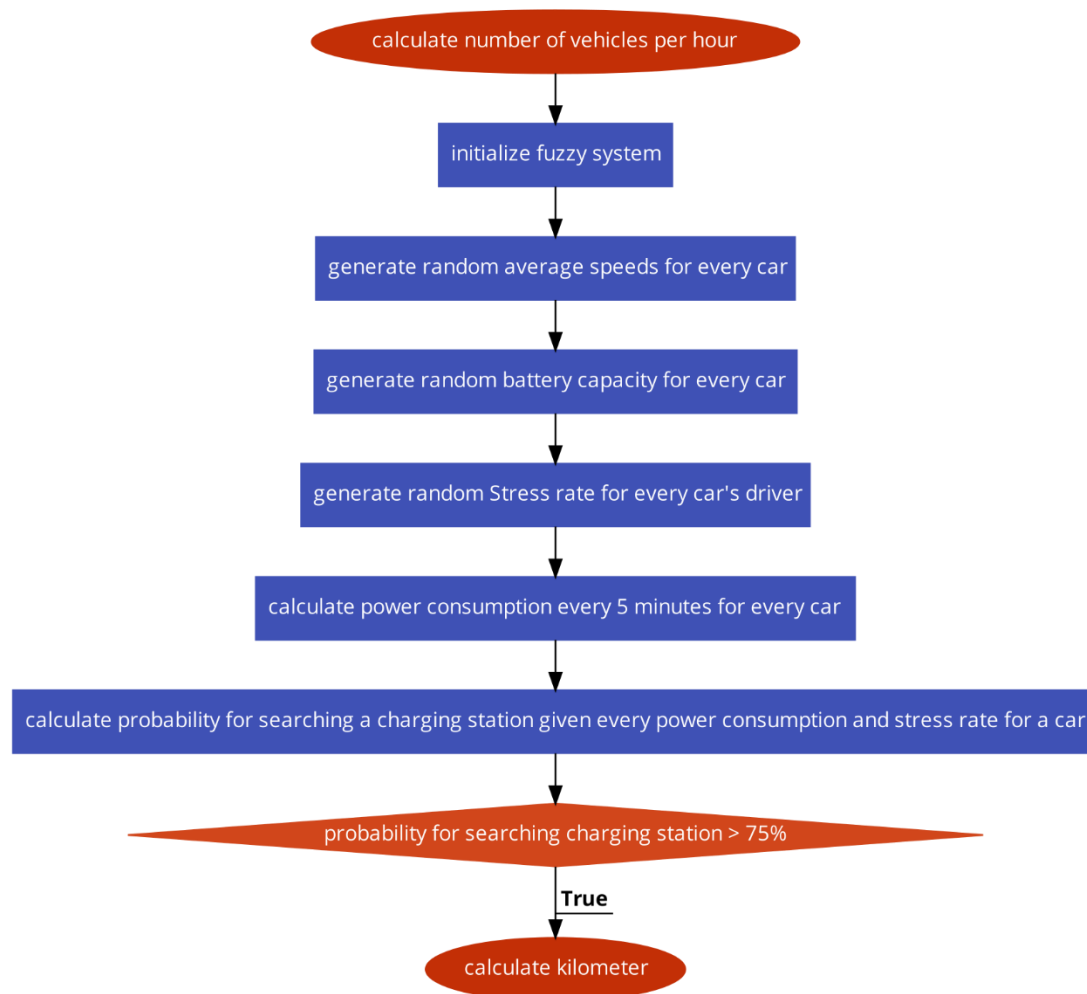


Εικόνα 61: Η έξοδος του συστήματος fuzzy συναρτήσει των δύο εισόδων

Χρησιμοποιώντας, λοιπόν, τις εξισώσεις της παραγράφου 2.10 υπολογίζεται η ενεργειακή κατανάλωση για όλες τις ώρες της ημέρας και για κάθε αυτοκίνητο. Πιο συγκεκριμένα, η τιμή του συντελεστή αντίστασης κύλισης είναι 0,02 και η πυκνότητα του αέρα δέχεται την τιμή 1,184 που αντιστοιχεί σε 25°C. Επιπλέον, για την διαδρομή Χανιά-Ρέθυμνο οι τιμές της ταχύτητας, που αντιστοιχούν για κάθε αυτοκίνητο και θεωρούνται σταθερές μέχρι την άφιξη στον προορισμό, κυμαίνονται μεταξύ των 50 km/h και 76 km/h του πίνακα 10. Αντίθετα, οι ταχύτητες για την διαδρομή Ρέθυμνο-Ηράκλειο, που αντιστοιχούν για κάθε αυτοκίνητο και θεωρούνται σταθερές μέχρι την άφιξη στον προορισμό, κυμαίνονται μεταξύ των 50 km/h και 106 km/h. Τέλος, οι ταχύτητες για την διαδρομή Χανιά-Ηράκλειο, που αντιστοιχούν για κάθε αυτοκίνητο και θεωρούνται σταθερές μέχρι την άφιξη στον προορισμό, αποτελούν ένα συνδυασμό των παραπάνω, δηλαδή η μέση ταχύτητα για την διαδρομή Χανιά-Ρέθυμνο κυμαίνεται μεταξύ των 50 km/h έως 76 km/h και για την διαδρομή Ρέθυμνο-Ηράκλειο η μέση ταχύτητα κυμαίνεται μεταξύ 50 km/h έως 106 km/h.

Εφόσον έχει υπολογιστεί η συνολική ισχύς, που καταναλώνεται μέχρι την άφιξη στον τελικό προορισμό, εκτιμάται η πτώση του ποσοστού φόρτισης της μπαταρίας του αυτοκινήτου ανά 5 λεπτά. Στην συνέχεια, εισάγεται το ποσοστό του άγχους του

οδηγού και το ποσοστό φόρτισης της μπαταρίας ανά 5 λεπτά στο σύστημα fuzzy και εξάγεται η πιθανότητα ο οδηγός να αναζητάει σταθμό φόρτισης. Αν η τιμή εξόδου είναι μεγαλύτερη ή ίση του 75% τότε θεωρείται ότι ο οδηγός πρέπει να βρει σταθμό φόρτισης άμεσα. Παράλληλα, γνωρίζοντας την ταχύτητα του αυτοκινήτου και το λεπτό, στο οποίο το σύστημα fuzzy εξάγει πιθανότητα μεγαλύτερη ή ίση του 75%, μπορεί να υπολογιστεί το χιλιόμετρο στο οποίο ένας οδηγός αναζητάει σταθμό φόρτισης.



Διάγραμμα 2: Διάγραμμα ροής υπολογισμού χιλιομέτρου αναζήτησης σταθμού φόρτισης

Κεφάλαιο 5^ο

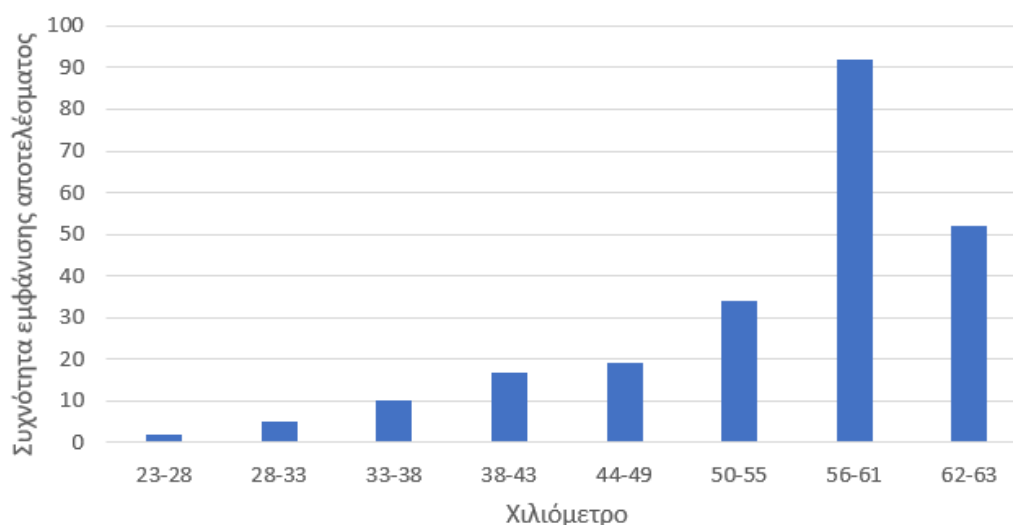
5.1 Αποτελέσματα διαδρομής Χανιά-Ρέθυμνο

Η μεθοδολογία, που αναπτύσσεται παραπάνω, χρησιμοποιείται για την ενεργειακή αξιολόγηση των τεσσάρων ηλεκτρικών αυτοκινήτων για όλες τις ώρες της ημέρας.

Παρακάτω απεικονίζονται τα αποτελέσματα για την διαδρομή Χανιά-Ρέθυμνο. Επιπλέον, αξίζει να αναφερθεί πως τα αποτελέσματα που προκύπτουν ομαδοποιούνται ανά 5 χιλιόμετρα.

Χιλιόμετρο	Συχνότητα εμφάνισης αποτελέσματος
23-28	2
28-33	5
33-38	10
38-43	17
44-49	19
50-55	34
56-61	92
62-63	52

Πίνακας 13: Αποτελέσματα διαδρομής Χανιά-Ρέθυμνο
Αποτελέσματα διαδρομής Χανιά-Ρέθυμνο

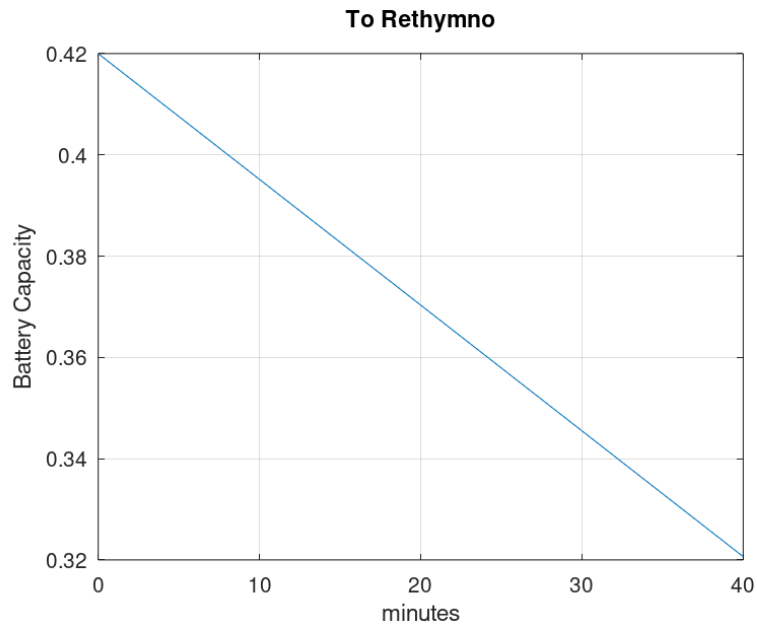


Εικόνα 62: Αποτελέσματα διαδρομής Χανιά-Ρέθυμνο

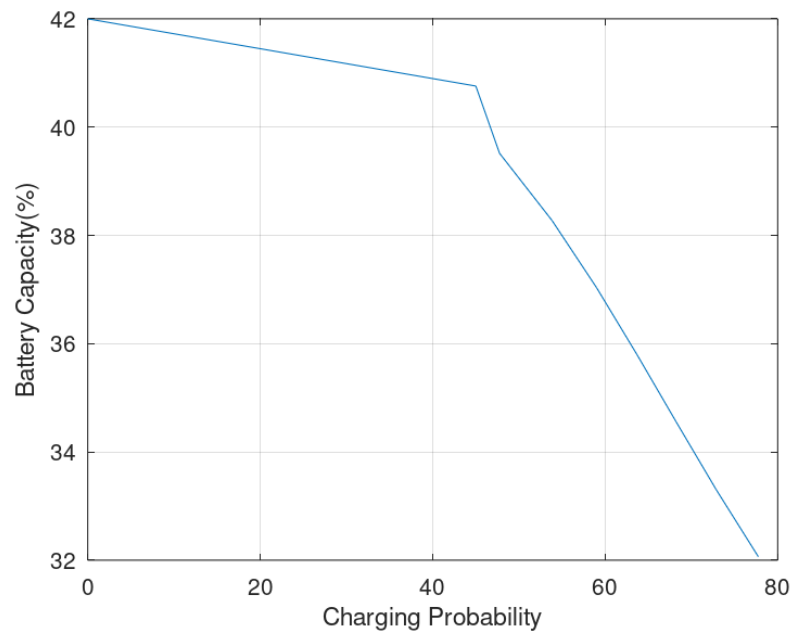
Στην συνέχεια, ακολουθούν ενδεικτικά παραδείγματα για τα ηλεκτρικά αυτοκίνητα, τα οποία περιλαμβάνουν την ταχύτητα του αυτοκινήτου, το αρχικό ποσοστό φόρτισης της μπαταρίας (initial State of Charge), το ποσοστό επιπέδου του άγχους του οδηγού, το αποτέλεσμα εκφρασμένο σε χιλιόμετρα, το διάγραμμα, που απεικονίζει την πτώση της μπαταρίας ανά 5 λεπτά και το διάγραμμα, το οποίο απεικονίζει το ποσοστό της μπαταρίας που υπάρχει ανά 5 λεπτά σε σχέση με το αποτέλεσμα του συστήματος fuzzy.

Audi Q4 e-tron 35

Ταχύτητα:	56 km/h
Ποσοστό επίπεδου άγχους:	39%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	42%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	41 ^ο χιλιόμετρο



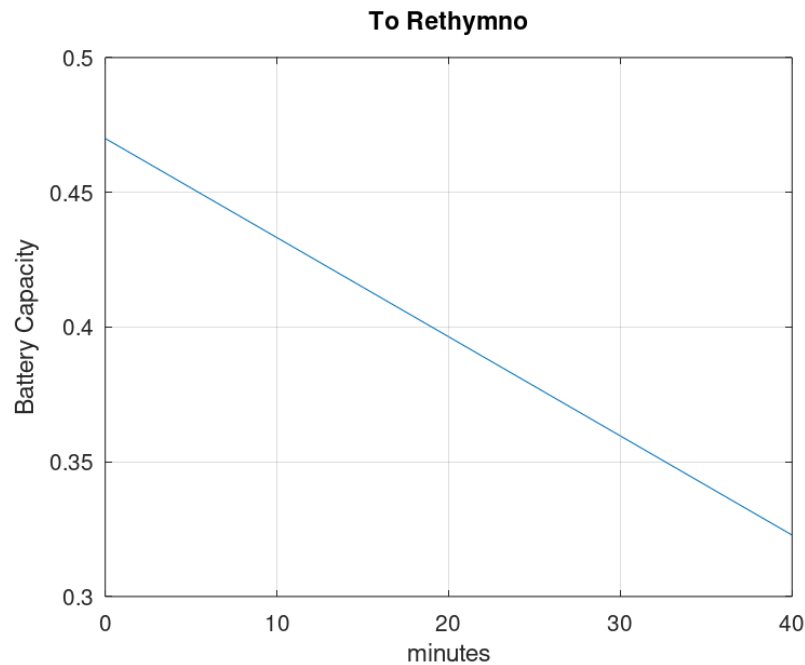
Εικόνα 63: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Audi, διαδρομή Χανιά-Ρέθυμνο)



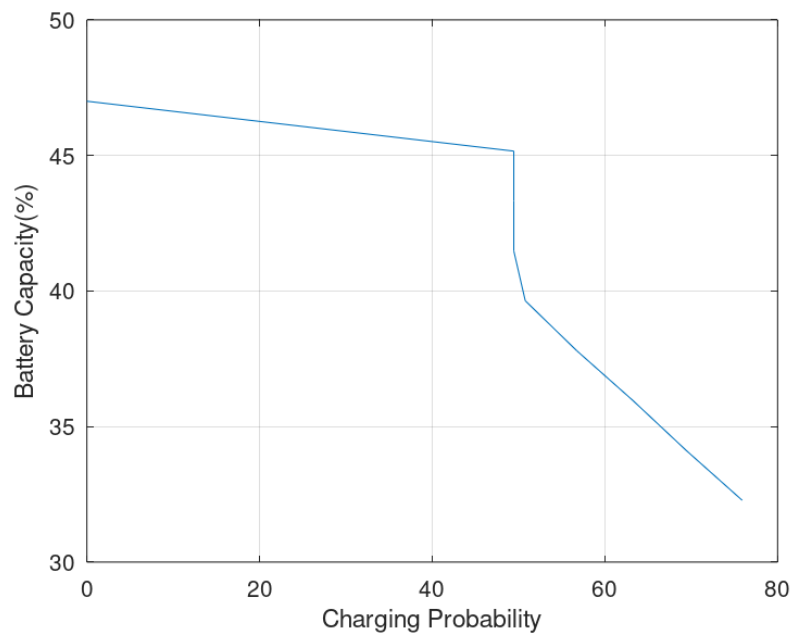
Εικόνα 64: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Audi, διαδρομή Χανιά-Ρέθυμνο)

Fiat 500e Hatchback

Ταχύτητα:	68 km/h
Ποσοστό επίπεδου άγχους:	57%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	47%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	51 ^ο χιλιόμετρο



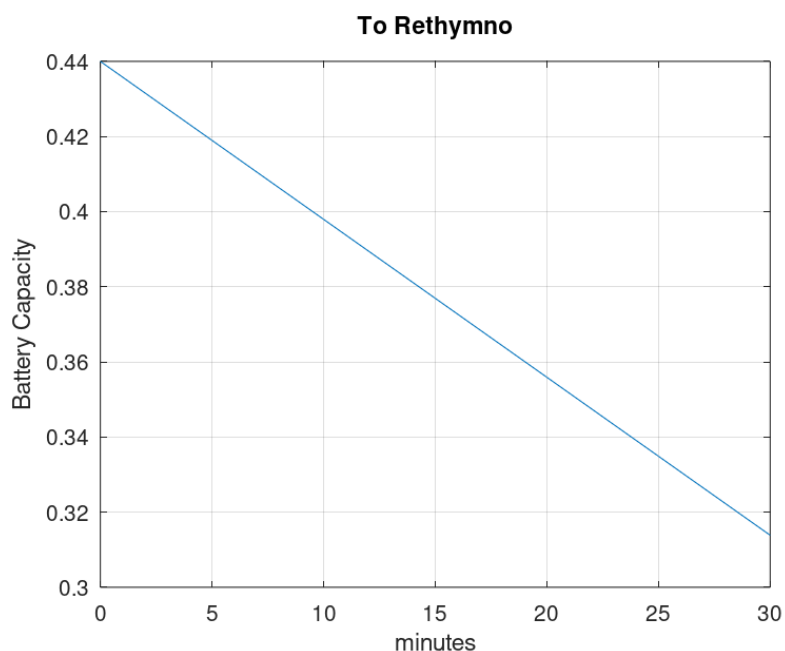
Εικόνα 65: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Fiat, διαδρομή Χανιά-Ρέθυμνο)



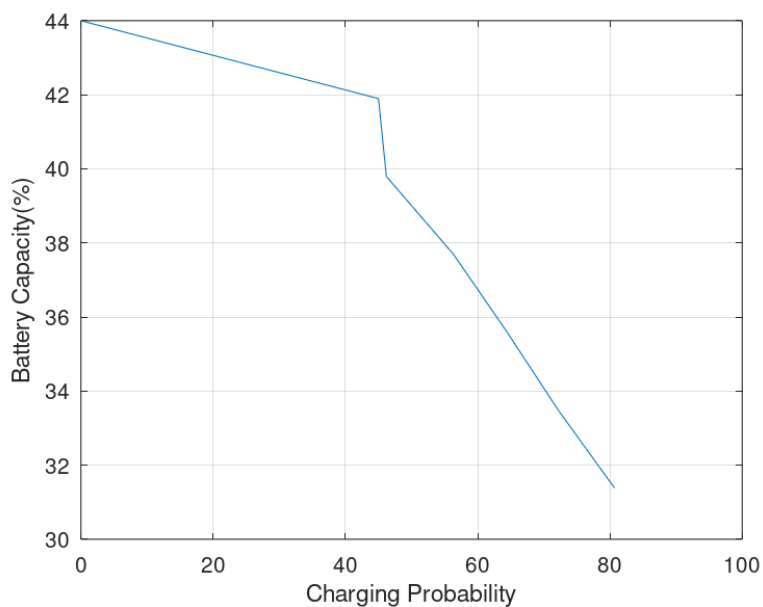
Εικόνα 66: Αποτέλεσμα συστήματος fuzzy συναρτήσεων του ποσοστού φόρτισης της μπαταρίας(Fiat, διαδρομή Χανιά-Ρέθυμνο)

Nissan Leaf

Ταχύτητα: 68 km/h
Ποσοστό επίπεδου άγχους: 37%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση: 44%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης: 34^ο χιλιόμετρο



Εικόνα 67: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Nissan, διαδρομή Χανιά-Ρέθυμνο)



Εικόνα 68: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Nissan, διαδρομή Χανιά-Ρέθυμνο)

Mazda MX-30

Ταχύτητα:

76 km/h

Ποσοστό επίπεδου άγχους:

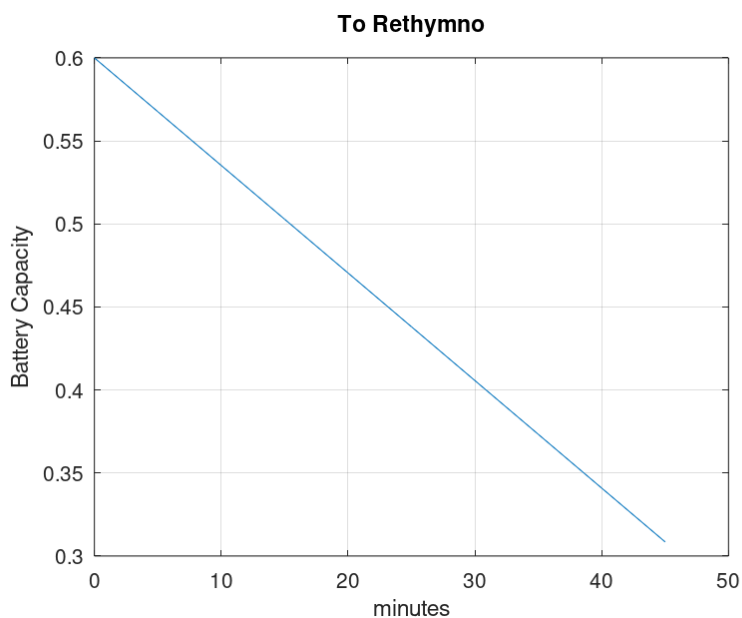
43%

Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:

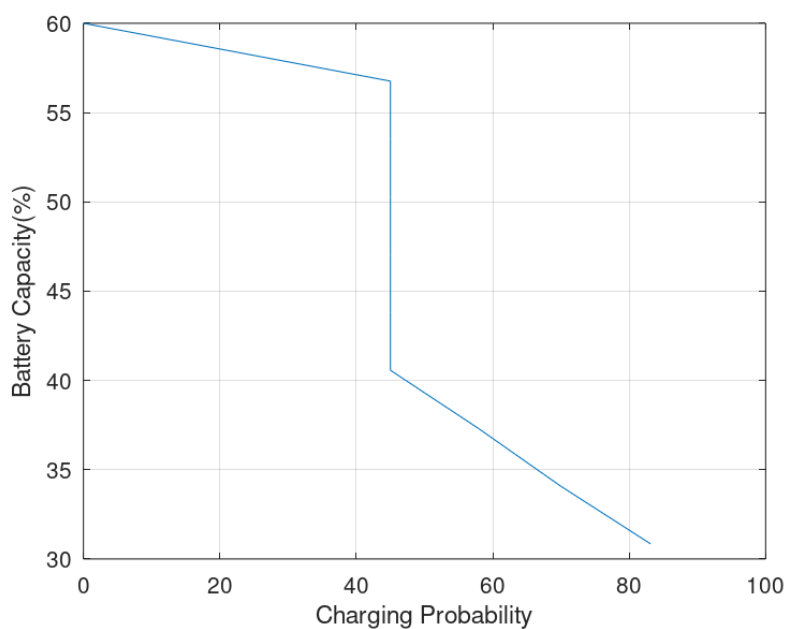
60%

Χιλιόμετρο αναζήτησης σταθμού φόρτισης:

57^ο χιλιόμετρο



Εικόνα 69: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Mazda, διαδρομή Χανιά-Ρέθυμνο)



Εικόνα 70: Αποτέλεσμα συστήματος fuzzy συναρτήσε του ποσοστού φόρτισης της μπαταρίας(Mazda, διαδρομή Χανιά-Ρέθυμνο)

5.2 Αποτελέσματα διαδρομής Χανιά-Ηράκλειο

Αντίστοιχα, στον παρακάτω πίνακα απεικονίζονται τα αποτελέσματα για την διαδρομή Χανιά-Ηράκλειο, τα οποία έχουν ομαδοποιηθεί ανά 5 χιλιόμετρα.

Χιλιόμετρο	Συχνότητα εμφάνισης αποτελέσματος
64-69	6
70-75	9
77-82	42
83-88	101
89-94	198
95-100	256
101-106	399
108-113	315
114-119	427
120-125	307
126-131	481
132-137	327
138-142	210

Πίνακας 14: Αποτελέσματα διαδρομής Χανιά-Ηράκλειο

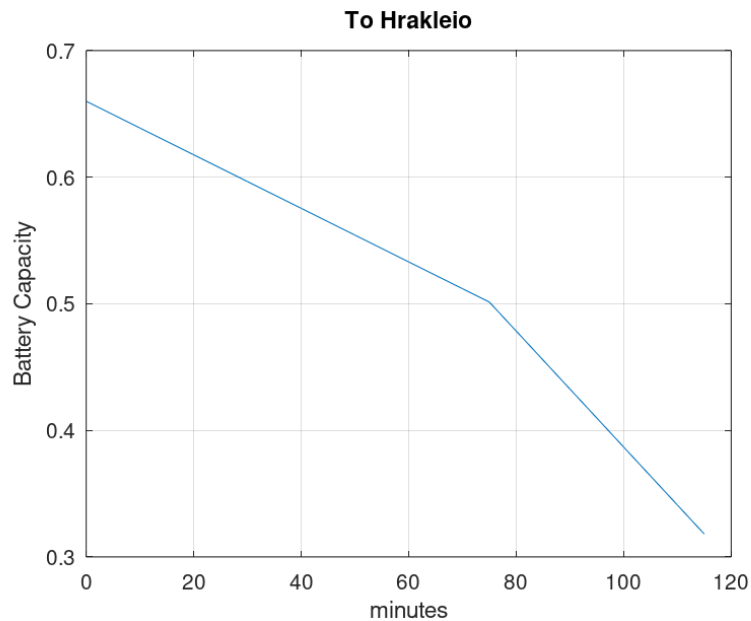


Εικόνα 71: Αποτελέσματα διαδρομής Χανιά-Ηράκλειο

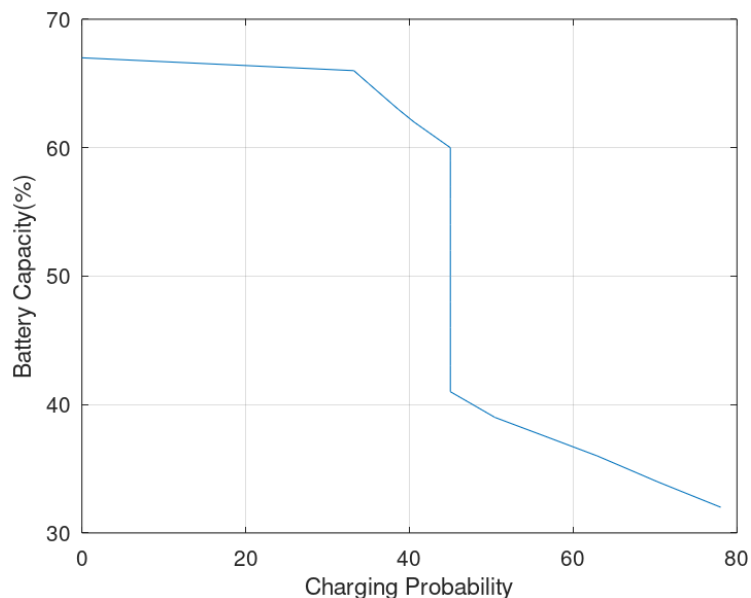
Στην συνέχεια, ακολουθούν ενδεικτικά παραδείγματα για τα ηλεκτρικά αυτοκίνητα, τα οποία περιλαμβάνουν την ταχύτητα του αυτοκινήτου, το αρχικό ποσοστό φόρτισης της μπαταρίας (initial State of Charge), το ποσοστό επιπέδου του άγχους του οδηγού, το αποτέλεσμα εκφρασμένο σε χιλιόμετρα, το διάγραμμα, που απεικονίζει την πτώση της μπαταρίας ανά 5 λεπτά και το διάγραμμα, το οποίο απεικονίζει το ποσοστό της μπαταρίας που υπάρχει ανά 5 λεπτά σε σχέση με το αποτέλεσμα του συστήματος fuzzy.

Audi Q4 e-tron 35

Ταχύτητα διαδρομής Χανιά-Ρέθυμνο:	62 km/h
Ταχύτητα διαδρομής Ρέθυμνο-Ηράκλειο:	82 km/h
Ποσοστό επίπεδου άγχους:	47%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	67%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	119 ^ο χιλιόμετρο



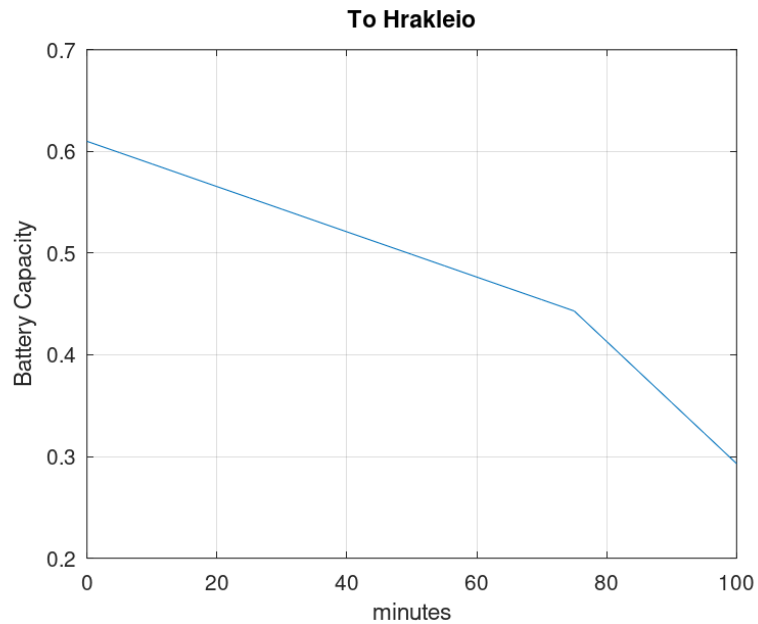
Εικόνα 72: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Audi, διαδρομή Χανιά-Ηράκλειο)



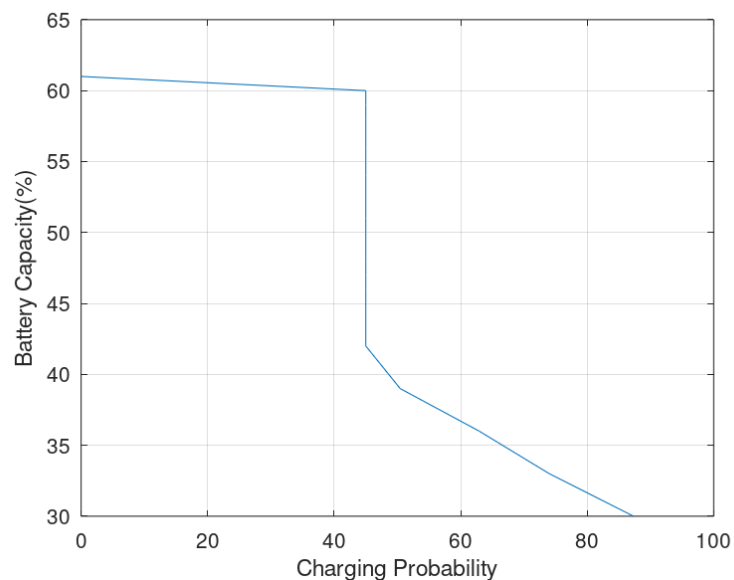
Εικόνα 73: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Audi, διαδρομή Χανιά-Ηράκλειο)

Fiat 500e Hatchback

Ταχύτητα διαδρομής Χανιά-Ρέθυμνο:	50 km/h
Ταχύτητα διαδρομής Ρέθυμνο-Ηράκλειο:	88 km/h
Ποσοστό επίπεδου άγχους:	50%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	61%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	101 ^ο χιλιόμετρο



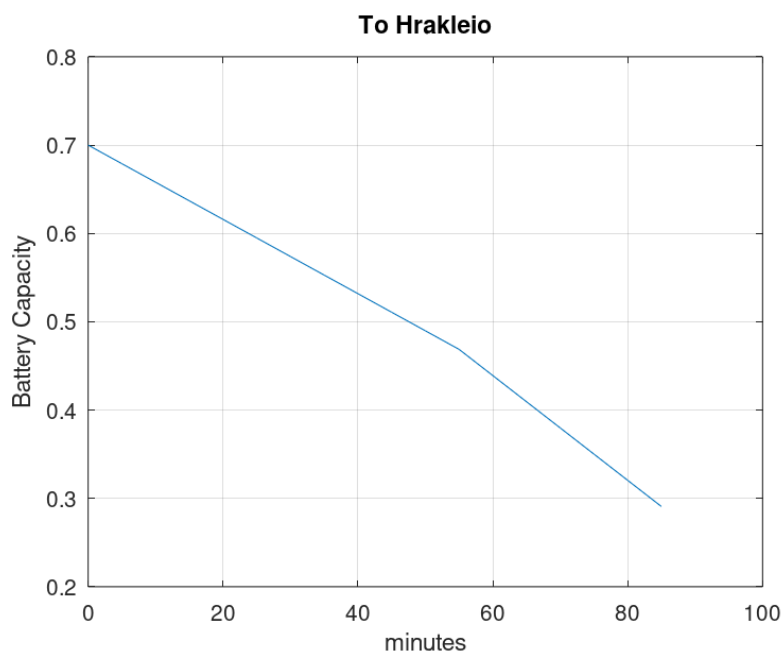
Εικόνα 74: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Fiat, διαδρομή Χανιά-Ηράκλειο)



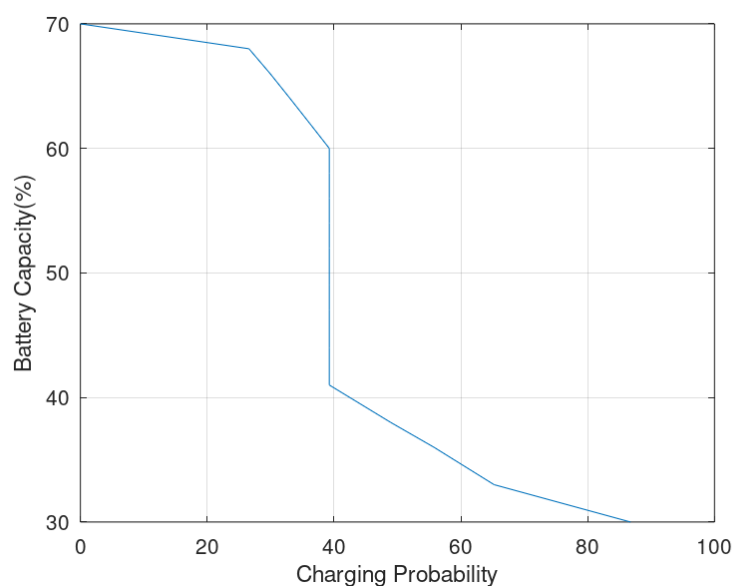
Εικόνα 75: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Fiat, διαδρομή Χανιά-Ηράκλειο)

Nissan Leaf

Ταχύτητα διαδρομής Χανιά-Ρέθυμνο:	68 km/h
Ταχύτητα διαδρομής Ρέθυμνο-Ηράκλειο:	82 km/h
Ποσοστό επίπεδου άγχους:	31%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	70%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	105 ^ο χιλιόμετρο



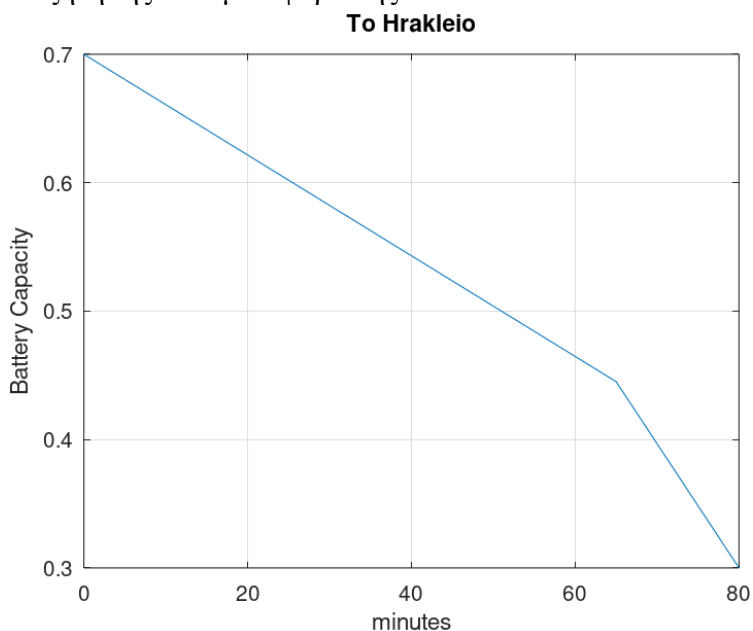
Εικόνα 76: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Nissan, διαδρομή Χανιά-Ηράκλειο)



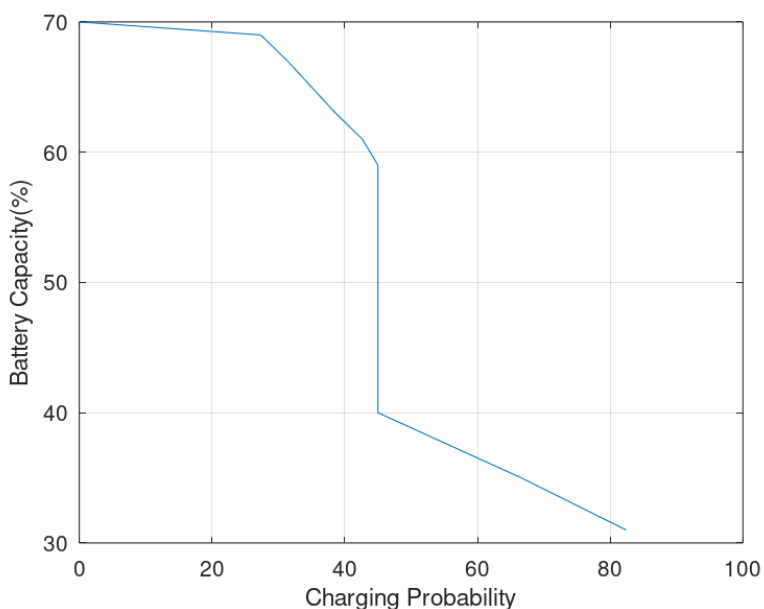
Εικόνα 77: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Nissan, διαδρομή Χανιά-Ηράκλειο)

Mazda MX-30

Ταχύτητα διαδρομής Χανιά-Ρέθυμνο:	56 km/h
Ταχύτητα διαδρομής Ρέθυμνο-Ηράκλειο:	94 km/h
Ποσοστό επίπεδου άγχους:	42%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	70%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	88 ^ο χιλιόμετρο



Εικόνα 78: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Mazda, διαδρομή Χανιά-Ηράκλειο)



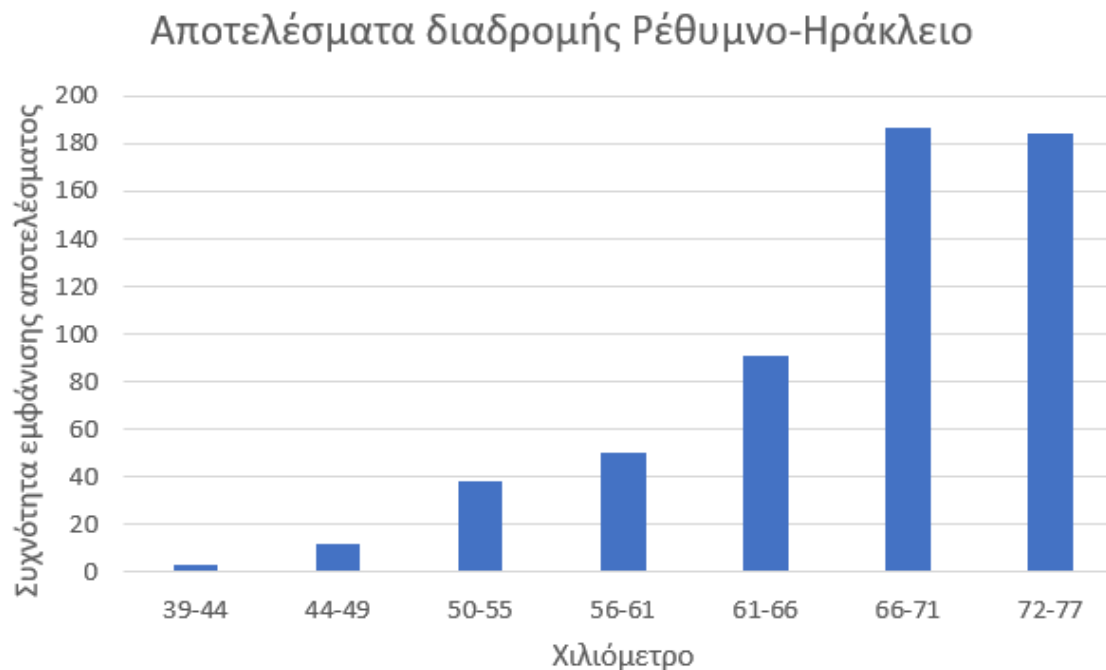
Εικόνα 79: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Mazda, διαδρομή Χανιά-Ηράκλειο)

5.3 Αποτελέσματα διαδρομής Ρέθυμνο-Ηράκλειο

Αντίστοιχα, στον παρακάτω πίνακα απεικονίζονται τα αποτελέσματα για την διαδρομή Χανιά-Ηράκλειο, τα οποία έχουν ομαδοποιηθεί ανά 5 χιλιόμετρα.

Χιλιόμετρο	Συχνότητα εμφάνισης αποτελέσματος
39-44	3
44-49	12
50-55	38
56-61	50
61-66	91
66-71	187
72-77	184

Πίνακας 15: Αποτελέσματα διαδρομής Ρέθυμνο-Ηράκλειο

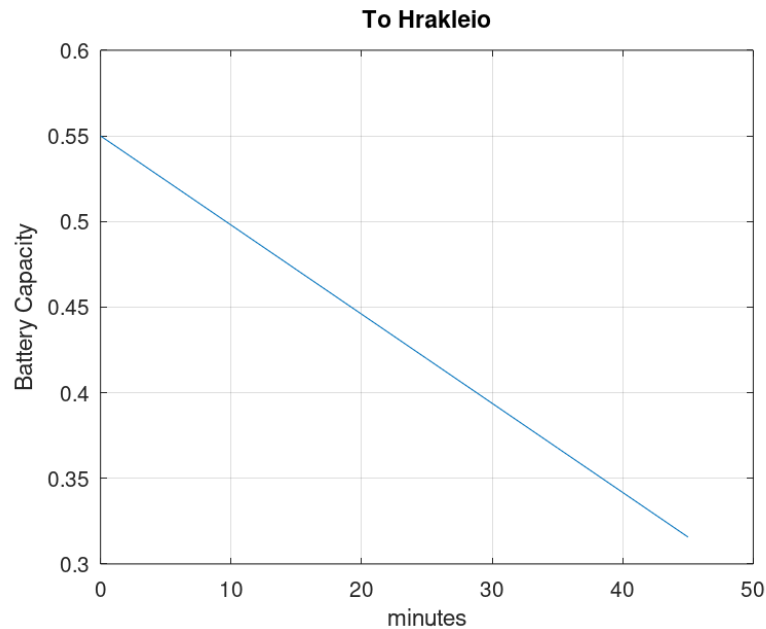


Εικόνα 80: Αποτελέσματα διαδρομής Ρέθυμνο-Ηράκλειο

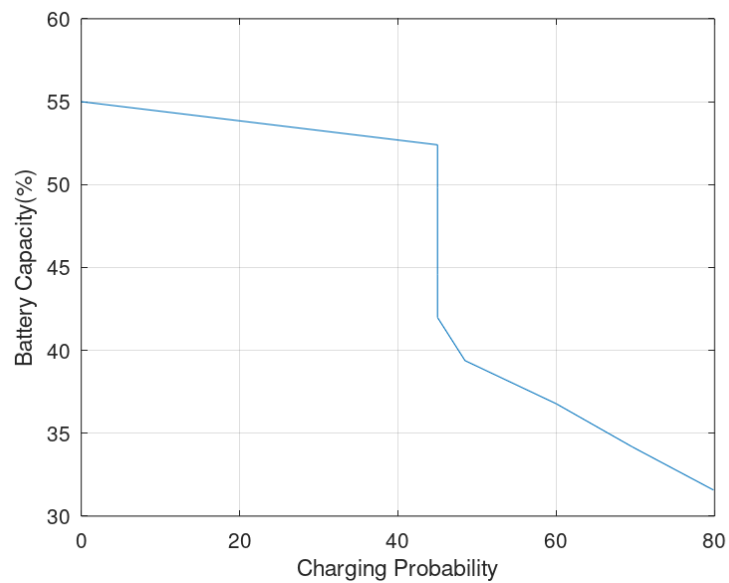
Στην συνέχεια, ακολουθούν ενδεικτικά παραδείγματα για τα ηλεκτρικά αυτοκίνητα, τα οποία περιλαμβάνουν την ταχύτητα του αυτοκινήτου, το αρχικό ποσοστό φόρτισης της μπαταρίας (initial State of Charge), το ποσοστό επιπέδου του άγχους του οδηγού, το αποτέλεσμα εκφρασμένο σε χιλιόμετρα, το διάγραμμα, που απεικονίζει την πτώση της μπαταρίας ανά 5 λεπτά και το διάγραμμα, το οποίο απεικονίζει το ποσοστό της μπαταρίας που υπάρχει ανά 5 λεπτά σε σχέση με το αποτέλεσμα του συστήματος fuzzy.

Audi Q4 e-tron 35

Ταχύτητα διαδρομής:	88 km/h
Ποσοστό επίπεδου άγχους:	37%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	55%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	66 ^ο χιλιόμετρο



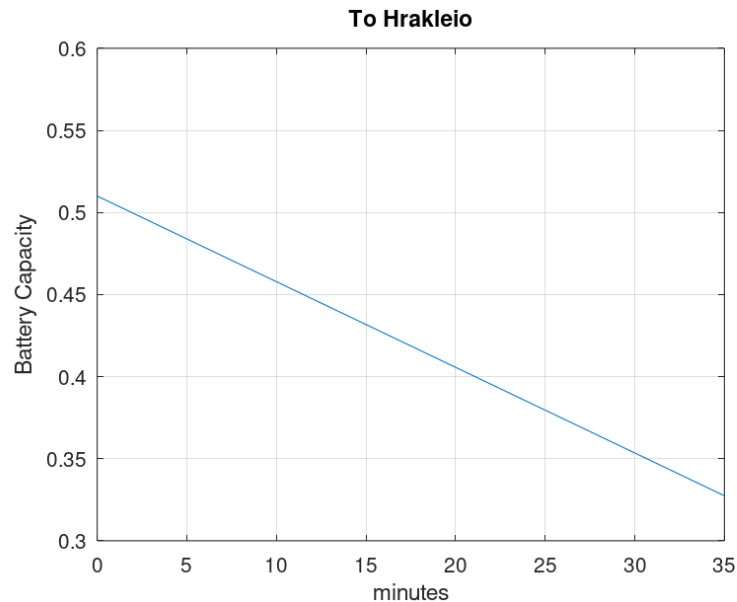
Εικόνα 81: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Audi, διαδρομή Ρέθυμνο-Ηράκλειο)



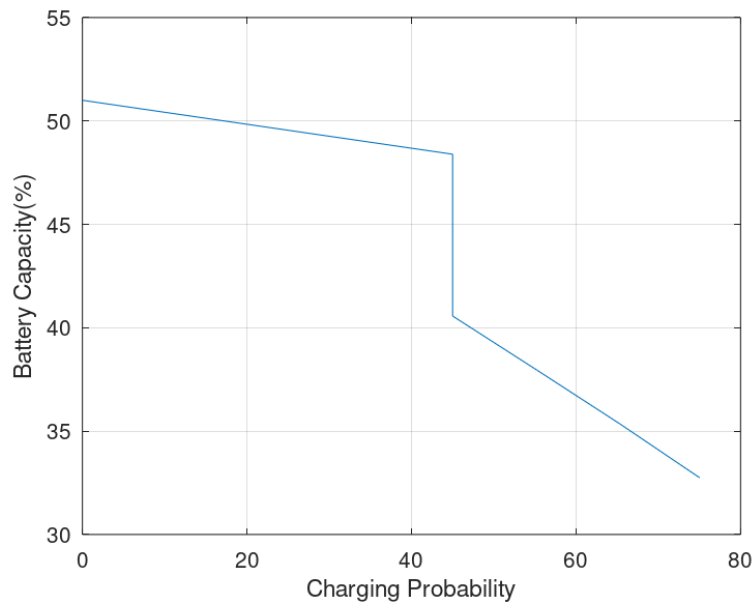
Εικόνα 82: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Audi, διαδρομή Ρέθυμνο-Ηράκλειο)

Fiat 500e Hatchback

Ταχύτητα διαδρομής:	82 km/h
Ποσοστό επίπεδου άγχους:	35%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:	51%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης:	54 ^ο χιλιόμετρο



Εικόνα 83: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Fiat, διαδρομή Ρέθυμνο-Ηράκλειο)



Εικόνα 84: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Fiat, διαδρομή Ρέθυμνο-Ηράκλειο)

Nissan Leaf

Ταχύτητα διαδρομής:

88 km/h

Ποσοστό επίπεδου άγχους:

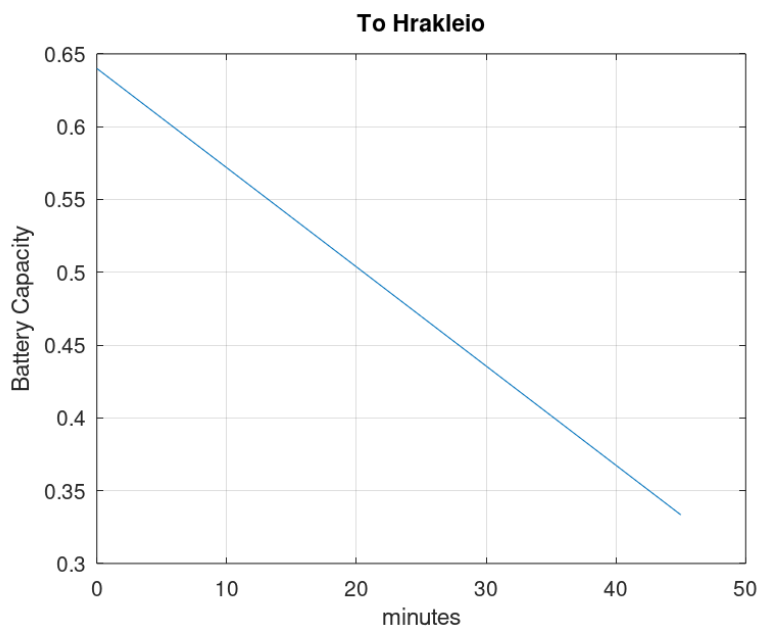
66%

Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση:

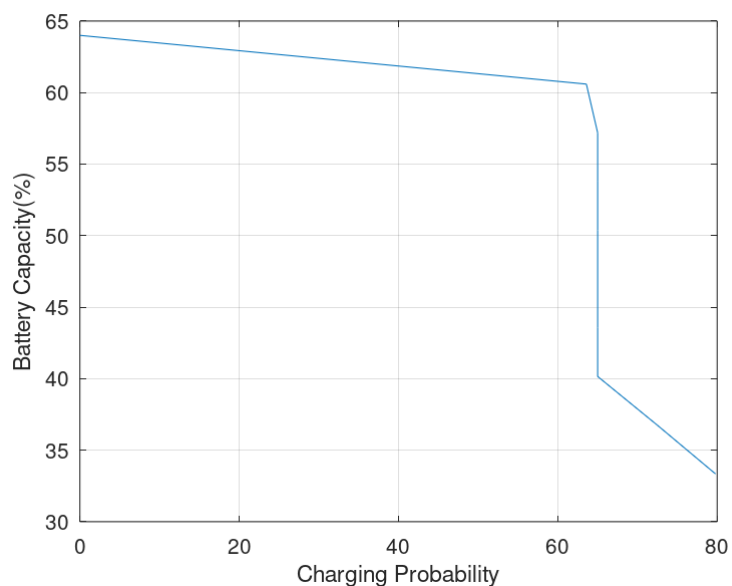
64%

Χιλιόμετρο αναζήτησης σταθμού φόρτισης:

66^ο χιλιόμετρο



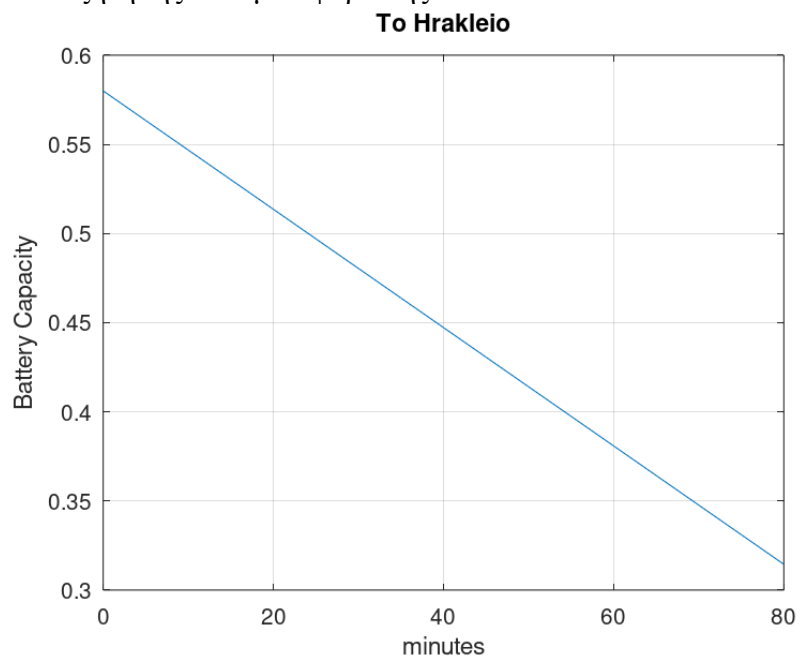
Εικόνα 85: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Nissan, διαδρομή Ρέθυμνο-Ηράκλειο)



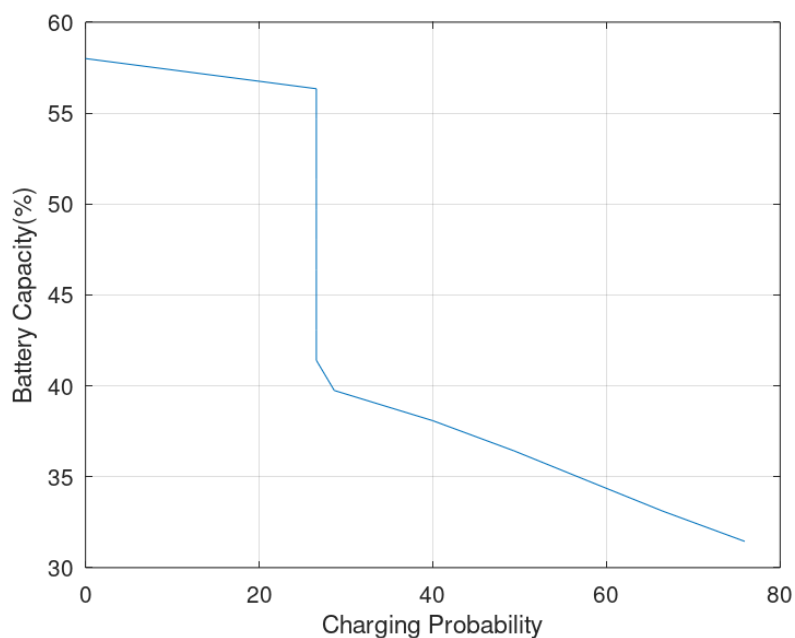
Εικόνα 86: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Nissan, διαδρομή Ρέθυμνο-Ηράκλειο)

Mazda MX-30

Ταχύτητα διαδρομής: 50 km/h
Ποσοστό επίπεδου άγχους: 21%
Ποσοστό φόρτισης της μπαταρίας κατά την εκκίνηση: 58%
Χιλιόμετρο αναζήτησης σταθμού φόρτισης: 70^ο χιλιόμετρο



Εικόνα 87: Πτώση ποσοστού φόρτισης της μπαταρίας μέχρι την χρονική στιγμή αναζήτησης σταθμού φόρτισης(Mazda, διαδρομή Ρέθυμνο-Ηράκλειο)



Εικόνα 88: Αποτέλεσμα συστήματος fuzzy συναρτήσει του ποσοστού φόρτισης της μπαταρίας(Mazda, διαδρομή Ρέθυμνο-Ηράκλειο)

Κεφάλαιο 6^ο

6.1 Μελλοντικές Βελτιώσεις

Οι μεθοδολογίες, που αναπτύσσονται παραπάνω, τόσο για την ανίχνευση και καταγραφή της ταχύτητας των αυτοκινήτων όσο και για την ενεργειακή αξιολόγησή τους μπορούν να βελτιωθούν ώστε να αποδώσουν ταχύτερα τα αποτελέσματα.

Αρχικά, το dataset, στο οποίο βασίζεται η ανίχνευση των αντικειμένων σε μία εικόνα ή ένα βίντεο, δύναται να μεταβληθεί. Πιο συγκεκριμένα, δεν απαιτούνται και οι 80 κλάσεις που περιέχει το COCO dataset για έναν αλγόριθμο, ώστε να αναγνωρίσει τα οχήματα που κινούνται σε έναν αυτοκινητόδρομο. Μία εναλλακτική λύση, λοιπόν, είναι η δημιουργία ενός dataset που περιέχει μόνο οχήματα, όπως είναι τα αυτοκίνητα, τα φορτηγά και τα λεωφορεία. Με αυτόν τον τρόπο, ο χρόνος που απαιτείται για την ανίχνευση των οχημάτων μειώνεται σημαντικά.

Επιπλέον, δεδομένου ότι ο αλγόριθμος ανίχνευσης κίνησης παρέχει τις συντεταγμένες των οχημάτων δεν χρειάζεται ο αλγόριθμος YOLO για την αναγνώριση των οχημάτων. Αντίθετα, μπορεί να εφαρμοστεί ένα συνελικτικό νευρωνικό δίκτυο (CNN), το οποίο δέχεται την εικόνα του κινούμενου οχήματος από τον αλγόριθμο ανίχνευσης κίνησης και εξάγει την πιθανότητα να είναι αυτοκίνητο με τη βοήθεια του dataset που αναφέρεται παραπάνω. Αποτέλεσμα των παραπάνω είναι η δραματική μείωση του χρόνου επεξεργασίας του εκάστοτε frame του βίντεο.

Ολοκληρώνοντας, η καταγραφή των αυτοκινήτων λαμβάνει χώρα σε τρία σημεία κατά μήκος του BOAK και κατά την ενεργειακή αξιολόγηση θεωρείται πως το αυτοκίνητο κινείται με σταθερή ταχύτητα μέχρι να μεταβεί στον προορισμό του. Μια λύση, που ανταποκρίνεται περισσότερο στην πραγματικότητα, είναι η καταγραφή των αυτοκινήτων και των ταχυτήτων τους ανά 5 ή 10 χιλιόμετρα μέχρι τον τελικό προορισμό. Με αυτόν τον τρόπο, η ενεργειακή αξιολόγηση γίνεται με κάθε ταχύτητα και τα αποτελέσματα είναι πιο αντιπροσωπευτικά.

Βιβλιογραφία

- [1] Britannica Encyclopedia, <https://www.britannica.com/technology/electric-automobile>, accessed on: 10/1/2022
- [2] Mitchell Tom, “Machine Learning”, Mcgraw-Hill Science/Engineering/Math (1997)
- [3] Computer Science Center, <https://www.csc.com.gr/machine-learning-μηχανική-μάθηση-τι-είναι/>, accessed on : 10/1/2022
- [4] IBM, <https://www.ibm.com/cloud/learn/supervised-learning#toc-supervised-IL4f88xL>, accessed on: 10/1/2022
- [5] IBM, <https://www.ibm.com/cloud/learn/unsupervised-learning>, accessed on: 10/1/2022
- [6] Deep Sense, <https://deepsense.ai/what-is-reinforcement-learning-the-complete-guide/>, accessed on: 10/1/2022
- [7] <https://machinelearningmastery.com/what-is-semi-supervised-learning/>, accessed on: 10/1/2022
- [8] Mobidev, <https://mobidev.biz/blog/5-essential-machine-learning-techniques>, accessed on: 10/1/2022
- [9] JavaTpoint, <https://www.javatpoint.com/classification-algorithm-in-machine-learning>, accessed on: 10/1/2022
- [10] GeeksforGeeks, <https://www.geeksforgeeks.org/clustering-in-machine-learning/>, accessed on: 10/1/2022
- [11] Great Learning, <https://www.mygreatlearning.com/blog/types-of-neural-networks/>, accessed on: 10/1/2022
- [12] Mizuho Nishio, Rikiya Yamashita, Richard Kinh Gian Do, Kaori Togashi, “Convolutional neural networks: an overview and application in radiology”, Springer Open (2018)
- [13] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, Silvio Savarese, “Generalized Intersection over Union: A Metric and A Loss for Bounding Box Regression”, Cornell University (2019)
- [14] Jan Hosang, Rodrigo Benenson, Bernt Schiele, “Learning non-maximum suppression”, Cornell University (2017)

- [15] Towards Data Science, <https://towardsdatascience.com/a-better-map-for-object-detection-32662767d424>, accessed on: 10/1/2022
- [16] Zhengxia Zou, Zhenwei Shi, Yuhong Guo, Jieping Ye, “Object Detection in 20 Years: A Survey”, Cornell University (2019)
- [17] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, Cornell University (2013)
- [18] Ross Girshick, “Fast R-CNN”, Cornell University (2015)
- [19] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, “Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks”, Cornell University (2015)
- [20] Jinsoo Kim, Jeongho Cho, “Exploring a Multimodal Mixture-Of-YOLOs Framework for Advanced Real-Time Object Detection”, MDPI (2020)
- [21] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection”, Cornell University (2015)
- [22] Joseph Redmon, Ali Farhadi, “YOLO9000: Better, Faster, Stronger”, Cornell University (2016)
- [23] Joseph Redmon, Ali Farhadi, “YOLOv3: An Incremental Improvement”, Cornell University (2018)
- [24] Alexey Bochkovskiy, Chien-Yao Wang, Hong-Yuan Mark Liao, “YOLOv4: Optimal Speed and Accuracy of Object Detection”, Cornell University (2020)
- [25] Kang Liu, Gellert Mattyus, “Fast Multiclass Vehicle Detection on Aerial Images”, IEEE (2015)
- [26] Hao Long, Yi-Nung Chung, Jun-de Li, “Automatic Vehicle Speed Estimation Method for Unmanned Aerial Vehicle Images”, Kuas University (2018)
- [27] Sania Bhatti, Liaquat A. Thebo, Mir Muhammad B. Talpur, Mohsin A. Memon, “A Video based Vehicle Detection, Counting and Classification System”, Research Gate (2018)
- [28] Aderonke A. Oni, Nicholas Kajoh, “Video-Based Vehicle Counting System for Urban Roads in Nigeria Using Yolo and DCF-CSR Algorithms”, Covenant University (2019)
- [29] Huansheng Song, Haoxiang Liang, Huaiyu Li, Zhe Dai, Xu Yun, “Vision-based vehicle detection and counting system using deep learning in highway scenes”, Springer Open (2019)

- [30] Debojit Biswas, Hongbo Su, Chengyi Wang, Aleksandar Stevanovic, “Speed Estimation of Multiple Moving Objects from a Moving UAV Platform”, MDPI (2019)
- [31] Kirill Khazukov, Vladimir Shepelev, Tatiana Karpeta, Salavat Shabiev, Ivan Slobodin, Irakli Charbadze, Irina Alferova, “Real-time monitoring of traffic parameters”, Springer Open (2020)
- [32] Mehrdad Ehsani, Yimin Gao, Ali Emadi, “Modern Electric, Hybrid Electric, and Fuel Cell Vehicles”, CRC Press (2010)
- [33] IWM, <https://www.iwm.org.uk/history/a-brief-history-of-drones>, accessed on: 10/1/2022
- [34] DJI, <https://www.dji.com/gr/phantom-4-pro-v2/specs>, accessed on: 10/1/2022
- [35] Wikipedia, <https://el.wikipedia.org/wiki/Python>, accessed on: 10/1/2022
- [36] Opencv, <https://opencv.org/about/>, accessed on : 10/1/2022
- [37] Nvidia Cuda Manual, https://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf, accessed on: 10/1/2022
- [38] Github, <https://github.com/pjreddie/darknet/blob/master/data/coco.names>, accessed on : 10/1/2022
- [39] Github, <https://github.com/AlexeyAB/darknet/wiki/CFG-Parameters-in-the-%5Bnet%5D-section>, accessed on : 10/1/2022
- [40] Github, <https://github.com/AlexeyAB/darknet/wiki/CFG-Parameters-in-the-different-layers>, accessed on : 10/1/2022
- [41] Opencv, https://docs.opencv.org/4.x/d1/dc5/tutorial_background_subtraction.html, accessed on : 10/1/2022
- [42] Packorn KadevTraKuPong, Richard Bowden, “An improved adaptive background mixture model for real-time tracking with shadow detection”, Research Gate (2001)
- [43] Andrew B. Godbehare, Akihiro Matsukawa, Ken Goldberg, “Visual tracking of human visitors under variable-lighting conditions for a responsive audio art installation”, IEEE (2012)
- [44] Zoran Zivkovic, Ferdinand van der Heijden, “Efficient adaptive density estimation per image pixel for the task of background subtraction”, Science Direct (2006)

[45] Leandro Marcomini, André Luiz Cunha, “A Comparison between Background Modelling Methods for Vehicle Segmentation in Highway Traffic Videos”, Research Gate (2018)

[46] Opencv, https://docs.opencv.org/3.4/d9/d61/tutorial_py_morphological_ops.html, accessed on: 10/1/2022

[47] Ανδρέας Τζανής, Εισαγωγή στο Matlab και στο Octave

[48] EV-Database, <https://ev-database.org/car/1179/Audi-Q4-e-tron-35>, accessed on: 10/1/2022

[49] EV-Database, <https://ev-database.org/car/1285/Fiat-500e-Hatchback-42-kWh>, accessed on: 10/1/2022

[50] EV-Database, <https://ev-database.org/car/1106/Nissan-Leaf>, accessed on: 10/1/2022

[51] EV-Database, <https://ev-database.org/car/1241/Mazda-MX-30>, accessed on: 10/1/2022

[52] Mathworks, <https://www.mathworks.com/matlabcentral/fileexchange/34101-random-numbers-from-a-discrete-distribution>, accessed on: 10/1/2022

[53] Mathworks, <https://www.mathworks.com/help/fuzzy/newfis.html>, accessed on: 10/1/2022