

**Ε.Π.Ο.Σ. - σύστημα Επαυξημένης
Πραγματικότητας
για την Οδική ασφάλεια έξυπνων πόλεων
Συνδεδεμένων αυτόνομων οχημάτων**

Βασίλειος Καραϊσκος

Μεταπτυχιακή Εργασία Εξειδίκευσης



Ιωάννινα, Δεκέμβριος 2025



ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ & ΠΛΗΡΟΦΟΡΙΚΗΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
UNIVERSITY OF IOANNINA

Ε.Π.Ο.Σ. - προτεινόμενο σύστημα Επαυξημένης
Πραγματικότητας για την Οδική ασφάλεια
έξυπνων πόλεων Συνδεδεμένων αυτόνομων
οχημάτων

Η Μεταπτυχιακή Διπλωματική Εργασία

υποβάλλεται στην ορισθείσα

από τη Συνέλευση

του Τμήματος Μηχανικών Η/Υ και Πληροφορικής

Εξεταστική Επιτροπή

από τον

Βασίλειο Καραϊσκό

ως μέρος των υποχρεώσεων για την απόκτηση του

ΔΙΠΛΩΜΑΤΟΣ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΣΤΗ ΜΗΧΑΝΙΚΗ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΥΠΟΛΟΓΙΣΤΙΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ

ΜΕ ΕΙΔΙΚΕΥΣΗ
ΣΤΑ ΠΡΟΗΓΜΕΝΑ ΥΠΟΛΟΓΙΣΤΙΚΑ ΣΥΣΤΗΜΑΤΑ

Πανεπιστήμιο Ιωαννίνων

Πολυτεχνική Σχολή

Ιωάννινα 2025

Εξεταστική Επιτροπή:

- **Ιωάννης Φούντος**, Καθηγητής, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων (Επιβλέπων)
- **Χριστόφορος Νίκου**, Καθηγητής, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων
- **Βασίλειος Δημακόπουλος**, Καθηγητής, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων

ΑΦΙΕΡΩΣΗ

Στην οικογένεια μου και στα υπέροχα κατοικίδια μου, Λίλη και Άκιρα.

ΕΥΧΑΡΙΣΤΙΕΣ

Θα ήθελα να εκφράσω την ειλικρινή μου ευγνωμοσύνη προς τον επιβλέποντά μου, Καθηγητή Ιωάννη Φούντο, για την πολύτιμη καθοδήγηση και την αδιάκοπη υποστήριξη που μου έδειξε καθ' όλη τη διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας. Η συμβολή του, τόσο σε επίπεδο επιστημονικής καθοδήγησης όσο και έμπνευσης, υπήρξε καθοριστική για την ολοκλήρωση του έργου αυτού.

Επιπλέον, θα ήθελα να ευχαριστήσω θερμά τον Δρ. Σωτήριο Κοντογιάννη για τη σημαντική συνδρομή του, την καθοδήγηση και εμπιστοσύνη που έδειξε στο πρόσωπο μου κατά την επαγγελματική συνεργασία μας, η οποία συνέβαλε ουσιαστικά στη βελτίωση και την εμβάθυνση της τεχνογνωσίας που χρειάστηκε για την εκτέλεση της παρούσας μελέτης.

Τέλος, εκφράζω την βαθιά μου ευγνωμοσύνη προς την οικογένειά μου για τη συνεχή στήριξη, την υπομονή, την αγάπη και την αμέριστη ενθάρρυνσή τους καθ' όλη τη διάρκεια των σπουδών μου και της εκπόνησης της εργασίας αυτής. Χωρίς τη δική τους παρουσία και πίστη στις δυνατότητές μου, η ολοκλήρωση της παρούσας μελέτης δεν θα ήταν εφικτή.

ΠΕΡΙΕΧΟΜΕΝΑ

Κατάλογος Σχημάτων	iii
Κατάλογος Πινάκων	v
Κατάλογος Αλγορίθμων	vi
Περίληψη	vii
Extended Abstract	ix
1 Εισαγωγή	1
1.1 Ερευνητικό Πεδίο	1
1.1.1 Ερευνητικά Ερωτήματα	2
1.2 Στόχοι και Προσδοκώμενα Αποτελέσματα	2
1.3 Μέθοδοι Συλλογής Δεδομένων και Μεθοδολογία που Εφαρμόσθηκε	4
1.4 Δομή της Διατριβής	5
2 Ερευνητικό Υποβαθρο	7
2.1 Συναφείς Εργασίες	7
3 Αρχιτεκτονική	18
3.1 Προτεινόμενο Σύστημα IoT Έξυπνων Πόλεων για την Οδική Ασφάλεια	18
3.1.1 Σχήμα	19
3.1.2 Άξονας I: CAV AR	20
3.1.3 Άξονας II: Έξυπνος Σηματοδότης	30
3.1.4 Άξονας III: Υποδομή Νέφους	32
3.1.5 Επεκτασιμότητα	33
3.2 Οπτικοποίηση Αντικειμένων Ενδιαφέροντος	34
3.2.1 Οπτικοποίηση ως Σημείο	34

3.2.2	Οπτικοποίηση ως Τρισδιάστατο Πλαίσιο Οριοθέτησης	35
3.2.3	Οπτικοποίηση ως Δισδιάστατο Φύλλο Ψηφιογραφικών Αντικει- μένων	35
3.2.4	Οπτικοποίηση ως Τρισδιάστατο Μοντέλο	35
3.2.5	Προσέγγιση Οπτικοποίησης	36
4	Προτεινόμενο Σύστημα Επαυξημένης Πραγματικότητας Συνδεδεμένων Αυτόνομων Οχημάτων	38
4.1	Εισαγωγή	38
4.2	Συλλογή Δεδομένων	39
4.3	Εφυΐα Συστήματος	41
4.4	Διαδικασίες Διαγραμμάτων Ροής	42
4.5	Παραδείγματα Λειτουργείας	51
5	Αξιολόγηση Συστήματος	64
5.1	Πειραματική Διάταξη	64
5.2	Αποτελέσματα	65
5.3	Αξιολόγηση	66
5.4	Βελτιστοποιήσεις και Περιορισμοί	71
6	Συμπεράσματα	74
6.1	Συμπεράσματα	74
6.2	Μελλοντικές Έξελξεις	76
	Βιβλιογραφία	77
A	Πηγαίος Κώδικας Βασικού Συστήματος	81

ΚΑΤΑΛΟΓΟΣ ΣΧΗΜΑΤΩΝ

3.1	Η Αρχιτεκτονική του Προτεινόμενου Συστήματος.	20
4.1	Διάγραμμα Κύριας Διεργασίας.	43
4.2	Διάγραμμα Αρχικοποίησης Συστήματος.	44
4.3	Διάγραμμα Επεξεργασίας ανά Καρέ Εικόνας.	45
4.4	Διάγραμμα Ανίχνευσης Αντικειμένων μέσω Μηχανικής Μάθησης. . . .	46
4.5	Διάγραμμα Υπολογισμού Βάθους μέσω LIDAR.	46
4.6	Διάγραμμα Υπολογισμού Εισερχόμενων Αντικειμένων.	47
4.7	Διάγραμμα Εικονικοποίησης 3D Αντικειμένου.	48
4.8	Διάγραμμα Εικονικοποίησης 3D Πλαισίου Οριοθέτησης.	49
4.9	Διάγραμμα Εικονικοποίησης 2D Ψηφιογραφικού Αντικειμένου.	49
4.10	Διάγραμμα Σχεδίασης 2D Ψηφιογραφικού Αντικειμένου.	50
4.11	Προσωποποιημένη Προβολή Επαυξημένης Πραγματικότητας.	51
4.12	Σύγκριση Εικονικοποίησης μέσω 2D Ψηφιογραφικού Αντικειμένου και 3D Πλαισίου Οριοθέτησης I.	52
4.13	Σύνθεση Εικόνων μέσω Απόστασης / Καναλιού Διαφάνειας.	53
4.14	Προσαρμογή Έντασης Προβολής Επαυξημένης Πραγματικότητας. . . .	54
4.15	Εικονικοποίηση Απόστασης / Βάθους μέσω LIDAR.	55
4.16	Προβολή Επιπρόσθετων Πληροφοριών.	56
4.17	Αναγνώριση Διαφορετικών Αντικειμένων Ενδιαφέροντος μέσω Υπολο- γιστικής Όρασης / Μηχανικής Μάθησης.	57
4.18	Ανίχνευση και Εικονικοποίηση Ποδηλάτη.	57
4.19	Ανίχνευση Οχήματος σε Περιοχή Εκτός Οπτικού Πεδίου.	58
4.20	Ανίχνευση Πεζού σε Περιοχή Εκτός Οπτικού Πεδίου.	59
4.21	Ανίχνευση Πεζού σε Περιοχή με Οπτικό Κενό Αντίληψης.	60
4.22	Εικονικοποίηση Πεζού σε Περιοχή Εκτός Οπτικού Πεδίου.	61

4.23 Σύγκριση Εικονικοποίησης μέσω 2D Ψηφιογραφικού Αντικειμένου και 3D Πλαισίου Οριοθέτησης II.	63
5.1 Γράφημα Αποτελεσμάτων.	66

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

5.1 Πίνακας Αποτελεσμάτων Καρέ ανα Δευτερόλεπτο.	66
--	----

ΚΑΤΑΛΟΓΟΣ ΑΛΓΟΡΙΘΜΩΝ

ΠΕΡΙΛΗΨΗ

Βασίλειος Καραΐσκος, Δ.Μ.Σ. στη Μηχανική Δεδομένων και Υπολογιστικών Συστημάτων, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πολυτεχνική Σχολή, Πανεπιστήμιο Ιωαννίνων, 2025.

Ε.Π.Ο.Σ. - προτεινόμενο σύστημα Επαυξημένης Πραγματικότητας για την Οδική ασφάλεια έξυπνων πόλεων Συνδεδεμένων αυτόνομων οχημάτων.

Επιβλέπων: Ιωάννης Φούντος, Καθηγητής.

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η ανάδειξη της, οπτικοποίησης αντικείμενων ενδιαφέροντος, με την χρήση προηγμένου συστήματος Επαυξημένης Πραγματικότητας, με απώτερο σκοπό την ασφάλεια των συνδεδεμένων αυτοματοποιημένων οχημάτων στις έξυπνες πόλεις, στην εποχή της 4ης Βιομηχανικής Επανάστασης, χρησιμοποιώντας την μηχανική μάθηση, καθώς και να προβάλει την πολυδιάστατη συνεισφορά της, στην ενίσχυση της βιωματικής εμπειρίας από την πλευρά του χρήστη. Το ερευνητικό πεδίο της διπλωματικής εργασίας αφορά τα συνδεδεμένα αυτοματοποιημένα οχήματα, το διαδίκτυο των πραγμάτων (IOT), αυτοκινούμενα οχήματα που χρησιμοποιούν Datasheets χρώματος/βάθους, καθώς και την αρκετά προηγμένη τεχνολογία kitti 360, την αποτελεσματική αποθήκευση δεδομένων (NOSQL Databases for IOT) και κάποια πράγματα πάνω στον τομέα της μηχανικής μάθησης, καθώς και ενδιαφέροντες και αρκετά σημαντικούς αλγορίθμους που χρησιμοποιήθηκαν σε παρεμφερή προηγμένα συστήματα.

Μέσα από την συγκεκριμένη διπλωματική εργασία, αφότου πρώτα γίνει μια σύντομη αποτύπωση της κατάστασης στις μέρες μας, αναφορικά με το κομμάτι των οχημάτων και της οδήγησης στους δρόμους, καταδεικνύεται η αναγκαιότητα εξέλιξης του οδικού συστήματος, της πρόληψης, καθώς και της απρόσκοπτης αποφυγής ατυχημάτων, στους ραγδαίους ρυθμούς εξέλιξης της 4ης Βιομηχανικής Επανάστασης. Επιπροσθέτως, υπογραμμίζεται η αξία της μηχανικής μάθησης, πως μέσα από αυτήν το σύστημα μας εκπαιδεύεται και γίνεται έξυπνότερο και κατ'επέκταση και

αποτελεσματικότερο. Ακόμη, επισημαίνεται η ύπαρξη του χρηστοκεντρικού σχεδιασμού, στο πλαίσιο υλοποίησης και συλλογής εμπειριών, περιγράφονται οφέλη της εφαρμογής των αρχών ευχρηστίας, καθώς και της ομαλής πλοηγησιμότητας, καθώς και τα οφέλη της απόκτησης βιωματικών εμπειριών στα πλαίσια της μηχανικής μάθησης.

Τέλος, παρουσιάζεται η εφαρμογή αρχικής επίδειξης λειτουργικότητας καθώς και μια συνοπτική επισκόπηση των βασικών εργαλείων σχεδίασης Επαυξημένης Πραγματικότητας (Inertial Navigation and GPS/GNSS, Lidar, Augment reality user interface) που χρησιμοποιήθηκαν στον σχεδιασμό του συστήματος, προκειμένου το όχημα να ανιχνεύει τα αντικείμενα ενδιαφέροντος κατά την διάρκεια της διαδρομής του, να ενημερώνει τον χρήστη και να εκτελεί λειτουργίες μεταεπεξεργασίας, λήψης αποφάσεων ή αναμετάδοσης σε άλλα οχήματα.

Λέξεις – Κλειδιά: Επαυξημένη Πραγματικότητα, 4η Βιομηχανική Επανάσταση, Μηχανική μάθηση, Ψηφιακός μετασχηματισμός μάθησης, οπτικός σχεδιασμός, τεχνολογία της πληροφορίας και των επικοινωνιών.

EXTENDED ABSTRACT

Vasileios Karaiskos, M.Sc. in Data and Computer Systems Engineering, Department of Computer Science and Engineering, School of Engineering, University of Ioannina, Greece, 2025.

A.R.R.C. - proposed Augmented Reality system for Road safety in smart cities with Connected and autonomous vehicles .

Advisor: Ioannis Fudos, Professor.

The present thesis investigates the enhancement, visualization, and contextual understanding of objects of interest through the integration of an advanced Augmented Reality (AR) system specifically designed to strengthen road safety and situational awareness for connected and autonomous vehicles within the framework of smart cities and the broader paradigm of the Fourth Industrial Revolution (Industry 4.0). By employing machine learning (ML) algorithms and techniques of computer vision, this work demonstrates how AR can bridge the gap between human perception and autonomous decision-making systems, offering real-time spatial awareness, predictive hazard detection, and a more immersive, user-centered driving experience.

The research scope of the thesis spans across several interrelated technological domains, including connected and autonomous vehicles (CAVs), the Internet of Things (IoT), multi-sensor data fusion, and 3D spatial mapping. The system utilizes datasets derived from the KITTI-360 benchmark, particularly leveraging rectified RGB images (Cam0), LiDAR point clouds, and GNSS/OXTS positional data to generate a comprehensive perception model of the driving environment. These multimodal data sources are harmonized through geospatial transformations—converting geodetic coordinates (latitude, longitude, altitude) into Cartesian frames (x, y, z)—to ensure consistent alignment between the visual, spatial, and semantic layers of information. Additionally, NoSQL databases are proposed for scalable, low-latency data management, facilitating the continuous flow of sensory information within an IoT infrastructure.

At the core of this research lies the CAV AR architectural framework, which integrates sensory subsystems (LiDAR, RGB cameras, and GNSS) with a real-time Augmented Reality User Interface (AR UI). This interface functions as a dynamic bridge between the computational backbone and the human operator, visualizing critical traffic and environmental data directly within the driver's field of view. Through adaptive overlay and alpha-blending techniques, the system maintains high visual clarity and minimizes cognitive overload. The machine learning inference module, implemented via the YOLOv11 object detection model, identifies and classifies objects with high confidence, while the LiDAR depth maps provide precise spatial localization. Together, these modules enable the driver or the autonomous agent to perceive, evaluate, and react to potential hazards—including those beyond the line of sight—thus mitigating blind spots and enhancing proactive decision-making.

The proof-of-concept (PoC) developed for this study validates the theoretical framework by demonstrating real-time object detection, rendering, and AR-based visualization in driving scenarios. Performance analysis (in frames per second) revealed a clear trade-off between visual realism and computational efficiency, emphasizing that 2D sprite-based rendering and 3D bounding box visualization offer an optimal balance between interpretability and system responsiveness—especially in low-power embedded platforms.

From a design and human-factors perspective, the thesis underscores the importance of user-centered design (UCD) as a guiding principle in AR-based human-machine interfaces. By tailoring visual density, transparency, and spatial layering to user preferences and environmental context, the system ensures a non-intrusive yet informative experience. Moreover, the integration of experiential learning paradigms within the AR interface allows users to adaptively learn from their environment, enhancing long-term situational awareness and promoting safer driving behaviors.

Beyond technical implementation, this research aligns with the emerging vision of Industry 4.0 transportation ecosystems, where digitalization, automation, and human-AI collaboration converge. The proposed AR architecture not only enhances road safety and efficiency but also contributes to the digital transformation of mobility, setting a foundation for collaborative vehicle-to-vehicle (CAV2CAV) communication, cloud-based data sharing, and predictive analytics for urban traffic management.

In conclusion, this thesis presents a holistic synthesis of Augmented Reality, Machine Learning, and Human-Centered Interaction Design in the context of smart

mobility. The findings demonstrate that AR has the potential to redefine the driving experience by merging real and virtual perception spaces, fostering proactive safety mechanisms, and paving the way toward a new paradigm of intelligent, connected, and human-aware transportation systems.

Keywords: Augmented Reality, Fourth Industrial Revolution, Machine Learning, Digital Learning Transformation, Visual Design, Information and Communication Technologies.

ΚΕΦΑΛΑΙΟ 1

ΕΙΣΑΓΩΓΗ

1.1 Ερευνητικό Πεδίο

1.2 Στόχοι και Προσδοκώμενα Αποτελέσματα

1.3 Μέθοδοι Συλλογής Δεδομένων και Μεθοδολογία που Εφαρμόστηκε

1.4 Δομή της Διατριβής

1.1 Ερευνητικό Πεδίο

Τα Συστήματα Επαυξημένης Πραγματικότητας (AR), μηδενός εξαιρουμένου αποτελούν σύγχρονες τεχνολογικές λύσεις, οι οποίες παρέχουν την δυνατότητα της απρόσκοπτης ενοποίησης μεταξύ της εικονικής πραγματικότητας και του πραγματικού κόσμου. Με το πέρασμα των χρόνων η επίδραση της τεχνολογίας στην ζωή μας αυξάνεται όλο και περισσότερο, πράγμα το οποίο έχει ως άμεση απόρροια την όλο και μεγαλύτερη επίδραση και εφαρμογή της Επαυξημένης Πραγματικότητας σε όλο ένα και περισσότερους κλάδους σε παγκόσμιο επίπεδο. Ακόμη, με την ραγδαία εξέλιξη της τεχνολογίας προκύπτουν όλο ένα και περισσότερες προκλήσεις και ως εκ τούτου τα εξειδικευμένα συστήματα Επαυξημένης Πραγματικότητας, χρησιμοποιούνται όλο και περισσότερο και με το πέρασμα του χρόνου, επί του πρακτέου αποτελούν τον ακρογωνιαίο λίθο για την βελτίωση της παραγωγικότητας, καθώς και για την βελτίωση της εμπειρίας και της οπτικής πλευράς του χρήστη γενικότερα. Η ενσωμάτωση Συστημάτων Επαυξημένης Πραγματικότητας (AR) στην ζωή μας, τα οποία εκπαιδεύονται όλο και περισσότερα και εξελίσσονται συλλέγοντας εμπειρίες

μέσω της μηχανικής μάθησης, μας παρέχει όλο και περισσότερα οφέλη, μεταβαίνοντας σε έναν καθολικό ψηφιακό μετασχηματισμό μάθησης. Η συνεχής εξέλιξη της τεχνολογίας και των υποσυστημάτων της έχει ως άμεση απόρροια την σταδιακή διαμόρφωση ενός νέου είδους εκπαίδευσης και δια βίου κατάρτισης όλων των ανθρώπων, σε όλες τις βαθμίδες όπως για παράδειγμα, μαθητές, φοιτητές, εργαζόμενους κ.α.

Η συγκεκριμένη Διπλωματική Εργασία εστιάζει στην ανάδειξη της χρήσης, της οπτικοποίησης αντικείμενων ενδιαφέροντος, με την χρήση προηγμένου συστήματος Επαυξημένης Πραγματικότητας στην σημερινή εποχή, στην διερεύνηση της πολυδιάστατης συνεισφοράς τους στην διαμόρφωση της εμπειρίας από την οπτική γωνία του απλού χρήστη, καθώς και στην διαδικασία της μετάβασης σε ένα νέο μοντέλο οδικής κυκλοφορίας και ασφάλειας.

1.1.1 Ερευνητικά Ερωτήματα

Ειδικότερα, η παρούσα Διπλωματική Εργασία αναδεικνύει κάποια ερευνητικά ερωτήματα όπως:

- Πως εξελίσσεται η βιωματική εμπειρία του χρήστη στο πλαίσιο της χρήσης της Επαυξημένης Πραγματικότητας, καθώς και την σημασία του χρηστοκεντρικού σχεδιασμού στην απόκτηση εμπειριών μέσω της μηχανικής μάθησης;
- Ποιες είναι οι βασικές προκλήσεις, τα οφέλη και οι μελλοντικές προοπτικές εφαρμογής της Επαυξημένης Πραγματικότητας στον τομέα της οδικής ασφάλειας και της εμπειρίας του χρήστη;
- Ποια είναι η συμβολή της Επαυξημένης Πραγματικότητας στη μετάβαση προς ένα νέο μοντέλο οδικής κυκλοφορίας και ασφάλειας, και πώς μπορεί να ενισχύσει την εκπαίδευση και τη λήψη αποφάσεων των χρηστών στον δρόμο;

1.2 Στόχοι και Προσδοκώμενα Αποτελέσματα

Σκοπός της παρούσας διπλωματικής αποτελεί η καταγραφή και η ανάλυση συγκεκριμένων νέων τεχνολογιών, καθώς και η χαρτογράφηση των αποτελεσμάτων τους, αναλογικά με τις αναδυόμενες ανάγκες, στο φάσμα αξιοποίησης και ενσωμάτωσης της επαυξημένης πραγματικότητας.

Επιπροσθέτως, μέσα από την έρευνα που έχει διεξαχθεί, επιδιώκεται η εκτίμηση των θεμελιωδών προκλήσεων που προκύπτουν στις μέρες μας, επικεντρώνοντας ένα μεγάλο μέρος του ερευνητικού μας ενδιαφέροντος σε δεξιότητες, λειτουργίες και αρχιτεκτονικές, που αποτελούν κομβικούς παράγοντες για την εδραίωση της επαυξημένης πραγματικότητας και της μηχανικής μάθησης, καθώς και για την εξέλιξη τους στην πορεία του χρόνου.

Μέσα από την συγκεκριμένη διπλωματική εργασία, γίνεται εύκολα αντιληπτό το γεγονός ότι για την ορθότερη μετάβαση στην νέα εποχή, εκτός από την εξέλιξη του τομέα της τεχνολογίας πράγμα το οποίο αποτελεί καταλύτη, είναι και ο μεθοδευμένος εκσυγχρονισμός της μαθησιακής διαδικασίας, με την μηχανική μάθηση, καθώς επαυξημένη πραγματικότητα, η οποία αποτελεί έναν πάρα πολύ κομβικό παράγοντα, συμβολής προσαρμοστικότητας, εξειδίκευσης και ομαλής μετάβασης σε μια νέα και αρκετά εξελιγμένη πραγματικότητα, με εικονικό τρισδιάστατο και πολυμεσικό περιεχόμενο, το οποίο θα αποτελεί με μαθηματική ακρίβεια το καθιερωμένο σύστημα κυκλοφορίας και εντοπισμού οχημάτων και εμποδίων του μέλλοντος.

Επίσης, σε αυτό το νέο τεχνολογικά εξελισσόμενο μοντέλο, η εξατομικευμένη μαθησιακή εμπειρία βρίσκεται στο επίκεντρο και ο στόχος της κατάκτησης και της κατανόησης της γνώσης και της εξέλιξης των δυνατοτήτων, στηρίζεται κατά κύριο λόγο σε βιωματικές μεθόδους που κυριαρχούνται μέσω της επαυξημένης πραγματικότητας και της μηχανικής μάθησης. Ακόμη, πραγματοποιείται διερεύνηση της δυνατότητας συνεισφοράς, εξετάζοντας την ευχρηστία της επαυξημένης πραγματικότητας, σε διάφορες εφαρμογές και πτυχές της καθημερινής μας ζωής από όμορους επιστημονικούς κλάδους όπως για παράδειγμα της γεωργίας ακριβείας και της βιομηχανικής συντήρησης, με ταυτόχρονη αξιολόγηση της απόρροιας των μαθησιακών της αποτελεσμάτων. Εν κατακλείδι, η συγκεκριμένη διπλωματική εργασία έχει μια σειρά από συγκεκριμένους στόχους, μέσα από τους οποίους επιδιώκονται τα εξής:

- Να παρουσιάσουν αντιπροσωπευτικά παραδείγματα διαφόρων εφαρμογών επαυξημένης πραγματικότητας από ποικίλους τομείς της καθημερινότητας μας και κατά κύριο λόγο από τον χώρο της βιομηχανίας.
- Να ορισθεί με επιστημονικό τρόπο η έννοια της επαυξημένης πραγματικότητας, καθώς και των πολλαπλών πλεονεκτημάτων της.
- Να αξιολογηθούν τα πολλαπλά πλεονεκτήματα της βιωματικής μάθησης - μηχανικής μάθησης, παρουσιάζοντας την συνολική εκπαιδευτική τους αξία, την

επίδραση των νέων αυτών τεχνολογιών και μεθόδων εκπαίδευσης, ταυτόχρονα με την εμπειρία του χρήστη και την συμβολή της επαυξημένης πραγματικότητας.

- Να εξετασθεί η συμβολή των γραφικών στην σχεδίαση εμπειριών του απλού χρήστη, στην επαυξημένη πραγματικότητα και στις νέες αυτές τεχνολογίες.
- Να παρουσιασθούν και να αναλυθούν οι τεχνολογίες των συνδεδεμένων αυτόνομων οχημάτων, της μηχανικής μάθησης και υπολογιστικής όρασης, της επαυξημένης πραγματικότητας, της αξιοποίησης γεωδευτικών πληροφοριών και επεξεργασίας αυτών, της αλληλεπίδρασης χρήστη /μηχανής καθώς και μηχανής με μηχανή.
- Να πραγματοποιηθεί εκτεταμένη ανάλυση των χαρακτηριστικών σχεδίασης, επεξεργασίας και διαχείρισης δεδομένων στην τεχνολογία των συνδεδεμένων αυτόνομων οχημάτων με αυτόματα αναγνώριση κινδύνου.
- Να αναλυθούν τα τεχνικά τους χαρακτηριστικά και ο τρόπος λειτουργίας τους.
- Να εξετασθεί η αναγκαιότητα και η χρησιμότητα της οπτικοποίησης των αντικειμένων, κάνοντας χρήση προηγμένων συστημάτων επαυξημένης πραγματικότητας για την ασφάλεια των συνδεδεμένων αυτοματοποιημένων οχημάτων στις έξυπνες πόλεις.

Επισημαίνεται ότι η παρούσα μελέτη πέτυχε να ανταποκριθεί αποτελεσματικά στους στόχους που τέθηκαν, παρέχοντας σαφείς και τεκμηριωμένες απαντήσεις και στα τρία ερευνητικά ερωτήματα. Μέσα από την ανάλυση, την πειραματική διαδικασία και την αξιολόγηση των αποτελεσμάτων, επιβεβαιώνεται ότι η έρευνα εκπλήρωσε τον αρχικό της σκοπό και συνέβαλε ουσιαστικά στην κατανόηση και διερεύνηση του υπό μελέτη αντικειμένου.

1.3 Μέθοδοι Συλλογής Δεδομένων και Μεθοδολογία που Εφαρμόσθηκε

Κατά την διάρκεια της εκπόνησης της παρούσας διπλωματικής εργασίας συλλέχθηκαν αρκετές πληροφορίες, τόσο από ξενόγλωσσα όσο και από ελληνικά άρθρα

που είχαμε στην διάθεση μας, αλλά και από βιβλιογραφίες γενικότερα, μέσω αναζήτησης, σε ψηφιακές βιβλιοθήκες, αλλά κατά κύριο λόγο στο διαδίκτυο. Οι θεματολογίες που τίθενται προς ανάλυση και προς μελέτη στην συγκεκριμένη εργασία εστιάζουν κατά κύριο λόγο στην επαυξημένη πραγματικότητα (AR), καθώς και στις εφαρμογές τις, στην τέταρτη Βιομηχανική Επανάσταση και στις απόρροιες της, στην σχεδίαση εμπειρίας χρήστη μέσω της μηχανικής μάθησης. Επίσης, η ερευνητική διαδικασία που ακολουθείται αποτελεί έναν συνδυασμό βιβλιογραφικής και διαδικτυακής ανασκόπησης διακεκριμένων επιστημονικών πηγών, για όλα τα θεωρητικά ζητήματα που παρατίθενται.

Επιπλέον, στο πλαίσιο της παρούσας έρευνας υλοποιήθηκε τμήμα της προτεινόμενης αρχιτεκτονικής του συστήματος, όπως αυτή περιγράφεται στον Άξονα I και αναλύεται εκτενώς στην αντίστοιχη ενότητα. Η υλοποίηση πραγματοποιήθηκε σε Python 3.8, αξιοποιώντας τα εργαλεία ανάπτυξης που παρείχε η ομάδα δημιουργίας του συνόλου δεδομένων KITTI-360. Ο πλήρης κώδικας της εφαρμογής παρατίθεται στο Παράρτημα Α.

1.4 Δομή της Διατριβής

Η παρούσα διπλωματική διατριβή διαρθρώνεται σε έξι κεφάλαια, τα οποία οδηγούν τον αναγνώστη από τη θεωρητική θεμελίωση και τη σύγχρονη έρευνα, στην ανάπτυξη της προτεινόμενης αρχιτεκτονικής και την πειραματική υλοποίηση, καταλήγοντας στην αξιολόγηση των αποτελεσμάτων.

Έπειτα της ολοκλήρωσης του Κεφαλαίου 1, το Κεφάλαιο 2 (Ερευνητικό Υπόβαθρο) εξετάζει τη συναφή βιβλιογραφία και τις σύγχρονες τεχνολογίες, παρέχοντας μια επισκόπηση συναφών εργασιών που εστιάζουν σε συστήματα AR οχημάτων και IoT για την οδική ασφάλεια.

Το Κεφάλαιο 3 (Αρχιτεκτονική) περιγράφει λεπτομερώς την προτεινόμενη ολοκληρωμένη αρχιτεκτονική συστήματος IoT Έξυπνων Πόλεων, η οποία διαρθρώνεται στους τρεις κύριους Άξονες: τον Άξονα I (CAV AR), τον Άξονα II (Έξυπνος Σηματοδότης) και τον Άξονα III (Υποδομή Νέφους), αναλύοντας παράλληλα τις προσεγγίσεις για την οπτικοποίηση των αντικειμένων ενδιαφέροντος. Ακολουθεί το Κεφάλαιο 4 (Προτεινόμενο Σύστημα Επαυξημένης Πραγματικότητας Συνδεδεμένων Αυτόνομων Οχημάτων), το οποίο εστιάζει στην πειραματική υλοποίηση του

Άξονα I, αναλύοντας τα στάδια της Συλλογής Δεδομένων, της Εφύιας Συστήματος, τις Διαδικασίες Διαγραμμάτων Ροής (απόκτησης, επεξεργασίας, απόδοσης) και παρουσιάζοντας σχετικά Παραδείγματα Λειτουργίας.

Τέλος, το Κεφάλαιο 5 (Αξιολόγηση Συστήματος) παρουσιάζει την Πειραματική Διάταξη, τα Αποτελέσματα και την Αξιολόγηση του πρωτοτύπου, η οποία περιλαμβάνει ποσοτικές μετρικές απόδοσης και ποιοτική μελέτη χρηστών για το AR UI, ολοκληρώνοντας με την ανάλυση των βελτιστοποιήσεων και των περιορισμών. Η διατριβή ολοκληρώνεται με το Κεφάλαιο 6 (Συμπεράσματα), το οποίο συνοψίζει τα βασικά επιτεύγματα της εργασίας και διατυπώνει προτάσεις για Μελλοντικές Εξελίξεις και επεκτάσεις του συστήματος.

ΚΕΦΑΛΑΙΟ 2

ΕΡΕΥΝΗΤΙΚΟ ΥΠΟΒΑΘΡΟ

2.1 Συναφείς Εργασίες

2.1 Συναφείς Εργασίες

Η υπάρχουσα βιβλιογραφία περιλαμβάνει πληθώρα ερευνητικών προσεγγίσεων που μελετούν το πρόβλημα της οδικής ασφάλειας στον τομέα των έξυπνων πόλεων αξιοποιώντας τις καινοτόμες τεχνολογίες στον τομέα της αυτοκίνησης καθώς και στον τομέα της επαυξημένης πραγματικότητας. Ακολουθώντας, γίνεται αναφορά σε κρίσιμους τομείς που αποτελούν πηγαία γνώση για την παρούσα εργασία: Συνδεδεμένα Αυτοματοποιημένα Οχήματα (Connected and Autonomous Vehicles - CAVs), Επαυξημένη Πραγματικότητα (Augmented Reality - AR), αλληλεπίδραση χρήστη - μηχανής, οδική ασφάλεια, διασυνδεσιμότητα και υπολογισμοί ακμής / νέφους, παραγωγή, διαχείριση και κατανάλωση Μεγαδεδομένων (Big Data), υπολογιστική όραση και μηχανική / βαθιά μάθηση και εν κατακλείδι της ολιστικής και ολοκληρωμένης σχεδίασης πληροφοριακών συστημάτων Διαδικτύου των Πραγμάτων (Internet of Things - IoT) για την Βιομηχανία 4.0 με εφαρμογή τα βιομηχανικά περιβάλλοντα, τις Έξυπνες Πόλεις (Smart Cities), την γεωργία ακριβείας και ούτω καθεξής.

Στον τομέα των συνδεδεμένων αυτοματοποιημένων οχημάτων, προηγμένων τεχνολογιών μια αρκετά εκτεταμένη έρευνα έχει πραγματοποιηθεί από τους Aslam et al.(2023)[1] , η οποία πραγματεύεται κατά κύριο λόγο την ενσωμάτωση των Connected and Autonomous Vehicles (CAVs) στο Metaverse, με κεντρικούς άξονες τα

Ψηφιακά Δίδυμα (Digital Twins) και τη Θεωρία Παιγνίων (Game Theory). Μέσα από την συγκεκριμένη εργασία μπορούμε να κατανοήσουμε τον τρόπο με τον οποίο τα ψηφιακά δίδυμα δρουν ως άκρως σημαντικός συνδετικός κρίκος ανάμεσα στον φυσικό και τον ψηφιακό κόσμο, κάνοντας εφικτή την ρεαλιστική προσομοίωση και τον έλεγχο οχημάτων σε πολυεπίπεδες μετα-πραγματικότητες. Μια από τις πιο σημαντικές συνεισφορές του συγκεκριμένου άρθρου, αποτελεί η πρόταση μιας πολυεπίπεδης αρχιτεκτονικής Metaverse για CAVs, η οποία ενσωματώνει τεχνολογίες όπως το edge computing, η τεχνητή νοημοσύνη, η επαυξημένη/εικονική πραγματικότητα (AR/VR/XR) και το blockchain, ώστε να διασφαλίζεται η απρόσκοπτη διαλειτουργικότητα, η ασφάλεια, και η αξιοπιστία του συστήματος. Παράλληλα, η συγκεκριμένη έρευνα εντοπίζει και καταγράφει αναλυτικά τις τεχνικές θεωρίες των παιγνίων, οι οποίες εφαρμόζονται στο φάσμα της συνεργασίας μεταξύ οχημάτων, της βελτιστοποίησης πόρων, την βέλτιστη κατανομή ισχύος και την δρομολόγηση. Επιπροσθέτως, στην συγκεκριμένη εργασία διακυβεύονται οι σύγχρονες προκλήσεις σε θέματα ιδιωτικότητας, αξιοπιστίας και κυβερνοασφάλειας και τίθενται κατευθυντήριες ερευνητικές γραμμές έρευνας υποστήριξης αποκεντρωμένων, ευφυών και βιώσιμων μεταφορών.

Ένα ακόμη αρκετά ενδιαφέρον άρθρο των Brik et al. [2] το οποίο είναι μέρος του τεύχους IEEE Internet of Things Magazine (Μάρτιος 2023), με τίτλο "Multi-Access Networking for Extended Reality and Metaverse" πραγματεύεται τις τρέχουσες εξελίξεις, τις τεχνολογικές απαιτήσεις, καθώς και τις υπάρχουσες προκλήσεις για την υλοποίηση του Metaverse μέσω δικτύων πολλαπλής πρόσβασης τεχνολογιών XR (Extended Reality). Οι συγγραφείς του εν λόγω επιστημονικού άρθρου δίνουν έμφαση στην XR, παρουσιάζοντας την ως ένα υπερσύνολο, το οποίο περιλαμβάνει την επαυξημένη πραγματικότητα την μικτή και την εικονική, αξιολογώντας το Metaverse ως την επόμενη γενιά του διαδικτύου, στην οποία η επικοινωνία ανάμεσα στον ψηφιακό και στον πραγματικό κόσμο γίνεται αδιάλειπτα και πολυαισθητηριακά. Επιπροσθέτως, δίνεται έμφαση στο κομμάτι του Internet of Things (IoT), των υπολογιστικών υποδομών αιχμής (edge/cloud), της τεχνητής νοημοσύνης, του blockchain και των δικτύων 5G/6G, των οποίων η συνδυαστική λειτουργία διαμορφώνει σε αρκετά μεγάλο βαθμό το τεχνολογικό υπόβαθρο του μετασύμπαντος. Επίσης, το συγκεκριμένο άρθρο παραθέτει την αναγκαιότητα της διεπιστημονικής συνεργασίας, με τεχνικές της θεωρίας των παιγνίων, της μηχανικής μάθησης και της βελτιστοποίησης διαλειτουργικών και ασφαλών εμπειριών. Στο τελευταίο μέρος,

παρατίθεται μια επισκόπηση έξι ερευνητικών μελετών του τεύχους, οι οποίες περιλαμβάνουν, την εξέλιξη των πρότυπων 5G, αρχιτεκτονική του Metaverse, εφαρμογές XR με drones, χρήση LiDAR για φυσικά προσδεδεμένες ψηφιακές αναπαραστάσεις, και την αξιοποίηση NFTs για ιδιοκτησία και ασφάλεια.

Αξιοσημείωτη συμβολή στον τομέα της συνδυασμένης χρήσης επαυξημένης πραγματικότητας (AR) και υποδομών υπολογιστικού νέφους για ευφυή οχήματα παρουσιάζεται στο έργο των Zhou et al.[3] με τίτλο "Enhanced Augmented Reality Applications in Vehicle-to-Edge Networks". Στην συγκεκριμένη επιστημονική εργασία, παρατίθεται η αρχιτεκτονική EARVE (Enhanced Augmented Reality Vehicle-to-Edge), η οποία αποτελεί ένα αρκετά εξελιγμένο σύστημα επικοινωνίας Vehicle-to-Edge (V2E), το οποίο έχει ως απώτερο στόχο την τελειοποίηση της χρονοευσαισθησίας άκρως σημαντικών εφαρμογών σε δίκτυα οχημάτων. Η EARVE κάνοντας χρήση των υπολογιστικών πόρων που διαθέτουν τα οχήματα (edge servers), μειώνει σημαντικά την χρονοκαθυστέρηση, πολλές φορές ακόμη και σε ποσοστό της τάξης του 26%, συγκριτικά με άλλα παραδοσιακά συστήματα, τα οποία βρίσκονται στο cloud. Παράλληλα, επιτυγχάνει αρκετά μεγάλη μείωση του όγκου των δεδομένων, προς το cloud σε ποσοστό της τάξης του 98%, πράγμα το οποίο επί του πρακτέου αποτελεί αποσυμφόρηση του δικτύου. Χαρακτηριστικό παράδειγμα αποτελεί η εφαρμογή "View Share", η οποία αποτελεί μια AR κατασκευή εντοπισμού και αναπαράστασης διαφόρων αντικειμένων μέσα στον χώρο του οχήματος, ακόμη και όταν αυτά δεν είναι στο πεδίο ορατότητας του οδηγού. Η διαδικασία εντοπισμού των αντικειμένων καθίσταται εφικτή με YOLOv3 και GPU επιτάχυνση, ενώ η απόδοση του συστήματος αξιολογείται με χρήση δεδομένων από το Λονδίνο του Ηνωμένου Βασιλείου. Ακόμη, το EARVE αποτελείται από πολυεπίπεδη αρχιτεκτονική με Tier 1 και Tier 2 edge servers ενσωματώνοντας μηχανισμούς για Ιδιωτικότητα, ασφάλεια και αποδοτική δρομολόγηση ειδοποιήσεων. Ως εκ τούτου η συγκεκριμένη επιστημονική εργασία αποτελεί ένα πρότυπο εφαρμογής edge computing δικτύων συνδεδεμένων αυτόνομων οχημάτων, με την ανάδειξη της επεξεργασίας δεδομένων κοντά στην πηγή χρησιμοποιώντας την τεχνολογία AR, καθιστώντας την οδηγική εμπειρία αρκετά ασφαλέστερη.

Στην προσπάθεια ενίσχυσης της διαφάνειας και της αλληλεπίδρασης μεταξύ αυτόνομων οχημάτων και πεζών, η εργασία των Cheng et al.(2024)[4] προτείνει ένα σύστημα επαυξημένης πραγματικότητας (AR) που επιτρέπει στους πεζούς να παρακολουθούν σε πραγματικό χρόνο τη δυναμική συμπεριφορά αυτόνομων οχημάτων

μέσω φορητών συσκευών. Το σύστημα εξάγει δεδομένα από το λογισμικό Autoware (π.χ. αισθητηριακές αντιλήψεις, προγραμματισμένες τροχιές, κινηματικά χαρακτηριστικά) και τα μεταδίδει μέσω cloud σε AR εφαρμογή σε Smartphone, η οποία προβάλλει πληροφορίες όπως τροχιακά βέλη, αναγνωρισμένα εμπόδια και καταστάσεις οχήματος στο οπτικό πεδίο του πεζού. Η εφαρμογή, συμβατή με ARKit/ARCore, αξιολογήθηκε πειραματικά στο Πανεπιστήμιο του Τόκιο, όπου οι συμμετέχοντες ανέφεραν σημαντική αύξηση στην αντίληψη ασφάλειας και κατανόηση της πρόθεσης του οχήματος, επισημαίνοντας την χρησιμότητα της AR ως διεπαφή επικοινωνίας V2P (Vehicle-to-Pedestrian). Πρακτικά, η εργασία αυτή συμβάλλει ουσιαστικά στο πεδίο της ανθρώπινης διασύνδεσης με αυτόνομα συστήματα μετακίνησης, αναδεικνύοντας τον ρόλο της επαυξημένης πραγματικότητας ως εργαλείο για βελτίωση της εμπιστοσύνης, της προβλεψιμότητας και της ασφάλειας σε μικτές συνθήκες κυκλοφορίας.

Παράλληλα, στα πλαίσια της ενίσχυσης της οδικής ασφάλειας και της εμπειρίας οδήγησης, το άρθρο των Wang et al. [5] προτείνει ένα προηγμένο σύστημα υποβοήθησης οδηγού (ADAS) που ενσωματώνει τεχνολογίες επαυξημένης πραγματικότητας (AR) και διασυνδεδεμένων οχημάτων (Connected Vehicles). Ειδικότερα, η συγκεκριμένη επιστημονική εργασία εστιάζει στην ανάπτυξη ενός συστήματος που προβάλλει κρίσιμες πληροφορίες κυκλοφορίας και προειδοποιήσεις για κινδύνους απευθείας στο οπτικό πεδίο του οδηγού, αξιοποιώντας οθόνες τύπου head-up display (HUD) ή AR γυαλιά. Η αρχιτεκτονική, η οποία χρησιμοποιείται βασίζεται σε συνδυασμό δεδομένων από αισθητήρες (κάμερες, LIDAR, ραντάρ), μονάδες V2X επικοινωνίας, καθώς και τεχνικές αναγνώρισης αντικειμένων σε πραγματικό χρόνο (real time). Ακόμη, το σύστημα διαθέτει δυνατότητα ειδοποίησης για επικείμενες συγκρούσεις, την ύπαρξη πεζών, εμποδίων ή αλλαγές στην κυκλοφορία, ενώ ταυτόχρονα προσφέρει AR πλοήγηση, καθώς και οδηγίες πλοήγησης. Η εργασία αναδεικνύει τη δυναμική της επαυξημένης πραγματικότητας ως εργαλείο ενίσχυσης της αντίληψης του οδηγού, επιτυγχάνοντας καλύτερη επίγνωση της κατάστασης (situational awareness) και βελτιωμένο χρόνο αντίδρασης σε κρίσιμες συνθήκες. Το άρθρο αποτελεί μία από τις πιο ολοκληρωμένες προσεγγίσεις στον συνδυασμό AR με ADAS, τονίζοντας τον ρόλο των έξυπνων και διασυνδεδεμένων οχημάτων στην επόμενη γενιά μεταφορών.

Μια ακόμη αξιοσημείωτη επιστημονική δουλειά στο κομμάτι της επαυξημένης πραγματικότητας στα συνδεδεμένα αυτόνομα οχήματα (CAVs), αποτελεί η εργα-

σία των Sharma et al. [6] η οποία παρουσιάστηκε στο συνέδριο IEEE IC2PCT 2024. Στην συγκεκριμένη επιστημονική εργασία, παρατίθενται οι τρόποι η επαυξημένη πραγματικότητα δύναται να ενισχύσει την εμπειρία οδήγησης, την ασφάλεια και την αλληλεπίδραση μεταξύ οδηγού και οχήματος, προσφέροντας ρεαλιστική, προσαρμοσμένη και διαδραστική πληροφόρηση σε πραγματικό χρόνο. Οι κάτοχοι της συγκεκριμένης εργασίας εξετάζουν ενδελεχώς ένα πλήρες φάσμα τεχνολογιών που υποστηρίζουν την υλοποίηση αυτών των συστημάτων, όπως τα δίκτυα 5G/IoT, το edge/cloud computing, οι αισθητήρες LIDAR και κάμερες, καθώς και αλγόριθμοι μηχανικής μάθησης για την αναγνώριση αντικειμένων και καταστάσεων. Παρουσιάζεται επίσης αναλυτική επισκόπηση της βιβλιογραφίας, η οποία καλύπτει AR HUDs για οδηγούς και επιβάτες, εφαρμογές σε urban navigation, on boarding σε αυτόνομα οχήματα, και αξιολόγηση της εμπιστοσύνης και αποδοχής από τους χρήστες. Παράλληλα, υπογραμμίζονται οι πολλαπλές δυνατότητες της επαυξημένης πραγματικότητας, στο κομμάτι της προσφοράς εξελιγμένων λειτουργιών πλοήγησης, προειδοποιήσεων συντήρησης και ασφαλείας σε συνδυασμό με την υπολογιστική ισχύ, την λεπτομερή απεικόνιση, καθώς και την διαλειτουργικότητα. Συμπερασματικά διαφαίνεται ότι επαυξημένη πραγματικότητα είναι μία από τις πλέον υποσχόμενες τεχνολογίες για την επόμενη γενιά έξυπνων μεταφορών, συνδυάζοντας καινοτομία, ασφάλεια, καθώς και εξατομικευμένη εμπειρία.

Επιπροσθέτως, μία από τις πλέον θεμελιώδεις και καινοτόμες προσεγγίσεις στον χώρο της επαυξημένης πραγματικότητας για αυτόνομα συνδεδεμένα οχήματα παρουσιάζεται από τους Qiu et al. [7] στο πλαίσιο της εργασίας "AVR: Augmented Vehicular Reality", που δημοσιεύθηκε στο ACM MobiSys 2018. Στην εργασία αυτή εισάγεται η έννοια της Επαυξημένης Οχηματικής Πραγματικότητας (Augmented Vehicular Reality - AVR), η οποία επιτρέπει την επέκταση της οπτικής αντίληψης ενός οχήματος μέσω της ανταλλαγής αισθητηριακών δεδομένων με γειτονικά οχήματα. Η αρχιτεκτονική AVR βασίζεται στη συγχώνευση point clouds από 3D αισθητήρες (π.χ. stereo vision), εντοπισμού οχημάτων με τεχνικές SLAM και στην υλοποίηση μηχανισμών για αποδοτική μετάδοση μόνο των κρίσιμων (δυναμικών) πληροφοριών. Μέσω έξυπνων στρατηγικών μετασχηματισμού προοπτικής, dead-reckoning και συνεργατικής λήψης αποφάσεων, το σύστημα επιτυγχάνει χαμηλή καθυστέρηση (<100ms), υψηλή ακρίβεια εντοπισμού (έως 20 cm) και σημαντική μείωση του απαιτούμενου εύρους ζώνης. Η συγκεκριμένη επιστημονική εργασία αναδεικνύει την αξία της συνεργατικής επαυξημένης πραγματικότητας ως εργαλείου για την ενίσχυση της ασφάλ-

λεια και της αποτελεσματικότητας σε πολύπλοκα οδηγικά περιβάλλοντα, ιδιαίτερα σε καταστάσεις περιορισμένης ορατότητας ή πυκνής κυκλοφορίας. Η συνεισφορά αυτή θέτει τα θεμέλια για μελλοντικά συστήματα ADAS αυτόνομης οδήγησης που θα βασίζονται όχι μόνο σε τοπική αντίληψη, αλλά και σε καταναεμημένη συλλογική όραση μεταξύ οχημάτων.

Τέλος επί του τομέα των συνδεδεμένων αυτόνομων οχημάτων, στην επιστημονική εργασία των Wang et al.[8] με τίτλο: “Metamobility: Connecting Future Mobility with Metaverse” παρατίθεται ο όρος Metamobility, ο οποίος πρακτικά παρουσιάζει την ολιστική ενσωμάτωση των τεχνολογιών του metaverse (όπως XR, Digital Twins, Blockchain, Edge AI) σε συνδεδεμένα αυτόνομα οχήματα (CAVs), με απώτερο σκοπό την δημιουργία μιας προσωποποιημένης, καθηλωτικής και χωρίς τριβές εμπειρίας μετακίνησης. Ειδικότερα, η συγκεκριμένη επιστημονική μελέτη αναλύει ένα πλήρες αρχιτεκτονικό μοντέλο του metamobility, το οποίο λειτουργεί σύμφωνα με τρεις θεμελιακές αρχές οι οποίες είναι: Οι φυσικές υποδομές (οχήματα, XR συσκευές), το οικοσύστημα ψηφιακής διάδρασης, όπου κάθε φυσική οντότητα αποκτά το δικό της ψηφιακό avatar και τις τεχνολογικές πλατφόρμες (όπως δίκτυα 6G, edge/cloud υπολογιστική, AI). Παρουσιάζονται δύο χαρακτηριστικά παραδείγματα εφαρμογών, οι Tactile Live Maps, οι οποίοι εμπλουτίζουν την χαρτογράφηση με πολυεπίπεδα φυσικά και εικονικά δεδομένα για την υποστήριξη της αυτόνομης και διαδραστικής οδήγησης και τα Meta-Empowered ADAS, τα οποία αποτελούν προηγμένα συστήματα υποβοήθησης οδηγού που αξιοποιούν δεδομένα από πολλαπλές πηγές και προσφέρουν πληροφορίες μέσω AR Head-Up Displays.

Στον τομέα των Μεγαδεδομένων για τον κλάδο της αυτοματισμένης οδήγησης στις Έξυπνες Πόλεις, μια από τις πιο πλήρεις και συστηματικές καταγραφές συνόλων δεδομένων για εφαρμογές υπολογιστικής όρασης αποτελεί η επισκόπηση των Lopez et al.[9], στην οποία παρουσιάζονται και αναλύονται 231 δημόσια διαθέσιμα RGB-D datasets, δηλαδή σύνολα δεδομένων που περιλαμβάνουν ταυτόχρονα πληροφορία χρώματος (RGB) και βάθους (Depth). Η συγκεκριμένη επιστημονική εργασία καλύπτει μια ευρεία γκάμα εφαρμογών, από τρισδιάστατη κατανόηση σκηνών και αντικειμένων έως αναγνώριση ανθρώπινων δραστηριοτήτων και ιατρική απεικόνιση, και περιλαμβάνει γνωστά datasets όπως τα KITTI, NYUv2, ScanNet και SUN RGB-D. Ιδιαίτερη έμφαση δίνεται στην ταξινόμηση των συνόλων με βάση τη χρήση τους (π.χ. SLAM, 3D segmentation, reconstruction, tracking) αλλά και στην τεχνολογική βάση των αισθητήρων (structured light, stereo vision, time-of-flight, LiDAR). Επιπλέον, η

επισκόπηση αναδεικνύει την αυξανόμενη σημασία των συνθετικών δεδομένων (synthetic datasets), καθώς και την ανάγκη για μεγάλα, σημασιολογικά πλούσια σύνολα με χρονική συνοχή και ακριβή χωρική πληροφορία. Τέλος, η εργασία πλαισιώνεται από έναν διαδραστικό online κατάλογο που επιτρέπει την ταχεία αναζήτηση και φιλτράρισμα κατάλληλων datasets, διευκολύνοντας τη στοχευμένη επιλογή για ερευνητικές εφαρμογές. Η συγκεκριμένη συνεισφορά αποτελεί ένα χρήσιμο εργαλείο αναφοράς για τη σχεδίαση και αξιολόγηση μοντέλων μηχανικής μάθησης και ορατότητας σε πολυδιάστατα περιβάλλοντα.

Ειδικότερα, το ενδιαφέρον επικεντρώνεται στην εργασία των Liao et al.[10], που αποτελεί μια αξιοσημείωτη εξέλιξη στο πεδίο της αντίληψης σκηνών για αυτόνομα οχήματα, μέσω της κυκλοφορίας του συνόλου δεδομένων KITTI-360, το οποίο προτάθηκε ως επέκταση του ευρέως χρησιμοποιούμενου KITTI dataset. Η επιστημονική εργασία παρουσιάζει ένα πλήρως συγχρονισμένο και γεωαναφερμένο σύνολο δεδομένων, που ενσωματώνει αισθητήρες στερεοσκοπικής όρασης, fisheye κάμερες, LIDAR και συστήματα εντοπισμού θέσης υψηλής ακρίβειας (GPS+IMU). Το KITTI-360 διαθέτει σημασιολογική και επιμέρους (instance-level) σήμανση σε 2D και 3D, με χρονολογική συνέπεια, επιτρέποντας έτσι την ανάλυση δυναμικών αστικών σκηνών με λεπτομέρεια. Επιπλέον, η εργασία συνοδεύεται από ένα ολοκληρωμένο σύνολο πλαισίων αξιολόγησης (benchmarks) για καθήκοντα όπως 3D semantic segmentation, instance segmentation, scene completion και semantic SLAM, καθιστώντας το dataset κατάλληλο για εφαρμογές που απαιτούν αντίληψη περιβάλλοντος σε βάθος, όπως στην αυτόνομη πλοήγηση. Μέσω της ενιαίας αναπαράστασης σκηνών σε 360°, το KITTI-360 ενισχύει την έρευνα στην πλήρως χωρική κατανόηση αστικών περιβαλλόντων, συνδέοντας τις περιοχές της ρομποτικής, της τεχνητής όρασης και των αυτόνομων συστημάτων.

Στον τομέα των εφαρμογών μηχανικής και βαθιάς μάθησης μια εφαρμοσμένη και ιδιαίτερα επίκαιρη προσέγγιση στην αξιοποίηση τεχνικών βαθιάς μάθησης για λόγους οδικής ασφάλειας παρουσιάζεται στην εργασία των Sisiyas et al.[11], όπου αναπτύσσεται ένα σύστημα αυτόματης αναγνώρισης οχημάτων που μεταφέρουν επικίνδυνα υλικά (ADR), με σκοπό την διαχείριση ρίσκου σε σήραγγες. Η προτεινόμενη μέθοδος αποτελείται από τρεις διαδοχικές φάσεις: (α) εντοπισμός βαρέων οχημάτων με χρήση μοντέλων MobileNet SSD και YOLO, (β) αναγνώριση της πορτοκαλί ADR πινακίδας μέσω Faster R-CNN, και (γ) εφαρμογή OCR (EasyOCR) για την εξαγωγή του τύπου κινδύνου και του κωδικού υλικού. Η συνολική ακρίβεια

εντοπισμού ξεπερνά το 99%, ενώ η αναγνώριση χαρακτήρων φτάνει ποσοστά άνω του 90% ακόμα και σε ρεαλιστικές συνθήκες. Το σύστημα ενσωματώνεται στην εφαρμογή HMRT-EF, η οποία υποστηρίζει cloud επεξεργασία, αυτόνομη λειτουργία και διαδραστική διεπαφή, καθιστώντας το κατάλληλο για χρήση από διαχειριστές υποδομών και φορείς πολιτικής προστασίας. Η εργασία αναδεικνύει την πρακτική αξιοποίηση τεχνολογιών deep learning, ανίχνευσης αντικειμένων και αναγνώρισης κειμένου σε πραγματικά δεδομένα από την Εγνατία Οδό, προσφέροντας ένα χαμηλού κόστους και υψηλής ακρίβειας εργαλείο πρόληψης και διαχείρισης κινδύνου σε σενάρια μεταφοράς επικίνδυνων φορτίων.

Παράλληλα, μια αξιοσημείωτη συμβολή στην εφαρμογή τεχνικών βαθιάς μάθησης στη γεωργία ακριβείας παρουσιάζεται στην εργασία των Kontogiannis et al.[12], στην οποία παρουσιάζεται ένα πλήρως ολοκληρωμένο cloud-based σύστημα έγκαιρης ανίχνευσης περονόσπορου στην αμπελουργία, με χρήση καμερών και drones συνδεδεμένων με ενσωματωμένες συσκευές (IoT). Ο βασικός πυλώνας της αρχιτεκτονικής που διέπει το συγκεκριμένο σύστημα είναι η λήψη εικόνων υψηλής ανάλυσης σε πραγματικό χρόνο (real time), η αποστολή τους σε cloud υποδομές και στην συνέχεια η επεξεργασία τους με μοντέλα deep learning για ανίχνευση αντικειμένων, όπως Faster R-CNN και YOLO. Το εν λόγω σύστημα εφαρμόστηκε πιλοτικά σε αμπελώνα της ποικιλίας Ντεμπίνα στην Ήπειρο, όπου αξιολογήθηκαν διάφορες εκδοχές των αλγορίθμων (VGG16, MobileNetV3, ResNet, SqueezeNet, ViTDet) ως προς την ακρίβεια και την ταχύτητα ταξινόμησης. Μετά από ενδελεχή έλεγχο επί των αποτελεσμάτων, διαπιστώθηκε ότι το MobileNetV3 αποτελεί το πιο αποδοτικό μοντέλο για ενσωματωμένες συσκευές, ενώ τα Faster R-CNN με ResNet-101/152 είχαν την μεγαλύτερη ακρίβεια σε cloud περιβάλλον. Το σύστημα υποστηρίχθηκε από υποδομές ανοικτού λογισμικού και επέδειξε υψηλή πρακτικότητα και επεκτασιμότητα, συνεισφέροντας στη μείωση χρήσης φυτοφαρμάκων και στην έγκαιρη παρέμβαση σε εστίες μόλυνσης. Η εργασία αυτή αναδεικνύει τη δυναμική των συστημάτων τεχνητής νοημοσύνης σε πραγματικό χρόνο για την υποστήριξη βιώσιμων πρακτικών στη γεωργική παραγωγή.

Επιπροσθέτως, σημαντική προσέγγιση στην εφαρμογή της μηχανικής μάθησης σε συνδιασμό υπολογισμών ακμής και νέφους παρουσιάζεται σε προηγούμενη συνεισφορά στην εργασία Gkamas et al.[13] με εφαρμογή σε βιομηχανικά περιβάλλοντα, όπου προτείνεται ένα υβριδικό σύστημα ταξινόμησης βιομηχανικών δεδομένων με υποστήριξη cloud, σχεδιασμένο για ανίχνευση κρίσιμων συμβάντων και υποστή-

ριξη προληπτικής συντήρησης. Η συγκεκριμένη αρχιτεκτονική στηρίζεται κατά κύριο λόγο στην χρήση ελαφρών νευρωνικών δικτύων τύπου Multi-Layer Perceptron (MLP), με περιορισμένα χαρακτηριστικά (2 κρυφά επίπεδα $\times$ 7 νευρώνες), τα οποία μπορούν να εκτελούνται ακόμη και σε συσκευές χαμηλών πόρων, όπως Raspberry Pi και μικροελεγκτές, με δυνατότητα over-the-air ενημέρωσης μοντέλων από το cloud. Το σύστημα αξιολογήθηκε ενδελεχώς σε πραγματικά δεδομένα από αισθητήρες που παρακολουθούν κρίσιμες μονάδες αντλιών σε διυλιστήριο πετρελαίου, καταγράφοντας παραμέτρους όπως κραδασμούς, θερμοκρασία και πίεση. Παράλληλα, τα αποτελέσματα έδειξαν ότι το προτεινόμενο μοντέλο πέτυχε ακρίβεια περίπου 85,6%, διατηρώντας πολύ χαμηλό αποτύπωμα μνήμης (4,4 KB), γεγονός που το καθιστά κατάλληλο για ενσωμάτωση σε πραγματικό χρόνο. Τέλος, η εργασία καταδεικνύει την δυνατότητα εφαρμογής τεχνικών TinyML σε συνδυασμό με cloud υποδομές, ώστε να επιτευχθεί αυτόνομη, χαμηλού κόστους και αποδοτική παρακολούθηση βιομηχανικών συστημάτων, υποστηρίζοντας τον μετασχηματισμό προς Βιομηχανία 4.0/5.0 με έμφαση στην λειτουργική βιωσιμότητα και την τεχνητή νοημοσύνη αιχμής.

Στον τομέα της ασφαλούς και αποδοτικής διαχείρισης των Μεγαδεδομένων, προηγούμενη συνεισφορά στην εργασία Gkamas et al. [14] αποτελεί μια αρκετά αξιόλογη επιστημονική εκπόνηση, η οποία επικεντρώνεται στην αξιολόγηση της απόδοσης διαφορετικών τύπων βάσεων δεδομένων ως υπηρεσία (DBaaS) σε περιβάλλοντα Βιομηχανικού Διαδικτύου των Πραγμάτων (IIoT). Μέσω εκτεταμένων πειραμάτων με πραγματικά δεδομένα αισθητήρων από βιομηχανικό συμπιεστή, συγκρίνονται οι τεχνολογίες MongoDB (ως NoSQL προσέγγιση), PostgreSQL με JSON υποστήριξη και PostgreSQL με παραδοσιακό σχεσιακό σχήμα, υπό τρεις τύπους φορτίου: εισαγωγή δεδομένων, απλές ανακτήσεις και υπολογισμός στατιστικών συσχετίσεων. Τα αποτελέσματα έδειξαν ότι η MongoDB παρουσιάζει ανώτερο throughput και χαμηλότερη καθυστέρηση στις μαζικές εισαγωγές, ενώ η PostgreSQL με σχεσιακό σχήμα υπερέχει κατά πολύ σε ανακτήσεις και αναλύσεις, με ως και τετραπλάσια βελτίωση στον χρόνο απόκρισης και σημαντικά υψηλότερη απόδοση σε queries υπολογιστικής φύσης. Η μελέτη αποδεικνύει την ανάγκη επιλογής βάσης δεδομένων ανάλογα με τις απαιτήσεις της εφαρμογής IIoT, προτείνοντας την υβριδική χρήση NoSQL για αποδοτική συλλογή δεδομένων και SQL για αποθήκευση, ανάλυση και επιχειρησιακή αξιοποίηση. Η εργασία συμβάλλει ουσιαστικά στη βελτιστοποίηση των αρχιτεκτονικών δεδομένων για συστήματα Βιομηχανίας 4.0.

Επέκταση της προηγούμενης τεχνογνωσίας καθώς και μια ιδιαίτερα σημαντική

προσέγγιση στην αξιολόγηση τεχνολογιών αποθήκευσης και επεξεργασίας δεδομένων σε περιβάλλοντα Βιομηχανίας 4.0 παρουσιάζεται στη συνεισφορά επι της εργασίας Gkamas et al.[15], όπου συγκρίνονται τρεις διαφορετικές στρατηγικές κατανα-
μημένων βάσεων δεδομένων – συγκεκριμένα, η MongoDB (NoSQL), η PostgreSQL με υποστήριξη JSON, και η PostgreSQL με παραδοσιακό σχεσιακό σχήμα. Τα αποτελέ-
σματα καταδεικνύουν ότι η MongoDB υπερέχει σε ρυθμούς εισαγωγής δεδομένων (throughput) υπό συνθήκες αυξημένου φόρτου, ενώ η PostgreSQL με σχεσιακό σχήμα αποδίδει καλύτερα σε queries επιλογής και διασταύρωσης. Επιπλέον, οι συγγραφείς ενσωματώνουν τις δοκιμασμένες λύσεις στην πλατφόρμα ARIS (Augmented Real-
ity Information System), η οποία συνδυάζει τεχνολογίες Docker, MQTT και REST-
ful APIs για την παροχή στατιστικών αναλύσεων, απομακρυσμένης πρόσβασης και οπτικοποίησης δεδομένων συντήρησης μέσω επαυξημένης πραγματικότητας. Η συ-
γκεκριμένη επιστημονική εργασία αποτελεί ουσιαστική συμβολή στην κατανόηση των πλεονεκτημάτων και περιορισμών διαφορετικών τύπων βάσεων δεδομένων σε εφαρμογές IIoT και προτείνει πρακτικές λύσεις για την υλοποίηση ευέλικτων, επε-
κτάσιμων και ασφαλών υποδομών αποθήκευσης βιομηχανικών δεδομένων.

Συνοψίζοντας το επιστημονικό υπόβαθρο που προκύπτει από τους τομείς των εφαρμογών μηχανικής και βαθιάς μάθησης, της ασφαλούς και αποδοτικής διαχείρισης των Μεγαδεδομένων, της αποκεντρωμένης τηλεδιοίκησης και τηλεεμποπτείας, καθώς και γενικά του σχεδιασμού και υλοποίησης κρίσιμων συστημάτων Internet of Things για την Βιομηχανία 4.0, η εργασία Karaïskos et al.[16] αποτελεί κομβική επιστημο-
νική συμβολή στο πλαίσιο της μετάβασης προς τη Βιομηχανία 4.0. Επι της εργασίας προτείνεται ένα ολιστικό πλαίσιο συντήρησης κρίσιμων βιομηχανικών υποδομών, το οποίο αξιοποιεί τεχνολογίες ψηφιακών διδύμων (digital twins), επαυξημένης πραγ-
ματικότητας (AR), και υποστηριζόμενης λήψης αποφάσεων μέσω μηχανικής μάθη-
σης. Το προτεινόμενο σύστημα ARIS-IMS εφαρμόζεται πιλοτικά σε διυλιστήριο της HEL.PE. και υποστηρίζει επιτόπια παρακολούθηση εξοπλισμού μέσω 3D αποτύπω-
σης με χρήση LiDAR, cloud-based καταγραφή και ανάλυση μετρήσεων αισθητήρων, και διαχείριση εργασιών συντήρησης μέσω web/mobile διεπαφών. Κεντρικό ρόλο στο πλαίσιο παίζουν οι τρεις βασικές δράσεις: (1) Οπτικοποίηση μέσω 3D μοντέ-
λων, (2) αποκεντρωμένη και διαφανής διαχείριση μέσω cloud και (3) υποστήριξη αποφάσεων με χρήση ιστορικών δεδομένων και τεχνικών machine learning. Η ερ-
γασία συνδυάζει τεχνολογίες όπως MQTT, MongoDB, InfluxDB και Grafana για real-time οπτικοποίηση και δίνει ιδιαίτερη έμφαση στην ενσωμάτωση σε φορητές

συσκευές πιστοποιημένες για εκρήξιμα περιβάλλοντα (ATEX Zone 2), καθιστώντας την κατάλληλη για απαιτητικά και επικίνδυνα βιομηχανικά περιβάλλοντα. Η προσέγγιση αυτή αναδεικνύεται ως ένα σύγχρονο, ευέλικτο και επεκτάσιμο πρότυπο για ευφυή, αποκεντρωμένη και τεκμηριωμένη βιομηχανική συντήρηση.

ΚΕΦΑΛΑΙΟ 3

ΑΡΧΙΤΕΚΤΟΝΙΚΗ

3.1 Προτεινόμενο Σύστημα IoT Έξυπνων Πόλεων για την Οδική Ασφάλεια

3.2 Οπτικοποίηση Αντικειμένων Ενδιαφέροντος

3.1 Προτεινόμενο Σύστημα IoT Έξυπνων Πόλεων για την Οδική Ασφάλεια

Η αρχιτεκτονική που αναπτύσσεται στο πλαίσιο της παρούσας Διπλωματικής Εργασίας στοχεύει πρωτίστως στην ενίσχυση της οδικής ασφάλειας μέσα από την ενσωμάτωση τεχνολογιών Επαυξημένης Πραγματικότητας (Ε.Π.), την ανάλυση δεδομένων αισθητήρων και την ευφυή επικοινωνία μεταξύ οχημάτων και υποδομών. Η προσέγγιση αυτή αποσκοπεί στη δημιουργία ενός συνεργατικού οικοσυστήματος συνδεδεμένων αυτόνομων οχημάτων, ικανών να προβλέπουν και να αποτρέπουν επικίνδυνες καταστάσεις πριν αυτές επέλθουν.

Η αρχιτεκτονική του συστήματος οργανώνεται γύρω από τρεις κύριους άξονες – πυλώνες, οι οποίοι λειτουργούν συμπληρωματικά και συνθέτουν μια πλήρως διασυνδεδεμένη υποδομή. Ο Άξονας 1 (CAR AR) αποτελεί το λειτουργικό πυρήνα του συστήματος, εστιάζοντας στις τεχνολογίες Επαυξημένης Πραγματικότητας που εφαρμόζονται στο εσωτερικό του οχήματος για την οπτικοποίηση κρίσιμων πληροφοριών κυκλοφορίας, με στόχο τη βελτίωση της αντιληπτικής επίγνωσης του οδηγού. Ο Άξονας 2 (Έξυπνος Σηματοδότης) επικεντρώνεται στην τρισδιάστατη ανίχνευση και αποτύπωση του περιβάλλοντος μέσω αισθητήρων LiDAR και στην εκπομπή

πληροφοριών σε συνδεδεμένα οχήματα (CAV) για έγκαιρη πρόβλεψη κινδύνων. Τέλος, ο Άξονας 3 (Υπηρεσίες Νέφους) αποτελεί το κεντρικό υπολογιστικό υπόβαθρο της αρχιτεκτονικής, υπεύθυνο για την αποθήκευση, ανάλυση και ανακατανομή των μεγαδεδομένων οδήγησης.

Η ανωτέρω τριμερής αρχιτεκτονική αποτελεί τη θεμελιώδη βάση πάνω στην οποία αναπτύσσεται το προτεινόμενο σύστημα Επαυξημένης Πραγματικότητας για την Οδική ασφάλεια έξυπνων πόλεων Συνδεδεμένων αυτόνομων οχημάτων (Ε.Π.Ο.Σ.), και θα αναλυθεί διεξοδικά στα επόμενα κεφάλαια, με στόχο την παρουσίαση της λειτουργικής συνέργειας των υποσυστημάτων του και της συμβολής τους στην ενίσχυση της οδικής ασφάλειας.

3.1.1 Σχήμα

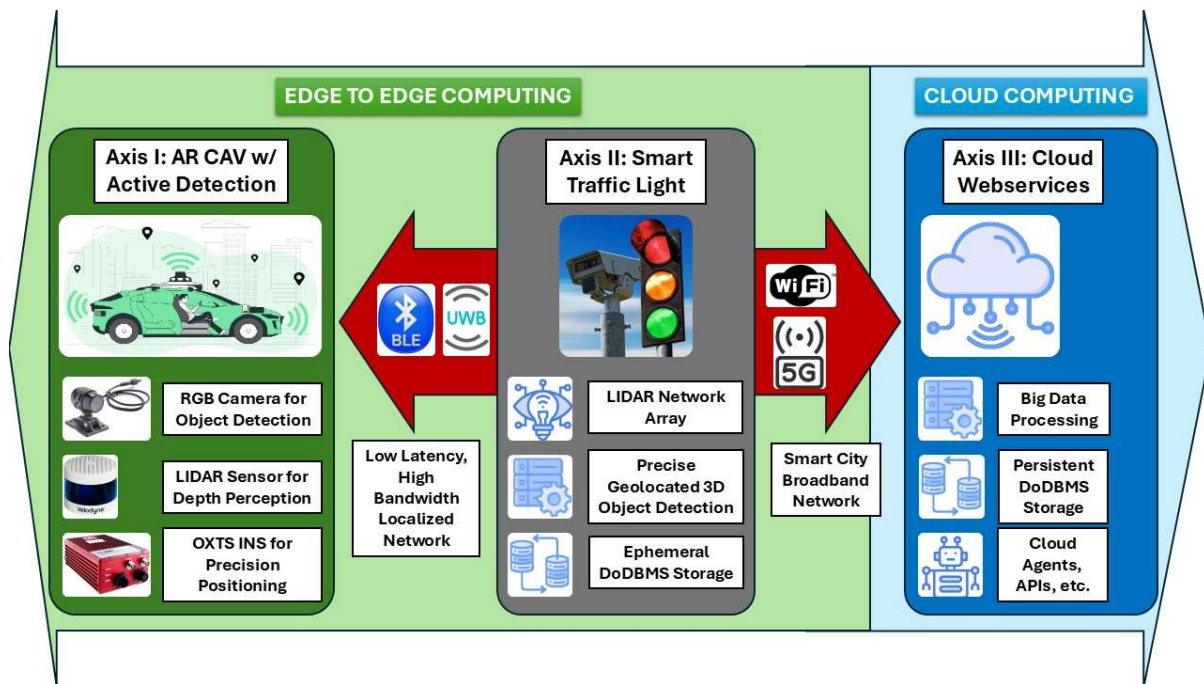
Το σχήμα 3.1 παρουσιάζει τη συνολική αρχιτεκτονική του προτεινόμενου συστήματος, το οποίο βασίζεται στη συνεργασία μεταξύ Edge-to-Edge Computing και Cloud Computing υποδομών. Αποτελείται από τρεις κύριους άξονες λειτουργίας:

Άξονας I – AR CAV με Ενεργή Ανίχνευση (Active Detection): Αντιπροσωπεύει το συνδεδεμένο αυτόνομο όχημα με δυνατότητες επαυξημένης πραγματικότητας, το οποίο είναι εξοπλισμένο με αισθητήρες RGB κάμερας, LIDAR και μονάδα αδρανειακής πλοήγησης (OXTS INS) για ακριβή εντοπισμό και αντίληψη βάθους.

Άξονας II – Έξυπνος Σηματοδότης (Smart Traffic Light): Αποτελεί τον ενδιάμεσο κόμβο επεξεργασίας δεδομένων στο επίπεδο του Edge, διαθέτοντας δικτυακό σύστημα LIDAR, δυνατότητες ακριβούς γεωεντοπισμένης τρισδιάστατης ανίχνευσης αντικειμένων και προσωρινή βάση δεδομένων (ephemeral DoDBMS storage).

Άξονας III – Υπηρεσίες Cloud (Cloud Webservices): Υλοποιεί την υποδομή cloud για την αποθήκευση, επεξεργασία και ανάλυση μεγάλου όγκου δεδομένων (Big Data), παρέχοντας επιπλέον APIs και cloud agents για τη διαχείριση και διάχυση πληροφορίας.

Η επικοινωνία μεταξύ των αξόνων επιτυγχάνεται μέσω δικτύων χαμηλής καθυστέρησης και υψηλού εύρους ζώνης (Bluetooth, UWB, Wi-Fi, 5G), επιτρέποντας τη συνεργατική λειτουργία μεταξύ των οχημάτων, των έξυπνων σηματοδοτών και των cloud υπηρεσιών. Συνολικά, το σχήμα αποτυπώνει τη ροή δεδομένων και τις σχέσεις μεταξύ των επιπέδων του συστήματος, παρέχοντας μια ολοκληρωμένη εικόνα της λειτουργικής αρχιτεκτονικής, η οποία θα αναλυθεί διεξοδικότερα στα επόμενα



Σχήμα 3.1: Η Αρχιτεκτονική του Προτεινόμενου Συστήματος.

κεφάλαια.

3.1.2 Άξονας I: CAV AR

Ο Άξονας I – CAR AR συγκροτεί τον βασικό τεχνολογικό πυρήνα του συστήματος, περιλαμβάνοντας όλα τα απαραίτητα τεχνικά χαρακτηριστικά και υποσυστήματα που απαιτούνται για την απρόσκοπτη λειτουργία του οχήματος στο πλαίσιο της Επαυξημένης Πραγματικότητας. Η αρχιτεκτονική του άξονα στηρίζεται σε ένα ολοκληρωμένο πλέγμα αισθητήρων και αλγοριθμικών μονάδων, οι οποίες συνεργάζονται συντονισμένα για την απόκτηση, προεπεξεργασία, ανάλυση και οπτικοποίηση δεδομένων του περιβάλλοντος.

Η λειτουργικότητα των επιμέρους ενοτήτων ενοποιείται μέσω μιας διεπαφής Επαυξημένης Πραγματικότητας (AR User Interface), η οποία προσαρμόζεται δυναμικά στις εκάστοτε συνθήκες οδήγησης και στα χαρακτηριστικά του χρήστη. Σκοπός της αρχιτεκτονικής αυτής είναι η προβολή κρίσιμων πληροφοριών κυκλοφορίας στο οπτικό πεδίο του οδηγού, επιτρέποντας την έγκαιρη αντίληψη και πρόβλεψη δυνητικών κινδύνων, ενισχύοντας έτσι την οδική ασφάλεια και συμβάλλοντας ουσιαστικά στη μείωση ατυχημάτων σε περιβάλλοντα συνδεδεμένων αυτόνομων οχημάτων. Ακολούθως περιγράφονται τα επιμέρους συστήματα αισθητήρων, αλγορίθμων και το

σύστημα διεπαφής επαυξημένης πραγματικότητας ανθρώπου / υπολογιστή.

Εν αρχή και ακρογωνιαίος λίθος τεχνικών χαρακτηριστικών ενός συνδεδεμένου αυτόνομου οχήματος είναι ένα σύστημα θέσης OXTS (Oxford Technical Solutions), το οποίο είναι ένα πολύ ακριβές σύστημα εντοπισμού θέσης, αρκετά πιο ακριβές από το γνωστό σε όλους μας GPS. Αναλυτικότερα, το OXTS αποτελεί ένα σύστημα μέτρησης θέσης, ταχύτητας και προσανατολισμού, που χρησιμοποιείται κυρίως σε εφαρμογές αυτόνομης οδήγησης, χαρτογράφησης και πλοήγησης υψηλής ακρίβειας. Το σύστημα θέσης OXTS, αναπτύχθηκε από την ομόνυμη εταιρεία, η οποία εξειδικεύεται στην υλοποίηση συστημάτων GNSS/INS (Global Navigation Satellite System / Inertial Navigation System). Ακόμη, το σύστημα OXTS συνδυάζει δορυφορικά σήματα από άλλα συστήματα όπως GPS, GLONASS, Galileo ή BeiDou, αδρανειακούς αισθητήρες (IMU – Inertial Measurement Unit: γυροσκόπια, επιταχυνσιόμετρα). Επιπροσθέτως, παρέχει λεπτομερείς 3D συντεταγμένες (γεωγραφικό πλάτος, μήκος, υψόμετρο), ταχύτητα, καθώς και επιτάχυνση, προσανατολισμό (roll, pitch, yaw/heading), καθώς και χρονισμό υψηλής ακρίβειας. Επίσης, το συγκεκριμένο σύστημα διαθέτει αρκετά πλεονεκτήματα όπως:

- Πολύ υψηλή ακρίβεια θέσης ακόμη και με περιορισμένο σήμα GNSS (χάρη στο INS),
- εναρμόνιση με πολλά πρωτόκολλα (NMEA, CAN, Ethernet),
- συγχρονισμός δεδομένων με Pulse Per Second (PPS) για χρονική ακρίβεια,
- δυνατότητα καταγραφής δεδομένων και σε συνθήκες χαμηλής ορατότητας δορυφόρων.

Επιπροσθέτως, το σύστημα OXTS (Oxford Technical Solutions) αποτελεί ένα ολοκληρωμένο σύστημα GNSS/INS που συνδυάζει δεδομένα από δορυφορική πλοήγηση (GNSS), καθώς και αδρανειακή μέτρηση (INS) προκειμένου να παρέχει ακριβή υπολογισμό θέσης, ταχύτητας και προσανατολισμού ενός κινούμενου οχήματος ή πλατφόρμας.

Το συγκεκριμένο σύστημα, χρησιμοποιεί σήματα από πολλαπλά δορυφορικά συστήματα πλοήγησης (GPS, GLONASS, Galileo, BeiDou) σε συνδυασμό με δεδομένα από την αδρανειακή μονάδα μέτρησης (IMU), η οποία περιλαμβάνει: Γυροσκόπια για μέτρηση γωνιακών ταχυτήτων (roll, pitch, yaw) και επιταχυνσιόμετρα για μέτρηση γραμμικών επιταχύνσεων.

Κάνοντας χρήση αλγορίθμων συγχώνευσης αισθητήρων (sensor fusion), το OXTS δύναται να εκτιμήσει με αρκετά μεγάλη ακρίβεια: Ταχύτητα και επιτάχυνση, χρονισμό (με χρονική ακρίβεια υπό millisecond), τον προσανατολισμό του οχήματος, καθώς και το γεωγραφικό πλάτος, το μήκος, καθώς και το υψόμετρο του. Ειδικότερα, αποτελεί ένα αρκετά εξειδικευμένο σύστημα εντοπισμού θέσης, το οποίο διαθέτει επίσης και μια σειρά από πλεονεκτήματα διαλειτουργικότητας και συνεργασίας με άλλα συστήματα. Ειδικότερα, παρουσιάζει αρκετά μεγάλη αντοχή σε απώλειες σήματος GNSS μέσω πρόβλεψης θέσης από το INS, έχει την δυνατότητα του χρονικού συγχρονισμού (PPS – Pulse Per Second) για ακριβή αντιστοίχιση με δεδομένα από LiDAR και κάμερες, υποστηρίζει διάφορες διεπαφές επικοινωνίας, όπως Ethernet, CAN, NMEA, καταγράφει δεδομένα για offline ανάλυση και όταν χρησιμοποιεί RTK GNSS, υπολογίζει την θέση με ακρίβεια εκατοστού.

Ακόμη, στο πλαίσιο ενός συστήματος αντίληψης για αυτόνομα οχήματα ή ρομποτικά συστήματα, το OXTS λειτουργεί ως κεντρική πηγή αναφοράς θέσης και προσανατολισμού με το σύστημα LiDAR, το οποίο περιγράφεται σε επόμενη παράγραφο. Οι μετρήσεις θέσης/προσανατολισμού χρησιμοποιούνται για την γεωαναφορά του νέφους σημείων (point cloud), επιτρέποντας την αντιστοίχιση με γεωγραφικές συντεταγμένες ή τη συγχώνευση δεδομένων από πολλαπλές σαρώσεις. Ακολούθως, κάμερες ορατού φωτός χρησιμοποιούν τα δεδομένα OXTS για ευθυγράμμιση εικόνων και υπολογισμό της ακριβούς θέσης του οχήματος κατά την στιγμή λήψης κάθε καρέ, βελτιώνοντας την απόδοση των αλγορίθμων ανίχνευσης αντικειμένων και SLAM.

Αναφορικά με τις κάμερες ορατού φάσματος, αυτές αποτελούν έναν από τους πλέον κρίσιμους αισθητήρες στον τομέα της ενεργής ανίχνευσης αντικειμένων ενδιαφέροντος. Οι κάμερες υψηλής ευκρίνειας και υψηλού ρυθμού καταγραφής είναι απαραίτητες για την τροφοδότηση των αλγορίθμων μηχανικής μάθησης, εξασφαλίζοντας ταχεία και αξιόπιστη αναγνώριση αντικειμένων στο περιβάλλον του οχήματος. Η παρουσία τουλάχιστον μίας κάμερας θεωρείται αναγκαία, ενώ σε πλήρως αυτόνομα συστήματα παρατηρείται η χρήση πολλαπλών καμερών για ευρύτερη γωνία κάλυψης και καλύτερη αντίληψη βάθους. Τα εξαγόμενα καρέ βίντεο ευθυγραμμίζονται στη συνέχεια με τα δεδομένα των υπολοίπων αισθητήρων μέσω του προαναφερθέντος συστήματος αδρανειακής πλοήγησης και γεωαναφοράς OXTS, επιτρέποντας τη συγχώνευση δεδομένων (sensor fusion) σε κοινό αναφορικό σύστημα και τη συνεκτική απεικόνιση του περιβάλλοντος στο πλαίσιο της Επαυξημένης Πραγματι-

κότητας.

Το LiDAR (Light Detection and Ranging) ολοκληρώνει τους βασικότερους αισθητήρες της αρχιτεκτονικής του συστήματος, προσφέροντας ακριβή τρισδιάστατη αποτύπωση του περιβάλλοντος χώρου. Ο αισθητήρας λειτουργεί με την εκπομπή παλμών laser μέσω ενός πομπού, οι οποίοι ανακλώνται στις επιφάνειες των αντικειμένων που βρίσκονται εντός του πεδίου σάρωσης. Στη συνέχεια, ο δέκτης του LiDAR μετρά τον χρόνο επιστροφής της ακτίνας (time of flight), από τον οποίο υπολογίζεται η απόσταση σύμφωνα με τη σχέση: $\text{distance} = (\text{speed of light} \times \text{time}) / 2$.

Η διαδικασία αυτή επαναλαμβάνεται χιλιάδες φορές το δευτερόλεπτο, παράγοντας ένα νέφος σημείων (point cloud) που αποτελείται από τρισδιάστατες συντεταγμένες (x, y, z). Κάθε σημείο φέρει επιπλέον τιμή έντασης (intensity), η οποία αντιστοιχεί στην ανακλώμενη ενέργεια του παλμού και παρέχει πληροφορία για τη φύση και την ανακλαστικότητα της επιφάνειας.

Το παραγόμενο νέφος σημείων χρησιμοποιείται για τη δημιουργία χαρτών βάθους (depth maps) και την αντίληψη του τρισδιάστατου περιβάλλοντος σε πραγματικό χρόνο, όπως αποστάσεις από οχήματα, πεζούς ή εμπόδια. Ο συνδυασμός των δεδομένων LiDAR με τις RGB εικόνες της κάμερας (camera-LiDAR fusion) οδηγεί σε μια πιο λεπτομερή και ακριβή αναπαράσταση του περιβάλλοντος, ενισχύοντας σημαντικά την αξιοπιστία της ανίχνευσης και της οπτικοποίησης.

Στον τομέα των αλγοριθμικών μονάδων, βασικό στοιχείο του συστήματος αποτελεί η ανίχνευση αντικειμένων (object detection) με μηχανική μάθηση, η οποία είναι αρκετά κρίσιμη διεργασία για αυτόνομα συστήματα, καθώς επιτρέπει την αναγνώριση και τον εντοπισμό αντικειμένων στο περιβάλλον. Στο παρόν σύστημα χρησιμοποιείται το YOLOv11 (You Only Look Once, έκδοση 11), το οποίο αποτελεί ένα state-of-the-art deep learning μοντέλο που επιτυγχάνει ισορροπία μεταξύ ακρίβειας και ταχύτητας σε πραγματικό χρόνο.

Ειδικότερα, το YOLOv11 αποτελεί ένα CNN-based μοντέλο (Convolutional Neural Network) που λειτουργεί με αρχιτεκτονική single-shot detection. Αντί να σαρώνει την εικόνα τμηματικά (region proposals), εκτελεί ολική πρόβλεψη (end-to-end prediction) σε ένα μόνο πέρασμα, εξάγοντας: Bounding boxes (πλαίσια γύρω από αντικείμενα), Class labels (κατηγορίες αντικειμένων), καθώς και Confidence scores. Ως εκ τούτου, αυτό το καθιστά ιδανικό για real-time εφαρμογές σε αυτόνομα οχήματα.

Παράλληλα, το μοντέλο προ-εκπαιδεύεται στο COCO dataset (Common Objects

in Context), το οποίο αποτελεί ένα από τα πιο διαδεδομένα datasets για ανίχνευση αντικειμένων. Επιπλέον, το σύνολο δεδομένων COCO περιλαμβάνει περίπου 330.000 εικόνες, κατανεμημένες σε 80 διακριτές κατηγορίες αντικειμένων — όπως οχήματα, πεζούς, ποδήλατα και φωτεινούς σηματοδότες — ενώ συνοδεύεται από πολλαπλές επισημειώσεις (annotations) που καλύπτουν εργασίες ανίχνευσης αντικειμένων (object detection), τμηματοποίησης (segmentation) και ανάλυσης σημείων ενδιαφέροντος (keypoint detection), προσφέροντας έτσι ένα πλούσιο και ποικιλόμορφο σύνολο δεδομένων για την εκπαίδευση και αξιολόγηση αλγορίθμων όρασης υπολογιστή. Η προεκπαίδευση στο COCO επιτρέπει στο YOLOv11 να έχει γενίκευση σε ποικίλα σενάρια, ακόμα και σε περιβάλλοντα που δεν έχουν χρησιμοποιηθεί στην διαδικασία της εκπαίδευσης.

- Input: RGB εικόνες με χειροκίνητα σχολιασμένα αντικείμενα (bounding boxes + labels).
- Data Augmentation: Εφαρμογή τεχνικών (crop, rotate, flip, color jitter) για ενίσχυση γενίκευσης.
- Loss Function: Συνδυασμός classification loss, localization loss και objectness loss.
- Backpropagation: Βελτίωση βαρών με χρήση SGD/Adam optimizer.
- Output: Εκπαιδευμένο μοντέλο που μαθαίνει να αναγνωρίζει αντικείμενα με βάση τα χαρακτηριστικά τους.

Επί του πρακτέου, κατά τη φάση εκτέλεσης (inference), το YOLOv11 δέχεται ως είσοδο μια RGB εικόνα προερχόμενη από την κάμερα του οχήματος και την επεξεργάζεται μέσω συνελικτικών επιπέδων (Convolutional Neural Network – CNN layers) για την εξαγωγή ιεραρχικών χαρακτηριστικών του οπτικού πεδίου. Στη συνέχεια, το δίκτυο παράγει προβλέψεις για πλαίσια οριοθέτησης (bounding boxes) και αντίστοιχες κατηγορίες αντικειμένων (labels) για κάθε ανιχνευμένο στοιχείο.

Για τη βελτίωση της ακρίβειας των αποτελεσμάτων, εφαρμόζεται αλγόριθμος Καταστολής Μη Μέγιστων (Non-Maximum Suppression, NMS), ο οποίος απορρίπτει επικαλυπτόμενα ή πλεονάζοντα πλαίσια, διατηρώντας μόνο εκείνα με το υψηλότερο ποσοστό εμπιστοσύνης. Το τελικό αποτέλεσμα αποτελείται από ένα σύνολο ανιχνευμένων αντικειμένων, τα οποία απεικονίζονται ως πλαίσια οριοθέτησης (bound-

ing boxes) εντός του περιβάλλοντος Επαυξημένης Πραγματικότητας, παρέχοντας άμεση και ευανάγνωστη ανατροφοδότηση στον οδηγό.

Μια από τις βασικές αλγοριθμικές λειτουργίες του Άξονα Ι αφορά τη μετατροπή των δεδομένων του GNSS (Global Navigation Satellite System) από γεωδαιτικές σε καρτεσιανές συντεταγμένες, διαδικασία καθοριστικής σημασίας για τη συνεπή και ακριβή ολοκλήρωση των χωρικών δεδομένων του συστήματος. Τα δεδομένα του GNSS εκφράζονται αρχικά σε γεωδαιτικές συντεταγμένες — γεωγραφικό πλάτος (latitude, ϕ), γεωγραφικό μήκος (longitude, λ) και υψόμετρο (height, h) ως προς το ελλειψοειδές αναφοράς WGS84 — οι οποίες δεν είναι απευθείας συμβατές με τις αναπαραστάσεις που απαιτούνται για αλγορίθμους μηχανικής μάθησης, τρισδιάστατη οπτικοποίηση και συγχώνευση δεδομένων αισθητήρων (sensor fusion).

Για τον σκοπό αυτό, πραγματοποιείται μετασχηματισμός από γεωδαιτικές σε καρτεσιανές συντεταγμένες, ξεκινώντας από την προβολή των σημείων στο παγκόσμιο σύστημα ECEF (Earth-Centered, Earth-Fixed). Οι συντεταγμένες (x, y, z) υπολογίζονται σύμφωνα με τις εξισώσεις:

$$N = a / \sqrt{1 - e^2 \sin^2 \Phi} \quad (3.1)$$

$$x = (N + h) \cos \Phi \cos \lambda \quad (3.2)$$

$$y = (N + h) \cos \Phi \sin \lambda \quad (3.3)$$

$$z = ((1 - e^2)N + h) \sin \Phi \quad (3.4)$$

όπου a είναι ο μεγάλος ημιάξονας του ελλειψοειδούς WGS84, e η εκκεντρότητα του, Φ το γεωγραφικό πλάτος (μοίρες βόρειο/νότιο), λ το γεωγραφικό μήκος (μοίρες ανατολικό/δυτικό) και h το ελλειψοειδές ύψος (μέτρα πάνω από το ελλειψοειδές WGS84).

Στη συνέχεια, τα δεδομένα μετατρέπονται από το παγκόσμιο σύστημα ECEF σε τοπικό σύστημα αναφοράς ENU (East–North–Up), μέσω γραμμικού μετασχηματισμού περιστροφής και μετατόπισης:

$$\begin{bmatrix} E \\ N \\ U \end{bmatrix} = R_{ECEF \rightarrow ENU} \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \quad (3.5)$$

όπου

$$(\mathbf{x}_0, \mathbf{y}_0, \mathbf{z}_0)$$

είναι οι συντεταγμένες του σημείου αναφοράς (π.χ. θέση οχήματος) και

$$\mathbf{R}_{\text{ECEF} \rightarrow \text{ENU}}$$

ο πίνακας περιστροφής που καθορίζει τον προσανατολισμό του τοπικού συστήματος.

Ο μετασχηματισμός αυτός εξασφαλίζει ότι όλα τα δεδομένα — LiDAR, κάμερα και GNSS — εκφράζονται σε κοινό πλαίσιο συντεταγμένων, επιτρέποντας ακριβή γεωαναφορά, συγχώνευση αισθητήρων και ρεαλιστική οπτικοποίηση στο περιβάλλον Επαυξημένης Πραγματικότητας (AR UI).

Τα δεδομένα των αισθητήρων συλλέγονται, συγχρονίζονται και επεξεργάζονται στον κεντρικό υπολογιστή ελέγχου του οχήματος, ο οποίος αποτελεί τον πυρήνα της υπολογιστικής αρχιτεκτονικής του συστήματος. Λόγω του υψηλού υπολογιστικού φόρτου που απαιτείται για την επεξεργασία δεδομένων σε πραγματικό χρόνο —ιδίως για την εκτέλεση αλγορίθμων μηχανικής μάθησης και την απεικόνιση περιβάλλοντος επαυξημένης πραγματικότητας— οι παραδοσιακοί μικροελεγκτές όπως οι Arduino και ESP32 δεν επαρκούν. Ακόμη και πιο προηγμένες πλατφόρμες, όπως ορισμένες εκδόσεις του Raspberry Pi, βρίσκονται στα όρια των δυνατοτήτων τους. Για τον λόγο αυτό, επιλέγονται ισχυρότεροι ελεγκτές με πολυπύρηνους επεξεργαστές, μεγαλύτερη χωρητικότητα μνήμης και υποστήριξη επιτάχυνσης μέσω καρτών γραφικών (GPU), ώστε να εξασφαλίζεται η γρήγορη εκτέλεση προβλέψεων μηχανικής μάθησης και η αποδοτική οπτικοποίηση των γραφικών στοιχείων της διεπαφής επαυξημένης πραγματικότητας. Επιπλέον, τα υπολογιστικά αυτά συστήματα ενσωματώνουν μονάδες επικοινωνίας Ultra-Wideband (UWB) και Bluetooth Low Energy (BLE), προσφέροντας υψηλής ταχύτητας και σταθερότητας επικοινωνία στο τοπικό πεδίο, σε συνδυασμό με αντίστοιχα πρωτόκολλα διασύνδεσης για ασφαλή και αποδοτική ανταλλαγή δεδομένων μεταξύ των υποσυστημάτων.

Στη συνέχεια, ένα από τα πιο κρίσιμα υποσυστήματα του προτεινόμενου αρχιτεκτονικού πλαισίου αφορά το μέσο οπτικοποίησης της Επαυξημένης Πραγματικότητας (ΕΠ), δηλαδή το φυσικό υπόβαθρο προβολής της πληροφορίας στο οπτικό πεδίο του οδηγού. Για την κάλυψη αυτής της ανάγκης προτείνονται λύσεις εμπορικά διαθέσιμες (COTS), όπως το In-Plane Holographic Optical Element (IP-HOE) display[17] της Texas Instruments, το Holographic Windshield Display[18] της HYUNDAI MOBIS,

το Switchable Smart Film[19] της Intelligent Glass, το Custom Transparent AR Film[20] της Anyractive καθώς και το Holographic Invisible Screen[21] της Muxwave, μεταξύ άλλων.

Η τεχνολογία των εναλλασσόμενων έξυπνων μεμβρανών (Switchable Smart Films) αναδεικνύεται ως μία καινοτόμος και ευέλικτη επιλογή για την οπτικοποίηση της ΕΠ, προσφέροντας τη δυνατότητα μετατροπής οποιασδήποτε γυάλινης επιφάνειας —όπως το εσωτερικό παρμπρίζ του οχήματος— σε δυναμική επιφάνεια προβολής. Το υλικό αυτό μεταβάλλει ηλεκτρονικά τη διαπερατότητά του από διάφανο σε αδιαφανές (frosted), επιτρέποντας την προβολή γραφικών στοιχείων και κρίσιμων πληροφοριών απευθείας στο οπτικό πεδίο του οδηγού. Στην ενεργή (αδιαφανή) κατάσταση λειτουργεί ως οθόνη εμπρόσθιας ή οπίσθιας προβολής υψηλής ευκρίνειας, υποστηρίζοντας καθαρή και χωρίς αντανάκλαση παρουσίαση εικονικών αντικειμένων, σημάνσεων και ενδείξεων. Αντίθετα, στην παθητική (διάφανη) λειτουργία επιτρέπει την ανεμπόδιστη θέαση του φυσικού περιβάλλοντος, διατηρώντας τη ρεαλιστικότητα και την ασφάλεια της οδηγικής εμπειρίας. Η ενσωμάτωσή της στο σύστημα ΕΠ δημιουργεί έναν υβριδικό μηχανισμό αλληλεπίδρασης, μέσω του οποίου το ίδιο το παρμπρίζ μετατρέπεται σε έξυπνη, προσαρμόσιμη διεπαφή, ικανή να προβάλλει σε πραγματικό χρόνο πληροφορίες κυκλοφορίας, προειδοποιήσεις και δεδομένα πλοήγησης, ενισχύοντας την αντιληπτική επίγνωση, την εργονομία και την οδική ασφάλεια.

Η διεπαφή χρήστη Επαυξημένης Πραγματικότητας (Augmented Reality User Interface – AR UI) αποτελεί το τελικό και πλέον κρίσιμο επίπεδο αλληλεπίδρασης μεταξύ του συστήματος και του οδηγού, λειτουργώντας ως σύνδεσμος ανάμεσα στον αλγοριθμικό πυρήνα επεξεργασίας δεδομένων και στον τελικό χρήστη. Η αρχιτεκτονική της AR UI ενοποιεί δεδομένα από αισθητήρες LiDAR, κάμερες, συστήματα θέσης OXTS, καθώς και αλγορίθμους μηχανικής μάθησης (όπως το YOLOv11 για ανίχνευση αντικειμένων), αποδίδοντάς τα με οπτική μορφή στο πεδίο αντίληψης του οδηγού.

Η υλοποίηση βασίζεται σε τεχνικές overlay και blending, εξασφαλίζοντας οπτική καθαρότητα και ελαχιστοποίηση της απόσπασης προσοχής. Ο σχεδιασμός της στοχεύει σε μια δυναμική, προσαρμόσιμη και εργονομική εμπειρία, που ενισχύει την επίγνωση της οδικής σκηνής χωρίς να υπερφορτώνει το οπτικό πεδίο. Η διεπαφή απεικονίζει σε πραγματικό χρόνο κρίσιμες πληροφορίες, όπως ανιχνευμένα αντικείμενα, εκτιμώμενες αποστάσεις, κατευθύνσεις κίνησης, σημάνσεις πλοήγησης και

ζώνες κινδύνου, συμβάλλοντας έτσι στην πρόληψη ατυχημάτων και στην ενίσχυση της οδικής ασφάλειας.

Μεταξύ των βασικών χαρακτηριστικών της διεπαφής συγκαταλέγονται η απεικόνιση αντικειμένων μέσω bounding boxes, wireframes ή εικονιδίων (sprites) για πιο φιλική και κατανοητή οπτικοποίηση, η αναπαράσταση βάθους (Depth Maps) με χρωματική κωδικοποίηση αποστάσεων και δυνατότητα ένθεσης στο πραγματικό περιβάλλον, καθώς και η παρουσίαση ετικετών και μετρικών που περιλαμβάνουν το είδος του αντικειμένου (όπως όχημα ή πεζός), την εκτιμώμενη απόσταση και την κατεύθυνσή του. Επιπλέον, ενσωματώνονται πληροφορίες πλοήγησης και προειδοποιήσεις μέσω επικαλυπτόμενων ενδείξεων λωρίδων κυκλοφορίας, κατευθύνσεων πορείας και σημάνσεων επικίνδυνων περιοχών, όπως ζώνες σύγκρουσης ή περιοχές μειωμένης ορατότητας, συμβάλλοντας στην ενίσχυση της αντίληψης και της ασφάλειας του οδηγού.

Τέλος, μέσω δυναμικής διαχείρισης διαφάνειας και ιεράρχησης οπτικών επιπέδων (layering), η διεπαφή προσαρμόζει την οπτική προτεραιότητα των στοιχείων ανάλογα με την απόσταση, την ταχύτητα και τη σημασία του αντικειμένου, επιτυγχάνοντας ισορροπία μεταξύ πληροφορίας και ασφάλειας. Η AR UI, συνεπώς, αποτελεί βασικό μηχανισμό αντίληψης και υποστήριξης αποφάσεων, καθιστώντας το περιβάλλον οδήγησης πιο έξυπνο, ασφαλές και εργονομικά αποδοτικό.

Συμπερασματικά, σύμφωνα με όλα τα παραπάνω μια συνοπτική διαδικασία του συστήματος θα έχει ως εξής:

- Αρχικοποίηση Συστήματος

Η αρχικοποίηση του συστήματος, πρακτικά διέπεται από τα εξής 4 βήματα: Την εκκίνηση της εφαρμογής, την αρχικοποίηση καθολικών μεταβλητών, κλάσεων και σταθερών (config, thresholds, camera/LiDAR/OXTS params), την φόρτωση πόρων στην μνήμη (sprite images, 3D assets) και την δημιουργία/φόρτωση προσομοιωμένων ροών δεδομένων (JSON).

- Συλλογή Δεδομένων ανά Καρέ

Σε αυτό το στάδιο, διενεργείται η πολυσensθητηριακή απόκτηση (Multi-Sensor Data Acquisition) δεδομένων: RGB frames (κάμερα), LiDAR point cloud, και GNSS/INS (OXTS) για θέση, προσανατολισμό και χρονισμό (PPS). Όλα τα δεδομένα υφίστανται συγχρονισμό μέσω κοινών χρονοσφραγίδων (timestamps).

- Προεπεξεργασία και Ευθυγράμμιση

Το στάδιο της προεπεξεργασίας ξεκινάει με ego-motion compensation στο point cloud, μετασχηματισμό GNSS (φ,λ,h) $\rightarrow$ ECEF $\rightarrow$ ENU (τοπικό καρτεσιανό), εξωτερική/εσωτερική βαθμονόμηση: ευθυγράμμιση LiDAR – Camera – Vehicle frame, και προεπεξεργασία εικόνας (resize/normalize) και point cloud (downsample/outlier removal).

- Παράλληλες Ροές Επεξεργασίας Αντίληψης

Στο στάδιο αυτό, εκτελούνται παράλληλα οι ροές επεξεργασίας αντίληψης, οι οποίες διαρθρώνονται σε τρεις διακριτές ενότητες: την Ανίχνευση Αντικειμένων, η οποία χρησιμοποιεί το μοντέλο YOLOv11 για την ανάλυση των RGB πλαισίων εικόνας, με σκοπό την εξαγωγή πλαισίων οριοθέτησης, σημασιολογικών ετικετών και βαθμών εμπιστοσύνης, έπειτα από εφαρμογή του αλγορίθμου Μη-Μέγιστης Καταστολής, τη Χαρτογράφηση Βάθους, η οποία προβάλλει τα 3D σημεία του νέφους LiDAR στον εικονοχώρο για τη δημιουργία χαρτών απόστασης ανά pixel , και τη διαχείριση των μεταφορτομένων 3D αντικειμένων, κατά την οποία λαμβάνονται εξωτερικά δεδομένα (REST - Web Socket - MQTT) και μετασχηματίζονται στο σύστημα αναφοράς του οχήματος, ακολουθούμενη από φιλτράρισμα frustum.

- Επίπεδα Απόδοσης

Το πέμπτο στάδιο αφορά τα επίπεδα απόδοσης της διεπαφής AR UI. Η διαδικασία εκκινεί με την απεικόνιση των ανιχνευμένων αντικειμένων, χρησιμοποιώντας πλαίσια οριοθέτησης, τρισδιάστατα μοντέλα ή ψηφιογραφικά αντικείμενα. Ακολουθεί η απόδοση του Επιπέδου Βάθους , το οποίο οπτικοποιείται μέσω χρωματικού χάρτη ή ρύθμισης της διαφάνειας. Εν συνεχεία, αποδίδονται τα 3D αντικείμενα (ως mesh ή sprite) μαζί με σημεία αναφοράς (anchors) και ετικέτες. Τέλος, εφαρμόζεται δυναμική διαφάνεια, η οποία προσαρμόζεται βάσει της σχετικής απόστασης και της σπουδαιότητας του κάθε στοιχείου, διασφαλίζοντας τη βέλτιστη εργονομία και την ελάχιστη οπτική φόρτιση.

- Σύνθεση Επιπέδων και Διεπαφή Χρήστη Ε.Π.

Το έκτο στάδιο περιλαμβάνει την τελική απόδοση και διαχείριση της Διεπαφής Χρήστη Επαυξημένης Πραγματικότητας (AR UI). Η διαδικασία εκκινεί

με τη σύνθεση επιπέδων (Layer Blending), κατά την οποία τα ψηφιακά γραφικά συνδυάζονται με το πλαίσιο εικόνας RGB του πραγματικού περιβάλλοντος. Ακολουθεί η προβολή κρίσιμων πληροφοριών, όπως οδηγίες πλοήγησης και προειδοποιήσεις κινδύνου. Τέλος, ενεργοποιούνται οι ελεγκτές UI για τη διαχείριση της διεπαφής (π.χ., εναλλαγή επιπέδων, ρύθμιση ορατότητας), καταλλήγοντας στην εμφάνιση της τελικής AR σκηνής στον χρήστη.

- **Καταγραφή και Τηλεμετρία**

Το στάδιο της καταγραφής και τηλεμετρίας εκκινεί με τη συστηματική καταγραφή των πλαισίων, των ανιχνεύσεων, των δεδομένων OXTS/GNSS/INS, και των χρόνων καθυστέρησης. Παράλληλα, διενεργείται η αποθήκευση κρίσιμων συμβάντων, ιδίως των περιπτώσεων ψευδώς θετικών/αρνητικών ανιχνεύσεων, τα οποία εξάγονται για μετέπειτα αξιολόγηση εκτός σύνδεσης.

- **Βρόχοι Απόδοσης και Ασφάλειας**

Στο όγδοο στάδιο της διαδικασίας, πραγματοποιείται η παρακολούθηση και βελτιστοποίηση απόδοσης, η οποία περιλαμβάνει τη μέτρηση των ρυθμών καρέ (FPS), του χρόνου καθυστέρησης, και της χρήσης πόρων (GPU/CPU usage). Ταυτόχρονα, διενεργείται ρύθμιση των παραμέτρων (input size, NMS, thresholds) για τη διασφάλιση της λειτουργίας πραγματικού χρόνου (real-time). Τέλος, καθορίζονται οι πολιτικές εφεδρείας για την αντιμετώπιση κρίσιμων καταστάσεων, όπως η απώλεια του GNSS, η χαμηλή ορατότητα ή οι υπερβάσεις χρόνου (timeouts).

- **Αξιολόγηση (Εκτός Σύνδεσης)**

Το τελικό στάδιο, η αξιολόγηση, επικεντρώνεται σε τρεις άξονες: (α) Ποσοτικές μετρικές ανίχνευσης (mAP, Precision/Recall). (β) Χρονισμός της συνολικής ροής επεξεργασίας (rendering/blending). (γ) Μελέτη χρηστών (user study) για την AR UI (αντίληψη ασφάλειας/ευχρηστία), με σκοπό την ποιοτική αξιολόγηση της αντίληψης ασφάλειας και της ευχρηστίας του συστήματος.

3.1.3 Άξονας II: Έξυπνος Σηματοδότης

Ο δεύτερος άξονας του συστήματος, με την κωδική ονομασία Axis 2: LiDAR – Traffic Light, επικεντρώνεται στην επεξεργασία τρισδιάστατων αντικειμένων στο υπολογιστικό νέφος (cloud backend) και στη μετάδοση αυτών προς τα συνεργαζόμενα

αυτόνομα οχήματα (CAVs). Ο άξονας αυτός ενσωματώνει τις ακόλουθες βασικές λειτουργικότητες:

- Μετατροπή γεωχωρικών δεδομένων GNSS σε Καρτεσιανές συντεταγμένες (GNSS to Cartesian)
- Μετασχηματισμός δεδομένων LiDAR σε τρισδιάστατα μοντέλα (LiDAR to 3D objects)
- Επικοινωνία με συνδεδεμένα αυτόνομα οχήματα μέσω χρήσης τεχνολογιών ασύρματης επικοινωνίας εγγύς πεδίου, όπως UWB (Ultra-Wideband) και BLE (Bluetooth Low Energy)
- Διασύνδεση με δίκτυα ευρείας κάλυψης, όπως Wi-Fi, 4G/5G και NB-IoT (Narrowband Internet of Things)
- Διαχείριση δεδομένων μέσω JSON δομών σε τοπικές / βραχυπρόθεσμες και κατανεμημένες / μακροπρόθεσμες NoSQL βάσεις τύπου εγγράφου (π.χ. MongoDB)

Η χρήση δεδομένων GNSS, μετασχηματισμένων σε καρτεσιανό σύστημα αναφοράς, επιτρέπει την ακριβή γεωχωρική αποτύπωση των τρισδιάστατων μοντέλων που προκύπτουν από τη σάρωση LiDAR, τα οποία προβάλλονται ως ψηφιακά δίδυμα (digital twins) στο περιβάλλον.

Το σύστημα αξιοποιεί τεχνολογίες ασύρματης επικοινωνίας μικρής εμβέλειας, όπως UWB (Ultra-Wideband) και BLE (Bluetooth Low Energy), για την ανταλλαγή δεδομένων σε πραγματικό χρόνο μεταξύ υποδομών και οχημάτων που βρίσκονται σε άμεση γειτνίαση. Οι τεχνολογίες αυτές προσφέρουν υψηλή ακρίβεια εντοπισμού, χαμηλή κατανάλωση ενέργειας και περιορισμένες καθυστερήσεις, καθιστώντας τις κατάλληλες για εφαρμογές τοπικής αντίληψης και συγχρονισμένης λειτουργίας μεταξύ κόμβων του συστήματος.

Για τη μεταφορά μεγαλύτερων όγκων πληροφορίας και τη διασύνδεση με το υπολογιστικό νέφος, το σύστημα υποστηρίζει επικοινωνία μέσω δικτύων ευρείας ζώνης, όπως Wi-Fi, 4G/5G και NB-IoT (Narrowband Internet of Things). Οι τεχνολογίες αυτές επιτρέπουν τη συνεχή αποστολή δεδομένων αισθητήρων και μεταδεδομένων σε cloud πλατφόρμες ή άλλες υποδομές, διασφαλίζοντας υψηλό εύρος ζώνης, αξιοπιστία στη σύνδεση, καθώς και δυνατότητα κλιμάκωσης του συστήματος σε αστικές περιοχές μεγάλης πυκνότητας.

Η αποθήκευση και διαχείριση των δεδομένων πραγματοποιείται σε δομές τύπου JSON (JavaScript Object Notation), εντός τοπικών ή κατανεμημένων βάσεων δεδομένων τύπου εγγράφου (Document-Oriented Databases – DoDBMS), ανάλογα με τον χρονικό ορίζοντα της χρήσης. Για βραχυχρόνια αποθήκευση, τα δεδομένα αποθηκεύονται σε τοπικές βάσεις ή στη μνήμη του συστήματος με στόχο την ταχεία προσπέλαση και την υποστήριξη σε επεξεργασίες πραγματικού χρόνου (π.χ. αντιδράσεις σε συμβάντα ή τοπική απόδοση οπτικής πληροφορίας). Αντίθετα, η μακροχρόνια αποθήκευση πραγματοποιείται σε κατανεμημένες βάσεις με δυνατότητες κλιμακούμενης αποθήκευσης, αναπαραγωγής (replication) και υψηλής διαθεσιμότητας, προκειμένου να διασφαλιστεί η αξιοπιστία, η ανάκτηση ιστορικών δεδομένων, και η δυνατότητα μεταγενέστερης ανάλυσης (όπως μοντέλα πρόβλεψης ή αναγνώρισης προτύπων). Η επιλογή του τύπου αποθήκευσης εξαρτάται από τη φύση των δεδομένων (αισθητήρες, 3D μοντέλα, μεταδεδομένα), την απαιτούμενη χρονική διάρκεια διατήρησής τους, και τις επιδόσεις που απαιτούνται σε κάθε επιμέρους υποσύστημα.

3.1.4 Άξονας III: Υποδομή Νέφους

Ο τρίτος άξονας του προτεινόμενου συστήματος διαφοροποιείται σημαντικά από τους δύο προηγούμενους, καθώς σχετίζεται με την αξιοποίηση τεχνολογιών υπολογιστικού νέφους (cloud computing). Στο πλαίσιο της παρούσας διπλωματικής εργασίας, το cloud χρησιμοποιείται κυρίως για τη συλλογή και αποθήκευση δεδομένων, τα οποία μπορούν να αξιοποιηθούν σε μεταγενέστερες αναλύσεις. Ενδεικτικά, η αναγνώριση προτύπων (pattern recognition) εφαρμόζεται με στόχο τον εντοπισμό περιοχών με αυξημένη κινητικότητα πεζών υπό συνθήκες χαμηλής ορατότητας, με απώτερο σκοπό την προληπτική διαχείριση της ασφάλειας σε διασταυρώσεις υψηλού κινδύνου.

Τα δεδομένα αυτά αποθηκεύονται σε κατανεμημένες βάσεις δεδομένων τύπου εγγράφου, οι οποίες υποστηρίζουν δυνατότητες όπως κατακερματισμός δεδομένων (sharding), αντιγραφή δεδομένων (replication) και αρθρωτό (modular) σχεδιασμό, εξασφαλίζοντας υψηλό ρυθμό μετάδοσης μέσω δρομολογητών ερωτημάτων (query routers) και εξυπηρετητών ρυθμίσεων (config servers).

Επιπλέον, τα δεδομένα καθίστανται προσβάσιμα από πράκτορες νέφους (cloud agents), δηλαδή αυτοματοποιημένα λογισμικά συστήματα, τα οποία είτε προωθούν

τα δεδομένα προς περαιτέρω ανάλυση από εξωτερικά συστήματα, είτε πραγματοποιούν άμεσες διαδικασίες ανάλυσης επί των διαθέσιμων πληροφοριών. Παράλληλα, παρέχονται διεπαφές προγραμματισμού εφαρμογών (APIs) για τη διασύνδεση του συστήματος με εξωτερικές υποδομές και πλατφόρμες, όπως είναι, ενδεικτικά, τα συστήματα έξυπνων πόλεων (smart cities) που δύναται να χρησιμοποιούν οι υπηρεσίες της αστυνομίας, της πυροσβεστικής και άλλα.

3.1.5 Επεκτασιμότητα

Η αρχιτεκτονική του συστήματος έχει σχεδιαστεί με γνώμονα τη επεκτασιμότητα και τη διαλειτουργικότητα, επιτρέποντας τη μελλοντική του διασύνδεση με tactile maps (απτικές χαρτογραφικές απεικονίσεις). Η ενσωμάτωση αυτών των χαρτών θα μπορούσε να προσφέρει πολυαισθητηριακή ανατροφοδότηση, συνδυάζοντας οπτικά και απτικά δεδομένα για την ενίσχυση της αντίληψης του περιβάλλοντος, ιδίως σε συνθήκες μειωμένης ορατότητας ή για χρήστες με οπτικές ιδιαιτερότητες. Μέσω αυτής της διασύνδεσης, το σύστημα θα μπορούσε να υποστηρίξει νέες μορφές διεπαφής ανθρώπου-μηχανής, καθιστώντας το πλαίσιο της οδικής ασφάλειας πιο προσβάσιμο, ευφύες και προσαρμοστικό.

Τέλος, ως μια επιπλέον βασική λειτουργικότητα του συστήματος προτείνεται το CAV2CAV (Connected and Automated Vehicle to Connected and Automated Vehicle), το οποίο περιγράφει την επικοινωνία μεταξύ δύο ή περισσότερων αυτόνομων/συνδεδεμένων οχημάτων. Αποτελεί εξειδικευμένη μορφή του V2V (Vehicle-to-Vehicle) και εντάσσεται στην λογική του V2X (Vehicle-to-Everything). Ειδικότερα, η επικοινωνία CAV2CAV επεκτείνει το παραδοσιακό μοντέλο V2V παρέχοντας την δυνατότητα συνεργατικής αντίληψης και ανταλλαγής πληροφορίας αισθητήρων σε πραγματικό χρόνο. Ακόμη, μέσω της ανταλλαγής δεδομένων θέσης, της ταχύτητας και των ανιχνεύσεων αντικειμένων, τα οχήματα αποκτούν ικανότητα αντίληψης πέρα από την οπτική γραμμή, γεγονός που ενισχύει την ασφάλεια και μειώνει τον κίνδυνο συγκρούσεων. Παράλληλα, επιτρέπει πιο αποδοτικές στρατηγικές πλοήγησης, όπως το platooning και την συνεργατική διαχείριση διασταυρώσεων. Η ενσωμάτωση αυτής της πληροφορίας στην AR διεπαφή παρέχει στον οδηγό μια ενισχυμένη αίσθηση του περιβάλλοντος, καθιστώντας την οδηγική του εμπειρία πιο ασφαλή, εργονομική, καθώς και αρκετά αποδοτική.

3.2 Οπτικοποίηση Αντικειμένων Ενδιαφέροντος

Η οπτικοποίηση πληροφοριών σε περιβάλλον Επαυξημένης Πραγματικότητας (ΕΠ) για την οδική ασφάλεια αποτελεί κρίσιμο παράγοντα για την αντίληψη του χώρου και την έγκαιρη αναγνώριση κινδύνων από τον οδηγό. Η επιλογή της κατάλληλης μορφής οπτικοποίησης επηρεάζει άμεσα τη γνωστική επιβάρυνση, τον χρόνο αντίδρασης και το επίπεδο κατανόησης του περιβάλλοντος. Οι διαφορετικοί τρόποι αναπαράστασης – από απλά σημεία έως πλήρη τρισδιάστατα μοντέλα – παρουσιάζουν αντίστροφη αναλογία μεταξύ πολυπλοκότητας και ευκολίας αντίληψης: όσο πιο σύνθετη είναι η οπτική μορφή, τόσο αυξάνεται η πληροφοριακή πληρότητα αλλά μειώνεται η ταχύτητα αναγνώρισης απειλών. Αντίθετα, οι πιο αφαιρετικές μορφές προσφέρουν άμεση αναγνωρισιμότητα, εις βάρος όμως της λεπτομέρειας και του χωρικού ρεαλισμού. Στις παρακάτω ενότητες παρουσιάζονται οι κύριες κατηγορίες οπτικοποίησης που εφαρμόστηκαν στο πλαίσιο της παρούσας εργασίας, με έμφαση στον ρόλο τους στη διευκόλυνση της αντίληψης κινδύνου και της αποτελεσματικής υποστήριξης του οδηγού σε συνθήκες ΕΠ.

3.2.1 Οπτικοποίηση ως Σημείο

Η οπτικοποίηση ως σημείο αποτελεί την πιο στοιχειώδη μορφή απεικόνισης δεδομένων στον χώρο, όπου κάθε στοιχείο αντιπροσωπεύεται από μία μοναδική χωρική συντεταγμένη (x, y, z). Η προσέγγιση αυτή χρησιμοποιείται κυρίως για την απεικόνιση μεμονωμένων θέσεων, ανιχνεύσεων ή συμβάντων, όπως η αναγνώριση ενός αντικειμένου ή η ένδειξη μιας πιθανής απειλής στο περιβάλλον. Στο πλαίσιο της Επαυξημένης Πραγματικότητας (ΕΠ) για την οδική ασφάλεια, η συγκεκριμένη αναπαράσταση παρέχει άμεση και γρήγορη κατανόηση του σημείου κινδύνου, χάρη στη χαμηλή οπτική πολυπλοκότητα και την περιορισμένη πληροφοριακή φόρτιση. Ωστόσο, η απλότητα αυτή ενέχει τον κίνδυνο μειωμένης χωρικής επίγνωσης, καθώς το σημείο δεν μεταφέρει πληροφορίες σχετικά με το μέγεθος, την κατεύθυνση ή τη δυναμική του κινδύνου. Έτσι, η αναγνώριση παραμένει μεν ταχεία, αλλά λιγότερο περιγραφική. Επέκταση αυτής της στρατηγικής αποτελεί η οπτικοποίηση ως δίσκος ή σφαίρα, φέροντας τα ίδια μειονεκτήματα, ενώ καθιστώντας την περισσότερο κατανοητή.

3.2.2 Οπτικοποίηση ως Τρισδιάστατο Πλαίσιο Οριοθέτησης

Η οπτικοποίηση ως τρισδιάστατο πλαίσιο οριοθέτησης (3D bounding box) προσφέρει μια πιο πλούσια αναπαράσταση, επιτρέποντας στον οδηγό ή στο σύστημα ΕΠ να αντιληφθεί με μεγαλύτερη ακρίβεια τη θέση, τον όγκο και τα όρια ενός ανιχνευμένου αντικειμένου. Αυτή η μορφή απεικόνισης ενισχύει σημαντικά την αντιληπτική σαφήνεια και συμβάλλει στην πρόβλεψη πιθανών κινδύνων, αφού καθιστά άμεσα ορατό το πεδίο κατάληψης κάθε αντικειμένου στο χώρο. Παρόλο που είναι πιο υπολογιστικά απαιτητική από την απλή σημειακή απεικόνιση, παραμένει σχετικά ελαφριά και λειτουργεί σε πραγματικό χρόνο, προσφέροντας μια ισορροπία μεταξύ ακρίβειας και ταχύτητας αναγνώρισης. Στην πράξη, η μορφή αυτή διευκολύνει την έγκαιρη λήψη αποφάσεων του οδηγού, χωρίς να υπερφορτώνει το οπτικό πεδίο.

3.2.3 Οπτικοποίηση ως Δισδιάστατο Φύλλο Ψηφιογραφικών Αντικειμένων

Η οπτικοποίηση ως δισδιάστατο φύλλο ψηφιογραφικών αντικειμένων (2D spritesheet) χρησιμοποιεί συμβολικά γραφικά στοιχεία για την αναπαράσταση γεγονότων, οχημάτων ή προειδοποιήσεων. Η επιλογή αυτής της μεθόδου στο περιβάλλον ΕΠ προάγει την ευκολία και ταχύτητα αναγνώρισης επικίνδυνων καταστάσεων, καθώς τα εικονίδια είναι άμεσα αναγνωρίσιμα και συνδέονται με καθιερωμένους οδικούς συμβολισμούς. Παράλληλα, η δισδιάστατη φύση τους διατηρεί την απλότητα του οπτικού πεδίου, μειώνοντας τη γνωστική επιβάρυνση του οδηγού. Ωστόσο, η αφαιρετικότητα αυτής της απεικόνισης συνεπάγεται περιορισμένη χωρική πληροφόρηση, γεγονός που καθιστά τη μέθοδο περισσότερο ενδεικτική παρά αναλυτική. Παρ' όλα αυτά, η υψηλή αντιληπτική της απόδοση την καθιστά εξαιρετικά χρήσιμη για γρήγορη προειδοποίηση κινδύνων σε εφαρμογές οδικής ασφάλειας.

3.2.4 Οπτικοποίηση ως Τρισδιάστατο Μοντέλο

Η οπτικοποίηση ως τρισδιάστατο μοντέλο (3D model visualization) αντιπροσωπεύει την πιο ρεαλιστική και ολοκληρωμένη μορφή αναπαράστασης. Μέσω γεωμετρικής απεικόνισης, υλικών και φωτισμού, παρέχει υψηλό βαθμό οπτικής πιστότητας και προσφέρει στον οδηγό την αίσθηση πλήρους χωρικής αντίληψης του περιβάλλοντος. Στο πλαίσιο της ΕΠ, αυτή η μέθοδος συμβάλλει στην ακριβή αξιολόγηση σύνθε-

των σκηνών, όπως η εκτίμηση αποστάσεων, ταχύτητας και κατεύθυνσης κινούμενων αντικειμένων. Ωστόσο, η αυξημένη πολυπλοκότητα και ο ρεαλισμός ενδέχεται να καθυστερήσουν ελαφρώς την αντίληψη άμεσων κινδύνων, λόγω του αυξημένου όγκου οπτικών πληροφοριών. Βελτιστοποίηση των 3D model μέσω ελαχιστοποίησης της πολυπλοκότητας των πολυγώνων μπορεί να βελτιώσει την απόκριση του συστήματος μέσω διαδεδομένων συστημάτων δημιουργίας και επεξεργασίας μοντέλων με κύριο αντιπρόσωπο το Blender. Συνεπώς, αν και προσφέρει ανώτερη εμπειρία και ακρίβεια, απαιτεί προσεκτική σχεδίαση ώστε να μην επιβαρύνει τη γνωστική διεργασία του οδηγού σε κρίσιμες στιγμές.

3.2.5 Προσέγγιση Οπτικοποίησης

Για την αναπαράσταση και απεικόνιση των δεδομένων στο προτεινόμενο σύστημα επιλέχθηκε η προσέγγιση άμεσης επεξεργασίας εικόνας (image array/matrix rendering) έναντι της κλασικής διαδικασίας γραφικής απεικόνισης μέσω του OpenGL rendering pipeline. Στην κλασική προσέγγιση, η απεικόνιση περνά από όλα τα στάδια γραφικής επεξεργασίας — όπως μετασχηματισμοί κορυφών (vertex transformations), ραστεροποίηση (rasterization) και σκίαση αποσπασμάτων (fragment shading) — γεγονός που εξασφαλίζει υψηλή οπτική ποιότητα και ρεαλιστική απεικόνιση, αλλά ταυτόχρονα αυξάνει σημαντικά το υπολογιστικό κόστος.

Αντίθετα, η μέθοδος που υιοθετήθηκε βασίζεται στην απευθείας προβολή των pixel arrays στον framebuffer, παρακάμπτοντας ενδιάμεσες διεργασίες όπως το clipping, ο φωτισμός (lighting) ενώ για το blending προτείνεται συνάρτηση μίξης καναλιού διαφάνειας με γνώμονα την απόσταση των αντικειμένων ενδιαφέροντος από το όχημα. Η συνάρτηση φέρει ως αποτέλεσμα τα πλησιέστερα αντικείμενα να αχνοφαίνονται καθιστώντας τα λιγότερο επεμβατικά στη σκηνή, ενώ αντικείμενα σε μεγάλη απόσταση να εικονίζονται εντονότερα, γεγονός που συνεπάγεται την αναγνώριση αντικειμένων που υπερκαλύπτονται από πλησιαιέστερα αντικείμενα. Με αυτόν τον τρόπο, η διαδικασία οπτικοποίησης καθίσταται σημαντικά ταχύτερη, καθώς περιορίζεται στην αντιστοίχιση δεδομένων RGB ή depth arrays στην έξοδο, επιτυγχάνοντας υψηλή απόδοση σε πραγματικό χρόνο.

Αν και η συγκεκριμένη προσέγγιση υστερεί ελαφρώς ως προς τη γραφική ρεαλιστικότητα, εντούτοις εξυπηρετεί πλήρως τον λειτουργικό στόχο του συστήματος, ο οποίος επικεντρώνεται στην ταχύτητα επεξεργασίας και την απόκριση σε πραγ-

ματικό χρόνο, στοιχεία κρίσιμα για εφαρμογές επαυξημένης πραγματικότητας και
αυτόνομης οδήγησης.

ΚΕΦΑΛΑΙΟ 4

ΠΡΟΤΕΙΝΟΜΕΝΟ ΣΥΣΤΗΜΑ ΕΠΑΥΞΗΜΕΝΗΣ ΠΡΑΓΜΑΤΙΚΟΤΗΤΑΣ ΣΥΝΔΕΔΕΜΕΝΩΝ ΑΥΤΟΝΟΜΩΝ ΟΧΗΜΑΤΩΝ

-
- 4.1 Εισαγωγή
 - 4.2 Συλλογή Δεδομένων
 - 4.3 Εφύια Συστήματος
 - 4.4 Διαδικασίες Διαγραμμάτων Ροής
 - 4.5 Παραδείγματα Λειτουργείας
-

4.1 Εισαγωγή

Στο πλαίσιο της παρούσας διπλωματικής εργασίας, και με γνώμονα την προτεινόμενη αρχιτεκτονική, αναπτύχθηκε μια πειραματική υλοποίηση ως απόδειξη της ιδέας. Η υλοποίηση αυτή βασίστηκε στον Άξονα I (CAV AR) όπως περιγράφηκε σε προηγούμενο κεφάλαιο και θέτει ως πρωταρχικό στόχο την απόδειξη της επίδρασης της Επαυξημένης Πραγματικότητας στην οδηγική ασφάλεια.

Ως εκ τούτου, η έμφαση δόθηκε στην λειτουργικότητα και την αξιολόγηση της επίδρασης, παραβλέποντας εσκεμμένα την βελτιστοποίηση ή την προεκπαίδευση των μοντέλων μηχανικής μάθησης, καθώς και την πολυπλοκότητα της δικτύωσης των επιμέρους συστημάτων με τους Έξυπνους Σηματοδότες.

Στη συνέχεια, ακολουθεί η ανάλυση των δεδομένων που αξιοποιεί η συγκεκριμένη υλοποίηση, η αναφορά στο επιλεγμένο μοντέλο μηχανικής μάθησης, τα διαγράμματα ροής της υλοποίησης ως επεξήγηση των λειτουργιών της, καθώς και παραδείγματα χρήσης του συστήματος με συνοδευτικά στιγμιότυπα οθόνης.

4.2 Συλλογή Δεδομένων

Για την ανάπτυξη, αξιολόγηση και οπτικοποίηση του προτεινόμενου Συστήματος Επαυξημένης Πραγματικότητας για την οδική ασφάλεια, χρησιμοποιήθηκε το σύνολο δεδομένων KITTI-360, ένα από τα πλέον αναγνωρισμένα και ευρέως χρησιμοποιούμενα datasets στον χώρο της αυτόνομης οδήγησης και της υπολογιστικής όρασης. Το KITTI-360 παρέχει πραγματικά δεδομένα οδήγησης από αστικά και περιαστικά περιβάλλοντα, συλλεγμένα μέσω πολυαισθητηριακής πλατφόρμας οχήματος-παρατηρητή.

Στο πλαίσιο της παρούσας υλοποίησης αξιοποιήθηκε η σειρά 00 (sequence 00) του συνόλου δεδομένων, και συγκεκριμένα η κάμερα Cam0, η οποία παρέχει διορθωμένα (rectified) RGB δεδομένα υψηλής ανάλυσης. Η συγκεκριμένη ροή εικόνων χρησιμοποιήθηκε ως οπτική είσοδος στο μοντέλο μηχανικής μάθησης για την ανίχνευση και ταξινόμηση αντικειμένων ενδιαφέροντος σε πραγματικό χρόνο. Παράλληλα, αξιοποιήθηκαν τα αντίστοιχα δεδομένα από τον αισθητήρα LiDAR, τα οποία περιλαμβάνουν τρισδιάστατες νεφώσεις σημείων (point clouds) που αντιπροσωπεύουν με ακρίβεια το χωρικό περίγραμμα του περιβάλλοντος. Τα δεδομένα αυτά χρησιμοποιήθηκαν για τη συγχώνευση οπτικής και χωρικής πληροφορίας, επιτρέποντας την εκτίμηση αποστάσεων, την προβολή βάθους και την παραμετροποίηση της οπτικοποίησης της Επαυξημένης Πραγματικότητας βάσει πραγματικών γεωμετρικών σχέσεων.

Επιπροσθέτως, ενσωματώθηκαν τα δεδομένα θέσης και προσανατολισμού (Oxts), τα οποία παρέχουν πληροφορίες GPS και IMU, όπως γεωγραφικό πλάτος, μήκος, υψόμετρο, ταχύτητα, επιτάχυνση και γωνιακό ρυθμό. Οι πληροφορίες αυτές επιτρέπουν τον συγχρονισμό και την ακριβή γεωτοποθέτηση των εικόνων και των LiDAR μετρήσεων, διασφαλίζοντας την χωροχρονική συνέπεια όλων των αισθητηριακών ροών.

Συνολικά, η υποενότητα δεδομένων που χρησιμοποιήθηκε αποτελείται από:

- Περίπου 4.500 RGB εικόνες (Cam0, rectified) ανάλυσης 1408×376 pixels τριών καναλιών.
- Αντίστοιχες LiDAR νεφώσεις σημείων, με πάνω από 100.000 σημεία ανά καρέ μεγέθους περίπου 1.6 MiB ανά αρχείο.
- Oxts δεδομένα με ρυθμό 10 Hz, συνολικού όγκου περίπου 15 MB, που αντιστοιχούν σε πλήρη γεωαναφορά της διαδρομής, με περίπου 460 B αντιστοίχισης δεδομένων ανά καρέ.

Παράλληλα, αξιοποιήθηκαν προϋπάρχοντα δεδομένα πλαισίων οριοθέτησης (bounding boxes), τα οποία χρησιμοποιήθηκαν ως πιθανές εισροές στο σύστημα. Από το σύνολο αυτό επιλέχθηκαν οι κατηγορίες “στατικά οχήματα” και “πεζοί”, καθώς αντιπροσωπεύουν κρίσιμες οντότητες για την ανάλυση οδικών κινδύνων και την ενίσχυση της προληπτικής ασφάλειας.

Το αρχικό αρχείο καταγραφής δεδομένων σε μορφή XML (μέγεθος περίπου 8.33 MiB) μετατράπηκε σε μορφή JSON, ώστε να καταστεί συμβατό με το πρωτόκολλο επικοινωνίας των Έξυπνων Σηματοδοτών στο σύστημα. Η νέα δομή αποτελείται από πίνακα εγγράφων που περιέχουν μεταδεδομένα όπως: αριθμός ευρετηρίου (index), ετικέτα αντικειμένου (label), δυναμικό αντικείμενο, και ο πίνακας μετασχηματισμού του αντικειμένου επιτρέποντας την άμεση ενσωμάτωση των ανιχνεύσεων στο υποσύστημα Επαυξημένης Πραγματικότητας. Ο πίνακας μετασχηματισμού περιλαμβάνει τα μεταδεδομένα διάστασης (αριθμός στηλών και γραμμών) καθώς και τον τύπο των δεδομένων που τον απαρτίζουν. Ο ίδιος ο πίνακας είναι τετραδιάστατος (4×4) και αποτελείται από έναν υποπίνακα περιστροφής 3×3, έναν διάνυσμα μετατόπισης 1×3, ενώ ολοκληρώνεται με τη γραμμή ομογενών συντεταγμένων [0 0 0 1], η οποία εξασφαλίζει τη μετάβαση σε ομογενείς συντεταγμένες και καθιστά δυνατή τη σύνθεση περιστροφής και μετατόπισης σε έναν ενιαίο μετασχηματισμό.

Ενδεικτική μορφή καταγραφής JSON:

```
[
{
  "index": "4381",
  "label": "car",
  "dynamic": "0",
  "transform": {
    "rows": "4",
```

```

"cols": "4",
"dt": "f",
"data": [ 2.92481422, -1.63844514, 0.0002430073, 966.490295,
4.16620302, 1.15024292, -0.0007323388, 3784.74097,
0.0015391032, 0.0008158242, 1.52057445, 114.976791,
0.0, 0.0, 0.0, 1.0 ]
}
},
...]
```

Για την ολοκλήρωση και έλεγχο ορθής λειτουργίας χρησιμοποιήθηκαν οι υπόλοι-
πες σειρές του συνόλου δεδομένων KITTI 360.

Δείγματα φύλλων ψηφιογραφικών αντικειμένων λήφθηκαν από τις διαδικτυακές
πηγές 'Freepik'[22] και 'Pngwing'[23] καθώς και το τρισδιάστατο μοντέλο «Low
Poly Lada Vesta» του χρήστη 'Kulonee'[24] υπό την άδεια creative commons license
CC-BY-4.0.

4.3 Εφύια Συστήματος

Στα πλαίσια της παρούσης υλοποίησης χρησιμοποιήθηκε το μοντέλο YOLO11n για
την ενεργή ανίχνευση των αντικειμένων ενδιαφέροντος σε καταστάσεις πραγματικού
χρόνου.

Το YOLO11 (You Only Look Once v11) αποτελεί τη νεότερη εξέλιξη της οικο-
γένειας μοντέλων YOLO, σχεδιασμένη για ανίχνευση αντικειμένων σε πραγματικό
χρόνο με βελτιωμένη ακρίβεια και αποδοτικότητα. Σε σχέση με προηγούμενες εκ-
δόσεις, το YOLO11 εισάγει αναβαθμισμένη αρχιτεκτονική backbone και head, που
επιτρέπει καλύτερη εξαγωγή πολυκλιμακικών χαρακτηριστικών και πιο σταθερή
σύγκλιση κατά την εκπαίδευση. Επιπλέον, ενσωματώνει βελτιστοποιημένες τεχνι-
κές εκπαίδευσης και αποδοτικότερους μηχανισμούς υπολογισμού, μειώνοντας το
χρόνο επεξεργασίας ανά καρέ χωρίς απώλεια στην ακρίβεια. Το αποτέλεσμα είναι
ένα σύστημα ικανό να επιτυγχάνει υψηλότερες τιμές mAP με λιγότερους υπολογι-
στικούς πόρους, καθιστώντας το ιδανικό για εφαρμογές πραγματικού χρόνου, όπως
αυτόνομη οδήγηση, επιτήρηση και επαυξημένη πραγματικότητα.

Το YOLO11n (nano) είναι το πιο ελαφρύ μοντέλο της οικογένειας YOLO11, σχε-

διασμένο για υψηλή ταχύτητα και αποδοτικότητα σε συσκευές με περιορισμένους πόρους. Εκπαιδεύεται στο σύνολο δεδομένων COCO, χρησιμοποιεί εικόνες εισόδου 640×640 και με μέση τιμή ακρίβειας (mAP) υπολογισμένη για τιμές IoU από 0.50 έως 0.95 επιτυγχάνει 39.5 διατηρώντας παράλληλα εξαιρετικά χαμηλή καθυστέρηση (1.5 ms σε GPU και 2.6 ms σε ONNX). Με μόλις 6.5 B FLOPs και 2.6 M παραμέτρους, προσφέρει πραγματικό χρόνο απόδοσης, καθιστώντας το ιδανικό για εφαρμογές όπως φορητές συσκευές, edge computing και συστήματα οχημάτων όπου η ταχύτητα και η ενεργειακή αποδοτικότητα υπερέχουν έναντι της μέγιστης ακρίβειας.

Το μοντέλο προεκπαιδύεται στο σύνολο δεδομένων COCO και χρησιμοποιείται μόνο για συμπερασματολογία στην παρούσα υλοποίηση. Το COCO (Common Objects in Context) είναι ένα από τα πιο ευρέως χρησιμοποιούμενα σύνολα δεδομένων στη μηχανική μάθηση για ανίχνευση, τμηματοποίηση και αναγνώριση αντικειμένων. Περιλαμβάνει πάνω από 330.000 εικόνες και 1,5 εκατομμύριο επισημασμένα αντικείμενα σε 80 κατηγορίες, προσφέροντας πλούσια ποικιλία σε σκηνές της καθημερινότητας. Η ιδιαιτερότητά του έγκειται στην ποικιλία των περιβαλλόντων και των γωνιών λήψης, κάτι που το καθιστά ιδανικό για την εκπαίδευση ανθεκτικών και γενικεύσιμων μοντέλων όρασης υπολογιστή.

Οι κατηγορίες ενδιαφέροντος που προκύπτουν από την προεκπαίδευση είναι:

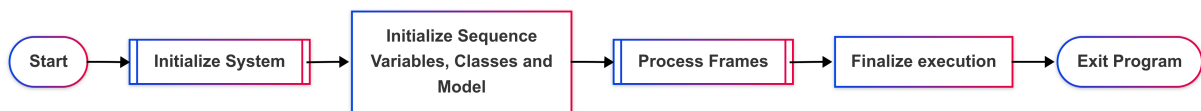
[0: 'pedestrian' (person), 1: 'bicycle', 2: 'car', 3: 'motorcycle', 5: 'bus', 11: 'stop'], με δυνατότητες επέκτασης σύμφωνα με την πιθανότητα κατηγοριών αντικειμένων να εμφανιστούν στο οδικό δίκτυο ως:

[0: 'pedestrian' (person), 1: 'bicycle', 2: 'car', 3: 'motorcycle', 5: 'bus', 6: 'train', 7: 'truck', 8: 'boat', 9: 'traffic light', 11: 'stop sign', 14: 'bird', 15: 'cat', 16: 'dog', 17: 'horse', 18: 'sheep', 19: 'cow', 21: 'bear', 32: 'sports ball', 36: 'skateboard', 39: 'bottle'].

4.4 Διαδικασίες Διαγραμμάτων Ροής

Στην συγκεκριμένη ενότητα, παραθέτουμε τα διαγράμματα ροής (flowcharts) από την εφαρμογή μας, αναλύοντας και συσχετίζοντας τα με την πραγματική απεικόνιση των καταστάσεων κατά την διάρκεια της λειτουργίας της εφαρμογής μας.

Η κύρια λογική του συστήματος παρουσιάζεται στο Διάγραμμα 4.1, το οποίο

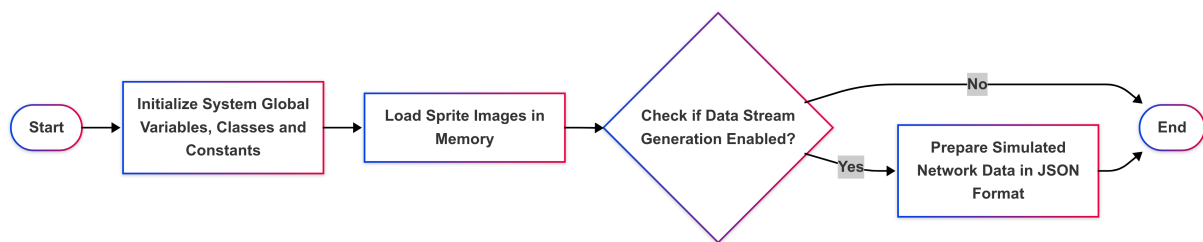


Σχήμα 4.1: Διάγραμμα Κύριας Διεργασίας.

περιγράφει τα στάδια εκτέλεσης της εφαρμογής στην υπολογιστική μονάδα ενός συνδεδεμένου αυτόνομου οχήματος. Κατά την εκκίνηση του συστήματος, ακολουθεί η διεργασία αρχικοποίησης, η οποία αναλύεται σε επόμενη παράγραφο. Στη συνέχεια, αρχικοποιούνται οι μεταβλητές της σκληνής βάσει του επιλεγμένου συνόλου δεδομένων, καθώς και οι αντίστοιχες κλάσεις και το μοντέλο μηχανικής μάθησης που χρησιμοποιείται κατά τη λειτουργία του συστήματος. Έπειτα, εκτελείται η επαναληπτική διεργασία επεξεργασίας ανά καρέ εικόνας, μέχρι να επιλεγεί η λειτουργία τερματισμού από τον χρήστη. Τότε εκτελούνται οι ενέργειες τερματισμού του συστήματος, με τελικό αποτέλεσμα τη διακοπή της εκτέλεσης της εφαρμογής.

Αναλύοντας το διαγράμμα ροής 4.2, αρχής γενημένης από το ξεκίνημα (Start), παρατηρούμε την προαναφερόμενη ροή εκτέλεσης αρχικοποίησης, η οποία ακολουθεί μια προκαθορισμένη διαδικασία προετοιμασίας και ενεργοποίησης δυναμικών λειτουργιών του συστήματος. Ακολούθως μετά την εκκίνηση της εφαρμογής έπεται η αρχικοποίηση καθολικών μεταβλητών, κλάσεων και σταθερών, οι οποίες αποτελούν την βάση για την διαχείριση της μνήμης, της λογικής ροής και της διαλειτουργικότητας μεταξύ των επί μέρους λειτουργιών. Στην συνέχεια, το σύστημα φορτώνει τα δυσδιάστατα φύλλα ψηφιογραφικών Αντικειμένων (spritesheet) στην διαθέσιμη μνήμη, τα οποία χρησιμοποιούνται για την απεικόνιση στοιχείων στο γραφικό περιβάλλον ή στο περιβάλλον προσομοίωσης. Ακολούθως, πραγματοποιείται ένας έλεγχος ενεργοποίησης της παραγωγής δεδομένων ροής, ο οποίος προσδιορίζει το αν η εφαρμογή θα προχωρήσει στην παραγωγή προσομοιωμένων δεδομένων δικτύου ή όχι. Εφόσον η επιλογή αυτή είναι ενεργή, το σύστημα δημιουργεί και διαμορφώνει τα δεδομένα σε μορφή JSON, ώστε να είναι άμεσα προσπελάσιμα και επεξεργάσιμα από άλλες διεπαφές ή υπηρεσίες. Εάν η παραγωγή δεδομένων δεν είναι ενεργοποιημένη, η ροή οδηγείται άμεσα στον τερματισμό. Αυτή η απλή αλλά ευέλικτη αρχιτεκτονική καθιστά την λύση κατάλληλη για δοκιμαστικά σενάρια, δημιουργία εικονικών δεδομένων και αυτοματοποιημένη επεξεργασία σε περιβάλλοντα προσομοίωσης ή έξυπνων συστημάτων.

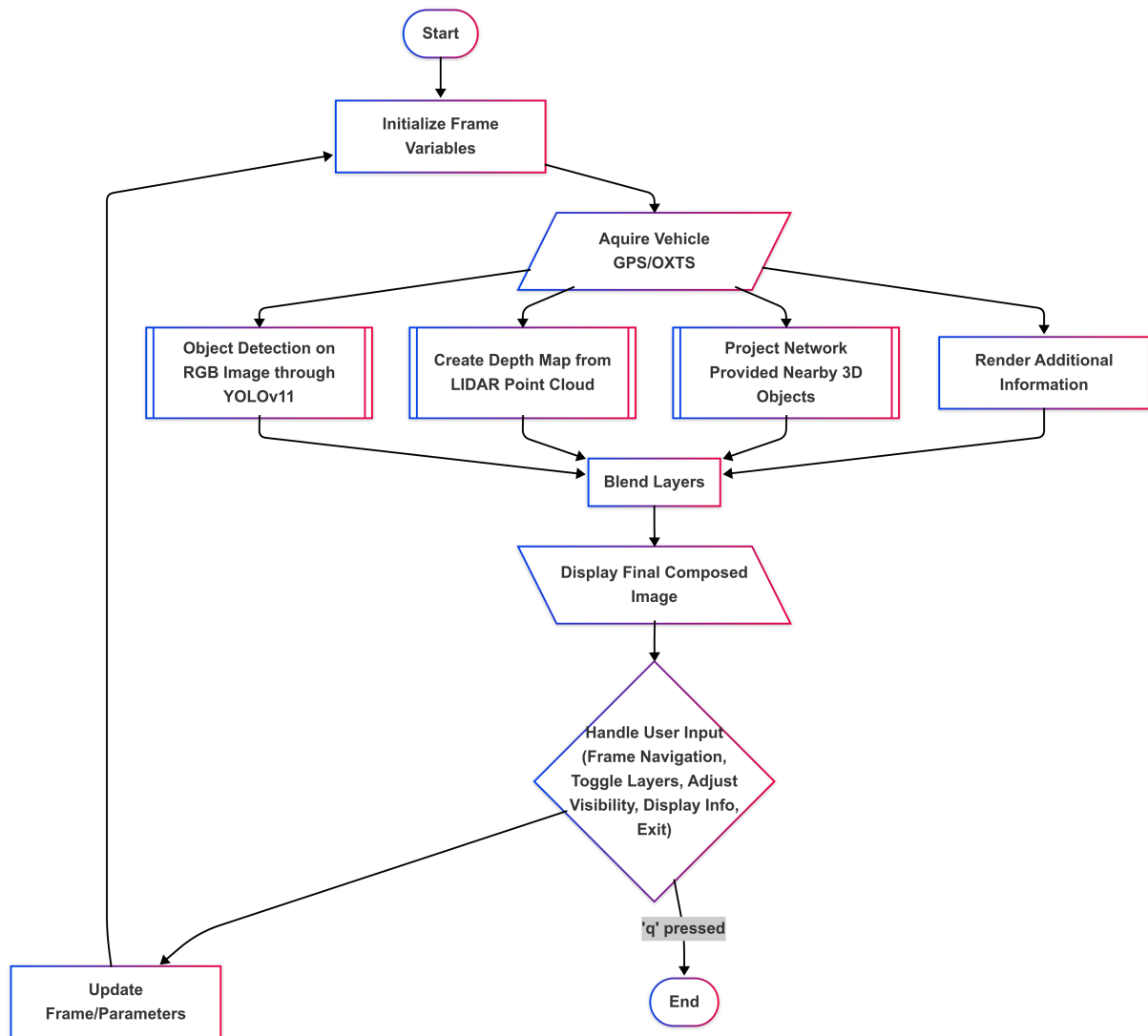
Το διαγράμμα ροής 4.3 παρουσιάζει την διεργασία επεξεργασίας ανά καρέ ει-



Σχήμα 4.2: Διάγραμμα Αρχικοποίησης Συστήματος.

κόννας. Ακολουθώντας, η λειτουργία της αρχικοποίησης μεταβλητών πλαισίου του συστήματος (Initialize Frame Variables), βασίζεται σε έναν επαναλαμβανόμενο κύκλο λήψης, ανάλυσης και σύνθεσης δεδομένων που προέρχονται από πολλαπλές πηγές. Κάθε κύκλος ξεκινά με την αρχικοποίηση των μεταβλητών του καρέ και την απόκτηση δεδομένων τοποθεσίας από αισθητήρες GPS/OXTS. Έπειτα, το σύστημα ενεργοποιεί τέσσερις επεξεργαστικές ροές: την ανίχνευση αντικειμένων μέσω YOLO v11 στην εικόνα RGB, την παραγωγή χάρτη βάθους από νέφος σημείων LIDAR, την προβολή κοντινών 3D αντικειμένων που έχουν ληφθεί από το δίκτυο, και την απόδοση συμπληρωματικών πληροφοριών όπως μετρήσεις ή σχόλια. Όλα τα παραπάνω ομαδοποιούνται σε μια σύνθετη εικόνα μέσω τεχνικής συνδυασμού επιπέδων (layer blending) και εμφανίζονται στον χρήστη. Ακόμη, η διαδικασία επιτρέπει την αλληλεπίδραση με τον τελικό χρήστη, παρέχοντας του την δυνατότητα να περιηγηθεί στο περιεχόμενο, να ενεργοποιήσει/απενεργοποιήσει ορατές τις πληροφορίες, να προσαρμόσει την απεικόνιση ή να εξέλθει. Ο βρόγχος συνεχίζεται μέχρι να πατηθεί η εντολή εξόδου, εξασφαλίζοντας δυναμική και συνεχώς ενημερωμένη αναπαράσταση του περιβάλλοντος.

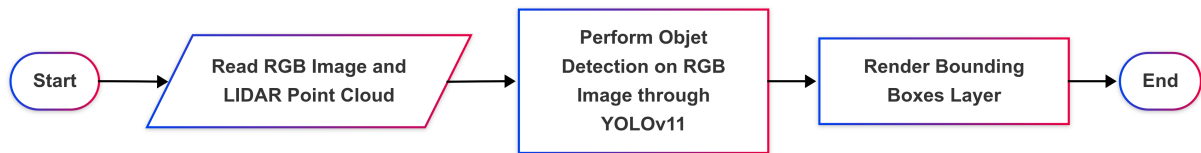
Μία από τις βασικές λειτουργίες του συστήματος, αφορά τον εντοπισμό αντικειμένων σε πραγματικό χρόνο από εικόνες RGB, αξιοποιώντας το εξελεγμένο δίκτυο YOLOv11. Επί του διαγράμματος ροής 4.4, η διαδικασία εκκινεί με την ταυτόχρονη φόρτωση της RGB εικόνας και των αντίστοιχων δεδομένων βάθους από LIDAR, επιτρέποντας την συσχέτιση πληροφοριών 2D και 3D. Εν συνεχεία, εφαρμόζεται το μοντέλο YOLOv11 για την αναγνώριση αντικειμένων εντός του καρέ, με το μοντέλο να επιστρέφει θέσεις και κατηγορίες αντικειμένων υπό την μορφή ορθογωνίων πλαισίων. Παράλληλα, τα αποτελέσματα προβάλλονται με την μορφή διακριτού επιπέδου γραφικής απεικόνισης (bounding boxes layer), το οποίο δύναται να ενοποιηθεί με άλλα επίπεδα πληροφορίας όπως δεδομένα πλοήγησης, χάρτες βάθους ή επαυξημένο περιεχόμενο. Η διαδικασία αυτή αποτελεί θεμέλιο για την οπτική κα-



Σχήμα 4.3: Διάγραμμα Επεξεργασίας ανά Καρέ Εικόνας.

τανόηση σκηνών σε εφαρμογές όπως αυτόνομη πλοήγηση, έξυπνη κινητικότητα και επαυξημένη πραγματικότητα.

Το διάγραμμα ροής 4.5 αποτελεί κρίσιμη διεργασία για την αναπαράσταση του περιβάλλοντος με γεωμετρική ακρίβεια. Η παρούσα υλοποίηση βασίζεται στην επεξεργασία δεδομένων LIDAR, τα οποία εισάγονται ως σημειακό νέφος και αρχικά ευθυγραμμίζονται με σταθερό σύστημα αναφοράς μέσω αντιστάθμισης εγωκίνησης (ego-motion compensation). Στην συνέχεια, τα δεδομένα μετασχηματίζονται στο σύστημα της κάμερας, χρησιμοποιώντας πλήρη βαθμονόμηση ώστε να εξασφαλιστεί συμβατότητα με εικόνες RGB ή άλλα οπτικά δεδομένα. Παράλληλα, τα τρισδιάστατα σημεία προβάλλονται στο επίπεδο εικόνας και παράγεται χάρτης βάθους, ο οποίος απεικονίζει με ακρίβεια την απόσταση κάθε pixel από τον αισθητήρα. Ο



Σχήμα 4.4: Διάγραμμα Ανίχνευσης Αντικειμένων μέσω Μηχανικής Μάθησης.

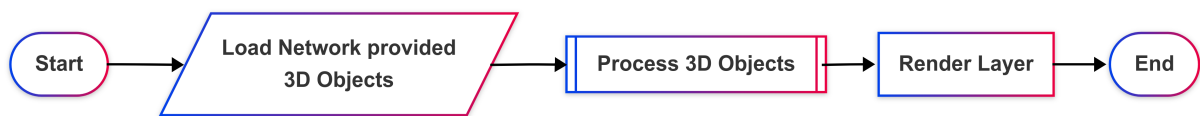


Σχήμα 4.5: Διάγραμμα Υπολογισμού Βάθους μέσω LIDAR.

τελικός χάρτης αποδίδεται ως οπτικό layer, χρήσιμο για ενίσχυση αντιληπτικών συστημάτων, ενοποίηση με AR, ή υπολογιστική ανάλυση σκηνών. Η διαδικασία αυτή ενισχύει την κατανόηση του περιβάλλοντος και την οπτική ενοποίηση μεταξύ δεδομένων 2D και 3D.

Παράλληλα, το διάγραμμα ροής 4.6 απεικονίζει την λήψη, επεξεργασία και την απόδοση τρισδιάστατων αντικειμένων τα οποία προέρχονται δυναμικά από δικτυακές πηγές. Αρχικά, το σύστημα πραγματοποιεί φόρτωση των 3D αντικειμένων από κάποιον απομακρυσμένο κόμβο (server ή edge device) αξιοποιώντας REST APIs ή άλλα πρωτόκολλα ανταλλαγής δεδομένων όπως MQTT ή Web Sockets. Τα δεδομένα αυτά περιλαμβάνουν γεωμετρική πληροφορία και συνοδευτικά μεταδεδομένα. Στην συνέχεια, εφαρμόζεται προκαθορισμένη λογική επεξεργασίας ώστε τα αντικείμενα να ενσωματωθούν σωστά στο τοπικό περιβάλλον απόδοσης: μετασχηματίζονται στον χώρο της κάμερας ή της AR σκηνής, κλιμακώνονται, περιστρέφονται και εναρμονίζονται με τα υπόλοιπα επίπεδα πληροφορίας. Τέλος, αποδίδονται ως ανεξάρτητο επίπεδο (rendered layer), έτοιμο να συνδυαστεί με δεδομένα αισθητήρων, χάρτες βάθους ή πληροφορίες εντοπισμού αντικειμένων, ενισχύοντας έτσι την τελική απεικόνιση σε πλαίσια επαυξημένης πραγματικότητας ή αυτόνομης πλοήγησης.

Αναλυτικότερα, το διάγραμμα ροής 4.7 περιγράφει τη λογική επεξεργασίας που αναφέρθηκε προηγουμένως. Σύμφωνα με το εν λόγω διάγραμμα, η επεξεργασία των τρισδιάστατων αντικειμένων περιλαμβάνει τη χαρτογράφηση και την απεικόνισή τους, με σκοπό είτε την κατανόηση της σκηνής είτε την υποστήριξη εφαρμογών επαυξημένης πραγματικότητας. Σε κάθε κύκλο εκτέλεσης, το σύστημα εξάγει τις γεωμετρικές διαστάσεις του αντικειμένου, καθώς και την απόστασή του από την κάμερα. Στη συνέχεια, εφαρμόζεται φίλτρο βάσει της μέγιστης απόστασης και του οπτικού πεδίου (view frustum), ώστε να απορριφθούν σημεία που βρίσκονται



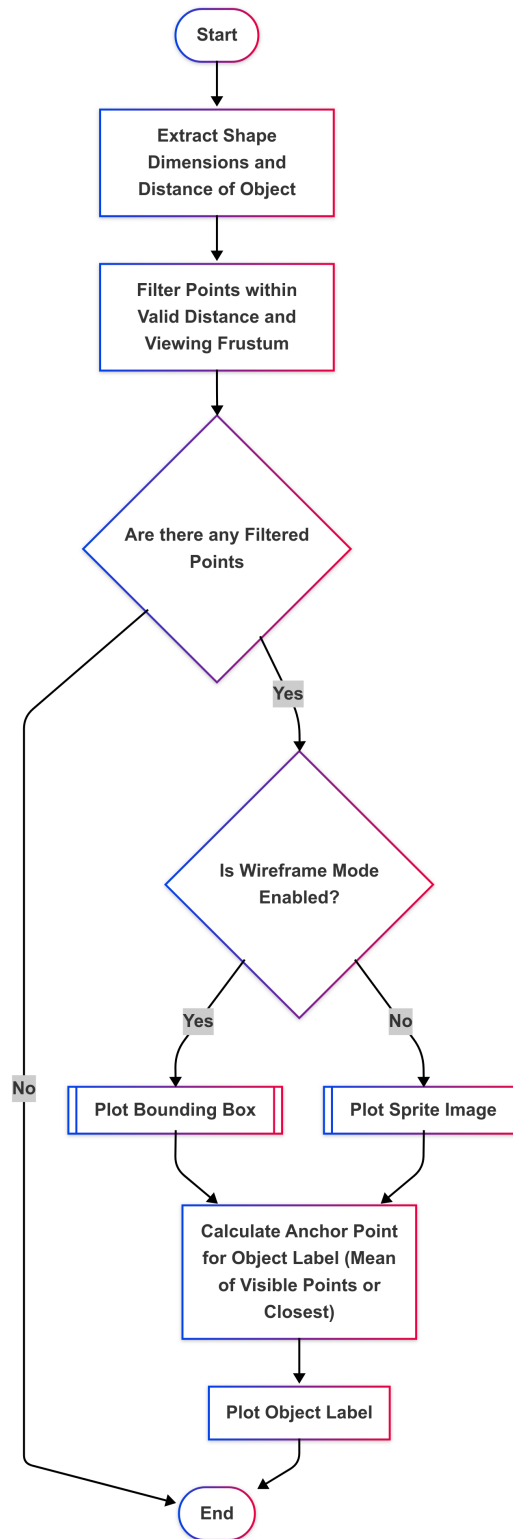
Σχήμα 4.6: Διάγραμμα Υπολογισμού Εισερχόμενων Αντικειμένων.

εκτός της περιοχής ενδιαφέροντος. Εφόσον εντοπιστούν έγκυρα σημεία, το σύστημα καθορίζει την οπτική απόδοση του αντικειμένου. Αν έχει επιλεγεί λειτουργία wire-frame, σχεδιάζεται ένα 3D πλαίσιο οριοθέτησης (bounding box) γύρω από το αντικείμενο· εναλλακτικά, αποδίδεται ένα εικονίδιο (sprite). Τέλος, υπολογίζεται ένα σημείο αγκύρωσης (anchor point), είτε με βάση τον μέσο όρο των σημείων είτε με βάση το πλησιέστερο σημείο, και αποδίδεται η αντίστοιχη ετικέτα του αντικειμένου, ενισχύοντας έτσι την αντίληψη του περιβάλλοντος.

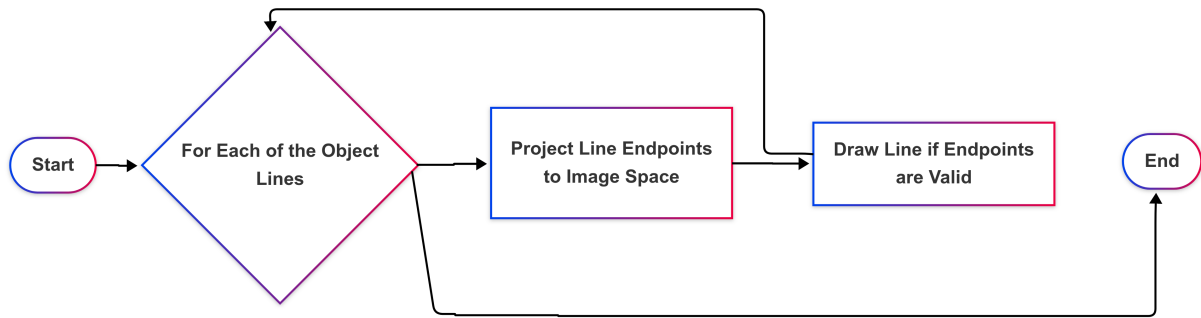
Η διεργασία εικονικοποίησης 3D πλαισίου οριοθέτησης παρουσιάζεται στο διάγραμμα ροής 4.8. Όπως φαίνεται στο εν λόγω διάγραμμα, η αποτύπωση των περιγραμμάτων των τρισδιάστατων αντικειμένων στην εικόνα πραγματοποιείται μέσω της προβολής των ακμών τους στο επίπεδο της κάμερας. Συγκεκριμένα, για κάθε ακμή (γραμμή) του αντικειμένου, τα δύο άκρα της μετασχηματίζονται από τις τρισδιάστατες συντεταγμένες στον χώρο της κάμερας και προβάλλονται στο δισδιάστατο επίπεδο της εικόνας. Εφόσον και τα δύο σημεία βρίσκονται εντός του οπτικού πεδίου (view frustum) και θεωρούνται έγκυρα, η αντίστοιχη ακμή σχεδιάζεται, δημιουργώντας σταδιακά το περίγραμμα (bounding box) του αντικειμένου. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να αποδοθούν όλες οι ακμές, προσφέροντας μια γραφική αναπαράσταση που ενισχύει την κατανόηση της γεωμετρίας του τρισδιάστατου περιεχομένου μέσα στην εικόνα.

Στο πλαίσιο της απόδοσης αντικειμένων σε επαυξημένα περιβάλλοντα οδήγησης, αξιοποιείται η τεχνική αποτύπωσης με χρήση εικονιδίων (sprites) προερχομένων από φύλλων ψηφιογραφικών αντικειμένων, όπως περιγράφεται στο διάγραμμα ροής 4.9. Αρχικά, υπολογίζεται ο προσανατολισμός του αντικειμένου με βάση τη μέση κατεύθυνση των διαθέσιμων διανυσμάτων θέσης. Στη συνέχεια, αναλύεται η τρισδιάστατη δομή μέσω των πλεγμάτων (meshes), σχεδιάζοντας τα περιγράμματα κάθε επιμέρους τριγώνου. Αφού καθοριστεί η περιοχή ενδιαφέροντος και το αντίστοιχο bounding box, γίνεται η τελική αποτύπωση του αντικειμένου με χρήση εικονιδίου με την διεργασία σχεδίασης που αναλύεται στην συνέχεια. Αν ζητηθεί, εμφανίζεται και το πλαίσιο (bounding box) για σκοπούς ενίσχυσης της αντίληψης. Η μέθοδος

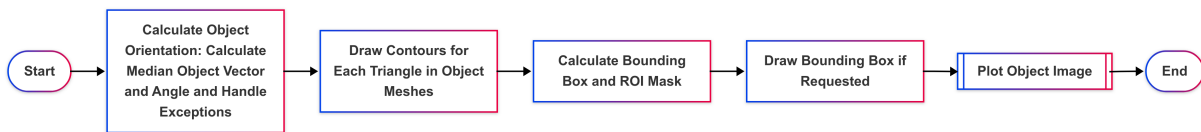
αυτή προσφέρει μια πιο φιλική και καθαρή απεικόνιση, ιδιαίτερα σε περιβάλλοντα με πολλαπλά αντικείμενα, διευκολύνοντας την κατανόηση από την πλευρά του εκάστοτε χρήστη.



Σχήμα 4.7: Διάγραμμα Εικονικοποίησης 3D Αντικειμένου.

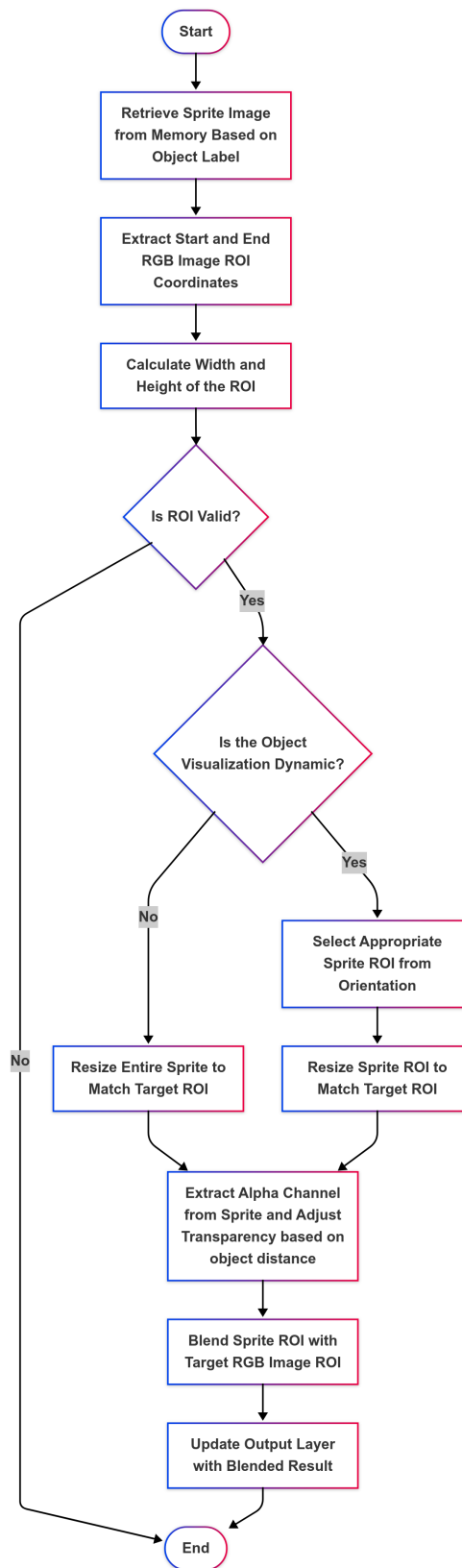


Σχήμα 4.8: Διάγραμμα Εικονικοποίησης 3D Πλαισίου Οριοθέτησης.



Σχήμα 4.9: Διάγραμμα Εικονικοποίησης 2D Ψηφιογραφικού Αντικειμένου.

Τέλος, το διάγραμμα ροής 4.10, αποτελεί την διαδικασία τελικής απεικόνισης του δυσδιάστατου ψηφιογραφικού αντικειμένου, η οποία περιλαμβάνει την ανάκτηση της εικόνας από την μνήμη, τον υπολογισμό της περιοχής ενδιαφέροντος στο RGB καρέ και τον έλεγχο εγκυρότητας της προβολής. Ανάλογα με το αν η απεικόνιση είναι δυναμική, το σύστημα είτε επιλέγει συγκεκριμένο τμήμα του sprite ανάλογα με τον προσανατολισμό, είτε προσαρμόζει ολόκληρη την εικόνα. Στην συνέχεια, εφαρμόζεται διαφάνεια (άλφα κανάλι) που σχετίζεται με την απόσταση του αντικειμένου και πραγματοποιείται συγχώνευση της περιοχής sprite με την RGB εικόνα. Το αποτέλεσμα αποθηκεύεται στο τελικό στρώμα απεικόνισης, εξασφαλίζοντας ρεαλιστική ένθεση πληροφορίας στην οθόνη.



Σχήμα 4.10: Διάγραμμα Σχεδίασης 2D Ψηφιογραφικού Αντικειμένου.



Σχήμα 4.11: Προσωποποιημένη Προβολή Επαυξημένης Πραγματικότητας.

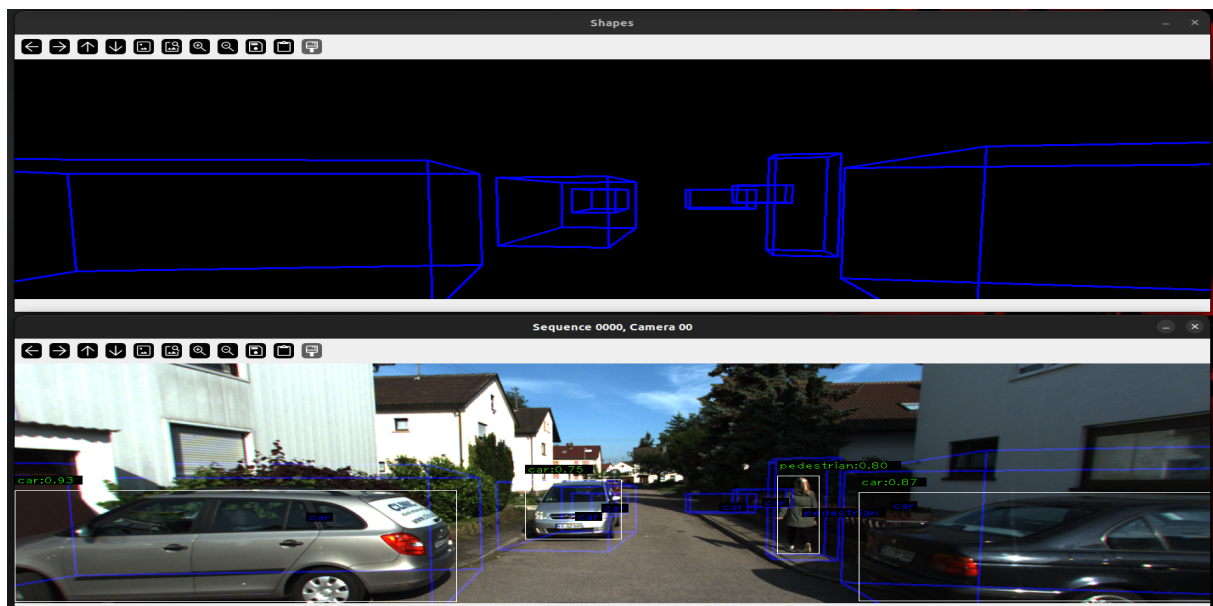
4.5 Παραδείγματα Λειτουργείας

Ολοκληρώνοντας το τρέχον κεφάλαιο, ακολουθούνε παραδείγματα λειτουργείας της προαναφερθείσας υλοποίησης του συστήματος Ε.Π.Ο.Σ με αντίστοιχες καταγραφές οθόνης οι οποίες συλλέχθηκαν, κατά την εκπόνηση της συγκεκριμένης Διπλωματικής Εργασίας, προς επίρρωσης της σημαντικότητας της εφαρμογής μας, αναφορικά με την πρόβλεψη και την αναγνώριση αντικειμένων στις οδικές διαδρομές των σύγχρονων Έξυπνων Πόλεων, με απώτερο σκοπό την πρόληψη ατυχημάτων, της ετοιμότητας των οδηγών, καθώς και της ασφαλέστερης και βελτιωμένης λειτουργίας των συνδεδεμένων αυτόνομων οχημάτων.

Στο σχήμα 4.11 παρουσιάζεται ένα ενδεικτικό παράδειγμα οπτικοποίησης της Επαυξημένης Πραγματικότητας (AR) στο ταμπλό του οδηγού-χρήστη. Η διεπαφή είναι σε υψηλό βαθμό παραμετροποιήσιμη, προσφέροντας στον οδηγό τη βέλτιστη δυνατή οδηγική εμπειρία και προσαρμογή στις ατομικές του προτιμήσεις. Χαρακτηριστικά, διακρίνονται τα λευκά δισδιάστατα πλαίσια οριοθέτησης, τα οποία έχουν παραχθεί μέσω του μοντέλου μηχανικής μάθησης YOLOv11, ενώ επάνω από κάθε πλαίσιο εμφανίζεται η κατηγορία του ανιχνευθέντος αντικειμένου μαζί με το ποσοστό εμπιστοσύνης (confidence score). Το ποσοστό αυτό καθορίζει και το χρώμα της γραμματοσειράς, το οποίο μεταβάλλεται από πράσινο (100% εμπιστοσύνη) έως κόκκινο (χαμηλή εμπιστοσύνη). Επιπλέον, απεικονίζονται αντικείμενα που προέρχονται από τους Έξυπνους Σηματοδότες, με μπλε ετικέτες και αντίστοιχα εικονίδια για κάθε κατηγορία, των οποίων το μέγεθος και η κατεύθυνση υπολογίζεται βάσει τις πληροφορίες που λήφθηκαν από τον σηματοδότη. Στην παρούσα περίπτωση, τα αντικείμενα εμφανίζονται με ημιδιαφανή υφή, κατόπιν παραμετροποίησης του

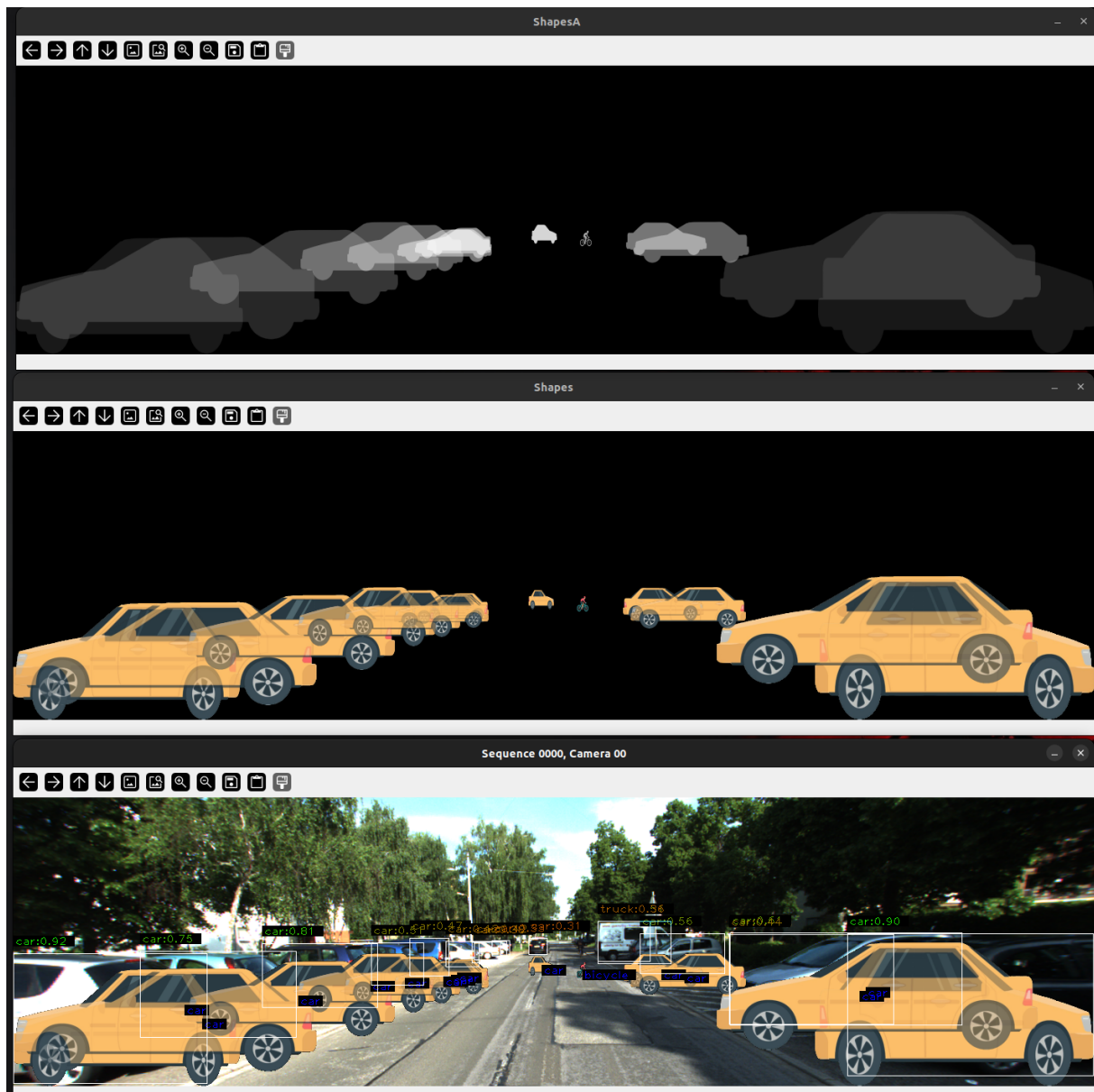


(α)



(β)

Σχήμα 4.12: Σύγκριση Εικονικοποίησης μέσω 2D Ψηφιογραφικού Αντικειμένου και 3D Πλαισίου Οριοθέτησης I.



Σχήμα 4.13: Σύνθεση Εικόνων μέσω Απόστασης / Καναλιού Διαφάνειας.

οδηγού–χρήστη, ώστε να αποφεύγεται η υπερφόρτωση του οπτικού πεδίου και να διατηρείται η φυσική αντίληψη του περιβάλλοντος.

Ο χρήστης έχει τη δυνατότητα επιλογής μεταξύ δύο διαφορετικών εμπειριών οπτικοποίησης: της εμπειρίας μέσω τρισδιάστατων πλαισίων οριοθέτησης (3D bounding boxes) ή της εμπειρίας μέσω δισδιάστατων ψηφιογραφικών αντικειμένων (2D sprite visualization), όπως απεικονίζεται στο Σχήμα 4.12. Επιπλέον, παρουσιάζονται τα διαδοχικά στάδια εικονικοποίησης για κάθε περίπτωση. Στο ανώτερο επίπεδο απεικονίζεται η διαβάθμιση διαφάνειας βάσει της απόστασης από το όχημα, όπου τα απομακρυσμένα αντικείμενα αποδίδονται με αυξημένη αδιαφάνεια, ενώ



(α)



(β)

Σχήμα 4.14: Προσαρμογή Έντασης Προβολής Επαυξημένης Πραγματικότητας.

τα εγγύτερα εμφανίζονται πιο διαφανή, προκειμένου να ενισχυθεί η αντίληψη των αντικειμένων εκτος περιοχής ορατής παρατήρησης του οδηγού. Στη συνέχεια, οι αντίστοιχες οπτικοποιήσεις τοποθετούνται στο ίδιο εικονικό επίπεδο, λαμβάνοντας υπόψη τη χωρική ιεραρχία που προκύπτει από τη διαβάθμιση. Τέλος, παράγονται οι τελικές εμπειρίες Επαυξημένης Πραγματικότητας, οι οποίες συνδυάζουν τα παραπάνω επίπεδα απεικόνισης, προσφέροντας στον οδηγό μια προσαρμόσιμη και λειτουργικά βελτιστοποιημένη οπτική εμπειρία.

Το σχήμα 4.13 επαναπαρουσιάζει τα διαδοχικά βήματα εικονικοποίησης (visualization pipeline) που ακολουθούνται στην εμπειρία μέσω δισδιάστατων ψηφιογραφικών αντικειμένων, από την αρχική ανίχνευση των αντικειμένων έως τη σύνθεση της τελικής εμπειρίας Επαυξημένης Πραγματικότητας. Η αναπαράσταση αυτή αποτυπώνει με σαφήνεια τον τρόπο με τον οποίο τα δεδομένα συλλέγονται, επεξεργάζονται και τελικά απεικονίζονται στο περιβάλλον διεπαφής, αναδεικνύοντας την επιρροή της στην ενίσχυση της οδικής ασφάλειας, καθώς επιτρέπει την οπτικοποίηση αντικειμένων πέρα από το άμεσο οπτικό πεδίο του οδηγού, συμβάλλοντας έτσι στην εξάλειψη των λεγόμενων “νεκρών γωνιών” και στη βελτίωση της αντιληπτικής επίγνωσης (situational awareness) του οδηγού-χρήστη. Με αυτόν τον τρόπο,



(α)



(β)

Σχήμα 4.15: Εικονικοποίηση Απόστασης / Βάθους μέσω LIDAR.

η Επαυξημένη Πραγματικότητα μετατρέπεται σε ενεργό υποστηρικτικό μηχανισμό πρόληψης κινδύνων, ενισχύοντας την έγκαιρη αναγνώριση εμποδίων και τη λήψη ασφαλέστερων αποφάσεων κατά την οδήγηση.

Στον τομέα της παραμετροποίησης της εμπειρίας του οδηγού, παρέχεται η δυνατότητα ρύθμισης της έντασης και του βαθμού επιρροής της Επαυξημένης Πραγματικότητας επί της σκηνής (σχήμα 4.14), μέσω τροποποίησης της παγκόσμιας (global) τιμής διαφάνειας των υπερτιθέμενων στοιχείων. Με τον τρόπο αυτό, ο κάθε χρήστης δύναται να επιλέξει μεταξύ πλήρους, μερικής ή μηδενικής ενσωμάτωσης Ε.Π., προσαρμόζοντας δυναμικά το επίπεδο εικονικής πληροφορίας ανάλογα με τις προσωπικές του προτιμήσεις ή τις εκάστοτε οδηγικές συνθήκες. Παράλληλα, προσφέρονται πρόσθετα επίπεδα (layers) οπτικοποίησης που ενσωματώνουν δεδομένα βάθους ή απόστασης προερχόμενα από αισθητήρες LiDAR, καθώς και συμπληρωματικές πληροφορίες όπως γεωτοποθεσία (GPS), χρονική σήμανση (timestamp) ή μεταδεδομένα καρέ εικόνας, όπως παρουσιάζονται στα σχήματα 4.15 και 4.16 αντίστοιχα.

Το σχήμα 4.17 απεικονίζει τη λειτουργία της ενεργής ανίχνευσης του περιβάλλοντος οχήματος, η οποία βασίζεται στη λήψη δεδομένων από πολυκάμερο σύστημα τριών χρωματικών καναλιών (RGB) ως είσοδο και στην επεξεργασία τους μέσω



(α)



(β)



(γ)

Σχήμα 4.16: Προβολή Επιπρόσθετων Πληροφοριών.

του μοντέλου μηχανικής μάθησης YOLOv11 για την πρόβλεψη και κατηγοριοποίηση αντικειμένων ενδιαφέροντος. Το μοντέλο έχει εκπαιδευτεί στο σύνολο δεδομένων COCO, το οποίο περιλαμβάνει εκτενή και ποικίλη λίστα κατηγοριών αντικειμένων, καλύπτοντας ένα ευρύ φάσμα πιθανών στοιχείων που δύναται να εμφανιστούν στην οδική διαδρομή ενός οχήματος (όπως πεζοί, ποδήλατα, οχήματα, σηματοδότες



Σχήμα 4.17: Αναγνώριση Διαφορετικών Αντικειμένων Ενδιαφέροντος μέσω Υπολογιστικής Όρασης / Μηχανικής Μάθησης.



(α)



(β)

Σχήμα 4.18: Ανίχνευση και Εικονικοποίηση Ποδηλάτη.

κ.ά.). Στο σχήμα 4.18 παρουσιάζεται ενδεικτικά η ανίχνευση ποδηλάτη σε πραγματικό χρόνο, αποδεικνύοντας τη δυνατότητα του συστήματος να αναγνωρίζει κρίσιμα αντικείμενα ακόμη και σε σύνθετες κυκλοφοριακές συνθήκες.



Σχήμα 4.19: Ανίχνευση Οχήματος σε Περιοχή Εκτός Οπτικού Πεδίου.

Στη συνέχεια παρουσιάζονται πραγματικά παραδείγματα οδικών σεναρίων, στα οποία η ενεργή ανίχνευση του οχήματος αποδεικνύεται ανεπαρκής λόγω περιορισμών του οπτικού πεδίου και φυσικών εμποδίων στο περιβάλλον. Το φαινόμενο αυτό οφείλεται στο γεγονός ότι το όχημα, ως παρατηρητής εξοπλισμένος με οπτικούς αισθητήρες, υπόκειται σε παρόμοια κενά οπτικής αντίληψης (blind spots) με εκείνα ενός ανθρώπινου οδηγού. Η αντιμετώπιση του προβλήματος επιτυγχάνεται μέσω της διασύνδεσης με έξυπνους σηματοδότες (Smart Traffic Lights), οι οποίοι λειτουργούν ως εξωτερικοί κόμβοι ανίχνευσης και μεταδίδουν σε πραγματικό χρόνο τα δεδομένα των ανιχνευμένων αντικειμένων ενδιαφέροντος προς το όχημα-παρατηρητή.

Στο σχήμα 4.19 παρουσιάζεται χαρακτηριστικά η ανίχνευση οχημάτων εκ δεξιών του δρόμου, τα οποία είναι πλήρως καλυμμένα από εμπόδια όπως θάμνοι, φράχτες ή κτήρια. Χάρη στη μετάδοση πληροφορίας από τους έξυπνους σηματοδότες, ο οδηγός (ή το αυτόνομο σύστημα) ενημερώνεται προληπτικά, πριν καν προσεγγίσει το κρίσιμο σημείο δυνητικής σύγκρουσης. Αντίστοιχα, στο σχήμα 4.20 απεικονίζονται τρία διαδοχικά καρέ στα οποία έχει πραγματοποιηθεί ανίχνευση πεζού, επιτρέποντας την άμεση ηλεκτρονική απόκριση του αυτόνομου οχήματος μέσω ενεργοποίησης επείγοντος φρεναρίσματος (emergency braking).

Το σχήμα 4.21 αποτυπώνει τη διαχείριση στιγμιαίων κενών αντίληψης — περιπτώσεων όπου ένα αντικείμενο, όπως ένας πεζός, παροδικά αποκρύπτεται από εμπόδια του περιβάλλοντος. Ειδικότερα, στο καρέ 4.21α ο πεζός καλύπτεται από δέντρο, στο 4.21β παρουσιάζεται η εικονικοποιημένη εκτίμηση της θέσης του, ενώ στο 4.21γ το σύστημα επαναβεβαιώνει την πλήρη ανίχνευσή του. Τέλος, το σχήμα 4.22 αναπαριστά ένα από τα πιο κρίσιμα σενάρια στην οδική ασφάλεια: έναν πεζό



(α)



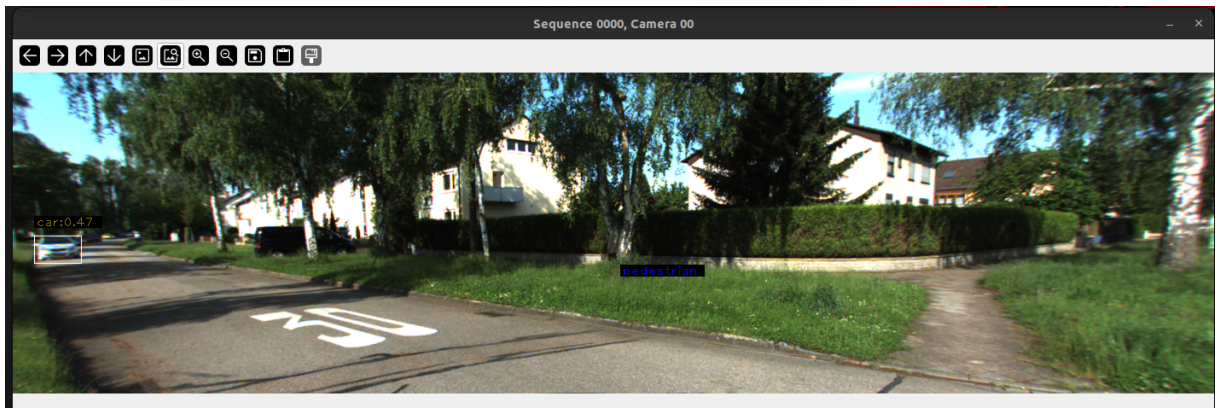
(β)



(γ)

Σχήμα 4.20: Ανίχνευση Πεζού σε Περιοχή Εκτός Οπτικού Πεδίου.

που διασχίζει το οδόστρωμα όντας πλήρως κρυμμένος από διερχόμενο όχημα. Χάρη στη συνεργατική ανίχνευση και προβολή μέσω Επαυξημένης Πραγματικότητας, το σύστημα επιτυγχάνει έγκαιρη προειδοποίηση του οδηγού ή προληπτική επέμβαση του αυτόνομου ελέγχου, συμβάλλοντας καθοριστικά στην προστασία ευάλωτων χρηστών του οδικού δικτύου, όπως παιδιά και ηλικιωμένοι.



(α)



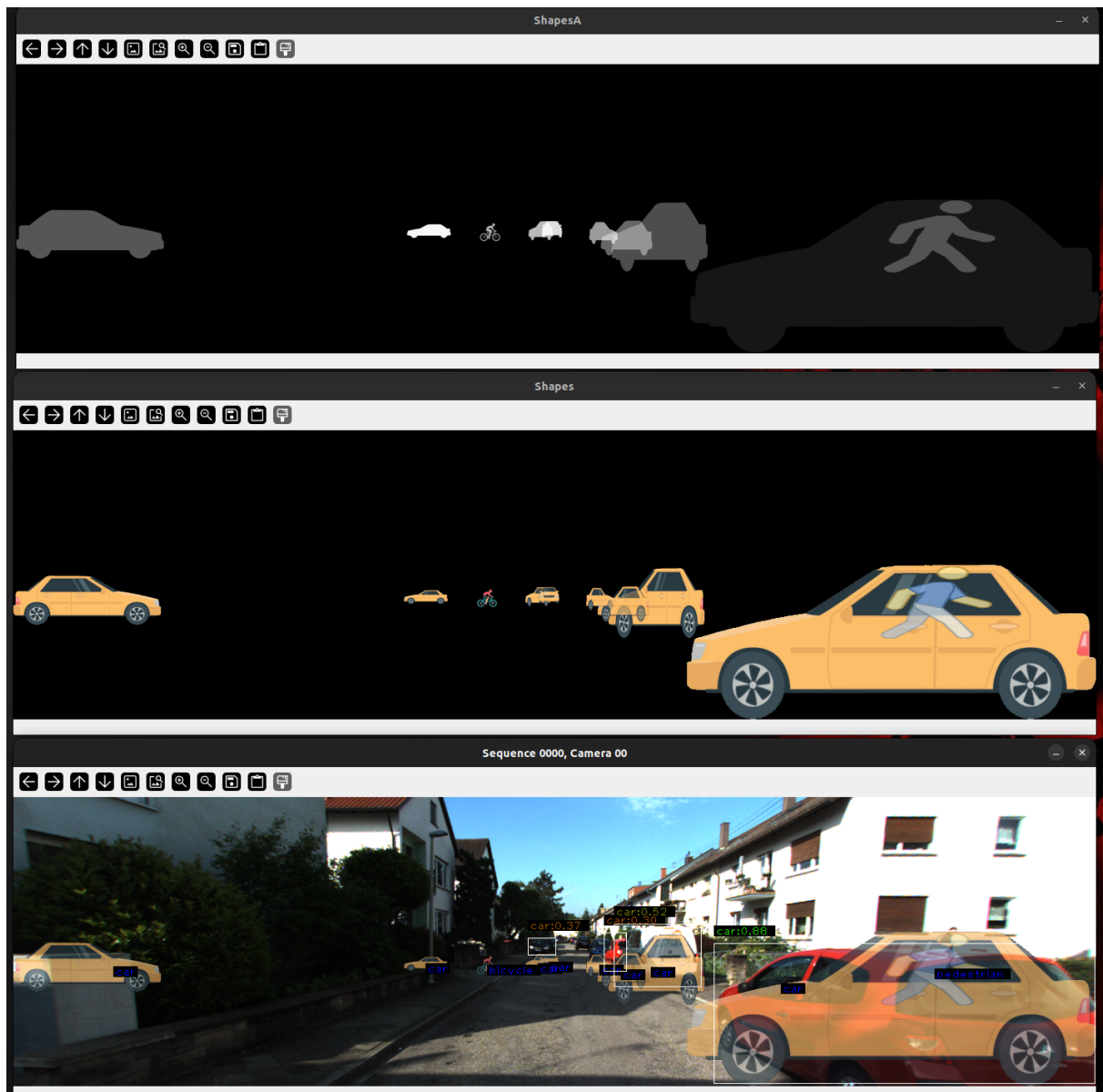
(β)



(γ)

Σχήμα 4.21: Ανίχνευση Πεζού σε Περιοχή με Οπτικό Κενό Αντίληψης.

Η συνολική παρουσίαση των παραδειγμάτων χρήσης του συστήματος Επαυξημένης Πραγματικότητας αναδεικνύει μια ολιστική προσέγγιση ενίσχυσης της οδικής ασφάλειας, βασισμένη στη σύζευξη οπτικοποίησης, ανίχνευσης και συνεργατικής επικοινωνίας. Μέσω προηγμένων αλγορίθμων μηχανικής μάθησης και αισθητήρων υψηλής ακρίβειας, το σύστημα επιτυγχάνει πραγματικού χρόνου αναγνώριση και



Σχήμα 4.22: Εικονικοποίηση Πεζού σε Περιοχή Εκτός Οπτικού Πεδίου.

προβολή αντικειμένων ενδιαφέροντος, βελτιώνοντας την αντιληπτική επίγνωση του οδηγού και περιορίζοντας τα πεδία μειωμένης ορατότητας.

Η δυνατότητα προσαρμογής της εμπειρίας Επαυξημένης Πραγματικότητας ως προς τη διαφάνεια, το βάθος και την ποσότητα προβαλλόμενης πληροφορίας (σχήμα 4.23) προσφέρει έναν ισορροπημένο συνδυασμό ρεαλισμού και εργονομίας, αποτρέποντας την οπτική υπερφόρτωση και ενισχύοντας τη συγκέντρωση του χρήστη. Παράλληλα, η ενσωμάτωση δεδομένων από εξωτερικές υποδομές, όπως έξυπνοι σηματοδότες και cloud συστήματα, επεκτείνει την αντίληψη του οχήματος πέρα από τα φυσικά όρια του οπτικού πεδίου, εξαλείφοντας αποτελεσματικά τις “νεκρές γω-

νίες” και υποστηρίζοντας προληπτική λήψη αποφάσεων.

Συνοψίζοντας, η προτεινόμενη υλοποίηση προσεγγίζει την Επαυξημένη Πραγματικότητα όχι ως απλό μέσο απεικόνισης, αλλά ως ενεργό μηχανισμό γνωστικής υποστήριξης και ασφάλειας, ικανό να συνδυάζει την ανθρώπινη αντίληψη με τα δεδομένα της τεχνητής νοημοσύνης και των συνεργατικών συστημάτων μεταφορών. Μέσω αυτής της σύζευξης, επιτυγχάνεται ένα πιο ασφαλές, ευφύες και συνδεδεμένο οδηγικό περιβάλλον, προσαρμοσμένο στις ανάγκες των σύγχρονων συνδεδεμένων αυτόνομων οχημάτων.

ΚΕΦΑΛΑΙΟ 5

ΑΞΙΟΛΟΓΗΣΗ ΣΥΣΤΗΜΑΤΟΣ

5.1 Πειραματική Διάταξη

5.2 Αποτελέσματα

5.3 Αξιολόγηση

5.4 Βελτιστοποιήσεις και Περιορισμοί

5.1 Πειραματική Διάταξη

Για την πειραματική αξιολόγηση του συστήματος χρησιμοποιήθηκε ένας προσωπικός υπολογιστής με τον επεξεργαστή Ryzen 9 3950X της AMD, με 16 επεξεργαστικούς πυρήνες και 2 νήματα ανά πυρήνα, καθώς και 32 GiB μνήμης τυχαίας προσπέλασης τεχνολογίας DDR4 με ταχύτητα ρολογιού 2667 MHz. Στον υπολογιστή χρησιμοποιήθηκε κάρτα γραφικών με τον μικροεπεξεργαστή GeForce RTX 2080 SUPER της εταιρείας Nvidia, με εσωτερική μνήμη μεγέθους 8GiB, τεχνολογίας GDDR6 και έκδοση CUDA 12.2. Αποθηκευτικό μέσο τύπου NVMe χρησιμοποιήθηκε για την προσπέλαση του σετ δεδομένων εικόνας και νέφους σημείων με ταχύτητα ρολογιού 33MHz και μεγέθους εντολών στα 64 bits.

Στη συνέχεια, η διαδικασία δοκιμών επεκτάθηκε και στις υπόλοιπες σειρές του συνόλου δεδομένων που αντιστοιχούν στην κάμερα 0, προκειμένου να επιτευχθεί μια πιο ολοκληρωμένη αξιολόγηση των επιδόσεων του συστήματος. Για κάθε σειρά δεδομένων πραγματοποιήθηκαν μετρήσεις του ρυθμού ανανέωσης της εικόνας ανά

δευτερόλεπτο (frames per second) και υπολογίστηκε η μέση τιμή του ρυθμού ανανέωσης για κάθε σειρά σκηνών. Ακολούθως, ο μέσος όρος ανά χαρακτηριστική κατηγορία οπτικοποίησης προέκυψε από τον μέσο όρο των επιμέρους μέσων όρων των αντίστοιχων σκηνών.

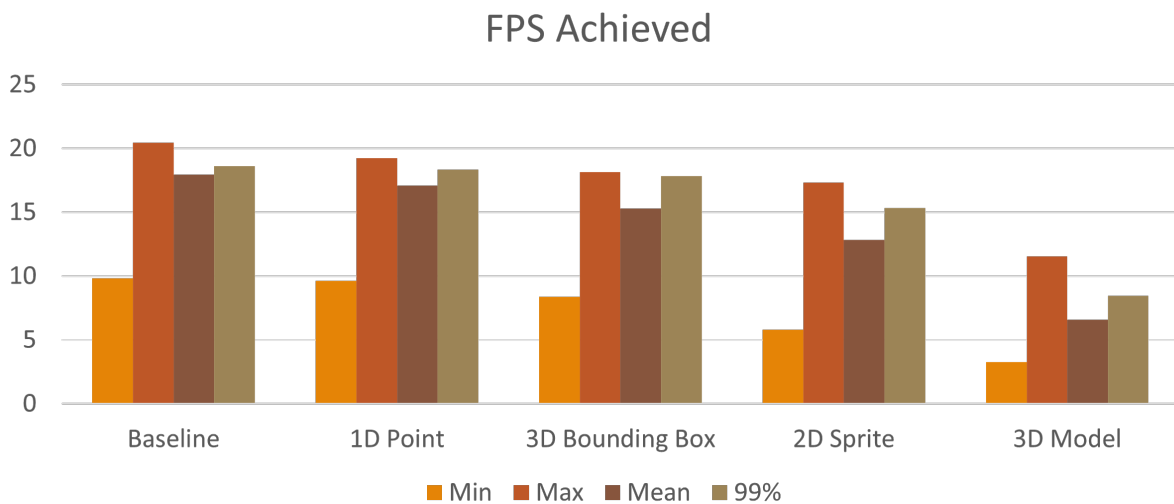
Οι κατηγορίες οπτικοποίησης που αξιολογήθηκαν είναι: οπτικοποίηση ως δίσκος (ευανάγνωστο σημείο), ως τρισδιάστατο πλαίσιο οριοθέτησης, ως δισδιάστατο ψηφιογραφικό αντικείμενο και ως τρισδιάστατο μοντέλο. Οι βασικές στατιστικές παράμετροι που εξήχθησαν περιλαμβάνουν τις τιμές του Ελαχίστου, του Μέγιστου, του Μέσου Όρου και του 99ου Εκατοστημορίου, προσφέροντας μια σαφή και αξιόπιστη απεικόνιση της υπολογιστικής απόδοσης κατά την επεξεργασία των οπτικών δεδομένων.

5.2 Αποτελέσματα

Η αξιολόγηση των δεδομένων του πίνακα 5.1, όπου οι μετρήσεις αντιστοιχούν σε ρυθμό καρέ ανά δευτερόλεπτο (FPS) οπτικοποιείται στο γράφημα 5.1 και αναδεικνύει την επίδραση της μορφής οπτικοποίησης στην απόδοση του συστήματος Επαυξημένης Πραγματικότητας. Στην Αρχική Κατάσταση, οι τιμές κυμαίνονται μεταξύ 9.82 και 20.44 FPS, με μέσο όρο 17.96, αντιπροσωπεύοντας τη βασική επίδοση αναφοράς. Με την υιοθέτηση της μεθόδου δίσκου, παρατηρείται ελαφρά υποβάθμιση στη σταθερότητα και στη μέση απόδοση (17.08 FPS), γεγονός που υποδηλώνει μια αποδοτικότερη απεικόνιση με μειωμένο γραφικό φόρτο εις βάρος της ικανότητας αντίληψης του χρήστη. Η απεικόνιση μέσω τρισδιάστατου πλαισίου οριοθέτησης (3D bounding box) επιβαρύνει περαιτέρω τον ρυθμό απεικόνισης (Μ.Ο. 15.29 FPS), ενώ διατηρεί μια ικανοποιητική ισορροπία μεταξύ λεπτομέρειας και ταχύτητας. Η μέθοδος 2D ψηφιογραφικού αντικειμένου (sprite rendering) εμφανίζει ακόμη χαμηλότερες τιμές (Μ.Ο. 12.84 FPS, 99ο εκατοστημόριο 15.35), που ενώ κυμαίνεται σε αποδεκτό ρυθμό απεικόνισης, αποδεικνύει ότι οι προηγούμενες απλούστερες αναπαραστάσεις μειώνουν δραστικά το υπολογιστικό κόστος και επιτρέπουν πιο ομαλή λειτουργία. Τέλος, η πλήρης τρισδιάστατη μοντελοποίηση (3D Model), αν και προσφέρει ρεαλισμό, εμφανίζει τις χαμηλότερες επιδόσεις (Μ.Ο. 6.58 FPS), λόγω της αυξημένης πολυπλοκότητας και των απαιτήσεων σε υπολογιστικούς πόρους. Συνολικά, τα αποτελέσματα αποτυπώνουν μια αντίστροφη σχέση μεταξύ πολυπλο-

Πίνακας 5.1: Πίνακας Αποτελεσμάτων Καρέ ανα Δευτερόλεπτο.

	Ελάχιστο	Μέγιστο	Μέσος Όρος	99 Εκατοστημόριο
Αρχική Κατάσταση	9.82	20.44	17.96	18.62
Δίσκος	9.63	19.26	17.08	18.34
3D Πλαίσιο Οριοθέτησης	8.38	18.14	15.29	17.82
2D Ψηφιογραφικό Αντικείμενο	5.8	17.32	12.84	15.35
Μοντέλο 3D	3.27	11.56	6.58	8.47



Σχήμα 5.1: Γράφημα Αποτελεσμάτων.

κότητας της οπτικοποίησης και ρυθμού απόδοσης, επιβεβαιώνοντας ότι σε εφαρμογές οδικής ασφάλειας, όπου η άμεση και ρευστή απεικόνιση κινδύνων είναι κρίσιμη, οι ελαφρύτερες μορφές οπτικοποίησης προσφέρουν σημαντικό πλεονέκτημα έναντι των πιο ρεαλιστικών αλλά βαρύτερων μοντέλων.

5.3 Αξιολόγηση

Η αξιολόγηση του υλοποιημένου συστήματος απόδειξης ιδέας και της τρέχουσας ερευνητικής εργασίας καταδεικνύει την επιτυχία της προτεινόμενης αρχιτεκτονικής ως προς την ενίσχυση της οδικής ασφάλειας μέσω Επαυξημένης Πραγματικότητας

(ΕΠ) και συνεργατικής αντίληψης. Τα αποτελέσματα των πειραματικών δοκιμών επιβεβαιώνουν ότι το σύστημα είναι ικανό να ανιχνεύει, να επεξεργάζεται και να οπτικοποιεί κρίσιμες πληροφορίες κυκλοφορίας σε πραγματικό χρόνο, αξιοποιώντας δεδομένα από LiDAR, κάμερες, GNSS/OXTS και αλγορίθμους μηχανικής μάθησης (YOLOv11). Η AR διεπαφή αποδείχθηκε αποτελεσματική στην προβολή πληροφοριών χωρίς να επιβαρύνει το οπτικό πεδίο, ενώ η προσομοίωση εξωτερικών παρατηρητών από τους έξυπνους σηματοδότες ανέδειξε τη δυνατότητα αντίληψης πέρα από το οπτικό πεδίο (beyond line of sight), περιορίζοντας τα «τυφλά σημεία» και βελτιώνοντας την προγνωστική ικανότητα του οδηγού. Επιπλέον, η συγκριτική αξιολόγηση των μεθόδων εικονικοποίησης ανέδειξε τον αναγκαίο συμβιβασμό μεταξύ ταχύτητας εκτέλεσης και ανθρώπινης αντιληπτικής απόδοσης, ιδίως σε περιβάλλοντα με ελεγχτές περιορισμένων υπολογιστικών πόρων. Η ανάλυση κατέδειξε ότι οι μέθοδοι τρισδιάστατου πλαισίου οριοθέτησης (3D bounding box) και δισδιάστατης εικονογραφικής απεικόνισης (2D sprite rendering) αποτελούν τις πλέον αποδοτικές και ισορροπημένες προσεγγίσεις, προσφέροντας ικανοποιητική απόδοση τόσο ως προς τη ρεαλιστική απεικόνιση όσο και ως προς την υπολογιστική αποδοτικότητα, και συνεπώς προτείνονται ως οι βέλτιστες επιλογές στο πλαίσιο της παρούσας εργασίας.

Παρά τους περιορισμούς του πρωτοτύπου, η τρέχουσα έρευνα αποδεικνύει τη βιωσιμότητα και την τεχνική ωριμότητα της προσέγγισης, παρέχοντας ένα ισχυρό υπόβαθρο για μελλοντική εξέλιξη προς πλήρως διασυνδεδεμένα και αυτόνομα περιβάλλοντα έξυπνης κινητικότητας.

Η ανάλυση των αποτελεσμάτων επιτρέπει την επανασύνδεση της ερευνητικής μελέτης με τα αρχικά ερωτήματα, ξεκινώντας από το πρώτο, που αφορά την εξέλιξη της βιωματικής εμπειρίας του χρήστη στο πλαίσιο της Επαυξημένης Πραγματικότητας και τον ρόλο του χρηστοκεντρικού σχεδιασμού στη διαμόρφωσή της.

Τα ευρήματα της παρούσας εργασίας καταδεικνύουν ότι η εμπειρία του χρήστη δεν περιορίζεται πλέον στην παθητική λήψη πληροφοριών, αλλά εξελίσσεται σε ενεργή, πολυαισθητηριακή αλληλεπίδραση με το περιβάλλον οδήγησης. Αναλύοντας το ερευνητικό αυτό ερώτημα, καθίσταται σαφές ότι η Επαυξημένη Πραγματικότητα (ΕΠ) και η Μηχανική Μάθηση (ML) αποτελούν αλληλένδετες έννοιες που διαμορφώνουν καθοριστικά την ποιότητα της αλληλεπίδρασης του χρήστη με το ειχονικό περιεχόμενο και, κατ' επέκταση, την αποτελεσματικότητα του συστήματος. Η ΕΠ, ως χωρική εμπειρία, απαιτεί ομαλή ενσωμάτωση των εικονικών στοιχείων

στο φυσικό περιβάλλον, ώστε να επιτυγχάνεται ρεαλιστική και απρόσκοπτη οπτική σύνθεση, γεγονός που καθιστά τον σχεδιασμό της ένα πολύπλοκο και απαιτητικό επιστημονικό πεδίο.

Η χρηστοκεντρική προσέγγιση (User-Centered Design) αναδεικνύεται κρίσιμη, καθώς τοποθετεί τον χρήστη στο επίκεντρο της σχεδίασης, με στόχο τη βελτίωση της αντιληπτικής επίγνωσης και τη μείωση της γνωστικής επιβάρυνσης. Η AR διαπαφή προσφέρει ένα δυναμικό και εργονομικό περιβάλλον αλληλεπίδρασης, όπου η παρουσίαση πληροφορίας είναι προσαρμοσμένη στις ανάγκες, το επίπεδο προσοχής και το πλαίσιο του οδηγού, επιτυγχάνοντας υψηλά επίπεδα ευχρηστίας και ασφάλειας. Παράλληλα, η συνδυαστική χρήση ΕΠ και ML δημιουργεί ένα προσαρμοστικό οικοσύστημα εμπειριών, στο οποίο η πληροφορία μετατρέπεται σε πρακτική γνώση μέσω αλληλεπίδρασης.

Έρευνες έχουν δείξει ότι η ενεργή συμμετοχή του χρήστη σε εμπειρίες εξατομικευμένα σχεδιασμένες σύμφωνα με τον ρυθμό και τις ανάγκες του ενισχύει την παρακίνηση, τη διατήρηση της γνώσης και τη μαθησιακή απόδοση. Έτσι, η ΕΠ δύναται να λειτουργήσει ως πλατφόρμα βιωματικής μάθησης και εκπαίδευσης σε πραγματικό χρόνο, επιτρέποντας την καλύτερη κατανόηση του περιβάλλοντος και την έγκαιρη αναγνώριση κινδύνων. Συνολικά, τα ευρήματα υπογραμμίζουν ότι η χρησιμότητα και η χρηστικότητα μιας εφαρμογής Επαυξημένης Πραγματικότητας είναι άρρηκτα συνδεδεμένες με την ποιότητα της προσφερόμενης εμπειρίας, ενώ η ενσωμάτωση μηχανικής μάθησης και χρηστοκεντρικού σχεδιασμού οδηγεί στη δημιουργία έξυπνων, ασφαλών και προσαρμοστικών συστημάτων, τα οποία συμβάλλουν ουσιαστικά στη βελτίωση της οδικής ασφάλειας και στη μετάβαση προς μια πιο ανθρώπινη και βιωματικά ενσυνείδητη τεχνολογική εμπειρία.

Ως απάντηση στο δεύτερο ερευνητικό ερώτημα, το οποίο διερευνά ποιες είναι οι βασικές προκλήσεις, τα οφέλη και οι μελλοντικές προοπτικές εφαρμογής της Επαυξημένης Πραγματικότητας (ΕΠ) στον τομέα της οδικής ασφάλειας και της εμπειρίας του χρήστη, προκύπτει ότι η ενσωμάτωση της ΕΠ στο πλαίσιο των τεχνολογιών του Industry 4.0 συνιστά έναν από τους πιο δυναμικούς και υποσχόμενους τομείς της σύγχρονης τεχνολογικής εξέλιξης. Μελέτες που έχουν διεξαχθεί, έχουν αποδείξει ότι η εφαρμογή των τεχνολογιών του Industry 4.0 και των συναφών ψηφιακών εργαλείων, όπως η Επαυξημένη Πραγματικότητα (ΕΠ), η Τεχνητή Νοημοσύνη (ΑΙ), το Διαδίκτυο των Πραγμάτων (IoT) και τα Μεγάλα Δεδομένα (Big Data), επιφέρει ένα πλήθος σημαντικών πλεονεκτημάτων στον τομέα της οδικής ασφάλειας και της

εμπειρίας του χρήστη. Οι τεχνολογίες αυτές συνεισφέρουν σε αυξημένη αποδοτικότητα, ταχύτερη επεξεργασία δεδομένων, βελτιωμένη ακρίβεια ανίχνευσης κινδύνων και προληπτική λήψη αποφάσεων, καθιστώντας τα οχήματα και τις αστικές υποδομές πιο έξυπνα, ασφαλή και συνεργατικά. Η ενσωμάτωση της ΕΠ στο οικοσύστημα του Connected and Autonomous Driving (CAV) συμβάλλει στη μείωση των τυφλών σημείων, στην ενίσχυση της αντιληπτικής επίγνωσης του οδηγού, καθώς και στην προσομοίωση επικίνδυνων καταστάσεων με τρόπο βιωματικό και εκπαιδευτικό.

Ειδικότερα, στα πλαίσια της Τέταρτης Βιομηχανικής Επανάστασης και της τρέχουσας εξελικτικής πορείας αναδεικνύονται μια σειρά από προκλήσεις όπως για παράδειγμα οι κίνδυνοι που ελλοχεύουν από επιθέσεις κυβερνοασφάλειας, οι προστασία ευαίσθητων και σημαντικών προσωπικών δεδομένων (GDPR), εξαιτίας των τεχνικών κενών ασφαλείας των διαφόρων συστημάτων, η σωστή διαχείριση και ανάλυση του εκθετικά αυξανόμενου όγκου των big data, το αρκετά υψηλό κόστος εγκατάστασης και συντήρησης των νέων τεχνολογικών συστημάτων, το οποίο δυσχεραίνει σε έναν μεγάλο βαθμό την πλήρη ψηφιακή μετάβαση των επιχειρήσεων και των βιομηχανιών, την αύξηση των κοινωνικών, καθώς και των επαγγελματικών ανισοτήτων εξαιτίας του ψηφιακού αναλφαριθμητισμού, ή ημιμάθειας των όλο και περισσότερο εξελισσόμενων τεχνολογιών, η έλλειψη ανθρώπινου δυναμικού υψηλής τεχνολογικής εξειδίκευσης, η δυσχέρεια στην διαλειτουργικότητα μεταξύ της πληθώρας διαφορετικών λογισμικών και υπολογιστικών υλικών που πρέπει απαραίτητα να εναρμονιστούν μεταξύ τους, η τεράστια και επιτακτική ανάγκη για πράσινα φιλικά προς το περιβάλλον συστήματα κ.α. Όσο τα ζητήματα που προαναφέρθηκαν δεν αντιμετωπίζονται ουσιαστικά, ελλοχεύει ο κίνδυνος να μετατραπούν από προκλήσεις σε αδυναμίες μετάβασης στην νέα εξελιγμένη εποχή του Industry 4.0. Ακόμη, όπως προαναφέρθηκε τον ακρογωνιαίο λίθο αποτελεί ο ψηφιακός μετασχηματισμός (digital transformation), των επιχειρήσεων, των έξυπνων πόλεων και των βιομηχανιών, ο οποίος κρίνεται επιτακτικής σημασίας, καθώς θα αποτελέσει τον αρωγό στην βελτίωση της παραγωγικότητας, στην αύξηση της παραγωγικότητας και της αποδοτικότητας, στην καινοτομία και στην γενικότερη εναρμόνιση τους με τους όλο και περισσότερο καλπάζοντες τεχνολογικούς ρυθμούς της σύγχρονης εποχής. Επίσης, βασικό πυλώνα αυτής της μετάβασης εκτός από την εγκατάσταση και την χρήση σύγχρονων συστημάτων, είναι και η επιτακτική ανάγκη για εργατικό δυναμικό υψηλής και διαρκούς ειδίκευσης στις νέες τεχνολογίες.

Συμπερασματικά, μέσα από την εκτεταμένη μελέτη του θέματος, είναι εμφανές

ότι η Επαυξημένη Πραγματικότητα δύναται να αναβαθμίσει ριζικά την οδική ασφάλεια και την εμπειρία του χρήστη, εφόσον αντιμετωπιστούν οι παραπάνω προκλήσεις με ολιστικό σχεδιασμό, τεχνολογική ωριμότητα και ανθρωποκεντρική προσέγγιση. Οι εργαζόμενοι σε βιομηχανίες, επιχειρήσεις και οργανισμούς επηρεάζονται σε αρκετά μεγάλο βαθμό από την εμφάνιση των νέων τεχνολογιών, καθώς οι διάφορες διαδικασίες αυτοματοποιούνται συνεχώς μέσω των έξυπνων συστημάτων και λόγω αυτού προκύπτει όλο και περισσότερο η ανάγκη για την απόκτηση εξειδικευμένων γνώσεων, προκειμένου να είναι αρκετά ανταγωνιστικοί και αποδοτικοί ούτως ώστε να μπορέσουν να διατηρήσουν την εργασία τους. Επιπροσθέτως, οι τεχνολογίες του Industry 4.0 (IoT, big data, AI, robotics, cloud computing κ.λ.π.), οδηγούν με το πέρασμα του χρόνου στην επιτακτική ανάγκη της υιοθέτησης ενός νέου μοντέλου δια βίου μάθησης και επανειδίκευσης του εργατικού δυναμικού των επιχειρήσεων και των βιομηχανιών. Παράλληλα, εγείρονται μια σειρά από ηθικά και κοινωνικά ζητήματα, τα οποία αφορούν κατά κύριο λόγο την βίαιη προσαρμογή στις νέες συνθήκες, και στους νέους όρους εργασίας, τους οποίους φέρνει η τεχνολογική μετάβαση σε πλείστους επαγγελματικούς κλάδους. Ο τεχνολογικός αναλφαριθμητισμός, η έλλειψη εξειδικευμένων ψηφιακών δεξιοτήτων και τα ανεπαρκή προσόντα των εργαζομένων είναι δυσχέρειες, οι οποίες χρήζουν άμεσης αντιμετώπισης. Οι μελλοντικές προοπτικές εστιάζουν στη δημιουργία συνεργατικών, προσαρμοστικών και βιώσιμων οικοσυστημάτων κινητικότητας, όπου ο άνθρωπος και η τεχνολογία συνυπάρχουν αρμονικά, προάγοντας την ασφάλεια, την εκπαίδευση και την κοινωνική πρόοδο.

Εν κατακλείδι, η παρούσα μελέτη ολοκληρώνεται απαντώντας στο τρίτο ερευνητικό ερώτημα, το οποίο διερευνά τη συμβολή της Επαυξημένης Πραγματικότητας στη μετάβαση προς ένα νέο μοντέλο οδικής κυκλοφορίας και ασφάλειας, καθώς και τους πιθανούς τρόπους ενίσχυσης της εκπαίδευσης και της λήψης αποφάσεων των χρηστών στο οδικό δίκτυο. Η ανάλυση καταδεικνύει ότι η Επαυξημένη Πραγματικότητα (ΕΠ) συνιστά καθοριστικό παράγοντα στη διαμόρφωση των έξυπνων συστημάτων μεταφορών της επόμενης γενιάς, λειτουργώντας ως καταλύτης για τη μετάβαση από την παραδοσιακή οδηγική εμπειρία σε ένα προσαρμοστικό, διαδραστικό και γνωσιακά ενισχυμένο περιβάλλον οδήγησης. Η ΕΠ επιτρέπει την ενσωμάτωση πληροφοριών ασφάλειας, πλοήγησης και κινδύνου απευθείας στο οπτικό πεδίο του οδηγού, συμβάλλοντας στην πρόληψη ατυχημάτων και στην άμεση αντίδραση σε δυναμικά γεγονότα του δρόμου. Παράλληλα, λειτουργεί ως εκπαιδευτικό

εργαλείο, επιτρέποντας στους χρήστες να αναπτύσσουν αντιληπτικές δεξιότητες, να κατανοούν σύνθετες κυκλοφοριακές συνθήκες και να λαμβάνουν πιο τεκμηριωμένες αποφάσεις μέσω πρακτικής βιωματικής μάθησης.

Η συμβολή της ΕΠ ενισχύεται περαιτέρω από τις γραφικές τέχνες και τον επιστημονικό σχεδιασμό διεπαφών, οι οποίοι διασφαλίζουν ότι η πληροφορία αποδίδεται αισθητικά ισορροπημένα και λειτουργικά, χωρίς να προκαλεί υπερφόρτωση ή σύγχυση στον χρήστη. Ο γραφιστικός σχεδιασμός εμπειριών (experience design) συνδυάζει τις αρχές της οπτικής επικοινωνίας, της αντίθεσης, της ιεραρχίας και της ισορροπίας, με στόχο τη δημιουργία ρεαλιστικών, ευανάγνωστων και εργονομικών περιβαλλόντων ΕΠ. Η σωστή χρήση χρώματος, τυπογραφίας, φωτισμού και μορφής δεν αποτελεί απλώς αισθητική επιλογή, αλλά καθοριστικό παράγοντα για την κατανόηση, τη συγκέντρωση και τη λήψη αποφάσεων του οδηγού. Έτσι, η διεπαφή ΕΠ μετατρέπεται σε γνωσιακό υποστηρικτικό σύστημα, ικανό να ενισχύσει τη συμπεριφορική ασφάλεια και να μειώσει την πιθανότητα ανθρώπινου σφάλματος.

Στο πλαίσιο του Industry 4.0, η σύζευξη ΕΠ, Μηχανικής Μάθησης και Γραφιστικής Επιστήμης επιτρέπει τη μετατροπή των πολύπλοκων τεχνολογιών αντίληψης και πρόβλεψης σε φιλικά προς τον χρήστη περιβάλλοντα (user-friendly interfaces), αυξάνοντας την αποδοχή, την προσβασιμότητα και την αποτελεσματικότητα των συστημάτων αυτών. Μέσα από αυτή τη σύμπραξη, η Επαυξημένη Πραγματικότητα παύει να είναι απλώς ένα εργαλείο απεικόνισης και εξελίσσεται σε ενεργό μηχανισμό εκπαίδευσης, καθοδήγησης και γνωσιακής υποστήριξης.

Συνοψίζοντας, η συμβολή της Επαυξημένης Πραγματικότητας στη μετάβαση προς ένα νέο, ασφαλέστερο και πιο διαδραστικό μοντέλο οδικής κυκλοφορίας είναι πολυδιάστατη: ενώνει τη λειτουργικότητα με την αισθητική, τη μάθηση με την πράξη, και τον άνθρωπο με την τεχνολογία. Μέσω αυτής της ολοκληρωμένης προσέγγισης, διαμορφώνεται μια νέα εποχή οδικής εμπειρίας, όπου η πληροφορία γίνεται αντίληψη, η αντίληψη γνώση, και η γνώση ασφάλεια.

5.4 Βελτιστοποιήσεις και Περιορισμοί

Η επίδοση του συστήματος μπορεί να βελτιωθεί σημαντικά μέσω της αξιοποίησης υπολογιστικών πόρων υψηλής απόδοσης, όπως οι κάρτες γραφικών που υποστηρίζουν την πλατφόρμα CUDA, που θεωρητικά προβλέπονται σε ελεγκτές συνδεδε-

μένων αυτονόμων οχημάτων. Η εκτέλεση των αλγορίθμων μηχανικής μάθησης σε GPU και επεξεργαστές A.I., αντί για CPU, επιτρέπει την παράλληλη επεξεργασία μεγάλων όγκων δεδομένων εικόνας, μειώνοντας δραστικά τον χρόνο εκτέλεσης των διαδικασιών ανίχνευσης και οπτικοποίησης. Εναλλακτικά, η μεταγλώττιση του Python κώδικα σε C ή C++ μέσω εργαλείων όπως το Cython ή το PyBind11 μπορεί να επιφέρει σημαντικά κέρδη σε ταχύτητα, περιορίζοντας την καθυστέρηση που οφείλεται στη δυναμική φύση της Python.

Στον αντίποδα πρέπει να αναφερθεί ότι η παρούσα υλοποίηση βασίστηκε σε επεξεργασία στατικών εικόνων (σειρά φωτογραφιών), γεγονός που απλοποιεί τη ροή δεδομένων και μειώνει την πολυπλοκότητα της ανάλυσης. Σε ένα σύστημα ζωντανής ροής (real-time video) προστίθενται επιπλέον στάδια όπως η αποσυμπίεση και αποκωδικοποίηση των καρέ, τα οποία προηγούνται της ανίχνευσης αντικειμένων και αυξάνουν τον συνολικό χρόνο απόκρισης. Η μετάβαση σε πραγματικό χρόνο απαιτεί, επομένως, βελτιστοποιημένους αλγορίθμους, παράλληλη επεξεργασία και εξειδικευμένο υλικό GPU, ώστε να διασφαλιστεί η σταθερή και άμεση λειτουργία του συστήματος σε περιβάλλοντα υψηλής δυναμικής, όπως αυτά της οδικής κυκλοφορίας.

Επιπλέον, κατά τη διάρκεια της πειραματικής υλοποίησης και αξιολόγησης του συστήματος, προέκυψαν ζητήματα που σχετίζονταν με τον ρεαλισμό της οπτικοποίησης και την πολυπλοκότητα της προβολής τρισδιάστατων μοντέλων, κυρίως λόγω της μη χρήσης βιβλιοθηκών γραφικών, όπως η OpenGL. Η απόφαση αυτή ελήφθη με στόχο τη βελτιστοποίηση της ταχύτητας εκτέλεσης και τη μείωση του υπολογιστικού κόστους, ώστε το σύστημα να είναι συμβατό με συσκευές χαμηλής υπολογιστικής ισχύος (edge devices) και να λειτουργεί αυτόνομα, χωρίς εξάρτηση από υποδομές υπολογιστικού Νέφους (Cloud). Η επιλογή αυτή εξασφάλισε πραγματικό χρόνο επεξεργασίας, αλλά ταυτόχρονα οδήγησε σε περιορισμούς ως προς την απεικόνιση και την ποιότητα γραφικών.

Πιο συγκεκριμένα, η απουσία των υπολογιστικών λειτουργιών της OpenGL, όπως σκίαση, clipping και smoothing, εμπόδισε την παραγωγή γραφικών υψηλής πιστότητας σε πραγματικό χρόνο, καθιστώντας τη μέθοδο εικονικοποίησης μέσω 3D μοντέλων λιγότερο αποδοτική και, συνεπώς, μη προτεινόμενη για το παρόν στάδιο ανάπτυξης του συστήματος. Αντίστοιχα, κατά την εικονικοποίηση μέσω 2D ψηφιογραφικών αντικειμένων, παρατηρήθηκε ασυμφωνία μεταξύ της προβαλλόμενης εικόνας και της πραγματικής προοπτικής του αντικειμένου ενδιαφέροντος, γεγονός

που επηρέαζε την αντιληπτική συνέπεια της Επαυξημένης Πραγματικότητας.

Μία πιθανή λύση για τη βελτίωση της απόδοσης θα μπορούσε να αποτελέσει η εφαρμογή της τεχνικής billboarding, δηλαδή του δυναμικού προσανατολισμού δισδιάστατων αντικειμένων προς την κάμερα, ώστε να διατηρείται η οπτική συνέπεια ανεξάρτητα από τη θέση του παρατηρητή. Η συγκεκριμένη τεχνική, ωστόσο, δεν υλοποιήθηκε λόγω της έλλειψης πολλαπλών γωνιών προβολής στα διαθέσιμα ψηφιογραφικά αντικείμενα. Σε ένα περιβάλλον παραγωγής, η συνεργασία με επαγγελματίες σχεδιαστές γραφικών θα μπορούσε να επιλύσει αυτόν τον περιορισμό, παρέχοντας γραφικά πολλαπλής προοπτικής και υψηλότερης αισθητικής πιστότητας, βελτιώνοντας σημαντικά τη ρεαλιστική απεικόνιση και τη συνολική εμπειρία χρήστη.

ΚΕΦΑΛΑΙΟ 6

ΣΥΜΠΕΡΑΣΜΑΤΑ

6.1 Συμπεράσματα

6.2 Μελλοντικές Έξελξεις

6.1 Συμπεράσματα

Η συνολική ανάλυση της παρούσης διπλωματικής εργασίας αναδεικνύει τον πολυδιάστατο ρόλο της Επαυξημένης Πραγματικότητας (AR) και των σύγχρονων τεχνολογιών του Industry 4.0 στη διαμόρφωση μιας νέας ψηφιακής πραγματικότητας, όπου ο χρήστης βρίσκεται στο επίκεντρο του σχεδιασμού και της εμπειρίας. Η μελέτη δείχνει ότι η επαυξημένη πραγματικότητα, σε συνδυασμό με τεχνολογίες όπως η μηχανική μάθηση, το διαδίκτυο των πραγμάτων (IoT) και η τεχνητή νοημοσύνη (AI), προσφέρει ένα καινοτόμο πλαίσιο βιωματικής μάθησης και αλληλεπίδρασης, που ξεπερνά τα όρια της απλής ψηφιακής αναπαράστασης. Η χρηστοκεντρική προσέγγιση (User Centered Design) καθίσταται καθοριστική, καθώς επιτρέπει στους σχεδιαστές να δημιουργούν εμπειρίες προσαρμοσμένες στις ανάγκες, τις ικανότητες και τον ρυθμό του κάθε χρήστη, μετατρέποντας τη χρήση της τεχνολογίας σε μία πραγματικά προσωποποιημένη και βιωματική διαδικασία. Παράλληλα, το Industry 4.0 σηματοδοτεί τη μετάβαση προς μια νέα εποχή αυτοματοποίησης, συνδεσιμότητας και έξυπνων συστημάτων, η οποία επηρεάζει όχι μόνο τις επιχειρήσεις και τη βιομηχανία, αλλά και την καθημερινότητα των ανθρώπων. Παρά τα σημαντικά πλεονεκτήματα αναδύονται προκλήσεις που σχετίζονται με την κυβερνοασφάλεια,

την προστασία προσωπικών δεδομένων, την έλλειψη ψηφιακών δεξιοτήτων και το υψηλό κόστος υλοποίησης των νέων τεχνολογιών. Σε αυτό το περιβάλλον, οι γραφικές τέχνες διαδραματίζουν καθοριστικό ρόλο, καθώς γεφυρώνουν τη λειτουργικότητα με την αισθητική, δημιουργώντας διαισθητικά και ελκυστικά περιβάλλοντα διεπαφής (UI/UX), που διευκολύνουν την κατανόηση και την αποδοχή των νέων τεχνολογιών από το ευρύ κοινό. Η σωστή εφαρμογή των αρχών οπτικής επικοινωνίας, της ισορροπίας, της ιεραρχίας και της εργονομίας ενισχύει την ευχρηστία και τον ρεαλισμό των AR εμπειριών, καθιστώντας τις πιο προσιτές και αποτελεσματικές. Τελικά, η συνέργεια μεταξύ επαυξημένης πραγματικότητας, χρηστοκεντρικού σχεδιασμού, γραφικών τεχνών και τεχνολογιών του Industry 4.0 διαμορφώνει ένα νέο, ανθρωποκεντρικό τεχνολογικό οικοσύστημα, στο οποίο η γνώση, η εμπειρία και η αλληλεπίδραση συνενώνονται, ενισχύοντας την εκπαίδευση, την επαγγελματική ανάπτυξη και την ποιότητα ζωής σε όλα τα επίπεδα.

Το σύστημα που υλοποιήθηκε στο πλαίσιο της επαυξημένης πραγματικότητας για τα συνδεδεμένα αυτόνομα οχήματα αφορά τμήμα του Ε.Π.Ο.Σ. - σύστημα Επαυξημένης Πραγματικότητας για την Οδική ασφάλεια έξυπνων πόλεων Συνδεδεμένων αυτόνομων οχημάτων, μια ολοκληρωμένη αρχιτεκτονική που συνδυάζει αισθητήρες LiDAR, κάμερες RGB, το σύστημα ακριβούς εντοπισμού OXTS GNSS/INS, και αλγορίθμους μηχανικής μάθησης όπως το YOLOv11, με στόχο την ενίσχυση της οδικής ασφάλειας. Το σύστημα συλλέγει, επεξεργάζεται και συνδυάζει δεδομένα θέσης, βάθους και οπτικής αντίληψης για την αναγνώριση αντικειμένων, την παραγωγή χαρτών βάθους και την προβολή 3D αντικειμένων στο οπτικό πεδίο του οδηγού μέσω διεπαφής επαυξημένης πραγματικότητας (AR UI). Η διεπαφή αυτή προβάλλει δυναμικά πληροφορίες όπως ανιχνευμένα οχήματα, πεζούς, αποστάσεις, προειδοποιήσεις και ζώνες κινδύνου, χρησιμοποιώντας τεχνικές overlay, blending και δυναμικής διαφάνειας ώστε να διασφαλίζεται η καθαρότητα και η ασφάλεια της οδήγησης. Το σύστημα αποτελεί μια πολυεπίπεδη πλατφόρμα επαυξημένης αντίληψης που ενσωματώνει real-time επεξεργασία, cloud intelligence και AR απεικόνιση, συμβάλλοντας ουσιαστικά στη βελτίωση της επίγνωσης και της ασφάλειας του οδηγού σε συνθήκες έξυπνης και αυτόνομης κινητικότητας.

6.2 Μελλοντικές Έξελιξεις

Στο πλαίσιο της μελλοντικής εξέλιξης του συστήματος, προτείνεται η πλήρης επέκταση και ολοκλήρωση του συστήματος Ε.Π.Ο.Σ., με στόχο την περαιτέρω ενίσχυση της λειτουργικότητας και της αξιοπιστίας του. Επόμενο βήμα αποτελεί η υλοποίηση του Έξυπνου Σηματοδότη, ο οποίος θα λειτουργεί ως κόμβος επικοινωνίας και λήψης αποφάσεων σε πραγματικό χρόνο, ενσωματώνοντας δεδομένα από ποικίλους αισθητήρες και πηγές πληροφόρησης. Παράλληλα, προβλέπεται η ανάπτυξη υποδομής cloud, η οποία θα επιτρέπει την αποθήκευση, τη μετεπεξεργασία και την ανάλυση των μεγαδεδομένων αυτόνομης οδήγησης μέσω πρακτόρων βαθιάς μάθησης (deep learning agents). Η αξιοποίηση αυτών των δεδομένων στοχεύει στη συνεχή βελτίωση της οδικής ασφάλειας και στη δημιουργία μηχανισμών πρόβλεψης και αποφυγής κινδύνων.

Παράλληλα, ιδιαίτερη έμφαση δίνεται στην υλοποίηση και πλήρη υποστήριξη της συνεργατικής επικοινωνίας CAV2CAV (Connected and Autonomous Vehicles to Connected and Autonomous Vehicles). Η τεχνολογία αυτή θα επιτρέπει την ανταλλαγή δεδομένων αισθητήρων και τρισδιάστατων αντικειμένων μεταξύ οχημάτων, προσφέροντας τη δυνατότητα πρόβλεψης κινδύνων πέρα από το οπτικό πεδίο και συμβάλλοντας καθοριστικά στη μείωση των ατυχημάτων και στην αύξηση του επιπέδου ασφάλειας στο οδικό δίκτυο.

Τέλος, προτείνεται η διερεύνηση εναλλακτικών στρατηγικών βελτιστοποίησης που αφορούν την απόδοση και την οπτική ποιότητα του συστήματος. Συγκεκριμένα, δύναται να μελετηθεί η πλήρης αξιοποίηση μηχανών γραφικής απεικόνισης επαγγελματικής κλίμακας, με κυριότερους εκπροσώπους την Unity και την Unreal Engine. Οι πλατφόρμες αυτές παρέχουν προηγμένα εργαλεία φυσικής προσομοίωσης, φωτισμού, σκίασης και απόδοσης υλικών, τα οποία μπορούν να συμβάλουν σημαντικά στη βελτίωση της ρεαλιστικότητας και της αποδοτικότητας των τρισδιάστατων απεικονίσεων. Η χρήση τέτοιων μηχανών θα επιτρέψει τη συγκριτική αξιολόγηση της γραφικής ποιότητας και της απόδοσης έναντι βασικότερων μεθόδων σχεδίασης, όπως η OpenGL, προσφέροντας πολύτιμα δεδομένα σχετικά με την αποτελεσματικότητα και την αποδοτικότητα του συστήματος. Μέσω της προσέγγισης αυτής, μπορεί να επιτευχθεί μια ουσιαστική αναβάθμιση της συνολικής εμπειρίας χρήστη, ιδιαίτερα σε εφαρμογές Επαυξημένης Πραγματικότητας και αυτόνομης οδήγησης όπου η οπτικοποίηση δεδομένων αποτελεί κρίσιμο παράγοντα επιτυχίας.

ΒΙΒΛΙΟΓΡΑΦΙΑ

- [1] A. M. Aslam, R. Chaudhary, A. Bhardwaj, N. Kumar, and R. Buyya, “Digital twins-enabled game theoretical models and techniques for metaverse connected and autonomous vehicles: A survey,” *Journal of Network and Computer Applications*, vol. 238, p. 104138, 2025. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1084804525000359>
- [2] B. Brik, H. Moustafa, Y. Zhang, A. Lakas, and S. Subramanian, “Guest editorial: Multi-access networking for extended reality and metaverse,” *IEEE Internet of Things Magazine*, vol. 6, no. 1, pp. 12–13, 2023.
- [3] P. Zhou, W. Zhang, T. Braud, P. Hui, and J. Kangasharju, “Enhanced augmented reality applications in vehicle-to-edge networks,” in *2019 22nd Conference on Innovation in Clouds, Internet and Networks and Workshops (ICIN)*, 2019, pp. 167–174.
- [4] Y. Cheng, J. Nakazato, E. Javanmardi, C.-M. Chang, and M. Tsukada, “Pedestrian-centric augmented reality visualization of real-time autonomous vehicle dynamics,” in *2023 IEEE 12th International Conference on Cloud Networking (CloudNet)*, 2023, pp. 460–465.
- [5] Z. Wang, K. Han, and P. Tiwari, “Augmented reality-based advanced driver-assistance system for connected vehicles,” in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2020, pp. 752–759.
- [6] S. Sharma, S. Bhadula, and A. Juyal, “Augmented reality for next generation connected and automated vehicles,” in *2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT)*, vol. 5, 2024, pp. 23–28.

- [7] H. Qiu, F. Ahmad, F. Bai, M. Gruteser, and R. Govindan, “Avr: Augmented vehicular reality,” in *Proceedings of the 16th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys ’18. New York, NY, USA: Association for Computing Machinery, 2018, p. 81–95. [Online]. Available: <https://doi.org/10.1145/3210240.3210319>
- [8] H. Wang, Z. Wang, D. Chen, Q. Liu, H. Ke, and K. K. Han, “Metamobility: Connecting future mobility with the metaverse,” *IEEE Vehicular Technology Magazine*, vol. 18, no. 3, pp. 69–79, 2023.
- [9] A. Lopes, R. Souza, and H. Pedrini, “A survey on rgb-d datasets,” *Computer Vision and Image Understanding*, vol. 222, p. 103489, Sep. 2022. [Online]. Available: <http://dx.doi.org/10.1016/j.cviu.2022.103489>
- [10] Y. Liao, J. Xie, and A. Geiger, “KITTI-360: A novel dataset and benchmarks for urban scene understanding in 2d and 3d,” *Pattern Analysis and Machine Intelligence (PAMI)*, 2022.
- [11] G. Siasias, M. Konstantinidou, and S. Kontogiannis, “Deep learning process and application for the detection of dangerous goods passing through motorway tunnels,” *Algorithms*, vol. 15, no. 10, 2022. [Online]. Available: <https://www.mdpi.com/1999-4893/15/10/370>
- [12] S. Kontogiannis, M. Konstantinidou, V. Tsioukas, and C. Pikridas, “A cloud-based deep learning framework for downy mildew detection in viticulture using real-time image acquisition from embedded devices and drones,” *Information*, vol. 15, no. 4, 2024. [Online]. Available: <https://www.mdpi.com/2078-2489/15/4/178>
- [13] T. Gkamas, S. Kontogiannis, V. Karaiskos, C. Pikridas, and I. A. Karolos, “Proposed cloud-assisted machine learning classification process implemented on industrial systems: Application to critical events detection and industrial maintenance,” in *2022 5th World Symposium on Communication Engineering (WSCE)*, 2022, pp. 95–99.
- [14] T. Gkamas, V. Karaiskos, and S. Kontogiannis, “Evaluation of cloud databases as a service for industrial iot data,” in *Proceedings of Seventh International Congress*

- on *Information and Communication Technology*, X.-S. Yang, S. Sherratt, N. Dey, and A. Joshi, Eds. Singapore: Springer Nature Singapore, 2023, pp. 273–281.
- [15] —, “Performance evaluation of distributed database strategies using docker as a service for industrial iot data: Application to industry 4.0,” *Information*, vol. 13, no. 4, 2022. [Online]. Available: <https://www.mdpi.com/2078-2489/13/4/190>
- [16] V. Karaiskos, N. Zinas, T. Gkamas, I. A. Karolos, C. Pikridas, N. Vrettos, V. Tsioukas, and S. Kontogiannis, “Proposed industry 4.0 maintenance framework for critical and demanding infrastructures and processes,” in *2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, 2022, pp. 1–5.
- [17] Texas Instruments, “In-Plane Holographic Optical Element (IP-HOE) Display — texas instruments application report,” Texas Instruments, Tech. Rep. DLPA118, 2024. [Online]. Available: <https://www.ti.com/lit/wp/dlpa118/dlpa118.pdf>
- [18] HYUNDAI-MOBIS, “Holographic windshield display,” Jan. 2025. [Online]. Available: <https://www.youtube.com/watch?v=tItXoT-VAQ0>
- [19] Intelligent-Glass, “Switchable smart film,” 2025. [Online]. Available: <https://www.intelligentglass.net/products/switchable-smart-film/>
- [20] H. Kim and I.-K. Lee, “In-vehicle full-window augmented reality system (fars) for passengers: Effects on spatial knowledge and user experience,” *Research Square*, January 2024, preprint, Version 1. [Online]. Available: <https://doi.org/10.21203/rs.3.rs-3849600/v1>
- [21] S. M. T. C. Ltd., “Transparent led film screen – muxwave holographic invisible screen,” 2025. [Online]. Available: <https://www.muxwave.com/transparent-led-screen/led-film-screen/>
- [22] “Δείγματα φύλλων ψηφιογραφικών αντικειμένων (spritesheets),” www.freepik.com, 2025, Άδεια: Creative Commons License CC-BY-4.0 (<http://creativecommons.org/licenses/by/4.0/>).
- [23] “Δείγματα φύλλων ψηφιογραφικών αντικειμένων (spritesheets),” www.pngwing.com, 2025, Άδεια: Creative Commons License CC-BY-4.0 (<http://creativecommons.org/licenses/by/4.0/>).

[24] Kulonee, “Low poly lada vesta,” <https://sketchfab.com/3d-models/low-poly-lada-vesta-b804fab0498f42be8b40f0201d7ed7c0>, 2025, Χρήστης: kulonee (<https://sketchfab.com/ynesolefru>); Άδεια: Creative Commons License CC-BY-4.0 (<http://creativecommons.org/licenses/by/4.0/>).

ΠΑΡΑΡΤΗΜΑ Α

ΠΗΓΑΙΟΣ ΚΩΔΙΚΑΣ ΒΑΣΙΚΟΥ ΣΥΣΤΗΜΑΤΟΣ

```
#!/usr/bin/python
# -*- coding: utf-8 -*-

#####
## Import modules
#####
from enum import Enum
import math
import shutil
import sys
import os
import glob
import numpy as np
import cv2
from ultralytics import YOLO
import pandas as pd
import open3d
import json
from collections import namedtuple
import pandas as pd
import xml.etree.ElementTree as ET
import time
```

```

# import open3d as o3d

kitti360Path = '/home/pcs00480/kitti/kitti360_dataset/dataset/'
spritePath=kitti360Path + 'sprites/'
Point = namedtuple('Point', ['x', 'y'])
er = 6378137. # average earth radius at the equator
maxDrawDistance=50

YOLO_classes={0: 'pedestrian', 1: 'bicycle', 2: 'car', 3: 'motorcycle', 5: 'bus',
6: 'train', 7: 'truck', 8: 'boat', 9: 'traffic light', 11: 'stop sign',
14: 'bird', 15: 'cat', 16: 'dog', 17: 'horse', 18: 'sheep', 19: 'cow',
21: 'bear', 32: 'sports ball', 36: 'skateboard', 39: 'bottle'}

KITTI_classes={0: 'pedestrian',1: 'bicycle', 2: 'car', 3: 'motorcycle', 5: 'bus',
6: 'train', 7: 'truck',9: 'trafficLight', 11: 'stop', 80: 'bigPole', 81: 'box',
82: 'bridge', 83: 'building', 84: 'caravan', 85: 'garage', 86: 'lamp',
87: 'rider', 88: 'smallPole', 89: 'trailer', 90: 'trafficSign', 91: 'trashbin',
92: 'tunnel', 93: 'unknownConstruction', 94: 'unknownObject', 95: 'unknownVehicle',
96: 'vendingmachine'}

img_classes={0: 'pedestrian',1: 'bicycle', 2: 'car', 3: 'motorcycle', 5: 'bus',
11: 'stop'}
image_dict={}

class Orientation(Enum):
    FRONT=0,
    RIGHT=1,
    BACK=2,
    LEFT=3

class SpriteData(Enum):
    SIDE=0,

```

```

FRONT=1,
BACK=2

CAR={SpriteData.SIDE:((0,0),(1400,500)),
     SpriteData.FRONT:((1410,0),(2075,500)),
     SpriteData.BACK:((2150,0),(2815,500))}
MOTORBIKE={SpriteData.FRONT:((0,0),(324,500)),
            SpriteData.SIDE:((340,0),(992,500))}
BUS={SpriteData.FRONT:((0,0),(457,500)),
     SpriteData.SIDE:((474,0),(1889,500))}

# input_ply = "/home/pcs00480/Downloads/car3.ply"
# plycar=o3d.io.read_triangle_mesh(input_ply)

def loadImages():
    for index, label in img_classes.items():
        imageFile = spritePath +str(index)+ '.*.png'
        for fname in glob.glob(imageFile):
            if not os.path.isfile(fname):
                raise RuntimeError('Image file %s does not exist!' % fname)
            # Read the image
            image = cv2.imread(fname, cv2.IMREAD_UNCHANGED)
            image_dict[index]={'label':img_classes[index], 'image':image}

def getPositionSelf(seq, frame):
    filename = os.path.join(kitti360Path, 'data_poses_oxts/data_poses',
                             seq, 'oxts/data','%010d.txt' % frame)
    checkfile(filename)
    data=np.loadtxt(filename)
    lat=data[0]
    lon= data[1]
    alt=data[2]
    return lat, lon, alt

```

```

def loadTs(seq, is_oxts=False):

    filename = os.path.join(kitti360Path, 'data_3d_raw', seq,
        'velodyne_points/timestamps.txt')
    if is_oxts:
        filename = os.path.join(kitti360Path, 'data_poses_oxts/data_poses', seq,
            'oxts/timestamps.txt')

    checkfile(filename)

    content = pd.read_csv(filename, header=None, names=['timestamp'],
        parse_dates=['timestamp'])
    return content

def latlonToMercator(lat,lon,scale):
    ''' converts lat/lon coordinates to mercator coordinates using mercator scale '''

    mx = scale * lon * np.pi * er / 180
    my = scale * er * np.log( np.tan((90+lat) * np.pi / 360) )
    return mx,my

def postprocessPoses (poses_in):

    R = np.array([[1,0,0,0], [0,-1,0,0], [0,0,-1,0], [0,0,0,1]])

    poses = []

    for i in range(len(poses_in)):
        # if there is no data => no pose
        if not len(poses_in[i]):
            continue

        P = poses_in[i]
        poses.append( np.matmul(R, P.T).T )

```



```
return poses
```

```
def convertOxtsToPose(oxts):
```

```
''' converts a list of oxts measurements into metric poses,
starting at (0,0,0) meters, OXTS coordinates are defined as
x = forward, y = right, z = down (see OXTS RT3000 user manual)
afterwards, pose[i] contains the transformation which takes a
3D point in the i'th frame and projects it into the oxts
coordinates with the origin at a lake in Karlsruhe. '''
```

```
single_value = not isinstance(oxts, list)
```

```
if single_value:
```

```
oxts = [oxts]
```

```
# origin in OXTS coordinate
```

```
origin_oxts = [48.9843445, 8.4295857] # lake in Karlsruhe
```

```
# compute scale from lat value of the origin
```

```
''' compute mercator scale from latitude '''
```

```
scale = np.cos(origin_oxts[0] * np.pi / 180.0)
```

```
# origin in Mercator coordinate  ox 615824.312818603 oy 4116230.2656191494
```

```
ox,oy = latlonToMercator(origin_oxts[0],origin_oxts[1],scale)
```

```
origin = np.array([ox, oy, 0])
```

```
pose      = []
```

```
# for all oxts packets do
```

```
for i in range(len(oxts)):
```

```
# if there is no data => no pose
```

```
if not len(oxts[i]):
```

```
pose.append([])
```

```

continue

# translation vector
tx, ty = latlonToMercator(oxts[i][0],oxts[i][1],scale)
t = np.array([tx, ty, oxts[i][2]])

# rotation matrix (OXTS RT3000 user manual, page 71/92)
rx = oxts[i][3] # roll
ry = oxts[i][4] # pitch
rz = oxts[i][5] # heading
Rx = np.array([[1,0,0],[0,np.cos(rx),-np.sin(rx)],[0,np.sin(rx),np.cos(rx]])
# base => nav (level oxts => rotated oxts)
Ry = np.array([[np.cos(ry),0,np.sin(ry)],[0,1,0],[-np.sin(ry),0,np.cos(ry)]])
# base => nav (level oxts => rotated oxts)
Rz = np.array([[np.cos(rz),-np.sin(rz),0],[np.sin(rz),np.cos(rz),0],[0,0,1]])
# base => nav (level oxts => rotated oxts)
R = np.matmul(np.matmul(Rz, Ry), Rx)

# normalize translation
t = t-origin

# add pose
pose.append(np.vstack((np.hstack((R,t.reshape(3,1))),np.array([0,0,0,1]))))

if single_value:
    pose = pose[0]

return pose

def checkfile(filename):
    if not os.path.isfile(filename):
        raise RuntimeError('%s does not exist!' % filename)

def readVariable(fid,name,M,N):

```

```

# rewind
fid.seek(0,0)

# search for variable identifier
line = 1
success = 0
while line:
    line = fid.readline()
    if line.startswith(name):
        success = 1
        break

# return if variable identifier not found
if success==0:
    return None

# fill matrix
line = line.replace('%s:' % name, '')
line = line.split()
assert(len(line) == M*N)
line = [float(x) for x in line]
mat = np.array(line).reshape(M, N)

return mat

def loadCalibrationCameraToPose(filename):
    # check file
    checkfile(filename)

    # open file
    fid = open(filename,'r');

    # read variables
    Tr = {}

```

```

cameras = ['image_00', 'image_01', 'image_02', 'image_03']
lastrow = np.array([0,0,0,1]).reshape(1,4)
for camera in cameras:
    Tr[camera] = np.concatenate((readVariable(fid, camera, 3, 4), lastrow))

# close file
fid.close()
return Tr

def gradient_color(conf):
    # Clamp the confidence between 0 and 1
    conf = np.clip(conf, 0, 1)

    # Interpolate between red (1, 0, 0) and green (0, 1, 0)
    red = int((1 - conf) * 255) # Decrease red as confidence increases
    green = int(conf * 255)      # Increase green as confidence increases
    blue = 0
    return (blue, green, red) # OpenCV uses BGR format

#simulate live feed
def getOutsourcedData(seq, frame):

    file = os.path.join(kitti360Path, 'datastream_3d_bboxes', seq, '%010d.json'% frame)
    data=[]
    data_dict={}

    try:
        checkfile(file)
        with open(file, 'r') as json_file:
            data_dict = json.load(json_file)
        except FileNotFoundError:
            print(f"Error: The file '{file}' was not found.")
        except json.JSONDecodeError as e:
            print(f"Error: Failed to decode JSON. Details: {e}")

```

```

except PermissionError:
    print(f"Error: Permission denied when accessing the file '{file}'.")
except Exception as e:
    print(f"An unexpected error occurred: {e}")
else:
    for k in data_dict:
        mat=[]
        datastring=k['transform']
        t=datastring
        mat=np.array(t['data'],dtype=np.float64)
        mat = np.reshape(mat, [int(t['rows']),int(t['cols'])])    # [4, 4]
        data.append({"entity_id":k['index'],
            "label":k['label'],
            "transform":mat,
            "dynamic":k['dynamic']
        })
    finally:
        return {"entities":data}

def parse_xml(xml_file):
    """
    Parse an OpenCV storage XML file and convert its contents into a Pandas DataFrame.

    :param xml_file: Path to the XML file.
    :return: A Pandas DataFrame containing parsed data.
    """
    tree = ET.parse(xml_file)
    root = tree.getroot()

    # Prepare a list to hold object data
    data = []

    # Iterate through each object (e.g., object1, object2, etc.)
    for obj in root:

```

```

obj_data = {}
for child in obj:
    # Handle simple elements directly
    if child.tag not in ["transform", "vertices", "faces"]:
        obj_data[child.tag] = child.text

    # Handle transform only, vertices, faces elements separately
    elif child.tag in ["transform"]:
        matrix_data = parse_matrix(child)
        obj_data[child.tag] = matrix_data

# Append the object's data to the list
data.append(obj_data)

# Convert the list of dictionaries to a DataFrame
df = pd.DataFrame(data)
return df

def parse_matrix(element):
    """
    Parse a matrix element from the XML.

    :param element: The XML element containing matrix data.
    :return: A dictionary representing the matrix, including rows, cols, and flattened data.
    """
    matrix_info = {}
    for subchild in element:
        if subchild.tag == "data":
            # Flatten the data, split by whitespace
            matrix_info["data"] = list(map(float, subchild.text.split()))
        else:
            # For rows, cols, and dt
            matrix_info[subchild.tag] = subchild.text
    return matrix_info

```

```

def prepareOutsourcedData(seq):
    xml_file = os.path.join(kitti360Path,
        'data_3d_bboxes/train/2013_05_28_drive_%04d_sync.xml'% seq)
    df = parse_xml(xml_file)
    df_filtered = df[df['label'].isin(['pedestrian', 'bicycle', 'car', 'motorcycle',
        'bus', 'stop'])]
    df_filtered=df_filtered.drop(columns=['semanticId_orig', 'semanticId', 'instanceId',
        'category'])

    #prepare data directory
    data_path = os.path.join(kitti360Path, 'datastream_3d_bboxes/')
    if not os.path.isdir(data_path):
        os.makedirs(data_path)
    data_seq_path = os.path.join(data_path, '2013_05_28_drive_%04d_sync/'% seq)
    try:
        os.makedirs(data_seq_path, exist_ok=True)
    # Create the directory if it doesn't exist
    except OSError:
        # If the directory already exists, delete its contents
        for filename in os.listdir(data_seq_path):
            file_path = os.path.join(data_seq_path, filename)
            try:
                if os.path.isfile(file_path) or os.path.islink(file_path):
                    os.unlink(file_path)
                elif os.path.isdir(file_path):
                    shutil.rmtree(file_path)
            except Exception as e:
                print(f"Failed to delete {file_path}. Reason: {e}")

    #convert to time series
    for i in range(min(df_filtered['start_frame'].astype(int)),
        max(df_filtered['end_frame'].astype(int))):
        frame_df=df_filtered[(df_filtered['start_frame'].astype(int) <= i) &

```

```

(df_filtered['end_frame'].astype(int) >= i)]
res=frame_df[(frame_df['timestamp'].astype(int) == -1) |
(df_filtered['timestamp'].astype(int) == i)]
res=res.drop(columns=['start_frame','end_frame','timestamp'])
output_json=os.path.join(data_seq_path,'%010d.json'% i)
res.to_json(output_json,orient='records')

CSUPPORT = True
# Check if C-Support is available for better performance
if CSUPPORT:
    try:
        import curlVelodyneData
    except:
        CSUPPORT = False
    print('CSUPPORT is required for unwrapping the velodyne data!')
    print('Run ``CYTHONIZE_EVAL= python setup.py build_ext --inplace``
        to build with cython')
    sys.exit(-1)

class Camera(object):
    def __init__(self, root_dir, seq='2013_05_28_drive_0000_sync', cam_id=0):    #9
        calib_dir = os.path.join(root_dir, 'calibration')
        self.intrinsic_file = os.path.join(calib_dir, 'perspective.txt')
        fileCameraToPose = os.path.join(calib_dir, 'calib_cam_to_pose.txt')
        self.camToPose = loadCalibrationCameraToPose(fileCameraToPose)['image_%02d' %
            cam_id]
        self.cam_id = cam_id
        self.sequence=seq
        # load intrinsics
        self.load_intrinsics(self.intrinsic_file)

    def loadOxtsDataSingle(self,oxts_dir, frame):
        file_name = os.path.join(oxts_dir, 'data', '%010d.txt'%frame)
        try:

```



```

oxts = np.loadtxt(file_name)
except:
oxts = []
return oxts

def load_intrinsics(self, intrinsic_file):
''' load perspective intrinsics '''

intrinsic_loaded = False
width = -1
height = -1
with open(intrinsic_file) as f:
intrinsics = f.read().splitlines()
for line in intrinsics:
line = line.split(' ')
if line[0] == 'P_rect_%02d:' % self.cam_id:
K = [float(x) for x in line[1:]]
K = np.reshape(K, [3,4])
intrinsic_loaded = True
elif line[0] == 'R_rect_%02d:' % self.cam_id:
R_rect = np.eye(4)
R_rect[:3,:3] = np.array([float(x) for x in line[1:]]).reshape(3,3)
elif line[0] == "S_rect_%02d:" % self.cam_id:
width = int(float(line[1]))
height = int(float(line[2]))
assert(intrinsic_loaded==True)
assert(width>0 and height>0)

self.K = K
self.width, self.height = width, height
self.R_rect = R_rect

def cam2image(self, points):
ndim = points.ndim

```

```

if ndim == 2:
    points = np.expand_dims(points, 0)
    points_proj = np.matmul(self.K[:3,:3].reshape([1,3,3]), points)
    depth = points_proj[:,2,:]
    depth[depth==0] = -1e-6
    u = np.round(points_proj[:,0,:]/np.abs(depth)).astype(int)    #(np.int)
    v = np.round(points_proj[:,1,:]/np.abs(depth)).astype(int)

if ndim==2:
    u = u[0]; v=v[0]; depth=depth[0]
    return u, v, depth

#for annotation3d bbox from kitti
def cam2world(self,frameid):
    oxts_dir = os.path.join(kitti360Path, 'data_poses_oxts/data_poses',
        self.sequence, 'oxts')
    if not os.path.isdir(oxts_dir):
        raise ValueError('%s does not exist! \nPlease specify KITTI360_DATASET
            in your system path.\nPlease check if you have downloaded OXTS poses
            (data_poses_oxts.zip) and unzipped them under KITTI360_DATASET' % oxts_dir)
    oxts = self.loadOxtsDataSingle(oxts_dir,frameid)
    poses_pre = convertOxtsToPose(oxts)
    poses = postprocessPoses(poses_pre)
    pose=np.array(poses)

    cam2worldFrame=np.matmul(np.matmul(pose, self.camToPose),
        np.linalg.inv(self.R_rect))
    return cam2worldFrame

#for annotation3d bbox from kitti
def world2cam(self, points, R, T, inverse=False):
    assert (points.ndim==R.ndim)
    assert (T.ndim==R.ndim or T.ndim==(R.ndim-1))
    ndim=R.ndim

```

```

if ndim==2:
R = np.expand_dims(R, 0)
T = np.reshape(T, [1, -1, 3])
points = np.expand_dims(points, 0)
if not inverse:
points = np.matmul(R, points.transpose(0,2,1)).transpose(0,2,1) + T
else:
points = np.matmul(R.transpose(0,2,1), (points - T).transpose(0,2,1))

if ndim==2:
points = points[0]

return points

#for annotation3d bbox from kitti
def project_vertices(self, vertices, frameId, inverse=True):

# current camera pose TODO check this
curr_pose = self.cam2world(frameId)
T = curr_pose[:3, 3]
R = curr_pose[:3, :3]

# convert points from world coordinate to local coordinate
points_local = self.world2cam(vertices, R, T, inverse)

# perspective projection
u,v,depth = self.cam2image(points_local)

return (u,v), depth

#for annotation3d bbox from kitti
def __call__(self, obj3d, frameId):

vertices = obj3d.vertices

```

```

uv, depth = self.project_vertices(vertices, frameId)

obj3d.vertices_proj = uv
obj3d.vertices_depth = depth
obj3d.generateMeshes()

# the main class that loads raw 3D scans
class Kitti360Viewer3DRaw(object):

# Constructor
def __init__(self, seq=0, mode='velodyne'):

self.sensor_dir='velodyne_points'

sequence = '2013_05_28_drive_%04d_sync' % seq
self.raw3DPcdPath = os.path.join(kitti360Path, 'data_3d_raw', sequence,
    self.sensor_dir, 'data')

self.kitti360Path = kitti360Path
self.sequence = sequence
self.loadExtrinsics()

def generatePose(self,frame):
file_name = os.path.join(kitti360Path,'data_poses_oxts/data_poses', self.sequence,
    'oxts/data', '%010d.txt'%frame)
checkfile(file_name)
oxts = np.loadtxt(file_name)

# convert to Mercator coordinate
poses = convertOxtsToPose(oxts)

# convert coordinate system from
#   x=forward, y=right, z=down

```

```

# to
#   x=forward, y=left, z=up
poses = postprocessPoses(poses)
poseTrMtx=np.array(poses)   #[:-1]       #3*4
return poseTrMtx

def loadCalibrationRigid(self, filename):
# check file
checkfile(filename)

lastrow = np.array([0,0,0,1]).reshape(1,4)
return np.concatenate((np.loadtxt(filename).reshape(3,4), lastrow))

def loadExtrinsics(self):
# cam_0 to velo
fileCameraToVelo = os.path.join(self.kitti360Path, 'calibration',
    'calib_cam_to_velo.txt')
TrCam0ToVelo = self.loadCalibrationRigid(fileCameraToVelo)

# all cameras to system center
fileCameraToPose = os.path.join(self.kitti360Path, 'calibration',
    'calib_cam_to_pose.txt')
TrCamToPose = loadCalibrationCameraToPose(fileCameraToPose)

self.TrVeloToPose = TrCamToPose['image_00'] @ np.linalg.inv(TrCam0ToVelo)

# velodyne to all cameras
self.TrVeloToCam = {}
for k, v in TrCamToPose.items():
# Tr(cam_k -> velo) = Tr(cam_k -> cam_0) @ Tr(cam_0 -> velo)
TrCamkToCam0 = np.linalg.inv(TrCamToPose['image_00']) @ TrCamToPose[k]
TrCamToVelo = TrCam0ToVelo @ TrCamkToCam0
# Tr(velo -> cam_k)
self.TrVeloToCam[k] = np.linalg.inv(TrCamToVelo)

```

```

def loadVelodyneData(self, frame=0):
    pcdFile = os.path.join(self.raw3DPcdPath, '%010d.bin' % frame)
    if not os.path.isfile(pcdFile):
        raise RuntimeError('%s does not exist!' % pcdFile)
    pcd = np.fromfile(pcdFile, dtype=np.float32)
    pcd = np.reshape(pcd, [-1, 4])
    return pcd

def curlParameterFromPoses(self, frame):
    Tr_pose_pose = np.eye(4)
    a=self.generatePose(frame)

    if frame==0:
        Tr_pose_pose = np.linalg.inv(self.generatePose(frame+1)) @ a
    else:
        Tr_pose_pose = np.linalg.inv(a) @ self.generatePose(frame-1)
    Tr_delta = np.linalg.inv(self.TrVeloToPose) @ Tr_pose_pose @ self.TrVeloToPose

    r = self.Rodrigues(Tr_delta[0:3,0:3])
    t = Tr_delta[0:3,3]
    return r.flatten(),t

def curlVelodyneData(self, frame, pcd):
    pcd=pcd.astype(np.float64)
    pcd_curled = np.copy(pcd)
    # get curl parameters
    r,t = self.curlParameterFromPoses(frame)
    # unwrap points to compensate for ego motion
    pcd_curled = curlVelodyneData.cCurlVelodyneData(pcd, pcd_curled, r, t)
    return pcd_curled.astype(np.float32)

# Convert rotation matrix to axis angle
def Rodrigues(self,matrix):

```

```

"""Convert the rotation matrix into the axis-angle notation.
Conversion equations
=====
From Wikipedia (http://en.wikipedia.org/wiki/Rotation\_matrix),
the conversion is given by::
x = Qzy-Qyz
y = Qxz-Qzx
z = Qyx-Qxy
r = hypot(x,hypot(y,z))
t = Qxx+Qyy+Qzz
theta = atan2(r,t-1)
@param matrix: The 3x3 rotation matrix to update.
@type matrix: 3x3 numpy array
@return: The 3D rotation axis and angle.
@rtype: numpy 3D rank-1 array, float
"""

# Axes.
axis = np.zeros(3, np.float64)
axis[0] = matrix[2,1] - matrix[1,2]
axis[1] = matrix[0,2] - matrix[2,0]
axis[2] = matrix[1,0] - matrix[0,1]

# Angle.
r = np.hypot(axis[0], np.hypot(axis[1], axis[2]))
t = matrix[0,0] + matrix[1,1] + matrix[2,2]
theta = np.arctan2(r, t-1)

# Normalise the axis.
axis = axis / r

# Return the data.
return axis * theta

```

```

class KITTI360Bbox3D(object):
# Constructor
def __init__(self):
# the polygon as list of points
self.vertices = np.array([[ 0.5 , 0.5,  0.5 ],
[ 0.5,  0.5, -0.5 ],
[ 0.5, -0.5,  0.5 ],
[ 0.5, -0.5, -0.5 ],
[-0.5,  0.5, -0.5 ],
[-0.5,  0.5,  0.5 ],
[-0.5, -0.5, -0.5 ],
[-0.5, -0.5,  0.5 ]])
self.faces = np.array([[0, 2, 1],
[2, 3, 1],
[4, 6, 5],
[6, 7, 5],
[4, 5, 1],
[5, 0, 1],
[7, 6, 2],
[6, 3, 2],
[5, 7, 0],
[7, 2, 0],
[1, 3, 4],
[3, 6, 4]])
self.lines = [[0,5],[1,4],[2,7],[3,6],
[0,1],[1,3],[3,2],[2,0],
[4,5],[5,7],[7,6],[6,4]]

# the ID of the corresponding object
self.entityId = ''
self.label = ''
self.timestamp = -1

# projected vertices

```



```

self.vertices_proj = None
self.meshes = []

def __str__(self):
    return self.label

def generateMeshes(self):
    self.meshes = []
    if self.vertices_proj:
        for fidx in range(self.faces.shape[0]):
            self.meshes.append( [ Point(self.vertices_proj[0][int(x)],
                self.vertices_proj[1][int(x)]) for x in self.faces[fidx]] )

def transformVertices(self, transform):
    #transform = 4*4 np.array of float
    R = transform[:3,:3]
    T = transform[:3,3]
    vertices = np.matmul(R, self.vertices.transpose()).transpose() + T
    self.vertices = vertices
    self.R = R
    self.T = T

def initBbox(self, id, lbl):
    self.entityId = id
    self.label = lbl

# class KITTI360Sprite3D(object):
#     # Constructor
#     def __init__(self):
#         # the polygon as list of points
#         self.vertices = []
#         # np.array([[ 0.5 , 0.5,  0.5 ],
#         #           [ 0.5,  0.5, -0.5 ],
#         #           [ 0.5, -0.5,  0.5 ],

```

```

#         [ 0.5, -0.5, -0.5 ],
#         [-0.5,  0.5, -0.5 ],
#         [-0.5,  0.5,  0.5 ],
#         [-0.5, -0.5, -0.5 ],
#         [-0.5, -0.5,  0.5 ]])
#
# self.faces = []
#
# # np.array([[0, 2, 1],
# #         [2, 3, 1],
# #         [4, 6, 5],
# #         [6, 7, 5],
# #         [4, 5, 1],
# #         [5, 0, 1],
# #         [7, 6, 2],
# #         [6, 3, 2],
# #         [5, 7, 0],
# #         [7, 2, 0],
# #         [1, 3, 4],
# #         [3, 6, 4]])
#
# self.lines = []
#
# # [[0,5],[1,4],[2,7],[3,6],
# #
# #         [0,1],[1,3],[3,2],[2,0],
# #
# #         [4,5],[5,7],[7,6],[6,4]]
#
#
# # the ID of the corresponding object
#
# self.entityId = ''
#
# self.label = ''
#
# self.lat = 0
#
# self.lon = 0
#
# self.alt = 0
#
# # timestamp of the bbox (-1 if statis)
#
# self.timestamp = -1
#
#
# # projected vertices
#
# self.vertices_proj = None

```

```

#         self.meshes = []

#     def __str__(self):
#         return self.label

#     #in 2D
#     def generateMeshes(self):
#         self.meshes = []
#         if self.vertices_proj:
#             for fidx in range(self.faces.shape[0]):
#                 self.meshes.append( [ (self.vertices_proj[0][int(x)],
# self.vertices_proj[1][int(x)]) for x in self.faces[fidx]] )
#                 # self.meshes.append([np.array([self.vertices_proj[0][int(x)],
# self.vertices_proj[1][int(x)]]) for x in self.faces[fidx]])
#                 # self.meshes.append( [ Point(self.vertices_proj[0][int(x)],
# self.vertices_proj[1][int(x)]) for x in self.faces[fidx]] )

#     def transformVertices(self, transform):
#         #transform = 4*4 np.array of float
#         R = transform[:3,:3]
#         T = transform[:3,3]
#         # vertices=np.array(self.vertices)
#         vertices = np.matmul(R, self.vertices.transpose()).transpose() + T
#         self.vertices = vertices
#         self.R = R
#         self.T = T

#     def initBbox(self, id, lbl): #,lat,lon,alt,ts):
#         self.entityId = id
#         self.label = lbl
#         self.vertices=np.asarray(plycar.vertices)
#         self.faces=np.asarray(plycar.triangles)
#         unique_edges = set()
#         for tri in self.faces:

```

```

#         edges = [
#             tuple(sorted((tri[0], tri[1]))),
#             tuple(sorted((tri[1], tri[2]))),
#             tuple(sorted((tri[2], tri[0]))))
#         ]
#         unique_edges.update(edges) # Add to set to ensure uniqueness

#         self.lines=np.array(list(unique_edges))
#         # self.lat = lat
#         # self.lon = lon
#         # self.alt = alt
#         # self.timestamp = ts

def display_detections(image, detection,colorlist,depth,textLayer, detail=False,):
# Draw bounding boxes and labels on the image
for box in detection.bboxes:
cid=int(box.cls.tolist()[0])
if cid in YOLO_classes:
x1, y1, x2, y2 = map(int, box.xyxy[0])
roi = depth[y1:y2, x1:x2]
distance=np.mean(roi[roi != 0])
# #https://www.mrt.kit.edu/z/publ/download/2013/GeigerAl2013IJRR.pdf
# II. SENSOR SETUP 4th bullet 0.2 cm accuracy
# distance=distance*0.2
# d=roi.reshape(-1,3).T[2].reshape(len(roi),-1)
classname=YOLO_classes[cid]
color = colorlist[cid]
conf= box.conf.tolist()[0]
label = classname+": "+str(f"{conf:.2f}")+" x:"+str(f"{x1:.2f}")+" y:"
+str(f"{y1:.2f}")+" z:"+str(f"{distance:.2f}") if detail else classname
+": "+str(f"{conf:.2f}")
cv2.rectangle(image, (x1, y1), (x2, y2), color=color.tolist(), thickness=1)
# Draw rectangle
insert_text(label,x1,y1-23,textLayer,font_colour=gradient_color(conf))

```

```

def projectVeloToImage(cam_id=0, seq=0):
    from PIL import Image
    import matplotlib.pyplot as plt

    showLidar=False
    showFrameData=False
    showWireframes=False
    # showPrimitive=False
    # showBbox=True
    frame=0
    depth_limit_meters= 50      #for LIDAR visualization
    draw_max_distance=50

    alpha = 0.1
    alpha_ML = 0.1

    sub_dir = 'data_rect'
    sequence = '2013_05_28_drive_%04d_sync'%seq
    model = YOLO('yolo11n.pt')
    classnames=model.names
    colorlist= np.random.uniform(low=0,high=255,size=(len(classnames),3)).astype(int)

    image_files=len([entry for entry in os.listdir(os.path.join(kitti360Path,
        'data_2d_raw', sequence, 'image_%02d' % cam_id, sub_dir)) if entry.lower()
        .endswith('.png')])

    #synchronise timestamps
    timestampOXTS = loadTs(sequence,is_oxts=True)

    # color map for visualizing depth map
    cm = plt.get_cmap('jet')

    #INIT CLASSES

```

```

camera = Camera(kitti360Path, sequence, cam_id)
velo = Kitti360Viewer3DRaw(mode='velodyne', seq=seq)

# take the rectification into account for perspective cameras
TrVeloToRect = np.matmul(camera.R_rect, velo.TrVeloToCam['image_%02d' % cam_id])

# visualize a frame
while True:
    start= time.time()
    lat, lon, alt = getPositionSelf(seq=sequence,frame=frame)
    # load RGB image for visualization
    imagePath = os.path.join(kitti360Path, 'data_2d_raw', sequence, 'image_%02d'
        % cam_id, sub_dir, '%010d.png' % frame)
    if not os.path.isfile(imagePath):
        raise RuntimeError('Image file %s does not exist!' % imagePath)

    # Read the image
    image = cv2.imread(imagePath)
    # Read the velodyne pointcloud
    points = velo.loadVelodyneData(frame)
    # curl velodyne
    points = velo.curlVelodyneData(frame, points)    #in local coords, pose origin=0->lake

    #####
    # tx, ty = latlonToMercator(lat,lon, scale=0.6562652279433248)
    # # real pos in world coords, obsolete, used in GPS entities calculation
    # tx= tx - 615824.312818603
    # ty= ty - 4116230.2656191494
    # #####

    points[:,3] = 1

    pointsCam = np.matmul(TrVeloToRect, points.T).T
    pointsCam = pointsCam[:, :3]

```

```

# project to image space
u,v, depth= camera.cam2image(pointsCam.T)
u = u.astype(int)
v = v.astype(int)

# prepare depth map for visualization
depthMap = np.zeros((camera.height, camera.width))
depthImage = np.zeros((camera.height, camera.width, 3))
mask = np.logical_and(np.logical_and(np.logical_and(u>=0, u<camera.width), v>=0),
    v<camera.height)

# visualize points within 30 meters
mask = np.logical_and(np.logical_and(mask, depth>0), depth<depth_limit_meters)
depthMap[v[mask],u[mask]] = depth[mask]

colorImage = image / 255.
depthImage = cm(depthMap/depthMap.max())[...,:3] if (depthMap.max()!=0) else
    cm(depthMap)[...,:3]

#Detect through YOLO
detections = model.predict(image)    #must predict in raw non normalized image
detection = detections[0]

if showLidar:
    colorImage[depthMap>0] = depthImage[depthMap>0]

#load outsourced objects
oobjects=[]
oData=getOutsourcedData(seq=sequence,frame=frame)

for entity in oData['entities']:
    entity_id=entity["entity_id"]
    entity_label=entity["label"]
    entity_transform=entity["transform"]

```

```

# object = KITTI360Sprite3D() if showWireframes else KITTI360Bbox3D()
object =KITTI360Bbox3D ()
object.initBbox(entity_id,entity_label) #,entity_dynamic,ts)
object.transformVertices(entity_transform)
camera(obj3d=object,frameId=frame)
oobjects.append(object)

#generate layers
shapes = np.zeros_like(colorImage, np.uint8)
alpha_channel = np.full((shapes.shape[0], shapes.shape[1]), 0, dtype=np.uint8)
# 255 means fully opaque
# Add the alpha channel to the image
shapes = np.dstack((shapes, alpha_channel))

image_with_detections = np.zeros_like(shapes, np.float64)
textLayer = np.zeros_like(shapes, np.uint8)

# plot all objects
if showFrameData:
    oxtsTsLabel='oxtsTS: '+ timestampOXTS.iloc[frame,0].strftime('%Y-%m-%d %H:%M:%S')
    insert_text('Frame: %010d'% (frame),0,0,textLayer)
    insert_text('Lat: '+ str(lat),0,15,textLayer)
    insert_text('Lon: '+ str(lon),0,30,textLayer)
    insert_text('Alt: '+ str(alt),0,45,textLayer)
    insert_text(oxtsTsLabel,0,60,textLayer)

# oobjects.sort(key=lambda x: np.median(x.vertices_depth), reverse=False)
for obj in oobjects:
    # if obj.entityId=='5216':
    plotObject(showWireframes, shapes, obj,draw_max_distance,showFrameData,textLayer)
    display_detections(image_with_detections, detection,colorlist,depthMap,textLayer,
        showFrameData)

alpha_channelFl = np.ones((colorImage.shape[0], colorImage.shape[1]), dtype=np.float64)

```



```

# 255 means fully opaque
colorImage = np.dstack((colorImage, alpha_channelFl))
shapes=shapes/255
textLayer=textLayer/255
out = blend_layers(alpha, colorImage, shapes)
out = blend_layers(alpha_ML,out, image_with_detections)

out = blend_layers2(0.01,out, textLayer)

cv2.imshow('ShapesA', shapes[:, :,3])
# cv2.imshow('textLayer', textLayer[:, :,3])
cv2.imshow('Shapes', shapes[:, :,3])
cv2.imshow('Sequence %04d, Camera %02d' % (seq, cam_id), out)
end = time.time()
tlength = end - start
print ("total time: ",tlength," FPS: ", round(1/tlength, 2) )

key = cv2.waitKey(0) # Wait indefinitely for a key press
if key == ord('q'): # Quit on 'q' key press
    break
if key == ord('+'): # Quit on 'q' key press
    if (alpha>=0):alpha -= 0.05
else: alpha=1
if key == ord('-'): # Quit on 'q' key press
    if (alpha<=1):alpha += 0.05
else: alpha=0
elif key == ord('i'): # show info
    showFrameData = not showFrameData
elif key == ord('w'): # show info Wireframe
    showWireframes = not showWireframes
elif key == ord('l'): # show info
    showLidar = not showLidar
elif key == 81: # Left arrow key
    frame = (frame - 1) % image_files

```

```

elif key == 83: # Right arrow key
    frame = (frame + 1) % image_files
    cv2.destroyAllWindows()

def blend_layers(alpha, image1, image2):
    out = image1.copy()

    mask=image2[:, :, :].astype(bool)

    out[mask] = cv2.addWeighted(image1, alpha, image2, 1 - alpha, 0)[mask]
    return out

def blend_layers2(alpha, image1, image2):
    out = image1.copy()

    mask=image2[:, :, 3].astype(bool)

    out[mask] = cv2.addWeighted(image1, alpha, image2, 1 - alpha, 0)[mask]
    return out

def plotObject(showWireframes, shapes, obj, draw_max_distance, showFrameData, textLayer):
    maxH, maxW= np.shape(shapes[:, :, -1])

    avgDepth= np.mean(obj.vertices_depth)
    positive_points = [(x, y) for x, y , z in zip(obj.vertices_proj[0],
        obj.vertices_proj[1], obj.vertices_depth) if z >= 0] #x > 0 and y > 0
    visible_points =[point for point in positive_points if 0<=point[0]<=maxW and
        0<=point[1]<=maxH ]
    if visible_points and avgDepth<=maxDrawDistance:
        if showWireframes:
            for a,b in obj.lines:
                a1=obj.vertices_proj[0][a].astype(int)
                a2=obj.vertices_proj[1][a].astype(int)
                depth1=obj.vertices_depth[a]

```

```

pt1 = (a1,a2)
b1=obj.vertices_proj[0][b].astype(int)
b2=obj.vertices_proj[1][b].astype(int)
depth2=obj.vertices_depth[b]
pt2 = (b1,b2)
if (depth1>=0) and (depth2>=0):
if (a1<=maxW and a2<=maxH) or (b1<=maxW and b2<=maxH):
cv2.line(shapes, pt1,pt2, (255, 0, 0), 2)
else:
# https://stackoverflow.com/questions/22093174/copying-non-rectangular-roi-opencv
anglePoint=None
anglePoints=[]

for x,y in ((1,4),(3,6)):
if (obj.vertices_depth[x]>=0) and (obj.vertices_depth[y]>=0):
anglePoints.append((obj.vertices_proj[0][x]-obj.vertices_proj[0][y],
obj.vertices_proj[1][x]-obj.vertices_proj[1][y]))
if anglePoints:
anglePoint=np.mean(anglePoints, axis=0)

if anglePoint is None:      #no all front OR all back points detected
refPoint=np.mean(visible_points,axis=0)
if (obj.vertices_depth[1]<0) or (obj.vertices_depth[3]<0) :
#an den vlepw to mprosta
orient=Orientation.LEFT if (refPoint[0]<maxW/2) else Orientation.RIGHT
else:
orient=Orientation.RIGHT if (refPoint[0]<maxW/2) else Orientation.LEFT
else:
orient=get_orientation(anglePoint)

mask=np.zeros_like(shapes[:, :, 1])
alpha= np.round(255*avgDepth/draw_max_distance).astype(np.uint8)

for mesh in obj.meshes:

```

```

a, b, c =mesh
if (a in positive_points and b in positive_points and c in positive_points):
triangle_cnt = np.array( [a, b, c] )
cv2.drawContours(mask, [triangle_cnt], 0,255, -1)

x,y,w,h = cv2.boundingRect(mask)
if showFrameData:
cv2.rectangle(shapes,(x,y),(x+w,y+h),(255,0,0),2)

plotImage(shapes,orient,obj.label,alpha,(x,y),(x+w,y+h))

if showFrameData:
lbl=obj.label+' '+ obj.entityId
anchor = min(visible_points, key=lambda point: math.sqrt(point[0]**2 + point[1]**2))
else:
lbl=obj.label
mean_x, mean_y = np.round(np.mean(visible_points, axis=0)).astype(int)
anchor= (mean_x,mean_y)
if anchor:
insert_text(lbl,anchor[0], anchor[1],textLayer,font_colour=(255, 0, 0))

def showPoints(image_with_detections, obj):
cv2.circle(image_with_detections,(obj.vertices_proj[0][0],
obj.vertices_proj[1][0]),3,(0,0,255),-1)      #0 red top front left
cv2.circle(image_with_detections,(obj.vertices_proj[0][1],
obj.vertices_proj[1][1]),3,(255,255,0),-1)    #1 blue bot front left
cv2.circle(image_with_detections,(obj.vertices_proj[0][2],
obj.vertices_proj[1][2]),3,(0,255,255),-1)   #2 yellow top front right
cv2.circle(image_with_detections,(obj.vertices_proj[0][3],
obj.vertices_proj[1][3]),3,(0,255,0),-1)     #3 green bot front right
cv2.circle(image_with_detections,(obj.vertices_proj[0][4],
obj.vertices_proj[1][4]),3,(255,0,255),-1)   #4 magenta bot back left
cv2.circle(image_with_detections,(obj.vertices_proj[0][5],
obj.vertices_proj[1][5]),3,(255,255,255),-1) #5 white top back left

```

```

cv2.circle(image_with_detections,(obj.vertices_proj[0][6],
    obj.vertices_proj[1][6]),3,(0,0,0),-1)        #6 black bot back right
cv2.circle(image_with_detections,(obj.vertices_proj[0][7],
    obj.vertices_proj[1][7]),3,(255,0,0),-1)      #7 blue top back right

def get_orientation(anglePoint):
    angle = (math.degrees(math.atan2(anglePoint[0], anglePoint[1])) + 360) % 360
    # Normalize to [0, 360)
    if 45 < angle <= 135:
        return Orientation.RIGHT
    elif 135 < angle <= 225:
        return Orientation.BACK
    elif 225 < angle <= 315:
        return Orientation.LEFT
    else:
        return Orientation.FRONT

def plotImage(image,orient,label,alpha, start, end):
    key= next((k for k,v in img_classes.items() if v==label),None)
    sprite=image_dict[key]['image']

    xMin, yMin = start #(0,0) #head
    xMax, yMax = end # (1,1) #tail

    if (xMin>xMax):
        xMax, xMin = xMin, xMax
    if (yMin>yMax):
        yMax, yMin = yMin, yMax

    width= xMax-xMin
    height = yMax-yMin

    if (width!=0) and (height!=0):
        if key in [2,3,5]:

```

```

#dynamic
inverse = False
target=None
if orient==Orientation.RIGHT:
    inverse=True
    target= SpriteData.SIDE
elif orient==Orientation.LEFT:
    target=SpriteData.SIDE
elif orient== Orientation.BACK:
    target=SpriteData.BACK
else:
    target=SpriteData.FRONT

yMaxSp= xMaxSp=yMinSp= xMinSp=0

if key==3:
    if target==SpriteData.BACK: target=SpriteData.FRONT
    (xMinSp, yMinSp),(xMaxSp, yMaxSp) = MOTORBIKE[target]

elif key==2:
    (xMinSp, yMinSp),(xMaxSp, yMaxSp) = CAR[target]

else:
    if target==SpriteData.BACK: target=SpriteData.FRONT
    (xMinSp, yMinSp),(xMaxSp, yMaxSp) = BUS[target]

roi_sprite=sprite[yMinSp:yMaxSp, xMinSp:xMaxSp]
scaled_image = cv2.resize(roi_sprite, (width, height), interpolation=cv2.INTER_AREA)
#cv2.INTER_LINEAR) cv2.INTER_CUBIC
roi_image=np.zeros_like(scaled_image)
roi_image = image [yMin:yMax, xMin:xMax]

if inverse:
    scaled_image= np.fliplr(scaled_image)

```

```

rHeight, rWidth, _ = np.shape(roi_image)
scaled_image=scaled_image[:rHeight ,
:rWidth]

else:
scaled_image = cv2.resize(sprite, (width, height), interpolation=cv2.INTER_AREA)
#cv2.INTER_LINEAR) cv2.INTER_CUBIC
roi_image = image [yMin:yMax, xMin:xMax]

a=scaled_image[:, :,3]/255
a=(a * alpha).astype(np.uint8)
scaled_image[:, :, 3] = a

blended_image = blend_with_alpha(scaled_image, roi_image)
image[yMin:yMax, xMin:xMax] = blended_image

def blend_with_alpha(scaled_image, roi_image):
color1 = roi_image[:, :, :3]
alpha1 = roi_image[:, :, 3] / 255.0 # Normalize alpha to [0, 1]

color2 = scaled_image[:, :, :3]
alpha2 = scaled_image[:, :, 3] / 255.0 # Normalize alpha to [0, 1]

# Compute the blended alpha channel
blended_alpha = alpha1 + alpha2 * (1 - alpha1)

# Blend the RGB channels
blended_color = (
color1 * alpha1[:, :, None] +
color2 * alpha2[:, :, None] * (1 - alpha1[:, :, None])
) / blended_alpha[:, :, None]

# Handle division by zero in fully transparent areas
blended_color[np.isnan(blended_color)] = 0

```

```

# Combine the blended color and alpha into a new image
blended_image = np.dstack((blended_color.astype(np.uint8), (blended_alpha * 255)
    .astype(np.uint8)))
return blended_image

def insert_text(msg,start_x,start_y,image, font_colour=(255,255,255)):
    msglen=len(msg)*9.5 +4  #9 pixels per char + 2 offset
    sub_img = image[start_y:start_y+14, start_x:start_x+round(msglen)]
    cv2.putText(sub_img, msg, (2, 12), fontFace=cv2.FONT_HERSHEY_PLAIN, fontScale=1,
        color=font_colour,thickness= 1)
    sub_img[:, :,3]=255
    image[start_y:start_y+14, start_x:start_x+round(msglen)] = sub_img

if __name__=='__main__':

    # generate expected broadcasted message data in JSON form
    genDataStream=False
    # sequence index
    seq = 0
    cam_id = 0

    np.random.seed(20)

    loadImages()

    if genDataStream:
        prepareOutsourcedData(seq)

    # visualize raw 3D velodyne scans in 2D
    projectVeloToImage(seq=seq, cam_id=cam_id)

```


ΔΗΜΟΣΙΕΥΣΕΙΣ ΣΥΓΓΡΑΦΕΑ

- T. Gkamas, V. Karaiskos, and S. Kontogiannis, “Performance evaluation of distributed database strategies using docker as a service for industrial iot data: Application to industry 4.0,” *Information*, vol. 13, no. 4, 2022. [Online]. Available: <https://www.mdpi.com/2078-2489/13/4/190>
- T. Gkamas, S. Kontogiannis, V. Karaiskos, C. Pikridas, and I. A. Karolos, “Proposed cloud-assisted machine learning classification process implemented on industrial systems: Application to critical events detection and industrial maintenance,” in *2022 5th World Symposium on Communication Engineering (WSCE)*, 2022, pp. 95–99.
- V. Karaiskos, N. Zinas, T. Gkamas, I. A. Karolos, C. Pikridas, N. Vrettos, V. Tsioukas, and S. Kontogiannis, “Proposed industry 4.0 maintenance framework for critical and demanding infrastructures and processes,” in *2022 7th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, 2022, pp. 1–5.
- T. Gkamas, V. Karaiskos, and S. Kontogiannis, “Evaluation of cloud databases as a service for industrial iot data,” in *Proceedings of Seventh International Congress on Information and Communication Technology*, X.-S. Yang, S. Sherratt, N. Dey, and A. Joshi, Eds. Singapore: Springer Nature Singapore, 2023, pp. 273–281.

ΣΥΝΤΟΜΟ ΒΙΟΓΡΑΦΙΚΟ

Ο κ. Καραΐσκος Βασίλειος είναι διπλωματούχος Μηχανικός Πληροφορικής του Τμήματος Μηχανικών Πληροφορικής ΤΕΙ Αθηνών, και υποψήφιος μεταπτυχιακός του Τμήματος Μηχανικών Ηλεκτρονικών Υπολογιστών και Πληροφορικής της πολυτεχνικής σχολής του Πανεπιστημίου Ιωαννίνων με ειδίκευση στα Προηγμένα Υπολογιστικά Συστήματα. Εργάζεται ως μηχανικός πληροφορικής στο τμήμα εφαρμογών μετεωρολογικής υποστήριξης της Εθνικής Μετεωρολογικής Υπηρεσίας του Δημοσίου τομέα, ενώ στο παρελθόν έχει εργαστεί ως ελεύθερος επαγγελματίας με ειδίκευση στις υπηρεσίες και συστήματα υπολογιστικής νέφους και διαδικτύου των πράγματων, κατέχοντας θέση επικεφαλής του τμήματος υπηρεσιών νέφους και ανάπτυξης λογισμικού στο Εργαστήριο Μικροϋπολογιστών Συστημάτων του τμήματος Μαθηματικών του Πανεπιστημίου Ιωαννίνων. Έχει πολυετή εμπειρία σε τεχνολογίες και συστήματα αποκεντρωμένης τηλεμετρίας, τηλε-εποπτείας και τηλεδιοίκησης Βιομηχανικών Εγκαταστάσεων, Έξυπνων Πόλεων και Διαχείρισης Αυτοκινητοδρόμων σε συνεργασία με το Πανεπιστήμιο Ιωαννίνων, την Εγνατία Οδός Α.Ε., την Tekmon I.K.E., κ.α. Έχει ενεργό ερευνητικό υπόβαθρο με δημοσιεύσεις σε περιοδικά και συνέδρια πάνω σε κατανεμημένες Βάσεις Δεδομένων MongoDB και PostgreSQL για Big Data, IoT συστήματα βασισμένα σε Raspberry Pi και Arduino, αλγορίθμους και πράκτορες Τεχνητής Νοημοσύνης, τεχνολογίες επαυξημένης προβολής Ψηφιακών Διδύμων καθώς και τεχνολογίες εικονικοποίησης υπολογιστικών συστημάτων με χρήση Docker. Επιπρόσθετα, έχει εμπειρία πάνω σε ανάπτυξη εφαρμογών πλήρους στοίβας για το διαδίκτυο, υπηρεσιών αυτοματοποίησης και διασύνδεσης συστημάτων βασισμένων σε Unix, τεχνολογιών τρισδιάστατης προβολής OpenGL και Direct3D, και υλοποίηση αλγορίθμων και πρωτοκόλλων πάνω σε νέες ασύρματες τεχνολογίες.