



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών  
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

**Ευφυείς Τεχνικές για την Ταξινόμηση Μαγνητικών Τομογραφιών σε  
Καρκινοπαθείς**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΚΥΡΙΑΚΟΣ-ΠΑΝΑΓΙΩΤΗΣ ΑΝΔΡΕΣΑΚΗΣ**

**Επιβλέπων :** Ανδρέας - Γεώργιος Σταφυλοπάτης  
Καθηγητής Ε.Μ.Π.

**Συνεπιβλέπων :** Θάνος Τάγαρης  
Υποψήφιος Διδάκτορας Ε.Μ.Π.

Αθήνα, Ιούλιος 2018





## ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών  
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

### Ευφυείς Τεχνικές για την Ταξινόμηση Μαγνητικών Τομογραφιών σε Καρκινοπαθείς

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΚΥΡΙΑΚΟΣ-ΠΑΝΑΓΙΩΤΗΣ ΑΝΔΡΕΣΑΚΗΣ**

**Επιβλέπων :** Ανδρέας - Γεώργιος Σταφυλοπάτης  
Καθηγητής Ε.Μ.Π.

**Συνεπιβλέπων :** Θάνος Τάγαρης  
Υποψήφιος Διδάκτορας Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 24<sup>η</sup> Ιουλίου 2018.

.....  
Ανδρέας-Γεώργιος Σταφυλοπάτης  
Καθηγητής Ε.Μ.Π.

.....  
Γεώργιος Στάμου  
Αναπληρωτής Καθηγητής Ε.Μ.Π.

.....  
Κωνσταντίνα Νικήτα  
Καθηγήτρια Ε.Μ.Π.

Αθήνα Ιούλιος 2018

.....  
**ΚΥΡΙΑΚΟΣ-ΠΑΝΑΓΙΩΤΗΣ ΑΝΔΡΕΣΑΚΗΣ**

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Κυριάκος-Παναγιώτης Ανδρεσάκης, 2018.

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.



## Περίληψη

---

Το 2007, σύμφωνα με την Αμερικανική Εταιρεία Καρκίνου [1] στην ετήσια έκθεση που παρουσίασε, ανακοινώθηκε ότι κάθε μέρα πέθαιναν από καρκίνο σε όλο τον κόσμο 20.000 άνθρωποι. Σε ένα χρόνο συνολικά παρουσιάζονταν 7,6 εκατομμύρια περίπου θάνατοι τέτοιας αιτίας. Επίσης, κατά το ίδιο έτος, τουλάχιστον 12,3 εκατομμύρια νέα κρούσματα καρκίνου είχαν διαγνωστεί. Η πλειονότητα των νέων κρουσμάτων, 6,7 εκατ. και των θανάτων 4,7 εκατ. έχουν σημειωθεί στις αναπτυσσόμενες χώρες, ενώ 5,4 εκατ. κρούσματα και 2,9 εκατ. θάνατοι έχουν σημειωθεί στις οικονομικά ανεπτυγμένες χώρες. Σύμφωνα με τον Παγκόσμιο Οργανισμό Υγείας, ο αριθμός των θανάτων από καρκίνο αναμένεται να ξεπεράσει τα 10 εκατομμύρια το 2020 και οι νέες περιπτώσεις της νόσου να αυξηθούν στα 16 εκατομμύρια!

Η έγκαιρη ανίχνευση καρκινικών κυττάρων αυξάνει το προσδόκιμο ζωής και τις πιθανότητες επιβίωσης των ασθενών. Σύγχρονοι τρόποι επίτευξης του παραπάνω σκοπού προϋποθέτουν συστήματα όπως ηλεκτρονικούς ή μαγνητικούς τομογράφους, με σκοπό τον εντοπισμό τέτοιων κυττάρων σε εικόνες. Συγκεκριμένα, όσον αφορά τον ανθρώπινο εγκέφαλο, οι ειδικοί, αναζητούν πιθανές ανωμαλίες στο μαλακό ιστό, τη φαιά ουσία, τη λευκή ουσία ή το εγκεφαλονωτιαίο υγρό, εξετάζοντας τις παραπάνω εικόνες. Η διαδικασία, παρ' όλ' αυτά, η οποία καλείται να φέρει εις πέρας ο ειδήμων, είναι χρονοβόρα, διαφέρει σε μεγάλο βαθμό ανάλογα με τη συγκεκριμένη περίπτωση που έχει να αντιμετωπίσει, και τέλος εξαρτάται από την ατομική του ικανότητα στην επίλυση του προβλήματος. Αντίθετα, η ανάλυση τέτοιων εικόνων με χρήση ηλεκτρονικού υπολογιστή, συντελεί σε ταχύτερη λήψη απόφασης, γενίκευση του προβλήματος, ευκολότερη δημιουργία αρχείων για τους ασθενείς, ασφαλέστερη διάγνωση και βοήθα στη δημιουργία πλάνου θεραπείας. Η επίλυση του ανωτέρου προβλήματος με το δεύτερο τρόπο, δηλαδή με χρήση ηλεκτρονικού υπολογιστή, μπορεί να συμβεί μέσω διαφόρων μοντέλων μηχανικής μάθησης. Δηλαδή, συστημάτων που βασίζονται στη χρήση αλγορίθμων και ανάλυση πειραματικών δεδομένων, με σκοπό την πρόβλεψη και τη λήψη απόφασης, μέσω της αναγνώρισης χαρακτηριστικών. Τα υπολογιστικά μοντέλα με τη μεγαλύτερη ακρίβεια στην αναγνώριση χαρακτηριστικών, ειδικά στην περίπτωση των εικόνων, είναι τα νευρωνικά δίκτυα [2] και συγκεκριμένα μία υποκατηγορία αυτών, η οποία εξειδικεύεται στην επεξεργασία εικόνας, τα συνελκτικά νευρωνικά δίκτυα [3].

Σκοπός της παρούσας διπλωματικής εργασίας είναι η σύγκριση ανάμεσα σε αρχιτεκτονικές βαθιών συνελκτικών νευρωνικών δικτύων [4], μεθόδων προεπεξεργασίας δεδομένων [5], αλλά και άλλων παραμέτρων όπως αλγορίθμων μάθησης [6], με σκοπό τη βέλτιστη κατηγοριοποίηση μαγνητικών τομογραφιών σε δύο ομάδες: γλοιώματα χαμηλής κακοήθειας και γλοιώματα υψηλής κακοήθειας. Τα μοντέλα που υλοποιήθηκαν, εκπαιδεύτηκαν, έτσι ώστε να μεγιστοποιηθεί η πιθανότητα επιτυχίας στην κατηγοριοποίηση, δεδομένης μίας εικόνας εισόδου. Πειράματα σε μια μεγάλη βάση δεδομένων για εκπαίδευση, αξιολόγηση και έλεγχο λειτουργίας, όπως είναι η βάση δεδομένων του διαγωνισμού BRATS2015 [7], την οποία και χρησιμοποιήσαμε, αποδεικνύουν την ακρίβεια των παραπάνω μοντέλων, γεγονός που επαληθεύουμε ποσοτικά. Συγκεκριμένα, πέρα από αρκετά μοντέλα τύπου ResNet50 [8] και VGGNet16 [9] εκπαιδεύσαμε και μία πιο ρηχή αρχιτεκτονική που υλοποιήθηκε στα πλαίσια της παρούσας εργασίας, με σκοπό τη σύγκρισή της με τις παραπάνω πιο βαθιές αρχιτεκτονικές. Όσον αφορά τις μεθόδους προεπεξεργασίας πειραματιστήκαμε με δύο διαφορετικές μεθόδους εξομάλυνσης-κανονικοποίησης δεδομένων, αλλά και με τις μεθόδους ανισοτροπικής διάχυσης και διόρθωσης πεδίου πόλωσης. Τέλος, χρησιμοποιήσαμε δύο διαφορετικούς αλγορίθμους μάθησης Adam [10] και SGD [11].

**Λέξεις-κλειδιά:** ανίχνευση καρκινικών κυττάρων, ανθρώπινος εγκέφαλος, κατηγοριοποίηση, εκπαίδευση, βάση δεδομένων, βαθιά-ρηχή αρχιτεκτονική, μέθοδοι προεπεξεργασίας, BRATS2015, VGGNet16, ResNet50 εξομάλυνση-κατηγοριοποίηση, ανισοτροπική διάχυση, διόρθωση πεδίου πόλωσης, αλγόριθμοι μάθησης, Adam, SGD



## *Abstract*

---

In 2007, according to the American Cancer Association [1] in its annual report, it was announced that every day 20,000 people died of cancer. Every year, there were about 7.6 million deaths of this kind. Also, in the same year, at least 12.3 million new cases of cancer were diagnosed. The majority of new cases, 6.7 million and 4.7 million deaths, have occurred in developing countries, with 5.4 million cases and 2.9 million deaths in economically developed countries. According to the World Health Organization, the number of cancer deaths is expected to rise over 10 million every year in 2020 and new cases of disease will increase to 16 million!

Early detection of cancer cells increases the life expectancy and survival chances of patients. Modern ways of achieving the goal above, include systems such as electronic or magnetic tomographies, in order to identify such cells in images. Specifically, as far as the human brain is concerned, experts are looking for possible abnormalities in soft tissue, gray matter, white matter or cerebrospinal fluid by looking at these specific images. The process, however, which the expert is called upon to carry out, is time-consuming, varies greatly depending on the specific case he has to deal with, and ultimately depends on his/hers individual capacity to solve this specific problem. On the contrary, the analysis of such images using a computer, leads to faster decision making, generalizing the problem, easier archiving for patients, safer diagnosis and helping to create a plan for treatment. Solving the above problem, by utilizing the latter method, can occur through various models of machine learning. Such are systems, based on the use of algorithms and analysis of experimental data, for the purpose of predicting and deciding through attribute recognition. The most accurate computational models in image recognition challenges, especially on the field of image recognition, are neural networks [2] and, in particular, a subclass of those that specialize in image processing, called convolutional neural networks [3].

The purpose of this diploma these is to compare deep convolutional neural network architectures [4], preprocessing techniques [5] and several other parameters like learning algorithms [6], in order to optimally categorize MRIs in two groups: low-grade gliomas and high-grade gliomas. The models that were implemented were trained as to maximize the probability of success in categorization given an input image. Experiments in a large database for training, evaluation and operational control, such as the database of the [7] contest, which we have used, prove the accuracy of the above models. The previous statement was proved through results measured during this diploma thesis. In addition to several ResNet50 [8] and VGGNet16 [9] models, we have also trained a model based on a shallower architecture that has been designed specifically for this paper and in order for it to be compared with the above deeper architectures. As far as preprocessing methods are concerned, we have utilized two different types of data normalization and except from that two other preprocessing methods, anisotropic diffusion and bias field correction. Finally, we used two different learning algorithms named Adam [10] and SGD [11].

**Key-words** : cancer cell detection, human brain, categorize, training, database, deep-shallow architecture, preprocessing methods, BRATS2015, VGGNet16, ResNet50, normalization, anisotropic diffusion, bias field correction, learning algorithms, Adam, SGD





## Ευχαριστίες

---

Η παρούσα εργασία αποτελεί διπλωματική εργασία στα πλαίσια των σπουδών μου στο Εθνικό Μετσόβιο Πολυτεχνείο και σηματοδοτεί το τέλος μίας μεγάλης προσπάθειας. Η τελευταία προσπάθεια, καθώς και η συγγραφή της παρούσας διπλωματικής εργασίας, θα ήταν αδύνατες δίχως τη βοήθεια, τη συμπαράσταση και τη στήριξη ανθρώπων, τους οποίους και θα ήθελα να ευχαριστήσω σε αυτό το σημείο.

Αρχικά, οφείλω να ευχαριστήσω τον κ. Ανδρέα – Γεώργιο Σταφυλλοπάτη, καθηγητή Ε.Μ.Π., για την ευκαιρία που μου έδωσε, αναθέτοντάς μου την παρούσα διπλωματική εργασία, αλλά και για την πολύτιμη βοήθεια και καθοδήγησή του κατά τη διάρκεια αυτής. Ιδιαίτερη ευγνωμοσύνη οφείλω επίσης στον υποψήφιο διδάκτορα Θάνο Τάγαρη, για την καθοδήγησή και στήριξή του σε όλη τη διάρκεια της εργασίας, αλλά και για τις χρήσιμες συμβουλές του, χωρίς τις οποίες η τελευταία δε θα ήταν δυνατόν να πραγματοποιηθεί. Στη συνέχεια θα ήθελα να ευχαριστήσω τους διδάκτορες Γιώργο Αλεξανδρίδη και Άρη Λαναρίδη, τόσο για τις συμβουλές τους όσο και για τη βοήθεια που μου παρείχαν στην αντιμετώπιση διαφόρων προβλημάτων που παρουσιάστηκαν κατά τη διάρκεια της εργασίας. Οφείλω επίσης, να ευχαριστήσω όλους τους διδάκτορες και υποψήφιους διδάκτορες του Εργαστηρίου Ευφυών Υπολογιστικών Συστημάτων για τη θετική τους διάθεση και το καλό κλίμα που επικρατεί στο Εργαστήριο, αλλά και για τη βοήθεια που μου παρείχαν κατά τη διάρκεια συγγραφής της παρούσας εργασίας.

Τέλος, θα ήθελα να ευχαριστήσω και όλους τους κοντινούς μου ανθρώπους, οι οποίοι με στήριξαν πριν και κατά τη διάρκεια των σπουδών μου. Αρχικά, θα ήθελα να ευχαριστήσω την οικογένειά μου και ειδικά τους γονείς μου Παύλο και Αργυρώ, οι οποίοι μου εξασφάλισαν το καλύτερο δυνατό περιβάλλον για τη φοιτητική μου ζωή και μου έδωσαν βοήθεια και στήριξη σε όλη αυτή την προσπάθεια. Επίσης θα ήθελα να ευχαριστήσω την αδερφή μου Άννα και τους φίλους μου που έχω την τύχη να γνωρίζω από μικρός και με βοήθησαν όπως σε πολλές στιγμές της ζωής μου και εδώ Μάνθο, Κώστα και Αντρέα, παρέχοντας απεριόριστη στήριξη και βοήθεια. Τελικά, θα ήθελα να ευχαριστήσω και τους φίλους που είχα την τύχη να γνωρίσω και να συναναστραφώ κατά τη διάρκεια των σπουδών μου Σπύρο, Μανώλη, Αγγελή, Χρήστο και Γιώργο, οι οποίοι με βοήθησαν και αυτοί όχι μόνο στην πραγματοποίηση της εργασίας, αλλά συνολικά στην προσπάθεια όλων αυτών των χρόνων για την ολοκλήρωση των σπουδών.



# Περιεχόμενα

---

<b>1. Κεφάλαιο 1<sup>ο</sup> Εισαγωγή</b> .....	1
1.1 Τεχνητή Νοημοσύνη.....	2
1.2 Μηχανική Μάθηση .....	3
1.2.1 Ιστορικά στοιχεία .....	3
1.2.2 Τεχνητά Νευρωνικά Δίκτυα .....	3
1.2.2.1 Ισχύς .....	3
1.2.2.2 Ευχρηστία.....	4
1.2.3 Προέλευση Τεχνητών Νευρωνικών Δικτύων .....	5
1.2.4 Εκπαίδευση Τεχνητών Νευρωνικών Δικτύων .....	7
1.2.4.1 Μάθηση με επιτήρηση .....	8
1.2.4.2 Μάθηση χωρίς επιτήρηση .....	8
1.2.5 Επιτηρούμενη Μάθηση Τ.Ν.Δ .....	8
1.3 Επεξεργασία Εικόνας.....	9
1.3.1 Image Large Scale Visual Recognition Challenge(ILSVRC) .....	3
<b>2. Κεφάλαιο 2<sup>ο</sup> Συνελικτικά Νευρωνικά Δίκτυα (CNNs)</b> .....	11
2.1 Σύγκριση Αρχιτεκτονικών.....	11
2.2 Τύποι Επιπέδων.....	13
2.3 Παραδείγματα Αρχιτεκτονικής.....	14
2.4 Συνελικτικό Επίπεδο.....	15
2.4.1 Αναλογία από πλευράς εγκεφάλου .....	16
2.4.2 Τοπική Συνδεσιμότητα .....	16
2.4.3 Χωρική Οργάνωση.....	17
2.4.4 Χρήση του zero-padding .....	19
2.4.5 Επιλογή στην τιμή του Stride .....	19
2.4.6 Μοίρασμα Παραμέτρων .....	20
2.4.7 Αλγόριθμος Ανάστροφης Διάδοσης.....	22
2.4.8 1x1 Συνέλιξη .....	23
2.4.9 Διασταλμένη Συνέλιξη .....	23
2.5 Επίπεδο Pooling.....	23
2.5.1 Αλγόριθμος Ανάστροφης Διάδοσης.....	25
2.5.2 Αποφυγή χρήσης επιπέδου Pooling .....	25

2.6 Πλήρως Συνδεδεμένα Επίπεδα .....	25
2.6.1 Μετατροπή πλήρως συνδεδεμένων επιπέδων σε συνελκτικά .....	26
2.6.2 Διαδικασία μετατροπής .....	26
2.7 Αρχιτεκτονικές Συνελκτικών Νευρωνικών Δικτύων .....	28
2.7.1 Επιλογή Μεγεθών και Παραμέτρων στα Επίπεδα .....	29
2.7.2 Επιλογή Τιμής για Stride $S=1$ .....	30
2.7.3 Χρήση zero-padding .....	30
2.7.4 Συμβιβασμός λόγω των περιορισμών μνήμης .....	30
2.8 Παράμετροι Μάθησης .....	31
2.8.1 Βελτιστοποίηση .....	31
2.8.2 Αλγόριθμοι Βελτιστοποίησης .....	32
2.8.2.1 SGD .....	32
2.8.2.2 Momentum .....	33
2.8.2.3 Adam .....	34
2.9 Διάφορες Αρχιτεκτονικές Συνελκτικών Νευρωνικών Δικτύων .....	34
<b>3. Κεφάλαιο 3<sup>ο</sup> Μοντέλα και Μέθοδοι Προεπεξεργασίας .....</b>	<b>36</b>
3.1 Μοντέλα Συνελκτικών Νευρωνικών Δικτύων .....	36
3.1.1 VGGNet .....	36
3.1.1.1 Αρχιτεκτονική VGGNet16 .....	36
3.1.2 ResNet .....	40
3.1.2.1 Αντιμετώπιση Μείωσης Απόδοσης .....	40
3.1.2.2 Χαρακτηριστικά υλοποίησης ResNet .....	41
3.1.2.3 Περιπτώσεις υπολειπόμενων συνδέσεων .....	43
3.1.2.4 Απόδοση .....	43
3.1.2.5 Τύποι αρχιτεκτονικών ResNet .....	44
3.1.3 Απλοποιημένη αρχιτεκτονική .....	45
3.2 Προεπεξεργασία Δεδομένων .....	46
3.2.1 Κανονικοποίηση - Normalization .....	46
3.2.1.1 Local Mean Normalization .....	48
3.2.1.2 Global Mean Normalization .....	49
3.2.2 Ανισοτροπική Διάχυση .....	50
3.2.3 Διόρθωση Πεδίου Πόλωσης (N4) .....	54
3.2.3.1 Διόρθωση Πεδίου Πόλωσης (N3) .....	54
3.2.3.1.1 Στρατηγική Διόρθωσης .....	55
3.2.3.1.2 Εκτίμηση Πεδίου .....	57
3.2.3.1.3 Εκτίμηση της $U$ .....	57

3.2.4 Αφαίρεση εικόνων με μικρή περιεκτικότητα σε χαρακτηριστικά .....	59
<b>4. Κεφάλαιο 4<sup>ο</sup> Πειραματικό Στάδιο.....</b>	<b>60</b>
4.1 Μαγνητικός Τομογράφος.....	60
4.2 Αφαίρεση εικόνων από το σύνολο εκπαίδευσης .....	61
4.2.1 Σύνολο Επαλήθευσης.....	62
4.3 Εκπαίδευση .....	62
4.3.1 Μετρικές.....	62
4.4 Περιβάλλον Python 2.7 .....	64
4.4.1 Keras - Tensorflow.....	64
4.4.2 SimpleITK.....	64
4.4.3 Συναρτήσεις Προεπεξεργασίας Δεδομένων .....	64
4.4.4 Εισαγωγή δεδομένων στο σύστημα.....	65
4.4.5 Tensorboard .....	66
4.4.6 Μείωση βαθμού μάθησης σε περίπτωση σταθεροποίησης δικτύου ..	67
4.4.7 ModelCheckpoint .....	67
4.4.8 Συναρτήσεις υλοποίησης δικτύων.....	67
4.4.9 Εποχές, επαναλήψεις και πλήθος εικόνων ανά επανάληψη .....	68
4.4.10 Αρχικοποίηση ImageNet .....	69
4.4.11 Λίστα συχνότητας για κάθε κατηγορία.....	70
4.5 Transfer Learning .....	71
4.5.1 Fine-tuning .....	71
4.5.1.1 Cats vs Dogs .....	71
4.6 Παράμετροι Εκπαίδευσης.....	72
4.6.1 ResNet50 .....	73
4.6.2 VGGNet16 .....	73
4.6.3 Custom – Απλοποιημένη Αρχιτεκτονική .....	74
<b>5. Κεφάλαιο 5<sup>ο</sup> Αποτελέσματα και Συμπεράσματα .....</b>	<b>75</b>
5.1 Στατιστικά Αποτελέσματα .....	75
5.1.1 Επιλογή Μεθόδου Κανονικοποίησης - Εξομάλυνσης .....	75
5.1.2 Επιλογή Μεθόδου προεπεξεργασίας δεδομένων .....	78
5.1.3 Επιλογή Αλγόριθμου Μάθησης .....	80
5.1.4 Επιλογή εκπαιδευσίμων επιπέδων και αρχικοποίηση βαρών .....	83
5.1.5 Επιλογή μοντέλου .....	85
5.2 Παρουσίαση Γραφικών Εκπαίδευσης .....	87
5.2.1 Παρουσίαση Γραφημάτων Εκπαίδευσης με Ανιστροπική Διάχυση ..	88

5.2.2	Παρουσίαση Γραφημάτων Εκπαίδευσης με Ωιόρθωση Π.Πόλωσης..	89
5.2.3	Παρουσίαση Γραφημάτων Εκπαίδευσης χωρίς Προεπεξεργασία.....	90
5.2.4	Παρουσίαση Γραφημάτων Εκπαίδευσης με Συνδυασμό μεθόδων....	92
5.3	Συμπεράσματα και Σχολιασμός .....	85
5.3.1	Σχολιασμός Αποτελεσμάτων Κανονικοποίησης.....	93
5.3.2	Σχολιασμός Αποτελεσμάτων Προεπεξεργασίας .....	94
5.3.3	Σχολιασμός Επιλογής Αλγόριθμου Μάθησης .....	95
5.3.4	Σχολιασμός Επιλογής πλήθους επιπέδων - αρχικοποίησης βαρών ...	96
5.3.5	Σχολιασμός Επιλογής Αρχιτεκτονικής Σ.Ν.Δ.....	97

## Κατάλογος εικόνων

---

Εικόνα 1.1.....	6
Εικόνα 1.2.....	7
Εικόνα 2.1.....	13
Εικόνα 2.2.....	13
Εικόνα 2.3.....	15
Εικόνα 2.4.....	17
Εικόνα 2.5.....	18
Εικόνα 2.6.....	21
Εικόνα 2.7.....	22
Εικόνα 2.9.....	33
Εικόνα 3.1.....	37
Εικόνα 3.2.....	38
Εικόνα 3.3.....	38
Εικόνα 3.4.....	39
Εικόνα 3.5.....	39
Εικόνα 3.6.....	39
Εικόνα 3.7.....	41
Εικόνα 3.8.....	42
Εικόνα 3.9.....	43
Εικόνα 3.10.....	44
Εικόνα 3.11.....	44
Εικόνα 3.12.....	45
Εικόνα 3.13.....	51
Εικόνα 3.14.....	54
Εικόνα 3.15.....	56
Εικόνα 3.16.....	58
Εικόνα 4.1.....	68
Εικόνα 5.1.....	76
Εικόνα 5.2.....	77
Εικόνα 5.3.....	79
Εικόνα 5.4.....	81
Εικόνα 5.5.....	82
Εικόνα 5.6.....	83
Εικόνα 5.7.....	84



Εικόνα 5.8.....	85
Εικόνα 5.9.....	86
Εικόνα 5.11.....	88
Εικόνα 5.12.....	89
Εικόνα 5.14.....	92

## *Κατάλογος πινάκων*

---

Πίνακας 5.1.....	86
Πίνακας 5.2.....	88
Πίνακας 5.3.....	89
Πίνακας 5.4.....	





# Κεφάλαιο 1<sup>ο</sup> Εισαγωγή

---

## 1.1 Τεχνητή Νοημοσύνη

Ανάμεσα σε ένα μεγάλο αριθμό από πεδία που υπάγονται στον τομέα της πληροφορικής, αλματώδη ανάπτυξη, εξαιρετικό ενδιαφέρον και μεγάλα επιτεύγματα προς όφελος του ανθρώπου παρουσιάζει ο κλάδος της Τεχνητής Νοημοσύνης [12].

Με τον παραπάνω όρο αναφερόμαστε στο πεδίο εκείνο, το οποίο ασχολείται με τη σχεδίαση και την υλοποίηση υπολογιστικών συστημάτων που μιμούνται τα στοιχεία της ανθρώπινης συμπεριφοράς τα οποία υπονοούν έστω και στοιχειώδη ευφυΐα: μάθηση, προσαρμοστικότητα, εξαγωγή συμπερασμάτων, κατανόηση από τα συμφραζόμενα, επίλυση προβλημάτων κτλ.

Η τεχνητή νοημοσύνη, αποτελεί σημείο τομής μεταξύ πολλαπλών επιστημών όπως της πληροφορικής, της ψυχολογίας, της φιλοσοφίας, της επιστήμης των μηχανών κ.α, με στόχο τη σύνθεση ευφυούς συμπεριφοράς, με στοιχεία συλλογιστικής, μάθησης και προσαρμογής στο περιβάλλον, ενώ συνήθως εφαρμόζεται σε μηχανές ή υπολογιστές ειδικής κατασκευής.

Διαίρεται στη συμβολική τεχνητή νοημοσύνη, η οποία επιχειρεί να εξομοιώσει την ανθρώπινη νοημοσύνη *αλγοριθμικά* χρησιμοποιώντας σύμβολα και λογικούς κανόνες *υψηλού επιπέδου*, και στην υποσυμβολική τεχνητή νοημοσύνη, η οποία προσπαθεί να αναπαράγει την ανθρώπινη ευφυΐα χρησιμοποιώντας *στοιχειώδη αριθμητικά μοντέλα* που συνθέτουν επαγωγικά νοήμονες συμπεριφορές με τη διαδοχική αυτοοργάνωση απλούστερων δομικών συστατικών («συμπεριφορική τεχνητή νοημοσύνη»), προσομοιώνουν πραγματικές βιολογικές διαδικασίες όπως η εξέλιξη των ειδών και η λειτουργία του εγκεφάλου («υπολογιστική νοημοσύνη»), ή αποτελούν εφαρμογή στατιστικών μεθοδολογιών σε προβλήματα τεχνητής νοημοσύνης.

Η σύγχρονη τεχνητή νοημοσύνη αποτελεί ένα από τα πλέον «μαθηματικοποιημένα» και ταχέως εξελισσόμενα πεδία της πληροφορικής. Σήμερα, ο τομέας αξιοποιεί περισσότερο υποσυμβολικές μεθόδους και εργαλεία καταγόμενα από τα εφαρμοσμένα μαθηματικά και τις επιστήμες μηχανικών, παρά από τη θεωρητική πληροφορική και τη μαθηματική λογική όπως συνέβαινε πριν το 1990.

Σε ακαδημαϊκό επίπεδο η τεχνητή νοημοσύνη μελετάται επίσης από την ηλεκτρονική μηχανική, ενώ συνιστά ένα από τα σημαντικότερα θεμελιακά συστατικά του διεπιστημονικού γνωστικού πεδίου της γνωσιακής επιστήμης.

Κατά τη δεκαετία του 1940 εμφανίστηκε η πρώτη μαθηματική περιγραφή τεχνητού νευρωνικού δικτύου, με πολύ περιορισμένες δυνατότητες επίλυσης αριθμητικών προβλημάτων.

Καθώς ήταν εμφανές ότι οι ηλεκτρονικές υπολογιστικές συσκευές που κατασκευάστηκαν μετά τον Β' Παγκόσμιο Πόλεμο ήταν ένα τελείως διαφορετικό είδος μηχανής από ό,τι προηγήθηκε, η συζήτηση για την πιθανότητα εμφάνισης μηχανών με νόηση ήταν στην ακμή της. Το 1950 ο μαθηματικός Άλαν Τούρινγκ, πατέρας της θεωρίας υπολογισμού και προπάτορας της τεχνητής νοημοσύνης, πρότεινε τη δοκιμή Τούρινγκ, μία απλή δοκιμασία που θα μπορούσε να εξακριβώσει αν μία μηχανή διαθέτει «ευφυΐα».

Η τεχνητή νοημοσύνη θεμελιώθηκε τυπικά ως πεδίο στη συνάντηση ορισμένων επιφανών Αμερικανών επιστημόνων του τομέα το 1956 (Τζον Μακάρθι, Μάρβιν Μίνσκυ, Κλοντ Σάνον κλπ). Τη χρονιά αυτή παρουσιάστηκε για πρώτη φορά και το *Logic Theorist*, ένα πρόγραμμα το οποίο στηριζόταν σε συμπερασματικούς κανόνες τυπικής λογικής και σε ευρετικούς αλγορίθμους αναζήτησης για να αποδεικνύει μαθηματικά θεωρήματα. Στο παρόν κείμενο θα ασχοληθούμε με τον υποκλάδο της ΤΝ, ο οποίος ονομάζεται Μηχανική Μάθηση [13] και συγκεκριμένα με τη χρήση τεχνητών νευρωνικών δικτύων.

## 1.2 Μηχανική Μάθηση

Ο Arthur Samuel, το 1959, όρισε τη Μηχανική Μάθηση ως το “Πεδίο Μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί”. Είναι υποπεδίο της επιστήμης των υπολογιστών, το οποίο αναπτύχθηκε από τη μελέτη της αναγνώρισης προτύπων και της υπολογιστικής θεωρίας μάθησης στην τεχνητή νοημοσύνη.

Αντικείμενο διερεύνησης της είναι η μελέτη και η κατασκευή αλγορίθμων, οι οποίοι στη εφαρμοζονται σε μοντέλα, τα οποία μπορούν να “μαθαίνουν” από πειραματικά δεδομένα και να κάνουν προβλέψεις βασιζόμενες σε αυτά τα δεδομένα ή να εξάγουν αποφάσεις που εκφράζονται ως το αποτέλεσμα.

Ο Tom M. Mitchell πρότεινε έναν πιο επίσημο ορισμό που χρησιμοποιείται ευρέως: «Ένα πρόγραμμα υπολογιστή λέγεται ότι μαθαίνει από εμπειρία  $E$  ως προς μια κλάση εργασιών  $T$  και ένα μέτρο επίδοσης  $P$ , αν η επίδοσή του σε εργασίες της κλάσης  $T$ , όπως αποτιμάται από το μέτρο  $P$ , βελτιώνεται με την εμπειρία  $E$ ». Αυτός ο ορισμός είναι σημαντικός για τον καθορισμό της μηχανικής μάθησης σε βασικό λειτουργικό πλαίσιο παρά με γνωστικούς όρους, ακολουθώντας έτσι την πρόταση του Alan Turing στην εργασία του «Υπολογιστικές μηχανές και Νοημοσύνη», ότι το ερώτημα αν μπορούν οι μηχανές να σκεφτούν, μπορεί να αντικατασταθεί με το ερώτημα αν μπορούν οι μηχανές να κάνουν αυτό που εμείς, ως σκεπτόμενες οντότητες, μπορούμε να κάνουμε.

Η Μηχανική Μάθηση, εφαρμόζεται σε μία σειρά από υπολογιστικές εργασίες, όπου τόσο ο σχεδιασμός όσο και ο ρητός προγραμματισμός των αλγορίθμων είναι ανέφικτος.

Παραδείγματα εφαρμογών αποτελούν τα φίλτρα spam [14], οπτική αναγνώριση χαρακτήρων [15], οι μηχανές αναζήτησης [16] και η υπολογιστική όραση [17]. Στο πεδίο ανάλυσης δεδομένων, η Μηχανική Μάθηση είναι μία μέθοδος που χρησιμοποιείται για την επινόηση πολύπλοκων μοντέλων και αλγορίθμων που οδηγούν στην πρόβλεψη.

Τα αναλυτικά μοντέλα επιτρέπουν στους ερευνητές, τους επιστήμονες δεδομένων, τους μηχανικούς και τους αναλυτές να παράγουν αξιόπιστες αποφάσεις και αποτελέσματα και να αναδειξουν αλληλοσυσχετίσεις μέσω της μάθησης από ιστορικές σχέσεις και τάσεις στα δεδομένα.

Οι εργασίες Μηχανικής Μάθησης διακρίνονται συνήθως σε τρεις μεγάλες κατηγορίες, ανάλογα με τη φύση του εκπαιδευτικού «σήματος» ή την «ανατροφοδότηση» που υλοποιείται σε ένα σύστημα εκμάθησης.

- Επιτηρούμενη μάθηση [18] (αλλιώς επιβλεπόμενη μάθηση ή μάθηση με επίβλεψη) (supervised learning): Το υπολογιστικό πρόγραμμα δέχεται τις παραδειγματικές εισόδους καθώς και τα επιθυμητά αποτελέσματα από έναν «δάσκαλο», και ο στόχος είναι να μάθει έναν γενικό κανόνα προκειμένου να αντιστοιχίσει τις εισόδους με τα αποτελέσματα.
- Μη επιτηρούμενη μάθηση (αλλιώς επίβλεπτη μάθηση ή μάθηση χωρίς επίβλεψη) (unsupervised learning): Χωρίς να παρέχεται κάποια εμπειρία στον αλγόριθμο μάθησης, πρέπει να βρει την δομή των δεδομένων εισόδου. Η Μη επιβλεπόμενη μάθηση μπορεί να είναι αυτοσκοπός (ανακαλύπτοντας κρυμμένα μοτίβα σε δεδομένα) ή μέσο για ένα τέλος (χαρακτηριστικό της μάθησης).
- Ενισχυτική μάθηση: Ένα πρόγραμμα υπολογιστή αλληλεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να επιτευχθεί ένας συγκεκριμένος στόχος (όπως η οδήγηση ενός οχήματος), χωρίς κάποιος δάσκαλος να του λέει ρητά αν έχει φτάσει

κοντά στο στόχο του. Ένα άλλο παράδειγμα είναι να μάθει να παίζει ένα παιχνίδι εναντίον κάποιου αντιπάλου.

### 1.2.1 Ιστορικά Στοιχεία

Ως επιστημονικό εγχείρημα, η μηχανική μάθηση αναπτύχθηκε από την αναζήτηση για την τεχνητή νοημοσύνη. Ήδη από την πρώιμη περίοδο της έρευνας στον τομέα της τεχνητής νοημοσύνης σε ακαδημαϊκό επίπεδο, το ζήτημα της κατασκευής μηχανών που θα μάθαιναν από δεδομένα απασχόλησε τους ερευνητές.

Προσπάθησαν να προσεγγίσουν το πρόβλημα με διάφορες συμβολικές μεθόδους, καθώς και με τα λεγόμενα νευρωνικά δίκτυα. Αυτά ήταν ως επί το πλείστον perceptrons [21] και μοντέλα, που όπως διαπιστώθηκε αργότερα ήταν επανεφευρέσεις των γενικευμένων γραμμικών μοντέλων της στατιστικής. Επίσης χρησιμοποιήθηκε η πιθανοθεωρητική λογική, ιδιαίτερα στην αυτοματοποιημένη ιατρική διάγνωση.

Ωστόσο, μια αυξανόμενη έμφαση σε προσεγγίσεις που βασίζονται στην λογική γνώση προκάλεσε ένα ρήγμα μεταξύ Τεχνητής Νοημοσύνης και μηχανικής μάθησης. Τα πιθανοθεωρητικά συστήματα μαστίζονταν από θεωρητικά και πρακτικά προβλήματα απόκτησης δεδομένων και αναπαράστασής τους.

Από το 1980, έμπειρα συστήματα επικράτησαν στο πεδίο της Τεχνητής Νοημοσύνης, και ο ρόλος της στατιστικής υποχώρησε. Η εργασία σε συμβολική/βασισμένη σε γνώση εκμάθηση συνεχίστηκε εντός της ΤΝ, οδηγώντας στον επαγωγικό λογικό προγραμματισμό, αλλά οι κατευθυντήριες γραμμές της στατιστικής ήταν τώρα έξω από το χώρο της τεχνητής νοημοσύνης, στην αναγνώριση προτύπων και στην ανάκτηση πληροφοριών. Η έρευνα για νευρωνικά δίκτυα εγκαταλήφθηκε από την ΤΝ και την Επιστήμη Υπολογιστών τον ίδιο περίπου καιρό. Η ίδια επίσης κατεύθυνση ακολουθήθηκε πέρα από την ΤΝ και την πληροφορική, από ερευνητές άλλων ειδικοτήτων, συμπεριλαμβανομένων των Hopfield, Rumelhart και Hinton. Η επιτυχία ήρθε στα μέσα της δεκαετίας του 1980 με την επανεφεύρεση της μεθόδου ανάστροφης μετάδοσης (backpropagation) [22].

### 1.2.2 Τεχνητά Νευρωνικά Δίκτυα

Τα τελευταία χρόνια έχει υπάρξει μία έκρηξη ενδιαφέροντος για τα νευρωνικά δίκτυα καθώς εφαρμόζονται με μεγάλη επιτυχία σε ένα ασυνήθιστα μεγάλο φάσμα τομέων της επιστήμης και της τεχνολογίας, όπως τα χρηματοοικονομικά, η ιατρική, η επιστήμη μηχανικού, η γεωλογία, η φυσική, η ρομποτική, η επεξεργασία σήματος κτλ. Στην πραγματικότητα, τα νευρωνικά δίκτυα εισάγονται οπουδήποτε τίθεται θέμα πρόβλεψης, ταξινόμησης ή ελέγχου. Η σαρωτική αυτή επιτυχία, μπορεί να αποδοθεί σε δύο βασικά στοιχεία: την ισχύ και την ευχρηστία.

#### 1.2.2.1 Ισχύς

Τα νευρωνικά δίκτυα είναι πολύ εξελιγμένες τεχνικές μη γραμμικής μοντελοποίησης, ικανές να μοντελοποιήσουν εξαιρετικά πολύπλοκες λειτουργίες. Η γραμμική μοντελοποίηση υπήρξε

ευρέως διαδεδομένη για πολύ καιρό, δεδομένου ότι στα γραμμικά μοντέλα εφαρμόζονται πολύ γνωστές στρατηγικές βελτιστοποίησης.

Στις συνήθειες, όμως, περιπτώσεις όπου η γραμμική προσέγγιση δεν ήταν έγκυρη, τα μοντέλα αυτά αποτύγχαναν αναλόγως. Τα νευρωνικά δίκτυα βέβαια, αν και επιτρέπουν τη μη γραμμικότητα μέσω χρήσης μη γραμμικών συναρτήσεων ενεργοποίησης, μεταθέτουν με τη σειρά τους το πρόβλημα στο ζήτημα της διάστασης (του πλήθους των διαφορετικών εισόδων και εξόδων), το οποίο αποτελεί αγκάθι στις προσπάθειες μοντελοποίησης μη γραμμικών συναρτήσεων με μεγάλο αριθμό μεταβλητών.

### 1.2.2.2 Ευχρηστία

Τα νευρωνικά δίκτυα εκπαιδεύονται με παραδείγματα. Ο χρήστης συγκεντρώνει αντιπροσωπευτικά δεδομένα και στη συνέχεια, καθώς τα τροφοδοτεί συστηματικά στο δίκτυο μέσω των κατάλληλων αλγορίθμων εκπαίδευσης, το δίκτυο «αντιλαμβάνεται» αυτομάτως τη δομή των δεδομένων και η «γνώση» αυτή εκφράζεται ως κατάλληλες επιλογές συναπτικών βαρών.

Επομένως, το τελικό αποτέλεσμα της εκπαίδευσης με ένα συγκεκριμένο σύνολο παραδειγμάτων είναι ο προσδιορισμός των κατάλληλων βαρών του δικτύου.

Ο χρήστης χρειάζεται να έχει κάποιες ουσιώδεις γνώσεις σχετικά με τον τρόπο επιλογής και προετοιμασίας των δεδομένων, τον τρόπο εκλογής του κατάλληλου νευρωνικού δικτύου και τέλος, όσον αφορά το πως θα ερμηνευτούν τα αποτελέσματα.

Παρ' όλ' αυτά, το επίπεδο των γνώσεων του χρήστη που απαιτούνται για μια επιτυχημένη εφαρμογή των νευρωνικών δικτύων, είναι πολύ χαμηλότερο συγκριτικά με κάποια περίπτωση που θα χρησιμοποιούνταν ορισμένες πιο παραδοσιακές, μη γραμμικές στατιστικές μέθοδοι.

Τα νευρωνικά δίκτυα είναι εφαρμόσιμα σχεδόν σε κάθε κατάσταση στην οποία ισχύει μια σχέση μεταξύ μεταβλητών πρόβλεψης (ανεξάρτητες, είσοδοι) και προβλεπόμενες μεταβλητές (εξαρτημένες, έξοδοι), ακόμα και όταν αυτή η σχέση είναι πολύ περίπλοκη για να αποδοθεί με τους συνηθισμένους όρους της «συσχέτισης» ή των «διαφόρων ομάδων». Ενδεικτικά αντιπροσωπευτικά παραδείγματα προβλημάτων στα οποία η ανάλυση των νευρωνικών δικτύων έχει εφαρμοστεί αναφέρονται παρακάτω :

- Ιατρική διάγνωση: Ένα ευρύ φάσμα ιατρικά συσχετιζόμενων ενδείξεων, όπως ο συνδυασμός της καρδιακής συχνότητας, τα επίπεδα των διαφόρων ουσιών στο αίμα, ο ρυθμός της αναπνοής μπορούν να παρακολουθηθούν. Η εκδήλωση μιας συγκεκριμένης ιατρικής κατάστασης, γίνεται να συσχετιστεί με ένα πολύπλοκο συνδυασμό μεταβολών σε ένα υποσύνολο μεταβλητών που παρακολουθούνται. Τα νευρωνικά δίκτυα έχουν χρησιμοποιηθεί για την αναγνώριση αυτού του προτύπου πρόβλεψης, ώστε να χορηγηθεί η κατάλληλη θεραπεία.
- Χρηματιστηριακές προβλέψεις: Οι διακυμάνσεις των τιμών των μετοχών και των χρηματιστηριακών δεικτών είναι ακόμα ένα παράδειγμα ενός πολύπλοκου, πολυδιάστατου, αλλά και σε ορισμένες περιπτώσεις επαναλαμβανόμενου φαινομένου. Τα νευρωνικά δίκτυα χρησιμοποιούνται από πολλούς τεχνικούς αναλυτές, ώστε να κάνουν προβλέψεις σχετικά με τις τιμές των μετοχών, βασιζόμενοι σε ένα μεγάλο αριθμό



παραγόντων, όπως δηλαδή, τις προηγούμενες επιδόσεις άλλων αποθεμάτων και διαφόρων οικονομικών δεικτών.

- Πιστωτική ανάθεση: Μια ποικιλία από κομμάτια πληροφοριών, τα οποία είναι συνήθως γνωστά για ένα απαιτούμενο δάνειο. Για παράδειγμα, η ηλικία του αιτούντος, η εκπαίδευση, το επάγγελμα και πολλά άλλα στοιχεία που μπορεί να είναι διαθέσιμα. Μετά την εκπαίδευση ενός νευρωνικού δικτύου σε ιστορικά δεδομένα η ανάλυση μπορεί να εκτοπίσει τα πιο κατάλληλα και σχετικά χαρακτηριστικά και να τα χρησιμοποιήσει για την ταξινόμηση των αιτούντων ως χαμηλού ή υψηλού κινδύνου.

### 1.2.3 Προέλευση Τεχνητών Νευρωνικών Δικτύων

Τα νευρωνικά δίκτυα αναπτύχθηκαν μέσα από τις διεξαγωγές ερευνών της Τεχνητής Νοημοσύνης. Προσπάθειες, δηλαδή, μίμησης της ανοχής σε βλάβες και τη δυνατότητα εξόρυξης γνώσης μέσα από βιολογικά νευρωνικά συστήματα, μοντελοποιώντας τη δομή των χαμηλών επιπέδων του εγκεφάλου. Στη συνέχεια, από το κύριο μέρος της έρευνας της Τεχνητής Νοημοσύνης που έγινε το 1960-1980, προήλθαν τα Έμπειρα Συστήματα (Expert Systems) [23]. Τα συστήματα αυτά έχουν βασιστεί σε ένα μοντέλο λογικής διεργασίας υψηλού επιπέδου, το οποίο δημιουργήθηκε από τον τρόπο με τον οποίο έχει δομηθεί η συλλογιστική διεργασία των ανθρώπων σχετικά με το χειρισμό των συμβόλων. Έγινε γρήγορα εμφανές πως αυτά τα συστήματα αν και ήταν πολύ χρήσιμα σε κάποιους τομείς, απέτυχαν να συλλάβουν καίριες πτυχές της ανθρώπινης νοημοσύνης. Σύμφωνα με μια πτυχή της μελέτης, το γεγονός αυτό οφειλόταν στην αποτυχία τους να μιμηθούν τη βασική δομή του εγκεφάλου. Προκειμένου να αναπαραχθεί νοημοσύνη, κρίνεται απαραίτητη η δημιουργία συστημάτων με παρόμοια δομή.

Ο ανθρώπινος εγκέφαλος αποτελείται κατά κύριο λόγο από ένα ευρύ φάσμα νευρώνων (86.000.000.000 κατά προσέγγιση), οι οποίοι είναι μαζικά διασυνδεδεμένοι με ένα μέσο όρο από διάφορες χιλιάδες διασυνδέσεις ανά νευρώνα. Κάθε νευρώνας είναι ένα εξειδικευμένο κύτταρο το οποίο έχει τη δυνατότητα μετάδοσης ενός ηλεκτροχημικού σήματος. Ο νευρώνας έχει μια διακλαδωτική διάρθρωση εισροών, τους δενδρίτες (dendrites), ένα κυτταρικό σώμα και μια διακλαδωτική δομή εκροών (τον άξονα). Οι άξονες ενός κυττάρου συνδέονται με τους δενδρίτες ενός άλλου, μέσω μιας σύναψης. Όταν, λοιπόν, ένας άξονας ενεργοποιηθεί, πυροδοτεί ένα ηλεκτροχημικό σήμα κατά μήκος του άξονα. Ένας νευρώνας εκτελεί αυτή τη διαδικασία μόνο όταν το συνολικό σήμα το οποίο λήφθηκε από τους δενδρίτες, υπερβεί ένα συγκεκριμένο επίπεδο, δηλαδή, το σημείο ενεργοποίησης (firing threshold).

Η ισχύς ενός σήματος που λαμβάνεται από ένα νευρώνα, εξαρτάται από την αποτελεσματικότητα των συνάψεων. Κάθε σύναψη περιέχει ένα κενό με νευροδιαβιβαστές χημικών ουσιών (neurotransmitter chemicals) που είναι σε ετοιμότητα για μετάδοση ενός μηνύματος. Ο Donald Hebb, ένας από τους πιο σημαντικούς ερευνητές στα νευρολογικά συστήματα, έθεσε ως ζήτημα πως η μάθηση συνιστάται κυρίως από τη μεταβολή της ισχύος των συναπτικών συνδέσεων.

Το τεχνητό νευρωνικό δίκτυο είναι ένα δίκτυο από απλούς υπολογιστικούς κόμβους (νευρώνες, νευρώνια), διασυνδεδεμένους μεταξύ τους. Είναι εμπνευσμένο από το Κεντρικό Νευρικό Σύστημα (ΚΝΣ), το οποίο προσπαθεί να προσομοιώσει.

Οι νευρώνες είναι τα δομικά στοιχεία του δικτύου. Κάθε τέτοιος κόμβος δέχεται ένα σύνολο αριθμητικών εισόδων από διαφορετικές πηγές (είτε από άλλους νευρώνες, είτε από το περιβάλλον), επιτελεί έναν υπολογισμό με βάση αυτές τις εισόδους και παράγει μία έξοδο.

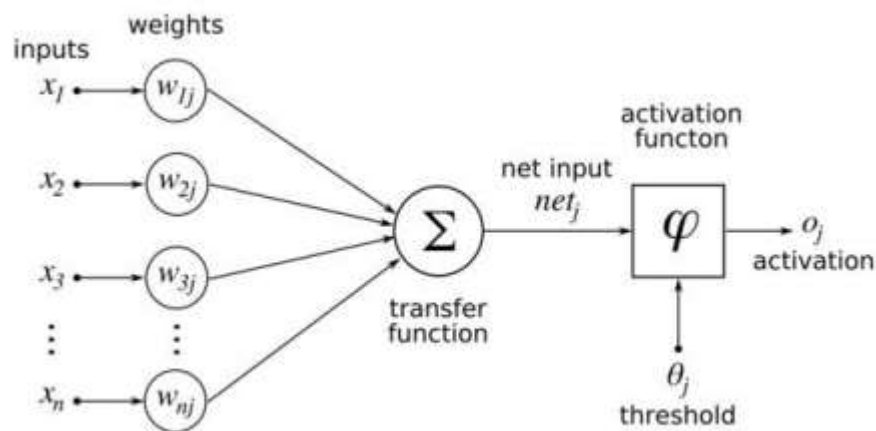
Η εν λόγω έξοδος είτε κατευθύνεται στο περιβάλλον, στη συνέχεια τροφοδοτείται ως είσοδος σε άλλους νευρώνες του δικτύου για περαιτέρω υπολογισμούς ή για τη λήψη απόφασης. Υπάρχουν τρεις τύποι νευρώνων: οι νευρώνες εισόδου, οι νευρώνες εξόδου και οι υπολογιστικοί νευρώνες ή κρυμμένοι νευρώνες.

Οι νευρώνες εισόδου δεν επιτελούν κανέναν υπολογισμό, μεσολαβούν απλώς ανάμεσα στις περιβαλλοντικές εισόδους του δικτύου και στους υπολογιστικούς νευρώνες. Οι νευρώνες εξόδου διοχετεύουν στο περιβάλλον τις τελικές αριθμητικές εξόδους του δικτύου. Οι υπολογιστικοί νευρώνες πολλαπλασιάζουν κάθε είσοδό τους με το αντίστοιχο συναπτικό βάρος και υπολογίζουν το ολικό άθροισμα των γινομένων.

Το άθροισμα αυτό τροφοδοτείται ως όρισμα στη συνάρτηση ενεργοποίησης, την οποία υλοποιεί εσωτερικά κάθε κόμβος. Η τιμή που λαμβάνει η συνάρτηση για το εν λόγω όρισμα είναι και η έξοδος του νευρώνα για τις τρέχουσες εισόδους και βάρη.

Εάν  $x_{ki}$  είναι η  $k$ -οστή είσοδος του νευρώνα,  $w_{ki}$  το  $i$ -οστό συναπτικό βάρος του  $k$ -νευρώνα και  $\phi(\cdot)$  η συνάρτηση ενεργοποίησης του νευρωνικού δικτύου, τότε η έξοδος του νευρώνα δίνεται από την εξίσωση:

$$y_k = \phi \left( \sum_{i=0}^N x_{ki} w_{ki} \right)$$



Εικόνα 1.1 : Απλουστευμένη περιγραφή της λειτουργίας τεχνητού νευρωνικού δικτύου

Στον  $k$ -οστό νευρώνα υπάρχει ένα συναπτικό βάρος με ιδιαίτερη σημασία, το οποίο καλείται **πόλωση** ή **κατώφλι** (bias, threshold). Η τιμή της εισόδου του είναι πάντα η μονάδα,  $x_{k0} = 1$ . Εάν το συνολικό άθροισμα από τις υπόλοιπες εισόδους του νευρώνα είναι μεγαλύτερο από την τιμή αυτή, τότε ο νευρώνας ενεργοποιείται. Εάν είναι μικρότερο, τότε ο νευρώνας παραμένει ανενεργός. Η ιδέα προέκυψε από τα βιολογικά νευρικά κύτταρα.

Όπως είναι φανερό, οι αριθμοί οι οποίοι συναποτελούν το διάνυσμα εισόδου (κάθε στοιχείο του διανύσματος τροφοδοτείται κατά τη λειτουργία του δικτύου σε έναν νευρώνα εισόδου), αλλά και οι αριθμοί οι οποίοι συναποτελούν το διάνυσμα εξόδου (κάθε στοιχείο του

οποίου εμφανίζεται, μετά το πέρας του ολικού υπολογισμού, σε έναν νευρώνα εξόδου), περιγράφουν χαρακτηριστικά του προς επίλυση προβλήματος.

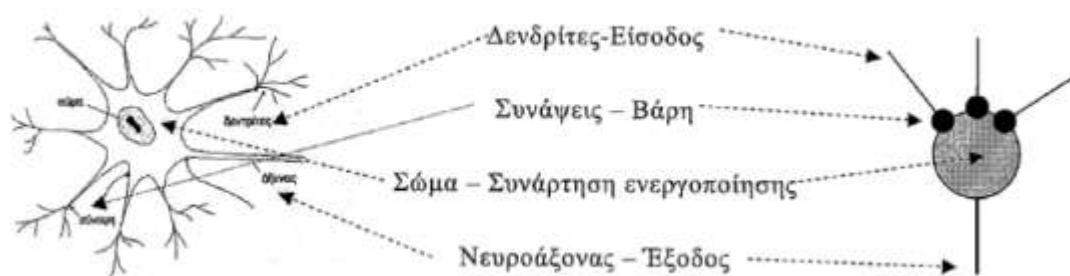
Συνήθως αυτό που μας ενδιαφέρει είναι το δίκτυο να απεικονίζει με ορθό τρόπο διανύσματα εισόδου σε κατάλληλα διανύσματα εξόδου, το πρόβλημα δηλαδή είναι η υλοποίηση μίας συνάρτησης πολλαπλών μεταβλητών, κατά κανόνα περίπλοκης και με άγνωστο ακριβή τύπο.

Τέτοιες απεικονίσεις έχουν εφαρμογή σε ποικιλία τομέων της επιστήμης και της τεχνολογίας, αφού λειτουργούν ως αριθμητικά μοντέλα για πολλά διαφορετικά ζητήματα. Το ίδιο δίκτυο μπορεί να υλοποιήσει άπειρες διαφορετικές απεικονίσεις, μία για κάθε διαφορετική επιλογή συνόλου συναπτικών βαρών.

Το κύριο χαρακτηριστικό των νευρωνικών δικτύων είναι η εγγενής ικανότητα **μάθησης**. Ως μάθηση μπορεί να οριστεί η σταδιακή βελτίωση της ικανότητας του δικτύου να επιλύει κάποιο πρόβλημα (π.χ. η σταδιακή προσέγγιση μίας συνάρτησης).

Η μάθηση επιτυγχάνεται μέσω της **εκπαίδευσης**, μίας επαναληπτικής διαδικασίας σταδιακής προσαρμογής των παραμέτρων του δικτύου (συνήθως των βαρών και της πόλωσής του) σε τιμές κατάλληλες ώστε να επιλυθεί με επαρκή επιτυχία το προς εξέταση πρόβλημα. Αφού ένα δίκτυο εκπαιδευτεί, οι παράμετροί του συνήθως «παγώνουν» στις κατάλληλες τιμές και από εκεί κι έπειτα είναι σε λειτουργική κατάσταση.

Το ζητούμενο είναι το λειτουργικό δίκτυο να χαρακτηρίζεται από μία ικανότητα **γενίκευσης**: αυτό σημαίνει πως δίνει ορθές εξόδους για εισόδους καινοφανείς και διαφορετικές από αυτές με τις οποίες εκπαιδεύτηκε.



Εικόνα 1.2 : Αναλογία μεταξύ δομικών στοιχείων φυσικού και τεχνητού νευρωνικού δικτύου

## 1.2.4 Εκπαίδευση Τεχνητών Νευρωνικών Δικτύων

Μια από τις πιο βασικές ιδιότητες των Νευρωνικών Δικτύων είναι η ικανότητά τους για εκπαίδευση. Η εκπαίδευση αυτή επιτυγχάνεται μέσω της ανταλλαγής τιμών και βαρών, που αποσκοπεί στη βαθμιαία σύλληψη της πληροφορίας η οποία στη συνέχεια θα είναι διαθέσιμη προς ανάκτηση.

Υπάρχουν, βέβαια, πολλοί αλγόριθμοι που η εφαρμογή τους έχει ως στόχο την προσαρμογή των τιμών των βαρών ενός Τεχνητού Νευρωνικού Δικτύου. Όλες οι μέθοδοι μάθησης μπορούν να καταταχθούν σε δύο κατηγορίες : τη **μάθηση με επιτήρηση** (supervised learning) και τη **μάθηση χωρίς επιτήρηση** (unsupervised learning).

#### 1.2.4.1 Μάθηση με επιτήρηση

Η μάθηση αυτή είναι μια διαδικασία η οποία συνδυάζει έναν εξωτερικό εκπαιδευτή και τη συνολική ή γενικευμένη πληροφορία. Κάποιες από τις μεθόδους οι οποίες συγκαταλέγονται σε αυτή την κατηγορία είναι η μάθηση με διόρθωση σφάλματος, ή στοχαστική μάθηση.

Παραδείγματα τα οποία αντιπροσωπεύουν την μάθηση με επίβλεψη συμπεριλαμβάνουν αποφάσεις για το πότε θα πρέπει να σταματήσει η διαδικασία εκπαίδευσης, αποφάσεις αναφορικά με τη συχνότητα παρουσίας στο δίκτυο τα πρότυπα εκπαίδευσης και η παρουσίαση προόδου του δικτύου.

Η μάθηση με επίβλεψη χωρίζεται σε δύο ακόμα κατηγορίες: στη **δομική** (structural) και στην **χρονική** (temporal) εκμάθηση. Οι αλγόριθμοι οι οποίοι βρίσκονται στην πρώτη κατηγορία, χρησιμοποιούνται για την εύρεση της βέλτιστης σχέσης μεταξύ εισόδων και εξόδων για κάθε ξεχωριστό ζευγάρι προτύπων.

Παραδείγματα της δομικής εκμάθησης αποτελούν η αναγνώριση και η κατηγοριοποίηση προτύπων, ενώ παραδείγματα της χρονικής εκμάθησης η πρόβλεψη και ο έλεγχος.

#### 1.2.4.2 Μάθηση χωρίς επιτήρηση

Οι αλγόριθμοι της εν λόγω μάθησης αναφέρονται ως αυτό-οργανώμενοι (self-organized) και είναι διαδικασίες οι οποίες δεν απαιτούν να είναι παρών ένας «εξωτερικός» δάσκαλος ή επιβλέπων.

Βασίζονται, μάλιστα, μόνο σε τοπική πληροφορία καθ' όλη τη διάρκεια της εκπαίδευσης του Τεχνητού Νευρωνικού Δικτύου. Οι συγκεκριμένοι αλγόριθμοι οργανώνουν τα δεδομένα και ανακαλύπτουν τις σημαντικές συλλογικές ιδιότητες. Για παράδειγμα, αλγόριθμοι εκπαίδευσης χωρίς επίβλεψη είναι ο αλγόριθμος Hebbian, ο διαφορικός αλγόριθμος Hebbian και ο Min-Max αλγόριθμος.

### 1.2.5 Επιτηρούμενη Μάθηση Τεχνητών Νευρωνικών Δικτύων

Πολλά πρακτικά προβλήματα είναι δυνατόν να αναπαρασταθούν με μία απεικόνιση της μορφής  $f : X \rightarrow Y$ , όπου  $X$  είναι το σύνολο των εισόδων και  $Y$  το σύνολο των εξόδων. Στην περίπτωση μας  $X$  είναι το σύνολο των εικόνων από μαγνητικούς τομογράφους, οι οποίες θα εισαχθούν στο νευρωνικό δίκτυο και  $Y$  είναι το διάστημα  $[0,1]$ .

Η έξοδος του συστήματος κάθε φορά ανάλογα με την εικόνα εισόδου είναι ένας πραγματικός αριθμός με ακραίες τιμές αυτές του προαναφερθέντος διαστήματος. Η τιμή αυτή συμβολίζει την πιθανότητα μίας εικόνας να ανήκει σε ένα από τα δύο σύνολα.

Όσο πιο κοντά στο 0 είναι η τιμή τόσο μεγαλύτερη είναι και η πιθανότητα η εικόνα να ανήκει στο σύνολο των εικόνων με υψηλού επιπέδου γλοιώμα, ενώ όσο πιο κοντά είναι στο 1, τόσο μεγαλύτερη είναι η πιθανότητα η εικόνα να ανήκει σε ασθενή με γλοιώμα χαμηλού επιπέδου.

Ωστόσο, πολλές φορές είναι αρκετά δύσκολο να ορίσουμε επακριβώς τη συνάρτηση  $f(\cdot)$ . Στο σημείο αυτό η επιβλεπόμενη μάθηση προσφέρει μία εναλλακτική προσέγγιση, με το πλεονέκτημα ότι έχουμε στην κατοχή μας ένα μεγάλο αριθμό από παραδείγματα  $(x,y)$ , (δηλαδή τις εικόνες αλλά και την εκάστοτε κατηγορία στην οποία ανήκει η κάθε εικόνα), τα οποία είναι κομμάτια της παραπάνω απεικόνισης. Μέσω των στοιχείων αυτών θα υλοποιήσουμε μία

αντιστοίχιση ανάμεσα στο σύνολο των εικόνων και των κατηγοριών, στις οποίες ανήκουν οι εικόνες αυτές.

Υποθέτουμε ένα σύνολο εκπαίδευσης με  $N$  το πλήθος ζεύγη παραδειγμάτων  $(x, y)$ , εκ των οποίων το καθένα περιέχει δύο στοιχεία. Μία εικόνα από το σύνολο  $X$  και μία κατηγορία (ετικέτα), στην οποία ανήκει η συγκεκριμένη εικόνα  $x$ , και η οποία κατηγορία, παράλληλα, ανήκει στο σύνολο  $Y$ .

Μέσω των συγκεκριμένων παραδειγμάτων μπορούμε να εκπαιδεύσουμε την απεικόνιση  $f: X \rightarrow Y$  με τον εξής τρόπο. Αναζητούμε σε μία συγκεκριμένη κλάση από συναρτήσεις  $F$ , τη συνάρτηση εκείνη  $f(\cdot)$ , η οποία «συμφωνεί» σε μεγαλύτερο βαθμό με τα δεδομένα εκπαίδευσής μας. Επιλέγοντας, λοιπόν, μία συγκεκριμένη κλάση  $F$ , από την οποία διαλέγουμε μία συνάρτηση κόστους  $f$ , υπολογίζουμε το σφάλμα  $E(y, y)$ , όπου  $y$  είναι η έξοδος του συστήματος μας, δηλαδή η κατηγορία στην οποία το συστήμα μας «αποφάσισε» να ταξινομήσει μία συγκεκριμένη εικόνα  $x$  και  $y$ , η επιθυμητή έξοδος του συστήματος, δηλαδή η κατηγορία, στην οποία ανήκει πραγματικά η εικόνα. Απότερος σκοπός είναι η εύρεση της συνάρτησης  $f(\cdot)$ , η οποία και θα ικανοποιεί την παρακάτω σχέση:

$$f = \arg (\min (E_{(x,y)}(f(x), y))), \text{ όπου } f \in F$$

Δηλαδή, αναζητούμε την  $f(\cdot)$ , η οποία θα ελαχιστοποιεί την τιμή της συνάρτησης σφάλματος  $E(\cdot)$ , ανάμεσα στην πραγματική τιμή  $y$  και την έξοδο του συστήματος,  $\hat{y}$ . Μετά την εύρεση της συνάρτησης, δε χρειαζόμαστε πλέον τα δεδομένα εκπαίδευσης, παρά μόνο την ίδια την  $f(\cdot)$ , την οποία και θα χρησιμοποιούμε για να κάνουμε την αντιστοίχιση από το σύνολο  $X$  στο σύνολο  $Y$ .

Η πιο ευρέως χρησιμοποιούμενη συνάρτηση κόστους, την οποία χρησιμοποιήσαμε και εμείς για τους σκοπούς της παρούσας διπλωματικής εργασίας είναι η cross-entropy loss (ή συνάρτηση διασχιζόμενης εντροπίας) και έχει την παρακάτω μορφή.

$$L(\hat{y}, y) = - \sum_{k=1}^K y_k \log \hat{y}_k = - \log \hat{y}_{y=1}$$

### 1.3 Επεξεργασία Εικόνας

Επεξεργασία εικόνας ονομάζεται κάθε μορφή αλγοριθμικής επεξεργασίας, ανάλυσης και χειρισμού ψηφιακών δεδομένων εικόνας ή βίντεο, όπως και το σχετικό επιστημονικό πεδίο της πληροφορικής.

Στην επεξεργασία εικόνας, τόσο η είσοδος όσο και η έξοδος των υπολογισμών είναι δεδομένα εικόνας / βίντεο (έγχρωμα, ασπρόμαυρα ή σε αποχρώσεις του γκρι). Από την επεξεργασία εικόνας εκπορεύονται επίσης και αλγόριθμοι ανάλυσης / κατανόησης εικόνας, αλλά εκεί υφίσταται επικάλυψη με το συγγενές γνωστικό πεδίο της τεχνητής νοημοσύνης ονόματι μηχανική όραση.

Μεγάλο μέρος του επιστημονικού υποβάθρου της επεξεργασίας εικόνας παρέχεται από την επεξεργασία σήματος, καθώς η ψηφιακή εικόνα μπορεί να θεωρηθεί δυσδιάστατο χωρικό σήμα και το βίντεο τρισδιάστατο χωροχρονικό σήμα.

### 1.3.1 Image Large Scale Visual Recognition Challenge

Το ILSVRC είναι ένας διαγωνισμός, ο οποίος αξιολογεί τους αλγόριθμους ανίχνευσης αντικειμένων και ταξινόμησης εικόνων σε μία πολύ ευρεία κλίμακα.

Μία από τις μεγαλύτερες δυνατότητες, τις οποίες παρέχει είναι η σύγκριση των αποτελεσμάτων στην έρευνα, ανάμεσα σε ένα μεγάλο αριθμό από ερευνητές σε όλο τον κόσμο. Συγκεκριμένα, δίνει τη δυνατότητα σύγκρισης πάνω σε αλγόριθμους εύρεσης και ταξινόμησης για μία πληθώρα διαφορετικών αντικειμένων, ενώ ταυτόχρονα αποτελεί μία μετρική όσον αφορά την εξέλιξη της υπολογιστικής όρασης (computer vision)

Κάθε χρόνο, σε ένα από τα κορυφαία συνέδρια υπολογιστικής όρασης στον κόσμο, ανακοινώνονται οι τα αποτελέσματα αυτού του διαγωνισμού. Δηλαδή, γνωστοποιούνται οι αποδοτικότερες και πιο καινοτόμες ιδέες στο χώρο της αναγνώρισης εικόνας, οι οποίες έδωσαν σε αυτούς που τις υλοποίησαν υψηλή θέση στον προηγούμενο διαγωνισμό.

Τα τελευταία χρόνια, σύμφωνα με τα αποτελέσματα του παραπάνω διαγωνισμού, οι αποδοτικότερες μέθοδοι και τεχνικές για τον εντοπισμό αντικειμένων, αλλά και για την κατηγοριοποίηση τους σε επιμέρους κατηγορίες, περιλαμβάνουν σχεδόν εξ'ολοκλήρου συνελκτικά νευρωνικά δίκτυα.

Ανάμεσα σε ένα μεγάλο αριθμό από διάφορες τεχνικές, οι οποίες μπορούν να χρησιμοποιηθούν για την κατηγοριοποίηση μίας εικόνας, τη μεγαλύτερη ακρίβεια παρουσιάζουν τα συνελκτικά νευρωνικά δίκτυα. Τα τελευταία είναι μία υποκατηγορία των κλασικών νευρωνικών δικτύων, τα οποία όμως έχουν μεγάλο βάθος στην αρχιτεκτονική τους και η δομή ανάμεσα στις συνδέσεις των επιπέδων τους διαφέρει σε μεγάλο βαθμό από αυτή των κλασικών δικτύων. Η δομή τους, οπότε, τους προσδίδει χαρακτηριστικά, με βάση τα οποία κατατάσσονται στα συστήματα με το μεγαλύτερο βαθμό ακρίβειας στο χώρο της ανάλυσης εικόνας και βίντεο. Το χαρακτηριστικό, παρ'όλα αυτά, τα οποία παρουσιάζουν τη μεγαλύτερη σημασία στην ανάλυση εικόνας είναι αυτά της αμεταβλητότητας (translation invariance) και της τοπικής συσχέτισης (spatial correlation). Η αμεταβλητότητα είναι η ικανότητα του συστήματος να αναγνωρίζει μία συγκεκριμένη ομάδα αντικειμένων και να τα εντάσσει στην ίδια κατηγορία, ακόμη και αν τα χαρακτηριστικά τους διαφέρουν σε ένα μικρό βαθμό, ενώ η τοπική συσχέτιση είναι η ιδιότητα που έχει το σύστημα, στην προσπάθειά του να κατηγοριοποιήσει ένα συγκεκριμένο πίξελ της εικόνας, να δίνει μεγαλύτερη βαρύτητα στα «κοντινότερα» ως προς αυτό πίξελ, προκειμένου να λάβει απόφαση, παρά σε εκείνα που έχουν μεγαλύτερη απόσταση από αυτό.

Στην παρούσα διπλωματική εργασία στόχος μας είναι να συγκρίνουμε υπερπαραμέτρους και αρχιτεκτονικές συνελκτικών νευρωνικών δικτύων, προκειμένου να εξετάσουμε τη σύγκλιση και την απόδοσή τους πάνω στο πρόβλημα της κατηγοριοποίησης μαγνητικών τομογραφιών που απεικονίζουν τον ανθρώπινο εγκέφαλο. Η κατηγοριοποίηση αφορά δύο επιμέρους κατηγορίες, οι οποίες αποτελούνται από εικόνες που απεικονίζουν είτε ασθενείς με υψηλού βαθμού γλοιώμα είτε ασθενείς με χαμηλού βαθμού γλοιώμα.

## Κεφάλαιο 2<sup>ο</sup> Συνελικτικά Νευρωνικά Δίκτυα (CNNs)

Τα συνελικτικά νευρωνικά δίκτυα, όπως προαναφέρθηκε, παρουσιάζουν πολλές ομοιότητες με τα κλασσικά νευρωνικά δίκτυα. Αποτελούνται από νευρώνες, οι οποίοι στην είσοδο και στην έξοδό τους συνδέονται με συνδέσεις, οι οποίες ονομάζονται βάρη και έχουν τιμές που αλλάζουν κατά τη διάρκεια της εκπαίδευσης.

Κάθε υπολογιστικός νευρώνας, δέχεται ένα πλήθος από εισόδους και υπολογίζει το εσωτερικό γινόμενο ανάμεσα στις εισόδους αυτές και τις τιμές των βαρών, ενώ προαιρετικά το αποτέλεσμα εισάγεται σε μία συνάρτηση μη-γραμμικότητας [24]. Ολόκληρο το δίκτυο, και στην περίπτωση των συνελικτικών νευρωνικών δικτύων, υλοποιεί μία διαφορίσιμη συνάρτηση, η οποία είναι ένας δείκτης της πιθανότητας να ανήκει η εκάστοτε εικόνα σε μία συγκεκριμένη κατηγορία.

Τέλος, συνεχίζει να υπάρχει η συνάρτηση σφάλματος(πχ Softmax [25]), η οποία υπολογίζει τη διαφορά ανάμεσα στην επιθυμητή έξοδο και στην έξοδο του δικτύου, με σκοπό να ενημερώσει τα βάρη και να αυξήσει την απόδοση του συστήματος. Η συνάρτηση αυτή βρίσκεται στα τελευταία επίπεδα(layers) του δικτύου, τα οποία είναι υπεύθυνα για τη λήψη απόφασης, ενώ όπως θα εξηγήσουμε και στη συνέχεια η κύρια λειτουργία των προηγούμενων επιπέδων είναι η εξαγωγή των χαρακτηριστικών εκείνων με τη μεγαλύτερη βαρύτητα στην ορθή κατηγοριοποίηση της κάθε εικόνας.

Παρόλες τις ομοιότητες στην αρχιτεκτονική τους με τα κλασσικά νευρωνικά δίκτυα, τα συνελικτικά παρουσιάζουν μεγαλύτερη απόδοση στην επεξεργασία της εικόνας, κυρίως λόγω του ότι κάνουν την παραδοχή ότι η είσοδός τους είναι αποκλειστικά εικόνες, ενώ τα κλασσικά νευρωνικά δίκτυα δεν εξειδικεύονται σε αυτή την κατηγορία.

Σαν αποτέλεσμα, λοιπόν, εισάγονται στην αρχιτεκτονική των δεύτερων, στοιχεία, τα οποία μειώνουν σε μεγάλο βαθμό της απαιτούμενες παραμέτρους του δικτύου, ενώ καθιστούν και τις λειτουργίες πιο αποτελεσματικές.

### 2.1 Σύγκριση Αρχιτεκτονικών

Όπως είδαμε και στο προηγούμενο κεφάλαιο, τα κλασσικά νευρωνικά δίκτυα δέχονται σαν είσοδο ένα διάνυσμα, το οποίο στη συνέχεια επεξεργάζονται σε μία σειρά από κρυφά επίπεδα(hidden layers). Κάθε επίπεδο αποτελείται από ένα σύνολο νευρώνων, του οποίου η έξοδος είναι συνδεδεμένη με κάθε υπολογιστικό νευρώνα του επόμενου επιπέδου και στην είσοδό του υπάρχουν πάλι συνδέσεις με κάθε νευρώνα του προηγούμενου επιπέδου.

Το σύνολο νευρώνων που ανήκουν στο ίδιο επίπεδο δεν επικοινωνούν μεταξύ του μέσω καμίας σύνδεσης, δηλαδή οι νευρώνες αυτού δεν είναι συνδεδεμένοι με βάρη. Το τελευταίο επίπεδο, ονομάζεται επίπεδο εξόδου και είναι αυτό που σε προβλήματα κατηγοριοποίησης, δίνει σαν έξοδο ένα δείκτη της πιθανότητας να ανήκει η είσοδος σε κάθε επιμέρους κατηγορία.

Το πρόβλημα με την κλασική αυτή αρχιτεκτονική είναι ότι δεν μπορεί να επεκταθεί αποτελεσματικά σε περιπτώσεις ανάλυσης εικόνας, καθώς οι συνδέσεις που απαιτεί το δίκτυο με είσοδο μία εικόνας αυξάνονται σε πολύ μεγάλο βαθμό.

Σε ένα από τα πιο γνωστά σύνολα δεδομένων, όσον αφορά το πεδίο της αναγνώρισης εικόνας, τουλάχιστον, το CIFAR-10[26], οι εικόνες είναι μόνο μεγέθους 32x32x3 (32 pixel ύψος, 32 pixel πλάτος και 3 pixel βάθος). Αυτό έχει σαν αποτέλεσμα στο πρώτο κρυμμένο επίπεδο, το οποίο συνδέεται με την είσοδο ο ένας νευρώνας να απαιτεί από μόνος του  $32*32*3 = 3072$  συνδέσεις (ή αλλιώς βάρη). Το πλήθος των συνδέσεων για εικόνες αυτού του μεγέθους, λοιπόν, μπορεί να θεωρηθεί «υποφερτό», όμως σε περίπτωση που η ίδια η εικόνα έχει μεγαλύτερες διαστάσεις, ο αριθμός των συνδέσεων αυξάνεται σε πολύ μεγάλο βαθμό, επιφέροντας αρνητικές επιπτώσεις στην αποτελεσματικότητα του δικτύου.

Συγκεκριμένα, όταν το σύστημα υλοποιείται στα πλαίσια μίας πλήρως συνδεδεμένης αρχιτεκτονικής και η εικόνα εισόδου έχει μέγεθος 200x200x3, τότε ένας μόνο νευρώνας απαιτεί  $200*200*3 = 120,000$  συνδέσεις. Αναλογιζόμενοι ότι χρειαζόμαστε «πολλούς» τέτοιους νευρώνες για να υλοποιήσουμε το σύστημά μας ο αριθμός των παραμέτρων ανεβαίνει δραματικά. Σε τέτοιες περιπτώσεις αυξάνεται κατά πολύ ο χρόνος εκπαίδευσης, λόγω των πολλών παραμέτρων που χρειάζεται να υπολογιστούν, ενώ συνήθως το σύστημα οδηγείται σε overfitting [27]. Δηλαδή, μαθαίνει να αναγνωρίζει μόνο τις εικόνες στις οποίες το εκπαιδεύσαμε, χάνοντας έτσι την ικανότητά του για γενίκευση.

Τι διαφορές παρουσιάζουν, όμως τα συνελκτικά νευρωνικά δίκτυα που τα καθιστούν πιο αποτελεσματικά στο χώρο της ανάλυσης εικόνας;

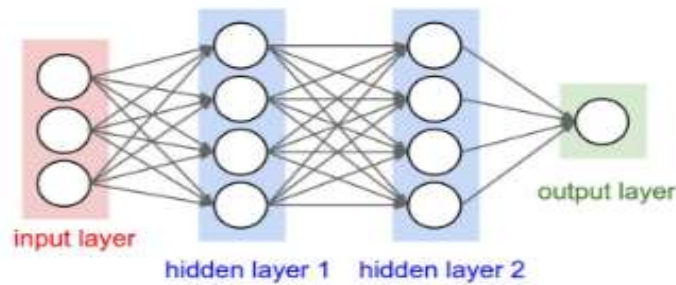
Τα αντίστοιχα συνελκτικά μοντέλα, λοιπόν, έχουν το πλεονέκτημα σε σχέση με τα κλασικά, ότι τα δεδομένα εισόδου θα ανήκουν σε κατηγορία ενός μόνο τύπου. Δηλαδή, έχουν υλοποιηθεί με τη θεώρηση ότι θα δέχονται κυρίως εικόνες, οποιασδήποτε κατηγορίας. Σαν αποτέλεσμα, στην αρχιτεκτονική τους αντανακλάται η ιδέα του να επεξεργαζόμαστε, σε συντριπτική πλειοψηφία, εικόνες με διάφορους τρόπους.

Συγκεκριμένα, σε αντίθεση με τα κλασικά νευρωνικά δίκτυα, τα επίπεδα του αντίστοιχου συνελκτικού μοντέλου είναι διατεταγμένα σε τρεις διαστάσεις: ύψος, πλάτος και βάθος. Με τον όρο βάθος δεν αναφερόμαστε στον αριθμό των κρυμμένων επιπέδων του δικτύου, αλλά στο ότι η είσοδος και η έξοδος κάθε επιπέδου, όπως θα δούμε και παρακάτω είναι ένας τρισδιάστατος πίνακας.

Για παράδειγμα, οι εικόνες εισόδου στην περίπτωση του CIFAR-10 είναι πίνακες από αριθμητικές τιμές, σε τρεις διαστάσεις: ύψος, πλάτος και βάθος. Όπως θα δούμε και στη συνέχεια ο κάθε νευρώνας ενός επιπέδου θα είναι συνδεδεμένος μόνο με μία μικρή περιοχή του επόμενου. Δηλαδή, αντικαθίσταται η πλήρης σύνδεση μεταξύ των νευρώνων διαδοχικών επιπέδων, με συνδέσεις μεταξύ συγκεκριμένων περιοχών στα διαδοχικά αυτά επίπεδα. Τέλος, στην περίπτωση του CIFAR-10, το επίπεδο εξόδου θα είχε διαστάσεις 1x1x10, διότι σκοπός του συστήματος είναι στην έξοδο του να υπολογίζει ένα διάνυσμα με τιμές που αντιστοιχούν στην πιθανότητα κάθε εικόνας να ανήκει σε καθεμία από τις δέκα διαφορετικές κατηγορίες.

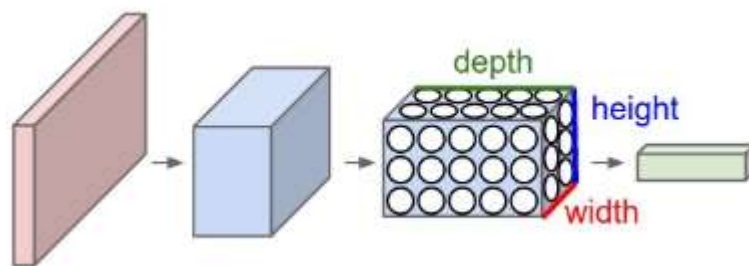


## Κλασική αρχιτεκτονική



Εικόνα 2.1 : Απλουστευμένη αρχιτεκτονική νευρωνικού δικτύου με δύο κρυφά επίπεδα

## Ροή Πληροφορίας σε συνελκτικό νευρωνικό δίκτυο



Εικόνα 2.2 : Απλουστευμένη περιγραφή της λειτουργίας συνελκτικού νευρωνικού δικτύου

Όπως φαίνεται και στην αντίστοιχη εικόνα, ένα επίπεδο είναι οργανωμένο σε τρεις διαστάσεις και επεξεργάζεται έναν τρισδιάστατο πίνακα για να δώσει σαν έξοδο πάλι έναν τρισδιάστατο πίνακα, αλλά ο δεύτερος είναι διαφορετικών διαστάσεων. Όπως προχωράμε σε επίπεδα το ύψος και το πλάτος της πληροφορίας μειώνεται όσον αφορά το μέγεθος της διάστασης, αλλά αυξάνεται το βάθος.

## 2.2 Τύποι Επιπέδων

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, τα συνελκτικά νευρωνικά δίκτυα επεξεργάζονται την πληροφορία της εισόδου σε ένα πλήθος επιπέδων, τα οποία βρίσκονται ανάμεσα στο επίπεδο εισόδου και το επίπεδο εξόδου και ονομάζονται κρυφά επίπεδα. Τα τελευταία δεν είναι όλα ίδια μεταξύ τους, αλλά υπάρχουν διαφορετικοί τύποι, εκ των οποίων το καθένα επιτελεί και διαφορετική λειτουργία.

Οι τέσσερις βασικοί τύποι επιπέδων, από τους οποίους αποτελούνται τα συνελκτικά νευρωνικά δίκτυα είναι οι εξής:

- Συνελκτικό Επίπεδο (Convolutional Layer) [28]
- Επίπεδο Pooling (Pooling Layer) [29]
- Πλήρως Συνδεδεμένο Επίπεδο (Fully-Connected layer) [30]
- Επίπεδο Μη Γραμμικότητας (Rectified Linear Unit Layer) [31]

## 2.3 Παράδειγμα Αρχιτεκτονικής

Περισσότερες λεπτομέρειες για το κάθε επίπεδο παρατείνονται στη συνέχεια του κεφαλαίου, ενώ σε αυτό το σημείο θα εξετάσουμε την αλληλουχία με την οποία συνδέονται αυτά τα επίπεδα σε ένα συνελκτικό νευρωνικό δίκτυο, μέσω ενός απλού παραδείγματος πάνω στο σύνολο δεδομένων (CIFAR-10)

Η αρχιτεκτονική μας, στη συγκεκριμένη περίπτωση αποτελείται από πέντε διαφορετικά επίπεδα

### [INPUT-CONV-RELU-POOL-FC]

Πιο συγκεκριμένα για το κάθε επίπεδο:

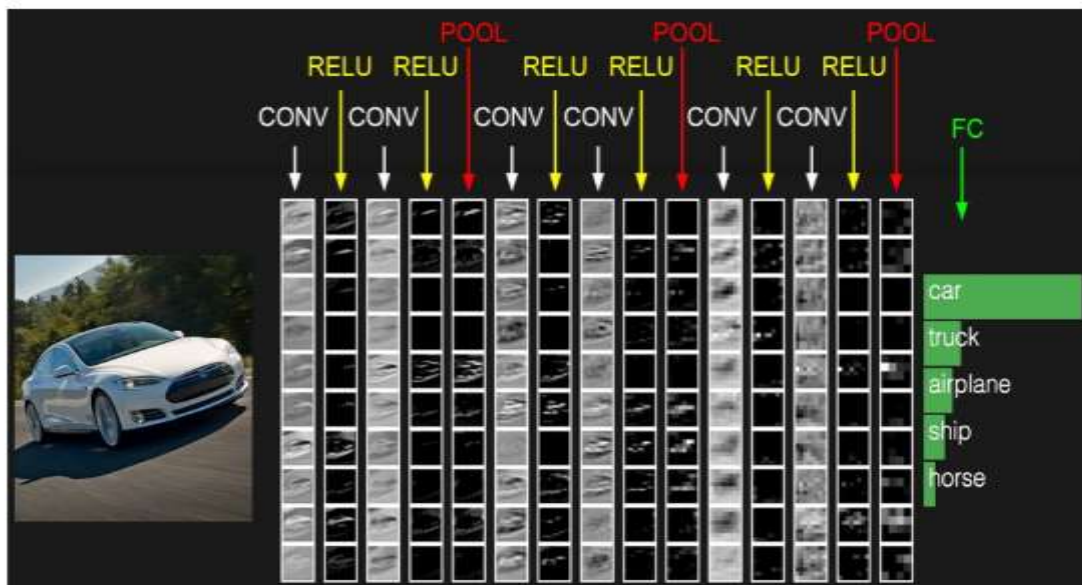
- INPUT [32x32x3] ή επίπεδο εισόδου. Σε αυτό το επίπεδο εισάγονται οι τιμές των πιξελ της εικόνας. Σε αυτή την περίπτωση έχουμε μία εικόνα με ύψος 32, πλάτος 32 και βάθος 3. Κάθε μονάδα βάθους αντιστοιχεί και σε ένα από τα τρία χρώματα κόκκινο, πράσινο και μπλε (RGB). Διαφορετικά για κάθε έγχρωμη εικόνα λέμε ότι έχουμε τρία κανάλια, ένα για κάθε χρώμα από αυτά που αναφέρθηκαν πριν.
- CONV ή συνελκτικό επίπεδο. Πραγματοποιείται ο υπολογισμός της εξόδου των νευρώνων, οι οποίοι συνδέονται ο καθένας με ένα συγκεκριμένο τμήμα της εισόδου. Ο κάθε νευρώνας είναι υπεύθυνος για την εύρεση του εσωτερικού γινομένου ανάμεσα στα βάρη με τα οποία είναι συνδεδεμένος και τις εισόδους που αντιστοιχούν στα βάρη αυτά (σε κάθε νευρώνα σαν είσοδος δεν εισάγεται ολόκληρη η εικόνα, αλλά είναι συνδεδεμένος μόνο με ένα μικρό τμήμα αυτής). Η παραπάνω διαδικασία θα έχει σαν αποτέλεσμα έναν πίνακα τριών διαστάσεων [32x32x12], στην περίπτωση που επιλέξουμε να χρησιμοποιήσουμε 12 φίλτρα.
- RELU ή επίπεδο μη γραμμικότητας. Εισάγεται μία συνάρτηση της μορφής  $\max(0, x)$ , η οποία εισάγει μη γραμμικότητα στη μορφή των δεδομένων. Η είσοδος και η έξοδος αυτού του επιπέδου έχουν ίδια μορφή [32x32x12].
- POOL. Εφαρμόζεται μία συνάρτηση, η οποία διατηρεί την ίδια πληροφορία στην είσοδο, αλλά ταυτόχρονα «μικραίνει» τη χωρητικότητά της στις διαστάσεις μήκους και ύψους. Η έξοδος έχει πλέον τη μορφή [16x16x12]
- FULLY-CONNECTED ή πλήρως συνδεδεμένο επίπεδο. Είναι το τελευταίο επίπεδο της αρχιτεκτονικής μας και σε αυτό το σημείο η είσοδος της μορφής [16x16x12], μετατρέπεται στην έξοδο της μορφής [1x1x10]. Δηλαδή, στο σημείο αυτό υπολογίζονται δέκα έξοδοι, όπου η καθεμία είναι και δείκτης της πιθανότητας μία συγκεκριμένη είσοδος να ανήκει σε κάθε κατηγορία, από αυτές που θέλουμε να «μάθει» να αναγνωρίζει το νευρωνικό μας δίκτυο. Πρακτικά από την έξοδο θέλουμε να φαίνεται σε ποιά από τις δέκα κατηγορίες του CIFAR-10 συνόλου δεδομένων ανήκει η είσοδος. Τέλος, λόγω του ότι το επίπεδο είναι πλήρως συνδεδεμένο, κάθε υπολογιστικός νευρώνας «επικοινωνεί» με καθένα από τα [16x16x12] στοιχεία της εισόδου.

Με τον παραπάνω τρόπο τα συνελκτικά νευρωνικά δίκτυα μετοχηματίζουν την αρχική είσοδο, η οποία είναι μία εικόνα, ή αλλιώς, ένας πίνακας με τις τιμές του κάθε πιξελ, σε ένα διάνυσμα, τα στοιχεία του οποίου αποτελούν έναν δείκτη της πιθανότητας να ανήκει η είσοδος σε καθεμία από τις κατηγορίες εξόδου.

Σημαντικό είναι το γεγονός ότι μερικά επίπεδα εμπεριέχουν παραμέτρους, ενώ άλλα όχι. Συγκεκριμένα, το συνελκτικό και το πλήρως συνδεδεμένο επίπεδο, εφαρμόζουν μετασχηματισμούς, οι οποίοι δεν εξαρτώνται μόνο από την είσοδο, αλλά και από τα βάρη,

τα οποία συνδέουν την είσοδο αυτή με τους νευρώνες του επιπέδου. Σε αντίθεση με τα παραπάνω επίπεδα, το επίπεδο Pooling και το επίπεδο μη γραμμικότητας εμπεριέχουν μία «απλή» συνάρτηση με πολύ μικρότερες σε αριθμό παραμέτρους, οι οποίες δεν αλλάζουν κατά την εκπαίδευση. Η εκπαίδευση του δικτύου, αποσκοπεί στην «ενημέρωση» των βαρών στα δύο πρώτα επίπεδα, δηλαδή το συνελικτικό και το πλήρως συνδεδεμένο, σύμφωνα με τον αλγόριθμο βελτιστοποίησης gradient descent, έτσι ώστε η έξοδος του συστήματος, δηλαδή η «πρόβλεψη», να πλησιάζει όσο περισσότερο γίνεται στην πραγματική έξοδο, η οποία και δείχνει την κατηγορία στην οποία ανήκει η εικόνα.

Εικόνα πιο περίπλοκης αρχιτεκτονικής που επιλύει το ίδιο πρόβλημα



Εικόνα 2.3 : Παρουσίαση δομής και διαφορετικού πλήθους επιπέδων συνελικτικού νευρωνικού δικτύου για την αναγνώριση οχημάτων.

Στη συνέχεια, και αφού έχουμε παρουσιάσει ένα παράδειγμα, το οποίο δείχνει τον τρόπο με τον οποίο ενώνονται τα διαφορετικά μεταξύ τους επίπεδα, θα αναλύσουμε τον τρόπο με τον οποίο λειτουργούν αυτά, αλλά και τι προσφέρει το καθένα στο σύστημα.

## 2.4 Συνελικτικό Επίπεδο

Το επίπεδο αυτό αποτελεί το βασικό στοιχείο στην αρχιτεκτονική οποιουδήποτε συνελικτικού νευρωνικού δικτύου, ενώ ταυτόχρονα εκτελεί και το μεγαλύτερο πλήθος υπολογισμών.

Προκειμένου, λοιπόν, να γίνει κατανοητή η λειτουργία του, θα το εξετάσουμε αποκλειστικά και στη συνέχεια θα παρουσιάσουμε την αναλογία που παρουσιάζει με τις αντίστοιχες λειτουργίες του ανθρώπινου εγκεφάλου. Οι παράμετροι του συνελικτικού επιπέδου αποτελούνται από ένα σύνολο φίλτρων, τα οποία έχουν την ικανότητα να

βελτιώνονται, μέσω της εκπαίδευσης, όσον αφορά την εξαγωγή συγκεκριμένων χαρακτηριστικών.

Το καθένα από αυτά έχει σημαντικά μικρότερες διαστάσεις από την είσοδο, όσον αφορά το ύψος και το πλάτος, αλλά επεκτείνεται σε όλο το βάθος της εισόδου. Για παράδειγμα, ένα τυπικό φίλτρο στο πρώτο επίπεδο ενός συνελκτικού νευρωνικού δικτύου, μπορεί να έχει μέγεθος 5x5x3 (5 πίξελ πλάτος, 5 πίξελ ύψος και 3 πίξελ βάθος, καθένα από τα τρία πίξελ του βάθους σχετίζεται με τα τρία διαφορετικά κανάλια χρώματος στο σύστημα RGB). Κατά τη διάρκεια της μεταφοράς των δεδομένων από την είσοδο στην έξοδο του δικτύου (forward pass), «ολισθαίνεται» κάθε φίλτρο σε όλο το ύψος και το πλάτος του πίνακα εισόδου (δηλαδή της εικόνας) και υπολογίζεται το εσωτερικό γινόμενο ανάμεσα στις τιμές του φίλτρου και το συγκεκριμένο τμήμα της εικόνας που εξετάζεται σε κάθε βήμα της ολίσθησης.

Όπως ολισθαίνεται το φίλτρο πάνω στον πίνακα εισόδου, παράγεται σαν αποτέλεσμα ένας δυσδιάστατος πίνακας, τον οποίον ονομάζουμε χάρτη εφαρμογής (activation map), από τον οποίο φαίνεται η εφαρμογή του κάθε φίλτρου πάνω στην είσοδο και συγκεκριμένα σε κάθε μικρό 3x3 (για το συγκεκριμένο φίλτρο), τμήμα της εικόνας. Μια πιο διαισθητική ερμηνεία του παραπάνω είναι ότι το σύστημα θα «μάθει» τα φίλτρα, τα οποία ενεργοποιούνται όταν αναγνωρίσει κάποιου τύπου οπτικά χαρακτηριστικά. Τέτοια είναι η κορυφή μία γωνίας ή ένα σύνολο από πίξελ ενός συγκεκριμένου χρώματος, ή και πιο περίπλοκα χαρακτηριστικά, όπως πρόσωπα ή αντικείμενα, τα οποία «μαθαίνει» να αναγνωρίζει κυρίως σε υψηλότερα (όσον αφορά τη σειρά) επίπεδα.

Στο συγκεκριμένο παράδειγμα καθένα από τα δώδεκα φίλτρα, τα οποία διαθέτει το σύστημά μας, θα εφαρμοστούν στην είσοδο και το καθένα από αυτά θα έχει σαν αποτέλεσμα ένα δυσδιάστατο χάρτη εφαρμογής. Η συνολική έξοδος του επιπέδου, για όλα τα φίλτρα μαζί, θα είναι δώδεκα τέτοιοι πίνακες διατεταγμένοι στη σειρά όσον αφορά τη διάσταση του βάθους, ή πρακτικά ένας τρισδιάστατος πίνακας με βάθος δώδεκα και σε κάθε μονάδα βάθους θα βρίσκεται το αποτέλεσμα από κάθε ξεχωριστό φίλτρο.

#### 2.4.1 Αναλογία από την πλευρά του εγκεφάλου

Κατ'αναλογία με την παραπάνω περιγραφή, κάθε στοιχείο του παραπάνω τρισδιάστατου πίνακα μπορεί να θεωρηθεί σαν έξοδος ενός εγκεφαλικού νευρώνα, ο οποίος «βλέπει» ένα μεμονωμένο κομμάτι της συνολικής εικόνας, ενώ μοιράζεται παραμέτρους με όλους τους «κοντινούς» τριγύρω, όσον αφορά τη χωρική διάταξη εγκεφαλικούς νευρώνες.

Στη συνέχεια η πληροφορία αυτών περνά σε άλλα στρώματα εγκεφαλικών νευρώνων, τα οποία εμφανίζουν μεγαλύτερη πολυπλοκότητα στη μεταξύ τους συνδεσμολογία και είναι υπεύθυνα για την αναγνώριση και κατηγοριοποίηση της οπτικής αυτής πληροφορίας. Στη συνέχεια θα εξετάσουμε τις λεπτομέρειες της συνδεσμολογίας μεταξύ των νευρώνων, τη διάταξη στο χώρο και τις παραμέτρους, τις οποίες μοιράζονται.

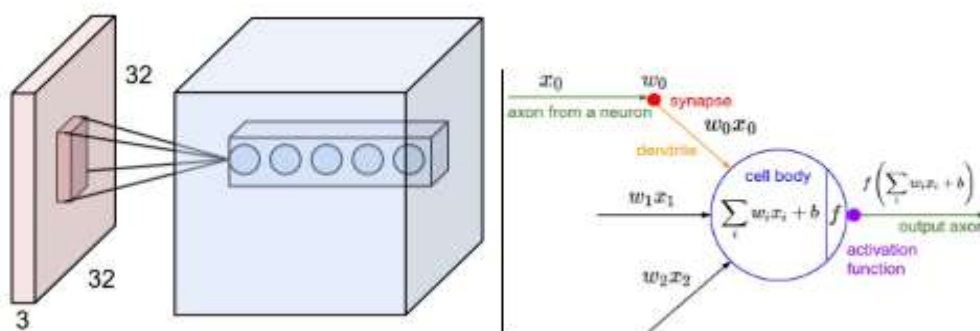
#### 2.4.2 Τοπική Συνδεσιμότητα (Local Connectivity)

[28] Στην περίπτωση που ασχολούμαστε με εισόδους υψηλών διαστάσεων, όσον αφορά το πλήθος των πίξελ, όπως αναφέρθηκε και σε προηγούμενη παράγραφο, δυσκολεύει κατά

πολύ το έργο μας να συνδέουμε κάθε τμήμα της εισόδου-εικόνας με όλους τους νευρώνες του επιπέδου που ακολουθεί.

Προκειμένου να αποφευχθεί το παραπάνω, λοιπόν, συνδέουμε κάθε νευρώνα με ένα αρκετά «μικρότερο», όσον αφορά τις διαστάσεις, τμήμα της εισόδου. Η τοπική έκταση της συνδεσιμότητας αυτής, για τον ένα μόνο νευρώνα κάθε φορά είναι μία υπερπαράμετρος, η οποία ονομάζεται πεδίο πρόσληψης (receptive field) του νευρώνα.

Ισοδύναμα η υπερπαράμετρος αυτή για το δίκτυο είναι το μέγεθος του φίλτρου. Η έκταση της συνδεσιμότητας αυτής, όσον αφορά τη διάσταση του βάθους θα είναι πάντα ίση με το βάθος της εισόδου. Τέλος, είναι σημαντικό να αναφερθεί, πως η βάση υπό την οποία εξετάζουμε την πληροφορία σχετικά με τις διαστάσεις είναι η ίδια. Η συνδεσιμότητα «περιορίζεται» όσον αφορά το πλάτος και το ύψος, ενώ ταυτόχρονα επεκτείνεται σε όλο το βάθος.



Εικόνα 2.4 : Εφαρμογή διαφορετικών φίλτρων πάνω σε ίδια τμήματα της ίδιας εικόνας. Κάθε φίλτρο επεκτείνεται σε ολόκληρο το βάθος της εικόνας και συνολικά όλοι οι δυσδιάστατοι χάρτες εφαρμογής δομούν έναν τρισδιάστατο πίνακα εξόδου.

### 2.4.3 Χωρική Οργάνωση (Spatial Arrangement)

[28] Στην προηγούμενη παράγραφο αναλύσαμε τη σύνδεση της εισόδου με το συνελκτικό επίπεδο, αλλά δεν αναφερθήκαμε καθόλου στη συνδεσιμότητα και το πλήθος των νευρώνων στην έξοδο ή πως οργανώνονται οι νευρώνες αυτοί. Οι υπερπαράμετροι, λοιπόν, οι οποίοι ελέγχουν το μέγεθος της εξόδου του συνελκτικού επιπέδου είναι τρεις: depth, stride και zero-padding.

- Depth: Το μέγεθος αυτό πρακτικά είναι το βάθος της εξόδου. Επειδή κάθε φίλτρο έχει σαν έξοδο ένα δυσδιάστατο πίνακα, η υπερπαράμετρος αυτή αντιστοιχεί στο πλήθος των διαφορετικών φίλτρων, το οποίο δίνει τελικά το βάθος του πίνακα εξόδου. Καθένα από αυτά τα φίλτρα έχει σα σκοπό να ανιχνεύει ξεχωριστά χαρακτηριστικά της εισόδου. Για παράδειγμα, εάν το πρώτο συνελκτικό επίπεδο δέχεται σαν είσοδο τις τιμές των πίξελ τις εικόνας, χωρίς να μεσολαβεί άλλο επίπεδο, τότε διαφορετικοί κάθε φορά νευρώνες όσον αφορά το βάθος μπορεί να ενεργοποιούνται στην παρουσία χαρακτηριστικών όπως προσανατολισμένες γωνίες ή όμοια χρωματισμένες επιφάνειες. Θα αναφερόμαστε σε ένα σύνολο νευρώνων που «κοιτούν» στην ίδια περιοχή της εισόδου σαν βάθος στήλης (depth column).

- **Stride:** Η υπερπαραμέτρος αυτή ορίζει τον αριθμό των πίξελ που ολισθαίνει το φίλτρο στις διαστάσεις του πλάτους και του ύψους, πριν συμβεί ο υπολογισμός των εσωτερικών γινομένων, καθώς το φίλτρο μετακινείται πάνω στην είσοδο μέχρι να εξετάσει την εικόνα στο σύνολό της. Όταν το stride λαμβάνει τιμή ένα, τότε μετακινούμε το φίλτρο ένα πίξελ κάθε φορά, δύο πίξελ για τιμή δύο κτλ. Με αυτόν τον τρόπο η έξοδος που θα παραχθεί θα έχει συνήθως μικρότερο μέγεθος «χωρικά».
- **Zero padding:** Πολλές φορές είναι αρκετά χρήσιμο να εισάγουμε μηδενικές τιμές περιμετρικά της εικόνας, παρ'ότι αυτό μπορεί να αυξήσει το μέγεθος των διαστάσεών της. Η υπερπαραμέτρος αυτή ονομάζεται zero-padding και η βασική του ιδιότητα είναι ότι μας δίνει τη δυνατότητα να ελέγξουμε το μέγεθος των διαστάσεων της εξόδου. Πιο συγκεκριμένα, το μέγεθος αυτό χρησιμοποιείται με σκοπό να διατηρηθούν ίδιες οι διαστάσεις του ύψους και του πλάτους από την είσοδο στην έξοδο.

Στη συνέχεια, με δεδομένες τις παραπάνω υπερπαραμέτρους, είναι δυνατόν να υπολογιστεί το μέγεθος του πίνακα εξόδου, σαν συνάρτηση των παρακάτω:

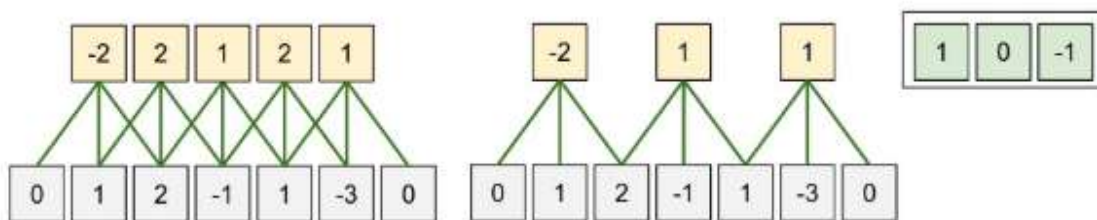
1. Το μέγεθος του πίνακα εισόδου, το οποίο θα συμβολίσουμε με **W**
2. Το μέγεθος του πεδίου πρόσληψης (receptive field) του νευρώνα στο συνελκτικό επίπεδο ή **F**
3. Το stride, το οποίο εφαρμόζεται ή **S**
4. Το zero padding, συμβολιζόμενο ως **P**

Ανάμεσα στα τέσσερα παραπάνω μεγέθη έχει διαπιστωθεί πως στην πλειονότητα των περιπτώσεων η συνθήκη, η οποία αυξάνει την απόδοση είναι η παρακάτω:

$$(W - F + 2P)/S + 1$$

Για παράδειγμα μία είσοδος, η οποία είναι της τάξης 7x7 (,όσον αφορά το ύψος και το πλάτος) και ένα φίλτρο με διαστάσεις 3x3, εάν επιλέξουμε stride 1 και zero padding 0, θα έχει σαν αποτέλεσμα μία έξοδο της τάξης 5x5. Αλλάζοντας το stride σε 2, ο πίνακας εξόδου θα ήταν της τάξης του 3x3, κοκ.

Ακολουθεί παράδειγμα, στην περίπτωση που η είσοδος μπορεί να παρασταθεί σαν πίνακας μόνο μίας διάστασης.



Εικόνα 2.5 : Εφαρμογή διαφορετικού βήματος (stride), στην περίπτωση μονοδιάστατου πίνακα πληροφορίας

Στη συγκεκριμένη περίπτωση, υπάρχει μόνο μία διάσταση όσον αφορά τη δομή της πληροφορίας, δηλαδή τον τρόπο με τον οποίο διαλέγουμε να αναπαραστήσουμε τα δεδομένα και να τα εισάγουμε στο σύστημα, ενώ ταυτόχρονα επιλέγουμε την είσοδο να είναι της τάξης  $W=5$ . Κάθε νευρώνας έχει πεδίο υποδοχής (receptive field)  $F=3$ . Στο αριστερό σχήμα, ολισθαίνοντας κατά μία θέση (stride = 1),

$S = 1$ , λαμβάνουμε έξοδο  $(5 - 3 + 2)/1 + 1 = 5$ . Στο σχήμα στα δεξιά, επιλέγουμε  $S = 2$  και αυτό έχει σαν αποτέλεσμα έξοδο μεγέθους  $(5 - 3 + 2)/2 + 1 = 3$ .

Είναι σημαντικό να σημειωθεί, ότι δε θα ήταν δυνατόν να επιλέξουμε  $S = 3$ , διότι σε αυτή την περίπτωση δεν θα ήταν δυνατόν για το φίλτρο να εξετάσει την είσοδο στην πληρότητά της. Όσον αφορά την παραπάνω σχέση, εάν είναι δυνατόν να επιλεγεί ένας αριθμός  $N$ , σαν το  $stride$ , μπορούμε να το γνωρίζουμε διότι θα πρέπει αυτός να διαιρεί ακριβώς το μέγεθος  $(W - F + 2P)$ . Στη συγκεκριμένη περίπτωση θα πρέπει να διαιρεί ακριβώς τον αριθμό  $(5 - 3 + 2) = 4$ , το οποίο δε συμβαίνει όταν  $S = 3$ .

#### 2.4.4 Χρήση του zero-padding

[32] Αναλύοντας το παράδειγμα του παραπάνω σχήματος, και συγκεκριμένα την εικόνας στα αριστερά, παρατηρούμε ότι η διάσταση της εισόδου ήταν ίση με αυτή της εξόδου ( $W = 5$ ). Αυτό συνέβη, λόγω του ότι το πεδίο προσληψης (receptive field), ήταν  $F = 3$  και χρησιμοποιήσαμε zero-padding  $P = 1$ . Στην περίπτωση που δε χρησιμοποιούσαμε καθόλου zero-padding, τότε η διάσταση της εξόδου θα ήταν 3, γιατί τόσοι νευρώνες θα ήταν μόνο δυνατόν να ενωθούν με τα στοιχεία της εισόδου, χρησιμοποιώντας  $S = 1$ .

Ο κανόνας, ο οποίος χρησιμοποιείται σε γενική περίπτωση, προκειμένου να ταυτιστούν οι διαστάσεις της εισόδου και της εξόδου είναι να χρησιμοποιείται zero-padding,  $P = (F - 1)/2$ , σε συνδυασμό με  $S = 1$ . Είναι πολύ συχνό φαινόμενο, λοιπόν, να χρησιμοποιείται το zero-padding με αυτόν τον τρόπο όταν επιλέγονται οι παράμετροι ενός συνελκτικού νευρωνικού δικτύου, όπως θα σχολιάσουμε και παρακάτω.

#### 2.4.5 Περιορισμοί στην επιλογή του Stride

Σημαντικό επίσης είναι ότι οι παραπάνω παράμετροι δεν επιλέγονται ανεξάρτητα, αλλά σε πολύ μεγάλο βαθμό η επιλογή μίας από αυτές εξαρτάται από τις υπόλοιπες. Για παράδειγμα, όταν η είσοδος έχει μέγεθος  $W = 10$ , δε χρησιμοποιείται zero-padding ( $P = 0$ ), και η τάξη του φίλτρου είναι  $F = 3$ , θα ήταν αδύνατον να επιλεγεί τιμή  $S = 2$  για το  $stride$ , καθώς η παραπάνω σχέση, δίνει:

$$(W - F + 2P)/S + 1 = (10 - 3 + 0)/2 + 1 = 4.5.$$

Το παραπάνω αποτέλεσμα δεν είναι ακέραιο. Ο συνδυασμός, λοιπόν, των παραμέτρων αυτών είναι αδύνατος, καθώς το πλήθος των νευρώνων του επόμενου επιπέδου, δεν θα αντιστοιχεί επακριβώς, όσον αφορά τις συνδέσεις με όλα τα στοιχεία της εισόδου. Το παραπάνω θα ήταν πρόβλημα στην περίπτωση που για ένα σύστημα είχαν επιλεγεί αυτές οι υπερπαραμέτροι. Προς λύση αυτού, σε μία βιβλιοθήκη που υλοποιεί αρχιτεκτονικές συνελκτικών νευρωνικών δικτύων, υλοποιούνται κομμάτια κώδικα, τα οποία αντιμετωπίζουν το παραπάνω πρόβλημα. Είτε ενημερώνουν το χρήστη για τη λανθασμένη επιλογή υπερπαραμέτρων (με warnings), είτε αφαιρούν κομμάτια της εισόδου, προκειμένου να αντιστοιχιστούν πλήρως οι συνδέσεις (crop), είτε επιλέγουν αυτόματα κατάλληλο zero-padding, προκειμένου να γίνει δυνατός, ο συνδυασμός των αντίστοιχων υπερπαραμέτρων. Η τελευταία μέθοδος, όπως προαναφέρθηκε, είναι και η πιο συνηθισμένη.

#### 2.4.6 Μοίρασμα Παραμέτρων (Parameter sharing)

[28] Το μοίρασμα παραμέτρων χρησιμοποιείται στην υλοποίηση συνελκτικών νευρωνικών δικτύων, σαν μέθοδος ελέγχου του πλήθους των παραμέτρων. Στην πραγματικότητα υλοποιούνται δίκτυα με  $55 \times 55 \times 96 = 290.400$  το πλήθος νευρώνες στο πρώτο συνελκτικό επίπεδο. Καθένας από αυτούς έχει  $11 \times 11 \times 3 = 363$  βάρη (,δηλαδή συνδέσεις) και ένα επιπρόσθετο μέγεθος, το οποίο σχετίζεται με την τιμή ενεργοποίησης του κάθε νευρώνα, δηλαδή λειτουργεί σαν κατώφλι και ονομάζεται bias. Συνολικά, οπότε, από το σύνολο των νευρώνων λαμβάνουμε  $290.400 \times (363 + 1) = 105.705.600$  παραμέτρους στο πρώτο επίπεδο του συνελκτικού νευρωνικού δικτύου. Είναι εμφανές ότι αυτός ο αριθμός έχει αρκετά μεγάλη τιμή.

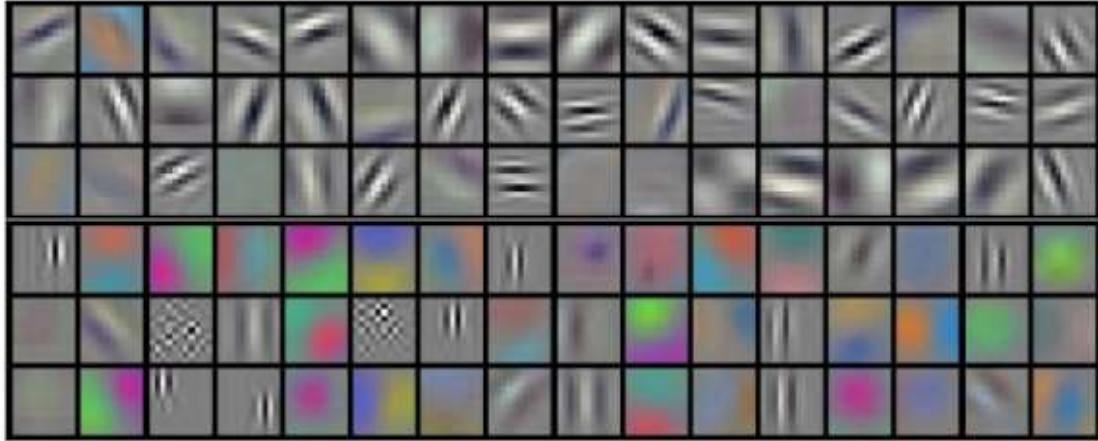
Παρ' όλ' αυτά έχουμε τη δυνατότητα να μειώσουμε δραματικά τον παραπάνω αριθμό, κάνοντας την εξής απλή υπόθεση. Εάν ένα χαρακτηριστικό είναι χρήσιμο να υπολογιστεί σε μία συγκεκριμένη θέση  $(x, y)$ , τότε θα ήταν σίγουρα χρήσιμο να υπολογιστεί και σε μία διαφορετική θέση  $(x_2, y_2)$ . Πιο συγκεκριμένα, αναλογιζόμενοι το παραπάνω παράδειγμα, σύμφωνα με το οποίο στο πρώτο επίπεδο υπήρχαν  $55 \times 55 \times 96$  το πλήθος νευρώνες, ορίζουμε σαν κομμάτι βάθους (depth slice), έναν πίνακα διαστάσεων  $55 \times 55$ , από τους 96 δυσδιάστατους πίνακες που περιέχουν τους νευρώνες.

Σε κάθε ένα από τα παραπάνω 96 κομμάτια βάθους (depth slices), θα περιορίζουμε τους νευρώνες να χρησιμοποιούν το ίδιο σύνολο από βάρη και το ίδιο bias. Σύμφωνα με αυτή τη μέθοδο, το πρώτο συνελκτικό επίπεδο στο παράδειγμά μας, θα περιλάμβανε μόνο 96 διαφορετικά σύνολα από βάρη, δηλαδή συνολικά  $96 \times 11 \times 11 = 34.843$  ξεχωριστές συνδέσεις, ή 34.944 παραμέτρους συνολικά εάν προσθέταμε και τα biases ( Υποθέτουμε ότι κάθε νευρώνας είναι συνδεδεμένος με μία περιοχή  $11 \times 11 \times 3$  του πίνακα εισόδου). Με αυτόν τον τρόπο, όλοι οι  $55 \times 55$  νευρώνες από κάθε κομμάτι βάθους, θα χρησιμοποιούν τις ίδιες παραμέτρους. Πρακτικά, κατά τη διάρκεια της εκπαίδευσης και συγκεκριμένα του αλγόριθμου backpropagation, ο οποίος θα σχολιαστεί και παρακάτω, κάθε νευρώνας υπολογίζει την παράγωγο των βαρών του, αλλά αυτές οι παράγωγοι, από όλα τα κομμάτια βάθους, θα προστεθούν και στη συνέχεια θα ενημερώσουν (update) ένα μοναδικό σύνολο από βάρη ανά κάθε συγκεκριμένο κομμάτι.

Αξίζει να σημειωθεί, ότι εάν για το σύνολο των νευρώνων σε ένα κομμάτι βάθους (depth slice), χρησιμοποιείται το ίδιο διάνυσμα βαρών (συνδέσεων), τότε όπως η πληροφορία μεταφέρεται από την είσοδο στην έξοδο του συνελκτικού επιπέδου, ο υπολογισμός των παραμέτρων σε κάθε κομμάτι βάθους είναι πρακτικά η συνέλιξη των βαρών των διαφορετικών συνόλων από νευρώνες με την είσοδο.

Από το συγκεκριμένο προκύπτει και η ονομασία συνελκτικά νευρωνικά δίκτυα. Γ' αυτό το λόγο είναι αρκετά συχνό να αναφερόμαστε στο σύνολο των βαρών ενός κομματιού βάθους σαν φίλτρο (filter), ή σαν πυρήνα (kernel) που συνελίσεται με την είσοδο.





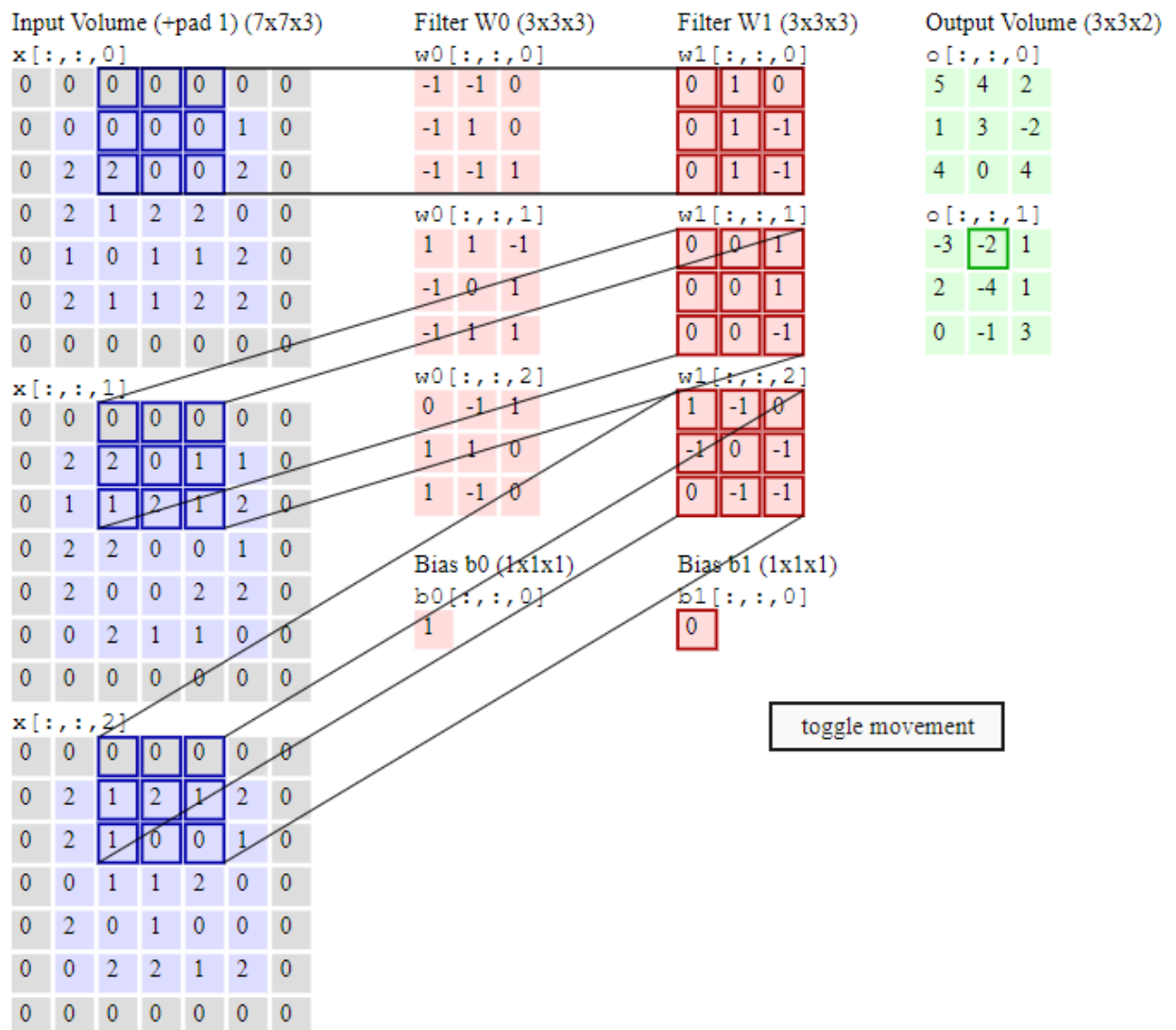
Εικόνα 2.6 : 96 σε πλήθος διαφορετικά φίλτρα μεγέθους 11x11x3, προς αναγνώριση διαφορετικών χαρακτηριστικών

Στην παραπάνω εικόνα παρουσιάζονται παραδείγματα από 96 διαφορετικά φίλτρα μεγέθους 11x11x3, καθένα από τα οποία αντιστοιχεί σε 55x55 διαφορετικούς νευρώνες του ίδιου κομματιού βάθους.

Το χαρακτηριστικό του μοιράσματος παραμέτρων είναι μία ασφαλής επιλογή για την αύξηση της απόδοσης του δικτύου. Συγκεκριμένα, στην περίπτωση που είναι σημαντικό να ανιχνευθεί μία οριζόντια γραμμή σε μία υποπεριοχή της εικόνας, είναι ασφαλές να υποθέσουμε ότι το συγκεκριμένο χαρακτηριστικό θα είναι χρήσιμο να εντοπιστεί και σε μία διαφορετική περιοχή της ίδιας εικόνας. Το προηγούμενο στηρίζεται από την ιδιότητα των εικόνων να περιγράφουν την ίδια έννοια, ακόμα και αν η θέση του αντικειμένου ή ακόμα και η δομή του διαφέρουν ελαφρώς από εικόνα σε εικόνα. Η ιδιότητα αυτή όπως προαναφέραμε ονομάζεται translation invariance. Γι αυτό το λόγο μπορούμε να «αποφύγουμε» το να μάθουμε σε καθένα από τους 55\*55 διαφορετικούς νευρώνες κάθε κομματιού βάθους να αναγνωρίζει το ίδιο χαρακτηριστικό ξεχωριστά, όπως εδώ την έννοια της οριζόντιας γραμμής.

Παρ' όλ' αυτά, η ιδέα του μοιράσματος παραμέτρων (parameter sharing), δεν κρίνεται σε όλες τις περιπτώσεις χρήσιμη. Αυτό ισχύει ακόμα περισσότερο στην περίπτωση που οι εικόνες εισόδου σε ένα συνελκτικό νευρωνικό δίκτυο, έχουν πολύ συγκεκριμένη δομή. Στην περίπτωση δηλαδή που πολύ διαφορετικά χαρακτηριστικά θα πρέπει να αναγνωριστούν για παράδειγμα στο πάνω αριστερά κομμάτι της εικόνας σε σχέση με το κάτω δεξιά. Πιο συγκεκριμένα, σε αυτό το σύνολο περιπτώσεων ανήκει και το παράδειγμα της αναγνώρισης προσώπων, τα οποία όμως είναι «κεντραρισμένα».

Δηλαδή οι εικόνες αποτελούνται από πρόσωπα, στα οποία η γωνία του κεφαλιού σε σχέση με τον κατακόρυφο άξονα είναι μηδενική. Σαν αποτέλεσμα είναι ασφαλές να συμπεράνουμε, ότι διάφορα χαρακτηριστικά για τα μάτια ή για τα μαλλιά θα βρίσκονται σε διαφορετικά χωρικά κομμάτια της εικόνας. Τα μαλλιά θα βρίσκονται πάντα στα σχετικά πιο πάνω πίξελ, ενώ τα μάτια θα βρίσκονται πάντα στο κέντρο. Για να αντιμετωπίσουμε το πρόβλημα αυτό, «χαλαρώνουμε» το βαθμό μοιράσματος παραμέτρων και απλώς αποκαλούμε το συγκεκριμένο επίπεδο τοπικά συνδεδεμένο (locally connected).



Εικόνα 2.7 : Παρουσίαση του τρόπου λειτουργίας δύο διαφορετικών συνελκτικών φίλτρων πάνω σε τρία διαφορετικά «μήματα βάθους» του πίνακα εισόδου.

Στο παραπάνω παράδειγμα οι διαστάσεις του πίνακα εισόδου (μπλε) είναι 5x5, χρησιμοποιείται zero-padding, με τιμή 1 (γι αυτό βλέπουμε ότι οι τιμές στα όρια του πίνακα είναι μηδενικές), ενώ τα φίλτρα (κόκκινοι πίνακες) έχουν διαστάσεις 3x3. Αυτό έχει σαν αποτέλεσμα η έξοδος (πράσινο) να είναι ένας πίνακας 3x3.

#### 2.4.7 Αλγόριθμος Ανάστροφης Διάδοσης (Backpropagation)

Κατά τη διάρκεια του αλγόριθμου αυτού, δηλαδή όταν υπολογίζονται οι παράγωγοι, προκειμένου να ενημερωθούν τα βάρη με καινούριες τιμές, η διάδοση της πληροφορίας γίνεται από τα μπροστά στα πίσω επίπεδα, ενώ ο υπολογισμός συμβαίνει πάλι μέσω της πράξης της συνέλιξης. Η διαφορά είναι, όμως, ότι σε αυτήν την περίπτωση, τα φίλτρα έχουν

περιστραφεί χωρικά. Το παραπάνω είναι μία σχετικά σαφής και μη χρονοβόρα διαδικασία στην περίπτωση της μίας διάστασης. Όσο ανεβαίνει η πολυπλοκότητα των επιπέδων, τόσο αυξάνεται και η δυσκολία στην εφαρμογή της μεθόδου.

#### 2.4.8 1x1 Συνέλιξη

[34] Στην περίπτωση των συνελικτικών νευρωνικών δικτύων, αρκετά συχνά χρησιμοποιείται η μέθοδος της 1x1 συνέλιξης. Πιο συγκεκριμένα αυτό συμβαίνει, διότι χειριζόμαστε δεδομένα, τα οποία αποκτούν τρισδιάστατη δομή στο εσωτερικό του δικτύου και τα φίλτρα που χρησιμοποιούνται πάντα επεκτείνονται σε όλο το βάθος της εισόδου.

Για παράδειγμα, εάν η μορφή της εισόδου είναι 32x32x3, όταν εκτελείται συνέλιξη μορφής 1x1, τότε, πρακτικά, γίνονται εσωτερικά γινόμενα τριών διαστάσεων, επειδή υπάρχουν τρία κανάλια στην είσοδο (η διάσταση του βάθους είναι τρία).

#### 2.4.9 Διασταλμένη συνέλιξη (dilated convolution)

[35] Μία πρόσφατα αναπτυσσόμενη μέθοδος είναι να εισάγουμε άλλη μία υπερπαράμετρο στο συνελικτικό επίπεδο, την οποία και θα ονομάσουμε διαστολή (dilation).

Μέχρι στιγμής εξετάστηκαν φίλτρα, τα οποία ήταν συναφή, δηλαδή δεν υπήρχε κενό ανάμεσα στα στοιχεία του φίλτρου. Παρ' όλ' αυτά είναι πιθανό να έχουμε φίλτρα, τα οποία έχουν κενά μεταξύ των κελιών(στοιχείων) τους. Το παραπάνω φαινόμενο ονομάζεται διαστολή (dilation). Για παράδειγμα ένα συναφές φίλτρο θα υπολόγιζε με είσοδο  $x$ , την τιμή:  $w[0]*x[0] + w[1]*x[1] + w[2]*x[2]$ . Σε αυτή την περίπτωση, όπου το φίλτρο είναι συναφές, η διαστολή είναι 0. Για διαστολή ίση με 1, ο υπολογισμός λαμβάνει την εξής τιμή:  $w[0]*x[0] + w[1]*x[2] + w[2]*x[4]$ . Πιο απλά υπάρχει ένα κενό μίας τιμής, όσον αφορά την εφαρμογή των στοιχείων του φίλτρου. Τέτοιου τύπου φίλτρα είναι πολύ χρήσιμα, όταν χρησιμοποιούνται σε συνδυασμό με συναφή φίλτρα, διότι επιτρέπουν τη συγχώνευση χωρικών χαρακτηριστικών πάνω στην εικόνα σε πιο έντονο βαθμό και κάνοντας χρήση λιγότερων επιπέδων.

Για παράδειγμα, εάν συνδέσουμε στη σειρά δύο συνελικτικά επίπεδα, το καθένα από τα οποία έχει φίλτρα της τάξης του 3x3, τότε οι νευρώνες στο δεύτερο επίπεδο είναι συνάρτηση ενός 5x5 κομματιού της εισόδου. Δηλαδή, θα λέγαμε οτι πρακτικά το πεδίο πρόσληψης του νευρώνα είναι 5x5. Στην περίπτωση της διασταλμένης συνέλιξης, το πεδίο πρόσληψης θα αυξανόταν πιο γρήγορα.

## 2.5 Επίπεδο Pooling

Πολύ συχνά, ανάμεσα σε δύο συνελικτικά επίπεδα, επιλέγουμε να τοποθετήσουμε επίπεδα Pooling, όταν υλοποιούμε μία αρχιτεκτονική συνελικτικού νευρωνικού δικτύου.

Η λειτουργία του επιπέδου αυτού είναι να μειώνει από την είσοδο στην έξοδό του το μέγεθος της πληροφορίας. Αυτό έχει σαν αποτέλεσμα να μειώνεται ανάλογα, ο αριθμός των παραμέτρων και υπολογισμών στο δίκτυο, οπότε, πρακτικά, να αποτελεί μέθοδο αντιμετώπισης του overfitting (Όταν το σύστημα έχει γίνει πολύ καλό στο να αναγνωρίζει

μόνο τα δεδομένα εκπαίδευσης και χάνει τη δυνατότητά του για γενίκευση). Το επίπεδο αυτό, ακόμη, δρα ανεξάρτητα σε κάθε κομμάτι βάθους της εισόδου και αλλάζει τις διαστάσεις του στο χώρο, χρησιμοποιώντας τη συνάρτηση  $\max(x,0)$ .

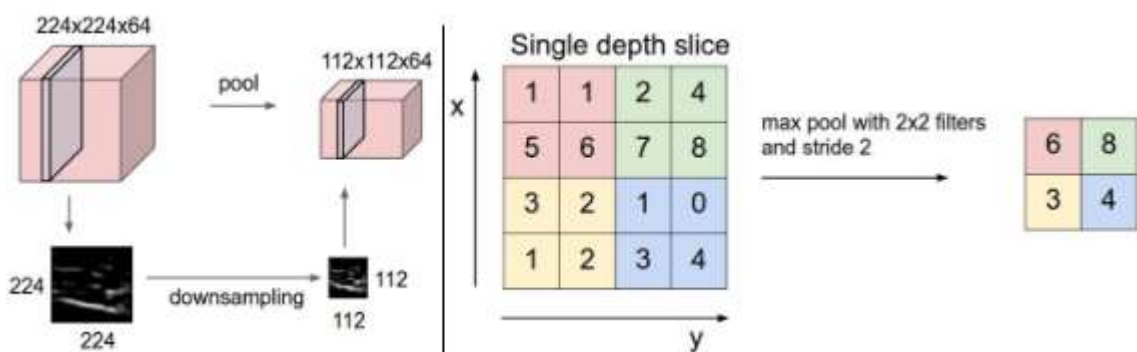
Η πιο συχνή μορφή, στην οποία συναντάμε αυτό το επίπεδο είναι με φίλτρα  $2 \times 2$ , βήμα (stride) 2 σε κάθε κομμάτι βάθους, για τις διαστάσεις του ύψους και του πλάτους. Με τη συγκεκριμένη επιλογή των υπερπαραμέτρων, από τις τέσσερις τιμές που αντιστοιχούν σε κάθε «βήμα» του φίλτρου, επιλέγεται κάθε φορά η μέγιστη.

Οπότε, η έξοδος του συγκεκριμένου επιπέδου είναι ένας πίνακας, ο οποίος έχει «κρατήσει» μόνο το 25% της πληροφορίας. Το υπόλοιπο 75% έχουμε επιλέξει να μην περάσει στην έξοδο και σαν αποτέλεσμα χάνεται. Η διάσταση του βάθους παραμένει ανεπηρέαστη, καθ'ότι όλοι οι μετασχηματισμοί αφορούν τις διαστάσεις του ύψους και του πλάτους.

Πιο συγκεκριμένα, το παραπάνω επίπεδο:

- Δέχεται έναν πίνακα στη είσοδό του, της μορφής  $W_1 \times H_1 \times D_1$ .
- Απαιτεί δύο υπερπαραμέτρους :
  - τη «χωρική» έκταση του φίλτρου  $F$ , δηλαδή, ανά βήμα, το πλήθος των τιμών στη διάσταση του ύψους και του πλάτους που θα εξετάζει.
  - Το stride  $S$ , δηλαδή το πλήθος των πίξελ που θα προχωράει το φίλτρο στις διαστάσεις του ύψους και του πλάτους ανά βήμα.
- Παράγει έναν πίνακα εξόδου με διαστάσεις  $W_2 \times H_2 \times D_2$ , όπου:
  - $W_2 = (W_1 - F) / S + 1$
  - $H_2 = (H_1 - F) / S + 1$
  - $D_2 = D_1$
- Δεν εισάγει καμία νέα παράμετρο στο σύστημα, καθώς εκτελεί μία συγκεκριμένη συνάρτηση (πχ  $\max(0,x)$ ).
- $P=0$ , διότι δεν είναι συνηθισμένη η χρήση zero-padding στην περίπτωση των επιπέδων Pooling.

Αξίζει τελικά να σημειωθεί, ότι στην πράξη χρησιμοποιούνται μόνο δύο διαφοροποιήσεις του παραπάνω επιπέδου. Η πρώτη είναι πιο σπάνια και υλοποιείται με επιλογή παραμέτρων  $F=3$  και  $S=2$ , ενώ στην πιο διαδεδομένη επιλέγουμε παραμέτρους  $F=2$  και  $S=2$ . Οι περιπτώσεις, όπου επιλέγονται υψηλότερης τάξης παράμετροι για το σύστημα δεν είναι αποτελεσματικές, διότι χάνεται ένα πολύ μεγαλύτερο κομμάτι πληροφορίας από την είσοδο στην έξοδο (όπως είδαμε και πριν, στην απλή περίπτωση όπου  $S=2$  και  $F=2$ , «χάνεται» το 75% της πληροφορίας).



Εικόνα 2.8 : Παρουσίαση της λειτουργίας του επιπέδου max-pool με παραδείγματα την περίπτωση μίας εικόνας εισόδου και ενός πίνακα εισόδου.

Στο παραπάνω παράδειγμα, το επίπεδο Pooling, μειώνει τις διαστάσεις του πίνακα της πληροφορίας από την είσοδο στην έξοδο. Στα αριστερά, η είσοδος έχει τη μορφή 224x224x64 και ύστερα από την επεξεργασία της με τιμές παραμέτρων  $F=2$ ,  $S=2$  μετατρέπεται στην έξοδο της μορφής 112x112x64.

Η πιο διαδεδομένη συνάρτηση, την οποία αξιοποιεί το επίπεδο αυτό είναι η συνάρτηση μεγίστου, και το αντίστοιχο επίπεδο ονομάζεται max pooling. Συνήθως χρησιμοποιείται stride 2, το οποίο έχει σαν αποτέλεσμα να λαμβάνεται μία τιμή ανά τέσσερις του πίνακα εισόδου, οπότε να «περνάει» στο επόμενο στάδιο το ένα τέταρτο της πληροφορίας.

### 2.5.1 Αλγόριθμος ανάστροφης διάδοσης (Backpropagation)

Κατά τη διάρκεια εκτέλεσης του παραπάνω αλγορίθμου, όταν η ροή της πληροφορίας είναι από την έξοδο στην είσοδο, δηλαδή έχει ανάποδη φορά, όσον αφορά τη συνάρτηση  $\max(x,y)$ , επιλέγεται να καρτηθεί σαν τιμή, ανά «τετράδα», μόνο η είσοδος, η οποία μας έδωσε τη μεγαλύτερη τιμή, κατά τη διάρκεια της εμπρόσθιας διάδοσης της πληροφορίας. Οπότε, στην εμπρόσθια διάδοση, συνηθίζεται να αποθηκεύεται ένας δείκτης στην είσοδο εκείνη, η οποία είχε την υψηλότερη τιμή (συνήθως αυτή ονομάζεται switch). Με αυτόν τον τρόπο επιτυγχάνεται αποτελεσματικά η υλοποίηση του παραπάνω αλγορίθμου.

### 2.5.2 Αποφυγή χρήσης επιπέδου Pooling

Σε αρκετές περιπτώσεις υλοποίησης αρχιτεκτονικών που αφορούν συνελκτικά νευρωνικά δίκτυα, προτιμάται η αφαίρεση των επιπέδων pooling και η εισαγωγή συνελκτικών επιπέδων στη θέση των πρώτων.

Μιά τέτοια αρχιτεκτονική, προκειμένου να είναι αποτελεσματική από πλευράς πλήθους παραμέτρων, χρησιμοποιεί υψηλότερης τιμής stride, εξισορροπώντας με αυτόν τον τρόπο την αφαίρεση του επιπέδου Pooling. Επίσης, σε περιπτώσεις αφαίρεσης του τελευταίου, τα αντίστοιχα μοντέλα παρουσιάζουν υψηλότερη ικανότητα γενίκευσης, όπως variational autoencodes (VAEs) [36] ή generative adversarial networks (GANs) [37].

Στο μέλλον είναι αρκετά πιθανό από πλευράς αρχιτεκτονικής, να παραλείπονται τέτοια επίπεδα συνολικά στο δίκτυο ή να χρησιμοποιούνται σπάνια.

## 2.6 Πλήρως Συνδεδεμένα Επίπεδα

Σε αυτού του τύπου τα επίπεδα, υπάρχουν πλήρεις συνδέσεις από κάθε έξοδο του προηγούμενου επιπέδου, σε νευρώνα αυτού του επιπέδου. Οι υπολογισμοί έχουν τη μορφή ενός πολλαπλασιασμού πινάκων, που ακολουθείται από μία πρόσθεση μίας τιμής ενεργοποίησης (bias). Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, η λειτουργία των επιπέδων αυτών μπορεί να γίνει εύκολα αντιληπτή, με την αντίστοιχη αναλογία από πλευράς ανθρώπινου εγκεφάλου.

Συγκεκριμένα, τα στοιχεία-νευρώνες των επιπέδων αυτών, μοιάζουν με τους εγκεφαλικούς νευρώνες, οι οποίοι αναλύουν τις οπτικές πληροφορίες που δέχονται από το

ανθρώπινο μάτι. Οι τελευταίοι είναι υπεύθυνοι για το συσχετισμό της οπτικής αυτής πληροφορίας με την αντίστοιχη εννοιολογική κατηγορία στην οποία ανήκει η πληροφορία αυτή.

Με παρόμοιο, λοιπόν, τρόπο τα πλήρως συνδεδεμένα επίπεδα λαμβάνουν υψηλού επιπέδου χαρακτηριστικά, τα οποία έχουν αποκωδικοποιηθεί από τα συνελικτικά επίπεδα, με σκοπό να αλληλοσυσχετισθεί η οπτική πληροφορία με την εκάστοτε έννοια.

### 2.6.1 Μετατροπή πλήρως συνδεδεμένων επιπέδων σε συνελικτικά

Είναι αξιοσημείωτο, πως μεταξύ του συνελικτικού και του πλήρως συνδεδεμένου επιπέδου, υπάρχουν μόνο δύο βασικές διαφορές. Στην περίπτωση του πρώτου, ένα μικρό μόνο τμήμα της εισόδου συνδέεται με κάθε νευρώνα του επιπέδου, ενώ στο δεύτερο κάθε στοιχείο της εισόδου οδηγείται μέσω αντίστοιχων συνδέσεων σε κάθε νευρώνα του επιπέδου.

Ακόμα, στο συνελικτικό επίπεδο, οι νευρώνες μοιράζονται παραμέτρους, ενώ στο πλήρως συνδεδεμένο όχι. Παρ' όλ' αυτά, οι νευρώνες και στους δύο τύπους επιπέδων, υπολογίζουν εσωτερικά γινόμενα, με αποτέλεσμα η λειτουργική τους μορφή να είναι πανομοιότυπη. Οπότε, οδηγούμαστε στο συμπέρασμα ότι είναι δυνατή η εναλλαγή ανάμεσα σε αυτούς τους δύο τύπους επιπέδων [38].

- Για κάθε συνελικτικό επίπεδο, υπάρχει ένα πλήρως συνδεδεμένο επίπεδο, το οποίο υλοποιεί την ίδια «συνάρτηση». Με άλλα λόγια, μετασχηματίζει τις ίδιες εισόδους, σε ίδιες εξόδους. Ο πίνακας(μήτρα) βαρών, θα είχε μεγάλο πλήθος στοιχείων, τα οποία θα ήταν στην πλειονότητά τους μηδενικά, με εξαίρεση συγκεκριμένα κομμάτια εισόδων που θα ήταν τα μόνα συνδεδεμένα σε κάθε νευρώνα του συνελικτικού επιπέδου. Το παραπάνω ισχύει, λόγω της τοπικής συνδεσιμότητας, ενώ παράλληλα οι παράμετροι σε πολλά από τα κομμάτια εισόδου, θα ήταν ίσα, σαν αποτέλεσμα του μοιράσματος παραμέτρων.
- Αντίστροφα, κάθε πλήρως συνδεδεμένο επίπεδο, μπορεί να μετατραπεί σε συνελικτικό. Για παράδειγμα, ένα πλήρως συνδεδεμένο επίπεδο με πλήθος νευρώνων  $K=4096$ , το οποίο δέχεται σαν είσοδο έναν πίνακα διαστάσεων  $7 \times 7 \times 512$ , μπορεί να εκφραστεί ισοδύναμα σαν ένα συνελικτικό επίπεδο με  $F=7$ ,  $P=0$ ,  $S=1$  και  $K=4096$ . Με άλλα λόγια, επιλέγουμε το μέγεθος του φίλτρου να είναι ίδιο σε διαστάσεις με την είσοδο, με αποτέλεσμα η έξοδος να είναι της μορφής  $1 \times 1 \times 4096$ , διότι μόνο μία στήλη, όσον αφορά το βάθος, εξάγεται από την εφαρμογή του συγκεκριμένου φίλτρου στην είσοδο (έχουν ίσες διαστάσεις, οπότε το φίλτρο εκτελεί μόνο μία επανάληψη πάνω στην είσοδο). Με αυτόν τον τρόπο λαμβάνουμε αποτέλεσμα ίδιο, με αυτό στην περίπτωση του πλήρως συνδεδεμένου επιπέδου.

### 2.6.2 Διαδικασία μετατροπής

Από τις δύο παραπάνω μετατροπές, εξαιρετικά χρήσιμη στην πράξη είναι η μετατροπή του πλήρως συνδεδεμένου επιπέδου σε συνελικτικό. Έστω μία αρχιτεκτονική συνελικτικού νευρωνικού δικτύου, η οποία λαμβάνει στην είσοδο της μία εικόνας διαστάσεων  $224 \times 224 \times 3$ . Στη συνέχεια, την πληροφορία επεξεργάζεται ένα σύνολο συνελικτικών και pooling

επιπέδων, προκειμένου να μειωθούν οι διαστάσεις της εισόδου και ο τελικός πίνακας να αποκτήσει μορφή  $7 \times 7 \times 512$ .

Από αυτό το σημείο και μετά χρησιμοποιούνται πλήρως συνδεδεμένα επίπεδα. Συγκεκριμένα, στην αρχιτεκτονική τύπου AlexNet [39], μετά από αυτό το σημείο εισάγονται δύο πλήρως συνδεδεμένα επίπεδα. Το πρώτο από αυτά έχει πλήθος νευρώνων 4096 και συνδέεται με το δεύτερο που έχει αντίστοιχα 1000 νευρώνες, οι οποίοι υπολογίζουν το αποτέλεσμα. Το τελευταίο, όπως προαναφέρθηκε είναι μία μετρική, για καθεμία από τις χίλιες παραπάνω κλάσεις, οι οποίες τελικά δείχνουν την πιθανότητα να ανήκει η είσοδος στη συγκεκριμένη κλάση.

Είναι δυνατό να μετατρέψουμε καθένα από τα τρία παραπάνω πλήρως συνδεδεμένα επίπεδα σε συνελκτικά, όπως ακριβώς περιγράψαμε και παραπάνω:

- Αντικατάσταση του πρώτου πλήρως συνδεδεμένου επιπέδου, στο οποίο εισάγεται πίνακας της μορφής  $7 \times 7 \times 512$ , με ένα συνελκτικό επίπεδο, το οποίο χρησιμοποιεί μέγεθος φίλτρου  $F=7$ , δίνοντας έτσι έξοδο  $1 \times 1 \times 4096$ .
- Αντικατάσταση του δεύτερου πλήρως συνδεδεμένου επιπέδου και αυτή τη φορά με συνελκτικό επίπεδο, μεγέθους φίλτρου  $F=1$ , προκειμένου να επιτευχθεί έξοδος της μορφής  $1 \times 1 \times 4096$ .
- Τελικά, αντίστοιχη μετατροπή του τρίτου πλήρως συνδεδεμένου επιπέδου, πάλι με συνελκτικό φίλτρου  $F=1$ , προκειμένου να λάβουμε την επιθυμητή έξοδο  $1 \times 1 \times 1000$ .

Κάθεμία από τις παραπάνω μετατροπές, στην πράξη, συμπεριλαμβάνει και αντίστοιχους μετασχηματισμούς, έτσι ώστε από τους πίνακες βαρών  $W$  των πλήρως συνδεδεμένων επιπέδων, να λάβουμε τα αντίστοιχα φίλτρα για τα συνελκτικά επίπεδα.

Για τη μετατροπή αυτή τελικά αξίζει να αναφερθεί ότι μας επιτρέπει να «ολισθήσουμε» το αρχικό δίκτυο, με αποδοτικό τρόπο, σε πολλά διαφορετικά τμήματα της εικόνας, σε μόνο μία εμπρόσθια διάδοση της πληροφορίας.

Για παράδειγμα, εάν μία εικόνα διαστάσεων  $224 \times 224$ , μετατραπεί, μέσω ενός συνόλου διαδικασιών, σε πίνακα της μορφής  $7 \times 7 \times 512$ , ή διαφορετικά, εάν το πλήθος της πληροφορίας, μειωθεί πολλαπλασιαστικά με έναν παράγοντα της τάξης του 32, τότε μία αντίστοιχη εικόνα, διαστάσεων  $384 \times 384$ , μέσω της ίδιας αρχιτεκτονικής, θα είχε σαν αποτέλεσμα πίνακα της μορφής  $12 \times 12 \times 512$ , καθότι  $384/32=12$ . Συνεχίζοντας με τα τρία πλήρως συνδεδεμένα επίπεδα, τα οποία μόλις μετατρέψαμε από συνελκτικά, θα λαμβάναμε πίνακα εξόδου διαστάσεων  $6 \times 6 \times 1000$ , διότι  $(12 - 7)/1 + 1 = 6$ . Η παραπάνω διαδικασία έχει σαν αποτέλεσμα, τελικά, αντί για ένα στοιχείο, να λάβουμε  $6 \times 6$  πίνακα για τη μετρική κάθε κλάσης, πάνω στην  $384 \times 384$  εικόνα.

Εφαρμόζοντας στο αρχικό συνελκτικό δίκτυο με τα πλήρως συνδεδεμένα επίπεδα κάθε φορά  $224 \times 224$  κομμάτια της συνολικά  $384 \times 384$  εικόνας, με βήμα 32 πίξελ, έχει αποτέλεσμα ίδιο με αυτό του δικτύου, στο οποίο μετατρέψαμε τα πλήρως συνδεδεμένα επίπεδα σε συνελκτικά.

Η εισαγωγή της πληροφορίας στο δίκτυο, μόνο μία φορά, είναι εμφανώς πιο αποδοτική από την εφαρμογή του ίδιου δικτύου πάνω σε 36 κομμάτια της εικόνας, καθ'ότι σε αυτά αντιστοιχούν ίδιοι υπολογισμοί, οι οποίοι και επαναλαμβάνονται.

Η παραπάνω τεχνική χρησιμοποιείται πολλές φορές στην πράξη, προκειμένου να επιτευχθεί υψηλότερη απόδοση. Παραδείγματος χάριν, συνηθισμένο φαινόμενο είναι, να μεγαλώνουμε τις διαστάσεις της εικόνας και να την εισάγουμε σε συνελκτικό νευρωνικό δίκτυο, το οποίο έχει υποστεί μετατροπή, όπως παραπάνω.

Στη συνέχεια, λάμβάνουμε το μέσο όρο του συνόλου των τιμών από κάθε τμήμα της εικόνας, με αποτέλεσμα ο μέσος όρος αυτός να αποτελεί ένα πιο αντικειμενικό δείκτη για την κλάση στην οποία ανήκει η εικόνα αυτή.

Τελικά, αν θέλαμε να εφαρμόσουμε το αρχικό δίκτυο πάνω σε μία εικόνα, με μικρότερο βήμα από αυτό του 32, θα έπρεπε να συμβούν πολλαπλές εμπρόσθιες διαδόσεις της πληροφορίας. Δηλαδή για  $stride$  ( $S=16$ ), θα έπρεπε να συνδυάσουμε τους πίνακες εξόδου από το δίκτυο δύο φορές. Μία φορά πάνω στην αρχική εικόνας και άλλη μία πάνω στην ίδια εικόνας, αλλά ενώ έχει ολισθήσει κατά 16 πίξελ σε πλάτος και ύψος.

## 2.7 Αρχιτεκτονικές Συνελκτικών Νευρωνικών Δικτύων

Οι συνηθισμένες αρχιτεκτονικές συνελκτικών νευρωνικών δικτύων αποτελούνται, όπως είδαμε και σε προηγούμενη παράγραφο, από τρία βασικά είδη επιπέδων. Το συνελκτικό επίπεδο, το επίπεδο Pooling (στην συντριπτική πλειοψηφία χρησιμοποιείται το max pooling) [40] και τέλος το πλήρως συνδεδεμένο επίπεδο.

Θα εντάξουμε σε αυτή την κατηγορία και το επίπεδο μη γραμμικότητας, το οποίο πρακτικά είναι μία συνάρτηση μη-γραμμικότητας, η οποία εφαρμόζεται σε κάθε στοιχείο του πίνακα των δεδομένων και εισάγει, όπως αναφέρεται και από το όνομά της, μη γραμμικότητα.

Παρακάτω θα αναφερθεί η διαδικασία, μέσω της οποίας τα παραπάνω επίπεδα συνδέονται μεταξύ τους, με συγκεκριμένους τρόπους, προκειμένου να υλοποιηθεί μία αποδοτική αρχιτεκτονική ενός συνελκτικού νευρωνικού δικτύου.

Για παράδειγμα:

```
INPUT -> [[CONV -> RELU]*N -> POOL?]*M -> [FC -> RELU]*K -> FC,
```

,όπου το σύμβολο \* υποδηλώνει επανάληψη της παρένθεσης στα αριστερά του τόσες φορές όσες το σύμβολο που βρίσκεται στα δεξιά του. Το σύμβολο ? μετά το επίπεδο Pooling, δείχνει ότι το επίπεδο αυτό είναι προαιρετικό και συνήθως ( $N \leq 3$ ), ( $M \geq 0$ ),

( $0 \leq K \leq 3$ ). Στη συνέχεια ακολουθούν παραδείγματα αρχιτεκτονικών, τα οποία ακολουθούν τον παραπάνω κανόνα.

- `INPUT -> FC` ,υλοποιεί απλώς έναν γραμμικό ταξινομητή  $N=M=K=0$
- `INPUT -> CONV -> RELU -> FC`
- `INPUT -> [CONV -> RELU -> POOL]*2 -> FC -> RELU -> FC` ,όπου παρατηρείται μόνο ένα συνελκτικό επίπεδο μεταξύ επιπέδων Pooling.
- `INPUT -> [CONV -> RELU -> CONV -> RELU -> POOL]*3 -> [FC -> RELU]*2 -> FC` . Σε αυτό το σημείο παρατηρούμε παραπάνω από ένα συνελκτικά επίπεδα μεταξύ διαδοχικών επιπέδων Pooling. Συνδεσμολογίες τέτοιας μορφής, αποτελούν γενικότητα



σε περιπτώσεις υλοποίησης πολύπλοκων και βαθιών δικτύων. Τα συνελικτικά επίπεδα, όταν συνδέονται διαδοχικά, χωρίς να παρεμβάλεται άλλο επίπεδο, πρακτικά υλοποιούν φίλτρα με ικανότητα εξαγωγής πιο περίπλοκων χαρακτηριστικών. Στη συνέχεια ακολουθεί το επίπεδο Pooling, το οποίο μειώνει δραματικά το μέγεθος της πληροφορίας, για χάρη της μείωσης παραμέτρων.

Αξίζει επίσης να αναφερθεί ότι συνήθως προτιμάται η διαδοχική σύνδεση συνελικτικών επιπέδων με μικρά φίλτρα, χωρίς συναρτήσεις μη γραμμικότητας ενδιάμεσα, σε σχέση με ένα μόνο επίπεδο, το οποίο διαθέτει φίλτρο μεγαλύτερων διαστάσεων [41].

Έστω ότι υλοποιείται αρχιτεκτονική, η οποία απαιτεί τρία συνελικτικά επίπεδα στη σειρά, με φίλτρα τάξης  $3 \times 3$ . Με αυτόν τον τρόπο το πρώτο επίπεδο «βλέπει» ένα  $3 \times 3$  τμήμα του πίνακα εισόδου. Το επίμενο επίπεδο, λόγω του ότι θα λάβει την έξοδο του πρώτου επιπέδου θα «βλέπει» ένα  $5 \times 5$  κομμάτι της αρχικής εισόδου, ενώ τελικά το τελευταίο επίπεδο σε σχέση με την αρχική είσοδο θα επεξεργαστεί ένα  $7 \times 7$  τμήμα της.

Σε αντίθετη περίπτωση, όπου χρησιμοποιείται μόνο ένα επίπεδο, θα πραγματοποιείται ένα μοναδικό φίλτρο της τάξης του  $7 \times 7$ , το οποίο και θα «βλέπει» όπως και πριν ένα  $7 \times 7$  κομμάτι της αρχικής εικόνας, αλλά με αρκετά μειονεκτήματα σε σχέση με την πρώτη περίπτωση.

Αρχικά, στη δεύτερη περίπτωση οι νευρώνες θα υλοποιούσαν μία συνάρτηση μη γραμμικότητας πάνω σε κάθε στοιχείο του πίνακα εισόδου, ενώ στην πρώτη περίπτωση, τα τρία διαδοχικά επίπεδα, υλοποιούν συναρτήσεις μη γραμμικότητας, με αποτέλεσμα τα χαρακτηριστικά που εξάγουν να είναι πιο εκφραστικά.

Στη συνέχεια, εάν θεωρήσουμε ότι ο αριθμός των καναλιών, δηλαδή η διάσταση του βάθους της εικόνας, εκφράζεται από την παράμετρο  $C$ , τότε ένα συνελικτικό επίπεδο με φίλτρο  $7 \times 7$ , θα αποτελούταν από  $C \times (7 \times 7 \times C) = 49C^2$  παραμέτρους.

Αντίθετα, στην πρώτη περίπτωση, τα τρία συνελικτικά επίπεδα θα συμπεριλάμβαναν  $3 \times (C \times (3 \times 3 \times C)) = 27C^2$  παραμέτρους. Διαισθητικά, η σύνδεση πολλών συνελικτικών επιπέδων με μικρά φίλτρα στη σειρά, σε σχέση με ένα επίπεδο, το οποίο διαθέτει μεγαλύτερων διαστάσεων φίλτρο, μας επιτρέπει να εκφράσουμε χαρακτηριστικά με μεγαλύτερη πολυπλοκότητα, αλλά και με μικρότερο αριθμό παραμέτρων.

Σαν μειονέκτημα, παρ'όλα αυτά, η επιλογή μεγαλύτερου αριθμού φίλτρων απαιτεί περισσότερη μνήμη, κατά τη διάρκεια της εφαρμογής του αλγορίθμου ανάστροφης διάδοσης, καθώς πρέπει να αποθηκευτούν όλα τα ενδιάμεσα αποτελέσματα των διαφορετικών συνελικτικών επιπέδων.

### 2.7.1 Επιλογή Μεγεθών και Παραμέτρων στα Επίπεδα

Στο σημείο αυτό θα ασχοληθούμε με κανόνες, οι οποίοι μας υποδεικνύουν την αποδοτική επιλογή παραμέτρων για κάθε επίπεδο. Στις προηγούμενες παραγράφους, έγινε απλώς αναφορά κάποιων παραμέτρων, δίχως να γίνει επεξήγηση πάνω στις συγκεκριμένες επιλογές των παραμέτρων αυτών.

Αρχικά, το επίπεδο εισόδου θα πρέπει να έχει διαστάσεις, οι οποίες σε αριθμό πίξελ, να είναι πολλαπλάσιες του δύο. Αρκετά παραδείγματα συμπεριλαμβάνουν διαστάσεις όπως 32 πίξελ, 64, 96, 224, 384 ή και 512.

Ακόμη, τα συνελκτικά επίπεδα θα πρέπει να χρησιμοποιούν φίλτρα μικρών διαστάσεων. Δηλαδή φίλτρα με πεδίο πρόσληψης 3x3 ή 5x5. Στη συνέχεια, συνίσταται η χρήση zero padding, τέτοιο ώστε να διατηρούνται οι διαστάσεις του πίνακα εισόδου, ενώ για βήμα (stride), επιλέγεται συνήθως τιμή S=1. Σε γενικότερες περιπτώσεις, όπως αναφέρθηκε και σε προηγούμενη παράγραφο, εάν επιλέξουμε φίλτρο μεγαλύτερων διαστάσεων, τότε προσαρμόζουμε το zero-padding, σύμφωνα με τη σχέση  $P = (F-1)/2$ , αποσκοπώντας στην διατήρηση των διαστάσεων του πίνακα εισόδου. Φίλτρα μεγαλύτερου μεγέθους, παρ'όλα αυτά, είναι συνηθισμένα μόνο στο πρώτο συνελκτικό επίπεδο, στο οποίο τροφοδοτούνται απ'ευθείας τα πίξελ της εικόνας. Τα επίπεδα pooling είναι υπεύθυνα με την μείωση των διαστάσεων της πληροφορίας, η οποία εισάγεται σαν είσοδος ή λαμβάνεται σαν έξοδος από τα επίπεδα του δικτύου. Η πιο συνηθισμένη ρύθμιση είναι max pooling, με φίλτρο 2x2 και stride S=2.

Με αυτόν τον τρόπο παραμένει μόνο το 25% της πληροφορίας του πίνακα εισόδου και έτσι μειώνουμε τον αριθμό των παραμέτρων. Ένας ακόμα συνδυασμός παραμέτρων, ο οποίος συνηθίζεται είναι φίλτρα 3x3 και S=2. Φίλτρα μεγαλύτερα από αυτά δε συνηθίζονται κατά την υλοποίηση των δικτύων, διότι με αυτόν τον τρόπο χάνεται πολύ μεγαλύτερο κομμάτι της πληροφορίας και οδηγούμαστε σε χαμηλότερες αποδόσεις.

Κάνοντας χρήση των παραπάνω κανόνων, ακόμη, μας δίνεται η δυνατότητα να απαλαγούμε από περιττούς υπολογισμούς. Διατηρούμε σταθερές διαστάσεις στον πίνακα της πληροφορίας στα υπόλοιπα επίπεδα, ενώ τα επίπεδα pooling είναι τα μόνα υπεύθυνα για τη μείωση των διαστάσεων της πληροφορίας. Διαφορετικά, εάν για παράδειγμα χρησιμοποιήσουμε βήμα (stride) διάφορο του ένα ή καθόλου zero padding, τότε θα πρέπει να υπολογίσουμε ακριβώς τις διαστάσεις των πινάκων εισόδου και εξόδου για κάθε επίπεδο, προκειμένου να είμαστε σίγουροι ότι το δίκτυο έχει τον απαραίτητο αριθμό συνδέσεων και δεν παρουσιάζεται σφάλμα κατά την υλοποίησή του, με τη δοθείσα επιλογή παραμέτρων.

### 2.7.2 Επιλογή τιμής για Stride S=1

Στην πράξη, τα φίλτρα μικρότερου μεγέθους παρουσιάζουν υψηλότερη απόδοση, ενώ με αυτόν τον τρόπο διατηρούμε και ίδιες διαστάσεις από την είσοδο στην έξοδο, οπότε ελέγχουμε τις διαστάσεις μόνο μέσω των επιπέδων pooling.

### 2.7.3 Χρήση zero-padding

Εκτός από τη διατήρηση των διαστάσεων της πληροφορίας, κάνοντας χρήση του zero padding, αυξάνεται η απόδοση. Χωρίς αυτήν την παράμετρο, η πληροφορία στην άκρη του πίνακα εισόδου, από επίπεδο σε επίπεδο θα χανόταν, λόγω του ότι θα μειωνόταν η διάστασή του.

### 2.7.4 Συμβιβασμός λόγω των περιορισμών μνήμης

Σε αρκετές περιπτώσεις (,κυρίως νωρίς στην υλοποίηση αρχιτεκτονικών για συνελκτικά νευρωνικά δίκτυα), το ποσό μνήμης, το οποίο απαιτείται, μπορεί να αυξηθεί κατά πολύ, όταν χρησιμοποιούμε τους παραπάνω κανόνες.

Για παράδειγμα, επεξεργάζοντας μία εικόνα διαστάσεων  $224 \times 224 \times 3$ , με 64 φίλτρα διαστάσεων  $3 \times 3$  και zero padding με τιμή  $P=1$ , θα έχει σαν αποτέλεσμα τρεις πίνακες με μέγεθος  $224 \times 224 \times 64$ . Αυτό έχει σαν αποτέλεσμα περίπου δέκα εκατομμύρια μεταβλητές ή 72 MB μνήμης (ανά εικόνα, για μεταβλητές και παραγώγους μαζί). Σαν αποτέλεσμα, σε πολλές περιπτώσεις η κάρτα γραφικών, με την οποία υλοποιείται η εκπαίδευση του δικτύου, χάνει σε απόδοση λόγω της έλλειψης μνήμης.

Ο τρόπος, λοιπόν, με τον οποίο αντιμετωπίζουμε το πρόβλημα αυτό είναι να συμβιβαστούμε σε τιμές παραμέτρων, τέτοιες ώστε να ελαχιστοποιήσουμε τη μνήμη. Στην πράξη, κατά την υλοποίηση της πλειονότητας των αρχιτεκτονικών, συμβοβαζόμαστε μόνο στην επιλογή παραμέτρων για το πρώτο συνελκτικό επίπεδο του δικτύου. Δηλαδή, μπορεί αντί για τρία συνελκτικά επίπεδα με μικρά φίλτρα, να επιλεγεί ένα, αλλά με φίλτρο  $7 \times 7$  και stride 2 ή ακόμα και φίλτρα διαστάσεων  $11 \times 11$  και stride 4.

## 2.8 Παράμετροι Μάθησης

### 2.8.1 Βελτιστοποίηση (Optimization)

Στα πεδία των μαθηματικών, της μαθηματικής βελτιστοποίησης και του μαθηματικού προγραμματισμού, ο όρος βελτιστοποίηση αναφέρεται στην επιλογή κάποιου στοιχείου, με βάση ένα δεδομένο κριτήριο, ανάμεσα σε ένα πλήθος διαφορετικών επιλογών, με σκοπό το στοιχείο αυτό να αυξήσει την απόδοση επίλυσης του προβλήματος.

Στην απλούστερη περίπτωση, όταν αντιμετωπίζουμε ένα πρόβλημα βελτιστοποίησης, προσπαθούμε να μεγιστοποιήσουμε ή να ελαχιστοποιήσουμε μία πραγματική συνάρτηση. Δηλαδή, επιλέγουμε συγκεκριμένες εισόδους για να εισάγουμε στο σύστημα, με σκοπό να υπολογίσουμε την τιμή που θα εξάγει για τις τιμές αυτές. Η έννοια της βελτιστοποίησης έχει γενικευτεί και σε διάφορους άλλους τομείς, αποτελώντας έτσι ένα πολύ ευρύ αντικείμενο όσον αφορά το πεδίο των εφαρμοσμένων μαθηματικών. Σε γενική περίπτωση, η βελτιστοποίηση συμπεριλαμβάνει την εύρεση της βέλτιστης λύσης, ανάμεσα σε ένα σύνολο πιθανών λύσεων για μία αντικειμενική συνάρτηση και για ένα σαφώς ορισμένο σύνολο εισόδων.

Τα σύνολα εισόδων και πιθανών βέλτιστων λύσεων, παρ' όλ'αυτά, μπορεί να παρουσιάζουν μεγάλο πλήθος και ποικιλομορφία, με αποτέλεσμα η εύρεση της βέλτιστης αυτής λύσης να αποτελεί εξαιρετικά χρονοβόρα διαδικασία.

Πιο συγκεκριμένα, στα πεδία της μηχανικής μάθησης, αλλά και της στατιστικής εκτίμησης, η βελτιστοποίηση αναφέρεται στο κομμάτι της μεγιστοποίησης ή ελαχιστοποίησης μία αντικειμενικής συνάρτησης στη μορφή αθροίσματος

$$Q(\mathbf{w}) = \frac{1}{n} \sum_{i=1}^n Q_i(\mathbf{w})$$

, όπου η παράμετρος  $\mathbf{w}$ , που ελαχιστοποιεί την τιμή  $Q(\mathbf{w})$ , πρέπει να υπολογιστεί.

Όταν καλούμαστε να ελαχιστοποιήσουμε τη παραπάνω συνάρτηση, η μέθοδος φθίνουσας παραγώγου (gradient descent) είτε στην απλή της μορφή είτε στη μορφή που η παράγωγος υπολογίζεται σε ένα τμήμα(batch), της συνολικής εισόδου, θα εκτελούνται οι παρακάτω επαναλήψεις:

$$w := w - \eta \nabla Q(w) = w - \eta \sum_{i=1}^n \nabla Q_i(w) / n.$$

Η παράμετρος  $\eta$ , είναι το μέγεθος που αναφέρεται στο πόσο έχουμε επιλέξει να μειώσουμε τις τιμές προς την κατεύθυνση της παραγώγου και ονομάζεται βαθμός μάθησης (learning rate).

### 2.8.2 Optimizers ( Αλγόριθμοι βελτιστοποίησης)

Όσον αφορά το πεδίο των νευρωνικών δικτύων και ειδικότερα αυτό των συνελκτικών νευρωνικών δικτύων, έχουμε ένα μεγάλο πλήθος από αλγόριθμους βελτιστοποίησης. Στα πλαίσια της παρούσας διπλωματικής εργασία και λαμβάνοντας υπ' όψη τις ιδιαιτερότητες του προβλήματος, το οποίο έχουμε κληθεί να αντιμετωπίσουμε επιλέξαμε δύο από αυτούς προκειμένου να βελτιστοποιήσουμε την απόδοση.

Ο πρώτος ονομάζεται Adam και ο δεύτερος SGD. Καθένας από αυτούς, ανάλογα με το συνδυασμό των υπόλοιπων υπερπαραμέτρων και την επιλογή μοντέλων συνελκτικών νευρωνικών δικτύων, έχει σαν αποτέλεσμα, όπως θα δούμε και σε επόμενο κεφάλαιο, την εμφάνιση διαφορών στη σύγκλιση, το χρόνο εκπαίδευσης, το overfitting, την απόδοση κ.α.

#### 2.8.2.1 SGD (stochastic gradient descent)

Στην περίπτωση αυτού του αλγορίθμου, η παράγωγος της προηγούμενης συνάρτησης  $Q(w)$ , υπολογίζεται από από την παράγωγο ενός μοναδικού παραδείγματος.

$$w := w - \eta \nabla Q_i(w).$$

Όσο ο αλγόριθμος εφαρμόζεται πάνω στο σύνολο των δεδομένων εκπαίδευσης, υλοποιείται ο παραπάνω κανόνας για κάθε μοναδικό αντικείμενο εισόδου, στην περίπτωση μας για κάθε εικόνα.

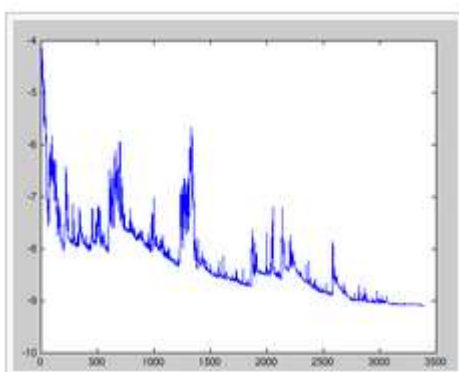
Ο αριθμός των επαναλήψεων πάνω στο σύνολο των δεδομένων εκπαίδευσης συνήθως πρέπει να είναι μεγάλος προκειμένου το δίκτυο να συγκλίνει. Δηλαδή, είναι αρκετά πιθανό να απαιτούνται πολλές επαναλήψεις πάνω στο σύνολο των δεδομένων εκπαίδευσης [11].

Στην περίπτωση αυτή πρέπει να αλλάζουμε τη σειρά με την οποία τα αντικείμενα εισόδου εισάγονται στο δίκτυο, προκειμένου να αποφευχθούν επαναλήψεις. Σε αρκετές περιπτώσεις χρησιμοποιείται ένας προσαρμοστικός βαθμός μάθησης, με σκοπό πάλι τη σύγκλιση του δικτύου [42].

Σε ψευδοκώδικα έχουμε:

- Επέλεξε ένα αρχικό διάνυσμα παραμέτρων  $w$  και βαθμό μάθησης  $\eta$ .
- Επανάλαβε μέχρι μία συγκεκριμένη τιμή ελαχίστου έχει επιτευχθεί:
  - “Ανακάτεψε” τυχαία τα δεδομένα στο σύνολο εκπαίδευσης
  - Για  $i = 1, 2, \dots, n$ , εκτέλεσε:
    - $w := w - \eta \nabla Q_i(w)$ .

Παρακάτω παρουσιάζεται ένα γράφημα, το οποίο δείχνει το σφάλμα σε συνάρτηση με τον αριθμό των επαναλήψεων. Όπως φαίνεται, μέχρι το δίκτυο να συγκλίνει παρατηρούνται αρκετές αυξομειώσεις, οι οποίες μειώνονται σε αριθμό και σε διακύμανση όσο ανεβαίνει ο αριθμός των επαναλήψεων.



Εικόνα 2.9 :Γράφημα σφάλματος συναρτήσει εποχών για ένα τυχαίο πρόβλημα, κάνοντας χρήση του αλγόριθμου μάθησης SGD. Παρατηρούνται αρκετές αυξομειώσεις μέχρι να συμβεί σύγκλιση.

Τελικά αξίζει να αναφερθεί ότι κάνοντας χρήση του παραπάνω αλγόριθμου και μειώνοντας το βαθμό μάθησης με κατάλληλο ρυθμό, καταλήγουμε σχεδόν βέβαια σε ολικό ή τοπικό ελάχιστο.

### 2.8.2.2 Momentum

[43] Αξίζει σε αυτό το σημείο να αναφερθεί ότι σε αρκετές περιπτώσεις, όσον αφορά την επιλογή παραμέτρων μάθησης, εκτός από την επιλογή αλγορίθμων βελτιστοποίησης, χρησιμοποιείται και η μέθοδος ορμής (momentum method).

Για παράδειγμα στον αλγόριθμο SGD με μέθοδο ορμής, αποθηκεύεται η ενημέρωση  $\Delta w$  σε κάθε επανάληψη. Η επόμενη ενημέρωση των βαρών είναι ένας γραμμικός συνδυασμός της πιο πρόσφατα υπολογιζόμενης παραγώγου και της προηγούμενης ενημέρωσης. Έχουμε:

$$\begin{aligned}\Delta w &:= \alpha \Delta w - \eta \nabla Q_i(w) \\ w &:= w + \Delta w\end{aligned}$$

, και έτσι οδηγούμαστε στη μοναδική σχέση

$$w := w - \eta \nabla Q_i(w) + \alpha \Delta w$$

### 2.8.2.3 Adam (adaptive moment estimation)

Ο παραπάνω αλγόριθμος βελτιστοποίησης είναι πρακτικά μία μέθοδος, η οποία υπολογίζει προσαρμοστικούς ρυθμούς μάθησης για κάθε παράμετρο. Εκτός του ότι αποθηκεύει ένα εκθετικά φθίνοντα μέσο όρο από παλιές παραγώγους υψωμένες στη δεύτερη δύναμη  $u_t$ , εκτελεί ακριβώς το ίδιο και για τις παραγώγους στην πρώτη δύναμη  $m_t$ , το οποίο είναι παρόμοιο με τη μέθοδο ορμής.

Χαρακτηριστικό του παραπάνω αλγόριθμου είναι η επιλογή επίπεδων τοπικών ελαχίστων, στο χώρο σφαλμάτων, το οποίο σημαίνει ότι καταλήγει να συγκλίνει σε περιοχές που τα σφάλματα είναι ελάχιστο και δεν αυξομειώνεται, αλλά μένει σταθερό στην «κοντινή» επιφάνεια. Υπολογίζουμε τις παραπάνω δύο τιμές ως εξής:

$$\begin{aligned}m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2\end{aligned}$$

Τα παραπάνω μεγέθη είναι αντίστοιχα εκτιμήσεις της μέσης τιμής και της διακύμανσης. Όταν αρχικοποιούνται σε διανύσματα μηδενικών τιμών, παρατηρείται ότι τείνουν στο μηδέν, ειδικά στα αρχικά βήματα αλλά και όσο ο βαθμός μάθησης είναι μικρός.

Αυτό διορθώνεται με τη σωστή εκτίμηση των παραπάνω παραμέτρων, δίχως να λαμβάνεται υπ' όψην η τάση η πόλωση στο μηδέν.

$$\begin{aligned}\hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1 - \beta_2^t}\end{aligned}$$

Από τα παραπάνω λαμβάνεται η σχέση σνημέρωσης των βαρών για τον αλγόριθμο Adam.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t$$

## 2.9 Διάφορες Αρχιτεκτονικές Συνελικτικών Νευρωνικών Δικτύων

Έχοντας υπ' όψην τους παραπάνω κανόνες, στο πεδίο των συνελικτικών νευρωνικών δικτύων, έχει υλοποιηθεί ένας μαγάλος αριθμός από διαφορετικές αρχιτεκτονικές. Κάθεμία από αυτές έχει και τα δικά της προτερήματα και μειονεκτήματα. Επιγραμματικά αναφέρουμε μερικές από αυτές παρακάτω:

1. **LeNet** [44]
2. **AlexNet** [45]
3. **ZF Net**
4. **GoogleNet** [46]
5. **VGGNet**
6. **ResNet**

Στη συνέχεια της παρούσας διπλωματικής εργασίας θα ασχοληθούμε με τρεις διαφορετικές αρχιτεκτονικές συνελκτικών νευρωνικών δικτύων και θα συγκρίνουμε τα αποτελέσματά που έχουν πάνω στο ίδιο πρόβλημα, ενώ αλλάζουμε κάποιες παραμέτρους τους ή εφαρμόζουμε διάφορες μεθόδους στις εικόνες, προκειμένου να αυξήσουμε την απόδοση.

Θα ασχοληθούμε με τις αρχιτεκτονικές VGGNet, Resnet αλλά και με μία καινούρια, μικρότερης έκτασης, η οποία υλοποιήθηκε από εμάς, την οποία και θα συγκρίνουμε με τις δύο παραπάνω, που θεωρούνται από τις πιο αποδοτικές που υπάρχουν.

## Κεφάλαιο 3<sup>ο</sup> Μοντέλα και Μέθοδοι Προεπεξεργασίας

### 3.1 Μοντέλα Συνελικτικών Νευρωνικών Δικτύων

Όπως αναφέρθηκε και στο τέλος του προηγούμενου κεφαλαίου, υπάρχει ένας πολύ μεγάλος αριθμός από αρχιτεκτονικές συνελικτικών νευρωνικών δικτύων. Δηλαδή, υπάρχουν πολλοί διαφορετικοί τρόποι, με τους οποίους μπορούμε να συνδέσουμε επίπεδα στη σειρά για να επεξεργαστούμε την είσοδο.

Εμείς στα πλαίσια της παρούσας διπλωματικής εργασίας, θα αναλύσουμε τρεις διαφορετικές αρχιτεκτονικές. Η πρώτη από αυτές ονομάζεται VGGNet, η δεύτερη ResNet και η τρίτη είναι μία αρχιτεκτονική την οποία υλοποιήσαμε εμείς, είναι πολύ μικρότερων διαστάσεων και έχουμε αποσκοπούμε στο να τη συγκρίνουμε με τις δύο παραπάνω.

#### 3.1.1 VGGNet

Ανάμεσα σε πολλούς διαγωνιζόμενους, στο διαγωνισμό ILSVRC 2014 [47], συμμετείχαν οι Karen Simonyan και Andrew Zisserman. Η αρχιτεκτονική, με την οποία έλαβαν μέρος στο διαγωνισμό ονομάστηκε VGGNet. Η βασική συνεισφορά της στο πεδίο της μηχανικής μάθησης ήταν η υποστήριξη της θέσης ότι ο αριθμός των επιπέδων σε ένα συνελικτικό νευρωνικό δίκτυο, ή αλλιώς, βάθος του δικτύου, είναι πολύ βασική προϋπόθεση για την αύξηση της απόδοσης.

Από τα μοντέλα που παρουσίασαν, την υψηλότερη απόδοση είχε το μοντέλο με το μεγαλύτερο βάθος στην αρχιτεκτονική του. Αποτελούνταν από 16 συνολικά σε πλήθος συνελικτικά και πλήρως συνδεδεμένα επίπεδα, ενώ από τα πρώτα μέχρι τα τελευταία επίπεδα παρατηρείται αρχιτεκτονική με μεγάλο βαθμό ομοιογένειας. Όλο το δίκτυο στηρίζεται αποκλειστικά σε συνελικτικά επίπεδα, με μέγεθος φίλτρου 3x3, τα οποία συνδέονται σε σειρά, προκειμένου να αυξηθεί η πολυπλοκότητα στην εξαγωγή χαρακτηριστικών και σε επίπεδα Pooling, με φίλτρο 2x2.

Μειονεκτήματα, παρ'όλαυτά της παραπάνω υλοποίησης είναι οι πρακτικές δυσκολίες στην αξιολόγηση, αλλά και οι μεγάλες απαιτήσεις σε μνήμη, λόγω του αριθμού παραμέτρων (140 M). Οι περισσότερες από αυτές, όμως, επειδή βρίσκονται στα πρώτα πλήρως συνδεδεμένα επίπεδα της αρχιτεκτονικής, είναι δυνατόν να αφαιρεθούν από το σύστημα, μέσω της εξαγωγής των παραπάνω επιπέδων. Στη συνέχεια, και κατόπιν αντίστοιχης μελέτης, αποδείχτηκε ότι η αφαίρεση των επιπέδων αυτών έχει σαν αποτέλεσμα τη μείωση των απαραίτητων παραμέτρων σε μεγάλο βαθμό με μηδενική αντίστοιχη μείωση της απόδοσης.

##### 3.1.1.1 Αρχιτεκτονική VGGNet16

Όσον αφορά την υλοποίηση του παραπάνω δικτύου, υπάρχουν διάφορες επιλογές για το πλήθος των συνελικτικών επιπέδων που μεσολαβούν ανάμεσα στα επίπεδα pooling. Εμείς επιλέξαμε να χρησιμοποιήσουμε την αρχιτεκτονική εκείνη, με 16 συνολικά σε πλήθος



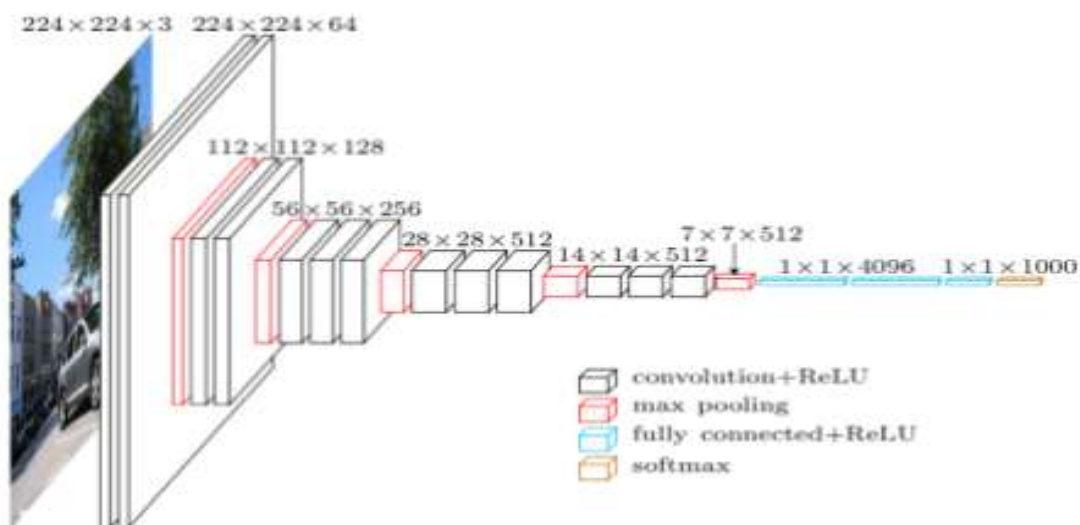
επίπεδα, λόγω του ότι συγκριτικά με το πλήθος των παραμέτρων που χρησιμοποιούνται, παρουσιάζει υψηλή απόδοση, ενώ δεν επιβαρύνει κατά πολύ τη μνήμη. Αρχικά, η εικόνα που εισάγεται στο δίκτυο έχει διαστάσεις  $224 \times 224 \times 3$ . Αυτό σημαίνει ότι υπάρχουν τρία κανάλια, εκ των οποίων το καθένα έχει πλάτος ίσο με το ύψος, 224 πίξελ. Πριν η εικόνα περάσει στο επόμενο επίπεδο, αλλά και πριν από κάθε συνελικτικό επίπεδο, εφαρμόζεται zero padding, με τιμή  $P=1$ .

Ακόμη, στο τέλος κάθε συνελικτικού επιπέδου, εφαρμόζεται συνάρτηση μη γραμμικότητας (relu). Μετά το επίπεδο εισόδου ακολουθούν δύο συνελικτικά επίπεδα, τα οποία έχουν το καθένα 64 φίλτρα διαστάσεων  $3 \times 3$  και στη συνέχεια το επίπεδο pooling, το οποίο εφαρμόζει τη συνάρτηση max, με φίλτρο  $2 \times 2$  και stride 2.

Στη συνέχεια, συνδέονται ακόμα δύο συνελικτικά επίπεδα με ακριβώς τον ίδιο τρόπο, αλλά ο αριθμός των φίλτρων αυτή τη φορά είναι 128. Έπειτα, εισάγουμε άλλα τρία συνελικτικά επίπεδα, με τη μόνη διαφορά σε σχέση με πριν να παρουσιάζεται στον αριθμό των φίλτρων των συνελικτικών επιπέδων. Από 128 γίνεται 256. Τέλος, με ακριβώς τον ίδιο τρόπο, συνδέουμε στη σειρά και άλλα έξι συνελικτικά επίπεδα, τα οποία είναι ρυθμισμένα με ακριβώς τις ίδιες παραμέτρους, απλώς ο αριθμός των φίλτρων για ακόμα μία φορά αυξάνεται σε 512.

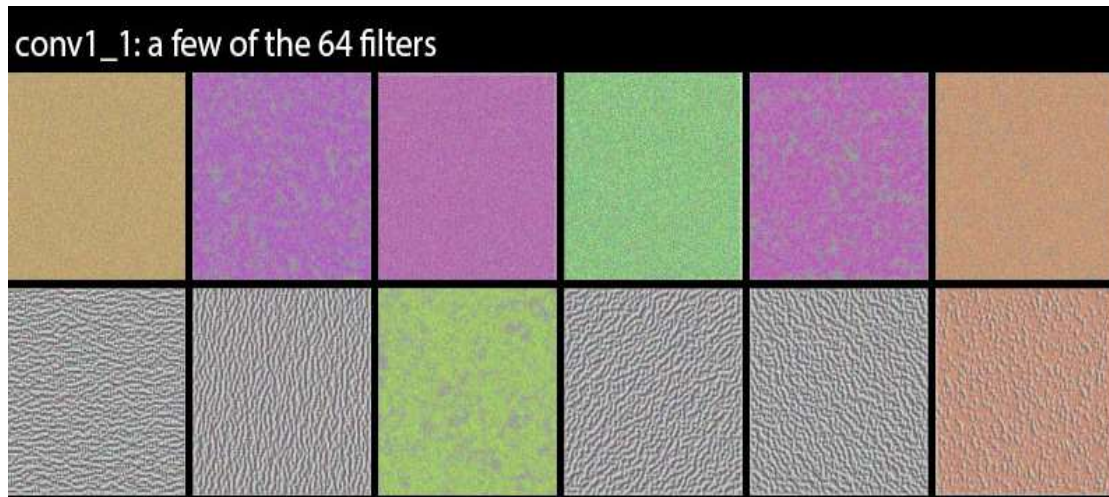
Στα επίπεδα αυτά συμβαίνει η εξαγωγή χαρακτηριστικών, η οποία είναι ακρετά έντονη, λόγω του βάθους του δικτύου, αλλά και του μεγάλου αριθμού των φίλτρων. Τα επόμενα επίπεδα είναι πλήρως συνδεδεμένα και είναι υπεύθυνα για τη λήψη της απόφασης, όσον αφορά την κατηγορία στην οποία ανήκει η εικόνα. Ακόμα, στη έξοδο των δύο εκ των τριών πλήρως συνδεδεμένων επιπέδων, έχει εφαρμοστεί συνάρτηση μη γραμμικότητας (relu). Τα επίπεδα αυτά, έχουν το καθένα 4096 νευρώνες και αποασχολούν ένα μεγάλο τμήμα της μνήμης που απαιτεί το δίκτυο.

Το τελευταίο επίπεδο είναι και αυτό πλήρως συνδεδεμένο και ακολουθείται από μία συνάρτηση softmax. Διαθέτει 1000 νευρώνες και είναι το επίπεδο εξόδου. Δηλαδή, το τελευταίο αποφασίζει τις μετρικές των κλάσεων στις οποίες ανήκει η εικόνα. Στα πλαίσια της παρούσας διπλωματικής εργασίας, ανάμεσα σε διάφορες αλλαγές που θα εφαρμόσουμε στο δίκτυο, θα αλλάξουμε και τον αριθμό των νευρώνων του τελευταίου επιπέδου από 1000 σε 2, διότι οι κατηγορίες στις οποίες θέλουμε να ταξινομήσουμε τις εικόνες είναι δύο.

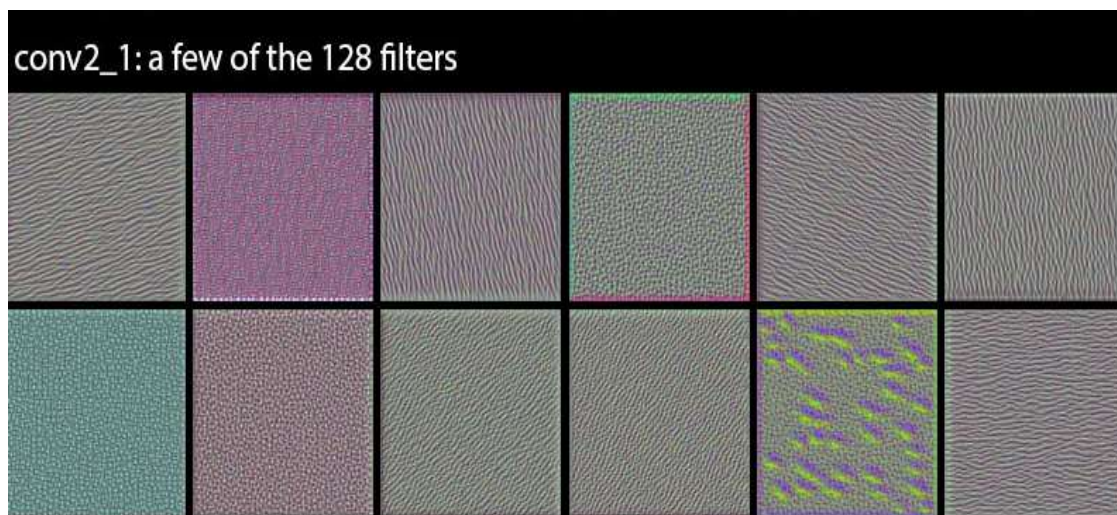


Εικόνα 3.1 : Παρουσίαση δομής και επιπέδων αρχιτεκτονικής VGGNet16.

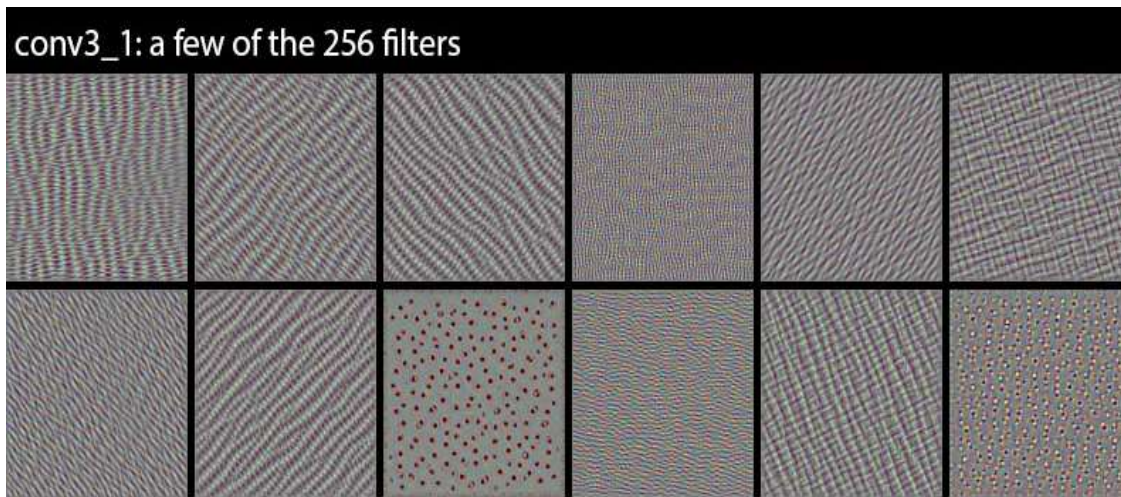
Βασικό χαρακτηριστικό της παραπάνω αρχιτεκτονικής, στο οποίο οφείλεται και η υψηλή της απόδοση είναι τα διαδοχικά συνελκτικά επίπεδα αύξοντα αριθμού φίλτρων. Αυτό έχει σαν αποτέλεσμα να εξάγονται από τα τελευταία συνελκτικά επίπεδα χαρακτηριστικά μεγάλης πολυπλοκότητας, όπως φαίνεται παρακάτω:



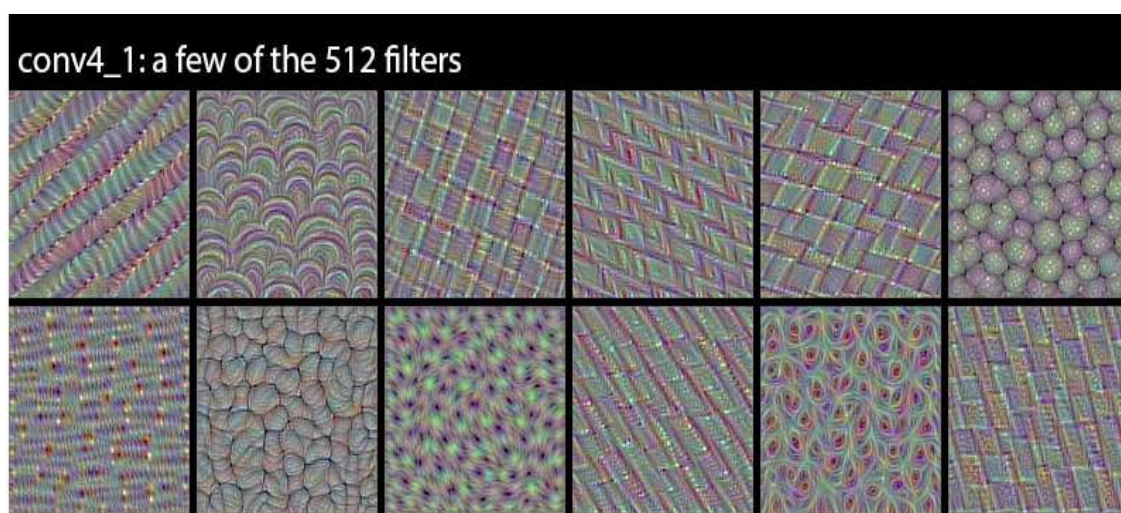
Εικόνα 3.2 : Περιπτώσεις από 64 συνελκτικά φίλτρα του πρώτου συνελκτικού επιπέδου



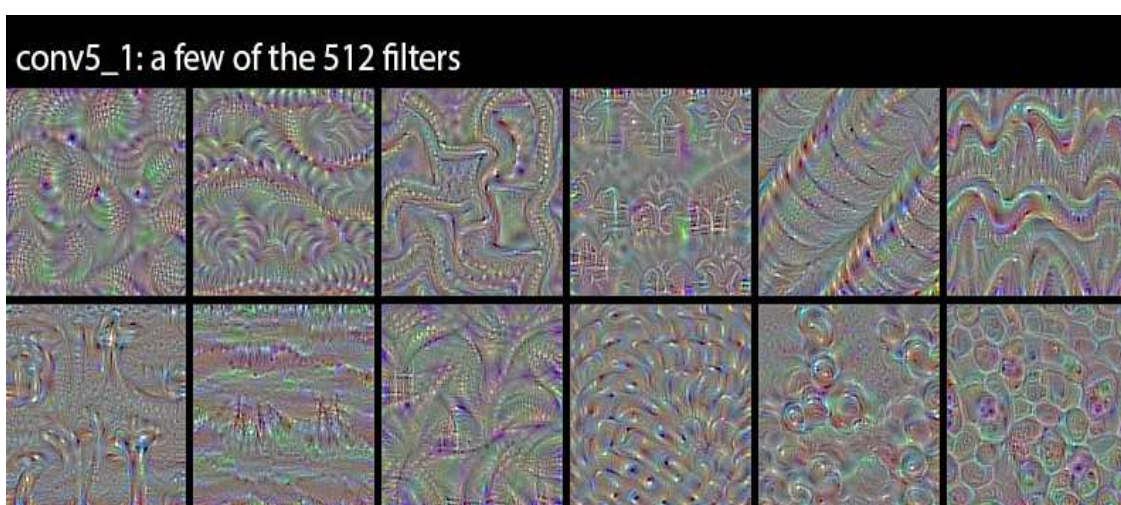
Εικόνα 3.3 : Περιπτώσεις από 128 συνελκτικά φίλτρα του δεύτερου συνελκτικού επιπέδου



Εικόνα 3.4 : Περιπτώσεις από 256 συνελκτικά φίλτρά του τρίτου συνελκτικού επιπέδου



Εικόνα 3.5 : Περιπτώσεις από 512 συνελκτικά φίλτρά του τέταρτου συνελκτικού επιπέδου



Εικόνα 3.6 : Περιπτώσεις από 512 συνελκτικά φίλτρά του πέμπτου συνελκτικού επιπέδου

### 3.1.2 ResNet

Η δεύτερη, σε σειρά, αρχιτεκτονική, την οποία υλοποιήσαμε στα πλαίσια της παρούσας διπλωματικής εργασίας ονομάζεται ResNet. Ανάμεσα σε πολλές διαφορετικές αρχιτεκτονικές στο διαγωνισμό ILSVRC 2015 [48], ήταν αυτή με την καλύτερη απόδοση, παρουσιάζοντας σφάλμα μόνο 3,25% (top 5 error rate). Επιτρέπει αποδοτική εκπαίδευση με αντίστοιχο αποτέλεσμα σε πολύ μεγάλο βάθος, δηλαδή από 100 μέχρι και 1000 επίπεδα.

Σε περιπτώσεις αρχιτεκτονικών με μεγάλο βάθος, όταν το δίκτυο αρχίζει και συγκλίνει, ενώ εκπαιδεύεται, παρατηρείται πρόβλημα μείωσης της ακρίβειας, αλλά και παράλληλα στην ικανότητα γενίκευσης. Δηλαδή, όσον αφορά το παραπάνω, η ακρίβεια αυξάνεται, αλλά με εξαιρετικά φθίνοντα ρυθμό και στη συνέχεια παρατηρείται σημαντικά μεγάλη μείωση της. Όταν συγκρίνουμε, για παράδειγμα, δύο αρχιτεκτονικές με ίδια αλληλουχία επιπέδων, αλλά η δεύτερη έχει μεγαλύτερο βάθος, υπάρχουν δύο περιπτώσεις

- Πρώτον, τα δύο δίκτυα, δηλαδή το ρηχό (αυτό με τα λιγότερα επίπεδα σε αριθμό) και το βαθύ (αυτό με τα περισσότερα επίπεδα σε αριθμό), θα διαφέρουν στην ακρίβεια των προβλέψεών τους και συγκεκριμένα το βαθύ θα έχει καλύτερη απόδοση.
- Δεύτερον, και στη χειρότερη για εμάς περίπτωση, τα δύο αυτά δίκτυα, δε θα διαφέρουν σε ακρίβεια, παρά την πιο «ακριβή» από πλευράς παραμέτρων αρχιτεκτονική του δικτύου με το μεγαλύτερο βάθος

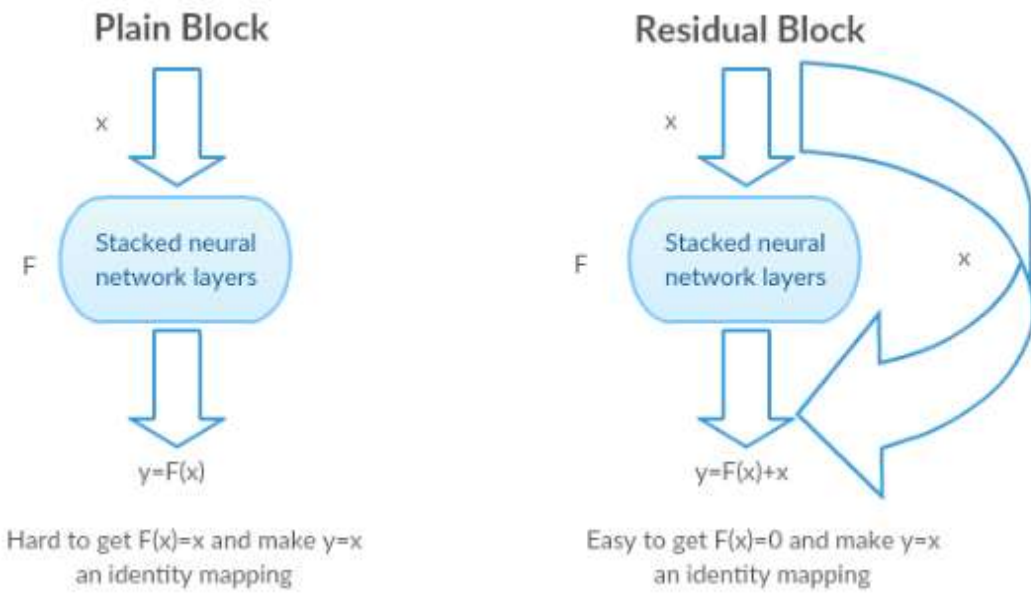
Παρότι ο μεγαλύτερος αριθμός σε παραμέτρους, που εξασφαλίζει μία πιο βαθιά αρχιτεκτονική, θα έπρεπε να σημαίνει και αντίστοιχη αύξηση της απόδοσης, έχει παρατηρηθεί σε πρακτικές εφαρμογές, ότι πολύ βαθιές αρχιτεκτονικές δεν εξασφαλίζουν υψηλή απόδοση. Αντίθετα πολλές φορές δεν αποδίδουν τόσο καλά όσο πιο ρηχές αντίστοιχες αρχιτεκτονικές. Δηλαδή, παρατηρείται το παραπάνω φαινόμενο, κατά το οποίο ένα πιο βαθύ δίκτυο συγκλίνει σε μία τιμή με φθίνοντα ρυθμό και στη συνέχεια μειώνεται κατά πολύ η απόδοσή του.

Γι αυτό το λόγο επιλέξαμε, όπως θα δούμε και παρακάτω μία από τις αρχιτεκτονικές τύπου ResNet, με σχετικά μικρό αριθμό επιπέδων σε σχέση με τις υπόλοιπες.

#### 3.1.2.1 Αντιμετώπισης μείωσης απόδοσης

Σε αντίθεση με την υλοποίηση μόνο ενός πίνακα παραμέτρων ανάμεσα σε δύο επίπεδα συνδεδεμένα στη σειρά, μπορούμε να «περνάμε» κάποια από αυτά και να δημιουργούμε συνδέσεις μεταξύ επιπέδων που δεν είναι συνεχόμενα το ένα με το άλλο [8].

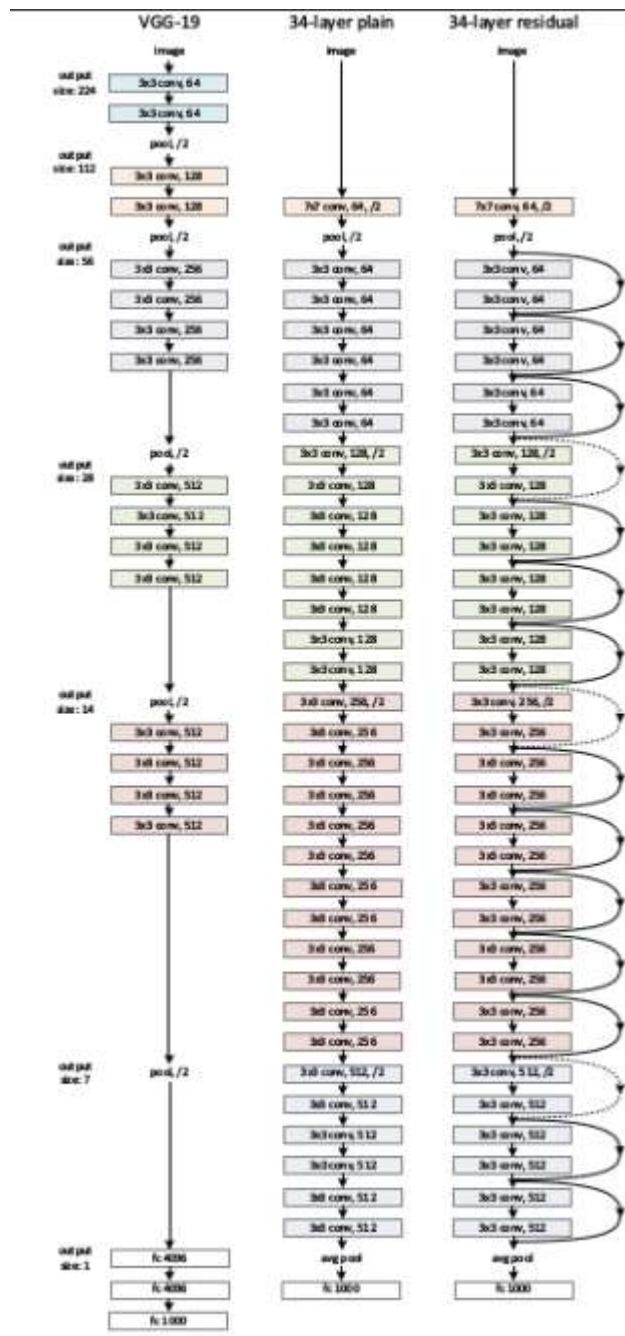
Σύμφωνα με την παραπάνω λογική, μπορούμε να θεωρήσουμε τη συνάρτηση  $F(x) = H(x) - x$ , η οποία γίνεται  $H(x) = F(x) + x$ , όπου  $F$  και  $x$  αναπαριστούν αντίστοιχα τα συνδεδεμένα επίπεδα, τα οποία δε βρίσκονται στη σειρά και την ταυτοτική συνάρτηση (είσοδος = έξοδος). Ο λόγος ύπαρξης της συνάρτησης αυτής είναι, κυρίως, η πεποίθηση ότι η συνάρτηση  $F$  βελτιστοποιείται πιο εύκολα σε σχέση με την  $H$  [8].



Εικόνα 3.7 : Απλουστευμένο παράδειγμα απλής και υπολειπόμενης σύνδεσης μεταξύ επιπέδων.

### 3.1.2.2 Χαρακτηριστικά υλοποίησης ResNet

1. Χρήση φίλτρων  $3 \times 3$  στα συνελκτικά επίπεδα και όχι μεγαλύτερων διαστάσεων.
2. Μείωση του όγκου της πληροφορίας, η οποία μεταφέρεται ανάμεσα στα επίπεδα, χρησιμοποιώντας επίπεδα pooling με βήμα 2.
3. Χρήση επιπέδου pooling, το οποίο υλοποιεί τη συνάρτηση μέσου όρου και όχι τη συνάρτηση μεγίστου. Στη συνέχεια τοποθετούμε πλήρως συνδεδεμένο επίπεδο με αριθμό νευρώνων όσες και οι κλάσεις της ταξινόμησης. Συνήθως επιλέγεται για αυτό το επίπεδο η συνάρτηση Softmax.



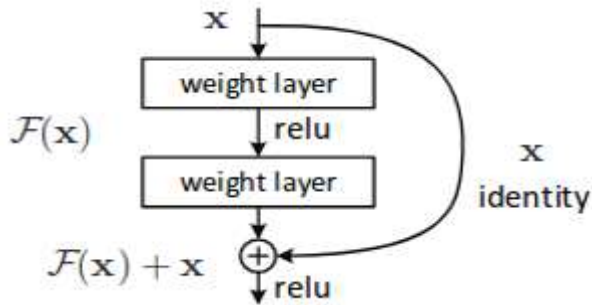
Εικόνα 3.8 : Παρουσίαση τριών διαφορετικών αρχιτεκτονικών συνελκτικών νευρωνικών δικτύων. VGGNet16, ResNet50 χωρίς και με υπολειπόμενες συνδέσεις.

Στην παραπάνω εικόνα συγκρίνουμε τον τρόπο σύνδεσης των επιπέδων στις αρχιτεκτονικές ResNet και VGGNet. Όπως φαίνεται στην περίπτωση του ResNet παρατηρούμε μεγαλύτερο αριθμό συνδέσεων και επιπέδων. Υπάρχει, δηλαδή μεγαλύτερο βάθος, ενώ στην περίπτωση του VGGNet παρατηρείται μεγαλύτερος αριθμός πλήρως συνδεδεμένων επιπέδων στο τέλος του δικτύου.

Η μεγαλύτερη όμως διαφορά στα δύο αυτά δίκτυα έγκυται στην διαφορετική συνδεσμολογία μεταξύ των επιπέδων, διότι όπως αναφέρθηκε και πριν, σε αρχιτεκτονικές τύπου ResNet υπάρχουν συνδέσεις τόσο μεταξύ διαδοχικών όσο και μη διαδοχικών επιπέδων. Όσον αφορά τη δεύτερη περίπτωση, θα ορίσουμε αυτόν τον τρόπο συνδεσμολογίας ως υπολειπόμενες συνδέσεις (residual connections).

### 3.1.2.3 Περιπτώσεις υπολειπόμενων συνδέσεων

Σε αυτό το σημείο, αναφέρουμε ότι υπάρχουν δύο τρόποι, που συναντώνται στην πράξη, σύμφωνα με τους οποίους μπορούμε να συνδέσουμε τα επίπεδα του δικτύου, κάνοντας χρήση υπολειπόμενων συνδέσεων. Στην πρώτη περίπτωση, όταν η έξοδος που παράγει το ένα επίπεδο ταυτίζεται σε διαστάσεις με την είσοδο του δεύτερου επιπέδου, τότε δεν απαιτείται κανένας μετασχηματισμός μεταξύ της εξόδου του πρώτου και της εισόδου του δεύτερου επιπέδου [8].



Εικόνα 3.9 : Παρουσίαση συνδεσμολογίας υπολειπόμενης σύνδεσης με απόσταση δύο ενδιάμεσων επιπέδων.

Θεωρούμε ότι η συνάρτηση ανάμεσα στα δύο μη διαδοχικά επίπεδα είναι η ταυτοτική, διότι δεν μετασχηματίζει την πληροφορία που δρομολογείται από το ένα επίπεδο στο άλλο.

$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \{W_i\}) + \mathbf{x}.$$

Το παραπάνω δεν ισχύει στην περίπτωση, όπου οι διαστάσεις μεταξύ των επιπέδων αλλάζουν. Τότε διακρίνουμε δύο διαφορετικές μεθόδους. Στην πρώτη από αυτές, όταν η έξοδος του πρώτου επιπέδου, δεν έχει ίσες διαστάσεις με την είσοδο του δεύτερου, πριν η πληροφορία, δηλαδή ο πίνακας, μεταβεί στην είσοδο του δεύτερου, εφαρμόζεται zero padding, με παράμετρο P, τέτοια ώστε οι διαστάσεις του δεύτερου επιπέδου και του πίνακα εξόδου του πρώτου επιπέδου να ταυτιστούν. Στη δεύτερη προκειμένου να ταυτιστούν οι διαστάσεις κάνουμε χρήση της παρακάτω συνάρτησης :

$$\mathbf{y} = \mathcal{F}(\mathbf{x}, \{W_i\}) + W_s \mathbf{x}.$$

Στην πρώτη περίπτωση δεν εισάγονται παραπάνω παράμετροι, ενώ στη δεύτερη εισάγονται με τη μορφή της μεταβλητής  $\mathbf{W}_s \mathbf{x}$ .

### 3.1.2.4 Απόδοση

Στην περίπτωση υλοποίησης υπολειπόμενων συνδέσεων, παρατηρείται ότι η αρχιτεκτονική ResNet αποδίδει καλύτερα, όσο αυξάνεται το βάθος του δικτύου, ενώ με απλή συνδεσμολογία, δίκτυα μικρότερου αριθμού επιπέδων, παρουσιάζουν μικρότερα σφάλματα

	plain	ResNet
<b>18 layers</b>	<b>27.94</b>	<b>27.88</b>
<b>34 layers</b>	<b>28.54</b>	<b>25.03</b>

Εικόνα 3.10: Πίνακας τελικών σφαλμάτων απλής αρχιτεκτονικής και αρχιτεκτονικής ResNet50 για διαφορετικό αριθμό επιπέδων.

### 3.1.2.5 Τύποι αρχιτεκτονικών ResNet

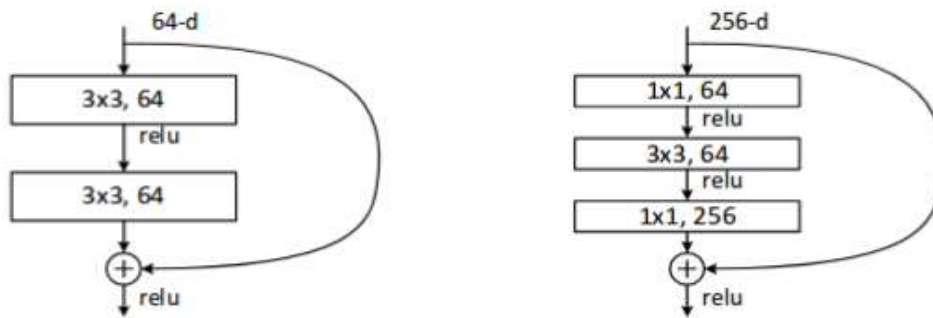
layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
		3×3 max pool, stride 2				
conv2_x	56×56	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 64 \\ 3 \times 3, 64 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28×28	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 128 \\ 3 \times 3, 128 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 8$
conv4_x	14×14	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 256 \\ 3 \times 3, 256 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 23$	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 36$
conv5_x	7×7	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 2$	$\begin{bmatrix} 3 \times 3, 512 \\ 3 \times 3, 512 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		$1.8 \times 10^9$	$3.6 \times 10^9$	$3.8 \times 10^9$	$7.6 \times 10^9$	$11.3 \times 10^9$

### ResNet Architectures

Εικόνα 3.11: Αρχιτεκτονικές τύπου ResNet50 διαφορετικών επιπέδων.

Στα πλαίσια της παρούσας διπλωματικής εργασίας, όπως αναφέρθηκε και σε προηγούμενη παράγραφο θα ασχοληθούμε με την απόδοση και τα χαρακτηριστικά του ResNet50, του οποίου τα χαρακτηριστικά υλοποίησης φαίνονται στον παραπάνω πίνακα. Όσον αφορά τις υπολοιπούμενες συνδέσεις, παρουσιάζονται δύο τρόποι με τους οποίους αυτές υλοποιούνται στην πράξη. Συγκεκριμένα, συνήθως επιλέγεται σύνδεση ανά δύο ή ανά τρία επίπεδα.





ResNet 2 layer and 3 layer Block

Εικόνα 3.12: Συνδεσμολογία υπολοιπόμενων συνδέσεων με δύο και τρία ενδιάμεσα συνελκτικά επίπεδα.

Τελικά αξίζει να αναφερθεί ότι οι αρχιτεκτονικές τύπου ResNet συγκλίνουν πιο γρήγορα και σε υψηλότερη ακρίβεια σε σχέση με αρχιτεκτονικές ίδιου βάθους, δίχως υπολοιπόμενες συνδέσεις δεύτερου τύπου.

Στην πλειονότητα τους, δηλαδή, οι υπολοιπόμενες συνδέσεις χρησιμοποιούνται ανάμεσα σε επίπεδα όπου η πληροφορία δε χρειάζεται αλλαγή διαστάσεων. Υπό το ίδιο σκεπτικό, συνδέσεις όπου απαιτούνται μετασχηματισμοί ανάμεσα στα επίπεδα, σε μεγάλο βαθμό παραλείπονται, κυρίως επειδή δεν επιφέρουν ιδιαίτερη αύξηση στην ακρίβεια.

### 3.1.3 Απλοποιημένη αρχιτεκτονική

Η τελευταία αρχιτεκτονική, η οποία θα υλοποιήσουμε, έχει μικρότερο βάθος σε σχέση με τις άλλες δύο, ενώ έχουμε μειώσει σε σχετικά μεγάλο βαθμό το πλήθος των φίλτρων στα συνελκτικά επίπεδα.

Θέλουμε να εξετάσουμε την επίδραση των πιο απλών φίλτρων και του βάθους της αρχιτεκτονικής στο συγκεκριμένο πρόβλημα, δηλαδή την ταξινόμηση μαγνητικών τομογραφιών σε υψηλού ή χαμηλού επιπέδου. Επειδή τα χαρακτηριστικά τα οποία εξετάζονται για την ταξινόμηση της κάθε εικόνας δεν είναι τόσο περίπλοκα όπως για παράδειγμα ένα ανθρώπινο πρόσωπο, επιλέγουμε να μειώσουμε σε αριθμό τα συνελκτικά επίπεδα. Και αυτό, διότι πολλά συνελκτικά επίπεδα στη σειρά τοποθετούνται σε προβλήματα που χρειαζόμαστε περίπλοκα φίλτρα, προκειμένου να αναγνωριστούν από κάθε εικόνα χαρακτηριστικά με μεγάλο βαθμό πολυπλοκότητας [9].

Η αρχιτεκτονική η οποία υλοποιήσαμε, λοιπόν, είναι η παρακάτω:

Πριν από κάθε διαδοχικό συνελκτικό επίπεδο πέρα από το αρχικό εφαρμόζεται zero padding, προκειμένου να διατηρήσουμε τον πίνακα πληροφορίας στις ίδιες διαστάσεις. Ακόμη μετά από κάθε συνελκτικό επίπεδο εισάγεται συνάρτηση μη γραμμικότητας, η οποία εισάγει μη γραμμικότητα στα δεδομένα.

Αρχικά, έχουμε δύο συνελκτικά επίπεδα, τα οποία έχουν το καθένα 64 φίλτρα διαστάσεων 3x3. Στη συνέχεια ακολουθεί επίπεδο pooling, το οποίο μειώνει το πλήθος των δεδομένων

στο 25%, εφαρμόζοντας τη συνάρτηση  $\max$ , με φίλτρο διαστάσεων  $2 \times 2$ . Τα τρία αυτά επίπεδα στη συνέχεια επαναλαμβάνονται με τη διαφορά ότι το πλήθος των φίλτρων στα δύο συνελικτικά επίπεδα, δηλαδή το τρίτο και το τέταρτο αυξάνονται από 64 σε 128. Οι διαστάσεις των φίλτρων στα συνελικτικά αυτά επίπεδα παραμένουν ίδια, ενώ πάλι ακολουθεί επίπεδο pooling, με χαρακτηριστικά ίδια ακριβώς με αυτά του προηγούμενου αντίτιχου επιπέδου.

Τέλος, η αρχιτεκτονική καταλήγει σε τρία πλήρως συνδεδεμένα επίπεδα, εκ των οποίων το τρίτο έχει δύο νευρώνες. Καθένας από αυτούς όταν ενεργοποιείται δηλώνει σε ποιά κατηγορία ανήκει η εικόνα. Το πρώτο πλήρες συνδεδεμένο επίπεδο έχει 256 νευρώνες σε αριθμό και το δεύτερο 32. Το τελευταίο επίπεδο χρησιμοποιεί τη συνάρτηση sigmoid, η οποία και εμφανίζει υψηλότερη απόδοση στα προβλήματα δυαδικής ταξινόμησης, όπως είναι το δικό μας.

## 3.2 Προεπεξεργασία Δεδομένων

Εκτός από την επιλογή μοντέλων συνελικτικών νευρωνικών δικτύων, παραμέτρων μάθησης, αλλά και δεδομένων εκπαίδευσης, προκειμένου να λάβουμε σαν αποτέλεσμα ένα σύστημα με υψηλή απόδοση, εξαιρετική σημασία έχει και η επεξεργασία των δεδομένων εκπαίδευσης πριν αυτά εισαχθούν σαν είσοδοι στο σύστημα.

Δηλαδή, η παραπάνω διαδικασία, η οποία ονομάζεται προεπεξεργασία δεδομένων, μετασχηματίζει τα δεδομένα με σκοπό να τα μετατρέψει σε μία μορφή η οποία είναι πιο εύκολα επεξεργάσιμη από το σύστημα. Υπάρχουν πολλές διαφορετικές μέθοδοι, οι οποίες αυξάνουν την απόδοση των δικτύων και κάθεμία από αυτές μπορεί να εφαρμοστεί ανεξάρτητα στα δεδομένα.

Στα πλαίσια της παρούσας διπλωματικής εργασία, θα ασχοληθούμε με δύο διαφορετικές μεθόδους εξομάλυνσης-κανονικοποίησης των δεδομένων (data normalization), με τη μέθοδο ανισοτροπικής διάχυσης (anisotropic diffusion), τη μέθοδο διόρθωσης πεδίου πόλωσης (bias field correction) και τέλος την αφαίρεση εικόνων με πολύ μικρή περιεκτικότητα σε χαρακτηριστικά.

Σε επόμενο κεφάλαιο έχουμε ασχοληθεί με τη σύγκριση των παραπάνω μεθόδων τόσο μεταξύ τους, όσο και με δεδομένα, τα οποία δεν έχουν υποστεί καμία μέθοδο προεπεξεργασίας, με σκοπό να τονιστεί η σημαντικότητα των παραπάνω τεχνικών στην αύξηση της απόδοσης του συστήματος.

### 3.2.1 Κανονικοποίηση - Εξομάλυνση (Normalization)

[49] Όσον αφορά το πεδίο προεπεξεργασίας της εικόνας, η εξομάλυνση ή κανονικοποίηση (normalization), είναι η διαδικασία μέσω της οποίας μετασχηματίζεται το εύρος των τιμών

των πίξελ. Εφαρμογές εξομάλυνσης εικόνας συμπεριλαμβάνουν, για παράδειγμα, φωτογραφίες, οι οποίες έχουν υψηλή αντίθεση και είναι δυσδιάκριτες λόγω έντονης λάμψης.

Κάθε ασπρόμαυρη εικόνα ορίζεται σαν ένας δυσδιάστατος πίνακας, ο οποίος σε κάθε πίξελ λαμβάνει μία τιμή από το 0(μαύρο χρώμα)μέχρι το 255(λευκό χρώμα). Το σύνολο των ενδιαμέσων τιμών ανήκει σε τιμές, οι οποίες κωδικοποιούν 253 ενδιαμέσες αποχρώσεις του γκρι.

Προφανώς, όσο πιο κοντά είναι η τιμή στο 255 τόσο πιο φωτεινό(πιο κοντά στο λευκό) απεικονίζεται το πίξελ, ενώ όσο πιο κοντά στο 0 είναι η τιμή, τόσο πιο κοντά στο μαύρο χρώμα απεικονίζεται το πίξελ. Κάθε στοιχείο του δυσδιάστατου πίνακα, που αναφέρεται και σε ένα συγκεκριμένο και μοναδικό πίξελ της εικόνας, έχει, αρχικά, όπως προαναφέρθηκε, μία τιμή ανάμεσα στους αριθμούς 0 και 255. Μέσω της εξομάλυνσης κάθε στοιχείο αποκτά διαφορετική τιμή.

Ο σκοπός της δυναμικής αυτής αλλαγής τιμών σε πολλές εφαρμογές είναι συνήθως για να μετασχηματίσουμε την εικόνα ή οποιουδήποτε είδους σήμα, σε μία μορφή η οποία είναι πιο ομαλή και οικεία στις ανθρώπινες αισθήσεις. Από αυτό προκύπτει και ο όρος εξομάλυνση.

Πολλές φορές το κίνητρο πίσω από την προεπεξεργασία, όσον αφορά ένα σύνολο δεδομένων, είναι να μειωθούν οι έντονες διαφορές ανάμεσα στα στοιχεία αυτού του συνόλου, ή διαφορετικά, να μειώσουμε τη διακύμανση μεταξύ των τιμών των διαφορετικών στοιχείων.

Σκοπός, λοιπόν, είναι η εξομάλυνση των στοιχείων, προκειμένου να αποφευχθεί κόπωση και ασάφεια από πλευράς του δέκτη της πληροφορίας. Για παράδειγμα, σε μία εφημερίδα, οι συντάκτες τις θα προσπαθήσουν να απεικονίσουν όλες τις εικόνες ενός τεύχους με όσο το δυνατόν πιο «κοντινές» αποχρώσεις του χρώματος γκρι, αποσκοπώντας σε πιο ομοιόμορφο αποτέλεσμα.

Όπως προαναφέρθηκε, λοιπόν, στόχος της εξομάλυνσης είναι ο μετασχηματισμός μίας  $n$ -διάστατης ασπρόμαυρης εικόνας. Ορίζουμε σαν  $I$  την απεικόνιση από τις θέσεις των στοιχείων της εικόνας σε ένα σύνολο εντάσεων φωτεινότητας που αναφέρονται στα πίξελ.

Επεξεργάζοντας τα πίξελ της εκάστοτε εικόνας, σύμφωνα με τη μέθοδο της εξομάλυνσης εξάγεται μία καινούρια απεικόνιση  $I_N$ , η οποία πρακτικά αλλάζει το εύρος των τιμών φωτεινότητας στα πίξελ.

$$I : \{X \subseteq \mathbb{R}^n\} \rightarrow \{\text{Min}, \dots, \text{Max}\}$$

$$I_N : \{X \subseteq \mathbb{R}^n\} \rightarrow \{\text{newMin}, \dots, \text{newMax}\}$$

Min, Max είναι αντίστοιχα οι μεταβλητές, οι οποίες αναφέρονται στην ελάχιστη τιμή φωτεινότητας πάνω στον σύνολο των πίξελ όλων των εικόνων πριν το μετασχηματισμό, ενώ newMin, newMax είναι οι αντίστοιχες τιμές ελαχίστου και μεγίστου μετά από το μετασχηματισμό.

Η γραμμική εξομάλυνση σε μία ασπρόμαυρη εικόνα δίνεται από την παρακάτω σχέση:

$$I_N = (I - \text{Min}) \frac{\text{newMax} - \text{newMin}}{\text{Max} - \text{Min}} + \text{newMin}$$

Για παράδειγμα, εάν η ελάχιστη και μέγιστη τιμή στην ένταση της φωτεινότητας είναι αντίστοιχα 50 και 180 και το επιθυμητό εύρος τιμών είναι 0 ως 255, τότε αφαιρείται 50 από την τιμή καθενός πίξελ μεταφέροντας τις εντάσεις στο εύρος 0 με 130. Τελικά, η τιμή πολλαπλασιάζεται με 255/130 και τελικά λαμβάνουμε τιμές από 0 ως 255.

Παρόλ'αυτά, οι μετασχηματισμοί κατά την εξομάλυνση δεν είναι απαραίτητο να είναι γραμμικοί. Το παραπάνω συμβαίνει σε περιπτώσεις, όπου η σχέση μεταξύ της απεικόνισης  $I$  και  $I_N$  δεν είναι γραμμική.

Ένα παράδειγμα αυτού είναι όταν κατά την εξομάλυνση ακολουθείται η σιγμοειδής συνάρτηση. Αυτό έχει σαν αποτέλεσμα να εφαρμόζεται σαν μετασχηματισμός η παρακάτω σχέση:

$$I_N = (\text{newMax} - \text{newMin}) \frac{1}{1 + e^{-\frac{I-\beta}{\alpha}}} + \text{newMin}$$

Η μεταβλητή  $\alpha$  στην παραπάνω σχέση εκφράζει το πλάτος του εύρους φωτεινότητας, ενώ η μεταβλητή  $\beta$  την ένταση γύρω από την οποία επικεντρώνεται το εύρος.

Εμείς, στα πλαίσια της παρούσας διπλωματικής εργασίας, θα εφαρμόσουμε στο σύνολο των δεδομένων μας δύο μεθόδους εξομάλυνσης

### 3.2.1.1 Εξομάλυνση τοπικού μέσου όρου – Local Mean Normalization

Σύμφωνα με την πρώτη μέθοδο εξομάλυνσης, μηδενίζουμε το μέσο όρο σε κάθε δείγμα εισόδου και στη συνέχεια αλλάζουμε την κλίμακα στην τιμή του κάθε πίξελ πολλαπλασιάζοντας με ένα σταθερό όρο, ο οποίος είναι μικρότερος του μηδενός.

Στόχος και των δύο βημάτων είναι να εισάγουμε, τελικά, στο δίκτυο, παραμέτρους με μεγαλύτερο βαθμό ευκολίας στην επεξεργασία τους, αλλά παράλληλα που συντελούν στη μείωση της συνάρτησης σφάλματος.

Συγκεκριμένα, κατά την εκπαίδευση, η αφαίρεση του μέσου όρου των πίξελ της εικόνας, βοηθά στο να επικεντρωθούν οι τιμές των πίξελ γύρω από μία σταθερή τιμή. Ακόμα, για τον ίδιο λόγο, συνήθως διαιρούμε και με σταθερές τιμές κάθε τιμή πίξελ.

Ο λόγος που συμβαίνουν τα παραπάνω είναι διότι κατά την εκπαίδευση πολλαπλασιάζουμε τις παραμέτρους εισόδου με βάρη και προσθέτουμε με άλλες σταθερές τιμές, όπως την πόλωση (bias). Στη συνέχεια, τα αποτελέσματα των παραπάνω υπολογισμών διαδίδονται ανάστροφα στο δίκτυο, μέσω της διαδικασίας της ανάστροφής διάδοσης, σε συνδυασμό με τις παραγώγους, έτσι ώστε να εκπαιδευτεί το μοντέλο. Θα θέλαμε, οπότε, για κάθε ξεχωριστό χαρακτηριστικό, που ζητάμε να αναγνωριστεί από το σύστημα, να είχαμε ένα

σταθερό εύρος τιμών για να μην λαμβάνουν ανεξέλεγκτα μεγάλες τιμές οι παράγωγοι (και ακόμα για να έχουμε μόνο ένα γενικό πολλαπλασιαστική βαθμού μάθησης).

Ένας επιπρόσθετος λόγος, για τον οποίο επιθυμούμε τη διαδικασία της εξομάλυνσης είναι ότι σε γενική περίπτωση, τα συνελκτικά νευρωνικά δίκτυα μοιράζονται έναν πολύ μεγάλο αριθμό παραμέτρων. Οπότε, εάν δεν αλλάζαμε την κλίμακα στις τιμές της εισόδου, με τέτοιο τρόπο ώστε να προκύπτει εύρος τιμών μικρό σχετικά ανάμεσα σε ξεχωριστά χαρακτηριστικά, τότε η διαδικασία αυτή θα ήταν αρκετά δύσκολη. Και αυτό, διότι τα διανύσματα βαρών σε ένα τμήμα τους θα ήταν πολύ μεγάλα και σε άλλο πολύ μικρά, με αποτέλεσμα να μην υπάρχει αντικειμενικότητα στην εξαγωγή χαρακτηριστικών, λόγω του μεγάλου εύρους.

Οπότε, κατά την εφαρμογή της πρώτης μεθόδου εξαγάμε από κάθε σύνολο εικόνων εισόδου την τιμή εκείνη, η οποία ανήκει στο πίξελ με την ισχυρότερη ένταση. Η τιμή αυτή ήταν 255. Στη συνέχεια και για κάθε εικόνα, φέραμε το μέσο όρο των τιμών των πίξελ της στο 0, για τους παραπάνω λόγους.

Συγκεκριμένα αφαιρέσαμε από κάθε εικόνα τον παραπάνω μέσο όρο των πίξελ και στη συνέχεια διαιρέσαμε με τη μέγιστη τιμή 255. Σαν αποτέλεσμα το σύνολο τιμών των πίξελ λαμβάνει τιμές στο σύνολο  $[-\mu/255, +\mu/255]$ , το οποίο είναι υποσύνολο του συνόλου  $[-1,1]$ .

### 3.2.1.2 Κλίμακα μεγίστου-ελαχίστου (min-max scaling) ή εξομάλυνση ολικού μέσου όρου-Global Mean Normalization

Παρόμοια με την παραπάνω διαδικασία, είναι και η μέθοδος, που θα περιγράψουμε παρακάτω, σύμφωνα με την οποία αφαιρέσαμε από κάθε τιμή πίξελ, το μέσο όρο των μέσων όρων όλων των πίξελ, ενώ στο τέλος διαιρέσαμε τα αποτελέσματα με το μέσο όρο του άθροισματος μεγίστων-ελαχίστων τιμών πίξελ από το σύνολο των εικόνων.

Συγκεκριμένα, ακολουθήσαμε την εξής διαδικασία. Όπως αναφέρθηκε και παραπάνω κάθε ασπρόμαυρη εικόνα θεωρείται ένας δυσδιάστατος πίνακας, ο οποίος σε κάθε θέση του έχει μία τιμή που υποδηλώνει την ένταση του κάθε πίξελ. Από κάθε εικόνα λάβαμε δύο τιμές. Πρώτα άθροισαμε όλες τις τιμές και στη συνέχεια διαιρέσαμε με το πλήθος τους. Με αυτόν τον τρόπο λάβαμε από κάθε εικόνα το μέσο όρο  $\mu$ . Δεύτερον, λάβαμε από κάθε εικόνα τη μέγιστη τιμή πίξελ  $\max$  και την ελάχιστη τιμή πίξελ  $\min$ . Σε όλες τις εικόνες η δεύτερη τιμή ήταν ίση με μηδέν, οπότε η διαφορά αυτή ταυτίζεται με την αριθμητική τιμή  $\max$ . Στη συνέχεια, θελήσαμε αντί να μηδενίσουμε το μέσο όρο για κάθε εικόνα ξεχωριστά όπως πριν και στη συνέχεια να αλλάξουμε την κλίμακα, να βρούμε τις δύο παρακάτω τιμές.

Έστω  $N$  το πλήθος των εικόνων. Διαιρέσαμε το άθροισμα των μέσων όρων όλων των εικόνων με το πλήθος των εικόνων και λάβαμε το μέσο όρο των μέσων όρων  $\mu_{\mu}$ . Επίσης, ακολουθήσαμε την ίδια διαδικασία με τη διαφορά μεγίστων ελαχίστων τιμών. Υπολογίσαμε το άθροισμά τους και στη συνέχεια διαιρέσαμε το αποτέλεσμα αυτό με το πλήθος  $N$ . Σαν αποτέλεσμα προέκυψε ο μέσος όρος των διαφορών μεγίστων-ελαχίστων πάνω στο σύνολο των εικόνων  $dif_{\mu}$ .

Πιο συγκεκριμένα  $\mu_{\mu} = (\sum_1^N \mu) / N$  και  $dif_{\mu} = (\sum_1^N dif) / N$

Η διαφορά ανάμεσα σε αυτή τη μέθοδο και την παραπάνω είναι ότι εδώ δε μηδενίζουμε το μέσο όρο πάνω σε κάθε εικόνα ξεχωριστά και μετά αλλάζουμε την κλίμακα με το συνολικό

μέγιστο, αλλά μηδενίζουμε το μέσο όρο όλων των εικόνων συνολικά και στη συνέχεια αλλάζουμε την κλίμακα λαμβάνοντας μία τιμή μεγίστου ( και ταυτόχρονα διαφοράς μεγίστου ελαχίστου) από όλες τις εικόνες.

Αυτό έχει σαν αποτέλεσμα πιο συνολική εξομάλυνση των δεδομένων, διότι μέσω της διαδικασίας αυτής τα επεξεργαζόμαστε συνολικά και όχι σαν μοναδικές περιπτώσεις. Οπότε το δίκτυο κατά την επεξεργασία των παραμέτρων θα αναγνωρίσει «κοντινές» τιμές για αντίστοιχα χαρακτηριστικά πανω στο σύνολο των εικόνων, διευκολύνοντας έτσι και τη σύγκλιση και τελικά αυξάνοντας την απόδοση.

Ο μετασχηματισμός αυτή τη φορά ακολουθεί τη σχέση:

$$I_n = (I - \mu_n) / \text{dif}_n$$

Τελικά, αξίζει να αναφερθεί ότι κατά οποιαδήποτε μέθοδο εξομάλυνσης λαμβάνουμε τιμές από τα πίξελ της κάθε εικόνας. Αυτό έχει σαν αποτέλεσμα, εάν θέλουμε να εφαρμόσουμε πρώτα ένα διαφορετικό μετασχηματισμό, έστω  $M$ , πριν από τη μέθοδο της εξομάλυνσης, η τελευταία να πρέπει να εφαρμοστεί στις καινούριες τιμές που θα προκύψουν από τον  $M$ . Για παράδειγμα, όπως θα δούμε και παρακάτω, πρώτα θα εφαρμόσουμε τη μέθοδο της ανισοτροπικής διάχυσης, η οποία θα αλλάξει τις τιμές των πίξελ σε κάθε εικόνα και μετά θα εφαρμόσουμε κάθεμία από τις παραπάνω μεθόδους στις ήδη μετασχηματισμένες εικόνες.

### 3.2.2 Ανισοτροπική Διάχυση (Anisotropic diffusion)

[50] Στην επεξεργασία εικόνας (image processing) και την όραση υπολογιστών (computer vision), η ανισοτροπική διάχυση ή Perona-Malik διάχυση (Perona-Malik diffusion), είναι μία τεχνική, η οποία αποσκοπεί στη μείωση του θορύβου, δηλαδή της τυχαίας παραλλαγής του χρώματος ή της φωτεινότητας της εικόνας, δίχως να αφαιρούνται σημαντικά κομμάτια του περιεχομένου της. Συνήθως γραμμές, ακμές ή και άλλες λεπτομέρειες της εικόνας, στοιχεία, δηλαδή, τα οποία είναι σημαντικά για την ερμηνεία της. Η ανισοτροπική διάχυση μοιάζει τη διαδικασία κατά την οποία μία εικόνα παράγει μία παραμετρική οικογένεια από εικόνες, που η επόμενη σε σχέση με την προηγούμενη παρουσιάζει μεγαλύτερο βαθμό θόλωσης (blur) και βασίζεται στη διαδικασία διάχυσης (diffusion process).

Κάθεμία από τα παραπάνω αποτελέσματα-εικόνες, που ανήκουν στην παραμετρική οικογένεια, είναι το αποτέλεσμα μία συνέλιξης ανάμεσα στην εικόνα και ένα δυσδιάστατο Γκαουσιανό φίλτρο, όπου το πλάτος του φίλτρου αυξάνεται ανάλογα με την δοθείσα παράμετρο. Η διαδικασία διάχυσης είναι ένας γραμμικός μετασχηματισμός και ανεξάρτητος χωρικών μεταβλητών. Το τελευταίο σημαίνει ότι σε οποιοδήποτε σύνολο πίξελ της εικόνας, ανεξάρτητα από το που βρίσκεται αυτό σε σχέση τα υπόλοιπα, θα εφαρμοστεί με τον ίδιο τρόπο. Η ανισοτροπική διάχυση είναι μία γενίκευση της διαδικασίας διάχυσης. Παράγει μία παραμετρική οικογένεια εικόνων, αλλά κάθε εικόνα-αποτέλεσμα της οικογένειας αυτής, είναι ένας συνδυασμός ανάμεσα στην αρχική εικόνα και ένα φίλτρο, το οποίο εξαρτάται από το τοπικό περιεχόμενο της εικόνας στο σημείο εφαρμογής. Σαν αποτέλεσμα, η ανισοτροπική διάχυση είναι μη-γραμμικός μετασχηματισμός και εξαρτάται πλήρως από το σημείο της εικόνας, στο οποίο εφαρμόζεται (space-variant).

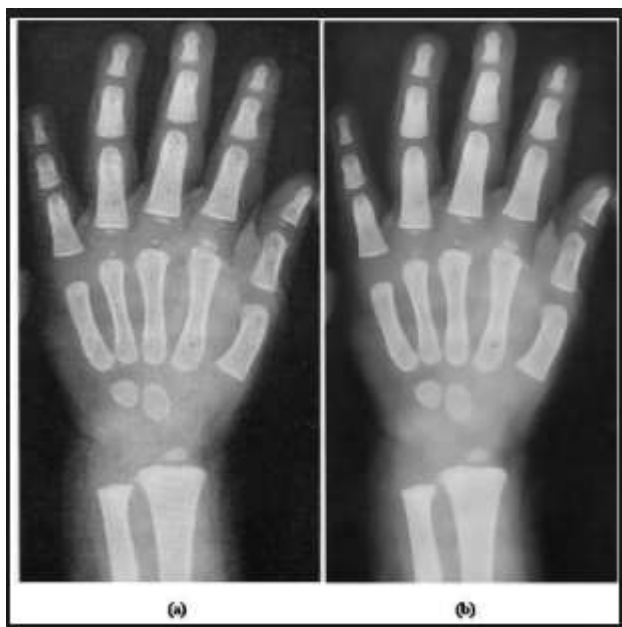
Στην αρχική διατύπωσή του, η οποία παρουσιάστηκε από τους Perona και Malik, το 1987, το χωρικά-εξαρτόμενο (space-variant) φίλτρο, ήταν ισοτροπικό αλλά εξαρτόταν από το περιεχόμενο της εικόνας, διότι παρουσίαζε μία συνάρτηση  $\delta$  του Dirac, σε ακμές ή στοιχεία,

τα οποία έπρεπε να διατηρηθούν από την εικόνα, σε όλα τα πιθανά αποτελέσματα των εικόνων που θα παρήγαγε το φίλτρο με την αρχική εικόνα. Η μέθοδος ορίστηκε σαν ανιστροπική διάχυση από τους Perona και Malik, παρότι το φίλτρο που εφαρμοζόταν τοπικά στην εικόνα ήταν ιστροπικό.

Μία πιο γενική διατύπωση επιτρέπει στο φίλτρο αυτό να είναι όντως ανιστροπικό κοντά σε γραμμικά στοιχεία, όπως είναι ακμές ή ευθείες. Έχει έναν προσανατολισμό, ο οποίος εξαρτάται από το κάθε συγκεκριμένο στοιχείο. Εμφανίζει μεγάλο βαθμό επιμήκυνσης σε αυτόν και μικρότερο βαθμό σε «κοντινά» σημεία, τα οποία δεν ανήκουν σε αυτόν. Αυτό έχει σαν αποτέλεσμα, οι εικόνες που παράγονται να διατηρούν τα γραμμικά τους στοιχεία, ενώ παράλληλα, αυξάνεται ο δείκτης ευκρίνειας σε όλη την έκταση τους.

Αυτές οι περιπτώσεις μπορούν να χαρακτηριστούν γενίκευση της συνάρτησης διάχυσης, όπου ο συντελεστής διάχυσης, αντί να είναι μία σταθερά είναι μία συνάρτηση της θέσης στην εικόνα και λαμβάνει μία τιμή πίνακα.

Παρότι το αποτέλεσμα της οικογένειας εικόνων μπορεί να περιγραφεί σαν συνδυασμός ανάμεσα στην αρχική εικόνα και χωρικά-εξαρτώμενα φίλτρα, το τοπικά εφαρμοζόμενο φίλτρο και ο συνδυασμός με την εικόνα δε χρειάζεται να οριστούν επακριβώς στην πράξη. Η ανιστροπική διάχυση, συνήθως, υλοποιείται με μία προσέγγιση της γενικευμένης συνάρτησης διάχυσης. Κάθε επόμενη εικόνα, σε αυτή την οικογένεια, προκύπτει με εφαρμογή της συνάρτησης πάνω στην προηγούμενη εικόνα. Σαν αποτέλεσμα, είναι μία επαναληπτική μέθοδος, όπου ένα μικρό σχετικά σύνολο υπολογισμών, χρησιμοποιούνται για να υπολογιστεί κάθε επόμενη εικόνα στην οικογένεια, ενώ η διαδικασία συνεχίζεται μέχρι να επιτευχθεί ένας συγκεκριμένος βαθμός ευκρίνειας.



Εικόνα 3.13: Μαγνητικές τομογραφίες χωρίς (αριστερά) και με (δεξιά) μέθοδο προεπεξεργασίας ανιστροπικής διάχυσης.

Η εικόνα στα αριστερά δεν έχει υποστεί τη μέθοδο της ανισοτροπικής διάχυσης, σε αντίθεση με τη δεξιά, η οποία παρουσιάζει υψηλότερη ευκρίνεια.

### Τυπικός ορισμός

Τυπικά έστω  $\Omega \subset \mathbb{R}^2$ , ένα υποσύνολο του πεδίου και έστω ακόμα μία απεικόνιση

$I(\cdot, t): \Omega \rightarrow \mathbb{R}^2$ , της οικογένειας από ασπρόμαυρες εικόνες. Σε αυτή την περίπτωση η ανισοτροπική διάχυση ορίζεται σαν :

$$\frac{\partial I}{\partial t} = \operatorname{div}(c(x, y, t)\nabla I) = \nabla c \cdot \nabla I + c(x, y, t)\Delta I$$

Στην παραπάνω εξίσωση,  $\Delta$  είναι το σύμβολο της Λαπλασιανής, δηλαδή  $\nabla^2$ ,

$\nabla$  είναι ο τελεστής που υποδηλώνει την παράγωγο προς κάθε κατεύθυνση,  $\operatorname{div}(\dots)$ , είναι ο τελεστής της τυπικής απόκλισης, και  $c(\mathbf{x}, \mathbf{y}, \mathbf{t})$ , είναι ο συντελεστής διάχυσης.

Η τελευταία συνάρτηση  $c(\mathbf{x}, \mathbf{y}, \mathbf{t})$ , ελέγχει το βαθμό διάχυσης και συνήθως επιλέγεται συνδυαστικά με την παράγωγο που δίνει η εικόνα, προκειμένου να διαφυλαχτούν κατά τη διαδικασία βασικά στοιχεία αυτής, όπως είναι για παράδειγμα ακμές, καμπύλες κτλ. Οι δύο συναρτήσεις που προτάθηκαν, αρχικά, από τους Perona και Malik, μαζί με την παρουσίαση της συνολικής μεθόδου το 1990, σαν συντελεστές διάχυσης είναι οι εξής:

$$c(\|\nabla I\|) = e^{-(\|\nabla I\|/K)^2}$$

$$c(\|\nabla I\|) = \frac{1}{1 + \left(\frac{\|\nabla I\|}{K}\right)^2}$$

Η σταθερά  $K$ , ελέγχει την ευαισθησία της μεθόδου γύρω από τις ακμές και συνήθως επιλέγεται είτε μέσω δοκιμών είτε σε μία συνάρτηση θορύβου μέσα στην εικόνα.

Έστω  $\mathbf{M}$ , η πολλαπλότητα των εικόνων, στις οποίες έχει αυξηθεί η ευκρίνεια. Τότε, οι συναρτήσεις διάχυσης, που παρουσιάστηκαν παραπάνω, μπορούν να ερμηνευθούν σε συναρτήσεις εύρεσης ελαχίστου (gradient descent), για την ελαχιστοποίηση της συνάρτησης  $E: \mathbf{M} \rightarrow \mathbb{R}$ , η οποία ορίζεται σαν

$$E[I] = \frac{1}{2} \int_{\Omega} g(\|\nabla I(x)\|^2) dx$$

Η συνάρτηση  $g: \mathbb{R} \rightarrow \mathbb{R}$ , είναι μία συνάρτηση πραγματικών τιμών, η οποία θα δούμε ότι είναι στενά συνδεδεμένη στη σταθερά διάχυσης. Ακόμα, για κάθε επ'άπειρον διαφορίσιμη συνάρτηση δοκιμής  $h$  με την ιδιότητα να λαμβάνει μόνο την τιμή μηδέν εκτός ενός συμπαγούς συνόλου (compact support), ισχύει:



$$\begin{aligned}
\frac{d}{dt} \Big|_{t=0} E[I + th] &= \frac{d}{dt} \Big|_{t=0} \frac{1}{2} \int_{\Omega} g(\|\nabla(I + th)(x)\|^2) dx \\
&= \int_{\Omega} g'(\|\nabla I(x)\|^2) \nabla I \cdot \nabla h dx \\
&= - \int_{\Omega} \operatorname{div}(g'(\|\nabla I(x)\|^2) \nabla I) h dx
\end{aligned}$$

,όπου η τελευταία γραμμή ακολουθεί από πολυδιάστατες ολοκληρώσεις σε τμήματα.παράμετρος  $\nabla E_I$ , δηλώνει την κλίση της  $E$ , βασιζόμενη στο εσωτερικό γινόμενο  $L^2(\Omega, \mathbf{R})$ , υπολογιζόμενο στο  $I$ . Οπότε, έχουμε:

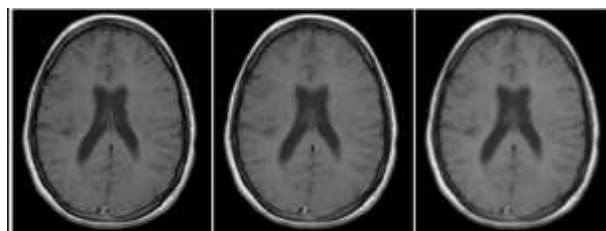
$$\nabla E_I = -\operatorname{div}(g'(\|\nabla I(x)\|^2) \nabla I)$$

Σαν αποτέλεσμα, οι συναρτήσεις που υπολογίζουν την ελάχιστη τιμή (gradient descent), της  $E$ , δίνονται από την παρακάτω σχέση:

$$\frac{\partial I}{\partial t} = -\nabla E_I = \operatorname{div}(g'(\|\nabla I(x)\|^2) \nabla I)$$

Οπότε θεωρώντας ότι  $c = g'$ , λαμβάνουμε τις συναρτήσεις ανισοτροπικής διάχυσης.

Η ανισοτροπική διάχυση, τελικά, έχει την ιδιότητα να αφαιρεί το θόρυβο από τις ψηφιακές εικόνες, δίχως να θολώνει τις ακμές. Με μία σταθερή τιμή σε σταθερά διάχυσης, οι συναρτήσεις της ανισοτροπικής διάχυσης μειώνουν το θόρυβο, αλλά δε διακρίνουν τις ακμές. Στην περίπτωση που για σταθερά διάχυσης επιλεγεί μία συνάρτηση αναζήτησης ακμών, όπως στην περίπτωση μεθόδου Perona-Malik, οι συναρτήσεις που προκύπτουν ενθαρρύνουν τη διάχυση(αυξάνουν την ευκρίνεια) μέσα σε όμοιες περιοχές, ενώ την απαγορεύουν στην περίπτωση εμφάνισης ισχυρών ακμών. Αυτό έχει σαν αποτέλεσμα να διατηρούνται οι ακμές, ενώ ταυτόχρονα να μειώνεται ο θόρυβος. Παράλληλα με τη μείωση του θορύβου, η ανισοτροπική διάχυση μπορεί να χρησιμοποιηθεί σε αλγόριθμους εύρεσης ακμών, με την εκτέλεση του αλγόριθμου και εισάγοντας στη θέση της σταθεράς διάχυσης μία συνάρτηση, η οποία ανιχνεύει ακμές. Εκτελώντας τον αλγόριθμο αυτό για ένα συγκεκριμένο αριθμό επαναλήψεων και εισάγοντας τις παραπάνω παραμέτρους, λαμβάνουμε σαν αποτέλεσμα μία εικόνα, στην οποία οι ακμές της διαχωρίζουν τα επιμέρους τμήματά της, τα οποία με τη σειρά τους εμφανίζουν πολύ «κοτινές» τιμές φωτεινότητας. Σαν αποτέλεσμα, κάθε επιμέρους τμήμα διαχωρίζεται με δύο τρόπους. Πρώτον μέσω των ακμών και δεύτερων μέσω της διαφορετικής έντασης σε φωτεινότητα.



Εικόνα 3.14: Διαδοχικές περιπτώσεις εφαρμογής της μεθόδου ανισοτροπικής διάχυσης πάνω στην ίδια μαγνητική τομογραφία.

Στα πλαίσια της παρούσας διπλωματικής εργασίας θα εφαρμόσουμε τη μέθοδο αυτή στις εικόνες του συνόλου δεδομένων πριν την εκπαίδευση, αποσκοπώντας στον εντονότερο διαχωρισμό των χαρακτηριστικών, προκειμένου να διευκολύνουμε την επεξεργασία. Συγκεκριμένα, διαχωρίζοντας τον κάθε ιστό στον εγκέφαλο και κάνοντας εντείνοντας τη διαφορά μεταξύ του ιστού αυτού και των γλοιωμάτων, ενισχύουμε την ικανότητα του δικτύου στην ταξινόμηση.

### 3.2.3 Διόρθωση πεδίου πόλωσης (N4 bias field correction)

Το σήμα πεδίου πόλωσης (bias signal), είναι ένα χαμηλής συχνότητας, ομαλό σήμα, το οποίο διαφθείρει την ποιότητα των εικόνων που προκύπτουν από μαγνητικούς τομογράφους. Οι τελευταίοι, παρ'όλα αυτά, εάν είναι σύγχρονοι, μπορεί να υλοποιούν μεθόδους, που μειώνουν την επίδραση του παραπάνω σήματος, με αποτέλεσμα το πρόβλημα αυτό να παρουσιάζεται κυρίως σε αντίστοιχα παλιότερα μηχανήματα. Σε αυτές τις περιπτώσεις, όμως, όταν δηλαδή το σήμα ούτε μειώνεται σε ικανοποιητικό βαθμό, αλλά ούτε και αφαιρείται πλήρως, το αποτέλεσμα των αλγορίθμων ταξινόμησης, ανάλυσης υφής (texture analysis) ή και τμηματοποίησης ασπρόμαυρων εικόνων δεν είναι ικανοποιητικό. Λύση για το παραπάνω αποτελούν οι μέθοδοι διόρθωσης του σήματος πόλωσης στις εικόνες πριν εφαρμοστούν οι αντίστοιχοι αλγόριθμοι ή η εισαγωγή σε αυτούς μεθόδων που διορθώνουν το σήμα [51].

Πιο συγκεκριμένα, το σήμα πεδίου πόλωσης διαφθείρει την ποιότητα των εικόνων, λόγω της ανομοιογένειας των μαγνητικών πεδίων στο εσωτερικό του μαγνητικού τομογράφου. Έχει την ιδιότητα να θολώνει τις εικόνες, με αποτέλεσμα να μειώνει τα περιεχόμενα υψηλής συχνότητας όπως είναι ακμές, περιγράμματα και εντάσεις των πίκσελ, με αποτέλεσμα να εντείνεται η ανομοιογένεια και η διακύμανση ανάμεσα στα πίκσελ του ίδιου ιστού. Όταν το φαινόμενο αυτό δεν είναι έντονο, τότε δεν παρουσιάζεται μείωση της απόδοσης των παραπάνω αλγορίθμων, αλλά σε αντίθετες περιπτώσεις, δηλαδή όταν το σήμα είναι ισχυρό, σε αλγόριθμους σαν τους παραπάνω που βασίζονται στη χωρική αμεταβλητότητα (spatial invariance), η απόδοση ελαχιστοποιείται κατά πολύ.

Προς αντιμετώπιση του παραπάνω, έχει αναπτυχθεί ένας μεγάλος αριθμός μεθόδων προεπεξεργασίας δεδομένων, οι οποίοι αποσκοπούν στη διόρθωση του παραπάνω σήματος στις εικόνες. Στα πλαίσια της παρούσας διπλωματικής εργασίας θα ασχοληθούμε με τη μέθοδο διόρθωσης πεδίου πόλωσης N4, η οποία όμως είναι μία διαφοροποίηση της πιο δημοφιλούς, μη-παραμετρικής μεθόδου εξομάλυνσης ανομοιομορφίας έντασης σε πίκσελ, της μεθόδου N3. Θα παρουσιάσουμε οπότε, πρώτα τη μέθοδο N3 και στη συνέχεια θα εξηγήσουμε τις διαφοροποιήσεις της σε σχέση με τη μέθοδο N4, την οποία και χρησιμοποιήσαμε.

#### 3.2.3.1 N3 μέθοδος διόρθωσης πεδίου πόλωσης

[51] Η μη παραμετρική μέθοδος εξομάλυνσης ανομοιομορφίας (N3), στηρίζεται στην διόρθωση της έντασης ανομοιομορφίας μαγνητικών τομογραφιών. Δεν είναι παραμετρική από την άποψη ότι δε χρειάζεται παραμετρικό μοντέλο των εντάσεων των πίκσελ για τον ιστό, ούτε το διαμοιρασμό της κατανομής στην ένταση των πίκσελ σε έναν περιορισμένο αριθμό κατηγοριών ιστού. Χωρίς καμία υπόθεση, όσον αφορά το είδος της ανατομίας που

παρουσιάζεται στην μαγνητική τομογραφία, η μέθοδος αυτή αφαιρεί ελαφρές διαφοροποιήσεις στην ένταση των πίξελ, που παρουσιάζονται σε τομογραφίες εγκεφάλων.

Το πρόβλημα της διόρθωσης στην ένταση της ανομοιογένειας απλοποιείται σε μεγάλο βαθμό, εάν μοντελοποιηθεί σαν μία ομαλή πολλαπλασιαστική περιοχή. Στην περίπτωση αυτού του μοντέλου πεδίου, υπάρχει σταθερότητα με την πολλαπλασιαστική ανομοιογένεια, η οποία προκύπτει από την ευαισθησία στο πηνίο υποδοχής του μηχανήματος και σε μικρότερο βαθμό στην ανομοιογένεια λόγω επαγόμενων ρευμάτων και ανομοιομορφιών διεγέρσεων.

Έστω, οπότε το παρακάτω μοντέλο για το σχηματισμό της εικόνας από μαγνητικό τομογράφο:

$$v(x) = u(x)f(x) + n(x)$$

Η μεταβλητή  $x$  ορίζει μία συγκεκριμένη περιοχή στην εικόνα,  $v$  είναι το εκτημόμενο σήμα,  $u$  είναι το πραγματικό σήμα, το οποίο εκπέμπεται από τον ιστό,  $f$  είναι ένα άγνωστο, ομαλά μεταβαλλόμενο πεδίο πόλωσης και  $n$  λευκός Γκαουσιανός θόρυβος, για τον οποίο θεωρούμε ότι είναι ανεξάρτητος της μεταβλητής  $u$ . Το πρόβλημα της έντασης ανομοιογένειας, οπότε μετατρέπεται σε πρόβλημα υπολογισμού της  $f$ . Ο συνδυασμός, παρ'όλα αυτά της προσθετικής και πολλαπλασιαστικής παρέμβασης καθιστά το πρόβλημα αρκετά δύσκολο.

Στην περίπτωση, για παράδειγμα, που δεν υπάρχει ο θόρυβος  $n$ , οι πραγματικές εντάσεις  $u$ , σε κάθε περιοχή  $x$ , είναι ανεξάρτητες και ίδιας κατανομής τυχαίες μεταβλητές. Χρησιμοποιώντας το συμβολισμό  $v'(x) = \log(u(x))$ , το μοντέλο μετατρέπεται σε προσθετικό:

$$v'(x) = u'(x) + f'(x)$$

Θεωρούμε ότι η κατανομή των τιμών που λαμβάνει η  $f'$  πάνω στα σημεία ενδιαφέροντος είναι μία κατανομή πιθανότητας μίας τυχαίας μεταβλητής. Για παράδειγμα, εάν  $f'$ , είναι ένα γραμμικά αυξανόμενο πεδίο, ευθυγραμμισμένο πάνω σε μία τετράγωνη περιοχή, τότε η τελευταία θα έχει ομοιόμορφη κατανομή.

Έστω  $\mathbf{U}$ ,  $\mathbf{V}$  και  $\mathbf{F}$ , οι πυκνότητες πιθανότητας των  $u'$ ,  $v'$  και  $f'$  αντίστοιχα. Με την προσέγγιση ότι οι  $u'$  και  $f'$  είναι ανεξάρτητες ή ασυσχέτιστες τυχαίες μεταβλητές, η κατανομή του αθροίσματός τους υπολογίζεται μέσω της πράξης της συνέλιξης ως εξής:

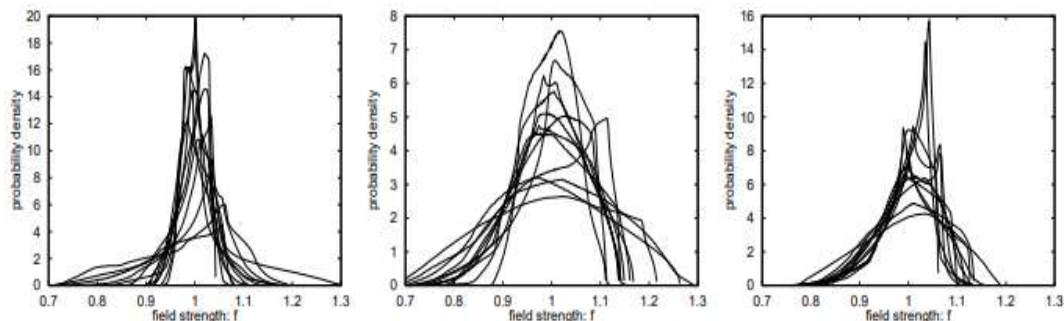
$$V(u') = F(u') * U(u') = \int F(v' - u')U(u')du'$$

Η κατανομή ανομοιογένειας της  $\mathbf{F}$ , μπορεί να ερμηνευθεί ως η θόλωση της κατανομής έντασης  $\mathbf{U}$ .

### 3.2.3.1.1 Στρατηγική Διόρθωσης

Αναλύοντας το φαινόμενο υπό το πρίσμα της ανάλυσης σήματος, η θόλωση εξ' αιτίας του πεδίου μειώνει τα υψηλής συχνότητας στοιχεία της  $\mathbf{U}$ . Το έργο διόρθωσης της έντασης ανομοιογένειας, στηρίζεται πρακτικά στην επαναφορά των στοιχείων συχνότητας της  $\mathbf{U}$ . Επειδή το σχήμα του πυρήνα θόλωσης της  $\mathbf{F}$  είναι άγνωστο, δεν είναι ξεκάθαρο ποιά στοιχεία συχνότητας της  $\mathbf{U}$ , πρέπει να επαναφερθούν, έτσι ώστε να μεταβούμε από τη φαινομενική κατανομή  $\mathbf{V}$ , στην πραγματική κατανομή  $\mathbf{U}$ . Παρ'όλα αυτά, λόγω του ότι το πεδίο ανομοιογένειας  $f'$ , πρέπει να είναι ομαλό και με αργό ρυθμό εναλλασσόμενο, υπάρχουν σχετικά λίγες σε αριθμό πιθανές κατανομές  $\mathbf{U}$ , οι οποίες αντιστοιχούν στη δοθείσα κατανομή

V. Στα πλαίσια του αλγόριθμου N3, η προσέγγιση στη διόρθωση της ανομοιογένειας είναι η εύρεση ενός ομαλού, αργά μεταβαλλόμενου πολλαπλασιαστικού πεδίου, το οποίο μεγιστοποιεί το περιεχόμενο συχνοτήτων της  $\mathbf{U}$ .



Εικόνα 3.15: Κατανομές πιθανότητας της  $F$ , 12 διαφορετικών ασθενών για τρεις διαφορετικούς μαγνητικούς τομογράφους.

Στις τρεις παραπάνω εικόνες φαίνονται οι αντίστοιχες κατανομές της  $F$ , για δώδεκα διαφορετικούς ασθενείς σε τρεις διαφορετικούς μαγνητικούς τομογράφους. Μία εικόνα αντιστοιχεί σε κάθε μηχάνημα, ενώ φαίνεται και στις τρεις περιπτώσεις πως το σήμα είναι απλής μορφής, όπως τονίστηκε και παραπάνω.

Αναλύοντας, στη συνέχεια, το κριτήριο βελτιστοποίησης, θα ήταν δυνατό να εξετάσουμε όλα τα πιθανά πεδία  $f'$ , με σκοπό να καταλήξουμε σε αυτό που μεγιστοποιεί το περιεχόμενο συχνοτήτων της  $\mathbf{U}$ . Όμως υπάρχουν δύο προβλήματα στην προηγούμενη αντιμετώπιση. Η αναζήτηση όλων των τρισδιάστατων πεδίων  $f'$ , είναι εξαιρετικά μεγάλη, με αποτέλεσμα οι φασματικές προσεγγίσεις και οι σχετικές μετρήσεις τους όπως αυτή της εντροπίας να είναι πολύ δύσκολο να υπολογιστούν με αρκετή ακρίβεια, προκειμένου να ανιχνευθούν ασθενείς μεταβολές στη  $\mathbf{U}$ .

Η προσέγγισή που υιοθετείται είναι να προτείνουμε μία κατανομή για τη  $\mathbf{U}$ , μέσω της συγκέντρωσης της κατανομής  $\mathbf{V}$  και στη συνέχεια να προσεγγίσουμε το ακόλουθο ομαλό πεδίο  $f'$ , το οποίο παράγει κατανομή  $\mathbf{U}$  «κοντά» σε αυτό που προτάθηκε. Ψάχνοντας τη  $\mathbf{U}$ , ανάμεσα σε ένα μεγάλο σύνολο στοιχείων, μπορεί να φαίνεται αντίστοιχης δυσκολίας με την εύρεση της  $f'$ , αλλά η σημαντική διαφορά, την οποία και εκμεταλλευόμαστε είναι ότι η κατανομή  $F$ , έχει πολύ απλή μορφή. Έστω ότι η τελευταία είναι Γκαουσιανής μορφής. Τότε αρκεί μόνο να αναζητήσουμε όλες τις κατανομές  $\mathbf{U}$ , οι οποίες αντιστοιχούν σε  $F$  Γκαουσιανής κατανομής, με μηδενική μέση τιμή και δοθείσα διασπορά. Με αυτόν τον τρόπο όλες οι κατανομές  $\mathbf{U}$ , γίνονται μίας διάστασης, και συγκεκριμένα το πλάτος της κατανομής  $F$ .

Στην πράξη, η  $F$  προσεγγίζεται σαν Γκαουσιανή κατανομή και μερικές από τις υποθέσεις μας, όπως ο μηδενικός θόρυβος αγνοούνται. Αυτές οι δυσκολίες αντιμετωπίζονται με το να εφαρμόσουμε μία επαναληπτική διαδικασία, όσον αφορά την εκτίμηση της κατανομής  $\mathbf{U}$  και του αντίστοιχου πεδίου  $f'$ . Γνωρίζουμε ότι κάθε Γκαουσιανή κατανομή μπορεί να αναλυθεί σε συνέλιξη «στενότερων» Γκαουσιανών κατανομών. Δηλαδή κατανομών με κύριο λοβό μικρότερου πλάτους. Σαν αποτέλεσμα, ο χώρος όλων των πιθανών κατανομών για τη  $\mathbf{U}$ , που αντιστοιχούν σε Γκαουσιανή κατανομή της  $F$ , μπορεί να αναζητηθεί μέσω της αντίστροφης

συνέλιξης «στενών» Γκαουσιανών κατανομών από συνεχόμενες προσεγγίσεις της  $\mathbf{U}$ . Το θετικό στοιχείο της μεθόδου αυτής είναι ότι ανάμεσα σε συνεχόμενες προσεγγίσεις της  $\mathbf{U}$ , υπολογίζεται και ένα αντίστοιχο πεδίο  $f'$ . Ο περιορισμός ότι το πεδίο πρέπει να είναι ομαλό, αλλάζει το σχήμα της προτεινόμενης κατανομής  $\mathbf{U}$ , με την προοπτική να είναι συνεπές με το πεδίο. Οι διαταραχές της  $\mathbf{U}$ , αλλάζουν τη Γκαουσιανή μορφή της  $\mathbf{F}$ , εκφράζοντας έτσι τη διαταραχή που προκαλείται στη  $\mathbf{V}$ , λόγω θορύβου και άλλων παραγόντων. Η μέθοδος αυτή επαναλαμβάνεται μέχρι να μην μπορούν να προκύψουν άλλες αλλαγές σε  $f'$  και  $\mathbf{U}$  από την αντίστροφη διαδικασία της συνέλιξης στενών Γκαουσιανών κατανομών της  $\mathbf{V}$ .

### 3.2.3.1.2 Εκτίμηση πεδίου

Στην παράγραφο αυτή αναλύουμε σε μεγαλύτερο βαθμό τη διαδικασία μέσω της οποίας προτείνουμε κατανομές για τη  $\mathbf{U}$ , ενώ παράλληλα υπολογίζουμε αντίστοιχα τα πεδία. Για απλότητα θεωρούμε ότι η πραγματική κατανομή των εντάσεων  $\mathbf{U}$ , λαμβάνεται σε μία μοναδική επανάληψη μέσω της παραπάνω διαδικασίας.

Δοθείσας της  $\mathbf{U}$ , η μέθοδος εκτίμησης του πεδίου είναι η παρακάτω:

Για μία μέτρηση της  $v'$ , σε μία θέση  $x$ , υπολογίζεται η  $u'$ , μέσω των κατανομών  $\mathbf{U}$  και  $\mathbf{F}$ . Επειδή η επιλογή του σημείου συμβαίνει αυθαίρετα, η μέτρηση της  $u'$ , μπορεί να αντιμετωπιστεί σαν ένα τυχαίο δείγμα της κατανομής  $\mathbf{V}$ . Οπότε λαμβάνουμε την  $u'$  από τη  $v'$ , ως εξής:

$$E[u' | v'] = \int_{-\infty}^{\infty} u' p(u' | v') du' = \int_{-\infty}^{\infty} \frac{p(u', v')}{p(v')} du'$$

Γράφοντας την  $p(u')$  σαν  $V(u')$ , λαμβάνουμε:

$$E[u' | v'] = \frac{1}{V(u')} \int_{-\infty}^{\infty} u' p(u', v') du' = \frac{\int_{-\infty}^{\infty} u' F(v'-u') U(u') du'}{\int_{-\infty}^{\infty} F(v'-u') U(u') du'}$$

Με αυτόν τον τρόπο μπορεί να ληφθεί μία προσέγγιση της  $u'$ , ως εξής:

$$f'e(v') = E[f' | v'] = v' - E[u' | v']$$

,όπου  $f'e$  είναι η εκτίμηση της  $f'$ , στο σημείο  $x$ , βασισμένη στη μοναδική εκτίμηση  $v'$  στο σημείο  $x$ . Αναφέρουμε επίσης ότι σε περίπτωση που καθίσταται αυτό απαραίτητο η παραπάνω εκτίμηση είναι δυνατό να εξομαλυνθεί. Οι πυκνότητες πιθανότητας  $\mathbf{U}, \mathbf{V}$  και  $\mathbf{F}$ , είναι δυνατόν να ληφθούν από τον υπολογισμό των ιστογραμμάτων των  $u', v'$  και  $f'$ , αντίστοιχα. Η εκτίμηση του ομαλού φίλτρου μετά από μία επανάληψη δεν επαρκεί για να εξαφανιστεί πλήρως η ανομοιομορφία, αλλά είναι αρκετή για να διαφοροποιηθούν οι ιστοί. Επαναλαμβάνοντας όμως τη διαδικασία, μαζί με τον υπολογισμό της  $f'e$ , έχουμε ικανοποιητικά αποτελέσματα όσον αφορά την παραπάνω μέθοδο.

### 3.2.3.1.3 Εκτίμηση της $\mathbf{U}$

Δοθείσας της κατανομής  $\mathbf{F}$ , και της εκτιμώμενης κατανομής εντάσεων  $\mathbf{V}$ , η  $\mathbf{U}$  εκτιμάται ως εξής:

$$G' = \frac{F^*}{|F'|^2 + Z^2} \quad \text{και} \quad U' = G'V'$$

Σε αυτή την περίπτωση το σύμβολο \* υποδηλώνει τον αντίστοιχο συζυγή αριθμό,  $F'$  είναι ο μετασχηματισμός Fourier της  $F$ , και  $Z$  είναι ένας σταθερός όρος, που εισάγεται στην παραπάνω σχέση για να μειωθεί το μέγεθος της μεταβλητής  $G'$ . Η ληφθείσα εκτίμηση της  $U$ , οπότε χρησιμοποιείται για να εκτιμηθεί το αντίστοιχο πεδίο  $f'$ .

Αναλύοντας την προηγούμενη μέθοδο παραλήψαμε εσκεμένα τον εκτιμητή B-spline, ο οποίος εφαρμόζεται ως εξής:

$$\begin{aligned} \hat{u}^n &= \hat{v} - \hat{f}_e^n \\ &= \hat{v} - S\{\hat{v} - E[\hat{u} | \hat{u}^{n-1}]\} \end{aligned}$$

Στην παραπάνω σχέση, η οποία χρησιμοποιείται στη μέθοδο N3, εκτιμάται η πραγματική κατανομή του σήματος  $u$ , μέσω του τελεστή  $S\{\cdot\}$ .

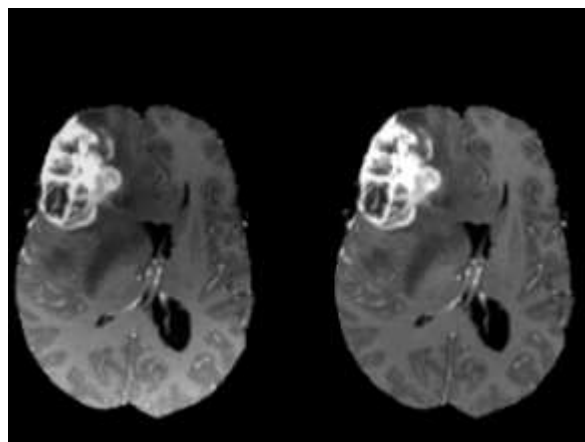
Στην περίπτωση του αλγόριθμου N4ITK, η σχέση γίνεται

$$\begin{aligned} \hat{u}^n &= \hat{u}^{n-1} - \hat{f}_r^n \\ &= \hat{u}^{n-1} - S^*\{\hat{u}^{n-1} - E[\hat{u} | \hat{u}^{n-1}]\} \end{aligned}$$

$S^*$ , είναι μία διαφορετική περίπτωση εκτιμητή και  $f_r^n$  είναι η παράμετρος  $f_e$ , στη  $n$ -οστή επανάληψη.

Μέσω των παραπάνω αλλαγών, επιτρέπεται έλεγχος σε μικρότερη έκταση, με αποτέλεσμα να επιτυγχάνεται με πεδία ισχυρότερης έντασης, δίχως πιθανότητα αστοχίας στον αλγόριθμο.

Ακόμα, μέσω αυτού δε χρειάζονται παράμετροι όπως αυτή της τεχνητής κανονικοποίησης, ενώ επιτυγχάνονται μικρότερος χρόνος σύγκλισης και αποδοτικότερες στρατηγικές, όσον αφορά την ένταση πεδίων υψηλότερης διαμόρφωσης για το σήμα πόλωσης.



Εικόνα 3.16: Εικόνα μαγνητικού τομογράφου πριν (αριστερά) και μετά (δεξιά) από τη μέθοδο προεπεξεργασίας διόρθωσης πεδίου πόλωσης.

Από τις δύο παραπάνω εικόνες η δεξιά έχει υποστεί τη μέθοδο με αποτέλεσμα να παρουσιάζεται μικρότερη ανομοιογένεια στο περιεχόμενο της.

#### 3.2.4 Αφαίρεση εικόνων με μικρή περιεκτικότητα σε χαρακτηριστικά

Αξίζει τέλος να σημειωθεί ότι το σύνολο των δεδομένων, έχει παραχθεί με τον εξής τρόπο: Ο μαγνητικός τομογράφος, όπως θα εξηγήσουμε και παρακάτω παράγει μαγνητικό παλμό προκειμένου να διεγερθούν τα άτομα υδρογόνου. Αυτό δε συμβαίνει ακαριαία, αλλά ο εγκέφαλος αρχίζει να εμφανίζεται από ένα χρονικό σημείο και μετά σε εικόνες που αφορούν την ίδια τομογραφία. Από τη δημιουργία μέχρι και τη λήξη του παλμού, λαμβάνονται 155 εικόνες, από τις οποίες πολλές είναι κενές, καθώς τα άτομα υδρογόνου δεν έχουν αποδιεγερθεί για να σταλεί το σήμα πίσω και να εμφανιστεί η εικόνα.

Σαν αποτέλεσμα πολλές εικόνες δεν περιέχουν πληροφορία, ενώ ακόμα περισσότερες δεν εμφανίζουν το πλήθος χαρακτηριστικών που θα θέλαμε. Δηλαδή, φαίνεται μόνο το 10% ή το 20% το εγκεφάλου. Επειδή, λοιπόν, οι εικόνες αυτές δεν προσφέρουν καμία πληροφορία στο δίκτυό μας, κατά τη διαδικασία της εκπαίδευσης, αποφασίσαμε να αφαιρέσουμε αυτές, των οποίων το 20% των πίξελ τουλάχιστον είναι κενά. Σαν αποτέλεσμα, το σύνολο εικόνων που απέμεινε, είναι μικρότερο, άρα το δίκτυο μπορεί να εκπαιδευτεί με πιο γρήγορο ρυθμό, ενώ ταυτόχρονα δε «σπαταλάμε» φίλτρα από τα συνελκτικά επίπεδα του δικτύου, για να αναγνωριστούν κενές εικόνες, δίχως πληροφορία, και να ενταχθούν σε κάποια κατηγορία εκ των δύο που μας ενδιαφέρουν.

Επίσης, οι περισσότερες εικόνες είναι τύπου (HGG), υψηλού βαθμού γλοιωμάτων, οπότε το δίκτυο σε κάθε περίπτωση θα έτεινε, λανθασμένα, να συνδυάσει όλες τις κενές εικόνες με αυτή την κατηγορία. Τέλος, αν δεν αφαιρούσαμε τις εικόνες αυτές, οι οποίες περιέχουν μικρό τμήμα του εγκεφάλου και πολλές φορές δεν εμφανίζουν το γλοιώμα, τότε το σύστημα πάλι θα έτεινε να συνδυάσει μόνο το συγκεκριμένο τμήμα του εγκεφάλου με μία εκ των δύο κατηγοριών. Αυτό δεν θα ήταν επιθυμητό, καθώς προσπαθούμε να «εξαναγκάσουμε» το σύστημά μας να αναγνωρίζει συγκεκριμένα χαρακτηριστικά, με σκοπό να τα εντάσσει κάθε εικόνα σε μία εκ των δύο κατηγοριών. Δίχως το παραπάνω θα συνδυάζε συγκεκριμένες μορφολογίες εγκεφάλων με κάθε τύπο γλοιώματος.

## Κεφάλαιο 4<sup>ο</sup> Πειραματικό Στάδιο

---

Στα πλαίσια της παρούσας διπλωματικής εργασίας, ύστερα από την επιλογή των μοντέλων μηχανικής μάθησης αλλά και των τεχνικών προεπεξεργασίας, που θα χρησιμοποιηθούν, θα ασχοληθούμε με το σύνολο των δεδομένων πάνω στο οποίο εκπαιδεύτηκαν τα μοντέλα και εφαρμόστηκαν οι παραπάνω τεχνικές, αλλά και οι τρόποι μέσω των οποίων εκτελέσαμε τα πειράματα.

Σκοπός του κεφαλαίου αυτού, δηλαδή, είναι η ανάλυση και παρουσίαση των χαρακτηριστικών του συνόλου εικόνων, αλλά η παρουσίαση των εργαλείων που χρησιμοποιήθηκαν τόσο στην υλοποίηση όσο και στην ανάλυση των πειραμάτων.

Το σύνολο δεδομένων μας, δηλαδή το σύνολο των εικόνων που χρησιμοποιήσαμε για την εκπαίδευση των μοντέλων, αλλά και για την επαλήθευση του δικτύου, όπως προαναφέρθηκε και στην εισαγωγή, προέρχεται από τον διαγωνισμό MICCAI BRATS CHALLENGE 2015. Ο τελευταίος πραγματοποιείται κάθε χρόνο από το 2012 και μετά. Κάθε χρόνο, παρουσιάζεται ένα σύνολο δεδομένων από εικόνες μαγνητικών τομογραφιών πάνω στον ανθρώπινο εγκέφαλο. Ομάδες από όλο τον κόσμο δοκιμάζουν τεχνικές πάνω στο σύνολο αυτό, με σκοπό την ταξινόμηση των εικόνων του σε επιμέρους κατηγορίες. Σαν αποτέλεσμα, κάθε χρόνο συγκρίνονται τα αποτελέσματα των τεχνικών όλων των ομάδων που έλαβαν μέρος και παρουσιάζονται οι τεχνικές εκείνες, οι οποίες παρουσίασαν υψηλότερη απόδοση.

Αξίζει ακόμη να αναφερθεί ότι το σύνολο δεδομένων δεν είναι το ίδιο σε κάθε διαγωνισμό. Συγκεκριμένα, το σύνολο των εικόνων του επόμενου χρόνου περιέχει όλες τις εικόνες του προηγούμενου, αλλά ταυτόχρονα προστείνονται και καινούριες, με αποτέλεσμα το πρόβλημα να παρουσιάζει κάθε χρόνο νέας μορφής δυσκολίες.

Το σύνολο των εικόνων του παραπάνω διαγωνισμού, αποτελείται από 274 μαγνητικές τομογραφίες διαφορετικών ασθενών με γλοιώματα στον εγκέφαλο. Από αυτές, οι 54 απεικονίζουν χαμηλού βαθμού γλοιώματα και οι υπόλοιπες 220 υψηλού βαθμού γλοιώματα.

Το παραπάνω σύνολο αποτελεί το σύνολο δεδομένων του διαγωνισμού BRATS CHALLENGE 2015, από τη βάση δεδομένων VSD ( Virtual Skeleton Database). Από το σύνολο των 274 ασθενών, των οποίων οι μαγνητικές απαρτίζουν το σύνολο των δεδομένων, υπάρχουν τέσσερις διαφορετικοί τύποι τομογραφιών. Σαν αποτέλεσμα, λαμβάνουμε μία πιο ολοκληρωμένη εικόνα από το εσωτερικό του εγκεφάλου, καθώς και των γλοιωμάτων, που βρίσκονται στο εσωτερικό του.

Πριν προχωρήσουμε στα διαφορετικά είδη μαγνητικών τομογραφιών, όμως, είναι χρήσιμο να αναφέρουμε μερικές από τις τεχνικές πάνω στις οποίες βασίζεται η λειτουργία του μαγνητικού τομογράφου. Μέσω αυτού θα γίνει αντιληπτό σε μεγαλύτερο βαθμό η διαφοροποίηση των τεχνικών προεπεξεργασίας που χρησιμοποιήθηκαν και που αποσκοπεί η κάθεμία.

### 4.1 Μαγνητικός Τομογράφος

Η μαγνητική τομογραφία είναι μία μέθοδος ιατρικής απεικόνισης, η οποία χρησιμοποιείται στη ραδιολογία, προκειμένου να παράχθούν εικόνες που αφορούν την ανατομία, αλλά και τις λειτουργίες του οργανισμού. Είναι χρήσιμη τόσο σε ασθενή, αλλά και σε υγιή άτομα, διότι στους πρώτους βοηθά στην αναγνώριση του τύπου καρκίνου, ενώ στους δεύτερους χρησιμοποιείται κυρίως προληπτικά. Οι μαγνητικοί τομογράφοι χρησιμοποιούν μαγνητικά πεδία, προκειμένου να απεικονίσουν όργανα στο εσωτερικό του σώματος των ασθενών.



Συγκεκριμένα, ο ασθενής τοποθετείται στο εσωτερικό του μαγνητικού τομογράφου, ο οποίος παράγει ένα ισχυρό μαγνητικό πεδίο, περιμετρικά γύρω από αυτόν, στην περιοχή που επιλέγουμε να απεικονίσουμε. Στις περισσότερες ιατρικές εφαρμογές, τα πρωτόνια (πυρήνες ατόμων υδρογόνου), στο εσωτερικό ιστών που περιέχουν μόρια νερού, παράγουν ένα σήμα, το οποίο και χρησιμοποιείται, με σκοπό την απεικόνιση του σώματος.

Αρχικά, ενέργεια από ένα ταλαντώμενο μαγνητικό πεδίο εφαρμόζεται προσωρινά στον ασθενή σε κατάλληλη συχνότητα συντονισμού. Τα διεγερμένα άτομα υδρογόνου εκπέμπουν μία ραδιοσυχνότητα, η οποία προσλαμβάνεται από ένα πηνίο λήψης. Το σήμα αυτό μπορεί να κωδικοποιεί πληροφορίες τοποθεσίας, μέσω της εναλλαγής του βασικού μαγνητικού πεδίου, κάνοντας χρήση ειδικών πηνίων.

Η αντίθεση ανάμεσα σε διαφορετικούς ιστούς καθορίζεται από το χρονικό διάστημα κατά το οποίο απατείται, προκειμένου τα διεγερμένα άτομα να επιστρέψουν στην κατάσταση ισορροπίας τους.

Κάθε διαφορετικός τύπος ιστού επιστρέφει στην κατάσταση ισορροπίας του ανάλογα με τη διαδικασία χαλάρωσης αυτού, η οποία αναλύεται σε δύο επιμέρους κατηγορίες.

Χαλάρωση τύπου T1, όπου ο μαγνητισμός συμβαίνει στην κατεύθυνση του στατικού μαγνητικού πεδίου και χαλάρωση τύπου T2, όπου ο μαγνητισμός είναι κάθετος στο πεδίο. Μαγνητική τύπου T1 επιτυγχάνεται πριν τη μέτρηση του μαγνητικού σήματος, αλλάζοντας το χρόνο μεταξύ διαδοχικών διεγέρσεων, ενώ μαγνητική τύπου T2 επιτυγχάνεται μέσω της απόσβεσης του μαγνητισμού, πριν τη μέτρηση του μαγνητικού σήματος, με την εναλλαγή του χρόνου ηχούς (echo time).

Με γνωστά τα παραπάνω, θα αναλύσουμε τις παραπάνω κατηγορίες τομογραφιών που μας παρέχει το σύνολο δεδομένων:

- T1: Αυτού του τύπου οι μαγνητικές τομογραφίες συμβάλλουν στην εκτίμηση του εγκεφαλικού φλοιού, αναγνωρίζουν το λιπώδη ιστό και γενικά βοηθούν στην κατανόηση μορφολογικών χαρακτηριστικών.
- T2: Σε αυτή την περίπτωση είναι δυνατός ο εντοπισμός οιδήματος και φλεγμονής, καθώς και ανίχνευση αλλοίωσης στη λευκή ουσία.
- T1c: Σε αυτή την περίπτωση, είναι δυνατή η ανίχνευση κακοηθών όγκων (enhancing tumor). Τα καρκινικά κύτταρα που ανήκουν σε αυτό το σύνολο, δηλαδή, εμφανίζουν έντονη αναπαραγωγή.
- FLAIR: Εμφανίζεται ολόκληρος ο όγκος, οπότε δίνει τη μέγιστη πληροφορία, όσον αφορά το σχήμα και τη διάταξή του στην εικόνα.

## 4.2 Αφαίρεση εικόνων από το σύνολο εκπαίδευσης

Αξίζει τελικά να αναφερθεί ότι δε χρησιμοποιήσαμε το σύνολο των δεδομένων για την εκπαίδευση του δικτύου, αλλά για λόγους που θα εξηγηθούν αναλυτικότερα παρακάτω από το αρχικό σύνολο των εικόνων, οι οποίες ήταν περίπου 200.000, όπως προαναφέρθηκε, αφαιρέσαμε αυτές που είχαν με μικρό δείκτη πληροφορίας.

Τις υπόλοιπες 120.000, εκ των οποίων οι 100.000 απεικονίζουν εγκεφάλους ασθενών με υψηλού βαθμού γλοιώματα και οι άλλες 20.000, ασθενείς με χαμηλού βαθμού γλοιώματα, χωρίσαμε σε δύο σύνολα. Συγκεκριμένα, το 90% κάθε συνόλου από τα παραπάνω απομονώθηκε με τέτοιο τόπο ώστε με το υπόλοιπο 10% να μην απεικονίζει τους ίδιους ασθενείς και αποτέλεσε το σύνολο εκπαίδευσης. Δηλαδή, 108.000 εικόνες (90.000 υψηλού βαθμού γλοιωμάτων και 18.000 χαμηλού βαθμού). Οι υπόλοιπες, 12.000 εικόνες (10.000 υψηλού βαθμού και 2.000 χαμηλού βαθμού) ανήκουν στο σύνολο επαλήθευσης.

Τελικά είναι αναγκαίο να τονιστεί ότι το σύνολο εκπαίδευσης και το σύνολο επαλήθευσης δεν περιέχουν εικόνες από ίδιους ασθενείς. Και αυτό, διότι σε τέτοια περίπτωση, το σύνολο επαλήθευσης δεν θα ήταν αντικειμενικός δείκτης, καθώς τα δεδομένα επαλήθευσης θα ήταν «γνωστά» από το σύνολο εκπαίδευσης.

#### 4.2.1 Σύνολο Επαλήθευσης

Σε αυτό το σημείο πρέπει να αναφερθεί ότι όσον αφορά την απόδοση του δικτύου, η χρήση όλων των εικόνων για την εκπαίδευση ενέχει τους εξής κινδύνους.

Πρώτον να οδηγηθούμε σε μοντέλο μηχανικής μάθησης, το οποίο δεν δίνει λύση στο πρόβλημά μας. Αυτό συμβαίνει, στην περίπτωσή μας, διότι στο μεγάλο πλήθος εικόνων (200.000), οι περισσότερες εικόνες περιέχουν μηδενική ή ελάχιστη ποσότητα χαρακτηριστικών. Οπότε, το σύστημα καταλήγει να μην ταξινομεί κάθε δείγμα σε κατηγορίες βάση αντικειμενικών χαρακτηριστικών, με αποτέλεσμα να μην μπορεί να χρησιμοποιηθεί. Δεύτερον, παρατηρείται το φαινόμενο του overfitting. Δηλαδή, το ελάχιστο σφάλμα κατά την εκπαίδευση παρατηρείται σε ένα συγκεκριμένο χρονικό σημείο  $t$ , και στη συνέχεια αρχίζει και αυξάνεται συνεχόμενα έως ότου συγκλίνει σε μία σταθερή τιμή. Στόχος μας είναι να βρούμε το ελάχιστο σφάλμα τη συγκεκριμένη αυτή χρονική στιγμή και να «παγώσουμε» το δίκτυο σε αυτές τις παραμέτρους που εμφάνισαν το ελάχιστο αυτό σφάλμα. Ο δείκτης, ο οποίος μας δίνει την πληροφορία αυτή μέσω μετρικών δεν είναι το σύνολο εκπαίδευσης, διότι όσο το δίκτυο εκπαιδεύεται τόσο αποδοτικότερο θα γίνει στο να αναγνωρίζει τα δεδομένα εκπαίδευσης. Η λύση βρίσκεται, οπότε, στο σύνολο επαλήθευσης, το οποίο αποτελεί αντικειμενικό δείκτη και αυτό επειδή τα δεδομένα αυτού δεν θα «μαθαίνονται» αλλά μόνο θα προβλέπονται από το σύστημα.

Λόγω των παραπάνω, το σύνολο εκπαίδευσης δεν αρκεί για την αποδοτική εκπαίδευση. Χρειαζόμαστε δηλαδή και ένα άλλο σύνολο το οποίο και θα χρησιμοποιούμε μόνο για μετρήσεις και του οποίου τα δεδομένα δε θα «μαθαίνονται» σκόπιμα από το σύστημα, προκειμένου να έχουμε έναν δείκτη για το πόσο αποδοτικό είναι ανά πάσα χρονική στιγμή το σύστημά μας. Το σύνολο επαλήθευσης χρησιμοποιείται στο τέλος κάθε εποχής, μετά τις ενημερώσεις των βαρών, και ανάλογα με τις προβλήψεις, στις οποίες επιτυγχάνει το σύστημα, οδηγούμαστε και σε αντίστοιχες μετρικές, οι οποίες μας πληροφορούν για την κατάσταση και την απόδοσή του συστήματος.

### 4.3 Εκπαίδευση

Στα πλαίσια της παρούσας διπλωματική εργασίας, χρησιμοποιήσαμε τρία μοντέλα συνελκτικών νευρωνικών δικτύων και πέντε μεθόδους προεπεξεργασίας δεδομένων.

Δοκιμάσαμε αρκετούς συνδυασμούς ανάμεσα στις μεθόδους προεπεξεργασίας, στα συνελκτικά μοντέλα, στους αλγόριθμους εκπαίδευσης και στον τρόπο με τον οποίο εισάγαμε τις εικόνες στο εκάστοτε μοντέλο. Σε όλες τις περιπτώσεις, παρ'όλ'αυτά, αφαιρέσαμε τις εικόνες με μηδενική ή ελάχιστη περιεκτικότητα σε πληροφορία, όπως είναι οι κενές εικόνες ή αυτές που εμφανίζουν πολύ μικρό τμήμα του εγκεφάλου.

Ακόμη, στο σύνολο των εκπαιδεύσεων, για οποιοδήποτε μοντέλο μηχανικής μάθησης και συνδυασμό παραμέτρων χρησιμοποιήσαμε μεθόδους που θα αναλύσουμε στη συνέχεια, αφού πρώτα εξηγήσουμε τις μετρικές σύμφωνα με τις οποίες αξιολογήσαμε τα αποτελέσματα :

#### 4.3.1 Μετρικές

Η αξιολόγηση των μοντέλων σε κάθε περίπτωση από αυτές που θα αναφερθούν στη συνέχεια, έγινε κάνοντας χρήση των παρακάτω τεσσάρων στατιστικών εγαλείων, οι σχέσεις των οποίων στηρίζονται στους τέσσερις παρακάτω βασικούς ορισμούς:

**TP** : Η περίπτωση κατά την οποία η πρόβλεψη ταυτίζεται με την αλήθεια και το προβλεπόμενο δείγμα ανήκει όντως στη δοθείσα κατηγορία.

**TN** : Η περίπτωση κατά την οποία η πρόβλεψη ταυτίζεται με την αλήθεια και το προβλεπόμενο δείγμα δεν ανήκει στη δοθείσα κατηγορία.

**FP** : Η περίπτωση κατά την οποία η πρόβλεψη δεν ταυτίζεται με την αλήθεια και προβλεψαμε ότι το δείγμα ανήκει σε μία κατηγορία, ενώ στην πραγματικότητα δεν ανήκει.

**FN** : Η περίπτωση κατά την οποία η πρόβλεψη δεν ταυτίζεται με την αλήθεια και προβλεψαμε ότι το δείγμα δεν ανήκει σε μία κατηγορία, ενώ στην πραγματικότητα ανήκει σε αυτό.

- **Accuracy** : Η πιο συνηθισμένη μέθοδος μέσω της οποίας ερμηνεύονται τα αποτελέσματα ενός πιθανοτικού μοντέλου, όπως στην περίπτωση μας, είναι η ακρίβεια. Ορίζεται ως το κλάσμα με αριθμητή το πλήθος των ορθών προβλέψεων και παρονομαστή το σύνολο των προβλέψεων. Επειδή πάντα οι ορθές προβλέψεις θα είναι λιγότερες σε πλήθος από τις συνολικές προβλέψεις και η ελάχιστη δυνατή τιμή των ορθών προβλέψεων είναι μηδέν, το σύνολο τιμών της ακρίβειας είναι όλοι οι πραγματικοί αριθμοί στο διάστημα από 0 έως και 1,  $x \in [0,1]$

$$Accuracy = \frac{TP + TN}{P + N} = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Precision** : Η μετρική αυτή ορίζεται σαν το πλήθος των παρατηρήσεων, οι οποίες εντάχθηκαν από το σύστημα σε ορθή κατηγορία, προς το άθροισμα αυτών μαζί με τις εικόνες που δεν εντάχθηκαν σωστά στην ίδια κατηγορία. Στην περίπτωση μας αυτό είναι σημαντικό, διότι οι εικόνες μας ανήκουν σε δύο κατηγορίες με μεγάλη διαφορά στη συχνότητα εμφάνισης. ( $90.000 / 18.000 = 5$ ). Στην εκπαίδευση, δηλαδή, θα εμφανίζονται 5 εικόνες υψηλού βαθμού γλοιώματος και μία χαμηλού. Σαν αποτέλεσμα, η πρώτη κατηγορία με τις περισσότερες εικόνες έχει μεγαλύτερο αντίκτυπο, όσον αφορά τη μετρική της ακρίβειας σε σχέση με τη δεύτερη. Οπότε, η μετρική precision, είναι πιο αντικειμενικός δείκτης.

$$Precision = \frac{TP}{TP + FP}$$

- **Recall** : Αυτή η περίπτωση μετρικής, η οποία ορίζεται ως το σύνολο των ορθών προβλέψεων μίας κατηγορίας προς το άθροισμα αυτών μαζί με τις λανθασμένα μη εισαγόμενες εικόνες στην ίδια κατηγορία. Η χρήση της μετρικής αυτής είναι αποτελεσματική στο πρόβλημά μας για τον ίδιο λόγο που είναι και η παραπάνω μετρική (precision). Διότι, μας δίνει πληροφορία, η οποία είναι σε κάποιο βαθμό ανεξάρτητη από τη συχνότητα μεταξύ των κατηγοριών.

$$Recall = \frac{TP}{TP + FN}$$

- **F1-score**: Τέλος, αυτή η περίπτωση μετρικής είναι ο αρμονικός μέσος όρος των precision και recall. Είναι η μετρική εκείνη η οποία χρησιμοποιήθηκε μαζί με αυτή της ακρίβειας στο επόμενο κεφάλαιο στην παρουσίαση των γραφικών, διότι πρώτον συνδυάζει τις δύο παραπάνω και δεύτερον αποτελεί και αυτή δείκτη ανεξάρτητο σε κάποιο βαθμό από τη διαφορά συχνότητας μεταξύ των διαφορετικών κατηγοριών.

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

## 4.4 Περιβάλλον Python 2.7

Σε αυτό το σημείο και αφού αναλύσαμε το σύνολο δεδομένων, αλλά και τις μετρικές σύμφωνα με τις οποίες αξιοποιήσαμε το κάθε μοντέλο, θα αναφέρουμε μερικές πληροφορίες για τα εργαλεία, τα οποία χρησιμοποιήθηκαν, προκειμένου να υλοποιήσουμε τα μοντέλα μηχανικής μάθησης, την εκπαίδευση αλλά και την εξαγωγή πληροφοριών από τα μοντέλα αυτά.

Αρχικά, θα αναφέρουμε ότι το προγραμματιστικό περιβάλλον, το οποίο χρησιμοποιήθηκε είναι αυτό της γλώσσας προγραμματισμού Python, στην έκδοση 2.7 .

Επιλέξαμε το συγκεκριμένο περιβάλλον, διότι καθιστά εξαιρετικά εύκολη τη διαχείριση των δεδομένων. Μεταξύ άλλων χαρακτηριστικών μετατρέπει τα δεδομένα ενός τύπου αυτόματα σε αντίστοιχα δεδομένα του επιθυμητού τύπου και κυρίως μας παρέχει βιβλιοθήκες, οι οποίες υλοποιούν διαδικασίες προεπεξεργασίας δεδομένων, μετατροπής εικόνων σε διάφορες μορφές, δόμηση μοντέλων μηχανικής μάθησης όπως είναι τα συνελκτικά νευρωνικά δίκτυα, παρουσίασης αποτελεσμάτων σε γραφήματα, καθώς και άλλου τύπου τεχνικές, που θα αναφερθούν παρακάτω.

Τα περισσότερα από τα παραπάνω πραγματοποιούνται μέσω δύο βιβλιοθηκών, οι οποίες έχουν υλοποιηθεί σε περιβάλλον Python και ονομάζονται keras και tensorflow.

### 4.4.1 Keras-Tensorflow

Είναι, όπως αναφέρθηκε και παραπάνω, βιβλιοθήκες, οι οποίες λειτουργούν και σε περιβάλλον Python 2.7. Σκοπός τους είναι η υλοποίηση μοντέλων μηχανικής μάθησης, προκειμένου να επιτευχθεί άμεσα πειραματισμός με τα παραπάνω και να εξαχθούν αποτελέσματα και συμπεράσματα. Όλες οι υλοποιήσεις των μοντέλων που θα χρησιμοποιήσουμε στα πλαίσια της παρούσας διπλωματικής εργασίας, έχουν συμβεί με τη βοήθεια αυτών των βιβλιοθηκών και συναρτήσεων που παρέχονται από αυτές.

Συγκεκριμένα, τα διάφορα επίπεδα που συνθέτουν το εκάστοτε δίκτυο όπως τα συνελκτικά ή τα πλήρως συνδεδεμένα επίπεδα, συναρτήσεις ενεργοποίησης, ακόμα και αλγόριθμοι εκμάθησης, παρέχονται σε μορφή κώδικα μέσω των παραπάνω βιβλιοθηκών.

### 4.4.2 SimpleITK (Insight Segmentation and Registration Toolkit - ITK)

Το παραπάνω είναι εργαλείο ανοιχτού κώδικα, το οποίο παρέχει μεταξύ άλλων τη δυνατότητα μετατροπής αρχείων σε διάφορες μορφές και χρησιμοποιήθηκε στα πλαίσια της εργασίας αυτής με τον εξής τρόπο.

Για κάθε ασθενή αρχικά λάβαμε τέσσερις τύπους από τομογραφίες, όπως αναφέραμε και παραπάνω. Τα αρχεία του τύπου αυτού ήταν σε μορφή .mha. Η παραπάνω μορφή δεν είναι αναγνωρίσιμη από το σύστημα μας και σαν αποτέλεσμα καθίσταται απαραίτητη η μετατροπή της σε εικόνες τύπου .png.

Κάθε αρχείο της πρώτης μορφής (.mha), περιέχει 155 εικόνες σε συμπιεσμένη μορφή. Μέσω συναρτήσεων του SimpleITK, αλλά και άλλων που υλοποιήθηκαν στα πλαίσια της εργασίας αυτής ή μας παρείχε η βιβλιοθήκη keras, μετατρέψαμε κάθε αρχείο τέτοιου τύπου σε 155 εικόνες τύπου png.

Σαν αποτέλεσμα, λάβαμε ένα σύνολο εικόνων σε μορφή κατάλληλη για επεξεργασία από το σύστημα.

### 4.4.3 Συναρτήσεις προεπεξεργασίας δεδομένων

Όσον αφορά τις μεθόδους προεπεξεργασίας, οι οποίες αναλύθηκαν θεωρητικά στο προηγούμενο κεφάλαιο, τις δύο από αυτές, δηλαδή την ανισοτροπική διάχυση και τη μέθοδο

διόρθωσης πεδίου πόλωσης, υλοποιήσουμε κάνοντας χρήση συναρτήσεων από τις εξής βιβλιοθήκες.

Η μέθοδος διόρθωσης πεδίου πόλωσης παρέχεται σαν συνάρτηση στη βιβλιοθήκη που αναφέραμε παραπάνω, τη SimpleITK. Επεξεργαστήκαμε, οπότε, με συναρτήσεις που μας παρείχε η παραπάνω βιβλιοθήκη κάθε εικόνα ξεχωριστά και από το αρχικό σύνολο δεδομένων, λάβαμε ένα δεύτερο σύνολο στο οποίο είχε εφαρμοστεί η συγκεκριμένη μέθοδος, μέσω της συνάρτησης `N4BiasFieldCorrectionImageFilter()`.

Η ανισοτροπική διάχυση, παρέχεται και αυτή ως έτοιμη συνάρτηση στη βιβλιοθήκη `medpy`. Η τελευταία είναι μία βιβλιοθήκη, που παρέχει συναρτήσεις με σκοπό την επεξεργασία και την προεπεξεργασία εικόνων σχετικών με ιατρικές διαδικασίες, όπως είναι οι μαγνητικές τομογραφίες ή τα ραδιογραφήματα. Ανάμεσα σε πολλές συναρτήσεις, παρέχεται και η συνάρτηση `anisotropic_diffusion()`, μέσω της οποίας υλοποιείται και η αντίστοιχη μέθοδος όπως στην περίπτωση της διόρθωσης πεδίου πόλωσης.

Δηλαδή, επεξεργαζόμαστε για ακόμα μία φορά την κάθε εικόνα του αρχικού συνόλου ξεχωριστά και λαμβάνουμε σαν αποτέλεσμα ένα σύνολο, το οποίο έχει υποστεί την τελευταία μέθοδο.

Στη συνέχεια, οι δύο μέθοδοι εξομάλυνσης, όπως και η αφαίρεση εικόνων με μικρό δείκτη πληροφορίας εφαρμόστηκαν μέσω συναρτήσεων, οι οποίες υλοποιήθηκαν στα πλαίσια της παρούσας διπλωματικής. Δηλαδή, λαβαμε κάθε εικόνα σε μορφή πίνακα (`numpy array`) και στη συνέχεια επεξεργαστήκαμε τις τιμές των πινάκων με σκοπό την εφαρμογή συγκεκριμένων διαδικασιών.

Στην περίπτωση της εξομάλυνσης ολικού μέσου όρου υπολογίσαμε τις τιμές μέσου όρου για όλα τα πίζελ, ενώ ταυτόχρονα ανά εικόνα λάβαμε και τη μέγιστη τιμή. Στη συνέχεια υπολογίσαμε τους μέσους όρους των παραπάνω τιμών, δηλαδή το μέσο όρο των μεγίστων και το μέσο όρο των μέσων όρων, αξιοποιώντας αριθμητικές μεθόδους. Τελικά, στα πίζελ όλων των εικόνων εφαρμόσαμε την παραπάνω συνάρτηση προεπεξεργασίας, αφαιρώντας το μέσο όρο των μέσων όρων και διαιρώντας με το μέσο όρο των μεγίστων σε κάθε τιμή πίζελ.

Το ίδιο συνέβη και στην άλλη μέθοδο εξομάλυνσης, με τη μοναδική διαφορά ότι ο μέσος όρος κάθε εικόνας μοναδικά και όχι στο σύνολο των δεδομένων είναι δυνατόν να υπολογιστεί από τη συνάρτηση `ImageDataGenerator()`. Ακόμη, η διαίρεση συνέβη μέσω της παραμέτρου `rescale`, η οποία σε κάθε περίπτωση είχε λόγω του μέγιστου πίζελ τιμή 255.

Ακόμη, όσον αφορά την αφαίρεση εικόνων με μηδενικό ή μικρό δείκτη πληροφορίας, επεξεργαστήκαμε κάθε εικόνα σαν ένα δυσδιάστατο πίνακα με τιμές και εφαρμόσαμε τη συνθήκη εάν κάτω από το 20% των πίζελ της είναι μηδενικά, τότε η εικόνα να αφαιρείται.

Τέλος, αξίζει να σημειωθεί ότι δοκιμάστηκαν και συνδυασμοί των παραπάνω μεθόδων, κυρίως μία εκ των δύο εξομαλύνσεων σε συνδυασμό με τη μέθοδο διόρθωσης πεδίου πόλωσης ή τη μέθοδο ανισοτροπικής διάχυσης. Σε αυτές τις περιπτώσεις, πρώτα λάβαμε το σύνολο εικόνων, στο οποίο είχε εφαρμοστεί η εκάστοτε μέθοδος και στη συνέχεια υπολογίσαμε τις μέσες τιμές των πίζελ των εικόνων και αυτό διότι οι δύο παραπάνω μέθοδοι προεπεξεργασίας αλλάζουν τις αριθμητικές τιμές των πίζελ.

#### 4.4.4 Εισαγωγή δεδομένων στο σύστημα

Εκτός των παραπάνω, μεγάλη σημασία για την εκπαίδευση έχει και ο τρόπος με τον οποίο εισάγονται οι εικόνες στο σύστημα. Η εισαγωγή καθίσταται αναγκαίο να συμβαίνει με μη επαναλαμβανόμενο τρόπο, δηλαδή οι εικόνες να μη δίνονται στο σύστημα με την ίδια σειρά κάθε εποχή, διότι σε αυτή την περίπτωση είναι πιο πιθανό να καταλήξουμε σε τοπικό και όχι ολικό ελάχιστο της συνάρτησης σφάλματος.

Το παραπάνω συμβαίνει, διότι η το σύστημα αντιδρά με τον ίδιο σχεδόν τρόπο όταν αναγνωρίζει εικόνες με την ίδια σειρά, οπότε συγκλίνει και πιο γρήγορα, αλλά σχεδόν ποτέ σε ολικό ελάχιστο, κυρίως σε προβλήματα σαν το δικό μας, τα οποία χαρακτηρίζονται από μεγάλο αριθμό παραμέτρων.

Αλλάζοντας τη σειρά των εικόνων, «εξαναγκάζουμε» το σύστημα να αναζητήσει περισσότερα τοπικά ελάχιστα, διότι με αυτόν τον τρόπο τα δεδομένα εμφανίζουν μικρότερο

συσχετισμό μεταξύ τους. Η υλοποίηση του παραπάνω, στις περιπτώσεις που θα μας απασχολήσουν συμβαίνει μέσω των συναρτήσεων της βιβλιοθήκης keras, ImageDataGenerator() και flow\_from\_directory().

Οι δύο προηγούμενες είναι συναρτήσεις, οι οποίες υλοποιούν τις διαδικασίες λήψης των εικόνων, προεπεξεργασίας τους σε κάποιο βαθμό και εν τέλει εισαγωγής τους στο σύστημα.

Παρέχουν τις εξής δυνατότητες, τις οποίες και χρησιμοποιήσαμε. Αρχικά, λαμβάνουν σαν παραμέτρους το σύνολο εκπαίδευσης και σύνολο επαλήθευσης, τα οποία θα χρησιμοποιηθούν από τη συνάρτηση εκπαίδευσης. Επίσης, μέσω παραμέτρων, μας δίνουν τη δυνατότητα να ορίσουμε το πλήθος των εικόνων που θα εισάγονται στο σύστημα και αυτό θα επεξεργάζεται σε κάθε χρονική στιγμή (batch size) ή τον τύπο των παραπάνω εικόνων, ο οποίος είναι ασπρόμαυρος.

Οι εικόνες βρίσκονται σε αντίστοιχους φακέλους στη μνήμη του Η/Υ και πρακτικά οι παραπάνω συναρτήσεις υλοποιούν τη μεταφορά των εικόνων αυτών στην είσοδο του νευρωνικού δικτύου, ύστερα από την προεπεξεργασία τους. Τελικά αξίζει να αναφερθεί ότι η μοναδική λειτουργία που δεν υλοποιείται αυτόματα μέσω των δύο παραπάνω συναρτήσεων είναι η εξισορρόπηση της συχνότητας εμφάνισης των δύο διαφορετικών κατηγοριών.

Αυτό είναι, παρ'όλ'αυτά αναμενόμενο, διότι όπως αναφέραμε και σε προηγούμενη παράγραφο, το πλήθος εικόνων της πρώτης κατηγορίας (HGG) είναι πολύ μεγαλύτερο σε σχέση με αυτό της δεύτερης (LGG). Σαν αποτέλεσμα, πρέπει να βρεθεί ένας τρόπος, σύμφωνα με τον οποίο θα το σύστημα θα εκπαιδεύεται και για τις δύο παραπάνω κατηγορίες στον ίδιο βαθμό. Περσσότερα, όμως για το συγκεκριμένο θέμα θα δούμε σε επόμενη παράγραφο.

#### 4.4.5 Tensorboard

Tensorboard είναι μία εφαρμογή, μέσω της οποίας μπορούμε να «οπτικοποιήσουμε» τα αποτελέσματα της εκπαίδευσης, διευκολύνοντας έτσι την ανάλυση του μοντέλου και της εκπαίδευσης πάνω σε ένα συγκεκριμένο σύνολο δεδομένων.

Το συγκεκριμένο πρόγραμμα παρέχει πληροφορίες για τις μετρικές του συστήματος σε κάθε εποχή, το χρόνο που χρειάστηκε κάθε εποχή, γραφήματα για τη συνάρτηση σφάλματος, την ακρίβεια και για όποια άλλη μετρική ορίσουμε, ενώ μας δίνει και τη δυνατότητα να συγκρίνουμε διαφορετικά αποτελέσματα μεταξύ τους εισάγοντάς τα στις ίδιες γραφικές παραστάσεις.

Όπως αναφέρθηκε και σε προηγούμενη παράγραφο, σημαντική παράμετρος για την επίλυση του προβλήματός μας είναι η επιλογή και η χρήση μετρικών, μέσω των οποίων θα μπορούσαμε να εξάγουμε συμπεράσματα, όσον αφορά το ποιές παράμετροι έχουν μεγαλύτερη βαρύτητα στην απόδοση, ποιοί αλγόριθμοι εκμάθησης ελαχιστοποιούν τη συνάρτηση σφάλματος, ποιά αρχιτεκτονική δικτύου παρέχει υψηλότερη ακρίβεια κτλ.

Κατά τη διάρκεια της εκπαίδευσης, λοιπόν, κρίνεται απαραίτητο με κάποιο τρόπο να «παρακολουθούμε» την πορεία των παραπάνω μετρικών για διάφορους λόγους.

Αρχικά, η εκπαίδευση, από ένα σημείο επαναλήψεων και μετά κρίνεται αναγκαίο να τερματιστεί, λόγω του φαινομένου του overfitting(φαινόμενο στο οποίο το δίκτυο αναγνωρίζει μόνο τις εικόνες του συνόλου εκπαίδευσης, αλλά χάνει την ικανότητά του για γενίκευση). Ύστερα, είναι πολύ χρήσιμο να γνωρίζουμε το χρονικό σημείο, στο οποίο οι μετρικές του συστήματος αρχίζουν να συγκλίνουν σε μία σταθερή τιμή, διότι από αυτό το σημείο και μετά η εκπαίδευση πάνω στο ίδιο σύνολο δεδομένων, δεν επιφέρει περεταίρω αποτέλεσμα. Επίσης, συστήματα με συγκεκριμένες επιλογές παραμέτρων, όπως μεγάλοι βαθμοί εκμάθησης, εμφανίζουν ταλάντωση στις τιμές των μετρικών τους και σαν αποτέλεσμα είναι απαραίτητο να τερματιστούν και να επανεκπαιδευτούν με μικρότερο ρυθμό μάθησης.

Σε τέτοιες περιπτώσεις κρίνεται απαραίτητη η διακοπή της εκπαίδευσης και η αλλαγή των παραμέτρων. Για τους παραπάνω λόγους, κατά τη διάρκεια της εκπαίδευσης κάνουμε χρήση του προγράμματος Tensorboard. Κάθε εποχή τα αποτελέσματα των μετρικών αποθηκεύονται σε ένα γράφημα, το οποίο εν συνεχεία εμφανίζει πληροφορίες για τη διαδικασία της εκπαίδευσης. Δηλαδή, μας δείχνει την τιμή των μετρικών, των αριθμό των εποχών τη

μείωση σφάλματος κτλ, βοηθώντας μας έτσι να αντλήσουμε συμπεράσματα πιο γρήγορα και να κάνουμε συγκρίσεις μεταξύ διαφόρων μοντέλων και μεθόδων προεπεξεργασίας.

#### 4.4.6 Μείωση βαθμού μάθησης σε περίπτωση σταθεροποίησης του δικτύου (Reduce learning rate on Plateau)

Σε πολλές περιπτώσεις, όπου γίνεται αντιληπτό ότι η ακρίβεια του δικτύου σταθεροποιείται, δηλαδή σε περιπτώσεις που ακόμη και να συνεχιστεί για αρκετό χρονικό διάστημα η εκπαίδευση δεν θα υπάρξει σημαντική διαφορά στην απόδοση, μειώνουμε το βαθμό εκμάθησης, έτσι ώστε το δίκτυο να οδηγηθεί σε ταχύτερη σύγκλιση.

Στα πλαίσια της παρούσας διπλωματικής εργασίας, κάναμε χρήση της συνάρτησης `reduceLROplateau`, η οποία υλοποιείται στη βιβλιοθήκη `keras`. Συγκεκριμένα, μέσω της συνάρτησης αυτής, ορίσαμε να παρακολουθείται η συνάρτηση σφάλματος και ο ρυθμός μάθησης να μειώνεται ανάλογα με την τιμή  $\sqrt{0.1}$ .

Ακόμη, ορίσαμε ο αριθμός των εποχών που θα «αναμένει» η συνάρτηση δίχως μείωση του βαθμού μάθησης, ενώ δεν παρουσιάζεται βελτίωση της απόδοσης στο δίκτυο θα είναι 10 και τέλος ο ελάχιστος ρυθμός μάθησης θα είναι  $10^{-5}$ . Μέσω των παραπάνω παραμέτρων επιτυγχάνονται γρηγορότεροι χρόνοι εκπαίδευσης, καθώς αναγνωρίζονται και αντιμετωπίζονται περιπτώσεις, στις οποίες η περαιτέρω αυτή εκπαίδευση δεν επιφέρει βελτίωση στην απόδοση.

#### 4.4.7 ModelCheckpoint

Ακόμα μία συνάρτηση, που παρέχει η βιβλιοθήκη `keras`, είναι η `ModelCheckpoint`. Η συνάρτηση αυτή μας δίνει τη δυνατότητα να αποθηκεύσουμε τις παραμέτρους του συστήματος τη στιγμή εκείνη, κατά την οποία παρατηρήθηκε η ελάχιστη τιμή σφάλματος.

Αποθηκεύει, δηλαδή, τα βάρη του δικτύου, τα οποία ελαχιστοποίησαν τη συνάρτηση σφάλματος. Το παραπάνω είναι εξαιρετικά χρήσιμο, διότι γνωρίζοντας την αρχιτεκτονική του εκάστοτε δικτύου και ενημερώνοντας μόνο τα βάρη με τις τιμές εκείνες, που απέδωσαν περισσότερο, πρακτικά μας δίνεται η δυνατότητα να ανακατασκευάσουμε, σε πολύ μικρό χρονικό διάστημα, το ίδιο δίκτυο που εκπαίδευσαμε και είχε βέλτιστο αποτέλεσμα, προκειμένου αυτό να χρησιμοποιηθεί ή και να εκπαιδευτεί περαιτέρω, σε περιπτώσεις όπου η εκπαίδευση ήταν αναγκαστικό να διακοπεί.

Επίσης, η συνάρτηση αυτή αποτελεί και μία μέθοδο αντιμετώπισης του `overfitting`, καθώς τα αποθηκευμένα βάρη δεν αλλάζουν τιμές. Σαν αποτέλεσμα, στην περίπτωση που το παραπάνω δίκτυο παρουσιάσει `overfitting`, τερματίζουμε την εκπαίδευση και απλώς έχουμε σαν λύση του προβλήματος τα αποθηκευμένα-βέλτιστα βάρη.

#### 4.4.8 Συναρτήσεις υλοποίησης δικτύων

Η βιβλιοθήκη `keras`, όπως αναφέρθηκε και σε προηγούμενη παράγραφο μεταξύ άλλων, παρέχει και συναρτήσεις υλοποίησης μοντέλων μηχανικής μάθησης και συγκεκριμένα συνελκτικών νευρωνικών δικτύων.

Στην περίπτωση της απλής αρχιτεκτονικής, που περιγράψαμε στο προηγούμενο κεφάλαιο, χρησιμοποιήσαμε συναρτήσεις, που υλοποιούν η καθεμία ένα διαφορετικό τύπο επιπέδου με παραμέτρους της επιλογής μας. Συνδέοντας συνολικά τα διαφορετικά επίπεδα υλοποιήσαμε την αρχιτεκτονική στο σύνολό της.

Όσον αφορά τις άλλες δύο αρχιτεκτονικές, για ακόμη μία φορά, υλοποιούνται σε ήδη υπάρχουσες συναρτήσεις της βιβλιοθήκης `keras`. Συγκεκριμένα, όσον αφορά την περίπτωση του `ResNet`, υπάρχει η συνάρτηση `ResNet50()`, η οποία και υλοποιεί την αντίστοιχη αρχιτεκτονική, με αποτέλεσμα εμείς να πρέπει μόνο να υλοποιήσουμε τα τελευταία πλήρως συνδεδεμένα επίπεδα, και να τα συνδέσουμε με το ήδη υπάρχον μοντέλο.

Αντίστοιχα για το VggNet, υπάρχει η συνάρτηση VGG16(). Κάνοντας κλήση αυτής με παραμέτρους όμοιες σε σχέση με αυτές που επιλέξαμε κατά την κλήση της ResNet50(), υλοποιήσαμε για και σε αυτή την περίπτωση το αντίστοιχο μοντέλο. Τέλος, για ακόμη μία φορά, υλοποιήσαμε τα τελικά πλήρως συνδεδεμένα επίπεδα ανεξάρτητα και τα συνδέσαμε με το υπόλοιπο δίκτυο.

Το σημαντικό στις δύο τελευταίες περιπτώσεις, πέρα από την απλότητα στην υλοποίησή τους που μας παρέχει το keras, είναι και η επιλογή του να ξεκινήσουμε την εκπαίδευση του συστήματος όχι με τυχαία βάρη, αλλά με αυτά του ImageNet. Περισσότερο με την τελευταία περίπτωση, όμως, θα ασχοληθούμε στην επόμενη παράγραφο.

#### 4.4.9 Εποχές, επαναλήψεις και πλήθος εικόνων ανά επανάληψη

Όσον αφορά την εκπαίδευση των νευρωνικών δικτύων, υπάρχουν τρία μεγέθη, τα οποία σχετίζονται με τον τρόπο εισαγωγής δεδομένων στο σύστημά μας και η εναλλαγή των τιμών τους μπορεί να επιφέρει διαφοροποιήσεις στις μετρικές που παρουσιάζει το μοντέλο.

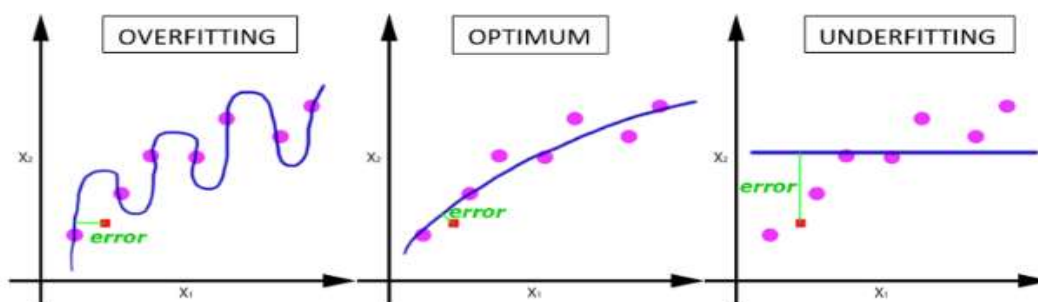
Μία εποχή (epoch), ορίζεται ως η εμπρόσθια και ανάστροφη διάδοση όλων των δειγμάτων του συνόλου εκπαίδευσης μία φορά. Δηλαδή, εάν για παράδειγμα το σύνολο εκπαίδευσης αποτελείται από 100 εικόνες, τότε μία εποχή θεωρείται όταν το σύστημα έχει «δει» και τις 100 αυτές εικόνες του συνόλου δεδομένων μία φορά. Ο αριθμός των εποχών, στον οποίον το σύστημα λαμβάνει μέγιστη απόδοση, εξαρτάται από το κάθε πρόβλημα ξεχωριστά.

Με μικρό αριθμό εποχών εμφανίζεται το φαινόμενο της υποπροσαρμογής (underfitting) [52]. Σύμφωνα με αυτό ο αριθμός των εποχών δεν αρκεί, έτσι ώστε το μοντέλο να «μάθει» να αναγνωρίζει το σύνολο δεδομένων, οπότε σαν αποτέλεσμα, δεν μπορεί να χρησιμοποιηθεί πρακτικά για κατηγοριοποίηση, λόγω μικρής απόδοσης.

Όταν το πλήθος των επαναλήψεων είναι πολύ μεγάλο, τότε συμβαίνει ένα διαφορετικό φαινόμενο, το οποίο και ονομάζουμε υπερπροσαρμογής (overfitting), το οποίο και έχουμε αναφέρει σε προηγούμενη παράγραφο. Σε αυτές τις περιπτώσεις, το μοντέλο παρουσιάζει πολύ μεγάλο ποσοστό ευστοχίας στην κατηγοριοποίηση περιπτώσεων, που παρουσιάζονται μέσα στο σύνολο δεδομένων. Παρ'όλ' αυτά δείγματα τα οποία δεν ανήκουν στο παραπάνω σύνολο και μπορεί να εισαχθούν για πρόβλεψη δεν αναγνωρίζονται.

Με άλλα λόγια το σύστημά μας έχει εκπαιδευτεί τόσο πολύ πάνω στο συγκεκριμένο σύνολο δεδομένων, το οποίο έχουμε χρησιμοποιήσει και έχει μάθει να αναγνωρίζει μόνο δείγματα αυτού. Οποιαδήποτε εικόνα δεν ανήκει στο σύνολο δεδομένων, ακόμη και αν εμφανίζει τα χαρακτηριστικά που απαιτούνται για να ανήκει σε μία από τις προς αναγνώριση κατηγορίες, το σύστημα δεν την εντάσσει στην αντίστοιχη κατηγορία. Αποτέλεσμα του τελευταίου, είναι το σύστημά να χάνει την ικανότητά του για γενίκευση και σαν αποτέλεσμα, δε δίνει λύση στο πρόβλημά.

Τελικά, καταλήγουμε, στο συμπέρασμα ότι ο αριθμός των εποχών πρέπει να μην είναι ούτε πολύ μικρός, διότι το δίκτυο δεν «μαθαίνει» τα δεδομένα, ούτε πολύ μεγάλος, καθώς θα μπορεί να αναγνωρίζει μόνο αυτά. Οπότε, το πλήθος των εποχών πρέπει να είναι ένας αριθμός ανάμεσα στις δύο παραπάνω ακραίες περιπτώσεις.



Εικόνα 4.1: Διαγράμματα μοντέλων στις περιπτώσεις της υπερπροσαρμογής, βέλτιστης λύσης καθώς και υποπροσαρμογής



Στα πλαίσια της παρούσας διπλωματικής εργασίας θέσαμε τον αριθμό των εποχών την πλειονότητα των φορών σε 200. Λιγότερες φορές και όπου αυτό κρινόταν απαραίτητο μειώναμε τον αριθμό αυτό σε 150. Όπως θα δούμε και από σχετικά διαγράμματα το δίκτυο πολλές φορές παρουσίαζε σύγκλιση σε πολύ μικρότερο αριθμό εποχών από αυτές. Επειδή όμως λαμβάναμε τις μετρικές πάνω στο σύνολο επιβεβαίωσης (validation set), και όχι πάνω στο σύνολο εκπαίδευσης, ενώ ταυτόχρονα με τη συνάρτηση ModelCheckpoint(), αποθηκεύαμε τα βάρη του πιο αποδοτικού μοντέλου δεν υπήρχε κίνδυνος υπερπροσαρμογής. Ακόμη, επειδή ο αριθμός των εποχών ήταν αρκετός για να φανεί ότι το σύστημα συγκλίνει, αποφύγαμε έτσι και το πρόβλημα της υποπροσαρμογής (underfitting).

Όσον αφορά τον αριθμό επαναλήψεων (iterations) και το πλήθος των εικόνων ανά επανάληψη (batch size), αρχικά πρέπει να γίνει αντιληπτό ότι όταν θέλουμε να εκπαιδεύσουμε το εκάστοτε μοντέλο σε ολόκληρο το σύνολο των δεδομένων εκπαίδευσης, τότε τα δύο παραπάνω μεγέθη είναι αντιστρόφως ανάλογα και συνδέονται με τη σχέση:  $batch\_size \times iterations = number\_samples$ . Συγκεκριμένα, με τον όρο `batch_size` ορίζουμε το πλήθος εικόνων ή γενικότερα των δειγμάτων που θα εισάγουμε στο δίκτυο ανά επανάληψη. Πολλαπλασιάζοντας με το πλήθος των επαναλήψεων, ο οποίος έχει οριστεί από πριν λαμβάνουμε σαν αποτέλεσμα το πλήθος των εικόνων. Το τελευταίο σημαίνει ότι στο δίκτυο, τελικά, εμφανίζονται ανά εποχή οι εικόνες στο σύνολό τους.

Στα πλαίσια της παρούσας διπλωματικής εργασίας έχουμε χρησιμοποιήσει διαφορετικά `batch_size` για διαφορετικά μοντέλα, ενώ σε κάθε περίπτωση, επειδή θέλουμε το δίκτυο να εκπαιδευτεί στο σύνολο των δεδομένων, έχουμε θέσει τον αριθμό των επαναλήψεων σαν:

$$iterations = \frac{number\_samples}{batch\_size}$$

Αξίζει τελικά να αναφερθεί ότι όσο αυξάνεται το `batch_size` μειώνεται ο αριθμός των εποχών και επιτυγχάνονται, σαν αποτέλεσμα, μικρότεροι χρόνοι εκπαίδευσης, αλλά επηρεάζεται και η απόδοση.

#### 4.4.10 Αρχικοποίηση με βάρη κατά ImageNet (ImageNet initialization)

Χρησιμοποιώντας τον όρο ImageNet [53], εννοούμε μία μεγάλη βάση δεδομένων, η οποία έχει σχεδιαστεί για χρήση λογισμικού οπτικής αναγνώρισης εικόνων. Η βάση αυτή περιέχει πολλούς είδους κατηγορίες εικόνων και αντίστοιχα εικόνες για εκπαίδευση συνελκτικών νευρωνικών δικτύων σε αυτές τις κατηγορίες.

Σε αυτή τη βάση δεδομένων, μεταξύ πολλών χρήσιμων εργαλείων μηχανικής μάθησης, υπάρχουν βάρη για πολλά είδη νευρωνικών δικτύων, τα οποία έχουν προκύψει από εκπαίδευση αυτών των δικτύων προς αναγνώριση πολυάριθμων κατηγοριών. Σαν αποτέλεσμα, μπορούμε να χρησιμοποιήσουμε τα βάρη αυτά σε δίκτυα της επιλογής μας, αντί να τα αρχικοποιήσουμε τυχαία.

Όπως αναφέραμε και πριν, μία σημαντική ιδιότητα των δύο συναρτήσεων που υλοποιούν τις δύο αρχιτεκτονικές, που αναφέρθηκαν και παραπάνω (VggNet και ResNet) είναι ότι μας δίνουν τη δυνατότητα, κατά την κλήση τους να αρχικοποιήσουμε τις τιμές των βαρών τους όχι με τυχαίες μεταβλητές, αλλά με τα βάρη του ImageNet.

Στην πράξη, η εκπαίδευση συνελκτικών νευρωνικών δικτύων δε συμβαίνει σε όλα τα επίπεδα, διότι υπάρχει κίνδυνος να σπαταληθεί μεγάλο χρονικό διάστημα, προκειμένου να καταλήξουμε σε αποτέλεσμα, το οποίο δεν είναι επιθυμητό, διότι δεν παρουσιάζει αρκετά υψηλή απόδοση.

Αντί, λοιπόν, να εκπαιδεύσουμε το σύστημά μας από την αρχή με τυχαία βάρη, αρχικοποιούμε τα τελευταία με τις τιμές ενός άλλου συνελκτικού νευρωνικού δικτύου, ίδιας

αρχιτεκτονικής, το οποίο έχει εκπαιδευτεί σε ένα πολύ μεγάλο σύνολο δεδομένων, με σκοπό να αναγνωρίζει αντικείμενα, που ανήκουν σε πολλές κατηγορίες.

Προφανώς, εμείς δε χρειαζόμαστε όλο το παραπάνω δίκτυο, διότι το σύστημα μας δεν επιδιώκουμε να αναγνωρίζει αντικείμενα μεγάλου αριθμού κατηγοριών. Παρ'όλ'αυτά τα βάρη του παραπάνω συστήματος, επειδή θα έχουν προκύψει από εκπαίδευση πάνω σε μεγάλο αριθμό κατηγοριών, υλοποιούν φίλτρα, τα οποία είναι χρήσιμα προς αναγνώριση ενός μεγάλου αριθμού αντικειμένων.

Σαν αποτέλεσμα, αρχικοποιώντας με τα αντίστοιχα βάρη, το δίκτυό μας έχει εξ'αρχής «μάθει», πολύ χρήσιμα φίλτρα, μέσω των οποίων μπορεί να «συνεχίσει» την εκπαίδευσή του, παρουσιάζοντας με αυτόν τον τρόπο μεγαλύτερη απόδοση σε μικρότερο αριθμό εποχών.

#### 4.4.11 Λίστα συχνότητας για κάθε κατηγορία (class weight dictionary)

Πολλές φορές σε διάφορα προβλήματα μηχανικής μάθησης και συγκεκριμένα κατά τη χρήση συνελκτικών νευρωνικών δικτύων παρουσιάζεται μεγάλη διαφορά στη συχνότητα εμφάνισης παραδειγμάτων που ανήκουν σε διαφορετικές κατηγορίες.

Για παράδειγμα, έστω ένα σύστημα, το οποίο θέλουμε να εκπαιδύσουμε στο να αναγνωρίζει αυτοκίνητα και φορτηγά, χρησιμοποιώντας ένα σύνολο από 100 διαφορετικές εικόνες, εκ των οποίων οι 90 ανήκουν στην κατηγορία «αυτοκίνητο» και οι υπόλοιπες 10 στην κατηγορία «φορτηγό». Σε αυτή την περίπτωση, για το σύστημά μας, μετά από πολλές επαναλήψεις και εποχές ενδέχεται ο εξής βασικός κίνδυνος. Το σύστημα, καταλήγει να κατηγοριοποιεί όλες τις εισόδους στην κατηγορία «αμάξι», διότι έχει λάβει πολλά περισσότερα παραδείγματα αυτής της κατηγορίας.

Συγκεκριμένα, σε περιπτώσεις όπου η μία προς αγνώριση κλάση εμφανίζει πολύ μεγαλύτερη συχνότητα, επειδή επιλέγεται και με αντίστοιχα αυξημένη συχνότητα κατά την εκπαίδευση, η συνάρτηση σφάλματος καταλήγει σε τοπικό ελάχιστο για το πρόβλημα, δίνοντας θεωρητικά μία βέλτιστη λύση.

Πρακτικά, όμως, επειδή η επιλογή αυτή συμβαίνει ανεξάρτητα από την αναγνώριση χαρακτηριστικών, η ίδια η κατηγοριοποίηση των κλάσεων καταλήγει να συμβαίνει δίχως την αναγνώριση των χαρακτηριστικών εκείνων, που εντάσσουν την εικόνα στην εκάστοτε κλάση. Σαν αποτέλεσμα, το σύστημα δε «μαθαίνει» να ξεχωρίζει βάση χαρακτηριστικών, αλλά επιλέγει συνέχεια την ίδια απάντηση, διότι με αυτόν τον τρόπο μειώνεται η τιμή της συνάρτησης σφάλματος.

Το παραπάνω πρόβλημα συναντάμε και στην περίπτωση μας, καθώς οι 90.000 εικόνες ανήκουν σε ασθενείς με υψηλού βαθμού γλοιώμα, ενώ μόλις οι 18.000 ανήκουν σε ασθενείς με χαμηλού βαθμού γλοιώμα. Η αναλογία είναι, δηλαδή, ένα προς πέντε.

Προκειμένου να αντιμετωπιστεί το παραπάνω, κάναμε χρήση της επιλογής `class_weight_dictionary`, η οποία είναι παράμετρος της συνάρτησης `train`, η οποία υλοποιεί την εκπαίδευση του δικτύου. Συγκεκριμένα, η παράμετρος αυτή λαμβάνει μία τιμή για κάθε κλάση και στη συνέχεια κάθε φορά που ένα παράδειγμα επηρεάζει τη συνάρτηση σφάλματος, η μεταβολή λαμβάνει υπ'όψην και την τιμή που έχει δοθεί για κάθε αντίστοιχη κλάση, συμπεριλαμβάνοντας με αυτόν τον τρόπο και τη συχνότητα εμφάνισης.

Με αυτόν τον τρόπο είναι εμφανές ότι ακόμη και αν το σύνολο δεδομένων εκπαίδευσης παρουσιάζει μεγάλη διαφορά ως προς τη συχνότητα κάθε κατηγορίας, άρα σαν αποτέλεσμα το δίκτυο θα καταλήξει να επεξεργαστεί πολλές παραπάνω περιπτώσεις που αφορούν τη μία κλάση, επειδή η μεταβολή στη συνάρτηση σφάλματος θα συμβεί ανάλογα με ένα συγκεκριμένο βάρος, τότε το σύστημα μπορεί να παραμείνει ανεπηρέαστο από την ανομοιογένεια αυτή του συνόλου δεδομένων.

Στην περίπτωσή μας, οπότε, θέτουμε σαν τιμή της λιγότερο συχνά εμφανιζόμενης κατηγορίας τον αριθμό 3 και σαν τιμή πιο συχνά εμφανιζόμενης κατηγορίας τον αριθμό 0.6. Η αναλογία ανάμεσα στις δύο αυτές κατηγορίες, είναι πέντε προς ένα, ίδια δηλαδή με την αναλογία συχνότητων των δύο κατηγοριών.

Επιλέγοντας τιμή για την πρώτη κλάση, η οποία ισούται με τη συχνότητα εμφάνισης της δεύτερης κλάσης και το αντίστροφο, στο πρόβλημά μας, αλλά και σε κάθε πρόβλημα

δυναδικής ταξινόμησης, καταφέρνουμε να εκμηδενίσουμε τις επιδράσεις των διαφορετικών συχνοτήτων κάθε κατηγορίας.

## 4.5 Transfer Learning (Μεταφερόμενη Μάθηση)

Με τον όρο μεταδιδόμενη μάθηση, εννοούμε το ερευνητικό πρόβλημα στον τομέα της μηχανικής μάθησης, που αποσκοπεί στην εφαρμογή της «γνώσης» που συσσωρεύεται κατά τη διαδικασία επίλυσης ενός προβλήματος, σε ένα δεύτερο προς επίλυση πρόβλημα, το οποίο όμως πρέπει να παρουσιάζει κάποιες ομοιότητες σε σχέση με το πρώτο.

Για παράδειγμα, όλη η «γνώση», η οποία συσσωρεύεται στο σύστημα, όταν εκπαιδευτεί με σκοπό να αναγνωρίζει αυτοκίνητα, μπορεί να χρησιμοποιηθεί σε ένα άλλο σύστημα, έτσι ώστε το δεύτερο να «μάθει» να αναγνωρίζει φορτηγά. Όπως προαναφέραμε, φαίνεται ότι οι δύο κατηγορίες, οι οποίες συσχέτισαμε παρουσιάζουν πολλά κοινά οπτικά χαρακτηριστικά.

Στην περίπτωση μας, αποσκοπούμε στο να μεταφέρουμε ήδη «υπάρχουσα» γνώση από τα βάρη που μας παρέχονται μέσω ImageNet στο εκάστοτε δικό μας σύστημα, υπό το σκεπτικό ότι στα βάρη του ImageNet υλοποιούνται φίλτρα για την εύρεση πολλών κατηγοριών και διαφορετικών μεταξύ τους σχημάτων, οπότε σε σχέση με τα τυχαία αρχικοποιημένα βάρη θεωρούμε πως το εκάστοτε προς σύγκριση σύστημά μας, θα παρουσιάσει υψηλότερες μετρικές. Το τελευταίο θα το αναλύσουμε και στη συνέχεια, παρουσιάζοντας και αντίστοιχες μετρικές.

### 4.5.1 Fine-Tuning

[54] Με τον όρο συντονισμό ενός δικτύου (fine-tuning), εννοούμε τη διαδικασία σύμφωνα με την οποία επεξεργαζόμαστε ένα σύστημα που έχει εκπαιδευτεί να εκτελεί μία συγκεκριμένη εργασία και το ρυθμίζουμε με τέτοιο τρόπο, ώστε να εκτελέσει μία διαφορετική, αλλά παρόμοια διαδικασία, σε σχέση με την πρώτη.

Θεωρώντας ότι η καινούρια προς εκτέλεση διεργασία είναι όμοια με την παλιά, η χρήση ενός δικτύου, το οποίο έχει ήδη σχεδιαστεί και εκπαιδευτεί, μας επιτρέπει να εκμεταλλευτούμε την εξαγωγή χαρακτηριστικών, η οποία λαμβάνει χώρα στα συνελκτικά επίπεδα του δικτύου. Λόγω αυτού, δεν απαιτείται η ανάπτυξη της εξαγωγής χαρακτηριστικών, εκπαιδευοντάς το δίκτυο από την αρχή.

Σύμφωνα με αυτή τη διαδικασία συμβαίνουν τα εξής:

- Αντικαθίσταται το επίπεδο εξόδου, το οποίο αρχικά είχε εκπαιδευτεί να αναγνωρίζει ένα διαφορετικό αριθμό από κατηγορίες, με αντίστοιχο επίπεδο εξόδου, το οποίο όμως αναγνωρίζει τον αριθμό των κατηγοριών που ανήκουν στο πρόβλημα.
- Το παραπάνω νέο επίπεδο εξόδου, το οποίο εισάγεται στο μοντέλο, στη συνέχεια εκπαιδευτεί, με σκοπό την πρόσληψη των χαμηλότερων σε πολυπλοκότητα χαρακτηριστικών από τα κατώτερα επίπεδα και τη δημιουργία της αντιστοίχισης ανάμεσα στα χαρακτηριστικά αυτά και τις προς αναγνώριση κατηγορίες. Η εκπαίδευση συνήθως υλοποιείται μέσω του αλγορίθμου SGD.
- Όταν συμβεί το παραπάνω, και άλλα υψηλά επίπεδα εκτός από το τελευταίο (επίπεδο εξόδου) μπορούν να εκπαιδευτούν με σκοπό το σύστημα να επιτύχει υψηλότερη απόδοση, προσαρμόζοντας τα βάρη του με τέτοιο τρόπο ώστε να ανταποκρίνονται καλύτερα στις ανάγκες του δεύτερου προβλήματος.

#### 4.5.1.1 Πρόβλημα αναγνώρισης εικόνων γάτας ή σκύλου (Cats vs Dogs)

Ένα από τα πιο γνωστά παραδείγματα, όσον αφορά τη διαδικασία του fine-tuning, είναι αυτό του προβλήματος αναγνώρισης γάτας ή σκύλου.

- Το αρχικό προς επίλυση πρόβλημα ήταν η κατηγοριοποίηση εικόνων σε 1000 κατηγορίες.
- Το καινούριο πρόβλημα ήταν η ταξινόμηση σε δύο επιμέρους κατηγορίες (γάτες και σκύλοι)
- Από τον ορισμό του fine-tuning, το τελευταίο επίπεδο αφαιρέθηκε και στη θέση αυτού προστέθηκε στο σύστημα καινούριο επίπεδο εξόδου που αναγνώριζε αυτή τη φορά μόνο 2 κατηγορίες και εκπαιδεύτηκε από μόνο του.
- Κανένα από τα υπόλοιπα επίπεδα δεν εκπαιδεύτηκε περαιτέρω, διότι όλα είχαν ήδη υποστεί εκπαίδευση, με σκοπό την αναγνώριση των 1000 κατηγοριών του πρώτου προβλήματος.

## 4.6 Παράμετροι Εκπαίδευσης

Όσον αφορά την εκπαίδευση των μοντέλων, με τα οποία θα ασχοληθούμε στα πλαίσια της παρούσας διπλωματικής ισχύουν τα εξής:

- Στα περισσότερα μοντέλα, τα βάρη δεν έχουν αρχικοποιηθεί με τυχαίο τρόπο, αλλά έχει χρησιμοποιηθεί αρχικοποίηση “ImageNet”, σύμφωνα με την οποία όπως αναφέραμε και πριν τα βάρη έχουν εκπαιδευτεί στην εύρεση 1000 κατηγοριών. Οι περιπτώσεις που έχουμε χρησιμοποιήσει τυχαία βάρη θα αναφερθούν αλλά είναι μικρότερες σε συχνότητα.
- Από τις παραπάνω περιπτώσεις εκπαίδευσης, σε αυτές, δηλαδή, όπου τα βάρη των μοντέλων δεν αρχικοποιούνται τυχαία, δοκιμάσαμε να «παγώσουμε» όλα τα προηγούμενα επίπεδα εκτός από την έξοδο και ένα μικρό αριθμό επιπέδων πριν από αυτή. Στην πλειονότητα των περιπτώσεων, εκπαιδεύτηκαν μόνο τα πέντε τελευταία επίπεδα από το τέλος, ενώ παράλληλα πειραματιστήκαμε και με διαφορετικό αριθμό «παγωμένων» επιπέδων.
- Στην πλειονότητα των περιπτώσεων χρησιμοποιήσαμε τον αλγόριθμο μάθησης Adam, ενώ σε άλλες πειραματιστήκαμε και με τον SGD. Ο αρχικός ρυθμός μάθησης στο σύνολο των διαφορετικών εκπαίδευσεων παρέμεινε σταθερός, ενώ επηρρεάζοταν μόνο από τη συνάρτηση `ReduceLROnPlateau()`, προκειμένου το εκάστοτε μοντέλο να συγκλίνει.
- Ο αριθμός των εποχών παρέμεινε σταθερός και ίσος με 200 για την πλειοψηφία των εκπαίδευσεων. Αυτό, παρ’όλ’αυτά, όπως προαναφέραμε δεν ήταν καίριας σημασίας, καθώς αποθηκεύαμε κάθε φορά το μοντέλο εκείνο με την υψηλότερη απόδοση πάνω στο σύνολο επαλήθευσης. Σύγκλιση όπως θα φανεί και στο επόμενο κεφάλαιο με τα αποτελέσματα συνέβαινε συνήθως πριν την 100<sup>η</sup> εποχή.
- Το `batch_size`, ανά εκπαίδευση και μοντέλο διέφερε. Στην πλειονότητα των περιπτώσεων ήταν 128. Όταν αυτό κρινόταν αδύνατο λόγω έλλειψης μνήμης, μειώναμε την τιμή του και αυξάναμε ταυτόχρονα το πλήθος των επαναλήψεων, με σκοπό να μείνει σταθερός ο αριθμός των εικόνων και ίσος με 108.000. Δοκιμάσαμε οπότε και άλλες τιμές όπως 256 ή 64 για το `batch_size`, τα αποτελέσματα των οποίων επηρέασαν κυρίως το χρόνο σύγκλισης.
- Όσον αφορά, τις μεθόδους προεπεξεργασίας, δοκιμάσαμε συνδυαστικά τη μέθοδο διόρθωσης πεδίου πόλωσης μαζί με την ανιστροπική διάχυση δύο φορές. Στις υπόλοιπες περιπτώσεις, δοκιμάστηκαν συνδυαστικά η μέθοδος ανιστροπικής διάχυσης μαζί με τις δύο περιπτώσεις εξομάλυνσης και η μέθοδος διόρθωσης πεδίου πόλωσης μόνο με την εξομάλυνση ολικού μέσου όρου, η οποία στηρίζεται στην εξαγωγή των μέσων και εφαρμογή τους πάνω στο σύνολο των πίξελ. Επίσης, πειραματιστήκαμε και με τις μεθόδους εξομάλυνσης αποκλειστικά πάνω στο αρχικό σύνολο των δεδομένων, χωρίς δηλαδή το τελευταίο να έχει υποστεί μέθοδο διόρθωσης πεδίου πόλωσης ή ανιστροπική διάχυση. Τελικά, όπως αναφέραμε και σε

προηγούμενη παράγραφο, σε όλες τις περιπτώσεις εκπαίδευσης αφαιρέσαμε τις εικόνες με μηδενικό ή μικρό ποσοστό πληροφορίας.

Οι παραπάνω επιλογές υπερπαραμέτρων και μεθόδων αναφέρθηκαν αναξάρτητα από τα μοντέλα μηχανικής μάθησης που χρησιμοποιήθηκαν. Στη συνέχεια του κεφαλαίου, λοιπόν, θα εξετάσουμε τις επιλογές των παραμέτρων συναρτήσει των μοντέλων και θα εξηγήσουμε τις επιλογές αυτές.

#### 4.6.1 ResNet50

Οι περισσότερες περιπτώσεις εκπαιδεύσεων, αφορούν το μοντέλο ResNet50. Ο λόγος, για τον οποίο συνέβη το παραπάνω είναι απλούστατα, διότι η εκπαίδευση του τελευταίου σε σχέση με το μοντέλο VGG16, είναι πολύ πιο εύκολη και απαιτεί πολύ λιγότερο χρόνο. Το ResNet50, αποτελείται από 25.5 εκατομμύρια παραμέτρους, σε αντίθεση με το VGG16, το οποίο αποτελείται από 138 εκατομμύρια παραμέτρους. Παρ'ότι, λοιπόν, το μοντέλο ResNet50, φαίνεται να έχει πιο περίπλοκη αρχιτεκτονική από αυτό του VGG16, κυρίως λόγω των υπολοιπόμενων συνδέσεων, η εκπαίδευσή του είναι σαφώς πιο εύκολη.

Λαμβάνοντας υπ'όψην το παραπάνω, ένας επιπρόσθετος λόγος είναι ότι στην περίπτωση του ResNet50, η εναπομείνουσα μνήμη, που δε δεσμεύεται από τις παραμέτρους του μοντέλου, μπορεί να χρησιμοποιηθεί στην αύξηση του `batch_size`, προκειμένου να επιτευχθούν μικρότεροι χρόνοι σύγκλισης.

Ακόμη, ένας επιπρόσθετος λόγος είναι και η δυσκολία στην εκπαίδευση του VGG16. Σε πολλές περιπτώσεις, που θα αναλύσουμε στο επόμενο κεφάλαιο, το τελευταίο μοντέλο, δε σύγκλινε σε λύση του προβλήματος, αλλά προέβλεπε συνέχεια την ίδια κατηγορία. Παρ'ότι είχαμε αντιμετωπίσει αντίστοιχο πρόβλημα στην περίπτωση του ResNet, το πρόβλημα αυτό λύθηκε κάνοντας χρήση της επιλογής `class_weight_dictionary`, η οποία αναφέρθηκε παραπάνω. Στην περίπτωση του VGG16, όμως το πρόβλημα παρέμεινε με αποτέλεσμα το μοντέλο πρακτικά να μην μπορεί να χρησιμοποιηθεί.

Συνολικά, λοιπόν, το ResNet50, είναι το μοντέλο εκείνο, το οποίο χρησιμοποιήσαμε στην πλειονότητα των εκπαιδεύσεων. Χρησιμοποιήθηκε σε συνδυασμό με τη μέθοδο ανισοτροπικής διάχυσης, χωρίς εξομάλυνση, με εξομάλυνση τοπικού μέσου όρου, αλλά και με εξομάλυνση ολικού μέσου όρου. Σε συνδυασμό με τη μέθοδο διόρθωσης πεδίου πόλωσης χρησιμοποιήθηκε μία φορά και αυτή μαζί με εξομάλυνση ολικού μέσου όρου. Μόνο με εξομάλυνση και χωρίς άλλη μέθοδο προεπεξεργασίας, εκπαιδεύσαμε ένα μοντέλο τύπου ResNet50, ενώ χωρίς καμία από τους παραπάνω μεθόδους εκπαιδεύτηκε ένα μοντέλο, προκειμένου να φανεί η αξία των παραπάνω τεχνικών. Και στις δύο περιπτώσεις έγινε χρήση του αλγόριθμου μάθησης Adam. Τελικά, εκπαιδεύτηκαν δύο μοντέλα τέτοιου τύπου σε συνδυασμό με εξομάλυνση ολικού μέσου όρου, αλλά χρησιμοποιώντας ταυτόχρονα και τις μεθόδους διόρθωσης πεδίου πόλωσης και ανισοτροπικής διάχυσης ταυτόχρονα. Στο ένα μοντέλο έγινε χρήση του αλγόριθμου μάθησης SGD και στο άλλο Adam.

Όσον αφορά τις υπόλοιπες υπερπαραμέτρους χρησιμοποιήσαμε δοκιμάσαμε και το «πάγωμα» διαφορετικών επιπέδων, προκειμένου να εξάγουμε συμπεράσματα για την απόδοση σε σχέση με τον αριθμό των επιπέδων που αποδίδουν περισσότερο στη διαδικασία του fine tuning.

#### 4.6.2 VGGNet16

Όπως αναφέρθηκε και στην προηγούμενη παράγραφο, το συγκεκριμένο μοντέλο παρουσίασε αρκετές δυσκολίες στην εκπαίδευση, όσον αφορά το συγκεκριμένο πρόβλημά δυαδικής ταξινόμησης. Αρχικά, όπως είπαμε και πριν οι παράμετροί του δικτύου αυτού, δεσμεύουν σε μεγάλο βαθμό τη μνήμη του συστήματος. Σαν αποτέλεσμα δεν είναι ιδιαίτερα εύκολο να πειραματιστούμε κάνοντας χρήση του VGG16 σε μεγάλα `batch_size`.

Το χρησιμοποιήσαμε, παρ'όλ'αυτά σε συνδυασμό με όλες τις μεθόδους προεπεξεργασίας. Όσον αφορά την ανισοτροπική διάχυση, χρησιμοποιήθηκε δύο φορές. Τη μία φορά με χρήση του αλγόριθμου μάθησης SGD, αντί για Adam και τη δεύτερη φορά με τον αλγόριθμο Adam. Σε συνδυασμό με τη μέθοδο διόρθωσης πεδίου πόλωσης, χρησιμοποιήθηκε μία φορά με εξομάλυνση ολικού μέσου όρου.

Αρκετές φορές πειραματιστήκαμε με εναλλαγή του αριθμού «παγωμένων» επιπέδων. Στην πλειονότητα των περιπτώσεων, όπως αναφέραμε και σε προηγούμενη παράγραφο, επιλέγουμε να «παγώσουμε» όλα τα επίπεδα του δικτύου εκτός των πέντε τελευταίων. Δηλαδή, του επιπέδου εξόδου, των πλήρως συνδεδεμένων επιπέδων πριν από αυτό, αλλά και των πιο «κοντινών» ως προς την έξοδο συνελκτικών επιπέδων. Στην περίπτωση όμως αυτή πειραματιστήκαμε πρώτον, με το εκπαιδεύσουμε όλα τα επίπεδα, δεύτερον με το να εκπαιδεύσουμε το επίπεδο εξόδου μαζί με τα έξι πριν από αυτό και τρίτον με την εκπαίδευση του επιπέδου εξόδου και των δεκατριών πριν από αυτό επίπεδα. Πειραματιστήκαμε ακόμα με τις δύο μεθόδους προεπεξεργασίας συνδυαστικά με την εξομάλυνση των δεδομένων. Εφαρμόσαμε δηλαδή, τη μέθοδο ανισοτροπικής διάχυσης μαζί με τη μέθοδο διόρθωσης πεδίου πόλωσης και στη συνέχεια εκπαυδεύσαμε το δίκτυο αυτό «παγώνοντας» όλα τα επίπεδα εκτός από τα πέντε τελευταία.

#### 4.6.3 Custom – Απλοποιημένη αρχιτεκτονική

Στην περίπτωση του τελευταίου μοντέλου, δηλαδή του συνελκτικού νευρωνικού δικτύου, που αναπτύξαμε στα πλαίσια της παρούσας διπλωματικής εργασίας, το οποίο εμφανίζει πολύ μικρότερο αριθμό παραμέτρων σε σχέση με τα άλλα δυο, επιλέξαμε να το εκπαιδεύσουμε με τη μέθοδο ανισοτροπικής διάχυσης, σε συνδυασμό με εξομάλυνση ολικού μέσου όρου.

Θελήσαμε να συγκρίνουμε την απόδοση ενός δικτύου, το οποίο είναι πιο εύκολο να εκπαιδευτεί, καθώς εμφανίζει πολύ μικρότερο πλήθος παραμέτρων σε σχέση με τα δύο παραπάνω, ενώ ταυτόχρονα λόγω του παραπάνω εκπαυδεύεται και σε μικρότερο χρονικό διάστημα.

Επίσης, λόγω της μικρής δέσμευσης μνήμης που απαιτεί το μοντέλο μπορέσαμε να επιλέξουμε σχετικά μεγάλο batch\_size, το οποίο επιτάχυνε ακόμα περισσότερο τη διαδικασία εκπαίδευσης. Ο αριθμός των εποχών κρατήθηκε σταθερός και ίσος με 200, ενώ σημαντική διαφορά ήταν η τυχαία αρχικοποίηση των βαρών. Δε χρησιμοποιήθηκαν, δηλαδή, τα βάρη κατά "ImageNet", στη συγκεκριμένη περίπτωση.

## Κεφάλαιο 5<sup>ο</sup> Αποτελέσματα και Συμπεράσματα

---

Σκοπός του παρόντος κεφαλαίου είναι η παρουσίαση των αποτελεσμάτων για τα μοντέλα μηχανικής μάθησης, τα οποία χρησιμοποιήθηκαν. Συγκεκριμένα, στο πρώτο μέρος του κεφαλαίου θα παρουσιάσουμε σε μορφή γραφικών τα αποτελέσματα των «εκπαιδευμένων» μοντέλων πάνω στο σύνολο επαλήθευσης και θα συγκρίνουμε τα αποτελέσματα αυτά σε συνδυασμό με τις υπερπαραμέτρους που χρησιμοποιήθηκαν για κάθε μοντέλο. Με τη σύγκριση αυτή αποσκοπούμε στο να καταλήξουμε σε συμπεράσματα για τις υπερπαραμέτρους, τις διαφορετικές αρχιτεκτονικές, αλλά και τις τεχνικές προεπεξεργασίας δεδομένων που χρησιμοποιήθηκαν.

Στο πρώτο μέρος θα παρουσιάσουμε διαγράμματα σύγκρισης μεταξύ των μοντέλων που εκπαιδεύσαμε. Δηλαδή, σε κάθε διάγραμμα θα συγκρίνουμε διαφορετικά μοντέλα μηχανικής μάθησης μεταξύ τους, διατηρώντας ίδιες παραμέτρους συνολικά εκτός από μία την οποία και θα αναφέρουμε. Σκοπός αυτού είναι να εξετάσουμε τη βαρύτητα κάθε παραμέτρου, δηλαδή να αναγνωρίσουμε πόσο σημαντικός είναι ο ρόλος της στην αντιμετώπιση του προβλήματος.

Στο δεύτερο μέρος του κεφαλαίου θα παρουσιάσουμε τις γραφικές που εξήχθησαν από την εκπαίδευση και θα σχολιάσουμε χρόνους σύγκλισης, αριθμό εποχών, αλλά και λοιπές επιδράσεις διαφορετικών παραμέτρων, όσον αφορά την εκπαίδευση.

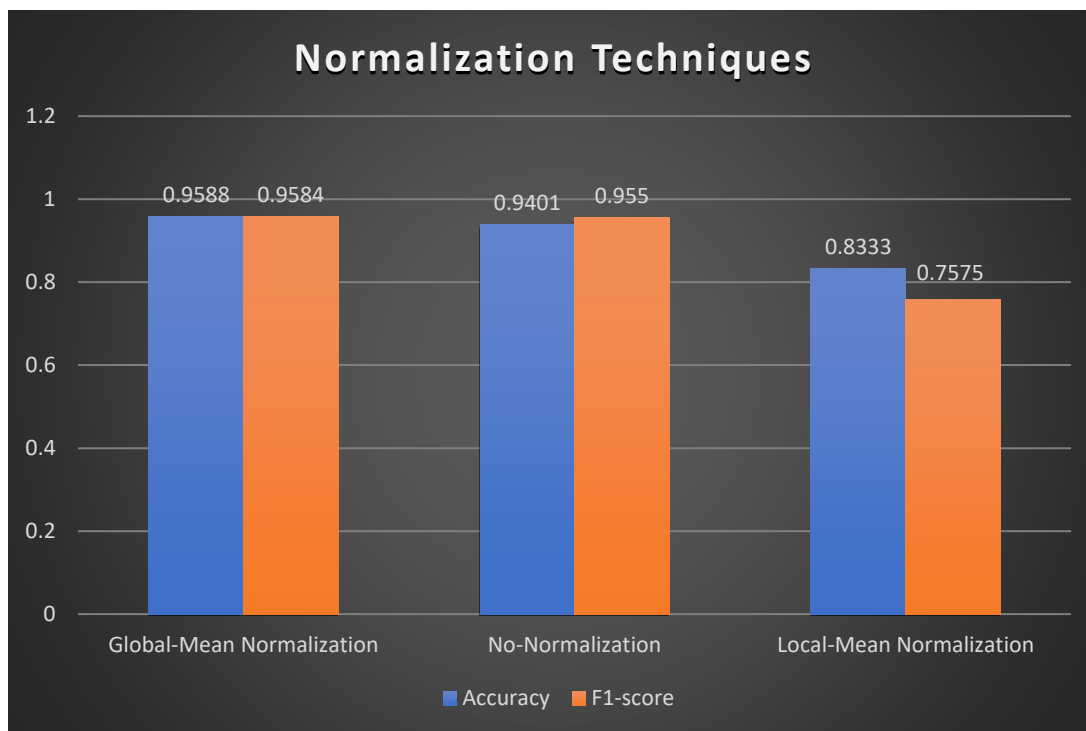
Τελικά, στο τρίτο μέρος του κεφαλαίου θα χρησιμοποιήσουμε τα αποτελέσματα και τα συμπεράσματα από τα προηγούμενα δύο μέρη και θα προσπαθήσουμε να καταλήξουμε σε γενικά συμπεράσματα, όσον αφορά το πρόβλημα δυαδικής ταξινόμησης μαγνητικών τομογραφιών σε υψηλού επιπέδου γλοιώματα και χαμηλού επιπέδου γλοιώματα.

### 5.1 Στατιστικά Αποτελέσματα

#### 5.1.1 Επιλογή μεθόδου κανονικοποίησης - εξομάλυνσης

Όπως αναφέρθηκε και στην προηγούμενη παράγραφο, ανεξάρτητα από τις μεθόδους ανιστροπικής διάχυσης και διόρθωσης πεδίου πόλωσης, σε κάθε μοντέλο χρησιμοποιήσαμε κάποια μέθοδο εξομάλυνσης δεδομένων. Σε αυτό το σημείο θα παρουσιάσουμε τα αποτελέσματα ίδιων μοντέλων, εναλλάσσοντας τις μεθόδους εξομάλυνσης, με σκοπό να δείξουμε ποιά από αυτές απέδωσε σε μεγαλύτερο βαθμό.

### Διάγραμμα Μεθόδων Εξομάλυνσης I



Εικόνα 5.1: Γράφημα διαφορετικών περιπτώσεων εξομάλυνσης δεδομένων, σε συνδυασμό με τη μέθοδο ανισοτροπικής διάχυσης.

Στο παραπάνω διάγραμμα φαίνονται τα αποτελέσματα τριών μοντέλων τύπου ResNet50, εκπαιδευμένα για τον ίδιο αριθμό εποχών, με ίδιο batch\_size, αλγόριθμο μάθησης Adam, ενώ στο σύνολο δεδομένων έχει πρώτα εφαρμοστεί μέθοδος προεπεξεργασίας ανισοτροπικής διάχυσης. Επίσης, και στις τρεις περιπτώσεις, δεν έχουν χρησιμοποιηθεί τυχαίες αρχικές τιμές για τα βάρη, αλλά έχει γίνει αρχικοποίηση “ImageNet”, ενώ ταυτόχρονα ο αριθμός των εκπαιδευσίμων επιπέδων είναι τα πέντε τελευταία επίπεδα, μαζί με το επίπεδο εξόδου. Όλα τα υπόλοιπα επίπεδα διατηρούνται ανεπιρεάστα κατά τη διαδικασία της εκπαίδευσης.

Η πρώτη από τις τρεις περιπτώσεις (Global-Mean Normalization στα αριστερά του διαγράμματος) είναι αυτή της μεθόδου εξομάλυνσης με χρήση του ολικού μέσου όρου, η οποία επιτυγχάνεται μέσω της εύρεσης των μέσων όρων των μεγίστων από όλες τις εικόνες, αλλά και τον μέσο όρο από το σύνολο των μέσων όρων κάθε εικόνας. Η δεύτερη περίπτωση είναι το μοντέλο που δεν έχει υποστεί καμία μέθοδο εξομάλυνσης (No-Normalization στο κέντρο του διαγράμματος), ενώ η τρίτη και τελευταία περίπτωση (L.Mean-Normalization στα αριστερά του διαγράμματος) αντιστοιχεί στη μέθοδο εξομάλυνσης τοπικού μέσου όρου, κατά την οποία και έχουμε αφαιρέσει από κάθε εικόνα το δικό της μέσο όρο και εν συνεχεία έχουμε διαιρέσει με τη μέγιστη που υπάρχει στο σύνολο των πίξελ για όλες τις εικόνες.

Όπως φαίνεται, λοιπόν, και από το παραπάνω διάγραμμα, την υψηλότερη απόδοση λαμβάνουμε, από τη μέθοδο εξομάλυνσης πάνω στο σύνολο των εικόνων, δηλαδή την πρώτη περίπτωση. Το παραπάνω αποτέλεσμα είναι αναμενόμενο, καθώς όπως αναφέραμε κατά τη θεωρητική ανάλυση των μεθόδων εξομάλυνσης σε προηγούμενο κεφάλαιο, μέσω των

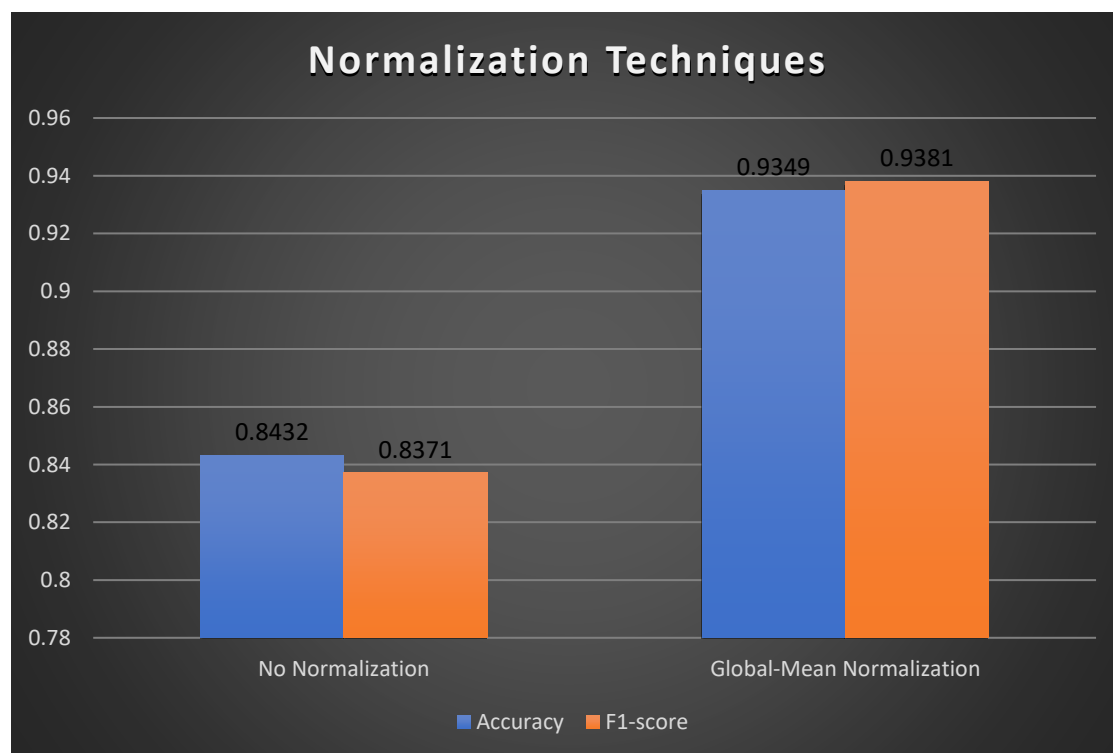


τελευταίων επιτυγάνεται εξισσορόπιση στην επίδραση των διαφορετικών χαρακτηριστικών κατά την εκπαίδευση. Η σημασία της τελευταίας μεθόδου, επίσης, φαίνεται μέσω της σύγκρισης με το αντίστοιχο διάγραμμα, το οποίο αντιστοιχεί σε μοντέλο, του οποίου τα δεδομένα εισόδου δεν έχουν υποστεί εξομάλυνση. Η διαφορά στην ακρίβεια είναι πάνω από 1%, δηλαδή παρατηρείται σημαντική διαφορά κατά τη χρήση της παραπάνω μεθόδου εξομάλυνσης.

Ακόμη, αξίζει να αναφερθεί ότι κατά τη χρήση της δεύτερης μεθόδου εξομάλυνσης (δεξιά του διαγράμματος), το αντίστοιχο μοντέλο παρουσίασε το πρόβλημα κατά το οποίο προβλέπεται μόνο μία κατηγορία, και έχουμε αναφέρει σε προηγούμενο κεφάλαιο της παρούσας διπλωματικής εργασίας. Πιθανότατα, η χρήση διαφορετικών μέσων όρων σε κάθε εικόνα, είχε σαν αποτέλεσμα αντίστοιχου τύπου χαρακτηριστικά να μην έχουν την ίδια βαρύτητα κατά την εκπαίδευση, οπότε η συνάρτηση σφάλματος δε μεγιστοποιήθηκε με τον τρόπο που θέλαμε, δηλαδή μέσω της εύρεσης κοινών χαρακτηριστικών, αλλά μέσω της πρόβλεψης μίας κατηγορίας, αυτής με τη μεγαλύτερη συχνότητα. Το τελευταίο είχε σαν αποτέλεσμα, το παραπάνω μοντέλο να καθίσταται μη ικανό για χρήση, καθώς όπως προαναφέρθηκε προβλέπει συνεχώς το ίδιο αποτέλεσμα ανεξαρτήτως των χαρακτηριστικών της εικόνας.

Τελικά, ύστερα από τη σύγκριση των παραπάνω αποτελεσμάτων επιλέγουμε στην πλειονότητα των εκπαιδεύσεων για τα υπόλοιπα μοντέλα μηχανικής μάθησης, να χρησιμοποιήσουμε την πρώτη μέθοδο εξομάλυνσης, η οποία και αντιστοιχεί σε συστήματα με υψηλότερη απόδοση, σε σχέση με αντίστοιχα μοντέλα χωρίς καμία μέθοδο εξομάλυνσης αλλά και σε συστήματα, στα οποία έχει γίνει χρήση της δεύτερης μεθόδου εξομάλυνσης.

### **Διάγραμμα Μεθόδων Εξομάλυνσης II**



Εικόνα 5.2: Γράφημα διαφορετικών περιπτώσεων εξομάλυνσης δεδομένων, δίχως εφαρμογή άλλων μεθόδων προεπεξεργασίας δεδομένων.

Προκειμένου να φανεί η αξία της εξομάλυνσης των δεδομένων στα πλαίσια του συγκεκριμένου προβλήματος, παρουσιάζουμε και το παραπάνω διάγραμμα, στο οποίο φαίνονται τα αποτελέσματα δύο μοντέλων μηχανικής μάθησης με ίδιες παραμέτρους, με τη μόνη διαφορά ανάμεσα στα δύο συστήματα ότι το ένα από αυτά έχει υποστεί εξομάλυνση κάνοντας χρήση του ολικού μέσου όρου των εικόνων, ενώ η διαφορά σε σχέση με το προηγούμενο διάγραμμα είναι ότι τα δεδομένα δεν έχουν υποστεί ούτε ανισοτροπική διάχυση ούτε διόρθωση πεδίου πόλωσης.

Συγκεκριμένα εκπαιδεύσαμε δύο μοντέλα μηχανικής μάθησης τύπου ResNet50. Η εκπαίδευση είχε τις ίδιες ακριβώς παραμέτρους εκτός την εξομάλυνση όπως αναφέρθηκε και παραπάνω. Δηλαδή, ο αριθμός εποχών ήταν ο ίδιος, ο αλγόριθμος μάθησης κοινός (“Adam” και για τις δύο περιπτώσεις), όπως και το `batch_size`. Τέλος, ο αριθμός των εκπαιδευσιμων επιπέδων ήταν ο ίδιος, δηλαδή πέντε και για τις δύο περιπτώσεις ενώ είχε χρησιμοποιηθεί αρχικοποίηση βαρών “ImageNet”.

Η πρώτη περίπτωση (αριστερά στο διάγραμμα) είναι αυτή κατά την οποία δε χρησιμοποιείται καμία μέθοδος εξομάλυνσης, ενώ στη δεύτερη περίπτωση (δεξιά στο διάγραμμα) έχει γίνει εξομάλυνση κάνοντας χρήση του ολικού μέσου όρου των εικόνων.

Τα αποτελέσματα του παραπάνω διαγράμματος συμφωνούν με το αρχικό διάγραμμα, καθώς παρατηρείται πολύ έντονη διαφορά από πλευράς μετρικών κατά τη χρήση της εξομάλυνσης πάνω στο σύνολο των δεδομένων. Επίσης, επειδή τα δεδομένα εισόδου δεν έχουν υποστεί κάποια άλλη μέθοδο προεπεξεργασίας (ανισοτροπική διάχυση ή διόρθωση πεδίου πόλωσης), αναδεικνύεται με αυτόν τον τρόπο η αξία της εξομάλυνσης, όταν συμβαίνει πάνω σε ένα σύνολο δεδομένων, το οποίο δεν έχει υποστεί καμία περαιτέρω προεπεξεργασία. Η διαφορά που παρατηρείται σε αυτή την περίπτωση ανάμεσα στις μετρικές των δύο μοντέλων είναι πολύ πιο έντονη από την αντίστοιχη διαφορά στο πρώτο διάγραμμα.

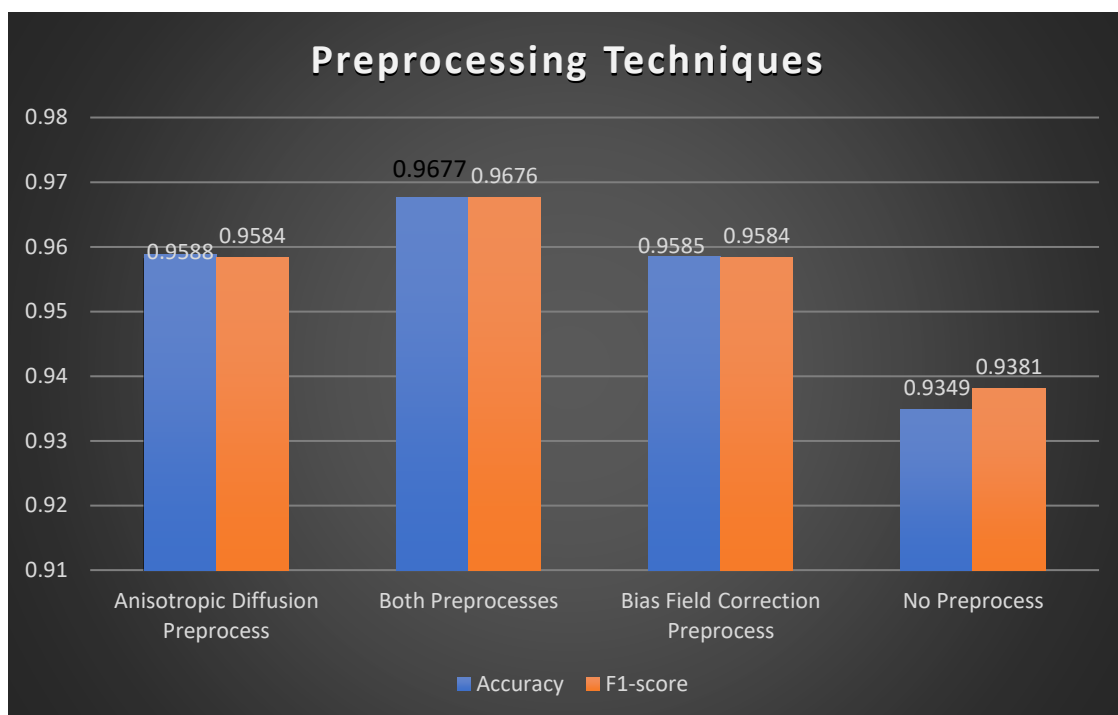
Από το παραπάνω φαίνεται ότι η απόδοση αυξάνεται σε έντονο βαθμό όταν χρησιμοποιείται έστω και μία από τις παραπάνω μεθόδους, δηλαδή εξομάλυνση, ανισοτροπική διάχυση ή διόρθωση πεδίου πόλωσης. Η διαφορά ανάμεσα στις μετρικές των δύο μοντέλων είναι 9%, πολύ εντονότερη δηλαδή από αυτή του προηγούμενου διαγράμματος, αν και στο προηγούμενο διάγραμμα παρατηρήθηκαν υψηλότερες αποδόσεις.

Τελικά, μέσω του παραπάνω διαγράμματος φαίνεται, ακόμη, πως η χρήση της μεθόδου εξομάλυνσης στο σύνολο των δεδομένων, βελτιώνει την απόδοση του συστήματος ανεξάρτητα από άλλες μεθόδους προεπεξεργασίας, γι’ αυτό και θα χρησιμοποιηθεί εν συνεχεία στην πλειονότητα των εκπαιδεύσεων, όπως αναφέραμε και πριν.

### 5.1.2 Επιλογή μεθόδου προεπεξεργασίας δεδομένων

Αφού καταλήξαμε στη χρήση της μεθόδου εξομάλυνσης στο σύνολο των δεδομένων, σημαντικό ρόλο στην αύξηση της απόδοσης του συστήματος έχει και η μέθοδος προεπεξεργασίας, που θα χρησιμοποιηθεί. Δηλαδή, πέρα από την εξομάλυνση, εάν θα χρησιμοποιηθεί ανισοτροπική διάχυση, διόρθωση πεδίου πόλωσης και οι δύο ή καμία από αυτές.

### Διάγραμμα Μεθόδων Προεπεξεργασίας



Εικόνα 5.3: Γράφημα διαφορετικών περιπτώσεων προεπεξεργασίας δεδομένων, σε συνδυασμό με εξομάλυνση ολικού μέσου.

Στο προηγούμενο διάγραμμα παρουσιάζονται τέσσερις διαφορετικοί μέθοδοι προεπεξεργασίας δεδομένων, πάνω σε μοντέλο μηχανικής μάθησης ίδιων υπερπαραμέτρων, ενώ ταυτόχρονα έχει γίνει εξομάλυνση κάνοντας χρήση του ολικού μέσου όρου των εικόνων.

Συγκεκριμένα, εκπαιδεύσαμε τέσσερα συνελκτικά νευρωνικά δίκτυα τύπου ResNet50, με όλους τους πιθανούς συνδυασμούς των μεθόδων ανισοτροπικής διάχυσης και διόρθωσης πεδίου πόλωσης. Όλα τα μοντέλα μηχανικής μάθησης, των οποίων τα αποτελέσματα παρουσιάζονται στο προηγούμενο διάγραμμα, έχουν εκπαιδευτεί με τις ίδιες υπερπαραμέτρους. Ίδιο αριθμό εποχών, batch\_size, αλλά και αλγόριθμο μάθησης “Adam”. Επίσης, έχει γίνει αρχικοποίηση “ImageNet”, κρατώντας αναλλοίωτα όλα τα επίπεδα πέρα από τα πέντε τελευταία.

Στο διάγραμμα από αριστερά προς τα δεξιά φαίνονται τα αποτελέσματα του ίδιου μοντέλου που εκπαιδεύτηκε κάθε φορά, πρώτον με δεδομένα που προέκυψαν από εφαρμογή της μεθόδου ανισοτροπικής διάχυσης, δεύτερον ανισοτροπικής διάχυσης και διόρθωσης πεδίου πόλωσης ταυτόχρονα, τρίτον με δεδομένα χωρίς καμία μέθοδο προεπεξεργασίας και τέταρτον δεδομένα που προέκυψαν κάνοντας χρήση της μεθόδου διόρθωσης πεδίου πόλωσης αποκλειστικά.

Αρχικά, όσον αφορά το διάγραμμα παρατηρούμε ότι οποιαδήποτε μέθοδο προεπεξεργασίας από τις δύο και να χρησιμοποιήσουμε ή ακόμα και συνδυασμό τους, η

απόδοση είναι σε κάθε περίπτωση υψηλότερη σε σχέση με τη χρήση καμίας μεθόδου από τις δύο παραπάνω.

Ακόμη, παρατηρούμε ότι από μόνες τους οι μέθοδοι ανισοτροπικής διάχυσης και διόρθωσης πεδίου πόλωσης παρουσιάζουν μεταξύ τους αντίστοιχες αποδόσεις και επιτυγχάνουν τουλάχιστον 2% υψηλότερη ακρίβεια σε σχέση με την περίπτωση που συμβαίνει μόνο εξομάλυνση. Όσον αφορά μόνο την ακρίβεια η μέθοδος ανισοτροπικής διάχυσης παρουσιάζει ελάχιστα υψηλότερη απόδοση της τάξης του 0.03%.

Τέλος, με τον παραπάνω συνδυασμό υπερπαραμέτρων, ο συνδυασμός και των δύο μεθόδων προεπεξεργασίας είναι αυτός που δίνει και τη βέλτιστη τιμή μετρικών. Παρατηρούμε, δηλαδή, ότι όταν εφαρμόσουμε και τις δύο παραπάνω μεθόδους στο σύνολο των δεδομένων μας, τότε η επιτυγχάνονται τιμές μετρικών, οι οποίες είναι κατά 1% υψηλότερες σε σχέση με τη χρήση της κάθε μεθόδου ξεχωριστά.

Σαν αποτέλεσμα, φαίνεται, όσον αφορά τα μέχρι στιγμής αποτελέσματα, ότι ο συνδυασμός των δύο παραπάνω μεθόδων αποδίδει περισσότερο. Όσον αφορά τον αλγόριθμο μάθησης “Adam”, εξάγεται, λοιπόν, το παραπάνω συμπέρασμα, διότι όπως θα δούμε και στη συνέχεια, όταν γίνεται χρήση του αλγόριθμου μάθησης “SGD”, δε συμβαίνει το ίδιο.

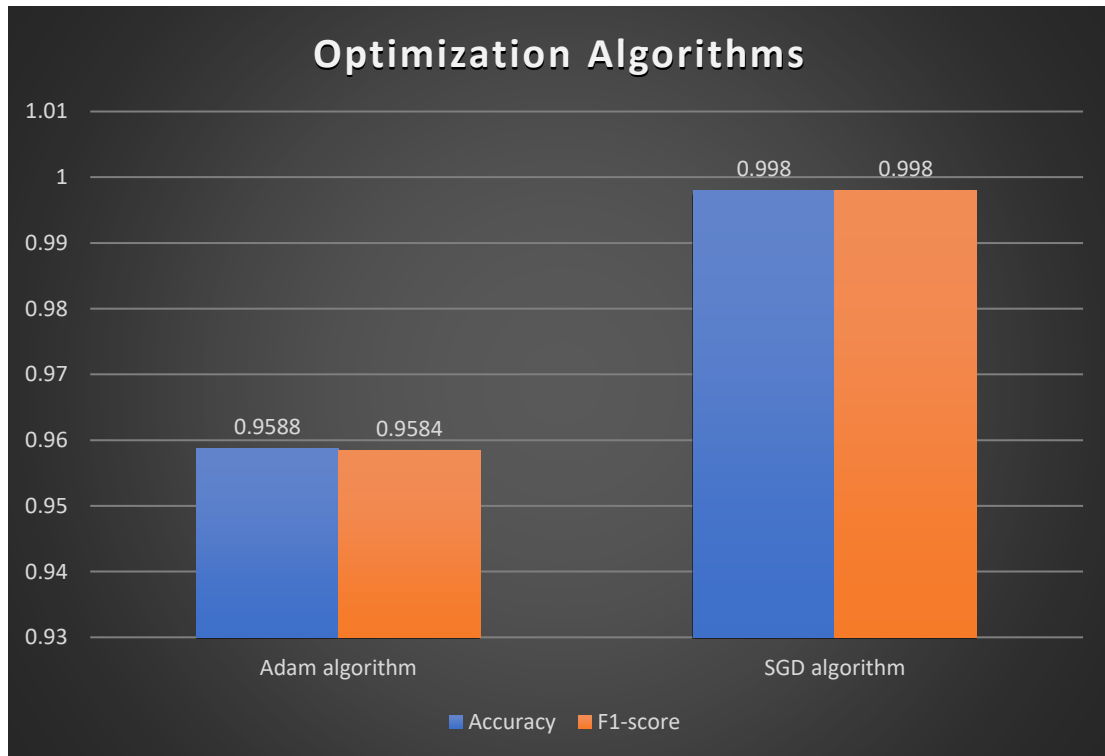
Φαίνεται, παρ’όλ’ αυτά ότι σε κάθε περίπτωση η χρήση έστω και μίας μεθόδου προεπεξεργασίας αυξάνει σημαντικά την απόδοση, οπότε στη συνέχεια των αποτελεσμάτων εκτός ορισμένων περιπτώσεων, το σύνολο των δεδομένων θα έχει υποστεί πριν την εκπαίδευση τουλάχιστον μία από τις παραπάνω μεθόδους.

Ο λόγος, επίσης, που μέχρι στιγμής έχει χρησιμοποιηθεί κατα κύριο λόγο αρχιτεκτονική τύπου ResNet50, είναι ότι παρατηρούνται, όπως θα δείξουμε και στο δεύτερο μέρος του παρόντος κεφαλαίου, μικρότερες αστοχίες κατά την εκπαίδευση, καθώς οι αρχιτεκτονική VGGNet, την πλειονότητα των φορών, είχε σαν αποτέλεσμα συστήματα που προέβλεπαν μόνο μία κατηγορία, οπότε δεν είχαν πρακτική χρησιμότητα.

### 5.1.3 Επιλογή αλγόριθμου μάθησης

Ακόμα μία υπερπαραμέτρος, η οποία ελέγχθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας, πέρα από τις μεθόδους προεπεξεργασίας, αλλά και τις διάφορες τεχνικές εξομάλυνσης, είναι ο αλγόριθμος μάθησης. Μέχρι στιγμής, όπως αναφέρθηκε και στην προηγούμενη παράγραφο, έγινε χρήση του αλγόριθμου “Adam”, ενώ στη συνέχεια δοκιμάσαμε και τον αλγόριθμο μάθησης “SGD”, με σκοπό να συγκρίνουμε τα τις αποδόσεις τους.

### Διάγραμμα Αλγόριθμων Μάθησης I



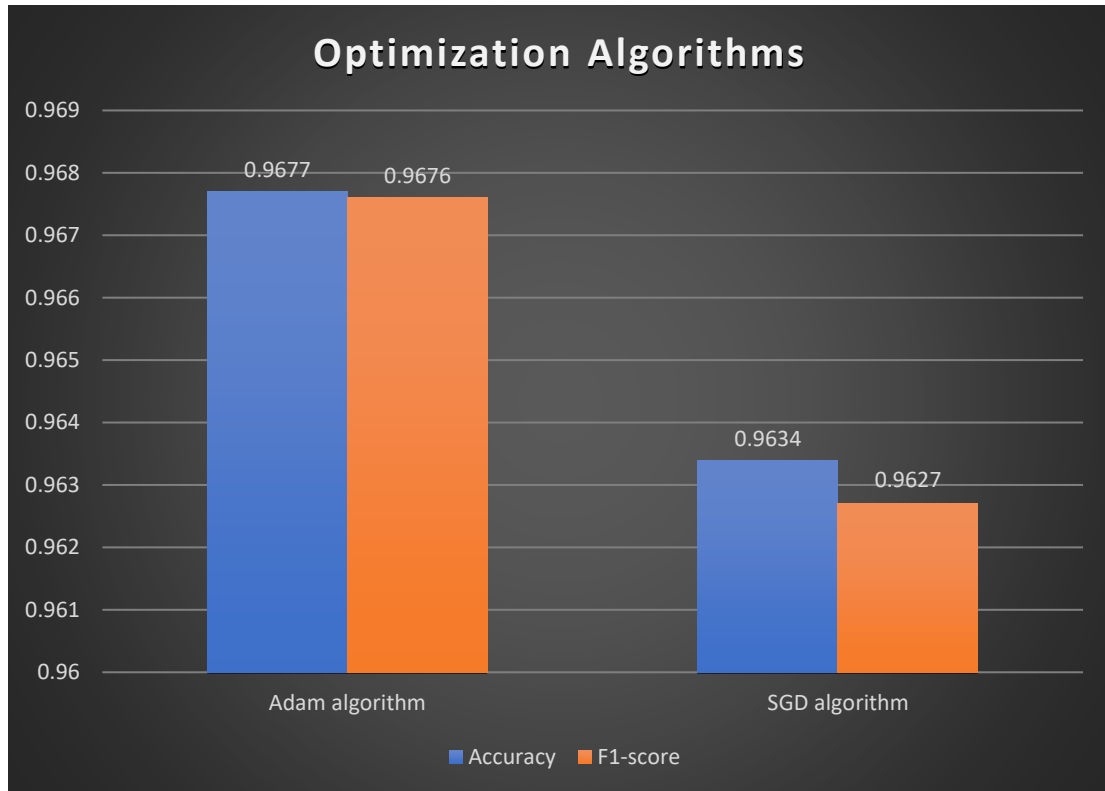
Εικόνα 5.4: Γράφημα αποτελεσμάτων εκπαίδευσης δύο συστημάτων, με διαφορετικούς αλγόριθμους μάθησης, SGD και Adam, σε συνδυασμό με ανισοτροπική διάχυση

Στο παραπάνω διάγραμμα, εμφανίζονται τα αποτελέσματα, ανάμεσα σε δύο μοντέλα τύπου ResNet, στα δεδομένα εισόδου των οποίων έχει εφαρμοστεί ανισοτροπική διάχυση και στη συνέχεια εξομάλυνση, κάνοντας χρήση του ολικού μέσου όρου των εικόνων. Ακόμη, και στις δύο περιπτώσεις έχουμε κοινό αριθμό εποχών, ίδιο batch\_size, ενώ ταυτόχρονα έχει γίνει αρχικοποίηση “imageNet”, διατηρώντας ίδια τα βάρη όλων των επιπέδων, πέρα από τα πέντε τελευταία επίπεδα μαζί με το επίπεδο εξόδου.

Η μοναδική διαφορά ανάμεσα σε αυτές τις δύο περιπτώσεις είναι ο αλγόριθμος μάθησης. Η πρώτη περίπτωση (αριστερά στο διάγραμμα) αντιστοιχεί στο μοντέλο κατά την εκπαίδευση του οποίου έγινε χρήση του αλγόριθμου μάθησης “Adam”, ενώ η δεύτερη περίπτωση (δεξιά στο διάγραμμα) αντιστοιχεί στο μοντέλο που χρησιμοποιήθηκε ο αλγόριθμος μάθησης “SGD”.

Παρατηρούμε, λοιπόν, ότι μέχρι στιγμής, την υψηλότερη απόδοση σε σχέση με οποιοδήποτε μοντέλο μηχανικής μάθησης παρουσίασε σύστημα με αλγόριθμο μάθησης “SGD”. Συγκεκριμένα, όπως φαίνεται και από το παραπάνω διάγραμμα όταν αλλάξαμε τον αλγόριθμο μάθησης από “Adam” σε “SGD”, παρατηρήθηκε αύξηση των μετρικών κατά τουλάχιστον 3%, στην περίπτωση βέβαια που χρησιμοποιήθηκε αποκλειστικά η μέθοδος προεπεξεργασίας δεδομένων ανισοτροπικής διάχυσης.

### Διάγραμμα Αλγόριθμων Μάθησης II



Εικόνα 5.5: Γράφημα αποτελεσμάτων εκπαίδευσης δύο συστημάτων, με διαφορετικούς αλγόριθμους μάθησης, SGD και Adam, σε συνδυασμό με μέθοδο διόρθωσης πεδίου πόλωσης και ανισοτροπική διάχυση.

Στο παραπάνω διάγραμμα, παρουσιάζονται τα αποτελέσματα ανάμεσα σε δύο μοντέλα μηχανικής μάθησης τύπου ResNet, με εκπαίδευση η οποία έχει συμβεί σε ίδιο αριθμό εποχών, με ίδιο batch\_size και αρχικοποίηση τύπου "ImageNet" και σταθερά τα βάρη όλων των επιπέδων πέρα των πέντε τελευταίων.

Τα δεδομένα εισόδου έχουν προκύψει από το συνδυασμό της ανισοτροπικής διάχυσης και της διόρθωσης πεδίου πόλωσης, ενώ έχει γίνει και εξομάλυνση, κάνοντας χρήση του ολικού μέσου όρου.

Στα αριστερά του διαγράμματος φαίνονται τα αποτελέσματα των μετρικών του συστήματος, κατά την εκπαίδευση του οποίου χρησιμοποιήθηκε ο αλγόριθμος μάθησης "Adam", ενώ στα δεξιά το ίδιο ακριβώς μοντέλο αλλά με χρήση αλγόριθμου μάθησης "SGD".

Σε αντίθεση με το προηγούμενο διάγραμμα, κατά το συνδυασμό των δύο μεθόδων προεπεξεργασίας δεδομένων, ο αλγόριθμος μάθησης "SGD", εμφάνισε μικρότερη απόδοση από αυτή του "Adam", παρ'ότι η διαφορά στη απόδοση των δύο συστημάτων είναι μικρή. Τελικά, φαίνεται ότι η χρήση του αλγόριθμου "SGD" συνδυάζεται αποτελεσματικότερα με

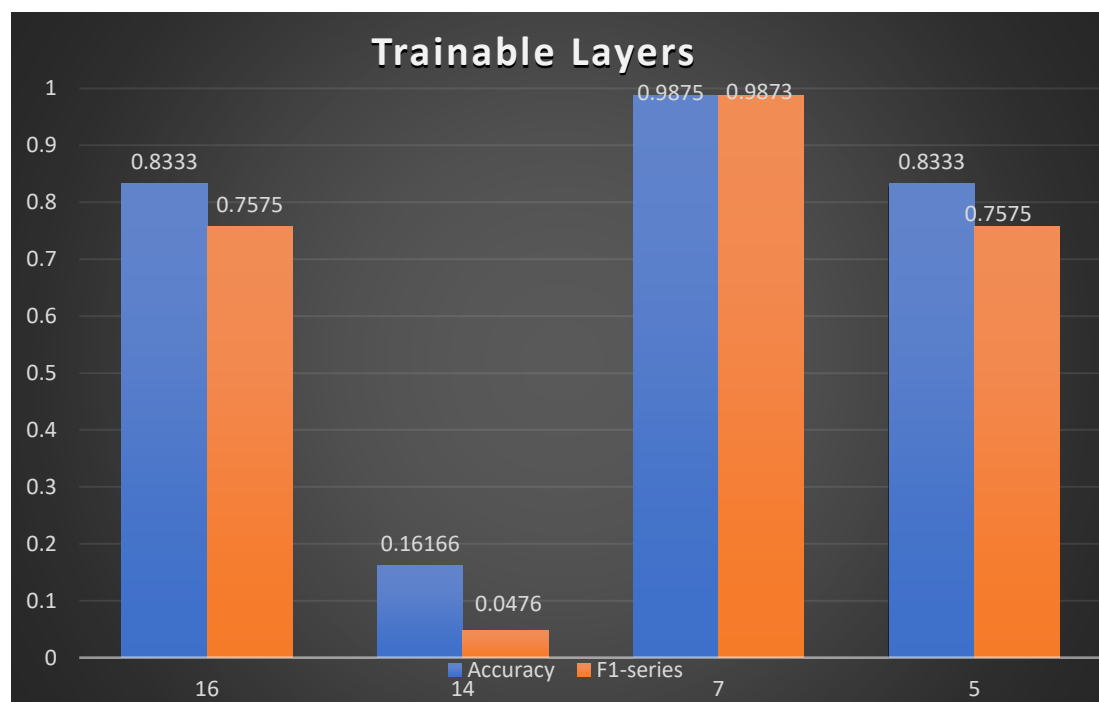
ταυτόχρονη χρήση της ανισοτροπικής διάχυσης και εξομάλυνσης με χρήση του ολικού μέσου όρου.

#### 5.1.4 Επιλογή εκπαιδεύσιμων επιπέδων και αρχικοποίηση βαρών

Όσον αφορά την εκπαίδευση του δικτύου VGGNet16, παρατηρήθηκε σε πολλές περιπτώσεις αδυναμία του μοντέλου στην αναγνώριση αναγνώριση χαρακτηριστικών σε συνδυασμό με συγκεκριμένες υπερπαραμέτρους.

Σαν αποτέλεσμα, η συνάρτηση σφάλματος δεν ελαχιστοποιούταν μέσω της αναγνώριση χαρακτηριστικών, αλλά με τη συνεχόμενη πρόβλεψη της ίδιας κατηγορίας, η οποία και εμφάνιζε περισσότερα δείγματα, δηλαδή της κατηγορίας υψηλού επιπέδου γλοιωμάτων.

#### Διάγραμμα πλήθους εκπαιδεύσιμων επιπέδων



Εικόνα 5.6: Γράφημα αποτελεσμάτων εκπαίδευσης τεσσάρων διαφορετικών συστημάτων VGGNet16, με διαφορετικό αριθμό εκπαιδεύσιμων επιπέδων.

Οι τέσσερις παραπάνω περιπτώσεις εκπαίδευσης μοντέλων VGGNet16, εκτελούνται με όλες τις υπερπαραμέτρους ίδιες, εκτός από τον αριθμό εκπαιδεύσιμων επιπέδων. Το πλήθος εποχών είναι ίδιο, όπως και ο αλγόριθμος μάθησης "SGD", αλλά και το batch\_size.

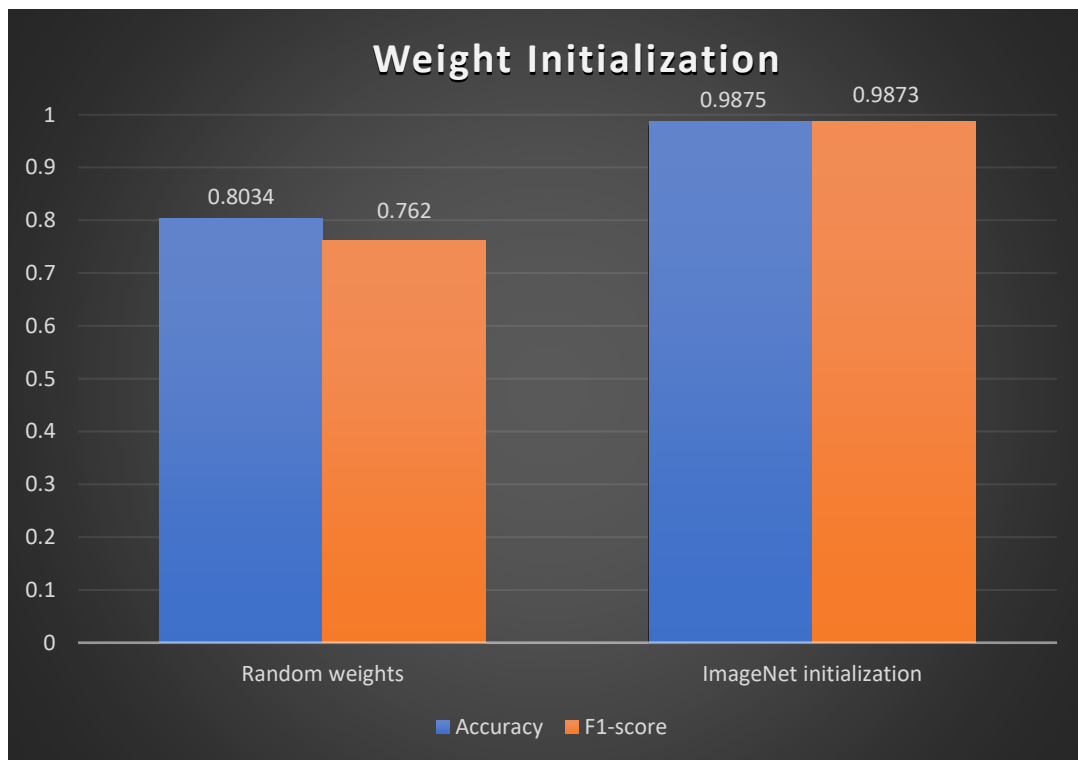
Ακόμη, έχει γίνει εξομάλυνση με χρήση ολικού μέσου όρου, ενώ δεν έχει εφαρμοστεί καμία άλλη μέθοδος προεπεξεργασίας.

Από αριστερά προς τα δεξιά του διαγράμματος, τα μοντέλα μηχανικής μάθησης εμφανίζουν όλο και λιγότερα εκπαιδεύσιμα επίπεδα. Συγκεκριμένα από αριστερά προς τα

δεξιά έχουμε τις περιπτώσεις, όπου τα εκπαιδευσιμα επίπεδα είναι αρχικά 16, δηλαδή εκπαιδεύουμε όλο το δίκτυο, 14, 7 και τέλος 5.

Οι περιπτώσεις όπου τα εκπαιδευσιμα επίπεδα είναι «πολλά» ή «λίγα» εμφανίζουν το πρόβλημα της πρόβλεψης μίας μόνο κατηγορίας. Όταν «παγώσουμε» όλα τα επίπεδα εκτός από δύο με αρχικοποίηση “ImageNet”, η οποία και έχει συμβεί σε όλα τα παραπάνω μοντέλα μηχανικής μάθησης, παρατηρείται πολύ μεγάλη αστοχία στην αναγνώριση της κάθε κατηγορίας (ακρίβεια = 0.1616). Τέλος, στην περίπτωση όπου ο αριθμός των εκπαιδευσιμων επιπέδων είναι επτά, τότε παρατηρείται ακρίβεια συγκρίσιμη, αλλά ελαφρώς μικρότερη σε απόδοση, όσον αφορά συστήματα που έχουν εκπαιδευτεί πάνω σε σύνολα δεδομένων, στα οποία έχει γίνει προεπεξεργασία.

### Διάγραμμα αρχικών βαρών



Εικόνα 5.7: Γράφημα αποτελεσμάτων εκπαίδευσης δύο διαφορετικών συστημάτων VGGNet16, με διαφορετική αρχικοποίηση βαρών, τυχαία αρχικοποίηση και αρχικοποίηση ImageNet.

Ακόμη, αξίζει να σχολιαστεί, όπως έγινε και σε προηγούμενη ανάλυση από πιο θεωρητική σκοπιά, η αξία της αρχικοποίησης “ImageNet”, σε συνδυασμό με τη μέθοδο του fine-tuning.

Οι δύο παραπάνω περιπτώσεις εκπαίδευσης αφορούν δύο συστήματα VGGNet16, με ίδιο αριθμό εποχών, ίδιο batch\_size και αλγόριθμο μάθησης “SGD”. Στα αριστερά του διαγράμματος το δίκτυο έχει εκπαιδευτεί συνολικά σε όλα του τα επίπεδα, δίχως να έχει γίνει αρχικοποίηση “ImageNet”, ενώ στα δεξιά του διαγράμματος για το αντίστοιχο μοντέλο έχουμε «παγώσει» όλα τα επίπεδα εκτός των επτά τελευταίων, με τα υπόλοιπα να μην εκπαιδεύονται, αλλά να έχει γίνει αρχικοποίηση “ImageNet”.

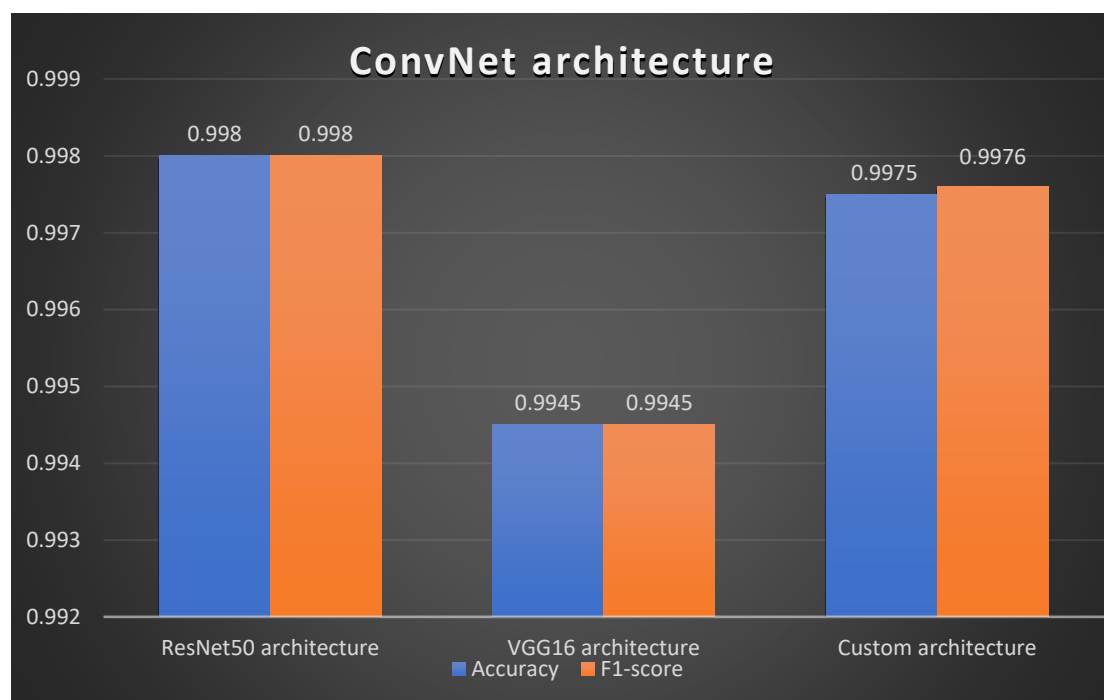


Όπως φαίνεται από το διάγραμμα, η δεύτερη περίπτωση παρουσιάζει κατά πολύ υψηλότερες μετρικές (τουλάχιστον 17%), με αποτέλεσμα να διαφαίνεται η σημαντικότητα της αρχικοποίησης των βαρών συνδυαστικά με την εκπαίδευση συγκεκριμένων μόνο υψηλών επιπέδων.

### 5.1.5 Επιλογή μοντέλου

Μία ακόμη παράμετρος, η οποία και θα εξεταστεί στα πλαίσια της παρούσας διπλωματικής εργασίας είναι η επιλογή της αρχιτεκτονικής συνελικτικού δικτύου. Η επιλογή αυτή θα γίνει ανάμεσα σε τρεις διαφορετικές αρχιτεκτονικές. Οι δύο πρώτες εξ αυτών είναι οι ResNet50 και VGG16. Η τρίτη είναι μία πιο ρηχή αρχιτεκτονική με μικρό αριθμό επιπέδων, την οποία και αποσκοπούμε στο να συγκρίνουμε με τις άλλες δύο, οι οποίες και θεωρούνται από τις πιο αποτελεσματικές, όσον αφορά το χώρο της αναγνώρισης εικόνας και θα την ονομάσουμε Custom αρχιτεκτονική.

**Διάγραμμα αρχιτεκτονικών συνελικτικών νευρωνικών δικτύων**



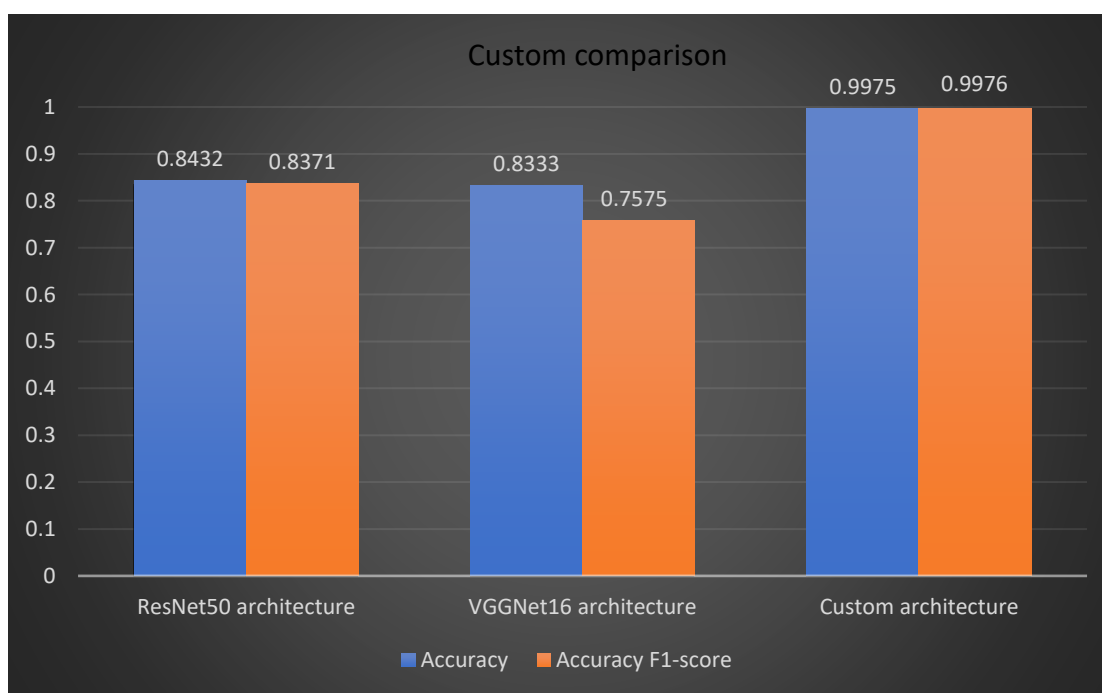
**Εικόνα 5.8: Τρεις διαφορετικές περιπτώσεις αρχιτεκτονικών με εφαρμογή ίδων μεθόδων προεπεξεργασίας δεδομένων VGGNet16, ResNet50 και Custom.**

Οι τρεις παραπάνω περιπτώσεις αρχιτεκτονικών συνελικτικών νευρωνικών δικτύων από αριστερά προς τα δεξιά, όσον αφορά το διάγραμμα, είναι αρχιτεκτονική τύπου ResNet50, VGG16 και η αρχιτεκτονική που υλοποιήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας, την οποία αποκαλούμε Custom.

Σε όλα τα παραπάνω μοντέλα μηχανικής μάθησης χρησιμοποιήθηκε ο αλγόριθμος μάθησης "SGD", ενώ εφαρμόσαμε μεθόδους προεπεξεργασίας ανισοτροπικής διάχυση και

εξομάλυνσης με χρήση ολικού μέσου όρου. Όσον αφορά τις υπόλοιπες υπερπαραμέτρους και τα τρία μοντέλα εκπαιδεύτηκαν στον ίδιο αριθμό εποχών, με ίδιο batch\_size. Όσον αφορά την αρχιτεκτονική ResNet50, «παγώσαμε» τα πέντε τελευταία επίπεδα, στην αρχιτεκτονική VGGNet16, τα επτά τελευταία επίπεδα και τελικά στην Custom αρχιτεκτονική δεν «παγώσαμε» κανένα επίπεδο, δηλαδή το δίκτυο εκπαιδεύτηκε συνολικά. Στις δύο πρώτες περιπτώσεις, εν τέλει, εφαρμόσαμε αρχικοποίηση βαρών “ImageNet”. Όπως φαίνεται και από το διάγραμμα, οι διαφορές ανάμεσα στην απόδοση των τριών δικτύων είναι σχετικά μικρή, όσον αφορά τις μετρικές της ακρίβειας και του F1-score. Παρ’όλ’αυτά, το ResNet50 είναι το μοντέλο εκείνο που στο σύνολο δεδομένων επαλήθευσης απέδωσε περισσότερο και στη συνέχεια βρίσκεται η Custom αρχιτεκτονική με μόλις 0.05% μικρότερη ακρίβεια. Τελικά, το VGGNet16 απέχει περίπου 0.3%. Συγκρίνοντας, οπότε, τις αποδόσεις των παραπάνω μοντέλων φαίνεται ότι όσον αφορά τουλάχιστον το συγκεκριμένο πρόβλημα, όπως θα ανφερθεί και παρακάτω, η επιλογή του μοντέλου έχει μικρότερη βαρύτητα όσον αφορά την απόδοση, σε σχέση με τις τεχνικές προεπεξεργασίας, εξομάλυνσης, επιλογής αλγόριθμου μάθησης, αλλά και πλήθος εκπαιδευσιμων επιπέδων. Επίσης, όπως θα ανφέρουμε και σε επόμενο μέρος του κεφαλαίου, μία πιο ρηχή αρχιτεκτονική μπορεί να αποδώσει εξίσου καλά με μία πιο βαθιά στο συγκεκριμένο πρόβλημα, καθώς η αναγνώριση των χαρακτηριστικών πιστεύουμε ότι στηρίζεται σε πιο «απλά» και όχι τόσο «περίπλοκα» φίλτρα, σαν αυτά τα οποία υλοποιούνται στα υψηλά επίπεδα των ResNet50 και VGGNet16.

#### Διάγραμμα σύγκρισης Custom με «απλές» περιπτώσεις



Εικόνα 5.9: Τρεις διαφορετικές περιπτώσεις αρχιτεκτονικών με εφαρμογή μεθόδων προεπεξεργασίας δεδομένων μόνο στην περίπτωση της Custom αρχιτεκτονικής.

Σε αυτή την περίπτωση θα συγκρίνουμε τις δύο «απλές» περιπτώσεις μοντέλων μηχανικής μάθησης ResNet50 και VGGNet16, χωρίς την εφαρμογή καμίας μεθόδου προεπεξεργασίας ή

εξομάλυνσης στα δεδομένα και κάνοντας χρήση του αλγόριθμου μάθησης «Adam», με την περίπτωση της Custom αρχιτεκτονικής, τα δεδομένα της οποία έχουν υποστεί ανισοτροπική διάχυση, εξομάλυνση με χρήση του ολικού μέσου όρου, ενώ έχει γίνει χρήση του αλγόριθμου μάθησης “SGD”.

Ο αριθμός των εποχών είναι ίδιος και για τις τρεις περιπτώσεις, όπως και το batch\_size, ενώ μόνο για τα πρώτα δύο μοντέλα έχει εφαρμοστεί αρχικοποίηση “ImageNet” και έχουμε «παγώσει» όλα τα επίπεδα εκτός των πέντε τελευταίων για το ResNet και των επτά τελευταίων για το VGGNet16.

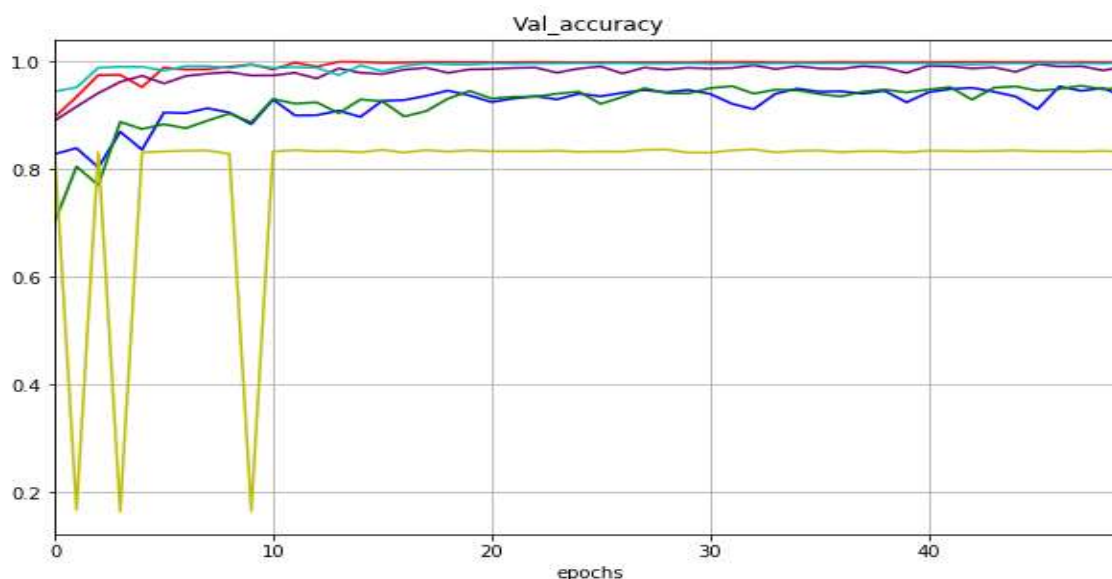
Παρατηρείται, λοιπόν, όπως θα αναφέρουμε και σε επόμενο μέρος του κεφαλαίου ότι σε σχέση με την επιλογή ενός μοντέλου με μεγάλο αριθμό παραμέτρων, η πιο «στοχευμένη» επιλογή υπερπαραμέτρων και μεθόδων προεπεξεργασίας, τουλάχιστον όσον αφορά το συγκεκριμένο πρόβλημα, έχει σαν αποτέλεσμα μία πιο ρηχή και όχι τόσο «ακριβή» από πλευράς παραμέτρων, άρα και μνήμης αρχιτεκτονική να επιφέρει τελικά υψηλότερη απόδοση.

## 5.2 Παρουσίαση Γραφικών Εκπαίδευσης

Σε αυτό το μέρος του κεφαλαίου, αποσκοπούμε στο να παρουσιάσουμε πειραματικά δεδομένα, τα οποία συλλέξαμε κατά τη διάρκεια της εκπαίδευσης των παραπάνω μοντέλων μηχανικής μάθησης, με σκοπό να εξετάσουμε παραμέτρους όπως ρυθμό σύγκλισης, αριθμό εποχών, αλλά και λοιπά χαρακτηριστικά των εκπαιδευόμενων συστημάτων. Συγκεκριμένα, θα παρουσιάσουμε σε κοινές γραφικές, τα αποτελέσματα που εξήχθησαν, κατά τη διάρκεια της εκπαίδευσης συστημάτων, στα δεδομένα εισόδου των οποίων είχαν εφαρμοστεί οι ίδιες μέθοδοι προεπεξεργασίας. Στη συνέχεια, θα συγκρίνουμε τα αποτελέσματα αυτά ανεξάρτητα από τις μεθόδους προεπεξεργασίας, με σκοπό την εξαγωγή συμπερασμάτων, όσον αφορά τα χαρακτηριστικά που αναφέρθηκαν και παραπάνω, όπως είναι χρόνοι, ρυθμοί σύγκλισης, αλλά και πλήθος εποχών.

## 5.2.1 Παρουσίαση Γραφημάτων Μοντέλων στην περίπτωση Ανιστροπικής Διάχυσης

### Γραφική Εκπαίδευσης Συστημάτων Ανιστροπικής Διάχυσης



Εικόνα 5.11: Γραφική σύγκλισης συστημάτων στην περίπτωση εφαρμογής ανιστροπικής διάχυσης για τις 50 πρώτες εποχές.

Χρώμα Γραφικής	Μοντέλο Μηχανικής Μάθησης
<b>Κόκκινο</b>	ResNet50 - Global Mean - SGD
<b>Γαλάζιο</b>	Custom - Global Mean - SGD
<b>Μώβ</b>	VGGNet16 - Global Mean – SGD
<b>Πράσινο</b>	ResNet50 - No Norm – Adam
<b>Μπλε</b>	ResNet50 - Global Mean - Adam
<b>Κίτρινο</b>	ResNet50 - Local Mean - Adam

**Πίνακας 5.1: Αντιστοιχία χρώματος με μοντέλο εκπαιδευμένο σε δεδομένα, στα οποία έχει εφαρμοστεί ανιστροπική διάχυση.**

Στο παραπάνω γράφημα, παρουσιάζονται το πλήθος των εποχών πάνω στο σύνολο επαλήθευσης για το σύνολο των συστημάτων, τα οποία εκπαιδεύτηκαν με δεδομένα που προήλθαν αποκλειστικά από την εφαρμογή της μεθόδου προεπεξεργασίας ανιστροπικής διάχυσης.

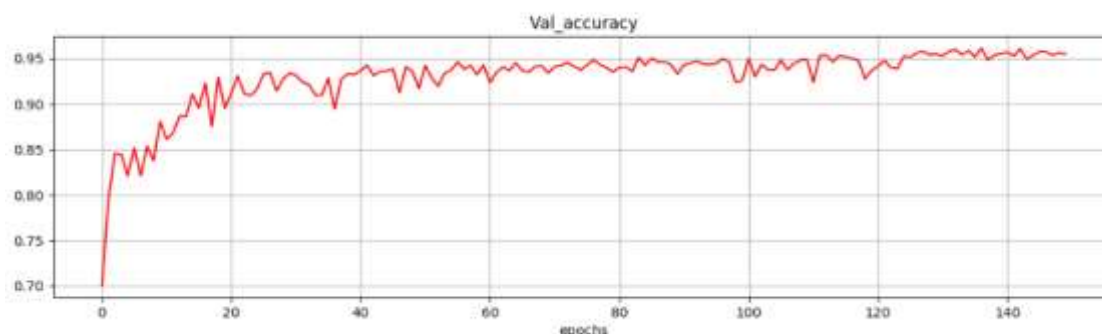
Παρατηρούμε ότι στο σύνολο τους, τα μοντέλα μηχανικής μάθησης συνέχισαν την εκπαίδευσή τους για διαφορετικό αριθμό εποχών. Η εκπαίδευση του κάθε μοντέλου μηχανικής μάθησης, τερματίστηκε από τη συνάρτηση `ReduceLROnPlateau()` της βιβλιοθήκης “keras”, όταν η επόμενη μέτρηση σε σχέση με την προηγούμενη παρουσίασαν ελάχιστη διαφορά, οπότε η περαιτέρω εκπαίδευση δεν θα επέφερε παραπάνω αποτέλεσμα.

Συγκεκριμένα, το μοντέλο μηχανικής μάθησης, στο οποίο έγινε χρήση της αρχιτεκτονικής τύπου ResNet50 και του αλγόριθμου μάθησης “SGD”, τερμάτισε την εκπαίδευσή του στις 53 εποχές, ενώ το αντίστοιχο σύστημα VGGNet16 με χρήση “SGD” τερμάτισε την εκπαίδευσή του μετά την ολοκλήρωση των 49 εποχών. Επίσης, παρατηρούμε ότι όσον αφορά το συνολικό χρόνο εκπαίδευσης, για την περίπτωση τουλάχιστον του δικτύου ResNet50, ο συνδυασμός με τον αλγόριθμο μάθησης “SGD”, σημαίνει μείωση των εποχών από 85 σε 49, σε σχέση με τη χρήση του “Adam. Παρ’όλ’αυτά, η σύγκλιση σε «σχετικά» σταθερή απόδοση, όσον αφορά τουλάχιστον τις μετρικές πάνω στο σύνολο δεδομένων, με τη βοήθεια του “SGD”, συμβαίνει στην 20<sup>η</sup> εποχή, ενώ μεγαλύτερης διακύμανσης αυξομειώσεις συμβαίνουν κατά τη χρήση του “Adam”, μέχρι και το τέλος της εκπαίδευσης. Μοναδική εξαίρεση σε αυτό αποτελεί το ResNet, στο οποίο έχει γίνει χρήση της μεθόδου εξομάλυνσης, κάνοντας χρήση του τοπικού μέσου όρου των εικόνων. Σε αυτή την περίπτωση, βέβαια, παρατηρείται επίσης και αστοχία του δικτύου στην αναγνώριση των απαιτούμενων χαρακτηριστικών των εικόνων, γι’αυτό παρατηρούνται και χαμηλές μετρικές ακρίβειας. Όσον αφορά την περίπτωση του VGGNet, σε συνδυασμό με την περίπτωση του “SGD”, το πλήθος των εποχών που απαιτούνται για σύγκλιση είναι, όπως αναφέραμε και πριν 53, ενώ σε αυτή την περίπτωση παρουσιάζονται αυξομειώσεις μέχρι και το τέλος της εκπαίδευσης, τουλάχιστον όσον αφορά τη μετρική της ακρίβειας.

Τελικά, το Custom δίκτυο, όσον αφορά τη σύγκλιση, απαιτεί 20 εποχές και στη συνέχεια παρατηρούνται πολύ μικρές αυξομειώσεις, οι οποίες και δεν είναι ικανές να σταματήσουν την εκπαίδευση μέσω της αντίστοιχης συνάρτησης του “keras”. Σε σχέση, οπότε με την περίπτωση προεπεξεργασίας δεδομένων ανισοτροπικής διάχυσης, φαίνεται ότι σε συνδυασμό με την αύξηση της ακρίβειας, η οποία προκύπτει από τη χρήση του αλγόριθμου μάθησης “SGD”, παρατηρείται και μείωση στο πλήθος των εποχών εκπαίδευσης. Επίσης, η σύγκλιση σε «σχετικά» σταθερή τιμή συμβαίνει σε μικρό αριθμό εποχών για το σύνολο των μοντέλων μηχανικής μάθησης εκτός του “VGGNet16”. Μικρότερο αριθμό εποχών απαιτεί η περίπτωση του ResNet50, στη συνέχεια το VGGNet16 και τέλος η Custom αρχιτεκτονική, για την οποία και η εκπαίδευση συνεχίστηκε για παραπάνω από 100 εποχές σε πλήθος. Οι μέθοδοι εξομάλυνσης φαίνεται να μην επηρεάζουν, με μεγάλη βαρύτητα τουλάχιστον, το χρόνο σύγκλισης και το χρόνο εκπαίδευσης στο παραπάνω γράφημα. Τέλος, κατά τη χρήση του “SGD”, παρατηρείται και μικρότερη διακύμανση στις τιμές των διαδοχικών εποχών.

### 5.2.2 Παρουσίαση Γραφημάτων Μοντέλων στην περίπτωση Διόρθωσης Πεδίου Πόλωσης

#### Γραφική Εκπαίδευσης Συστημάτων Διόρθωσης Πεδίου Πόλωσης



Εικόνα 5.12: Γραφική σύγκλισης του συστήματος ResNet50 στην περίπτωση εφαρμογής διόρθωσης πεδίου πόλωσης για τις 50 πρώτες εποχές.

<b>Κόκκινο</b>	ResNet50 - Global Mean – Adam
----------------	-------------------------------

**Πίνακας 5.2: Αντιστοιχία χρώματος με γραφική μοντέλου στην περίπτωση προεπεξεργασίας διόρθωσης πεδίου πόλωσης.**

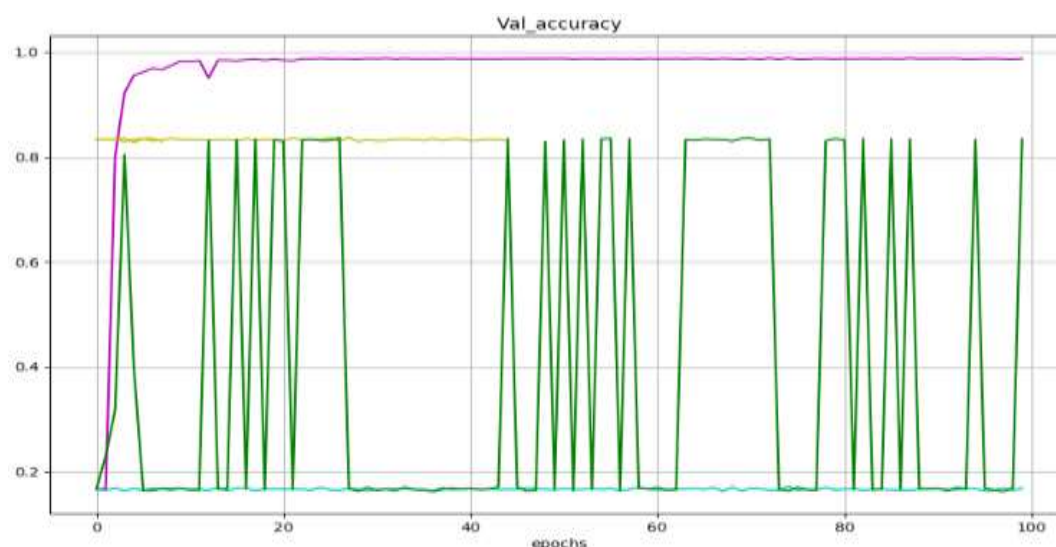
Στην περίπτωση της μεθόδου διόρθωσης πεδίου πόλωσης, το μοντέλο ResNet50 είναι το μόνο, το οποίο και στην περίπτωση της συγκεκριμένης προεπεξεργασίας απέδωσε συγκρίσιμα με τα μοντέλα της προηγούμενης γραφικής, τουλάχιστον όσον αφορά τη μετρική της ακρίβειας. Το τελευταίο εκπαιδεύτηκε τόσο χρόνο όσο και το ResNet με “SGD” στην περίπτωση της προηγούμενης γραφικής.

Παρατηρούμε για ακόμα μία φορά ότι κατά τη χρήση του “Adam”, η σύγκλιση συμβαίνει πολύ μετά την 20<sup>η</sup> εποχή και κοντά στην 145<sup>η</sup> εποχή. Παρατηρείται, ακόμη, μεγάλη διακύμανση στις τιμές της ακρίβειας κατά τη διάρκεια της εκπαίδευσης, το οποίο για ακόμα μία φορά συμφωνεί με τα αποτελέσματα του προηγούμενου διαγράμματος. Η εκπαίδευση, στη συγκεκριμένη περίπτωση, συνέβη για το σύνολο των εποχών που είχαμε ορίσει αρχικά.

Μία πιθανή εξήγηση του παραπάνω είναι η εξής: Η μέθοδος διόρθωσης πεδίου πόλωσης προκαλεί μικρές διαφορές στη μορφή της εικόνας εισόδου, ενώ η ανισοτροπική διάχυση προκαλεί μεγαλύτερες διαφορές, διότι εφαρμόζεται σε όλους τους διαφορετικούς τύπους ιστών, οπότε προκαλεί μεγαλύτερη διακύμανση σε πίξελ, που ανήκουν σε διαφορετικά χαρακτηριστικά. Αυτό έχει σαν αποτέλεσμα, ένα σύνολο εικόνων με μεγαλύτερη διαφορετικότητα στα χαρακτηριστικά του να είναι δυνατόν να «μαθευτεί» πιο γρήγορα και σε μεγαλύτερο βαθμό από το εκάστοτε σύστημα.

### 5.2.3 Παρουσίαση Γραφημάτων Μοντέλων στην περίπτωση Απουσίας Προεπεξεργασίας

#### Γραφική Εκπαίδευσης Συστημάτων Χωρίς Μέθοδο Προεπεξεργασίας



**Εικόνα 5.13: Γραφική σύγκλισης VGGNet16 στην περίπτωση εφαρμογής μόνο μεθόδου εξομάλυνσης ολικού μέσου για 100 εποχές με διαφορετικό αριθμό εκπαιδευσίμων επιπέδων.**

Χρώμα Γραφικής	Μοντέλο Μηχανικής Μάθησης
Κίτρινο	VGGNet16 - Global - 5 layers - SGD
Γαλάζιο	VGGNet16 - Global -14 layers - SGD
Μώβ	VGGNet16 - Global - 7 layers - SGD
Πράσινο	VGGNet16 - Global - Random W. - SGD

**Πίνακας 5.3: Αντιστοιχία χρώματος με γραφική μοντέλου στην περίπτωση προεπεξεργασίας διόρθωσης πεδίου πόλωσης.**

Το παραπάνω γράφημα αποτελείται από μετρικές πάνω σε διαφορετικά συστήματα (κυρίως VGGNet16), τα οποία έχουν εκπαιδευτεί σε σύνολα δεδομένων στα οποία δεν έχουν εφαρμοστεί άλλες μέθοδοι προεπεξεργασίας δεδομένων, πέρα από διαφορετικές εξομαλύνσεις.

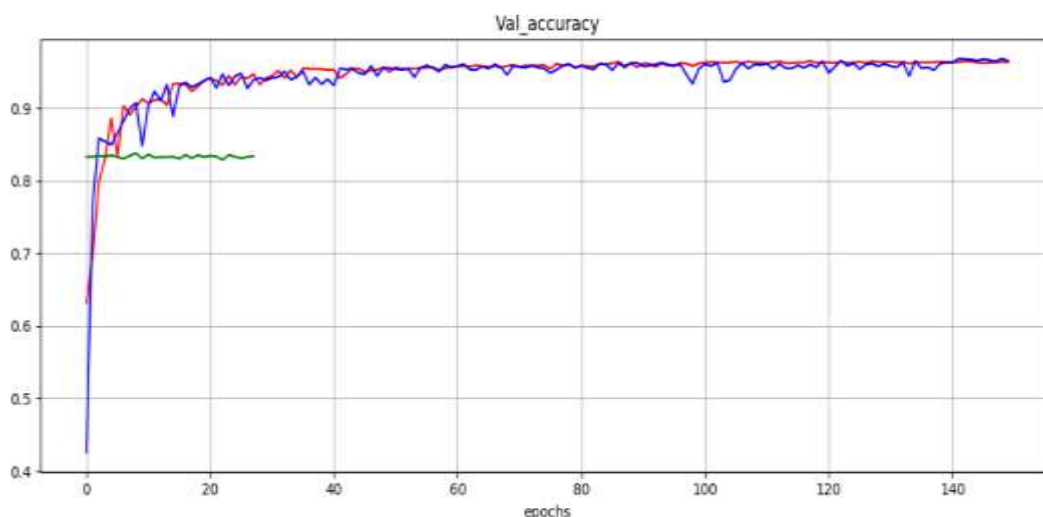
Παρατηρούμε ότι σε όλες τις παραπάνω περιπτώσεις του VGGNet16, πέρα από αυτή που έχουμε την υψηλότερη ακρίβεια στα επτά εκπαιδευσιμα επίπεδα, ότι είτε έχει εμφανιστεί το πρόβλημα πρόβλεψης μίας κατηγορίας είτε το σύστημα αδυνατεί να «μάθει» τα απαιτούμενα χαρακτηριστικά, με αποτέλεσμα να οδηγούμαστε σε πολύ χαμηλές αποδόσεις.

Ειδικά στην περίπτωση της τυχαίας αρχικοποίησης βαρών, εκτός από χαμηλές μετρήσεις στη μετρική της απόδοσης, παρατηρείται, ταυτόχρονα και τεράστια διακύμανση σε διαδοχικές τιμές των μετρικών.

Ακόμη, παρατηρούμε πως το απαιτούμενο πλήθος εποχών για σύγκλιση αυξάνεται σύμφωνα με το πλήθος των εκπαιδευσιμων επιπέδων. Το παραπάνω είναι αναμενόμενο, καθώς όσα παραπάνω σε πλήθος είναι τα επίπεδα, τόσο περισσότερες είναι και οι προς ενημέρωσι παράμετροι, άρα απαιτείται και μεγαλύτερος χρόνος στην εκπαίδευση του εκάστοτε δικτύου.

## 5.2.4 Παρουσίαση Γραφημάτων Μοντέλων στην περίπτωση Συνδυασμού Μεθόδων Προεπεξεργασίας

### Γραφική Εκπαίδευσης Συστημάτων Και Με Τις 2 Μέθόδους Προεπεξεργασίας



Εικόνα 5.14: Γραφικές σύγκλισης συστημάτων στην περίπτωση εφαρμογής διόρθωσης πεδίου πόλωσης συνδυαστικά με μέθοδο ανισοτροπικής διάχυσης.

Χρώμα Γραφικής	Μοντέλο Μηχανικής Μάθησης
Πράσινο	VGGNet16 - Global - 7 layers - SGD
Κόκκινο	ResNet50 - Global -5 layers – SGD
Μπλε	ResNet50 - Global - 5 layers – Adam

Πίνακας 5.4: Αντιστοιχία χρώματος με γραφική μοντέλου στην περίπτωση συνδυασμού μεθόδων προεπεξεργασίας.

Στο συγκεκριμένο γράφημα, στα δεδομένα εισόδου των προς εκπαίδευση συστημάτων έχουν εφαρμοστεί και οι δύο μέθοδοι προεπεξεργασίας. Η περίπτωση του δικτύου VGGNet16, εμφανίζει το πρόβλημα της πρόβλεψης μίας κατηγορίας, ενώ οι υπόλοιπες περιπτώσεις, στις οποίες η εκπαίδευση έχει συνεχιστεί ομαλά είναι αυτές των ResNet50, με τους δύο διαφορετικούς αλγόριθμους μάθησης “SGD” και “Adam”.

Παρατηρείται, λοιπόν, για ακόμα μία φορά ότι στην περίπτωση του πρώτου αλγόριθμου μάθησης “SGD”, επιτυγχάνεται σταθερή τιμή στη μετρική της ακρίβειας, σε μικρό πλήθος εποχών, δηλαδή κατά την 20<sup>η</sup> εποχή και πριν από αυτή παρατηρούνται σχετικά μικρές διακυμάνσεις στις διαδοχικές τιμές ακρίβειας. Σε αντίθεση με αυτό στην περίπτωση του “Adam”, η διακύμανση των διαδοχικών τιμών είναι κατά πολύ μεγαλύτερη, όπως φαίνεται και από το σχετικό διάγραμμα, ενώ η εκπαίδευση ολοκληρώνεται κατά την 150<sup>η</sup> εποχή, σε πολύ μεγαλύτερο, δηλαδή, αριθμό εποχών σε σχέση με τη χρήση το “SGD” .



## 5.3 Συμπεράσματα και Σχολιασμός

Στο παρόν μέρος του κεφαλαίου αυτού, θα σχολιάσουμε τις επιδράσεις, τις οποίες είχαν διάφορες υπερπαραμέτροι στην απόδοση του εκάστοτε προς εκπαίδευση συστήματος.

Συγκεκριμένα, θα χρησιμοποιήσουμε τα αποτελέσματα από τα προηγούμενα δύο μέρη του κεφαλαίου, με σκοπό να σχολιάσουμε την επιλογή των υπερπαραμέτρων, μεθόδων προεπεξεργασίας δεδομένων, αρχιτεκτονικών συνελκτικών νευρωνικών δικτύων, τα οποία και παρουσίασαν τα βέλτιστα αποτελέσματα. Επίσης, θα γίνει σχολιασμός του αριθμού των εποχών σύγκλισης σε συνάρτηση με τους αλγόριθμους μάθησης, αλλά και τις τεχνικές προεπεξεργασίας δεδομένων.

Σκοπός του παραπάνω είναι να καταλήξουμε σε γενικά συμπεράσματα όσον αφορά το πρόβλημα δυαδικής ταξινόμησης μαγνητικών τομογραφιών σε υψηλού και χαμηλού βαθμού γλοιώματα.

### 5.3.1 Σχολιασμός επιλογής μεθόδου κανονικοποίησης

Όσον αφορά τις μεθόδους προεπεξεργασίας δεδομένων, όπως φάνηκε στα δύο αρχικά διαγράμματα του κεφαλαίου, χρησιμοποιήσαμε συνολικά δύο διαφορετικές μεθόδους εξομάλυνσης δεδομένων, αυτή του ολικού και αυτή του τοπικού μέσου και τις συγκρίναμε με την περίπτωση απουσίας εξομάλυνσης.

Στο σύνολο των μετρικών που εξήχθησαν από τα εκπαιδευμένα συστήματα, η περίπτωση της εξομάλυνσης με χρήση του ολικού μέσου, είχε σαν αποτέλεσμα υψηλότερη απόδοση σε σχέση με την περίπτωση της απουσίας εξομάλυνσης, ενώ η χρήση της δεύτερης τεχνικής εξομάλυνσης, δηλαδή αυτή του τοπικού μέσου, είχε σαν αποτέλεσμα συστήματα, τα οποία προέβλεπαν συνεχώς την ίδια κατηγορία, άρα δεν ήταν δυνατόν να χρησιμοποιηθούν για αναγνώριση μαγνητικών τομογραφιών.

Όπως αναφέρθηκε και στη θεωρητική ανάλυση των μεθόδων εξομαλύνσεων, η μείωση της διακύμανσης και του μεγέθους των τιμών στα πίξελ των εικόνων, έχει σαν αποτέλεσμα διαφορετικά χαρακτηριστικά στην εικόνα να «περιγράφονται» από κοντινές τιμές πίξελ. Οπότε, στην ενημέρωση των βαρών του δικτύου, χαρακτηριστικά που θα περιγράφονταν με υψηλές τιμές πίξελ, χωρίς την εξομάλυνση, θα επιδρούσαν πολύ πιο έντονα στο δίκτυο, σε σχέση με χαρακτηριστικά χαμηλών πίξελ.

Με τη χρήση της εξομάλυνσης αποφεύγουμε το παραπάνω, οπότε και το σύστημά μας ενημερώνει τα βάρη συναρτήσει των διαφορετικών χαρακτηριστικών, αλλά με πιο «αντικειμενικό» τρόπο.

Ο παραπάνω ισχυρισμός διαφαίνεται μέσω της χρήσης του ολικού μέσου όρου για την εξομάλυνση των εικόνων, διότι σε αυτή την περίπτωση λαμβάνονται υπ' όψη τα χαρακτηριστικά του συνόλου των εικόνων, με την ίδια βαρύτητα και παρουσιάζεται υψηλότερη απόδοση σε σχέση με τη μη χρήση εξομάλυνσης.

Ο λόγος για τον οποίο απέτυχε η χρήση του τοπικού μέσου όρου στη διαδικασία της εξομάλυνσης, θεωρούμε πως είναι ότι όταν λαμβάνουμε διαφορετικό μέσο όρο από κάθε εικόνα, τότε τα ίδια χαρακτηριστικά σε διαφορετικές εικόνες θα έχουν διαφορετικές

διακυμάνσεις στα πίξελ τους, οπότε θα είναι και πιο δύσκολο το εκάστοτε προς εκπαίδευση σύστημα να τα συσχετίσει.

Λόγω του παραπάνω οδηγούμαστε σε συστήματα, τα οποία δεν είναι δυνατόν να χρησιμοποιηθούν πρακτικά, καθώς η συνάρτηση σφάλματος μεγιστοποιείται ανεξάρτητα από τα προς αναγνώριση χαρακτηριστικά και καταλήγει να αναγνωρίζει τη μία κατηγορία δηλαδή αυτή με την υψηλότερη συχνότητα (υψηλού επιπέδου γλοιώματα).

Ακόμη, όπως φάνηκε και στο δεύτερο διάγραμμα του κεφαλαίου οι μέθοδοι εξομάλυνσης είναι ανεξάρτητοι από τη χρήση ανισοτροπικής διάχυσης ή μεθόδου διόρθωσης πεδίου πόλωσης, διότι αντίστοιχα συμπεράσματα εξάγαμε και στην περίπτωση της χρήσης καθεμίας από τις παραπάνω τεχνικές προεπεξεργασίας.

Αξίζει επίσης να αναφερθεί ότι σε σχέση με τον αριθμό των εποχών σύγκλισης, η επιλογή μεθόδου εξομάλυνσης δεν απέφερε ιδιαίτερες διαφορές, όπως συνέβη στην περίπτωση της μετρικής της ακρίβειας.

Τελικά, ανεξάρτητα από λοιπές μεθόδους προεπεξεργασίας δεδομένων, καταλήγουμε στο συμπέρασμα, ότι όσον αφορά τουλάχιστον το συγκεκριμένο πρόβλημα δυαδικής ταξινόμησης μαγνητικών τομογραφιών, η χρήση της μεθόδου εξομάλυνσης, κάνοντας χρήση του ολικού μέσου όρου των εικόνων, είναι πιο αποτελεσματική και από τη μη χρήση εξομάλυνσης, αλλά και σε σχέση με τη χρήση τοπικού μέσου, διότι συντελεί στην αύξηση της ακρίβειας.

### 5.3.2 Σχολιασμός Επιλογής Μεθόδου Προεπεξεργασίας Δεδομένων

Ανεξάρτητα από τις μεθόδους εξομάλυνσης δεδομένων, μεγάλη σημασία για την απόδοση του εκάστοτε συστήματος, έχει και η χρήση άλλων τεχνικών προεπεξεργασίας δεδομένων.

Στην περίπτωση μας, δηλαδή, στα πλαίσια της παρούσας διπλωματικής εργασίας, έχει γίνει σύγκριση μεταξύ τεσσάρων μοντέλων μηχανικής μάθησης. Οι διαφορές είναι στις μεθόδους προεπεξεργασίας, αλλά με ταυτόχρονη χρήση του αλγόριθμου μάθησης "Adam". Δηλαδή, το πρώτο σύστημα έχει εκπαιδευτεί πάνω σε σύνολο δεδομένων, στο οποίο έχει εφαρμοστεί μέθοδος ανισοτροπικής διάχυσης, το δεύτερο σε δεδομένα που προέκυψαν από το συνδυασμό της ανισοτροπικής διάχυσης και της μεθόδου διόρθωσης πεδίου πόλωσης, το τρίτο πάνω σε δεδομένα που δεν έχουν υποστεί καμία άλλη προεπεξεργασία, παρά μόνο εξομάλυνση ολικού μέσου και το τέταρτο σε σύνολο δεδομένων, στο οποίο έχει εφαρμοστεί αποκλειστικά μέθοδος διόρθωσης πεδίου πόλωσης.

Η περίπτωση εκείνη, η οποία παρουσιάζει τη μικρότερη απόδοση είναι αυτή της απουσίας μεθόδου προεπεξεργασίας. Στις περιπτώσεις όπου γίνεται χρήση αποκλειστικά μίας τεχνικής προεπεξεργασίας παρατηρείται σημαντική διαφορά στη μετρική της ακρίβειας, σε σχέση με την πρώτη περίπτωση, ενώ ο συνδυασμός και των δύο τεχνικών δίνει την υψηλότερη απόδοση ανάμεσα στα τέσσερα συστήματα. Αποκλειστικά οι χρήσεις διαφορετικών μεθόδων παρουσιάζουν πολύ «κοντινές» αποδόσεις μεταξύ τους.

Το παραπάνω είναι αναμμενόμενο, καθώς, η ανισοτροπική διάχυση πρακτικά ενισχύει τη διαφορετικότητα μεταξύ των διάφορων χαρακτηριστικών στην ίδια εικόνα, ενώ η μέθοδος διόρθωσης πεδίου πόλωσης αφαιρεί το θόρυβο.

Σαν αποτέλεσμα, η επιλογή οποιασδήποτε μεθόδου προεπεξεργασίας δεδομένων συντελεί στην αύξηση της απόδοσης, αλλά ειδικά ο συνδυασμός τους στην περίπτωση του “Adam” βελτιστοποιεί την απόδοση. Το ίδιο, όμως σε συμβαίνει και στην περίπτωση του “SGD”, όπως θα σχολιάσουμε και παρακάτω. Στην περίπτωση χρήσης του αλγόριθμου “SGD”, οπότε, παρατηρείται ότι ο συνδυασμός των δύο τεχνικών προεπεξεργασίας δεδομένων επιφέρει ίδια απόδοση με αυτόν του “Adam”, ενώ αποκλειστικά η περίπτωση της ανισοτροπικής διάχυσης από πλευράς προεπεξεργασίας δεδομένων, σε συνδυασμό με τον αλγόριθμο “SGD” παρουσιάζει υψηλότερες μετρικές σε σχέση με όλες τις προαναφερθείσες περιπτώσεις.

Όσον αφορά τον απαιτούμενο αριθμό εποχών σύγκλισης, επειδή έχει γίνει χρήση του αλγόριθμου “Adam”, όπως θα αναφερθεί και παρακάτω, παρατηρούνται σημαντικές αυξομειώσεις σε διαδοχικές τιμές της ακρίβειας, με την εκπαίδευση να ολοκληρώνεται σε μικρότερο αριθμό εποχών αποκλειστικά κατά τη χρήση της ανισοτροπικής διάχυσης. Η μέθοδος διόρθωσης πεδίου πόλωσης «καθαρίζει» την εικόνα από ένα θορυβώδες μαγνητικό σήμα, αλλά αλλάζει τα πίκσελ σε μικρό βαθμό. Οι διαφορές, δηλαδή, δεν εντείνονται, όπως στην περίπτωση της ανισοτροπικής διάχυσης, άρα στις περιπτώσεις που έχει γίνει χρήση τη πρώτης μεθόδου παρατηρείται μεγαλύτερο πλήθος εποχών σύγκλισης.

Το ίδιο παρατηρείται και στην περίπτωση του “SGD”, αλλά με τη διαφορά ότι η σύγκλιση για το συγκεκριμένο αλγόριθμο μάθησης απαιτεί μικρότερο αριθμό εποχών, όπως θα σχολιαστεί και σε επόμενη παράγραφο. Τελικά, όσον αφορά τις μεθόδους προεπεξεργασίας δεδομένων, ανεξάρτητα από την επιλογή μεθόδου εξομάλυνσης, συμπαιρνούμε πως βελτιώνουν την απόδοση συνδυαστικά με την επιλογή αλγόριθμου μάθησης. Στην περίπτωση του “Adam” λειτουργεί αποδοτικότερα ο συνδυασμός μεθόδων προεπεξεργασίας, ενώ στην περίπτωση του “SGD”, αποκλειστικά η μέθοδος ανισοτροπικής διάχυσης.

Συμπέρασμα του παραπάνω είναι ότι κάθε αλγόριθμος μάθησης βελτιστοποιεί την απόδοσή του ανά το πρόβλημα και με διαφορετικό συνδυασμό παραμέτρων και τεχνικών. Στην περίπτωση μας υψηλότερη απόδοση παρουσιάζει η χρήση του “SGD”, συνδυαστικά με την ανισοτροπική διάχυση.

### 5.3.3 Σχολιασμός Επιλογής Αλγόριθμου Μάθησης

Σημαντική παράμετρος, όπως φαίνεται και από την παρουσίαση των γραφικών και στο πρώτο αλλά και στο δεύτερο μέρος είναι η επιλογή του αλγόριθμου μάθησης. Τα μοντέλα μηχανικής μάθησης στο σύνολό τους εκπαιδεύτηκαν είτε κάνοντας χρήση του “SGD”, είτε κάνοντας χρήση του “Adam”.

Σε όλες τις περιπτώσεις που χρησιμοποιήθηκε ο αλγόριθμος “Adam”, συνδυαστικά με το δίκτυο VGGNet16, παρατηρήθηκε το πρόβλημα της πρόβλεψης μίας κατηγορίας, οπότε θα σχολιάσουμε μόνο τη χρήση του VGGNet16 συνδυαστικά με “SGD”. Σε κάθε περίπτωση χρήσης του αλγόριθμου “SGD”, ανεξάρτητα από την αρχιτεκτονική συνελκτικού νευρωνικού δικτύου, είχαμε πολύ υψηλότερες αποδόσεις σε σχέση με τη χρήση του “Adam”, με μοναδική εξαίρεση το συνδυασμό των δύο τεχνικών προεπεξεργασίας δεδομένων, όπου οι δύο αλγόριθμοι μάθησης παρουσίασαν πρακτικά ίδιες αποδόσεις. Η βελτιστοποίηση της απόδοσης, κατά τη χρήση του “SGD”, είναι αναμμενόμενη [55], καθώς ο συγκεκριμένος αλγόριθμος μάθησης, συνδυάζεται αποδοτικότερα με τη μέθοδο fine-tuning, σε σχέση με τον

“Adam”. Επίσης, ο “SGD”, παρουσιάζει μεγαλύτερες αποδόσεις, σε περιπτώσεις όπως το πρόβλημά μας, όπου το σύνολο δεδομένων εκπαίδευσης έχει μεγάλο πλήθος δειγμάτων[11].

Επίσης, από πλευράς σύγκλισης ο “SGD” απαιτεί πολύ μικρότερο πλήθος εποχών, με τις τιμές ακρίβειας να παρουσιάζουν σύγκλιση από την 20<sup>η</sup> εποχή και μετά. Σε αντίθεση με αυτό ο αλγόριθμος “Adam” απαιτεί μεγαλύτερο πλήθος εποχών, όσον αφορά την εκπαίδευση και τη σύγκλιση του, ενώ ταυτόχρονα παρουσιάζει και μεγάλες διακυμάνσεις στις διαδοχικές του τιμές στο σύνολο της εκπαίδευσης. Συνολικά, όσον αφορά το πρόβλημά μας, η περίπτωση του “SGD” εμφανίζει τόσο υψηλότερη απόδοση, όσο και ταχύτερη σύγκλιση, στην περίπτωση βέβαια, όπου έχει εφαρμοστεί προεπεξεργασία δεδομένων, πέρα από την εξομάλυνση, ανισοτροπικής διάχυσης και όχι μεθόδου διόρθωσης πεδίου πόλωσης.

#### 5.3.4 Σχολιασμός Επιλογής πλήθους επιπέδων - Αρχικοποίηση βαρών

Όσον αφορά την επιλογή του πλήθους των εκπαιδευσιμων επιπέδων για το μοντέλο τύπου ResNet50, τα πέντε εκπαιδευσιμα επίπεδα αποδίδουν ικανοποιητικά, οπότε δεν πειραματιστήκαμε με την αναλλαγή του πλήθους τους.

Όσον αφορά την περίπτωση του VGGNet16, επειδή το πλήθος των πέντε εκπαιδευσιμων επιπέδων, όπως φαίνεται και από το σχετικό γράφημα, δεν αποδίδει ικανοποιητικά και συγκεκριμένα παρατηρείται το πρόβλημα πρόβλεψης μίας κατηγορίας, προσπαθήσαμε να αλλάξουμε το πλήθος αυτό σε διαδοχικές εκπαιδεύσεις.

Συγκεκριμένα, στις περιπτώσεις όπου τα επίπεδα είναι πολλά ή λίγα σε σχέση με τα συνολικά του μοντέλου, δηλαδή 5 ή όλα 16, τουλάχιστον για την περίπτωση του VGGNet16, οδηγούμαστε σε αστοχία εκμάθησης των απαιτούμενων χαρακτηριστικών. Η μοναδική περίπτωση, όπου παρατηρείται αποτελεσματική εκπαίδευση είναι αυτή των επτά εκπαιδευσιμων επιπέδων, το οποίο είναι αναμενόμενο, διότι σύμφωνα με τη διαδικασία του fine-tuning, ένα μικρό πλήθος από τα τελικά επίπεδα πρέπει να εκπαιδευτούν, με σκοπό την επεξεργασία της ήδη υπάρχουσας πληροφορίας, η οποία και προέρχεται από τα φίλτρα στα συνελκτικά και πιο χαμηλά επίπεδα.

Επίσης, όσον αφορά τη σύγκλιση, λόγω του “SGD”, παρατηρείται για ακόμα μία φορά σταθεροποίηση της τιμής στα επτά εκπαιδευσιμα επίπεδα. Σε αντίθεση με το παραπάνω, όλα τα συστήματα που δεν ανταποκρίθηκαν στο πρόβλημα παρουσίασαν ή πολύ χαμηλή απόδοση, οπότε δεν είχε νόημα να εξετάσουμε τη σύγκλιση ή τεράστιες διακυμάνσεις, οι οποίες αντιστοιχούν σε ασταθές μοντέλο μηχανικής μάθησης. Όσον αφορά την περίπτωση αρχικοποίησης βαρών κατά “ImageNet”, σε σχέση με την τυχαία αρχικοποίησή τους, φαίνεται και από το διάγραμμα, πως η πρώτη περίπτωση συντελεί σε πολύ ανώτερη απόδοση, αλλά και ταχύτερη και πιο σταθερή σύγκλιση σε σχέση με τη δεύτερη περίπτωση.

Τελικά, ο αριθμός των εκπαιδευσιμων επιπέδων, τουλάχιστον στο πρόβλημά μας, αλλάζει σε σχέση με το σύστημα, αλλά σε κάθε περίπτωση πρέπει να είναι ένα πλήθος επιπέδων «κοντά» στην έξοδο, το οποίο συντελεί και στην αύξηση της απόδοσης, αλλά και στη σύγκλιση. Ακόμη, η τυχαία αρχικοποίηση, όσον αφορά και τη σύγκλιση, αλλά και την απόδοση, φαίνεται να επιφέρει χειρότερα αποτελέσματα, σε σχέση με την περίπτωση της αρχικοποίησης “ImageNet”. Η τελευταία, δηλαδή, παρουσιάζει και πιο σταθερή σύγκλιση, αλλά και υψηλότερη απόδοση. Σαν αποτέλεσμα, η μέθοδος του fine-tuning, παρατηρείται ότι βοηθά στην ανάπτυξη συστημάτων με υψηλές αποδόσεις και σταθερότητα σύγκλισης.

### 5.3.5 Σχολιασμός Επιλογής Σ.Ν.Δ

Στα πλαίσια της παρούσας διπλωματικής εργασίας χρησιμοποιήθηκαν τρεις διαφορετικές αρχιτεκτονικές μοντέλων μηχανικής μάθησης. Αυτές ήταν οι εξής: ResNet50, VGGNet16 και η Custom αρχιτεκτονική που υλοποιήθηκε από εμάς.

Παρατηρήθηκε, λοιπόν, ότι ο συνδυασμός μεθόδων προεπεξεργασίας δεδομένων, η επιλογή αλγόριθμου μάθησης, αλλά και η εναλλαγή εκπαιδευσιμων επιπέδων, σε συνδυασμό με την αρχικοποίηση “ImageNet”, για τις δύο πρώτες περιπτώσεις αρχιτεκτονικών είχε σαν αποτέλεσμα διαφορετικές μετρικές, οι οποίες βελτιστοποιούνταν κυρίως με την εναλλαγή αυτών των παραμέτρων και όχι κατά την επιλογή των μοντέλων μηχανικής μάθησης. Δηλαδή, μπορεί το μοντέλο μηχανικής μάθησης ResNet50, σε συνδυασμό με ανισοτροπική διάχυση και αλγόριθμο μάθησης “SGD”, να επέφερε την υψηλότερη απόδοση, αλλά αυτή ήταν ελάχιστα υψηλότερη από αυτή των άλλων δύο αρχιτεκτονικών, ακόμα και στην περίπτωση του Custom δικτύου, το αποτελούταν από πολύ μικρότερο πλήθος συνελικτικών επιπέδων.

Το συμπέρασμα το οποίο προκύπτει, οπότε, από το παραπάνω είναι ότι η επιλογή του μοντέλου δεν έχει την ίδια σημαντικότητα με την επιλογή λοιπών παραμέτρων και μεθόδων προεπεξεργασίας δεδομένων. Το παραπάνω αιτιολογείται από το γεγονός ότι τα χαρακτηριστικά, τα οποία πρέπει να αναγνωριστούν από το σύστημα, προκειμένου αυτό να αποδόσει δεν εμφανίζουν μεγάλη πολυπλοκότητα. Σαν αποτέλεσμα ακόμα και μία πιο ρηχή αρχιτεκτονική, με σαφώς μικρότερο πλήθος παραμέτρων αρκεί για να τα αναγνωρίσει.

Οπότε, καταλήγουμε στο ότι, όσον αφορά την αντιμετώπιση του συγκεκριμένου προβλήματος, μία πιο ρηχή αρχιτεκτονική, αλλά με κατάλληλο συνδυασμό υπερπαραμέτρων και μεθόδων προεπεξεργασίας, όπως φαίνεται και από το τελευταίο διάγραμμα του πρώτου μέρους, μπορεί να παρουσιάσει υψηλότερη απόδοση από μία πολύ πιο «ακριβή» και σύνθετη αρχιτεκτονική συνελικτικού νευρωνικού δικτύου από πλευράς παραμέτρων. Ακόμα και στη περίπτωση που εφαρμοστεούν και στα τρία συστήματα οι ίδιες τεχνικές πάλι η ρηχή αρχιτεκτονική πολύ «κοντά» σε απόδοση με τις άλλες δύο, με σημαντικό πλεονέκτημα την ελαχιστοποίηση της απαιτούμενης μνήμης, η οποία απαιτείται από το σύστημα.

## Βιβλιογραφία

---

- [1] Jemal, Ahmedin, et al. "Cancer statistics, 2007." *CA: a cancer journal for clinicians* 57.1 (2007): 43-66.
- [2] Lippmann, Richard. "An introduction to computing with neural nets." *IEEE Assp magazine* 4.2 (1987): 4-22
- [3] Pereira, Sérgio, et al. "Brain tumor segmentation using convolutional neural networks in MRI images." *IEEE transactions on medical imaging* 35.5 (2016): 1240-1251.
- [4] Hoo-Chang, Shin, et al. "Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning." *IEEE transactions on medical imaging* 35.5 (2016): 1285.
- [5] Friston, Karl J., et al. "Spatial registration and normalization of images." *Human brain mapping* 3.3 (1995): 165-189.
- [6] Macaluso, Sebastian, and David Shih. "Pulling Out All the Tops with Computer Vision and Deep Learning." *arXiv preprint arXiv:1803.00107* (2018)
- [7] Menze, Bjoern H., et al. "The multimodal brain tumor image segmentation benchmark (BRATS)." *IEEE transactions on medical imaging* 34.10 (2015): 1993.
- [8] Szegedy, Christian, et al. "Inception-v4, inception-resnet and the impact of residual connections on learning." *AAAI*. Vol. 4. 2017.
- [9] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556*(2014).
- [10] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014)
- [11] Bottou, Léon. "Large-scale machine learning with stochastic gradient descent." *Proceedings of COMPSTAT'2010*. Physica-Verlag HD, 2010. 177-186
- [12] Russell, Stuart J., and Peter Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.
- [13] Goldberg, David E., and John H. Holland. "Genetic algorithms and machine learning." *Machine learning* 3.2 (1988): 95-99
- [14] Guzella, Thiago S., and Walmir M. Caminhas. "A review of machine learning approaches to spam filtering." *Expert Systems with Applications* 36.7 (2009): 10206-10222
- [15] Netzer, Yuval, et al. "Reading digits in natural images with unsupervised feature learning." *NIPS workshop on deep learning and unsupervised feature learning*. Vol. 2011. No. 2. 2011.
- [16] McCallum, Andrew, et al. "A machine learning approach to building domain-specific search engines." *IJCAI*. Vol. 99. 1999.
- [17] Cipolla, Roberto, and Sebastiano Battiato. *Machine Learning for Computer Vision*. Ed. Giovanni Maria Farinella. Vol. 5. Springer, 2013.
- [18] Møller, Martin Fodstlette. "A scaled conjugate gradient algorithm for fast supervised learning." *Neural networks* 6.4 (1993): 525-533.

- [19] Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. "Unsupervised learning." *The elements of statistical learning*. Springer, New York, NY, 2009. 485-585.
- [20] Baluja, Shumeet. *Population-based incremental learning. a method for integrating genetic search based function optimization and competitive learning*. No. CMU-CS-94-163. Carnegie-Mellon Univ Pittsburgh Pa Dept Of Computer Science, 1994.
- [21] Nachev, Anatoli, and Borislav Stoyanov. "An Approach to Computer Aided Diagnosis by Multi-Layer Preceptrons." *IC-AI*. 2010.
- [22] Werbos, Paul J. "Backpropagation through time: what it does and how to do it." *Proceedings of the IEEE* 78.10 (1990): 1550-1560.
- [23] Waterman, Donald. "A guide to expert systems." (1986).
- [24] Li, Jia, et al. "Hyperbolic linear units for deep convolutional neural networks." *Neural Networks (IJCNN), 2016 International Joint Conference on*. IEEE, 2016.
- [25] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European conference on computer vision*. Springer, Cham, 2014.
- [26] Krizhevsky, Alex, and G. Hinton. "Convolutional deep belief networks on cifar-10." *Unpublished manuscript* 40.7 (2010).
- [27] Cogswell, Michael, et al. "Reducing overfitting in deep networks by decorrelating representations." *arXiv preprint arXiv:1511.06068* (2015).
- [28] Liu, Lingqiao, Chunhua Shen, and Anton van den Hengel. "The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015
- [29] Scherer, Dominik, Andreas Müller, and Sven Behnke. "Evaluation of pooling operations in convolutional architectures for object recognition." *Artificial Neural Networks–ICANN 2010*. Springer, Berlin, Heidelberg, 2010. 92-101.
- [30] Chen, Liang-Chieh, et al. "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs." *IEEE transactions on pattern analysis and machine intelligence* 40.4 (2018): 834-848.
- [31] Dahl, George E., Tara N. Sainath, and Geoffrey E. Hinton. "Improving deep neural networks for LVCSR using rectified linear units and dropout." *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*. IEEE, 2013.
- [32] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- [33] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
- [34] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016
- [35] Kudo, Yasunori, and Yoshimitsu Aoki. "Dilated convolutions for image classification and object localization." *Machine Vision Applications (MVA), 2017 Fifteenth IAPR International Conference on*. IEEE, 2017
- [36] Walker, Jacob, et al. "An uncertain future: Forecasting from static images using variational autoencoders." *European Conference on Computer Vision*. Springer, Cham, 2016

- [37] Goodfellow, Ian, et al. "Generative adversarial nets." *Advances in neural information processing systems*. 2014.
- [38] Long, Jonathan, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
- [39] Ballester, Pedro, and Ricardo Matsumura de Araújo. "On the Performance of GoogLeNet and AlexNet Applied to Sketches." *AAAI*. 2016
- [40] Tolias, Giorgos, Ronan Sicre, and Hervé Jégou. "Particular object retrieval with integral max-pooling of CNN activations." *arXiv preprint arXiv:1511.05879* (2015)
- [41] Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
- [42] Fan, Yang, et al. "Learning What Data to Learn." *arXiv preprint arXiv:1702.08635* (2017)
- [43] Dozat, Timothy. "Incorporating nesterov momentum into adam." (2016).
- [44] LeCun, Yann. "LeNet-5, convolutional neural networks." URL: <http://yann.lecun.com/exdb/lenet>(2015): 20.
- [45] Zeiler, Matthew D., and Rob Fergus. "Visualizing and understanding convolutional networks." *European conference on computer vision*. Springer, Cham, 2014.
- [46] Zhong, Zhuoyao, Lianwen Jin, and Zecheng Xie. "High performance offline handwritten chinese character recognition using googlenet and directional feature maps." *Document Analysis and Recognition (ICDAR), 2015 13th International Conference on*. IEEE, 2015
- [47] Szegedy, Christian, et al. "Scalable, high-quality object detection." *arXiv preprint arXiv:1412.1441*(2014).
- [48] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- [49] Friston, Karl J., et al. "Spatial registration and normalization of images." *Human brain mapping*3.3 (1995): 165-189.
- [50] Weickert, Joachim. *Anisotropic diffusion in image processing*. Vol. 1. Stuttgart: Teubner, 1998.
- [51] Tustison, N., and J. Gee. "N4ITK: Nick's N3 ITK implementation for MRI bias field correction." *Insight Journal* 9 (2009)
- [52] Van der Aalst, Wil MP, et al. "Process mining: a two-step approach to balance between underfitting and overfitting." *Software & Systems Modeling* 9.1 (2010): 87
- [53] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*. 2012
- [54] Tajbakhsh, Nima, et al. "Convolutional neural networks for medical image analysis: Full training or fine tuning?." *IEEE transactions on medical imaging* 35.5 (2016): 1299-1312.
- [55] Wilson, Ashia C., et al. "The marginal value of adaptive gradient methods in machine learning." *Advances in Neural Information Processing Systems*. 2017.



