



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΑΓΡΟΝΟΜΩΝ ΚΑΙ ΤΟΠΟΓΡΑΦΩΝ ΜΗΧΑΝΙΚΩΝ
ΜΗΧΑΝΙΚΩΝ ΓΕΩΠΛΗΡΟΦΟΡΙΚΗΣ

ΔΙΑΤΜΗΜΑΤΙΚΟ ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΓΕΩΠΛΗΡΟΦΟΡΙΚΗΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΕΦΑΡΜΟΓΗ ΤΕΧΝΙΚΩΝ ΒΑΘΙΑΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ ΓΙΑ ΤΟΝ
ΕΝΤΟΠΙΣΜΟ ΑΝΘΡΩΠΟΓΕΝΩΝ ΚΑΤΑΣΚΕΥΩΝ ΣΕ ΜΗ ΚΑΤΟΙΚΗΜΕΝΕΣ
ΠΕΡΙΟΧΕΣ ΑΠΟ ΔΟΡΥΦΟΡΙΚΕΣ ΕΙΚΟΝΕΣ**

Μάριος Σπανάκης

Επιβλέπων

Νικόλαος Δουλάμης

Καθηγητής ΕΜΠ

Τριμελής επιτροπή

Νικόλαος Δουλάμης

Καθηγητής
Ε.Μ.Π.

Βασίλειος Βεσκούκης

Καθηγητής
Ε.Μ.Π.

Αναστάσιος Δουλάμης
Αναπληρωτής Καθηγητής
Ε.Μ.Π.

Αθήνα, Ιούνιος 2024

Ευχαριστίες

Με την εκπόνηση της παρούσας διπλωματικής εργασίας, ολοκληρώνεται ο μεταπτυχιακός κύκλος σπουδών μου στη Σχολή Αγρονόμων και Τοπογράφων Μηχανικών – Μηχανικών Γεωπληροφορικής του Εθνικού Μετσόβιου Πολυτεχνείου. Ελπίζω, πρωτίστως με τη βοήθεια του Θεού, στην επαγγελματική και ακαδημαϊκή μου εξέλιξη, καθώς ως Αξιωματικός δυνάμεως του Γεωγραφικού σώματος, θεωρώ υποχρέωσή μου την άρτια επαγγελματική μου κατάρτιση.

Ευχαριστώ τη μέλλουσα γυναίκα μου και αξιόλογη συνάδελφο Στυλιανή Ενωτιάδου, τους γονείς μου, Ζαχαρία και Μαργαρίτα Σπανάκη, και τον αδερφό μου Γεώργιο Σπανάκη για τη στήριξη, την καθοδήγηση και την αγάπη που μου έδωσαν και συνεχίζουν να μου δίνουν.

Ευχαριστώ τον αρχικά Διευθυντή μου και σήμερα Υποδιοικητή της Διακλαδική Διοίκηση Ειδικών Επιχειρήσεων όπου υπηρετώ, Ταξίαρχο Θεοφάνη Δημάκο, ο οποίος πίστεψε στις δυνατότητές μου από την πρώτη στιγμή που με γνώρισε, μου δίδαξε ως προϊστάμενος αξιωματικός αρχές διοίκησης και ηγεσίας ενώ ταυτόχρονα με διευκόλυνε να παρακολουθήσω το πρόγραμμα εκπαίδευσης του ΔΠΜΣ Γεωπληροφορικής.

Ευχαριστώ τους συναδέλφους μου Δημήτριο Σακελλαρίου και Εμμανουήλ Τσουμενή, με τους οποίους αντιμετωπίσαμε τις ίδιες δυσκολίες από τη Σχολή Ευελπίδων μέχρι και σήμερα και από τη μέρα που παρουσιαστήκαμε στη ΓΥΣ ήμασταν πάντα ο ένας για τον άλλο.

Ευχαριστώ τον Καθηγητή κ. Νικόλαο Δουλάμη, καθώς ως καθηγητής μου στη Σχολή Τοπογραφίας της ΓΥΣ αλλά και στο ΕΜΠ, μου έδωσε πολλά εφόδια για την κατάρτισή μου ως μηχανικός. Περισσότερο όμως τον ευχαριστώ γιατί μου εμφύσησε ένα διαφορετικό τρόπο αντίληψης και αντιμετώπισης καταστάσεων και γιατί έδειξε εμπιστοσύνη στις δυνατότητές μου.

Τέλος, θα ήθελα να ευχαριστήσω τον υποψήφιο Δρ. Αναστάσιο Τέμενο για την άψογη συνεργασία και τη συνεχή καθοδήγηση στην εκπόνηση αυτής της εργασίας.

Περίληψη

Οι αυτοματοποιημένες διαδικασίες εντοπισμού αντικείμενων από εικόνες μέσω μεθόδων βαθιάς μάθησης σε συνδυασμό με τηλεπισκοπικές αρχές, αποτελεί μια επίπονη διαδικασία, η οποία όμως μπορεί να συνδράμει στην επίλυση ποικίλων προβλημάτων. Ο όγκος και το κόστος των δεδομένων από δορυφορικούς δέκτες, η μεγάλη υπολογιστή ισχύς που απαιτείται για την επεξεργασία των δεδομένων και η ανεύρεση ή δημιουργία δεδομένων εκπαίδευσης νευρωνικών δικτύων, είναι από τις πρωταρχικές δυσχέρειες που συναντά ένας μελετητής στην επίλυση ανάλογων ζητημάτων. Παρά ταύτα η εκτίμηση κατάστασης σε περίπτωση κρίσης, η πληροφόρηση για λήψη μέτρων και αποφάσεων, καθώς και η επικαιροποίηση βάσης δεδομένων, αποτελούν ζητήματα διεπιστημονικού επιπέδου, στα οποία μπορούν να συνδράμουν αποτελεσματικά οι προαναφερθείσες διαδικασίες αυτόματου εντοπισμού. Στην παρούσα μελέτη με την εκπαίδευση ενός πλήρως συνελκτικού νευρωνικού δικτύου U-Net και χρησιμοποιώντας ελεύθερο πακέτο δεδομένων εκπαίδευσης από την SpaceNet, επετεύχθη η ανάδειξη βέλτιστου σεναρίου για τον εντοπισμό ανθρωπογενών κατασκευών σε μη κατοικημένες περιοχές από δορυφορικές εικόνες.

Λέξεις κλειδιά

Αυτόματη εξαγωγή ανθρωπογενών κατασκευών, τηλεπισκόπηση, SpaceNet 1, βαθιά μάθηση, U-Net

Abstract

Automated processes of object detection from images using deep learning methods combined with remote sensing principles is a laborious process, but one that can help solve a variety of problems. The volume and cost of data from satellite receivers, the large computing power required to process the data, and finding or generating neural network training data are among the primary difficulties a researcher encounters in solving analogous problems. Nevertheless, the assessment of the situation in case of a crisis, the information for taking measures and decisions, as well as the updating of the database, are issues of interdisciplinary level, which can be effectively assisted by the aforementioned automatic detection procedures. In this study, by training a fully convolutional U-Net neural network and using a free training data package from SpaceNet, the emergence of a best-case scenario for the detection of man-made structures in uninhabited areas from satellite images was achieved.

Keywords

Automatic extraction of man-made structures, remote sensing, SpaceNet 1, deep learning, U-Net

Περιεχόμενα

Ευχαριστίες	2
Περίληψη	4
Λέξεις κλειδιά.....	4
Abstract.....	5
Keywords.....	5
Περιεχόμενα	6
Ευρετήριο εικόνων.....	8
Ευρετήριο πινάκων	9
Κατάλογος συντομογραφιών	9
ΚΕΦΑΛΑΙΟ 1	11
Εισαγωγή	11
1.1 Η σημασία αυτόματου εντοπισμού ανθρωπογενών κατασκευών.....	11
1.2 Προκλήσεις.....	11
1.3 Συνεισφορά	13
1.4 Δομή εργασίας.....	14
ΚΕΦΑΛΑΙΟ 2	15
Βιβλιογραφική ανασκόπηση	15
2.1 Αυτόματη ανίχνευση αλλαγών κτιρίων σε αεροφωτογραφίες με χρήση συνελικτικών νευρωνικών δικτύων και χειροποίητων χαρακτηριστικών	15
2.2 Χαρτογράφηση άτυπων οικισμών στις αναπτυσσόμενες χώρες με πολυφασματικά δεδομένα πολλαπλής ανάλυσης	15
2.3 Ανίχνευση ανθρωπογενών κατασκευών και αλλαγών σε δορυφορικές εικόνες με ένα σύστημα ανάκτησης πληροφοριών βασισμένο στο περιεχόμενο που βασίζεται στην αυτο-οργάνωση	16
2.4 Ανίχνευση τεχνητών στόχων από δεδομένα Polarimetric SAR μέσω μη στασιμότητας και ασυμμετρίας.....	16
ΚΕΦΑΛΑΙΟ 3	18
Θεωρητικό Υπόβαθρο	18
3. Εισαγωγικά.....	18
3.1 Τεχνητή Νοημοσύνη	18
3.2 Συνελικτικά νευρωνικά δίκτυα.....	24
3.3 Πλήρως συνελικτικά νευρωνικά δίκτυα	27

3.4 Τηλεπισκοπικές Αρχές	30
ΚΕΦΑΛΑΙΟ 4	35
Προγραμματισμός διαδικασιών σε Python	35
4. Εισαγωγικά.....	35
4.1 Η Python ως γλώσσα προγραμματισμού	35
4.2 Google Colab	36
4.3 Βιβλιοθήκες που χρησιμοποιήθηκαν	36
4.4 Υλοποίηση σε Python.....	37
ΚΕΦΑΛΑΙΟ 5	40
Διαχείριση δεδομένων	40
5 Εισαγωγή.....	40
5.1 Πηγές δεδομένων	40
5.2 Διαχείριση δεδομένων.....	42
ΚΕΦΑΛΑΙΟ 6	44
Μελέτη σεναρίων.....	44
6. Εισαγωγικά.....	44
6.1 Δείκτες αξιολόγησης αποτελεσμάτων.....	44
6.4 Παράμετροι εκπαίδευσης U-Net.....	47
6.3 Παράθεση και αξιολόγηση σεναρίων	49
1° ΣΕΝΑΡΙΟ: Εικόνες τριών καναλιών (RGB).....	50
2° ΣΕΝΑΡΙΟ: Εικόνες τεσσάρων καναλιών (RGBNir)	51
3° ΣΕΝΑΡΙΟ: Εικόνες NDVI	53
4° ΣΕΝΑΡΙΟ: Εικόνες έγχρωμου σύνθετου ανίχνευσης τσιμέντου (cement)	55
5° ΣΕΝΑΡΙΟ: Εικόνες ενός καναλιού (Nir).....	57
6.4 Ανάδειξη βέλτιστου σεναρίου	59
ΚΕΦΑΛΑΙΟ 7	62
Συμπεράσματα.....	62
7.1 Γενικά	62
7.2 Δυσχέρειες	62
7.3 Πορίσματα.....	62
7.4 Μελλοντικές επεκτάσεις.....	63
Αναφορές.....	64
Παραρτήματα	67

A. Μοντέλα εκπαίδευσης δικτύου U-Net σε πέντε σενάρια.....	67
B. Κώδικας σε γλώσσα προγραμματισμού Python.....	67

Ευρετήριο εικόνων

Εικόνα 1. Τεχνητή νοημοσύνη.....	18
Εικόνα 2. Φάσεις αλγορίθμου Μηχανικής Μάθησης.....	20
Εικόνα 3. Τυπική δομή πολυεπίπεδων μοντέλων.....	21
Εικόνα 4. Λειτουργία τεχνητού νευρώνα.....	22
Εικόνα 5. Γραφική παράσταση συνάρτησης ReLU.....	23
Εικόνα 6. Παράδειγμα πράξης συνέλιξης σε εικόνα τριών καναλιών.....	25
Εικόνα 7. Παράδειγμα πράξης max pooling σε εικόνα.....	26
Εικόνα 8. Συνολική αρχιτεκτονική ενός CNN.....	27
Εικόνα 9. Σχηματική αναπαράσταση μετατροπής ενός CNN σε FCN.....	27
Εικόνα 10. Η αρχιτεκτονική U-Net.....	29
Εικόνα 11. Η έννοια της φασματοσκοπικής απεικόνισης.....	31
Εικόνα 12. Φασματικές υπογραφές κοινών υλικών.....	32
Εικόνα 13. Εισαγωγή βιβλιοθηκών.....	37
Εικόνα 14. Εντολή αρχικοποίησης παραμέτρων εισαγωγής.....	38
Εικόνα 15. Κλάδος Down-sampling της αρχιτεκτονικής του δικτύου U-Net.....	38
Εικόνα 16. Κλάδος Down-sampling της αρχιτεκτονικής του δικτύου U-Net.....	39
Εικόνα 17. Καταγραφή στατιστικών αποτελεσμάτων.....	39
Εικόνα 18. Περιοχή κάλυψης μωσαϊκού ψηφιδωτών δεδομένων.....	40
Εικόνα 19. Δορυφορικός αισθητήρας WorldView-2 (Copyright © MAXAR. All rights reserved.).....	41
Εικόνα 20. Απεικόνιση των δεδομένων εισόδου 1 ^{ου} σεναρίου. Αριστερά στήλη οι δορυφορικές εικόνες RGB (input RGB images). Δεξιά στήλη οι μάσκες των κτηρίων (ground truth labels).....	43
Εικόνα 21. Απεικόνιση των δεδομένων εισόδου 4 ^{ου} σεναρίου. Αριστερά στήλη οι δορυφορικές εικόνες cement (input cement images). Δεξιά στήλη οι μάσκες των κτηρίων (ground truth labels).....	43
Εικόνα 22. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 1ου σεναρίου.....	51
Εικόνα 23. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 2ου σεναρίου.....	53
Εικόνα 24. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 3ου σεναρίου.....	55
Εικόνα 25. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 4ου σεναρίου.....	57
Εικόνα 26. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 5ου σεναρίου.....	59

Εικόνα 27. Παράθεση δεδομένων εισόδου και πρόβλεψης 1ου, 2ου και 3ου σεναρίων	60
-------------------------------------------------------------------------------------	----

Ευρετήριο πινάκων

Πίνακας 1. Προτεινόμενα έγχρωμα σύνθετα Worldview-2 Σφάλμα! Δεν έχει οριστεί σελιδοδείκτης.	
Πίνακας 2. σενάρια ψηφιδωτών δεδομένων εκπαίδευσης (raster input x-train)	42
Πίνακας 3. Ορισμός πίνακα σύγχυσης (Confusion Matrix).....	44
Πίνακας 4. Συγκεντρωτικός πίνακας παραμέτρων δικτύου	49
Πίνακας 5. Σενάρια εκπαίδευσης	49

Κατάλογος συντομογραφιών

RGB	Red-Green-Bleu
GNSS	Global Navigation Satellite System
GIS	Geographical Information Systems
NIR	Near InfraRed
NDVI	Normalized Difference Vegetation Index
CNN	Convolutional Neural Network
FCN	Fully-Convolutional Network
CRF	Conditional Random Field
MUL	Multi-Spectral
PAN	Panchromatic
ReLU	Rectified Linear Activation Unit
MLP	Multiple layer perceptron
MDFV	Multidimensional feature vector
OSM	OpenStreetMap
BDT	Binary distance transform
SGD	Stochastic Gradient Descent
ADAM	Adaptive Movement
HSRI	High Spatial Resolution Imagery
PCA	Principal Component Analysis
CIR	Color Infrared
DIM	Dense Image Matching
AI	Artificial Intelligence
NN	Neural Network
ANNs	Artificial neural networks
RNN	Recurrent Neural Networks
D	Dimensions
ROI	Rectangle of Interest
GC	Garbage Collection

API	Application Programming Interface
OpenCV	Open-Source Computer Vision Library
sklearn	Scikit-learn
CUDA	Compute Unified Device Architecture
GPU	graphics processing unit)
CPU	central processing unit
RAM	Random access memory
AdaGrad	Adaptive Gradient Algorithm
RMSProp	Root Mean Square Propagation
TP	True Positive
TN	True Negative
FP	False Positive
FN	False Negative

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή

1.1 Η σημασία αυτόματου εντοπισμού ανθρωπογενών κατασκευών.

Η γνώση του περιβάλλοντος χώρου, η αποτύπωσή του και η ταξινόμηση των αντικειμένων που τον απαρτίζουν αποτελεί αντικείμενο μελέτης σε διεπιστημονικό επίπεδο. Σε πολλούς κλάδους, αναφερόμενοι από την βιομηχανία, τα δημόσια έργα [1] έως και σε ζητήματα υγείας [2], η τηλεπισκόπηση ως επιστήμη σε συνδυασμό με την όραση υπολογιστών (computer vision) [3], έχουν εφαρμογή. Τέτοιες εφαρμογές είναι για παράδειγμα ο εντοπισμός και η αποτύπωση μιας μισγάγγειας ή ενός μεγάλου τεχνικού έργου σε μη αστική περιοχή, ακόμη και η πρόβλεψη εκδήλωσης ασθενειών από φασματική ανάλυση ακτινογραφιών. Διαπιστώνουμε ότι τα ζητήματα όπου μπορούν να συνδράμουν για την επίλυση τους, αυτοματοποιημένες διαδικασίες εντοπισμού είναι ποικίλα και διαχωρίζονται κατά περίπτωση ανάλογα με τη φύση του ζητήματος ή την αναγκαία κλίμακα αποτύπωσης.

Σήμερα, χαρακτηριστικά χαρτών όπως δρόμοι, αποτυπώματα κτιρίων και σημεία ενδιαφέροντος δημιουργούνται κυρίως με χειροκίνητες τεχνικές [4]. Η πρόοδος των αυτοματοποιημένων τεχνικών εξαγωγής χαρακτηριστικών θα εξυπηρετήσει σημαντικές μεταγενέστερες χρήσεις των χαρτογραφικών δεδομένων, συμπεριλαμβανομένης της ανθρωπιστικής βοήθειας και της αντιμετώπισης καταστροφών. Επιπλέον, η επίλυση αυτής της πρόκλησης αποτελεί σημαντικό σκαλοπάτι για την απελευθέρωση της ισχύος των προηγμένων αλγορίθμων υπολογιστικής όρασης που εφαρμόζονται σε μια ποικιλία εφαρμογών δεδομένων τηλεπισκόπησης τόσο στον δημόσιο όσο και στον ιδιωτικό τομέα.

Για την εκπόνηση της παρούσας μελέτης χρησιμοποιήθηκε το U-Net, το οποίο είναι ένας τύπος συνελκτικού νευρωνικού δικτύου (CNN) που αναπτύχθηκε για την ανάλυση ιατρικών εικόνων [5], αλλά χρησιμοποιείται ευρέως και σε άλλα πεδία που απαιτούν ακριβή τμηματοποίηση εικόνας [6]. Το κύριο χαρακτηριστικό του U-Net είναι η ικανότητά του να χρησιμοποιεί πληροφορίες τόσο από τα χαμηλά όσο και από τα υψηλά επίπεδα ανάλυσης για να παράγει μια ακριβή τμηματοποίηση. Επίσης αποτελεί ένα ισχυρό εργαλείο για την τμηματοποίηση εικόνων λόγω της αρχιτεκτονικής του που επιτρέπει την αποδοτική αξιοποίηση των χωρικών και χαρακτηριστικών πληροφοριών της εικόνας.

1.2 Προκλήσεις.

Η υλοποίηση του αυτόματου εντοπισμού ανθρωπογενών κατασκευών από δορυφορικές εικόνες μέσω μεθόδων βαθιάς μάθησης αντιμετωπίζει αρκετές προκλήσεις, οι οποίες μπορούν να ταξινομηθούν σε τεχνικές [7], επιχειρησιακές [8] και ηθικές [9].

1.2.1 Τεχνικές Προκλήσεις

Αρχικό ζήτημα αποτελεί η ποιότητα άντλησης δεδομένων τόσο για την εκπαίδευση των δικτύων όσο και για την εξαγωγή συμπερασμάτων ανά περίπτωση μελέτης. Οι δορυφορικές εικόνες ποικίλλουν σε ανάλυση και ποιότητα. Οι εικόνες χαμηλής ανάλυσης μπορεί να δυσκολέψουν την ακριβή ανίχνευση και αναγνώριση κατασκευών. Επίσης, οι θόρυβοι και οι ανωμαλίες στις εικόνες μπορούν να επηρεάσουν την ακρίβεια των μοντέλων.

Έπειτα η πολυπλοκότητα της περιοχής μελέτης, καθώς και η συνάφεια δομικών υλικών των κατασκευών με φυσικά υλικά δυσχεραίνουν την ακρίβεια των αυτοματισμών. Οι αστικές και αγροτικές περιοχές περιλαμβάνουν ποικιλία αντικειμένων και δομών, όπως δέντρα, δρόμοι, ποτάμια και κτίρια, τα οποία μπορεί να είναι δύσκολο να διαχωριστούν και να αναγνωριστούν σωστά από τα μοντέλα βαθιάς μάθησης, αν τα δεδομένα εισόδου δεν είναι επαρκή.

Όπως γνωρίζουμε τα δορυφορικά δεδομένα είναι αρχεία μεγάλου αποθηκευτικού όγκου. Η επεξεργασία μεγάλων όγκων δορυφορικών δεδομένων απαιτεί σημαντική υπολογιστική ισχύ και πόρους, κάτι που μπορεί να είναι δαπανηρό και να απαιτεί εξειδικευμένο εξοπλισμό.

Η ανάπτυξη, εκπαίδευση και βελτιστοποίηση των μοντέλων βαθιάς μάθησης μπορεί να είναι χρονοβόρα και απαιτεί εξειδικευμένη γνώση. Τα μοντέλα πρέπει να είναι αρκετά πολύπλοκα για να εντοπίζουν λεπτομερείς δομές, αλλά ταυτόχρονα αποδοτικά ώστε να μπορούν να εφαρμοστούν σε μεγάλης κλίμακας δεδομένα. Η δημιουργία ενός μεγάλου και καλά εντοπισμένου (annotations) συνόλου δεδομένων είναι απαραίτητη για την εκπαίδευση των μοντέλων. Αυτό απαιτεί μεγάλο ανθρώπινο κόπο και είναι χρονοβόρο. Ελλιπή ή ανακριβή δεδομένα εκπαίδευσης μπορεί να μειώσει την απόδοση των μοντέλων.

1.2.2 Επιχειρησιακές Προκλήσεις

Αρχικά τίθεται θέμα κόστους πρόσβασης σε καλής ποιότητας δεδομένα. Η απόκτηση δορυφορικών εικόνων υψηλής ανάλυσης μπορεί να είναι δαπανηρή και να απαιτεί συνεργασία με οργανισμούς που κατέχουν τα δεδομένα αυτά. Ακόμη η ενσωμάτωση των λύσεων ανίχνευσης με τα υφιστάμενα πληροφοριακά συστήματα και τις υποδομές μπορεί να είναι περίπλοκη και να απαιτεί σημαντικές προσαρμογές.

Τα μοντέλα βαθιάς μάθησης πρέπει να ενημερώνονται και να επαναεκπαιδεύονται τακτικά με νέα δεδομένα για να παραμένουν ακριβή και επίκαιρα, κάτι που απαιτεί συνεχή παρακολούθηση και διαχείριση.

1.2.3 Νομικές και Ηθικές Προκλήσεις

Η χρήση δορυφορικών εικόνων μπορεί να εγείρει ζητήματα ιδιωτικότητας και ασφάλειας, ειδικά όταν αφορά στην ανίχνευση ατόμων ή ιδιωτικών ιδιοκτησιών. Ταυτόχρονα η χρήση και η διανομή δορυφορικών δεδομένων υπόκειται σε διάφορους κανονισμούς και πολιτικές, οι οποίες μπορεί να διαφέρουν ανάλογα με τη χώρα ή την περιοχή.

Η αντιμετώπιση αυτών των προκλήσεων απαιτεί μια διεπιστημονική προσέγγιση, συνδυάζοντας την εξειδικευμένη τεχνογνωσία στην επεξεργασία εικόνων, την τεχνητή νοημοσύνη, την υπολογιστική ισχύ και τη γνώση των τομέων εφαρμογής. Η συνεχής έρευνα και η ανάπτυξη νέων τεχνολογιών θα συμβάλλουν στη βελτίωση της ακρίβειας και της αποδοτικότητας αυτών των συστημάτων.

1.3 Συνεισφορά

Ο αυτόματος εντοπισμός ανθρωπογενών κατασκευών από δορυφορικές εικόνες με χρήση μεθόδων βαθιάς μάθησης έχει μεγάλη σημασία σε διάφορους τομείς και εφαρμογές. Ακολουθούν αναφορικά τομείς και πώς μπορεί να συνεισφέρει σε αυτούς η εν λόγω αυτοματοποιημένη διαδικασία.

➤ Χωροταξικός Σχεδιασμός και Διαχείριση Γης

Οι δορυφορικές εικόνες μπορούν να χρησιμοποιηθούν για την παρακολούθηση της ανάπτυξης των πόλεων, την ανίχνευση νέων κτιρίων και την αξιολόγηση των αλλαγών στη χρήση γης. Αυτό βοηθά τους πολεοδόμους και τους κυβερνητικούς φορείς να λαμβάνουν ενημερωμένες αποφάσεις για την ανάπτυξη και διαχείριση των αστικών και αγροτικών περιοχών.

➤ Περιβαλλοντική Παρακολούθηση

Η ανίχνευση ανθρωπογενών κατασκευών βοηθά στον εντοπισμό και την παρακολούθηση των περιβαλλοντικών επιπτώσεων από την ανθρώπινη δραστηριότητα, όπως η αποψίλωση των δασών, η ρύπανση και οι αλλαγές στη βιοποικιλότητα.

➤ Αντιμετώπιση Φυσικών Καταστροφών

Μετά από φυσικές καταστροφές, όπως σεισμοί, πλημμύρες ή πυρκαγιές, ο αυτόματος εντοπισμός των κατεστραμμένων κατασκευών βοηθά στον άμεσο εντοπισμό των περιοχών που έχουν πληγεί και την εκτίμηση των ζημιών. Αυτό επιταχύνει την ανταπόκριση και την παροχή βοήθειας.

➤ Παρακολούθηση Υποδομών

Η τακτική παρακολούθηση των υποδομών όπως δρόμοι, γέφυρες και κτίρια βοηθά στον έγκαιρο εντοπισμό προβλημάτων και στην προληπτική συντήρηση, μειώνοντας τον κίνδυνο ατυχημάτων και βελτιώνοντας τη δημόσια ασφάλεια.

➤ Εμπορικές και Επιχειρηματικές Εφαρμογές

Οι επιχειρήσεις μπορούν να χρησιμοποιούν δορυφορικές εικόνες για την ανάλυση της τοποθεσίας νέων κτιρίων ή εγκαταστάσεων, την αξιολόγηση των ανταγωνιστών και τη λήψη αποφάσεων για επενδύσεις και ανάπτυξη.

1.4 Δομή εργασίας

Η παρούσα διπλωματική εργασία διαρθρώνεται συνολικά σε επτά (7) κεφάλαια.

- Στο **Κεφάλαιο 1** όπως παρουσιάστηκε παραπάνω, περιέχεται η εισαγωγή και η ανάλυση του αντικείμενου της έρευνας που διεξήχθη.
- Στο **Κεφάλαιο 2** πραγματοποιείται μια βιβλιογραφική ανασκόπηση σε μελέτες αντίστοιχου περιεχομένου με την παρούσα, που έχουν εντυφίσει σε κάποιο βαθμό σε μεθόδους αυτοματοποιημένου εντοπισμού αντικειμένων από δορυφορικές εικόνες .
- Στο **Κεφάλαιο 3** παρουσιάζονται οι θεωρητικές γνώσεις που απαιτούνται για την κατανόηση όλων των εννοιών που αποτελούν αντικείμενο της παρούσας μελέτης.
- Στο **Κεφάλαιο 4** περιγράφονται γενικά χαρακτηριστικά της γλώσσας προγραμματισμού Python και αναλύεται λεπτομερώς ο κώδικας που συντάχθηκε για την επίλυση του προβλήματος.
- Στο **Κεφάλαιο 5** γίνεται αναφορά στις πηγές άντλησης δεδομένων, αναλύονται οι διαδικασίες για την προετοιμασία των δεδομένων ώστε να εισαχθούν ως δεδομένα εκπαίδευσης στο δίκτυο.
- Στο **Κεφάλαιο 6** αναλύονται τα μέτρα αξιολόγησης που χρησιμοποιήθηκαν. Επιπρόσθετα παρουσιάζονται και αξιολογούνται τα πειραματικά αποτελέσματα για κάθε σενάριο ξεχωριστά αλλά και συγκεντρωτικά.
- Στο **Κεφάλαιο 7** συγκεντρώνονται τα συμπεράσματα που προέκυψαν από την παρούσα μελέτη μεταξύ των οποίων και οι δυσχέρειες που προκλήθηκαν κατά τη διαδικασία ενώ προτείνονται μελλοντικές επεκτάσεις της.

ΚΕΦΑΛΑΙΟ 2

Βιβλιογραφική ανασκόπηση

Προτού εισαχθούμε και εντρυφήσουμε στο θεωρητικό υπόβαθρο, στο παρόν κεφάλαιο παρατίθεται μια βιβλιογραφική ανασκόπηση. Μέσω αυτής της σύντομης εξέτασης μεθόδων αυτόματου εντοπισμού αντικειμένων από δορυφορικές εικόνες με τη χρήση όρασης υπολογιστών, θα αποκτήσουμε μια πιο σφαιρική άποψη επί του θέματος, οπότε θα είμαστε σε θέση να κατανοήσουμε καλύτερα τη φυσική ροή της παρούσας μελέτης.

2.1 Αυτόματη ανίχνευση αλλαγών κτιρίων σε αεροφωτογραφίες με χρήση συνελκτικών νευρωνικών δικτύων και χειροποίητων χαρακτηριστικών

Στην εν λόγω μελέτη παρουσιάζεται ένα νέο πλαίσιο για την επίλυση του έργου της ανίχνευσης αλλαγών σε κτίρια [10], κάνοντας χρήση ενός εξελικτικού νευρωνικού δικτύου (CNN) για το στάδιο της ανίχνευσης κτιρίων και ενός συνόλου χειροποίητων χαρακτηριστικών για την ανίχνευση αλλαγών. Τα κτίρια εξάγονται χρησιμοποιώντας τη μέθοδο που ονομάζεται Mask R-CNN, η οποία είναι ένα νευρωνικό δίκτυο που χρησιμοποιείται για την τμηματοποίηση αντικειμένων με βάση το αντικείμενο και έχει δοκιμαστεί σε διάφορες μελέτες περίπτωσης για την τμηματοποίηση διαφορετικών τύπων αντικειμένων, επιτυγχάνοντας καλά αποτελέσματα. Τα κτίρια ανιχνεύονται σε *bitemporal* εικόνες, όπου χρησιμοποιούνται τρεις διαφορετικές μετρικές σύγκρισης MSE, PSNR και SSIM για να διακρίνονται αν υπάρχουν αλλαγές στα κτίρια, χρησιμοποιήθηκαν αυτές τις μετρικές στην αναπαράσταση της απόχρωσης, του κορεσμού και της φωτεινότητας της εικόνας. Τέλος, τα χαρακτηριστικά ταξινομούνται από δύο αλγόριθμους, Support Vector Machine και Random Forest, έτσι ώστε να μπορούν να συγκριθούν και τα δύο αποτελέσματα. Τα πειράματα πραγματοποιήθηκαν σε ένα μεγάλο σύνολο δεδομένων που ονομάζεται σύνολο δεδομένων κτιρίων WHU, το οποίο περιέχει αεροφωτογραφίες πολύ υψηλής ανάλυσης (VHR). Τα αποτελέσματα που προέκυψαν κρίθηκαν ως αξιόλογα και αξιόπιστα.

2.2 Χαρτογράφηση άτυπων οικισμών στις αναπτυσσόμενες χώρες με πολυφασματικά δεδομένα πολλαπλής ανάλυσης

Ο εντοπισμός και η χαρτογράφηση άτυπων οικισμών [11] περιλαμβάνει αρκετούς από τους στόχους βιώσιμης ανάπτυξης των Ηνωμένων Εθνών. Αυτό συμβαίνει επειδή οι άτυποι οικισμοί φιλοξενούν τους πιο ευάλωτους κοινωνικά και οικονομικά ανθρώπους στον πλανήτη. Έτσι, η κατανόηση του πού βρίσκονται αυτοί οι οικισμοί είναι υψίστης σημασίας τόσο για την κυβέρνηση όσο και για μη κυβερνητικές

οργανώσεις (ΜΚΟ), όπως το Ταμείο των Ηνωμένων Εθνών για τα Παιδιά (UNICEF), οι οποίες μπορούν να χρησιμοποιήσουν αυτές τις πληροφορίες για να παράσχουν αποτελεσματική κοινωνική και οικονομική βοήθεια. Για την εκπόνηση της μελέτης προτάθηκαν δύο αποτελεσματικές μέθοδοι για τον εντοπισμό και τη χαρτογράφηση των θέσεων των άτυπων οικισμών. Η μία χρησιμοποιεί μόνο χαμηλής ανάλυσης (LR), ελεύθερα διαθέσιμες, πολυφασματικές δορυφορικές εικόνες Sentinel-2 με θορυβώδεις σχολιασμούς, ενώ η άλλη είναι μια προσέγγιση βαθιάς μάθησης που χρησιμοποιεί μόνο δαπανηρές δορυφορικές εικόνες πολύ υψηλής ανάλυσης (VHR). Ενδέχεται να είναι η πρώτη για την εποχή της μελέτη που χαρτογράφησε με επιτυχία άτυπους οικισμούς με δορυφορικές εικόνες χαμηλής ανάλυσης. Αξιολογούμε και συγκρίνουμε εκτενώς τις προτεινόμενες μεθόδους.

2.3 Ανίχνευση ανθρωπογενών κατασκευών και αλλαγών σε δορυφορικές εικόνες με ένα σύστημα ανάκτησης πληροφοριών βασισμένο στο περιεχόμενο που βασίζεται στην αυτο-οργάνωση

Στην συγκεκριμένη μελέτη αναδεικνύεται η αυξανόμενη ποσότητα και ανάλυση των δορυφορικών αισθητήρων απαιτεί νέες τεχνικές για την περιήγηση στα αρχεία εικόνων τηλεπισκόπησης [12]. Η αναζήτηση με βάση το περιεχόμενο επιτρέπει την αποτελεσματική ανάκτηση εικόνων με βάση τις πληροφορίες που περιέχουν και όχι την ημερομηνία λήψης ή τη γεωγραφική τους έκταση. Οι αυτοοργανωτικοί χάρτες (SOM) έχουν εφαρμοστεί με επιτυχία στο σύστημα PicSOM για την ανάκτηση εικόνων βάσει περιεχομένου σε βάσεις δεδομένων συμβατικών εικόνων. Διερευνώνται και επεκτείνονται οι δυνατότητες του PicSOM για την ανάλυση δεδομένων τηλεπισκόπησης. Προτείνονται μέθοδοι για την ανίχνευση ανθρωπογενών κατασκευών, καθώς και την εποπτευόμενη και μη εποπτευόμενη ανίχνευση αλλαγών, με βάση το ίδιο πλαίσιο. Δημιουργήθηκε τεχνητά μια βάση δεδομένων με τη διάσπαση κάθε δορυφορικής εικόνας προς ανάλυση σε μικρές εικόνες. Αφού εκπαιδεύτηκε το PicSOM σε αυτή τη βάση δεδομένων imagelet, πραγματοποιήθηκαν τόσο διαδραστικά όσο και off-line ερωτήματα για την ανίχνευση ανθρωπογενών κατασκευών, καθώς και αλλαγών μεταξύ δύο εικόνων πολύ υψηλής ανάλυσης από διαφορετικά έτη. Τα πειραματικά αποτελέσματα αξιολογήθηκαν ποσοτικά και συζητήθηκαν ποιοτικά και υποδεικνύουν ότι αυτή η νέα προσέγγιση είναι κατάλληλη για την ανάλυση οπτικών δορυφορικών εικόνων πολύ υψηλής ανάλυσης. Πιθανές εφαρμογές αυτής της εργασίας περιλαμβάνουν τη διαδραστική ανίχνευση τεχνητών κατασκευών ή την εποπτευόμενη παρακολούθηση ευαίσθητων περιοχών.

2.4 Ανίχνευση τεχνητών στόχων από δεδομένα Polarimetric SAR μέσω μη στασιμότητας και ασυμμετρίας

Η ανίχνευση ανθρωπογενών στόχων σε αστικές περιοχές με τη χρήση Polarimetric SAR δεδομένων, ραντάρ συνθετικού ανοίγματος (PolSAR) [13] έχει γίνει ένας πολλά

υποσχόμενος ερευνητικός τομέας, δεδομένου ότι έχει στενή σχέση με τον αστικό σχεδιασμό, την υπηρεσία διάσωσης κ.λπ. Στην συγκεκριμένη εργασία παρουσιάζεται μια βελτιωμένη μέθοδος ανίχνευσης ανθρωπογενών στόχων για δεδομένα PolSAR με βάση τη μη στασιμότητα και την ασυμμετρία. Η μη στασιμότητα στη διεύθυνση του αζιμούθιου χρησιμοποιείται ήδη για το διαχωρισμό των τεχνητών και των φυσικών στόχων σε αστικές περιοχές. Ωστόσο, εξακολουθούν να υπάρχουν ορισμένα μειονεκτήματα. Ορισμένοι μικροί ανθρωπογενείς στόχοι και δρόμοι δεν μπορούν να ανιχνευθούν αποτελεσματικά. Επιπλέον, η μη στασιμότητα μπορεί να εμφανιστεί και σε ορισμένες άλλες φυσικές επιφάνειες, όπως οι καλλιεργήσιμες εκτάσεις με συντονισμό Bragg. Ως εκ τούτου, για να επιλυθούν αυτά τα προβλήματα, ενσωματώνεται η ασυμμετρία ανάκλασης στη μέθοδο εξαγωγής μη στασιμότητας από το αζιμούθιο για να βελτιώσουμε την ακρίβεια ανίχνευσης των ανθρωπογενών στόχων, δηλαδή αφαιρούνται οι φυσικές περιοχές και ανιχνεύονται οι μικροί στόχοι. Χρησιμοποιούνται αερομεταφερόμενα δεδομένα ESAR και διαστημικά δεδομένα PALSAR για την επικύρωση της απόδοσης της προτεινόμενης μεθόδου. Το αποτέλεσμα που λαμβάνεται με την προτεινόμενη μέθοδο παρουσιάζει 20% υψηλότερη ακρίβεια από το αποτέλεσμα που βασίζεται στην αρχική μέθοδο εξαγωγής μη στασιμότητας. Οι φυσικές περιοχές με συντονισμό Bragg αφαιρούνται. Επιπλέον, τα περισσότερα κτίρια και ορισμένοι μεταλλικοί φράκτες κατά μήκος του δρόμου μπορούν επίσης να ανιχνευθούν με ακρίβεια.

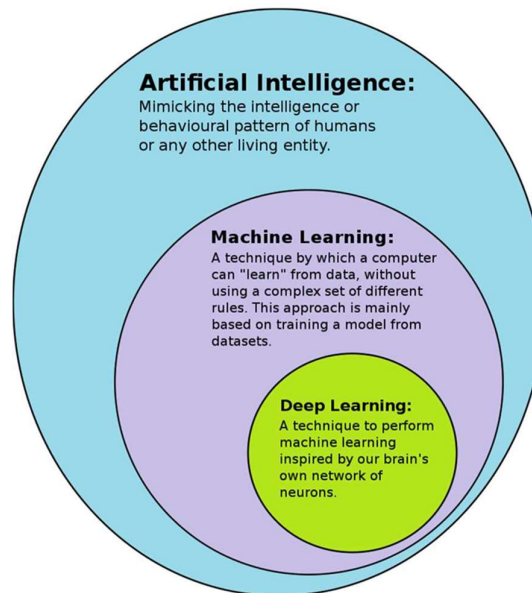
ΚΕΦΑΛΑΙΟ 3

Θεωρητικό Υπόβαθρο

3. Εισαγωγικά

Όπως διαπιστώνουμε ο αυτόματος εντοπισμός αντικειμένων και συγκεκριμένα ανθρωπογενών κατασκευών από δορυφορικές, είναι μια διεπιστημονικού ενδιαφέροντος μεθοδολογία. Για την κατανόηση των διαδικασιών που ακολουθήθηκαν αλλά και των αλγορίθμων που επιλέχθηκαν και προσπελάστηκαν, απαραίτητο κρίνεται να αναφερθούμε σε θεωρητικό υπόβαθρο. Στο παρόν κεφάλαιο παρουσιάζονται οι θεωρητικές γνώσεις που απαιτούνται για την κατανόηση όλων των εννοιών που αποτελούν αντικείμενο της παρούσας μελέτης. Αρχικά αναλύονται βασικές έννοιες μηχανικής μάθησης και καταλήγουμε στο πλήρως συνελκτικό νευρωνικό δίκτυο (U-Net) και την ανάλυση της αρχιτεκτονικής του. Επιπρόσθετα παρατίθενται και τηλεπισκοπικές αρχές για την επεξήγηση επιλογής των σεναρίων εκπαίδευσης του νευρωνικού δικτύου.

3.1 Τεχνητή Νοημοσύνη



Εικόνα 1. Τεχνητή νοημοσύνη.

Η επιβίωση του ανθρώπινου είδους και η θέση του στην τροφική αλυσίδα οφείλεται στο επίπεδο νοημοσύνης που διαθέτει. Για την επίλυση πολύπλοκων μαθηματικών προβλημάτων και εξαγωγή συμπερασμάτων από μελέτη προτύπων, αναπτύχθηκαν συστήματα τεχνητής νοημοσύνης. Στο πλαίσιο της παρούσας μελέτης, όπως και σε άλλες αντίστοιχες που μελετήσαμε στο κεφάλαιο της βιβλιογραφικής ανασκόπησης, εκπαιδεύτηκαν νευρωνικά δίκτυα, για την αντιμετώπιση του ζητήματος της

αυτόματης εξαγωγής κτηρίων, τα οποία αποτελούν βασικά εργαλεία βαθιάς μάθησης.

Η βαθιά μάθηση αποτελεί κομμάτι της μηχανικής μάθησης, η οποία αποτελεί κομμάτι της τεχνητής νοημοσύνης. Στο παρόν κεφάλαιο θα κάνουμε μια σύντομη ανασκόπηση σε αυτές τις βασικές έννοιες, ώστε να είμαστε σε θέση εν συνεχεία να αντιληφθούμε και να αξιολογήσουμε τα αποτελέσματα εκπαίδευσης των μοντέλων πρόβλεψης.

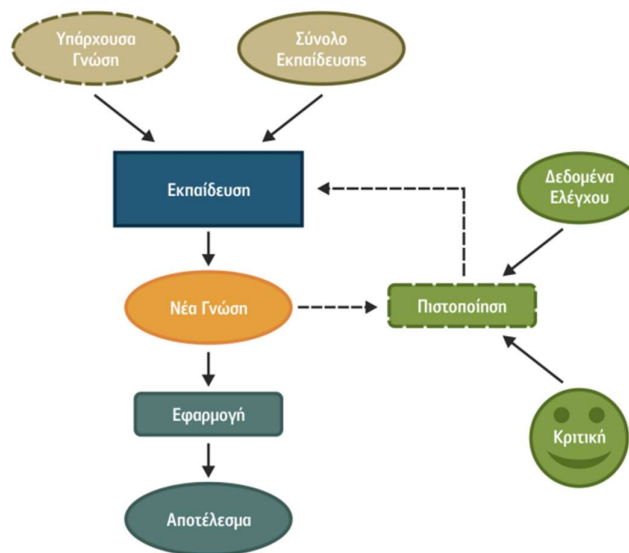
3.1.1 Μηχανική Μάθηση (machine learning)

Η Μάθηση (Learning) είναι μία από τις θεμελιώδεις ιδιότητες της νοήμονος συμπεριφοράς του ανθρώπου. Με κύριο εργαλείο τη μάθηση λαμβάνουμε αποφάσεις για την αντιμετώπιση προβλημάτων που συναντάμε ξανά ή και για πρώτη φορά. Εμπνευσμένοι από αυτή μας την ικανότητα, για την αντιμετώπιση δύσκολων υπολογιστικά και προγραμματιστικά προβλημάτων αναπτύξαμε τη Μηχανική Μάθηση (machine learning) [14]. Αυτή μπορεί να οριστεί ως το φαινόμενο κατά το οποίο ένα σύστημα βελτιώνει την απόδοσή του κατά την εκτέλεση μιας συγκεκριμένης εργασίας, χωρίς να υπάρχει ανάγκη να προγραμματιστεί εκ νέου. Βάσει του ορισμού αυτού, η Μηχανική Μάθηση έχει ως σκοπό τη δημιουργία μηχανών ικανών να μαθαίνουν, να βελτιώνουν, δηλαδή, την απόδοσή τους σε κάποιους τομείς μέσω της αξιοποίησης προηγούμενης γνώσης και εμπειρίας. Ένας αλγόριθμος μηχανικής μάθησης είναι ένας αλγόριθμος που είναι σε θέση να μαθαίνει από δεδομένα. Η εισαγωγή της μηχανικής μάθησης επέτρεψε στους υπολογιστές να αντιμετωπίσουν προβλήματα που αφορούν τη γνώση του πραγματικού κόσμου και να λαμβάνουν αποφάσεις που φαίνονται υποκειμενικές. Εν γένει, ο τομέας της Μηχανικής Μάθησης αναπτύσσει τρεις τρόπους μάθησης, ανάλογους με τους τρόπους με τους οποίους μαθαίνει ο άνθρωπος:

- Επιβλεπόμενη Μάθηση (Supervised Learning) είναι η διαδικασία όπου ο αλγόριθμος κατασκευάζει μια συνάρτηση που απεικονίζει δεδομένες εισόδους (σύνολο εκπαίδευσης) σε γνωστές επιθυμητές εξόδους, με απώτερο στόχο τη γενίκευση της συνάρτησης αυτής και για εισόδους με άγνωστη έξοδο.
- Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning), όπου ο αλγόριθμος κατασκευάζει ένα μοντέλο για κάποιο σύνολο εισόδων υπό μορφή παρατηρήσεων χωρίς να γνωρίζει τις επιθυμητές εξόδους.
- Ενισχυτική Μάθηση (Reinforcement Learning), όπου ο αλγόριθμος μαθαίνει μια στρατηγική ενεργειών μέσα από άμεση αλληλεπίδραση με το περιβάλλον.

Για κάθε πρόβλημα προς επίλυση στο χώρο της Μηχανικής Μάθησης υπάρχει ένας κατάλληλος τρόπος μάθησης και για κάθε τρόπο μάθησης υπάρχει τουλάχιστον ένας

κατάλληλος αλγόριθμος που μπορεί να χρησιμοποιηθεί. Η βασικότερη φάση κάθε αλγόριθμου είναι η **εκπαίδευση**, όπου ο αλγόριθμος χρησιμοποιεί ως είσοδο ένα σύνολο δεδομένων εκπαίδευσης (training set) προς επίτευξη του σκοπού του, τη δημιουργία νέας γνώσης. Επιπλέον, μπορεί είτε να χρησιμοποιήσει λιγότερο ή περισσότερο την υπάρχουσα γνώση είτε να μην τη χρησιμοποιήσει καθόλου. Την εκπαίδευση ακολουθεί η φάση της **πιστοποίησης** της παραγόμενης νέας γνώσης. Συνήθως, η πιστοποίηση πραγματοποιείται καταρχάς από τον ίδιο τον αλγόριθμο μέσω διαδικασιών ανάκλησης (recall) με τη βοήθεια δεδομένων ελέγχου (test data) και, στη συνέχεια, μέσω κριτικής που κάνει ο χρήστης βάσει των γνώσεων που διαθέτει για το πρόβλημα που επιχειρεί να λύσει ο αλγόριθμος. Τέλος, η νέα γνώση δίνεται προς χρήση σε εφαρμογές στις οποίες είναι απαραίτητη, για να λυθούν πραγματικά προβλήματα.



Εικόνα 2.Φάσεις αλγορίθμου Μηχανικής Μάθησης.

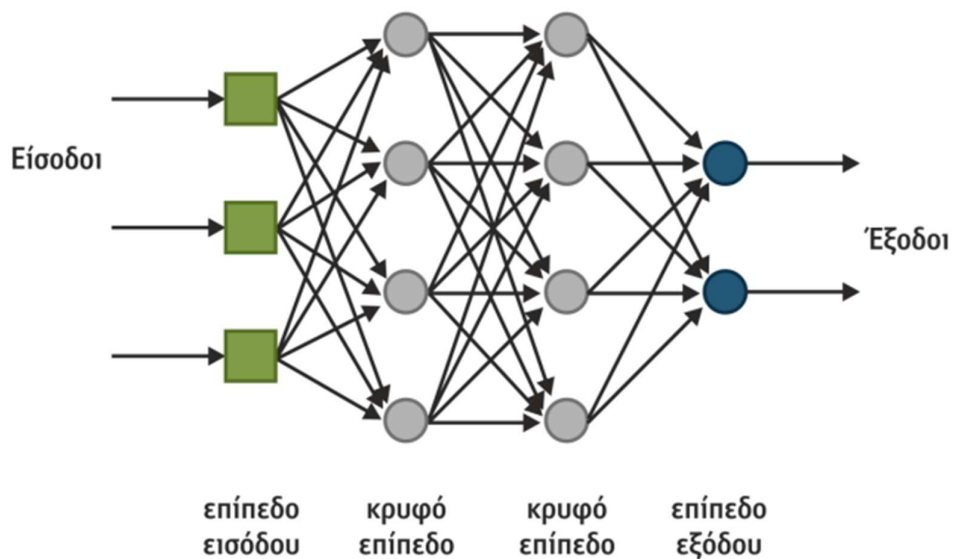
Για να αξιολογηθούν οι ικανότητες ενός αλγορίθμου μηχανικής μάθησης, πρέπει να σχεδιαστεί ένα ποσοτικό μέτρο της απόδοσής του. Συνήθως αυτό το μέτρο απόδοσης είναι ειδικά για την εργασία που εκτελείται από το σύστημα. Για παράδειγμα, στην ταξινόμηση, μετράται συχνά η ακρίβεια (accuracy) του μοντέλου, η οποία ορίζεται ως η αναλογία παραδειγμάτων για τα οποία το μοντέλο παράγει τη σωστή έξοδο [15]. Τα μέτρα απόδοσης και οι δείκτες αξιολόγησης για την παρούσα εργασία θα αναλυθούν παρακάτω και συγκεκριμένα στο έκτο κεφάλαιο.

3.1.2 Βαθιά Μάθηση (deep learning)

Η βαθιά μάθηση (deep learning) [16] είναι μια προσέγγιση στη μηχανική μάθηση που έχει αντλήσει σε μεγάλο βαθμό τις γνώσεις μας για τον ανθρώπινο εγκέφαλο, τη στατιστική και τα εφαρμοσμένα μαθηματικά, καθώς αναπτύχθηκε τις τελευταίες

δεκαετίες. Τα τελευταία χρόνια, έχει γνωρίσει τεράστια αύξηση της δημοτικότητας και της χρησιμότητάς της, η οποία οφείλεται σε μεγάλο βαθμό στους ισχυρότερους υπολογιστές, στα μεγαλύτερα σύνολα δεδομένων και στις τεχνικές εκπαίδευσης βαθύτερων δικτύων.

Τα θεωρητικά θεμέλια της βαθιάς μάθησης έχουν τις ρίζες τους στην κλασική βιβλιογραφία των νευρωνικών δικτύων (NN). Όμως, σε αντίθεση με την πιο παραδοσιακή χρήση των NN, η βαθιά μάθηση προβλέπει τη χρήση πολλών κρυφών νευρώνων και στρωμάτων - τυπικά περισσότερων από δύο - ως αρχιτεκτονικό πλεονέκτημα σε συνδυασμό με νέα παραδείγματα εκπαίδευσης. Ενώ η προσφυγή σε πολλούς νευρώνες επιτρέπει την εκτεταμένη κάλυψη των ακατέργαστων δεδομένων που βρίσκονται στη διάθεσή μας.



Εικόνα 3. Τυπική δομή πολυεπίπεδων μοντέλων.

Η βαθιά μάθηση επιτρέπει σε υπολογιστικά μοντέλα που αποτελούνται από πολλαπλά επίπεδα επεξεργασίας να μαθαίνουν αναπαραστάσεις δεδομένων με πολλαπλά επίπεδα αφαίρεσης. Αυτές οι μέθοδοι έχουν βελτιώσει δραματικά την κατάσταση στην αναγνώριση ομιλίας, στην οπτική αναγνώριση αντικειμένων, στην ανίχνευση αντικειμένων και σε πολλούς άλλους τομείς, όπως η ανακάλυψη φαρμάκων και η γονιδιωματική. Η βαθιά μάθηση ανακαλύπτει περίπλοκες δομές σε μεγάλα σύνολα δεδομένων.

3.1.3 Νευρωνικά Δίκτυα (Neural Networks)

Ο όρος Νευρωνικά Δίκτυα (Neural Networks) [17] περιγράφει έναν αριθμό από διαφορετικά μαθηματικά μοντέλα, εμπνευσμένα από αντίστοιχα βιολογικά μοντέλα, δηλαδή μοντέλα που προσπαθούν να μιμηθούν τη συμπεριφορά των νευρώνων του ανθρώπινου εγκεφάλου.

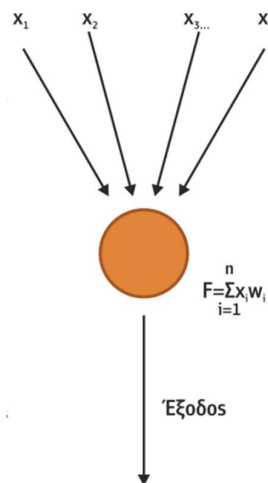
Τα μαθηματικά μοντέλα των τεχνητών νευρωνικών δικτύων, αποτελούνται από έναν αριθμό απλών και με υψηλό βαθμό εσωτερικής διασύνδεσης επεξεργαστικών μονάδων, οργανωμένων σε στρώματα. Τα Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks ANN) επεξεργάζονται πληροφορίες ανταποκρινόμενα δυναμικά σε εξωτερικά ερεθίσματα (εισόδους). Κάθε τεχνητός νευρώνας αποτελείται από πολλές εισόδους x_i και μία μόνο έξοδο y . Κάθε είσοδος x_i “ζυγίζεται” με ένα βάρος w_i και τα αποτελέσματα αθροίζονται μέσω της συνάρτησης αθροίσματος (summation function) F :

$$F = \sum_i^n x_i w_i \quad (1)$$

Ο τεχνητός νευρώνας δίνει έξοδο μέσω της συνάρτησης μετάβασης (transfer function), μόνο όταν το ζυγισμένο άθροισμα των εισόδων είναι μεγαλύτερο μιας ορισμένης τιμής κατωφλίου (threshold value) θ , δηλαδή όταν:

$$\sum_i^n x_i w_i - \theta > 0 \quad (2)$$

Τα όσα περιεγράφηκαν για τη λειτουργία του τεχνητού νευρώνα οπτικοποιούνται για καλύτερη κατανόηση στο ακόλουθο σχήμα.



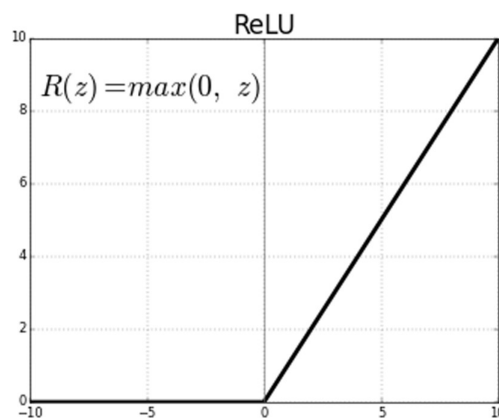
Εικόνα 4. Λειτουργία τεχνητού νευρώνα.

Το αποτέλεσμα τροφοδοτείται ως όρισμα μέσα από την λεγόμενη συνάρτηση ενεργοποίησης, η οποία καθορίζει την έξοδο ενός νευρικού δικτύου ως αυστηρό όριο (hard limit). Η συνάρτηση συνδέεται με κάθε νευρώνα στο δίκτυο και καθορίζει εάν

θα πρέπει να «πυροδοτείται» ή όχι, με κριτήριο κατά πόσο η πρόβλεψη του μοντέλου είναι σχετική με την τιμή εισόδου στο δίκτυο.

Οι συναρτήσεις ενεργοποίησης χωρίζονται σε δυο κατηγορίες τις γραμμικές και τις μη γραμμικές. Στα νευρωνικά δίκτυα οι μη γραμμικές είναι πιο επιθυμητές, για τον λόγο ότι τα πραγματικά δεδομένα που εμφανίζονται στον κόσμο είναι σε κάθε περίπτωση μη γραμμικά.

Υπάρχουν πάρα πολλές συναρτήσεις ενεργοποίησης, ωστόσο στην συγκεκριμένη εργασία θα γίνει εστίαση στην Ανορθωτική Γραμμική Συνάρτηση (Rectified Linear activation Unit – ReLU) [18].



Εικόνα 5. Γραφική παράσταση συνάρτησης ReLU

Η βαθιά μάθηση εστιάζει, με τη χρήση νευρωνικών δικτύων, στην αυτόματη εξαγωγή προτύπων/οντοτήτων (patterns/features) μέσα από πρωτογενή δεδομένα. Οι αλγόριθμοι βαθιάς μάθησης χρησιμοποιούνται κυρίως επειδή μπορούν να παράγουν κατευθείαν οντότητες σε μια ιεραρχική δομή. Παραδείγματος χάριν για την αναγνώριση κτηρίων μέσα από τις εικόνες μια χρήσιμη οντότητα που παράγεται είναι ο υπολογισμός ακμών, ιεραρχικά τουλάχιστον τέσσερις ακμές ορίζουν ένα κλειστό πολύγωνο που μπορεί να οριστεί ως οικία, αυτές οι οικίες άμα είναι συγκεντρωμένες μαζί αποτελούν τον αστικό ιστό. Αυτές οι οντότητες βοηθούν τους αλγορίθμους της βαθιάς μάθησης να υπολογίσουν την πιθανότητα μια συστάδα από εικονοστοιχεία ή γενικότερα ένα σύνολο δεδομένων εισόδου να ανήκουν σε μια κατηγορία.

Βασικό κεφάλαιο της βαθιάς μάθησης με πληθώρα εφαρμογών στην καθημερινότητα, αποτελούν τα συνελκτικά νευρωνικά δίκτυα (Convolutional neural networks – CNN), στα οποία θα γίνει εστίαση καθώς αποτελούν και το βασικό μοντέλο επίλυσης του προβλήματος, για την παρούσα μελέτη.

3.2 Συνελικτικά νευρωνικά δίκτυα

Τα συνελικτικά νευρωνικά δίκτυα (Convolutional neural networks) [19], ή κοινώς CNN, αποτελούν μια εξειδικευμένη μορφή νευρωνικών δικτύων που έχουν ως κύριο στόχο την επεξεργασία δεδομένων με τοπολογική μορφή καννάβου. Χαρακτηριστικό παράδειγμα είναι οι εικόνες (images), οι οποίες αποτελούν δισδιάστατες (2D) δομές πληροφορίας σε μορφή καννάβου με φατνίο το εικονοστοιχείο.

Τα συνελικτικά νευρωνικά δίκτυα πέρα από την ικανότητά τους να επεξεργάζονται δεδομένα σε μορφή εικόνων έχουν την βασική ιδιότητα να μειώνουν τον όγκο της εικόνας διατηρώντας την καίρια πληροφορία ζωντανή, ενώ ταυτόχρονα αυτή αποτελείται από παραγόμενα χαρακτηριστικά (features).

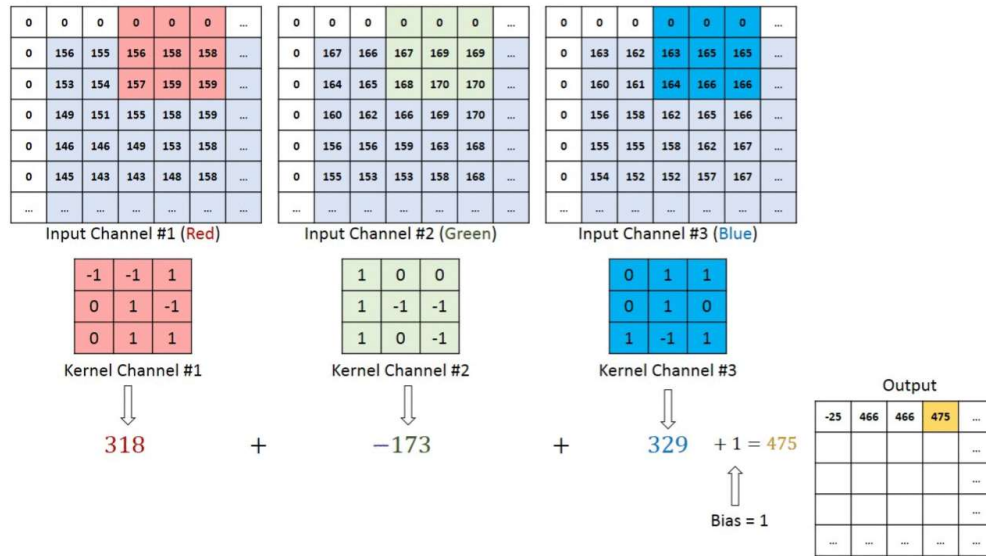
Το όνομα «συνελικτικά νευρωνικά δίκτυα», υποδεικνύει ότι το δίκτυο εφαρμόζει ένα ειδικό είδος γραμμικής πράξης η οποία ονομάζεται συνέλιξη. Ειδικότερα τα συνελικτικά νευρωνικά δίκτυα είναι απλώς νευρωνικά δίκτυα τα οποία χρησιμοποιούν την πράξη της συνέλιξης στη θέση του γενικού πολλαπλασιασμού πινάκων για τουλάχιστον ένα από τα επίπεδά του.

3.2.1 Επίπεδο συνέλιξης (Convolution layer)

Η πράξη της συνέλιξης δέχεται ως είσοδο μια εικόνα η οποία είναι δομημένη σε μορφή τανιστή (τένσορας – tensor) και υλοποιεί μια πράξη με ένα πυρήνα ή αλλιώς φίλτρο (kernel) ίδιου βάθους. Το αποτέλεσμα αυτής της πράξης αποτελεί έναν χάρτη χαρακτηριστικών (feature map).

$$g(x, y) = w * f(x, y) = \sum_{dx=-a}^a \sum_{dy=-b}^b w(dx, dy) f(x + dx, y + dy), \quad (3)$$

Όπου $g(x, y)$, αποτελεί την φιλτραρισμένη εικόνα, $f(x, y)$ την εικόνα εισόδου και w τον πυρήνα. Στο σχήμα 13, ακολουθεί και η γραφική αναπαράσταση της συνέλιξης.



Εικόνα 6. Παράδειγμα πράξης συνέλιξης σε εικόνα τριών καναλιών

Η παραπάνω διαδικασία εφαρμόζεται ολισθαίνοντας το φίλτρο σύμφωνα με ένα χωρικό βήμα (stride) κάθε φορά. Ένας νευρώνας στην συγκεκριμένη περίπτωση αποτελεί κάθε συστάδα εικονοστοιχείων, με διαστάσεις όσες και του φίλτρου για το οποίο η πράξη συνέλιξης βρίσκει εφαρμογή, καθώς η ίδια η πράξη αποτελείται από το άθροισμα ενός εσωτερικού γινομένου, τοποθετώντας στην θέση κάθε στοιχείου του φίλτρου βάρη. Συνεπώς οι παράμετροι ενός επιπέδου συνέλιξης που μπορούν να εκπαιδευτούν αποτελούν τα στοιχεία του πυρήνα του φίλτρου.

Λόγω της μη γραμμικής συμπεριφοράς τέτοιων δεδομένων (εικόνων) είναι απαραίτητη η εφαρμογή μετά από κάθε πράξη συνέλιξης μιας συνάρτησης ενεργοποίησης. Συνήθως στα CNN χρησιμοποιείται η ReLU [18], η οποία μετατρέπει τις αρνητικές τιμές σε μηδέν ενώ στις θετικές λειτουργεί ως ταυτοτική συνάρτηση, δηλαδή παραμένουν οι τιμές ίδιες.

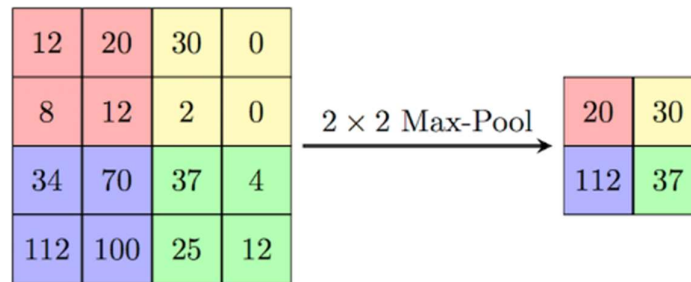
Η ReLU αποτελεί συνήθης επιλογή κατά τη διαδικασία μάθησης καθώς έχει σταθερή παράγωγο για τις θετικές τιμές [18]. Έτσι αντιμετωπίζει το πρόβλημα της εξαφάνισης κλίσης (vanishing gradient problem), το οποίο εμφανίζεται σε αλγορίθμους μάθησης βασισμένους σε μοντέλα κλίσεων (gradient-based learning methods) και οπισθοδιάδοσης (Backpropagation).

3.2.2 Επίπεδο Υποδειγματοληψίας (Pooling layer)

Ένα τυπικό επίπεδο συνελικτικού νευρωνικού δικτύου αποτελείται από τρία στάδια. Στο πρώτο στάδιο εφαρμόζονται πολλαπλές πράξεις συνέλιξης στα δεδομένα εισόδου σε μια παράλληλη διαδικασία. Στο δεύτερο στάδιο κάθε παραγόμενος όγκος περνάει από μια συνάρτηση ενεργοποίησης και συγκεκριμένα την ReLU, με σκοπό την εξάλειψη της μη γραμμικής συμπεριφοράς των δεδομένων. Το τρίτο και

τελευταίο στάδιο αποτελεί το επίπεδο υποδειγματοληψίας. Το συγκεκριμένο επίπεδο έχει ως στόχο την μείωση των διαστάσεων της εικόνας, χωρίς όμως η χρήσιμη πληροφορία να χάνεται. Η μείωση των διαστάσεων έχει ως βασικό σκοπό την μείωση της υπολογιστικής ισχύς που απαιτείται για την επεξεργασία των δεδομένων.

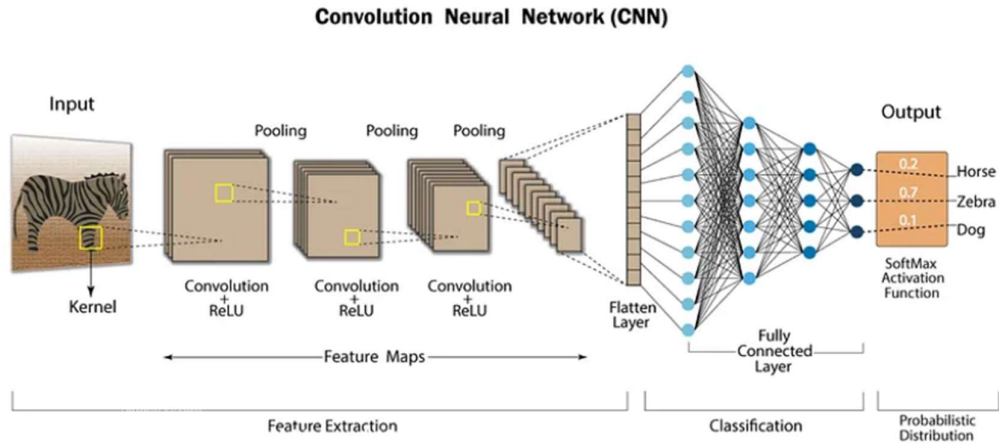
Η τιμή αυτή προκειμένου να διατηρεί την χαρακτηριστική πληροφορία που έχει εξαχθεί από τα δυο προηγούμενα επίπεδα πρέπει να επιλεγεί με βάση μια μαθηματική συνάρτηση. Στην πράξη εφαρμόζεται η μέγιστη τιμή (Max Pooling operation), ωστόσο εφαρμογή βρίσκουν και ο μέσος όρος (Average Pooling operation). Αλγοριθμικά υλοποιείται ένας πυρήνας συνήθως διαστάσεων 2×2 πάνω σε κάθε επίπεδο του παραγόμενου χάρτη χαρακτηριστικών, για τον οποίο επιλέγεται η μέγιστη τιμή. Το παράθυρο ολισθαίνει σύμφωνα με ένα βήμα στο επίπεδο όλης της εικόνας επιλέγοντας κάθε φορά την μέγιστη τιμή.



Εικόνα 7. Παράδειγμα πράξης max pooling σε εικόνα

3.2.3 Πλήρως συνδεδεμένο επίπεδο (Fully Connected Layer)

Έπειτα από την ολοκλήρωση των τριών βασικών σταδίων ενός συνελκτικού επιπέδου είτε εφαρμοσμένα μια φορά, είτε σε κάποια ιεραρχική δομή παράγεται ένας τελικός όγκος χαρακτηριστικών που έχει σχετικά μικρές διαστάσεις, αλλά αναλογικά διαθέτει πολύ μεγάλο αριθμό καναλιών. Σε αυτό το σημείο ο όγκος μετατρέπεται από τριών διαστάσεων πληροφορία σε διάνυσμα και τροφοδοτείται σε ένα πλήρες συνδεδεμένο επίπεδο προκειμένου να γίνει η διαδικασία της ταξινόμησης. Η διαδικασία της μετατροπής των δεδομένων εξόδου, σε δεδομένα εισόδου για το πλήρες συνδεδεμένο επίπεδο ονομάζεται flatten. Περνώντας σε αυτό το στάδιο σημειώνεται πως για κάθε εικόνα είναι γνωστή η ετικέτα της, καθώς η διαδικασία μάθησης είναι επιβλεπόμενη.

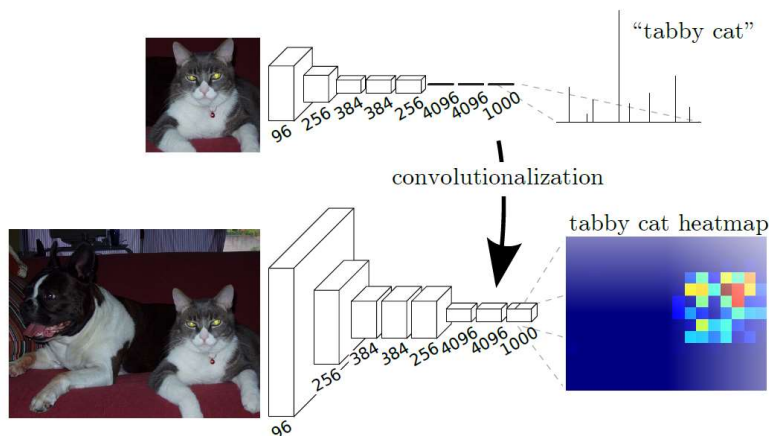


Εικόνα 8. Συνολική αρχιτεκτονική ενός CNN

Η παραπάνω εικόνα αποτελεί μια συνολική παρουσίαση ενός συνελκτικού νευρωνικού δικτύου που έχει ως απώτερο στόχο την ταξινόμηση εικόνων. Κάθε εικόνα που εισέρχεται στο δίκτυο περνάει από την διαδικασία παραγωγής ενός χάρτη ενεργοποίησης σύμφωνα με τα τρία στάδια που αναφέρθηκαν σε αυτό το κεφάλαιο Έπειτα ο παραγόμενος όγκος διανυσματοποιείται και περνάει σε ένα εμπρόσθιας τροφοδοσίας νευρωνικό δίκτυο το οποίο ταξινομεί την εικόνα σύμφωνα με το σύνολο των κλάσεων.

3.3 Πλήρως συνελκτικά νευρωνικά δίκτυα

Αντικαθιστώντας το στάδιο του πλήρως συνδεδεμένου επιπέδου με προσθήκη παραπάνω επιπέδων συνέλιξης και εφαρμόζοντας μια τεχνική πρόβλεψης για κάθε εικονοστοιχείο, οδηγούμαστε σε καινοτόμους αλγορίθμους κατάτμησης Fully Convolutional Networks (FCN) [20].



Εικόνα 9. Σχηματική αναπαράσταση μετατροπής ενός CNN σε FCN

Στην παραπάνω εικόνα φαίνεται ότι αντί η εικόνα να ενταχθεί σε ένα πλήρες συνδεδεμένο επίπεδο, συνεχίζει να τροφοδοτείται σε επίπεδα συνέλιξης. Η διαδικασία της κατάτμησης υλοποιείται σε δυο βήματα. Πρώτα στην εικόνα εφαρμόζεται μια συνέλιξη 1×1 με μέγεθος καναλιού όσες και οι κλάσεις που επιθυμεί ο χρήστης (+1 για το φόντο) να προβλεφθούν για κάθε εικονοστοιχείο. Έπειτα με εφαρμογή ενός επιπέδου υπερδειγματοληψίας, συνήθως της αντίστροφης πράξης συνέλιξης, παράγεται η εικόνα.

3.3.1 Σημασιολογική κατάτμηση

Στην διαδικασία της απλής ταξινόμησης, προβλέπεται για μια εικόνα ποια είναι η πιθανότητα να ανήκει σε ένα σύνολο κλάσεων, σύμφωνα με τις οποίες έχει εκπαιδευτεί ένα σύνολο εικόνων σε ένα συνελικτικό νευρωνικό δίκτυο. Η σημασιολογική κατάτμηση εικόνων, είναι μια διαδικασία που αντί να αποδίδει σε μια ολόκληρη εικόνα μια ετικέτα, ταξινομεί κάθε εικονοστοιχείο της εικόνας, δηλαδή προβλέπει την πιθανότητα κάθε εικονοστοιχείο να αντιστοιχεί σε ένα σύνολο κλάσεων. Έτσι το τελικό προϊόν της σημασιολογικής κατάτμησης είναι η παραγωγή μιας ταξινομημένης εικόνας, δηλαδή μετατρέπει μια εικόνα πολλών διαστάσεων σε μια εικόνα μίας διάστασης, όπου τα εικονοστοιχεία έχουν χρωματιστεί σύμφωνα με την κλάση που ανήκουν.

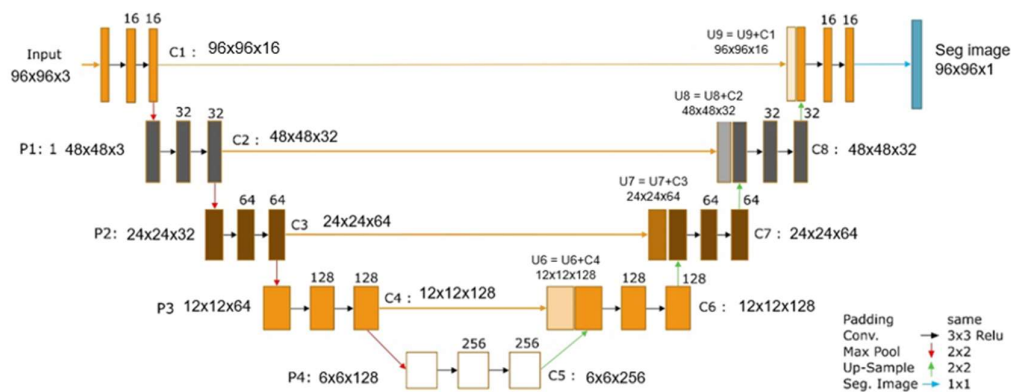
Μια από τις πιο δημοφιλείς προσεγγίσεις της σημασιολογικής κατάτμησης αποτελεί η ανάπτυξη ενός μοντέλου με δυο κλάδους τύπου encoder/decoder. Ο πρώτος κλάδος (encoder) έχει ως στόχο να μειώνει (down-sample) το μέγεθος εισαγωγής των δεδομένων παράγοντας χαμηλής χωρικής ακρίβειας χαρτών χαρακτηριστικών. Αντίθετα, με τον δεύτερο κλάδο (decoder) αυξάνεται η χωρική ανάλυση (up-sample) του παραγόμενου χάρτη χαρακτηριστικών με τελικό στόχο την παραγωγή ενός χάρτη κατάτμησης σε πλήρες μέγεθος. Αξίζει να σημειωθεί πως ο πρώτος κλάδος μιας τέτοιας αρχιτεκτονικής, δεν διαφέρει σε τίποτα από την διαδικασία εξαγωγής χαρακτηριστικών που εκτελεί ένα συνελικτικό νευρωνικό δίκτυο.

Η αντίστροφη συνέλιξη (transposed convolution) συστήνεται ως καλύτερος αλγόριθμος unpooling, καθώς αποτελεί έναν αλγόριθμο που εκτελεί την αντίθετη κατεύθυνση της κανονικής συνέλιξης που έχει επιτευχθεί στα επίπεδα συνέλιξης (encoder). Δηλαδή μια δομή πληροφορίας που έχει την μορφή εξόδου από συνέλιξη να μετατρέπεται σε μια διαφορετική, με το σχήμα της εισόδου, διατηρώντας παράλληλα ένα μοτίβο συνδεσιμότητας, που είναι συμβατό με την εν λόγω συνέλιξη. Βασικό πλεονέκτημα αυτής της μεθόδου υπερδειγματοληψίας, αποτελεί το γεγονός πως τα βάρη της αντίστροφης συνέλιξης δέχονται εκπαίδευση. Γι' αυτό δεν χρειάζεται η εφαρμογή κάποιου προκαθορισμένου αλγορίθμου παρεμβολής [21].

3.3.2 Η αρχιτεκτονική U-Net

Το U-Net είναι ένα πλήρως συνελκτικό νευρωνικού δικτύου [5], που χρησιμοποιείται ευρέως στον τομέα της εικονικής επεξεργασίας και αναγνώρισης εικόνων, ιδιαίτερα στον τομέα της ιατρικής εικόνας. Η αρχιτεκτονική του U-Net προτάθηκε αρχικά από τους Olaf Ronneberger, Philipp Fischer και Thomas Brox το 2015.

Η βασική ιδέα πίσω από το U-Net είναι η σύζευξη μιας πυκνής πορείας (encoder) με μια αντίστροφη πορεία (decoder), όπου η πυκνή πορεία χρησιμοποιείται για την εξαγωγή χαρακτηριστικών από την εικόνα εισόδου και η αντίστροφη πορεία χρησιμοποιείται για την ανακατασκευή μιας εικόνας με υψηλή ανάλυση. Οι συνδέσεις μεταξύ των διαφορετικών επιπέδων του encoder και του decoder επιτρέπουν τη μετάδοση λεπτομερειών κατά την ανακατασκευή της εικόνας. Ακολουθεί μια απλοποιημένη αναλυτική περιγραφή της αρχιτεκτονικής του UNet.



Εικόνα 10. Η αρχιτεκτονική του δικτύου U-Net

Επίπεδα Downsampling (Encoder)

Συνήθως αποτελούνται από συνελκτικά επίπεδα (convolutional layers), ακολουθούμενα από επίπεδα συγκέντρωσης (pooling layers) ή αντίστοιχα. Η κάθε συνελκτική επίστρωση μειώνει την διάσταση της εικόνας, ενώ αυξάνει τον αριθμό των καναλιών (channels) χαρακτηριστικών, χωρίς να χάνεται η χωρική πληροφορία.

Επίπεδο Bottleneck

Αποτελεί ένα σημείο στο οποίο η διάσταση των χαρακτηριστικών είναι πολύ χαμηλή αλλά ο αριθμός των καναλιών είναι υψηλός.

Επίπεδα Upsampling (Decoder)

Παρόμοια με τα επίπεδα downsampling, αλλά χρησιμοποιούν επίστρωση αποσυστάθμισης (upsampling layer) για την αύξηση της διάστασης των

χαρακτηριστικών. Στα επίπεδα αυτά συνήθως προστίθενται πληροφορίες από την παρόδο εκείνης του αντίστοιχου επιπέδου downsampling, χρησιμοποιώντας συνήθως συνένωση (concatenation) ή προσθέτοντας πληροφορίες με μετασχηματισμό (skip connections). Σε τεικό επίπεδο ανασυνδυάζει τις πληροφορίες από τα επίπεδα upsampling για να παράξει το τελικό αποτέλεσμα, το οποίο συνήθως είναι μια εικόνα με την ίδια διάσταση με την είσοδο.

3.4 Τηλεπισκοπικές Αρχές

Ως τηλεπισκόπηση (Remote sensing) ορίζεται η τεχνική απόκτησης πληροφοριών για αντικείμενα που βρίσκονται στη γήινη επιφάνεια, μέσα από την ανάλυση δεδομένων που συλλέγονται από ειδικά όργανα (δορυφορικοί δέκτες, πολυφασματικές κάμερες σε μη επανδρωμένα αεροσκάφη) τα οποία όμως δεν έχουν φυσική επαφή με τα αντικείμενα. Έτσι, η τηλεπισκόπηση μπορεί να αποδοθεί και ως η αναγνώριση ενός αντικειμένου από απόσταση. Η τηλεπισκόπηση έχει χαρακτηριστεί και ως τέχνη, καθώς ο συνδυασμός της καταγραφόμενης πληροφορίας με σκοπό την ερμηνεία και την απόκτηση γνώσης αναμφίβολα χρειάζεται μελέτη, πειραματικές διαδικασίες αλλά και έμπνευση.

3.4.1 Δορυφορικοί Δέκτες

Ο δέκτης είναι μια συσκευή που μπορεί να εντοπίζει τα ηλεκτρομαγνητικά κύματα που ανακλώνται ή εκπέμπονται από ένα αντικείμενο. Ανάλογα με την πηγή που προέρχεται η ηλεκτρομαγνητικής (ΗΜ) ακτινοβολία διακρίνονται δύο είδη δεκτών/αισθητήρων, ο παθητικός και ο ενεργητικός. Ένα από τα σημαντικότερα πλεονεκτήματα των δορυφορικών δεκτών είναι ότι επιτυγχάνουν τη θέαση από μακριά (bird's eye view) [22]. Το γεγονός αυτό τους επιτρέπει να «βλέπουν» μεγάλες περιοχές της γης ταυτόχρονα και έτσι μπορούν να συλλέγουν περισσότερα δεδομένα και με μεγαλύτερη ταχύτητα από τα επίγεια όργανα.

Οι δορυφόροι παρέχουν πληροφορίες για τα σύννεφα, τους ωκεανούς, τους πάγους, τα αέρια στην ατμόσφαιρα (π.χ. όζον και διοξείδιο του άνθρακα) και την ποσότητα ενέργειας που απορροφά και εκπέμπει η γη. Οι δορυφόροι παρακολούθησης της γης, όπως η σειρά Landsat και Sentinel χρησιμοποιούνται επίσης για την παρακολούθηση πυρκαγιών, ηφαιστειών και του καπνού που παράγουν αυτά και γενικότερα αλλαγών των δασών, του νερού και άλλων μερών της γήινης επιφάνειας με την πάροδο του χρόνου.

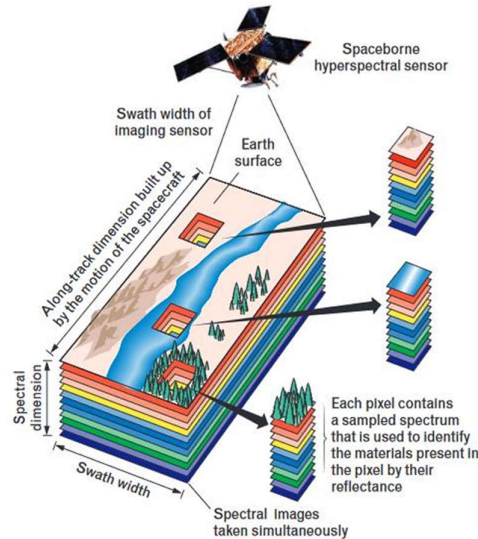
Για την εκπόνηση της παρούσας μελέτης το σύνολο δεδομένων εκπαίδευσης (dataset) που χρησιμοποιήθηκαν είναι εικόνες οκτώ καναλιών από τον δορυφορικό δέκτη Worldview-2 της MAXAR [23]. Ο εν λόγω πολυφασματικός δέκτης έχει τα ακόλουθα κανάλια:

- Coastal Band
- Blue
- Green
- Yellow
- Red
- Red Edge
- Near Infrared 1
- Near Infrared 2

Αναλυτικότερα θα ασχοληθούμε με τα προϊόντα του στο κεφάλαιο διαχείρισης δεδομένων (κεφάλαιο 5) της παρούσας.

3.4.2 Φασματικά κανάλια

Με τον όρο «φασματικό κανάλι» [24] νοείται μια περιοχή μήκους κύματος στο φάσμα της ανακλώμενης ή εκπεμπόμενης ΗΜ ενέργειας στην οποία είναι ευαίσθητος ένας απομακρυσμένος αισθητήρας. Η ΗΜ ακτινοβολία έχει ένα τεράστιο εύρος ενέργειας που αποτελείται από διαφορετικά μήκη κύματος. Μόλις ένας αισθητήρας συλλέξει δεδομένα από μια φασματική ζώνη, αποθηκεύει τα δεδομένα σε ένα αρχείο ή τμήμα αρχείου δηλαδή σε ένα κανάλι. Έτσι, μια εικόνα είναι μια μορφή επικάλυψης διαφορετικών καναλιών.



Εικόνα 11. Η έννοια της φασματοσκοπικής απεικόνισης.

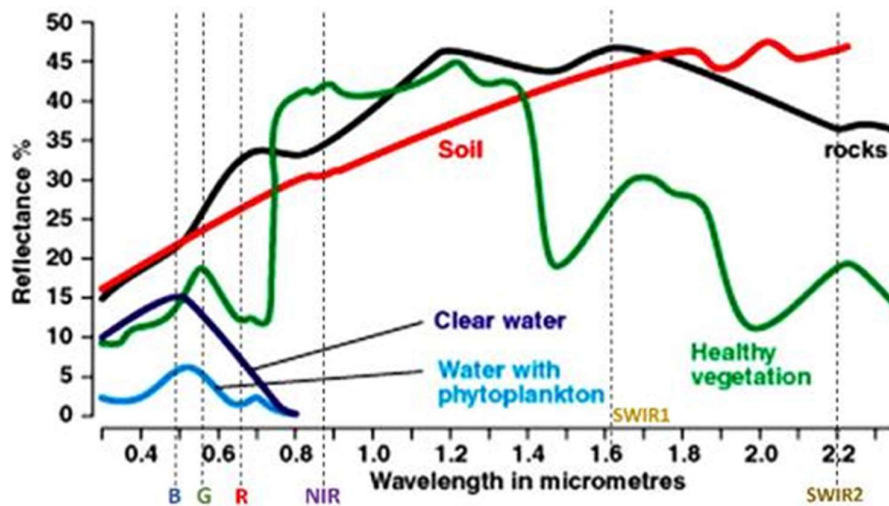
(Πηγή: Gary A. Shaw & Hsiao – Hua K. Burke, “Spectral Imaging for Remote Sensing”)

Στην τηλεπισκόπηση, ένα κανάλι αναφέρεται σε ένα συγκεκριμένο εύρος μήκους κύματος του ΗΜ φάσματος που χρησιμοποιείται για την καταγραφή και τη μέτρηση της ανακλαστικότητας ή της ακτινοβολίας ενός αντικειμένου ή μιας επιφάνειας από έναν αισθητήρα. Οι δορυφορικοί δέκτες, διαθέτουν συνήθως πολλαπλές ζώνες/κανάλια που μπορούν να χρησιμοποιηθούν για τη απόκτηση διαφορετικών

τύπων πληροφοριών σχετικά με την επιφάνεια της γης, αυτοί χαρακτηρίζονται ως πολυφασματικοί (multispectral) δέκτες.

3.4.3 Φασματικές υπογραφές

Η φασματική υπογραφή αναφέρεται στη σχηματική αναπαράσταση της ποσότητας της ΗΜ ακτινοβολίας που καταγράφει ένας αισθητήρας του δορυφόρου από μια συγκεκριμένη κάλυψη γης. Η ποσότητα αυτή αντιστοιχεί σε κάποια ραδιομετρική τιμή. Παρόλα αυτά, κάθε εικονοστοιχείο της ίδιας κατηγορίας δεν παρουσιάζει απαραίτητως την ίδια ραδιομετρική τιμή στο ίδιο φασματικό κανάλι. Οι τιμές όμως είναι κοντινές εάν η μελετώμενη κάλυψη γης είναι ομοιογενής. Έτσι, το παρακάτω διάγραμμα έχει προκύψει βάσει των μέσων όρων των τιμών όσο το δυνατόν μεγαλύτερης συγκέντρωσης εικονοστοιχείων της ίδιας κατηγορίας.



Εικόνα 12. Φασματικές υπογραφές κοινών υλικών
(Πηγή: Ψηφιακή Τηλεπισκόπηση – ΕΜΠ, 2021)

Παρατηρούμε ότι τα δομικά υλικά (soil, rocks) ανακλούν περισσότερο στο ορατό φάσμα και ιδιαίτερα στο κόκκινο, σε αντίθεση με τη βλάστηση που στο ορατό φάσμα έχει χαμηλότερη ανακλαστικότητα. Αντίστροφα στην περιοχή του εγγύς υπέρυθρου η βλάστηση ανακλά πιο έντονα, συνεπώς ανιχνεύεται και ευκολότερα σε σύγκριση με τα δομικά υλικά.

3.4.4 Επιθέσεις - Πράξεις καναλιών

Για τον εντοπισμό χαρακτηριστικών και κατ' επέκταση την αναγνώριση αντικειμένων βοηθά η επίθεση φασματικών καναλιών με σκοπό τη δημιουργία έγχρωμων σύνθετων ή η υλοποίηση πράξεων μεταξύ καναλιών με σκοπό τη δημιουργία λόγων ή δεικτών.

Για την επίτευξη οπτικής αντίληψης μη ορατών καταγεγραμμένων ΗΜ ακτινοβολιών χρησιμοποιούνται ψευδοχρώματα και με κατάλληλες επιθέσεις εντοπίζονται και ταξινομούνται παρατηρούμενα αντικείμενα. Χαρακτηριστικό παράδειγμα που χρησιμοποιήσαμε και στην παρούσα μελέτη, πέραν του ορατού έγχρωμου σύνθετου (red-green-blue), είναι το (nir2-yellow-blue) των καναλιών του δορυφορικού πολυφασματικού δέκτη world view 2. Το εν λόγω σύνθετο χρησιμοποιείται για τον εντοπισμού τσιμέντου, καθώς σε αυτό ανακλά περισσότερο.

Οι λόγοι καναλιών είναι μία από τις πιο χρήσιμες αριθμητικές πράξεις που μπορούν να εφαρμοστούν αφού με την κατάλληλη επιλογή καναλιών συμβάλουν στην ανάδειξη συγκεκριμένων αντικειμένων και θεματικών κατηγοριών. Πέραν των λόγων, μπορούν να πραγματοποιηθούν πιο σύνθετες πράξεις όπως είναι οι δείκτες (indexes) εδαφοκάλυψης. Οι δείκτες ανήκουν στην κατηγορία των τεχνικών φασματικής ενίσχυσης (spectral enhancement) των εικόνων. Ουσιαστικά, είναι εικόνες οι οποίες προκύπτουν μετά από μαθηματικές πράξεις μεταξύ των φασματικών καναλιών της ίδιας εικόνας ή διαφορετικών εικόνων. Οι δείκτες καναλιών είναι πολλές φορές προτιμότεροι έναντι των λόγων καθώς οι πρώτοι είναι συνήθως κανονικοποιημένοι και έχουν καθορισμένο εύρος (-1,1). Ο πιο γνωστός δείκτης για την ανάδειξη της βλάστησης είναι ο NDVI (Normalized Difference Vegetation Index). Πρόκειται για έναν κανονικοποιημένο δείκτη που προκύπτει από το λόγο ανάκλασης του κόκκινου καναλιού προς το εγγύς υπέρυθρο και υπολογίζεται ως εξής:

$$NDVI = \frac{Nir - Red}{Nir + Red} \quad (4)$$

Οι τιμές του NDVI κυμαίνονται από -1 έως +1 με τις θετικές τιμές να υποδηλώνουν την ύπαρξη βλάστησης. Συγκεκριμένα, οι τιμές 0,1 – 0,5 περιγράφουν την αραιή βλάστηση και οι τιμές > 0,6 την πυκνή πράσινη βλάστηση. Οι αρνητικές τιμές και η τιμή 0 εκφράζουν την απουσία βλάστησης και την ύπαρξη άλλων καλύψεων γης.

3.4.5 Εξειδικευμένα έγχρωμα σύνθετα

Στην αναζήτηση της βιβλιογραφίας, για την επιλογή βέλτιστων σεναρίων εκπαίδευσης του δικτύου, εντοπίσθηκε έγχρωμο ένα σύνθετο ονομαζόμενο ως «**cement**» δηλαδή τσιμέντο. Δημιουργείται από τον συνδυασμό τριών καναλιών του δορυφορικού δέκτη Worldview-2 και συγκεκριμένα τα κανάλια Εγγύς υπέρυθρο 2 (NIR2), Κίτρινο (yellow) και Μπλε (blue).

Επιλέχθηκε η δημιουργία εικόνων βάση του συγκεκριμένου σύνθετου, ως δεδομένα εκπαίδευσης στο δίκτυο, επειδή υπάρχει μεγαλύτερη διάκριση με αυτό στις εικόνες, όπου υπάρχει τσιμέντο (π.χ. βαμμένες λωρίδες και αυτοκίνητα) σε σύγκριση με άλλα πιθανά σύνθετα.

Πίνακας 1. Προτεινόμενα έγχρωμα σύνθετα Worldview-2

A/A	FEATURE	R	G	B
1	Vegetation	Nir2	Yellow	Coastal
2	Shadow	Nir2	Red Edge	Yellow
3	Cement	Nir2	Yellow	Blue
4	Sedimentation	Red	Yellow	Coastal
5	Mud flats	Red Edge	Yellow	Coastal
6	Red roofs	Red	Yellow	Green
7	Water depth	Yellow	Green	Coastal

Βέβαια πέραν του «cement» αναφέρονται και άλλα έγχρωμα σύνθετα, κατάλληλα για ανίχνευση λοιπών αντικειμένων, λόγω ανακλαστικών ιδιοτήτων. Ακολουθεί σχετικός πίνακας.

ΚΕΦΑΛΑΙΟ 4

Προγραμματισμός διαδικασιών σε Python

4. Εισαγωγικά

Στο παρόν κεφάλαιο παρουσιάζονται και αναλύονται η γλώσσα προγραμματισμού Python, οι βιβλιοθήκες που χρησιμοποιήθηκαν και το περιβάλλον στο οποίο έγινε η δημιουργία και η προσπέλαση των αλγορίθμων (Google Colab). Επιπρόσθετα αναλύεται η αρχιτεκτονική του πλήρους συνελκτικού δικτύου U-Net, που χρησιμοποιήθηκε για την εκπόνηση της παρούσας.

4.1 Η Python ως γλώσσα προγραμματισμού

Η γλώσσα προγραμματισμού Python χρησιμοποιήθηκε για την υλοποίηση του τεχνικού τμήματος της παρούσας μελέτης, καθώς με τη χρήση αυτής προγραμματίστηκαν το πλήρως συνελκτικό νευρωνικό δίκτυο (U-Net), το οποίο προσπελάστηκε σε πέντε διαφορετικά σενάρια εκ των οποίων επιλέχθηκε το βέλτιστο.

Η Python είναι μια διερμηνευμένη, διαδραστική, αντικειμενοστραφής γλώσσα προγραμματισμού [25]. Παρέχει δομές δεδομένων υψηλού επιπέδου και αυτό την καθιστά προσιτή και ευνόητη. Διαθέτει ένα εξαιρετικά απλό και κομψό συντακτικό και είναι όμως μια ισχυρή και γενικού σκοπού γλώσσα προγραμματισμού. Σχεδιάστηκε το 1990 από τον Guido van Rossum.

Διακρίνεται λόγω του ότι έχει πολλές βιβλιοθήκες που διευκολύνουν ιδιαίτερα αρκετές συνηθισμένες εργασίες και για την ταχύτητα εκμάθησης της. Μειονεκτεί όμως, στο γεγονός ότι επειδή είναι διερμηνευόμενη είναι πιο αργή από τις μεταγλωττιζόμενες (compiled) γλώσσες, όπως η C και η C++. Για αυτόν τον λόγο δεν είναι κατάλληλη για γραφή λειτουργικών συστημάτων.

Όπως και πολλές άλλες γλώσσες σεναρίων είναι δωρεάν, ακόμη και για εμπορικούς σκοπούς, και μπορεί να εκτελεστεί σε πρακτικά οποιονδήποτε σύγχρονο υπολογιστή. Ένα πρόγραμμα python μεταγλωττίζεται αυτόματα από τον διερμηνέα σε κώδικα byte ανεξάρτητο από την πλατφόρμα, ο οποίος στη συνέχεια διερμηνεύεται. Εκτελούμε μη τροποποιημένα στοιχεία γραμμένα σε Python κάτω από linux, Windows NT, 98, 95, IRIX, SunOS, OSF.

Το περιβάλλον που χρησιμοποιήθηκε για την εκμετάλλευση των δυνατοτήτων της γλώσσας προγραμματισμού Python είναι το Google Colab.

4.2 Google Colab

Το Colaborator [26], ή εν συντομία "Colab", είναι ένα προϊόν της Google Research. Το Colab επιτρέπει σε οποιονδήποτε να γράφει και να εκτελεί αυθαίρετο κώδικα Python μέσω του προγράμματος περιήγησης και είναι ιδιαίτερα κατάλληλο για μηχανική μάθηση, ανάλυση δεδομένων και εκπαίδευση. Πιο τεχνικά, το Colab είναι μια φιλοξενούμενη υπηρεσία σημειωματάριου Jupyter που δεν απαιτεί καμία εγκατάσταση για να χρησιμοποιηθεί, ενώ παρέχει δωρεάν πρόσβαση σε υπολογιστικούς πόρους, συμπεριλαμβανομένων των GPU.

Επιπροσθέτως, το Google Colab έχει ήδη εγκατεστημένες σχεδόν όλες τις απαραίτητες βιβλιοθήκες, οι οποίες είναι διαθέσιμες ανά πάσα στιγμή. Ένα ιδιαίτερο χαρακτηριστικό του προγράμματος είναι πως πραγματοποιεί αυτόματη αποθήκευση στο Cloud και έτσι δεν υπάρχει κίνδυνος να καταστραφεί ο αλγόριθμος που έχει ήδη υλοποιηθεί. Ακόμα, το Google Colab, επιτρέπει τη συνεργασία μεταξύ των προγραμματιστών για μία κοινή εργασία.

4.3 Βιβλιοθήκες που χρησιμοποιήθηκαν

Το TensorFlow [27], είναι μια ολοκληρωμένη πλατφόρμα ανοιχτού κώδικα για μηχανική μάθηση. Διαθέτει ένα ολοκληρωμένο, ευέλικτο σύστημα εργαλείων, βιβλιοθηκών και πόρων που επιτρέπει στους ερευνητές να προωθήσουν την τελευταία λέξη της τεχνολογίας στη μηχανική μάθηση και οι προγραμματιστές να κατασκευάζουν και να αναπτύσσουν εύκολα εφαρμογές με υποστήριξη μηχανικής μάθησης.

Το Keras [28], είναι ένα API βαθιάς μάθησης γραμμένο σε γλώσσα προγραμματισμού Python, που τρέχει πάνω στην πλατφόρμα μηχανικής μάθησης TensorFlow. Αναπτύχθηκε με έμφαση στη δυνατότητα γρήγορης δημιουργίας αλγορίθμων και πειραματισμού αυτών. Με βασικό στόχο, να είσαι σε θέση να πηγαίνει από την ιδέα στο αποτέλεσμα όσο το δυνατόν γρηγορότερα, ένας στόχος που αποτελεί καίριος στην σωστή και ορθολογική έρευνα.

Η βιβλιοθήκη Pandas, [29] αποτελεί μια βιβλιοθήκη της Python η οποία προσφέρει δομές δεδομένων και λειτουργίες για το χειρισμό αριθμητικών πινάκων και χρονοσειρών. Είναι δωρεάν λογισμικό που κυκλοφορεί με την άδεια BSD τριών ρητρών. Το όνομα προέρχεται από τον όρο "Panel Data", ένας όρος οικονομετρίας για σύνολα δεδομένων που περιλαμβάνουν παρατηρήσεις σε πολλαπλές χρονικές περιόδους για τα ίδια άτομα. Υπάρχει η δυνατότητα εισαγωγής δεδομένων από διάφορες μορφές αρχείων, όπως CSV, JSON, πίνακες ή ερωτήματα βάσεων δεδομένων SQL και Microsoft Excel. Επιτρέπει διάφορες λειτουργίες χειρισμού δεδομένων, όπως συγχώνευση, αναμόρφωση, επιλογή, καθώς και λειτουργίες καθαρισμού δεδομένων και σύγχυσης δεδομένων.

Η NumPy, αποτελεί μια βιβλιοθήκη της Python η οποία υλοποιήθηκε με στόχο την επεξεργασία και την διαχείριση μαθηματικών προβλημάτων μέσω της δομής

πινάκων [30]. Η NumPy βρίσκει πληθώρα εφαρμογών σε προβλήματα γραμμικής άλγεβρας, μετασχηματισμού Fourier και υλοποίηση πράξεων μεταξύ πινάκων. Η NumPy δημιουργήθηκε το 2005 από τον Travis Oliphant και είναι ένα εργαλείο ανοιχτού κώδικα που μπορεί να χρησιμοποιηθεί ελεύθερα. Ο λόγος που χρησιμοποιείται η συγκεκριμένη βιβλιοθήκη είναι γιατί στην Python υπάρχουν οι λίστες που εξυπηρετούν το σκοπό των πινάκων, αλλά είναι αργές στην επεξεργασία τους. Η NumPy στοχεύει στην παροχή ενός πίνακα που για οποιαδήποτε πράξη είναι αρκετά πιο γρήγορος από μια παραδοσιακή λίστα.

Η βιβλιοθήκη Rasterio [31] της Python εξασφαλίζει πρόσβαση σε ψηφιδωτά γεωχωρικά (raster data). Τα συστήματα γεωγραφικών πληροφοριών χρησιμοποιούν το GeoTIFF και άλλες μορφές για την οργάνωση και αποθήκευση συνόλων δεδομένων raster με πλέγμα, όπως δορυφορικές εικόνες και μοντέλα εδάφους. Το Rasterio διαβάζει και γράφει αυτές τις μορφές και παρέχει ένα API Python που βασίζεται σε NumPy N-διάστατους πίνακες και GeoJSON.

Τέλος η βιβλιοθήκη Matplotlib είναι μία βιβλιοθήκη της Python που έχει σχεδιαστεί για την παραγωγή διαγραμμάτων. Παρέχει αντικειμενοστραφή API, για ενσωματωμένα γραφήματα σε εφαρμογές χρησιμοποιώντας τις εργαλειοθήκες γενικής χρήσης του GUI. Η Matplotlib δημιουργήθηκε από τον John D. Hunter και έκτοτε έχει ενεργή κοινότητα ανάπτυξης και είναι διαθέσιμη με την άδεια ανοιχτού λογισμικού BSD. Στην εργασία, χρησιμοποιείται με σκοπό να αποδοθούν τα απαραίτητα γραφήματα.

4.4 Υλοποίηση σε Python

Στο παρόν υποκεφάλαιο αναλύεται τμηματικά το σύνολο του κώδικα που συντέθηκε για την εκπόνηση της μελέτης. Συνοπτικά θα αναφέρουμε ότι, μετά την προετοιμασία των δεδομένων, όπως θα αναλύσουμε στο επόμενο κεφάλαιο, αυτά εισήχθησαν στα μοντέλα για την εκπαίδευση και την αξιολόγησή τους, με σκοπό την ανάδειξη του βέλτιστου σεναρίου πρόβλεψης.

Αρχικά εισάγουμε τις βιβλιοθήκες που χρειάστηκαν για την κλήση των κατάλληλων αλγορίθμων.

```
import tensorflow as tf
import keras
import numpy as np
import cv2
import glob
from datetime import datetime
```

Εικόνα 13.Εισαγωγή βιβλιοθηκών

Αφού οι εικόνες έχουν μετασχηματιστεί σε επιθυμητές διαστάσεις και αφού μέσω μιας αλγοριθμικής διαδικασίας έχουν μετατραπεί σε μορφή πινάκων numpy (4D tensor), το ίδιο γίνεται και για τις μάσκες ώστε από διανυσματικά δεδομένα να

μπορούν να εισαχθούν στο δίκτυο ως ψηφιδωτά δεδομένα. Επόμενο βήμα αποτελεί η αρχικοποίηση των παραμέτρων εισαγωγής στο δίκτυο.

```
# me to rasterio ανοικσε thn eikona
with rasterio.open(trainingxfile) as src:
    # pare ena parathyro 96x96
    img = src.read(window=Window(0, 0, IMG_WIDTH, IMG_HEIGHT))

# apomonose ta kanalia |
red = bandeq(img[4])
green = bandeq(img[2])
blue = bandeq(img[1])
# nir = bandeq(img[6]) # ayto to kleineis kai to anoigeis gia na ftiakseis ta senaria einai to NIR
```

Εικόνα 14. Εντολή αρχικοποίησης παραμέτρων εισαγωγής

Πέραν του παραθύρου διαστάσεων της εικόνας, καθορίζεται και το έγχρωμο σύνθετο για το ανάλογο σενάριο, όπως αυτά αναλύονται στο ακόλουθο κεφάλαιο. Με αυτό τον τόπο δημιουργείται το διάνυσμα X-train και αντίστοιχα το Y-train για τις μάσκες εντοπισμού των αντικειμένων.

```
# Downsampling layers
# DownRes 1, convolution + pooling
conv_128 = conv_block(inputs, FILTER_SIZE, FILTER_NUM, dropout_rate, batch_norm)
pool_64 = layers.MaxPooling2D(pool_size=(2,2))(conv_128)
# DownRes 2
conv_64 = conv_block(pool_64, FILTER_SIZE, 2*FILTER_NUM, dropout_rate, batch_norm)
pool_32 = layers.MaxPooling2D(pool_size=(2,2))(conv_64)
# DownRes 3
conv_32 = conv_block(pool_32, FILTER_SIZE, 4*FILTER_NUM, dropout_rate, batch_norm)
pool_16 = layers.MaxPooling2D(pool_size=(2,2))(conv_32)
# DownRes 4
conv_16 = conv_block(pool_16, FILTER_SIZE, 8*FILTER_NUM, dropout_rate, batch_norm)
pool_8 = layers.MaxPooling2D(pool_size=(2,2))(conv_16)
# DownRes 5, convolution only
conv_8 = conv_block(pool_8, FILTER_SIZE, 16*FILTER_NUM, dropout_rate, batch_norm)
```

Εικόνα 15. Κλάδος Down-sampling της αρχιτεκτονικής του δικτύου U-Net.

Με την ολοκλήρωση της δημιουργίας διανυσμάτων εκπαίδευσης ξεκινάει η εκπαίδευση του μοντέλου. Ο κλάδος down-sampling (Εικόνα 15) αποτελείται από συνελκτικά επίπεδα (convolutional layers). Η συνάρτηση Conv2D η οποία εκτελεί πράξη συνέλιξης με το πεδίο s που υλοποιήθηκε πιο πάνω. Τα ορίσματα της συνάρτησης αφορούν τον αριθμό των πυρήνων (16), το μέγεθος των πυρήνων (3×3), την συνάρτηση ενεργοποίησης η οποία είναι η ReLU, την μέθοδο αρχικοποίησης των βαρών που ορίζουν το κάθε φίλτρο και τέλος την μέθοδο padding που ορίζει την τιμή 0 σαν ένα πλαίσιο σε κάθε εικόνα προκειμένου να διατηρείται το σύνολο των pixels του. Η συγκεκριμένη συνάρτηση παράγει ως αποτέλεσμα 16 εικόνες, που αποθηκεύονται ως κανάλια, δηλαδή για κάθε εικόνα το μέγεθος tensor είναι $96 \times 96 \times 16$. Το κάθε επίπεδο κλείνει ακολουθούμενα από επίπεδα συγκέντρωσης (pooling layers) MaxPooling2D του Keras.layers η οποία έχει ως στόχο την υποδειγματοληψία ορίζοντας ένα φίλτρο 2×2 . Διαπιστώνουμε ότι κάθε συνελκτική επίστρωση μειώνει την διάσταση της εικόνας, ενώ αυξάνει τον αριθμό των καναλιών (channels) χαρακτηριστικών, χωρίς να χάνεται η χωρική πληροφορία.

Ακολουθεί ο up-sampling κλάδος του δικτύου, όπου γίνεται η αντίστροφη διαδικασία (Conv2DTranspose), ενώ αξίζει να σημειωθεί ότι οι δύο κλάδοι συνδέονται συμμετρικά και πληροφορία από τον decoder τροφοδοτεί τον encoder μέσω του «concatenate». Το δίκτυο «ανεβάζει» την εικόνα τόσα επίπεδα όσα και στον προηγούμενο κλάδο.

```
# Upsampling layers

up_16 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE), data_format="channels_last")(conv_8)
up_16 = layers.concatenate([up_16, conv_16], axis=3)
up_conv_16 = conv_block(up_16, FILTER_SIZE, 8*FILTER_NUM, dropout_rate, batch_norm)
# UpRes 7

up_32 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE), data_format="channels_last")(up_conv_16)
up_32 = layers.concatenate([up_32, conv_32], axis=3)
up_conv_32 = conv_block(up_32, FILTER_SIZE, 4*FILTER_NUM, dropout_rate, batch_norm)
# UpRes 8

up_64 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE), data_format="channels_last")(up_conv_32)
up_64 = layers.concatenate([up_64, conv_64], axis=3)
up_conv_64 = conv_block(up_64, FILTER_SIZE, 2*FILTER_NUM, dropout_rate, batch_norm)
# UpRes 9

up_128 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE), data_format="channels_last")(up_conv_64)
up_128 = layers.concatenate([up_128, conv_128], axis=3)
up_conv_128 = conv_block(up_128, FILTER_SIZE, FILTER_NUM, dropout_rate, batch_norm)

# 1*1 convolutional layers

conv_final = layers.Conv2D(NUM_CLASSES, kernel_size=(1,1))(up_conv_128)
conv_final = layers.BatchNormalization(axis=3)(conv_final)
conv_final = layers.Activation('sigmoid')(conv_final) #Change to softmax for multichannel
```

Εικόνα 16. Κλάδος Down-sampling της αρχιτεκτονικής του δικτύου U-Net.

Με την ολοκλήρωση της εκπαίδευσης του μοντέλου, καταγράφονται τα αποτελέσματα των μετρικών δεικτών που έχουν επιλεγεί για την αξιολόγηση εκπαίδευσης του δικτύου. Μέσω των αρχείων μορφότυπου csv που δημιουργήθηκαν, αξιολογήθηκαν τα αποτελέσματα στο Κεφάλαιο 6.

```
[ ] import pandas as pd
# Creating a Pandas DataFrame
data = {'accuracy': acc, 'precision': precision, 'recall': recall, 'f1': f1, 'jac': jac}
print(data)
df = pd.DataFrame(data, columns=['accuracy', 'precision', 'recall', 'f1', 'jac' ])

df.to_csv('/content/drive/MyDrive/dataset/csv_results/rgb_unet_results.csv')
df.describe().to_csv('/content/drive/MyDrive/dataset/csv_results/rgb_unet_results_describe.csv')
df.describe()
```

Εικόνα 17. Καταγραφή στατιστικών αποτελεσμάτων.

Τέλος γίνεται οπτικοποίηση των αποτελεσμάτων πρόβλεψης σε σύγκριση με τα δεδομένα εισόδου, ώστε να είναι δυνατή και η οπτική σύγκριση των σεναρίων. Επίσης υπολογίζονται και γραφήματα Training & Validation Accuracy – Loss για την διαπίστωση καλής ή όχι προσαρμογής του δικτύου με τα δεδομένα εκπαίδευσης (μέτρα αξιολόγησης – Κεφάλαιο 6).

ΚΕΦΑΛΑΙΟ 5

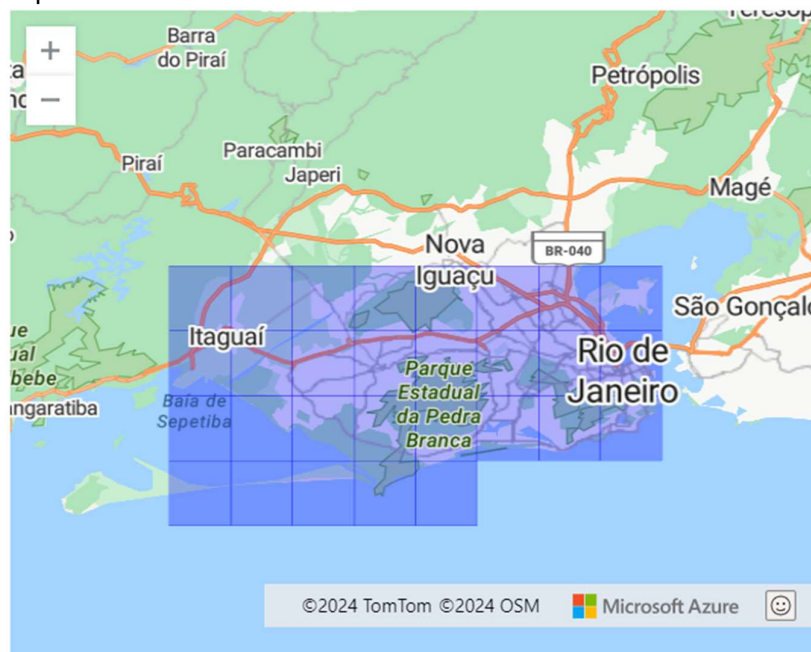
Διαχείριση δεδομένων

5 Εισαγωγή

Στο παρόν κεφάλαιο γνωστοποιείται η πηγή άντλησης των δεδομένων, καθώς και οι διαδικασίες διαχείρισης αυτών για την αξιοποίησή τους, τόσο δεδομένων εκπαίδευσης του πλήρους συνελκτικού νευρωνικού δικτύου (U-net) όσο και δεδομένων ελέγχου αυτού.

5.1 Πηγές δεδομένων

Το σύνολο των δεδομένων για την εκπόνηση της παρούσας εργασίας αντλήθηκαν από την ιστοσελίδα της SpaceNet, με χρήση της πλατφόρμας Amazon Web Services (AWS). Αντλήθηκε μωσαϊκό ψηφιδωτών δεδομένων (raster mosaic) οκτώ καναλιών εδαφοψηφίδας 50cm και τριών καναλιών εδαφοψηφίδας 1m, με πραγματική έκταση κάλυψης 2784 km², στην πόλη της Βραζιλίας Ρίο ντε Τζανέιρο. Το εν λόγω μωσαϊκό αποτελεί προϊόν συλλογής δεδομένων από τον δορυφορικό δέκτη WorldView-2. Επιπρόσθετα αντλήθηκαν και διανυσματικά δεδομένα σε μορφότυπο GeoJSON, με την χωρική πληροφορία των κτιρίων στην περιοχή μελέτης ώστε να είναι δυνατή η εκπαίδευση του δικτύου.



Εικόνα 18. Περιοχή κάλυψης μωσαϊκού ψηφιδωτών δεδομένων.

5.1.1 SpaceNet

Η SpaceNet [32] ιδρύθηκε από την CosmiQ Works της IQT Labs και τη Maxar (τότε DigitalGlobe) τον Αύγουστο του 2016. Αυτή η άτυπη συνεργασία αποσκοπούσε στην επιτάχυνση των δυνατοτήτων μηχανικής μάθησης ανοικτού κώδικα ειδικά για περιπτώσεις γεωχωρικής χρήσης, προσφέροντας ένα αποθετήριο ελεύθερα διαθέσιμων εικόνων με συν-καταχωρημένα χαρακτηριστικά χαρτών. Πριν από το SpaceNet, οι ερευνητές της όρασης υπολογιστών είχαν ελάχιστες επιλογές για να αποκτήσουν δωρεάν, επισημασμένες με ακρίβεια και υψηλής ανάλυσης δορυφορικές εικόνες. Σήμερα, το SpaceNet φιλοξενεί σύνολα δεδομένων που αναπτύχθηκαν από τη δική του ομάδα, μαζί με σύνολα δεδομένων από έργα όπως το Functional Map of the World (fMoW) της IARPA.

5.1.2 Amazon Web Services (AWS)

Η Amazon Web Services (AWS) [32] είναι μια ασφαλής πλατφόρμα υπηρεσιών cloud, η οποία προσφέρει υπολογιστική ισχύ, αποθήκευση σε βάσεις δεδομένων, διανομή περιεχομένου και άλλες λειτουργίες που βοηθούν τις επιχειρήσεις να κλιμακώνονται και να αναπτύσσονται. Έτσι, εκατομμύρια πελάτες αξιοποιούν σήμερα τα προϊόντα και τις λύσεις cloud της AWS για την ανάπτυξη εξελιγμένων εφαρμογών με αυξημένη ευελιξία, επεκτασιμότητα και αξιοπιστία.

5.1.3 WorldView-2

Ο δορυφορικός αισθητήρας WorldView-2 της MAXAR [23] εκτοξεύτηκε στις 8 Οκτωβρίου 2009. Ο αισθητήρας WorldView-2 παρέχει μια παγχρωματική ζώνη υψηλής ανάλυσης και οκτώ (8) πολυφασματικές ζώνες- τέσσερα (4) τυπικά χρώματα (κόκκινο, πράσινο, μπλε και εγγύς υπέρυθρο 1) και τέσσερις (4) νέες ζώνες (πράσινο, κίτρινη, κόκκινη άκρη και εγγύς υπέρυθρο 2), εικόνες πλήρους χρώματος για βελτιωμένη φασματική ανάλυση, χαρτογράφηση ορυκτών πόρων, παρακολούθηση άγριας ζωής, σχεδιασμό χρήσεων γης, ανακούφιση από καταστροφές, άμυνα και κλιματική αλλαγή.



Εικόνα 19. Δορυφορικός αισθητήρας WorldView-2 (Copyright © MAXAR. All rights reserved.)

5.2 Διαχείριση δεδομένων

Το πλήθος των τμημάτων του μωσαϊκού που χρησιμοποιήθηκαν είναι 6940 tiles. Ακολούθως αναλύεται πώς έγινε η διαχείριση τόσο των ψηφιδωτών όσο και των διανυσματικών δεδομένων.

5.2.1 Δεδομένα εκπαίδευσης (y train)

Με το πέρας της άντλησης των μωσαϊκών ψηφιδωτών δεδομένων (raster mosaic) αλλά και των διανυσματικών δεδομένων (GeoJSON), από την πλατφόρμα Amazon Web Services (AWS), πρώτο μέλημα ήταν η δημιουργία ψηφιδωτών δεδομένων που εμπεριέχουν την χωρική πληροφορία εντοπισμού των κτιρίων. Τα διανυσματικά αυτά αρχεία χρησιμοποιήθηκαν ως μάσκες και κατ' επέκταση δομικά στοιχεία, για την δημιουργία του διανύσματος γ-εκπαίδευσης του νευρωνικού δικτύου (y train), όπως αυτή αναλύθηκε στο Κεφάλαιο 4.

5.2.1 Δεδομένα εκπαίδευσης (x train)

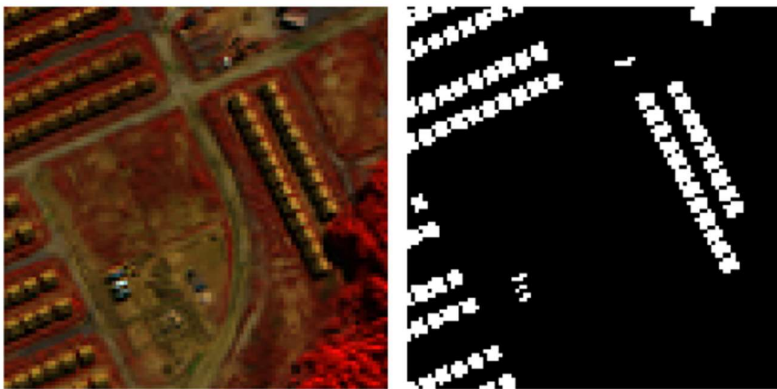
Για την εκπαίδευση του νευρωνικού δικτύου, ώστε να είναι εφικτός ο αυτοματοποιημένος εντοπισμός κτιρίων σε δορυφορικές εικόνες, πέραν του y-train απαιτείται και η δημιουργία του διανύσματος x-εκπαίδευσης (x-train). Για την υλοποίηση του εν λόγω διανύσματος χρησιμοποιήθηκαν οι εικόνες οκτώ καναλιών και με σκοπό την ανάδειξη βέλτιστα εκπαιδευμένου πλήρους συνελκτικού νευρωνικού δικτύου (U-net) δημιουργήθηκαν τα σενάρια ψηφιδωτών δεδομένων εκπαίδευσης (raster input x-train) όπως παρουσιάζονται στον ακόλουθο πίνακα.

Πίνακας 2. Σενάρια ψηφιδωτών δεδομένων εκπαίδευσης (raster input x-train)

A/A	ΣΕΝΑΡΙΟ ΨΗΦΙΔΩΤΩΝ ΔΕΔΟΜΕΝΩΝ ΕΚΠΑΙΔΕΥΣΗΣ	ΔΙΑΣΤΑΣΕΙΣ ΔΙΑΝΥΣΜΑΤΟΣ x-train
1	εικόνες τριών καναλιών (RGB)	(6940, 96, 96, 3)
2	εικόνες τεσσάρων καναλιών (RGBNir)	(6940, 96, 96, 4)
3	εικόνες NDVI	(6940, 96, 96, 1)
4	εικόνες έγχρωμου σύνθετου ανίχνευσης τσιμέντου (cement)	(6940, 96, 96, 3)
5	εικόνες ενός καναλιού (Nir)	(6940, 96, 96, 1)



Εικόνα 20. Απεικόνιση των δεδομένων εισόδου 1ου σεναρίου. Αριστερά στήλη οι δορυφορικές εικόνες RGB (input RGB images). Δεξιά στήλη οι μάσκες των κτηρίων (ground truth labels).



Εικόνα 21. Απεικόνιση των δεδομένων εισόδου 4ου σεναρίου. Αριστερά στήλη οι δορυφορικές εικόνες cement (input cement images). Δεξιά στήλη οι μάσκες των κτηρίων (ground truth labels).

Ο λόγος που επιλέχθηκαν τα ανωτέρω σεναρία αναλύεται στο κεφάλαιο που πραγματεύεται το θεωρητικό υπόβαθρο. Εκεί αναλύονται οι φασματικές ιδιότητες και η συμβολή των έγχρωμων σύνθετων (πχ cement), των λόγων καναλιών καθώς και των δεικτών (πχ NDVI), στην τέχνη της τηλεπισκόπησης. Η αλγοριθμική διαδικασία δημιουργίας των ανωτέρω ψηφιδωτών δεδομένων αναλύεται στο κεφάλαιο 4, ενώ η αξιολόγηση των αποτελεσμάτων εκπαίδευσης των δικτύων ακολουθεί στο επόμενο κεφάλαιο.

Αυτό που αξίζει να σημειωθεί είναι ότι αρχικά τα τμήματα ψηφιδωτών δεδομένων (raster tiles) είχαν επιλεγεί να έχουν διαστάσεις 100x100 εικονοστοιχεία. Δεν κατέστη όμως δυνατή η προσπέλαση των αλγορίθμων του δικτύου U-Net με αυτές τις διαστάσεις, λόγω ύπαρξης πέντε (5) επιπέδων Υποδειγματοληψίας δηλαδή του κλάδου καθόδου (down-sample) και αντίστοιχα πέντε επίπεδα στον κλάδο ανόδου (up-sample). Για την αντιμετώπιση του ζητήματος και την ορθή προσπέλαση των αλγοριθμικών διαδικασιών της αρχιτεκτονικής του δικτύου, οι εικόνες σε όλα τα ανωτέρω σεναρία μετασχηματίστηκαν σε διαστάσεις 96x96 εικονοστοιχείων. Συναφώς μετασχηματίστηκαν και τα δεδομένα του γ-διανύσματος εκπαίδευσης, ώστε να μην τίθεται ζήτημα διαστάσεων κατά την υλοποίηση πράξεων μεταξύ μητρών ανόμοιων διαστάσεων.

ΚΕΦΑΛΑΙΟ 6

Μελέτη σεναρίων

6. Εισαγωγικά

Το πέρας της προετοιμασίας των δεδομένων και δημιουργίας των πέντε σεναρίων, που αναλύθηκαν στο προηγούμενο κεφάλαιο, ακολουθεί η προσπέλαση των αλγορίθμων την εκπαίδευση του δικτύου. Για την ανάδειξη του βέλτιστου σεναρίου χρειαζόμαστε δείκτες αξιολόγησης των αποτελεσμάτων. Στο παρόν κεφάλαιο αναλύονται οι μετρικοί δείκτες που επιλέχθηκαν και στη συνέχεια παρουσιάζονται και αξιολογούνται τα αποτελέσματα εκπαίδευσης του δικτύου.

6.1 Δείκτες αξιολόγησης αποτελεσμάτων

Σε αυτή την ενότητα, παρουσιάζονται οι μετρικοί δείκτες που εφαρμόστηκαν σε κάθε εικόνα του αποθετηρίου δεδομένων ελέγχου. Συγκεκριμένα, χρησιμοποιήθηκαν οι δείκτες ακρίβειας (accuracy), ακρίβειας ταξινόμησης (precision), ανάκλησης (recall), ο F1 δείκτης (F1 score) και, τέλος, ο δείκτης Jaccard (Jaccard score).

Δείκτης accuracy

Ο δείκτης accuracy ορίζεται ως:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (5)$$

τιμές TP, TN, FP, FN αντιπροσωπεύουν τις έννοιες True Positive, True Negative, False Positive και False Negative αντίστοιχα. Αυτές οι τιμές προκύπτουν από τον πίνακα σύγχυσης (Confusion Matrix), ο οποίος υπολογίζεται με βάση την προβλεπόμενη εικόνα του μοντέλου (predicted label) και την θεωρητικά σωστή εικόνα (annotated label). Σε ένα πρόβλημα σημασιολογικής κατάτμησης με 2 κλάσεις, τα αποτελέσματα του πίνακα σύγχυσης περιγράφουν τα τέσσερα πιθανά σενάρια που καταγράφουν τη σχέση των pixels μεταξύ των δύο εικόνων εισόδου.

Πίνακας 3. Ορισμός πίνακα σύγχυσης (Confusion Matrix)

		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

- **True Positive:** Οι περιπτώσεις για τις οποίες προβλέφθηκε κατασκευή και στην πραγματικότητα είναι κατασκευή.

- **True Negative:** Οι περιπτώσεις για τις οποίες δεν προβλέφθηκε κατασκευή και στην πραγματικότητα δεν υπάρχει κατασκευή.
- **False Positive:** Οι περιπτώσεις κατά τις οποίες προβλέφθηκε κατασκευή και στην πραγματικότητα δεν υπάρχει κατασκευή.
- **False Negative:** Οι περιπτώσεις κατά τις οποίες δεν προβλέφθηκε κατασκευή, ωστόσο στην πραγματικότητα, υπάρχει.

Δείκτης Precision

Ο δείκτης Precision, είναι ο λόγος των σωστά προβλεπόμενων θετικών παρατηρήσεων προς τις συνολικές προβλεπόμενες θετικές παρατηρήσεις. Το ερώτημα που απαντάει αυτή η μετρητική είναι από το σύνολο των κτηρίων που εντοπίστηκαν, ποια από αυτά ήταν όντως κτήρια. Ορίζεται ως:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (6)$$

Δείκτης Recall

Ο δείκτης Recall, είναι ο λόγος των σωστά προβλεπόμενων θετικών παρατηρήσεων προς όλες τις παρατηρήσεις στην πραγματική κλάση (κτήριο). Το ερώτημα που απαντάει αυτή η μετρητική είναι ότι από το σύνολο σωστών κτηρίων που εντοπίστηκαν πόσα από αυτά αναφέρθηκαν. Ορίζεται ως:

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (7)$$

Ο δείκτης F1 score αποτελεί έναν αρμονικό μέσο όρο των δυο παραπάνω, καθώς προσπαθεί να συνδυάσει και να υπολογίσει σε ένα μοναδικό δείκτη τις απαντήσεις που δίνουν οι δυο παραπάνω, ενώ δίνεται ως:

$$\text{F1} = 2 \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}} \quad (8)$$

Δείκτης Jaccard

Ο επόμενος δείκτης ονομάζεται Jaccard score και αποτελεί τον λόγο της τομής δυο συνόλων προς την ένωσή τους. Πιο συγκεκριμένα υποθέτοντας ένα σύνολο A το οποίο εκφράζει την εικόνα με τα ψηφιοποιημένα κτήρια από τους ειδικούς και B το σύνολο που εκφράζει την εικόνα με τα προβλεπόμενα από το δίκτυο κτήρια ο δείκτης Jaccard score ορίζεται ως:

$$\text{Jaccard} = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (9)$$

Οι πέντε δείκτες που προαναφέρθηκαν υπολογίστηκαν για κάθε εικόνα του σετ ελέγχου και σε δεύτερο στάδιο τα αποτελέσματα των δεικτών χρησιμοποιήθηκαν στον τελικό υπολογισμό της μέσης τιμής, τυπικής απόκλισης, μέγιστης και ελάχιστης τιμής για κάθε δείκτη προκειμένου να υπάρχει μια σφαιρική εικόνα της απόδοσης του μοντέλου. Πέραν των ανωτέρω κατασκευάστηκαν διαγράμματα από την μελέτη δεικτών Accuracy και Loss.

Training & Validation Accuracy - Loss

Όταν εκπαιδύουμε μοντέλα μηχανικής μάθησης, ιδιαίτερα νευρωνικά δίκτυα, η παρακολούθηση της συμπεριφοράς αυτών των δεικτών είναι κρίσιμη για την κατανόηση της απόδοσης του μοντέλου και τη διάγνωση πιθανών προβλημάτων.

Training Loss

Ορίζεται ως η απώλεια εκπαίδευσης και μετρά το σφάλμα των προβλέψεων του μοντέλου στο σύνολο δεδομένων εκπαίδευσης. Συνήθως υπολογίζεται με μια συνάρτηση απώλειας (π.χ., Μέσο Τετραγωνικό Σφάλμα, Απώλεια Διασταυρούμενης Εντροπίας) που ποσοτικοποιεί τη διαφορά μεταξύ των προβλεπόμενων εξόδων και των πραγματικών στόχων.

Κατά την εκπαίδευση, η απώλεια εκπαίδευσης θα πρέπει να μειώνεται καθώς το μοντέλο μαθαίνει τα μοτίβα στα δεδομένα εκπαίδευσης. Μια συνεχώς υψηλή απώλεια εκπαίδευσης υποδηλώνει ότι το μοντέλο μπορεί να υποπροσαρμόζεται (underfitting).

Validation loss

Ορίζεται ως η απώλεια επαλήθευσης και μετρά το σφάλμα των προβλέψεων του μοντέλου σε ένα ξεχωριστό σύνολο δεδομένων επαλήθευσης που δεν χρησιμοποιήθηκε κατά την εκπαίδευση. Παρέχει ένδειξη για το πόσο καλά γενικεύει το μοντέλο σε νέα, άγνωστα δεδομένα.

Αρχικά, η απώλεια επαλήθευσης θα πρέπει επίσης να μειώνεται καθώς το μοντέλο βελτιώνεται. Ωστόσο, αν η απώλεια επαλήθευσης αρχίσει να αυξάνεται ενώ η απώλεια εκπαίδευσης συνεχίζει να μειώνεται, αυτό υποδηλώνει ότι το μοντέλο υπερπροσαρμόζεται (overfitting) στα δεδομένα εκπαίδευσης.

Training Accuracy

Ορίζεται ως η ακρίβεια εκπαίδευσης και μετρά το ποσοστό των σωστών προβλέψεων του μοντέλου στο σύνολο δεδομένων εκπαίδευσης. Υπολογίζεται ως ο αριθμός των σωστών προβλέψεων διαιρεμένος με το συνολικό αριθμό των δειγμάτων στο σύνολο δεδομένων εκπαίδευσης.

Κατά την εκπαίδευση, η ακρίβεια εκπαίδευσης θα πρέπει να αυξάνεται καθώς το μοντέλο μαθαίνει τα μοτίβα στα δεδομένα εκπαίδευσης. Μια συνεχώς χαμηλή ακρίβεια εκπαίδευσης υποδηλώνει ότι το μοντέλο μπορεί να υποπροσαρμόζεται.

Validation Accuracy

Ορίζεται ως η ακρίβεια επαλήθευσης και μετρά το ποσοστό των σωστών προβλέψεων του μοντέλου σε ένα ξεχωριστό σύνολο δεδομένων επαλήθευσης που δεν χρησιμοποιήθηκε κατά την εκπαίδευση. Παρέχει ένδειξη για το πόσο καλά γενικεύει το μοντέλο σε νέα, άγνωστα δεδομένα.

Αρχικά, η ακρίβεια επαλήθευσης θα πρέπει επίσης να αυξάνεται καθώς το μοντέλο βελτιώνεται. Ωστόσο, αν η ακρίβεια επαλήθευσης σταματήσει να αυξάνεται ή αρχίσει να μειώνεται ενώ η ακρίβεια εκπαίδευσης συνεχίζει να αυξάνεται, αυτό υποδηλώνει ότι το μοντέλο υπερπροσαρμόζεται στα δεδομένα εκπαίδευσης.

Από την διαδικασία μελέτης και αξιολόγησης των ανωτέρω δεικτών διακρίνονται οι ακόλουθες περιπτώσεις:

➤ Υποπροσαρμογή (Underfitting)

Οι τιμές Loss (εκπαίδευσης και επαλήθευσης) είναι υψηλές και δεν μειώνονται σημαντικά, και αντίστοιχα η ακρίβεια εκπαίδευσης και η ακρίβεια επαλήθευσης είναι χαμηλές. Αυτό δείχνει ότι το μοντέλο είναι πολύ απλό για να συλλάβει τα υποκείμενα μοτίβα στα δεδομένα.

➤ Υπερπροσαρμογή (Overfitting)

Η απώλεια εκπαίδευσης συνεχίζει να μειώνεται ενώ η απώλεια επαλήθευσης αρχίζει να αυξάνεται μετά από κάποιο σημείο, και αντίστοιχα η ακρίβεια εκπαίδευσης συνεχίζει να αυξάνεται ενώ η ακρίβεια επαλήθευσης σταματά να αυξάνεται ή αρχίζει να μειώνεται. Αυτό σημαίνει ότι το μοντέλο μαθαίνει τον θόρυβο και τις συγκεκριμένες λεπτομέρειες από τα δεδομένα εκπαίδευσης που δεν γενικεύονται στα δεδομένα επαλήθευσης.

➤ Καλή Προσαρμογή

Και οι δύο απώλειες (εκπαίδευσης και επαλήθευσης) μειώνονται και σταθεροποιούνται σε χαμηλή τιμή, ενώ αντίστοιχα οι ακρίβειες αυξάνονται και σταθεροποιούνται σε υψηλή τιμή. Αυτό δείχνει ότι το μοντέλο έχει προσαρμοστεί καλά στα δεδομένα εκπαίδευσης και γενικεύει καλά σε άγνωστα δεδομένα.

6.4 Παράμετροι εκπαίδευσης U-Net

Στην παρούσα ενότητα θα γίνει ανάλυση των παραμέτρων με σκοπό την άρτια εκπαίδευση του δικτύου U-Net. Αρχικά να αναφέρουμε ότι για την προσπέλαση των αλγορίθμων χρησιμοποιήθηκε το περιβάλλον εργασίας Colab, ώστε να επιτευχθεί η ταχύτητα επεξεργασίας.

Επίσης πολύ σημαντική παράμετρος αποτελεί το batch size που εκφράζει πόσες εικόνες του τετραδιάστατου τανιστή θα χρησιμοποιούνται σε κάθε εποχή μέχρι να

εκπαιδευτεί πλήρως το δίκτυο. Η τιμή αυτή ήταν 16, προκειμένου παράμετροι όπως π.χ. η απώλεια να υπολογίζονται από ένα ικανοποιητικό σύνολο, ενώ ταυτόχρονα να μην επιβαρύνεται με πολλούς υπολογισμούς η τοπική μονάδα επεξεργασίας.

Το δίκτυο εκπαιδεύτηκε λοιπόν χρησιμοποιώντας batches των 16 εικόνων για 30 εποχές. Μια εποχή έχει ολοκληρωθεί όταν όλο το σετ δεδομένων εκπαίδευσης έχει ανακυκλωθεί στο δίκτυο. Με σκοπό την αποφυγή της υπερπροσαρμογής χρησιμοποιήθηκαν μερικά λογικά κριτήρια τα οποία προσπαθούν να προσεγγίσουν το κρίσιμο σημείο πριν την υπερπροσαρμογή. Αυτά είναι το Early stopping και το Model checkpoint. Το EarlyStopping, ως κριτήριο μελετάει την τιμή της συνάρτησης απώλειας (loss) στο validation set (validation loss) και σύμφωνα με μια τιμή υπομονής αν στο μοντέλο δεν έχει βελτιωθεί το αποτέλεσμα της validation loss τότε σταματάει η εκπαίδευση. Στο πρόγραμμα η τιμή υπομονής ήταν 5 και το δίκτυο σταμάτησε να εκπαιδεύεται μετά από 16 εποχές. Τέλος, υπάρχει το ModelCheckpoint, κατά το οποίο αποθηκεύεται το καλύτερο μοντέλο, ανάμεσα στις εποχές που έχουν οριστεί και αποθηκεύει αποκλειστικά τα βάρη.

Επόμενη παράμετρος της εκπαίδευσης του δικτύου αποτελεί η επιλογή αλγορίθμου βελτιστοποίησης. Στην συγκεκριμένη περίπτωση έχει χρησιμοποιηθεί ο Adam (Adaptive movement) [33]. Ο αλγόριθμος Adam αποτελεί μια μέθοδο αποτελεσματικής στοχαστικής βελτιστοποίησης (stochastic optimization) που απαιτεί μόνο παραγώγους πρώτης τάξης ενώ παράλληλα έχει ελάχιστες απαιτήσεις σε μνήμη. Η μέθοδος υπολογίζει μεμονωμένα προσαρμοστικά ποσοστά μάθησης (learning rates), για διαφορετικές παραμέτρους από εκτιμήσεις πρώτων και δεύτερων στιγμών των κλίσεων. Το όνομα Adam προέρχεται από adaptive moment estimation. Ο Adam optimizer, χρησιμοποιείται καθώς συνδυάζει τα πλεονεκτήματα δύο στοχαστικών αλγορίθμων μείωσης κλίσης του Adaptive Gradient Algorithm (AdaGrad) [34], ο οποίος δουλεύει καλά με αραιές κλίσεις και του Root Mean Square Propagation (RMSProp) [35] και λειτουργεί πολύ καλά σε γραμμικές και μη στατικές συνθήκες.

Μερικά από τα πλεονεκτήματα του Adam είναι ότι τα μεγέθη των παραμέτρων ενημέρωσης είναι αναλλοίωτα στην κλίμακα της κλίσης, ότι τα μεγέθη των βημάτων του αρχικοποιούνται περίπου από το μέγεθος βήματος (step size) της υπερπαραμέτρου, ότι δεν απαιτεί σταθερό στόχο, ότι λειτουργεί με αραιές κλίσεις και ότι εκτελεί μια μορφή μεγέθους βημάτων ανόπτησης.

Επόμενη σημαντική παράμετρος του δικτύου, αποτελεί η συνάρτηση απώλειας η οποία όπως έχει αναφερθεί και προηγουμένως αποτελεί καίριο παράγοντα της εκπαίδευσης του δικτύου. Το πρόβλημα της αυτόματης εξαγωγής κτηρίων είναι δυαδικής φύσεως, καθώς απώτερος στόχος αποτελεί η ταξινόμηση κάθε εικονοστοιχείου σε δυο κατηγορίες, κτήριο ή όχι κτήριο. Δηλαδή μετά το πέρας κάθε εποχής για κάθε εικονοστοιχείο αναγράφεται αυτή η υπολογιζόμενη πιθανότητα.

Η συνάρτηση απώλειας έχει ως βασικό στόχο να αξιολογήσει πόσο ορθές ή μη ορθές είναι οι υπολογιζόμενες πιθανότητες. Πρέπει να επιστρέφει υψηλές τιμές για τις μη ορθές προβλέψεις και χαμηλές τιμές για τις ορθές.

Ακόμα, τελευταία παράμετρος της εκπαίδευσης είναι η μετρητική συνάρτηση, η οποία χρησιμοποιείται για να αξιολογήσει την απόδοση του μοντέλου. Οι μετρητικές συναρτήσεις, είναι παρόμοιες με τις συναρτήσεις απώλειας με ειδοποιό διαφορά πως το αποτέλεσμα της μετρητικής συνάρτησης δεν χρησιμοποιείται κατά τη διαδικασία της εκπαίδευσης παρά μόνο για αξιολόγηση του μοντέλου. Στην συγκεκριμένη περίπτωση, ως μετρητική συνάρτηση χρησιμοποιείται ο δείκτης accuracy. Ο συγκεκριμένος δείκτης, αξιολογεί πόσες από τις προβλεπόμενες κλάσεις ταυτίζονται με τις αληθινές κλάσεις.

Πίνακας 4. Συγκεντρωτικός πίνακας παραμέτρων δικτύου

PARAMETERS	
Batch size	16
Image Width	96
Image Height	96
Optimizer	Adam
Loss function	Binary cross entropy
Activation function	Sigmoid
Metric	Accuracy
N° Epochs	30
Earlystopping (patience)	5
ModelCheckpoint	Save best only
Total Parameters	1,968,229
Trainable Parameters	1,965,283

6.3 Παράθεση και αξιολόγηση σεναρίων

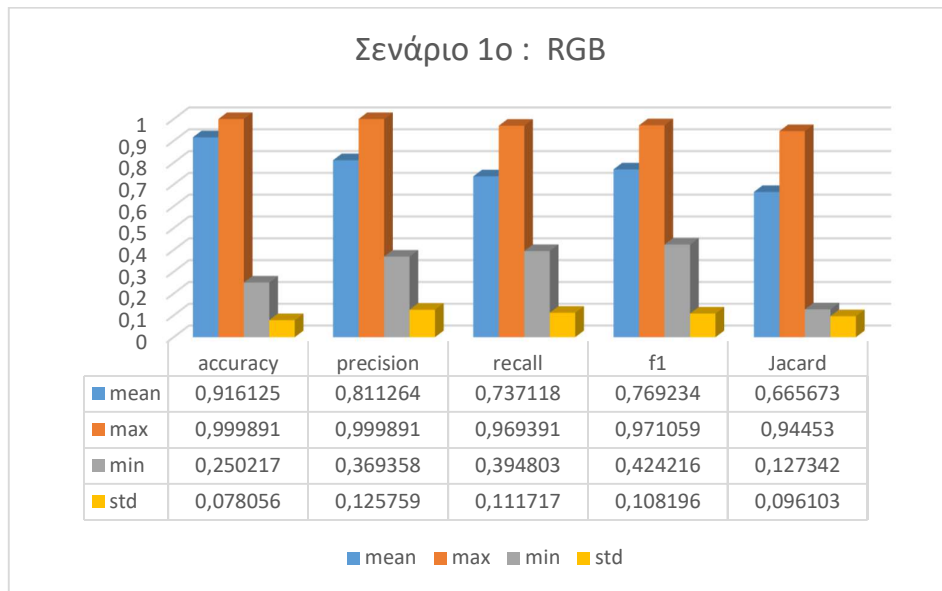
Ακολουθεί παρουσίαση και αξιολόγηση των αποτελεσμάτων εκπαίδευσης του πλήρους συνελκτικού νευρωνικού δικτύου U-Net σε τριάντα εποχές και σε πέντε διαφορετικά σενάρια όπως αυτά αναλύθηκαν στο προηγούμενο κεφάλαιο.

Πίνακας 5. Σενάρια εκπαίδευσης

A/A	ΣΕΝΑΡΙΟ ΨΗΦΙΔΩΤΩΝ ΔΕΔΟΜΕΝΩΝ ΕΚΠΑΙΔΕΥΣΗΣ
1	εικόνες τριών καναλιών (RGB)
2	εικόνες τεσσάρων καναλιών (RGBNir)
3	εικόνες NDVI
4	εικόνες έγχρωμου σύνθετου ανίχνευσης τσιμέντου (cement)
5	εικόνες ενός καναλιού (Nir)

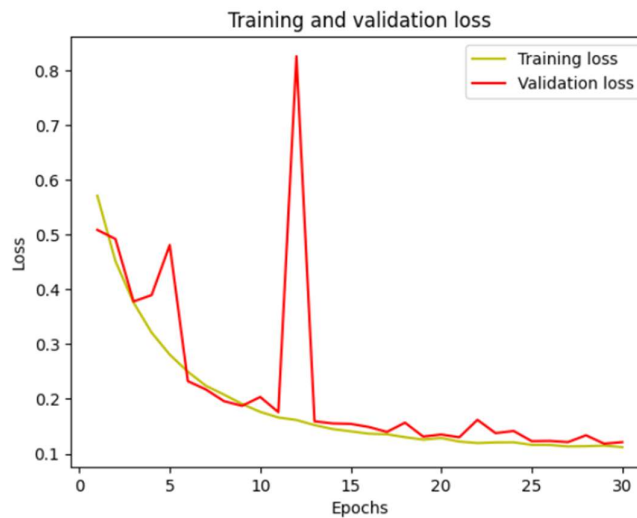
1^ο ΣΕΝΑΡΙΟ: Εικόνες τριών καναλιών (RGB)

Στο πρώτο σενάριο ως δεδομένα εισόδου για την εκπαίδευση του δικτύου επιλέχθηκαν εικόνες με έγχρωμο σύνθετο του ορατού φάσματος (RGB). Παρατηρούμε ικανοποιητικές αποδόσεις στους μέσους όρους των μετρικών δεικτών, όπου σε συνδυασμό με τις χαμηλές τιμές τυπικής απόκλισης είναι και αντιπροσωπευτικοί.

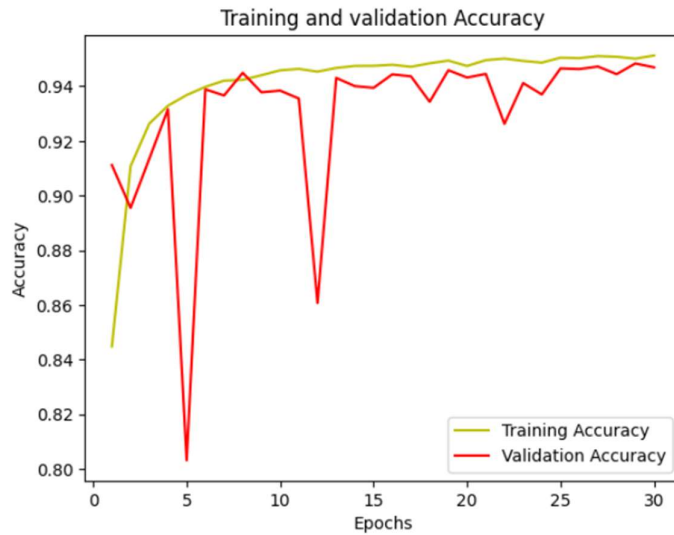


Γράφημα 1. Γράφημα στατιστικής μελέτης δεικτών στις εικόνες 1ου σεναρίου

Τα γραφήματα Training & Validation Accuracy – Loss υποδεικνύουν μια καλή προσαρμογή του δικτύου, καθώς οι τιμές Training & Validation Loss μειώνονται στο πέρασμα των εποχών και δείχνουν να σταθεροποιούνται σε χαμηλή τιμή. Παράλληλα, οι ακρίβειες αυξάνονται και σταθεροποιούνται σε υψηλή τιμή.

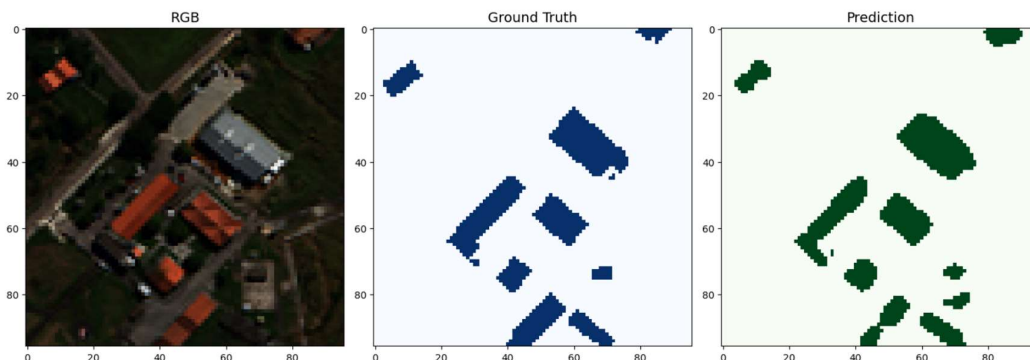


Γράφημα 2. Γραφήματα Training & Validation Loss για το 1ο σενάριο



Γράφημα 3. Γραφήματα Training & Validation Accuracy για το 1ο σενάριο

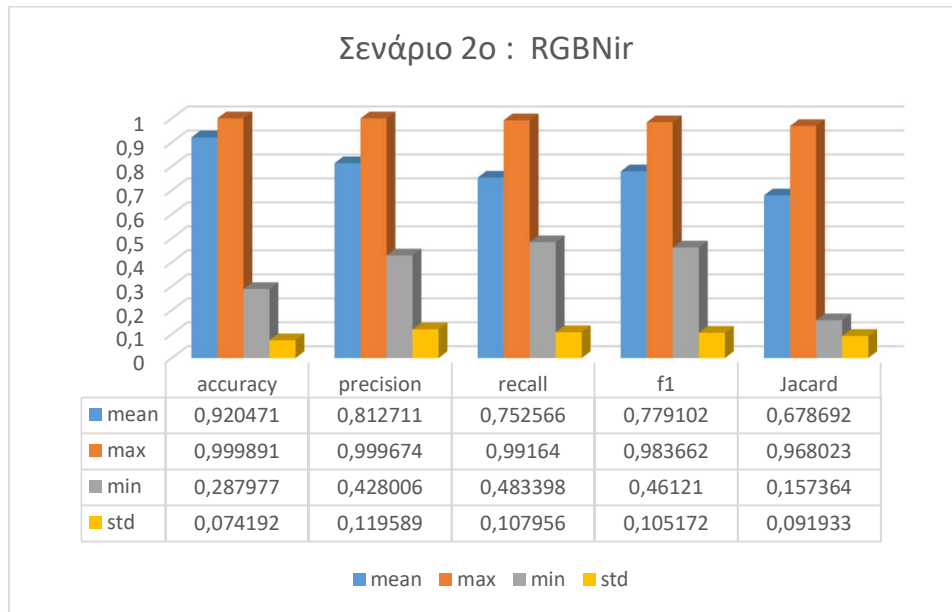
Ακολουθεί οπτική σύγκριση μιας εικόνας εισόδου (RGB) στο δίκτυο, με την μάσκα εκπαίδευσης (Ground Truth) όπως αντλήσαμε από την SpaceNet για την περιοχή μελέτης και την πρόβλεψη που κατάφερε το δίκτυο (Prediction). Παρατηρούμε σφάλματα και λάθη σε ακμές, παρά ταύτα τα αποτελέσματα κρίνονται ικανοποιητικά.



Εικόνα 22. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 1ου σεναρίου.

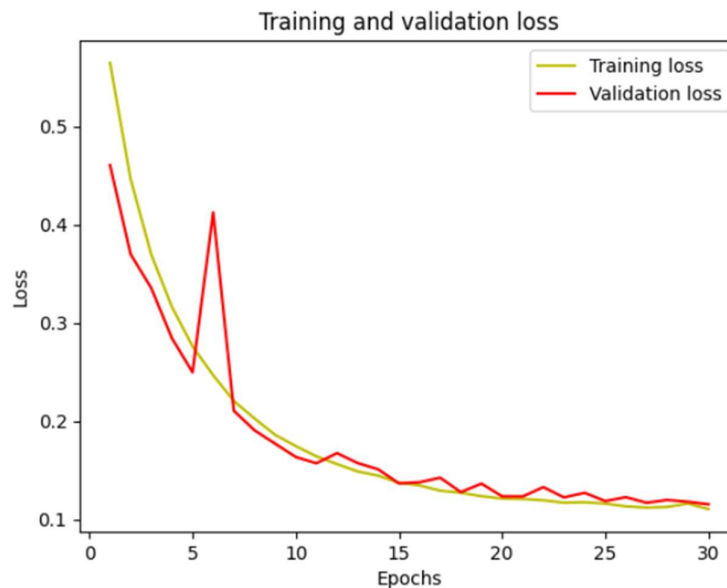
2^ο ΣΕΝΑΡΙΟ: Εικόνες τεσσάρων καναλιών (RGBNir)

Στο δεύτερο σενάριο ως δεδομένα εισόδου για την εκπαίδευση του δικτύου επιλέχθηκαν εικόνες με έγχρωμο σύνθετο τεσσάρων καναλιών, καθώς συνδυάσαμε τα κανάλια του ορατού φάσματος με το εγγύς υπέρυθρο (RGBNir). Παρατηρούμε ικανοποιητικές αποδόσεις στους μέσους όρους των μετρικών δεικτών, όπου σε συνδυασμό με τις χαμηλές τιμές τυπικής απόκλισης είναι και αντιπροσωπευτικοί.



Γράφημα 4. Γράφημα στατιστικής μελέτης δεικτών στις εικόνες 2ου σεναρίου

Τα γραφήματα Training & Validation Accuracy – Loss υποδεικνύουν μια καλή προσαρμογή του δικτύου, καθώς οι τιμές Training & Validation Loss μειώνονται στο πέρασμα των εποχών και δείχνουν να σταθεροποιούνται σε χαμηλή τιμή. Αντίστοιχα, οι ακρίβειες αυξάνονται και σταθεροποιούνται σε υψηλή τιμή. Αν και παρατηρείται μια τάση υπερπροσαρμογής έως την 12^η εποχή, το δίκτυο στη συνέχεια δείχνει να σταθεροποιείται.

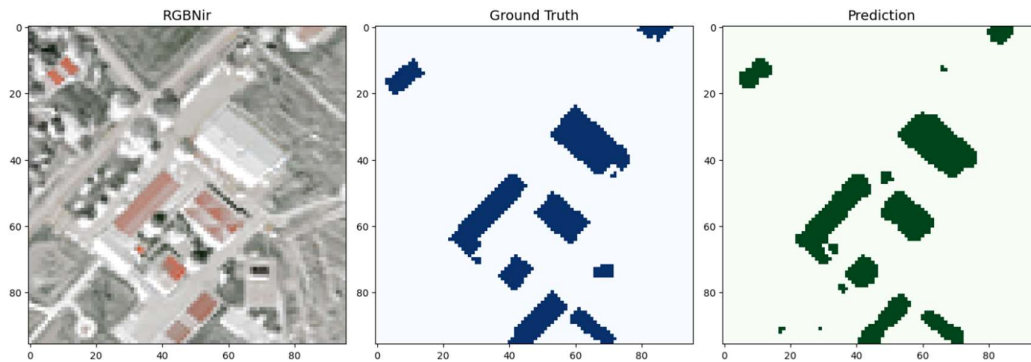


Γράφημα 5. Γραφήματα Training & Validation Loss για το 2ο σενάριο



Γράφημα 6. Γραφήματα Training & Validation Accuracy για το 2ο σενάριο

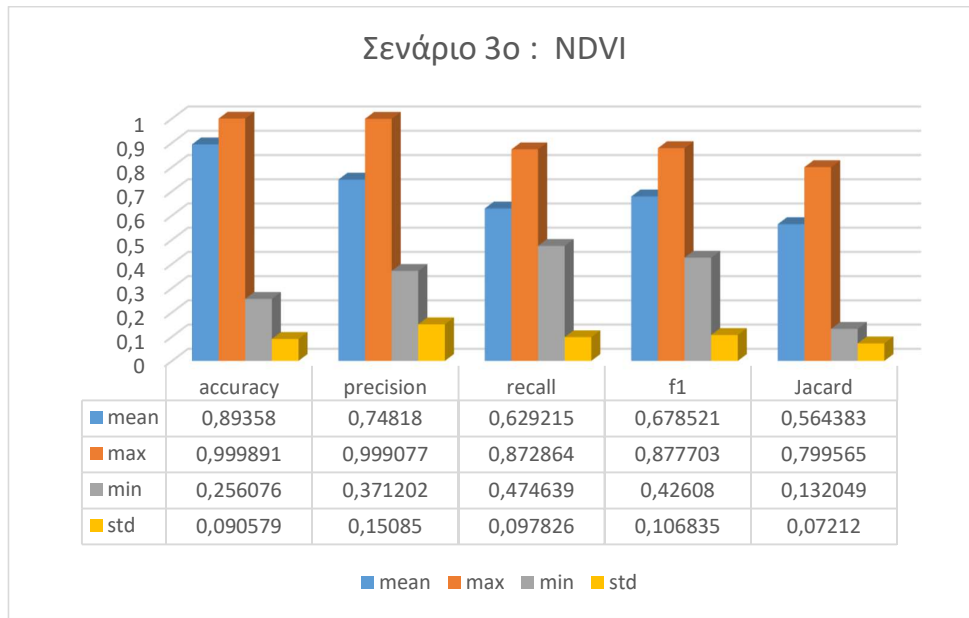
Ακολουθεί οπτική σύγκριση μιας εικόνας εισόδου (RGBNir) στο δίκτυο, με την μάσκα εκπαίδευσης (Ground Truth) όπως αντλήσαμε από την SpaceNet για την περιοχή μελέτης και την πρόβλεψη που κατάφερε το δίκτυο (Prediction). Παρατηρούμε σφάλματα και λάθη σε ακμές, παρόλα αυτά τα αποτελέσματα κρίνονται ικανοποιητικά.



Εικόνα 23. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 2ου σεναρίου.

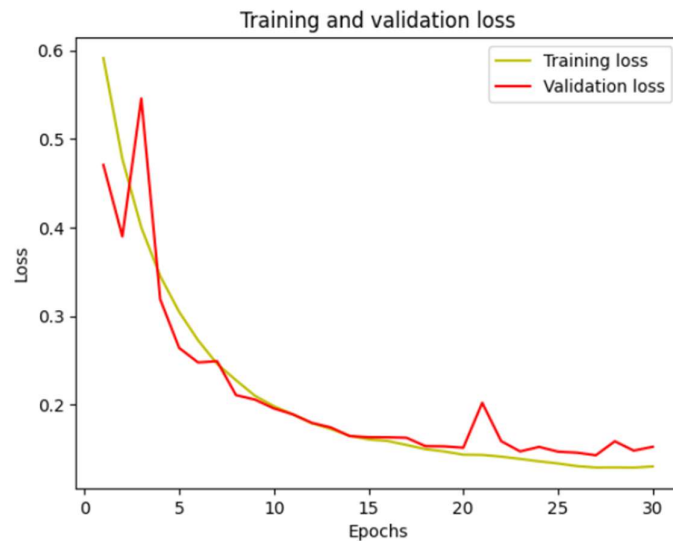
3^ο ΣΕΝΑΡΙΟ: Εικόνες NDVI

Στο τρίτο σενάριο ως δεδομένα εισόδου για την εκπαίδευση του δικτύου επιλέχθηκαν εικόνες ενός καναλιού με τιμές τα αποτελέσματα υπολογισμού του κανονικοποιημένου δείκτη για εντοπισμό βλάστησης (NDVI). Η επιλογή του εν λόγω σεναρίου έγινε καθώς προσπαθούμε να εκπαιδεύσουμε ένα δίκτυο για την ανίχνευση κατασκευών που εδράζονται σε περιοχές κυρίως με βλάστηση. Παρατηρούμε ικανοποιητικές αποδόσεις στους μέσους όρους των μετρικών δεικτών, όπου σε συνδυασμό με τις χαμηλές τιμές τυπικής απόκλισης είναι και αντιπροσωπευτικοί.

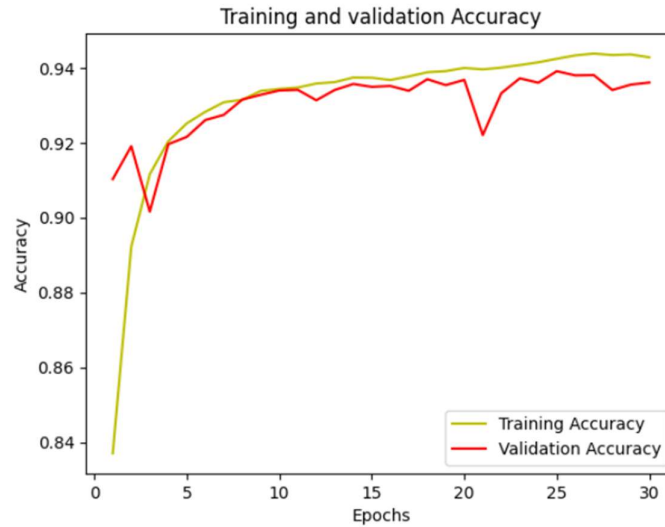


Γράφημα 7. Γράφημα στατιστικής μελέτης δεικτών στις εικόνες 3ου σεναρίου

Τα γραφήματα Training & Validation Accuracy – Loss υποδεικνύουν μια καλή προσαρμογή του δικτύου, καθώς οι τιμές Training & Validation Loss μειώνονται στο πέρασμα των εποχών και δείχνουν να σταθεροποιούνται σε χαμηλή τιμή. Επίσης, οι ακρίβειες αυξάνονται και σταθεροποιούνται σε υψηλή τιμή. Αν και παρατηρείται μια τάση υπερπροσαρμογής έως την 12^η εποχή, το δίκτυο στη συνέχεια δείχνει να σταθεροποιείται.

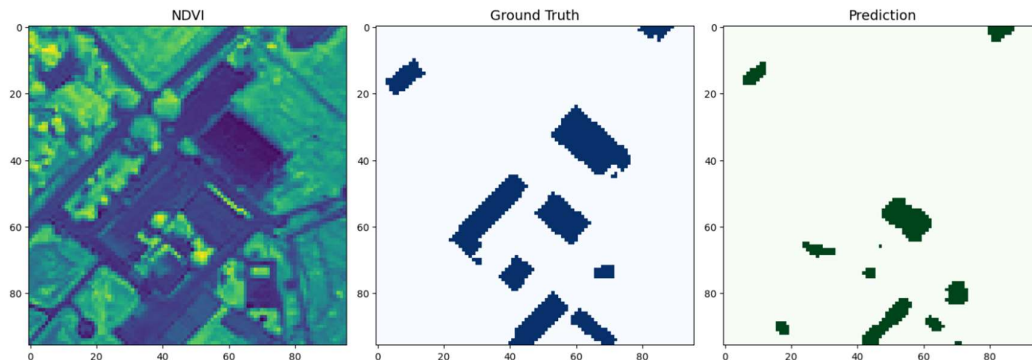


Γράφημα 8. Γραφήματα Training & Validation Loss για το 3ο σενάριο



Γράφημα 9. Γραφήματα Training & Validation Accuracy για το 3ο σενάριο

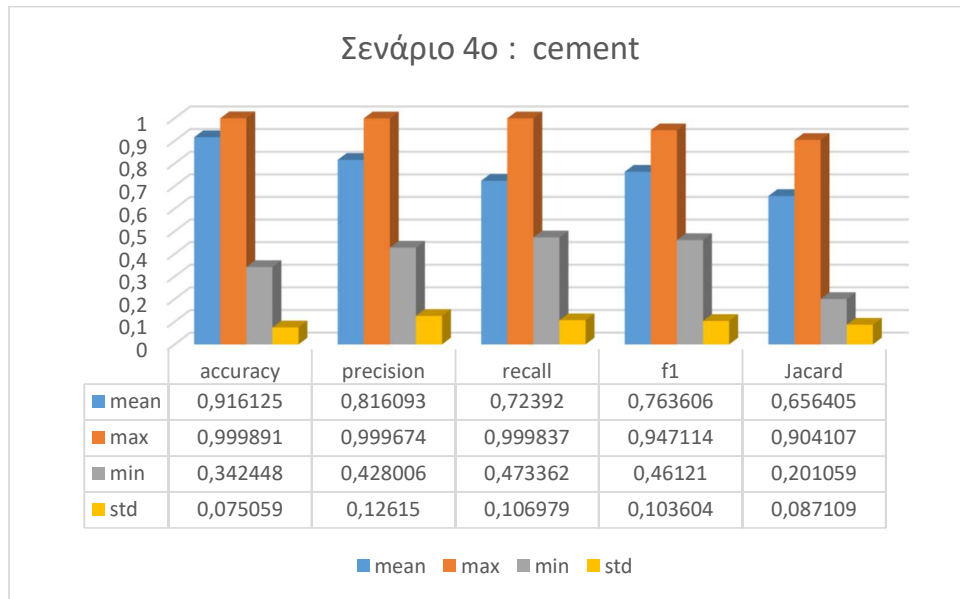
Ακολουθεί οπτική σύγκριση μιας εικόνας εισόδου (NDVI) στο δίκτυο, με την μάσκα εκπαίδευσης (Ground Truth) όπως αντλήσαμε από την SpaceNet για την περιοχή μελέτης και την πρόβλεψη που κατάφερε το δίκτυο (Prediction). Παρατηρούμε σφάλματα και λάθη σε ακμές, και σε σύγκριση με τα δύο προηγούμενα σενάρια απουσιάζει αρκετή πληροφορία στη πρόβλεψη.



Εικόνα 24. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 3ου σεναρίου.

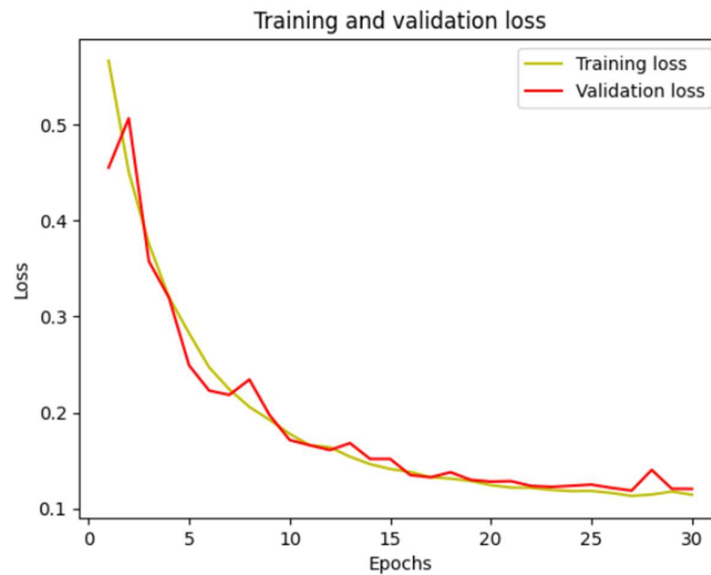
4^ο ΣΕΝΑΡΙΟ: Εικόνες έγχρωμο σύνθετου ανίχνευσης τσιμέντου (cement)

Στο τέταρτο σενάριο, ως δεδομένα εισόδου για την εκπαίδευση του δικτύου επιλέχθηκαν εικόνες με έγχρωμο σύνθετο τριών καναλιών (NIR2, Yellow, Blue). Η επιλογή του εν λόγω σεναρίου έγινε, καθώς όπως διαπιστώθηκε από την βιβλιογραφία υπάρχει μεγαλύτερη διάκριση στο τσιμέντο. Παρατηρούμε ικανοποιητικές αποδόσεις στους μέσους όρους των μετρικών δεικτών, όπου σε συνδυασμό με τις χαμηλές τιμές τυπικής απόκλισης είναι και αντιπροσωπευτικοί.



Γράφημα 10. Γράφημα στατιστικής μελέτης δεικτών στις εικόνες 4ου σεναρίου

Τα γραφήματα Training & Validation Accuracy – Loss υποδεικνύουν μια καλή προσαρμογή του δικτύου, καθώς οι τιμές Training & Validation Loss μειώνονται στο πέρασμα των εποχών και δείχνουν να σταθεροποιούνται σε χαμηλή τιμή. Αντίστοιχα οι ακρίβειες αυξάνονται και σταθεροποιούνται σε υψηλή τιμή. Αν και παρατηρείται μια τάση υπερπροσαρμογής έως την 12^η εποχή, το δίκτυο στη συνέχεια δείχνει να σταθεροποιείται.

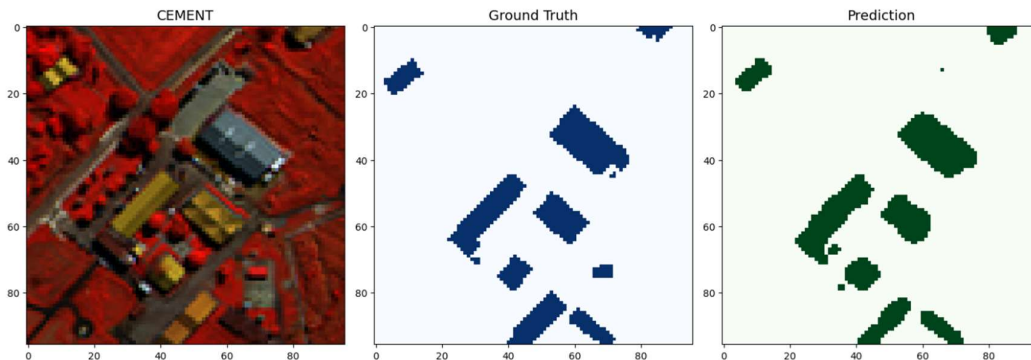


Γράφημα 11. Γραφήματα Training & Validation Loss για το 4ο σενάριο



Γράφημα 12. Γραφήματα Training & Validation Accuracy για το 4ο σενάριο

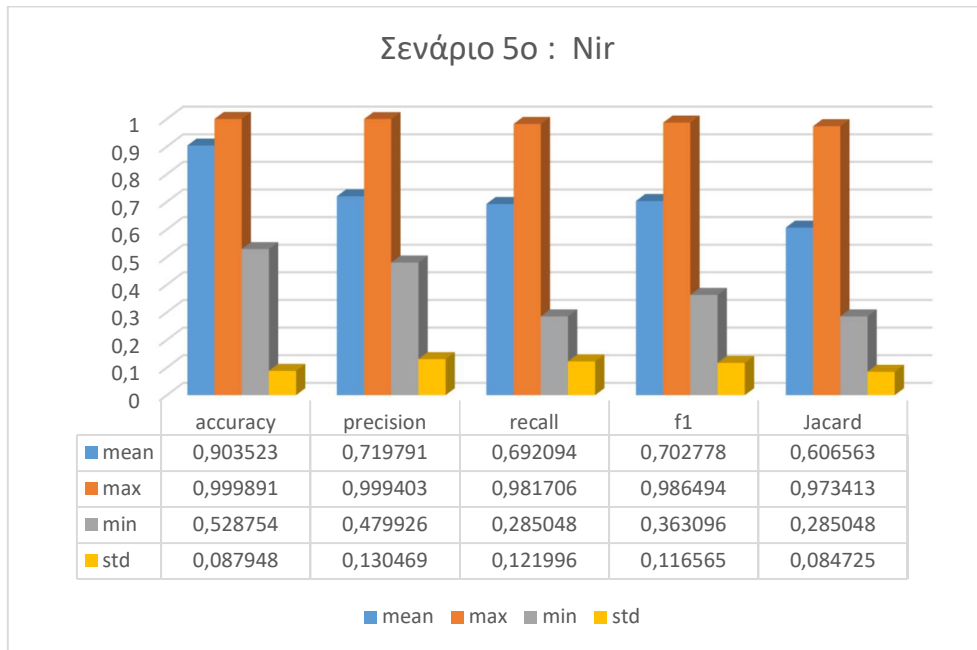
Ακολουθεί οπτική σύγκριση μιας εικόνας εισόδου (cement) στο δίκτυο, με την μάσκα εκπαίδευσης (Ground Truth) όπως αντλήσαμε από την SpaceNet για την περιοχή μελέτης και την πρόβλεψη που κατάφερε το δίκτυο (Prediction). Παρατηρούμε σφάλματα και λάθη σε ακμές, και σε σύγκριση με τα δύο προηγούμενα σενάρια απουσιάζει αρκετή πληροφορία στη πρόβλεψη.



Εικόνα 25. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 4ου σεναρίου.

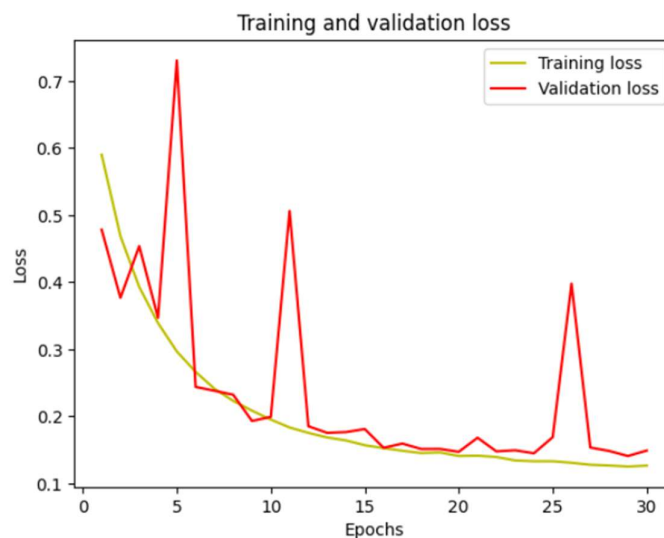
5° ΣΕΝΑΡΙΟ: Εικόνες ενός καναλιού (Nir)

Στο τελευταίο σενάριο χρησιμοποιήθηκαν ψηφιδωτά δεδομένα εκπαίδευσης ενός μόνο καναλιού και συγκεκριμένα του εγγύς υπέρυθρου καναλιού (Nir). Η επιλογή αυτή έγινε καθώς η βλάστηση αντανακλά, δηλαδή καταγράφονται υψηλές τιμές σε αυτό το κανάλι, σε σύγκριση με τις κατασκευές που παρουσιάζουν χαμηλότερες τιμές. Παρατηρούμε ικανοποιητικές αποδόσεις στους μέσους όρους των μετρικών δεικτών, όπου σε συνδυασμός με τις χαμηλές τιμές τυπικής απόκλισης είναι και αντιπροσωπευτικοί.



Γράφημα 13. Γράφημα στατιστικής μελέτης δεικτών στις εικόνες 5ου σεναρίου

Τα γραφήματα Training & Validation Accuracy – Loss υποδεικνύουν μια όχι και τόσο καλή προσαρμογή του δικτύου, καθώς οι τιμές Training & Validation Loss μειώνονται στο πέρασμα των εποχών και δείχνουν να σταθεροποιούνται σε χαμηλή τιμή. Σε αντίθεση, οι ακρίβειες αυξάνονται και σταθεροποιούνται σε υψηλή τιμή. Παρατηρείται μια τάση υπερπροσαρμογής έως την 12^η εποχή και το δίκτυο στη συνέχεια δείχνει να σταθεροποιείται καθώς συνεχίζουν να υπάρχουν έντονες αυξομειώσεις.

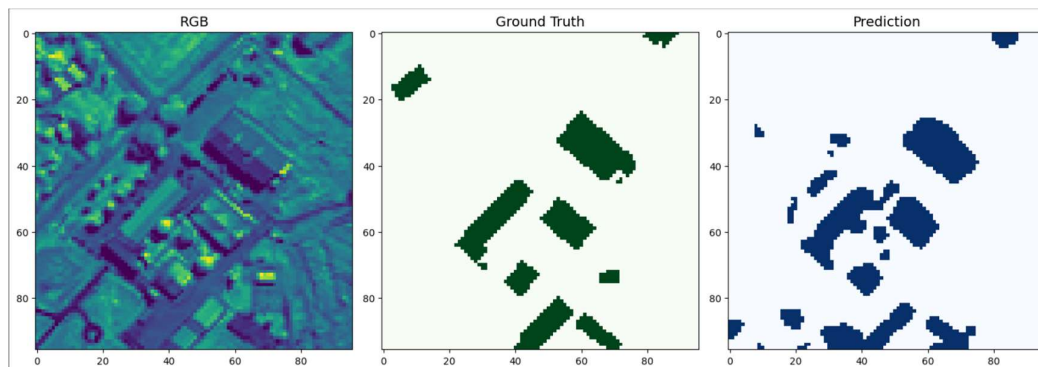


Γράφημα 14. Γραφήματα Training & Validation Loss για το 5ο σενάριο



Γράφημα 15. Γραφήματα Training & Validation Accuracy για το 5ο σενάριο

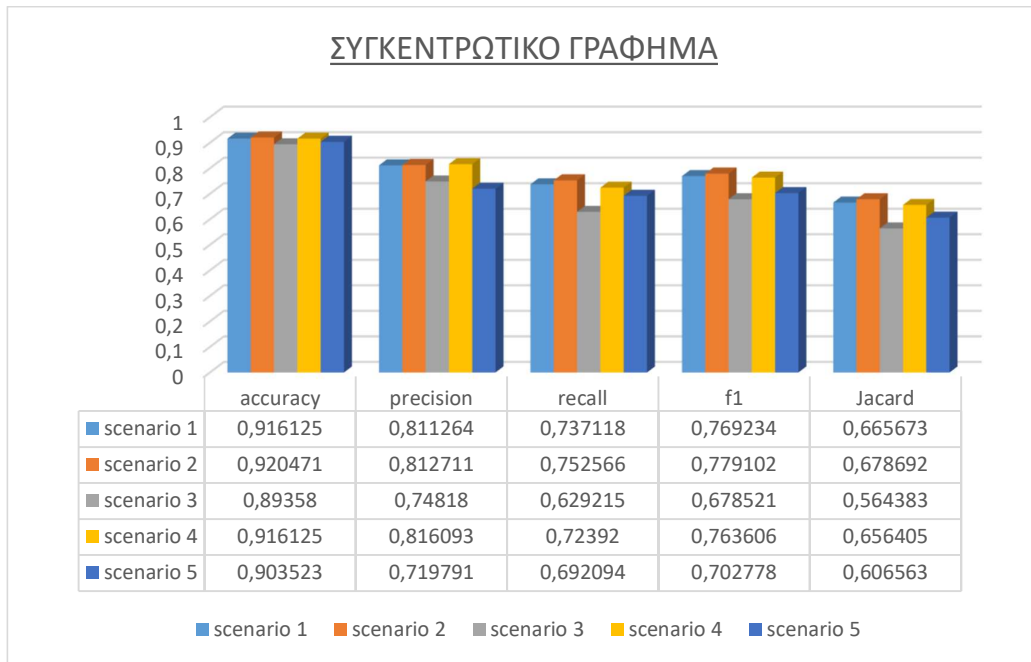
Ακολουθεί οπτική σύγκριση μιας εικόνας εισόδου (Nir) στο δίκτυο, με την μάσκα εκπαίδευσης (Ground Truth) όπως αντλήσαμε από την SpaceNet για την περιοχή μελέτης και την πρόβλεψη που κατάφερε το δίκτυο (Prediction). Παρατηρούμε σφάλματα και λάθη περισσότερα σε σύγκριση με τα προηγούμενα δίκτυα και λαμβάνοντας υπόψιν και τα στατιστικά, το παρόν σενάριο κρίνεται ως το λιγότερο αποδοτικό σε σύγκριση με τα υπόλοιπα.



Εικόνα 26. Οπτική σύγκριση αποτελέσματος πρόβλεψης και δεδομένων εισόδου 5ου σεναρίου.

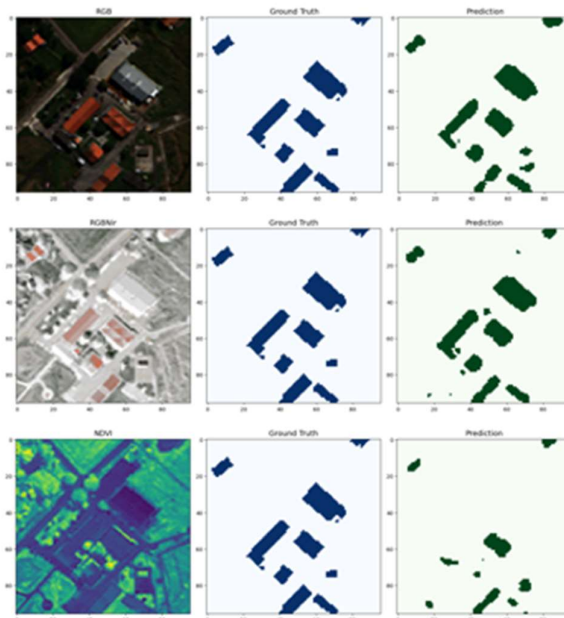
6.4 Ανάδειξη βέλτιστου σεναρίου

Από το συγκεντρωτικό γράφημα σύγκρισης μετρικών δεικτών των πέντε σεναρίων, τα σενάρια 3 και 5 είναι τα χειρότερα, ενώ ως καλύτερα αναδεικνύονται το πρώτο και το δεύτερο.



Γράφημα 16. Συγκεντρωτικό γράφημα δεικτών όλων των σεναρίων.

Τα αποτελέσματα, θεωρητικά εκφράζουν μια πολύ υψηλή απόδοση του μοντέλου, όμως πρέπει να συγκριθούν οπτικά προκειμένου να διασφαλιστεί η ανάδειξη του βέλτιστου σεναρίου. Επιλέχθηκαν για να συγκριθούν οι οπτικοποιήσεις δεδομένων εισόδου και πρόβλεψης από τα δύο στατιστικά βέλτιστα και το λιγότερο αποδοτικό σενάριο.



Εικόνα 27. Παράθεση δεδομένων εισόδου και πρόβλεψης 1ου, 2ου και 3ου σεναρίου

Στην πρώτη στήλη βλέπουμε τα δεδομένα εισόδου των σεναρίων, στη δεύτερη στήλη τις μάσκες ως δεδομένα εισόδου για την εκπαίδευση του δικτύου και στην τρίτη στήλη τα αποτελέσματα πρόβλεψης. Διαπιστώνουμε πως σίγουρα το σενάριο με τον NDVI παρουσιάζει τις μεγαλύτερες αδυναμίες, ενώ το δεύτερο σενάριο παρουσιάζει καλύτερα αποτελέσματα πρόβλεψης σε σύγκριση με το πρώτο.

Κλείνοντας το παρόν κεφάλαιο, ολοκληρώνεται το τεχνικό τμήμα εκπόνησης της εργασίας και ακολουθεί το τελευταίο κεφάλαιο στο οποίο αποτυπώνονται συμπεράσματα, δυσχέρειες και μελλοντικές επεκτάσεις της παρούσας μελέτης.

Συμπεράσματα

7.1 Γενικά

Ο στόχος της παρούσας διπλωματικής εργασίας επετεύχθη, καθώς μέσω της αξιολόγησης των αποτελεσμάτων εκπαίδευσης του πλήρους συνελκτικού νευρικού δικτύου U-Net, σε πέντε διαφορετικά σενάρια δεδομένων εκπαίδευσης, αναδείχθηκε βέλτιστο σενάριο για την εντοπισμό ανθρωπογενών κατασκευών σε μη κατοικημένες περιοχές.

7.2 Δυσχέρειες

Ως βασική δυσχέρεια, τεχνικού χαρακτήρα, αναφέρεται η αρχική δημιουργία εικόνων διαστάσεων 100x100 εικονοστοιχεία, ως δεδομένα εισόδου στο δίκτυο. Δύσκολα αντιλαμβάνεται κανείς ειδικά εάν είσαι νέος μελετητής ότι εφόσον το δίκτυο έχει πέντε συνελκτικά επίπεδα τόσο στον encoder όσο και στον decoder, πως οι εικόνες με τις εν λόγω διαστάσεις δεν δύναται να εισαχθούν στο μοντέλο. Επιλέξαμε λοιπόν διαστάσεις εικόνων 96x96 εικονοστοιχείων, ώστε να είναι σε θέση ο αλγόριθμος να υποδιπλασιάσει πέντε φορές το μέγεθος αυτό και στη συνέχεια να εκτελέσει το αντίστροφο, για την επίτευξη της ορθής του εκπαίδευσης.

Επίσης, άξιο λόγου είναι το γεγονός ότι δεν υπάρχουν πολλά ποιοτικά πακέτα δεδομένων ελεύθερα για έρευνα όπως αυτά της SpaceNet που χρησιμοποιήσαμε στην παρούσα. Το Ρίο ντε Τζανέιρο ως περιοχή μελέτης δεν μπορεί να χαρακτηριστεί ως μη κατοικημένη περιοχή, ωστόσο η δόμηση στο εν λόγω πακέτο δεδομένων (dataset) δεν είναι ιδιαίτερα πυκνή και η βλάστηση είναι έντονη. Συνεπώς χρησιμοποιήσαμε το συγκεκριμένο dataset κάνοντας μια παραδοχή.

7.3 Πορίσματα

Με την εκπόνηση της παρούσας πέραν του βέλτιστου μοντέλου αναδείχτηκε επίσης, η δυνατότητα επίλυσης δύσκολων και ουσιαστικών υπολογιστικών προβλημάτων με αλγόριθμους βαθιάς μάθησης που χρησιμοποιήθηκαν.

Ακόμη αναδείχθηκε το σχεδόν μηδενικό κόστος της μελέτης, αφού τα δεδομένα αντλήθηκαν από το διαδίκτυο και η επεξεργασία και η εξαγωγή των αποτελεσμάτων και των συμπερασμάτων πραγματοποιήθηκε σε χαμηλού κόστους υπολογιστική μονάδα, σε υψηλού επιπέδου (κοντά στο χρήστη) γλώσσα προγραμματισμού.

7.4 Μελλοντικές επεκτάσεις

Σαφώς αναδείχθηκε βέλτιστο σενάριο εκ των πέντε φασματικών σεναρίων που ορίστηκαν με σκοπό την τριβή του νέου μελετητή με αλγορίθμους μηχανικής μάθησης και εφαρμογές όρασης υπολογιστών σε συνδυασμό με τηλεπισκοπικές αρχές. Πέραν αυτού η παρούσα μελέτη θα μπορούσε να συνεχιστεί με σκοπό την βελτιστοποίηση του δικτύου από πλευράς δεδομένων εκπαίδευσής του, όσο και μέσω της σύγκρισής του με άλλες μεθόδους αυτοματοποίησης.

Η εξασφάλιση μεθόδων αυτοματοποίησης εντοπισμού αντικειμένων όπως αναφέραμε και στην εισαγωγή έχει να συνδράμει πολλά σε διεπιστημονικό επίπεδο. Χαρακτηριστικό παράδειγμα αποτελεί η μελέτη που αναφέρθηκε στο κεφάλαιο της βιβλιογραφικής ανασκόπησης και αφορούσε χαρτογράφηση άτυπων οικισμών [11] καθώς αυτό συνδράμει σε αρκετούς από τους στόχους βιώσιμης ανάπτυξης των Ηνωμένων Εθνών.

Κλείνοντας, ο αλγόριθμος της παρούσας μελέτης θα μπορούσε να συνδυαστεί με αλγορίθμους ταχείας χαρτογράφησης (rapid mapping). Η ταχεία χαρτογράφηση είναι μια διαδικασία γρήγορης δημιουργίας χαρτών [36], συχνά για την αντιμετώπιση καταστάσεων έκτακτης ανάγκης, φυσικών καταστροφών ή άλλων επειγουσών καταστάσεων. Ο κύριος σκοπός της ταχείας χαρτογράφησης είναι να παρέχει έγκαιρες και ακριβείς γεωγραφικές πληροφορίες, οι οποίες μπορούν να υποστηρίξουν τη λήψη αποφάσεων και τη διανομή πόρων κατά τη διάρκεια κρίσεων.

Αναφορές

- [1] B. Cheng κ.ά., ‘Multi-Class Objects Detection Method in Remote Sensing Image Based on Direct Feedback Control for Convolutional Neural Network’, *IEEE Access*, τ. PP, σσ. 1–1, Σεπτεμβρίου 2019, doi: 10.1109/ACCESS.2019.2943346.
- [2] G. Litjens κ.ά., ‘A survey on deep learning in medical image analysis’, *Med. Image Anal.*, τ. 42, σσ. 60–88, Δεκεμβρίου 2017, doi: 10.1016/j.media.2017.07.005.
- [3] R. Szeliski, *Computer Vision: Algorithms and Applications*. στο Texts in Computer Science. London: Springer, 2011. doi: 10.1007/978-1-84882-935-0.
- [4] ‘SpaceNet 1: Building Detection v1’. Ημερομηνία πρόσβασης: 26 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://spacenet.ai/spacenet-buildings-dataset-v1/>
- [5] O. Ronneberger, P. Fischer, και T. Brox, ‘U-Net: Convolutional Networks for Biomedical Image Segmentation’. arXiv, 18 Μάιος 2015. doi: 10.48550/arXiv.1505.04597.
- [6] Ö. Çiçek, A. Abdulkadir, S. S. Lienkamp, T. Brox, και O. Ronneberger, ‘3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation’. arXiv, 21 Ιούνιος 2016. doi: 10.48550/arXiv.1606.06650.
- [7] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, και Y. Gong, ‘Locality-constrained Linear Coding for image classification’, στο *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Ιουνίου 2010, σσ. 3360–3367. doi: 10.1109/CVPR.2010.5540018.
- [8] ‘Spectral library creation and analysis of urban built-up surfaces and materials using field spectrometry | Arabian Journal of Geosciences’. Ημερομηνία πρόσβασης: 26 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://link.springer.com/article/10.1007/s12517-021-06723-1>
- [9] T. Clements, S. Suon, D. S. Wilkie, και E. J. Milner-Gulland, ‘Impacts of Protected Areas on Local Livelihoods in Cambodia’, *World Dev.*, τ. 64, σσ. S125–S134, Δεκεμβρίου 2014, doi: 10.1016/j.worlddev.2014.03.008.
- [10] D. Quispe και J. Sulla-Torres, ‘Automatic Building Change Detection on Aerial Images using Convolutional Neural Networks and Handcrafted Features’, *Int. J. Adv. Comput. Sci. Appl.*, τ. 11, Ιανουαρίου 2020, doi: 10.14569/IJACSA.2020.0110683.
- [11] P. Helber κ.ά., ‘Mapping Informal Settlements in Developing Countries with Multi-resolution, Multi-spectral Data’. arXiv, 30 Νοέμβριος 2018. Ημερομηνία πρόσβασης: 26 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <http://arxiv.org/abs/1812.00812>
- [12] M. Molinier, J. Laaksonen, και T. Hame, ‘Detecting Man-Made Structures and Changes in Satellite Imagery With a Content-Based Information Retrieval System Built on Self-Organizing Maps’, *IEEE Trans. Geosci. Remote Sens.*, τ. 45, τχ. 4, σσ. 861–874, Απριλίου 2007, doi: 10.1109/TGRS.2006.890580.
- [13] D. Xiang, T. Tang, Y. Ban, και Y. Su, ‘Man-Made Target Detection from Polarimetric SAR Data via Nonstationarity and Asymmetry’, *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.*, τ. 9, τχ. 4, σσ. 1459–1469, Απριλίου 2016, doi: 10.1109/JSTARS.2016.2520518.

- [14] T. Mitchell, B. Buchanan, G. DeJong, T. Dietterich, P. Rosenbloom, και A. Waibel, 'Machine Learning', *Annu. Rev. Comput. Sci.*, τ. 4, τχ. Volume 4, 1990, σσ. 417–433, Ιουνίου 1990, doi: 10.1146/annurev.cs.04.060190.002221.
- [15] A. Banerjee, A. Bej, και T. N. Chatterjee, 'On the existence of a long range correlation in the Geomagnetic Disturbance storm time (Dst) index', *Astrophys. Space Sci.*, τ. 337, τχ. 1, σσ. 23–32, Ιανουαρίου 2012, doi: 10.1007/s10509-011-0836-1.
- [16] T. M. Mitchell, *Machine learning*, Nachdr. στο McGraw-Hill series in Computer Science. New York: McGraw-Hill, 2013.
- [17] Y. LeCun, Y. Bengio, και G. Hinton, 'Deep learning', *Nature*, τ. 521, τχ. 7553, σσ. 436–444, Μαΐου 2015, doi: 10.1038/nature14539.
- [18] A. F. Agarap, 'Deep Learning using Rectified Linear Units (ReLU)', *ArXiv180308375 Cs Stat*, Φεβρουαρίου 2019, Ημερομηνία πρόσβασης: 15 Ιούνιος 2021. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <http://arxiv.org/abs/1803.08375>
- [19] Y. LeCun και Y. Bengio, 'Convolutional networks for images, speech, and time series', *Handb. Brain Theory Neural Netw.*, τ. 3361, τχ. 10, σ. 1995, 1995.
- [20] J. Long, E. Shelhamer, και T. Darrell, 'Fully Convolutional Networks for Semantic Segmentation'. arXiv, 8 Μάρτιος 2015. doi: 10.48550/arXiv.1411.4038.
- [21] V. Dumoulin και F. Visin, 'A guide to convolution arithmetic for deep learning', *ArXiv160307285 Cs Stat*, Ιανουαρίου 2018, Ημερομηνία πρόσβασης: 15 Ιούνιος 2021. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <http://arxiv.org/abs/1603.07285>
- [22] Q. Zhao κ.ά., 'An Overview of the Applications of Earth Observation Satellite Data: Impacts and Future Trends', *Remote Sens.*, τ. 14, σ. 1863, Απριλίου 2022, doi: 10.3390/rs14081863.
- [23] 'WorldView-2 Satellite Sensor | Satellite Imaging Corp'. Ημερομηνία πρόσβασης: 20 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://www.satimagingcorp.com/satellite-sensors/worldview-2/>
- [24] 'Band Definition | GIS Dictionary'. Ημερομηνία πρόσβασης: 26 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://support.esri.com/en-us/gis-dictionary/band>
- [25] G. van Rossum (Guido), 'Python reference manual', *Department of Computer Science [CS]*, τχ. R 9525. CWI, 1 Ιανουάριος 1995. Ημερομηνία πρόσβασης: 18 Οκτώβριος 2022. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://ir.cwi.nl/pub/5008>
- [26] M. Canesche, L. Bragança, O. P. V. Neto, J. A. Nacif, και R. Ferreira, 'Google Colab CAD4U: Hands-On Cloud Laboratories for Digital Design', στο *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, Φεβρουαρίου 2021, σσ. 1–5. doi: 10.1109/ISCAS51556.2021.9401151.
- [27] P. Goldsborough, 'A Tour of TensorFlow'. arXiv, 1 Οκτώβριος 2016. doi: 10.48550/arXiv.1610.01178.
- [28] A. Gulli και S. Pal, *Deep Learning with Keras*. Packt Publishing Ltd, 2017.
- [29] W. McKinney, *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython*. O'Reilly Media, Inc., 2012.
- [30] S. van der Walt, S. C. Colbert, και G. Varoquaux, 'The NumPy Array: A Structure for Efficient Numerical Computation', *Comput. Sci. Eng.*, τ. 13, τχ. 2, σσ. 22–30, Μαρτίου 2011, doi: 10.1109/MCSE.2011.37.

- [31] M. Mommert, 'Dealing with Geospatial Raster Data in Python with Rasterio', Medium. Ημερομηνία πρόσβασης: 21 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://medium.com/@mommermiscience/dealing-with-geospatial-raster-data-in-python-with-rasterio-775e5ba0c9f5>
- [32] 'About Us'. Ημερομηνία πρόσβασης: 20 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://spacenet.ai/about-us/>
- [33] D. P. Kingma και J. Ba, 'Adam: A Method for Stochastic Optimization', *ArXiv1412.6980 Cs*, Ιανουαρίου 2017, Ημερομηνία πρόσβασης: 15 Ιούνιος 2021. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <http://arxiv.org/abs/1412.6980>
- [34] J. Duchi, E. Hazan, και Y. Singer, 'Adaptive subgradient methods for online learning and stochastic optimization.', *J. Mach. Learn. Res.*, τ. 12, τχ. 7, 2011.
- [35] T. Tieleman και G. Hinton, 'Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude', *COURSERA Neural Netw. Mach. Learn.*, τ. 4, τχ. 2, σσ. 26–31, 2012.
- [36] 'Rapid Mapping: geomatics role and research opportunities | Rendiconti Lincei. Scienze Fisiche e Naturali'. Ημερομηνία πρόσβασης: 26 Μάιος 2024. [Έκδοση σε ψηφιακή μορφή]. Διαθέσιμο στο: <https://link.springer.com/article/10.1007/s12210-015-0410-9>

Παραρτήματα

A. Μοντέλα εκπαίδευσης δικτύου U-Net σε πέντε σενάρια

https://drive.google.com/drive/folders/1_ZxC6K-kl0OgC4SDNRvIR1cjAEvq180

B. Κώδικας σε γλώσσα προγραμματισμού Python

```
#connecting
from google.colab import drive
drive.mount('/content/drive')
```

```
#import libraries
import tensorflow as tf
import keras
import numpy as np
import cv2
import glob
from datetime import datetime
```

```
# network structure
FILTER_NUM = 16 # number of basic filters for the first layer
FILTER_SIZE = 3 # size of the convolutional filter
UP_SAMP_SIZE = 2 # size of upsampling filters

'''
A few useful metrics and losses
'''

def dice_coef(y_true, y_pred):
    y_true_f = K.flatten(y_true)
    y_pred_f = K.flatten(y_pred)
    intersection = K.sum(y_true_f * y_pred_f)
    return (2.0 * intersection + 1.0) / (K.sum(y_true_f) + K.sum(y_pred_f) + 1.0)

def jacard_coef(y_true, y_pred):
    y_true_f = K.flatten(y_true)
    y_pred_f = K.flatten(y_pred)
    intersection = K.sum(y_true_f * y_pred_f)
    return (intersection + 1.0) / (K.sum(y_true_f) + K.sum(y_pred_f) - intersection + 1.0)

def jacard_coef_loss(y_true, y_pred):
```

```

    return -jaccard_coef(y_true, y_pred)

def dice_coef_loss(y_true, y_pred):
    return -dice_coef(y_true, y_pred)

#####
'''
Useful blocks to build Unet
conv - BN - Activation - conv - BN - Activation - Dropout (if enabled)
'''

def conv_block(x, filter_size, size, dropout, batch_norm=False):

    conv = layers.Conv2D(size, (filter_size, filter_size), padding="same")(x)
    if batch_norm is True:
        conv = layers.BatchNormalization(axis=3)(conv)
    conv = layers.Activation("relu")(conv)

    conv = layers.Conv2D(size, (filter_size, filter_size), padding="same")(conv)
    if batch_norm is True:
        conv = layers.BatchNormalization(axis=3)(conv)
    conv = layers.Activation("relu")(conv)

    if dropout > 0:
        conv = layers.Dropout(dropout)(conv)

    return conv

def repeat_elem(tensor, rep):
    # lambda function to repeat Repeats the elements of a tensor along an axis
    #by a factor of rep.
    # If tensor has shape (None, 256,256,3), lambda will return a tensor of shape
    #(None, 256,256,6), if specified axis=3 and rep=2.

    return layers.Lambda(lambda x, repnum: K.repeat_elements(x, repnum, axis=3),
        arguments={'repnum': rep})(tensor)

def res_conv_block(x, filter_size, size, dropout, batch_norm=False):
    '''
    Residual convolutional layer.
    Two variants....
    Either put activation function before the addition with shortcut
    or after the addition (which would be as proposed in the original resNet).

```

1. conv - BN - Activation - conv - BN - Activation
- shortcut - BN - shortcut+BN
2. conv - BN - Activation - conv - BN
- shortcut - BN - shortcut+BN - Activation

Check fig 4 in <https://arxiv.org/ftp/arxiv/papers/1802/1802.06955.pdf>
'''

```
conv = layers.Conv2D(size, (filter_size, filter_size), padding='same')(x)
if batch_norm is True:
    conv = layers.BatchNormalization(axis=3)(conv)
conv = layers.Activation('relu')(conv)

conv = layers.Conv2D(size, (filter_size, filter_size), padding='same')(conv)
if batch_norm is True:
    conv = layers.BatchNormalization(axis=3)(conv)
#conv = layers.Activation('relu')(conv) #Activation before addition with shortcut
if dropout > 0:
    conv = layers.Dropout(dropout)(conv)

shortcut = layers.Conv2D(size, kernel_size=(1, 1), padding='same')(x)
if batch_norm is True:
    shortcut = layers.BatchNormalization(axis=3)(shortcut)

res_path = layers.add([shortcut, conv])
res_path = layers.Activation('relu')(res_path) #Activation after addition with
shortcut (Original residual block)
return res_path

def gating_signal(input, out_size, batch_norm=False):
    """
    resize the down layer feature map into the same dimension as the up layer feature
    map
    using 1x1 conv
    :return: the gating feature map with the same dimension of the up layer feature
    map
    """
    x = layers.Conv2D(out_size, (1, 1), padding='same')(input)
    if batch_norm:
        x = layers.BatchNormalization()(x)
    x = layers.Activation('relu')(x)
    return x
```

```

def attention_block(x, gating, inter_shape):
    shape_x = K.int_shape(x)
    shape_g = K.int_shape(gating)

    # Getting the x signal to the same shape as the gating signal
    theta_x = layers.Conv2D(inter_shape, (2, 2), strides=(2, 2), padding='same')(x) # 16
    shape_theta_x = K.int_shape(theta_x)

    # Getting the gating signal to the same number of filters as the inter_shape
    phi_g = layers.Conv2D(inter_shape, (1, 1), padding='same')(gating)
    upsample_g = layers.Conv2DTranspose(inter_shape, (3, 3),
                                        strides=(shape_theta_x[1] // shape_g[1], shape_theta_x[2] //
shape_g[2]),
                                        padding='same')(phi_g) # 16

    concat_xg = layers.add([upsample_g, theta_x])
    act_xg = layers.Activation('relu')(concat_xg)
    psi = layers.Conv2D(1, (1, 1), padding='same')(act_xg)
    sigmoid_xg = layers.Activation('sigmoid')(psi)
    shape_sigmoid = K.int_shape(sigmoid_xg)
    upsample_psi = layers.UpSampling2D(size=(shape_x[1] // shape_sigmoid[1],
shape_x[2] // shape_sigmoid[2]))(sigmoid_xg) # 32

    upsample_psi = repeat_elem(upsample_psi, shape_x[3])

    y = layers.multiply([upsample_psi, x])

    result = layers.Conv2D(shape_x[3], (1, 1), padding='same')(y)
    result_bn = layers.BatchNormalization()(result)
    return result_bn

def UNet(input_shape, NUM_CLASSES=1, dropout_rate=0.0, batch_norm=True):
    """
    UNet,
    """

    inputs = layers.Input(input_shape, dtype=tf.float32)

    # Downsampling layers
    # DownRes 1, convolution + pooling
    conv_128 = conv_block(inputs, FILTER_SIZE, FILTER_NUM, dropout_rate,
batch_norm)
    pool_64 = layers.MaxPooling2D(pool_size=(2,2))(conv_128)

```

```
# DownRes 2
conv_64 = conv_block(pool_64, FILTER_SIZE, 2*FILTER_NUM, dropout_rate,
batch_norm)
pool_32 = layers.MaxPooling2D(pool_size=(2,2))(conv_64)
# DownRes 3
conv_32 = conv_block(pool_32, FILTER_SIZE, 4*FILTER_NUM, dropout_rate,
batch_norm)
pool_16 = layers.MaxPooling2D(pool_size=(2,2))(conv_32)
# DownRes 4
conv_16 = conv_block(pool_16, FILTER_SIZE, 8*FILTER_NUM, dropout_rate,
batch_norm)
pool_8 = layers.MaxPooling2D(pool_size=(2,2))(conv_16)
# DownRes 5, convolution only
conv_8 = conv_block(pool_8, FILTER_SIZE, 16*FILTER_NUM, dropout_rate,
batch_norm)

# Upsampling layers

up_16 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(conv_8)
up_16 = layers.concatenate([up_16, conv_16], axis=3)
up_conv_16 = conv_block(up_16, FILTER_SIZE, 8*FILTER_NUM, dropout_rate,
batch_norm)
# UpRes 7

up_32 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(up_conv_16)
up_32 = layers.concatenate([up_32, conv_32], axis=3)
up_conv_32 = conv_block(up_32, FILTER_SIZE, 4*FILTER_NUM, dropout_rate,
batch_norm)
# UpRes 8

up_64 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(up_conv_32)
up_64 = layers.concatenate([up_64, conv_64], axis=3)
up_conv_64 = conv_block(up_64, FILTER_SIZE, 2*FILTER_NUM, dropout_rate,
batch_norm)
# UpRes 9

up_128 = layers.UpSampling2D(size=(UP_SAMP_SIZE, UP_SAMP_SIZE),
data_format="channels_last")(up_conv_64)
up_128 = layers.concatenate([up_128, conv_128], axis=3)
up_conv_128 = conv_block(up_128, FILTER_SIZE, FILTER_NUM, dropout_rate,
batch_norm)
```



```

# 1*1 convolutional layers

conv_final = layers.Conv2D(NUM_CLASSES, kernel_size=(1,1))(up_conv_128)
conv_final = layers.BatchNormalization(axis=3)(conv_final)
conv_final = layers.Activation('sigmoid')(conv_final) #Change to softmax for
multichannel

# Model
model = models.Model(inputs, conv_final, name="UNet")
print(model.summary())
return model

```

#Reading the Data

```

path_8band = '/content/drive/MyDrive/dataset/8band/'
path_8band_masks = '/content/drive/MyDrive/dataset/output_masks_8band/'
IMG_WIDTH = 96
IMG_HEIGHT = 96

```

```

! pip install rasterio
import rasterio
from matplotlib import pyplot as plt
from rasterio.windows import Window
import numpy
import glob

```

initialize path x-train

```

path_for_xtrain = sorted(glob.glob(path_8band + '*tif'), key=lambda x:
int(".join(filter(str.isdigit, x))))

```

```

# arxikopoihsh ths listas tw n eikonwn
X_train = []

```

```

# gia kathe directory sthn lista tw n directories tw n eikonwn
for trainingxfile in path_for_xtrain:
print(trainingxfile)

```

```

# me to rasterio anoikse thn eikona
with rasterio.open(trainingxfile) as src:
# pare ena parathyro 96x96
img = src.read(window=Window(0, 0, IMG_WIDTH, IMG_HEIGHT))

```

```

# apomonose ta kanalia

```

```

red = bandeq(img[4])
green = bandeq(img[2])
blue = bandeq(img[1])
# nir = bandeq(img[6]) # ayto to kleineis kai to anoigeis gia na ftiakseis ta senaria
einai to NIR

```

```

#edw ftiaxneis ton ndvi
# numerator=numpy.subtract(nir,red)
# denominator=numpy.add(nir,red) + 1
# ndvi=numpy.divide(numerator,denominator)
# ndvi = bandeq(ndvi)
# print(ndvi.shape, ndvi.min(), ndvi.max(), ndvi.dtype)

```

```

# kane stack tis eikones me osa kai opia kanalia thes
stack = np.stack((red,green,blue), axis =2)
print(stack.shape, stack.min(), stack.max(), stack.dtype)

```

```

# apothikeyse thn stacked eikona sthn lista
X_train.append(stack)
#break

```

```

# numpy array
X_train = np.array(X_train)
print('X_train shape:', X_train.shape)

```

```

# save x-train
export_path = '/content/drive/MyDrive/dataset/3band_npy/'
np.save(export_path+'x_train_3band.npy', X_train)

```

```

# initialize path x-train
path_for_ytrain = sorted(glob.glob(path_8band_masks + '*tif'), key=lambda x:
int(".join(filter(str.isdigit, x))))

```

```

Y_train = []
for trainingyfile in path_for_ytrain:
    print(trainingyfile)
    # me to rasterio anoikse thn eikona
    with rasterio.open(trainingyfile) as src:
        # pare ena parathyro 100x100
        img = src.read(window=Window(0, 0, IMG_WIDTH, IMG_HEIGHT))
        img = numpy.rollaxis(img, 0, 3)
        print(img.shape)
        Y_train.append(img)
#break

```

```
Y_train = np.array(Y_train)
print('Y_train shape:', Y_train.shape)

# save y-train
export_path = '/content/drive/MyDrive/dataset/3band_npy/'
np.save(export_path+'y_train_3band.npy', Y_train)

#Train set
X_train = np.load('/content/drive/MyDrive/dataset/3band_npy/x_train_3band.npy')
print('X_train shape:', X_train.shape)
Y_train = np.load('/content/drive/MyDrive/dataset/3band_npy/y_train_3band.npy')
print('Y_train shape:', Y_train.shape)

# Create a mask for NaN values in either mask_array or s1_raw_array
nan_mask = np.any(np.isnan(Y_train) | np.isnan(X_train), axis=(-3, -2, -1))

# Filter out images with NaN values from both arrays
X_train_filtered = np.array(X_train[~nan_mask])
Y_train_filtered = np.array(Y_train[~nan_mask])

# Print the shapes of the filtered arrays
print("X_train_filtered shape:", X_train_filtered.shape)
print("Y_train_filtered shape:", Y_train_filtered.shape)

import sklearn
from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X_train_filtered , Y_train_filtered,
test_size=0.20, random_state=42, shuffle = True)

#Initial Parameters
IMG_WIDTH = X_train.shape[1]
IMG_HEIGHT = X_train.shape[2]
IMG_CHANNELS = X_train.shape[3]

num_labels = 1 #Binary
input_shape = (IMG_HEIGHT,IMG_WIDTH,IMG_CHANNELS)
print(input_shape)
batch_size = 16
epochs = 30

# run the model
start1 = datetime.now()
```

```
unet_model = UNet(input_shape)
unet_model.compile(optimizer='Adam',
loss='binary_crossentropy', metrics=['accuracy'])
unet_model.summary()

# Model checkpoint - save the model after every epoch
# Save best only - the latest best model will not be overwritten

# Fit model
# earlystopper_unet = keras.callbacks.EarlyStopping(patience=10, verbose=1)
# checkpointer_unet =
keras.callbacks.ModelCheckpoint('UNet_'+training_samples+'_mch.h5', verbose=1,
save_best_only=True)

###unet_results = unet_model.fit(X_train, y_train, validation_split=0.1,
batch_size=batch_size, epochs=epochs)
#callbacks=[checkpointer_unet])

# save the final model, once training is completed
u_net_model_name = '/content/drive/MyDrive/dataset/models/UNet_3band.h5'
###unet_model.save(u_net_model_name)

stop1 = datetime.now()
#Execution time of the model
execution_time_Unet = stop1-start1
print("UNet execution time is: ", execution_time_Unet)

from sklearn.metrics import confusion_matrix
from sklearn.metrics import jaccard_score
import sklearn

unet_model = tf.keras.models.load_model(u_net_model_name) ###

acc = []
precision = []
recall = []
f1 = []
jac = []
k = []

start1_test = datetime.now()
for idx in range(0,X_test.shape[0]):
    mask = y_test[idx]
```

```

prediction = unet_model.predict(np.expand_dims(X_test[idx], axis=0))
prediction = np.squeeze(prediction, axis=0)
ret, thresh_pred = cv2.threshold(prediction, 0.5, 1, cv2.THRESH_BINARY)
ret2, mask = cv2.threshold(mask, 0.5, 1, cv2.THRESH_BINARY)

mask = list(mask.ravel())
thresh_pred = list(thresh_pred.ravel())

cm = confusion_matrix(mask, thresh_pred)

if cm.shape == (2,2):
    #print(cm)

    # TP = cm[0][0]
    # FP = cm[0][1]
    # FN = cm[1][0]
    # TN = cm[1][1]

    accuracyscore = sklearn.metrics.accuracy_score(mask,thresh_pred)
    precisionscore = sklearn.metrics.precision_score(mask,thresh_pred,
average='macro')
    recallscore = sklearn.metrics.recall_score(mask,thresh_pred, average='macro')
    f1score = 2*(precisionscore*recallscore)/(precisionscore + recallscore)
    jaccard = jaccard_score(mask,thresh_pred, average='macro')
    # kappa = 2*((TP*TN) - (FN*FP))/((TP+FP)*(FP+TN)+(TP+FN)*(FN+TN))

    print('stats ; ',idx, ';', accuracyscore, ';', precisionscore, ';', recallscore, ';', f1score,
';', jaccard)
    acc.append(accuracyscore)
    precision.append(precisionscore)
    recall.append(recallscore)
    f1.append(f1score)
    jac.append(jaccard)

stop1_test = datetime.now()
#Execution time of the model
execution_time_Unet_test = stop1_test-start1_test
print("UNet testing time is: ", execution_time_Unet_test)

acc = np.array(acc)
precision = np.array(precision)
recall= np.array(recall)
f1 = np.array(f1)
jac = np.array(jac)

```

```
# k = np.array(k)
print("")
print("averages:", "accuracy",          np.average(acc),          "precision",
      np.average(precision), "recall",    np.average(recall), "f1",    np.average(f1), "jac",
      np.average(jac))
```

```
#plot sample
index_to_visualize = 1368 # Replace 0 with the desired index
```

```
prediction = unet_model.predict(np.expand_dims(X_test[index_to_visualize],
axis=0))
prediction = np.squeeze(prediction, axis=0)
ret, thresh_pred = cv2.threshold(prediction, 0.5, 1, cv2.THRESH_BINARY)
```

```
fig, axs = plt.subplots(1, 3, figsize=(15, 5))
```

```
axs[0].imshow(X_test[index_to_visualize])
axs[0].set_title('RGB', fontsize=14)
axs[0].axis('on')
```

```
axs[1].imshow(y_test[index_to_visualize], cmap='Blues')
axs[1].set_title('Ground Truth', fontsize=14)
axs[1].axis('on')
```

```
axs[2].imshow(thresh_pred, cmap='Greens')
axs[2].set_title('Prediction', fontsize=14)
axs[2].axis('on')
```

```
plt.tight_layout()
# plt.savefig('DATASET.jpg', format='jpg', dpi=600)
# plt.savefig('SWIR-DATA.eps', format='eps')
# plt.savefig('SWIR-DATA.svg', format='svg')
```

```
plt.show()
```

```
# save statistics
import pandas as pd
data = {'accuracy': acc, 'precision': precision, 'recall': recall, 'f1': f1, 'jac': jac}
print(data)
df = pd.DataFrame(data, columns=['accuracy', 'precision', 'recall', 'f1', 'jac' ])

df.to_csv('/content/drive/MyDrive/dataset/csv_results/rgb_unet_results.csv')
```

```
df.describe().to_csv('/content/drive/MyDrive/dataset/csv_results/rgb_unet_results_
describe.csv')
df.describe()
```

```
#plot the training and validation accuracy and loss at each epoch
loss = unet_results.history['loss']
val_loss = unet_results.history['val_loss']
epochs = range(1, len(loss) + 1)
plt.plot(epochs, loss, 'y', label='Training loss')
plt.plot(epochs, val_loss, 'r', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()

acc = unet_results.history['accuracy']
val_acc = unet_results.history['val_accuracy']

plt.plot(epochs, acc, 'y', label='Training Accuracy')
plt.plot(epochs, val_acc, 'r', label='Validation Accuracy')
plt.title('Training and validation Accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()
plt.show()
```