Οπτική παρακολούθηση και προσδιορισμός τρισδιάστατης θέσης πόρτας υπό συναρμολόγηση σε βιομηχανικό περιβάλλον

Διπλωματική εργασία

Κωνσταντίνος Φραγκούλης

Επιβλέπουσα: Μ. Πατεράκη

Σχολή Αγρονόμων και Τοπογράφων Μηχανικών - Μηχανικών Γεωπληροφορικής

Εθνικό Μετσόβιο Πολυτεχνείο

Μάρτιος 2024

Visual tracking and car door position estimation during assembly tasks in industrial environments

Diploma thesis

Konstantinos Fragkoulis

Supervisor: M. Pateraki

School of Rural, Surveying and Geoinformatics Engineering

National Technical University of Athens

March 2024

Περίληψη

Αντικείμενο της παρούσας εργασίας είναι η παρακολούθηση μιας πόρτας αυτοκινήτου κατά τη συναρμολόγησή της από ανθρώπους, σε βιομηχανικό περιβάλλον. Η παρακολούθηση της πόρτας είναι σημαντική για την ανάλυση του κύκλου εργασιών που πραγματοποιούνται, ενώ μπορεί να βοηθήσει ένα έξυπνο σύστημα ή ρομπότ δίνοντάς του γνώση, ώστε να είναι σε θέση να λάβει αποφάσεις. Στόχος είναι η δημιουργία ενός αλγορίθμου οπτικού εντοπισμού και παρακολούθησης για πόρτες υπο συναρμολόγηση σε γραμμή παραγωγής, αξιοποιώντας δεδομένα εικόνων.

Το πρόβλημα επιλύεται μέσα από δύο διαφορετικές προσεγγίσεις. Στην πρώτη προσέγγιση έγινε ο εντοπισμός δυαδικών στόχων (ArUco markers) σε εικόνες. Στις εικόνες απεικονίζονται πόρτες αυτοκινήτου σε βιομηχανικό περιβάλλον, όπου ενυπάρχουν και οι στόχοι ArUco. Διεξάχθηκε πλήθος δοκιμών οι οποίες αξιολογήθηκαν ποιοτικά και ποσοτικά, προκειμένου να βρεθεί η βέλτιστη μέθοδος εντοπισμού.

Στην δεύτερη προσέγγιση έγινε η εκπαίδευση ενός δικτύου βαθιάς μάθησης ώστε να μπορεί να ανιχνεύει το πλαίσιο και το περίγραμμα πόρτας αυτοκινήτου σε εικόνες. Δημιουργήθηκαν δεδομένα εκπαίδευσης/ελέγχου και η εκπαίδευση βασίστηκε σε ένα ήδη εκπαιδευμένο μοντέλο στο οποίο έγινε επανεκπαίδευση. Υλοποιήθηκε ένα πλήθος δοκιμών οι οποίες αξιολογήθηκαν ποιοτικά και ποσοτικά, ώσπου τελικά προέκυψε το τελικό μοντέλο. Επισημαίνεται ότι το τελικό μοντέλο αποφασίστηκε να ανιχνεύει τις πόρτες αλλά και τους ανθρώπους στη σκηνή, λόγω των αποκρύψεων που δημιουργεί η αλληλεπίδραση των ανθρώπων με τις πόρτες.

Τέλος, παρουσιάζονται κάποια βασικά συμπεράσματα και σχόλια πάνω στη διαδικασία, όπως και προτάσεις για μελλοντική εργασία.

Abstract

The objective of the present thesis is the tracking of a car door during assembly tasks executed by humans, in an industrial environment. Tracking the car door is important for analysing the operations' cycle, while at the same time it can provide context related knowledge to a smart system or robot and enable decision making. The aim is to develop an algorithm for visual detection and tracking of car doors in assembly tasks in a production line, utilizing image sequences.

The problem is solved by two different approaches. In the first approach the detection of binary markers (ArUco markers) in images is exploited. The images depict car doors in industrial environments with detected ArUco markers. A number of tests were conducted, followed by qualitative and quantitative evaluation until the optimal detection method was found.

The second approach concerns the training of a deep learning network in order to be able to detect the bounding box an the outline of car doors in images. At first training and validation data were created and the training process used a pretrained model as a starting point for retraining. Various tests were conducted, which were evaluated qualitatively and quantitatively to select the final model was created. It should also be stated that the final model was decided to detect both doors and humans in the scene, due to the occlusions that occur from humans interacting with the car doors.

Finally, some main conclusions and comments are presented together with proposals for future work.

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Περιεχόμενα

1 Εισαγωγή

1.1 Βασικό κίνητρο

Η παρούσα εργασία αφορά την παρακολούθηση μιας πόρτας αυτοκινήτου κατά τη διάρκεια της συναρμολόγησής της σε βιομηχανικό περιβάλλον. Η συναρμολόγηση της πόρτας γίνεται από ανθρώπους οι οποίοι, καθώς η πόρτα κινείται πάνω σε έναν ταινιόδρομο, προσαρμόζουν πάνω της τα διάφορα εξαρτήματα που απαιτούνται. Η εργασία εντάσσεται στο πλαίσιο έρευνας που αφορά τη συνεργασία ανθρώπων και ρομπότ σε γραμμή παραγωγής εντός βιομηχανικού περιβάλλοντος. Ο κάθε κύκλος συναρμολόγησης της πόρτας εκτελείται σε συγκεκριμένο χρονικό πλαίσιο και οι ενέργειες και οι εργασίες από τον κάθε άνθρωπο σε κάθε σταθμό εργασίας είναι καθορισμένες. Η παρακολούθηση της πόρτας αυτοκινήτου είναι καθοριστική για την ανάλυση του κύκλου εργασιών καθώς μπορεί να χρησιμοποιηθεί μαζί με τις κινήσεις των ανθρώπων σαν στοιχείο στη μετέπειτα εκπαίδευση δικτύων βαθιάς μάθησης για τον προσδιορισμό μιας υποεργασίας ή φάσης ενός συγκεκριμένου κύκλου. Αυτή η πληροφορία μπορεί να βοηθήσει ένα έξυπνο σύστημα ή ρομπότ να αποκτήσει γνώση για στοιχεία του περιβάλλοντος και των δράσεων που εκτελούνται και αντίστοιχα να λάβει αποφάσεις για συγκεκριμένες ενέργειες.

Το αντικείμενο (πόρτα αυτοκινήτου) έχει σχετικά απλό και συμπαγές σχήμα μεταξύ ορθογωνίου παραλληλογράμμου και τραπεζίου, με καμπυλωμένες ακμές. Μόνη αξιοσημείωτη ιδιαιτερότητα είναι η τρύπα του παραθύρου, η οποία εντοπίζεται στο πάνω μέρος της πόρτας και δημιουργεί ένα λεπτό πλαίσιο στις τρεις πλευρές γύρω της. Επιπλέον η πόρτα αυτοκινήτου είναι τοποθετημένη σε μία βάση στήριξης που την κρατά σε κατακόρυφη θέση, ώστε να μπορεί να υλοποιείται η συναρμολόγηση. Η πόρτα αυτοκινήτου μαζί με τον χώρο που την περιβάλλει (γραμμή παραγωγής) παρακολουθείται, και από τις δύο πλευρές, από ζεύγη καμερών (στέρεο-κάμερες) οι οποίες καταγράφουν την διαδικασία συναρμολόγησης. Οι καταγραφές αυτές αποτελούν την οπτική πληροφορία που λαμβάνεται ως είσοδος για το πρόβλημα και είναι έγχρωμες εικόνες/ακολουθίες εικόνων. Τα δεδομένα αυτά αξιοποιούνται στην επεξεργασία για τον εντοπισμό του σχήματος και της θέσης του αντικειμένου, δηλαδή της πόρτας αυτοκινήτου. Επομένως, για τον εντοπισμό διατίθενται παθητικοί αισθητήρες και συγκεκριμένα κάμερες και αυτός ο περιορισμός προσδίδει ιδιαίτερο ενδιαφέρον στο πρόβλημα του εντοπισμού. Η συλλογή των δεδομένων είναι απλή, χωρίς μεγάλο κόστος, καθώς δεν απαιτεί ιδιαίτερα εξειδικευμένο εξοπλισμό όπως ενεργητικούς αισθητήρες (μη οπτικοί αισθητήρες).

Ο παθητικός αισθητήρας που χρησιμοποιήθηκε για τη συλλογή των δεδομένων είναι η κάμερα StereoLabs ZED 2 [1]. Πρόκειται για μία στέρεο-κάμερα κατάλληλη για εφαρμογές όρασης υπολογιστών και δημιουργίας ψηφιακών διδύμων (digital twins [2]), η οποία σαφώς καταγράφει εικόνες και βίντεο αλλά δημιουργεί, με το λογισμικό που την συνοδεύει, εικόνες βάθους για κάθε σκηνή. Το λογισμικό της συγκεκριμένη κάμερας χρησιμοποιεί τεχνικές βαθιάς μάθησης για την εκτίμηση του βάθους από εικόνες, ενώ μπορεί να προσδιορίζει και τη θέση της κάμερας από δεδομένα GNSS [3].

1.2 Στόχος

Στόχος της εργασίας είναι η δημιουργία ενός αλγορίθμου οπτικού εντοπισμού και παρακολούθησης για πόρτες υπο συναρμολόγηση σε γραμμή παραγωγής αξιοποιώντας διαθέσιμα δεδομένα εικόνων. Το πρόβλημα που καλείται να λύσει ο αλγόριθμος είναι ένα πρόβλημα Υπολογιστικής Όρασης, ουσιαστικά βασίζεται σε μεθόδους Υπολογιστικής Όρασης με χρήση τεχνικών βαθιάς μάθησης (Deep Learning) και συνελικτικών νευρωνικών μοντέλων (CNN). Απώτερος σκοπός είναι η λήψη και επεξεργασία οπτικής πληροφορίας-εικόνων σε πραγματικό χρόνο. Στο πλαίσιο της συγκεκριμένης εργασίας η εφαρμογή πραγματικού χρόνου προσεγγίζεται με τη χρήση βίντεο (ακολουθίες εικόνων), ώστε να μπορούν να γίνουν δοκιμές. Οι ακολουθίες εικόνων που χρησιμοποιήθηκαν απεικονίζουν διάφορες σκηνές από βιομηχανικά περιβάλλοντα όπου συναρμολογούνται πόρτες αυτοκινήτου από ανθρώπους.

Ο εντοπισμός και η παρακολούθηση της πόρτας σε σειρά εικόνων έγινε με βάση δύο διαφορετικές προσεγγίσεις. Η πρώτη προσέγγιση εστιάζει στην ανίχνευση ειδικών δυαδικών στόχων (ArUco markers) που εχουν τοποθετηθεί στη πλατφορμα φόρτωσης ενώ η δεύτερη προσέγγιση εξετάζει την ανίχνευση του πλαισίου της πόρτας εκπαιδεύοντας ένα δίκτυο βαθιάς μάθησης. Στόχος εδώ είναι η επανεκπαίδευση ενός νευρωνικού μοντέλου για τον εντοπισμό πόρτας με νέα δεδομένα που έχουν συλλεχθεί για τις ανάγκες της εργασίας. Η ανίχνευση του πλαισίου της πόρτας είναι ένα πρόβλημα εντοπισμού αντικειμένου (object detection) και κατάτμησης (segmentation).

1.3 Απαιτήσεις

Οι απαιτήσεις στη συγκεκριμένη εργασία χωρίζονται με βάση τις δύο προσεγγίσεις: την ανίχνευση των στόχων ArUco και την εκπαίδευση του νευρωνικού δικτύου.

- **Ανίχνευση στόχων ArUco:**
  Στο συγκεκριμένο στάδιο η λειτουργική απαίτηση είναι η δημιουργία ενός αλγορίθμου που θα ανιχνεύει τους στόχους ArUco στα δεδομένα. Η μετρική απαίτηση δεν είναι αυστηρά καθορισμένη. Επομένως θα θεωρηθεί ικανοποιητική η ανίχνευση των στόχων με επίπεδο ορθότητας (accuracy) μεγαλύτερο του 50%, δηλαδή σε 100 εικόνες να ανιχνεύονται ορθά στόχοι σε περισσότερες από 50 από αυτές. Προυπόθεση όμως είναι το μικρό πλήθος λανθασμένα ορθών εντοπισμών (false positives), κάτω από 5%.

- **Εκπαίδευση συνελικτικού νευρωνικού δικτύου:**
  Σε αυτό το στάδιο λειτουργική απαίτηση είναι η δημιουργία ενός μοντέλου που θα εντοπίζει και θα κατατμεί τις πόρτες αυτοκινήτου σε εικόνες, εκπαιδεύοντας ένα συνελικτικό νευρωνικό δίκτυο. Επίσης είναι σημαντικό η κατάτμηση να μπορεί να προσδιορίσει με ακρίβεια το κέντρο της πόρτας, δηλαδή οι μάσκες κατάτμησης που θα προκύπτουν να επισημειώνουν ολόκληρο το αντικείμενο κάθε φορά. Ούτε σε αυτό

το στάδιο υπάρχει αυστηρά καθορισμένη μετρική απαίτηση. Δεδομένου ότι ένα τέλειο μοντέλο θα έχει μέση ακρίβεια (Average Precision) 1 (100%), θα πρέπει η μέση ακρίβεια του τελικού μοντέλου να είναι όσο το δυνατό πιο κοντά στο 1. Από την άλλη ένα μοντέλο με μέση ακρίβεια 50% θεωρείται τυχαίο, δηλαδή το αν θα υπάρξει επιτυχής εντοπισμός είναι τυχαίο, πράγμα που φυσικά δεν είναι επιθυμητό. Επομένως, σαν μετρική απαίτηση μπορεί να θεωρηθεί η μέση ακρίβεια (Average Precision) να είναι >50% και όσο πιο κοντά στο 100% γίνεται.

1.4 Δομή κειμένου

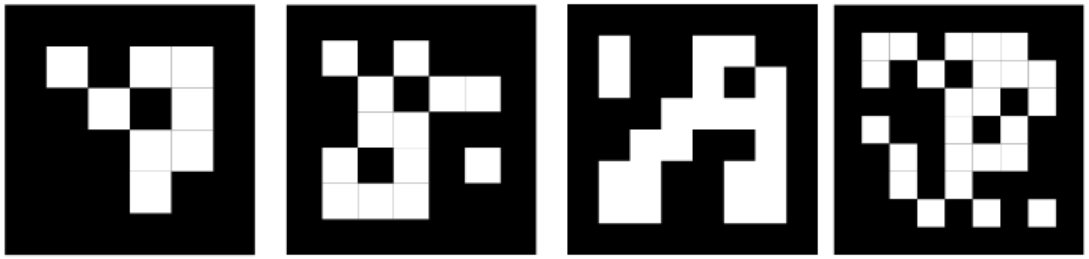
Το παρόν κείμενο διαρθρώνεται ως εξής. Αρχικά αναλύεται συνοπτικά το θεωρητικό υπόβαθρο που μελετήθηκε για την περάτωση της εργασίας και στη συνέχεια παρουσιάζονται οι βασικές έννοιες που χρησιμοποιούνται. Έπειτα, αναλύεται η μεθοδολογία που εφαρμόστηκε, τα πειράματα που υλοποιήθηκαν, για την ανίχνευση των στόχων ArUco και την εκπαίδευση του νευρωνικού δικτύου, οι αποφάσεις που πάρθηκαν σε κάθε στάδιο, όπως και η μέθοδος για την ποσοτική αξιολόγηση που εφαρμόστηκε. Ακολουθεί η παρουσίαση των αποτελεσμάτων, η ανάλυση και ο σχολιασμός τους (ποιοτική και ποσοτική αξιολόγηση). Τέλος, διατυπώνονται κάποια γενικά συμπεράσματα όπως και προτάσεις για βελτίωση ή για μελλοντική εργασία.

2 Θεωρητικό υπόβαθρο

2.1 Οπτική ανίχνευση βάσει δυαδικών στόχων (Marker-based detection)

Ο προσδιορισμός της θέσης ενός αντικειμένου στο χώρο βάσει προσημασμένων στόχων είναι μια μέθοδος γνωστή και από τη φωτογραμμετρία. Σε μια φωτογραμμετρική διαδικασία μπορούν να χρησιμοποιηθούν προσημασμένοι στόχοι των οποίων οι συντεταγμένες έχουν προσδιοριστεί (πιθανώς με μία άλλη μέθοδο σε κάποιο σύστημα αναφοράς) και με τη βοήθειά τους γίνεται ο προσδιορισμός της θέσης του αντικειμένου στο χώρο. Όταν η ανίχνευση αντικειμένων γίνεται σε εικόνες το σύστημα αναφοράς είναι το σύστημα εικονοσυντεταγμένων των εικόνων με αφετηρία (0, 0) το πάνω αριστερό άκρο της κάθε εικόνας.

Οι στόχοι που χρησιμοπιήθηκαν ονομάζονται ArUco (ArUco markers) [4] και πρόκειται για τετράγωνους δυαδικούς στόχους με εκτεταμένη χρήση σε εφαρμογές όρασης υπολογιστών. Οι στόχοι ArUco αποτελούνται από ένα μαύρο περίγραμμα, που επιτρέπει τη γρήγορη ανίχνευση, και ένα δυαδικό πίνακα στο εσωτερικό του, που λειτουργεί ως το αναγνωριστικό του κάθε στόχου.



Σχήμα 1: Παραδείγματα στόχων ArUco, πηγή: www.researchgate.net

Οι στόχοι ArUco οργανώνονται σε λεξικά (dictionaries) με παραμέτρους τις διαστάσεις των στόχων ArUco, το πλήθος των στόχων που απαρτίζουν το λεξικό και το inter-marker distance. Η παράμετρος inter-marker distance επιτρέπει στο λεξικό να εντοπίζει και να διορθώνει σφάλματα. Το inter-marker distance είναι η απόσταση Hamming μεταξύ των στόχων που περιέχει το λεξικό. Οι στόχοι ArUco αντιμετωπίζονται ως μια ακολουθία από 0 και 1 και η απόσταση Hamming είναι το ελάχιστο πλήθος ψηφίων που πρέπει να αλλαχθούν για την μετάβαση από τον ένα στόχο στον άλλο [5]. Τα λεξικά ArUco (ArUco dictionaries) υπάρχουν έτοιμα και μπορούν να χρησιμοποιηθούν ως έχουν σε εφαρμογές είτε να δημιουργηθούν νέα ανάλογα με τις απαιτήσεις της κάθε εργασίας.

Προκειμένου να χρησιμοποιηθούν οι στόχοι εκτυπώνονται και τοποθετούνται πάνω στο αντικείμενο ενδιαφέροντος προκειμένου να φωτογραφηθούν. Στη συγκεκριμένη εργασία η προσπάθεια ανίχνευσης των στόχων ArUco έγινε σε εικόνες. Στις εικόνες που χρησιμοποιήθηκαν ως δεδομένα ενυπάρχει ένας στόχος ArUco ο οποίος έχει τοποθετηθεί πάνω στο πλαίσιο στήριξης της πόρτας αυτοκινήτου σε συγκεκριμένη θέση ώστε να

καταγράφεται από την κάμερα καθώς η πόρτα κινείται πάνω στον ταινιόδρομο.

Η ανίχνευση των στόχων ArUco εικόνες επιστρέφει το σύνολο των αναγνωρισμένων στόχων και για κάθε έναν απ' αυτούς γνωστοποιεί τις θέσεις των 4 κορυφών του και την ταυτότητά του (id). Η διαδικασία της ανίχνευσης έχει δύο βήματα, τον εντοπισμό των περιοχών της εικόνας που είναι πιθανοί στόχοι και στη συνέχεια την αναγνώριση του (id) του στόχου. Στο πρώτο βήμα εφαρμόζεται προσαρμοστική κατωφλίωση (adaptive thresholding) προκειμένου να κατατμηθούν οι στόχοι, στη συνέχεια εξάγονται τα περιγράμματα και όσα δεν είναι κυρτά ή δεν έχουν σχήμα που προσεγγίζει το τετράγωνο απορρίπτονται. Εφαρμόζεται και ένα επιπλέον φιλτράρισμα όπου απορρίπτονται τα περιγράμματα που είναι πολύ μικρά, πολύ μεγάλα, ή είναι πολύ κοντά μεταξύ τους. Στο δεύτερο βήμα οι περιοχές τις εικόνας που έχουν αναγνωρισθεί ως πιθανοί στόχοι μετασχηματίζονται κατάλληλα ώστε να αποκτήσουν τερτράγωνο σχήμα, έπειτα κατωφλιώνονται χρησιμοποιώντας την μέθοδο Otsu [6] για να διαχωριστεί η εικόνα σε λευκά και μαύρα εικονοστοιχεία. Οι περιοχές αυτές χωρίζονται σε τμήματα - ανάλογα με το μέγεθος του δυαδικού πίνακα των αναμενόμενων στόχων και το μέγεθος του περιγράμματός τους - και σε κάθε ένα από αυτά καταμετρώνται τα λευκά και μαύρα εικονοστοιχεία για να διαπιστωθεί αν το υπό εξέταση τμήμα είναι λευκό ή μαύρο. Τέλος, ελέγχεται αν ο υποψήφιος στόχος ανήκει σε κάποιο λεξικό και αν όντως ανήκει η ανίχνευση είναι επιτυχής.

Η διαδικασία ανίχνευσης περιλαμβάνει ένα πλήθος παραμέτρων (βλ. κεφάλαιο 3.1 για περισσότερη ανάλυση των παραμέτρων εντοπισμού) που μπορούν να τροποποιηθούν από τον χρήστη βελτιστοποιώντας την ανίχνευση ανάλογα με τις ιδιαιτερότητες της κάθε εφαρμογής.

2.2 Συνελικτικά Νευρωνικά Δίκτυα (CNNs)

Τα **νευρωνικά δίκτυα** (Neural Networks) [7] είναι μια μέθοδος τεχνιτής νοημοσύνης/μηχανικής μάθησης, συγκεκριμένα βαθιάς μάθησης που χρησιμοποιείται στον προγραμματισμό για την επίλυση σύνθετων προβλημάτων. Αντί να καταστρώνονται πολύπλοκοι και εκτεταμένοι αλγόριθμοι και έτσι ο υπολογιστής να είναι σε θέση να λύσει ένα πρόβλημα, τα νευρωνικά δίκτυα επιτρέπουν στον υπολογιστή να μαθαίνει παρατηρώντας όπως κάνουν και οι άνθρωποι. Όπως ισχύει και για τους ανθρώπους, προκειμένου να μάθει ένα νευρωνικό δίκτυο να λύνει ένα πρόβημα χρειάζεται μεγαλός όγκος δεδομένων. Αυτά τα δεδομένα ονομάζονται δεδομένα εκπαίδευσης στα οποία, προκειμένου να γίνει η εκπαίδευση, υποδεικνύεται με κάποιο τρόπο στο νευρωνικό δίκτυο αυτό που καλείται να μάθει.

Τα νευρωνικά δίκτυα αποτελούνται από επίπεδα (layers) καθένα από τα οποία μετασχηματίζει με κάποιο τρόπο τα δεδομένα. Όλα τα νευρωνικά δίκτυα διαθέτουν ένα επίπεδο εισόδου (input layer), ένα επίπεδο εξόδου (output layer) και ενδιάμεσα επίπεδα (hidden layers). Αν το δίκτυο διαθέτει περισσότερα από 2 ενδιάμεσα επίπεδα τότε καλείται βαθύ νευρωνικό δίκτυο (Deep Neural Network).

Η εκπαίδευση των νευρωνικών δικτύων γίνεται με τη μέθοδο empirical risk minimiza-

tion [8], δηλαδή ελαχιστοποιούν το ρίσκο της πρόβλεψης, που κάνουν, εμπειρικά. Το ρίσκο αφορά την πιθανότητα το δίκτυο να προβλέψει λανθασμένα μια τιμή σε σχέση με την πραγματική τιμή που υπάρχει στα δεδομένα. Η μέθοδος ελαχιστοποίησης του ρίσκου βελτιστοποιεί τις παραμέτρους του δικτύου ώστε αυτό να είναι σε θέση να κάνει όσο το δυνατό λιγότερες λάθος προβλέψεις. Η εκτίμηση των παραμέτρων του νευρωνικού δικτύου γίνεται με μεθόδους που βασίζονται στην εκτίμηση με βάση το ανάδελτα gradient [9], δηλαδή στο διάνυσμα που δείχνει πως μεταβάλλεται ένα μέγεθος στον τριδιάστατο χώρο. Κατά την εκπαίδευση, το νευρωνικό δίκτυο μαθαίνει από επισημειωμένα δεδομένα εκπαίδευσης τροποποιώντας επαναλαμβανόμενα τις παραμέτρους του ώσπου να ελαχιστοποιηθεί η συνάρτηση loss [10]. Η συνάρτηση loss [10] αναπαριστά τη διαφορά μεταξύ της της πρόβλεψης που κάνει το δίκτυο και της παραγματικής τιμής. Τα δεδομένα εκπαίδευσης αρχικά περνούν απ το δίκτυο (forward propagation) [11], απ' όλα του τα επίπεδα ώστε να γίνει η πρόβλεψη, υπολογίζεται το loss και στη συνέχεια τα δεδομένα εκπαίδευσης περνούν αντίστροφα από το τελευταίο επίπεδο στο πρώτο (backpropagation) [11], ώστε να εκτιμηθεί πόσο πρέπει να τροποποιηθούν οι παράμετροι του δικτύου με βάση το gradient. Κάθε μία τροποποίηση των παραμέτρων του δικτύου αποτελεί και μία επανάληψη iteration. Η κάθε επανάληψη δεν περνάει απαραίτητα όλο το σύνολο των δεδομένων εκπαίδευσης από το νευρωνικό δίκτυο, αλλά από ένα υποσύνολό τους το οποίο μπορεί να οριστεί με κριτήριο τη μνήμη που διαθέτει ο υπολογιστής που πραγματοποιεί την εκπαίδευση του νευρωνικού δικτύου. Όσο περισσότερη είναι η διαθέσιμη μνήμη, τόσο και μεγαλύτερο μέρος των δεδομένων εκπαίδευσης μπορεί να χρησιμοποιηθεί σε μία επανάληψη. Ένα πλήρες πέρασμα όλων των διαθέσιμων δεδομένων εκπαίδευσης από το νευρωνικό δίκτυο αποτελεί ένα epoch.

Τα **συνελικτικά νευρωνικά δίκτυα** (Convolutional Neural Networks) [14] χρησιμοποιούνται πολύ συχνά σε εφαρμογές ταξινόμησης και υπολογιστικής όρασης και γενικότερα για την επεξεργασία δεδομένων που οργανώνονται σε πίνακες (εικόνες, χρονοσειρές). Ο πυρήνας των συνελικτικών νευρωνικών δικτύων και αυτό που τα διαφοροποιεί από τα νευρωνικά δίκτυα είναι η συνέλιξη, μια πράξη μεταξύ δύο πινάκων (μιας εικόνας και ενός φίλτρου) που απαντάται σε ένα η περισσότερα από τα ενδιάμεσα επίπεδά τους.

Η συνέλιξη περνάει ένα φίλτρο (kernel) 2 διαστάσεων από τον πίνακα των δεδομένων και το αποτέλεσμά της είναι ένα feature map. Πρακτικά, υπολογίζεται το γινόμενο πινάκων μεταξύ του τμήματος του πίνακα των δεδομένων που καλύπτει κάθε φορά το φίλτρο και του ίδιου του φίλτρου και αυτό αποτελεί το πρώτο στοιχείο του feature map. Στη συνέχεια το φίλτρο κινείται προς τα δεξιά, διατρέχοντας τελικά ολόκληρο τον πίνακα των δεδομένων. Αυτή η διαδικασία ελλατώνει τις διαστάσεις του feature map σε σχέση με τις διαστάσεις του πίνακα των δεδομένων κατά τόσα στοιχεία, στην οριζόντια και την κατακόρυφη διάσταση, όσα το διπλάσιο του ακέραιου μέρους της διαίρεσης της διάστασης του φίλτρου με το 2. Δηλαδή αν το φίλτρο έχει διάσταση 3*3 και ο αρχικός πίνακας δεδομένων 11*11, τότε το feature map που θα προκύψει θα έχει διάσταση 9*9. Αντίστοιχα αν το φίλτρο είχε διάσταση 5*5, το feature map θα είχε διάσταση 7*7. Αυτό αντιμετωπίζεται με την προσθήκη padding

περιμετρικά του αρχικού πίνακα δεδομένων, τόσο όσο το πλήθως των στοιχείων που θα έχανε ο πίνακας αν δεν υπήρχε το padding. Συνήθως στο padding χρησιμοποιείται η τιμή 0 (zero padding) αλλά μπορούν να χρησιμοποιηθούν και άλλες τιμές. Το βήμα (stride) με το οποίο κινείται το φίλτρο στα δεδομένα συνήθως έχει την τιμή 1, αλλά μπορεί να πάρει και μεγαλύτερες τιμές, όμως το feature map που θα προκύψει θα έχει ακόμα μικρότερη διάσταση σε σχέση με τα αρχικά δεδομένα.

Τα συνελικτικά νευρωνικά δίκτυα αποτελούνται από 3 ειδών επίπεδα:

- **Convolutional layer**: επίπεδα συνέλιξης όπου αποτελούνται από πράξεις συνέλιξης μεταξύ των δεδομένων εισόδου, σε μορφή πίνακα, και ενός φίλτρου (kernel).
- **Pooling layer**: παρόμοια με τα επίπεδα συνέλιξης και αυτά περνούν ένα φίλτρο από τα δεδομένα αλλά το φίλτρο έχει ως στόχο την μείωση του όγκου των δεδομένων, κάνοντας μια σύνοψη των γειτονικών δεδομένων. Υπάρχουν διάφορα είδη pooling layer με τα κυριότερα να είναι τα επίπεδα max pooling (μέγιστη τιμή ανά περιοχή που περνάει το φίλτρο), τα επίπεδα average pooling (μέση τιμή ανά περιοχή), ενώ μπορεί να υπολογίζεται και ένας σταθμισμένος μέσος με βάση την απόσταση από το κεντρικό στοιχείο.
- **Fully-connected (FC) layer**: πρόκειται για έναν μετασχηματισμό που συνδέει το επίπεδο εισόδου με το επίπεδο εξόδου.

Τα παραπάνω επίπεδα εμφανίζονται στη δομή ενός συνελικτικού δικτύου με την εξής σειρά. Αρχικά τα δεδομένα περνούν από ένα επίπεδο συνέλιξης (convolutional layer) και ακολουθούν επιπλέον επίπεδα συνέλιξης ή pooling layers, σε αλληλουχία που εξαρτάται από την αρχιτεκτονική που ακολουθείται. Στο τέλος συνήθως τοποθετούνται ένα ή περισσότερα fully-connected (FC) layers. Τα πρώτα επίπεδα του δικτύου επικεντρώνονται στα πιο απλά χαρακτηριστικά όπως χρώματα και ακμές, ενώ στα επόμενα επίπεδα το δίκτυο αρχίζει να αναγνωρίζει σχήματα και όλο και πιο σύνθετα χαρακτηριστικά μέχρι να φτάσει στην αναγνώριση των ζητούμενων αντικειμένων.

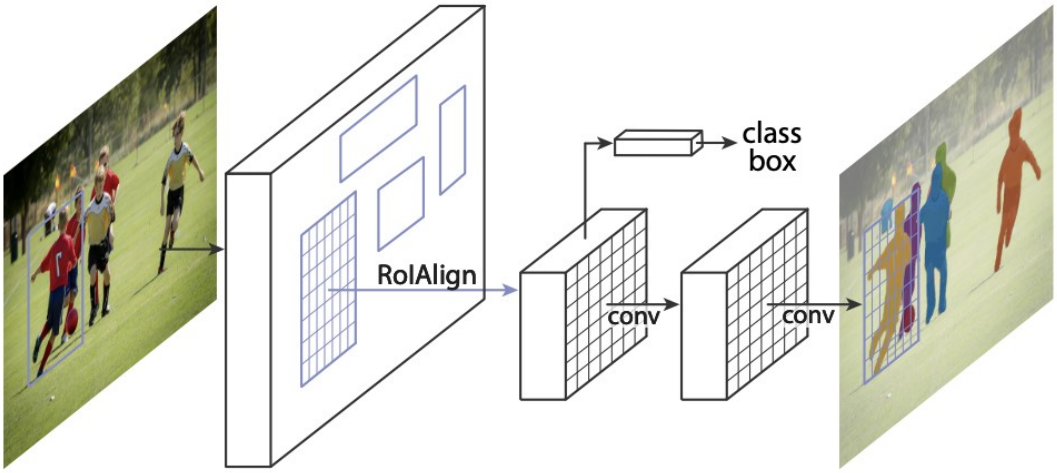
2.3 Mask R-CNN

Ο αλγόριθμος Mask R-CNN [15] αποτελεί μια απλή και ευέλικτη δομή για κατάτμηση εικόνων που δημιουργήθηκε το 2017 επεκτείνοντας τις δυνατότητες των προηγούμενων R-CNN [17], Fast R-CNN [18] και Faster R-CNN [19], προσθέτοντας την δυνατότητα να δημιουργεί μάσκες για τα αντικείμενα που αναγνωρίζει. Ο αλγόριθμος Mask R-CNN αναγνωρίζει αντικείμενα σε εικόνες και προσδιορίζει λεπτομερώς τη θέση τους, αφενός δίνοντας το πλαίσιο οριοθέτησης του κάθε αντικειμένου και αφετέρου δημιουργώντας την μάσκα κατάτμησης του. Η μάσκα κατάτμησης ορίζει ποιά από τα εικονοστοιχεία της εικόνας εικόνας αντιστοιχούν στο εκάστοτε αντικείμενο (object class).

Η διαδικασία που ακολουθεί ο αλγόριθμος Mask R-CNN έχει τη βάση της στον αλγόριθμο R-CNN: regions with CNN features (περιοχές με χαρακτηριστικά συνελικτικών

νευρωνικών δικτύων) και αποτελείται από 4 βήματα.

1. Εισαγωγή μιας εικόνας στο δίκτυο.
2. Εξαγωγή προτεινόμενων περιοχών, που μπορεί να περιέχουν αντικείμενα.
3. Εξαγωγή χαρακτηριστικών για κάθε περιοχή με τη χρήση ενός εκπαιδευμένου συνελικτικού νευρωνικού δικτύου.
4. Κατηγοριοποίηση περιοχών με βάση τα χαρακτηριστικά που έχουν εξαχθεί.



Σχήμα 2: Δομή Mask R-CNN για κατάτμηση αντικειμένων, πηγή: Mask R-CNN paper

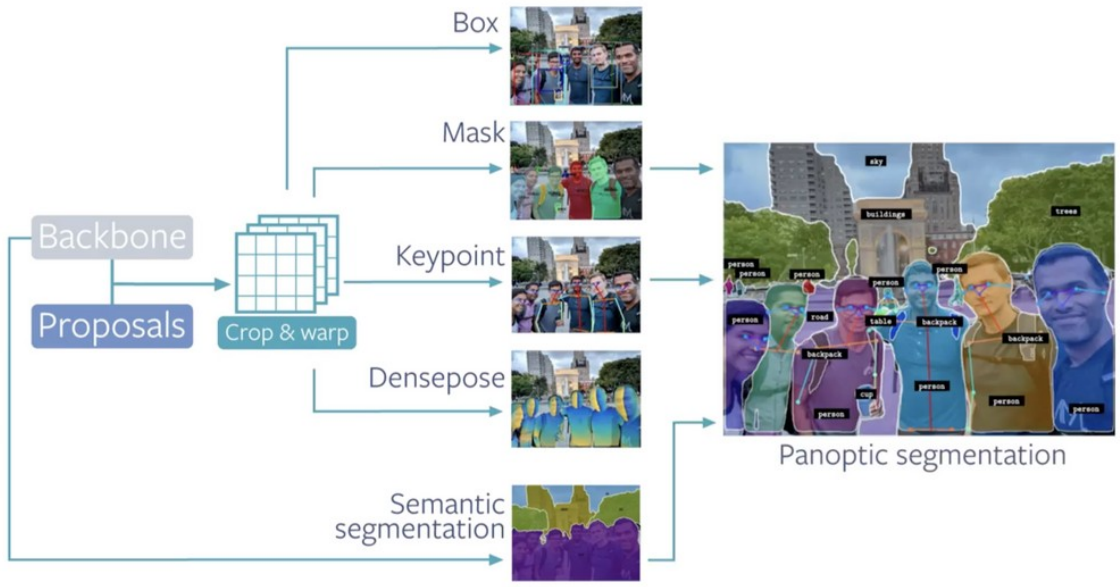
Βασικό πρόβλημα του αλγορίθμου R-CNN είναι ότι είναι αργός για εφαρμογές πραγματικού χρόνου, διότι μπορεί να χρειστεί χρόνος της τάξης των 40-50 δευτερολέπτων για την επεξεργασία μίας εικόνας σε έναν τυπικό υπολογιστή για εφαρμογές βαθιάς μάθησης [20]. Προκειμένου να αυξηθεί η ταχύτητα δημιουργήθηκε ο αλγόριθμος Fast R-CNN, ο οποίος διαφοροποιεί το βήμα της εξαγωγής προτεινόμενων περιοχών από τη διαδικασία Selective Search (επιλεκτική αναζήτηση) σε ROI: Regions Of Interest (περιοχές ενδιαφέροντος) οι οποίες πάλι βασίζονται στην επιλεκτική αναζήτηση. Αυτή η αλλαγή καθιστά τον αλγόριθμο Fast R-CNN εκπαιδεύσιμο από την αρχή μέχρι το τέλος. Δεδομένου ότι και πάλι ο αλγόριθμος βασίζεται στην επιλεκτική αναζήτηση το βήμα της εξαγωγής προτεινόμενων περιοχών απαιτεί πάλι κάποιο χρόνο. Ο αλγόριθμος Fast R-CNN είναι σε θέση να επεξεργάζεται μία εικόνα σε περίπου 2 δευτερόλεπτα [20], σε ένα τυπικό υπολογιστή για βαθιά μάθηση, καθιστώντας τον χρησιμοποιήσιμο σε εφαρμογές πραγματικού χρόνου.

Αυτό που χρειάστηκε να γίνει για να αυξηθεί η ταχύτητα του αλγορίθμου ήταν η ενσωμάτωση της διαδικασίας εξαγωγής προτεινόμενων περιοχών στα βήματα του αλγορίθμου R-CNN. Αυτό έκανε ο αλγόριθμος Faster R-CNN χρησιμοποιώντας την διαδικασία RPN: Region Proposal Network (δίκυο προτεινόμενων περιοχών), εξαλείφωντας την ανάγκη

για χρήση της διαδικασίας της επιλεκτικής αναζήτησης (Selective Search). Ο αλγόριθμος αυτός επιτυγχάνει να εκτελείται με μια ταχύτητα της τάξης των 7 με 10 εικόνων ανά δευτερόλεπτο.

Τέλος, ο αλγόριθμος Mask R-CNN βελτιώνει περαιτέρω τις δυνατότητες των προηγούμενων αλγορίθμων. Η ταχύτητα εκτέλεσής του φτάνει τις 5 εικόνες ανά δευτερόλεπτο, δεδομένου ότι προσθέτει και την δυνατότητα δημιουργίας μασκών κατάμησης για τα αντικείμενα που εντοπίζει, όπως αναφέρθηκε και παραπάνω.

2.4 Detectron2



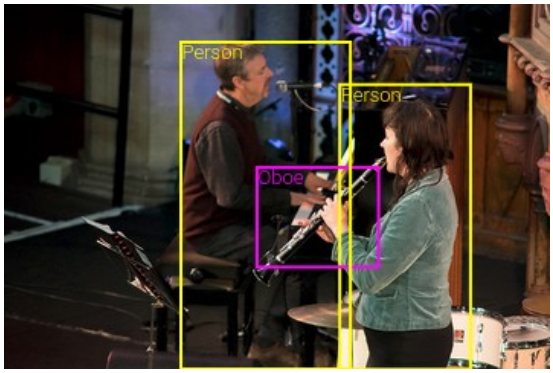
Σχήμα 3: Detectron2, πηγή: blog.roboflow.com

Το Detectron2 [21] είναι μια εύχρηστη πλατφόρμα ανοιχτού κώδικα που χρησιμοποιείται ευρέως σε εφαρμογές ανίχνευσης αντικειμένων και κατάτμησης εικόνων που δημιουργήθηκε από την εταιρεία Meta και συγκεκριμένα από την Facebook AI Research (FAIR). Διαθέτει αποθετήριο (model zoo) με πλήθος από σύγχρονα ήδη εκπαιδευμένα μοντέλα που μπορούν να χρησιμοποιηθούν σε διαφόρων ειδών εφαρμογές. Επιπλέον διαθέτει έτοιμες συναρτήσεις που διευκολύνουν την επανεκαπίδευση των μοντέλων ώστε να εξατομικευτούν ανάλογα με τις απαιτήσεις της κάθε εφαρμογής αλλά και την αξιολόγηση τη εκπαίδευσης. Βασικό πλεονέκτημα της συγκεκριμένης πλατφόρμας, που την κατέστησε ιδιαιτέρως κατάλληλη για να χρησιμοποιηθεί στη συγκεκριμένη εργασία, είναι το γεγονός ότι διαθέτει ενσωματωμένο τον αλγόριθμο Mask R-CNN που παρουσιάστηκε παραπάνω.

2.5 Open Images Dataset

Το σύνολο δεδομένων Open Images [23] αποτελεί ένα ανοιχτό σύνολο δεδομένων που περιλαμβάνει περίπου 9 εκατομμύρια εικόνες. Για όλο αυτό το σύνολο εικόνων διατίθενται annotation που αποτελούνται από τα εξής:

1. Τα αντίστοιχα **labels** με τα ονόματα όλων των κατηγοριών που ενυπάρχουν.
2. Τα **πλαίσια οριοθέτησης** (bounding boxes) (σχήμα 4) για κάθε κατηγορία, δηλαδή είναι ορθογώνιο πλαίσιο που εμπεριέχει μία κατηγορία.



Σχήμα 4: Παράδειγμα πλαισίου οριοθέτησης, πηγή: Open Images Dataset

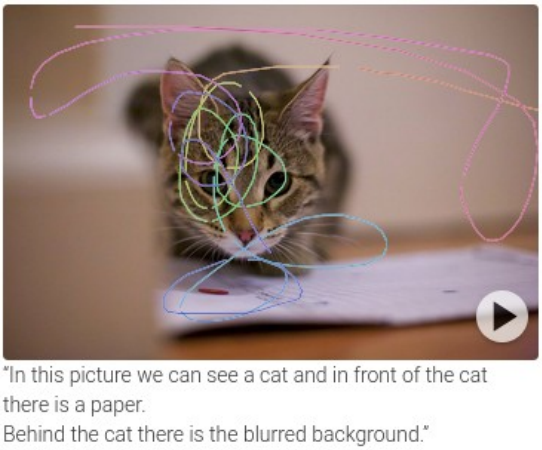
3. **Μάσκες κατάτμησης** (segmentation masks) (σχήμα 5), οι οποίες επισημαίνουν το περίγραμμα ενός αντικειμένου με μεγάλη λεπτομέρεια.

Σχήμα 5: Παράδειγμα μάσκας κατάτμησης, πηγή: Open Images Dataset

4. **Οπτικές σχέσεις** (visual relationships) (σχήμα 6), που επισημαίνουν ζευγάρια αντικειμένων που συσχετίζονται μεταξύ τους.

Σχήμα 6: Παράδειγμα οπτικής σχέσης, πηγή: Open Images Dataset

5. **Τοπικές αφηγήσεις** (localized narratives) (σχήμα 7), οι οποίες αποτελούν περιγραφείς που συνδυάζουν συγχρονισμένη φωνητική περιγραφή, κείμενο και την επίδειξη των περιοχών ή των αντικειμένων της εικόνας που αναφέρονται στην περιγραφή.



Σχήμα 7: Παράδειγμα τοπικής αφήγησης, πηγή: Open Images Dataset

Όλα τα παραπάνω μπορούν να χρησιμοποιηθούν σε εφαρμογές ταξινόμησης, ανίχνευσης αντικειμένων και εφαρμογές κατάμησης εικόνων που σε συνδυασμό με τις τοπικές αφηγήσεις υπστηρίζουν πλήρως η κατανόηση μιας σκηνής. Τα δεοδομένα αυτά δηλαδή μπορούν να χρησιμοποιηθούν ως δεδομένα εκπαίδευσης και ελέγχου κατά τη διαδικασία εκπαίδευσης ενός νευρωνικού δικτύου, χωρίς να απαιτείται η δημιουργία τους εφόσων αυτά καλύπτουν τις ανάγκες της εκάστοτε εφαρμογής.

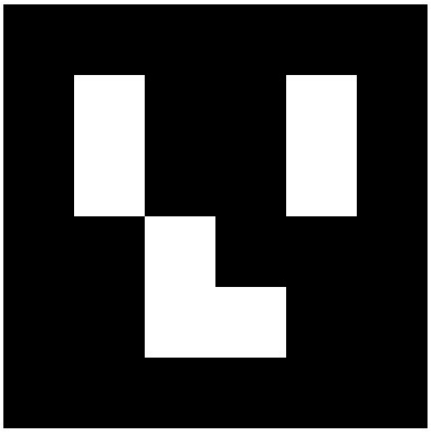
Αντίστοιχα με το σύνολο δεδομένων Open Images υπάρχουν και άλλα σύνολα δεοδμένων, όπως το COCO dataset [24], τα οποία διαφοροποιούνται ως προς το πλήθος των εικόνων, το πλήθος των κλάσεων που περιγράφουν και ως προς τη δομή των annotation που διαθέτουν.

3 Σχεδίαση, μεθοδολογία και υλοποίηση

3.1 Ανίχνευση στόχων ArUco

Το πρώτο στάδιο της εργασίας αφορά στην αναγνώριση και τον εντοπισμό ειδικών στόχων (ArUco markers), που έχουν τοποθετηθεί στη βάση στήριξης της πόρτας σε συγκεκριμένη θέση, μέσα από οπτική πληροφορία (εικόνες) που έχουν ληφθεί. Στο σχήμα 8 παρουσιάζεται η βάση στήριξης πάνω στην οποία τοποθετούνται οι πόρτες αυτοκινήτου αλλά και οι στόχοι ArUco. Η βάση στήριξης είναι μια μεταλική κατασκευή η οποία στηρίζει την πόρτα όσο αυτή συναρμολογείται, δίνοντάς της και την ικανότητα να κινείται στον κατακόρυφο άξονα. Διαθέτει ρόδες για να μπορεί να μετακινείται εύκολα και δύο επιφάνειες, δεξιά και αριστερά της πόρτας, όπου μπορούν να τοποθετηθούν προσωρινά εργαλεία και εξαρτήματα που χρησιμοποιούνται κατά την συναρμολόγηση της πόρτας. Στο αριστερό της μέρος (όπως φαίνεται στο σχήμα 8) είναι τοποθετημένος ο στόχος ArUco.

Σχήμα 8: Πόρτα αυτοκινήτου πάνω σε βάση στήριξης



Σχήμα 9: Στόχος ArUco με id 3

Στις πόρτες εντοπίζεται ένας προσημασμένος στόχος ArUco και συγκεκριμένα αυτός με id 3, ο οποίος παρουσιάζεται στο παραπάνω σχήμα 9.

Η ανίχνευση των στόχων ArUco έγινε μέσω της βιβλιοθήκης opencv [25] για την python και συγκεκριμένα μέσω του σχετικού πακέτου ”aruco”. Η opencv διαθέτει συναρτήσεις που επιτρέπουν τη δημιουργία, τον εντοπισμό στόχων ArUco και την βελτιστοποίηση της διαδικασίας εντοπισμού, προσαρμόζοντας τις παραμέτρους εντοπισμού.

Ο εντοπισμός των στόχων έγινε με δύο τρόπους, σε εικόνες και σε βίντεο. Αρχικά, συντάχθηκε ένα αρχείο κώδικα σε python που διατρέχει το βίντεο εισαγωγής και αποθηκεύει ένα προς ένα τα καρέ του ως εικόνες προκειμένου να γίνει εντοπισμός των στόχων σε αυτές. Ο εντοπισμός των στόχων σε βίντεο έγινε εξάγοντας κάθε καρέ του βίντεο στη μνήμη RAM του υπολογιστή και εκτελώντας τον εντοπισμό των ArUco. Ο εντοπισμός των ArUco πραγματοποιήθηκε πολύ έυκολα χρησιμοποιώντας τη συνάρτηση ”detectMarkers” του πακέτου ”aruco” της opencv. Πραγματοποιήθηκε μια πρώτη δοκιμή με τις προκαθορισμένες παραμέτρους εντοπισμού και τόσο στον εντοπισμό σε εικόνες, όσο και στο βίντεο το αποτέλεσμα ήταν το ίδιο. Διατηρήθηκε η διαδικασία εντοπισμού στόχων σε βίντεο για λόγους απλότητας, καθώς δεν απαιτεί το βήμα της εξαγωγής των καρέ του βίντεο ως εικόνες.

Στη συνέχεια πραγματοποιήθηκε μια σειρά δοκιμών εντοπισμού στόχων ArUco ξεκινώντας από τις προκαθορισμένες τιμές των παραμέτρων εντοπισμού και σταδιακά τροποιώντας ορισμένες απ’ αυτές ώστε να προκύψουν καλύτερα αποτελέσματα εντοπισμού. Στον παρακάτω πίνακα παρουσιάζονται οι παράμετροι εντοπισμού στόχων ArUco και οι τιμές τους σε όλες τις δοκιμές που πραγματοποιήθηκαν. Έγιναν 6 πειράματα τα οποία αναφέρονται στη συνέχεια ως «δοκιμή Δ0, δοκιμή Δ1 - δοκιμή Δ5» όπου η δοκιμή Δ0 είναι αυτή όπου οι παράμετροι εντοπισμού έχουν τις προκαθορισμένες τιμές.

Παράμετρος	**Δ0**	**Δ1**	**Δ2**	**Δ3**	**Δ4**	**Δ5**
adaptiveThreshWinSizeMin	5	5	**4**	4	4	4
adaptiveThreshWinSizeMax	25	25	**50**	50	50	50
adaptiveThreshWinSizeStep	5	5	**2**	2	2	2
adaptiveThreshConstant	7	7	7	**10**	10	10
minMarkerPerimeterRate	0,07	0,07	0,07	0,07	**0,06**	**0,05**
maxMarkerPerimeterRate	0,3	0,3	0,3	0,3	**0,4**	**0,5**
polygonalApproxAccuracyRate	0,05	**0,06**	0,06	0,06	0,06	0,06

Πίνακας 1: Τιμές παραμέτρων στα Πειράματα εντοπισμού στόχων ArUco

Οι παράμετροι που παρουσιάζονται στον παρακάτω πίνακα είναι αυτοί, οι τιμές των οποίων δεν τροποιήθηκαν διότι η μεταβολή τους από τις προκαθορισμένες τιμές είχε σχεδόν μηδενική επίδραση στο αποτέλεσμα του εντοπισμού. Επομένως στον παρακάτω πίνακα παρουσιάζονται οι υπόλοιπες παράμετροι εντοπισμού στόχων ArUco με τις προκαθορισμένες τους τιμές.

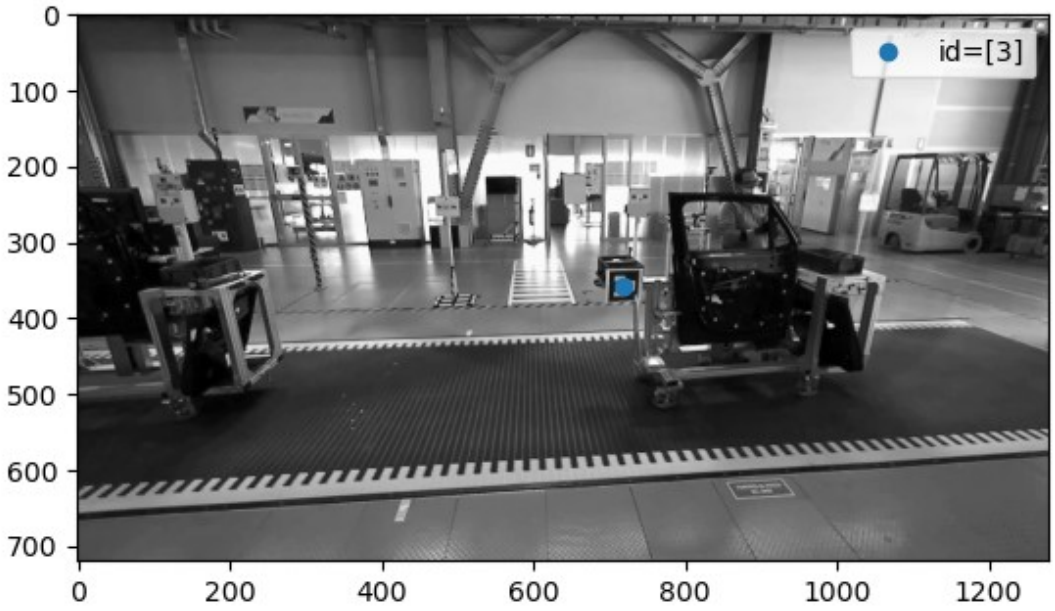
Παράμετρος	Τιμή στις δοκιμές Δ0-Δ5
minCornerDistanceRate	0,2
minMarkerDistanceRate	1,0
minDistanceToBorder	3
markerBorderBits	1
minOtsuStdDev	5,0
perspectiveRemovePixelPerCell	4
perspectiveRemoveIgnoredMarginPerCell	0,13
maxErroneousBitsInBorderRate	0,35
errorCorrectionRate	0,6
cornerRefinementMethod	1
cornerRefinementWinSize	5
cornerRefinementMaxIterations	30
cornerRefinementMinAccuracy	0,1

Πίνακας 2: Τιμές παραμέτρων εντοπισμού με σταθερή τιμή σε όλες τις δοκιμές

Δοκιμή Δ0:

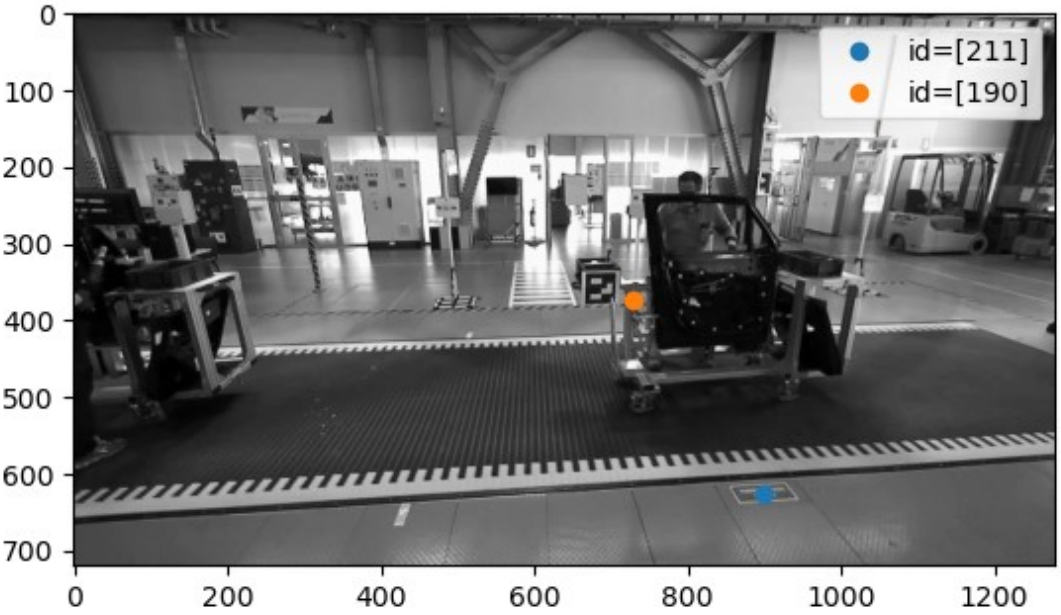
Στην πρώτη δοκιμή όπου οι τιμές των παραμέτρων εντοπισμού ήταν οι προκαθορισμένες τα αποτελέσματα ήταν προβληματικά. Συγκεκριμένα οι επιτυχημένοι εντοπισμοί, δηλαδή οι περιπτώσεις όπου ο εντοπισμένος στόχος ArUco ήταν στη σωστή θέση και είχε id 3 ήταν ελάχιστες (σχήμα 10). Στις περισσότερες περιπτώσεις δεν εντοπιζόταν κανένας στόχος και σε άλλες, ενώ υπήρχε εντοπισμός, αυτός δεν ήταν ορθός διότι εντοπίζονταν στόχοι που δεν υπήρχαν στις εικόνες, δηλαδή ο αλγόριθμος συνέχεε αλληλουχίες εικονοστοιχείων (pixel) που αναπαριστούν αντικείμενα του φόντου με στόχους ArUco (σχήμα 11, σχήμα 12).

```
Frame 50:
Detected AruCo ids: [[3]]
```

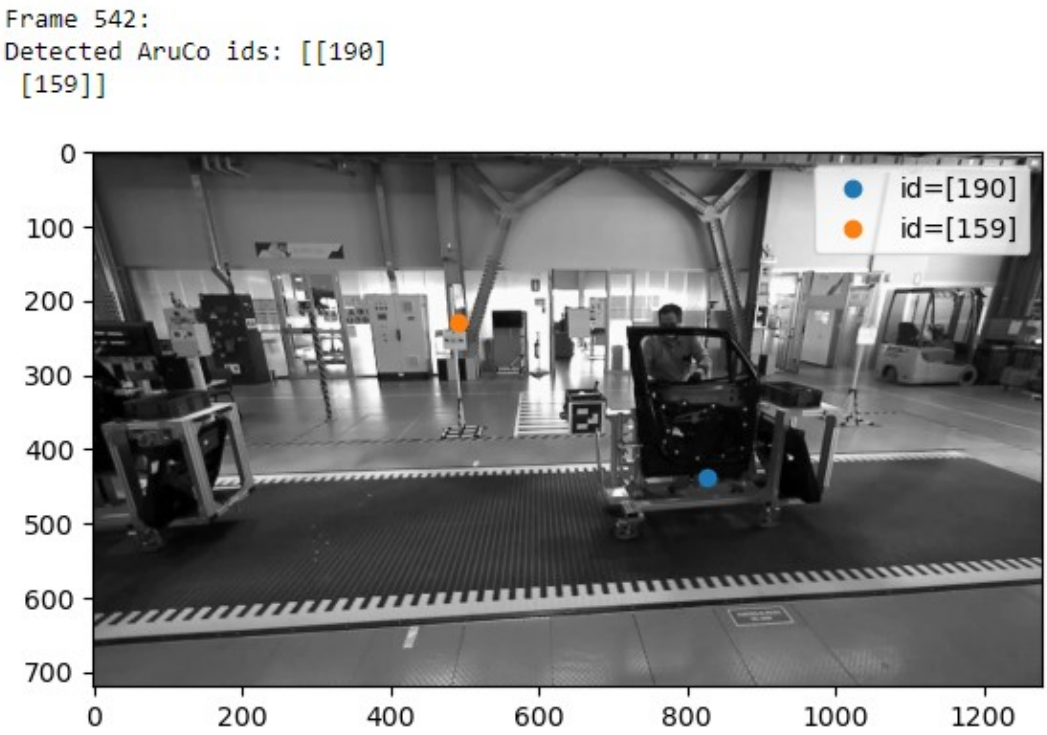


Σχήμα 10: Επιτυχημένος εντοπισμός στόχου ArUco

```
Frame 348:
Detected AruCo ids: [[211]
 [190]]
```



Σχήμα 11: Περίπτωση λανθασμένου εντοπιμού στόχων ArUco, 1



Σχήμα 12: Περίπτωση λανθασμένου εντοπιμού στόχων ArUco, 2

Δοκιμή Δ1:

Στη συνέχεια τροποποιούνταν διαδοχικά οι παράμετροι εντοπισμού με βάση τις ιδιαιτερότητες της σκηνής, που απεικονίζεται σε κάθε εικόνα, αλλά και μετά από δοκιμές. Σε πρώτη φάση τροποποιήθηκε η παράμετρος ”polygonalApproxAccuracyRate”, η οποία είναι το μέγιστο αποδεκτό σφάλμα της πολυγωνικής προσέγγισης, που χρησιμοποιείται για τον εντοπισμό των ArUco και μάλιστα για τον προσδιορισμό των υποψήφιων τετράγωνων περιοχών που θα εξεταστούν περαιτέρω. Μετά από δοκιμές σε εύρος τιμών από 0.03 έως 0.1 επιλέχθηκε σαν τιμή που δίνει τα καλύτερα αποτελέσματα η 0.06.

Δοκιμή Δ2:

Επόμενες παράμετροι που τροποποιήθηκαν ήταν οι ”adaptiveThreshWinSizeMin”, ”adaptiveThreshWinSizeMax” και ”adaptiveThreshWinSizeStep”. Αυτές αντιπροσωπεύουν το διάστημα (ελάχιστο, μέγιστο και βήμα) στο οποίο επιλέγεται το μέγεθος (σε εικονοστοιχεία) του παραθύρου κατωφλίωσης κατά τη διαδικασία της προσαρμοστικής κατωφλίωσης (adaptive thresholding). Ξεκινώντας από τις προκαθορισμένες τιμές: 3 (ελάχιστο), 23 (μέγιστο), 10 (βήμα), έγιναν διαδοχικές δοκιμές αυξομειώνοντας τις τιμές και τα καλύτερα αποτελέσματα προέκυψαν από τις τιμές 4, 50 και 20 για τις παραμέτρους ”adaptiveThreshWinSizeMin”, ”adaptiveThreshWinSizeMax” και ”adaptiveThreshWinSizeStep” αντίστοιχα.

Δοκιμή Δ3:

Η παράμετρος adaptiveThreshConstant αντιπροσωπεύει μια σταθερή τιμή που προστίθεται στην τιμή του κατωφλιού κατά την κατωφλίωση των εικόνων για τον εντοπισμού των στόχων ArUco. Η προκαθορισμένη τιμή είναι 7 και έπειτα από δοκιμές - τόσο με μικρότερες, όσο και με μεγαλύτερες τιμές - επιλέχθηκε η τιμή 10.

Δοκιμή Δ4:

Οι παράμετροι minMarkerPerimeterRate και maxMarkerPerimeterRate αντιπροσωπεύουν το μέγιστο και το ελάχιστο μέγεθος των στχωνό ArUco που ενυπάρχουν στα δεδομένα. Συγκεκριμένα πρόκειται για τη μέγιστη και την ελάχιστη περίμετρο των στόχων, όχι σε πλήθος εικονοστοιχείων, αλλά σε σχέση με τη μέγιστη διάσταση της εικόνας που δίνεται σαν είσοδος στον αλγόριθμο εντοπισμού. Με βάση τις διαστάσεις των εικόνων οι παράμετροι minMarkerPerimeterRate και maxMarkerPerimeterRate προέκυψαν 0.06 και 0.4 αντίστοιχα. Οι διαστάσεις εικόνων είναι 1280x720 εικονοστοιχεία. Εκτιμήθηκε η διάσταση του μικρότερου και του μεγαλύτερου ArUco (και αυτή σε εικονοστοιχεία) και από τη διαίρεση αυτών των μεγεθών με την μεγάλη διάσταση της εικόνας προέκυψαν οι τιμές των δύο παραμέτρων.

Δοκιμή Δ5:

Οι παράμετροι minMarkerPerimeterRate και maxMarkerPerimeterRate τροποιήθηκαν ξανά με δοκιμές προκειμένου να εξεταστεί αν μπορεί να αυξηθεί περαιτέρω η ακρίβεια του εντοπισμού. Έτσι οι τιμές 0.06 και 0.4 έγιναν 0.05 και 0.5 για το minMarkerPerimeterRate και το maxMarkerPerimeterRate αντίστοιχα. Σε αυτό το σημείο σημειώνεται ότι οι τροποίηση των υπόλοιπον παραμέτρων εντοπισμού δεν επέφερε βελτίωση στο αποτέλεσμα του εντοπισμού οπότε και διατηρήθηκαν οι προκαθορισμένες τιμές τους.

Ποιοτικά και ποσοτικά αποτελέσματα από τις παραπάνω δοκιμές παρουσιάζονται στο κεφάλαιο 4 Αποτελέσματα και αξιολόγηση.

3.2 Δημιουργία annotation

Η δημιουργία data annotation είναι η διαδικασία κατά την οποία επισημειώνονται τα δεδομένα εκπαίδευσης ενός μοντέλου μηχανικής μάθησης ώστε να του υποδειχθεί η πρόβλεψη που είναι επιθυμητό να κάνει. Όπως είναι προφανές είναι απαραίτητο βήμα που προηγείται της εκπαίδευσης ενός μοντέλου. Στην παρούσα εργασία, το data annotation έγινε με τη δημιουργία μασκών που επισημειώνουν στα δεδομένα (εικόνες) τα περιγράμματα πορτών αυτοκινήτου, προκειμένου το μοντέλο που θα εκπαιδευτεί να είναι σε θέση να αναγνωρίζει αυτό το αντικείμενο. Χρησιμοποιήθηκε το εργαλείο Vgg Image Annotator (V.I.A.) [26] το οποίο αποτελεί ένα απλό περιβάλλον δημιουργίας annotation σε εικόνες και βίντεο. Στη συγκεκριμένη εργασία η δημιουργία annotation έγινε σε εικόνες και ο τύπος αρχείου που επιλέχθηκε είναι ο τύπος json.

Αρχικά έγινε η επιλογή του πλήθους των δεδομένων εκπαίδευσης του νευρωνικού δικτύου για τα οποία δημιουργήθηκαν annotations. Το σύνολο των διαθέσιμων δεδομένων χωρίστηκε σε δεδομένα εκπαίδευσης (80%) και ελέγχου (validation)(20%). Οι εικόνες που χρησιμοποιήθηκαν ως δεδομένα εξήχθησαν από ένα βίντεο με διάρκεια 2 λεπτά και 8 δευτερόλεπτα και ρυθμό καταγραφής 30 εικόνες ανά δευτερόλεπτο και έτσι προέκυψαν 3847 εικόνες. Αυτό το πλήθος εικόνων δεν είναι όλο αξιοποιήσιμο, διότι στο βίντεο που αναφέρθηκε παραπάνω η πόρτα κινείται αργά πάνω στον ταινιόδρομο, με αποτέλεσμα οι διαφορές μεταξύ διαδοχικών εικόνων είναι ελάχιστες. Γι' αυτό ορίστηκε ένα βήμα ανά 15 εικόνες με το οποίο επιλέχθηκε το σύνολο των εικόνων που θα αποτελέσουν τα δεδομένα εκπαίδευσης και ελέγχου και θα δημιουργηθούν annotations. Επομένως από τις 3847 εικόνες επιλέχθηκαν οι 257. Από αυτές οι 205 αποτελούν τα δεδομένα εκπαίδευσης και οι 52 τα δεδομένα ελέγχου.

Η διαδικασία δημιουργίας annotation δεν διαφοροποιείται για τα δεδομένα εκπαίδευσης και τα δεδομένα ελέγχου και είναι χειροκίνητη. Για κάθε μια εικόνα σχεδιάζεται, στο περιβάλλον του εργαλείου V.I.A., το περίγραμμα του αντικειμένου το οποίο πρέπει να εκπαιδευτεί να αναγνωρίζει ο αλγόριθμος μετέπειτα, δηλαδή την πόρτα του αυτοκινήτου. Η σχεδίαση γίνεται με την τοποθέτηση των κόμβων μιας κλειστής πολυγωνικής γραμμής (polyline) στις κατάλληλες θέσεις ώστε να περιγράφεται ορθά η πόρτα του αυτοκινήτου και να δημιουργείται μία μάσκα που να την καλύπτει πλήρως. Το κενό της πόρτας του αυτοκινήτου αποτέλεσε μια δυσκολία καθώς ό,τι φαίνεται μέσα από αυτό δεν αποτελεί μέρος της πόρτας του αυτοκινήτου και δεν πρέπει να περιλαμβάνεται στην μάσκα που δημιουργείται. Το περιβάλλον V.I.A. δεν επιτρέπει σχεδίαση μασκών με τρύπες, δηλαδή κενά εσωτερικά του περιγράμματος της μάσκας που δεν αποτελούν τμήμα της μάσκας. Η λύση που εφαρμόστηκε ήταν η σχεδίαση μασκών σχήματος "C" με τα δύο άκρα να ακουμπούν το ένα στο άλλο και να δημιουργούν το κενό εσωτερικά της μάσκας, όπως φαίνεται στο παρακάτω σχήμα 13.

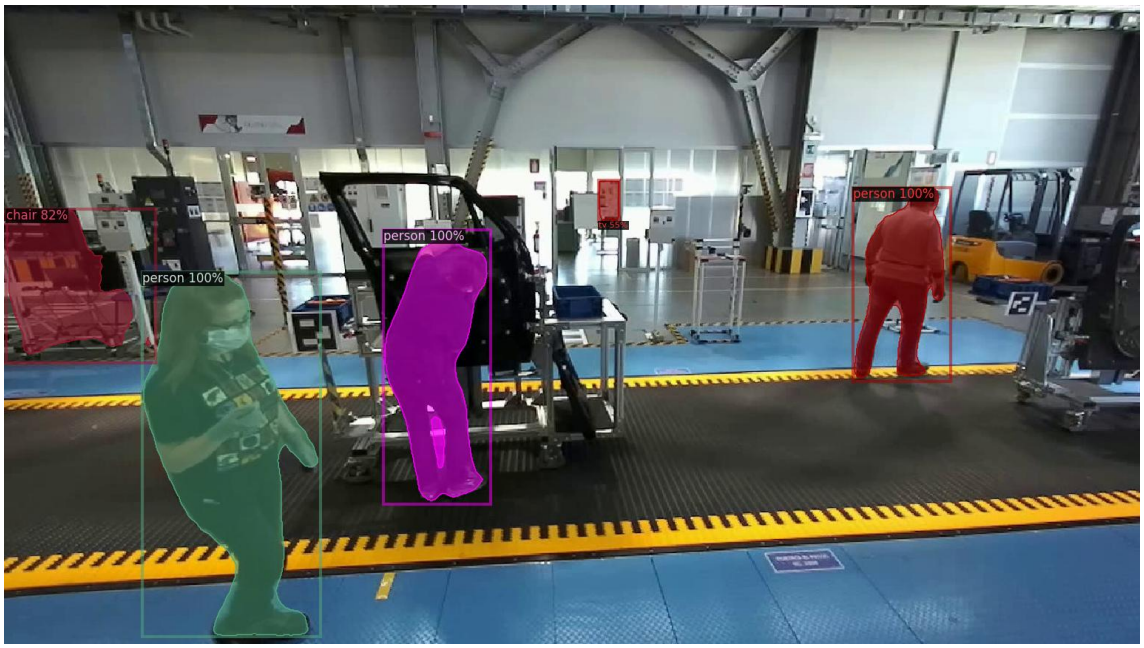
Σχήμα 13: Μάσκα σχήματος ”C” για την πόρτα αυτοκινήτου

Ένα δεύτερο ζήτημα που προέκυψε κατά τη διαδικασία δημιουργίας annotation ήταν η αλληλεπίδραση των ανθρώπων που έχουν καταγραφεί στο βίντεο με την πόρτα του αυτοκινήτου. Οι άνθρωποι καθώς συναρμολογούν τα εξαρτήματα της πόρτας δημιουργούν αποκρύψεις όταν τα χέρια ή το σώμα τους βρίσκονται μπροστά από την πόρτα σε σχέση με τη θέση της κάμερας. Προκειμένου να διαπιστωθεί πως πρέπει να αντιμετωπιστεί η αλληλεπίδραση ανθρώπων και πορτών για τη διαδικασία δημιουργίας annotation, έγιναν οι εξής δύο δοκιμές. Δημιουργήθηκαν annotation που περιλαμβάνουν τα τμήματα ανθρώπινου σώματος που αποκρύπτουν την πόρτα αυτοκινήτου, σαν να μην υπήρχαν, και annotation που αφήνουν εκτός τα τμήματα των ανθρώπινων σωμάτων που δημιουργούν αποκρύψεις. Στο επόμενο στάδιο, που είναι η εκπαίδευση του νευρωνικού δικτύου, θα διαπιστωθεί ποιο από τα δύο είδη annotation εξυπηρετεί καλύτερα τον σκοπό της εργασίας.

Όπως έχει ήδη διαπιστωθεί η αλληλεπίδραση των ανθρώπων με την πόρτα αυτοκινήτου είναι πολύ σημαντικό χαρακτηριστικό των δεδομένων της εργασίας. Για το λόγο αυτό αποφασίστηκε να δημιουργηθούν annotation και για τους ανθρώπους με σκοπό να χρησιμοποιηθούν κι αυτά στην εκπαίδευση του νευρωνκού δικτύου, ώστε το τελικό μοντέλο να είναι σε θέση να εντοπίζει και τους ανθρώπους στην σκηνή. Οι μάσκες για τους ανθωπους αυτή τη φορά δημιουργήθηκαν με έναν πιο αυτόματο τρόπο, αξιοποιώντας ένα ήδη εκπαιδευμένο μοντέλο, το Mask R-CNN R50 FPN. Το έτοιμο μοντέλο επιλέχθηκε από το αποθετήριο μοντέλων (model zoo) της πλατφόρμας Detectron2 και είναι βασισμένο στον αλγόριθμο του Mask R-CNN με βασικό του πλεονέκτημα τη δημιουργία αναλυτικών μασκών κατάτμησης (segmentation masks). Επιπλέον, χρησιμοποιεί ως συνδυασμό

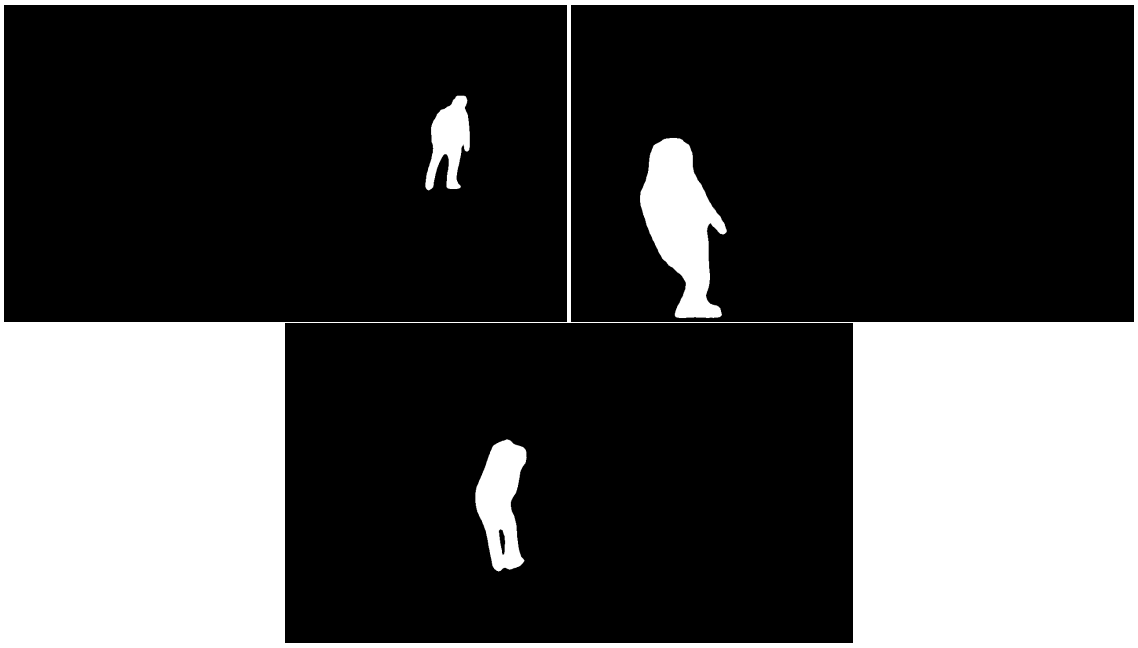
backbone και baseline το R50 FPN [27] που του επιτρέπει να κάνει πρόβλεψη γρηγορότερα, θυσιάζοντας λίγο την ακρίβεια (βλ. κεφάλαιο 3.3 για πιο αναλυτική περιγραφή του μοντέλου). Το συγκεκριμένο μοντέλο κλήθηκε να εντοπίσει τους ανθρώπους στα δεδομένα εκπαίδευσης και ελέγχου, πράγμα που έκανε με επιτυχία, καθώς κατάφερε να τους εντοπίσει σε όλες τις εικόνες που του δόθηκαν και μάλιστα με πολύ καλή λεπτομέρεια. Οι μάσκες κατάτμησης που προέκυψαν μετά την πρόβλεψη ουσιαστικά έδωσαν έτοιμες τις μάσκες που αφορούσαν τους ανθρώπους. Επόμενο βήμα είναι η διαμόρφωσή τους με κατάλληλο τρόπο ώστε να είναι συμβατές με το αρχείο json που περιέχει τα annotation που δημιουργήθηκαν με το εργαλείο V.I.A. και να μπορούν στη συνέχεια να ενωθούν σε ένα ενιαίο αρχείο annotation.

Για να δημιουργηθούν αρχεία annotation συμβατά με τα annotation που έχουν ήδη δημιουργηθεί χειροκίνητα για τις πόρτες, οι μάσκες κατάτμησης που προέκυψαν από την πρόβλεψη του μοντέλου Mask R-CNN R50 FPN χρειάστηκε να υποστούν κάποια επεξεργασία. Στο σχήμα 14 παρουσιάζεται ένα παράδειγμα των κατηγοριών που ανιχνεύτηκαν από το Mask R-CNN R50 FPN σε μία εικόνα.



Σχήμα 14: Αποτελέσματα εντοπισμού ανθρώπων από Mask R-CNN R50 FPN, εικόνα 3270

Η κλάση «άνθρωπος» (”person”) που προκύπτει απ’ το Mask R-CNN R50 FPN αποτελείται από το όνομα της κλάσης («άνθρωπος»), το πλαίσιο οριοθέτησης (bounding box) του ανθρώπου και τη μάσκα κατάτμησης (segmentation mask) για κάθε εικόνα (καρέ του βίντεο) που δώθηκε. Σημειώνεται ότι ο κάθε άνθρωπος που ενυπάρχει σε μια εικόνα αποτελεί ένα instance, επομένως αν σε μια εικόνα υπάρχουν 3 άνθρωποι για την κλάση «άνθρωπος» θα προκύψουν 3 instances (σχήμα 15). Η μάσκα κατάτμησης είναι ένας δυαδικός πίνακας στο μέγεθος της εικόνας από τον οποίο πρέπει να εξαχθεί το περίγραμμα της μάσκας, σαν πολυγωνική γραμμή, που αφορά τον άνθρωπο και αυτό να εγγραφεί στη συνέχεια σε ένα αρχείο json. Πρώτο βήμα είναι το άθροισμα όλων τωνinstances για την κλάση «άνθρωπος». Δηλαδή το άθροισμα όλων των δυαδικών μασκών, που αφορούν τους ανθρώπους, για κάθε εικόνα σε μία δυαδική εικόνα (σχήμα 16).



Σχήμα 15: Δυαδικές μάσκες για ανθρώπους, εικόνα 3270

Σχήμα 16: Άθροισμα δυαδικών μασκών για μία εικόνα (εικόνα 3270)

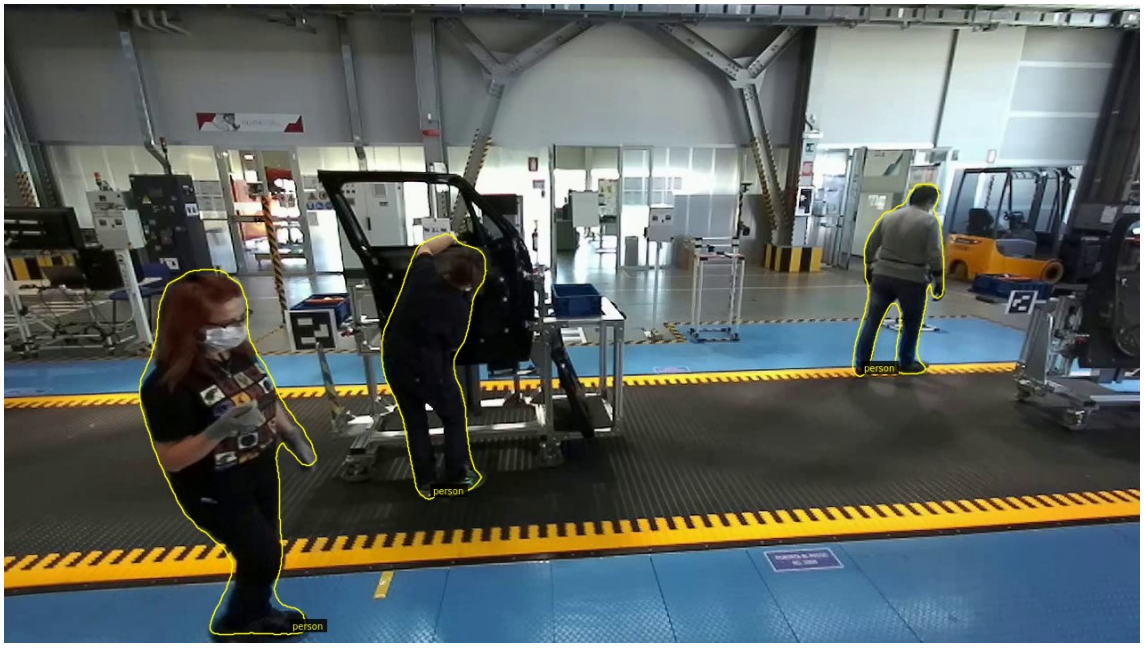
Η εξαγωγή των περιγραμμάτων - όπως και όλη η διαδικασία δημιουργίας των αρχείων json που περιέχουν τα annotation για τους ανθρώπους - έγινε με κώδικα σε γλώσσα προγραμματισμού python. Αρχικά χρησιμοποιήθηκε η συνάρτηση ”find_contours” που εξάγει περιγράμματα από εικόνες, ώστε να εξάγει την κλειστή πολυγωνική γραμμή που περιγράφει κάθε άνθρωπο από τη δυαδική εικόνα. Στο σχήμα 17 παρουσιάζονται τα περιγράμματα των ανθρώπων που έχουν εξαχθεί από τις δυαδικές μάσκες για τους ανθρώπους.

Σχήμα 17: Περιγράμματα ανθρώπων στην εικόνα 3270

Στη συνέχεια όλα τα περιγράμματα για τους ανθρώπους εγγράφηκαν σε ένα αρχείο τύπου json συνοδευόμενα από το όνομα της κλάσης που περιγράφουν, δηλαδή τους ανθρώπους. Το αρχείο json συμπληρώθηκε και με όλα τα υπόλοιπα απαραίτητα στοιχεία που για κάθε εικόνα είναι:

- Το όνομα του αρχείου.
- Το μέγεθος του αρχείου.
- Το είδος της γεωμετρίας που περιγράφουν οι κορυφές της πολυγωνικής γραμμής, δηλαδή πολύγωνο.
- Τις κορυφές της πολυγωνικής γραμμής (πρώτα όλες τις συντεταγμένες στον οριζόντιο άξονα-x και μετά τις συντεταγμένες στον κατακόρυφο άξονα-y).
- Το όνομα της κλάσης που περιγράφεται, δηλαδή «άνθρωπος».

Στο σχήμα 18 παρουσιάζεται ένα παράδειγμα annotation που δημιουργήθηκε, με την παραπάνω διαδικασία, για την κλάση «άνθρωπος».



Σχήμα 18: Οπτικοποίηση δημιουργημένων annotation για ανθρώπους, εικόνα 3270

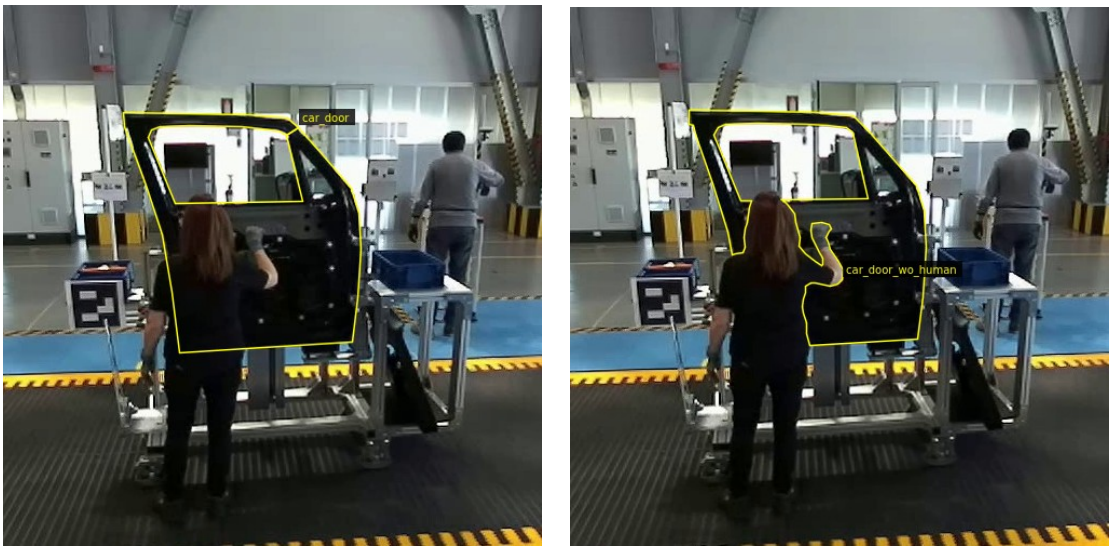
3.3 Εκπαίδευση συνελικτικού νευρωνικού δικτύου (CNN)

Έχοντας δημιουργήσει τα annotation για το σύνολο των εικόνων που θα χρησιμοποιηθούν ως δεδομένα εκπαίδευσης και ελέγχου, ακολουθεί η εκπαίδευση ενός συνελικτικού νευρωνικού δικτύου (CNN) ώστε να μπορεί να εντοπίζει την πόρτα του αυτοκινήτου στην σκηνή.

Η διαδικασία της εκπαίδευσης του συνελικτικού νευρωνικού δικτύου δεν ξεκίνησε από το μηδέν, αλλά από ένα ήδη εκπαιδευμένο μοντέλο, συγκεκριμένα το Mask R-CNN R50 FPN που χρησιμοποιήθηκε και στη δημιουργία των annotation για τους ανθρώπους. Το μοντέλο αυτό έγινε διαθέσιμο μέσω του αποθετηρίου μοντέλων (model zoo) του Detectron2, όπως αναφέρθηκε και παραπάνω. Το γεγονός ότι το Mask R-CNN R50 FPN χρησιμοποιεί τον αλγόριθμο του Mask R-CNN με αποτέλεσμα να δημιουργεί μάσκες κατάτμησης (segmentation masks) το καθιστά ιδιαιτέρως ταιριαστό στις απαιτήσεις της εργασίας, διότι μετά την εκπαίδευση θα μπορεί να αναγνωρίζει περιοχές τις εικόνας που αντιστοιχούν σε πόρτα αυτοκινήτου. Η επιλογή ενός ήδη εκπαιδευμένου μοντέλου, επίσης, απλοποιεί και διευκολύνει σημαντικά την διαδικασία, μειώνει το απαιτούμενο υπολογιστικό κόστος αλλά και τον χρόνο που χρειάζεται για τη δημιουργία ενός νέου μοντέλο από την αρχή. Το μοντέλο Mask R-CNN R50 FPN έχει εκπαιδευτεί στο σύνολο δεδομένων του COCO dataset που αποτελείται από 328.000 εικόνες και μετά την εκπαίδευσή του το μοντέλο είναι σε θέση να αναγνωρίζει 91 κλάσεις που αντιστοιχούν σε καθημερινά αντικείμενα που μπορεί να αναγνωρίζει ένα παιδί 4 ετών [24] όπως άνθρωπος, καρέκλα, τηλεόραση, ψυγείο και άλλα. Επιπλέον. το μοντέλο χρησιμοποιεί ως συνδυασμό backbone και baseline το R50 FPN [27], το οποίο για τη συγκεκριμένη εργασία έχει κάποια αξιόλογα πλεονεκτήματα. Το backbone είναι υπεύθυνο για την εξαγωγή των χαρακτιριστικών από τα δεδομένα, ενώ το baseline είναι ένα απλό μοντέλο που χρησιμοποιείται ως βάση για αξιολόγηση και σύγκριση με άλλες τεχνικές [28]. Το R50 FPN που χρησιμοποιεί το μοντέλο Mask R-CNN R50 FPN, του επιτρέπει να εκπαιδεύεται και να κάνει πρόβλεψη σε μικρότερο χρόνο και με μικρότερη απαίτηση σε μνήμη για την εκπαίδευση σε σχέση με άλλους συνδυασμούς backbone και baseline, άρα διευκολύνει τη διαδικασία των πειραμάτων. Μειονέκτημα είναι η χαμηλότερη ακρίβεια όπου για την συγκεκριμένη εργασία γίνεται αποδεκτή.

Η επανεκπαίδευση του συνελικτικού νευρωνικού δικτύου διατηρεί την ήδη αποκτημένη γνώση από την αρχική εκπαίδευση και προσθέτει σε αυτήν τις κλάσεις για τις οποίες έχουν δωθεί annotation. Σημαντικό είναι να σημειωθεί πως το τελικό μοντέλο μπορεί να αναγνωρίσει και κατά συνέπεια να δώσει μάσκες κατάτμησης μόνο για τις νέες κλάσεις που έχουν προστεθεί με την επανεκπαίδευση. Όπως έχει αναφερθεί και προηγουμένως, ζητούμενο είναι ο εντοπισμός και η κατάτμηση των πορτών αυτοκινήτου αλλά και των ανθρώπων που βρίσκονται στην σκηνή και αλληλεπιδρουν με τις πόρτες (συναρμολογούν εξαρτήματα). Επομένως, δεδομένου ότι δημιουργήθηκαν annotation για πόρτες αυτοκινήτου και ανθρώπους, αυτές θα είναι και οι βασικές κλάσεις που θα αναγνωρίζει και θα εντοπίζει το συνελικτικό νευρωνικό μοντέλο μετά την επανεκπαίδευση.

Πρώτο στάδιο της εκπαίδευσης είναι η επανεκπαίδευση του μοντέλου για εντοπισμό μόνο των πορτών αυτοκινήτου σε δύο περιπτώσεις. Στην πρώτη χρησιμοποιώντας τα annotation που περιλαμβάνουν τμήματα των ανθρώπινων σωμάτων που καλύπτουν την πόρτα ως κλάση ”πόρτα αυτοκινήτου” (car door), (σχήμα 19α). Στη δεύτερη χρησιμοποιώντας τα annotation που εξαιρούν τα τμήματα των ανθρώπινων σωμάτων, που δημιουργούν αποκρύψεις, από την κλάση ”πόρτα αυτοκινήτου” (car door), (σχήμα 19β).



(α□) Πόρτα με επικάλυψη ανθρώπου (β□) Πόρτα χωρίς επικάλυψη ανθρώπου

Σχήμα 19: Περιπτώσεις ανίχνευσης πορτών αυτοκινήτου

Οι δύο αυτές περιπτώσεις εξετάστηκαν προκειμένου να επιλεγεί μία απ’ τις δύο ως ο τρόπος που θα περιγράφεται η κλάση ”πόρτα αυτοκινήτου” (car door). Όπως έχει ήδη γίνει αντιληπτό, χρησιμοποιήθηκαν δύο διαφορετικά σετ annotation, όμως και στις δύο περιπτώσεις η επανεκπαίδευση του μοντέλου Mask R-CNN R50 FPN σε 205 εικόνες διήρκησε περίπου 4 λεπτά, χρησιμοποιώντας τυπικές τιμές στις παραμέτρους εκπαίδευσης. Στη συνέχεια παρουσιάζονται οι παράμετροι εκπαίδευσης καθώς και οι τιμές που χρησιμοποιήθηκαν για κάθε μία απ’ αυτές.

1. cfg.DATASETS.TRAIN: Δίνεται η λίστα με τα ονόματα των συνόλων εικόνων που θα χρησιμοποιηθούν στην εκπαίδευση του δικτύου. Δώθηκε ένα όνομα (”car_door_train”) διότι υπήρχε ένα σύνολο εικόνων.

2. cfg.DATASETS.TEST: Αντίστοιχα δίνεται η λίστα με τα ονόματα των νέων συνόλων εικόνων στα οποία θα γίνει δοκιμή του μοντέλου. Δεν αφορά την εκπαίδευση οπότε δεν χρησιμοποιήθηκε.

3. cfg.DATALOADER.NUM_WORKERS: Αφορά το πλήθος των threads που φορτώνουν τα δεδομένα. Χρησιμοποιήθηκε η τιμή 2.

4. cfg.MODEL.WEIGHTS: Δίνεται η θέση του μοντέλου (path) από το οποίο θα ξεκινήσει η εκπαίδευση, όπως για παράδειγμα το model zoo του Detectron2. Και στις δύο περιπτώσεις δώθηκε το ήδη εκπαιδευμένο μοντέλο Mask R-CNN R50 FPN από το model zoo.

5. cfg.SOLVER.IMS_PER_BATCH: Είναι το πλήθος των εικόνων που φορτώνονται κάθε φορά στη μνήμη για να γίνει η εκπαίδευση. Χρησιμοποιήθηκε η τιμή 2, δηλαδή κάθε φορά να φορτώνονται δύο εικόνες.

6. cfg.SOLVER.BASE_LR: Πρόκειται για το ρυθμό με τον οποίο «μαθαίνει» το δίκτυο (learning rate). Ορίστηκε η τιμή 0,00025 που είναι μια καλή τιμή για τέτοιες εφαρμογές.

7. cfg.SOLVER.MAX_ITER: Είναι το πλήθος των επαναλήψεων, όπου μία επανάληψη είναι το ένα βήμα της εκπαίδευσης στο οποίο ορίζεται το πλήθως τον εικόνων που θα περιλαμβάνει με την παράμετρο cfg.SOLVER.IMS_PER_BATCH. Και στις δύο περιπτώσεις ορίστηκε η τιμή 500.

8. cfg.SOLVER.STEPS: Είναι η επανάληψη στην οποία θα μειωθεί ο ρυθμός εκμάθησης (learning rate) κατά μια τιμή. Στη συγκεκριμένη εφαρμογή δεν εφαρμόστηκε κάτι τέτοιο.

9. cfg.MODEL.ROI_HEADS.BATCH_SIZE_PER_IMAGE: Πρόκειται για το πλήθος των περιοχών ενδιαφέροντος (Regions Of Interest) που χρησιμοποιούνται κατά την εκπαίδευση, ανά εικόνα. Η τυπική τιμή είναι 512 οπότε και αυτή χρησιμοποιήθηκε.

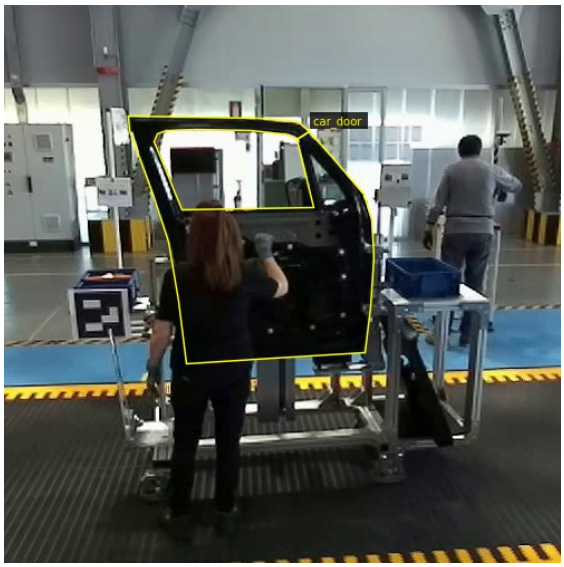
10. cfg.MODEL.ROI_HEADS.NUM_CLASSES: Αντιστοιχεί στο πλήθος των κλάσεων του προσκηνίου, δηλαδή το πλήθος τον κλάσεων που καλείται να μάθει να αναγνωρίζει το δίκτυο. Και στις δύο περιπτώσεις είναι μία: η πόρτα με την επικάλυψη από ανθρώπους και η πόρτα χωρίς επικαλύψεις.

11. cfg.MODEL.ROI_HEADS.SCORE_THRESH_TEST: Χρησιμοποιείται μόνο όταν γίνεται δοκιμή σε ένα καινούριο, άγνωστο σύνολο εικόνων και πρόκειται για μια τιμή, στο εύρος 0 έως 1, που εξισορροπεί την υψηλή ευαισθησία (recall) χωρίς να υπάρχουν ανιχνεύσεις με χαμηλή ακρίβεια (precision) όπου θα καθυστερεί την πρόβλεψη σε επόμενο βήμα.

Τελικά, επιλέχθηκε η πρώτη περίπτωση, όπου οι μάσκες για τις πόρτες περιλαμβάνουν και τυχόν επικαλύψεις από ανθρώπους, καθώς οι διαφορές (βλ. κεφάλαιο 4.2 με αποτελέσματα της εκπαίδευσης του νευρωνικού δικτύου) στον τρόπο που το νευρωνικό δίκτυο χειρίστηκε την επικάλυψη (εξαίρεση ή μη) του ανθρώπου με την πόρτα στις δύο περιπτώσεις δεν ήταν αξιόλογες. Επιπλέον, η πρώτη περίπτωση, κατά την εκπαίδευση δίνει στο δίκτυο το περίγραμμα ολόκληρης της πόρτας, διατηρώντας το σχήμα της σε

γενικές γραμμές αναλοίωτο, γεγονός που καθιστά τα δεδομένα εκπαίδευσης πιο αντιπροσωπευτικά ως προς το σχήμα των πορτών αυτοκινήτου γενικότερα. Επιπλέον, η μάσκα ολόκληρης της πόρτας ως δεδομένο εκπαίδευσης συμβαδίζει με την απαίτηση που έχει τεθεί για τον προσδιορισμό του κέντρου της πόρτας από τις μάσκες κατάτμησης που προβλέπει το τελικό μοντέλο.

Το επόμενο στάδιο της εκπαίδευσης του νευρωνικού δικτύου περιλαμβάνει και την ένταξη των ανθρώπων στα δεδομένα εκπαίδευσης. Το πλήθος των εικόνων που χρησιμοποιήθηκαν για την εκπαίδευση είναι και πάλι το ίδιο αλλά για κάθε εικόνα υπάρχουν δύο κλάσεις annotation, αυτές για την πόρτα αυτοκινήτου και για τους ανθρώπους (σχήμα 20).



(α□) Παράδειγμα μάσκας για πόρτα αυτοκινήτου, εικόνα 2205

(β□) Παραδείγματα μασκών για ανθρώπους, εικόνα 2205

Σχήμα 20: Παράδειγμα μασκών που χρησιμοποιήθηκαν στα δεδομένα εκπαίδευσης για την ανίχνευση πορτών αυτοκινήτου και ανθρώπων, εικόνα 2205

Με τις ίδιες τυπικές τιμές των παραμέτρων εκπαίδευσης, η επανεκπαίδευση του μοντέλου Mask R-CNN R50 FPN στις ίδιες 205 εικόνες διήρκησε περίπου 4 λεπτά. Στη συνέχεια τροποποιήθηκαν οι παράμετροι εκπαίδευσης, προκειμένου το αποτέλεσμα να είναι καλύτερο, και ο τελικός χρόνος επανεκπαίδευσης του μοντέλου Mask R-CNN R50 FPN έφτασε τα 9 λεπτά περίπου.

Η τελική μορφή των παραμέτρων εκπαίδευσης επιλέχθηκε με γνώμονα την συνολική απώλεια total loss της εκπαίδευσης. Η συνολική απώλεια είναι επιθυμητό να φθίνει απότομα και να σταθεροποιείται σύντομα γύρω από την τιμή 0. Αυτό επιτυγχάνεται κυρίως με την αύξηση των επαναλήψεων. Είναι σημαντικό οι επαναλήψεις που έχουν επιλεγεί να είναι ακέραιο πολλαπλάσιο του πλήθους των δεδομένων εκπαίδευσης ώστε να έχουν όλες οι εικόνες το ίδιο βάρος στην εκπαίδευση. Το πέρασμα του μοντέλου από όλες τις εικόνες των δεδομένων εκπαίδευσης μία φορά είναι ένα epoch. Επιλέχθηκε το πλήθος των 10 epochs ως ένα πλήθος που δίνει καλά αποτελέσματα και αυτό αντιστοιχεί σε 1025 επαναλήψεις.

Επιπλέον, τροποποιήθηκε η παράμετρος cfg.MODEL.ROI_HEADS.BATCH_SIZE_PER_IMAGE από την τιμή 512 σε 128, ώστε να εκτελείται γρηγορότερα η εκπαίδευση. Έτσι, κατά την εκπαίδευση χρησιμοποιούνται 128 περιοχές ενδιαφέροντος (Regions Of Interest) ανά εικόνα και η εκπαίδευση προχωράει πιο γρήγορα. Τέλος, δεδομένου ότι το δίκτυο εκπαιδεύεται για να ανιχνεύει δύο κλάσεις (πόρτες αυτοκινήτου, άνθρωπος), στην παράμετρο cfg.MODEL.ROI_HEADS.NUM_CLASSES τέθηκε η τιμή 2. Οι υπόλοιπες παράμετροι διατηρήθηκαν ως είχαν διότι η μεταβολή τους είτε δεν επέφερε αξιοσημείωτη μεταβολή στο αποτέλεσμα, είτε δυσκόλευε την διαδικασία διότι οι πόροι του συστήματος που χρησιμοποιήθηκε ήταν ανεπαρκείς. Η τελική μορφή των παραμέτρων εκπαίδευσης παρουσιάζεται συγκεντρωτικά στον παρακάτω πίνακα.

Παράμετρος	**Τιμή**
cfg.DATALOADER.NUM_WORKERS	2
cfg.SOLVER.IMS_PER_BATCH	2
cfg.SOLVER.BASE_LR	0,00025
cfg.SOLVER.MAX_ITER	1025
cfg.MODEL.ROI_HEADS.BATCH_SIZE_PER_IMAGE	128
cfg.MODEL.ROI_HEADS.NUM_CLASSES	2

Πίνακας 3: Τελικές τιμές παραμέτρων εκπαίδευσης

3.4 Μετρικές για την ποσοτική αξιολόγηση

Η ποσοτική αξιολόγηση των πειραμάτων είναι πολύ σημαντική διότι περιγράφει με μετρήσιμο τρόπο την ποιότητα των αποτελεσμάτων. Έχοντας μια μετρήσιμη τιμή που περιγράφει το αποτέλεσμα ενός πειράματος μπορεί εύκολα να αποφασιστεί αν το συγκεκριμένο αποτέλεσμα ικανοποιεί τις απαιτήσεις ή χρειάζεται να γίνουν κι άλλα πειράματα προκειμένου να επιτευχθεί κάτι καλύτερο. Στη συγκεκριμένη εργασία ποσοτική αξιολόγηση έγινε σε όλα τα στάδια των πειραμάτων ώστε να κριθεί η αποτελεσματικότητα, η αξιοπιστία και η ευαισθησία των μεθόδων που εφαρμόστηκαν.

Πρώτο στάδιο της ποσοτικής αξιολόγησης είναι ο προσδιορισμός των παρακάτω μεγεθών τα οποία στο πλαίσιο της συγκεκριμένης εργασίας, όπου πρόκειται για πειράματα εντοπισμού αντικειμένων σε εικόνες, ερμηνεύονται ως εξής:

- True positives: πλήθος εντοπισμών που είναι όντως ορθοί.
- True negatives: πλήθος μη εντοπισμών που όντως είναι ορθοί.
- False positives: πλήθος εντοπισμών που δεν είναι πραγματικά ορθοί.
- False negatives: Πλήθος μη εντοπισμών που όμως δεν είναι ορθοί.

Στη συνέχεια και με βάση τα παραπάνω μεγέθη μπορούν να προσδιοριστούν η ορθότητα (accuracy), ακρίβεια (precision), ανάκληση/ευαισθησία (recall) και F1 score ως εξής.

Η **ορθότητα (accuracy)** περιγράφει το πόσο συχνά ο αλγόριθμος κάνει σωστή ανίχνευση και αποτελεί τον λόγο των ορθών ανιχνεύσεων προς τον αριθμό των συνολικών ανιχνεύσεων.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Η **ακρίβεια (precision)** περιγράφει πόσες από τις ανιχνευμένες περιπτώσεις είναι όντως ορθές. Χρησιμοποιεαι κυρίως όταν τα False positives προβληματίζουν περισσότερο από τα False negatives.

$$Precision = \frac{TP}{TP + FP}$$

Η **ανάκληση/ευαισθησία (recall)** περιγράφει πόσες από τις πραγματικά ορθές περιπτώσεις μπόρεσε να ανιχνεύσει ο αλγόριθμος.

$$Recall = \frac{TP}{TP + FN}$$

Το **F1 score** είναι ένα συνδυαστικό μέτρο της αξιοπιστίας και της ανάκλησης και μεγιστοποιείται όταν η ανάκληση και η αξιοπιστία είναι ίδιες.

$$F1score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Μια επιπλέον μετρική που χρησιμοποιείται στην αξιολόγηση εφαρμογών ανίχνευσης αντικειμένων είναι το **Intersection over Union, IoU** [30]. Πρόκειται για μια μετρική που συγκρίνει τα πλαίσια οριοθέτησης και τις μάσκες κατάτμησης που έχουν προκύψει από την ανίχνευση, με αυτά που έχουν δωθεί στα δεδομένα ελέγχου. Η σύγκριση γίνεται με τον υπολογισμό ενός λόγου, όπου αριθμητής είναι η περιοχή της τομής, δηλαδή η κοινή περιοχή των δύο σχημάτων, και παρονομαστής είναι η περιοχή της ένωσης, δηλαδή h περιοχή που καλύπτουν μαζί και τα δύο σχήματα.

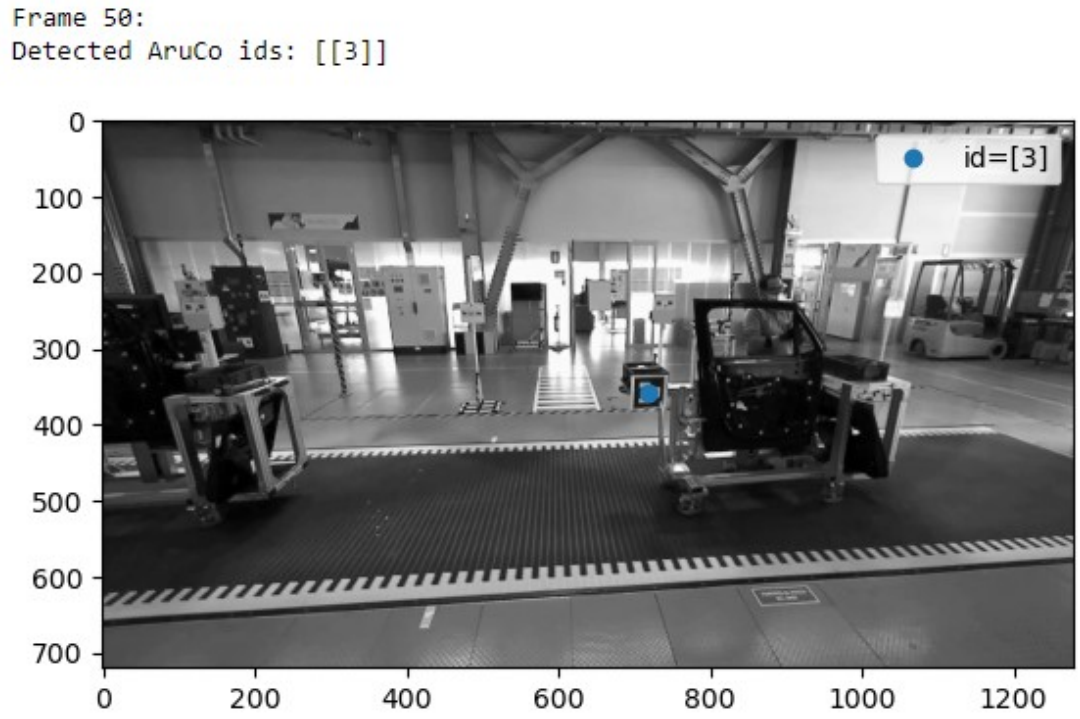
$$IoU = \frac{area\ of\ overlap}{area\ of\ union}$$

Η συγκεκριμένη μετρική αξιολογεί το κατά πόσο οι μάσκες που προκύπτουν απ' τον εντοπισμό και οι μάσκες που δίνονται στα δεδομένα ελέγχου ταυτίζονται. Αν η τιμή της μετρικής IoU είναι μεγαλύτερη από 0.7 θεωρείται μια καλή τιμή. Όπως είναι λογικό η τιμή IoU=1 είναι αδύνατη, διότι είναι αδύνατο η μάσκα που έχει προκύψει από τον εντοπισμό να ταυτίζεται πλήρως με την μάσκα που έχει δημιουργηθεί στα δεδομένα ελέγχου.

4 Αποτελέσματα και αξιολόγηση

4.1 Αποτελέσματα εντοπισμού στόχων ArUco

Η αξιολόγηση των αποτελεσμάτων της ανίχνευσης των στόχων ArUco έγινε ποιοτικά και ποσοτικά. Η ποιοτική αξιολόγηση έγινε ελέγχοντας αν τα ανιχνευμένα ArUco εντοπίζονται στις σωστές θέσεις μέσα στην εικόνα όπου και ανιχνεύθηκαν, γι' αυτό και από τον αλγόριθμο που δημιουργήθηκε εκτυπώθηκαν οι ανιχνευμένες θέσεις ArUco πάνω σε κάθε εικόνα του βίντεο που έγινε ανίχνευση για έλεγχο. Αυτός ο έλεγχος έγινε δειγματοληπτικά και σε πρώτη φάση προκειμένου να διαπιστωθεί ότι δεν υπάρχει κάποιο χονδροειδές σφάλμα στη διαδικασία.



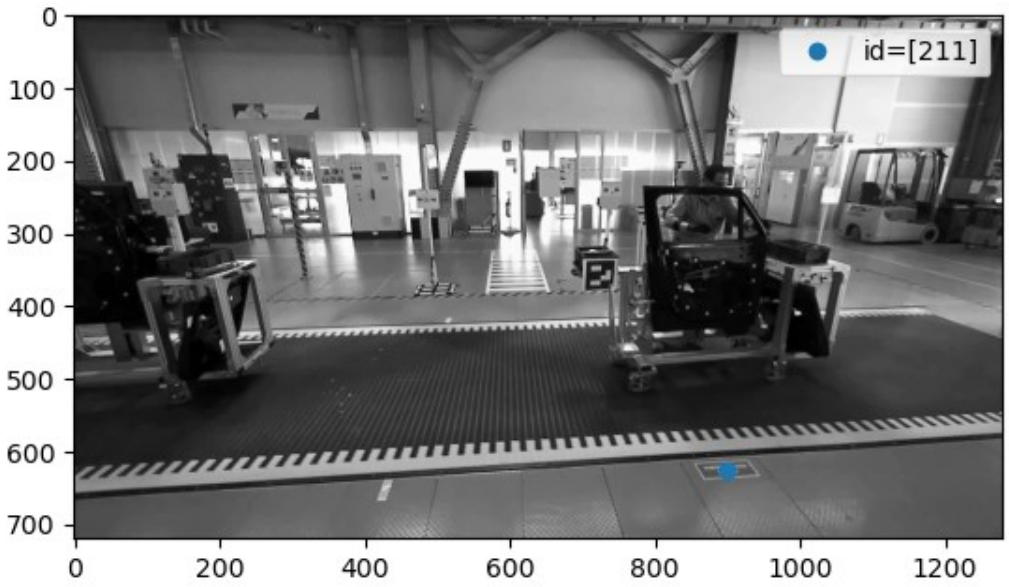
Σχήμα 21: Παράδειγμα επιτυχημένου εντοπισμού στόχου ArUco

Στο παραπάνω σχήμα 21 απεικονίζεται ένα παράδειγμα επιτυχημένου εντοπισμού. Όπως φαίνεται και στην οπτικοποίηση, η απεικόνιση της θέσης του εντοπισμένου στόχου ArUco ταυτίζεται με τη θέση του στόχου στην εικόνα. Επιπλέον έχει εντοπιστεί ο σωστός στόχος ArUco (με id 3), ενώ ορθά δεν έχει εντοπιστεί κάποιος άλλος στόχος ArUco.

Στη συνέχεια παρουσιάζονται ορισμένα παραδείγματα λανθασμένων εντοπισμών όπου είτε έχουν ανιχνευτεί στόχοι ArUco σε θέσεις όπου δεν υπάρχουν πραγματικά (σχήμα 22) και πρόκειται για άλλα αντικείμενα ή έχουν εντοπιστεί παραπάνω από ένας στόχος (σχήμα 23, 24), πράγμα αδύνατο αφού σε κάθε εικόνα του βίντεο ενυπάρχει ένας στόχος ArUco, αυτός με id 3. Επιπλέον, μια άλλη περίπτωση λάθος εντοπισμού είναι ο μη εντοπι-

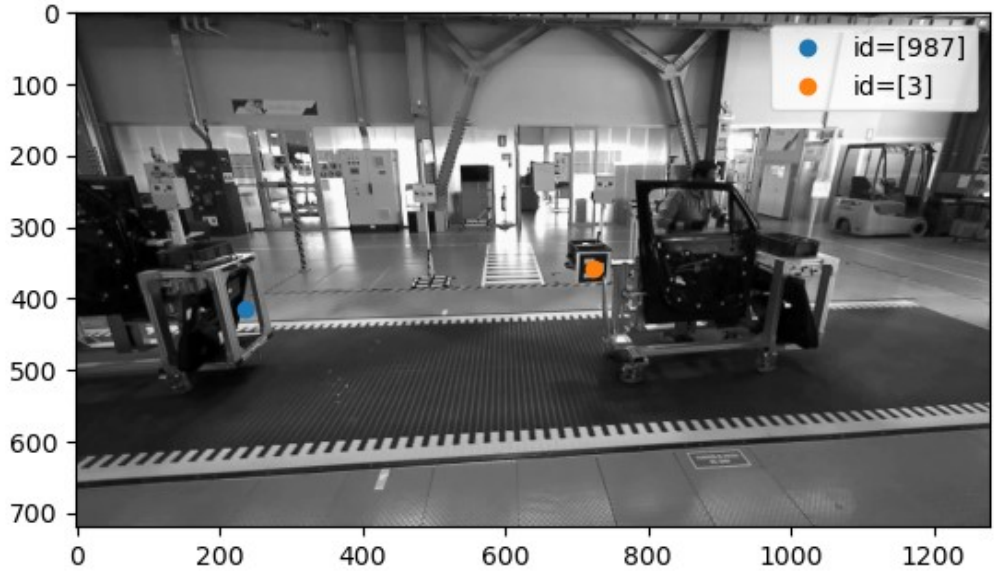
σμός κανενός στόχου, παρόλο που όπως αναφέρθηκε και προηγουμένως σε κάθε εικόνα του βίντεο είναι ορατός ένας στόχος ArUco.

```
Frame 4:
Detected AruCo ids: [[211]]
```

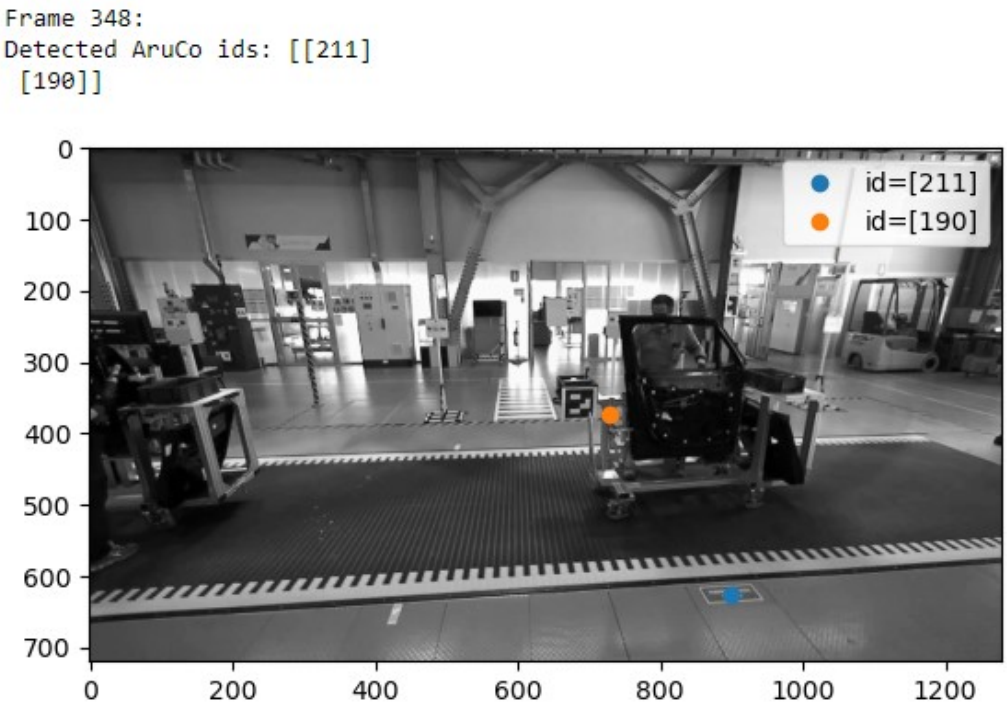


Σχήμα 22: Λανθασμένος εντοπισμός ArUco, περίπτωση 1

```
Frame 13:
Detected AruCo ids: [[987]
 [  3]]
```



Σχήμα 23: Λανθασμένος εντοπισμός ArUco, περίπτωση 2



Σχήμα 24: Λανθασμένος εντοπισμός ArUco, περίπτωση 3

Ο ποσοτικός έλεγχος της διαδικασίας ανίχνευσης στόχων ArUco δίνει πιο αντιπροσωπευτική εικόνα σε σχέση με τον βαθμό επιτυχίας της μεθόδου που εφαρμόστηκε και αποτελεί μια πιο επιστημονικά ορθή μέθοδο αξιολόγησης. Για την ποσοτική αξιολόγηση της διαδικασίας ανίχνευσης στόχων ArUco υπολογίστηκαν τα μεγέθη: ακρίβεια (accuracy), αξιοπιστία (precision), ανάκληση/ευαισθησία (recall) και F1 score. Πριν υπολογιστούν τα μεγέθη που μόλις αναφέρθηκαν χρειάστηκε να προσδιοριστεί το πλήθος των True positives, True negatives, False positives, False negatives, όπως αναφέρθηκε και στο σχετικό κεφάλαιο 3.4 «Μετρικές για την ποσοτική αξιολόγηση».

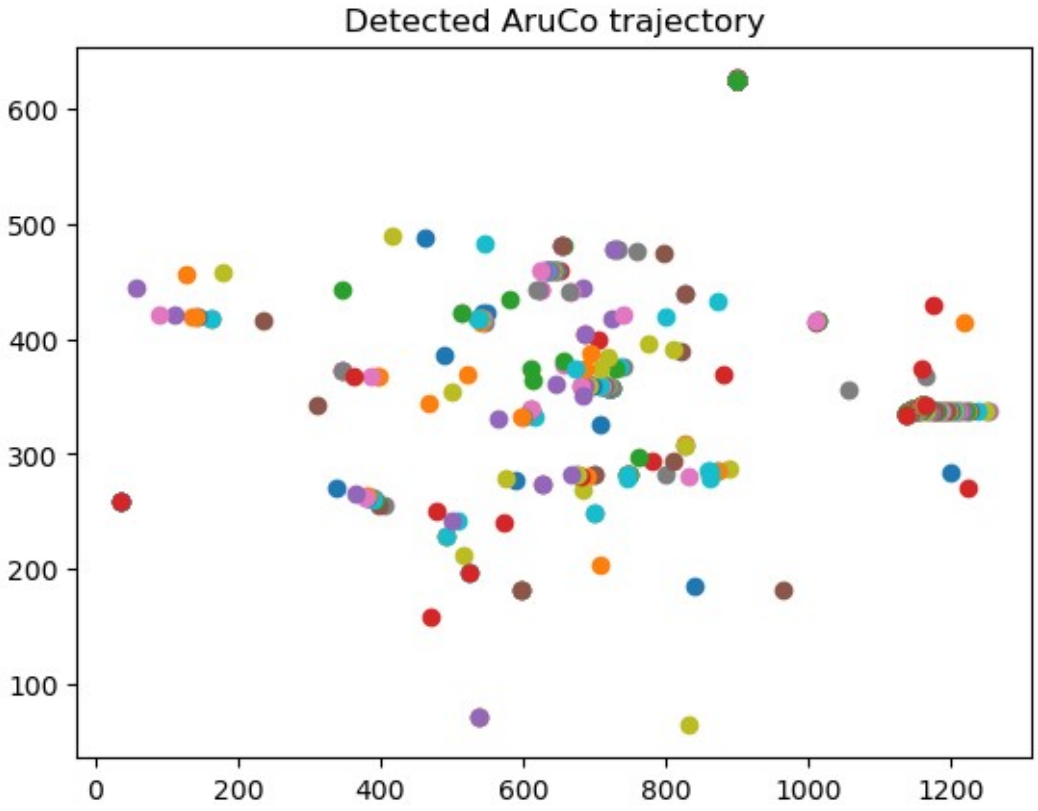
Στη συνέχεια παρουσιάζονται τα αποτελέσματα των πειραμάτων που διεξάχθηκαν προκειμένου να προσδιοριστούν οι βέλτιστες τιμές παραμέτρων για τα δεδομένα που διαθέτει η συγκεκριμένη εργασία. Όπως έχει ήδη αναφερθεί η δοκιμή Δ0 διαθέτει τις προκαθορισμένες τιμές παραμέτρων εντοπισμού και αποτελεί τη βάση για τις επόμενες δοκιμές. Οι επόμενες δοκιμές τροποποιούν κάποιες από τις παραμέτρους με στόχο την μεγιστοποίηση της ακρίβειας εντοπισμού στόχων ArUco.

Αποτελέσματα δοκιμής Δ0 - Μετρικές ποσοτικής αξιολόγησης:

- True positives: 661 (έχει εντοπιστεί ο σωστός στόχος ArUco)
- True negatives: 0 (δεν υπάρχουν εικόνες χωρίς στόχο ArUco)
- False positives: 943 (έχει εντοπιστεί λάθος στόχος)

- False negatives: 2480 (δεν έχει εντοπιστεί κανένας στόχος)
- Ορθότητα (Accuracy): 16,19%
- Ακρίβεια (Precision): 0,412
- Ευαισθησία (Recall): 0,210
- F1 score: 0,279

Τα αποτελέσματα της πρώτης δοκιμής δεν είναι καλά και αυτό γίνεται αισθητό ποιοτικά, παρατηρώντας την εκτύπωση των εντοπισμένων στόχων πάνω στις εικόνες, αλλά και ποσοτικά, από τις μετρικές που υπολογίστηκαν και παρουσιάστηκαν παραπάνω. Οι ορθοί εντοπισμοί (True positives) είναι πολύ λίγοι, οι λανθασμένοι εντοπισμοί (False positives) είναι πολλοί και οι περιπτώσεις όπου δεν έχει γίνει κάποιος εντοπισμός (False negatives) είναι η πλειοψηφία των αποτελεσμάτων.



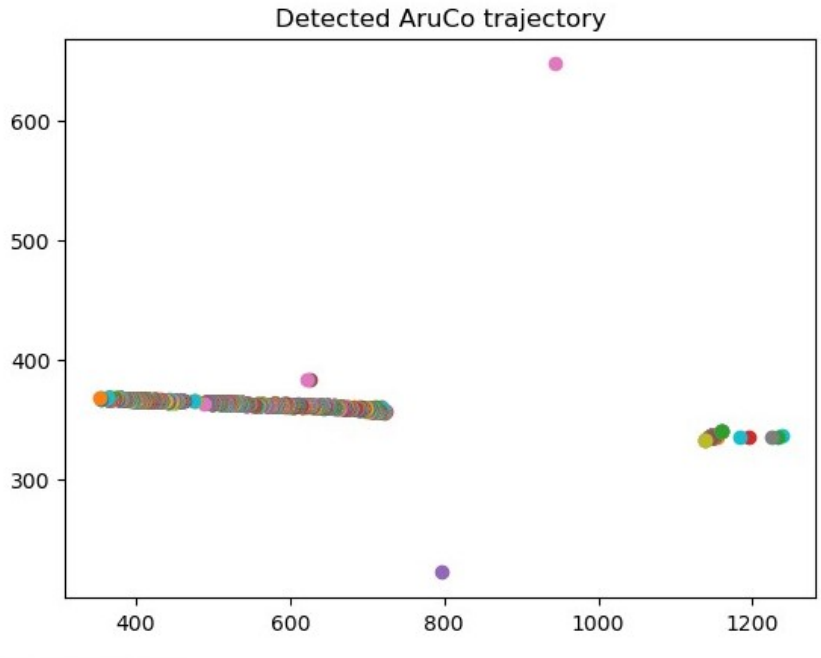
Σχήμα 25: Τροχιά εντοπισμένων ArUco, δοκιμή Δ0, άξονες X-Y: οριζόντια και κατακόρυφη διάσταση εικόνων (720,1280)

Στο παραπάνω σχήμα 25 παρουσιάζεται η τροχιά των στόχων ArUco που εντοπίστηκαν στη δοκιμή Δ0, χρησιμοποιώντας τις προκαθορισμένες τιμές παραμέτρων εντοπισμού. Συγκεκριμένα, παρουσιάζονται οι θέσεις των εντοπισμένων στόχων εντός των

ορίων της εικόνας (ενός καρέ του βίντεο με διαστάσεις 1280x720 εικονοστοιχεία). Όπως είναι προφανές τα αποτελέσματα του εντοπισμού δεν είναι καλά καθώς οι θέσεις των εντοπισμένων στόχων είναι διάσπαρτες μέσα στην εικόνα. Δεδομένου ότι η πόρτα έχει μια συγκεκριμένη τροχιά κατά τη διάρκεια του βίντεο - κινείται οριζόντια από τα δεξιά προς τα αριστερά χωρίς να αλλάζει θέση στον κατακόρυφο άξονα - αν οι εντοπισμοί ήταν επιτυχείς, θα σχημάτιζαν μια ευθεία περίπου οριζόντια και στο μέσο της εικόνας.

Αποτελέσματα δοκιμής Δ1 - Μετρικές ποσοτικής αξιολόγησης:

- True positives: 1459 (έχει εντοπιστεί ο σωστός στόχος ArUco)
- True negatives: 0 (δεν υπάρχουν εικόνες χωρίς στόχο ArUco)
- False positives: 4 (έχει εντοπιστεί λάθος στόχος)
- False negatives: 2428 (δεν έχει εντοπιστεί κανένας στόχος)
- Ορθότητα (Accuracy): 37,50%
- Ακρίβεια (Precision): 0,997
- Ευαισθησία (Recall): 0,375
- F1 score: 0,545

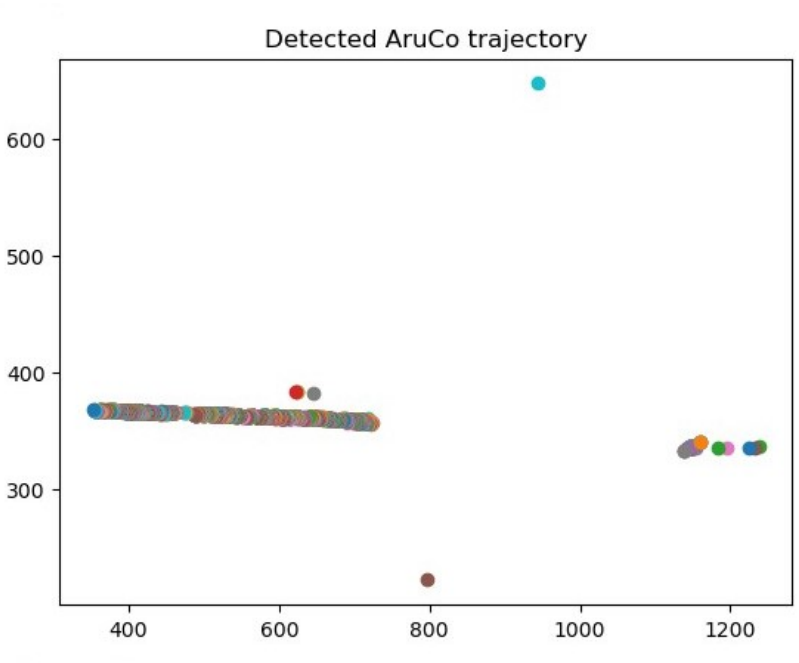


Σχήμα 26: Τροχιά εντοπισμένων ArUco, δοκιμή Δ1, άξονες X-Y: οριζόντια και κατακόρυφη διάσταση εικόνων (720,1280)

Όπως φαίνεται και από τις μετρικές και από το αντίστοιχο σχήμα 26 με την τροχιά των εντοπισμένων στόχων ArUco για τη δοκιμή Δ1, με την πρώτη τροποποίηση των παραμέτρων εντοπισμού τα αποτελέσματα του εντοπισμού παρουσιάζουν αισθητή βελτίωση. Σημαντικότερη βελτίωση παρουσιάζει το πλήθος των False positives, καθώς προέκυψαν μόνο 4 περιπτώσεις, γεγονός που αυξάνει σημαντική την ακρίβεια του εντοπισμού. Στο διάγραμμα με τις τροχιές των εντοπισμένων στόχων φαίνεται ότι οι επιτυχημένοι εντοπισμοί σχηματίζουν μια ευθεία, ενώ τα False positives βρίσκονται εκτός αυτής.

Αποτελέσματα δοκιμής Δ2 - Μετρικές ποσοτικής αξιολόγησης:

- True positives: 1596 (έχει εντοπιστεί ο σωστός στόχος ArUco)
- True negatives: 0 (δεν υπάρχουν εικόνες χωρίς στόχο ArUco)
- False positives: 6 (έχει εντοπιστεί λάθος στόχος)
- False negatives: 2304 (δεν έχει εντοπιστεί κανένας στόχος)
- Ορθότητα (Accuracy): 40,86%
- Ακρίβεια (Precision): 0,996
- Ευαισθησία (Recall): 0,409
- F1 score: 0,580

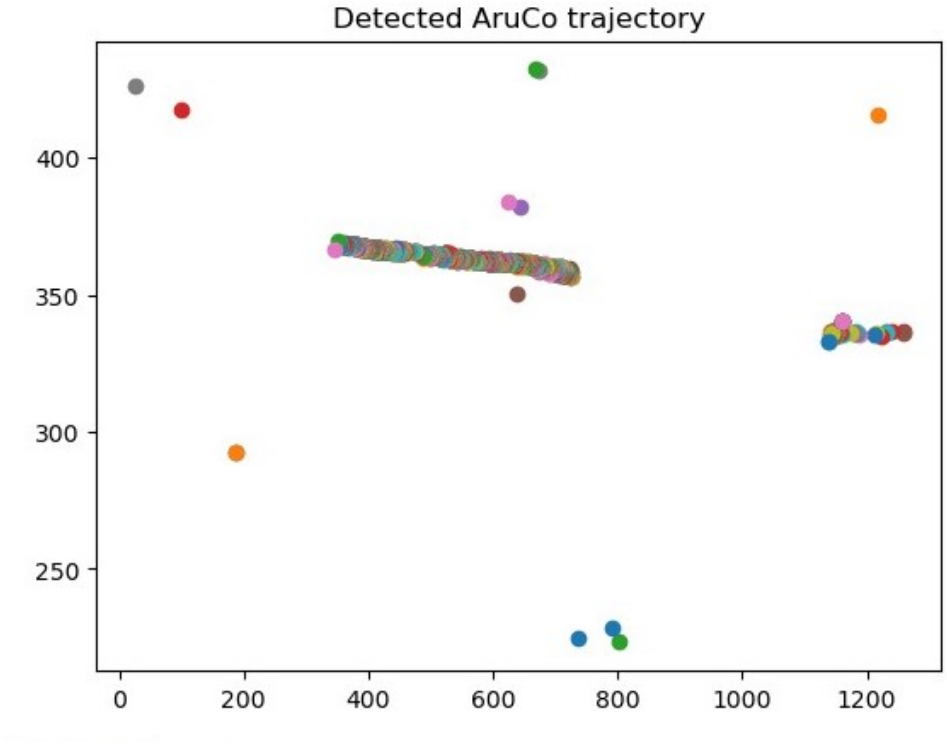


Σχήμα 27: Τροχιά εντοπισμένων ArUco, δοκιμή Δ2, άξονες X-Y: οριζόντια και κατακόρυφη διάσταση εικόνων (720,1280)

Τα αποτελέσματα της δοκιμής Δ2 (σχήμα 27) αυξάνουν το πλήθος των True positives και μειώνουν το πλήθος των False negatives, βελτιώνοντας της ορθότητα της διαδικασίας εντοπισμού. Το πλήθος των False positives έχει αυξηθεί κατά 2 σε σχέση με την δοκιμή Δ2 αλλά δεν αποτελεί πρόβλημα.

Αποτελέσματα δοκιμής Δ3 - Μετρικές ποσοτικής αξιολόγησης:

- True positives: 1934 (έχει εντοπιστεί ο σωστός στόχος ArUco)
- True negatives: 0 (δεν υπάρχουν εικόνες χωρίς στόχο ArUco)
- False positives: 13 (έχει εντοπιστεί λάθος στόχος)
- False negatives: 2037 (δεν έχει εντοπιστεί κανένας στόχος)
- Ορθότητα (Accuracy): 48,54%
- Ακρίβεια (Precision): 0,993
- Ευαισθησία (Recall): 0,487
- F1 score: 0,654

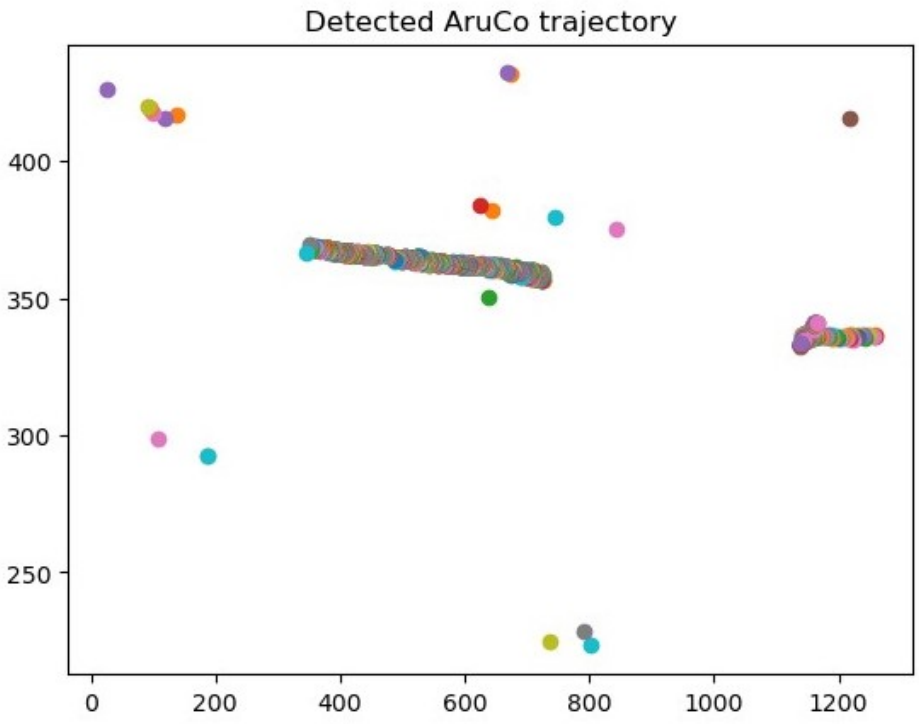


Σχήμα 28: Τροχιά εντοπισμένων ArUco, δοκιμή Δ3, άξονες X-Y: οριζόντια και κατακόρυφη διάσταση εικόνων (720,1280)

Στη δοκική Δ3 (σχήμα 28) αυξάνεται κι άλλο το πλήθος των True positives και μειώνεται περαιτέρω το πλήθος των False negatives. Αυξάνεται λίγο ακόμα το πλήθος των False positives όμως ακόμα κινείται σε πολύ χαμηλές τιμές δεδομένου του συνόλου των εικόνων που χρησιμοποιήθηκαν (3847).

Αποτελέσματα δοκιμής Δ4 - Μετρικές ποσοτικής αξιολόγησης:

- True positives: 2386 (έχει εντοπιστεί ο σωστός στόχος ArUco)
- True negatives: 0 (δεν υπάρχουν εικόνες χωρίς στόχο ArUco)
- False positives: 21 (έχει εντοπιστεί λάθος στόχος)
- False negatives: 1809 (δεν έχει εντοπιστεί κανένας στόχος)
- Ορθότητα (Accuracy): 56,59%
- Ακρίβεια (Precision): 0,991
- Ευαισθησία (Recall): 0,569
- F1 score: 0,723



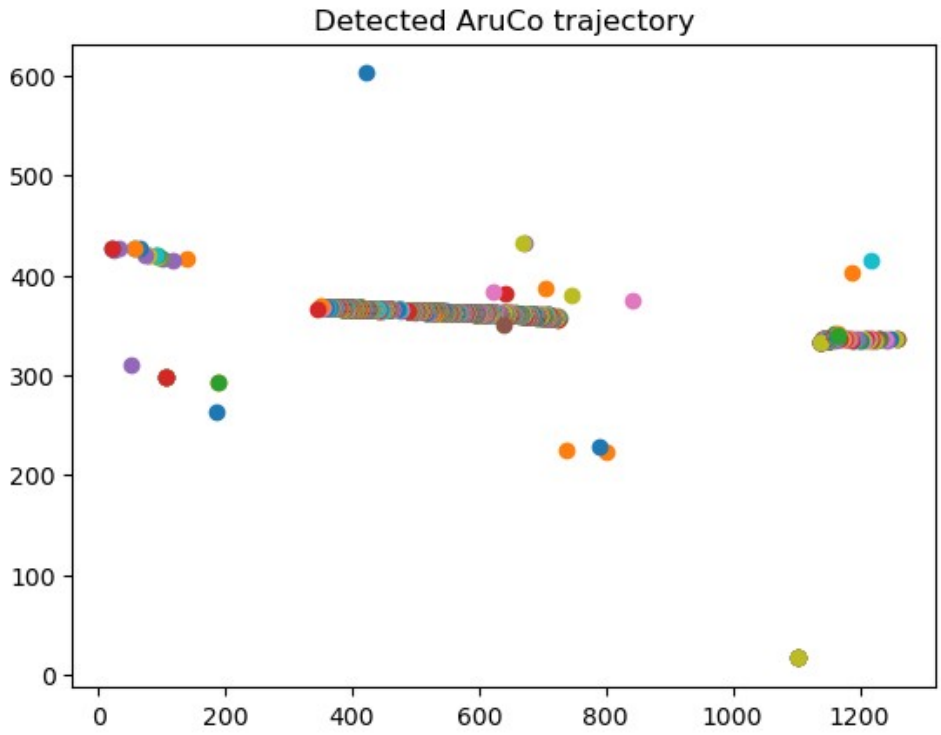
Σχήμα 29: Τροχιά εντοπισμένων ArUco, δοκιμή Δ4, άξονες X-Y: οριζόντια και κατακόρυφη διάσταση εικόνων (720,1280)

Τα αποτελέσματα είναι τα καλύτερα που μπόρεσαν να επιτευχθούν για τα διαθέσιμα δεδομένα. Μετά την βελτιστοποίηση των παραμέτρων εντοπισμού κατά τη δοκιμή Δ4, το πλήθος των ορθών επιτυχημένων εντοπισμών (True Positives) αυξήθηκε σημαντικά και το πλήθος των λανθασμένα επιτυχημένων εντοπισμών (False positives) είναι ακόμα πολύ μικρό για τον όγκο των δεδομένων (3487 εικόνες). Η ορθότητα του εντοπισμού έφτασε στο 56,59% που αποτελεί μια ικανοποιητική τιμή, ενώ οι υπόλοιπες μετρικές έχουν επίσης ικανοποιητικές τιμές.

Στο παραπάνω σχήμα 29 όπου παρουσιάζεται η τροχιά των εντοπισμένων ArUco κατά τη δοκιμή Δ5, φαίνεται και σχηματικά πως η βελτιστοποίηση των παραμέτρων εντοπισμού στόχων ArUco, σε σχέση με τη δοκιμή Δ0, αυξάνει το πλήθος των ορθών εντοπισμών (True Positives) και κατά συνέπεια την ακρίβεια εντοπισμού. Οι εντοπισμοί στην πλειοψηφία τους και εδώ παρουσιάζονται συγκεντρωμένοι σε μια γραμμή σχεδόν οριζόντια, γεγονός που επιβεβαιώνει το μεγάλο πλήθος των ορθών εντοπισμών. Σημειώνεται ότι τα True Positives σχηματίζουν ευθεία σε δύο τμήματα διότι το αρχικό βίντεο ξεκινάει με μια πόρτα αυτοκινήτου στη μέση του κάδρου και στη συνέχεια αυτή κινείται προς τα αριστερά ώσπου να αποχωρήσει απ' το κάδρο και στη συνέχεια, στα τελευταία δευτερόλεπτα, έρχεται μια νέα πόρτα από δεξιά.

Αποτελέσματα δοκιμής Δ5 - Μετρικές ποσοτικής αξιολόγησης:

- True positives: 2386 (έχει εντοπιστεί ο σωστός στόχος ArUco)
- True negatives: 0 (δεν υπάρχουν εικόνες χωρίς στόχο ArUco)
- False positives: 47 (έχει εντοπιστεί λάθος στόχος)
- False negatives: 1791 (δεν έχει εντοπιστεί κανένας στόχος)
- Ορθότητα (Accuracy): 56,49%
- Ακρίβεια (Precision): 0,981
- Ευαισθησία (Recall): 0,571
- F1 score: 0,722



Σχήμα 30: Τροχιά εντοπισμένων ArUco, δοκιμή Δ5, άξονες X-Y: οριζόντια και κατακόρυφη διάσταση εικόνων (720,1280)

Τα αποτελέσματα της δοκιμής Δ5 (σχήμα 30) είναι πολύ κοντά σε αυτά της δοκιμής Δ4 αλλά λίγο χειρότερα. Το πλήθος όμως των True positives δεν έχει μεταβληθεί σε σχέση με την δοκιμή Δ4 και δεδομένου ότι οι μετρικές είναι χειρότερες από αυτές της δοκιμής Δ4, τα αποτελέσματα της δοκιμής Δ4 θεωρούνται καλύτερα. Η προσπάθεια για περαιτέρω βελτιστοποίηση των παραμέτρων εντοπισμού έδεινε όλο και χειρότερα αποτελέσματα. Επομένως θεωρήθηκε ότι προσδιορίστηκαν οι βέλτιστες τιμές των παραμέτρων εντοπισμού στόχων ArUco για τα διαθέσιμα δεδομένα, στην δοκιμή Δ4.

Στη συνέχεια παρουσιάζεται ένας συγκεντρωτικός πίνακας με τις μετρικές ποσοτικής αξιολόγησης για όλες τις δοκιμές που έγιναν για τον εντοπισμό στόχων ArUco. Στον πίνακα επισημαίνονται οι μετρικές για τη δοκιμή Δ4 ως αυτές που έδωσαν τα καλύτερα αποτελέσματα εντοπισμού.

	Δ0	**Δ1**	**Δ2**	**Δ3**	**Δ4**	**Δ5**
TP	661	1459	1596	1934	**2386**	2386
TN	0	0	0	0	**0**	0
FP	943	4	6	13	**21**	47
FN	2480	2428	2304	2037	**1809**	1791
Accuracy	16,19%	37,50%	40,86%	48,54%	**56,59%**	56,49%
Precision	0,412	0,997	0,996	0,993	**0,991**	0,981
Recall	0,210	0,375	0,409	0,487	**0,569**	0,571
F1 score	0,279	0,545	0,580	0,654	**0,723**	0,722

Πίνακας 4: Συγκεντρωτικός πίνακας μετρικών ποσοτικής αξιολόγησης εντοπισμού ArUco

4.2 Αποτελέσματα εκπαίδευσης νευρωνικού δικτύου

Την εκπαίδευση του νευρωνικού δικτύου ακολουθεί το στάδιο της αξιολόγησης. Σε αυτό το στάδιο ελέγχεται ποιοτικά και ποσοτικά η ικανότητα του νευρωνικού δικτύου να ανιχνεύει τις κλάσεις που του ζητήθηκαν, αλλά και αν τα αποτελέσματα ικανοποιούν τις απαιτήσεις που έχουν τεθεί. Η αξιολόγηση έγινε χρησιμοποιώντας το εργαλείο COCOEvaluator [24] που διαθέτει η πλατφόρμα Detectron2 και πρόκειται για ένα εργαλείο αξιολόγησης που μπορεί να χρησιμοποιηθεί σε οποιοδήποτε μοντέλο εντοπισμού ή κατάτμησης εικόνων. Το συγκεκριμένο εργαλείο υπολογίζει τη μέση ακρίβεια (Average Precision) και τη μέση ευαισθησία (Average Recall) ανά εντοπισμένη κατηγορία για τα πλαίσια οριοθέτησης (bounding boxes) και τις μάσκες κατάτμησης (segmentation masks), που προκύπτουν από την πρόβλεψη του μοντέλου.

Η μέση ακρίβεια είναι ο μέσος όρος της μέσης τιμής της ακρίβειας σε 6 περιπτώσεις. Το COCOEvaluator χρησιμοποιεί την μετρική IoU (Intersection over Union) και υπολογίζει τη μέση ακρίβεια σε διάφορες τιμές της. Χρησιμοποιεί τις τιμές IoU=0.50:0.05:0.95 (10 τιμές από 0.50 μέχρι 0.95 με βήμα 0.05 από τις οποίες βγάζει μέσο όρο της ακρίβειας), IoU=0,5 και IoU=0.75. Επιπλέον, τα ανιχνευμένα απ' το δίκτυο αντικείμενα χωρίζονται σε μικρά, μεσαία, μεγάλα με βάση την έκταση της μάσκας κατάτμησης σε εικονοστοιχεία (pixel) και υπολογίζεται η μέση ακρίβεια και σε αυτές τις περιπτώσεις.

Η μέση ευαισθησία υπολογίζεται κι αυτή σε 6 διαφορεικές περιπτώσεις. Υπολογίζεται η μέγιστη ευαισθησία όταν υπάρχουν 1, 10 και 100 ανιχνεύσεις ανά εικόνα, ενώ και σε αυτή την περίπτωση τα ανιχνευμένα αντικείμενα χωρίζονται σε μικρά, μεσαία, μεγάλα και υπολογίζεται η μέση ευαισθησία για αυτά ξεχωριστά.

Σε αυτό το στάδιο αξιοποιούνται τα δεδομένα ελέγχου (validation data) που έχουν δημιουργηθεί. Σε αυτό το σύνολο δεδομένων γίνεται πρόβλεψη των κλάσεων «πόρτα αυτοκινήτου» και «άνθρωπος» και με βάση τα annotation που έχουν δημιουργθεί και χρησιμοποιούνται ως δεδομένα εκπαίδευσης ελέγχεται το κατά πόσο οι ανιχνευμένες μάσκες και τα annotation αυτά ταυτίζονται (Intersection over Union) ενώ υπολογίζονται και οι τιμές των μετρικών της μέσης ακρίβειας και της μέσης ευαισθησίας. Στην παρουσίαση των αποτελεσμάτων δίνεται μεγαλύτερη βάση στην μέση ακρίβεια ανά κατηγορία.

4.2.1 1η περίπτωση: «πόρτα αυτοκινήτου» με επικαλύψεις από ανθρώπους

Σε αυτή την περίπτωση το μοντέλο εκπαιδεύτηκε ώστε να αναγνωρίζει τις πόρτες, χρησιμοποιώντας annotation τα οποία περιγράφουν ολόκληρη την πόρτα, ακόμα και τα τμήματά της που αποκρύπτονται λόγω της παρουσίας ανθρώπων (σχήμα 31).

Προκειμένου να αξιολογηθεί ποιοτικά η εκπαίδευση του μοντέλου να ανγνωρίζει πόρτες αυτοκινήτου, παρουσιάζεται ένα παράδειγμα annotation που χρησιμοποιήθηκε στα δεδομένα εκπαίδευσης (σχήμα 31) και η πρόβλεψη που έκανε το δίκτυο στην ίδια εικόνα (σχήμα 32). Από την σύκριση των δύο σχημάτων φαίνεται ότι το μοντέλο επιτυγχάνει να αναγνωρίσει την πόρτα στην εικόνα και να ξεχωρίσει το περίγραμμά της.

Σχήμα 31: Δείγμα annotation από τα δεδομένα εκπαίδευσης, 1

Σχήμα 32: Παράδειγμα αποτελέσματος εντοπισμού πόρτας αυτοκινήτου, 1

Η ποσοτική αξιολόγηση, χρησιμοποώντας το εργαλείο COCOEvaluator, έδωσε αρκετά καλά αποτελέσματα τόσο στα πλαίσια οριοθέτησης, όσο και στις μάσκες κατάτμησης. Συγκεκριμένα στα πλαίσια οριοθέτησης το μοντέλο πέτυχε μέση ακρίβεια (Average Precision) 85,07% και στις μάσκες κατάτμησης 80,86%.

bounding boxes	segmentation masks
85,07%	80,86%

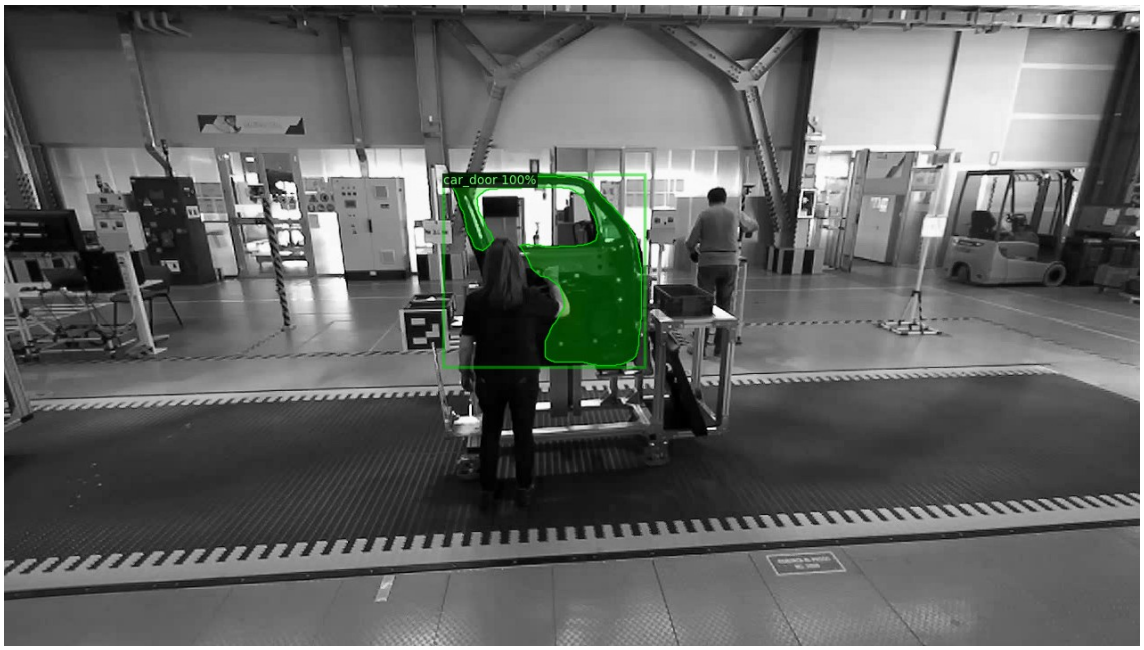
Πίνακας 5: Μέση ακρίβεια εντοπισμού πόρτας με επικαλύψεις από ανθρώπους

Η μικρότερη αξιοπιστία στις μάσκες κατάτμησης είναι φυσιολογική δεδομένου ότι το σχήμα τους είναι αρκετά περίπλοκο σε σχέση με τα πλαίσια οριοθέτησης και γι αυτό είναι αναμενόμενο να υπάρχουν διαφοροποιήσεις μεταξύ των annotation και των μασκών που δημιουργεί το εκπαιδευμένο μοντέλο. Σε κάθε περίπτωση οι τιμές της μέσης αξιοπιστίας (άνω του 80%) είναι ικανοποιητικές στο πλάισιο της συγκεκριμένης εργασίας.

4.2.2 2η περίπτωση: «πόρτα αυτοκινήτου» χωρίς επικαλύψεις από ανθρώπους

Στη δεύτερη περίπτωση, όπως αναφέρθηκε σε προηγούμενο κεφάλαιο, το μοντέλο και πάλι εκπαιδεύτηκε να αναγνωρίζει πόρτες αυτοκινήτου, με τη διαφορά όμως ότι τα annotation που δημιουργήθηκαν περιγράφουν μόνο τα εικονοστοιχεία της εικόνας που αντιστοιχούν σε πόρτα αυτοκινήτου, βγάζοντας εκτός ό,τι δημιουργεί αποκρύψεις (σχήμα 33).

Σχήμα 33: Δείγμα annotation από τα δεδομένα εκπαίδευσης, 2



Σχήμα 34: Παράδειγμα αποτελέσματος εντοπισμού πόρτας αυτοκινήτου, 2

Ποιοτικά, συγκρίνοντας το σχήμα 33 με το σχήμα 34, φαίνεται ότι το μοντέλο σε αυτή την περίπτωση έχει καταφέρει να ξεχωρίσει τον άνθρωπο από την πόρτα και έχει εξάγει ορθά το περίγραμμα της πόρτας.

Η ποσοτική αξιολόγηση έγινε και πάλι με το COCOEvaluator και τα αποτελέσματα έχουν ως εξής. Η μέση αξιοπιστία του μοντέλου για τα πλαίσια οριοθέτησης ανέρχεται στο 92,89%, ενώ για τις μάσκες κατάτμησης στο 66,98%.

bounding boxes	segmentation masks
92,89%	66,98%

Πίνακας 6: Μέση ακρίβεια εντοπισμού πόρτας χωρίς επικαλύψεις από ανθρώπους

Σε αντίθεση με την προηγούμνεη περίπτωση η μέση αξιοπιστία για τα πλαίσια οριοθέτησης και τις μάσκες κατάτμησης απέχει σημαντικά, μάλιστα η μέση αξιοπιστία για τα πλαίσια οριοθέτησης είναι μεγαλύτερη στη δεύτερη περίπτωση και για τις μάσκες κατάτμησης μικρότερη. Αυτό αφενός οφείλεται στο περίπλοκο σχήμα των μασκών, όπως αναφέρθηκε και παραπάνω. Αφετέρου αξίζει να αναφερθεί ότι το μοντέλο επανεκπαιδεύτηκε ώστε να αναγνωρίζει πόρτες αυτοκινήτου και μάλιστα σε μια συγκεκριμένη σκηνή, ενώ διαθέτει ήδη, από την αρχική εκπαίδευση, την ικανότητα να αναγνωρίζει ανθρώπους. Αυτό ενδεχομένως να καθιστά τα annotation της δεύτερης περίπτωσης περισσότερο συμβατά με το μοντέλο, δεδομένου ότι οι περιοχές των εικόνων όπου ο άνθρωπος επικαλύπτει την πόρτα δεν δηλώνονται στα annotation ως πόρτα αυτοκινήτου. Έτσι, το μοντέλο μπορεί καλύτερα να ξεχωρίσει τις δύο αυτές κλάσεις και κατά συνέπεια τα πλάισια οριοθέτησης που θα δημιουργήσει σχεδόν να ταυτίζονται με αυτά που δίνονται από τα annotation, παρόλο που οι άνθρωποι σε αυτή τη φάση δεν ζητείται να εντοπιστούν.

Από την πρώτη περίπτωση (σχήμα 35α) διαφαίνεται ότι το μοντέλο δεν χειρίζεται την επικάλυψη των πορτών από ανθρώπους ακριβώς όπως του δίνεται στη φάση της εκπαίδευσης. Αποτέλεσμα είναι να προκύπτουν μάσκες που αφήνουν εκτός τμήματα από τους ανθρώπους παρόλο που η εκπαίδευση έχει γίνει με annotation που περιλαμβάνουν τον άνθρωπο που καλύπτει την πόρτα. Στη δεύτερη περίπτωση (σχήμα 35β) η συμπεριφορά του μοντέλου κατά την πρόβλεψη δεν προβληματίζει, δεδομένου ότι τα annotation, που χρησιμοποιήθηκν κατά την εκπαίδευση, επισημειώνουν ως πόρτα μόνο τα τμήματα της πόρτα που δεν καλύπτονται από κάποιο εμπόδιο.

Αυτό δείχνει ότι η διαφορά μεταξύ των δύο περιπτώσεων εκπαίδευσης του μοντέλου ώστε να ανγνωρίζει πόρτες αυτοκινήτου δεν διαφοροποιείται σημαντικά παρόλο που τα δεδομένα εκπαίδευσης που χρησιμοποιούνται έχουν διαφορετικές προσεγγίσεις. Για αυτόν κυρίως το λόγο, όπως αναφέρθηκε και στο κεφάλαιο 3, για τη συνέχεια των πειραμάτων επιλέχθηκε το σύνολο των δεδομένων εκπαίδευσης που παρουσιάστηκε στην πρώτη περίπτωση.

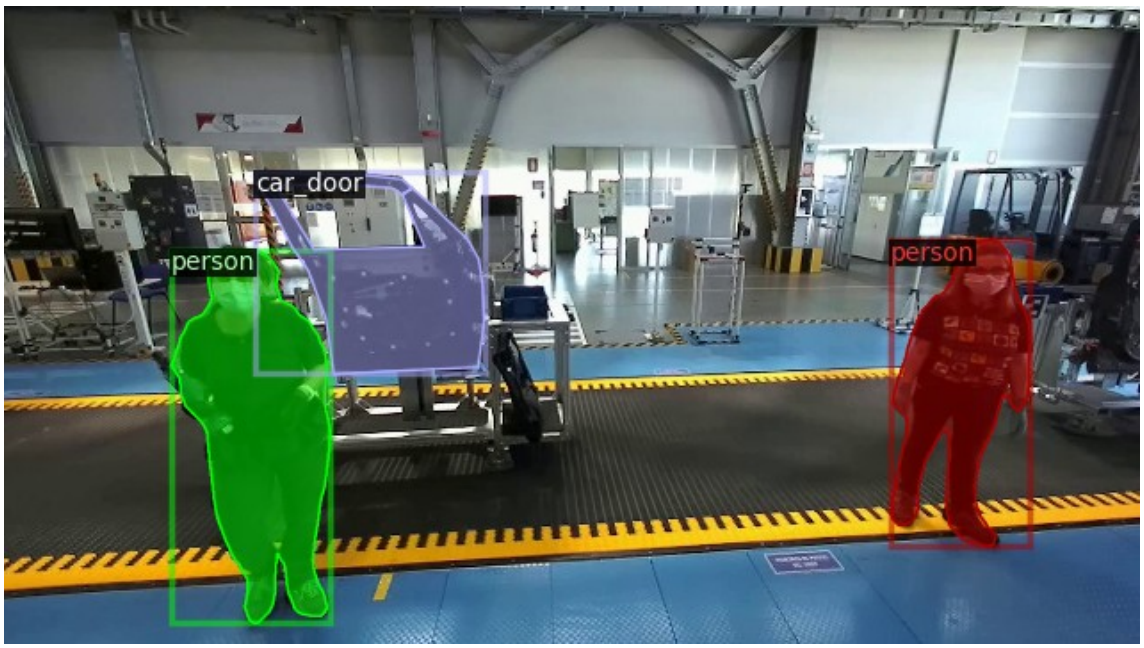
(α□) Εντοπισμένη πόρτα, περίπτωση 1

(β□) Εντοπισμένη πόρτα, περίπτωση 2

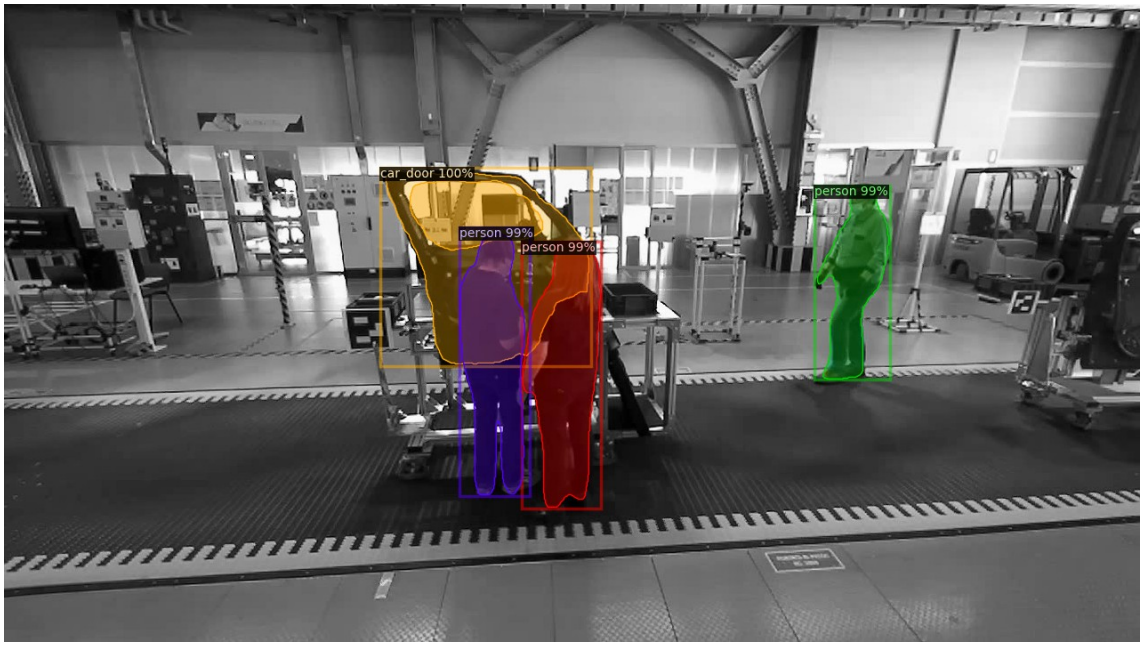
Σχήμα 35: Παράδειγμα εντοπισμού πόρτας αυτοκινήτου στην 1η και τη 2η περίπτωση στην ίδια εικόνα

4.2.3 Ανίχνευση πόρτας αυτοκινήτου και ανθρώπων

Όπως έχει αναφερθεί και προηγουμένως, για τη δημιουργία του τελικού μοντέλου, χρησιμοποιήθηκαν annotation για την πόρτα και τους ανθρώπους (σχήμα 36).



Σχήμα 36: Δείγμα annotation από τα τελικά δεδομένα εκπαίδευσης



Σχήμα 37: Παράδειγμα αποτελέσματος εντοπισμού πόρτας αυτοκινήτου και ανθρώπων

Με το τελικό σετ παραμέτρων εκπαίδευσης που χρησιμοποιήθηκε το μοντέλο πέτυχε πολύ καλά αποτελέσματα (37), δεδομένου και του σχετικά μικρού συνόλου δεδομένων εκπαίδευσης. Συγκεκριμένα το εκπαιδευμένο μοντέλο πέτυχε μέση αξιοπιστία (Average Precision) 88,20% στα πλαίσια οριοθέτησης (bounding boxes) και 79,63% στις μάσκες κατάτμησης (segmentation masks). Αναλυτικότερα, για κάθε κλάση η μέση αξιοπιστία είναι ως εξής. Στην κλάση «πόρτα αυτοκινήτου» το μοντέλο πέτυχε μέση αξιοπιστία 90,75% στα πλαίσια οριοθέτησης και 73,56% στις μάσκες κατάτμησης. Στην κλάση «άνθρωπος» το μοντέλο πέτυχε μέση αξιοπιστία 85,65% στα πλαίσια οριοθέτησης και 85,69% στις μάσκες κατάμησης.

	bounding boxes	segmentation masks
πόρτα αυτοκινήτου	90,75%	73,56%
άνθρωπος	85,65%	85,69%
Μ. Ο.	88,20%	79,63%

Πίνακας 7: Μέση ακρίβεια εντοπισμού πόρτας αυτοκινήτου και ανθρώπων

Όπως φαίνεται και από τα αριθμητικά στοιχεία, η μέση αξιοπιστία του μοντέλου είναι μεγαλύτερη στα πλαίσια οριοθέτησης απ' ότι στις μάσκες κατάτμησης και αυτό οφείλεται στο σχήμα των περιοχών που ανιχνεύονται. Τα πλάισια οριοθέτησης είναι ορθογώνια τμήματα των εικόνων που στο εσωτερικό τους περιέχουν το αντικείμενο που αντιστοιχεί στην κάθε εντοπισμένη κλάση, επομένως η διαφορά μεταξύ μασκών και annotation αναμένεται να είναι η θέση του πλαισίου στην κάθε εικόνα, η διαφορά δεν μπορεί να είναι μεγάλη. Οι μάσκες κατάτμησης, επειδή περιγράφουν με λεπτομέρεια το σχήμα των αντικειμένων που εντοπίζονται, έχουν ακανόνιστο σχήμα και άρα είναι αναμενόμενο οι μάσκες

που προβλέπει το μοντέλο και τα annotation που έχουν δημιουργηθεί να έχουν διαφορές.

Θεωρώντας την εκπαίδευση και την αξιολόγηση του συνελικτικού νευρωνικού δικτύου ολοκληρωμένη, προέκυψε το τελικό μοντέλο. Μια άλλη ένδειξη της αξιοπιστίας του τελικού μοντέλου είναι η δοκιμή του σε νέα δεδομένα διαφορετικά από αυτά που χρησιμοποιήθηκαν για την εκπαίδευση και την αξιολόγησή του. Στη συγκεκριμένη περίπτωση χρησιμοποιήθηκαν δεδομένα από 5 βίντεο που απεικονίζουν διαφορετικές σκηνές από συναρμολόγηση πορτών αυτοκινήτου στο ίδιο βιομηχανικό περιβάλλον. Αυτό που διαφοροποιείται σε κάποιες περιπτώσεις είναι η θέση της κάμερας. Υπάρχουν δύο είδη δεδομένων: εικόνες όπου η κάμερα βρίσκεται στην ίδια περίπου θέση με τα δεδομένα εκπαίδευσης (βλέπει την πόρτα αυτοκινήτου από την ίδια πλευρά) αλλά έχουν ληφθεί άλλη χρονική στιγμή και εικόνες όπου η θέση της κάμερας είναι διαφορετική (βλέπει την πόρτα αυτοκινήτου από την άλλη πλευρά). Διαφορές επίσης εντοπίζονται στο πλήθος των πορτών που ενυπάρχουν σε κάθε εικόνα αλλά και στο μέγεθος των πορτών, δηλαδή στην απόσταση των πορτών από την κάμερα. Συνολικά χρησιμοποιήθηκαν δεδομένα (εικόνες) από 5 βίντεο, όπου δοκιμάστηκε η συμπεριφορά του τελικού μοντέλου, που αναφέρθηκε παραπάνω. Έγιναν 5 δοκιμές όπου σε όλες υπήρχαν δύο ή περισσότερες πόρτες αυτοκινήτου και επίσης υπήρχαν άνθρωποι που αλληλεπιδρούσαν με τις πόρτες. Χρησιμοποιήθηκαν λοιπόν 5 βίντεο στα 3 από τα οποία η κάμερα βλέπει την πόρτα από την εσωτερική πλευρά (**θέση κάμερας 1**) και στα υπόλοιπα 2 η κάμερα βλέπει την πόρτα από την εξωτερική πλευρά (**θέση κάμερας 2**). Ο έλεγχος έγινε ποιοτικά, καθώς δεν δημιουργήθηκαν δεδομένα ελέγχου.

- **Θέση κάμερας 1**: Δοκιμάστηκαν 3 τέτοια βίντεο όπου διακρίνονται δύο πόρτες τοποθετημένες στη βάση στήριξης, σε αντίστοιχη απόσταση από την κάμερα με τα δεδομένα εκπαίδευσης (σχήμα 38). Σημειώνεται επίσης ότι οι πόρτες είναι ορατές από την ίδια πλευρά (εσωτερική) όπως στα δεδομένα εκπαίδευσης, αυτό σχολιάζεται διότι η βάση στήριξης δεν είναι ίδια και από τις δύο πλευρές.

Σχήμα 38: Παράδειγμα εικόνας από τη θέση κάμερας 1

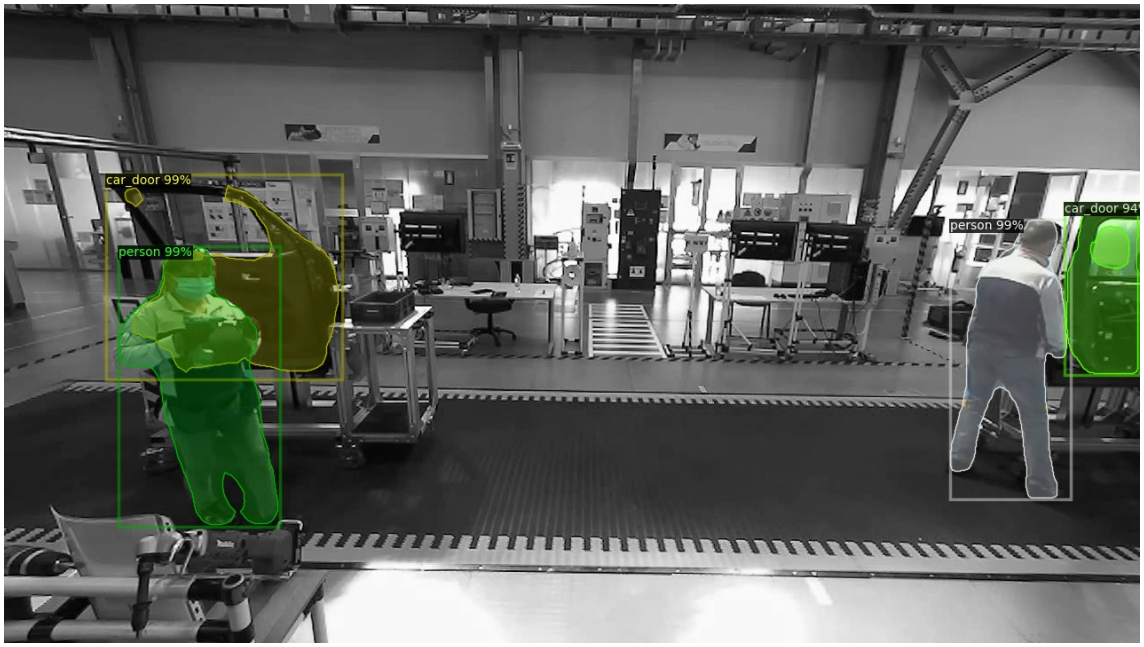
- **Θέση κάμερας 2**: Δοκιμάστηκαν 2 τέτοια βίντεο στα οποία υπάρχουν δύο πόρτες όπου η απόστασή τους από την κάμερα είναι αντίστοιχη με αυτήν στα δεομένα εκπαίδευσης, ενώ στο υπόβαθρο, σε μεγαλύτερη απόσταση από την κάμερα περιστασικά περνούν κι άλλες πόρτες πάνω στις βάσεις στήριξής τους (σχήμα 39). Σημαντική διαφοροποίηση με το προηγούμενο βίντεο είναι ότι οι πόρτες είναι ορατές από την αντίθετη πλευρά (εξωτερική) και η βάση στήριξης καλύπει μέρος της πόρτας.

Σχήμα 39: Παράδειγμα εικόνας από τη θέση κάμερας 2

Και στις 2 θέσεις κάμερας που εξετάστηκαν (και στα 5 βίντεο) το μοντέλο επιτυγχάνει να ανιχνεύσει πόρτες αυτοκινήτου, όχι όμως με την ίδια επιτυχία. Αντίθετα οι άνθρωποι δεν αποτελούν πρόβλημα και εντοπίζονται με επιτυχία σε όλες τις περιπτώσεις.

Στη θέση κάμερας 1, όπου οι συνθήκες ήταν πολύ παρόμοιες με αυτές στις οποίες είχε εκπαιδευτεί το μοντέλο το αποτέλεσμα ήταν αρκετά καλό με το μοντέλο να αναγνωρίζει και τις δύο πόρτες και τους ανθρώπους στα περισσότερα καρέ του βίντεο (σχήμα 40, σχήμα 41). Οι μάσκες κατάτμησης έχουν αρκετά καλή λεπτομέρεια και επισημειώνουν ολόκληρη την πόρτα ή τον άνθρωπο, χωρίς να αφήνουν τμήματά εκτός.

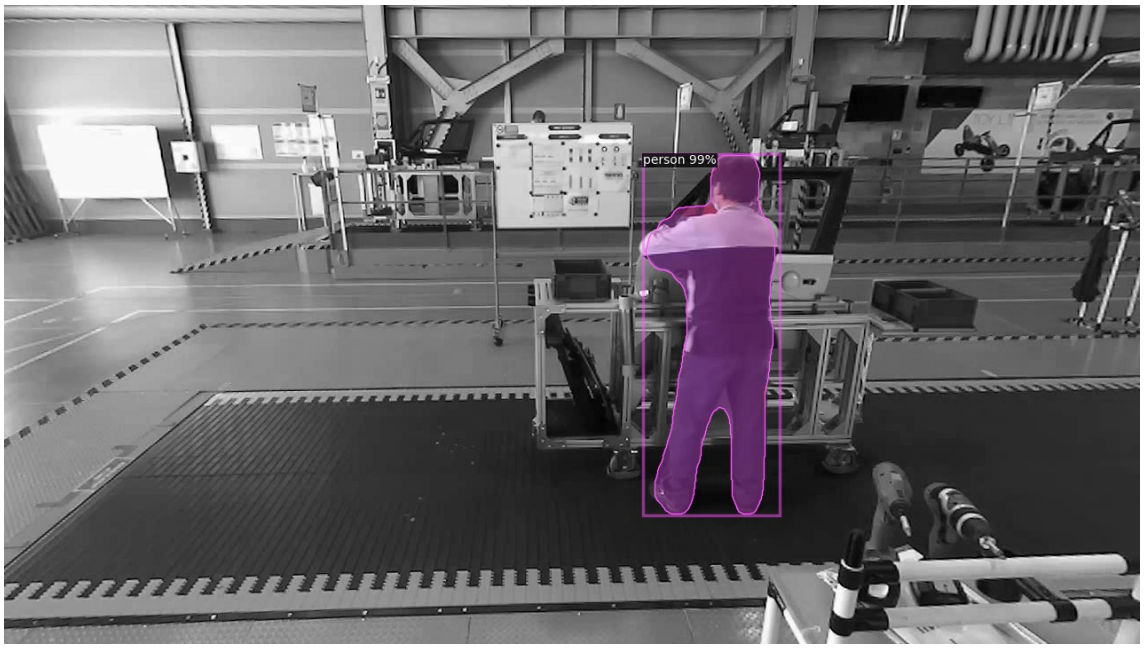
Σχήμα 40: Δοκιμή τελικού μοντέλου, θέση κάμερας 1, επιτυχής εντοπισμός πορτών



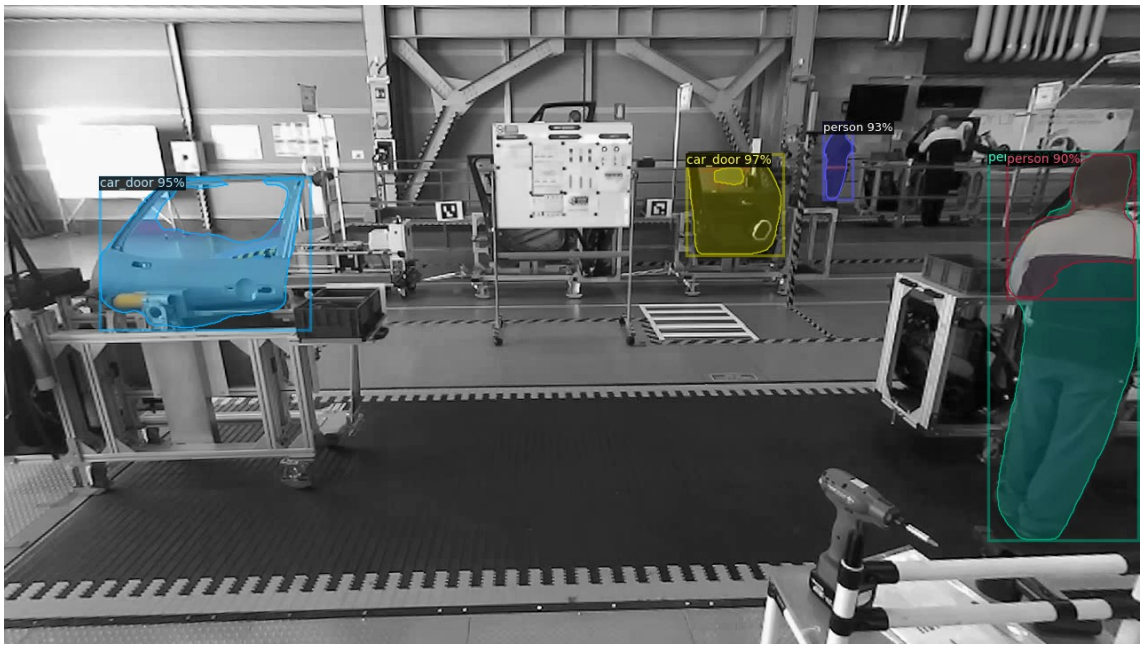
Σχήμα 41: Δοκιμή τελικού μοντέλου, θέση κάμερας 1, επιτυχής εντοπισμός πορτών αυτοκινή του και ανθρώπων

Στη θέση κάμερας 2 το γεγονός ότι οι πόρτες ήταν ορατές από την εξωτερική πλευρά και η βάση στήριξης κάλυπτε τμήμα τους φαίνεται να δυσκόλεψε το μοντέλο στον εντοπισμό και έτσι ήταν λίγες οι επιτυχημένες ανιχνεύσεις (σχήμα 42, σχήμα 43). Οι πόρτες που βρίσκονταν μακρυά από την κάμερα επίσης φαίνεται να αποτέλεσαν δυσκολία διότι δεν εντοπίζονταν συχνά ίσως και λόγω των αποκρύψεών τους από τις μπροστινές πόρ-

τες ή τους ανθρώπους που περνούσαν. Αντίθετα οι άνθρωποι, όπως αναφέρθηκε και προηγουμένως, εντοπίστηκαν με επιτυχία ανεξάρτητα από το που βρίσκονταν μέσα στην σκηνή (σχήμα 42, σχήμα 43). Και σε αυτή την περίπτωση οι μάσκες φαίνεται να έχουν ικανοποιητική λεπτομέρεια και να επισημειώνουν ολόκληρο το αντικείμενο.



Σχήμα 42: Δοκιμή τελικού μοντέλου, θέση κάμερας 2, επιτυχής εντοπισμός μόνο ανθρώπων και όχι των πορτών αυτοκινήτου



Σχήμα 43: Δοκιμή τελικού μοντέλου, θέση κάμερας 2, επιτυχής εντοπισμός κάποιων πορτών

5 Συμπεράσματα και προτάσεις για βελτίωση

5.1 Γενικά συμπεράσματα και σχόλια

Οι δύο μέθοδοι που χρησιμοποιήθηκαν για τον εντοπισμών των πορτών αυτοκινήτου έδωσαν πολύ ικανοποιητικά αποτελέσματα χωρίς να απαιτούν μεγάλη περιπλοκότητα στην εφαρμογή. Παρόλα αυτά δεν πρόκειται για τις πιο αποδοτικές μεθόδους εντοπισμού είτε των στόχων ArUco, είτε των ίδιων των πορτών. Αυτό δεν αποτελούσε σημαντική απαίτηση σε καμία φάση της εργασίας, ο μόνος περιορισμός ήταν οι αλγόριθμοι που δημιουργήθηκαν να μπορούν να υποστηρικτούν από τους διαθέσιμους πόρους, κάτι όμως που δεν αποτέλεσε πρόβλημα. Μια επέκταση της παρούσας εργασίας θα μπορούσε να είναι η βελτιστοποίηση των αλγορίθμων που δημιουργήθηκαν για τον εντοπισμό των στόχων ArUco και για την εκπαίδευση του νευρωνικού δικτύου, ώσε να αξιοποιούν τους διαθέσιμους πόρους πιο συντηρητικά και να εκτελούνται πιο γρήγορα.

Σχετικά με την διαδικασία εντοπισμού των στόχων ArUco, όπως παρουσιάστηκε και στα αντίστοιχα κεφάλαια, χάρη στα διαθέσιμα πακέτα της opencv και σε γλώσσα προγραμματισμού python η διαδικασία ήταν απλή χωρίς να απαιτεί εξειδικευμένες γνώσεις προγραμματισμού. Με μικρές τροποιήσεις στις παραμέτρους εντοπισμού και μερικές δοκιμές ήταν εφικτό να προκύψουν πολύ ικανοποιητικά αποτελέσματα.

Αναφορικά με την διαδικασία εντοπισμού των πορτών αυτοκινήτου σε εικόνες χρησιμοποιώντας ένα συνελικτικό νευρωνικό δίκτυο, όπως παρουσιάστηκε και στα παραπάνω κεφάλαια, προκύπτει ότι η διαδικασία απλοποιείται σημαντικά με τη βοήθεια της πλατφόρμας Detectron2. Χρησιμοποιώντας τη συγκεκριμένη πλατφόρμα το στήσιμο ενός αλγορίθμου που θα χρησιμοποιεί ένα έτοιμο εκπαιδευμένο μοντέλο ή η εκπαίδευση ενός μοντέλου είναι εξαιρετικά απλή και δεν απαιτεί μεγάλη εμπειρία στον προγραμματισμό ή στα νευρωνικά δίκτυα. Η μόνη διαδικασία που απαιτεί περισσότερο κόπο είναι η διαδικασία δημιουργίας (annotation) για την προετοιμασία των δεδομένων εκπαίδευσης και ελέγχου.

5.2 Τροποποίηση της εκπαίδευσης του νευρωνικού δικτύου

Στην παρούσα εργασία χρησιμοποιήθηκε ένα ήδη εκπαιδευμένο μοντέλο, το Mask R-CNN R50 FPN, το οποίο επανεκπαιδεύτηκε ώστε να αναγνωρίζει πόρτες αυτοκινήτου και ανθρώπους. Το τελικό μοντέλο διαθέτει όλη την αποκτημένη γνώση από την αρχική εκπαίδευση και με τη δεύτερη εκπαίδευση είναι σε θέση να αναγνωρίζει επιπλέον δύο κλάσεις. Η πρόβλεψη του μοντέλου όμως κατατμεί στις δοσμένες εικόνες μόνο τις πόρτες αυτοκινήτου και του ανθρώπους, τις κλάσεις δηλαδή που εκπαιδεύτηκε να αναγνωρίζει στη δεύτερη εκπαίδευση.

Μια επέκταση στη δουλειά που έχει γίνει στο πλάισιο της εργασίας είναι η τροποποίηση του τελικού μοντέλου που δημιουργήθηκε ώστε, μαζί με τις κλάσεις «πόρτα αυτοκινήτου» και «άνθρωπος», να μπορεί να προβλέπει μάσκες από όλες τις κλάσεις που διαθέτει το

COCO dataset [24], στο οποίο έχει γίνει και η αρχική του εκπαίδευση. Ένας τρόπος που ίσως μπορούσε να υλοποιηθεί αυτό είναι στο στάδιο της επανεκπαίδευσης του μοντέλου να δωθούν μαζί με τα δεδομένα εκπαίδευσης που δημιουργήθηκαν και όλα τα δεδομένα εκπαίδευσης του COCO dataset για τις 91 κλάσεις που διαθέτει. Τα δεδομένα αυτά είναι διαθέσιμα και μπορούν εύκολα να αποκτηθούν, όμως το πλήθος και ο όγκος τους είναι πολύ μεγάλος. Συγκεκριμένα το COCO dataset αποτελείται από 328.000 εικόνες με annotation για 91 κλάσεις που αφορούν καθημερινά αντικείμενα. Αυτός ο όγκος δεδομένων εκπαίδευσης είναι πολύ απαιτητικός σε πόρους και θα χρειάζεται μεγάλο χρονικό διάστημα για την εκπαίδευση του δικτύου.

5.3 Εκτίμηση πόζας της πόρτας με βάση τις ακμές

Πηγαίνοντας την παρακολούθηση της πόρτας αυτοκινήτου ένα βήμα παραπέρα, πέρα από την κατάτμηση εικόνων από βίντεο για την εξαγωγή μασκών που περιγράφουν την πόρτα κάθε χρονική στιγμή, μπορεί να γίνει και η εκτίμηση της τρισδιάστατης πόζας της πόρτας με βάση τις ακμές. Σε αυτό το σημείο μπορούν να χρησιμοποιηθούν και οι εικόνες βάθους από την κάμερα ZED [1] που χρησιμοποιήθηκε για την συλλογή των δεδομένων. Η εξαγωγή των ακμών σε συνδυασμό με την πληροφορία του βάθους μπορούν να δώσουν την θέση της πόρτας στο χώρο, εφόσον μπορεί να προσδιοριστεί η θέση της κάθε εκμής στο χώρο.

Αναφορές

[1] StereoLabs (2024) Cameras. Διαθέσιμο στο: https://store.stereolabs.com/en-eu/collections/cameras (Πρόσβαση: Μάρτιος 2024)

[2] Wikipedia (2024) Digital twin. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Digital_twin (Πρόσβαση: Φεβρουάριος 2024)

[3] StereoLabs (2024) TERRA AI. Διαθέσιμο στο: https://www.stereolabs.com/our-technology (Πρόσβαση: Μάρτιος 2024)

[4] OpenCV (2024) Detection of ArUco Markers. Διαθέσιμο στο: https://docs.opencv.org/3.4/d5/dae/tutorial_aruco_detection.html (Πρόσβαση: Φεβρουάριος 2024)

[5] Minor, B. (2021) Reverse-Engineering Fiducial Markers For Perception. Διαθέσιμο στο: https://www.tangramvision.com/blog/reverse-engineering-fiducial-markers-for-perception (Πρόσβαση: Φεβρουάριος 2024)

[6] Wikipedia (2024) Otsu's method. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Otsu%27s_method (Πρόσβαση: Φεβρουάριος 2024)

[7] Wikipedia (2024) Neural network (machine learning). Διαθέσιμο στο: https://en.wikipedia.org/wiki/Neural_network_(machine_learning) (Πρόσβαση: Φεβρουάριος 2024)

[8] Wikipedia (2024) Empirical risk minimization. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Empirical_risk_minimization (Πρόσβαση: Φεβρουάριος 2024)

[9] Wikipedia (2024) Gradient. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Gradient (Πρόσβαση: Φεβρουάριος 2024)

[10] Wikipedia (2024) Loss functions for classification. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Loss_functions_for_classification (Πρόσβαση: Φεβρουάριος 2024)

[11] Machine Learning in Plain English (2023) Deep Learning Course — Lesson 5: Forward and Backward Propagation. Διαθέσιμο στο: https://medium.com/@nerdjock/deep-learning-course-lesson-5-forward-and-backward-propagation-ec8e4e6a8b92 (Πρόσβαση: Φεβρουάριος 2024)

[12] IBM (2024) What are convolutional neural networks?. Διαθέσιμο στο: https://www.ibm.com/topics/convolutional-neural-networks (Πρόσβαση: Φεβρουάριος 2024)

[13] Goodfellow, I. and Bengio, Y. and Courville, A. (2016) Deep Learning. MIT Press. Διαθέσιμο στο: https://www.deeplearningbook.org/

[14] Wikipedia (2024) Convolutional neural network. Διαθέσιμο στο: https://en.wikipedia.org/wiki/Convolutional_neural_network (Πρόσβαση: Φεβρουάριος 2024)

[15] He, K. and Gkioxari, G. and Dollár, P. and Girshick, R. (2018) Mask R-CNN. Διαθέσιμο στο: https://arxiv.org/abs/1703.06870

[16] Rosebrock, A. (2018) Mask R-CNN with OpenCV. Διαθέσιμο στο: https://pyimagesearch.com/2018/11/19/mask-r-cnn-with-opencv/ (Πρόσβαση: Φεβρουάριος 2024)

[17] Girshick, R. and Donahue, J. and Darrell, T. and Malik, J. (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. Διαθέσιμο στο: https://arxiv.org/abs/1311.2524

[18] Girshick R. (2015) Fast R-CNN. Διαθέσιμο στο: https://arxiv.org/abs/1504.08083

[19] Ren, S. and He, K. and Girshick, R. and Sun, J. (2016) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Διαθέσιμο στο: https://arxiv.org/abs/1506.01497

[20] Majumder, S. (2020) Object Detection Algorithms-R CNN vs Fast-R CNN vs Faster-R CNN. Διαθέσιμο στο: https://medium.com/analytics-vidhya/object-detection-algorithms-r-cnn-vs-fast-r-cnn-vs-faster-r-cnn-3a7bbaad2c4a (Πρόσβαση: Μάρτιος 2024)

[21] Meta (2024) Detectron2. Διαθέσιμο στο: https://ai.meta.com/tools/detectron2/ (Πρόσβαση: Φεβρουάριος 2024)

[22] Honda, H. (2020) Digging into Detectron 2 — part 1. Διαθέσιμο στο: https://medium.com/@hirotoschwert/digging-into-detectron-2-47b2e794fabd (Πρόσβαση: Φεβρουάριος 2024)

[23] Open Images Dataset V7 (2022) Open Images Dataset V7 and Extensions. Διαθέσιμο στο: https://storage.googleapis.com/openimages/web/index.html (Πρόσβαση: Φεβρουάριος 2024)

[24] Lin, T.-Y. and Maire, M. and Belongie, S. and Bourdev, L. and Girshick, R. and Hays, J. and Perona, P. and Ramanan, D. and Zitnick, L. and Dollár, P. (2015) Microsoft COCO: Common Objects in Context. Διαθέσιμο στο: https://arxiv.org/abs/1405.0312

[25] OpenCV team (2024) OpenCV. Διαθέσιμο στο: https://opencv.org/ (Πρόσβαση: Φεβρουάριος 2024)

[26] Dutta, A. and Zissermann, A. (2019) 'The VIA Annotation Software for Images, Audio and Video'. In Proceedings of the 27th ACM International Conference on Multimedia (MM '19), October 21–25, 2019, Nice, France. ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3343031.3350535. Διαθέσιμο στο: https://www.robots.ox.ac.uk/~vgg/software/via/

[27] Wu, Y. and Facebook Community Bot (2021) detectron2/MODEL_ZOO.md. Διαθέσιμο στο: https://github.com/facebookresearch/detectron2/blob/main/MODEL_ZOO.md (Πρόσβαση: Μάρτιος 2024)

[28] Shroff, M. (2023) Know your Neural Network architecture more by understanding these terms. Διαθέσιμο στο: https://medium.com/@shroffmegha6695/know-your-neural-network-architecture-more-by-understanding-these-terms-67faf4ea0efb (Πρόσβαση: Μάρτιος 2024)

[29] Agrawal, S. (2024) Metrics to Evaluate your Classification Model to take the right decisions. Διαθέσιμο στο: https://www.analyticsvidhya.com/blog/2021/07/metrics-to-evaluate-your-classification-model-to-take-the-right-decisions/ (Πρόσβαση: Φεβρουάριος 2024)

[30] Rosebrock, A. (2016) Intersection over Union (IoU) for object detection. Διαθέσιμο στο: https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/ (Πρόσβαση: Φεβρουάριος 2024)