



ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ
ΤΜΗΜΑ ΗΛΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ
ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΩΝ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗ-
ΡΟΦΟΡΙΑΣ
ΕΡΓΑΣΤΗΡΙΟ ΕΝΣΥΡΜΑΤΗΣ ΤΗΛΕΠΙΚΟΙΝΩΝΙΑΣ ΚΑΙ ΤΕ-
ΧΝΟΛΟΓΙΑΣ ΤΗΣ ΠΛΗΡΟΦΟΡΙΑΣ

**ΕΥΘΥΓΡΑΜΜΙΣΗ ΝΕΦΟΥΣ ΣΗΜΕΙΩΝ
ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΣΗΜΑΣΙΟΛΟΓΙΚΗ
ΚΑΤΑΤΜΗΣΗ ΣΕ ΣΥΣΤΗΜΑΤΑ
ΣΥΝΕΡΓΑΤΙΚΩΝ ΟΧΗΜΑΤΩΝ**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΧΑΛΚΙΑΣ ΚΩΝΣΤΑΝΤΙΝΟΣ ΤΟΥ ΕΥΣΤΡΑΤΙΟΥ

ΕΠΙΒΛΕΠΩΝ: ΜΟΥΣΤΑΚΑΣ ΚΩΝΣΤΑΝΤΙΝΟΣ

ΠΑΤΡΑ - ΙΑΝΟΥΑΡΙΟΣ 2024

Πανεπιστήμιο Πατρών, Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών.

Χαλκιάς Κωνσταντίνος του Ευστρατίου

© 2024 – Με την επιφύλαξη παντός δικαιώματος

Το σύνολο της εργασίας αποτελεί πρωτότυπο έργο, παραχθέν από τον Χαλκιά Κωνσταντίνο του Ευστρατίου, και δεν παραβιάζει δικαιώματα τρίτων καθ' οιονδήποτε τρόπο. Αν η εργασία περιέχει υλικό, το οποίο δεν έχει παραχθεί από τον ίδιο, αυτό είναι ευδιάκριτο και αναφέρεται ρητώς εντός του κειμένου της εργασίας ως προϊόν εργασίας τρίτου, σημειώνοντας με παρομοίως σαφή τρόπο τα στοιχεία ταυτοποίησής του, ενώ παράλληλα βεβαιώνει πως στην περίπτωση χρήσης αυτούσιων γραφικών αναπαραστάσεων, εικόνων, γραφημάτων κ.λπ., έχει λάβει τη χωρίς περιορισμούς άδεια του κατόχου των πνευματικών δικαιωμάτων για την συμπερίληψη και επακόλουθη δημοσίευση του υλικού αυτού.

ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η Διπλωματική Εργασία με τίτλο

ΕΥΘΥΓΡΑΜΜΙΣΗ ΝΕΦΟΥΣ ΣΗΜΕΙΩΝ ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΣΗΜΑΣΙΟΛΟΓΙΚΗ ΚΑΤΑΤΜΗΣΗ ΣΕ ΣΥΣΤΗΜΑΤΑ ΣΥΝΕΡΓΑΤΙΚΩΝ ΟΧΗΜΑΤΩΝ

του φοιτητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας
Υπολογιστών

Χαλκιά Κωνσταντίνου του Ευστρατίου
Αριθμός Μητρώου: 1066615

Παρουσιάστηκε δημόσια στο τμήμα Ηλεκτρολόγων Μηχανικών και
Τεχνολογίας Υπολογιστών στις

01/02/2024

και εξετάστηκε από την ακόλουθη εξεταστική επιτροπή:

Μουστάκας Κωνσταντίνος, Καθηγητής, τΗΜΤΥ (επιβλέπων)

Σκόδρας Αθανάσιος, Καθηγητής, τΗΜΤΥ (μέλος επιτροπής)

Μπεχλιούλης Χαράλαμπος, Αναπληρωτής Καθηγητής, τΗΜΤΥ (μέλος
επιτροπής)

Ο Επιβλέπων

Ο Διευθυντής του Τομέα

Μουστάκας Κωνσταντίνος
Καθηγητής

Λογοθέτης Μιχαήλ
Καθηγητής

*“Science and engineering are not just about facts and figures;
they are also about creativity and imagination.”*

ΕΥΧΑΡΙΣΤΙΕΣ

Αρχικά, θα ήθελα να εκφράσω τη βαθύτατη ευγνωμοσύνη μου στον Καθ. Κωνσταντίνο Μουστάκα, τον επιβλέποντα της παρούσας διπλωματικής εργασίας. Το πάθος του για τη διδασκαλία και την έρευνα, καθώς και η ενθάρρυνση και η υποστήριξη που μου παρείχε καθ' όλη τη διάρκεια της εκπόνησης της εργασίας, συνέβαλαν καθοριστικά στην ακαδημαϊκή μου ανάπτυξη και στη διαμόρφωση του τρόπου σκέψης μου. Οι συμβουλές και η καθοδήγησή του ήταν ανεκτίμητες και μου επέτρεψαν να ολοκληρώσω με επιτυχία την εργασία μου.

Θα ήθελα επίσης να ευχαριστήσω θερμά την οικογένεια μου για την αμέριστη αγάπη και την υποστήριξή τους. Η ενθάρρυνση και η πίστη τους σε μένα ήταν η κινητήρια δύναμη πίσω από την επιτυχία μου. Χωρίς τη βοήθειά τους, δεν θα είχα καταφέρει να φτάσω σε αυτό το σημείο.

Ευχαριστώ όλους εσάς που κάνατε δυνατό όλο αυτό. Είμαι πραγματικά ευγνώμων για τις συνεισφορές σας.

ΠΕΡΙΛΗΨΗ

ΕΥΘΥΓΡΑΜΜΙΣΗ ΝΕΦΟΥΣ ΣΗΜΕΙΩΝ ΧΡΗΣΙΜΟΠΟΙΩΝΤΑΣ ΣΗΜΑΣΙΟΛΟΓΙΚΗ ΚΑΤΑΤΜΗΣΗ ΣΕ ΣΥΣΤΗΜΑΤΑ ΣΥΝΕΡΓΑΤΙΚΩΝ ΟΧΗΜΑΤΩΝ

Φοιτητής: Χαλκιάς Κωνσταντίνος

Επιβλέπων: Μουστάκας Κωνσταντίνος

Τα αυτόνομα οχήματα και γενικότερα η αυτόνομη οδήγηση, που για πολλούς θεωρείτε το μέλλον της αυτοκίνησης και των μεταφορών, θα αρχίσει σε λίγο καιρό να μπαίνει σε μεγάλο βαθμό στην καθημερινότητα των ανθρώπων αλλάζοντας τις συνήθειές τους ριζικά. Αυτά τα οχήματα είναι εξοπλισμένα με προηγμένους αισθητήρες και αλγορίθμους που τους δίνουν την δυνατότητα να μπορούν να πλοηγούνται στους δρόμους χωρίς ανθρώπινη παρέμβαση. Παρόλο που δεν υπάρχει ο ανθρώπινος παράγοντας μπορούν να μειώσουν σε μεγάλο βαθμό τα αυτοχήματα αλλά και να αυξήσουν την ασφάλεια στους δρόμους καθώς σε λίγα χρόνια ο αριθμός παράγωγης θα αυξηθεί σε τέτοιο βαθμό που δεν θα παράγονται συμβατικά οχήματα.

Αυτό θα έχει σαν αποτέλεσμα να κυκλοφορούν όλο και περισσότερα αυτόνομα οχήματα, τα οποία θα πρέπει με κάποιον τρόπο να επικοινωνούν και να ανταλλάσσουν πληροφορίες. Για να γίνει αυτό με έναν αποδοτικό τρόπο πρέπει να αναπτυχθούν μέθοδοι ο οποίες θα διευκολύνουν τον διαμοιρασμό δεδομένων ανάμεσα στα οχήματα. Αυτός ο διαμοιρασμός της πληροφορίας ανάμεσα στα οχήματα είναι εξαιρετικά σημαντικός, καθώς για πρακτικούς λόγους ένα αυτόνομο όχημα μπορεί να μην έχει την πλήρη γνώση του περιβάλλοντος που βρίσκεται. Ένα τέτοιο παράδειγμα μπορεί να αποτελείται από έναν πεζό που δεν φαίνεται από την οπτική γωνία του οχήματος ή ένα μηχανάκι που κινείται δίπλα από ένα μεγάλο όχημα (λεωφορείο, φορτηγό κ.α.) και εμποδίζει την ανίχνευση του δικύκλου.

Η συγκεκριμένη διπλωματική εργασία προτείνει ένα τρόπο για ευθυγράμμιση νεφών σημείων που μπορεί να λύσει ένα τέτοιο πρόβλημα χρησιμοποιώντας έναν αποδοτικό και αποτελεσματικό τρόπο αναπαράστασης του περιβάλλοντος που βρίσκεται το κάθε αυτόνομο όχημα. Πιο συγκεκριμένα, το κάθε αυτόνομο όχημα κατασκευάζει έναν γράφο που έχει για κορυφές του κάποια χαρακτηριστικά του περιβάλλοντός του που παράχθηκαν μετά από σημασιολογική κατάτμηση της σκηνής που βρίσκεται. Αυτά τα χαρακτηριστικά μπορεί να αποτελούνται από άλλα οχήματα, πεζούς ή και σταθερά αντικείμενα όπως κολώνες φωτισμού ή πινακίδες σήμανσης. Τέλος, ακολουθεί το τελευταίο στάδιο το οποίο είναι η αντιστοίχηση των γράφων των οχημάτων που εμπλέκονται σε μια σκηνή. Μέσα από αυτήν τη αντιστοίχηση προκύπτει η σχετική θέση του ενός αυτονόμου οχήματος σε σχέση με ενός άλλου και άρα μπορεί να δημιουργηθεί μια νέα σκηνή που περιέχει πληροφορίες και από τα δύο οχήματα καλύπτοντας τα ενδεχόμενα κενά τους.

EXTENSIVE ENGLISH SUMMARY

POINT CLOUD REGISTRATION USING SEMANTIC SEGMENTATION IN COLLABORATIVE VEHICLES SYSTEMS

Student: Chalkias Konstantinos

Supervisor: Moustakas Konstantinos

Autonomous vehicles and in general autonomous driving, which for many you consider the future of the automotive sector and transport, will soon begin to enter people's daily lives to a large extent, changing their habits radically. These vehicles are equipped with advanced sensors and algorithms that enable them to navigate the roads without human intervention. Although there is no human factor, they can greatly reduce accidents and also increase road safety as in a few years the production number will increase to such an extent that conventional vehicles will not be produced.

This will result in more and more autonomous vehicles, which will have to somehow communicate and exchange information. To do this in an efficient way methods must be developed which will facilitate the sharing of data between vehicles. This sharing of information between vehicles is extremely important, as for practical reasons an autonomous vehicle may not have full knowledge of its environment. Such an example can consist of a pedestrian who is not visible from the perspective of the vehicle or a motorcycle that moves next to a large vehicle (bus, truck, etc.) and prevents the detection of the two-wheeler.

This thesis proposes a way to register point clouds that can solve such a problem using an efficient and effective way of representing the environment in which each autonomous vehicle is located. More specifically, each autonomous vehicle constructs a graph whose vertices are some features of its environment that were produced after semantic segmentation of the scene it is in. These features may consist of other vehicles, pedestrians, or even fixed objects such as light poles or road signs. Finally, the last stage follows which is the matching of the graphs of the vehicles involved in a scene. Through this matching, the relative position of one autonomous vehicle in relation to another is derived, and thus a new scene can be created containing information from both vehicles filling in any gaps.

Περιεχόμενα

Περιεχόμενα	ix
Κατάλογος Σχημάτων	xii
Κατάλογος Πινάκων	xiii
1 Εισαγωγή	1
1.1 Περιγραφή του Προβλήματος	1
1.2 Διάρθρωση της Διπλωματικής Εργασίας	1
2 Θεωρητικό Υπόβαθρο	3
2.1 Αυτόνομα Οχήματα	3
2.1.1 Συνεργατικά Αυτόνομα Οχήματα	4
2.2 Τεχνητή Νοημοσύνη	6
2.2.1 Μηχανική μάθηση	6
2.2.2 Βαθιά Μάθηση	6
2.2.3 Αρχιτεκτονικές Νευρωνικών Δικτύων	6
2.2.4 Βασικές Ορολογίες στην Εκπαίδευση Νευρωνικών Δικτύων	15
2.3 Γεωμετρικοί Μετασχηματισμοί	20
2.3.1 Τύποι Γεωμετρικών Μετασχηματισμών	20
2.4 Μοντελοποίηση Κάμερας	21
2.4.1 Εξωτερικοί Παράμετροι Κάμερας	21
2.4.2 Εσωτερικοί Παράμετροι Κάμερας	22
2.5 Ευθυγράμμιση Νέφους Σημείων	22
2.5.1 Global Registration	23
2.5.2 Local Registration	23
2.6 Ομαδοποίηση 3Δ σημείων	24
2.6.1 DBSCAN	25
2.6.2 HDBSCAN	25
2.7 Γράφοι	26
2.7.1 Αντιστοίχηση Γράφων	27
3 Σχετική Εργασία	29
3.1 Σημασιολογική Κατάτμηση Εικόνας	29
3.1.1 Encoder-Decoder	30
3.1.2 Two Stages Networks	32
3.1.3 Pyramid Pooling Architectures	33
3.1.4 ViT	35
3.2 Σημασιολογική Κατάτμηση Νέφους Σημείων	37

3.2.1	Map 3D to 2D	37
3.2.2	Native segmentation	38
3.3	Ευθυγράμμιση Νέφους Σημείων	40
4	Υλοποίηση	43
4.1	Dataset	43
4.2	Επισκόπηση της υλοποίησης	44
4.2.1	Σημασιολογική Κατάτμησή του Νέφους Σημείων	44
4.2.2	Αναπαράσταση της Σκηνής	51
4.2.3	Ευθυγράμμιση Νέφους Σημείων	54
5	Αποτελέσματα και η Οπτικοποίηση τους	60
5.1	Μεγάλη Επικάλυψη	60
5.1.1	Οπτικοποίηση	61
5.2	Μικρή Επικάλυψη	62
5.2.1	Οπτικοποίηση	62
5.3	Αποτελέσματα	63
6	Επίλογος	70
6.1	Συμπεράσματα	70
6.2	Μελλοντικές Επεκτάσεις	71
Βιβλιογραφία		72
Παράρτημα		76

Κατάλογος Σχημάτων

2.1	Τύποι αισθητήρων σε αυτόνομα οχήματα	4
2.2	Διαδικασία λήψης αποφάσεων σε αυτόνομα οχήματα [1]	4
2.3	Σύστημα με συνεργατικά οχήματα V2V	5
2.4	Η μηχανική μάθηση ως υποσύνολο της Τεχνητής Νοημοσύνης	7
2.5	Μοντέλο ενός perceptron	7
2.6	Αναπαράσταση ενός MLP με ένα κρυφό στρώμα	8
2.7	Ένα τυπικό Συνελικτικό Δίκτυο	9
2.8	Παράδειγμα εφαρμογής της συνέλιξης	10
2.9	Παράδειγμα εφαρμογής της συνέλιξης με διάφορες τιμές της παραμέτρου dilation	10
2.10	Παράδειγμα εφαρμογής του στρώματος Max Pooling και Average Pooling	11
2.11	Scaled Dot-Product Attention	12
2.12	Multi-Head Attention	13
2.13	Αρχιτεκτονική ενός Transformer (από αριστερά είναι κωδικοποιητής ενώ στα δεξιά ο αποκωδικοποιητής)	14
2.14	Αρχιτεκτονική ενός Vision Transformer	15
2.15	Οπτικοποίηση του Dice Coefficient	16
2.16	Οπτικοποίηση του αλγόριθμου gradient-descent	17
2.17	Αλγόριθμος gradient-descent για διάφορους ρυθμούς εκμάθησης	18
2.18	Οπτικοποίηση Overfitting και Underfitting	18
2.19	Οι πιο γνωστές συναρτήσεις ενεργοποίησης	20
2.20	Τύποι Μετασχηματισμών [2]	21
2.21	Προβολή ενός 3D σημείου στο επίπεδο εικόνας σύμφωνα με το Pinhole Camera Model [3]	22
2.22	Ενθυγράμμιση Νέφους Σημείων [4]	23
2.23	Ο αλγόριθμος DBSCAN	25
2.24	Παράδειγμα γράφου και του αντίστοιχου πίνακα γειτνίασης [5]	26
2.25	Βοηθητικοί γράφοι για την αντιστοίχηση χρησιμοποιώντας τον αλγόριθμο RRWM	28
3.1	Αρχιτεκτονική του δικτύου FCN	30
3.2	Αρχιτεκτονική ενός δικτύου U-Net	31
3.3	Αρχιτεκτονική ενός δικτύου Mask R-CNN	32
3.4	Αρχιτεκτονική του Pyramid Scene Parsing Network	33
3.5	Αρχιτεκτονική του δικτύου DeepLab V3	34
3.6	Αρχιτεκτονική του δικτύου SegFormer	35
3.7	Αρχιτεκτονική του δικτύου SETR	36
3.8	Οι παραλλαγές του αποκωδικοποιητή στο δίκτυο SETR	36
3.9	Σημασιολογική κατάτμηση νέφους σημείων χρησιμοποιώντας κατάτμηση εικόνας	37
3.10	Αρχιτεκτονική του δικτύου PointNet++	38

3.11	Αρχιτεκτονική του δικτύου PointNet	39
3.12	Αρχιτεκτονική του δικτύου Cylinder3D	39
3.13	Αναλυτικά τα τμήματα του Cylinder3D	40
3.14	Αρχιτεκτονική της μεθόδου DEEPSIR	41
3.15	Διαδικασία ευθυγράμμισης του προτάθηκε από το Segregator	42
4.1	Εικόνες από τις κάμερες ενός οχήματος	44
4.2	Η διάταξη των αισθητήρων στο όχημα και στο δρόμο για το Dataset V2X-Sim, αντίστοιχα	44
4.3	Παράδειγμα λάθους κατηγοριοποίησης σημείου	46
4.4	Προβολή του νέφους σημείων στο επίπεδο εικόνας για την μπροστινή κάμερα (το χρώμα κάθε σημείου αντιπροσωπεύει την απόσταση από την κάμερα)	46
4.5	Παράδειγμα λάθος κατηγοριοποίησης χρησιμοποιώντας προεκπαιδευμένα δίκτυα	47
4.6	Γραφικές εκπαίδευσης για το PSPNet	48
4.7	Γραφικές εκπαίδευσης για το DeepLabv3+	49
4.8	Γραφικές εκπαίδευσης για το SegFormer	50
4.9	Οι μάσκες κατάτμησης για όλες τις εικόνες ενός οχήματος που προέκυψαν μετά την επιπλέον εκπαίδευση του DeepLabv3+	51
4.10	Κατάτμηση νέφους σημείων χρησιμοποιώντας τις μάσκες κατάτμησης από τις κάμερες	51
4.11	Παράδειγμα ομοιόποιησης σε σημεία κατηγοριοποιημένα ως οχήματα	52
4.12	Σύγκριση παραμέτρων για το αλγόριθμο HDBSCAN και οπτικοποίηση των αποτελεσμάτων	53
4.13	Διαδικασία για τη δημιουργία του γράφου σκηνής που βλέπει το όχημα	53
4.14	Συνάρτηση ομοιότητας 4.1 για διάφορες τιμές της παραμέτρου “σ”	55
4.15	Οι γράφοι σκηνής για δύο οχήματα για την ίδια χρονική στιγμή	55
4.16	Πίνακας ομοιότητας και το αντίστοιχο αποτέλεσμα του αλγορίθμου RRWM	56
4.17	Οπτικοποίηση της αντιστοίχησης	56
4.18	Αποτέλεσμα ευθυγράμμισης μετά την αντιστοίχηση γράφων	57
4.19	Οπτικοποίηση της τελικής ευθυγράμμισης	58
4.20	Διαδικασία για την ευθυγράμμιση των νεφών σημείων χρησιμοποιώντας το γράφο σκηνής	59
5.1	Τροχιές όλων των οχημάτων	60
5.2	Αντιστοίχηση γράφων σε παράδειγμα με μεγάλο ποσοστό επικάλυψης	61
5.3	Αποτελέσματα ευθυγράμμισης πριν και μετά τον ICP για παράδειγμα με μεγάλο ποσοστό επικάλυψης	61
5.4	Αντιστοίχηση γράφων στο πρώτο παράδειγμα με μικρό ποσοστό επικάλυψης	62
5.5	Αποτελέσματα ευθυγράμμισης πριν και μετά τον ICP για το πρώτο παράδειγμα με μικρό ποσοστό επικάλυψης	62
5.6	Αντιστοίχηση γράφων στο δεύτερο παράδειγμα με μικρό ποσοστό επικάλυψης	63
5.7	Αποτελέσματα ευθυγράμμισης πριν τον ICP και η και η ground truth ευθυγράμμιση	63
5.8	Σφάλματα ευθυγράμμισης για το ζεύγος οχημάτων 1,4	64
5.9	Σφάλματα ευθυγράμμισης σε παραδείγματα με μικρή επικάλυψη	65
5.10	Σφάλματα ευθυγράμμισης για το ζεύγος οχημάτων 4,5	66
5.11	Κατανομή σφαλμάτων πριν την εφαρμογή του ICP	67
5.12	Κατανομή σφαλμάτων για διάφορες τιμές του κατωφλίου στον αλγόριθμο ICP	68

Κατάλογος Πινάκων

4.1	Αποτελέσματα κατάτμησης του δικτύου PSPNet στο test set	48
4.2	Αποτελέσματα κατάτμησης του δικτύου DeepLabv3+ στο test set	49
4.3	Αποτελέσματα κατάτμησης του δικτύου SegFormer στο test set	50
4.4	Σύγκριση των αποτελεσμάτων ανάμεσα στα δίκτυα που χρησιμοποιήθηκαν για αρχικοποίηση των βαρών και αυτών μετά το transfer learning	51
5.1	Σφάλματα ευθυγράμμισης σε παραδείγματα με μικρό και μεγάλο ποσοστό επικάλυψης	64
5.2	Σφάλματα ευθυγράμμισης (συγκεντρωτικός πίνακας)	67

Κεφάλαιο 1

Εισαγωγή

1.1 Περιγραφή του Προβλήματος

Φανταστείτε ότι κάποια αυτόνομα οχήματα (AVs) βρίσκονται σε μια διασταύρωση και το κάθε ένα βλέπει την ίδια διασταύρωσή από διαφορετική οπτική γωνία. Αυτό έχει σαν συνέπεια να μην έχουν όλα τα οχήματα άμεση πρόσβαση σε ολόκληρη τη σκηνή (διασταύρωση) καθώς κάποια χρήσιμα αντικείμενα της σκηνής (πινακίδες, φανάρια, οχήματα, πεζοί, κτλ) κρύβονται πίσω από άλλα αντικείμενα ή κτίρια. Όμως αντά τα χρήσιμα αντικείμενα μπορεί να είναι ορατά από κάποια άλλη οπτική γωνία που μπορεί να έχει κάποιο άλλο αυτόνομο όχημα. Αυτό το σενάριο αποτελεί ένα σύνηθες περιστατικό σε αστικά περιβάλλοντα όπου για παράδειγμα ένα ογκώδες όχημα (φορτηγό, λεωφορείο) κρύβει έναν μοτοσικλετιστή ή έναν πεζό.

Στην εποχή που ζούμε μπορεί να μην είναι πολύ πιθανό να βρίσκονται στην ίδια διασταύρωση δύο αυτόνομα οχήματα άλλα μετά από δέκα ή και λιγότερο χρόνια μπορεί αυτό το σενάριο να αποτελεί την κανονικότητα και όχι κάτι το ασυνήθιστο. Σύμφωνα με την έρευνά [6] το 2022 η παγκόσμια αγορά των ημι-αυτόνομων και αυτόνομων οχημάτων ήταν στα 121.78B USD και προβλέπεται να αγγίξει το 2032 τα 2,356.93B USD, δηλαδή 20 φορές το τωρινό μέγεθος. Αυτό έχει επακόλουθου όλο και πιο πολλές εταιρίες να εισέρχονται στον κλάδο της αυτόνομης οδήγησης και να παίρνουν άδειες για δοκιμές των οχημάτων τους σε δημόσιους δρόμους [7]. Με αυτόν τον τρόπο το ποσοστό των αυτόνομων οχημάτων συνεχώς αυξάνεται και αυτό έχει σαν επακόλουθο την ανάγκη για ανάπτυξη νέων μεθόδων και λύσεων.

Η παρούσα εργασία αποτελεί μία προσπάθεια επίλυσης του προβλήματος που περιγράφηκε παραπάνω χρησιμοποιώντας μία πιο γενική και αφηρημένη αναπαράσταση της σκηνής με σκοπό να γίνει πιο εύκολη η εύρεση των σχετικών θέσεων των αυτόνομων οχημάτων μέσα στη σκηνή και κατά επέκταση ο μετασχηματισμός της πληροφορίας από το ένα σύστημα συντεταγμένων του ενός οχήματος στο άλλο.

1.2 Διάρθρωση της Διπλωματικής Εργασίας

Ακολουθεί μια περιγραφή για τα περιεχόμενα του κάθε κεφαλαίου και της δομής της υπόλοιπης εργασίας. Ξεκινώντας από το Κεφάλαιο 2 το οποίο αποτελείται από την περιγραφή των βασικών εννοιών και την παρουσίαση των αναγκαίων γνώσεων για την κατανόηση της εργασίας. Μέσα σε αυτό κεφάλαιο εξηγείται μεταξύ άλλων τι είναι η μηχανική μάθηση, πώς προσομοιώνεται μια κάμερα αλλά και το τι είναι ευθυγράμμιση νέφους σημείων. Ακολουθεί το Κεφάλαιο 3, όπου περιγράφονται και αναλύονται σχετικές εργασίες από άλλους ερευνητές, οι οποίες χρησιμοποιήθηκαν ή έπαιξαν σημαντικό ρόλο κατά τη διαδικασία της υλοποίησης. Στη συνέχεια

στο Κεφάλαιο 4 περιγράφεται η προσέγγιση που ακολουθήθηκε για την επίλυση του προβλήματος συνοδευόμενο με παρουσιάσεις όλων των υπο-προβλημάτων και τους τρόπους επίλυσης αυτών. Ακολουθεί το Κεφάλαιο 5 όπου παρουσιάζονται τα αποτελέσματα της υλοποίησης του προηγούμενου κεφαλαίου μαζί με τις κατάλληλες οπτικοποιήσεις για το κάθε υπο-πρόβλημα. Τέλος, το Κεφάλαιο 6 είναι ο επίλογος, ο οποίος περιέχει μια ανακεφαλαίωση όλων των προηγούμενων ενοτήτων μαζί με προτάσεις για μελλοντικές επεκτάσεις.

Κεφάλαιο 2

Θεωρητικό Υπόβαθρο

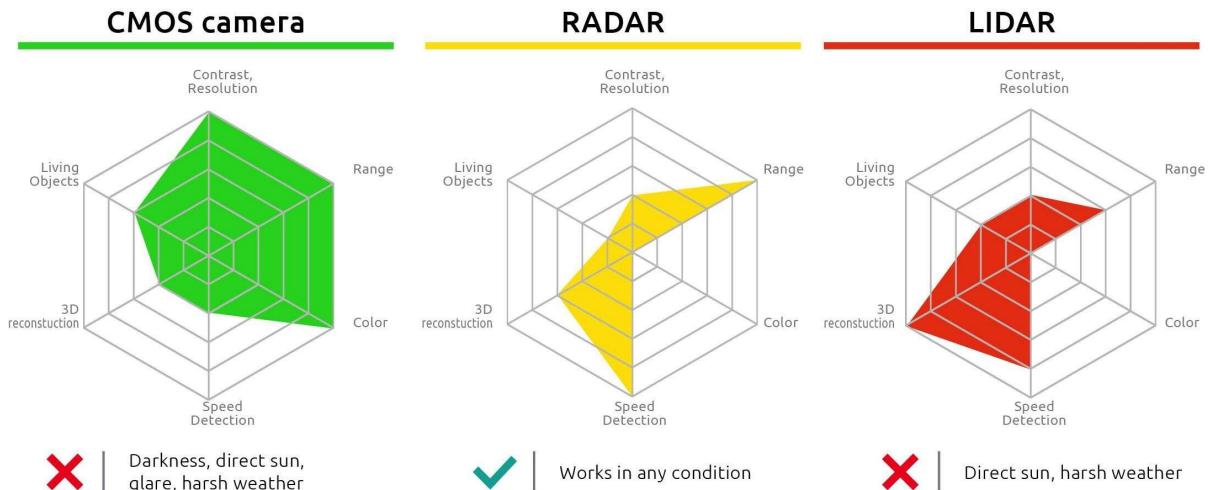
2.1 Αυτόνομα Οχήματα

Τα αυτόνομα οχήματα αντιπροσωπεύουν μια επαναστατική εξέλιξη στον τομέα της οδήγησης. Αυτού του είδους οχήματα είναι εξοπλισμένα με προηγμένες τεχνολογίες και αισθητήρες και μπορούν να λειτουργούν χωρίς την παρέμβαση ανθρώπου. Αυτή η εξέλιξη στο χώρο της μετακίνησης ανοίγει νέες προοπτικές στον χώρο των μεταφορών, με αλλαγές σε πολλούς τομείς, συμπεριλαμβανομένων των ορίων ασφάλειας που θέτουν οι θεσμοί στις αυτοκινητοβιομηχανίες άλλα και στις δυνατότητες του βασικού εξοπλισμού και των συμπεριλαμβανομένων αισθητήρων που θα διαθέτει να όχημα.

Οι αισθητήρες αποτελούν την “έξυπνη” προέκταση των αυτόνομων οχημάτων, επιτρέποντάς τους να αντιλαμβάνονται το περιβάλλον τους με ακρίβεια και αξιοπιστία. Ανάμεσα στους αισθητήρες που χρησιμοποιούνται συχνά σε αυτά τα οχήματα περιλαμβάνονται οι κάμερες, οι ραντάρ, οι λέιζερ και οι αισθητήρες υπερήχων. Μέσω της συνεργασίας αυτών των αισθητήρων, τα αυτόνομα οχήματα αντλούν δεδομένα για το περιβάλλον τους και λαμβάνουν αποφάσεις που επηρεάζουν την αποτελεσματικότητα τους άλλα και την ασφάλεια των επιβατών [8]. Πιο αναλυτικά:

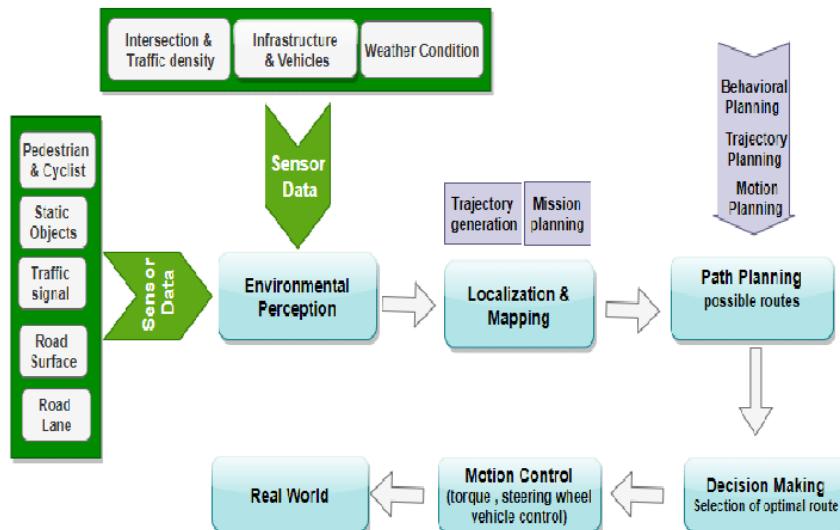
- **Κάμερες:** Οι κάμερες χρησιμοποιούνται για τη συλλογή εικόνων του περιβάλλοντος του οχήματος. Οι εικόνες αυτές μπορούν να χρησιμοποιηθούν για την αναγνώριση άλλων οχημάτων, πεζών και εμποδίων.
- **Ραντάρ ή Σόναρ:** Τα ραντάρ χρησιμοποιούνται για τη μέτρηση της απόστασης και της ταχύτητας των αντικειμένων στο περιβάλλον του οχήματος. Μπορούν επίσης να χρησιμοποιηθούν για την ανίχνευση αντικειμένων που δεν είναι ορατά στις κάμερες, όπως οχήματα που βρίσκονται πίσω από άλλα οχήματα ή όταν επικρατούν δυσμενείς καιρικές συνθήκες.
- **Λέιζερ (LIDAR):** Τα λέιζερ χρησιμοποιούνται για τη δημιουργία τρισδιάστατων χαρτών του περιβάλλοντος του οχήματος. Αυτοί οι χάρτες μπορούν να χρησιμοποιηθούν για τον σχεδιασμό της πορείας του οχήματος, για την αποφυγή εμποδίων αλλά και αναγνώριση αντικείμενων.

Η λήψη αποφάσεων είναι ένα από τα βασικά συστατικά της αυτόνομης οδήγησης και επιτυγχάνεται μέσω της επεξεργασίας των δεδομένων από τους αισθητήρες [1]. Οι αισθητήρες συλλέγουν πληροφορίες για το περιβάλλον του οχήματος, όπως η θέση και η ταχύτητα των άλλων οχημάτων, οι συνθήκες του δρόμου και οι πιθανοί κίνδυνοι. Αυτές οι πληροφορίες στη



Σχήμα 2.1: Τύποι αισθητήρων σε αυτόνομα οχήματα

συνέχεια επεξεργάζονται από ένα σύστημα, το οποίο υπολογίζει την καλύτερη πορεία για το οχηματα και βάσης αυτής της απόφασης δρα ανάλογα το οχηματα.



Σχήμα 2.2: Διαδικασία λήψης αποφάσεων σε αυτόνομα οχήματα [1]

2.1.1 Συνεργατικά Αυτόνομα Οχήματα

Τα συνεργατικά αυτόνομα οχήματα είναι μια νέα τεχνολογία που έχει την δυνατότητα να αλλάξει ριζικά τον τρόπο με τον οποίο μετακινούμαστε. Αυτού του είδους οχήματα μπορούν να συνεργάζονται μεταξύ τους μοιράζοντας πληροφορία για την κατάσταση της κυκλοφορίας, τα αντικείμενα που έχουν εντοπίσει στο περιβάλλον τους άλλα και για την διαδρομή που θέλουν να ακολουθήσουν. Με αυτόν τον τρόπο τα άλλα οχήματα μπορούν να παίρνουν καλύτερες αποφάσεις γνωρίζοντας εκ των προτέρων για την κατάσταση του χώρου (που βρίσκονται πεζοί, οχήματα κ.α.) αλλά και του δρόμου (προβλεπόμενη πορεία άλλων οχημάτων). Υπάρχουν τρεις μεγάλες κατηγορίες τέτοιων οχημάτων και χωρίζονται ανάλογα με ποια άλλα αντικείμενα ανταλλάσσουν πληροφορία [9]. Αναλυτικά:

- Η επικοινωνία **V2I (Vehicle-to-Infrastructure)** αναφέρεται στην επικοινωνία μεταξύ οχημάτων και οδικής υποδομής, όπως φανάρια, οδικές πινακίδες και αισθητήρες. Αυτή η επικοινωνία μπορεί να χρησιμοποιηθεί για να παρέχει στα οχήματα πληροφορίες σε πραγματικό χρόνο σχετικά με τις κυκλοφοριακές συνθήκες, το κλείσιμο δρόμων και τις καιρικές συνθήκες. Για παράδειγμα, ένα όχημα που πλησιάζει σε κόκκινο φανάρι μπορεί να λάβει ένα σήμα από το φανάρι που του λέει πότε το φως πρόκειται να ανάψει πράσινο.
- Η επικοινωνία **V2V (Vehicle-to-Vehicle)** αναφέρεται στην άμεση επικοινωνία μεταξύ οχημάτων. Αυτή η επικοινωνία μπορεί να χρησιμοποιηθεί για την ανταλλαγή πληροφοριών σχετικά με τις θέσεις, τις ταχύτητες και τις προθέσεις των οχημάτων, κάτι που μπορεί να βοηθήσει στην πρόληψη ατυχημάτων. Για παράδειγμα, εάν ένα όχημα ανιχνεύσει ότι πρόκειται να συγκρουστεί με άλλο όχημα, μπορεί να στείλει ένα προειδοποιητικό σήμα στο άλλο όχημα, ώστε ο οδηγός να λάβει μέτρα αποφυγής.
- Το **V2X (Vehicle-to-Everything)** είναι ένας ευρύς όρος που περιλαμβάνει όλους τους τύπους επικοινωνίας από όχημα σε όχημα (V2V) και από όχημα σε υποδομή (V2I). Η τεχνολογία V2X επιτρέπει στα οχήματα να μοιράζονται πληροφορίες μεταξύ τους και με το περιβάλλον τους, όπως οδικές συνθήκες, κινδύνους στην κυκλοφορία και καταστάσεις έκτακτης ανάγκης. Αυτές οι πληροφορίες μπορούν να χρησιμοποιηθούν για τη βελτίωση της ασφάλειας και της αποτελεσματικότητας.
- Η επικοινωνία **V2P (Vehicle-to-Pedestrian)** ή **V2D (Vehicle-to-Device)** η οποία επιτρέπει την επικοινωνία με ευάλωτους χρήστες του δρόμου, όπως πεζούς και ποδηλάτες, για να βελτιώσουν την ασφάλεια.

Η ανάπτυξη των συνεργατικών οχημάτων έχει πολλά οφέλη, αλλά και ορισμένες προκλήσεις. Μία από τις πιο σημαντικές προκλήσεις είναι η διαχείριση του εύρους ζώνης του δικτύου. Τα συνδεδεμένα οχήματα πρέπει να ανταλλάσσουν δεδομένα με άλλα οχήματα και με την υποδομή του δρόμου, κάτι που απαιτεί ένα μεγάλο εύρος ζώνης. Εάν το εύρος ζώνης δεν είναι επαρκές, μπορεί να προκληθεί συμφόρηση του δικτύου, η οποία θα μπορούσε να οδηγήσει σε καθυστερήσεις και ατυχήματα. Ωστόσο, αυτή η τεχνολογία έχει τη δυνατότητα να εξελιχθεί σημαντικά τα επόμενα χρόνια βελτιώνοντας την ασφάλεια στους δρόμους και αλλάζοντας ριζικά τον τρόπο με τον οποίο μετακινούμαστε.



Σχήμα 2.3: Σύστημα με συνεργατικά οχήματα V2V

2.2 Τεχνητή Νοημοσύνη

Ο όρος τεχνητή νοημοσύνη (Artificial Intelligence) αναφέρεται στον κλάδο της πληροφορικής ο οποίος ασχολείται με τη σχεδίαση και την υλοποίηση υπολογιστικών συστημάτων και αλγορίθμων που μιμούνται στοιχεία της ανθρώπινης συμπεριφοράς τα οποία υπονοούν έστω και στοιχειώδη ευφυΐα. Κάποια από αυτά τα στοιχεία είναι η μάθηση, εξαγωγή συμπερασμάτων, κατανόηση από συμφραζόμενα, επίλυση προβλημάτων κλπ.

2.2.1 Μηχανική μάθηση

Η μηχανική μάθηση (Machine Learning) έχει τις ρίζες της στις αρχές του 20ου αιώνα, αλλά άρχισε να αναπτύσσεται δυναμικά τα τελευταία 30 χρόνια και αποτελεί έναν κλάδο της επιστήμης των υπολογιστών που ασχολείται με την ανάπτυξη αλγορίθμων που μπορούν να μάθουν από δεδομένα χωρίς να προγραμματιστούν εξ ολοκλήρου από ανθρώπους. Η μηχανική μάθηση χρησιμοποιείται σε μια ευρεία γκάμα εφαρμογών από την αναγνώριση εικόνων έως και την πρόβλεψη πωλήσεων αλλά και την εξατομικευμένη ιατρική διάγνωση.

Οι εργασίες μηχανικής μάθησης συνήθως ταξινομούνται σε τρεις μεγάλες κατηγορίες, ανάλογα με την «ανατροφοδότηση» που είναι διαθέσιμη σε ένα σύστημα εκμάθησης. Αυτές είναι:

- Η **επιτηρούμενη μάθηση** είναι ένα είδος μηχανικής μάθησης στο οποίο ένα υπολογιστικό πρόγραμμα εκπαιδεύεται με ένα σύνολο δεδομένων που περιλαμβάνει δείγματα εισόδου (data) και επιθυμητά αποτελέσματα (label). Ο στόχος της επιτηρούμενης μάθησης είναι να δημιουργήσει ένα μοντέλο που μπορεί να προβλέψει τα επιθυμητά αποτελέσματα για νέες εισόδους.
- Η **μη επιτηρούμενη μάθηση** είναι η κατηγορία της μηχανικής μάθησης, στην οποία ο αλγόριθμος μάθησης δεν παρέχεται με προκατασκευασμένα αποτελέσματα (data-label) για να μάθει. Αντί αυτού, το μοντέλο πρέπει να ανακαλύψει μόνο του μοτίβα και συσχετίσεις στα δεδομένα εισόδου και να εξάγει τα συμπεράσματα.
- Η **ενισχυτική μάθηση** είναι μια τεχνική μηχανικής μάθησης στην οποία ένα σύστημα μαθαίνει να παίρνει αποφάσεις μέσα από την αλληλεπίδραση με το περιβάλλον του. Το σύστημα λαμβάνει μια ανταμοιβή ή τιμωρία για κάθε απόφαση που παίρνει, και χρησιμοποιεί αυτές τις πληροφορίες για να βελτιώσει τις μελλοντικές του αποφάσεις.

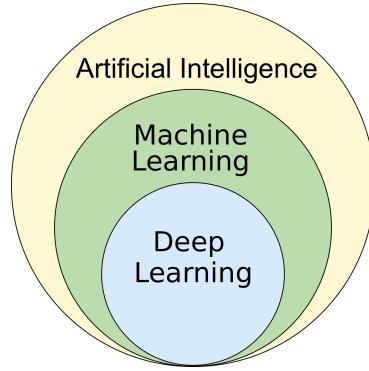
2.2.2 Βαθιά Μάθηση

Η βαθιά μάθηση (Deep Learning) είναι ένας τύπος μηχανικής μάθησης που χρησιμοποιεί τεχνητά νευρωνικά δίκτυα για να εκπαιδεύσει συστήματα να αναγνωρίζουν μοτίβα σε δεδομένα. Η βαθιά μάθηση είναι “βαθιά” επειδή χρησιμοποιεί πολλαπλά επίπεδα νευρώνων. Αυτό επιτρέπει στο σύστημα να μαθαίνει πιο σύνθετα μοτίβα από ό,τι θα μπορούσε με ένα μόνο επίπεδο νευρώνων. Κάποιες βασικές αρχιτεκτονικές δίκτυων βαθιάς μάθησης που χρησιμοποιούνται ευρέως στην μηχανική όραση αποτελούν τα συνελικτικά νευρωνικά δίκτυα (CNN) και οι μετασχηματιστές (Transformers).

2.2.3 Αρχιτεκτονικές Νευρωνικών Δικτύων

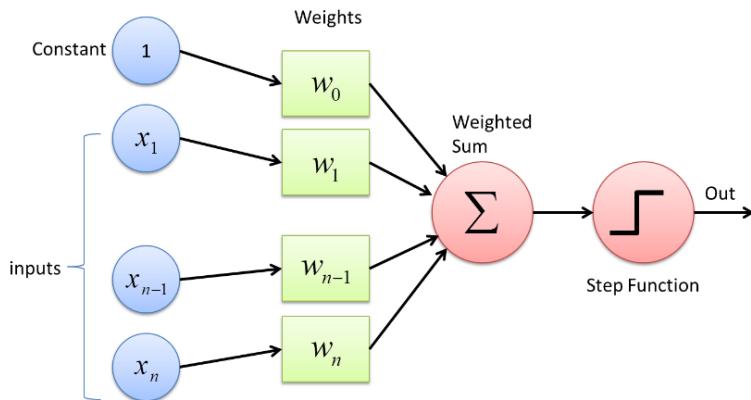
Perceptron

Στη μηχανική μάθηση, το perceptron (ή ο νευρώνας McCulloch-Pitts) [10] είναι ένας αλγόριθμος για την εποπτευόμενη εκμάθηση δυαδικών ταξινομητών. Ένας δυαδικός ταξινομητής



Σχήμα 2.4: Η μηχανική μάθηση ως υποσύνολο της Τεχνητής Νοημοσύνης

είναι μια συνάρτηση που μπορεί να αποφασίσει εάν μια είσοδος, που αντιπροσωπεύεται από ένα διάνυσμα αριθμών, ανήκει ή όχι σε κάποια συγκεκριμένη κλάση. Είναι ένας τύπος γραμμικού ταξινομητή, δηλαδή ένας αλγόριθμος ταξινόμησης που κάνει τις προβλέψεις του με βάση μια συνάρτηση γραμμικής πρόβλεψης που συνδυάζει ένα σύνολο βαρών με το διάνυσμα χαρακτηριστικών [11]. Αρχικά perceptron χρησιμοποιούσε μια συνάρτηση βήματος Heaviside (step function) για συνάρτηση ενεργοποίησης, αλλά πλέον στα πιο σύγχρονα μοντέλα έχει αντικατασταθεί από άλλες γραμμικές και μη συναρτήσεις.



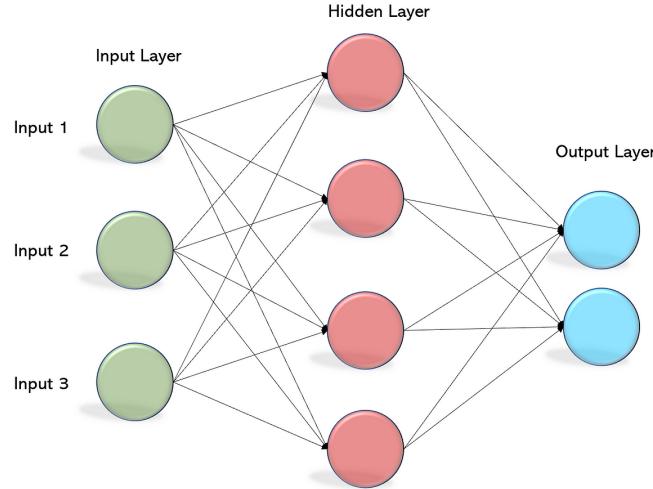
Σχήμα 2.5: Μοντέλο ενός perceptron

Dense Layer και MLP

Ένα MLP (Multilayer perceptron) ή Dense Layer είναι η εξέλιξη ενός κλασικού perceptron και μπορεί να θεωρηθεί το πρώτο είδος νευρωνικού δικτύου. Πρωτοεμφανίστηκε στις αρχές του 1960 [12], [13] και αποτελείται από πλήρως συνδεδεμένους νευρώνες - κόμβους οργανωμένους σε τουλάχιστον τρία επίπεδα (στρώμα εισόδου, κρυφό στρώμα, στρώμα εξόδου) και με τουλάχιστον ένα κρυφό στρώμα. Το κρυφό στρώμα ονομάζεται έτσι, καθώς οι κομβοί οι οποίοι περιέχονται δεν συνδέονται άμεσα με το διάνυσμα εισόδου ή με την έξοδο. Παίζουν πολύ σημαντικό ρόλο καθώς επεξεργάζονται και μετασχηματίζουν τα δεδομένα εισόδου με τέτοιο τρόπο ώστε να μπορούν να βρουν σύνθετα μοτίβα και να μπορούν να εξάγουν συμπεράσματα μέσα από αυτά.

Οι νευρώνες κάθε στρώματος λειτουργούν παρόμοια με τους νευρώδους του εγκεφάλου, δηλαδή λαμβάνουν, επεξεργάζονται και στέλνουν πληροφορία σε άλλους νευρώνες - κόμβους

που είναι συνδεδεμένοι. Μόλις ένας νευρώνας επεξεργαστεί την είσοδο που έλαβε, την τροφοδότη στην έξοδό του και γίνεται είσοδος σε άλλους νευρώνες ή πέρνα στο διάνυσμα εξόδου του δικτύου. Στο σχήμα 2.6 φαίνεται ένα δίκτυο με ένα κρυφό στρώμα και μέγεθος διανύσματος εισόδου και εξόδου τρία και δύο, αντίστοιχα.



Σχήμα 2.6: Αναπαράσταση ενός MLP με ένα κρυφό στρώμα

Ένα δίκτυο μπορεί εύκολα να περιγράφει μαθηματικά με διαδοχικούς πολλαπλασιασμούς πινάκων. Ακολουθεί το μαθηματικό μοντέλο για ένα τυχαίο στρώμα (κρυφό ή και όχι), εξίσωση: 2.1.

$$Y = f(WX) \quad (2.1)$$

όπου:

X : Διάνυσμα εισόδου

W : Πίνακας με τα βάρη του αντίστοιχου στρώματος

$f(\cdot)$: Συνάρτηση ενεργοποίησης

Y : Διάνυσμα εξόδου

Για παράδειγμα εάν θέλουμε να μοντελοποιήσουμε το πρώτο κρυφό στρώμα εισόδου του σχήματος 2.6 θα είχαμε:

$$X = (i_1, i_2, i_3)^T \quad (2.2)$$

$$Y = (y_1, y_2, y_3, y_4)^T \quad (2.3)$$

$$W = (w_1, w_2, w_3) = \begin{pmatrix} w_1^1 & w_1^2 & w_1^3 \\ w_2^1 & w_2^2 & w_2^3 \\ w_3^1 & w_3^2 & w_3^3 \\ w_4^1 & w_4^2 & w_4^3 \end{pmatrix} \quad (2.4)$$

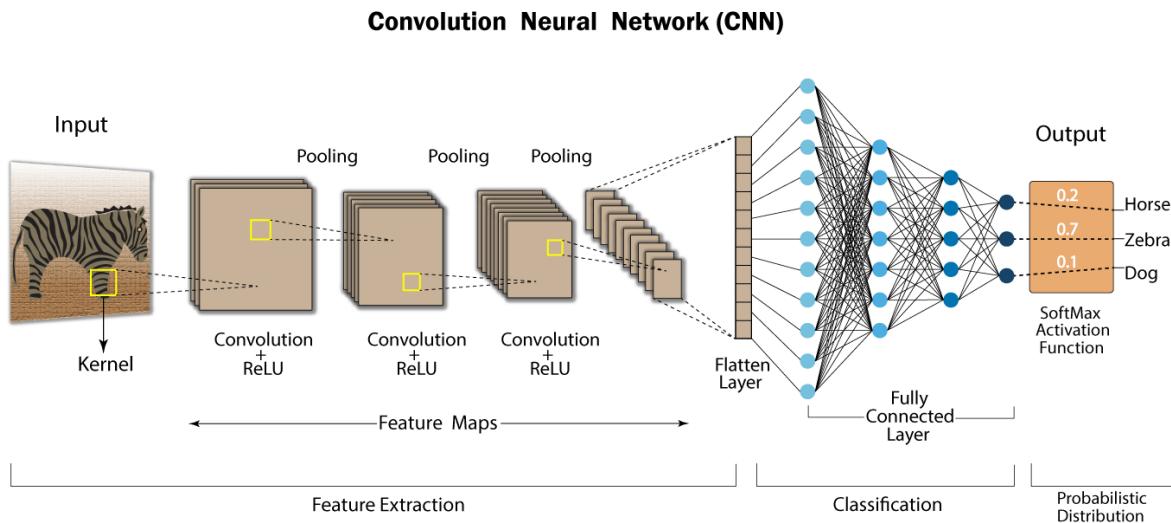
όπου:

w_i^j : Το βάρος του νευρώνα i που τον ενώνει με τον νευρώνα j του προηγούμενου στρώματος

Convolutional Neural Network

To Convolutional Neural Network ή CNN είναι ένα είδος νευρωνικού δικτύου που εμφανίστηκε στις αρχές του 1990 [14]. Είναι ιδιαίτερα κατάλληλο και προτιμάτε σε εφαρμογές που έχουν

σχέση με την αναγνώριση αντικείμενων σε εικόνες αλλά και με το γενικότερο κλάδο της επεξεργασία εικόνας. Ως επί το πλείστον αποτελείται από τρεις κατηγορίες στρωμάτων που το ένα ακολουθεί το άλλο (2.7). Αυτά τα στρωματά είναι το συνεκτικό στρώμα (convolution layer), το στρώμα συμπίεσης (pooling layer) και το πλήρως συνδεδεμένο στρώμα (fully connected layer). Τα CNN είναι σε θέση να μαθαίνουν πολύπλοκα μοτίβα λόγω της δομής τους από εικόνες και είναι σε θέση να ταξινομούν εικόνες με βάση το περιεχόμενο τους έχοντας υψηλό ποσοστό επιτυχίας.



Σχήμα 2.7: Ένα τυπικό Συνελικτικό Δίκτυο

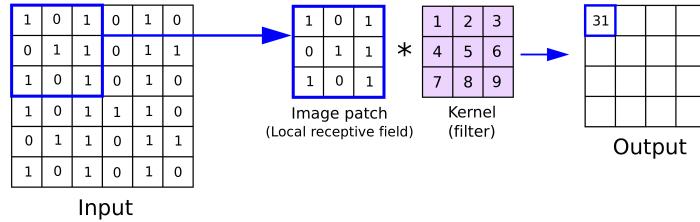
Πιο αναλυτικά για το κάθε στρώμα:

- Τα **Convolution layers** είναι τα θεμελιώδη δομικά στοιχεία των συνελικτικών νευρωνικών δικτύων (CNN). Είναι υπεύθυνα για την εξαγωγή χαρακτηριστικών από εικόνες (feature maps), στα οποία θα βασιστεί η διαδικασία της αναγνώρισης (image recognition) ή όποιας άλλης λειτουργίας εξυπηρετεί το δίκτυο. Η βασική λειτουργία ενός συνεκτικού επιπέδου είναι η λειτουργία συνέλιξης, η οποία είναι μία μαθηματική τεχνική για την ανάλυση και τον μετασχηματισμό δεδομένων 2.8. Αν και στην πραγματικότητα η μαθηματική πράξη που εκτελείται δεν είναι συνέλιξη αλλά ετεροσυσχέτιση (Cross-correlation) καθώς δεν γίνεται κάποια αντιστροφή στο φίλτρο όπως ορίζει ο ακριβής μαθηματικός ορισμός. Στην πράξη όμως έχει επικρατήσει ο όρος της συνέλιξης όταν αναφερόμαστε σε CNN.

Η λειτουργία συνέλιξης περιλαμβάνει την ολίσθηση ενός φίλτρου (kernel) πάνω από μια εικόνα εισόδου. Το φίλτρο είναι μια μικρή μήτρα βαρών που αντιπροσωπεύει ένα συγκεκριμένο χαρακτηριστικό ενδιαφέροντος, όπως άκρες, γωνίες ή υφές. Καθώς το φίλτρο κινείται κατά μήκος της εικόνας, πολλαπλασιάζεται στοιχείο προς στοιχείο με τα αντίστοιχα pixels στην εικόνα εισόδου. Τα αποτελέσματα αυτών των πολλαπλασιασμών αθροίζονται για να δημιουργήσουν μια ενιαία τιμή.

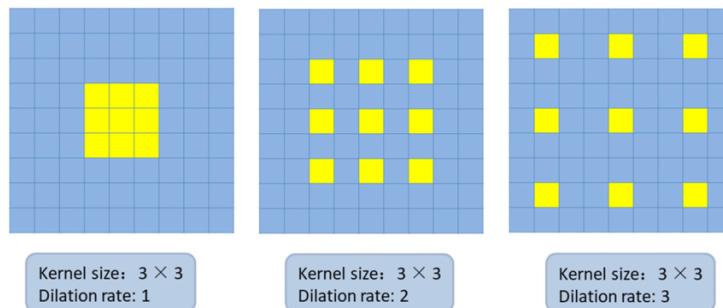
Βασικές παράμετροι ενός στρώματος συνέλιξης είναι stride, padding και το dilation rate. Όλοι αυτοί οι παράμετροι επηρεάζουν το μέγεθος του feature map σύμφωνα με την εξισώση: κανις

$$output_{size_i} = \frac{input_{size_i} - filter + 2 \cdot padding}{stride} + 1 \quad (2.5)$$



Σχήμα 2.8: Παράδειγμα εφαρμογής της συνέλιξης

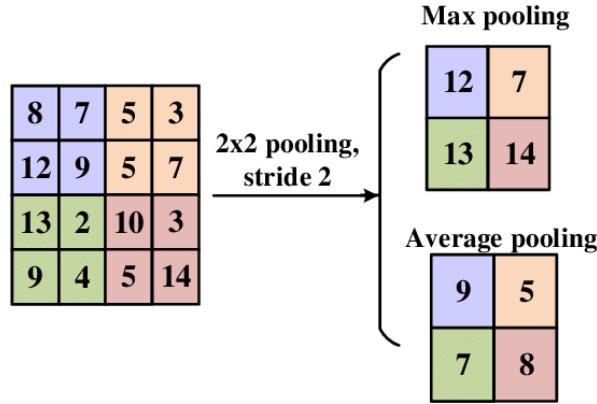
- **Stride:** Καθορίζει τον αριθμό των pixel που μεταπηδά το φίλτρο μεταξύ των υπολογισμών. Όταν το stride είναι 1 σημαίνει ότι το φίλτρο μετακινείται ένα pixel κάθε φορά, ενώ όταν είναι 2 σημαίνει ότι παρακάμπτει ένα pixel μεταξύ κάθε εφαρμογής.
- **Padding:** Προσθήκη γραμμών και στηλών με μηδενικά γύρω από την εικόνα εισόδου για τη διαχείριση του μεγέθους του feature map, δηλαδή το padding αποτρέπει τη συρρίκνωση του feature map κατά την διάρκεια εφαρμογής ενός φίλτρου.
- **Dilation:** Επέκταση του μεγέθους του φίλτρου προσθέτοντας κενά μεταξύ των τιμών του για την καταγραφή μεγαλύτερων χαρακτηριστικών [15]. Αυτή η παράμετρος είναι αρκετά χρήσιμη σε εφαρμογές σημασιολογικής κατάτμησης [16].



Σχήμα 2.9: Παράδειγμα εφαρμογής της συνέλιξης με διάφορες τιμές της παραμέτρου dilation

- Τα **pooling layers** χρησιμοποιούνται για τη μείωση της διάστασης των feature maps αλλά και για ανθεκτικότητα σε μικρές αλλαγές. Με αυτόν τον τρόπο μειώνονται οι υπολογισμοί στα επόμενα στρώματα του δικτύου και οι εκπαιδεύσιμοι παράμετροι του δικτύου. Η συνάρτηση υπεύθυνη για το πως θα γίνει η μείωση των διαστάσεων εφαρμόζεται σε μικρές περιοχής του feature map και μπορεί να υπολογίζει τη μέγιστη τιμή, το μέσο όρο ή κάποιο σταθμισμένο μέσω όρο αυτής της υποπεριοχής χρησιμοποιώντας κάποια σταθερά βάρη.
- Το **Fully Connected layer** είναι το τελευταίο στρώμα σε ένα CNN και παίζουν κρίσιμο ρόλο στη μετάβαση από τους feature maps στην τελική κατηγοριοποίηση ή στην τελική έξοδο του δικτύου, χρησιμοποιώντας τις πληροφορίες που συλλέχθηκαν από τα προηγούμενα συνελικτικά επίπεδα.

Εν ολίγοις, τα συνελικτικά νευρωνικά δίκτυα ή CNN έχουν φέρει επανάσταση στο πεδίο της υπολογιστικής όρασης, επιτρέποντας στις μηχανές να αντιλαμβάνονται και να κατανοούν τον οπτικό κόσμο με αρκετά καλή ακρίβεια. Η ικανότητά τους να εξάγουν περίπλοκα μοτίβα μέσα από εικόνες άνοιξε το δρόμο για πρωτοποριακές εφαρμογές σε διάφορους τομείς. Αν και



Σχήμα 2.10: Παράδειγμα εφαρμογής των στρώματος Max Pooling και Average Pooling

πλέον έχουν ανακαλυφθεί νέες αρχιτεκτονικές δικτύων όπως οι ViT με συγκρίσιμες αποδόσεις των CNNs, συνεχίζουν τα CNNs να παίζουν καθοριστικό ρόλο και στη βιομηχανία αλλά και στην ακαδημαϊκή κοινότητα.

Transformers και ViT

Οι Transformers [17] και οι Vision Transformers [18] κάνουν χρήση ενός τύπου μηχανισμού “**attention**”. Ο μηχανισμός αυτός επιτρέπει σε ένα μοντέλο να εστιάζει σε συγκεκριμένα μέρη μιας ακολουθίας εισόδου όταν δημιουργεί μια έξοδο. Εισήχθη για πρώτη φορά στον τομέα της μηχανικής μετάφρασης [19], αλλά έκτοτε έχει εφαρμοστεί σε μια μεγάλη ποικιλία εργασιών επεξεργασίας φυσικής γλώσσας (NLP), συμπεριλαμβανομένης της σύνοψης κειμένου, της απάντησης ερωτήσεων και της αναγνώρισης ομιλίας αλλά για την παραγωγή λεζαντών σε εικόνες. Ο μηχανισμός αυτός λειτουργεί αναθέτοντας βάρη σε διαφορετικά μέρη της ακολουθίας εισόδου ανάλογα με το πόσο σημαντικά είναι για την παραγωγή της εξόδου, δηλαδή όσο μεγαλύτερο είναι το βάρος, τόσο περισσότερη προσοχή θα δώσει το μοντέλο σε αυτό το τμήμα της εισόδου κατά την παραγωγή της εξόδου.

Ένα είδος τέτοιου μηχανισμού που χρησιμοποιείται στους transformers είναι ο “**self-Attention**” ή “**Scaled Dot-Product**”. Αυτός ο μηχανισμός μπορεί να θεωρηθεί ως μία μέθοδο για τον υπολογισμό της συνάφειας (attention) μιας λέξης ή φράσης με μια άλλη μέσα σε μία ακολουθία. Αυτή η μέθοδος λαμβάνει τρεις εισόδους:

- **Query ($Q(x_i)$):** Αντιπροσωπεύει μία ερώτηση που μπορεί να κάνουμε ώστε να εξαγούμε κάποιο συμπέρασμα μέσω από την ακολουθία εισόδου.
- **Key ($K(x_i)$):** Μπορεί να θεωρηθεί ως μία αναπαράσταση της λέξης, η οποία έχει σχέση με το ερώτημα που κάναμε μέσω του Query.
- **Value ($V(x_i)$):** Παρέχει τις ουσιώδεις πληροφορίες που το μοντέλο προσπαθεί να εξαγάγει από την ακολουθία εισόδου μέσω της ερώτησης

Αυτές οι τιμές ($Q(x_i)$, $K(x_i)$, $V(x_i)$) υπολογίζονται με τη βοήθεια τριών πινάκων με βάρη και πρακτικά μετατρέπουν το διάνυσμα εισόδου (embedding) σε μία καλύτερη αναπαράσταση που βοηθάει στην άντληση πληροφορίας από την ίδια ακολουθία-πρόταση ενώ παράλληλα περιγράφει με καλύτερο τρόπο τη σχέση που έχουν οι λέξεις μεταξύ τους.

- $Q(x_i) = W^Q \cdot x_i$

- $K(x_i) = W^K \cdot x_i$
- $V(x_i) = W^V \cdot x_i$

όπου:

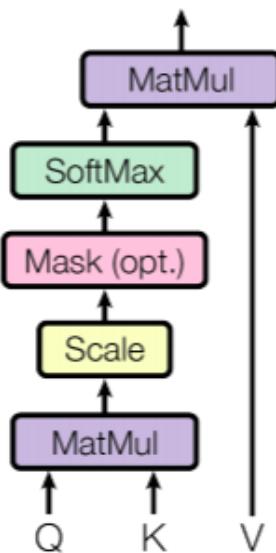
- W^Q, W^K, W^V : είναι εκπαιδεύσιμοι παράμετροι
 x_i : το embedding εισόδου για την είσοδο i. (Embedding είναι μία αναπαράσταση που κωδικοποιεί τη σημασία μίας λέξης)

Αυτή η διαδικασία για όλες τις τιμές εισόδου μπορεί να συνοψιστεί υπό την εξίσωση 2.6 που γραφικά περιγράφεται μέσα από το σχήμα 2.11

$$\text{Attention}(Q, K, V) = \text{softmax} \left(\frac{QK^T}{\sqrt{d_k}} \right) \cdot V \quad (2.6)$$

όπου:

- d_k : είναι η διάσταση που έχει το διάνυσμα $K(x_i)$
 $\sqrt{d_k}$: είναι ένας παράγοντας κλιμάκωσης που βοηθά στο να αποτραπεί να γίνουν πολύ μεγάλο το εσωτερικό γινόμενο QK^T και να προκαλέσει αριθμητική αστάθεια.



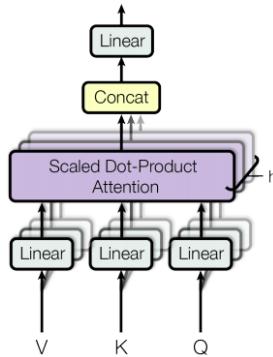
Σχήμα 2.11: Scaled Dot-Product Attention

Κάνοντας αυτή τη διαδικασία (self-Attention) για μόνο ένα συνδυασμό βαρών (W^Q, W^K, W^V) έχουμε μόνο μία αναπαράσταση της ακολουθίας εισόδου. Τις περισσότερες φορές αυτό δεν είναι αρκετό για να μπορούμε να εξάγουμε σωστά συμπεράσματα και για αυτό αυτή η διαδικασία επαναλαμβάνεται με διαφορετικούς πίνακες βαρών. Με αυτόν τον τρόπο έχουμε μια διαφορετική αναπαράσταση, η οποία αντιπροσωπεύει διαφορετικές συσχετίσεις των εισόδων αφού κάθε σετ των πινάκων Q, K και V (ή head) μπορεί να βρει διαφορετικές συσχετίσεις. Αυτή τη διαδικασία με τα διαφορετικά σετ βαρών ονομάζεται **“Multi-Head Attention”** και μπορούμε να το φανταστούμε σαν να κάνουμε μία διαφορετική ερώτηση κάθε φορά, η οποία να εστιάζει σε διαφορετικό χαρακτηριστικό της ακολουθίας εισόδου. Πρακτικά αυτό υλοποιείται υπολογίζοντας παράλληλα διαφορετικά “Scaled Dot-Product” αφού το κάθε head είναι ανεξάρτητο από το άλλο. Αυτή η διαδικασία περιγράφεται και στο σχήμα 2.12 αλλά και στην εξίσωση 2.7

$$\text{MultiHead}(Q, K, V) = \text{concat}(\text{head}_1, \text{head}_2, \dots, \text{head}_n) \cdot W_0 \quad (2.7)$$

όπου:

$$\text{head}_i : \text{Attention}(W_i^Q Q, W_i^K K, W_i^V V)$$



Σχήμα 2.12: Multi-Head Attention

Προτού αναλύσουμε το δομή ενός transformer πρέπει πρώτα να περιγράψουμε τα “**Positional Embedding (PE)**” 2.8. Τα PE είναι διανύσματα που περιγράφουν τη θέση για το κάθε στοιχείου εισόδου. Έχουν μέγεθος όσο είναι τα embeddings εισόδου και οι τιμές τις κάθε διάστασης υπολογίζονται μέσα από τις εξισώσεις 2.8, όπου η μία αφορά τις ζυγές και η άλλη τις μονές διαστάσεις.

$$\begin{aligned} PE_{pos,2i} &= \sin\left(\frac{pos}{10000^{\frac{2i}{d}}}\right) \\ PE_{pos,2i+1} &= \cos\left(\frac{pos}{10000^{\frac{2i}{d}}}\right) \end{aligned} \quad (2.8)$$

όπου:

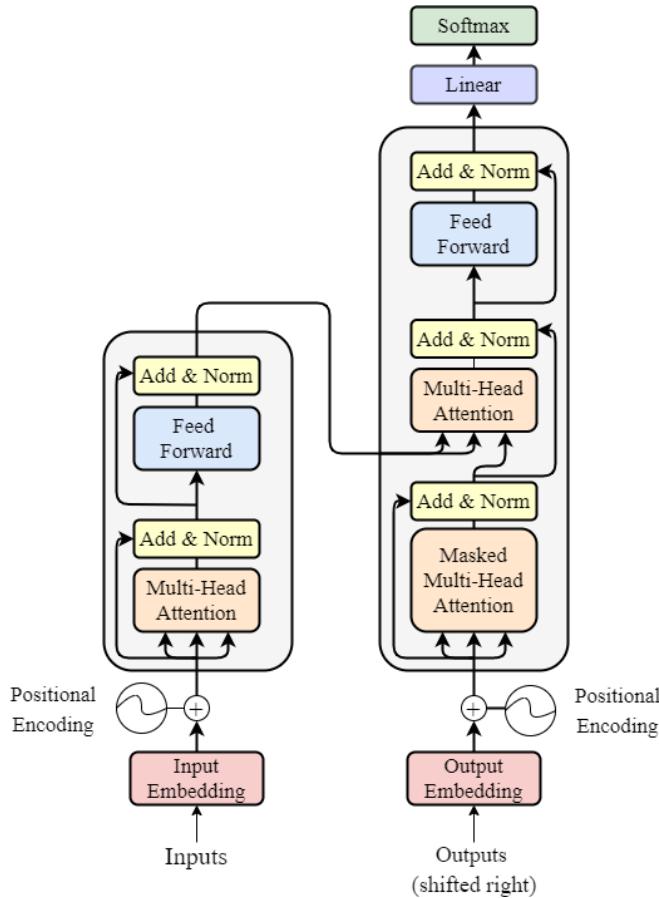
pos : Αύξων αριθμός από το 0,...,T-1 που περιγράφει τη θέση της λέξης μέσα στην ακολουθία

i : Αντιπροσωπεύει τον δείκτη διάστασης μέσα στο PE και παίρνει τιμές $0,...,d/2$

d : Το μέγεθος των διανύσματος που αφορά τα embeddings εισόδου

Έχοντας καλύψει όλα τα βασικά στοιχεία ενός **Transformer** μπορούμε να περιγράψουμε την λειτουργία και του ίδιου. Οι Transformers [17] εμφανίστηκαν για πρώτη φορά το 2017 από ερευνητές της Google σε μια εφαρμογή machine translation. Αποτελούνται από δύο κύρια στοιχεία έναν κωδικοποιητή (encoder) και έναν αποκωδικοποιητή (decoder). Ο κωδικοποιητής παίρνει μια ακολουθία εισόδου και τη μετατρέπει σε μια αναπαράσταση που αποτυπώνει το νόημα του κειμένου. Στη συνέχεια, ο αποκωδικοποιητής παίρνει αυτήν την αναπαράσταση και δημιουργεί μια ακολουθία εξόδου.

Το βασικότερο υποστοιχείο και του κωδικοποιητή αλλά και του αποκωδικοποιητή είναι το “Multi-Head Attention”, που πρακτικά είναι υπεύθυνο για την αναπαράσταση της πληροφορίας. Το Normalization στρώμα εφαρμόζεται μετά από κάθε επίπεδο για να βοηθήσει στη ευστάθεια κατά τη διαδικασία εκπαίδευσης και στην αποφυγή του overfitting, δηλαδή έχει παρόμοια λειτουργία με το στρώμα BatchNorm που συναντάμε σε άλλα είδη δικτύων. Από την άλλη το Feed Forward Network (FFN) είναι ένα Fully Connected Network που προσθέτει μη γραμμικότητα στο μοντέλο και είναι υπεύθυνο για την εκμάθηση πιο περίπλοκων μοτίβων και σχέσεων μεταξύ λέξεων που δεν μπορούν να αποτυπωθούν μόνο από τον μηχανισμό “self-attention”. Τέλος,



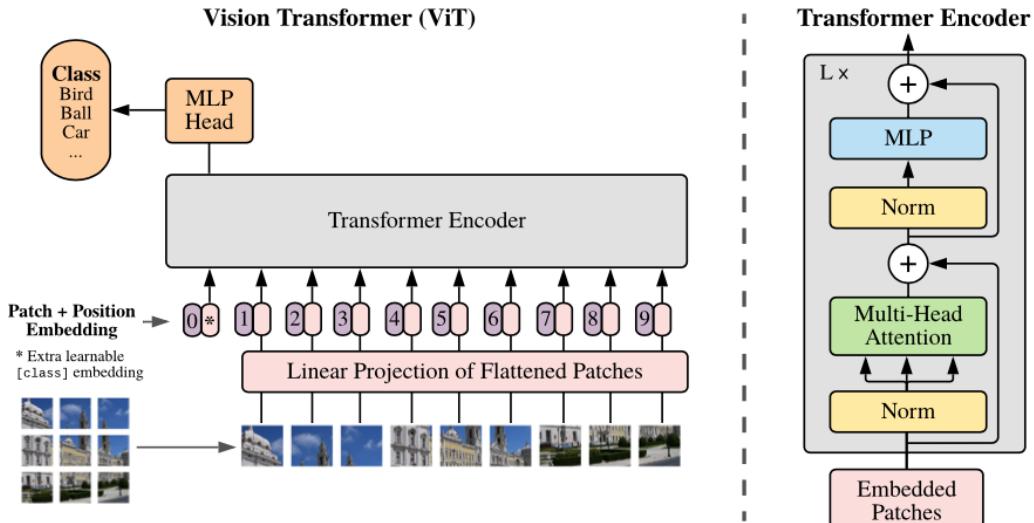
Σχήμα 2.13: Αρχιτεκτονική ενός Transformer (από αριστερά είναι κωδικοποιητής ενώ στα δεξιά ο αποκωδικοποιητής)

οι συνδέσεις Residual βελτιώνουν τη ροή πληροφορίας μέσα στο δίκτυο αλλά και μεταφέρουν την πληροφορία της θέσης κάθε λέξης μέσα σε ολόκληρη τη δομή.

Παρόλο που οι transformers αρχικά είχαν χρησιμοποιηθεί σε εφαρμογές που είχαν σχέση με την επεξεργασία φυσικής γλώσσας (NLP), μέσω της δημοσίευσης [18] προτάθηκε και ένας νέος τρόπος χρήσης τους που έχει να κάνει με εφαρμογές της υπολογιστικής όρασης. Η αρχιτεκτονική που προτάθηκε ονομάστηκε **Vision Transformer - ViT** και πρακτικά είναι ο κωδικοποιητής του transformer της δημοσίευσης [17] αλλά με διαφορετικό μηχανισμό στην είσοδό του.

Πιο αναλυτικά καθώς ο transformer δέχεται σαν είσοδο μία ακολουθία, πρέπει με κάποιον τρόπο η εικόνα, $H \times W \times C$, να μετατραπεί σε αυτόν τον τύπο εισόδου. Αυτό γίνεται χωρίζοντας την εικόνα σε μικρότερα τετράγωνα (patches), $P \times P$, που στη συνέχεια τα περνάμε μέσα από ένα γραμμικό μετασχηματιστή που μετατρέπει την εικόνα σε δομή με ίδια λειτουργία που έχουν τα embeddings εισόδου στον κλασικό transformer. Ακόμα τα Position Embeddings πλέον δεν υπολογίζονται από τις συναρτήσεις ημιτόνου και συνημιτόνου αλλά αποτελούν εκπαιδεύσιμες παράμετροι του δικτύου. Τέλος, ακολουθεί ένα MLP-Head που είναι υπεύθυνο για την μετατροπή των παραγόμενων αναπαραστάσεων που εξάγονται από τον κωδικοποιητή σε πιθανότητες κλάσης, επιτρέποντας στο μοντέλο να εκτελεί εργασίες ταξινόμησης εικόνων.

Εν κατακλείδι, αν και οι αρχιτεκτονικές που βασίζονται στο μηχανισμό του “self-attention” έχουν συγκρίσιμες και σε κάποιες περιπτώσεις καλύτερες αποδόσεις από αρχιτεκτονικές που δεν τον χρησιμοποιούν, υπάρχουν κάποια σημαντικά προβλήματα. Το πιο βασικό είναι ότι για να φτάσει ένα τέτοιο δίκτυο να έχει τόσο καλά αποτελέσματα χρειάζεται μια σημαντική ποσότητα δεδομένων αλλά και υπολογιστικής ισχύς, καθώς η πράξη του “Scaled Dot-Product



Σχήμα 2.14: Αρχιτεκτονική ενός Vision Transformer

Attention” είναι αρκετά χρονοβόρα αν σκεφτούμε το πόσες φορές χρειάζεται να υπολογιστεί αλλά και αρκετά απαιτητική στη χωρητικότητα της μνήμης.

2.2.4 Βασικές Ορολογίες στην Εκπαίδευση Νευρωνικών Δικτύων

Συνάρτηση Κόστους

Ένα σημαντικό μέρος της εκπαίδευσης νευρωνικών δικτύων είναι η συνάρτηση κόστους (cost function). Χρησιμεύει ως μέτρο σύγκρισης για την αξιολόγηση της απόδοσης του μοντέλου αλλά και την καθοδήγηση της εκπαιδευτικής διαδικασίας. Η συνάρτηση κόστους μετρά την απόκλιση μεταξύ των προβλεπόμενων εξόδων του δικτύου και των πραγματικών τιμών. Με την διαδικασία της ελαχιστοποίησης της συνάρτησης κόστους, το νευρωνικό δίκτυο εκπαιδεύεται να κάνει ακριβείς προβλέψεις σε δεδομένα που δεν έχει δει μεταβάλλοντας τους πίνακες με τα βάρη και πρακτικά αυτό αποτελεί την διαδικασία της εκπαίδευσης του δικτύου.

Στα νευρωνικά δίκτυα χρησιμοποιούνται διάφορες συναρτήσεις κόστους ή και συνδυασμός αυτών, καθεμία προσαρμοσμένη σε συγκεκριμένους τύπους προβλημάτων και δεδομένων. Μερικές από τις πιο συχνά χρησιμοποιούμενες συναρτήσεις κόστους περιλαμβάνουν:

- **Μέσο τετράγωνο σφάλμα (MSE):** Το MSE είναι μια ευρέως χρησιμοποιούμενη συνάρτηση κόστους που χρησιμοποιείτε σχεδόν σε όλα τα δίκτυα (LSTMs, MLPs, CNNs) στα οποία ο στόχος είναι η πρόβλεψη συνεχών αριθμητικών τιμών. Μετρά τη μέση τετραγωνική διαφορά μεταξύ των προβλεπόμενων και των πραγματικών τιμών.

$$MSE = \frac{1}{n} \cdot \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.9)$$

- **Binary Cross-Entropy (BCE):** Το BCE είναι μία κατάλληλη συνάρτηση κόστους για εργασίες δυαδικής ταξινόμησης, όπου το δίκτυο προβλέπει ένα δυαδικό αποτέλεσμα (π.χ. 0 ή 1). Μετρά τη μέση διασταυρούμενη εντροπία (cross-entropy) μεταξύ των προβλεπόμενων αποτελεσμάτων και των πραγματικών ετικετών.

$$BCE = -\frac{1}{n} \cdot \sum_{i=1}^n (y_i \cdot \log(\hat{y}_i) + (1 - y_i) \cdot \log(1 - \hat{y}_i)) \quad (2.10)$$

- **Categorical Cross-Entropy (CCE):** Το CCE είναι μια επέκταση του BCE για εργασίες ταξινόμησης πολλαπλών κλάσεων άλλα και για κατάτμηση εικόνας που έχει αρκετές κατηγορίες. Μετρά τη μέση διασταυρούμενη εντροπία μεταξύ της προβλεπόμενης κατανομής πιθανοτήτων και των πραγματικών ετικετών.

$$CCE = - \sum_{i=1}^n (y_i \cdot \log(\hat{y}_i)) \quad (2.11)$$

- **Dice Loss:** είναι μια συνάρτηση απώλειας που χρησιμοποιείται συνήθως σε εργασίες κατάτμησης εικόνας. Μετρά την ομοιότητα μεταξύ δύο συνόλων pixel μέσω της επικάλυψης που μπορεί να έχουν και συνήθως αυτά τα σύνολα είναι η Ground Truth μάσκα και η προβλεπόμενη μάσκα κατάτμησης. Σε αντίθεση με άλλες συναρτήσεις όπως η διασταυρούμενη εντροπία ή το μέσο τετράγωνο σφάλμα, το Dice Loss επικεντρώνεται στην επικάλυψη μεταξύ των δύο συνόλων καθιστώντας το ιδιαίτερα αποτελεσματικό για εργασίες τμηματοποίησης όπου τα όρια μεταξύ περιοχών είναι εξαιρετικά σημαντικά.

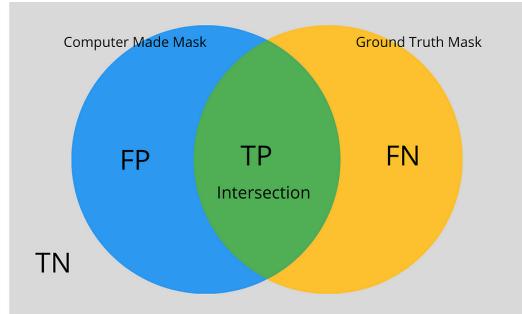
$$DiceLoss = 1 - \frac{2 \cdot TP}{FP + FN + 2 \cdot TP} \quad (2.12)$$

όπου:

TP : True Positives

FP : False Positives

FN : False Negatives



Σχήμα 2.15: Οπτικοποίηση του Dice Coefficient

- **Focal Loss [20]:** προτάθηκε για να λύσει το πρόβλημα της ανισορροπίας μεταξύ των κλάσεων κατά τη διάρκεια της εκπαίδευσης, δηλαδή να έχουμε περισσότερα παραδείγματα για μια κατηγορία από μια άλλη. Διαισθητικά, ο παράγοντας κλιμάκωσης $(1 - p_t)^\gamma$ μπορεί αυτόματα να μειώσει τη συμβολή των εύκολων παραδειγμάτων κατά τη διάρκεια της εκπαίδευσης και να εστιάσει γρήγορα το μοντέλο σε δύσκολα παραδείγματα.

$$FocalLoss = (1 - p_t)^\gamma \log(p_t) \quad (2.13)$$

όπου:

p_t : η πιθανότητα να έχει προβλέψει σωστά το μοντέλο το εκάστοτε παράδειγμα

γ : focusing parameter, όσο μεγαλύτερο τόσο πιο μεγάλο βάρος δίνουμε στα δύσκολα παραδείγματα

Gradient Descent και Backpropagation

Ας δούμε τώρα τις έννοιες του Gradient Descent και του Backpropagation. Αυτοί οι αλγόριθμοι είναι θεμελιώδεις στην εκπαίδευση νευρωνικών δικτύων και χρησιμοποιούνται σε όλα τα είδη δικτύων. Μαζί αποτελούν τη ραχοκοκαλιά της εκπαίδευσης στα νευρωνικά δίκτυα.

Πιο αναλυτικά ο **Gradient Descent** [21] είναι ένας ευρέως χρησιμοποιούμενος αλγόριθμος βελτιστοποίησης στη μηχανική μάθηση, ιδιαίτερα στην εκπαίδευση νευρωνικών δικτύων. Είναι ένας επαναληπτικός αλγόριθμος που στοχεύει στην ελαχιστοποίηση μιας συνάρτησης απώλειας (cost function) προσαρμόζοντας τα βάρη του νευρωνικού δικτύου. Αυτόν τον αλγόριθμο μπορούμε να τον παρομοιάσουμε σαν να ψάχνουμε τον πιο γρήγορο μονοπάτι για να κατέβουμε από την πλαγιά ενός βουνού. Σύμφωνα με αυτόν θα επιλέξουμε το μονοπάτι που έχει την πιο απότομη κλίση, αλλά αυτή η μέθοδος δεν είναι απαραίτητο να μας οδηγήσει στο ολικό ελάχιστο καθώς επηρεάζεται αρκετά από τις αρχικές συνθήκες. Η εξίσωση με την οποία υπολογίζουμε την κάθε παράμετρο για το επόμενο βήμα βρίσκεται μέσω της εξίσωσης 2.14

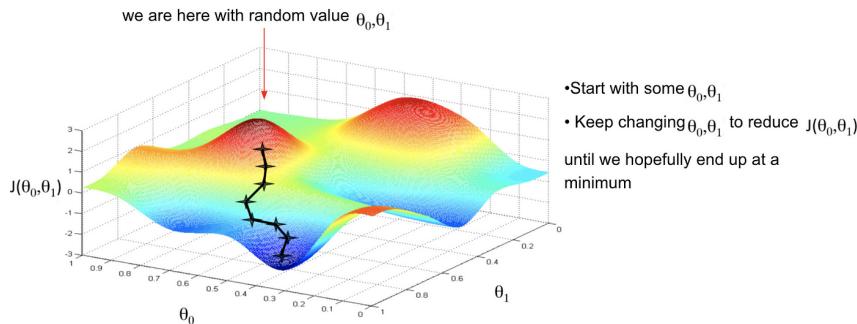
$$\theta_{t+1} = \theta_t - \alpha \nabla J(\theta_t) \quad (2.14)$$

όπου:

θ_t : Διάνυσμα των παραμέτρων στην επανάληψη t

α : ρυθμός εκμάθησης (learning rate)

$\nabla J(\theta_t)$: Η κλίση της συνάρτησης κόστους υπολογισμένη τη χρονική στιγμή t

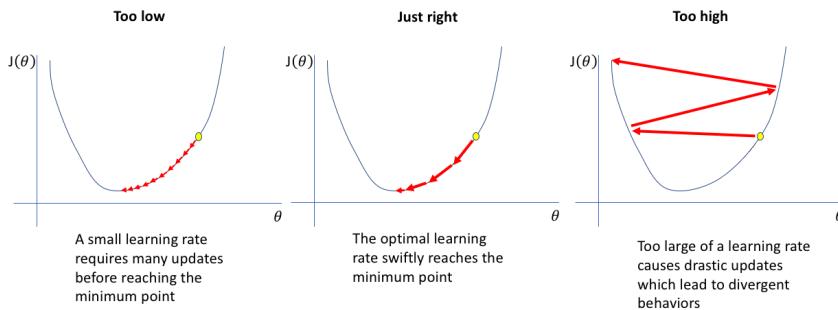


Σχήμα 2.16: Οπτικοποίηση του αλγόριθμου gradient-descent

Όπως παρατηρούμε και από το σχήμα 2.17 ο ρυθμός εκμάθησης έχει σημαντική επίδραση τόσο για το αποτέλεσμα (μεγάλος ρυθμός) όσο και για το χρόνο υπολογισμού (μικρός ρυθμός). Ως εκ τούτου προτάθηκαν και άλλες υλοποιήσεις που βελτιώνουν τα πρόβληματα της αρχικής υλοποιήσεις. Κάποιες από αυτές είναι:

- **Stochastic gradient descent (SGD)** ενημερώνει τα βάρη του δικτύου μετά από κάθε παράδειγμα εκπαίδευσης. Αυτό μπορεί να κάνει το SGD γρηγορότερο από άλλες υλοποιήσεις, αλλά μπορεί επίσης να το κάνει να έχει περισσότερη διακύμανση.
- **Mini-batch gradient descent** ενημερώνει τα βάρη του δικτύου μετά από ένα μικρό batch παραδειγμάτων εκπαίδευσης
- **Adaptive learning rate**: είναι μια τεχνική που μπορεί να βοηθήσει στη βελτίωση της απόδοσης του gradient descent προσαρμόζοντας τον ρυθμό μάθησης καθώς προχωρά η εκπαίδευση.

Από την άλλη μεριά έχουμε το **Backpropagation** ή **Error Propagation** που προτάθηκε για πρώτη φορά το 1961 [22] και αποτελεί έναν αποτελεσματικό τρόπο υπολογισμού της κλίσης



Σχήμα 2.17: Αλγόριθμος gradient-descent για διάφορους ρυθμούς εκμάθησης

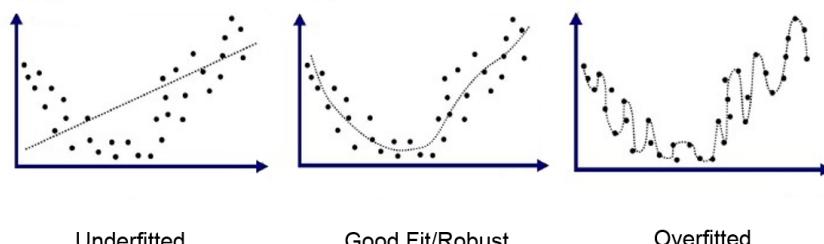
της συνάρτησης απώλειας που χρησιμοποιεί η gradient descent. Στην αρχή χρησιμοποιεί το Forward pass για να υπολογιστεί η έξοδος και το σφάλμα που έχει την πραγματική τιμή και στη συνέχεια μέσω του Backward pass το σφάλμα διαδίδεται προς τα πίσω σε ολόκληρο το δίκτυο. Για να γίνει αυτό και να υπολογίσει η κλίση σε κάθε στρώμα του δικτύου χρησιμοποιείται ο κανόνας της αλυσίδας, που πρακτικά υπολογίζει τις παραγώγους συναρτήσεων μέσω άλλων γνωστών παραγώγων.

Overfitting και Underfitting

Στη μηχανική μάθηση, η υπερπροσαρμογή και η υποπροσαρμογή (overfitting και underfitting) είναι δύο προβλήματα με κοινούς λόγους εμφάνισης που συμβαίνουν κατά τη διάρκεια της εκπαίδευσης και μπορούν να επηρεάσουν σημαντικά την απόδοση ενός μοντέλου.

Το **overfitting** συμβαίνει όταν ένα μοντέλο μαθαίνει πολύ καλά τα δεδομένα εκπαίδευσης, συμπεριλαμβανομένου του θορύβου και των άσχετων μοτίβων, σε βαθμό που δεν μπορεί να γενικεύσει καλά σε νέα δεδομένα. Αυτό έχει ως αποτέλεσμα ένα μοντέλο που έχει καλή απόδοση στα δεδομένα εκπαίδευσης (training set) αλλά κακή στα δεδομένα δοκιμών (test set). Αυτό μπορεί να συμβαίνει διότι δίκτυο μπορεί να έχει πάρα πολλές παραμέτρους ή όταν τα δεδομένα εκπαίδευσης περιέχουν θόρυβο ή άσχετα μοτίβα τα οποία θα οδηγήσουν σε κακή γενίκευση.

Από την άλλη πλευρά το **underfitting** είναι ακριβώς το αντίθετο. Συμβαίνει όταν ένα μοντέλο είναι πολύ απλό για να καταγράψει τα μοτίβα στα δεδομένα εκπαίδευσης. Αυτό οδηγεί σε ένα μοντέλο που έχει κακή απόδοση τόσο στα δεδομένα εκπαίδευσης όσο και στα δεδομένα δοκιμών.



Σχήμα 2.18: Οπτικοποίηση Overfitting και Underfitting

Συνάρτηση Ενεργοποίησης

Μια συνάρτηση ενεργοποίησης είναι μια μαθηματική συνάρτηση που εισάγει μη γραμμικότητα στην έξοδο ενός νευρώνα σε ένα νευρωνικό δίκτυο. Αυτή η μη γραμμικότητα είναι απαραίτητη

για τα νευρωνικά δίκτυα να μαθαίνουν και να αναπαριστούν σύνθετα μοτίβα σε δεδομένα. Χωρίς συναρτήσεις ενεργοποίησης, τα νευρωνικά δίκτυα θα περιορίζονταν σε γραμμικές σχέσεις, οι οποίες δεν είναι αρκετά ισχυρές για να μοντελοποιήσουν πολλά φαινόμενα του πραγματικού κόσμου.

Υπάρχουν πολλοί διαφορετικές συναρτήσεις ενεργοποίησης, αλλά μερικές από τις πιο συνηθισμένες είναι:

- **Sigmoid:** συμπιέζει την είσοδό της σε μια τιμή μεταξύ 0 και 1. Αυτό είναι χρήσιμο για εργασίες όπως η δυαδική ταξινόμηση, όπου η έξοδος του δικτύου είναι είτε 0 είτε 1. Όμως μπορεί να υποφέρει από το πρόβλημα των vanishing gradients, το οποίο εμφανίζεται όταν είναι πολύ μικροί παράγωγοι με αποτέλεσμα να μην γίνεται σωστά το error propagation, καθιστώντας δύσκολη την εκπαίδευση των βαθιών νευρωνικών δικτύων

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (2.15)$$

- **Υπερβολική εφαπτομένη (tanh):** είναι παρόμοια με τη συνάρτηση σιγμοειδούς, αλλά συμπιέζει την είσοδό της σε μια τιμή μεταξύ -1 και 1. Καθώς μοιάζει αρκετά την σιγμοειδή υποφέρει και αυτή από το πρόβλημα των vanishing gradients, αλλά το γεγονός ότι η έξοδος είναι κεντραρισμένη γύρω στο 0, την κάνει καλύτερη επιλογή από την σιγμοειδή συνάρτηση.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.16)$$

- **Rectified Linear Unit (ReLU):** βγάζει την είσοδό της στην έξοδο εάν είναι θετική η είσοδος και 0 εάν η είσοδος είναι αρνητική. Η ReLU είναι μια πολύ δημοφιλής συνάρτηση ενεργοποίησης επειδή είναι γρήγορη στον υπολογισμό και είναι λιγότερο επιρρεπής στο vanishing gradients από τη σιγμοειδή και τη tanh αλλά και σε προβλήματα κορεσμού. Το γεγονός ότι η έξοδος δεν έχει κάποιο όριο καθιστά δύσκολο τον έλεγχο της συμπεριφοράς του δικτύου αλλά και όταν παράγει 0 στην έξοδο δεν ανανεώνονται οι παράμετροι των προηγούμενων στρωμάτων κατά την εκπαίδευση του δικτύου.

$$\text{ReLU}(x) = \max(0, x) \quad (2.17)$$

- **Parametric Leaky ReLU:** είναι μια παραλλαγή της συνάρτησης ReLU που εξάγει μία μικρή μη μηδενική τιμή εάν η είσοδος είναι αρνητική. Αυτό μπορεί να βοηθήσει στην αποτροπή της μη ενεργοποίησης ενός μέρους του δικτύου εάν συνεχώς ένας νευρώνας παράγει αρνητική τιμή, που εμφανίζεται όταν χρησιμοποιείται η ReLU. Από την άλλη εισάγει μια νέα παράμετρο που χρειάζεται να ρυθμιστεί κατάλληλα.

$$\text{PLReLU}(x) = \begin{cases} \alpha x, & \text{εάν } x < 0, \text{ συνήθως } \alpha = 0.01 \\ x, & \text{εάν } x \geq 0 \end{cases} \quad (2.18)$$

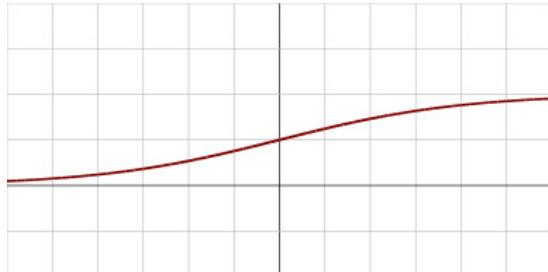
- **Softmax:** είναι μια συνάρτηση ενεργοποίησης σε αντίθεση με τις προηγούμενες, που χρειάζεται όλες τις εξόδους ενός στρώματος για να υπολογιστεί την έξοδος. Χρησιμοποιείται για ταξινόμηση πολλαπλών κλάσεων και εξάγει μια κατανομή πιθανότητας στις πιθανές κλάσεις, όπου η πιθανότητα κάθε κλάσης είναι ανάλογη της εξόδου της. Όμως είναι υπολογιστικά ακριβή σε σύγκριση με άλλες συναρτήσεις ειδικά σε περιπτώσεις όπου υπάρχουν πάρα πολλές έξοδοι όπως σε εφαρμογές NLP.

$$\sigma(x_i) = \frac{e^{x_i}}{\sum_{j=1}^K e^{x_j}} \quad (2.19)$$

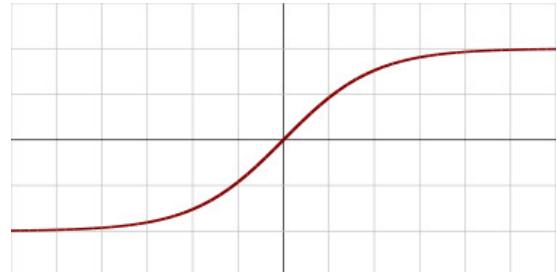
όπου:

x_i : Οι έξοδοι των νευρώνων

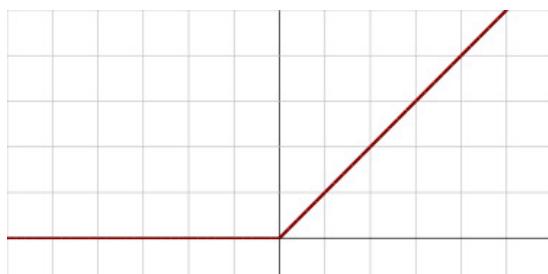
Η επιλογή της συνάρτησης ενεργοποίησης μπορεί να έχει σημαντικό αντίκτυπο στην απόδοση ενός νευρωνικού δικτύου και εξαρτάται από τη συγκεκριμένη εφαρμογή και την αρχιτεκτονική του νευρωνικού δικτύου. Γενικά, θεωρείτε μια καλή πρακτική να δοκιμάζονται διάφορες συναρτήσεις και να αξιολογούνται τα αποτελέσματα από κάθε επιλογή, καθώς δεν υπάρχουν αυστηροί κανόνες για την καταλληλότερη συνάρτηση.



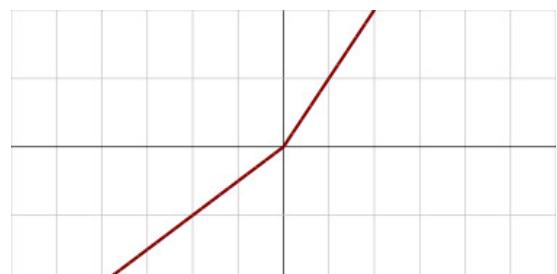
(i) $\text{Sigmoid}(x)$



(ii) $\tanh(x)$



(iii) $\text{ReLU}(x)$



(iv) $\text{PLReLU}(x)$

Σχήμα 2.19: Οι πιο γνωστές συναρτήσεις ενεργοποίησης

2.3 Γεωμετρικοί Μετασχηματισμοί

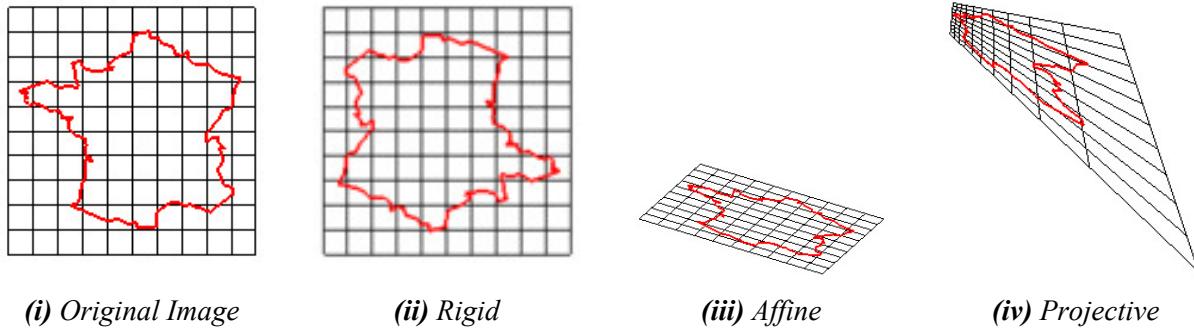
Στα μαθηματικά, ένας μετασχηματισμός είναι μια συνάρτηση f , συνήθως με κάποια γεωμετρική ερμηνεία, που αντιστοιχίζει ένα σύνολο X στον εαυτό του, δηλαδή $f : X \rightarrow X$. Κάποιοι μετασχηματισμοί είναι οι γραμμικοί μετασχηματισμοί διανυσματικών χώρων και οι γεωμετρικοί μετασχηματισμοί, οι οποίοι περιλαμβάνουν προοπτικούς μετασχηματισμούς, affine μετασχηματισμούς, όπως περιστροφές, αντανακλάσεις και μετακινήσεις [2].

2.3.1 Τύποι Γεωμετρικών Μετασχηματισμών

- **Άκαμπτος Μετασχηματισμός (Rigid Transformation)** είναι ένας γεωμετρικός μετασχηματισμός ενός Ευκλείδειου χώρου που διατηρεί την Ευκλείδεια απόσταση μεταξύ κάθε ζεύγους σημείων, τέτοιοι μετασχηματισμοί είναι η μετατόπιση και η περιστροφή.
- **Μετασχηματισμός Affine (Affine Transformation)** είναι ένας γεωμετρικός μετασχηματισμός υπερσύνολο του άκαμπτου μετασχηματισμού ο οποίος διατηρεί τις γραμμές

και τον παραλληλισμό, αλλά όχι απαραίτητα τις Ευκλείδειες αποστάσεις και γωνίες. Παραδείγματα τέτοιων μετασχηματισμών περιλαμβάνουν την μετατόπιση, κλιμάκωση, ανάκλαση, περιστροφή, στρέβλωση αλλά συνδυασμούς αυτών.

- **Προοπτικός Μετασχηματισμός (Projective Transformation)** αποτελεί μια συσχέτιση μεταξύ ευθειών από έναν χώρο σε έναν άλλο, χωρίς απαραίτητα να διατηρείται ο παραλληλισμός. Είναι χρήσιμος όταν θέλουμε να κάνουμε ανακατασκευή μιας εικόνας ενός αντικείμενου, το οποίο το έχουμε φωτογραφίσει από δύο διαφορετικές οπτικές γωνίες.



Σχήμα 2.20: Τύποι Μετασχηματισμών [2]

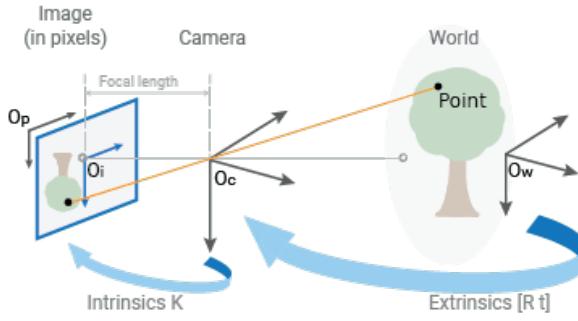
2.4 Μοντελοποίηση Κάμερας

Το πιο γνωστό μοντέλο που προσημειώνει μία κάμερα είναι το “Pinhole Camera Model” [23]. Αυτό μοντέλο περιγράφει τη μαθηματική σχέση μεταξύ των συντεταγμένων ενός σημείου στον τρισδιάστατο χώρο και της προβολής του στο επίπεδο εικόνας μιας ιδανικής κάμερας pinhole, όπου το διάφραγμα της κάμερας περιγράφεται ως ένα σημείο. Το μοντέλο δεν περιλαμβάνει, για παράδειγμα, γεωμετρικές παραμορφώσεις ή θόλωση μη εστιασμένων αντικειμένων που προκαλούνται από φακούς και πεπερασμένα διαφράγματα. Τέλος, η διαδικασία της εύρεσης των απαραίτητων παραμέτρων του μαθηματικού μοντέλου λέγεται βαθμονόμηση και είναι αρκετά σημαντική σε εφαρμογές υπολογιστής όρασης. Ακολουθεί το μαθηματικό μοντέλο 2.20.

$$w \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = K (R|t) \begin{pmatrix} x \\ y \\ z \\ 1 \end{pmatrix} \quad (2.20)$$

2.4.1 Εξωτερικοί Παράμετροι Κάμερας

Οι εξωτερικοί παράμετροι (Extrinsic Parameters) μιας κάμερας είναι χρήσιμοι για να περιγράψουν τη θέση και την κατεύθυνση της κάμερας στο χώρο. Χρησιμοποιούνται για να μετατρέψουν τις 3Δ συντεταγμένες των σημείων στον πραγματικό κόσμο σε συντεταγμένες όπου στο κέντρο του κόσμου βρίσκεται η ίδια η κάμερα. Για να γίνει αυτό χρειάζεται να γνωρίζουμε τους πίνακες **R** (2.21) και **T** (2.22) της εξίσωσης (2.20) οι οποίοι είναι υπεύθυνοι για την περιστροφή



Σχήμα 2.21: Προβολή ενός 3Δ σημείου στο επίπεδο εικόνας σύμφωνα με το Pinhole Camera Model [3]

και την μετατόπιση αντίστοιχα.

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix} \quad (2.21)$$

$$t = \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \quad (2.22)$$

2.4.2 Εσωτερικοί Παράμετροι Κάμερας

Οι εσωτερικοί παράμετροι (Intrinsic Parameters) αποτελούνται ως επί το πλείστον από το οπτικό κέντρο c_x, c_y και την εστιακή απόσταση της κάμερας f_x, f_y και είναι υπεύθυνοι για την σωστή προβολή του 3Δ σημείου πάνω στο επίπεδο προβολής (Image Plane). Ακολουθεί ο πίνακας των εσωτερικών παραμέτρων 2.23.

$$K = \begin{pmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{pmatrix} \quad (2.23)$$

όπου:

c_x, c_y : Το οπτικό κέντρο (the principal point), σε pixels

f_x, f_y : Εστιακή απόσταση σε pixel

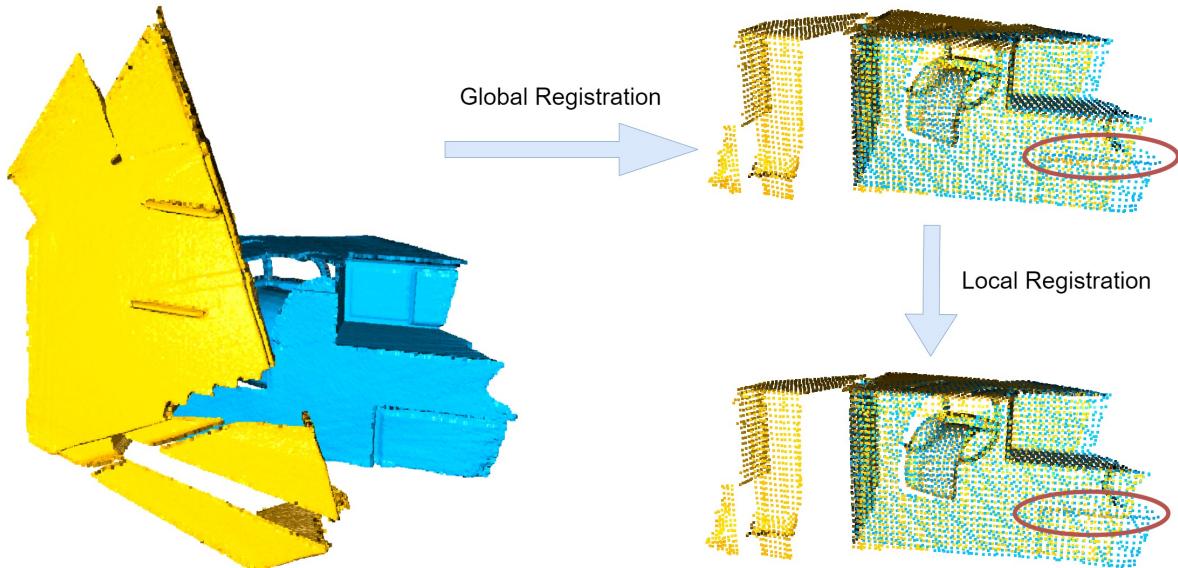
s : Συντελεστής κλίσης (skew), μηδέν όταν οι άξονες του Image Plane είναι κάθετοι

2.5 Ευθυγράμμιση Νέφους Σημείων

Η ευθυγράμμιση σε νέφους σημείων είναι η διαδικασία κατά την οποία δύο ή περισσότερων σύνολα τρισδιάστατων σημείων μετασχηματίζονται σε ένα κοινό σύστημα συντεταγμένων. Αυτό είναι ένα θεμελιώδες πρόβλημα στην τρισδιάστατη υπολογιστική όραση και χρησιμοποιείται σε μια μεγάλη ποικιλία εφαρμογών, όπως τρισδιάστατη ανακατασκευή, παρακολούθηση κίνησης αλλά και ανίχνευση αντικειμένων [24], [25]. Συχνά η ευθυγράμμιση νέφους σημείων χωρίζεται σε δύο κατηγορίες ανάλογα με τον μετασχηματισμό που χρειάζεται να βρούμε. Αυτές είναι η καθολική και η τοπική ευθυγράμμιση.

Η καθολική (Global Registration) και η τοπική (Local Registration) καταχώρηση νεφών σημείων είναι δύο διαφορετικές προσεγγίσεις για την ευθυγράμμιση δύο ή παραπάνω σετ σημείων.

Οι global και οι local μέθοδοι ευθυγράμμισης χρησιμοποιούνται συχνά μαζί σε μια διαδικασία δύο σταδίων (Σχήμα: 2.22). Πρώτον, χρησιμοποιείται μία μέθοδος καθολικής ευθυγράμμισης για να βρεθεί ένας πρόχειρος μετασχηματισμός μεταξύ των δύο νέφουν σημείων. Στη συνέχεια, χρησιμοποιείται μια τοπική μέθοδος ευθυγράμμισης για να τελειοποιηθεί η ευθυγράμμιση και να παραχθεί ένα πιο ακριβές αποτέλεσμα.



Σχήμα 2.22: Ευθυγράμμιση Νέφους Σημείων [4]

2.5.1 Global Registration

Οι μέθοδοι καθολικής ευθυγράμμισης προσπαθούν να βρουν τον γενικό μετασχηματισμό (μετατόπιση, περιστροφή και κλιμάκωση) που ευθυγραμμίζει τα δύο σετ σημείων. Αυτοί οι μέθοδοι ευθυγράμμισης δεν απαιτούν μία καλή αρχική ευθυγράμμιση μεταξύ των δυο νεφών αλλά μπορεί να είναι υπολογιστικά ακριβές. Ωστόσο, οι μέθοδοι καθολικής ευθυγράμμισης είναι σε θέση να χειριστούν μετασχηματισμούς με μεγάλες γωνιές περιστροφής και μεγάλες μετατοπίσεις ενώ παράλληλα μπορούν να παράγουν ακριβή αποτελέσματα ακόμα και όταν τα δύο νέφη σημείων έχουν σημαντικές διάφορές.

2.5.2 Local Registration

Οι τοπικές μέθοδοι ευθυγράμμισης προσπαθούν να βρουν τον τοπικό μετασχηματισμό (μετατόπιση, περιστροφή και κλιμάκωση) που ευθυγραμμίζει τα δύο νέφη σημείων σε μια μικρή γειτονιά γύρω από κάθε σημείο. Οι τοπικές μέθοδοι εγγραφής είναι συνήθως λιγότερο δαπανηρές υπολογιστικά από τις καθολικές μεθόδους αλλά χρειάζονται μια καλή αρχική συνθήκη ενώ ταυτόχρονα είναι πιο ανθεκτικές σε θόρυβο. Ωστόσο, οι τοπικές μέθοδοι ευθυγράμμισης μπορεί να είναι ευαίσθητες στα τοπικά ελάχιστα και δεν είναι σε θέση να χειριστούν μεγάλους μετασχηματισμούς.

Iterative Closest Point (ICP)

Ο ICP είναι ένας από του πιο γνωστούς αλγορίθμους που χρησιμοποιείται για την ελαχιστοποίηση της διαφοράς μεταξύ διαφορετικών νεφών σημείων. Αυτός ο αλγόριθμος χρησιμοποιείται

κυρίως για την ανακατασκευή 2Δ ή 3Δ επιφανειών από διαφορετικές σαρώσεις αλλά και για τη ευθυγράμμιση 3Δ μοντέλων. Ο ICP συγκλίνει μονοτονικά στο πλησιέστερο τοπικό ελάχιστο που βελτιστοποιεί το μέσο τετραγωνικό σφάλμα [26] ως εκ τούτου χρειάζεται μια καλή αρχική συνθήκη ώστε το τοπικό ελάχιστο που θα συγκλίνει ο αλγόριθμος να ταυτίζεται και με το ολικό ελάχιστο.

$$e(T) = \sum_{(p,q) \in K} \|p - Tq\|^2 \quad (2.24)$$

Point to Point ICP

όπου:

p, q : Σημεία του νέφους \mathbf{P} και \mathbf{Q} , αντίστοιχα

K : Το σύνολο αντιστοιχίας, που αντιστοιχεί σημεία του \mathbf{P} στο \mathbf{Q}

T : Ο πίνακας μετασχηματισμού

Όμως υπάρχουν διαφορετικές παραλλαγές του ICP χρησιμοποιούν διαφορετική συνάρτηση κόστους, όπως [27] με συνάρτηση 2.25, η οποία έχει βρεθεί ότι συγκλίνει πιο γρήγορα σε σχέση με το αλγόριθμο Point-to-Point ICP [28], ή η παραλλαγή [29] που χρησιμοποιείται όταν το νέφος σημείων έχει πληροφορία και για τα χρώματα.

$$e(T) = \sum_{(p,q) \in K} ((p - Tq) \cdot n_p)^2 \quad (2.25)$$

Point to Plane ICP

όπου:

p, q : Σημεία του νέφους \mathbf{P} και \mathbf{Q} , αντίστοιχα

K : Το σύνολο αντιστοιχίας, που αντιστοιχεί σημεία του \mathbf{P} στο \mathbf{Q}

T : Ο πίνακας μετασχηματισμού

n_p : Το κάθετο διάνυσμα (normal Vector) του σημείου p

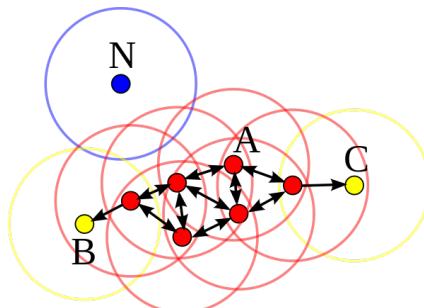
2.6 Ομαδοποίηση 3Δ σημείων

Η ομαδοποίηση σε τρισδιάστατα νέφη σημείων αλλά και γενικότερα περιλαμβάνει τη διαίρεση ενός συνόλου σημείων σε μια συλλογή ξεχωριστών συστάδων (clusters), όπου κάθε ομάδα αντιπροσωπεύει ένα σετ σημείων που μοιράζονται παρόμοιες χωρικές σχέσεις και χαρακτηριστικά. Ο στόχος της ομαδοποίησης είναι ο εντοπισμός αυτών των ομαδοποιήσεων και η εξαγωγή σημαντικών πληροφοριών από τα δεδομένα. Έχουν αναπτυχθεί πολυνάριθμοι αλγόριθμοι ομαδοποίησης για την επίλυση του προβλήματος της τρισδιάστατης ομαδοποίησης σημείων. Αυτοί οι αλγόριθμοι μπορούν να ταξινομηθούν σε δύο κύριες κατηγορίες, ιεραρχική ομαδοποίηση (hierarchical clustering) και τμηματική ομαδοποίηση (partitional clustering). Η μεν πρώτη κατηγορία ομαδοποίησης οργανώνει τα σημεία σε μία ιεραρχία συστάδων, ξεκινώντας από ένα ενιαίο σύμπλεγμα που περιέχει όλα τα σημεία. Επαναληπτικά συγχωνεύουν ή διαχωρίζουν συστάδες με βάση ένα μέτρο ομοιότητας, δημιουργώντας ένα δενδρόγραμμα που αντιπροσωπεύει τις ιεραρχικές σχέσεις μεταξύ των συστάδων. Από την άλλη πλευρά η δεύτερη κατηγορία ομαδοποίησης διαιρεί απευθείας τα δεδομένα σε έναν προκαθορισμένο αριθμό συστάδων ή προκύπτει αυτός ο αριθμός από τις εσωτερικές παραμέτρους του αλγορίθμου. Αυτοί οι αλγόριθμοι

στοχεύουν στην ελαχιστοποίηση της διακύμανσης εντός της ομάδας ή στη μεγιστοποίηση της ομοιότητας μεταξύ των συστάδων.

2.6.1 DBSCAN

Ένας από τους πιο γνωστούς αλγορίθμους είναι ο “Density-based spatial clustering of applications with noise” ή DBSCAN που προτάθηκε για πρώτη φορά το 1996 [30]. Είναι ένας αλγόριθμος ομαδοποίησης που βασίζεται στην πυκνότητα των δεδομένων ενός συνόλου σημείων σε κάποιο χώρο. Αυτό σημαίνει ότι ομαδοποιεί σημεία που βρίσκονται αρκετά κοντά μεταξύ τους, δηλαδή σημεία με πολλούς κοντινούς γείτονες. Με αυτόν τον τρόπο απομακρύνει τα ακραία σημεία (outliers) που βρίσκονται μόνα τους σε περιοχές χαμηλής πυκνότητας και οι κοντινότεροι τους γείτονες είναι πολύ μακριά. Ο αλγόριθμος αυτός δουλεύει με τον αρχικό εντοπισμό των ομάδων (κόκκινα σημεία στην εικόνα 2.23), οι οποίες αποτελούνται από σημεία δεδομένων και έχουν τουλάχιστον ένα προκαθορισμένο αριθμό ελάχιστων σημείων αλλά και να βρίσκονται αρκετά κοντά. Στη συνέχεια οι ομάδες αυτές λειτουργούν ως σημεία εκκίνησης για την επέκταση των συστάδων διευρύνοντας την ίδια την ομάδα. Αυτό γίνεται ελέγχοντας τα γείτονικά σημεία που βρίσκονται εντός της ελάχιστης απόστασης, αλλά δεν έχουν από μόνα τους απαραίτητους γείτονες (κίτρινα σημεία στην εικόνα 2.23). Αν πληρούν τις προϋποθέσεις τότε η ομάδα επεκτείνεται και με αυτά τα σημεία και συνερίζεται η ίδια διαδικασία επαναληπτικά. Τέλος όσα σημεία έχουν απομείνει (μπλε σημεία στην εικόνα 2.23) θεωρούνται ότι είναι θόρυβος και δεν ανήκουν σε καμία ομάδα.



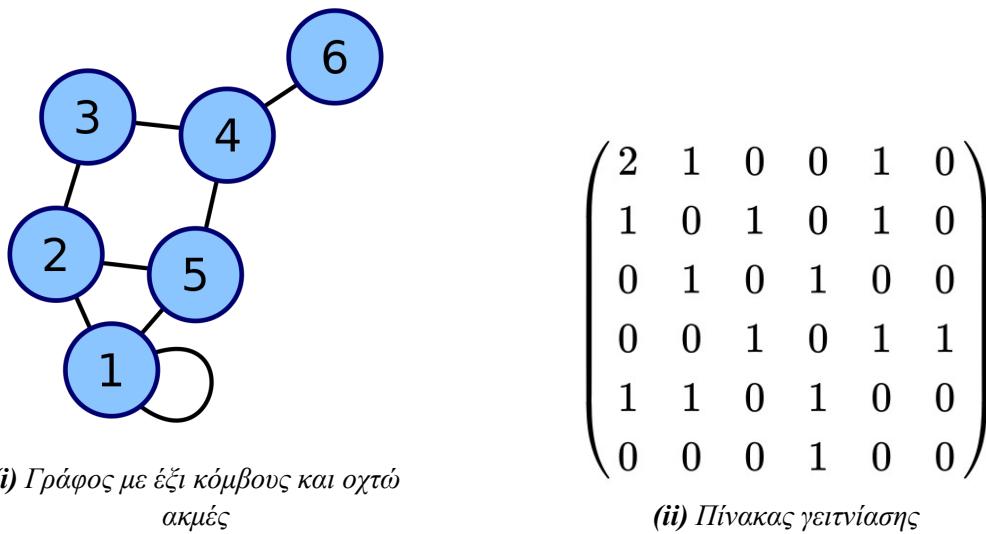
Σχήμα 2.23: Ο αλγόριθμος DBSCAN

2.6.2 HDBSCAN

Από την άλλη πλευρά ένας άλλος επίσης γνωστός αλγόριθμος είναι ο HDBSCAN (Hierarchical DBSCAN) [31]. Είναι ένας αλγόριθμος ομαδοποίησης που βασίζεται στην πυκνότητα και είναι ιδιαίτερα χρήσιμος για την επεξεργασία συνόλων δεδομένων με ποικίλες πυκνότητες και ομάδες διαφορετικών μεγεθών και σχημάτων. Όπως φανερώνει και το όνομά του είναι μία επέκταση του αλγορίθμου DBSCAN που ξεπερνά ορισμένους από τους περιορισμούς του και είναι πιο ανθεκτικός στο θόρυβο. Ο HDBSCAN ξεκινά με τον εντοπισμό κεντρικών σημείων όπως και ο κλασικός DBSCAN, τα οποία είναι σημεία που έχουν έναν ελάχιστο αριθμό γειτόνων μέσα σε ένα συγκεκριμένο όριο απόστασης. Αυτά τα βασικά σημεία συνδέονται στη συνέχεια μεταξύ τους για να σχηματίσουν συστάδες. Η μόνη διαφορά είναι ότι ο συγκεκριμένος αλγόριθμος χρησιμοποιεί μια ιεραρχική προσέγγιση ομαδοποίησης, όπου οι ομάδες συγχωνεύονται και διαχωρίζονται ανάλογα με τις ανάγκες. Με αυτό τον τρόπο ο HDBSCAN μπορεί να αναγνωρίσει αποτελεσματικά ομάδες σε σύνολα δεδομένων με ποικίλες πυκνότητες, σε αντίθεση με τους παραδοσιακούς αλγόριθμους ομαδοποίησης που αντιμετωπίζουν αυτό το πρόβλημα.

2.7 Γράφοι

Στα μαθηματικά και πιο συγκεκριμένα στη θεωρία γράφων, ένας **γράφος** $G = (V, E)$ είναι μία δομή που αντιστοιχεί σε ένα σύνολο αντικειμένων στα οποία ορισμένα ζεύγη αντικειμένων είναι κατά κάποια έννοια «σχετικά». Τα αντικείμενα αντιστοιχούν σε μαθηματικές έννοιες που ονομάζονται κορυφές, V , (ονομάζονται επίσης κόμβοι ή σημεία) και κάθε ένα από τα σχετικά ζεύγη κορυφών ονομάζεται ακμή, E , (ονομάζεται επίσης σύνδεσμος ή γραμμή) [32]. Βρίσκουν εφαρμογή σε πολλές επιστήμες εκτός των μαθηματικών, όπως τη βιολογία για την περιγραφή των σχέσεων μεταξύ γονιδίων σε ένα γονιδίωμα, τη χημεία για περιγραφή της δομής μιας χημικής ένωσης αλλά και στη θεωρία κυκλωμάτων για την περιγραφή ενός κυκλώματος. Τέλος, ένας γράφος μπορεί να προσδιορίζεται πλήρως από τον πίνακα γειτνίασής του A (Adjacency matrix), ο οποίος είναι ένας $n \times n$ τετράγωνος πίνακας (n το πλήθος των κόμβων), με το A_{ij} να καθορίζει το βάρος της εκάστοτε ακμής εάν πρόκειται για σταθμισμένο γράφο ή στη γενική περίπτωση δηλώνει το εάν υπάρχει ακμή που συνδέει την κορυφή i με την κορυφή j .



Σχήμα 2.24: Παράδειγμα γράφου και του αντίστοιχου πίνακα γειτνίασης [5]

Υπάρχουν πολλοί διαφορετικοί τύποι γράφων οι οποίοι συνδυάζουν έναν ή και περισσότερες από τις κατηγορίες. Μερικές από τις πιο γνωστές είναι:

- **Κατευθυνόμενος Γράφος:** είναι ένα γράφος στο οποίο οι ακμές έχουν προσανατολισμό, δηλαδή είναι διαφορετική η ακμή που ενώνει τον κόμβο i με τον j (E_{ij}) και η ακμή που ενώνει τον j με τον i (E_{ji})
- **Μη Κατευθυνόμενος Γράφος:** είναι ακριβώς το αντίθετο του κατευθυνόμενου, δηλαδή οι ακμές δεν έχουν προσανατολισμό και αποτελεί τον πιο συνηθισμένο τύπο γράφου.
- **Συνδεδεμένος Γράφος:** στον οποίο όλοι οι κόμβοι συνδέονται μεταξύ τους μεσολαβώντας τουλάχιστον μία ακμή.
- **Πλήρη Γράφος:** όπου κάθε κόμβος συνδέεται με όλους τους άλλους κόμβους μεσολαβώντας μια ακμή.
- **Ακυκλος Γράφος:** στον οποίο δεν υπάρχουν κύκλοι (loops). Κύκλοι θεωρούνται οι διαδρομές που ξεκινούν και τελειώνουν στον ίδιο κόμβο.

- **Σταθμισμένος Γράφος:** στον οποίο κάθε ακμή έχει ένα βάρος που σχετίζεται με τον εαυτό της. Το βάρος μιας ακμής μπορεί να αντιπροσωπεύει την απόσταση μεταξύ δύο κόμβων, το κόστος της διαδρομής μεταξύ δύο κόμβων ή την ισχύ της σύνδεσης μεταξύ δύο κόμβων.

2.7.1 Αντιστοίχηση Γράφων

Η Αντιστοίχηση Γράφων (Graph Matching - GM) είναι ένα θεμελιώδες αλλά δύσκολο πρόβλημα στην αναγνώριση προτύπων αλλά και στην εξόρυξη δεδομένων. Στοχεύει να βρει μία αντιστοίχηση κόμβου με κόμβο μεταξύ πολλαπλών γράφων, επιλύοντας ένα πρόβλημα NP-hard [33]. Ένα πρόβλημα NP-hard είναι ένα πρόβλημα που είναι τουλάχιστον εξίσου δύσκολο με τα δυσκολότερα προβλήματα στο NP (nondeterministic polynomial time), το οποίο είναι μία κατηγορία προβλημάτων που μπορούν να επαληθευτούν γρήγορα οι λύσεις τους.

Η επίλυση ενός προβλήματος αντιστοίχησης γράφων μπορεί να χωριστεί στα ακόλουθα βήματα:

1. Εξαγωγή χαρακτηριστικών των ακμών και (ή) κορυφών από τα γράφους που θέλουμε να αντιστοιχίσουμε. Αυτά τα χαρακτηριστικά μπορεί να είναι τα ίδια τα βάρη του γράφου εάν είναι σταθμισμένος ή οποιοδήποτε άλλο χαρακτηρισμό θέλουμε εμείς.
2. Δημιουργία του πίνακα ομοιότητας (affinity matrix) από τα χαρακτηριστικά των κόμβων/άκμων.
3. Λύση το προβλήματος αντιστοίχησης γράφων με κάποιο κατάλληλο αλγόριθμο.

Αν θέλαμε να περιγράψουμε το συγκεκριμένο πρόβλημα με μαθηματικούς όρους αντό θα είχε την ακόλουθη μορφή, γνωστή ως Lawler's Quadratic Assignment Problem (Lawler's QAP) [34], [35]:

$$\begin{aligned} & \max_{\mathbf{X}} \text{vec}(\mathbf{X})^T K \text{vec}(\mathbf{X}) \\ & \text{με } \mathbf{X} \in \{0, 1\}^{n_1 \times n_2}, \mathbf{X}\mathbf{1} = \mathbf{1}, \mathbf{X}^T \mathbf{1} \leq \mathbf{1} \end{aligned} \quad (2.26)$$

όπου:

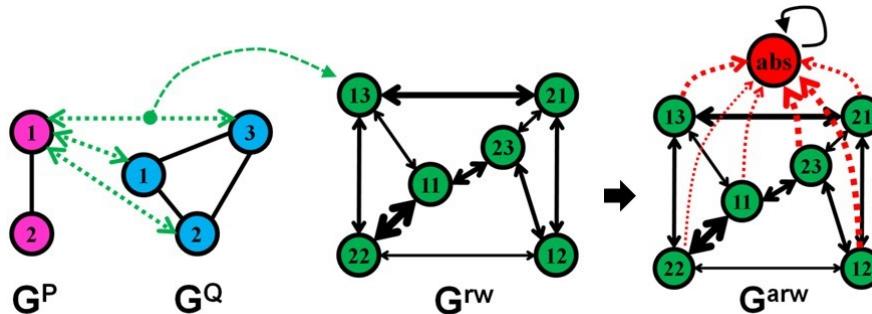
- n_1, n_2 : Το πλήθος των κόμβων των δύο γράφων
- \mathbf{X} : είναι γνωστός ως ο πίνακας μετάθεσης (permutation matrix) που κωδικοποιεί το αποτέλεσμα της αντιστοίχησης, υποθέτοντας: $n_1 \leq n_2$
- $\text{vec}(\mathbf{X})$: είναι μορφή διανυσματοποίησης κατά στήλη του \mathbf{X}
- $\mathbf{1}$: διάνυσμα με όλα τα στοιχεία του να είναι “1”
- K : είναι γνωστός ως ο πίνακας ομοιότητας (affinity matrix) που κωδικοποιεί το πόσο μοιάζουν και οι ακμές μεταξύ τους (μη διαγώνια στοιχεία) αλλά και οι κορυφές επίσης μεταξύ τους (διαγώνια στοιχεία) των γράφων εισόδου.

Αλγόριθμοι Επίλυσης

Ένας από τους πιο γνωστούς αλγορίθμους για την επίλυση προβλημάτων αντίστοιχης γράφων είναι ο “Reweighted Random Walks for Graph Matching - RRWM” [36]. Ο συγκεκριμένος αλγόριθμός επιτυγχάνει την επίλυση του προβλήματος ακόμη και όταν υπάρχει αρκετός θόρυβος στα δεδομένα αλλά και όταν υπάρχουν μη συνηθισμένες τιμές (outliers) μέσα σε αυτά, πράγμα αρκετά συνηθισμένο σε πραγματικά δεδομένα. Αυτό το επιτυγχάνει ενημερώνοντας

και αξιοποιώντας σε κάθε επανάληψη το πόσο σίγουρος είναι για μια συγκεκριμένη αντιστοίχηση.

Πιο αναλυτικά, έστω ότι έχουμε δύο γράφους $G^P = (V^P, E^P, A^P)$ και $G^Q = (V^Q, E^Q, A^Q)$ με κάποια χαρακτηριστικά A^P, A^Q να συνοδεύουν τους κόμβους και (ή) τις ακμές. Το πρόβλημα της αντιστοίχησης μεταξύ των δύο γράφων G^P και G^Q μπορεί να ερμηνευτεί ως ένας τυχαίος περίπατος (random walk) στον γράφο συσχέτισης $G^{rw} = (V^{rw}, E^{rw}, A^{rw})$, ο οποίος αντιπροσωπεύει όλους τους δυνατούς τρόπους συσχέτισης μεταξύ των κορυφών των δύο γράφων. Ο έξτρα κόμβος που έχει προστεθεί στο γράφο G^{arw} βοηθάει στο γεγονός ότι ένας κόμβος (αντιστοίχηση) με περισσότερες ακμές έχει μεγαλύτερη βαρύτητα από άλλους υποψηφίους, πράγμα που θα οδηγούσε σε λάθος αποτελέσματα. Ο αλγόριθμος προσομοιώνει τυχαίους περιπάτους στο γράφο συσχέτισης, οι οποίοι βοηθούν στο να επανασταθμίζονται οι κόμβοι για να ευνοούν τα ακμές που συνδέουν κόμβους που αντιστοιχούν σε σωστή αντιστοίχηση. Αυτό βοηθά στην ενίσχυση των σωστών επιλογών και στην αποδυνάμωση των εσφαλμένων. Μετά από κάθε τυχαίο περίπατο, το βάρος κάθε κόμβου ενημερώνεται με βάση τον αριθμό των φορών που επισκέφτηκε κατά τη διάρκεια του τυχαίου περιπάτου, πράγμα που βοηθά στη βελτίωση της λύσης με την πάροδο του χρόνου. Αυτή η διαδικασία επαναλαμβάνεται συνεχώς μέχρι ο αλγόριθμος να συγκλίνει και να δώσει τον πίνακα αντίστοιχης. Στο τελικό βήμα διακριτοποίησης, όπου ο πίνακας αντίστοιχης πρέπει να έχει μόνο “0” και “1”, μπορεί να υιοθετηθεί οποιοσδήποτε αλγόριθμος γραμμικής ανάθεσης, όπως ένας άπληστος αλγόριθμος ή ο αλγόριθμος Hungarian [37].



Σχήμα 2.25: Βοηθητικοί γράφοι για την αντιστοίχηση χρησιμοποιώντας τον αλγόριθμο RRWM

Ο αλγόριθμος “Hungarian Method” [37] είναι ένας συνδυαστικός αλγόριθμος βελτιστοποίησης που λύνει το πρόβλημα ανάθεσης σε πολυωνυμικό χρόνο. Το πρόβλημα ανάθεσης περιλαμβάνει κάποιους πράκτορες και έναν αριθμό εργασιών. Ο σκοπός είναι να γίνουν όσο το δυνατόν περισσότερες εργασίες αναθέτοντας το πολύ έναν πράκτορα σε κάθε εργασία και το πολύ μία εργασία σε κάθε πράκτορα, με τέτοιο τρόπο ώστε το συνολικό κόστος της ανάθεσης να ελαχιστοποιείται [38]. Αυτό το πρόβλημα μοιάζει αρκετά με την αντιστοίχηση γράφων όταν έχουμε ήδη έναν πίνακα με τις πιθανότητες αντίστοιχης ενός κόμβου από τον έναν γράφο με τον αντίστοιχο κόμβο του άλλου γράφου και θέλουμε να κάνουμε την τελική αντιστοίχηση.

Κεφάλαιο 3

Σχετική Εργασία

Σε αυτό το κεφάλαιο θα περιγράφουν δημοσιεύσεις που έχουν να κάνουν με το κομμάτι της κατάτμησης είτε εικόνας είτε νέφους σημείων αλλά και θα παρουσιαστούν μέθοδοι που προτάθηκαν για την ευθυγράμμιση νεφών σημείων χρησιμοποιώντας πληροφορία από την κατάτμηση.

3.1 Σημασιολογική Κατάτμηση Εικόνας

Η σημασιολογική κατάτμηση (semantic segmentation) είναι ένας κλάδος της υπολογιστικής όρασης που στοχεύει στο να δώσει μία ετικέτα κλάσης σε κάθε pixel μιας εικόνας. Αυτό έρχεται σε αντίθεση με την ανίχνευση αντικειμένων, η οποία αναγνωρίζει και εντοπίζει μόνο αντικείμενα σε μια εικόνα. Στη σημασιολογική κατάτμηση, ο στόχος είναι να οριοθετηθούν με ακρίβεια τα όρια διαφορετικών αντικειμένων στην εικόνα, δημιουργώντας μια μάσκα pixel για κάθε κατηγορία αντικειμένων. Η σημασιολογική κατάτμηση είναι ένα ισχυρό εργαλείο για την ανάλυση και την κατανόηση εικόνων. Έχει ένα ευρύ φάσμα εφαρμογών, όπως:

- **Ιατρικές εικόνες:** Η σημασιολογική κατάτμηση μπορεί να χρησιμοποιηθεί για τον εντοπισμό και την τμηματοποίηση διαφορετικών δομών σε ιατρικές εικόνες, όπως όργανα ή όγκοι. Αυτό μπορεί να χρησιμοποιηθεί για τη διάγνωση, τον σχεδιασμό θεραπείας και την παρακολούθηση της εξέλιξης της νόσου.
- **Αυτόνομα οχήματα:** Η σημασιολογική τμηματοποίηση είναι απαραίτητη για αυτού του είδους οχήματα για να κατανοήσουν το περιβάλλον τους. Το αυτοκίνητο πρέπει να μπορεί να αναγνωρίζει και να κατηγοριοποιεί αντικείμενα όπως πεζούς, οχήματα και σήματα κυκλοφορίας με σκοπό να χρησιμοποιούνται αυτές τις πληροφορίες για την ασφαλή πλοήγηση.
- **Δορυφορικές εικόνες:** Σε αυτήν την εφαρμογή η σημασιολογική κατάτμηση μπορεί να χρησιμοποιηθεί για την ανάλυση δορυφορικών εικόνων για εφαρμογές όπως η ταξινόμηση χρήσεων γης, η ανίχνευση της αποψύλωσης των δασών αλλά και για τον πολεοδομικό σχεδιασμό.

Η σημασιολογική κατάτμηση είναι ένας ταχέως αναπτυσσόμενος τομέας έρευνας και έχει σημειωθεί σημαντική πρόοδος τα τελευταία χρόνια. Οι αλγόριθμοι βαθιάς μάθησης ήταν ιδιαίτερα επιτυχημένοι στη σημασιολογική κατάτμηση και είναι πλέον η τελευταία λέξη της τεχνολογίας για πολλές εφαρμογές. Η επιτυχία των αλγορίθμων βαθιάς μάθησης έχει οδηγήσει στην ανάπτυξη πολλών διαφορετικών αρχιτεκτονικών δικτύων με καθένα τη δικιά του προσέγγιση.

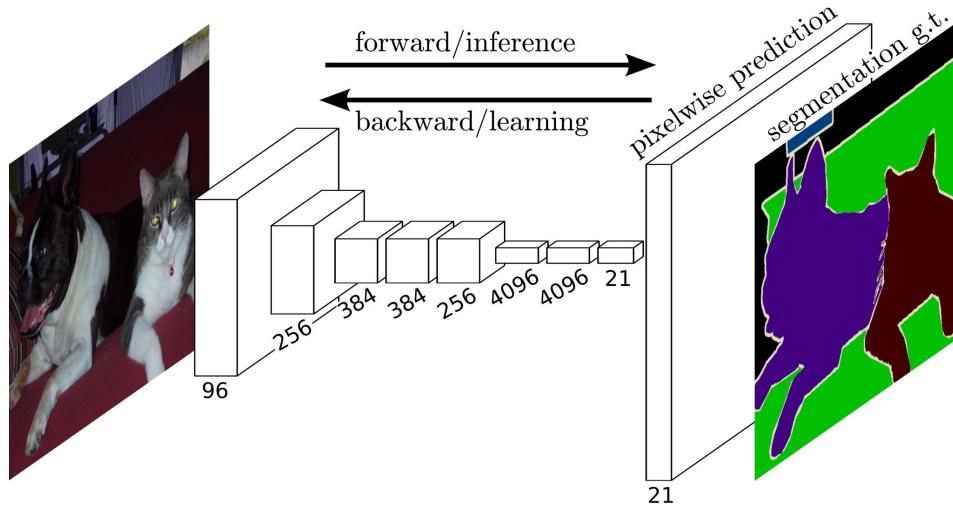
3.1.1 Encoder-Decoder

Οι αρχιτεκτονικές κωδικοποιητή-αποκωδικοποιητή (encoder-decoder) έχουν γίνει μια δημοφιλής επιλογή για εργασίες σημασιολογικής κατάτμησης στην επεξεργασία εικόνας. Αυτές οι αρχιτεκτονικές αποτελούνται από δύο κύρια στοιχεία: έναν κωδικοποιητή και έναν αποκωδικοποιητή. Ο κωδικοποιητής είναι υπεύθυνος για την εξαγωγή χαρακτηριστικών από την εικόνα εισόδου, ενώ ο αποκωδικοποιητής αναδομεί την μάσκα της εικόνα στην επιθυμητή ανάλυση ενώ εκχωρεί σημασιολογικές ετικέτες σε κάθε pixel.

Fully Convolutional Network (FCN)

Από τα πρώτα νευρωνικά δίκτυα που προτάθηκαν για κατάτμηση εικόνας ήταν το “*Fully Convolutional Networks for Semantic Segmentation*” [39]. Το FCN είναι ένας τύπος αρχιτεκτονικής βαθιών νευρωνικών δίκτυων που χρησιμοποιείται για τη σημασιολογική κατάτμηση. Σε αντίθεση με τα παραδοσιακά συνελικτικά νευρωνικά δίκτυα (CNN), τα οποία χρησιμοποιούν πλήρως συνδεδεμένα επίπεδα (MLP) στο τέλος τους για την ταξινόμηση εικόνων, τα FCN χρησιμοποιούν μόνο συνελικτικά και pooling στρώματα. Αυτό επιτρέπει στα FCN να παράγουν προβλέψεις εξόδου, οι οποίες μπορούν να χρησιμοποιηθούν για την κατάτμηση εικόνων σε πολλαπλές κλάσεις.

Ένα από τα κύρια πλεονεκτήματα των FCN είναι ότι μπορούν να χρησιμοποιηθούν για την κατάτμηση εικόνων ανεξαρτήτου μεγέθους. Αυτό συμβαίνει επειδή τα FCN δεν χρησιμοποιούν πλήρως συνδεδεμένα επίπεδα, τα οποία έχουν σταθερό μέγεθος, αλλά συνελίξεις οι οποίες είναι τοπικές εκ φύσης τους (χρησιμοποιούν μόνο ένα μέρος της εικόνας εισόδου). Ως αποτέλεσμα, τα FCN μπορούν να εφαρμοστούν σε εικόνες οποιουδήποτε πλάτους ή ύψους, καθιστώντας τα ένα ευέλικτο εργαλείο για μία ποικιλία εφαρμογών. Τέλος η αποφυγή της χρήσης πυκνών στρωμάτων (MLP) σημαίνει λιγότερες παραμέτρους καθιστώντας τα δίκτυα πιο γρήγορα στην εκπαίδευση τους.



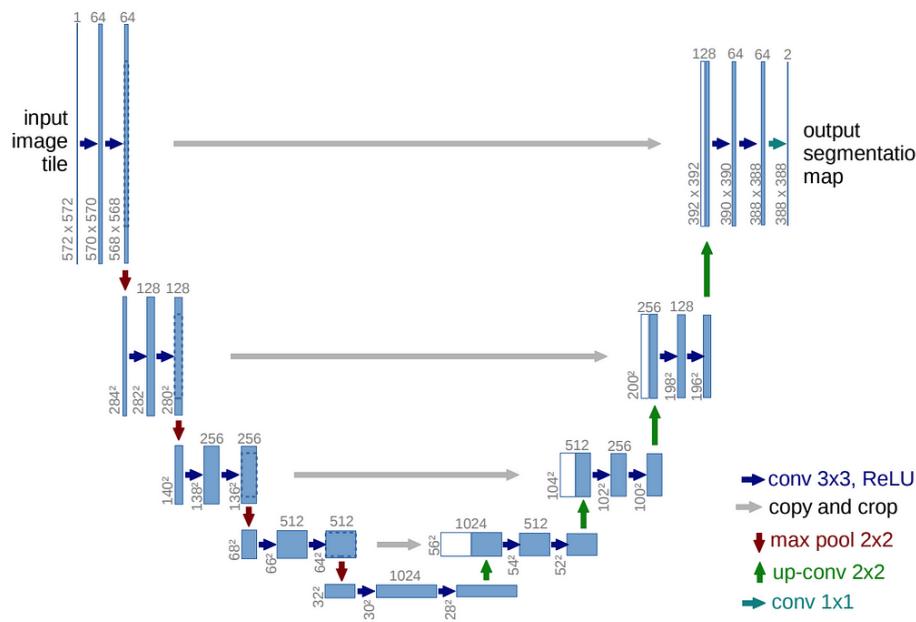
Σχήμα 3.1: Αρχιτεκτονική του δικτύου FCN

Unet

Το U-Net είναι μια αρχιτεκτονική συνελικτικού νευρωνικού δικτύου (CNN) για κατάτμηση εικόνας. Αναπτύχθηκε αρχικά για τμηματοποίηση βιοϊατρικών εικόνων μέσα από το “*U-Net*:

Convolutional Networks for Biomedical Image Segmentation” [40], αλλά έκτοτε έχει εφαρμοστεί σε ένα ευρύ φάσμα εργασιών, συμπεριλαμβανομένης της κατάτμησης εικόνων από αστικά περιβάλλοντα αλλά και εικόνων από δορυφόρους.

Η ονομασία U-Net προέρχεται από την δομή της αρχιτεκτονικής του δικτύου, η οποία έχει σχήμα “U”. Το δίκτυο αποτελείται από δύο κύρια μέρη: τον κωδικοποιητή (encoder) και τον αποκωδικοποιητή (decoder). Ο κωδικοποιητής είναι υπεύθυνος για την εξαγωγή χαρακτηριστικών από την εικόνα εισόδου, ενώ ο αποκωδικοποιητής είναι υπεύθυνος για την ανακατασκευή της εξόδου εικόνας. Ο κωδικοποιητής και ο αποκωδικοποιητής συνδέονται μεταξύ τους με συνδέσεις άλματος (skip connections), οι οποίες επιτρέπουν στον αποκωδικοποιητή να έχει πρόσβαση σε χαρακτηριστικά από τον κωδικοποιητή σε διαφορετικά επίπεδα κλίμακας.



Σχήμα 3.2: Αρχιτεκτονική ενός δικτύου U-Net

Πιο αναλυτικά, η αρχιτεκτονική U-Net αποτελείται από μια διαδρομή συστολής (κωδικοποιητής) και μια διαδρομή που διαστέλλει το αποτέλεσμα (αποκωδικοποιητής). Η διαδρομή συστολής αποτελείται από μια σειρά συνελικτικών και pooling στρωμάτων που μειώνουν τις χωρικές διαστάσεις της εικόνας εισόδου ενώ αυξάνουν τον αριθμό των καναλιών. Με αυτόν τον τρόπο θα μπορούσαμε να πούμε ότι το δίκτυο συμπιέζει την πληροφορία της εικόνας έχοντας όμως κάποιες απώλειες οι οποίες δεν είναι απαραίτητες για την ανακατασκευή της μάσκας κατάτμησης. Στη συνέχεια ακολουθεί η αντίθετη διαδρομή που αποτελείται και αυτή από μια σειρά αντίστροφων συνελικτικών και pooling στρωμάτων που αυξάνουν τις χωρικές διαστάσεις της εικόνας εξόδου ενώ μειώνουν τον αριθμό των καναλιών. Παράλληλα υπάρχουν και οι συνδέσεις παράκαμψης συνδέουν τα αντίστοιχα στρώματα στη διαδρομή συστολής με τα αντίστοιχα στρώματα στην αντίθετη διαδρομή. Αυτές επιτρέπουν στο U-Net να συλλαμβάνει τόσο λεπτομερή όσο και γενικευμένες πληροφορίες στην είσοδο εικόνα, πράγμα που δεν υπάρχει στο δίκτυο FCN.

Εν κατακλείδι, το U-Net έχει πολλά πλεονεκτήματα σε σχέση με άλλες αρχιτεκτονικές CNN για την κατάτμηση εικόνας. Αρχικά, είναι σε θέση να τμηματοποιήσει τις εικόνες με υψηλή ακρίβεια, ακόμη και όταν τα αντικείμενα ενδιαφέροντος είναι μικρά ή έχουν πολύπλοκα σχήματα. Επιπλέον, είναι σχετικά εύκολο στην εκπαίδευση αλλά και στη χρήση, ακόμη και με μεγάλα σύνολα δεδομένων. Τέλος, λόγω της δομής του προσαρμόζεται εύκολα σε καινούργιους

εφαρμογές με αποτέλεσμα να μπορεί να εφαρμοστεί σε ένα ευρύ φάσμα εργασιών κατάτμησης εικόνας.

3.1.2 Two Stages Networks

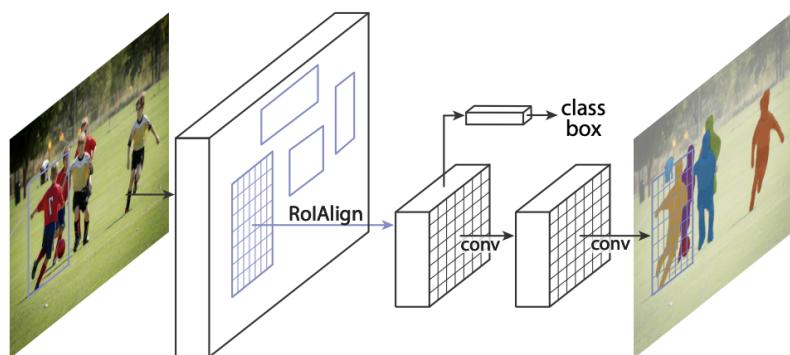
Ένα δίκτυο δύο σταδίων είναι ένας τύπος νευρωνικών δικτύων που αποτελείται από δύο στάδια επεξεργασίας. Το πρώτο στάδιο είναι υπεύθυνο για την εξαγωγή χαρακτηριστικών από τα δεδομένα εισόδου που έχουν να κάνουν με τη θέση χρήσιμων υποπεριοχών τις εικόνας, ενώ το δευτέρο στάδιο είναι υπεύθυνο για την πραγματοποίηση προβλέψεων με βάση αυτών των χαρακτηριστικών. Ένα από τα πιο γνωστά δίκτυα τέτοιου τύπου που προτάθηκαν για κατάτμηση εικόνας είναι το “*Mask R-CNN*”.

Mask R-CNN

Το “*Mask R-CNN*” [41] είναι ένα μοντέλο βαθιάς εκμάθησης που συνδυάζει την ανίχνευση αντικειμένων και το instance segmentation (διαχωρισμό μεμονωμένων αντικειμένων της ίδιας κατηγορίας μέσα σε μια εικόνα). Είναι μια επέκταση της αρχιτεκτονικής “*Faster R-CNN*” [42] αλλά με την προσθήκη της δεινότητας για τμηματοποίησή εικόνας.

Οπως τα περισσότερα δίκτυα αυτού του είδους αποτελείται από δύο στάδια με το πρώτο να είναι το Region Proposal Network (RPN). Το RPN παίρνει μια εικόνα εισόδου και δημιουργεί ένα σύνολο ορθογώνιων προτάσεων (Regions Of Interest - ROI) μέσα στην αρχική εικόνα, καθεμία από τις οποίες προβλέπεται ότι περιέχει ένα αντικείμενο. Το RPN χρησιμοποιεί ένα συνελικτικό νευρωνικό δίκτυο (CNN) για να εξάγει χαρακτηριστικά από την εικόνα εισόδου. Αυτά τα χαρακτηριστικά χρησιμοποιούνται στη συνέχεια για την πρόβλεψη της τοποθεσίας και του μεγέθους των περιοχών.

Το δεύτερο στάδιο του Mask R-CNN παίρνει κάθε πρόταση από το RPN και χρησιμοποιώντας ένα CNN ταξινομεί την υποεικόνα σε μία από τις διαθέσιμες κατηγορίες και προβλέπει μια δυαδική μάσκα (binary mask) για το αντικείμενο μέσα στην πρόταση. Καθώς αυτό γίνεται για κάθε ROI προκύπτουν διαφορετικές μάσκες για κάθε αντικείμενο μέσα στην εικόνα με αποτέλεσμα να έχουμε instance segmentation.



Σχήμα 3.3: Αρχιτεκτονική ενός δικτύου *Mask R-CNN*

Το “*Mask R-CNN*” είναι ένα από τα πιο διαδεδομένα δίκτυα για instance segmentation και υπάρχουν διάφοροι λόγοι για αυτό. Αρχικά είναι ένα από τα πιο ακριβείς μοντέλα για αυτό το σκοπό ενώ παράλληλα είναι δεν είναι πάρα πολύ μεγάλο πράγμα που σημαίνει ότι μπορεί να επεξεργαστεί εικόνες γρήγορα. Τέλος όμως έχει και κάποια μειονεκτήματα που έχουν

να κάνουν κυρίως με την περιπλοκότητα του μοντέλου, πράγμα που το καθιστά δύσκολο να εκπαιδευτεί αλλά και να υλοποιηθεί.

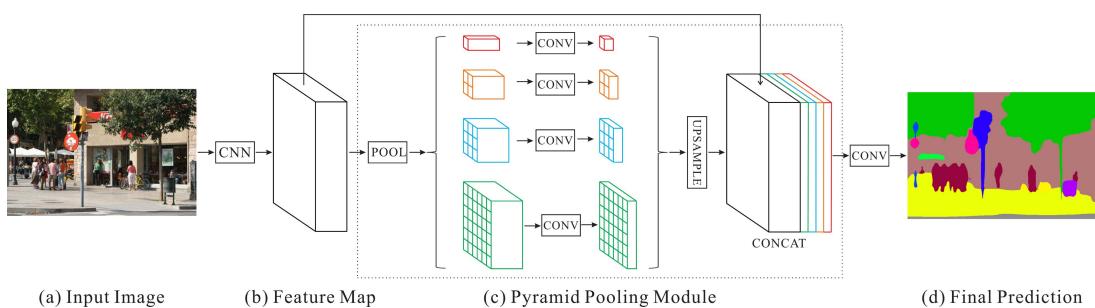
3.1.3 Pyramid Pooling Architectures

Οι αρχιτεκτονικές Pyramid Pooling είναι ένας τύπος αρχιτεκτονικής νευρωνικών δικτύων που χρησιμοποιείται συνήθως σε εργασίες σημασιολογικής κατάτμησης. Έχουν σχεδιαστεί για να αντλούν πληροφορίες από διαφορετικές κλίμακες μιας εικόνας, οι οποίες μπορούν να βοηθήσουν στη βελτίωση της ακρίβειας της εργασίας τμηματοποίησης. Οι αρχιτεκτονικές αυτές συνήθως αποτελούνται από δύο κύρια στοιχεία: ένα κυρίως δίκτυο (Backbone Network) και μια μονάδα pooling. Το δίκτυο κορμού είναι υπεύθυνο για την εξαγωγή χαρακτηριστικών από την εικόνα εισόδου ενώ το pyramid pooling είναι υπεύθυνο να αλλάζει τη κλίμακα της εικόνας λαμβάνοντας με αυτό το τρόπο λεπτομερείς και γενικές πληροφορίες.

PspNet

Η δημοσίευση “*Pyramid Scene Parsing Network*” [43] είναι μια από τις πιο γνωστές που προτίνουν την αρχιτεκτονική Pyramid Pooling για σημασιολογική κατάτμηση. Η βασική ιδέα πίσω από το PSPNet είναι η χρήση μιας μονάδας Pyramid Pooling Module (PPM) για την καταγραφή πληροφοριών σε διαφορετικές κλίμακες. Αυτή η μονάδα συγκεντρώνει χαρακτηριστικά από την εικόνα εισόδου σε διαφορετικές κλίμακες, από ολόκληρη την εικόνα σε μικρές περιοχές. Αυτό επιτρέπει στο δίκτυο να μάθει μια πιο ολοκληρωμένη αναπαράσταση της σκηνής, η οποία μπορεί να βελτιώσει την ακρίβεια κατάτμησης.

Πιο αναλυτικά το PSPNet χρησιμοποιεί συνήθως ένα προεκπαιδευμένο CNN για την εξαγωγή του feature map ως το κυρίως δίκτυο. Το τελικό μέγεθος χάρτη χαρακτηριστικών είναι $1/8$ της εικόνας εισόδου, το οποίο χρησιμοποιείτε ως είσοδο για τη μονάδα PPM. Η συγκεκριμένη δημοσίευση χρησιμοποιεί μία πυραμίδα 4 επιπέδων, εκ των οποίων οι πυρήνες pooling καλύπτουν ολόκληρο, το μισό ή μικρά τμήματα της εικόνας. Στη συνέχεια, ενώνουμε αυτά τα αποτελέσματα, φέρνοντάς τα όμως όλα στις ίδιες χωρικές διαστάσεις, με τον αρχικό χάρτη χαρακτηριστικών και τέλος ακολουθεί ένα στρώμα συνέλιξης για τη δημιουργία του τελικού χάρτη κατάτμησης.



Σχήμα 3.4: Αρχιτεκτονική του *Pyramid Scene Parsing Network*

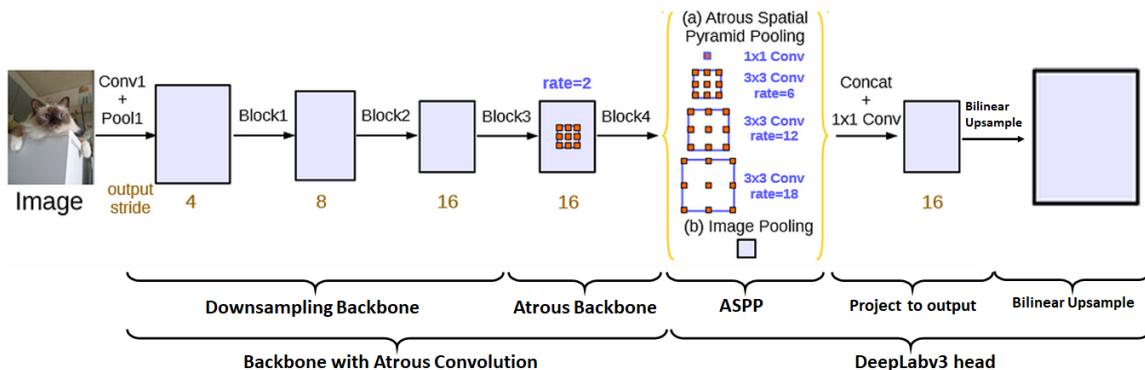
Το συγκεκριμένο δίκτυο αν και μοιάζει αρκετά με τις αρχιτεκτονικές encoder - decoder έχει κάποιες βασικές διάφορές και η πιο σημαντική είναι η εισαγωγή της μονάδας Pyramid Pooling Module (PPM). Σε ένα κλασικό δίκτυο encoder - decoder καθώς “συμπιέζουμε” την εικόνα ακολουθούμε μια γραμμική διαδικασία, δηλαδή το αποτέλεσμα του ενός στρώματος συμπίεσης τροφοδοτείται στο επόμενο, ενώ στη συγκεκριμένη δημοσίευση γίνεται παράλληλα.

DeepLab V3

Παρόμοιο δίκτυο με το PSPNet προτάθηκε μέσω της δημοσίευσης “*Rethinking Atrous Convolution for Semantic Image Segmentation*” [44]. Ωστόσο, διαφέρουν ως προς την προσέγγισή που ακολούθησαν για τη διαδικασία του pooling σε διάφορες κλίμακες. Το μεν PSPNet χρησιμοποίησε κλασικό max-pooling ενώ το DeepLabV3 πρότεινε να χρησιμοποιηθεί ένα είδος συνέλιξης (atrous convolution) το οποίο δεν επηρεάζει πολύ το μέγεθος του αποτελέσματος.

Πιο λεπτομερώς το DeepLabv3 χρησιμοποιεί το atrous convolution (ή dilated convolution) για να μεγαλώσει την περιοχή της εικόνας που λαμβάνεται υπόψιν χωρίς να μικρύνει την ίδια την εικόνα ούτε όμως να αυξήσει τον αριθμό των παραμέτρων μέσω του μεγέθους των φίλτρων (kernel size). Το γεγονός ότι το δίκτυο λαμβάνει περισσότερη περιοχή της εικόνας υπόψιν είναι ένας σημαντικός παράγοντας για να γίνει σωστά η παραγωγή της μάσκας κατάτμησης και αυτό επιτυγχάνεται μέσω του ASPP (Atrous Spatial Pyramid Pooling).

Η μονάδα ASPP προτάθηκε για πρώτη φορά μέσω του “*DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs*” [45] και του DeepLabV2. Αυτό το κομμάτι του δικτύου αποτελείται από πολλαπλές παράλληλες atrous convolution με διαφορετικούς ρυθμούς διαστολής (dilation rate) η κάθε μία και με αυτόν τον τρόπο είναι ικανό να καταγράψει χαρακτηριστικά σε διαφορετικές κλίμακες. Κάθε στρώμα αυτής της συνέλιξης εξάγει χαρακτηριστικά από τον feature map εισόδου σε διαφορετική κλίμακα, παρέχοντας μια ολοκληρωμένη εικόνα του χωρικού πλαισίου που περιβάλλει κάθε pixel.



Σχήμα 3.5: Αρχιτεκτονική του δικτύου DeepLab V3

Γενικά οι αρχιτεκτονικές pyramid pooling έχουν αποδειχθεί αποτελεσματικές σε μια ποικιλία εργασιών, συμπεριλαμβανομένης και της σημασιολογικής κατάτμησης και ο κύριος λόγος είναι ότι μπορούν να αντλούν πληροφορία από διάφορες κλίμακες του feature map. Αυτό είναι σημαντικό επειδή τα αντικείμενα στις εικόνες μπορούν να εμφανίζονται σε διαφορετικά μεγέθη και μια ενιαία κλίμακα εξαγωγής χαρακτηριστικών μπορεί να μην μπορεί να καταγράψει όλες τις σχετικές πληροφορίες. Χρησιμοποιώντας πολλαπλές κλίμακες, η αρχιτεκτονική pyramid pooling μπορεί να αντλήσει τόσο γενικές πληροφορίες όσο και και πιο λεπτομερείς πληροφορίες, οι οποίες μπορούν να βελτιώσουν την απόδοση του δικτύου. Ένας άλλος λόγος είναι ότι οι παραδοσιακές τεχνικές pooling (average/ max) μπορεί να οδηγήσουν σε απώλεια πληροφοριών με την απόρριψη χωρικών λεπτομερειών. Το Pyramid Pooling Module αντιμετωπίζει αυτό το ζήτημα εκτελώντας ομαδοποίηση σε πολλαπλές κλίμακες ανεξάρτητα και στη συνέννοντας τα αποτελέσματα. Αυτό μειώνει τον κίνδυνο απώλειας κρίσιμων πληροφοριών κατά τη διαδικασία pooling. Τέλος, η δομή της πυραμίδας ενισχύει τη χωρική ιεραρχία εντός του χάρτη χαρακτηριστικών (feature map), δηλαδή διαφορετικά επίπεδα πυραμίδας αντιστοιχούν σε διαφορετικά μεγέθη δεκτικών πεδίων (receptive fields), επιτρέποντας στο δίκτυο να κατανο-

ήσει τις iεραρχικές σχέσεις μεταξύ περιοχών εικόνας, πράγμα που βοηθάει για τη σημασιολογική κατάτμηση.

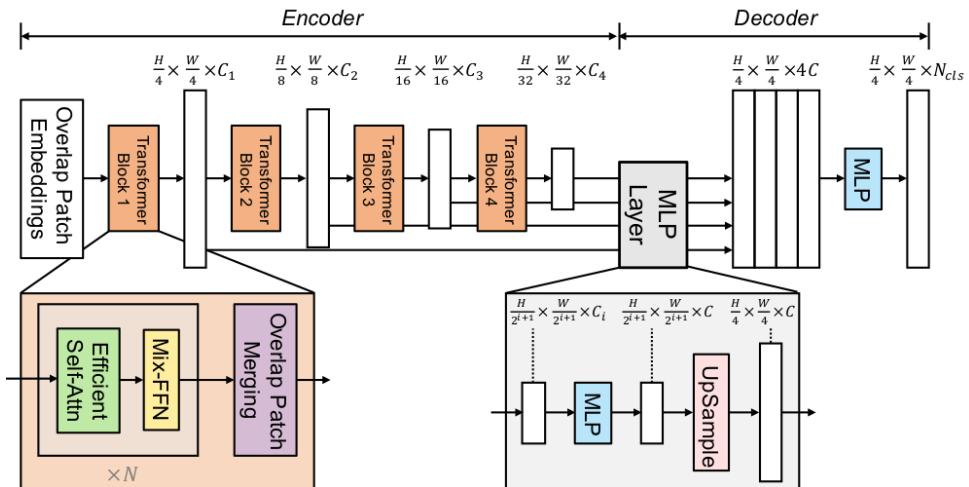
3.1.4 ViT

Μετά την δημοσίευση που προτάθηκαν οι ViT για πρώτη φορά, άρχισαν να παρουσιάζονται και αρχιτεκτονικές που τους χρησιμοποιούσαν και σε εφαρμογές σημασιολογικής κατάτμησης εκτός από κατηγοριοποίησης. Ενδεικτικά κάποιες από αυτές θα παρουσιαστούν στη συνέχεια.

SegFormer

Μία από τις πιο γνωστές αρχιτεκτονικές προτάθηκε από τη δημοσίευση “*SegFormer: Simple and Efficient Design for Semantic Segmentation with Transformers*” [46]. Το SegFormer είναι ένα απλό, αποτελεσματικό και ισχυρό δίκτυο σημασιολογικής κατάτμησης που συνδυάζει μετασχηματιστές με πολυστρωματικά perceptrons (MLP). Το γεγονός ότι αποτελείται από πολύ λιγότερες παραμέτρους σε σχέση μα άλλα συναγωνίσημα δίκτυα το κάνει ιδανική επιλογή σε εφαρμογές όπου χρειάζονται τα αποτελέσματα να εξάγονται σε πραγματικό χρόνο.

Πιο αναλυτικά αυτή η αρχιτεκτονική μοιάζει αρκετά με δομή κωδικοποιητή - αποκωδικοποιητή, αλλά αντί να χρησιμοποιούνται συνελικτικά δίκτυα στις δομές αυτές χρησιμοποιούνται μετασχηματιστές και MLP, αντίστοιχα. Ο SegFormer περιλαμβάνει έναν νέο iεραρχικά δομημένο κωδικοποιητή transformer που εξάγει χαρακτηριστικά από πολλαπλές κλίμακες. Ακόμα, δεν χρειάζεται κωδικοποίηση θέσης (positional encoding), αποφεύγοντας έτσι τα positional embeddings που οδηγούν σε μειωμένη απόδοση όταν η ανάλυση των εικόνων εκπαίδευσης διαφέρουν από αυτές των εικόνων κατά τη διάρκεια του testing. Άλλα χρησιμοποιούνται υποτύμημα της εικόνας που έχουν επικάλυψη μεταξύ (σε αντίθεση με τον κλασικό ViT), με σκοπό να λυθεί το πρόβλημα της ασυνέχειας ανάμεσα σε αυτά.

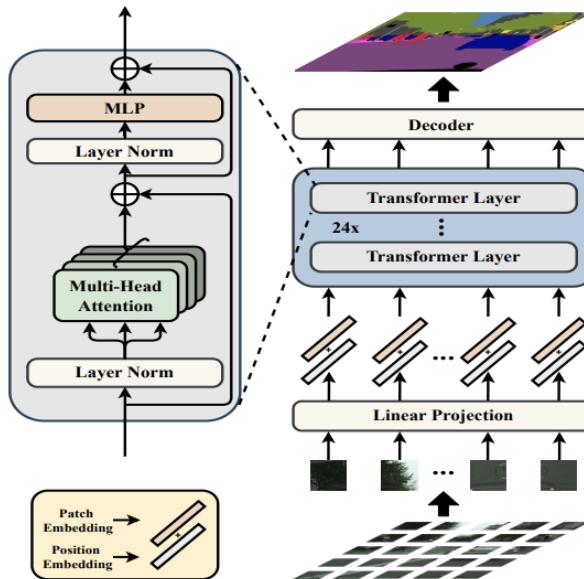


Σχήμα 3.6: Αρχιτεκτονική του δικτύου SegFormer

Τελος, ακολουθεί ο αποκωδικοποιητής MLP, οποίος λαμβάνει τα χαρακτηριστικά πολλαπλής κλίμακας από τον κωδικοποιητή και παράγει την τελική πρόβλεψη κατάτμησης. Ο SegFormer ενσωματώνει ένα ελαφρύ αποκωδικοποιητή που αποτελείται μόνο από στρώματα MLP και με αυτό τον τρόπο αποφεύγει τα υπολογιστικά απαιτητικά στοιχεία που χρησιμοποιούνται συνήθως σε άλλες μεθόδους. Το κύριο στοιχείο που επιτρέπει την χρήση ενός τόσο απλού αποκωδικοποιητή είναι ότι ο iεραρχικός κωδικοποιητής που λόγω της δομής του έχει μεγαλύτερο receptive field από τους παραδοσιακούς κωδικοποιητές CNN.

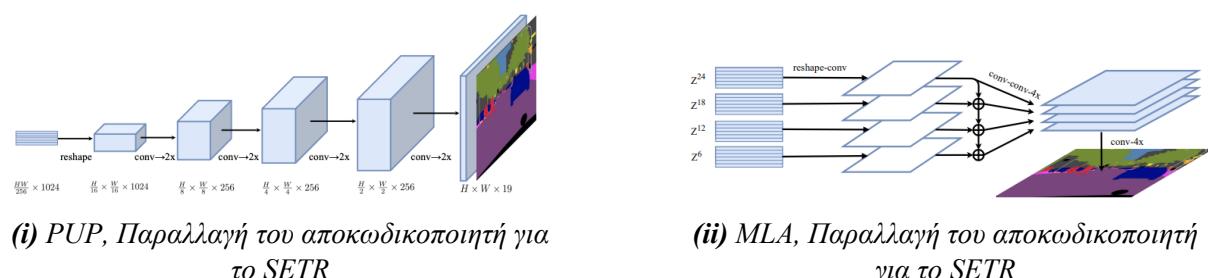
SETR

Μια άλλη εξίσου γνωστή αρχιτεκτονική προτάθηκε μέσω της δημοσίευσής “*Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers*” [47]. Όπως και οι πλειοψηφία των δημοσιεύσεων αποτελείται από δύο κομμάτια, έναν κωδικοποιητή και έναν αποκωδικοποιητή. Ένα από τα χαρακτηριστικά που ξεχωρίζουν αυτήν την αρχιτεκτονική από άλλες είναι ότι δεν υπάρχει μείωση της χωρική ανάλυσης καθώς προχωράμε σε πιο βαθιά επίπεδα του δίκτυου αλλά αντίθετα η πληροφορία της αρχικής εικόνας μοντελοποιείται σε κάθε επίπεδο του κωδικοποιητή.



Σχήμα 3.7: Αρχιτεκτονική του δίκτυου SETR

Ένα άλλο σημαντικό χαρακτηριστικό είναι ότι για να αξιολογηθεί η αποτελεσματικότητα των αναπαραστάσεων των χαρακτηριστικών του κωδικοποιητή, προτάθηκαν τρία διαφορετικά σχέδια αποκωδικοποιητή για την εκτέλεση κατάτμησης. Αρχικά είναι ο αποκωδικοποιητής “**Naive**”, ο οποίος είναι ένα απλό δίκτυο 2 επιπέδων με αρχιτεκτονική, το πρώτο ένα μικρό συνεκτικό δίκτυο και στη συνέχεια ένα στρώμα upsampling. Η άλλη παραλλαγή ονομάζεται “**Progressive UPSampling (PUP)**” και αποτελείται από τέσσερα στάδια το ένα μετά το άλλο, τα οποία αποτελούνται συνελίξεις και στρώματά upsampling μέχρι $2 \times$ για μειωθεί η εισαγωγή θορύβου. Τέλος, προτάθηκε και το “**Multi-Level feature Aggregation (MLA)**”, το οποίο έχει την ίδια λογική με τις pyramid pooling αρχιτεκτονικές, αλλά διαφέρει στο ότι το κάθε επίπεδο έχει την ίδια ανάλυση.



Σχήμα 3.8: Οι παραλλαγές του αποκωδικοποιητή στο δίκτυο SETR

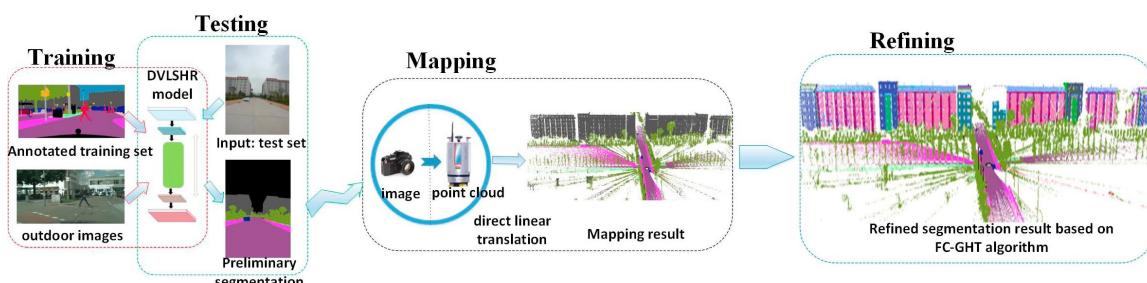
3.2 Σημασιολογική Κατάτμηση Νέφους Σημείων

Η κατάτμηση νέφους σημείων είναι η διαδικασία κατάτμησης ενός συνόλου τρισδιάστατων σημείων σε ένα σύνολο τμημάτων, όπου κάθε τμήμα αποτελείται από σημεία που μοιράζονται παρόμοια χαρακτηριστικά ή ανήκουν στο ίδιο αντικείμενο. Αυτό είναι ένα θεμελιώδες πρόβλημα στην ανάλυση 3D δεδομένων, με εφαρμογές σε ένα ευρύ φάσμα πεδίων, συμπεριλαμβανομένης της ρομποτικής, των αυτόνομων οχημάτων αλλά και άλλων.

3.2.1 Map 3D to 2D

Η δημοσίευση “*Fusion of images and point clouds for the semantic segmentation of large-scale 3D scenes based on deep learning*” [48] προτείνει μια νέα μέθοδο για τη σημασιολογική κατάτμηση τρισδιάστατων σκηνών μεγάλης κλίμακας με τη χρήση εικόνων και τρισδιάστατων σημείων. Σύμφωνα με την προτεινόμενη μέθοδο αρχικά απαιτείται η εξαγωγή χαρακτηριστικών από εικόνες χρησιμοποιώντας ένα μοντέλο μεγάλης κλίμακας και υψηλής ανάλυσης που βασίζεται στο DeepLab-Vgg16 (DVLSHR), που θα βοηθήσουν στη σημασιολογική κατάτμηση της εικόνας. Τα αποτελέσματα κατάτμησης των εικόνων αντιστοιχίζονται στη συνέχεια σε τρισδιάστατα νέφη σημείων σύμφωνα με τις σχέσεις συντεταγμένων μεταξύ των εικόνων και του νέφους σημείων.

Πιο αναλυτικά, αυτή τη μέθοδο μπορούμε να τη χωρίσουμε σε τρία μέρη. Αρχικά απαιτείται η σημασιολογική κατάτμηση των εικόνων από ένα καλά εκπαιδευμένο δίκτυο, το οποίο θα είναι ικανό να διακρίνει τις λεπτομέρειες της σκηνής. Έπειτα ακολουθεί η διαδικασία της αντιστοίχησης των τρισδιάστατων σημείων πάνω στη εικόνες κατάτμησης του προηγούμενου σταδίου. Αυτή η διαδικασία είναι εξαιρετικά σημαντική και απαιτεί να είναι γνωστή η σχετική θέση των αισθητήρων (κάμερα και LIDAR) ώστε να γίνει σωστά ο μετασχηματισμός των 3D σημείων. Τέλος, το τελευταίο στάδιο έχει να κάνει με την γεωμετρία της σκηνής και τα γενικά χαρακτηριστικά της. Δηλαδή μέσω μεθόδων που βρίσκουν τις διάφορες επίπεδες επιφάνειες γίνεται και κατάτμηση ανάμεσα στα κτήρια της σκηνής ή άλλα μεγάλα αντικείμενα.



Σχήμα 3.9: Σημασιολογική κατάτμηση νέφους σημείων χρησιμοποιώντας κατάτμηση εικόνας

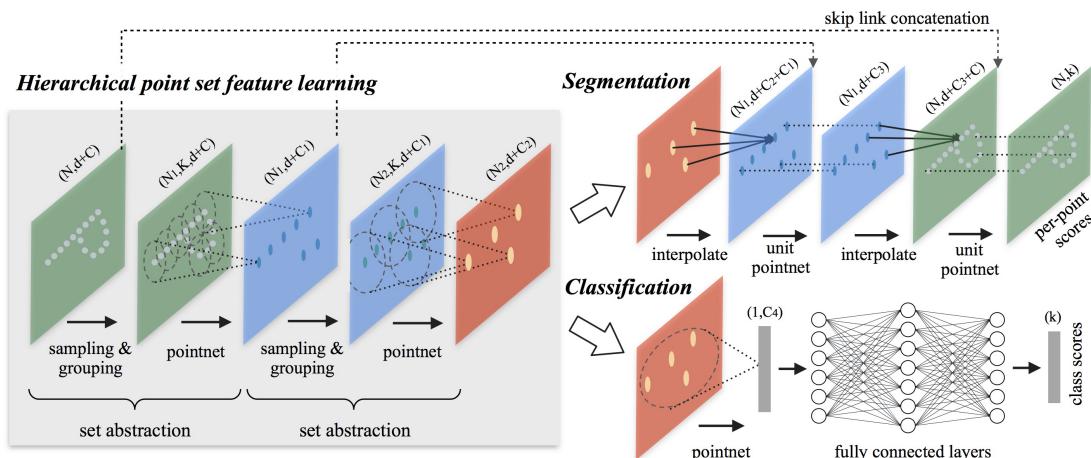
Η προτεινόμενη μέθοδος έχει πολλά πλεονεκτήματα σε σχέση με άλλες μεθόδους που χρησιμοποιούνται στη σημασιολογική κατάτμηση τρισδιάστατων σκηνών. Αρχικά, η μέθοδος χρησιμοποιεί αποτελεσματικά την πληθώρα πληροφοριών που παρέχουν οι εικόνες και έχουν σχέση με τις λεπτομέρειες ενός αντικείμενου αλλά και της ίδιας της σκηνής, ενώ παράλληλα χρησιμοποιεί το νέφος σημείων για την ακριβή περιγραφή της τρισδιάστατης γεωμετρίας. Αυτό το χαρακτηριστικό είναι εξαιρετικά χρήσιμο για την εξαγωγή τοπικών και καθολικών χαρακτηριστικών. Τα τοπικά χαρακτηριστικά εξάγονται από τις εικόνες και έχουν σχέση με τα μικρά αντικείμενα τις σκηνής που δύσκολα διακρίνονται σε ένα νέφος σημείων, ενώ τα καθολικά χαρακτηριστικά εξάγονται από τα τρισδιάστατα νέφη σημείων και περιλαμβάνουν περισσότερο μεγάλα αντικείμενα που επηρεάζουν σε μεγάλη κλίμα τη γεωμετρία της σκηνής.

3.2.2 Native segmentation

Εκτός από τις αρχιτεκτονικές που βασίζονται στη εικόνα για να γίνει η κατάτμηση του νέφους σημείων έχουν προταθεί και αρχιτεκτονικές που βασίζονται εξολοκλήρου στην τρισδιάστατη πληροφορία. Κάποιες από αυτές παρουσιάζονται παρακάτω.

PointNet++

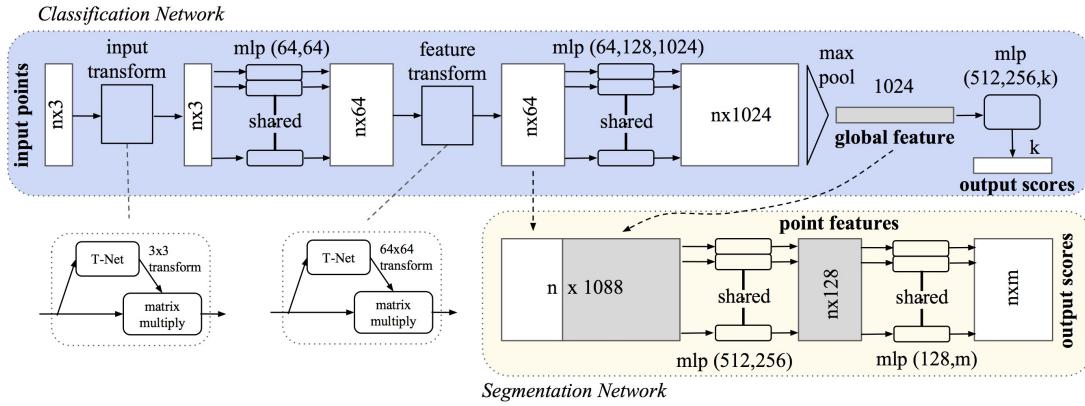
Μία αρκετά δημοφιλής αρχιτεκτονική είναι αυτή που δημοσιεύτηκε μέσω της “*PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space*” [49], η οποία αποτελεί εξέλιξη της “*PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation*” [50]. Είναι μια αρχιτεκτονική για την επεξεργασία συνόλων σημείων και λειτουργεί εφαρμόζοντας επαναληπτικά μια μονάδα “set abstraction” στο σύνολο σημείων εισόδου, η οποία ομαδοποιεί τα σημεία σε τοπικές περιοχές εξάγει χαρακτηριστικά για τις περιοχές αυτές.



Σχήμα 3.10: Αρχιτεκτονική του δικτύου PointNet++

Το στρώμα “set abstraction” παίρνει ως είσοδο τα σημεία και ένα σύνολο διανυσμάτων χαρακτηριστικών. Το επίπεδο ομαδοποιεί πρώτα τα σημεία σε τοπικές περιοχές και στη συνέχεια, για κάθε περιοχή, το στρώμα εξάγει ένα διάνυσμα χαρακτηριστικών με την εφαρμογή του pointnet, που πρακτικά αποτελείται από πλήρως συνδεδεμένα δίκτυα και ένα στρώμα max-pooling. Αυτή η διαδικασία επαναλαμβάνεται με αποτέλεσμα να μειώνεται το πλήθος των σημείων σε κάθε επανάληψη με παρόμοια λογική που ακολουθείται σε μια αρχιτεκτονική κωδικοποιητή- αποκωδικοποιητή. Τέλος, ακολουθεί το κομμάτι που είναι υπεύθυνο για κατηγοριοποίηση και για την κατάτμηση, τα οποία περιέχουν κυρίως πλήρως συνδεδεμένα επίπεδα (MLP) και στρώματα upsampling, αντίστοιχα.

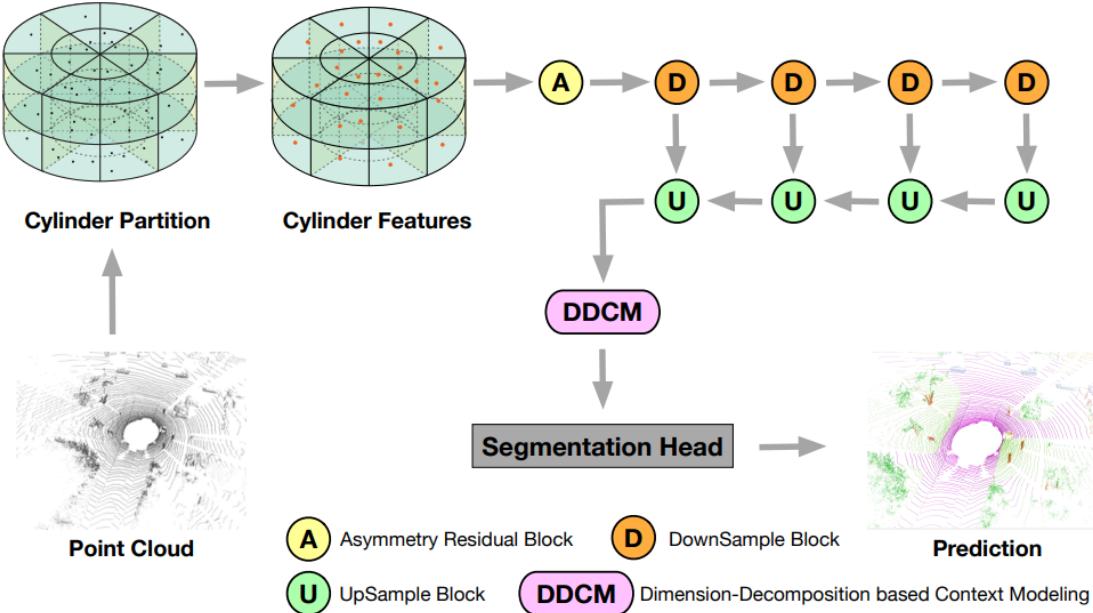
Το PointNet++ είναι μια καλή αρχιτεκτονική για την επεξεργασία συνόλων σημείων, προσφέροντας πολλά πλεονεκτήματα σε σχέση με τις παραδοσιακές μεθόδους. Η ιεραρχική του δομή του επιτρέπει να συλλαμβάνει τόσο λεπτομερή όσο και πιο γενικά χαρακτηριστικά, καθιστώντας το κατάλληλο για εργασίες όπως η ταξινόμηση αλλά και η σημασιολογική κατάτμηση. Επιπλέον, η δομή του διατηρεί την χωρική συσχέτιση μεταξύ των σημείων με αποτελέσματα οι προβλέψεις του να έχουν λάβει υπόψιν και αυτήν τη πληροφορία βελτιώνοντας την απόδοση του δικτύου.



Σχήμα 3.11: Αρχιτεκτονική του δικτύου PointNet

Cylinder3D

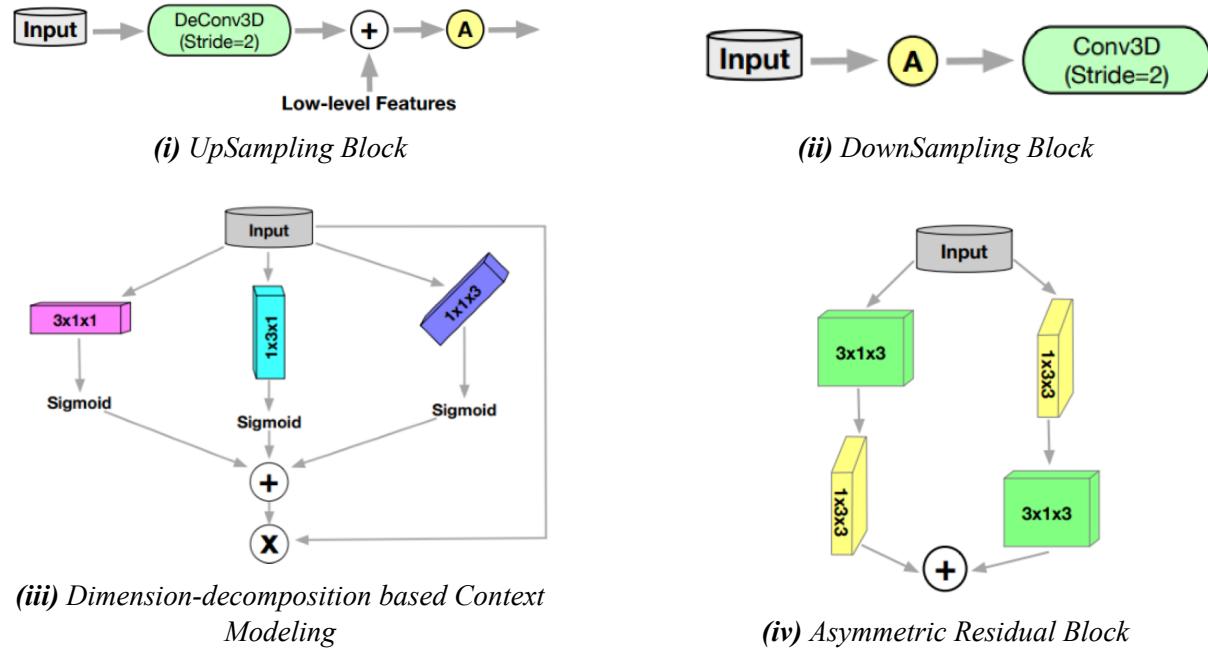
To Cylinder3D, το οποίο προτάθηκε μέσω της “*Cylinder3D: An Effective 3D Framework for Driving-scene LiDAR Semantic Segmentation*” [51], είναι ένα δίκτυο για τη τρισδιάστατη σημασιολογική κατάτμηση LIDAR σκηνής από αστικό περιβάλλον. Τα νέφη σημείων τέτοιων σκηνών έχουν ένα σημαντικό μειονέκτημα το οποίο έχει να κάνει με την διακύμανση της πυκνότητας σε διαφορετικά τμήματα της σκηνής. Για να λυθεί αυτό το πρόβλημα χρησιμοποιήθηκε κυλινδρική τμηματοποίηση του χώρου και με αυτόν τον τρόπο αυξήθηκε και η αποτελεσματικότητα και η ακρίβεια της κατάτμησης.



Σχήμα 3.12: Αρχιτεκτονική του δικτύου Cylinder3D

Η αρχιτεκτονική πέρα από την αρχική τμηματοποίηση του χώρου και την εξαγωγή χαρακτηριστικών για αυτόν τον χώρο αποτελείται από ένα συνδυασμό τεσσάρων κομματιών που θυμίζουν αρκετά την αρχιτεκτονική encoder-decoder. Πιο αναλυτικά, αρχικά χωρίζεται ο τρισδιάστατος χώρος σε μικρότερους με κύριο χαρακτηριστικό τα τμήματα που βρίσκονται πιο μακριά από τον αισθητήρα να έχουν μεγαλύτερο μέγεθος. Με αυτόν τον τρόπο λύνεται πρόβλημα της διακύμανσης της πυκνότητας. Έπειτα ακολουθεί η διαδικασία της εξαγωγής χαρακτηριστικών, η οποία γίνεται με τη χρήση ενός δικτύου βασισμένο στο pointnet, που παρουσιάστηκε

νωρίτερα. Καθώς όμως ο χώρος έχει χωριστεί με αυτόν τον τρόπο (κυκλικά τμήματα) είναι δύσκολο για το δίκτυο να αναγνωρίσει αντικείμενα τα οποία είναι επί των πλείστων πιο κυβοειδή. Για αυτό το λόγο χρησιμοποιήθηκε το “Asymmetric Residual Block” το οποίο βοηθάει σε ακριβώς αυτό το πράγμα. Στη συνέχεια ακολουθούν τμήματα τα οποία έχουν σχέση με “συμπίεση” και την μετέπειτα αποκαδικοποίηση των χαρακτηριστικών βασισμένα στην τρισδιάστατη συνέλιξη (η τρισδιάστατη συνέλιξη είναι ακριβώς η ίδια με την δισδιάστατη αλλά εφαρμόζεται σε τρισδιάστατα δεδομένα και ο πυρήνας της συνέλιξης είναι επίσης τρισδιάστατος). Τέλος ακολουθεί το “DDCM” το οποίο βοηθάει στην ποικιλότητα των αντικείμενων ίδιας κατηγορίας που υπάρχουν στη σκηνή και τη σωστή εξαγωγή γενικών χαρακτηριστικών.



Σχήμα 3.13: Αναλυτικά τα τμήματα του Cylinder3D

Εν κατακλείδι, το Cylinder3D έχει επιδείξει αρκετά καλή απόδοση σε εργασίες σημασιολογικής κατάτμησης σε αστικές σκηνές χωρίς κάποιο μεγάλο υπολογιστικό κόστος ενώ παράλληλα έχει επιτύχει κορυφαία αποτελέσματα σε αρκετά γνωστά σετ δεδομένων.

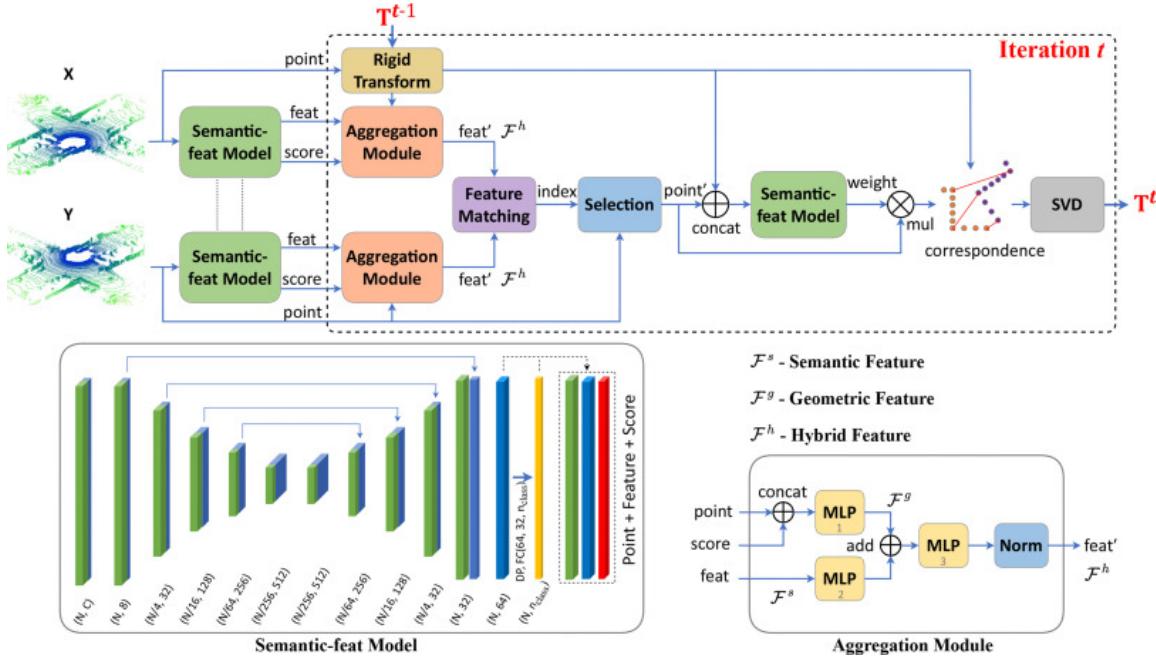
3.3 Ευθυγράμμιση Νέφους Σημείων

DeepSIR

Το DeepSIR είναι μια νέα μέθοδος που προτάθηκε μέσω της “DeepSIR: Deep semantic iterative registration for LiDAR point clouds” [52]. Αποτελεί μια επαναληπτική διαδικασία ευθυγράμμισης για τρισδιάστατα νέφη σημείων από πραγματικές σκηνές που παρουσιάζεται με λεπτομέρεια στο σχήμα 3.14.

Η αρχιτεκτονική του δικτύου βασίζεται σε ένα δίκτυο το οποίο είναι υπεύθυνο για τη σημασιολογική κατηγοριοποίηση του κάθε σημείου. Αυτό το δίκτυο παίρνει στην είσοδο του όλα τα σημεία του νέφους και παράγει δύο αποτελέσματα, το ένα έχει σχέση με την κατηγορία του κάθε σημείου και το άλλο με τα χαρακτηριστικά του κάθε σημείου που εξήγθησαν μέσω του δικτύου. Ακόμα μέσω αυτών των χαρακτηριστικών υπολογίζεται ένα ακόμα σκορ που έχει σχέση με τη γεωμετρία του χώρου που βρίσκεται το σημείο αλλά και την κατηγορία που ταξινομήθηκε

το σημείο, έχοντας υπόψιν τις κατηγορίες που παίζουν μεγαλύτερο ρόλο στην ευθυγράμμιση του νέφους με ένα άλλο.



Σχήμα 3.14: Αρχιτεκτονική της μεθόδου DEEPSIR

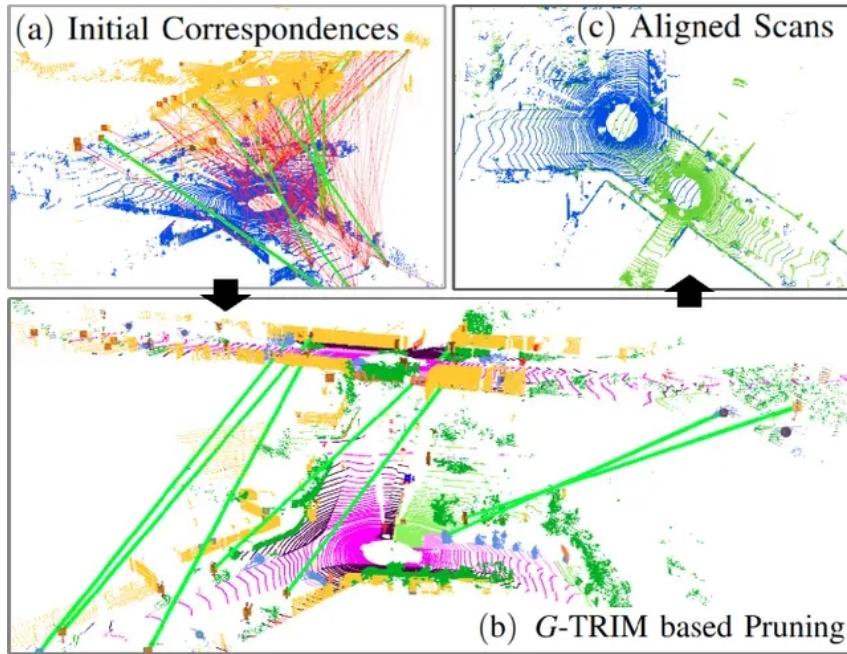
Τέλος, ακολουθεί το επαναληπτικό τμήμα της μεθόδου, το οποίο προβλέπει ένα μετασχηματισμό ανάμεσα στα δύο νέφη. Αυτό το κομμάτι αποτελείται μεταξύ άλλων από μια βασική μονάδα, η οποία είναι η “aggregation module”. Αυτό το τμήμα του αλγορίθμου αποτελείται κυρίως από δίκτυα MLP που εξυπηρετούν κυρίως το μετασχηματισμό της εξόδου το μοντέλου κατάτμησης σε μια κατάλληλη μορφή χρήσιμη για την αντιστοίχηση σημείων μεταξύ των δύο νεφών.

Συνολικά, το DeepSIR είναι μια πολλά υποσχόμενη νέα προσέγγιση για την ευθυγράμμιση νεφών σημείων, καθώς είναι μια γρήγορη και ακριβής μέθοδος που μπορεί να χρησιμοποιηθεί για ποικίλες εφαρμογές, όπως η αυτόνομη οδήγηση αλλά και η ρομποτική.

Segregator

Μια άλλη εξίσου νέα μέθοδος προτάθηκε μέσω της “*Segregator: Global Point Cloud Registration with Semantic and Geometric Cues*” [53]. Το Segregator είναι μια μέθοδος για καθολική ευθυγράμμιση σετ σημείων, η οποία αντί να εστιάζει αποκλειστικά στα χαρακτηριστικά των σημείων, εκμεταλλεύεται τόσο τις σημασιολογικές πληροφορίες όσο και τη γεωμετρική κατανομή για να δημιουργήσει αποτελεσματικά αντιστοιχίες ανθεκτικές στο θόρυβο και σε λάθος προβλέψεις. Αυτό

Αυτή η μέθοδος αρχικά κατηγοριοποιεί κάθε σημείο σε κάποια κατηγορία (όχημα, πεζός, κτίριο, ...) και στη συνέχεια σημεία της ίδιας κατηγορίας δημιουργούν ένα ξεχωριστό νέφος σημείων. Σε αυτό νέφος εφαρμόζονται τεχνικές ομαδοποιήσεις, ώστε να βρεθεί το κάθε αντικείμενο ξεχωριστά. Αυτό γίνεται με σκοπό να βρεθεί το κέντρο βάρους του κάθε αντικειμένου και να δημιουργηθεί μια αντιστοιχία όλων των αντικειμένων της ίδιας κατηγορίας από το ένα νέφος στο άλλο. Αυτή η διαδικασία βοηθά στη μείωση της πολυπλοκότητας του προβλήματος της ευθυγράμμισης και βελτιώνει την ανθεκτικότητα στο θόρυβο. Κάνοντας αυτό το πράγμα για όλες τις κατηγορίες που έχει νόημα η εύρεση του κέντρου βάρους δημιουργούνται



Σχήμα 3.15: Διαδικασία ενθυγράμμισης των προτάθηκε από το Segregator

κάποιες αντιστοιχίες μεταξύ των δυο νεφών (κόκκινες γραμμές στο σχήμα 3.15 α). Όμως αυτές οι αντιστοιχίες δεν είναι όλες σωστές, για να λυθεί αυτό το πρόβλημα χρησιμοποιήθηκε ο G-TRIM (Gaussian distribution-based Translation and Rotation Invariant Measurements), οποίος μοντελοποιεί κάθε σημείο ως μια κανονική κατανομή που εξαρτάται από τα γειτονικά σημεία. Με αυτό το τρόπο βρίσκονται ποιες είναι οι σωστές αντιστοιχίες, ο οποίες θα συμβάλουν στο υπολογισμό του τελικού μετασχηματισμού.

Κεφάλαιο 4

Υλοποίηση

Σε αυτήν την ενότητα θα περιγράφει ο διαδικασία που ακολουθήθηκε ώστε να γίνει η ευθυγράμμιση των δύο σετ σημείων, αλλά και θα αναλυθούν οι λόγοι που οδήγησαν σε κάποιες αποφάσεις, που είχαν σχέση με την υλοποίηση.

4.1 Dataset

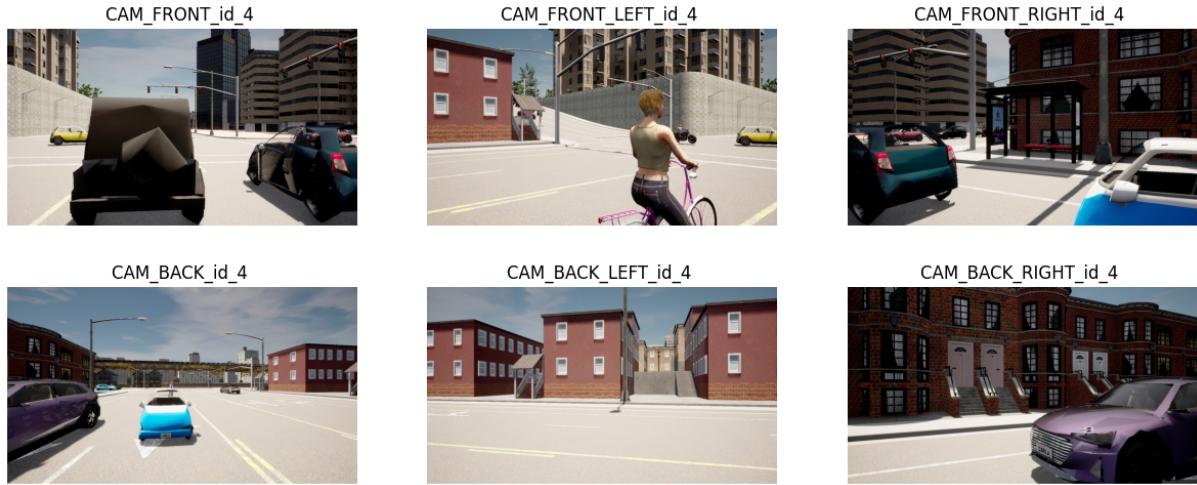
Η ανάπτυξη αυτόνομων και συνεργατικών οχημάτων είναι ένας ταχέως αναπτυσσόμενος τομέας με τη δυνατότητα να φέρει επανάσταση στις μεταφορές, για το λόγο χρειάζονται καλά σετ δεδομένων, τα οποία θα βοηθήσουν στη γρήγορη εξέλιξη αυτού του κλάδου. Ένα κάλο dataset παρέχει τα απαραίτητα δεδομένα (δεδομένα με τα αντίστοιχα σχόλια-annotations) για την εκπαίδευση και την αξιολόγηση αλγορίθμων για αντίληψη, πρόβλεψη και λήψη αποφάσεων.

Για την υλοποίηση αυτής της εργασίας χρειαζόμαστε ένα dataset το οποίο να περιέχει τα απαραίτητα δεδομένα από τα οχήματα της σκηνής για την ίδια σκηνή αλλά και για την ίδια χρονική στιγμή, ώστε τα αποτελέσματα να είναι όσο τον δυνατόν καλύτερα. Όμως, τα κλασικά dataset, όπως το KITTI και το nuScenes, συλλέγονται από ένα μόνο όχημα, πράγμα που μας οδηγεί στην επιλογή ενός άλλου σετ δεδομένων. Το γεγονός ότι χρειάζονται δεδομένα με τόσο αυστηρές προϋποθέσεις (ίδια χρονική στιγμή) οδήγησε του ερευνητές στη δημιουργία προσομοιώσεων, στις οποίες μπορούν να φτιάξουν οποίο σενάριο θέλουν αυτοί. Το πιο γνωστό εργαλείο για αυτό τον σκοπό είναι το CARLA [54], το οποίο έχει βοηθήσει στην ανάπτυξη πολλών datasets που περιέχουν δεδομένα από προσημειώσεις και όχι από πραγματικές καταστάσεις.

Το dataset που επιλέχθηκε είναι το V2X-Sim [55]. Αποτελεί ένα αρκετά γνωστό σετ δεδομένων για εφαρμογές συνδεδεμένων οχημάτων καθώς περιέχει ένα αρκετά μεγάλο πλήθος αισθητήρων μαζί με τα αντίστοιχα ground truth αποτελέσματα. Πιο αναλυτικά το κάθε όχημα είναι εξοπλισμένο με έξι έγχρωμες κάμερες (μπροστά/ μπροστά αριστερά/ μπροστά δεξιά/ πίσω/ πίσω αριστερά/ πίσω δεξιά), ένα αισθητήρα LIDAR (32-channels) για την καταγραφή της τρισδιάστατης πληροφορίας και μια μονάδα που καταγράφει τη θέση του οχήματος μαζί και με τις μεταβολές της ταχύτητας και της επιτάχυνσης. Ακόμα σε αυτό το dataset περιέχονται μάσκες σημασιολογικής κατάτμησης για όλες τις κάμερες μαζί με τις αντίστοιχες εικόνες βάθους.

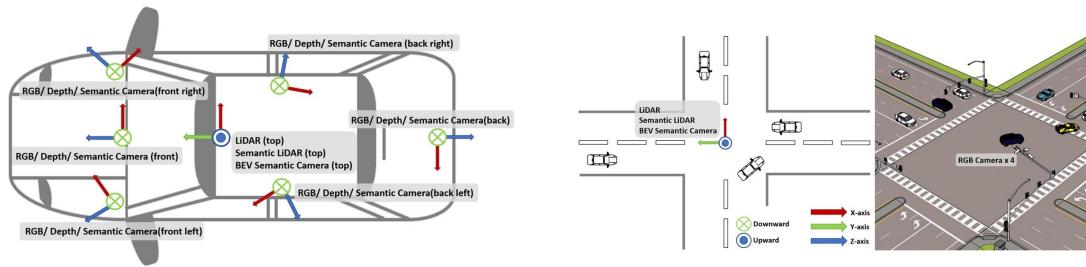
Το γεγονός ότι το συγκεκριμένο σετ δεδομένων αποτελείται από εικόνες προσομοίωσης κάνει τη χρήση των είδη εκπαίδευμένων μοντέλων αρκετά αναξιόπιστη, καθώς τα γραφικά και γενικότερα οι εικόνες των οχημάτων δεν είναι αρκετά ρεαλιστικές ώστε να επιτρέψουν τη χρήση έτοιμων δικτύων. Για αυτό το λόγο η ύπαρξη των σωστών αποτελεσμάτων κατάτμησης αποτελεί ένα μεγάλο πλεονέκτημα του dataset.

Τέλος, το συγκεκριμένο dataset παρέχει και δεδομένα από μια μονάδα που βρίσκεται σε



Σχήμα 4.1: Εικόνες από τις κάμερες ενός οχήματος

σταθερή θέση στο δρόμο (roadside unit - RSU) και παρακολουθεί την κίνηση των οχημάτων. Αν και δεν χρησιμοποιήθηκε αυτή η δυνατότητα του dataset, είναι αρκετά χρήσιμο να σημειωθεί για χρήση σε μελλοντικές επεκτάσεις.



Σχήμα 4.2: Η διάταξη των αισθητήρων στο όχημα και στο δρόμο για το Dataset V2X-Sim, αντίστοιχα

4.2 Επισκόπηση της υλοποίησης

Το πρόβλημα που προσπαθεί να λύσει αυτή η εργασία είναι η ευθυγράμμιση δύο διαφορετικών σετ τρισδιάστατων σημείων με ένα ποσοστό επικάλυψης (μεγάλο ή μικρο). Για να γίνει αυτό υλοποιήθηκε μια διαδικασία που αποτελείται από δύο βασικά τμήματα. Το πρώτο έχει να κάνει με την δημιουργία μιας αφηρημένης και γενικής αναπαράστασης της σκηνής που βλέπει το κάθε όχημα από την οπτική του και περιγράφεται με αρκετή λεπτομέρεια μέσω του σχήματος 4.13. Ένώ το δεύτερο τμήμα έχει σχέση με τελική ευθυγράμμιση και την αντιστοίχηση χαρακτηριστικών από τη μια αναπαράσταση του ενός οχήματος στην αναπαράσταση του άλλου οχήματος. Αυτή η διαδικασία περιγράφεται αναλυτικά μέσω του σχήματος 4.20.

4.2.1 Σημασιολογική Κατάτμησή του Νέφους Σημείων

Ένα από τα βασικά υποτομήματα για τη δημιουργία αυτής της αφηρημένης αναπαράστασης είναι σημασιολογική κατάτμηση του νέφους σημείων που έχει το κάθε όχημα μέσω του αισθητήρα LIDAR. Αυτό είναι κομμάτι είναι εξαιρετικά σημαντικό καθώς μέσω της σημασιολογικής πληροφορίας θα δομηθεί η σκηνή που βλέπει το κάθε όχημα.

Η σημασιολογική κατάτμηση του νέφους σημείων έγινε με παρόμοιο τρόπο με αυτόν που περιγράφηκε στην υποενότητα 3.2.1. Περιληπτικά χρησιμοποιήθηκαν οι εικόνες από τις κάμερες του οχήματος, ο οποίες έχουν μία πλήρη και αναλυτική περιγραφή για το περιβάλλοντα χώρο του οχήματος και όχι μόνο τα ίδια τα σημεία του νέφους. Αυτή η επιλογή βασίσθηκε στους εξής λόγους:

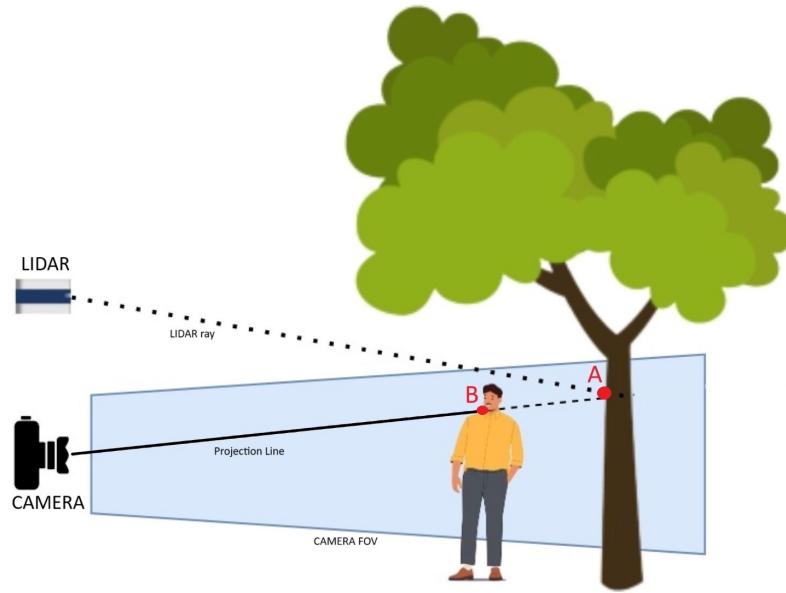
1. Οι περισσότεροι αλγόριθμοι που έχουν προταθεί για κατάτμηση νέφους σημείων δεν λειτουργούν αποτελεσματικά σε εξωτερικές συνθήκες ή σε αστικά περιβάλλοντα με σκηνές από αυτόνομα οχήματα, καθώς τα τρισδιάστατα σημεία έχουν ποικίλες πυκνότητες ανάλογα με τις αποστάσεις τους από τον αισθητήρα και θέτουν μεγάλες προκλήσεις στους αλγόριθμους.
2. Η κατάτμηση ενός νέφους σημείων χρησιμοποιώντας μόνο το ίδιο το νέφος δεν παρέχει την ίδια ποσότητα πληροφορίας με το να γινόταν χρήση και εικόνων του περιβάλλοντα χώρου. Για παράδειγμα, δεν είναι καθόλου εύκολο να διακριθεί μια κολόνα φωτισμού από ένα Φανάρι ή από μια κολόνα που περιέχει κάποιο σήμα οδήγησης χρησιμοποιώντας αποκλειστικά το νέφος σημείων, καθώς όλα αυτά τα αντικείμενα έχουν αρκετά παρόμοια γεωμετρία (πολύ λεπτά με σχετικά μεγάλο ύψος).

Αναλυτικά, αυτή η διαδικασία αποτελείται από δύο μέρη. Το πρώτο έχει να κάνει με τη σωστή προβολή ενός τρισδιάστατου σημείου πάνω σε ένα επίπεδο στην προκείμενη περίπτωση πάνω στο επίπεδο της εικόνας. Ενώ το δεύτερο μέρος έχει σχέση με την σημασιολογική κατάτμηση της εικόνας που λαμβάνει το όχημα από τις κάμερες και πάνω στις οποίες προβάλλονται τη σημεία του νέφους.

Προβολή των τρισδιάστατων σημείων πάνω στις εικόνες

Χρησιμοποιώντας την εξίσωση 2.20 προβάλλαμε όλα τα σημεία του νέφους πάνω στο επίπεδο της εκάστοτε εικόνας. Πρακτικά αυτή η εξίσωση μετασχηματίζει το τρισδιάστατο σημείο σε ένα δισδιάστατο, του οποίου οι συντεταγμένες αντιστοιχούν στις συντεταγμένες πάνω στο επίπεδο προβολής. Με αυτόν το τρόπο όμως προβάλλαμε και σημεία που είναι έκτος του οπτικού πεδίου της κάμερας πάνω στο επίπεδο προβολής. Για το λόγο αυτό ακολουθεί ένα βήμα στο οποίο φιλτράρουμε ποια δισδιάστατα σημεία έχουν σωστές συντεταγμένες. Σωστές συντεταγμένες θεωρούμε αυτές που δεν ξεπερνούν την ανάλυση της εικόνας με αποτέλεσμά να προβάλλονται εντός της και αυτές που βρίσκονται από μπροστά της. Σημαντικό είναι να διευκρινιστεί ότι η διαδικασία του φιλτραρίσματος γίνεται μετά την προβολή των σημείων. Αν και προσθέτει παραπάνω υπολογιστικό κόστος αυτός ο τρόπος λόγο του ότι προβάλλονται και μη χρήσιμα σημεία είναι αρκετά πιο εύκολο να διακρίνουμε εκ των υστέρων ποια ήταν τελικά τα σημεία εκείνα που βρίσκονταν όντος εντός του οπτικού πεδίου.

Από την άλλη πλευρά παρόλο που είναι εύκολος αυτός ο τρόπος κατάτμησης υπάρχει πιθανότητα να κάνει λάθος στην εκτίμηση της κατηγορίας ενός σημείου. Αυτή η περίπτωση φαίνεται στην εικόνα 4.3, όπου το σημείο “A” κατά τη διαδικασία της προβολής κρύβεται από τον άνθρωπο, με το αποτέλεσμα να ταξινομηθεί σε λάθος κατηγορία. Στην πραγματικότητα αυτό γίνεται καθώς οι δύο αισθητήρες (LIDAR, κάμερα) δεν έχουν το ίδιο οπτικό πεδίο και κατά συνέπεια υπάρχει πιθανότητα ένα σημείο του νέφους να μην έχει απευθείας οπτική επαφή με την κάμερα αλλά να παρεμβάλλεται ένα άλλο αντικείμενο. Αυτό το πρόβλημα εμφανίζεται και στην εικόνα 4.4 όπου κάποια σημεία που έχουν προβληθεί στο αριστερό μέρος του μαύρου οχήματος αντιστοιχούν στον δρόμο. Αυτό το καταλαβαίνουμε από το γεγονός ότι αυτά τα σημεία δεν έχουν το ίδιο χρώμα με την πλειοψηφία των άλλων σημείων που προβάλλονται στο όχημα.



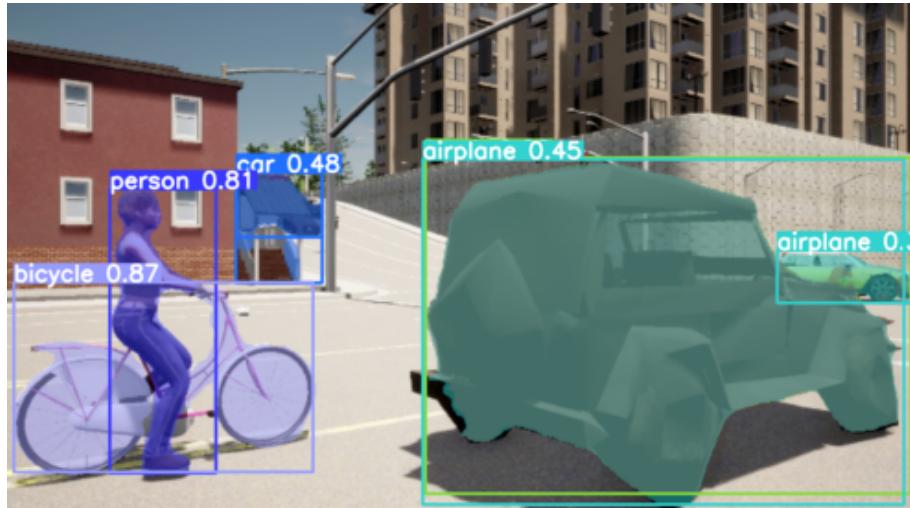
Σχήμα 4.3: Παράδειγμα λάθους κατηγοριοποίησης σημείου



Σχήμα 4.4: Προβολή των νέφους σημείων στο επίπεδο εικόνας για την μπροστινή κάμερα (το χρώμα κάθε σημείου αντιπροσωπεύει την απόσταση από την κάμερα)

Σημασιολογική κατάτμηση εικόνας

Στην ενότητα 3.1 περιγράφηκαν αρκετές αρχιτεκτονικές για τη σημασιολογική κατάτμηση εικόνας η κάθε μία με τη δικιά της προσέγγιση. Παρόλο που υπάρχουν αρκετά δίκτυα ήδη εκπαιδευμένα σε εφαρμογές σημασιολογικής κατάτμησης εικόνας σε αστικά περιβάλλοντα, η απευθείας χρήση τους δεν είχε τα προσδοκώμενα αποτελέσματα. Αυτό συμβαίνει καθώς αυτά τα δίκτυα έχουν εκπαιδευτεί σε αληθινές εικόνες και όχι σε εικόνες μέσα προσομοίωση με αποτέλεσμα κάποιες φορές να μην μπορούν να διακρίνουν το είδος ενός οχήματος ή να το κατηγοριοποιούν σε κάτι τέλειος άσχετο. Ένα τέτοιο παράδειγμα φαίνεται στην εικόνα 4.5, όπου χρησιμοποιήθηκε το δίκτυο YOLOv8 [56] εκπαιδευμένο στο σετ δεδομένων COCO. Σε αυτήν την εικόνα φαίνεται ξεκάθαρα ότι το δίκτυο παρόλο που έχει κάνει σωστά την κατάτμηση έχει χαρακτηρίσει το όχημα ως αεροπλάνο αν και είναι αρκετά ξεκάθαρη η κατηγορία του αντικείμενου.



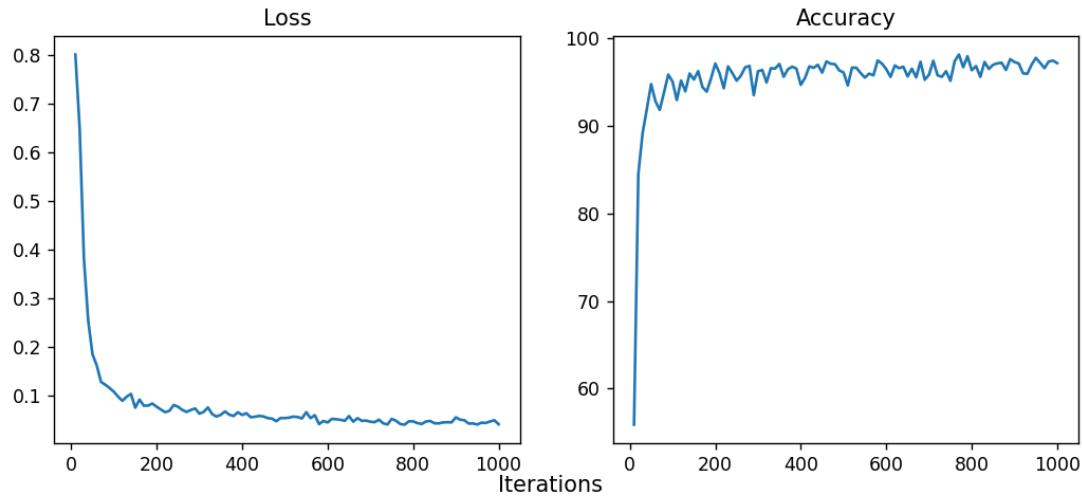
Σχήμα 4.5: Παράδειγμα λάθος κατηγοριοποίησης χρησιμοποιώντας προεκπαιδευμένα δίκτυα

Για αυτό το λόγο χρησιμοποιήθηκε η τεχνική **transfer learning**. To transfer learning είναι μια τεχνική μηχανικής εκμάθησης που επιτρέπει να επαναχρησιμοποιείται ένα μοντέλο που έχει εκπαιδευτεί σε ένα μεγάλο σύνολο δεδομένων για μια σχετική εργασία, ακόμα και αν η νέα εφαρμογή έχει διαφορετική κατανομή δεδομένων ή τελικό στόχο. Αυτό γίνεται με την επαναχρησιμοποίηση των βαρών του πρώτου μοντέλου ως σημείο εκκίνησης για την εκπαίδευση του δεύτερου μοντέλου. Αυτό μπορεί να βελτιώσει σημαντικά την απόδοση του δεύτερου μοντέλου, καθώς μπορεί να επικεντρωθεί στην εκμάθηση των συγκεκριμένων χαρακτηριστικών που σχετίζονται με τη νέα εφαρμογή και τα αντίστοιχα δεδομένα της, αντί να χρειάζεται να μάθει τα πάντα από την αρχή. Εκτός από την βελτιωμένη απόδοση αυτή η τεχνική μπορεί να μείωση σημαντικά το χρόνο που απαιτείται για την εκπαίδευση ενός νέου μοντέλου.

Από κάτω παρουσιάζονται οι γραφικές της εκπαίδευσης και τα αποτελέσματα κάποιων από των δικτύων που παρουσιάστηκαν στην υποενότητα 3.1. Η διαδικασία που ακολουθήθηκε είναι ίδια σε όλα τα δίκτυα, δηλαδή χρησιμοποιήθηκε το transfer learning με αρχικά βάρη να προκύπτουν μετά από εκπαίδευση στο σετ δεδομένων “CityScape” και η αναλογία των εικόνων στο training set με το test set παρέμεινε η ίδια. Το μόνο που άλλαζε σε κάθε εκπαίδευση είναι το μέγεθος του batch (πλήθος εικόνων εκπαίδευσης που επεξεργάζεται το δίκτυο πριν από την ενημέρωση των βαρών του) λόγω περιορισμών στο υλικό του υπολογιστή που χρησιμοποιήθηκε για την εκπαίδευση.

- **PSPNet**

Παρατηρώντας τις γραφικές παραστάσεις του PSPNet κατά τη διάρκεια της εκπαίδευσης βλέπουμε ότι το σφάλμα μειώνεται αρκετά γρήγορα (μέσα στις πρώτες 100 επαναλήψεις), πράγμα που οφείλετε στη μέθοδο του transfer learning και στην καλή αρχική τιμή των βαρών που προσφέρει. Ακόμα μέσω του πίνακα 4.1 παρατηρούμε ότι οι “σημαντικές” κατηγορίες, δηλαδή αυτές που θα χρησιμοποιήσουμε στη συνέχεια για τη δημιουργία της αναπαράστασης της σκηνής έχουν αρκετά μεγάλο ποσοστό ακρίβειας. Τέλος, είναι αρκετά σημαντικό να σχολιάσουμε το πόσο γρήγορα συγκλίνει το δίκτυο. Μέσα στις περίπου πρώτες 400 επαναλήψεις το δίκτυο έχει φτάσει σε ένα αρκετά καλό αποτέλεσμα και μετά από αυτές δεν μπορεί να βρει εύκολα καινούργια χαρακτηριστικά.



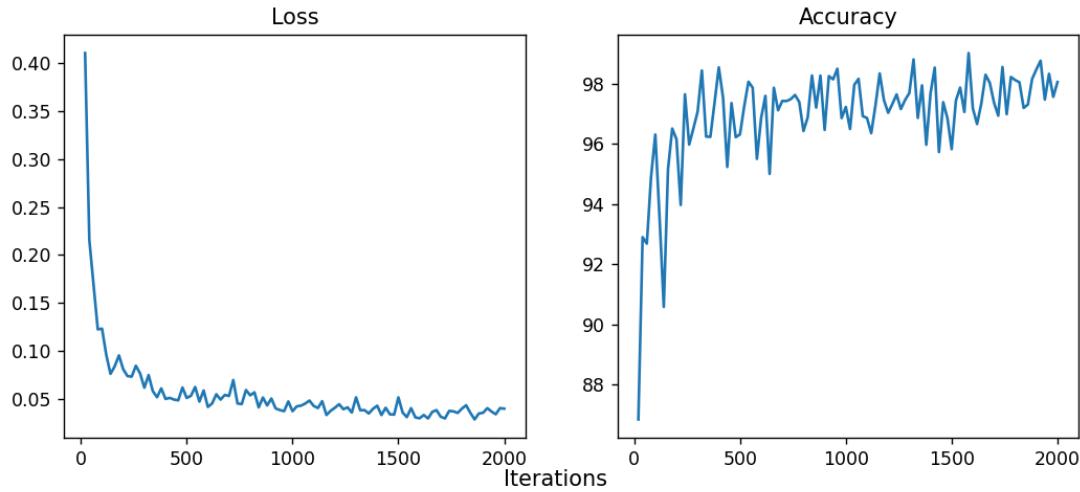
Σχήμα 4.6: Γραφικές εκπαίδευσης για το PSPNet

Class	IoU	Accuracy
<i>road</i>	93.25	99.32
<i>sidewalk</i>	77.55	83.25
<i>building</i>	93.38	96.93
<i>pole</i>	35.73	38.35
<i>traffic light</i>	10.80	11.08
<i>vegetation</i>	58.07	77.37
<i>sky</i>	93.80	97.12
<i>person</i>	62.76	69.60
<i>vehicle</i>	86.95	92.89
mean	68.03	73.99

Πίνακας 4.1: Αποτελέσματα κατάτμησης του δικτύου PSPNet στο test set

- **DeepLabv3+**

Συγκρίνοντας αυτό το δίκτυο με το προηγούμενο βλέπουμε ότι και τα δύο συνέκλιναν αρκετά γρήγορα, αλλά το DeepLabv3+ ξεκίνησε σε αρκετά καλύτερη κατάσταση συγκρίνοντάς τα μόνο από το ολικό σφάλμα άλλα και την ακρίβεια των αποτελεσμάτων. Ακόμα αξίζει να σχολιαστεί ότι το DeepLabv3+ έχει περίπου 7% καλύτερα αποτελέσματα βλέποντας τον πίνακα 4.2 με τον αντίστοιχο του πρώτου δικτύου, πράγμα που οφείλετε κυρίως στην αρχιτεκτονική του αλλά και στη χρήση της atrous convolution.



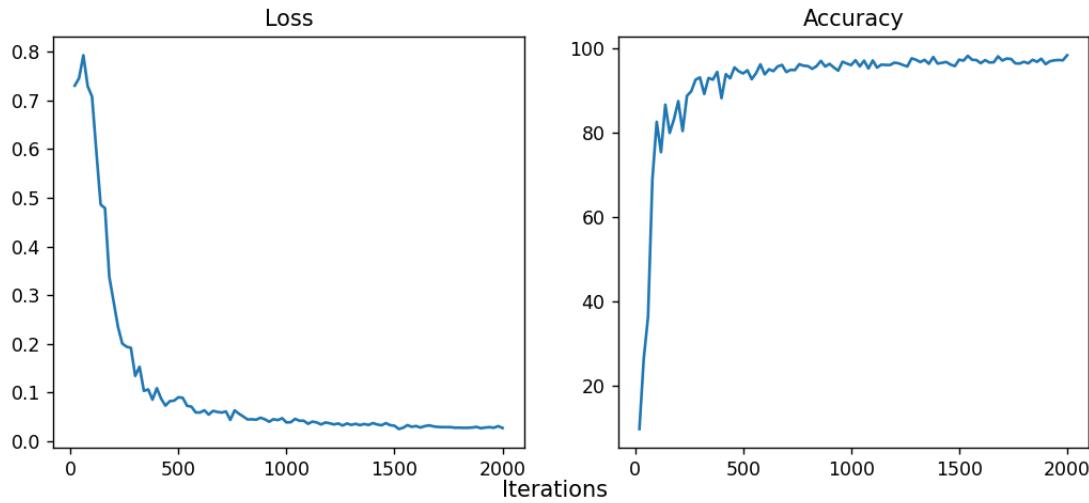
Σχήμα 4.7: Γραφικές εκπαίδευσης για το DeepLabv3+

Class	IoU	Accuracy
<i>road</i>	95.73	98.84
<i>sidewalk</i>	84.50	87.10
<i>building</i>	95.27	97.43
<i>pole</i>	55.83	61.28
<i>traffic light</i>	45.29	49.96
<i>vegetation</i>	60.10	85.57
<i>sky</i>	95.66	98.93
<i>person</i>	67.88	83.72
<i>vehicle</i>	91.24	96.81
mean	76.83	86.18

Πίνακας 4.2: Αποτελέσματα κατάτμησης του δικτύου DeepLabv3+ στο test set

- **SegFormer**

Τέλος εκπαιδεύτηκε και ένα δίκτυο βασισμένο στους visual transformers. Αν και τα δίκτυα που χρησιμοποιούν transformers έχουν αρκετά καλύτερα αποτελέσματα από τα αυτά που κάνουν χρήση κάποιου είδους συνέλιξης, στην προκείμενη περίπτωση δεν ισχύει αυτό. Ο πιο πιθανός λόγος είναι το ότι το σετ δεδομένων αποτελείται από περίπου 3000 εικόνες, που είναι αρκετά μικρός αριθμός για τα δεδομένα τέτοιων δικτύων. Το γεγονός όμως ότι χρησιμοποιήθηκε η τεχνική transfer learning, εξαλείφει εν μέρη αυτό το πρόβλημα κάνοντάς το συγκρίσιμο με τις άλλα δίκτυα. Ακόμα αξίζει να σημειωθεί ότι το συγκεκριμένο δίκτυο συγκλίνει και αυτό αρκετά γρήγορα σε μια σταθερή κατάσταση αλλά οι επαλείψεις που χρειάζονται είναι περίπου οι διπλάσιες σε σχέση με τα άλλα δίκτυα, που χρησιμοποιούν συνέλιξη. Αυτή η παρατήρηση επιβεβαιώνει το γεγονός ότι τέτοια δίκτυα χρειάζονται αρκετά δεδομένα κατά τη διάρκεια της εκπαίδευσης.



Σχήμα 4.8: Γραφικές εκπαίδευσης για το SegFormer

Class	IoU	Accuracy
<i>road</i>	92.96	98.89
<i>sidewalk</i>	74.93	80.84
<i>building</i>	94.22	96.79
<i>pole</i>	46.66	55.65
<i>traffic light</i>	28.52	31.43
<i>vegetation</i>	61.20	79.98
<i>sky</i>	95.59	98.36
<i>person</i>	63.89	71.72
<i>vehicle</i>	88.16	95.35
mean	71.79	79.11

Πίνακας 4.3: Αποτελέσματα κατάτμησης των δικτύων SegFormer στο test set

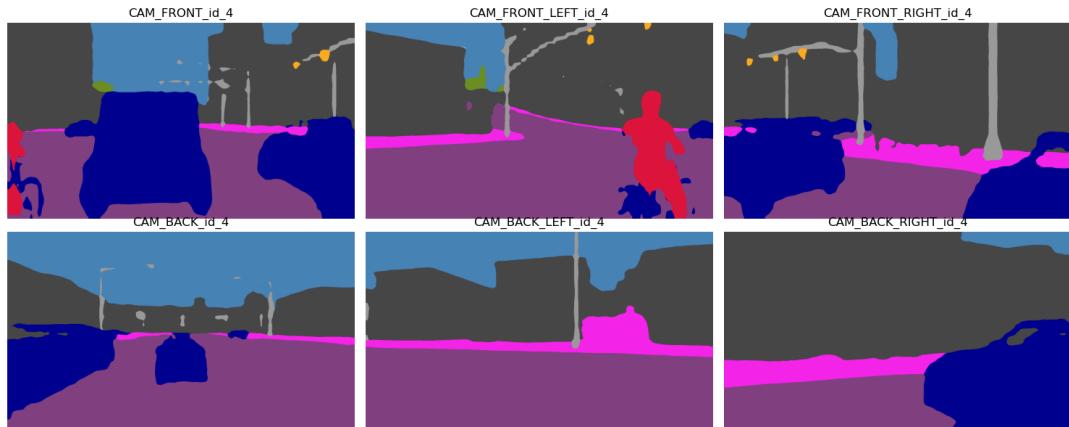
Συμπερασματικά, παρατηρώντας όλα τα δίκτυα και τα αποτελέσματά τους συγκρινόμενα με αυτά των αντίστοιχων δικτύων που χρησιμοποιήθηκαν για την αρχικοποίηση των βαρών τους μέσω του πίνακα 4.4 είναι ξεκάθαρο ότι ναι μεν απέχουν αλλά σε κάποιες περιπτώσεις όχι τόσο πολύ. Ακόμα μέσω του ίδιου πίνακα (4.4) παρατηρούμε ότι τα δίκτυα μετά την επιπλέον διαδικασία της εκπαίδευσης δεν έχουν και μεγάλες διαφορές ως προς την απόδοσή τους και τα αποτέλεσμα που παράγουν δεν έχουν ουσιώδεις διαφορές. Συνολικά, τα αποτελέσματα της σύγκρισης των αποδόσεων των δικτύων δείχνουν ότι η επιπλέον εκπαίδευση οδήγησε σε βελτίωση της απόδοσης των δικτύων συγκρίνοντάς τα με το αν χρησιμοποιόντουσαν τα δίκτυα χωρίς επιπλέον προσαρμογή στα νέα δεδομένα. Ωστόσο, η βελτίωση αυτή δεν είναι πάντα σημαντική και εξαρτάται από διάφορους παράγοντες.

Ο πιο σημαντικός είναι το μέγεθος και η ποιότητα των δεδομένων εκπαίδευσης. Το συγκεκριμένο σετ δεδομένων αποτελείται από περίπου 3000 εικόνες με τις αντίστοιχες μάσκες κατάτμησης. Μπορεί να φαίνονται αρκετές, καθώς χρησιμοποιήθηκε η μέθοδος του transfer learning, αλλά αν σκεφτούμε ότι το σετ είναι φτιαγμένο για οχήματα που βρίσκονται σε διασταυρώσεις, που εκεί αναγκαστικά κάποια οχήματα είναι ακίνητα για κάποιο χρονικό διάστημα, το πραγματικό πλήθος εικόνων μικραίνει αρκετά.

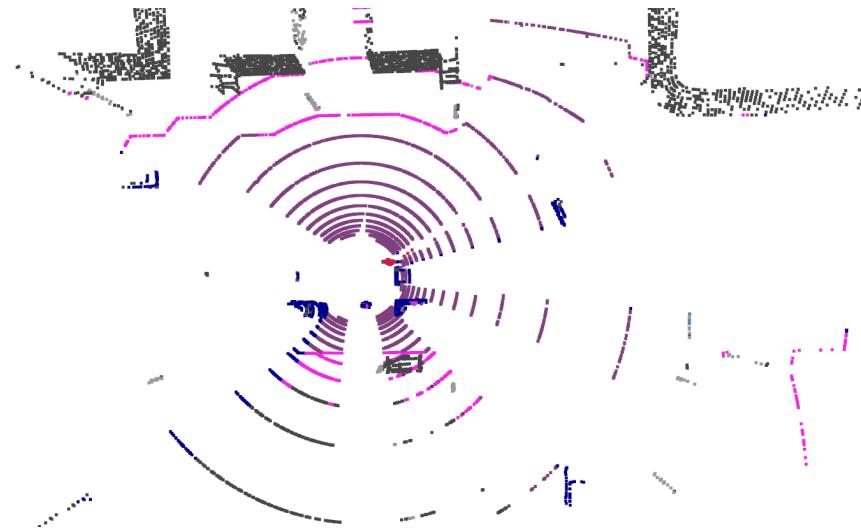
Network	mIoU	
	Initial Network	Μετά το Transfer Learning
PSPNet (R-50-D8)	77.85	68.03
DeepLabv3+ (R-50-D8)	79.61	76.83
SegFormer (MIT-B0)	76.54	71.79

Πίνακας 4.4: Σύγκριση των αποτελεσμάτων ανάμεσα στα δίκτυα που χρησιμοποιήθηκαν για αρχικοποίηση των βαρών και αυτών μετά το transfer learning

Στη συνέχεια παρουσιάζονται τα αποτελέσματα της κατάτμησης των εικόνων από ένα όχημα 4.9 και το αντίστοιχο αποτέλεσμα κατάτμησης του νέφους σημείων 4.10.



Σχήμα 4.9: Οι μάσκες κατάτμησης για όλες τις εικόνες ενός οχήματος που προέκυψαν μετά την επιπλέον εκπαίδευση του DeepLabv3+



Σχήμα 4.10: Κατάτμηση νέφους σημείων χρησιμοποιώντας τις μάσκες κατάτμησης από τις κάμερες

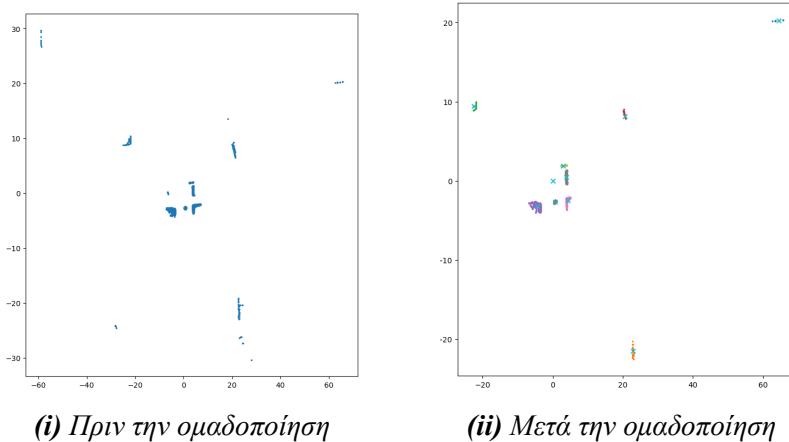
4.2.2 Αναπαράσταση της Σκηνής

Έχοντας δώσει σε όλα τα τρισδιάστατα σημεία του νέφους μία κατηγορία (label), απομένει το τελικό στάδιο. Αυτό το στάδιο είναι η δημιουργία του γράφου σκηνής, ο οποίος είναι μία

δομή για την αναπαράσταση της σκηνής με τρόπο τέτοιο ώστε να διευκολύνεται η αντιστοίχηση χαρακτηριστικών από την μια οπτική στην άλλη.

Για τη δημιουργία του γράφου πρώτα επιλέγουμε ποιες κατηγορίες μπορούν να αποτελούν κόμβους του γράφου. Προϋπόθεση για να θεωρείτε μια κατηγορία κατάλληλη πρέπει η θέση ενός αντικείμενο αυτής της κατηγορίας να μπορεί να περιγράφει από μόνο ένα τρισδιάστατο σημείο, τέτοιες κατηγορίες είναι οι άνθρωποι, τα οχήματα, οι κολώνες και όχι τα κτήρια ή το πεζοδρόμιο. Έχοντας επιλέξει τις κατηγορίες από τις οποίες θα εξαχθούν οι κόμβοι του γράφου ακολουθεί το βήμα για την εύρεση των ίδιων των κόμβων. Πρακτικά αυτό το πρόβλημα είναι πρόβλημα ομαδοποίησης (clustering), καθώς έχοντας απομονώσει τα σημεία που αντιστοιχούν στην εκάστοτε κατηγορία πρέπει να τα ομαδοποιήσουμε με τρόπο τέτοιο ώστε κάθε ξεχωριστή ομάδα να αντιστοιχεί σε ένα αντικείμενο της κατηγορίας. Αυτή η διαδικασία είναι εξαιρετικά σημαντική για το σωστό τελικό αποτέλεσμα, καθώς ο παραγόμενος γράφος αποτελεί τη βάση για την τελική ευθυγράμμιση και για αυτό το λόγο πρέπει να επιλέγει ο κατάλληλος αλγόριθμος ομαδοποίησης και να ρυθμιστούν κατάλληλα οι εσωτερικές του παράμετροι.

Στο αριστερό σχήμα της εικόνας 4.11 φαίνονται όλα τα σημεία που έχουν κατηγοριοποιηθεί ως οχήματα μετά από ένα φίλτρο άρισμα για να μειωθούν οι επιπτώσεις του προβλήματος που παρουσιάστηκε μέσω της εικόνας 4.3. Ενώ στο δεξιό φαίνεται το αποτέλεσμα της ομαδοποίησης. Όπως παρατηρούμε τα κέντρα των οχημάτων που υπολογίστηκαν είναι τα κέντρα της κάθε ομάδας σημείων και όχι το πραγματικό κέντρο του κάθε οχήματος. Αυτό ενδεχομένως θα μπορούσε να δημιουργήσει πρόβλημα σε σενάρια όπου το ίδιο όχημα παρατηρείται από διαφορετικές πλευρές, αλλά μέσα από την κατάλληλη παραμετροποίηση της εξίσωσης 4.1 το πρόβλημα αυτό μειώνεται αρκετά. Πιο αναλυτικά η επιλογή της κατάλληλης παραμέτρου περιγράφεται στην υποενότητα 4.2.3. Λόγω ανεπάρκειας στις γωνίες γύρω από τους άξονες X και Y,

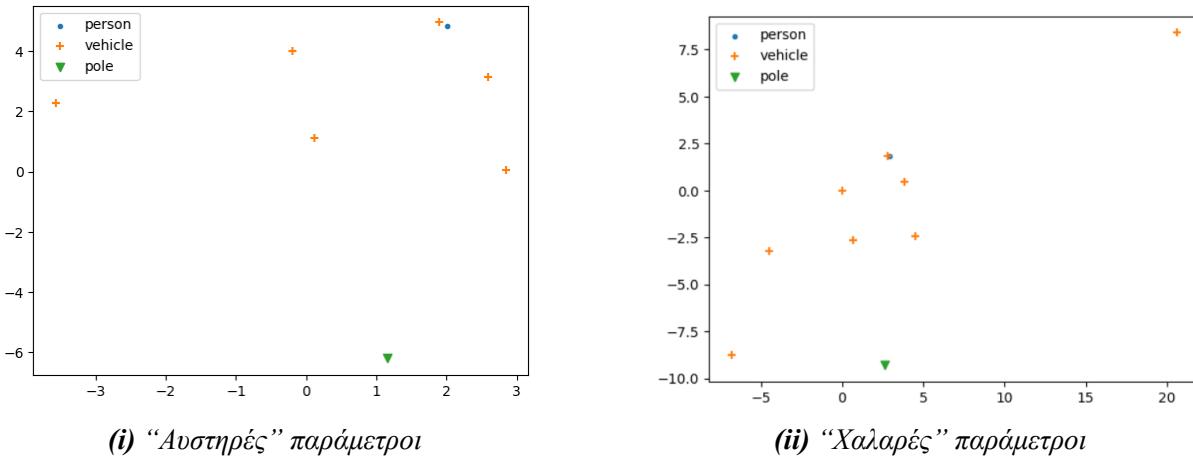


Σχήμα 4.11: Παράδειγμα ομαδοποίησης σε σημεία κατηγοριοποιημένα ως οχήματα

εισήχθη το τελευταίο στάδιο ευθυγράμμισης, περιλαμβάνοντας την εφαρμογή του προηγμένου αλγορίθμου ICP. Αυτός ο αλγόριθμος απαιτεί την καθορισμένη μέγιστη απόσταση (κατώφλι), η οποία μπορεί να αντιληφθεί ως ακτίνα κύκλου. Εντός αυτής της ακτίνας, εκτελείται η αναζήτηση για τον κατάλληλο γείτονα ενός σημείου στον χώρο του pointcloud, σε σχέση με το αντίστοιχο σημείο σε άλλο pointcloud.

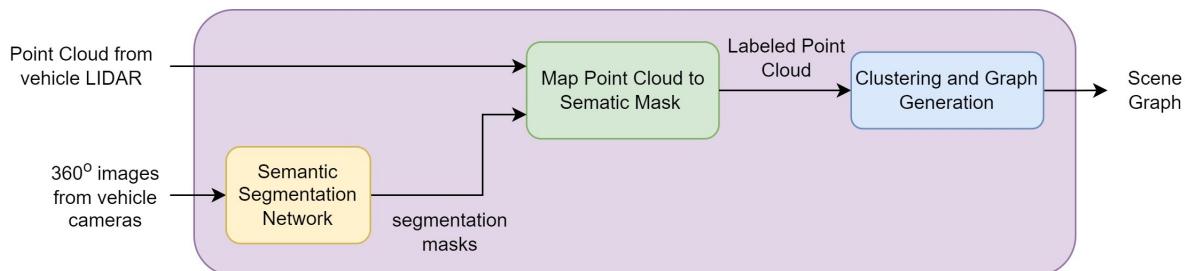
Ο πιο κατάλληλος αλγόριθμος είναι ο HDBSCAN (περιγράφηκε στην υποενότητα 2.6.2), διότι τα σημεία του νέφους αλλάζουν πυκνότητα καθώς απομακρύνονται από τον αισθητήρα και άρα ο κλασικός DBSCAN δεν θα λειτουργούσε όπως θα θέλαμε. Όμως με τη χρήση αυτής της μεθόδου (HDBSCAN) παρουσιάστηκε ένα μειονέκτημα το οποίο είναι όταν οι παράμετροι είναι πολύ “αυστηρές”, για παράδειγμα υψηλό ελάχιστο μέγεθος συνόλου, ο αλγόριθμος θα

προσπαθήσει να βρει πυκνές συστάδες και ενδέχεται να αγνοήσει μικρά σύνολα δεδομένων που δεν πληρούν αυτά τα κριτήρια. Αντίθετα, όταν οι παράμετροι είναι πιο “χαλαρές”, είναι πιθανόν να περιλαμβάνονται περισσότερα σημεία σε συστάδες, ακόμα και αν δεν ανήκουν πραγματικά σε κάποια συστάδα, και μπορεί να θεωρηθούν ως θόρυβος. Αυτό το πρόβλημα παρουσιάζεται στο σχήμα 4.12, όπου στο πρώτο σχήμα δεν έχει βρεθεί η πληροφορία που βρίσκεται μακριά από τον αισθητήρα λόγω του ότι εκεί τα σημεία είναι πιο αραιά και δεν υπάρχουν αρκετά στα μακρινά αντικείμενα, αλλά ότι αντικείμενο έχει ανιχνευθεί αντιστοιχεί σε ένα πραγματικό. Ενώ στο διπλανό σχήμα παρόλο που έχει βαρεθεί το μακρινό όχημα που βρίσκεται περίπου στο (20, 7.5) έχει ανιχνευθεί και ένα λάθος όχημα στο (-5, -7.5), το οποίο δεν αντιστοιχεί σε πραγματικό.



Σχήμα 4.12: Σύγκριση παραμέτρων για το αλγόριθμο HDBSCAN και οπτικοποίηση των αποτελεσμάτων

Τέλος ενώνονται όλοι οι κόμβοι ανεξαρτήτως κατηγορίας με όλους τους άλλους και για βάρη των ακμών αυτών επιλέχτηκε η ευκλείδεια απόσταση μεταξύ των κόμβων που ενώνει η εκάστοτε ακμή. Η ευκλείδεια απόσταση είναι κατάλληλη, καθώς αυτό που έχουμε κωδικοποίηση μέσω των κόμβων είναι η θέση των αντίστοιχων αντικείμενων στο τρισδιάστατο περιβάλλον. Επιπλέον με αυτήν την συνδεσμολογία των κόμβων σιγουρεύουμε ότι θα υπάρχουν ακμές που θα χρησιμεύουν στην αντιστοίχηση ακόμα και αν κάποιος κόμβος λείπει από την το γράφο του άλλου οχήματος. Ακολουθεί η εικόνα 4.13 με όλη τη διαδικασία για τη δημιουργία του γράφου σκηνής.



Σχήμα 4.13: Διαδικασία για τη δημιουργία του γράφου σκηνής που βλέπει το όχημα

4.2.3 Ευθυγράμμιση Νέφους Σημείων

Εύρεση αρχικής αντιστοίχησης

Έχοντάς δημιουργήσει το γράφο για τα οχήματα που εμπλέκονται στη σκηνή μένει να υλοποιηθεί το τελευταίο και πιο βασικό κομμάτι που είναι και η αντιστοίχηση αυτών. Αυτή η διαδικασία γίνεται ανά δύο οχήματα και περιλαμβάνει την αντιστοίχηση κάθε κόμβου του γράφου σκηνής του πρώτου οχήματος με έναν κόμβο του γράφου του άλλου οχήματος και περιγράφεται περιληπτικά στην υποενότητα 2.7.1.

Ένα από τα καίρια κομμάτια για τη σωστή αντιστοίχηση είναι δημιουργία ενός καλού πίνακα αντιστοίχησης. Ο πίνακας αυτός πρέπει να κωδικοποιεί κατάλληλα τις ομοιότητες μεταξύ των κόμβων αλλά και των ακμών. Στην προκείμενη περίπτωση καθώς οι κόμβοι συμβολίζουν την κατηγορία του αντίστοιχου χαρακτηριστικού μπορούν να θεωρηθούν ως ένα διάνυσμα που έχει σε όλες τις θέσεις του “0” εκτός από εκείνη που αντιστοιχίζεται στην κλάση που βρίσκεται. Με αυτόν τον τρόπο το εσωτερικό γινόμενο δυο κόμβων που είναι στην ίδια κατηγορία θα δώσει αποτέλεσμα “1” ενώ διαφορετικά “0”. Σε αντίθετη περίπτωσή με τις ακμές οι οποίες συμβολίζουν την ευκλείδεια απόσταση μεταξύ των κόμβων δεν μπορεί να χρησιμοποιηθεί η προηγούμενή υλοποίηση. Αυτό γίνεται καθώς θα έδιναν το ίδιο αποτέλεσμα ένα ζεύγος ακμών με μεγάλη ομοιότητα με ένα που δεν μοιάζει καθόλου. Για παράδειγμα το ζεύγος ακμών με μεγέθη {3,3} θα είχε το ίδιο αποτέλεσμα με το ζεύγος ακμών {1,9}, οι οποίες έχουν μεγάλη διαφορά. Για αυτό το λόγο χρησιμοποιήθηκε μία άλλη συνάρτηση ομοιότητας, η οποία φαίνεται παρακάτω (εξίσωση 4.1). Χρησιμοποιώντας αυτήν την εξίσωση πλέον η τέλεια ομοιότητα αντιστοιχίζεται στο “1” και καθώς αυξάνεται η διαφορά τόσο πιο πολύ τείνει αυτός ο παράγοντας στο “0” που συμβολίζει τη μηδενική ομοιότητα.

$$\exp \left(-\frac{(\mathbf{f}_1 - \mathbf{f}_2)^2}{\sigma} \right) \quad (4.1)$$

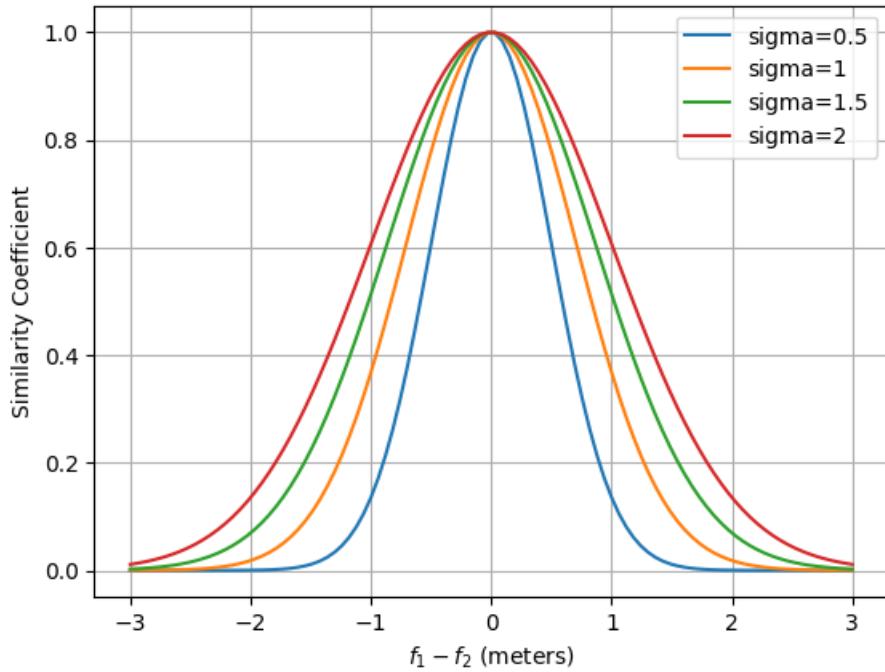
όπου:

\mathbf{f}_i : διάνυσμα χαρακτηριστικών

σ : παράμετρος απόκλισης, συνήθως είναι 1

Χρησιμοποιώντας την εξίσωση 4.1 εκτός από την κανονικοποίηση των αποτελεσμάτων στο διάστημα [0, 1], που είναι απαραίτητη ώστε να συμβαδίζουν οι συντελεστές ομοιότητας των ακμών με αυτές των κόμβων, πετυχαίνουμε και μία μη γραμμικότητα που είναι αρκετά χρήσιμη για το τελικό αποτέλεσμα. Καθώς οι ακμές των γράφων αντιπροσωπεύουν την ευκλείδεια απόσταση μεταξύ των αντίστοιχων κόμβων, είναι λογικό ότι κόμβοι που έχουν μικρή απόσταση είναι αρκετά πιο σημαντικοί για την τελική ευθυγράμμιση από του κόμβους που βρίσκονται μακριά. Άρα με αυτόν τον τρόπο δίνουμε μεγαλύτερη βαρύτητα στις ακμές που έχουν μικρές διαφορές στα βάρη τους, διότι ενδεχομένως αυτές αποτελούν και σωστές αντιστοιχίσεις.

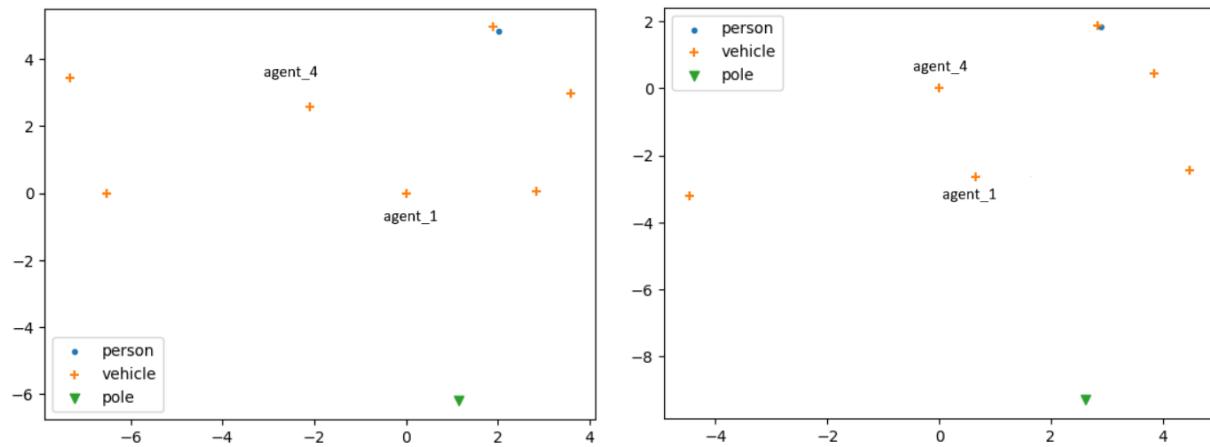
Καθώς η εξίσωση 4.1 αποτελεί τη βάση για την αντιστοίχηση των γράφων και κατά συνέπεια για το τελικό αποτέλεσμα της ευθυγράμμισης, πρέπει να επιλέγουν οι κατάλληλοι παράμετροι ώστε να έχουμε το βέλτιστο αποτέλεσμα. Πιο αναλυτικά αυτή η εξίσωση παίρνει σαν παράμετρο το “ σ ”, το οποίο είναι υπεύθυνο για το ποιες αποστάσεις πρέπει να θεωρούνται ότι είναι όμοιες και ποιες όχι. Μέσα από το σχήμα 4.14 βλέπουμε ότι για μικρές τιμές της παραμέτρου έχουμε και μικρό εύρος για τη διαφορές των βαρών ώστε να θεωρούνται όμοιες. Από τη στιγμή που οι κόμβοι των οχημάτων που αντιπροσωπεύουν μεγαλύτερο χώρο ο κάθε ένας σε σχέση με τις άλλες κατηγορίες αλλά και δεδομένου ότι είναι πιο πολλοί, βάση αυτών θα καθοριστεί η τιμή αυτής της παραμέτρου. Καθώς μπορεί να έχουμε το ίδιο όχημα από διαφορετικές οπτικές υπάρχει η περίπτωση να εμφανίζει μια διακύμανση η θέση του ανάλογα από ποια πλευρά το βλέπουμε. Αυτή η διάφορα στη θέση σπάνια μπορεί να ξεπεράσει τα δύο μέτρα και βάση



Σχήμα 4.14: Συνάρτηση ομοιότητας 4.1 για διάφορες τιμές της παραμέτρου “σ”

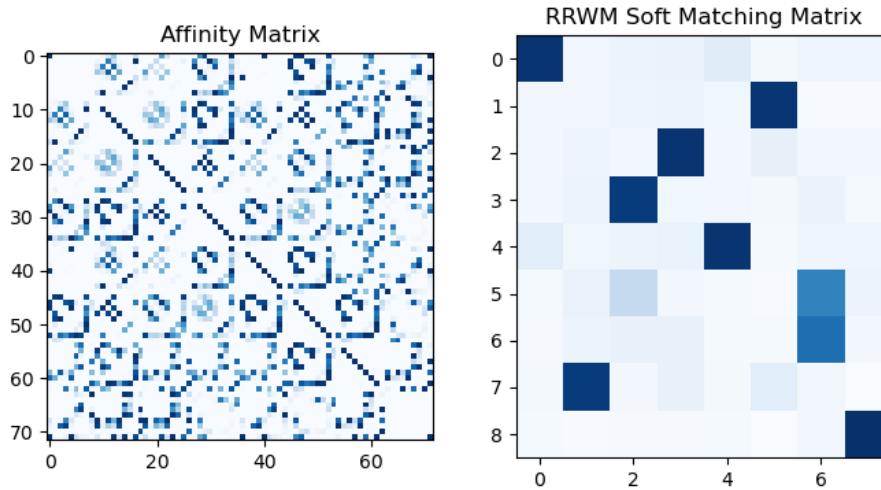
αυτού επιλέχτηκε η τιμή της παραμέτρου, χωρίς αυτό να σημάνει ότι άλλες τιμές κοντινές στην θα δώσουν λάθος αποτελέσματα.

Στην εικόνα 4.15 φαίνονται οι γράφοι σκηνής για δύο οχήματα που βρίσκονται το ένα δίπλα στο άλλο για την ίδια χρονική στιγμή. Έχοντας σαν δεδομένο αυτές τις αναπαραστάσεις προέκυψε ο πίνακας ομοιότητας που φαίνεται στην εικόνα 4.16 μαζί με το αποτέλεσμα του αλγορίθμου αντίστοιχησης. Ενώ στην εικόνα 4.17 φαίνεται η οπτικοποίηση του αποτελέσματος της αντίστοιχης.

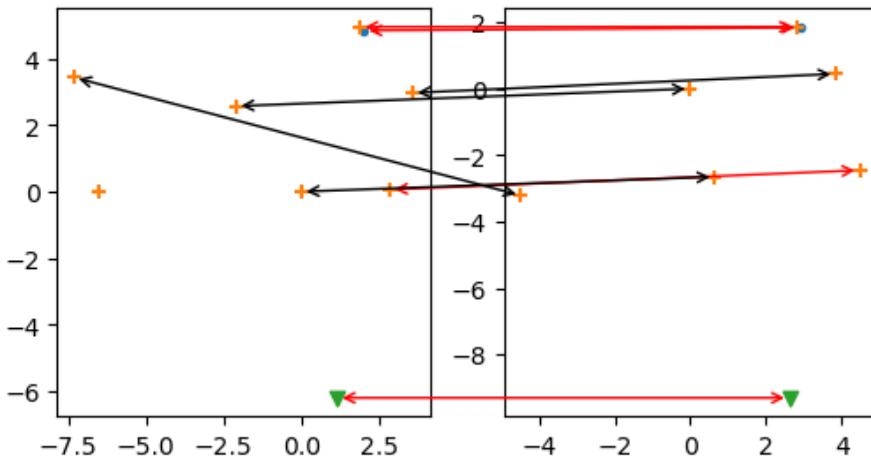


Σχήμα 4.15: Οι γράφοι σκηνής για δύο οχήματα για την ίδια χρονική στιγμή

Έχοντας ολοκληρώσει τη διαδικασία της αντίστοιχης των κόμβων ακολουθεί το κομμάτι της εύρεσης του μετασχηματισμού που μετατρέπει το ένα σύστημα συντεταγμένων στο άλλο. Καθώς ο μετασχηματισμός που γίνεται είναι rigid, δηλαδή περιλαμβάνει μόνο μετατοπίσεις και περιστροφές αρκούν μόνο τρεις αντίστοιχεις για να βρεθεί ο αντίστοιχος πίνακας μετασχημα-



Σχήμα 4.16: Πίνακας ομοιότητας και το αντίστοιχο αποτέλεσμα του αλγορίθμου RRWM



Σχήμα 4.17: Οπτικοποίηση της αντιστοιχησης

τισμού. Όμως καθώς κάποιες αντιστοιχίσεις είναι πιθανό να είναι λανθασμένες δεν μπορούμε να λάβουμε όλες υπόψιν και για αυτό το λόγο διαλέγουμε τις τρεις κορυφαίες για τις οποίες είμαστε πιο σίγουροι ότι είναι σωστές και αυτές συμβολίζονται με κόκκινο χρώμα στο σχήμα 4.17.

Η διαδικασία εύρεσης του πίνακα μετασχηματισμού γίνεται ακολουθώντας τα από κάτω βήματα:

1. Υπολογισμός το κέντρου βάρους κάθε σετ σημείων
2. Μετακινούμε κάθε σετ ώστε το κέντρο βάρους του να είναι στην αρχή των αξόνων. Με αυτόν το τρόπο αφαιρούμε την συνιστώσα της μετατόπισης και ασχολούμαστε μόνο με την περιστροφή.
3. Υπολογισμός του πίνακα συνδιακύμανσης μεταξύ των μετατοπισμένων σετ (Cross-Conv Matrix) μέσω της εξίσωσης 4.2

$$H = (A - centroid_A)(B - centroid_B)^T \quad (4.2)$$

όπου:

A, B : Η λίστα με τα τρισδιάστατα σημεία κάθε σετ

4. Ακολουθεί η ανάλυση των πίνακα του προηγούμενο βήματος σε σε ιδιάζουσες τιμές, δηλαδή $[U, S, V] = SVD(H)$, και υπολογισμός της τιμής $d = sign(\det(VU^T))$.
5. Υπολογισμός της κατάλληλης περιστροφής των δύο σετ μέσω της εξίσωσης 4.3

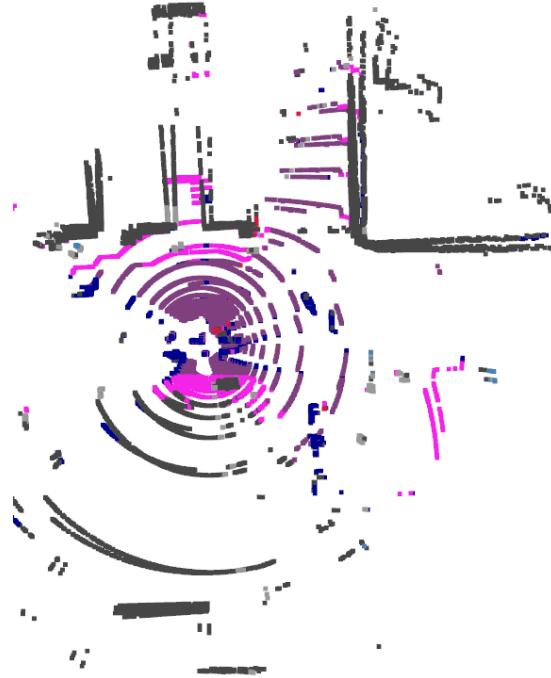
$$R = V \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & d \end{pmatrix} U^T \quad (4.3)$$

6. Υπολογισμός της μετατόπισης μέσω της εξίσωσης 4.4

$$t = centroid_B - R \times centroid_A \quad (4.4)$$

Η από πάνω μεθοδολογία περιγράφεται στο [57] ενώ πιο συγκεκριμένα τα βήματα 3-5 είναι ο αλγόριθμος του Kabsch [58], οποίος υπολογίζει τη βέλτιστη περιστροφή ανάμεσα σε δύο σετ σημείων ώστε να ταυτίζονται.

Έχοντας κάνει όλα τα βήματα που έχουν περιγράφει μέχρι σε αυτό το σημείο φτάνουμε σε ένα αποτέλεσμα σαν αυτό της εικόνας 4.18. Παρατηρώντας το συγκεχυμένη εικόνα και κρίνοντας κυρίως από το πόσο καλά έχουν ευθυγραμμιστεί οι τοίχοι των κτιρίων μπορούμε να θεωρήσουμε ότι το συγκεκριμένο αποτέλεσμα είναι αρκετά καλό δεδομένου ότι μέχρι τώρα έχει υλοποιηθεί το κομμάτι του global registration.



Σχήμα 4.18: Αποτέλεσμα ευθυγράμμισης μετά την αντιστοίχηση γράφων

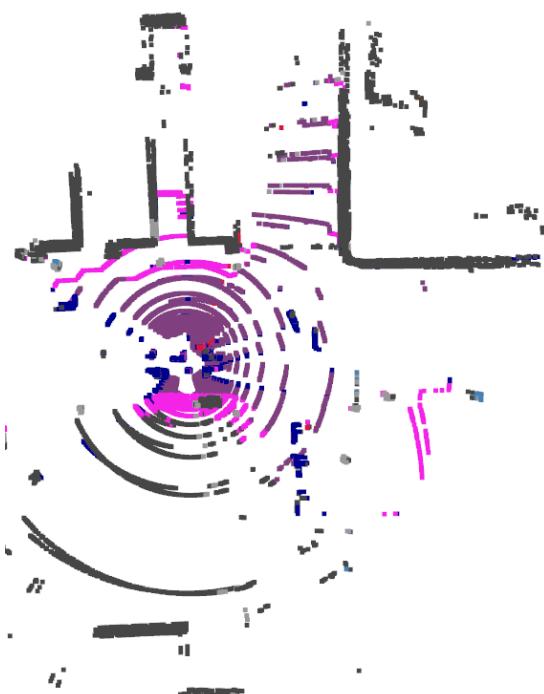
Τελική ευθυγράμμιση

Η τοπική ευθυγράμμιση αποτελεί το επόμενο βήμα της διαδικασίας, κατά το οποίο υπολογίζεται ο τελικός μετασχηματισμός. Για αυτό το στάδιο χρησιμοποιείται ο αλγόριθμος ICP και πιο

συγκεκριμένα ο point-to-point ICP. Ο αλγόριθμος αυτός λαμβάνει ως είσοδο δύο σύνολα σημείων και υπολογίζει τον καλύτερο μετασχηματισμό που ελαχιστοποιεί μια συνάρτηση κόστους. Καθώς όμως τα νέφη σημείων προέρχονται από αυτόνομα οχήματα περιέχουν χαρακτηριστικά κυκλικά σχήματα (ή δακτυλίους), που στο τέλος εισάγουν μεγάλα σφάλματα στην τελική ευθυγράμμιση. Για το λόγο αυτό, πρέπει με κάποιο τρόπο να φιλτραριστούν και να μην ληφθούν υπόψη στο στάδιο αυτό. Ο λόγος που τα μοτίβα αυτά εισάγουν μεγάλα σφάλματα είναι ότι τα αυτόνομα οχήματα έχουν διαφορετικούς κυκλικούς δακτυλίους μεταξύ τους. Ως αποτέλεσμα, τα σημεία που αντιστοιχίζονται σε αυτά να μην υπάρχουν πιθανόν καθόλου στο άλλο νέφος σημείων. Αυτό έχει ως αποτέλεσμα οποιαδήποτε προσπάθεια αντιστοίχησης να οδηγεί σε λανθασμένες εκτιμήσεις του πίνακα μετασχηματισμού.

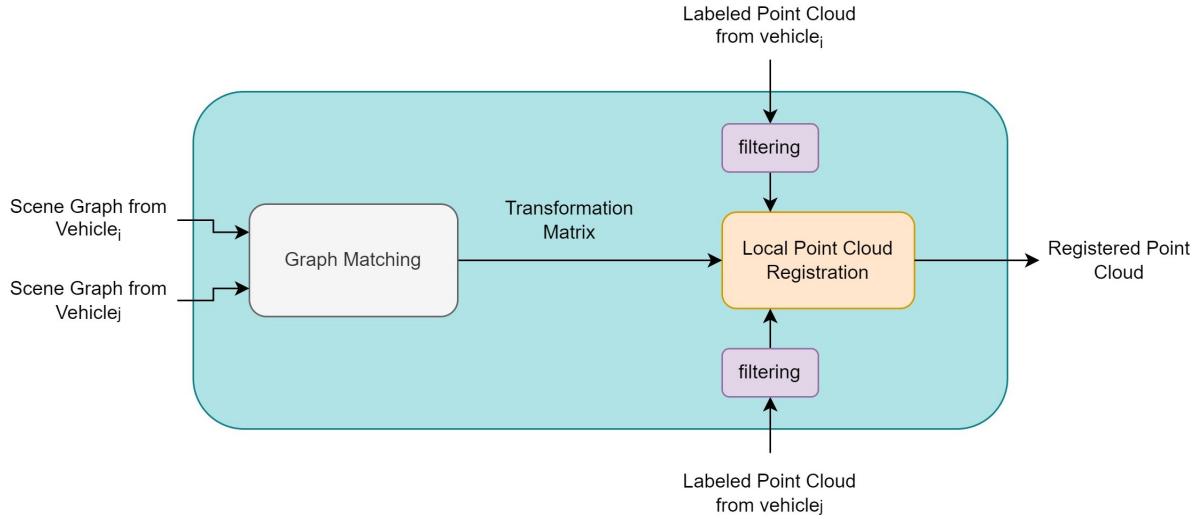
Αυτό το πρόβλημα λύνεται αρκετά εύκολα καθώς παρατηρώντας τις εικόνες κατάτμησης είναι αρκετά ξεκάθαρο ότι τα συγκεκριμένα σημεία αντιστοιχούν είτε σε σημεία του δρόμου είτε του πεζοδρομίου. Έχοντας φιλτράρει αυτά τα σημεία η μετέπειτα μεθοδολογία είναι αρκετά απλή και εύκολη. Απλώς δίνουμε στον αλγόριθμο τα φιλτραρισμένα νέφη σημείων και αυτός υπολογίζει τον βέλτιστο πίνακα μετασχηματισμού. Αυτή η μέθοδος θα δούλευε ικανοποιητικά αν διαδικασία της κατάτμησης δεν είχε λάθη, άλλα όπως φαίνεται και από το αποτέλεσμα της εικόνας 4.18 είναι αρκετά πιθανό να γίνουν και σε μερικές περιπτώσεις μπορεί να είναι ένας αρκετά μεγάλος αριθμός σημείων που έχουν ταξινομηθεί σε λάθος κατηγορία. Για αυτό το λόγο υπάρχει και ένα δεύτερο στάδιο φιλτραρίσματος πριν τη χρήση των σημείων από τον ICP, το οποίο περιλαμβάνει την αφαίρεση όλων των σημείων αδιακρίτως που βρίσκονται κάτω από ένα συγκεκριμένο όριο, δηλαδή είναι σαν να διαγράφουμε όλα τα σημεία που έχουν στον άξονα του ύψους (z-άξονας) μια τιμή μικρότερη από αυτή του ορίου. Με αυτόν τον τρόπο αφαιρούμε και τα λάθος ταξινομημένα σημεία κάνοντας την όλη διαδικασία της ευθυγράμμισης αρκετά πιο ανθεκτική.

Έχοντας ολοκληρώσει όλα τα στάδια προκύπτει η ευθυγράμμιση της εικόνας 4.19 με αρκετά βελτιωμένα αποτελέσματα σχετικά με αυτά μετά την αντιστοίχηση των γράφων σκηνής (εικόνα 4.18)



Σχήμα 4.19: Οπτικοποίηση της τελικής ευθυγράμμισης

Τέλος, όλη αυτή η διαδικασία της ευθυγράμμισης μπορεί να περιγραφεί με επάρκεια μέσω του σχεδιαγράμματος της εικόνας 4.20.

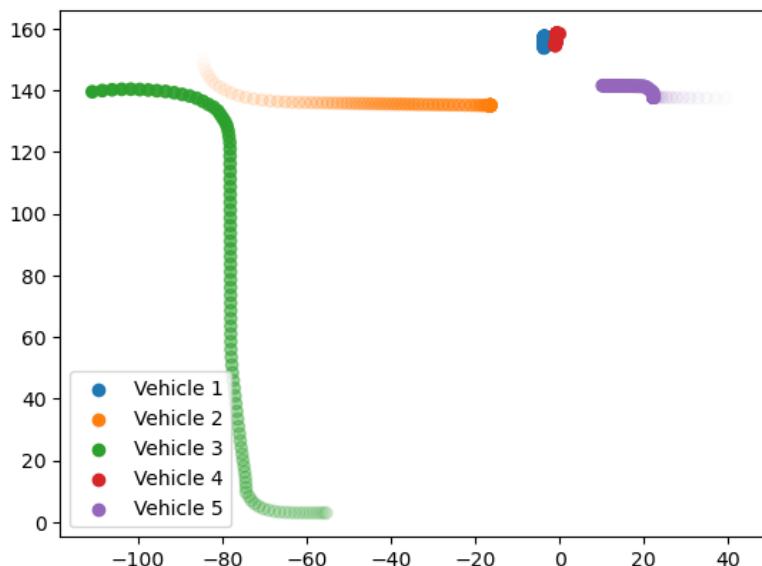


Σχήμα 4.20: Διαδικασία για την ευθυγράμμιση των νεφών σημείων χρησιμοποιώντας το γράφο σκηνής

Κεφάλαιο 5

Αποτελέσματα και η Οπτικοποίηση τους

Το σετ δεδομένων που χρησιμοποιήθηκε καθ' όλη την ανάπτυξη της συγκεκριμένης εργασίας περιέχει δεδομένα από πέντε διαφορετικά οχήματα τα οποία ακολουθούν διαφορετικές τροχιές το κάθε ένα. Οι τροχιές αυτές φαίνονται στο σχήμα 5.1 και εκτός από το όχημα 3 όλα τα άλλα έχουν κάποια κοινή στιγμή που να βλέπουν την ίδια σκηνή ή διασταύρωση. Επιπλέον είναι σημαντικό να σημειωθεί ότι τα υπόλοιπα οχήματα αντιλαμβάνονται διάφορα τμήματα της ίδιας σκηνής με διαφορετικές οπτικές γωνίες. Αυτό δημιουργεί μια επικάλυψη στα δεδομένα, προσφέροντας έτσι μια ολοκληρωμένη εικόνα, όπου κάποια τμήματα μπορεί να έχουν μικρή επικάλυψη, ενώ άλλα σε άλλα μπορεί να είναι πιο μεγάλη. Ένα τέτοιο παράδειγμα είναι το ζεύγος οχημάτων 1 και 4 που είναι δίπλα το ένα στο άλλο έχοντας μεγαλύτερη επικάλυψη στους παραγόμενους γράφους σκηνής.



Σχήμα 5.1: Τροχιές όλων των οχημάτων

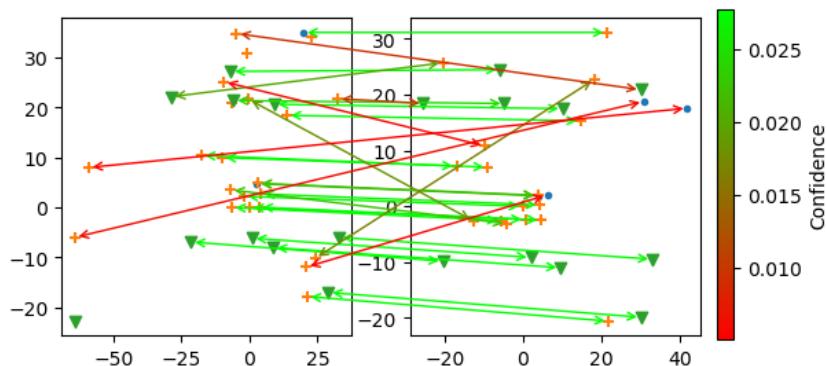
5.1 Μεγάλη Επικάλυψη

Στα παραδείγματα όπου υπάρχει σημαντική επικάλυψη στους γράφους σκηνής, δηλαδή στα οχήματα 1 και 4, τα οποία βρίσκονται σε πολύ κοντινή απόσταση και έχουν τον ίδιο προσανατολισμό και κατεύθυνση, παρατηρείται ότι αυτή η συνάφεια έχει θετικό αντίκτυπο στην ακρίβεια

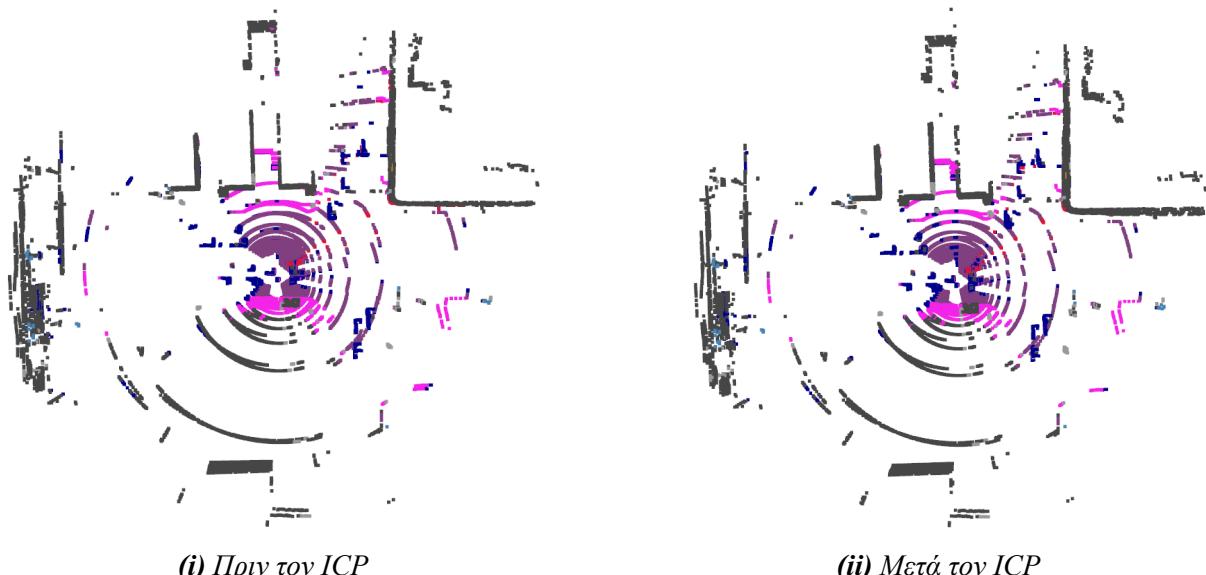
και την αποτελεσματικότητα της διαδικασίας αντιστοίχησης των γράφων σκηνής. Η ύπαρξη δύο παρόμοιων γράφων επιτρέπει μια πιο αξιόπιστη αναγνώριση και σύγκριση, οδηγώντας έτσι σε πιο αξιόπιστα και ακριβή αποτελέσματα. Αυτό κάνει την όλη διαδικασία της αντιστοίχησης αρκετά εύκολη και δύσκολα προκύπτουν σημαντικά λάθη αλλά και στην περίπτωση που έχουν αντιστοιχηθεί λάθος κάποιοι κόμβοι κατά πάσα πιθανότητα δεν θα βρίσκονται στις πρώτες επιλογές για την εύρεση του πίνακα μετασχηματισμού.

5.1.1 Οπτικοποίηση

Στην από κάτω εικόνα (σχήμα 5.2) φαίνονται το πως αντιστοιχήθηκαν οι κόμβοι του κάθε γράφου. Πιο αναλυτικά με πράσινο φαίνονται οι αντιστοιχίσεις για τις οποίες είμαστε πιο σίγουροι ότι είναι σωστές, ενώ με κόκκινο είναι αυτές που είμαστε λιγότερο σίγουροι. Ακόμα τα χρώματα κάθε αντιστοίχησης έχουν διαβαθμιστεί έτσι ώστε να αντιπροσωπεύουν το πόσο σίγουροι είμαστε για αυτήν, κάνοντας με αυτόν το τρόπο πιο εύκολη τη σύγκριση μεταξύ των αποτελεσμάτων. Τέλος εύκολα παρατηρούμε ότι οι κόκκινες αντιστοιχίσεις είναι πράγματι λανθασμένες και σωστά έχουν ταξινομηθεί στο τέλος της λίστας με κόκκινο χρώμα.



Σχήμα 5.2: Αντιστοίχηση γράφων σε παράδειγμα με μεγάλο ποσοστό επικάλυψης



Σχήμα 5.3: Αποτέλεσμα ευθυγράμμισης πριν και μετά τον ICP για παράδειγμα με μεγάλο ποσοστό επικάλυψης

5.2 Μικρή Επικάλυψη

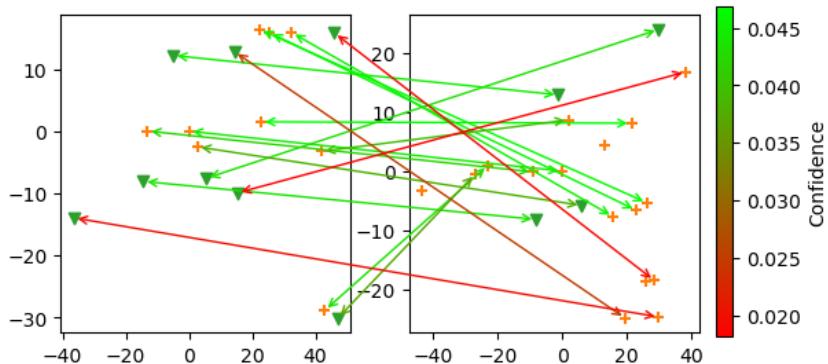
Από την άλλη πλευρά όταν η επικάλυψη των γράφων σκηνής είναι μικρότερη και κατά επέκταση τα νέφη σημείων έχουν λιγότερα κοινά στοιχεία τα αποτελέσματα της ευθυγράμμισης έχουν μεγαλύτερα σφάλματα είτε σε κάποιες περιπτώσεις είναι τελείως λάθος.

5.2.1 Οπτικοποίηση

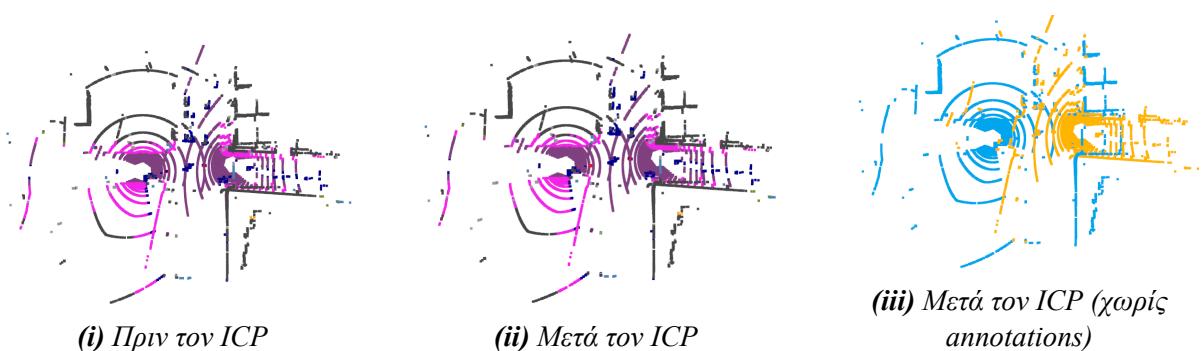
Στη συνέχεια ακολουθούν δύο παραδείγματα στα οποία φαίνονται το πόσο καλά ή αλλά και κακά έχει γίνει η ευθυγράμμιση. Στο πρώτο παρόλο που δεν υπάρχει αρκετά μεγάλη επικάλυψη, η προτεινομένη μέθοδος καταφέρνει να αναγνωρίσει κοινά μοτίβα και να τα αντιστοιχίσει. Αντίθετα, στο δεύτερο παράδειγμα ο αλγόριθμος δεν καταφέρνει να υπολογίσει τον σωστό πίνακα μετασχηματισμού, με συνέπεια το τελικό αποτέλεσμα να απέχει αρκετά από το πραγματικό.

Παράδειγμα 1

Παρατηρώντας το τρίτο σχήμα της εικόνας 5.5 βλέπουμε ότι η ευθυγράμμιση έχει γίνει αρκετά καλά αν σκεφτούμε τα δύο οχήματα δεν έχουν απευθείας επαφή το ένα με το άλλο. Αντό το καταλαβαίνουμε από το γεγονός ότι δεν υπάρχουν σημεία στο νέφος ενός οχήματος στη θέση που βρίσκεται το άλλο αυτόνομο όχημα.



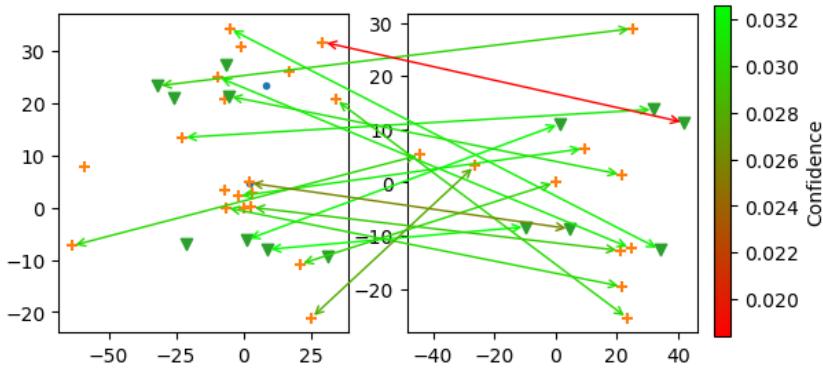
Σχήμα 5.4: Αντιστοίχηση γράφων στο πρώτο παράδειγμα με μικρό ποσοστό επικάλυψης



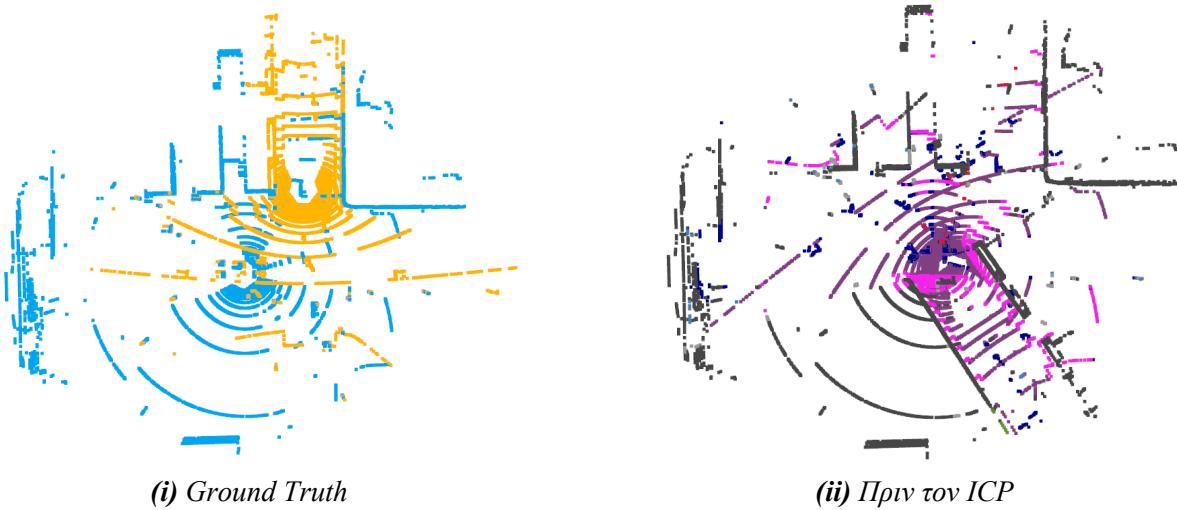
Σχήμα 5.5: Αποτελέσματα ευθυγράμμισης πριν και μετά τον ICP για το πρώτο παράδειγμα με μικρό ποσοστό επικάλυψης

Παράδειγμα 2

Σε αυτό το παράδειγμα χρησιμοποιήθηκαν δεδομένα από τα οχήματα 1 και 5, τα οποία βρίσκονται σε δρόμους που διασταυρώνονται κάθετα και ως εκ τούτου δεν υπάρχει αρκετή κοινή πληροφορία στου γράφους ώστε να γίνουν οι σωστές αντιστοιχίσεις. Αυτό έχει σαν συνέπεια να υπολογιστεί ένας εξολοκλήρου λάθος πίνακας μετασχηματισμού που να διαφέρει σε μεγάλο βαθμό από τον πραγματικό και για αυτόν τον λόγο έχει προκύψει το αποτέλεσμα της εικόνας 5.7.



Σχήμα 5.6: Αντιστοιχηση γράφων στο δεύτερο παράδειγμα με μικρό ποσοστό επικάλυψης



Σχήμα 5.7: Αποτελέσματα ενθυγράμμισης πριν τον ICP και η και η ground truth ενθυγράμμιση

5.3 Αποτελέσματα

Παρατηρώντας και τα οπτικά αποτελέσματα αλλά και τον πίνακα 5.1 βλέπουμε ότι αν έχει βρεθεί ο σωστός πίνακας μετασχηματισμού η τελική ενθυγράμμιση είναι αρκετά καλή. Ακόμα βλέπουμε ότι το τελευταίο στάδιο της ενθυγράμμισης που περιλαμβάνει την εφαρμογή του αλγορίθμου ICP δεν προσφέρει σημαντική βελτίωση, πράγμα που φαίνεται μέσα από τον πίνακα και πιο συγκεκριμένα από το γεγονός ότι σε κάποιες περιπτώσεις το απόλυτο σφάλμα αυξάνεται αντί να μειώνεται.

Τέλος, αξίζει να περιγράφει ο τρόπος υπολογισμού των σφαλμάτων τόσο για την μετατόπιση όσο και την περιστροφή, που φαίνονται πίνακα 5.1. Αυτό γίνεται συγκρίνοντας τους

ground truth μετασχηματισμούς με αυτούς που προέκυψαν στο τέλος της μεθοδολογίας και πιο συγκεκριμένα τη μετατόπιση αλλά την περιστροφές γύρω από τους άξονες (X, Y, Z) που κωδικοποιούν αυτοί οι πίνακες. Για παράδειγμα, ας υποθέσουμε ότι ο πίνακας G αναπαριστά τη σωστό μετασχηματισμό με $t_G = [x_G, y_G, z_G]$ για τη μετατόπιση και $\theta_G = [\theta_G^x, \theta_G^y, \theta_G^z]$ για την περιστροφή. Έστω επίσης ότι ο μετασχηματισμός A που προέκυψε από τη μεθοδολογία που χρησιμοποιήθηκε. Σε αυτήν την περίπτωση, τα σφάλματα υπολογίζονται ως εξής:

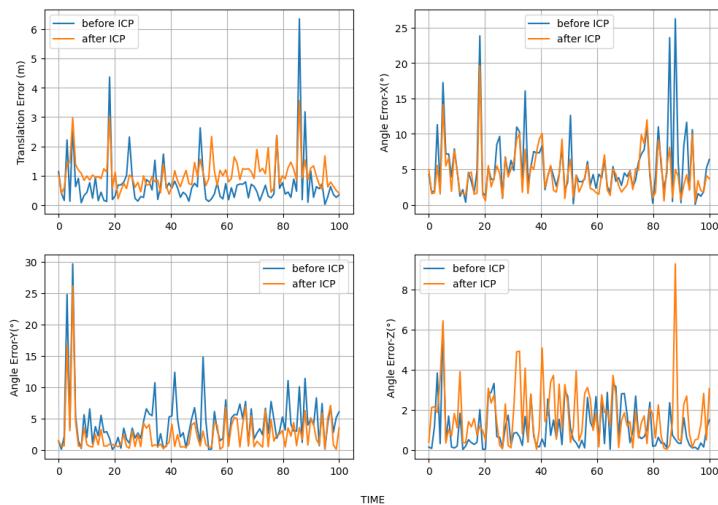
$$\text{Translation Error} = \|t_A - t_G\|$$

$$\text{Angle Error} = |\theta_A - \theta_G|$$

		Translation Error (m)	Angle-X Error (°)	Angle-Y Error (°)	Angle-Z Error (°)
Μεγάλη Επικάλυψη	Πριν τον ICP	0.28	3.27	1.19	0.09
	Μετά τον ICP	0.89	1.71	0.49	0.58
Μικρή Επικάλυψη (Παράδειγμα 1)	Πριν τον ICP	2.55	0.07	0.73	3.95
	Μετά τον ICP	2.47	0.11	0.51	4.11
Μικρή Επικάλυψη (Παράδειγμα 2)	Πριν τον ICP	30.75	-0.55	2.96	212.93
	Μετά τον ICP	29.91	1.91	2.14	209.87

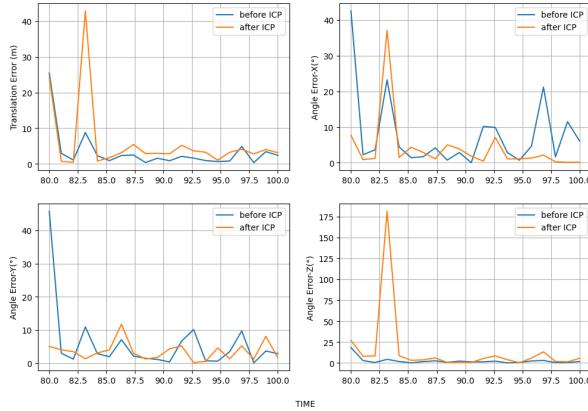
Πίνακας 5.1: Σφάλματα ευθυγράμμισης σε παραδείγματα με μικρό και μεγάλο ποσοστό επικάλυψης

Στη συνέχεια παρουσιάζονται τα αποτελέσματα της ευθυγράμμισής για τα οχήματα καθόλη τη διάρκεια της προσομοιώσεις. Ξεκινώντας από τα οχήματα που έχουν συνεχώς μεγάλο ποσοστό επικάλυψης (οχήματα 1 και 4) παρατηρούμε ότι πολύ λίγες φορές το τελικό σφάλμα αυξάνεται τόσο ώστε να θεωρηθεί λάθος το αποτέλεσμα και τις φορές που συμβαίνει οριακά ξεπερνάει τα δύο μέτρα το σφάλμα μετατόπισης.

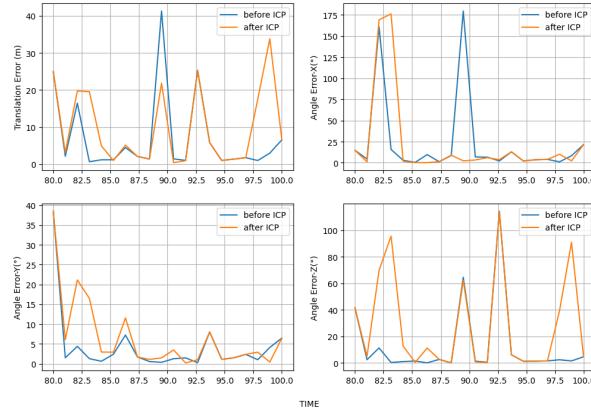


Σχήμα 5.8: Σφάλματα ευθυγράμμισης για το ζεύγος οχημάτων 1,4

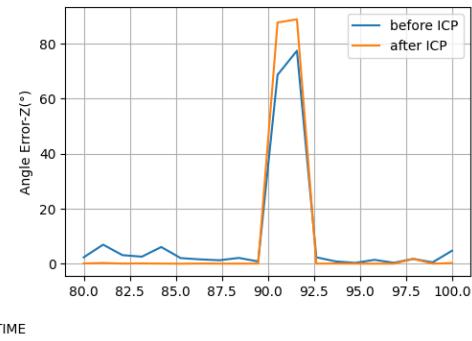
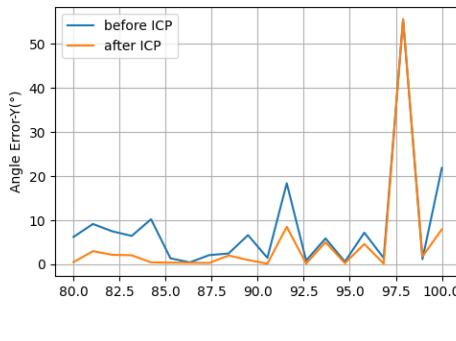
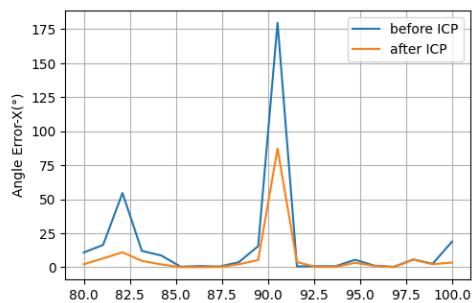
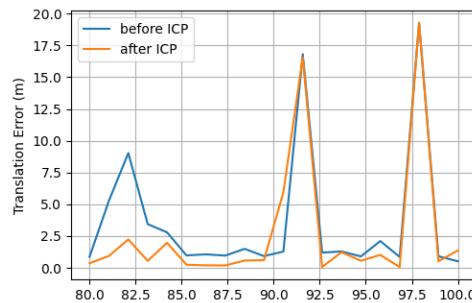
Οι από κάτω γραφικές παραστάσεις (σχήμα 5.9) περιλαμβάνουν τα σφάλματα για το όχημα 2. Καθώς το συγκεκριμένο όχημα δεν βλέπει την ίδια περιοχή με τα υπόλοιπα οχήματα της σκηνής καθ' όλη τη διάρκεια της προσομοίωσης, αλλά μόνο στο τέλος, έχει κοινά στοιχεία μόνο στις τελευταίες χρονικές στιγμές της. Αυτό έχει σαν αποτέλεσμα να είναι αδύνατο να βρεθούν κοινά στοιχεία μεταξύ των διαφορετικών οπτικών και για αυτόν τον λόγο έχουν παραβλεφθεί οι αρχικές χρονικές στιγμές.



(i) Ζεύγος οχημάτων 2,4



(ii) Ζεύγος οχημάτων 2,1



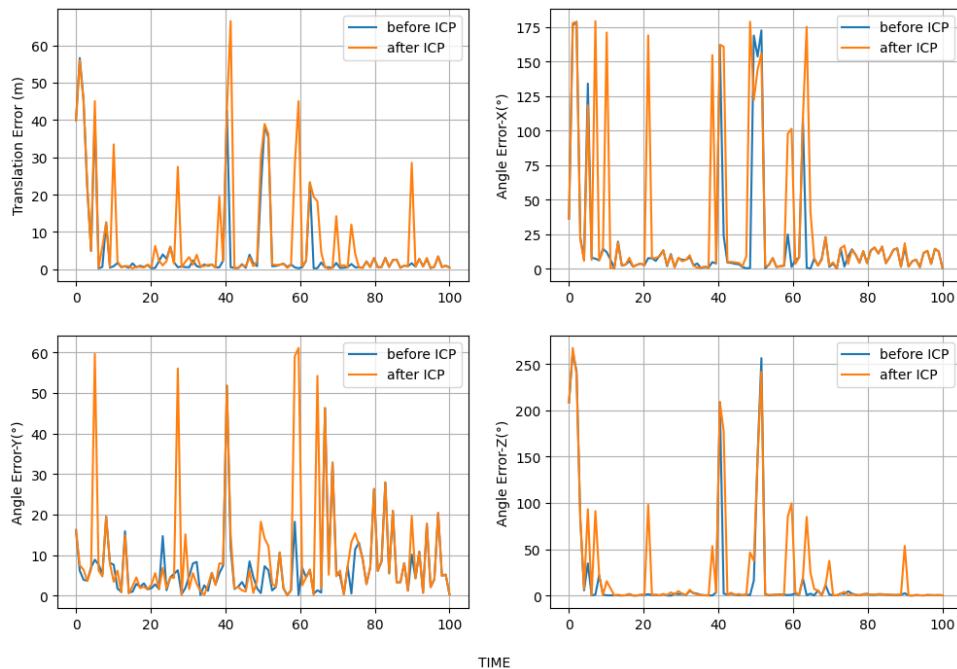
(iii) Ζεύγος οχημάτων 2,5

Σχήμα 5.9: Σφάλματα ενθυγράμμισης σε παραδείγματα με μικρή επικάλυψη

Γενικά παρατηρώντας τα αποτελέσματα όλων γραφικών παραστάσεων παρατηρούμε ότι το τελευταίο στάδιο της ενθυγράμμισης δεν οδηγεί σε αισθητά καλύτερα αποτελέσματα σε σύγκριση με το αν δεν υπήρχε. Αυτό είναι πιθανό να οφείλεται στο γεγονός ότι τα νέφη σημείων που χρησιμοποιήθηκαν είναι αρκετά αραιά με πυκνότητα σημείων που μεταβάλλεται σε σχέση με την απόσταση από τον αισθητήρα. Δεδομένο αυτών είναι αρκετά δύσκολο να βρίσκει ο αλγόριθμος ICP τα σωστά σημεία αντιστοίχησης, ώστε να μπορεί να ελαχιστοποιήσει την κατάλληλη απόσταση. Τέλος, αξίζει να σημειωθεί ότι το τελευταίο στάδιο ενθυγράμμισης που

περιλαμβάνει την εφαρμογή του ICP ως επί των πλείστων βελτιώνει μόνο τα σφάλματα των γωνιών γύρω από τους άξονες X και Y που αντιστοιχούν στο roll και pitch, αντίστοιχα, ενώ αντίθετα αυξάνει τα σφάλματα που έχουν σχέση με την μετατόπιση και τη γωνία γύρω από το άξονα Z (yaw).

Κλείνοντας, παρουσιάζονται τα σφάλματα για το ζεύγος οχημάτων 4 και 5 (σχήμα 5.10). Παρόλο που αυτά τα οχήματα έχουν μικρή επικάλυψη στα νέφη σημείων τους η προτεινόμενη μέθοδος καταφέρνει και βρίσκει τη σχετική θέση των οχημάτων αρκετά νωρίς σε σχέση με τα υπόλοιπα παραδείγματα. Επιπλέον, πρέπει να σημειωθεί ότι το όχημα 5 βρίσκεται σε κεκλιμένο επίπεδο κάνοντας με αυτόν τον τρόπο αρκετά πιο σημαντικές της γωνίες γύρω από τους άξονες X και Y καθώς είναι υπεύθυνες για το πως τέμνονται τα επίπεδα των δρόμων.



Σχήμα 5.10: Σφάλματα ενθυγράμμισης για το ζεύγος οχημάτων 4,5

Όλες οι από πάνω γραφικές παραστάσεις μπορούν να συγκεντρωθούν στον πίνακα 5.2, όπου φαίνονται οι μέσες τιμές και οι τυπικές αποκλίσεις για κάθε προσομοίωση. Για να αξιολογηθεί η αποτελεσματικότητα της μεθόδου κυρίως στις περιπτώσεις όπου η μέθοδος έχει υπολογίσει σωστά τον μετασχηματισμό, έχουν απορριφθεί οι στιγμές στις οποίες είχαμε μεγάλα σφάλματα και πιο συγκεκριμένα όταν το σφάλμα μετατόπισης ήταν μεγαλύτερο των 5m και όταν τα σφάλματα των γωνιών ήταν μεγαλύτερα των 10° και επιπλέον για όλα τα αποτελέσματα χρησιμοποιήθηκε τιμή κατωφλίου για τον ICP 1m.

Παρατηρώντας τον πίνακα 5.2 και συγκρίνοντας τα αποτελέσματα πριν και μετά τη χρήση του ICP επιβεβαιώνονται οι παρατηρήσεις σχετικά με την αποτελεσματικότητα της μεθόδου που είχαν εξαχθεί μέσω των γραφικών. Πιο αναλυτικά παρόλο που ο ICP βελτιώνει τα σφάλματα των γωνιών που έχουν σχέση με το πως τέμνονται τα επίπεδα των δρόμων (γωνίες γύρω από τους άξονες X και Y) κάποιες φορές κατά 50% χειροτερεύει αισθητά το σφάλμα μετατόπισης και το σφάλμα της γωνίας γύρω από το άξονα Z, τα οποία είναι αρκετά πιο σημαντικά για την κατανοήσει της σκηνής, καθώς έχουν άμεση σχέση με τις σχετικές αποστάσεις των αντικειμένων της σκηνής.

Τέλος, αξίζει να σχολιαστεί η διαφορά στα δείγματα που χρησιμοποιήθηκαν για την εξαγωγή των στατιστικών μεταξύ του πριν και μετά του ICP. Η διαφορά αυτή υποδηλώνει τις στιγμές

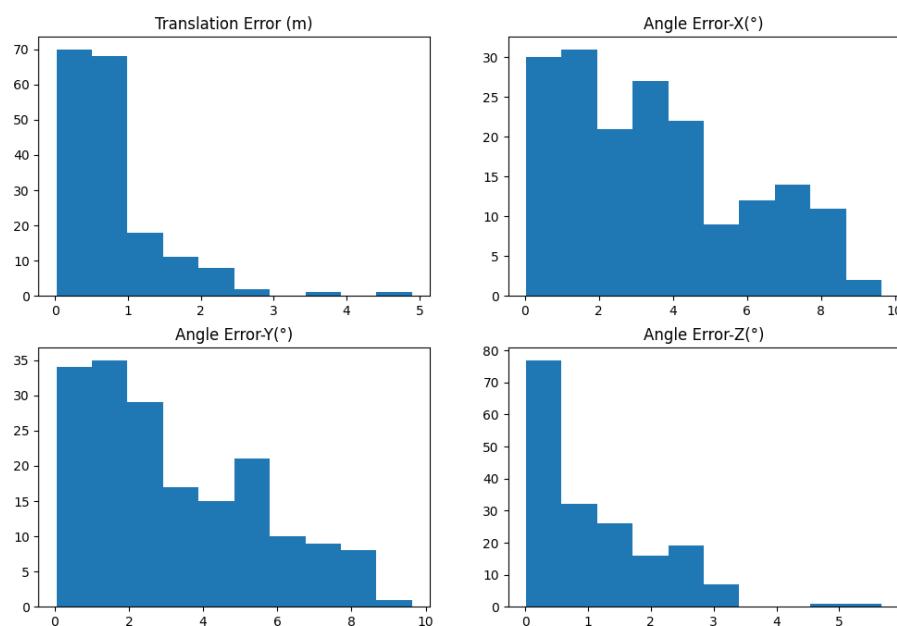
	Translation Error (m)		Angle-X Error (°)		Angle-Y Error (°)		Angle-Z Error (°)		#Samples	Vehicles
	mean	std	mean	std	mean	std	mean	std		
Before ICP	0.49	0.35	4.10	2.23	3.34	2.15	0.99	1.01	85/100	(1,4)
After ICP	1.27	0.59	2.12	1.51	1.18	1.39	1.86	1.49	97/100	
Before ICP	0.73	0.61	2.01	1.76	2.65	1.70	1.32	0.91	43/60	(2,1) (2,4) (2,5)
After ICP	2.09	0.86	2.52	0.96	1.58	2.29	3.41	2.23	51/60	
Before ICP	1.10	0.92	3.78	2.74	2.99	2.40	1.07	1.10	57/100	(4,5)
After ICP	1.88	1.01	3.70	2.07	3.23	2.55	1.89	1.87	62/100	

Πίνακας 5.2: Σφάλματα ενθυγράμμισης (συγκεντρωτικός πίνακας)

στις οποίες το σφάλμα πριν τον ICP ήταν εκτός ορίων ενώ μετά τη χρήση του έπαψε να είναι. Αν υπολογιστεί το ποσοστό των φορών που η χρήση του ICP βελτίωσε τα σφάλματα ώστε να είναι εντός των ορίων είναι περίπου 5%-10% ανάλογα την προσομοίωση. Το γεγονός ότι αυτό το ποσοστό είναι θετικό, δηλαδή αυξάνονται τα δείγματα από το πριν στο μετά, φανερώθει επίσης ότι η αύξηση των σφαλμάτων μετατόπισης και γωνίας yaw δεν είναι τόσο μεγάλες ώστε να θέσει εκτός ορίων κάποια δείγματα, αλλά αντιθέτως λειτουργεί με ωφέλιμο τρόπο για το τελικό αποτέλεσμα ως σύνολο.

Επιπλέον ο πίνακας 5.2 που περιέχει όλα τα σφάλματα πριν την εφαρμογή του ICP μπορεί να περιγράφει και από τις γραφικές παραστάσεις του σχήματος 5.11, οποίες συνοψίζουν τα σφάλματα για όλα τα ζευγάρια οχημάτων που περιέχονται στον συγκεκριμένο πίνακα. Σε αυτό το σχήμα φαίνονται οι κατανομές των σφαλμάτων μετά το φίλτρο που περιγράφηκε στο τέλος της σελίδας 66.

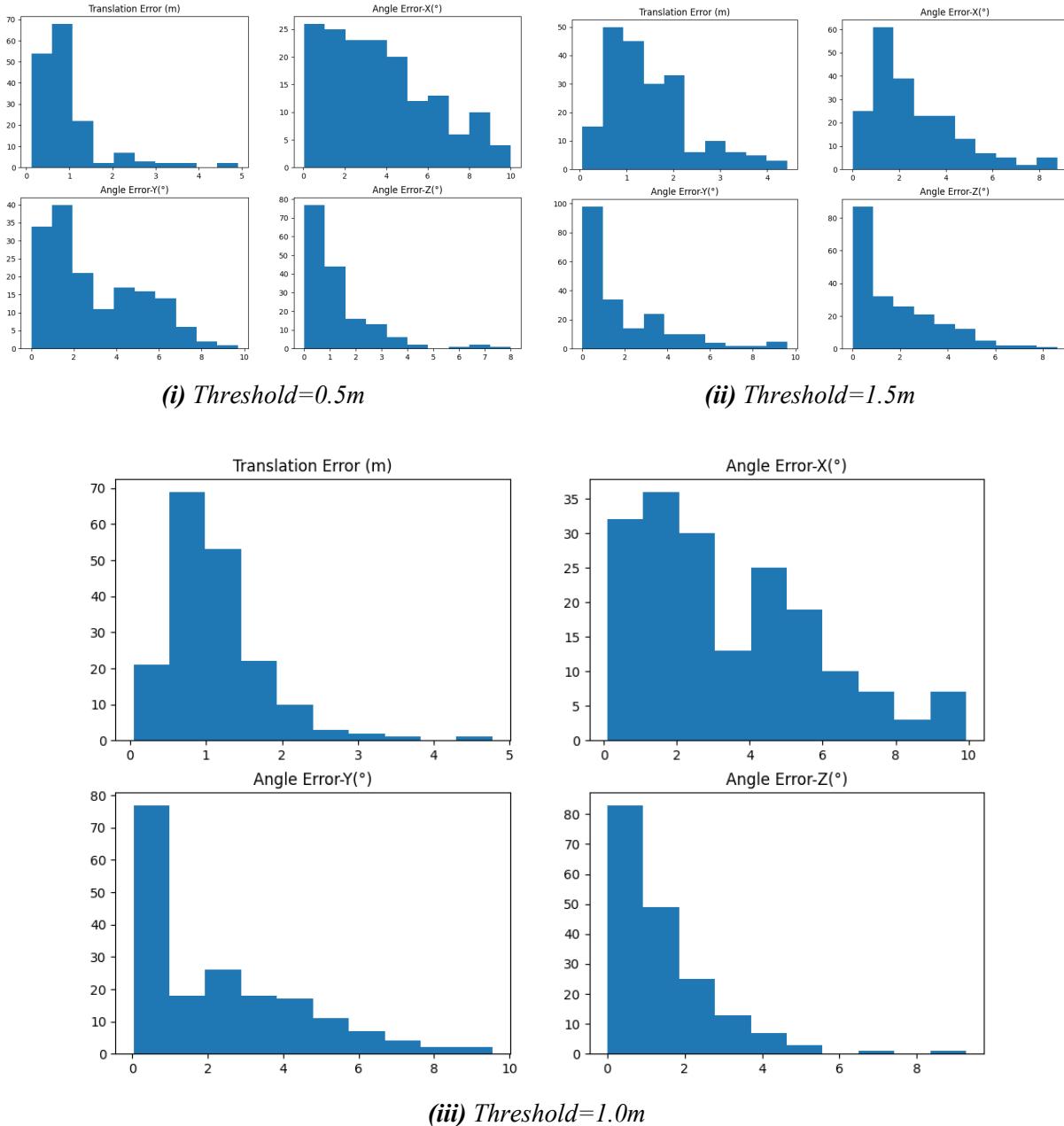
Όπως παρατηρούμε η προτεινόμενη μέθοδος με την αντιστοίχηση γράφων αποδίδει αρκετά καλά στην εύρεση της μετατόπισης αλλά και της γωνίας γύρω από τον άξονα Z, αλλά έχει μεγάλα σφάλματα στις υπόλοιπες γωνίες με σχετικά μεγάλη διακύμανση. Είναι πιθανό ότι λόγω του ότι οι σκηνές που χρησιμοποιήθηκαν έχουν τα χαρακτηριστικά τους κατά μήκος του επίπεδου του δρόμου να μειώνουν την ακρίβεια της μεθόδου.



Σχήμα 5.11: Κατανομή σφαλμάτων πριν την εφαρμογή του ICP

Λόγω των σφαλμάτων που υπάρχουν στις γωνίες γύρω από τους άξονες X και Y εισήχθηκε

και το τελευταίο στάδιο ευθυγράμμισης που περιλαμβάνει τη χρήση του αλγορίθμου ICP. Ο αλγόριθμος αυτός δέχεται σαν παράμετρο τη μέγιστη απόσταση (Threshold), η οποία μπορεί να θεωρηθεί ως μια ακτίνα κύκλου. Εντός αυτής του θα γίνει η αναζήτηση για τον κατάλληλο “γείτονα” ενός σημείου του pointcloud στο αντίστοιχο σημείο του άλλου pointcloud. Δεδομένο αυτού είναι εξαιρετικά σημαντικό να βρεθεί η κατάλληλη τιμή αυτής της παραμέτρου.



Σχήμα 5.12: Κατανομή σφαλμάτων για διάφορες τιμές του κατωφλίου στον αλγόριθμο ICP

Στο σχήμα 5.12 φαίνονται οι κατανομές σφαλμάτων για διάφορες τιμές της παραμέτρου κατωφλίου. Σχολιάζοντας τα αποτελέσματα για το σχήμα 5.12i παρατηρούμε ότι οι κατανομές δεν άλλαξαν σε μεγάλο βαθμό, αλλά υπήρξε μία μικρή βελτίωση σχεδόν σε όλα τα κρίσιμα μεγέθη. Από την άλλη πλευρά όταν η παράμετρος πήρε την τιμή “Threshold=1.5m” (σχήμα 5.12ii) παρατηρούμε με βελτίωση στα σφάλματα των όλων των γωνιών αλλά και μια αισθητή αλλαγή προς την λάθος πλευρά για το σφάλμα μετατόπισης. Τέλος, στο σχήμα 5.12iii βλέπουμε ότι και εδώ σχεδόν όλες οι γωνίες βελτιώθηκαν λιγότερο μεν αυτή που είναι γύρω από τον άξονα

Χ συγκριτικά με τις άλλες. Όμως σε αυτό το παράδειγμα δεν ήταν τόσο έντονη η χειροτέρευση στο σφάλμα μετατόπισης, που παρατηρήθηκε όταν το κατώφλι ήταν 1.5m.

Αυτή η διακύμανση στα σφάλματα για τις διάφορες τιμές του κατωφλίου μπορεί να εξηγηθεί από το γεγονός ότι το νέφος σημείων καθώς είναι αρκετά αραιό σε μεγάλες αποστάσεις από τον αισθητήρα και δεδομένου ότι αυτά τα σημεία είναι πιθανόν να ανήκουν σε κτίρια είναι αρκετά δύσκολο να υπάρχει ακριβώς το ίδιο σημείο και στα δύο νέφη. Άρα είναι λογικό να υπάρχουν μικρές μεταβολές όταν το κατώφλι είναι παίρνει χαμηλές τιμές, λόγω του ότι δεν υπάρχουν σημεία εντός του κύκλου που σχετίζεται με την εύρεση του κατάλληλου γείτονα. Από την άλλη πλευρά όταν η ακτίνα του κύκλου είναι μεγάλη αυξάνεται και η πιθανότητα συμπερίληψης ακραίων τιμών στον υπολογισμό της αντιστοιχίας μεταξύ των δύο νεφών. Τα ακραία σημεία (outliers) θεωρούμε σημεία που δεν αντιστοιχούν καλά μεταξύ των νεφών και η συμπερίληψή τους στη διαδικασία ευθυγράμμισης μπορεί να οδηγήσει σε ανακρίβειες.

Κεφάλαιο 6

Επίλογος

6.1 Συμπεράσματα

Στο πλαίσιο αυτής της διπλωματικής εργασίας, αρχικά πραγματοποιήθηκε μια εκτενής επισκόπηση της λειτουργίας των νευρωνικών δικτύων, αλλά και γενικότερα όλων των τεχνολογιών που χρησιμοποιήθηκαν κατά τη διάρκεια της διαδικασίας ανάπτυξης, ενώ στη συνέχεια παρουσιάστηκε ένας καινοτόμος τρόπος για την ευθυγράμμιση νεφών σημείων που εκμεταλλεύεται τη σημασιολογική πληροφορία της σκηνής. Τέλος, πραγματοποιήθηκαν κάποια πειράματα σε συνθετικό περιβάλλον όπου συμμετέχουν και επικοινωνούν κάποια αυτόνομα οχήματα, ώστε να αξιολογηθεί αυτή η μέθοδος. Με αυτόν τον τρόπο μπορέσαμε να καταλάβουμε που υστερεί και που υπερτερεί η προτεινόμενη μέθοδος.

Πιο αναλυτικά και ξεκινώντας από το σετ δεδομένων παρατηρούμε ότι είχε κάποιες ασυνέπειες όσον αφορά τις αντιστοιχίες των εικόνων με το αντίστοιχο νέφος σημείων. Πιο συγκεκριμένα παρόλο που σε κάποια σημεία φαινόταν ότι υπάρχει κάποιο κτίριο στο νέφος αυτό δεν φαινόταν (εικόνα 4.10 και 4.1, το κτίριο στα δεξιά του οχήματος), δημιουργώντας με αυτόν το τρόπο προβλήματα στην σημασιολογική κατάτμηση του. Ακόμα το γεγονός ότι τα δεδομένα προέρχονταν μέσα από προσδομοίωση έκαναν τη χρήση έτοιμων προεκπαιδευμένων δικτύων μη αποτελεσματική εισάγοντας με αυτόν τον τρόπο άλλο ένα στάδιο στην όλη διαδικασία της ανάπτυξης.

Εκτός από αυτό, το πρόβλημα της κατηγοριοποίησης των τρισδιάστατων σημείων και ο τρόπος με τον οποίο προσπαθήθηκε να επιλυθεί παρουσιάζουν αρκετά μειονεκτήματα. Ένα από αυτά αναδύεται λεπτομερώς μέσω της εικόνας 4.3, όπου η κάμερα και ο αισθητήρας LIDAR έχουν διαφορετικά οπτικά πεδία. Αυτό οδηγεί σε εσφαλμένη κατηγοριοποίηση ορισμένων σημείων του νέφους που βρίσκονται πίσω από άλλα αντικείμενα. Παρόλο που στην πλειονότητα των περιπτώσεων τα σημεία που κατηγοριοποιήθηκαν εσφαλμένα ανήκουν σε περιοχές του δρόμου ή το πεζοδρομίου, επιτυγχάνοντας έτσι την ευκολότερη φιλτράρισή τους, αυτή η μέθοδος δεν αποτελεί τον ενδεδειγμένο τρόπο αντιμετώπισης του προβλήματος. Η διαγραφή των σημείων που βρίσκονται κάτω από ένα συγκεκριμένο κατακόρυφο όριο αν και αποτελεί ένα είδος φιλτραρίσματος με βάση το ύψος που υλοποιείται αρκετά εύκολα δεν είναι και σωστός τρόπος καθώς μπορεί να εφαρμοστεί μόνο όταν η σκηνή βρίσκεται σε περιβάλλον που δεν έχει κλίση, πράγμα που εισάγει αρκετούς περιορισμούς.

Τέλος, αξίζει να σχολιαστεί η επίδοση της μεθόδου όταν η επικάλυψη των νεφών των οχημάτων είναι σχετικά μικρή και οι γράφοι σκηνής δεν έχουν αρκετούς κοινούς κόμβους. Σε αυτήν την περίπτωση δεν είναι που εύκολο να βρεθούν σωστές συσχετίσεις μεταξύ των γράφων, δεδομένου ότι χρειάζονται τουλάχιστον τρεις κόμβοι που να είναι κοινοί και στους δύο γράφους για να βρεθεί ένας καλός αρχικός πίνακας μετασχηματισμού. Είναι, συνεπώς, προφανές ότι η

μέθοδος αντιμετωπίζει προκλήσεις σε συνθήκες όπου η πυκνότητα των δεδομένων είναι χαμηλή και η επικάλυψη είναι περιορισμένη, επηρεάζοντας με αυτόν τον τρόπο την συνολική απόδοση της μεθόδου αρνητικά και καθιστώντας την ανεπαρκή για εφαρμογή σε περιβάλλοντα με περιορισμένη διαθεσιμότητα δεδομένων.

6.2 Μελλοντικές Επεκτάσεις

Είναι σαφές ότι η προτεινόμενη μέθοδος έχει αρκετά καλές επιδόσεις υπό συγκεκριμένες καταστάσεις, ωστόσο αξίζει να γίνει μια περαιτέρω μελέτη για τρόπους βελτίωσής της, που να να μπορούν να επιλύσουν κάποιες από τις αδυναμίες της. Πιο αναλυτικά η χρήση δύο αισθητήρων που έχουν μία σχετική απόσταση μεταξύ τους εισάγει κάποια προβλήματα που μπορούν να επιλυθούν αν γινόταν χρήση μόνο του νέος και πιο συγκεκριμένα μόνο των καμερών. Πλέον υπάρχουν αρκετά αξιόπιστα μοντέλα που μπορούν να προβλέψουν το χάρτι βάθους (depth map) μιας εικόνας εξαλείφοντας με αυτόν τον τρόπο τη χρήση του αισθητήρα LIDAR. Ακόμα αξίζει να μελετηθούν και μέθοδοι που κάνουν χρήση μεθόδων αντιστοίχησης χαρακτηριστικών ανάμεσα σε ένα ζεύγος εικόνων (image matching). Με αυτόν τον τρόπο ο γράφος σκηνής θα γινόταν ακόμα πιο πυκνός βοηθώντας την τελική αντιστοίχηση ειδικά σε περιβάλλοντα με ελάχιστη σημασιολογική πληροφορία. Τέλος από τη στιγμή που δεν γίνεται χρήση του LIDAR μπορεί να γενικευτεί το πρόβλημα από την ευθυγράμμιση νεφών σημείων σε πρόβλημα δημιουργίας μιας εικονικής σκηνής που θα μοιράζεται ανάμεσα στα εμπλεκόμενα οχήματα παρέχοντας έτσι αρκετά μεγάλη ανάλυση και πλήθος πληροφορίας. Αυτό όμως θα χρειαστεί και ένα καινούργιο τρόπο αξιολόγησης των αποτελεσμάτων δημιουργώντας την ανάγκη για ανάπτυξη περισσοτέρων σκηνών προσομοίωσης.

Βιβλιογραφία

- [1] Farrukh Hafeez, Usman Ullah Sheikh, Nasser Alkhaldi, Hassan Al Garni, A Garni, Zeeshan Arfeen, and Saifulnizam Khalid. Insights and strategies for an autonomous vehicle with a sensor fusion innovation: A fictional outlook. *IEEE Access*, 8:1–14, 08 2020.
- [2] WIKIPEDIA. Geometric transformation. https://en.wikipedia.org/wiki/Geometric_transformation.
- [3] MathWorks. What is camera calibration? <https://www.mathworks.com/help/vision/ug/camera-calibration.html>.
- [4] open3d. Global registration. http://www.open3d.org/docs/release/tutorial/pipelines/global_registration.html.
- [5] WIKIPEDIA. Adjacency matrix. https://en.wikipedia.org/wiki/Adjacency_matrix.
- [6] PRECEDENCE RESEARCH. Autonomous vehicles market size. <https://www.precedenceresearch.com/autonomous-vehicle-market>.
- [7] WIRED.com. Gm and cruise's self-driving car: Just add software. <https://www.wired.com/story/gm-cruise-generation-3-self-driving-car/>.
- [8] M V Rajasekhar and Anil Kumar Jaswal. Autonomous vehicles: The future of automobiles. In *2015 IEEE International Transportation Electrification Conference (ITEC)*, pages 1–6, 2015.
- [9] Alain Allouis, Iyad Dayoub, and Soumaya Cherkaoui. On 5g-v2x use cases and enabling technologies: A comprehensive survey. *IEEE Access*, 9:107710 – 107737, 07 2021.
- [10] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, Dec 1943.
- [11] WIKIPEDIA. Perceptron. <https://en.wikipedia.org/wiki/Perceptron>.
- [12] F. Rosenblatt. *The Perceptron, a Perceiving and Recognizing Automaton Project Para*. Report: Cornell Aeronautical Laboratory. Cornell Aeronautical Laboratory, 1957.
- [13] Terence Sanger and Pallavi N. Baljekar. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408, 1958.
- [14] Y. Le Cun, B. Boser, J. S. Denker, R. E. Howard, W. Hubbard, L. D. Jackel, and D. Henderson. *Handwritten Digit Recognition with a Back-Propagation Network*, page 396–404. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1990.

- [15] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.
- [16] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *CoRR*, abs/1606.00915, 2016.
- [17] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017.
- [18] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020.
- [19] Dzmitry Bahdanau, Kyunghyun Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *ArXiv*, 1409, 09 2014.
- [20] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017.
- [21] P. Baldi. Gradient descent learning algorithm overview: a general dynamical systems perspective. *IEEE Transactions on Neural Networks*, 6(1):182–195, 1995.
- [22] Frank Rosenblatt. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. *American Journal of Psychology*, 76:705, 1963.
- [23] Kenji Hata and Silvio Savarese. Cs231a course notes 1: Camera models. https://web.stanford.edu/class/cs231a/course_notes/01-camera-models.pdf.
- [24] Qi Chen, Sihai Tang, Qing Yang, and Song Fu. Cooper: Cooperative perception for connected autonomous vehicles based on 3d point clouds. *CoRR*, abs/1905.05265, 2019.
- [25] Xiaoshui Huang, Guofeng Mei, Jian Zhang, and Rana Abbas. A comprehensive survey on point cloud registration. *CoRR*, abs/2103.02690, 2021.
- [26] P.J. Besl and Neil D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.
- [27] Yang Chen and Gérard Medioni. Object modelling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992. Range Image Understanding.
- [28] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Proceedings Third International Conference on 3-D Digital Imaging and Modeling*, pages 145–152, 2001.
- [29] Jaesik Park, Qian-Yi Zhou, and Vladlen Koltun. Colored point cloud registration revisited. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 143–152, 2017.

- [30] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In Knowledge Discovery and Data Mining, 1996.
- [31] Ricardo J. G. B. Campello, Davoud Moulavi, and Joerg Sander. Density-based clustering based on hierarchical density estimates. In Advances in Knowledge Discovery and Data Mining, pages 160–172, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [32] WIKIPEDIA. Graph. [https://en.wikipedia.org/wiki/Graph_\(discrete_mathematics\)](https://en.wikipedia.org/wiki/Graph_(discrete_mathematics)).
- [33] pygmtools. Graph matching library. <https://pygmtools.readthedocs.io/en/latest/guide/introduction.html>.
- [34] Junchi Yan, Xu-Cheng Yin, Weiyao Lin, Cheng Deng, Hongyuan Zha, and Xiaokang Yang. A short survey of recent advances in graph matching. Proceedings of the 2016 ACM on International Conference on Multimedia Retrieval, 2016.
- [35] Junchi Yan, Shuang Yang, and Edwin R. Hancock. Learning for graph matching and related combinatorial optimization problems. In International Joint Conference on Artificial Intelligence, 2020.
- [36] Minsu Cho, Jungmin Lee, and Kyoung Mu Lee. Reweighted random walks for graph matching. In Kostas Daniilidis, Petros Maragos, and Nikos Paragios, editors, Computer Vision – ECCV 2010, pages 492–505, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [37] H. W. Kuhn. The hungarian method for the assignment problem. Naval Research Logistics Quarterly, 2(1-2):83–97, 1955.
- [38] WIKIPEDIA. Assignment problem. https://en.wikipedia.org/wiki/Assignment_problem.
- [39] Evan Shelhamer, Jonathan Long, and Trevor Darrell. Fully convolutional networks for semantic segmentation. CoRR, abs/1605.06211, 2016.
- [40] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. CoRR, abs/1505.04597, 2015.
- [41] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick. Mask R-CNN. CoRR, abs/1703.06870, 2017.
- [42] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. CoRR, abs/1506.01497, 2015.
- [43] Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. CoRR, abs/1612.01105, 2016.
- [44] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. CoRR, abs/1706.05587, 2017.
- [45] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. CoRR, abs/1606.00915, 2016.

- [46] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, José M. Álvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *CoRR*, abs/2105.15203, 2021.
- [47] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip H. S. Torr, and Li Zhang. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. *CoRR*, abs/2012.15840, 2020.
- [48] Rui Zhang, Guangyun Li, Minglei Li, and Li Wang. Fusion of images and point clouds for the semantic segmentation of large-scale 3d scenes based on deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143:85–96, 2018. ISPRS Journal of Photogrammetry and Remote Sensing Theme Issue “Point Cloud Processing”.
- [49] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413, 2017.
- [50] Charles Ruizhongtai Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CoRR*, abs/1612.00593, 2016.
- [51] Hui Zhou, Xinge Zhu, Xiao Song, Yuexin Ma, Zhe Wang, Hongsheng Li, and Dahua Lin. Cylinder3d: An effective 3d framework for driving-scene lidar semantic segmentation. *CoRR*, abs/2008.01550, 2020.
- [52] Qing Li, Cheng Wang, Chenglu Wen, and Xin Li. Deepsir: Deep semantic iterative registration for lidar point clouds. *Pattern Recognition*, 137:109306, 2023.
- [53] Pengyu Yin, Shenghai Yuan, Haozhi Cao, Xingyu Ji, Shuyang Zhang, and Lihua Xie. Segregator: Global point cloud registration with semantic and geometric cues. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2848–2854, 2023.
- [54] Alexey Dosovitskiy, Germán Ros, Felipe Codevilla, Antonio M. López, and Vladlen Koltun. CARLA: an open urban driving simulator. *CoRR*, abs/1711.03938, 2017.
- [55] Yiming Li, Dekun Ma, Ziyan An, Zixun Wang, Yiqi Zhong, Siheng Chen, and Chen Feng. V2x-sim: Multi-agent collaborative perception dataset and benchmark for autonomous driving. *IEEE Robotics and Automation Letters*, 7(4):10914–10921, 2022.
- [56] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics yolov8. <https://github.com/ultralytics/yolov8>, 2023.
- [57] K. S. Arun, T. S. Huang, and S. D. Blostein. Least-squares fitting of two 3-d point sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9(5):698–700, 1987.
- [58] WIKIPEDIA. Kabsch algorithm. https://en.wikipedia.org/wiki/Kabsch_algorithm.
- [59] MMSegmentation Contributors. MMSegmentation: Openmmlab semantic segmentation toolbox and benchmark. <https://github.com/open-mmlab/mmsegmentation>, 2020.

Παράρτημα

Περιβάλλον Υλοποίησης και Εργαλεία

Στο παρόν κομμάτι εξετάζονται τα εργαλεία που αξιοποιήθηκαν, καθώς και το γενικό περιβάλλον υλοποίησης. Για να διευκρινιστεί περαιτέρω, όλες οι δοκιμές πραγματοποιήθηκαν σε ένα σύστημα εξοπλισμένο με επεξεργαστή Intel i5-8300H, 16GB RAM και κάρτα γραφικών Nvidia GTX 1050 Mobile. Αν και το εν λόγω σύστημα διαθέτει την απαιτούμενη υπολογιστική ισχύ για να εκτελέσει τα χρησιμοποιηθέντα μοντέλα τεχνητής νοημοσύνης, δεν αποτελεί την ιδανική επιλογή για την εκπαίδευσή τους. Αυτό οφείλεται στο γεγονός ότι η κάρτα γραφικών δεν διαθέτει το απαραίτητο αποθηκευτικό χώρο για την αποθήκευση των παραμέτρων των δικτύων, αλλά και των ίδιων των δεδομένων. Λόγω των περιορισμών που παρουσιάζει το υλικό του συγκεκριμένου συστήματος, αποφασίστηκε να πραγματοποιηθεί η εκπαίδευση των δικτύων στο Google Colab. Το Google Colab παρέχει μια δωρεάν υποδομή υπολογιστικού νέφους, η οποία είναι ιδανική για εκπαίδευτικούς σκοπούς.

Ένα κρίσιμο ρόλο έπαιξαν τα εργαλεία και οι open source βιβλιοθήκες που χρησιμοποιήθηκαν. Αρχικά όλη η εργασία αναπτύχθηκε εξολοκλήρου σε γλώσσα Python χρησιμοποιώντας το Anaconda σαν εικονικό περιβάλλον ανάπτυξής. Με αυτόν τον τρόπο ήταν εύκολη η δοκιμή και η αξιολόγηση διαφόρων βιβλιοθηκών. Το πιο σημαντικό ρόλο έπαιξε η βιβλιοθήκη MMSegmentation [59], η οποία είναι ένα open source framework για σημασιολογική κατάτμηση εικόνας με υψηλή απόδοση που βασίζεται στο PyTorch. Ακόμα το συγκεκριμένο framework προσφέρει έτοιμες υλοποιήσεις δημοφιλών μεθόδων και αρχιτεκτονικών για εφαρμογές σημασιολογικής κατάτμησης ενώ παράλληλα παρέχει μία εύχρηστη διεπαφή για την εκπαίδευση και την αξιολόγηση των δικτύων αυτών.

Εκτός από αυτό το framework εξίσου σημαντικό ρόλο έπαιξε και η βιβλιοθήκη Pygmitools [33], Python Graph Matching Tools. Είναι μια βιβλιοθήκη που παρέχει μια ποικιλία εργαλείων για την επίλυση προβλημάτων αντιστοίχησης γράφων, δηλαδή έκτος από παραδοσιακούς αλγορίθμους υποστηρίζει και πιο καινοτόμες προτάσεις που βασίζονται σε μεθόδους βαθιάς μάθησης. Ακόμα όλη η αρχιτεκτονική της βιβλιοθήκης παρέχει μια τυποποιημένη διεπαφή με τους αλγορίθμους κάνοντας εύκολη και την αξιολόγηση των αποτελεσμάτων αλλά και την γενικότερη ανάπτυξη ενός προγράμματος βασισμένο σε αυτή τη βιβλιοθήκη.

Τέλος, για την οπτικοποίηση των αποτελεσμάτων χρησιμοποιείται η βιβλιοθήκη Open3D αλλά και η Matplotlib. Και οι δύο βιβλιοθήκες χρησιμοποιούνται ευρέως για την οπτικοποίηση δεδομένων, αλλά η κάθε μία για διαφορετικό σκοπό. Από τη μία πλευρά, η Open3D έχει σχεδιαστεί ειδικά για την απεικόνιση τρισδιάστατων δεδομένων, όπως νέφη σημείων, τρισδιάστατα μοντέλα και εικόνες RGBD. Παρέχει ένα ισχυρό σύνολο εργαλείων για τη φόρτωση, την επεξεργασία και την απόδοση τρισδιάστατων δεδομένων και μπορεί επίσης να χρησιμοποιηθεί για την εκτέλεση διαφόρων λειτουργιών τρισδιάστατων δεδομένων, όπως ευθυγράμμιση, κατάτμηση και ανακατασκευή. Από την άλλη πλευρά, η Matplotlib είναι μια βιβλιοθήκη δισδιάστατης σχεδίασης για γενική χρήση που παρέχει ένα πλούσιο σύνολο χαρακτηριστικών για

τη δημιουργία προσαρμοσμένων γραφημάτων ενώ παράλληλα μπορεί να χρησιμοποιηθεί για την απεικόνιση μιας μεγάλης ποικιλίας τύπων δεδομένων, συμπεριλαμβανομένων αριθμητικών δεδομένων αλλά και εικόνων.

Διαθεσιμότητα κώδικα

Οι οδηγίες για την εγκατάσταση και ο αναλυτικός κώδικας που συνδέονται με την παρούσα διπλωματική βρίσκονται διαθέσιμοι στο προφίλ μου στο GitHub και ειδικότερα στο αποθετήριο που ονομάζεται “PC_Registration-and-Collaborative_Vehicles”¹. Μέσα σε αυτό το αποθετήριο, υπάρχουν λεπτομερείς οδηγίες για την εγκατάσταση των απαραίτητων προγραμμάτων και ένας οδηγός ανάλυσης που στοχεύει στην επίδειξη των δυνατοτήτων και των αποτελεσμάτων της παρούσας διπλωματικής.

¹https://github.com/K0stasCH/PC_Registration-and-Collaborative_Vehicles

