



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ & ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ

ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ – ΕΡΓΑΣΤΗΡΙΟ ΤΗΛΕΠΙΣΚΟΠΗΣΗΣ

---

## ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

# ΠΑΡΑΚΟΛΟΥΘΗΣΗ ΠΟΛΛΑΠΛΩΝ ΟΧΗΜΑΤΩΝ ΑΠΟ ΔΟΥΡΥΦΟΡΙΚΑ ΔΕΔΟΜΕΝΑ ΒΙΝΤΕΟ ΜΕ ΧΡΗΣΗ ΦΙΛΤΡΩΝ

## KALMAN

---



**Κωνσταντίνος Μαστοράκης**

*Αθήνα, Ιούνιος 2016*



NATIONAL TECHNICAL UNIVERSITY OF ATHENS  
SCHOOL OF RURAL AND SURVEYING ENGINEERING  
DEPARTMENT OF TOPOGRAPHY – REMOTE SENSING LAB.

---

## DIPLOMA THESIS

# TRACKING MULTIPLE VEHICLES FROM SATELLITE VIDEO USING KALMAN FILTERS

---



**Konstantinos Mastorakis**

*Athens, June 2016*



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ & ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ  
ΤΟΜΕΑΣ ΤΟΠΟΓΡΑΦΙΑΣ – ΕΡΓΑΣΤΗΡΙΟ ΤΗΛΕΠΙΣΚΟΠΗΣΗΣ

**ΠΑΡΑΚΟΛΟΥΘΗΣΗ ΠΟΛΛΑΠΛΩΝ ΟΧΗΜΑΤΩΝ ΑΠΟ ΔΟΡΥΦΟΡΙΚΑ  
ΔΕΔΟΜΕΝΑ VIDEO ΜΕ ΧΡΗΣΗ ΤΟΥ ΦΙΛΤΡΟΥ KALMAN**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

*Μαστοράκης Κωνσταντίνος*

Επιβλέπων Καθηγητής: Καράντζαλος Κωνσταντίνος

Επ. Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή επιτροπή την ..... του 2016

.....  
.....  
Κ. Καράντζαλος  
Επ. Καθηγητής Ε.Μ.Π.

.....  
.....  
Γ. Καρράς  
Καθηγητής Ε.Μ.Π.

.....  
.....  
Κ. Αντωνίου  
Αν. Καθηγητής Ε.Μ.Π.

---

Αθήνα, Ιούνιος 2016

Μαστοράκης Κωνσταντίνος

Διπλωματούχος Αγρονόμος & Τοπογράφος Μηχανικός Ε.Μ.Π.

---

Copyright © *Μαστοράκης Κωνσταντίνος, 2016*

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς το συγγραφέα.



## ΠΡΟΛΟΓΟΣ

Η παρούσα διπλωματική εργασία με τίτλο *“Παρακολούθηση πολλαπλών οχημάτων από δορυφορικά δεδομένα βίντεο με χρήση φίλτρων Kalman”* ανατέθηκε και εκπονήθηκε στο πλαίσιο των προπτυχιακών μου σπουδών, από το Εργαστήριο Τηλεπισκόπησης της Σχολής Αγρονόμων & Τοπογράφων Μηχανικών του Εθνικού Μετσόβιου Πολυτεχνείου, με κύριο στόχο την διερεύνηση, ανάπτυξη και αξιολόγηση μεθόδου παρακολούθησης οχημάτων σε αλληλουχία δορυφορικών εικόνων, τα οποία κινούνται σε τμήμα οδικού άξονα και με απώτερο σκοπό την εξαγωγή στατιστικών στοιχείων σε μαζική κλίμακα για ποικίλες εφαρμογές σε συγκοινωνιακές μελέτες.

Σε περιοχές όπου συγκεντρώνονται μεγάλοι πληθυσμοί η μεγάλη και μαζική ανάγκη για μεταφορά, οδηγεί χωρίς αμφιβολία στην μεγάλη αύξηση των οχημάτων πάσης φύσεως. Αυτό δοκιμάζει όλο και πιο έντονα τα οδικά δίκτυα, ιδιαίτερα εντός των μεγάλων αστικών κέντρων που συνεχώς καλούνται να προσαρμόζονται στις ανάγκες αυτές. Για την βέλτιστη καταγραφή κυκλοφοριακών προτύπων σε κλίμακα συνοικίας, δήμου ή και πόλης και την ανάλυση τάσεων και οδηγικών συμπεριφορών, η αξιοποίηση των νέων δορυφορικών δεδομένων τύπου βίντεο μπορεί να αλλάξει τον τρόπο με τον οποίο πραγματοποιούνται τέτοιου είδους μελέτες.

## Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή μου, Κωνσταντίνο Καράντζαλο, που μου παρείχε την απαραίτητη καθοδήγηση καθ' όλη την πορεία εκπόνησης της εργασίας μου, τον κ. Χρήστο Ιωσηφίδη που μου παρείχε άφθονο υλικό για την βελτίωση των προγραμματιστικών μου ικανοτήτων και για τους νέους τρόπους σκέψης που μου δημιούργησε μεταδίδοντάς μου τις γνώσεις του περί ελεύθερου λογισμικού και της φιλοσοφίας που το διέπει, τον κ. Κωνσταντίνο Μακαντάση, χωρίς τη συμβολή του οποίου και το ενδιαφέρον που επέδειξε για να με βοηθήσει να ξεπεράσω τις τεχνικές δυσκολίες που προέκυψαν με το προγραμματιστικό σκέλος, το αποτέλεσμα της εργασίας θα ήταν σίγουρα υποδεέστερο του παρόντος και τέλος τον κ. Γιώργο Καρρά για την γραμματική και συντακτική επιμέλεια του τεύχους.

Τέλος, θα ήθελα να ευχαριστήσω τους γονείς μου, για την υποστήριξή τους και την υπομονή τους όλα αυτά τα χρόνια, για όλα όσα επένδυσαν σε μένα και τις σπουδές μου συνολικά, γιατί χωρίς τη δική τους συνδρομή δεν θα είχα την ευκαιρία αυτή.

## ΠΕΡΙΛΗΨΗ

Αντικείμενο της παρούσας διπλωματικής εργασίας αποτελεί η μελέτη τεχνικών παρακολούθησης πολλαπλών αντικειμένων/οχημάτων (multiple object tracking) τα οποία κινούνται σε οδικούς άξονες με βάση μοντέλα πρόβλεψης θέσης που βασίζονται στη χρήση φίλτρων τύπου Kalman. Η ανάλυση και παρακολούθηση πραγματοποιείται σε δορυφορικά δεδομένα βίντεο πολύ υψηλής χωρικής ανάλυσης, στα οποία μπορεί να γίνονται διακριτά τα οχήματα ως συσσωρεύσεις λιγοστών pixel. Μετά τον εντοπισμό των κέντρων των οχημάτων στις αρχικές εικόνες της αλληλουχίας με βάση τεχνικές αφαίρεσης υποβάθρου, εφαρμόζονται πολλαπλά και ανά αντικείμενο μοντέλα πρόβλεψης θέσης στο επόμενο καρέ με βάση φίλτρα τύπου Kalman. Με βάση την πρόβλεψη πραγματοποιείται αξιολόγηση σε σχέση με το αποτέλεσμα του αλγορίθμου αφαίρεσης φόντου. Βασικό αποτέλεσμα της παρούσας εργασίας αποτελεί αλγόριθμος που υλοποιεί την παραπάνω μεθοδολογία καθώς και τα σετ των παραμέτρων που οδηγούν σε βέλτιστα αποτελέσματα για τα δεδομένα που χρησιμοποιήθηκαν. Παρ' όλα αυτά το αναπτυγμένο λογισμικό δεν μπορεί να θεωρηθεί έτοιμο για επιχειρησιακή χρήση και απαιτείται εκτεταμένη αξιολόγηση και περαιτέρω ανάπτυξη. Τα αποτελέσματα όμως φαίνεται ότι είναι ιδιαίτερος ελπιδοφόρος για ποικίλες εφαρμογές σε κυκλοφοριακές και συγκοινωνιακές μελέτες.



## **ABSTRACT**

The main goal of this diploma thesis is to study tracking methods of multiple objects, which move on large scale roads based on prediction models which rely on Kalman filters theory. The analysis and tracking is realized on satellite video data of very high spatial resolution, in which vehicles can be distinguished as accumulations of few pixels. After locating the centroids of the vehicles on the raw images of the video sequence using background subtraction techniques, multiple and object by object location prediction models based on Kalman type filters are being recursively applied for each frame. Evaluation of the process is being carried out by comparing the prediction results to those of the background subtraction algorithm. The main product of this diploma thesis consists of an algorithm that implements the methodology described above, as well as the set of parameters that lead to the optimized results for the given datasets. However, the developed software should not be considered ready for operational use. Extended evaluation and further research is required for this to be accomplished. Even so, the results are definitely promising for multiple traffic and transport applications.







# ΠΕΡΙΕΧΟΜΕΝΑ

ΠΡΟΛΟΓΟΣ.....	i
ΠΕΡΙΛΗΨΗ.....	iii
ABSTRACT.....	v
ΠΕΡΙΕΧΟΜΕΝΑ.....	viii
Κεφάλαιο 1 – Εισαγωγή.....	1
1.1 – Δομή της εργασίας.....	1
1.2 – Αντικείμενο και στόχος της εργασίας.....	2
1.3 – Κίνητρο και συνεισφορά της εργασίας.....	3
Κεφάλαιο 2 – Ανασκόπηση βιβλιογραφίας.....	4
2.1 – Ανασκόπηση εργασιών που αφορούν σε tracking με επίγεια μέσα.....	7
2.2 – Ανασκόπηση εργασιών που αφορούν σε tracking με μη επίγεια μέσα.....	9
Κεφάλαιο 3 – Μεθοδολογική προσέγγιση.....	15
3.1 – Εισαγωγή και παρουσίαση περιοχής μελέτης.....	15
3.2 – Επιλογή της μεθόδου για την αντιμετώπιση του προβλήματος εντοπισμού.....	17
3.2.1 – Φίλτρα Kalman : Τι είναι και πώς λειτουργούν.....	18
3.2.2 – Φίλτρα Kalman : Μαθηματική διατύπωση υλοποίηση.....	20
3.2.3 – Χρήση του φίλτρου Kalman για object tracking.....	23
3.3 – Υλοποίηση του αλγορίθμου εντοπισμού σε προγραμματιστικό περιβάλλον.....	26
3.3.1 – Εισαγωγικά.....	26
3.3.2 – Παρουσίαση, επεξήγηση και ανάλυση του κώδικα : Βοηθητικές συναρτήσεις.....	28
3.3.3 – Παρουσίαση, επεξήγηση και ανάλυση του κώδικα : Βασικές συναρτήσεις.....	32
Κεφάλαιο 4 – Πειραματικά Αποτελέσματα και Αξιολόγηση.....	43
4.1 – Παρουσίαση δεδομένων.....	43
4.2 – Αποτελέσματα - Ποιοτική αξιολόγηση.....	46
4.3 – Ποσοτική αξιολόγηση.....	57
Κεφάλαιο 5 – Συμπεράσματα.....	65
5.1 – Εισαγωγικά.....	65
5.2 – Συμπεράσματα.....	65
Κεφάλαιο 6 – Προτάσεις.....	67
Βιβλιογραφία.....	71
Διαδικτυακοί Ιστότοποι.....	72
Παράρτημα I – Κατάλογος Εικόνων.....	74
Παράρτημα II – Κατάλογος Πινάκων και Γραφημάτων.....	76



# Κεφάλαιο 1 – Εισαγωγή

Στην Γεωπληροφορική και τις Μεθόδους Παρατήρησης Γης υπάρχουν πλέον διαθέσιμα την τελευταία τετραετία και για εμπορικούς σκοπούς δορυφορικά δεδομένα βίντεο. Συγκεκριμένα, τα δεδομένα αυτά είναι υψηλής χωρικής ανάλυσης, με αποτέλεσμα να μπορούν να χρησιμοποιηθούν εκτός από εφαρμογές παρακολούθησης του φυσικού περιβάλλοντος και σε εφαρμογές στον αστικό και περιαστικό χώρο.

Προς την κατεύθυνση αυτή, στην παρούσα εργασία βασική επιδίωξη αποτελεί η ανάλυση δορυφορικών δεδομένων βίντεο με σκοπό τον συνεχή εντοπισμό και χαρτογράφηση σε κάθε καρέ του βίντεο της θέσης των οχημάτων στους οδικούς άξονες.

Η παρακολούθηση (tracking) πολλαπλών αντικείμενων ουσιαστικά αναφέρεται στη μελέτη και αξιολόγηση τεχνικών όρασης υπολογιστών για τον αυτοματοποιημένο εντοπισμό και πρόβλεψη θέσης των αντικείμενων σε αλληλουχία εικόνων. Η μεθοδολογία που εξετάστηκε βασίζεται στα φίλτρα Kalman..

## 1.1 – Δομή της εργασίας

Η ενότητα αυτή έχει ως σκοπό να δώσει μια συνοπτική εικόνα για τον τρόπο που έχει δομηθεί και διαρθρωθεί η εργασία. Η σειρά των κεφαλαίων ταιριάζει και με την πορεία εκπόνησης της.

### **Πιο αναλυτικά:**

Το πρώτο κεφάλαιο περιέχει κάποιες εισαγωγικές σημειώσεις, ενώ παρουσιάζεται το αντικείμενο της εργασίας και οι στόχοι και η συνεισφορά της.

Στο κεφάλαιο 2 περιγράφεται η σχετική πρόσφατη βιβλιογραφία σε σχέση με τα αντικείμενα της παρακολούθησης αντικειμένων.

Στο κεφάλαιο 3 περιγράφεται η περιοχή μελέτης, ενώ ακολουθεί συνοπτικά περιγραφή βασικών σημείων της αναπτυγθείσας μεθοδολογίας. Παρουσιάζονται επίσης αναλυτικά στοιχεία από την υλοποίηση της μεθοδολογίας σε προγραμματιστικό περιβάλλον.

Στο κεφάλαιο 4 ακολουθεί η παρουσίαση των αρχικών δεδομένων καθώς και προϊόντων μετά την επεξεργασία τους. Παρατίθενται επίσης και τα αποτελέσματα έπειτα από διάφορα πειράματα που πραγματοποιήθηκαν, καθώς και η αξιολόγηση τους, τόσο ποιοτικά όσο και ποσοτικά.

Τέλος, στο κεφάλαιο 5 αναφέρονται τα συνολικά συμπεράσματα που εξήχθησαν κατά τη διάρκεια της εργασίας αλλά και από τα αποτελέσματα της αξιολόγησης. Επίσης αναφέρονται συνοπτικά και προτάσεις που κρίνεται



πως θα συμβάλλουν στην περαιτέρω ανάπτυξη, βελτίωση και αυτοματοποίηση του κώδικα, αν κάποιος θελήσει να ασχοληθεί με το συγκεκριμένο αντικείμενο και πάρει ως βάση την παρούσα εργασία.

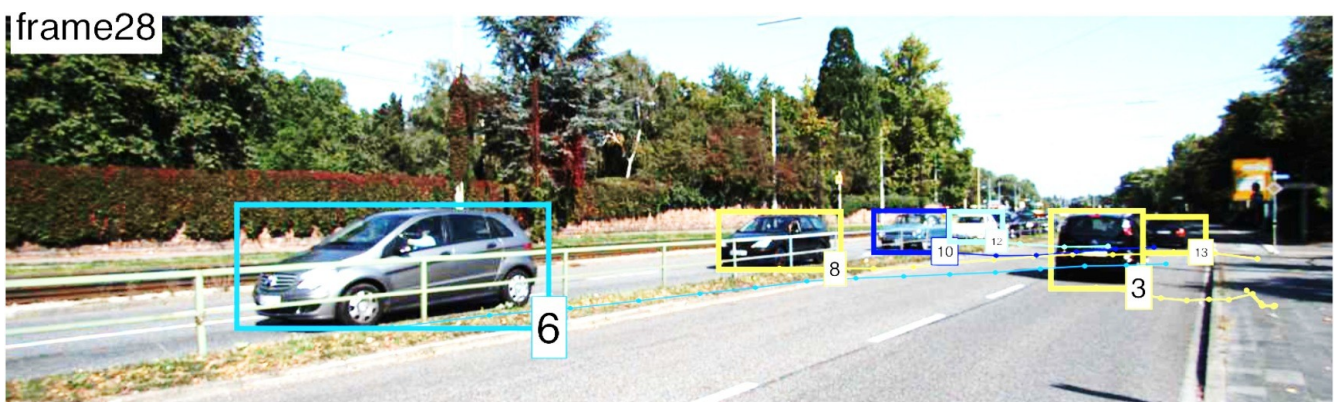
## 1.2 – Αντικείμενο και στόχος της εργασίας

Το αντικείμενο της εργασίας αυτής ήταν η μελέτη και αξιολόγηση μεθοδολογίας για τον εντοπισμό και ταυτοποίηση πολλαπλών αντικειμένων/οχημάτων τα οποία κινούνται από καρέ σε καρέ. Το αντικείμενο αυτό υπάγεται σε τομείς όπως οι Μέθοδοι Παρατήρησης Γης, Τηλεπισκόπηση και Όραση Υπολογιστών.

Στόχος της εργασίας ήταν να απαντηθούν ερωτήματα όπως κατά πόσο είναι δυνατόν να εντοπιστούν και να ταυτοποιηθούν ανά καρέ πολλαπλά αντικείμενα (multiple object tracking) που κινούνται κατά μήκος ενός μεγάλου οδικού άξονα, με βάση δορυφορικά δεδομένα βίντεο υψηλής χωρικής ανάλυσης. Σημειώνεται ότι τα αντικείμενα αυτά αποτυπώνονται με μόλις μερικά εικονοστοιχεία σε κάθε εικόνα.

Η παρακολούθηση πολλαπλών αντικειμένων είναι αντικείμενο που έχει απασχολήσει ήδη αρκετά την επιστημονική κοινότητα. Η σημαντική διαφορά που υπάρχει στο αντικείμενο που εξετάστηκε εδώ είναι ότι τα δεδομένα είναι δορυφορικά και όχι επίγεια, επιχειρείται η ταυτόχρονη ταυτοποίηση πολλαπλών αντικειμένων ενώ όπως αναφέρθηκε τα αντικείμενα περιγράφονται από πολύ μικρό αριθμό εικονοστοιχείων.

Χαρακτηριστικά παρουσιάζεται παρακάτω αποτέλεσμα αλγορίθμου που πραγματοποιεί εντοπισμό πολλαπλών αντικειμένων.



*Εικόνα 1.1 : Παράδειγμα αλγορίθμου που εκτελεί εντοπισμό πολλαπλών αντικειμένων*

### 1.3 – Κίνητρο και συνεισφορά της εργασίας

Το βασικό κίνητρο που οδήγησε στην διπλωματική αυτή εργασία, ήταν το προσωπικό μου ενδιαφέρον για την όραση υπολογιστών, η ανάγκη να το συνδυάσω με χωρικά και γενικότερα με τοπογραφικής φύσης δεδομένα, αλλά και η πρόκληση να υλοποιηθεί όλο αυτό το εγχείρημα σε δορυφορικά δεδομένα, κάτι που έχει πολύ μεγάλες δυνατότητες ως αντικείμενο. Πιο συγκεκριμένα, ένα πολύ σημαντικό κίνητρο-πρόκληση ήταν το ότι θα δοκιμαζόταν αλγόριθμος παρακολούθησης σε βίντεο.

Από επιστημονικής σκοπιάς, βασικό κίνητρο αποτέλεσε η ιδέα να δημιουργηθεί ένα τέτοιου είδους λογισμικό, που θα έδινε νέα δυναμική όσον αφορά τις συγκοινωνιακές μελέτες οδικών αξόνων με πραγματικά δεδομένα και που είναι δυνατόν να λειτουργήσει σε πραγματικό χρόνο, επιτρέποντας τη συλλογή τους και την επεξεργασία τους με σκοπό την εξαγωγή στατιστικών δεδομένων.

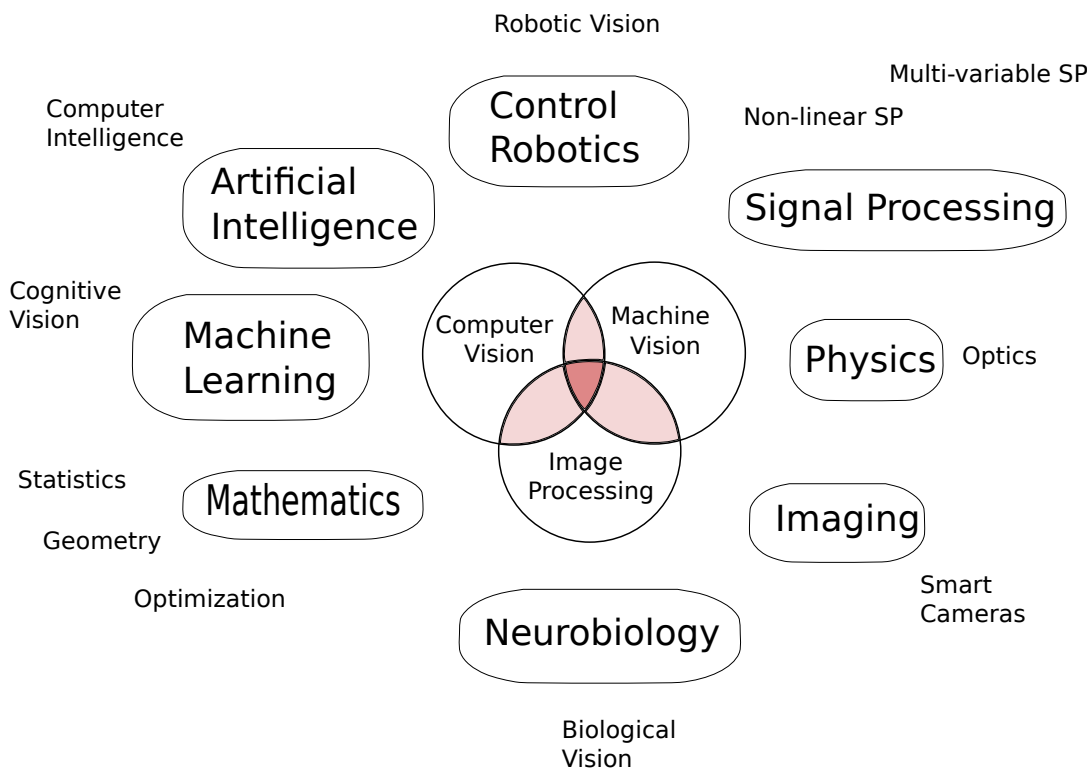
Κύρια σημεία της συνεισφοράς αποτελούν η ανάπτυξη του εν λόγω λογισμικού με δυνατότητες εντοπισμού κινούμενων αντικειμένων και πρόβλεψης θέσης στο επόμενο καρέ/ χρονική στιγμή. Συμπεράσματα από την ανάλυση και αξιολόγηση των αποτελεσμάτων όσο και η μεθοδολογία αυτή για τα δορυφορικά δεδομένα βίντεο ενδεχομένως μπορούν να βρουν και άλλες εφαρμογές, αλλά κυρίως επισημαίνεται το γεγονός πως μπορούν δυνητικά να παρέχουν τη βάση ενός πολύ χρήσιμου ολοκληρωμένου συστήματος, για την παρακολούθηση, μελέτη και εξαγωγή ζωτικής σημασίας στατιστικών στοιχείων που αφορούν κυκλοφοριακές και συγκοινωνιακές παραμέτρους.

## Κεφάλαιο 2 – Ανασκόπηση βιβλιογραφίας

Η εργασία έχει δύο διακριτά θεωρητικά υπόβαθρα, εκείνα της παρακολούθησης αντικειμένων από ηλεκτρονικό υπολογιστή και των φίλτρων Kalman. Στο κεφάλαιο αυτό θα παρουσιαστεί βιβλιογραφία σχετική μόνο με το πρώτο, καθώς αναφορά στο θεωρητικό υπόβαθρο των φίλτρων Kalman θα πραγματοποιηθεί στο κεφάλαιο της μεθοδολογίας που ακολουθεί.

Η βιβλιογραφία σχετικά με την επιστήμη της όρασης των υπολογιστών εντοπίζεται για πρώτη φορά χρονολογικά περίπου το 1960, όταν ο κατά κοινή αποδοχή πατέρας της επιστήμης αυτής Larry Roberts ( T.S. Huang, *Computer Vision: Evolution and Promise* ) εξέδωσε την διδακτορική του διατριβή, στην οποία μεταξύ άλλων κάνει λόγο για το αν είναι δυνατό να εξαχθεί τρισδιάστατη γεωμετρική πληροφορία από δισδιάστατες εικόνες πολύεδρων αντικειμένων, από διαφορετική οπτική γωνία μέσω ενός ηλεκτρονικού υπολογιστή.

Το αντικείμενο της επιστήμης αυτής είναι ιδιαίτερα εκτεταμένο και καλύπτει ένα ευρύτατο φάσμα πεδίων μελέτης. Ταυτόχρονα αλληλεπιδρά και συσχετίζεται και με πολλές ακόμα επιστήμες. Η παρακάτω εικόνα δείχνει ποιες είναι οι αλληλεπιδράσεις αυτές.



Εικόνα 2.1: Αλληλεπίδραση της Όρασης Υπολογιστών με άλλα επιστημονικά πεδία

[Πηγή: [https://en.wikipedia.org/wiki/Computer\\_vision](https://en.wikipedia.org/wiki/Computer_vision)]

Έκτοτε σε συνδυασμό με την εξέλιξη της τεχνολογίας η επιστήμη αυτή και οι εφαρμογές έχουν κάνει αλματώδη πρόοδο. Αυτό οφείλεται στον πολλαπλασιασμό της υπολογιστικής ισχύος των ηλεκτρονικών υπολογιστών, την δραματική αύξηση των τρόπων που αποκτάται και μεταφέρεται η πληροφορία, αλλά και στο γεγονός ότι είναι πολύ πιο εύκολο και προσιτό για τον καθένα να αποκτήσει πρόσβαση στις πληροφορίες αυτές και στα μέσα για να τις εκμεταλλευτεί.

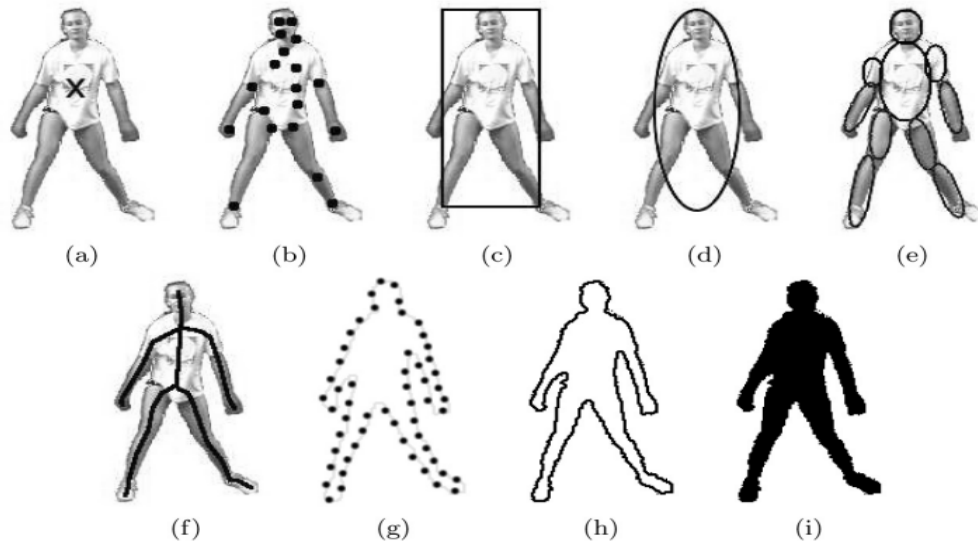
Η παρούσα εργασία τοποθετείται σε μια υπό-περιοχή της όρασης των υπολογιστών, τον εντοπισμό και την παρακολούθηση κινούμενων αντικειμένων (object tracking). Ο τομέας αυτός είναι ένας πολλά υποσχόμενος και ταχέως αναπτυσσόμενος τομέας της ευρύτερης επιστήμης της όρασης υπολογιστών (computer vision) με πλούσια βιβλιογραφία καθώς αποτελεί ένα από τα προβλήματα του οποίου η λύση συνιστά μια από τις μεγαλύτερες προκλήσεις της επιστήμης όρασης υπολογιστών.

Στο κεφάλαιο αυτό γίνεται μια συνοπτική ανασκόπηση βιβλιογραφίας σχετικής με το αντικείμενο της εργασίας με την εξής δομή. Βιβλιογραφία σχετική με τον εντοπισμό και παρακολούθηση (object tracking) αντικειμένων από επίγεια μέσα θα συμπεριληφθεί σε μία ενότητα, ενώ βιβλιογραφία σχετική με το (object tracking) από μη επίγεια μέσα θα συμπεριληφθεί σε μία δεύτερη ενότητα.

Πριν όμως και απ τις δύο ενότητες αξίζει να γίνει αναφορά σε μία δημοσίευση η οποία αφορά τις μεθόδους με τις οποίες μπορεί να επιτευχθεί το tracking.

Το 2006 οι *Yilmaz, A., Javed, O., and Shah, M.* δημοσίευσαν την εργασία τους με τίτλο “**Object Tracking: A Survey**”. Στόχος τους είναι να κάνουν μια ανασκόπηση του state of the art των μεθόδων για παρακολούθηση αντικειμένων, να τις κατηγοριοποιήσει αλλά και να μελετήσει νέες τάσεις. Αποτελεί μία πολύ ενδιαφέρουσα μελέτη καθώς αναφέρεται σε θεμελιώδεις έννοιες που περιλαμβάνονται στο πεδίο αυτό.

Μια εκ των σημαντικότερων εννοιών που αναδεικνύεται και αναλύεται είναι αυτή της αναπαράστασης των αντικειμένων, δηλαδή ο τρόπος με τον οποίο ο υπολογιστής αναπαριστά τα προς εντοπισμό αντικείμενα αφού τα “διακρίνει”. Η παρακάτω εικόνα απεικονίζει πιθανούς τρόπους αναπαράστασης αντικειμένων.



**Fig. 1.** Object representations. (a) Centroid, (b) multiple points, (c) rectangular patch, (d) elliptical patch, (e) part-based multiple patches, (f) object skeleton, (g) complete object contour, (h) control points on object contour, (i) object silhouette.

*Εικόνα 2.2: Αναπαραστάσεις αντικειμένων*

[Πηγή: A. Yilmaz et al., 2006]

Η εργασία καταλήγει ότι οι μέθοδοι για εντοπισμό χωρίζονται σε 3 κατηγορίες με κριτήριο τη χρήση της αναπαράστασης του αντικειμένου και οι οποίες κατηγορίες είναι οι εξής:

- Μέθοδοι που δημιουργούν αντιστοιχία σημείων
- Μέθοδοι που κάνουν χρήση πολύ απλών γεωμετρικών μοντέλων
- Μέθοδοι που κάνουν χρήση της τεχνικής του contour evolution (αναφέρεται ο αγγλικός όρος καθώς δεν βρέθηκε κάποιος όρος που να αποδίδει σωστά την τεχνική αυτή στα ελληνικά)



## 2.1 – Ανασκόπηση εργασιών που αφορούν σε tracking με επίγεια μέσα

Το 2011 στην εργασία τους με τίτλο “**Robust Object Tracking Using Kalman Filters with Dynamic Covariance**” οι *Sheldon Xu* και *Anthony Chang* προσπάθησαν να πραγματοποιήσουν παρακολούθηση αντικειμένων χρησιμοποιώντας ανεξάρτητους αλγορίθμους παρακολούθησης ως δεδομένα εισόδου σε ένα και μοναδικό φίλτρο Kalman. Για τον υπολογισμό του σφάλματος του κάθε αλγορίθμου από χαρακτηριστικά σχετικά με αυτόν χρησιμοποιήθηκε μια συνάρτηση, η οποία εκπαιδεύτηκε με τη μέθοδο της γραμμικής παλινδρόμησης. Το σφάλμα αυτό χρησιμοποιείται ως η μεταβλητότητα της μέτρησης του αλγορίθμου. Η απώτερη ιδέα όλου του εγχειρήματος, είναι η δημιουργία ενός συνολικού “έξυπνου” φίλτρου παρακολούθησης που συνδυάζει / επιλέγει τον αλγόριθμο που ανταποκρίνεται καλύτερα σε κάθε περίπτωση. Ταυτόχρονα περιορίζει τη χρήση του υπό δυσμενείς συνθήκες. Παράλληλα είχαν σαν σκοπό το φίλτρο αυτό να είναι πολύ καινοτόμο και ανεξάρτητο από την ανάγκη να προγραμματίζονται κάθε φορά κανόνες σε σχέση με το περιβάλλον.

Τα αποτελέσματα που προέκυψαν από τα πειράματά τους ήταν “ανάμεικτα” όπως οι ίδιοι δηλώνουν. Η αιτία ήταν ότι το φίλτρο Kalman που σχεδίασαν και δημιούργησαν αύξανε την ακρίβεια εντοπισμού σε έναν τύπο περιβάλλοντος, στις περισσότερες περιπτώσεις, ο σχετικά απλούστερος αλγόριθμος CamShift είτε το έφτανε σε ακρίβεια, είτε το ξεπερνούσε. Συνεπώς η προσπάθεια αυτή είχε μερική επιτυχία και όχι απόλυτη.

Άλλη μια πολύ ενδιαφέρουσα προσέγγιση στη διαδικασία παρακολούθησης αντικειμένων και μάλιστα πολλαπλών υπάρχει στην εργασία των *Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua* με τίτλο “**Multiple Object Tracking using K-Shortest Paths Optimization**”, η οποία αποσκοπεί να λύσει το εξής πρόβλημα που δημιουργείται κατά την διαδικασία παρακολούθησης πολλών αντικειμένων ταυτόχρονα: Όταν υπάρχουν πολλά αντικείμενα, το στάδιο της σύνδεσης των ανιχνεύσεων των αντικειμένων στα συνεχόμενα καρέ, που ουσιαστικά υλοποιεί την παρακολούθηση, καταλήγει να είναι ένα δύσκολο πρόβλημα βελτιστοποίησης λόγω πολλών οικογενειών σημείων ή περιγραμμάτων κ.λ.π. και τροχιών. Σύμφωνα με τους συγγραφείς, η δυσκολία αυτή αντιμετωπίζεται με δειγματοληψία ή “άπληστη αναζήτηση” βασισμένες πάνω σε μεταβλητές του δυναμικού προγραμματισμού, που όμως είναι πολύ πιθανό να μην συμπεριφερθούν βέλτιστα.

Για το λόγο αυτό αναδιαμορφώνουν το συγκεκριμένο στάδιο-βήμα ως μια ήπια βελτιστοποίηση της ροής καταλήγοντας σε ένα κυρτό πρόβλημα. Εκμεταλλευόμενοι αυτήν την συγκεκριμένη δομή λύνουν το πρόβλημα αυτό χρησιμοποιώντας έναν αλγόριθμο συντομότερης διαδρομής (k-shortest paths algorithm)

Το αποτέλεσμα του αλγορίθμου που δημιούργησαν φαίνεται στις παρακάτω εικόνες, οι οποίες είναι τραβηγμένες από διαφορετικές οπτικές γωνίες.



**Εικόνα 2.3: Στιγμιότυπα με το αποτέλεσμα του αλγορίθμου**

[Πηγή : Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua , “*Multiple Object Tracking using K-Shortest Paths Optimization*” , February 2011]

Το συμπέρασμα στο οποίο κατέληξαν, είναι ότι είναι δυνατό και μάλιστα το πέτυχαν, να υποβαθμίσουν τη δυσκολία του προβλήματος της εκτίμησης σε ένα τυπικό γραμμικό πρόβλημα. Με τον αλγόριθμο k-shortest path για την βελτιστοποίηση του προβλήματος, η πολυπλοκότητά του μειώθηκε στο ελάχιστο σε σχέση με την αρχική προσέγγιση, δίνοντας έναν αποδοτικό συνολικό αλγόριθμο που πραγματοποιεί πολύ ικανοποιητικά την παρακολούθηση σε πραγματικό χρόνο σε έναν μέσο ηλεκτρονικό υπολογιστή.

## 2.2 – Ανασκόπηση εργασιών που αφορούν σε tracking με μη επίγεια μέσα

Η εργασία των *Lingfei Meng, John P. Kerekes*, με τίτλο “**Object Tracking Using High Resolution Satellite Imagery**”, αποτελεί μια εξαιρετικά ενδιαφέρουσα εργασία ως προς το αντικείμενο για το λόγο ότι ο αλγόριθμος εντοπισμού εφαρμόζεται σε δορυφορικές εικόνες. Δηλαδή έχει ίδιας φύσης δεδομένα με την παρούσα εργασία, ασχέτως του ότι πρόκειται για μεμονωμένα καρέ και όχι για βίντεο. Πιο συγκεκριμένα αυτή οι συγγραφείς εισάγουν έναν αλγόριθμο εντοπισμού, ο οποίος περιέχει εκτίμηση της θέσης κινούμενων αντικειμένων, μοντελοποίηση του αντικειμένου-στόχου που τίθεται προς εντοπισμό και ταυτοποίηση των αντικειμένων/στόχων σε με μια διαδικασία τριών σταδίων. Δυνητικά τα κινούμενα αντικείμενα είναι τα πρώτα που αναγνωρίζονται στη χρονοσειρά των εικόνων. Το αντικείμενο-στόχος μοντελοποιείται με την εξαγωγή φασματικών όσο και χωρικών χαρακτηριστικών / ιδιοτήτων. Κατά την διαδικασία της ταυτοποίησης των αντικειμένων η απόσταση Bhattacharyya, η αλληλοεπικάλυψη των ιστογραμμάτων και η ομοιότητα του πλήθους των pixel, συνδυάζονται σε ένα καινοτόμο τοπικό τελεστή. Ο αλγόριθμος έχει δοκιμαστεί χρησιμοποιώντας ένα σετ με ακολουθίες εικόνων από πολλές οπτικές γωνίες που προέρχονται από τον δορυφόρο WorldView-2. Η επίδοσης όσον αφορά την παρακολούθηση αναλύονται με τον υπολογισμό των recall, precision, και του δείκτη F1, ο υπολογισμός του οποίου προϋποθέτει τη γνώση των δύο προηγούμενων μεγεθών.

Σχηματικά απεικονίζεται το σχετικό διάγραμμα ροής:

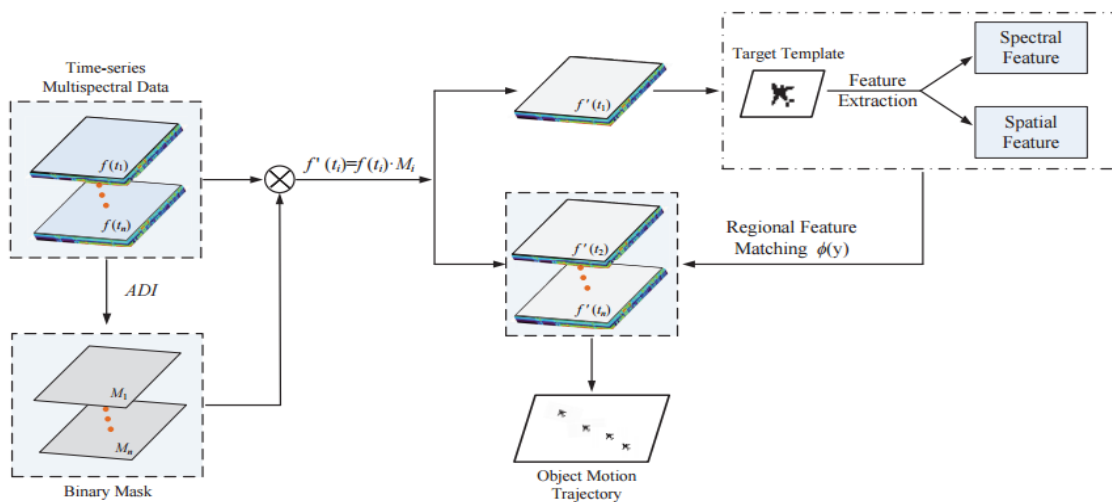


Fig. 1: Flowchart of the proposed object tracking algorithm.

### Εικόνα 2.4: Διάγραμμα ροής του παρουσιαζόμενου αλγορίθμου

[Πηγή: L. Meng, J. P. Kerekes, “*Object Tracking Using High Resolution Satellite Imagery*”, January 2012 ]

Το αποτέλεσμα είναι ο επιτυχής εντοπισμός τριών αντικειμένων σε δυο περιοχές, ενός πλοίου σε ένα λιμάνι και δύο αεροσκαφών στο χώρο ενός αεροδρομίου.

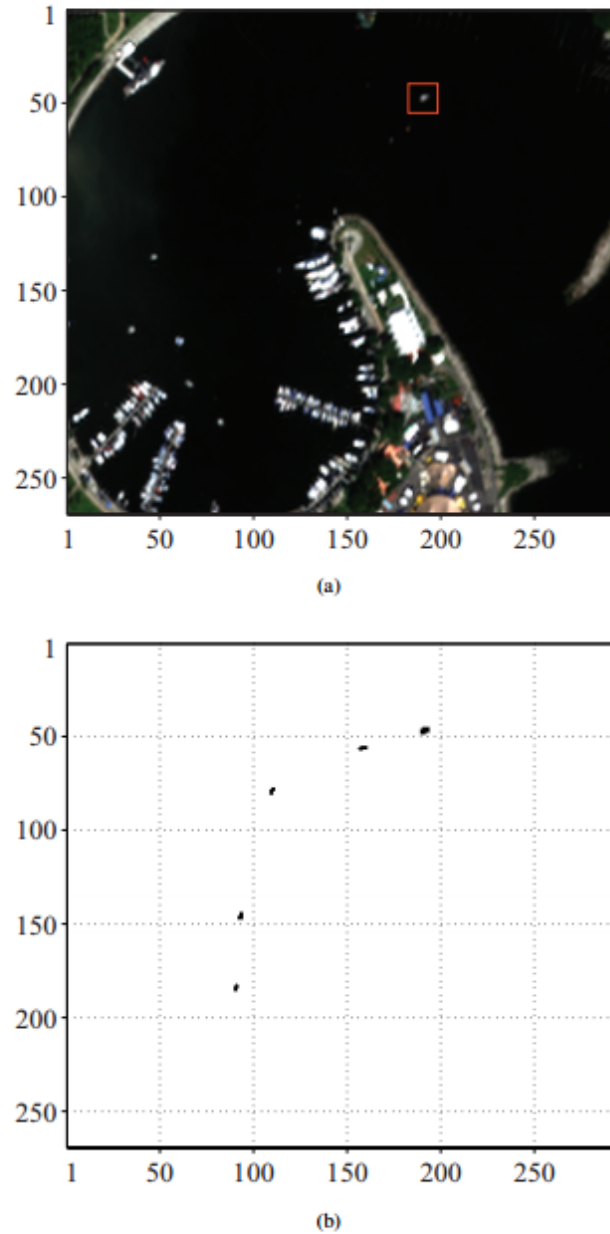


Fig. 5: Ship tracking in the harbor area. (a) Harbor area RGB image, where the ship is within red outline. (b) Ship movement trajectory.

*Εικόνα 2.5: Εντοπισμός του πλοίου στην περιοχή του λιμανιού (a) με το πλοίο να βρίσκεται εντός του κόκκινου περιγράμματος. (b) Η τροχιά κίνησης του πλοίου.*

[Πηγή: L. Meng, J. P. Kerekes, “Object Tracking Using High Resolution Satellite Imagery”, January 2012 ]

Και αντιστοίχως για την περιοχή του αεροδρομίου:

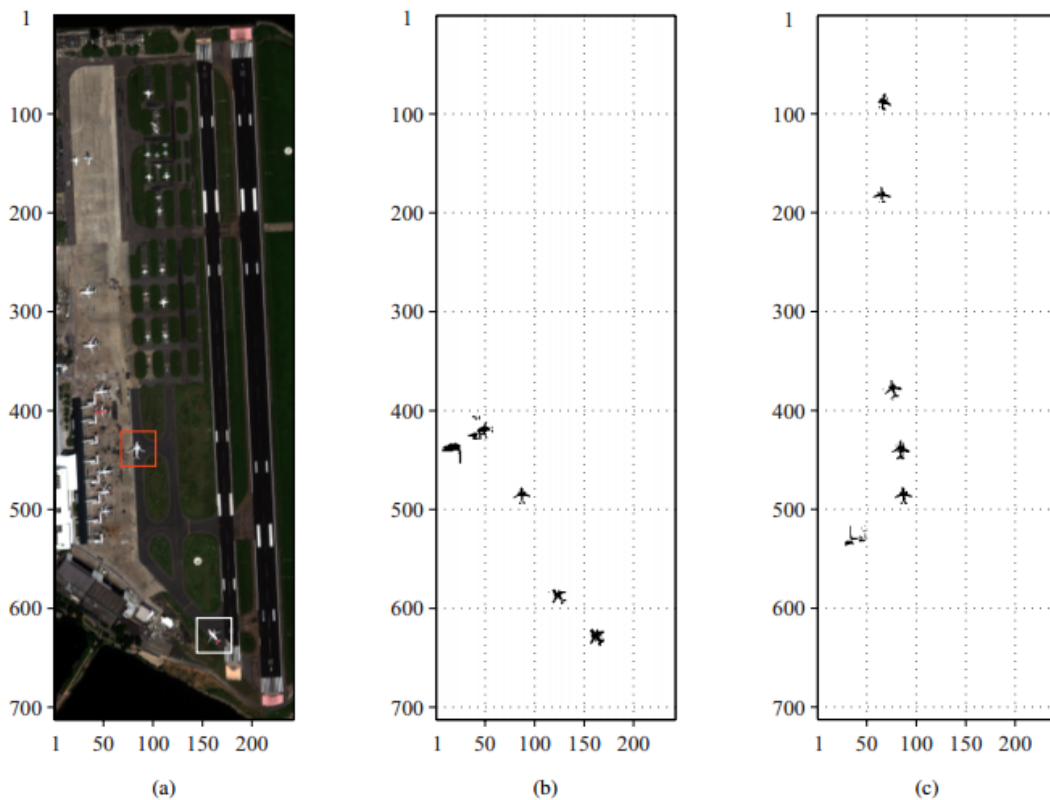


Fig. 6: Aircraft tracking in the airport area. (a) Airport area RGB image, where aircraft A is within the white outline and aircraft B is within the red outline. (b) Aircraft A movement trajectory. (c) Aircraft B movement trajectory.

**Εικόνα 2.6: Εντοπισμός των αεροσκαφών στην περιοχή του αεροδρομίου (a). Το αεροσκάφος A είναι μέσα σε άσπρο πλαίσιο και το B σε κόκκινο. (b) Τροχιά κίνησης του αεροσκάφους A. (c) Τροχιά κίνησης του αεροσκάφους B.**

[Πηγή: L. Meng, J. P. Kerekes, “*Object Tracking Using High Resolution Satellite Imagery*”, January 2012 ]

Τα συμπεράσματα στα οποία καταλήγουν οι συγγραφείς είναι πως η τεχνική που εφαρμόζεται στον αλγόριθμο αυτό, βρίσκει δυνητικά εφαρμογή και σε άλλους τομείς έρευνας, όπως στην εκτίμηση της ταχύτητας ενός αντικειμένου και την επίβλεψη της οδικής κυκλοφορίας.

Σημειώνουν επίσης πως με βάση τη χωρική ανάλυση των διαθέσιμων σετ δεδομένων καθώς και το σχετικά μεγάλο χρονικό διάστημα που μεσολαβεί ανάμεσα σε δύο καρέ, είναι πολύ δύσκολο να αναγνωριστούν οπτικά πανομοιότυπα αντικείμενα μικρού γεωμετρικού μεγέθους στις ακολουθίες των εικόνων. Ο εντοπισμός μικρών αντικειμένων, όπως οχήματα, σε ένα πιο πυκνό περιβάλλον είναι μια μεγαλύτερη πρόκληση και θα ληφθεί υπ' όψιν σε μελλοντικές εργασίες τους, όπως επίσης και επεκτάσεις του αλγορίθμου ούτως ώστε να συμπεριληφθεί ένα κινηματικό μοντέλο για να περιγραφεί η ομαλή κίνηση των αντικειμένων.

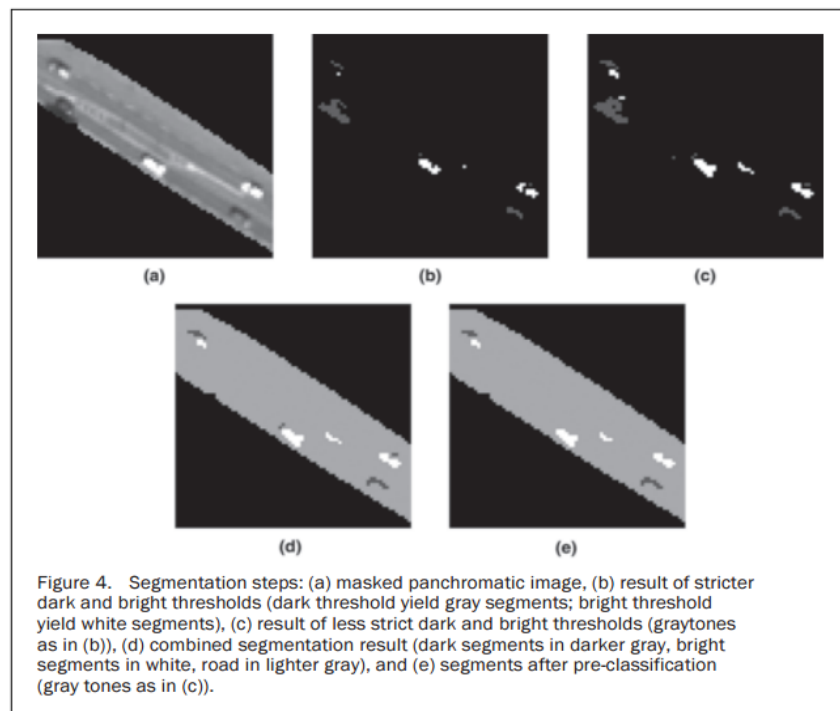
Η επόμενη δημοσίευση τοποθετείται ακόμα πιο κοντά στην παρούσα διπλωματική εργασία καθώς το αντικείμενο της, δηλαδή η μελέτη της οδικής κυκλοφορίας βασίζονται στην ίδια ιδέα· την μελέτη της οδικής κυκλοφορίας από δορυφορικά δεδομένα υψηλής ανάλυσης.

Η εργασία αυτή εκπονήθηκε το 2009 από τους *Siri Øyen Larsen, Hans Koren, και Rune Solberg* και έχει τίτλο : **“Traffic Monitoring using Very High Resolution Satellite Imagery”**.

Σύμφωνα με τους συγγραφείς, η χρήση δορυφορικών δεδομένων πολύ υψηλής ανάλυσης για την μελέτη της οδικής κυκλοφορίας, έχει πολλά πλεονεκτήματα σε σχέση με τις παραδοσιακές μεθόδους. Ο μόνος βασικός περιορισμός είναι η χρονική ανάλυση, δηλαδή ο χρόνος μεταξύ διαδοχικών εικόνων.

Όσον αφορά ειδικότερα το αντικείμενο της εργασίας τους, σε συνεργασία με τη νορβηγικές αρχές που είναι αρμόδιες για την οδική κυκλοφορία το εθνικό συμβούλιο έρευνα και το νορβηγικό διαστημικό κέντρο, ανέπτυξαν μια προσέγγιση για ανίχνευση οχημάτων από εικόνες του δορυφόρου QuickBird. Ο αλγόριθμος αποτελείται από ένα βήμα κατάτμησης της εικόνας, ακολουθούμενο από ταξινόμηση μέγιστης πιθανοφάνειας βασισμένη στο αντικείμενο που ζητείται να εντοπιστεί.

Η αμέσως επόμενη εικόνα παρουσιάζει τα στάδια κατάτμησης / τμηματοποίησης της εικόνας:



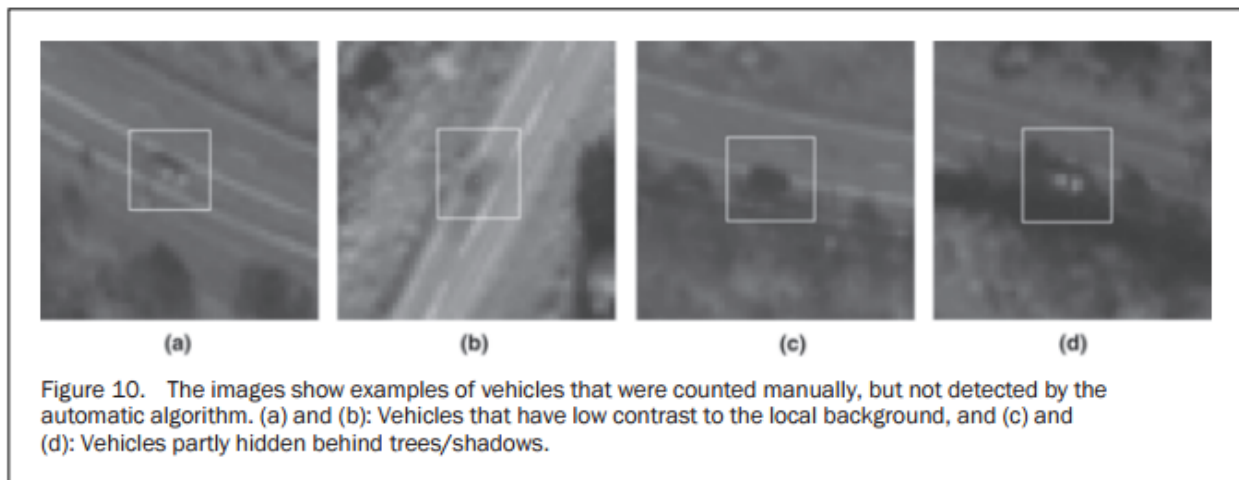
**Εικόνα 2.7: Στάδια τμηματοποίησης (segmentation) της εικόνας**

[Πηγή: *Siri Øyen Larsen, Hans Koren, Rune Solberg* , **“Traffic Monitoring using Very High Resolution Satellite Imagery”** , July 2009 ]



Επίσης προτείνουν μια νέα προσέγγιση για την πρόβλεψη των σκιών των οχημάτων. Η πληροφορία που προκύπτει για τις σκιές χρησιμοποιείται ως ένα χαρακτηριστικό άρρηκτα συνδεδεμένο με το όχημα με σκοπό την βελτίωση της ταξινόμησης. Η αξιολόγηση της σωστής ταξινόμησης είναι 89%, μη συμπεριλαμβανομένου του θορύβου. Η προτεινόμενη μέθοδος τείνει να υποτιμά το πλήθος των οχημάτων συγκρινόμενη με τη χειροκίνητη καταμέτρηση και την καταμέτρηση που προκύπτει από εντός του οδικού άξονα εξοπλισμό.

Μια αδυναμία του αλγορίθμου όπως άλλωστε δηλώθηκε και πριν, είναι πως παρουσιάζει πρόβλημα στον εντοπισμό, όταν δεν υπάρχει μεγάλη αντίθεση με το υπόβαθρο της εικόνας.



**Εικόνα 2.8: Περιπτώσεις αδυναμίας εντοπισμού των αντικειμένων από τον αλγόριθμο**

[Πηγή: Siri Øyen Larsen, Hans Koren, Rune Solberg , “**Traffic Monitoring using Very High Resolution Satellite Imagery**” , July 2009 ]

Μεταξύ άλλων συμπερασμάτων τα κυριότερα στα οποία καταλήγουν οι συγγραφείς είναι πως οι σκιές των οχημάτων παρέχουν πολύτιμη πληροφορία για τα τμήματα της εικόνας όπου πιθανώς υπάρχει κάποιο όχημα. Η πληροφορία αυτή πρέπει -κατά τη γνώμη τους- να χρησιμοποιηθεί στη διαδικασία της ανίχνευσης.

Από τις κλάσεις τις οποίες δημιούργησαν για την ταξινόμηση των εντοπιζόμενων αντικειμένων, το 20% περίπου του συνολικού αριθμού των τμημάτων που εντοπίστηκαν, δεν ανήκουν σε καμία κλάση και εκτός αυτών το 28% είναι λανθασμένα ταξινομημένο ως οχήματα.

Τέλος η αστοχία εντοπισμού κάποιων οχημάτων οφείλεται σε τοπικές μεταβολές του φωτισμού, ή εμποδιζόμενη ορατότητα στο δρόμο λόγω βλάστησης.

### 2.3 – Συμπεράσματα για τη φύση του προβλήματος

Το σημαντικότερο συμπέρασμα που εξάγεται από την βιβλιογραφία περί εντοπισμού και παρακολούθησης αντικειμένων, είναι πως σίγουρα δεν έχει λυθεί το πρόβλημα σε όλη του την έκταση. Κάθε προσπάθεια είναι μια νέα προσέγγιση επίλυσης ενός τμήματος του συνολικότερου προβλήματος, που και πάλι δεν επιλύεται στο ακέραιο. Είναι σαφές πως τα χαρακτηριστικά και οι ιδιότητες της εικόνας, αλλά και το περιεχόμενό της, παίζουν καθοριστικό ρόλο στην προσέγγιση που πραγματοποιείται, στο αποτέλεσμα της προσπάθειας και στην ακρίβεια του αλγορίθμου. Ιδιαίτερα σε περιπτώσεις, που η κλίμακα των δεδομένων μεγαλώνει λόγω του μέσου λήψης (π.χ. εναέριες ή δορυφορικές), τόσο μεγαλύτερη ανομοιογένεια ως προς τα αντικείμενα που απεικονίζονται, τη φωτεινότητα, την αντίθεση και την υφή αναμένεται να παρουσιαστεί. Αυτός είναι και ο λόγος, που η συνολική επίλυση του προβλήματος σε μεγάλης κλίμακας δεδομένα αποτελεί μια μεγάλη πρόκληση.

Κατά την προσωπική μου γνώμη, στην περίπτωση που υπήρχε ένας αλγόριθμος παρακολούθησης αντικειμένων, που να μπορούσε να εφαρμοστεί παντού, θα επρόκειτο για έναν πολύ περίπλοκο αλγόριθμο, με στοιχεία νοημοσύνης μέσα σε αυτόν, ούτως ώστε να αυτοπροσαρμόζεται στις εκάστοτε ανάγκες, όχι μόνο από εικόνα σε εικόνα, αλλά και σε τμήματα μέσα στην εικόνα. Ουσιαστικά θα ήταν ένας αλγόριθμος αλγορίθμων, με υλοποιημένες όλες τις προσεγγίσεις που αποδίδουν βέλτιστα για κάθε περίπτωση και ανάγκη κάτι που είναι πολύ δύσκολο, αν όχι αδύνατο να επιτευχθεί.

Η εργασία αυτή έγινε εν γνώσει όλων των παραπάνω και δεν αποσκοπεί σε καμία περίπτωση να λύσει ολοκληρωτικά το πρόβλημα της παρακολούθησης και εξαγωγής στατιστικών δεδομένων από δορυφορικά δεδομένα. Αυτό που κάνει είναι να υλοποιήσει πειραματικά ένα συνδυασμό μεθόδων, γνωστών εδώ και δεκαετίες σε μεγαλύτερης κλίμακας δεδομένα.



## Κεφάλαιο 3 – Μεθοδολογική προσέγγιση

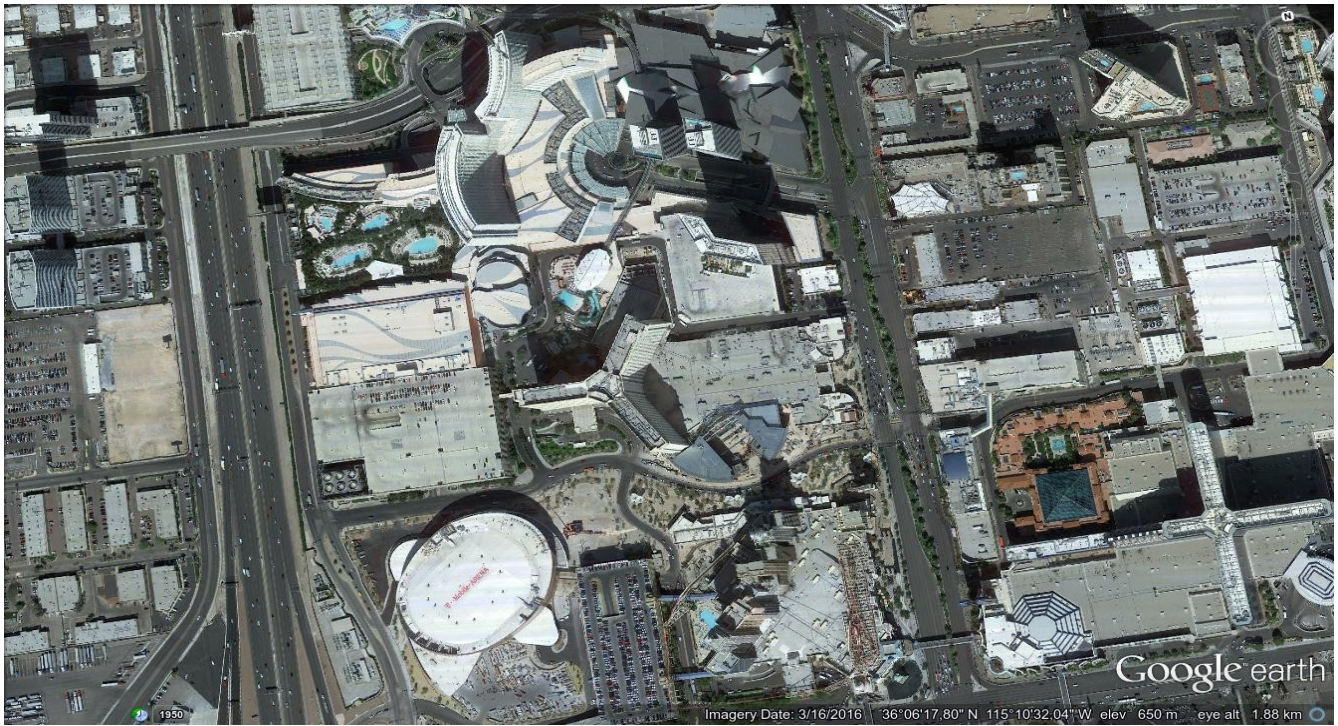
Η όραση υπολογιστών ως επιστήμη απευθύνεται κυρίως σε ηλεκτρολόγους μηχανικούς, ηλεκτρονικούς μηχανικούς, προγραμματιστές και θεωρητικά δεν έχει ιδιαίτερη σχέση με τομείς της τοπογραφίας. Η προσωπική μου άποψη ήταν, ότι με τα μέσα που έχει πλέον στη διάθεση της η σύγχρονη τοπογραφία, αλλά και η τηλεπισκόπηση (πάσης φύσεως δορυφόρους, τελευταίας τεχνολογίας γεωδαιτικούς σταθμούς κ.λ.π.) , η χρήση της όρασης υπολογιστών ως βοηθητικό εργαλείο, θα αποτελέσει καταλυτικό παράγοντα στην επεξεργασία των δεδομένων και μάλιστα δεδομένα μικρής όσο και μεγάλης κλίμακας καθώς πλέον και η ίδια η τοπογραφία αλλά και η τηλεπισκόπηση συλλέγουν ως δεδομένα εκτός των άλλων και εικόνες. Αυτός είναι και ο λόγος που επιλέχθηκε το συγκεκριμένο και όχι κάτι αμιγώς γεωδαιτικό ή κάτι με έντονο τοπογραφικό χαρακτήρα. Η επιλογή αυτή επηρέασε καθοριστικά την μεθοδολογική προσέγγιση που ακολουθήθηκε και παρουσιάζεται στο κεφάλαιο αυτό.

### 3.1 – Εισαγωγή και παρουσίαση περιοχής μελέτης

Τα δεδομένα για την εκπόνηση της εργασίας προέρχονται από ένα δορυφορικό βίντεο υψηλής χωρικής ανάλυσης τραβηγμένο στις 25 Μαρτίου 2014, από δορυφόρο της εταιρείας *Terra Bella*. Η εταιρεία αυτή αποτελεί μετονομασία της εταιρείας *Skybox Imaging*, μια start-up επιχείρηση την οποία η Google αγόρασε το 2014 και μετονόμασε σε *Terra Bella* σύμφωνα με (βλ. <http://www.techcrunch.com/2016/03/08/google-renames-its-satellite-startup-skybox-imaging-to-terra-bella-and-adds-focus-on-image-analysis>).

Περισσότερες πληροφορίες για την εταιρία αυτή μπορούν να βρεθούν στην ιστοσελίδα της ([www.terrabella.google.com](http://www.terrabella.google.com)).

Η απεικονιζόμενη περιοχή αποτελεί τμήμα του Las Vegas, Nevada και φαίνεται στην παρακάτω εικόνα που λήφθηκε από το Google Earth. Έχει διαστάσεις περίπου 850 m x 1500 m όπως μετρήθηκε με το αντίστοιχο εργαλείο που παρέχει το εν λόγω πρόγραμμα, συνεπώς και τα τμήματα των οδικών αξόνων που φαίνονται έχουν περίπου το ίδιο μήκος.



***Εικόνα 3.1 : Η περιοχή μελέτης όπως απεικονίζεται στο Google Earth***

Αντίστοιχα η ίδια περιοχή όπως απεικονίζεται στο πρώτο καρέ του βίντεο φαίνεται στην επόμενη εικόνα:



***Εικόνα 3.2 : Το πρώτο καρέ όπως αυτό εξήχθη από το βίντεο***

Μια σημαντική διαφοροποίηση της περιοχής είναι πως τμήμα του δρόμου που έχει περίπου κάθετο προσανατολισμό στο πρώτο καρέ του βίντεο (Εικόνα 3.2) στο κάτω τμήμα, έχει μετατραπεί πλέον σε καμπύλο όπως φαίνεται στην (Εικόνα 3.1) που είναι τραβηγμένη στις 16/3/2016. Η διαφορά αυτή δεν παίζει κάποιο σημαντικό ρόλο στην μετέπειτα πορεία της εργασίας και αναφέρεται εδώ γιατί αφορά οδικό άξονα που είναι και το τμήμα της εικόνας το οποίο επιχειρείται να αναλυθεί και συγκεντρώνει όλο το ενδιαφέρον.

### **3.2 – Επιλογή της μεθόδου για την αντιμετώπιση του προβλήματος εντοπισμού**

Όπως έχει ήδη αναφερθεί, κύριος στόχος της εργασίας, ήταν να βρεθεί ένας τρόπος, να μπορέσουν να εντοπιστούν από τον υπολογιστή τα οχήματα που κινούνται επί των οδικών αξόνων. Μετά από αρκετή σκέψη και πειραματισμό ως προς το ποια είναι η καταλληλότερη μέθοδος, που να ανταποκρίνεται καλύτερα στις ιδιομορφίες και τις ανάγκες των προς επεξεργασία εικόνων, ως καταλληλότερη μέθοδος εντοπισμού αποφασίστηκε πως ήταν το φίλτρο Kalman. Σε αυτό συνέβαλε το γεγονός ότι το πλήθος των αντικειμένων είναι αρκετά μεγάλο, ότι το φίλτρο Kalman δεν έχει πολύπλοκους και χρονοβόρους υπολογισμούς, αλλά και το γεγονός ότι η κίνηση των οχημάτων μπορεί να προσεγγιστεί από το μοντέλο της ευθύγραμμης ομαλής κίνησης. Δεδομένης της κλίμακας της περιοχής, μπορεί να περιγραφεί από ένα πολύ απλό φυσικό μοντέλο, πράγμα που καθιστά την επιλογή του φίλτρου πολύ ελκυστική, καθώς όπως θα αναλυθεί παρακάτω, η αντιπροσωπευτική μοντελοποίηση ενός φαινομένου για το φίλτρο Kalman είναι πολύ σημαντικός παράγοντας για τη σωστή λειτουργία του. Επίσης δύο ακόμα βασικοί λόγοι είναι, ότι μπορεί να λειτουργήσει σε πραγματικό χρόνο λόγω μη χρονοβόρων σχετικά υπολογισμών καθώς και η ευκολία να σχεδιαστεί και να υλοποιηθεί χωρίς να απαιτείται βαθιά γνώση της υποκείμενης θεωρίας.

Σε επόμενες ενότητες θα ακολουθήσει μια εις βάθος θεωρητική εισαγωγή στις βασικές έννοιες του φίλτρου Kalman, στις αρχές που διέπουν τη λειτουργία του και σε κάποιες βασικές παραμέτρους του.

### 3.2.1 – Φίλτρα Kalman : Τι είναι και πώς λειτουργούν

Όταν αναφέρεται κανείς σε φίλτρα Kalman, ουσιαστικά αναφέρεται σε αλγοριθμικές διαδικασίες, που χρησιμοποιούν σειρές μετρήσεων (διακριτών δεδομένων), οι οποίες συσχετίζονται με το χρόνο. Περιέχουν στατιστικό θόρυβο και λοιπές ανακρίβειες και δίνουν ως αποτέλεσμα εκτιμήσεις αγνώστων μεταβλητών, που τείνουν να είναι ακριβέστερες από εκείνες οι οποίες βασίζονται απλά και μόνο σε μια μέτρηση.

Έχουν πάρει το όνομά του από τον Rudolf E. Kalman, έναν από τους πρώτους που ασχολήθηκαν με τη θεωρία αυτή και παρουσίασε έναν ολοκληρωμένο αλγόριθμο. Τα φίλτρα Kalman δεν είναι κάτι καινούριο. Η θεωρία τους πρωτοεμφανίστηκε στα τέλη της δεκαετίας του '50 και αρχές του '60.

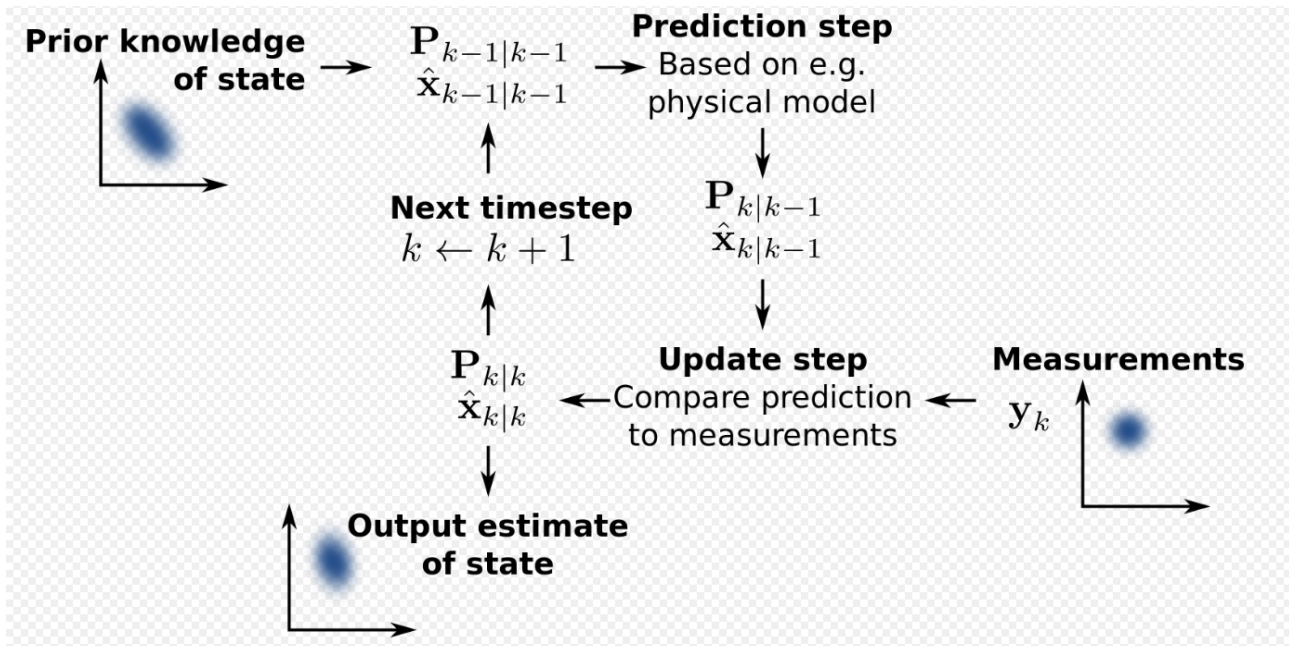
Έκτοτε έχουν βρει εφαρμογή σε πάρα πολλές τεχνολογικές εφαρμογές. Η πιο συνηθισμένη ίσως εφαρμογή του, είναι στην καθοδήγηση, την πλοήγηση και γενικότερα τον έλεγχο οχημάτων, ως επί το πλείστον αεροσκαφών και διαστημικών σκαφών. Επίσης βρίσκουν ευρεία εφαρμογή σε ανάλυση χρονοσειρών, που αφορούν την επεξεργασία σήματος αλλά και την οικονομετρία. Χρησιμοποιούνται ακόμα στην ρομποτική και κυρίως στον έλεγχο της κίνησης και για την βελτιστοποίηση της τροχιάς που θα ακολουθήσει το εκάστοτε αντικείμενο.

Ο αλγόριθμος που υλοποιεί ένα φίλτρο Kalman δουλεύει σε δύο στάδια. Τα στάδια αυτά είναι: το στάδιο της πρόβλεψης και το στάδιο της αναβάθμισης της πρόβλεψης/βελτιστοποίησης.

Στο στάδιο της πρόβλεψης, το φίλτρο παρέχει εκτιμήσεις των μεταβλητών της τρέχουσας κατάστασης, μαζί με την αβεβαιότητα των εκτιμήσεων αυτών. Μόλις επέλθει νέα μέτρηση που αφορά στην αμέσως επόμενη κατάσταση του συστήματος, η οποία αναπόφευκτα θα περιέχει κάποιο σφάλμα, οι εκτιμήσεις αυτές βελτιώνονται με τη χρήση ενός σταθμικού μέσου όρου. Το μεγαλύτερο βάρος δίνεται στις εκτιμήσεις με την μεγαλύτερη αξιοπιστία. Ο αλγόριθμος λειτουργεί αναδρομικά και μπορεί να τρέξει σε πραγματικό χρόνο, χρησιμοποιώντας μόνο τις παρούσες μετρήσεις ως δεδομένα εισόδου και την αμέσως προηγούμενη υπολογισμένη κατάσταση του συστήματος. Απαιτείται επίσης ο πίνακας συμμεταβλητότητάς τους, χωρίς η γνώση των παλαιότερων καταστάσεων να είναι αναγκαία. Το παρακάτω σχήμα περιγράφει οπτικά τη λειτουργία του φίλτρου με απλό και παραστατικό τρόπο.

Για καλύτερη κατανόηση παρατίθεται η λεζάντα της παρακάτω εικόνας όπως αυτή μεταφράστηκε από την αρχική σελίδα-πηγή: “ Το φίλτρο Kalman παρακολουθεί την εκτίμηση/πρόβλεψη της κατάστασης του συστήματος και της μεταβλητότητας αυτής. Η εκτίμηση αυτή αναβαθμίζεται/βελτιώνεται με τη χρήση ενός μεταβατικού μοντέλου ( που περιγράφει τη μετάβαση μεταξύ διαδοχικών καταστάσεων) αλλά και μετρήσεων. Το  $\hat{x}_{k|k-1}$  συμβολίζει την εκτίμηση της κατάστασης του συστήματος τη χρονική στιγμή  $k$  πριν ληφθεί υπ' όψιν η μέτρηση  $y_k$  που αντιστοιχεί και αυτή στη χρονική στιγμή  $k$  . Η μεταβλητή  $P_{k|k-1}$  είναι η αντιστοιχούσα αβεβαιότητα της εκτίμησης αυτής. ”





**Εικόνα 3.3 : Σχηματική αναπαράσταση της λειτουργίας του φίλτρου Kalman**

[Πηγή: Magdi S. Mahmoud, Haris M. Khalid, *“Bibliographic Review on Distributed Kalman Filtering”*, March 15, 2013

Βασικό πλεονέκτημα που του επιτρέπει να λειτουργεί σε πραγματικό χρόνο, είναι η απλότητα των πράξεων του. Χαρακτηριστικότερο παράδειγμα είναι ότι δεν απαιτείται σε κανένα σημείο αντιστροφή των πινάκων, που αφορούν τις εξισώσεις των παρατηρήσεων, πράγμα που είναι μια αρκετά χρονοβόρα διαδικασία, ακόμα και για έναν υπολογιστή, ειδικά όταν τα δεδομένα είναι πολλά και πολυπαραμετρικά. Εκτός μόνο από μία αντιστροφή ενός πίνακα που περιέχει τιμές συμμεταβλητότητας.

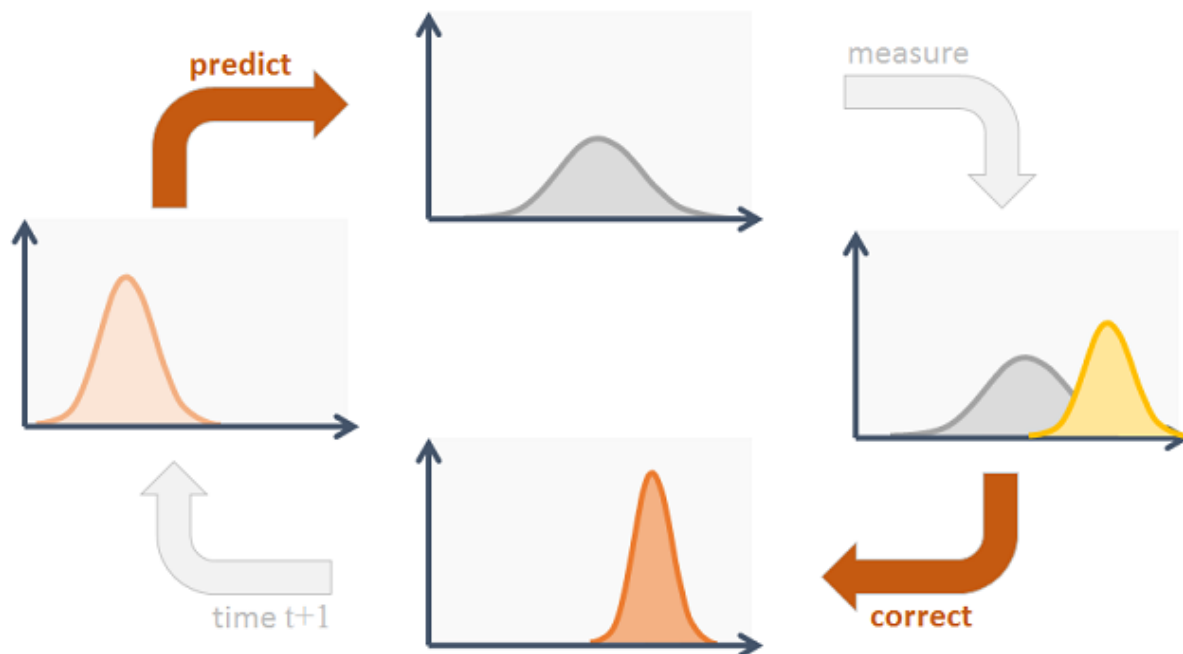
Επιπρόσθετα αναφέρεται πως, το φίλτρο Kalman δεν απαιτεί καμία υπόθεση ότι τα σφάλματα των μετρήσεων υπακούουν στη Γκαουσιανή κατανομή. Στην περίπτωση που αυτή η υπόθεση είναι αληθής, τότε ελαχιστοποιεί το μέσο τετραγωνικό σφάλμα των εκτιμώμενων μεταβλητών.

Εδώ αξίζει να σημειωθεί, ότι ο όρος φίλτρο χρησιμοποιείται κάπως αυθαίρετα αφού ο αλγόριθμος δεν έχει τη δομή ενός κανονικού φίλτρου, π.χ. ενός φίλτρου που ενισχύει τις ακμές μιας εικόνας. Ως εκ τούτου ο όρος “φίλτρο” έχει επικρατήσει λόγω του ότι η διαδικασία εύρεσης της βέλτιστης εκτίμησης από δεδομένα που εξ ορισμού υπόκεινται σε θόρυβο. Η διαδικασία αυτή αποτελεί κατά κάποιο τρόπο φιλτράρισμα του θορύβου των μετρήσεων, με σκοπό την εξάλειψή του.

Στα πλαίσια της αναφοράς στο φίλτρο Kalman, αξίζει να γίνει αναφορά στο Extended Kalman filter (EKF) ή όπως θα μπορούσε να αποδοθεί στα ελληνικά τη μη γραμμική έκδοση του απλού γραμμικού φίλτρου Kalman. Πρόκειται ουσιαστικά για ένα φίλτρο Kalman, το οποίο γραμμικοποιεί γύρω από μια εκτίμηση του τρέχοντος μέσου όρου και της συμμεταβλητότητας της μεταβλητής σε συστήματα μη γραμμικών εξισώσεων. Στα πλαίσια αυτής της εργασίας δεν χρειάστηκε σε κανένα σημείο η υλοποίηση, η χρήση και γενικά η ενασχόληση με το EKF και απλά αναφέρεται εδώ χάριν αναφοράς στη θεωρία περί των φίλτρων Kalman.

### 3.2.2 – Φίλτρα Kalman : Μαθηματική διατύπωση υλοποίηση

Το φίλτρο Kalman συγκροτείται μέσω ενός συστήματος εξισώσεων, που υλοποιούν την πρόβλεψη και βελτιστοποίηση των εκτιμήσεων των μεταβλητών της παρούσας κατάστασης του συστήματος και της αμέσως επόμενης αντίστοιχα. Το παρακάτω σχήμα, δίνει μία απλουστευμένη εικόνα της λειτουργίας του φίλτρου και πιο συγκεκριμένα του αναδρομικού “κύκλου” που το χαρακτηρίζει.



**The cycle of a Kalman filter.** Much of what the Kalman filter does can be reduced to propagating and updating Gaussians and updating their covariances. First the filter predicts the next state from the provided state transition (e.g. motion model), then if applicable, the noisy measurement information is incorporated in the correction phase. The cycle is repeated.

**Εικόνα 3.4: Ο τρόπος λειτουργίας του φίλτρου Kalman**

[Πηγή: <http://www.codeproject.com/Articles/865935/Object-Tracking-Kalman-Filter-with-Ease>]

Ακολουθεί μια συνοπτική αναφορά στο πρακτικό σκέλος της μαθηματικής διατύπωσης του φίλτρου Kalman, η οποία είναι ουσιαστικά η διατύπωση των εξισώσεων που το υλοποιούν. Σημειώνεται ότι δεν θα δοθεί ιδιαίτερη εμβάθυνση στο θεωρητικό υπόβαθρο καθώς αυτό δεν κρίνεται σκόπιμο.

Το φίλτρο Kalman έρχεται να δώσει λύση στο γενικό πρόβλημα της εκτίμησης της κατάστασης  $\mathbf{x} \in \mathbb{R}^n$  μιας διακριτής διαδικασίας η οποία μεταβάλλεται με το χρόνο και υπακούει στη στοχαστική εξίσωση διαφορών:

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_k + \mathbf{w}_k \quad (1)$$

με μέτρηση  $\mathbf{y} \in \mathbb{R}^m$  όπου:

$$\mathbf{y}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \quad (2)$$

Οι τυχαίες μεταβλητές  $\mathbf{w}_k$  και  $\mathbf{v}_k$  αντιπροσωπεύουν τον θόρυβο της διαδικασίας και της μέτρησης αντίστοιχα. Βάσει παραδοχής θεωρούνται ανεξάρτητες η μία με την άλλη, ο θόρυβος λευκός και πως υπακούν στην κανονική κατανομή  $\mathbf{p}(\mathbf{w}) = \mathbf{N}(0, \mathbf{Q})$  και  $\mathbf{p}(\mathbf{v}) = \mathbf{N}(0, \mathbf{R})$ .

Ο τετραγωνικός πίνακας  $\mathbf{A}$  διαστάσεων  $n \times n$  συσχετίζει την κατάσταση της προηγούμενης χρονικής στιγμής με αυτή την παρούσας. Ο πίνακας  $\mathbf{B}$  διαστάσεων  $n \times 1$  συσχετίζει τα προαιρετικά δεδομένα εισόδου  $\mathbf{u} \in \mathbb{R}$  με την κατάσταση  $\mathbf{x}$ . Ο πίνακας  $\mathbf{H}$  διαστάσεων  $m \times n$  συσχετίζει την κατάσταση με την μέτρηση  $\mathbf{y}_k$ . Τέλος τα  $\mathbf{Q}$  και  $\mathbf{R}$  αντιστοιχούν στους πίνακες μεταβλητότητας-συμμεταβλητότητας του θορύβου και της παρατήρησης αντίστοιχα.

Όπως έχει ήδη αναφερθεί, η διαδικασία εφαρμογής του φίλτρου αποτελείται από δύο ξεχωριστά μεν, αλληλένδετα δε στάδια: το στάδιο της πρόβλεψης και το στάδιο της αναβάθμισης της πρόβλεψης/βελτιστοποίησης. Στου στάδιο της πρόβλεψης, προβλέπεται ποια θα είναι η επόμενη κατάσταση του συστήματος με δεδομένο τις μετρήσεις της αμέσως προηγούμενης κατάστασης. Στο στάδιο της αναβάθμισης, εκτιμάται η τρέχουσα κατάσταση του συστήματος με δεδομένο τη μέτρηση εκείνης της χρονικής στιγμής. Οι εξισώσεις που περιγράφουν τα δύο αυτά στάδια παρουσιάζονται παρακάτω.

**Στάδιο πρόβλεψης:**

$$\mathbf{X}_{k^-} = \mathbf{A}_{k-1}\mathbf{X}_{k-1} + \mathbf{B}_k\mathbf{U}_k \quad (3)$$

$$\mathbf{P}_{k^-} = \mathbf{A}_{k-1}\mathbf{P}_{k-1}\mathbf{A}_{k-1}^T + \mathbf{Q}_{k-1} \quad (4)$$

Στάδιο αναβάθμισης της πρόβλεψης:

$$V_k = Y_k - H_k X_k^- \quad (5)$$

$$S_k = H_k P_k^- H_k^T + R_k \quad (6)$$

$$K_k = P_k^- H_k^T S_k^{-1} \quad (7)$$

$$X_k = X_k^- + K_k V_k \quad (8)$$

$$P_k = P_k^- - K_k S_k K_k^T \quad (9)$$

όπου :

- $X_k^-$  και  $P_k^-$  είναι ο μέσος όρος και η συμμεταβλητότητα της κατάστασης, που προβλέπεται για τη χρονική στιγμή k πριν να γίνει γνωστή η μέτρηση.
- $X_k$  και  $P_k$  είναι ο μέσος όρος και η συμμεταβλητότητα της κατάστασης που έχει αναβαθμιστεί/ υπολογιστεί για τη χρονική στιγμή k αφού έχει γίνει γνωστή η μέτρηση.
- $Y_k$  είναι ο μέσος όρος του υπολοίπου της μέτρησης (διαφορά με την πρόβλεψη) τη χρονική στιγμή k.
- $V_k$  είναι ο βελτιωμένος / διορθωμένος μέσος όρος του υπολοίπου της μέτρησης τη χρονική στιγμή k.
- $S_k$  είναι ο πίνακας συμμεταβλητότητας της πρόβλεψης των μετρήσεων τη χρονική στιγμή k.
- $K_k$  είναι το κέρδος του φίλτρου (filter gain), το οποίο υποδηλώνει κατά πόσο οι προβλέψεις θα πρέπει να διορθωθούν τη χρονική στιγμή k.



### 3.2.3 – Χρήση του φίλτρου Kalman για object tracking

Υποθέτοντας ότι ένα αντικείμενο κινείται επίπεδο 2-D, η κατάσταση του τη χρονική στιγμή  $t$  μπορεί να προσδιοριστεί από την θέση του  $[x_{1t}^a \ x_{2t}^a]$  και την ταχύτητά του  $[v_{1t}^a \ v_{2t}^a]$ .

Συνεπώς και πάλι υπό την υπόθεση ότι δεν χρησιμοποιείται κάποιο σήμα ελέγχου (ο πίνακας  $B$  είναι μηδενικός), το μοντέλο με το οποίο μπορεί να περιγραφεί η κίνηση του αντικειμένου είναι το παρακάτω:

$$\begin{pmatrix} x_{1t}^a \\ x_{2t}^a \\ v_{1t}^a \\ v_{2t}^a \end{pmatrix} = \begin{pmatrix} 1 & 0 & \Delta & 0 \\ 0 & 1 & 0 & \Delta \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_{1,t-1}^a \\ x_{2,t-1}^a \\ v_{1,t-1}^a \\ v_{2,t-1}^a \end{pmatrix} + \begin{pmatrix} w_{1t}^a \\ w_{2t}^a \\ w_{3t}^a \\ w_{4t}^a \end{pmatrix} \quad (10)$$

όπου το  $\Delta$  αντιπροσωπεύει την περίοδο δειγματοληψίας.

Με την υπόθεση τώρα ότι υπάρχει δυνατότητα παρατήρησης της θέσης  $\begin{pmatrix} y_{1t}^a & y_{2t}^a \end{pmatrix}$  του αντικειμένου, αλλά όχι της ταχύτητάς του, η παρατηρούμενη θέση υπόκειται σε Γκαουσιανό θόρυβο, και συνεπώς μοντελοποιείται ως εξής:

$$\begin{pmatrix} y_{1t}^a \\ y_{2t}^a \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{1t}^a \\ x_{2t}^a \\ v_{1t}^a \\ v_{2t}^a \end{pmatrix} + \begin{pmatrix} v_{1t}^a \\ v_{2t}^a \\ v_{3t}^a \\ v_{4t}^a \end{pmatrix} \quad (11)$$

Για την εκτέλεση των επιμέρους σταδίων του φίλτρου Kalman, της πρόβλεψης και της αναβάθμισής της, πρέπει να προσδιοριστεί η αρχική πεποίθηση σχετικά με την κατάσταση του αντικειμένου την πρωταρχική χρονική στιγμή, ή με άλλα λόγια την αρχική κατάσταση του συστήματος. Αφού γίνει αυτό, σύμφωνα με τις εξισώσεις (3) – (9), μπορούν πλέον να γίνουν προβλέψεις και εκτιμήσεις όσον αφορά τις διαδοχικές καταστάσεις του συστήματος.

Στην περίπτωση που είναι επιθυμητό να εντοπιστούν πολλαπλά αντικείμενα η μαθηματική διατύπωση είναι παρεμφερής. Στην περίπτωση που είναι επιθυμητό να εντοπιστούν δύο αντικείμενα π.χ τα αντικείμενα  $\alpha, \beta$ . Το σύστημα μπορεί να μοντελοποιηθεί ως εξής:

$$\begin{pmatrix} X_{1t}^\alpha \\ X_{2t}^\alpha \\ v_{1t}^\alpha \\ v_{2t}^\alpha \\ X_{1t}^\beta \\ X_{2t}^\beta \\ v_{1t}^\beta \\ v_{2t}^\beta \end{pmatrix} = \begin{pmatrix} 1 & 0 & \Delta & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & \Delta & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & \Delta & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & \Delta \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X_{1,t-1}^\alpha \\ X_{2,t-1}^\alpha \\ v_{1,t-1}^\alpha \\ v_{2,t-1}^\alpha \\ X_{1,t-1}^\beta \\ X_{2,t-1}^\beta \\ v_{1,t-1}^\beta \\ v_{2,t-1}^\beta \end{pmatrix} + \begin{pmatrix} w_{1t}^\alpha \\ w_{2t}^\alpha \\ w_{3t}^\alpha \\ w_{4t}^\alpha \\ w_{1t}^\beta \\ w_{2t}^\beta \\ w_{3t}^\beta \\ w_{4t}^\beta \end{pmatrix} \quad (12)$$

ενώ οι παρατηρούμενες θέσεις των αντικειμένων μοντελοποιούνται ως εξής:

$$\begin{pmatrix} y_{1t}^\alpha \\ y_{2t}^\alpha \\ y_{1t}^\beta \\ y_{2t}^\beta \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} X_{1,t}^\alpha \\ X_{2,t}^\alpha \\ v_{1,t}^\alpha \\ v_{2,t}^\alpha \\ X_{1,t}^\beta \\ X_{2,t}^\beta \\ v_{1,t}^\beta \\ v_{2,t}^\beta \end{pmatrix} + \begin{pmatrix} v_{1t}^\alpha \\ v_{2t}^\alpha \\ v_{3t}^\alpha \\ v_{4t}^\alpha \\ v_{1t}^\beta \\ v_{2t}^\beta \\ v_{3t}^\beta \\ v_{4t}^\beta \end{pmatrix} \quad (13)$$

Ακριβώς η ίδια μεθοδολογία ακολουθείται, όταν κανείς θέλει να εφαρμόσει το φίλτρο Kalman και για παραπάνω από δύο αντικείμενα, δηλαδή και στην περίπτωση της εργασίας αυτής όπου το πλήθος των αντικειμένων είναι λίγες εκατοντάδες.

Η αντιστροφή του πίνακα  $S$  που χρειάζεται να γίνει στην εξίσωση (7) αποτελεί το μεγαλύτερο υπολογιστικό “αγκάθι” κατά την εκτέλεση του αλγορίθμου. Παρ'όλα αυτά, λόγω του τρόπου που διατυπώνεται το πρόβλημα, οι πίνακες είναι πολύ αραιοί με συνέπεια αυτή η αντιστροφή να μπορεί να πραγματοποιηθεί με έναν υπολογιστικά πολύ αποδοτικό τρόπο.

Όλα όσα έχουν αναφερθεί μέχρι τώρα σχετικά με την επιλεγείσα μεθοδολογία, περιορίζονται στο θεωρητικό επίπεδο και χωρίς ιδιαίτερη εμβάθυνση καθώς σκοπός της εργασίας δεν ήταν τόσο η εις βάθος κατανόηση της φιλοσοφίας του εν λόγω φίλτρου, όσο η προσπάθεια να εφαρμοστεί για τον εντοπισμό πολλαπλών αντικειμένων, που η κίνησή τους μπορεί να περιγραφεί από ένα μοντέλο κίνησης αρκετά ικανοποιητικά.

Για το λόγο αυτό στην επόμενη ενότητα θα γίνει λόγος για το πρακτικό σκέλος της εργασίας αυτής, δηλαδή επί του αλγόριθμου αυτού καθ' αυτόν, όπως αυτός σχεδιάστηκε και υλοποιήθηκε.

### 3.3 – Υλοποίηση του αλγορίθμου εντοπισμού σε προγραμματιστικό περιβάλλον

#### 3.3.1 – Εισαγωγικά

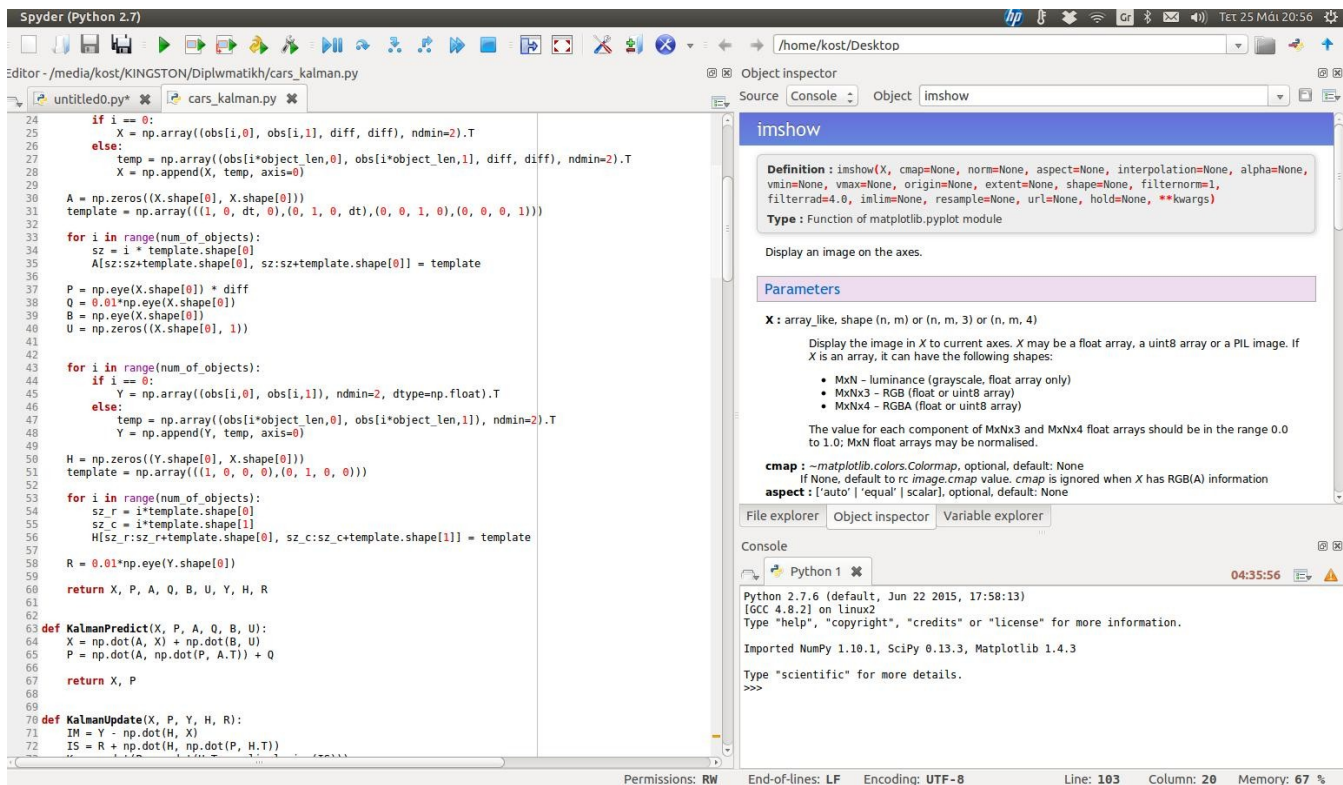
Όπως ήδη προαναφέρθηκε, στην ενότητα αυτή θα παρουσιαστεί ο αλγόριθμος που δημιουργήθηκε και θα εξηγηθεί η λειτουργία του. Αυτό θα γίνει τμηματικά, καθώς είναι τέτοιος ο τρόπος που έχει δομηθεί ώστε να προσφέρεται για τμηματική εξήγηση. Κατ' αυτόν τον τρόπο είναι ευκολότερη η κατανόηση της δομής αλλά κυρίως της λειτουργίας του.

Είναι γνωστό ότι υπάρχουν πολλά προγραμματιστικά περιβάλλοντα για την υλοποίηση αλγορίθμων. Κάθε απόπειρα υλοποίησης ενός συγκεκριμένου αλγόριθμου, θα πρέπει να προηγείται η επιλογή του κατάλληλου περιβάλλοντος προγραμματισμού, της γλώσσας προγραμματισμού. Κάθε περιβάλλον έχει τα δικά του πλεονεκτήματα και αντίστοιχα τους δικούς του περιορισμούς. Συνεπώς η ορθή επιλογή του -κατά συνθήκη- σωστού περιβάλλοντος είναι καθοριστικής σημασίας για την εξέλιξη του όλου εγχειρήματος.

Στο πλαίσιο αυτής της εργασίας, η γλώσσα προγραμματισμού που επιλέχθηκε είναι η Python. Οι λόγοι που κατέληξε αυτή ως μια ικανοποιητική γλώσσα, είναι ότι πρόκειται για μια γλώσσα προγραμματισμού υψηλού επιπέδου, δηλαδή πιο κοντά στην ανθρώπινη γλώσσα και όχι τόσο κοντά στη γλώσσα μηχανής. Αυτό κάνει τη γλώσσα πολύ πιο προσιτή σε κάποιον που δεν έχει ασχοληθεί εις βάθος με τον προγραμματισμό. Είναι επίσης δυναμική γλώσσα, το οποίο σημαίνει ότι πολλές λειτουργίες που μια στατική γλώσσα εκτελεί κατά τη μεταγλώττισή της σε γλώσσα μηχανής (compiling), η Python όντας δυναμική, τις εκτελεί κατά τη διάρκεια που τρέχει το πρόγραμμα. Αυτή η ιδιότητα είναι ζωτικής σημασίας καθώς η εν γένει αναδρομική λειτουργία του φίλτρου Kalman, θα καθιστούσε ιδιαίτερα κοπιαστική, αν όχι αδύνατη την υλοποίηση του εν λόγω αλγορίθμου σε μια στατική γλώσσα όπως π.χ. η C.

Η φιλοσοφία του σχεδιασμού της Python εξ αρχής, ήταν η αναγνωσιμότητα του κώδικα και η ευκολία να περιγράψει κανείς ενέργειες με λιγότερες γραμμές κώδικα, απ' ότι π.χ. σε αντίστοιχες γλώσσες, όπως C++ και Java. Ένα εξίσου σημαντικό χαρακτηριστικό που συνέβαλε στην επιλογή αυτή, είναι η ύπαρξη μεγάλης κοινότητας προγραμματιστών σε Python. Αυτό κάνει την λύση τεχνικών δυσκολιών σχετικά με τη γλώσσα, μια αρκετά εύκολη διαδικασία και μη χρονοβόρα.

Το ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) στο οποίο έγινε η ανάπτυξη του κώδικα, είναι το Spyder μια εφαρμογή ανοιχτού κώδικα για προγραμματισμό σε Python. Πληροφορίες μπορεί να βρει κανείς εδώ: [https://en.wikipedia.org/wiki/Spyder\\_\(software\)](https://en.wikipedia.org/wiki/Spyder_(software)) . Το λειτουργικό σύστημα στο οποίο εκπονήθηκε η εργασία είναι το Linux Ubuntu 14.04 LTS. Ακολουθεί ένα στιγμιότυπο από το IDE για να δοθεί μια εικόνα του περιβάλλοντος εκπόνησης του αλγορίθμου.



Εικόνα 3.5: Στιγμιότυπο από το περιβάλλον του Spyder

Στην αμέσως επόμενη ενότητα παρουσιάζεται ο κώδικας αυτούσιος, ενώ παράλληλα αναλύεται και επεξηγείται η λειτουργία του. Σε περίπτωση που κάποιος δεν είναι εξοικειωμένος με τον προγραμματισμό ή την Python ειδικότερα, ενδεχομένως το επόμενο κεφάλαιο να παρουσιάσει κάποια δυσκολία ως προς την κατανόηση. Σίγουρα όμως θα σχηματίσει μια γενική εικόνα από την επεξήγηση για τη λειτουργία του κώδικα και θα κατανοήσει τις γενικές αρχές της.

Στην αντίθετη περίπτωση που υπάρχει εξοικείωση του αναγνώστη με προγραμματιστικά περιβάλλοντα και γενικότερα με τις βασικές δομές και λειτουργίες του προγραμματισμού, οι επόμενες ενότητες θα αποτελέσουν μια κατατοπιστική εικόνα για την κατανόηση όλων των παραμέτρων του αλγορίθμου και της δομής του.

### 3.3.2 – Παρουσίαση, επεξήγηση και ανάλυση του κώδικα : Βοηθητικές συναρτήσεις

Ο τρόπος που έχει δομηθεί ο αλγόριθμος είναι ο εξής: Υπάρχει το κυρίως πρόγραμμα και οι επιμέρους συναρτήσεις, τις οποίες χρησιμοποιεί το κυρίως πρόγραμμα για να δώσει το ζητούμενο αποτέλεσμα. Όλα τα τμήματα του κώδικα είναι απολύτως απαραίτητα για την σωστή λειτουργία του κώδικα, σαφώς αλληλένδετα μεταξύ τους και αλληλοσυμπληρούμενα.

Αφού παρουσιαστούν οι επιμέρους συναρτήσεις και αναλυθεί η λειτουργία τους, θα ακολουθήσει και η ανάλυση του κυρίως προγράμματος. Εξηγούνται πρώτα οι συναρτήσεις, ούτως ώστε να είναι ήδη γνωστή η λειτουργία τους, όταν φτάσει να εξηγηθεί το κυρίως πρόγραμμα που τις καλεί.

Πρώτα παρουσιάζεται η συνάρτηση `load_coords`:

```
1  def load_coords(current_dir):
2      images = list()
3      final_coords = list()
4      for item in sorted(os.listdir(current_dir)):
5          ext = os.path.splitext(item)
6          print item
7          if ext[1] in item:
8              image = cv2.imread(os.path.join(current_dir, item), 0)
9              if image is not None:
10                 images.append(image)
11
12                 coord = np.where(image>0) #dinei apotelesma tuple
13                 coord_a=np.asarray(coord) #kanw to tuple array
14                 b=coord_a.T #anastrofos tou array
15                 coord_final = np.fliplr(b)
16
17                 final_coords.append(coord_final)
18
19  return images, final_coords
```

Η λειτουργία της παραπάνω συνάρτησης είναι να παίρνει το directory στο οποίο βρίσκονται τα δυαδικά καρέ και να επιστρέφει μια λίστα με τις διαβασμένες εικόνες ταξινομημένες με τη σωστή σειρά (αύξουσα) και μια λίστα με τις συντεταγμένες (σε pixel) όλων των αντικειμένων σε κάθε καρέ που έχει δοθεί. Ουσιαστικά η μεταβλητή `final_coords` που εμπεριέχει τις συντεταγμένες είναι μια λίστα με λίστες. Κάθε λίστα της λίστας αντιστοιχεί στις συντεταγμένες όλων των αντικειμένων που υπάρχουν σε κάθε καρέ.

Ακολουθεί η συνάρτηση `DistCalculation`:

```
1 def DistCalculation(X, ground_truth, current_frame, threshold=10.0):
2     X_reshape = np.reshape(X, (-1,4))
3     points = list()
4     indices = list()
5     for i in range(num_of_cars):
6         point = X_reshape[i,0:2]
7         points.append(point)
8
9         min_dist = threshold
10        min_index = -1
11        for j in range(ground_truth[current_frame].shape[0]):
12            dist = cdist(np.reshape(point, (1,2)), np.array(ground_truth[current_frame][j,:], ndmin=2), "euclidean")
13            if dist < min_dist:
14                min_dist = dist
15                min_index = j
16
17        indices.append(min_index)
18
19    return points, indices
```

Ως δεδομένα εισόδου παίρνει τον πίνακα `X` με τις προβλέψεις του αλγορίθμου, τη λίστα με τις λίστες των αληθών θέσεων των αντικειμένων, όπως προέκυψαν από την αφαίρεση υποβάθρου, για όλα τα καρέ (μεταβλητή `ground_truth`), το τρέχον καρέ με τη μορφή ακεραίου αριθμού, και την τιμή ανοχής, της οποίας η μεταβολή είναι προαιρετική και έχει προεπιλεγμένη τιμή τα 10 pixel. Επίσης γίνεται η παραδοχή πως για τον υπολογισμό της μικρότερης απόστασης, το αντικείμενο δεν εξαφανίζεται .

Όπως υποδηλώνει και το όνομά της, η συγκεκριμένη συνάρτηση έχει ως σκοπό να υπολογίζει μια συγκεκριμένη απόσταση. Η απόσταση αυτή είναι η απόσταση της θέσης που προβλέπεται από τον αλγόριθμο (πρόβλεψη) για τη χρονική στιγμή  $t$ , από την πραγματική θέση που είναι γνωστή την ίδια χρονική στιγμή. Η απόσταση αυτή πρέπει να είναι εντός μιας ανοχής. Η ανοχή μπορεί να μεταβληθεί κατά τη διάρκεια κλήσης της συνάρτησης, αλλά σε περίπτωση που δεν οριστεί από το χρήστη, ορίζεται αυτόματα με μια default τιμή 5 pixel. Η τιμή αυτή επιλέχθηκε μετά από πειραματισμό και εκτίμηση των συνθηκών λειτουργίας του αλγορίθμου.

Ο λόγος που υπολογίζει την απόσταση, είναι για να ταυτοποιείται το αντικείμενο στην αλληλουχία των καρτέ, με την θεώρηση ότι είναι το αντικείμενο εκείνο που βρίσκεται πιο κοντά στην πρόβλεψη του αλγορίθμου. Καταχωρείται δηλαδή ως ζητούμενο αντικείμενο, αυτό που έχει την ελάχιστη απόσταση από την πρόβλεψη.

Ο ακριβής τρόπος λειτουργίας είναι ο εξής: Υπολογίζει την απόσταση μεταξύ του κάθε σημείου (θέση), όπως έχει προκύψει απ' την πρόβλεψη του φίλτρου, με όλες τις πραγματικές γνωστές θέσεις που υπάρχουν σε ένα καρτέ και κρατάει μια λίστα με τα σημεία, για τα οποία έχει πραγματοποιηθεί πρόβλεψη. Κρατάει δηλαδή το prediction και έναν πίνακα ακόμα που περιέχει τα index numbers (αύξοντες αριθμούς στη λίστα με τις πραγματικές θέσεις για κάθε εικόνα) των σημείων, για τα οποία η εκάστοτε θέση που προβλέπει ο αλγόριθμος, παρουσιάζει την ελάχιστη απόσταση για κάθε αντικείμενο πάντα τηρουμένου του ορίου ανοχής.

Επόμενη συνάρτηση είναι η συνάρτηση LoadCars:

```
1 def LoadCars(num_of_cars, ground_truth, car_list):
2     if num_of_cars == 1:
3         observations = np.reshape(ground_truth[0][car_list[0],:], (1,2))
4     else:
5         car_0 = ground_truth[0][car_list[0],:]
6         car_1 = ground_truth[0][car_list[1],:]
7         observations = np.reshape( np.concatenate((car_0,car_1)), (2,2))
8
9     current_car = 2
10
11     while current_car < num_of_cars:
12         car_temp = ground_truth[0][car_list[current_car],:]
13         observations = np.concatenate((observations, np.reshape((car_temp), (1,2))), axis=0)
14         current_car = current_car + 1
15
16     return observations
```



Η συνάρτηση αυτή αποσκοπεί στην παροχή των λεγόμενων παρατηρήσεων (μεταβλητή *observations* εδώ, αλλά στη βιβλιογραφία συχνά αναφέρεται ως *measurements*) στον αλγόριθμο. Οι παρατηρήσεις αυτές δεν είναι παρά οι αληθείς συντεταγμένες (στο πρώτο καρέ) των αντικειμένων που ζητείται να εντοπιστούν.

Ως δεδομένα εισόδου η συνάρτηση αυτή χρειάζεται το πλήθος των αντικειμένων που ζητείται να εντοπιστούν, τη λίστα με τις λίστες των αληθών θέσεων (*ground\_truth*) και τέλος χρειάζεται τη λίστα των προς παρακολούθηση οχημάτων, η οποία είναι μια λίστα με τους αύξοντες αριθμούς (*index numbers*) των αντικειμένων όπως αυτά υπάρχουν στο πρώτο καρέ του σετ δεδομένων.

Σημειώνεται εδώ πως αυτή είναι η συνάρτηση που δίνει τις αρχικές τιμές, από εκεί δηλαδή όπου αντλείται ένα μέρος των δεδομένων που είναι απαραίτητα για την αρχικοποίηση του φίλτρου Kalman.

Ο τρόπος λειτουργίας αυτής της συνάρτησης είναι σχετικά απλός. Αφού της δοθούν τα παραπάνω δεδομένα δημιουργεί έναν πίνακα (*observations*) στον οποίο υπάρχουν οι συντεταγμένες των προς παρακολούθηση αντικειμένων. Ο πίνακας συμπληρώνεται επαναληπτικά με τις συντεταγμένες κάθε αντικειμένου να προστίθενται στο τέλος του πίνακα, μέχρις ότου ο πίνακας συγκεντρώσει τις συντεταγμένες όλων των αντικειμένων που έχουν ζητηθεί.

Οι συναρτήσεις που παρουσιάστηκαν μέχρι τώρα δεν υλοποιούν το φίλτρο Kalman αυτό καθ' αυτό. Αποτελούν βοηθητικές συναρτήσεις που είτε παρέχουν δεδομένα στο φίλτρο είτε εκτελούν κάποια δευτερεύουσα διαδικασία (χωρίς φυσικά να μειώνεται η χρησιμότητά τους, αφού χωρίς αυτές ο κώδικας είναι ημιτελής και δεν μπορεί να λειτουργήσει).

Σε αντίθεση με αυτές, οι συναρτήσεις που θα παρουσιαστούν παρακάτω είναι η καρδιά του αλγορίθμου, ή με άλλα λόγια η υλοποίηση του φίλτρου. Είναι τρεις επιμέρους συναρτήσεις και ονομάζονται *KalmanInit*, *KalmanPredict* και *KalmanUpdate*, αντίστοιχα.

### 3.3.3 – Παρουσίαση, επεξήγηση και ανάλυση του κώδικα : Βασικές συναρτήσεις

Πρώτα παρουσιάζεται η συνάρτηση *KalmanInit* η οποία είναι υπεύθυνη για την αρχικοποίηση του φίλτρου:

```
1 def KalmanInit(obs, dt, num_of_objects, object_len):
2     print '... Kalman initialization'
3
4     diff = 0.1
5
6     for i in range(num_of_objects):
7         if i == 0:
8             X = np.array((obs[i,0], obs[i,1], diff, diff), ndmin=2).T
9         else:
10            temp = np.array((obs[i*object_len,0], obs[i*object_len,1], diff, diff), ndmin=2).T
11            X = np.append(X, temp, axis=0)
12
13    A = np.zeros((X.shape[0], X.shape[0]))
14    template = np.array(((1, 0, dt, 0),(0, 1, 0, dt),(0, 0, 1, 0),(0, 0, 0, 1)))
15
16    for i in range(num_of_objects):
17        sz = i * template.shape[0]
18        A[sz:sz+template.shape[0], sz:sz+template.shape[0]] = template
19
20    P = np.eye(X.shape[0]) * diff
21    Q = 0.01*np.eye(X.shape[0])
22    B = np.eye(X.shape[0])
23    U = np.zeros((X.shape[0], 1))
24
25
26    for i in range(num_of_objects):
27        if i == 0:
28            Y = np.array((obs[i,0], obs[i,1]), ndmin=2, dtype=np.float).T
29        else:
```

```

30     temp = np.array((obs[i*object_len,0], obs[i*object_len,1]), ndmin=2).T
31     Y = np.append(Y, temp, axis=0)
32
33     H = np.zeros((Y.shape[0], X.shape[0]))
34     template = np.array(((1, 0, 0, 0),(0, 1, 0, 0)))
35
36     for i in range(num_of_objects):
37         sz_r = i*template.shape[0]
38         sz_c = i*template.shape[1]
39         H[sz_r:sz_r+template.shape[0], sz_c:sz_c+template.shape[1]] = template
40
41     R = 0.01*np.eye(Y.shape[0])
42
43     return X, P, A, Q, B, U, Y, H, R

```

Η συγκεκριμένη συνάρτηση είναι υπεύθυνη για την αρχικοποίηση του φίλτρου, η οποία πραγματοποιείται μια φορά σε κάθε εκτέλεση του αλγορίθμου. Αρχικοποιεί όλους τους πίνακες που είναι απαραίτητοι για την λειτουργία του φίλτρου και για τους οποίους έχει γίνει λόγος στη θεωρητική επεξήγηση του φίλτρου Kalman (βλ. ενότητα 3.2.2).

Ως δεδομένα εισόδου δέχεται τις παρατηρήσεις/μετρήσεις όπως αυτές δίνονται από την προηγούμενη συνάρτηση που αναλύθηκε, την περίοδο δειγματοληψίας (dt) -για την οποία θα γίνει εκτενέστερη αναφορά στην πορεία-, το πλήθος των αντικειμένων που ζητείται να εντοπιστούν και μια παράμετρο που ουσιαστικά αντιπροσωπεύει το “μήκος του αντικειμένου”, ορίζεται ως 1 και δεν μεταβάλλεται καθ' όλη την εκτέλεση του αλγορίθμου.

Στη συνέχεια αναφέρεται αναλυτικά ο τρόπος λειτουργίας της και εξηγείται πώς ορίστηκαν κάποιες τιμές.

Αρχικά, η τιμή *diff* είναι η αβεβαιότητα των πρωτογενών μετρήσεων, όπως αυτές παρέχονται από το δορυφόρο. Ουσιαστικά πρόκειται για το πόσο καλά θεωρείται ότι ανταποκρίνεται στην πραγματικότητα η εικόνα και δεν υποφέρει από σφάλματα απεικόνισης κ.λ.π. , δηλαδή το κατά πόσο θεωρούνται αξιόπιστες οι μετρήσεις. Επειδή δεν συντρέχει κανένας λόγος αμφισβήτησης της αξιοπιστίας των μετρήσεων, η τιμή αυτή ορίστηκε 0.1 , που φανερώνει μεγάλη εμπιστοσύνη στις μετρήσεις. Η παράμετρος αυτή υπεισέρχεται στον πίνακα *X*, ο οποίος κατά την αρχικοποίηση περιέχει τις αρχικές θέσεις των αντικειμένων και την παράμετρο αυτή, όπως φαίνεται και

στον κώδικα. Στη συνέχεια, κατά την αναδρομική λειτουργία του αλγορίθμου μεταβάλλεται συνεχώς (ο πίνακας) παράγοντας τα ζητούμενα αποτελέσματα. Ακολουθεί η αρχικοποίηση του πίνακα  $X$  που έχει προαναφερθεί και του  $A$ , ο οποίος είναι ο πίνακας που συσχετίζει την κατάσταση της προηγούμενης χρονικής στιγμής, με αυτή την παρούσας και στον οποίο υπεισέρχεται η περίοδος δειγματοληψίας ( $dt$ ).

Εν συνεχεία δημιουργείται ο πίνακας συμμεταβλητότητας  $P$  (ή πίνακας βαρών), ο οποίος είναι ένας διαγώνιος πίνακας με την τιμή της αβεβαιότητας ( $diff$ ) στην κύρια διαγώνιο. Το γεγονός ότι έχουν όλες οι μετρήσεις την ίδια αβεβαιότητα, υποδηλώνει την παραδοχή, πως πρόκειται για ισοβαρείς παρατηρήσεις, για το οποίο συμπέρασμα δεν απαιτείται ιδιαίτερη σκέψη, αφού πρόκειται για μέτρηση συντεταγμένων σε εικόνα που προέρχεται από την ίδια πηγή.

Έπειτα σχηματίζονται και αρχικοποιούνται:

Οι πίνακες  $Q$  και  $R$  που είναι οι πίνακες συμμεταβλητότητας του θορύβου και συμμεταβλητότητας των μετρήσεων/ παρατηρήσεων αντίστοιχα οι οποίοι έχουν την τιμή 0.01 μόνο στην κύρια διαγώνιο. Η τιμή αυτή έχει επιλεγεί αυθαίρετα ως το 1/10 της αβεβαιότητας της μέτρησης.

Ο πίνακας  $B$  που συσχετίζει τα προαιρετικά δεδομένα εισόδου με την κατάσταση  $X$ . Πρόκειται ουσιαστικά για έναν πίνακα ο οποίος περιέχει συντελεστές (βάρη), που αντιστοιχούν στα προαιρετικά δεδομένα εισόδου κάθε μέτρησης, δηλαδή καθορίζει το πόσο θα επηρεάσουν τα δεδομένα αυτά, το μοντέλο. Στην περίπτωση που δεν είναι γνωστό το αν υπάρχει διαφοροποίηση στο βαθμό που επηρεάζουν τα συγκεκριμένα δεδομένα εισόδου ή ακόμα και από προεπιλογή, ο πίνακας αυτός είναι μοναδιαίος, όπως και στα πλαίσια της εργασίας αυτής.

Ο πίνακας  $U$  των επιπλέον (προαιρετικών) δεδομένων ελέγχου που ενδεχομένως υπάρχουν. Στην περίπτωση που μελετάται δεν υπάρχει κάποιο τέτοιου είδους δεδομένο και συνεπώς ο πίνακας  $U$  είναι ένας μηδενικός πίνακας. Αυτό πρακτικά σημαίνει ότι ο όρος  $BU$  που υπάρχει στην εξίσωση (3) είναι μηδενικός, καθώς και πως όποιες τιμές και να δοθούν στον πίνακα  $B$ , δεν επηρεάζουν καθόλου το τελικό αποτέλεσμα από τη στιγμή που δεν υπάρχουν επιπλέον δεδομένα εισόδου για να συσχετιστούν με τα αρχικά.

Ο πίνακας  $H$  που συσχετίζει την κατάσταση στην οποία βρίσκονται τα αντικείμενα, με τις αντίστοιχες μετρήσεις που υπάρχουν για αυτά. Είναι ο αντίστοιχος πίνακας με εκείνον που περιέχει τα νούμερα στην εξίσωση (13) και το μέγεθος του εξαρτάται άμεσα από το πλήθος των αντικειμένων που ζητείται να εντοπιστούν.

Αυτοί είναι όλοι οι πίνακες που πρέπει να δημιουργηθούν και να αρχικοποιηθούν για να μπορέσει εν συνεχεία ο αλγόριθμος να εκτελέσει τις απαιτούμενες λειτουργίες, της πρόβλεψης και της βελτιστοποίησής της. Είναι προφανές πως κάποιιοι από τους πίνακες αυτούς, μεταβάλλονται σε κάθε επανάληψη που εκτελεί ο αλγόριθμος λόγω της αναδρομικής ιδιότητάς του. Κάποιοι άλλοι παραμένουν αμετάβλητοι.

Εφ' όσον λοιπόν δημιουργηθούν και πάρουν τις κατάλληλες τιμές οι παραπάνω πίνακες, τότε και μόνο τότε είναι δυνατό να τρέξουν οι συναρτήσεις *KalmanPredict* και *KalmanUpdate*, οι οποίες παρουσιάζονται ευθύς αμέσως.

Η συνάρτηση *KalmanPredict*:

```
1 def KalmanPredict(X, P, A, Q, B, U):
2     X = np.dot(A, X) + np.dot(B, U)
3     P = np.dot(A, np.dot(P, A.T)) + Q
4
5     return X, P
```

Η συνάρτηση αυτή είναι ουσιαστικά η υλοποίηση των εξισώσεων (3) και (4) και υλοποιεί το στάδιο της πρόβλεψης, για το οποίο αναφορά στο πώς λειτουργεί θεωρητικά γίνεται στην ενότητα 3.2.2. Πρόκειται για την εκτέλεση της πρόβλεψης της επόμενης κατάστασης του συστήματος πριν ο αλγόριθμος λάβει υπ' όψιν τη μέτρηση που αντιστοιχεί σε κείνη τη στιγμή για την οποία προβλέπει.

Η συνάρτηση *KalmanUpdate*:

```
1 def KalmanUpdate(X, P, Y, H, R):
2     IM = Y - np.dot(H, X)
3     IS = R + np.dot(H, np.dot(P, H.T))
4     K = np.dot(P, np.dot(H.T, np.linalg.inv(IS)))
5     X = X + np.dot(K, IM)
6     P = P - np.dot(K, np.dot(IS, K.T))
7
8     return X, P
```

Όπως και στην προηγούμενη συνάρτηση, έτσι και εδώ πρόκειται για την υλοποίηση των εξισώσεων (5) – (9) που εκτελούν το στάδιο της βελτίωσης της αρχικής πρόβλεψης/αναβάθμισης της πρόβλεψης. Οι γραμμές 2-6 του κώδικα όπως απεικονίζονται στο παραπάνω τμήμα αντιστοιχούν μια προς μία στις θεωρητικές εξισώσεις

(5) – (9) όπως αυτές παρουσιάζονται στην ενότητα 3.2.2. Η εκτέλεση της συνάρτησης αυτής υλοποιεί ουσιαστικά τη διαδικασία αναβάθμισης της εκτίμησης του αλγορίθμου, πάνω στην ήδη γνωστή πρόβλεψη. Αναβαθμίζει την ήδη υπάρχουσα εκτίμηση λαμβάνοντας υπ' όψιν την μέτρηση της χρονικής στιγμής, για την οποία παρέχεται τελικά η βελτιωμένη εκτίμηση.

Στη συνέχεια παρουσιάζεται ο κύριος κώδικας ο οποίος κάνει χρήση των προαναλυθεισών συναρτήσεων και εξάγει τα τελικά αποτελέσματα τα οποία παρουσιάζονται στο επόμενο κεφάλαιο.

Ο κώδικας παρουσιάζεται συνολικά παρακάτω αλλά η επεξήγηση του γίνεται τμηματικά για καλύτερη κατανόηση.

Κύριος κορμός του κώδικα:

```
1  current_dir = '/media/kost/KINGSTON/Diplwmatikh/Data/Ana_1'
2  images, ground_truth = load_coords(current_dir)
3
4
5  dt = 1
6  object_len = 1
7  num_of_objects = len(ground_truth[0])
8
9  print "Give number of cars to be tracked.\nNumber must be between 0 and %d." %num_of_objects
10
11
12  num_of_cars = int(raw_input("Type a number of cars to be tracked or 0 to cancel.\nOnly integers must be typed:"))
13
14  if num_of_cars == 0:
15      sys.exit("Process canceled")
16
17  while num_of_cars < 1 or num_of_cars > num_of_objects:
18      print "Cars must be between 1 and %d. \nTry again." %num_of_objects
19      num_of_cars = int(raw_input("Give number of cars to be tracked:"))
20
21  randspec = raw_input("Type 'r' for random objects or 's' for specified objects.\nIf something else than 'r' or 's' given thi
22
23  if randspec == 's':
```

```

24 input_list = raw_input("Give index number of cars to be tracked.\nNumbers must be integers and given separated by s
25 input_list = input_list.split()
26 index_list = map(int, input_list)
27 if len(index_list) != num_of_objects:
28     while len(index_list) != num_of_cars:
29         print "Indexes must be the same number as number of cars"
30         input_list = raw_input("Give index number again separated by space:")
31         input_list = input_list.split()
32         index_list = map(int, input_list)
33
34 else:
35     index_list = list(random.sample(range(0, num_of_objects), num_of_cars))
36
37
38
39 observations = LoadCars(num_of_cars, ground_truth, index_list)
40
41 X, P, A, Q, B, U, Y, H, R = KalmanInit(observations, dt, num_of_cars, object_len)
42
43 X, P = KalmanPredict(X, P, A, Q, B, U)
44
45 #####
46
47 max_frames = len(ground_truth)
48 current_frame = 1
49 size = 50
50 states = list()
51 all_indices=list()
52 all_indices.append(index_list)
53
54 while current_frame < len(images):
55     X_reshape = np.reshape(X, (-1,4))
56     points, indices = DistCalculation(X, ground_truth, current_frame, threshold=5)
57     all_indices.append(indices)

```

```

58 X_reshape = np.reshape(X, (-1,4))
59
60 Y = np.zeros((num_of_cars,2), np.float)
61 for i in range(num_of_cars):
62     if indices[i] == -1:
63         Y[i,:] = X_reshape[i,0:2]
64     else:
65         Y[i,:] = ground_truth[current_frame][indices[i,:]]
66
67 Y = np.reshape(Y, (num_of_cars*2,1))
68
69 X, P = KalmanUpdate(X, P, Y, H, R)
70
71 state = np.reshape(X, (-1,4))
72 state = state[:,0:2]
73 states.append(state)
74
75
76
77 fig = plt.figure(figsize=(18.5, 12.0))
78 plt.imshow(images[current_frame], cmap="Greys_r")
79 Y_temp = np.reshape(Y, (num_of_cars,-1))
80 for c in range(num_of_cars):
81     plt.scatter(state[c,0], state[c,1], s=100, c='c', marker='x')
82     plt.scatter(Y_temp[c,0], Y_temp[c,1], s=100, c='m', marker='+')
83 plt.imshow(images[current_frame], cmap="Greys_r")
84
85 plt.axis('off')
86 plt.savefig('Frame' + str(current_frame) + '.png', dpi = 72, bbox_inches='tight', pad_inches=0)
87
88
89 X, P = KalmanPredict(X, P, A, Q, B, U)
90
91

```



```

92     current_frame = current_frame + 1
93
94
95     all_dx = list()
96     all_dy = list()
97     all_dist = list()
98
99     for a in range(num_of_cars):
100         for i in range(len(images)-1):
101             dx = states[i][a,0] - ground_truth[i+1][all_indices[i+1][a] ,0]
102             all_dx.append(dx)
103
104             dy = states[i][a,1] - ground_truth[i+1][all_indices[i+1][a] ,1]
105             all_dy.append(dy)
106
107
108             square_dist = (states[i][a,0] - ground_truth[i+1][all_indices[i+1][a] ,0])**2 + (states[i][a,1] - ground_truth[i+1][all
109             dist = square_dist ** 0.5
110             all_dist.append(dist)
111
112     all_dx_array = np.asarray(all_dx)
113     dx_dif = np.reshape(all_dx_array, (9,-1), order = 'F' )
114
115     all_dy_array = np.asarray(all_dy)
116     dy_dif = np.reshape(all_dy_array, (9,-1), order = 'F' )
117
118     all_dist_array = np.asarray(all_dist)
119     dist_final = np.reshape(all_dist_array, (9,-1), order = 'F' )
120
121
122
123
124     pdf = matplotlib.backends.backend_pdf.PdfPages("output.pdf")
125     for fig in xrange(1, figure().number):

```

```

126 pdf.savefig( fig )
127 pdf.close()
128
129
130 np.savetxt('dx.txt', dx_dif, fmt='%-7.4f')
131 np.savetxt('dy.txt', dy_dif, fmt='%-7.4f')
132 np.savetxt('dist.txt', dist_final, fmt='%-7.4f')

```

Στη γραμμή 1 του παραπάνω κώδικα ορίζεται στη μεταβλητή *current\_dir* το directory στο οποίο βρίσκονται οι διαδοχικές εικόνες. Δηλαδή το σετ δεδομένων, πάνω στο οποίο θα τρέξει ο αλγόριθμος. Εν συνεχεία στη γραμμή 2 γίνεται κλήση της συνάρτησης *load\_coords*, η οποία εισάγει τις εικόνες και τις συντεταγμένες στο πρόγραμμα.

- Γραμμές 5-7 : Ορίζονται οι παράμετροι *dt* και *object\_len* και βρίσκεται ο αριθμός των αντικειμένων στο πρώτο καρέ του σετ δεδομένων.

Εδώ πρέπει να σημειωθούν κάποια πράγματα για την παράμετρο *dt*. Όπως έχει προαναφερθεί, η παράμετρος *dt* είναι η περίοδος δειγματοληψίας του αλγορίθμου. Δίνοντας την τιμή 1 επί της ουσίας ο αλγόριθμος παίρνει τα καρέ ανά 1 (στο εκάστοτε σετ δεδομένων) . Όταν επιχειρήθηκε η μεταβολή της τιμής αυτής για πειραματισμό, τα αποτελέσματα δεν μπορούσαν να ερμηνευθούν ικανοποιητικά. Ο λόγος ήταν η ύπαρξη αμφιβολιών για το αν έχει μονάδες η παράμετρος αυτή, καθώς και ποιες είναι αυτές, αφού στις δοκιμές υπήρχαν αλληλοσυγκρουόμενα συμπεράσματα. Λόγω αυτού και σε συνδυασμό με το ότι κρίθηκε αρκετά χρονοβόρα η διαδικασία για την αποσαφήνιση του ρόλου της, αποφασίστηκε να παραμείνει στην προεπιλεγμένη (default) τιμή της. Αναφορά στην παράμετρο αυτή γίνεται και στα συμπεράσματα στο πέμπτο κεφάλαιο.

- Γραμμές 9-35: Στις γραμμές αυτές ο αλγόριθμος κατ' αρχάς ζητάει από το χρήστη, το πλήθος των αντικειμένων που θα εντοπιστούν. Στη συνέχεια δίνει στο χρήστη την επιλογή να επιλέξει αν θέλει εκείνος, τα προς εντοπισμό οχήματα ή αν θέλει να τα επιλέξει ο αλγόριθμος τυχαία για αυτόν. Είναι προφανές ότι ο αριθμός αυτός κυμαίνεται μεταξύ της μονάδας και του συνόλου των αντικειμένων που υπάρχουν στο πρώτο καρέ του κάθε σετ δεδομένων.
- Γραμμές 39-43: Καλείται η συνάρτηση *LoadCars*, η οποία επιστρέφει τον πίνακα των παρατηρήσεων βάσει των αντικειμένων που έχουν επιλεγεί προς παρακολούθηση. Στη συνέχεια καλείται η συνάρτηση *KalmanInit*, η οποία δημιουργεί και αρχικοποιεί τους πίνακες που χρειάζονται, όπως έχει προαναφερθεί

εκτενώς παραπάνω. Στη συνέχεια καλείται η συνάρτηση *KalmanPredict*, η οποία δίνει την πρώτη πρόβλεψη για τα εν λόγω αντικείμενα.

- Γραμμές 47-52: Αρχικά υπολογίζονται το πλήθος των καρτέ και δημιουργείται η μεταβλητή *current\_frame*, που είναι απαραίτητη σε πολλές επαναληπτικές διαδικασίες του κώδικα. Δημιουργούνται επίσης οι λίστες *states*, *all\_indices*, οι οποίες χρησιμεύουν στο να αποθηκεύονται στη συνέχεια συγκεντρωτικά οι καταστάσεις του συστήματος για κάθε καρτέ και οι κωδικοί των αντικειμένων από τις *index lists* των εικόνων για κάθε όχημα. Δηλαδή για κάθε όχημα, είναι γνωστό σε ποια θέση βρίσκεται στην *index list*, που αντιστοιχεί σε κάθε καρτέ.
- Γραμμές 54-92: Το τμήμα αυτό του αλγορίθμου είναι το τμήμα που πρακτικά υλοποιείται το φίλτρο Kalman. Πρόκειται για ένα επαναληπτικό βρόχο ο οποίος λειτουργεί ως εξής:  
Καλείται η συνάρτηση *DistCalculation*, της οποίας η λειτουργία έχει περιγραφεί σε προηγούμενη ενότητα και εντοπίζονται οι πλησιέστερες στο *prediction* του αλγορίθμου πραγματικές θέσεις. Σημειώνεται εδώ, πως αν δεν βρεθεί κάποιο αντικείμενο εντός της ανοχής, τότε ως *ground truth* (μέτρηση) στον πίνακα **Y** μπαίνει το *prediction*.  
Σε κάθε επανάληψη συμπληρώνεται και η λίστα *all\_indices* με τα *index numbers* των αντικειμένων ανά καρτέ. Επίσης δημιουργείται ο πίνακας **Y** που περιέχει τις αληθείς θέσεις, για το τρέχον καρτέ για κάθε αντικείμενο που θεωρείται ότι έχει εντοπιστεί από το προηγούμενο καρτέ, με σκοπό να τις παράσχει στο στάδιο της εκτίμησης.  
Στη συνέχεια καλείται η συνάρτηση *KalmanUpdate* και γίνεται το *update* στους πίνακες **X**, **P**, το οποίο και αποθηκεύεται στη λίστα *states* (κατάσταση του συστήματος) η οποία αναφέρεται και παραπάνω.
- Οι γραμμές 77-86 αφορούν την οπτικοποίηση του αποτελέσματος. Ορίζουν παραμέτρους όπως το μέγεθος του παραθύρου όπου θα εμφανιστούν οι εικόνες με το αποτέλεσμα, τα σύμβολα που θα δείχνουν τις θέσεις των προβλέψεων, τα χρώματα των συμβόλων και λοιπές οπτικές λεπτομέρειες. Έπειτα, και αφού οριστούν όλα αυτά, ο αλγόριθμος σώζει κάθε καρτέ όπως εμφανίζεται κατά την εκτέλεσή του στη θέση που βρίσκεται αποθηκευμένο και το *script* του κώδικα.
- Στη γραμμή 89 επανακαλείται η συνάρτηση *KalmanPredict*, αυτή τη φορά όμως της δίνεται ως είσοδος ο πίνακας **X** που έχει αναβαθμιστεί (*update*). Έτσι πραγματοποιεί την εκτίμηση (*prediction*) για την αμέσως επόμενη χρονική στιγμή και η διαδικασία επαναλαμβάνεται, προχωρώντας μια χρονική στιγμή σε κάθε επανάληψη.

- Τέλος στη γραμμή 92 αυξάνεται κατά 1 η τιμή της *current\_frame*, αφού η διαδικασία έχει προχωρήσει κατά ένα frame, που είναι απαραίτητο για την σωστή λειτουργία του βρόχου και για να μην γίνει ο βρόχος ατέρμων.  
Με αυτόν τον τρόπο εκτιμώνται όλες οι θέσεις για όλα τα οχήματα και υλοποιείται ουσιαστικά το φίλτρο Kalman. Στην συνέχεια, οι γραμμές κώδικα που ακολουθούν δεν αφορούν το φίλτρο, αλλά την εκτίμηση και ποσοτικοποίηση της αξιοπιστίας του αλγορίθμου. Πρόκειται δηλαδή για διαδικασία ελέγχου του αλγορίθμου.
- Γραμμή 95-119: Αρχικά δημιουργούνται 3 λίστες κενές και στη συνέχεια υπολογίζεται το οριζοντιογραφικό σφάλμα της θέσης. Το σφάλμα αυτό είναι η απόσταση της εκτιμώμενης θέσης από την αληθή καθώς επίσης και οι δύο συνιστώσες της κατά  $\chi$  και κατά  $\psi$ , οι οποίες συμπληρώνουν τις λίστες αυτές. Αφού τελειώσει η διαδικασία των υπολογισμών, οι λίστες μετατρέπονται σε πίνακες για χάρη της αναγνωσιμότητας.
- Γραμμές 124-132: Αρχικά αποθηκεύονται σε ένα αρχείο pdf όλα τα καρέ συγκεντρωμένα και με τη σωστή σειρά και στη συνέχεια αποθηκεύονται σε ένα αρχείο txt οι πίνακες με τα σφάλματα που αναφέρθηκαν προηγουμένως σε φορμάτ που να διευκολύνει την ανάγνωση και την παρατήρηση.

Στο σημείο αυτό ολοκληρώνεται η παρουσίαση και η ανάλυση της μεθοδολογίας, που επιλέχθηκε και υλοποιήθηκε προγραμματιστικά, με την οποία πραγματοποιείται ο εντοπισμός των αντικειμένων. Αξίζει να σημειωθεί πως επειδή υπάρχουν πάρα πολλές παράμετροι τόσο εσωτερικές (εντός του αλγορίθμου), όσο και εξωτερικές λόγω της φύσης του προβλήματος και των δεδομένων, στο πλαίσιο της εργασίας αυτής, ενώ αρχικός σκοπός ήταν να δοκιμαστούν πολλά παραπάνω πράγματα και να εξελιχθεί περαιτέρω ο αλγόριθμος, τελικά αυτό δεν κατέστη δυνατό. Αποδείχθηκε δηλαδή στην πράξη, ότι χρειάζονται πολλές δοκιμές και έλεγχοι προκειμένου να δοκιμαστεί η λειτουργία του αλγορίθμου για κάθε πιθανό σενάριο.

Σε κάθε περίπτωση όμως, όπως θα φανεί και στο αμέσως επόμενο κεφάλαιο, το αποτέλεσμα ήταν πολύ θετικό και ελπιδοφόρο με τον αλγόριθμο να λειτουργεί πολύ ικανοποιητικά σε ομαλές εξωτερικές συνθήκες και με τις παραμέτρους του σταθερές.

## Κεφάλαιο 4 – Πειραματικά Αποτελέσματα και Αξιολόγηση

### 4.1 – Παρουσίαση δεδομένων

Στο παρόν κεφάλαιο παρουσιάζονται τα αποτελέσματα των πειραμάτων, που πραγματοποιήθηκαν με τον αλγόριθμο που παρουσιάστηκε στο προηγούμενο κεφάλαιο και ακολουθεί η αξιολόγηση τους, τόσο ποσοτικά όσο και ποιοτικά.

Ως πρωτογενή δεδομένα θεωρούνται τα δεδομένα τα οποία λαμβάνονται χωρίς να έχουν υποστεί καμία επεξεργασία. Εν αντιθέσει ως δεδομένα εισόδου του αλγορίθμου, θεωρούνται τα δεδομένα που δίνονται κατευθείαν στον αλγόριθμο, αφού έχουν υποστεί την κατάλληλη επεξεργασία, για να φτάσουν να αποτελούν δεδομένα εισόδου.

Τα πρωτογενή δεδομένα είναι το βίντεο και κατ' επέκταση τα καρέ που έχουν εξαχθεί απ' ευθείας από το βίντεο, δηλαδή αληθείς εικόνες (true images), όπως οι εικόνα 4.1 που φαίνεται παρακάτω. Πρόκειται για εικόνες ανάλυσης 1280 x 720 pixel σε τόνους του γκρι.

Τα δεδομένα που χρησιμοποιήθηκαν ως δεδομένα εισόδου του αλγορίθμου έχουν εξαχθεί από δορυφορικό βίντεο της Skybox Imaging (πλέον Terra Bella). Το βίντεο υπάρχει στο διαδικτυακό χώρο YouTube και μπορεί κανείς να το παρακολουθήσει στον ακόλουθο σύνδεσμο: <https://www.youtube.com/watch?v=IKNAY5ELUZY> . Πρόκειται για βίντεο τραβηγμένο από δορυφόρο σε υψηλή ανάλυση, πάνω από την περιοχή του Las Vegas. Τα χαρακτηριστικά/ιδιότητες του βίντεο, όπως ακριβώς δίνονται από το πρόγραμμα αναπαραγωγής VLC, είναι τα παρακάτω.

Type: Video

Codec: H264 – MPEG-4 AVC (part10) (avc1)

Resolution: 1280 x 738

Display resolution: 1280 x 720

Frame rate: 30

Decoded format: Planar 4:2:0 YUV

Εδώ πρέπει να σημειωθεί πως, εφ' όσον γίνεται αναφορά σε δορυφορικό βίντεο και ο δορυφόρος βρίσκεται σε τροχιά γύρω απ' τη γη, αυτό θα σήμαινε ότι η οπτική γωνία που φαίνεται η περιοχή θα έπρεπε να μεταβάλλεται συνεχώς αφού παραμένει σταθερό το κάδρο. Ο λόγος που αυτό δεν συμβαίνει είναι ότι η κάμερα του δορυφόρου κινείται με τέτοιο τρόπο ώστε να αντισταθμίζει την παράλλαξη που θα προκαλείτο μεταξύ των εικόνων.

Βάσει φωτογραμμετρίας, αυτό μπορεί να γίνει μόνο για ένα υψομετρικό επίπεδο και για το λόγο αυτό φαίνεται μια συνεχής κίνηση των υψηλών κορυφών των κτιρίων, καθώς λόγω διαφοράς υψομέτρου από το έδαφος δεν είναι δυνατόν να αντισταθμιστεί η παράλλαξη. Το γεγονός ότι η κάμερα καταγράφει το βίντεο σαν να ήταν ακίνητη, (όσον αφορά τουλάχιστον το επίπεδο του δρόμου, που είναι αυτό που συγκεντρώνει όλο το ενδιαφέρον στο πλαίσιο αυτής της εργασίας), διευκολύνει πολύ τη λειτουργία του αλγορίθμου και σίγουρα επηρεάζει θετικά την αξιοπιστία του. Σε αντίθετη περίπτωση θα έπρεπε να ακολουθηθούν περαιτέρω διαδικασίες επεξεργασίας των εικόνων, π.χ. να εφαρμοστεί κάποια τεχνική image registration για να μπορέσει να εκτελεστεί ο εντοπισμός.

Εν τέλει η επεξεργασία που εφαρμόστηκε ήταν η εξής: Από τα αληθιά καρέ έγινε αφαίρεση του υποβάθρου και στη συνέχεια οι εικόνες μετατράπηκαν σε δυαδικές (binary images) με την εφαρμογή μια δυαδικής μάσκας. Η διαδικασία αυτή καθιστά πολύ απλή την ανίχνευση των οχημάτων επί των τελικών επεξεργασμένων εικόνων, οι οποίες και αποτέλεσαν τα δεδομένα εισόδου του αλγορίθμου που εκπονήθηκε. Αμέσως παρακάτω παρουσιάζονται στιγμιότυπα από τη διαδικασία δυαδικοποίησης των αρχικών εικόνων, μέχρι αυτές να αποτελέσουν τα τελικά δεδομένα εισόδου. Στο αμέσως επόμενο καρέ παρουσιάζεται ένα αρχικό αληθές καρέ, όπως αυτό εξήχθη από το βίντεο.

Τα πρωτογενή δεδομένα που έχουν εξαχθεί από το βίντεο έχουν την εξής μορφή:

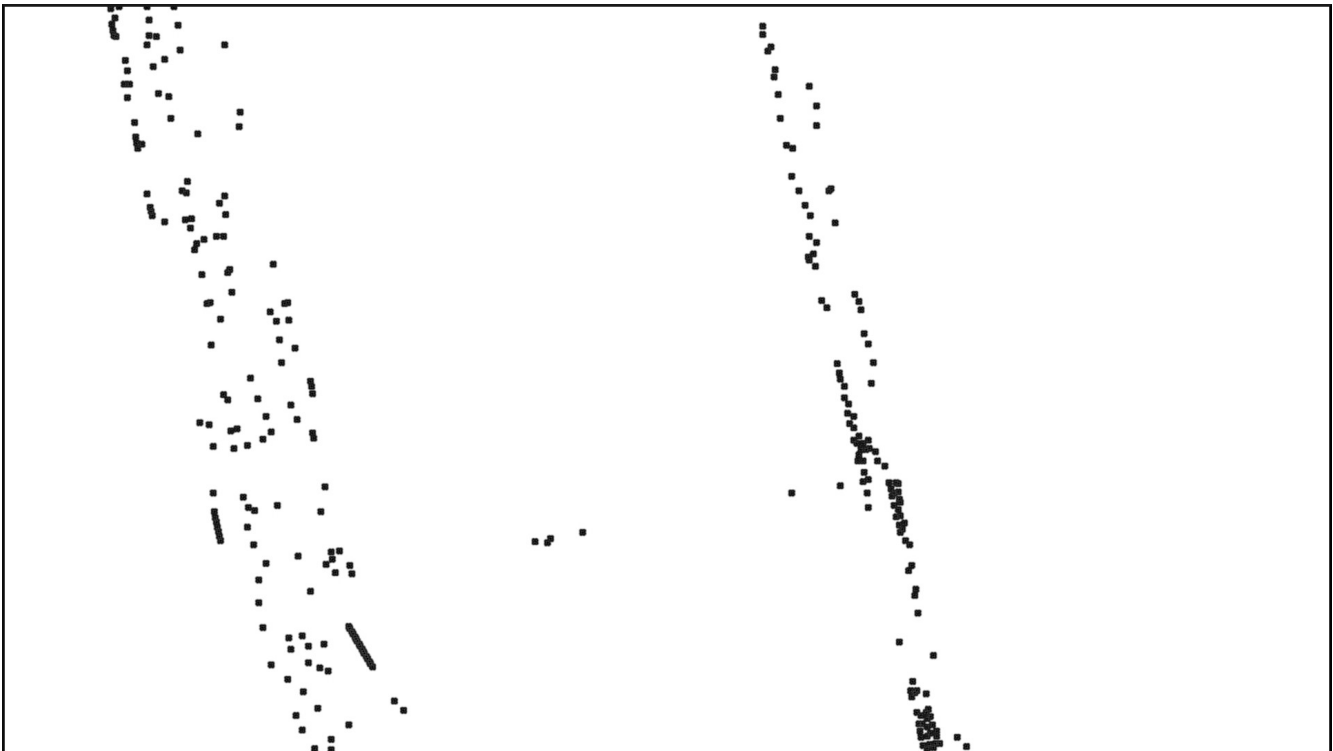


*Εικόνα 4.1 : Καρέ #256 του αρχικού βίντεο*

Τα τμήματα της παραπάνω εικόνας και γενικότερα όλων των εικόνων που συγκεντρώνουν όλο το ενδιαφέρον στο πλαίσιο της εργασίας αυτής, είναι οι δύο μεγάλοι οδικόι άξονες που υπάρχουν σχεδόν κάθετα στην εικόνα. Υπάρχει επίσης ένας πιο μικρός σχεδόν οριζόντιος ως προς το καρέ δρόμος, που βρίσκεται στο κάτω τμήμα της εικόνας. Πιο συγκεκριμένα, τα οχήματα που κινούνται σε αυτούς είναι τα αντικείμενα, γύρω από τα οποία στρέφεται όλη η μελέτη και ως εκ τούτου οι οδικόι αυτοί άξονες ορίζονται ως η περιοχή μελέτης.

Για να καταστεί δυνατό να αποτελέσουν οι εικόνες τα δεδομένα εισόδου στον αλγόριθμο και να ανιχνευτούν τα οχήματα, οι εικόνες όπως έχει ήδη ειπωθεί, μετατράπηκαν σε δυαδικές (binary). Με τη μέθοδο αυτή τα οχήματα - τα οποία εμφανίζονται με τόνους που προσεγγίζουν το λευκό - γίνονται σαφώς διακριτά από το υπόβαθρο τους και μπορεί ο υπολογιστής να τα αναγνωρίζει σε κάθε καρέ με σχετική ευκολία. Η διαδικασία μετατροπής μιας εικόνας σε δυαδική είναι η εξής: Με δεδομένη μια εικόνα σε τόνους του γκρι (grayscale) ορίζεται μια τιμή που αντιπροσωπεύει έναν τόνο του γκρι ως τιμή ενός κατωφλίου. Όποιες τιμές των pixels της εικόνας, υπερβαίνουν την τιμή αυτή παίρνουν την τιμή 1, ενώ σε αντίθετη περίπτωση την τιμή 0. Δημιουργείται δηλαδή μια νέα εικόνα που είναι η δυαδική. Αμέσως παρακάτω παρουσιάζεται μια αρνητική δυαδική εικόνα (για λόγους οπτικούς) όπως αυτή προέκυψε από το αληθές καρέ μετά την αφαίρεση του υποβάθρου.

Μετά την αφαίρεση του υποβάθρου και την μετατροπή τους σε δυαδικές, οι εικόνες είχαν την παρακάτω μορφή:



*Εικόνα 4.2 : Καρέ #45 σε δυαδική μορφή (binary) σε αρνητικό*

Επισημαίνεται ότι παρουσιάζεται το αρνητικό της αρχικής εικόνας με ελαφρώς διογκωμένα τα αντικείμενα για καλύτερη οπτική αντίληψη των δεδομένων από τον αναγνώστη αυτού του τεύχους.

Τα αποτελέσματα του αλγορίθμου από οπτικής σκοπιάς, είναι διαδοχικές εικόνες σε binary μορφή όπως οι παραπάνω με μεταβλητό μεταξύ τους διάστημα ανά σετ εικόνων. Έχουν επαυξηθεί οπτικά με την θέση που ο αλγόριθμος εκτιμάει πως θα βρίσκεται το όχημα, καθώς και με επισημασμένη την πραγματική θέση. Παρήχθησαν δηλαδή αρκετά σετ δεδομένων, με διαφορετικό βήμα διαδοχής των καρέ το καθένα (ανά 1, 5, 10, 15, 20, 30, 50 καρέ) στα οποία δοκιμάστηκε η απόδοση του αλγορίθμου μέχρι να βρεθεί το όριο στο οποίο ο αλγόριθμος έπαψε να λειτουργεί αποτελεσματικά.

## 4.2 – Αποτελέσματα - Ποιοτική αξιολόγηση

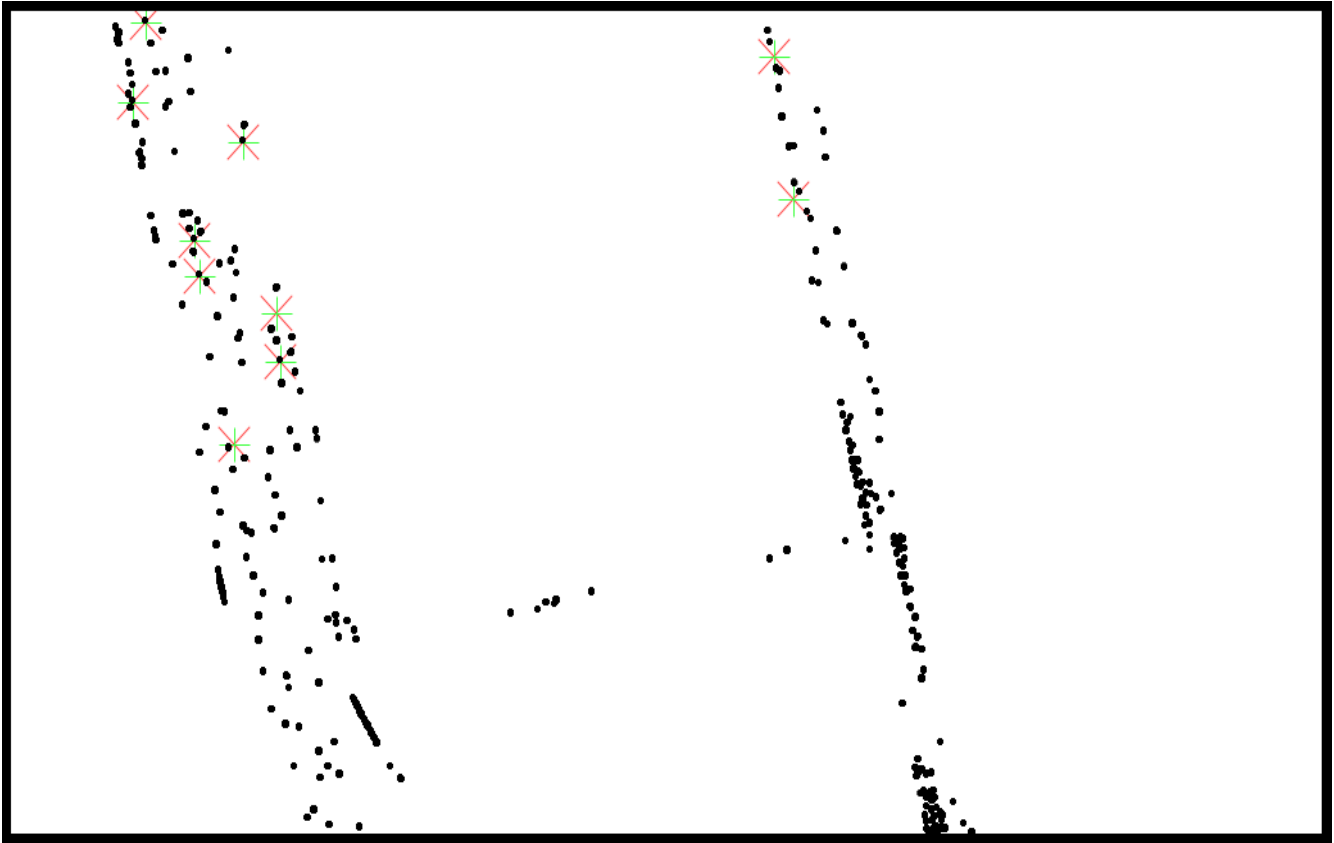
Σε αυτήν την ενότητα παρουσιάζονται αναλυτικά τα αποτελέσματα που προέκυψαν κατά τη διάρκεια της αξιολόγησης του αλγορίθμου. Στόχος της αξιολόγησης ήταν να εξεταστεί, το κατά πόσον μπορεί να γίνει παρακολούθηση των οχημάτων με τη χρήση του φίλτρου Kalman. Ο αλγόριθμος λειτουργεί για εύρος από 1 έως όσων αντικειμένων / οχημάτων εντοπίζονται από τον υπολογιστή στο πρώτο καρέ κάθε σετ δεδομένων.

Πιο αναλυτικά η λειτουργία του αλγορίθμου είναι να λειτουργεί επαναληπτικά  $n-1$  φορές, όπου  $n$  το πλήθος των καρέ του εκάστοτε σετ δεδομένων που του δίνονται ως δεδομένα εισόδου. Σε κάθε επανάληψη δίνει ως προϊόν τις εκτιμώμενες θέσεις (συντεταγμένες), των οχημάτων που έχουν επιλεγθεί εξ αρχής για κάθε επανάληψη. Ο λόγος που τα καρέ είναι  $n-1$  και όχι  $n$ , οφείλεται στο ότι το φίλτρο χρειάζεται αρχικές τιμές (initialization) για να μπορέσει να ξεκινήσει να δίνει εκτιμήσεις και ουσιαστικά να λειτουργήσει, τις οποίες και παίρνει από το πρώτο καρέ.

Για τις ανάγκες της αξιολόγησης έχουν επιλεγθεί 10 αυτοκίνητα, όχι απαραίτητα τα ίδια, για κάθε σετ δεδομένων. Ο λόγος αλλά και τα κριτήρια που επιλέχθηκαν τα εκάστοτε οχήματα για κάθε σετ δεδομένων είναι καθαρά τεχνικός, έτσι ώστε να διασφαλίζει ότι φαίνονται όλα τα οχήματα σε όλα τα καρέ. Ένα ακόμα κριτήριο που συντέλεσε καθοριστικά στην χειροκίνητη επιλογή των οχημάτων ήταν οι απρόσκοπτες συνθήκες κίνησης. Δηλαδή να μην εμποδίζεται είτε από προπορευόμενο όχημα, είτε από οποιοδήποτε άλλο παράγοντα, να μην εξαφανίζεται σε κανένα καρέ κανένα από τα επιλεγμένα οχήματα και να έχει μια σχετικά ομαλή κίνηση. Αυτό συμβαίνει με την συντριπτική πλειονότητα των οχημάτων, δεδομένου και του μεγέθους της απεικονιζόμενης περιοχής και της απόστασης λήψης των δεδομένων. Τα αποτελέσματα έχουν μορφή τόσο γραφική όσο και



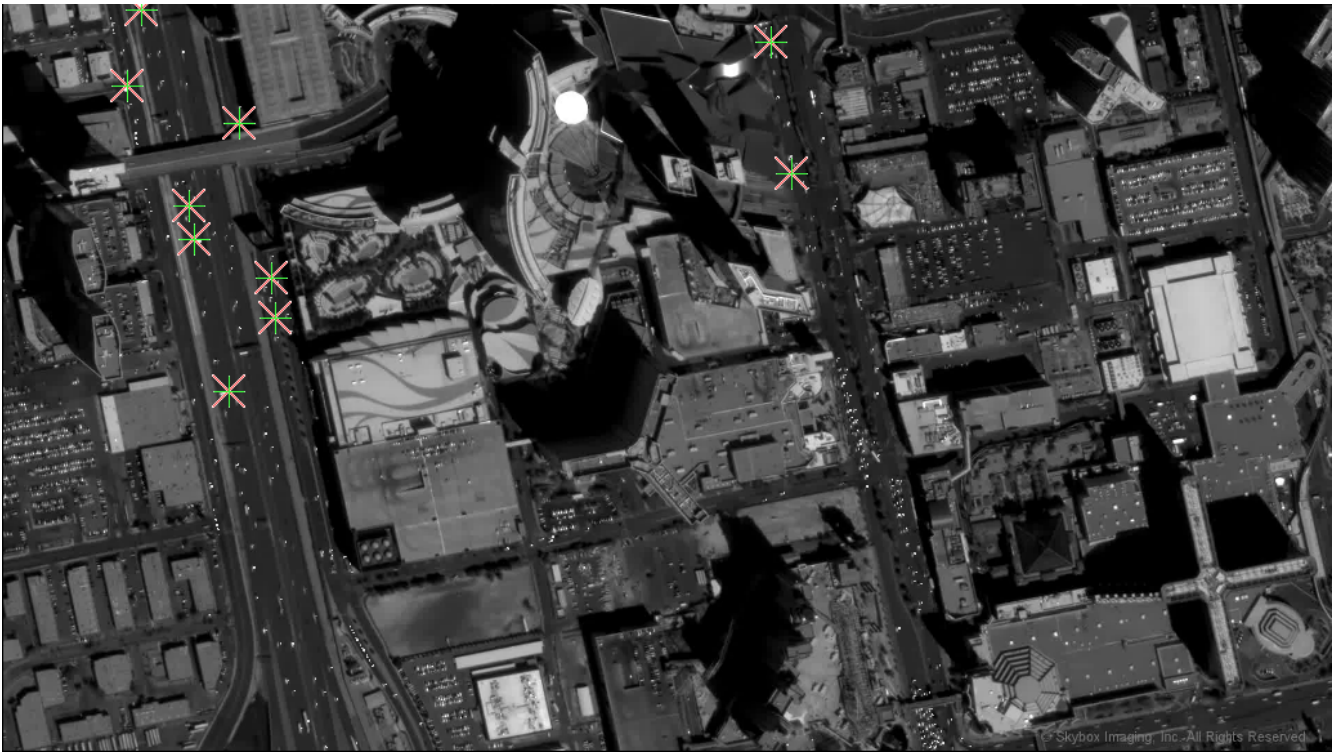
αριθμητική. Το γραφικό / οπτικό αποτέλεσμα παρουσιάζεται παρακάτω και πιο συγκεκριμένα ακολουθεί ένα καρέ από τα δεδομένα εισόδου επαυξημένο με τα αποτελέσματα του αλγορίθμου με γραφικό τρόπο.



*Εικόνα 4.3 : Δεδομένα εισόδου με το αποτέλεσμα του αλγορίθμου (Στιγμιότυπο 1 καρέ)*

Σ' αυτήν την εικόνα με μαύρες κουκκίδες φαίνονται τα αυτοκίνητα και ο αλγόριθμος επισημαίνει με + την πραγματική θέση του οχήματος, που είναι γνωστή εξ' αρχής για κάθε καρέ και με X τη θέση στην οποία το φίλτρο υπολογίζει ότι βρίσκεται το ίδιο όχημα. Στην ουσία η εικόνα αυτή αποτελεί γραφικό έλεγχο της αξιοπιστίας του αλγορίθμου, μιας και γίνεται πολύ εύκολα αντιληπτό με το μάτι, πόσο κοντά βρίσκονται η εκτίμηση με την πραγματική θέση.

Στην αμέσως επόμενη εικόνα φαίνεται το δεύτερο καρέ του βίντεο (αληθής εικόνα), επαυξημένο με το αποτέλεσμα του αλγορίθμου. Εδώ τα pixels που υπήρχαν παραπάνω και αντιπροσώπευαν τα οχήματα έχουν αντικατασταθεί από τα ίδια τα οχήματα αφού ως υπόβαθρο υπάρχει η αληθής εικόνα όπως αυτή έχει εξαχθεί από το βίντεο. Το στιγμιότυπο αυτό δίνει μια πολύ καλύτερη ιδέα σχετικά με το προηγούμενο καθώς η εικόνα είναι φυσική, με την απλή προσθήκη της εκτίμησης και της πραγματικής θέσης με σύμβολα.

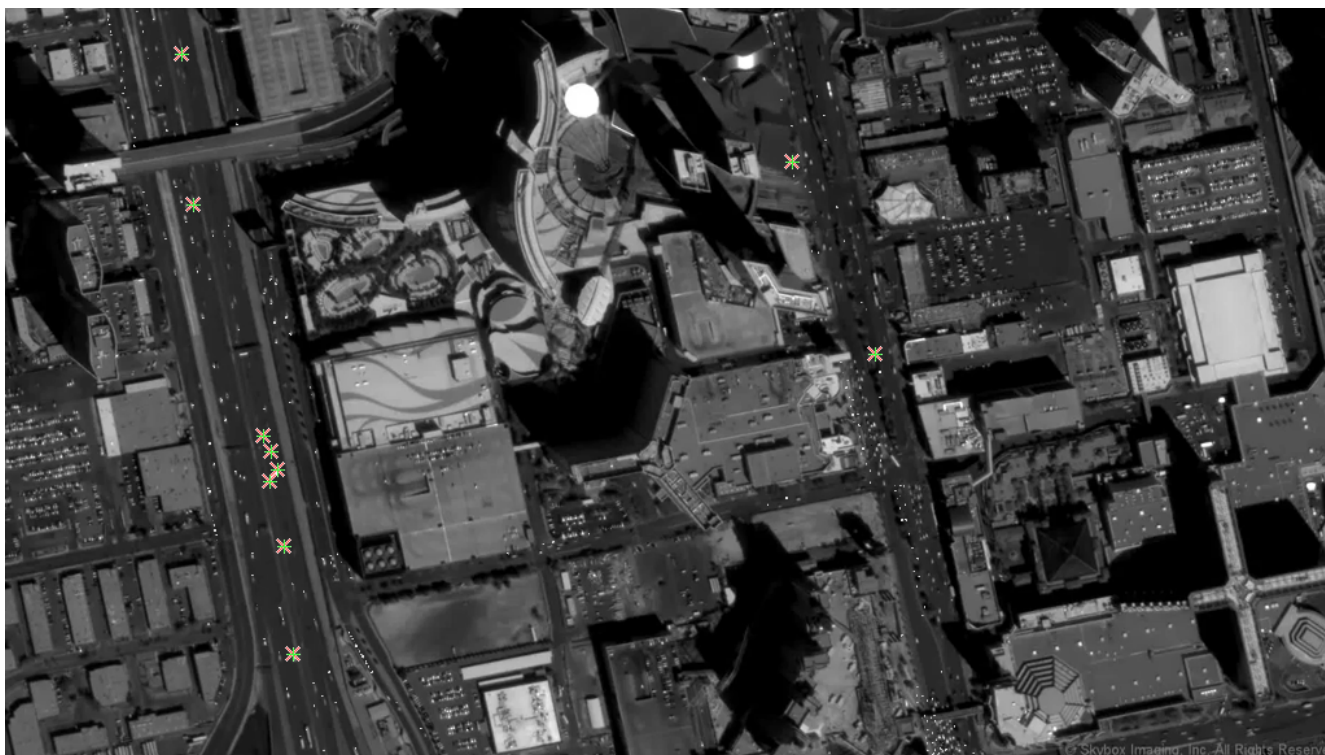


*Εικόνα 4.4 :Καρέ #2 επαυξημένο με το αποτέλεσμα του αλγορίθμου*

Οι εικόνες αυτές (4.4) και οποιοδήποτε παραγωγό τους (π.χ. βίντεο που θα προέκυπτε από τις εικόνες αυτές, περαιτέρω επεξεργασία κ.λ.π.) αποτελούν τον ποιοτικό έλεγχο για την αξιοπιστία του αλγορίθμου. Με βάση το οπτικό αποτέλεσμα, όπως αυτό παρουσιάζεται παραπάνω η ακρίβεια του αλγορίθμου φαίνεται αρκετά ικανοποιητική, λαμβανομένου υπ' όψιν ότι τα δεδομένα έχουν ληφθεί από δορυφόρο. Αυτό το συμπέρασμα ενισχύεται αν αναλογιστεί κανείς την έκταση της περιοχής που απεικονίζεται και το πλήθος των οχημάτων που μπορούν να εντοπιστούν και να παρακολουθηθούν ταυτόχρονα και δυναμικά σε πραγματικό χρόνο.

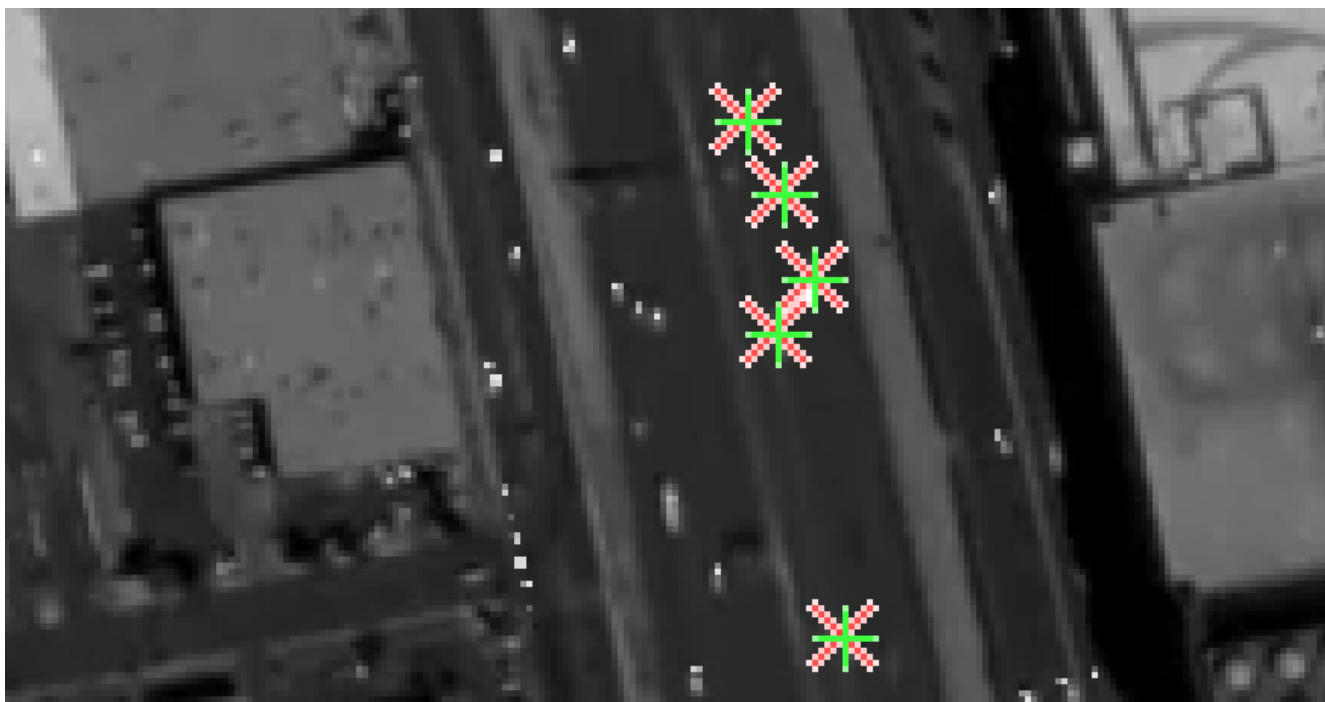
Παρακάτω φαίνονται τα αληθή στιγμιότυπα από κάθε σετ δεδομένων επαυξημένα με το αποτέλεσμα του αλγορίθμου γραφικά, αλλά και τμήματα των στιγμιότυπων αυτών μεγεθυμένα κατά 400% για καλύτερη οπτική αντίληψη της ικανότητας εντοπισμού του αλγορίθμου. Παρουσιάζεται ένα στιγμιότυπο από κάθε σετ δεδομένων και στη συνέχεια ένα τμήμα του σε μεγέθυνση. Η σειρά με την οποία παρατίθενται τα στιγμιότυπα είναι η εξής: σετ δεδομένων με βήμα 1, 5, 10, 15, 20, 30 καρέ. Ο λόγος για τον οποίο δεν υπάρχει οπτικό αποτέλεσμα για το σετ δεδομένων των 50 καρέ εξηγείται στο τέλος της ενότητας αυτής.

Ακολουθεί στιγμιότυπο από το σετ δεδομένων με βήμα 1 καρέ:



*Εικόνα 4.5 : Καρέ #5, από σετ δεδομένων με βήμα 1 καρέ*

Στο επόμενο καρέ φαίνεται η περιοχή που έχει επισημανθεί στην πρώτη εικόνα σε μεγέθυνση 400%:



*Εικόνα 4.6 : Τμήμα του καρέ #5 από σετ δεδομένων με βήμα 1 καρέ*

Παρατηρείται, ότι ακόμα και σε μεγέθυνση της τάξης του 400%, απαιτείται προσεκτική παρατήρηση από τον αναγνώστη για να φανεί η μικρή διαφοροποίηση (σφάλμα του αλγορίθμου) της θέσης του οχήματος ως προς τη θέση που εκτιμά ο αλγόριθμος. Όπως θα φανεί παρακάτω το σφάλμα στην περίπτωση του συγκεκριμένου σετ δεδομένων είναι της τάξης του 10% του pixel και το μέγεθος της ποσότητας αυτής δείχνει το κατά πόσον ο ποιοτικός έλεγχος μπορεί να δώσει ακριβή στοιχεία σχετικά με την απόδοση του αλγορίθμου, καθώς από οθόνη σε οθόνη ή ακόμα και από φορμάτ σε φορμάτ της ίδιας της εικόνας ενδεχομένως να επηρεάζεται η εικόνα και μια τόσο μικρή διαφοροποίηση να στρεβλώνεται. Παρ' όλα αυτά με διατήρηση της μεγέθυνσης πάντα στο ίδιο επίπεδο και του φορμάτ της εικόνας σταθερού, μπορεί να γίνει αισθητή τόσο η αντίληψη, όσο και η διαφοροποίηση του σφάλματος όσο μεταβάλλονται τα σετ δεδομένων.

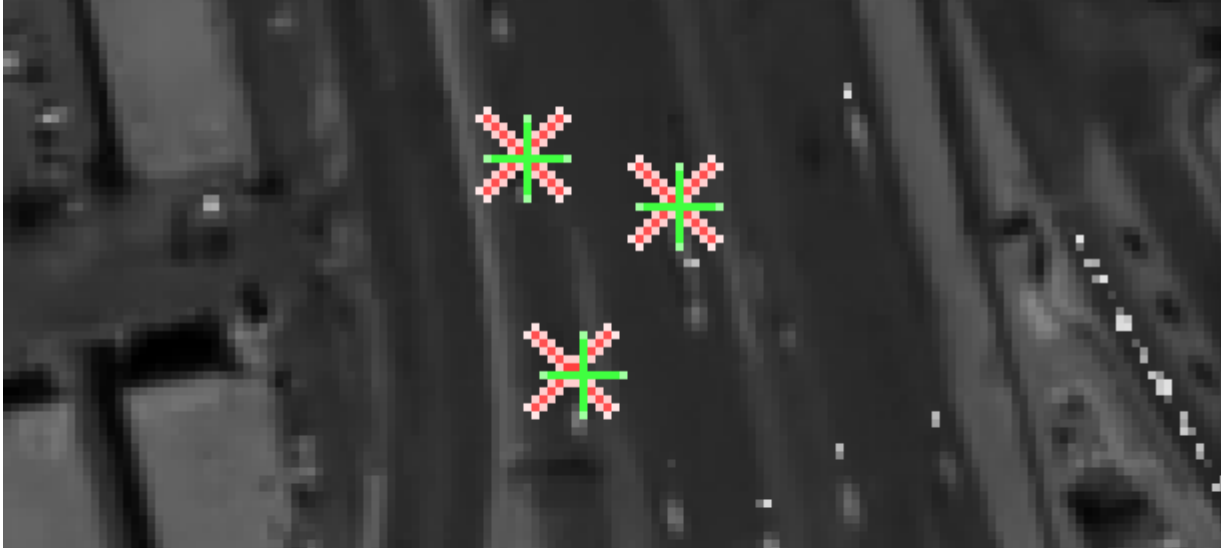
Ακολουθεί στιγμιότυπο από το σετ δεδομένων με βήμα 5 καρέ:



*Εικόνα 4.7 : Καρέ #25, σετ δεδομένων με βήμα 5 καρέ*



Και αντίστοιχα ακολουθεί ένα τμήμα της εικόνας σε ζουμ 400%:



*Εικόνα 4.8 : Τμήμα του καρέ #25 από σετ δεδομένων με βήμα 5 καρέ*

Φαίνεται καθαρά πως το σφάλμα έχει μεγαλώσει ως ένα βαθμό για το συγκεκριμένο σετ δεδομένων, αλλά και πάλι παραμένει σε ανεκτά πλαίσια δεδομένου ότι η εικόνα αυτή είναι σε ζουμ 400%.

Ακολουθεί στιγμιότυπο από το σετ δεδομένων με βήμα 10 καρέ:



*Εικόνα 4.9 : Καρέ #41, σετ δεδομένων με βήμα 10 καρέ*

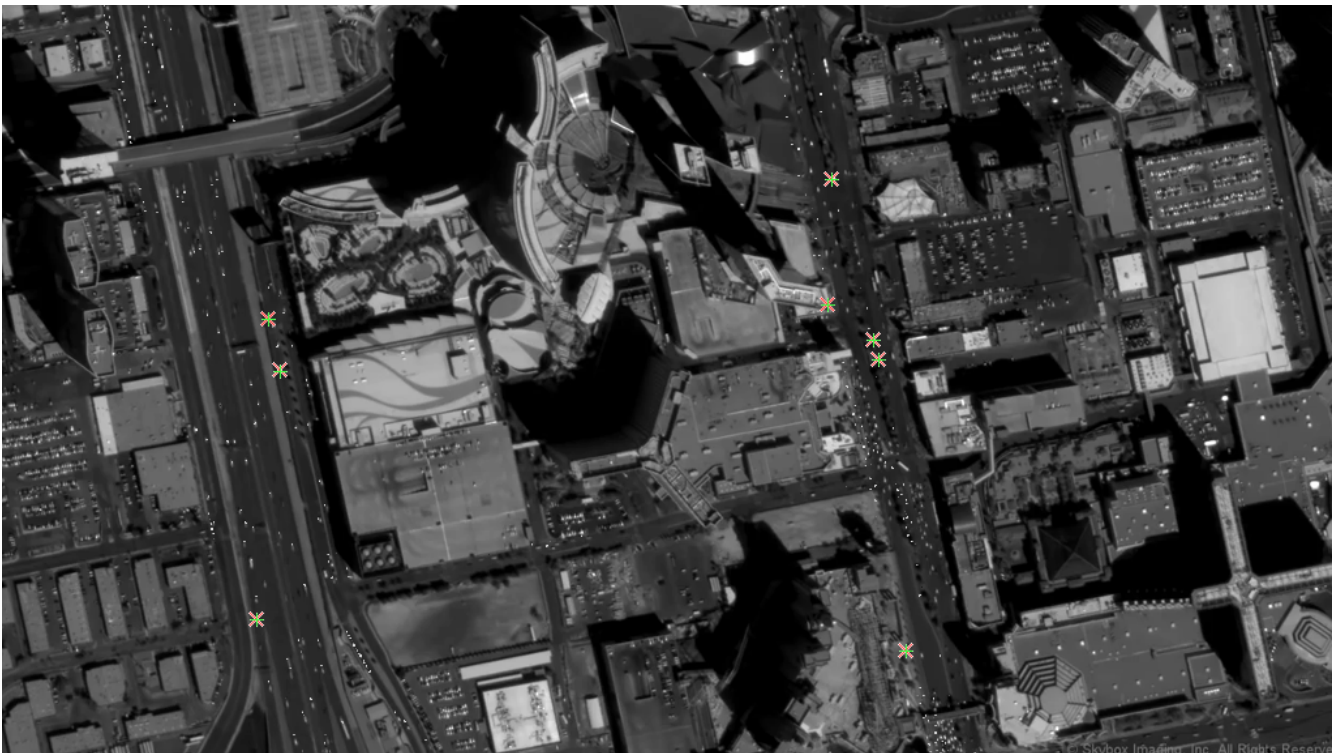
Και αντίστοιχα ακολουθεί ένα τμήμα της εικόνας σε ζουμ 400%:



*Εικόνα 4.10 : Τμήμα του καρτέ #41 από σετ δεδομένων με βήμα 10 καρτέ*

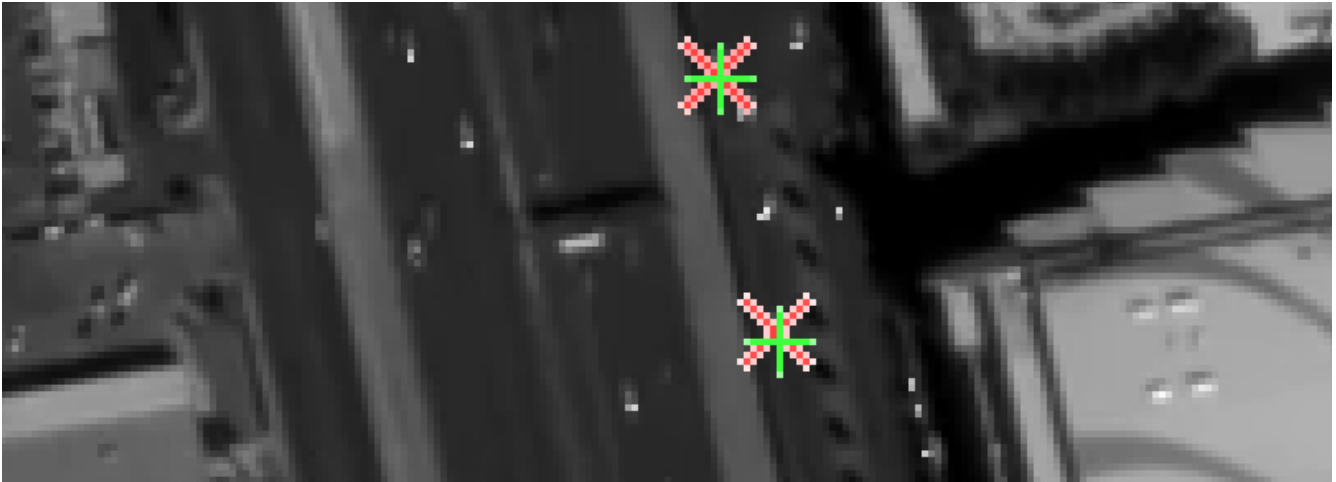
Παρατηρείται εδώ ότι το σφάλμα δεν διαφέρει σχεδόν καθόλου από το αντίστοιχο του προηγούμενου σετ δεδομένων με βήμα 5 καρτέ. Ενδεχομένως να είναι λίγο μεγαλύτερο στην περίπτωση αυτή πράγμα λογικό και αναμενόμενο, αλλά οπτικά δεν είναι δυνατό να διαπιστωθεί με βεβαιότητα.

Ακολουθεί στιγμιότυπο από το σετ δεδομένων με βήμα 15 καρτέ:



*Εικόνα 4.11 : Καρτέ #61, σετ δεδομένων με βήμα 15 καρτέ*

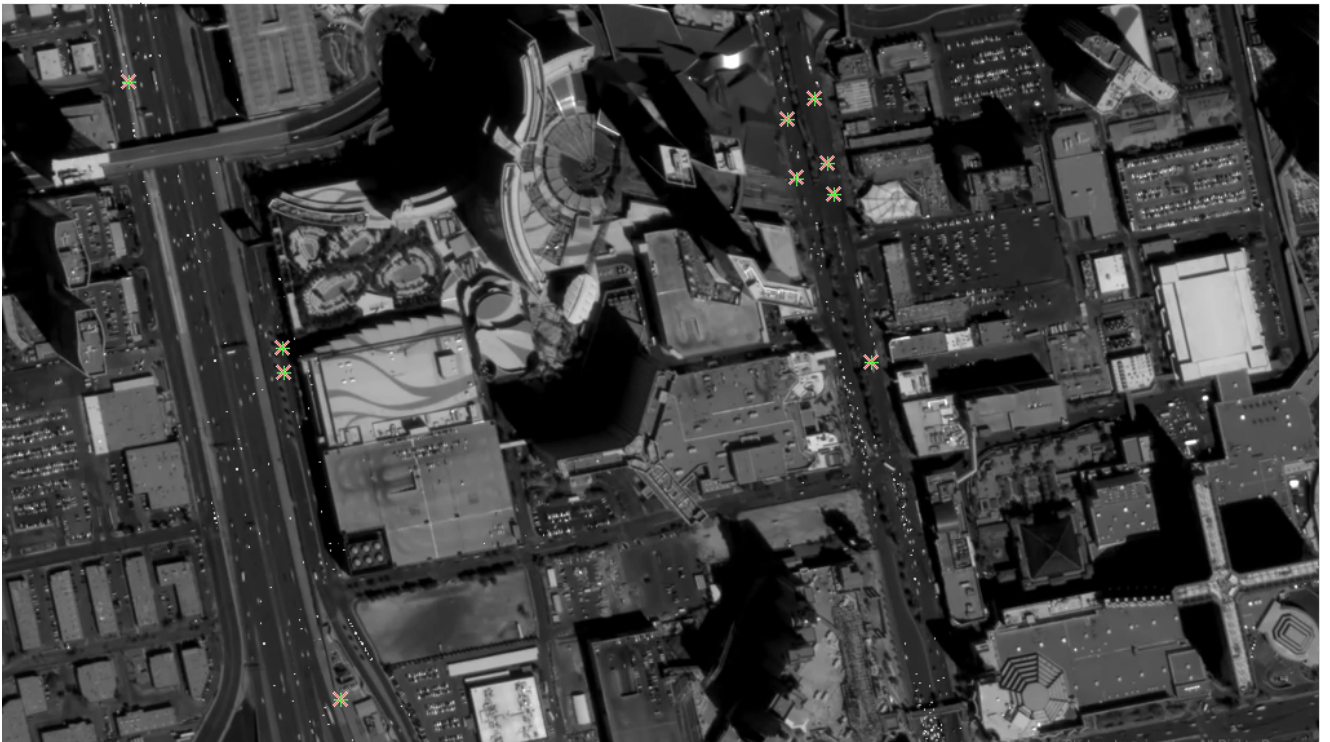
Και αντίστοιχα ακολουθεί ένα τμήμα της εικόνας σε ζουμ 400%:



*Εικόνα 4.12 : Τμήμα του καρτέ #61 από σετ δεδομένων με βήμα 15 καρτέ*

Στο συγκεκριμένο σετ δεδομένων αρχίζει να γίνεται πιο εμφανές ότι ο αλγόριθμος χάνει σε ακρίβεια όσο αυξάνει το βήμα μεταξύ των καρτέ, κάτι που είναι και οπτικά αντιληπτό, καθώς όπως φαίνεται ξεκάθαρα στην παραπάνω εικόνα η διαφορά μεταξύ εκτίμησης και πραγματικής θέσης μεγαλώνει σε σχέση με τα προηγούμενα σετ δεδομένων.

Ακολουθεί στιγμιότυπο από το σετ δεδομένων με βήμα 20 καρτέ:



*Εικόνα 4.13 : Καρτέ #81, σετ δεδομένων με βήμα 20 καρτέ*



Και αντίστοιχα ακολουθεί ένα τμήμα της εικόνας σε ζουμ 400%:



*Εικόνα 4.14 : Τμήμα του καρέ #81 από σετ δεδομένων με βήμα 20 καρέ*

Παρότι έχει αυξηθεί το βήμα διαδοχής των καρέ, δεν παρατηρείται κάποια ιδιαίτερη αύξηση του σφάλματος του αλγορίθμου. Πιθανή αιτία είναι ότι από 15 το βήμα σε αυξήθηκε σε 20 καρέ και η όποια αύξηση του σφάλματος εντοπισμού δεν είναι ορατή οπτικά ούτε σε ζουμ της τάξης του 400%.

Ακολουθεί στιγμιότυπο από το σετ δεδομένων με βήμα 30 καρέ:



*Εικόνα 4.15 : Καρέ #121, σετ δεδομένων με βήμα 30 καρέ*



Και αντίστοιχα ακολουθεί ένα τμήμα της εικόνας σε ζουμ 400%:

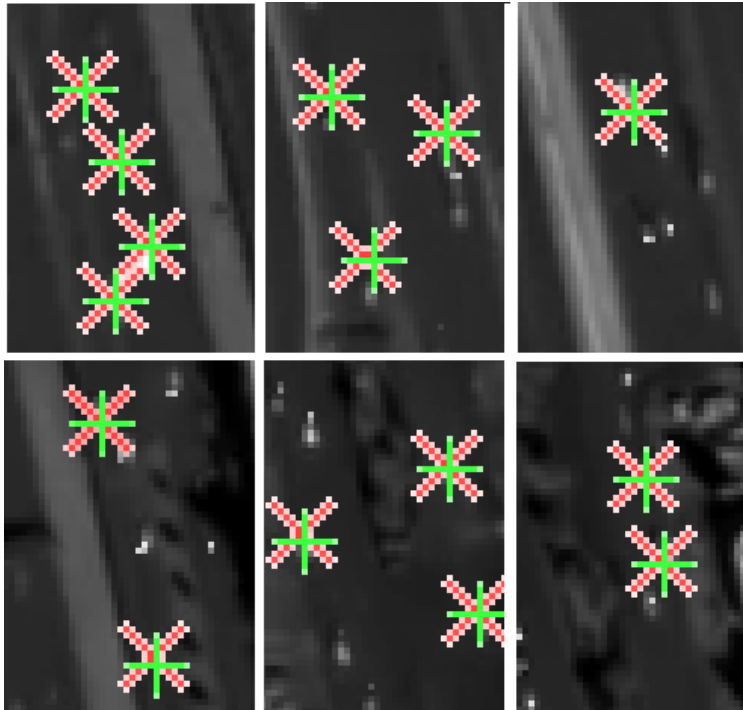


*Εικόνα 4.16 : Τμήμα του καρτέ #121 από σετ δεδομένων με βήμα 30 καρτέ*

Το σφάλμα και εδώ οπτικά είναι παρόμοιο με εκείνο του προηγούμενου σετ δεδομένων με βήμα 20 καρτέ. Αυτό ίσως οφείλεται στο γεγονός ότι τα συγκεκριμένα οχήματα κινούνται πιο ομαλά από τα αντίστοιχα του προηγούμενου σετ δεδομένων. Συνεπώς, παρ' ότι έχει αυξηθεί το βήμα διαδοχής των καρτέ το σφάλμα παραμένει στα ίδια επίπεδα. Βέβαια αυτό θα ελεγχθεί και ποσοτικά στην επόμενη ενότητα.

Στην περίπτωση κατά την οποία το βήμα διαδοχής αυξήθηκε στα 50 καρτέ, δεν κατέστη δυνατό να εντοπιστούν με χειροκίνητο τρόπο (όπως δηλαδή και στις προηγούμενα σετ δεδομένων), οχήματα τα οποία ο αλγόριθμος να μπορεί να εντοπίσει σωστά. Σε καμία περίπτωση δεν μπορεί να θεωρηθεί αξιόπιστος με το εν λόγω βήμα διαδοχής των καρτέ. Το πιο συχνό φαινόμενο ήταν να χάνει ο αλγόριθμος το όχημα από το πρώτο κιάλας καρτέ και να δίνει αποτελέσματα εντελώς λανθασμένα. Για το λόγο αυτό, δεν ήταν δυνατό να υπάρξει οπτικό αποτέλεσμα το οποίο να μπορεί να παρουσιαστεί, ακόμα και για να αναδειχθεί η ανεπάρκεια του αλγορίθμου και συνεπώς δεν υπάρχει κάποιο οπτικό αποτέλεσμα.

Ακολουθεί μια συνοπτική παρουσίαση της οπτικής αξιολόγησης, με παράθεση τμήματος των μεγεθυμένων εικόνων που έχουν παρουσιαστεί παραπάνω με σκοπό την καλύτερη παρατήρηση του σφάλματος ποιοτικά. Όπως και τα τμήματα των αρχικών καρτέ, έτσι και όλα τα τμήματα των εικόνων που φαίνονται παρακάτω είναι σε μεγέθυνση της τάξης του 400%.



*Εικόνα 4.17 : Συνοπτική παρουσίαση της ποιοτικής αξιολόγησης για κάθε σετ δεδομένων.*

*Πάνω σειρά από αριστερά προς δεξιά: Σετ δεδομένων με βήμα 1 καρε , 5 , 10*

*Κάτω σειρά από αριστερά: Βήμα 15, 20, 30*

Με μια πιο προσεκτική ματιά παρατηρείται ότι καθώς αυξάνεται το βήμα φαίνεται να αυξάνεται και το σφάλμα. Αυτό δεν μπορεί να εκτιμηθεί με απόλυτη σιγουριά για όλα τα οχήματα καθώς δεν είναι απαραίτητο ότι εντοπίζονται τα ίδια οχήματα για κάθε σετ δεδομένων. Όπως όμως έχει ειπωθεί και πιο πριν, είναι δύσκολο να εξαχθεί οπτικά επακριβές συμπέρασμα, για το κατά πόσο επηρεάζεται το σφάλμα. Κρίνεται δηλαδή απαραίτητη και η ποσοτική αξιολόγηση για την κατανόηση της τάξης μεγέθους του σφάλματος εντοπισμού του αλγορίθμου.

### 4.3 – Ποσοτική αξιολόγηση

Σε αντίθεση με την προηγούμενη υποενοότητα σε αυτήν αξιολογείται ποσοτικά το σφάλμα (ακρίβεια) του φίλτρου Kalman στον εντοπισμό πολλαπλών αντικειμένων/οχημάτων ανά σταθερό αριθμό καρτέ. Όπως έχει προαναφερθεί, έχουν επιλεγεί διαστήματα 1, 5, 10, 15, 20, 30 και 50 καρτέ.

Η μεθοδολογία που ακολουθήθηκε για την ποσοτικοποίηση της ακρίβειας του αλγορίθμου είναι η εξής: Ως σφάλμα εντοπισμού, θεωρήθηκε η απόσταση της πραγματικής θέσης (ground truth) σε κάθε καρτέ, από την θέση που εκτιμά ο αλγόριθμος σε κάθε επανάληψη που εκτελεί. Αυτό σημαίνει ότι αν ο αλγόριθμος εντοπισμού εκτελεί  $n$  επαναλήψεις, το οποίο εξαρτάται αποκλειστικά από τον αριθμό των καρτέ που εισάγονται ως δεδομένα, θα έχουμε  $n$  σφάλματα για κάθε όχημα που έχει επιλεγεί για εντοπισμό εξ αρχής.

Πιο συγκεκριμένα, για να ελεγχθεί η αξιοπιστία του αλγορίθμου δημιουργήθηκε, μέσα στον ίδιο τον αλγόριθμο, ένα υποπρόγραμμα. Αυτό υπολογίζει για κάθε όχημα που έχει επιλεγεί εξ αρχής και για κάθε καρτέ την απόσταση της εκτιμώμενης θέσης -σύμφωνα με τον αλγόριθμο- από την πραγματική. Η απόσταση μετρείται σε pixel καθώς στα πλαίσια αυτής της εργασίας η έμφαση δόθηκε στο κατά πόσο είναι εφικτός ο εντοπισμός με το εν λόγω φίλτρο και όχι τόσο στη χωρική πλευρά του ζητήματος. Θεωρητικά θα ήταν δυνατό να δοθεί χωρική διάσταση του pixel προσεγγιστικά, αν μετρηθούν οι διαστάσεις της περιοχής (π.χ. Google Earth) ή και επακριβώς αν δίνονταν τα απαραίτητα δεδομένα σχετικά με το δορυφορικό δέκτη από την εταιρία κατασκευής του.

Σε αυτό το σημείο πρέπει να επισημανθεί πως το όριο του μέσου σφάλματος, το οποίο ντετερμινιστικά επιλέχθηκε ως το όριο άνω του οποίου ο αλγόριθμος δεν θεωρείται πλέον ακριβής και αξιόπιστος, είναι 0.5 pixel.

Τα αποτελέσματα που παράγονται από το υποπρόγραμμα αυτό έχουν τη μορφή πίνακα, ο οποίος περιέχει τη διαφορά της απόστασης για το κάθε σετ δεδομένων και έχει την εξής μορφή: Γραμμές του πίνακα είναι τα καρτέ του εν λόγω σετ δεδομένων και στήλες τα οχήματα με τη σειρά που έχουν επιλεγεί. Όλα τα σετ δεδομένων ξεκινούν από το καρτέ νούμερο 1, εκτός από το σετ με βήμα 5 που ξεκινάει από το καρτέ νούμερο 5. Ακολουθεί ένας συνοπτικός πίνακας (4.1), στον οποίο φαίνεται ποια καρτέ υπάρχουν σε κάθε σετ δεδομένων καθώς και ο χρόνος μεταξύ δυο διαδοχικών αλλά και ο συνολικός χρόνος για κάθε σετ δεδομένων.

Βήμα διαδοχής	Αρχικο καρέ	Τελικό καρέ	Χρόνος μεταξύ δύο διαδοχικών καρέ (s)	Συνολικός χρόνος Dataset (s)
1	1	10	0.03	0.3
5	5	50	0.17	1.7
10	1	91	0.33	3.3
15	1	136	0.5	5
20	1	181	0.67	6.7
30	1	271	1	10
50	1	451	1.67	16.7

**Πίνακας 4.1 : Καρέ που περιλαμβάνονται σε κάθε σετ δεδομένων**

Για καλύτερη κατανόηση παρουσιάζεται παρακάτω ένας πλήρης πίνακας με τα σφάλματα για το σετ δεδομένων με βήμα διαδοχής 1 καρέ.

	Όχημα 18	Όχημα 43	Όχημα 51	Όχημα 89	Όχημα 119	Όχημα 125	Όχημα 134	Όχημα 139	Όχημα 175	Όχημα 210
Frame 2	0.006	0.006	0.058	0.006	0.006	0.050	0.006	0.050	0.006	0.041
Frame 3	0.008	0.091	0.069	0.103	0.103	0.034	0.103	0.120	0.008	0.105
Frame 4	0.144	0.084	0.120	0.078	0.063	0.113	0.078	0.085	0.144	0.088
Frame 5	0.110	0.080	0.240	0.077	0.172	0.075	0.091	0.172	0.110	0.081
Frame 6	0.059	0.219	0.170	0.046	0.049	0.099	0.084	0.070	0.117	0.047
Frame 7	0.025	0.079	0.301	0.021	0.066	0.088	0.184	0.070	0.069	0.157
Frame 8	0.170	0.097	0.158	0.171	0.041	0.049	0.048	0.176	0.193	0.051
Frame 9	0.046	0.141	0.231	0.046	0.159	0.179	0.183	0.050	0.061	0.067
Frame 10	0.190	0.066	0.044	0.065	0.051	0.044	0.227	0.065	0.068	0.041
				Average	Standard Deviation					
				0.093	0.062					

**Πίνακας 4.2 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 1 καρέ**

Παρατηρείται ότι το μέγιστο σφάλμα δεν υπερβαίνει το 0.300 pixel, ενώ η μέση τιμή είναι 0.093 pixel και η τυπική απόκλιση 0.062 pixel. Πρόκειται για μια αρκετά ικανοποιητική ακρίβεια δεδομένης της κλίμακας των καρέ αλλά και το μέγεθος των αντικειμένων.

Ακολουθούν οι ίδιοι πίνακες και για τα υπόλοιπα σετ δεδομένων. Να σημειωθεί εκ νέου ότι ο λόγος που δεν έχουν επιλεγεί τα ίδια οχήματα για όλα τα σετ δεδομένων, είναι ότι αφ' ενός μπορεί να μην φαίνονταν σε όλα τα καρέ κάθε σετ δεδομένων και αφ' ετέρου μπορεί να μην υπήρχαν καν μέσα στο καρέ. Δηλαδή το ίδιο όχημα στην περίπτωση του σετ δεδομένων με βήμα 1 που φαίνεται σε όλα τα καρέ ξεκάθαρα και ευδιάκριτα μπορεί με βήμα 30 να βγαίνει εκτός καρέ όσο προχωράει η ακολουθία. Να σημειωθεί επίσης ότι για δύο κυρίως λόγους μπορεί να χάνεται ένα αντικείμενο από το καρέ. Είτε εξέρχεται από αυτό, είτε χάνεται λόγω αδυναμίας του υπολογιστή να το αντιληφθεί για λόγους όπως π.χ. κατά τη διαδικασία της δημιουργίας της δυαδικής εικόνας να μην υπάρχει σαφής διάκριση μεταξύ αντικειμένου και υποβάθρου, είτε λόγω μιας αερογέφυρας που υπάρχει

πάνω από τον δρόμο, είτε ακόμα και θόρυβος στην αρχική εικόνα που υπάρχει σε κάθε δέκτη. Επίσης είναι πολύ πιθανό ένα αντικείμενο που χάνεται για κάποιο αριθμό καρέ να επανεμφανιστεί στα επόμενα.

Ακολουθεί ο πίνακας σφάλματος για το σετ δεδομένων με βήμα 5 καρέ:

	'Όχημα 19	'Όχημα 21	'Όχημα 57	'Όχημα 59	'Όχημα 63	'Όχημα 194	'Όχημα 196	'Όχημα 202	'Όχημα 204	'Όχημα 222
Frame 10	0.182	0.096	0.150	0.193	0.237	0.087	0.182	0.058	0.138	0.096
Frame 15	0.137	0.211	0.277	0.142	0.205	0.316	0.229	0.155	0.267	0.024
Frame 20	0.356	0.110	0.114	0.361	0.121	0.091	0.117	0.064	0.181	0.003
Frame 25	0.099	0.095	0.159	0.117	0.273	0.169	0.219	0.164	0.135	0.237
Frame 30	0.251	0.235	0.110	0.179	0.083	0.118	0.029	0.146	0.280	0.068
Frame 35	0.264	0.223	0.164	0.065	0.049	0.121	0.117	0.127	0.333	0.094
Frame 40	0.032	0.087	0.165	0.077	0.201	0.160	0.197	0.112	0.254	0.059
Frame 45	0.066	0.132	0.084	0.047	0.174	0.113	0.266	0.159	0.149	0.224
Frame 50	0.159	0.069	0.119	0.268	0.074	0.117	0.184	0.080	0.184	0.072
	<b>Average</b>		<b>Standard Deviation</b>							
	0.152		0.078							

*Πίνακας 4.3 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 5 καρέ*

Παρατηρείται ότι το μέγιστο σφάλμα δεν υπερβαίνει το 0.36 pixel, ενώ η μέση τιμή είναι 0.152 pixel και η τυπική απόκλιση 0.078 pixel. Υπάρχει μια αύξηση του μέσου σφάλματος της τάξης του 60% σε σχέση με το προηγούμενο σετ δεδομένων, αλλά παρόλα αυτά παραμένει σε αρκετά χαμηλά επίπεδα και θεωρείται ανεκτό. Η αύξηση αυτή μπορεί να δικαιολογηθεί αν αναλογιστεί κανείς ότι το βήμα διαδοχής πενταπλασιάστηκε.

Ακολουθεί ο αντίστοιχος πίνακας για το σετ δεδομένων με βήμα 10 καρέ:

	'Όχημα 10	'Όχημα 17	'Όχημα 29	'Όχημα 56	'Όχημα 65	'Όχημα 77	'Όχημα 85	'Όχημα 178	'Όχημα 188	'Όχημα 203
Frame 11	0.316	0.369	0.182	0.193	0.369	0.182	0.182	0.316	0.268	0.138
Frame 21	0.514	0.655	0.229	0.338	0.356	0.137	0.324	0.495	0.564	0.174
Frame 31	0.268	0.089	0.117	0.051	0.341	0.060	0.042	0.264	0.295	0.128
Frame 41	0.113	0.116	0.052	0.029	0.208	0.189	0.031	0.050	0.059	0.101
Frame 51	0.127	0.075	0.019	0.326	0.153	0.042	0.149	0.084	0.083	0.151
Frame 61	0.163	0.037	0.351	0.104	0.302	0.188	0.118	0.217	0.127	0.175
Frame 71	0.108	0.164	0.088	0.137	0.198	0.260	0.115	0.083	0.070	0.133
Frame 81	0.110	0.185	0.220	0.093	0.185	0.170	0.070	0.122	0.103	0.100
Frame 91	0.188	0.121	0.106	0.095	0.130	0.108	0.193	0.097	0.266	0.313
	<b>Average</b>		<b>Standard Deviation</b>							
	0.180		0.123							

*Πίνακας 4.4 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 10 καρέ*

Παρατηρείται ότι το μέγιστο σφάλμα δεν υπερβαίνει το 0.65 pixel, που είναι άνω του επιτρεπτού ορίου αλλά πρόκειται για μεμονωμένο γεγονός και δεν χρήζει ιδιαίτερης προσοχής, ενώ η μέση τιμή είναι 0.180 pixel και η

τυπική απόκλιση 0.123 pixel. Υπάρχει μια αύξηση του μέσου σφάλματος της τάξης του 20% σε σχέση με το προηγούμενο σετ δεδομένων αλλά παρ' όλα αυτά το σφάλμα παραμένει σε αρκετά χαμηλά επίπεδα και συνεχίζει να βρίσκεται εντός του ορίου.

Ακολουθεί ο αντίστοιχος πίνακας για το σετ δεδομένων με βήμα 15 καρέ καθώς και αυτοί για 20 και 30 αντίστοιχα.

	Όχημα 53	Όχημα 77	Όχημα 78	Όχημα 79	Όχημα 80	Όχημα 88	Όχημα 89	Όχημα 96	Όχημα 188	Όχημα 203
Frame 16	0.150	0.282	0.193	0.325	0.199	0.316	0.141	0.186	0.452	0.182
Frame 31	0.548	0.145	0.337	0.395	0.819	0.399	0.205	0.111	0.775	0.229
Frame 46	0.489	0.193	0.324	0.244	0.591	0.263	0.111	0.161	0.275	0.117
Frame 61	0.285	0.125	0.194	0.110	0.263	0.284	0.051	0.136	0.129	0.052
Frame 76	0.158	0.183	0.228	0.107	0.179	0.114	0.230	0.244	0.185	0.158
Frame 91	0.103	0.208	0.213	0.089	0.083	0.090	0.138	0.239	0.136	0.129
Frame 106	0.093	0.242	0.242	0.202	0.210	0.112	0.157	0.066	0.304	0.069
Frame 121	0.092	0.087	0.087	0.102	0.102	0.180	0.135	0.169	0.255	0.181
Frame 136	0.092	0.087	0.088	0.113	0.113	0.107	0.297	0.176	0.103	0.193
	<b>Average</b>				<b>Standard Deviation</b>					
	0.205				0.139					

**Πίνακας 4.5 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 15 καρέ**

Και εδώ ισχύει το ίδιο με προηγουμένως, δηλαδή έχουμε μέγιστη τιμή σφάλματος 0.82 pixel αλλά πρόκειται για μεμονωμένο γεγονός καθώς η αμέσως επόμενη τιμή σε φθίνουσα σειρά είναι 0.59 και αφορά το επόμενο καρέ για το ίδιο όχημα. Τα υπόλοιπα μεγέθη είναι λογικά και αναμενόμενα, με την μέση τιμή να είναι της τάξης του ενός πέμπτου του pixel, αυξημένη κατά 14% περίπου από την αντίστοιχη προηγούμενη και την τυπική απόκλιση να έχει τιμή 0.139 pixel.

Πίνακας σφάλματος για το σετ δεδομένων με βήμα 20 καρέ.

	Όχημα 21	Όχημα 29	Όχημα 41	Όχημα 44	Όχημα 53	Όχημα 66	Όχημα 85	Όχημα 104	Όχημα 88	Όχημα 261
Frame 21	0.132	0.369	0.337	0.141	0.199	0.425	0.414	0.282	0.407	0.311
Frame 41	0.371	0.465	0.504	0.613	0.667	0.535	0.427	0.384	0.654	0.758
Frame 61	0.214	0.076	0.434	0.464	0.664	0.335	0.486	0.185	0.321	0.480
Frame 81	0.102	0.225	0.281	0.517	0.836	0.349	0.136	0.102	0.150	0.249
Frame 101	0.215	0.477	0.026	0.143	0.114	0.014	0.168	0.139	0.125	0.266
Frame 121	0.372	0.384	0.353	0.103	0.176	0.060	0.048	0.196	0.062	0.161
Frame 141	0.211	0.243	0.130	0.146	0.096	0.243	0.179	0.232	0.112	0.075
Frame 161	0.150	0.120	0.234	0.149	0.164	0.118	0.065	0.137	0.083	0.206
Frame 181	0.232	0.061	0.130	0.120	0.142	0.086	0.226	0.067	0.083	0.229
	<b>Average</b>				<b>Standard Deviation</b>					
	0.256				0.179					

**Πίνακας 4.6 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 20 καρέ**

Στην περίπτωση αυτή το σφάλμα ανέρχεται στο ένα τέταρτο περίπου του pixel, δηλαδή αυξημένο κατά 25% σε σχέση με το προηγούμενο σετ δεδομένων, ενώ η τυπική απόκλιση είναι 0.179 pixel. Η μέγιστη τιμή σφάλματος είναι 0.83 pixel και όπως στις προηγούμενες περιπτώσεις, αποτελεί μεμονωμένο συμβάν.

Πίνακας σφάλματος για το σετ δεδομένων με βήμα 30 καρέ.

	Όχημα 44	Όχημα 81	Όχημα 89	Όχημα 96	Όχημα 150	Όχημα 203
<b>Frame 31</b>	0.193	0.293	0.282	0.282	0.328	0.369
<b>Frame 61</b>	0.850	0.350	0.384	0.355	0.396	0.465
<b>Frame 91</b>	0.742	0.350	0.194	0.182	0.202	0.076
<b>Frame 121</b>	0.270	0.198	0.166	0.203	0.079	0.185
<b>Frame 151</b>	0.077	0.283	0.383	0.109	0.080	0.283
<b>Frame 181</b>	0.167	0.189	0.360	0.247	0.332	0.358
<b>Frame 211</b>	0.140	0.093	0.530	0.405	0.318	0.400
<b>Frame 241</b>	0.254	0.210	0.343	0.394	0.123	0.457
<b>Frame 271</b>	0.389	0.036	0.178	0.266	0.408	0.248
	<b>Average</b>		<b>Standard Deviation</b>			
	0.286		0.154			

*Πίνακας 4.7 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 30 καρέ*

Στη συγκεκριμένη περίπτωση γίνεται αντιληπτό εκ πρώτης όψεως, ότι το πλήθος των αντικειμένων/οχημάτων είναι μικρότερο από εκείνο όλων των προηγούμενων περιπτώσεων. Αυτό συνέβη, γιατί δυσκολεύει κατά πολύ η διαδικασία εύρεσης οχημάτων, για τα οποία ο αλγόριθμος να δουλεύει σωστά και να τα εντοπίζει σε όλα τα καρέ. Για το λόγο αυτό και προκειμένου να εξαχθεί ένα σωστό συμπέρασμα για το σφάλμα εντοπισμού με βήμα 30 καρέ, επιλέχθηκαν 6 οχήματα, παρ' όλο που είναι λιγότερο αντιπροσωπευτικό δείγμα από τα 10 που έχουν επιλεγεί σε όλα τα προηγούμενα σετ. Γενικώς, καθ' όλη τη διάρκεια των πειραμάτων, τα οχήματα για τα οποία απέδωσε σωστά ο αλγόριθμος σε κάθε σετ δεδομένων, είχαν όλα σφάλμα εντοπισμού παρόμοιας τάξης μεγέθους και θεωρητικά ακόμα και από ένα όχημα θα μπορούσε να εξαχθεί μια εκτίμηση της ακρίβειας του αλγορίθμου.

Για τα 6 αυτά οχήματα το μέσο σφάλμα εντοπισμού είναι 0.286 του pixel, αυξημένο κατά 12% περίπου από το προηγούμενο σετ, η τυπική απόκλιση 0.154 και το μέγιστο σφάλμα 0.85 pixel . Η ακρίβεια εντοπισμού για όσα οχήματα δουλεύει σωστά ο αλγόριθμος, είναι ακόμα και στην περίπτωση αυτή ικανοποιητική και εντός του ορίου του 0.5 pixel.

Στόχος της δημιουργίας πολλών σετ δεδομένων με διαφορετικό κάθε φορά βήμα διαδοχής των καρτέ ήταν αφ' ενός η αξιολόγηση της συμπεριφοράς του αλγορίθμου όσο μεταβάλλεται το βήμα, αφ' ετέρου η εύρεση του βήματος εκείνου για το οποίο το σφάλμα εντοπισμού φτάνει στο όριο που αυθαίρετα έχει οριστεί ως ανεκτό.

Όσον αφορά το τελευταίο σετ δεδομένων με βήμα διαδοχής 50 καρτέ, προέκυψε το εξής: όποιο αντικείμενο και αν επιλέχθηκε για παρακολούθηση, το οποίο είχε εξακριβωθεί οπτικά εκ των προτέρων ότι υπάρχει σε όλα τα καρτέ (10), δεν μπόρεσε να εντοπιστεί με επιτυχία από τον αλγόριθμο. Πιθανή αιτία είναι το γεγονός ότι, λόγω των πολλών καρτέ που μεσολαβούν, η θέση που εκτιμά ο αλγόριθμος πως θα βρίσκεται το κάθε όχημα καταλήγει να είναι το ground truth καθώς η συνάρτηση DistCalculation είναι κατασκευασμένη έτσι ούτως ώστε να δίνει ως ground truth το prediction, αν δεν βρεθεί αντικείμενο εντός της ανοχής που της έχει προκαθοριστεί. Ουσιαστικά δηλαδή ακυρώνεται το στάδιο του update του φίλτρου, με αποτέλεσμα ο αλγόριθμος να μην δουλεύει σωστά. Αυτό εξηγεί και τα μικρά σφάλματα που παρουσιάζονται και σχολιάζονται στην επόμενη παράγραφο.

Παρ' όλα αυτά στην προσπάθεια εκτίμησης του σφάλματος, αφού ζητήθηκαν όλα τα αντικείμενα που εντοπίζει ο αλγόριθμος στο πρώτο καρτέ ως αντικείμενα προς εντοπισμό (269), παράχθηκε ο πίνακας σφαλμάτων με διαστάσεις 9 x 269. Από τον πίνακα αυτό κρατήθηκαν μόνο τα σφάλματα που οι τιμές τους φαίνονταν λογικές. Εξαλείφθηκαν δηλαδή τιμές που προσέγγιζαν το 0 στην τάξη των λίγων εκατοστών του pixel και τιμές υπερβολικά μεγάλες άνω των μερικών δεκάδων pixel. Έτσι από τα 269 αντικείμενα κρατήθηκαν μόνο τα 74 για τα οποία ο μέσος όρος του σφάλματος προέκυψε 0.52 pixel, δηλαδή αυξημένος κατά 80% σε σχέση με το προηγούμενο σετ, με τυπική απόκλιση 0.42 pixel και μέγιστη τιμή 1.73 pixel. Σε καμία περίπτωση δεν μπορούν να θεωρηθούν τα αποτελέσματα αξιόπιστα, απλώς παρουσιάζονται στα πλαίσια της παρούσας εργασίας λόγω του ερευνητικού της χαρακτήρα.

Με το πέρας των παραπάνω δοκιμών φαίνεται ξεκάθαρα και από ποσοτικής άποψης ότι όσο αυξάνεται το βήμα διαδοχής των καρτέ, τόσο αυξάνεται και το μέσο σφάλμα. Είναι κάτι που μπορεί να χαρακτηριστεί απόλυτα λογικό και αναμενόμενο, καθώς όσο πιο αραιά είναι τα δεδομένα τόσο πιθανότερο είναι να υπάρχει μεγαλύτερο σφάλμα.

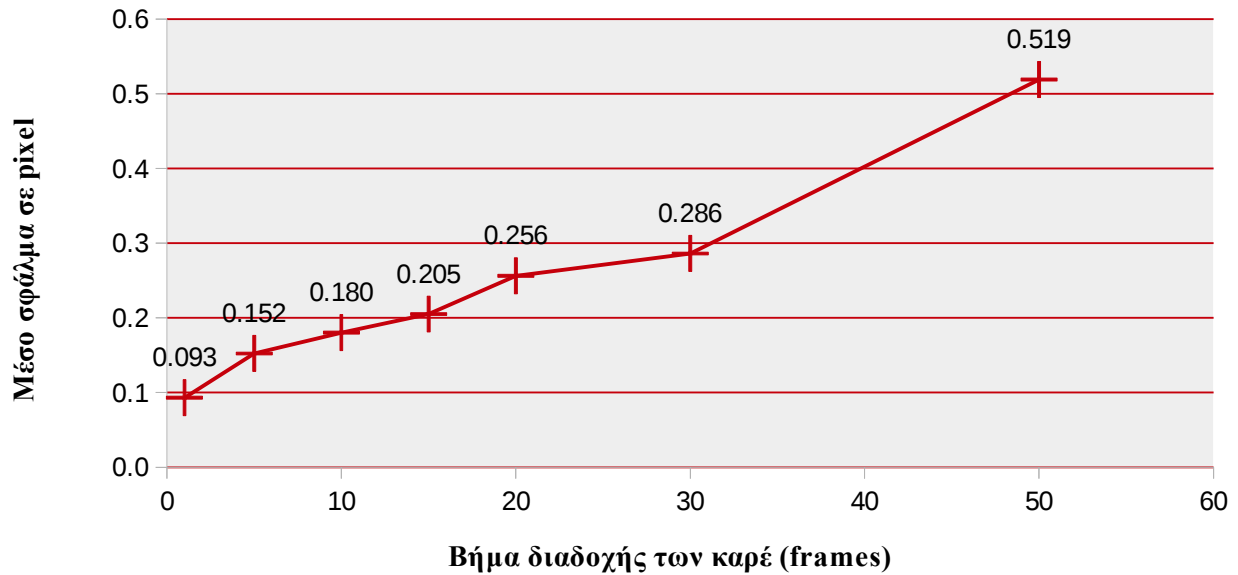
Παρακάτω παρουσιάζεται ένας πίνακας όπου φαίνεται συνοπτικά το μέσο σφάλμα και η τυπική απόκλιση του για κάθε σετ δεδομένων. Το σφάλμα του σετ δεδομένων με βήμα 50 παρουσιάζεται μεν, χωρίς όμως να θεωρείται αξιόπιστο.

1 FRAME		5 FRAMES		10 FRAMES		15 FRAMES		20 FRAMES		30 FRAMES		50 FRAMES	
Average	Std	Average	Std	Average	Std	Average	Std	Average	Std	Average	Std	Average	Std
0.093	0.062	0.152	0.078	0.180	0.123	0.205	0.139	0.256	0.179	0.286	0.154	0.519	0.419

*Πίνακας 4.8 : Συγκεντρωτικός πίνακας μέσων σφάλματος και τυπικής απόκλισης για όλα τα σετ δεδομένων*



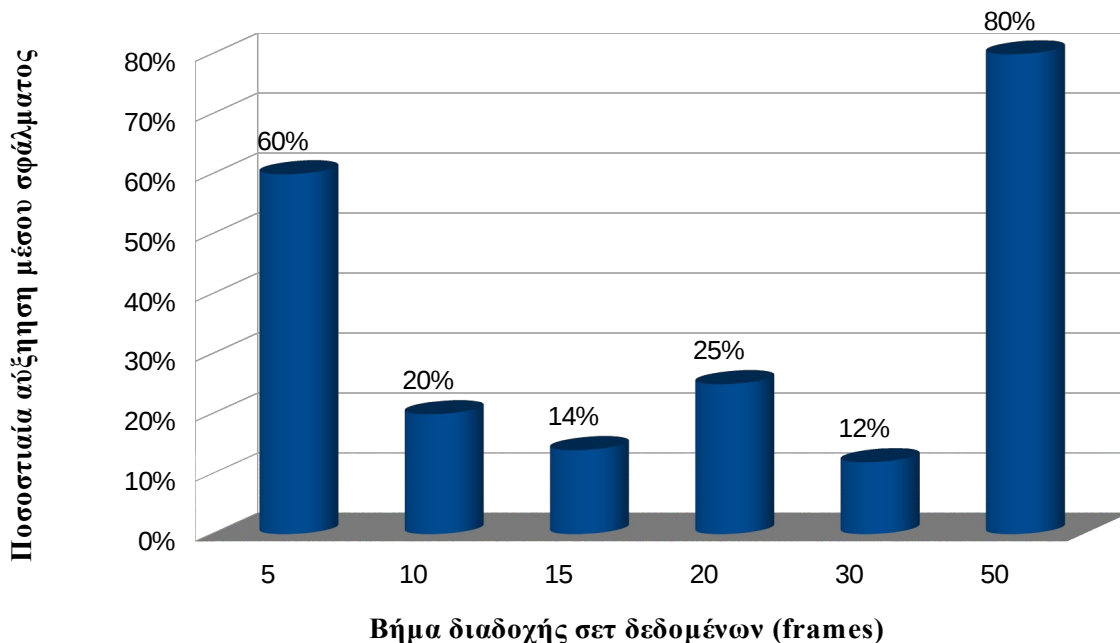
Ενώ το ίδιο αποτέλεσμα και γραφικά απεικονισμένο:



*Γράφημα 4.1 : Γραφική απεικόνιση του μέσου σφάλματος για όλα τα σετ δεδομένων*

Με την παραπάνω γραφική απεικόνιση παρατηρείται ότι το μέσο σφάλμα συνδέεται σχεδόν γραμμικά με το βήμα διαδοχής των καρτέ ειδικά στην περιοχή 5-20 καρτέ. Από τα 20 στα 30 καρτέ το μέσο σφάλμα δεν αυξάνεται με τον ίδιο ρυθμό όσο στα μικρότερα βήματα διαδοχής. Αντιθέτως, από τα 30 στα 50 καρτέ η αύξηση είναι της τάξης του 80%, που δεν μπορεί όμως να χαρακτηριστεί αντιπροσωπευτική εφ' όσον και τα σφάλματα από τα οποία προέκυψε το μέσο σφάλμα δεν μπορούν να θεωρούν αξιόπιστα.

Ακολουθεί γράφημα στο οποίο απεικονίζεται η ποσοστιαία αύξηση του μέσου σφάλματος συγκριτικά με το αμέσως προηγούμενο σετ δεδομένων. Όπως είναι λογικό το γράφημα ξεκινάει από το σετ δεδομένων με βήμα 5 καρτέ. Υπενθυμίζεται ότι το μέσο σφάλμα για το σετ δεδομένων με βήμα διαδοχής 1 καρτέ είναι 0.093 pixel.



*Γράφημα 4.2 : Ποσοστιαία αύξηση μέσου σφάλματος σε σχέση με το προηγούμενο σετ δεδομένων*

Το παραπάνω γράφημα δίνει την εντύπωση πως όταν το βήμα μεγαλώνει κατά μία σταθερή τιμή, το σφάλμα αυξάνει με ρυθμό ίδιας τάξης μεγέθους περίπου. Πολύ έντονη αύξηση παρατηρείται από το βήμα του 1 καρέ στα 5 όπου το σφάλμα αυξάνει κατά 60% περίπου και στο 30 προς 50 καρέ όπου η αύξηση είναι της τάξης του 80% φτάνοντας και ξεπερνώντας κατά λίγο το όριο ανοχής, χωρίς όμως να μπορεί να θεωρηθεί αξιόπιστο ως προς την ακρίβεια που δίνει το σετ των 50 καρέ.

Γενικότερα, αξίζει να σημειωθεί εδώ πως τα σφάλματα του αλγορίθμου που προέκυψαν είναι αντιπροσωπευτικά ως προς την ακρίβεια υπό τις ευνοϊκότερες συνθήκες. Δηλαδή έχει δοκιμασθεί η λειτουργία του αλγορίθμου σε απρόσκοπτες συνθήκες κίνησης, δηλαδή σε μη συνωστισμένες περιοχές και επίσης σε οχήματα που δεν εξαφανίζονται σε κανένα καρέ από τα υπό μελέτη.

Συνολικά, ο αλγόριθμος μπορεί να αξιολογηθεί ως ικανοποιητικά ακριβής ως προς τη λειτουργία του και ως προς το κατά πόσο εκπλήρωσε τον αρχικό σκοπό για τον οποίο σχεδιάστηκε, δηλαδή να εντοπίζει πολλαπλά αντικείμενα από δορυφορικό βίντεο. Αναμφισβήτητα μπορούν να γίνουν πάρα πολλές ενέργειες για τη βελτίωση του, όπως και δοκιμές με σκοπό να ελεγχθεί το πώς αποδίδει σε πιο δυσμενείς συνθήκες, αλλά και την περαιτέρω βελτιστοποίηση, πράγμα στο οποίο θα γίνει εκτενέστερη αναφορά στην ενότητα των συμπερασμάτων.

## Κεφάλαιο 5 – Συμπεράσματα

### 5.1 – Εισαγωγικά

Με βάση τα αποτελέσματα και την αξιολόγηση που παρουσιάστηκε καθώς και από τα διάφορα στάδια υλοποίησης της διπλωματικής εργασίας έχουν προκύψει διάφορα συμπεράσματα. Αυτά συνοψίζονται σε συμπεράσματα σχετικά με τις παραμέτρους του συστήματος, σε συμπεράσματα σχετικά με τις δυνατότητες και τα όρια του αλγορίθμου (ως προς την ορθή λειτουργία του) αλλά και συμπεράσματα που θα συμβάλουν καθοριστικά στην μελλοντική επέκτασή του.

Στόχος εδώ είναι η συνοπτική ανάδειξη των ιδιοτήτων του συγκεκριμένου αλγορίθμου ώστε να υπογραμμιστούν κάποια σημαντικά σημεία αλλά και να διατυπωθούν οι -κατά κάποιο τρόπο- προδιαγραφές του αλγορίθμου για βέλτιστα αποτελέσματα.

### 5.2 – Συμπεράσματα

*Είναι δυνατή η παρακολούθηση πολλαπλών αντικειμένων σε εκτεταμένες περιοχές σε δορυφορικά δεδομένα βίντεο ακόμα και για σχετικά μικρά κινούμενα αντικείμενα.*

Αυτό αποτελεί ένα βασικό συμπέρασμα, ενώ ταυτόχρονα αποτέλεσε και μια σημαντική πρόκληση απ' αρχής του σχεδιασμού και της εκπόνηση της εργασίας. Το γεγονός ότι ο αλγόριθμος έδωσε πολύ θετικά αποτελέσματα είναι ελπιδοφόρο στοιχείο για την μετεξέλιξή του και τη βελτίωση προς την ανάπτυξη επιχειρησιακών συστημάτων παρακολούθησης.

*Ο αλγόριθμος μπορεί να δώσει ποσοτικά ικανοποιητικά αποτελέσματα, ακόμα και όταν τα καρέ που αποτελούν τα δεδομένα εισόδου έχουν βήμα διαδοχής 30 καρέ, το οποίο για το συγκεκριμένο βίντεο μεταφράζεται σε χρονικό διάστημα 1 δευτερολέπτου.*

Το συγκεκριμένο συμπέρασμα απαιτεί ορισμένες διευκρινίσεις για να μην δημιουργηθεί εσφαλμένη εντύπωση σχετικά με τις επιδόσεις του αλγορίθμου. Ο όρος “ικανοποιητικό” αποτέλεσμα αναφέρεται σε ακρίβειες πρόβλεψης θέσης μικρότερες του μισού pixel. Επίσης το ότι ο αλγόριθμος μπορεί να δώσει αποτέλεσμα με βήμα διαδοχής έως και 30 καρέ περίπου, δεν σημαίνει ότι μπορεί να το κάνει για το ίδιο πλήθος οχημάτων και με την ίδια ευκολία με εκείνη που το κάνει όσο μειώνεται το εν λόγω βήμα.

***Όσο αυξάνει το βήμα διαδοχής των καρτέ, αυξάνει και το μέσο σφάλμα εντοπισμού των οχημάτων. Η σχέση αυτή είναι σχεδόν γραμμική.***

Το συμπέρασμα αυτό είναι λογικό και προβλέψιμο, παρ' όλα αυτά αναφέρεται διότι διερευνήθηκε και επιβεβαιώθηκε από τα πειράματα. Είναι λογικό και αναμενόμενο, το ότι όσο πιο αραιά δεδομένα παρέχονται στον αλγόριθμο, τόσο περισσότερο να δυσκολεύεται η πρόβλεψη αλλά και η σωστή εκτίμηση της θέσης, με αποτέλεσμα μεγαλύτερες ανακρίβειες στην προσπάθεια εντοπισμού (σφάλμα εντοπισμού).

***Σε περιοχές συνωστισμού οχημάτων και όσο αυξάνει το βήμα των καρτέ ο εντοπισμός είναι αρκετά δύσκολος.***

Καθ' όλη τη διάρκεια των δοκιμών για τον προσδιορισμό των επιδόσεων του αλγορίθμου, παρατηρήθηκε ότι σε περιοχές με πολλά κοντινά αντικείμενα και ταυτόχρονα στα σετ δεδομένων με μεγάλο βήμα διαδοχής, ο εντοπισμός γίνεται αρκετά δύσκολος έως και αδύνατος. Ο λόγος που χρησιμοποιείται ο όρος “αντικείμενα” και όχι οχήματα είναι ότι στα καρτέ, μετά τη μετατροπή τους σε δυαδικά, υπάρχουν και πολλές λευκές κουκκίδες που προέρχονται όμως από κτίρια και γενικότερα σταθερά αντικείμενα και επηρεάζουν κατά πολύ τη λειτουργία του αλγορίθμου.

***Χρειάζεται περαιτέρω έλεγχος για τον τρόπο που η παράμετρος ‘περίοδος δειγματοληψίας’ επηρεάζει το τελικό αποτέλεσμα.***

Όπως έχει ήδη αναφερθεί αναλυτικά στο κεφάλαιο της μεθοδολογίας, η περίοδος δειγματοληψίας ( $dt$ ) είναι μια πολύ ευαίσθητη παράμετρος του αλγορίθμου, η οποία δεν κατέστη δυνατό να μελετηθεί εκτενώς. Απαιτείται αποσαφήνιση πώς και κατά πόσο επηρεάζει τη λειτουργία του αλγορίθμου.

***Η επαναρχικοποίηση του φίλτρου σε κάθε καρτέ δεν ήταν τελικά εύκολο εγχείρημα λόγω δυσκολίας στην δημιουργία κριτηρίων ταυτοποίησης και αποκλεισμού.***

Αυτό το συμπέρασμα προέκυψε από την προσπάθεια επαναρχικοποίησης του φίλτρου σε κάθε καρτέ. Ο τρόπος που θα εξυπηρετούσε αυτή η ενέργεια είναι σύνθετος. Η αντικειμενική δυσκολία του εγχειρήματος ήταν κατά κύριο λόγο η εισαγωγή κατάλληλων κριτηρίων ταυτοποίησης και αποκλεισμού κινούμενων αντικειμένων (πχ. για να περιλαμβάνονται τυχόν νέο-εισερχόμενα οχήματα, οχήματα που εξαφανίζονται και επανεμφανίζονται λόγω μιας γέφυρας, κοκ). Απαιτούνται δηλαδή πάντα πολύπλοκα κριτήρια, συγχώνευση (fusion) της πληροφορίας και πολλές δοκιμές μέχρι να υλοποιηθεί μια ικανοποιητική λύση για το πρόβλημα αυτό.

## Κεφάλαιο 6 – Προτάσεις

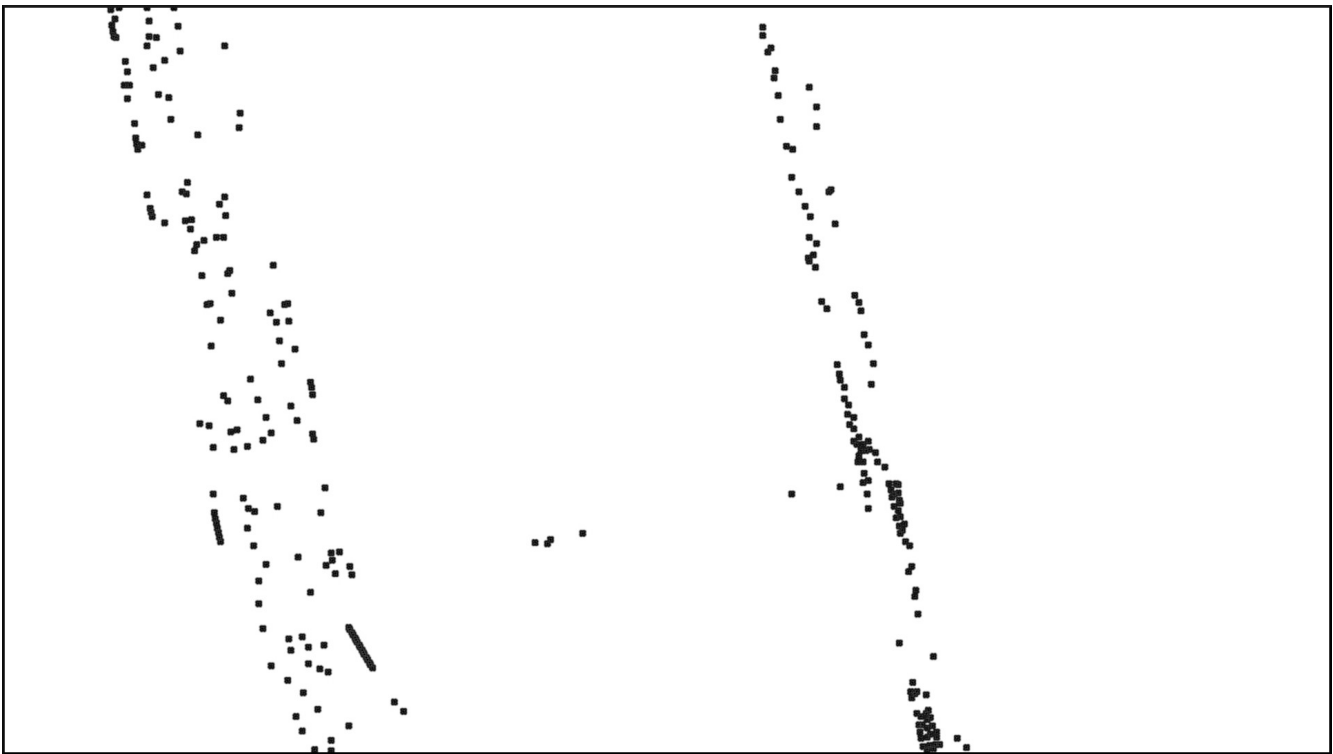
Ο αλγόριθμος είναι ακόμα σε πρώιμο στάδιο από άποψη αυτοματισμών και δυνατοτήτων. Στο πλαίσιο της παρούσας εργασίας δεν ήταν δυνατό να εξελιχθεί περισσότερο. Για να επιτευχθεί αυτό απαιτείται εξειδικευμένη και βαθιά γνώση αλλά και εξοικείωση με τις τεχνικές του προγραμματισμού στο δεδομένο περιβάλλον. Κυρίως όμως χρειάζεται αρκετά μεγάλο χρονικό διάστημα δοκιμών για τον πειραματισμό και τη δοκιμή αρκετών επιλογών μέχρι την κατάληξη σε κάτι ολοκληρωμένο και αυτοματοποιημένο.

Αντί αυτών, στην ενότητα αυτήν παρατίθενται ορισμένες προτάσεις ως απόρροια κάποιων εκ των συμπερασμάτων. Κύριος στόχος των προτάσεων αυτών είναι δείξουν την εξελιξιμότητα και τη δυναμική του αλγορίθμου. Περιγράφουν τις κύριες κατευθύνσεις που πρέπει να ακολουθηθούν για την ουσιαστική αναβάθμισή του. Αυτός είναι και ο λόγος που οι προτάσεις αυτές παρατίθενται με τη σειρά κατά την οποία κρίνεται από το συγγραφέα ότι πρέπει να υλοποιηθούν.

***Προτείνεται ανεπιφύλακτα η χρήση μάσκας στην εικόνα ούτως ώστε να απομονωθεί ο δρόμος που είναι εξ' άλλου και η μόνη περιοχή που απασχολεί τον αλγόριθμο.***

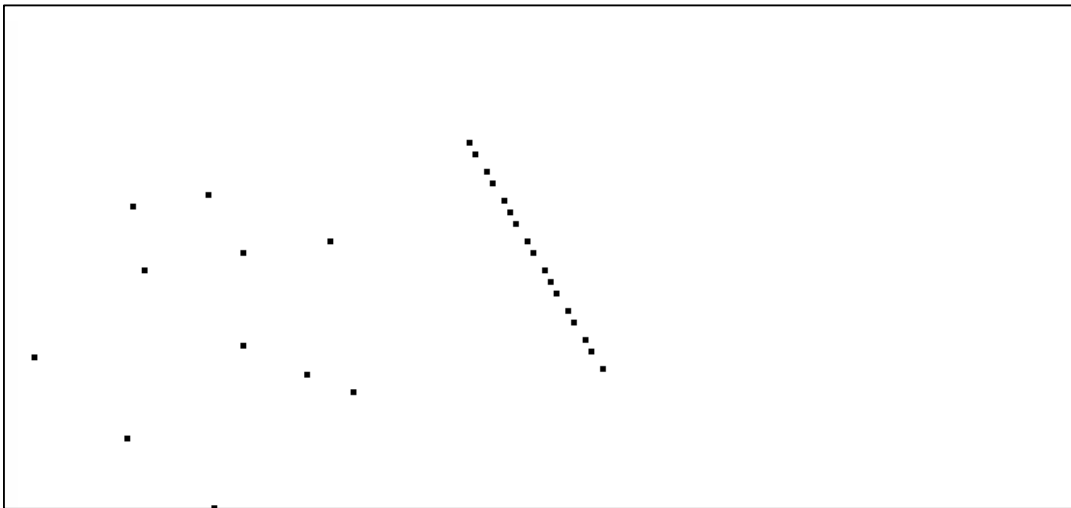
Η χρήση μάσκας θα διευκόλυνε πολύ τον αλγόριθμο, γιατί περιορίζει την περιοχή στην οποία αυτός ανιχνεύει. Υπάρχουν περιοχές όπου εμφανίζονται κάποιες λευκές κουκκίδες οι οποίες προέρχονται από άσπρες αποχρώσεις ακμών κτιρίων, τεχνικών έργων κ.λ.π. και γενικά οτιδήποτε δεν είναι όχημα. Οτιδήποτε δεν είναι όχημα εντός του οδικού άξονα αποτελεί θόρυβο και θα έπρεπε να αποκοπεί. Με τον τρόπο αυτό μπορούν να γίνουν κατά πολύ αποδοτικότεροι οι έλεγχοι. Επίσης μειώνεται αρκετά ο υπολογιστικός φόρτος, που λειτουργεί προς όφελος της λειτουργίας σε πραγματικό χρόνο.

Παρουσιάζεται ένα στιγμιότυπο του βίντεο στο οποίο υπάρχουν κουκκίδες οι οποίες δεν αποτελούν οχήματα και αντίστοιχα το τμήμα της περιοχής όπου παρουσιάζεται το πρόβλημα του θορύβου σε μεγέθυνση 400%.



*Εικόνα 6.1 : Καρέ #45 σε δυαδική μορφή (binary) σε αρνητικό*

Και αντίστοιχα η περιοχή που έχει επισημανθεί με την έλλειψη σε μεγέθυνση 400%. Εδώ πρέπει να σημειωθεί ότι, το πρόβλημα του συνωστισμού εν μέρει ταυτίζεται με αυτή την περίπτωση, η οποία είναι και ένας λόγος που δυσκολεύεται και μάλιστα άσκοπα ο αλγόριθμος.



*Εικόνα 6.2 : Περιοχή συνωστισμού από pixels που δεν αντιστοιχούν σε σχήματα*

***Θα μπορούσε να διευκολυνθεί η επιλογή και η παρακολούθηση των οχημάτων και γενικότερα η μελέτη, αν γινόταν labeling των οχημάτων επί του οπτικού αποτελέσματος.***

Καθ' όλη τη διάρκεια που επιλέγονταν χειροκίνητα τα οχήματα τα οποία επρόκειτο να εντοπιστούν, η διαδικασία για να αντιστοιχιστεί το όχημα που φαίνεται στο καρτέ με τον εαυτό του στη λίστα των περιεχομένων οχημάτων ήταν πολύ κοπιαστική. Για να συνδεθεί η κουκκίδα με τον αύξοντα αριθμό στη λίστα έπρεπε να γίνει χειροκίνητα η παρατήρηση των συντεταγμένων του επί της οθόνης και η αναζήτηση τους στην εν λόγω λίστα.

Αν αντί αυτού δίπλα σε κάθε όχημα που εντοπιζόταν υπήρχε ο αριθμός ευρετηρίου του στη λίστα αυτή θα ήταν πολύ πιο εύκολα όλα τα πειράματα καθώς θα εξοικονομούσαν χρόνο και θα υπήρχε καλύτερη εποπτεία. Ακόμα, αν ήταν δυνατή η εμφάνιση αυτού του αριθμού για κάθε όχημα με την επιλογή του με το ποντίκι για παράδειγμα, θα ήταν υπερβολικά απλή και χωρίς πιθανότητα λάθους η επιλογή οχημάτων που θα συμπεριληφθούν στα πειράματα.





## Βιβλιογραφία

- [1] Faragher R. , **“Understanding the Basis of the Kalman Filter Via a Simple and Intuitive Derivation”** , IEEE Signal Processing Magazine , Volume:29 , Issue: 5 , pages 128 – 132, September 2012
- [2] Hugh Durrant-Whyte, **“Introduction to Estimation and the Kalman Filter”** , January 2001
- [3] Horst Possegger Thomas, Mauthner, Peter M. Roth, Horst Bischof , **“Occlusion Geodesics for Online Multi-Object Tracking”** , Institute for Computer Graphics and Vision, Graz University of Technology , 2014
- [4] Jerome Berclaz, Francois Fleuret, Engin Turetken, Pascal Fua , **“Multiple Object Tracking using K-Shortest Paths Optimization”** , February 2011
- [5] Konstantinos Makantasis, **“Kalman Filtering implementation in Python”** , May 2015
- [6] L. Meng, J. P. Kerekes, **“Object Tracking Using High Resolution Satellite Imagery”** , January 2012
- [7] Magdi S. Mahmoud, Haris M. Khalid, **“Bibliographic Review on Distributed Kalman Filtering”** , March 15, 2013
- [8] P.J. Hargrave. **“A tutorial introduction to Kalman filtering. In IEE Colloquium on Kalman Filters: Introduction, Applications and Future Developments”** , pages 1/1–1/6, February 1989.
- [9] R. E. Kalman, **“A New Approach to Linear Filtering and Prediction Problems. Journal of Fluids Engineering”** , 82(1):35–45, March 1960.
- [10] Siri Øyen Larsen, Hans Koren, Rune Solberg , **“Traffic Monitoring using Very High Resolution Satellite Imagery”** , July 2009
- [11] T. S. Huang, **“Computer Vision: Evolution and Promise”** , September 1996
- [12] Welch G. , Bishop G. , **“An Introduction to the Kalman Filter”** , SIGGRAPH 2001
- [13] Xu S., Chang A. , **“Robust Object Tracking Using Kalman Filters with Dynamic Covariance”** , 2014
- [14] Yilmaz, A., Javed, O., and Shah, M. 2006. **“Object tracking: A survey”**. ACM Comput. Surv. 38, 4, Article 13 (Dec. 2006), 45 pages. DOI = 10.1145/1177352.1177355 <http://doi.acm.org/10.1145/1177352.1177355>

## Διαδικτυακοί Ιστότοποι

<http://www.codeproject.com/Articles/865935/Object-Tracking-Kalman-Filter-with-Ease>

[https://en.wikipedia.org/wiki/Kalman\\_filter](https://en.wikipedia.org/wiki/Kalman_filter)

[http://biorobotics.ri.cmu.edu/papers/sbp\\_papers/integrated3/kleeman\\_kalman\\_basics.pdf](http://biorobotics.ri.cmu.edu/papers/sbp_papers/integrated3/kleeman_kalman_basics.pdf)

<http://www.mathworks.com/>

<http://soe.rutgers.edu/~meer/UGRAD/cv1intro.pdf>

<http://learnpythonthehardway.org/>

<http://hilite.me/>

Φωτογραφίες εξωφύλλου:

<http://www.digitalsecuritymagazine.com/en/2015/07/02/vivotek-ofrece-monitorizacion-de-trafico-en-tiempo-real-con-ip816a-lpc/>

[http://www.atelier.net/en/trends/articles/terra-bella-beyond-geospatial-imagery-precise-numerical-data\\_440618](http://www.atelier.net/en/trends/articles/terra-bella-beyond-geospatial-imagery-precise-numerical-data_440618)



## Παράρτημα Ι – Κατάλογος Εικόνων

Εικόνα 1.1 : Παράδειγμα αλγορίθμου που εκτελεί εντοπισμό πολλαπλών αντικειμένων.....	2
Εικόνα 2.1: Αλληλεπίδραση της Όρασης Υπολογιστών με άλλα επιστημονικά πεδία.....	4
Εικόνα 2.2: Αναπαραστάσεις αντικειμένων.....	6
Εικόνα 2.3: Στιγμιότυπα με το αποτέλεσμα του αλγορίθμου.....	8
Εικόνα 2.4: Διάγραμμα ροής του παρουσιαζόμενου αλγορίθμου.....	9
.....	10
Εικόνα 2.5: Εντοπισμός του πλοίου στην περιοχή του λιμανιού (a) με το πλοίο να βρίσκεται εντός του κόκκινου περιγράμματος. (b) Η τροχιά κίνησης του πλοίου.....	10
Εικόνα 2.6: Εντοπισμός των αεροσκαφών στην περιοχή του αεροδρομίου (a). Το αεροσκάφος A είναι μέσα σε άσπρο πλαίσιο και το B σε κόκκινο. (b) Τροχιά κίνησης του αεροσκάφους A. (c) Τροχιά κίνησης του αεροσκάφους B.....	11
Εικόνα 2.7: Στάδια τμηματοποίησης (segmentation) της εικόνας.....	12
Εικόνα 2.8: Περιπτώσεις αδυναμίας εντοπισμού των αντικειμένων από τον αλγόριθμο.....	13
Εικόνα 3.1 : Η περιοχή μελέτης όπως απεικονίζεται στο Google Earth.....	16
Εικόνα 3.2 : Το πρώτο καρέ όπως αυτό εξήχθη από το βίντεο.....	16
Εικόνα 3.3 : Σχηματική αναπαράσταση της λειτουργίας του φίλτρου Kalman.....	19
Εικόνα 3.4: Ο τρόπος λειτουργίας του φίλτρου Kalman.....	20
<i>Εικόνα 3.5: Στιγμιότυπο από το περιβάλλον του Spyder.....</i>	<i>27</i>
Εικόνα 4.1 : Καρέ #256 του αρχικού βίντεο.....	44
Εικόνα 4.2 : Καρέ #45 σε δυαδική μορφή (binary) σε αρνητικό.....	45
<i>Εικόνα 4.3 : Δεδομένα εισόδου με το αποτέλεσμα του αλγορίθμου (Στιγμιότυπο 1 καρέ).....</i>	<i>47</i>
Εικόνα 4.4 :Καρέ #2 επαυξημένο με το αποτέλεσμα του αλγορίθμου.....	48
Εικόνα 4.5 : Καρέ #5, από σετ δεδομένων με βήμα 1 καρέ.....	49
Εικόνα 4.6 : Τμήμα του καρέ #5 από σετ δεδομένων με βήμα 1 καρέ.....	49
Εικόνα 4.7 : Καρέ #25, σετ δεδομένων με βήμα 5 καρέ.....	50
Εικόνα 4.8 : Τμήμα του καρέ #25 από σετ δεδομένων με βήμα 5 καρέ.....	51
Εικόνα 4.9 : Καρέ #41, σετ δεδομένων με βήμα 10 καρέ.....	51
Εικόνα 4.10 : Τμήμα του καρέ #41 από σετ δεδομένων με βήμα 10 καρέ.....	52

Εικόνα 4.11 : Καρέ #61, σετ δεδομένων με βήμα 15 καρέ.....	52
Εικόνα 4.12 : Τμήμα του καρέ #61 από σετ δεδομένων με βήμα 15 καρέ.....	53
Εικόνα 4.13 : Καρέ #81, σετ δεδομένων με βήμα 20 καρέ.....	53
Εικόνα 4.14 : Τμήμα του καρέ #81 από σετ δεδομένων με βήμα 20 καρέ.....	54
Εικόνα 4.15 : Καρέ #121, σετ δεδομένων με βήμα 30 καρέ.....	54
Εικόνα 4.16 : Τμήμα του καρέ #121 από σετ δεδομένων με βήμα 30 καρέ.....	55
Εικόνα 4.17 : Συνοπτική παρουσίαση της ποιοτικής αξιολόγησης για κάθε σετ δεδομένων.....	56
Εικόνα 6.1 : Καρέ #45 σε δυαδική μορφή (binary) σε αρνητικό.....	68
Εικόνα 6.2 : Περιοχή συνωστισμού από <i>pixels</i> που δεν αντιστοιχούν σε σχήματα.....	68

## Παράρτημα II – Κατάλογος Πινάκων και Γραφημάτων

Πίνακας 4.1 : Καρέ που περιλαμβάνονται σε κάθε σετ δεδομένων.....	58
Πίνακας 4.2 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 1 καρέ.....	58
Πίνακας 4.3 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 5 καρέ.....	59
Πίνακας 4.4 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 10 καρέ.....	59
Πίνακας 4.5 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 15 καρέ.....	60
Πίνακας 4.6 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 20 καρέ.....	60
Πίνακας 4.7 : Πίνακας σφαλμάτων για σετ δεδομένων με βήμα 30 καρέ.....	61
Πίνακας 4.8 : Συγκεντρωτικός πίνακας μέσου σφάλματος και τυπικής απόκλισης για όλα τα σετ δεδομένων.....	62
Γράφημα 4.1 : Γραφική απεικόνιση του μέσου σφάλματος για όλα τα σετ δεδομένων.....	63
Γράφημα 4.2 : Ποσοστιαία αύξηση μέσου σφάλματος σε σχέση με το προηγούμενο σετ δεδομένων...	64