ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

«ΕΚΤΙΜΗΣΗ ΒΑΘΟΥΣ ΜΕ ΤΗΝ ΧΡΗΣΗ ΔΕΔΟΜΕΝΩΝ ΑΠΟ ΔΥΟ ΚΑΜΕΡΕΣ»

Του φοιτητή
Τετεπουλίδη Δημήτριου
Αρ. Μητρώου: 185286

Επιβλέπων
Διαμαντάρας Κωνσταντίνος
Καθηγητής

06/09/2023

Τίτλος Π.Ε.: Πρόβλεψη βάθους με την χρήση δεδομένων από δύο κάμερες.
Κωδικός Δ.Ε. 23162
Ονοματεπώνυμο φοιτητή/των: Τετεπουλίδης Δημήτριος
Ονοματεπώνυμο εισηγητή: Διαμαντάρας Κωνσταντίνος
Ημερομηνία ανάληψης Δ.Ε. 20/1/2023
Ημερομηνία περάτωσης Δ.Ε. 06/09/2023

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

Πρόλογος

Πάντα είχα μία σχέση αγάπης – μίσους με τις προκλήσεις, μου άρεσε να προκαλώ τον εαυτό μου σε περιπτώσεις που μπορούσα να αποσκοπήσω κέρδος μέσα από αυτό, αλλά ταυτόχρονα απέφευγα «μάχες» που θεωρούσα αφαίμαξη χρόνου. Με αυτήν την σκέψη στο μυαλό, όταν ήρθε η ώρα να αποφασίσω πάνω στο τι θέλω να ασχοληθώ, επέλεξα τον πιο εξειδικευμένο και «τρομακτικό» τομέα που είχα την ευκαιρία να ασχοληθώ στην φοιτητική μου καριέρα, την Μηχανική Μάθηση. Έτσι κατέληξα να ψάχνω ενδιαφέρον προβλήματα που λύνει η Μηχανική Μάθηση, τα οποία έχουν σημαντική προστιθέμενη αξία. Έχοντας κοιτάξει έναν πληθώρα αριθμό προβλημάτων και θεμάτων που επιλύονται με Μηχανική Μάθηση, επέλεξα από την όραση υπολογιστή, την πρόβλεψη βάθους. Θεωρώντας το βάθος σαν δεδομένα πολύ σημαντικά για την αυτοματοποίηση της οδήγησης, αποφάσισα να καταλήξω στην «εκτίμηση» των εκάστοτε δεδομένων από μίμηση λειτουργιών του ματιού, δηλαδή την πρόβλεψη με την βοήθεια δεδομένων εικόνας.

Περίληψη

Η Τεχνητή Νοημοσύνη και οι εφαρμογές της πρόκειται να παίξουν πολύ σημαντικό ρόλο στον σχεδιασμό και στην ανάπτυξη των μελλόντων κοινωνιών. Ως παρακλάδι της επιστήμης των υπολογιστών σε συνδυασμό με τις υπόλοιπες επιστημονικές θεματικές που μπορεί να προσφέρει αυτή η επιστήμη, έχει καταφέρει και έχει δημιουργήσει εφαρμογές που μπορούν και βοηθούν πολλούς άλλους κλάδους της καθημερινότητάς μας, αλλά ταυτόχρονα επαυξάνουν με διάφορους τρόπους μικρές λειτουργίες που υπό διαφορετικές συνθήκες γινόντουσαν με ανθρώπινη παρέμβαση. Θεωρώντας την Μηχανική Μάθηση ως κομμάτι της ΤΝ (Τεχνητής Νοημοσύνης), με την χρήση αλγορίθμων και μοντέλων Μηχανικής Μάθησης έχει απλουστευτεί η υλοποίηση αλγορίθμων που λύνουν προβλήματα όρασης υπολογιστή. Με την εκτίμηση βάθους να είναι από τα σημαντικότερα προβλήματα που χρειάζονται επίλυση στον τομέα της «όραση υπολογιστή». Σωστές εκτιμήσεις μπορούν να αυτοματοποιήσουν διάφορες εργασίες της καθημερινότητας όπως είναι η αυτόματη οδήγηση βοηθούμενη από ΤΝ .Άλλες εφαρμογές που μπορεί να βοηθήσει η εκτίμηση βάθους είναι ο καθαρισμός θολωμένων σημείων εικόνας, η βελτίωση “rendering” τρισδιάστατων σκηνών, η Ρομποτική, η Ιατρική αλλά και διάφορες θεματικές που έχουν να κάνουν με γραφικά υπολογιστών.

Οι υλοποιήσεις και οι δοκιμές που θα παρουσιαστούν στην εργασία, είχαν ως στόχο την εκτίμηση του βάθους σε περιβάλλον εξωτερικού χώρου με το σημείο θέασης να μοιάζει στο σημείο θέασης που έχει ένας οδηγός αυτοκινήτου. Για την επίτευξη αυτού του στόχου χρησιμοποιήθηκαν διάφορα μοντέλα εκπαιδευμένα σε διαφορετικά datasets (σύνολα δεδομένων) που αντιπροσωπεύουν διαφορετικής φύσης δεδομένα (εσωτερικού χώρου, εξωτερικού χώρου, ψηφιακά αντικείμενα κ.α.). Κάθε μοντέλο που υλοποιήθηκε, δοκιμάστηκε στο dataset KITTI που αποτελεί ένα από τα σημαντικότερα datasets για μοντέλα μηχανικής μάθησης που λύνουν προβλήματα που αφορούν την όραση υπολογιστή. Στην εργασία θα παρουσιαστούν τα αποτελέσματα αυτών των μοντέλων Μηχανικής Μάθησης πάνω σε διάφορες δοκιμές καθώς και οι τεχνολογίες που χρησιμοποιήθηκαν για την διεκπεραίωση των δοκιμών. Τέλος θα παρουσιαστούν κομμάτια κώδικα αλλά και οδηγίες για την χρήση των προγραμμάτων και αλγορίθμων.

Depth Estimation with the use of two cameras

Dimitrios Tetepoulidis

(στην αγγλική γλώσσα)

Abstract

Artificial Intelligence and its applications seem to play a crucial role in designing and developing future societies. Being a part of Computer science (CS), in collaboration with other specialization fields of CS, AI has managed to create applications and programs that can assist other scientific fields as well as help with regular daily tasks where in other cases human intervention would be necessary. Considering Machine Learning (ML) as part of the Artificial Intelligence specialization, with the usage of ML and Algorithms and Models, the implementation of Computer Vision-related problems has been significantly simplified, with Depth Estimation being one of the most crucial problems in need of a solution. Useful estimations could automate a variety of daily tasks and jobs like Automated Driving. Other applications that depend on solid depth estimation data include smoothing blurred parts of an image, better rendering of 3D scenes, robotics, medical applications and computer graphics related tasks.

The implementations and tests presented in this thesis were oriented in predicting depth in an exterior space with the cameras' point of view mimicking the driver's point of view. To achieve this goal a plethora of ML models were used, trained in a variety of datasets which represent different kinds of data (interior, exterior, computer generated etc.). Every model implemented in this paper, was tested in KITTI dataset which is considered one of the most important datasets for solving computer vision problems. On this paper results from all evaluated models will be presented as well as the technologies used and code snippets from the models.

Ευχαριστίες

Αρχικά, θα ήθελα να ευχαριστήσω τους ανθρώπους που μου έδωσαν την ευκαιρία να ασχοληθώ με ένα τόσο ενδιαφέρον θέμα. Αναφέρομαι φυσικά στον καθηγητή Κωνσταντίνο Διαμαντάρα και τους φοιτητές και απόφοιτους της ομάδα του “DRAIVE” που μου παρείχαν πρόσβαση σε μηχάνημα για να μπορέσω να τρέξω τις δοκιμές που χρειαζόμουν για την εργασία. Εντός της ομάδας θέλω να ευχαριστήσω συγκεκριμένα τον απόφοιτο του τμήματος Γεώργιο Μακρή που με κατεύθυνε κατά την διάρκεια της διεκπεραίωσης της εργασίας. Επιπλέον θέλω να ευχαριστήσω ανθρώπους εκτός της ομάδας DRAIVE, τον φίλο και συνάδελφο Ευάγγελο Παναγιωτόπουλο τελειόφοιτο του τμήματος Μηχανικών Πληροφορικής Τ.Ε., καθώς και τους γονείς μου, που με υποστήριξαν κατά την διάρκεια της εκπόνησης της εργασίας.

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Συντομογραφίες

3Δ	Τριών Διαστάσεων
ΔΙΠΑΕ	Διεθνές Πανεπιστήμιο Ελλάδος
Π.Ε.	Πτυχιακή Εργασία
Τ.Ν.	Τεχνητή Νοημοσύνη
Abs Rel	Absolute Relative error
ACV	Attention Concatenation Volume
ACVNet	Attention Concatenation Volume Network
AnyNet	Anytime Stereo Network
API	Application Programming Interface
CN	Convolutional Neural
CNN	Convolutional Neural Network
CPU	Central Processing Unit
C.S.	Computer Science
CUDA	Compute Unified Device Architecture
D1-bg	D1 - background
D1-fg	D1 - foreground
DSLR	Digital Single-Lens Reflex
ECC	Error Correcting Code
FN	False Negative
FP	False Positive
GB	Gigabyte
GPU	Graphics Processing Unit
GwcNet	Group-wise Correlation Stereo network
HDD	Hard Disk Drive
HITNet	Hierarchical Iterative Tile refinement Network
KITTI	Karlsruhe Institute of Technology and Toyota Technological Institute
LiDAR	Light Detection And Ranging
MAE	Mean Absolute Error
M.L.	Machine Learning
MLP	Multi-Layer Perceptron

MP4	MPEG-4 Part 14
MPEG	Motion Picture Experts Group
MSE	Mean Square Error
Noc	No occlusions
NumPy	Numerical Python
OpenCL	Open Computing Language
OpenCV	Open-Source Computer Vision Library
PCA	Principal Component Analysis
PNG	Portable Network Graphics
PSMNet	Pyramid Stereo Matching Network
PWCNet	Pyramid, Warping and Cost volume Network
RAM	Random-Access Memory
RDVO	Rigid-aware Direct Visual Odometry
ReLU	Rectified Linear Network
ResNet	Residual Network
RL	Reinforcement Learning
RMSE	Root Mean Square Error
RMSE log	Root Mean Square Logarithmic Error
SEO	Search Engine Optimization
SOTA	State-Of-The-Art
SPP	Spatial Pyramid Pooling
Sq Rel	Square Relative Error
SSD	Solid-State Drive
SVD	Singular Value Decomposition
TB	Terabyte
TN	True Negative
TP	True Positive
TPU	Tensor Processing Unit
UnOS	Unified Unsupervised Optical-flow and Stereo-depth Estimation
VGG	Visual Geometry Group
VRAM	Video Random-Access Memory

Κεφάλαιο 1ο: Εισαγωγή

Η εκτίμηση βάθους είναι ένα θέμα της υπολογιστικής όρασης που έχει απασχολήσει την επιστημονική κοινότητα για πολλές δεκαετίες. Τα τελευταία χρόνια έχει υπάρξει μία άνοδος όσον αφορά την έρευνα σχετικά με την συγκεκριμένη εργασία. Αυτό οφείλεται περισσότερο στις εφαρμογές που μπορεί να χρησιμοποιηθούν εκτιμήσεις όσον αφορά το βάθος, καθώς τα τελευταία χρόνια υπάρχει μία αυξημένη ζήτηση σε εφαρμογές που εκμεταλλεύονται την πληροφορία του βάθους.

Η πληροφορία του βάθους μπορεί να φανεί χρήσιμη σε διάφορες εφαρμογές. Συγκεκριμένα, τα δεδομένα του βάθους βρίσκουν χρησιμότητα σε Ιατρικές εφαρμογές, την αυτόνομη οδήγηση, για εργοστασιακή χρήση, την ρομποτική κ.α.. Υπάρχουν διάφοροι τρόποι που μπορεί να χρησιμοποιηθούν τα δεδομένα του βάθους σε κάποια εφαρμογή, αλλά οι περισσότεροι τρόποι συνδυάζονται με μεθόδους και αλγόριθμους τεχνητής νοημοσύνης και μηχανικής μάθησης.

Διάφορες τεχνικές που προσπαθούν να εκτιμήσουν το βάθος ασχολούνται με την χρήση αισθητήρων και καμερών. Οι αισθητήρες βάθους μπορούν και δημιουργούν δεδομένα βάθους που πλησιάζουν τις αληθινές τιμές. Ενώ με την χρήση καμερών και των νευρωνικών δικτύων, μπορούμε επίσης να εκτιμήσουμε αντίστοιχα το βάθος με ικανοποιητική ευκρίνεια.

Σε αυτήν την εργασία θα εξεταστούν και θα παρουσιαστούν διάφοροι μέθοδοι μηχανικής μάθησης που επιλύουν το συγκεκριμένο πρόβλημα. Επιπλέον θα γίνει μια εξέταση των επιστημονικών πεδίων με τα οποία ασχολούνται αυτοί οι μέθοδοι, όπως και επίσης και μια εξέταση στο ευρύ φάσμα της όρασης υπολογιστή. Τέλος θα γίνει μία σύγκριση μεταξύ των μεθόδων και θα βγουν συμπεράσματα σχετικά με την απόδοσή τους.

1.1 Πώς εκτιμούμε το βάθος

Τα ανθρώπινα μάτια εκτιμούν το βάθος συγκρίνοντας τις εικόνες που παρατηρούν από το δεξί και το αριστερό μάτι. Αυτή η διαφορά απόστασης διαφόρων σημείων κλειδιών που παρατηρούμε με το δεξί και το αριστερό μάτι είναι αρκετή για να αποκτήσουμε μία υποτυπώδη αίσθηση του βάθους. Σε συνδυασμό με την μεταβλητότητα που μπορούν να έχουν τα μάτια μας ως προς την κατεύθυνση και την εμπειρία που έχουμε αποκτήσει «παρατηρώντας» πράγματα μάς δίνει την δυνατότητα να έχουμε μία καλή αίσθηση του βάθους [1].

1.2 Υπολογιστική εκτίμηση βάθους

Οι οθόνες που χρησιμοποιούνται στη σημερινή εποχή είναι δύο διαστάσεων, κάθε φωτογραφία και βίντεο της πραγματικότητας απεικονίζει έναν τρισδιάστατο κόσμο. Συνήθως η καταγραφή, αποθήκευση και παρουσίαση των συγκεκριμένων εικόνων γίνεται σε αυτές τις δύο διαστάσεις έχοντας σαν αποτέλεσμα να χάνουμε σημαντική πληροφορία όπως αυτή του βάθους. Για τις περισσότερες περιπτώσεις η δισδιάστατη απεικόνιση θεωρείται χρήσιμη ή και βέλτιστη, αλλά υπάρχουν περιπτώσεις στις οποίες που υπάρχει ανάγκη για την χρήση πληροφοριών που θα ήταν αδύνατη να επιτευχθεί με την χρήση δισδιάστατων εικόνων, όπως η πληροφορία του βάθους [1].

Η εκτίμηση βάθους σαν πρόβλημα είναι η εργασία της δημιουργίας δεδομένων που εκτιμούν το βάθος ενός χώρου μέσα από δεδομένα απλών φωτογραφιών [2]. Λόγω της μεγάλης ποικιλίας εφαρμογών χρησιμοποιούνται οι εκτιμήσεις, τα τελευταία χρόνια έχει προσελκύσει μεγάλη προσοχή όσον αφορά τα προβλήματα της όρασης υπολογιστή [3]. Σε πρώιμες έρευνες οι εκτιμήσεις βάθους που παράγονταν

από τις εκάστοτε έρευνες είχαν σαν αποτέλεσμα να έχουν χαμηλή ανάλυση εικόνας λόγω των διαδικασιών που εκτελούσαν τα Νευρωνικά Δίκτυα με τα οποία γινόταν η υλοποίηση. Συγκεκριμένα η υλοποίηση γινόταν με την χρήση Convolutional Neural Network (CNN).

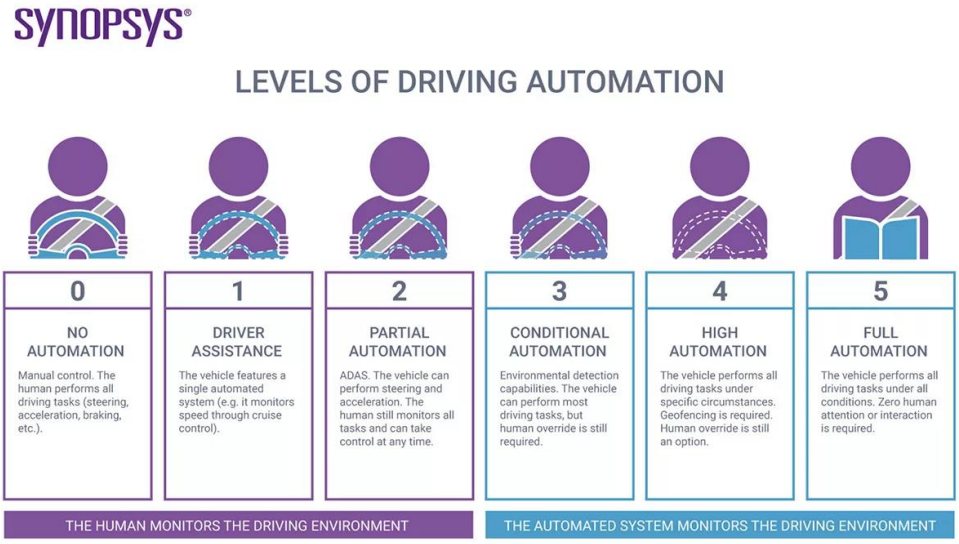
1.3 Μέθοδοι εκτίμησης βάθους

Διάφορα μοντέλα και προσεγγίσεις έχουν δημιουργηθεί για την επίλυση του συγκεκριμένου προβλήματος με τους πιο σημαντικούς να ονομάζονται Monocular Depth Estimation, Stereo Depth Estimation και αισθητήρων LiDAR. Το Stereo Depth Estimation αναφέρεται στην χρήση δύο καμερών για την εκτίμηση του βάθους και οι αισθητήρες LiDAR που παίρνουν δεδομένα από το περιβάλλον τους με την χρήση αισθητήρων φωτός. Αντιθέτως, η υλοποίηση του Monocular Depth Estimation δεν χρησιμοποιεί αισθητήρες παρά μόνο μία κάμερα.

Σαν βέλτιστη λύση για την εκτίμηση βάθους έχει θεωρηθεί ο συνδυασμός χρήσης αισθητήρων LiDAR και η υλοποίηση CNN μοντέλων που υλοποιούν Stereo Depth Estimation [4], με τα αρνητικά αυτού του συνδυασμού να είναι η δυσχρηστία τους σε περίπλοκα περιβάλλοντα, η πολυπλοκότητα των αλγορίθμων αλλά και το κόστος του υλικού (hardware) που απαιτείται. Αντιθέτως, η χρήση μίας μόνο κάμερας έχει μικρό κόστος και είναι πιο προσαρμόσιμη σε περίπλοκα περιβάλλοντα [5].

1.4 Εφαρμογές

Τα σωστά δεδομένα σχετικά με το βάθος ενός χώρου έχουν πολλές εφαρμογές. Μπορούν να χρησιμοποιηθούν σε εφαρμογές Επαυξημένης και Εικονικής Πραγματικότητας, στη Ρομποτική, την επεξεργασία εικόνας [1] αλλά και στην αυτόνομη οδήγηση. Αυτόνομη οδήγηση θεωρείται όταν ένα αυτοκίνητο είναι ικανό να αντιλαμβάνεται το περιβάλλον του και να δρα εντός αυτού χωρίς να χρειάζεται να παρέμβει ο ανθρώπινος παράγοντας [6].



Σχήμα 1.1: Επίπεδα αυτοματοποίησης οδήγησης σύμφωνα με την Synopsys [6].

1.4.1 Αυτόνομη οδήγηση

Η αυτόνομη οδήγηση μπορεί να διαχωριστεί σε 6 επίπεδα, με το επίπεδο 0 να αναφέρεται σε ένα όχημα που τον έλεγχο του αυτοκινήτου (επιτάχυνση, φρένα κτλ.) τον έχει μόνο ο άνθρωπος. Το επίπεδο 1

αναφέρεται σε τεχνολογίες που βοηθούν τον οδηγό σε συγκεκριμένες εργασίες οδήγησης. Γνωστό παράδειγμα σε σύγχρονα αυτοκίνητα, είναι η χρήση της τεχνολογίας cruise control. Στο επίπεδο 2 αναφερόμαστε σε μερική αυτοματοποίηση ενεργειών της οδήγησης, όπως στην επιτάχυνση ή την δυνατότητα του αυτοκινήτου να στρίβει από μόνο του. Στο επίπεδο 3 αναφερόμαστε σε αυτοματοποίηση υπό συνθήκες. Η συγκεκριμένη αυτοματοποίηση δίνει την δυνατότητα στο αυτοκίνητο να καταλαβαίνει το περιβάλλον του και να μπορεί να εκτελεί τις περισσότερες ενέργειες εντός αυτού. Στο επίπεδο 4 το όχημα θα μπορεί να εκτελεί όλες τις ενέργειες που χρειάζονται να γίνουν σε ένα αυτοκίνητο, αλλά η εκτέλεσή τους θα γίνεται υπό συνθήκες. Τέλος στο επίπεδο 5 αναφερόμαστε σε ένα σημείο στο οποίο δεν θα υπάρξει καμία ανάγκη ανθρώπινου παράγοντα για να μπορέσει να διενεργήσει το όχημα. Είναι σημαντικό να αναφέρουμε πως από το επίπεδο 0 μέχρι το επίπεδο 2, δεδομένα σχετικά με το περιβάλλον μπορεί να λάβει και να επεξεργαστεί μόνο ο οδηγός, ενώ η αυτοματοποίηση υπάρχει μόνο για να υποστηρίζει τις αποφάσεις του οδηγού. Επιπροσθέτως, στα επίπεδα 3 και 4 υπάρχει ακόμα η δυνατότητα στον οδηγό να παρακάμψει τις αυτόματες λειτουργίες και να πάρει τον έλεγχο του οχήματος [6].

1.5 Υλοποιήσεις

Το πρόβλημα της εκτίμησης βάθους έχει μελετηθεί σε μεγάλο βαθμό ανά τα χρόνια. Οι υλοποιήσεις περιλαμβάνουν την χρήση πολλαπλών δεδομένων εστίασης από δεδομένα εικόνας [7], τη χρήση αισθητήρων αλλά και την εκτίμηση βάθους με τη χρήση νευρωνικών δικτύων.

Η χρήση δεδομένων από μία κάμερα θεωρείται μία καλή αρχική προσέγγιση στην επίλυση του προβλήματος του βάθους λόγω του μικρού κόστους αλλά και της ευχρηστίας τους [8]. Οι περισσότερες υλοποιήσεις για την εκτίμηση βάθους από δεδομένα μίας κάμερας συνήθως χρησιμοποιούν την βοήθεια των νευρωνικών δικτύων. Οι παραδοσιακές μέθοδοι Μηχανικής Μάθησης που χρησιμοποιούνται για την εκτίμησή του χωρίζονται σε δύο κατηγορίες.

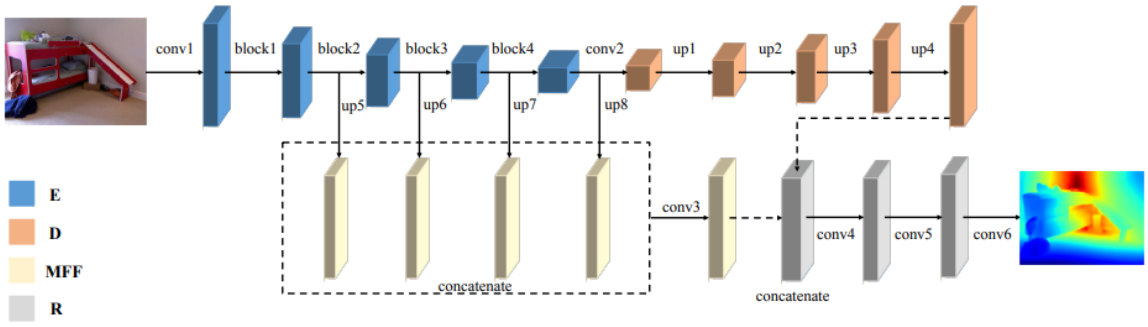
- Παραμετρική εκπαίδευση Νευρωνικών Δικτύων
- Μη παραμετρική εκπαίδευση Νευρωνικών Δικτύων

Στην παραμετρική εκπαίδευση, οι μέθοδοι που χρησιμοποιούνται συλλέγουν δεδομένα για τις παραμέτρους κατά την διάρκεια της εκπαίδευσής τους. Αυτή η μεθοδολογία θεωρείται αναγνωρισμένη και χρησιμοποιείται από πολλές μεθόδους για την εκτίμηση βάθους από μία κάμερα [8]. Σε συνδυασμό με τα δεδομένα εκπαίδευσης αλλά και τις προβλέψεις, με την χρήση Reinforcement Learning (RL) μεθόδων, το πρόβλημα της εκτίμησης βάθους γίνεται όλο και πιο διαχειρίσιμο. Ενώ για τις μεθόδους που βασίζονται στην μη παραμετρική εκπαίδευση χρησιμοποιούνται, δημιουργούν μια εκτίμηση βάθους χρησιμοποιώντας δεδομένα από την κάμερα, και συγκρίνοντάς τα με δεδομένα από ήδη προϋπάρχοντα datasets.

Με την ραγδαία ανάπτυξη των Convolutional Neural Networks τα τελευταία χρόνια, νέοι τρόποι προσέγγισης αυτής της μεθοδολογίας έχουν δημιουργηθεί και βασίζονται πάνω στην βαθιά μάθηση (Deep Learning). Αυτές οι μέθοδοι κατηγοριοποιούνται σε Supervised Learning (μάθηση με επίβλεψη) και Self-supervised Learning (Αυτό-επιβλεπόμενη μάθηση). Οι μέθοδοι που χρησιμοποιούν μάθηση με επίβλεψη περιλαμβάνουν μία συνάρτηση κόστους όπου συγκρίνουν την αρχική εικόνα και το εκτιμώμενο βάθος. Στην συνέχεια η διαφορά αθροίζεται στην συνάρτηση κόστους. Αυτές οι μέθοδοι συνήθως πετυχαίνουν μεγαλύτερη ακρίβεια στις εκτιμήσεις τους συγκριτικά με τις μη επιβλεπόμενες. Εναλλακτικά οι μέθοδοι αυτό-επιβλεπόμενης μάθησης χρησιμοποιούνται για να παρακάμψουν τον

περιορισμό που έχει η μάθηση με επίβλεψη. Υπάρχουν δύο κύριοι τρόποι εκτίμησης βάθους που χρησιμοποιούν την αυτό-επιβλεπόμενη μάθηση, η Monocular Depth Estimation & Stereo Depth Estimation.

Στην υλοποίηση αλγορίθμων για εκτίμηση βάθους, έχουν προταθεί πολλές ιδέες για την προσέγγιση με την χρήση δύο καμερών. Οι περισσότεροι αλγόριθμοι που βασίζονται σε δεδομένα εικόνων από ζευγάρια προσπαθούν να συγκρίνουν τις ανισότητες σε σημεία «κλειδιά» μεταξύ των ζευγών των εικόνων. Ωστόσο όμως αυτή η προσέγγιση βασίζεται στην τριγωνική φύση που έχει εκτίμηση του βάθους και επηρεάζεται σε μεγάλο βαθμό από την γραμμή βάσης της εικόνας, δίνοντας ανακριβή αποτελέσματα όταν αυτή η γραμμή διαφέρει [9]. Τέτοιες μέθοδοι βασίζονται πολύ στην φύση των δεδομένων που χρησιμοποιούνται για την εκπαίδευση των νευρωνικών δικτύων, και για αυτόν τον λόγο είναι δύσκολο να προσαρμοστούν σε διαφορετικής φύσεως περιβάλλοντα [10].



Σχήμα 1.2: Το διάγραμμά του προτεινόμενου δικτύου των Junjie Hu, Mete Ozay, Yan Zhang, Takayuki Okatani [11].

1.5.1 Υλοποιήσεις Monocular Depth Estimation

Η υλοποίηση των [Junjie Hu, Mete Ozay, Yan Zhang, Takayuki Okatani] [11] έχει δημιουργήσει μία αρχιτεκτονική από έναν συνδυασμό CN δικτύων χωρισμένη σε τέσσερα κομμάτια (Σχήμα 1.2): ένα κωδικοποιητή για εξαγωγή χαρακτηριστικών πολλαπλών κλιμάκων, ένα αποκωδικοποιητή [E] που χρησιμοποιείται για την αποκωδικοποίηση [D] των εξαγόμενων χαρακτηριστικών, ένα κομμάτι πολλών κλιμάκων που ενώνει τα παραγόμενα χαρακτηριστικά που έχουν δημιουργηθεί από τον κωδικοποιητή σε διάφορες αναλύσεις [MFF] και την μονάδα βελτιστοποίησης [R], που χρησιμοποιεί τα δεδομένα από την δεύτερη και την τρίτη μονάδα, για να δημιουργήσει την τελική εκτίμηση.

Η αξία των εκτιμήσεων βάθους συνήθως καθορίζεται με την χρήση της συνάρτησης κόστους, που για τις περισσότερες περιπτώσεις ορίζεται με τον τύπο

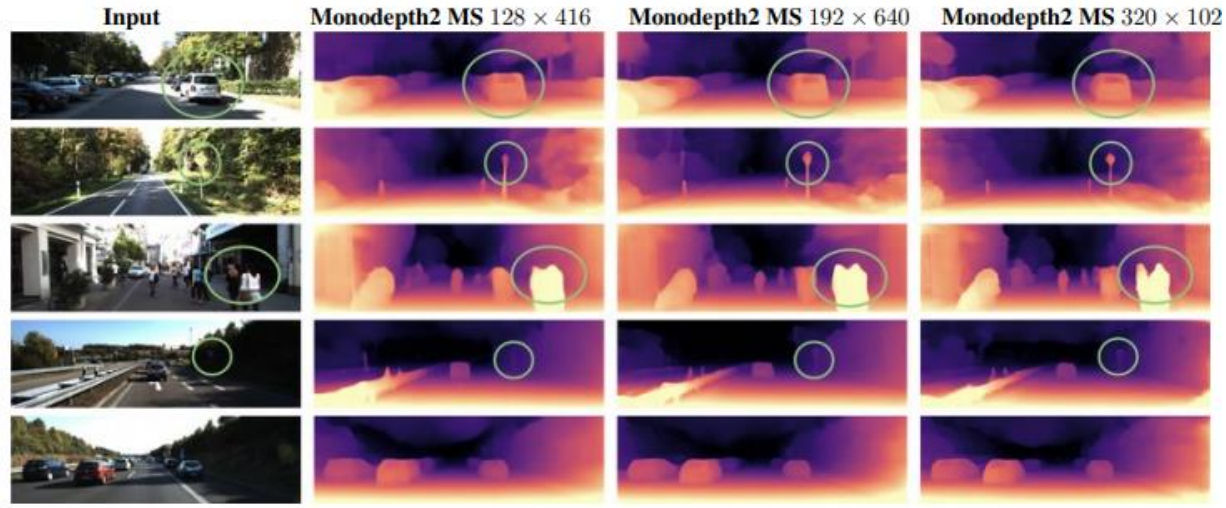
$$l_1 = \frac{1}{n}\sum_{i=1}^{n} e_i \qquad (1.1)$$

όπου το $e_i = |d_i - g_i|$, και το d_i να αντιπροσωπεύει την εκτίμηση του βάθους για την εικόνα i και το g_i να αντιπροσωπεύει το πραγματικό βάθος της εκάστοτε εικόνας. Ωστόσο η συγκεκριμένη συνάρτηση

θεωρείται αδύναμη για την εκτίμηση του κόστους όσον αφορά το βάθος λόγω της φύσεως του προβλήματος και της μεταβλητότητας που έχουν οι πραγματικές τιμές σύγκρισης απόστασης [11].

Άλλη πρόταση βασισμένη στην μεθοδολογία του Monocular Depth Estimation με προσέγγιση αυτό-επιβλεπόμενης μάθησης είναι το Monodepth2 [12] που δέχεται σαν δεδομένα εισόδου μία φωτογραφία I και δημιουργεί σαν δεδομένα εξόδου την εκτίμηση D. Το μοντέλο προτείνεται ως μια εναλλακτική λύση για τις μεθόδους που χρησιμοποιούν δύο κάμερες, λόγω της πολυπλοκότητας και της διαλειτουργικότητας που έχει η αρχιτεκτονική του μοντέλου. Το συγκεκριμένο μοντέλο θεωρείται πως έχει πετύχει SOTA (state-of-the-art) αποτελέσματα στο πρόβλημα της εκτίμησης βάθους βάσει των τριών σημαντικότερων συνεισφορών του.

- Το ελάχιστο κόστος επαναπροβολής που υπολογίζεται για κάθε pixel.
- Η λειτουργία να αγνοεί το μοντέλο ακίνητα ή «μπερδεμένα» pixel.
- Η μέθοδος δειγματοληψίας διαφόρων αναλύσεων και κλιμάκων που χρησιμοποιεί.



Σχήμα 1.3: Εκτιμήσεις βάθους μοντέλου Monodepth2 σε διάφορες αναλύσεις πάνω στο dataset KITTI [13]

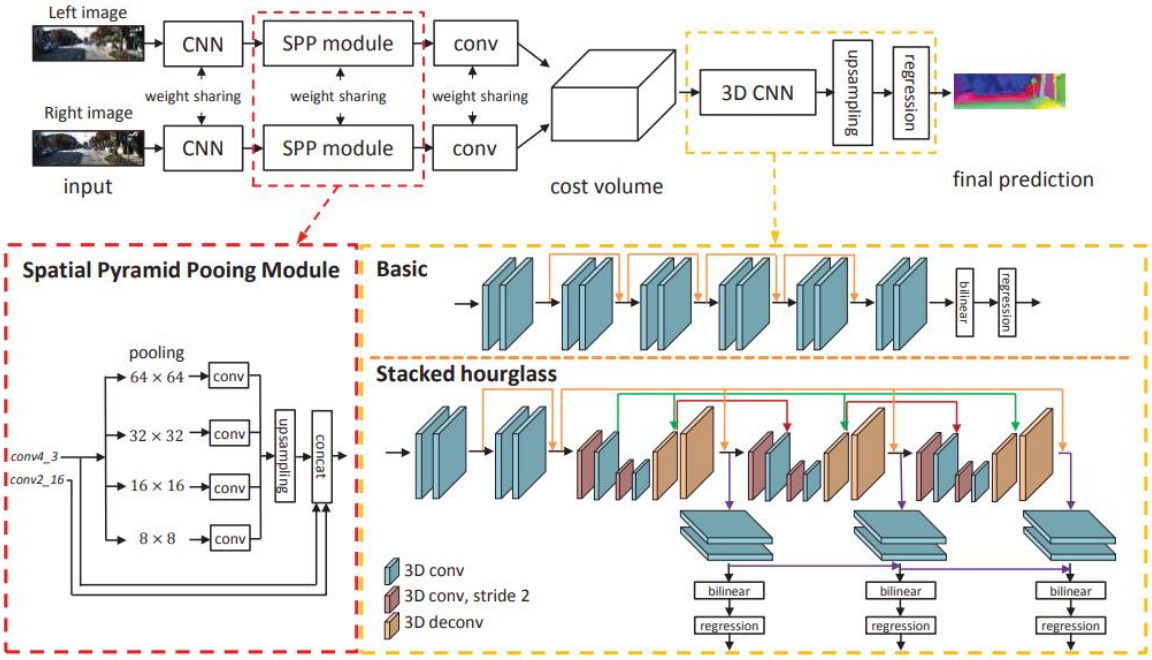
Εν κατακλείδι παρατηρούμε ότι περεταίρω υλοποιήσεις Monocular Depth Estimation βασίζονται κατά κύριο λόγο στην χρήση CN δικτύων [5] [14] αλλά και στην δημιουργία αρχιτεκτονικών πολλών στρωμάτων και περίπλοκων μοντέλων. Για την προετοιμασία των δεδομένων χρησιμοποιούν τεχνικές Data Augmentation (επαύξησης δεδομένων), και για την διατίμηση του κάθε μοντέλου, επιστρατεύουν προσαρμοσμένες στις ανάγκες των μοντέλων συναρτήσεις κόστους. Επιπλέον τα πειράματά τους γίνονται εντός ελεγμένων προδιαγραφών και περιβαλλόντων με την χρήση συγκεκριμένων συνόλων δεδομένων όπως αυτό του KITTI αλλά και διαφόρων άλλων όπως το NYU Depth V2 [15], ένα Dataset που απεικονίζει φωτογραφίες εσωτερικού χώρου.

1.5.2 Υλοποιήσεις Stereo Depth Estimation

Μια διαφορετική μεθοδολογία για την εκτίμηση βάθους όπως προαναφέρθηκε νωρίτερα στο κείμενο, είναι με την χρήση δεδομένων από δύο κάμερες, το “stereo depth estimation”. Υπάρχουν διάφορες υλοποιήσεις για stereo depth estimation. Αυτές βασίζονται κατά κύριο λόγο στην χρήση CN δικτύων αλλά και στην τριγωνομετρική έννοια μίας εικόνας συγκρίνοντας την διαφορά που υπάρχει στα κύρια σημασιολογικά σημεία κάθε ζευγαριού εικόνων.

Τα μοντέλα που χρησιμοποιούν την μέθοδο Stereo Depth Estimation, συνήθως χρησιμοποιούν έναν συνδυασμό από διάφορα σύνολα δεδομένων. Τα σύνολα δεδομένων που παίζουν πρωταγωνιστικό ρόλο στο Stereo Depth Estimation είναι το KITTI, το SceneFlow [16], το Middleburry και το ETH3D [17]. Παρακάτω στην εργασία θα παρουσιαστούν υλοποιήσεις από τρία μοντέλα, το UnOS, το HITNET και το ACVNet της υλοποίησης FAST-ACVnet. Μαζί με τις υλοποιήσεις θα παρουσιαστούν αποσπάσματα δοκιμών αλλά και τα αποτελέσματα στο dataset διατίμησης του KITTI. Οι υλοποιήσεις είναι γραμμένες σε κώδικα της γλώσσας Python, η κάθε μία σε διαφορετική έκδοσή της. Για την δημιουργία και την διαχείριση των νευρωνικών μοντέλων χρησιμοποιήθηκαν δύο framework της γλώσσας python, το TensorFlow και το PyTorch.

Ένα από τα σημαντικότερα μοντέλα για Stereo Depth Estimation είναι το PSMNet (Pyramid Stereo Matching Network). Το PSMNet είναι ένα μοντέλο που εκμεταλλεύεται τις δυνατότητες των CN δικτύων και σε συνδυασμό με διάφορες υπολογιστικές τεχνικές που χρησιμοποιούνται στην επεξεργασία εικόνας, επεκτείνει τις δυνατότητες του μοντέλου σε επίπεδο pixel για διαφορετικές κλίμακες οπτικών πεδίων [18].



Σχήμα 1.4: Προεπισκόπηση της αρχιτεκτονικής του PSMNet [18].

Με εξαίρεση την περίπλοκη αρχιτεκτονική του μοντέλου, έχουν προταθεί διαφοροποιημένες εκδοχές της συνάρτησης του κόστους και της συνάρτησης παλινδρόμησης ανισότητας. Η πιθανότητα της κάθε ανισότητας *d*, υπολογίζεται από το εκτιμώμενο κόστος c(d) μέσω της συνάρτησης softmax. Η εκτιμώμενη ανισότητα με το N=Dmax, να υπολογίζεται ως:

$$\hat{d} = \sum_{d=0}^{N} d \times \sigma(-c_d) \quad (1.2)$$

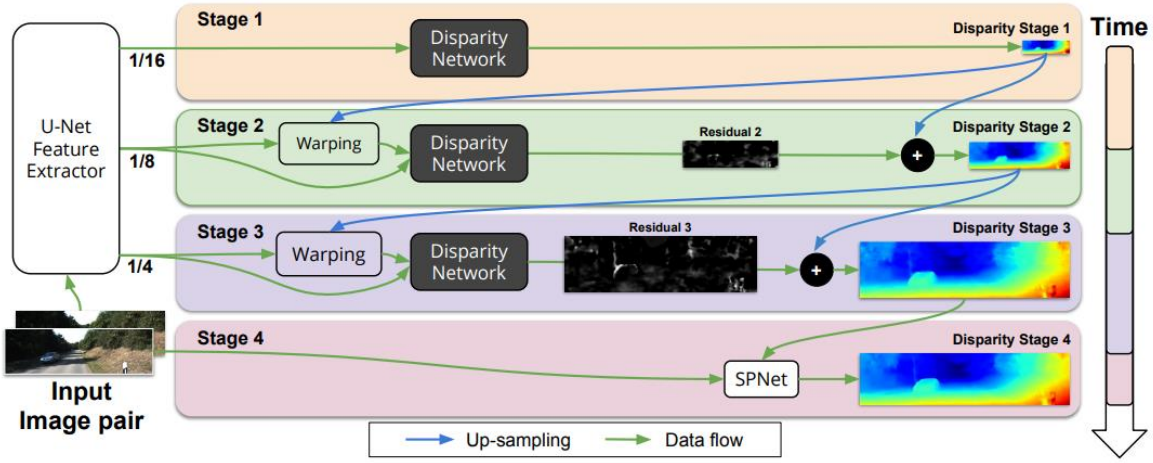
Αντίστοιχα με τα μοντέλα Monocular Depth Estimation που παρουσιάστηκαν, το PSMnet εισάγει μία νέα συνάρτηση κόστους, με το N να αντιπροσωπεύει τον αριθμό των pixels με ετικέτα, το d την «αληθινή» ανισότητα και το $\hat{d}$ την εκτιμώμενη ανισότητα.

$$L(d, \hat{d}) = \frac{1}{N} \sum_{i=1}^{N} Smooth_{L_1}(d_i - \hat{d}_i) \quad (1.3)$$

$$Smooth_{L_1}(x) = \begin{cases} 0.5x^2, |x| < 1 \\ |x| - 0.5, \dots \end{cases} \quad (1.4)$$

Με αυτές τις προδιαγραφές, το PSMNet κατάφερε να πετύχει SOTA αποτελέσματα στο σύνολο δεδομένων διατίμησης του KITTI.

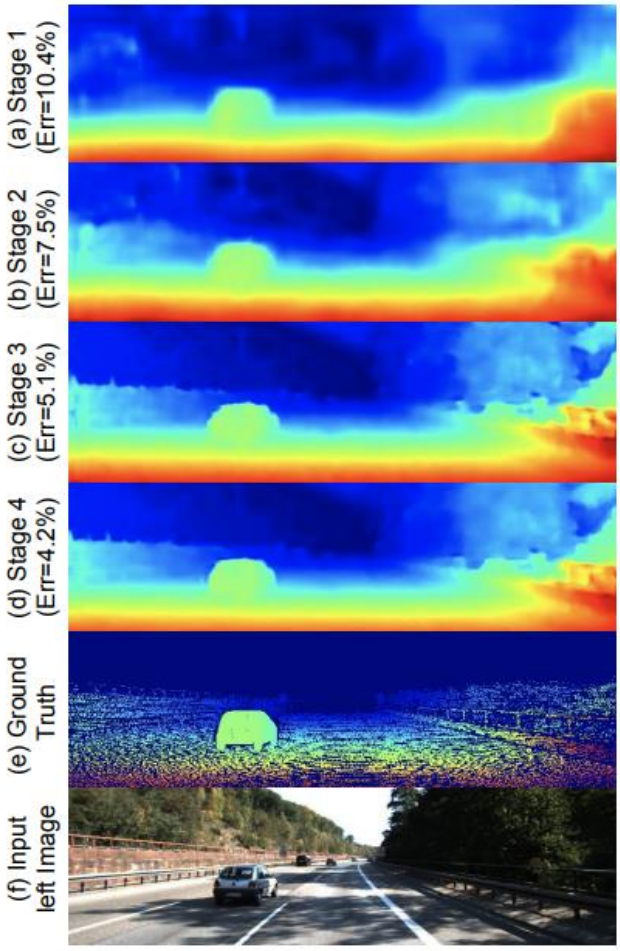
Σε σύγκριση με το PSMNet, το μοντέλο AnyNet σχεδιάστηκε για να αμφισβητήσει την αποτελεσματικότητα του PSMNet λόγω του μεγάλου χρόνου υπολογισμού βάθους που χρειάζεται σε κάθε σετ φωτογραφιών [19], καθώς υποστηρίζει ότι λύνει αυτό το πρόβλημα πετυχαίνοντας μέχρι και 100 φορές πιο γρήγορα αποτελέσματα με την χρήση της NVIDIA κάρτας γραφικών NVIDIA Jetson TX2, σε αναλύσεις φωτογραφιών 1242x375. Το πετυχαίνει αυτό δίνοντας μια μεταβλητή φύση στις αναλύσεις των ζευγών φωτογραφιών χωρίζοντάς το σε 4 στάδια. Το πρώτο στάδιο είναι και το πιο γρήγορο για τον υπολογισμό της κάθε εκτίμησης, με κόστος όμως την ανάλυση της φωτογραφίας, ενώ το τέταρτο στάδιο είναι το πιο αργό αλλά δημιουργεί εκτίμηση σε υψηλές αναλύσεις.



Σχήμα 1.5: Η αρχιτεκτονική του μοντέλου AnyNet [19]

Για να βγει μία εκτίμηση τα στάδια δουλεύουν συνδυαστικά. Ανάλογα το περιβάλλον στο οποίο βρίσκεται το τελικό σημείο που πρόκειται να καλέσει το μοντέλο για μία εκτίμηση, δίνεται ένας μέγιστος επιτρεπόμενος χρόνος για την παραγωγή μίας εκτίμησης. Για παράδειγμα αν δοθεί χρόνος εκτίμησης 50milliseconds (0.05 δευτερόλεπτα), τότε το μοντέλο θα μπορεί να δημιουργεί συνολικά 20

εκτιμήσεις το δευτερόλεπτο (1 / 0.05 = 20), το οποίο μπορεί να μεταφραστεί σε 20FPS [20]. Σε περίπτωση που δοθεί αρκετός χρόνος για να φτάσει σε μεγαλύτερο στάδιο, τότε κάθε επόμενο στάδιο εκμεταλλεύεται την πρόβλεψη του προηγούμενου σταδίου για να βγάλει πιο ακριβή αποτελέσματα. Το μοντέλο όμως είναι έτσι σχεδιασμένο έτσι ώστε ο μέγιστος χρόνος που χρειάζεται για να δημιουργηθεί μία εκτίμηση να είναι μεταβλητός ανάλογα την ανάγκη. Σε σημεία που απαιτείται μεγάλη λεπτομέρεια τότε πρόκειται το μοντέλο να εκμεταλλευτεί όλα τα στάδια, αντίθετα όταν η ανάγκη για την πληροφορία του βάθους χρειάζεται άμεσα τότε παρέχεται μικρότερος χρόνος και δεν περνάει από όλα τα στάδια.



Σχήμα 1.6: Εκτιμήσεις ανισότητας του AnyNet (a,b,c,d) σε κάθε στάδιο, οι «αληθινές» τιμές βάθους (e) και η φωτογραφία (f) που αντιπροσωπεύουν. Μέσα εμπεριέχεται και το σφάλμα (err) που έχει η κάθε εκτίμηση [19].

Κεφάλαιο 2ο: Τεχνητή Νοημοσύνη & Μηχανική Μάθηση

2.1 Τι είναι η Τεχνητή Νοημοσύνη

Στην ερώτηση τι είναι η Τεχνητή Νοημοσύνη δεν υπάρχει μία απάντηση, αλλά μια πληθώρα ορισμών που ορίζει τη σημασία της ΤΝ. Ένας από τους επικρατούντες ορισμούς ορίζει την Τεχνητή Νοημοσύνη ως την επιστήμη και τη μηχανική της δημιουργίας υπολογιστών με νοημοσύνη και εφαρμογών και υπόσταση ευφυίας. Ένας άλλος ορισμός ορίζει την Τεχνητή Νοημοσύνη ως ένα επιστημονικό πεδίο που συνδυάζει την επιστήμη των υπολογιστών σε συνδυασμό με μεγάλα σύνολα δεδομένων για την επίλυση περίπλοκων προβλημάτων. Αυτός ο ορισμός δίνει υπόσταση στα υπό-πεδία της Μηχανικής Μάθησης και της Βαθιάς Μάθησης που πλέον θεωρούνται ταυτόσημες έννοιες με την Τεχνητή Νοημοσύνη [21].

2.2 Μηχανική Μάθηση

Η μάθηση ως ορισμός ανάλογα το επιστημονικό πεδίο, πρόκειται να έχει διαφορετική έννοια αλλά και διαφορετική προσέγγιση. Αναφορικά με την οπτική γωνία της επιστήμης των υπολογιστών και συγκεκριμένα με το πεδίο της Μηχανικής Μάθησης, ως μάθηση ορίζεται η διαδικασία βελτίωσης της επίδοσης ενός συστήματος σε μία συγκεκριμένη εργασία μετά από την παρατήρηση πολλών παραδειγμάτων [22]. Σημαντικές εργασίες που βελτιώνονται μέσα από τη μάθηση περιλαμβάνουν την αναγνώριση αντικειμένων, την πρόβλεψη της τιμής και της αξίας μετρήσιμων ποσοτήτων, την συσταδοποίηση ομοειδών αντικειμένων, την ανάλυση και συμπίεση δεδομένων και την ανάπτυξη στρατηγικών. Ως άνθρωποι, μέσω της μάθησης μπορούμε να κατανοήσουμε καινούριες επαφές, πράξεις και γεγονότα, ταυτίζοντάς τα με προηγούμενες εμπειρίες.

Με τη μάθηση να θεωρείται ένα από τα σημαντικότερα σημεία της νοημοσύνης του ανθρώπου, η Μηχανική Μάθηση είναι ένας τομέας της Τεχνητής Νοημοσύνης που ασχολείται με την ανάπτυξη αλγορίθμων μάθησης [22]. Πρόκειται για αλγόριθμους που, προσομοιώνοντας τον τρόπο με τον οποίο μαθαίνει ο άνθρωπος, προσπαθούν να λύσουν προβλήματα σαν τα προαναφερθέντα προβλήματα που μπορούν να λυθούν με τη χρήση μάθησης. Για να επιτευχθεί αυτό, χρησιμοποιούμε μία πληθώρα δεδομένων που ονομάζεται dataset (σύνολο δεδομένων) με τη διαδικασία της βελτίωσης να γίνεται κατά κύριο λόγο σταδιακά λόγω της επαναληπτικής φύσης που έχει ο αλγόριθμος βελτίωσης. Αυτές οι επαναλήψεις ονομάζονται «εποχές μάθησης» ή «εποχές». Η διαδικασία που εμπεριέχει αυτά τα χαρακτηριστικά και εκτελεί τις συγκεκριμένες ενέργειες ονομάζεται Εκπαίδευση (Model Training) [23].

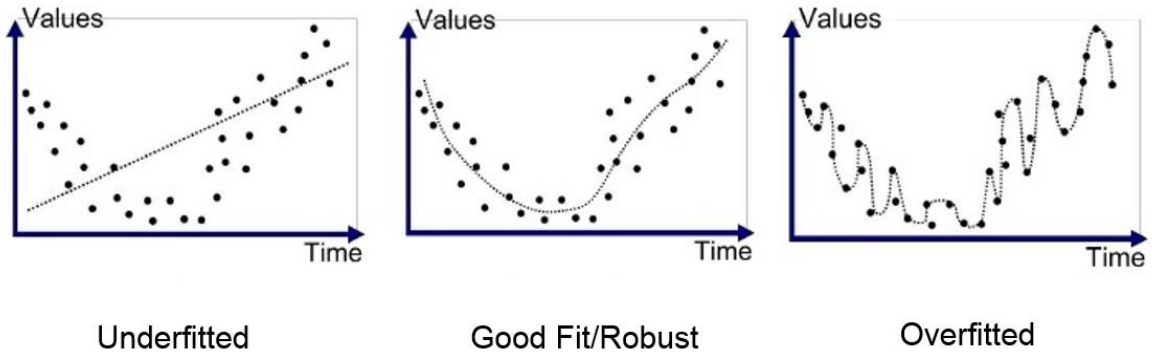
Τα δεδομένα που χρησιμοποιούμε σε ένα μοντέλο μηχανικής μάθησης χωρίζονται σε δύο κατηγορίες, τα training data (δεδομένα εκπαίδευσης) και τα testing data (δεδομένα δοκιμών). Τα δεδομένα εκπαίδευσης συνήθως περιέχουν το μεγαλύτερο κομμάτι δεδομένων που χρησιμοποιεί ένα μοντέλο μηχανικής μάθησης για να εκπαιδευτεί. Τη στιγμή που τελειώσει η διαδικασία της εκπαίδευσης ενός μοντέλου, ξεκινάει η διαδικασία της δοκιμής του (evaluation). Στη διαδικασία της δοκιμής του χρησιμοποιούμε τα δεδομένα δοκιμών. Τα δεδομένα δοκιμών, για να μπορούν να είναι χρήσιμα, θα πρέπει να έχουν την ίδια μορφή που έχουν και τα δεδομένα εκπαίδευσης και να είναι αρκετά, έτσι ώστε να μπορέσουν να βγουν σωστές εκτιμήσεις [24].

Σε διάφορα προβλήματα, η διαδικασία της μάθησης είναι αδύνατο να επιτευχθεί με την εκπαίδευση για κάθε πιθανή περίπτωση, καθώς είναι πρακτικά ανέφικτο να δημιουργηθούν δεδομένα που να καλύπτουν όλες τις πιθανότητες. Με βάση τα παραπάνω, καταλαβαίνουμε ότι η μάθηση δεν είναι μία απλή εργασία

αποθήκευσης και αναζήτησης δεδομένων. Αντιθέτως, σύμφωνα με το [25], στόχος της μάθησης είναι η δυνατότητα παραγωγής σωστών εκτιμήσεων σχετικά με δεδομένα τα οποία αντιμετωπίζονται για πρώτη φορά από το σύστημα.

Δύο πολύ σημαντικοί όροι στη Μηχανική Μάθηση είναι το overfitting (υπέρ-προσαρμογή) και το underfitting (υπό-προσαρμογή). Overfitting εννοούμε όταν ένα μοντέλο μηχανικής μάθησης δεν βγάζει σωστές εκτιμήσεις στα δεδομένα αποτίμησης. Αυτό συμβαίνει όταν ένα μοντέλο εκπαιδεύεται με μεγάλο αριθμό δεδομένων που καταλήγει να χρησιμοποιεί ανακριβή σημεία δεδομένων και δεδομένα με «θόρυβο» για την εκπαίδευσή του. Αυτό έχει ως αποτέλεσμα να μην μπορεί να εκτιμήσει ή να κατηγοριοποιήσει τα δεδομένα με σωστό τρόπο λόγω του θορύβου και του μεγάλου αριθμού λεπτομερειών. Το ακριβώς αντίθετο λέγεται underfitting. Όταν ένα μοντέλο δεν μπορεί να καταλάβει και να κατανοήσει την τάση που ακολουθούν τα δεδομένα, αυτό έχει σαν αποτέλεσμα να βγάζει ικανοποιητικά αποτελέσματα στα δεδομένα εκπαίδευσης, αλλά απογοητευτικά αποτελέσματα στα δεδομένα δοκιμής [26].

Η μηχανική μάθηση έχει εφαρμογές σε μια πλειάδα υπολογιστικών προβλημάτων. Παραδείγματα εξειδίκευσης περιέχουν την Ασφάλεια Υπολογιστών και Δικτύων με την αναγνώριση κακόβουλων λογισμικών, τα Χρηματοοικονομικά με την πρόβλεψη τιμών, την Εμπορία, τη διαφήμιση και την αυτόνομη οδήγηση. Αξιοσημείωτες εφαρμογές έχει επίσης και σε ενέργειες και πράξεις της ανθρώπινης καθημερινότητας, όπως είναι η επεξεργασία και η κατανόηση φυσικού λόγου, η αναγνώριση φυσικής γλώσσας, αλλά και η αναγνώριση εικόνας. Επίσης, έχουν υπάρξει εφαρμογές σε πολύ εξειδικευμένους τομείς, όπως είναι η υγεία και οι βιοεπιστήμες.



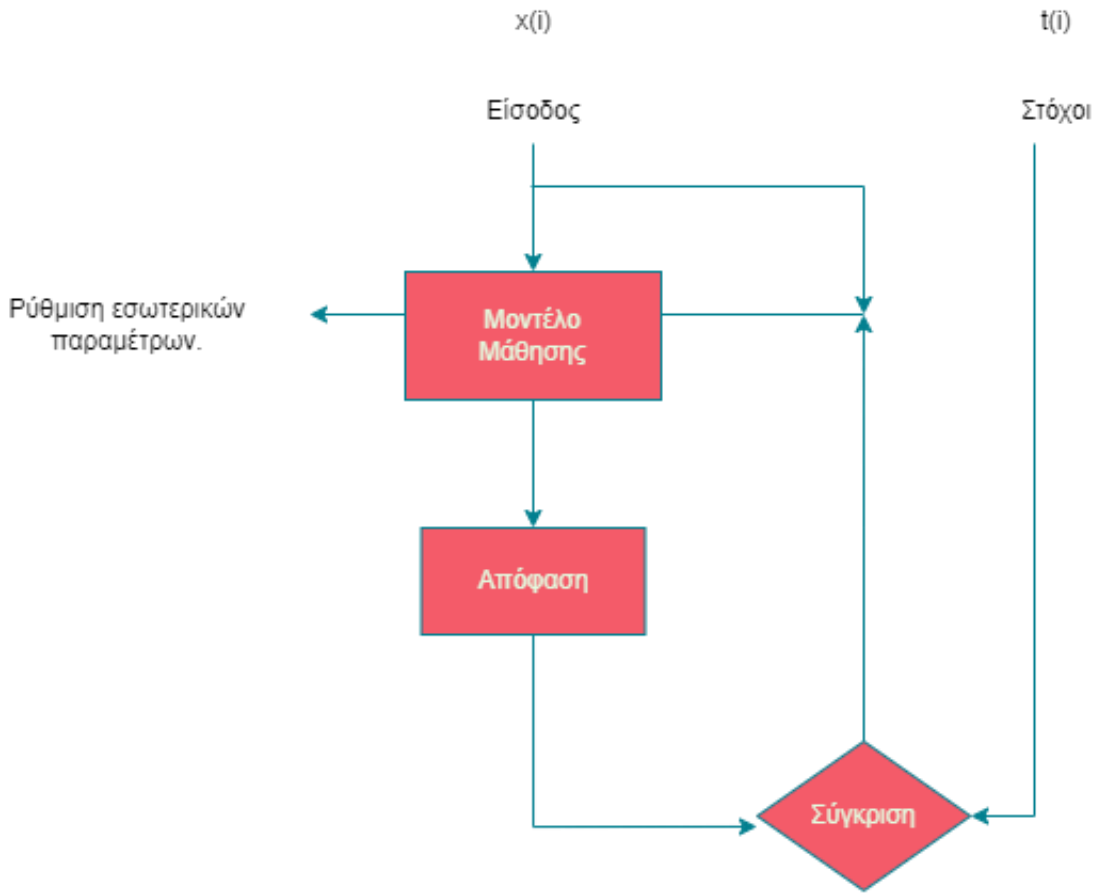
Σχήμα 2.1: Σύγκριση γραφήματος δεδομένων μεταξύ Overfit, Underfit και σωστής εκπαίδευσης [27].

2.2.1 Τύποι μάθησης

Η Μηχανική Μάθηση ορίζει τους τρεις κύριους τρόπους μάθησης, την μάθηση με επίβλεψη, την μάθηση χωρίς επίβλεψη και την μάθηση με ενίσχυση.

Μάθηση με επίβλεψη εννοούμε τον τύπο μάθησης που κατά την διάρκεια της εκπαίδευσης, αντιστοιχεί σε κάθε πρότυπο εισόδου, ένα πρότυπο εξόδου. Τα πρότυπα αναφέρονται συνήθως σε διανύσματα διαστάσεων απροσδιόριστου μεγέθους και στόχος τους είναι για κάθε πρότυπο εισόδου x, να προσπαθήσουμε να προβλέψουμε το πρότυπο εξόδου y, με την πραγματική έξοδο t να αντιπροσωπεύει τα αληθινά δεδομένα εξόδου. Τα προβλήματα που χρησιμοποιούν μάθηση με επίβλεψη χωρίζονται σε δύο κατηγορίες:

- Προβλήματα ταξινόμησης.
- Προβλήματα Παλινδρόμησης.



Σχήμα 2.2: Το βασικό μοντέλο της μάθησης με επίβλεψη

Προβλήματα ταξινόμησης (classification problems) είναι τα προβλήματα στα οποία το πλήθος των στόχων είναι ένας τετελεσμένος αριθμός C που όλοι οι στόχοι αντιπροσωπεύουν διακριτές κλάσεις ταξινόμησης, π.χ. στην ταξινόμηση φωτογραφιών οι φωτογραφίες αντιπροσωπεύονται με αριθμητικά ψηφία. Αν υπάρχουν δέκα διακριτές κλάσεις, αυτές αντιστοιχίζονται από τον αριθμό μηδέν (0) μέχρι τον αριθμό εννιά (9).

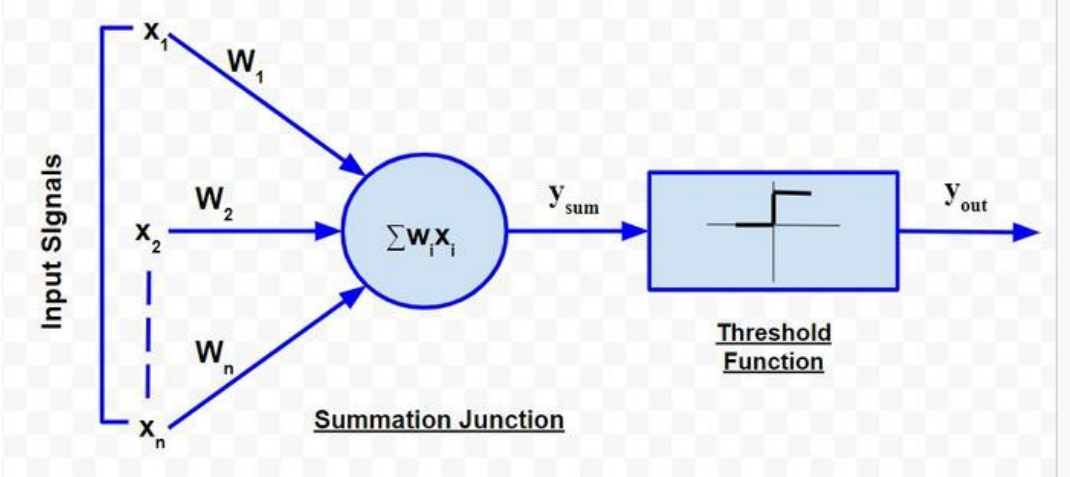
Στα προβλήματα παλινδρόμησης (regression problems) οι στόχοι των εκτιμήσεων αντιπροσωπεύουν είτε συνεχής τιμές, είτε το πλήθος των κλάσεων που ταξινομείται είναι απεριόριστο. Παράδειγμα προβλήματος παλινδρόμησης είναι ο εκτιμητής τιμών ακίνητης περιουσίας, ο οποίος δεν μπορεί να κατηγοριοποιηθεί σε έναν συγκεκριμένο αριθμό διακριτών κλάσεων, αλλά απαιτεί έναν πιο ευέλικτο και μεταβλητό τρόπο εκτίμησης.

Όταν αναφερόμαστε στην μάθηση χωρίς επίβλεψη (unsupervised learning), αναφερόμαστε στην διαδικασία της μάθησης και σε μοντέλα που ενώ έχουν πρότυπα εισόδου x, δεν υπάρχουν αντιστοιχήσεις σχετικά με τους στόχους.

Αντιθέτως, όταν μιλάμε για Μάθηση με Ενίσχυση (reinforcement learning) αναφερόμαστε σε μοντέλα που χρησιμοποιούν ακολουθίες εισόδων x και οι στόχοι κατατάσσονται σε ανταμοιβές και τιμωρίες. Αυτά τα συστήματα συνήθως επιβραβεύονται ή τιμωρούνται στο τέλος κάθε ακολουθίας βάσει της επίδοσής τους.

2.2.2 Μοντέλο McCulloch-Pitts

Τα πρώτα παραδείγματα Τεχνιτών Νευρωνικών Δικτύων έχουν τις ρίζες τους στο 1943, ενώ το πρώτο υπολογιστικό μοντέλο που προσμοίαζε στη λογική νευρώνα είναι ο νευρώνας McCulloch-Pitts [28, 29]. Το μοντέλο McCulloch-Pitts μπορεί να δεχτεί δύο είδη εισόδων x, που είτε δίνουν θετικές τιμές πάνω στα βάρη του νευρώνα, είτε δίνουν αρνητικές. Οι είσοδοι του νευρώνα μπορούν να είναι είτε 0 είτε 1, με την έξοδο y να είναι μία βηματική συνάρτηση που παίρνει είτε τις τιμές {-1,1}, είτε τις τιμές {0,1}. Η τιμή αυτή καθορίζεται από το άθροισμα Y_{sum}των εισόδων.



Σχήμα 2.3:Τεχνητό Νευρωνικό Μοντέλο McCulloch-Pitts [29]

2.3 Βαθιά μάθηση

Η βαθιά μάθηση είναι ένα υπό-πεδίο της Μηχανικής Μάθησης και συνήθως αναφέρεται σε νευρωνικά δίκτυα που έχουν τρία ή και παραπάνω στρώματα. Αυτά τα νευρωνικά δίκτυα εξάγουν πληροφορία μέσα από μεγάλα σύνολα δεδομένων. Ενώ τα απλά νευρωνικά δίκτυα μπορούν να κάνουν κάποιες σχετικές εκτιμήσεις, τα επιπλέον στρώματα στα οποία αναφέρεται η βαθιά μάθηση μπορούν να βελτιστοποιήσουν την αποδοτικότητα ενός νευρωνικού δικτύου.

Αν και η Βαθιά Μάθηση θεωρείται κομμάτι της Μηχανικής Μάθησης, υπάρχουν διάφορα σημεία στα οποία διαφέρουν αυτά τα δύο πεδία. Το πιο σημαντικό σημείο διαφοράς αυτών των δύο πεδίων είναι η φύση των δεδομένων. Οι αλγόριθμοι Μηχανικής Μάθησης απαιτούν μία συγκεκριμένη δομή και πολλές φορές μία προ-επεξεργασία από τα δεδομένα τα οποία πρόκειται να χρησιμοποιήσουν για να κάνουν τις εκτιμήσεις. Αντιθέτως, οι αλγόριθμοι βαθιάς μάθησης δεν απαιτούν προ-επεξεργασία και μπορούν να διαχειριστούν δεδομένα που δεν υπάγονται σε συγκεκριμένες δομές (π.χ. εικόνες, κείμενο). Επιπλέον, οι αλγόριθμοι μπορούν να λύσουν προβλήματα δυσκολότερης φύσεως, όπως είναι η αναγνώριση και εξόρυξη χαρακτηριστικών σε φωτογραφίες. Τέλος, αντίθετα από τους απλούς αλγόριθμους Μηχανικής Μάθησης, οι αλγόριθμοι βαθιάς μάθησης απαιτούν μεγάλη υπολογιστική ισχύ, την οποία πετυχαίνουν με τη χρήση καρτών γραφικών, αυξάνοντας έτσι και το κόστος των αλγορίθμων [30].

2.4 Μάθηση με επίβλεψη

Σε ένα μοντέλο μηχανικής μάθησης, οι είσοδοι x παίρνουν την μορφή διανύσματος και ονομάζονται πρότυπο. Με την χρήση των δεδομένων που περιέχει αυτό το διάνυσμα x, γίνεται προσπάθεια εκτίμησης

της ζητούμενης ποσότητας t. Για παράδειγμα, στο πρόβλημα αναγνώρισης αντικειμένων επίπλων με τη χρήση φωτογραφιών, οι είσοδοι x είναι παίρνουν τιμές που αναπαριστούν δεδομένα pixel. Η ζητούμενη τιμή t αντιπροσωπεύει κλάσεις αντικειμένων επίπλων (καναπές, πολυθρόνα, τραπέζι κτλ.). Η κωδικοποίηση των κλάσεων σε τιμές t γίνονται με διάφορους τρόπους. Σε περίπτωση που θέλουμε να χωρίσουμε την αναγνώριση αντικειμένων σε δύο κλάσεις (π.χ. Καναπές, Πολυθρόνα), τότε οι τιμές που αναπαριστούν τις κλάσεις παίρνουν είτε τις τιμές T{0,1}, είτε τις τιμές T{-1,1}. Σε περίπτωση που ο αριθμός κλάσεων είναι μεγαλύτερος του 2, τότε η κωδικοποίηση των τάξεων μπορεί να γίνει με διάφορους τρόπους.

Πίνακας 2.1: Συνήθεις τρόποι κωδικοποίησης στόχων σε προβλήματα ταξινόμηση με αριθμό κλάσεων C>2 [22]

T = {0,1,2…,C-1}	Με το T να αναπαριστά τους στόχους, και το C να αναπαριστά τον αριθμό κλάσεων, η συγκεκριμένη κωδικοποίηση χωρίζει τις κλάσεις σε αριθμούς από το 0 μέχρι το C-1. Αν έχουμε 9 κλάσεις και μια φωτογραφία ταξινομείται στην κλάση 5, τότε ο στόχος T θα είναι 4, αν είναι κλάση 1 τότε ο στόχος T θα είναι 0, αν είναι κλάση 9 τότε ο στόχος T θα είναι 9.
T = {1,2,3,…,C}	Με το T να αναπαριστά τους στόχους, και το C να αναπαριστά τον αριθμό κλάσεων, η συγκεκριμένη κωδικοποίηση χωρίζει τις κλάσεις σε αριθμούς από το 1 μέχρι το C. Κάθε αριθμός κλάσης αντιστοιχεί στον αντίστοιχο αριθμό T. Η κλάση 9 στον αριθμό 9, η κλάση 1 στον αριθμό 1 κτλ.
T = [0 …. 0 1 0 …. 0]	Στην περίπτωση αυτή με το T να αναπαριστά τους στόχους, και το k την θέση στο διάνυσμα μεγέθους C, το διάνυσμα παίρνει τις τιμές 0 σε όλο το διάνυσμα με εξαίρεση την τιμή στην θέση k που παίρνει την τιμή 1. Για παράδειγμα αν έχουμε 4 κλάσεις τότε ο στόχος T για την κλάση 2 θα είναι της μορφής [0 1 0 0] με το k = 2, το C=4. Σε περίπτωση που είναι κλάση 4 με C=4, τότε το k=4 και ο στόχος T είναι το διάνυσμα T = [0 0 0 1]

Υπάρχουν πολλές περιπτώσεις στις οποίες οι εκτιμήσεις των μοντέλων δεν πρόκειται να αναπαριστούν τα πραγματικά δεδομένα. Για να κρίνουμε την απόδοση ενός μοντέλου ταξινόμησης σε δύο κλάσεις, χρησιμοποιούμε χαρακτηριστικά του λεγόμενου πίνακα σύγχυσης (confusion matrix). Σαν πρώτο μέτρο σύγκρισης χρησιμοποιούμε την ακρίβεια (accuracy) που υπολογίζεται από το άθροισμα των πετυχημένων εκτιμήσεων διαιρεμένο από το σύνολο του πλήθους των εισόδων. Έτσι ισχύει ότι:

$$accuracy = \frac{TN+TP}{TN+TP+FN+FP},\ 0 \le accuracy \le 1 \qquad (2.1)$$

Με το TN και το TP να αντιπροσωπεύει όλες τις σωστές εκτιμήσεις για τις κλάσεις 0 και 1 αντίστοιχα, και το FN, FP να είναι το πλήθος των λανθασμένων εκτιμήσεων για τις κλάσεις 0 και 1 αντίστοιχα.

Σε περίπτωση που ο αριθμός κλάσεων C είναι μεγαλύτερος του 2, τότε η ερμηνεία των παραπάνω τιμών παίρνει διαφορετική αλλά παρόμοια έννοια, θέτοντας τις μεταβλητές True Negative (TN), True Positive (TP), False Negative (FN) και False Positive (FP).

- **TN**: Όταν η εκτίμηση βγήκε αρνητική για την κλάση i, και η πραγματική τιμή δεν ανήκει στην συγκεκριμένη κλάση.
- **TP:** Όταν η εκτίμηση βγήκε θετική για την κλάση i και η πραγματική τιμή ανήκει στην συγκεκριμένη κλάση.
- **FN:** Όταν η εκτίμηση βγήκε αρνητική για την κλάση i και η πραγματική τιμή ανήκει στην συγκεκριμένη κλάση.
- **FP**: Όταν η εκτίμηση βγήκε θετική για την κλάση i και η πραγματική τιμή δεν ανήκει στην συγκεκριμένη κλάση.

Έτσι, σε περιπτώσεις στις οποίες ο αριθμός κλάσεων C > 2, ορίζουμε τη μέση ακρίβεια (average accuracy) ως:

$$Acc_m = \frac{1}{C}\sum_{i=1}^{C} \frac{tp_i+tn_i}{tp_i+tn_i+fp_i+fn_i} \quad (2.2)$$

Στα προβλήματα παλινδρόμησης χρησιμοποιούμε διαφορετικό τρόπο για την αξιολόγηση της επίδοσης ενός μοντέλου. Διότι σε αυτά τα προβλήματα οι στόχοι αναφέρονται συνήθως σε πραγματικές τιμές, ενώ η χρήση της συνάρτησης της ακρίβειας πρακτικά δεν είναι χρήσιμη. Τα πιο δημοφιλή παραδείγματα για την εκτίμηση ακρίβειας παλινδρόμησης είναι το μέσο τετραγωνικό σφάλμα, το μέσο απόλυτο σφάλμα, η ομοιότητα συνημίτονου και η ομοιότητα Pearson.

Θέτοντας το N ως πλήθος προτύπων εισόδου X(i), με τον επιθυμητό στόχο T(i) για κάθε πρότυπο εισόδου i παρουσιάζονται τα εξής [22]:

- Το μέσο τετραγωνικό σφάλμα – Mean Square Error (MSE): Μετράει την μέση ευκλείδεια απόσταση μεταξύ διανυσμάτων των εκτιμήσεων Y, και των διανυσμάτων των στόχων T για κάθε πρότυπο i με το σφάλμα να παίρνει πάντα τιμές ίσες ή μεγαλύτερες του μηδενός με την βέλτιστη τιμή να είναι το μηδέν.

$$J_{MSE} = \sum_{p=1}^{N} \|t_p - y_p\|^2 \quad (2.3)$$

- Η ομοιότητα συνημίτονου – Cosine Similarity: Αντίθετα με τα προηγούμενα κριτήρια επίδοσης παλινδρόμησης, η ομοιότητα συνημίτονου δεν βασίζεται στη στατιστική επεξεργασία των δεδομένων, αλλά εφαρμόζεται σε ζευγάρια διανυσμάτων των πραγματικών τιμών t και των εκτιμήσεων y, μετρώντας τη μεταξύ τους ομοιότητα με βάση το κανονικοποιημένο εσωτερικό τους γινόμενο.

$$J_{MAE} = \sum_{p=1}^{N} \sum_{i=1}^{m} |t_{p,i} - y_{p,i}| \quad (2.4)$$

- Η ομοιότητα συνημίτονου – Cosine Similarity: Αντίθετα με τα προηγούμενα κριτήρια επίδοσης παλινδρόμησης, η ομοιότητα συνημίτονου δεν βασίζεται στην στατιστική επεξεργασία των δεδομένων αλλά εφαρμόζεται σε ζευγάρια διανυσμάτων των πραγματικών τιμών t και των εκτιμήσεων y, μετρώντας την μεταξύ τους ομοιότητα βάση το κανονικοποιημένο εσωτερικό τους γινόμενο.

$$J_{cos} = \frac{t^T y}{\|t\|\|y\|} = \frac{\sum_{i=1}^{m} t_i y_i}{\sqrt{\sum_{i=1}^{m} t_i^2}\sqrt{\sum_{i=1}^{m} y_i^2}} \quad (2.5)$$

- Η ομοιότητα Pearson – Pearson Similarity: Αντίστοιχα με την ομοιότητα συνημίτονου, η ομοιότητα Pearson χρησιμοποιείται σε ζευγάρια εκτιμήσεων και πραγματικών δεδομένων συσχετίζοντάς τα με τον εξής τύπο:

$$J_P = \frac{(t-\bar{t})^T(y-\bar{y})^T}{\|t-\bar{t}\|\|y-\bar{y}\|} = \frac{\sum_{i=1}^{m}(t_i-\bar{t}_i)(y_i-\bar{y}_j)}{\sqrt{\sum_{i=1}^{m}(t_i-\bar{t}_i)^2}\sqrt{\sum_{i=1}^{m}(y_i-\bar{y}_i)^2}} \quad (2.6)$$

με το $\bar{t}$ και το $\bar{y}$ να είναι οι μέσες τιμές των t και y αντίστοιχα.

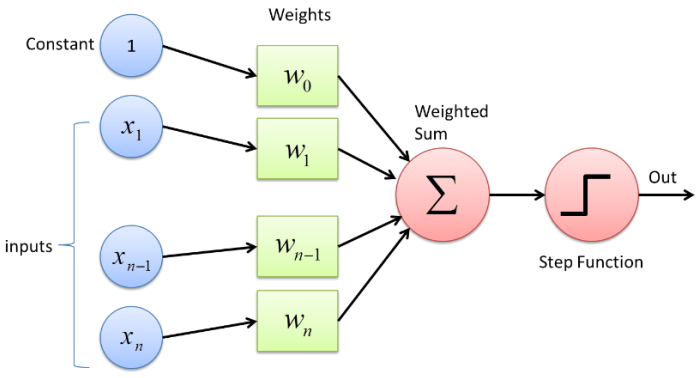
2.4.1 Νευρωνικό δίκτυο Perceptron

Η σχεδίαση του πιο απλού νευρωνικού δικτύου προϋποθέτει την ύπαρξη ενός και μόνο τεχνητού νευρώνα. Αυτήν τη σχεδίαση την υλοποιεί το νευρωνικό δίκτυο Perceptron. Σχεδιασμένο για την επίλυση προβλημάτων γραμμικής ταξινόμησης, το Perceptron θεωρείται ένα από τα απλούστερα είδη των τεχνητών νευρωνικών δικτύων [31].

Το Perceptron έχει τρεις κύριες παραμέτρους, τις τιμές εισόδου (inputs), βάρη (weights) και πόλωση (bias), τις συναρτήσεις μεταφοράς και ενεργοποίησης (activation function). Ως τιμές εισόδου x αναφέρονται τα αρχικοποιημένα δεδομένα που εισέρχονται στο σύστημα για την περεταίρω ανάλυση και επεξεργασία τους. Τα βάρη (w) αντιπροσωπεύουν την ισχύ που έχει η κάθε μονάδα του πρότυπου της εισόδου και οι τιμές τους είναι άμεσα συνδεδεμένες με τον καθορισμό του αποτελέσματος της εξόδου, ενώ η πόλωση (b) μπορεί να θεωρηθεί ως ο ρυθμός μετατόπισης της ευθείας μιας γραμμικής εξίσωσης. Η συνάρτηση μεταφοράς υλοποιείται από τη μονό-νευρωνική φύση του Perceptron ως:

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \quad (2.7)$$

Το y είναι η έξοδος, x_i είναι οι είσοδοι, w_i είναι τα βάρη και b είναι η πόλωση. Η τελευταία παράμετρος ενός δικτύου Perceptron είναι η συνάρτηση ενεργοποίησης [Σχήμα 2.3]. Υπάρχουν διάφοροι τύποι συνάρτησης ενεργοποίησης, από τους οποίους οι πιο γνωστοί είναι η βηματική συνάρτηση (step function), η Σιγμοειδής (sigmoid), η Υπερβολική Εφαπτομένη (hyperbolic tangent), η συνάρτηση κατωφλίου (threshold function) και η συνάρτηση ράμπας γνωστή και ως ανορθωμένη γραμμική μονάδα (Rectified Linear Unit – ReLU).



Σχήμα 2.4: Παράδειγμα αρχιτεκτονικής δικτύου Perceptron [32]

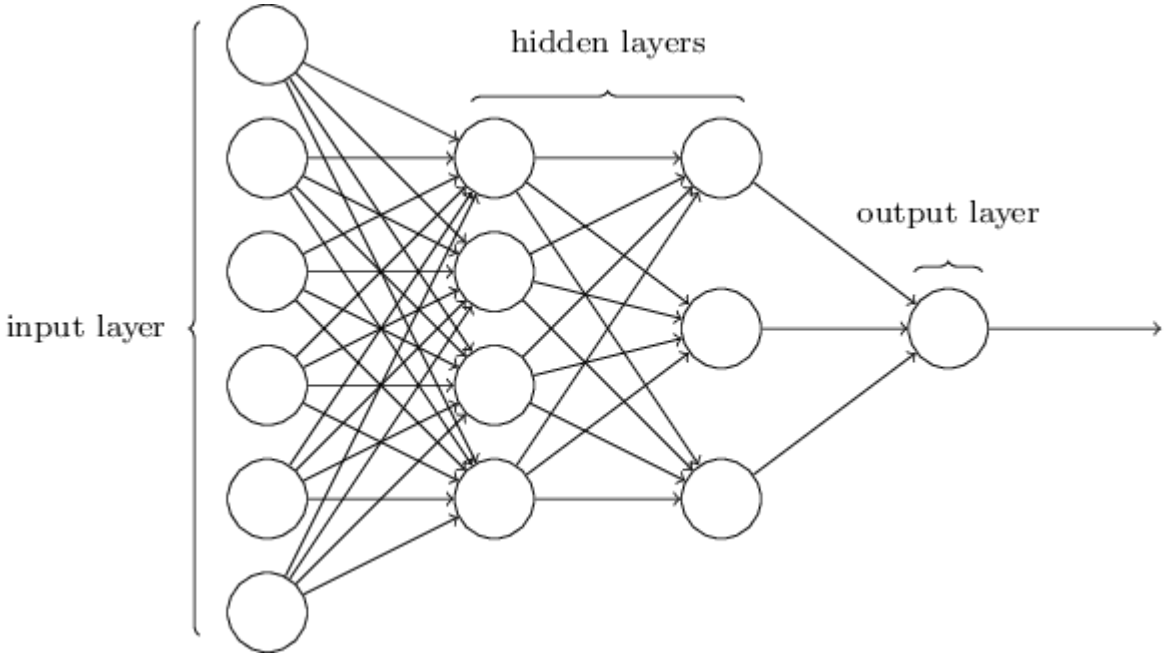
Το νευρωνικό δίκτυο Perceptron λειτουργεί με δύο βασικά βήματα. Στο πρώτο βήμα πολλαπλασιάζει όλες τις τιμές εισόδου με τις αντίστοιχες τιμές βάρους. Έπειτα εφόσον τις έχει πολλαπλασιάσει, τις αθροίζει για να υπολογίσει το γενικό σύνολο (συνάρτηση μεταφοράς). Το σύνολο συνήθως υπολογίζεται ως $\Sigma_{x_i\omega_i} = x_1w_1 + x_2w_2 + \cdots + x_nw_n$ και στη συνέχεια προσθέτουμε την τιμή της πόλωσης (b) και το συνολικό αποτέλεσμα υπολογίζεται ως: $\Sigma_{x_i\omega_i} + b$. Στο δεύτερο βήμα το μοντέλο

χρησιμοποιεί τη συνάρτηση ενεργοποίησης για να κρίνει το αθροισμένο αποτέλεσμα σε μορφή εξόδου Y, είτε σε δυαδική μορφή, είτε σε συνεχείς τιμές.

2.4.2 Multi-layer Perceptron

Το Multi-layer Perceptron (MLP) μοντέλο, όπως λέει και το όνομά του, είναι ένα νευρωνικό μοντέλο βασισμένο στο δίκτυο Perceptron. Διαφέρει όμως σε σχέση με το Perceptron διότι περιέχει πολλαπλά στρώματα. Η επεξεργαστική ισχύς του MLP είναι πολύ μεγαλύτερη σε σχέση με το απλό δίκτυο Perceptron και για αυτό μπορεί να λύσει μη γραμμικά προβλήματα ταξινόμησης.

Η πιο απλή μορφή ενός MLP είναι η αύξηση των νευρώνων που χρησιμοποιούνται σε ένα δίκτυο Perceptron. Αυτή η αύξηση επιτυγχάνεται με τη χρήση κρυφών στρωμάτων (hidden layers) που ακολουθούν παρόμοια λογική με αυτή που ακολουθεί ένα νευρωνικό δίκτυο Perceptron. Κάθε MLP μοντέλο πρέπει να έχει τουλάχιστον ένα κρυφό στρώμα. Μέσα σε αυτό στο στρώμα εμπεριέχονται οι τεχνητοί νευρώνες που προσομοιάζουν με την ίδια λογική που προσομοίαζαν στο νευρωνικό δίκτυο Perceptron.



Σχήμα 2.5: Παράδειγμα αρχιτεκτονικής μοντέλου MLP [33].

Βάσει του παραδείγματος στο [Σχήμα 2.5] παρατηρούμε ότι το νευρωνικό δίκτυο περιέχει τέσσερα στρώματα: ένα στρώμα εισόδου, ένα στρώμα εξόδου και δύο κρυφά στρώματα. Τα στρώματα εισόδου τροφοδοτούν κάθε ένα από τους νευρώνες στο πρώτο κρυφό στρώμα όπως τροφοδοτούνταν ο νευρώνας στο δίκτυο Perceptron. Στην συνέχεια, το αποτέλεσμα μετά από την επεξεργασία και ανάλυσή τους στο πρώτο στρώμα τροφοδοτεί το δεύτερο κρυφό στρώμα και αυτά με τη σειρά τους επεξεργάζονται τα δεδομένα και τροφοδοτούν το στρώμα εξόδου.

Ενώ με μία πρώτη εκτίμηση φαίνεται μία ανωτερότητα του MLP σε σχέση με τον Perceptron, τα συμπεράσματα τείνουν να ποικίλουν. Τα MLP μπορούν να λύσουν περίπλοκα μη γραμμικά προβλήματα, βρίσκουν εκτιμήσεις αμέσως μετά την εκπαίδευση και είναι διαχειριστικά σε σύνολα δεδομένων διαφόρων μεγεθών. Αντίθετα, τα MLP απαιτούν μεγάλη επεξεργαστική ισχύ και οι υπολογισμοί τους είναι χρονοβόροι, ενώ η ποιότητα του κάθε μοντέλου βασίζεται στην ποιότητα της

διαδικασίας της εκπαίδευσης, καθώς υπάρχει και δυσκολία στη διαχείριση των εξαρτημένων μεταβλητών και στο πόσο επηρεάζουν κάθε ανεξάρτητη μεταβλητή [31].

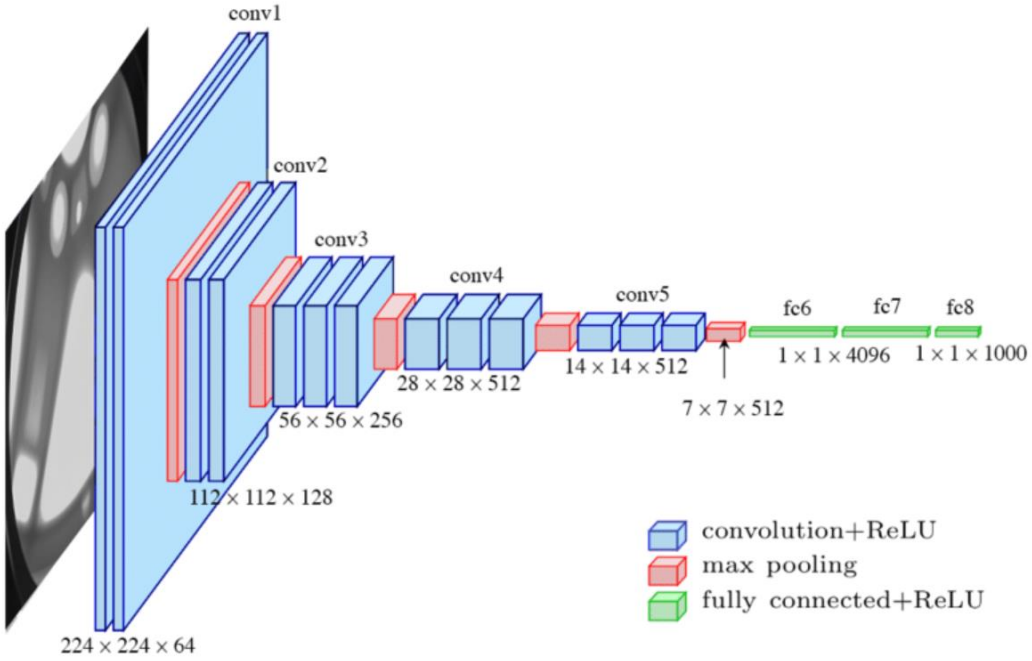
2.4.3 Convolutional Neural Networks

Εξίσου σημαντική κατηγορία νευρωνικών δικτύων πολλών στρωμάτων είναι και τα συνελικτικά νευρωνικά δίκτυα (convolutional neural networks – CNN). Τα CNN είναι νευρωνικά δίκτυα που εξειδικεύονται σε εργασίες αναγνώρισης εικόνας, προτάθηκαν από τον Yann LeCun το 1980 και θεωρούνται μοντέλα βαθιάς μάθησης [34]. Το μεγαλύτερο πρόβλημα που υπάρχει στη χρήση απλών νευρωνικών δικτύων για την επίλυση προβλημάτων εικόνας είναι η έλλειψη επεκτασιμότητας που έχουν. Ενώ μπορεί ένα δίκτυο MLP να βγάλει ικανοποιητικά αποτελέσματα σε μία ασπρόμαυρη εικόνα χαμηλής ανάλυσης, όταν αυξηθεί η πολυπλοκότητα της φωτογραφίας είτε σε ανάλυση είτε σε στρώματα χρωμάτων (από ασπρόμαυρη σε έγχρωμη), τότε αυξάνεται η υπολογιστική ισχύς που απαιτείται για να μπορέσει να ανταπεξέλθει το νευρωνικό δίκτυο. Εκτός από την υπολογιστική ισχύ, με την πάροδο του χρόνου και των εποχών, δημιουργείται και πρόβλημα υπέρ-προσαρμογής του μοντέλου. Αντιθέτως, ένα CNN χρησιμοποιεί διάφορες τεχνικές που το καθιστούν τη βέλτιστη επιλογή για επίλυση προβλημάτων εικόνας.

Τα εσωτερικά στρώματα που αποτελούν ένα CNN διακρίνονται στο στρώμα συνέλιξης και στο στρώμα υποδειγματοληψίας. Τα στρώματα συνέλιξης περιέχουν έναν αριθμό χαρτών χαρακτηριστικών, ενώ κάθε χάρτης χαρακτηριστικών περιέχει ένα δισδιάστατο πλέγματος μεγέθους N * N με το N να είναι φυσικός αριθμός. Αυτό το πλέγμα ονομάζεται αλλιώς και μάσκα. Έτσι μπορεί να βγει το συμπέρασμα ότι κάθε συνελικτικό στρώμα περιέχει τόσους νευρώνες όσο το γινόμενο του δισδιάστατου πλέγματος και του πλήθους των χαρτών χαρακτηριστικών. Ο μαθηματικός τύπος για τον νευρώνα στη θέση (i,j) στον χάρτη χαρακτηριστικών νούμερο k ορίζεται από τον παρακάτω τύπο:

$$y_{ij}^{k} = f\left(\sum_{l=1}^{c'}\sum_{a=1}^{m}\sum_{\beta=1}^{m} w_{a,\beta,l}^{k} x_{i-a,j-\beta}^{l} + b^{k}\right) \quad (2.8)$$

Η πράξη εντός της παρένθεσης της συνάρτησης f ονομάζεται συνέλιξη και το αποτέλεσμά της όπως και στα προαναφερθέντα νευρωνικά δίκτυα περνάει μέσα από μία συνάρτηση ενεργοποίησης μη γραμμικής φύσης όπως η ReLU και η συνάρτηση υπερβολικής εφαπτομένης. Αυτό που κάνει ουσιαστικά ένα συνελικτικό στρώμα είναι να λειτουργεί σαν φίλτρο και να εξάγει χαρακτηριστικά από μια φωτογραφία εισόδου προσπαθώντας να προσομοιώσει τις λειτουργίες του οπτικού φλοιού του ματιού [22].



Σχήμα 2.6: Παράδειγμα αρχιτεκτονικής συνελικτικού νευρωνικού δικτύου [35].

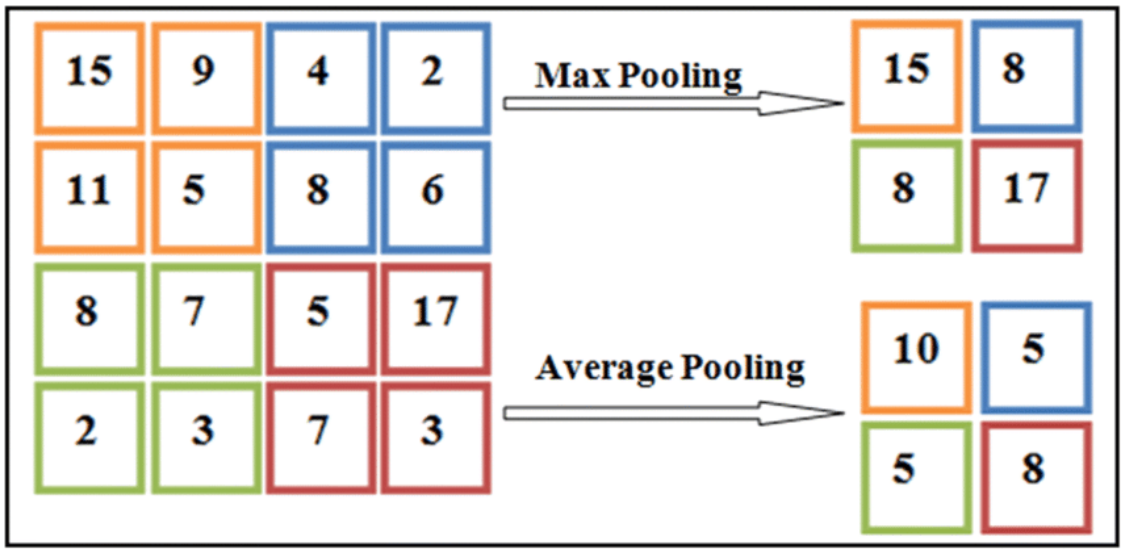
Το στρώμα υποδειγματοληψίας (sub-sampling) ακολουθεί τα στρώματα συνέλιξης και σαν κύριο στόχο έχει να μειώσει τα σφάλματα που μπορούν να προκληθούν από την ευαίσθητη φύση που έχουν τα δεδομένα εικόνων, αφαιρώντας πλεονάζουσες λεπτομέρειες χωρίς να προκαλεί απώλεια δεδομένων εικόνας. Στη συνέχεια, συμπιέζει τα δεδομένα και μειώνει τον συνολικό αριθμό των πράξεων που χρειάζονται να γίνουν. Ουσιαστικά, στο στρώμα υποδειγματοληψίας, η έξοδος y κάθε νευρώνα θεωρείται η σύνοψη των εξόδων των νευρώνων, δηλαδή παίρνει τις τιμές από το προηγούμενο στρώμα και επιστρέφει μία συνοπτική τιμή. Ένα απλό παράδειγμα συνάρτησης που αντικατοπτρίζει συνοπτική τιμή είναι η υποδειγματοληψία της μέσης τιμής (average-pooling) και ορίζεται ως:

$$y_{ij} = \frac{1}{m^2} \sum_{\alpha=0}^{m-1} \sum_{b=0}^{m-1} x_{im+\alpha,jm+b} \quad (2.9)$$

Ένα άλλο εξίσου σημαντικό και δημοφιλές παράδειγμα είναι η υποδειγματοληψία της μέγιστης τιμής (max-pooling) που ορίζεται ως εξής:

$$y_{ij} = max\left(x_{im+\alpha,jm+b}\right), a \geq 0, b \leq m-1 \quad (2.10)$$

Η υποδειγματοληψία της μέγιστης τιμής έχει καλύτερα αποτελέσματα σε χαρακτηριστικά μικρότερης διάστασης και για αυτό προτείνεται περισσότερο σε σχέση με την υποδειγματοληψία της μέσης τιμής [22].

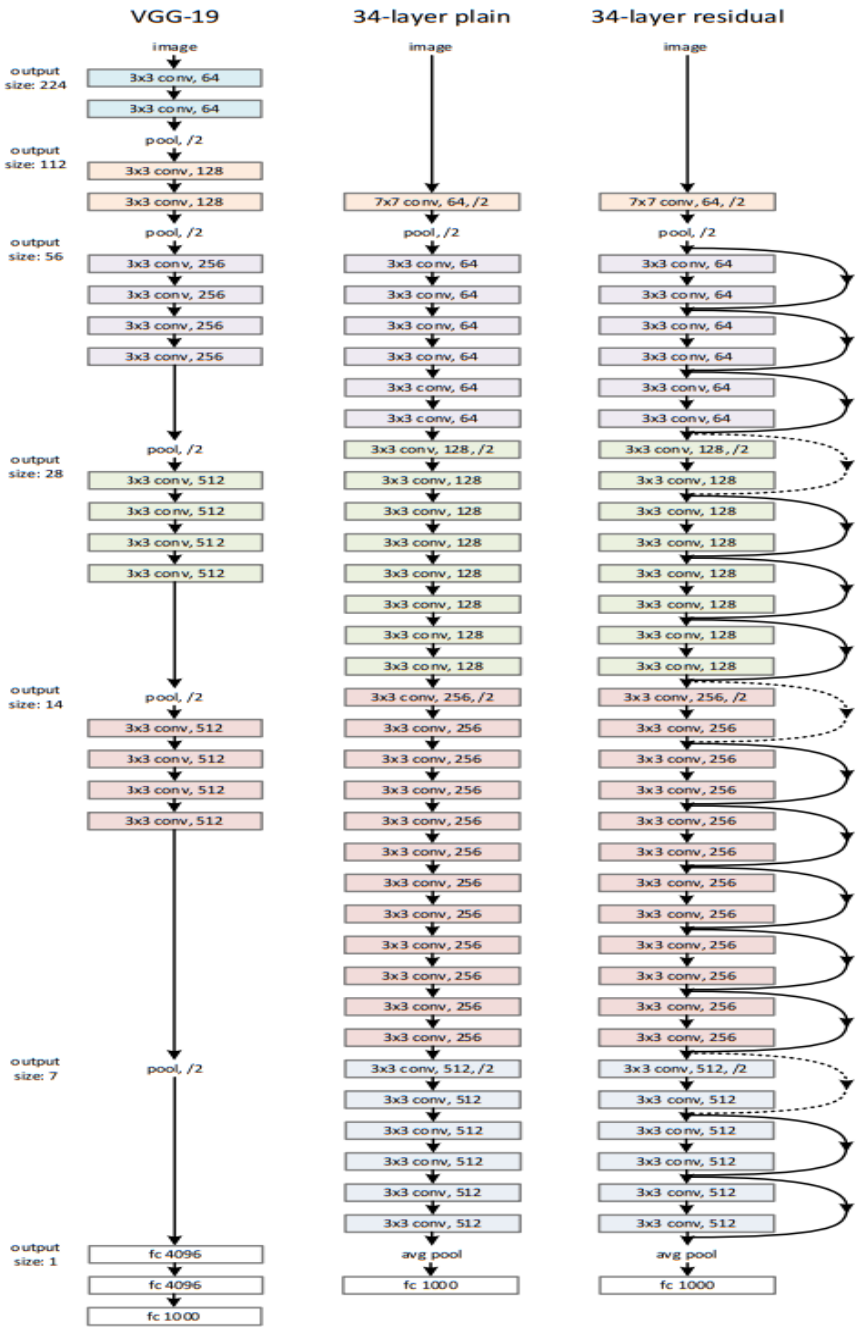


Σχήμα 2.7 Σύγκριση υποδειγματοληψίας μέσης και μέγιστης τιμής [36].

2.4.4 Παραδείγματα Convolutional Neural Network

Ανά τα χρόνια έχουν υπάρξει διάφορες υλοποιήσεις και μοντέλα τύπου CNN, όπως είναι τα δίκτυα σαν το LeNet-5 που έχουν τις ρίζες τους το 1998 [37] και το AlexNet του 2012 που έστρωσε τον δρόμο για τα μοντέρνα συνελικτικά νευρωνικά δίκτυα, δημιουργώντας μεγάλα χάσματα μεταξύ των προηγούμενων SOTA συνελικτικών δικτύων. Άλλα παραδείγματα εμπεριέχουν τα VGG-16 και VGG-19 CNN μοντέλα, τα οποία έχουν 16 και 19 στρώματα αντίστοιχα, ενώ σημαντικά παραδείγματα είναι και το Google Net (ή Inception v1) μοντέλο με 22 συνολικά στρώματα. Το συγκεκριμένο μοντέλο κατάφερε να πετύχει συνολική ακρίβεια ποσοστού 93.3% στον διαγωνισμό ImageNet του 2014 και σε προβλήματα ταξινόμησης αλλά και ανίχνευσης αντικειμένων, σε φωτογραφίες [38].

Άλλα σημαντικά παραδείγματα νευρωνικών δικτύων είναι οι υλοποιήσεις ResNet και το DenseNet. Το ResNet δημιουργήθηκε το 2015 από τη Microsoft με σκοπό να λύσει προβλήματα που είχαν τα προηγούμενα συνελικτικά δίκτυα μεγάλου βάθους παρουσιάζοντας μία νέα τεχνική που ονομάζεται skip connections. Αυτό που πετυχαίνει η συγκεκριμένη τεχνική είναι να συνδέει τα εσωτερικά στρώματα του νευρωνικού δικτύου με άλλα στρώματα εκτός του αμέσως επόμενου. Για το ResNet έχουν προταθεί διάφορες υλοποιήσεις, οι οποίες περιέχουν μέχρι και 152 στρώματα [39]. Με αντίστοιχο τρόπο δουλεύει και το μοντέλο DenseNet, με τη διαφορά ότι ενώ το μοντέλο ResNet δέχεται σαν είσοδο την έξοδο ενός εκ των προηγούμενων τριών στρωμάτων, το DenseNet συγκεντρώνει όλες τις εξόδους για τα στρώματα πριν από το κάθε στρώμα [40].



Σχήμα 2.8: Σύγκριση αρχιτεκτονικών συνελικτικών δικτύων. Στην πρώτη στήλη απεικονίζεται η αρχιτεκτονική του μοντέλου VGG-19, στην δεύτερη στήλη απεικονίζεται ένα απλό συνελικτικό δίκτυο 34 στρωμάτων, και στην τρίτη στήλη απεικονίζεται ένα ResNet τύπου μοντέλο 34 στρωμάτων [39].

2.5 Μάθηση χωρίς επίβλεψη

Η μάθηση χωρίς επίβλεψη (Unsupervised Learning) είναι ο τρόπος μάθησης για τη Μηχανική Μάθηση που χρησιμοποιεί δεδομένα χωρίς ετικέτα. Στη μάθηση με επίβλεψη υπάρχουν τα δεδομένα και οι αντίστοιχοι στόχοι. Στη μάθηση χωρίς επίβλεψη, μέσα στα δεδομένα δεν εμπεριέχονται οι συγκεκριμένοι στόχοι. Συνεπώς, αντί να χρησιμοποιεί συνδυασμό προτύπων δεδομένων με συγκεκριμένο στόχο, χρησιμοποιεί μόνο τα δεδομένα. Η συγκεκριμένη μεθοδολογία μπορεί και βρίσκει μοτίβα πάνω στα δεδομένα χωρίς τη χρήση ανθρώπινης παρέμβασης. Οι δυνατότητες που έχουν αυτοί

οι αλγόριθμοι να βρίσκουν ομοιότητες και διαφορές στα δεδομένα τους καθιστούν ιδανικούς για προβλήματα αναγνώρισης εικόνας.

2.5.1 Χρήσεις Μαθήσεως χωρίς επίβλεψη

Οι εργασίες στις οποίες χρησιμοποιούνται οι αλγόριθμοι μάθησης χωρίς επίβλεψη είναι τρεις: η ομαδοποίηση (clustering), η συσχέτιση (association) και η μείωση διαστάσεων (dimensionality reduction) των δεδομένων. Ως μείωση διαστάσεων των δεδομένων ορίζεται η εργασία που έχει ως σκοπό να μειώσει το πλήθος των δεδομένων με κύριο στόχο να διατηρηθεί η αρχική πληροφορία [41]. Η ομαδοποίηση δεδομένων είναι μία τεχνική που χρησιμοποιείται στην εξόρυξη πληροφορίας που ως κύριο στόχο έχει να ομαδοποιήσει μεγάλο αριθμό δεδομένων χωρίς ετικέτα (unlabeled data), βάσει των ομοιοτήτων και των διαφορών τους. Υπάρχουν διαφορετικών ειδών αλγόριθμοι ομαδοποίησης.

Ένα είδος αλγορίθμων ομαδοποίησης είναι η αποκλειστική ομαδοποίηση (Exclusive Clustering). Στην αποκλειστική ομαδοποίηση, συσχετίζουμε τα δεδομένα με κεντρικό σημείο που υπάρχει μόνο σε ένα σύμπλεγμα. Παράδειγμα αυτής της μεθοδολογίας είναι ο αλγόριθμος K-Means, όπου το K αναφέρεται στα κεντρικά σημεία των συστάδων. Τα δεδομένα χωρίζονται σε K πλήθος συστάδων, όπου το K αναπαριστά τον αριθμό των συστάδων βάσει της απόστασης από το κέντρο του κάθε συμπλέγματος. Τα δεδομένα κατατάσσονται βάσει της απόστασής τους από τα κεντρικά σημεία των συστάδων και τοποθετούνται σε αυτό με τη μικρότερη απόσταση από το κέντρο του. Αντίστοιχα, η αποκλειστική ομαδοποίηση δουλεύει με παρόμοιο τρόπο με τη διαφορά ότι κεντρικά σημεία των συστάδων ανήκουν σε πολλαπλά συμπλέγματα.

Εξίσου σημαντικό παράδειγμα αλγορίθμου ομαδοποίησης είναι η Ιεραρχική Ομαδοποίηση (Hierarchical Clustering). Στην Ιεραρχική Ομαδοποίηση τα κεντρικά σημεία τους σε πρώτο στάδιο απομονώνονται σε διαφορετικές ομάδες και, στη συνέχεια, σε κάθε στάδιο ομαδοποιούνται βάσει των ομοιοτήτων τους μέχρι να μείνει ένα ακριβώς σύμπλεγμα. Χρησιμοποιούνται τέσσερις κύριες μέθοδοι για να υπολογιστεί η ομοιότητά τους:

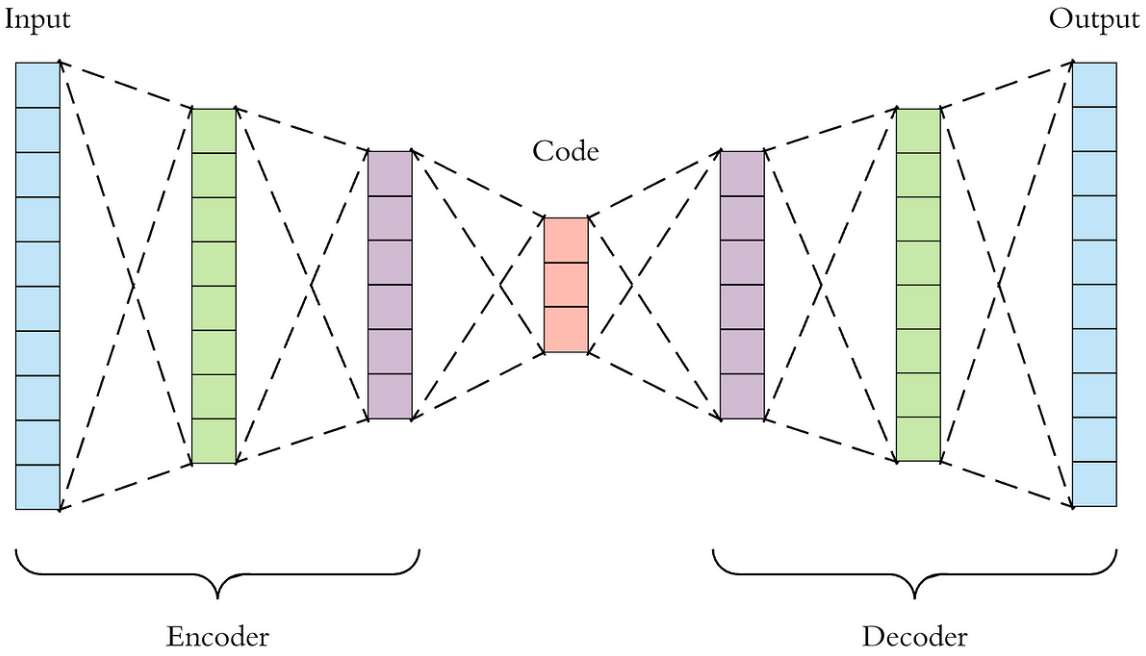
- Σύνδεση Ward (Ward's Linkage): Αυτή η μέθοδος ορίζει ότι η απόσταση μεταξύ δύο συστάδων είναι ορισμένη από την αύξηση του αθροίσματος του τετραγώνου μετά την ένωση των δύο συστάδων.
- Μέση Σύνδεση (Average Linkage): Η συγκεκριμένη μέθοδος ορίζεται από τη μέση απόσταση μεταξύ σημείων δύο συστάδων.
- Μέγιστη Σύνδεση (Maximum Linkage): Η μέθοδος ορίζεται από τη μέγιστη απόσταση μεταξύ σημείων σε δύο συστάδες.
- Ελάχιστη Σύνδεση (Minimum Linkage): Η μέθοδος αυτή ορίζεται από την ελάχιστη απόσταση μεταξύ σημείων για δύο συστάδες.

Η τελευταία κατηγορία αλγορίθμων ομαδοποίησης είναι η Πιθανολογική Ομαδοποίηση (Probabilistic Clustering). Πιθανολογικά μοντέλα χρησιμοποιούνται για να λύσουν απλά (soft) προβλήματα ομαδοποίησης. Στην πιθανολογική ομαδοποίηση, τα κεντρικά σημεία των συστάδων κατηγοριοποιούνται βάσει της πιθανότητάς τους να ανήκουν στη συγκεκριμένη συστάδα. Η πιο συνήθης ομάδα μοντέλων που χρησιμοποιείται στην πιθανολογική ομαδοποίηση είναι τα GMM (Gaussian Mixture Models).

Εξίσου σημαντική εργασία με την ομαδοποίηση είναι και η συσχέτιση (κανόνες συσχέτισης). Οι κανόνες συσχέτισης είναι μέθοδοι βασισμένες σε κανόνες που ως στόχο έχουν να βρίσκουν συνδέσεις μεταξύ μεταβλητών στα εκάστοτε σύνολα δεδομένων. Αυτές οι μέθοδοι χρησιμοποιούνται συνήθως σε προβλήματα τύπου market basket analysis [42] και αλγορίθμων Απριόρι (Apriori algorithms) [43]

δίνοντας τη δυνατότητα σε επιχειρήσεις να βρουν συσχετίσεις μεταξύ διαφορετικών προϊόντων, αλλά και να δημιουργούν προηγμένα συστήματα συστάσεων.

Η μείωση διαστάσεων είναι μία εργασία που, ενώ φαινομενικά φαίνεται αχρείαστη ή και επιβλαβής για τα δεδομένα και τη φύση τους, είναι πολύ σημαντική για την επίλυση προβλημάτων, των οποίων η φύση ξεκινά από την πλεονάζουσα πληροφορία των δεδομένων. Παραδείγματα αλγορίθμων που συμπεριλαμβάνονται σε αυτήν την κατηγορία είναι το Principal Component Analysis (PCA) που συμπιέζει τα δεδομένα εξάγοντας τα χαρακτηριστικά τους και το Singular Value Decomposition (SVD) που παραγοντοποιεί έναν πίνακα A σε τρεις μικρότερους. Επιπροσθέτως, ένα εξίσου σημαντικό παράδειγμα αλγορίθμων μείωσης διαστάσεων είναι οι αυτό-κωδικοποιητές (Autoencoders), οι οποίοι συμπιέζουν και αναδημιουργούν τα δεδομένα σε μιμήσεις των πηγαίων δεδομένων.



Σχήμα 2.9: Παράδειγμα αυτό-κωδικοποιητή με διαδικασία κωδικοποίησης – συμπίεσης και αποκωδικοποίησης – αναδημιουργίας [44].

Η Μάθηση χωρίς επίβλεψη έχει προσφέρει τη δυνατότητα να αναλύονται τα δεδομένα με διαφορετικό τρόπο και να παρατηρούνται μοτίβα σε μεγάλους όγκους δεδομένων. Έτσι, έχει δημιουργηθεί μια πληθώρα μοντέλων και αλγορίθμων που βρίσκουν εφαρμογές σε διάφορα προβλήματα πραγματικού κόσμου, όπως είναι οι ειδησεογραφικές υπηρεσίες, η όραση υπολογιστή, η Ιατρική, τα συστήματα συστάσεων, η αναγνώριση ανωμαλιών, αλλά και το «χτίσιμο» πελατειακών προφίλ [45].

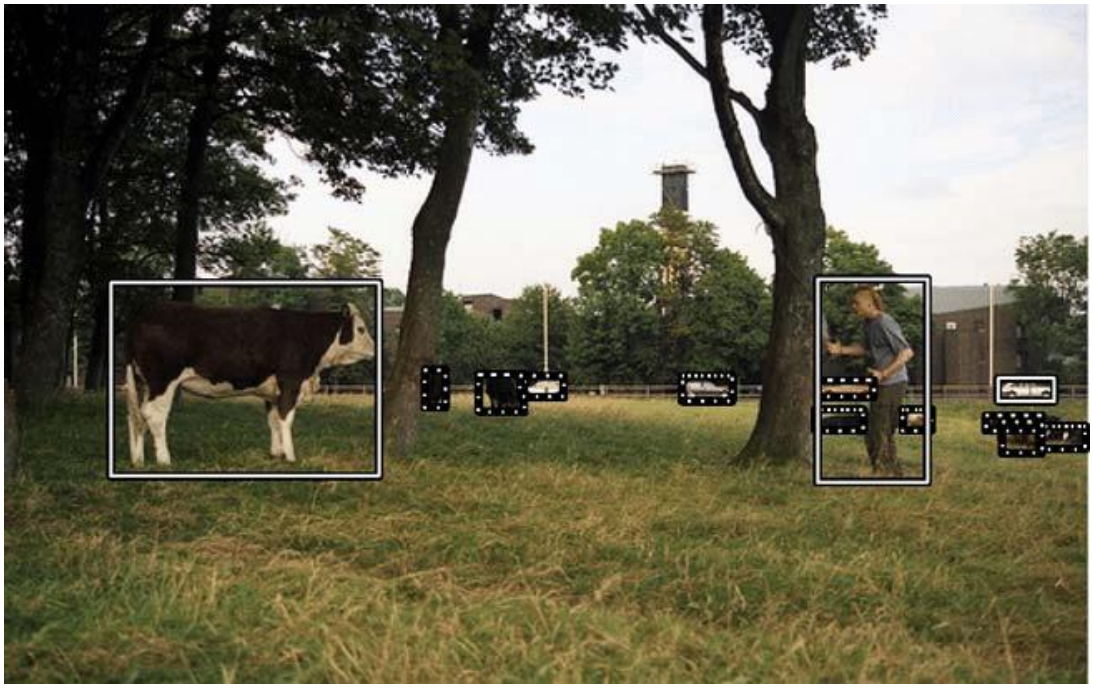
Κεφάλαιο 3ο: Όραση υπολογιστή & εκτίμηση βάθους

3.1 Προβλήματα όρασης υπολογιστή

Τα προβλήματα όρασης με υπολογιστή (Computer Vision) είναι μία κατηγορία προβλημάτων που αφορά διάφορες εργασίες της καθημερινότητας που έχουν να κάνουν με την όραση. Ο άνθρωπος χρησιμοποιεί την όραση για να αναγνωρίσει πρόσωπα και αντικείμενα, να εκτιμήσει το βάθος και αναγνωρίσει το περιβάλλον του κ.ά. Αντίστοιχα, προβλήματα Όρασης με υπολογιστή εμφάνιζαν λύσεις και προκλήσεις από μοντέλα Μηχανικής Μάθησης από το 1950.

Ένα από τα σημαντικότερα προβλήματα της όρασης υπολογιστή είναι η ταξινόμηση εικόνας (Image classification). Η πρώτη διεργασία όρασης υπολογιστή που χρησιμοποιήθηκε από αλγόριθμους μηχανικής μάθησης ήταν για την ταξινόμηση εικόνας και συγκεκριμένα χρησιμοποιήθηκε το μοντέλο Perceptron για την υλοποίησή του. Τα μεγαλύτερα βήματα στην ταξινόμηση εικόνας έγιναν με τη χρήση των Συνελικτικών νευρωνικών δικτύων όπως το LeNet [37], και στη συνέχεια με την εισαγωγή του AlexNet που ήταν το πρώτο CNN μοντέλο που νίκησε τον διαγωνισμό ImageNet.

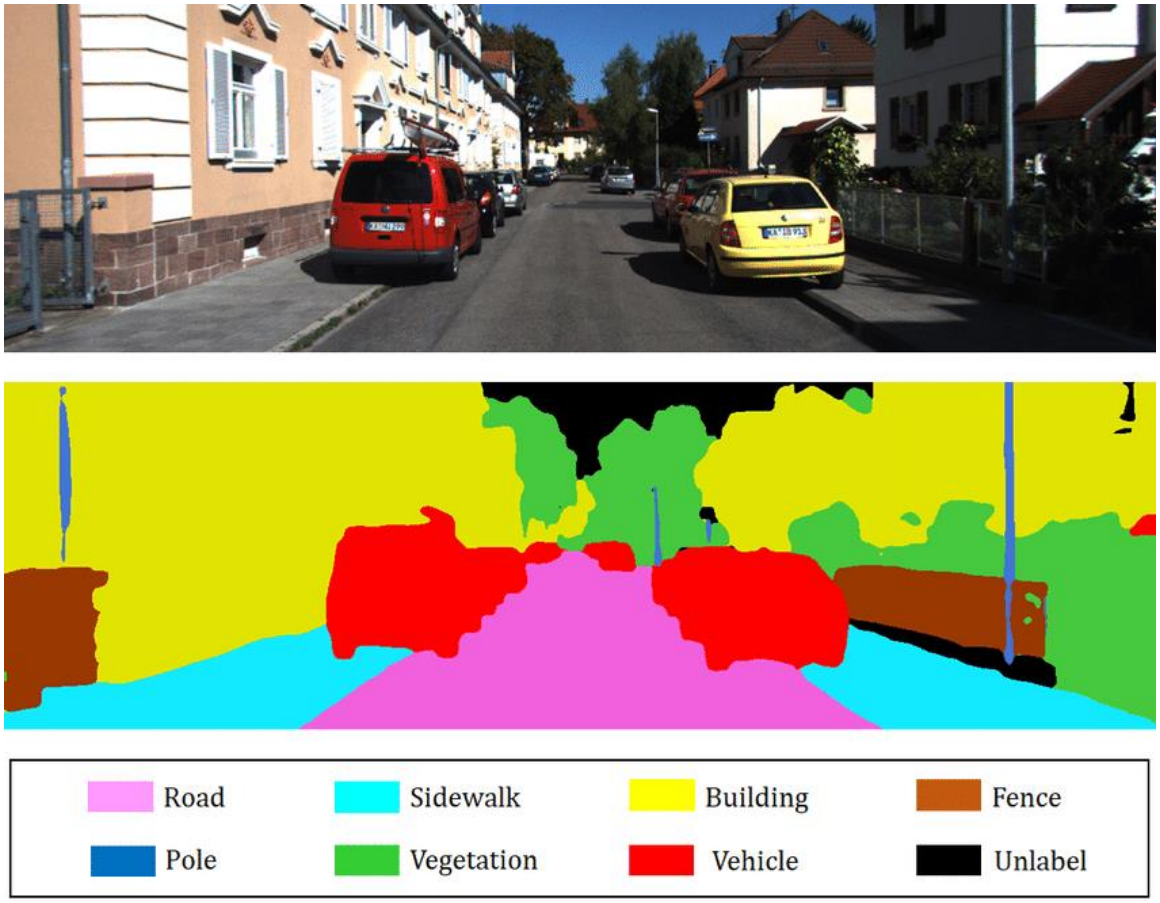
Ένα ακόμη σημαντικό πρόβλημα της όρασης υπολογιστή είναι η τοποθέτηση και αναγνώριση αντικειμένων σε φωτογραφία (Object Localization and Detection). Σε αυτό το πρόβλημα ο αλγόριθμος έχει ως σκοπό δεδομένου κάποια αντικείμενα – στόχος να τα εντοπίσει στο πλαίσιο μίας φωτογραφίας. Το συγκεκριμένο πρόβλημα έχει πολλές εφαρμογές σε Ρομποτική, αυτόνομη οδήγηση, επαυξημένη πραγματικότητα, αλλά βρίσκει εφαρμογή και σε προβλήματα ιατρικής φύσεως [46].



Σχήμα 3.1: Παράδειγμα Object Localization and Detection από το dataset PASCAL VOC [46]

Εξίσου σημαντικό πρόβλημα είναι αυτό της ταυτοποίησης εικόνας (Image segmentation). Η διεργασία της ταυτοποίησης εικόνας χωρίζει μία εικόνα σε διάφορα σημασιολογικά κομμάτια. Ο στόχος αυτής της διεργασίας είναι να συνδέσει κάθε pixel μίας εικόνας σε ένα σημασιολογικό κομμάτι. Διάφοροι αλγόριθμοι έχουν χρησιμοποιηθεί για την επίλυση της συγκεκριμένης διεργασίας. Έχουν υπάρξει

λύσεις που χρησιμοποιούν παραδοσιακούς διαδικαστικούς αλγορίθμους, αλλά με την άνοδο της βαθιάς μάθησης σαν κλάδο, τα αποτελέσματα με τη χρήση νευρωνικών δικτύων έχουν βελτιωθεί ραγδαία [47].



Σχήμα 3.2: Παράδειγμα Image segmentation με την εικόνα να χωρίζεται σε 8 σημασιολογικά κομμάτια [48].

Το πιο σημαντικό πρόβλημα της όρασης υπολογιστή είναι το «ταίριασμα» στοιχείων και χαρακτηριστικών, μεταξύ δύο ζευγών φωτογραφιών. Ουσιαστικά, είναι η αντιστοίχιση των Pixels ενός αντικειμένου που παρουσιάζεται σε δύο διαδοχικές εικόνες. Από αυτό το πρόβλημα δημιουργούνται διάφορα νέα προβλήματα, συμπεριλαμβανομένου και του προβλήματος της οπτικής κίνησης (optical flow). Η οπτική κίνηση είναι ένα διάνυσμα που δημιουργείται μεταξύ της μετακίνησης δύο διαδοχικών φωτογραφιών και αυτό το διάνυσμα παρουσιάζει τη μετακίνηση που είχαν τα pixel επάνω στις εικόνες [49].

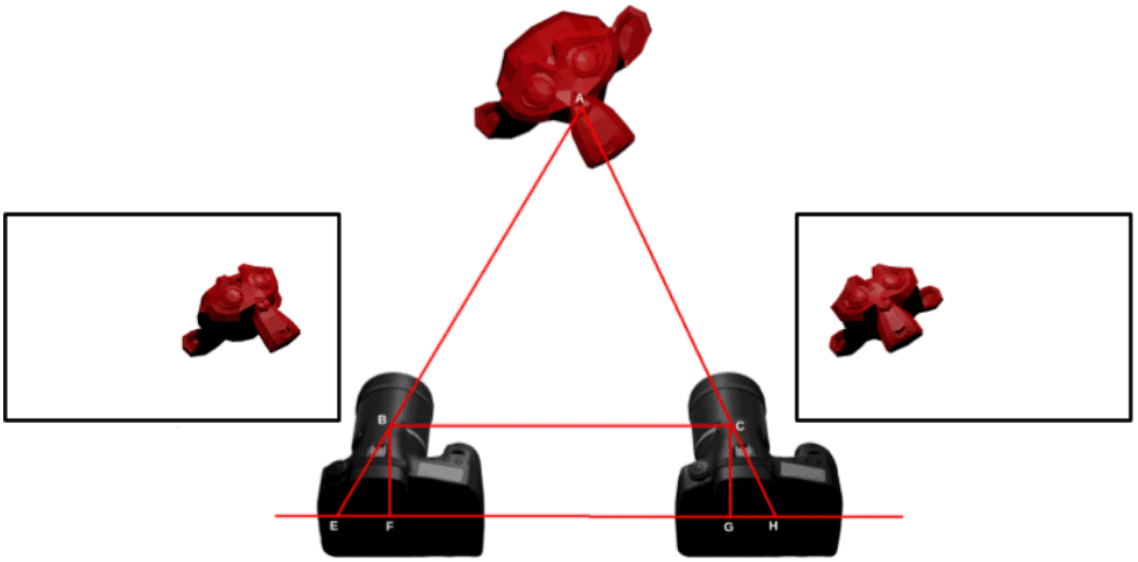
3.2 Εκτίμηση Βάθους

Όπως έχει προαναφερθεί στο πρώτο κεφάλαιο, υπάρχουν δύο τεχνικές για να εκτιμήσουμε το βάθος με τη χρήση απλών καμερών. Αυτές οι τεχνικές ονομάζονται Monocular Depth Estimation και Stereo Depth Estimation. Όταν αναφερόμαστε στην τεχνική του Monocular Depth Estimation, αναφερόμαστε στη χρήση δεδομένων εικόνας από μία κάμερα για την εκτίμηση του βάθους. Αυτό επιτυγχάνεται με τη χρήση νευρωνικών δικτύων και μοντέλων βαθιάς μάθησης, τα οποία έχουν δει μία σημαντική άνοδο τα τελευταία χρόνια.

Στο Stereo Depth Estimation, αντίθετα από το Monocular Depth Estimation, δεν έχουμε απλά να λύσουμε την εργασία της εκτίμησης βάθους με ωμά δεδομένα από δύο κάμερες. Όπως αναφέρθηκε και στην παράγραφο της οπτικής κίνησης, ένα κύριο κομμάτι του Stereo Depth Estimation περιλαμβάνει το

ταίριασμα pixel σε δύο διαδοχικές εικόνες. Σε αυτό το κομμάτι θέτουμε ένα ζεύγος φωτογραφιών αριστερής και δεξιάς φωτογραφίας και υπάρχει σαν κύριος στόχος το ταίριασμα των pixel των αντικειμένων μεταξύ αυτών των δύο φωτογραφιών.

Η εργασία της εκτίμησης ανισότητας (Disparity Estimation) είναι αυτή που υπολογίζει την ανισότητα μεταξύ δύο φωτογραφιών. Οι κλασικές μέθοδοι που χρησιμοποιούνταν για την εκτίμηση της ανισότητας χρησιμοποιούν ζεύγη φωτογραφιών από αριστερό και δεξί πεδίο θέασης και υπολογίζουν την απόσταση σε pixels που έχουν οι δύο φωτογραφίες όσον αφορά συγκεκριμένα αντικείμενα.



Σχήμα 3.3: Ζεύγος φωτογραφιών από αριστερό και δεξί πεδίο θέασης [50]

Άλλες τεχνικές εκτίμησης ανισότητας περιλαμβάνουν τη χρήση νευρωνικών δικτύων και μοντέλων βαθιάς μάθησης. Η εκπαίδευση των μοντέλων γίνεται σε προϋπάρχοντα σύνολα δεδομένων (dataset) όπως το KITTI, το SceneFlow, το Middlebury κ.ά. [50].

3.2.1 Βαθμονόμηση δύο καμερών (Stereo Camera Calibration).

Το βάθος είναι μία πληροφορία που προέρχεται από την ύπαρξη της τρίτης διάστασης του χώρου, όταν καταγράφουμε έναν χώρο με τη χρήση καμερών σε μορφή δισδιάστατης φωτογραφίας χάνουμε την πληροφορία του βάθους. Οι φωτογραφίες παρουσιάζουν σε δισδιάστατη μορφή, ενώ τα pixels της φωτογραφίας περιέχουν μηδαμινή πληροφορία σχετικά με το βάθος. Η αξιοποίηση δύο διαδοχικών φωτογραφιών που αντιπροσωπεύουν την αριστερή και δεξιά οπτική γωνία ως προς έναν χώρο βοηθά στη δημιουργία ενός ψηφιακού τρισδιάστατου χώρου μετά από σύγκριση των φωτογραφιών με τη χρήση της προαναφερθείσας ανισότητας.

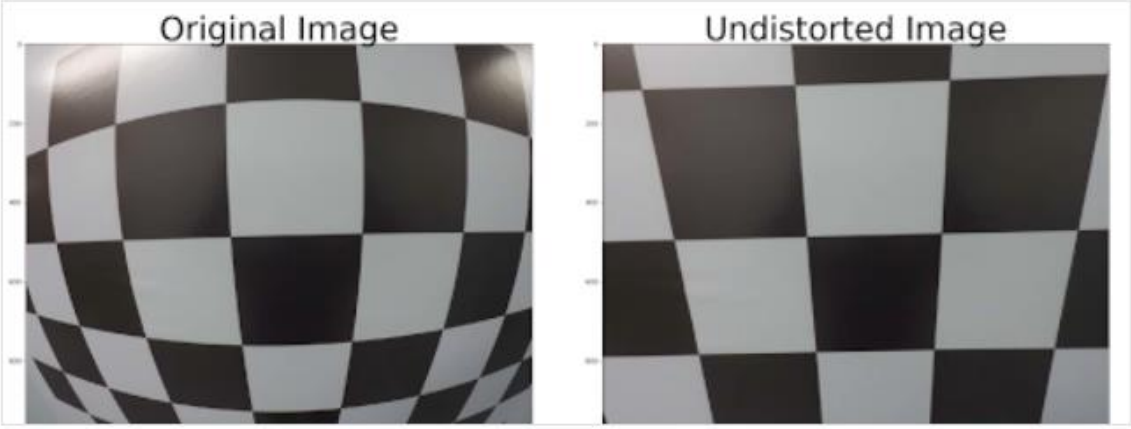
Με αντίστοιχο τρόπο δουλεύουν και τα ανθρώπινα μάτια. Παρατηρούν μία σκηνή και τον χώρο από δύο διαφορετικές οπτικές γωνίες και, συγκρίνοντας τη διαφορά της απόστασης μεταξύ των αντικειμένων στον χώρο, δημιουργείται μία αρχική εκτίμηση. Για να προβλέψει το βάθος ένα σύστημα δύο καμερών, τοποθετείται σε συγκεκριμένη απόσταση και δημιουργεί ζεύγος φωτογραφιών με τις φωτογραφίες να ορίζονται ως αριστερή και δεξιά, οι οποίες δημιουργούνται ταυτόχρονα. Με τη θέση που έχουν οι κάμερες, χρησιμοποιώντας διάφορες μεθόδους, μπορούμε να δημιουργήσουμε μία

γεωμετρική ερμηνεία της σκηνής. Για να επιτευχθούν, όμως, σωστά αποτελέσματα σε αυτά τα συστήματα, απαιτείται μία βαθμονόμηση (calibration) των εικόνων.

Στην πράξη, η εκτίμηση βάθους με δύο κάμερες γίνεται σε τέσσερα στάδια:

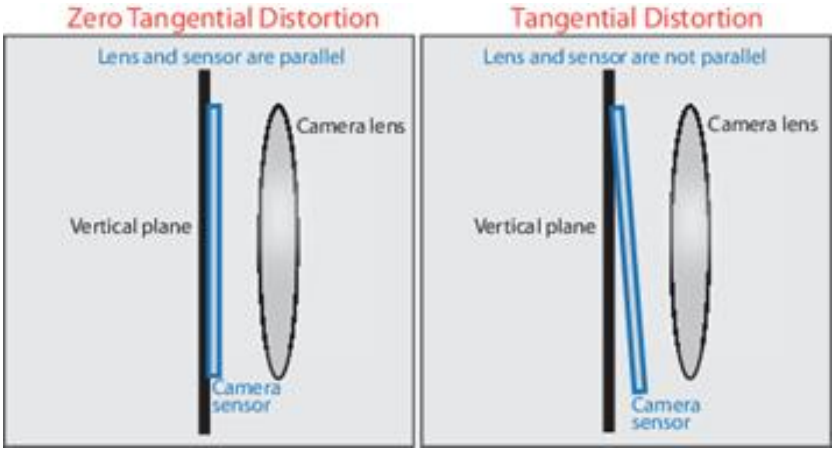
- Διόρθωση παραμόρφωσης (Distortion Correction): Αφαιρεί τις παραμορφώσεις των φωτογραφιών που δημιουργούνται από τον φακό της κάμερας.
- Ρύθμιση γωνιών και αποστάσεων μεταξύ των καμερών κατά την επεξεργασία: Οι εικόνες, για να είναι χρήσιμες, πρέπει να είναι ευθυγραμμισμένες ως προς την κατεύθυνσή τους, αλλά και να βρίσκονται στο ίδιο ύψος.
- Διαδικασία αντιστοίχισης σημείων: Σημεία-κλειδιά πάνω στο πλάνο που έχουν δημιουργηθεί οι φωτογραφίες από την αριστερή και δεξιά κάμερα πρέπει να οριστούν ως σημεία αντιστοίχισης, για να δημιουργηθεί ένας «χάρτης ανισοτήτων» που θα περιέχει τις ανισότητες μεταξύ αριστερής και δεξιάς φωτογραφίας.
- Μετά από σωστή γεωμετρική τοποθέτηση των δύο καμερών, το αποτέλεσμα είναι ένας τριγωνοποιημένος χάρτης ανισότητας των δύο εικόνων. Αυτό το στάδιο ονομάζεται επαναπροβολή (reprojection) και δημιουργεί έναν «χάρτη βάθους» που απαιτείται για τη δημιουργία της τρισδιάστατης σκηνής.

Η βαθμονόμηση της κάμερας είναι η εργασία που δημιουργεί εκτιμήσεις πάνω στις παραμέτρους της κάμερας. Αυτές οι παράμετροι χρειάζονται για να δημιουργήσουν μία σύνδεση μεταξύ των τρισδιάστατων σκηνών στον αληθινό κόσμο και των δισδιάστατων φωτογραφιών που παράγονται από την προ-βαθμονομημένη κάμερα. Ο κύριος λόγος για τη βαθμονόμηση της κάμερας είναι η αφαίρεση των παραμορφώσεων που δημιουργούνται από τους φακούς συνδέοντας τα pixels της φωτογραφίας με τις διαστάσεις των αληθινών σκηνών.



Σχήμα 3.4: Παράδειγμα φωτογραφίας μετά από την αφαίρεση της παραμόρφωσης. Στα αριστερά βρίσκεται η φωτογραφία χωρίς επεξεργασία, και στα αριστερά η φωτογραφία μετά την αφαίρεση της παραμόρφωσης. [51]

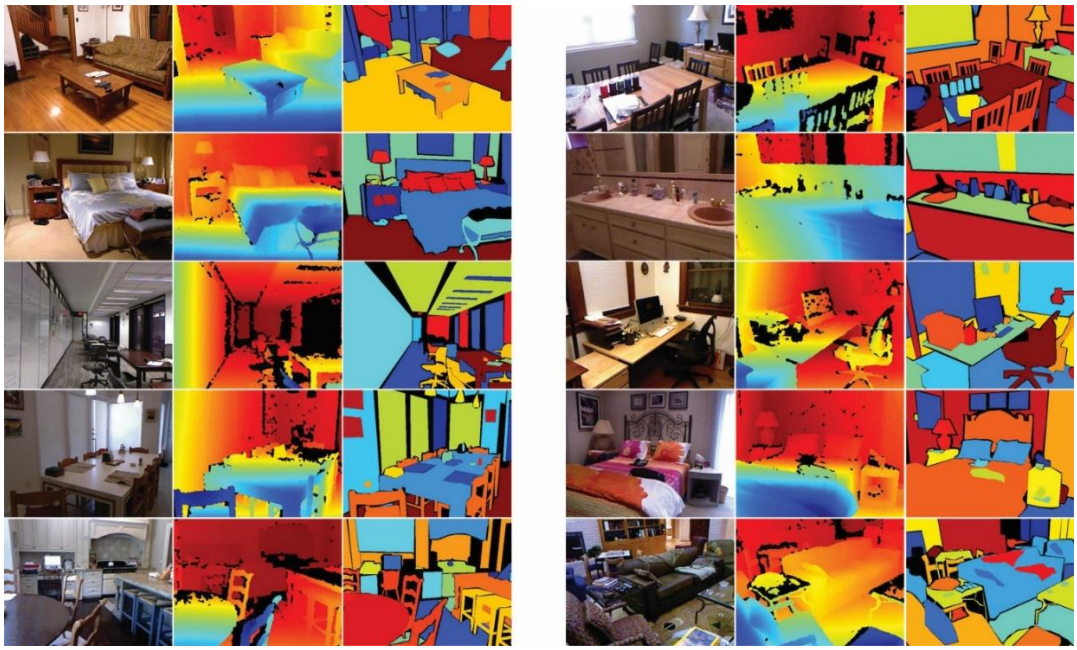
Οι τύποι της παραμόρφωσης χωρίζονται σε δύο κατηγορίες, την παραμόρφωση εφαπτομένης και την παραμόρφωση ακτίνας. Η παραμόρφωση ακτίνας δημιουργείται όταν το φως «λυγίζει» περισσότερο στις άκρες του φακού της κάμερας ενάντια στο κέντρο του, κάνοντας ουσιαστικά τις ευθείες γραμμές να μοιάζουν κυρτές. Από την άλλη, η παραμόρφωση εφαπτομένης δημιουργείται όταν οι φακοί της κάμερας δεν είναι τελείως ευθυγραμμισμένοι με τα υπόλοιπα μέρη της κάμερας. Αυτό έχει ως αποτέλεσμα να δείχνει αντικείμενα πιο κοντά ή πιο μακριά στον φακό της κάμερας από όπου βρίσκονται στην πραγματικότητα [51].



Σχήμα 3.5: Σύγκριση μεταξύ μηδενικής παραμόρφωσης της εφαπτομένης και μερικής παραμόρφωσης της εφαπτομένης με τα μέρη της κάμερας να ορίζονται ως φακός κάμερας και αισθητήρας κάμερας [51].

3.3 Datasets

Όταν αναφερόμαστε στον όρο Dataset (σύνολα δεδομένων), αναφερόμαστε σε μία συλλογή δεδομένων διαφόρων ειδών που είναι αποθηκευμένη σε ψηφιακή μορφή. Όπως έχει προαναφερθεί, τα σύνολα δεδομένων χρησιμοποιούνται για να γίνει η διαδικασία εκπαίδευσης και αποτίμησης ενός μοντέλου μηχανικής μάθησης. Συνήθως, τα δεδομένα μέσα σε αυτά τα σύνολα περιέχουν πληροφορία σε μορφή διανύσματος, αλλά στα περισσότερα προβλήματα που έχουν να κάνουν με την όραση υπολογιστή, τα δεδομένα σε αυτά τα σύνολα είναι πληροφορίες που αντιπροσωπεύουν εικόνες σε διάφορες μορφές.



Σχήμα 3.6: Παραδείγματα τριπλέτας με RGB φωτογραφία, τον χάρτη του βάθους αλλά και την ταυτοποίηση της εικόνας από το dataset NYU-V2 [15].

3.3.1 NYU Depth Dataset

Το NYU Depth είναι ένα σύνολο δεδομένων μεγάλου μεγέθους που περιέχει βίντεο-σεκάνς από διάφορες σκηνές εσωτερικού χώρου. Μέσα στο σύνολο δεδομένων εμπεριέχονται φωτογραφίες σε διάφορες μορφές, το αντίστοιχο εκτιμώμενο βάθος, αλλά και δεδομένα που εμπεριέχουν τις αληθινές τιμές ταυτοποίησης εικόνας (image segmentation ground truth). Το σύνολο δεδομένων έχει δύο διαφορετικές εκδόσεις. Η πρώτη έκδοση εμπεριέχει 2347 από τις προαναφερθέντες τριπλέτες (φωτογραφία, βάθος, ταυτοποίηση) για 12 κατηγορίες αντικειμένων. Η δεύτερη έκδοση περιέχει ένα σύνολο 407,024 φωτογραφιών, παραγόμενων από εκατοντάδες σεκάνς βίντεο και ταξινομεί τα αντικείμενα που απεικονίζονται σε αυτές σε 894 διαφορετικές κλάσεις [52, 15].

3.3.2 KITTI

Το σύνολο δεδομένων KITTI (Karlsruhe Institute of Technology and Toyota Technological Institute) είναι ένα από τα πιο διαδεδομένα σύνολα δεδομένων για μοντέλα και αλγόριθμους Μηχανικής Μάθησης που ασχολούνται με την όραση υπολογιστή. Το συγκεκριμένο σύνολο δεδομένων εμπεριέχει δεδομένα και πληροφορίες πέρα από αυτές της εικόνας, αλλά για τις ανάγκες της εργασίας θα αναφερθούμε στα "raw" δεδομένα και στο sub-dataset (υποσύνολο δεδομένων) KITTI 2012 STEREO και KITTI 2015 STEREO. Τα δεδομένα προέκυψαν με τη δημιουργία μίας πλατφόρμας πάνω σε ένα αυτοκίνητο και οδηγώντας σε πόλεις, αγροτικές περιοχές, αλλά και δρόμους ταχείας κυκλοφορίας. Η πλατφόρμα αυτή εμπεριέχει δύο συστήματα καμερών υψηλής ανάλυσης που καταγράφουν τα δεδομένα σε έγχρωμες αλλά και ασπρόμαυρες εικόνες, αλλά και εξειδικευμένα συστήματα σαρωτών λέιζερ και συστημάτων εντοπισμού.

Τα "raw" δεδομένα περιέχουν φωτογραφίες σε ζεύγη που απεικονίζουν διάφορες σκηνές σε διαφορετικά πεδία. Τα ζευγάρια είναι αποτέλεσμα εξαγωγής από σεκάνς βίντεο και χωρίζονται σε πέντε κατηγορίες, Residential, Road, Campus, City και Person. Οι φωτογραφίες της κατηγορίας City απεικονίζουν κεντρικά, αλλά και προαστιακά μέρη μίας πόλης. Η κατηγορία Residential παρουσιάζει κατοικημένες περιοχές και σπίτια χαμηλού υψομέτρου. Στην κατηγορία road, οι φωτογραφίες απεικονίζονται από περιφερειακούς ή επαρχιακούς δρόμους, ενώ η κατηγορία Campus εμπεριέχει φωτογραφίες που παρουσιάζουν οδήγηση αυτοκινήτου εντός μίας Πανεπιστημιούπολης. Τέλος, η κατηγορία Person παρουσιάζει φωτογραφίες σε σταθερό σημείο θέασης και ανθρώπους να διασχίζουν το οπτικό πεδίο της κάμερας. Τέλος στα "raw" δεδομένα εμπεριέχονται τα αρχεία βαθμονόμησης (calibration files), που μπορούν να χρησιμοποιηθούν για τη βέλτιστη χρήση των φωτογραφιών [53].

Το KITTI 2012 είναι μια ανανεωμένη έκδοση των "raw" δεδομένων. Το συγκεκριμένο dataset έρχεται προετοιμασμένο με την πληροφορία που χρειάζεται για να επιλύσει διάφορα προβλήματα, όπως είναι η στέρεο ταυτοποίηση εικόνας, η οπτική κίνηση, αλλά και άλλα προβλήματα της όρασης υπολογιστή. Με τη στέρεο ταυτοποίηση εικόνας να είναι το πιο χρήσιμο κομμάτι για την εκτίμηση του βάθους, το σύνολο δεδομένων προσφέρει 194 ζεύγη φωτογραφιών εκπαίδευσης και 195 ζεύγη φωτογραφιών αποτίμησης. Οι φωτογραφίες είναι ανάλυσης 1240 × 376 μετά από διόρθωση με τις αληθινές τιμές (ground truth) [13].

Σχήμα 3.7: Παράδειγμα φωτογραφίας και αντίστοιχο βάθος από το dataset KITTI 2012. Η εικόνα απεικονίζεται από σημείο θέασης εντός δρόμου με σχετικά χαμηλή απόσταση από το έδαφος, και παρουσιάζει έναν δρόμο με διάφορα αυτοκίνητα παρκαρισμένα στην δεξιά πλευρά του δρόμου. Το μέρος που απεικονίζεται φαίνεται να είναι εντός της πόλης [13].

Αντίστοιχα με το KITTI 2012, δημιουργήθηκε το KITTI 2015. Όπως και το KITTI 2012, η έκδοση του 2015 είναι ένα σύνολο δεδομένων – υποσύνολο των "raw" δεδομένων του KITTI dataset. Σε αντίθεση με την έκδοση του 2012, η συγκεκριμένη περιέχει 200 ζεύγη φωτογραφιών για την εκπαίδευση και 200 ζεύγη φωτογραφιών για την αποτίμηση. Οι αληθινές τιμές του συγκεκριμένου dataset μπορούν να χρησιμοποιηθούν για την εκτίμηση της διαφοράς, την εκτίμηση του βάθους, της οπτικής κίνησης και πολλά άλλα προβλήματα της όρασης υπολογιστή [54, 55, 56].

3.3.3 SceneFlow Datasets

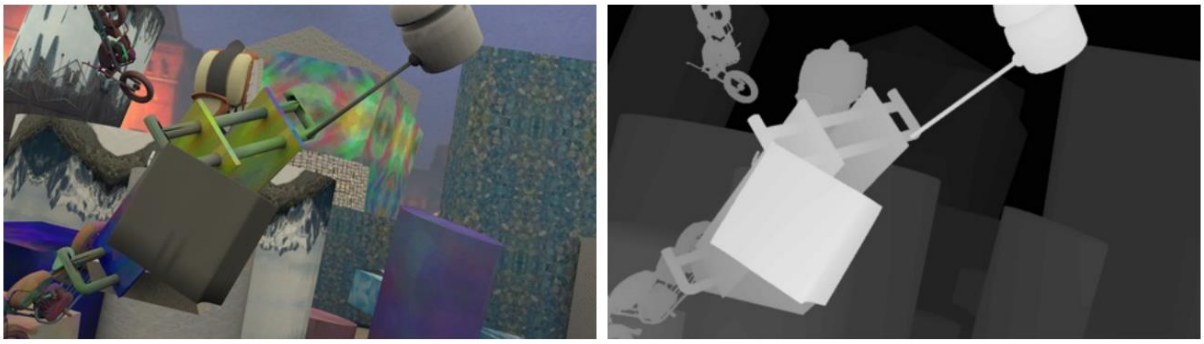
Τα SceneFlow Datasets είναι μία συλλογή από σύνολα δεδομένων που χρησιμοποιούνται για την εκτίμηση της ανισότητας και της οπτικής κίνησης. Το SceneFlow είναι άξιο αναφοράς λόγω του μεγάλου όγκου δεδομένων που εμπεριέχει, καθώς και την ποικιλία των δεδομένων. Ενώ πριν αναφερθήκαμε σε συγκεκριμένα σύνολα δεδομένων με φωτογραφίες, όπου οι φωτογραφίες απεικονίζουν μέρη και τοπία συγκεκριμένης φύσης, το SceneFlow δεν είναι ένα απλό σύνολο δεδομένων. Το SceneFlow είναι ένας συνδυασμός τριών συνόλων δεδομένων που ιδρύουν ένα ενιαίο μεγάλου μεγέθους dataset, στο οποίο οι φωτογραφίες τους απεικονίζουν μία ευρεία γκάμα σκηνών. Αποτελείται συνολικά από 39000 ζεύγη φωτογραφιών ανάλυσης 960x540 pixel, ενώ οι σκηνές είναι ψηφιακά φτιαγμένες με τα εργαλεία της σουίτας γραφικών 3Δ Blender [57].

Πίνακας 3.1: Σύγκριση των Dataset KITTI, NYUV2 και SceneFlow, καθώς και διαφόρων χαρακτηριστικών. Στις πρώτες τρεις σειρές γίνεται η σύγκριση του συνόλου δεδομένων, στην τέταρτη σειρά συγκρίνονται οι αναλύσεις των φωτογραφιών. Στις τελευταίες έξι σειρές παρουσιάζονται τα υπόλοιπα χαρακτηριστικά καθώς και σε τι βαθμό εμπεριέχεται το ground truth συγκριτικά με κάθε πρόβλημα [58].

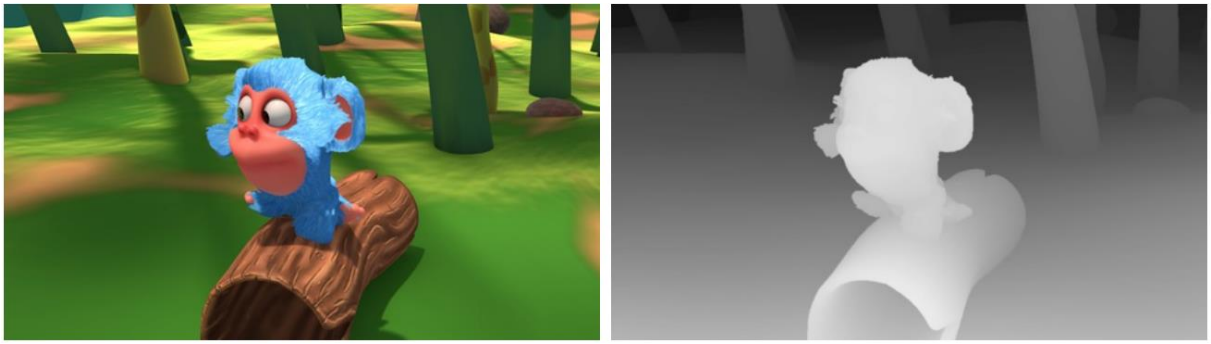
Dataset	KITTI Benchmark Suite		NYU2	SceneFlow Dataset		
	2012	2015		FlyingThings3D	Monkaa	Driving
#Training Frames	194	200	1449	21818	8591	4392
#Test frames	195	200	--	4248	--	--
#Training Scenes	194	200	464	2247	8	1
Resolution	1226*370	1242*375	640*480	960*540		
Ανισότητα/Βάθος	Αραιή	Αραιή	Ναι	Ναι	Ναι	Ναι
Διαφορά Ανισότητας	Όχι	Όχι	Όχι	Ναι	Ναι	Ναι
Οπτική κίνηση	Αραιή	Αραιή	Όχι	Ναι	Ναι	Ναι
Ταυτοποίηση	Όχι	Όχι	Ναι	Ναι	Ναι	Ναι
Όρια κίνησης	Όχι	Όχι	Όχι	Ναι	Ναι	Ναι
Φυσικότητα	Ναι	Ναι	Ναι	Όχι	Όχι	Ναι

Τα υποσύνολα δεδομένων στα οποία χωρίζεται το SceneFlow είναι τα εξής:

- FlyingThings3D: Το συγκεκριμένο υποσύνολο δεδομένων αποτελείται από ζεύγη φωτογραφιών που δημιουργήθηκαν σε ψηφιακό περιβάλλον. Οι φωτογραφίες απεικονίζουν αντικείμενα καθημερινής χρήσης να ίπτανται σε διαφορετικές κατευθύνσεις. Έχουν δημιουργηθεί περίπου 25,000 ζεύγη φωτογραφιών με τις αληθινές τιμές (στόχους). Αντίθετα με άλλα υποσύνολα δεδομένων των οποίων τα δεδομένα εξειδικεύονται σε συγκεκριμένες θεματικές, το FlyingThings3D περιέχει μία ποικιλία θεματικών με περίπου 200 στατικά αντικείμενα τυχαίων σχημάτων.
- Monkaa: Το δεύτερο dataset που εμπεριέχεται μέσα στο SceneFlow είναι το Monkaa. Είναι δημιουργημένο στο Blender, με αντικείμενα (assets) ανοιχτού κώδικα από την ταινία μικρού μήκους Monkaa. Η επιλογή του συγκεκριμένου dataset είναι σημαντική, διότι μέσα στο ψηφιακό περιβάλλον εμπεριέχονται βατές και αργές κινήσεις του αντικειμένου σε μορφή σεκάνς φωτογραφιών, όπως επίσης και τρίχωμα πάνω στο αντικείμενο που απεικονίζεται.
- Driving: Οι σκηνές του Driving συνόλου δεδομένων προσπαθούν να μιμηθούν τη φυσικότητα αλλά και τον τύπο δεδομένων που προσφέρει το Dataset KITTI. Χρησιμοποιεί ψηφιακά μοντέλα αυτοκινήτων που χρησιμοποιούνται και στο σύνολο FlyingThings3D, όπως επίσης και ψηφιακά δέντρα με βαθιά λεπτομέρεια [58].



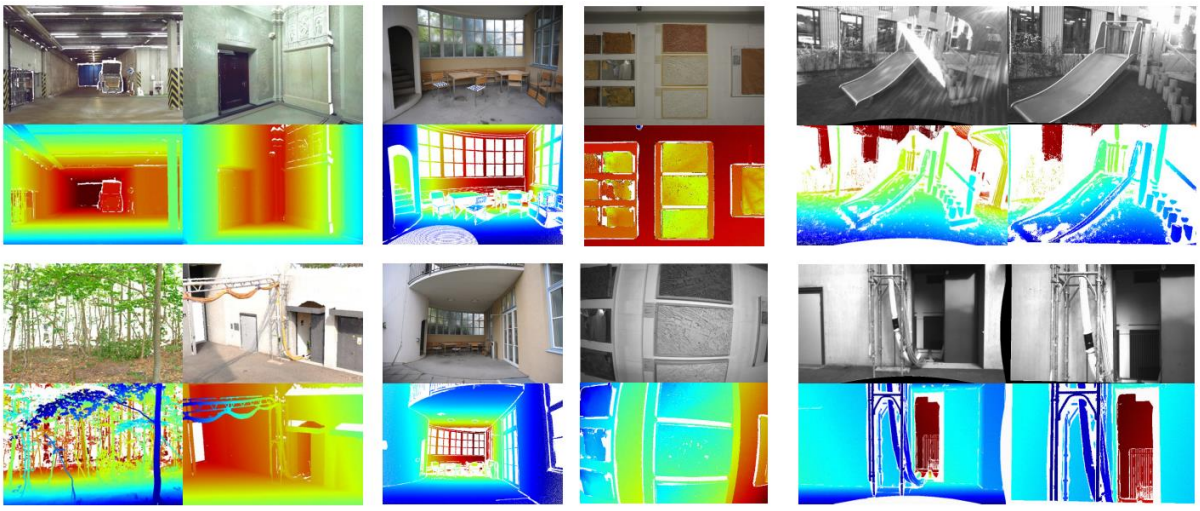
Σχήμα 3.8: Παράδειγμα από το Dataset FlyingThings3D του SceneFlow. Η πρώτη φωτογραφία απεικονίζει την αριστερή φωτογραφία ενός ζεύγους φωτογραφιών που απεικονίζει ιπτάμενα αντικείμενα και η δεξιά φωτογραφία την πραγματική ανισότητα μεταξύ του προαναφερθέντος ζεύγους [58].



Σχήμα 3.9: Παράδειγμα από το Dataset Monkee του SceneFlow. Η πρώτη φωτογραφία απεικονίζει την αριστερή φωτογραφία ενός ζεύγους και η δεξιά φωτογραφία την πραγματική ανισότητα μεταξύ του προαναφερθέντος ζεύγους [58].

3.3.4 ETH3D

Το ETH3D είναι ένα σύνολο δεδομένων πολλαπλών οπτικών γωνιών που χρησιμοποιείται σε μία πληθώρα εργασιών της όρασης υπολογιστή. Το σύνολο δεδομένων συμπεριλαμβάνει μία ποικιλία σκηνών εσωτερικού και εξωτερικού χώρου, όπως επίσης και την αληθινή γεωμετρία του χώρου. Οι αληθινές γεωμετρικές τιμές των σκηνών δημιουργήθηκαν με τη χρήση σαρωτή λέιζερ υψηλής ευκρίνειας, ενώ για τις καταγραφές των φωτογραφιών των σκηνών χρησιμοποιήθηκε μία κάμερα DSLR, όπως και ένα συγχρονισμένο σύστημα πολλαπλών καμερών με διαφορετικές οπτικές γωνίες. Η χρήση της DSLR βοηθά το dataset να καταγράψει φωτογραφίες σε υψηλότερη ανάλυση και με καλύτερη ευκρίνεια, κάτι που δεν υπάρχει στα προαναφερθέντα σύνολα δεδομένων. Αυτό έχει ως αποτέλεσμα τα δεδομένα του ETH3D να μπορούν να χρησιμοποιηθούν για την κατασκευή χώρων 3 διαστάσεων με περισσότερες λεπτομέρειες [59].



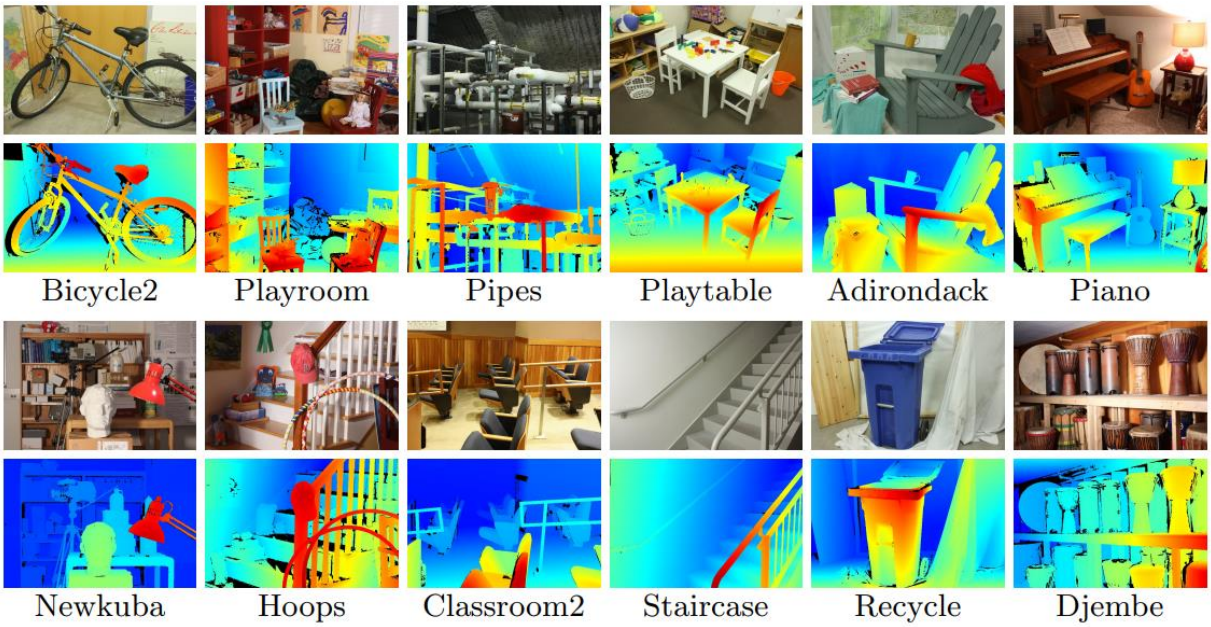
(a) Scene type (b) View point (c) Camera type (d) Field of view

Σχήμα 3.10: Παραδείγματα από το ETH3D dataset που παρουσιάζουν την καταγεγραμμένη φωτογραφία και το αντίστοιχο βάθος του χώρου. Στο (a) παρουσιάζονται σκηνές εσωτερικού και εξωτερικού χώρου. Στο (b) παρουσιάζονται φωτογραφίες από DSLR κάμερα σε διαφορετική οπτική γωνία. Στο (c) φαίνεται η σύγκριση μεταξύ εικόνας και βάθους από κάμερα DSLR (πάνω) και του συστήματος πολλαπλών καμερών (κάτω). Στο (d) φαίνονται φωτογραφίες από το σύστημα πολλαπλών καμερών.

3.3.5 Middlebury Datasets

Τα σύνολα δεδομένων Middlebury αποτελούνται από διπλά ζεύγη σκηνών υψηλής ανάλυσης. Μέσα στο σύνολο δεδομένων εμπεριέχονται αληθινές τιμές ανισότητας μεγάλης ακρίβειας που ελήφθησαν με τη χρήση τεχνικών που χρησιμοποιούν οργανωμένο φωτισμό χωρίς οι προβολείς φωτός να απαιτούν βαθμονόμηση [60].

Έχουν υπάρξει διάφορες εκδόσεις για το σύνολο δεδομένων Middlebury, οι οποίες είναι χωρισμένες ανά χρονιά. Εκδόσεις έχουν δημοσιευθεί το 2001, το 2003, το 2005, το 2006, το 2014 και το 2021. Η τελευταία χρησιμοποίησε φορητή συσκευή (mobile) για να δημιουργηθεί το dataset [61]. Πιο συγκεκριμένα, στην έκδοση του 2014, εμπεριέχονται 33 μικρότερα υποσύνολα δεδομένων που περιέχουν ζεύγη φωτογραφιών υψηλής ανάλυσης με σκηνές εσωτερικού χώρου. Το σύστημα που χρησιμοποιήθηκε για την καταγραφή των σκηνών είναι ένα σύστημα με δύο κάμερες DSLR και δύο κάμερες τύπου “point-and-shoot”. Επιπροσθέτως, μέσα στο σύνολο δεδομένων εμπεριέχονται οι αληθινές τιμές βάθους και ανισότητας [62].



Σχήμα 3.11: Αριστερές φωτογραφίες από κάθε ζεύγος φωτογραφιών του Middlebury (2014) dataset με τον πίνακα ανισότητας για κάθε υποσύνολο δεδομένων. Ο τύπος αντικειμένων που παρουσιάζεται σε κάθε υποσύνολο (πάνω φωτογραφία) και η αντίστοιχη ανισότητα (κάτω φωτογραφία) [62].

Κεφάλαιο 4ο: Τεχνολογίες, Πλατφόρμες & Μηχανήματα

Στο συγκεκριμένο κεφάλαιο θα παρουσιαστούν οι τεχνολογίες, οι βιβλιοθήκες και οι πλατφόρμες που χρησιμοποιήθηκαν για να γίνουν οι δοκιμές των χρησιμοποιημένων μοντέλων για την εργασία. Σε συνδυασμό με τις τεχνολογίες, θα γίνει αναφορά στα μηχανήματα με τα οποία έγιναν οι δοκιμές της εργασίας, αλλά και αναφορά στα τεχνολογικά και υλικά εμπόδια που βρέθηκαν κατά τη διάρκεια των δοκιμών. Τέλος, θα παρουσιαστούν συγκεκριμένα κομμάτια κώδικα που σχετίζονται με τις προαναφερθείσες τεχνολογίες και βιβλιοθήκες.

4.1 Μηχανήματα & υλικό.

Με τον όρο Μηχανήματα αναφερόμαστε στους υπολογιστές που χρησιμοποιήθηκαν στις δοκιμές, αλλά και στο υλικό τους, το οποίο περιλαμβάνει τα εξαρτήματα του κάθε Μηχανήματος. Λόγω της φύσης του συγκεκριμένου προβλήματος, η χρήση μεθόδων και μοντέλων μηχανικής μάθησης κάνει την υλοποίηση πιο εύκολη. Όμως, με τη χρήση των τεχνικών Μηχανικής Μάθησης και των Νευρωνικών μοντέλων, δημιουργείται ένα μεγάλο μειονέκτημα, το οποίο αφορά το υπολογιστικό κόστος των υλοποιήσεων. Όπως έχει προαναφερθεί στο κείμενο, τα μοντέλα μηχανικής μάθησης έχουν «βαρύ» υπολογιστικό κόστος, και πιθανόν πολύ μεγαλύτερο έναντι των άλλων εφαρμογών. Για αυτόν τον λόγο, για να υλοποιηθούν οι συγκεκριμένες μέθοδοι και τα μοντέλα, απαιτούνται SOTA μηχανήματα και υλικά.

Υπάρχουν μεγάλες απαιτήσεις σε διάφορα μέτωπα υλικών. Αρχικά, υπάρχει μεγάλη ανάγκη για αποθηκευτικό χώρο. Λόγω της φύσης του προβλήματος, τα σύνολα δεδομένων που χρειάζονται για να γίνει η εκπαίδευση των μοντέλων είναι πολύ μεγάλα σε μέγεθος. Τα σύνολα δεδομένων που χρησιμοποιήθηκαν, όπως έχει προαναφερθεί, περιέχουν φωτογραφίες, πολλές φορές σε ζεύγη και πολλές φορές σε σεκάνς – δημιουργημένα από βίντεο. Επιπρόσθετα, στις φωτογραφίες, πολλές φορές εμπεριέχονται οι αληθινές τιμές που χρειάζονται για να γίνει η διαδικασία της εκπαίδευσης. Πέρα από αυτά, οι φωτογραφίες πρέπει να έχουν υψηλή ευκρίνεια και λεπτομέρεια, αυξάνοντας περεταίρω το συνολικό φόρτο που αποδίδουν στους δίσκους, όπου είναι αποθηκευμένα τα datasets.

Πέρα από τη μεγάλη ανάγκη που υπάρχει για αποθηκευτικό χώρο, υπάρχει και μεγάλη ανάγκη σε κύρια μνήμη (μνήμη RAM). Κατά τη διάρκεια των δοκιμών, η χρήση 32GB Ram φάνηκε μη επαρκής σε αρκετές περιπτώσεις και χρειάστηκαν να γίνουν αλλαγές στον κώδικα και στις παραμέτρους των μοντέλων, έτσι ώστε να μπορέσουν να γίνουν οι δοκιμές. Αυτό συνέβαινε διότι έπρεπε πολλές φορές τα μοντέλα που χρησιμοποιήθηκαν να αποθηκεύουν πολλά δεδομένα φωτογραφιών, καθώς και του ίδιου του μοντέλου στην κύρια μνήμη. Πέρα από τα δεδομένα των φωτογραφιών, η εκπαίδευση του μοντέλου είναι μία βαριά υπολογιστική διαδικασία. Διάφορα μοντέλα προσπαθούν να εκμεταλλευτούν όσο το δυνατό περισσότερο χώρο στην κύρια μνήμη, είτε γιατί είναι πιο βολικό στον κώδικα, είτε για να επιτευχθεί η διαδικασία της εκπαίδευσης πιο γρήγορα.

Συνολικά, χρησιμοποιήθηκαν δύο μηχανήματα για την υλοποίηση των μοντέλων. Το πρώτο μοντέλο εκπαιδεύτηκε και αποτιμήθηκε πάνω στον διακομιστή (server) Bastion, ενώ τα υπόλοιπα δύο μοντέλα δοκιμάστηκαν στο τοπικό μηχάνημα. Ο Bastion είναι ένας server του Τμήματος Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙΠΑΕ, ενώ το τοπικό μηχάνημα είναι το προσωπικό μου μηχάνημα.

Πίνακας 4.1: Σύγκριση μεταξύ δύο μηχανημάτων, του Bastion και το τοπικό μηχάνημα.

Μηχάνημα	Bastion	Τοπικό
Motherboard | Μητρική κάρτα	Asus Pro WS C621-64L SAGE	Asus Prime B660-PLUS D4
CPU | Επεξεργαστής	Intel Xeon-Silver 4210	Intel Core i5-13500
#CPU Cores (#Threads) | Πυρήνες Επεξεργαστή	10 (20)	6 + 8 (20)
RAM | Κύρια μνήμη	ECC 32GB	NON ECC 32GB
GPU | Κάρτα γραφικών	Nvidia A4000	RTX 3060 Ti
VRAM | Μνήμη κάρτας γραφικών	16GB	8GB
Storage | Αποθηκευτικός Χώρος	~2TB	240GB + ~2TB + ~4TB (3 δίσκους, έναν SSD, έναν HDD και έναν εξωτερικό HDD).

Στον Πίνακα 4.1, παρουσιάζονται οι σημαντικότερες διαφορές που έχουν στο υλικό τα δύο συστήματα που χρησιμοποιήθηκαν. Οι κύριες διαφορές φαίνονται στην κάρτα γραφικών, στον επεξεργαστή, αλλά και στον αποθηκευτικό χώρο, ενώ μικρότερες είναι οι διαφορές σχετικά με τη μνήμη ram. Επειδή το μηχάνημα Bastion είναι server, τα υλικά που το διακατέχουν τείνουν να προτείνονται για επαγγελματίες και servers. Αντίθετα, στο τοπικό μηχάνημα χρησιμοποιούνται υλικά, προσφερόμενα στην αγορά κατά κύριο λόγο για μηχανήματα καθημερινής χρήσης. Μεγάλη διαφορά υπάρχει στον αποθηκευτικό χώρο των μηχανημάτων, καθώς το τοπικό μηχάνημα έχει τη δυνατότητα να αποθηκεύσει περισσότερα δεδομένα. Παρόμοιες φαίνονται να είναι οι κύριες μνήμες των μηχανημάτων, με τη διαφορά ότι στο μηχάνημα bastion οι μνήμες είναι τύπου ECC (Error Correction Code). Οι ECC μνήμες είναι ένας τύπος μνημών που χρησιμοποιούνται κατά κύριο λόγο από servers, διότι έχουν τη δυνατότητα να εντοπίσουν και να επιδιορθώσουν σφάλματα στη μνήμη [63]. Όμως, τον σημαντικότερο ρόλο έπαιξε η διαφορά στη VRAM που έχουν αυτά τα δύο μηχανήματα, καθώς η VRAM της RTX 3060 Ti κάρτας γραφικών κρίθηκε ανεπαρκής στις πρώτες δοκιμές του πρώτου μοντέλου που υλοποιήθηκε.

4.2 Πλατφόρμες & Τεχνολογίες

Έχοντας παρουσιάσει το υλικό που χρησιμοποιήθηκε για την υλοποίηση των μοντέλων, σε αυτό το κεφάλαιο θα παρουσιαστούν οι τεχνολογίες που χρησιμοποιήθηκαν για την εκπαίδευση και την αποτίμηση των μοντέλων. Επιπροσθέτως, θα παρουσιαστούν όλες οι υποστηρικτικές τεχνολογίες που χρειάστηκαν για να δημιουργηθούν τα «λειτουργικά» κομμάτια, αλλά και αποσπάσματα κώδικα. Πριν γίνει η εισαγωγή στις τεχνολογίες, πρέπει να αναφερθεί ότι τα λειτουργικά συστήματα που χρησιμοποιήθηκαν για την εργασία είναι βασισμένα στο λειτουργικό Linux [64].

4.2.1 Anaconda

Η Anaconda είναι μία δωρεάν πλατφόρμα ανοιχτού κώδικα που επιτρέπει στον χρήστη να γράψει και να εκτελέσει κώδικα της γλώσσας προγραμματισμού Python [65]. Δημιουργήθηκε το 2012 από τον Peter Wang και τον Travis Oliphant. Η δημιουργία της βασίστηκε στην ανάγκη της αγοράς να εισάγει τη γλώσσα προγραμματισμού Python στην επιχειρηματική ανάλυση δεδομένων, αλλά και στην έλλειψη αντίστοιχων εργαλείων ανοιχτού κώδικα [66]. Η συγκεκριμένη πλατφόρμα χρησιμοποιείται από πάνω από 30 εκατομμύρια ανθρώπους παγκοσμίως και είναι διαθέσιμη για τα περισσότερα λειτουργικά συστήματα.

Η επιλογή της συγκεκριμένης πλατφόρμας για την εκτέλεση κώδικα python βασίζεται σε διάφορους λόγους. Ο πιο σημαντικός είναι η ευχρηστία που παρέχει στη διαχείριση πακέτων και βιβλιοθηκών της γλώσσας python που διευκολύνουν την υλοποίηση των προγραμμάτων. Πιο συγκεκριμένα, η πλατφόρμα βοηθάει στη δημιουργία ψηφιακών περιβαλλόντων, με διαφορετικές εκδόσεις της python αλλά και των αντίστοιχων βιβλιοθηκών. Διάφορες εκδόσεις της πλατφόρμας είναι το Conda που επιτρέπει την εγκατάσταση, ενημέρωση και διαγραφή πακέτων με τη χρήση εντολών τερματικού. Επιπρόσθετα, στο Conda υπάρχει και το Miniconda, που είναι μία μικρότερη έκδοση του Anaconda [67].

4.2.2 Python

Η Python είναι μία αντικειμενοστραφής γλώσσα προγραμματισμού υψηλού επιπέδου γενικής χρήσης. Η δυναμικότητα που κατέχει βάσει του υψηλού επιπέδου της την καθιστά πολύ ελκυστική επιλογή για ταχεία ανάπτυξη εφαρμογών, αλλά και για τη δημιουργία scripts και ενδιάμεσων μερών μεταξύ εφαρμογών. Η γλώσσα python έχει απλό και ευανάγνωστο συντακτικό. Αυτό έχει σαν αποτέλεσμα να μειώνει το κόστος συντήρησης σε προγράμματα που είναι γραμμένα σε python. Επιπροσθέτως, η Python υποστηρίζει πακέτα και βιβλιοθήκες κάνοντας τα προγράμματα πιο ελαστικά και αυτόνομα, ενθαρρύνοντας έτσι την επαναχρησιμοποίηση κώδικα [68].

Από τη δημιουργία της το 1991, η Python έχει γίνει μία από τις δημοφιλέστερες γλώσσες προγραμματισμού για εταιρίες startup και προγραμματιστές. Ως τον Φεβρουάριο του 2023 κατέχει μέρισμα χρήσης 15.5% βάσει του μετρητή TIOBE. Η γλώσσα χρησιμοποιείται από διάφορους τεχνολογικούς κολοσσούς, όπως είναι η Amazon, η Google και η Facebook, όπως επίσης και από εφαρμογές, όπως το Spotify, το Netflix, το Dropbox, το Uber κ.ά. [69].

Λόγω της ευκολίας που μπορεί να προσφέρει η Python, βρίσκει εφαρμογές σε πολλά επιστημονικά πεδία της επιστήμης των υπολογιστών. Η Python μπορεί να χρησιμοποιηθεί για την ανάπτυξη εφαρμογών ιστού με τη χρήση πλαισίων προγραμματισμού (framework) όπως το Django, το Pyramid και το Flask. Πέρα από προγραμματισμό εφαρμογών Ιστού, η python μπορεί να χρησιμοποιηθεί για προγραμματισμό απλοϊκών ηλεκτρονικών παιχνιδιών. Επιπρόσθετα, η Python βρίσκει εφαρμογή σε τομείς, όπως τα χρηματοοικονομικά, καθώς αποτελεί ένα από τα σημαντικότερα εργαλεία για αυτοματοποίηση, αλλά και διαχείριση των τάσεων του χρηματοοικονομικού κόσμου. Εφαρμογές βρίσκει επίσης και σε τομείς, όπως το SEO, αλλά και στον σχεδιασμό γραφικών, ενώ κάνει αισθητή την επιρροή της σε άλλες γλώσσες προγραμματισμού, καθώς εμψύχωσε τη δημιουργία πολλών άλλων γλωσσών προγραμματισμού, όπως είναι η Cobra [70], η CoffeeScript [71] και η Go [72]. Τέλος, τα δημοφιλέστερα πεδία για τη γλώσσα Python είναι η Αναλυτική δεδομένων, η T.N. και η Μηχανική Μάθηση [73].

4.2.3 Βιβλιοθήκες και Πακέτα Python

Η Python περιέχει μία μεγάλη πληθώρα εξωτερικών βιβλιοθηκών και πακέτων. Στη συνέχεια της εργασίας θα παρουσιαστούν τα σημαντικότερα και δημοφιλέστερα πακέτα που χρησιμοποιούνται στη Μηχανική Μάθηση. Πιο συγκεκριμένα θα γίνει αναφορά στη βιβλιοθήκη NumPy, OpenCV, TensorFlow και PyTorch. Οι περισσότερες βιβλιοθήκες της python που χρησιμοποιήθηκαν δεν έρχονται προ-εγκατεστημένες με τη γλώσσα, αλλά απαιτείται ένας διαχειριστής πακέτων (Package Manager) για να εγκατασταθούν και να χρησιμοποιηθούν. Ο κύριος διαχειριστής πακέτων που χρησιμοποιείται στην Python είναι ο "pip", αλλά για τις ανάγκες της εργασίας χρησιμοποιήθηκαν και άλλοι διαχειριστές, όπως αυτός που περιέχει η πλατφόρμα Anaconda.

Ένα από τα πιο χρήσιμα πακέτα στην Python είναι το NumPy. Το NumPy είναι ένα από τα βασικότερα πακέτα της Python που χρησιμοποιούνται για επιστημονική πληροφορική (scientific computing). Το NumPy δίνει τη δυνατότητα στον προγραμματιστή να δημιουργήσει και να διαχειριστεί πίνακες πολλών και περίπλοκων διαστάσεων. Επιπρόσθετα, το NumPy υποστηρίζει μία μεγάλη πληθώρα απλών και περίπλοκων πράξεων μεταξύ πινάκων, αλλά και προγραμματιστικές πράξεις, όπως είναι η ταξινόμηση, η επιλογή, αλλά και η αλλαγή «σχήματος» πινάκων [74].

Μία εξίσου σημαντική βιβλιοθήκη της Python είναι το OpenCV. Το OpenCV (Open Source Computer Vision Library) είναι μία βιβλιοθήκη ανοιχτού κώδικα που χρησιμοποιείται για εργασίες όρασης υπολογιστή και Μηχανικής Μάθησης. Δημιουργήθηκε με σκοπό να παρέχει κοινές υποδομές για εφαρμογές όρασης υπολογιστή, αλλά και για να επιταχύνει τη μηχανική αντίληψη στα εμπορικά προϊόντα. Η βιβλιοθήκη περιέχει πάνω από 2,500 αλγορίθμους συμπεριλαμβανομένων διάφορων SOTA και κλασσικών αλγόριθμων Μηχανικής Μάθησης. Πέρα από την Python, παρέχεται και για γλώσσες σαν την C++ και την Java [75].

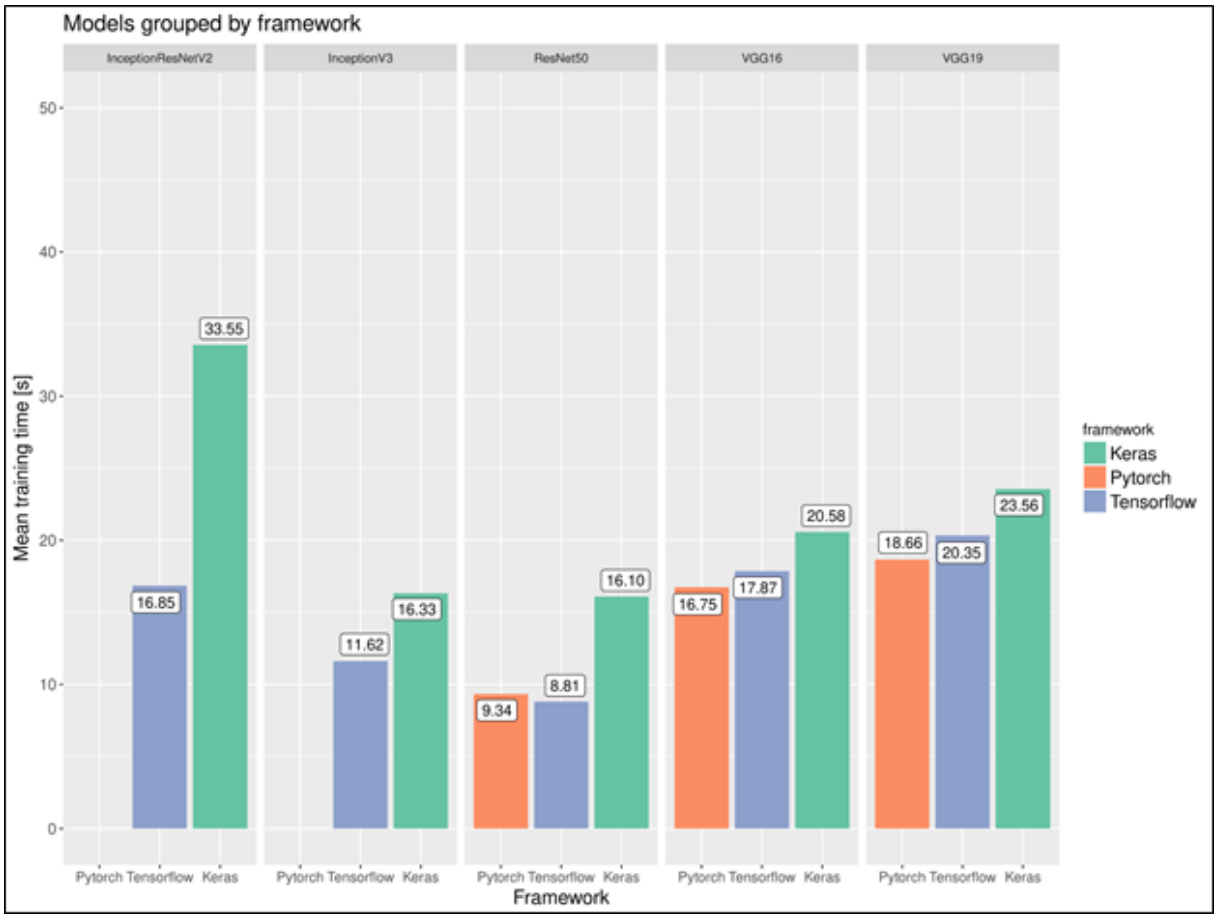
Πέρα από τις προαναφερθείσες βιβλιοθήκες, που είναι περισσότερο υποστηρικτικές για τον προγραμματιστή για να εκτελέσει διάφορες λειτουργίες, υπάρχουν δύο κύριες πλατφόρμες που χρησιμοποιούνται για εφαρμογές Μηχανικής Μάθησης. Αυτές οι πλατφόρμες είναι η TensorFlow και η PyTorch. Παραδείγματα κώδικα των συγκεκριμένων τεχνολογιών θα παρουσιαστούν στο επόμενο κεφάλαιο της εργασίας, σε συνδυασμό με το αντίστοιχο μοντέλο που υλοποιείται.

Η TensorFlow είναι μία πλατφόρμα Μηχανικής Μάθησης που επιτρέπει στον προγραμματιστή να δημιουργήσει μοντέλα επιπέδου «παραγωγής». Δημιουργήθηκε με σκοπό να απλουστεύσει τη δημιουργία των μοντέλων και αλγορίθμων μηχανικής μάθησης, καθώς έρχεται προ-εγκατεστημένη με προ-εκπαιδευμένα μοντέλα. Η απλούστευση του προγραμματισμού των μοντέλων μπορεί να υλοποιηθεί με τη χρήση του «υψηλού επιπέδου» API Keras, ενώ για πιο περίπλοκες απαιτήσεις παρέχονται λύσεις και τρόποι για τη δημιουργία αντίστοιχων μοντέλων και αλγορίθμων. Επιπροσθέτως, η TensorFlow έρχεται εξοπλισμένη με SOTA μοντέλα και αρχιτεκτονικές, ενώ καθίσταται ιδανική για ερευνητικούς σκοπούς [76, 77]. Άξια αναφοράς είναι δυνατότητα που δίνει η πλατφόρμα στην εκμετάλλευση πόρων και επεξεργαστικής ισχύος που μπορεί να αποκτήσει με τη χρήση των GPU ή TPU. Αυτή η τεχνική ονομάζεται Accelerated Computing και βασίζεται σε τεχνολογίες παράλληλου προγραμματισμού (parallel computing), όπως είναι η τεχνολογία CUDA της Nvidia [78, 79].

Η PyTorch είναι μία πλατφόρμα ανοιχτού κώδικα για Μηχανική Μάθηση βασισμένη στη γλώσσα προγραμματισμού Python και στη βιβλιοθήκη Torch. Η Torch είναι μία βιβλιοθήκη ανοιχτού κώδικα για Μηχανική Μάθηση που χρησιμοποιείται περισσότερο για τη δημιουργία νευρωνικών δικτύων βαθιάς μάθησης και είναι γραμμένη στη γλώσσα "Lua". Η PyTorch υποστηρίζει πάνω από 200 διαφορετικούς μαθηματικούς τύπους, ενώ είναι μία πολύ δημοφιλής επιλογή για εφαρμογές Μηχανικής

Μάθησης, καθώς καθιστά πιο απλή τη δημιουργία και υλοποίηση νευρωνικών δικτύων και αλγορίθμων Μηχανικής Μάθησης [80].

Η PyTorch ως αυτοτελής βιβλιοθήκη έχει πολλά πλεονεκτήματα, αλλά πολύ συχνά τείνει να συγκρίνεται με την πλατφόρμα TensorFlow. Η κύρια διαφορά τους βασίζεται στην αντικειμενοστραφή φύση που έχει η PyTorch έναντι των περισσότερων επιλογών που μπορεί να προσφέρει η TensorFlow για την υλοποίηση των μοντέλων της. Η PyTorch καλύπτει ανάγκες για ταχύτητα, ελαστικότητα αλλά και δυνατότητες εντοπισμού σφαλμάτων των προγραμματιστών. Από την άλλη πλευρά, η TensorFlow προσφέρει μία πληθώρα εκπαιδευμένων μοντέλων, καθώς και καλύτερη γραφική απεικόνιση επιτρέποντας στους προγραμματιστές να εντοπίσουν ευκολότερα τυχόν σφάλματα, αλλά και να παρατηρήσουν καλύτερα τη διαδικασία της εκπαίδευσης. Επιπρόσθετα, η TensorFlow κατέχει υπεροχή στην ανάπτυξη μοντέλων μηχανικής μάθησης για παραγωγή, ενώ η PyTorch υποστηρίζει διάφορες μεθόδους που μπορούν να κάνουν τη διαδικασία της εκπαίδευσης γρηγορότερη [81].



Σχήμα 4.1: Σύγκριση μεταξύ Keras, PyTorch και TensorFlow σε διάφορα μοντέλα μηχανικής Μάθησης. Στον κάθετο άξονα υπολογίζεται ο διάμεσος χρόνος εκπαίδευσης ενώ στους οριζόντιους άξονες γίνεται ο διαχωρισμός μεταξύ μοντέλων και βιβλιοθήκης [82].

Στο Σχήμα 4.1 γίνεται σύγκριση μεταξύ των τριών βιβλιοθηκών, το Keras, το TensorFlow και το PyTorch. Αν και το Keras εμπεριέχεται μέσα στην πλατφόρμα TensorFlow, οι επιδόσεις του είναι διαφορετικές και για αυτό έχει απομονωθεί στις συγκρίσεις του. Από το γράφημα μπορεί να γίνει η παρατήρηση πως στο Keras τείνει να καθυστερεί περισσότερο η εκπαίδευσή του, ενώ στο PyTorch και στο TensorFlow τείνουν να έχουν παρόμοιες επιδόσεις.

4.2.4 Nvidia’s CUDA

Παραπάνω στο κείμενο έχει γίνει αναφορά σε μία τεχνολογία που ονομάζεται CUDA. Η τεχνολογία CUDA είναι μία πλατφόρμα της Nvidia που δίνει τη δυνατότητα του «παράλληλου προγραμματισμού» (parallel computing), επιτρέποντας σε προγραμματιστές να επιταχύνουν τις εφαρμογές τους με τη χρήση καρτών γραφικών [83]. Παρότι έχουν προταθεί και διάφορες άλλες πλατφόρμες, όπως το OpenCL, που απαιτούν κάρτες γραφικών εκτός της Nvidia, ο συνδυασμός CUDA με κάρτες γραφικών της Nvidia έχει κυριαρχήσει στον χώρο της εκπαίδευσης βαθέων νευρωνικών δικτύων. Η σπουδαιότητα της συγκεκριμένης πλατφόρμας φαίνεται από τη διαφορά στις επιδόσεις που έχει σε σύγκριση με την υπολογιστική ισχύ του επεξεργαστή, πετυχαίνοντας έως και 50 φορές καλύτερες επιδόσεις σε σχέση με τους απλούς επεξεργαστές. Η CUDA και οι κάρτες γραφικών της Nvidia χρησιμοποιούνται σε μία μεγάλη γκάμα επιστημονικών, εμπορικών, αλλά και επαγγελματικών χώρων που απαιτούν μεγάλη επεξεργαστική ισχύ. Κάποιοι από αυτούς είναι η χρηματοοικονομική, η πρόβλεψη καιρικών φαινομένων, η ανάλυση και η επιστήμη των δεδομένων, οι κατασκευαστικοί κλάδοι, τα μέσα ψυχαγωγίας, η Ιατρική, η πετρέλαιο-βιομηχανία κ.ά. [84].

Κεφάλαιο 5ο: Υλοποιήσεις Μοντέλων.

Όπως έχει προαναφερθεί στο κείμενο, υπάρχουν διάφοροι τρόποι με τους οποίους μπορεί κάποιος να έχει τα δεδομένα του βάθους ενός χώρου. Πιο συγκεκριμένα, κάποιοι από τους εξής τρόπους χρησιμοποιούν μεθόδους και αλγορίθμους Μηχανικής Μάθησης. Οι δύο τύποι μοντέλων που θα συναντήσει κανείς όσον αφορά την εκτίμηση του βάθους είναι το "Monocular Depth Estimation" και το "Stereo Depth Estimation", γνωστά και ως εκτίμηση βάθους από δεδομένα μίας κάμερας και εκτίμηση βάθους από δεδομένα δύο καμερών αντίστοιχα.

Στο συγκεκριμένο κεφάλαιο, θα παρουσιαστούν τα υλοποιημένα και δοκιμασμένα μοντέλα που χρησιμοποιήθηκαν για την εργασία. Πιο συγκεκριμένα, θα γίνει αναφορά στο ερευνητικό έργο του κάθε μοντέλου, στα υποστηριζόμενα και αποτιμημένα αποτελέσματά τους, σε κομμάτια κώδικα που υπάρχουν για την υλοποίηση του κάθε μοντέλου, αλλά και στα τεχνολογικά και τεχνικά χαρακτηριστικά τους. Επιπλέον, θα παρουσιαστούν οι εκτιμήσεις των μοντέλων, ενώ θα γίνει μία σύγκριση εικόνα προς εικόνα, σημείο προς σημείο για μια ποικιλία φωτογραφιών και των αντίστοιχων εκτιμήσεων. Τέλος, θα ασκηθεί μία υποκειμενική κριτική σχετικά με τις εκτιμήσεις βάσει των συγκρίσεων που θα γίνουν μεταξύ αυτών των μοντέλων, όπως επίσης και της γενικής εικόνας του κάθε μοντέλου.

Τα μοντέλα που επιλέχθηκαν για την υλοποίηση τους βασίστηκαν σε τρεις κύριους γνώμονες, την ταχύτητα, την προσβασιμότητα αλλά και την αποτελεσματικότητα. Πιο συγκεκριμένα η επιλογή έγινε βάσει των μετρικών τους και των αποτελεσμάτων που παρουσιάζονταν στο σύνολο δεδομένων KITTI. Επιπλέον, από τα μοντέλα που αποτιμήθηκαν στο KITTI, επιλέχθηκαν αυτά που είχαν και υλοποιημένο κώδικα. Μεταξύ των επιλογών, τα μοντέλα που επιλέχθηκαν να χρησιμοποιηθούν έπρεπε να δημιουργούν εκτιμήσεις σε χαμηλό χρόνο. Έτσι επιλέχθηκαν τα μοντέλα UnOS, HITNet και ACV-Net.

Οι φωτογραφίες που θα χρησιμοποιηθούν για να γίνει η σύγκριση των εκτιμήσεων των μοντέλων είναι από το σύνολο δεδομένων KITTI 2015 και έχουν χωριστεί σε 7 κατηγορίες. Κάθε κατηγορία αντιπροσωπεύει σκηνές με διαφορετικά χαρακτηριστικά. Ο διαχωρισμός των φωτογραφιών σε αυτές τις κατηγορίες έγινε κατόπιν παρατήρησης των κύριων χαρακτηριστικών που αντιπροσωπεύουν οι φωτογραφίες του συνόλου δεδομένων. Οι κατηγορίες χωρίζονται ως εξής:

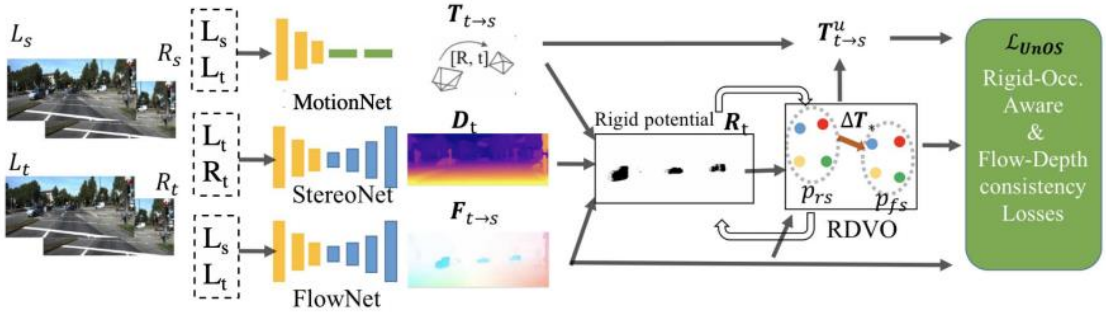
- **Αυτοτελή κτήρια:** Φωτογραφίες που παρουσιάζουν ελάχιστα στοιχεία δρόμου και περιβάλλοντος, δείχνοντας μόνο σημεία κτηρίων.
- **Δρόμος μεγάλης πυκνότητας αντικειμένων:** Φωτογραφίες που απεικονίζουν δρόμους με μεγάλο αριθμό αντικειμένων και ανθρώπων, έχοντας παράλληλα ελάχιστα σημεία που απεικονίζουν ορίζοντα.
- **Δρόμος μικρής πυκνότητας αντικειμένων:** Φωτογραφίες που απεικονίζουν δρόμους αλλά και μεγάλο κομμάτι ορίζοντα, ενώ τα αντικείμενα είναι λίγα και αραιωμένα στο περιβάλλον.
- **Δρόμους εκτός κατοικημένης:** Φωτογραφίες που απεικονίζουν δρόμους ταχείας κυκλοφορίας και δρόμους εκτός κατοικημένων περιοχών (δεν υπάρχει απαραίτητα έλλειψη πεζών). Μέσα σε αυτές είναι ορατή η έλλειψη κτηρίων στο πλαίσιο των φωτογραφιών.
- **Μεγάλη πυκνότητα πράσινου:** Η συγκεκριμένη κατηγορία περιέχει φωτογραφίες μεγάλο ποσοστό των οποίων περιέχεται από «πράσινο» (δέντρα, φυτά, λουλούδια, γρασίδι κ.α.).
- **Μη ομαλός δρόμος:** Σε αυτήν την κατηγορία εμπεριέχονται φωτογραφίες που απεικονίζουν δρόμο με ιδιαίτερα χαρακτηριστικά.
- **Υψηλά κτήρια:** Σε αυτήν την κατηγορία υπάρχουν φωτογραφίες που απεικονίζουν κτήρια μεγάλου όγκου και ύψους.

Η απόφαση στον συγκεκριμένο διαχωρισμό των φωτογραφιών έγινε μετά από παρατήρηση. Οι συγκεκριμένες κατηγορίες είναι αποτέλεσμα υποκειμενικής σκέψης και παρατήρησης. Οι συγκρίσεις μεταξύ των φωτογραφιών και των εκτιμήσεων είναι επίσης αποτελέσματα υποκειμενικής παρατήρησης.

5.1 Μοντέλο UnOS

Το μοντέλο UnOS, είναι ένα μοντέλο μηχανικής μάθησης που χρησιμοποιείται για να εκτιμήσει το βάθος και την οπτική κίνηση. Το μοντέλο UnOS χρησιμοποιεί διάφορες τεχνικές κατά τη διάρκεια της εκπαίδευσής του, όπως επίσης και τρόπους μάθησης χωρίς επίβλεψη (unsupervised learning). Πιο συγκεκριμένα, το μοντέλο εκπαιδεύεται σε διαφορετικά στάδια και κομμάτια και έχει τη δυνατότητα να κάνει εκτιμήσεις εκτός του προβλήματος του βάθους. Όπως προαναφέρθηκε, το συγκεκριμένο μοντέλο εκπαιδεύεται με σκοπό να κάνει εκτιμήσεις σχετικά με την οπτική κίνηση αλλά και για άλλα προβλήματα της όρασης υπολογιστή, όπως είναι το “ego-motion”, δηλαδή η εκτίμηση της κίνησης της κάμερας.

Το μοντέλο UnOS είναι χωρισμένο σε τέσσερα κομμάτια, συμπεριλαμβανομένων τριών μοντέλων μηχανικής μάθησης και ενός κομματιού που υπολογίζει το RDVO (Rigid-aware Direct Visual Odometry). Η Οδομετρία Εικόνας (Visual Odometry) στην όραση υπολογιστή είναι μία εργασία που χρησιμοποιείται για την εκτίμηση της θέσης, αλλά και του προσανατολισμού αντικειμένων, αναλύοντας δεδομένα φωτογραφιών. Η RDVO είναι η χρήση της Οδομετρίας Εικόνας σε πεδία και χώρους που μπορούν να θεωρηθούν άκαμπτοι [85].



Σχήμα 5.1: Αρχιτεκτονική του μοντέλου UnOS [86].

Τα υπόλοιπα τρία κομμάτια που αποτελούν το UnOS συμπεριλαμβάνουν τα τρία προαναφερθέντα υπό-μοντέλα. Τα συγκεκριμένα μοντέλα ονομάζονται MotionNet, StereoNet και FlowNet και εκπαιδεύονται για να εκτιμήσουν την κίνηση της κάμερας, το βάθος, αλλά και την οπτική κίνηση αντίστοιχα. Το Σχήμα 5.1 παρουσιάζει την αρχιτεκτονική του μοντέλου UnOS. Πιο συγκεκριμένα, το μοντέλο παίρνει σαν είσοδο δύο ζεύγη φωτογραφιών, αυτά τα ζεύγη περνάνε από τα προαναφερθέντα μοντέλα και δημιουργούν τις εκτιμήσεις T για την κίνηση της κάμερας, D για το εκτιμώμενο βάθος και F για την οπτική κίνηση. Στη συνέχεια, όσον αφορά την εκτίμηση T, αυτή περνάει μέσα από το κομμάτι που εκτελεί την εργασία RDVO που έχει προαναφερθεί, με σκοπό να βελτιώσει τα αποτελέσματα της κίνησης της κάμερας. Τέλος, όλες οι εκτιμήσεις περνάνε στο κομμάτι υπολογισμού σφάλματος και αποτίμησης του μοντέλου [86].

5.1.1 Προσεγγίσεις UnOS

Το UnOS σαν μοντέλο προτείνει διάφορες προσεγγίσεις για να μπορέσει να εκτελέσει τις εργασίες του με τα προαναφερθέντα χαρακτηριστικά. Ένα στοιχείο του UnOS που το κάνει να διαφέρει από τα

περισσότερα μοντέλα που εκτιμούν το βάθος είναι ο τύπος της εκπαίδευσής του. Όπως προαναφέρθηκε, το UnOS είναι ένα μοντέλο που χρησιμοποιεί μάθηση χωρίς επίβλεψη. Αυτό το πετυχαίνει δημιουργώντας μικρό-εργασίες εντός του μοντέλου που επιτρέπουν στο μοντέλο να αυτό επιβλέπεται. Αυτές οι εργασίες χωρίζονται σε τρία κύρια μέρη:

- Αναγνώριση αντίστοιχων pixels: Με τη χρήση των δύο ζευγών φωτογραφιών, το μοντέλο προσπαθεί να αντιστοιχίσει τα σημεία – pixels του πρώτου ζεύγους με το δεύτερο ζεύγος. Αυτά τα δύο ζεύγη αντιπροσωπεύουν σε δύο διαδοχικά καρέ μία σεκάνς βίντεο.
- Επίβλεψη με σύνθεση: Το σύστημα του μοντέλου εκπαιδεύεται δημιουργώντας συνθετικές εικόνες βάσει των δεδομένων, ελαχιστοποιώντας το σφάλμα μεταξύ των συνθετικών και των κανονικών εικόνων.
- Κανονικοποίηση με ανίχνευση ομαλότητας των ακρών: Η διαδικασία αντιστοίχισης των pixels μόνο βάσει του χρώματος τους δεν μπορεί να χρησιμοποιηθεί μόνη της. Για αυτό απαιτείται μία τεχνική κανονικοποίησης που επιτρέπει την ομαλότητα των ακρών της εικόνας για την κάθε εκτίμηση επιβάλλοντας ποινές για ασυνεπείς προβλέψεις βάσει των ακρών της εικόνας.

Μία άλλη τεχνική που προτάθηκε είναι η ένωση των δύο προβλημάτων της εκτίμησης βάθους και της οπτικής κίνησης σε μία συνδυασμένη μορφή εκπαίδευσης. Η συγκεκριμένη τεχνική, όμως, δημιουργεί πολλές προκλήσεις που μπορούν να επηρεάσουν την αποδοτικότητα της εκπαίδευσης του μοντέλου, δημιουργώντας προβλήματα για την εγκυρότητα και την αποδοτικότητα των εργασιών. Για να αποφευχθεί αυτό, μέσα στο μοντέλο υλοποιήθηκε μία σειρά τεχνικών, συμπεριλαμβανομένης μίας ειδικής «μαλακής» μάσκας για τα άκαμπτα σημεία των περιοχών της εικόνας, όπως και της προαναφερθείσας RDVO μεθοδολογίας για την κίνηση της κάμερας.

Με την ένωση των δύο προβλημάτων της οπτικής κίνησης και της εκτίμησης βάθους, το μοντέλο UnOS χρησιμοποιεί την ίδια αρχιτεκτονική νευρωνικών δικτύων και για τα δύο προβλήματα. Πιο συγκεκριμένα, η αρχιτεκτονική που χρησιμοποιείται στο MotionNet και στο StereoNet είναι βασισμένη στην αρχιτεκτονική του PWCNet. Το PWCNet είναι ένα νευρωνικό δίκτυο αποτελούμενο από συνελικτικά στρώματα, τα οποία επιλύουν σαν κύρια εργασία την οπτική κίνηση. Το PWCNet έχει 8.75 εκατομμύρια εκπαιδεύσιμες παραμέτρους, ενώ για την υλοποίηση στο UnOS διαφοροποιήθηκε έτσι ώστε να ικανοποιεί τις ανάγκες του μοντέλου [87, 86].

5.1.2 Αποτιμήσεις

Όσον αφορά την αποτίμηση του μοντέλου, οι δημιουργοί του έχουν χωρίσει τα αποτελέσματά τους σε διάφορα κομμάτια. Η αποτίμηση των εκτιμήσεων έγινε με τη βοήθεια του συνόλου δεδομένων KITTI και οι εκτιμήσεις είναι σε ανάλυση 832 * 256 pixels. Όσον αφορά τα δεδομένα εκπαίδευσης, χρησιμοποιήθηκαν τα “raw” δεδομένα από το KITTI χωρίς τη χρήση σκηνών που συμπεριλαμβάνονται στα σύνολα αποτίμησης και KITTI 2015. Οι αποτιμήσεις του μοντέλου συμπεριλαμβάνουν εκτιμήσεις σχετικά με την οπτική κίνηση, την οδομετρία εικόνας, την εκτίμηση βάθους, αλλά και την ταυτοποίηση κίνησης (motion segmentation). Οι εκτιμήσεις του βάθους απομιμούνται με τη χρήση του υποσυνόλου δεδομένων και χωρίζονται σε διάφορες κατηγορίες.

Άξιο αναφοράς όσον αφορά την εκπαίδευση του μοντέλου είναι ο διαχωρισμός της. Πιο συγκεκριμένα, το μοντέλο UnOS εκπαιδεύεται σε τρία στάδια, έτσι ώστε να μπορεί να δημιουργεί εκτιμήσεις για την πληθώρα των εργασιών στις οποίες μπορεί να χρησιμοποιηθεί. Στο πρώτο στάδιο εκπαιδεύεται το FlowNet στην οπτική κίνηση, στο δεύτερο στάδιο γίνεται η εκπαίδευση του StereoNet και του MotionNet για τις εργασίες του βάθους, αλλά και της κίνησης κάμερας αντίστοιχα. Τέλος, στο τρίτο

στάδιο, χρησιμοποιείται το κομμάτι που εκτελεί την εργασία RDVO τελειώνοντας έτσι τα στάδια της εκπαίδευσης.

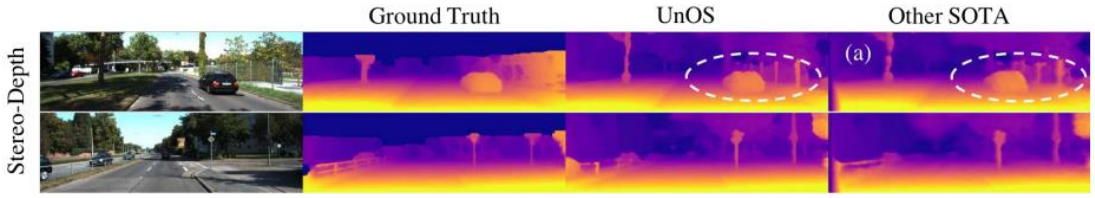
Για τις εκτιμήσεις του βάθους παρουσιάζονται αποτελέσματα με τη χρήση διαφορετικών σταδίων. Λόγω της ανεξαρτητοποίησης του μοντέλου από την αυστηρή χρήση όλων των σταδίων, παρουσιάζονται αποτελέσματα από το μοντέλο UnOS εκπαιδευμένα με διαφορετικά στάδια. Πιο συγκεκριμένα, θα παρουσιαστούν τα αποτελέσματα αποτίμησης που υποστηρίζονται από την Έρευνα για το μοντέλο UnOS που είναι εκπαιδευμένο μόνο για την εκτίμηση του βάθους, το UnOS που έχει εκπαιδευτεί στα πρώτα δύο στάδια, αλλά και το UnOS που έχει εκπαιδευτεί σε όλα τα στάδια.

Πίνακας 5.1: Σύγκριση των αποτελεσμάτων αποτίμησης του μοντέλου εκπαιδευμένο σε διαφορετικά στάδια [86].

Μέθοδος	Μικρότερο = καλύτερο					Υψηλότερο = καλύτερο		
	Abs Rel	Sq Rel	RMSE	RMSE log	D1-all	$\delta < 1.25^1$	$\delta < 1.25^2$	$\delta < 1.25^3$
UnOS (Stereo-only)	0.060	0.833	4.187	0.135	7.073%	0.955	0.981	0.99
UnOS (Ego-motion)	0.052	0.593	3.488	0.121	6.431%	0.964	**0.985**	0.992
UnOS (Full)	**0.049**	**0.515**	**3.404**	**0.121**	5.943%	**0.965**	0.984	**0.992**

Στον πίνακα 5.1 φαίνονται τα καλύτερα αποτελέσματα που έχει το μοντέλο UnOS όταν είναι εκπαιδευμένο σε διαφορετικά στάδια. Οι συγκρίσεις μεταξύ τους εμπεριέχουν 8 διαφορετικές μετρικές. Οι μετρικές Abs Rel, Sq Rel, RMSE, RMSE log και $\delta < 1.25, 1.25^2, 1.25^3$ είναι κλασσικές μετρικές για την αποτίμηση βάθους, ενώ το D1-all είναι το ποσοστό σφάλματος για τον υπολογισμό της ανισότητας. Οι μετρικές Abs Rel, Sq Rel, RMSE και RMSE log αναφέρονται στο μέσο απόλυτο σφάλμα, στο τετραγωνισμένο σφάλμα, στη ρίζα του μέσου τετραγωνισμένου σφάλματος και στη ρίζα του μέσου τετραγωνισμένου λογαριθμικού σφάλματος. Ενώ, η σύγκριση δ με τις δυνάμεις του 1.25 αναφέρονται στο κλάσμα με την αναλογία μεταξύ των αληθινών τιμών και των εκτιμώμενων τιμών μεταξύ x και 1/x. Οι πρώτες πέντε μετρικές που φαίνονται στον πίνακα θεωρούν τις μικρότερες τιμές καλύτερες, ενώ οι τελευταίες πέντε μετρικές δηλώνουν ότι όσο μεγαλύτερες είναι οι τιμές τόσο το καλύτερο [88].

Παρατηρώντας τις τιμές του πίνακα αλλά και τις μεθόδους που αντιστοιχούν, εξάγεται το συμπέρασμα ότι σε όσα περισσότερα στάδια εκπαιδεύεται το μοντέλο, τόσο καλύτερες επιδόσεις πρόκειται να έχει. Με μία μικρή εξαίρεση στη δεύτερη μετρική από το τέλος που φαίνεται ότι το μοντέλο που είναι εκπαιδευμένο στα πρώτα δύο στάδια έχει καλύτερες επιδόσεις σε όλες τις υπόλοιπες περιπτώσεις, ενώ τα μοντέλα που είναι εκπαιδευμένα σε λιγότερα στάδια έχουν χειρότερες τιμές από το πλήρως εκπαιδευμένο UnOS.



Σχήμα 5.2: Σύγκριση αποτελεσμάτων του UnOS, με τις αληθινές τιμές και αντίστοιχα SOTA μοντέλα στην εκτίμηση βάθους.

5.1.3 Τεχνικά και Τεχνολογικά χαρακτηριστικά

Στο κεφάλαιο 4, έγινε αναφορά στα τεχνικά χαρακτηριστικά των μηχανημάτων που χρησιμοποιήθηκαν για τις δοκιμές των μοντέλων, όπως επίσης και στις τεχνολογίες που χρησιμοποιήθηκαν. Πιο συγκεκριμένα, έγινε αναφορά στα δύο μηχανήματα, το τοπικό και τον server bastion. Για τις δοκιμές του UnOS μοντέλου χρησιμοποιήθηκε ο server bastion. Όσον αφορά το τεχνολογικό υπόβαθρο αλλά και τις πλατφόρμες, χρησιμοποιήθηκαν παλιές εκδόσεις των αντίστοιχων βιβλιοθηκών και πακέτων λόγω της φύσης του μοντέλου. Πιο συγκεκριμένα, οι δοκιμές του μοντέλου έγιναν σε προγράμματα γραμμένα σε γλώσσα Python έκδοσης 2.7.18. Η συγκεκριμένη έκδοση δημιουργήθηκε το 2010 και σταμάτησε να υποστηρίζεται από τις αρχές του 2020. Για τα μοντέλα μηχανικής μάθησης χρησιμοποιήθηκε η πλατφόρμα TensorFlow της έκδοσης 1.2.0, ενώ οι υπόλοιπες βιβλιοθήκες που χρησιμοποιήθηκαν παρουσιάζονται στο Σχήμα 5.3.

```
backports.functools-lru-cache==1.6.4
backports.weakref==1.0rc1
bleach==1.5.0
certifi==2020.6.20
cloudpickle==1.3.0
cycler==0.10.0
decorator==4.4.2
enum34==1.1.10
funcsigs==1.0.2
html5lib==0.9999999
kiwisolver==1.1.0
Markdown==2.2.0
matplotlib==2.2.5
mock==3.0.5
networkx==2.2
numpy==1.16.6
opencv-python==3.4.2.17
Pillow==5.4.0
protobuf==3.17.3
pyparsing==2.4.7
pypng==0.0.20
python-dateutil==2.8.2
pytz==2022.7.1
PyWavelets==1.0.3
scikit-image==0.14.5
scipy==1.2.3
six==1.16.0
subprocess32==3.5.4
tensorflow==1.2.0
Werkzeug==1.0.1
(undepthflow-old) root@92daf649644f:/workspace/UnDepthflow-thesis-2023# pip show python
```

Σχήμα 5.3: Εκδόσεις τεχνολογιών που χρησιμοποιήθηκαν για τις υλοποιήσεις του UnOS.

5.1.4 Παρουσίαση αποτελεσμάτων

Στο συγκεκριμένο κομμάτι της εργασίας θα γίνει η παρουσίαση και σχολιασμός των αποτελεσμάτων που έβγαλε το μοντέλο UnOS στην εργασία της εκτίμησης βάθους, όπως επίσης και παραδείγματα των εκτιμήσεων σε σύγκριση με τις κανονικές φωτογραφίες. Τα παραδείγματα που θα παρουσιαστούν εκπαιδεύτηκαν για μια ποικιλία αριθμού επαναλήψεων εκπαίδευσης. Συγκεκριμένα, θα παρουσιαστούν αποτελέσματα για το "Stereo-only" μοντέλο, εκπαιδευμένο για 10, 30, 50, 65 και 75 χιλιάδες εποχές, ενώ ο ρυθμός εκπαίδευσης του μοντέλου είναι 0.0001. Τα παραδείγματα θα είναι από το σύνολο δεδομένων KITTI, πιο συγκεκριμένα το KITTI 2015, και θα φροντίσουν να καλύψουν ένα ευρύ φάσμα κατηγοριών εξωτερικού χώρου. Εκτιμήσεις θα παρουσιαστούν σε διάφορες διαβαθμίσεις του μοντέλου που είναι εκπαιδευμένο μόνο για την εκτίμηση βάθους. Οι εκτιμήσεις αντιπροσωπεύουν σχετικές τιμές μεταξύ του 0 και του 255 που απεικονίζονται σε διαβαθμίσεις των χρωμάτων του μαύρου και του λευκού αντίστοιχα. Όσο πιο λευκό είναι το αντικείμενο στο πλαίσιο, τόσο πιο κοντινή είναι η απόσταση σε σχέση με την κάμερα, ενώ όσο πιο μαύρο είναι το αντικείμενο, τόσο πιο μακριά είναι. Τέλος, θα γίνει ένας συγκριτικός σχολιασμός μεταξύ των διαβαθμίσεων, αλλά και των αποτελεσμάτων καθαυτών.

Πίνακας 5.2: Σύγκριση δοκιμασμένων αποτελεσμάτων του μοντέλου UnOS.

Εποχές εκπαίδευσης	Μικρότερο = Καλύτερο					Μεγαλύτερο = Καλύτερο		
	Abs Rel	Sq Rel	RMSE	RMSE log	D1 - all	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
10,000	0.212	**2.854**	7.940	0.315	50.702%	0.726	0.861	0.931
30,000	**0.197**	4.015	**7.593**	0.287	28.013%	0.804	0.888	0.935
50,000	0.198	4.319	7.740	**0.285**	**27.030%**	**0.812**	**0.891**	**0.936**
65,000	0.205	4.589	8.002	0.288	27.254%	0.808	0.891	0.935
75,000	0.203	4.503	7.950	0.287	27.391%	0.809	0.891	0.935

Η διαδικασία αποτίμησης έγινε σε μοντέλο που εκπαιδεύτηκε μόνο για τη διαδικασία της εκπαίδευσης. Αυτή η επιλογή βασίστηκε στο υπολογιστικό και διαδικαστικό κόστος που έχει ο συγκεκριμένος αλγόριθμος που αν και βάση βιβλιογραφίας δεν θεωρείται βέλτιστος, υλοποιήθηκε με τέτοιο τρόπο ώστε να είναι εντός χρονικών ορίων. Οι αποτιμήσεις των εκπαιδευμένων μοντέλων έγιναν σε μοντέλα εκπαιδευμένα για διαφορετικό αριθμό εποχών. Τα αποτελέσματα, όπως και ο αριθμός των εποχών, φαίνονται στον Πίνακα 5.2. Όπως και στα αποτελέσματα της βιβλιογραφίας, χρησιμοποιούνται 8 μετρικές για να γίνει η σύγκριση των αποτελεσμάτων των μοντέλων. Στον Πίνακα 5.2 παρατηρούμε ότι το μοντέλο, αν και προσέγγισε, δεν κατάφερε να φτάσει τα αποτελέσματα που απεικονίζονται στον πίνακα 5.1. Ενώ, παρατηρούμε ότι μεγαλύτερος αριθμός εποχών δεν σημαίνει αυτόματα καλύτερα αποτελέσματα. Πιο συγκεκριμένα μπορεί να παρατηρηθεί ότι όταν το μοντέλο εκπαιδεύτηκε για 50,000 εποχές είχε τα συνολικά καλύτερα αποτελέσματα. Ενώ, σε μερικές περιπτώσεις, ακόμα και όταν το μοντέλο ήταν εκπαιδευμένο για λιγότερες εποχές είχε επίσης καλύτερα αποτελέσματα.

Οι μετρικές μπορούν να θέτουν τον ιδανικό αριθμό εποχών εκπαίδευσης στις 50,000, αλλά τα συμπεράσματα που θα παρουσιαστούν στη συνέχεια προέκυψαν έπειτα από υποκειμενική σύγκριση. Πιο συγκεκριμένα, θα παρουσιαστούν οι εκτιμήσεις για κάθε παρουσιασμένο μοντέλο στον Πίνακα 5.2,

με φωτογραφίες διαφορετικής φύσεως από τα δεδομένα αποτίμησης, δεδομένα που δεν εμπεριέχουν τις αληθινές τιμές μέσα στο σύνολο δεδομένων.

Σχήμα 5.4: Εκτιμήσεις βάθους από το μοντέλο UnOS (1) – Δρόμος μεγάλης πυκνότητας αντικειμένων.

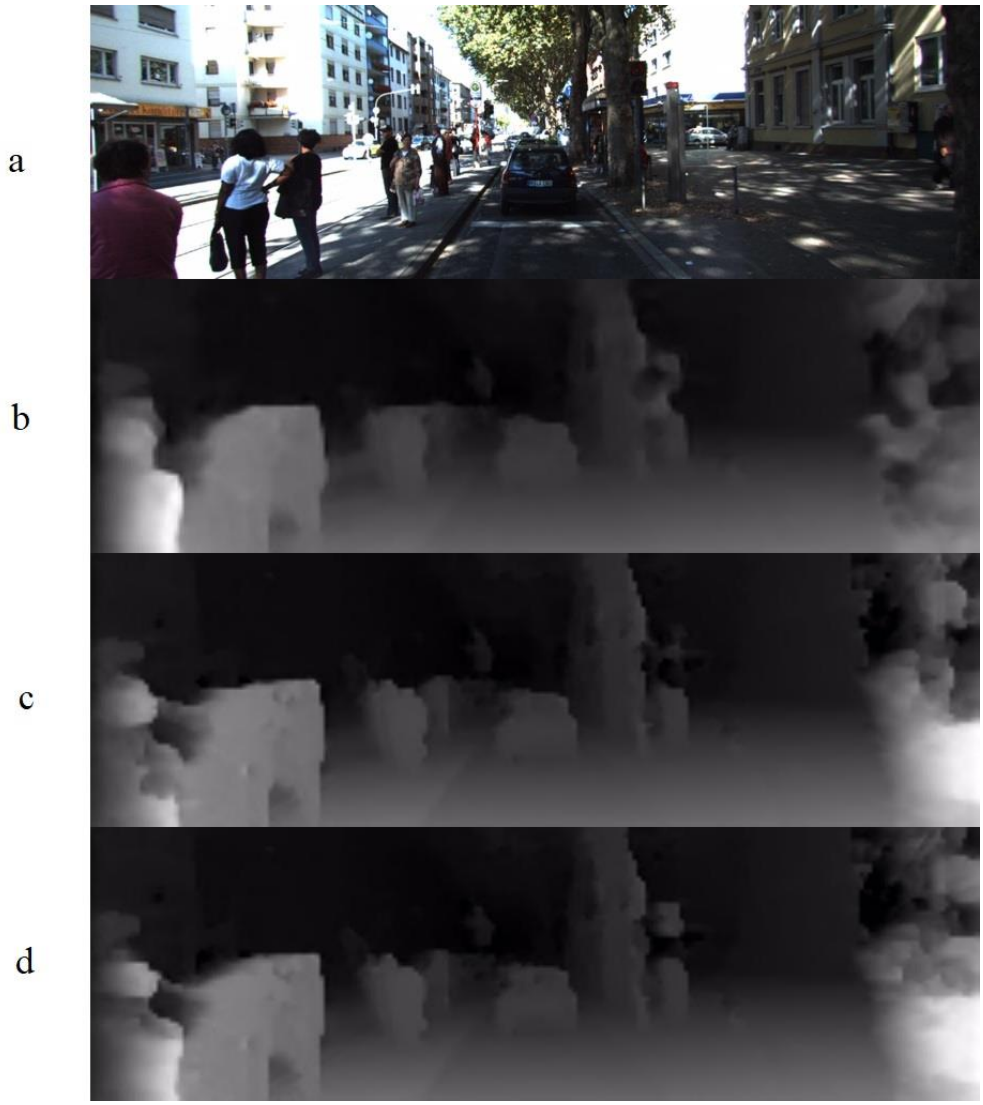
Στο Σχήμα 5.4 παρουσιάζονται η φωτογραφία και οι εκτιμήσεις βάθους από τις υλοποιημένες εκδόσεις του μοντέλου UnOS. Το [a] αντιστοιχεί στην πηγαία φωτογραφία, ενώ τα [b,c,d,e,f] αντιστοιχούν στις εκτιμήσεις των διαβαθμίσεων των μοντέλων για 10000, 30000, 50000, 65000, και 75000 αντίστοιχα. Η φωτογραφία στο Σχήμα 5.4 τοποθετείται στην κατηγορία «Φωτογραφίες με μεγάλη πυκνότητα αντικειμένων». Με μία πρώτη ματιά σε αυτές τις εκτιμήσεις είναι φυσιολογικό να θεωρήσουμε την εκτίμηση που φαίνεται στο [b] πολύ γενικευμένη ως προς τη φωτογραφία. Ενώ αναγνωρίζει τη γενική γεωμετρία της εικόνας, δυσκολεύεται στην αναγνώριση λεπτομέρειών, με αποτέλεσμα να λείπει ο διαχωρισμός των αντικειμένων.

Σχήμα 5.5: Πηγαία φωτογραφία από το Dataset KITTI 2015 [55] – Δρόμος μεγάλης πυκνότητας αντικειμένων.

Σχήμα 5.6: Εκτίμηση βάθους UnOS εκπαιδευμένο για 10,000 εποχές – Δρόμος μεγάλης πυκνότητας αντικειμένων.

Στα Σχήματα 5.5 και 5.6 παρατηρούμε την πηγαία φωτογραφία στην οποία βασίστηκε η εκτίμηση, όπως επίσης και το αποτέλεσμα που εκτίμησε το εκπαιδευμένο μοντέλο των 10,000 εποχών. Όπως προαναφέρθηκε, ενώ υπάρχει μία γενική εκτίμηση και ένας διαχωρισμός της γεωμετρικής ερμηνείας της φωτογραφίας, δημιουργείται δυσκολία στον διαχωρισμό αντικειμένων. Στο Σχήμα 5.5 παρατηρούμε αντικείμενα όπως δέντρα, στάση λεωφορείου, πινακίδες, αυτοκίνητα στο βάθος κ.ά. Στο σχήμα 5.6, με εξαίρεση τα αντικείμενα που βρίσκονται πολύ κοντά στην κάμερα (το γκρι αυτοκίνητο), όλα τα υπόλοιπα ενώνονται μαζί με το περιβάλλον και ο διαχωρισμός τους είναι ανύπαρκτος.

Αναφορικά με τη συγκεκριμένη φωτογραφία, μπορούμε να παρατηρήσουμε πως, όσο αυξάνεται ο αριθμός εποχών εκπαίδευσης για το μοντέλο, τόσο περισσότερο μπορεί να διαχωρίσει τα αντικείμενα στον χώρο και να αναγνωρίσει λεπτομέρειες. Ταυτόχρονα, σε όλες τις εκτιμήσεις φαίνεται να μην μπορεί να αναγνωριστεί εύκολα η γεωμετρική ερμηνεία των κτηρίων στο βάθος. Στο Σχήμα 5.5 παρατηρούμε ότι ο δρόμος περνάει ανάμεσα από δύο κτήρια που φαίνεται να βρίσκονται σε ίση απόσταση σε σχέση με τη θέση της κάμερας. Πέρα από την απόσταση σε σχέση με την κάμερα, τα σπίτια που απεικονίζονται έχουν διάφορα σημεία που βρίσκονται σε διαφορετική απόσταση μεταξύ τους. Το μοντέλο, όμως, σε όλες τις εκτιμήσεις του τα έχει υπολογίσει με παρόμοιο βάθος.

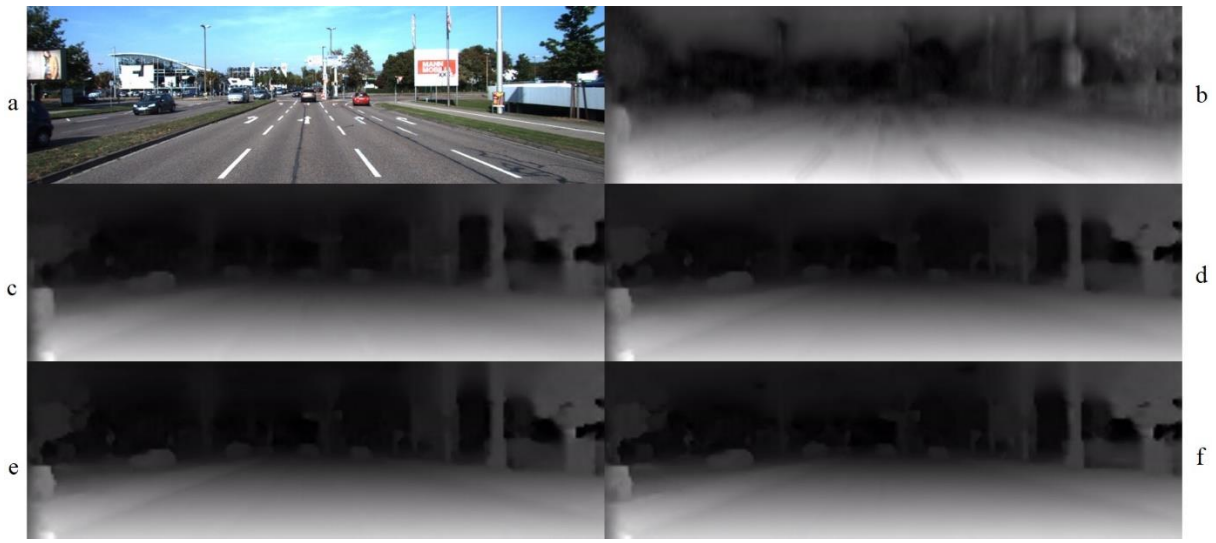


Σχήμα 5.7: Εκτιμήσεις βάθους από το μοντέλο UnOS (2) – Δρόμος μεγάλης πυκνότητας αντικειμένων. Το [a] αντιπροσωπεύει την πηγαία φωτογραφία, ενώ τα [b,c,d] αντιπροσωπεύουν τις εκτιμήσεις για το μοντέλο UnOS εκπαιδευμένο για 30, 65 και 75 χιλιάδες εποχές αντίστοιχα.

Στο Σχήμα 5.7 παρατηρούμε μία ακόμα σύγκριση εκτιμήσεων με την πηγαία φωτογραφία. Τη συγκεκριμένη φωτογραφία την έχω κατηγοριοποιήσει επίσης στην κατηγορία «Δρόμος μεγάλης πυκνότητας αντικειμένων». Οι συγκεκριμένες εκτιμήσεις πάσχουν από τα ίδια προβλήματα που είχαν και οι εκτιμήσεις στο Σχήμα 5.4, με τη διαφορά ότι έχουν προστεθεί και άλλα. Η φωτογραφία στο Σχήμα 5.7 είναι φορτωμένη με περισσότερη λεπτομέρεια σε σχέση με τη φωτογραφία στο Σχήμα 5.4, για αυτό και οι εκτιμήσεις είναι γεμισμένες με θόρυβο.

Αρχικά, σε όλες τις εκτιμήσεις, δυσκολεύεται να καταλάβει το δεξί κομμάτι της φωτογραφίας, δημιουργώντας ένα αποτέλεσμα γεμάτο θόρυβο που δεν αντιπροσωπεύει κατά προσέγγιση το πραγματικό βάθος. Στο [b] της φωτογραφίας μπορεί να παρατηρηθεί μία ταυτοποίηση, αλλά και εκτίμηση απόστασης των ανθρώπων που φαίνονται στη φωτογραφία Με τα μοντέλα με την αυξημένη εκπαίδευση, αυτός ο διαχωρισμός και οι εκτιμήσεις τείνουν να χάνονται και να δημιουργείται πρόβλημα στην πληροφορία του βάθους. Επιπρόσθετα, φαίνεται να υπάρχει ένα πρόβλημα όταν προσπαθεί να αναγνωρίσει τα λευκά κτήρια που φαίνονται στο πλαίσιο, συγχέοντάς τα με το ελάχιστο κομμάτι ορίζονται που φαίνεται σε ένα κομμάτι της φωτογραφίας. Η γενική εικόνα για τη συγκεκριμένη

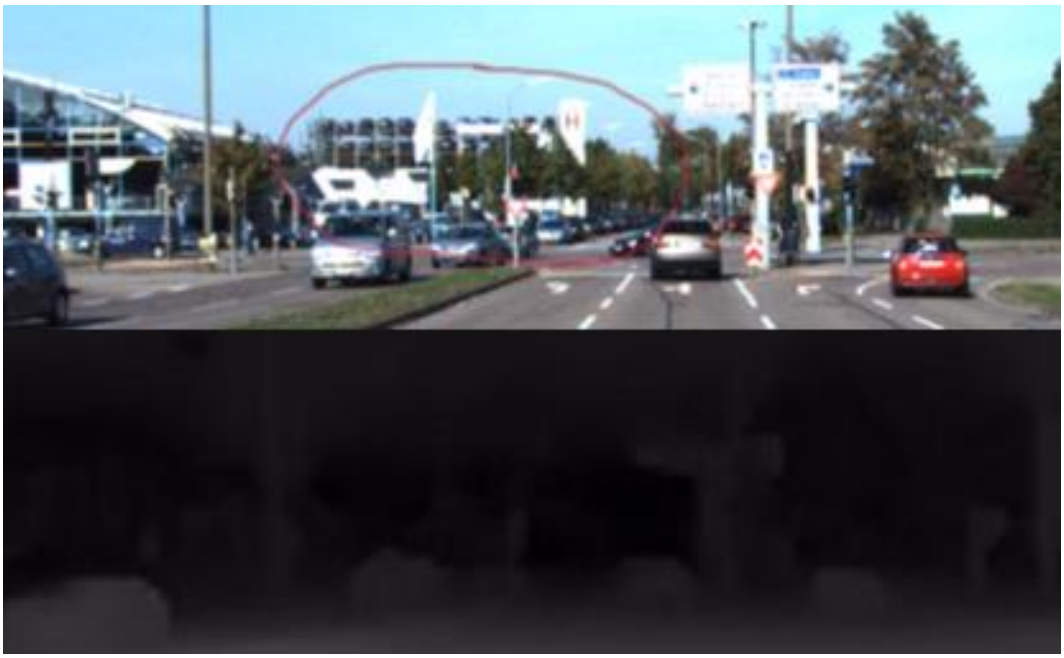
φωτογραφία μπορεί να θεωρηθεί ότι αναγνωρίζει κομμάτια της, αλλά σε καμία περίπτωση δεν μπορεί να συγκριθεί η ποιότητα των εκτιμήσεων του 5.4 με την ποιότητα των εκτιμήσεων του 5.7.



Σχήμα 5.8: Εκτιμήσεις βάθους από το μοντέλο UnOS (3) – Δρόμος μικρής πυκνότητας αντικειμένων

Στο Σχήμα 5.8 παρουσιάζονται 6 φωτογραφίες. Αντίστοιχα με τα προηγούμενα παραδείγματα, το [a] αντιπροσωπεύει την πηγαία φωτογραφία και τα [b,c,d,e,f], τον αύξοντα αριθμό εποχών για το εκπαιδευμένο μοντέλο UnOS. Σε αντίθεση με τις προηγούμενες φωτογραφίες που περιείχαν μεγάλο αριθμό αντικειμένων εντός του πλαισίου τους, η συγκεκριμένη φωτογραφία περιέχει λιγότερα αντικείμενα και φαίνεται ξεκάθαρα ο ορίζοντας του ουρανού πάνω σε αυτήν. Είναι εύκολο πλέον να συμπεράνουμε πως οι εκτιμήσεις του μοντέλου, όταν είναι εκπαιδευμένο για μικρό αριθμό εποχών, είναι πολύ γενικευμένες και δεν αντιπροσωπεύουν σε μεγάλο βαθμό την αλήθεια. Αφιερώνοντας περισσότερη προσοχή στα αποτελέσματα, παρατηρούμε ότι, όταν υπάρχει μικρός αριθμός λεπτομερειών, όσο πιο εκπαιδευμένο είναι το μοντέλο, τόσο πιο εύκολα μπορεί να διαχωρίσει αυτές τις λίγες λεπτομέρειες μέσα στο πλαίσιο. Κάθε αντικείμενο τοποθετείται σε προσεγγιστικά σωστό σχετικό βάθος βάσει της απόστασης που έχει με την κάμερα, ενώ μπορούμε εύκολα να διακρίνουμε και λεπτομέρειες, όπως αυτοκίνητα και πινακίδες στο βάθος. Δύσκολα, όμως, μπορεί να γίνει ο εντοπισμός κτηρίων και αντίστοιχα η προσεγγιστική εκτίμησή τους ως προς το βάθος.

Στο Σχήμα 5.9 γίνεται μία σύγκριση μεταξύ της πηγαίας φωτογραφίας και των εκτιμήσεων του [f] από το Σχήμα 5.8. Η προσοχή δίνεται στο σχήμα που εμφανίζεται στο βάθος. Ενώ βρίσκεται ξεκάθαρα σε μία σχετικά ενδιάμεση απόσταση μεταξύ της κάμερας και του ορίζοντα, η εκτίμηση δυσκολεύεται να κάνει τον διαχωρισμό μεταξύ αυτού και των υπόλοιπων χαρακτηριστικών της φωτογραφίας. Χάνεται ο διαχωρισμός μεταξύ του σχήματος, των αντικειμένων στη νησίδα, του ουρανού, του υπόλοιπου μέρους του δρόμου που βρίσκεται δεξιά του κτηρίου, αλλά και των αυτοκινήτων που εμπεριέχονται μέσα σε αυτό.



Σχήμα 5.9: Σύγκριση μεταξύ φωτογραφίας από το KITTI 2015 [55] (πάνω φωτογραφία) και το εκτιμώμενο βάθος από το UnOS εκπαιδευμένο για 75,000 εποχές (κάτω φωτογραφία).

Τα κυκλωμένα αποτελέσματα περιέχουν το συγκεκριμένο μοντέλο φαίνεται να δυσκολεύεται να κάνει τον διαχωρισμό μεταξύ τους, ενώ σε φωτογραφίες που έχουν ένα πιο ευρύ πλάνο και περιέχουν μικρό βαθμό λεπτομέρειας, τα αποτελέσματα τείνουν να προσεγγίζουν καλύτερα την πραγματικότητα.

Σχήμα 5.10: Εκτιμήσεις βάθους από το μοντέλο UnOS (4) – Αυτοτελή Κτήρια

Σχήμα 5.11: Εκτιμήσεις βάθους από το μοντέλο UnOS (5) – Υψηλά Κτήρια

Όσο μειώνεται ο ανοιχτός χώρος και ο ορίζοντας που απεικονίζεται στις φωτογραφίες, τόσο πιο ανακριβείς γίνονται και οι εκτιμήσεις. Στα Σχήματα 5.10 και 5.11 παρατηρούμε δύο εκτιμήσεις βάσει των πηγαίων φωτογραφιών που απεικονίζουν κατά κύριο λόγο αυτοτελή και υψηλά κτήρια. Πιο συγκεκριμένα, στο Σχήμα 5.10 βλέπουμε τη σύγκριση μεταξύ της πηγαίας φωτογραφίας (πάνω) και της αντίστοιχης εκτίμησης στο εκπαιδευμένο για 75 χιλιάδες εποχές μοντέλο. Ενώ, στο 5.11 παρατηρούμε σύγκριση μεταξύ πηγαίας φωτογραφίας [a] και εκτιμήσεων για 50000 εποχές [b] και 75000 εποχές [c] εκπαίδευσης για το UnOS. Και στις δύο περιπτώσεις παρατηρούμε μία περίεργη συμπεριφορά σχετικά με τα αυτοκίνητα που φαίνονται στο κτήριο. Στο Σχήμα 5.11 αυτό φαίνεται πολύ πιο έντονα. Αυτή η συμπεριφορά φαίνεται να αυξάνεται, όσο αυξάνεται και ο αριθμός εποχών εκπαίδευσης.

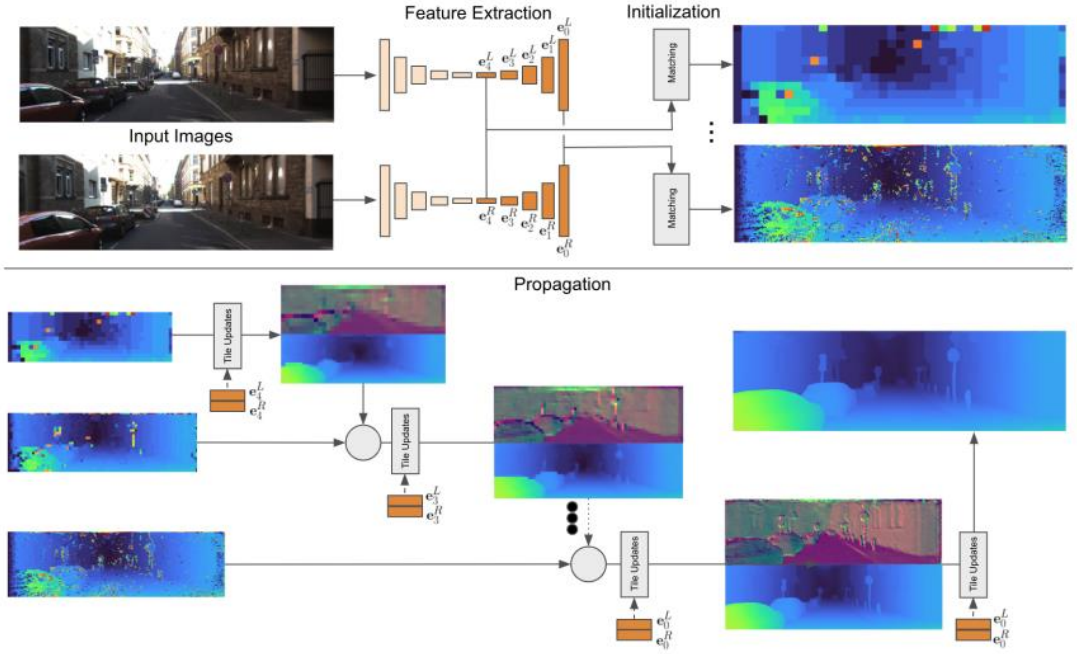
Σχήμα 5.12: Εκτίμηση βάθους από το μοντέλο UnOS – 30000 εποχές εκπαίδευσης.

Στο Σχήμα 5.12 παρουσιάζεται η εκτίμηση της πηγαίας φωτογραφίας από το Σχήμα 5.11. Μπορεί να παρατηρηθεί η προσέγγιση στη λεπτομέρεια του αυτοκινήτου στη δεξιά πλευρά του Σχήματος 5.12. Κάτι επιπλέον που μπορεί να παρατηρηθεί στο Σχήμα 5.11 είναι η έλλειψη διαχωρισμού μεταξύ του χώρου των ανθρώπων. Πιο συγκεκριμένα, και στο [b] και στο [c] κομμάτι του σχήματος, μπορεί να παρατηρηθεί μία ενοποιημένη επιφάνεια με παρόμοιο εκτιμώμενο βάθος των ανθρώπων δίπλα της. Παρατηρώντας την πηγαία φωτογραφία, μπορούμε να καταλάβουμε πως συγκεκριμένα στοιχεία στο πλαίσιο της φωτογραφίας βρίσκονται πολύ πιο μακριά απ' ότι απεικονίζονται.

5.2 Μοντέλο HITNet

Το HITNet [89] είναι ένα μοντέλο μηχανικής μάθησης που σαν κύριο στόχο έχει να δημιουργεί εκτιμήσεις βάθους. Η έρευνα του HITNet δημιουργήθηκε με γνώμονα τη μείωση του υπολογιστικού κόστους που είχαν άλλα μοντέλα που χρησιμοποιούν ζεύγη φωτογραφιών. Η βασική αρχή της προσέγγισης του μοντέλου είναι να δημιουργήσει δεδομένα και πληροφορίες με τη χρήση διαφόρων τεχνικών διάδοσης, up-sampling [90] και παραμόρφωσης εικόνας. Τα κύρια χαρακτηριστικά του HITNet είναι τα εξής:

- Γρήγορο και προσαρμόσιμο σε μεγάλες αναλύσεις.
- Αποτελεσματική δισδιάστατη διάδοση ανισοτήτων.
- SOTA αποτελέσματα σε δείκτες αναφοράς αποτίμησης του μοντέλου συγκριτικά με άλλα μοντέλα.



Σχήμα 5.13: Διάγραμμα του δικτύου HITNet [89].

Ο σχεδιασμός του μοντέλου δεν διαφοροποιείται πολύ σε σχέση με τους κλασσικούς αλγόριθμους ταυτοποίησης ενός ζεύγους φωτογραφιών. Οι πιο αποτελεσματικοί αλγόριθμοι βασίζουν την μεθοδολογία τους σε τρία κύρια χαρακτηριστικά, στη συμπαγή εξαγωγή των αναπαριστώμενων χαρακτηριστικών, στην αρχικοποίηση της ανισότητας του ζεύγους των φωτογραφιών βάσει των εξαγόμενων χαρακτηριστικών και σε ένα αποτελεσματικό σύστημα διάδοσης και εκπαίδευσης του μοντέλου για τη βελτίωση των εκτιμήσεων. Η γενική μέθοδος, όπως παρουσιάζεται και στο διάγραμμα του Σχήματος 5.13, βασίζεται σε ένα πολύ μικρό δίκτυο τύπου «U-Net» [91], όπου τα εξαγόμενα χαρακτηριστικά του αποκωδικοποιητή χρησιμοποιούνται από τις υπόλοιπες διαδικασίες. Στη συνέχεια,

αρχικοποιούνται οι χάρτες ανισότητας των φωτογραφιών και βάσει αυτών των χαρτών θεωρούνται διάφοροι άλλοι υποθετικοί χάρτες ανισότητας, και επιλέγεται ο βέλτιστος. Στο τελευταίο στάδιο, ξεκινά η διαδικασία της διάδοσης της πληροφορίας, βελτιώνοντας την υπόθεση με επαναληπτικό τρόπο [89].

Στο Σχήμα 5.13 παρουσιάζεται η προαναφερθείσα διαδικασία. Το πάνω κομμάτι του σχήματος εξάγει τα χαρακτηριστικά και δημιουργεί τους υποθετικούς χάρτες ανισότητας βάσει της μάσκας που χρησιμοποιείται. Στο κάτω σημείο του σχήματος παρουσιάζεται η επαναληπτική φύση του σταδίου της διάδοσης. Σε κάθε επανάληψη επιλέγεται ο εκτιμώμενος χάρτης ανισότητας με το μικρότερο κόστος – σφάλμα, μέχρι να βγει ο τελικός χάρτης ανισότητας.

Η αρχιτεκτονική του μοντέλου HITNet, αποτελείται από έναν συνδυασμό πολλών στρωμάτων. Εντός της αρχιτεκτονικής εμπεριέχονται συνελικτικά στρώματα διαφόρων μεγεθών, στρώματα τύπου MLP, η συνάρτηση ενεργοποίησης ReLU και ο αριθμός των παραμέτρων του HITNet μοντέλου που ανέρχεται στις 450,000.

5.2.1 Διαδικασία Εκπαίδευσης

Το μοντέλο HITNet χρησιμοποίησε μία πληθώρα από σύνολα δεδομένων κατά τη διάρκεια της εκπαίδευσής του. Τα σύνολα που χρησιμοποίησε είναι το SceneFlow [58], το KITTI [13], το ETH3D [59] και το Middlebury [60].

Για την εκπαίδευση του μοντέλου στο dataset SceneFlow, χρησιμοποιήθηκε μόνο το υποσύνολο flying things, λόγω της έλλειψης δεδομένων αποτίμησης. Εκπαιδεύτηκε συνολικά σε 35,000 φωτογραφίες για πάνω από 1,4 εκατομμύριο επαναλήψεις με τη χρήση του Adam Optimizer [92] με ρυθμό εκπαίδευσης 0.0004 μειώνοντάς το σε 0.0001, 0.00004 και τέλος σε 0.00001 μετά από 1 εκατομμύριο, 1,3 εκατομμύριο και 1,4 εκατομμύριο επαναλήψεις.

Για την εκπαίδευση του μοντέλου σε δεδομένα πραγματικού κόσμου, χρησιμοποιήθηκε το KITTI, το οποίο και εκπαιδεύτηκε για το 75% των δεδομένων, ενώ το υπόλοιπο 25% χρησιμοποιήθηκε για την αποτίμησή του. Εκπαιδευμένο με τη χρήση τεχνικών data augmentation και με τυχαίως αναπροσαρμοσμένη ανάλυση σε 311*1178 pixels, το μοντέλο εκπαιδεύτηκε για 400,000 επαναλήψεις με ρυθμό εκπαίδευσης 0.0004, ακολουθούμενο από 8,000 και 2,000 επαναλήψεις με αλλαγμένο τον ρυθμό εκπαίδευσης σε 0.0001 και 0.00004 αντίστοιχα.

Για την αποτίμηση στο dataset ETH3D, χρησιμοποιήθηκε ένας συνδυασμός από σύνολα δεδομένων για να επιτευχθεί καλή εκπαίδευση και να αποφευχθεί το “over-fitting”. Λόγω του μικρού όγκου δεδομένων που περιέχει το ETH3D, χρησιμοποιήθηκαν δεδομένα και από το dataset KITTI όπως και ¾ από το σύνολο Middlebury V3. Εκπαιδεύτηκε με παρόμοιες παραμέτρους που χρησιμοποιήθηκαν για 115,000 επαναλήψεις. Αντίστοιχα, στο σύνολο δεδομένων Middlebury, λόγω του μικρού όγκου δεδομένων εκπαίδευσής του, εκπαιδεύτηκε πάνω στην εκπαίδευση που έγινε για το σύνολο SceneFlow και στη συνέχεια βελτιστοποιήθηκε πάνω στα δεδομένα του Middlebury προ-εκπαιδευμένο για 445,000 επαναλήψεις και με τυχαία αναπροσαρμογή ανάλυσης στα 512*960 pixels, με ρυθμό εκπαίδευσης 0.0004 και τη σταδιακή του μείωση σε 0.0001, 0.00004 και 0.00001 μετά από 300,000, 400,000 και 435,000 επαναλήψεις αντίστοιχα. Τέλος, έγινε ένα τελευταίο κομμάτι εκπαίδευσης για 5,000 επαναλήψεις με ρυθμό εκπαίδευσης 0.00001.

5.2.2 Τεχνικά και Τεχνολογικά χαρακτηριστικά

Το μοντέλο HITNet μοιράζεται πολλά τεχνικά χαρακτηριστικά συγκριτικά με το UnOS. Όπως και το προηγούμενο μοντέλο, ο υλοποιημένος κώδικας είναι γραμμένος σε γλώσσα Python και για την

εκπαίδευση, διαχείριση και αποτίμηση του μοντέλου χρησιμοποιεί την πλατφόρμα TensorFlow. Όμως, σε αντίθεση με το UnOS, το HITNet είναι ένα πιο σύγχρονο μοντέλο και χρησιμοποιεί νεότερες εκδόσεις των αντίστοιχων τεχνολογιών που χρησιμοποιεί το UnOS. Η υλοποίηση του μοντέλου είναι γραμμένη σε Python 3, ενώ συγκεκριμένα για τις αποτιμήσεις χρησιμοποιήθηκε η έκδοση της Python 3.10.6 που κυκλοφόρησε στις 8 Αυγούστου του 2022.

Αντίστοιχα, οι βιβλιοθήκες, οι πλατφόρμες και τα σύνολα που χρησιμοποιήθηκαν κατά την υλοποίηση, είναι νεότερων εκδόσεων συγκριτικά με το προηγούμενο μοντέλο. Το TensorFlow χρησιμοποιείται στην έκδοση 2.11.0, το NumPy 1.24 και το OpenCV είναι στην έκδοση 4.7.0.72. Οι περεταίρω τεχνολογίες που χρησιμοποιήθηκαν για το μοντέλο παρουσιάζονται στο Σχήμα 5.14.

PyQt5-sip==12.12.1
pyrsistent==0.18.1
python-apt=2.4.0+ubuntu1
python-dateutil==2.8.2
python-debian==0.1.43+ubuntu1.1
python-dotenv=-0.19.2
Markdown=-3.4.1
python-version==0.0.2
pytz=-2023.3
pyxattr=-0.7.2
PYYAML==5.4.1
reportlab -3.6.8
requests-oauthlib==1.3.1
screen-resolution-extra==0.0.0
requests==2.25.1
rsa==4.9
SecretStorage==3.3.1
six==1.16.0
systemd-python==234
tensorboard==2.11.2
libclang==15.0.6.1
MarkupSafe==2.1.2
matplotlib==3.7.2
more-itertools==8.10.0
netifaces==0.11.0
numpy=-1.24.2
oauthlib=-3.2.0
olefile==0.46
opencv-python=-4.7.0.72
openpyxl=-3.1.2
opt-einsum =3.3.0
packaging==23.0
pafy=-0.5.5
gast==0.4.0
google-auth==2.16.2
google-auth-oauthlib=-0.4.6
google-pasta-=0.2.0
gpg== 1.20.0-unknown
grpcio--1.51.3
gyp==0.1
h5py==3.8.0
httplib2--0.20.2
idna -3.3
importlib-metadata=-4.6.4
imread-from-url==0.1.3
jeepney=-0.7.1
jsonschema -3.2.0
keras 2.11.0
Keras-Preprocessing==1.1.2
keyring==23.5.0
kiwisolver=-1.4.4
language-selector=-0.1
launchpadlib==1.10.16
lazr.restfulclient=-0.14.4
lazr.uri--1.0.6
tensorboard-data-server==0.6.1
tensorboard-plugin-wit==1.8.1
tensorflow==2.11.0
tensorflow-estimator ==2.11.0
tensorflow-io-gcs-filesystem==0.31.0
termcolor==2.2.0
texttable==1.6.4
tornado==6.1
typing_extensions==4.5.0
tzdata==2023.3
ubuntu-drivers-common==0.0.0
ufw==0.36.1
urllib3=-1.26.5
wadllib==1.3.6
websocket-client=-1.2.3
Werkzeug==2.2.3
wrapt -1.15.0
xkit=-0.0.0
youtube-dl=-2021.12.17
zipp==1.0.0

Σχήμα 5.14: Λίστα με τεχνολογίες και πλατφόρμες που χρησιμοποιήθηκαν για το μοντέλο HITNet.

5.2.3 Αποτιμήσεις και Εκτιμήσεις

Στο συγκεκριμένο κομμάτι της εργασίας θα παρουσιαστούν τα αποτελέσματα των αποτιμήσεων που προέκυψαν από την Έρευνα. Επιπρόσθετα, θα παρουσιαστούν πηγαίες φωτογραφίες και οι εκτιμήσεις τους για το σύνολο δεδομένων KITTI 2015 [55] με τη χρήση του μοντέλου HITNet. Οι φωτογραφίες είναι κομμάτι των φωτογραφιών αποτίμησης του KITTI 2015 και τα αποτελέσματα των εκτιμήσεων θα έχουν υποκειμενική φύση. Οι φωτογραφίες που θα επιλεχθούν θα είναι είτε ίδιες, είτε παρόμοιες με αυτές που παρουσιάστηκαν για το μοντέλο UnOS. Στον Πίνακα 5.3, παρατηρούμε τα αποτελέσματα μεταξύ του HITNet και διάφορες διαβαθμίσεις του μοντέλου UnOS. Συγκεκριμένα, παρουσιάζονται 3

βιβλιογραφικά αποτελέσματα αποτίμησης και τα αποτελέσματα που είχε το μοντέλο κατά την αποτίμηση της υλοποίησης των 50,000 εποχών.

Πίνακας 5.3: Αποτίμηση και σύγκριση αποτελεσμάτων στο KITTI 2015 μεταξύ HITNet και UnOS (Μικρότερο = καλύτερο).

Μέθοδος	KITTI 2015		
	D1-bg	D1-fg	D1-all
HITNet	1.740%	3.200%	1.980%
UnOS (Full)	-	-	5.943%
UnOS (Stereo Only)	-	-	7.073%
UnOS (Υλοποιήσιμο) 50,000 Epochs	-	-	27.030%

Παρατηρώντας τις συγκρίσιμες μετρικές μεταξύ του HITNet και του UnOS, μπορούμε να παρατηρήσουμε ξεκάθαρη βελτίωση στο πρόβλημα της εκτίμησης βάθους. Η καλύτερη εκδοχή του UnOS πετυχαίνει σχεδόν 6% στην μετρική D1-all, ενώ το μοντέλο HITNet επιτυγχάνει ποσοστό μικρότερο του 2%. Μέσα στον πίνακα 5.3, εμπεριέχονται και άλλες μετρικές, όπως είναι η D1-bg και η D1-fg, οι οποίες σημαίνουν το ποσοστό σφάλματος των pixels, με σφάλμα στην ανισότητα άνω του ενός pixel για φωτογραφίες με περιοχές σε υπόβαθρο και περιοχές σε μπροστινό πλάνο αντίστοιχα.

Σχήμα 5.15: Πηγαία φωτογραφία από το KITTI 2015 (πάνω), και η αντίστοιχη εκτίμηση του HITNet (κάτω).

Παρατηρώντας το Σχήμα 5.15, μπορούμε να καταλάβουμε ξεκάθαρα την εκτίμηση που έχει δημιουργήσει το μοντέλο αναφορικά με το βάθος. Οι διαβαθμίσεις των χρωμάτων της φωτογραφίας δίνουν μια ξεκάθαρη εικόνα όσον αφορά τα δεδομένα του βάθους. Ο διαχωρισμός μεταξύ των λεπτομερειών είναι ξεκάθαρος, καθώς εντοπίζει και επιπλέον λεπτομέρειες που δεν φαίνονται στο Σχήμα 5.4. Όμως, όπως και στο προηγούμενο μοντέλο, το HITNet, αδυνατεί να κάνει τον διαχωρισμό στο βάθος σε αντικείμενα στο πλαίσιο της φωτογραφίας, όπως είναι τα σπίτια. Το παράδειγμα στο

Σχήμα 5.15 δείχνει επίσης παρόμοιες διαβαθμίσεις σε όλο το μήκος του σπιτιού από την αριστερή πλευρά, με το μόνο σημείο που διαβαθμίζεται στο χρώμα προς το τέλος του σπιτιού.

Σχήμα 5.16: Πηγαία φωτογραφία από το KITTI 2015 (πάνω) και η αντίστοιχη εκτίμηση του HITNet (κάτω) με σημειωμένες κύριες λεπτομέρειες.

Αξιοσημείωτο σε αυτό το παράδειγμα είναι να κοιτάξουμε συγκεκριμένες λεπτομέρειες από την εκτίμηση. Στο Σχήμα 5.16, έχουν σημειωθεί οι πιο σημαντικές λεπτομέρειες και οι αντίστοιχες εκτιμήσεις με κόκκινο πλαίσιο. Ξεκινώντας από αριστερά προς τα δεξιά, παρατηρούμε τον τρόπο με τον οποίο έχει εκτιμήσει τις λεπτομέρειες στο αριστερό σημείο της φωτογραφίας που έχει διάφορα αντικείμενα περίπλοκου σχήματος. Στην εκτίμηση μπορεί να παρατηρηθεί μία διαβάθμιση και ένας διαχωρισμός των συγκεκριμένων αντικειμένων, παρουσιάζοντας ξεκάθαρα τη διαφορά τους στο βάθος ως προς την κάμερα. Αντίστοιχα, στη μέση της φωτογραφίας, δεξιά του δρόμου, παρουσιάζεται μία σειρά από σπίτια το ένα πίσω από το άλλο. Η αντίστοιχη εκτίμηση διαβαθμίζει αυτά τα σπίτια με διαφορετικό χρώμα, καθώς αντιπροσωπεύουν και σημεία στη φωτογραφία σε διαφορετική απόσταση από την κάμερα. Τέλος, στα δεξιά της φωτογραφίας, παρατηρούμε ένα σημείο που είναι δύσκολα επεξεργάσιμο με το μάτι, όμως τα αποτελέσματα των εκτιμήσεων μπορούν και δείχνουν πέρα από τον εντοπισμό των αντικειμένων, όπως τα δέντρα και το στέγαστρο, αλλά και διαβαθμίσεις στο βάθος του στεγάστρου, που θα μπορούσαν να αντιπροσωπεύουν τις αληθινές τιμές.

Σχήμα 5.17: Πηγαία φωτογραφία από το Dataset KITTI 2015 (πάνω) [55] και η αντίστοιχη εκτίμηση του HITNet (κάτω) – Δρόμος εκτός κατοικημένης.

Στο Σχήμα 5.17 παρουσιάζεται μια φωτογραφία διαφορετικής κατηγορίας από ό,τι έχει ήδη παρουσιαστεί. Η πηγαία φωτογραφία που απεικονίζεται στο πάνω μέρος του Σχήματος είναι από την κατηγορία «Δρόμος εκτός κατοικημένης». Πιο συγκεκριμένα, η φωτογραφία απεικονίζει στιγμιότυπο από έναν δρόμο ταχείας κυκλοφορίας, στην είσοδο μίας υπόγειας διάβασης κάτω από γέφυρα. Παρατηρώντας την εκτίμηση, μπορούμε να καταλάβουμε την κεντρική ιδέα του βάθους του χώρου, το περίεργο όμως παρατηρείται στο αριστερό κομμάτι της φωτογραφίας. Εκεί το μοντέλο αδυνατεί να κατανοήσει τη γεωμετρία του χώρου και στην εκτίμηση παράγει θόρυβο. Επίσης, φαίνεται ότι το μοντέλο μπερδεύεται στο σημείο που πρέπει να διαχωρίσει τον ορίζοντα με το κομμάτι του τοίχου του δρόμου.

Σχήμα 5.18: Πηγαία φωτογραφία από το Dataset KITTI 2015 [55] (πάνω) και η αντίστοιχη εκτίμηση του HITNet (κάτω) – Δρόμος μικρής πυκνότητας αντικειμένων.

Αντίθετα με τη φωτογραφία και την εκτίμηση στο Σχήμα 5.15, παρατηρούμε κάτι ιδιαίτερο στο Σχήμα 5.18. Η πηγαία φωτογραφία στο συγκεκριμένο σχήμα είναι η ίδια φωτογραφία που χρησιμοποιήθηκε και στο Σχήμα 5.8. Ενώ στο Σχήμα 5.8 γινόταν ξεκάθαρος διαχωρισμός των αντικειμένων του χώρου, στο Σχήμα 5.18 παρατηρούμε ότι είναι δύσκολο να παρατηρηθεί με γυμνό μάτι. Φυσικά, αυτή η παρατήρηση δεν καθιστά την εκτίμηση λανθασμένη ή μη αντιπροσωπευτική της αλήθειας. Με επίμονη παρατήρηση, μπορούν να βρεθούν εντός του εκτιμώμενου βάθους, αντικείμενα, όπως αυτοκίνητα και πινακίδες. Φυσικά, το πρόβλημα που τίθεται να λύσει το συγκεκριμένο μοντέλο δεν είναι η ταυτοποίηση των αντικειμένων στον χώρο, αλλά η εκτίμηση του βάθους. Σε μία φωτογραφία που δεν είναι πυκνωμένη με αντικείμενα εντός του πλαισίου της και όταν οι εκτιμήσεις βγαίνουν σε σχετικές τιμές ως προς την κάμερα, τότε δεν είναι παράλογο να υπάρχει χαμηλή διαβάθμιση των χρωμάτων και των τιμών, εφόσον αυτό που παρουσιάζει η εικόνα είναι ένας ανοιχτός χώρος με παρόμοιο σχετικό βάθος ως προς την κάμερα.

Σχήμα 5.19: Πηγαία φωτογραφία από το Dataset KITTI 2015 [55] (πάνω) και η αντίστοιχη εκτίμηση του HITNet (κάτω) – Μη ομαλός δρόμος.

Στο Σχήμα 5.19 γίνεται η σύγκριση της πηγαίας φωτογραφίας (πάνω) με την αντίστοιχη εκτίμηση που έβγαλε το μοντέλο HITNet. Η φωτογραφία έχει μπει στην κατηγορία «Μη ομαλός δρόμος». Ο λόγος που έχει τοποθετηθεί σε αυτήν την κατηγορία είναι η ύπαρξη των γραμμών τρένου που υπάρχουν στο αριστερό κομμάτι της, όπως και η ύπαρξη της γέφυρας στο πάνω κομμάτι της φωτογραφίας.

Παρατηρώντας την εκτίμηση του βάθους αλλά και τη φωτογραφία στο Σχήμα 5.19, μπορούμε να κάνουμε τις εξής παρατηρήσεις. Αρχικά, υπάρχει μία ξεκάθαρη διαφορά στο απόλυτο ύψος που βρίσκονται οι ράγες συγκριτικά με το υπόλοιπο οδόστρωμα, ως προς την επιφάνεια της γης. Αν και είναι μικρή αυτή η διαφορά του ύψους, είναι αρκετή για να υπάρξει διαφοροποίηση των σημείων του οδοστρώματος, αλλά και των ραγών ως προς το σημείο της κάμερας. Αυτή η διαφορά στη σχετική απόσταση ως προς την κάμερα μετατρέπεται σε διαφορετικό βάθος στην εκτίμηση. Το μοντέλο HITNet πέρα από το γεγονός ότι καταφέρνει να το αναγνωρίσει ως κάτι διαφορετικό από το υπόλοιπο οδόστρωμα, μπορεί και εντοπίζει τη διαφορά της απόστασης που έχει από την κάμερα.

Αντίστοιχα, το μοντέλο μπορεί και αναγνωρίζει ως προς την κάμερα τη σχετική απόσταση που έχει με τη γέφυρα, το δεξιό τείχος του δρόμου, αλλά και τον βράχο στα αριστερά. Πέρα από τις διαφοροποιήσεις στο βάθος που έχουν μεταξύ τους, αλλά και το φάσμα των διαφορετικών σχετικών τιμών που καλύπτει κάθε ένα από αυτά τα σημεία, γίνεται και ο διαχωρισμός τους σε σχέση με τον ορίζονται και τα κτήρια που φαίνονται στη μέση.

Σχήμα 5.20: Πηγαία φωτογραφία από το Dataset KITTI 2015 [55] (πάνω) και η αντίστοιχη εκτίμηση του HITNet (κάτω) – Υψηλά κτήρια.

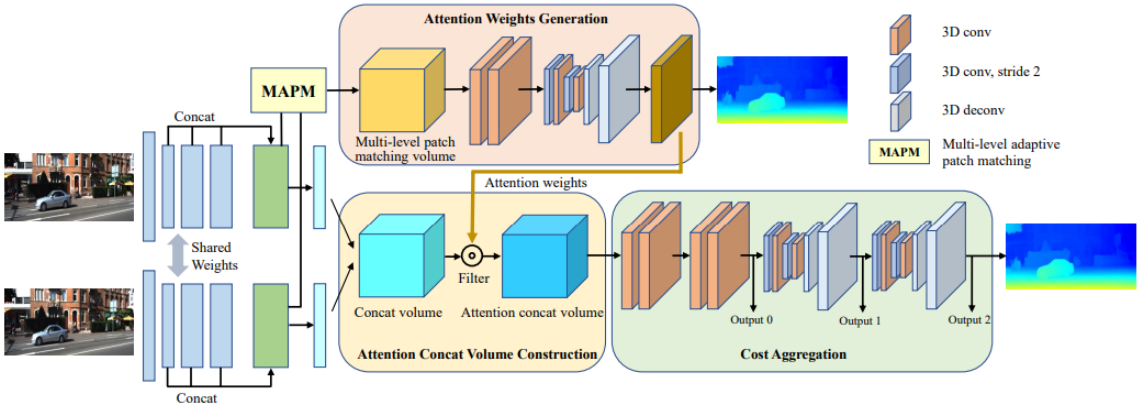
Στο σχήμα 5.20 γίνεται σύγκριση μεταξύ της πηγαίας φωτογραφίας και του εκτιμώμενου βάθους που έβγαλε το μοντέλο HITNet. Η φωτογραφία ανήκει στην κατηγορία «Υψηλά κτήρια». Οι τιμές που αντιπροσωπεύουν το βάθος από το HITNet είναι πάντα σε σχετικές τιμές. Αυτές οι τιμές αντιπροσωπεύουν ένα χρώμα: όσο πιο λευκό είναι το χρώμα, τόσο πιο κοντά στην κάμερα βρίσκεται, ενώ όσο πιο μαύρο είναι το χρώμα στη φωτογραφία, τόσο πιο μακριά βρίσκεται. Συγκριτικά, όμως, με της προηγούμενες εκτιμήσεις, στο παράδειγμα του Σχήματος 5.20 παρατηρείται ένα παράδοξο.

Στις περισσότερες εκτιμήσεις, το μέρος του δρόμου που βρίσκεται μπροστά από την κάμερα, παρουσιάζεται με λευκό χρώμα. Αντίθετα, στο παράδειγμα στο Σχήμα 5.20, το μέρος του δρόμου που βρίσκεται πιο κοντά στην κάμερα είναι πολύ πιο κοντά σε σκούρο γκρι παρά σε λευκό. Αυτό δημιουργείται λόγω των αντικειμένων που βρίσκονται στο αριστερό σημείο της φωτογραφίας. Τα συγκεκριμένα αντικείμενα επειδή βρίσκονται συγκριτικά πολύ πιο κοντά στην κάμερα έναντι του μπροστινού μέρους του δρόμου, παίρνουν μικρότερες σχετικές τιμές της απόστασης και, άρα, παρουσιάζονται πιο κοντά στο λευκό. Αυτό έχει ως αποτέλεσμα να δημιουργεί σχετικά ασυνεπείς προβλέψεις χωρίς όμως να σημαίνει ότι η κάθε μία πρόβλεψη ξεχωριστά δεν αντιπροσωπεύει κομμάτι των αληθινών τιμών.

5.3 Μοντέλο ACVNet & Fast-ACVNet

Το μοντέλο ACVNet είναι, επίσης, ένα μοντέλο μηχανικής μάθησης και η κύρια εργασία που επιλύει είναι να παράγει εκτιμήσεις βάθους με τη χρήση “Stereo Depth Estimation” μεθόδων. Η βιβλιογραφία που βασίζεται το ACVNet αναγνωρίζει τέσσερα βασικά βήματα στις αρχιτεκτονικές των υπόλοιπων SOTA μοντέλων: την εξαγωγή χαρακτηριστικών (feature extraction), τη δημιουργία και συνάθροιση

του κόστους, αλλά και την οπισθοδρόμηση εκπαίδευσης της ανισότητας. Βασισμένο στην αρχιτεκτονική του ήδη προ-υπάρχοντος μοντέλου GwcNet [93], το ACVNet προτείνεται με κύριο γνώμονα τη χρήση ενός ACV κομματιού, που εκμεταλλεύεται πληροφορίες σχετικά με την ένταση του συσχετισμού των δεδομένων από τα ζευγάρια των φωτογραφιών και την παραγωγή βαρών που θα λειτουργούν σαν φίλτρο αναφορικά με τη συνένωσή τους. Για να υπάρχουν ακριβή αποτελέσματα όσον αφορά το κομμάτι της συσχέτισης, προτείνεται μία πολύ-επίπεδη μέθοδος που θα ταυτίζει αυτά τα δεδομένα με τη χρήση μασκών διαφόρων μεγεθών [94].



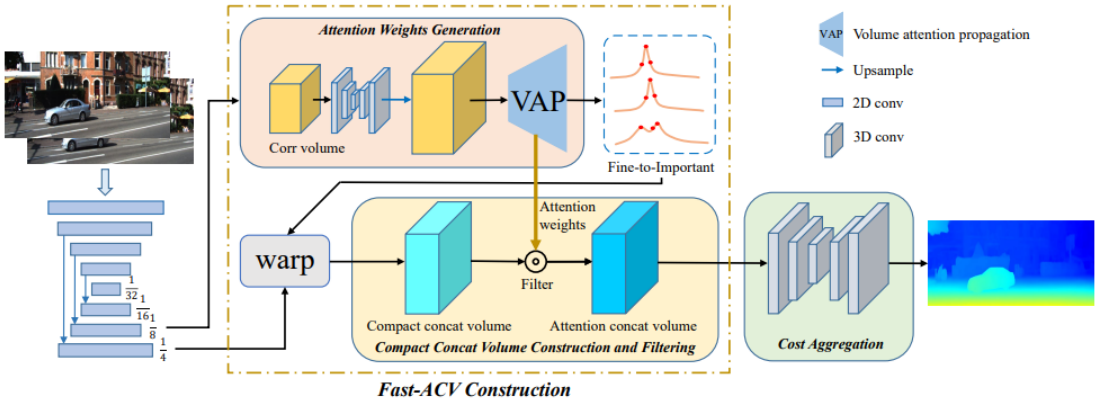
Σχήμα 5.21: Αρχιτεκτονική του μοντέλου ACVNet

Στο Σχήμα 5.21 παρουσιάζεται η αρχιτεκτονική του μοντέλου ACVNet, καθώς και οι διαδικασίες που περνάει το κάθε ζευγάρι φωτογραφιών. Οι φωτογραφίες περνούν από τον “feature extractor”, για την εξαγωγή των πληροφοριών των φωτογραφιών. Το συγκεκριμένο κομμάτι αποτελείται από CNN τύπου ResNet [39], αποτελούμενο από 3 συνελικτικά στρώματα και 22 υπολειπόμενα στρώματα που παράγουν πληροφορία σχετικά με τα χαρακτηριστικά. Επιπρόσθετα, στο μοντέλο εμπεριέχεται το ACV κομμάτι που έχει προαναφερθεί, το οποίο αποτελείται από τρία κύρια βήματα: τη δημιουργία του σημείου συνένωσης, την παραγωγή των βαρών που θα χρησιμοποιηθούν στο φιλτράρισμα και την εκμετάλλευση των παραγόμενων βαρών με τη χρήση του αρχικού σημείου συνένωσης για τη δημιουργία δεδομένων που αφορά τη συνένωσή τους. Στη συνέχεια, αθροίζεται το κόστος στο σημείο του “Cost Aggregation”, με τη χρήση τεσσάρων 3Δ συνελικτών και συνάρτηση ενεργοποίησης τύπου ReLU. Τέλος, υπάρχει το κομμάτι στο οποίο δημιουργούνται οι εκτιμήσεις της ανισότητας. Πιο συγκεκριμένα, δημιουργούνται τρεις έξοδοι με τη χρήση 3Δ συνελικτών και τη μέθοδο soft argmin που υπολογίζει την πιθανότητα των εκτιμώμενων ανισοτήτων [94]. Η soft argmin υπολογίζεται ως εξής:

$$d = \sum_{k=0}^{D_{max}-1} k \cdot p_k \quad (5.1)$$

Όπου το k να αναφέρεται στον βαθμό ανισότητας, το p_kνα δείχνει την αντίστοιχη πιθανότητα και οι τρεις εκτιμήσεις βάθους να αναφέρονται ως d0, d1 και d2.

Μια εναλλαγή του μοντέλου ACVNet είναι το Fast-ACVNet [95]. Το Fast-ACVNet, όπως λέει και ο τίτλος του, είναι μια ταχύτερη έκδοση του ACVNet, δημιουργημένο για να εφαρμόζεται σε συνθήκες που απαιτούν τις εκτιμήσεις σε πραγματικό χρόνο. Το Fast-ACVNet υλοποιεί τον ίδιο “feature extractor” που έχει και το ACVNet, αλλά με λιγότερα στρώματα.



Σχήμα 5.22: Αρχιτεκτονική Fast-ACVNet [95]

5.3.1 Διαδικασία Εκπαίδευσης

Για την εκπαίδευση του μοντέλου χρησιμοποιήθηκαν συνολικά τρία σύνολα δεδομένων, το KITTI συμπεριλαμβανομένου του KITTI 2012 και KITTI 2015, αλλά και το Scene Flow. Τα συγκεκριμένα σύνολα δεδομένων έχουν χρησιμοποιηθεί και στο HITNet [89]. Το προ-εκπαιδευμένο μοντέλο που χρησιμοποιήθηκε για τις δοκιμές βασίζεται στην αρχιτεκτονική του Fast-ACVNet και εκπαιδεύτηκε για 64 εποχές στο dataset SceneFlow. Κατά τη διάρκεια της εκπαίδευσης στο dataset SceneFlow, ο ρυθμός εκπαίδευσης ήταν στο 0.001 και μειωνόταν σταδιακά με την πάροδο των εποχών στις 32, 40 και 56 εποχές.

Εφόσον το μοντέλο είχε εκπαιδευτεί στο σύνολο δεδομένων Scene Flow, στη συνέχεια εκπαιδεύτηκε στο KITTI. Πιο συγκεκριμένα, χρησιμοποιήθηκαν 500 εποχές για να εκπαιδευτεί το μοντέλο στη μίξη των δύο υποσυνόλων KITTI 2012 και KITTI 2015 και άλλες 500 εποχές για να εκπαιδευτεί ξεχωριστά σε κάθε ένα από αυτά. Ο ρυθμός εκπαίδευσης ήταν επίσης στο 0.001, ο οποίος διχοτομείται στη μέση μετά από την 300η εποχή. Η κύρια έκδοση του μοντέλου ACVNet περιέχει 6.22 εκατομμύρια παραμέτρους εκπαίδευσης, ενώ δεν δίνεται ξεκάθαρος αριθμός σχετικά με την εκδοχή του Fast-ACVNet. Σύμφωνα με το [95], μπορούμε να συμπεράνουμε ότι είναι μικρότερος από το ACVNet.

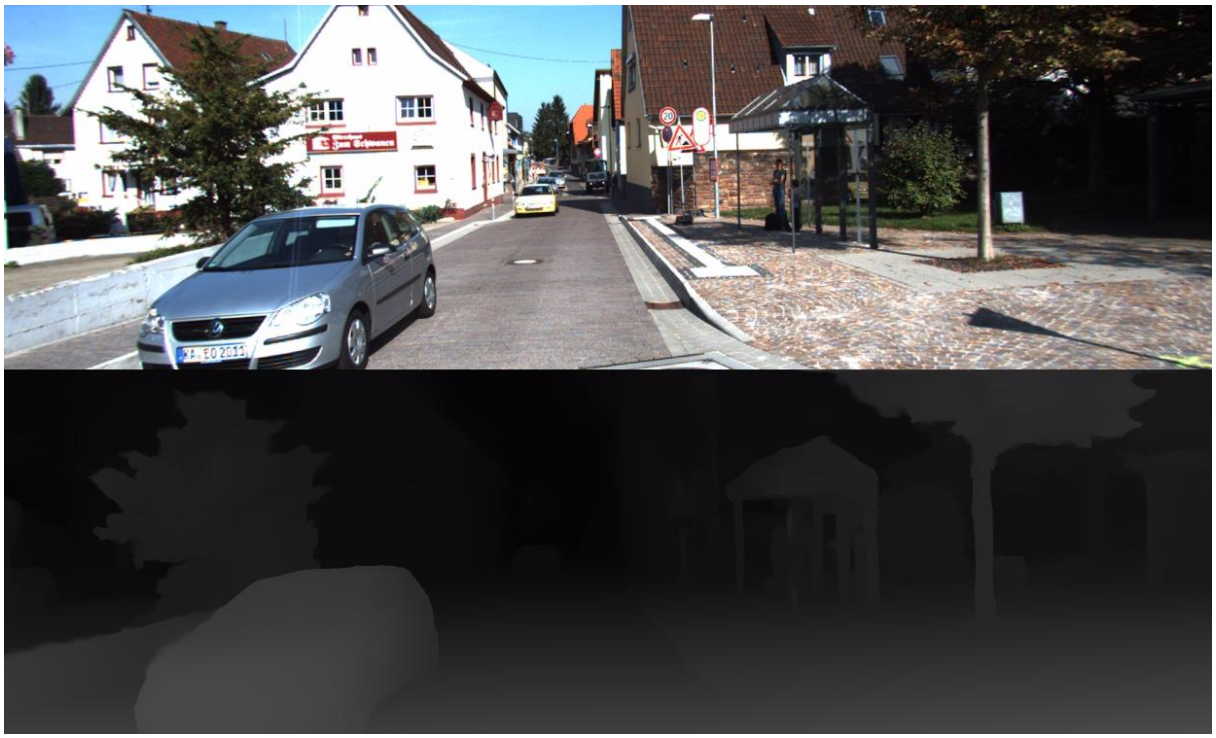
5.3.2 Τεχνικά και Τεχνολογικά χαρακτηριστικά

Η υλοποιήσεις του μοντέλου του Fast-ACVNet μοιάζουν αρκετά σε τεχνολογικό επίπεδο με αυτό του μοντέλου HITNet. Αρχικά, αξίζει να αναφερθεί ότι οι υλοποιήσεις του ACVNet δοκιμάστηκαν στην έκδοση της Python 3.8.16, η οποία κυκλοφόρησε στις 6 Δεκεμβρίου του 2022. Η Python 3.8 και η Python 3.10 δεν έχουν μεγάλες διαφορές, καθώς ανήκουν και οι δύο στην οικογένεια της Python 3. Παρά τις ομοιότητες του HITNet και του ACVNet, όσον αφορά το τεχνολογικό κομμάτι, η κύρια διαφορά τους έγκειται στην πλατφόρμα μηχανικής μάθησης που χρησιμοποίησαν για τις υλοποιήσεις τους.

Ενώ τα προηγούμενα μοντέλα που παρουσιάστηκαν είχαν μοντέλα μηχανικής μάθησης γραμμένα στην πλατφόρμα TensorFlow, η υλοποίηση του ACVNet και του Fast-ACVNet χρησιμοποιούν PyTorch και συγκεκριμένα την PyTorch Cuda 11.3. Άλλες τεχνολογίες που μοιράζεται το συγκεκριμένο μοντέλο είναι το OpenCV, η βιβλιοθήκη NumPy, το matplotlib κ.ά.

5.3.3 Αποτιμήσεις και Εκτιμήσεις.

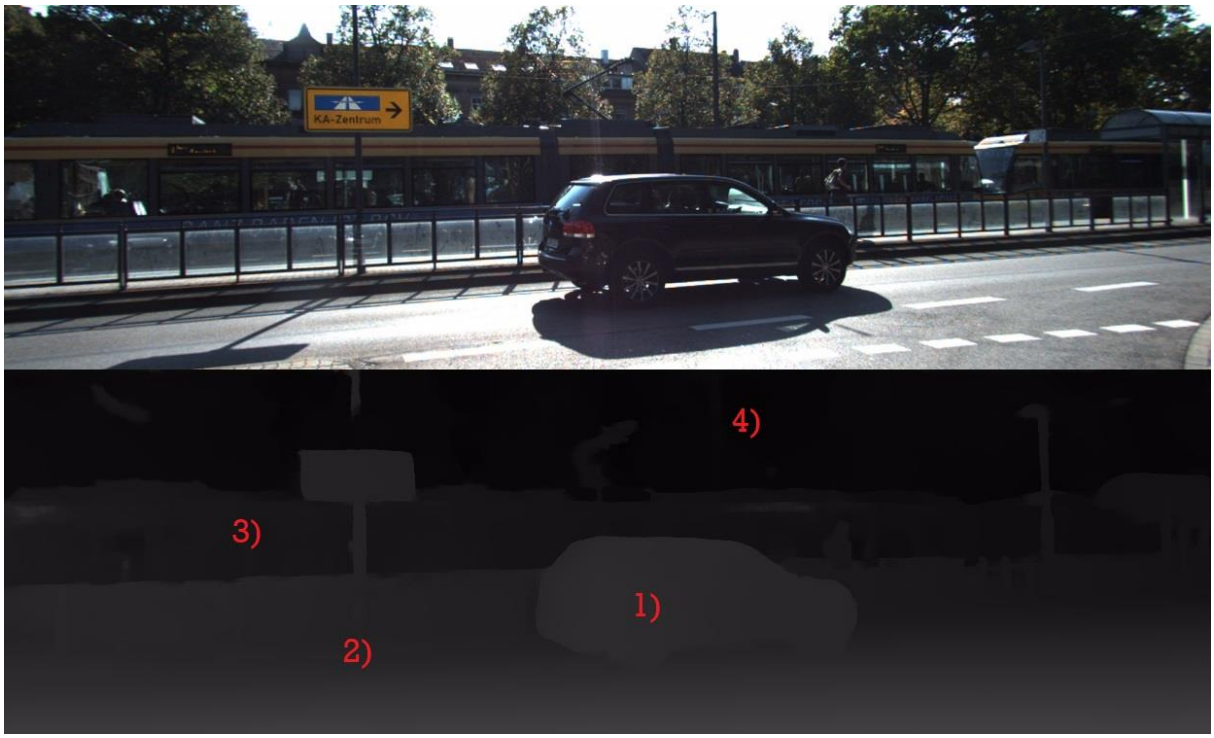
Στο συγκεκριμένο κεφάλαιο θα παρουσιαστούν τα αποτελέσματα, οι εκτιμήσεις και οι βιβλιογραφικές αποτιμήσεις. Πιο συγκεκριμένα θα συγκρίνουμε τις πηγαίες φωτογραφίες με τις αντίστοιχες εκτιμήσεις τους, θα κοιτάξουμε τα αποτελέσματα των μετρικών που αναφέρονται στη βιβλιογραφία και θα γίνει μία κριτική όσον αφορά το μοντέλο Fast-ACVNet.



Σχήμα 5.23: Πηγαία φωτογραφία από το KITTI 2015 (πάνω), και η αντίστοιχη εκτίμηση του Fast-ACVNet (κάτω) – Δρόμος μεγάλης πυκνότητας αντικειμένων (1).

Στο Σχήμα 5.23 μπορούμε και συγκρίνουμε μία πηγαία φωτογραφία του συνόλου δεδομένων KITTI 2015 και το αντίστοιχο εκτιμώμενο βάθος από το μοντέλο Fast-ACVNet. Η συγκεκριμένη φωτογραφία έχει χρησιμοποιηθεί για σύγκριση και στα προηγούμενα δεδομένα, αλλά αυτό δεν καθιστά τα αποτελέσματα λιγότερο ενδιαφέροντα.

Αρχικά, το μοντέλο καταφέρνει και αναγνωρίζει τα κύρια χαρακτηριστικά και αντικείμενα μέσα στον χώρο. Στο κάτω σημείο του Σχήματος 5.23 αυτό φαίνεται με τον διαχωρισμό του αυτοκινήτου μπροστά, των δέντρων δεξιά και αριστερά, αλλά και των πινακίδων και της στάσης λεωφορείου που βρίσκονται δεξιά από τον δρόμο. Αξίζει να αναφερθεί πως όσο πιο μακριά φαίνεται ένα σημείο στη φωτογραφία, τόσο δυσκολεύεται το μοντέλο να το διαχωρίσει από τα υπόλοιπα αντικείμενα. Για παράδειγμα, ενώ μπορούμε να αναγνωρίσουμε μία φιγούρα που μπορεί να αντιπροσωπεύει το ταξί στη μέση της φωτογραφίας, δεν φαίνεται να υπάρχει κάποιος περεταίρω διαχωρισμός των αυτοκινήτων πίσω από το ταξί. Κάτι άλλο που μπορεί να παρατηρηθεί είναι η έλλειψη διαχωρισμού των διαβαθμίσεων του βάθους στα κτήρια. Δεξιά και αριστερά του δρόμου υπάρχουν δύο κτήρια, στην εκτίμηση, όμως, ενώ υπάρχει διαχωρισμός τους ως προς τον ορίζονται και το υπόλοιπο πεδίο της φωτογραφίας, δεν δείχνει να παρουσιάζεται ο διαχωρισμός τους ως προς τα διαφορετικά σημεία τους. Για παράδειγμα, το σπίτι στα δεξιά περιέχει σημεία που βρίσκονται πιο κοντά και πιο μακριά από την κάμερα. Το κάτω σημείο της σκεπής βρίσκεται πιο κοντά στην κάμερα από το πίσω μέρος του σπιτιού, αλλά τα αποτελέσματα δεν φαίνεται να αντιπροσωπεύουν αυτήν τη διαφορά τους ως προς το βάθος.



Σχήμα 5.24: Πηγαία φωτογραφία από το KITTI 2015 (πάνω), και η αντίστοιχη εκτίμηση του Fast-ACVNet (κάτω) – Αυτοτελή Κτήρια.

Στο Σχήμα 5.24 παρουσιάζεται μία πηγαία φωτογραφία του KITTI 2015 dataset και η αντίστοιχη εκτίμηση που έβγαλε το μοντέλο Fast-ACVNet ως προς το βάθος. Για το συγκεκριμένο παράδειγμα έχουν δημιουργηθεί κάποιες σημειώσεις για συγκεκριμένα σημεία όσον αφορά την εκτίμηση. Το πρώτο σημείο της εκτίμησης που έχει σημειωθεί με τον αριθμό 1 αντιπροσωπεύει ένα αυτοκίνητο που φαίνεται στο πλαίσιο της πηγαίας φωτογραφίας. Ενώ η εκτίμηση δείχνει ξεκάθαρα να το αναγνωρίζει ως αυτοκίνητο, ουσιαστικά τα δεδομένα εκπαίδευσης το αναγνωρίζουν σαν αντικείμενο που είναι πολύ κοντά στην κάμερα. Αυτό μας παραθέτει στο σημείο με τον αριθμό 2, που απεικονίζει τον δρόμο της πηγαίας φωτογραφίας. Ο δρόμος περιέχει σημεία που αντιστοιχούν σε διαφορετικό βάθος, στην εκτίμηση, όμως, φαίνεται μία ενοποιημένη εκτίμηση με ίδιες ή αντίστοιχες τιμές. Αξίζει να αναφερθεί πως υπάρχουν σημεία στον δρόμο που, ενώ θα έπρεπε να γίνουν αντιληπτά ως πιο κοντινά στην κάμερα ως προς το αντικείμενο στο σημείο 1, αναγνωρίζονται να βρίσκονται πιο μακριά από ότι θα έπρεπε. Το σημείο 3 δείχνει το κτήριο που κατά κύριο λόγο απεικονίζεται σε αυτήν τη φωτογραφία. Το αναφερόμενο κτήριο φαίνεται να είναι μία εξωτερική στάση από μέσο μαζικής μεταφοράς. Στην φωτογραφία απεικονίζεται με παρόμοιο βάθος, αλλά με διαφορετικές διαβαθμίσεις. Όπως φαίνεται και στη φωτογραφία, τα σημεία του κτηρίου στη δεξιά πλευρά θα πρέπει να βρίσκονται πιο μακριά σε απόσταση συγκριτικά με το δεξιό μέρος του κτηρίου. Οι αλλαγές στις διαβαθμίσεις του χρώματος φαίνεται να αντιπροσωπεύουν τη διαφορά στο βάθος τους, αλλά δεν φαίνεται να αντιπροσωπεύει αυτό που περιέχει το κτήριο, το οποίο είναι οι καθρέπτες. Ενώ είναι κομμάτι του κτηρίου, υπάρχουν σημεία στο 3, που απεικονίζουν το βάθος που απεικονίζεται στο 4. Το σημείο 4 θα έπρεπε να χωριστεί σε δύο σημεία, με το πρώτο σημείο να πρέπει να αναγνωρίζει τα σπίτια και το συγκριτικό βάθος που έχουν ως προς την κάμερα και το δεύτερο σημείο να αναγνωρίζει τον ουρανό και τον ορίζοντα. Η εκτίμηση του μοντέλου, όμως, ενοποιεί αυτά τα δύο διαφορετικά αντικείμενα, υποθέτοντας ότι βρίσκονται σε ίδια απόσταση, χωρίς να τα διαχωρίζει.

Σχήμα 5.25: Πηγαία φωτογραφία από το KITTI 2015 (πάνω), και η αντίστοιχη εκτίμηση του Fast-ACVNet (κάτω) – Δρόμος μεγάλης πυκνότητας αντικειμένων (2).

Το Σχήμα 5.25 παρουσιάζει επίσης την εκτίμηση του Fast-ACVNet σε φωτογραφία που απεικονίζει δρόμο με μεγάλη πυκνότητα αντικειμένων. Συγκεκριμένα, μπορούμε να διακρίνουμε πολλά αυτοκίνητα σκορπισμένα στον χώρο της φωτογραφίας. Αντίστοιχα στην εκτίμηση, γίνεται ένας κύριος διαχωρισμός των αυτοκινήτων και ταυτόχρονα γίνεται μια καλή προσέγγιση για να εκτιμηθεί το αντίστοιχο βάθος τους. Φυσικά, η εκτίμηση πάσχει από τα ίδια προβλήματα που είχαν και οι προηγούμενες εκτιμήσεις του Fast-ACVNet. Αριστερά του δρόμου, βρίσκονται δύο γκρι αυτοκίνητα παρκαρισμένα, που βρίσκονται ξεκάθαρα σε διαφορετική απόσταση από την κάμερα. Τα δεδομένα της εκτίμησης, όμως, δεν διαχωρίζουν την απόστασή τους.

Το Σχήμα 5.26 περιέχει την πηγαία φωτογραφία και την αντίστοιχη εκτίμηση του μοντέλου σε δρόμο που φαίνεται να είναι εκτός κατοικημένης περιοχής και, συγκεκριμένα, σε έναν δρόμο περιφερειακό ή ταχείας κυκλοφορίας, που δεν υπάρχουν πολλά κτήρια στο πλαίσιο της φωτογραφίας. Παρατηρώντας την εκτίμηση, φαίνεται ξεκάθαρα πως το Fast-ACVNet έχει καταλάβει τη γεωμετρία της φωτογραφίας και αντίστοιχα κάνει εκτιμήσεις που αντιπροσωπεύουν ή προσεγγίζουν τις αληθινές τιμές. Άξιος αναφοράς είναι ο διαχωρισμός της μπάρας στα αριστερά του δρόμου που απεικονίζεται στην εκτίμηση με τιμές που αντιπροσωπεύουν μικρότερη απόσταση από την κάμερα.

Σχήμα 5.26: Πηγαία φωτογραφία από το KITTI 2015 (πάνω), και η αντίστοιχη εκτίμηση του Fast-ACVNet (κάτω) – Δρόμος εκτός κατοικημένης.

Η ολική συμπεριφορά του ACVNet αναφορικά με τις εκτιμήσεις είναι πολύ θετική. Δεδομένου ότι τα αποτελέσματα εκτιμήσεων που παρουσιάστηκαν ήταν οι εκτιμήσεις που έβγαλε η παραποιημένη έκδοση Fast-ACVNet, οι εκτιμήσεις φαίνονται να εκπροσωπούν ή τουλάχιστον να προσεγγίζουν ως ένα βαθμό την αληθινή γεωμετρική ερμηνεία του βάθους των φωτογραφιών. Φυσικά, αυτοί οι σχολιασμοί είναι υποκειμενικοί και βασίζονται στον τρόπο με τον οποίο κρίνει ένας άνθρωπος με τα μάτια του μία εκτίμηση βάθους ως προς την κάμερα, που παρουσιάζεται σε σχετικές τιμές.

Στον Πίνακα 5.4 παρουσιάζονται τα συγκριτικά αποτελέσματα των μετρικών του KITTI 2015, μεταξύ των μοντέλων του ACVNet και του μοντέλου του HITNet. Παρατηρούμε πως το ACVNet βγάζει καλύτερα αποτελέσματα στο KITTI 2015 συγκριτικά με τα υπόλοιπα μοντέλα. Στα παραδείγματα παραπάνω, όμως, παρουσιάστηκαν οι εκτιμήσεις από το Fast-ACVNet μοντέλο, για το οποίο παρατηρούμε ότι έχει τα χειρότερα αποτελέσματα συγκριτικά με τις υπόλοιπες εκδοχές του ACVNet, χωρίς όμως αυτό να καθιστά τα αποτελέσματά του λανθασμένα.

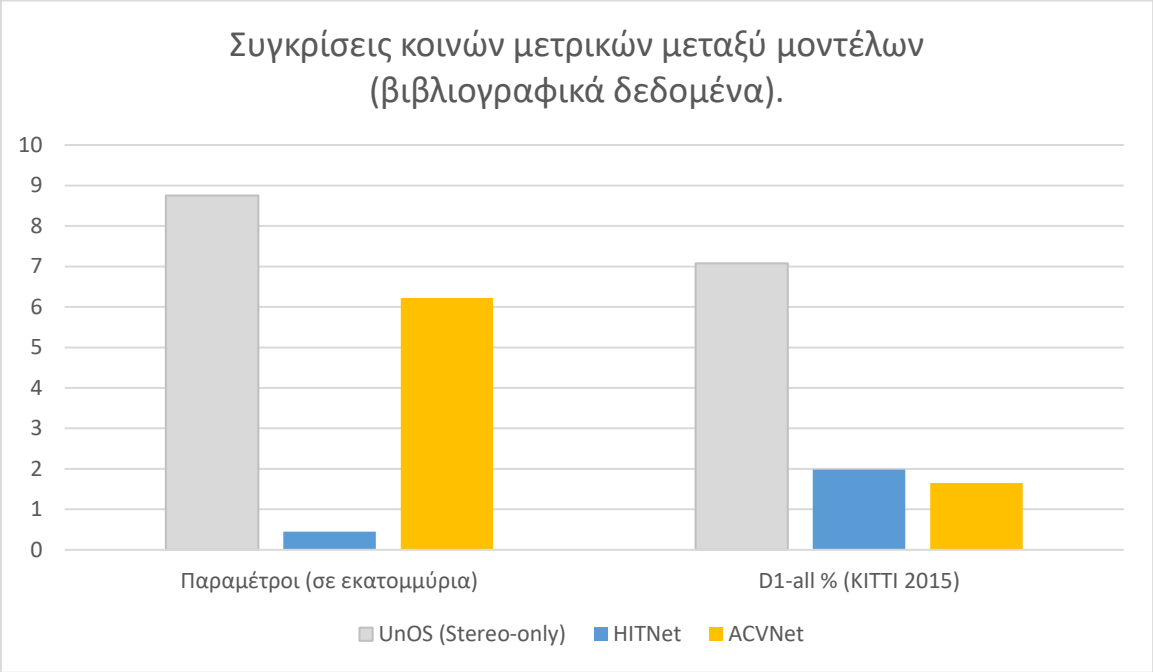
Πίνακας 5.4: Συγκρίσεις αποτελεσμάτων στις μετρικές του KITTI 2015 D1-bg, D1-fg και D1-all (μικρότερο = καλύτερο), μεταξύ του HITNet [89] και των μοντέλων ACVNet, Fast-ACVNet και Fast-ACVNet+ [94, 95].

Μέθοδος	KITTI 2015		
	D1-bg	D1-fg	D1-all
ACVNet	**1.37%**	**3.07%**	**1.65%**
Fast-ACVNet	1.82%	3.93%	2.17%
Fast-ACVNet+	1.70%	3.53%	2.01%
HITNet	1.74%	3.20%	1.98%

5.4 Συγκρίσεις μοντέλων.

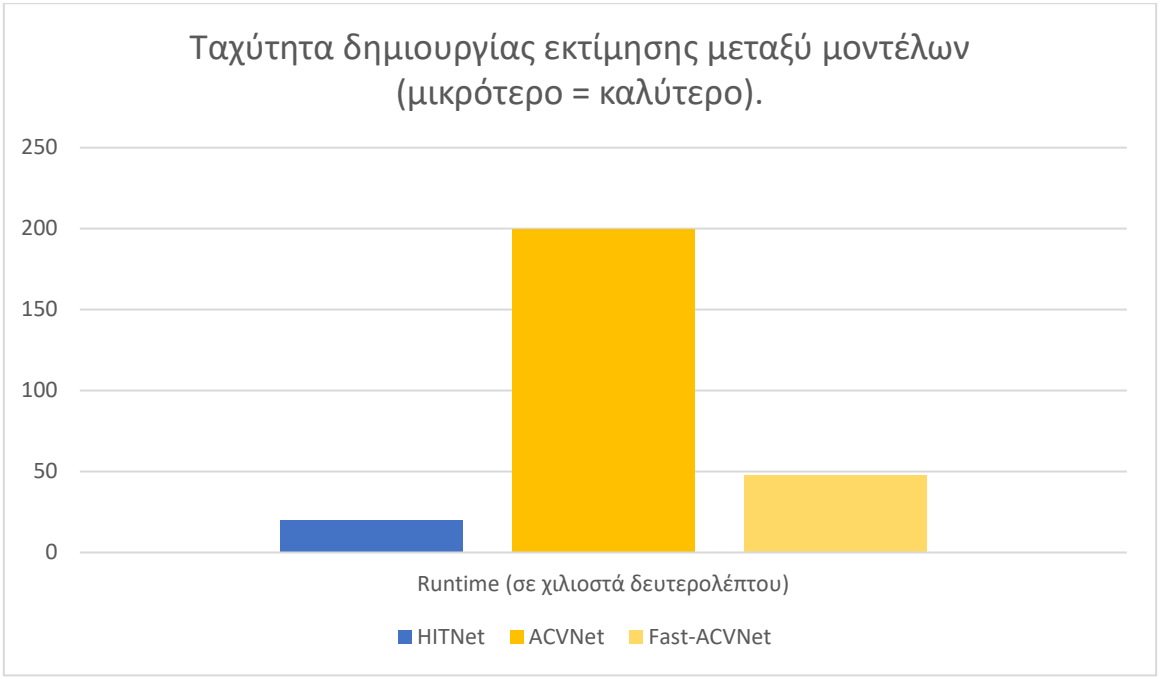
Στο συγκεκριμένο κομμάτι της εργασίας, θα παρουσιαστούν μετρικές που συγκρίνουν τα προαναφερθέντα μοντέλα. Ειδικότερα, θα παρουσιαστούν μετρικές που υπάρχουν στα σύνολα δεδομένων ETH3D, KITTI και SceneFlow. Μετά από τις μετρικές, θα γίνει μία σύγκριση των χαρακτηριστικών των μοντέλων και των εκτιμήσεών τους.

5.4.1 Συγκρίσεις Μετρικών.



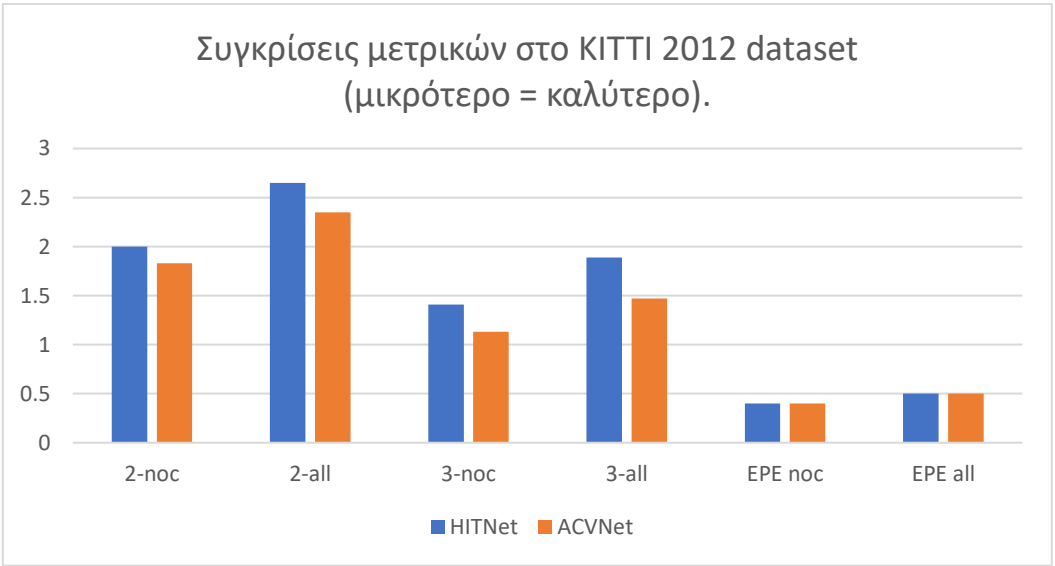
Σχήμα 5.27: Συγκρίσεις μεταξύ κοινών μετρικών και μετρήσιμων χαρακτηριστικών των μοντέλων UnOS, HITNet και ACVNet βάσει της βιβλιογραφίας τους.

Τα μοντέλα HITNet, ACVNet και UnOS, ενώ κατά κύριο λόγο μπορούν και επιλύουν ως έναν βαθμό την ίδια εργασία, δεν τείνουν να έχουν πολλά κοινά σημεία. Μεταξύ των λίγων κοινών σημείων και μετρικών που μπορούν να συγκριθούν είναι η μετρική D1-all και οι παράμετροι. Τα συγκεκριμένα στοιχεία ξαναπαρουσιάζονται στο Σχήμα 5.27, όπου μπορούμε και παρατηρούμε κάτι πολύ ενδιαφέρον. Κατά κύριο μπορούμε να σκεφτούμε πως όσο μικρότεροι είναι οι αριθμοί που παρουσιάζονται στα γραφήματα, τόσο καλύτερα είναι τα αποτελέσματα. Από τα αποτελέσματα του σχήματος παρατηρείται εύκολα ότι οι επιδόσεις και τα χαρακτηριστικά του UnOS δεν είναι στο ίδιο επίπεδο που είναι και τα υπόλοιπα δίκτυα και καθίστανται σαφώς κατώτερα από το HITNet και το ACVNet. Όσον αφορά τις παραμέτρους, η μικρότερη ποσότητά τους καθιστά ένα μοντέλο μικρότερο σε μέγεθος, και ταυτόχρονα πιο ελαστικό για χρήσεις σε πιο αδύναμες συσκευές. Οι παράμετροι του HITNet είναι λιγότερες από μισό εκατομμύριο, αλλά ταυτόχρονα πετυχαίνει καλύτερα αποτελέσματα στο KITTI 2015 από το UnOS και φτάνει σχεδόν τις επιδόσεις του ACVNet. Από την άλλη, ενώ το ACVNet έχει τα καλύτερα αποτελέσματα, όσον αφορά το KITTI 2015 dataset, αυτό μένει πίσω στο θέμα της ελαστικότητας, καθώς ο αριθμός των παραμέτρων του έχει ξεπεράσει τις 6 εκατομμύρια, φτάνοντας σχεδόν τον αριθμό παραμέτρων του UnOS.



Σχήμα 5.28: Σύγκριση μεταξύ HITNet, ACVNet και Fast-ACVNet διάρκειας χρόνου για τη δημιουργία εκτίμησης βάθους.

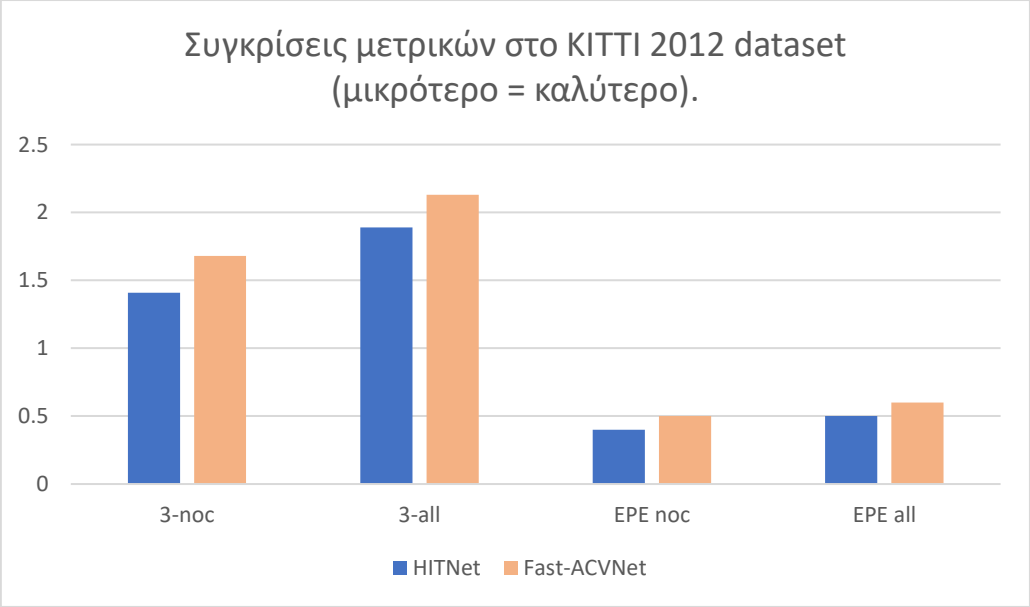
Όσον αφορά την ταχύτητα των μοντέλων, τα αποτελέσματα αντικατοπτρίζουν αυτά των παραμέτρων. Το HITNet είναι το μικρότερο μοντέλο σε αριθμό παραμέτρων, άρα χρειάζεται και τη μικρότερη διάρκεια για τη δημιουργία εκτίμησης. Αντίστοιχα, το ACVNet, που έχει 6,22 εκατομμύρια παραμέτρους, χρειάζεται 2 δέκατα του δευτερολέπτου για να δημιουργήσει μία εκτίμηση, καθιστώντας το, το αργότερο μεταξύ των τριών μοντέλων που παρουσιάζονται στο Σχήμα 5.28. Επιπλέον, στο σχήμα προστέθηκαν και οι επιδόσεις αναφορικά με την ταχύτητα του υλοποιήσιμου Fast-ACVNet. Ο τίτλος του μοντέλου εννοεί ότι δημιουργεί εκτιμήσεις πιο γρήγορα συγκριτικά με την απλή έκδοση, ενώ το runtime επιβεβαιώνει αυτήν τη θεωρία. Αν και δεν φτάνει τις επιδόσεις του HITNet ως προς την ταχύτητα, το Fast-ACVNet κάνει βήματα προς τη σωστή κατεύθυνση.



Σχήμα 5.29: Συγκρίσεις μεταξύ HITNet και ACVNet στις μετρικές του dataset αποτίμησης KITTI 2012.

Στο Σχήμα 5.29 παρουσιάζονται τα συγκριτικά αποτελέσματα μεταξύ HITNet και ACVNet, στις μετρικές του συνόλου δεδομένων KITTI 2012. Μέσα στις μετρικές εμπεριέχονται τα x-noc και x-all, όπου το x αντιπροσωπεύει την τιμή ενός αριθμού και το noc / all αντιπροσωπεύει τη φύση των αληθινών τιμών σύγκρισης. Ουσιαστικά το noc (no occlusions) αναφέρεται στα pixels των αληθινών τιμών για τα οποία υπάρχει αξιόπιστη πληροφορία, ενώ το all αντιπροσωπεύει όλες τις πληροφορίες των αληθινών τιμών. Το x που αναφέρθηκε πριν αντιπροσωπεύει τον μέγιστο αριθμό pixels όπου το σφάλμα στην ανισότητα δεν μετριέται. Ουσιαστικά, δουλεύει με τον ίδιο τρόπο που δουλεύει το d1-all, απλά για διαφορετικό αριθμό pixels. Τέλος, η μετρική EPE υπολογίζει τη μέση ευκλείδεια απόσταση μεταξύ της εκτιμώμενης και της αληθινής τιμής της ανισότητας.

Βάσει των αποτελεσμάτων που παρουσιάζονται στο Σχήμα 5.29, μπορεί να παρατηρηθεί η διαφορά των μοντέλων και σε άλλα πλαίσια εκτός του KITTI 2015. Συγκεκριμένα, με εξαίρεση τις μετρικές EPE στις οποίες τα δύο παρουσιαζόμενα μοντέλα τείνουν να έχουν παρόμοιες επιδόσεις, σε όλες τις υπόλοιπες επιδόσεις το ACVNet υπερτερεί έναντι του HITNet, χωρίς όμως η διαφορά αυτή να έχει μεγάλη ένταση.



Σχήμα 5.30: Συγκρίσεις μεταξύ του HITNet και του Fast-ACVNet, στο dataset KITTI 2012.

Στις εκτιμήσεις που παρουσιάστηκαν παραπάνω στο κείμενο, όσον αφορά την υλοποίηση του ACVNet, οι εκτιμήσεις ήταν του υπό-μοντέλου Fast-ACVNet. Το Fast-ACVNet, όπως έχει προαναφερθεί, είναι μία ελαφρύτερη και μικρότερη έκδοση του Fast-ACVNet. Επίσης, στους σχολιασμούς των εκτιμήσεων, έγινε αναφορά σε σημεία στα οποία δυσκολευόταν το μοντέλο να διαχωρίσει συγκεκριμένα αντικείμενα εντός του πλαισίου της φωτογραφίας. Αντίστοιχα, στις μετρικές του Σχήματος 5.30, μπορούμε να παρατηρήσουμε ότι το μοντέλο HITNet υπερτερεί σε όλες τις παρουσιαζόμενες μετρικές, χωρίς όμως να υπάρχει συγκριτικά μεγάλη διαφορά. Άξιο αναφοράς είναι πως, ενώ στο Σχήμα 5.29 φαίνεται πως το HITNet έχει αντίστοιχες επιδόσεις όσον αφορά τη μετρική EPE, εδώ βλέπουμε πως το Fast-ACVNet έχει χειρότερες επιδόσεις συγκριτικά με το HITNet, άρα και συγκριτικά με το ACVNet. Οπότε εδώ τίθεται το ερώτημα, αν αξίζει αυτό το κόστος σε επιδόσεις που προτίθεται να πληρώσει κάποιος με τη χρήση του Fast-ACVNet έναντι του ACVNet, για να κερδίσει σε μεγάλο βαθμό ως προς την ταχύτητα.

Πίνακας 5.5: Σύγκριση μετρικών των dataset ETH3D και SceneFlow για τα μοντέλα HITNet, ACVNet, Fast-ACVNet (μικρότερο = καλύτερο).

Μοντέλο	Scene Flow	ETH3D	
	EPE (px)	bad 1.0 (%)	bad 2.0 (%)
HITNet	**0.43**	2.79	0.8
ACVNet	0.48	**2.58**	**0.57**
Fast-ACVNet	0.64	-	-

Στον Πίνακα 5.5 παρουσιάζονται οι συγκρίσεις των μετρικών των Scene Flow και ETH3D datasets. Όπως και στα προηγούμενα σύνολα δεδομένων, υπάρχουν μετρικές που υπολογίζουν το ποσοστό των pixel, στα οποία η εκτιμώμενη ανισότητα είναι λανθασμένη κατά x pixels. Αυτές οι μετρικές είναι οι bad 1.0 και bad 2.0 που αναφέρονται σε ποσοστό που η εκτιμώμενη ανισότητα είναι λανθασμένη κατά ένα ή δύο pixel αντίστοιχα. Το EPE, όπως αναφέρθηκε προηγουμένως στο κείμενο, αντικατοπτρίζει τη μέση ευκλείδεια διανυσματική απόσταση ως προς τις αληθινές τιμές. Στο dataset Scene Flow μπορούμε να συμπεράνουμε πως τα αποτελέσματα του HITNet είναι σαφώς καλύτερα, έτσι καθίσταται η καλύτερη επιλογή, λαμβάνοντας υπόψιν ότι το HITNet είναι το μικρότερο και γρηγορότερο από τα παρουσιαζόμενα μοντέλα. Φυσικά, όσον αφορά τις αποτιμήσεις του ETH3D, παρατηρούμε ενδιαφέροντα αποτελέσματα μεταξύ του HITNet και του ACVNet. Συγκρίνοντας σε πραγματικούς αριθμούς, η διαφορά που έχουν αυτά τα δύο μοντέλα στο 1 pixel είναι μικρότερη από τη διαφορά που έχουν αυτά τα δύο μοντέλα σε pixels. Αυτή είναι μία ενδιαφέρουσα παρατήρηση αναφορικά με τη συμπεριφορά των μοντέλων.

5.4.2 Συγκρίσεις Εκτιμήσεων.

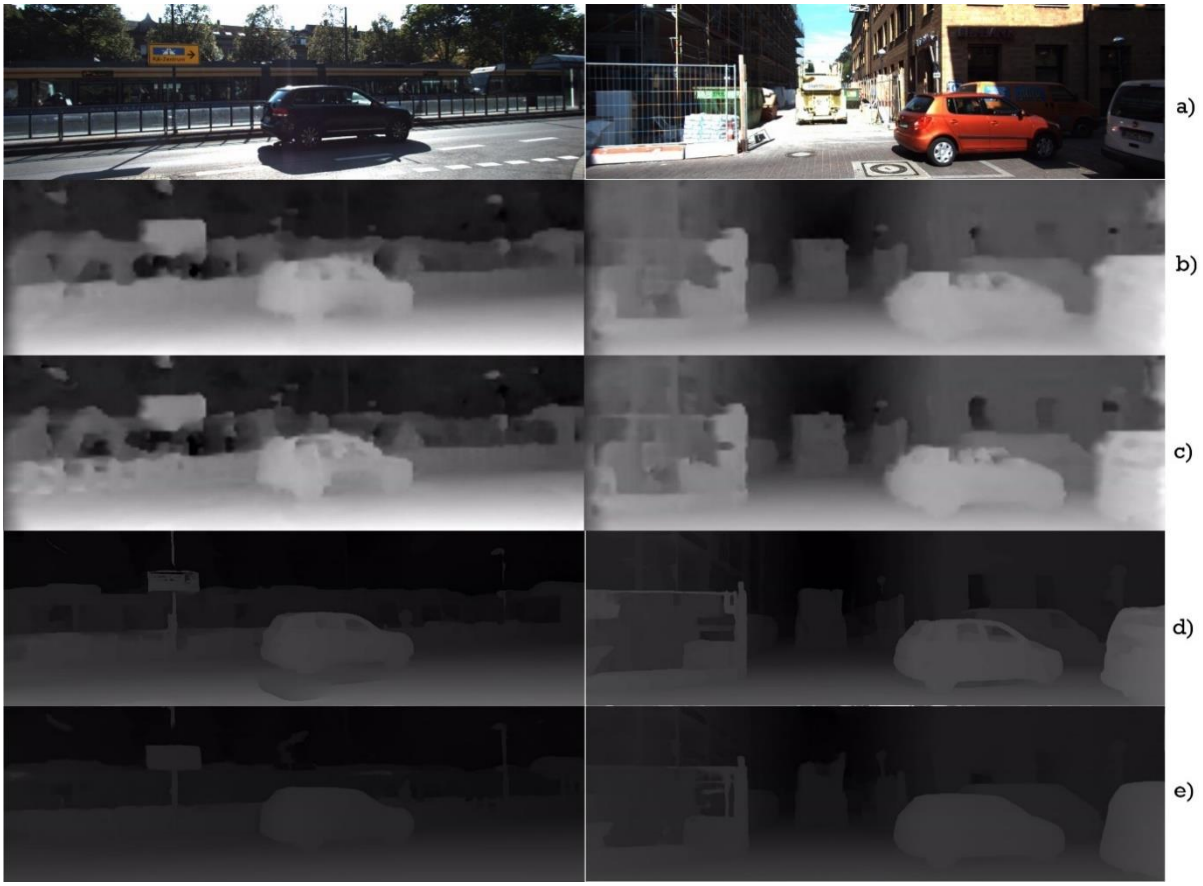
Στο συγκεκριμένο κομμάτι της εργασίας θα παρουσιαστούν σε παρτίδες, πηγαίες φωτογραφίες από το KITTI 2015 dataset και οι αντίστοιχες εκτιμήσεις των μοντέλων. Συγκεκριμένα, οι εκτιμήσεις θα περιέχουν δύο εκτιμήσεις από το UnOS μοντέλο και τις εκτιμήσεις από τα μοντέλα HITNet και Fast-ACVNet.

Στο σχήμα 5.31 παρουσιάζονται δύο πηγαίες φωτογραφίες του πακέτου δεδομένων KITTI 2015. Οι συγκεκριμένες φωτογραφίες έχουν κατηγοριοποιηθεί ως «αυτοτελή κτήρια». Επιπλέον, στο σχήμα φαίνονται και οι αντίστοιχες εκτιμήσεις από τα μοντέλα UnOS (50,000 και 75,000 εποχές) καθώς και οι εκτιμήσεις του HITNet και Fast-ACVNet.

Στην πρώτη φωτογραφία του Σχήματος 5.31, παρατηρούμε ότι είναι η ίδια φωτογραφία που χρησιμοποιήθηκε και στο Σχήμα 5.24. Συγκρίνοντας τις δύο εκτιμήσεις του UnOS μεταξύ τους, είναι εύκολο να διακριθεί η διαφορά στις λεπτομέρειες και στο πώς τις διαχειρίζονται. Ενώ το μοντέλο που είναι εκπαιδευμένο για 50,000 εποχές (b στήλη) έχει κατανοήσει τη γεωμετρία του αντικειμένου που φαίνεται στο πεδίο της φωτογραφίας, το UnOS που είναι εκπαιδευμένο για 75,000 εποχές δυσκολεύεται να κατανοήσει αυτές τις λεπτομέρειες, με αποτέλεσμα να παράγει μεγάλο θόρυβο. Αντίστοιχα προβλήματα μπορούμε να παρατηρήσουμε και για τις υπόλοιπες λεπτομέρειες της φωτογραφίας, όπως είναι το κτήριο και το φόντο. Και οι δύο προβλέψεις δυσκολεύονται να κατανοήσουν και να ερμηνεύσουν το συγκεκριμένο υπόβαθρο, δίνοντας στην εκτίμηση μεγάλες τιμές θορύβου.

Κρίνοντας από την προαναφερόμενη συμπεριφορά, μπορούμε να καταλάβουμε πως το μοντέλο UnOS δυσκολεύεται να κατανοήσει το περιβάλλον και το βάθος του, όταν μέσα στη φωτογραφία υπάρχει

μεγάλος αριθμός λεπτομέρειας και σχήματα με περίπλοκη γεωμετρία (π.χ. δέντρα). Αντιθέτως, όσον αφορά αντικείμενα απλού σχήματος και ξεκάθαρης τοποθέτησης στο πλαίσιο της εικόνας, το μοντέλο μπορεί να δημιουργήσει μία προσεγγιστική εκτίμηση.

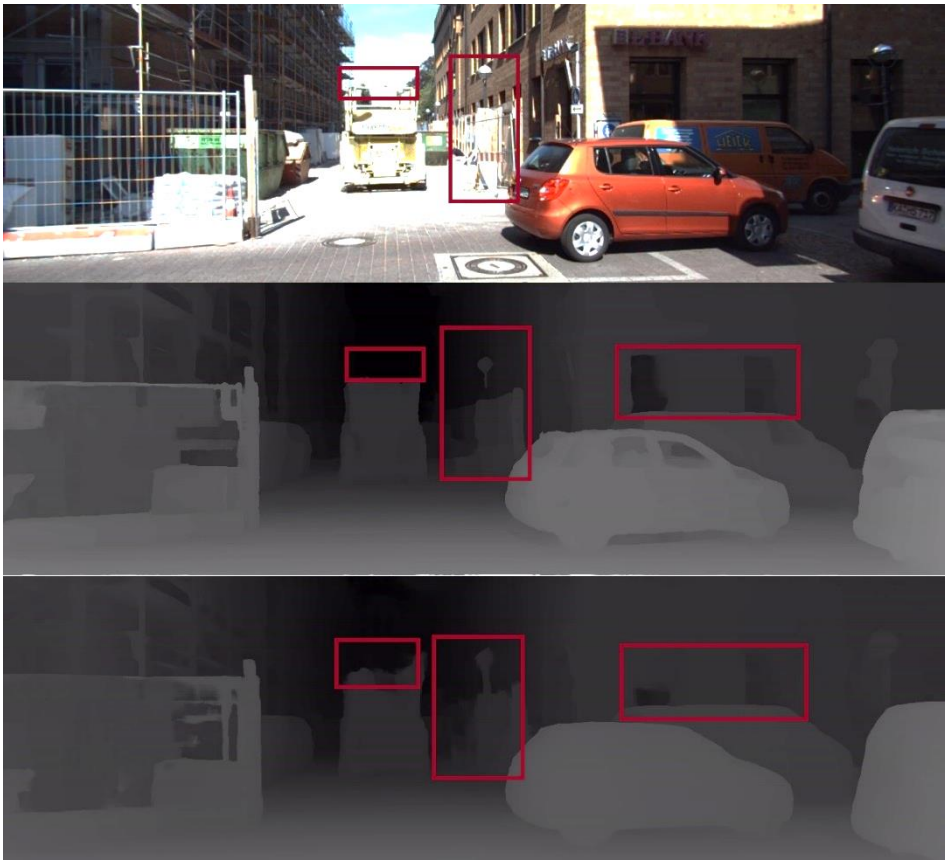


Σχήμα 5.31: Σύγκριση πηγαίων φωτογραφιών από το σύνολο δεδομένων KITTI 2015 (a), κατηγοριοποιημένες σαν «αυτοτελή κτήρια» και οι εκτιμήσεις των μοντέλων. Το UnOS στις 50,000 εποχές αντιστοιχεί στη σειρά b), στις 75,000 εποχές στη σειρά c), ενώ το d) είναι για το HITNet μοντέλο και το e) για το μοντέλο Fast-ACVNet.

Συγκρίνοντας τις εκτιμήσεις της πρώτης εικόνας από το Σχήμα 5.31, μεταξύ των μοντέλων HITNet και ACVNet είναι εμφανές ποιες είναι οι καλύτερες εκτιμήσεις. Σε αντίθεση με τις εκτιμήσεις του UnOS, οι εκτιμήσεις των δύο τελευταίων μοντέλων καταφέρνουν και προσεγγίζουν καλύτερα την πραγματικότητα, κάτι που έχει αναφερθεί και σε προηγούμενα παραρτήματα της εργασίας. Συγκρίνοντας αυτές τις δύο εκτιμήσεις, μπορούν να παρατηρηθούν παρόμοιες εκτιμήσεις με ελάχιστες διαφορές. Μία από αυτές είναι η προσέγγιση στη λεπτομέρεια. Για παράδειγμα, ο τρόπος που το Fast-ACVNet ερμηνεύει το αυτοκίνητο στη μέση της φωτογραφίας το θέτει ως συμπαγές αντικείμενο, ενώ το HITNet κατανοεί ότι υπάρχουν σημεία στο συγκεκριμένο αντικείμενο που είναι διαφανή και παρουσιάζουν κομμάτια διαφορετικού βάθους από αυτό που βρίσκεται το παράθυρο. Αντίστοιχα, το HITNet φαίνεται να παράγει λιγότερο θόρυβο στο φόντο της φωτογραφίας, αν και ο θόρυβος που υπάρχει στην εκτίμηση του Fast-ACVNet είναι ελάχιστος. Ένα πρόβλημα, όμως, που μπορεί να παρατηρηθεί από το HITNet όσον αφορά την εκτίμησή του, είναι ο τρόπος που ερμηνεύει τη σκιά του αυτοκινήτου, δημιουργώντας σφάλμα στην εκτίμηση ως προς το βάθος, κάτι που φαίνεται να λείπει από την εκτίμηση του Fast-ACVNet.

Στη δεύτερη στήλη του Σχήματος 5.31, φαίνεται η δεύτερη φωτογραφία που είναι κατηγοριοποιημένη ως «αυτοτελή κτήρια». Αντίστοιχα με την πρώτη φωτογραφία, το UnOS, αποτυγχάνει να αναγνωρίσει τις λεπτομέρειες και το σχετικό βάθος τους. Όμως, σε αντίθεση με την πρώτη φωτογραφία, εδώ παρουσιάζεται σε μικρότερη ένταση. Αυτό μπορεί να έγκειται περισσότερο στο γεγονός ότι η πηγαία φωτογραφία παρουσιάζει περισσότερα αντικείμενα με απλοϊκό γεωμετρικό σχήμα σε σχέση με την πρώτη φωτογραφία. Στα λίγα αντικείμενα, όμως, που περιέχουν μεγάλο βαθμό λεπτομέρειας, το UnOS αποτυγχάνει να τα διαχωρίσει και αντίστοιχα να εκτιμήσει το βάθος τους. Άξιο αναφοράς είναι, όμως, πως, αν αγνοηθούν οι λεπτομέρειες στην εκτίμηση, και στις δύο εκτιμήσεις (50,000 εποχών και 75,000 εποχών) υπάρχει μια καλή προσέγγιση ως προς το βάθος για τον χώρο που παρουσιάζει το πλαίσιο. Στις εκτιμήσεις των μοντέλων μπορούν να αναγνωριστούν τα κτήρια, ο δρόμος, ο φράκτης, αλλά και ο ορίζοντας, όπως επίσης και τα αντικείμενα που βρίσκονται μπροστά στην κάμερα.

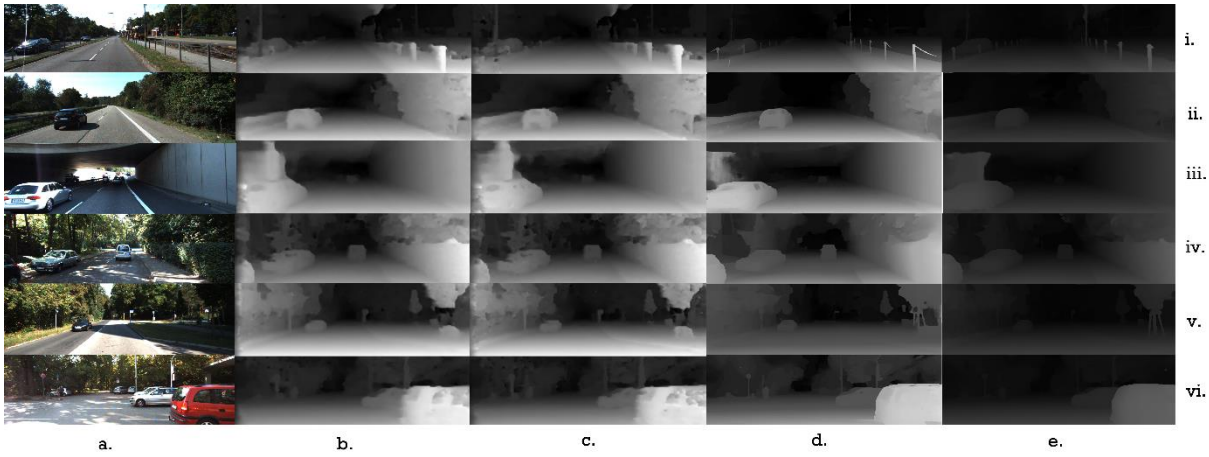
Στις εκτιμήσεις των μοντέλων HITNet και Fast-ACVNet, όμως, υπάρχει μεγαλύτερη προσοχή όσον αφορά τη λεπτομέρεια. Όπως και στην προηγούμενη φωτογραφία, το Fast-ACVNet αδυνατεί να κατανοήσει τα παράθυρα του αυτοκινήτου, αλλά ταυτόχρονα δημιουργεί θόρυβο στην εκτίμηση, γύρω από τα σημεία της φωτογραφίας που περιέχουν μικρές λεπτομέρειες.



Σχήμα 5.32: Σύγκριση μεταξύ πηγαίας φωτογραφίας (πάνω) και εκτιμήσεων βάθους μοντέλων HITNet (μεσαία), Fast-ACVNet (κάτω) δίνοντας έμφαση σε συγκεκριμένα σημεία.

Στο Σχήμα 5.32 γίνεται η σημείωση μεταξύ των κύριων σημείων που πιστεύω πως οι εκτιμήσεις του HITNet αντιπροσωπεύουν περισσότερο τις αληθινές τιμές σε σχέση με τις εκτιμήσεις του Fast-ACVNet. Μέσα σε αυτά είναι οι προαναφερθείσες λεπτομέρειες, οι εκτιμήσεις του Fast-ACVNet τείνουν να είναι πιο θορυβώδεις, κάτι που είναι εμφανές και σε άλλα σημεία όπως τα παράθυρα του κτηρίου, που το μοντέλο προσπαθεί να τα αναγνωρίσει σαν συμπαγή αδιαφανή αντικείμενα. Με εξαίρεση αυτά τα χαρακτηριστικά, και οι δύο εκτιμήσεις τείνουν να αντιπροσωπεύουν τις πραγματικές

τιμές, καθώς αναγνωρίζουν τα κύρια χαρακτηριστικά του χώρου, όπως τον δρόμο, τα κτήρια, τα αντικείμενα και τον ορίζοντα.



Σχήμα 5.33: Σύγκριση μεταξύ πηγαίων φωτογραφιών από το KITTI 2015, και οι αντίστοιχες εκτιμήσεις των μοντέλων UnOS (b,c), HITNet (d) και Fast-ACVNet (e).

Στο σχήμα 5.33 παρουσιάζονται συνολικά 6 φωτογραφίες και οι αντίστοιχες εκτιμήσεις τους από τα μοντέλα UnOS, HITNet και Fast-ACVNet. Το Σχήμα χωρίζεται σε 5 στήλες και 6 γραμμές. Κάθε γραμμή αναφέρεται σε διαφορετική φωτογραφία και διαχωρίζεται με λατινική αρίθμηση. Θα αναφερόμαστε σε κάθε φωτογραφία ανάλογα με τη λατινική αρίθμησή της. Επίσης, όπως και στην προηγούμενη σύγκριση, τα μοντέλα είναι διαχωρισμένα με αγγλικά γράμματα. Συγκεκριμένα, έχουμε εκτιμήσεις του UnOS για 50 και 75 χιλιάδες εποχές, ενώ υπάρχουν και οι εκτιμήσεις του HITNet και Fast-ACVNet. Όταν θα αναφερόμαστε σε συγκεκριμένη εκτίμηση, θα αναφέρουμε τη φωτογραφία ανάλογα με την αρίθμηση της γραμμής σε συνδυασμό με το όνομα του μοντέλου.

Οι φωτογραφίες που χρησιμοποιούνται στο Σχήμα 5.33, είναι χωρισμένες σε δύο κατηγορίες. Οι κατηγορίες είναι «Δρόμος εκτός κατοικημένης» για τις φωτογραφίες στις γραμμές i, ii και iii, ενώ οι φωτογραφίες στις γραμμές iv, v και vi κατηγοριοποιούνται ως «Μεγάλη πυκνότητα πράσινου». Αυτή η κατηγοριοποίηση έγινε διότι στις πρώτες φωτογραφίες φαίνονται ξεκάθαρα να είναι στιγμιότυπα από οδήγηση σε δρόμους ταχείας κυκλοφορίας ή δρόμος εκτός κατοικημένης περιοχής, ενώ οι τελευταίες φωτογραφίες περιέχουν πολλά δέντρα, θάμνους αλλά και γρασίδι.

Ξεκινώντας με ένα γνώριμο παράδειγμα, στη φωτογραφία στη θέση iii, μπορούμε να παρατηρήσουμε και να κατανοήσουμε τη γεωμετρία, αλλά και το βάθος εύκολα από όλες τις εκτιμήσεις των μοντέλων. Όπως και στα προηγούμενα παραδείγματα, το UnOS αδυνατεί να διαχωρίσει τις λεπτομέρειες που φαίνονται στο πλαίσιο, χωρίς όμως αυτό να σημαίνει ότι η εκτίμηση δεν προσεγγίζει την πραγματικότητα. Πράγματι, ενώ μπορεί να παρατηρηθεί μεγάλη ένταση θορύβου, αν προσπαθήσει κάποιος να αναγνωρίσει τις τιμές της εκτίμησης ως τιμές βάθους και όχι ως αναγνώριση αντικειμένων στον χώρο, είναι εύκολο να καταλάβει πως σε εκείνο το σημείο της κάμερας βρίσκεται ένα αντικείμενο που έχει μικρή σχετική απόσταση ως προς την κάμερα. Η συγκεκριμένη λεπτομέρεια φαίνεται με μεγαλύτερη ευκρίνεια στο HITNet και το Fast-ACVNet χωρίς φυσικά να τα καθιστά αλάνθαστα ή με έλλειψη θορύβου, καθώς και τα δύο μοντέλα αδυνατούν να αναγνωρίσουν και αντίστοιχα να εκτιμήσουν το βάθος της πλευράς του δρόμου αριστερά του αυτοκινήτου. Στα υπόλοιπα σημεία της τρίτης φωτογραφίας, όλα τα μοντέλα καταφέρνουν και προσεγγίζουν μία καλή εκτίμηση, όλα με τον δικό τους τρόπο.

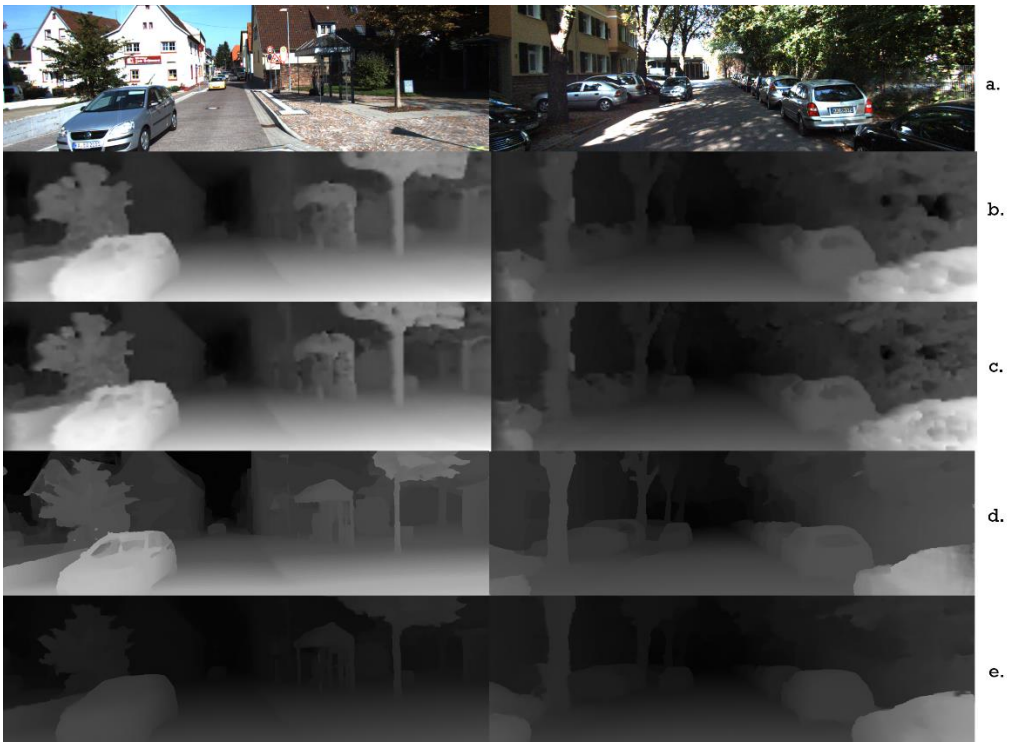
Αντίστοιχες εκτιμήσεις μπορούν να παρατηρηθούν και στις φωτογραφίες της σειράς i και ii του Σχήματος 5.33. Οι εκτιμήσεις του UnOS αδυνατούν να αναγνωρίσουν τις λεπτομέρειες που περιέχονται μέσα στη φωτογραφία, χωρίς όμως να καθίσταται το εκτιμώμενο βάθος τους τελείως αποσυνδεδεμένο από τις πραγματικές τιμές. Το μεγαλύτερο πρόβλημα του UnOS έχει να κάνει με το εκτιμώμενο βάθος του ορίζοντα. Και στις δύο φωτογραφίες υπάρχει ξεκάθαρος διαχωρισμός του ορίζοντα σε σχέση με το υπόλοιπο πλαίσιο της φωτογραφίας. Μέσα στο υπόλοιπο πλαίσιο εμπεριέχονται επίσης δέντρα και αντικείμενα σε σχετικά «μεσαία» απόσταση από την κάμερα. Το μοντέλο, όμως, αναγνωρίζει το βάθος τους λανθασμένα, ταυτίζοντας ή μπερδεύοντάς τα με άλλα σημεία της φωτογραφίας.

Αντιθέτως, στις εκτιμήσεις του HITNet και του Fast-ACVNet, αυτό το φαινόμενο τείνει να εκλείπει. Και τα δύο μοντέλα καταφέρνουν και κάνουν έναν ξεκάθαρο διαχωρισμό του Ορίζοντα σε σχέση με τα υπόλοιπα σημεία της φωτογραφίας. Επιπλέον, και τα δύο μοντέλα αναγνωρίζουν τις λεπτομέρειες της κάθε φωτογραφίας, προσεγγιστικά το καθένα με τον δικό του τρόπο, χωρίς να μπορεί να βγει συνολικό συμπέρασμα για το ποια εκτίμηση είναι ανώτερη. Το μεγαλύτερο πρόβλημα όλων των μοντέλων, όμως, φαίνεται στον διαχωρισμό του «πράσινου» (δέντρα, βλάστηση κτλ.), κάτι που φαίνεται και από τις φωτογραφίες στις θέσεις iv, v και vi.

Οι φωτογραφίες στις θέσεις iv, v και vi του Σχήματος 5.33 απεικονίζουν φωτογραφίες κατηγοριοποιημένες σε «μεγάλη πυκνότητα πράσινου». Η συγκεκριμένη κατηγορία δεν έχει παρουσιαστεί στις προηγούμενες εκτιμήσεις αν και τα αποτελέσματα έχουν μεγάλο ενδιαφέρον. Όλα τα μοντέλα δυσκολεύονται να ερμηνεύσουν τα δέντρα και την περίπλοκη γεωμετρία τους. Αυτό έχει ως αποτέλεσμα όλα τα μοντέλα να εκτιμούν λανθασμένα το βάθος, το καθένα σε διαφορετικό βαθμό. Στη φωτογραφία της σειράς iv, όλα τα μοντέλα δυσκολεύονται να εκτιμήσουν το βάθος της αριστερής πλευράς της φωτογραφίας που περιέχει βλάστηση με περίπλοκη γεωμετρία. Ο μικρότερος θόρυβος σε αυτό το κομμάτι της φωτογραφίας φαίνεται να υπάρχει στην εκτίμηση του Fast-ACVNet, το οποίο προσεγγίζει καλύτερα το βάθος από όλα τα υπόλοιπα μοντέλα, χωρίς όμως αυτό να καθιστά την εκτίμηση αλάνθαστη ή βέλτιστη.

Στη φωτογραφία στη σειρά v, τα αποτελέσματα φαίνονται να είναι καλύτερα για όλες τις εκτιμήσεις, ενώ οι εκτιμήσεις του UnOS προσφέρουν τη χειρότερη βελτίωση. Και οι δύο εκτιμήσεις προσεγγίζουν την κύρια ιδέα όσον αφορά το βάθος που μπορεί να κατανοηθεί από το πλαίσιο της φωτογραφίας. Το κύριο πρόβλημα πάλι προέρχεται από τις λεπτομέρειες που υπάρχουν στις άκρες των εκτιμήσεων, στα δεξιά και στα αριστερά. Αντίθετα, στο HITNet, το εκτιμώμενο βάθος των συγκεκριμένων σημείων δίνει μία ξεκάθαρη εικόνα του πλαισίου της φωτογραφίας. Οι λεπτομέρειες ξεχωρίζουν ξεκάθαρα με τα υπόλοιπα σημεία της φωτογραφίας, ενώ το βάθος γίνεται αντιληπτό σε μεγάλο βαθμό. Αντίστοιχα αποτελέσματα με το HITNet έχει και το Fast-ACVNet, με τη διαφορά όμως ότι δημιουργεί ακόμα λιγότερο θόρυβο στην εκτίμησή του, καθώς και διαφορές όσον αφορά την ερμηνεία διαφόρων αντικειμένων που φαίνονται στο πλαίσιο, χωρίς όμως αυτό να σημαίνει ότι η εκτίμηση είναι είτε ανώτερη είτε κατώτερη του HITNet.

Στα σχήματα 5.34 και 5.35 παρουσιάζονται πηγαίες φωτογραφίες του KITTI 2015, κατηγοριοποιημένες ως «Δρόμος μεγάλης πυκνότητας αντικειμένων», με τις αντίστοιχες εκτιμήσεις των μοντέλων UnOS, HITNet και Fast-ACVNet. Η γενική εικόνα δεν διαφέρει πολύ από τις προηγούμενες εκτιμήσεις των μοντέλων. Οι εκτιμήσεις του UnOS χάνουν πληροφορία πάνω σε λεπτομέρειες, αλλά αναγνωρίζουν τη γενική γεωμετρία, ενώ το HITNet και το Fast-ACVNet έχουν τις πιο ρεαλιστικές εκτιμήσεις με διαφορετικό θόρυβο, τοποθετημένο σε διαφορετικά σημεία των εκτιμήσεων.



Σχήμα 5.34: Σύγκριση πηγαίων φωτογραφιών από το dataset KITTI 2015 (a.) και οι αντίστοιχες εκτιμήσεις των μοντέλων UnOS (b,c), HITNet (d) και Fast-ACVNet (e) (1) – Δρόμος μεγάλης πυκνότητας αντικειμένων.

Σχήμα 5.35: Σύγκριση πηγαίων φωτογραφιών από το dataset KITTI 2015 (a.) και οι αντίστοιχες εκτιμήσεις των μοντέλων UnOS (b,c), HITNet (d) και Fast-ACVNet (e) (2) – Δρόμος μεγάλης πυκνότητας αντικειμένων.

Σχήμα 5.36: Σύγκριση μεταξύ συγκεκριμένων λεπτομερειών σε πηγαία φωτογραφία του KITTI 2015 (πάνω), και στις εκτιμήσεις του βάθους από το HITNet (μεσαία) και το Fast-ACVNet (κάτω).

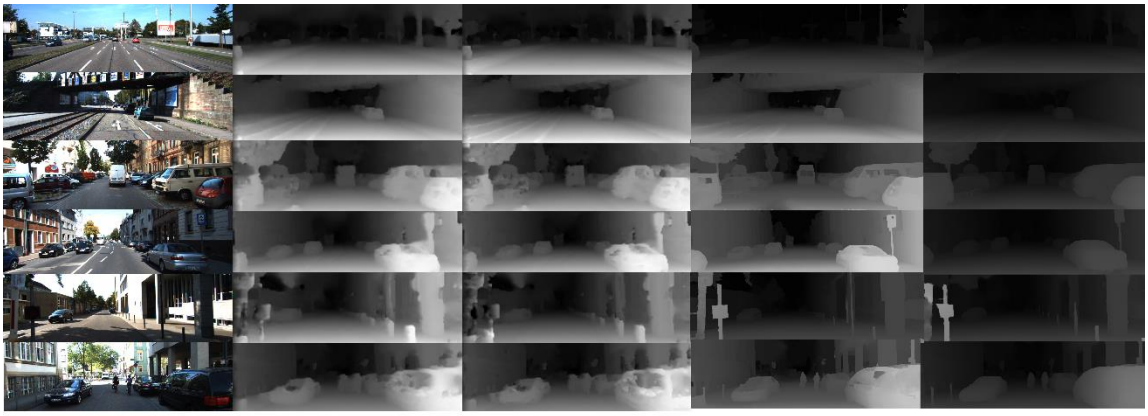
Στο Σχήμα 5.36 μπορεί να παρατηρηθεί καλύτερα αυτό που αναφέρθηκε στην προηγούμενη παράγραφο. Σημειωμένα με κόκκινο και πράσινο πλαίσιο φαίνονται τα σημεία που πιστεύω πως το Fast-ACVNet εκτιμά καλύτερα. Το HITNet, όπως φαίνεται στο σχήμα αλλά και στα προηγούμενα παραδείγματα, αναγνωρίζει καλύτερα τα παράθυρα, αλλά και την παραλλαγή του βάθους που αντιπροσωπεύουν. Στο συγκεκριμένο παράδειγμα, όμως, τα παράθυρα λειτουργούν και σαν καθρέπτες, δημιουργώντας έτσι ένα εσφαλμένο αναγνωρισμένο βάθος. Αν θεωρήσουμε πως μέσα από ένα παράθυρο μπορεί να φανεί ο χώρος που βρίσκεται πίσω από αυτό, άρα και η διαφοροποίηση στο βάθος, όταν δεν κατοπτρίζει το περιβάλλον και είναι διαφανή, τότε θα πρέπει το εκτιμώμενο βάθος να είναι αυτό του χώρου πίσω από το παράθυρο. Όταν, όμως, κάτι λειτουργεί σαν καθρέπτης, τότε αναγνωρίζοντάς το σαν διαφανές αντικείμενο και προσπαθώντας να εκτιμήσει τα αντικείμενα με διαφορετικό βάθος, δημιουργεί λανθασμένα αποτελέσματα.

Έχοντας αυτό υπόψιν, ουσιαστικά, δημιουργείται μία σύγκρουση στον τρόπο διαχείρισης των παραθύρων των δύο μοντέλων. Το Fast-ACVNet το αναγνωρίζει ως ένα συμπαγές αντικείμενο με υπόσταση, ενώ το HITNet αναγνωρίζει τον διαφορετικό βαθμό ανισότητας και δημιουργεί μια διαφορετική εκτίμηση.

Στο Σχήμα 5.35, μπορούμε να παρατηρήσουμε και άλλα παραδείγματα της κατηγορίας «Δρόμος μεγάλης πυκνότητας αντικειμένων» μαζί με τις εκτιμήσεις. Το σημείο που είναι άξιο αναφοράς ως προς

τις συγκεκριμένες εκτιμήσεις είναι η σύγκριση μεταξύ HITNet και Fast-ACVNet. Υπάρχουν κάποια κύρια σημεία όσον αφορά τις λεπτομέρειες που το μοντέλο HITNet καταφέρνει και εκτιμά καλύτερα συγκριτικά με το Fast-ACVNet. Αυτά τα σημεία φαίνονται και στο Σχήμα 5.37, όπου είναι κυκλωμένα με κόκκινο χρώμα. Φαίνεται πως το HITNet σε αυτά τα σημεία μπορεί να αναγνωρίσει καλύτερα τις λεπτομέρειες, αλλά και να διαχωρίσει τα συγκεκριμένα αντικείμενα από το γύρω περιβάλλον τους. Αυτό, όμως, δεν σημαίνει πως το εκτιμώμενο βάθος του Fast-ACVNet είναι λανθασμένο. Και τα δύο προσεγγίζουν σε καλό βαθμό την πραγματικότητα.

Σχήμα 5.37: Σύγκριση μεταξύ των εκτιμήσεων του HITNet (πάνω) και του Fast-ACVNet (κάτω) βάσει της δεύτερης φωτογραφίας από το Σχήμα 5.35.



Σχήμα 5.38: Σύγκριση μεταξύ πηγαίων φωτογραφιών του KITTI 2015 (a.) και των αντίστοιχων εκτιμήσεων των υπολοίπων μοντέλων για διάφορες φωτογραφίες διαφορετικών κατηγοριών. Οι εκτιμήσεις είναι των μοντέλων UnOS (b,c), του HITNet (d) και του Fast-ACVNet (e).

Στο Σχήμα 5.38 παρουσιάζονται οι τελευταίες φωτογραφίες με τις αντίστοιχες εκτιμήσεις. Οι φωτογραφίες αποτελούνται από τις τελευταίες τρεις κατηγορίες: «δρόμος μικρής πυκνότητας αντικειμένων» (1η), «μη ομαλός δρόμος» (2η) και «υψηλά κτήρια» (3η – 6η). Οι συγκεκριμένες εκτιμήσεις δεν παρουσιάζουν κάτι που δεν έχει ήδη παρουσιαστεί στις προηγούμενες συγκρίσεις. Το μοντέλο UnOS δυσκολεύεται να διαχωρίσει τις λεπτομέρειες, αλλά κατανοεί τη γενική γεωμετρία του χώρου. Αυτό έχει ως αποτέλεσμα να βγάζει ικανοποιητικές εκτιμήσεις σε φωτογραφίες με λίγες λεπτομέρειες και αντικείμενα εντός των πλαισίων τους. Από την άλλη μεριά, και οι εκτιμήσεις του

HITNet, αλλά και οι εκτιμήσεις του Fast-ACVNet είναι εξίσου καλές, ενώ κάποιες φορές το HITNet παράγει καλύτερες εκτιμήσεις και σε άλλες περιπτώσεις το Fast-ACVNet υπερτερεί σε σχέση με το HITNet, αν και η διαφορά πάντα είναι μικρή και βασίζεται σε λεπτομέρειες.

Κεφάλαιο 6ο: Συμπεράσματα, προτάσεις και περιορισμοί.

Μέσα στην εργασία συνδυάστηκαν πολλές τεχνολογίες, επιστημονικά πεδία, αλλά και μέθοδοι. Ενώ το πρόβλημα που επιλύεται είναι πολύ σημαντικό και θα καταφέρει να υποστηρίξει πολλές εφαρμογές στο μέλλον, κατά την υλοποίηση και διερεύνησή του, ερχόμαστε αντιμέτωποι με πολλούς περιορισμούς. Οι περιορισμοί μπορούν να χωριστούν σε τέσσερις κατηγορίες:

- **Κόστος:** Η δημιουργία ενός SOTA μηχανισμού που μπορεί να εκτιμάει το βάθος με δύο κάμερες έχει τεράστιο κόστος. Μέσα στο κόστος συμπεριλαμβάνεται το κόστος των καμερών που θα χρησιμοποιηθούν για τη δημιουργία του συνόλου δεδομένου ή ακόμα και για τη χρήση του. Επιπλέον κόστος προστίθεται για την εκπαίδευση των μοντέλων μηχανικής μάθησης, καθώς τα περισσότερα μοντέλα ήταν εκπαιδευμένα σε μηχανήματα με ακριβό εξοπλισμό.
- **Χρόνος:** Κατά τη διάρκεια των υλοποιήσεων, για να μπορέσουν να εκπαιδευτούν και να χρησιμοποιηθούν και τα τρία μοντέλα που παρουσιάστηκαν, χρειάστηκε μια χρονική διάρκεια 6 μηνών για μέση διάρκεια εβδομαδιαίας απασχόλησης 20 ωρών. Μέσα σε αυτόν τον χρόνο συμπεριλαμβάνεται η διαχείριση των πακέτων, η αποσφαλμάτωση κώδικα, η επιδιόρθωση σπασμένων σημείων κώδικα, αλλά και ο εκσυγχρονισμός τους. Επιπλέον, υπάρχει μεγάλη απαίτηση σε χρόνο για την εκπαίδευση των μοντέλων, καθώς συγκεκριμένα μοντέλα χρειάζονταν μέχρι και εβδομάδες για να εκπαιδευτούν σε σημείο που να μπορούν να κάνουν σωστές εκτιμήσεις.
- **Γνώση:** Για να επιτευχθούν καλές εκτιμήσεις βάθους με τη χρήση μηχανικής μάθησης, απαιτείται από τον χρήση να έχει γνώση σε διάφορες τεχνολογίες. Πέρα από τις γνώσεις σε τεχνολογικό επίπεδο (Python, NumPy, TensorFlow, PyTorch), απαιτείται και γνώση σε πολλά επιστημονικά πεδία. Απαιτείται να μπορεί ο χρήστης να κατανοήσει το πεδίο της Μηχανικής Μάθησης σε εξειδικευμένο βαθμό, ενώ ταυτόχρονα θα πρέπει να έχει τις απαραίτητες μαθηματικές γνώσεις για να υποστηρίξει τα υποστηρικτικά κομμάτια. Μέσα στα υποστηρικτικά κομμάτια εμπεριέχεται η μετατροπή της ανισότητας σε βάθος, η δυνατότητα βαθμονόμησης των καμερών, κ.ά.
- **Τεχνολογικοί περιορισμοί:** Ένα πρόβλημα που κλήθηκα να αντιμετωπίσω κατά τη διάρκεια των υλοποιήσεων ήταν οι τεχνολογικοί περιορισμοί. Επειδή τα μοντέλα είναι υλοποιημένα σε μία γλώσσα που σταμάτησε να υποστηρίζεται εδώ και χρόνια, σε συνδυασμό με την έλλειψη ξεκάθαρων πληροφοριών όσον αφορά τα σφάλματα δημιουργήθηκαν περιορισμοί από πλευρά των τεχνολογιών.

Έχοντας, όμως, έτοιμα τα μοντέλα να εκτιμήσουν το βάθος, μας δίνεται η δυνατότητα να χρησιμοποιηθούν για να βελτιώσουν πολλές εργασίες. Κατόπιν σωστής προετοιμασίας και βαθμονόμησης καμερών, η συγκεκριμένη εργασία μπορεί να βρει εφαρμογή σε «αυτόνομη οδήγηση», καθώς η πληροφορία του βάθους είναι απαραίτητη για να παίρνει αποφάσεις ένα «έξυπνο» αυτοκίνητο. Πέρα από την οδήγηση, η πληροφορία του βάθους μπορεί να βρει εφαρμογές στη ρομποτική, την Ιατρική, την ανακατασκευή 3Δ χώρων, κ.ά.

Όσον αφορά τη βελτίωση των μεθόδων, για να μπορέσουν να αποτιμηθούν καλύτερα, αλλά και ταυτόχρονα να μπορέσουν να αυξήσουν την αποτελεσματικότητά τους, απαιτούνται κάποια βήματα.

1. **Δημιουργία πακέτου αποτίμησης**: Πρέπει να δημιουργηθεί ένα σύνολο αποτίμησης που να εκτιμά με παρόμοια αποτελεσματικότητα που είχαν και τα παρουσιαζόμενα σύνολα. Μέσα από αυτό το σύνολο, θα δημιουργηθούν οι απαραίτητες γνώσεις και πληροφορίες όσον αφορά τη βαθμονόμηση των καμερών, καταφέρνοντας έτσι να χρησιμοποιήσουν τα μοντέλα σε πραγματικές συνθήκες.

2. **Αγορά εξοπλισμού**: Με το κατάλληλο εξοπλισμό, συμπεριλαμβανομένων καμερών, μηχανημάτων και αισθητήρων, θα μπορεί να αυξηθεί η ποιότητα της πηγαίας πληροφορίας. Με αυτόν τον τρόπο, τα μοντέλα θα έχουν περισσότερη δύναμη στην κατανόηση του περιβάλλοντος, αλλά και του χώρου που απεικονίζεται στις φωτογραφίες.

3. **Έρευνα στα συστατικά των μοντέλων**: Όπως παρατηρήθηκε και στα αναφερόμενα μοντέλα, όλα προσπαθούσαν να προτείνουν καινούριες τακτικές που τα διαφοροποιούσαν από τις ήδη προ-υπάρχουσες. Η αρχιτεκτονική τους όμως δεν διέφερε τελείως σε σύγκριση με τις προ-υπάρχουσες. Με αυτόν τον τρόπο, μπορούν να επιτευχθούν είτε καλύτερα, είτε πιο άμεσα αποτελέσματα, μόνο με την αναγνώριση μοτίβων των φωτογραφιών.

Κοιτώντας τις μεθόδους που παρουσιάστηκαν, θα χωρίσουμε τα συμπεράσματα σε δύο κατηγορίες. Αρχικά, πρέπει να μιλήσουμε για τη μέθοδο του “Stereo Depth Estimation”. Στο συγκεκριμένο κεφάλαιο φαίνεται έντονα η αναφορά στο κόστος, κάτι που δεν μπορεί να εκλείπει όταν ασχολούμαστε με την εκτίμηση βάθους από δύο κάμερες. Η συγκεκριμένη μέθοδος, αν και είναι βασισμένη σε πολύ δυνατά επιστημονικά πεδία και σε σημαντικές παρατηρήσεις, έρχεται αντιμέτωπη με τους προαναφερθέντες περιορισμούς. Αυτό έχει ως αποτέλεσμα να καθιστά ακριβό το κόστος για τη δημιουργία μίας εκτίμησης, η οποία πρέπει να είναι σε ελεγμένο περιβάλλον. Αντίθετα, υπάρχουν άλλες μέθοδοι και τακτικές που, ενώ έχουν μικρότερη αποτελεσματικότητα, έχουν ξεκάθαρα μικρότερο κόστος (π.χ. Monocular Depth Estimation).

Αν όμως υπάρχουν οι ιδανικές συνθήκες για να χρησιμοποιηθούν οι αλγόριθμοι “Stereo Depth Estimation”, τότε οι προτεινόμενες μέθοδοι έχουν πολλές δυνατότητες και μπορούν να ενσωματωθούν για να υποστηρίξουν διάφορα συστήματα. Και τα τρία μοντέλα που χρησιμοποιήθηκαν αναγνωρίζουν έως ένα βαθμό το βάθος του περιβάλλοντος και καταλαβαίνουν τη γεωμετρία του χώρου. Ωστόσο, τα μοντέλα που πρωταγωνιστούν είναι το HITNet και το Fast-ACVNet. Κοιτώντας τις μετρικές, μπορούμε να συμπεράνουμε πως και τα δύο μοντέλα βγάζουν αντίστοιχα αποτελέσματα, ενώ η κύρια διαφορά των μοντέλων είναι ο χώρος και ο τρόπος που δουλεύουν. Ενώ το ACVNet φαίνεται να υπερτερεί κατά ένα μικρό βαθμό έναντι του HITNet, το HITNet είναι ένα μοντέλο πολύ πιο ευέλικτο, μικρότερο και γρηγορότερο σε σχέση με το ACVNet. Όμως, η υλοποίηση του Fast-ACVNet βάσει των μετρικών, φαίνεται να ακολουθεί το HITNet με ελάχιστα χειρότερα χαρακτηριστικά.

Συμπερασματικά, ανάλογα με την εφαρμογή και τον χρόνο που δίνεται σε ένα σύστημα να πάρει απόφαση, θα προτεινόταν και διαφορετικό μοντέλο. Σε σύστημα που απαιτείται γρήγορη απόκριση και μεγάλη ελαστικότητα, το HITNet είναι ιδανική επιλογή, ενώ σε περιπτώσεις που οι εκτιμήσεις δεν απαιτούνται σε πραγματικό χρόνο, το ACVNet φαίνεται σαν την καλύτερη επιλογή.

ΒΙΒΛΙΟΓΡΑΦΙΑ

[1] B. Raj, "Depth Estimation: A comprehensive review of techniques used to estimate depth using Machine Learning and classical methods.," *Medium*, 2019.

[2] D. Mwiti, "Research Guide for Depth Estimation with Deep Learning - KDnuggets," KDnuggets, 12 11 2019. [Online]. Available: https://www.kdnuggets.com/2019/11/research-guide-depth-estimation-deep-learning.html. [Accessed 12 June 2023].

[3] M. O. Y. Z. T. O. Junjie Hu, "Revisiting Single Image Depth Estimation: Toward Higher Resolution Maps with Accurate Object Boundaries," 2018.

[4] S. K. K. S. Kihong Park, "High-precision Depth Estimation with the 3D LiDAR and Stereo Fusion," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, 2018.

[5] C. C. Yuhang Zheng, "Monocular Depth Estimation with Multiscale Feature Fusion Networks," in *2022 2nd International Conference on Algorithms, High Performance Computing and Artificial Intelligence (AHPCAI)*, Guangzhou, China, 2022.

[6] Synopsys, "What is an Autonomous Car? – How Self-Driving Cars Work | Synopsys," Synopsys, [Online]. Available: https://www.synopsys.com/automotive/what-is-autonomous-car.html. [Accessed 29 May 2023].

[7] H. I. &. T. Hamamoto, "Depth Estimation Using Variant of Depth of Field by Horizontal Planes of Sharp Focus," in *FIRA RoboWorld Congress*, 2009.

[8] X. C. &. X. Q. Guofa Li, "Depth Estimation Based on Monocular Camera Sensors in Autonomous Vehicles: A Self-supervised Learning Approach," *Automotive Innovation*, 2023.

[9] S.-H. B. &. M. H. Kim, "Stereo Fusion Using a Refractive Medium on a Binocular Base," 2014.

[10] Y. N. S.-p. L. J. Z. Xiong-Zhi Wang, "Deep Convolutional Network for Stereo Depth Mapping in Binocular Endoscopy," 2020.

[11] M. O. Y. Z. T. O. Junjie Hu, "Revisiting Single Image Depth Estimation," 2018.

[12] O. M. A. M. F. G. B. Clement Godard, "Digging Into Self-Supervised Monocular Depth Estimation," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019.

[13] P. L. R. U. Andreas Geiger, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, USA, 2012.

[14] W. K. P. A. D. J. S. C. J. K. Doyeon Kim, "Global-Local Path Networks for Monocular Depth Estimation," 2022.

[15] D. H. P. K. R. F. Nathan Silberman, "Indoor segmentation and support inference from RGBD images," in *ECCV 2012 - 12th European Conference on Computer Vision*, Florence, 2012.

[16] N. a. E. a. P. a. P. a. D. a. A. a. T.Brox, "A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[17] J. L. S. S. G. T. S. K. S. M. P. A. G. T. Schöps, "A Multi-View Stereo Benchmark with High-Resolution Images and Multi-Camera Videos," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[18] Y.-S. C. Jia-Ren Chang, "Pyramid Stereo Matching Network," 2018.

[19] Z. L. G. H. B. H. W. L. v. d. M. M. C. K. Q. W. Yan Wang, "Anytime Stereo Image Depth Estimation on Mobile Devices," 2018.

[20] M. Rouse, "What is Frames Per Second (FPS)? - Definition from Techopedia," Techopedia Inc., 2022 March 14. [Online]. Available: https://www.techopedia.com/definition/7297/frames-per-second-fps. [Accessed 30 June 2023].

[21] IBM, "What is artificial intelligence (AI)? | IBM," IBM, [Online]. Available: https://www.ibm.com/topics/artificial-intelligence. [Accessed 03 June 2023].

[22] Κ. Δ. &. Δ. Μπότσης, ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ, ΚΛΕΙΔΑΡΙΘΜΟΣ, 2019.

[23] O. Technologies, "What Is Model Training? | Oden Technologies," Oden Technologies, [Online]. Available: https://oden.io/glossary/model-training/. [Accessed 11 June 2023].

[24] K. Barkved, "The Difference Between Training Data vs. Test Data in Machine Learning," obiously.ai, 11 February 2022. [Online]. Available: https://www.obviously.ai/post/the-difference-between-training-data-vs-test-data-in-machine-learning. [Accessed 05 June 2023].

[25] Δ. Μ. Κωνσταντίνος Διαμαντάρας, "ΣΤΟΧΟΣ ΚΑΙ ΑΞΙΑ ΤΗΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ," in *ΜΗΧΑΝΙΚΗ ΜΑΘΗΣΗ*, ΚΛΕΙΔΑΡΙΘΜΟΣ, 2019, p. 19.

[26] geeksforgeeks, "ML | Underfitting and Overfitting - GeeksforGeeks," geeksforgeeks, [Online]. Available: https://www.geeksforgeeks.org/underfitting-and-overfitting-in-machine-learning/. [Accessed 18 June 2023].

[27] A. Bhande, "What is underfitting and overfitting in machine learning and how to deal with it.," *GreyAtom*, 2018.

[28] A. L. Chandra, "McCulloch-Pitts Neuron — Mankind's First Mathematical Model Of A Biological Neuron," *Towards Data Science*, 24 July 2018.

[29] geeksforgeeks, "Implementing Models of Artificial Neural Network - GeeksforGeeks," geeksforgeeks, [Online]. Available: https://www.geeksforgeeks.org/implementing-models-of-artificial-neural-network/. [Accessed 16 June 2023].

[30] ibm, "What is deep learning? | IBM," ibm, [Online]. Available: https://www.ibm.com/topics/deep-learning. [Accessed 04 June 2023].

[31] Javatpoint, "Perceptron in Machine Learning - Javatpoin," Javatpoint, [Online]. Available: https://www.javatpoint.com/perceptron-in-machine-learning. [Accessed 06 June 2023].

[32] DeepAI, "Perceptron Definition | DeepAI," DeepAI, [Online]. Available: https://deepai.org/machine-learning-glossary-and-terms/perceptron. [Accessed 03 June 2023].

[33] R. Cassani, "rcassani/mlp-example: Code for a simple MLP (Multi-Layer Perceptron)," [Online]. Available: https://github.com/rcassani/mlp-example. [Accessed 11 June 2023].

[34] Κ. Δ. &. Δ. Μπότσης, "Συνελεκτικά Νευρωνικά Δίκτυα," in *Μηχανική Μάθηση*, ΚΛΕΙΔΑΡΙΘΜΟΣ, 2019, p. 276.

[35] A. Kumar, "Different Types of CNN Architectures Explained: Examples," *Vitaflux*, 2023.

[36] R. &. K. D. R. &. G. P. &. S. J. Rakshit, "Cross-resolution face identification using deep-convolutional neural network.," 2021.

[37] L. B. Y. B. P. H. Yann LeCun, "GradientBased Learning Applied to Document," 1998.

[38] A. T. Dang, "Top 10 CNN Architectures Every Machine Learning Engineer Should Know," *Towards Data Science*, 2021.

[39] geeksforgeeks, "Residual Networks (ResNet) – Deep Learning," geeksforgeeks, [Online]. Available: https://www.geeksforgeeks.org/residual-networks-resnet-deep-learning/.

[40] H. Gao, "The Efficiency of Densnet," *Medium*, 2017.

[41] R. Pramoditha, "11 Dimensionality reduction techniques you should know in 2021," *Towards Data Science*, 2021.

[42] S. Chaudhary, "Understanding Market Basket Analysis in Data Mining," Turing, [Online]. Available: https://www.turing.com/kb/market-basket-analysis. [Accessed 28 June 2023].

[43] javatpoint, "Apriori Algorithm - Javatpoint," javatpoint, [Online]. Available: https://www.javatpoint.com/apriori-algorithm. [Accessed 30 June 2023].

[44] A. Dertat, "Applied Deep Learning - Part 3: Autoencoders," *Towards Data Science*, 2017.

[45] IBM, "What is unsupervised learnig? | IBM," IBM, [Online]. Available: https://www.ibm.com/topics/unsupervised-learning. [Accessed 04 June 2023].

[46] Maher, "6 Significant Computer Vision Problems Solved by ML," *Heartbeat*, 2020.

[47] A. Acharya, "Guide to Image Segmentation in Computer Vision: Best Practices," *Encord Blog*, 2022.

[48] Towards AI, "Semantic Segmentaiton: A Complete Guide," *Towards AI*, 2021.

[49] S. Du, "Understanding Optical Flow & RAFT," *Towards Data Science*, 2020.

[50] A. Sahu, "Disparity Estimation Using Deep Learning | LearnOpenCV #," LearnOpenCV, 22 February 2022. [Online]. Available: https://learnopencv.com/disparity-estimation-using-deep-learning. [Accessed 12 June 2023].

[51] D. Biswas, "Stereo Camera Calibration and Depth Estimation from Stereo Images," *Medium*, 2021.

[52] C. f. L. N. Y. L. Camille Couprie, "Indoor Semantic Segmentation using depth Information," 2013.

[53] P. L. C. S. R. U. Andreas Geiger, "Vision meets Robotics: The KITTI Dataset," *International Journal of Robotics Research (IJRR)*, 2013.

[54] C. H. A. G. Moritz Menze, "Object Scene Flow," *ISPRS Journal of Photogrammetry and Remote Sensing*, 2018.

[55] C. H. A. G. Moritz Menze, "Joint 3D Estimation of Vehicles and Sceneflow," in *ISPRS Workshop on Image Sequence Analysis (ISA)*, 2015.

[56] A. G. Moritz Menze, "Object Scene Flow for Autonomous Vehicle," 2015.

[57] Blender, "blender.org - Home of the Blender project - Free and Open 3D Creation Software," Blender, [Online]. Available: https://www.blender.org/. [Accessed 15 June 2023].

[58] E. I. P. H. P. F. D. C. A. D. Nikolaus Mayer, "A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation," in *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[59] J. L. S. S. G. T. S. K. S. M. P. A. G. Thomas Schops, "A Multi-View Stereo Benchmark with High-Resolution Images and Multi-Camera Videos," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[60] paperswithcode, "Middlebury Dataset | Papers With Code," paperswithcode, [Online]. Available: https://paperswithcode.com/dataset/middlebury. [Accessed 26 July 2023].

[61] Middlebury, "https://vision.middlebury.edu/stereo/data/," Middlebury, [Online]. Available: https://vision.middlebury.edu/stereo/data/. [Accessed 26 July 2023].

[62] H. H. Y. K. G. K. N. N. X. W. P. W. Daniel Scharstein, "High-Resolution Stereo Datasets with Subpixel-Accurate Ground Truth," in *German Conference on Pattern Recognition (GCPR 2014)*, Münster, Germany, 2014.

[63] S. Harding, "What Is ECC Memory in RAM? A Basic Definition | Tom's Hardware," tom's Hardware, 01 January 2021. [Online]. Available: https://www.tomshardware.com/reviews/ecc-memory-ram-glossary-definition,6013.html. [Accessed 28 July 2023].

[64] Linux.com, "What is Linux? - Linux.com," Linux.com, [Online]. Available: https://www.linux.com/what-is-linux/. [Accessed 19 June 2023].

[65] Python, "Welcome to Python.org," Python, [Online]. Available: https://www.python.org/. [Accessed 30 June 2023].

[66] Anaconda, "About Anaconda," Anaconda, [Online]. Available: https://www.anaconda.com/about-us. [Accessed 25 July 2023].

[67] D. Ellis, "What is Anaconda for Python & Why Should You Learn it," HubSpot, 03 November 2022. [Online]. Available: https://blog.hubspot.com/website/anaconda-python. [Accessed 20 July 2023].

[68] Python, "What is Python? Executive Summary | Python.org," Python, [Online]. Available: https://www.python.org/doc/essays/blurb/. [Accessed 23 July 2023].

[69] N. Kvartalnyi, "Top 23 Applications Made with Python | Inoxoft," Inoxoft, 14 March 2023. [Online]. Available: https://inoxoft.com/blog/top-23-applications-made-with-python/. [Accessed 20 July 2023].

[70] Cobra Programming Language, "The Cobra Programming Language," Cobra Programming Language, [Online]. Available: http://cobra-language.com/.

[71] CoffeeScript, "CoffeeScript," CoffeeScript, [Online]. Available: https://coffeescript.org/. [Accessed 1 Aug 2023].

[72] GO, "The Go Programming Language," GO, [Online]. Available: https://go.dev/.

[73] Future Learn, "What is Python used for? 10 practical Python uses," Future Learn, 9 April 2021. [Online]. Available: https://www.futurelearn.com/info/blog/what-is-python-used-for. [Accessed 23 July 2023].

[74] NumPy, "What is NumPy? - NumPy v1.25 Manual," NumPy, [Online]. Available: https://numpy.org/doc/stable/user/whatisnumpy.html. [Accessed 10 July 2023].

[75] OpenCV, "About - OpenCV," OpenCV, [Online]. Available: https://opencv.org/about/. [Accessed 25 July 2023].

[76] TensorFlow, "TensorFlow," TensorFlow, [Online]. Available: https://www.tensorflow.org/. [Accessed 28 July 2023].

[77] TensorFlow, "Why TensorFlow," TensorFlow, [Online]. Available: https://www.tensorflow.org/about. [Accessed 28 July 2023].

[78] TensorFlow, "Use a GPU | TensorFlow Core," TensorFlow, [Online]. Available: https://www.tensorflow.org/guide/gpu. [Accessed 28 July 2023].

[79] R. Merritt, "What is Accelerated Computing," Nvidia, 1 September 2021. [Online]. Available: https://blogs.nvidia.com/blog/2021/09/01/what-is-accelerated-computing/. [Accessed 28 July 2023].

[80] S. L. Kinza Yasar, "What is PyTorch," Techtarget, [Online]. Available: https://www.techtarget.com/searchenterpriseai/definition/PyTorch. [Accessed 28 July 2023].

[81] J. Terra, "Pytorch Vs Tensorflow Vs Keras: Here are the Difference You Should Know," Simplilearn, 7 July 2023. [Online]. Available: https://www.simplilearn.com/keras-vs-tensorflow-vs-pytorch-article. [Accessed 28 July 2023].

[82] J. Johnson, "TensorFlow vs PyTorch: Choosing Your ML Framework," BMC, 14 February 2022. [Online]. Available: https://www.bmc.com/blogs/tensorflow-vs-keras/. [Accessed 28 July 2023].

[83] F. Oh, "What Is CUDA | NVIDIA Official Blog," Nvidia, 10 September 2012. [Online]. Available: https://blogs.nvidia.com/blog/2012/09/10/what-is-cuda-2/. [Accessed 28 July 2023].

[84] M. Heller, "What is CUDA? Parallel programming for GPUs," *InfoWorld*, 16 September 2022.

[85] Academic Accelerator, "Visual Odometry: The Most Up-to-Date Encyclopedia, News, Review & Research," Academic Accelerator. [Online]. [Accessed 03 Aug 2023].

[86] P. W. Z. Y. C. L. Y. Y. W. X. Yang Wang, "UnOS: Unified Unsupervised Optical-flow and Stereo-depth Estimation by Watching Videos," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019.

[87] X. Y. M.-Y. L. J. K. Deqing Sun, "PWC-Net: CNNs for Optical Flow Using Pyramid, Warping, and Cost Volume," 2017.

[88] A. G. H. Z. V. C. A. A. Hanhan Li, "Unsupervised Monocular Depth Learning in Dynamic Scenes," 2020.

[89] C. H. Y. Z. A. K. S. F. S. B. Vladimir Tankovich, "HITNet: Hierarchical Iterative Tile Refinement Network for Real-time Stereo Matching," 2020.

[90] DSPRELATED.com, "Upsampling and Downsampling | Spectral Audio Signal Processing," DSPRELATED.com, [Online]. Available: https://www.dsprelated.com/freebooks/sasp/Upsampling_Downsampling.html. [Accessed 7 Aug 2023].

[91] P. F. T. B. Olaf Ronneberger, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *n International Conference on Medical image computing and computer-assisted intervention (MICCAI)*, 2015.

[92] geeksforgeeks, "Intuition of Adam Optimizer - GeeksforGeeks," GeeksforGeeks, [Online]. Available: https://www.geeksforgeeks.org/intuition-of-adam-optimizer/. [Accessed 07 Aug 2023].

[93] K. Y. W. Y. X. W. H. L. Xiaoyang Guo, "Group-wise correlation stereo network," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

[94] J. C. P. G. X. Y. Gangwei Xu, "Attention Concatenation Volume for Accurate and Efficient Stereo Matching," 2022.

[95] Y. W. J. C. J. T. Gangwei Xu, "Accurate and Efficient Stereo Matching via Attention Concatenation Volume," 2022.

ΠΑΡΑΡΤΗΜΑ Α: ΚΩΔΙΚΑΣ

Ο κώδικας που χρησιμοποιήθηκε για την υλοποίηση των μοντέλων βρίσκεται εδώ:

UnOS: https://github.com/jimtete/UnDepthflow-thesis-2023/tree/Undepthflow

HITNet: https://github.com/jimtete/thesis-HITNET-Stereo-Depth-estimation

Fast-ACVNet: https://github.com/jimtete/thesis-Fast-ACVNet