



ΠΑΝΕΠΙΣΤΗΜΙΟ
ΘΕΣΣΑΛΙΑΣ

ΣΧΟΛΗ ΘΕΤΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ

ΑΝΑΓΝΩΡΙΣΗ ΤΡΟΧΑΙΩΝ ΠΑΡΑΒΙΑΣΕΩΝ ΜΕ ΤΕΧΝΙΚΕΣ ΒΑΘΙΑΣ ΜΑΘΗΣΗΣ

ΓΚΕΝΤΙΑΝ ΧΙΝΤΑ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

ΥΠΕΥΘΥΝΟΣ

ΚΟΛΟΜΒΑΤΣΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ
ΕΠΙΚΟΥΡΟΣ ΚΑΘΗΓΗΤΗΣ

Λαμία 2024



UNIVERSITY OF
THESSALY

SCHOOL OF SCIENCE

DEPARTMENT OF COMPUTER SCIENCE & TELECOMMUNICATIONS

DETECTION OF TRAFFIC VIOLATION WITH DEEP LEARNING TECHNIQUES

GKENTIAN CHINTA

FINAL THESIS

ADVISOR
KOLOMVASOS KONSTANTINOS
ASSISTANT PROFESSOR

Lamia 2024

«Με ατομική μου ευθύνη και γνωρίζοντας τις κυρώσεις ⁽¹⁾, που προβλέπονται από της διατάξεις της παρ. 6 του άρθρου 22 του Ν. 1599/1986, δηλώνω ότι:

1. Δεν παραθέτω κομμάτια βιβλίων ή άρθρων ή εργασιών άλλων αυτολεξεί **χωρίς να τα περικλείω σε εισαγωγικά** και χωρίς να αναφέρω το συγγραφέα, τη χρονολογία, τη σελίδα. Η αυτολεξεί παράθεση χωρίς εισαγωγικά χωρίς αναφορά στην πηγή, είναι λογοκλοπή. Πέραν της αυτολεξεί παράθεσης, λογοκλοπή θεωρείται και η παράφραση εδαφίων από έργα άλλων, συμπεριλαμβανομένων και έργων συμφοιτητών μου, καθώς και η παράθεση στοιχείων που άλλοι συνέλεξαν ή επεξεργάσθηκαν, χωρίς αναφορά στην πηγή. Αναφέρω πάντοτε με πληρότητα την πηγή κάτω από τον πίνακα ή σχέδιο, όπως στα παραθέματα.
2. Δέχομαι ότι η αυτολεξεί **παράθεση χωρίς εισαγωγικά**, ακόμα κι αν συνοδεύεται από αναφορά στην πηγή σε κάποιο άλλο σημείο του κειμένου ή στο τέλος του, είναι αντιγραφή. Η αναφορά στην πηγή στο τέλος π.χ. μιας παραγράφου ή μιας σελίδας, δεν δικαιολογεί συρραφή εδαφίων έργου άλλου συγγραφέα, έστω και παραφρασμένων, και παρουσίασή τους ως δική μου εργασία.
3. Δέχομαι ότι υπάρχει επίσης περιορισμός στο μέγεθος και στη συχνότητα των παραθεμάτων που μπορώ να εντάξω στην εργασία μου εντός εισαγωγικών. Κάθε μεγάλο παράθεμα (π.χ. σε πίνακα ή πλαίσιο, κλπ), προϋποθέτει ειδικές ρυθμίσεις, και όταν δημοσιεύεται προϋποθέτει την άδεια του συγγραφέα ή του εκδότη. Το ίδιο και οι πίνακες και τα σχέδια
4. Δέχομαι όλες τις συνέπειες σε περίπτωση λογοκλοπής ή αντιγραφής.

Ημερομηνία:/...../20.....

Ο – Η Δηλ.

(1) «Όποιος εν γνώσει του δηλώνει ψευδή γεγονότα ή αρνείται ή αποκρύπτει τα αληθινά με έγγραφη υπεύθυνη δήλωση του άρθρου 8 παρ. 4 Ν. 1599/1986 τιμωρείται με φυλάκιση τουλάχιστον τριών μηνών. Εάν ο υπαίτιος αυτών των πράξεων σκόπευε να προσπορίσει στον εαυτόν του ή σε άλλον περιουσιακό όφελος βλάπτοντας τρίτον ή σκόπευε να βλάψει άλλον, τιμωρείται με κάθειρξη μέχρι 10 ετών.»

ΠΕΡΙΛΗΨΗ

Στον τομέα της αυτοματοποιημένης επιτήρησης της τήρησης των κανόνων της κυκλοφορίας, η ανάγκη για αποτελεσματικά συστήματα είναι αδιαμφισβήτητη. Το παρόν έργο στοχεύει στην ανίχνευση παραβάσεων κυκλοφορίας μέσω τεχνικών βαθιάς μάθησης. Κύριος στόχος είναι η ανάπτυξη ενός συστήματος ικανού να αναγνωρίζει αυτοκίνητα και σημάνσεις στάσης, χρησιμοποιώντας ένα προσαρμοσμένο σύνολο δεδομένων.

Για την ανίχνευση και την παρακολούθηση, το σύστημα εκπαιδεύτηκε για 100 επαναλήψεις χρησιμοποιώντας τον αλγόριθμο YOLOv5, ο οποίος αναγνώρισε αυτοκίνητα και πινακίδες stop. Ωστόσο, επικεντρώθηκε αποκλειστικά στα αυτοκίνητα, αποκλείοντας άλλα οχήματα όπως φορτηγά. Ο αλγόριθμος SORT χρησιμοποιήθηκε για την παρακολούθηση των ανιχνευμένων αυτοκινήτων, παράγοντας ένα βίντεο εξόδου και ένα αρχείο CSV που περιέχει τις θέσεις των αυτοκινήτων με αντίστοιχους αναγνωριστικούς αριθμούς (π.χ., αυτοκίνητο1, αυτοκίνητο2). Η εκτίμηση της ταχύτητας των οχημάτων επιτεύχθηκε μέσω ενός εξειδικευμένου κώδικα, ο οποίος λαμβάνει υπόψη την πληροφορία του βίντεο, τα FPS, τα μέτρα ανά αυτοκίνητο και τα μέτρα της κάμερας κτλ. Αυτές οι ταχύτητες είναι πολύ κοντινές στις αληθινές, καθώς οι πραγματικές αποστάσεις δεν ήταν γνωστές, δεδομένου ότι τα βίντεο προήλθε από μια δευτερεύουσα πηγή. Επιπλέον, το σύστημα διαθέτει την ικανότητα να ανιχνεύει οχήματα που δεν συμμορφώνονται με τις σημάνσεις στοπ. Σε περίπτωση που ένα αυτοκίνητο δεν σταματήσει σε πινακίδα στοπ, το σύστημα καταγράφει αυτή την παράβαση ως σημαντικό αποτέλεσμα, αυξάνοντας την αξιοπιστία και την αποδοτικότητά του στην ανίχνευση τέτοιων περιπτώσεων.

Η εφαρμογή αυτού του συστήματος ενδέχεται να έχει σημαντικό αντίκτυπο στην παρακολούθηση και ανίχνευση παραβάσεων της κυκλοφορίας στην πραγματική ζωή. Ωστόσο, υπάρχουν ορισμένοι περιορισμοί και προκλήσεις, όπως η εκτίμηση αποστάσεων και ταχυτήτων, που πρέπει να ληφθούν υπόψη. Οι μελλοντικές βελτιώσεις του συστήματος μπορεί να περιλαμβάνουν πιο ακριβείς μεθόδους για την καταγραφή ταχυτήτων, καθώς και την επέκταση της ανίχνευσης σε όλα τα είδη οχημάτων. Η προσέγγιση αυτή δείχνει υποσχόμενη για την ενίσχυση της ασφάλειας της κυκλοφορίας και την αποτελεσματική διαχείριση των παραβάσεων της κυκλοφορίας, προσφέροντας μια πιο αξιόπιστη και αυτοματοποιημένη λύση.

ABSTRACT

In the field of automated traffic rule enforcement, the need for effective systems is undeniable. This project aims to detect traffic violations using deep learning techniques. The primary goal is to develop a system capable of recognizing cars and stop signs, utilizing a customized dataset.

For detection and tracking, the system was trained for 100 epochs using the YOLOv5 algorithm, which identified cars and stop signs. However, it focused exclusively on cars, excluding other vehicles such as trucks. The SORT algorithm was used for tracking the detected cars, producing an output video and a CSV file containing the positions of the cars with corresponding identifiers (e.g., car1, car2). The estimation of vehicle speeds was achieved through a specialized code, which takes into account the video information, frames per second (FPS), meters per car, and camera meters, etc. These speeds are very close to the actual speeds, as the real distances were unknown, given that the videos came from a secondary source. Additionally, the system has the ability to detect vehicles that do not comply with stop sign indications. In cases where a car does not stop at a stop sign, the system records this violation as a significant result, increasing its reliability and efficiency in detecting such instances.

The application of this system is likely to have a significant impact on monitoring and detecting traffic violations in real life. However, there are some limitations and challenges, such as the estimation of distances and speeds, which need to be considered. Future improvements to the system may include more accurate methods for recording speeds, as well as expanding detection to all types of vehicles. This approach shows promise for enhancing traffic safety and effective management of traffic violations, offering a more reliable and automated solution.

Table of Contents

ΠΕΡΙΛΗΨΗ	I
ABSTRACT	II
ΚΕΦΑΛΑΙΟ 1 ΕΙΣΑΓΩΓΗ	3
1.1 ΤΕΧΝΟΛΟΓΙΚΕΣ ΚΑΙΝΟΤΟΜΙΕΣ ΚΑΙ ΟΔΙΚΗ ΑΣΦΑΛΕΙΑ	3
1.1 Α) ΟΙ ΚΥΡΙΟΤΕΡΕΣ ΠΑΡΑΒΑΣΕΙΣ ΚΩΔΙΚΑ ΟΔΙΚΗΣ ΚΥΚΛΟΦΟΡΙΑΣ	3
1.1 Β) ΕΚΤΙΜΗΣΗ ΤΑΧΥΤΗΤΑΣ ΚΑΙ ΠΑΡΑΒΑΣΗ ΤΗΣ ΠΙΝΑΚΙΑΣ STOP: ΥΛΟΠΟΙΗΣΗ ΜΕΣΩ ΒΑΘΙΑΣ ΜΑΘΗΣΗΣ	4
1.1 Γ) Ο ΡΟΛΟΣ ΤΗΣ ΤΗΡΗΣΗΣ ΤΩΝ ΚΑΝΟΝΩΝ ΚΥΚΛΟΦΟΡΙΑΣ ΣΤΗΝ ΟΔΙΚΗ ΑΣΦΑΛΕΙΑ	4
1.2 ΑΥΤΟΜΑΤΟΠΟΙΗΜΕΝΗ ΕΠΙΤΗΡΗΣΗ ΚΥΚΛΟΦΟΡΙΑΣ	4
1.2 ΓΙΑΤΙ ΒΑΘΙΑ ΜΑΘΗΣΗ;	5
ΚΕΦΑΛΑΙΟ 2 ΜΗΧΑΝΙΚΗ ΜΑΘΗΜΗΣΗ ΚΑΙ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ	8
2.1 ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ ΚΑΙ ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ; [15]	8
2.2 ΚΥΡΙΟΤΕΡΑ ΕΙΔΗ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ	9
2.2 Α) SUPERVISED LEARNING [14]	9
2.2 Β) UNSUPERVISED LEARNING [13]	9
2.2 Γ) REINFORCEMENT LEARNING [16]	10
2.3 ΤΙ ΕΙΝΑΙ ΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ; [3],[7]	11
2.4 ΠΩΣ ΛΕΙΤΟΥΡΓΟΥΝ ΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ;	11
2.5 ΕΙΔΗ ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ	12
2.5 Α) CNN (CONVOLUTIONAL NEURAL NETWORK)	12
2.5 Β) RNN (RECURRENT NEURAL NETWORKS)	14
2.5 Γ) MPLS (MULTIPROTOCOL LABELS SWITCHING) [27]	14
ΚΕΦΑΛΑΙΟ 3 ΑΛΓΟΡΙΘΜΟΙ ΕΝΤΟΠΙΣΜΟΥ (DETECTION ALGORITHMS)	16
3.1 OBJECT DETECTION (ΕΝΤΟΠΙΣΜΟΣ ΑΝΤΙΚΕΙΜΕΝΩΝ) [20]	16
3.2 ΠΡΟΗΓΟΥΜΕΝΕΣ ΕΡΓΑΣΙΕΣ ΕΝΤΟΠΙΣΜΟΥ ΣΤΟΥΣ ΔΡΟΜΟΥΣ	18

3.2 Α) ΕΡΓΑΣΙΑ ΕΝΤΟΠΙΣΜΟΥ ΑΥΤΟΚΙΝΗΤΩΝ [30].....	18
3.2 Β)ΕΡΓΑΣΙΑ ΕΚΤΙΜΗΣΗ ΤΑΧΥΤΗΤΑΣ [31]	19
3.2 Γ) ΕΡΓΑΣΙΑ ΕΝΤΟΠΙΣΜΟΥ ΣΗΜΑΤΩΝ ΤΡΟΧΑΙΑΣ [32]	20
<u>3.3 ΤΥΠΟΙ ΑΛΓΟΡΙΘΜΩΝ.....</u>	<u>21</u>
3.3 Α) TWO-STAGE DETECTORS [23],[24]	21
3.3 Β) ONE-STAGE DETECTORS [6],[23],[25].....	22
<u>3.4 ΑΛΓΟΡΙΘΜΟΣ YOLO [26],[27],[28].....</u>	<u>22</u>
3.4 Α) ΤΙ ΕΙΝΑΙ Ο ΑΛΓΟΡΙΘΜΟΣ YOLO;	22
3.4 Β) ΠΩΣ ΛΕΙΤΟΥΡΓΕΙ;	22
3.5) YOLOv1 [22]	25
3.6) YOLOv3 [33]	26
3.6) YOLOv5 [26]	26
3.7) YOLOv9 [21]	28
<u>ΚΕΦΑΛΑΙΟ 4 ΕΚΠΑΙΔΕΥΣΗ ΤΟΥ ΑΛΓΟΡΙΘΜΟΥ YOLOV5 ΚΑΙ ΧΡΗΣΗ ΤΟΥ SORT</u>	<u>30</u>
<u>4.1 DATASET.....</u>	<u>30</u>
4.1 Α)ΤΙ ΕΙΝΑΙ ΕΝΑ DATASET; [12]	30
4.1 Β)ΔΗΜΙΟΥΡΓΙΑ DATASET ΓΙΑ ΕΝΤΟΠΙΣΜΟ ΑΥΤΟΚΙΝΗΤΩΝ ΚΑΙ ΠΙΝΑΚΙΔΩΝ STOP	30
<u>4.2 ΕΚΠΑΙΔΕΥΣΗ YOLOV5.....</u>	<u>32</u>
4.2 Α) ΕΚΠΑΙΔΕΥΣΗ YOLOV5 ΜΕ ΤΙΣ ΚΑΤΑΛΛΗΛΕΣ ΠΑΡΑΜΕΤΡΟΥΣ.....	32
4.2 Β) ΕΠΕΞΕΡΓΑΣΙΑ ΒΙΝΤΕΟ	33
<u>4.3 ΧΡΗΣΗ ΑΛΓΟΡΙΘΜΟΥ SORT</u>	<u>36</u>
4.3 Α) ΑΛΓΟΡΙΘΜΟΣ SORT[28][29].....	36
4.3 Β) ΕΝΤΟΠΙΣΜΟΣ ΑΥΤΟΚΙΝΗΤΩΝ ΣΕ ΒΙΝΤΕΟ.....	37
<u>ΚΕΦΑΛΑΙΟ 5 ΕΚΤΙΜΗΣΗ ΤΑΧΥΤΗΤΑΣ</u>	<u>40</u>
<u>5.1 ΕΙΣΑΓΩΓΗ ΚΑΙ ΑΝΑΛΥΣΗ ΔΕΔΟΜΕΝΩΝ</u>	<u>40</u>
<u>5.2 ΥΠΟΛΟΓΙΣΜΟΣ ΤΑΧΥΤΗΤΑΣ ΚΑΙ ΠΑΡΑΒΙΑΣΗ STOP.....</u>	<u>40</u>
5.2 Α) ΥΠΟΛΟΓΙΣΜΟΣ ΤΑΧΥΤΗΤΑΣ	40
5.2 Β) ROC CURVE [17]	43
5.2 Γ) ΠΑΡΑΒΙΑΣΗ STOP.....	43

ΚΕΦΑΛΑΙΟ 6 ΣΥΜΠΕΡΑΣΜΑΤΑ.....	47
6.1 ΣΥΜΠΕΡΑΣΜΑΤΑ.....	47
6.2 ΜΕΛΛΟΝΤΙΚΕΣ ΒΕΛΤΙΩΣΕΙΣ.....	47
ΒΙΒΛΙΟΓΡΑΦΙΑ.....	48

ΚΕΦΑΛΑΙΟ 1 Εισαγωγή

1.1 Τεχνολογικές Καινοτομίες και Οδική Ασφάλεια

1.1 α) Οι Κυριότερες Παραβάσεις Κώδικα Οδικής Κυκλοφορίας

Η αύξηση των παραβιάσεων του κώδικα οδικής κυκλοφορίας είναι ένα πρόβλημα που αντιμετωπίζουν πολλές κοινωνίες παγκοσμίως. Οι πιο συχνές παραβιάσεις συνήθως περιλαμβάνουν:

- **Υπέρβαση των ορίων ταχύτητας:** Μία από τις πιο συνηθισμένες παραβάσεις είναι η υπέρβαση των ορίων ταχύτητας. Η οδήγηση με υπερβολική ταχύτητα αυξάνει τον κίνδυνο ατυχημάτων και μειώνει το χρόνο αντίδρασης σε επείγουσες καταστάσεις.
- **Χρήση κινητού τηλεφώνου κατά την οδήγηση:** Η απόσπαση της προσοχής από τον οδηγό λόγω χρήσης κινητού τηλεφώνου είναι εξαιρετικά επικίνδυνη και μπορεί να οδηγήσει σε σοβαρά ατυχήματα.
- **Παράβαση των πινακίδων STOP (Στοπ):** Η μη τήρηση των πινακίδων STOP αποτελεί σοβαρή παράβαση και μπορεί να οδηγήσει σε συγκρούσεις ή άλλα επικίνδυνα περιστατικά στα διασταυρούμενα οχήματα.
- **Παράβαση των κανόνων προτεραιότητας:** Η μη τήρηση των κανόνων προτεραιότητας μπορεί επίσης να οδηγήσει σε ατυχήματα και συγκρούσεις στα σημεία όπου οι οδηγοί πρέπει να δώσουν προτεραιότητα.



Figure 1 ΠΙΝΑΚΙΔΕΣ ΚΟΚ

1.1 β) Εκτίμηση Ταχύτητας και Παράβαση της Πινακίδας STOP: Υλοποίηση μέσω Βαθιάς Μάθησης

Σε αυτή την εργασία θα εστιάσουμε στην εκτίμηση ταχύτητας και την παράβαση της πινακίδας STOP. Θα χρησιμοποιήσουμε τεχνικές βαθιάς μηχανικής μάθησης για να επιτύχουμε αυτό το σκοπό. Αρχικά, θα εκπαιδεύσουμε ένα μοντέλο βαθιάς μάθησης χρησιμοποιώντας φωτογραφίες και βίντεο που περιέχουν πινακίδες STOP και αυτοκίνητα. Στη συνέχεια, θα χρησιμοποιήσουμε διαφορετικές εικόνες και βίντεο για να εντοπίσουμε τα αυτοκίνητα και να εκτιμήσουμε τις ταχύτητές τους και την αναγνώριση των πινακίδων STOP. Με αυτό τον τρόπο, θα μπορέσουμε να αναπτύξουμε ένα σύστημα που μπορεί να ανιχνεύει και να προβλέπει τη συμπεριφορά των οδηγών και των οχημάτων σε διάφορες οδικές συνθήκες.

1.1 γ) Ο ρόλος της τήρησης των κανόνων κυκλοφορίας στην οδική ασφάλεια

Το να τηρούνται οι κανόνες κυκλοφορίας αποτελεί έναν κρίσιμο παράγοντα για τη διασφάλιση της οδικής ασφάλειας. Οι κανόνες αυτοί καθορίζουν τις συμπεριφορές που πρέπει να τηρούν οι οδηγοί, οι πεζοί και οι άλλοι συμμετέχοντες στην οδική κυκλοφορία, με σκοπό να μειωθεί ο κίνδυνος ατυχημάτων και να διασφαλιστεί η ασφαλής μετακίνηση όλων. Η τήρηση αυτών των κανόνων συμβάλλει σημαντικά στην πρόληψη ατυχημάτων και την προστασία της ζωής και της υγείας των ανθρώπων που συμμετέχουν στην κυκλοφορία.

Οι κανόνες κυκλοφορίας περιλαμβάνουν πτυχές όπως οι ταχύτητες που πρέπει να τηρούνται, ο τρόπος στάθμευσης, ο σεβασμός των προτεραιοτήτων, οι σηματοδότες, οι περιορισμοί των πεζών και πολλά άλλα. Η μη τήρηση αυτών των κανόνων μπορεί να οδηγήσει σε επικίνδυνες καταστάσεις στο οδικό δίκτυο, όπως ατυχήματα, τραυματισμούς και ακόμα και θανάτους.

Επομένως, η τήρηση των κανόνων κυκλοφορίας αποτελεί βασική προϋπόθεση για την οδική ασφάλεια και την αποτελεσματική διαχείριση της κυκλοφορίας. Εφαρμόζοντας τους κανόνες αυτούς, οι οδηγοί συμβάλλουν ενεργά στη μείωση των ατυχημάτων και στη διασφάλιση της ασφαλούς και της κυκλοφορίας.

1.2 Αυτοματοποιημένη Επιτήρηση Κυκλοφορίας

Η επιτήρηση της κυκλοφορίας είναι ένα θέμα κρίσιμης σημασίας για τη διασφάλιση της οδικής ασφάλειας και την αποτελεσματική διαχείριση της κυκλοφοριακής ροής. Με την πρόοδο της τεχνολογίας, η επιτήρηση της κυκλοφορίας έχει εξελιχθεί σημαντικά, με τη χρήση αυτοματοποιημένων συστημάτων που βασίζονται σε τεχνολογίες όπως η τεχνητή νοημοσύνη και η μηχανική μάθηση.

Τα αυτοματοποιημένα συστήματα επιτήρησης της κυκλοφορίας χρησιμοποιούνται για την ανίχνευση και τον έλεγχο της κίνησης, την αναγνώριση παραβάσεων και τη διαχείριση της κυκλοφοριακής ροής. Αυτά τα συστήματα χρησιμοποιούν σύγχρονες τεχνολογίες όπως οι αλγόριθμοι αναγνώρισης αντικειμένων και η αυτόματη παρακολούθηση για να παρέχουν πληροφορίες που βοηθούν στην ασφαλή και αποτελεσματική λειτουργία των οδικών δικτύων. Με αυτόν τον τρόπο, η αυτοματοποιημένη επιτήρηση συμβάλλει στην αύξηση της οδικής ασφάλειας και τη βελτίωση της κυκλοφοριακής ροής

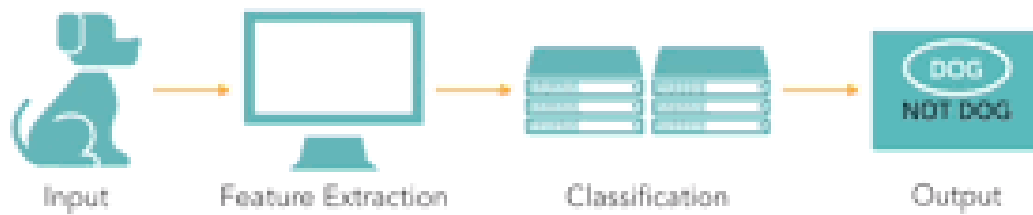


Figure 2 ΚΑΜΕΡΑ ΕΠΙΤΗΡΗΣΗΣ ΚΥΚΛΟΦΟΡΙΑΣ

1.2 Γιατί Βαθιά Μάθηση;

Η χρήση της μηχανικής μάθησης στην εργασία μας είναι ζωτικής σημασίας για την επίτευξη των στόχων μας. Μέσω της βαθιάς μάθησης, μπορούμε να εκπαιδεύσουμε μοντέλα που είναι ικανά να αναγνωρίζουν πρότυπα και να προβλέπουν συμπεριφορές με μεγάλη ακρίβεια. Στην περίπτωση μας, η χρήση βαθιάς μάθησης επιτρέπει στο σύστημά μας να ανιχνεύει την υπέρβαση ορίων ταχύτητας και την παράβαση της πινακίδας STOP με αξιόπιστο τρόπο. Μέσω της εκπαίδευσης με δεδομένα που περιλαμβάνουν πινακίδες STOP και οχήματα σε διάφορες καταστάσεις, το μοντέλο μας μπορεί να μάθει να αναγνωρίζει τέτοια πρότυπα και να προβλέπει τη συμπεριφορά των οδηγών και των οχημάτων. Με αυτόν τον τρόπο, η χρήση βαθιάς μάθησης μας επιτρέπει να δημιουργήσουμε ένα πολύπλοκο και αποτελεσματικό σύστημα που μπορεί να βοηθήσει στην αύξηση της οδικής ασφάλειας και στην πρόληψη ατυχημάτων.

TRADITIONAL MACHINE LEARNING



DEEP LEARNING



Figure 3 DEEP LEARNING VS MACHINE LEARNING

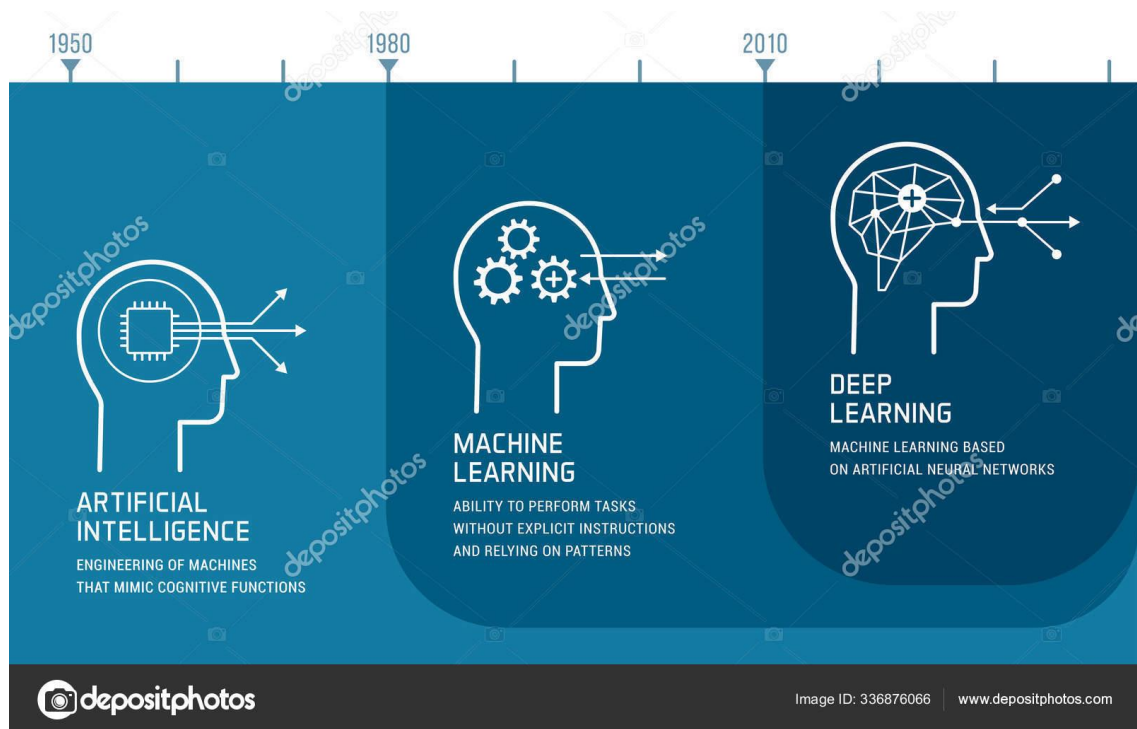


Figure 4 LEARNING STAGES

ΚΕΦΑΛΑΙΟ 2 ΜΗΧΑΝΙΚΗ ΜΑΘΗΜΗΣΗ ΚΑΙ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

2.1 Μηχανική Μάθηση και Πώς Λειτουργεί; [15]

Η μηχανική μάθηση (Machine Learning) είναι μια σημαντική υποκατηγορία της τεχνητής νοημοσύνης (Artificial Intelligence, AI) που έχει επαναπροσδιορίσει τον τρόπο με τον οποίο οι υπολογιστές αντιλαμβάνονται και αντιδρούν στα δεδομένα. Η ουσία της μηχανικής μάθησης είναι η ικανότητα του υπολογιστή να μαθαίνει από τα δεδομένα, ώστε να εξάγει μοτίβα, να κατανοεί τις σχέσεις μεταξύ αυτών και να προβλέπει τα μελλοντικά δεδομένα ή να λαμβάνει αποφάσεις με βάση τις προηγούμενες εμπειρίες του.

Με την αύξηση του όγκου των δεδομένων στη σύγχρονη εποχή, η ανάγκη για συστήματα που μπορούν να διαχειρίζονται αυτά τα πολύπλοκα σύνολα δεδομένων είναι πρωταρχική. Αυτό οφείλεται στην αναγνώριση της αξίας που κρύβουν τα δεδομένα και της δυνατότητας εξαγωγής σημαντικών πληροφοριών από αυτά. Σε αυτό το πλαίσιο, τα μεγάλα και πολύπλοκα δεδομένα, γνωστά ως Big Data, συχνά αντιμετωπίζονται και επεξεργάζονται από μοντέλα μηχανικής μάθησης.

Οι μηχανές εκπαιδεύονται με διάφορα δεδομένα, τα οποία μπορούν να προέλθουν από πολλαπλές πηγές, όπως αισθητήρες, κινητά τηλέφωνα, κοινωνικά δίκτυα, κ.λπ. Κατά τη διαδικασία εκπαίδευσης, οι μηχανές αναλύουν τα δεδομένα αυτά, εξάγουν συμπεράσματα και στη συνέχεια χρησιμοποιούν αυτά τα δεδομένα για να βελτιώσουν την απόδοσή τους και να επιτυγχάνουν ολοένα και πιο ακριβή αποτελέσματα με κάθε επανάληψη. Μέσω αυτής της διαδικασίας, οι αλγόριθμοι μηχανικής μάθησης είναι σε θέση να ανακαλύπτουν πρότυπα και να προβλέπουν τα δεδομένα με ακρίβεια, παρέχοντας ένα ισχυρό εργαλείο για τη λήψη αποφάσεων σε πολλούς τομείς της ανθρώπινης δραστηριότητας.

2.2 Κυριότερα Είδη Μηχανικής Μάθησης

2.2 α) Supervised Learning [14]

Η Supervised Learning (επιβλεπόμενη μάθηση) είναι ένα υποσύνολο του machine learning, όπου τα μοντέλα εκπαιδεύονται με ετικετοποιημένα σύνολα δεδομένων (labeled datasets). Τα ετικετοποιημένα σύνολα δεδομένων χρησιμοποιούνται για την εκπαίδευση αλγορίθμων που ταξινομούν δεδομένα ή κάνουν προβλέψεις με υψηλή ακρίβεια. Ο όρος "ετικετοποιημένο" υποδηλώνει ότι ορισμένα δεδομένα εισόδου έχουν ήδη τη σωστή έξοδο. Κάθε σημείο δεδομένων, συνεπώς, συσχετίζεται με την αντίστοιχη ετικέτα εξόδου. Η επιβλεπόμενη μάθηση μπορεί να χρησιμοποιηθεί για διάφορες περιπτώσεις, όπως αξιολόγηση κινδύνου, ανίχνευση απάτης, φιλτράρισμα ανεπιθύμητων μηνυμάτων (spam) και πολλά άλλα.

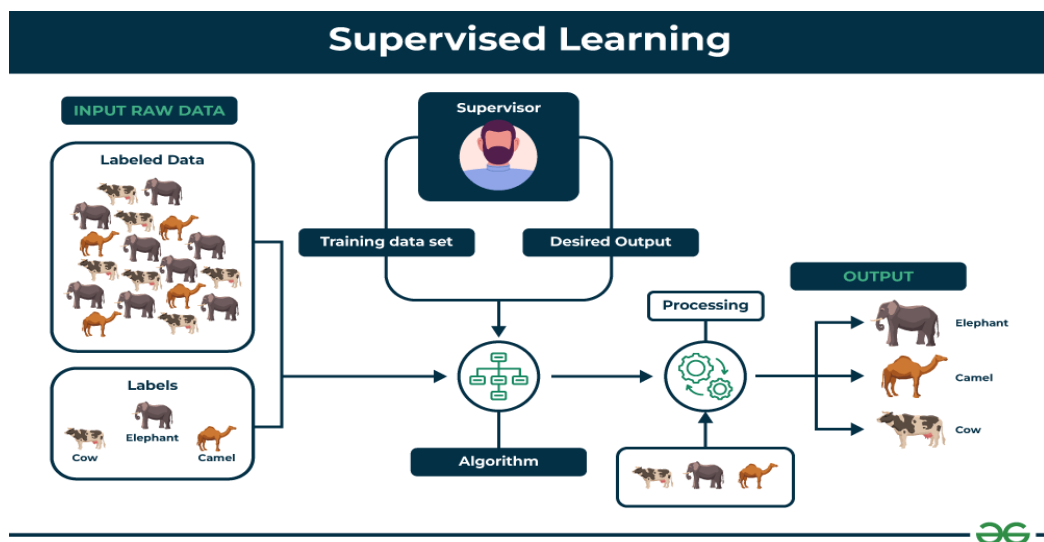


Figure 5 Supervised Learning

2.2 β) Unsupervised Learning [13]

Η Unsupervised Learning (Μη επιβλεπόμενη μάθηση) αναφέρεται σε μια κατηγορία τεχνικής μηχανικής μάθησης όπου αλγόριθμοι τεχνητής νοημοσύνης αναζητούν πρότυπα σε σύνολα δεδομένων χωρίς προκαθορισμένες κατηγορίες ή labels. Τα μοντέλα αυτά δεν απαιτούν επίβλεψη κατά την εκπαίδευση με δεδομένα, καθιστώντας την ιδανική για την εντοπισμό προτύπων, ομάδων και διαφορών σε μη δομημένα σύνολα δεδομένων. Είναι κατάλληλη για διαδικασίες όπως ο διαχωρισμός πελατών, η ανάλυση ανακάλυψης δεδομένων ή η αναγνώριση εικόνων.

Οι αλγόριθμοι Unsupervised Learning μπορούν να οργανώσουν και να ομαδοποιήσουν τα δεδομένα χωρίς εξωτερική καθοδήγηση

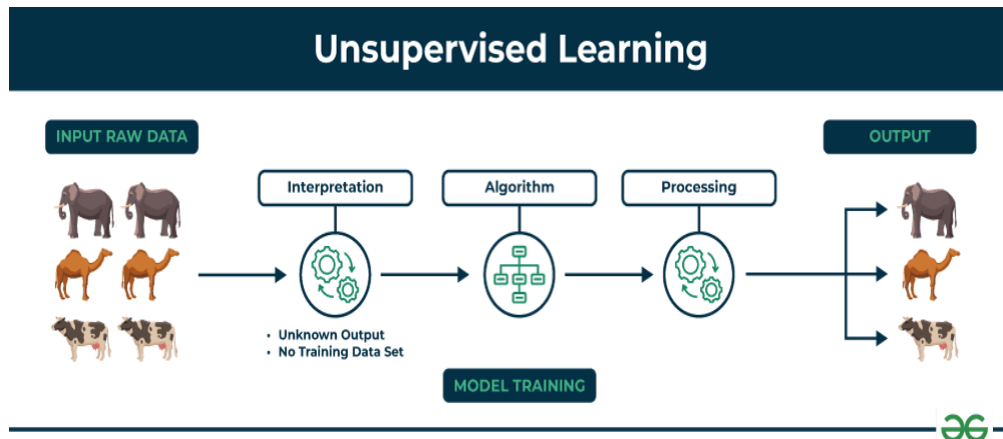


Figure 6 Unsupervised Learning

2.2 γ) Reinforcement Learning [16]

Η Reinforcement Learning (Ενισχυτική μάθηση) είναι η επιστήμη μεθοδολογία της λήψης αποφάσεων, που επιδιώκει να μάθει πώς να συμπεριφέρεται βέλτιστα σε ένα περιβάλλον για να αποκτήσει τη μεγαλύτερη δυνατή ανταμοιβή. Αυτή η βέλτιστη συμπεριφορά αποκτάται μέσω της δοκιμής και του λάθους, αντίστοιχα με το πώς τα παιδιά εξερευνούν και μαθαίνουν τις ενέργειες που τους οδηγούν σε επιτυχία. Χωρίς επόπτη, ο μαθητής πρέπει να ανακαλύψει ανεξάρτητα την καλύτερη ακολουθία ενεργειών που θα του φέρουν την μέγιστη ανταμοιβή, κάτι που γίνεται με την παρατήρηση των αντιδράσεων του περιβάλλοντος. Η ισχύς της ενίσχυσης μάθησης έγκειται στην ικανότητά της να μαθαίνει από την εμπειρία χωρίς εξωτερική καθοδήγηση.



REINFORCEMENT LEARNING MODEL

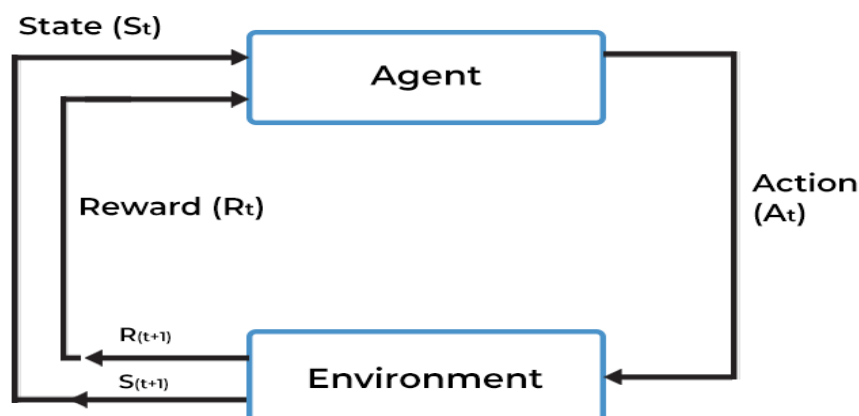


Figure 7 Reinforcement Learning

2.3 Τι είναι τα Νευρωνικά Δίκτυα; [3],[7]

Τα νευρωνικά δίκτυα, μια καινοτόμος κατηγορία αλγορίθμων μηχανικής μάθησης, εμπνέονται από τη δομή και τη λειτουργία του ανθρώπινου νευρικού συστήματος και αποτελούν σημαντικό τομέα στις φυσικές και τεχνολογικές επιστήμες. Αναπτύχθηκαν εντατικά τα τελευταία σαράντα χρόνια, παρουσιάζοντας σημαντική άνθηση και πολλαπλές εφαρμογές.

Αυτή η τεχνολογία αποτελείται από στρώματα διασυνδεδεμένων τεχνητών νευρώνων που επεξεργάζονται δεδομένα και παράγουν αποτελέσματα με βάση τα εισερχόμενα δεδομένα. Η ικανότητα αυτή των νευρωνικών δικτύων να μιμούνται την ανθρώπινη λογική και σκέψη, συνδυάζει μαθηματικούς υπολογισμούς με βιολογικές αρχές, προσφέροντας έναν καινοτόμο τρόπο για την επίλυση περίπλοκων προβλημάτων.

Η φιλοσοφία πίσω από τα νευρωνικά δίκτυα διαφέρει από τους κλασικούς υπολογιστές, καθώς ενσωματώνουν ιδέες που παραδοσιακά αποδίδονται στην ανθρώπινη σκέψη, όπως η μάθηση, η μνήμη και η λήψη, κάνοντας τα ιδανικά για εφαρμογές που απαιτούν έξυπνη αυτοματοποίηση και αποφάσεις. Τα νευρωνικά δίκτυα επιτρέπουν στους υπολογιστές να επεξεργάζονται και να αναλύουν πληροφορίες με τρόπο που προηγουμένως θεωρούνταν εφικτός μόνο για τον άνθρωπο, ανοίγοντας νέους ορίζοντες στην τεχνολογική ανάπτυξη.

2.4 Πώς λειτουργούν τα Νευρωνικά Δίκτυα;

Τα τεχνητά νευρωνικά δίκτυα αποτελούνται από τεχνητούς νευρώνες, που είναι κόμβοι λογισμικού οργανωμένοι σε δίκτυα, και χρησιμοποιούνται σε αλγορίθμους για την επίλυση διαφόρων υπολογιστικών και προβληματικών καταστάσεων. Ένα απλό νευρωνικό δίκτυο αποτελείται από τρία βασικά επίπεδα:

Επίπεδο εισόδου (input layer): Είναι το πρώτο επίπεδο του νευρωνικού δικτύου όπου γίνεται η πρωταρχική λήψη των δεδομένων. Οι κόμβοι εδώ αναλύουν και προωθούν τα δεδομένα στο επόμενο στάδιο.

Κρυφό επίπεδο (hidden layer): Τα κρυφά επίπεδα είναι τα μεσαία στρώματα που βρίσκονται ανάμεσα στο επίπεδο εισόδου και εξόδου, όπου πραγματοποιείται η πλειονότητα των υπολογισμών. Μπορεί να υπάρχουν πολλά τέτοια επίπεδα σε ένα νευρωνικό δίκτυο.

Επίπεδο εξόδου (output layer): Το τελικό επίπεδο του νευρωνικού δικτύου αποτελεί την έξοδο του συστήματος, με τον αριθμό των νευρώνων σε αυτό το επίπεδο να εξαρτάται από τη φύση του εκάστοτε προβλήματος που επιλύεται.

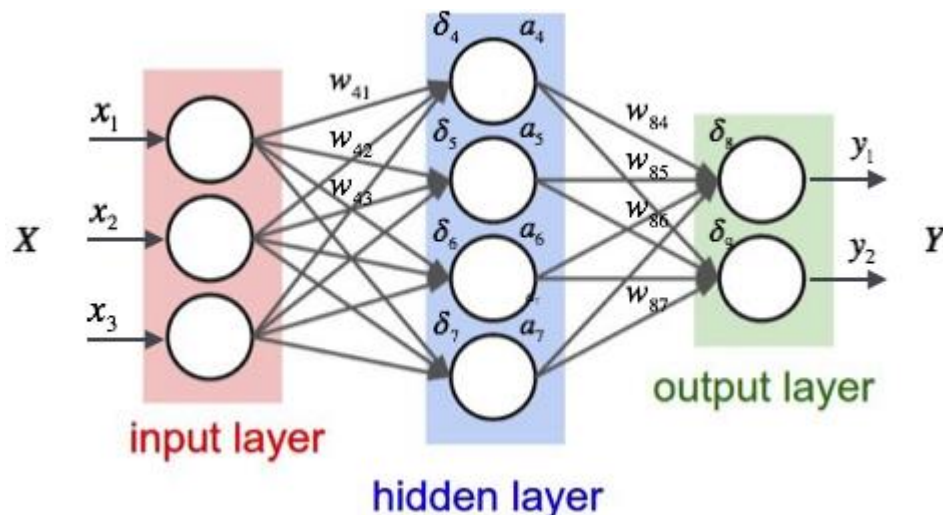


Figure 8 Neural Network Layers

2.5 Είδη Νευρωνικών Δικτύων

2.5 α) CNN (Convolutional Neural Network)

Τα Convolutional Neural Network (CNNs) είναι ανάλογα με τα παραδοσιακά ANNs(Artificial Neural Networks) στο ότι αποτελούνται από νευρώνες που αυτο-βελτιστοποιούνται μέσω της μάθησης. Κάθε νευρώνας θα λαμβάνει ακόμη μια είσοδο και θα εκτελεί μια λειτουργία (όπως ένα σκαλαρικό γινόμενο ακολουθούμενο από μια μη γραμμική λειτουργία) - η βάση αμέτρητων ANNs. Από την αρχική είσοδο της ακατέργαστης εικόνας μέχρι στην τελική έξοδο του class score, όλο το δίκτυο θα εκφράζει ακόμα μια ενιαία λειτουργία αντίληψης (the weight). Το τελευταίο layer θα περιλαμβάνει λειτουργίες απωλειών που σχετίζονται με τις κλάσεις, και όλες οι συνηθισμένες συμβουλές και τεχνικές που αναπτύχθηκαν για τα παραδοσιακά ANNs εφαρμόζονται ακόμα. Τα CNNs αποτελούνται από τρεις τύπους layers. Αυτά είναι το layer συνέλιξης, το layer της ομαδοποίησης και τα πλήρως συνδεδεμένα layers. Όταν αυτά τα επίπεδα είναι στοιβαγμένα, έχει δημιουργηθεί μια αρχιτεκτονική CNN.

Η βασική διαφορά μεταξύ των CNNs και των παραδοσιακών ANNs είναι η εφαρμογή τους στην αναγνώριση μοτίβων εντός εικόνων. Αυτό επιτρέπει την ενσωμάτωση ειδικών χαρακτηριστικών εικόνας στην αρχιτεκτονική του δικτύου, καθιστώντας το πιο αποτελεσματικό για εργασίες που εστιάζουν στις εικόνες και μειώνοντας ταυτόχρονα τους απαραίτητους παραμέτρους για τη διαμόρφωση του μοντέλου.

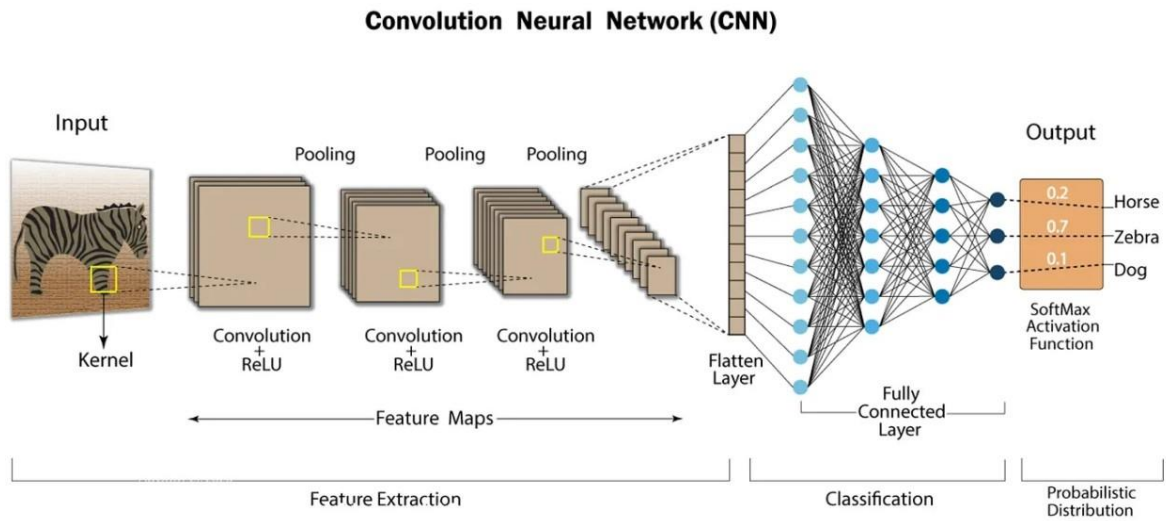


Figure 9 Convolution Neural Network 1

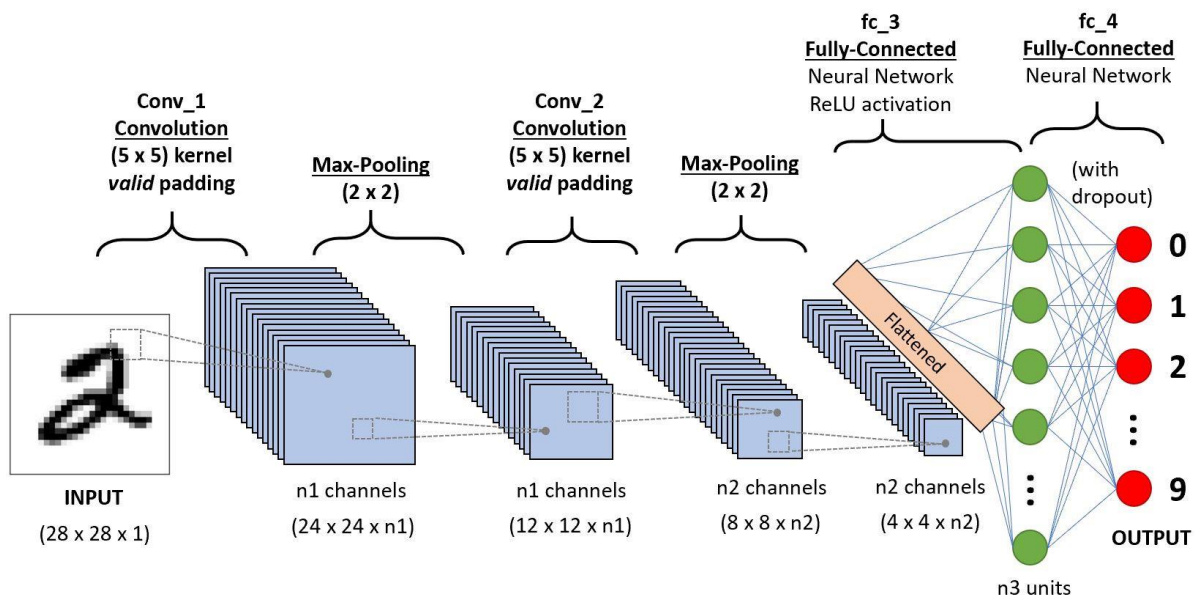


Figure 10 Convolution Neural Network 2

2.5 β) RNN (Recurrent Neural Networks)

Τα Recurrent Neural Networks (RNNs) είναι ένας τύπος αρχιτεκτονικής νευρωνικών δικτύων που χρησιμοποιείται κυρίως για τον εντοπισμό μοτίβων σε μια ακολουθία δεδομένων. Τα δεδομένα αυτά μπορεί να είναι χειρόγραφα, γονιδιώματα, κείμενο ή αριθμητικές χρονοσειρές που συχνά παράγονται σε βιομηχανικά περιβάλλοντα (π.χ. χρηματιστηριακές αγορές ή αισθητήρες). Ωστόσο, είναι εφαρμόσιμα και σε εικόνες εάν αυτές αναλυθούν ανάλογα σε μια σειρά από τμήματα και αντιμετωπιστούν ως ακολουθία. Ενώ τα Δίκτυα Feedforward μεταφέρουν πληροφορίες μέσω του δικτύου χωρίς κύκλους, τα RNN έχουν κύκλους και μεταδίδουν πληροφορίες πίσω στον εαυτό τους. Αυτό τους επιτρέπει να επεκτείνουν τη λειτουργικότητα των Δικτύων Feedforward, ώστε να λαμβάνουν υπόψη τους όχι μόνο την τρέχουσα είσοδο X_t , αλλά και προηγούμενες εισόδους $X_{0:t-1}$.

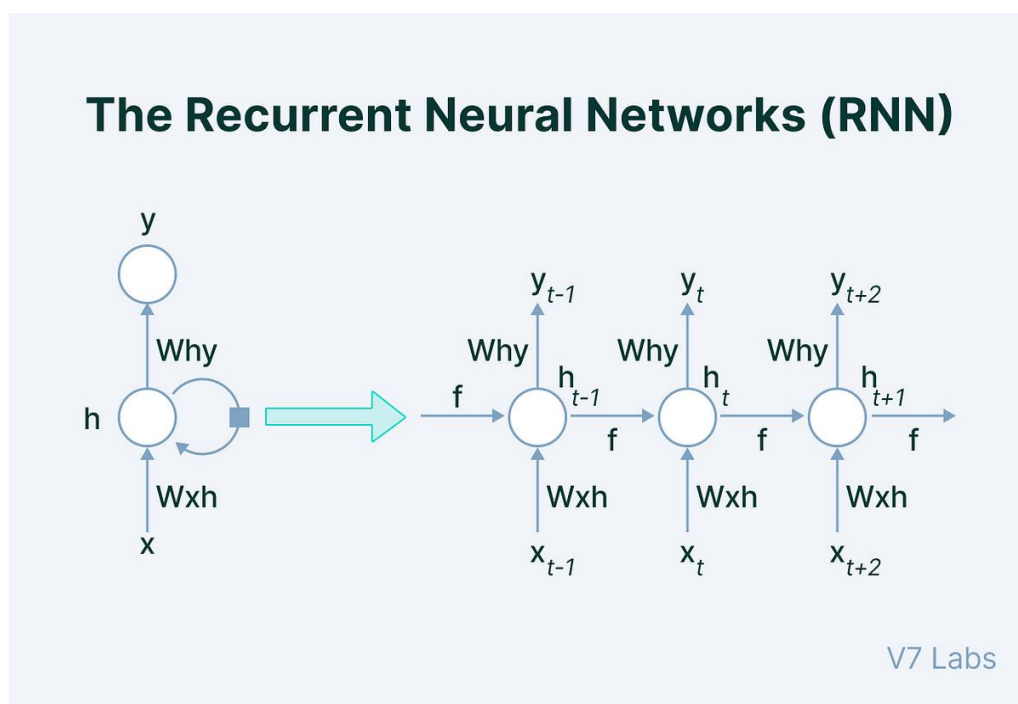


Figure 11 Recurrent Neural Networks

2.5 γ) MPLS (Multiprotocol Labels Switching) [27]

Το Multiprotocol Labels Switching, ή MPLS, είναι μια τεχνολογία δικτύωσης που δρομολογεί την κίνηση χρησιμοποιώντας την πιο σύντομη διαδρομή βάσει "labels", αντί για διευθύνσεις δικτύου, για να χειριστεί την προώθηση πάνω σε ιδιωτικά ευρείας περιοχής δίκτυα. Ως μια κλιμακούμενη και ανεξάρτητη από το πρωτόκολλο λύση, το MPLS αναθέτει ετικέτες σε κάθε πακέτο δεδομένων, ελέγχοντας την πορεία που ακολουθεί το πακέτο. Το MPLS βελτιώνει σημαντικά την ταχύτητα της κίνησης, ώστε οι χρήστες να μην βιώνουν καθυστερήσεις όταν συνδέονται στο δίκτυο.

Ένα παραδειγμα χρήσης του MPLS είναι εξηγώντας πώς ταξιδεύουν τα δεδομένα μέσω του διαδικτύου. Όταν στέλνετε ένα email, συνδέεστε σε VoIP ή τηλεδιάσκεψη, το δεδομένο ή το πακέτο IP αποστέλλεται από έναν δρομολογητή του διαδικτύου στον προορισμό του. Ο δρομολογητής πρέπει να αποφασίσει για κάθε πακέτο IP/δεδομένων πώς θα σταλεί στη διεύθυνση IP προορισμού. Κάθε πακέτο απαιτεί μια απόφαση, την οποία ο δρομολογητής παίρνει χρησιμοποιώντας περίπλοκους πίνακες δρομολόγησης για να την καθορίσει. Κάθε διαδρομή που φτάνει το πακέτο απαιτεί μια άλλη απόφαση προώθησης μέχρι να φτάσει στον προορισμό του. Αυτή η διαδικασία μπορεί να οδηγήσει σε κακή απόδοση για τους χρήστες, τις εφαρμογές που χρησιμοποιούν και να επηρεάσει το δίκτυο σε όλο τον οργανισμό. Το MPLS προσφέρει μια εναλλακτική για τις οργανώσεις για να αυξήσουν την απόδοση του δικτύου και να βελτιώσουν την εμπειρία του χρήστη.

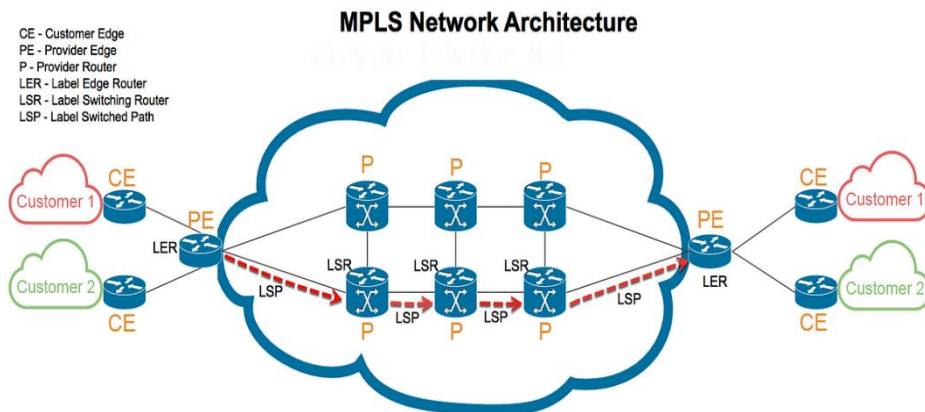


Figure 12 Multiprotocol Labels Switching

ΚΕΦΑΛΑΙΟ 3 Αλγόριθμοι Εντοπισμού (Detection algorithms)

3.1 Object Detection (Εντοπισμός Αντικειμένων) [20]

Η ανίχνευση και η κατηγοριοποίηση αντικειμένων σε μια μόνο εικόνα είναι ο στόχος της γενικής ανίχνευσης αντικειμένων, η οποία επίσης επισημαίνει τα αντικείμενα με ορθογώνια περιοχές που υποδεικνύουν την παρουσία τους με σιγουριά. Ένας από τους σημαντικότερους στόχους στην υπολογιστική όραση είναι η ανίχνευση αντικειμένων, η οποία αναζητά και αναγνωρίζει αντικείμενα μέσα σε μια εικόνα ή ένα καρέ βίντεο. Προχωρά πέρα από την απλή ταξινόμηση εικόνων με το να αναγνωρίζει όχι μόνο τι υπάρχει σε μια εικόνα, αλλά και να προσδιορίζει με ακρίβεια πού βρίσκεται κάθε αντικείμενο. Πολλές εφαρμογές, όπως η αυτόνομη οδήγηση, η παρακολούθηση, η ανάκτηση εικόνων και η επαυξημένη πραγματικότητα, εξαρτώνται από αυτό το χαρακτηριστικό. Σε αυτό το έργο θα χρησιμοποιηθεί για να αναγνωριστούν με ακρίβεια και να εντοπιστούν πολύμορφοι πολιποί του παχέος εντέρου.

Μια από τις βασικές προσεγγίσεις για την ανίχνευση αντικειμένων βασίζεται στη βαθιά μάθηση, ιδιαίτερα στα συνελκτικά νευρωνικά δίκτυα (CNNs). Τα CNNs ξεχωρίζουν στην εκμάθηση ιεραρχικών χαρακτηριστικών από τα ακατέργαστα δεδομένα pixel, κάνοντάς τα κατάλληλα για εργασίες όπως η ανίχνευση αντικειμένων. Μοντέλα όπως τα Faster R-CNN, YOLO (You Only Look Once) και SSD (Single Shot MultiBox Detector) έχουν κερδίσει αναγνώριση λόγω της αποτελεσματικότητάς τους και της ακρίβειάς τους στην ανίχνευση αντικειμένων σε πραγματικό χρόνο.

Συνήθως υπάρχουν αρκετά βήματα που εμπλέκονται στη διαδικασία ανίχνευσης αντικειμένων. Για την εξαγωγή χαρακτηριστικών από την εισαγόμενη εικόνα, χρησιμοποιείται ένα μοντέλο CNN που έχει ήδη εκπαιδευτεί. Οι επόμενοι επίπεδοι χρησιμοποιούν αυτές τις ιδιότητες για να προβλέψουν τις ετικέτες τάξης και τα ορθογώνια πλαίσια για κάθε αντικείμενο. Για τη βελτίωση της ακρίβειας και την αφαίρεση διπλών ανιχνεύσεων από την τελική συλλογή ανιχνευμένων αντικειμένων, συχνά χρησιμοποιούνται τεχνικές μετεπεξεργασίας όπως η μη-μέγιστη κατάθλιψη.

Οι στόχοι των πρόσφατων εξελίξεων στην ανίχνευση αντικειμένων ήταν η αύξηση της ανθεκτικότητας, της ακρίβειας και της ταχύτητας. Σημαντικές βελτιώσεις στην απόδοση έχουν γίνει δυνατές με τη χρήση μεθόδων όπως μηχανισμοί προσοχής, βελτίωση του κουτιού-άγκυρας και δίκτυα πυραμίδων χαρακτηριστικών (FPNs Feature Pyramid Networks). Επιπλέον, οι συστήματα ανίχνευσης αντικειμένων μπορούν τώρα να λειτουργούν με μεγαλύτερη επιτυχία σε δύσκολα πλαίσια με διάφορες συνθήκες φωτισμού και κρυψίματα χάρη στην ενσωμάτωση της βαθιάς μάθησης με άλλες τεχνολογίες όπως τα ραντάρ και τα LiDAR. Η ανίχνευση αντικειμένων έχει τη δυνατότητα να μετασχηματίσει πολλούς κλάδους και να ενθαρρύνει τις προόδους στην έρευνα της υπολογιστικής όρασης καθώς εξελίσσεται περαιτέρω.

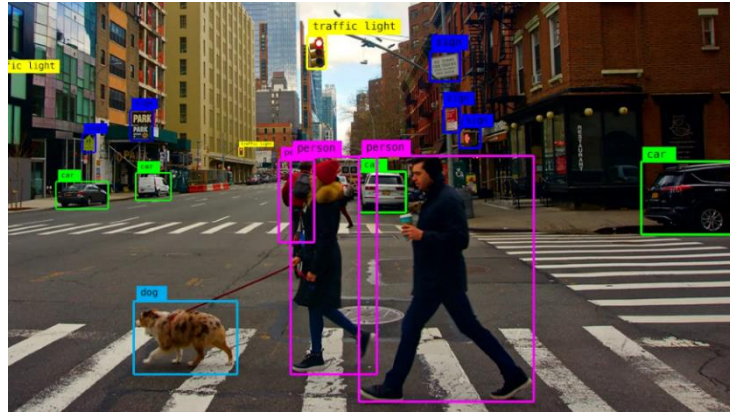


Figure 13 Object Detection with Bounding Boxes

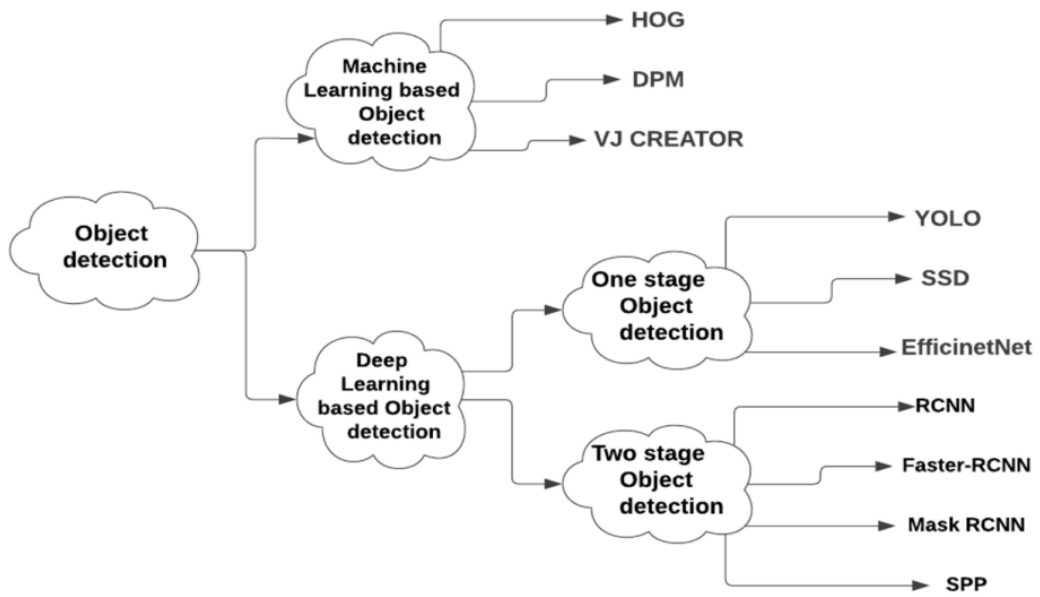


Figure 14 Εξέλιξη Αλγορίθμων Εντοπισμού

3.2 Προηγούμενες Εργασίες Εντοπισμού στους Δρόμους

3.2 α) Εργασία Εντοπισμού Αυτοκινήτων [30]

Το 2022, καθηγητές του SDU (Shandong University) ασχολήθηκαν με μια παρόμοια εργασία για εντοπισμό αυτοκινήτων και παρέχουν την υλοποίηση και την επίδειξη της χρήσης του αλγορίθμου YOLO για την ανίχνευση των γύρω αυτοκινήτων και αντικειμένων. Εκπαιδεύσαμε το μοντέλο για 81 κατηγορίες (Αυτοκίνητο, Πυροσβεστικός Κρουνός, Δέντρο, Άνθρωποι). Ο αλγόριθμος YOLO ξεκινά με την εξαγωγή μιας εικόνας από μια ροή βίντεο, η οποία στη συνέχεια αλλάζει μέγεθος σε 416x416 για να αποτελέσει την είσοδο στο δίκτυο YOLO v3. Το δίκτυο αυτό έχει 106 στρώσεις, περιλαμβάνοντας συνελκτικές, residual, upsampling στρώσεις και συνδέσεις παράκαμψης. Από την εικόνα εισόδου, ο αλγόριθμος επιστρέφει έναν τανυστή που περιλαμβάνει τις συντεταγμένες και τις θέσεις των πλαισίων περιγράμματος, την πιθανότητα ότι κάθε πλαίσιο περιγράμματος περιέχει ένα αντικείμενο, καθώς και τις πιθανότητες και το δείκτη εμπιστοσύνης για την κατηγορία του αντικειμένου. Η ανίχνευση πραγματοποιείται σε τρεις κλίμακες (13x13, 26x26, 52x52) για καλύτερη ανίχνευση μικρών αντικειμένων. Το δίκτυο μπορεί να ανιχνεύει πολλαπλά αντικείμενα ταυτόχρονα και η εκπαίδευσή του περιλαμβάνει την ανάλυση ολόκληρης της εικόνας για πιο ακριβείς προβλέψεις. Ο αλγόριθμος καταστέλλει τις λανθασμένες προβλέψεις χρησιμοποιώντας την καταστολή μη μέγιστης τιμής, διατηρώντας μόνο τις πιο αξιόπιστες προβλέψεις.



Figure 15 Παράδειγμα εργασίας α)

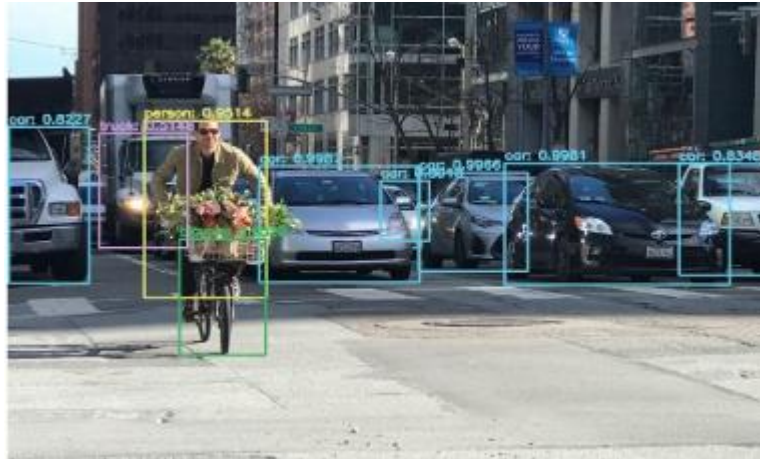


Figure 16 Παράδειγμα εργασίας α)

3.2 β)Εργασία Εκτίμηση Ταχύτητας [31]

Το 2022, καθηγητές του UCG (University of Montenegro) ασχολήθηκαν με μια παρόμοια εργασία [30] για την εκτίμηση ταχύτητας. Αυτό το άρθρο ασχολείται με την εκτίμηση της ταχύτητας οχημάτων χρησιμοποιώντας οπτικά δεδομένα από μια μόνο κάμερα βίντεο. Η μέθοδος προβλέπει με ακρίβεια την ταχύτητα ενός οχήματος χρησιμοποιώντας τον αλγόριθμο YOLO για την ανίχνευση και παρακολούθηση του οχήματος και ένα μονοδιάστατο συνελκτικό νευρωνικό δίκτυο (1D-CNN) για την εκτίμηση της ταχύτητας. Ο αλγόριθμος YOLO εξάγει πλαίσια περιγράμματος γύρω από τα ανιχνευμένα οχήματα, και η μέθοδος εισάγει ένα νέο χαρακτηριστικό που βασίζεται στην αλλαγή της περιοχής του πλαισίου περιγράμματος (CBBA) καθώς το όχημα πλησιάζει την κάμερα. Η μέθοδος εκπαιδεύτηκε και δοκιμάστηκε στο σύνολο δεδομένων VS13, δείχνοντας ότι μπορεί να προβλέψει την ταχύτητα του οχήματος με μέσο σφάλμα 2,76 χλμ/ώρα και με το καλύτερο όχημα να έχει μέσο σφάλμα μόλις 1,31 χλμ/ώρα. Η προτεινόμενη μέθοδος είναι στιβαρή και δεν απαιτεί προϋπάρχουσες γνώσεις διαστάσεων του πραγματικού κόσμου, όπως το μέγεθος του οχήματος ή η απόσταση και η γωνία της κάμερας.

Τα πειραματικά αποτελέσματα είναι πολύ υποσχόμενα. Η μέθοδος μπορεί να εκτιμήσει την ταχύτητα του οχήματος με μέσο σφάλμα 2,76 χλμ/ώρα. Επειδή το όχημα με την καλύτερη απόδοση έχει μέσο σφάλμα μόλις 1,31 χλμ/ώρα, που είναι κοντά στις μεθόδους ανίχνευσης ταχύτητας με ραντάρ, η μελλοντική έρευνα θα στοχεύσει στη βελτίωση της συνολικής ακρίβειας εκτίμησης εισάγοντας πρόσθετα χαρακτηριστικά. Επιπλέον, θα εξεταστεί η χρήση πολυτροπικής βαθιάς μάθησης, η οποία θα συνδυάζει δύο μορφές, την όραση και τον ήχο, με στόχο τη μείωση του σφάλματος εκτίμησης.

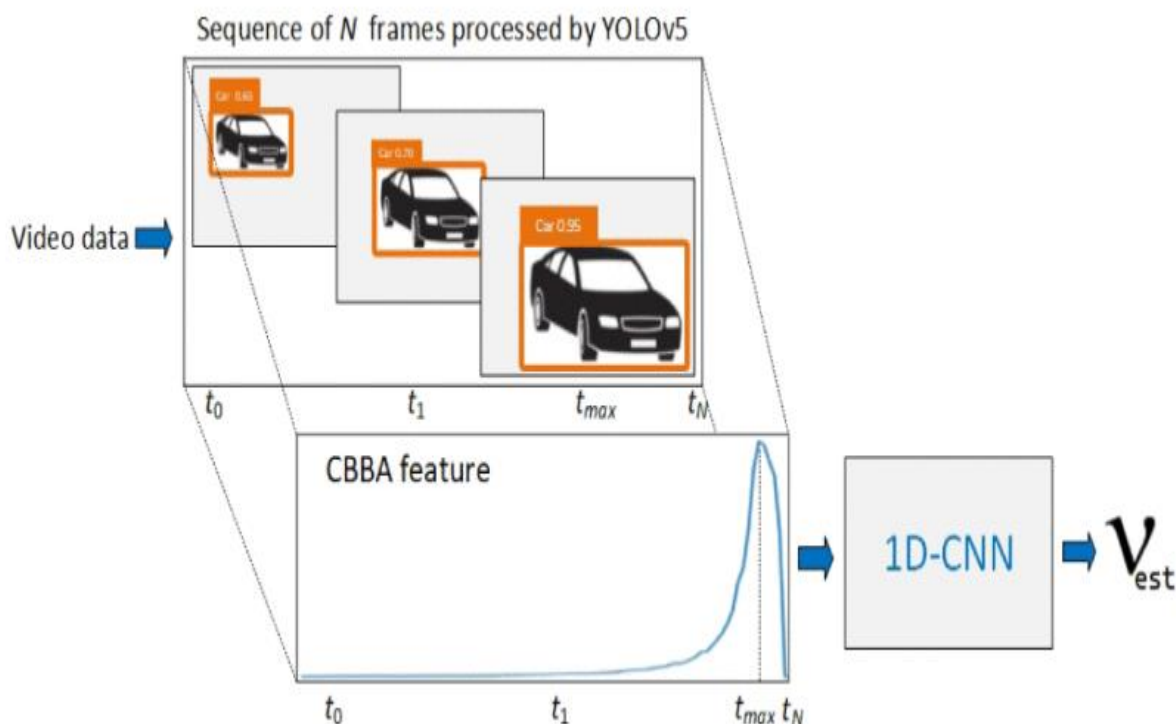


Figure 17 Τύπος Εργασίας β)

3.2 γ) Εργασία Εντοπισμού Σημάτων Τροχαίας [32]

Η εργασία που δημοσιεύτηκε το 2019 από τους Riadh Ayachi, Mouna Aff, Yahia Said και Mohamed Atri αναφέρεται στην ανάπτυξη ενός προηγμένου συστήματος ανίχνευσης τροχαίων σημάτων, χρησιμοποιώντας τεχνολογία βαθιάς μάθησης με συνελκτικά νευρωνικά δίκτυα. Το σύστημα στοχεύει στην αναγνώριση και κατανόηση των τροχαίων σημάτων στους δρόμους, προσφέροντας στον οδηγό μια ευρεία άποψη των σημάτων και βελτιώνοντας την οδική ασφάλεια. Η εφαρμογή χαρακτηρίζεται από την ικανότητά της να εντοπίζει διαφορετικά σήματα κάτω από πραγματικές περιβαλλοντικές συνθήκες, καθώς τα δεδομένα έχουν ληφθεί από τους δρόμους της Κίνας. Η εργασία παρουσιάζει την δημιουργία ενός συγκεκριμένου συνόλου δεδομένων που περιέχει 10,500 εικόνες από 73 κατηγορίες τροχαίων σημάτων, σχεδιασμένο για την εκπαίδευση του μοντέλου. Το μοντέλο βαθιάς μάθησης που αναπτύχθηκε καταφέρνει να επιτύχει υψηλή απόδοση, με μέσο όρο ακρίβειας πρόβλεψης 84,22%.

Αυτή η εργασία δοκιμάστηκε σε νέες εικόνες και απέδειξε την απόδοσή της με ακρίβεια ανίχνευσης πάνω από 80% εμπιστοσύνη ανά αντικείμενο. Η παρακάτω εικόνα παρουσιάζει την έξοδο της εφαρμογής ανίχνευσης τροχαίων σημάτων. Από την άλλη πλευρά, η εφαρμογή επιτυγχάνει μια ταχύτητα συμπερασματικής διαδικασίας 0,07 δευτερόλεπτα ανά εικόνα ή 15 καρέ ανά δευτερόλεπτο. Η επιτευχθείσα ταχύτητα συμπεράσματος επιτρέπει στην εφαρμογή να λειτουργεί για ανίχνευση σε πραγματικό χρόνο.



Figure 18 Αποτέλεσμα Εργασίας 3.2 γ)

3.3 Τύποι αλγορίθμων

3.3 α) Two-Stage Detectors [23],[24]

Οι Two-Stage Detectors βασίζονται σε υποψήφιες περιοχές (candidate regions). Αρχικά, παράγονται region proposals που δείχνουν την πιθανότητα τοποθέτισης ενός αντικειμένου. Στη συνέχεια λαμβάνει χώρα η διαδικασία ταξινόμησης αντικειμένων. Αυτοί οι αλγόριθμοι απαιτούν δύο στάδια, στο πρώτο στάδιο εξάγονται τα object proposals. Βελτιώνει τα object proposals δύο φορές, πράγμα που σημαίνει ότι αλγόριθμοι όπως τα Fast/Faster R-CNN μπορούν να λάβουν καλύτερα αποτελέσματα ανίχνευσης αντικειμένων σε σύγκριση με τους One-Stage Detectors. Το μόνο μειονέκτημα είναι ότι φαίνονται πολύ πιο αργοί από τους One-Stage Detectors και η διαδικασία εκπαίδευσης απαιτεί πολλή προσπάθεια επειδή αποτελεί πρόκληση η βελτιστοποίηση κάθε συστατικού του δικτύου.

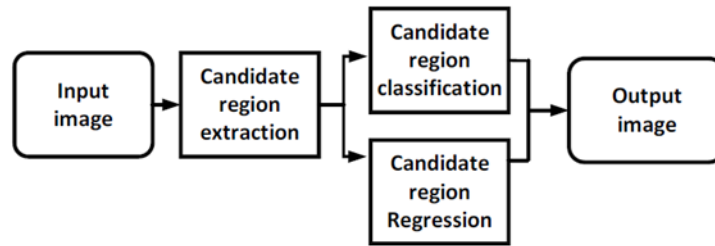


Figure 19 Two-Stage Detectors

3.3 β) One-Stage Detectors [6],[23],[25]

Οι One-Stage Detectors παράγουν τις προτάσεις περιοχών και προβλέπουν τα class probabilities σε μία αξιολόγηση. Έχουν ταχύτερη υπολογιστική ταχύτητα από τους Two-Stage Detectors, αλλά αυτό οδηγεί σε θυσία της ακρίβειας ανίχνευσης. Δεν έχουν πρόταση περιοχής, που σημαίνει ότι δεν διαχωρίζουν τη διαδικασία ανίχνευσης και πρότασης, καθιστώντας τη διαδικασία μονοφασική. Εναλλακτικά, είναι γνωστοί ως αλγόριθμοι ανίχνευσης αντικειμένων με βάση την παλινδρόμηση. Υποδεικνύει ότι το αντικείμενο στην εικόνα βρίσκεται και ταξινομείται αμέσως, και η εισαγόμενη εικόνα δεν υποβάλλεται πλέον σε επεξεργασία από την υποψήφια περιοχή.

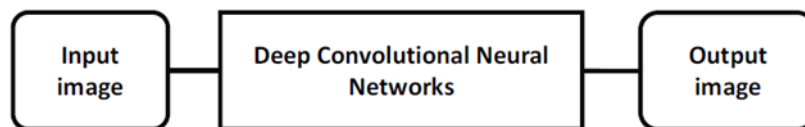


Figure 20 One-Stage Detectors

3.4 Αλγόριθμος YOLO [26],[27],[28]

3.4 α) Τι είναι ο αλγόριθμος Yolo;

Ο αλγόριθμος YOLO είναι ένας region-free one-stage αλγόριθμος ανίχνευσης, ο οποίος διαιρεί την εικόνα σε κελιά πλέγματος και θεωρεί κάθε κελί ως πρόταση για την ανίχνευση του αντικειμένου. Κάθε πλέγμα είναι υπεύθυνο μόνο για τα αντικείμενα των οποίων τα κέντρα βρίσκονται εντός του πλέγματος. Ο αλγόριθμος YOLO χρησιμοποιεί ένα νέο residual network που ονομάζεται Darknet, ένας παράγοντας που επιτρέπει καλύτερη εξαγωγή χαρακτηριστικών.

3.4 β) Πώς λειτουργεί;

Το δίκτυο προβλέπει κάθε bounding box για κάθε κλάση ταυτόχρονα, χρησιμοποιώντας χαρακτηριστικά από την πλήρη εικόνα. Πρώτα, η εισαγόμενη εικόνα διαιρείται σε ένα πλέγμα $S \times S$ από το σύστημα (για το YOLOv1, $S = 7$). Όταν το κέντρο ενός αντικειμένου πέφτει μέσα σε ένα κελί του πλέγματος, αυτό το κελί είναι υπεύθυνο για την αναγνώριση του αντικειμένου. Επίσης, κάθε κελί προβλέπει B πλαίσια περιορισμού καθώς και τους βαθμούς εμπιστοσύνης

για τα εν λόγω πλαίσια. Εάν δεν υπάρχει αντικείμενο σε ένα κελί, οι βαθμοί εμπιστοσύνης είναι ίσοι με μηδέν, διαφορετικά ο βαθμός εμπιστοσύνης είναι ίσος με το intersection over union (IOU) μεταξύ του predicted box και του ground truth.

Ένα bounding box αποτελείται από πέντε προβλέψεις: w (πλάτος πλαισίου), h (ύψος πλαισίου), x και y (θέση κέντρου πλαισίου), και εμπιστοσύνη (η οποία δηλώνει το IOU μεταξύ του predicted box και του ground truth box). Οι συντεταγμένες (x, y) είναι σε σχέση με τα όρια του κελιού πλέγματος, ενώ οι συντεταγμένες (w, h) είναι σε σχέση με το σύνολο της εικόνας.

Επιπλέον, κάθε κελί πλέγματος προβλέπει το C , τα conditional class probabilities ($\Pr(\text{Class}_i | \text{Object})$), οι οποίες εξαρτώνται από το κελί πλέγματος που περιέχει ένα αντικείμενο. Όταν έρθει η ώρα να δοκιμαστεί το δίκτυο, τα conditional class probabilities πολλαπλασιάζονται με τα confidence predictions του εκάστοτε πλαισίου, παρέχοντας τα confidence scores για κάθε πλαίσιο, συγκεκριμένα για κάθε κατηγορία. Το σκορ αυτό κωδικοποιεί και την πιθανότητα που η κατηγορία εμφανίζεται στο πλαίσιο καθώς και πόσο καλά ταιριάζει το προβλεπόμενο πλαίσιο στο αντικείμενο.

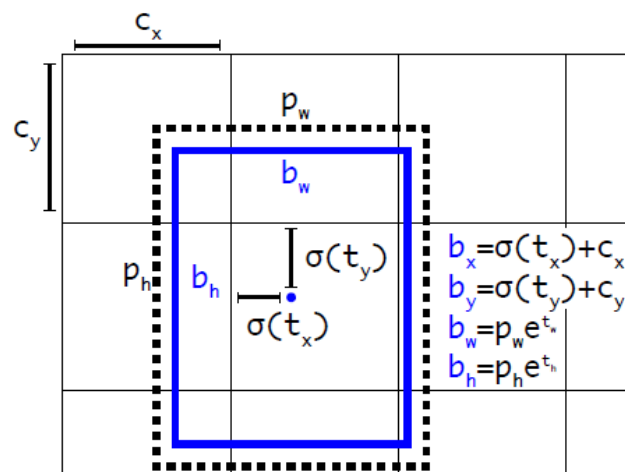


Figure 21 Yolo Bounding Boxes

Τα fully connected layers του δικτύου προβλέπουν τις πιθανότητες και τις συντεταγμένες εξόδου, ενώ τα αρχικά στρώματα συνέλιξης του δικτύου εξάγουν χαρακτηριστικά από την εικόνα. Αποτελείται από:

Backbone: Νευρωνικό δίκτυο συνέλιξης που μπορεί να δέχεται εικόνες διαφόρων μεγεθών και να διαμορφώνει τα συνολικά χαρακτηριστικά της εικόνας.

Neck: Σειρά από στρώματα δικτύου που συγχωνεύουν τα χαρακτηριστικά εικόνας που έχουν εξαχθεί από το backbone για να κάνουν την σημασιολογική πληροφορία πλουσιότερη και εξάγουν τα επεξεργασμένα χαρακτηριστικά ως την είσοδο του στρώματος πρόβλεψης.

Head: Προβλέπει τα εισαγόμενα χαρακτηριστικά. Ο classifier λαμβάνει την κατηγορία των αντικειμένων και παράγει τις τελικές συντεταγμένες του bounding box.

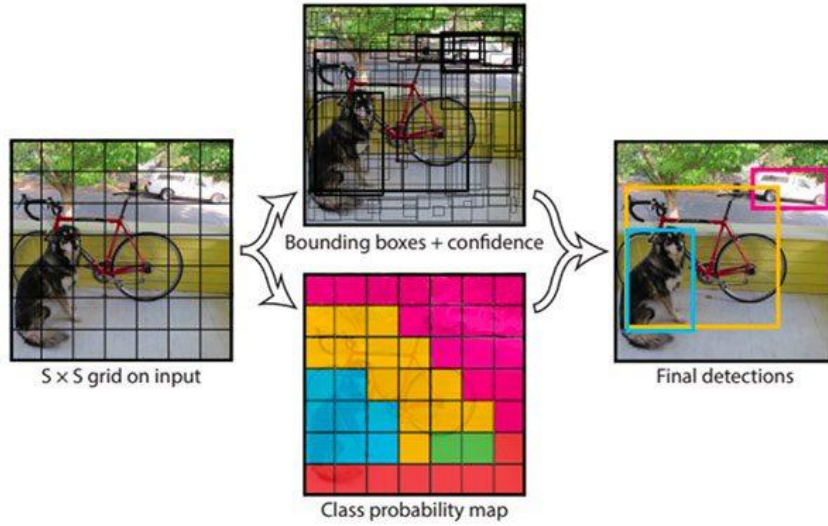


Figure 22 Χρήση Yolo

Δεδομένου ότι το backbone αποτελεί σημαντικό ποσοστό του υπολογιστικού κόστους, η σχεδιάσή του έχει κύριο αντίκτυπο στην αποδοτικότητα της συμπερασματολογίας. Το backbone επηρεάζει κυρίως την ικανότητα αναπαράστασης χαρακτηριστικών. Pyramid feature maps δημιουργούνται σε όλα τα επίπεδα συνδυάζοντας low-level physical information με high-level semantic features μέσω του neck. Το head, που αποτελείται από πολλά στρώματα συνέλιξης, προβλέπει τα αποτελέσματα ανίχνευσης με βάση τα χαρακτηριστικά πολλαπλών επιπέδων που έχει διαμορφώσει το neck. Από την άποψη της δομής, μπορεί να χωριστεί σε parameter-coupled head και parameter-decoupled head, or anchor-based και anchor-free.

Όσον αφορά το YOLOv5, η απώλεια υπολογίζεται ως ο συνδυασμός τριών μεμονωμένων συστατικών απώλειας: Classes Loss (BCE Loss), Objectness Loss (BCE Loss), και Location Loss (CloU Loss). Η συνάρτηση απώλειας είναι: Απώλεια = $\lambda_1 * L_{cls} + \lambda_2 * L_{obj} + \lambda_3 * L_{loc}$.

$$Loss_{obj} = Loss_{class} + Loss_{box} + Loss_{conf}$$

$$L_{class} = - \sum_{i=0}^{S*S} I_{ij}^{obj} \sum_{c \in classes} [\hat{P}_i^j \log P_i^j + (1 - \hat{P}_i^j) \log (1 - \hat{P}_i^j)]$$

$$L_{box} = 1 - CIoU$$

$$L_{conf} = - \sum_{i=0}^{S*S} I_{ij}^{obj} [\hat{M}_i^j \log M_i^j + (1 - \hat{M}_i^j) \log (1 - M_i^j)] - \lambda_{noobj} \sum_{i=0}^{S*S} \times \sum_{j=0}^N I_{ij}^{noobj} [\hat{M}_i^j \log M_i^j + (1 - \hat{M}_i^j) \log (1 - M_i^j)]$$

Figure 23 Loss YoloV5

3.6) YOLOv3 [33]

Λειτουργεί σε 22 ms με 28.2 mAP (320 x 320), που σημαίνει ότι έχει την ίδια ακρίβεια με το SSD αλλά είναι τρεις φορές πιο γρήγορο. Όπως και το YOLO9000, μπορεί να προβλέψει πλαίσια περιορισμού χρησιμοποιώντας ομάδες διαστάσεων και κουτιά-άγκυρες. Επιπλέον, προβλέπει έναν βαθμό αντικειμενικότητας για κάθε ένα από τα πλαίσια περιορισμού με τη χρήση λογιστικής παλινδρόμησης, που σημαίνει ότι θα πρέπει να είναι ίσο με 1 αν το προηγούμενο πλαίσιο περιορισμού ταιριάζει περισσότερο με ένα αντικείμενο αλήθειας σε σύγκριση με κάθε προηγούμενο πλαίσιο. Αν το προηγούμενο πλαίσιο δεν είναι το καλύτερο αλλά ταιριάζει με ένα αντικείμενο αλήθειας πάνω από κάποιο κατώφλι, η πρόβλεψη αγνοείται. Δεν χρησιμοποιείται SoftMax καθώς έχει κριθεί περιττό, αντ' αυτού χρησιμοποιούνται ανεξάρτητοι λογιστικοί ταξινομητές και κατά την εκπαίδευση χρησιμοποιείται η απώλεια δυαδικής διασταυρούμενης εντροπίας για τις προβλέψεις των κατηγοριών.

Μπορεί να προβλέψει πλαίσια σε 3 διαφορετικές κλίμακες. Ένας χάρτης χαρακτηριστικών συλλέγεται από το δίκτυο και στη συνέχεια συνδυάζεται με τα δείγματα με αναπροσαρμογή μέσω συγκέντρωσης, επιτρέποντας μας να λάβουμε σημαντικές σημασιολογικές πληροφορίες από τα χαρακτηριστικά με αναπροσαρμογή. Η συσταδοποίηση K-means χρησιμοποιείται ακόμη για τον προσδιορισμό των προτύπων πλαισίων περιορισμού, επιλέγονται 9 συστάδες μαζί με 3 κλίμακες αυθαίρετα, και τότε οι συστάδες διαιρούνται ισόμερα σε όλες τις κλίμακες (στο σύνολο δεδομένων COCO οι 9 συστάδες ήταν: (10×13), (16×30), (33×23), (30×61), (62×45), (59×119), (116×90), (156×198), (373×326)). Επηρεασμένο από το YOLOv2, χρησιμοποιείται μια υβριδική προσέγγιση μεταξύ του δικτύου του και του Darknet-19 που περιλαμβάνει επιτυχημένα συνεχόμενα στρώματα συνέλιξης 3 x 3 και 1 x 1, αλλά φαίνεται να έχει κάποιες συνδέσεις συντομεύσεων και μεγαλύτερο μέγεθος. Ονομάζεται Darknet-53 λόγω των 53 στρωμάτων συνέλιξης του, κάνοντάς το πιο ισχυρό από το Darknet-19 αλλά ακόμη πιο αποδοτικό από το ResNet-101 και ResNet-152.

Παρατηρήθηκε ότι καθώς το κατώφλι IOU αυξάνεται, η απόδοση μειώνεται δραστικά, υποδεικνύοντας ότι το YOLOv3 έχει δυσκολία στο να τοποθετήσει ακριβώς τα πλαίσια σε συνάρτηση με το αντικείμενο. Το YOLO είχε προηγουμένως προβλήματα με την αντιμετώπιση μικρών αντικειμένων. Τώρα, παρατηρούμε μια αντιστροφή αυτού του προτύπου. Με τις νέες προβλέψεις πολλαπλών κλιμάκων, βλέπουμε ότι το YOLOv3 έχει σχετικά υψηλή απόδοση APS. Ωστόσο, όταν πρόκειται για αντικείμενα μεσαίου και μεγαλύτερου μεγέθους, η απόδοσή του είναι σημαντικά χαμηλότερη.

3.6) YOLOv5 [26]

Η αρχιτεκτονική του αποτελείται από τρία κύρια συστατικά --> 1) Backbone: χρησιμοποιεί το CSP-Darknet53 που είχε χρησιμοποιηθεί προηγουμένως στο YOLOv4, 2) Neck: λειτουργεί ως σύνδεση μεταξύ του backbone και του κεφαλιού, στο YOLOv5 επιλέγονται το SPPF και το νέο CSP-PAN, και 3) Head: Χρησιμοποιεί το κεφάλι του YOLOv3 που παράγει το τελικό αποτέλεσμα. Οι αλλαγές που έγιναν σε σύγκριση με προηγούμενα δίκτυα είναι ότι η δομή Focus αντικαθίσταται από μια δομή Conv2d 6 x 6 με αποτέλεσμα η αυξημένη αποδοτικότητα

και η δομή SPP στο λαιμό αντικαθίσταται από το SPPF με αποτέλεσμα περισσότερο από διπλάσια ταχύτητα επεξεργασίας.

Όσον αφορά την ενίσχυση δεδομένων, το YOLOv5 χρησιμοποιεί την Mosaic Augmentation, μια τεχνική που συνδυάζει τέσσερις εικόνες εκπαίδευσης σε μία, ενθαρρύνοντας τα μοντέλα ανίχνευσης να χειρίζονται καλύτερα τις κλίμακες και τις μεταφορές αντικειμένων. Χρησιμοποιεί την Copy-Paste Augmentation, μια μέθοδο που αντιγράφει τυχαία τμήματα από μια εικόνα και τα επικολλά σε άλλη τυχαία επιλεγμένη εικόνα, δημιουργώντας έτσι ένα νέο δείγμα εκπαίδευσης. Χρησιμοποιεί επίσης Random Affine Transformations, που περιλαμβάνουν τυχαία περιστροφή, κλιμάκωση, μεταφορά και κοπή εικόνων. Χρησιμοποιεί την Ενίσχυση MixUP, δηλαδή δημιουργεί σύνθετες εικόνες παίρνοντας έναν γραμμικό συνδυασμό δύο εικόνων και των σχετικών ετικετών τους. Χρησιμοποιεί τις Albumentations, μια βιβλιοθήκη εικόνων για ενίσχυση εικόνων που υποστηρίζει ποικιλία τεχνικών ενίσχυσης, και την Ενίσχυση HSV που δημιουργεί τυχαίες αλλαγές στο Χρώμα, τον Κορεσμό και την Αξία των εικόνων. Τέλος, χρησιμοποιεί την Τυχαία Οριζόντια Αναστροφή, μια μέθοδο ενίσχυσης που μπορεί να αναστρέψει τυχαία τις εικόνες οριζόντια.

Όταν έρθει η ώρα για training, το YOLOv5 χρησιμοποιεί πλήθος στρατηγικών εκπαίδευσης για να ενισχύσει την απόδοση του μοντέλου περιλαμβάνοντας Multiscale Training, AutoAnchor, Warmup and Cosine LR Scheduler, Exponential Moving Average (EMA), Mixed Precision Training, και Hyperparameter Evolution.

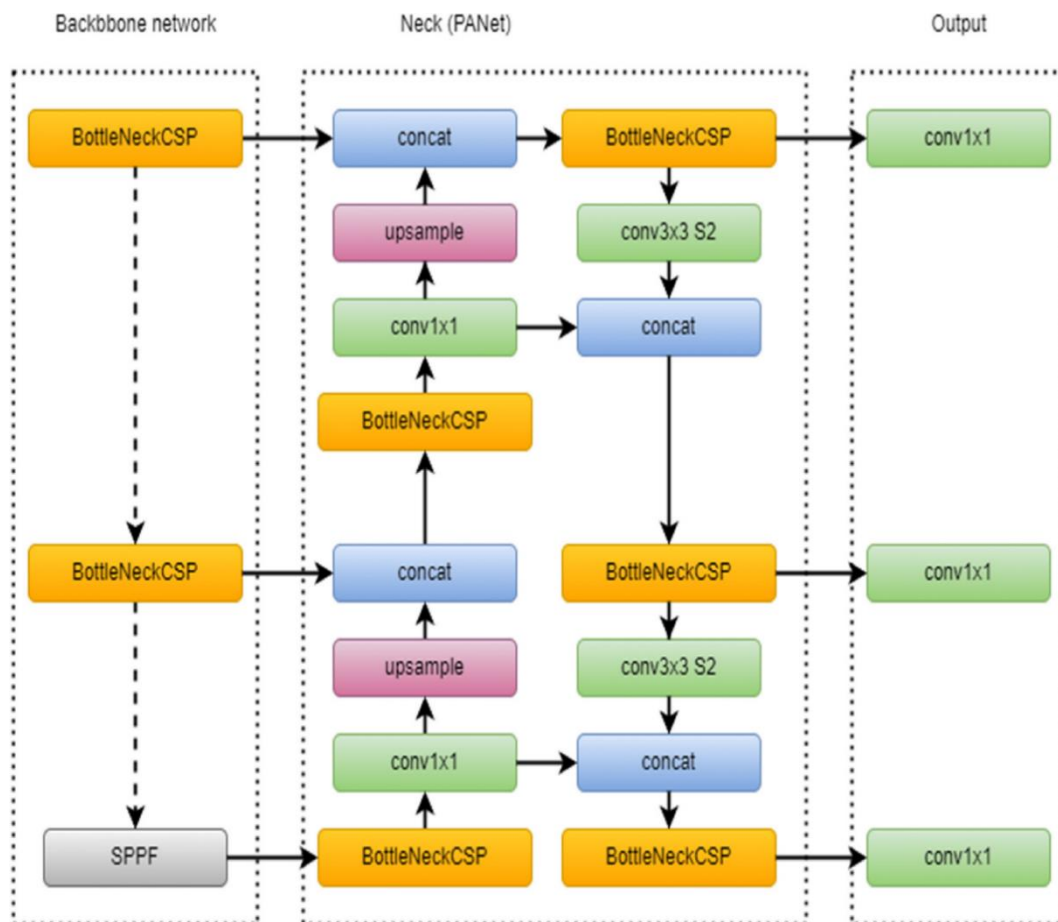


Figure 25 YOLOv5 Architecture

3.7) YOLOv9 [21]

Από Wang κ.ά., 2024 βασίζεται στο YOLOv7. Προτείνεται η έννοια του programmable gradient information (PGI) το οποίο μπορεί να παρέχει πλήρη εισερχόμενη πληροφορία για το στοχευμένο έργο με στόχο τον υπολογισμό της αντικειμενικής λειτουργίας. Έτσι, μπορούν να ληφθούν αξιόπιστες πληροφορίες κλίσης για την ενημέρωση των βαρών του δικτύου. Επίσης, χρησιμοποιεί μια νέα ελαφριά αρχιτεκτονική, το Generalized Efficient Layer Aggregation Network (G-ELAN) που βασίζεται στον προγραμματισμό της διαδρομής κλίσης. Για να επιτευχθεί μεγαλύτερη χρήση παραμέτρων από τις τεχνικές που δημιουργήθηκαν με βάση την κατακόρυφη συγκόλληση, το GELAN χρησιμοποιεί μόνο κλασικές λειτουργίες συγκόλλησης. Το PGI χρησιμοποιεί ένα auxiliary reversible branch για τη δημιουργία συνεπών κλίσεων, επιτρέποντας στα βαθιά χαρακτηριστικά να διατηρούν σημαντικές ιδιότητες που είναι απαραίτητες για την εκτέλεση του προβλεπόμενου έργου. Ο σημασιολογικός απολογισμός που θα μπορούσε να προκύψει από την ενσωμάτωση χαρακτηριστικών πολλαπλών διαδρομών σε μια τυπική διαδικασία βαθιάς επίβλεψης μπορεί να προληφθεί με τον σχεδιασμό ενός auxiliary reversible branch. Διαφορετικά ειπωμένο, βελτιστοποιούμε τα αποτελέσματα εκπαίδευσης προγραμματίζοντας τη διάδοση πληροφορίας κλίσης σε διάφορα σημασιολογικά επίπεδα. Επιτρέποντας την ελεύθερη επιλογή μιας λειτουργίας απώλειας που είναι κατάλληλη για την προβλεπόμενη δουλειά, το PGI επίσης λύνει τα ζητήματα που παρουσιάζει η μοντελοποίηση μάσκας.

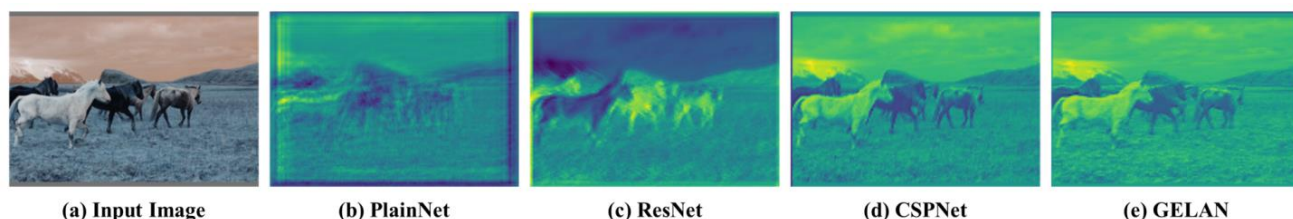


Figure 26 Yolov9 Stages

Το PGI επιλύει το ζήτημα ότι η βαθιά επίβλεψη περιορίζεται σε πολύ βαθιά νευρωνικά δίκτυα, καθιστώντας δυνατή την πραγματική υλοποίηση νέων ελαφριών αρχιτεκτονικών σε καθημερινές καταστάσεις. Για να επιτευχθεί υψηλότερη κατανάλωση παραμέτρων από το σχέδιο συγκόλλησης με βάση την πιο σοφιστική μηχανολογία, το GELAN που κατασκευάσαμε χρησιμοποιεί μόνο την παραδοσιακή συγκόλληση, ενώ επιδεικνύει εξαιρετικά πλεονεκτήματα ως ελαφρύ, γρήγορο και ακριβές. Η απόδοση ανίχνευσης αντικειμένων του YOLOv9 στο σύνολο δεδομένων MS COCO, όταν συνδυάζεται το προτεινόμενο PGI και GELAN, υπερέρχει σημαντικά των τρεχουσών ανιχνευτών αντικειμένων σε πραγματικό χρόνο σε κάθε τρόπο.

Το μονάδα Res2Net συγχωνεύει όλα τα μετατρεπόμενα τμήματα πριν τα περάσει πίσω και ενώνει διάφορα τμήματα εισόδου σε ιεραρχικό τρόπο με το επόμενο τμήμα. Για να ληφθούν όλες οι αρχικές πληροφορίες, το CBNet επαναφέρει τα αρχικά δεδομένα εισόδου χρησιμοποιώντας ένα σύνθετο κορμό. Στη συνέχεια χρησιμοποιεί μια ποικιλία τεχνικών σύνθεσης για να δημιουργήσει πολλά επίπεδα multilayer reversible information. Αν και αυτά

τα σχέδια δικτύου συνήθως έχουν πολύ καλή χρήση παραμέτρων, τα πρόσθετα σύνθετα επίπεδα οδηγούν σε αργούς χρόνους.

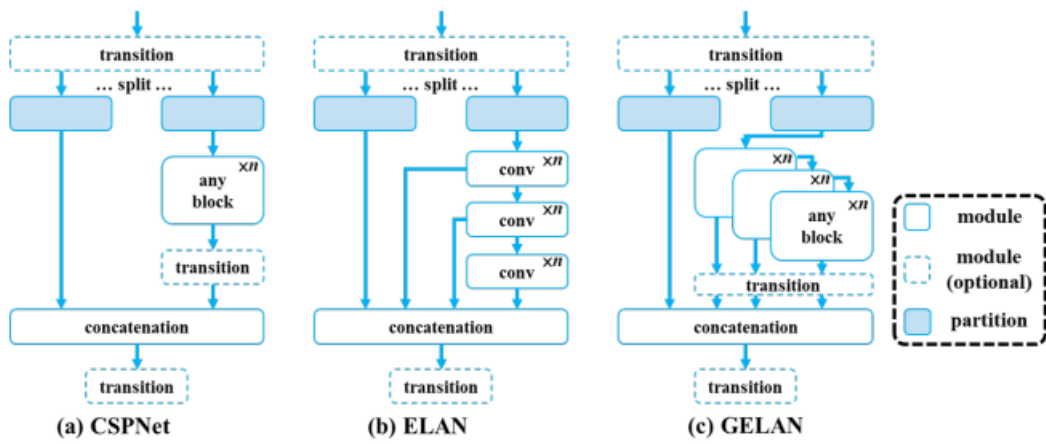


Figure 27 Yolov9 Stages

ΚΕΦΑΛΑΙΟ 4 Εκπαίδευση του αλγορίθμου Yolon5 Και Χρήση του Sort

4.1 Dataset

4.1 α) Τι είναι ένα Dataset; [12]

Ένα dataset, ή αλλιώς σύνολο δεδομένων, αποτελεί ένα βασικό στοιχείο στον τομέα της μηχανικής μάθησης και της τεχνητής νοημοσύνης. Αποτελείται από ένα σύνολο παραδειγμάτων, τα οποία μπορεί να είναι εικόνες, κείμενο, ήχος, βίντεο κλπ. Συνήθως, κάθε παράδειγμα στο dataset συνοδεύεται από μια ετικέτα ή αναφορά που περιγράφει την κατηγορία, την κλάση ή την ετικέτα που αντιστοιχεί σε αυτό. Τα datasets χρησιμοποιούνται για την εκπαίδευση και τον έλεγχο μοντέλων μηχανικής μάθησης. Κατά τη διαδικασία του training, τα μοντέλα μηχανικής μάθησης "διδάσκονται" χρησιμοποιώντας τα δεδομένα από το dataset, ενώ κατά τη διαδικασία του ελέγχου, αξιολογούνται η απόδοση και η ακρίβεια τους χρησιμοποιώντας επίσης το ίδιο dataset.

Τα datasets παίζουν καθοριστικό ρόλο στην επιτυχία των μοντέλων μηχανικής μάθησης, καθώς η ποιότητα και η ποσότητα των δεδομένων επηρεάζουν την ικανότητα του μοντέλου να εκπαιδευτεί και να γενικεύσει σωστά. Επιπλέον, η επιλογή του κατάλληλου dataset εξαρτάται από το είδος του προβλήματος που επιχειρεί να λύσει το μοντέλο, καθώς και από τον τύπο των δεδομένων που χρησιμοποιείται.

4.1 β) Δημιουργία Dataset για εντοπισμό αυτοκινήτων και πινακίδων STOP

Στην παρούσα εργασία, αναλαμβάνουμε τη δημιουργία ενός δικού μας dataset για την εκπαίδευση ενός μοντέλου YOLOv5 για τον ανίχνευση αυτοκινήτων και πινακίδων "Stop". Το dataset μας θα αποτελείται από εικόνες που περιέχουν αυτοκίνητα και πινακίδες "Stop" σε διάφορες περιβάλλοντες σκηνές κυκλοφορίας. Υπάρχουν όμως και προ-εκπαιδευμένα datasets που μπορούν να χρησιμοποιηθούν για διάφορα θέματα, συμπεριλαμβανομένων και εκείνων που αφορούν αυτοκίνητα και πινακίδες "Στοπ", η χρήση όμως προσαρμοσμένων datasets συνήθως παρέχει καλύτερη ευκαιρία για εξατομικευμένη εκπαίδευση του μοντέλου, βελτιώνοντας την απόδοση και τη γενική εφαρμοσιμότητά του.

Ο σκοπός μας είναι να συλλέξουμε μια ποικιλία εικόνων που καλύπτουν διάφορες συνθήκες φωτισμού, γωνίες λήψης και τύπους αυτοκινήτων και πινακίδων "Στοπ", προκειμένου να δημιουργήσουμε ένα ολοκληρωμένο και αντιπροσωπευτικό dataset. Επιπλέον, είναι σημαντικό να σημειωθεί ότι οι φωτογραφίες που θα χρησιμοποιηθούν στο dataset να είναι ελεύθερες προς χρήση, δηλαδή να διαθέτουν την κατάλληλη άδεια χρήσης που επιτρέπει τη χρήση τους για σκοπούς εκπαίδευσης μηχανικής μάθησης χωρίς περιορισμούς. Αυτό εξασφαλίζει ότι η χρήση των εικόνων είναι νόμιμη και δεν παραβιάζει τα πνευματικά δικαιώματα των δημιουργών τους.

Χρησιμοποιώντας τις φωτογραφίες που βρήκαμε και με τη χρήση του Roboflow, επιτεύχθηκε η κατάλληλη προεπεξεργασία για τη χρήση στην εκπαίδευση του YOLOv5.

Δηλώθηκαν οι κλάσεις "car" και "stop", χρησιμοποιώντας δύο για τις ανάγκες της εργασίας μας. Σε κάθε φωτογραφία επιλέχθηκε η περιοχή που περιέχει το αντικείμενο, ώστε να παραχθούν οι κατάλληλες συντεταγμένες για την εκπαίδευση, διαχωρίζοντας τις φωτογραφίες σε σύνολα εκπαίδευσης (train), επικύρωσης (validation) και δοκιμής (test). Αυτή η διαδικασία εξασφαλίζει ότι το μοντέλο YOLOv5 θα εκπαιδευτεί και θα ελεγχθεί αποτελεσματικά για την ανίχνευση των αντικειμένων "car" και "stop" στις εικόνες. Τέλος, έγινε η εξαγωγή του απαραίτητου αρχείου txt με τις συντεταγμένες, το οποίο περιέχει τις απαραίτητες πληροφορίες σχετικά με τη θέση και την κατηγορία των αντικειμένων σε κάθε εικόνα ξεχωριστά.

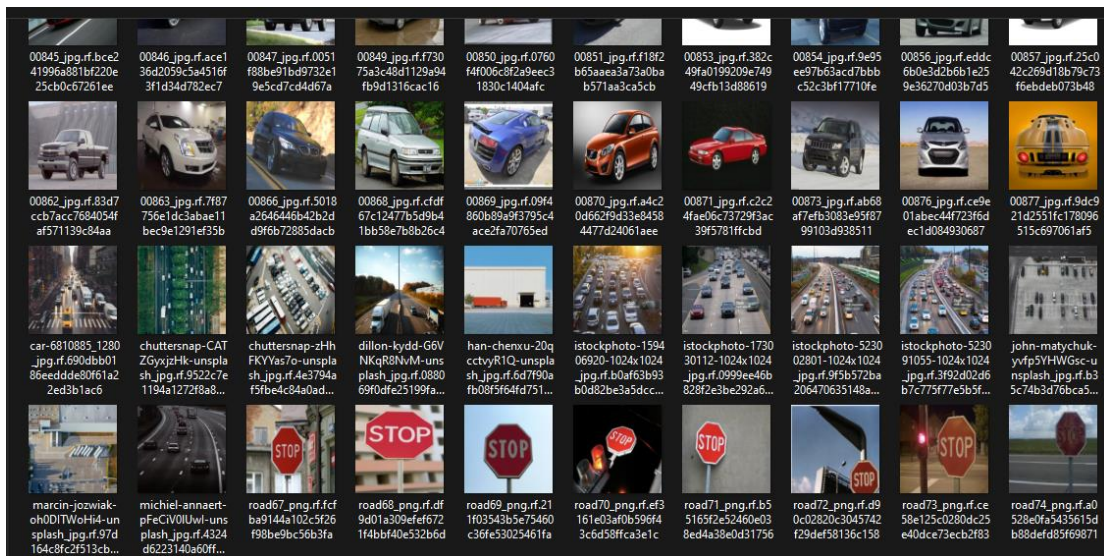


Figure 28 Παράδειγμα Dataset

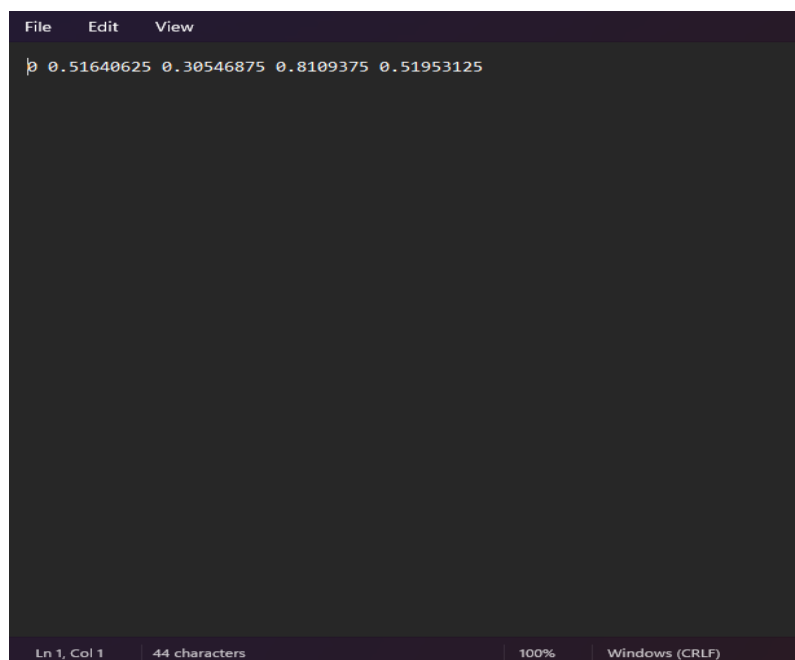


Figure 29 Παράδειγμα ενός TXT Αρχείου με τις Συντεταγμένες

4.2 Εκπαίδευση YOLOv5

4.2 α) Εκπαίδευση YOLOv5 με τις Κατάλληλες Παραμέτρους

Το αρχείο coco128.yaml περιέχει πληροφορίες σχετικά με τη διαμόρφωση της διαδικασίας εκπαίδευσης, τις ρυθμίσεις του μοντέλου, τις κλάσεις αντικειμένων, τα μονοπάτια για τα αρχεία δεδομένων και άλλες παραμέτρους που αφορούν την εκπαίδευση του YOLOv5. Τροποποιούμε το αρχείο coco128.yaml, αλλάζοντας τις παραμέτρους και τα Paths ώστε να αντιστοιχούν στα δικά μας δεδομένα εκπαίδευσης. Αυτό συμπεριλαμβάνει την αλλαγή των Paths για τις φωτογραφίες, τα αντίστοιχα αρχεία ετικετών, και άλλες παραμέτρους σύμφωνα με τις ανάγκες του έργου μας. Με αυτόν τον τρόπο προετοιμάζουμε το μοντέλο YOLOv5 για εκπαίδευση στα δικά μας δεδομένα εκπαίδευσης.

```
10 # Train/val/test sets as 1) dir: path/to/imgs, 2) file: path/to/imgs.txt, or 3) list: [path/to/imgs1, path/to/imgs2, ..]
11 path: C:\Users\johnh\OneDrive\Desktop\yolo\train_data # dataset root dir
12 train: C:\Users\johnh\OneDrive\Desktop\yolo\train_data\images\train # train images (relative to 'path') 128 images
13 val: C:\Users\johnh\OneDrive\Desktop\yolo\train_data\images\val # val images (relative to 'path') 128 images
14 test: C:\Users\johnh\OneDrive\Desktop\yolo\train_data\images\test
15
16 nc: 2
17 names: ['car', 'stop']
```

Figure 30 Coco128.yaml

Με τις φωτογραφίες μας έτοιμες, προχωράμε στην εκπαίδευση του YOLOv5 για τον εντοπισμό αντικειμένων. Χρησιμοποιώντας την εντολή **python train.py --epochs 100 --data C:\Users\johnh\OneDrive\Desktop\yolo\yolov5\data\coco128.yaml --weights yolov5s.pt** τρέχουμε τον κώδικα train.py του YOLOv5 με τις κατάλληλες παραμέτρους για εμάς. Εκτελέσαμε τη διαδικασία εκπαίδευσης με 100 επαναλήψεις λόγω του όγκου των φωτογραφιών που διαθέτουμε. Τα αποτελέσματα ήταν ικανοποιητικά, καθώς η εκπαίδευση έγινε με επιτυχία και το μοντέλο παρήγαγε αξιόπιστες προβλέψεις. Σε περίπτωση που είχαμε πάνω από 1500 εικόνες ανά κλάση, ο δημιουργός του YOLOv5 πρότεινε τη χρήση περισσότερων επαναλήψεων για την εκπαίδευση, προκειμένου να επιτευχθούν βέλτιστα αποτελέσματα.

```
PS C:\Users\johnd\OneDrive\Desktop\yolo\yolo> python train.py --epochs 100 --data C:\Users\johnd\OneDrive\Desktop\yolo\yolo\data\coco128.yaml --weights yolo6n.pt
train: weights=yolo6n.pt, cfg=C:\Users\johnd\OneDrive\Desktop\yolo\yolo\data\coco128.yaml, hyp-data\hyp\yolo\scratch-low.yaml, epochs=100, batch_size=16, imgsz=640, rect=False, resume=False, nosave=False, noval=False, noautoanchor=False, replots=False,
evolve=None, bucket=None, cache=None, image_weights=False, device=None, multi_scale=False, single_cls=False, optimizer=SGD, sync_bn=False, workers=8, project=runs/train, name=exp, exist_ok=False, quad=False, cos_lr=False, label_smoothing=0.0, patience=100, freeze=[0]
remote: Enumerating objects: 170, done.
remote: Counting objects: 100% (130/130), done.
remote: Compressing objects: 100% (32/32), done.
remote: Total 170 (delta 110), reused 122 (delta 107), pack-reused 31
Receiving objects: 100% (170/170), 132.63 KiB | 1021.00 KiB/s, done.
Resolving deltas: 100% (123/123), completed with 54 local objects.
From https://github.com/ultralytics/yolo6
  bcd4f8de..22361691 master -> origin/master
  * [new branch]      dependabot/github/workflows/slackapi/slack-github-action-1.26.0 -> origin/dependabot/github_actions/dot-github/workflows/slackapi/slack-github-action-1.26.0
  * [new branch]      syzk-fix-18f3672f1c40501ef6d20c1b34d6f9 -> origin/syzk-fix-18f3672f1c40501ef6d20c1b34d6f9
github: YOLOv5 is out of date by 57 commits. Use 'git pull' or 'git clone https://github.com/ultralytics/yolo6' to update.
YOLOv5 v7.0-247-g3f802dee Python-3.11.9 torch-2.1.1cpu CPU

Hyperparameters: lr=0.01, lr_f=0.01, momentum=0.937, weight_decay=0.0005, warmup_epochs=3.0, warmup_momentum=0.8, warmup_bias_lr=0.1, box=0.05, cls=0.5, cls_pw=1.0, obj=1.0, obj_pw=1.0, iou_t=0.2, anchor_t=4.0, fl_gamma=0.0, hsv_h=0.015, hsv_s=0.7, hsv_v=0.4, d
egrees=0.0, translate=0.1, scale=0.5, shear=0.0, perspective=0.0, flipud=0.0, flipud=0.0, mosaic=1.0, mixup=0.0, copy_paste=0.0
Commit: run 'pip install commit' to automatically track and visualize YOLOv5 runs in Comm
TensorBoard: Start with 'tensorboard --logdir runs/train', view at http://localhost:6006/
Overriding model.yaml nc=80 with nc=3

      from n  params  module                                arguments
      0      -1      2020  models.common.Conv                      [1, 32, 6, 2, 2]
      1      -1     18560  models.common.Conv                      [32, 64, 3, 2, 2]
      2      -1     18816  models.common.C3                        [64, 64, 1, 1]
      3      -1     78880  models.common.Conv                      [64, 128, 3, 2, 2]
      4      -1     115712  models.common.C3                        [128, 128, 1, 1]
      5      -1     295424  models.common.Conv                      [128, 256, 3, 2, 2]
      6      -1     625152  models.common.C3                        [256, 256, 1, 1]
      7      -1     1189072  models.common.Conv                      [256, 512, 3, 2, 2]
      8      -1     1162720  models.common.C3                        [512, 512, 1, 1]
      9      -1     656608  models.common.SPPF                      [512, 512, 1]
     10      -1     131584  models.common.Conv                      [512, 256, 1, 1]
     11      -1      0  torch.nn.modules.upsampling.Upsample    [None, 2, 'nearest']
     12      [-1, 6]      0  models.common.Concat                    [1]
     13      -1     361984  models.common.C3                        [512, 256, 1, False]
     14      -1     33824  models.common.Conv                      [256, 128, 1, 1]
     15      -1      0  torch.nn.modules.upsampling.Upsample    [None, 2, 'nearest']
     16      [-1, 4]      0  models.common.Concat                    [1]
     17      -1     90880  models.common.C3                        [256, 128, 1, False]
     18      -1     147712  models.common.Conv                      [128, 128, 3, 2, 2]
     19      [-1, 14]      0  models.common.Concat                    [1]
     20      -1     296448  models.common.C3                        [256, 256, 1, False]
     21      -1     598336  models.common.Conv                      [256, 256, 3, 2, 2]
     22      [-1, 10]      0  models.common.Concat                    [1]
     23      -1     1162720  models.common.C3                        [512, 512, 1, False]
     24      [12, 30, 22]      1  models.yolo.detect                      [3, [[10, 13, 16, 30, 33, 23], [30, 61, 62, 45, 50, 110], [116, 90, 156, 190, 373, 326]], [128, 256, 512]]
Model summary: 214 layers, 7627270 parameters, 7627270 gradients, 16.8 GFLOPs

Transferred 343/349 items from yolo6n.pt
optimizer: SGD(lr=0.01) with parameter groups 57 weight(decay=0.0), 60 weight(decay=0.0005), 60 bias
augmentations: blur(p=0.01, blur_limit=(3, 7)), medianblur(p=0.01, blur_limit=(3, 7)), fourray(p=0.01), clamp(p=0.01, clip_limit=(1, 4.0), tile_grid_size=(8, 8))
Export: Scanning C:\Users\johnd\OneDrive\Desktop\yolo\yolo\data\coco128\train data\labels\train.cache... 434 images, 0 background, 0 corrupt images... 1.43M/434 [100.00%] 211s]
```

Figure 31 Start of training process

Κατά την εκπαίδευση του μοντέλου YOLOv5 δημιουργείται ένας φάκελος που περιέχει σημαντικά αρχεία για την πορεία του έργου. Τα αρχεία best.pt και last.pt αποτελούν τα σημαντικότερα, καθώς αποθηκεύουν τα weights του μοντέλου από την καλύτερη και την τελευταία εποχή της εκπαίδευσης αντίστοιχα. Το αρχείο best.pt είναι ιδιαίτερα σημαντικό για τη συνέχιση του έργου, καθώς αντιπροσωπεύει το μοντέλο με την καλύτερη απόδοση στο σύνολο δεδομένων επικύρωσης και αυτό που θα χρησιμοποιήσουμε για την συνέχεια στην εργασία μας. Επίσης τα διαγράμματα που παράγονται κατά τη διάρκεια της εκπαίδευσης παρέχουν επίσης σημαντικές πληροφορίες σχετικά με την πορεία της απόδοσης του μοντέλου.

4.2 β) Επεξεργασία Βίντεο

- Αφού εντοπίσουμε ένα κατάλληλο βίντεο που να απεικονίζει αυτοκίνητα να κινούνται σε έναν ευθύ δρόμο. Για να προχωρήσουμε στην τελική επεξεργασία πρέπει πρώτα να γνωρίζουμε τα στοιχεία από το αυθεντικό βίντεο. Αυτό επιτυγχάνεται τρέχοντας κώδικα σε Python και χρησιμοποιώντας συγκεκριμένα τις εντολές **cap=cv2.VideoCapture(video_path)** για να εισαγάγουμε το βίντεο και την κατάλληλη διαδικασία. Στη συνέχεια, χρησιμοποιούμε **cap.get(cv2.CAP_PROP_FPS)** για να πάρουμε τα FPS και **int(cap.get(cv2.CAP_PROP_FRAME_COUNT))** για να πάρουμε το συνολικό αριθμό των frames. Όλα αυτά ολοκληρώνονται προσθέτοντας τις κατάλληλες εντολές print για να τα εμφανίσουμε.

```
... FPS: 24.0
Total Frames: 307
...
```

Figure 32 Παράδειγμα fps και frames του αρχικού βίντεο


```

1  import cv2
2
3  def play_video(video_path):
4      cap = cv2.VideoCapture(video_path)
5
6
7      fps = cap.get(cv2.CAP_PROP_FPS)
8      total_frames = int(cap.get(cv2.CAP_PROP_FRAME_COUNT))
9
10     print(f"FPS: {fps}")
11     print(f"Total Frames: {total_frames}")
12
13     while cap.isOpened():
14         ret, frame = cap.read()
15
16         if not ret:
17             print("Error reading the video frame")
18             break
19
20         cv2.imshow("Video", frame)
21
22         if cv2.waitKey(int(1000/fps)) & 0xFF == ord('q'):
23             break
24
25     cap.release()
26     cv2.destroyAllWindows()
27
28
29     video_path = 'traffic.mp4'
30     play_video(video_path)

```

Figure 33 Κώδικας για τα FPS

- ii. Για την επεξεργασία του βίντεο, πρέπει να προσαρμόσουμε την αρχική πληροφορία του βίντεο μας για την εργασία μας. Αρχικά, χρησιμοποιούμε την εντολή **cv2.VideoWriter_fourcc(*'XVID')** για να επιλέξουμε τον κατάλληλο κωδικοποιητή, ο οποίος είναι σημαντικός για την ποιότητα και τη συμπίεση του βίντεο. Στη συνέχεια, χρησιμοποιούμε την εντολή **cv2.VideoWriter('new_video.mp4',fourcc, 5, (1280,720))** για να δημιουργήσουμε ένα νέο βίντεο με το όνομα "new_video.mp4", με **5 fps** και ανάλυση **1280x720**. Με αυτόν τον τρόπο δημιουργείται ένα νέο βίντεο το οποίο μπορούμε να αναπαράγουμε από τη βιβλιοθήκη cv2.

```

1
2
3 import cv2
4 import numpy as np
5
6 cap = cv2.VideoCapture('stop2.mp4')
7
8 fourcc = cv2.VideoWriter_fourcc(*'XVID')
9 out = cv2.VideoWriter('stop2.2.mp4',fourcc, 5, (1280,720))
10
11 while True:
12     ret, frame = cap.read()
13     if ret == True:
14         b = cv2.resize(frame,(1280,720),fx=0,fy=0, interpolation = cv2.INTER_CUBIC)
15         out.write(b)
16     else:
17         break
18
19 cap.release()
20 out.release()
21 cv2.destroyAllWindows()
22

```

Figure 34 Κώδικας Επεξεργασίας Βίντεο

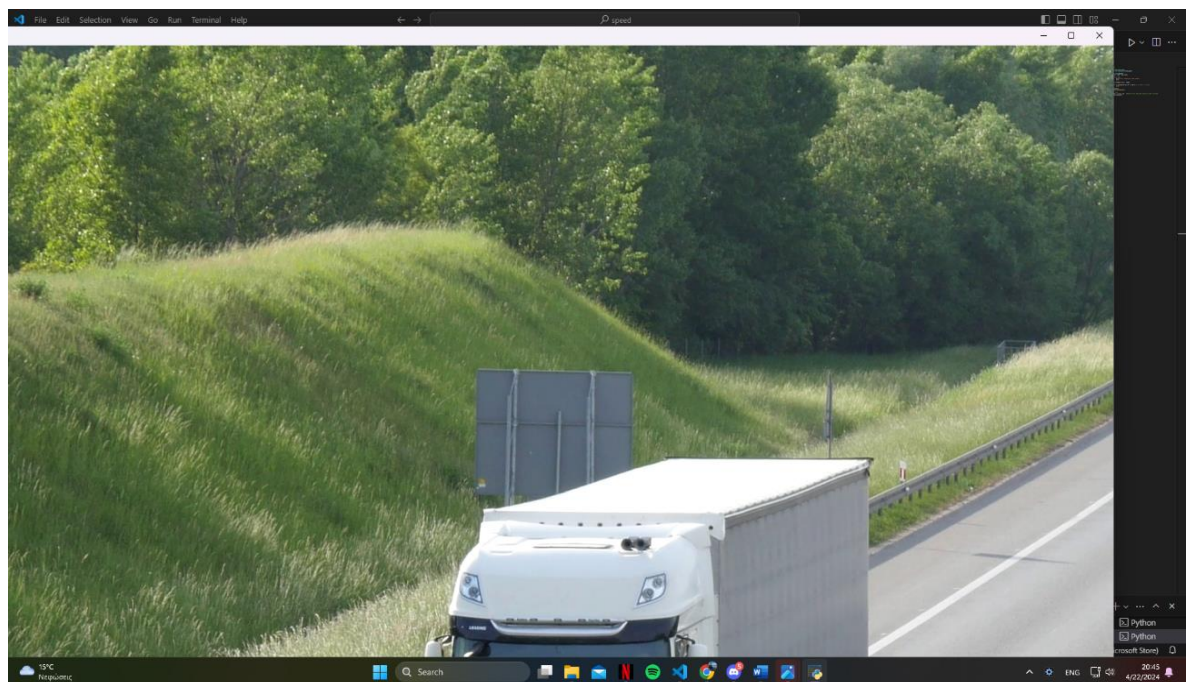


Figure 35 Βίντεο πριν την Επεξεργασία

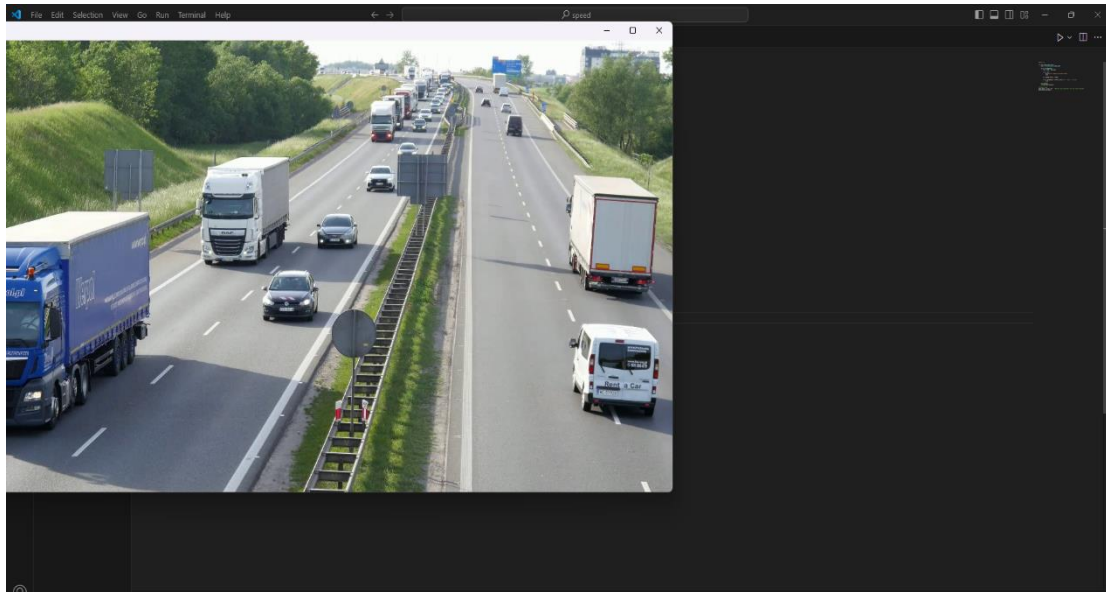


Figure 36 Βίντεο Μετά την Επεξεργασία

4.3 Χρήση αλγόριθμου SORT

4.3 α) Αλγόριθμος SORT[28][29]

Ο αλγόριθμος sort αναφέρεται σε μια οικογένεια αλγορίθμων που χρησιμοποιούνται για την οργάνωση των στοιχείων ενός συνόλου σε μια συγκεκριμένη σειρά. Αυτή η σειρά μπορεί να είναι αύξουσα ή φθίνουσα, ανάλογα με τον στόχο της εφαρμογής. Οι αλγόριθμοι sort είναι βασικοί στην επιστήμη των υπολογιστών και χρησιμοποιούνται ευρέως σε πολλές εφαρμογές. Μερικοί από τους πιο γνωστούς αλγορίθμους sort περιλαμβάνουν τον αλγόριθμο Bubble Sort, τον αλγόριθμο Quick Sort, τον αλγόριθμο Merge Sort, τον αλγόριθμο Insertion Sort και το Tim Sort ο οποίος είναι ο συνδυασμός των δύο προηγούμενων. Κάθε ένας από αυτούς τους αλγορίθμους έχει τις δικές του ιδιαιτερότητες και απαιτήσεις από την άποψη της απόδοσης, επιλέγοντας ανάλογα με την εφαρμογή τους. Εμείς στην εργασία μας χρησιμοποιούμε τον Tim. Συνολικά, ο αλγόριθμος sort αποτελεί ένα βασικό εργαλείο στον χώρο της πληροφορικής για την αποτελεσματική οργάνωση και διαχείριση δεδομένων.

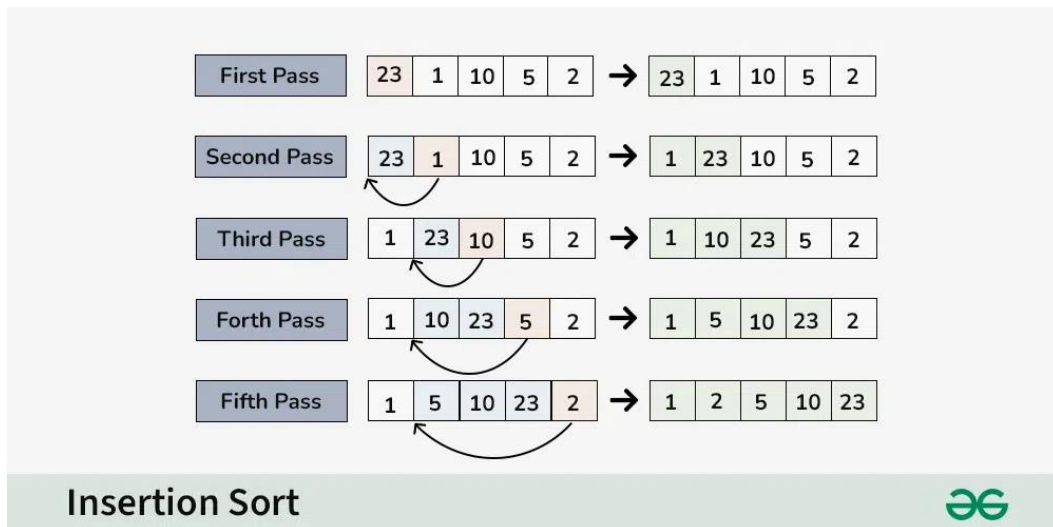


Figure 37 Inserion Sort



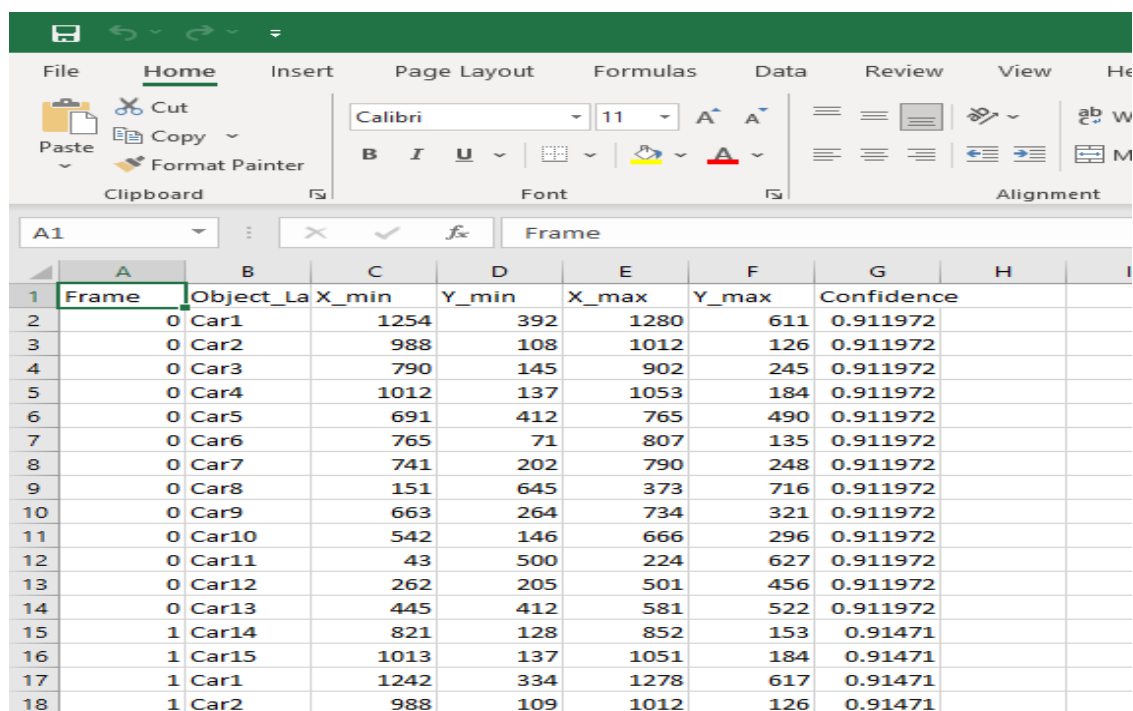
Figure 38 Timm Sort

4.3 β) Εντοπισμός αυτοκινήτων σε βίντεο

Μετά από την επιτυχή εκπαίδευση του YOLOv5 και την απόκτηση των απαραίτητων δεδομένων, μπορούμε να χρησιμοποιήσουμε τον αλγόριθμο SORT για να εντοπίσουμε αυτοκίνητα και πινακίδες στα βίντεο. Με τη δημιουργία ενός κώδικα που επεξεργάζεται τα καλύτερα αποτελέσματα από την εκπαίδευση του YOLOv5, το αρχείο με την καλύτερη εκπαίδευση (best.pt), και λαμβάνει ως είσοδο το βίντεο και δημιουργεί ένα αρχείο CSV με τα δεδομένα των αυτοκινήτων. Τα δεδομένα περνάνε στο αρχείο ανα frame και κάθε αυτοκίνητο είναι αναγνωρισμένο με ένα μοναδικό όνομα όπως "car1", "car2", κ.λπ. Τα στοιχεία του περιλαμβάνουν τις συντεταγμένες από το πλαίσιο που το περικλείει τα αυτοκίνητα (bounding boxes), την κλάση που εντόπισε και το confidence του εντοπισμού.

Κατά τη διάρκεια της ανίχνευσης, έχουμε τη δυνατότητα να παρακολουθούμε την κίνηση των αυτοκινήτων στο βίντεο, καθώς αυτό αναπαράγεται κανονικά. Αυτή η διαδικασία μας παρέχει τη δυνατότητα να βλέπουμε ταυτόχρονα αν οι εντοπισμοί μας είναι σωστοί. Αυτό καθιστά τη διαδικασία εντοπισμού πιο πλήρη και περιληπτική για την ανάλυση των δεδομένων.

Γενικά, η χρήση του YOLOv5 σε συνδυασμό με τον αλγόριθμο SORT επιτρέπει την αποτελεσματική ανίχνευση των αυτοκινήτων και των πινακίδων στο βίντεο, ενώ παράλληλα παρέχει πλούσια πληροφορία για την κίνηση και το περιβάλλον τους. Αυτό το σύστημα μπορεί να είναι χρήσιμο για πολλούς τομείς, συμπεριλαμβανομένης της ασφάλειας και της κυκλοφορίας.



	A	B	C	D	E	F	G	H	I
	Frame	Object_Label	X_min	Y_min	X_max	Y_max	Confidence		
2	0	Car1	1254	392	1280	611	0.911972		
3	0	Car2	988	108	1012	126	0.911972		
4	0	Car3	790	145	902	245	0.911972		
5	0	Car4	1012	137	1053	184	0.911972		
6	0	Car5	691	412	765	490	0.911972		
7	0	Car6	765	71	807	135	0.911972		
8	0	Car7	741	202	790	248	0.911972		
9	0	Car8	151	645	373	716	0.911972		
10	0	Car9	663	264	734	321	0.911972		
11	0	Car10	542	146	666	296	0.911972		
12	0	Car11	43	500	224	627	0.911972		
13	0	Car12	262	205	501	456	0.911972		
14	0	Car13	445	412	581	522	0.911972		
15	1	Car14	821	128	852	153	0.91471		
16	1	Car15	1013	137	1051	184	0.91471		
17	1	Car1	1242	334	1278	617	0.91471		
18	1	Car2	988	109	1012	126	0.91471		

Figure 39 Παράδειγμα του Αρχείου CSV Μετά τον εντοπισμό

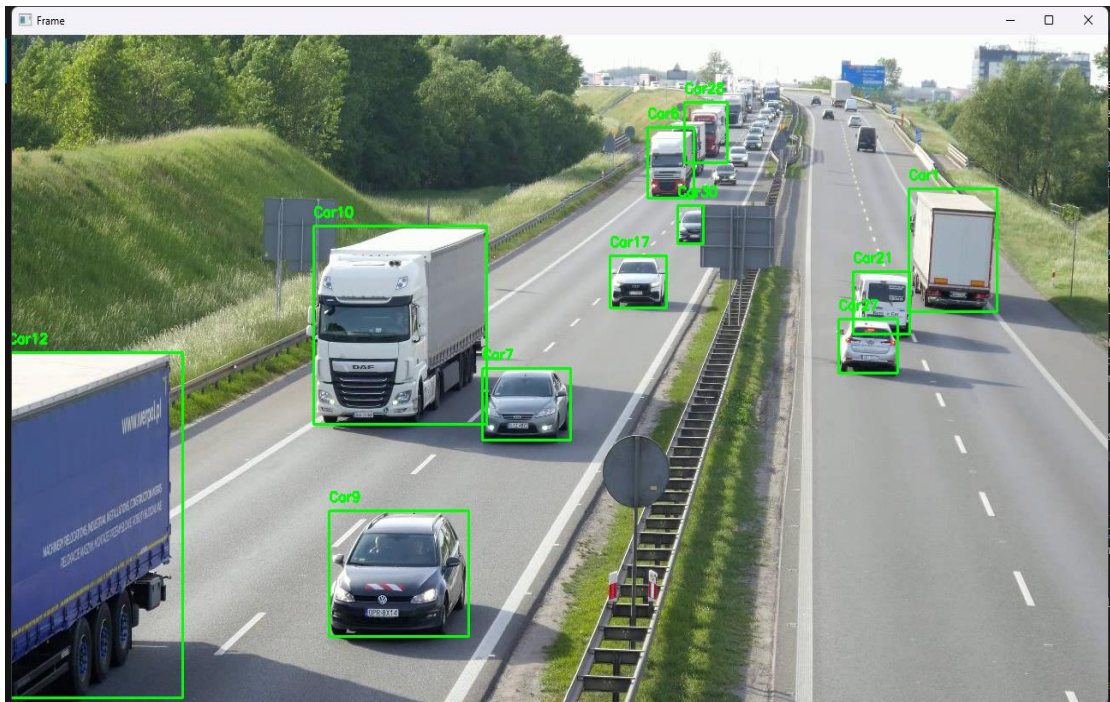


Figure 40 Παράδειγμα 1

ΚΕΦΑΛΑΙΟ 5 Εκτίμηση Ταχύτητας

5.1 Εισαγωγή και Ανάλυση Δεδομένων

Η διαδικασία ανάλυσης της ταχύτητας των αυτοκινήτων αρχίζει με την εισαγωγή και επεξεργασία των δεδομένων από το αρχείο CSV που έχουμε δημιουργήσει από τον εντοπισμό προηγούμενος, το οποίο περιέχει τις συντεταγμένες των Bounding Boxes για κάθε όχημα που εντοπίζεται στο βίντεο όπως είπαμε και πάνω. Αυτές οι συντεταγμένες είναι κρίσιμες για την παρακολούθηση της θέσης και της κίνησης κάθε αυτοκινήτου με την πάροδο του χρόνου. Δημιουργούμε έναν αλγόριθμο του οποίου ο κώδικας αναλύει την κίνηση των Bounding Boxes σε κάθε frame του βίντεο, παρέχοντας ένα δυναμικό στιγμιότυπο των μετακινήσεων των οχημάτων. Αυτή η προσέγγιση επιτρέπει τη λεπτομερή καταγραφή της απόστασης που διανύει κάθε όχημα.

5.2 Υπολογισμός Ταχύτητας και Παραβίαση Stop

5.2 α) Υπολογισμός Ταχύτητας

Μετά την ανάλυση των κινήσεων των οχημάτων, προχωρούμε στον υπολογισμό της ταχύτητας. Γνωρίζοντας ότι τα frames per second (FPS) του βίντεο είναι 5 και το scale factor, ο οποίος προσδιορίζει τα μέτρα ανά pixel και δεν είναι γνωστός με ακρίβεια (το ορίζουμε 4,5), υπολογίζουμε τη μέση ταχύτητα κάθε αυτοκινήτου σε μέτρα ανά δευτερόλεπτο. Αυτό το δεδομένο μετατρέπεται στη συνέχεια σε χιλιόμετρα ανά ώρα (kkm/h), πολλαπλασιάζοντας το με 3.6, για να προσφέρει μια πιο κατανοητή μονάδα μέτρησης. Αυτή η μετατροπή είναι ουσιαστική για την αξιολόγηση της ταχύτητας των αυτοκινήτων σε πραγματικές συνθήκες οδήγησης, δίνοντας μας μια πλήρη εικόνα του πόσο γρήγορα κινούνται τα οχήματα στο δρόμο.

Χρησιμοποιώντας το αρχείο CSV με τα κατάλληλα δεδομένα και με τη βοήθεια ενός αλγορίθμου που δημιουργούμε για τους απαραίτητους υπολογισμούς, έχουμε τη δυνατότητα να προσδιορίσουμε τη μέση ταχύτητα για κάθε όχημα ξεχωριστά. Ας πάρουμε ως παράδειγμα στο αυτοκίνητο με την ονομασία "car9". Ο αλγόριθμος υπολογίζει τη μέση ταχύτητα αυτού του οχήματος με βάση τα δεδομένα που έχουμε και τα εκτυπώνει στην οθονή μας. Μπορούμε να κάνουμε το ίδιο με οποιοδήποτε αμάξι θέλουμε. Ακολουθούν τα παραδείγματα:

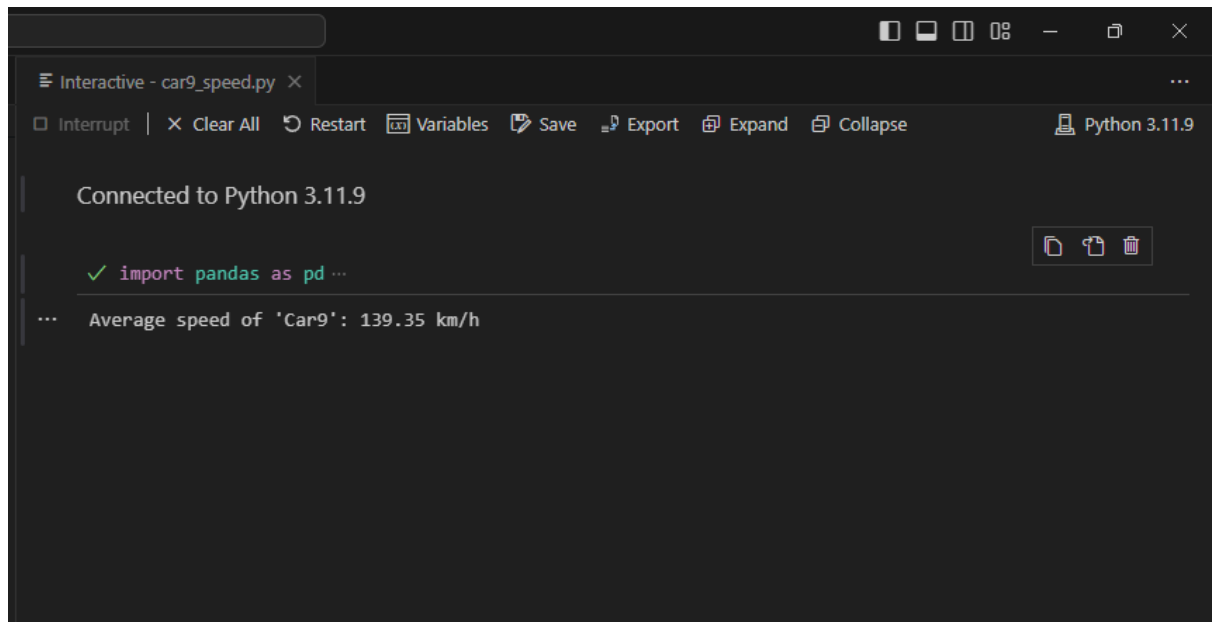


Figure 41 Μέση Ταχύτητα του CAR9 στο Παράδειγμα 1

```
7 Average speed of Car7: 128.12 km/h
8 Average speed of Car8: 245.45 km/h
9 Average speed of Car9: 139.35 km/h
10 Average speed of Car10: 80.74 km/h
11 Average speed of Car11: 166.18 km/h
12 Average speed of Car12: 104.22 km/h
```

Figure 42 Μέση Ταχύτητα απο κάποια αυτοκίνητα στο Παράδειγμα 1

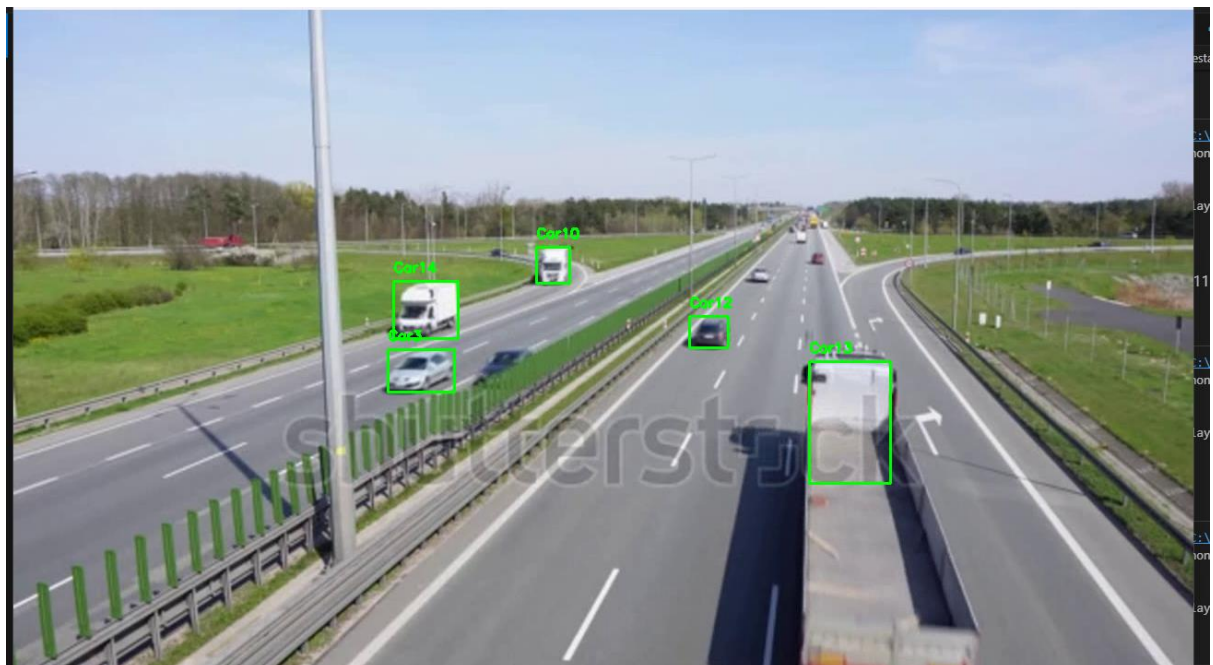


Figure 43 Παράδειγμα 2
detections/example 2

```
✓ import pandas as pd ...  
... Average speed of 'Car3': 128.31 km/h
```

Figure 44 Μέση Ταχύτητα του CAR3 στο Παράδειγμα 2

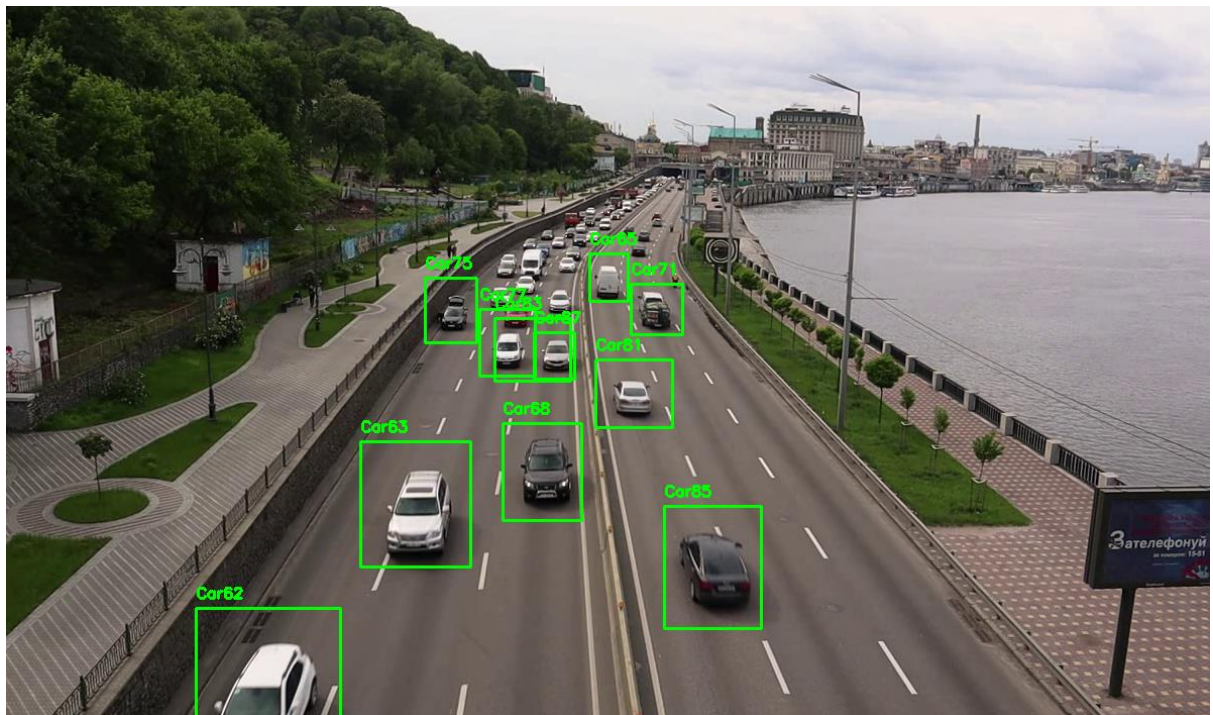


Figure 45 Παράδειγμα 3

```
Interrupt | X Clear All | Restart | Variables | Save | Export | Expand | Collapse | Python 3.11.9  
Connected to Python 3.11.9  
✓ import cv2 ...  
... Average speed of 'Car68': 78.69 km/h  
car68 not stop in any sign
```

Figure 46 Μέση Ταχύτητα του CAR68 στο Παράδειγμα 3

5.2 β) ROC Curve [17]

Η καμπύλη ROC (Receiver Operating Characteristics Curve) είναι μια γραφική παράσταση που απεικονίζει τον πραγματικό θετικό ρυθμό (ευαισθησία) έναντι του ψευδώς θετικού ποσοστού (1-ειδικότητα) για διαφορετικά κατώφλια ταξινόμησης. Το πραγματικό θετικό ποσοστό (True Positive Rate TPR) είναι η αναλογία των πραγματικών θετικών περιπτώσεων που το μοντέλο προσδιορίζει σωστά. Αντίθετα, το ψευδώς θετικό ποσοστό (False Positive Rate, FPR) είναι το ποσοστό των πραγματικών αρνητικών περιπτώσεων που εσφαλμένα προσδιορίστηκαν ως θετικές.

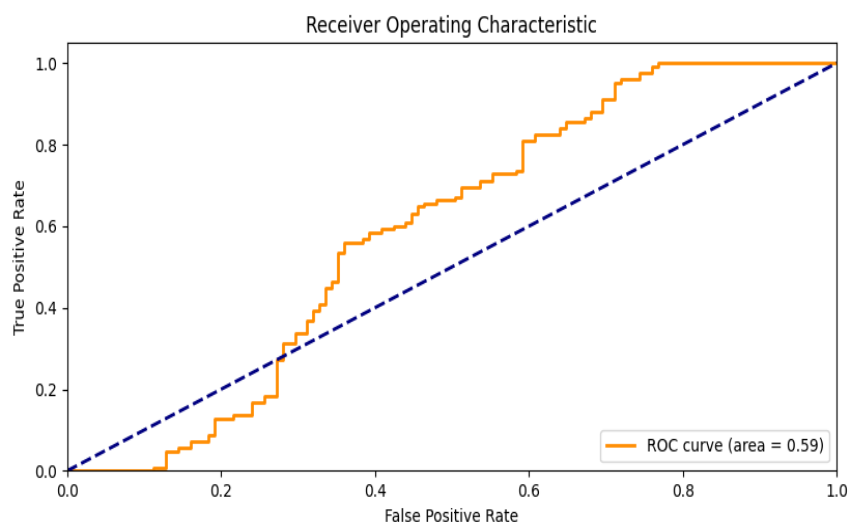


Figure 47 ROC curve από τον Εντοπισμό Αυτοκινήτων στο Παράδειγμα 1

5.2 γ) Παραβίαση Stop

Για να αξιολογήσουμε αν ένα αυτοκίνητο έχει σταματήσει σε πινακίδα στοπ, αναλύουμε το βίντεο σε διαδοχικά frames. Κρίσιμος παράγοντας είναι η ακινησία του αυτοκινήτου σε συνεχόμενα frames, κάτι που δηλώνει ότι το αυτοκίνητο πράγματι σταμάτησε. Στην περίπτωση μας υποθέτουμε ότι υπάρχει πινακίδα στοπ σε ένα σημείο του βίντεο, εφόσον δεν διαθέτουμε βίντεο με πραγματική πινακίδα στοπ και τα οχήματα να καταγράφονται από κάμερα που θα μπορούμε να αναλύσουμε τα ζητούμενά μας. Αυτή η προσέγγιση μας επιτρέπει να εκτιμήσουμε αν το αυτοκίνητο συμμορφώνεται με τη σήμανση στοπ, παρόλο που δεν υπάρχει άμεση οπτική απεικόνιση της πινακίδας στο δρόμο. Σε περίπτωση που υπήρχε σήμανση μέσω της θέσης του bounding box θα μπορούσαμε να δούμε και σε ποιο σημείο υπάρχει. Δυστυχώς δεν καταφέραμε να βρούμε κάποιο βίντεο που να υπάρχει πινακίδα στοπ σε δρόμο ή με αντίστοιχες άλλες να εντοπίσουμε την δικιά μας.

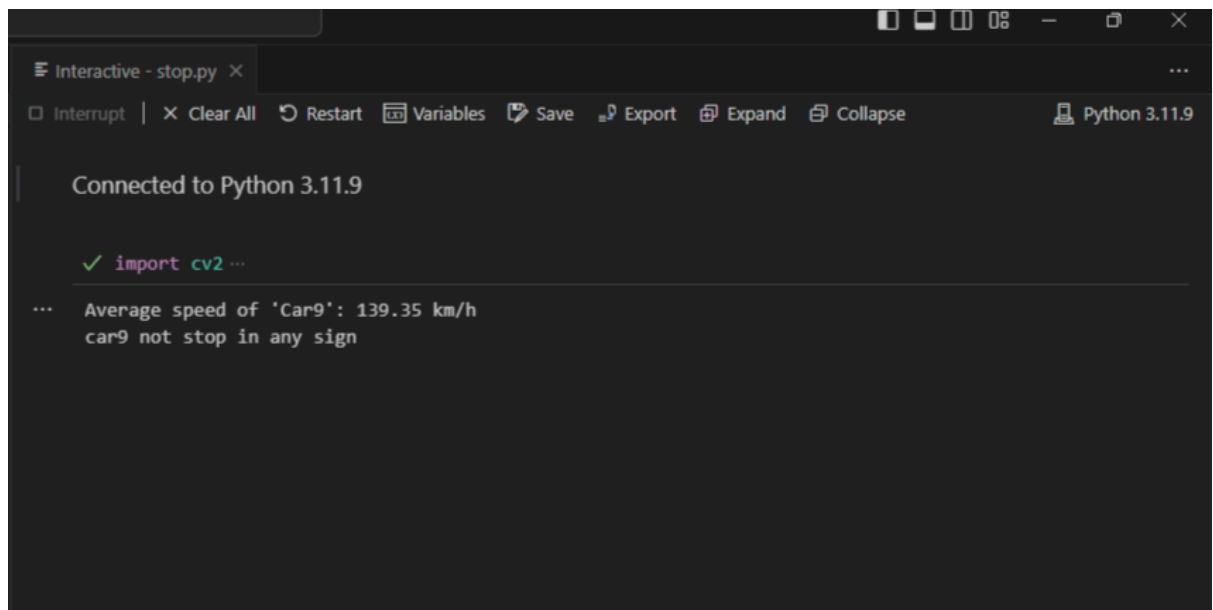


Figure 48 Έλεγχος αν το CAR9 σταμάτησε καθόλου στο παράδειγμα 1



Figure 49 Εντοπισμός Πινακίδας Stop v1

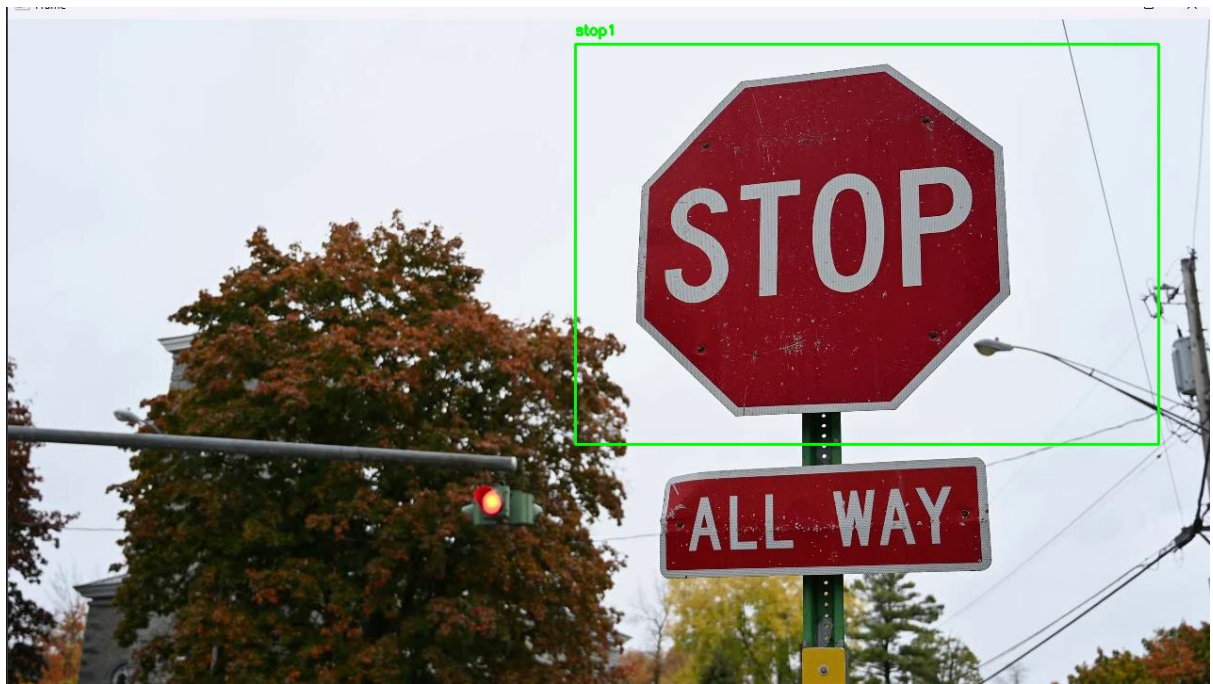


Figure 50 Εντοπισμός Πινακίδας Stop v2

```
✓ import pandas as pd ...  
... Stop sign detected but no cars in video.
```

Figure 51 Μήνυμα ότι υπάρχει μόνο STOP και όχι αυτοκίνητα στο Figure 49 και 50



Figure 52 Εντοπισμός Πινακίδας Stop v3 στην πόλη της Λαμίας

ΚΕΦΑΛΑΙΟ 6 Συμπεράσματα

6.1 Συμπεράσματα

Χρησιμοποιώντας τον αλγόριθμο YOLO (You Only Look Once), αναπτύξαμε ένα καινοτόμο σύστημα ανίχνευσης αυτοκινήτων και εκτίμησης της ταχύτητάς τους. Με τη χρήση αυτού, το σύστημά μας καταγράφει την θέση των αυτοκινήτων στο οπτικό πεδίο και υπολογίζει την ταχύτητά τους με βάση την αλλαγή θέσης μεταξύ διαδοχικών frames. Η εκτίμηση της ταχύτητας επιτρέπει την ανίχνευση της συμμόρφωσης των οδηγών με τα όρια ταχύτητας σε διάφορες ζώνες. Επιπρόσθετα, το σύστημα διαθέτει την ικανότητα να αναγνωρίζει πινακίδες σταματήματος. Κατά την ανίχνευση μιας πινακίδας STOP, το σύστημα παρακολουθεί την ταχύτητα των αυτοκινήτων και ειδοποιεί αν ένα αυτοκίνητο δεν σταματήσει πλήρως, βοηθώντας έτσι στην εφαρμογή των κανόνων στάθμευσης και σήμανσης στις διασταυρώσεις.

Αυτή η λειτουργικότητα αντικατοπτρίζει την εξαιρετική ικανότητα του συστήματος να συνεισφέρει στην ασφαλή οδήγηση, καθώς τα αυτοκίνητα που δεν συμμορφώνονται με τα σήματα σταματήματος μπορούν να αποτελέσουν μεγάλο κίνδυνο για ατυχήματα. Η ικανότητα του συστήματος να λειτουργεί αξιόπιστα και αποτελεσματικά σε πραγματικό χρόνο παρέχει μια σημαντική ενίσχυση στην τεχνολογία ασφάλειας οδικής κυκλοφορίας.

6.2 Μελλοντικές Βελτιώσεις

Στις μελλοντικές βελτιώσεις, η εργασία αναμένεται να επεκταθεί στην ικανότητα αναγνώρισης διαφόρων τύπων οχημάτων, καλύπτοντας λεωφορεία, μηχανές, και άλλα οχήματα. Αυτό θα επιτρέπει μια πιο ολοκληρωμένη και ακριβή επιτήρηση της κυκλοφορίας σε διάφορα αστικά και υπαίθρια περιβάλλοντα. Επιπλέον, η δυνατότητα της εργασίας να αναγνωρίζει διάφορες πινακίδες οδικής κυκλοφορίας και όρια ταχυτήτων θα ενισχυθεί. Αυτό θα επιτρέπει στο σύστημα να ειδοποιεί όταν ένα όχημα υπερβαίνει τα καθορισμένα όρια ταχύτητας, βελτιώνοντας την ασφάλεια και την τήρηση των κανονισμών. Η ακρίβεια της εργασίας θα βελτιωθεί μέσω της παροχής πιο ακριβών μετρήσεων και πληροφοριών για τα δεδομένα του βίντεο, ώστε τα αποτελέσματα να είναι πιο έγκυρα και αξιόπιστα. Με τη βελτίωση του dataset και τη σωστή εκπαίδευση του μοντέλου, όχι μόνο από περισσότερους τύπους οχημάτων αλλά και από περισσότερες οπτικές γωνίες και διαφορετικές φωτογραφίες αυτοκινήτων, αναμένουμε το τέλειο αποτέλεσμα. Αυτές οι εξελίξεις αναμένεται να αυξήσουν την εφαρμοσιμότητα της εργασίας σε πληθώρα σεναρίων αστικής κυκλοφορίας και οδικής ασφάλειας.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. <https://www.kaggle.com/datasets/andrewmvd/road-sign-detection/data>
2. <https://roboflow.com/>
3. <https://bigblue.academy/gr/neuronika-diktua>
4. Guo, Z., Wang, C., Yang, G., Huang, & Li, G. (2022). MSFT-YOLO: Improved YOLOv5 Based on Transformer for Detecting Defects of Steel Surface. Sensors, 22(9).
5. Du, L., Zhang, R., & Wang, X. (2020). Overview of two-stage object detection algorithms. Journal of Physics: Conference Series, 1544(1), 012033.
6. <https://www.paloaltonetworks.com/cyberpedia/mpls-what-is-multiprotocol-label-switching>
7. An Introduction to Neural Networks by Kevin Gurney
8. <https://www.vecteezy.com/video/7957364-car-and-truck-traffic-on-the-highway-in-europe-poland-summer-day>
9. <https://www.dpa.gr/el/enimerwtiko/deltia/emfanisi-synoloy-dedomenon-dataset-sto-diadiktyo>
10. <https://www.vecteezy.com/video/18733751-cars-drive-along-the-embankment-of-the-highway-in-kyiv-ukraine-traffic-in-both-directions-on-a-busy-road-aerial-view-of-cityscape-ukraine-kyiv-june-1-2020>
11. <https://www.pexels.com/video/video-of-a-sign-stop-in-desert-9010986/>
12. <https://www.techtarget.com/whatis/definition/data-set>
13. <https://www.techtarget.com/searchenterpriseai/definition/unsupervised-learning>
14. <https://bigblue.academy/en/supervised-learning>
15. <https://bigblue.academy/gr/ti-einai-to-machine-learning>
16. <https://www.synopsys.com/ai/what-is-reinforcement-learning.html>
17. <https://el.martech.zone/acronym/roc/>

18. Zhao, Z., Zheng, P., Xu, S., & Wu, X. (n.d.). Object Detection With Deep Learning: A Review. *IEEE Transactions on Neural Networks and Learning Systems*.
19. Wang, C., Yeh, I., & Liao, H. (2024). YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *ArXiv*, abs/2402.13616.
20. Redmon, J., Divvala, S.K., Girshick, R.B., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788.
21. Fan, J., Lee, J., Jung, I., & Lee, Y. (2021b). Improvement of object detection based on faster R-CNN and YOLO. *2021 36th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*.
22. Sun, J., Ge, H., & Zhang, Z. (2021). AS-YOLO: An Improved YOLOv4 based on Attention Mechanism and SqueezeNet for Person Detection. *2021 IEEE 5th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 5, 1451-1456.
23. Grzybowski, A., Pawlikowska-Łagód, K., & Lambert, W. C. (2024). A History of Artificial Intelligence. *Clinics in dermatology*, S0738-081X(23)00268-7. Advance online publication.
24. https://docs.ultralytics.com/yolov5/tutorials/architecture_description/#1-model-structure
25. Redmon, J., Divvala, S.K., Girshick, R.B., & Farhadi, A. (2015). You Only Look Once: Unified, Real-Time Object Detection. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 779-788.
26. [2] Redmon, J., & Farhadi, A. (2016). YOLO9000: Better, Faster, Stronger. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 6517-6525.
27. <https://www.paloaltonetworks.com/cyberpedia/mppls-what-is-multiprotocol-label-switching>
28. <https://github.com/abewley/sort>
29. <https://realpython.com/sorting-algorithms-python/>
30. Real-Time Vehicle Detection Based on Improved YOLO v5 by Yu Zhang 1,Zhongyin Guo 1ORCID,Jianqing Wu 2,3ORCID,Yuan Tian 2,3*,Haotian Tang 2 and Xinming Guo
31. Deep learning-based vehicle speed estimation using the YOLO detector and 1D-CNN Andrej Cvijetic, Slobodan Djukanovic and Andrija Perunicic
32. Traffic Signs Detection for Real-World Application of an Advanced Driving Assisting System Using Deep Learning by Riadh Ayachi, Mouna Aff, Yahia Said και Mohamed Atri

33. Redmon, J., & Farhadi, A. (2018). YOLOv3: An Incremental Improvement. ArXiv, abs/1804.02767.