



ΔΙΕΘΝΕΣ
ΠΑΝΕΠΙΣΤΗΜΙΟ
ΤΗΣ ΕΛΛΑΔΟΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΕΦΑΡΜΟΣΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ ΣΥΣΤΗΜΑΤΑ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ
«ΣΧΕΔΙΑΣΜΟΣ ΚΑΙ ΥΛΟΠΟΙΗΣΗ
ΣΥΣΤΗΜΑΤΟΣ ΑΣΦΑΛΕΙΑΣ ΠΑΡΟΥΣΙΑΣ
ΠΑΙΔΙΚΟΥ ΚΑΘΙΣΜΑΤΟΣ»



Του φοιτητή
Μιχαήλ Τσιλίκα
Αρ. Μητρώου: 52318m

Επιβλέπων
Δημήτριος Παπακόστας
Καθηγητής

Ημερομηνία 30 Σεπτεμβρίου 2025

Τίτλος Δ.Ε. Σχεδιασμός και Υλοποίηση Συστήματος Ασφαλείας Παρουσίας Παιδικού

Καθίσματος

Κωδικός Δ.Ε. 24245

Όνοματεπώνυμο φοιτητή: Μιχαήλ Τσιλίκας

Όνοματεπώνυμο εισηγητή: Δημήτριος Παπακόστας

Ημερομηνία ανάληψης Δ.Ε: 07-10-2024

Ημερομηνία περάτωσης Δ.Ε: 30 Σεπτεμβρίου 2025

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της μεταπτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Μεταπτυχιακό Πρόγραμμα Σπουδών «Εφαρμοσμένα Ηλεκτρονικά Συστήματα» στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η παρούσα εργασία αποτελεί πνευματική ιδιοκτησία του φοιτητή Μιχαήλ Τσιλίκα που την εκπόνησε. Στο πλαίσιο της πολιτικής ανοικτής πρόσβασης, ο συγγραφέας/δημιουργός εκχωρεί στο Διεθνές Πανεπιστήμιο της Ελλάδος άδεια χρήσης του δικαιώματος αναπαραγωγής, δανεισμού, παρουσίασης στο κοινό και ψηφιακής διάχυσης της εργασίας διεθνώς, σε ηλεκτρονική μορφή και σε οποιοδήποτε μέσο, για διδακτικούς και ερευνητικούς σκοπούς, άνευ ανταλλάγματος. Η ανοικτή πρόσβαση στο πλήρες κείμενο της εργασίας, δεν σημαίνει καθ' οιονδήποτε τρόπο παραχώρηση δικαιωμάτων διανοητικής ιδιοκτησίας του συγγραφέα/δημιουργού, ούτε επιτρέπει την αναπαραγωγή, αναδημοσίευση, αντιγραφή, πώληση, εμπορική χρήση, διανομή, έκδοση, μεταφόρτωση (downloading), ανάρτηση (uploading), μετάφραση, τροποποίηση με οποιονδήποτε τρόπο, τμηματικά ή περιληπτικά της εργασίας, χωρίς τη ρητή προηγούμενη έγγραφη συναίνεση του συγγραφέα/δημιουργού.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραίτητα και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Πρόλογος

Η παρούσα διπλωματική εργασία εκπονήθηκε στο πλαίσιο του Προγράμματος Μεταπτυχιακών Σπουδών «Εφαρμοσμένα Ηλεκτρονικά Συστήματα» και έχει ως στόχο τον σχεδιασμό και την υλοποίηση ενός συστήματος ανίχνευσης παρουσίας παιδιού σε παιδικό κάθισμα αυτοκινήτου. Η επιλογή του θέματος βασίστηκε στη σημασία της πρόληψης περιστατικών που θέτουν σε κίνδυνο τη ζωή μικρών παιδιών λόγω αμέλειας ή στιγμιαίας απροσεξίας.

Η εργασία αυτή αποτέλεσε μια πολύτιμη εμπειρία, καθώς συνδύασε θεωρητικές γνώσεις και πρακτικές δεξιότητες σε τομείς όπως τα ενσωματωμένα συστήματα, η τεχνητή νοημοσύνη και η ανάπτυξη εφαρμογών για κινητές συσκευές. Ιδιαίτερη έμφαση δόθηκε στη δημιουργία ενός λειτουργικού πρωτοτύπου που αξιοποιεί σύγχρονα τεχνολογικά εργαλεία με προσιτό και αξιόπιστο τρόπο.

Η επιλογή του θέματος πραγματοποιήθηκε με γνώμονα τις προκλήσεις που συνεπάγεται η υλοποίηση της διπλωματικής σε επίπεδο γνώσεων διαφορετικών συστημάτων και λογισμικού, αλλά και με κριτήριο την ιδιαίτερη αξία που προσδίδει στη διασφάλιση της ασφάλειας και της ζωής των ανθρώπων, και συγκεκριμένα των παιδιών, στο πλαίσιο μιας απαιτητικής και συχνά έντονα εναλλασσόμενης καθημερινότητας για τους γονείς. Η εφαρμογή αποσκοπεί στην ανάδειξη τρόπων πρόληψης περιστατικών που ενδέχεται να αποβούν μοιραία για τη ζωή μικρών παιδιών, συμβάλλοντας παράλληλα στη διαμόρφωση μιας ασφαλέστερης και ομαλότερης καθημερινότητας. Με τον τρόπο αυτό, μειώνονται οι κίνδυνοι και οι πιθανότητες αμέλειας από τους κηδεμόνες. Πρόκειται για ένα σύστημα, το οποίο φιλοδοξεί να ανταποκριθεί σε μια υπαρκτή κοινωνική ανάγκη, αξιοποιώντας την τεχνολογία στην υπηρεσία της οικογενειακής ασφάλειας.

Περίληψη

Η παρούσα εργασία εστιάζει στον σχεδιασμό και την υλοποίηση ενός ολοκληρωμένου συστήματος ασφαλείας που ανιχνεύει την παρουσία παιδιού σε παιδικό κάθισμα αυτοκινήτου με σκοπό να αποτρέψει ενδεχόμενες καταστάσεις που μπορεί να είναι επικίνδυνες λόγω αμέλειας. Η λύση που παρουσιάζεται κάνει συνδυασμό τεχνολογιών με αισθητήρες, τεχνητή νοημοσύνη και εφαρμογή κινητής συσκευής Android.

Ως βασικό στοιχείο που κάνει δυνατή την λειτουργία του συστήματος και την ανίχνευση της παρουσίας παιδιού, είναι η κάμερα τεχνητής νοημοσύνης της Raspberry με τον αισθητήρα Sony IMX500, ο οποίος συνδέεται με το Raspberry Pi Zero 2W. Επάνω σε αυτό το υποσύστημα εκτελείται νευρωνικό δίκτυο συνελκτικής αρχιτεκτονικής, και πιο συγκεκριμένα, το MobileNetV2, το οποίο έχει εκπαιδευτεί με πραγματικές εικόνες για την ανίχνευση παρουσίας παιδιού. Το μοντέλο έχει βελτιστοποιηθεί για αποδοτική λειτουργία σε συστήματα με χαμηλή κατανάλωση ενέργειας, προκειμένου να ικανοποιεί τις απαιτήσεις του φορητού ενσωματωμένου συστήματος. Επιπλέον, μέσω της τεχνολογίας Bluetooth Low Energy (BLE), το Raspberry Pi λειτουργεί ως διακομιστής GATT και μεταδίδει σε πραγματικό χρόνο τα αποτελέσματα στην κινητή συσκευή του χρήστη.

Η εφαρμογή KidGuard, που αναπτύχθηκε με χρήση της γλώσσας Kotlin και του Jetpack Compose, παρέχει ένα φιλικό περιβάλλον χρήστη. Περιλαμβάνει λειτουργίες όπως σάρωση και σύνδεση με συσκευές BLE, παρουσίαση της κατάστασης παρουσίας του παιδιού, υπολογισμό απόστασης μέσω της ισχύος σήματος Bluetooth (RSSI) και δυνατότητα βαθμονόμησης. Σε περίπτωση κινδύνου, ενεργοποιείται ειδοποίηση με ήχο, δόνηση και εμφάνιση μηνύματος στην οθόνη. Εάν ο χρήστης δεν ανταποκριθεί, αποστέλλεται αυτόματα SMS σε προκαθορισμένους από τον χρήστη αριθμούς με στίγμα GPS, προκειμένου να διευκολύνει την άμεση αντίδραση και αντιμετώπιση της κατάστασης. Όλες οι ρυθμίσεις (αριθμοί τηλεφώνου, μήνυμα ειδοποίησης, όρια συναγερμού, γλώσσα κλπ.) αποθηκεύονται μόνιμα τοπικά στην εφαρμογή.

Η σημασία της εργασίας αναδεικνύεται μέσω της δημιουργίας ενός λειτουργικού προτύπου που χρησιμοποιεί σύγχρονες τεχνολογίες για την πρόληψη ατυχημάτων λόγω παραμονής παιδιών σε οχήματα.

«Σχεδιασμός και Υλοποίηση Συστήματος Ασφαλείας Παρουσίας Παιδικού Καθίσματος»

(Design and Implementation of a Child Seat Presence Safety System)

«Μιχαήλ Τσιλίκας»

(Michail Tsilikas)

Abstract

This thesis focuses on the design and implementation of an integrated safety system that makes it possible to detect the presence of children, more specifically can inform the user that has left the kid unattended in the car. The purpose of this is to preserve the life and wellbeing of children while being in the car. The proposed solution makes use of technologies such as AI, Android application, Sensors, and camera with AI capabilities.

One of the main components that makes it possible for the system to detect the presence of a child is the AI Raspberry Camera integrating the sensor Sony IMX500. Using that camera together with a Raspberry Pi Zero 2W and utilizing the use of neural networks architecture ,in particular the MobileNetV2, it was possible to use video capture to detect the presence of the kid. The Raspberry Pi acts as a GATT server, using the BLE wireless protocol technology and manages to introduce every new measurement and data update to the connected Android app specifically developed for this task.

The mobile app , named KidGuard, developed in Kotlin and Jetpack Compose. It provides an easy interaction interface to control, set-up and monitor the device, while at the same time acting as an alarm in case the kid left unattended in the car.

The system's functionality is simple for the user, having specific fields that can be altered to represent the user's needs. It is possible to send SMS with embedded GPS location as an emergency option if the user does not respond to an alarm.

The value of this thesis is brought to light by the implementation of a working prototype incorporating modern technologies into the field of accident prevention and children safety, showing the possibilities that so far have not been fully explored or utilized.

Ευχαριστίες

Η πορεία στον τομέα των ηλεκτρονικών δεν θα μπορούσε να είναι δυνατή χωρίς την συμβολή της οικογένειάς μου, που με την συνεχή στήριξη και υπομονή τους μου δώσαν τις βάσεις για να εξελιχθώ. Παράλληλα, σημαντικό ρόλο έπαιξε η δημιουργία δεσμών με άτομα μέσω του μεταπτυχιακού, συμφοιτητές και καθηγητές, που ήταν έναυσμα για περαιτέρω προσωπική, αλλά και ακαδημαϊκή εξέλιξη. Φυσικά, αυτή η διπλωματική δεν θα μπορούσε να επιτευχθεί χωρίς την υποστήριξη του υπεύθυνου καθηγητή μου, κ. Δημητρίου Παπακώστα, που μου έδωσε και πάλι, μετά την υλοποίηση της πτυχιακής σε προπτυχιακό επίπεδο, την δυνατότητα να συνεργαστούμε, ώστε να δομηθεί και να έρθει εις πέρας η παρούσα διπλωματική αυτή εργασία.

Η ενασχόληση του μηχανικού με το αντικείμενο των ηλεκτρονικών είναι μια συνεχής πορεία ενάντια στον χρόνο και προσπάθεια για κατάκτηση της πληροφορίας και των νέων τεχνολογιών που εισέρχονται στην καθημερινότητα με ταχύτατη ροή. Η ζωή μας είναι οι άνθρωποι και οι σχέσεις με αυτούς. Ο ακαδημαϊκός τομέας, αν και πιο αποστειρωμένος θεωρητικά, σε εμένα προσωπικά έδωσε και δίνει ανθρώπους με ιδιαίτερη αξία, είτε αυτοί είναι συνάδελφοι, είτε καθηγητές μου. Ελπίζω να συνεχίσω να γνωρίζω τέτοιους ανθρώπους και να συνεργαζόμαστε αρμονικά και με διάθεση για δημιουργία.

Περιεχόμενα

Πρόλογος.....	iv
Περίληψη	v
Abstract.....	vi
Ευχαριστίες.....	vii
Περιεχόμενα	viii
Κατάλογος Εικόνων	xii
Κατάλογος Παράθεσης Κώδικα.....	xiv
Κατάλογος Πινάκων.....	xv
Συντομογραφίες	xvi
Κεφάλαιο 1ο: Εισαγωγή	1
1.1 Εισαγωγή	1
1.2 Εντοπισμός και Ανάλυση Προβλήματος.....	1
1.3 Στόχοι και Απαιτήσεις Συστήματος.....	1
1.4 Προκλήσεις και Περιορισμοί	2
1.5 Συνεισφορά και Καινοτομία	3
1.6 Δομή Διπλωματικής Εργασίας	3
1.7 Επίλογος	4
Κεφάλαιο 2ο: Σχεδιασμός και Αρχιτεκτονική Συστήματος.....	6
2.1 Εισαγωγή	6
2.2 Ανάλυση Εναλλακτικών Προσεγγίσεων	6
2.2.1 Επεμβατικοί, στην Δομή του Παιδικού Καθίσματος, Αισθητήρες.....	7
2.2.2 Μη Επεμβατικοί, στην Δομή του Παιδικού Καθίσματος, Αισθητήρες.....	8
2.3 Συνολική Αρχιτεκτονική Συστήματος.....	10
2.4 Απαιτήσεις και Επιλογή Υλικού.....	11
2.4.1 Κάμερα Τεχνητής Νοημοσύνης.....	11
2.4.2 Κεντρική Μονάδα Ελέγχου (Μικροϋπολογιστής).....	11
2.4.3 Τροφοδοσία και Περιφερειακά	12
2.5 Σενάρια Χρήσης	13
2.5.1 Σενάριο 1 ^ο : Κανονική Χρήση και Παρακολούθηση Παρουσίας.....	13
2.5.2 Σενάριο 2 ^ο : Ανίχνευση Παρουσίας και Ειδοποιήσεις	14
2.5.3 Σενάριο 3 ^ο : Κατάσταση Έκτακτης Ανάγκης και Ειδοποίηση Τρίτων	14
2.6 Μηχανισμοί Ασφαλείας.....	14
2.6.1 Ασφάλεια Δεδομένων.....	14

2.6.2	Ασφάλεια Συστήματος	14
2.6.3	Ειδοποίηση χρήστη	15
2.7	Επίλογος	15
Κεφάλαιο 3ο: Θεωρητικό Υπόβαθρο		16
3.1	Εισαγωγή	16
3.2	Τεχνητή νοημοσύνη και Μηχανική Όραση	16
3.2.1	Βασικές Έννοιες Μηχανικής Μάθησης και Νευρωνικών Δικτύων	17
3.2.2	Τεχνητή Νοημοσύνη Παρυφής (Edge AI).....	18
3.2.3	Μηχανική Όραση.....	19
3.2.4	MobileNetV2.....	20
3.2.5	Κβαντοποίηση Μοντέλων και ONNX	20
3.3	Διαδίκτυο των Πραγμάτων και Ενσωματωμένα Συστήματα	21
3.3.1	Διαδίκτυο των Πραγμάτων	21
3.3.2	Ενσωματωμένα Συστήματα και Πρωτόκολλο BLE.....	21
3.4	Εφαρμογές Κινητών Συσκευών Android.....	22
3.4.1	Ανάπτυξη Εφαρμογών Android	22
3.4.2	Αρχιτεκτονική Εφαρμογής και Λειτουργικότητα	22
3.4.3	Διεπαφή και Εμπειρία Χρήστη	23
3.5	Σχετικές Μελέτες-Εργασίες	23
3.5.1	Συναφείς Εργασίες σε Συστήματα Ασφαλείας με Τεχνητή Νοημοσύνη Παρυφής 23	
3.6	Επίλογος	24
Κεφάλαιο 4ο: Ανάπτυξη Μοντέλου Τεχνητής Νοημοσύνης		26
4.1	Εισαγωγή	26
4.2	Παρουσίαση των Βημάτων Σχεδιαστικής Του Συστήματος	26
4.3	Δημιουργία Συνόλου Δεδομένων	28
4.4	Εκπαίδευση Και Βελτιστοποίηση Μοντέλου.....	29
4.4.1	Αρχιτεκτονική Μοντέλου	31
4.4.2	Υπερπαράμετροι Εκπαίδευσης.....	33
4.4.3	Η διαδικασία της Εκπαίδευσης και οι Εποχές Εκπαίδευσης	35
4.4.4	Δημιουργία Διαγραμμάτων, Εξαγωγή Μετρικών και Ανάλυση του Τελικού Αποτελέσματος 39	
4.5	Εξαγωγή ONNX	41
4.6	Κβαντοποίηση Και Ανάπτυξη Στον IMX500(AI Κάμερα)	43
4.7	Επίλογος	44

Κεφάλαιο 5ο: Υλοποίηση Υποσυστήματος Raspberry Pi και Διακομιστή BLE.....	46
5.1 Εισαγωγή	46
5.2 Περιγραφή του Υποσυστήματος.....	46
5.3 Αυτόματη Εκκίνηση Λειτουργιών	47
5.3.1 Δεδομένα παρουσίας παιδιού	47
5.3.2 Διαχείριση Αισθητήρα Θερμοκρασίας Υγρασίας AHT20 και Παρακολούθηση Στάθμης Μπαταρίας	48
5.3.3 Διαχείριση Υπέρυθρων LED.....	49
5.3.4 Επικοινωνία BLE.....	49
5.3.5 Αυτόματη Εκκίνηση Υπηρεσίας.....	52
5.4 Επίλογος	53
Κεφάλαιο 6ο: Ανάπτυξη Εφαρμογής Android και Ενσωμάτωση Επικοινωνίας BLE	54
6.1 Εισαγωγή	54
6.2 Σχεδιασμός Αρχιτεκτονικής Εφαρμογής Android	54
6.2.1 Επιλογή Τεχνολογιών.....	54
6.2.2 Αρχιτεκτονική MVVM (Model-View-View-Model).....	55
6.2.3 Διαχείριση Κατάστασης	55
6.3 Υλοποίηση Λειτουργιών Πελάτη BLE (BLE Client)	56
6.3.1 Σάρωση και Ανακάλυψη BLE.....	56
6.3.2 Υλοποίηση Πελάτη GATT (GATT Client)	56
6.3.3 Διαχείριση Χαρακτηριστικών και υπηρεσιών BLE	57
6.3.4 Διαχείριση Σύνδεσης και Αποσύνδεσης.....	58
6.4 Υπολογισμός Απόστασης από την Συσκευή.....	58
6.5 Σενάρια Λειτουργίας	59
6.5.1 Κανονική λειτουργία.....	59
6.5.2 Λειτουργία Συναγερμού και Ειδοποιήσεων.....	59
6.6 Διεπαφή Χρήστη και Σχεδίαση Οθονών	61
6.6.1 Αρχική - Κύρια οθόνη	61
6.6.2 Οθόνη Σύνδεσης και Οθόνη Βαθμονόμησης	62
6.6.3 Οθόνη Ρυθμίσεων	63
6.6.4 Πολυγλωσσική Υποστήριξη	64
6.6.5 Οθόνη Πληροφοριών	65
6.6.6 Κουμπί SOS	66
6.6.7 Ειδοποίηση και Αποστολή SMS.....	68
6.6.8 Αποστολή SMS	70

6.7	Διαχείριση Αδειών και Αρχική Ρύθμιση	71
6.8	Επίλογος	74
Κεφάλαιο 7ο: Υλοποίηση, Δοκιμές και Αξιολόγηση Συστήματος		75
7.1	Εισαγωγή	75
7.2	Τρισδιάστατος Σχεδιασμός και Τελικό Μοντέλο	75
7.3	Δοκιμές Λειτουργίας	80
7.4	Αξιολόγηση και Σύγκριση με Υπάρχουσες Λύσεις	83
7.5	Επίλογος	85
Κεφάλαιο 8ο: Συμπεράσματα και Προτάσεις Μελλοντικής Ανάπτυξης		87
8.1	Εισαγωγή	87
8.2	Σύνοψη και Συμπεράσματα	87
8.3	Προτάσεις Μελλοντικής Εξέλιξης	87
8.4	Επίλογος	88
ΒΙΒΛΙΟΓΡΑΦΙΑ		89
ΠΑΡΑΡΤΗΜΑ Α. : Κώδικας Raspberry Pi		93
ΠΑΡΑΡΤΗΜΑ Β. :Κώδικας Διαχωρισμού Συνόλου Δεδομένων		99

Κατάλογος Εικόνων

Εικόνα 1-1: Διάγραμμα ροής της δομής της διπλωματικής εργασίας	4
Εικόνα 2-1: Διάγραμμα μεθοδολογίας επιλογής αισθητήρα	9
Εικόνα 2-2: Διάγραμμα αρχιτεκτονικής συστήματος.....	10
Εικόνα 2-3: Raspberry AI camera	11
Εικόνα 2-4: Μικροϋπολογιστής Raspberry Pi Zero 2W	12
Εικόνα 2-5: Συστήματος αδιάλειπτης τροφοδοσίας με μπαταρία (UPS Lite V1.3).....	12
Εικόνα 2-6: Αισθητήρας θερμοκρασίας και υγρασίας Adafruit AHT20.....	13
Εικόνα 3-1: Διάγραμμα τεχνητής νοημοσύνης, Ιεραρχία και Εφαρμογές	17
Εικόνα 4-1: Διάγραμμα σχεδίασης και υλοποίησης του μοντέλου	27
Εικόνα 4-2: Οργάνωση αρχείων και κώδικα εκπαίδευσης, διαχωρισμού και διαγραμμάτων.....	30
Εικόνα 4-3: Διάγραμμα ροής δεδομένων και δομής μοντέλου KidGuard.....	33
Εικόνα 4-4: Διάγραμμα ροής του κώδικα «train_kid_classifier_withMetricsSave.py».....	38
Εικόνα 4-5: Τερματικό – Αποτελέσματα Εκπαίδευσης.....	39
Εικόνα 4-6: Καμπύλες Ακρίβειας (Accuracy Curves)	40
Εικόνα 4-7: Πίνακας Σύγχυσης (Confusion Matrix).....	40
Εικόνα 4-8: Διάγραμμα Καλύτερης Ακρίβειας Επικύρωσης (Best Validation Accuracy).....	41
Εικόνα 4-9: Καμπύλες Απώλειας (Loss Curves)	41
Εικόνα 5-1: Διάγραμμα ενοτήτων του υποσυστήματος.....	46
Εικόνα 6-1: Προσομοίωση κινητού με την αρχική οθόνη της εφαρμογής	54
Εικόνα 6-2: Αρχική οθόνη εφαρμογής KidGuard.....	62
Εικόνα 6-3: Οθόνη σύνδεσης και οθόνη βαθμονόμησης εφαρμογής KidGuard.....	63
Εικόνα 6-4: Οθόνη ρυθμίσεων εφαρμογής KidGuard.....	64
Εικόνα 6-5: οθόνη πληροφοριών εφαρμογής KidGuard.....	66
Εικόνα 6-6 : Κουμπί SOS στην αρχική οθόνη της εφαρμογής KidGuard	67
Εικόνα 6-7: Ενημερώσεις και ειδοποιήσεις εφαρμογής KidGuard, με στιγμιότυπο οθόνης από το απεσταλμένο SMS με τοποθεσία.....	68
Εικόνα 6-8: Μήνυμα για παραχώρηση άδειας τοποθεσίας και άδειας διαχείρισης ασύρματης ζεύξης Bluetooth.....	72
Εικόνα 6-9: Ενημέρωση για την άδεια SMS και παραχώρηση άδειας ειδοποιήσεων.....	73
Εικόνα 6-10: Κινήσεις για την παραχώρηση άδειας διαχείρισης SMS	73
Εικόνα 7-1: Διάγραμμα ενοτήτων του υποσυστήματος μικροϋπολογιστή και των περιφερειακών του.....	75
Εικόνα 7-2: Προεπισκόπηση μοντέλου 3D στο πρόγραμμα "onshape"	76
Εικόνα 7-3: Απεικόνιση τμημάτων από την προσομοίωση της σχεδίασης.....	76
Εικόνα 7-4: Απεικόνιση του μοντέλου στο πρόγραμμα διαχείρισης για τις ρυθμίσεις εκτύπωσης.....	77
Εικόνα 7-5: Πρώτη σχεδίαση	77
Εικόνα 7-6: Δεύτερη σχεδίαση	78
Εικόνα 7-7: τελική μορφή περιβλήματος.....	78
Εικόνα 7-8: Τα προς συναρμολόγηση τμήματα.....	79
Εικόνα 7-9: Το υποσύστημα συναρμολογημένο, χωρίς καπάκι	79
Εικόνα 7-10: Καπάκι και βραχίονας στερέωσης	80
Εικόνα 7-11: Προσαρτημένος βραχίονας στο καπάκι	80
Εικόνα 7-12: Εμπρόσθια όψη της τελικής κατασκευής.....	80

Εικόνα 7-13: Στιγμιότυπα οθόνης από την ταξινόμηση εικόνων από τις δοκιμές (με παρουσία παιδιού).....	81
Εικόνα 7-14: Στιγμιότυπα οθόνης από την ταξινόμηση εικόνων από τις δοκιμές (χωρίς παρουσία παιδιού).....	82
Εικόνα 7-15:Στιγμιότυπα οθόνης από την ταξινόμηση εικόνων από τις δοκιμές (Λανθασμένα θετική ταξινόμηση ως εικόνα με παρουσία παιδιού)	82

Κατάλογος Παράθεσης Κώδικα

Κώδικας 4-1: Τροποποίηση του τελικού ταξινομητή.....	31
Κώδικας 4-2: Κώδικας εξαγωγής ONNX	43
Κώδικας 4-3: Κώδικας κβαντοποίησης του μοντέλου ONNX σε μορφή int8 για τον αισθητήρα IMX500	44
Κώδικας 4-4: Εκτέλεση της εντολής imx500-converter για μετατροπή του εκπαιδευμένου μοντέλου σε μορφή συμβατή με τον αισθητήρα IMX500	44
Κώδικας 5-1: Συνάρτηση ensure_classifier_running για εκκίνηση του kid-classifier	47
Κώδικας 5-2: Συνάρτηση read_kid_from_json για ανίχνευση παρουσίας παιδιού	48
Κώδικας 5-3: Συνάρτηση get_sensor για αρχικοποίηση αισθητήρα AHT20	49
Κώδικας 5-4: Συνάρτηση read_battery_percent για ανάγνωση στάθμης μπαταρίας	49
Κώδικας 5-5: Κλάση KidPresenceCharacteristic για ανάγνωση παρουσίας παιδιού μέσω BLE	50
Κώδικας 5-6: Κλάση BatteryLevelCharacteristic για ανάγνωση στάθμης μπαταρίας μέσω BLE	50
Κώδικας 5-7: Κλάση TemperatureCharacteristic για ανάγνωση θερμοκρασίας μέσω BLE.....	51
Κώδικας 5-8: Κλάση HumidityCharacteristic για ανάγνωση υγρασίας μέσω BLE.....	51
Κώδικας 5-9: Συνάρτηση setup_adapter για ρυθμίσεις BLE ανάπτορα	52
Κώδικας 5-10: Αρχείο μονάδας systemd που χρησιμοποιείται για την αυτόματη εκκίνηση και διαχείριση της υπηρεσίας KidCarGuard BLE κατά την εκκίνηση του συστήματος	53
Κώδικας 6-1: Μεταβλητή και ροή κατάστασης θερμοκρασίας	55
Κώδικας 6-2: Κλάση BleScannerViewModel για διαχείριση καταστάσεων BLE	55
Κώδικας 6-3: Επανεκκίνηση παραμέτρων κατά τη σύνδεση συσκευής	56
Κώδικας 6-4: Συνάρτηση αυτόματης επανασύνδεσης σε τελευταία συσκευή	56
Κώδικας 6-5: Αντικείμενο gattCallback για διαχείριση σύνδεσης BLE	57
Κώδικας 6-6: Ορισμός UUID χαρακτηριστικών και υπηρεσίας BLE.....	57
Κώδικας 6-7: Κώδικας ανάγνωσης χαρακτηριστικών παιδιού και θερμοκρασίας	57
Κώδικας 6-8: Υπολογισμός απόστασης από το RSSI	58
Κώδικας 6-9: Διαδικασία εύρεσης μέσου όρου RSSI	59
Κώδικας 6-10: Κώδικας ενεργοποίησης συναγερμού σε περίπτωση αποσύνδεσης συσκευής ...	60
Κώδικας 6-11: Συναρτήσεις αποθήκευσης και εφαρμογής γλώσσας στην εφαρμογή	65
Κώδικας 6-12: Κώδικας ελέγχου διπλού κλικ για αποστολή SOS	67
Κώδικας 6-13: Συνάρτηση αντίστροφης μέτρησης και συναγερμού	69
Κώδικας 6-15: Απόκτηση τοποθεσίας και δημιουργία συνδέσμου google.....	70
Κώδικας 6-14: Συνάρτηση αποστολής έκτακτου SMS με τοποθεσία.....	71

Κατάλογος Πινάκων

Πίνακας 2-1: Σύγκριση αισθητήρων για ανίχνευση παρουσίας παιδιού	7
Πίνακας 4-1: Διαχωρισμένα σύνολα δεδομένων σε φακέλους.....	29
Πίνακας 4-2: Περιβάλλον Εκπαίδευσης - Τεχνικές Προδιαγραφές	35
Πίνακας 4-3: Βιβλιοθήκες που χρησιμοποιήθηκαν.....	36
Πίνακας 7-1: Σύγκριση των συστημάτων για προστασία παιδιού σε παιδικό κάθισμα αυτοκινήτου	85

Συντομογραφίες

Γενικές Συντομογραφίες

Δ.Ε. Διπλωματική Εργασία

ΔΙΠΑΕ Διεθνές Πανεπιστήμιο Ελλάδος

Π.Ε. Πτυχιακή Εργασία

NHTSA National Highway Traffic Safety Administration (Εθνική Υπηρεσία Οδικής Ασφάλειας)

ISO International Organization for Standardization (Διεθνής Οργανισμός Τυποποίησης)

IEEE Institute of Electrical and Electronics Engineers (Ινστιτούτο Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών)

GDPR General Data Protection Regulation (Γενικός Κανονισμός Προστασίας Δεδομένων)

GPS - Global Positioning System (Παγκόσμιο Σύστημα Εντοπισμού Θέσης)

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ & ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ

AI Artificial Intelligence (Τεχνητή Νοημοσύνη)

ML Machine Learning (Μηχανική Μάθηση)

DL Deep Learning (Βαθιά Μάθηση)

CNN Convolutional Neural Networks (Συνελκτικά Νευρωνικά Δίκτυα)

NN Neural Networks (Νευρωνικά Δίκτυα)

CV Computer Vision (Μηχανική Όραση)

ONNX Open Neural Network Exchange (Ανοιχτή Ανταλλαγή Νευρωνικών Δικτύων)

HARDWARE & ΑΙΣΘΗΤΗΡΕΣ

IoT Internet of Things (Διαδίκτυο των Πραγμάτων)

UPS Uninterruptible Power Supply (Τροφοδοτικό Αδιάλειπτης Λειτουργίας)

IMX500 Sony IMX500 (Αισθητήρας Εικόνας με TN)

ADC Analog-to-Digital Converter (Μετατροπέας Αναλογικού σε Ψηφιακό)

I2C Inter-Integrated Circuit (Διασύνδεση Ολοκληρωμένων Κυκλωμάτων)

SPI Serial Peripheral Interface (Σειριακή Διεπαφή Περιφερειακών)

ΕΠΙΚΟΙΝΩΝΙΕΣ & ΔΙΚΤΥΑ

BLE Bluetooth Low Energy (Bluetooth Χαμηλής Ενέργειας)

GATT Generic Attribute Profile (Γενικό Προφίλ Χαρακτηριστικών)

UUID Universally Unique Identifier (Μοναδικός Καθολικός Αναγνωριστής)

RSSI Received Signal Strength Indicator (Ένδειξη Ισχύος Ληφθέντος Σήματος)

ΛΟΓΙΣΜΙΚΟ & ΑΝΑΠΤΥΞΗ

UI	User Interface (Διεπαφή Χρήστη)
UX	User Experience (Εμπειρία Χρήστη)
SDK	Software Development Kit (Πακέτο Ανάπτυξης Λογισμικού)
IDE	Integrated Development Environment (Ολοκληρωμένο Περιβάλλον Ανάπτυξης)
MVVM	Model-View-ViewModel (Μοντέλο-Όψη-Μοντέλο Όψης)
OS - Operating System	(Λειτουργικό Σύστημα)
JSON	JavaScript Object Notation (Σημειογραφία Αντικειμένων JavaScript)
XML	eXtensible Markup Language (Επεκτάσιμη Γλώσσα Σήμανσης)

ΜΕΤΡΙΚΕΣ & ΑΠΟΔΟΣΗ

FP	False Positive (Ψευδώς Θετικό)
FN	False Negative (Ψευδώς Αρνητικό)
TP	True Positive (Πραγματικά Θετικό)
TN	True Negative (Πραγματικά Αρνητικό)
FPS	Frames Per Second (Καρέ ανά Δευτερόλεπτο)

Κεφάλαιο 1ο: Εισαγωγή

1.1 Εισαγωγή

Το θέμα της ασφάλειας των παιδιών στα οχήματα έχει αποκτήσει μεγάλη σημασία τα τελευταία χρόνια, καθώς παρατηρούνται συχνά περιστατικά παραμονής παιδιών σε αυτοκίνητα που έχουν σταθμεύσει. Η επικινδυνότητα αυτής της πράξης, ειδικά κατά τους ζεστούς μήνες, μπορεί να οδηγήσει σε θερμικό σοκ ή ακόμη και θάνατο. Γεγονός που έχει επισημανθεί ως κοινωνικό φαινόμενο με μεγαλύτερη προβολή.

Σύμφωνα με διεθνή ερευνητικά δεδομένα, πολλά παιδιά χάνουν τη ζωή τους κάθε χρόνο από αυτούς τους λόγους, με παρόμοια περιστατικά να αναφέρονται και στην Ευρώπη, υπογραμμίζοντας την ανάγκη για πρόληψη [1][2]. Οι υπάρχουσες λύσεις συχνά είναι περιορισμένες με κάποιες εξ αυτών να είναι ανεπαρκείς ή ελλιπείς σε αξιοπιστία και χρηστικότητα.

Είναι πλέον επιτακτική η ανάγκη για ένα σύστημα που θα προσφέρει άμεση, αξιόπιστη και αυτόματη ειδοποίηση στους γονείς ή κηδεμόνες. Σε αυτό το πλαίσιο, η μελέτη αυτή προτείνει την ανάπτυξη ενός συστήματος με βάση την τεχνητή νοημοσύνη και τα ενσωματωμένα συστήματα, το οποίο θα ανιχνεύει την παρουσία παιδιού στο όχημα και θα ειδοποιεί τον γονέα-κηδεμόνα εμποδίζοντας έτσι πιθανές τραγικές εξελίξεις.

1.2 Εντοπισμός και Ανάλυση Προβλήματος

Το αντικείμενο της εργασίας είναι η δημιουργία ενός συστήματος το οποίο να μπορεί με ασφάλεια να εντοπίζει παιδιά που βρίσκονται σε καθίσματα αυτοκινήτων και να ειδοποιεί αμέσως τον γονέα ή κηδεμόνα. Αντίθετα με πολλές άλλες υπάρχουσες λύσεις, οι οποίες εκμεταλλεύονται συχνά αισθητήρες που βρίσκονται μέσα στο ίδιο το κάθισμα, η εν λόγω μέθοδος επιδιώκει να μην παρεμβαίνει στη δομή ή στη λειτουργία του καθίσματος. Έτσι, αποφεύγονται πιθανές αλλαγές που θα μπορούσαν να έχουν αρνητικές επιπτώσεις στην εργονομία ή στην πιστοποίηση ασφάλειας του καθίσματος.

Το κίνητρο που κρύβεται πίσω από την επιλογή αυτής της προσέγγισης είναι διπλό: αφενός, η αύξηση της ευκολίας χρήσης και του εύρους εφαρμογής, δεδομένου ότι το σύστημα μπορεί να χρησιμοποιηθεί σε οποιοδήποτε κάθισμα χωρίς την ανάγκη για τροποποιήσεις, αφετέρου, η βελτίωση της συνολικής ασφάλειας, καθώς η ανίχνευση της παρουσίας του παιδιού γίνεται με ανεξάρτητα μέσα από τον τρόπο κατασκευής του καθίσματος.

Η χρήση τεχνολογιών τεχνητής νοημοσύνης και ενσωματωμένων συστημάτων, όπως η κάμερα IMX500 σε συνδυασμό με το Raspberry Pi Zero 2W και διάφορους βοηθητικούς αισθητήρες, επιτρέπει τη δημιουργία ενός ευέλικτου συστήματος. Αυτό, συνδυάζει ανίχνευση μέσω επεξεργασίας εικόνας με δυνατότητες άμεσης ειδοποίησης, καλύπτοντας έτσι το κρίσιμο κενό μεταξύ θεωρητικής ασφάλειας και πρακτικής εφαρμογής.

1.3 Στόχοι και Απαιτήσεις Συστήματος

Στόχος της παρούσας διπλωματικής είναι η μελέτη μιας διαφορετικής προσέγγισης ενός προβλήματος, η απόδοση λύσης βασιζόμενης σε νέα τεχνολογικά μέσα, και τελικώς η ανάπτυξη ενός ολοκληρωμένου συστήματος ανίχνευσης παρουσίας παιδιών στο αυτοκίνητο. Το σύστημα σχεδιάστηκε ώστε να λειτουργεί με αυτονομία και παροχή άμεσων ειδοποιήσεων

στον γονέα-κηδεμόνα συμβάλλοντας στην ασφάλεια των παιδιών και ουσιαστικά στην πρόληψη επικίνδυνων περιστατικών.

Οι κύριοι στόχοι που τέθηκαν είναι οι εξής:

- Αξιόπιστη ανίχνευση παρουσίας παιδιού μέσω των αισθητήρων (ai camera IMX500) και της επεξεργασίας των εικόνων με το μοντέλο μηχανικής μάθησης (συνελκτικού νευρωνικού δικτύου CNN).
- Ειδοποιήσεις κατάστασης σε πραγματικό χρόνο. Ειδοποίηση και ενημέρωση του γονέα-κηδεμόνα μέσω της εφαρμογής «KidGuard» σε κινητό τηλέφωνο.
- Δυνατότητα αποστολής μηνύματος SMS σε κρίσιμες περιπτώσεις, ώστε να διασφαλίζεται η ενημέρωση τρίτων οικείων προσώπων για να γίνει αντιμετώπιση της κατάστασης.
- Χαμηλή κατανάλωση ενέργειας και δυνατότητα παραμονής της συσκευής σε λειτουργία ακόμα και μετά την διακοπή της εξωτερικής τροφοδοσίας.
- Αυτόνομη λειτουργία χωρίς την χρήση διαδικτύου και επιπλέον σύνδεσης της συσκευής. Χρήση μόνο Bluetooth Low Energy (BLE) επικοινωνίας για την διεκπεραίωση όλων των λειτουργιών και την διεπαφή συστήματος αισθητήρων-κάμερας με την εφαρμογή-κινητή συσκευή.
- Προστασία και ασφάλεια της ιδιοτικότητας. Ενώ γίνεται χρήση νευρωνικών δικτύων και δημιουργία ενός μοντέλου αναγνώρισης τύπου τεχνητής νοημοσύνης, κανένα από τα δεδομένα εικόνας-βίντεο δεν διαμοιράζεται εκτός της συσκευής και όλη η επεξεργασία γίνεται τοπικά (Edge AI/On device Inference). Με αυτό τον τρόπο γίνεται διασφάλιση της προστασίας των προσωπικών δεδομένων του παιδιού και δεν είναι εκτεθειμένα σε πιθανή διαρροή.

Τα παραπάνω δίνουν μια πρώτη εικόνα για την συσκευή που πραγματεύεται η διπλωματική εργασία. Απώτερος σκοπός ήταν η δημιουργία μιας συσκευής που χρησιμοποιεί την τεχνολογία του σήμερα, εφαρμόζοντάς την σε ανάγκες της καθημερινότητας και δίνοντας λύσεις σε ένα αυξανόμενο πρόβλημα των τελευταίων χρόνων.

1.4 Προκλήσεις και Περιορισμοί

Η ανάπτυξη οποιουδήποτε συστήματος ακόμα και σε πειραματική διάταξη, έχει προκλήσεις και περιορισμούς, πόσο μάλλον όταν γίνεται συζήτηση για συστήματα τα οποία αναφέρονται στην ασφάλεια και υγεία ανθρώπων.

Οι τεχνολογικές προκλήσεις που παρουσιάστηκαν ήταν η ανάγκη ενοποιημένου συστήματος με ανάπτυξη προγραμμάτων και λειτουργιών σε εντελώς διαφορετικά μεταξύ τους περιβάλλοντα, όπως την δημιουργία της Android εφαρμογής και την υλοποίηση του μοντέλου συνελκτικών δικτύων δυαδικής ταξινόμησης εικόνων. Για να καταστεί εφικτή η δημιουργία και των δυο, χρειάστηκε πολλή μελέτη, έρευνα και δοκιμές.

Επιπροσθέτως, η έρευνα επί του θέματος ανέδειξε την ανάγκη για δημιουργία συστήματος χωρίς την επέμβαση στο παιδικό κάθισμα, για λόγους ασφαλείας. Έρευνες αναφέρουν τον κίνδυνο μη σωστής συμπεριφοράς ακόμα και από μικρές παραμετροποιήσεις του καθίσματος με υλικό το οποίο δεν είναι ήδη προεγκατεστημένο ή δεν έχει προβλεφθεί για εγκατάσταση [3][4].

Τέλος, ένας επιπλέον περιορισμός ήταν η χρονική διάρκεια της διπλωματικής και η περιορισμένη πληροφορία για ανάπτυξη συστημάτων τεχνολογίας τεχνητής νοημοσύνης παρυφής. Οι διαθέσιμες πληροφορίες περιορίζονταν κυρίως σε τεχνικά εγχειρίδια, ιστοσελίδες με συζήτηση περί ηλεκτρονικών όπως την κοινότητα της Raspberry, αλλά παρά τις δυνατότητες που μπορεί να προσδώσει αυτή η

τεχνολογία, οι πληροφορίες και διαδικασίες υλοποίησης της και ενσωμάτωσής της, κατέχονται κυρίως από εταιρίες και δεν είναι εύκολα προσβάσιμες.

1.5 Συνεισφορά και Καινοτομία

Με την παρούσα διπλωματική επιχειρείται να δημιουργηθεί ένα σύστημα που στοχεύει την αύξηση της ασφάλειας των παιδιών κατά τις μετακινήσεις τους με το αυτοκίνητο.

Η καινοτομία της, εστιάζεται στην προσέγγιση που γίνεται, έναντι των κοινών συστημάτων, με μη επεμβατική ανίχνευση της παρουσίας του παιδιού και με χρήση τεχνολογίας τεχνητής νοημοσύνης στις παρυφές. Με αυτό τον τρόπο αποφεύγεται η παραμετροποίηση των παιδικών καθισμάτων και διασφαλίζεται η δυνατότητα προσαρμογής και εγκατάστασης σε οποιοδήποτε παιδικό κάθισμα και αυτοκίνητο, χωρίς να επηρεαστεί η ικανότητα προστασίας που παρέχεται σύμφωνα με την πιστοποίηση κάθε παιδικού καθίσματος.

Η συνεισφορά που γίνεται είναι η πρόταση μιας λύσης σε ένα πρόβλημα και η δημιουργία μιας πρώτης προσέγγισης με ενθαρρυντικά αποτελέσματα. Με αυτό τον τρόπο διαφαίνεται πως η προσέγγιση του συστήματος αυτού, θα είχε την δυνατότητα πραγματικής χρήσης βελτιώνοντας την ασφάλεια του παιδιού.

1.6 Δομή Διπλωματικής Εργασίας

Η δομή της παρούσας διπλωματικής εργασίας ακολουθεί την εξής ροή. Αρχικά γίνεται προσδιορισμός του προβλήματος και επεξήγησή του με αναφορά και της κρισιμότητάς του. Έπειτα γίνεται η παρουσίαση της αρχιτεκτονικής και της σύγκρισης των διαφορετικών προσεγγίσεων, ως προς την υλοποίησή του συστήματος, καταλήγοντας στους λόγους που οδήγησαν στην επιλογή της συγκεκριμένης προσέγγισης. Στην συνέχεια, γίνεται ανάλυση και επεξήγηση βασικών όρων και θεωρητικού υποβάθρου, ώστε να καταστεί δυνατή η κατανόηση της εργασίας. Επόμενο βήμα που ακολουθείται, είναι η ανάλυση της ανάπτυξης του μοντέλου τεχνητής νοημοσύνης, τα βήματα εκπαίδευσης και ο τρόπος αξιολόγησής του και η ενσωμάτωση με το υπόλοιπο σύστημα. Ακολουθεί η υλοποίησή του υποσυστήματος του μικροϋπολογιστή με τους αισθητήρες, την διαχείριση ενέργειας και την ασύρματη ζεύξη BLE με το υπόλοιπο σύστημα. Ακολούθως, έγινε η υλοποίηση της εφαρμογής Android, η οποία αποτελεί το μέσο διάδρασης του χρήστη με το σύστημα. Μετά, περιγράφεται η τελική υλοποίηση και τοποθέτηση των απαραίτητων υλικών σε περίβλημα, η ολοκλήρωση της υλοποίησης και η έναρξη των δοκιμών, ώστε να αξιολογηθεί η τελική αξιοπιστία και λειτουργία του συστήματος. Τέλος, η ολοκλήρωση της εργασίας γίνεται με την σύγκριση με υπάρχοντα εμπορικά συστήματα και στην συνέχεια την παράθεση των τελικών συμπερασμάτων και τις προτάσεις για μελλοντική εξέλιξη και βελτίωση.

Διάγραμμα Λογικής Ροής Διπλωματικής Εργασίας



Εικόνα 1-1: Διάγραμμα ροής της δομής της διπλωματικής εργασίας

1.7 Επίλογος

Συνοψίζοντας, στο πρώτο κεφάλαιο έγινε η παρουσίαση της σημασίας του προβλήματος που επιχειρεί να αντιμετωπίσει η παρούσα διπλωματική εργασία, αναλύθηκαν οι στόχοι και οι απαιτήσεις του συστήματος και αναδείχθηκαν οι βασικές προκλήσεις και περιορισμοί που συναντώνται στην ανάπτυξη αντίστοιχων λύσεων. Παράλληλα, τονίστηκε η καινοτομία της προτεινόμενης προσέγγισης, η οποία επιτυγχάνει μη επεμβατική ανίχνευση παρουσίας παιδιού με χρήση της τεχνολογίας τεχνητής

νοημοσύνης στις παρυφές, διασφαλίζοντας τόσο την προστασία του παιδιού, όσο και την ιδιωτικότητα του, ενώ παράλληλα γίνεται διατήρηση της πιστοποίησης του παιδικού καθίσματος λόγω της μη επέμβασης σε αυτό.

Το επόμενο κεφάλαιο επικεντρώνεται στον σχεδιασμό και την αρχιτεκτονική του συστήματος. Παρουσιάζονται οι τεχνολογίες που εξετάστηκαν για τον τρόπο εντοπισμού παρουσίας παιδιού, οι περιορισμοί που προέκυψαν από το περιβάλλον χρήσης και τελικά οι λόγοι που οδήγησαν στην επιλεγμένη αρχιτεκτονική. Ειδικότερα, θα περιγράψουν τα τρία κύρια υποσυστήματα – η εφαρμογή Android, το υποσύστημα ελέγχου με το Raspberry Pi Zero 2W και το υποσύστημα τεχνητής νοημοσύνης με την κάμερα Sony IMX500 – καθώς και ο τρόπος με τον οποίο συνεργάζονται για την επίτευξη του τελικού στόχου.

Κεφάλαιο 2ο: Σχεδιασμός και Αρχιτεκτονική Συστήματος

2.1 Εισαγωγή

Για τον σχεδιασμό ενός συστήματος ασφαλείας ανίχνευσης παρουσίας για παιδικό κάθισμα αυτοκινήτου κρίνεται απαραίτητο να μελετηθεί το ίδιο το περιβάλλον χρήσης, οι περιορισμοί και η κρισιμότητα της αξιοπιστίας του. Ως αποτέλεσμα αυτών, έγινε μελέτη των τεχνολογιών και των μέσων που μπορούν να αποδώσουν στην συσκευή τα κατάλληλα χαρακτηριστικά για να μπορεί να ανταπεξέλθει στις απαιτήσεις αυτές. Στο κεφάλαιο αυτό θα αναλυθούν οι εναλλακτικές προσεγγίσεις που εξετάστηκαν και οι λόγοι που οδήγησαν στην επιλογή της τελικής προσέγγισης λόγω των τεχνικών απαιτήσεων και των περιορισμών οι οποίοι εντοπίστηκαν.

Η κεντρική φιλοσοφία πίσω από τον σχεδιασμό του συστήματος είναι η ανάγκη για όσο γίνεται μικρότερη παρέμβαση στην λειτουργικότητα των χώρων του αυτοκινήτου, στην ίδια την κατασκευή του παιδικού καθίσματος, καθώς και στην διασφάλιση διατήρησης των προδιαγραφών ασφαλείας αυτού. Παράλληλα δόθηκε έμφαση στην διασφάλιση της προστασίας των δεδομένων του παιδιού και της ιδιοτικότητας με την προσέγγιση “privacy by design”, δηλαδή το σύστημα είναι σχεδιασμένο με βάση την προστασία της ιδιοτικότητας και του μη διαμοιρασμού των δεδομένων ή έκθεσης αυτών στο διαδίκτυο καθιστώντας τα ευάλωτα.

Το προτεινόμενο σύστημα αποτελείται από τρία κύρια υποσυστήματα:

- Την εφαρμογή (Android application) διασύνδεσης, διαχείρισης και επικοινωνίας με τον τελικό χρήστη.
- Το υποσύστημα ελέγχου, συλλογής και αποστολής των δεδομένων το οποίο βασίστηκε στον μικροϋπολογιστή Raspberry Pi zero 2W.
- Το υποσύστημα τεχνητής νοημοσύνης, υπεύθυνο για την εικονοταξινόμηση (image classification), το οποίο βασίζεται στον αισθητήρα, της Sony, IMX500 και είναι κάμερα με δυνατότητες επεξεργασίας και λειτουργίας μοντέλων τεχνητής νοημοσύνης.

Με την επιλογή αυτής της αρχιτεκτονικής συστήματος κατέστη δυνατό να επιμεριστεί ο φόρτος επεξεργασίας και λειτουργίας των υποσυστημάτων και να προσαρμοστεί πιο εύκολα το τελικό σύστημα στις απαιτήσεις της προτεινόμενης λύσης-εφαρμογής.

2.2 Ανάλυση Εναλλακτικών Προσεγγίσεων

Αρχική προσέγγιση του προβλήματος ήταν να βρεθούν τρόποι ανίχνευσης της παρουσίας του παιδιού στο παιδικό κάθισμα. Είναι σημαντικό να αναφερθεί ότι στην προκειμένη περίπτωση ως παιδί αναφερόμαστε σε ηλικίες από νεογέννητα βρέφη έως και παιδιά 12 ετών (συγκεκριμένα ισχύουν όρια ύψους και βάρους που συνήθως αντιστοιχούν σε παιδιά ηλικίας περίπου 12 ετών), τα οποία βάσει του ισχύοντος κώδικα οδικής κυκλοφορίας στην Ελλάδα και στην Ευρώπη πρέπει να επιβιβάζονται σε ειδικό κάθισμα σε Ι.Χ. οχήματα [5]. Στην συνέχεια θα αναφερθούν οι εναλλακτικές προσεγγίσεις και οι λόγοι που τελικά έγινε η επιλογή της συγκεκριμένης υλοποίησης.

Οι τεχνολογίες που θα μπορούσαν να χρησιμοποιηθούν είναι οι εξής:

1. Επεμβατικοί, στην δομή του παιδικού καθίσματος, αισθητήρες
 - Μετρητές παραμόρφωσης.
 - Αντιστάσεις ανίχνευσης πίεσης (Force sensing resistors).
 - Πιεζοηλεκτρικά στοιχεία.

- Χωρητικά στοιχεία.
- 2. Μη επεμβατικοί, στην δομή του παιδικού καθίσματος, αισθητήρες
 - Αισθητήρες υπερήχων.
 - Αισθητήρας παρουσίας μικροκυμάτων (mmWave human presence sensors).
 - Σύστημα μηχανικής όρασης(κάμερα και τεχνητή νοημοσύνη).
 - LiDaR (Light Detection and Ranging)

Πίνακας 2-1: Σύγκριση αισθητήρων για ανίχνευση παρουσίας παιδιού

Κριτήριο	ΜΗ ΕΠΕΜΒΑΤΙΚΟΙ ΑΙΣΘΗΤΗΡΕΣ				ΕΠΕΜΒΑΤΙΚΟΙ ΑΙΣΘΗΤΗΡΕΣ			
	Κάμερα + AI	mmWave	Lidar	Υπέρηχοι	Μετρητές Παραμόρφωσης	Αντιστάσεις Πίεσης	Πιεζοηλεκτρικά	Χωρητικά
Εμβέλεια	Μέτρια	Μέτρια	Μέτρια	Μικρή	Επαφή	Επαφή	Επαφή	Πολύ Μικρή
Ακρίβεια Ανίχνευσης	Μέτρια	Υψηλή	Υψηλή	Υψηλή	Υψηλή	Υψηλή	Υψηλή	Καλή
Ακρίβεια Γωνίας	Υψηλή	Καλή	Υψηλή	Καμία	Καμία	Καμία	Καμία	Καμία
Αντοχή σε Θερμοκρασία	Καλή	Καλή	Καλή	Μέτρια	Καλή	Μέτρια	Εξαιρετική	Καλή
Αντοχή σε Σκοτάδι/Φως	Μέτρια	Υψηλή	Υψηλή	Υψηλή	Υψηλή	Υψηλή	Υψηλή	Υψηλή
Διάκριση Ηλικίας	Εξαιρετική	Καμία	Καμία	Καμία	Καμία	Καμία	Καμία	Καμία
Ανίχνευση Έμβιων Όντων	Εξαιρετική	Πολύ Καλή	Χαμηλή	Χαμηλή	Καμία	Καμία	Καμία	Καμία
Ανίχνευση Κατάστασης Παιδιού	Εξαιρετική	Μέτρια	Καμία	Καμία	Καμία	Καμία	Καμία	Καμία
Ανίχνευση Στατικής Παρουσίας	Εξαιρετική	Πολύ Καλή	Εξαιρετική	Εξαιρετική	Εξαιρετική	Εξαιρετική	Εξαιρετική	Εξαιρετική
Δυσκολία Ενσωμάτωσης	Μέτρια	Μέτρια	Υψηλή	Μικρή	Μέτρια	Μικρή	Μέτρια	Μικρή
Επίδραση στην Άνεση	Καμία	Καμία	Καμία	Καμία	Υψηλή	Μέτρια	Υψηλή	Χαμηλή
Παράγοντας Κόστους	Υψηλό	Μέτριο	Εξαιρετικά Υψηλό	Μικρό	Μέτριο	Μικρό	Μέτριο	Μικρό
Αξιοπιστία	Πολύ Καλή	Υψηλή	Πολύ Καλή	Καλή	Υψηλή	Καλή	Υψηλή	Καλή

2.2.1 Επεμβατικοί, στην Δομή του Παιδικού Καθίσματος, Αισθητήρες

Ως πρώτη πιθανή προσέγγιση ερευνήθηκε η πιθανή χρήση κάποιου αισθητήρα ο οποίος θα εγκαθίσταντο στο παιδικό κάθισμα είτε στην ζώνη είτε στο σημείο που κάθεται το παιδί. Οι αισθητήρες στην κατηγορία 1 έχουν ως προϋπόθεση της λειτουργίας τους την επέμβαση στην δομή ή και λειτουργία του παιδικού καθίσματος ώστε να μπορεί να γίνει αξιόπιστη ανίχνευση παρουσίας. Όπως ήδη αναφέρθηκε στο πρώτο κεφάλαιο, για την διασφάλιση της σωστής συμπεριφοράς του παιδικού καθίσματος σε περίπτωση ατυχήματος και βάση των προτύπων ασφαλείας και πιστοποιήσεων, δεν πρέπει να γίνεται καμία παρέμβαση στην δομή τους. Επομένως, η συγκεκριμένη προσέγγιση με χρήση

αισθητήρων αυτού του τύπου, αν και μελετήθηκε βιβλιογραφικά και παρουσιάζει αρκετά κατάλληλα χαρακτηριστικά, ήταν αποτρεπτική και τελικά απορρίφθηκε.

2.2.2 Μη Επεμβατικοί, στην Δομή του Παιδικού Καθίσματος, Αισθητήρες

Στην συνέχεια μελετήθηκε η χρήση μη επεμβατικών αισθητήρων της κατηγορίας 2, οι οποίοι δεν χρειάζονται παρέμβαση στο παιδικό κάθισμα και λειτουργούν έχοντας απόσταση από αυτό. Στην κατηγορία αυτή, οι 4 αισθητήρες που αναφέρονται έχουν χαρακτηριστικά πλεονεκτήματα και πληρούν το κριτήριο της μη επέμβασης στο παιδικό κάθισμα.

2.2.2.1 Αισθητήρες Υπερήχων

Οι αισθητήρες υπερήχων λειτουργούν μετρώντας τον χρόνο που κάνει το σήμα υπερήχων (ηχητικό κύμα υψηλών συχνοτήτων) να φτάσει σε κάποιο αντικείμενο-επιφάνεια και να επιστρέψει στον ανιχνευτή, μετά από ανάκλαση σε κάποιο σημείο της πορείας του. Ενώ έχουν ικανοποιητική ακρίβεια, μικρό μέγεθος, ευκολία χρήσης και δεν απαιτούν φυσική επαφή, είναι επιρρεπείς σε μη ακριβείς μετρήσεις λόγω εξωτερικών παραγόντων θερμοκρασίας, υγρασίας, καθώς επίσης, δεν κάνουν και διάκριση έμβιων όντων και άβιων πραγμάτων. Αυτοί οι λόγοι οδήγησαν στην απόρριψη και αυτής της λύσης.

2.2.2.2 Αισθητήρας Παρουσίας Μικροκυμάτων

Οι αισθητήρες παρουσίας μικροκυμάτων λειτουργούν με παρόμοιο τρόπο λειτουργίας με τους αισθητήρες υπερήχων, με την βασική διαφορά ότι χρησιμοποιούν ηλεκτρομαγνητικά κύματα πολύ υψηλής συχνότητας αντί για ηχητικά κύματα. Η αρχή λειτουργίας τους βασίζεται στην μεταβολή της αρχικής συχνότητας του κύματος όταν αυτό προσπίπτει σε κάποια επιφάνεια-αντικείμενο. Αυτό τους καθιστά ικανούς, όχι μόνο να ανιχνεύσουν απόσταση, αλλά και κίνηση με εξαιρετική λεπτομέρεια, άρα και παρουσία. Προσφέρουν πλεονεκτήματα όπως χαμηλή κατανάλωση, μικρό μέγεθος, χαμηλό κόστος και ανθεκτικότητα σε εξωτερικές συνθήκες. Ωστόσο, ως αισθητήρες ενώ μπορούν να κάνουν διάκριση ανάμεσα σε έμβια όντα και άβια αντικείμενα, δεν είναι εύκολος ο διαχωρισμός των έμβιων όντων, δεν μπορεί να γίνει εύκολα διάκριση μεταξύ ενός οικόσιτου ζώου και ενός μικρού παιδιού. Στην περίπτωση που γινόταν χρήση αυτών των αισθητήρων, θα ήταν πιθανή η χρήση και δημιουργία μοντέλου τεχνητής νοημοσύνης για την εκπαίδευση αναγνώρισης και διαχωρισμού μεταξύ των αναγνωριζόμενων έμβιων οργανισμών, ενώ θα ήταν και απαραίτητη η προσεκτική βαθμονόμηση για την ορθή λειτουργία σε έναν τόσο μικρό χώρο όπως η καμπίνα του αυτοκινήτου, με πολλές περιοχές ανακλάσεων. Για τους λόγους αυτούς απορρίφθηκε η συγκεκριμένη προσέγγιση.

2.2.2.3 Αισθητήρες LiDaR

Οι αισθητήρες LiDaR προσφέρουν λεπτομερή αποτύπωση τρισδιάστατου χώρου μέσω εκπομπής παλμών λέιζερ και μέτρησης του χρόνου που χρειάζονται για να επιστρέψουν οι παλμοί στον αισθητήρα. Είναι αισθητήρες με αυξανόμενη τάση χρήσης σε εμπορικά προϊόντα και ενσωματώνονται σε συστήματα όπως αυτόνομα αυτοκίνητα και κινητά για ξεκλείδωμα με ανίχνευση προσώπου, αλλά το υψηλό κόστος τους καθώς και η πολυπλοκότητα ενσωμάτωσης και η ανάγκη για σημαντική επεξεργαστική ισχύ, ήταν παράγοντες που το καθιστούν την χρήση τους αποτρεπτική με τα τεχνολογικά δεδομένα του σήμερα.

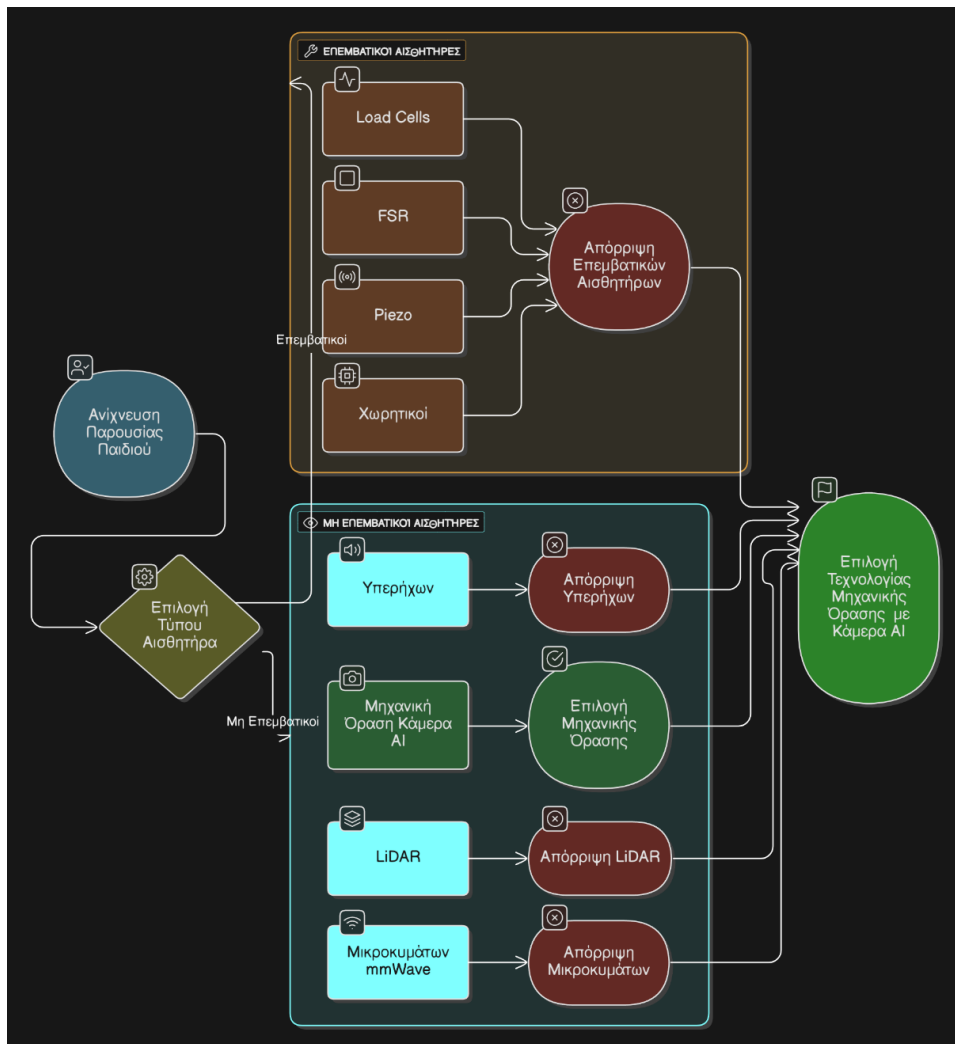
2.2.2.4 Σύστημα Μηχανικής Όρασης

Τα συστήματα μηχανικής όρασης αποτελούνται από το σύστημα επεξεργασίας, τον αλγόριθμο επεξεργασίας εικόνας και τεχνητής νοημοσύνης και κάμερες για να συλλάβουν την προς ανάλυση

εικόνα ή βίντεο. Αν και έχουν πολυπλοκότητα και απαιτούν υπολογιστική ισχύ, είναι πλήρως διαβαθμιζόμενα και έχουν δυνατότητα προσαρμογής απόλυτα στις ανάγκες κάθε λύσης. Μπορούν να προσαρμοστούν στο τι πρέπει να ανιχνεύσουν, αν θα κατηγοριοποιήσουν μια εικόνα αν θα ανιχνεύσουν κάτι μέσα στην εικόνα, μπορούν να κάνουν διάκριση μεταξύ ενήλικα και παιδιού. Έχουν επίσης δυνατότητες μετεξέλιξης και αναβάθμισης σύμφωνα με τις εκάστοτε ανάγκες. Τα παραπάνω σε συνδυασμό με την αξιοπιστία, την ακρίβεια και τις δυνατότητες που προσφέρουν για επεκτασιμότητα οδήγησαν στην επιλογή ως σύστημα για την ανίχνευση της παρουσίας παιδιού.

2.2.2.5 Συγκριτική Αξιολόγηση και Τελική Επιλογή

Μετά την ανάλυση και την μελέτη των παραπάνω τεχνολογιών έγινε σαφές πως παρότι κάθε τεχνολογία έχει ορισμένα πλεονεκτήματα, τις απαιτήσεις της εφαρμογής δεν τις πληρούσαν οι περισσότερες. Οι επεμβατικοί, στην δομή του παιδικού καθίσματος, αισθητήρες-τεχνολογίες απορρίφθηκαν λόγω της μη συμβατότητας με την απαίτηση για διατήρηση των προδιαγραφών ασφαλείας των παιδικών καθισμάτων. Ταυτόχρονα, οι μη επεμβατικοί, στην δομή του παιδικού καθίσματος, αισθητήρες-τεχνολογίες, υπερήχων, μικροκυμάτων και LiDaR πέραν των δυνατοτήτων τους παρουσιάζουν και κάποια χαρακτηριστικά που δεν ήταν συμβατά με τον σκοπό χρήσης, όπως κόστος και ικανή ανίχνευση ανθρώπου και συγκεκριμένα παιδιού.



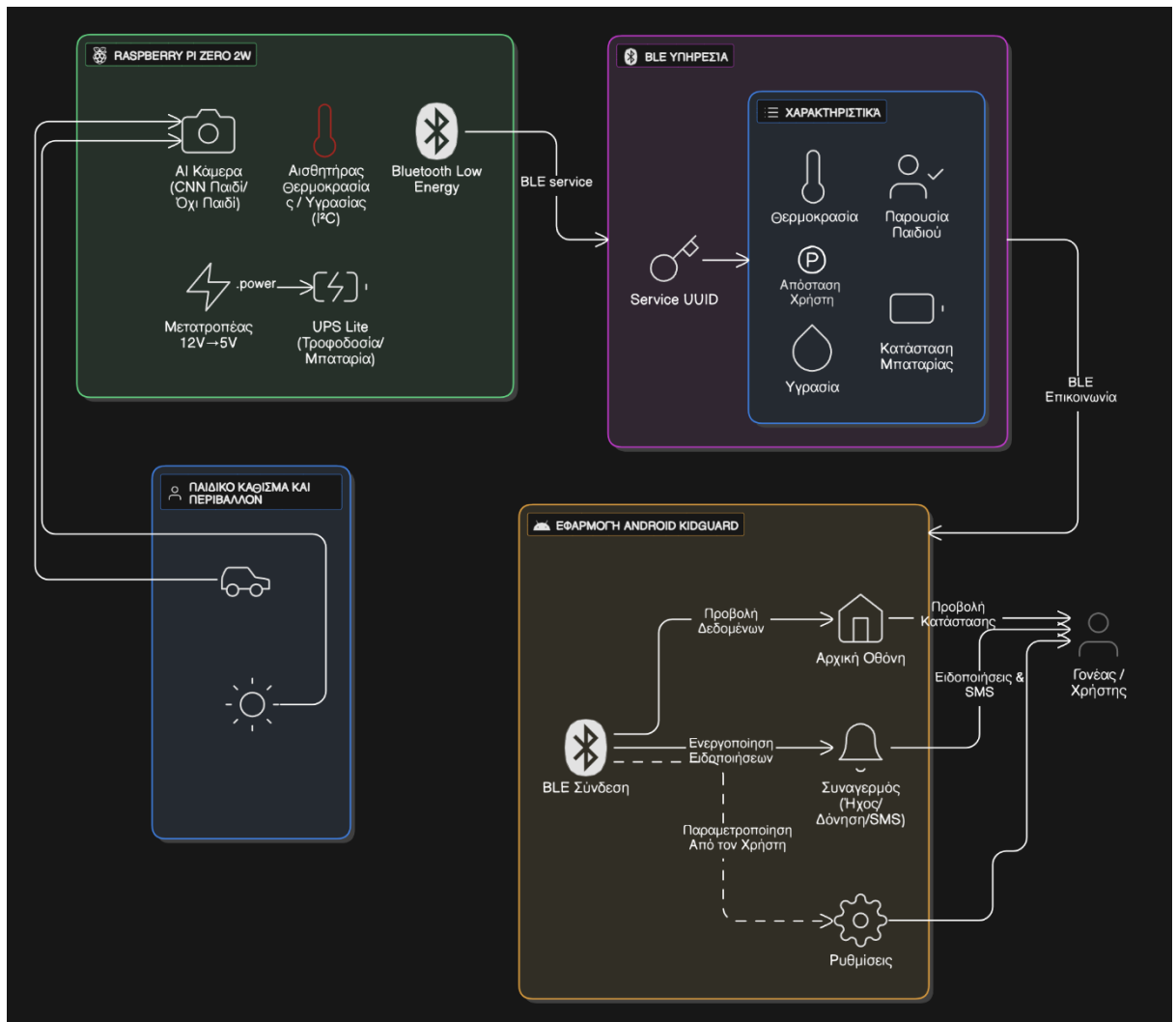
Εικόνα 2-1: Διάγραμμα μεθοδολογίας επιλογής αισθητήρα

Κεφάλαιο 2ο:

Αντιθέτως, η επιλογή της μηχανικής όρασης προσφέρει τα απαραίτητα χαρακτηριστικά, προσαρμοστικότητα στις ανάγκες και ήταν και μια πρόκληση για νέες γνώσεις και ανάπτυξη, απαραίτητο για κάθε μηχανικό που προσπαθεί να συντονίζεται με την πρόοδο και την συνεχή εξέλιξη σε υπάρχοντες και νέους τομείς της τεχνολογίας.

2.3 Συνολική Αρχιτεκτονική Συστήματος

Το σύστημα σχεδιάστηκε εμπεριέχοντας ξεχωριστά υποσυστήματα υπεύθυνα για συγκεκριμένες λειτουργίες που τελικώς επικοινωνούν μεταξύ τους προσφέροντας το επιθυμητό αποτέλεσμα με τον βέλτιστο δυνατό τρόπο. Αυτή η δομή επιτρέπει την αντιμετώπιση κάθε υποσυστήματος ξεχωριστά, την υλοποίησή του, την βελτιστοποίησή του και τέλος την σύνθεση όλων σε ένα τελικό σύστημα. Επιπροσθέτως, με αυτόν τον τρόπο έγινε διαμοιρασμός του υπολογιστικού φόρτου του κάθε συστήματος για πιο γρήγορη και αξιόπιστη λειτουργία.



Εικόνα 2-2: Διάγραμμα αρχιτεκτονικής συστήματος

Το πρώτο υποσύστημα, αυτό της κάμερας σε συνδυασμό με τον αλγόριθμο συνελκτικών νευρωνικών δικτύων, είναι υπεύθυνο για την οπτική αναγνώριση κάθε εισερχόμενης εικόνας και απόδοση της ετικέτας που αντιστοιχεί στην κατάσταση που αποτυπώνεται σε αυτήν, δηλαδή την

παρουσία ή απουσία παιδιού. Η διαδικασία αυτή γίνεται τοπικά , στον αισθητήρα και χωρίς την χρήση πόρων διαδικτύου.

Το δεύτερο υποσύστημα, ελέγχου, συλλογής και αποστολής των δεδομένων, το οποίο βασίστηκε στον μικροϋπολογιστή **Raspberry Pi zero 2W**, λειτουργεί ως ο κόμβος μεταξύ των αισθητήρων και της εφαρμογής Android. Τα δεδομένα που λαμβάνονται ως ετικέτες από την κάμερα, τα δεδομένα θερμοκρασίας και υγρασίας από τον αισθητήρα **AHT20** και τα δεδομένα στάθμης της μπαταρίας από το **UPS Lite V1.3** διαμοιράζονται μέσω της επικοινωνίας BLE και μπορεί η εφαρμογή στην συνέχεια να τα διαχειριστεί και να προχωρήσει σε ενέργειες σύμφωνα με αυτά.

Η **Εφαρμογή Android** είναι το τρίτο υποσύστημα, το οποίο είναι και το μέσο διάδρασης του χρήστη με το σύστημα. Η παραμετροποίηση των ρυθμίσεων σύμφωνα με το χρήστη γίνεται μέσω αυτής. Τα δεδομένα από τα άλλα δυο υποσυστήματα αξιοποιούνται στην εφαρμογή, οπτικοποιούνται για προβολή στον χρήστη και μέσω των συνθηκών που έχουν οριστεί η εφαρμογή ενημερώνει τον χρήστη και διασφαλίζει την προφύλαξη του παιδιού μέσα στο αυτοκίνητο.

2.4 Απαιτήσεις και Επιλογή Υλικού

2.4.1 Κάμερα Τεχνητής Νοημοσύνης

Για την κάμερα έγινε η επιλογή της **AI Raspberry camera module**, με ενσωματωμένο τον αισθητήρα **IMX500** της Sony για την δυνατότητα του να λειτουργεί το μοντέλο και να γίνεται η επεξεργασία αποκλειστικά τοπικά στον αισθητήρα.



Εικόνα 2-3: Raspberry AI camera

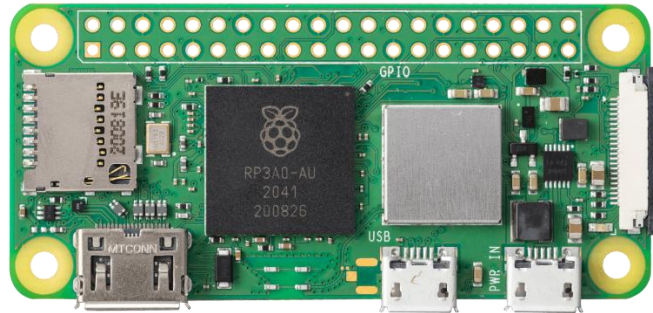
Ο συγκεκριμένος αισθητήρας είναι ο μόνος που διαθέτει μονάδα επεξεργασίας τεχνητής νοημοσύνης ενσωματωμένη στον αισθητήρα της εικόνας και παρέχει 4.2 TOPS (τρισεκατομμύρια πράξεις/ δευτερόλεπτο). Επίσης η δυνατότητα σύνδεσης με raspberry ήταν σημαντική για την ταχύτητα μετάδοσης δεδομένων και δυνατότητας μετάδοσης μεγάλου όγκου δεδομένων.

2.4.2 Κεντρική Μονάδα Ελέγχου (Μικροϋπολογιστής)

Ως κύρια μονάδα ελέγχου επιλέχθηκε ο μικροϋπολογιστής Raspberry Pi zero 2W αρχικά λόγω της κάμερας και στην συνέχεια λόγω του διαδομένου οικοσυστήματος και την δυνατότητα εύρεσης πληροφοριών και ήδη υλοποιημένων συστημάτων που μπορούν να είναι βάση για την εξέλιξη ενός προγράμματος ή μίας λειτουργίας.

Κεφάλαιο 2ο:

Επίσης παρείχε ιδανικό συνδυασμό κόστους και αποδόσεων, με αρκετή παρεχόμενη υπολογιστική ισχύ και ενσωματωμένα περιφερειακά για την υποστήριξη της επικοινωνίας BLE αλλά και της σύνδεσης WiFi για την απομακρυσμένη σύνδεση κατά την διαδικασία του προγραμματισμού και των ρυθμίσεων. Τέλος οι δυνατότητες επέκτασης και προσθήκης επιπλέον συστημάτων για την εξέλιξη της συσκευής ήταν ένας ακόμη παράγοντας για την επιλογή του.



Εικόνα 2-4: Μικροϋπολογιστής Raspberry Pi Zero 2W

2.4.3 Τροφοδοσία και Περιφερειακά

2.4.3.1 Τροφοδοσία

Η τροφοδοσία του συστήματος και πιο συγκεκριμένα του υποσυστήματος Raspberry Pi + κάμερα + περιφερειακά, γίνεται μέσω ενός συστήματος αδιάλειπτης τροφοδοσίας με μπαταρία, το UPS Lite V1.3.



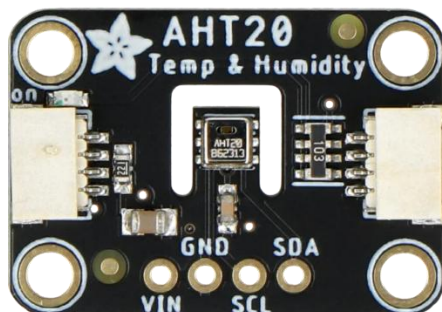
Εικόνα 2-5: Συστήματος αδιάλειπτης τροφοδοσίας με μπαταρία (UPS Lite V1.3)

Αυτό είναι υπεύθυνο για την διαχείριση ενέργειας από τον μετασχηματιστή $12V \rightarrow 5V$ ο οποίος συνδέεται στην πρίζα των 12V του αυτοκινήτου. Παράλληλα, φορτίζει την ενσωματωμένη μπαταρία και όταν γίνεται διακοπή της εξωτερικής τροφοδοσίας, παρέχει ενέργεια για ένα χρονικό διάστημα εξασφαλίζοντας την λειτουργία του συστήματος και διασφαλίζοντας πως ο γονέας-κηδεμόνας θα μπορέσει να ενημερωθεί και να αποφευχθεί τυχόν κατάσταση που θέτει σε κίνδυνο την υγεία και την ζωή του παιδιού στο παιδικό κάθισμα. Το σύστημα περιλαμβάνει μπαταρία ιόντων λιθίου (Li-ion), κύκλωμα διαχείρισης φόρτισης, και προστασία εκφόρτισης. Ταυτόχρονα κατά την λειτουργία του παρέχει πληροφορίες για την στάθμη της μπαταρίας μέσω I2C πρωτοκόλλου επικοινωνίας και μπορεί,

σε περίπτωση επαναφοράς της εξωτερικής τροφοδοσίας, να διακόψει ή να εκκινήσει την λειτουργία του μικροϋπολογιστή σύμφωνα με το πως έχει προγραμματιστεί.

2.4.3.2 Περιφερειακά

Επειδή το σύστημα στοχεύει στην προφύλαξη της υγείας παιδιών μέσα σε αυτοκίνητο και συνήθως ο κίνδυνος είναι η υψηλή ή υπερβολικά χαμηλή θερμοκρασία, κρίθηκε απαραίτητο να συμπεριληφθεί και αισθητήρας θερμοκρασίας υγρασίας.



Εικόνα 2-6: Αισθητήρας θερμοκρασίας και υγρασίας Adafruit AHT20

Για αυτήν την χρήση επιλέχθηκε ο AHT20, με αξιοπιστία για την χρήση που προορίζεται και με χαμηλό κόστος. Η επικοινωνία γίνεται μέσω I2C και κάθε μέτρηση λαμβάνεται από τον μικροϋπολογιστή και διαβάζεται από την εφαρμογή Android μέσω της επικοινωνίας BLE.

2.5 Σενάρια Χρήσης

Το παρόν σύστημα αναπτύχθηκε ώστε να ειδοποιεί έγκαιρα και αξιόπιστα τον γονέα-κηδεμόνα σε περίπτωση απομάκρυνσης του από το όχημα και παραμονής του παιδιού μέσα σε αυτό. Η βασική λειτουργία του συστήματος μπορεί να περιγράψει με τα παρακάτω σενάρια χρήσης:

2.5.1 Σενάριο 1^ο: Κανονική Χρήση και Παρακολούθηση Παρουσίας

Σε κανονική χρήση το σύστημα αναλαμβάνει την συνεχή παρακολούθηση της παρουσίας παιδιού στο παιδικό κάθισμα. Το συνελκτικό μοντέλο τρέχει στον αισθητήρα της κάμερας συνεχώς και σύμφωνα με την εικόνα που λαμβάνει με ρυθμό ανανέωσης 1-2 fps, δίνει την αντίστοιχη ετικέτα παρουσία παιδιού, όχι παρουσία παιδιού (Kid/noKid). Στην συνέχεια η ετικέτα λαμβάνεται από τον μικροϋπολογιστή και μαζί με τις τιμές θερμοκρασίας, υγρασίας, στάθμης μπαταρίας, εμφανίζονται στην εφαρμογή κινητού, η οποία είναι υπεύθυνη για τις ενέργειες που πρέπει να γίνουν σύμφωνα με τα στοιχεία που διαβάζει.

Η εφαρμογή όσο δεν διαβάζει την ετικέτα παρουσία παιδιού είναι σε κατάσταση συνεχούς παρακολούθησης, χωρίς όμως να χρειάζεται να κάνει περεταίρω δράσεις. Κατά την διάρκεια αυτή, η εφαρμογή προβάλλει όλα τα στοιχεία στην κινητή συσκευή για να υπάρχει συνεχής εποπτεία του συστήματος από τον χρήστη. Γίνεται υπολογισμός της απόστασης, αλλά δεν ενεργοποιούνται ειδοποιήσεις ή συναγερμοί, ωστόσο παρακολουθείται η ετικέτα για πιθανή αλλαγή κατάστασης της παρουσίας και όπλιση του συστήματος.

2.5.2 Σενάριο 2^ο: Ανίχνευση Παρουσίας και Ειδοποιήσεις

Στην περίπτωση που η εφαρμογή διαβάσει την ετικέτα παρουσίας και υπάρχει παιδί στο κάθισμα, πλέον ενεργοποιείται η διαδικασία ασφαλούς παρακολούθησης. Η εφαρμογή καλείται να παρακολουθεί τις οριακές τιμές που έχουν επιλεγθεί από τον χρήστη:

- **Απόσταση κινητού** (εφαρμογής) σε σχέση με τον μικροϋπολογιστή (Raspberry άρα και αυτοκινήτου, εφόσον είναι τοποθετημένο εντός αυτού).
- Θερμοκρασία καμπίνας.
- Επίπεδο στάθμης μπαταρίας.

Η σύνδεση BLE επίσης παρακολουθείται, για να διαπιστωθεί αν υπάρχει διακοπή αυτής, άρα και απουσία νέων δεδομένων στο σύστημα.

2.5.3 Σενάριο 3^ο: Κατάσταση Έκτακτης Ανάγκης και Ειδοποίηση Τρίτων

Οι ειδοποιήσεις προς τον χρήστη είναι κλιμακούμενες, από απλό μήνυμα στην οθόνη (toast) σε ήχο και δόνηση. Στην περίπτωση ειδοποίησης με ήχο και δόνηση, η μη διάδραση του χρήστη σε χρόνο 30 δευτερολέπτων οδηγεί σε αποστολή επείγοντος μηνύματος σε τρίτους.

Ειδοποιήσεις προκύπτουν εφόσον μόνο έχει ανιχνευθεί παρουσία παιδιού. Εφόσον υπάρχει παιδί στο όχημα, τότε μπορεί να προκύψει μήνυμα προειδοποίησης στην οθόνη για υψηλή θερμοκρασία. Αν η θερμοκρασία έχει φτάσει στο όριο που θεσπίστηκε ως επικίνδυνο από τον χρήστη, τότε υπάρχει ειδοποίηση ήχου-δόνησης, μη δράση του χρήστη οδηγεί στην αποστολή SMS με τοποθεσία του χρήστη στις 3 προκαθορισμένες επαφές. Η απομάκρυνση από το όχημα άνω του, θεσπισμένου από τον χρήστη, ορίου, η αποσύνδεση της εφαρμογής ή η κριτικά χαμηλή στάθμη μπαταρίας του UPS Lite V1.3 στο υποσύστημα του μικροϋπολογιστή, θα ενεργοποιήσει αντίστοιχη ειδοποίηση.

2.6 Μηχανισμοί Ασφαλείας

Το σύστημα που μελετάται, προορίζεται για χρήση παρακολούθησης και ασφαλείας, επομένως είναι απαραίτητο να διασφαλιστεί πως ότι και αν προκύψει έχει προβλεφθεί και έχει τρόπο να αντιμετωπιστεί.

2.6.1 Ασφάλεια Δεδομένων

Οι μεταδιδόμενες τιμές περιορίζονται σε αριθμητικά δεδομένα, χωρίς εικόνες, ονομασίες ή επιπλέον στοιχεία, ελαχιστοποιώντας με αυτό τον τρόπο ζητήματα ιδιοτικότητας που θα μπορούσαν να προκύψουν.

2.6.2 Ασφάλεια Συστήματος

Όλο το σύστημα έχει δομηθεί με την λογική «λειτουργία ασφαλούς αστοχίας» (fail safe), αυτό σημαίνει ότι σε περίπτωση αστοχίας να ειδοποιηθεί ο χρήστης ακόμα και αν δεν είναι απαραίτητο, αποτρέποντας έτσι καταστάσεις ψευδούς ασφάλειας προς τον χρήστη. Η εκπαίδευση του συνελκτικού μοντέλου έγινε λαμβάνοντας υπόψιν ότι είναι προτιμότερο να υπάρξουν ψευδώς θετικές ετικέτες παρουσίας, το οποίο δεν βάζει σε κίνδυνο το παιδί, αν και είναι πρόβλημα λειτουργικότητας και άνεσης για τον χρήστη. Αντίστοιχα η θερμοκρασία και η απόσταση θα μπορούσαν να μην συμπεριληφθούν, αλλά για διασφάλιση κάλυψης κάθε πιθανού σεναρίου και ως δικλείδα ασφάλειας παρακολουθούνται παράλληλα.

2.6.3 Ειδοποίηση χρήστη

Οι περιπτώσεις που ο χρήστης ξέχασε και το κινητό μαζί με το παιδί στο όχημα ή για κάποιον άλλο λόγο δεν αντιληφθεί την ηχητική + δόνηση ειδοποίηση, οδηγούν στην αποστολή ειδοποιήσεων σε τρίτους ώστε να ληφθούν τα αναγκαία μέτρα και να γίνει απομάκρυνση του παιδιού από το εσωτερικού του οχήματος.

2.7 Επίλογος

Συνοψίζοντας, στο κεφάλαιο αυτό αναλύθηκαν οι επιλογές σχετικά με τις πιθανές τεχνολογίες που θα μπορούσαν να χρησιμοποιηθούν για την ανίχνευση της παρουσίας παιδιού στο παιδικό κάθισμα στο αυτοκίνητο, γιατί απορρίφθηκαν κάποιες και τι οδήγησε στην τελική επιλογή, με την χρήση λύσης που βασίζεται στην χρήση μηχανικής όρασης. Επίσης έγινε ανάλυση της αρχιτεκτονικής του συστήματος, το πώς λειτουργεί, πώς γίνονται οι ειδοποιήσεις και πώς διασυνδέονται τα επιμέρους υποσυστήματα. Τέλος, έγινε αναφορά στην λογική της σχεδίασης με γνώμονα την ασφαλή αστοχία, ώστε σε κάθε πιθανή περίπτωση να μην τίθεται η ασφάλεια του παιδιού που είναι μέσα στο αμάξι σε κίνδυνο.

Στο επόμενο κεφάλαιο γίνεται η παρουσίαση του θεωρητικού υπόβαθρου που υποστηρίζει την υλοποίηση της προτεινόμενης λύσης. Γίνεται ανάλυση εννοιών της τεχνητής νοημοσύνης και της μηχανικής όρασης, η αρχιτεκτονική MobileNetV2 και οι τεχνικές κβαντοποίησης των υλοποιημένων μοντέλων, καθώς και οι τεχνολογίες του Διαδικτύου των Πραγμάτων (IoT) και του πρωτοκόλλου BLE. Παράλληλα, γίνεται επεξήγηση του τρόπου δομής και υλοποίησης Android εφαρμογών. Τέλος, παρατίθενται σχετικές, με το θέμα της ασφάλειας παιδιού σε παιδικό κάθισμα αυτοκινήτου, μελέτες και ερευνητικές εργασίες, έτσι παρέχεται το απαραίτητο επιστημονικό και τεχνολογικό πλαίσιο για την κατανόηση και τεκμηρίωση της προτεινόμενης λύσης.

Κεφάλαιο 3ο: Θεωρητικό Υπόβαθρο

3.1 Εισαγωγή

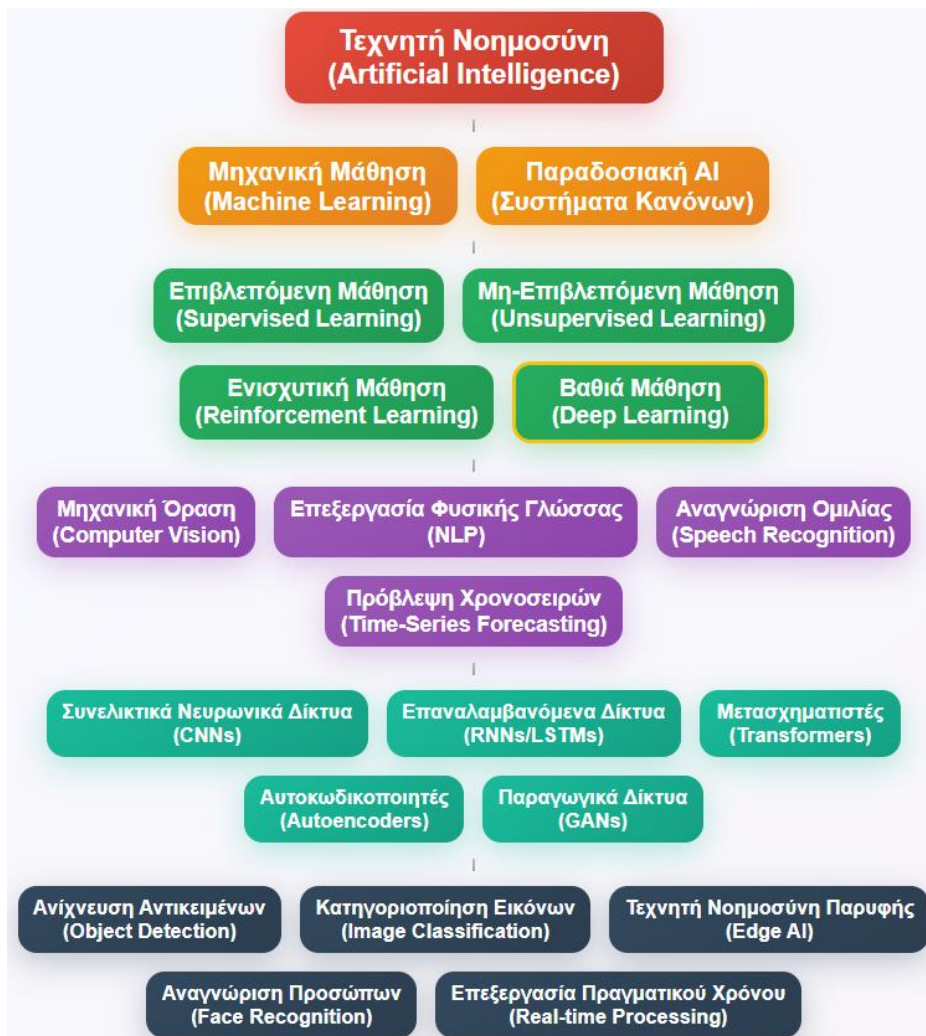
Για την κατανόηση της τεχνητής νοημοσύνης και του διαδικτύου των πραγμάτων, επομένως του θεωρητικού υπόβαθρου της παρούσας διπλωματικής εργασίας, σε αυτό το κεφάλαιο θα γίνει ανάλυση και παρουσίαση βασικών εννοιών, αρχιτεκτονικής και πρωτοκόλλων για την ανάπτυξη έξυπνων συστημάτων. Οι τεχνολογίες τεχνητής νοημοσύνης καθώς και διαδικτύου των πραγμάτων, είναι πλήρως αναπτυσσόμενες στη σημερινή εποχή, με συνεχή ανάπτυξη νέων τεχνικών, αποδοτικότερων αλγόριθμων, σμικρύνσεις του όγκου των συστημάτων και εφαρμογής τους σε κάθε τομέα της τεχνολογίας ανταποκρινόμενες στην ανάγκη για συστήματα αυτόνομα, με δυνατότητες άμεσης απόκρισης αλλά και λήψης αποφάσεων.

Η καθημερινότητα περιστρέφεται γύρω από τις φορητές συσκευές και πλέον τα κινητά τηλέφωνα είναι αναπόσπαστο κομμάτι της, οπότε προκύπτει πως μπορούν να είναι το μέσο χρήσης και διαχείρισης των συστημάτων είτε αυτά εντάσσονται στο διαδίκτυο των πραγμάτων άρα έξυπνες συσκευές, είτε είναι συστήματα τεχνητής νοημοσύνης. Για την διαχείριση αυτών μέσω των κινητών τηλεφώνων, συνήθως υπεύθυνες είναι οι εφαρμογές τύπου, Android και IOS, ανάλογα το λειτουργικό που χρησιμοποιεί η κάθε συσκευή.

Η εύρυθμη λειτουργία, επικοινωνία και συνεργασία αυτών των συστημάτων, τελικώς συνθέτουν συστήματα τεχνολογίας που απαντώνται σε λύσεις που ήδη χρησιμοποιούνται και προσφέρουν σε θέματα ευκολίας, ασφάλειας, απλοποίησης και επιτάχυνσης σε τομείς της καθημερινότητας.

3.2 Τεχνητή νοημοσύνη και Μηχανική Όραση

Ο όρος Τεχνητή Νοημοσύνη αποδίδεται με διαφορετικά νοήματα, μια περιεκτική και απλή περιγραφή του όρου είναι: «Artificial Intelligence refers to the development of systems that can perform tasks typically requiring human intelligence, such as reasoning, learning, and problem-solving.» [6] και στα Ελληνικά αποδίδεται ως : Ο όρος Τεχνητή Νοημοσύνη αναφέρεται στη δημιουργία συστημάτων που τυπικά απαιτούν ανθρώπινη ευφυΐα, όπως συλλογισμό, εκμάθηση και επίλυση προβλημάτων. Η θεμελίωση του πεδίου ξεκινάει με την δημοσίευση του Alan Turing το 1950, όπου προτάθηκε από τον ίδιο το Turing Test (Imitation Game) ως κριτήριο για την αξιολόγηση της νοημοσύνης συστημάτων [7]. Ένα παράδειγμα αυτού του τεστ είναι το CAPTCHA, ευρέως χρησιμοποιούμενο τεστ για την διάκριση ανθρώπου και μηχανής σε ιστοσελίδες και τον αποκλεισμό αυτοματοποιημένων, συνήθως κακόβουλων προγραμμάτων, από την πρόσβαση σε αυτές [8].



Εικόνα 3-1: Διάγραμμα τεχνητής νοημοσύνης, Ιεραρχία και Εφαρμογές

Η χρήση συστημάτων τεχνητής νοημοσύνης είναι πολύ διαδεδομένη και αναπτυσσόμενη διότι καλύπτει μια ευρεία κατηγορία τεχνολογιών και εφαρμογών. Εργαλεία που πλέον γίνεται η χρήση τους καθημερινά, ενσωματώνονται σε υπολογιστές, κινητά, σε συστήματα αυτοκινήτων και πολλές άλλες εφαρμογές. Όμως η ενσωμάτωση αυτής της τεχνολογίας είναι πολύ διαφορετική και ακολουθεί διαφορετικές διαδικασίες και μεθόδους, ώστε να επιτευχθεί ο τελικός στόχος και εφαρμογή.

3.2.1 Βασικές Έννοιες Μηχανικής Μάθησης και Νευρωνικών Δικτύων

Η Μηχανική Μάθηση αποτελεί μια από τις υποκατηγορίες της τεχνητής νοημοσύνης και έχει ως αντικείμενο την ανάπτυξη αλγορίθμων που με την ανάλυση των δεδομένων εκπαίδευσής τους και των συσχετίσεων που δημιουργούν αυτά βελτιστοποιούν τις αποδόσεις τους. Βάσει αυτών, στην συνέχεια καθίσταται δυνατό αυτοί οι αλγόριθμοι να μπορούν να κάνουν προβλέψεις ή να προχωρούν σε ενέργειες που δεν έχουν οριστεί προγραμματιστικά εξ αρχής.

Οι κύριες κατηγορίες της είναι:

- **Μη επιβλεπόμενη μάθηση (Unsupervised Learning):** Η μάθηση γίνεται τροφοδοτώντας δεδομένα και στην συνέχεια το σύστημα ανακαλύπτει συσχετίσεις, διαφορές και

ομοιότητες στα δεδομένα, εξαγοντας κανόνες και συμπεράσματα που το οδηγούν να προβλέπει τις εξόδους του συστήματος [9].

- **Επιβλεπόμενη μάθηση** (Supervised Learning): Η μάθηση γίνεται δημιουργώντας σχέσεις εισόδου με τις εξόδους των δεδομένων, κατά την διάρκεια της εκπαίδευσης στο σύστημα τροφοδοτούνται έτοιμα δεδομένα εισόδων και των αντίστοιχων εξόδων, οπότε δημιουργεί τελικά αυτές τις σχέσεις [9].
- **Ενισχυτική μάθηση** (Reinforcement Learning): Το σύστημα μαθαίνει αλληλοεπιδρώντας και έχοντας επιβραβεύσεις ή ποινές για τις σωστές ή της λάθος ενέργειες που κάνει [10].

Η εξαγωγή συμπερασμάτων και κανόνων από την διαδικασία της εκπαίδευσης ονομάζεται συμπερασματική διαδικασία (Inference) και είναι απαραίτητη για την ικανότητα του συστήματος να κάνει προβλέψεις μετά την εκπαίδευση του.

Τα **Νευρωνικά Δίκτυα** (Neural Networks) έχουν ως βάση της δομής τους τον τρόπο λειτουργίας του ίδιου του ανθρώπινου εγκεφάλου και του τρόπου διαχείρισης ερεθισμάτων και πληροφορίας από τα δομικά του στοιχεία, δηλαδή τους νευρώνες. Όπως οι νευρώνες του εγκεφάλου είναι διασυνδεδεμένοι, έτσι και στα νευρωνικά δίκτυα υπάρχει διασύνδεση κόμβων που επεξεργάζονται τις πληροφορίες. Η εισαγόμενη πληροφορία επεξεργάζεται και συνδυάζεται με βάρος (Weight) και παράγει την έξοδο σύμφωνα με μία συνάρτηση ενεργοποίησης[9]. Η έννοια βάρος στα νευρωνικά δίκτυα μπορεί να εξηγηθεί ως γνώση που αποκτάται από την εκπαίδευση, η κρισιμότητα του χαρακτηριστικού.

Τα **Συνελκτικά Νευρωνικά Δίκτυα** (Convolutional Neural Networks ή CNN) είναι κατηγορία των νευρωνικών δικτύων που σχεδιάζονται με σκοπό την επεξεργασία δεδομένων που έχουν την μορφή δισδιάστατου πίνακα ή πλέγματος, όπως είναι οι εικόνες. Η βασική ιδέα είναι η ανεύρεση, μέσω φίλτρων, ειδικών χαρακτηριστικών, στην εικόνα ή σε τμήματά της, όπως σχημάτων, ακμών, αλληλουχίας χρωμάτων[12][13].

Παράδειγμα της πρώτης αναγνωρισμένης επιτυχίας και σημαντικής προόδου στην δημιουργία συνελκτικών δικτύων απαντάται το 2012 με το AlexNet, με την κατάκτηση πρώτης θέσης στον διαγωνισμό ImageNet Large Scale Visual Recognition Challenge (ILSVRC 2012), με εισαγωγή πρωτοποριακών μεθόδων που απέτρεπαν την υπερπροσαρμογή του μοντέλου και δημιουργία ψευδώς θετικών ή αρνητικών αποτελεσμάτων καθώς και την δυνατότητα εκπαίδευσης σε μονάδες επεξεργασίας γραφικών (GPU) [14].

Τα μοντέλα της τεχνητής νοημοσύνης παρά τις δυνατότητες που παρέχουν, έχουν μεγάλη ζήτηση σε πόρους, όπως υπολογιστική ισχύ, μνήμη αλλά και ως αποτέλεσμα ενέργεια. Αυτό δημιουργεί μια σοβαρή πρόκληση για την χρήση τους σε ενσωματωμένα συστήματα και φορητές συσκευές, που όμως οδήγησε στην ανάπτυξη τεχνικών βελτιστοποίησης όπως την κβαντοποίηση των μοντέλων και την σχεδίαση πιο αποδοτικών μοντέλων, κατάλληλων να αποδώσουν σε συστήματα με μειωμένες δυνατότητες υπολογιστικής ισχύος και περιορισμένης παροχής ενέργειας.

3.2.2 Τεχνητή Νοημοσύνη Παρυφής (Edge AI)

Η εκτέλεση των αλγορίθμων με απευθείας λήψη των δεδομένων από το αντίστοιχο σύστημα, χωρίς την ανάγκη για αποστολή τους σε απομακρυσμένα συστήματα για επεξεργασία, ονομάζεται **Τεχνητή Νοημοσύνη Παρυφής** (Edge AI), είναι η τεχνητή νοημοσύνη που εκτελείται στα άκρα του δικτύου, χωρίς να χρειάζεται ουσιαστική μεταφορά της πληροφορίας. Η χρήση της απαντάται σε συσκευές όπως συσκευές του διαδικτύου των πραγμάτων, ενσωματωμένα συστήματα και κινητές συσκευές. Η τεχνητή νοημοσύνη παρυφής χαρακτηρίζει μια θεμελιώδη διαφοροποίηση του τρόπου

λειτουργίας των συστημάτων τεχνητής νοημοσύνης με την διαδικασία της επεξεργασίας των δεδομένων να ενσωματώνεται στο ίδιο σύστημα που παράγει η εντοπίζει την πληροφορία.

Κύρια χαρακτηριστικά της είναι:

- **Αξιοπιστία λειτουργίας** χωρίς εξάρτηση από εξωτερικά συστήματα και διασύνδεση με το διαδίκτυο.
- **Χαμηλή καθυστέρηση** απόκρισης λόγω της τοπικής επεξεργασίας και της απομάκρυνσης του χρόνου μεταφοράς των δεδομένων μέσω διαδικτύου από την εξίσωση.
- Μεγαλύτερες δυνατότητες **προστασίας της ιδιοτικότητας** λόγω της μη μεταφοράς δεδομένων στο διαδίκτυο για την επεξεργασία τους.

Έχοντας τους περιορισμούς που ορίζουν τις φορητές συσκευές αλλά και τα μη διασυνδεδεμένα με το διαδίκτυο συστήματα, η υλοποίηση τεχνητής νοημοσύνης παρυφής καλείται να λειτουργήσει με πιο μικρά μοντέλα, με λιγότερες απαιτήσεις επεξεργασίας, ώστε να είναι βιώσιμη στα συστήματα αυτά. Με τις παρακάτω τεχνικές επιτυγχάνεται η προσαρμογή των μοντέλων για την λειτουργία υπό αυτές τις συνθήκες:

- **Κβαντοποίηση Μοντέλων** (Quantization): Μείωση της ακρίβειας στην αναπαράσταση των αριθμητικών δεδομένων με μετατροπή των μοντέλων από 32bit κινητής υποδιαστολής → 8bit ακεραίους αριθμούς, κατά συνέπεια μειώνοντας το μέγεθος και τις απαιτήσεις μνήμης.
- **Αποκοπή Συνδέσεων** (Pruning): Επιλεκτική απομάκρυνση συνδέσεων, νευρώνων, βαρών ή φίλτρων του μοντέλου χωρίς να διακυβεύεται η αξιοπιστία του.
- **Απόσταξη Γνώσης** (Knowledge Distillation): Εξαγωγή και μεταβίβαση της "γνώσης" ενός εκπαιδευμένου μεγάλου μοντέλου – δασκάλου σε ένα πιο μικρό μοντέλο – μαθητή.
- **Συμπύεση Μοντέλων** (Model Compression): Κοινή χρήση βαρών, κωδικοποίηση Huffman και τεχνικές μείωσης μνήμης.
- Δημιουργία ελαφριών αρχιτεκτονικών νευρωνικών δικτύων: Εξ αρχής σχεδιασμός ελαφριών μοντέλων.

3.2.3 Μηχανική Όραση

Η Μηχανική Όραση είναι ο κλάδος της τεχνητής νοημοσύνης που καλείται να αντιληφθεί αλλά και να ερμηνεύσει εικόνες, οπτικές πληροφορίες, και κατά επέκταση και βίντεο με παρόμοιο τρόπο με την όραση των ανθρώπων. Είναι ένας κλάδος με μεγάλη άνθιση και πολλές εφαρμογές ακόμη και σε βιομηχανικά συστήματα, όπως για παράδειγμα, για την παρακολούθηση της παραγωγικής διαδικασίας και της ασφάλειας σε εργοστάσια.

Η διαδικασία που ακολουθείται για την ανάλυση των οπτικών πληροφοριών – δεδομένων που λαμβάνονται είναι η εξής:

- Η αρχική επεξεργασία της πληροφορίας (Pre-Processing) με μετασχηματισμούς των εικόνων, εφαρμογή φίλτρων, αλλαγή μεγέθους, αφαίρεση θορύβου, με τελικό στόχο την προσαρμογή των δεδομένων για να αφαιρεθούν παράγοντες που θα επηρέαζαν αρχικά την διαδικασία την εκπαίδευσης και τελικά την απόδοση του μοντέλου.
- Η εξαγωγή χαρακτηριστικών (Feature Extraction) και συσχετίσεων στις εικόνες, δημιουργώντας έτσι τους κανόνες της μεθόδου κρίσης και της λήψης αποφάσεων του μοντέλου. Η διαδικασία αυτή, συνήθως γίνεται με χρήση συνελκτικών δικτύων.

3.2.4 MobileNetV2

Όπως αναφέρθηκε και παραπάνω για την χρήση της τεχνητής νοημοσύνης παρυφής, γίνεται αξιοποίηση ελαφριών αρχιτεκτονικών, κατάλληλα σχεδιασμένων εξ αρχής για ανάπτυξη στις αντίστοιχες συσκευές. Μερικά μοντέλα με τέτοια αρχιτεκτονική είναι το SqueezeNet, το ShuffleNet και το MobileNet. Αυτά έχουν στην βάση της σχεδίασής τους την μείωση του υπολογιστικού φόρτου, διατηρώντας όμως την ακρίβεια στην εξαγωγή αποτελεσμάτων.

Το MobileNet αναπτύχθηκε αρχικά από τον Laurent Sifre το 2013-2014 ως εξέλιξη και παραλλαγή του AlexNet, μετέπειτα δημοσιευμένο από την Google ως MobileNetV1 το 2017 και τέλος ως MobileNetV2 το 2018 [15][16], αποτελεί ένα νευρωνικό δίκτυο με προηγμένη αρχιτεκτονική, που έχει σχεδιαστεί με σκοπό την χρήση του σε συστήματα με περιορισμό πόρων.

Βασικά χαρακτηριστικά της αρχιτεκτονικής του είναι:

- Σχεδίαση προσαρμοσμένη στην χρήση σε φορητές συσκευές.
- Χαμηλή υπολογιστική πολυπλοκότητα
- Χαμηλή ανάγκη μνήμης.
- Υπολειμματικές συνδέσεις μεταξύ εισόδου εξόδου για να γίνεται καλύτερη εκπαίδευση
- Δυνατότητες προσαρμογής, στις απαιτήσεις κάθε συστήματος, ανάμεσα στην ακρίβεια και την ταχύτητα απόδοσης αποτελεσμάτων.

Σε συγκριτικές μελέτες που πραγματοποιήθηκαν αποδείχθηκε να είναι βελτιωμένο, πιο συγκεκριμένα:

- Έγινε μείωση κατά 30% στις απαιτήσεις υπολογιστικής ισχύος σε σχέση με το MobileNetV1[16].
- Διατήρησε την ακρίβεια σε διεργασίες μηχανικής όρασης [16].
- Είχε βελτιωμένη απόδοση σε εργασίες ταξινόμησης ή ανίχνευσης αντικειμένων [16].

3.2.5 Κβαντοποίηση Μοντέλων και ONNX

Στην προσπάθεια για να βελτιστοποιηθούν τα μοντέλα μηχανικής μάθησης, αναδεικνύονται το πρότυπο ONNX και η τεχνική της Κβαντοποίησης.

Η κβαντοποίηση αποτελεί τεχνική που μειώνει τον φόρτο προς το νευρωνικό δίκτυο με βασικό χαρακτηριστικό την μεταβολή των μεταβλητών από κινητής υποδιαστολής σε ακέραιες τιμές (float → int). Με αυτό τον τρόπο γίνεται δυνατό να :

- Μειωθεί το μέγεθος των μοντέλων έως και 4 φορές [17].
- Επιταχυνθούν υπολογισμοί δυο με τρεις φορές [18].

Και επίσης ως συνέπεια των παραπάνω , να μειωθούν και οι απαιτήσεις ενέργειας σημαντικά.

Επιπροσθέτως, το ONNX (Open Neural Network Exchange) είναι ένα ανοιχτό πρότυπο, το οποίο καθιστά την αλληλοσυμβατότητα και μεταφορά μοντέλων μηχανικής μάθησης σε διαφορετικές πλατφόρμες. Κάποια από τα κύρια χαρακτηριστικά του είναι :

- Διαλειτουργικότητα σε άλλα περιβάλλοντα υλοποίησης.
- Απλοποίηση της διαδικασίας της ανάπτυξης των μοντέλων.
- Πιο εύκολη μεταφορά αλλά και βελτιστοποίηση των μοντέλων.

Στο πλαίσιο της παρούσας διπλωματικής και με σκοπό την βελτίωση των μοντέλων και την δυνατότητα να υλοποιηθούν σε συσκευές όπως το Raspberry Pi Zero 2W, θα αξιοποιηθούν οι παραπάνω τεχνικές.

3.3 Διαδίκτυο των Πραγμάτων και Ενσωματωμένα Συστήματα

3.3.1 Διαδίκτυο των Πραγμάτων

Με το Διαδίκτυο των Πραγμάτων δεν νοείται πλέον απλή λειτουργία συσκευών, αλλά διασύνδεσή τους και στο διαδίκτυο, δίνοντας έτσι την δυνατότητα αλληλεπίδρασης, δημιουργίας σεναρίων, δημιουργούνται πλέον έξυπνες συσκευές προσδίδοντας επιπλέον τρόπους διαχείρισης και λειτουργικότητας. Στα τέλη της δεκαετίας του 1990, ο Βρετανός πρωτοπόρος τεχνολογίας Kevin Ashton, συνιδρυτής του Auto-ID Laboratory στο MIT, επινόησε τον όρο "The Internet of Things" και οραματίστηκε έναν κόσμο με συσκευές καθημερινής χρήσης που θα είχαν νοημοσύνη και δυνατότητα διασύνδεσης [19].

Το διαδίκτυο των πραγμάτων σήμερα έχει διευρυνθεί και υιοθετηθεί τόσο, ώστε να έχει εδραιωθεί ως ένα οικοσύστημα που σε παγκόσμιο επίπεδο να έχει έναν εκτιμώμενο αριθμό διασυνδεδεμένων συσκευών άνω των 19.8 δισεκατομμυρίων (στατιστική αποτίμηση έτους 2025)[20]. Ενώ πλέον είναι χαρακτηριστικό συσκευών ευρείας χρήσης, παράλληλα έχει κάλυψη και τομέων όπως η ιατρική παρακολούθηση, η αστική διαχείριση και λειτουργία, η βιομηχανία, η γεωργία ακριβείας και η παρακολούθηση περιβαλλοντικών συνθηκών.

Ως τεχνολογία το διαδίκτυο των πραγμάτων βασίζεται άρρηκτα στην επικοινωνία και μετάδοση πληροφορίας. Επομένως, τα δίκτυα και τα πρωτόκολλα μετάδοσης των δεδομένων παίζουν κεντρικό ρόλο στην λειτουργία του, με αισθητήρες να συλλέγουν την πληροφορία η οποία τελικά θα οδηγηθεί για ανάλυση και επεξεργασία, σε πραγματικό χρόνο, σε πλατφόρμες νέφους (Cloud) και τεχνητής νοημοσύνης.

3.3.2 Ενσωματωμένα Συστήματα και Πρωτόκολλο BLE

Η ιστορική απαρχή των ενσωματωμένων συστημάτων έγινε την δεκαετία του 1960 στον τομέα της αεροδιαστημικής, με βασικό σταθμό, το διαστημικό πρόγραμμα της NASA, Apollo[21]. Από ψηφιακούς ελεγκτές απλής λειτουργίας, έγινε εξέλιξη σε πολύπλοκα υπολογιστικά συστήματα για διαχείριση απλών μέχρι και απόλυτα κρίσιμων λειτουργιών σε κάθε τομέα της τεχνολογίας.

Τα ενσωματωμένα συστήματα έχουν συγκεκριμένες λειτουργίες σε κάθε σύστημα και προορίζονται για να τις εκτελούν με αποδοτικότητα και αξιοπιστία σε πραγματικό χρόνο.

Οι κύριοι τομείς που κυριαρχούν σήμερα τα ενσωματωμένα συστήματα είναι:

- Ιατρικά μηχανήματα.
- Αυτοκινητοβιομηχανία.
- Βιομηχανικοί αυτοματισμοί.
- Εξοπλισμοί τηλεπικοινωνιακών δικτύων.

Το Bluetooth Χαμηλής Ενέργειας (BLE) επίσημα παρουσιάστηκε το 2010 [22] ως η λύση στην ανάγκη για πιο αποδοτική επικοινωνία φορητών συσκευών και ενσωματωμένων συστημάτων. Παρουσιάζει χαρακτηριστικά που το καθιστούν ανώτερο της επικοινωνίας Bluetooth, με σημαντικότερο για τις φορητές συσκευές την εξαιρετικά μικρότερη κατανάλωση ενέργειας, με

δυνατότητα λειτουργίας έως και 2 χρόνια με μπαταρία τύπου νομίσματος (CR2032) [22]. Άλλα χαρακτηριστικά του είναι:

- Η εμβέλεια επικοινωνίας έως και 50 μέτρα(σε ανοιχτό πεδίο).
- Οι ταχύτητες μετάδοσης δεδομένων που είναι έως 2Mbps, αλλά συνήθως λόγω πακέτων μετάδοσης και πρωτοκόλλου περιορίζονται σε μικρότερες τιμές περίπου στο 1Mbps.

3.4 Εφαρμογές Κινητών Συσκευών Android

3.4.1 Ανάπτυξη Εφαρμογών Android

Οι εφαρμογές Android δημιουργούν διεπαφές για τον χρήστη για να χρησιμοποιήσει τεχνολογίες και δυνατότητες εξωτερικών συσκευών, το διαδίκτυο, την ίδια την συσκευή με τους αισθητήρες που αυτή διαθέτει και τα συστήματα αναπαραγωγής πολυμέσων. Με ένα 73.9% των κινητών συσκευών παγκοσμίως να κάνουν χρήση του λογισμικού Android [23], για την ανάπτυξη εφαρμογών για αυτό δίνονται επιλογές τόσο στην επιλογή γλώσσας για την ανάπτυξη, διαφορετικών περιβαλλόντων προγραμματισμού και αρχιτεκτονικής.

Η γλώσσα προγραμματισμού που προτείνεται από την ίδια την Google, η οποία και διαθέτει τις εκδόσεις του λογισμικού Android στην αγορά, είναι η Kotlin. Αυτή έχει επικρατήσει ως γλώσσα για Android, όμως υπάρχουν και άλλες που χρησιμοποιούνται, όπως η Java,ως πρώην προτεινόμενη γλώσσα μέχρι πριν το 2019, και πιο σπάνια η C++.

Αντιστοίχως, αν και πιο διαδεδομένο και επίσημο περιβάλλον ανάπτυξης εφαρμογών Android το Android Studio, δεν είναι το μόνο που χρησιμοποιείται. Εναλλακτικά χρησιμοποιείται από προγραμματιστές το IntelliJ IDEA και πιο παλιά χρησιμοποιούνταν το Eclipse. Υπάρχουν και άλλα προγράμματα για την σύνθεση εφαρμογών, όπως και πλέον υπάρχουν προγράμματα τεχνητής νοημοσύνης όπου περιγράφεται η ζητηθείσα εφαρμογή και το πρόγραμμα συνθέτει την εφαρμογή.

3.4.2 Αρχιτεκτονική Εφαρμογής και Λειτουργικότητα

Η αρχιτεκτονική της εφαρμογής απαιτεί ξεκάθαρη σχεδίαση με προσοχή στην δομή, ώστε να εξασφαλίζεται:

- Δυνατότητα μελλοντικών επεκτάσεων.
- Ξεκάθαρος κώδικας και κατανοητός.
- Εύκολη συντήρηση.
- Ομαλή διαλειτουργικότητα σε όλες τις πιθανές Android συσκευές.

Το MVVM αντιστοιχεί στο πιο διαδεδομένο πλέον πρότυπο αρχιτεκτονικής, διαχωρίζοντας τις λειτουργικές περιοχές μίας εφαρμογής σε 3 κομμάτια:

- Model: Ανεξάρτητο από την αλληλεπίδραση του χρήστη και υπεύθυνο για την διαχείριση και επεξεργασία πληροφοριών.
- View: Activities, Fragments, XML layout, υπεύθυνο για την απεικόνιση των δεδομένων και για να συλλέγονται τα συμβάντα από τον χρήστη, Συμπεριλαμβάνει τα οπτικά στοιχεία που έχει αλληλεπίδραση ο χρήστης.
- ViewModel: Ο διαμεσολαβητής επικοινωνίας ανάμεσα στο View και στο Model.Υπεύθυνο για την μεταφορά των δεδομένων από το ένα στο άλλο, διατηρεί την τιμή δεδομένων χωρίς να επιφορτίζεται με αυτό τον ρόλο η λειτουργία View.

Στην αρχιτεκτονική αυτή, η ροή των δεδομένων είναι αμφίδρομη, το View μπορεί να καλέσει μεθόδους στο ViewModel και αυτό στην συνέχεια να αλληλοεπιδράσει με το Model κάνοντας τροποποίηση ή ανάκτηση δεδομένων. Οι τελικές πληροφορίες με την σειρά τους μεταφέρονται πάλι στο View συγχρονίζοντας και ενημερώνοντας με αποδοτικό τρόπο την παρουσίαση των δεδομένων αυτών προς τον χρήστη.

Μέσω της αρχιτεκτονικής MVVM επιτυγχάνεται διαχωρισμός των αρμοδιοτήτων καθιστώντας ευκολότερη την αποσφαλμάτωση, διευκολύνοντας την εκτέλεση δοκιμών και δίνοντας την δυνατότητα να αλλάζουν τμήματα της λειτουργίας της εφαρμογής χωρίς να επηρεαστεί η λειτουργικότητά της.

3.4.3 Διεπαφή και Εμπειρία Χρήστη

Μια εφαρμογή Android μπορεί να έχει πολλές δυνατότητες και να είναι εξαιρετικά χρήσιμη, όμως είναι πολύ σημαντικό να προβάλει τις δυνατότητες διαχείρισής της με τέτοιο τρόπο ώστε να μπορεί ο χρήστης να τις αξιοποιήσει. Αυτό τον ρόλο αναλαμβάνει ο σχεδιασμός της Διεπαφής και Εμπειρίας Χρήστη (UI & UX). Πέραν του προφανούς της απόδοσης των δεδομένων σε μία οθόνη η λειτουργικότητα της εφαρμογής οφείλει να είναι διαισθητική, οι χρήστες βασίζονται στην συνήθεια και στο πως λειτουργεί η πλειοψηφία των εφαρμογών όσον αφορά τα μενού, καρτέλες, κουμπιά και πεδία ελέγχου και εισαγωγής δεδομένων. Βάσει αυτών, ο σχεδιασμός πρέπει να προσαρμόζεται:

- Στο πεδίο χρήσης της εφαρμογής και στον σκοπό για τον οποίο αυτή έχει δημιουργηθεί.
- Στο προφίλ του τελικού χρήστη. Είναι διαφορετικός, για παράδειγμα, ο σχεδιασμός όταν απευθύνεται σε νέους και διαφορετικός όταν απευθύνεται σε ηλικιωμένους.
- Στις λειτουργικές απαιτήσεις της εφαρμογής.
- Τους περιορισμούς που ορίζονται από το εκάστοτε υλικολογισμικό χρήσης. Άλλες για παράδειγμα οι δυνατότητες ενός τάμπλετ και άλλες ενός κινητού τηλεφώνου.

Πολύ σημαντικό στοιχείο για τον σχεδιασμό είναι η προσαρμοστικότητα της διάταξης (Responsive Design), ώστε να είναι εφικτό η εφαρμογή να λειτουργεί σε συσκευές με μια πληθώρα διαφορετικών συνδυασμών μεγέθους και ανάλυσης οθόνης. Στόχος είναι μια εφαρμογή εύκολη στην λειτουργία που αποδίδει στον χρήστη το επιθυμητό αποτέλεσμα χωρίς να χρειάζεται περίπλοκη διαχείριση και αλληλεπίδραση.

3.5 Σχετικές Μελέτες-Εργασίες

Σε αυτήν την ενότητα παρατίθενται 2 παραδείγματα εφαρμογής τεχνολογιών τεχνητής νοημοσύνης παρυφής που χρησιμοποιούνται και για την υλοποίηση του συστήματος ασφαλείας και ανίχνευσης παρουσίας παιδιού στο παιδικό κάθισμα. Η βιβλιογραφική αναζήτηση παρόμοιων συστημάτων κατέδειξε πολλά αποτελέσματα, αναδεικνύοντας τον τρόπο αξιοποίησης και χρήσης της τεχνολογίας αυτής στην υπηρεσία της ασφάλειας των ανθρώπων.

3.5.1 Συναφείς Εργασίες σε Συστήματα Ασφαλείας με Τεχνητή Νοημοσύνη Παρυφής

- «*Factory Worker Safety Mask Monitoring System Based on Deep Learning and AI Edge Computing*» [Ασφάλεια εργαζομένων σε εργοστάσιο με σύστημα μάσκας ασφαλείας βασισμένο στην ενσωμάτωση βαθιάς μάθησης και τεχνητής νοημοσύνης παρυφής] [24].

Κεφάλαιο 3ο:

Η παραπάνω δημοσίευση έχει ως αντικείμενο την ασφάλεια των εργαζομένων και πιο συγκεκριμένα τον εντοπισμό της ορθής χρήσης του προστατευτικού εξοπλισμού σε εργοστάσια, για την αποφυγή τραυματισμών και την αποφυγή φθοράς ή καταστροφής εξοπλισμού ή μηχανημάτων.

Το προτεινόμενο σύστημα κάνει χρήση τεχνητής νοημοσύνης (AI) και τεχνητής νοημοσύνης παρυφής(Edge AI) σε συνδυασμό με βαθιά μάθηση(Deep Learning) για την εκπαίδευση μοντέλου κατάλληλου για τον σκοπό του εντοπισμού του προστατευτικού εξοπλισμού. Το μοντέλο λειτουργώντας τοπικά προσδιορίζει αν γίνεται σωστή χρήση του εξοπλισμού, προσφέροντας ένα εργαλείο για την έγκαιρη αντιμετώπιση και πρόβλεψη ατυχημάτων και συνεισφέροντας στην δημιουργία ενός έξυπνου και ασφαλούς βιομηχανικού περιβάλλοντος.

- « *Intelligent Edge-Based System for Driver Drowsiness Monitoring in Mobile Crowd-sourcing Environments*» [Έξυπνο σύστημα τεχνητής νοημοσύνης παρυφής για επιτήρηση υπνηλίας οδηγών σε πλαίσιο κινητής συνεργατικής συλλογής δεδομένων] [25]

Σε αυτή την δημοσίευση προτείνεται ένα σύστημα για την βελτίωση της οδηγικής ασφάλειας κάνοντας χρήση κινητών συσκευών-τηλεφώνων ως μέσα εντοπισμού σημαδιών κόπωσης και υπνηλίας των οδηγών.

Το σύστημα που προτείνεται, κάνει χρήση συνελκτικών δικτύων για να αναλύσει οπτικά δεδομένα, κάνοντας χρήση της κάμερα της κινητής συσκευής-τηλεφώνου, εκφράσεων προσώπου, κινήσεων και άλλα σημάδια που προδίδουν κατάσταση κόπωσης του οδηγού. Επίσης αναπτύσσεται με τεχνητή νοημοσύνη παρυφής(Edge AI) για την τοπική επεξεργασία των δεδομένων, χωρίς να κινδυνεύει η ιδιωτικότητα του χρήστη και εξασφαλίζοντας υψηλή ταχύτητα απόκρισης.

Η ακρίβεια του συστήματος φτάνει το 98.4% συνεισφέροντας έτσι στην πρόληψη των ατυχημάτων και ασφάλεια των οδικών μετακινήσεων.

3.6 Επίλογος

Το παρόν κεφάλαιο είχε στόχο να εισαγάγει κάποιες έννοιες και να περιγράψει κάποιες τεχνολογίες, των οποίων θα γίνει χρήση για την υλοποίηση του συστήματος της διπλωματικής αυτής. Κύρια σημεία ήταν η τεχνολογία της τεχνητής νοημοσύνης και οι υποκατηγορίες αυτής, η ανάπτυξη συστημάτων σε φορητές συσκευές και η ανάπτυξη εφαρμογών Android για κινητές συσκευές.

Η εξέλιξη της τεχνολογίας αποζητά την αποδοτικότητα, την φορητότητα, την μείωση της ενεργειακής απαίτησης των συσκευών και την απλούστευση προβλημάτων. Η ανάγκη για συνδυασμό των παραπάνω, έχει οδηγήσει σε τεχνολογίες που το κάνουν εφικτό. Τεχνολογίες επικοινωνίας, όπως το BLE, ελαφριές αρχιτεκτονικές όπως το MobileNetV2, συνελκτικά νευρωνικά δίκτυα και εφαρμογές Android ανεπτυγμένων δυνατοτήτων. Η εξέλιξη αυτή μπορεί να αναφερθεί ως έξυπνες συσκευές, συσκευές με προσαρμοστικότητα και με δυνατότητες αλληλεπίδρασης.

Είναι εφικτό με την αξιοποίηση των τεχνολογιών που προσφέρονται μέσω της τεχνητής νοημοσύνης να γίνεται μετασχηματισμός της καθημερινότητας με προσφορά λύσεων και αύξηση της ασφάλειας.

Στο επόμενο κεφάλαιο γίνεται η παρουσίαση της ανάπτυξης του μοντέλου τεχνητής νοημοσύνης το οποίο αποτέλεσε έναν από τους βασικούς πυλώνες της παρούσας εργασίας. Αναλύονται τα βήματα της σχεδίασης και της δημιουργίας του συνόλου δεδομένων, η επιλογή της αρχιτεκτονικής MobileNetV2, οι μέθοδοι εκπαίδευσης και βελτιστοποίησης, καθώς και η διαδικασία κβαντοποίησης και μετατροπής του μοντέλου σε μορφή κατάλληλη για να μπορέσει να ενσωματωθεί στον αισθητήρα Sony IMX500. Μέσα από αυτό το κεφάλαιο γίνεται η τεκμηρίωση της μεθοδολογίας που ακολουθήθηκε για να μπορέσει το μοντέλο να ανταποκριθεί στις απαιτήσεις για επεξεργασία δεδομένων τοπικά σε πραγματικό χρόνο.

Κεφάλαιο 4ο: Ανάπτυξη Μοντέλου Τεχνητής Νοημοσύνης

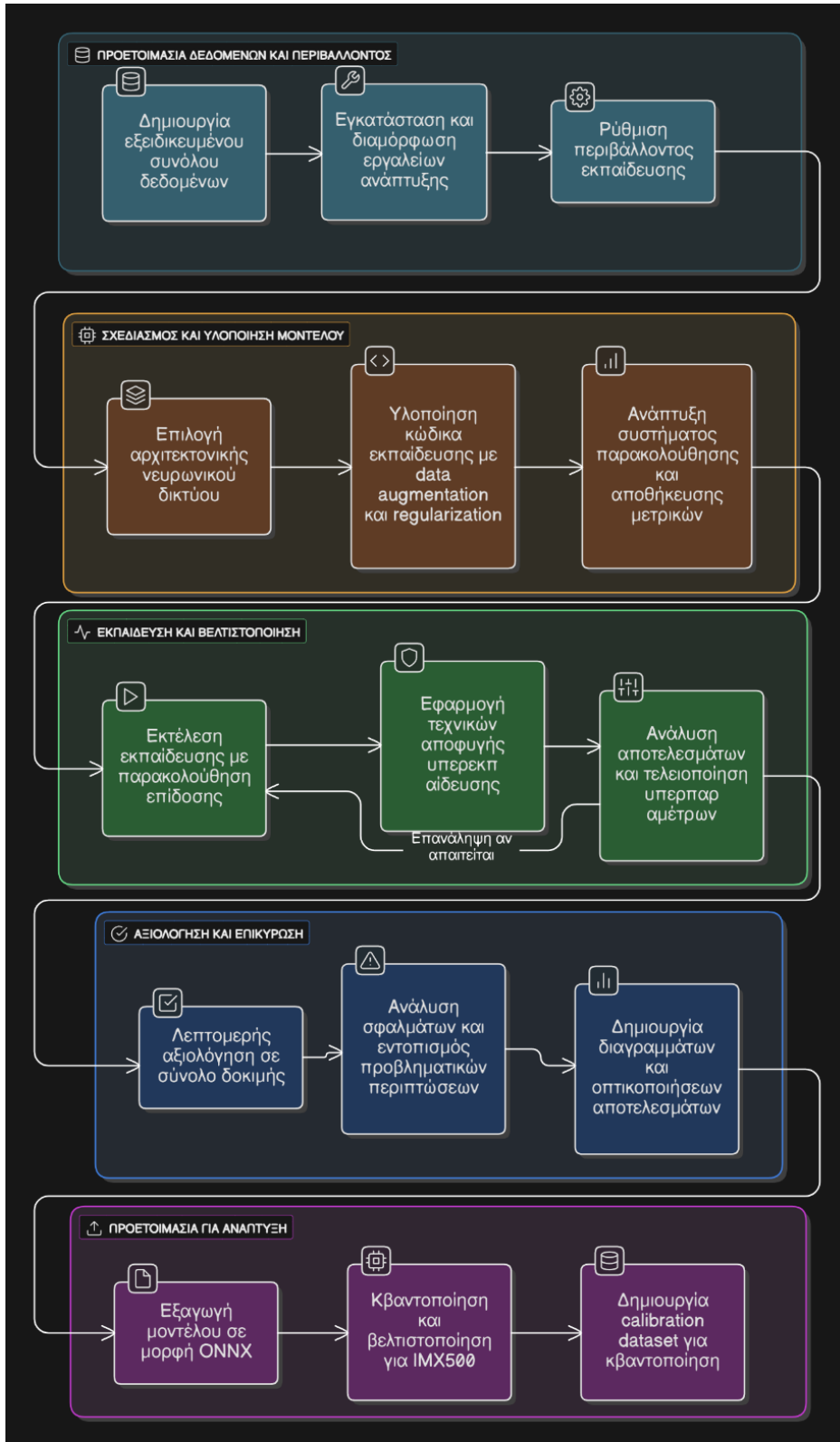
4.1 Εισαγωγή

Στο παρόν κεφάλαιο θα γίνει παρουσίαση της διαδικασίας της δημιουργίας του συνελκτικού μοντέλου τεχνητής νοημοσύνης για την ταξινόμηση εικόνων σε εικόνα με παιδί ή χωρίς παιδί (labels: Kid/noKid). Το μοντέλο βασίζεται στην αρχιτεκτονική MobileNetV2, η οποία ήταν κατάλληλη για την συγκεκριμένη διεργασία και την ανάπτυξη του μοντέλου σε φορητή συσκευή.

Παράλληλα θα γίνει ανάλυση της διαδικασίας δημιουργίας του συνόλου δεδομένων, άρα των φωτογραφιών, για την εκπαίδευση του μοντέλου. Ανάλυση της εκπαίδευσης και των μετρικών που προέκυψαν από αυτή και τέλος η διαδικασία της βελτιστοποίησης του μοντέλου και η ανάπτυξη στον αισθητήρα IMX500.

4.2 Παρουσίαση των Βημάτων Σχεδιαστικής Του Συστήματος

Πρώτα βήματα για την υλοποίηση του μοντέλου ήταν η ανάλυση των απαιτήσεων του αποδέκτη του μοντέλου και στην συγκεκριμένη περίπτωση της κάμερας, επομένως του αισθητήρα IMX500. Ο αισθητήρας έχει συγκεκριμένες απαιτήσεις στον τύπο του μοντέλου που δέχεται καθώς και στην διαδικασία που χρειάζεται για να ενσωματωθεί το μοντέλο στον αισθητήρα, οπότε ακολουθήθηκαν οι οδηγίες από την Sony και την Raspberry, ώστε να δημιουργηθεί σωστά το μοντέλο.



Εικόνα 4-1: Διάγραμμα σχεδίασης και υλοποίησης του μοντέλου

Επόμενο βήμα ήταν η εγκατάσταση των προγραμμάτων ανάπτυξης κώδικα και η εγκατάσταση των απαιτούμενων βιβλιοθηκών. Η γλώσσα προγραμματισμού που χρησιμοποιήθηκε ήταν η Python.

Κεφάλαιο 4ο:

Στην συνέχεια έγινε αναζήτηση και τελικώς σύνθεση και προετοιμασία του συνόλου δεδομένων (dataset) εικόνων, ώστε να μπορεί να γίνει εκπαίδευση του μοντέλου στην κατηγοριοποίηση.

Με Τα δεδομένα έτοιμα και ήδη επιλεγμένη την αρχιτεκτονική νευρωνικών δικτύων MobileNetV2, ακολούθησε η σύνθεση κώδικα για την εκπαίδευση. Παράλληλα ο κώδικας αυτός παρείχε και τα απαραίτητα μετρικά, ώστε να γίνει ανάλυση της αξιοπιστίας και απόδοσης του μοντέλου κατά την διάρκεια της εκπαίδευσης και τελικώς ως γενική επισκόπηση.

Η εκπαίδευση του μοντέλου στην συνέχεια επαναλήφθηκε με διάφορες παραμετροποιήσεις ώστε να εξαχθεί το επιθυμητό αποτέλεσμα και επόμενη φάση ήταν αυτή της τελικής αξιολόγησης του μοντέλου ακολουθούμενη από την εξαγωγή σε ONNX, κβαντοποίηση, βελτιστοποίηση και ανέβασμα στον αισθητήρα.

4.3 Δημιουργία Συνόλου Δεδομένων

Για το σύνολο δεδομένων, αρχικά έγινε αναζήτηση σε διαδικτυακές πηγές για αναζήτηση έτοιμων συνόλων δεδομένων, όμως, όπως φάνηκε με τις δοκιμές, δεν ήταν κατάλληλα. Άξιο αναφοράς είναι το σύνολο δεδομένων Sviro Dataset [26] το οποίο έχει αναπτυχθεί αποκλειστικά με συνθετικά δεδομένα, δημιουργημένα με πρόγραμμα προσομοίωσης και περιέχει έναν αριθμό φωτογραφιών άνω των 10 χιλιάδων με φωτογραφίες από 10 διαφορετικά μοντέλα αυτοκινήτων, με διαφορετικούς φωτισμούς, με παιδιά, ενήλικες και με πολλές παραλλαγές. Το Sviro αν και πολλά υποσχόμενο, δεν κατέστη δυνατό να προσφέρει ακριβής κατηγοριοποίηση, οπότε και απορρίφθηκε.

Το σύνολο δεδομένων τελικά συντέθηκε χειροκίνητα με λήψη εικόνων με άδεια χρήσης και με λήψη παρόμοιων συνόλου δεδομένων [27][28] και χειροκίνητη επιλογή και κατηγοριοποίηση εικόνων και τελικώς συντέθηκε το μοντέλο “Tsilikas_dataset_classification_kid”.

Για τις επιλεγμένες φωτογραφίες ήταν σημαντικό:

- Να είναι πραγματικές και όχι παραγμένες με προγράμματα προσομοίωσης ή τεχνητής νοημοσύνης.
- Να υπάρχει ποικιλία στις γωνίες λήψης και στην απόσταση λήψης.
- Να υπάρχουν εικόνες και με παιδιά και χωρίς.
- Να υπάρχουν εικόνες χωρίς και με παιδικό κάθισμα.
- Να υπάρχει ποικιλία στις συνθήκες φωτισμού.

Το σύνολο δεδομένων αποτελείται από τα εξής χαρακτηριστικά:

- Συνολικό μέγεθος: 684MB.
- Σύνολο εικόνων: 4639.
- Κατηγορίες για εκπαίδευση: 2 (Παρουσία ή όχι, παιδιού).

Για τον διαχωρισμό των εικόνων συντάχθηκε ο κώδικας στο ΠΑΡΑΡΤΗΜΑ Β και μέσω αυτού δημιουργήθηκαν οι φάκελοι του τελικού συνόλου δεδομένων Πίνακας 4-1.

Πίνακας 4-1: Διαχωρισμένα σύνολα δεδομένων σε φακέλους

Φάκελος	Εικόνες Kid	Εικόνες NoKid	Σύνολο	Kid %	NoKid %
Train (Εκπαίδευση)	2286	1144	3430	66,6	33,4
Val (Επικύρωση)	254	127	381	66,7	33,3
Test (Δοκιμή)	248	280	528	47	53
Calib (Βαθμονόμηση)	150	150	300	50	50

Ο διαχωρισμός σε εκπαίδευση, επικύρωση δοκιμή και τέλος βαθμονόμηση δίνει την δυνατότητα εξαγωγής συμπερασμάτων για το μοντέλο κατά την διάρκεια της εκπαίδευσής του. Με αυτό τον τρόπο κάθε φάκελος έχει συγκεκριμένο ρόλο:

- Val (Επικύρωση): Την παρακολούθηση της πρόοδο εκπαίδευσης και την δυνατότητα να σταματήσει πρόωρα εφόσον παρατηρηθεί υπερπροσαρμογή (Overfitting).
- Train (Εκπαίδευση): Την παροχή των δεδομένων για την εκπαίδευση.
- Test (Δοκιμή): Την αξιολόγηση του μοντέλου με δεδομένα που δεν έχει πρόσβαση κατά την εκπαίδευση.
- Calib (Βαθμονόμηση): Να γίνει η κβαντοποίηση με αντιπροσωπευτικό δείγμα των δεδομένων.

Παρατηρείται μη ομοιόμορφη κατανομή στις κλάσεις στους φακέλους με αναλογία συνόλου εκπαίδευσης 2:1 (kid:noKid), επικύρωσης 2:1 (kid:noKid), δοκιμής 2:1 (kid:noKid) και βαθμονόμησης 1:1 (ισοκατανομή). Αυτό αντιμετωπίστηκε με την προσαρμογή των βαρών.

4.4 Εκπαίδευση Και Βελτιστοποίηση Μοντέλου

Αρχικά, τα μοντέλα που δέχεται ο αισθητήρας IMX500 είναι PyTorch και TensorFlow. Μετά την δημιουργία του μοντέλου, πρέπει να χρησιμοποιηθεί το Edge AI Model Development Toolkit, δηλαδή το εξειδικευμένο εργαλείο για την επεξεργασία του μοντέλου και την παραμετροποίηση του, ώστε να μπορεί να χρησιμοποιηθεί στον αισθητήρα. Μέσω του εργαλείου αυτού γίνεται η κβαντοποίηση του μοντέλου για να μπορέσει να τρέξει με τους πόρους που διαθέτει ο αισθητήρας. Όπως επίσης μέσω του εργαλείου αυτού γίνεται και η μετατροπή του μοντέλου σε τύπο αρχείου που δέχεται ο αισθητήρας, ώστε να ενσωματωθεί στο σύστημα.

Τέλος, για την υλοποίηση των παραπάνω βημάτων η χρήση ισχυρού υπολογιστή είναι σχεδόν αναγκαία, κυρίως για την εκπαίδευση του μοντέλου, εφόσον επιλεγθεί να εκπαιδευτεί τοπικά. Το τελευταίο βήμα εξαιρείται, διότι εκτελείται στο Raspberry.

Κεφάλαιο 4ο:

```
img_classifier/
├── code/
│   ├── artifacts/
│   │   ├── best_run_meta.json           # Δεδομένα για την καλύτερη εκτέλεση
│   │   ├── kid_mnv2_best.pt           # Βέλτιστο μοντέλο ανα εποχή
│   │   ├── kid_mnv2_best_full.pth     # Πλήρες μοντέλο
│   │   ├── labels.txt                 # Ετικέτες κλάσεων
│   │   ├── training_log.csv           # πληροφορίες εκπαίδευσης
│   │   ├── y_pred.csv                 # Προβλέψεις μοντέλου
│   │   └── y_true.csv                 # Πραγματικές ετικέτες
│   ├── calibration_dataset_maker.py   # Δημιουργία συνόλου βαθμονόμησης
│   ├── export_onnx.py                 # Εξαγωγή σε μορφή ONNX
│   ├── plot_training_diagrams.py     # Δημιουργία γραφημάτων
│   └── train_kid_classifier_withMetricsSave.py # Εκπαίδευση+ εξαγωγή μετρικών
├── Tsilikas_dataset_classification_kid/ # φάκελος συνόλου δεδομένων (dataset)
│   ├── train/                         # Σύνολο Εκπαίδευσης
│   │   ├── kid/                       # Εικόνες με παιδιά
│   │   └── nokid/                     # Εικόνες χωρίς παιδιά
│   ├── test/                          # Σύνολο δοκιμής
│   │   ├── kid/
│   │   └── nokid/
│   ├── val/                           # Σύνολο επικύρωσης
│   │   ├── kid/
│   │   └── nokid/
│   └── calib/                          # Σύνολο βαθμονόμησης για κβαντοποίηση
│       ├── kid/
│       └── nokid/
```

Εικόνα 4-2: Οργάνωση αρχείων και κώδικα εκπαίδευσης, διαχωρισμού και διαγραμμάτων.

Για την διαδικασία της εκπαίδευσης τα δεδομένα και οι κώδικες οργανώθηκαν στον φάκελο `img_classifier` τοπικά στον υπολογιστή μαζί με το σύνολο δεδομένων και στην συνέχεια τα εξαγόμενα από την εκπαίδευση αποτελέσματα.

Για να επιτευχθεί η καλύτερη δυνατή απόδοση του μοντέλου, έγινε αύξηση του συνόλου των δεδομένων με τους ακόλουθους μετασχηματισμούς των εικόνων.

Μετασχηματισμοί για τα δεδομένα Εκπαίδευσης (Data Augmentation)

- `RandomResizedCrop(224, scale=(0.6, 1.0), ratio=(0.75, 1.33))`: Τυχαία περικοπή και αλλαγή μεγέθους
- `RandomHorizontalFlip()`: Οριζόντια αναστροφή
- `RandomRotation(15)`: Τυχαία περιστροφή έως 15 μοίρες
- `ColorJitter(0.2, 0.2, 0.2, 0.05)`: Αλλαγή φωτεινότητας, αντίθεσης, κορεσμού και απόχρωσης
- `RandomPerspective(0.2)`: Αλλαγή προοπτικής με πιθανότητα 20%
- `Normalize([0.485,0.456,0.406],[0.229,0.224,0.225])`: Κανονικοποίηση ImageNet

Μετασχηματισμοί για τα δεδομένα επικύρωσης (Validation):

- `Resize(256) → CenterCrop(224)`: Μετασχηματισμός της μικρότερης διάστασης μια εικόνας (ύψος ή πλάτος) σε 256px και μετά γίνεται περιμετρικά κόψιμο με σημείο αναφοράς το κέντρο της εικόνας, ώστε η τελική εικόνα που προκύπτει να είναι 224*224px.
- `Normalize([0.485, 0.456, 0.406], [0.229, 0.224, 0.225])`: Κανονικοποίηση ImageNet

4.4.1 Αρχιτεκτονική Μοντέλου

Το μοντέλο, κάνοντας χρήση της αρχιτεκτονικής MobileNetV2, κάνει χρήση των βαρών από την προεκπαίδευσή του στο σύνολο δεδομένων ImageNet. Η τεχνική που κάνει χρήση η συγκεκριμένη αρχιτεκτονική για να εξαχθούν τα χαρακτηριστικά με μείωση της ζήτησης υπολογιστικής ισχύος είναι η τεχνική των αντεστραμμένων υπολειμματικών μπλοκ (Inverted Residual Blocks). Τα υπολειμματικά μπλοκ είναι στοιχεία τα οποία χρησιμοποιούν δομές παράκαμψης (skip Connections) με το να προσθέτουν την αρχική είσοδο στην έξοδο, δηλαδή η έξοδος που θα προκύπτει έχει την μορφή:

$$\text{Έξοδος} = F(x) + x \quad (4-1)$$

Αυτό δίνει την δυνατότητα καλύτερης ροής της πληροφορίας χωρίς την εμφάνιση του φαινομένου της εξαφάνισης κλίσεων (vanishing gradient). Η πορεία των αντεστραμμένων υπολειμματικών μπλοκ έχει ως αφετηρία την επέκταση (expansion) των καναλιών αντί να έχει την συμπίεση, όπως συμβαίνει στα υπολειμματικά μπλοκ. Σε κάθε μπλοκ γίνεται εφαρμογή τριών λειτουργιών που εκτελούνται σε σειρά.

1. Αρχικά με την επέκταση (expansion), δηλαδή την αύξηση των καναλιών για καλύτερη αναπαράσταση.
2. Στην συνέχεια, με την κατά βάθος συνέλιξη (depthwise), υπεύθυνη για την εφαρμογή των φίλτρων αυτόνομα σε κάθε κανάλι, έτσι ώστε να μειωθεί η απαιτούμενη υπολογιστική ισχύ.
3. Και τέλος με την προβολή (Protection), ώστε να γίνει μείωση των καναλιών και επομένως συμπίεση της πληροφορίας.

Με την σταδιακή μείωση της χωρικής ανάλυσης, δηλαδή του μεγέθους των υπό εξέταση τμημάτων εικόνων, παράλληλα γίνεται αύξηση των καναλιών, αυτό έχει ως αποτέλεσμα τα χαμηλού επιπέδου χαρακτηριστικά (ακμές και υφές) να μετατρέπονται τελικώς, στο υψηλό επίπεδο, ως σημασιολογικές πληροφορίες (αντικείμενα και σχήματα).

Η υπάρχουσα ανισορροπία στις κλάσεις αντιμετωπίζεται με την εφαρμογή κανονικοποίησης (regularization) τυχαία νευρώνες, ώστε να αποφευχθεί η υπερπροσαρμογή του μοντέλου και καταλήξει να μην μπορεί να ανταποκριθεί αξιόπιστα σε πραγματικά δεδομένα.

Εφόσον το μοντέλο παρουσιάζει σημάδια στασιμότητας και μη βελτίωσης για συγκεκριμένο αριθμό εποχών εκπαίδευσης (epochs), γίνεται πρόωρη διακοπή της εκπαίδευσης (early stopping).

```
# Φόρτωση MobileNetV2 με προκαθορισμένα βάρη από ImageNet
model = models.mobilenet_v2(weights=models.MobileNet_V2_Weights.IMAGENET1K_V1)

# Τροποποίηση του τελικού ταξινομητή για 2 κλάσεις: kid, nokid
model.classifier[1] = nn.Linear(model.last_channel, 2)
```

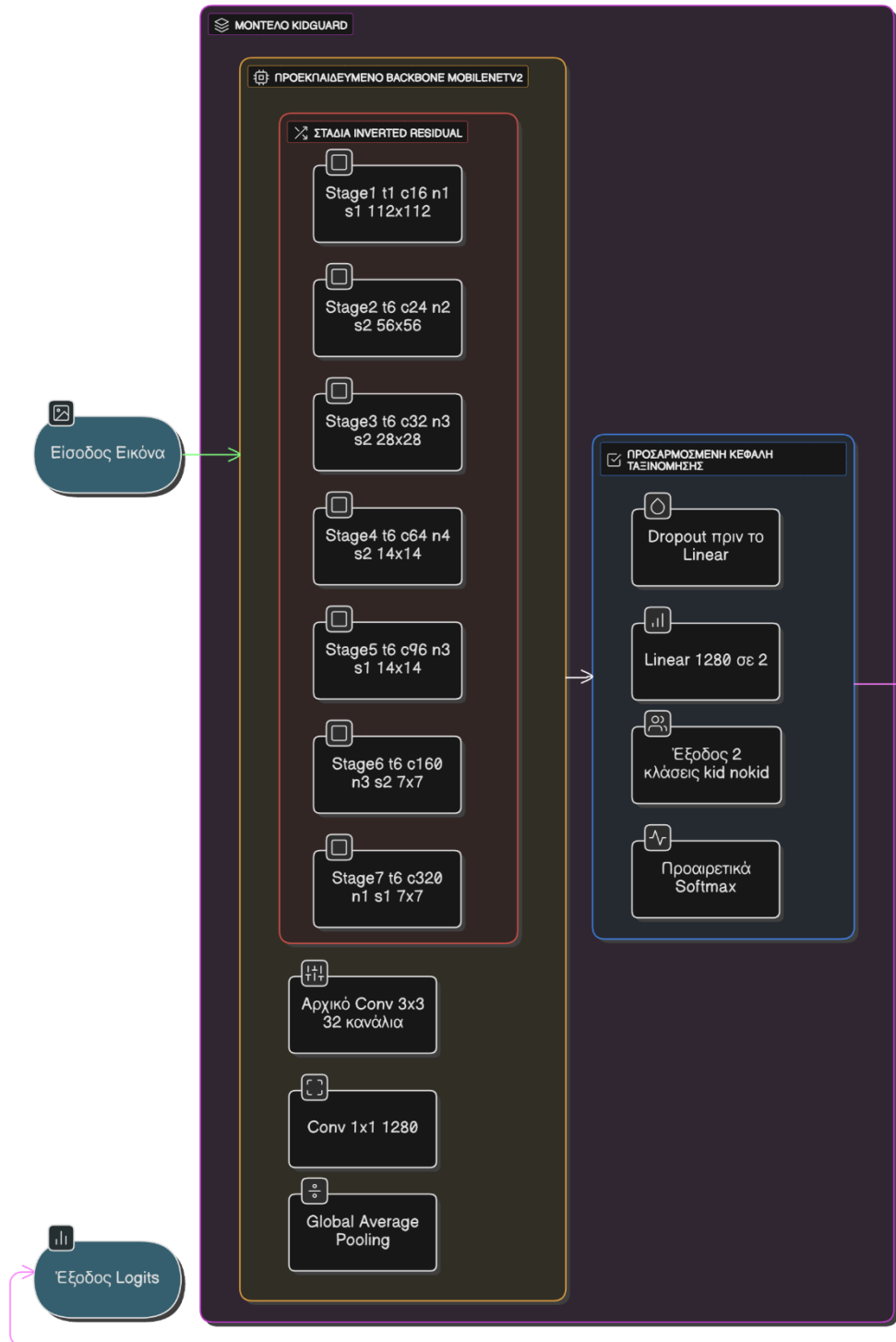
Κώδικας 4-1: Τροποποίηση του τελικού ταξινομητή

Για να εφαρμοστεί στο συγκεκριμένο μοντέλο η αρχιτεκτονική MobileNetV2, η τελική στρώση ταξινόμησης τροποποιείται, όπως φαίνεται [Error! Reference source not found.], ώστε να γίνεται δυαδική ταξινόμηση, λόγω των 2 κλάσεων που χρειάζονται για ταξινόμηση των εικόνων σε kid/noKid παρουσία δηλαδή και απουσία παιδιού.

Κεφάλαιο 4ο:

Το συνολικό διάγραμμα του μοντέλου απεικονίζεται στο **Error! Reference source not found.** και πιο αναλυτικά αποτελείται από:

- Βασική δομή
- MobileNetV2 κορμός του μοντέλου (Backbone)
 - Στάδια 1-7 : Διαφορετικές διαστάσεις καναλιών και χωρικές αναλύσεις (διαστάσεις εικόνας)
 - Σταδιακή μείωση χωρικών καναλιών (μείωση ανάλυσης εικόνων) από $112*112 \rightarrow 7*7$
 - Προοδευτική αύξηση του αριθμού των καναλιών από 16 έως 320, δηλαδή αύξηση των χαρακτηριστικών που ανιχνεύονται σε μια εικόνα (άκρες, σχήματα κτλ.)
- Επίπεδα εξαγωγής χαρακτηριστικών (Feature Extraction Layers)
 - Αρχικό Conv $3*3$ (συνέλιξη $3*3$ pixel) με 32 κανάλια εξόδου
 - Conv $1*1$, ανάλυση $1*1$ -pixel την φορά για την επέκταση σε 1280 χαρακτηριστικά
 - Global Average Pooling για μείωση χωρικών διαστάσεων, εξαγωγή μέσου όρου κάθε πίνακα τιμών χαρακτηριστικών 2 διαστάσεων, ώστε να γίνει μετατροπή σε 1 τιμή άρα και 1 διάσταση.
- Προσαρμοσμένη κεφαλή ταξινόμησης (Custom Classification Head)
 - Dropout layer (0.2), τυχαία απενεργοποίηση νευρώνων για κανονικοποίηση (regularization) και αποφυγή υπερπροσαρμογής (overfitting)
 - Linear layer: Μετατροπή $1280 \rightarrow 2$ για την δυαδική ταξινόμηση kid/noKid
 - Έξοδος Logits (ακατέργαστες βαθμολογίες) και μετατροπή σε πιθανότητες με SoftMax



Εικόνα 4-3: Διάγραμμα ροής δεδομένων και δομής μοντέλου KidGuard

4.4.2 Υπερπαράμετροι Εκπαίδευσης

Οι υπερπαράμετροι και οι τεχνικές για την εκπαίδευση του μοντέλου ήταν αναγκαίες για να δημιουργηθεί μοντέλο με υψηλά ποσοστά θετικής ανίχνευσης και για να γίνεται η εκπαίδευση σε μικρότερο χρόνο και να αναπροσαρμόζεται εύκολα για πολλαπλές δοκιμές. Η επιλογή των τιμών για

τις υπεραραμέτρους προέκυψε από την βιβλιογραφική έρευνα, συγκεκριμένα για το MobileNetV2 σε αντίστοιχα μοντέλα οπτικής ταξινόμησης, αλλά κυρίως με επαναλαμβανόμενες δοκιμές και παρατήρηση των αποτελεσμάτων.

Μέγεθος δειγμάτων (Batch Size): Ο αριθμός των δειγμάτων που λαμβάνονται κάθε φορά επηρεάζει την σταθερότητα και την αποδοτικότητα υπολογισμών των κλίσεων. Μικρός αριθμός (16-32) θα σήμαινε πιο καλή γενίκευση, αλλά περισσότερο θόρυβο στις κλίσεις. Αντίθετα για άνω του 128 θα απαιτούνταν πιο ισχυρές επεξεργαστικές δυνατότητες ή πολύς παραπάνω χρόνος. Το 64 είναι ισορροπημένο προσφέροντας ικανοποιητική εκπαίδευση σε χρόνο χωρίς να ζορίζει τον υπολογιστή.

Ρυθμός μάθησης (Learning Rate): Ο ρυθμός μάθησης προσδιορίζει πόσο γρήγορα τα νέα δεδομένα θα επηρεάσουν τα ήδη αποκτηθέντα χαρακτηριστικά που έχουν διαμορφωθεί από την προεκπαίδευση του μοντέλου βάσης πάνω στο σύνολο δεδομένων ImageNet. Υψηλότεροι ρυθμοί θα οδηγούσαν σε ασταθές μοντέλο καταστροφική αμνησία (catastrophic forgetting) χαρακτηριστικών, ενώ χαμηλοί θα έκαναν το μοντέλο να δημιουργεί τα νέα χαρακτηριστικά πολύ αργά, οπότε η διαδικασία μάθησης και εκπαίδευσης θα έπρεπε να είναι πολύ χρονοβόρα[29]. Για μοντέλα που στηρίζονται σε προεκπαιδευμένα μοντέλα και την μεταφορά γνώσης από αυτά, είναι σύνηθες να επιλέγεται ρυθμός μάθησης από $1e^{-4}$ έως $5e^{-4}$, στην συγκεκριμένη περίπτωση ύστερα από δοκιμές επιλέχθηκε η τιμή $3e^{-4}$.

Χρονοπρογραμματιστής ρυθμού μάθησης (Learning rate scheduler): Παράλληλα με την χρήση του ρυθμού μάθησης, χρησιμοποιήθηκε και ο Χρονοπρογραμματιστής ρυθμού μάθησης, ο οποίος προσφέρει την δυνατότητα προσαρμογής του ρυθμού μάθησης βαθμιαία με συνημιτονοειδή ρυθμό απόσβεσης (cosine annealing), ώστε να αποφευχθεί να φτάσει σε τοπικά ελάχιστα (local minima) η εκπαίδευση, και να γίνει ομαλότερη σύγκλιση (convergence), στοχεύοντας το καθολικό ελάχιστο την συνάρτησης απώλειας, δηλαδή το σημείο αυτό, όπου το μοντέλο έχει πλέον βελτιωθεί στα μέγιστα και δεν καθίσταται δυνατό να βελτιωθεί περαιτέρω.

Βελτιστοποιητής (optimizer) AdamW: Στην προσέγγιση αυτή, χρησιμοποιήθηκε ο βελτιστοποιητής (optimizer) AdamW που κάνει απόσβεση βαρών (weight decay) με τιμή $1e^{-4}$, προσαρμόζοντας τα βάρη κατά την διάρκεια της εκπαίδευσης, ώστε να γίνεται κανονικοποίηση και να αποφευχθεί η υπερπροσαρμογή.

Μέγιστος αριθμός εποχών και βαθμός υπομονής (Epochs and patience): Ο αριθμός εποχών καθορίζει πόσες φορές θα γίνει εκπαίδευση, με κάθε φορά που εκτελείται το μοντέλο να μαθαίνει και παραπάνω. Είναι ξεκάθαρο ότι το ιδανικό είναι να τρέξει όσες εποχές παραπάνω γίνεται, όμως αυτό δεν είναι δυνατόν, πρώτα από όλα λόγω χρόνου, αλλά επίσης δεν σημαίνει πως το μοντέλο μπορεί να βελτιώνεται πάντα σε κάθε εποχή. Κατά την εκπαίδευση για 30 εποχές υπήρξαν περιπτώσεις όπου το μοντέλο δεν εμφάνιζε βελτίωση απόδοσης μετά την 6^η εποχή, αλλά δεν γινόταν να σταματήσει η εκπαίδευση για να μην χαθούν τα δεδομένα. Ένας τρόπος που αυτό αντιμετωπίστηκε ήταν η εισαγωγή ρυθμού υπομονής. Αυτός είναι ένας αριθμός που προσδιορίζει πόσες εποχές μετά θα σταματήσει το μοντέλο από την εποχή που δεν παρατηρείται βελτίωση. Ως αποτέλεσμα, προέκυψε σε μία περίπτωση στις 60 εποχές συνολικά, η βελτίωση να σταματήσει στην 21 και το μοντέλο να σταματήσει εκπαίδευση στην 28, πράγμα το οποίο ήταν πολύ σημαντικό για την συνέχιση των δοκιμών, εξοικονόμηση χρόνου και της αποφυγής υπερπροσαρμογής. Η τιμή υπομονής 7 έδινε στο μοντέλο την δυνατότητα να έχει διακυμάνσεις στην βελτίωσή του, κάτι που είναι και αναμενόμενο λόγω του μικρού συνόλου δεδομένων που χρησιμοποιήθηκε.

Μέγεθος Εικόνας: Το ImageNet δέχεται ως είσοδο εικόνες με διαστάσεις 224*224, οπότε το ιδανικό μέγεθος για αποφυγή μη αποδοτικής λειτουργίας του μοντέλου ήταν αυτό, έτσι οι εικόνες του συνόλου δεδομένων μετασχηματίστηκαν σε τέτοιο μέγεθος.

Στρατηγικές εξισορρόπησης της ανισορροπίας των κλάσεων: Όπως αναφέρθηκε παραπάνω, στο σύνολο δεδομένων υπάρχει ανισορροπία μεταξύ των 2 κλάσεων, οπότε για την αντιμετώπιση αυτής, έγινε χρήση σταθμισμένης συνάρτησης απώλειας, δηλαδή του καθορισμού των βαρών στον κώδικα αντιστρόφως ανάλογα με την ποσότητα των δεδομένων, εφόσον έχουμε τις κλάσεις:

- Kid (2286 εικόνες) : noKid (1144εικόνες) →2:1 (Διπλάσιες εικόνων παιδιών)

Και συνολικό αριθμό εικόνων στο σύνολο δεδομένων 3430 εικόνες, τότε τα βάρη καθορίζονται αντιστρόφως ανάλογα με την ποσότητα των εικόνων, σε αυτό το παράδειγμα:

- $Weight(kid) = 3430/2286 = 1,5$
- $Weight(noKid) = 3430/1144 = 3$

Με αυτό τον τρόπο κάθε πιθανό λάθος στην κλάση noKid έχει διπλάσιο αντίκτυπο από ότι στην κλάση kid, έτσι εξισορροπείται το πρόβλημα ανισορροπίας των κλάσεων κατά την εκπαίδευση.

4.4.3 Η διαδικασία της Εκπαίδευσης και οι Εποχές Εκπαίδευσης

4.4.3.1 Περιβάλλον Εκπαίδευσης και Τεχνικές Προδιαγραφές

Για την δυνατότητα επαναληψιμότητας της διαδικασίας παρατίθενται παρακάτω (Πίνακας 4-2) τα στοιχεία του περιβάλλοντος που δημιουργήθηκε το μοντέλο. Για την υλοποίηση του μοντέλου επίσης χρησιμοποιήθηκε και το πρόγραμμα Anaconda Navigator για την διαχείριση και εγκατάσταση Pytorch με υποστήριξη CUDA (Computer Unified Device Architecture), του εργαλείου που επιτρέπει να γίνει η εκπαίδευση του μοντέλου με χρήση των υπολογιστικών πόρων από την κάρτα γραφικών, επιτυγχάνοντας την μείωση του αναγκαίου χρόνου εκπαίδευσης και δίνοντας έτσι την δυνατότητα για παραπάνω πειραματισμούς και βελτίωση του μοντέλου.

Πίνακας 4-2: Περιβάλλον Εκπαίδευσης - Τεχνικές Προδιαγραφές

Υλικό		Λογισμικό	
Επεξεργαστής	AMD Ryzen 7 7435HS (3.10 GHz, 8 cores/16 threads)	Περιβάλλον Ανάπτυξης	Visual Studio Code με Python extensions
Κάρτα γραφικών	NVIDIA GeForce RTX 4060 Laptop GPU (8 GB GDDR6 VRAM)	Python Environment	Conda virtual environment για isolation
Μνήμη (RAM)	16.0 GB DDR5 RAM (4800 MT/s)	Operating System	Windows 11 με NVIDIA drivers
Αποθηκευτικός χώρος	245 GB NVMe SSD	CUDA Support	CUDA-enabled PyTorch για GPU acceleration

Κεφάλαιο 4ο:

Για την υποστήριξη των λειτουργιών του κώδικα και για την εκπαίδευση του μοντέλου εγκαταστάθηκαν οι βιβλιοθήκες του πίνακα (Πίνακας 4-3).

Πίνακας 4-3: Βιβλιοθήκες που χρησιμοποιήθηκαν

Βιβλιοθήκη	Χρήση
PyTorch	Το βασικό εργαλείο για την δημιουργία των μοντέλων που μπορούν να δουλέψουν στον IMX500. Πλαίσιο ανάπτυξης βαθιάς μάθησης. Ευκολία χρήσης και συνεργασία με επεξεργασία στην κάρτα γραφικών.
Torchvision	Βιβλιοθήκη με εκπαιδευμένα μοντέλα, επεξεργασίες εικόνων και εργαλεία για εργασίες μηχανικής μάθησης.
Scikit-learn	Παροχή έτοιμων υπολογισμών για την εξαγωγή μετρικών.
Tqdm	Δημιουργία μπάρας προόδου και εικονοστοιχείων για απεικόνιση της προόδου κατά την εκπαίδευση.
Torchinfo	Λεπτομερής αναφορά για την αρχιτεκτονική και δομή του μοντέλου που δημιουργήθηκε.
matplotlib	Δημιουργία γραφημάτων και οπτικοποίηση των αποτελεσμάτων.
NumPy	Μαθηματικές μετατροπές και διαχείριση δεδομένων για την ανάλυση στα αποτελέσματα.

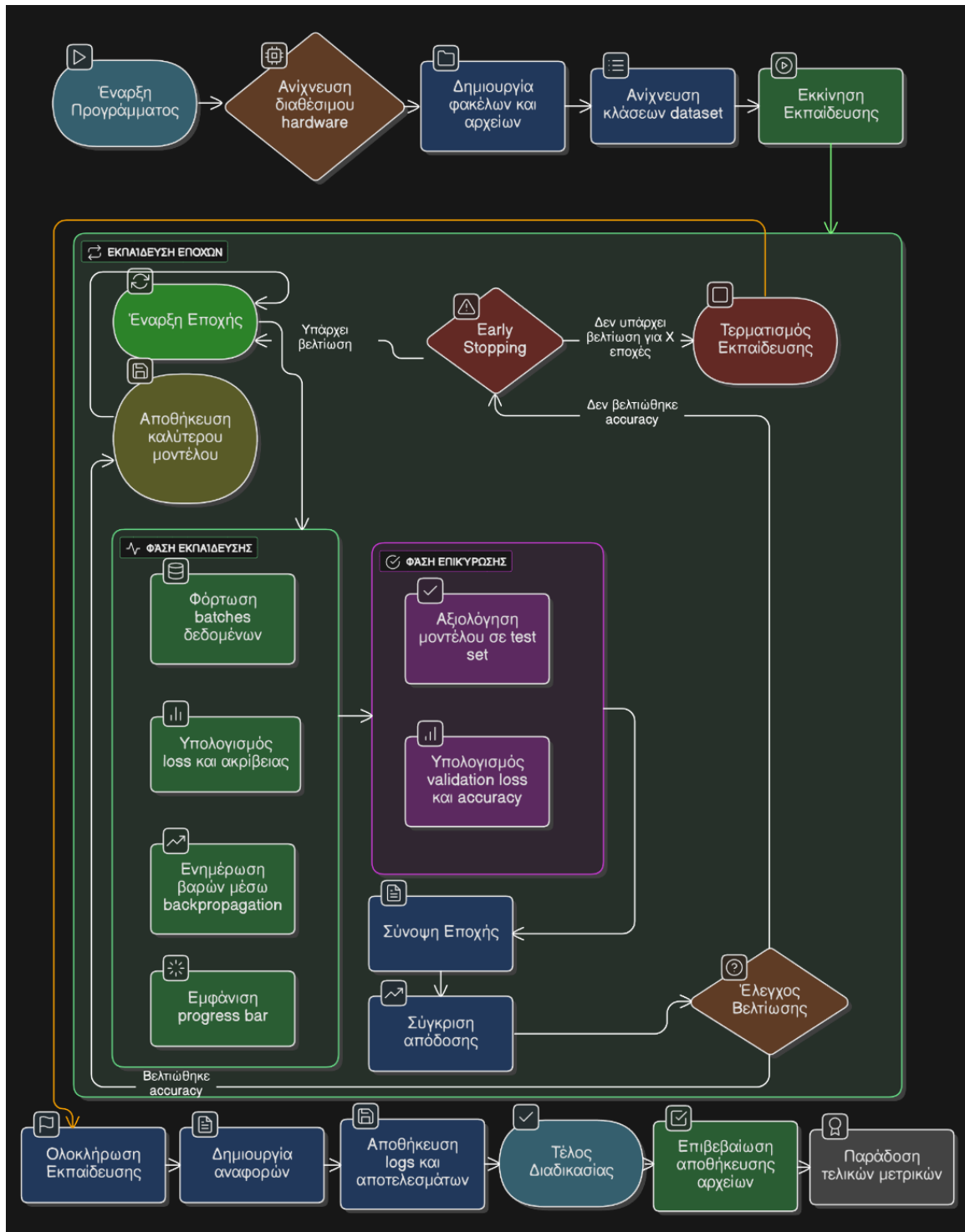
Τα μοντέλα που δημιουργήθηκαν ήταν 6, με διαφορετικές προσεγγίσεις και αποτελέσματα το κάθε ένα. Στην αρχή, χωρίς να βασίζονται σε ήδη εκπαιδευμένα μοντέλα, όμως αυτό για να αποφέρει ένα καλό αποτέλεσμα ήθελε έναν πολλαπλάσιο αριθμό εικόνων από τον διαθέσιμο στην βάση και αρκετό ελεύθερο χώρο. Το Sviro, για παράδειγμα, ως σύνολο δεδομένων καταλάμβανε περίπου 25 GB, βάσει αυτών, καθώς και του χρόνου που χρειάζεται η εκπαίδευση επιλέχθηκε η προσέγγιση που έχει αναλυθεί με το MobileNetV2.

4.4.3.2 Εκπαίδευση

Έχοντας το περιβάλλον εκτέλεσης έτοιμο και προσαρμοσμένο στις ανάγκες της εκπαίδευσης, και με την εκτέλεση του κώδικα εκπαίδευσης του μοντέλου από το αρχείο «train_kid_classifier_withMetricsSave.py», ξεκινάει η διαδικασία, με διακριτές τις ακόλουθες φάσεις:

- Εκκίνηση και προετοιμασία
 - Επιβεβαίωση της λειτουργίας του CUDA με εικονικό περιβάλλον εργασίας.
 - Δημιουργία αρχείων και καταλόγων, εφόσον δεν υπάρχουν.
 - Ανίχνευση αυτόματα των κλάσεων βάσει της δομής του συνόλου δεδομένων.
 - Αποθήκευση βασικών πληροφοριών για χρήση τους στην πορεία της εκπαίδευσης.
- Γίνεται η διαδικασία της εκπαίδευσης και παράλληλα εμφανίζει πληροφορίες
 - Αριθμός τρέχουσας εποχής.
 - Οπτική αναπαράσταση προόδου.
 - Λεπτομέρειες χρόνου.
 - Ταχύτητα επεξεργασίας δεδομένων .
 - Τρέχουσα τιμή σφάλματος (real time loss).
- Επικύρωση και σύνοψη εποχής
 - Μέση απώλεια και ακρίβεια στο σύνολο δεδομένων εκπαίδευσης.
 - Μέση απώλεια και ακρίβεια στο σύνολο δεδομένων δοκιμών.
 - Σύγκριση της απώλειας μεταξύ των συνόλων δεδομένων εκπαίδευσης και δοκιμών.

- Βέλτιστο μοντέλο
 - Σε κάθε εποχή γίνεται σύγκριση των μοντέλων που δημιουργούνται με το ήδη καλύτερο που έχει αποθηκευτεί. Εφόσον κάποιο νέο μοντέλο είναι καλύτερο, τότε αποθηκεύεται σε δυο μορφές. Επίσης γίνεται καταγραφή της βέλτιστης απόδοσης, ώστε να μπορέσουν να γίνουν στο τέλος οι συγκρίσεις.
- Πρόωρη διακοπή
 - Εφόσον παρατηρήθηκε ότι το μοντέλο δεν βελτιώνεται και έχει περάσει ο αριθμός της υπερπαραμέτροι, συγκεκριμένα 10 εποχές, τότε το μοντέλο σταματάει την εκπαίδευση.
- Κατά τη διάρκεια και μετά την εκπαίδευση δημιουργούνται τα εξής αρχεία:
 - Αρχεία Μοντέλου:
 - kid_mnv2_best.pt
 - kid_mnv2_best_full.pth: Πλήρες μοντέλο με αρχιτεκτονική
 - labels.txt: Λίστα κλάσεων
 - arxitektoniki_modelou.txt: Λεπτομερής δομή του μοντέλου
- Αρχεία Παρακολούθησης:
 - training_log.csv: Μετρικά κάθε εποχής (epoch, train_loss, train_acc, val_loss, val_acc)
 - y_true.csv: Πραγματικές ετικέτες (527 δείγματα)
 - y_pred.csv: Προβλέψεις μοντέλου για κάθε δείγμα
 - best_run_meta.json: Metadata καλύτερης εκτέλεσης με χρονοσήμανση
- Τελική σύνοψη με το πέρας όλων των εποχών ή με πρόωρο τερματισμό
 - Αναφορά που περιλαμβάνει:
 - Καλύτερη ακρίβεια
 - Μετρικά ανά κλάση
 - Πίνακα Σύγχυσης
 - Συνολικές μετρικές για την απόδοση



Εικόνα 4-4: Διάγραμμα ροής του κώδικα «train_kid_classifier_withMetricsSave.py»

4.4.4 Δημιουργία Διαγραμμάτων, Εξαγωγή Μετρικών και Ανάλυση του Τελικού Αποτελέσματος

Το μοντέλο που χρησιμοποιήθηκε χρειάστηκε περίπου 40 λεπτά για να φτάσει στην 19^η εποχή από τις 60 που ήταν το προκαθορισμένο όπου και σταμάτησε διότι για 10 εποχές δεν υπήρξε βελτίωση του μοντέλου. Η καλύτερη ακρίβεια επικύρωσης ήταν 89.4% στην 9^η εποχή. Αυτά παρουσιάζονται και στην Εικόνα 4-5, από το τερματικό στο Visual Studio.

```
n_kid_classifier_withMetricsSave.py
✓ Saved best model (acc=0.894) → C:\img_classifier\code\artifacts\kid_mnv2_best.pt
✓ Saved full model copy → C:\img_classifier\code\artifacts\kid_mnv2_best_full.pth
Epoch 10/60 | Train Loss: 0.0573 Acc: 0.978 | Val Loss: 0.7562 Acc: 0.786
Epoch 11/60 | Train Loss: 0.0559 Acc: 0.978 | Val Loss: 0.7320 Acc: 0.850
Epoch 12/60 | Train Loss: 0.0566 Acc: 0.974 | Val Loss: 1.0238 Acc: 0.733
Epoch 13/60 | Train Loss: 0.0437 Acc: 0.985 | Val Loss: 0.6814 Acc: 0.839
Epoch 14/60 | Train Loss: 0.0342 Acc: 0.988 | Val Loss: 0.5514 Acc: 0.866
Epoch 15/60 | Train Loss: 0.0252 Acc: 0.992 | Val Loss: 0.5807 Acc: 0.862
Epoch 16/60 | Train Loss: 0.0293 Acc: 0.987 | Val Loss: 0.7503 Acc: 0.826
Epoch 17/60 | Train Loss: 0.0424 Acc: 0.986 | Val Loss: 1.0072 Acc: 0.767
Epoch 18/60 | Train Loss: 0.0384 Acc: 0.985 | Val Loss: 0.7176 Acc: 0.839
Epoch 19/60 | Train Loss: 0.0320 Acc: 0.987 | Val Loss: 0.6936 Acc: 0.847
Early stopping at epoch 19 (no improvement in 10 epochs)

Best validation acc: 0.8939393939393939

Classification report:
              precision    recall  f1-score   support

     kid           0.81         0.88         0.84         248
    nokid           0.88         0.82         0.85         280

 accuracy           0.85
 macro avg           0.85         0.85         0.85         528
weighted avg           0.85         0.85         0.85         528

Confusion matrix:
[[217  31]
 [ 50 230]]

✓ Saved predictions to C:\img_classifier\code\artifacts\y_true.csv / C:\img_classifier\code\artifacts\y_pred.csv
✓ Saved metadata to C:\img_classifier\code\artifacts\best_run_meta.json

Artifacts written to: C:\img_classifier\code\artifacts
(pytorch_env) PS C:\img_classifier>
```

Εικόνα 4-5: Τερματικό – Αποτελέσματα Εκπαίδευσης

Τα παραπάνω επιβεβαιώνονται και με τα διαγράμματα. Για την εξαγωγή των διαγραμμάτων και των πληροφοριών εκπαίδευσης συνάχθηκε και εκτελέστηκε επιπλέον κώδικας, ο οποίος παρατίθεται στο παράρτημα [ΠΑΡΑΡΤΗΜΑ Β].

Πίνακας Σύγχυσης (Confusion Matrix) [Εικόνα 4-7]

Από τον πίνακα σύγχυσης φαίνεται η δυνατότητα του μοντέλου για αναγνώριση της παρουσίας παιδιού (Kid) με πιθανότητα 87.5% και της απουσίας (nokid) 82.1%, με 31 περιπτώσεις σφάλματος που δεν αναγνωρίστηκε το παιδί και 50 περιπτώσεις σφάλματος όπου φωτογραφία ενήλικα χαρακτηρίστηκε ως παιδί. Αποτέλεσμα αναμενόμενο και αποδεκτό για την υλοποίηση μιας πρώτης προσέγγισης.

Καμπύλες Απώλειας (Loss Curves) [Εικόνα 4-9]

Ενώ παρατηρείται η καμπύλη της εκπαίδευσης να παρουσιάζει σταθερή μείωση της απώλειας, υποδηλώνοντας αποτελεσματική μάθηση, η καμπύλη απώλειας της επικύρωσης εναλλάσσει τιμές με ιδιαίτερες διακυμάνσεις έχοντας μια γενική ανοδική πορεία, που είναι δείγμα υπερπροσαρμογής.

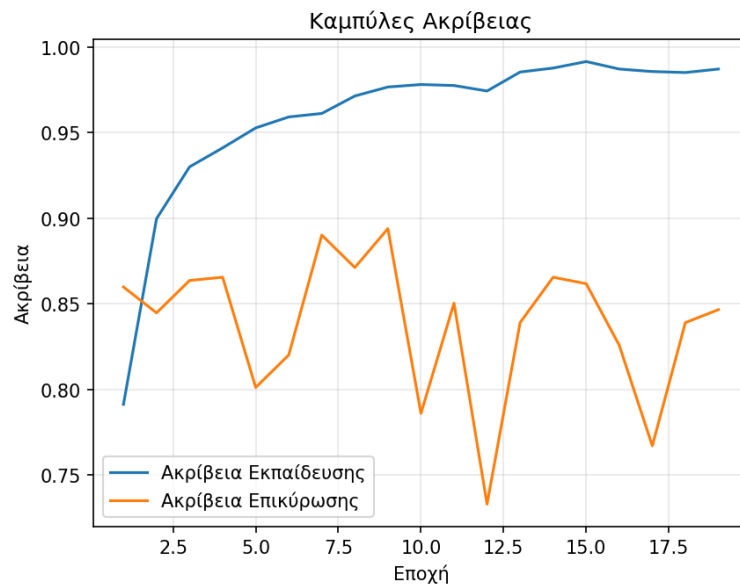
Καμπύλες Ακρίβειας (Accuracy Curves) [Εικόνα 4-6]

Κεφάλαιο 4ο:

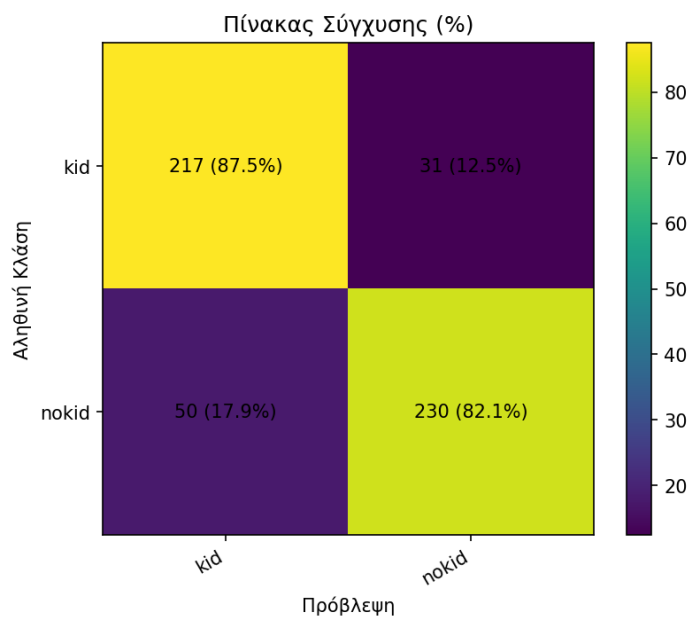
Με την καμπύλη ακρίβειας της εκπαίδευσης πολύ κοντά στο 99%, αλλά την καμπύλη ακρίβειας της επικύρωσης να είναι ασταθής και κοντά στο 89% , φαίνεται πως παρά που το μοντέλο έχει δυνατή ικανότητα μάθησης, δεν κάνει καλή γενίκευση στα δεδομένα που δεν γνωρίζει.

Διάγραμμα Καλύτερης Ακρίβειας Επικύρωσης (Best Validation Accuracy) [Εικόνα 4-8]

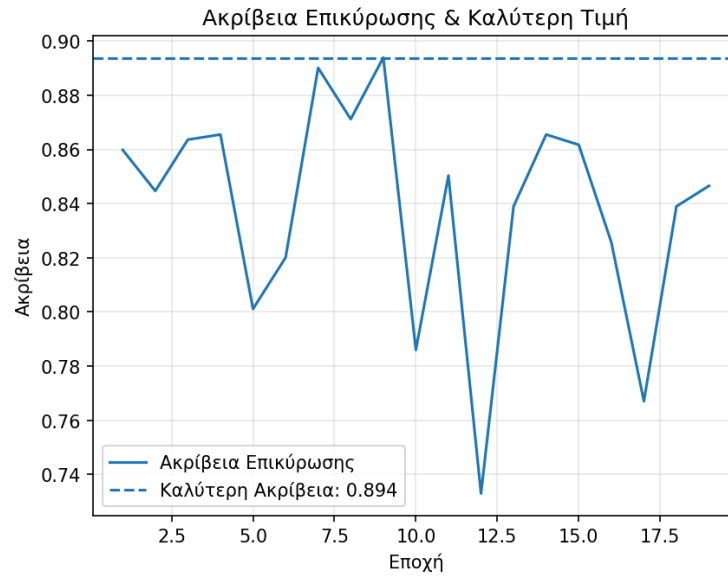
Όπως έχει αναφερθεί παραπάνω το μοντέλο επιτυγχάνει την καλύτερη ακρίβεια επικύρωσης στην εποχή 9, όπως φαίνεται και στην εικόνα με 89.4% και αποθήκευση του καλύτερου μοντέλου. Πλέον το μοντέλο έχει εξαχθεί και είναι έτοιμο για τα επόμενα βήματα.



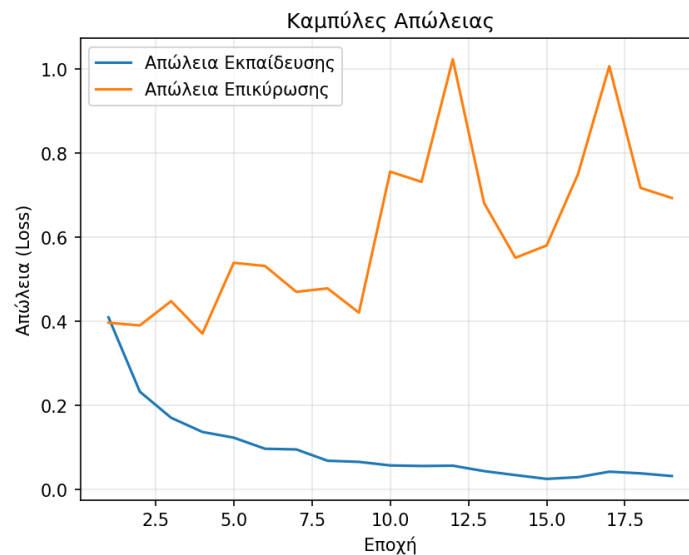
Εικόνα 4-6:Καμπύλες Ακρίβειας (Accuracy Curves)



Εικόνα 4-7: Πίνακας Σύγχυσης (Confusion Matrix)



Εικόνα 4-8: Διάγραμμα Καλύτερης Ακρίβειας Επικύρωσης (Best Validation Accuracy)



Εικόνα 4-9: Καμπύλες Απώλειας (Loss Curves)

4.5 Εξαγωγή ONNX

Με το μοντέλο να έχει δημιουργηθεί, είναι σειρά της εξαγωγής του σε μορφή ONNX για την συμβατότητα με το σύστημα τεχνητής νοημοσύνης παρυφής και την δυνατότητα να είναι συμβατό μετέπειτα με τον αισθητήρα της κάμερας τεχνητής νοημοσύνης. Το ONNX μοντέλο μπορεί να λειτουργήσει σε συσκευές χωρίς να επηρεάζεται από το πως και σε τι περιβάλλον έγινε η εκπαίδευση. Η διαδικασία για να εξαχθεί το ONNX είναι σημαντική και θέλει προσοχή όσον αφορά την δόμηση της αρχιτεκτονικής, διότι το εξαχθέν μοντέλο που θα εισάγουμε για μετατροπή έχει μόνο βάρη.

Ο κώδικας που δημιουργήθηκε αρχικά φτιάχνει ξανά την αρχιτεκτονική MobileNetV2 με ίδια κεφαλή εξόδου με την εκπαίδευση, κάνει φόρτωση των βαρών που έχουν αποθηκευτεί και τέλος ορίζει

Κεφάλαιο 4ο:

είναι εικονικό σήμα εισόδου, ώστε να λειτουργήσει το μοντέλο και να γίνει η καταγραφή-χαρτογράφηση του. Έπειτα γίνεται η εξαγωγή και αποθήκευση. Τα αποθηκευμένα αρχεία είναι 2 με την μορφοποίηση ONNX, το ένα με την χρονοσήμανση στο όνομα και το άλλο απλά «kid_mnv2.onnx». Το άλλο αρχείο που αποθηκεύεται είναι οι ετικέτες κλάσεων. Τέλος, μέσω του εργαλείου ONNX Checker γίνεται έλεγχος και εφόσον δεν είχε κάποιο πρόβλημα το μοντέλο, προχωράει η διαδικασία στο επόμενο βήμα, αυτό της Κβαντοποίησης και ανάπτυξης στον αισθητήρα IMX500. Ο κώδικας που δημιουργήθηκε εμφανίζεται παρακάτω:

```
import sys
import torch
import torch.nn as nn
from torchvision import models
from pathlib import Path
from datetime import datetime
import shutil
# Αρχεία
ARTIFACTS = Path(r"C:\img_classifier\code\artifacts")
CKPT = ARTIFACTS / "kid_mnv2_best.pt" # αρχείο με το εκπαιδευμένο
LABELS_TXT = ARTIFACTS / "labels.txt" # αποθήκευση labels

def main():
    print("== Export MobileNetV2 to ONNX ==")
    print(f"Artifacts dir: {ARTIFACTS}")
    print(f"Checkpoint path: {CKPT}")
    # Έλεγχος αν υπάρχει το checkpoint
    if not CKPT.exists():
        sys.exit(f"[ERROR] Checkpoint not found: {CKPT}\n"
                f"Make sure training saved it there or adjust CKPT.")
    # Φόρτωση checkpoint (περιέχει weights + labels)
    ckpt = torch.load(CKPT, map_location="cpu")
    if "model" not in ckpt or "labels" not in ckpt:
        sys.exit("[ERROR] Checkpoint missing keys 'model' or 'labels'.")
    labels = ckpt["labels"] # λίστα κλάσεις
    img_size = int(ckpt.get("img_size", 224))
    num_classes = len(labels)

    print(f"Loaded checkpoint. img_size={img_size}, classes={num_classes} ->
    {labels}")
    # Δημιουργία MobileNetV2
    model = models.mobilenet_v2(weights=None)
    model.classifier[1] = nn.Linear(model.last_channel, num_classes) #
προσαρμογή εξόδου
    model.load_state_dict(ckpt["model"]) # φόρτωση βαρών
    model.eval().to("cpu")

    dummy = torch.randn(1, 3, img_size, img_size, dtype=torch.float32)

    ARTIFACTS.mkdir(parents=True, exist_ok=True)
    # ονομα με timestamp
```

```

ts = datetime.now().strftime("%Y%m%d_%H%M%S")
onnx_timestamped = ARTIFACTS / f"kid_mnv2_{ts}.onnx"
onnx_latest      = ARTIFACTS / "kid_mnv2.onnx"
print(f"Exporting to ONNX (timestamped): {onnx_timestamped}")
torch.onnx.export(
    model,
    dummy,                    # input tensor
    str(onnx_timestamped),    # όνομα αρχείου εξόδου
    input_names=["images"],   # όνομα για input node
    output_names=["logits"],  # όνομα για output node
    opset_version=13,        # έκδοση ONNX opset
    do_constant_folding=True, # optimization
    dynamic_axes=None,       # σταθερό μέγεθος
    verbose=False,
)
# Δημιουργία αντιγράφου "kid_mnv2.onnx"
try:
    shutil.copy2(onnx_timestamped, onnx_latest)
    print(f"Also wrote latest alias -> {onnx_latest}")
except Exception as e:
    print(f"[WARN] Could not copy latest alias: {e}")
# Αποθήκευση labels
LABELS_TXT.write_text("\n".join(labels), encoding="utf-8")
print(f"Labels saved -> {LABELS_TXT}")
# Γρήγορος έλεγχος εγκυρότητας του ONNX
try:
    import onnx
    m = onnx.load(str(onnx_timestamped))
    onnx.checker.check_model(m)
    print("ONNX check (timestamped): OK")
except Exception as e:
    print(f"[WARN] Could not run onnx checker: {e}")
print("\nDone.")
print(f"Timestamped ONNX: {onnx_timestamped.resolve()}")
print(f"Latest ONNX      : {onnx_latest.resolve()}")
print(f"Labels           : {LABELS_TXT.resolve()}")

if __name__ == "__main__":
    main()

```

Κώδικας 4-2: Κώδικας εξαγωγής ONNX

4.6 Κβαντοποίηση Και Ανάπτυξη Στον IMX500(ΑΙ Κάμερα)

Για την προετοιμασία του μοντέλου για τον αισθητήρα Sony IMX500 ακολουθήθηκαν συγκεκριμένα βήματα τα οποία εκτελέστηκαν στον υπολογιστή (Laptop), ώστε να εκτελεστούν γρήγορα και εύκολα λόγω της επεξεργαστικής ισχύος, αλλά και στον μικροϋπολογιστή (Raspberry pi Zero 2W), ώστε το αρχείο να δημιουργηθεί στο κατάλληλο περιβάλλον, που ούτως ή άλλως θα κληθεί να λειτουργήσει στην συνέχεια .

Κεφάλαιο 4ο:

Αρχικά, έγινε η διαδικασία κβαντοποίησης με το σύνολο δεδομένων βαθμονόμησης (calibration dataset), ώστε να προκύψει το μοντέλο με μορφή ακεραίου από μορφή κινητής υποδιαστολής που ήταν. Η διαδικασία αυτή εκτελέστηκε μέσω της βιβλιοθήκης MCT (Model Compression Toolkit). Αυτό όπως ήδη αναφέρθηκε βοήθησε στην μείωση όγκου του μοντέλου και της ανάγκης του για υπολογιστική ισχύ κατά την διάρκεια της λειτουργίας του.

```
q_model = ptq.quantize_onnx(  
    onnx_model_path="kid_mnv2_static.onnx",  
    representative_data_dir="./calib",  
    config=cfg  
)  
onnx.save(q_model, "kid_mnv2_int8.onnx")
```

Κώδικας 4-3: Κώδικας κβαντοποίησης του μοντέλου ONNX σε μορφή int8 για τον αισθητήρα IMX500

Στο επόμενο βήμα, παραμένοντας στο περιβάλλον του υπολογιστή έγινε η μετατροπή του κβαντισμένου μοντέλου σε μια αναπαράσταση του μοντέλου που να μπορεί να εκτελεστεί από την μονάδα επεξεργασίας νευρωνικών δικτύων (NPU- Neural Processing Unit) του αισθητήρα. Αυτό έγινε με την βοήθεια του εργαλείου imx500-converter.

```
imx500-converter \  
--model kid_mnv2_int8.onnx \  
--input-nodes input \  
--output-nodes logits \  
--input-shape 1,3,224,224 \  
--mean 0.485,0.456,0.406 \  
--std 0.229,0.224,0.225 \  
--out-dir build_imx
```

Κώδικας 4-4: Εκτέλεση της εντολής imx500-converter για μετατροπή του εκπαιδευμένου μοντέλου σε μορφή συμβατή με τον αισθητήρα IMX500

Σε επόμενο βήμα, δημιουργήθηκε το πακέτο .grk με το εργαλείο imx500-packager, με σημαντική λεπτομέρεια πως αυτό το βήμα εκτελέστηκε στον μικροϋπολογιστή απευθείας .

Τέλος φορτώθηκε στον imx-500 και εκτελέστηκε το πακέτο .grk με την χρήση πάλι του μικροϋπολογιστή ως μέσο διαχείρισης και επικοινωνίας με τον αισθητήρα.

Μετά το βήμα αυτό, ο αισθητήρας έτρεχε το μοντέλο κατηγοριοποίησης των εικόνων, οπότε ήταν έτοιμος για την ενσωμάτωση με τα υπόλοιπα συστήματα ώστε να συντεθεί το τελικό σύστημα με την δυνατότητα να λαμβάνει τις πληροφορίες από τον αισθητήρα και ανάλογα με τα σενάρια που θεσπίστηκαν να προβαίνει και στις αντίστοιχες ενέργειες.

4.7 Επίλογος

Το κεφάλαιο αυτό περιλάμβανε την συνολική διαδικασία σύνθεσης του συνελκτικού μοντέλου τεχνητής νοημοσύνης μέχρι και την εγκατάσταση του στον αισθητήρα IMX500. Αναλύθηκαν τα

βήματα που ακολουθήθηκαν για την σχεδίαση και σύνθεση των συνόλων δεδομένων που χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου. Αναλύθηκε η αρχιτεκτονική της σύνθεσης του μοντέλου και ο κώδικας, με ανάλυση στοιχείων του, όπως οι υπερπαραμέτροι και οι εποχές εκπαίδευσης. Έγινε επίσης η παρουσίαση των δεδομένων που λήφθηκαν κατά και με το πέρασμα της εκπαίδευσης, δηλαδή των μετρικών για την αξιολόγηση της απόδοσης.

Ακολούθως, έγινε αναφορά της διαδικασίας εξαγωγής του μοντέλου σε ONNX μορφή, έτσι ώστε να μπορεί να συνεργαστεί με τα εργαλεία ανάπτυξης που προσφέρει η Sony. Συνεχίζοντας, έγινε κβαντοποίηση του μοντέλου ώστε να προσαρμοστεί στις λειτουργικές απαιτήσεις του αισθητήρα και η τελική μετατροπή σε πακέτο .gk. Τέλος το πακέτο φορτώθηκε και εκτελέστηκε στον αισθητήρα με επιτυχία.

Αυτή η διαδικασία ήταν η δημιουργία ουσιαστικά ενός εκ των υποσυστημάτων που δομούν την συνολική πρόταση για την παρακολούθηση της παρουσίας παιδιού σε παιδικό κάθισμα αυτοκινήτου.

Στο επόμενο κεφάλαιο θα γίνει η παρουσίαση της υλοποίησης του υποσυστήματος του μικροϋπολογιστή με τα περιφερειακά του (Raspberry Pi + αισθητήρες και τροφοδοσία μπαταρίας) και του διακομιστή BLE. Αυτά αποτελούν τον συνδετικό κρίκο ανάμεσα στο μοντέλο τεχνητής νοημοσύνης και την εφαρμογή του τελικού χρήστη. Γίνεται ανάλυση της αυτόματης εκκίνησης λειτουργιών, του τρόπου διαχείρισης των αισθητήρων θερμοκρασίας και υγρασίας, της επικοινωνία μέσω BLE, καθώς και η διαδικασία αποστολής ειδοποιήσεων προς την κινητή συσκευή. Με τον τρόπο αυτό, το σύστημα αποκτά πρακτική λειτουργικότητα και δυνατότητα άμεσης αλληλεπίδρασης με τον χρήστη.

Κεφάλαιο 5ο: Υλοποίηση Υποσυστήματος Raspberry Pi και Διακομιστή BLE

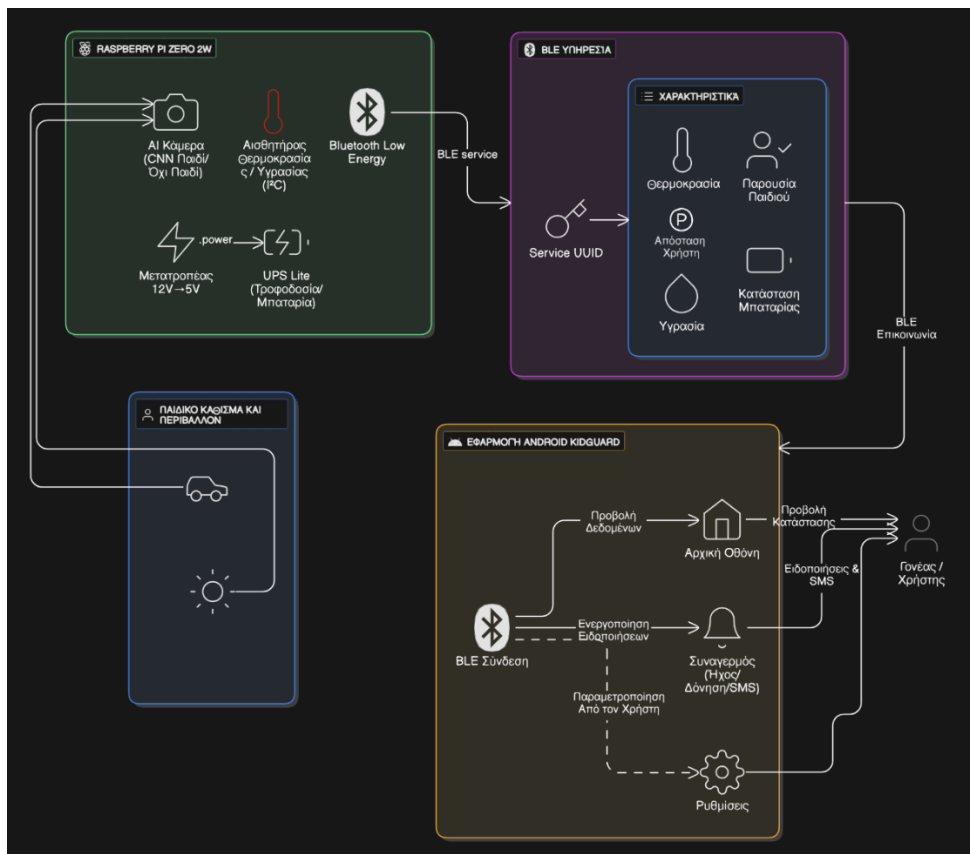
5.1 Εισαγωγή

Με το υποσύστημα δυαδικής κατηγοριοποίησης τεχνητής νοημοσύνης στις παρυφές, ήδη υλοποιημένο στην κάμερα, οι επόμενες κινήσεις είναι η αξιοποίηση των δεδομένων που αυτή παράγει, και μαζί με συλλογή επιπλέον δεδομένων, η αποστολή τους για τελική επεξεργασία, ανάλυση και ενέργειες βάσει του προγραμματισμού.

Πλέον έπρεπε να υλοποιηθεί η ρουτίνα διαχείρισης αυτών των δεδομένων, να επικοινωνήσουν και οι άλλοι αισθητήρες με τον μικροϋπολογιστή και τελικώς να καταστεί εφικτή η μετάδοση πληροφορίας στην εφαρμογή Android. Εκτός αυτού, η ρουτίνα αυτή πρέπει να εκκινείται αυτόματα σε κάθε εκκίνηση του λογισμικού του μικροϋπολογιστή οπότε έπρεπε να δημιουργηθεί και η υπηρεσία αυτόματης εκτέλεσης που αναλαμβάνει αυτό τον ρόλο.

5.2 Περιγραφή του Υποσυστήματος

Ο μικροϋπολογιστής (Raspberry Pi Zero 2W) είναι ο διαμεσολαβητής ανάμεσα στους αισθητήρες και στην ροή πληροφορίας προς την εφαρμογή Android.



Εικόνα 5-1: Διάγραμμα ενότητων του υποσυστήματος

Το υποσύστημα αποτελείται από:

- Raspberry Pi Zero 2W: Κεντρικός μικροϋπολογιστής.
- Προσαρμογέα BLE: Για επικοινωνία με την εφαρμογή Android.
- AI Camera (IMX500): Κάμερα για αναγνώριση παιδιών με χρήση TN.
- AHT20 Sensor: Αισθητήρας θερμοκρασίας και υγρασίας.
- UPS Lite v1.3: Μονάδα τροφοδοσίας με μπαταρία και δυνατότητα παρακολούθησης της μπαταρίας.
- IR LEDs: Υπέρυθρες δίοδοι για ενίσχυση φωτός για νυχτερινή λήψη.

Ροή Δεδομένων

Η ροή της πληροφορίας στο υποσύστημα με την έναρξή του ξεκινάει από τους αισθητήρες. Αρχικά η κάμερα με το μοντέλο δυαδικής ταξινόμησης αναλύει συνεχώς εικόνες και παράγει δεδομένα πρόβλεψης για την ύπαρξη ή όχι παιδιού στις αντίστοιχες εικόνες. Παράλληλα, μετράται συνεχώς η θερμοκρασία, η υγρασία και η στάθμη της μπαταρίας. Όλα αυτά τα δεδομένα συλλέγονται από τον μικροϋπολογιστή, επεξεργάζονται και διατίθενται μέσω της ασύρματης επικοινωνίας BLE για να τις διαβάσει η εφαρμογή Android

5.3 Αυτόματη Εκκίνηση Λειτουργιών

Για την ομαλή εκτέλεση όλων των απαιτούμενων λειτουργιών και την αυτοματοποίηση της έναρξης και λειτουργίας του υποσυστήματος δημιουργήθηκε και ενεργοποιήθηκε ο κώδικας που αναλύεται παρακάτω.

5.3.1 Δεδομένα παρουσίας παιδιού

Μέσω της υπηρεσίας kid-classifier@<user> γίνεται η εκτέλεση της ταξινόμησης των εισερχόμενων στον αισθητήρα εικόνων και τα αποτελέσματα αυτής γράφονται στο αρχείο kid_status.json. Τέθηκε ένα περιθώριο 90 δευτερολέπτων από την έναρξη λειτουργίας, για την διασφάλιση της ορθής λήψης των έγκυρων ταξινομήσεων και όχι ταξινομήσεων από προηγούμενη λειτουργία.

```
def ensure_classifier_running(user=KID_USER, timeout=90):
    """Start kid-classifier@<user> if not active and wait for fresh JSON."""
    try:
        r = subprocess.run(
            ["systemctl", "is-active", f"kid-classifier@{user}"],
            stdout=subprocess.PIPE, stderr=subprocess.DEVNULL, text=True
        )
        if r.stdout.strip() != "active":
            subprocess.run(["systemctl", "start", f"kid-classifier@{user}"],
                check=False)
```

Κώδικας 5-1: Συνάρτηση ensure_classifier_running για εκκίνηση του kid-classifier

Η παρακάτω συνάρτηση διαβάζει τα δεδομένα από το αρχείο JSON και στην συνέχεια προχωράει στην απόδοση τιμής «1» εφόσον η τιμή πρόβλεψης για παρουσία παιδιού (kid) είναι μεγαλύτερη ή ίση με 0.5, ειδάλως αποδίδει την τιμή 0 υποδηλώνοντας την μη παρουσία παιδιού (noKid).

Κεφάλαιο 5ο:

- $Confidence_{kid} \geq 50\% \rightarrow Παρουσία_{Παιδιου} = 1$
- $Confidence_{kid} < 50\% \rightarrow Παρουσία_{Παιδιου} = 0$

```
def read_kid_from_json(path=STATUS_PATH, threshold=0.5):
    """Return 1 if top label is 'kid' with score >= threshold, else 0."""
    try:
        with open(path, "r") as f:
            data = json.load(f)
            top_list = data.get("top", [])
            if not top_list:
                return 0
            top = max(top_list, key=lambda x: float(x.get("score", 0)))
            label = str(top.get("label", "")).strip().lower()
            score = float(top.get("score", 0))
            return 1 if (label == "kid" and score >= threshold) else 0
```

Κώδικας 5-2: Συνάρτηση read_kid_from_json για ανίχνευση παρουσίας παιδιού

5.3.2 Διαχείριση Αισθητήρα Θερμοκρασίας Υγρασίας AHT20 και Παρακολούθηση Στάθμης Μπαταρίας

Η πλακέτα UPS_LiteV1.3 κάνει χρήση του ολοκληρωμένου CW2015 fuel gauge ,ώστε να μπορεί να γίνεται παρακολούθηση της στάθμης της μπαταρίας μέσω I2C πρωτοκόλλου.

Η σύνδεση του αισθητήρα AHT20 επίσης γίνεται μέσω I2C και κάθε μέτρηση επιστρέφεται ως τιμή μορφής float32. Επίσης ο αισθητήρας αρχικοποιείται στο σημείο του προγράμματος που γίνεται χρήση του και όχι στην αρχή για ευκολότερη αποσφαλμάτωση και αποφυγή σταματήματος του προγράμματος σε περίπτωση προβλήματος με τον αισθητήρα.

Η επαλήθευση σύνδεσης έγινε της με την εντολή «sudo i2cdetect -y 1», η οποία επιστρέφει τις διευθύνσεις:

- 0x38: AHT20 → Για θερμοκρασία και υγρασία.
- 0x62: CW2015 → Για παρακολούθηση μπαταρίας.

```
def get_sensor():
    global _sensor
    if _sensor is None:
        try:
            import board
            import adafruit_ahtx0
            import busio
            i2c = busio.I2C(board.SCL, board.SDA)
            _sensor = adafruit_ahtx0.AHTx0(i2c)
            print("[BLE] AHTx0 sensor initialized")
        except Exception as e:
            print("[BLE] AHTx0 not available:", e)
            _sensor = None
    return _sensor
```

Κώδικας 5-3: Συνάρτηση get_sensor για αρχικοποίηση αισθητήρα AHT20

```
def read_battery_percent():
    """Return battery state of charge as a float percent (0..100)."""
    if not SMBUS_OK:
        return 0.0
    try:
        with SMBus(I2C_BUS) as bus:
            b = bus.read_i2c_block_data(UPS_ADDR, REG_SOC, 2) # [hi, lo] Q8.8
            soc = b[0] + b[1] / 256.0
            if soc < 0.0: soc = 0.0
            if soc > 100.0: soc = 100.0
            return float(soc)
    except Exception as e:
        print("[BLE] Battery read error:", e)
        return 0.0
```

Κώδικας 5-4: Συνάρτηση read_battery_percent για ανάγνωση στάθμης μπαταρίας

5.3.3 Διαχείριση Υπέρυθρων LED

Η λειτουργία των υπέρυθρων διόδων είναι αυτόνομη, με παροχή τροφοδοσίας από τον μικροϋπολογιστή, αλλά ανεξάρτητη λειτουργία σύμφωνα με τις συνθήκες φωτισμού, αξιοποιώντας την φωτοδίοδο που υπάρχει ενσωματωμένη. Έτσι εξασφαλίζεται σωστός φωτισμός, χωρίς να χρειάζεται περαιτέρω έλεγχος από τον μικροϋπολογιστή.

5.3.4 Επικοινωνία BLE

5.3.4.1 Χαρακτηριστικά BLE

Για την επικοινωνία υλοποιήθηκε BLE GATT εξυπηρετητής με τα παρακάτω χαρακτηριστικά :

- Service UUID: 12345678-1234-5678-1234-56789abcdef0
- Characteristics:

Κεφάλαιο 5ο:

- Kid→UUID: 12345678-1234-5678-1234-56789abcdef1
- uint8 (0/1)
- Temp→UUID: 12345678-1234-5678-1234-56789abcdef2
- float32 LE (C)
- Hum→UUID: 12345678-1234-5678-1234-56789abcdef3
- float32 LE (%RH)
- Battery→UUID: 12345678-1234-5678-1234-56789abcdef4
- uint8 (0...100)

Κάθε χαρακτηριστικό υλοποιείται στον κώδικα ως ξεχωριστή κλάση:

```
class KidPresenceCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_KID_PATH, CHAR_KID_UUID)

    @dbus.service.method("org.bluez.GattCharacteristic1", in_signature="a{sv}",
out_signature="ay")
    def ReadValue(self, options):
        val = read_kid_from_json() # 0 or 1
        print("[BLE] ReadValue kid ->", val)
        return [dbus.Byte(val)]
```

Κώδικας 5-5: Κλάση KidPresenceCharacteristic για ανάγνωση παρουσίας παιδιού μέσω BLE

```
class BatteryLevelCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_BATT_PATH, CHAR_BATT_UUID)

    @dbus.service.method("org.bluez.GattCharacteristic1", in_signature="a{sv}",
out_signature="ay")
    def ReadValue(self, options):
        pct = read_battery_percent() # float 0..100
        val = int(round(max(0.0, min(100.0, pct)))) # uint8 0..100
        print("[BLE] ReadValue battery(u8) -> %d%%" % val)
        return [dbus.Byte(val)]
```

Κώδικας 5-6: Κλάση BatteryLevelCharacteristic για ανάγνωση στάθμης μπαταρίας μέσω BLE

```

class TemperatureCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_TEMP_PATH, CHAR_TEMP_UUID)

        @dbus.service.method("org.bluez.GattCharacteristic1", in_signature="a{sv}",
out_signature="ay")
        def ReadValue(self, options):
            s = get_sensor()
            temp = 0.0
            if s is not None:
                try:
                    temp = float(s.temperature)
                except Exception:
                    temp = 0.0
            print("[BLE] ReadValue temp -> %.2f C" % temp)
            return [dbus.Byte(b) for b in struct.pack("<f", temp)]
    
```

Κώδικας 5-7: Κλάση TemperatureCharacteristic για ανάγνωση θερμοκρασίας μέσω BLE

```

class HumidityCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_HUM_PATH, CHAR_HUM_UUID)

        @dbus.service.method("org.bluez.GattCharacteristic1", in_signature="a{sv}",
out_signature="ay")
        def ReadValue(self, options):
            s = get_sensor()
            hum = 0.0
            if s is not None:
                try:
                    hum = float(s.relative_humidity)
                except Exception:
                    hum = 0.0
            print("[BLE] ReadValue hum -> %.2f percent" % hum)
            return [dbus.Byte(b) for b in struct.pack("<f", hum)]
    
```

Κώδικας 5-8: Κλάση HumidityCharacteristic για ανάγνωση υγρασίας μέσω BLE

5.3.4.2 Διαχείριση Προσαρμογέα BLE

Έγινε ρύθμιση ώστε ο μικροϋπολογιστής να είναι ανιχνεύσιμος ως BLE συσκευή και να εκθέτει την υπηρεσία «KidCarGuard»

```
def setup_adapter(bus):
    props = dbus.Interface(bus.get_object(BLUEZ_SERVICE, ADAPTER_PATH),
"org.freedesktop.DBus.Properties")
    props.Set(ADAPTER_IFACE, "Powered", dbus.Boolean(True))
    props.Set(ADAPTER_IFACE, "Discoverable", dbus.Boolean(True))
    props.Set(ADAPTER_IFACE, "Pairable", dbus.Boolean(True))
    props.Set(ADAPTER_IFACE, "Alias", dbus.String("KidCarGuard"))
```

Κώδικας 5-9: Συνάρτηση setup_adapter για ρυθμίσεις BLE αντάπτορα

5.3.5 Αυτόματη Εκκίνηση Υπηρεσίας

Για την ασφαλή εκκίνηση του υποσυστήματος του Raspberry Pi με όλα τα περιφερειακά του και εκκίνηση των λειτουργιών που έχουν δημιουργηθεί, έγινε παραμετροποίηση της υπηρεσίας του συστήματος. Το αρχείο παραμετροποίησης δημιουργήθηκε στο:

«/etc/systemd/system/kidcarguard.service».

Για την διασφάλιση της διαθεσιμότητας όλων των υπηρεσιών(BLE, I2C) και της ολοκληρωμένης εκκίνησης των υπηρεσιών του συστήματος του μικροϋπολογιστή χωρίς να υπάρχουν διενέξεις, προστέθηκε καθυστέρηση εκτέλεσης 10 δευτερολέπτων. Σε περίπτωση σφάλματος γίνεται επανεκκίνηση της υπηρεσίας.

Επίσης όλες οι καταγραφές κάνουν εγγραφή και στο systemd journal, για πιθανή χρήση για αποσφαλμάτωση. Με τις παρακάτω εντολές μπόρεσε να γίνει παρακολούθηση και αποσφαλμάτωση κατά την διάρκεια των δοκιμών.

- Προβολή logs:
 - sudo journalctl -u kidcarguard.service -f
- Προβολή τελευταίων 50 γραμμών των καταγραφών:
 - sudo journalctl -u kidcarguard.service -n
- Επανεκκίνηση της υπηρεσίας:
 - sudo systemctl restart kidcarguard.service

Για την ενεργοποίηση της αυτόματης εκκίνησης της υπηρεσίας, εκτελέστηκαν οι παρακάτω εντολές:

- Φόρτωση του service file:
 - sudo systemctl daemon-reload
- Ενεργοποίηση για αυτόματη εκκίνηση:
 - sudo systemctl enable kidcarguard.service
- Άμεση εκκίνηση της υπηρεσίας:
 - sudo systemctl start kidcarguard.service
- Έλεγχος κατάστασης:
 - sudo systemctl status kidcarguard.service

```

[Unit]
Description=KidCarGuard BLE Service
After=network.target bluetooth.service
Wants=bluetooth.service
Requires=network.target

[Service]
Type=simple
User=root
Environment=KID_USER=michalis
Environment=START_DELAY_SEC=10
ExecStartPre=/bin/sleep 5
ExecStart=/usr/bin/python3 /opt/kidcarguard/test.py
Restart=always
RestartSec=10
StandardOutput=journal
StandardError=journal

[Install]
WantedBy=multi-user.target

```

Κώδικας 5-10: Αρχείο μονάδας systemd που χρησιμοποιείται για την αυτόματη εκκίνηση και διαχείριση της υπηρεσίας KidCarGuard BLE κατά την εκκίνηση του συστήματος

5.4 Επίλογος

Το υποσύστημα με τον μικροϋπολογιστή, κάμερα, αισθητήρες και το όλο λογισμικό, είναι ουσιαστικά το υποσύστημα ανίχνευσης παρουσίας παιδιού σε παιδικό κάθισμα αυτοκινήτου, χωρίς όμως την δυνατότητα ειδοποιήσεων και προβολής των δεδομένων στον χρήστη.

Η υλοποίησή του παρέχει πλέον:

- Αναγνώριση παιδιού μέσω της δυαδικής ταξινόμησης εικόνων με την χρήση τεχνητής νοημοσύνης παρυφής.
- Συνεχή παρακολούθηση θερμοκρασίας και υγρασίας.
- Δυνατότητα ασύρματης επικοινωνίας.
- Αυτόματη εκκίνηση και λειτουργία χωρίς ανθρώπινη παρέμβαση.
- Παρακολούθηση της στάθμης μπαταρίας και διαχείριση ενέργειας.

Με την ολοκλήρωση του υποσυστήματος αυτού, απομένει η ανάπτυξη της εφαρμογής Android η οποία θα λειτουργεί ως διεπαφή με τον χρήστη, παρέχοντας ειδοποιήσεις σε πραγματικό χρόνο, δυνατότητες ρύθμισης και ελέγχου του συστήματος, καθώς και σενάρια συναγερμού. Στο επόμενο κεφάλαιο παρουσιάζεται αναλυτικά η αρχιτεκτονική της εφαρμογής, οι τεχνολογίες που επιλέχθηκαν, η υλοποίηση του πελάτη BLE, καθώς και ο σχεδιασμός των οθονών χρήστη.

Κεφάλαιο 6ο: Ανάπτυξη Εφαρμογής Android και Ενσωμάτωση Επικοινωνίας BLE

6.1 Εισαγωγή

Η εφαρμογή Android «KidGuard» που υλοποιήθηκε, είναι το μέσο διαχείρισης και εποπτείας του συστήματος ανίχνευσης παρουσίας παιδιού στο παιδικό κάθισμα αυτοκινήτου. Αυτό σημαίνει πως οι αποφάσεις, βάσει των δεδομένων που διαβάζει από τον μικροϋπολογιστή, λαμβάνονται εντός του κώδικα αυτής, είτε είναι προβολή μηνυμάτων είτε ειδοποιήσεις και μηνύματα. Η εφαρμογή σχεδιάστηκε με την λογική πως κάθε αστοχία που μπορεί να επιφέρει πιθανή παρέκκλιση του σκοπού αυτής, και κατά περιπτώσεις να θέσει σε κίνδυνο τη ζωή παιδιού, να έχει προβλεφθεί και ανάλογα την κρισιμότητα είτε να αφήνεται χρονικό περιθώριο ως αναμονή για επαναφορά σε κανονική λειτουργία, είτε να γίνεται γνωστοποίηση στον χρήστη. Σε περίπτωση που για κάποιο λόγο το σύστημα δεν μπορεί να ανταπεξέλθει σε κάποια συνθήκη, γίνεται ενημέρωση, με αναμονή επιβεβαίωσης από τον χρήστη ότι έλαβε γνώση, ειδάλλως αποστέλλεται μήνυμα κειμένου σε 2 καθορισμένων από τον χρήστη επαφών με τοποθεσία.

Η εφαρμογή λειτουργεί ως δικλίδα ασφαλείας ενεργώντας στο παρασκήνιο των μετακινήσεων με οχήματα.



Εικόνα 6-1: Προσομοίωση κινητού με την αρχική οθόνη της εφαρμογής

6.2 Σχεδιασμός Αρχιτεκτονικής Εφαρμογής Android

6.2.1 Επιλογή Τεχνολογιών

Η απαιτήσεις του συστήματος για αμεσότητα και ασφάλεια έδωσαν και κατευθυντήριες για την προσέγγιση της υλοποίησης της εφαρμογής.

Η **γλώσσα προγραμματισμού** της εφαρμογής είναι η Kotlin, προσφέροντας ασφάλεια έναντι πιθανών σφαλμάτων και κατά συνέπεια αστοχία και τερματισμό εφαρμογής. Η ασφάλεια μηδενικών τιμών που προσφέρει η Kotlin έναντι της Java, διασφαλίζει πως ακόμα και αν μια μεταβλητή δεν λάβει τιμή, το πρόγραμμα μπορεί να συνεχίσει την εκτέλεση.

Το **πλαίσιο διεπαφής χρήστη** (UI) είναι δομημένο με Jetpack compose, αντί για xml με άμεση ενημέρωση των στοιχείων στην οθόνη βάσει των δεδομένων που λαμβάνονται, καλύτερη προεπισκόπηση αλλαγών χωρίς να πρέπει να τρέξει η εφαρμογή και λιγότερο κώδικα, αν και προέκυψε αρκετός κώδικας για την εκτέλεση όλων των λειτουργιών.

Διαχείριση κατάστασης (state flow, state management) για την διαχείριση των δεδομένων σε πραγματικό χρόνο, ώστε να διατηρούνται οι πιο πρόσφατες τιμές. Ενσωμάτωση με συνεκτελεστικές ρουτίνες (coroutines) που μπορούν να λειτουργούν παράλληλα χωρίς να γίνεται μπλοκάρισμα της φυσικής ροής των διεργασιών, με κυριότερο πλεονέκτημα τις άμεσες αποκρίσεις.

```
private val _temperature = MutableStateFlow(0f)
val temperature: StateFlow<float> = _temperature
```

Κώδικας 6-1: Μεταβλητή και ροή κατάστασης θερμοκρασίας

Αποθήκευση δεδομένων (data storage): Για την αποθήκευση σημαντικών στοιχείων από τις ρυθμίσεις χρήστη, έγινε χρήση του DataStore, ώστε να μην υπάρξει πιθανή αλλοίωση.

6.2.2 Αρχιτεκτονική MVVM (Model-View-View-Model)

Διατηρώντας ανεξάρτητα μεταξύ τους την λειτουργία, με τον αλγόριθμο για τις ειδοποιήσεις από την διεπαφή χρήστη εξασφαλίστηκε η δυνατότητα να γίνει πιο σωστή οργάνωση, πιο δομημένη διαχείριση των εισερχόμενων δεδομένων και πιο ξεκάθαρος κώδικας.

Επίπεδο ViewModel: Στο BleScannerViewModel γίνεται η διαχείριση για την σύνδεση και λειτουργία BLE καθώς και η διαχείριση των αλγορίθμων ασφαλείας. Με αυτόν τον τρόπο συγκεντρώνεται η διαχείριση των παραμέτρων και η εκτίμησή τους ώστε να προκύπτουν τελικώς σωστές ενέργειες.

Επίπεδο προβολής (View): Στις συναρτήσεις composable έχουμε άμεση αντίδραση βάση της εναλλαγής των τιμών των μεταβλητών καθιστώντας την προβολή πάντα ενημερωμένη και έτοιμη για τον χρήστη με τα νεότερα δεδομένα.

```
class BleScannerViewModel(application: Application) :
    AndroidViewModel(application) {
    // Διαχείριση πολλαπλών καταστάσεων
    private val _connectedDevice = MutableStateFlow<BluetoothDevice?>(null)
    private val _kidPresence = MutableStateFlow("Unknown")
    private val _temperature = MutableStateFlow(0f)
}
```

Κώδικας 6-2: Κλάση BleScannerViewModel για διαχείριση καταστάσεων BLE

6.2.3 Διαχείριση Κατάστασης

Η διαχείριση κατάστασης καλείται να διαχειριστεί όλο τον συγχρονισμό και την οργάνωση της ορθής λειτουργίας της συσκευής, με στοιχεία προς διαχείριση τα παρακάτω:

Κεφάλαιο 6ο:

- Την κατάσταση της διασύνδεσης BLE, δηλαδή αν έχει συνδεθεί ή αποσυνδεθεί η συσκευή, ή αν υπάρχει κάποιο σφάλμα στην επικοινωνία.
- Τα δεδομένα που διαβάζονται από κάθε αισθητήρα του συστήματος.
- Την παρουσία του παιδιού.
- Την κατάσταση συναγερμού – ειδοποιήσεων.

Με κάθε νέα σύνδεση επαναφέρονται οι τιμές από σημαίες λογικής (flags) σε false , με ταυτόχρονη αποθήκευση και του χρόνου οπου αυτή η διαδικασία υλοποιήθηκε. Έτσι καταφέρνει να γίνει παρακολούθηση και αντίδραση από το σύστημα όποτε μόνο αυτό χρειάζεται και υπάρχει ξεκάθαρη στιγμή εκκίνησης και κατάστασης.

```
LaunchedEffect(device) {
    if (device != null) {
        connectStartMillis = System.currentTimeMillis()
        lowBatteryNotified = false
        disconnectionTriggered = false
    }
}
```

Κώδικας 6-3: Επανεκκίνηση παραμέτρων κατά τη σύνδεση συσκευής

6.3 Υλοποίηση Λειτουργιών Πελάτη BLE (BLE Client)

6.3.1 Σάρωση και Ανακάλυψη BLE

Για την σάρωση, επιλέχθηκε να γίνεται με ταχύτητα για 5 δευτερόλεπτα, χρόνο επαρκή για την εύρεση της συσκευής όπως και επίσης να γίνεται αυτόματη επανασύνδεση σε περίπτωση αποσύνδεσης για οποιανδήποτε λόγο. Είναι κρίσιμης σημασίας να μπορεί να γίνει **επανασύνδεση** στην τελευταία συνδεδεμένη συσκευή γιατί σε περίπτωση διακοπής σύνδεσης δεν υπάρχει ανατροφοδότηση δεδομένων.

```
private fun autoConnectToLastDevice() {
    viewModelScope.launch {
        LastDeviceStore.flow(getApplication()).collectLatest { address ->
            if (address != null && _connectedDevice.value == null) {
                startScan()
                // Αναζήτηση συγκεκριμένης συσκευής
            }
        }
    }
}
```

Κώδικας 6-4: Συνάρτηση αυτόματης επανασύνδεσης σε τελευταία συσκευή

6.3.2 Υλοποίηση Πελάτη GATT (GATT Client)

Ο πελάτης γενικού προφίλ χαρακτηριστικών (GATT) είναι ο πυρήνας για την αναμετάδοση και επικοινωνία της εφαρμογής με τον μικροϋπολογιστή και βασίζεται στην αυτόματη διόρθωση των σφαλμάτων.

Διαχείριση κύκλου ζωής σύνδεσης: Οι φάσεις για την σύνδεση BLE ελέγχονται και διαχειρίζονται αυτόματα από την εφαρμογή:

```
private val gattCallback = object : BluetoothGattCallback() {
    override fun onConnectionStateChange(gatt: BluetoothGatt, status: Int,
newState: Int) {
        if (newState == BluetoothProfile.STATE_CONNECTED) {
            _connectedDevice.value = gatt.device
            gatt.discoverServices() // Αυτόματη ανακάλυψη υπηρεσιών
            startRecurringReads() // Έναρξη συλλογής δεδομένων
            startRssiAveraging() // Έναρξη μέτρησης απόστασης
        }
    }
}
```

Κώδικας 6-5: Αντικείμενο gattCallback για διαχείριση σύνδεσης BLE

Αυτόματη ανακάλυψη των υπηρεσιών: Η ανακάλυψη των υπηρεσιών και των προς εμφάνιση χαρακτηριστικών γίνεται αυτόματα με την επιτυχή σύνδεση της συσκευής, ώστε να υπάρχει ο λιγότερο δυνατός χρόνος καθυστέρησης την ορθής λειτουργίας της εφαρμογής.

6.3.3 Διαχείριση Χαρακτηριστικών και υπηρεσιών BLE

Για την λειτουργία της συσκευής εκτίθεται από τον μικροπολογιστή η υπηρεσία με τα χαρακτηριστικά της, ώστε να γίνει διάβασμα από την εφαρμογή. Τα χαρακτηριστικά είναι οι μεταβλητές που πάει να παρακολουθήσει η εφαρμογή για να προβεί σε ενέργειες αν κριθεί απαραίτητο. Παρακάτω είναι η υπηρεσία με τα χαρακτηριστικά. Είναι Πρακτικά τα ίδια δεδομένα όπως και στην πλευρά του μικροϋπολογιστή.

```
private val TEMP_CHAR_UUID = UUID.fromString("12345678-1234-5678-1234-56789abcdef2") //χαρακτηριστικό θερμοκρασίας σε βαθμούς Κελσίου
private val HUM_CHAR_UUID = UUID.fromString("12345678-1234-5678-1234-56789abcdef3")//χαρακτηριστικό υγρασίας, η τιμή τελικά αποδίδεται ως ποσοστό
private val KID_CHAR_UUID = UUID.fromString("12345678-1234-5678-1234-56789abcdef1")//χαρακτηριστικό παρουσίας παιδιού
private val BATTERY_CHAR_UUID = UUID.fromString("12345678-1234-5678-1234-56789abcdef4") //χαρακτηριστικό στάθμης τάσης της μπαταρίας
private val SERVICE_UUID = UUID.fromString("12345678-1234-5678-1234-56789abcdef0")//Η υπηρεσία
```

Κώδικας 6-6: Ορισμός UUID χαρακτηριστικών και υπηρεσίας BLE

Ανάλογα με τον τύπο των δεδομένων, γίνεται και διαφορετική επεξεργασία:

```
when (uuid) {
    KID_CHAR_UUID -> {
        val flag = raw?.firstOrNull()?.toInt() ?: -1
        _kidPresence.value = if (flag == 1) "Yes" else "No"
    }
    TEMP_CHAR_UUID -> {
        if (raw.size == 4) {
            _temperature.value = ByteBuffer.wrap(raw)
                .order(ByteOrder.LITTLE_ENDIAN)
                .float
        }
    }
}
```

Κώδικας 6-7: Κώδικας ανάγνωσης χαρακτηριστικών παιδιού και θερμοκρασίας

Κεφάλαιο 6ο:

Η συλλογή δεδομένων γίνεται ανά 2 δευτερόλεπτα για τα χαρακτηριστικά, μετά την πρώτη ανάγνωση που εκτελείται το ταχύτερο δυνατό. Επίσης η ο υπολογισμός της απόστασης βάσει του δείκτη ισχύος ληφθέντος σήματος (RSSI) γίνεται κάθε 1 δευτερόλεπτο.

6.3.4 Διαχείριση Σύνδεσης και Αποσύνδεσης

Για την κατάσταση κατά την σύνδεση έχει εισαχθεί ένα περιθώριο χρονικό κανονικοποίησης του συστήματος, ώστε να λειτουργήσει και να εξισορροπηθούν τα δεδομένα, να υπάρχουν τιμές για όλα και να μην υπάρξουν τυχόν ψευδείς ειδοποιήσεις στον χρήστη. Με την πάροδο 30 δευτερολέπτων από την σύνδεση οι μετρήσεις λαμβάνονται υπόψιν και πλέον υπάρχει κανονική λειτουργία του συστήματος.

6.4 Υπολογισμός Απόστασης από την Συσκευή

Αυτή η μέτρηση είναι σημαντική για περιπτώσεις οριακών καταστάσεων, αν για παράδειγμα το όχημα είναι κοντά στον τελικό προορισμό του χρήστη πχ πάρκινγκ πιλοτής και διαμέρισμα πρώτου ορόφου, εκεί η αποσύνδεση απλά του BLE δεν θα ήταν ενδεικτικός παράγοντας για να καθορίσει η εφαρμογή ότι ο χρήστης απουσιάζει από το όχημα, διότι μπορεί να μην γινόταν ποτέ αποσύνδεση, οπότε και η συσκευή θα αποτύγγανε να λειτουργήσει για τον λόγο για τον οποίο σχεδιάστηκε. Οπότε σαν λύση ήταν η μέτρηση της ισχύος της ασύρματης λήψης και υπολογισμού της απόστασης βάσει της εξασθένισης που συμβαίνει λόγω απόστασης και εμποδίων.

Η σχέση που περιγράφει την απόσταση σε σχέση με την τον δείκτη της ισχύος ληφθέντος σήματος περιγράφεται από την παρακάτω εξίσωση:

$$d = 10^{\frac{T_xpower - RSSI}{10 * n}}$$

Όπου:

- d → Απόσταση (m).
- Tx(power) → Ισχύς εκπομπής (μετρούμενη στο 1m).
- RSSI → Δείκτης ισχύος ληφθέντος σήματος.
- n → Εκθέτης απώλεια διαδρομής.

Πρακτικά μέσα στον κώδικα υλοποιήθηκε με τον παρακάτω τρόπο:

```
import kotlin.math.pow

private fun estimateDistance(rssi: Int): Double {
    return 10.0.pow((calibratedTxPower - rssi) / (10.0 * pathLossExponent))
}
```

Κώδικας 6-8: Υπολογισμός απόστασης από το RSSI

Βέβαια για να είναι όσο πιο ακριβής η μέτρηση υλοποιήθηκε και ο κώδικας για βαθμονόμηση μέσω πραγματικών μετρήσεων. Η λογική ήταν να γίνει λήψη 10 μετρήσεων σε 1 μέτρο για να προσδιοριστεί ο μέσος όρος και βάση αυτού μετέπειτα να γίνονται οι υπολογισμοί.

```
// Συλλογή 10 μετρήσεων RSSI σε απόσταση 1 μέτρου
repeat(10) {
    currentRssi?.let { collectedRssi.add(it) }
    delay(1000)
}
val avg = collectedRssi.average().toInt()
```

Κώδικας 6-9: Διαδικασία εύρεσης μέσου όρου RSSI

6.5 Σενάρια Λειτουργίας

όπως ήδη αναφέρθηκε, για την λογική της εφαρμογής δημιουργήθηκαν σενάρια, όπως : ποιες συνθήκες μαζί με ποιες θα επιφέρουν άμεση κινητοποίηση και ενημέρωση του χρήστη, ποια δεδομένα υπό συνδυασμούς και συνθήκες , δεν επηρεάζουν και δεν επιφέρουν αλλαγή κατάστασης και ειδοποιήσεις.

6.5.1 Κανονική λειτουργία

Υπό την κανονική λειτουργία, ο χρήστης ξεκλειδώνει το αυτοκίνητο, μπαίνει το παιδί στο παιδικό κάθισμα, ξεκινάει την εφαρμογή και γίνεται αυτόματη σύνδεση. Πλέον η εφαρμογή κάνει παρακολούθηση όλων των παραμέτρων σε πραγματικό χρόνο. Όταν πλέον πρέπει να φύγει από το αυτοκίνητο, βγαίνει , με την εφαρμογή να παρακολουθεί αν ξεπεραστεί η μέγιστη απόσταση και έπειτα ο χρήστης παίρνει το παιδί και φεύγει. Η εφαρμογή εντοπίζοντας την απουσία παιδιού δεν βλέπει λόγο ενεργοποίησης ειδοποιήσεων.

6.5.2 Λειτουργία Συναγερμού και Ειδοποιήσεων

Πρώτο σενάριο : Παιδί στο κάθισμα και ο χρήστης απομακρύνεται.

Ο χρήστης απομακρύνεται, ενώ το παιδί βρίσκεται στο αυτοκίνητο. Η ισχύς του σήματος της ασύρματης ζεύξης BLE μειώνεται.

Λογική:

- Η απόσταση υπολογίζεται συνέχεια από το σύστημα.
- Η εφαρμογή εντοπίζει υπέρβαση του μέγιστου θεσπισμένου ορίου απόστασης (η προεπιλεγμένη τιμή έχει οριστεί ως 3 μέτρα).
- Εφόσον υπάρχει παρουσία παιδιού, ξεκινάει αντίστροφη μέτρηση 30 δευτερολέπτων.
- Στον χρήστη εμφανίζεται στην οθόνη(ακόμα και αν είναι κλειδωμένο το κινητό) ειδοποίηση και ενεργοποιείται και ηχητική σήμανση μαζί με δόνηση.
- Στην ειδοποίηση ενημερώνεται ο χρήστης ότι το παιδί βρίσκεται ακόμα στο αυτοκίνητο και μπορεί να βρίσκεται σε κίνδυνο.
- Εφόσον επιλέξει ότι έλαβε γνώση πατώντας ok, τότε η εφαρμογή σταματάει τις ειδοποιήσεις και παύει η λειτουργία της.
- Σε περίπτωση μη απόκρισης του χρήστη, γίνεται αποστολή SMS με τοποθεσία σε 2 επαφές που έχουν καταχωρηθεί από τον χρήστη και με μήνυμα επίσης προσαρμοζόμενο από αυτόν. Η μορφή του μηνύματος είναι : {προσαρμοσμένο κείμενο}+τοποθεσία.

Κεφάλαιο 6ο:

Δεύτερο σενάριο: Παιδί στο κάθισμα, χρήστης εντός ορίων απόστασης , αλλά υψηλή Θερμοκρασία.

Ο χρήστης βρίσκεται εντός του ορίου απόστασης, το παιδί είναι παρόν, αλλά η θερμοκρασία είναι υψηλή.

Εφόσον η θερμοκρασία υπερβεί το 'ασφαλές' όριο (θεσπισμένο από τον χρήστη), τότε:

- Εφόσον υπάρχει παρουσία παιδιού, ξεκινάει αντίστροφη μέτρηση 30 δευτερολέπτων.
- Στον χρήστη εμφανίζεται στην οθόνη(ακόμα και αν είναι κλειδωμένο το κινητό) ειδοποίηση και ενεργοποιείται και ηχητική σήμανση μαζί με δόνηση για υψηλή θερμοκρασία εντός του αυτοκινήτου.
- Εφόσον επιλέξει ότι έλαβε γνώση πατώντας οκ, τότε η εφαρμογή σταματάει τις ειδοποιήσεις και παύει η λειτουργία της.
- Σε περίπτωση μη απόκρισης του χρήστη, γίνεται αποστολή SMS όπως και παραπάνω.

Σε περίπτωση που η θερμοκρασία υπερβεί το κρίσιμο όριο θερμοκρασίας (θεσπισμένο από τον χρήστη), τότε:

- Γίνεται άμεσα αποστολή SMS με την ειδοποίηση ανάγκης και την τοποθεσία όπως και παραπάνω.

Τρίτο σενάριο: Παιδί στο κάθισμα και αποσύνδεση συσκευής.

Ο χρήστης βρίσκεται εντός του ορίου απόστασης, το παιδί είναι παρόν, αλλά η εφαρμογή αποσυνδέθηκε.

- Εμφάνιση προειδοποίησης.
- Αναμονή 30 δευτερόλεπτα για πιθανή επανασύνδεση.
- Αν γίνει επανασύνδεση, συνέχεια λειτουργίας.
- Αν δεν γίνει επανασύνδεση η λογική είναι όπως όταν απομακρύνεται ο χρήστης πέραν του θεσπισμένου ορίου. Ειδοποίηση → Αναμονή απόκρισης → Η μη απόκριση χρήστη αποστέλλει SMS

```
if (!disconnectionTriggered) {
    disconnectionTriggered = true
    if (kidPresence.equals("Yes", ignoreCase = true)) {
        startAlarmCountdown()
    } else {
        Toast.makeText(context, "Device disconnected", Toast.LENGTH_SHORT).show()
    }
}
```

Κώδικας 6-10: Κώδικας ενεργοποίησης συναγερμού σε περίπτωση αποσύνδεσης συσκευής

Τέταρτο σενάριο: Παιδί στο κάθισμα και χαμηλή στάθμη μπαταρίας.

Ο χρήστης βρίσκεται εντός του ορίου απόστασης, το παιδί είναι παρόν, αλλά η μπαταρία είναι 1-19%.

Στα πρώτα 30 δευτερόλεπτα σύνδεσης ή αν η μπαταρία εμφανίζει ποσοστό 0%, τότε γίνεται αγνόηση της μπαταρίας και της διαδικασίας προειδοποίησης, διότι λαμβάνονται ως μη αρχικοποιημένα ή λανθασμένα δεδομένα.

Όταν φτάσει κάτω από 20%:

- Στον χρήστη εμφανίζεται στην οθόνη(ακόμα και αν είναι κλειδωμένο το κινητό) ειδοποίηση και ενεργοποιείται και ηχητική σήμανση μαζί με δόνηση.
- Εφόσον επιλέξει ότι έλαβε γνώση πατώντας οκ, τότε η εφαρμογή σταματάει τις ειδοποιήσεις για μπαταρία μέχρι να φτάσει 25%, εφόσον φτάσει σε αυτήν την στάθμη, αν επανέλθει κάτω του 20% επαναλαμβάνεται η διαδικασία αυτή.
- Σε περίπτωση μη απόκρισης του χρήστη, γίνεται αποστολή SMS.

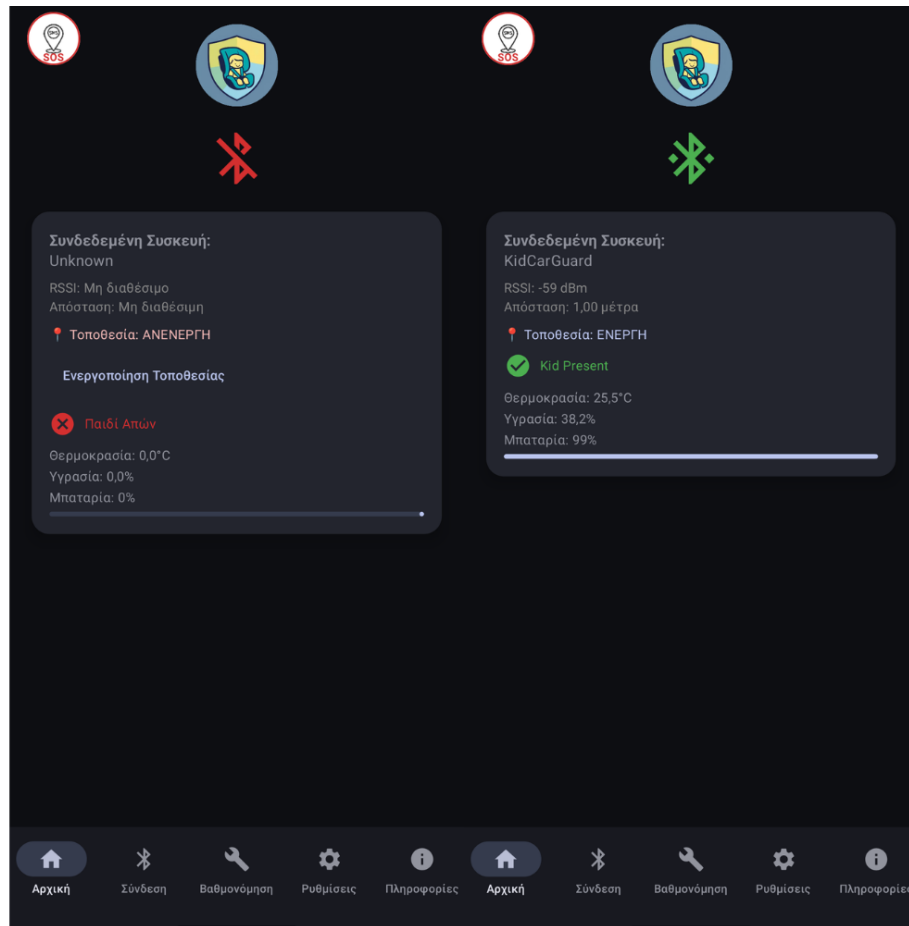
6.6 Διεπαφή Χρήστη και Σχεδίαση Οθονών

6.6.1 Αρχική - Κύρια οθόνη

Η κύρια οθόνη έχει σχεδιαστεί λαμβάνοντας υπόψιν ότι ζητείται απλότητα και πρέπει να φαίνονται μόνο σημαντικές πληροφορίες για να είναι ξεκάθαρη απλή και λειτουργική για τον χρήστη. Χρησιμοποιήθηκαν και εικονίδια ενδείξεων αλλά και χρώματα για την σύνδεση συσκευής, την κατάσταση της τοποθεσίας(ενεργή/ανενεργή) και της παρουσίας παιδιού.

Η εφαρμογή κάνει ενημέρωση συνέχεια και δεν περιμένει διάδραση από τον χρήστη για εμφάνιση νέων δεδομένων.Στην οθόνη εμφανίζονται:

- Κουμπί SOS
- Ένδειξη της κατάστασης σύνδεσης.
- Όνομα συνδεδεμένης συσκευής.
- RSSI.
- Απόσταση χρήστη-συσκευής.
- Τοποθεσία.
- Παρουσία παιδιού.
- Θερμοκρασία.
- Υγρασία.
- Στάθμη μπαταρίας.
- Μπάρα Πλοήγησης με όλες τις οθόνες.
- Αρχική (Η κύρια οθόνη).
- Σύνδεση.
- Βαθμονόμηση.
- Ρυθμίσεις.
- Πληροφορίες.



Εικόνα 6-2: Αρχική οθόνη εφαρμογής KidGuard

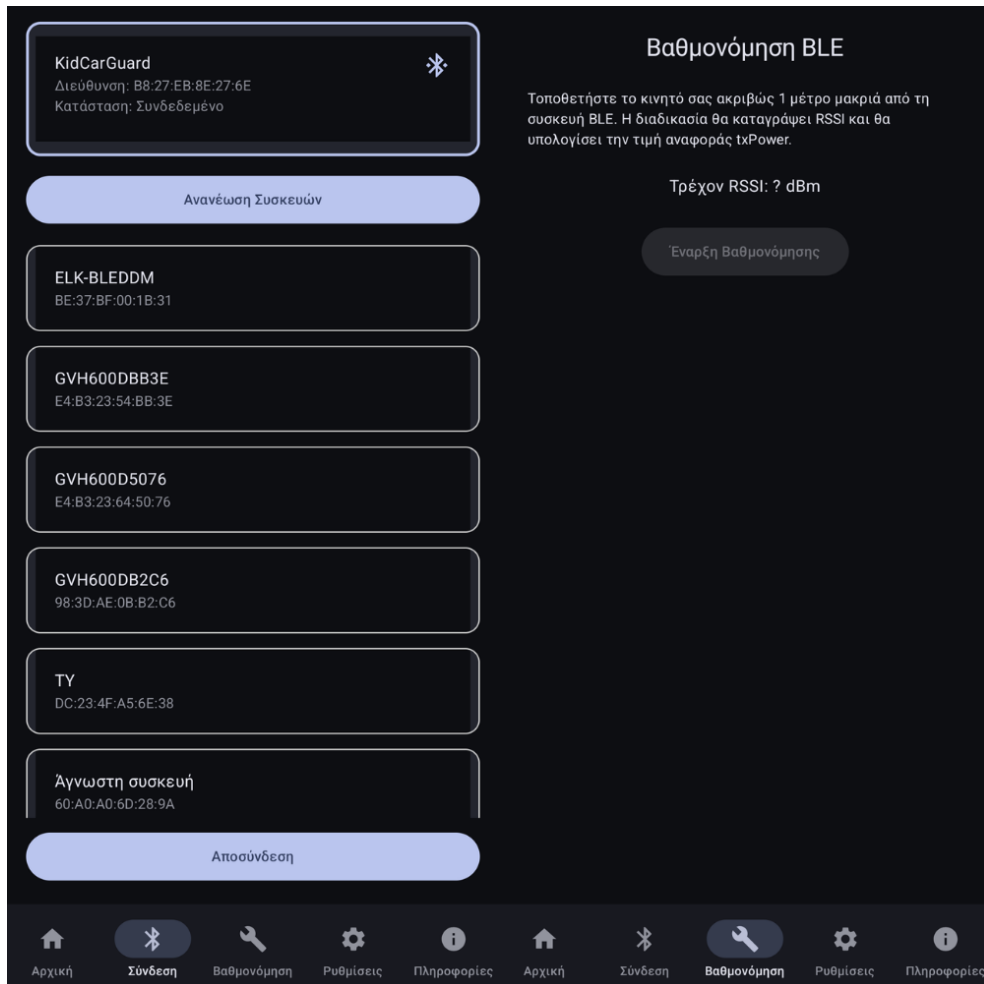
6.6.2 Οθόνη Σύνδεσης και Οθόνη Βαθμονόμησης

Οθόνη Σύνδεσης

Στην οθόνη αυτή ο χρήστης θα επιλέξει την συσκευή προς σύνδεση και θα του εμφανίσει την συνδεδεμένη συσκευή, επίσης έχει την δυνατότητα αποσύνδεσης αν αυτό χρειαστεί, με το αντίστοιχο κουμπί.

Οθόνη Βαθμονόμησης

Στην οθόνη βαθμονόμησης προβάλλονται οδηγίες για την διενέργεια βαθμονόμησης από τον χρήστη. Η βαθμονόμηση δεν είναι υποχρεωτική, αλλά σε περίπτωση μη ορθής λειτουργίας, δίνεται σαν επιλογή για πιο ακριβείς μετρήσεις απόστασης.

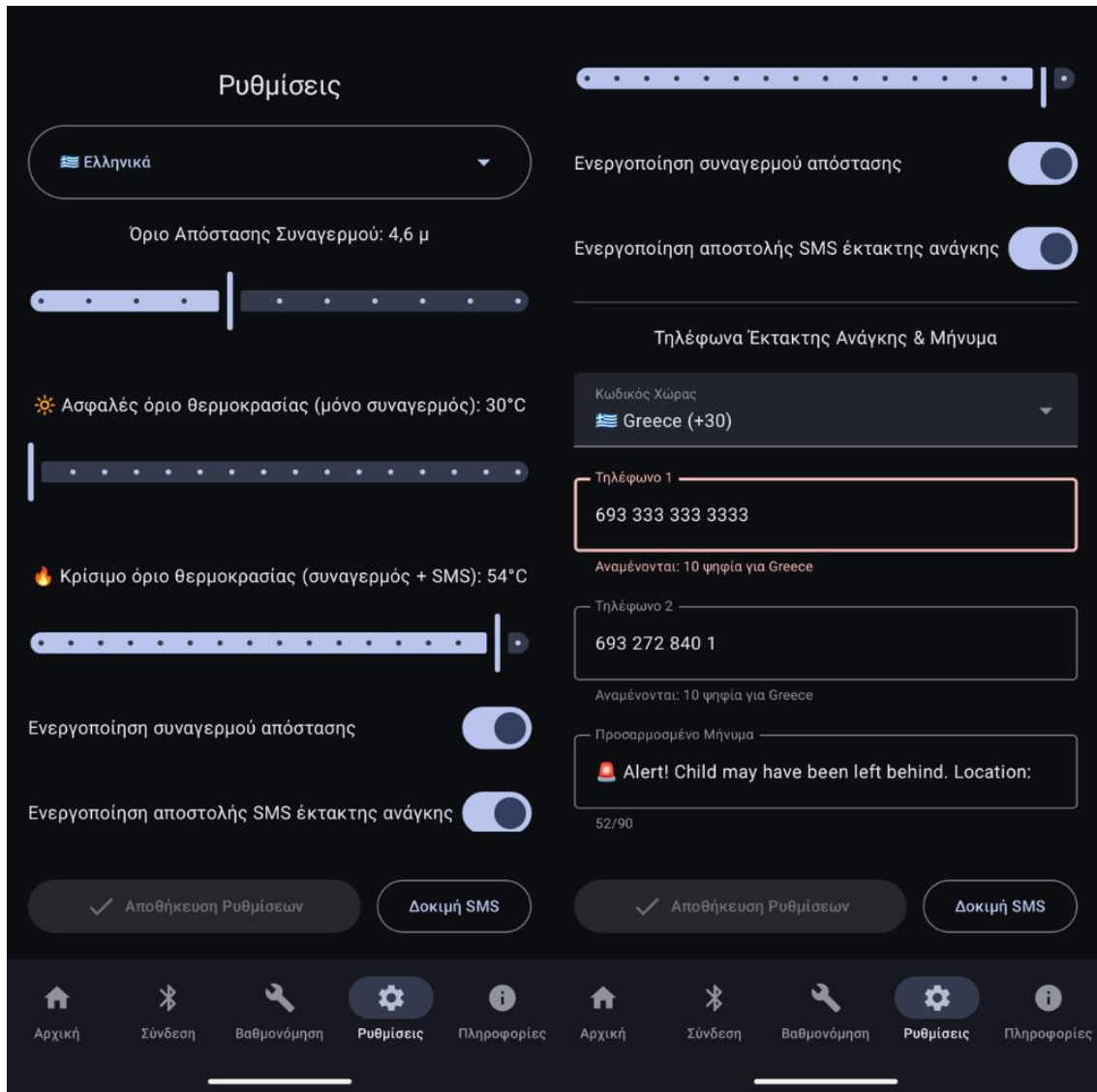


Εικόνα 6-3: Οθόνη σύνδεσης και οθόνη βαθμονόμησης εφαρμογής KidGuard

6.6.3 Οθόνη Ρυθμίσεων

Την οθόνη ρυθμίσεων καλείται ο χρήστης να:

- Επιλέξει μια από τις γλώσσες που υποστηρίζονται από την εφαρμογή.
- Επιλέξει το όριο απόστασης από 1 έως και 10 μέτρα.
- Τα όρια θερμοκρασιών
- Ασφαλές όριο, δηλαδή το όριο που μπορεί να θεσπίσει ως απλή προειδοποίηση.
- Κρίσιμο όριο, δηλαδή το όριο που γίνεται άμεσα αποστολή SMS ειδοποίησης.
- Επιλέξει αν ο τοπικός συναγερμός – ειδοποίηση θα είναι ενεργά.
- Ενεργοποιήσει/Απενεργοποιήσει την αποστολή SMS.
- Ρυθμίσει και ορίσει τα τηλέφωνα εκτάκτου ανάγκης.
- Σε αυτά έχει προσαρμοστεί το πεδίο, ώστε ανάλογα με τον κωδικό της χώρας να προτρέπεται ο χρήστης να εισάγει τον αντίστοιχο αριθμό ψηφίων και αν δεν είναι σωστός να μην γίνεται δεκτός.
- Προσαρμόσει το αυτοματοποιημένο μήνυμα εκτάκτου ανάγκης.



Εικόνα 6-4: Οθόνη ρυθμίσεων εφαρμογής KidGuard

6.6.4 Πολυγλωσσική Υποστήριξη

Για την υλοποίηση της εφαρμογής με πολυγλωσσική υποστήριξη ο κώδικας που γράφτηκε:

- Αρχικά κάνει την αποθήκευση της προτίμησης του χρήστη για γλώσσα εμφάνισης.
- Κάνει φόρτωση της επιλεγμένης γλώσσας και αλλαγή του Locale στην εφαρμογή.
- Δημιουργεί αναδυόμενο μενού λίστας (dropdown menu) για να γίνει επιλογή γλώσσας μέσα από τις 3 επιλογές που δημιουργήθηκαν, Ελληνικά, Αγγλικά και Ισπανικά.
- Κάνει χρήση του `stringResource()` αντί για ονόματα μέσα στις οθόνες, ώστε οι ονομασίες να δίδονται στο αρχείο `strings.xml` όλες μαζί και να είναι εύκολη η αλλαγή γλώσσας και η διαχείριση.
- Όταν γίνει αλλαγή γλώσσας επανεκκινεί την προβολή της εφαρμογής για εφαρμογή της νέας γλώσσας.

```

// LocaleSettingsStore.kt - αποθήκευση επιλεγμένης γλώσσας
fun saveLanguage(context: Context, languageCode: String) {
    runBlocking {
        context.localeSettingsDataStore.edit { prefs ->
            prefs[LANGUAGE_KEY] = languageCode
        }
    }
}
// φόρτωση αποθηκευμένης γλώσσας
fun loadLanguage(context: Context): String {
    return runBlocking {
        context.localeSettingsDataStore.data
            .map { it[LANGUAGE_KEY] ?: "en" }
            .first()
    }
}
// MainActivity.kt - εφαρμογή γλώσσας στην εκκίνηση
override fun onCreate(savedInstanceState: Bundle?) {
    val savedLang = LocaleSettingsStore.loadLanguage(this)
    setLocale(this, savedLang)
}
// αλλαγή locale του συστήματος
private fun setLocale(context: Context, languageCode: String) {
    val locale = Locale(languageCode)
    Locale.setDefault(locale)
    val config = context.resources.configuration
    config.setLocale(locale)
    context.resources.updateConfiguration(config,
context.resources.displayMetrics)
}
// SettingsScreen.kt - dropdown επιλογής γλώσσας
val availableLanguages = listOf(
    Triple("en", "English", "GB"),
    Triple("el", "Ελληνικά", "GR"),
    Triple("es", "Español", "ES")
)
// εφαρμογή νέας γλώσσας και επανεκκίνηση
fun applyLanguage(languageCode: String) {
    Locale.setDefault(Locale(languageCode))
    LocaleSettingsStore.saveLanguage(context, languageCode)
    (context as? Activity)?.recreate()
}
// HomeScreen.kt, SettingsScreen.kt, InfoScreen.kt κτλ - χρήση localized strings
Text(stringResource(R.string.settings_title))

```

Κώδικας 6-11: Συναρτήσεις αποθήκευσης και εφαρμογής γλώσσας στην εφαρμογή

6.6.5 Οθόνη Πληροφοριών

Στην οθόνη πληροφοριών παρουσιάζονται περιγραφή της εφαρμογής, οδηγίες και επισημάνσεις για τον χρήστη. Δίνεται επίσης συμβουλές ασφαλείας για να λειτουργεί σωστά η εφαρμογή, όπως και η πληροφόρηση για τον τρόπο λειτουργίας και χρήσης δεδομένων, κάνοντας ξεκάθαρο πως η εφαρμογή

Κεφάλαιο 6ο:

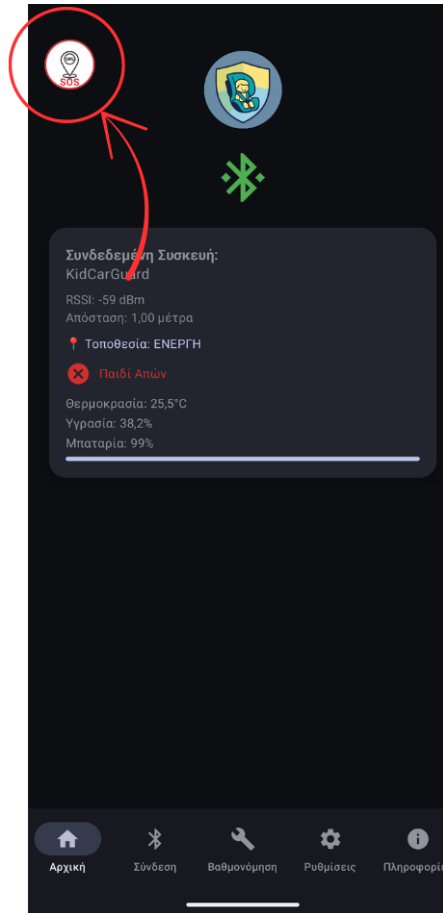
λειτουργεί τοπικά χωρίς διαμοιρασμό πληροφοριών φωτογραφιών και γενικότερα χωρίς να εκθέτει την ιδιοτικότητα του χρήστη ή παιδιών και ενηλίκων που εκτίθενται στο πεδίο λήψης της κάμερας του συστήματος.



Εικόνα 6-5: οθόνη πληροφοριών εφαρμογής KidGuard

6.6.6 Κουμπί SOS

Το κουμπί SOS έχει την λειτουργία άμεσης αποστολής τοποθεσίας με το μήνυμα SMS και είναι σε σημείο πάνω αριστερά πρώτον για να αποφεύγονται τυχαία πατήματα και δεύτερον για να φαίνεται πιο ξεκάθαρα στον χρήστη.



Εικόνα 6-6 : Κουμπί SOS στην αρχική οθόνη της εφαρμογής KidGuard

Για την εξασφάλιση πως δεν θα πατηθεί κατά λάθος, ενσωματώθηκε λογική που ανιχνεύει δεύτερο πάτημα μέσα σε χρόνο 2 δευτερολέπτων από το πρώτο ώστε να γίνει αποστολή. Με το πρώτο πάτημα, εμφανίζεται μήνυμα επικάλυψης της οθόνης «Πατήστε ξανά για αποστολή SOS SMS» και με το δεύτερο αποστέλλεται το μήνυμα και εμφανίζεται και το μήνυμα επικάλυψης οθόνης «Το SOS SMS εστάλη!»

```

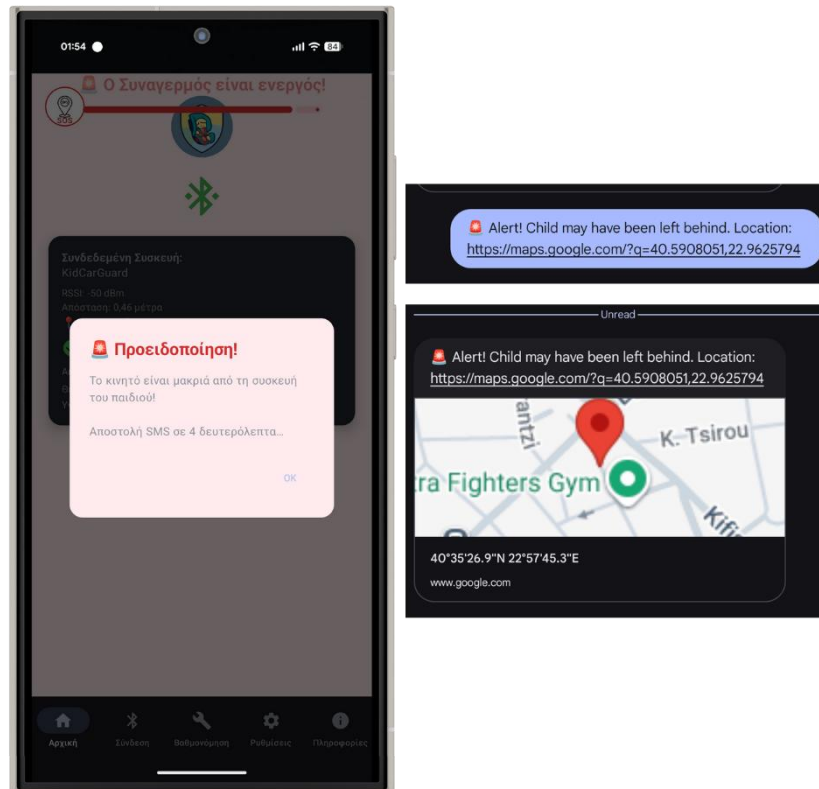
val now = SystemClock.elapsedRealtime()
if (now - lastClickTime < doublePressInterval) {
    onSendSms()
    Toast.makeText(context, context.getString(R.string.sos_toast_confirm),
    Toast.LENGTH_SHORT).show()
    lastClickTime = 0L //Reset σε 0
} else {
    Toast.makeText(context, context.getString(R.string.sos_toast_double_click),
    Toast.LENGTH_SHORT).show()
    lastClickTime = now // Αποθήκευση χρόνου πρώτου κλικ
}
    
```

Κώδικας 6-12: Κώδικας ελέγχου διπλού κλικ για αποστολή SOS

6.6.7 Ειδοποίηση και Αποστολή SMS

6.6.7.1 Ειδοποιήσεις

Οι ειδοποιήσεις συνδυάζουν δόνηση , με χρήση «VibrationEffect.createWaveform()», ηχητικά και οπτικά εφέ καθώς και μεγιστοποίηση της έντασης αυτόματα μέσω του «AudioManager», ώστε να είναι όσο πιο πιθανό γίνεται να αντιδράσει εγκαίρως ο χρήστης. Επίσης, εμφανίζεται το αναδυόμενο μήνυμα προσαρμοζόμενο ανάλογα με τον τύπο ‘συναγερμού’ που ενεργοποιείται και με την προεπιλεγμένη γλώσσα.



Εικόνα 6-7: Ενημερώσεις και ειδοποιήσεις εφαρμογής KidGuard, με στιγμιότυπο οθόνης από το απεσταλμένο SMS με τοποθεσία

```

fun startAlarmCountdown() {
    // Μεγιστοποίηση έντασης
    maximizeVolume(context)

    //Ενεργοποίηση alarm service
    AlarmHelper.start(context)

    // Συνεχής δόνηση με μοτίβο
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
        vibrator.vibrate(VibrationEffect.createWaveform(longArrayOf(0, 500, 500),
0))
    } else {
        vibrator.vibrate(1000)
    }

    // Οπτικά εφέ - αλλαγή φόντου και αναβοσβήνον κείμενο
    val backgroundColor by animateColorAsState(
        targetValue = if (showWarning) Color(0xFFFFEBEE) else MaterialTheme.col-
orScheme.background
    )

    val blinkAlpha by infiniteBlink.animateFloat(
        initialValue = 1f, targetValue = 0.4f,
        animationSpec = infiniteRepeatable(
            animation = tween(600), repeatMode = RepeatMode.Reverse
        )
    )
}

// Προβαλλόμενο μήνυμα διαλόγου με αντίστροφη μέτρηση
if (showWarning) {
    val warningText = if (device == null && kidPresence.equals("Yes", ignoreCase
= true)) {
        stringResource(R.string.warning_bt_disconnected, countdown)
    } else if (temperature >= criticalTempThreshold) {
        stringResource(R.string.warning_high_temperature, countdown)
    } else {
        stringResource(R.string.warning_far_from_device, countdown)
    }

    AlertDialog(
        onDismissRequest = {},
        title = { Text(" ⚠ Warning!", fontWeight = FontWeight.Bold, color =
Color(0xFFD32F2F)) },
        text = { Text(warningText, style = MaterialTheme.typography.bodyLarge) }

```

Κώδικας 6-13: Συνάρτηση αντίστροφης μέτρησης και συναγερμού

6.6.8 Αποστολή SMS

Για την αποστολή των SMS δημιουργείται μέσω του «SmsSender» αντικείμενο για τον έλεγχο των αδειών και την αποστολή. Αρχικά ελέγχονται αν έχουν εκχωρηθεί άδειες μέσω του «ActiveCompat.checkSelfPermission()» για το «Manifest.Permission.Send_SMS». Στην περίπτωση που δεν υπάρχουν, δεν μπορούν να αποσταλούν τα μηνύματα. Εφόσον υπάρχουν οι άδειες, τότε φορτώνονται οι πληροφορίες με τους αριθμούς που έχει εισάγει ο χρήστης και συνεχίζεται η σύνθεση του μηνύματος από τα δυο τμήματά του, το κείμενο χρήστη και την τοποθεσία.

```
// ☺ Get location and send SMS
val location = try {
    fusedLocationClient.lastLocation.await()
} catch (_: Exception) { null }

val locationString = location?.let {
    "https://maps.google.com/?q=${it.latitude},${it.longitude}"
}
```

Κώδικας 6-14: Απόκτηση τοποθεσίας και δημιουργία συνδέσμου google

```

object SmsSender {
    fun sendEmergencySms(context: Context, location: String? = null) {
        // ελεγχος αδειών
        if (ActivityCompat.checkSelfPermission(context,
Manifest.permission.SEND_SMS)
            != PackageManager.PERMISSION_GRANTED) {
            Log.e("SmsSender", "SMS permission not granted.")
            return
        }

        val settings = SmsSettingsStore.load(context)

        // μήνυμα
        val baseMessage = settings.message.trim().replace("\n", " ")
        val locationText = location?.trim()?.replace("\n", " ") ?: ""

        val fullMessage: String = if (locationText.isNotBlank()) {
            "$baseMessage $locationText"
        } else {
            baseMessage
        }

        try {
            val smsManager = SmsManager.getDefault()
            val parts = smsManager.divideMessage(fullMessage) // υποστήριξη
            μηνύματος πολλαπλών τμημάτων

            //αποστολή και στις 2 επαφές
            smsManager.sendMultipartTextMessage(settings.phone1, null, parts,
            null, null)
            smsManager.sendMultipartTextMessage(settings.phone2, null, parts,
            null, null)

            Log.d("SmsSender", "SMS sent to both numbers.")
        } catch (e: Exception) {
            Log.e("SmsSender", "Failed to send SMS: ${e.message}")
        }
    }
}

```

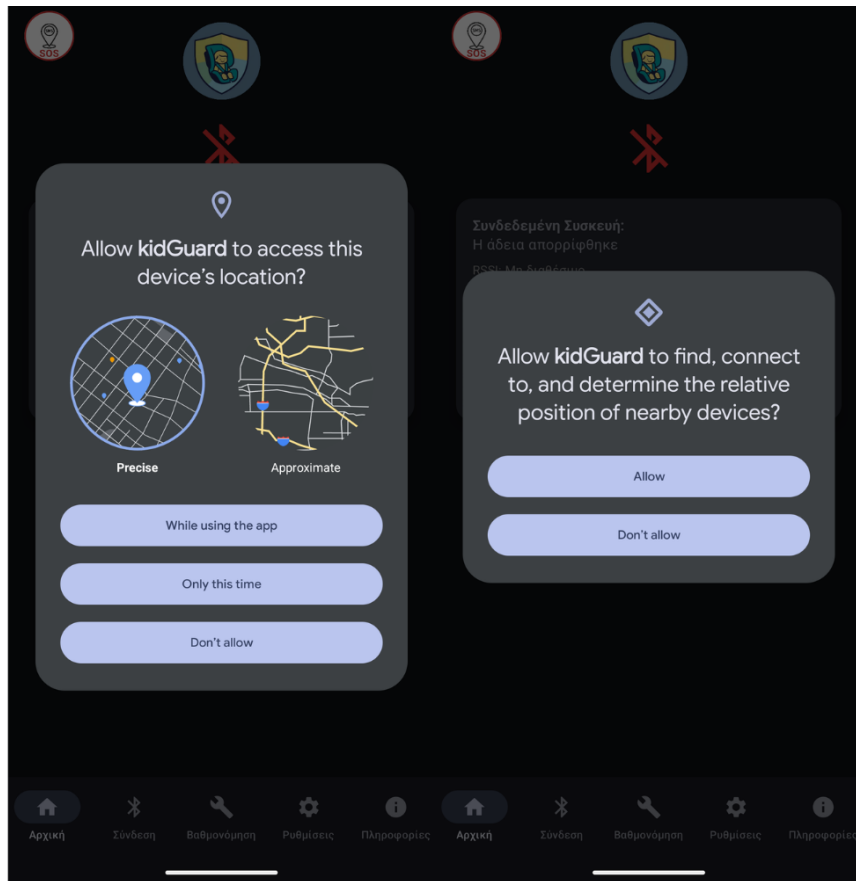
Κώδικας 6-15:Συνάρτηση αποστολής έκτακτου SMS με τοποθεσία

6.7 Διαχείριση Αδειών και Αρχική Ρύθμιση

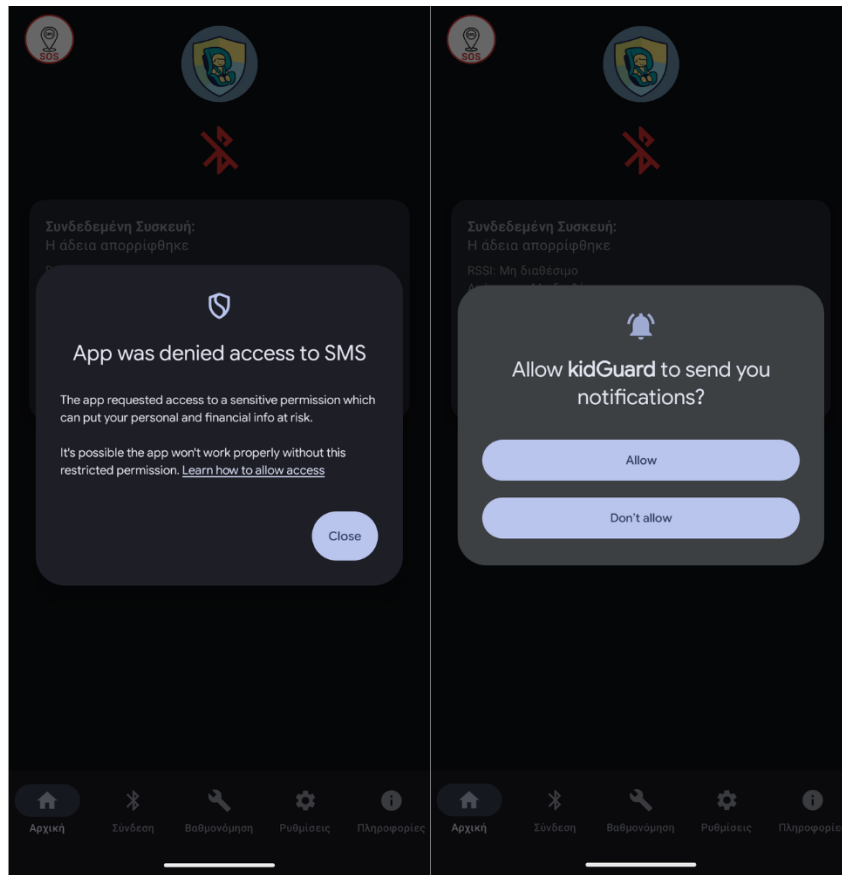
Η εφαρμογή κατά την πρώτη ενεργοποίηση μετά την εγκατάσταση καθοδηγεί τον χρήστη για την εκχώρηση αδειών τοποθεσίας, ειδοποιήσεων, διαχείρισης ασύρματης ζεύξης Bluetooth, απενεργοποίησης της βελτιστοποίησης μπαταρίας και λειτουργία της εφαρμογής στο παρασκήνιο και δημιουργίας και αποστολής SMS. Στην πρώτη εκτέλεση εμφανίζεται οθόνη που καθοδηγεί και δείχνει στον χρήστη τι δεν έχει ενεργοποιηθεί από τις άδειες. Συγκεκριμένα για την άδεια SMS πρέπει χειροκίνητα να ενεργοποιηθεί, διότι πλέον το Android είναι πιο αυστηρό στην εκχώρηση άδειας και δεν

Κεφάλαιο 6ο:

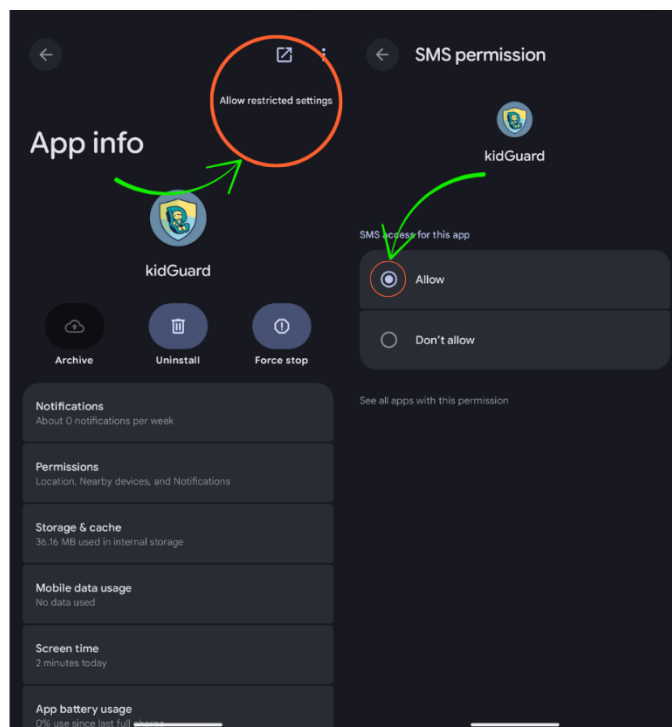
γίνεται αυτοματοποιημένα. Σε περίπτωση μη εκχωρημένων αδειών και επανεκκίνησης της εφαρμογής, επανελέγχονται και ζητώνται οι άδειες που δεν έχουν εκχωρηθεί.



Εικόνα 6-8: Μήνυμα για παραχώρηση άδειας τοποθεσίας και άδειας διαχείρισης ασύρματης ζεύξης Bluetooth



Εικόνα 6-9: Ενημέρωση για την άδεια SMS και παραχώρηση άδειας ειδοποιήσεων



Εικόνα 6-10: Κινήσεις για την παραχώρηση άδειας διαχείρισης SMS

6.8 Επίλογος

Στο κεφάλαιο αυτό έγινε η ανάλυση την Android εφαρμογής που δημιουργήθηκε για να ολοκληρώσει το λειτουργικό κομμάτι του συστήματος προστασίας παιδιών σε αυτοκίνητα. Αρχικά αναφέρθηκε ο τρόπος σύνθεσης της εφαρμογής, η χρήση γλώσσας Kotlin , η χρήση jetpack compose για την ευκολία υλοποίησης και συνδυασμού της διεπαφής χρήστη με την λειτουργία της εφαρμογής. Αναλύθηκαν τα περιεχόμενα της εφαρμογής και ο τρόπος λειτουργίας της. Παρατέθηκαν κομμάτια κώδικα για την παρουσίαση τρόπων προβολής, διαχείρισης και επεξεργασίας δεδομένων και για την υποστήριξη πολυγλωσσικότητας. Ακόμα, εξετάστηκαν οι επιμέρους λειτουργίες συλλογής δεδομένων μέσω της ασύρματης ζεύξης BLE και η υλοποίησή της. Στην συνέχεια αναφέρθηκαν τα σενάρια χρήσης και πως τα συλλεγόμενα δεδομένα καθορίζουν τον τρόπο αντίδρασης της εφαρμογής και ειδοποίησης του χρήστη.

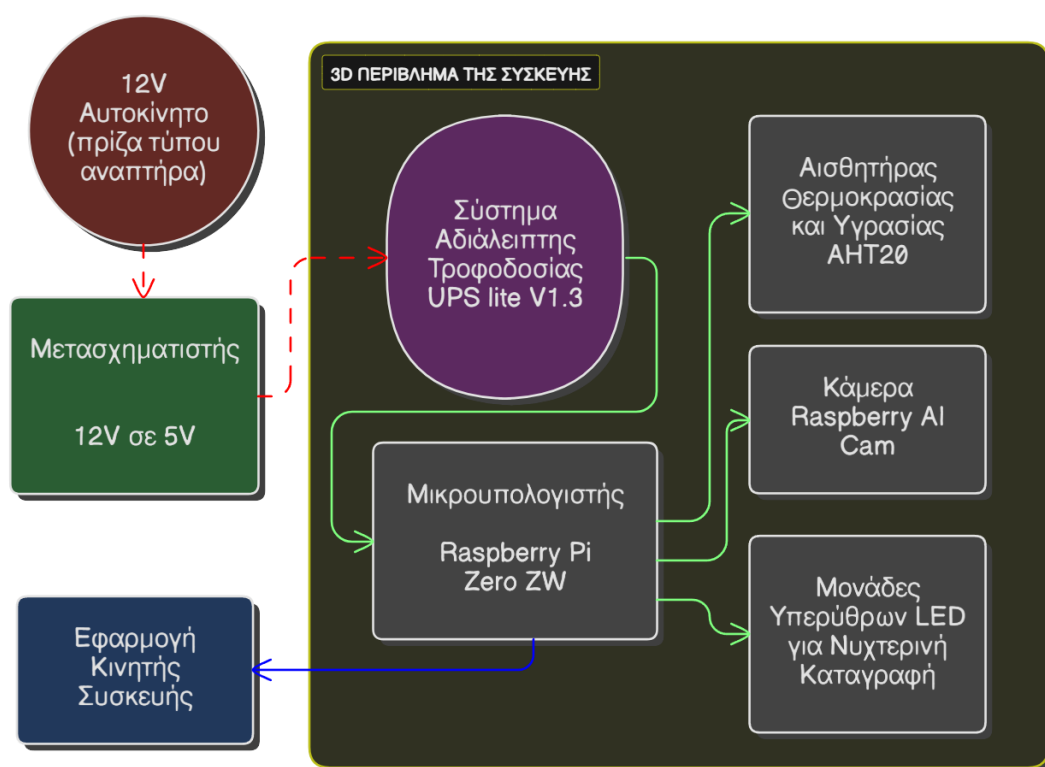
Με την ολοκλήρωση της ανάπτυξης της εφαρμογής, το σύστημα πλέον διαθέτει τα απαραίτητα υποσυστήματα: ανίχνευση παρουσίας μέσω τεχνητής νοημοσύνης, επεξεργασία δεδομένων αισθητήρων, επικοινωνία μέσω BLE και μέθοδο αλληλεπίδρασης με τον χρήστη μέσω της εφαρμογής. Στο επόμενο κεφάλαιο θα παρουσιαστεί η σχεδίαση, εκτύπωση και συναρμολόγηση του τελικού συστήματος, οι δοκιμές και η αξιολόγησή του, σε επίπεδο λειτουργικότητας και σε επίπεδο απόδοσης. Θα γίνει αναφορά και περιγραφή των πειραμάτων, των μετρήσεις που πραγματοποιήθηκαν και η σύγκριση με υπάρχουσες λύσεις, με στόχο να γίνει η εκτίμηση της αποτελεσματικότητας και της αξιοπιστίας της προτεινόμενης λύσης.

Κεφάλαιο 7ο: Υλοποίηση, Δοκιμές και Αξιολόγηση Συστήματος

7.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα παρουσιαστεί η διαδικασία ολοκλήρωσης της υλοποίησης του συστήματος οι δοκιμές, η αξιολόγηση της λειτουργίας και η σύγκριση με υπάρχοντες συσκευές και συστήματα που έχουν σχεδιαστεί για τον ίδιο σκοπό, την ασφάλεια των παιδιών στα αυτοκίνητα.

Πιο αναλυτικά, θα γίνει αναφορά της συναρμολόγησης, της δημιουργίας περιβλήματος για την συσκευή και οι δοκιμές μέχρι να βρεθεί η βέλτιστη σύνθεση των τμημάτων, αυτή που θα ικανοποιεί τις ανάγκες του συστήματος, ώστε να μπορεί να αποτελέσει σύστημα μέσα σε ένα αυτοκίνητο. Έπειτα, θα γίνει παρουσίαση της λειτουργίας και των δοκιμών, τόσο σε πραγματικές συνθήκες όσο και σε συνθήκες προσομοίωσης, στο εργαστήριο, και στην συνέχεια η αξιολόγηση της απόδοσης του συστήματος. Τέλος, βάσει των παραπάνω θα γίνει μια σύγκριση χαρακτηριστικών, τρόπου λειτουργίας και ασφαλείας, με συσκευές του εμπορίου που επιχειρούν να δώσουν λύση στο ίδιο πρόβλημα.



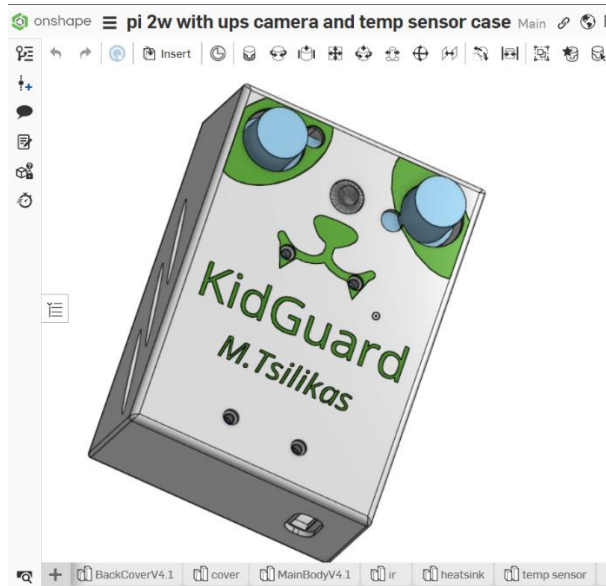
Εικόνα 7-1: Διάγραμμα ενοτήτων του υποσυστήματος μικροϋπολογιστή και των περιφερειακών του

7.2 Τρισδιάστατος Σχεδιασμός και Τελικό Μοντέλο

Για την τελική υλοποίηση, με τα υποσυστήματα έτοιμα και λειτουργικά, έπρεπε να δημιουργηθεί και το περίβλημα, για να φιλοξενήσει αισθητήρες και μικροϋπολογιστή. Οι έτοιμες επιλογές δεν ικανοποιούσαν τις ανάγκες για στήριξη, αερισμό σχήμα, υποδοχές και η λύση που επιλέχθηκε ήταν να σχεδιαστεί σύμφωνα με τις ανάγκες το κατάλληλο περίβλημα. Δηλαδή η βάση που θα τοποθετούνταν

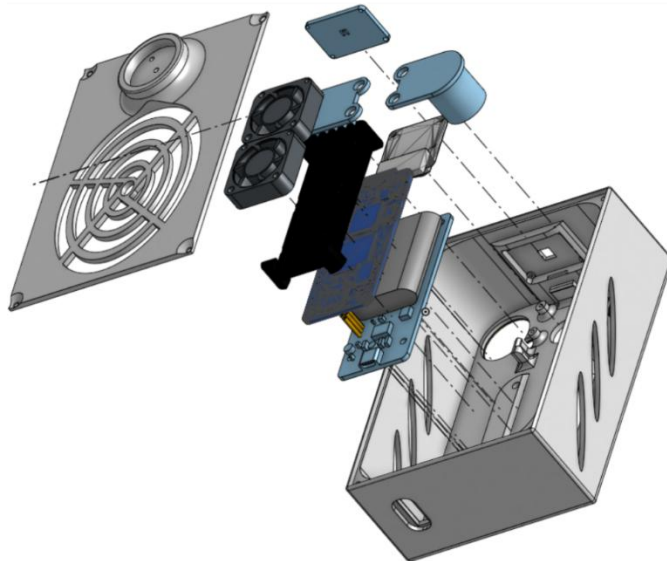
Κεφάλαιο 7ο:

και θα στηρίζονταν όλα τα επιμέρους τμήματα της υλοποίησης. Στόχος ήταν μικρό, σχετικά, μέγεθος, λειτουργικότητα, ευκολία εγκατάστασης και ασφαλής στήριξη των ηλεκτρονικών.



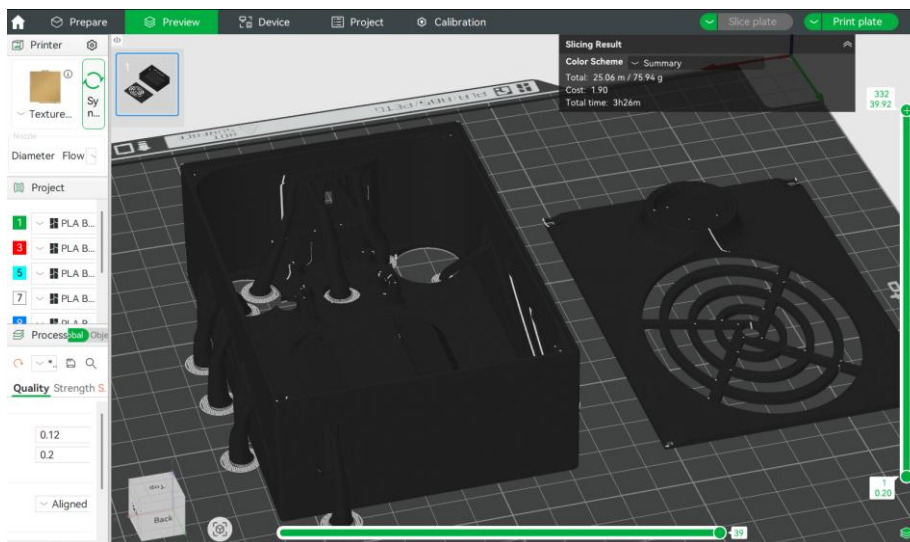
Εικόνα 7-2: Προεπισκόπηση μοντέλου 3D στο πρόγραμμα "onshape"

Η σχεδίαση έγινε σε διαδικτυακό δωρεάν πρόγραμμα σχεδίασης, το 'onshape'. Τα σχέδια που υλοποιήθηκαν μέχρι να γίνει το τελικό ήταν 3 διαφορετικά, ώστε να εξυπηρετηθούν ανάγκες που δεν είχαν προβλεφθεί, όπως δυο ανεμιστηράκια ή αλλαγές προσέγγισης στον τρόπο στήριξης μέσα στο όχημα, άρα και διαφορετικού τύπου προσαρμογέα της βάσης.



Εικόνα 7-3: Απεικόνιση τμημάτων από την προσομοίωση της σχεδίασης

Μετά την σχεδίαση ακολούθησε εισαγωγή στο πρόγραμμα 'Bambu Studio' ώστε να γίνει παραμετροποίηση, ρυθμίσεις ταχύτητας εκτύπωσης, ανάλυσης, υποστήριξης και να εξαχθεί ο G-code, η μορφή αρχείου που δίνει στο 3D εκτυπωτή όλες τις οδηγίες, ταχύτητες, συντεταγμένες πάνω στο επίπεδο, θερμοκρασίες. Για την εκτύπωση έγινε χρήση του εκτυπωτή Bambu Labs P1S. Το υλικό εκτύπωσης είναι PET-G, ώστε να αντέχει τις θερμοκρασίες που αναπτύσσονται στο εσωτερικό ενός αυτοκινήτου, να έχει αντοχή στην ηλιακή ακτινοβολία και να έχει μηχανική αντοχή.



Εικόνα 7-4: Απεικόνιση του μοντέλου στο πρόγραμμα διαχείρισης για τις ρυθμίσεις εκτύπωσης

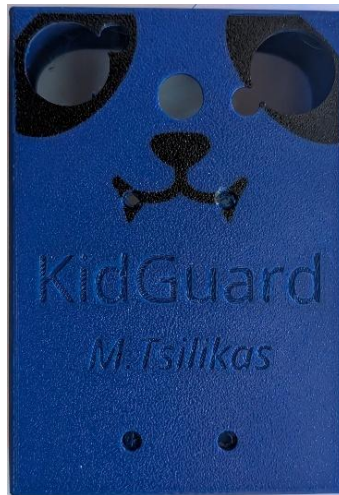
Οι δυο πρώτες απόπειρες σχεδιασμού, αν και είχαν μερικώς επιτυχία, δεν ικανοποιούσαν τα εξής χαρακτηριστικά που καλύφθηκαν από τον τελικό σχεδιασμό:

- Ανάγκη επαρκούς ενεργητικού αερισμού των ηλεκτρονικών.
- Στήριξη και κινητή βάση για την εύκολη εγκατάσταση.
- Μικρό όγκο.
- Κατάλληλη τοποθέτηση και στήριξη εξαρτημάτων.



Εικόνα 7-5: Πρώτη σχεδίαση

Κεφάλαιο 7ο:



Εικόνα 7-6: Δεύτερη σχεδίαση

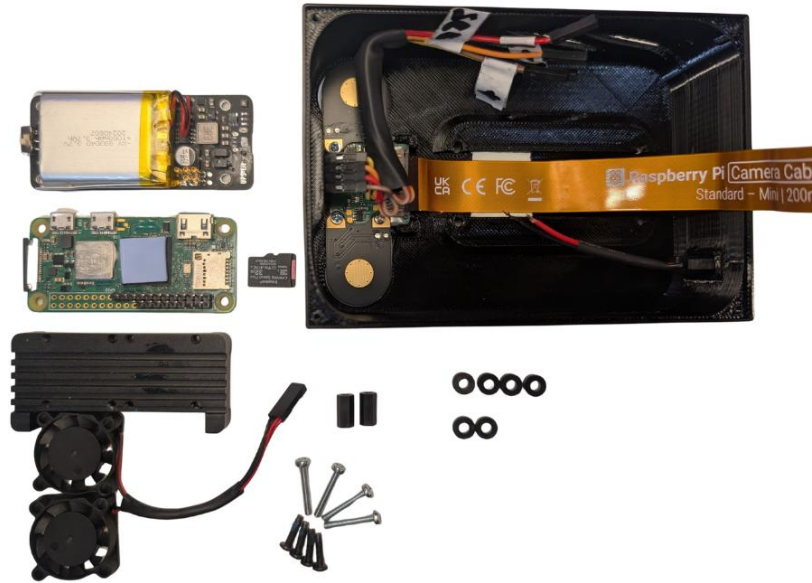
Η τρίτη σχεδίαση είναι και η τελική, έχουν επιτευχθεί οι στόχοι και ανταποκρίνεται στον σκοπό και την χρήση ως πρωτότυπο για απόδειξη λειτουργίας και δοκιμές στο πεδίο εφαρμογής, οι αισθητήρες προσαρμόζονται και εφαρμόζουν άρτια στο εσωτερικό, αφήνοντας εκτεθειμένα τα μέρη που προορίζονται για λήψη εικόνας, λήψη μετρήσεων θερμοκρασίας/υγρασίας και πρόσθετου φωτισμού για την τελική συσκευή προς χρήση.



Εικόνα 7-7: τελική μορφή περιβλήματος

Η διαδικασία συναρμολόγησης.

Παρακάτω στην Εικόνα 7-8 προβάλλονται όλα τα στοιχεία για την ολοκλήρωση της συναρμολόγησης της συσκευής, του υποσυστήματος δηλαδή μικροϋπολογιστή και περιφερειακών του. Στην εικόνα ήδη έχει προσαρμοστεί ο αισθητήρας IMX500 (κάμερα), οι δίοδοι υπέρυθρων, καθώς και η θύρα φόρτισης type-C.



Εικόνα 7-8: Τα προς συναρμολόγηση τμήματα

Το συναρμολογημένο εσωτερικό της συσκευής φαίνεται στην Εικόνα 7-9



Εικόνα 7-9: Το υποσύστημα συναρμολογημένο, χωρίς καπάκι

Κεφάλαιο 7ο:

Για την στερέωση στο αυτοκίνητο , επιλέχθηκε η στερέωση στο προσκέφαλο ενός εκ των μπροστινών καθισμάτων, για να γίνει αυτό χρησιμοποιήθηκε έτοιμος βραχίονας στήριξης τάμπλετ, του οποίου η τελική απόληξη (βάση τάμπλετ) αφαιρέθηκε και χρησιμοποιήθηκε η μπίλια που υπάρχει για να ενσωματωθεί στο πλαίσιο που σχεδιάστηκε και να προσφέρει ευελιξία κίνησης και προσαρμογής.



Εικόνα 7-10: Καπάκι και βραχίονας στερέωσης

Η τελική εμπρόσθια πλευρά της συναρμολογημένης συσκευής παρουσιάζεται στην **Error! Reference source not found.** Στην εμπρόσθια πλευρά φαίνονται ο αισθητήρας της κάμερας προσαρμοσμένος στο κέντρο και οι δύο διόδους υπέρυθρων για λήψη σε περιβάλλον χαμηλού φωτισμού.



Εικόνα 7-11: Προσαρτημένος βραχίονας στο καπάκι



Εικόνα 7-12: Εμπρόσθια όψη της τελικής κατασκευής

7.3 Δοκιμές Λειτουργίας

Με το σύστημα στην τελική του μορφή και έτοιμο προς χρήση, ξεκίνησε η διαδικασία των δοκιμών για να διαπιστωθεί η ορθή λειτουργία και τυχόν προβλήματα. Αρχικά, για την διενέργεια

δοκιμών, έγινε σύνδεση μέσω λογισμικού απομακρυσμένης πρόσβασης (Tiger VNC) στον μικροϋπολογιστή.

Με την απομακρυσμένη σύνδεση ενεργή, δημιουργήθηκε κώδικας για την ενεργοποίηση παραθύρου προεπισκόπησης της κάμερας και των ποσοστών βεβαιότητας ταξινόμησης. Μέσω αυτού, κατέστη δυνατό να γίνει προσομοίωση λειτουργίας με εικόνες προβαλλόμενες μπροστά από την κάμερα.

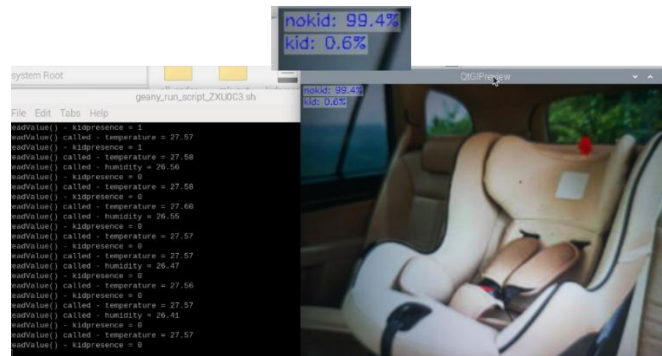
Παρακάτω φαίνονται κάποια δείγματα φωτογραφιών από την διαδικασία αυτή, με ποσοστά πάνω στην προεπισκόπηση της κάμερας. Από 50 φωτογραφίες με παιδιά και 50 με τυχαίους συνδυασμούς αντικειμένων, ενηλίκων και τοπίων, προέκυψαν τα εξής αποτελέσματα:

- Φωτογραφίες με **παρουσία παιδιού**:
 - 41 φωτογραφίες παιδιών αναγνωρίστηκαν σωστά με ποσοστό βεβαιότητας άνω του 95%.
 - 7 φωτογραφίες παιδιών αναγνωρίστηκαν σωστά με ποσοστό βεβαιότητας 80-94%
 - 2 φωτογραφίες παιδιών ΔΕΝ αναγνωρίστηκαν σωστά με ποσοστό βεβαιότητας 45% και 48%.
- Φωτογραφίες **χωρίς παρουσία παιδιού**:
 - 25 φωτογραφίες χωρίς παιδί είχαν ποσοστό βεβαιότητας ταξινόμησης ως παιδί κάτω από 0-19%.
 - 10 φωτογραφίες χωρίς παιδί είχαν ποσοστό βεβαιότητας ταξινόμησης ως παιδί κάτω από 20-30%.
 - 8 φωτογραφίες χωρίς παιδί είχαν ποσοστό βεβαιότητας ταξινόμησης ως παιδί κάτω από 30-50%.
 - 7 φωτογραφίες αναγνωρίστηκαν λανθασμένα ως παιδί.
 - 3 εξ αυτών με ποσοστά:
 - Φωτογραφία με αρκουδάκια σε παιδικό κάθισμα 99.9%.
 - Φωτογραφία με μπουλ φρούτων σε χαμηλό φωτισμό 67%.
 - Φωτογραφία πλήθους κόσμου σε πάρκο (χωρίς παρουσία παιδιών) 56%.

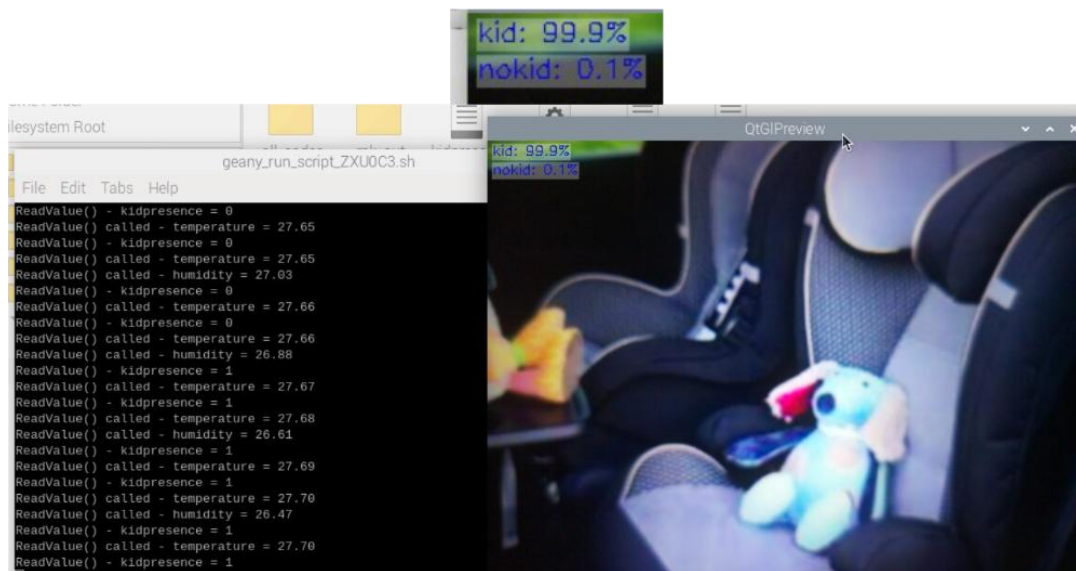


Εικόνα 7-13: Στιγμιότυπα οθόνης από την ταξινόμηση εικόνων από τις δοκιμές (με παρουσία παιδιού)

Κεφάλαιο 7ο:



Εικόνα 7-14: Στιγμιότυπα οθόνης από την ταξινόμηση εικόνων από τις δοκιμές (χωρίς παρουσία παιδιού)



Εικόνα 7-15: Στιγμιότυπα οθόνης από την ταξινόμηση εικόνων από τις δοκιμές (Λανθασμένα θετική ταξινόμηση ως εικόνα με παρουσία παιδιού)

Από τα παραπάνω δεδομένα προκύπτει πως:

- Το σύστημα είναι πιο καλά εκπαιδευμένο στο να ανιχνεύει την παρουσία παιδιού (96%).
- Λιγότερο αποτελεσματικό στο να μην κάνει λανθασμένες ταξινομήσεις φωτογραφιών χωρίς παιδιά (14% πιθανότητα λανθασμένης ταξινόμησης ως παρουσία παιδιού στην εικόνα).
- Συνολικό ποσοστό σωστής ταξινόμησης: 91%

Αυτά τα ποσοστά βέβαια, δεν ανταποκρίνονται απόλυτα στις πραγματικές συνθήκες, διότι χρειάζεται μεγαλύτερο δείγμα για να βγουν αξιόπιστα ποσοστά.

Παράλληλα με την ανίχνευση παρουσίας έγιναν και δοκιμές της εφαρμογής, με αυτήν να λειτουργεί όπως έχει προγραμματιστεί. Οι ειδοποιήσεις, τα μηνύματα, οι ρυθμίσεις όλα λειτουργούν με τον προκαθορισμένο τρόπο. Μόνο πρόβλημα που παρατηρήθηκε ήταν κάποιες φορές αργή ανταπόκριση στην αλλαγή κατάστασης της παρουσίας του παιδιού, όμως αυτό δικαιολογείται λόγω του φόρτου που είχε την δεδομένη χρονική στιγμή ο μικροϋπολογιστής, λειτουργώντας ταυτόχρονα με γραφικό περιβάλλον, απομακρυσμένη σύνδεση και προβολή των δεδομένων της κάμερας (και εικόνα

και στατιστικά). Από αυτό προκύπτει ως λογικό συμπέρασμα ότι η καθυστέρηση προκύπτει από τον μικροϋπολογιστή και όχι από την εφαρμογή.

Κατά την εκκίνηση του μικροϋπολογιστή, μετά από μέτρηση 5 εκκινήσεων, υπολογίστηκε μέσος χρόνος εμφάνισης στην εφαρμογή του ονόματος της συσκευής και επιτυχημένης σύνδεσης το 1 λεπτό και 29 δευτερόλεπτα και μέσος χρόνος έναρξης μετάδοσης δεδομένων τα 2 λεπτά και 6 δευτερόλεπτα.

Διεξήχθησαν και δοκιμές πεδίου, με το σύστημα να παραδίδεται σε γονέα παιδιού 8 ετών και να εγκαθίσταται για δοκιμή 4 ημερών. Στόχος ήταν να διεξαχθούν παραπάνω δοκιμές και προσομοιώσεις καταστάσεων αλλά αυτό δεν κατέστη δυνατόν λόγω φόρτου εργασίας του γονέα. Τα αποτελέσματα όμως, ήταν ενθαρρυντικά, με την αναφορά ενός λανθασμένου συναγερμού και ειδοποίησης του γονέα στις 11 μετακινήσεις και με το σύστημα να λειτουργεί και για την ειδοποίηση θερμοκρασίας, που αρχικά είχε οριστεί στους 30 βαθμούς κελσίου και έπειτα στους 35. Έγινε δοκιμή αποστολής SMS με το σύστημα να αποστέλλει σωστά το μήνυμα με έγκυρη τοποθεσία μετά το πέρας των 30 δευτερολέπτων χωρίς απόκριση.

7.4 Αξιολόγηση και Σύγκριση με Υπάρχουσες Λύσεις

Στην αγορά, κυρίως εκτός Ευρώπης υπάρχουν συστήματα που επιτελούν τον σκοπό που κλήθηκε να επιτελέσει το σύστημα που υλοποιήθηκε σε αυτή την διπλωματική εργασία. Τα περισσότερα με εντελώς διαφορετικές προσεγγίσεις. Παρακάτω θα παρουσιαστούν αυτά που έχουν το περισσότερο ενδιαφέρον και είναι και αυτά που είναι πιο διαδεδομένα και διαθέσιμα.

SensorSafe [30]

Αισθητήρας κλιπ, ο οποίος είναι συμβατός με συγκεκριμένα προϊόντα της εταιρίας 'evenflo'. Λειτουργεί με μπαταρία, ανιχνεύει αν είναι κουμπωμένος και λειτουργεί επιπρόσθετα της ζώνης του παιδικού καθίσματος.

Ανιχνεύει ξεκούμπωμα ζώνης παιδικού καθίσματος, θερμοκρασία περιβάλλοντος, παρατεταμένη παραμονή του παιδιού στο κάθισμα, ανίχνευση σβησίματος κινητήρα μέσω της θύρας OBD II και λειτουργία ειδοποίησης μέσω εφαρμογής.

Μέσω της θύρας OBD II επίσης, μπορεί να ηχήσει την κόρνα.

Στέλνει SMS σε περίπτωση αγνόησης του μηνύματος προειδοποίησης παρουσίας ανεπίβλεπτου παιδιού στο αμάξι.

Elepho eClip [31]

Πομπός-αισθητήρας, ο οποίος εγκαθίσταται σε όποιο σημείο του αυτοκινήτου επιλέξει ο χρήστης. Μετά την σύνδεση με Bluetooth, ειδοποιεί αν απομακρυνθεί ο χρήστης πέραν των 8 περίπου μέτρων. Λειτουργεί με εφαρμογή και έχει και ειδοποίηση για θερμοκρασία.

Ride&Remind[32]

Σύστημα με ανίχνευση ανοίγματος πίσω πόρτας, γίνεται εγκατάσταση χρησιμοποιώντας τους υπάρχοντες αισθητήρες ανοίγματος των πίσω πορτών με επέμβαση στην καλωδίωση. Διαθέσιμο μόνο Αμερική και Καναδά.

Kars4Kids Safety App[33]

Κεφάλαιο 7ο:

Εφαρμογή με υπενθυμίσεις αποκλειστικά για αυτό το σκοπό, μη επεμβατική, χωρίς αισθητήρες και αποκλειστική λειτουργία υπό ευθύνη του χρήστη, ορισμός ειδοποίησης χειροκίνητα χωρίς κάποιον αισθητήρα.

Sense A Life[34]

Προϊόν το οποίο εκτέθηκε προς χρηματοδότηση, το 2016, για την παραγωγή του, στην πλατφόρμα χρηματοδότησης από το κοινό ‘Kickstarter’. Τελικά δεν δημιουργήθηκε. Έχει αισθητήρα υπέρυθρων για την παρουσία οδηγού και επικοινωνία Bluetooth. Η ανίχνευση παρουσίας παιδιού γίνεται μέσω αισθητήρα βάρους. Θα είχε εφαρμογή και ηχητική ειδοποίηση μέσω μεγάρφωνου ενσωματωμένου στην συσκευή.

SaferCHILD ElitePad[35]

Έξυπνο μαξιλαράκι, για τοποθέτηση παιδικό κάθισμα. Ανιχνεύει παρουσία από το βάρος του παιδιού, έχει επιπλέον αναμεταδότη για την ανίχνευση της απόστασης και στέλνει ειδοποιήσεις μέσω εφαρμογής.

Μη εμπορικά διαθέσιμο, μελέτη – προσέγγιση.

DeepCPD: Deep Learning Based In-Car Child Presence Detection Using WiFi [36]

Αυτή η προσέγγιση χρησιμοποιεί σήματα ασύρματου δικτύου WiFi για να ανιχνεύσει παρουσία παιδιού και διάκρισή της από την παρουσία ενήλικου. Η επεξεργασία γίνεται τοπικά με εκπαιδευμένο μοντέλο συνελκτικών δικτύων και στην μελέτη παρουσιάζει 91.45% σωστή ανίχνευση παρουσίας παιδιού και 93.28% σωστή ανίχνευση ενήλικα. Δεν είναι τελικό προϊόν ή σύστημα ανίχνευσης και ειδοποίησης. Είναι σύστημα αποκλειστικά ανίχνευσης, το οποίο θα μπορούσε να υιοθετηθεί και χρησιμοποιηθεί απευθείας από τον κατασκευαστή του εκάστοτε αυτοκινήτου.

Σύμφωνα με τα στοιχεία των συστημάτων, θα γίνει σύγκριση με το σύστημα που υλοποιήθηκε στο πλαίσιο της διπλωματικής εργασίας και για ευκολία ονομάστηκε KidGuard.

Το KidGuard σαν προσέγγιση, όπως και κάποια άλλα συστήματα, εξαιρουμένου του επιθέματος ανίχνευσης βάρους ‘SaferCHILD ElitePad’, δεν επεμβαίνει στο ίδιο το κάθισμα οπότε δεν επηρεάζει την αξιοπιστία της συμπεριφοράς του καθίσματος σε περίπτωση ατυχήματος. Επίσης, μπορεί να προσαρμοστεί σε κάθε αυτοκίνητο και κάθε παιδικό κάθισμα. Δεν γίνεται επέμβαση σε λειτουργικά μέρη του οχήματος, με αυτό να είναι αρνητικό μόνο στην περίπτωση της σύγκρισης με το ‘SensorSafe’ το οποίο καταφέρνει και έχει την λειτουργία ειδοποίησης και μέσω της ήχησης της κόρνας του αυτοκινήτου.

Στην πρόταση χρήσης WiFi για τον εντοπισμό παρουσίας παιδιού εντοπίζεται μια πολύ καλή προσέγγιση στο κομμάτι της ανίχνευσης, με πιο πολλές δυνατότητες, διότι δεν περιορίζεται από φωτεινότητα, γωνίες λήψεις ή εμπόδια στον χώρο.

Πίνακας 7-1: Σύγκριση των συστημάτων για προστασία παιδιού σε παιδικό κάθισμα αυτοκινήτου

Σύστημα	Τεχνολογία Ανίχνευσης	Τρόπος Εγκατάστασης	Ειδοποιήσεις	Επέμβαση σε Κάθισμα/Όχημα	Διαθεσιμότητα	Πλεονεκτήματα	Μειονεκτήματα
KidGuard (Υλοποιημένο Σύστημα)	Κάμερα & Τεχνητή Νοημοσύνη (τοπική επεξεργασία)	Μη επεμβατική, τοποθέτηση σε οποιοδήποτε σημείο, ανάγκη οπτικής επαφής	Εφαρμογή smartphone, ειδοποιήσεις push, SMS	Όχι	Πρωτότυπο	Προσαρμοστικότητα σε όλα τα καθίσματα/οχήματα, χωρίς επέμβαση, τοπική επεξεργασία	Χρήζει βελτίωσης στην ταξινόμηση, εξαρτάται από φωτισμό/γωνία λήψης
SensorSafe	Αισθητήρας κλιπ σε ζώνη καθίσματος	Συμβατό με συγκεκριμένα προϊόντα Evenflo, σύνδεση με OBD II	Εφαρμογή, SMS, κόρνα αυτοκινήτου	Ναι (ζώνη καθίσματος & OBD II)	Εμπορικά διαθέσιμο	Πολλαπλές λειτουργίες (θερμοκρασία, ξεκούμπωμα), ήχηση κόρνας	Περιορισμένη συμβατότητα, επέμβαση σε κάθισμα
Elepho eClip	Πομπός Bluetooth	Τοποθέτηση οπουδήποτε στο αυτοκίνητο	Εφαρμογή (ειδοποίηση απόσταση ~8m), θερμοκρασία	Όχι	Εμπορικά διαθέσιμο	Απλή εγκατάσταση, μη επεμβατικό	Βασίζεται σε απόσταση Bluetooth, όχι άμεση ανίχνευση παιδιού
Ride&Remind	Αισθητήρες ανοίγματος πίσω πόρτας	Επέμβαση στην καλωδίωση πορτών	Ειδοποιήσεις βάσει ανοίγματος πόρτας	Ναι (καλωδίωση)	Αμερική & Καναδάς μόνο	Χρήση υπαρχόντων αισθητήρων	Επέμβαση σε όχημα, περιορισμένη διαθεσιμότητα
Kars4Kids Safety App	Χειροκίνητες υπενθυμίσεις	Εφαρμογή μόνο	Υπενθυμίσεις εφαρμογής	Όχι	Εμπορικά διαθέσιμο	Χωρίς κόστος υλικού, απλή	Χειροκίνητη λειτουργία, καμία αυτόματη ανίχνευση
Sense A Life	Αισθητήρας βάρους & υπέρυθρα	Τοποθέτηση στο κάθισμα	Εφαρμογή, ηχητική ειδοποίηση	Μερικώς (κάθισμα)	Δεν παρήχθη ποτέ	Συνδυασμός τεχνολογιών	Δεν υλοποιήθηκε, αποτυχία χρηματοδότησης
SaferCHILD ElitePad	Αισθητήρας βάρους σε μαξιλάρι	Μαξιλάρι στο παιδικό κάθισμα	Εφαρμογή, αναμεταδότης απόστασης	Ναι (μαξιλάρι στο κάθισμα)	Εμπορικά διαθέσιμο	Άμεση ανίχνευση παρουσίας	Επέμβαση σε κάθισμα, πιθανότητα επηρεασμού απόδοσης σε περίπτωση ατυχήματος
DeepCPD (WiFi)	Σήματα WiFi & CNN (Deep Learning)	Τοπική επεξεργασία	Ανίχνευση μόνο (91.45% παιδί, 93.28% ενήλικας)	Όχι	Ερευνητική μελέτη	Δεν επηρεάζεται από φωτισμό/εμπόδια, υψηλή ακρίβεια	Μόνο ανίχνευση (όχι ειδοποιήσεις), όχι τελικό προϊόν

Τελικώς, η συγκεκριμένη προσέγγιση με την χρήση κάμερας και τεχνητής νοημοσύνης με επεξεργασία δεδομένων σε τοπικό επίπεδο, έχει χαρακτηριστικά αντίστοιχα με εμπορικά συστήματα όσον αφορά τις ειδοποιήσεις και την πλευρά του χρήστη, όμως σημειώνει το προτέρημα προσαρμοστικότητας σε διαφορετικά παιδικά καθίσματα και οχήματα, χωρίς την επέμβαση στην λειτουργία τους. Αυτό φυσικά, δεν σημαίνει ότι δεν χρήζει βελτιώσεων και αξιοποίησης λειτουργιών όπως κάνουν άλλα συστήματα. Ένα από τα τμήματα που διαπιστώθηκε ότι πρέπει να βελτιωθεί είναι η ταξινόμηση εικόνων μέσω τεχνητής νοημοσύνης, αυτό μπορεί να γίνει αυξάνοντας τα διαθέσιμα δεδομένα εκπαίδευσης του νευρωνικού δικτύου. Χρειάζεται ένας αρκετά μεγαλύτερος αριθμός εικόνων με αρκετές εναλλαγές περιβάλλοντος και γωνίες λήψης.

7.5 Επίλογος

Συνοπτικά, στο κεφάλαιο αυτό έγινε η αναφορά και παρουσίαση των σχεδίων 3D και εκτύπωσης αυτών, ακολούθησε η παρουσίαση με εικόνες της διαδικασίας συναρμολόγησης του υποσυστήματος με τον μικροϋπολογιστή και τα περιφερειακά του μέσα στο εκτυπωμένο περίβλημα. Παρουσιάστηκαν οι δοκιμές λειτουργίας που διεξήχθησαν, με την ανάλυση και ερμηνεία αυτών. Τέλος, έγινε αναφορά των εμπορικά διαθέσιμων συστημάτων που έχουν παρόμοιο τελικό σκοπό, δηλαδή την προστασία των παιδιών στο αυτοκίνητο. Με την αναφορά αυτών, παράλληλα έγινε και η σύγκριση τους με το υλοποιημένο σύστημα της διπλωματικής και η ανάδειξη ελλείψεων η προτερημάτων του συστήματος.

Κεφάλαιο 7ο:

Με τις δοκιμές και την ανάλυση αυτών να έχουν ολοκληρωθεί, μπορεί να γίνει μια αξιολόγηση της αποτελεσματικότητας του συστήματος και της αξιοπιστίας που μπορεί να προσφέρει. Στο επόμενο κεφάλαιο θα παρουσιαστούν τα συμπεράσματα που προέκυψαν από την έρευνα και την υλοποίηση, καθώς και οι προτάσεις για μελλοντική ανάπτυξη και πιθανές βελτιώσεις του συστήματος, ώστε να καταστεί ακόμη πιο λειτουργικό.

Κεφάλαιο 8ο: Συμπεράσματα και Προτάσεις Μελλοντικής Ανάπτυξης

8.1 Εισαγωγή

Σε αυτό το κεφάλαιο θα γίνει μια σύνοψη της διαδικασίας για την υλοποίηση της παρούσας διπλωματικής εργασίας, καθώς και ανάλυση των συμπερασμάτων που προέκυψαν. Επίσης, μέσω της διαδικασίας υλοποίησης παρατηρήθηκαν σημεία που χρήζουν βελτιώσεις και σημεία με δυνατότητες που δεν έχουν αξιοποιηθεί στην συγκεκριμένη προσέγγιση, οπότε θα γίνει μια αναφορά αυτών, επεξήγηση της χρησιμότητάς τους και μικρή ανάλυση της μελλοντικής αυτής προσέγγισης.

8.2 Σύνοψη και Συμπεράσματα

Η παρούσα διπλωματική εργασία είχε ως στόχο την μελέτη και υλοποίηση μιας λύσης στο πρόβλημα της ελλιπούς ασφάλειας σε περίπτωση που παραμείνει ένα παιδί στο παιδικό κάθισμα σε αυτοκίνητο. Στα προηγούμενα κεφάλαια έχει γίνει παρουσίαση τόσο του θεωρητικού υπόβαθρου και της μελέτης εναλλακτικών τρόπων για την ανίχνευση παρουσίας παιδιού στο παιδικό κάθισμα, όσο και ανάλυση της προσέγγισης που επιλέχθηκε, με χρήση τεχνητής νοημοσύνης. Πιο συγκεκριμένα αναλύθηκε πώς έγινε η ανάπτυξη μοντέλου συνελκτικών νευρωνικών δικτύων για την ενσωμάτωση σε σύστημα τοπικής επεξεργασίας και κατά επέκταση δημιουργία συστήματος τεχνητής νοημοσύνης παρυφής. Εκτός αυτού αναλύθηκε η υλοποίηση του υποσυστήματος του μικροϋπολογιστή Raspberry και επικοινωνίας BLE με εφαρμογή Android και η υλοποίηση και οι λειτουργίες της εφαρμογής. Επιπλέον, έγινε περιγραφή των δοκιμών, των αποτελεσμάτων που προέκυψαν από αυτές, καθώς και σύγκριση με υπάρχοντα εμπορικά συστήματα για την τελική αξιολόγηση του συστήματος.

Βάση των αποτελεσμάτων και της ανάλυσης αυτών, μπορεί να γίνει εξαγωγή των παρακάτω συμπερασμάτων:

- Η προσέγγιση με μη επεμβατική, στο παιδικό κάθισμα, λύση είναι εφικτή και υλοποιήσιμη.
- Δεν υπάρχει εξάρτηση από τον τύπο αυτοκινήτου ή του παιδικού καθίσματος.
- Ο συνδυασμός των τεχνολογιών τεχνητής νοημοσύνης παρυφής με διαδίκτυο των πραγμάτων και εφαρμογή Android, μπορεί να δημιουργήσει ένα αξιόπιστο σύστημα με χρηστικότητα και μπορεί να παρέχει πολλές δυνατότητες για την διασφάλιση της υγείας του παιδιού μέσα στον χώρο του αυτοκινήτου.
- Το σύστημα ήδη μπορεί και ανταποκρίνεται με μεγάλο βαθμό αξιοπιστίας ακόμα και σε δοκιμές χρήσης πραγματικών συνθηκών.
- Ανάγκη για βελτιστοποίηση του συστήματος και συγκεκριμένα των ποσοστών βεβαιότητας ταξινόμησης με παραπάνω εκπαίδευση.
- Δυνατότητες επέκτασης και αύξησης των δυνατοτήτων.

8.3 Προτάσεις Μελλοντικής Εξέλιξης

Η συγκεκριμένη προσέγγιση διαθέτει πολλές δυνατότητες, οι οποίες δεν μελετήθηκαν ή εξελίχθηκαν, που θα μπορούσαν να αποτελέσουν ιδιαίτερα σημαντικά χαρακτηριστικά για την διασφάλιση της προστασίας παιδιών στα αυτοκίνητα.

Αρχικά όπως ήδη αναφέρθηκε, η αξιοπιστία και η δυνατότητα για ταξινόμηση των εικόνων με μεγαλύτερη ακρίβεια μπορεί να επιτευχθεί, αυτό θα χρειαστεί περισσότερο χρόνο και εντατική αναζήτηση και συλλογή εικόνων ή με την χρήση οικονομικών πόρων μπορεί να γίνει αγορά έτοιμων

Κεφάλαιο 8ο:

συνόλων δεδομένων από βάσεις. Επίσης θα μπορούσε να γίνει παράλληλη χρήση πολλαπλών τεχνολογιών για την ανίχνευση της παρουσίας, όπως για παράδειγμα της τεχνολογίας mmWave . Με αυτόν τον τρόπο η αξιοπιστία θα ήταν πολύ μεγαλύτερη και θα επαλήθευε η μία τεχνολογία την άλλη. Παράλληλα με την χρήση αυτής της τεχνολογίας θα μπορούσαν να παρέχονται και δεδομένα ανίχνευσης καρδιακών παλμών και να παραχθούν δεδομένα για την υγεία του παιδιού με αυτόν τον τρόπο.

Σε περίπτωση περαιτέρω εκπαίδευσης του μοντέλου, θα μπορούσαν να ανιχνεύονται επίσης δεδομένα όσον αφορά την υγεία του παιδιού, από τις εκφράσεις, χρωματισμούς του δέρματος ή κινήσεις. Ακόμα με εκπαίδευση του μοντέλου θα μπορούσε να παρατηρηθεί η σωστή χρήση της ζώνης και η κατάσταση αυτής, έτσι θα μπορούσε να ειδοποιηθεί ο χρήστης εφόσον αυτή ξεκούμπωσε είτε από αστοχία είτε από το ίδιο το παιδί. Με αυτόν τον τρόπο θα παρατηρούνταν η υγεία του παιδιού και η ασφάλειά του και κατά την διάρκεια της παραμονής στο αυτοκίνητο.

Η σμίκρυνση της συσκευής θα μπορούσε επίσης να είναι μια βελτίωση, ο χώρος που παραμένει ανεκμετάλλετος μέσα στο περίβλημα της συσκευής δείχνει ότι δεν αξιοποιήθηκαν στο έπακρο τα περιθώρια για την μείωση μεγέθους και μπορεί να αλλάξει κατά πολύ η μορφή της συσκευής.

Τέλος, η πιο ιδανική περίπτωση θα ήταν η ενσωμάτωση του συστήματος στο ίδιο το αυτοκίνητο από την παραγωγή του, με συνδυασμό τεχνολογιών και δυνατοτήτων του αυτοκινήτου. Ο συναγερμός θα μπορούσε να χρησιμοποιηθεί για ειδοποίηση του χρήστη, οι αισθητήρες συναγερμού επίσης θα μπορούσαν να έχουν διπλή λειτουργία, ανίχνευση του παιδιού στην μια λειτουργία και ανίχνευση παράβασης στην άλλη.

Οι Δυνατότητες είναι πολλές, για την υλοποίησή τους χρειάζεται μελέτη, χρόνος, οικονομική υποστήριξη και περισσότερα άτομα, ώστε να υπάρξει εξειδίκευση και καλύτερη χρήση των επιμέρους τεχνολογιών που αν και μπορούν να γίνουν από ένα άτομο, δεν γίνεται η αξιοποίησή τους στο έπακρο.

8.4 Επίλογος

Σε αυτό το κεφάλαιο, έγινε σύνοψη της όλης διαδικασίας για την υλοποίηση της διπλωματικής εργασίας. Παρουσιάστηκαν τα τελικά συμπεράσματα για την αξιολόγηση της μελέτης και υλοποίησης του συστήματος και δόθηκαν προτάσεις και ιδέες για την πιθανή εξέλιξη του συστήματος με καλύτερη απόδοση και χαρακτηριστικά που θα του απέδιδαν περισσότερες δυνατότητες και βαθμό ευελιξίας και αξιοπιστίας.

Η χρήση τεχνητής νοημοσύνης παρυφής και ενσωματωμένων συστημάτων, όπως παρουσιάζεται και από την παρούσα διπλωματική, μπορεί να αποτελέσει μια αποτελεσματική και ταυτόχρονα αξιόπιστη λύση, ώστε να διασφαλιστεί περαιτέρω η ασφάλεια των παιδιών κατά τη μετακίνησή τους με αυτοκίνητο. Με την μελέτη και την υλοποίηση της συγκεκριμένης διπλωματικής εργασίας, αποδεικνύεται για άλλη μια φορά πως με την αξιοποίηση νέων τεχνολογιών σε ένα πρόβλημα με πραγματικό κοινωνικό αντίκτυπο, είναι δυνατόν να δοθούν και νέες προσεγγίσεις και λύσεις. Με αυτό τον τρόπο θέτονται και οι βάσεις για περαιτέρω έρευνα επί του θέματος.

Τελικώς, η διπλωματική εργασία έχοντας ολοκληρωθεί, μπορεί να παρουσιαστεί, όχι μόνο ως μια τεχνική μελέτη αλλά και ως μια πρόταση για αύξηση της οδικής ασφάλειας στο πεδίο των μετακινήσεων παιδιών και της διαφύλαξης της ανθρώπινης ζωής.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1]

M. D. Chandler, P. G. Schnitzer, H. K. Dykstra, and J. Morag MacKay, “Pediatric vehicular heatstroke: An analysis of 296 cases from the National Fatality Review Case Reporting System,” *Traffic injury prevention (Online)/Traffic injury prevention*, vol. 25, no. 3, pp. 400–406, Dec. 2023, doi: <https://doi.org/10.1080/15389588.2023.2290454>.

[2]

F. S. Alowirdi, S. A. Al-harbi, O. Abid, O. S. Aldibasi, and S. F. Jamil, “Assessing parental awareness and attitudes toward leaving children unattended inside locked cars and the risk of vehicular heat strokes,” *International Journal of Pediatrics and Adolescent Medicine*, Nov. 2019, doi: <https://doi.org/10.1016/j.ijpam.2019.11.004>.

[3]

T. Canada, “Third-party aftermarket products for children’s restraint systems,” *00000000*, Jan. 30, 2019. <https://tc.canada.ca/en/road-transportation/defects-recalls-vehicles-tires-child-car-seats/third-party-aftermarket-products-children-s-restraint-systems>

[4]

“The Hidden Danger of Supplemental Car Seat Products,” *Center for Injury Research and Prevention*, Feb. 16, 2021. <https://injury.research.chop.edu/blog/posts/hidden-danger-supplemental-car-seat-products>

[5]

“Strengthening passenger rights within the European Union | EUR-Lex,” *Europa.eu*, 2025. <https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=LEGISSUM%3A124124> (accessed Sep. 21, 2025).

[6]

H. Rahnama, “AI Glossary/Dictionary – MIT Media Lab,” *MIT Media Lab*, 2025. <https://www.media.mit.edu/tools/ai-glossary-dictionary/>

[7]

A. Turing, “Computing Machinery and Intelligence,” *Mind*, vol. 59, no. 236, pp. 433–460, Oct. 1950, doi: <https://doi.org/10.1093/mind/LIX.236.433>.

[8]

“What is CAPTCHA? - Google Workspace Admin Help,” *support.google.com*. <https://support.google.com/a/answer/1217728>

[9]

J. Delua, “Supervised vs. Unsupervised learning: What’s the difference? | IBM,” *www.ibm.com*, Mar. 12, 2021. <https://www.ibm.com/think/topics/supervised-vs-unsupervised-learning>

[10]

J. Murel and E. Kavlakoglu, "Reinforcement Learning," *Ibm.com*, Mar. 25, 2024. <https://www.ibm.com/think/topics/reinforcement-learning>

[11]

R. E. Uhrig, "Introduction to artificial neural networks," *Proceedings of IECON '95 - 21st Annual Conference on IEEE Industrial Electronics*, vol. 1, 1995, doi: <https://doi.org/10.1109/iecon.1995.483329>.

[12]

V. Wiley and T. Lucas, "Computer Vision and Image Processing: A Paper Review," *International Journal of Artificial Intelligence Research*, vol. 2, no. 1, p. 22, Jun. 2018, doi: <https://doi.org/10.29099/ijair.v2i1.42>.

[13]

B. Liu, L. Yu, C. Che, Q. Lin, H. Hu, and X. Zhao, "Integration and Performance Analysis of Artificial Intelligence and Computer Vision Based on Deep Learning Algorithms," *arXiv.org*, Dec. 20, 2023. <https://arxiv.org/abs/2312.12872>

[14]

A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, May 2012.

[15]

M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Jun. 2018, doi: <https://doi.org/10.1109/cvpr.2018.00474>.

[16]

A. G. Howard *et al.*, "MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications," *arXiv.org*, Apr. 17, 2017. <https://arxiv.org/abs/1704.04861>

[17]

L. Wei, Z. Ma, C. Yang, and Q. Yao, "Advances in the Neural Network Quantization: A Comprehensive Review," *Applied Sciences*, vol. 14, no. 17, p. 7445, Aug. 2024, doi: <https://doi.org/10.3390/app14177445>.

[18]

M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. van Baalen, and T. Blankevoort, "A White Paper on Neural Network Quantization," *arXiv:2106.08295 [cs]*, Jun. 2021, Available: <https://arxiv.org/abs/2106.08295>

[19]

"Kevin Ashton Invents the Term 'The Internet of Things' : History of Information," *www.historyofinformation.com*. <https://www.historyofinformation.com/detail.php?id=3411>

[20]

"IoT connections worldwide 2034| Statista," *Statista*, 2025. <https://www.statista.com/statistics/1183457/iot-connected-devices->

worldwide/?srsltid=AfmBOoqwBHAI4uW3r_ruI3oVnwUe6hyDeBQzSLtjhKROSKAx4OXv6mjs (accessed Sep. 23, 2025).

[21]

M. Mattioli and M. Mattioli, “The Apollo Guidance Computer,” *IEEE Micro*, vol. 41, no. 6, pp. 179–182, Nov. 2021, doi: <https://doi.org/10.1109/mm.2021.3121103>.

[22]

“How low energy is bluetooth low energy? Comparative measurements with ZigBee/802.15.4 | IEEE Conference Publication | IEEE Xplore,” *ieeexplore.ieee.org*. <https://ieeexplore.ieee.org/document/6215496>

[23]

StatCounter, “Mobile Operating System Market Share Worldwide,” *StatCounter Global Stats*, Aug. 2024. <https://gs.statcounter.com/os-market-share/mobile/worldwide>

[24]

D. Fang, L. Tian, M. Li, Y. Wang, J. Liu, and X. Chen, “Factory Worker Safety Mask Monitoring System Based on Deep Learning and AI Edge Computing,” pp. 96–100, Nov. 2022, doi: <https://doi.org/10.1109/icmra56206.2022.10145674>.

[25]

D. Neels, S. Ramesh, A. N. Ahamed, S. Baskar, C. Ashok, and M. Charan, “Intelligent Edge-Based System for Driver Drowsiness Monitoring in Mobile Crowd-sourcing Environments,” pp. 252–256, Dec. 2024, doi: <https://doi.org/10.1109/icmacc62921.2024.10894633>.

[26]

“SVIRO: Synthetic Vehicle Interior Rear Seat Occupancy Dataset and Benchmark,” *typeset.io*, Mar. 01, 2020. <https://typeset.io/papers/sviro-synthetic-vehicle-interior-rear-seat-occupancy-dataset-59dchu3u54>

[27]

Iron486, “Children vs Adults Classification,” *Kaggle.com*, 2022. <https://www.kaggle.com/datasets/die9origephit/children-vs-adults-images>

[28]

Sijish, “Face Age Dataset,” *Roboflow*, 2021. <https://universe.roboflow.com/sijish/face-age> (accessed Sep. 29, 2025).

[29]

L. N. Smith, “Cyclical Learning Rates for Training Neural Networks,” *IEEE Xplore*, Mar. 01, 2017. <https://ieeexplore.ieee.org/document/7926641>

[30]

“SensorSafe Bluetooth Car Seat Alerts App and Product Collection | Evenflo® Official Site,” *Evenflo® Company, Inc*, 2025. https://www.evenflo.com/pages/sensorsafe?srsltid=AfmBOop2ptjs-QTjQeY-RKhyEMxShIEJQCj3L2Hu4KniBx9JlktXpC_w (accessed Sep. 29, 2025).

[31]

“eClip,” *Elepho*. <https://elepho.com/products/eclip-baby-reminder-for-your-car>

[32]

“How It Works | Back Seat Reminder System | Ride & Remind,” *Ride and Remind*, Sep. 18, 2024. <https://ridenremind.com/how-it-works/> (accessed Sep. 29, 2025).

[33]

Kars4Kids, “Kars4Kids Safety App - Never Forget Your Kid in the Car,” *Kars4Kids*. <https://www.kars4kids.org/safety-app/>

[34]

“Sense A Life: Saving Children With Smart Sensor Technology.” https://www.kickstarter.com/projects/1077611512/sense-a-life-saving-children-with-smart-sensor-tec?ref=nav_search

[35]

“SaferCHILD Landing Page,” *Carseatmonitorapp.com*, 2021. <http://carseatmonitorapp.com/products.html> (accessed Sep. 29, 2025).

[36]

S. S. Jayaweera, B. Wang, W.-H. Wang, and K. J. R. Liu, “DeepCPD: Deep Learning Based In-Car Child Presence Detection Using WiFi,” *IEEE Journal of Selected Areas in Sensors*, pp. 1–12, 2025, doi: <https://doi.org/10.1109/jsas.2025.3602722>.

ΠΑΡΑΡΤΗΜΑ Α. : Κώδικας Raspberry Pi

```
#!/usr/bin/env python3

from gi.repository import GLib
import dbus
import dbus.service
import dbus.mainloop.glib
import struct
import time
import json, os

import board
import adafruit_ahtx0
import busio

BLUEZ_SERVICE = 'org.bluez'
ADAPTER_PATH = '/org/bluez/hci0'
GATT_MANAGER_IFACE = 'org.bluez.GattManager1'
LE_ADVERTISING_MANAGER_IFACE = 'org.bluez.LEAdvertisingManager1'
ADAPTER_IFACE = 'org.bluez.Adapter1'

SERVICE_UUID = '12345678-1234-5678-1234-56789abcdef0'
CHAR_KID_UUID = '12345678-1234-5678-1234-56789abcdef1'
CHAR_TEMP_UUID = '12345678-1234-5678-1234-56789abcdef2'
CHAR_HUM_UUID = '12345678-1234-5678-1234-56789abcdef3'

SERVICE_PATH = '/org/bluez/example/service0'
CHAR_KID_PATH = SERVICE_PATH + '/char0'
CHAR_TEMP_PATH = SERVICE_PATH + '/char1'
CHAR_HUM_PATH = SERVICE_PATH + '/char2'
ADVERTISING_PATH = '/org/bluez/example/advertisement0'

#read kid/nokid from the camera
def read_kid_from_json(path="~/kid_status.json", threshold=0.5):
    """
    Returns 1 if top label == 'kid' and score >= threshold, else 0.
    On any error/missing file, returns 0.
    """
    try:
        with open(os.path.expanduser(path), "r") as f:
            data = json.load(f)
            top = max(data.get("top", []),
                      key=lambda x: float(x.get("score", 0))) if data.get("top")
    else None
        if not top:
            return 0
        label = str(top.get("label", "")).strip().lower()
```

```

        score = float(top.get("score", 0))
        return 1 if (label == "kid" and score >= threshold) else 0
    except Exception:
        return 0

#AHT20
i2c = busio.I2C(board.SCL, board.SDA)
sensor = adafruit_ahtx0.AHTx0(i2c)

class Advertisement(dbus.service.Object):
    def __init__(self, bus):
        self.path = ADVERTISING_PATH
        dbus.service.Object.__init__(self, bus, self.path)
        print("Advertisement object created.")

    def get_path(self):
        return dbus.ObjectPath(self.path)

    @dbus.service.method("org.freedesktop.DBus.Properties", in_signature='s',
out_signature='a{sv}')
    def GetAll(self, interface):
        return {
            "Type": "peripheral",
            "ServiceUUIDs": [SERVICE_UUID],
            "LocalName": "KidCarGuard",
            "IncludeTxPower": dbus.Boolean(True)
        }

    @dbus.service.method("org.freedesktop.DBus.Properties", in_signature='ss',
out_signature='v')
    def Get(self, interface, prop):
        return self.GetAll(interface)[prop]

    @dbus.service.method("org.bluez.LEAdvertisement1", in_signature='',
out_signature='')
    def Release(self):
        print("Advertisement released.")

class BaseCharacteristic(dbus.service.Object):
    def __init__(self, bus, path, uuid):
        self.path = path
        self.uuid = uuid
        dbus.service.Object.__init__(self, bus, self.path)

    def get_path(self):
        return dbus.ObjectPath(self.path)

    @dbus.service.method("org.freedesktop.DBus.Properties", in_signature="ss",
out_signature="v")

```

```

def Get(self, interface, prop):
    return self.GetAll(interface)[prop]

    @dbus.service.method("org.freedesktop.DBus.Properties", in_signature="s",
out_signature="a{sv}")
    def GetAll(self, interface):
        return {
            "UUID": self.uuid,
            "Service": dbus.ObjectPath(SERVICE_PATH),
            "Flags": ["read"]
        }

class KidPresenceCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_KID_PATH, CHAR_KID_UUID)

    @dbus.service.method("org.bluez.GattCharacteristic1",
in_signature="a{sv}", out_signature="ay")
    def ReadValue(self, options):
        val = read_kid_from_json() # 0 (nokid) or 1 (kid)
        print(f"ReadValue() - kidpresence = {val}")
        return [dbus.Byte(val)]

class TemperatureCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_TEMP_PATH, CHAR_TEMP_UUID)

    @dbus.service.method("org.bluez.GattCharacteristic1",
in_signature="a{sv}", out_signature="ay")
    def ReadValue(self, options):
        temp = sensor.temperature
        packed = struct.pack("<f", float(temp))
        print(f"ReadValue() called - temperature = {temp:.2f}")
        return [dbus.Byte(b) for b in packed]

class HumidityCharacteristic(BaseCharacteristic):
    def __init__(self, bus):
        super().__init__(bus, CHAR_HUM_PATH, CHAR_HUM_UUID)

    @dbus.service.method("org.bluez.GattCharacteristic1",
in_signature="a{sv}", out_signature="ay")
    def ReadValue(self, options):
        hum = sensor.relative_humidity
        packed = struct.pack("<f", float(hum))
        print(f"ReadValue() called - humidity = {hum:.2f}")
        return [dbus.Byte(b) for b in packed]

class KidPresenceService(dbus.service.Object):
    def __init__(self, bus):

```

```

self.path = SERVICE_PATH
self.kid_char = KidPresenceCharacteristic(bus)
self.temp_char = TemperatureCharacteristic(bus)
self.hum_char = HumidityCharacteristic(bus)
dbus.service.Object.__init__(self, bus, self.path)
print("Service object created.")

def get_path(self):
    return dbus.ObjectPath(self.path)

@dbus.service.method("org.freedesktop.DBus.Properties", in_signature="s",
out_signature="a{sv}")
def GetAll(self, interface):
    return {
        "UUID": SERVICE_UUID,
        "Primary": True
    }

@dbus.service.method("org.freedesktop.DBus.Properties", in_signature="ss",
out_signature="v")
def Get(self, interface, prop):
    return self.GetAll(interface)[prop]

class Application(dbus.service.Object):
    def __init__(self, bus):
        self.path = '/'
        self.bus = bus
        dbus.service.Object.__init__(self, bus, self.path)
        self.service = KidPresenceService(bus)
        self.characteristics = [
            self.service.kid_char,
            self.service.temp_char,
            self.service.hum_char
        ]
        print("GATT Application initialized.")

    def get_path(self):
        return dbus.ObjectPath(self.path)

@dbus.service.method("org.freedesktop.DBus.ObjectManager",
in_signature="", out_signature="a{oa{sa{sv}}}")
def GetManagedObjects(self):
    print("Reporting managed objects to BlueZ.")
    response = {}
    svc_path = self.service.get_path()
    response[svc_path] = {
        "org.bluez.GattService1": {
            "UUID": SERVICE_UUID,
            "Primary": True
        }
    }

```

```

    }
}
for char in self.characteristics:
    response[char.get_path()] = {
        "org.bluez.GattCharacteristic1": {
            "UUID": char.uuid,
            "Service": svc_path,
            "Flags": ["read"]
        }
    }
return response

def setup_adapter(bus):
    print("Configuring Bluetooth adapter...")
    props = dbus.Interface(bus.get_object(BLUEZ_SERVICE, ADAPTER_PATH),
"org.freedesktop.DBus.Properties")
    props.Set(ADAPTER_IFACE, "Powered", dbus.Boolean(True))
    props.Set(ADAPTER_IFACE, "Discoverable", dbus.Boolean(True))
    props.Set(ADAPTER_IFACE, "Pairable", dbus.Boolean(True))
    props.Set(ADAPTER_IFACE, "Alias", dbus.String("KidCarGuard"))
    print("Bluetooth adapter is powered and ready.")

def safe_unregister_previous_advertisement(bus):
    try:
        ad_obj = bus.get_object(BLUEZ_SERVICE, ADVERTISING_PATH)
        dbus.Interface(ad_obj, "org.bluez.LEAdvertisement1").Release()
        print("Cleaned up previous stuck advertisement.")
    except Exception as e:
        print("No old advertisement to clean up or error occurred:", e)

def main():
    print("Starting KidPresence BLE GATT server with temp and humidity...")
    dbus.mainloop.glib.DBusGMainLoop(set_as_default=True)
    bus = dbus.SystemBus()

    setup_adapter(bus)
    safe_unregister_previous_advertisement(bus)

    advertisement = Advertisement(bus)
    app = Application(bus)

    adv_manager = dbus.Interface(bus.get_object(BLUEZ_SERVICE, ADAPTER_PATH),
LE_ADVERTISING_MANAGER_IFACE)
    gatt_manager = dbus.Interface(bus.get_object(BLUEZ_SERVICE, ADAPTER_PATH),
GATT_MANAGER_IFACE)

    adv_manager.RegisterAdvertisement(advertisement.get_path(), {},
reply_handler=lambda: print("BLE advertising started"),
error_handler=lambda e: print("BLE advertising failed:", str(e)))

```

```
gatt_manager.RegisterApplication(app.get_path(), {},
    reply_handler=lambda: print("GATT service registered"),
    error_handler=lambda e: print("Failed to register GATT:", str(e)))

Glib.MainLoop().run()

if __name__ == '__main__':
    main()
```

ΠΑΡΑΡΤΗΜΑ Β. :Κώδικας Διαχωρισμού Συνόλου Δεδομένων

```
import random, shutil
from pathlib import Path

# Ρυθμίσεις βασικών φακέλων
DATA = Path(r"C:\img_classifier\Tsilikas_dataset_classification_kid")
SRC = DATA / "dataset_unsplit" # περιέχει kid/ και nokid/
TRAIN, VAL, TEST, CALIB = DATA/"train", DATA/"val", DATA/"test", DATA/"calib"

# Παράμετροι split και calibration
RATIOS = (0.75, 0.15, 0.10) # train, val, test
CALIB_PER_CLASS = 150 # μέγιστος αριθμός εικόνων ανά κλάση
EXT = {".jpg", ".jpeg", ".png", ".bmp", ".webp"}
SEED = 23 # σταθερή τυχαιότητα
COPY = True # True αντιγραφή, False μετακίνηση
CLEAN = True # καθαρισμός φακέλων προορισμού

def imgs(d): return [p for p in d.glob("*") if p.suffix.lower() in EXT]

# Δημιουργία κενού φακέλου
def empty(d):
    if CLEAN and d.exists(): shutil.rmtree(d)
    d.mkdir(parents=True, exist_ok=True)

def split_and_calib():
    random.seed(SEED)
    classes = sorted([p.name for p in SRC.iterdir() if p.is_dir()])

    # Δημιουργία φακέλων train/val/test
    for base in (TRAIN, VAL, TEST):
        for c in classes: empty(base/c)

    # Διαχωρισμός εικόνων ανά κλάση
    for c in classes:
        files = imgs(SRC/c); files.sort(); random.shuffle(files)
        n = len(files); n_tr = int(n*RATIOS[0]); n_va = int(n*RATIOS[1])
        parts = {TRAIN/c: files[:n_tr], VAL/c: files[n_tr:n_tr+n_va], TEST/c:
files[n_tr+n_va:]}
        for dst, items in parts.items():
            dst.mkdir(parents=True, exist_ok=True)
            for p in items:
                (shutil.copy2 if COPY else shutil.move)(p, dst/p.name)
    print("Split ολοκληρώθηκε.")

# Δημιουργία calibration set από train
empty(CALIB)
for c in classes:
```

```
src, dst = TRAIN/c, CALIB/c
dst.mkdir(parents=True, exist_ok=True)
pick = random.sample(imgs(src), min(CALIB_PER_CLASS, len(imgs(src))))
for p in pick: shutil.copy2(p, dst/p.name)
print(f"{c}: {len(pick)} εικόνες στο calib")
print("Calibration ολοκληρώθηκε.")

if __name__ == "__main__":
    split_and_calib()
```