

ΤΕΧΝΟΛΟΓΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΣΧΟΛΗ ΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΜΕΣΩΝ ΕΝΗΜΕΡΩΣΗΣ



Πτυχιακή Εργασία

ΣΥΣΤΗΜΑ ΑΥΤΟΕΚΜΑΘΗΣΗΣ,
ΔΙΑΧΕΙΡΙΣΗΣ ΚΑΙ ΑΝΑΛΥΣΗΣ
ΔΙΑΔΡΟΜΩΝ

Γιάννης Δεσπότης

Λεμεσός 2014

ΤΕΧΝΟΛΟΓΙΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΚΥΠΡΟΥ
ΣΧΟΛΗ ΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΜΕΣΩΝ ΕΝΗΜΕΡΩΣΗΣ
ΤΜΗΜΑ ΕΠΙΚΟΙΝΩΝΙΑΣ & ΣΠΟΥΔΩΝ ΔΙΑΔΙΚΤΥΟΥ

Πτυχιακή Εργασία

ΣΥΣΤΗΜΑ ΑΥΤΟΕΚΜΑΘΗΣΗΣ,
ΔΙΑΧΕΙΡΙΣΗΣ ΚΑΙ ΑΝΑΛΥΣΗΣ
ΔΙΑΔΡΟΜΩΝ

Γιάννης Δεσπότης

Σύμβουλος καθηγητής

Δρ. Κωνσταντίνος Τζιούβας

Λεμεσός 2014

Πνευματικά δικαιώματα

Copyright © Γιάννης Δεσπότης, 2014

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Η έγκριση της πτυχιακής εργασίας από το Τμήμα Επικοινωνίας και Σπουδών Διαδικτύου του Τεχνολογικού Πανεπιστημίου Κύπρου δεν υποδηλώνει απαραίτητως και αποδοχή των απόψεων του συγγραφέα εκ μέρους του Τμήματος.

Θα ήθελα να ευχαριστήσω ιδιαίτερα τον καθηγητή μου Δρ. Κωνσταντίνο Τζιούβα για την καθοδήγηση και την υποστήριξη του καθ' όλη τη διάρκεια διεκπεραίωσης της παρούσας πτυχιακής. Ευχαριστίες θα ήθελα να απευθύνω και στα αγαπημένα μου πρόσωπα για την ηθική υποστήριξη και την συμπαράσταση που μου προσέφεραν όλο αυτό τον καιρό.

Περίληψη

Ο ρυθμός ανάπτυξης και χρήσης νέων τεχνολογικών προϊόντων τα τελευταία χρόνια αναπτύσσεται ραγδαία. Συγκεκριμένα, τα κινητά τηλέφωνα έχουν πλέον μεταμορφωθεί σε κομψές, έξυπνες κινητές συσκευές οι οποίες ενσωματώνουν χαρακτηριστικά ενός ηλεκτρονικού υπολογιστή. Έτσι, πέρα από την βασική τους λειτουργία, προσφέρουν στον χρήστη την ευελιξία να πραγματοποιεί διάφορες διεργασίες οι οποίες εναλλακτικά η χρήση ενός ηλεκτρονικού υπολογιστή θα ήταν αναγκαία. Επιπλέον, οι πρόσφατες αυτές κινητές συσκευές, παρέχουν την βάση για την ανάπτυξη νέων έξυπνων συστημάτων και εφαρμογών που θα αξιοποιούσαν τις δυνατότητες που παρέχουν. Η παρούσα έρευνα, επικεντρώνεται στην συλλογή, ανάλυση και χρήση δεδομένων διαδρομών που συλλέχθηκαν από διάφορα οχήματα κατά την διαδικασία κίνησης τους για τη δημιουργία μιας υποδομής πάνω στην οποία θα μπορούν να αναπτυχθούν εφαρμογές οδικής κυκλοφορίας. Για την δημιουργία οποιασδήποτε εφαρμογής σχετικής με την οδική κυκλοφορία, απαραίτητη προϋπόθεση είναι ο σαφής διαχωρισμός των οδικών σημείων σε δύο κατηγορίες: οδικοί κόμβοι (junctions), και υπόλοιπα σημεία (non-junctions). Πρωταρχικός σκοπός της έρευνας είναι η δημιουργία μιας αυτοματοποιημένης μεθόδου αναγνώρισης κόμβων οδικών αρτηριών αξιοποιώντας αποκλειστικά ένα σύνολο δεδομένων που προέκυψε από την κίνηση των οχημάτων και συλλέχθηκε με τη χρήση Crowdsourcing. Επιπλέον, σκοπός αυτής της εργασίας αποτελεί η δημιουργία μιας εφαρμογής, η οποία σε συνδυασμό με τα αποτελέσματα της ανάλυσης των δεδομένων, θα θέσει τις βάσεις μιας υλοποίησης η οποία θα έχει την δυνατότητα να βοηθά τους χρήστες στην διακίνηση με τα οχήματα τους. Με την ανάλυση του συνόλου των δεδομένων που έχουν καταγραφεί, αποδείχθηκε ότι τα σημεία κόμβων, μπορούν να αναγνωριστούν αυτόματα, με ένα πολύ υψηλό ποσοστό επιτυχίας.

Πίνακας Περιεχομένων

Κατάλογος Πινάκων	vi
Κατάλογος Εικόνων.....	vii
Συνομογραφίες	ix
Απόδοση όρων	x
1. Εισαγωγή	1
2. Περιγραφή προβλήματος.....	4
2.1 Περιγραφή προβλήματος και αναγκαιότητα μελέτης	4
2.2 Σκοπός και στόχοι έρευνας.....	4
3. Βιβλιογραφική Ανασκόπηση.....	6
3.1 Σχετικές έρευνες και εφαρμογές.....	6
3.1.1 Αναγνώριση κόμβων με τη ανάλυση των ιχνών GPS	6
3.1.2 Αναγνώριση κόμβων με βάση εικόνες και διανύσματα	9
3.1.3 Αναγνώριση κόμβων μέσω πινακίδων κυκλοφορίας	10
3.1.4 Αναγνώριση κόμβων μέσω εικόνων από κινούμενο όχημα.....	12
3.1.5 Αναγνώριση κόμβων με την πρακτική CS - WAZE	13
3.2 Σύστημα Αυτοεκμάθησης.....	16
3.2.1 WEKA	17
3.3 Προγράμματα ανοικτού κώδικα (Open source software)	18
3.4 Crowdsourcing (πληθοπορισμός).....	18
3.5 Τεχνολογίες	20
3.5.1 Βάση Δεδομένων (ΒΔ).....	20
3.5.2 PHP (PHP: Hypertext Preprocessor).....	25
3.5.3 Java (γλώσσα προγραμματισμού)	25

3.5.4 Android SDK	26
4. Μεθοδολογία	27
4.1 Δεδομένα και κατασκευή ΒΔ.....	27
4.2 Ανάπτυξη εφαρμογής Android	29
4.2.1 Ανάλυση απαιτήσεων	29
4.2.2 Υλοποίηση εφαρμογής	30
4.2.2.1 Λειτουργίες εφαρμογής και κλάσεις	30
4.2.3 Σενάριο χρήσης εφαρμογής – διαδικασία λειτουργίας.....	34
4.2.3.1 Κανόνες στη διαδικασία καταγραφής	35
4.3 Λειτουργίες και αξιοποίηση του Server	37
4.3.1 Επεξεργασία δεδομένων από και προς την κινητή συσκευή.....	37
4.3.2 Ιστότοπος συστήματος	38
4.4 Διαδικασία εξαγωγής δεδομένων σε μορφή αρχείου WEKA (ARFF).....	40
4.4.1 Δομή αρχείου ARFF.....	41
5. Ανάλυση δεδομένων.....	43
5.1 Προεπεξεργασία δεδομένων.....	43
5.2 Ταξινόμηση δεδομένων (classification)	45
5.2.1 IBk (k - κοντινότεροι γείτονες)	47
5.2.2 Multilayer Perceptron (νευρωνικά δίκτυα).....	48
5.2.3 J48 (δέντρο απόφασης).....	49
5.3 Εξαγωγή κανόνων με τον αλγόριθμο J48.....	49
5.4 Απεικονίσεις δεδομένων.....	52
6. Εφαρμογή ΒΔ γράφων	54
7. Συμπεράσματα - Επίλογος.....	58
8. Βιβλιογραφία	60

Κατάλογος Πινάκων

Πίνακας 1: Βρίσκοντας φίλους σε εκτενές βάθος από δεδομένα μιας RDBMS σε σχέση με την ΒΔ Neo4j.....	24
Πίνακας 2: Αποτελέσματα εκπαίδευσης δεδομένων με τον αλγόριθμο IBk	48
Πίνακας 3: Αποτέλεσμα εκπαίδευσης αλγορίθμου Multilayer Perceptron	48
Πίνακας 4: Αποτέλεσμα εκπαίδευσης αλγορίθμου J48.....	49

Κατάλογος Εικόνων

Εικόνα 1: Συσκευή καταγραφής GPS RoyalTek RBT-2300.....	7
Εικόνα 2: Αποτυπώματα GPS.....	8
Εικόνα 3: Κόμβος (Road intersection).....	8
Εικόνα 4: Παραδείγματα διασταυρώσεων που συλλέχθηκαν από την έρευνα	9
Εικόνα 5: Αποτελέσματα της έρευνας.....	9
Εικόνα 6: Λανθασμένες ταυτοποιήσεις διασταυρώσεων. Αριστερά δεν αναγνωρίστηκε λόγω δέντρου, δεξιά λανθασμένα αναγνωρίστηκε μια γέφυρα	10
Εικόνα 7: Πρόβλημα πινακίδων λόγω φωτεινότητας.....	11
Εικόνα 8: Ταυτοποίηση πινακίδας με την μεγαλύτερη ποσότητα φωτός ανά μονάδα περιοχής (exposure)	11
Εικόνα 9: Περιοχές σήματος.....	12
Εικόνα 10: Αναγνώριση σημείων και κατεύθυνσης.....	12
Εικόνα 11: Αυτόματη τμηματοποίηση δρόμου	13
Εικόνα 12: Πιθανό σημείο διασταύρωσης - Διακοπή συνοχής της αριστερής λωρίδα του δρόμου	13
Εικόνα 13: Χρήση Waze - "Αθόρυβη" καταγραφή δεδομένων.....	15
Εικόνα 14: Λειτουργία αναφοράς συμβάντων.....	15
Εικόνα 15: Τροποποίηση σημείων διασταυρώσεων από το εργαλείο επεξεργασίας χάρτη στο σύστημα WAZE.....	16
Εικόνα 16: Παραδείγματα αριθμών γραμμένων στο χέρι.....	17
Εικόνα 17: Μοντέλο φίλων και των σχέσεων μεταξύ τους με γράφημα.....	22
Εικόνα 18: Facebook Graph Search.....	23
Εικόνα 19: Σχέσεις φίλων στο κοινωνικό γράφημα. Ο κεντρικός χρήστης απεικονίζεται στο κέντρο	23
Εικόνα 20: Αναπαράσταση συλλογής πληροφοριών.....	28
Εικόνα 21: ΒΔ συστήματος	29
Εικόνα 22: Οθόνη καλωσορίσματος εφαρμογής.....	32
Εικόνα 23: Οθόνη εισόδου χρήστη.....	33
Εικόνα 24: Οθόνη εγγραφής χρήστη	33

Εικόνα 25: Οθόνη καταγραφής δεδομένων	34
Εικόνα 26: Διαδικασία λειτουργίας εφαρμογής	36
Εικόνα 27: Διαδρομές που έχει καταγράψει ο χρήστης από τον ιστότοπο του συστήματος	39
Εικόνα 28: Υπόδειξη κόμβων στο σύστημα σε όλα τα δρομολόγια που έχουν καταγραφεί	40
Εικόνα 29: Ενδιάμεσα σημεία κόμβων (τα σημεία αρχής και τέλους κόμβου αναπαριστούν τα πράσινα σημεία).....	41
Εικόνα 30: Λειτουργία εξαγωγής συνόλου δεδομένων εκπαίδευσης.....	41
Εικόνα 31: Αρχείο ARFF από το σύνολο δεδομένων εκπαίδευσης απόσταση σημείων 3	42
Εικόνα 32: Διαφορά σημείων με απόσταση 2 που δεν καταχωρήθηκαν ως κόμβοι	44
Εικόνα 33: Διαφορά σημείων με απόσταση 3 που δεν καταχωρήθηκαν ως κόμβοι	44
Εικόνα 34: Διαφορά σημείων με απόσταση του σημείου αρχής κόμβου μέχρι το σημείο τέλους μόνο από σημεία κόμβων.....	44
Εικόνα 35: Εισαγωγή δεδομένων στο πρόγραμμα WEKA - Διαγραφή ιδιότητας ids	45
Εικόνα 36: Παράδειγμα αλγορίθμου ταξινόμησης.....	45
Εικόνα 37: Παράδειγμα k - κοντινότερων γειτόνων με $k = 5$. Το νέο στοιχείο x_q προβλέπεται αρνητικό.....	47
Εικόνα 38: Κανόνες που προέκυψαν από την ταξινόμηση με τον αλγόριθμο J48.....	50
Εικόνα 39: Ίχνη διαδρομών από την εφαρμογή σε απεικόνιση με Google Maps – API v3 (GPS traces).....	51
Εικόνα 40: Απεικόνιση αποτελεσμάτων - Άξονας Y: κλάση, Άξονας X: διαφορά απόστασης ..	52
Εικόνα 41: Απεικόνιση αποτελεσμάτων - Άξονας Y: κλάση, Άξονας X: διαφορά γωνίας κατεύθυνσης.....	53
Εικόνα 42: Διαδικασία δημιουργίας ΒΔ γράφων	54
Εικόνα 43: Παράδειγμα μέρος του συνόλου δεδομένων στην ΒΔ γράφων – Γράφημα ιδιοτήτων (property graph)	56

Συντομογραφίες

3G	Τρίτη γενιά τεχνολογίας τηλεπικοινωνιών
API	Application programming interface (Διασύνδεση Προγραμματισμού Εφαρμογών)
CS	Crowdsourcing (Πληθοπορισμός)
DBMS	Database Management System (Σύστημα διαχείρισης Βάσεων Δεδομένων)
GPS	Global Positioning System (Παγκόσμιο Σύστημα Θεσιθεσίας)
iOS	iPhone OS (Λειτουργικό σύστημα για τα κινητά τηλέφωνα iPhone)
MySQL	My - Structured Query Language (Δομημένη γλώσσα ερωτημάτων)
OSM	Open Street Maps
OSS	Open Source Software (Ανοικτού κώδικα λογισμικό)
RDBMS	Relational Database Management System (Σύστημα Διαχείρισης Σχεσιακών Βάσεων Δεδομένων)
SDK	Software Development Kit (Κιτ Ανάπτυξης Λογισμικού)
WEKA	Waikato Environment for Knowledge Analysis
Wi-Fi	Wireless Fidelity (Ασύρματη πιστότητα)
ΒΔ	Βάση Δεδομένων
ARFF	Attribute-Relation File Format (Μορφή αρχείου ιδιότητας – σχέσης)

Απόδοση όρων

Crowdsourcing	Πληθοπορισμός - μορφή συλλογικής διαδικτυακής δραστηριότητας
Server	Εξυπηρετητής ή διακομιστής
Spam	Ανεπιθύμητη ηλεκτρονική αλληλογραφία
Web Interface	Γραφικό Περιβάλλον Χρήστη ιστοσελίδας
Ιστότοπος	Σύνολο από ιστοσελίδες

1. Εισαγωγή

Η ραγδαία ανάπτυξη της τεχνολογίας τα τελευταία χρόνια, έχει οδηγήσει στη δημιουργία μιας πληθώρας καινοτόμων λύσεων οι οποίες βρίσκουν εφαρμογή σε διάφορους τομείς της καθημερινής μας ζωής σε προσωπικό, σε κοινωνικό και σε επαγγελματικό επίπεδο. Η τεχνολογία, όπως το αναφέρει και η ρίζα της λέξης, είναι ένα είδος τέχνης. Έτσι, όπως και η τέχνη, διακλαδώνεται σε διάφορους τομείς πάνω στους οποίους βασίζεται η καθημερινότητα των πλείστων ανθρώπων επιφέροντας ραγδαίες και ριζοσπαστικές αλλαγές στον τρόπο ζωής αφού έχει εμπλακεί στις αγορές, επικοινωνίες, συγκοινωνίες, ψυχαγωγία και σε πολλούς άλλους τομείς της καθημερινής μας ζωής.

Αυτή η τεράστια και ίσως απρόβλεπτη ανάπτυξη και διείσδυση της τεχνολογίας στην καθημερινότητα μας, οφείλεται κυρίως στην ανάπτυξη δύο επιμέρους τεχνολογιών. Οι τεχνολογίες διαδικτύου προσφέρουν πλέον φτηνές, γρήγορες, αξιόπιστες, ασφαλισμένες, και χωρικά εκτεταμένες συνδέσεις, επιτρέποντας σε κάθε χρήστη την ουσιαστικά μόνιμη σύνδεση του με το διαδίκτυο και την άντληση πληροφοριών από αυτό.

Η δεύτερη κατηγορία τεχνολογιών η οποία γνωρίζει ραγδαία ανάπτυξη τα τελευταία χρόνια δίνοντας τεράστια ώθηση στην καθημερινή χρήση της τεχνολογίας είναι αυτή των κινητών συσκευών (The Center of Technology, 2014). Οι κινητές συσκευές, έχουν πλέον τις ιδιότητες και επεξεργαστικές δυνατότητες ενός ηλεκτρονικού υπολογιστή και είναι σε θέση να αποστέλουν emails, να βγάζουν φωτογραφίες και να συνδέονται με το διαδίκτυο.

Ο συνδυασμός των πιο πάνω τεχνολογιών θέτει τα θεμέλια πάνω στα οποία διάφορες διαδικτυακές εφαρμογές μπορούν να αναπτυχθούν. Υπάρχουν διαφορετικά λειτουργικά συστήματα για κινητές συσκευές, μάλιστα κάποια από αυτά είναι ανοικτού κώδικα. Τα λειτουργικά που κυριαρχούν όμως αυτή τη στιγμή είναι το «Android» της εταιρείας Google, το «iOS» της Apple, τα «Windows mobile» από την εταιρία Microsoft και τα «BlackBerry» της εταιρίας BlackBerry. Επίσης, η γνωστή έκδοση Linux «Ubuntu» ξεκίνησε το 2013 μια έκδοση για κινητά τηλέφωνα (Covert, 2013).

Ανεξαρτήτως του λειτουργικού συστήματος το οποίο χρησιμοποιούν οι χρήστες των κινητών συσκευών έχουν τη δυνατότητα εγκατάστασης και χρήσης διαφόρων εφαρμογών. Κάποιες

αποσκοπούν στην ψυχαγωγία (πχ παιχνίδια) και κάποιες άλλες βοηθούν τον χρήστη σε διάφορες πτυχές όπως η αποθήκευση των αρχείων τους, η ανεύρεση κοντινών φαρμακείων, η συνομιλία μέσω κάμερας, παροχή ενημέρωσης κ.α. Οι πιο διαδεδομένες εφαρμογές και κυρίως εφαρμογές από μεγάλες εταιρείες, όπως για παράδειγμα η εφαρμογή «Gmail» από την Google, υπάρχει σχεδόν σε όλες τις τελευταίες εκδόσεις κινητών συσκευών. Ωστόσο, οι εταιρείες Google και Apple λόγω του τεράστιου μεριδίου αγοράς που κατέχουν, οι περισσότερες εφαρμογές είναι υλοποιημένες για τα λειτουργικά «Android» και «iOS».

Οι δύο εταιρίες που βρίσκονται πίσω από τα δύο πιο δημοφιλή λειτουργικά συστήματα κινητών συσκευών (Google για «Android» και Apple για «iOS») έχουν επιλέξει δυο αρκετά διαφορετικές προσεγγίσεις όσο αφορά τη δημιουργία και προώθηση επεκτάσεων των λειτουργικών τους συστημάτων. Η Google παρέχει τον πηγαίο κώδικα του λειτουργικού, ενώ η Apple όχι. Επίσης, για να διανέμει ένας προγραμματιστής την εφαρμογή του μέσω του «Apple store» πρέπει να πληρώσει και υποχρεούται αυτή η εφαρμογή να είναι συμβατή με ένα αριθμό προηγούμενων εκδόσεων του «iOS». Σε αντίθεση, η πολιτική της εταιρείας Google, επιτρέπει στον προγραμματιστή να δημοσιεύει εφαρμογές συμβατές με οποιαδήποτε έκδοση «Android» επιθυμεί. Επιπλέον, δεν απαιτείται οποιοδήποτε κόστος για την δημοσίευση και διάδοση της στο «Google Play».

Παρ' όλους τους διαφορετικούς περιορισμούς, οι προγραμματιστές των πλείστων εφαρμογών δημιουργούν εκδόσεις συμβατές με όλες τις διαφορετικές κινητές συσκευές, ή τουλάχιστον με αυτές που κατέχουν το μεγαλύτερο μερίδιο αγοράς. Είναι προφανές λοιπόν, ότι ανάμεσα σε όλες αυτές τις εταιρείες που ασχολούνται με την ανάπτυξη των κινητών συσκευών και των λειτουργικών τους συστημάτων, υπάρχει έντονος ανταγωνισμός.

Αποτέλεσμα του μεγάλου ανταγωνισμού είναι η αυξανόμενη παραγωγή νέων, καλύτερων, και καινοτόμων προϊόντων, γεγονός που επηρεάζει την αξία τους θετικά ως προς τον καταναλωτή. Οι κινητές συσκευές λοιπόν (συμπεριλαμβάνονται και τα «Tablet computers»), είναι πλέον προσιτές αλλά και πολύ ελκυστικές για τους καταναλωτές κυρίως λόγω χαμηλής τιμής σε σχέση με τις δυνατότητες που παρέχουν. Παρόλα αυτά, τον πρωταρχικό ρόλο για την καλύτερη αξιοποίησή και προσαρμογή των λειτουργιών των συσκευών αυτών, έχουν τα άτομα που ασχολούνται με την ανάπτυξη εφαρμογών για τις συσκευές αυτές.

Με την αξιοποίηση των νέων τεχνολογιών, οι οδηγοί έχουν αλλάξει τον τρόπο με τον οποίο αλληλοεπιδρούν με τα ίδια τους τα οχήματα και με το γενικότερο «περιβάλλον οδήγησης». Από την περίοδο που οι συσκευές πλοήγησης αλλά και οι εφαρμογές πλοήγησης πάνω σε κινητές συσκευές, έχουν εισχωρήσει στην καθημερινότητα των ανθρώπων, οι δρόμοι πλέον μπορούν να χαρακτηριστούν ως ένα μείγμα λογισμικού και υλικού.

Η παρούσα πτυχιακή εργασία επικεντρώνεται σε μεθόδους καταγραφής, επεξεργασίας και αξιοποίησης δεδομένων σχετικών με την οδική κυκλοφορία. Οι νέες τεχνολογίες θα αξιοποιηθούν για τη συλλογή, την ανάλυση, και αξιοποίηση των δεδομένων αυτών. Επιπρόσθετα, προτείνονται τεχνικές με τις οποίες τα δεδομένα αυτά θα αξιοποιηθούν για τη δημιουργία κανόνων που θα προκύψουν με τη χρήση τεχνικών μηχανικής μάθησης με απώτερο σκοπό την πλήρη αυτοματοποίηση διαδικασιών που αφορούν τη δημιουργία της απαραίτητης υποδομής πάνω στη οποία θα μπορούν να κτιστούν διάφορες εφαρμογές σχετικές με οδική κυκλοφορία.

Στην αναφορά που θα ακολουθήσει, αρχικά θα αναφερθεί το πρόβλημα, ο σκοπός και οι στόχοι που οδήγησαν στη διεκπεραίωση της παρούσας πτυχιακής εργασίας. Στην επόμενη ενότητα (Βιβλιογραφική ανασκόπηση), θα παρουσιαστούν υπάρχουσες έρευνες σχετικές με το θέμα της εργασίας αυτής. Έπειτα θα αναφερθούν διάφορες τεχνολογίες οι οποίες αξιοποιήθηκαν κατά τη διάρκεια της έρευνας. Στην ενότητα «Μεθοδολογία», θα γίνει αναφορά στην διαδικασία η οποία ακολουθήθηκε για τον σχεδιασμό και ανάπτυξη των διαφόρων τμημάτων που απαρτίζουν την εφαρμογή. Περαιτέρω, στην ενότητα «Ανάλυση δεδομένων», θα παρουσιαστεί η μέθοδος που ακολουθήθηκε για την ανάλυση των δεδομένων και πως προέκυψαν τα αποτελέσματα. Τέλος, στην ενότητα Εφαρμογή βάσης δεδομένων γράφων, θα αναφερθεί η αναγκαιότητα της εφαρμογής ενός διαφορετικού τύπου βάσης δεδομένων για την συγκεκριμένη υποδομή.

2. Περιγραφή προβλήματος

2.1 Περιγραφή προβλήματος και αναγκαιότητα μελέτης

Η αναγνώριση κόμβων (intersections) στις οδικές αρτηρίες έχει τεράστια σημασία σε συστήματα πλοήγησης, στις τοπογραφικές χαρτογραφήσεις και γενικότερα στις εφαρμογές σχετικές με την οδική κυκλοφορία.. Οι περισσότερες έρευνες γύρω από αυτό το θέμα είχαν ως σημείο αναφοράς διάφορες φωτογραφίες που απεικόνιζαν από κάποιο συγκεκριμένο υψόμετρο τις οδικές αρτηρίες. Στη συνέχεια, εφαρμόζοντας αλγορίθμους νευρωνικών δικτύων και επεξεργασίας εικόνων (image processing) η αναγνώριση κόμβων κατέστη δυνατή (Barsi & Heipke, 2003). Παρόλα αυτά, δεν έχει δημοσιευτεί ακόμη έρευνα η οποία αναγνωρίζει κόμβους χρησιμοποιώντας αποκλειστικά μια συλλογή δεδομένων που έχουν ληφθεί με το σύστημα GPS (Παγκόσμιο Σύστημα Θεσιθεσίας) και τα οποία να προκύπτουν από την κίνηση οχημάτων στους δρόμους.

Η διακίνηση με αυτοκίνητο ή οποιοδήποτε άλλο όχημα αποτελεί καθημερινότητα για τον κάθε πολίτη. Οι δρόμοι της Κύπρου, αλλά και οποιασδήποτε άλλης χώρας, συχνά παρουσιάζουν υπερβολική κυκλοφοριακή συμφόρηση στις πόλεις και συγκεκριμένα σε περιοχές και ώρες κατά τις οποίες οι πολίτες μεταβαίνουν στον εργασιακό τους χώρο. Μια καινοτόμα εφαρμογή η οποία με αυτοματοποιημένες ενέργειες θα προσφέρει πληροφορίες σχετικά με διαφορετικές πτυχές της οδικής κυκλοφορίας αποτελεί μεγάλη αναγκαιότητα για τον κάθε πολίτη του οποίου το κύριο μέσω μεταφοράς είναι το αυτοκίνητο.

2.2 Σκοπός και στόχοι έρευνας

Η παρούσα έρευνα εστιάζεται στη δημιουργία της απαραίτητης υποδομής πάνω στην οποία θα μπορούν να αναπτυχθούν διάφορες εφαρμογές σχετικές με τη οδική κυκλοφορία και τη διακίνηση πολιτών. Κεντρικό σημείο στην όλη προσπάθεια είναι η καταγραφή διαδρομών με τη χρήση Crowdsourcing (πληθοπορισμός) και η χρήση των δεδομένων αυτών για την

αυτόματη αναγνώριση οδικών διασταυρώσεων κάτι που θα επιτρέψει την χαρτογράφηση και τη δημιουργία στατιστικών στοιχείων για τις διάφορες οδικές αρτηρίες της Κύπρου.

Πιο συγκεκριμένα, για την υλοποίηση και πιλοτική εφαρμογή των πιο πάνω θα υλοποιηθεί μια εφαρμογή συμβατή με κινητές συσκευές λειτουργικού «Android». Η εφαρμογή αυτή θα καταγράφει δεδομένα σχετικά με τη συμπεριφορά των αυτοκινήτων - οδηγών (ταχύτητα, κατεύθυνση, τοποθεσία, επιτάχυνση) κατά την διάρκεια κίνησής τους με τη χρήση GPS (Global Positioning System), Wi-Fi, αισθητήρων επιτάχυνσης και πυξίδας. Επιπλέον οι πληροφορίες αυτές θα είναι ομαδοποιημένες ανά ημέρα και ώρα καταγραφής. Παράλληλα θα δημιουργηθεί μια διαδικτυακή εφαρμογή με την οποία ο χρήστης με τη χρήση του φυλλομετρητή (browser) του θα έχει τη δυνατότητα να αναλύσει και απεικονίσει γραφικά τα δεδομένα που έχουν συλλεχθεί και αναλυθεί.

Ένας επιπλέον σκοπός της έρευνας είναι να θέσει τις βάσεις για μια μετέπειτα υλοποίηση εφαρμογών που θα διευκολύνουν τη διακίνηση ατόμων με οχήματα, υποδεικνύοντάς τους βέλτιστες διαδρομές με βάση τα δεδομένα που έχουν καταχωρηθεί από όλους τους χρήστες του συστήματος.

Συνοψίζοντας, ο κύριος σκοπός της έρευνας εστιάστηκε στην αναγνώριση διασταυρώσεων χρησιμοποιώντας τεχνικές μηχανικής μάθησης με την ανάλυση δεδομένων από διάφορες διαδρομές. Ο τρόπος συλλογής δεδομένων πραγματοποιήθηκε με τη δημιουργία μιας αφαιρετικής εφαρμογής σε κινητές συσκευές η οποία λειτούργησε με την πρακτική του πληθοπορισμού (Crowdsourcing) και η οποία απαιτούσε ελάχιστες ενέργειες από την πλευρά του χρήστη αλλά και καμία εξειδικευμένη γνώση.

Τέλος, ένας στόχος ο οποίος είναι απόρροια της εργασίας αυτής, είναι η χρήση γραφημάτων (graphs) για την περαιτέρω ανάλυση και παροχή επιπρόσθετων πληροφοριών στους χρήστες (π.χ., υπόδειξη εναλλακτικών διαδρομών). Για το σκοπό αυτό θα αξιοποιηθούν προηγμένες τεχνολογίες βάσεων δεδομένων, οι βάσεις γράφων (Graph Databases). Εν τέλει, σε ατομικό επίπεδο, στόχος θεωρείται και η αποκόμιση γνώσεων και εμπειριών μέσα από μια ερευνητική και πρακτική εντρύφηση με διάφορες τεχνολογίες αιχμής.

3. Βιβλιογραφική Ανασκόπηση

3.1 Σχετικές έρευνες και εφαρμογές

Σχετικά με την αναγνώριση των σημείων διασταυρώσεων ασχολήθηκε ένας σημαντικός αριθμός ερευνών. Όπως αναφέρθηκε στην ενότητα Αναγκαιότητα Μελέτης, η αναγνώριση κόμβων στις οδικές αρτηρίες αποτελεί σημαντική παράμετρο στα συστήματα πλοήγησης. Ο τρόπος που γίνεται η αναγνώριση αυτών των σημείων από εταιρίες ανάπτυξης συστημάτων πλοήγησης, δεν είναι δημοσιευμένος, καθώς αυτό αποτελεί εταιρικό μυστικό και δεν είναι κατορθωτή η ανεύρεση τους. Έτσι, σε αυτή την αναφορά περιέχονται έρευνες που έχουν δημοσιευτεί από ερευνητικούς οργανισμούς. Οι έρευνες έχουν αποδείξει ότι οι κόμβοι μπορούν να αναγνωριστούν με κάποιο αυτοματοποιημένο τρόπο. Ωστόσο, καμία προσέγγιση δε χρησιμοποιεί αποκλειστικά δεδομένα GPS χωρίς την υποστήριξη άλλων τεχνολογιών πληροφορικής και κυρίως επεξεργασία εικόνων.

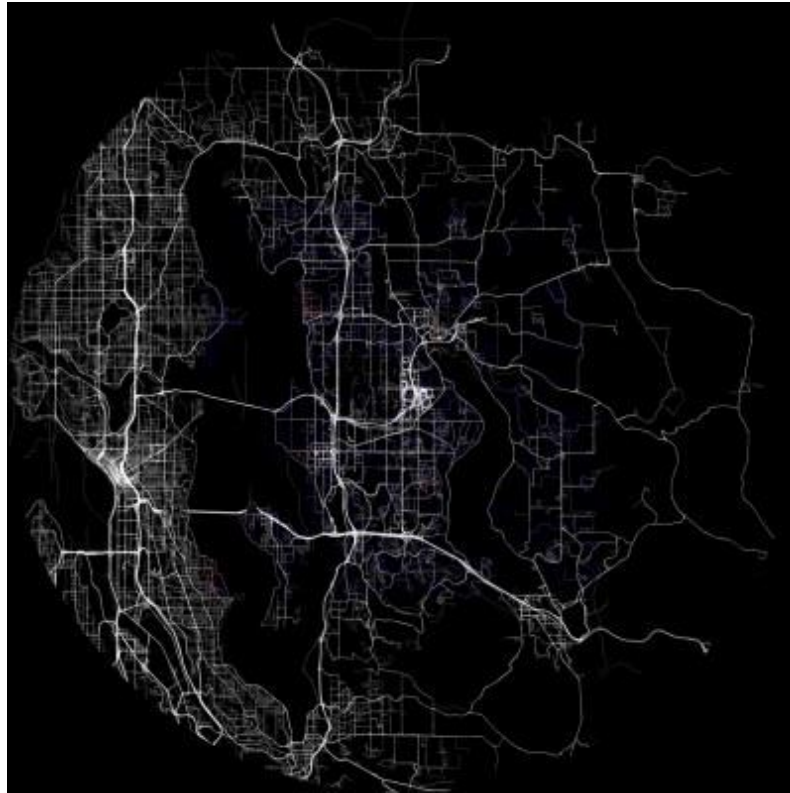
3.1.1 Αναγνώριση κόμβων με τη ανάλυση των ιχνών GPS

Σύμφωνα με την έρευνα «Detecting Road Intersections from GPS Traces» (Fathi & Krumm, 2010), υπάρχει η δυνατότητα ταυτοποίησης των κομβικών σημείων χωρίς την βοήθεια τοπογράφησης. Η εναλλακτική λύση που πρότειναν, είναι η συλλογή ενός μεγάλου συνόλου από ίχνη GPS σε οχήματα που έκαναν συχνές διαδρομές. Για την καταγραφή δεδομένων χρησιμοποιήθηκαν οι συσκευές «RoyalTek RBT-2300» (βλέπε Εικόνα 1), οι οποίες είχαν μια τυπική απόκλιση 4.1 μέτρα (σύμφωνα με την έρευνα).



Εικόνα 1: Συσκευή καταγραφής GPS RoyalTek RBT-2300

Οι καταγραφές δεδομένων έγιναν από 55 αυτοκίνητα και 252 λεωφορεία. Τα αυτοκίνητα είχαν χρησιμοποιηθεί για περισσότερες από δύο βδομάδες καταγράφοντας περίπου 99,5 ώρες το κάθε ένα. Τα λεωφορεία είχαν συλλέξει δεδομένα για περισσότερες από τέσσερις βδομάδες καταγράφοντας περίπου 444 ώρες το κάθε ένα. Το αποτέλεσμα ήταν μια αποτύπωση των οδικών αρτηριών (βλέπε Εικόνα 2). Με βάση την απεικόνιση των δεδομένων, ανέπτυξαν ένα αλγόριθμο ο οποίος είναι σε θέση να αναγνωρίσει κόμβους βάσει των σημείων που διασταυρώνονται (βλέπε Εικόνα 3). Τα δεδομένα έχουν τροποποιηθεί έτσι ώστε ο αλγόριθμος στο πρώτο στάδιο να εκπαιδευτεί με κάποιο συγκεκριμένο τρόπο, έτσι ώστε στο δεύτερο στάδιο να καταστεί δυνατό η ταυτοποίηση των διασταυρώσεων μέσω αρνητικών και θετικών εικόνων.



Εικόνα 2: Αποτυπώματα GPS

(Fathi & Krumm, 2010)



Εικόνα 3: Κόμβος (Road intersection)

(Fathi & Krumm, 2010)

3.1.2 Αναγνώριση κόμβων με βάση εικόνες και διανύσματα

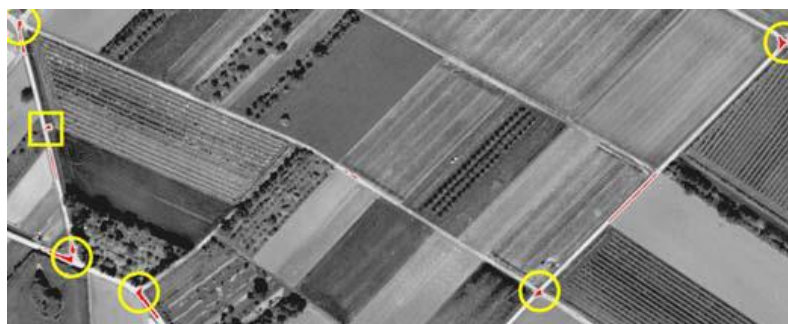
Ακόμα μια σχετική έρευνα, πραγματοποιήθηκε από τους Arpad Barsi και Christian Heipke, στην οποία έγινε μια προσπάθεια αναγνώρισης διασταυρώσεων χρησιμοποιώντας αλγορίθμους νευρωνικών δικτύων πάνω σε ασπρόμαυρες αεροφωτογραφίες (Barsi & Heipke, 2003). Κατά το πρώτο στάδιο της έρευνας, συλλέχθηκαν κομμάτια εικόνων με κόμβους μέχρι τεσσάρων δρόμων (βλέπε Εικόνα 4). Τα κομμάτια είχαν περιστραφεί σε ποικίλες κλίσεις έτσι ώστε να μειωθεί ο παράγοντας λάθους εξαιτίας της λανθασμένης κλίσης της εικόνας και να αυξηθούν τα διαφορετικά δείγματα.



Εικόνα 4: Παραδείγματα διασταυρώσεων που συλλέχθηκαν από την έρευνα

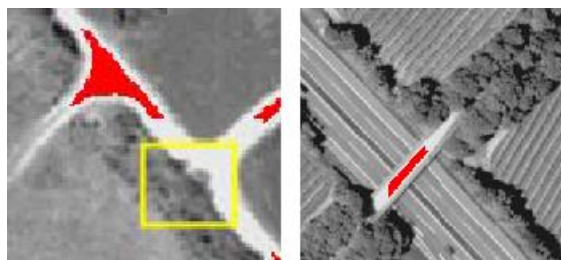
(Barsi & Heipke, 2003)

Η φάση εκπαίδευσης έγινε με αλγορίθμους νευρωνικών δικτύων λόγω της καλής ακρίβειας που έχουν. Η διαδικασία αναγνώρισης είχε θετικά αποτελέσματα (βλέπε Εικόνα 5), όπου οι περισσότεροι κόμβοι αναγνωρίστηκαν. Παρόλα αυτά, υπήρχαν περιπτώσεις λάθους όπου δεν αναγνωρίζονταν λόγω δέντρων ή αναγνωρίζονταν γέφυρες ως κόμβοι (βλέπε Εικόνα 6).



Εικόνα 5: Αποτελέσματα της έρευνας

(Barsi & Heipke, 2003)



Εικόνα 6: Λανθασμένες ταυτοποιήσεις διασταυρώσεων. Αριστερά δεν αναγνωρίστηκε λόγω δέντρου, δεξιά λανθασμένα αναγνωρίστηκε μια γέφυρα

(Barsi & Heipke, 2003)

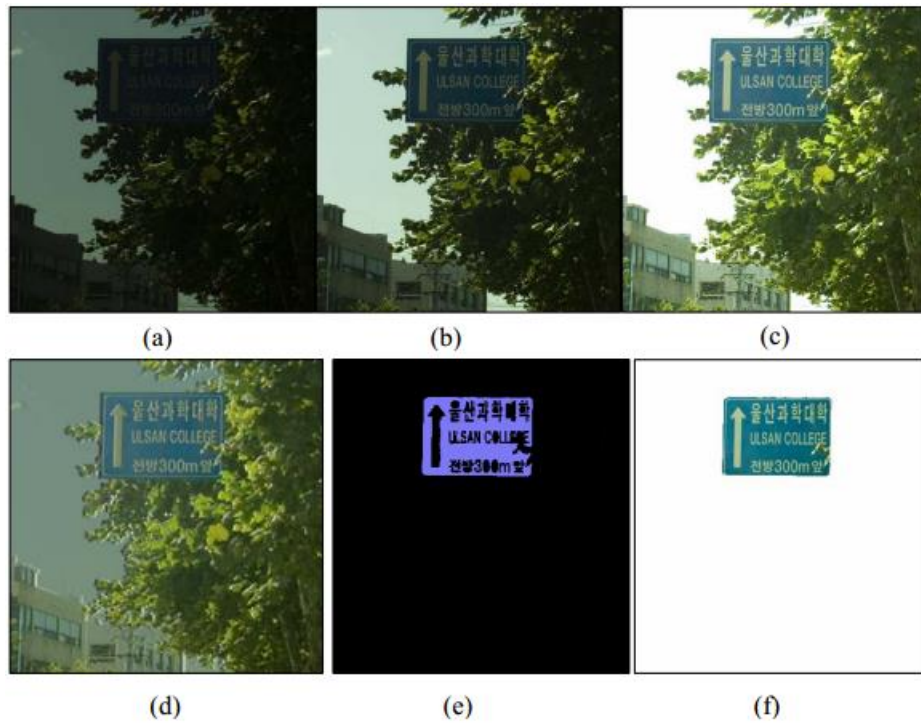
3.1.3 Αναγνώριση κόμβων μέσω πινακίδων κυκλοφορίας

Ακόμα μια προσέγγιση στην ταυτοποίηση διασταυρώσεων προτάθηκε από τους Andrey Vavilin και Kang-Hyun Jo (2009) αναλύοντας φωτογραφίες από πινακίδες. Οι φωτογραφίες λόγω του διαφορετικού τόνου φωτεινότητας, καθιστούσε δύσκολη την ανάλυσή τους (βλέπε Εικόνα 7). Έτσι, πριν το στάδιο της ανάλυσης είχαν εφαρμοστεί κάποιοι αλγόριθμοι οι οποίοι επέλεγαν το κομμάτι της φωτογραφίας με την μεγαλύτερη ποσότητα φωτός ανά μονάδα περιοχής (exposure) (βλέπε Εικόνα 8). Η έρευνα πραγματοποιήθηκε στην Κορέα όπου οι πινακίδες έχουν ορθογώνιο σχήμα και χρώμα μπλε, πράσινο ή καφέ. Ακολούθως, κάθε σήμα, τμηματοποιήθηκε σε τρεις περιοχές που είναι (α) η περιοχή της υπόδειξης κατεύθυνσης, (β) η περιοχή κειμένου και (γ) η περιοχή απόστασης (βλέπε Εικόνα 9). Αφού εφαρμόστηκαν τα φίλτρα και απομονώθηκε το σήμα από την πινακίδα, το επόμενο στάδιο ήταν η αναγνώριση των διάφορων συμβόλων του σήματος (βλέπε Εικόνα 10). Η συγκεκριμένη μέθοδος αποδείχθηκε αποτελεσματική αφού είχε 96% επιτυχία στην αναγνώριση κατεύθυνσης και 93% στην ταυτοποίηση πορείας (Vavilin & Jo, 2009).



Εικόνα 7: Πρόβλημα πινακίδων λόγω φωτεινότητας

(Vavilin & Jo, 2009)



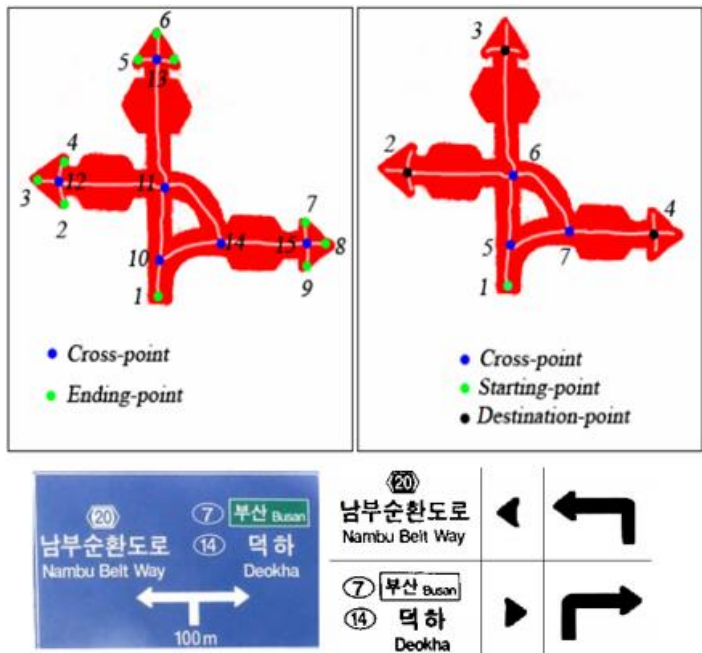
Εικόνα 8: Ταυτοποίηση πινακίδας με την μεγαλύτερη ποσότητα φωτός ανά μονάδα περιοχής (exposure)

(Vavilin & Jo, 2009)



Εικόνα 9: Περιοχές σήματος

(Vavilin & Jo, 2009)



Εικόνα 10: Αναγνώριση σημείων και κατεύθυνσης

(Vavilin & Jo, 2009)

3.1.4 Αναγνώριση κόμβων μέσω εικόνων από κινούμενο όχημα

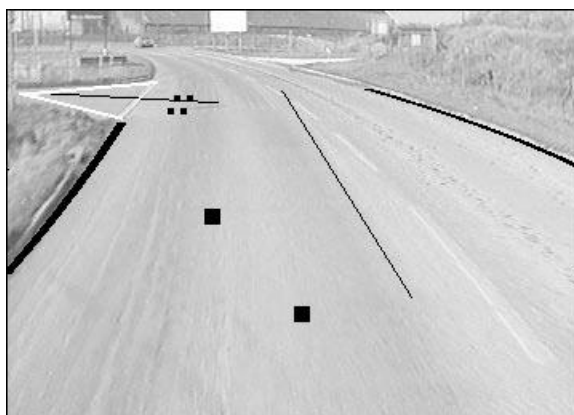
Η αυτόματη αναγνώριση κόμβων στους δρόμους έγινε επίσης εφικτή σε μέρος μιας προσπάθειας για δημιουργία αυτόνομου μηχανικού οχήματος (Gibbs & Thomas, 2000). Αναλύοντας εικόνες που βρίσκονται μπροστά από την πορεία του οχήματος, αναπτύχθηκαν αλγόριθμοι οι οποίοι είχαν την δυνατότητα να διακρίνουν τα άκρα του δρόμου και τα επιμέρους τμήματα του (βλέπε Εικόνα 11).



Εικόνα 11: Αυτόματη τμηματοποίηση δρόμου

(Gibbs & Thomas, 2000)

Αφού το πρώτο στάδιο επιτεύχθηκε, το επόμενο βήμα ήταν η ανάπτυξη άλλων αλγορίθμων οι οποίοι είχαν την δυνατότητα να αναγνωρίσουν αν κάποια από τις εξωτερικές λωρίδες του δρόμου παρουσίαζαν κάποιου συγκεκριμένου πλάτους κενό (βλέπε Εικόνα 12).



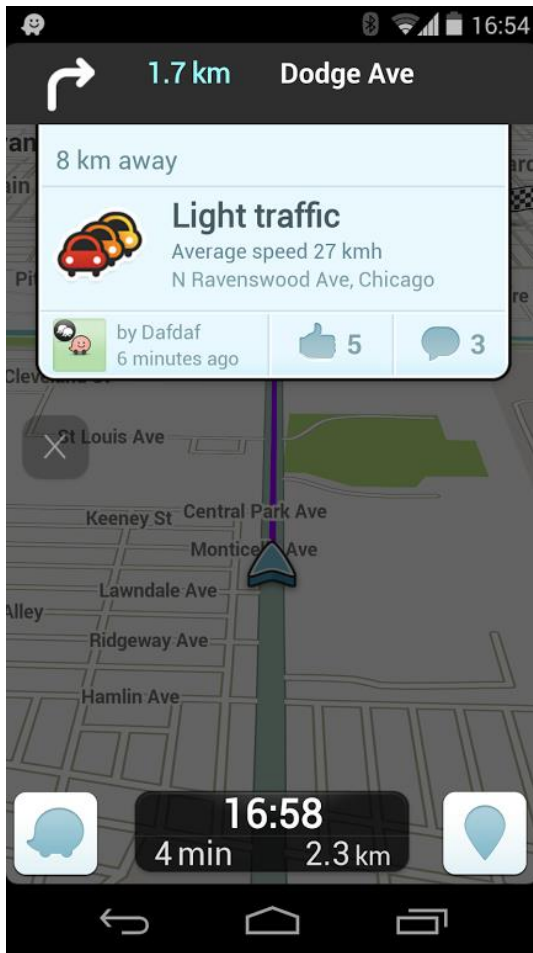
Εικόνα 12: Πιθανό σημείο διασταύρωσης - Διακοπή συνοχής της αριστερής λωρίδα του δρόμου

(Gibbs & Thomas, 2000)

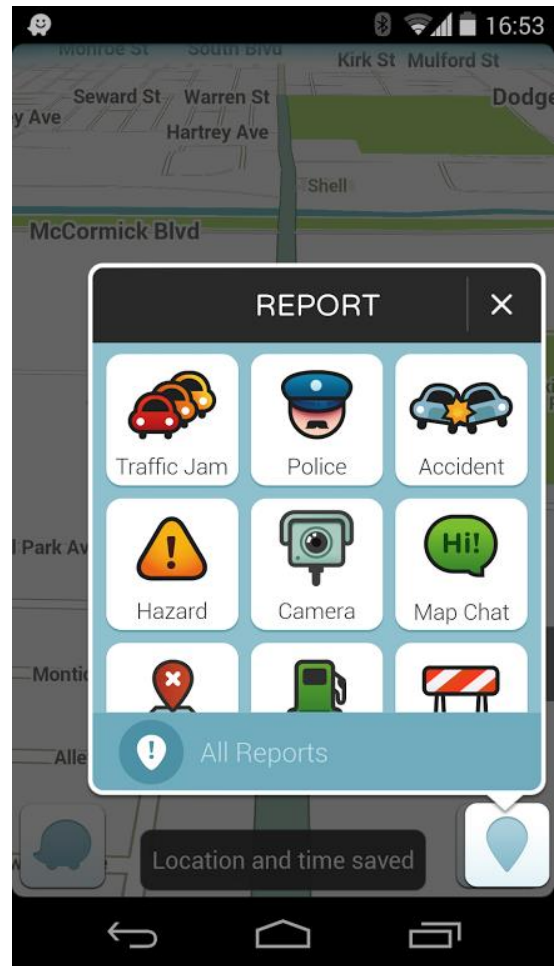
3.1.5 Αναγνώριση κόμβων με την πρακτική CS - WAZE

Το Waze» είναι μια εφαρμογή πλοήγησης η οποία διαδόθηκε στους χρήστες των κινητών συσκευών με λειτουργικό Windows Mobile, iOS και Android. Υλοποιήθηκε από την εταιρία Waze Ltd το 2008 αλλά τον Ιούνιο του 2013 αγοράστηκε από την Google μαζί με πενήντα

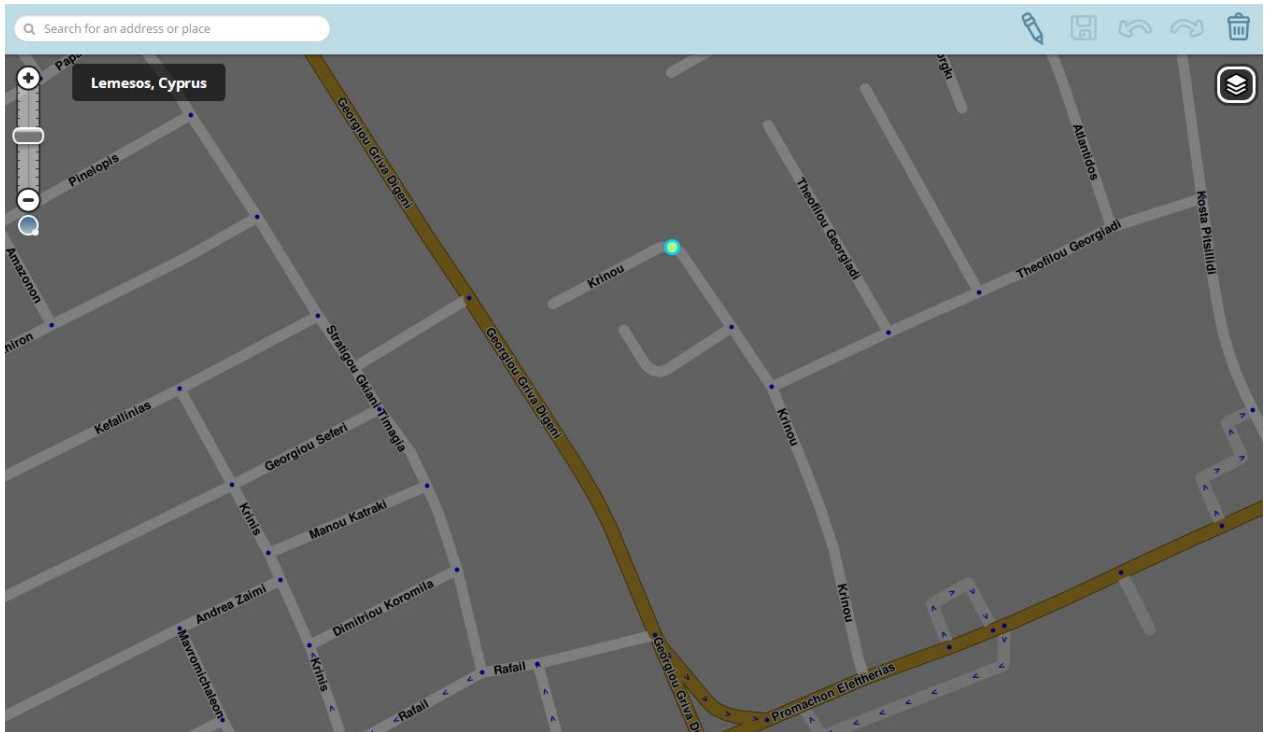
εκατομμύρια περίπου χρήστες (Josef & Max, 2013). Αυτό που έκανε την εφαρμογή να διαφέρει από τα υπόλοιπα συστήματα πλοήγησης και να ελκύσει το ενδιαφέρον της εταιρίας Google, είναι οι λειτουργίες για ευέλικτη τροποποίηση των δεδομένων στους χάρτες, βασισμένες στην πρακτική του Crowdsourcing (Sam & Gekker, 2014). Σύμφωνα με τους Sam Hind και Alex Gekker (2014), οι χρήστες της εφαρμογής τροποποιούν διάφορες πληροφορίες που αφορούν τις διαδρομές στον χάρτη όπως όρια ταχύτητας, σημεία διασταυρώσεων, δρόμοι που καταλήγουν σε αδιέξοδο κ.α. Όσον αφορά τις ενέργειες που πρέπει να ακολουθήσει ο χρήστης για να συμβάλει στην τροποποίηση των δεδομένων υπάρχουν δύο τρόποι. Ο πρώτος τρόπος χαρακτηρίζεται ως παθητική λειτουργία συλλογής δεδομένων. Αυτό πραγματοποιείται όταν οι χρήστες είναι ενεργοί οδηγοί στην φάση χρήσης της εφαρμογής. Η λειτουργία αυτή, κυρίως είναι χρήσιμη για επιβεβαίωση σωστών διαδρομών αλλά και για συλλογή στοιχείων που θα βοηθήσουν στην λειτουργία για προτεινόμενες διαδρομές με βάση την κυκλοφοριακή κίνηση (βλέπε Εικόνα 13). Ο δεύτερος τρόπος διαμόρφωσης των πληροφοριών του συστήματος, πραγματοποιείται μέσω κάποιων εργαλείων που έχουν αναπτυχθεί με σκοπό την ώθηση του χρήστη σε ένα πιο ενεργό ρόλο συμμετοχής. Παραδείγματα των εργαλείων αυτών είναι η επεξεργασία χάρτη μέσω του φυλλομετρητή και η λειτουργία αναφοράς τροχαίας κίνησης ή άλλων συμβάντων (τροχαίο ατύχημα, τιμές βενζίνης, πρόβλημα χάρτη κ.α.) στην εφαρμογή (βλέπε Εικόνα 14). Παρόλα αυτά, το σύστημα πίσω από την εφαρμογή WAZE ταυτοποιεί τα σημεία οδικών κόμβων κυρίως με την βοήθεια των χρηστών μέσω των εργαλείων τροποποίησης δεδομένων (βλέπε Εικόνα 15) και στρατηγικής Crowdsourcing.



Εικόνα 13: Χρήση Waze - "Αθόρυβη" καταγραφή δεδομένων.



Εικόνα 14: Λειτουργία αναφοράς συμβάντων



Εικόνα 15: Τροποποίηση σημείων διασταυρώσεων από το εργαλείο επεξεργασίας χάρτη στο σύστημα WAZE

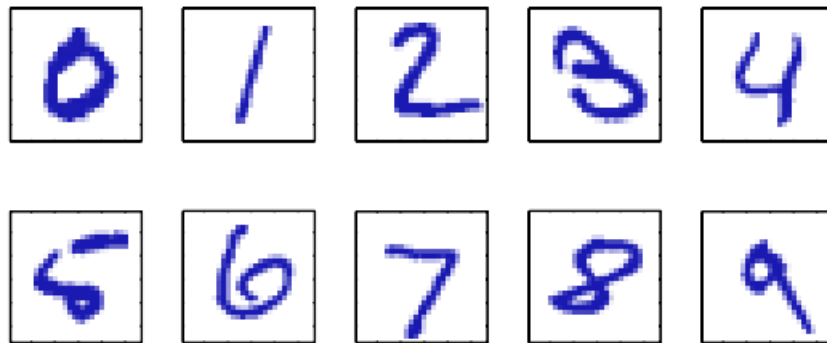
3.2 Σύστημα Αυτοεκμάθησης

Ένα σύστημα αυτοεκμάθησης αναφέρεται στην μηχανική μάθηση. Η μηχανική μάθηση είναι μια περιοχή της τεχνητής νοημοσύνης η οποία με διάφορες μεθόδους επιτρέπει στους υπολογιστές να «εκπαιδεύονται». Με την βοήθεια αλγορίθμων και μεθόδων καθίσταται εφικτή η κατασκευή προσαρμόσιμων (adaptable) προγραμμάτων υπολογιστών, τα οποία λειτουργούν με βάση την αυτοματοποιημένη ανάλυση συνόλων δεδομένων και όχι τη διαίσθηση των μηχανικών που τα προγραμμάτισαν. Κάποιες εφαρμογές της μηχανικής μάθησης είναι η αυτόματη κατηγοριοποίηση λέξεων κειμένου, η εύρεση των συντομότερων διαδρομών, η κατηγοριοποίηση «spam» ηλεκτρονικών μηνυμάτων (email) κ.α.

Για να γίνει πιο κατανοητή η διαδικασία εκμάθησης θεωρήστε το παράδειγμα εφαρμογής ευριστικού αλγορίθμου για την αναγνώριση ψηφίων γραμμένων στο χέρι (βλέπε Εικόνα 16). Η διαδικασία αναγνώρισης βασίζεται σε μια μεγάλη συλλογή αριθμών $\{X_1 \dots X_n\}$ που κάθε αριθμός

αντιστοιχεί σε μια εικόνα ενός χειρόγραφου αριθμού. Η συλλογή αυτή ονομάζεται σύνολο εκπαίδευσης (training set). Κατά την διαδικασία μάθησης, ο αλγόριθμος μπορεί να εκφραστεί ως $f(x)$ που λαμβάνει μια παράμετρο x , όπου x ισούται με μια ψηφιακή εικόνα αριθμού. Το αποτέλεσμα θα είναι ένας συνδυασμός από κανόνες με τους οποίους θα αναγνωρίζονται άλλοι αριθμοί.

Σχετικά με την ανάλυση διαδρομών, αναλύοντας το σύνολο δεδομένων από τις διαδρομές μέσω κάποιων μεθόδων μηχανικής μάθησης, είναι δυνατό να παραχθούν ορισμένοι κανόνες οι οποίοι επιτρέπουν την κατηγοριοποίηση κάποιου σημείου σε κόμβο ή μη. Για το σκοπό αυτό (για πρόβλεψη και ομαδοποίηση δεδομένων (clustering)) θα χρησιμοποιηθεί το εργαλείο WEKA.



Εικόνα 16: Παραδείγματα αριθμών γραμμένων στο χέρι

(Bishop, 2006)

3.2.1 WEKA

Το λογισμικό WEKA δημιουργήθηκε με σκοπό να βοηθήσει στις εφαρμογές μηχανικής μάθησης σε διάφορα προβλήματα του πραγματικού κόσμου. Υλοποιήθηκε από το πανεπιστήμιο Waikato το 1993 με τη χρήση ενός μείγματος διαφόρων γλωσσών προγραμματισμού όπως είναι η C και η TLC/TK. Το 1997 αποφασίστηκε το πρόγραμμα να υλοποιηθεί ξανά από την αρχή στην γλώσσα Java. Η πρώτη έκδοση του (3.0) πραγματοποιήθηκε το 1999 (Machine Learning Group at the University of Waikato., 2014). Στόχος των ατόμων που συνέβαλαν και συνεχίζουν να συμβάλουν στο έργο, είναι η ανάπτυξη ενός περιβάλλοντος με μια πληθώρα από διαδραστικά εργαλεία και αλγορίθμους που βοηθούν στην μηχανική μάθηση και την εξερεύνηση

νέων κανόνων αυτόματης αναγνώρισης και ταυτοποίησης δεδομένων (Holmes, Donkin, & Witten, 1994). Η πιο πρόσφατη έκδοση του είναι η 3.7 και είναι συμβατή με τα λειτουργικά Windows, Linux και Mac Os. Συνοψίζοντας, το εργαλείο WEKA προσφέρει μια εξελιγμένη διαδικασία πρόβλεψης (classify), ομαδοποίησης (clustering) και γραφικών απεικονίσεων χρησιμοποιώντας περίπλοκους αλγόριθμους νευρωνικών δικτύων, δέντρων απόφασης κ.α.

3.3 Προγράμματα ανοικτού κώδικα (Open source software)

Το λογισμικό ανοιχτού κώδικα (OSS) είναι το λογισμικό του οποίου ο πηγαίος κώδικάς του διατίθεται με άδεια στην οποία ο κάτοχος των πνευματικών δικαιωμάτων παρέχει τα δικαιώματα για τη μελέτη, την αλλαγή και τη διανομή του λογισμικού σε οποιοδήποτε άτομο και για οποιοδήποτε σκοπό. Το λογισμικό ανοικτού κώδικα δεν σημαίνει απαραίτητως ότι είναι δωρεάν, ούτε ελεύθερο λογισμικό σύμφωνα με τον ευρύ ορισμό που δίνει στο ελεύθερο λογισμικό ο οργανισμός Ελεύθερου Λογισμικού (Free Software Foundation, 2014). Αναφέρεται μόνο στο γεγονός ότι επιτρέπεται σε κάθε χρήστη να εξετάσει και να χρησιμοποιήσει τη γνώση και τις δυνατότητες που προσφέρει ο παρεχόμενος πηγαίος κώδικας. Παρόλα αυτά, τα περισσότερα προγράμματα ανοικτού κώδικα παρέχονται δωρεάν. Στην διαδικασία υλοποίησης της παρούσας πτυχιακής εργασίας χρησιμοποιήθηκε όσο δυνατόν περισσότερο υλικό από ανοικτού κώδικα λογισμικά.

3.4 Crowdsourcing (πληθοπορισμός)

Ο όρος «Crowdsourcing» (CS) αναφέρεται στην συλλογική δραστηριότητα η οποία έχει κάποιο σκοπό και εφαρμόζεται από ατομικό επίπεδο μέχρι σε οργανισμούς και επιχειρήσεις. Τον όρο έχει αναφέρει πρώτος ο Jeff Howe σε ένα άρθρο του στο περιοδικό «Wired» το 2006 (Esposti, 2010). Τα συστήματα CS χρησιμοποιούν ένα σύνολο από άτομα για να βοηθήσουν ή να συνεισφέρουν στην επίλυση ενός μεγάλου φάσματος προβλημάτων. Όσον αφορά τις εφαρμογές του όρου CS στο διαδίκτυο, χρησιμοποιούνται και άλλοι όροι όπως είναι το «user-powered systems», «user-generated content», «collaborative systems», «community systems»,

«collective intelligence» κ.α. Το σύνολο των ατόμων που θα προσφερθούν σε κάποιο κάλεσμα ποτέ δεν είναι προκαθορισμένος.

Ένα παράδειγμα CS είναι η διαδικτυακή εγκυκλοπαίδεια Wikipedia του μη κερδοσκοπικού οργανισμού «Wikimedia Foundation», η οποία συμπληρώνεται καθημερινά από χρήστες που γράφουν λήμματα αφίλοκερδώς. Ακόμη ένα παράδειγμα, είναι οι χάρτες Open Street Maps (OSM), οι οποίοι είναι αποτέλεσμα συλλογικής προσπάθειας μιας μεγάλης ομάδας χρηστών παγκοσμίως, με σκοπό την δημιουργία ενός λεπτομερή, αποστασιοποιημένου και ελεύθερου στο κοινό χάρτη. Παρόλα αυτά, ένα γεγονός CS δεν χρειάζεται κατ' ανάγκη να ξεκινήσει από κάποιο άτομο ή οργανισμό. Για παράδειγμα, το πρόσφατο ατυχές συμβάν της χαμένης πτήσης MH370 των Μαλαισιανών αερογραμμών που έγινε στις 8 Μαρτίου 2014, πάνω από 25,000 άτομα προσφέρθηκαν να βοηθήσουν στις έρευνες, ψάχνοντας για στοιχεία του χαμένου αεροσκάφους σε υψηλής ανάλυσης εικόνων από την DigitalGlobe (Woollaston, 2014).

Τα συστήματα CS αντιμετωπίζουν τέσσερις προκλήσεις. Η πρώτη και κυριότερη είναι το πώς θα προσελκύσουν άτομα για να συνεισφέρουν. Η δεύτερη είναι τι μπορούν να κάνουν αυτά τα άτομα και τρίτη πώς να συνδέσουν όλες τις επιμέρους συνεισφορές. Η τελευταία πρόκληση αφορά το πώς θα αντιμετωπιστούν οι καταχρήσεις που τυχόν να υπάρξουν στο σύστημα (Doan, Ramakrishnan, & Halevy, 2011). Στην περίπτωση των OSM, τα άτομα που συντονίζουν το έργο, επικαλούνται την ελεύθερη χρήση των χαρτών σε σχέση με τους χάρτες της εταιρίας Google, με σκοπό να κινητοποιήσουν εθελοντές να συμβάλουν. Όσον αφορά την δεύτερη πρόκληση, κατατοπίζουν τα άτομα με οδηγίες στο πως θα μπορέσουν να συνεισφέρουν στο έργο και με ποια εργαλεία. Στην τρίτη πρόκληση, το OSM έχει τροποποιήσει το σύστημα του έτσι ώστε οι πληροφορίες από τους διάφορες εθελοντές να καταχωρούνται αυτόματα. Τέλος, ένα θέμα στην περίπτωση κατάχρησης το έχουν επίσης αυτοματοποιήσει, βάζοντας όριο στην χρήση που δέχονται οι server (για τον προφανή λόγο ότι δεν έχουν απεριόριστους πόρους).

Στην παρούσα έρευνα, η συνεισφορά του κάθε χρήστη περιορίζεται αποκλειστικά στη συλλογή δεδομένων μέσω της εφαρμογής Android. Όλες οι υπόλοιπες διαδικασίες θα είναι αυτοματοποιημένες και διαφανείς προς τον χρήστη. Αυτή η προσέγγιση αναμένεται να επιλύσει τα προβλήματα που περιγράφηκαν πιο πάνω σχετικά με τη χρήση τεχνικών CS.

3.5 Τεχνολογίες

3.5.1 Βάση Δεδομένων (ΒΔ)

Με τον όρο δεδομένα εννοούμε τη μη επεξεργάσιμη μορφή οποιουδήποτε ψηφιακού στοιχείου. Τα δεδομένα γίνονται «πληροφορία» όταν αυτά επεξεργαστούν και κατηγοριοποιηθούν. Ένα σύστημα διαχείρισης βάσεων δεδομένων (DBMS) είναι μια συλλογή από προγράμματα που διαχειρίζονται τη δομή της βάσης δεδομένων και ελέγχουν την πρόσβαση στα δεδομένα που αποθηκεύονται μέσα σε αυτή. Χρησιμοποιώντας ένα σύστημα Βάσης Δεδομένων (ΒΔ) προσφέρει αρκετά πλεονεκτήματα. Το κυριότερο πλεονέκτημα είναι ότι τα δεδομένα μέσα σε μια ΒΔ μπορούν να χρησιμοποιηθούν από πολλές εφαρμογές και χρήστες ταυτόχρονα με σκοπό την εύκολη και γρήγορη πρόσβαση σε αυτά και τη μετατροπή τους (από τις εφαρμογές) σε πληροφορία. Επίσης μια ΒΔ μπορεί να βρίσκεται σε διαφορετική τοποθεσία με την εφαρμογή. Η ανάκτηση των δεδομένων είναι δυνατή μέσω αναζήτησης κατ' απαίτηση (ερωτήματα/queries) που μπορούν να γίνουν από πολλές γλώσσες προγραμματισμού, όπως για παράδειγμα στην Java, PHP, Python, C++ κ.α.

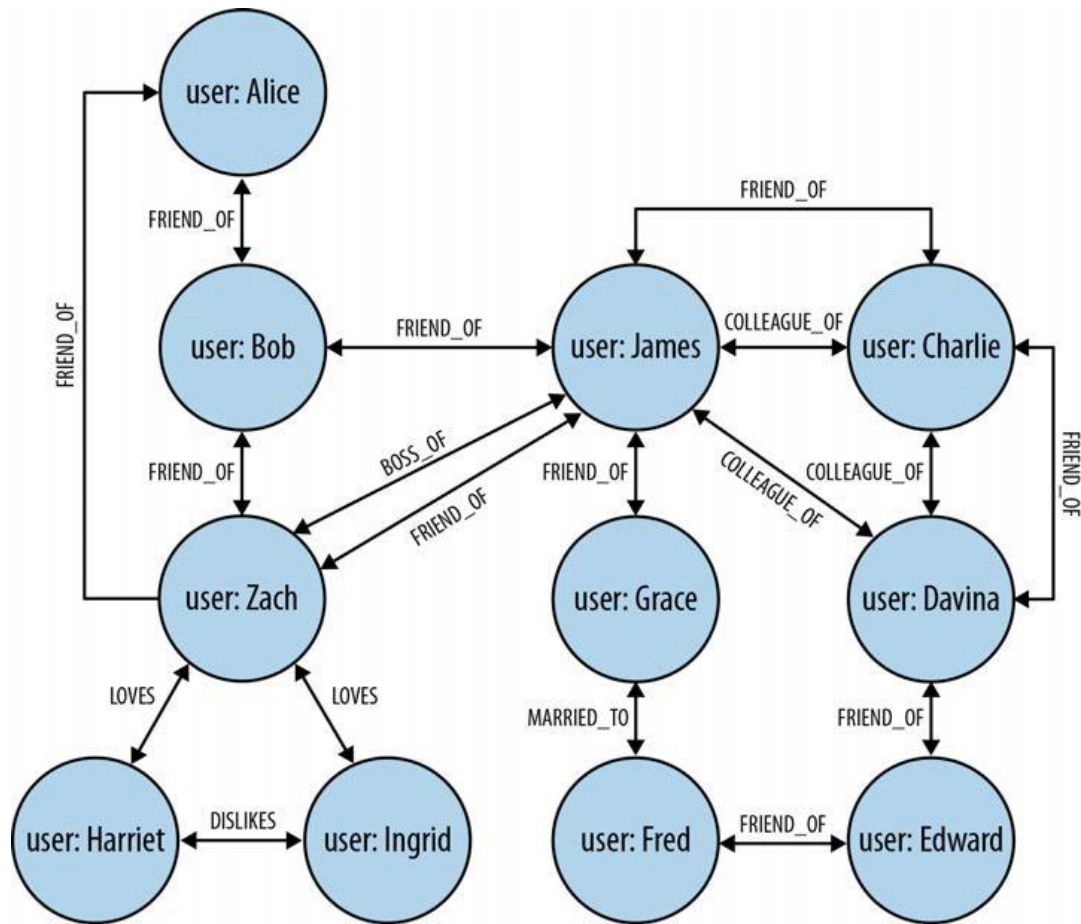
3.5.1.1 Σχεσιακή βάση δεδομένων (Relational database management system)

Η «MySQL» (My – Structured Query Language) είναι ένα από τα δημοφιλέστερα συστήματα διαχείρισης βάσεων δεδομένων (RDBMS – Relational Database Management System) για χρήση στις διαδικτυακές εφαρμογές και αποτελεί το δεύτερο πιο διαδεδομένο σύστημα μετά την «Sqlite». Το σχεσιακό μοντέλο ΒΔ αποτελείται από μια συλλογή πινάκων με στήλες και γραμμές. Για να δημιουργηθεί κάποια σχέση μεταξύ δύο πινάκων, θα πρέπει στον ένα πίνακα να υπάρχει μια στήλη η οποία θα δηλώνει το πρωτεύον κλειδί (primary key) και στον άλλο να υπάρχει μια στήλη η οποία να παραπέμπει στο πρωτεύον κλειδί του πρώτου πίνακα, το οποίο ονομάζεται ξένο κλειδί (foreign key). Η MySQL χρησιμοποιείται από αρκετά προγράμματα ανοικτού κώδικα όπως Joomla, WordPress, phpBB, MyBB, Drupal και μεγάλου βεληνεκούς ιστοσελίδες όπως Facebook, Google (όχι στις αναζητήσεις), Wikipedia κ.α. (Oracle, 2013). Η «MySQL» μαζί με το εργαλείο «PhpMyAdmin» ανήκουν στην κατηγορία ανοικτού κώδικα. Το

PhpMyAdmin είναι πρόγραμμα γραμμένο στη γλώσσα προγραμματισμού PHP και αποσκοπεί στη διαχείριση της «MySQL» μέσω προγράμματος περιήγησης (browser). Μπορεί να πραγματοποιήσει διάφορες εντολές όπως τη δημιουργία, τη μορφοποίηση ή τη διαγραφή ΒΔ, πινάκων, πεδίων ή απλά μια εγγραφή εκτελώντας εντολές «SQL». Στην εφαρμογή της παρούσας πτυχιακής εργασίας χρησιμοποιήθηκε ως ΒΔ RDBMS η «MySQL» και ως περιβάλλον διαχείρισης της το πρόγραμμα «PhpMyAdmin». Επίσης, στο προγραμματιστικό κομμάτι PHP, το οποίο επικοινωνεί με την ΒΔ χρησιμοποιήθηκε «MySqli» (Mysql improved), μια νέα επέκταση της «MySQL» που προσφέρει καλύτερη ασφάλεια και επίδοση.

3.5.1.2 Βάσεις Δεδομένων με το μοντέλο γράφων (Graph Databases)

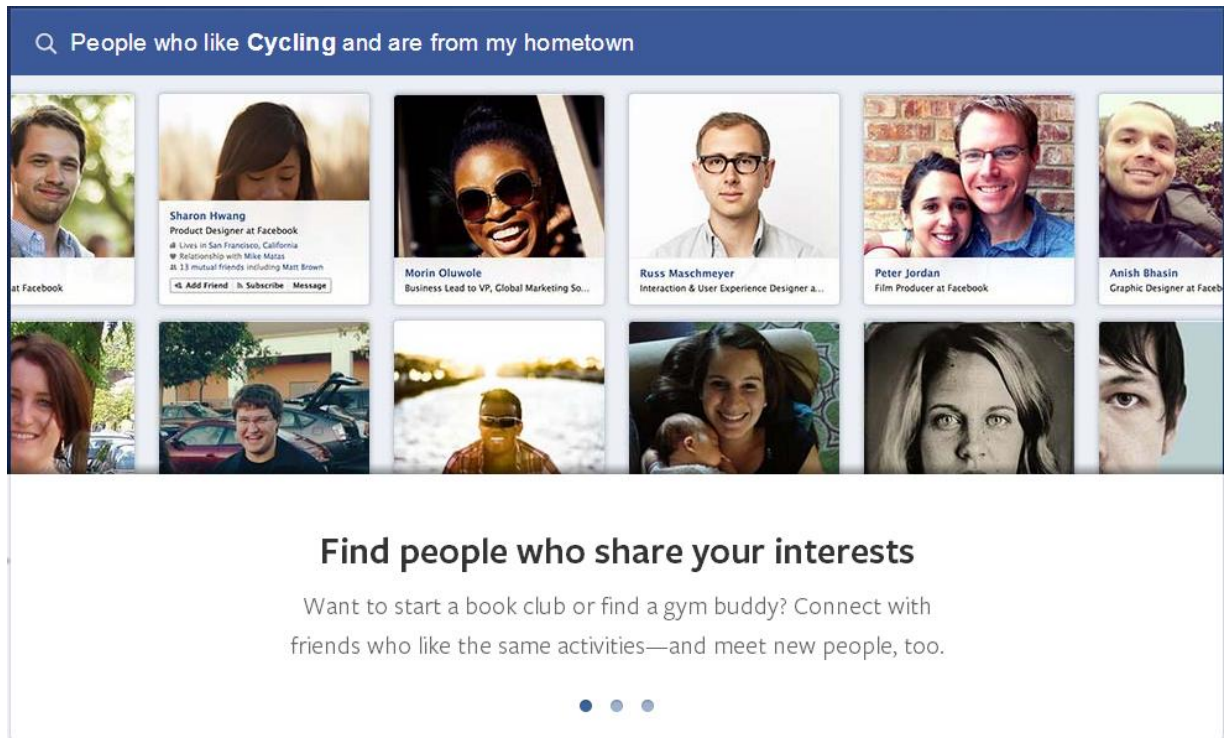
Μια ΒΔ γράφων, για να περιγράψει την δομή των δεδομένων της, χρησιμοποιεί την δομή ενός γραφήματος με κόμβους (nodes), ακμές (edges) και σχέσεις (relationships). Ένας γράφος αποτελείται από ένα σύνολο από οντότητες ως κόμβους και τους τρόπους με τον οποίους αυτές οι οντότητες συσχετίζονται μεταξύ τους, ως σχέσεις. Ένα γράφημα ιδιοτήτων (property graph) περιέχει (1) κόμβους και σχέσεις, (2) ιδιότητες του κάθε κόμβου (key-value pairs), (3) σχέσεις οι οποίες έχουν όνομα, ιδιότητες (key-value pairs), κατεύθυνση και απαραιτήτως έχουν αρχικό και τελικό κόμβο Έτσι, σε σχέση με μια RDBMS διαφέρει σε ένα αρκετά μεγάλο βαθμό. Στην Εικόνα 17 φαίνεται ένα μικρό κοινωνικό δίκτυο αποτυπωμένο σε γράφημα ιδιοτήτων, όπου οι σχέσεις των ατόμων διακρίνονται ξεκάθαρα.



Εικόνα 17: Μοντέλο φίλων και των σχέσεων μεταξύ τους με γράφημα

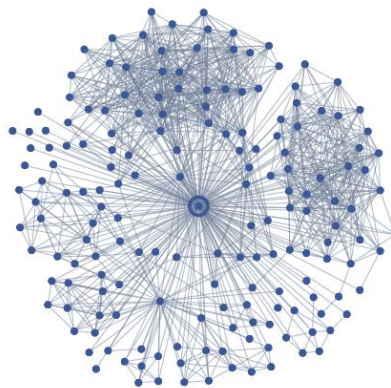
(Robinson, Webber, & Eifrem, 2013)

Οι ΒΔ γράφων έχουν την δυνατότητα να αποδώσουν πολύ καλύτερα σε σχέση με μια RDBMS, όταν η ΒΔ αφορά σχέσεις μεταξύ οντοτήτων, όπως περιγράφει η Εικόνα 17. Το Facebook, από το 2013 έχει αλλάξει τον τρόπο με τον οποίο οι χρήστες ψάχνουν στο σύστημά του, δημιουργώντας μια νέα λειτουργία αναζήτησης, το Graph Search. Αξιοποιώντας τις σχέσεις που ήδη έχουν δημιουργήσει οι χρήστες του, έχει προχωρήσει στην δημιουργία ενός νέου προγράμματος που έχει ονομάσει «Unicorn» (Curtiss et al., 2013), πετυχαίνοντας μέσα σε ελάχιστο χρόνο την επιστροφή αποτελεσμάτων σε πιο περίπλοκα ερωτήματα (queries) όπως για παράδειγμα στην Εικόνα 18.



Εικόνα 18: Facebook Graph Search

Το γεγονός αυτό, χωρίς το μοντέλο που έχει η ΒΔ γράφων δεν θα ήταν εφικτό. Με το υφιστάμενο μοντέλο της RDBMS, μέσα σε τρισεκατομμύρια χρήστες, για τέτοια ερωτήματα θα απαιτούσε πάρα πολλές συνενώσεις επιμέρους πινάκων (JOINS) και υπερβολικό χρόνο απόκρισης. Έτσι, η εταιρία Facebook έχει δημιουργήσει μια ξεχωριστή ΒΔ με γράφους, διατηρώντας μέσα τις σχέσεις των χρηστών της (βλέπε Εικόνα 19).



Εικόνα 19: Σχέσεις φίλων στο κοινωνικό γράφημα. Ο κεντρικός χρήστης απεικονίζεται στο κέντρο
(Curtiss et al., 2013)

Η ΒΔ γράφων είναι μια νέα τεχνολογία η οποία δημιουργήθηκε έχοντας στο επίκεντρο τις σχέσεις μεταξύ των οντοτήτων. Η προσέγγιση για την δημιουργία αποθήκευσης δεδομένων προσφέρει καλύτερη απόδοση σε ΒΔ με πολλά δεδομένα. Το Facebook έχει αναπτύξει τη δική του τεχνολογία για Graph Databases, όπως και πολλές άλλες μεγάλες εταιρίες στο διαδίκτυο. Παρόλα αυτά δεν είναι διαθέσιμη στο κοινό. Η Neo4j, είναι μια ανοικτού λογισμικού ΒΔ γράφων η οποία υποστηρίζεται την Neo Technology. Σε σχέση με την MySQL, χρησιμοποιεί παρομοίου τύπου γλώσσα για ερωτήματα, την Cypher. Στο πείραμα που διεξάχθηκε με σκοπό να αποδειχθεί η καλή απόδοση της ΒΔ Neo4j (Robinson, Webber, & Eifrem, 2013), χρησιμοποιήθηκε ένα σύνολο δεδομένων εκπροσωπώντας ένα εκατομμύριο ανθρώπους, που το κάθε άτομο είχε περίπου πενήντα φίλους. Σε βάθος 2 (friends-of-friends) (βλέπε Πίνακα 1), και οι δύο ΒΔ εκτέλεσαν το ερώτημα σε πολύ καλό χρόνο. Παρόλα αυτά, από το βάθος 3 (friends-of-friends-of-friends), είναι ξεκάθαρο πως η RDBMS με 30 δευτερόλεπτα χρόνο να ανταποκριθεί, δεν μπορεί να συγκριθεί με τον χρόνο απόκρισης της Neo4j, η οποία χρειάστηκε μόλις 168 χιλιοστά του δευτερολέπτου.

Βάθος	Χρόνος απόκρισης RDBMS (s)	Χρόνος απόκρισης Neo4j (s)	Αποτελέσματα που επέστρεψε
2	0,016	0,01	~2500
3	30,267	0,168	~110,000
4	1543,505	1,359	~600,000
5	Δεν τελείωσε	2,132	~800,000

Πίνακας 1: Βρίσκοντας φίλους σε εκτενές βάθος από δεδομένα μιας RDBMS σε σχέση με την ΒΔ Neo4j
(Robinson, Webber, & Eifrem, 2013)

Μια ΒΔ όπως η Neo4j, δεν χρησιμοποιείται απαραίτητα μόνο σε κοινωνικά δίκτυα. Αναφορικά, στην παρούσα πτυχιακή εργασία, μπορεί να αξιοποιηθεί για δύο σημαντικούς λόγους. Ο πρώτος λόγος αναφέρεται στα δεδομένα τα οποία συλλέχθηκαν από διαδρομές και απαιτούν πολλές εγγραφές στην RDBMS. Ο δεύτερος και σημαντικότερος λόγος είναι ότι το κάθε σημείο και η κάθε διασταύρωση που αναγνωρίζεται είναι άμεσα συσχετισμένα με άλλα σημεία. Έτσι, στην

περίπτωση που ο όγκος των δεδομένων μεγαλώσει και ζητηθεί να βρεθεί ένα μονοπάτι με βάση κάποιους κόμβους, χρησιμοποιώντας την ΒΔ Neo4j, η διαδικασία ανεύρεσης θα είναι πιο αποδοτική και τα αποτελέσματα θα επιστραφούν πολύ πιο γρήγορα.

3.5.2 PHP (PHP: Hypertext Preprocessor)

Η «PHP» είναι μια γλώσσα προγραμματισμού που χρησιμοποιείται για τη δημιουργία ιστοσελίδων με δυναμικό περιεχόμενο. Μια σελίδα PHP περνά από επεξεργασία και ερμηνεύεται από ένα συμβατό διακομιστή του Παγκόσμιου Ιστού (π.χ. Apache), ώστε το αποτέλεσμα της να παραχθεί σε πραγματικό χρόνο. Συνήθως, το αποτέλεσμα στέλνεται στο πρόγραμμα περιήγησης των επισκεπτών σε μορφή κώδικα HTML. Η «PHP» σχεδιάστηκε το 1994 και αρχικά ήταν έργο ενός ανθρώπου, του Rasmus Lerdorf (Tatroe, MacIntyre, & Lerdorf, 2013). Παρόμοιες γλώσσες προγραμματισμού είναι η JSP (JavaServer Pages), η Perl και ASP (Active Server Pages). Παρόλα αυτά, ο μεγάλος αριθμός ενσωματωμένων βιβλιοθηκών που έχει η PHP την καθιστά ανώτερη από τις υπόλοιπες, καθώς επίσης μπορεί να διασυνδεθεί με διαφορετικές βάσεις δεδομένων. Μέχρι τον Γενάρη του 2013, είχε χρησιμοποιηθεί σε περισσότερες από 244 εκατομμύρια ιστοσελίδες και σε 2.1 εκατομμύρια IPs¹. Στην παρούσα έρευνα η «PHP» χρησιμοποιήθηκε στην επεξεργασία δεδομένων από τη ΒΔ, στην ιστοσελίδα αλλά και στην επεξεργασία δεδομένων από και προς την εφαρμογή Android.

3.5.3 Java (γλώσσα προγραμματισμού)

Η «Java» είναι μια αντικειμενοστραφής γλώσσα προγραμματισμού που σχεδιάστηκε από την εταιρεία πληροφορικής Sun Microsystems. Στην εφαρμογή της πτυχιακής μου εργασίας θα χρησιμοποιηθεί μαζί με τις βιβλιοθήκες «Android» (SDK – Software Development Kit). Ένα από τα βασικότερα πλεονεκτήματα της Java σε σχέση με άλλες γλώσσες είναι η ανεξαρτησία του λειτουργικού συστήματος και πλατφόρμας. Τα προγράμματα που είναι γραμμένα σε Java τρέχουν χωρίς κανένα πρόβλημα σε Windows, Linux, Unix και Macintosh (σύντομα θα τρέχουν

¹ Usage Stats for January 2013 - <http://www.php.net/usage.php>

και σε Playstation καθώς και σε άλλες κονσόλες παιχνιδιών) χωρίς να χρειαστεί να ξαναγίνει μεταγλώττιση (compiling) ή να αλλάξει ο πηγαίος κώδικας για κάθε διαφορετικό λειτουργικό σύστημα. Για να επιτευχθεί όμως αυτό χρειαζόταν κάποιος τρόπος έτσι ώστε τα προγράμματα που ήταν γραμμένα σε Java να μπορούν να είναι «κατανοητά» από κάθε υπολογιστή ανεξάρτητα του είδους επεξεργαστή αλλά και λειτουργικού συστήματος. Ο λόγος είναι ότι κάθε κεντρική μονάδα επεξεργασίας κατανοεί διαφορετικό κώδικα μηχανής. Ο συμβολικός κώδικας (assembly) που μεταφράζεται και εκτελείται σε Windows είναι διαφορετικός από αυτόν που μεταφράζεται και εκτελείται σε έναν υπολογιστή Macintosh. Η λύση δόθηκε με την ανάπτυξη της Εικονικής Μηχανής (Virtual Machine) από την εταιρία.

3.5.4 Android SDK

Το Android, είναι λειτουργικό κυρίως για κινητές συσκευές τύπου ανοικτού κώδικα (OSS). Ξεκίνησε το 2003 από την εταιρία Android Inc., την περίοδο που οι μόνοι κύριοι ανταγωνιστές ήταν η Microsoft με τα λειτουργικά Symbian και Microsoft Windows. Το 2005 αγοράστηκε από την εταιρία Google, ωθώντας την εταιρία στην αγορά των έξυπνων κινητών συσκευών. Το λειτουργικό βασίζεται στον πυρήνα Linux και είναι γραμμένο στις γλώσσες προγραμματισμού C, C++ και Java. Το λειτουργικό Android, έχει την δυνατότητα να προσαρμοστεί σε πολλές κινητές συσκευές αλλά και σε έξυπνες τηλεοράσεις σε σχέση με το iOS που απαραίτητα πρέπει να λειτουργεί σε υλικό (hardware) της εταιρίας Apple. Με το SDK που παρέχει στους προγραμματιστές, τους δίνει το μέσο για να το αξιοποιήσουν στο έπακρο. Οι βιβλιοθήκες του SDK δίνουν πρόσβαση στο δίκτυο του κινητού τηλεφώνου, στις λειτουργίες εντοπισμού θέσης (location-based services) όπως είναι το GPS και η αναγνώριση τοποθεσίας μέσω Wi-Fi, πλήρη έλεγχο στα εργαλεία χαρτών, APIs στο υλικό για τους αισθητήρες της συσκευής όπως το βαρόμετρο, πυξίδα κ.α., σε άλλες διεργασίες της συσκευής όπως κλήσεις, αποστολή μηνυμάτων και σε πολλές άλλες λειτουργίες (Meier, 2012). Στο τμήμα της έρευνας όπου υλοποιήθηκε η εφαρμογή Android, χρησιμοποιήθηκαν κάποιες από τις πιο πάνω βιβλιοθήκες μέσω του ολοκληρωμένου περιβάλλοντος ανάπτυξης λογισμικού Eclipse.

4. Μεθοδολογία

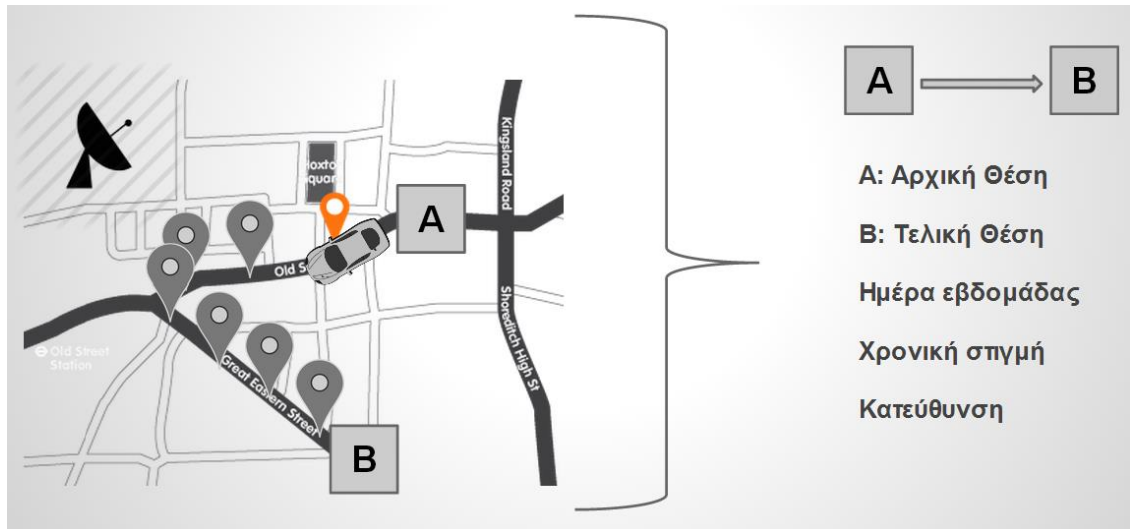
Πρωταρχικός σκοπός της παρούσας έρευνας είναι η κατασκευή μιας αυτοματοποιημένης διαδικασίας αναγνώρισης κόμβων χρησιμοποιώντας τεχνικές μηχανικής μάθησης με την χρήση και ανάλυση δεδομένων από διάφορες διαδρομές. Για την επίλυση του προβλήματος αυτού, ανίχνευσης κόμβων, αναπτύχθηκε μια διαδικασία η οποία αποτελείται από ένα σύνολο διαφορετικών τεχνολογιών και αποσκοπεί στην συλλογή, εμφάνιση, εξαγωγή και ανάλυση δεδομένων. Η προσέγγιση της παρούσας πτυχιακής εργασίας στην αναγνώριση κόμβων, πραγματοποιήθηκε αναλύοντας μόνο δεδομένα που πάρθηκαν κατά την διακίνηση οχημάτων μέσω μιας αφαιρετικής εφαρμογής Android με υποστήριξη τεχνολογίας GPS και μέσω της πρακτικής Crowdsourcing.

Το πρώτο στάδιο της έρευνας ήταν η αποσαφήνιση και καταγραφή των απαραίτητων πληροφοριών που έπρεπε να συλλεχθούν, και ακολούθως η σχεδίαση και υλοποίηση μιας ΒΔ στην οποία τα ακατέργαστα δεδομένα θα αποθηκεύονται. Έπειτα, δεύτερο βήμα ήταν η υλοποίηση της εφαρμογής Android που θα επέτρεπε την συλλογή πληροφοριών με την χρήση τεχνολογίας GPS. Στην επόμενη φάση, τα δεδομένα αναπαραστάθηκαν και τροποποιήθηκαν με στόχο την δημιουργία ενός συγκροτημένου συνόλου δεδομένων για ανάλυση. Το τελευταίο στάδιο ήταν η ανάλυση δεδομένων αξιοποιώντας τεχνικές μηχανικής μάθησης, και πιο συγκεκριμένα του προγράμματος WEKA. Τέλος, χρησιμοποιήθηκαν γραφήματα (graphs) για περαιτέρω ανάλυση και παροχή επιπρόσθετων πληροφοριών στους χρήστες (π.χ. υπόδειξη εναλλακτικών διαδρομών). Για τον σκοπό αυτό, αξιοποιήθηκαν οι βάσεις γράφων (Graph Databases). Η στρατηγική έρευνας που βασίστηκε η παρούσα πτυχιακή εργασία είναι ποσοτική. Η ερευνητική μέθοδος εναπόκειται σε πείραμα, αφού αφορά ανάπτυξη εφαρμογής, υλοποίηση συστήματος και εξαγωγή αποτελεσμάτων.

4.1 Δεδομένα και κατασκευή ΒΔ

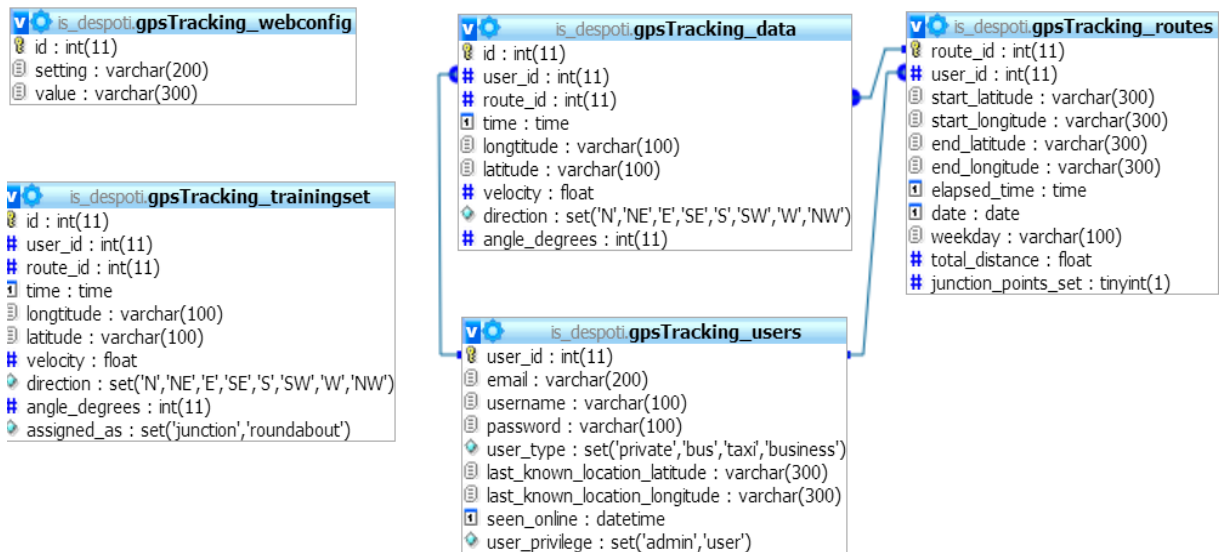
Ορίζουμε τη διαδρομή κάθε οχήματος σαν μια συλλογή από σημεία, με αυστηρά καθορισμένο το σημείο εκκίνησης και το σημείο τερματισμού. Τα σημεία αυτά αποτελούνται από τις

συντεταγμένες γεωγραφικού πλάτους (latitude) και γεωγραφικού μήκους (longitude). Παράλληλα, κατά την φάση μετακίνησης του οχήματος καταγράφουμε πληροφορίες όπως ταχύτητα του οχήματος, κατεύθυνση του οχήματος (π.χ. νοτιοανατολικά), γωνία κατεύθυνσης, ημέρα εβδομάδας και χρονική στιγμή (timestamp) (βλέπε Εικόνα 20).



Εικόνα 20: Αναπαράσταση συλλογής πληροφοριών

Επιπρόσθετα, εκτός από τα δεδομένα της διαδρομή, η ΒΔ χρησιμοποιήθηκε και για την αποθήκευση δεδομένων για τους χρήστες του συστήματος, τις ρυθμίσεις της ιστοσελίδας, αλλά και για το σύνολο δεδομένων το οποίο αναλύθηκε. Με βάση τα πιο πάνω στοιχεία, υλοποιήσαμε μια MySQL ΒΔ τύπου RDBMS η οποία περιγράφεται στο πιο κάτω σχήμα (βλέπε Εικόνα 21).



Εικόνα 21: ΒΔ συστήματος

4.2 Ανάπτυξη εφαρμογής Android

4.2.1 Ανάλυση απαιτήσεων

Στο πρώτο στάδιο, πριν από την υλοποίηση της εφαρμογής, έπρεπε να καθοριστούν οι απαιτήσεις και να οριστεί το πρωτόκολλο επικοινωνίας μεταξύ του συστήματος. Επιπλέον, σημαντική επιλογή στη φάση αυτή ήταν η συσκευή στην οποία η πιλοτική εφαρμογή θα έτρεχε. Το λειτουργικό το οποίο επιλέξαμε είναι το Android της εταιρείας Google. Εκτός από την ευρεία χρήση του, κύριος λόγος για την επιλογή αυτή είναι η ελευθερία που προσφέρει το πλαίσιο ανάπτυξης του στις διάφορες συσκευές. Η εφαρμογή υλοποιήθηκε με τη γλώσσα προγραμματισμού Java και με την αξιοποίηση των βιβλιοθηκών του Android SDK. Στη συνέχεια, επιλέξαμε την έκδοση Android 4.0 (έκδοση API 14) ως ελάχιστη έκδοση στο θέμα συμβατότητας, με βάση τις ελάχιστες απαιτήσεις που η εφαρμογή χρειάζεται να λειτουργήσει. Επιπρόσθετα, για να είναι συμβατή η εφαρμογή με την συσκευή, πέραν από τον καθορισμό της έκδοσης του λειτουργικού, η συσκευή χρειάζεται να υποστηρίζει σύνδεση στο διαδίκτυο (Wi-Fi), λειτουργία GPS και αισθητήρες ανίχνευσης μαγνητικών πεδίων (πυξίδα - compass). Το πρωτόκολλο αποστολής και λήψης δεδομένων στο σύστημα ορίστηκε με την μορφή JSON (JavaScript Object Notation). Η μορφή ανταλλαγής δεδομένων JSON αποτελεί μια από τις πιο

διαδεδομένες πρακτικές ανταλλαγής πληροφοριών/δεδομένων η οποία είναι βασισμένη στη μορφή «χαρακτηριστικό – τιμή» (attribute-value).

4.2.2 Υλοποίηση εφαρμογής

Η υλοποίηση της εφαρμογής βασίστηκε στη θεωρία της αφαιρετικότητας με σκοπό οι χρήστες να μπορούν να συμβάλουν στη συλλογή πληροφοριών με τις ελάχιστες ενέργειες και γνώσεις. Η θεωρία αυτή αφορά μεθόδους που εφαρμόζονται στον τρόπο λειτουργίας ενός συστήματος οι οποίες αφαιρούν σύνθετες λεπτομέρειες διατηρώντας έτσι μόνο τις πληροφορίες που είναι απαραίτητες. Με την βοήθεια του ολοκληρωμένου περιβάλλοντος ανάπτυξης λογισμικού Eclipse, αξιοποιήσαμε τις βιβλιοθήκες Android για να καταστεί δυνατή η πρόσβαση στις διάφορες λειτουργίες της συσκευής και λειτουργικού. Έτσι, προχωρήσαμε στη δημιουργία των απαραίτητων κλάσεων (αυτόνομες οντότητες με ταυτότητα, ιδιότητες και διαδικασίες), σε συνδυασμό με τις κλάσεις του Android SDK. Λαμβάνοντας ως σημαντικό παράγοντα τη θετική εμπειρία του χρήστη, δημιουργήσαμε μια επιπρόσθετη κλάση η οποία αφορά την οπτική επαφή του χρήστη με την πορεία της διαδρομής χρησιμοποιώντας βιβλιοθήκες «Osmdroid» από τον οργανισμό Open Street Maps (OSM) και τον χάρτη της Κύπρου. Για τον σκοπό αυτό, χρησιμοποιήσαμε την εφαρμογή «Maperitive» μέσω της οποίας κατέστη δυνατή η εξαγωγή του χάρτη της Κύπρου από τους χάρτες OSM. Αξιοποιώντας τα αντικείμενα των κλάσεων αυτών, προχωρήσαμε στην υλοποίηση δίνοντας προτεραιότητα στην ενεργοποίηση της λειτουργίας GPS της συσκευής, προτού ξεκινήσει η διαδικασία καταγραφής. Συνολικά γράφτηκαν 15 κλάσεις και δημιουργήθηκαν τέσσερις οθόνες.

4.2.2.1 Λειτουργίες εφαρμογής και κλάσεις

Για την επίτευξη του σκοπού της εφαρμογής κατηγοριοποιήσαμε τις λειτουργίες της εφαρμογής ως εξής:

Λειτουργίες:

1. Καταγραφής δεδομένων
2. Αποθήκευσης δεδομένων
3. Αποστολής/Λήψης δεδομένων
4. Ελέγχου
5. Οθόνης

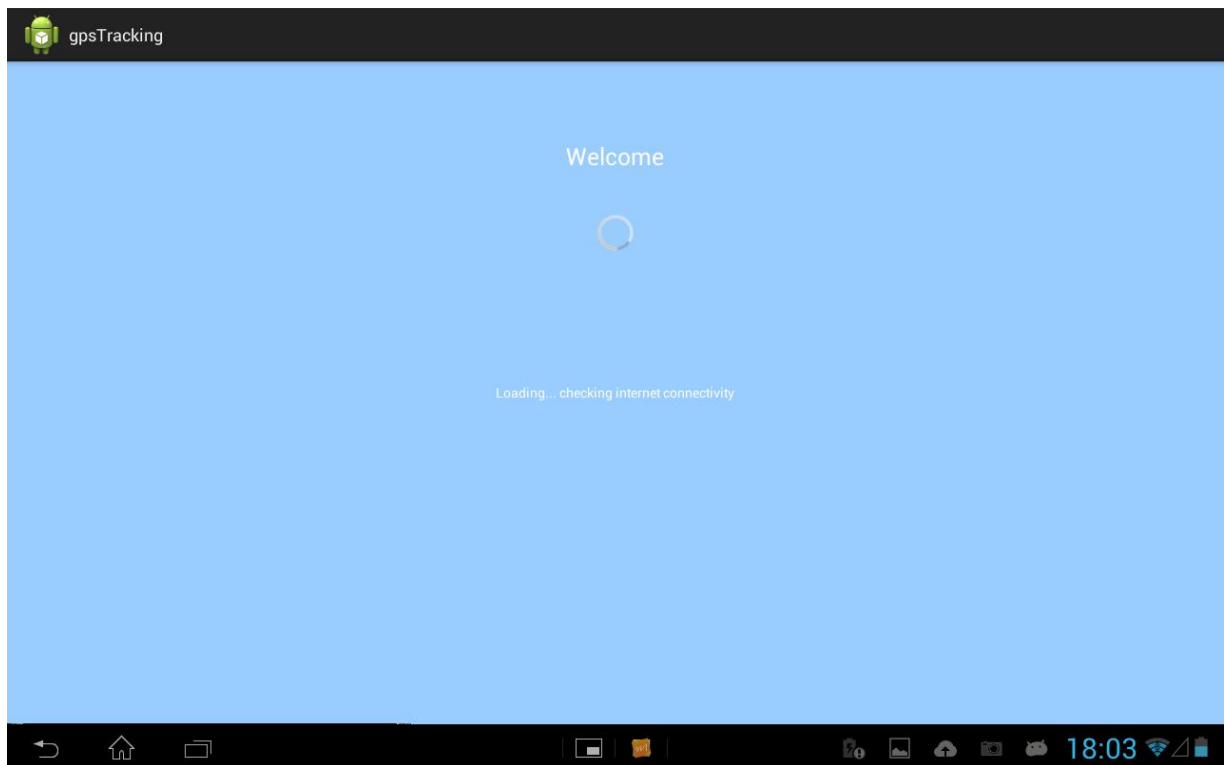
Αρχικά, στις λειτουργίες καταγραφής δεδομένων συγκαταλέγονται οι κλάσεις οι οποίες δίνουν πρόσβαση σε μεταβαλλόμενα δεδομένα όπως γεωγραφική θέση, ταχύτητα, γωνία κατεύθυνσης και χρονική στιγμή της συσκευής. Στην κατηγορία λειτουργιών αποθήκευσης δεδομένων, υλοποιήσαμε μηχανισμούς οι οποίοι διαχειρίζονται τα δεδομένα στην αποθηκευτική μνήμη της συσκευής, σε περιπτώσεις που δεν υπήρχε η δυνατότητα αποστολής στον Server (διακομιστή) λόγω απώλειας σύνδεσης στο διαδίκτυο. Επιπρόσθετα, προσθέσαμε λειτουργία η οποία αφορά την αποθήκευση ρυθμίσεων της εφαρμογής. Για παράδειγμα, κατά την είσοδο του χρήστη (login), να πραγματοποιείται αποθήκευση των στοιχείων του χωρίς να χρειάζεται να επαναληφθεί η διαδικασία εισόδου. Η λειτουργίες αποστολής δεδομένων, αναφέρονται στη διεξαγωγή σύνδεσης με τον Server και αποστολή ή λήψη δεδομένων ανάλογα με την αίτηση της εφαρμογής. Συνδέσεις με τον Server πραγματοποιούνται στις περιπτώσεις εγγραφής του χρήστη (register), εισόδου χρήστη και αποστολής δεδομένων από τις διαδρομές. Αναφορικά με τις λειτουργίες ελέγχου, συμπεριλαμβάνονται δύο διαδικασίες ελέγχων. Η μια αφορά τον έλεγχο συνδεσιμότητας στο διαδίκτυο, και η δεύτερη αποσκοπεί στον έλεγχο της λειτουργίας GPS της συσκευής. Τέλος, οι λειτουργίες οθόνης αφορούν κυρίως τον σχεδιασμό της διεπαφής του χρήστη και τους τρόπους αλληλεπίδρασης του με την εφαρμογή.

4.2.2.2 Οθόνες εφαρμογής

Παρακάτω αριθμείται η σειρά εμφάνισης των οθονών της παρούσας εφαρμογής:

1. Οθόνη καλωσορίσματος

Στην πρώτη οθόνη (βλέπε Εικόνα 22) καλωσορίζουμε τον χρήστη, καθώς η εφαρμογή φορτώνει τα απαραίτητα αρχεία και πραγματοποιεί έλεγχο συνδεσιμότητας στο διαδίκτυο.



Εικόνα 22: Οθόνη καλωσορίσματος εφαρμογής

2. Οθόνη εισόδου χρήστη

The screenshot shows the 'Sign in' screen of the 'gpsTracking' app. The title bar at the top left contains the app icon and the text 'Sign in'. The main area has a light blue background. In the center, there are two input fields: 'Email' and 'Password'. Below the 'Password' field is a 'Sign in' button. Underneath the button, there is a link that says 'Or create new account'. At the bottom of the screen, there is an Android navigation bar with icons for back, home, and recent apps, along with a status bar showing the time as 17:59 and various system icons.

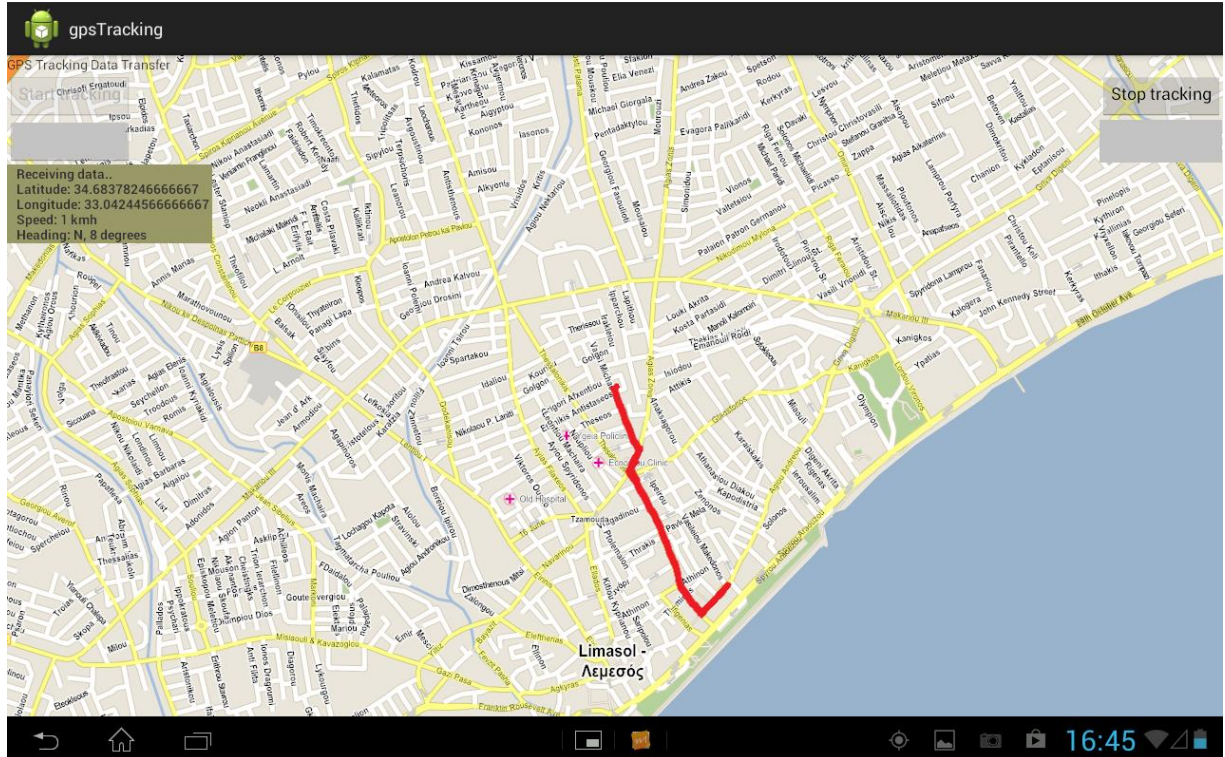
Εικόνα 23: Οθόνη εισόδου χρήστη

3. Οθόνη εγγραφής χρήστη

The screenshot shows the 'Register new account' screen of the 'gpsTracking' app. The title bar at the top left contains the app icon and the text 'gpsTracking'. The main area has a light blue background. The form consists of several fields: 'Create a username' with a sub-label 'create a username', 'Enter a valid email address' with a sub-label 'Enter a valid email address', 'Create password' with a sub-label 'Create password', and 'Re-type password' with a sub-label 'Re-type password'. A 'Register now' button is located at the bottom right of the form. At the bottom of the screen, there is an Android navigation bar with icons for back, home, and recent apps, along with a status bar showing the time as 17:59 and various system icons.

Εικόνα 24: Οθόνη εγγραφής χρήστη

4. Οθόνη καταγραφής δεδομένων



Εικόνα 25: Οθόνη καταγραφής δεδομένων

4.2.3 Σενάριο χρήσης εφαρμογής – διαδικασία λειτουργίας

Η διαδικασία χρήσης της εφαρμογής είναι απλοποιημένη σε μεγάλο βαθμό, δεδομένου ότι μόνο δύο ενέργειες απαιτούνται για τον σκοπό αυτό. Η καταγραφή δεδομένων ξεκινά με το κουμπί «Start tracking» και τελειώνει με το κουμπί «Stop tracking». Εντούτοις, η διαδικασία πίσω από την εφαρμογή διαφέρει σε πολυπλοκότητα (βλέπε Εικόνα 26). Κατά την ενεργοποίηση της εφαρμογής, πραγματοποιείται έλεγχος για την ύπαρξη πιθανών στοιχείων του χρήστη στη συσκευή. Αν δεν βρεθούν τα στοιχεία αυτά, τότε εμφανίζεται η οθόνη εισόδου και εγγραφής. Στη συνέχεια, εφαρμόζεται έλεγχος για προσβασιμότητα στο διαδίκτυο. Αν υπάρχει, τότε ένας άλλος έλεγχος παίρνει μέρος με τον οποίο η εφαρμογή ελέγχει αν υπάρχουν ήδη καταγραμμένα δεδομένα στη συσκευή. Αν παρατηρηθούν υπάρχοντα καταγραμμένα δεδομένα, τότε ακολουθείται η λειτουργία αποστολής δεδομένων. Ο χρήστης δεν βλέπει κάτι περισσότερο από

τον χάρτη και τα δύο κουμπιά, από τα οποία μόνο το κουμπί «Start tracking» είναι ενεργοποιημένο. Έτσι, όταν ο χρήστης ενεργοποιήσει την διαδικασία καταγραφής με το κουμπί «Start tracking», ελέγχεται αν η λειτουργία GPS της συσκευής είναι ανοιχτή. Αν δεν είναι ανοιχτή, τότε εμφανίζεται η οθόνη των ρυθμίσεων θέσης του λειτουργικού Android, με σκοπό την ενεργοποίηση της λειτουργίας GPS από τον χρήστη. Τέλος, όταν ο χρήστης τερματίσει τη διαδικασία με το κουμπί «Stop tracking», πραγματοποιείται πάλι έλεγχος για σύνδεση στο διαδίκτυο. Αν η δυνατότητα αυτή είναι εφικτή, τότε τα δεδομένα στέλνονται στον Server ο οποίος αναλαμβάνει την αποστολή τους στη ΒΔ. Αν δεν υπάρχει σύνδεση στο διαδίκτυο, τότε τα δεδομένα αποθηκεύονται με σκοπό να σταλούν την επόμενη φορά που θα καταστεί κατορθωτό.

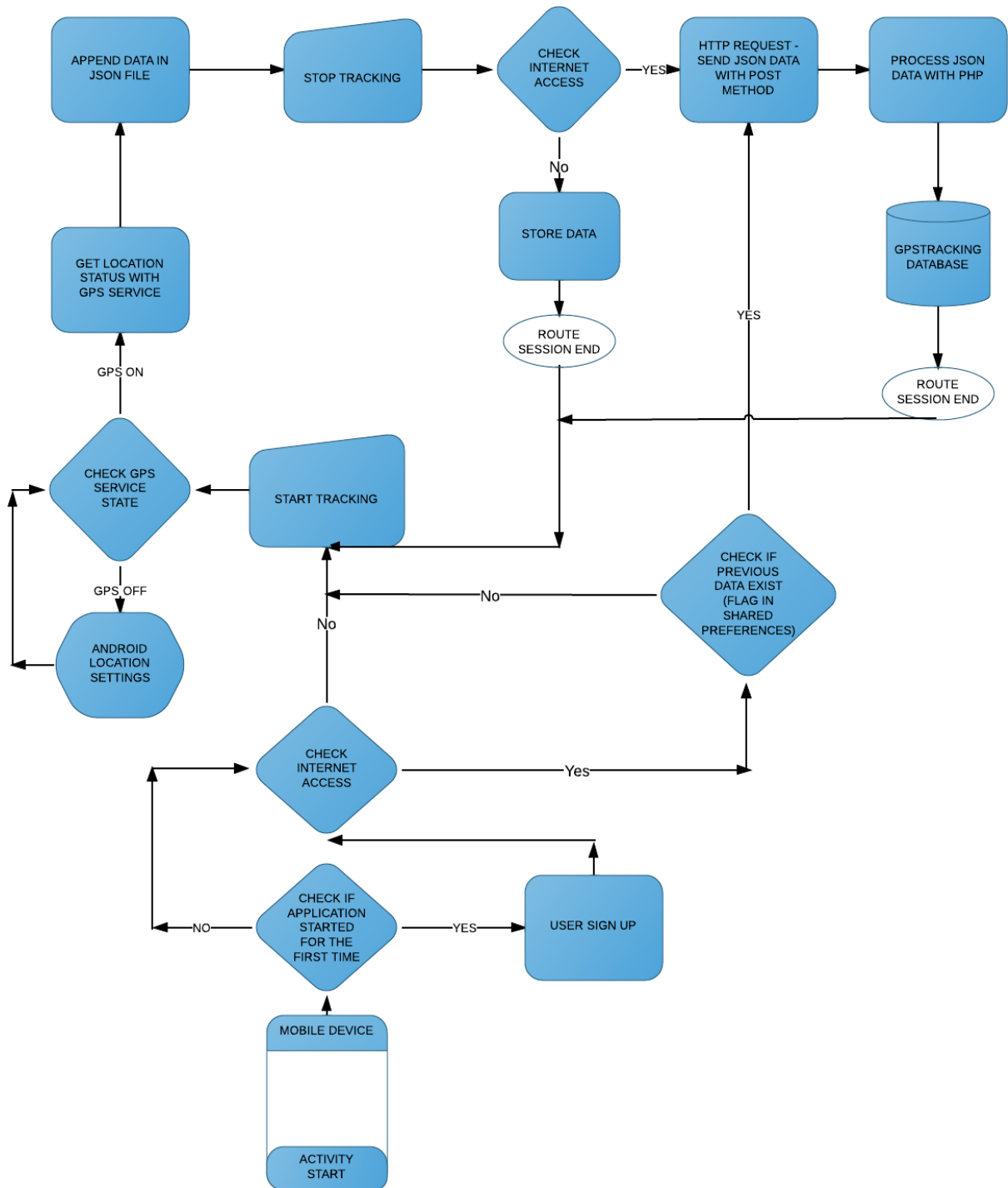
4.2.3.1 Κανόνες στη διαδικασία καταγραφής

Κατά τη διαδικασία καταγραφής δεδομένων από την εφαρμογή, προχωρήσαμε στη δημιουργία κάποιων κανόνων με σκοπό την παράκαμψη καταγραφής άσκοπων δεδομένων και την αποφυγή δημιουργίας ενός ογκώδους πίνακα στη ΒΔ.

Κανόνες:

1. Αποφυγή καταγραφής ίδιων γεωγραφικών τοποθεσιών μεταξύ δύο θέσεων
2. Τοποθέτηση ελάχιστης απόστασης μεταξύ δύο θέσεων τα οποία καταγράφονται στα 5 m
3. Τοποθέτηση ελάχιστου χρόνου καταγραφής στα 2 sec
4. Αν η ταχύτητα οχήματος είναι μεταξύ 0 – 40 km/h και η κατεύθυνση του οχήματος έχει αλλάξει, τότε η ελάχιστη απόσταση καταγραφής να παραμένει σε 5 m και ο ελάχιστος χρόνος σε 2 sec
5. Αν η ταχύτητα του οχήματος βρίσκεται μεταξύ 41 – 60 km/h τότε η ελάχιστη απόσταση καταγραφής να μεταβάλλεται σε 15 m και ο ελάχιστος χρόνος σε 3 sec
6. Αν η ταχύτητα του οχήματος βρίσκεται μεταξύ 61 – 80 km/h τότε η ελάχιστη απόσταση καταγραφής να μεταβάλλεται σε 20 m και ο ελάχιστος χρόνος σε 4 sec
7. Αν η ταχύτητα του οχήματος βρίσκεται μεταξύ 81 – 100 km/h τότε η ελάχιστη απόσταση καταγραφής να μεταβάλλεται σε 25 m και ο ελάχιστος χρόνος σε 5 sec

8. Αν η ταχύτητα του οχήματος βρίσκεται μεταξύ 101 – 120 km/h τότε η ελάχιστη απόσταση καταγραφής να μεταβάλλεται σε 30 m και ο ελάχιστος χρόνος σε 6 sec
9. Αν η ταχύτητα του οχήματος υπερβαίνει τα 121 km/h τότε η ελάχιστη απόσταση καταγραφής να μεταβάλλεται σε 35 m και ο ελάχιστος χρόνος σε 7 sec



Εικόνα 26: Διαδικασία λειτουργίας εφαρμογής

4.3 Λειτουργίες και αξιοποίηση του Server

Η χρήση ενός server στο σύστημα καθίσταται απαραίτητη καθώς το σύστημα πρέπει απαραίτητως να συγκεντρώνει τα δεδομένα από τις διάφορες κινητές συσκευές των διαφόρων χρηστών σε μια κεντρική ΒΔ. Η αξιοποίησή του αφορά επίσης τις διαδικασίες με τις οποίες η κινητή συσκευή επικοινωνεί με τη ΒΔ αλλά και τις διαδικασίες που ο server αναλαμβάνει να επεξεργαστεί δεδομένα πριν να τα στείλει στη συσκευή ή στη ΒΔ. Τέλος, ο server είναι απαραίτητος γιατί αποτελεί την υποδομή για τον ιστότοπο του συστήματος.

Χρησιμοποιώντας τη γλώσσα προγραμματισμού PHP και με βάση το αντικειμενοστραφές μοντέλο, δημιουργήσαμε μια υποδομή όπου συγκεντρώσαμε όλες τις επιμέρους λειτουργίες σε μεθόδους που αφορούσαν σύνδεση με την βάση, ασφάλεια και ελέγχους, ανάκτηση δεδομένων και άλλες απαραίτητες διαδικασίες.

4.3.1 Επεξεργασία δεδομένων από και προς την κινητή συσκευή

Οι διαδικασίες της εφαρμογής που χρειάζονται την απαραίτητη επικοινωνία με τον server είναι η διαδικασία εισόδου του χρήστη, η διαδικασία εγγραφής και η αποστολή δεδομένων που έχουν καταγραφεί από κάποια διαδρομή. Όπως αναφέρθηκε στην υποενότητα «4.2.1 Ανάλυση απαιτήσεων», όλα τα δεδομένα στέλλονται σε μορφή JSON. Κατά τη διαδικασία εισόδου του χρήστη, η κινητή συσκευή αποστέλλει την ηλεκτρονική διεύθυνση και το προσωπικό κωδικό του χρήστη, και ακολούθως λαμβάνει ένα μήνυμα επιβεβαίωσης. Η ίδια διαδικασία ακολουθείται και για την εγγραφή του χρήστη με τα αντίστοιχα δεδομένα. Τα αρχεία στον server που είναι υπεύθυνα να λαμβάνουν τα δεδομένα στις δύο αυτές περιπτώσεις, πραγματοποιούν σύνδεση με τη ΒΔ, εφαρμόζουν ελέγχους, στέλνουν δεδομένα στη ΒΔ (αν χρειαστεί) και δίνουν το αποτέλεσμα στην κινητή συσκευή σε μορφή JSON. Στην περίπτωση που αποστέλλονται δεδομένα από διαδρομές, όλα τα σημεία τα οποία αποτελούν την διαδρομή μαζί με τις αντίστοιχες πληροφορίες, στέλλονται ομαδοποιημένα με την μορφή JSON πίνακα. Έτσι, με συνδυασμό της γλώσσας προγραμματισμού PHP, κατέστη δυνατή η προσαρμογή των δεδομένων σε ερωτήματα SQL, χωρίς κανένα περιορισμό στον αριθμό διαδρομών που μπορούν να διαχειριστούν κάθε φορά.

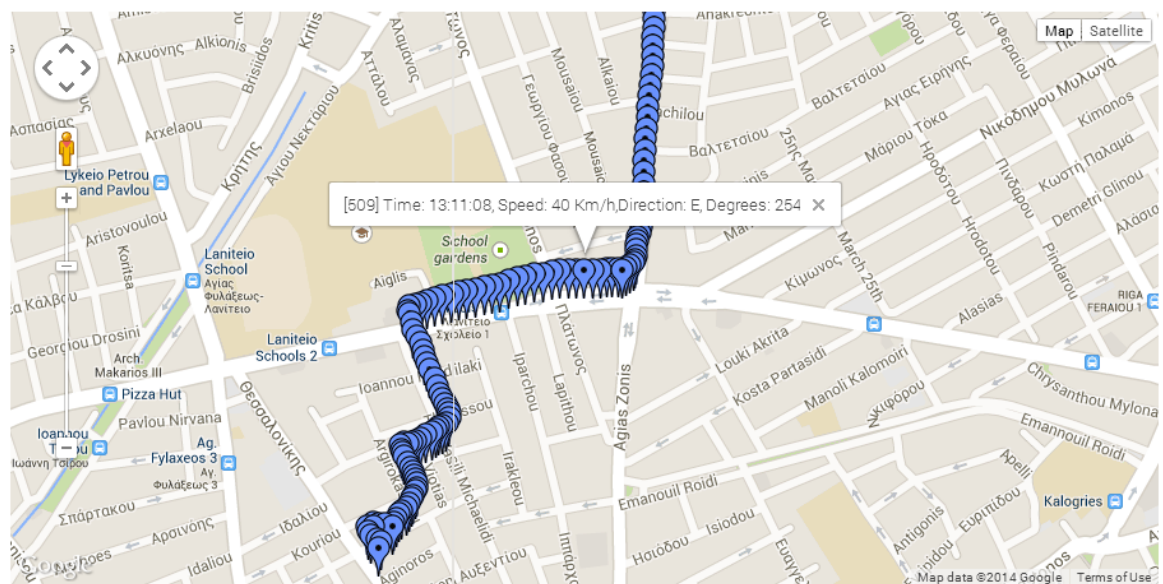
4.3.2 Ιστότοπος συστήματος

Στο πλαίσιο της παρούσας έρευνας, αναπτύξαμε ένα ιστότοπο (σύνολο από ιστοσελίδες) με σκοπό την παροχή πληροφοριών στον χρήστη για τις διαδρομές που έχει καταγράψει χρησιμοποιώντας Google Maps (API v3) (βλέπε Εικόνα 27). Επιπλέον, ο ιστότοπος παρέχει κατευθυντήριες οδηγίες για την εγκατάσταση και χρήση της εφαρμογής με σκοπό τη διευκόλυνση του χρήστη. Επίσης, μέσω του Ιστοτόπου παρέχουμε τη δυνατότητα στον χρήστη να εγγραφεί στο σύστημα, να λάβει (download) την Android εφαρμογή όπως επίσης και τον χάρτη της Κύπρου. Να σημειωθεί πως ο χάρτης δεν είναι απαραίτητος στην καταγραφή δεδομένων, αφού η εφαρμογή έχει τη δυνατότητα να λειτουργήσει χωρίς τον χάρτη. Ο λόγος για αυτή την ευελιξία του συστήματος αφορά αποκλειστικά τον μεγάλο όγκο του χάρτη. Έτσι, οι χρήστες οι οποίοι δεν έχουν αρκετό αποθηκευτικό χώρο στην κινητή συσκευή τους, έχουν την επιλογή να μην χρησιμοποιήσουν τον χάρτη.

Επιπλέον, ο ιστότοπος, αξιοποιήθηκε στη διαδικασία κατασκευής του συνόλου των δεδομένων που προορίζονταν για ανάλυση και χρήση στη διαδικασία του συστήματος για την αυτόματη αναγνώριση κόμβων. Έτσι, προχωρήσαμε στη δημιουργία μιας εφαρμογής η οποία μας έδωσε την δυνατότητα να υποδείξουμε στο σύστημα σε ποια σημεία των δρομολογίων παρουσιάζονταν κόμβοι (βλέπε Εικόνα 28). Αποθηκεύσαμε τα δεδομένα αυτά σε ξεχωριστό πίνακα στη ΒΔ με σκοπό να μην επηρεαστούν τα καταγραμμένα δεδομένα από τους χρήστες. Το σύνολο δεδομένων που παράχθηκε διαμέσου αυτής της διαδικασίας αποτέλεσε το σύνολο δεδομένων εκπαίδευσης (training set). Οι ανάγκες λειτουργίας του ιστοτόπου ανάγκασαν τη δημιουργία επιπέδων δικαιωμάτων για τους χρήστες. Έτσι, το δικαίωμα πρόσβασης στην λειτουργία υπόδειξης κόμβων στο σύστημα έχουν μόνο χρήστες οι οποίοι έχουν την ιδιότητα διαχειριστή (admin).

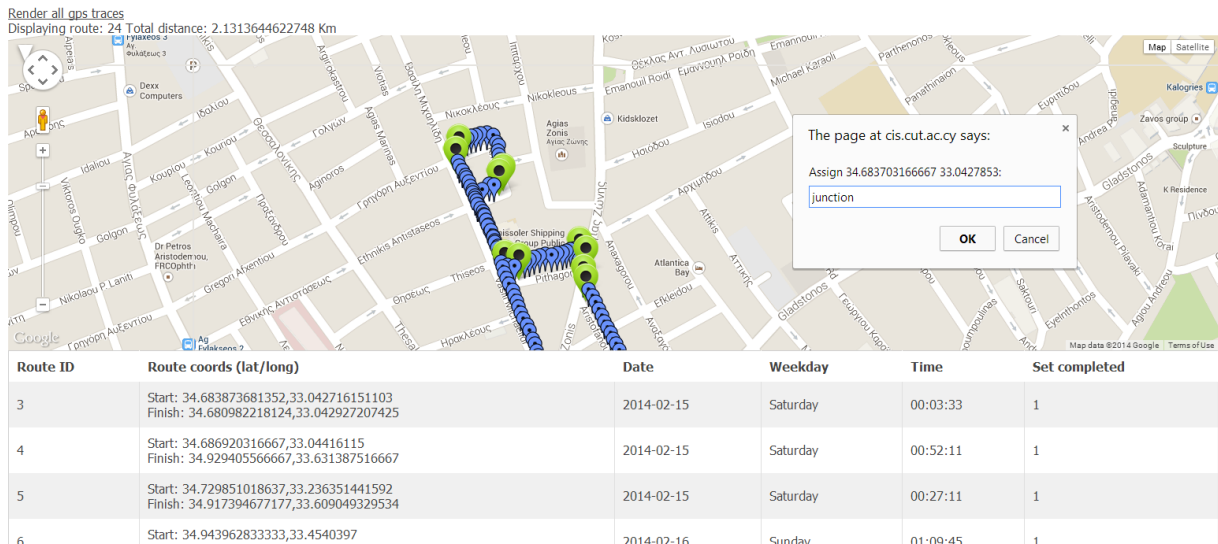
My Routes

Route ID	Route coords (lat/long)	Date	Weekday	Time
55	Start: 34.92663215,33.440269333333 Finish: 34.883488383333,33.428660733333	2014-03-22	Tuesday	01:14:06
56	Start: 34.944737633333,33.45969345 Finish: 34.949542666667,33.571358916667	2014-03-22	Tuesday	01:31:59
57	Start: 34.868343333333,33.5854205 Finish: 34.9290141,33.631721316667	2014-03-22	Tuesday	01:28:28



Εικόνα 27: Διαδρομές που έχει καταγράψει ο χρήστης από τον ιστότοπο του συστήματος

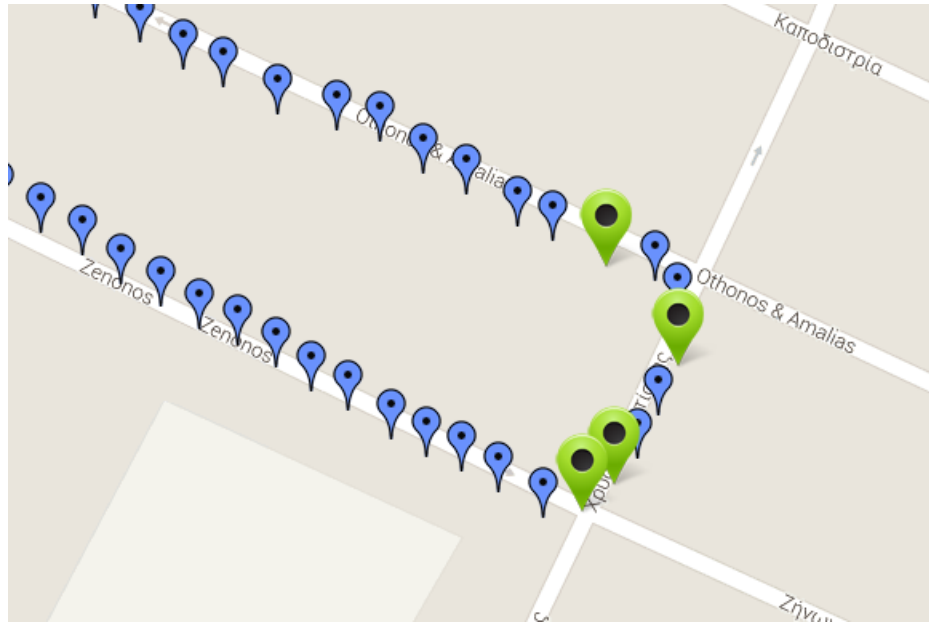
Training set data - Total nodes: **16607** Total junction points: **304** Total routes: **23**



Εικόνα 28: Υπόδειξη κόμβων στο σύστημα σε όλα τα δρομολόγια που έχουν καταγραφεί

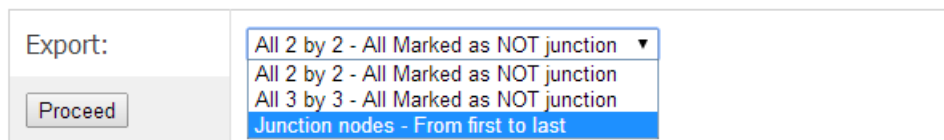
4.4 Διαδικασία εξαγωγής δεδομένων σε μορφή αρχείου WEKA (ARFF)

Η διαδικασία εξαγωγής δεδομένων είχε σημείο αναφοράς τη δημιουργία ενός συνόλου δεδομένων εκπαίδευσης, όπως προαναφέρθηκε στην υποενότητα «4.3.2 Ιστότοπος συστήματος». Κατά τη διαδικασία υπόδειξης κόμβων, σε κάθε σημείο κόμβου υποδείξαμε το σημείο αρχής και το σημείο τέλους του κόμβου. Παράλληλα, σε πολλές περιπτώσεις, ανάμεσα στα σημεία αρχής και τέλους του κόμβου παρουσιάζονταν ενδιάμεσα σημεία (βλέπε Εικόνα 29). Το επόμενο βήμα ήταν η διαδικασία που ακολουθήσαμε για να εξάγουμε τα δεδομένα. Μέσω του Ιστοτόπου, δημιουργήσαμε μια λειτουργία γραμμένη σε γλώσσα προγραμματισμού PHP η οποία εξάγει σε μορφή ARFF (Attribute-Relation File Format) τα δεδομένα που χρησιμοποιήθηκαν για την εκπαίδευση του συστήματος (βλέπε Εικόνα 30). Η λειτουργία αυτή, μας επέτρεψε να εμπλουτίσουμε τα δεδομένα με τη διαφορά των σημείων στα χαρακτηριστικά απόστασης, χρόνου, ταχύτητας και μοιρών κατεύθυνσης. Οι διαφορές προκύπτουν από την εξαγωγή των σημείων είτε ανά ζεύγη είτε ανά τριάδες. Επιπλέον, τροποποιήσαμε την εξαγωγή έτσι ώστε να λαμβάνονται υπόψη μόνο οι διαφορές των σημείων κόμβων χωρίς περιορισμό στον αριθμό σημείων. Το αποτέλεσμα ήταν η δημιουργία τριών αρχείων ARFF.



Εικόνα 29: Ενδιάμεσα σημεία κόμβων (τα σημεία αρχής και τέλους κόμβου αναπαριστούν τα πράσινα σημεία)

Export Data (WEKA format)



Εικόνα 30: Λειτουργία εξαγωγής συνόλου δεδομένων εκπαίδευσης

4.4.1 Δομή αρχείου ARFF

Τα αρχεία ARFF έχουν αναπτυχθεί αποκλειστικά για το πρόγραμμα μηχανικής μάθησης WEKA από τμήμα πληροφορικής του πανεπιστημίου Waikato. Ένα αρχείο ARFF χωρίζεται σε δύο τμήματα. Στο πρώτο τμήμα δηλώνεται το όνομα της σχέσης και η λίστα των ιδιοτήτων μαζί με τον τύπο δεδομένων, όπως στο παράδειγμα εξαγωγής δεδομένων εκπαίδευσης με απόσταση σημείων 3 (βλέπε Εικόνα 31). Παράλληλα, στις ιδιότητες ορίζεται η κλάση μαζί με τις διάφορες τιμές που μπορεί να πάρει. Στην περίπτωσή μας, η κλάση παίρνει τις τιμές Junction και Not-

Junction. Στο δεύτερο τμήμα ορίζονται όλα τα δεδομένα αντίστοιχα με τις ιδιότητες που έχουν ήδη οριστεί (Garner, 1995).

```
@RELATION routeData

@ATTRIBUTE ids STRING
@ATTRIBUTE distance NUMERIC
@ATTRIBUTE time NUMERIC
@ATTRIBUTE velocity NUMERIC
@ATTRIBUTE degrees NUMERIC
@ATTRIBUTE class {Junction,Not-Junction}

@DATA
487-484,17,00,2,159,Not-Junction
490-487,14,00,4,130,Not-Junction
493-490,21,00,11,0,Not-Junction
496-493,23,00,5,3,Not-Junction
499-496,25,00,14,0,Not-Junction
502-499,30,00,7,46,Not-Junction
505-502,30,00,2,0,Not-Junction
508-505,27,00,3,18,Not-Junction
511-508,24,00,5,48,Not-Junction
514-511,19,00,21,8,Not-Junction
517-514,18,00,12,4,Not-Junction
520-517,25,00,16,0,Not-Junction
```

Εικόνα 31: Αρχείο ARFF από το σύνολο δεδομένων εκπαίδευσης απόσταση σημείων 3

5. Ανάλυση δεδομένων

5.1 Προεπεξεργασία δεδομένων

Μέσω της λειτουργίας εξαγωγής συνόλου δεδομένων εκπαίδευσης δημιουργήσαμε τρία αρχεία ARFF, το κάθε ένα με διαφορετικό περιεχόμενο. Το πρώτο αρχείο περιείχε ένα σύνολο δεδομένων με τιμές που μετρούσαν την διαφορά δύο σημείων που δεν είχαν αναγνωριστεί ως κόμβοι (βλέπε Εικόνα 32). Στο δεύτερο αρχείο υπήρχε το σύνολο δεδομένων με τιμές που μετρούσαν την διαφορά τριών σημείων που δεν είχαν αναγνωριστεί ως κόμβοι (βλέπε Εικόνα 33) και το τρίτο αρχείο περιείχε τις τιμές από την διαφορά του σημείου αρχής μέχρι το σημείο τέλους του κόμβου (βλέπε Εικόνα 34). Στα στοιχεία τα οποία περιέχονται στα πρώτα δύο αρχεία είχαν οριστεί ως Not-Junction ενώ στο τρίτο αρχείο ως Junction. Παρόλα αυτά, στην περίπτωση που τα σημεία είχαν απόσταση δύο (δηλαδή γειτονικά σημεία), οι διαφορές στις τιμές ήταν ελάχιστες σε σχέση με τις τιμές στην περίπτωση του συνόλου δεδομένων κόμβων. Αυτό οφείλεται στην διαδικασία καταχώρησης κόμβων που τα σημεία αρχής και τέλος των κόμβων περιείχαν σε πολλές περιπτώσεις δύο και τρία σημεία ενδιάμεσα. Έτσι, επικεντρωθήκαμε στο αρχείο με τις τιμές από την απόσταση τριών σημείων.

Το επόμενο βήμα ήταν η συγχώνευση των δύο εναπομείναντων αρχείων σε ένα, έτσι ώστε το σύνολο δεδομένων να περιέχει διαφορές σημείων κόμβων και μη κόμβων. Τα δεδομένα πάρθηκαν συνολικά από 23 διαδρομές με 16607 σημεία θέσεων, από τα οποία τα 304 χαρακτηρίστηκαν ως κόμβοι.

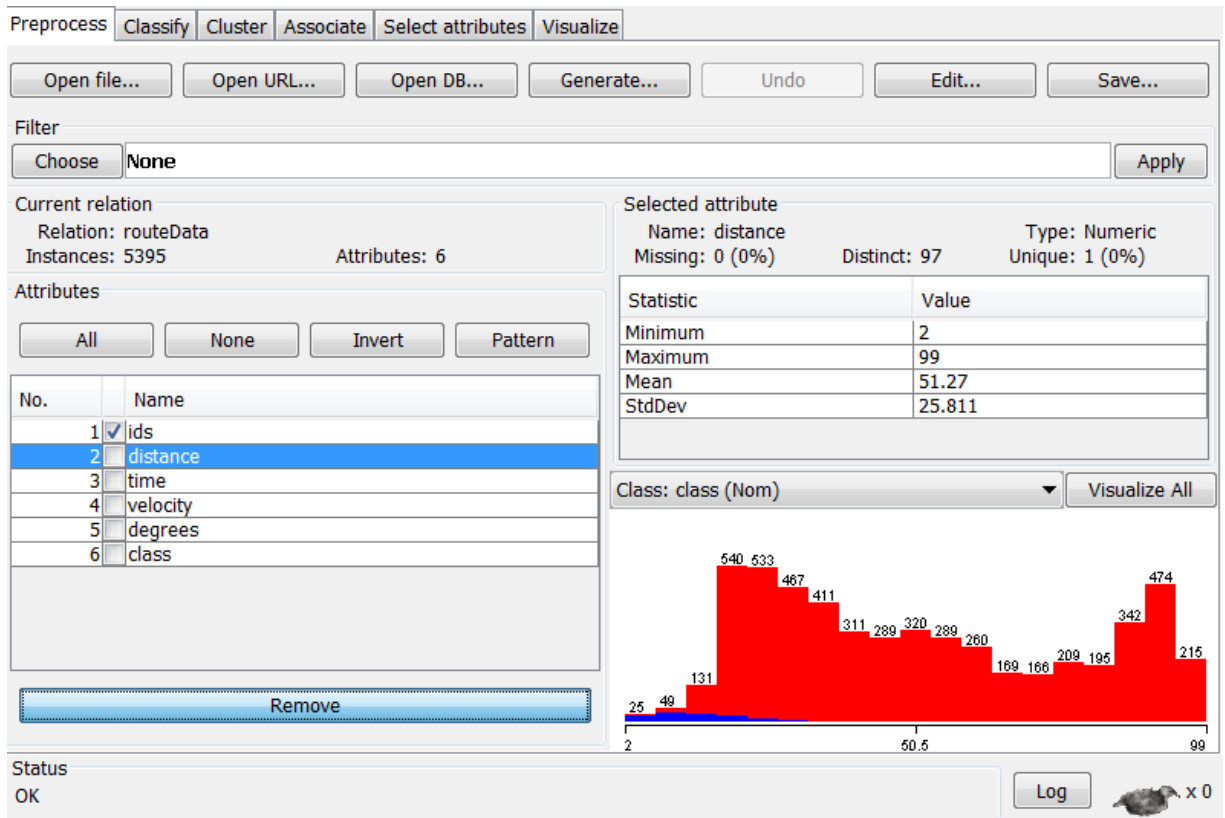
Στη συνέχεια, εισαγάγαμε το νέο αρχείο ARFF στο πρόγραμμα WEKA έκδοσης 3.7 (βλέπε Εικόνα 35). Ακολούθως διαγράψαμε την ιδιότητα `ids` αφού η τιμή αυτή δεν αφορά δεδομένα κίνησης και για τον λόγο αυτό δεν λήφθηκε υπόψη στο στάδιο της ταξινόμησης. Επιπλέον, αφαιρέθηκε και η ιδιότητα `χρόνος` επειδή όλοι οι κόμβοι βρίσκονται σε σημεία πόλεων, με αποτέλεσμα η ταχύτητα να είναι χαμηλή και η συνεισφορά της μεταβλητής αυτής αμελητέα. Έτσι η καταγραφή σημείων πραγματοποιείται σε τακτά χρονικά διαστήματα αφήνοντας στις περισσότερες περιπτώσεις διαφορά χρόνου 0.

<pre> @RELATION routeData @ATTRIBUTE ids STRING @ATTRIBUTE distance NUMERIC @ATTRIBUTE time NUMERIC @ATTRIBUTE velocity NUMERIC @ATTRIBUTE degrees NUMERIC @ATTRIBUTE class {Junction,Not-Junction} @DATA 485-486,7,00,1,27,Not-Junction 487-488,6,00,0,4,Not-Junction 489-490,5,00,3,12,Not-Junction 491-492,6,00,2,0,Not-Junction 493-494,8,00,4,0,Not-Junction 495-496,8,00,11,0,Junction 497-498,7,00,6,0,Not-Junction 499-500,9,00,4,47,Not-Junction 501-502,10,00,1,0,Not-Junction 503-504,10,00,0,0,Not-Junction 505-506,9,00,1,0,Not-Junction 507-508,9,00,0,0,Not-Junction 509-510,8,00,1,0,Not-Junction 511-512,6,00,5,0,Not-Junction 513-514,5,00,11,3,Not-Junction 515-516,7,00,5,0,Not-Junction 517-518,6,00,2,0,Not-Junction 519-520,13,00,13,0,Not-Junction 521-522,10,00,2,0,Not-Junction 523-524,14,00,2,8,Not-Junction 525-526,10,00,2,0,Not-Junction 527-528,5,00,24,0,Not-Junction 529-530,9,00,9,12,Not-Junction 531-532,9,00,1,0,Not-Junction 533-534,10,00,5,4,Not-Junction 535-536,9,00,2,0,Not-Junction 537-538,6,00,6,0,Not-Junction 539-540,6,00,10,0,Junction 541-542,8,00,4,0,Junction 543-544,6,00,3,0,Not-Junction 545-546,10,00,5,0,Not-Junction 547-548,14,00,4,0,Not-Junction 549-550,21,00,8,0,Not-Junction 551-552,18,00,6,0,Not-Junction : : : </pre>	<pre> @RELATION routeData @ATTRIBUTE ids STRING @ATTRIBUTE distance NUMERIC @ATTRIBUTE time NUMERIC @ATTRIBUTE velocity NUMERIC @ATTRIBUTE degrees NUMERIC @ATTRIBUTE class {Junction,Not-Junction} @DATA 487-484,17,00,2,159,Not-Junction 490-487,14,00,4,130,Not-Junction 493-490,21,00,11,0,Not-Junction 496-493,23,00,5,3,Not-Junction 499-496,25,00,14,0,Not-Junction 502-499,30,00,7,46,Not-Junction 505-502,30,00,2,0,Not-Junction 508-505,27,00,3,18,Not-Junction 511-508,24,00,5,48,Not-Junction 514-511,19,00,21,8,Not-Junction 517-514,18,00,12,4,Not-Junction 520-517,25,00,16,0,Not-Junction 523-520,36,00,8,0,Not-Junction 526-523,35,00,5,8,Not-Junction 529-526,22,00,19,18,Not-Junction 532-529,26,00,13,12,Not-Junction 535-532,29,00,6,7,Not-Junction 538-535,24,00,12,1,Not-Junction 541-538,16,00,13,69,Not-Junction 544-541,19,00,8,0,Not-Junction 547-544,31,00,17,0,Not-Junction 550-547,54,00,26,0,Not-Junction 553-550,57,00,6,0,Not-Junction 556-553,42,00,10,30,Not-Junction 559-556,37,00,8,4,Not-Junction 562-559,31,00,2,6,Not-Junction 565-562,31,00,1,7,Not-Junction 568-565,31,00,1,0,Not-Junction 571-568,33,00,2,0,Not-Junction 574-571,26,00,11,2,Not-Junction 577-574,21,00,13,3,Not-Junction 580-577,21,00,7,5,Not-Junction 583-580,21,00,3,0,Not-Junction : : : </pre>	<pre> @RELATION routeData @ATTRIBUTE ids STRING @ATTRIBUTE distance NUMERIC @ATTRIBUTE time NUMERIC @ATTRIBUTE velocity NUMERIC @ATTRIBUTE degrees NUMERIC @ATTRIBUTE class {Junction,Not-Junction} @DATA 495-496,8,00,11,0,Junction 540-541,5,00,1,69,Junction 3122-3123,5,00,0,23,Junction 3169-3170,8,00,3,16,Junction 3272-3273,7,00,2,169,Junction 3291-3293,11,00,3,64,Junction 3306-3307,7,00,4,45,Junction 5866-5867,8,00,7,39,Junction 5908-5909,10,00,2,12,Junction 5990-5994,24,00,7,7,Junction 6018-6022,21,00,3,16,Junction 6047-6048,8,00,7,0,Junction 6079-6084,30,00,4,20,Junction 6103-6104,6,00,1,31,Junction 6116-6118,14,00,10,116,Junction 6369-6371,12,00,6,70,Junction 6406-6409,21,00,14,79,Junction 6453-6456,21,00,13,79,Junction 6519-6520,8,00,3,20,Junction 6853-6856,18,00,13,166,Junction 6892-6893,6,00,1,73,Junction 6910-6912,17,00,3,53,Junction 6933-6935,10,00,8,10,Junction 6948-6949,7,00,12,25,Junction 6978-6981,20,00,5,95,Junction 6984-6986,18,00,1,76,Junction 7000-7001,7,00,10,59,Junction 7023-7025,14,00,0,58,Junction 7045-7046,6,00,0,97,Junction 7095-7097,17,00,2,138,Junction 7123-7126,18,00,13,166,Junction 7162-7163,6,00,1,73,Junction 7180-7182,17,00,3,53,Junction 7202-7205,18,00,3,10,Junction : : : </pre>
---	---	--

Εικόνα 32: Διαφορά σημείων με απόσταση 2 που δεν καταχωρήθηκαν ως κόμβοι

Εικόνα 33: Διαφορά σημείων με απόσταση 3 που δεν καταχωρήθηκαν ως κόμβοι

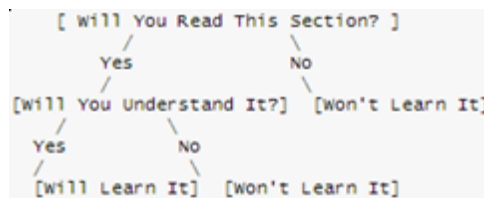
Εικόνα 34: Διαφορά σημείων με απόσταση του σημείου αρχής κόμβου μέχρι το σημείο τέλους μόνο από σημεία κόμβων



Εικόνα 35: Εισαγωγή δεδομένων στο πρόγραμμα WEKA - Διαγραφή ιδιότητας ids

5.2 Ταξινόμηση δεδομένων (classification)

Με σκοπό την πρόβλεψη της κλάσης στο σύνολο δεδομένων, αξιοποιήθηκε η μέθοδος της ταξινόμησης. Η ταξινόμηση, αλλιώς δέντρο ταξινόμησης, είναι ένας αλγόριθμος εξόρυξης δεδομένων ο οποίος δημιουργεί βήμα προς βήμα το αποτέλεσμα μιας νέας πρόβλεψης (βλέπε Εικόνα 36).



Εικόνα 36: Παράδειγμα αλγορίθμου ταξινόμησης

Στην περίπτωση της παρούσας έρευνας, αξιοποιήσαμε τη μέθοδο ταξινόμησης με στόχο την πρόβλεψη κόμβου με βάση τις αριθμητικές μεταβολές στις τιμές απόστασης, χρόνου, ταχύτητας και γωνίας κατεύθυνσης του οχήματος. Για τον σκοπό αυτό αξιολογήσαμε το σύνολο δεδομένων με τους ταξινομητές IBk (k - κοντινότεροι γείτονες), Multilayer Perceptron (νευρωνικά δίκτυα) και J48 (δέντρο απόφασης) με «cross validation» 10. Η επιλογή cross validation σε 10, σημαίνει ότι ολόκληρο το σύνολο δεδομένων χωρίζεται σε 10 υποσύνολα δεδομένων ίδιου μεγέθους. Τα 9 από τα 10 υποσύνολα θα συνδυαστούν και θα χρησιμοποιηθούν για εκπαίδευση ενώ το 1 για δοκιμή. Η διαδικασία αυτή επαναλαμβάνεται μέχρι όλα τα 10 μικρά σύνολα δεδομένων χρησιμοποιηθούν για δοκιμή. Τέλος, τα αποτελέσματα που παράγονται αποτελούν το μέσο όρο όλων των επιμέρους αποτελεσμάτων. Στους παρακάτω πίνακες θα αναφερθούν τα ποσοστά ορθότητας αλλά και τα ποσοστά λανθασμένων προβλέψεων. Επιπλέον, θα αναφερθούν οι τιμές ακρίβειας (precision) και οι τιμές ανάκλησης (recall). Οι τιμές ακρίβειας, αναφέρονται στο ποσοστό των δεδομένων που ανήκουν σε κάποια από τις δύο κλάσεις (Junction, Not-Junction) το οποίο διαιρέθηκε από το σύνολο των δεδομένων που έχουν προβλεφθεί ως την αντίστοιχη κλάση. Οι τιμές ανάκλησης, αφορούν το ποσοστό των δεδομένων που έχουν προβλεφθεί ως κάποια από τις δύο κλάσεις το οποίο διαιρέθηκε με τον αριθμό των δεδομένων που ανήκουν στην κλάση η οποία έχει προβλεφθεί.

Υπολογισμός Ακρίβειας (precision):

$$\text{Ακρίβεια}_{\text{κόμβου}} = \frac{|\text{Σύνολο κόμβων} \cap \text{Σύνολο προβλέψιμων κόμβων}|}{|\text{Σύνολο προβλέψιμων κόμβων}|}$$

$$\text{Ακρίβεια}_{\text{μη-κόμβου}} = \frac{|\text{Σύνολο μη κόμβων} \cap \text{Σύνολο προβλέψιμων μη κόμβων}|}{|\text{Σύνολο προβλέψιμων μη κόμβων}|}$$

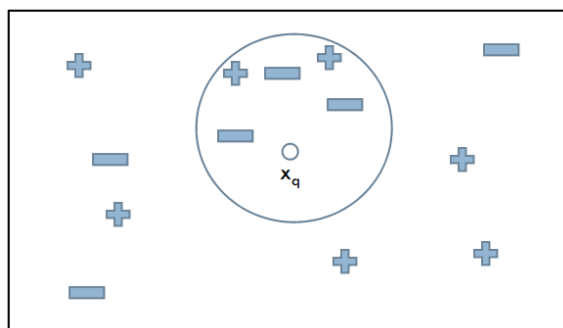
Υπολογισμός Ανάκλησης (recall):

$$\text{Ανάκληση}_{\text{κόμβου}} = \frac{|\text{Σύνολο κόμβων} \cap \text{Σύνολο προβλέψιμων κόμβων}|}{|\text{Σύνολο κόμβων}|}$$

$$\text{Ανάκληση}_{\text{μη-κόμβου}} = \frac{|\text{Σύνολο μη κόμβων} \cap \text{Σύνολο προβλέψιμων μη κόμβων}|}{|\text{Σύνολο μη κόμβων}|}$$

5.2.1 IBk (k - κοντινότεροι γείτονες)

Ο αλγόριθμος IBk ανήκει στους «οκνηρούς» ταξινομητές. Οκνηροί ταξινομητές, ορίζονται οι ταξινομητές οι οποίοι αποθηκεύουν τα δεδομένα εκπαίδευσης και πραγματοποιούν την ταξινόμηση την ώρα που διατίθεται το σύνολο δεδομένων για πρόβλεψη. Το χαρακτηριστικό τους είναι ο μικρός χρόνος εκπαίδευσης αλλά με μεγάλο χρόνο ταξινόμησης. Όλο το σύνολο δεδομένων για πρόβλεψη απεικονίζεται σε σημεία στον n-διάστατο χώρο και οι κοντινότεροι γείτονες ορίζονται συνήθως με βάση την ευκλείδεια απόσταση. Σημαντική παράμετρος του IBk είναι το k, όπου αναφέρεται στον αριθμό των κοντινότερων σημείων (βλέπε Εικόνα 37). Αν το k είναι πολύ μικρό τότε η ταξινόμηση παρουσιάζει ευαισθησία στον θόρυβο. Αν το k είναι πολύ μεγάλο η «γειτονιά» μπορεί να περιλαμβάνει δεδομένα άλλης τάξης. Σύμφωνα με τα αποτελέσματα του πίνακα με τα αποτελέσματα εκπαίδευσης δεδομένων IBk (βλέπε Πίνακα 2), με το k να ισούται με 3, πρόβλεψε σωστά το 98.27% από το σύνολο δεδομένων.



Εικόνα 37: Παράδειγμα k - κοντινότερων γειτόνων με k = 5. Το νέο στοιχείο x_q προβλέπεται αρνητικό

5.2.3 J48 (δέντρο απόφασης)

Η διαδικασία του δέντρου απόφασης ξεκινά με όλα τα χαρακτηριστικά/ιδιότητες των δεδομένων να βρίσκονται στη ρίζα. Ακολούθως, το δέντρο αναπτύσσεται από πάνω προς τα κάτω όπου τα δεδομένα διαχωρίζονται αναδρομικά βάσει των τιμών ενός επιλεγμένου χαρακτηριστικού. Το χαρακτηριστικό επιλέγεται βάσει μιας ευριστικής ή στατιστικής μέτρησης (π.χ. information gain). Το βήμα αυτό επαναλαμβάνεται μέχρι όλα τα στιγμιότυπα ενός κόμβου να ανήκουν στην ίδια τάξη ή όταν δεν υπάρχουν άλλα χαρακτηριστικά για περαιτέρω διαμοιρασμό. Ο αλγόριθμος J48, έδειξε σωστές προβλέψεις με 98.66% επιτυχία. Στην περίπτωση μας, σε σχέση με τους αλγόριθμους IBk και Multilayer Perceptron κατέχει το υψηλότερο ποσοστό ορθότητας.

Σωστές προβλέψεις **Λανθασμένες προβλέψεις** **Ακρίβεια (Precision)** **Ανάκληση (Recall)**

5323	72	Junction: 0.790	Junction: 0.538
98.6654 %	1.3346 %	Not-Junction: 0.990	Not-Junction: 0.997

Χρόνος ταξινόμησης: 0.02 sec

Πίνακας 4: Αποτέλεσμα εκπαίδευσης αλγορίθμου J48

5.3 Εξαγωγή κανόνων με τον αλγόριθμο J48

Σύμφωνα με τα αποτελέσματα των πιο πάνω αλγορίθμων, ο J48 κατείχε το υψηλότερο ποσοστό ορθότητας με 98.66%. Από την διαδικασία ταξινόμησης των δεδομένων με τον αλγόριθμο J48, έχουν παραχθεί κάποιοι κανόνες οι οποίοι με βάση τα χαρακτηριστικά καθορίζουν αν κάποια εγγραφή δεδομένων θα καθοριστεί ως κόμβος (βλέπε Εικόνα 38). Μια εγγραφή δεδομένων αποτελείται από την διαφορά των σημείων στις τιμές απόστασης, ταχύτητας και γωνίας κατεύθυνσης. Η διαφορά χρόνου έχει αφαιρεθεί στο στάδιο της προεπεξεργασίας δεδομένων.

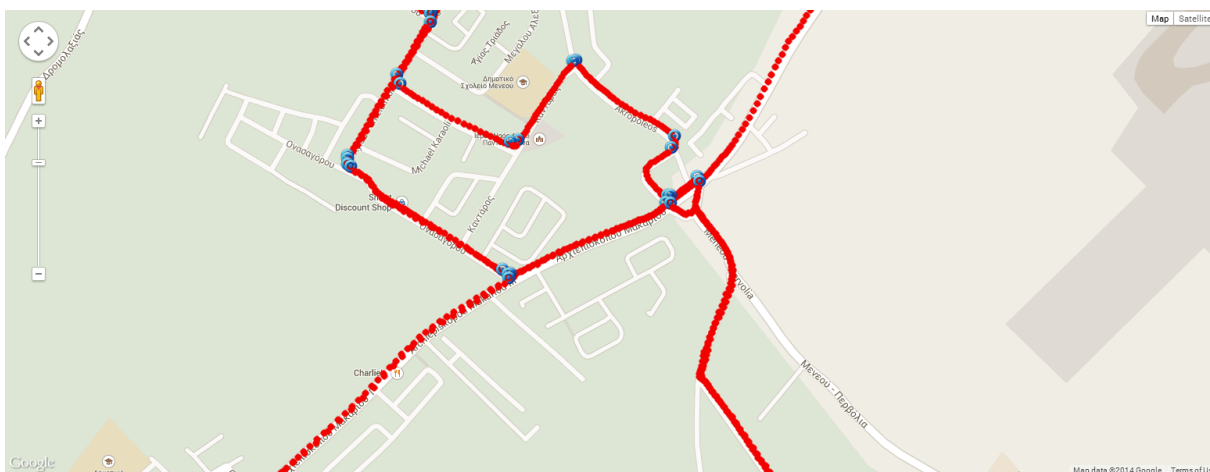


Εικόνα 38: Κανόνες που προέκυψαν από την ταξινόμηση με τον αλγόριθμο J48

Με βάση τους κανόνες στην Εικόνα 38, φαίνεται πως η απόσταση μεταξύ των γεωγραφικών θέσεων κατέχει τον καθοριστικό παράγοντα. Έτσι, αν η απόσταση μεταξύ των σημείων είναι πάνω από 15 μέτρα τότε τα σημεία δεν θεωρούνται κόμβος. Αν η διαφορά απόστασης βρίσκεται στα 15 μέτρα και κάτω, τότε η διαδικασία αναγνώρισης προχωρεί στον επόμενο κόμβο του δέντρου όπου αναφέρεται στην διαφορά των σημείων στην γωνία κατεύθυνσης. Αν η διαφορά γωνίας είναι μικρότερη ή ίση με 7 μοίρες τότε η πρόβλεψη μεταβαίνει στην διακλάδωση της απόστασης. Αν η διαφορά απόστασης είναι μεγαλύτερη από 9 μέτρα τότε τα σημεία δεν θεωρούνται κόμβος. Αν είναι μικρότερη ή ίση με 9 μέτρα, τότε θεωρείται κόμβος. Το δέντρο διακλαδώνεται μέχρι να καταλήξει σε κάποια από τις δύο κλάσεις (φύλα του δέντρου απόφασης). Οι αριθμοί στις παρενθέσεις αντιπροσωπεύουν τα δεδομένα που αναγνωρίστηκαν σωστά. Ο δεύτερος αριθμός, όπου υπάρχει, αναφέρεται στα δεδομένα τα οποία έχουν

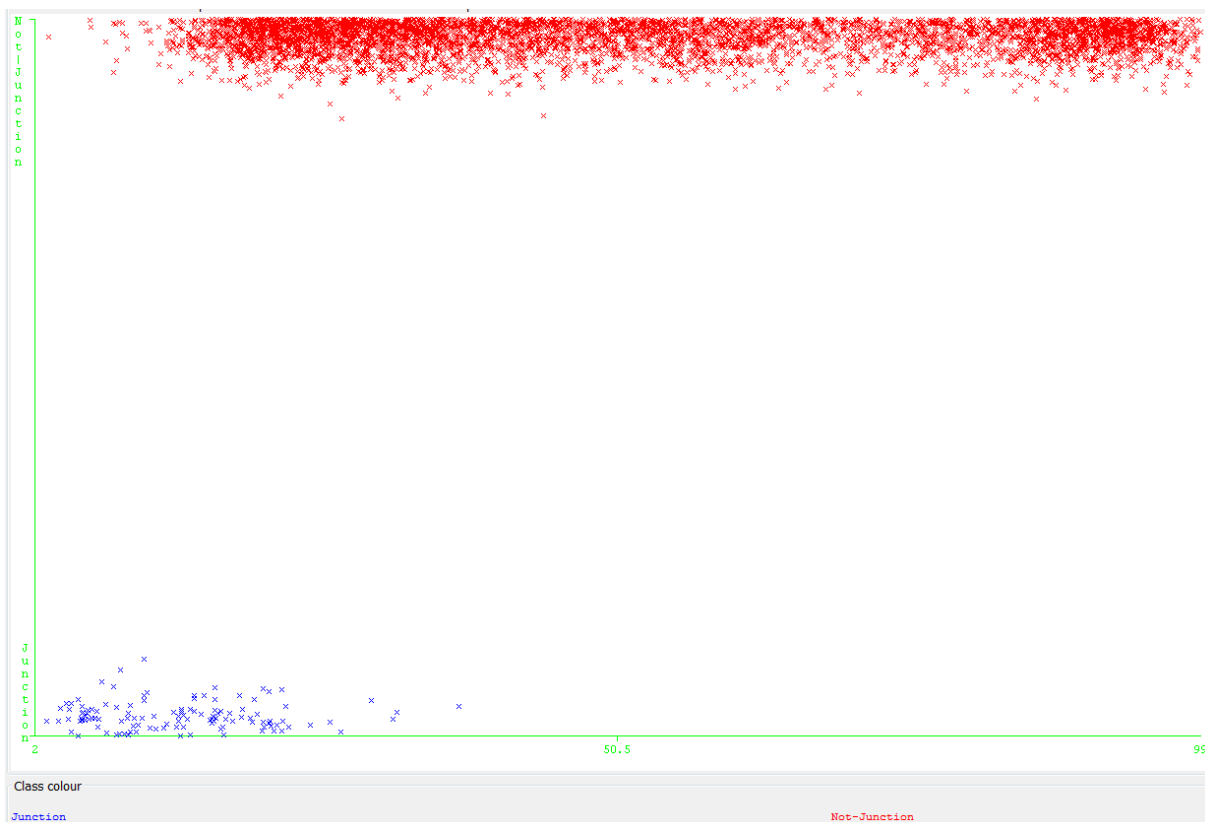
αναγνωριστεί λανθασμένα. Παράλληλα, για κάθε πρόβλεψη ξεχωριστά, προκύπτει ένα ποσοστό που επαληθεύει την ορθότητα της πρόβλεψης. Έτσι, μπορούμε να καθορίσουμε ένα όριο στο ποσοστό αυτό για καλύτερη ακρίβεια στις προβλέψεις κόμβων. Για παράδειγμα, αν το ποσοστό ορθότητας πρόβλεψης είναι τουλάχιστον 95% και η ίδια πρόβλεψη έχει επαληθευθεί από άλλο χρήστη, τότε θα υποθέσουμε το σημείο αυτό ως κόμβο.

Ο αλγόριθμος J48 δεν αναγνωρίζει σαν κόμβους τα σημεία που έχουν απόσταση μεταξύ τους μεγαλύτερη από 15 μέτρα. Το γεγονός αυτό οφείλεται στους κανόνες καταγραφής της εφαρμογής στην υποενότητα «4.2.3.1 Κανόνες στη διαδικασία καταγραφής», στην οποία αναφέραμε ότι η συχνότητα καταγραφής σημείων είναι ανάλογη της ταχύτητας κίνησης. Στην Εικόνα 39 η διαφορά ταχύτητας των οχημάτων διακρίνεται ξεκάθαρα. Όταν τα κόκκινα σημεία βρίσκονται σε μεγαλύτερη απόσταση το ένα από το άλλο, τότε στο συγκεκριμένο δρόμο τα οχήματα είχαν μεγαλύτερη ταχύτητα. Στους δρόμους όπου τα σημεία φαίνονται πιο πυκνά, τότε τα οχήματα είχαν χαμηλότερη ταχύτητα και μικρότερη απόσταση μεταξύ τους.



Εικόνα 39: Ίχνη διαδρομών από την εφαρμογή σε απεικόνιση με Google Maps – API v3 (GPS traces)

5.4 Απεικονίσεις δεδομένων

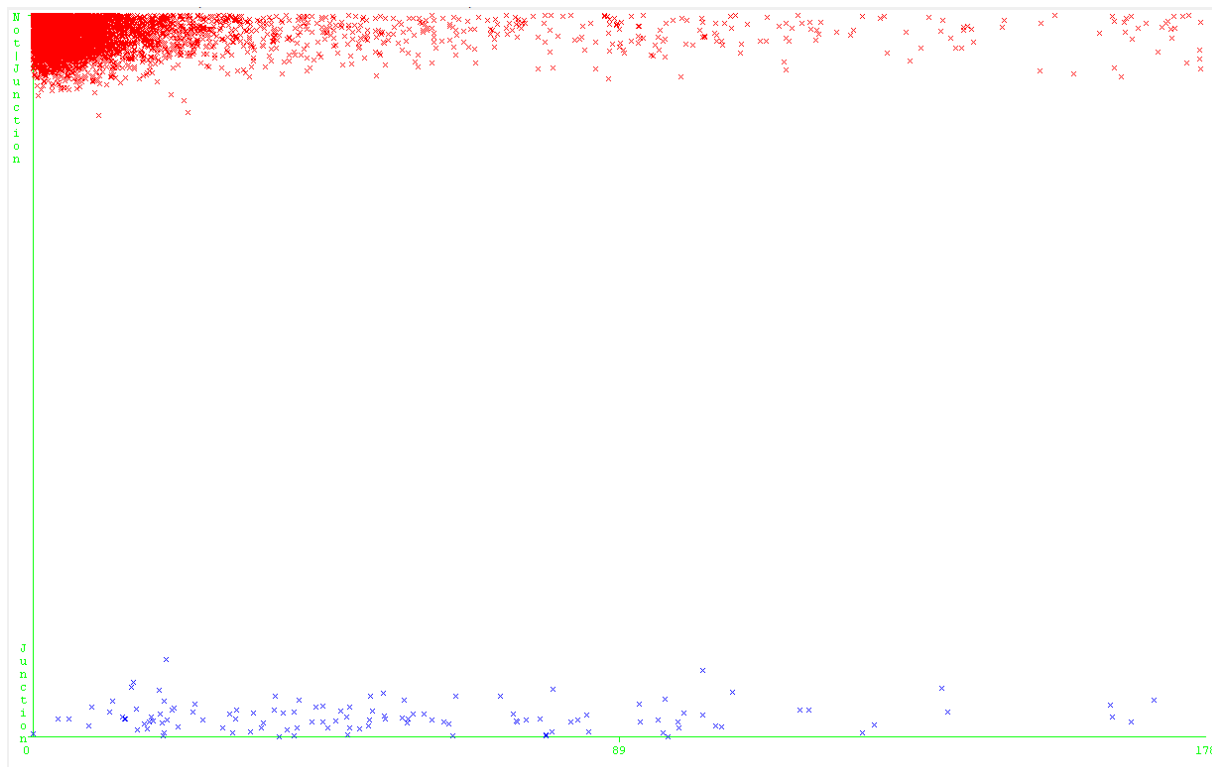


Εικόνα 40: Απεικόνιση αποτελεσμάτων - Άξονας Y: κλάση, Άξονας X: διαφορά απόστασης

Στην Εικόνα 40, απεικονίζονται όλα τα δεδομένα σε δύο άξονες. Στον άξονα Y βρίσκεται η κλάση και στον άξονα X η απόσταση. Τα σημεία που έχουν χρώμα μπλε αντιπροσωπεύουν τα σημεία κόμβων και είναι συγκεντρωμένα στις χαμηλές τιμές του άξονα X - διαφορές απόστασης. Παρατηρούμε επίσης, τα κόκκινα σημεία τα οποία δεν είναι κόμβοι, είναι διασκορπισμένα σχεδόν σε όλο το εύρος του άξονα X εκτός από το σημείο με τις τιμές χαμηλές τιμές του X. Μέσω αυτής της γραφικής παράστασης, μπορούμε να υποθέσουμε στις μικρές αποστάσεις μεταξύ των σημείων υπάρχει μεγαλύτερη πιθανότητα να παρουσιαστεί κόμβος.

Στην Εικόνα 41 αποτυπώνονται πάλι όλα τα δεδομένα, όπου αυτή τη φορά ο άξονας X αφορά την διαφορά γωνίας κατεύθυνσης και ο άξονας Y αναφέρεται στην κλάση. Σε αυτό το διάγραμμα παρατηρούμε την συγκέντρωση των περισσότερων σημείων (κόκκινων) που δεν είναι κόμβοι στις μικρές τιμές του άξονα X όπου η διαφορά στις μοίρες των σημείων είναι

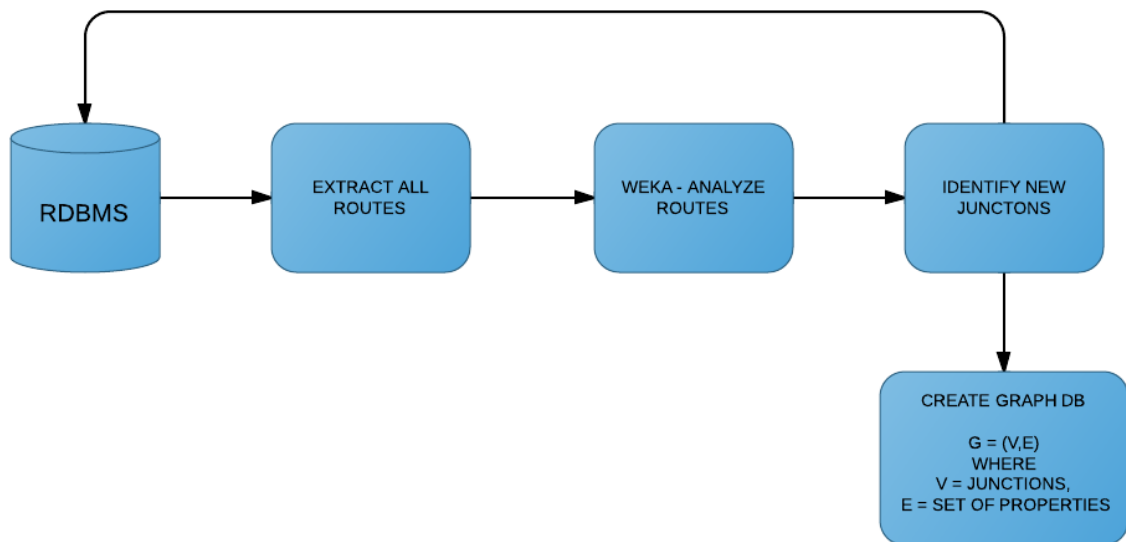
μικρή. Έτσι, μπορούμε να υποθέσουμε ξανά ότι σε μικρή διαφορά κλίσης των σημείων, υπάρχει μικρή πιθανότητα να παρουσιαστεί κόμβος.



Εικόνα 41: Απεικόνιση αποτελεσμάτων - Άξονας Y: κλάση, Άξονας X: διαφορά γωνίας κατεύθυνσης

6. Εφαρμογή ΒΔ γράφων

Ο σκοπός αξιοποίησης της ΒΔ γράφων αποσκοπεί στην περαιτέρω αξιοποίηση των δεδομένων αλλά και στην παροχή επιπρόσθετων πληροφοριών στους χρήστες του συστήματος. Στη δική μας περίπτωση για το σκοπό αυτό χρησιμοποιήσαμε τη neo4j, μια δωρεάν και ευρέως διαδεδομένη υλοποίηση ΒΔ γράφων. Η διαδικασία με την οποία προχωρήσαμε στην δημιουργία της νέας ΒΔ περιγράφεται αφαιρετικά στην Εικόνα 42. Αφού όλα τα δεδομένα των διαδρομών από την ΒΔ τύπου RDBMS αναλυθούν με τη βοήθεια μεθόδων μηχανικής μάθησης που παρέχει το πρόγραμμα WEKA, δημιουργήθηκε η ΒΔ γράφων.



Εικόνα 42: Διαδικασία δημιουργίας ΒΔ γράφων

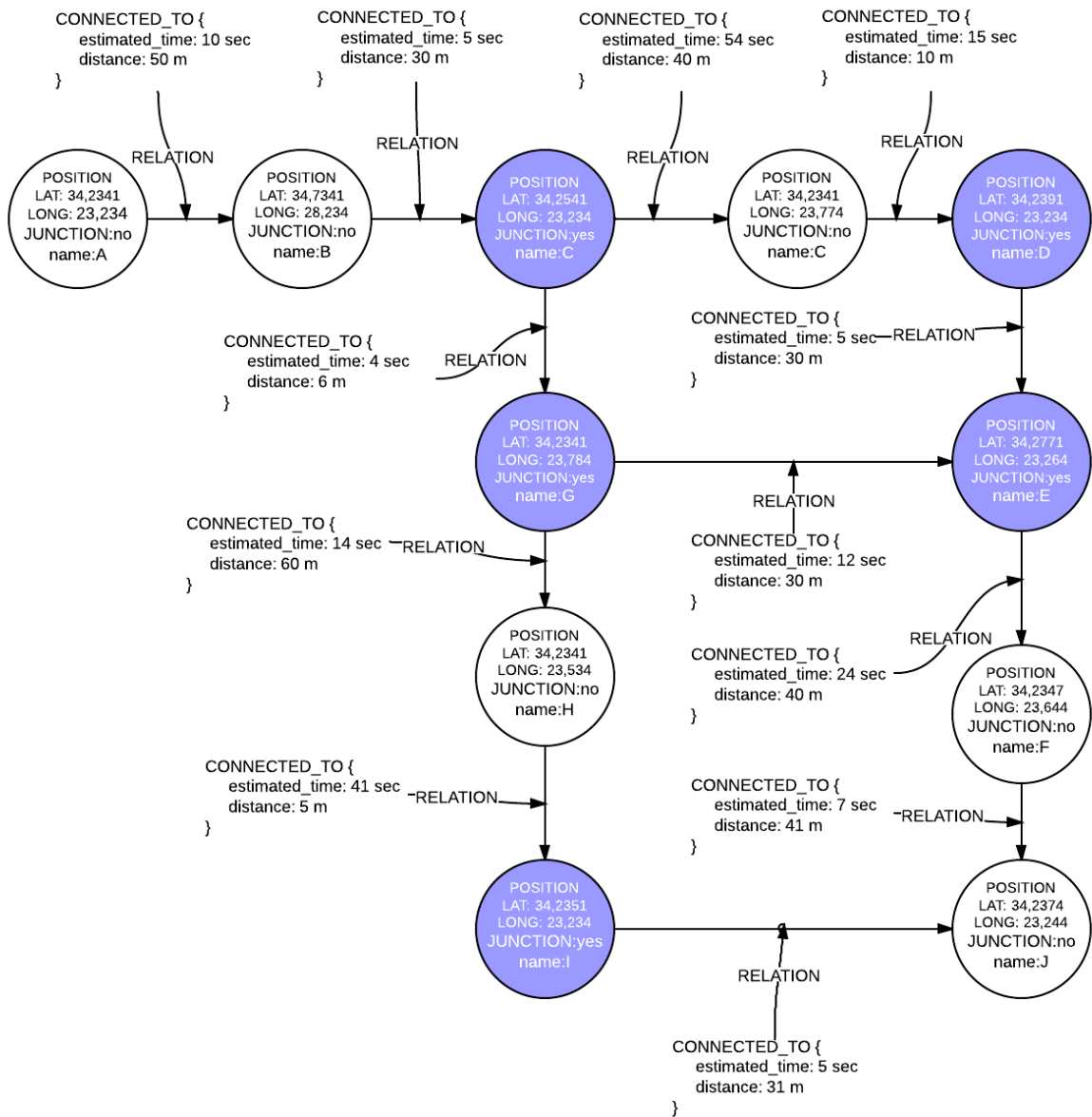
Μια απεικόνιση γράφου ιδιοτήτων ενός υποσυνόλου δεδομένων της RDBMS σε ΒΔ γράφων φαίνεται στην Εικόνα 43. Το γράφημα αποτελείται από κόμβους (nodes) που αντιστοιχούν σε γεωγραφικές θέσεις (Positions) με τα χαρακτηριστικά όνομα, κόμβος (ναι ή όχι), γεωγραφικό μήκος και πλάτος. Επιπλέον, παρουσιάζονται οι σχέσεις μεταξύ των κόμβων (nodes), οι ιδιότητες τους καθώς και οι ιδιότητες των σχέσεων τους. Για παράδειγμα, ο κόμβος με το όνομα POSITION με όνομα A έχει άμεση σχέση (είναι συνδεδεμένος) με τον κόμβο POSITION με

όνομα B. Το όνομα της σχέσης αυτών των δύο κόμβων ορίστηκε με το όνομα «CONNECTED_TO» έχοντας δύο χαρακτηριστικά. Τα χαρακτηριστικά της σχέσης ορίστηκαν με την μορφή «χαρακτηριστικό – τιμή» και αναφέρονται στην απόσταση των δύο γεωγραφικών θέσεων και τον αναμενόμενο χρόνο από το σημείο A στο σημείο B.

Για την δημιουργία της σχέσης που περιγράψαμε πιο πάνω χρησιμοποιήθηκε η γλώσσα Cypher της ΒΔ neo4j. Ένα μέρος των εντολών που παρουσιάζει τη δημιουργία των δύο πρώτων κόμβων του γραφήματος καθώς και της σχέσης τους (βλέπε Εικόνα 43) παρατίθεται πιο κάτω:

```
CREATE (POSITION {LAT: 34,2341, LONG: 23,234, JUNCTION: no, name: A}),  
      (POSITION {LAT: 34.7341, LONG: 28.234, JUNCTION: no, name: B}),  
      (POSITION (name='A')) -[:CONNECTED_TO {estimated_time:10,distance:50}] ->  
      (POSITION (name='B'))
```

Η αναπαράσταση των δεδομένων σε μορφή γράφων δίνει νέες δυνατότητες και διαστάσεις στην υποδομή που έχουμε δημιουργήσει καθώς διάφορες τεχνικές και αλγόριθμοι της θεωρίας γράφων μπορούν να εφαρμοστούν. Αυτές οι τεχνικές και αλγόριθμοι είναι ήδη ενσωματωμένοι στη ΒΔ γράφων neo4j, καθιστώντας τη χρήση του εξαιρετικά εύκολη. Ένα τέτοιο παράδειγμα είναι η εύρεση εναλλακτικών διαδρομών δεδομένης της έναρξης και του τέλους μιας διαδρομής με βάση συγκεκριμένα χαρακτηριστικά (π.χ. ελάχιστος χρόνος ή ελάχιστη απόσταση).



Εικόνα 43: Παράδειγμα μέρος του συνόλου δεδομένων στην ΒΔ γράφων – Γράφημα ιδιοτήτων (property graph)

Το ερώτημα πιο κάτω βρίσκει τη διαδρομή (από την θέση A μέχρι τη θέση J) με τον μικρότερο χρόνο με βάση το γράφημα ιδιοτήτων στην Εικόνα 43:

```

START      StartPosition = node: POSITION (name='A'),
           EndPosition = node: POSITION (name='J')

MATCH      paths = (StartPosition) - [:CONNECTED_TO*] -> (EndPosition)

RETURN     paths AS shortestPath,
           reduce(time=0, path in relationships(paths) : estimated_time + path.estimated_time) AS
           totalTime
           ORDER BY totalDistance ASC
           LIMIT 1;

```

Στο πιο πάνω ερώτημα Cypher, αρχικά ορίζεται η αρχική και τελική θέση στις μεταβλητές StartPosition και EndPosition αντίστοιχα. Ακολούθως, εντοπίζονται όλα τα πιθανά μονοπάτια από την αρχική μέχρι την τελική θέση, τα οποία καταχωρούνται στη μεταβλητή Paths. Στη συνέχεια, υπολογίζεται ο συνολικός χρόνος για το κάθε μονοπάτι ξεχωριστά με βάση το χαρακτηριστικό χρόνος (estimated_time) της σχέσης των θέσεων και εν τέλει επιστρέφεται το μονοπάτι με τον μικρότερο χρόνο.

Είναι εμφανές λοιπόν το γεγονός ότι με βάση τα χαρακτηριστικά των κόμβων (nodes) του γραφήματος αλλά και των χαρακτηριστικών των σχέσεων τους, μπορούν να προκύψουν ποικίλες διαδρομές όπως για παράδειγμα η αποφυγή αυτοκινητόδρομου, η παράκαμψη των αστυνομικών ελέγχων, προτίμηση διαδρομής η οποία να περνά από παραλιακό δρόμο και πολλές άλλες κατηγοριοποιήσεις διαδρομών.

7. Συμπεράσματα - Επίλογος

Η έρευνα αυτή εστιάστηκε στη δημιουργία μιας υποδομής για την ανάπτυξη εφαρμογών οδικής κυκλοφορίας. Το κυριότερο μέρος της εστιάστηκε στην αναγνώριση κόμβων στις οδικές αρτηρίες χρησιμοποιώντας αποκλειστικά τα δεδομένα που προέκυψαν από την κίνηση οχημάτων και συλλέχθηκαν με τη χρήση Crowdsourcing.

Για την επίτευξη των πιο πάνω, υλοποιήθηκε μια εφαρμογή σε κινητές συσκευές Android, με την οποία έχει καταστεί δυνατή η συλλογή δεδομένων σχετικών με την κίνηση των οχημάτων. Η εφαρμογή χρησιμοποιήθηκε από ένα αριθμό κινητών συσκευών καταγράφοντας δεδομένα από διαφορετικές διαδρομές. Κατά την πιλοτική εφαρμογή, καταγράφηκαν συνολικά 23 διαδρομές και 16607 σημεία από τα οποία τα 304 είχαν οριστεί ως κόμβοι. Με την βοήθεια των υπόλοιπων τεχνολογιών που είχαν αναφερθεί και κυρίως τις μεθόδους μηχανικής μάθησης για την ανάλυση των δεδομένων, έχουν παραχθεί κάποιοι αξιόλογοι κανόνες για την αναγνώριση κόμβων. Τα αποτελέσματα δείχνουν ότι με τη χρήση τεχνικών μηχανικής μάθησης για την ανάλυση δεδομένων κίνησης οχημάτων είναι δυνατή η αυτόματη αναγνώριση κόμβων με πολύ υψηλά ποσοστά επιτυχίας πέρα του 98%.

Περαιτέρω, η χρήση των βάσεων δεδομένων γραφών και η αποθήκευση σε αυτές του συνόλου των επεξεργασμένων δεδομένων, έχει δώσει επιπρόσθετες δυνατότητες και προοπτικές στην υποδομή που έχει δημιουργηθεί στο πρώτο σκέλος της παρούσας έρευνας. Η επέκταση αυτή, έχει προσφέρει τα εφόδια για να ωθήσει την παρούσα υποδομή σε νέες, καινοτόμες εφαρμογές πλοήγησης και διακίνησης πολιτών.

Η έρευνα εστιάστηκε στους κόμβους οι οποίοι αφορούν κυρίως οδικές διασταυρώσεις στην πόλη. Με την ίδια διαδικασία θα ήταν εφικτή η αυτόματη αναγνώριση εξόδων και εισόδων σε αυτοκινητόδρομους αλλά και σε κυκλικές διασταυρώσεις, παρατηρώντας αποκλειστικά τις μεταβολές κατεύθυνσης, ταχύτητας και χρόνου ενδιάμεσα των μεταβαλλόμενων θέσεων του οχήματος. Ο λόγος που δεν λήφθηκαν υπόψη αυτού του τύπου κόμβοι, έγκειται αποκλειστικά στο μικρό σύνολο δεδομένων που έχει καταγραφεί λόγω του περιορισμένου χρόνου που υπήρχε για την υλοποίηση της εργασίας αυτής. Τελειώνοντας, αξιοσημείωτο είναι το γεγονός ότι κατά τη διαδικασία καταγραφής με τη μέθοδο Crowdsourcing δεν έχουν παρουσιαστεί προβλήματα

με τους χρήστες του συστήματος, πράγμα που επιβεβαιώνει τη σωστή προσέγγιση στην αφαιρετική κατασκευή του συστήματος.

8. Βιβλιογραφία

- Barsi, A., & Heipke, C. (2003, May). Detecting road junctions by artificial neural networks. In *Remote Sensing and Data Fusion over Urban Areas, 2003. 2nd GRSS/ISPRS Joint Workshop on* (pp. 129-132). IEEE.
- Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. (M. Jordan, Kleinberg Jon, & Scholkopf Bernhard, Eds.) (1st ed.). Springer.
- Bostanci, B., & Bostanci, E. (2013). An Evaluation of Classification Algorithms Using McNemar's Test, 201. doi:10.1007/978-81-322-1038-2
- Covert, A. (2013). Ubuntu is coming to smartphones, but will anyone use it? Ανακτήθηκε στις 2 Απριλίου, 2014, από <http://money.cnn.com/2013/01/02/technology/mobile/ubuntu-smartphone-linux/>
- Curtiss, M., Becker, I., Bosman, T., Doroshenko, S., Grijincu, L., Jackson, T., Zhang, N. (2013). Unicorn : A System for Searching the Social Graph One-way.
- Doan, A., Ramakrishnan, R., & Halevy, A. Y. (2011). Crowdsourcing systems on the World-Wide Web. *Communications of the ACM*, 54(4), 86. doi:10.1145/1924421.1924442
- Esposti, C. (2010). Jeff Howe aka “Mr. Crowdsourcing” has himself been crowdsourced! Ανακτήθηκε στις 8 Μαΐου, 2014, από <http://www.crowdsourcing.org/editorial/jeff-howe-aka-mr-crowdsourcing-has-himself-been-crowdsourced-/1777>
- Fathi, A., & Krumm, J. (2010). Detecting road intersections from gps traces. In *Geographic Information Science* (pp. 56-69). Springer Berlin Heidelberg.
- Free Software Foundation. (2014). What is free software and why is it so important for society? Ανακτήθηκε στις 8 Απριλίου, 2014, από <http://www.fsf.org/about/what-is-free-software>

- Garner, S. R. (1995, April). Weka: The waikato environment for knowledge analysis. In *Proceedings of the New Zealand computer science research students conference* (pp. 57-64).
- Ekinci, M., Gibbs, F. W., & Thomas, B. T. (2000). Knowledge-based navigation for autonomous road vehicles. *Turkish Journal of Electrical Engineering and Computer*, 8(1).
- Holmes, G., Donkin, A., & Witten, I. H. (1994). *WEKA: A Machine Learning Workbench*. Department of Computer Science, University of Waikato.
- Josef, F., & Max, R. J. (2013). Waze sale signals new growth for Israeli high tech - Yahoo News. Ανακτήθηκε στις 12 Μαρτίου, 2014, από <http://news.yahoo.com/waze-sale-signals-growth-israeli-high-tech-174533585.html>
- Machine Learning Group at the University of Waikato. (2014). WEKA History. Ανακτήθηκε στις 8 Μαΐου, 2014, από <http://www.cs.waikato.ac.nz/ml/weka/history.html>
- Meier, R. (2012). *Professional Android 4 Application Development* (First Edit.). John Wiley & Sons, Inc.
- Oracle. (2013). *Top 10 Reasons to Choose MySQL for Next Generation Web Applications*.
- Robinson, I., Webber, J., & Eifrem, E. (2013). *Graph Databases* (First Edit.). O'Reilly Media, Inc.
- Sam, H., & Gekker, A. (2014). 'Outsmarting Traffic, Together': Driving as Social Navigation Sam Hind (University of Warwick , UK). *Exchanges: The Warwick Research Journal*, 1(2).
- Tatroe, K., MacIntyre, P., & Lerdorf, R. (2013). *Programming PHP*. (M. Blanchette & R. Roumeliotis, Eds.) (Third Edit., pp. 1–14). O'Reilly Media, Inc.

The Center of Technology. (2014). How Technology Has Changed Our Lives in the Last Five Years. Ανακτήθηκε στις 6 Μαΐου, 2014, από <http://www.centertech.org/>

Vavilin, A., & Jo, K. H. (2009). Graph-based Approach for Robust Road Guidance Sign Recognition from Differently Exposed Images. *J. UCS*, 15(4), 786-804.

Woollaston, V. (2014). Could YOU solve the mystery of the missing Malaysia Airlines flight? Satellite images made public to help locate MH370 jet. Ανακτήθηκε στις 8 Μαΐου, 2014, από <http://www.dailymail.co.uk/sciencetech/article-2579064/Could-YOU-solve-mystery-missing-Malaysia-Airlines-flight-Satellite-images-public-help-locate-MH370-jet.html>