



Πολυτεχνείο Κρήτης  
Τμήμα Μηχανικών Ορυκτών Πόρων

#### **ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ**

**Αυτόματη ταξινόμηση ορυκτών μέσω υλοποίησης συνελκτικού νευρωνικού δικτύου**

Κυριακόπουλος Δημήτριος  
ΑΜ: 2013020046

Εξεταστική επιτροπή  
Παναγιώτης Παρτσινέβελος,  
Καθηγητής ΜΗΧΟΠ ,επιβλέπων,  
Μιχαήλ Γαλετάκης, Καθηγητής ΜΗΧΟΠ,  
Γεώργιος Χρηστίδης, Καθηγητής ΜΗΧΟΠ

Χανιά, 07/11/2023





## Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω τα μέλη της τριμελούς επιτροπής. Πιο συγκεκριμένα τον κύριο Χρηστίδη για τις πολύτιμες συμβουλές του, τον κύριο Γαλετάκη για την αναφορά τον νευρωνικών δικτύων σε μάθημά του που υπήρξε πηγή έμπνευσης. Τέλος την εξαιρετική ομάδα του Senselab για την άψογη συνεργασία και φιλοξενία. Πιο συγκεκριμένα για την υπομονή και καθοδήγηση έναν εκπληκτικό διδάκτορα τον κύριο Γιώργο Πετράκη. Τίποτα από όλα αυτά δεν θα ήταν εφικτό χωρίς τον κύριο Παρτσινέβελο που με βοήθησε σε κάθε βήμα της ακαδημαϊκής μου πορείας και φυσικά για την ανάθεση της διπλωματικής. Η υποστήριξη και η φιλική ατμόσφαιρα στο εργαστήριο κατέστησαν δυνατή την επίτευξη των επιδιωκόμενων στόχων.

## Περιεχόμενα

Ευχαριστίες	4
Περιεχόμενα	5
Περίληψη	8
1. Εισαγωγή	9
1.1. Υπόβαθρο	9
1.2. Καθορισμός προβλήματος & Κίνητρο	9
1.3. Σκοπός και επιμέρους στόχοι	10
1.4. Ερευνητική Μεθοδολογία	10
1.5. Παραδοχές	11
1.6. Προσδοκώμενα αποτελέσματα	11
1.7. Απήχηση στην κοινωνία	11
1.8. Εφαρμογές	12
2. Θεωρητικό υπόβαθρο	13
2.1. Εισαγωγή	13
2.2. Ψηφιακή εικόνα	13
2.3. Τεχνικές εντοπισμού περιοχών ενδιαφέροντος	17
2.3.1. Εξισορρόπηση Ιστογράμματος (Histogram Equalization)	18
2.3.2. Κατωφλίωση ή Δυναδοποίηση (Thresholding – Binarization)	18
2.3.3. Συσταδοποίηση (Clustering)	20
2.3.4. Λαπλασιανό Φίλτρο (Laplacian Filter)	21
2.4. Μέθοδοι ταξινόμησης	22
2.4.1. Παραδοσιακοί ταξινομητές μηχανικής μάθησης	22
2.4.1.1. Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines)	22
2.4.1.2. Ταξινομητές Τυχαίων Δασών (Random Forest Classifiers)	23
2.4.1.3. Ταξινομητής K-Κοντινότερων Γειτόνων (K-Nearest Neighbors Classifier)	25
2.4.2. Ταξινομητές Στατιστικής Ανάλυσης (Statistical Classifiers)	26
2.4.2.1. Μπαεσιανός Ταξινομητής (Bayes Classifier)	26
2.4.2.2. «Αφελής» Μπαεσιανός Ταξινομητής (Naive Bayes Classifier)	26
2.4.3. Τεχνητά Νευρωνικά Δίκτυα (ΤΝΔ) (Artificial Neural Networks)	27
3. Τεχνητή Νοημοσύνη	29
3.1. Εισαγωγή	29
3.2. Η εξελικτική πορεία της τεχνητής νοημοσύνης	29
3.3. Μηχανική Μάθηση (Machine Learning)	31
3.4. Νευρωνικά Δίκτυα	32

3.4.1. Βιολογικά Νευρωνικά Δίκτυα	33
3.4.2. Τεχνητά Νευρωνικά Δίκτυα	34
3.4.2.1. Συνάρτηση Ενεργοποίησης (Activation Function)	37
3.4.2.2. Συνάρτηση Σφάλματος (Loss Function)	41
3.4.2.3. Οπισθοδιάδοση Σφάλματος (Backpropagation)	43
3.4.2.4. Ρυθμός Μάθησης	44
3.4.2.4. Τεχνικές Βελτιστοποίησης (Optimization)	45
3.4.2.5. Επιλογή Αρχιτεκτονικής	47
3.4.2.6. Δημιουργία του συνόλου δεδομένων εκπαίδευσης	48
3.4.2.7. Διαδικασία Εκπαίδευσης	49
3.4.2.8. Χρήση του εκπαιδευμένου μοντέλου	50
4. Συνελικτικά Νευρωνικά Δίκτυα (CNNs)	51
4.1. Εισαγωγή	51
4.2. Δομή των Συνελικτικών Νευρωνικών Δικτύων	52
4.2.1. Συνελικτικά Επίπεδα (Convolutional Layers)	53
4.2.2 Μη-Γραμμικά Επίπεδα (Non-Linear Layers)	55
4.2.3. Συγκεντρωτικά Επίπεδα (Pooling Layers)	56
4.2.4. Πλήρως Συνδεδεμένα Επίπεδα (Fully Connected Layers)	57
4.3. Τεχνικές Βελτίωσης της απόδοσης των ΤΝΔ	58
4.3.1. L1 & L2 Κανονικοποίηση (L1 & L2 Regularization)	58
4.3.2. Dropout	59
4.3.3. DropConnect	59
4.3.4. Πρόωρη Διακοπή (Early Stopping)	60
4.3.5. Επαύξηση Δεδομένων (Data Augmentation)	60
4.3.6. Κανονικοποίηση Δεσμίδας (Batch Normalization)	61
4.3.7. Κανονικοποίηση Επιπέδου (Layer Normalization)	62
4.4. Μετρικές αξιολόγησης της απόδοσης των ΣΝΔ	63
5. Πειραματική διαδικασία	67
5.1. Εισαγωγή	67
5.2. Σύνολο Δεδομένων	67
5.3. Δημιουργία Απαραίτητων Υπο-συνόλων	68
5.4. Μοντέλο 1 <sup>ο</sup> : Προσαρμοσμένη αρχιτεκτονική ΣΝΔ	71
5.5. Μοντέλο 2 <sup>ο</sup> : Συνελικτικό Νευρωνικό Δίκτυο VGG16	83
6. Ερμηνεία αποτελεσμάτων	89
8. Βιβλιογραφία - Αναφορές	91



## Περίληψη

Είναι ευρέως γνωστό ότι η παραγωγή ορυκτών πρώτων υλών στηρίζεται στην ανίχνευση και τον εντοπισμό νέων κοιτασμάτων καθώς και στον καλύτερο διαχωρισμό και εμπλουτισμό των ήδη υπαρχόντων. Η παρούσα εργασία έχει ως σκοπό την μελέτη και αξιολόγηση της αρχιτεκτονικής ενός συνελκτικού νευρωνικού δικτύου το οποίο υλοποιήθηκε στο πλαίσιο της παρούσας διπλωματικής για την αναγνώριση ορυκτών μέσω εικόνων. Λόγω έλλειψης διαθέσιμων βάσεων δεδομένων που χρησιμοποιούνται ως επί το πλείστον στην εκπαίδευση των νευρωνικών δικτύων βαθιάς μάθησης, οι εικόνες συλλέχθηκαν χειροκίνητα από ανοιχτές πηγές. Τα δεδομένα διαχωρίστηκαν σε κατηγορίες εκπαίδευσης και επαλήθευσης ενώ επίσης χρησιμοποιήθηκαν τεχνικές εμπλουτισμού-αύξησης δεδομένων (data augmentation). Επιπροσθέτως, η βάση δεδομένων εμπλουτίστηκε με εικόνες που έχουν μετασχηματισθεί βάσει του υψηλοδιαβατού φίλτρου συνέλιξης Laplace για τον εντοπισμό ακμών, εξερευνώντας την υπόθεση περί Νόμου της συμμετρίας των κρυσταλλικών σχημάτων η οποία αναφέρει πως η διάταξη των ατόμων ή των μορίων σε έναν κρύσταλλο έχει συγκεκριμένες συμμετρικές ιδιότητες που αντικατοπτρίζονται στις εξωτερικές μορφές του κρυστάλλου. Επιλέχθηκαν σημαντικές παράμετροι όπως η αρχιτεκτονική του δικτύου και το πλήθος των εικόνων ανά εκπαίδευση ενώ το δίκτυο βελτιστοποιήθηκε και ελέγχθηκε ως προς την ακρίβειά του. Το δίκτυο συγκρίθηκε με το νευρωνικό δίκτυο βαθιάς μάθησης VGG16, γνωστό για εφαρμογές αναγνώρισης εικόνων, το οποίο εκπαιδεύτηκε με την ίδια βάση δεδομένων. Τα αποτελέσματα έδειξαν ότι το υλοποιημένο νευρωνικό δίκτυο είναι αξιόπιστο με αποτελέσματα ακρίβειας και ανάκλησης περίπου στο 70%. Οι κατηγορίες που συμπεριλήφθηκαν στο μοντέλο περιλαμβάνουν τις παρακάτω κλάσεις ορυκτών: χαλαζίας, βιοτίτης, αιματίτης, ασβεστίτης, αζουρίτης, κινναβαρίτης, μαλαχίτης, κίτρινη σανδαράχη, και τιρκουάζ. Θα μπορούσαν τα αποτελέσματα να ήταν καλύτερα αν υπήρχαν περισσότερες πληροφορίες διαθέσιμες.



## 1. Εισαγωγή

### 1.1. Υπόβαθρο

Στην παρούσα πτυχιακή εργασία έχει δοθεί ιδιαίτερη βαρύτητα στην δημιουργία ενός νευρωνικού δικτύου που θα επιτρέψει την αναγνώριση ορυκτών όπως ο χαλαζίας, ο ασβεστίτης, ο Βιοτίτης κ.α. μόνο μέσω της επεξεργασίας εικόνας. Γιατί είναι σημαντική η αναγνώριση ορυκτών μέσω εικόνων; Για να απαντήσουμε σε αυτή την ερώτηση θα πρέπει να επικεντρωθούμε σε κάποιες από τις ανάγκες του σύγχρονου κόσμου ως προς την αναγνώριση ορυκτών γενικότερα. Για ερευνητικούς λόγους η ταυτοποίηση καθώς και η ταξινόμηση ορυκτών είναι χρήσιμη για γεωλογικές και ορυκτολογικές μελέτες όπως, για την κατανόηση των συνθηκών σχηματισμού της Γης. Δεύτερον η βιομηχανικές εφαρμογές καθώς και η διαχείριση πόρων εξαρτάται από την σωστή και ακριβής ταυτοποίηση. Τέλος για εκπαιδευτικούς σκοπούς και περιβαλλοντικές επιπτώσεις καθώς οι εικόνες αποτελούν εργαλείο εκπαίδευσης και δεν απαιτείται η ανθρώπινη επαφή σε επιβλαβείς ουσίες. Οι σύγχρονες διαδικασίες ταξινόμησης βασίζονται σε προηγμένες τεχνικές ανάλυσης και βελτιστοποίησης βαθιών νευρωνικών δικτύων (Deep Neural Networks), τα οποία σήμερα αποτελούν μία σύγχρονη προσέγγιση της μηχανικής μάθησης (Machine Learning), την βαθιά μάθηση (Deep Learning). Θεωρούνται από τις πιο αξιόπιστες τεχνικές αναγνώρισης προτύπων και βρίσκουν ευρεία εφαρμογή στη γεωλογία για στην ανάλυση και ταξινόμηση γεωλογικών δεδομένων.

### 1.2. Καθορισμός προβλήματος & Κίνητρο

Είναι πλέον γνωστό ότι η παραγωγή στηρίζεται στην ανίχνευση και τον εντοπισμό νέων κοιτασμάτων καθώς και στον καλύτερο διαχωρισμό των ειδών υπαρχών. Με την χρήση νευρωνικών δικτύων και την επεξεργασία εικόνας είναι εφικτό να βελτιώσουμε τις μεθόδους διαχωρισμού που είδη βρίσκονται στην παραγωγική διαδικασία. Η συμβολή τους μπορεί να επιτύχει μεγαλύτερη ακρίβεια, μείωση του χρόνου εμπλουτισμού μικρότερο περιβαλλοντικό κόστος. Η εφαρμοστικότητα τους είναι εύκολη και συμβατή με την τεχνολογία που εφαρμόζεται και επιτρέπει την τηλεργασία και τηλεπισκόπιση, δηλαδή δεν απαιτεί ανθρώπους στο πεδίο και αποστολή δειγμάτων σε εργαστήριο. Ένας από τους στόχους αυτής της εργασίας είναι η γνωστοποίηση του προβλήματος διότι δεν υπάρχει αρκετή διαθέσιμη πληροφορία στο κλάδο. Καθώς και η βελτιστοποίηση των δυνατοτήτων επεξεργασίας εικόνων που θα επιφέρει ακριβέστερη και πληθέστερη αναγνώριση προτύπων και ορυκτών .

### 1.3. Σκοπός και επιμέρους στόχοι

Ο σκοπός μας είναι η κατασκευή και εκπαίδευση ενός νευρωνικού δικτύου για την κατηγοριοποίηση εικόνων και πιο συγκεκριμένα στην εξειδίκευση ορυκτών όπως ο Χαλαζίας, ο Ασβεστίτης, ο Βιοτίτης και άλλων. Επιπλέον στόχος είναι η επίδειξη των δυνατοτήτων ενός τέτοιου δικτύου ώστε να γίνει ένα χρήσιμο εργαλείο για γρήγορο, εύελκτο εντοπισμό και παράλληλα διαχωρισμό ορυκτών. Ο διαχωρισμός ορυκτών είναι εφικτός με διάφορες μεθόδους εμπλουτισμού, όπως ο φυσικός διαχωρισμός (Κομνίτσας, Κ. (2017). Εμπλουτισμός Μεταλλευμάτων), ωστόσο ή χρήση τεχνητής νοημοσύνης σε συνδυασμό με την χρήση διαχωριστών είναι κάτι που ερευνάται τα τελευταία χρόνια. Οι κύριες ερευνητικές ερωτήσεις της παρούσας εργασίας είναι οι ακόλουθες:

- Θα μπορούσε να εφαρμοστεί ο Νόμος της συμμετρίας των κρυσταλλικών σχημάτων για την κατηγοριοποίηση ορυκτών μέσω εικόνων;
- Θα μπορούσε η τεχνητή νοημοσύνη σε συνδυασμό με τα νευρωνικά δίκτυα να αποτελεί βελτίωση του εμπλουτισμού και της παραγωγικής διαδικασίας;

Επιπλέον, προσωπικός στόχος είναι η κατάρτιση στις σύγχρονες τεχνολογίες προγραμματισμού και η κατανόηση της τεχνητής νοημοσύνης και νευρωνικών δικτύων.

### 1.4. Ερευνητική Μεθοδολογία

Για την κατασκευή του νευρωνικού δικτύου τα εργαλεία και η μεθοδολογία που χρησιμοποιήθηκαν ήταν τα εξής: η python(έκδοση 3.11.5) ως γλώσσα προγραμματισμού και το λογισμικό CUDA (Compute Unified Device Architecture, έκδοση 12.1) ως εργαλείο της κάρτας γραφικών. Η ύπαρξη μιας βιβλιοθήκης ορυκτών (κυρίως εικόνων) είναι απαραίτητη για την σωστή και επαρκή εκπαίδευση κάθε νευρωνικού δικτύου. Αφού όμως δεν υπήρχε καμία διαθέσιμη βάση δεδομένων, έγινε συλλογή δεδομένων χειροκίνητα με την επιλογή εικόνων μία προς μία από ανοιχτές πηγές πληροφοριών. Στην συνέχεια, έγινε η επιλογή κάποιων παραμέτρων όπως είναι η αρχιτεκτονική του δικτύου και η χρήση συνελκτικών νευρωνικών δικτύων, για την εκπαίδευση όπως είναι το μέγεθος παρτίδας εικόνων. Μαζί με την επιλογή των σημαντικών παραμέτρων, το νευρωνικό δίκτυο βελτιστοποιήθηκε και έγινε έλεγχος της ποιότητας του αλγορίθμου ως προς την ακρίβεια, την η ανάκληση, κ.α. Έγινε σύγκριση του προγραμματισμένου νευρωνικού δικτύου με γνωστό στον τομέα κατηγοριοποίησης εικόνων VGG16 (Visual Geometry Group 16) υστέρη από την εκπαίδευση του τελευταίου με την βάση δεδομένων που συλλέχθηκε. Τελευταίο αλλά εξίσου σημαντικό, έγινε εμπλουτισμός της βάσης δεδομένων με χρήση του φίλτρου Laplace στις εικόνες για τον εντοπισμό ακμών με σκοπό την εξερεύνηση της υπόθεσης μας περί Νόμου της συμμετρίας των κρυσταλλικών σχημάτων και την χρήση του μέσω εικόνων.

### 1.5. Παραδοχές

Για την άσκηση αυτής της πειραματικής διαδικασίας οι παραδοχές που υπήρξαν ήταν οι εξής:

1. Οι εικόνες για την δημιουργία της βάσης των δεδομένων ήταν αρκετές για την εκπαίδευση και την επαλήθευση του νευρωνικού δικτύου. Ωστόσο είναι γνωστό ότι για την διασφάλιση της ποιότητας εκπαίδευσης του νευρωνικού δικτύου είναι απαραίτητο να διασφαλιστεί ότι οι εικόνες καλύπτουν το εύρος των διαφορετικών συνθηκών και μεταβλητών που μπορεί να συναντηθούν στον πραγματικό κόσμο. Τέτοιες μπορεί να είναι διάφορες γωνίες λήψης, φωτισμός, θερμοκρασία, και πίεση, παράγοντες που επηρεάζουν, συν τοις άλλοις, και την εξωτερική εμφάνιση των ορυκτών.
2. Η χειροκίνητη κατηγοριοποίηση των εικόνων ήταν ορθή και επαρκής.
3. Ως προς την εφαρμογή του νευρωνικού δικτύου, αυτό μπορεί να συμπεριλάβει για όλα τα ορυκτά αν υπάρχουν αρκετές πληροφορίες (εικόνες). Ωστόσο για την απλοποίηση του προβλήματος επιλέχθηκαν τα 10 ορυκτά (και διαθέσιμα στο διαδίκτυο) για τον σχηματισμό των κλάσεων.
4. Η βελτιστοποίηση των δεδομένων σε συνδυασμό με τη χρήση του φίλτρου La Place κάνει εμφανείς τις γωνίες των ορυκτών ώστε να υπάρχει σύγκριση από το δίκτυο.

### 1.6. Προσδοκώμενα αποτελέσματα

Τελικό αποτέλεσμα αυτής της εργασίας είναι η ολοκλήρωση μιας βάσης δεδομένων και η κατασκευή ενός συνελκτικού νευρωνικού δικτύου. Το δίκτυο θα είναι κατάλληλα εκπαιδευμένο ώστε να μπορεί να έχει υψηλή αναγνώριση και ανάκληση και να συγκριθεί σε δυνατότητα με το γνωστό προ εκπαιδευμένο δίκτυο VGG16.

### 1.7. Απήχηση στην κοινωνία

Είναι ευρέως γνωστό ότι υπάρχουν πολλά πλεονεκτήματα στην χρήση νευρωνικών δικτύων. Η χρήση νευρωνικών δικτύων στην ορυκτολογία προσφέρει αυτοματοποίηση και ανάλυση μεγάλων δεδομένων. Αρχικά δεν είναι απαραίτητη η επιστημονική κατάρτιση για την εισαγωγή ανθρώπων στον κόσμο της ορυκτολογίας. Με την χρήση φορητής κάμερας (κινητό τηλέφωνο) θα μπορεί να υπάρχει μια ενδεικτική αναγνώριση ορυκτών. Για την παραγωγική διαδικασία δεν θα είναι απαραίτητη η εντατική εκπαίδευση του προσωπικού και εξελίσσονται προοπτικές για απομακρυσμένες συνθήκες εργασίας. Που είναι ασφαλέστερες για τον άνθρωπο διότι δεν χρειάζεται να έρθει σε επαφή με επιβλαβείς ουσίες ή να επισκεφτεί δυσπρόσιτες περιοχές

για την λήψη εικόνων. Πολύ σημαντικό είναι να αναφερθεί ότι η χρήση αυτής της τεχνολογίας αποτελεί ένα εργαλείο για την εξέλιξη της παραγωγής και όχι τρόπο αντικατάστασης κατάλληλα εκπαιδευμένων ανθρώπων όπως οι Ορυκτολόγοι και οι Μηχανικοί. Αντίθετα εδραιώνεται η θέση τους για την αποφυγή προκατειλημμένων αποτελεσμάτων και την λανθασμένη ερμηνεία τους.

#### 1.8. Εφαρμογές

Είναι εύκολα αντιληπτό πως υπάρχουν πολλές εφαρμογές των νευρωνικών δικτύων και επεξεργασίας εικόνων είναι πολλές. Μερικές από αυτές είναι η ανίχνευση και ο εντοπισμός ορυκτών από είδη λειτουργικά μέσα όπως είναι δορυφόροι και μη επανδρωμένα εργαλεία συλλογής εικόνων. Η εγκατάσταση τους σε συστήματα όπως οι διαχωριστές θα συμβάλει εντατικά στην μεγαλύτερη ακρίβεια των τελικών προϊόντων (καθαρότερα υλικά). Όπως και στους χρόνους διαχωρισμού δεν είναι αναγκαίες οι μεταφορές δείγματος υλικού στο εργαστήριο για πρόχειρο υπολογισμό της περιεκτικότητας. ( *Wei Lou, Dexian Zhang, Review of mineral recognition and its future ,Applied Geochemistry, Volume 122,2020,*)

## 2. Θεωρητικό υπόβαθρο

### 2.1. Εισαγωγή

Η ορυκτολογία και οι γεωλογικές επιστήμες επικεντρώνονται σε μεγάλο βαθμό στην ανάλυση της σύνθεσης της Γης, στην εξερεύνηση της γεωλογικής ιστορίας και στον εντοπισμό πολύτιμων αναξιοποίητων πόρων. Όπως είναι γνωστό η χρήση και η λήψη εικόνων στον τομέα είναι πολύ διαδεδομένη.

Στο πλαίσιο αυτό, η χρήση εξελιγμένων ταξινομητών εικόνων και τεχνικών ανίχνευσης περιοχών ενδιαφέροντος παίζει καθοριστικό ρόλο στη βελτίωση της αποτελεσματικότητας και της ακρίβειας της ανάλυσης εικόνων ορυκτών. Αυτά τα εργαλεία επιτρέπουν σε γεωλόγους και ερευνητές να αποκτήσουν βαθύτερες γνώσεις σχετικά με τους γεωλογικούς σχηματισμούς, να εντοπίζουν τα ορυκτά με ακρίβεια και να εξάγουν κρίσιμα δεδομένα από ψηφιακές εικόνες ορυκτών .

Οι ταξινομητές αποτελούν την καρδιά της αυτοματοποιημένης κατηγοριοποίησης εικόνων ορυκτών. Σήμερα, οι μηχανισμοί αυτοί συμβάλουν στο να ανιχνεύουν διάφορες ιδιότητες και χαρακτηριστικά των ορυκτών μέσω της μελέτης εικονικών αναπαραστάσεων, εμφανίζοντας παρόμοια επίπεδα ακρίβειας (ή ξεπερνώντας τα) με επαγγελματίες γεωλόγους και εξειδικευμένους ερευνητές. Στην περίπτωση της ανίχνευσης και ανάλυσης γεωλογικών δεδομένων, οι ταξινομητές, σε συνδυασμό με αλγόριθμους επεξεργασίας εικόνας (image processing), βρίσκουν μεγάλη εφαρμογή στην παροχή υποβοηθητικής ανάλυσης. Όπως γνωρίζουμε, οι γεωλόγοι πραγματοποιούν αναλύσεις με βάση την εκπαίδευση, την εμπειρία και την ατομική τους κρίση, κάτι που μπορεί να δημιουργήσει υποκειμενικότητα στην ερμηνεία των

### 2.2. Ψηφιακή εικόνα

Μια ψηφιακή εικόνα  $i[m, n]$  αναπτύσσεται σε ένα διακριτό χώρο δύο διαστάσεων και παράγεται από την ψηφιοποίηση μιας αναλογικής εικόνας  $i[x, y]$  που αναπτύσσεται σε ένα συνεχή χώρο με διαστάσεις  $x$  και  $y$ . Κατά την ψηφιοποίηση, η συνεχής εικόνα διαιρείται σε  $M$  σειρές και  $N$  στήλες. Τα σημεία τομής των σειρών με τις στήλες είναι τα εικονοστοιχεία (pixels), και αποτελούν το δομικό στοιχείο κάθε εικόνας. Έτσι, μία ψηφιακή εικόνα διαστάσεων  $M \times N$  γενικά παριστάνεται από ένα δυσδιάστατο πίνακα διαστάσεων  $(i, j)$ , όπου σε κάθε κελί (εικονοστοιχείο) αποθηκεύονται διάφορες τιμές. Η φωτεινότητα της κάθε ψηφιακής εικόνας είναι ανάλογη της τιμής  $I(i, j)$  του κάθε εικονοστοιχείου  $(i, j)$ . Συνεπώς, ο πίνακας  $I(i, j)$  είναι μια διακεκριμένη συνάρτηση που εκφράζει την ένταση της φωτεινότητας σε κάθε εικονοστοιχείο.

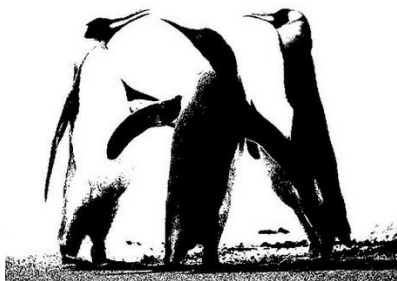
Οι ψηφιακές εικόνες κυριάρχησαν με την ραγδαία ανάπτυξη της τεχνολογίας. Μέσα από την πληθώρα των τεχνολογιών που προέκυψε, προτάθηκαν και διάφοροι τύποι

ψηφιακών εικόνων. Επικράτησαν όμως δύο μεγάλες κατηγορίες: τα αρχεία τύπου **ψηφιοκουκίδων (raster)** και τα **διανυσματικά (vector)** αρχεία εικόνας.

Η διαφορά μεταξύ των δύο κατηγοριών είναι στο ότι τα αρχεία τύπου ψηφιοκουκίδων σχηματίζονται από το άθροισμα ψηφιοκουκίδων (pixels) ενώ στα διανυσματικά η εικόνα σχηματίζεται από την αποθήκευση πληροφοριών σχετικά με γεωμετρικά σχήματα (γραμμές, κύκλοι, πολύγωνα, κείμενο), κάνοντας τα κατάλληλα για λογότυπα, σχέδια, γραφικά κ.α. Οι ψηφιακές εικόνες τύπου raster κυριάρχησαν σε εφαρμογές όπως η φωτογραφία και η επεξεργασία εικόνας, επειδή μπορούν να αναπαραστήσουν λεπτομέρειες και αποχρώσεις με μεγάλη ακρίβεια.

Για την παραγωγή μίας ψηφιακής φωτογραφίας απαιτούνται μέσα δειγματοληψίας, όπως ψηφιακές φωτογραφικές μηχανές, κάμερες και σαρωτές. Ανάλογα με το μέσο παραγωγής της μπορεί να είναι δυαδική εικόνα (binary image), μονοχρωματική αποχρώσεων του γκρι (gray-scale image) ή έγχρωμη εικόνα (colour image):

- **Δυαδική εικόνα (Binary image):** Σε δυαδικές εικόνες, κάθε pixel μπορεί να πάρει μόνο δύο τιμές, συνήθως 0 (μαύρο) ή 1 (λευκό). Χρησιμοποιούνται για την απεικόνιση αντικειμένων όπου ενδιαφέρον έχει μόνο η παρουσία ή η απουσία χαρακτηριστικών. Για παράδειγμα, δυαδικές εικόνες χρησιμοποιούνται συχνά στην αναγνώριση αντικειμένων, στην κατάτμηση εικόνας και στην ανίχνευση ακμών.



**Εικόνα 2.1.** Αναπαράσταση δυαδικής εικόνας με μαύρο και άσπρο χρώμα

- **Μονοχρωματική εικόνα αποχρώσεων του γκρι (Gray-scale image):** Η μονοχρωματική ψηφιακή εικόνα θα μπορούσε να χαρακτηριστεί και ως η εξέλιξη της δυαδικής εικόνας με τη μόνη διαφορά τους την ένταση χρώματος του κάθε εικονοστοιχείου. Το κάθε εικονοστοιχείο μπορεί να λάβει τιμές  $2^n$  (για  $n=8$ , 256 τιμές), από 0 (μαύρο) έως 255 (λευκό), που στη δυαδική αρίθμηση σημαίνει 8bit για το κάθε εικονοστοιχείο.

Λόγω του μεγάλου αριθμού αποχρώσεων της έντασης κάθε εικονοστοιχείου στις μονοχρωματικές εικόνες μπορούμε να δούμε με μεγάλη ευκρίνεια στις μορφές και τα σχήματα σε μια φυσική τους αναπαράσταση. Μία ψηφιακή εικόνα αποχρώσεων του γκρι διαστάσεων  $M \times N$  παριστάνεται στην Εικόνα 2.2.



**Εικόνα 2.2.** Αναπαράσταση μονοχρωματικής εικόνας αποχρώσεων του γκρι

- **Έγχρωμη εικόνα (Colour image):** Οι έγχρωμες εικόνες περιέχουν πληροφορίες για τα χρώματα σε κάθε pixel και αναπαρίστανται συνήθως χρησιμοποιώντας το μοντέλο RGB (Red, Green, Blue). Σε αυτό το μοντέλο, κάθε pixel περιέχει τρεις τιμές που αντιπροσωπεύουν την ένταση του κόκκινου (R), του πράσινου (G) και του μπλε (B) χρώματος. Με τον συνδυασμό αυτών των τριών χρωμάτων, δημιουργούνται όλες οι χρωματικές αποχρώσεις που βλέπουμε στην εικόνα 2.3. Η έγχρωμη εικόνα είναι κατάλληλη για φωτογραφίες και άλλες εικόνες όπου η ακριβής αναπαράσταση των χρωμάτων είναι σημαντική για την οπτική αναγνώριση και την επεξεργασία.

Μια έγχρωμη ψηφιακή εικόνα διαστάσεων  $M \times N$  αντιπροσωπεύεται από ένα τρισδιάστατο πίνακα  $M \times N \times 3$  (3 επίπεδα):  $M$ (αριθμός γραμμών)  $\times$   $N$ (αριθμός στηλών)  $\times$   $Z$ (τα 3 επίπεδα R-G-B). Το κάθε επίπεδο αντιπροσωπεύεται από ένα πίνακα δύο διαστάσεων όπου μέσα είναι καταχωρημένες οι εντάσεις του κάθε εικονοστοιχείου για κάθε επίπεδο. Η ένταση κάθε εικονοστοιχείου «**ERGB**» παράγεται από το γινόμενο των τριών εντάσεων:  $ER$  (επίπεδο R), επί  $EG$  (επίπεδο G), επί  $EB$  (επίπεδο B), και αντιπροσωπεύει συγκεκριμένο χρώμα από το ορατό φάσμα του φωτός σε κάθε εικονοστοιχείο. Δηλαδή **ένταση ERGB =  $ER \times EG \times EB$** . Η ένταση του χρώματος κάθε εικονοστοιχείου συνήθως αποθηκεύεται χρησιμοποιώντας 8 bits. Συνολικά, ένα εικονοστοιχείο στην έγχρωμη εικόνα απαιτεί 24 bits ( $3 \times 8 \text{bits}$ ). Έτσι, σε μία έγχρωμη ψηφιακή εικόνα χρωματικού βάθους 8 bits είναι δυνατό να διατυπωθούν 16,77 εκατομμύρια ( $2^8 \times 2^8 \times 2^8$ ) πιθανά χρώματα.



**Εικόνα 2.3.** RGB εικόνα και η διάσπασή της στα τρία επιμέρους χρώματα.

Κάποια άλλα χαρακτηριστικά των ψηφιακών συνοψίζονται στα ακόλουθα:

- **Ανάλυση (resolution):** Η ανάλυση αναφέρεται στον αριθμό των pixels που αποτελούν την εικόνα. Μεγαλύτερη ανάλυση σημαίνει περισσότερα pixels και υψηλότερη λεπτομέρεια. Παράλληλα υπάρχει και η σχετική ανάλυση, όπου υπολογίζεται ο αριθμός των εικονοστοιχείων σε μία μονάδα μήκους. Ευρέως διαδεδομένο είναι το ppi (pixel per inch = εικονοστοιχείο ανά ίντσα) ή και dpi (dots per inch = κουκίδες ανά ίντσα) (για εκτύπωση). Παραδείγματα τυπικών αναλύσεων είναι τα 1920x1080 (FHD) και 3840x2160 (4K UHD).
- **Χρωματικό βάθος (color depth):** Το χρωματικό βάθος αναφέρεται στον αριθμό των διαφορετικών χρωμάτων που μπορούν να αναπαρασταθούν σε κάθε pixel. Πιο υψηλό χρωματικό βάθος σημαίνει μεγαλύτερη ποικιλία χρωμάτων. Το συνηθέστερο είναι το 8-bit (256 χρώματα), αλλά υπάρχουν και 10-bit, 12-bit και 16-bit εικόνες με μεγαλύτερο χρωματικό βάθος.
- **Ευκρίνεια (clarity):** Η ευκρίνεια σε μια εικόνα αναφέρεται στο πόσο καλά μπορούν να διακριθούν και να αναγνωριστούν οι λεπτομέρειες και οι δομές που περιέχονται σε αυτήν. Καθορίζεται από παράγοντες όπως η ανάλυση της εικόνας (περισσότερη πληροφορία συνεπάγεται καλύτερη ευκρίνεια), ο φυσικός εξοπλισμός (προηγμένες κάμερες με μεγαλύτερους αισθητήρες μπορούν να καταγράψουν περισσότερες λεπτομέρειες), η φωτογραφική τεχνική (π.χ. έκθεση, εστίαση) και η ψηφιακή επεξεργασία.
- **Μέγεθος αρχείου (file size):** Το μέγεθος του αρχείου μίας ψηφιακής εικόνας μετράται σε KB, MB και GB όπως και όλα τα αρχεία σε ένα ηλεκτρονικό υπολογιστή. Το μέγεθος αρχείου σε μία ψηφιακή εικόνα είναι ανάλογο των διαστάσεών της σε εικονοστοιχεία αλλά και του χρωματικού της βάθους.



Συγκεκριμένα, το μέγεθος μίας ψηφιακής εικόνας υπολογίζεται:  $M$  (αριθμός γραμμών)  $\times N$  (αριθμός στηλών)  $\times m$  (χρωματικό βάθος bits). Εικόνες μεγάλων διαστάσεων και μεγάλου χρωματικού βάθους έχουν αρκετά μεγαλύτερο μέγεθος από εικόνες με μικρότερες διαστάσεις ή μικρότερο χρωματικό βάθος.

- **Φορμάτ εικόνας (image format):** Οι ψηφιακές εικόνες μπορούν να αποθηκευθούν σε διάφορα φορμάτ, όπως JPEG, PNG, GIF, BMP, TIFF κ.ά. Κάθε φορμάτ έχει τα δικά του πλεονεκτήματα και περιορισμούς όσον αφορά τη συμπίεση, τη διαφάνεια, και την ποιότητα των εικόνων.
- **Χρήση (intended use):** Η χρήση της εικόνας επηρεάζει τον τρόπο που θα αναπαρασταθεί και τις απαιτήσεις για ανάλυση, χρωματικό βάθος και άλλες παραμέτρους. Για παράδειγμα, η φωτογραφία απαιτεί υψηλή ανάλυση και χρωματικό βάθος, ενώ για μία εικόνα προφίλ σε μέσα κοινωνικής δικτύωσης μπορούν να χρησιμοποιηθούν συμπιεσμένα φορμάτ για τη γρήγορη φόρτωση της στην ιστοσελίδα.

Οι ψηφιακές εικόνες των ορυκτών ποικίλουν με τα παραπάνω χαρακτηριστικά. Αυτό δυσχεραίνει την σύγκριση των κλάσεων και εκπαίδευση του νευρωνικού δικτύου. Επιπρόσθετα σε πολλές περιπτώσεις οι εικόνες αποτελούνται από το ίδιο το ορυκτό και το περιβάλλον στο οποίο βρίσκεται. Για να αναδειχθεί το αντικείμενο από το περιβάλλον του χρησιμοποιούνται οι παρακάτω τεχνικές εντοπισμού περιοχών ενδιαφέροντος.

### 2.3. Τεχνικές εντοπισμού περιοχών ενδιαφέροντος

Οι τεχνικές εντοπισμού περιοχών ενδιαφέροντος (Region of Interest Detection - ROI) σε εικόνες είναι διαδικασίες που χρησιμοποιούνται για τον εντοπισμό και την επισήμανση περιοχών σε μια εικόνα που περιέχουν αντικείμενα ή χαρακτηριστικά που απασχολούν ιδιαίτερα το ενδιαφέρον του αναλυτή. Αυτές οι τεχνικές είναι συχνά απαραίτητες για εφαρμογές επεξεργασίας εικόνας, αναγνώρισης αντικειμένων, ανίχνευσης προσώπων, ανάλυσης ιατρικών εικόνων, ανίχνευσης ανεπιθύμητου περιεχομένου στο διαδίκτυο και πολλών άλλων.

Οι έρευνες αυτές περιλαμβάνουν τεχνικές κατάτμησης εικόνων (Image Segmentation), σε συνδυασμό με διάφορα στάδια προ-επεξεργασίας (Pre-Processing) των εικόνων. Το στάδιο της προ-επεξεργασίας, μέσω διαφόρων μετασχηματισμών, είναι αναγκαίο για τον περιορισμό του θορύβου ή των όποιων άλλων προβλημάτων υπάρχουν στην μορφολογία της εικόνας. Θεωρείται επιβεβλημένο για την προσαρμογή των δεδομένων εισόδου κατά την εκπαίδευση του εκάστοτε ταξινομητή, που θα ακολουθήσει αυτή τη διαδικασία .

### 2.3.1. Εξισορρόπηση Ιστογράμματος (Histogram Equalization)

Η εξισορρόπηση ιστογράμματος (histogram equalization) είναι μια ευρέως διαδεδομένη τεχνική επεξεργασίας εικόνας που χρησιμοποιείται για τη βελτίωση της αντίθεσης και της ευκρίνειας σε μια εικόνα. Ο σκοπός της εξισορρόπησης ιστογράμματος είναι να επιτύχει μια ομοιόμορφη κατανομή των επιπέδων φωτεινότητας στην εικόνα, ώστε οι λεπτομέρειες να είναι πιο εμφανείς.

Η διαδικασία περιλαμβάνει τα εξής βήματα:

1. **Υπολογισμός του ιστογράμματος της εικόνας:** Το ιστόγραμμα αντιπροσωπεύει τον αριθμό των εικονοστοιχείων για κάθε επίπεδο φωτεινότητας στην εικόνα.
2. **Υπολογισμός του κανονικοποιημένου ιστογράμματος:** Το κανονικοποιημένο ιστόγραμμα είναι μια προσαρμοσμένη έκδοση του ιστογράμματος που κανονικοποιεί τις συχνότητες εμφάνισης των επιπέδων φωτεινότητας, ώστε να κυμαίνονται από 0 έως 255 (ή από 0% έως 100%).
3. **Δημιουργία του νέου εξισορροπημένου πίνακα φωτεινότητας:** Από το κανονικοποιημένο ιστόγραμμα, δημιουργείται ένας νέος πίνακας φωτεινότητας, ο οποίος αντιστοιχεί τις τιμές των επιπέδων φωτεινότητας της αρχικής εικόνας σε νέες τιμές που περιέχονται σε ένα εύρος 0-255.
4. **Εφαρμογή του νέου πίνακα φωτεινότητας στην αρχική εικόνα:** Ο νέος πίνακας φωτεινότητας εφαρμόζεται στην αρχική εικόνα, μετασχηματίζοντας τη φωτεινότητα κάθε του κάθε εικονοστοιχείου (pixel) σύμφωνα με το νέο ιστόγραμμα.

Η λειτουργία της συγκεκριμένης τεχνικής βασίζεται στη μεγιστοποίηση της αντίθεσης της φωτεινότητας, σε συγκεκριμένες περιοχές της εικόνας. Επομένως, η μέθοδος εφαρμόζεται καλύτερα όταν στο ιστόγραμμα μιας εικόνας είναι συγκεντρωμένα εικονοστοιχεία με συγκεκριμένη φωτεινότητα. Στην ουσία, μέσω της εφαρμογής της μεθόδου εξισορρόπησης ιστογράμματος οι περιοχές της εικόνας με χαμηλή ένταση φωτεινότητας γίνονται σκοτεινότερες και οι περιοχές με υψηλότερη ένταση γίνονται φωτεινότερες.

### 2.3.2. Δημιουργία Κατωφλίου ή Δυαδοποίηση (Thresholding – Binarization)

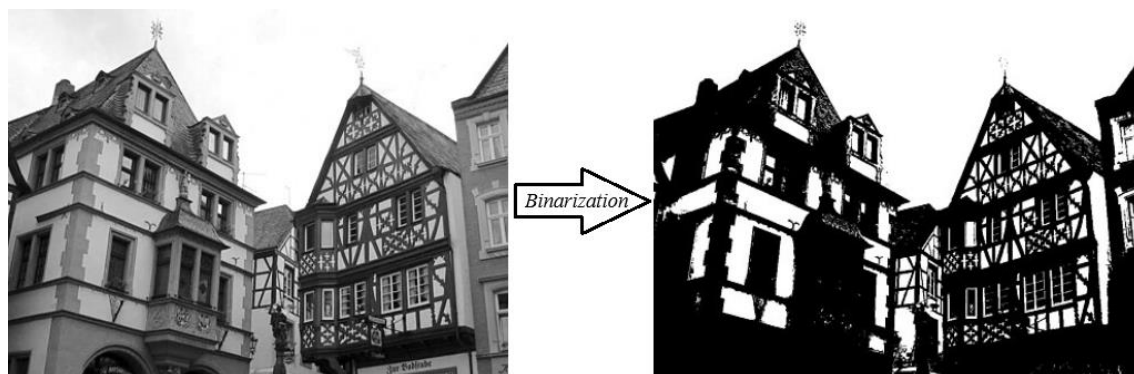
Η δημιουργία κατωφλίου είναι μια τεχνική επεξεργασίας εικόνας που χρησιμοποιείται για τον μετασχηματισμό μιας εικόνας σε δυαδική μορφή, όπου κάθε εικονοστοιχείο της εικόνας αντιστοιχεί είτε στην τιμή ένα (1) είτε σε μηδέν (0). Στόχος της κατωφλίωσης είναι να επισημάνει περιοχές της εικόνας που περιέχουν

αντικείμενα ή πληροφορίες ενδιαφέροντος από το υπόβαθρο, αντιμετωπίζοντας την ως ασπρόμαυρη.

Η κατωφλίωση λειτουργεί ως εξής:

1. **Καθορισμός του κατωφλίου:** Επιλέγεται ένα κατώφλι (threshold), που είναι ένας αριθμός μεταξύ 0 και 255 (σε κλίμακα 8-bit). Συνήθως, το ιστόγραμμα χωρίζεται σε δυο περιοχές (ζώνες) βάσει του κατωφλίου αυτού. Η πρώτη ζώνη καθορίζεται βάσει του φόντου της εικόνας, ενώ η δεύτερη βάσει του αντικειμένου. Οι τιμές των pixel που είναι μεγαλύτερες από το κατώφλι θα αντιστοιχούν στο λευκό χρώμα, ενώ οι τιμές μικρότερες από το κατώφλι θα αντιστοιχούν στο μαύρο χρώμα. Επεκτάσεις της μεθόδου μπορεί να περιλαμβάνουν πολλαπλά κατώφλια (adaptive thresholding) για διαφορετικά τμήματα της ψηφιακής εικόνας.
2. **Εφαρμογή του κατωφλίου στην εικόνα:** Κάθε εικονοστοιχείο της εικόνας συγκρίνεται με το κατώφλι. Αν η τιμή του είναι μεγαλύτερη από το κατώφλι, τότε αντιστοιχεί στο λευκό, ενώ αν είναι μικρότερη, αντιστοιχεί στο μαύρο.

Η επιλογή του κατωφλίου εξαρτάται από την εφαρμογή και από τα χαρακτηριστικά της εικόνας. Σκοπός είναι να επιλεγεί ένα κατώφλι που θα διαχωρίσει τις περιοχές της εικόνας που περιέχουν αντικείμενα ενδιαφέροντος από τις υπόλοιπες. Μπορεί σε ορισμένες περιπτώσεις να επιλεγεί χειροκίνητα με βάση την κατανομή των φωτεινотήτων, αλλά συνήθως επιλέγεται αυτόματα με τη χρήση αλγορίθμων. Παράδειγμα τέτοιου αλγορίθμου αποτελεί η μέθοδος Otsu, που έχει ως σκοπό τον υπολογισμό του κατωφλίου που ελαχιστοποιεί το άθροισμα εσωτερικής διακύμανσης, δηλαδή το κατώφλι που κατανέμει τα δεδομένα στις δύο κλάσεις (π.χ. κλάση μαύρου για το υπόβαθρο και κλάση άσπρου για την περιοχή ενδιαφέροντος) έτσι ώστε η διακύμανση σε κάθε κλάση να είναι η ελάχιστη δυνατή.



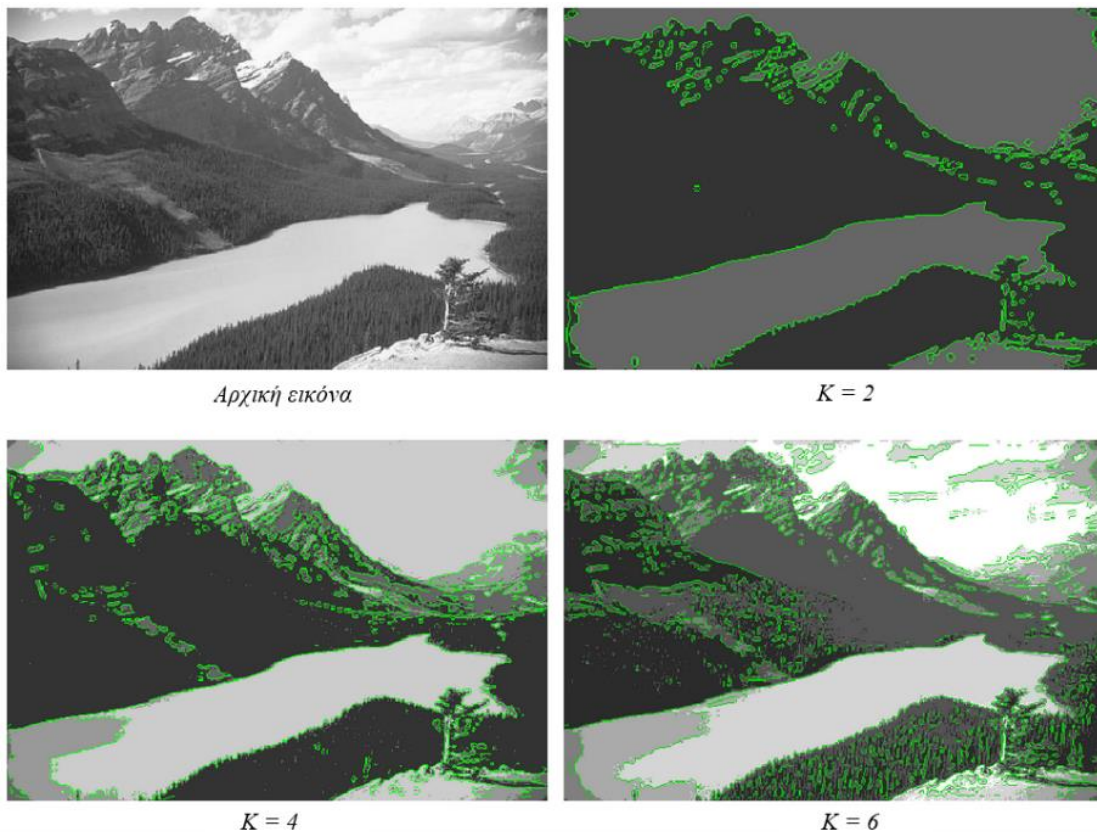
**Εικόνα 2.4.** Ψηφιακή εικόνα πριν και μετά την εφαρμογή της μεθόδου Otsu

### 2.3.3. Δημιουργία Συστάδων (Clustering)

Αποτελεί μία εναλλακτική μέθοδο κατάτμησης της εικόνας. Η λειτουργία της βασίζεται στη δημιουργία ομάδων για περιοχές από εικονοστοιχεία με κοινά χαρακτηριστικά, όπως το χρώμα, η φωτεινότητα και η υφή, προκειμένου να απομονωθούν οι περιοχές που ενδιαφέρουν τον αναλυτή.

Υπάρχουν διάφοροι αλγόριθμοι συσταδοποίησης που μπορούν να χρησιμοποιηθούν στο πλαίσιο αυτό. Δύο από τους πιο γνωστούς είναι οι ακόλουθοι:

- **Συσταδοποίηση K-Means:** Ο αλγόριθμος K-Means ανήκει στην κατηγορία της διαχωριστικής συσταδοποίησης (partitional clustering) και στοχεύει στον χωρισμό μιας εικόνας σε K ομάδες (clusters). Ο αριθμός K προκαθορίζεται από τον χρήστη. Σκοπός είναι η δημιουργία ομάδων από εικονοστοιχεία με όσο το δυνατόν πιο όμοια χαρακτηριστικά, και ταυτόχρονα όσο το δυνατόν πιο διαφορετικά χαρακτηριστικά από τα εικονοστοιχεία των υπολοίπων ομάδων.
- **Ιεραρχική Συσταδοποίηση (Hierarchical clustering):** Ο αλγόριθμος της ιεραρχικής συσταδοποίησης ξεκινά με τον χωρισμό της εικόνας σε μικρές συστάδες (όσες και τα εικονοστοιχεία) και στη συνέχεια συνδέει αυτές τις μικρές συστάδες μεταξύ τους με ιεραρχικό τρόπο δημιουργώντας σε κάθε βήμα μεγαλύτερες, λαμβάνοντας υπόψη κριτήρια όπως η ελάχιστη απόσταση (ομοιότητα), τον μέσο όρο της συστάδας ή την απόσταση μεταξύ κεντρικών σημείων (εικονοστοιχείων) της κάθε ομάδας. Η παραλλαγή αυτή ονομάζεται συσσωρευτική (agglomerative) ιεραρχική συσταδοποίηση ενώ η αντίστροφη διαδικασία, η οποία ξεκινάει με μία συστάδα που περιέχει όλα τα εικονοστοιχεία και με διαχωρισμούς καταλήγει σε αριθμό συστάδων όσα και τα εικονοστοιχεία, ονομάζεται διαιρετική (divisive) ιεραρχική συσταδοποίηση. Η ιεραρχική δομή που προκύπτει (δενδρόγραμμα) είναι χρήσιμη για τον εντοπισμό περιοχών ενδιαφέροντος σε διάφορα επίπεδα λεπτομέρειας. Ανάλογα με το επίπεδο που ο χρήστης θα επιλέξει, μπορούν να εντοπιστούν περιοχές ενδιαφέροντος που είναι από πολύ γενικές έως πολύ λεπτομερείς.



**Εικόνα 2.4.** Εφαρμογή συσταδοποίησης *K-Means* για διάφορες τιμές *K*

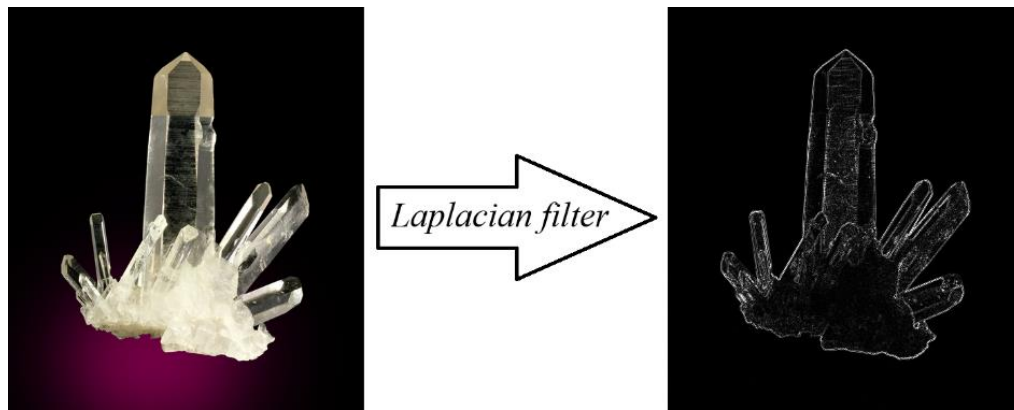
#### 2.3.4. Λαπλασιανό Φίλτρο (Laplacian Filter)

Το Λαπλασιανό φίλτρο αποτελεί ταυτόχρονα μέθοδο και φίλτρο. Είναι ένας μαθηματικός τελεστής που χρησιμοποιείται για την **ανίχνευση ακμών (edge detection)** σε ψηφιακές εικόνες. Υπολογίζει τη δεύτερη παράγωγο της εικόνας για να επισημάνει περιοχές σημαντικής και ταχείας αλλαγής έντασης φωτεινότητας, οι οποίες μπορεί να είναι ενδεικτικές ύπαρξης ακμών. Στην πράξη χρησιμοποιείται ως **φίλτρο συνέλιξης (convolution filter)**.( *Zhang J, Gao Q, Luo H, Long T. Mineral Identification Based on Deep Learning Using Image Luminance Equalization. Applied Sciences. 2022; 12(14):7055*)

Η Λαπλασιανή  $L(x, y)$  μιας εικόνας με τιμές έντασης των pixel  $I(x, y)$  δίνεται από:

$$L(x, y) = \partial^2 I / \partial x^2 + \partial^2 I / \partial y^2$$

Τα ορυκτά συχνά έχουν διακριτά όρια ή άκρες που μπορεί να είναι σημαντικά για την ταξινόμησή τους. Με την εφαρμογή του φίλτρου, αυτά τα όρια ενισχύονται, καθιστώντας ευκολότερο τον εντοπισμό και την διάκριση μεταξύ διαφορετικών ορυκτών.



**Εικόνα 2.5.** Εικόνα ορυκτού χαλαζία πριν και μετά την εφαρμογή Λαπλασιανού φίλτρου

## 2.4. Μέθοδοι ταξινόμησης

Οι ταξινομητές εικόνων είναι μοντέλα μηχανικής μάθησης που μπορούν να αντιστοιχίσουν ετικέτες ή κατηγορίες σε εικόνες με βάση το περιεχόμενό τους. Υπάρχουν διάφοροι τύποι ταξινομητών εικόνων που μπορούν να κατηγοριοποιηθούν με διαφορετικούς τρόπους με βάση τις υποκείμενες τεχνικές και αρχιτεκτονικές τους. Ακολουθούν ορισμένες κοινές κατηγορίες ταξινομητών εικόνων και μερικές τεχνικές για κάθε κατηγορία:

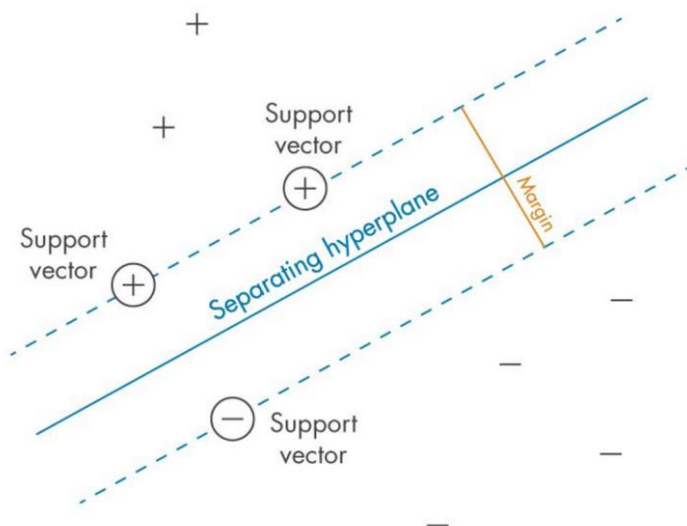
### 2.4.1. Παραδοσιακοί ταξινομητές μηχανικής μάθησης

#### 2.4.1.1. Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines)

Οι SVMs εδραιώθηκαν από το 2000 και έπειτα ως μία από τις πιο διαδεδομένες μεθόδους μηχανικής μάθησης για την επίλυση προβλημάτων ταξινόμησης και παλινδρόμησης. Αν και αρχικά αρχικά αναπτύχθηκαν για γραμμικά προβλήματα (V. Vapnik 1963), απέκτησαν δημοσιότητα όταν ενισχύθηκαν με το τέχνασμα του πυρήνα (kernel trick) που επέτρεψε τη χρήση τους και σε μη γραμμικώς διαχωρίσιμα προβλήματα.

Θεωρώντας ένα πρόβλημα δυαδικής ταξινόμησης με θετικά και αρνητικά παραδείγματα, μία επιφάνεια που λειτουργεί ως σύνορο μεταξύ των κλάσεων χαρακτηρίζεται από το γεγονός ότι χωρίζει τα θετικά από τα αρνητικά παραδείγματα. Η μέθοδος επιδιώκει να βρει το σύνορο που απέχει όσο το δυνατόν περισσότερο από τα παραδείγματα των κλάσεων που διαχωρίζει. Η υπερεπιφάνεια αυτή ονομάζεται επιφάνεια μεγίστου περιθωρίου (maximum margin hypersurface) και σε γραμμικώς διαχωρίσιμα προβλήματα ορίζεται από ένα πεπερασμένο αριθμό παραδειγμάτων του συνόλου εκπαίδευσης που ονομάζονται διανύσματα υποστήριξης (support vectors).





**Εικόνα 2.6.** Κατηγοριοποίηση αρνητικών και θετικών δειγμάτων με χρήση SVMs

Μέσω των συναρτήσεων πυρήνα (kernel functions), οι SVM μπορούν να μετασχηματίσουν τον αρχικό χώρο υποθέσεων έτσι ώστε μη γραμμικώς διαχωρίσιμα προβλήματα να μετατραπούν σε γραμμικώς διαχωρίσιμα και τελικά να λυθούν με την ίδια μεθοδολογία.

Επομένως το κύριο πλεονέκτημα των SVM είναι ότι όχι μόνο δημιουργούν ένα διαχωριστικό σύνορο μεταξύ των επιμέρους κλάσεων (κάτι που κάνουν άλλωστε και οι περισσότερες μέθοδοι ταξινόμησης), αλλά φροντίζουν αυτό το σύνορο να απέχει όσο το δυνατόν περισσότερο από τα παραδείγματα των κλάσεων που διαχωρίζει.

Η επιτυχία των SVMs στην κατηγοριοποίηση εικόνων εξαρτάται σε μεγάλο βαθμό από την ποιότητα των δεδομένων εκπαίδευσης. Πλέον χρησιμοποιούνται κυρίως όταν το σύνολο δεδομένων είναι σχετικά μικρό ή όταν οι υπολογιστικοί πόροι είναι περιορισμένοι, καθώς υπάρχουν πιο αποτελεσματικές τεχνικές βαθιάς μάθησης που θα δούμε παρακάτω. Παρόλα αυτά, οι SVMs παραμένουν μια από τις πιο αποτελεσματικές μεθόδους για την κατηγοριοποίηση εικόνων, ειδικά όταν έχουν εφαρμοστεί και ρυθμιστεί σωστά.

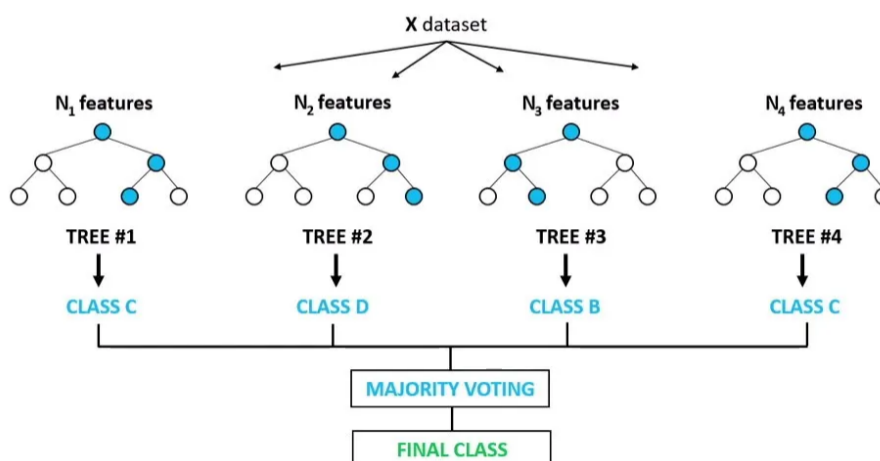
#### 2.4.1.2. Ταξινομητές Τυχαίων Δασών (Random Forest Classifiers)

Οι ταξινομητές τυχαίων δασών, γνωστοί και ως Random Forest Classifiers, αποτελούν ένα δημοφιλές μοντέλο μηχανικής μάθησης που χρησιμοποιείται ευρέως στην κατηγοριοποίηση εικόνων, καθώς και σε πολλές άλλες εφαρμογές. Τα τυχαία δάση είναι ένας τύπος ενσωματωμένης μάθησης (ensemble learning) μοντέλου, που σημαίνει ότι συνδυάζουν τα αποτελέσματα πολλών απλών μοντέλων μηχανικής μάθησης για να βελτιώσουν την απόδοση της κατηγοριοποίησης.

Η λειτουργία των ταξινομητών τυχαίων δασών μπορεί να περιγραφεί σύντομα ως εξής:

1. **Επιλογή Δειγμάτων:** Αρχικά, το Random Forest επιλέγει τυχαία ένα υποσύνολο των δειγμάτων (εικόνων) και χαρακτηριστικών από τον συνολικό χώρο δεδομένων.
2. **Κατασκευή Δέντρων Αποφάσεων (Decision Trees):** Για κάθε υποσύνολο δειγμάτων, κατασκευάζονται δέντρα αποφάσεων. Κάθε δέντρο αποφάσεων διαχωρίζει τα δείγματα με βάση τα χαρακτηριστικά τους με στόχο την κατηγοριοποίηση τους.
3. **Ψηφοφορία:** Κατά την κατηγοριοποίηση μιας νέας εικόνας, το τυχαίο δάσος επιτρέπει σε κάθε δέντρο αποφάσεων να ψηφίσει για την κατηγορία της εικόνας. Η τελική κατηγορία επιλέγεται μέσω απλής πλειοψηφίας, δηλαδή επιλέγεται η κατηγορία που έλαβε τις περισσότερες ψήφους από τα δέντρα αποφάσεων.

## Random Forest Classifier



**Εικόνα 2.7.** Αναπαράσταση τυχαίου δάσους για κατηγοριοποίηση μεταξύ 4 κλάσεων

Οι ταξινομητές τυχαίων δασών είναι στην ουσία ένα σύνολο από δέντρα αποφάσεων που συνεργάζονται για να κατηγοριοποιήσουν νέα δεδομένα. Αυτή η προσέγγιση βοηθά στην αποφυγή της υπερπροσαρμογής στα δεδομένα εκπαίδευσης (overfitting) που μπορεί να παρουσιαστεί μεμονωμένα σε ένα απλό δέντρο αποφάσεων. Επιπλέον, η τυχαιότητα που εισάγεται κατά την επιλογή δειγμάτων και χαρακτηριστικών βοηθά στη βελτίωση της γενικευμένης χρήσης (generalization) των




μοντέλων. Αν και ενδέχεται να μην αποδίδουν το ίδιο καλά με νεότερες τεχνικές, είναι ένα ακόμη πολύτιμο εργαλείο για την ταξινόμηση εικόνων.

#### 2.4.1.3. Ταξινομητής K-Κοντινότερων Γειτόνων (K-Nearest Neighbors Classifier)

Ο Ταξινομητής K-Κοντινότερων Γειτόνων (K-NN) είναι ένας αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται κυρίως για εφαρμογές ταξινόμησης δειγμάτων. Όταν εφαρμόζεται στην ταξινόμηση εικόνας, κάθε εικόνα αντιμετωπίζεται ως σημείο δεδομένων σε ένα χώρο χαρακτηριστικών υψηλών διαστάσεων, με κάθε pixel να αντιπροσωπεύει ένα χαρακτηριστικό. Το "K" υποδηλώνει τον αριθμό των πλησιέστερων γειτόνων που πρέπει να ληφθούν υπόψη κατά τη λήψη μιας απόφασης ταξινόμησης. Ο αλγόριθμος προσδιορίζει τα δείγματα εκπαίδευσης K που είναι πιο κοντά στη δεδομένη εικόνα στον χώρο των χαρακτηριστικών χρησιμοποιώντας μια μέτρηση απόστασης, όπως η Ευκλείδεια απόσταση ή η απόσταση Manhattan, για τη μέτρηση της ομοιότητας ή της ανομοιότητας μεταξύ των δεδομένων.

Ο αλγόριθμος χρησιμοποιεί στη συνέχεια έναν μηχανισμό επικράτησης της πλειοψηφίας. Υπολογίζει την απόσταση μεταξύ της εικόνας εισόδου και όλων των δειγμάτων εκπαίδευσης, επιλέγει τα K δείγματα εκπαίδευσης με τις μικρότερες αποστάσεις και εκχωρεί την κλάση που εμφανίζεται πιο συχνά μεταξύ αυτών των K πλησιέστερων γειτόνων ως την προβλεπόμενη κλάση για την εικόνα εισόδου.

n		y	$d(x_n, x_i)$
1		●	1.0
2		●	1.5
3		●	1.7
4		●	2.0
5		●	2.5
6		●	3.0

$$\hat{y} = \text{●}$$



**Εικόνα 2.8.** Κατηγοριοποίηση εικόνας μεταξύ σκύλου και γάτας με χρήση K-NN (για  $K=3$ )

Είναι σημαντικό ωστόσο να γνωρίζουμε ότι η επιλογή της τιμής για το K είναι κρίσιμη, καθώς μπορεί να επηρεάσει την ευαισθησία του ταξινομητή στον θόρυβο και την εξομάλυνσή του. Επιπλέον, η επιλογή της κατάλληλης μεθόδου μέτρησης απόστασης θα πρέπει να γίνεται με βάση τα χαρακτηριστικά των δεδομένων. Το υπολογιστικό

κόστος του K-NN μπορεί να γίνει αρκετά υψηλό, ειδικά με μεγάλα σύνολα δεδομένων, καθώς απαιτεί τον υπολογισμό των αποστάσεων από όλα τα δείγματα εκπαίδευσης για κάθε πρόβλεψη.

#### 2.4.2. Ταξινομητές Στατιστικής Ανάλυσης (Statistical Classifiers)

##### 2.4.2.1. Μπαεσιανός Ταξινομητής (Bayes Classifier)

Ο ταξινομητής Bayes είναι αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται για διάφορες εργασίες ταξινόμησης, συμπεριλαμβανομένης της ταξινόμησης εικόνων. Έχει τις ρίζες του στη Μπαεσιανή θεωρία πιθανοτήτων, μια θεμελιώδη έννοια στη στατιστική και τις πιθανότητες. Ο βασικός σκοπός του είναι να αντιστοιχίσει πιθανότητες σε διαφορετικές κλάσεις ή κατηγορίες και να κάνει προβλέψεις με βάση αυτές τις πιθανότητες.

Η λειτουργία του ταξινομητή περιλαμβάνει τον υπολογισμό της υπό όρους πιθανότητας για κάθε πιθανή κλάση. Αυτός ο υπολογισμός λαμβάνει υπόψη τόσο τα παρατηρούμενα χαρακτηριστικά μιας εικόνας όσο και αυτά που είναι γνωστά ως "προηγούμενες πιθανότητες" (prior probabilities). Οι προηγούμενες πιθανότητες είναι οι αρχικές πεποιθήσεις ή πιθανότητες που αποδίδονται σε κάθε κλάση πριν ληφθεί υπόψη οποιαδήποτε νέα πληροφορία χαρακτηριστικών. Χρησιμοποιούν ως αφετηρία για τη διαδικασία ταξινόμησης και βασίζονται τυπικά στην κατανομή των κλάσεων στα δεδομένα της εκπαίδευσης.

Ο ταξινομητής συνδυάζει αυτές τις προηγούμενες πιθανότητες με τις πιθανότητες που προκύπτουν από τα παρατηρούμενα χαρακτηριστικά της εικόνας για να καθορίσει την τελική ανάθεση κλάσης. Με άλλα λόγια, συνδυάζει τις αρχικές εκτιμήσεις με τα στοιχεία από τα χαρακτηριστικά της εικόνας. Η κλάση με την υψηλότερη δεσμευμένη πιθανότητα, γνωστή ως "μεταγενέστερη πιθανότητα" (posterior probability), επιλέγεται τελικά ως η προβλεπόμενη κλάση.

##### 2.4.2.2. «Αφελής» Μπαεσιανός Ταξινομητής (Naive Bayes Classifier)

Ο ταξινομητής Naive Bayes είναι ένας πιθανολογικός αλγόριθμος μηχανικής μάθησης που χρησιμοποιείται για διεργασίες ταξινόμησης, ιδιαίτερα στην ταξινόμηση κειμένου, στον εντοπισμό ανεπιθύμητων μηνυμάτων (spam), στην ανάλυση συναισθήματος (sentiment analysis) και σε ορισμένα σενάρια ταξινόμησης εικόνων. Ονομάζεται "αφελής" λόγω της απλουστευτικής του υπόθεσης ότι όλα τα χαρακτηριστικά που χρησιμοποιούνται στη διαδικασία ταξινόμησης είναι ανεξάρτητα, δεδομένης της ετικέτας - κλάσης. Με άλλα λόγια, υποθέτει ότι η παρουσία ή η απουσία ενός χαρακτηριστικού δεν επηρεάζει την παρουσία ή την απουσία οποιουδήποτε άλλου χαρακτηριστικού. Αυτή είναι συχνά μία μη ρεαλιστική υπόθεση, ειδικά σε τομείς όπως η ταξινόμηση εικόνων, όπου οι τιμές και τα

χαρακτηριστικά των εικονοστοιχείων μπορεί να έχουν πολύπλοκες σχέσεις και αλληλεξαρτήσεις.

Παρά αυτή την υπεραπλούστευση, ο ταξινομητής Naive Bayes χρησιμοποιείται ευρέως και μπορεί να αποδώσει εξαιρετικά στην πράξη. Ο λόγος της διαδεδομένης χρήσης του έγκειται στην υπολογιστική του αποτελεσματικότητα και στην ευκολία εφαρμογής του. Απαιτεί σχετικά μικρό όγκο δεδομένων εκπαίδευσης για την εκτίμηση των παραμέτρων και είναι σε θέση να κάνει προβλέψεις πολύ γρήγορα.

Στην ταξινόμηση κειμένου, για παράδειγμα, ο Naive Bayes μπορεί να χρησιμοποιηθεί για την ταξινόμηση εγγράφων σε κατηγορίες. Υπολογίζει την πιθανότητα παρατήρησης συγκεκριμένων λέξεων ή φράσεων σε ένα έγγραφο για κάθε κατηγορία και στη συνέχεια υπολογίζει την πιθανότητα το έγγραφο να ανήκει σε κάθε μία από αυτές. Η κατηγορία με την υψηλότερη πιθανότητα θα είναι και η προβλεπόμενη ετικέτα.

#### 2.4.3. Τεχνητά Νευρωνικά Δίκτυα (ΤΝΔ) (Artificial Neural Networks)

Τα τεχνητά νευρωνικά δίκτυα αποτελούν σήμερα την κορωνίδα των τεχνικών αναγνώρισης προτύπων. Εντάσσονται στις πιο αποτελεσματικές μεθόδους και χρησιμοποιούνται ευρύτατα από ερευνητές στο χώρο της ταξινόμησης εικόνων.

Τα ΤΝΔ παρουσιάζουν ξεχωριστά πλεονεκτήματα σε σχέση με τις κλασικές μεθόδους ταξινόμησης προτύπων. Είναι αυτοπροσαρμοστικές διατάξεις οδηγούμενες από τα δεδομένα, δηλαδή μπορούν να προσαρμοστούν τα ίδια στα δεδομένα με τα οποία εκπαιδεύονται, χωρίς την απαίτηση λειτουργικής διευκρίνισης για τη μορφή των υποκείμενων μοντέλων. Ακόμα, τα ΤΝΔ είναι μη-γραμμικά μοντέλα, πράγμα που τα καθιστά ευέλικτα στη μοντελοποίηση σύνθετων προβλημάτων του πραγματικού κόσμου. Ακόμα και αν υπάρχουν προβλήματα στο σύνολο των δεδομένων (π.χ. θόρυβος ή ακραίες τιμές) το σύστημα καταφέρνει να τα ταξινομήσει με σχετικά καλή ακρίβεια, δεδομένου ότι κάθε διαδικασία ταξινόμησης αναζητά στην ουσία μια σχέση μεταξύ των χαρακτηριστικών ενός αντικειμένου και των ιδιοτήτων του.

Τα τεχνητά νευρωνικά δίκτυα στην αναγνώριση εικόνας μπορούν να κατηγοριοποιηθούν σε τρεις βασικές κατηγορίες:

- **Συνελκτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs):** Τα CNN αποτελούν είδος τεχνητών νευρωνικών δικτύων που ειδικεύονται στην αναγνώριση εικόνων και την επεξεργασία γραφικού υλικού. Χρησιμοποιούν φίλτρα συνέλιξης (convolution filters) για να εξάγουν χαρακτηριστικά από διάφορες περιοχές της εικόνας και στη συνέχεια χρησιμοποιούν αυτά τα χαρακτηριστικά για να αναγνωρίσουν πρότυπα και αντικείμενα. Έχουν επιφέρει επαναστατικές αλλαγές στην αναγνώριση εικόνας και χρησιμοποιούνται ευρέως

σε εφαρμογές όπως η αναγνώριση προσώπων, η αυτόματη ταξινόμηση εικόνων και η ανίχνευση αντικειμένων.

- **Δίκτυα Μεταφερόμενης Μάθησης (Transfer Learning Networks):** Τα δίκτυα αυτά είναι προεκπαιδευμένα μοντέλα τα οποία έχουν εκπαιδευτεί σε πολύ μεγάλα σύνολα δεδομένων που περιλαμβάνουν εκατομμύρια εικόνες. Χρησιμοποιούνται για εργασίες γενικότερης αναγνώρισης εικόνων και μπορούν επίσης να ρυθμιστούν (tuning) με ακρίβεια και για συγκεκριμένες εργασίες ταξινόμησης. Δημοφιλείς αρχιτεκτονικές αποτελούν τα δίκτυα VGG, ResNet και Inception.
- **Προσαρμοσμένες Αρχιτεκτονικές (Custom Architectures):** Ορισμένες εργασίες ταξινόμησης εικόνων ενδέχεται να απαιτούν αρχιτεκτονικές προσαρμοσμένες σε συγκεκριμένα δεδομένα ή απαιτήσεις. Αυτές οι αρχιτεκτονικές μπορούν να περιλαμβάνουν διάφορες αρχές σχεδιασμού, αλλά είναι συνήθως κατασκευασμένες για συγκεκριμένο σκοπό και δεν αποτελούν μέρος προκαθορισμένων κατηγοριών. Συχνά απαιτούν αρκετό πειραματισμό για τη ρύθμιση των παραμέτρων, με συνεχή αποθήκευση των αποτελεσμάτων σε κάθε επανάληψη με σκοπό την συνεχή βελτίωση του μοντέλου.

### 3. Τεχνητή Νοημοσύνη

#### 3.1. Εισαγωγή

Κατά τη δεκαετία του 1950, αναδείχθηκε ένας καινούριος τομέας στην επιστήμη των υπολογιστών, ο οποίος καθιερώθηκε διεθνώς ως Τεχνητή Νοημοσύνη (TN) – Artificial Intelligence (AI). Η TN προέκυψε από την φιλοδοξία να δοθεί στους υπολογιστές η ικανότητα να σκέφτονται και να λύνουν προβλήματα όπως οι άνθρωποι. Αυτό συνεπάγεται την ανάπτυξη υπολογιστικών συστημάτων που είναι ικανά να αναγνωρίζουν πρότυπα, να προσομοιάζουν παρουσία λογικής σκέψης, και να προσαρμόζουν τη συμπεριφορά τους ανάλογα με τις συνθήκες.

Η TN διαιρείται στην συμβολική και την υπο-συμβολική νοημοσύνη. Η πρώτη συνίσταται στην προσπάθεια εξομίωσης της ανθρώπινης συμπεριφοράς με χρήση ειδικών αλγορίθμων, μέσα από ένα σύνολο συμβόλων και λογικών κανόνων υψηλού επιπέδου, ενώ η δεύτερη στόχο έχει την προσέγγιση ή ακόμα και την αναπαραγωγή της ανθρώπινης ευφυΐας μέσα από στοιχειώδη αριθμητικά μοντέλα τα οποία επαγωγικά συνθέτουν νοήμονες συμπεριφορές, προσομοιώνοντας πραγματικές βιολογικές διαδικασίες, όπως η εξέλιξη των ειδών και η λειτουργία του ανθρώπινου εγκεφάλου.

Η επιστήμη της TN συνεργάζεται σήμερα με πολλούς επιστημονικούς κλάδους όπως αυτόν της πληροφορικής, της ψυχολογίας, της φιλοσοφίας, της νευρολογίας, της γλωσσολογίας καθώς και της επιστήμης των μηχανικών. Μερικά μόνο παραδείγματα σύγχρονων επιτευγμάτων της αποτελούν η αναγνώριση εικόνων, η δημιουργία μουσικής και βίντεο, η αυτόνομη οδήγηση, η ακριβής μετάφραση κειμένων, η παροχή συμβουλευτικών υπηρεσιών σε εταιρείες, η βελτιστοποίηση της κυβερνοασφάλειας και πολλά άλλα.

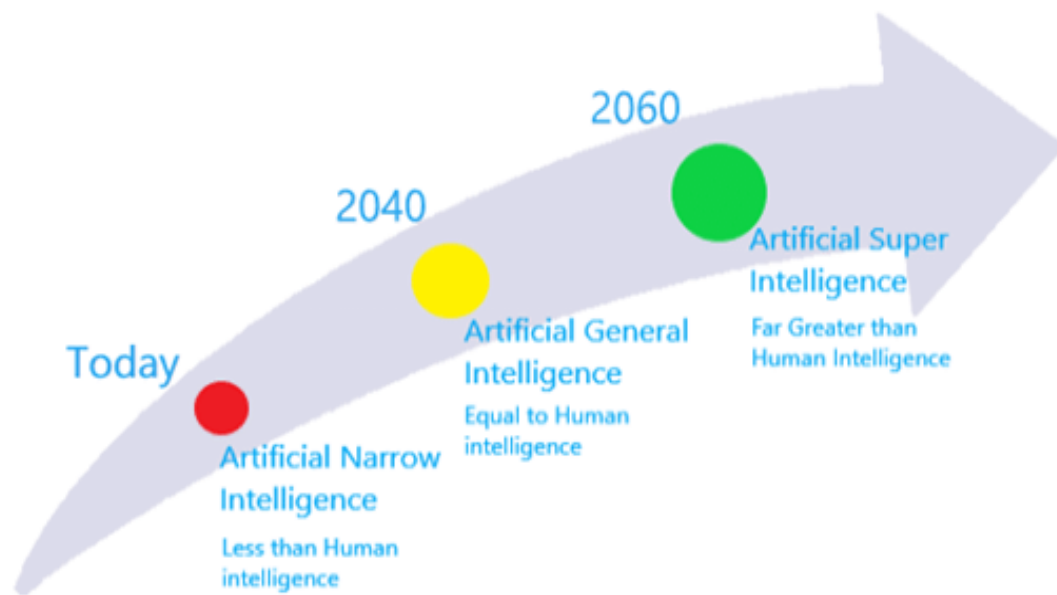
#### 3.2. Η εξελικτική πορεία της τεχνητής νοημοσύνης

Η εξέλιξη της τεχνητής νοημοσύνης μπορεί να κατηγοριοποιηθεί σε τρία στάδια:

1. **Αδύναμη Τεχνητή Νοημοσύνη (Artificial Narrow Intelligence)**, που αντιπροσωπεύει το αρχικό στάδιο στην εξέλιξη της. Σε αυτό το στάδιο, τα συστήματα έχουν σχεδιαστεί για συγκεκριμένες, καλά καθορισμένες εργασίες. Διαπρέπουν στην εκτέλεση αυτών των εργασιών, αλλά λειτουργούν σε περιορισμένο εύρος και δεν έχουν την ικανότητα να γενικεύουν τις γνώσεις τους σε μη σχετικούς τομείς. Τα αδύναμα συστήματα τεχνητής νοημοσύνης βασίζονται σε κανόνες ή σε αλγόριθμους και δεν διαθέτουν ανθρώπινη κατανόηση ή συνείδηση. Παραδείγματα αδύναμης τεχνητής νοημοσύνης αποτελούν οι εικονικοί βοηθοί (virtual assistants) όπως οι Siri και η Alexa, οι αλγόριθμοι συστάσεων σε πλατφόρμες ηλεκτρονικού εμπορίου και τα λογισμικά

αναγνώρισης εικόνων. Αυτά τα συστήματα είναι πολύτιμα στις καθορισμένες εφαρμογές τους, αλλά υστερούν στην προσαρμοστικότητα εν συγκρίσει με τους ανθρώπους.

2. **Η Τεχνητή Γενική Νοημοσύνη (Artificial General Intelligence - AGI)**, που αντιπροσωπεύει ένα μεγάλο άλμα στην ανάπτυξη της τεχνητής νοημοσύνης. Η AGI στοχεύει στο να δημιουργήσει μηχανές με ανθρώπινη νοημοσύνη, δίνοντάς τους τη δυνατότητα να μαθαίνουν, να κατανοούν και να εκτελούν ένα ευρύ φάσμα εργασιών σε διάφορους τομείς. Σε αντίθεση με την αδύναμη τεχνητή νοημοσύνη, τα συστήματα AGI έχουν την ικανότητα να γενικεύουν τη γνώση, να επιδεικνύουν λογική, να προσαρμόζονται σε νέα πλαίσια και να διαθέτουν αυτογνωσία. Η επίτευξη του AGI είναι ένας φιλόδοξος στόχος που ακόμη δεν έχει επιτευχθεί, καθώς απαιτεί την επίλυση σύνθετων προκλήσεων που σχετίζονται με τη γνώση, την κατανόηση και τη λήψη αποφάσεων που υπερβαίνουν τις καλώς καθορισμένες εργασίες. Οι ερευνητές εργάζονται ενεργά προς αυτό τον στόχο και η ενδεχόμενη υλοποίησή της μελλοντικά θα έχει ανατρεπτικές επιπτώσεις στην κοινωνία, τη βιομηχανία και την τεχνολογία.
3. **Ισχυρή Τεχνητή Νοημοσύνη (Strong Artificial Intelligence - ASI)**, γνωστή και ως «Τεχνητή Υπερευφυΐα (Artificial Superintelligence)», που αντιπροσωπεύει το υποθετικό στάδιο όπου η νοημοσύνη μηχανής ξεπερνά την ανθρώπινη νοημοσύνη σε όλες τις διαστάσεις. Η ASI υπονοεί ότι αυτές οι μηχανές θα είναι ικανές να ξεπερνούν τους ανθρώπους σε όλες σχεδόν τις γνωστικές εργασίες. Η επίτευξη της είναι ένα πρωτίστως θέμα κερδοσκοπίας και θα έχει σημαντικές ηθικές, κοινωνικές και υπαρξιακές επιπτώσεις. Η προοπτική της ASI εγείρει ανησυχίες σχετικά με τον έλεγχο, την ασφάλεια και τη δυνατότητα για γρήγορες, αυτόνομες εξελίξεις στα συστήματα AI.



**Εικόνα 3.1.** Τα τρία εξελικτικά στάδια της Τεχνητής Νοημοσύνης

Κάθε στάδιο παρουσιάζει μοναδικές προκλήσεις αλλά και ευκαιρίες. Η AGI αποτελεί ένα σημαντικό ορόσημο στην πορεία προς την πιθανή επίτευξη ισχυρής τεχνητής νοημοσύνης στο μέλλον. Η υλοποίηση μιας ισχυρής τεχνητής νοημοσύνης θα φέρει μια νέα εποχή στην τεχνολογία και την ηθική, και η προσεκτική προ-εξέταση των συνεπειών της είναι ζωτικής σημασίας.

### 3.3. Μηχανική Μάθηση (Machine Learning)

Η μηχανική μάθηση, ένας από τους σημαντικότερους κλάδους της ΤΝ, αφορά την ανάπτυξη κατάλληλων αλγορίθμων που θα δώσουν τη δυνατότητα της “εκμάθησης” στους υπολογιστές. Το λογισμικό που χρησιμοποιείται από τους υπολογιστές γίνεται πλέον ευέλικτο και προσαρμόσιμο με βάση την ανάλυση των δεδομένων που λαμβάνει από το χρήστη. Η ουσία της μηχανικής μάθησης συνοψίζεται στη χρήση αλγορίθμων ικανών να αναγνωρίζουν μοτίβα σε δεδομένα προκειμένου να λάβουν αποφάσεις, βασιζόμενες στην στατιστική, τη θεωρία των πιθανοτήτων και την βελτιστοποίηση. Χάρη την εκμάθηση μηχανών, απολαμβάνουμε υπηρεσίες όπως φίλτρα ανεπιθύμητης αλληλογραφίας (spam), αναγνώριση κειμένου και φωνής, γρήγορες και αξιόπιστες μηχανές αναζήτησης στο διαδίκτυο, αναγνώριση κινδύνων διαφθοράς και απάτης κ.α.

Οι αλγόριθμοι μηχανικής μάθησης χωρίζονται επιγραμματικά στις εξής κατηγορίες:

- **Επιτηρούμενη μάθηση ή Μάθηση Υπό Επίβλεψη (Supervised Learning)**, όπου ο αλγόριθμος κατασκευάζει μια συνάρτηση που απεικονίζει δεδομένες εισόδους-δείγματα (labeled samples) σε γνωστές επιθυμητές εξόδους (σύνολο εκπαίδευσης – training set), κάνοντας προβλέψεις και διορθώνοντας τις προβλέψεις σε περίπτωση λάθους, με στόχο τη γενίκευση της συνάρτησης αυτής για εισόδους με άγνωστη έξοδο (σύνολο ελέγχου – testing set). Παραδείγματα αποτελούν οι αλγόριθμοι ταξινόμησης και παλινδρόμησης (regression).
- **Μη Επιτηρούμενη Μάθηση ή Μάθηση Χωρίς Επίβλεψη (Unsupervised Learning)**, όπου ο αλγόριθμος κατασκευάζει ένα μοντέλο για κάποιο σύνολο δεδομένων εκπαίδευσης χωρίς να γνωρίζει επιθυμητές εξόδους-δείγματα (unlabeled samples) ανακαλύπτοντας δομές, όπως για παράδειγμα εξάγοντας γενικούς κανόνες. Παραδείγματα αποτελούν οι αλγόριθμοι συσταδοποίησης και οι αλγόριθμοι μείωσης διαστάσεων (dimensionality reduction).
- **Ημι-Επιτηρούμενη μάθηση (Semi-Supervised Learning)**, όπου τα δεδομένα εκπαίδευσης είναι μια μίξη γνωστών και αγνώστων δειγμάτων (mixture of labeled & unlabeled samples), όπου υπάρχει ένα επιθυμητό πρόβλημα πρόβλεψης, αλλά το μοντέλο πρέπει να ανακαλύψει δομές για να οργανώσει τα δεδομένα και να κάνει προβλέψεις. Αυτή η προσέγγιση είναι ιδιαίτερα χρήσιμη όταν η απόκτηση μεγάλων ποσοτήτων δεδομένων με ετικέτα είναι δαπανηρή ή χρονοβόρα.
- **Ενισχυμένη Μάθηση (Reinforcement Learning)**, προσέγγιση κατά την οποία ένας παράγοντας μαθαίνει να λαμβάνει μία σειρά αποφάσεων αλληλεπιδρώντας άμεσα με ένα περιβάλλον. Λειτουργεί παρατηρώντας την κατάσταση του περιβάλλοντος, επιλέγοντας ενέργειες με στόχο τη μεγιστοποίηση των αθροιστικών ανταμοιβών του και προσαρμόζοντας την πολιτική λήψης αποφάσεων μέσω δοκιμής και λάθους. Σκοπός είναι η ανακάλυψη της βέλτιστης στρατηγικής για τη μεγιστοποίηση των μακροπρόθεσμων ανταμοιβών, χρησιμοποιώντας μεθόδους όπως η εκμάθηση-Q ή τα βαθιά νευρωνικά δίκτυα. Παραδείγματα αποτελεσματικής εφαρμογής μεθόδων ενισχυτικής μάθησης σήμερα αποτελούν τα αυτόνομα ρομποτ, τα αυτό-οδηγούμενα αυτοκίνητα αλλά και υπολογιστές που μαθαίνουν να παίζουν παιχνίδια στρατηγικής (Atari, Poker, Go) με εντυπωσιακά αποτελέσματα.

### 3.4. Νευρωνικά Δίκτυα

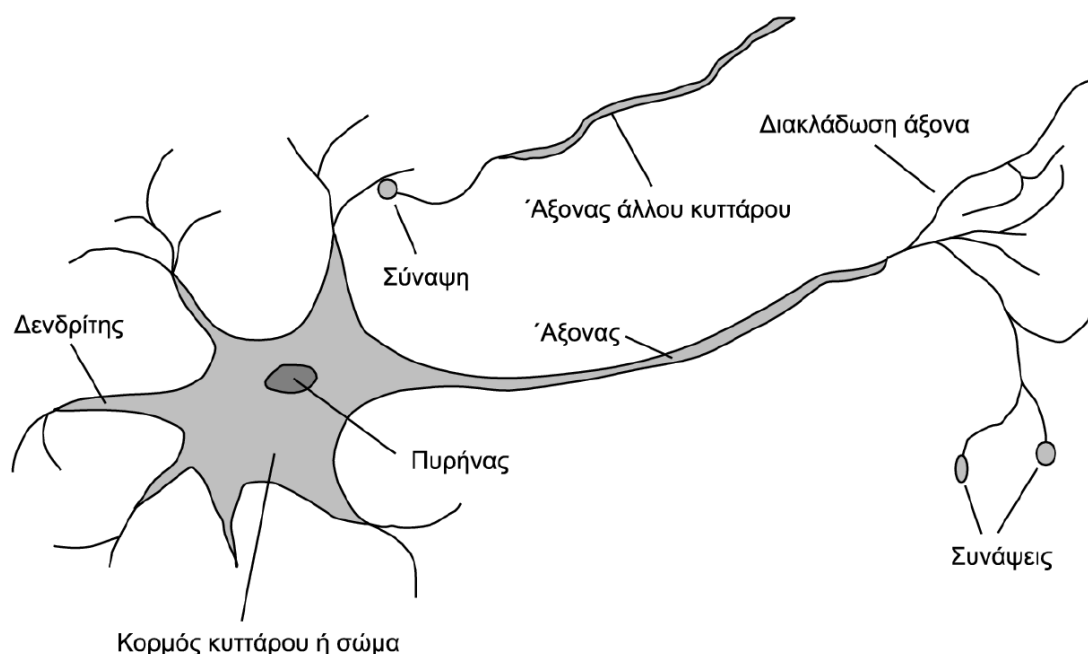
Τα Τεχνητά Νευρωνικά Δίκτυα πρωτοπαρουσιάστηκαν ως μαθηματικά μοντέλα τα οποία προσομοιώνουν την πολύπλοκη λειτουργία των νευρώνων του ανθρώπινου εγκεφάλου. Μιμούνται, κατά το δυνατό, εκείνα του **Βιολογικού Νευρωνικού Δικτύου (BND)** του εγκεφάλου μας. Με τον όρο λοιπόν Τεχνητό Νευρωνικό Δίκτυο εννοούμε ένα μαθηματικό μοντέλο που προσπαθεί να μιμηθεί τη δομή και τη λειτουργία ενός Βιολογικού Νευρωνικού δικτύου.



### 3.4.1. Βιολογικά Νευρωνικά Δίκτυα

Τα ΒΝΔ είναι το αποτέλεσμα εκατομμυρίων χρόνων εξέλιξης στον ανθρώπινο εγκέφαλο και άλλους οργανισμούς, και αποτελούν το βασικό μοντέλο για την κατανόηση του τρόπου με τον οποίο λειτουργεί η νευρολογία και η νευροεπιστήμη. Πρόκειται για μια θαυμάσια ανατομική και λειτουργική σύνθεση ενός από τα πιο πολύπλοκα και εκπληκτικά συστήματα στη φύση, τον ανθρώπινο εγκέφαλο. Αποτελούνται από δισεκατομμύρια νευρώνες, βιολογικό ιστό, χημικές ουσίες και ηλεκτρικά σήματα και είναι ο πυρήνας της γνώσης, της αντίληψης και της λήψης αποφάσεων.

Ο νευρώνας είναι ένας εξειδικευμένος τύπος κυττάρου, που αποτελεί τη βασική μονάδα των συστημάτων επεξεργασίας πληροφοριών που απαρτίζουν το νευρικό μας σύστημα. Έτσι, σε αναλογία με τους ηλεκτρονικούς υπολογιστές, οι νευρώνες αποτελούν το σύνολο των στοιχείων μνήμης, των λογικών κυκλωμάτων και των εντολών λειτουργίας του εγκεφάλου. Η δομή και οι συνδέσεις του εγκεφάλου είναι τέτοιες ώστε να καθιστούν δυνατή την παράλληλη επεξεργασία δεδομένων και τη δυνατότητα συνεχούς μάθησης μέσω αλληλεπίδρασης με το περιβάλλον. Αυτά τα δύο θεμελιώδη χαρακτηριστικά συμβάλλουν στην ικανότητα εκτέλεσης δύσκολων εργασιών (γρήγορη αναγνώριση πολύπλοκων μορφών, ταξινόμηση αντικειμένων κ.ά.), και στη συνεχή εξέλιξη και προσαρμογή, δηλαδή μαθαίνοντας από το περιβάλλον κατά την αλληλεπίδρασή του με αυτό.



**Εικόνα 3.2.** Αναπαράσταση ενός βιολογικού νευρώνα

Το ανθρώπινο σώμα αποτελείται από ένα τεράστιο αριθμό κυττάρων. Το νευρικό κύτταρο, η δομική και λειτουργική μονάδα του νευρικού συστήματος αποτελεί μια στοιχειώδη μονάδα μεταφοράς και επεξεργασίας ηλεκτρικών σημάτων και χημικών

διαδικασιών. Ένας νευρώνας αποτελείται από ένα **κυτταρικό σώμα** (ή περικάρυο) και τις αποφύσεις του, έναν άξονα και έναν ή περισσότερους **δενδρίτες**. Στο σώμα του νευρώνα γίνεται η επεξεργασία όλων των σημάτων που έχουν ληφθεί και στη συνέχεια αποστέλλεται το επεξεργασμένο σήμα μέσω του άξονα σε όλους τους γειτονικούς νευρώνες. Οι δενδρίτες επιτρέπουν την επικοινωνία μεταξύ των νευρώνων, λαμβάνοντας ηλεκτρικούς παλμούς από τον άξονα άλλων νευρώνων μέσω χημικών διεργασιών. Τα σημεία ανταλλαγής του ηλεκτρικού δυναμικού μεταξύ των νευρώνων ονομάζονται **συνάψεις**. Δηλαδή, οι συνάψεις ενώνουν τους βιολογικούς νευρώνες και δίνουν την δυνατότητα ανταλλαγής του ηλεκτρικού δυναμικού μεταξύ των δενδριτών. Ο μέσος ανθρώπινος εγκέφαλος εμπεριέχει περίπου  $10^{14}$  συνάψεις (κάθε νευρώνας συνδέεται με χιλιάδες γειτονικούς).

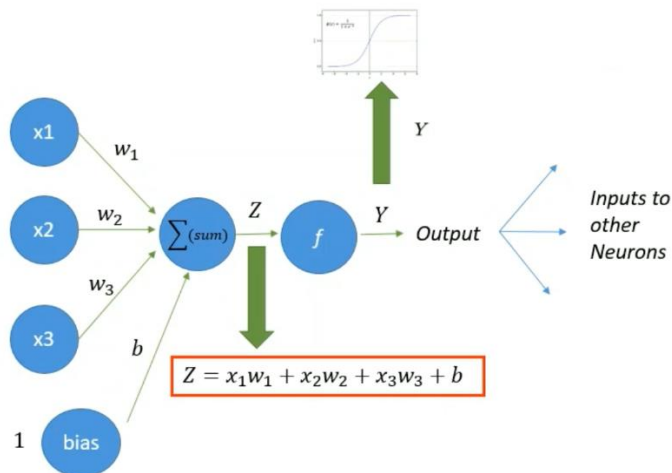
Ο αριθμός των αλληλεπιδράσεων που συμβαίνουν ανά δευτερόλεπτο στον εγκέφαλο είναι εξαιρετικά μεγάλος. Ο ακριβής αριθμός εξαρτάται από πολλούς παράγοντες, όπως η δραστηριότητα του ατόμου και η συναισθηματική του κατάσταση. Ακόμα όμως και σε κατάσταση ανάπαυσης, η εγκεφαλική δραστηριότητα είναι αρκετά υψηλή. Μερικές εκτιμήσεις υποδεικνύουν ότι συμβαίνουν περίπου  $10^{17}$  αλληλεπιδράσεις στον εγκέφαλο μας ανά δευτερόλεπτο! Ο υψηλός αυτός αριθμός αλληλεπιδράσεων αναδεικνύει την εκπληκτική πολυπλοκότητα του ανθρώπινου εγκεφάλου και τη διαρκή ενεργό διαδικασία επικοινωνίας μεταξύ νευρώνων που είναι απαραίτητη για τη λειτουργία του.

Τέλος, παρ' ότι η επεξεργαστική ισχύς ενός μεμονωμένου νευρώνα είναι πολύ μικρότερη από αυτή που διαθέτουν οι σύγχρονες ψηφιακές λογικές πύλες, οι δυνατότητες του εγκεφάλου, λόγω της παράλληλης μαζικής επεξεργασίας πληροφοριών, ξεπερνούν κάθε υπολογιστικό σύστημα που έχει κατασκευαστεί μέχρι σήμερα.

#### 3.4.2. Τεχνητά Νευρωνικά Δίκτυα

Κατ' αντιστοιχία με ένα βιολογικό δίκτυο νευρώνων, ένα **τεχνητό νευρωνικό δίκτυο** δομείται από ένα σύνολο **τεχνητών νευρώνων** που επικοινωνούν και αλληλοεπιδρούν, συνδεόμενοι μεταξύ τους με τις λεγόμενες συνάψεις (synapses). Κάθε σύναψη έχει διαφορετικό βαθμό αλληλεπίδρασης, ο οποίος καθορίζεται από τα συναπτικά βάρη (synaptic weights), τα οποία μεταβάλλονται (επαναπροσδιορίζονται) κατά τη διαδικασία της εκπαίδευσης. Πιο συγκεκριμένα, καθώς το νευρωνικό δίκτυο επικοινωνεί και προσαρμόζεται με το περιβάλλον ώστε να εκπαιδευτεί από αυτό (inputs), τα συναπτικά βάρη μεταβάλλονται συνεχώς, ενδυναμώνοντας ή αποδυναμώνοντας την ισχύ του κάθε δεσμού. Επομένως, αποτελούν την κωδικοποιημένη εμπειρική γνώση που αποκτά το νευρωνικό δίκτυο από την αλληλεπίδρασή του με το περιβάλλον. Αυτό είναι και το χαρακτηριστικό εκείνο που προσδίδει στο δίκτυο την ικανότητα για εξέλιξη και προσαρμογή στο περιβάλλον.

Ο τεχνητός νευρώνας είναι η βασική υπολογιστική μονάδα σε κάθε τεχνητό νευρωνικό δίκτυο. Είναι μια μαθηματική μονάδα που προσομοιώνει τη λειτουργία ενός βιολογικού νευρώνα. Ο τεχνητός νευρώνας δέχεται είσοδο από πολλές πηγές, εκτελεί υπολογισμούς πάνω σε αυτή την είσοδο, και παράγει μια έξοδο η οποία διανέμεται σε έναν ή περισσότερους νευρώνες.



**Εικόνα 3.3.** Αναπαράσταση ενός τεχνητού νευρώνα

Ο βασικός υπολογισμός που εκτελεί ένας τεχνητός νευρώνας είναι συνάρτηση του βάρους των εισόδων του. Κάθε είσοδος  $x = (x_1, x_2, \dots, x_n)$  (διάνυσμα εισόδου εκπαίδευσης) πολλαπλασιάζεται με τους αντίστοιχους συντελεστές **συναπτικών βαρών**  $W=(w_1, w_2, \dots, w_n)$  και στη συνέχεια αθροίζονται μαζί με μια σταθερή αριθμητική τιμή, την **σταθερά πόλωσης  $b$**  (bias unit). Αυτό το άθροισμα θα δοθεί στη συνέχεια ως όρισμα στη **συνάρτηση ενεργοποίησης (activation function)  $f(x)$**  η οποία και θα καθορίσει το αν θα πυροδοτηθεί ο νευρώνας.

Ένα τεχνητό νευρωνικό δίκτυο αποτελείται από διάφορα επίπεδα (layers), καθένα από τα οποία αποτελείται από έναν σύνολο τεχνητών νευρώνων:

- **Επίπεδο Εισόδου (Input Layer):** Αυτό το επίπεδο αποτελεί την αρχή του νευρωνικού δικτύου και δέχεται τις εισόδους από το περιβάλλον. Ο αριθμός των νευρώνων σε αυτό το επίπεδο αντιστοιχεί στον αριθμό των χαρακτηριστικών της εισόδου. Για παράδειγμα, αν επεξεργαζόμαστε εικόνες 28x28 pixels, το επίπεδο εισόδου θα έχει 784 νευρώνες (28x28). Το επίπεδο εισόδου δεν εκτελεί υπολογισμούς, δηλαδή δεν υπάρχει κάποια συνάρτηση ενεργοποίησης, αλλά απλώς διανέμει τις εισόδους στο επόμενο επίπεδο.
- **Κρυφά Επίπεδα (Hidden Layers):** Τα κρυφά επίπεδα αποτελούν τον πυρήνα του νευρωνικού δικτύου. Κάθε κρυφό επίπεδο περιλαμβάνει πολλούς τεχνητούς νευρώνες, και οι νευρώνες αυτοί επικοινωνούν μεταξύ τους μέσω συνάψεων. Ένα τεχνητό νευρωνικό δίκτυο μπορεί να έχει ένα ή περισσότερα κρυφά επίπεδα,

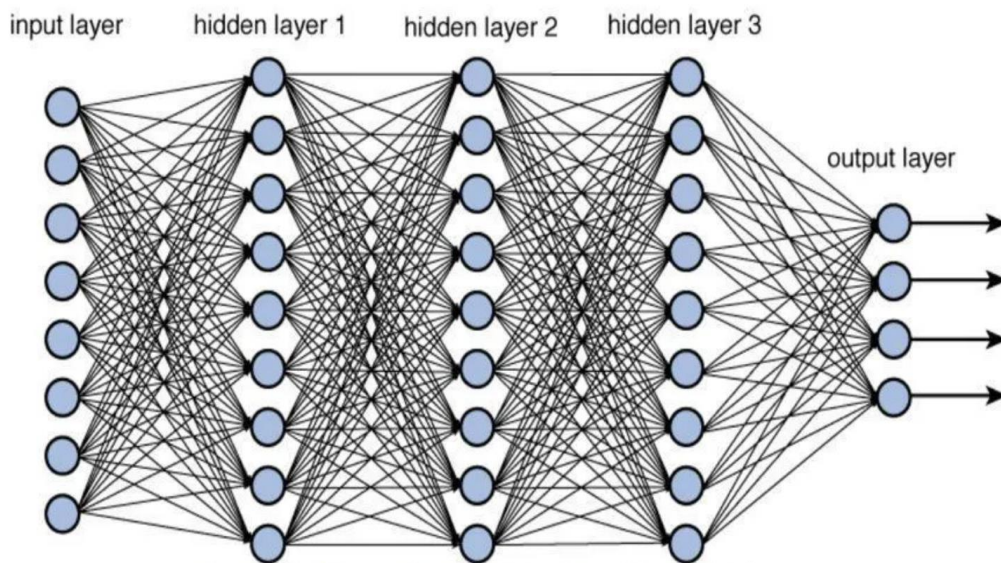
ανάλογα με την πολυπλοκότητα του προβλήματος. Ένα δίκτυο με ένα ή δύο κρυφά επίπεδα θεωρείται συνήθως "μη βαθύ" (shallow neural network). Αυτά τα δίκτυα είναι απλούστερα και προσανατολισμένα σε απλά προβλήματα. Από την άλλη, ένα δίκτυο με περισσότερα από δύο κρυφά επίπεδα θεωρείται "βαθύ" (deep neural network). Το κάθε επίπεδο εκτελεί μια σειρά υπολογισμών στις εισόδους που λαμβάνει και διαμορφώνει τα δεδομένα με βάση τα συναπτικά βάρη.

- **Επίπεδο Εξόδου (Output Layer):** Αυτό το επίπεδο παράγει τις τελικές εξόδους του νευρωνικού δικτύου με βάση την επεξεργασία που έχει διεξαχθεί στα προηγούμενα επίπεδα. Ο αριθμός των νευρώνων στο επίπεδο εξόδου εξαρτάται από τον τύπο του προβλήματος. Για παράδειγμα, σε ένα πρόβλημα δυαδικής ταξινόμησης, θα υπάρχει ένας νευρώνας εξόδου, ενώ σε ένα πρόβλημα ταξινόμησης πολλών κατηγοριών, το επίπεδο εξόδου θα έχει έναν νευρώνα για κάθε κατηγορία.

Ένα ΤΝΔ που περιέχει ενδιάμεσα επίπεδα ονομάζεται και πολύ-επίπεδο νευρωνικό δίκτυο. Τα επίπεδα δεν είναι αναγκαίο να έχουν το ίδιο πλήθος νευρώνων μεταξύ τους αλλά οι νευρώνες του ίδιου επιπέδου πρέπει να έχουν τις ίδιες συναρτήσεις ενεργοποίησης. Ανάλογα με το είδος της σύνδεσης που υπάρχει μεταξύ των νευρώνων στο κάθε επίπεδο, χωρίζονται σε **πλήρως συνδεδεμένα (fully connected layers)**, αν κάθε νευρώνας ενός επιπέδου είναι συνδεδεμένος με κάθε νευρώνα του επόμενου επιπέδου, και **μερικώς συνδεδεμένα (partially connected layers)**, αν δεν υπάρχουν συνδέσεις μεταξύ όλων των νευρώνων.

Σε ένα πλήρως συνδεδεμένο επίπεδο κάθε νευρώνας συνδέεται με κάθε νευρώνα στο επόμενο επίπεδο. Όταν οι πληροφορίες ρέουν μέσω ενός πλήρως συνδεδεμένου επιπέδου, κάθε νευρώνας μπορεί να αλληλεπιδράσει με κάθε άλλο νευρώνα στο διπλανό επίπεδο, άρα και η έξοδος κάθε νευρώνα επηρεάζεται από όλους τους νευρώνες του προηγούμενου επιπέδου. Τα πλήρως συνδεδεμένα επίπεδα χρησιμοποιούνται προς το τέλος των νευρωνικών δικτύων, όπως σε νευρωνικά δίκτυα τροφοδοσίας ή πολυεπίπεδα perceptrons, για την καταγραφή σύνθετων σχέσεων και μοτίβων στα δεδομένα.

Ένα μερικώς συνδεδεμένο επίπεδο έχει συνδέσεις που είναι πιο επιλεκτικές, καθώς μόνο ένα υποσύνολο νευρώνων από ένα επίπεδο συνδέεται με νευρώνες του επόμενου επιπέδου. Οι συνδέσεις καθορίζονται βάσει κάποιου κανόνα ή μοτίβου, όπως περιορισμοί τοπικής συνδεσιμότητας ή αραιότητας. Τα μερικώς συνδεδεμένα επίπεδα είναι λιγότερο κοινά στα παραδοσιακά νευρωνικά δίκτυα πρόσθιας τροφοδότησης (feedforward), αλλά μπορούν να βρεθούν σε εξειδικευμένες αρχιτεκτονικές όπως τα συνελκτικά νευρωνικά δίκτυα (Convolutional Neural Networks) ή τα αναδρομικά νευρωνικά δίκτυα (Recurrent Neural Networks).



**Εικόνα 3.4.** Αναπαράσταση ΤΝΔ με 3 κρυφά επίπεδα

Το κύριο πλεονέκτημα των νευρωνικών δικτύων είναι ότι έχουν τη δυνατότητα να αποθηκεύουν γνώση και εμπειρία από το περιβάλλον, την οποία στη συνέχεια μπορούν να ανακαλέσουν αξιολογώντας νέες άγνωστες ως τώρα εισόδους. Επιπλέον, ένα τεχνητό νευρωνικό δίκτυο προσφέρει την δυνατότητα γενίκευσης μίας εισόδου, δηλαδή την εξαγωγή των βασικών χαρακτηριστικών ενός συστήματος, ακόμα και από θορυβώδη δεδομένα.

#### 3.4.2.1. Συνάρτηση Ενεργοποίησης (Activation Function)

Οι συναρτήσεις ενεργοποίησης αποτελούν θεμελιώδες συστατικό των τεχνητών νευρωνικών δικτύων, παίζοντας σημαντικό ρόλο στη διαμόρφωση της ικανότητας του δικτύου να συλλαμβάνει και να αναπαριστά πολύπλοκα μοτίβα μέσα στα δεδομένα. Οι συναρτήσεις αυτές εισάγουν επίσης τη μη γραμμικότητα στο δίκτυο, επιτρέποντάς του να μαθαίνει και να προσεγγίζει ένα ευρύ φάσμα σχέσεων. Οι συναρτήσεις ενεργοποίησης καθορίζουν την έξοδο κάθε νευρώνα στο δίκτυο, χρησιμεύοντας ως το κρίσιμο όριο για την ενεργοποίηση (πυροδότηση) ή μη του νευρώνα. Η επιλογή της κατάλληλης συνάρτησης ενεργοποίησης είναι σημαντική κατά τον σχεδιασμό των νευρωνικών δικτύων, καθώς μπορεί να επηρεάσει σημαντικά την απόδοση κατά την εκπαίδευση, την ταχύτητα σύγκλισης και συνολικά την απόδοση του δικτύου. Παρακάτω δίνονται μερικές από τις πιο γνωστές συναρτήσεις ενεργοποίησης:

- **Σιγμοειδής Συνάρτηση (Sigmoid):** Χαρακτηρίζεται από την ομαλή καμπύλη που σχηματίζει σε σχήμα S, η οποία συμπιέζει τις τιμές εισόδου σε ένα εύρος τιμών μεταξύ 0 και 1. Είναι ιδιαίτερα χρήσιμη σε προβλήματα δυαδικής ταξινόμησης, καθώς αντιστοιχίζει τα δεδομένα εισόδου σε πιθανότητες, καθιστώντας την κατάλληλη για την εκτίμηση της πιθανότητας ένα σημείο δεδομένων να ανήκει σε μια συγκεκριμένη κλάση.

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

- **Βηματική Συνάρτηση (Binary Step Function):** Είναι μία πολύ απλή συνάρτηση ενεργοποίησης και χρησιμοποιείται σπάνια στη σύγχρονη βαθιά μάθηση λόγω των περιορισμών της. Λαμβάνει μια είσοδο και επιστρέφει μία από τις δύο τιμές 0 ή 1 με βάση ένα κατώφλι T (συνήθως το 0):

$$f(x) = \begin{cases} 0 & \text{for } x \leq T \\ 1 & \text{for } x > T \end{cases}$$

- **Γραμμική Συνάρτηση (Linear):** Είναι από τις απλούστερες συναρτήσεις ενεργοποίησης που χρησιμοποιούνται σε τεχνητά νευρωνικά δίκτυα. Δεν εισάγει μη γραμμικότητα στο μοντέλο, καθώς η έξοδος είναι ευθέως ανάλογη με την είσοδο.

$$f(x) = C * x$$

- **Συνάρτηση υπερβολικής εφαπτομένης (Hyperbolic Tangent):** Είναι παρόμοια με τη σιγμοειδή συνάρτηση αλλά με εύρος τιμών μεταξύ -1 και 1, ενώ έχει το κέντρο της στο 0. Αυτή η ιδιότητα μπορεί να βοηθήσει στην εκπαίδευση των νευρωνικών δικτύων επειδή βοηθά στην αποφυγή εξαφάνισης-μηδενισμού των κλίσεων (vanishing gradients) με το να επιτρέπει τόσο θετικές όσο και αρνητικές τιμές.

$$f(x) = (1 - e^{-2x}) / (1 + e^{-2x})$$

**Συνάρτηση ReLU (Rectified Linear Unit):** Είναι μια απλή και υπολογιστικά αποδοτική συνάρτηση ενεργοποίησης. Εξάγει απευθείας την είσοδο εάν αυτή είναι θετική και 0 σε κάθε άλλη περίπτωση. Χρησιμοποιείται ευρέως στη βαθιά μάθηση και είναι ιδιαίτερα αποτελεσματική στην εκπαίδευση σε βαθιά νευρωνικά δίκτυα. Ένα αρνητικό στοιχείο της συνάρτησης αυτής είναι ότι οι κλίσεις μηδενίζονται όταν η έξοδος της είναι μηδέν.

$$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$$

**Συνάρτηση Leaky ReLU:** Αποτελεί μια βελτιωμένη έκδοση της συνάρτησης ReLU. Επιτρέπει μια μικρή σταθερή κλίση ( $\alpha$ ) για τις αρνητικές τιμές εισόδου, εμποδίζοντας την μη-ενεργοποίηση των νευρώνων κατά τη διάρκεια της εκπαίδευσης.

$$f(x) = \{\alpha x\} \text{ for } x \leq 0, x \text{ for } x \geq 0$$

- **Συνάρτηση PReLU (Parametric Rectified Linear Unit):** Είναι παρόμοια με τη Leaky ReLU αλλά με τον παράγοντα ( $\alpha$ ) να είναι μια μη σταθερή παράμετρος. Αυτό επιτρέπει στο δίκτυο να προσαρμόζει την κλίση της συνάρτησης ενεργοποίησης για κάθε νευρώνα, οδηγώντας ενδεχομένως σε βελτιωμένη απόδοση.
- **Συνάρτηση ELU (Exponential Linear Unit):** Είναι μια ομαλή συνάρτηση ενεργοποίησης που έχει αρνητική κλίση για αρνητικές τιμές εισόδου, μετριάζοντας το πρόβλημα της «εξαφάνισης» των κλίσεων (vanishing gradients). Ωστόσο, μπορεί να εισάγει κάποια υπολογιστική επιβάρυνση λόγω του εκθετικού όρου.

- Εάν ( $x > 0$ ), τότε ( $f(x) = x$ ).

- Εάν ( $x \leq 0$ ), τότε ( $f(x) = a (\exp(x) - 1)$ ), όπου ( $a$ ) είναι μια παράμετρος και ( $\exp(x)$ ) αναπαριστά τη συνάρτηση εκθετικού ( $e^x$ ).

**Συνάρτηση SELU (Scaled Exponential Linear Unit):** Έχει σχεδιαστεί για να κάνει αυτο-κανονικοποίηση των νευρωνικών δικτύων. Ενσωματώνει ένα συντελεστή κλιμάκωσης ( $\lambda$ ) για τη διατήρηση της διακύμανσης στην έξοδο, κάτι που μπορεί να οδηγήσει σε γρηγορότερη σύγκλιση και γενίκευση του μοντέλου.

$$f(x) = \{ \lambda x \quad \text{αν } x > 0$$

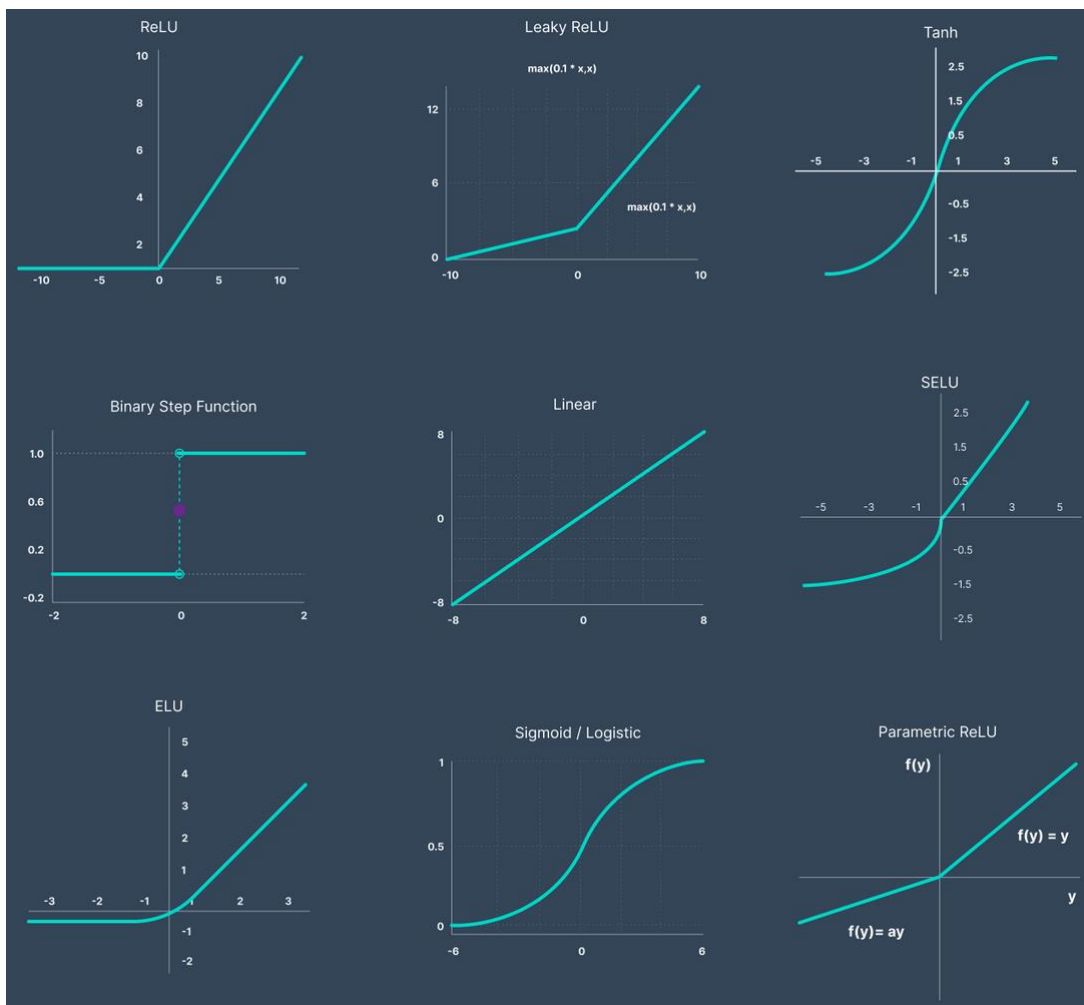
$$\lambda a (e^x - 1) \quad \text{αν } x \leq 0 \}$$



**Συνάρτηση Swish:** Σχεδιάστηκε ως μια εναλλακτική συνάρτηση ενεργοποίησης σε σχέση με άλλες δημοφιλείς συναρτήσεις όπως η ReLU και η Σιγμοειδής. Είναι ομαλή και συνεχής καμπύλη ενεργοποίησης και είναι ικανή να παράγει ανταγωνιστικά αποτελέσματα σε σχέση με συναρτήσεις όπως η ReLU, παρόλο που δεν είναι τόσο διαδεδομένη όσο αυτή.

$$f(x) = x \cdot \text{sigmoid}(x)$$

$$= x \cdot (1 + e^{(-x)})^{(-1)}$$



**Εικόνα 3.5.** Γραφική αναπαράσταση γνωστών συναρτήσεων ενεργοποίησης

**Συνάρτηση Softmax:** Χρησιμοποιείται συνήθως σε προβλήματα ταξινόμησης πολλαπλών κλάσεων και εφαρμόζεται στο επίπεδο εξόδου. Λαμβάνει ένα διάνυσμα ακατέργαστων βαθμολογιών-scores ως είσοδο και τις μετατρέπει σε κατανομή πιθανότητας σε πολλαπλές κλάσεις. Κάθε νευρώνας εξόδου αντιπροσωπεύει την πιθανότητα η είσοδος να ανήκει στην αντίστοιχη κλάση και η κλάση με την υψηλότερη πιθανότητα θεωρείται και η προβλεπόμενη κλάση. Επίσης διασφαλίζει



ότι η έξοδος που παράγει είναι κατάλληλη για τον υπολογισμό της κατηγορικής απώλειας διασταυρούμενης εντροπίας (categorical cross entropy loss) που χρησιμοποιείται συχνά κατά την εκπαίδευση ταξινομητών.

$$\sigma(z)_i = e^{z_i} / \sum_{j=1}^K e^{z_j}$$

#### 3.4.2.2. Συνάρτηση Σφάλματος (Loss Function)

Η συνάρτηση σφάλματος, επίσης γνωστή ως συνάρτηση κόστους (cost function) ή συνάρτηση απώλειας, είναι ένα ακόμη κρίσιμο στοιχείο στην εκπαίδευση των τεχνητών νευρωνικών δικτύων και άλλων μοντέλων μηχανικής μάθησης. Αποτελεί ένα μαθηματικό εργαλείο που χρησιμοποιείται για την αξιολόγηση του πόσο καλά το μοντέλο μας προβλέπει τα δεδομένα εκπαίδευσης.

Σκοπός της συνάρτησης σφάλματος είναι να μετράει κατά τη διάρκεια της εκπαίδευσης τη διαφορά μεταξύ των προβλέψεων του μοντέλου και των πραγματικών δεδομένων, και να ελαχιστοποιήσει κατά το δυνατόν το σφάλμα αυτό τροποποιώντας τις παραμέτρους του μοντέλου. Ο τύπος της συνάρτησης σφάλματος εξαρτάται από τον τύπο του προβλήματος που αντιμετωπίζουμε. Παρακάτω παρουσιάζονται ορισμένες συναρτήσεις σφάλματος ανά κατηγορία

Για προβλήματα παλινδρόμησης:

- **Μέσο Τετραγωνικό Σφάλμα (Mean Squared Error - MSE):** Χρησιμοποιείται συχνά σε προβλήματα παλινδρόμησης για να μετρήσει την τετραγωνική απόκλιση ανάμεσα στις προβλέψεις και τις πραγματικές τιμές.

$$MSE = (1 / 2n) * \sum (Y_i - \hat{Y}_i)^2$$

$\hat{Y}_i$  - Προβλεπόμενο Αποτέλεσμα

$Y_i$  - Πραγματικό Αποτέλεσμα

$n$  - Δείγματα Εκπαίδευσης σε κάθε mini-δεσμιδα

- **Μέσο Απόλυτο Σφάλμα (Mean Absolute Error):** Είναι αντίστοιχο του MSE, αλλά μετρά την απόλυτη απόκλιση αντί για την τετραγωνική.

$$MAE = (1/n) * \sum |Y_i - \hat{Y}_i|$$

- **Συνάρτηση Huber:** Αποτελεί συνδυασμό των δύο προηγούμενων συναρτήσεων. Έχει το πλεονέκτημα να είναι λιγότερο ευαίσθητη σε ακραίες τιμές σε σύγκριση με το MSE. Αυτό την καθιστά χρήσιμη όταν έχουμε δεδομένα με εκτεταμένες ακραίες τιμές που μπορεί να επηρεάσουν σημαντικά το MSE. Το κατώφλι  $\delta$  διαμορφώνει την ευαισθησία της συνάρτησης Huber στις ακραίες τιμές σφάλματος.

$$L_{\text{Huber}}(y, f(x)) = \begin{cases} (1/2)(y - f(x))^2, & \text{if } |y - f(x)| \leq \delta \\ \delta |y - f(x)| - (1/2)\delta^2, & \text{if } |y - f(x)| > \delta \end{cases}$$

$$\delta |y - f(x)| - (1/2)\delta^2, \}$$

Για προβλήματα ταξινόμησης:

- **Διαδική Διασταυρούμενη Απώλεια Εντροπίας (Binary Cross Entropy Loss):** Χρησιμοποιείται μόνο σε περιπτώσεις όπου η σωστή τιμή εξόδου είναι 0 ή 1. Η συνάρτηση ενεργοποίησης του επιπέδου εξόδου του δικτύου παράγει ένα δεκαδικό αριθμό με τιμή από 0.00 έως 1.00, ο οποίος αντιπροσωπεύει την πιθανότητα του να ανήκει η είσοδος στην προς ταξινόμηση κατηγορία. Ο αριθμός αυτός δίνεται ως όρισμα στη συνάρτηση κόστους, η οποία τον συγκρίνει με τη σωστή τιμή εξόδου, που είναι ένας ακέραιος αριθμός, 0 ή 1. Υπολογίζεται δηλαδή πόσο απέχει η τιμή εξόδου του δικτύου από την σωστή τιμή, παίρνοντας τον λογάριθμο της τιμής εξόδου του δικτύου με αρνητικό πρόσημο όταν η σωστή τιμή είναι 1 ή τον λογάριθμο της τιμής εξόδου του δικτύου αφαιρούμενης από τη μονάδα όταν η σωστή τιμή είναι 0. Το σύνολο όλων των επιμέρους διαφορών διαιρούμενο από τον αριθμό  $m$ , που αντιπροσωπεύει το πλήθος των τιμών εισόδου, δίνει το κόστος για ένα συγκεκριμένο πέρασμα των δεδομένων από το δίκτυο.

$$(1/n) * \sum (\sum [y_i * \log(p_i) + (1 - y_i) \log(1 - p_i)])$$

$y$ : Ετικέτα πραγματικής κατηγορίας (Actual class label)

$p$ : Προβλεπόμενη πιθανότητα για την κατηγορία (Predicted probability for the class)

$c$ : Αριθμός κατηγοριών (Number of classes)

$n$ : Αριθμός δειγμάτων (Number of samples)

- **Κατηγορική Διασταυρούμενη Απώλεια Εντροπίας (Categorical Cross Entropy Loss):** Χρησιμοποιείται αποκλειστικά σε περιπτώσεις ταξινόμησης μεταξύ πολλών κλάσεων. Σε αυτή την περίπτωση, αντί για έναν μοναδικό αριθμό εξόδου, χρησιμοποιείται ένα διάνυσμα αριθμών με μήκος όσο και οι τάξεις. Ο δείκτης της κάθε τάξης στο διάνυσμα έχει οριστεί εκ των προτέρων. Το διάνυσμα της σωστής εξόδου παίρνει την τιμή από 0 σε όλες τις θέσεις εκτός από αυτή της σωστής τάξης, όπου παίρνει την τιμή 1. Το διάνυσμα της εξόδου του νευρωνικού δικτύου περιλαμβάνει μια κατανομή πιθανότητας, ένα σύνολο δεκαδικών αριθμών με τιμή μεταξύ 0.00 και 1.00, καθένας από τους οποίους αντιπροσωπεύει την πιθανότητα του να ανήκει η τιμή εισόδου σε τάξη, με δείκτη που βρίσκεται στη συγκεκριμένη θέση του διανύσματος. Το άθροισμα των παραπάνω αριθμών που συμβολίζουν την εκάστοτε πιθανότητα, είναι 1. Η συγκεκριμένη συνάρτηση κόστους συγκρίνει το πόσο κοντά βρίσκεται αυτή η κατανομή πιθανοτήτων στην πραγματική κατηγοριοποίηση. Η συνολική απώλεια υπολογίζεται ως ο μέσος όρος των απωλειών για όλες τις παρατηρήσεις.

$$CE = -(1/n) * \sum(\sum[y_i * \log(y_i)])$$

#### 3.4.2.3. Οπισθοδιάδοση Σφάλματος (Backpropagation)

Σημαντικό ρόλο στην απόδοση των ΤΝΔ διαδραματίζει η δυναμική σχέση των σφαλμάτων με τις τιμές των συναπτικών βαρών εντός του δικτύου. Θεμελιώδης αλγόριθμος είναι ο **αλγόριθμος backpropagation** που χρησιμοποιείται κατά την επιβλεπόμενη μάθηση, δηλαδή στην περίπτωση που έχουμε ένα σύνολο δεδομένων για εκπαίδευση με ετικέτα, με σκοπό την ελαχιστοποίηση του σφάλματος ή της απώλειας μεταξύ της προβλεπόμενης εξόδου (prediction) και της πραγματικής εξόδου – στόχου (actual value).

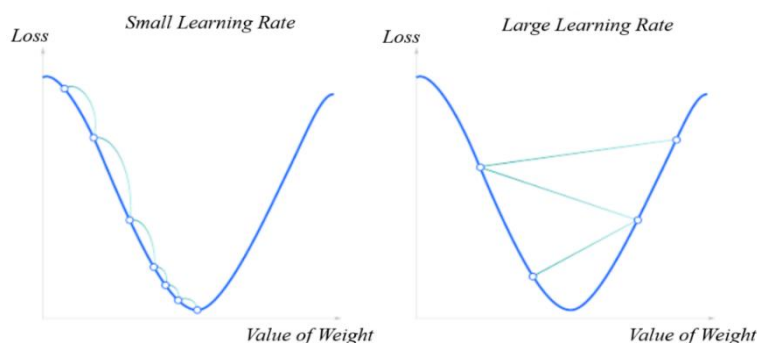
Το σφάλμα υπολογίζεται από τη **συνάρτηση σφάλματος J** συναρτήσει των **συναπτικών βαρών (w)**, των **πολώσεων (b)**, των **εισόδων (x<sub>1</sub>, x<sub>2</sub>, x<sub>3</sub>, ..., x<sub>n</sub>)**, και των **εξόδων (y<sub>1</sub>, y<sub>2</sub>, y<sub>3</sub>, ..., y<sub>n</sub>)** και στη συνέχεια διαδίδεται στο δίκτυο από το τελευταίο επίπεδο διαδοχικά προς τα πίσω με στόχο την κατάλληλη ρύθμιση των παραμέτρων έτσι ώστε να ελαχιστοποιηθεί η J. Ουσιαστικά, η διαδικασία εκπαίδευσης προσπαθεί να ρυθμίσει τα **βάρη W** και τις **σταθερές πόλωσης b**, έτσι ώστε να ελαχιστοποιηθεί η διαφορά μεταξύ των προβλέψεων του δικτύου και των πραγματικών τιμών. Τα βήματα εφαρμογής του αλγορίθμου, μπορούν να συνοψιστούν στα ακόλουθα:

1. **Αρχική απόδοση τυχαίων τιμών σε όλες τις τιμές βαρών** (π.χ. κανονικής κατανομής). Αρχικά δίδονται μικρές μη-μηδενικές τιμές καθώς μηδενικές τιμές παρουσιάζουν προβλήματα στη μάθηση του δικτύου (εξαφάνιση των κλίσεων - vanishing gradient).

2. **Προς τα εμπρός πέρασμα (Forward Pass)**, όπου υπολογίζεται το σταθμισμένο άθροισμα  $z$  για κάθε νευρώνα των κρυφών επιπέδων και εφαρμόζεται η συνάρτηση ενεργοποίησης  $f(z)$ . Το ίδιο εφαρμόζεται και στο επίπεδο εξόδου.
3. **Υπολογισμός του σφάλματος (Error Calculation)**, μεταξύ των προβλεπόμενων εξόδων και των πραγματικών τιμών-στόχων με χρήση μίας εκ των συναρτήσεων σφάλματος.
4. **Προς τα πίσω πέρασμα (Backward Pass)**, όπου ο αλγόριθμος ξεκινά από το επίπεδο εξόδου και κινείται προς τα πίσω μέσω του δικτύου, διαδίδοντας τη διαβάθμιση σφάλματος. Ο στόχος είναι να υπολογιστεί ο ρυθμός μεταβολής του σφάλματος σε σχέση με τα βάρη του δικτύου.
5. **Ενημέρωση των βαρών (Weights Update)**, όπου οι υπολογισμένες κλίσεις χρησιμοποιούνται για την αλλαγή των βαρών, με χρήση της μεθόδου καθόδου κλίσης (Gradient Descent) ή κάποιον άλλον σχετικό αλγόριθμο βελτιστοποίησης.

#### 3.4.2.4. Ρυθμός Μάθησης

Ο **ρυθμός μάθησης (learning rate)**  $\eta$  είναι μια υπερπαράμετρος που καθορίζει το μέγεθος του βήματος σε κάθε επανάληψη κατά την ενημέρωση των βαρών του μοντέλου. Ελέγχει δηλαδή το πόσο γρήγορα ή αργά θα συγκλίνει ο αλγόριθμος σε μία βέλτιστη, για την εκάστοτε ρύθμιση του δικτύου, λύση. Χρειάζεται όμως προσοχή, καθώς ενός υψηλότερος ρυθμός μάθησης οδηγεί σε ταχύτερη σύγκλιση, αλλά μπορεί να οδηγήσει σε «υπερπήδηση» του σημείου ελαχίστου σφάλματος, ενώ ένας χαμηλότερος ρυθμός μάθησης μπορεί να οδηγήσει σε πολύ αργή σύγκλιση αλλά με πιο ακριβή βήματα.



**Εικόνα 3.7.** Αναπαράσταση σύγκλισης για μικρό και μεγάλο ρυθμό μάθησης

#### 3.4.2.4. Τεχνικές Βελτιστοποίησης (Optimization)

Μερικές γνωστές τεχνικές βελτιστοποίησης του τρόπου ανανέωσης των βαρών είναι οι εξής:

- **Μέθοδος Καθόδου Κλίσης με Δεσμίδες (Batch Gradient Descent - BGD):**  
Αποτελεί θεμελιώδη αλγόριθμο βελτιστοποίησης για την εκπαίδευση μοντέλων μηχανικής μάθησης. Η ενημέρωση των παραμέτρων συμβαίνει με το πέρασμα ολόκληρου του συνόλου δεδομένων εκπαίδευσης κάθε φορά. Δηλαδή σε μία εποχή συμβαίνει μόνο μία ενημέρωση των παραμέτρων. Η μέθοδος αυτή συνήθως συγκλίνει στο ολικό βέλτιστο καθώς οι κλίσεις δεν εμπεριέχουν θόρυβο και είναι ιδανική για «καθαρά» και μικρά σύνολα δεδομένων. Το κύριο μειονέκτημα της μεθόδου είναι το υψηλό υπολογιστικό κόστος, ειδικά για μεγάλα σύνολα δεδομένων, και απαιτείται προσεκτική επιλογή του ρυθμού μάθησης. Ο κανόνας ενημέρωσης των παραμέτρων είναι:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_{\theta} \cdot J(\theta)$$

Όπου  $\theta_t$  είναι το διάνυσμα των παραμέτρων την στιγμή  $t$ , και  $J(\theta)$  το αθροιστικό σφάλμα όλων των δειγμάτων.

- **Μέθοδος Στοχαστικής Καθόδου Κλίσης (Stochastic Gradient Descent - SGD):**  
Λειτουργεί με την ενημέρωση των παραμέτρων του μοντέλου μετά το πέρασμα κάθε μεμονωμένου δείγματος των δεδομένων εκπαίδευσης. Δηλαδή κάθε εποχή αντιστοιχεί σε ένα μόνο πέρασμα ενός δείγματος. Αυτή η στοχαστική προσέγγιση εισάγει την τυχαιότητα στη διαδικασία βελτιστοποίησης, η οποία μπορεί να βοηθήσει τον αλγόριθμο να ξεφύγει από τα τοπικά ελάχιστα και να εξερευνήσει το χώρο των παραμέτρων πιο αποτελεσματικά. Ωστόσο, η στοχαστική φύση του SGD μπορεί επίσης να εισάγει προκλήσεις θορύβου και μη-σύγκλισης, οι οποίες συνήθως μετριάζονται χρησιμοποιώντας την δυναμική αλλαγή του ρυθμού εκμάθησης και τεχνικές όπως η ορμή.
- **Μέθοδος Καθόδου Κλίσης με μινι-Δεσμίδες (Mini-Batch Gradient Descent):**  
Αποτελεί μία επίσης ευρέως διαδεδομένη τεχνική βελτιστοποίησης που βρίσκει μία μέση λύση με υπολογιστικό κόστος στο μέσο μεταξύ αυτού των BGD και SGD. Το σύνολο δεδομένων εκπαίδευσης χωρίζεται σε μικρότερα τυχαία υποσύνολα, που αναφέρονται ως μινι-δεσμίδες, και μόνο μία μινι-δεσμίδα υποβάλλεται σε επεξεργασία σε κάθε εποχή. Προσφέρει σταθερότητα κατά της διάρκειας της εκπαίδευσης, χωρίς θορυβώδεις ενημερώσεις σε αντίθεση με της μέθοδο SGD. Κύριο πλεονέκτημα της μεθόδου είναι η αποτελεσματικότητα της ακόμα και σε μεγάλα σύνολα δεδομένων. Αξιοποιώντας τον **παραλληλισμό των υπολογισμών** και το σύγχρονο hardware των υπολογιστών, όπως οι κάρτες γραφικών (GPUs),

επιταχύνει τη σύγκλιση. Η επιλογή του μεγέθους της μίνι-δεσμίδα είναι αρκετά σημαντική για να υπάρξει ισορροπία μεταξύ της σταθερής σύγκλισης και της υπολογιστικής απόδοσης.

- **Ορμή (Momentum):** Χρησιμοποιείται συχνά σε συνδυασμό με την μέθοδο SGD για τη βελτίωση της απόδοσής του δικτύου. Εισάγει την έννοια της ταχύτητας (velocity) στην ενημέρωση των βαρών, βοηθώντας στην αντιμετώπιση των ταλαντώσεων και της αργής σύγκλισης. Η ορμή διατηρεί έναν όρο ταχύτητας, ο οποίος είναι ένας εκθετικά σταθμισμένος κινητός μέσος όρος των προηγούμενων κλίσεων. Αυτός ο όρος συσσωρεύει ορμή καθώς κινείται σταθερά προς μια συγκεκριμένη κατεύθυνση, εξομαλύνοντας τις κλίσεις και αποτρέποντας τις γρήγορες αλλαγές κατεύθυνσης. Αυτό έχει ως αποτέλεσμα, οι ενημερώσεις να γίνονται λιγότερο ασταθείς και ο αλγόριθμος τείνει να συγκλίνει ταχύτερα σε μια καλή λύση. Η ορμή είναι ιδιαίτερα χρήσιμη όταν αντιμετωπίζουμε συναρτήσεις κόστους που έχουν πολλά οροπέδια ή απότομες κοιλάδες.

$$vdw = \beta vdw + (1-\beta)dw \text{ (momentum step)}$$

$$vdb = \beta vdb + (1-\beta)db \text{ (step)}$$

$$w = w - \alpha vdw \text{ (weight update)}$$

$$b = b - \alpha vdb \text{ (bias update)}$$

**Εικόνα 3.8.** Επίδραση ορμής στην ενημέρωση των παραμέτρων (όπου  $0 \leq \beta \leq 1$ , και  $\alpha$  ρυθμός μάθησης)

- **RMSProp (Root Mean Square Propagation):** Η τεχνική αυτή αντιμετωπίζει ζητήματα που σχετίζονται με τη σύγκλιση των αλγορίθμων καθόδου κλίσης, ιδιαίτερα προβλήματα που προκαλούνται από την εξαφάνιση (vanishing) ή την έκρηξη (exploding) των κλίσεων. Αναπτύχθηκε προκειμένου να αντιμετωπίσει το πρόβλημα της επιλογής του ρυθμού μάθησης, καθώς σε κάθε βήμα τον προσαρμόζει για κάθε παράμετρο. Για να το πετύχει αυτό, διατηρεί έναν κινούμενο μέσο όρο των τετραγώνων των κλίσεων κάθε στοιχείου και διαιρεί τη νέα κλίση στοιχείο προς στοιχείο με αυτό. Στη χρονική στιγμή  $t$  της διαδικασίας βελτιστοποίησης θα έχουμε ότι:

$$g_t = (\nabla_{\theta} J(\theta))$$

$$E[g^2]_t = \gamma E[g^2]_{t-1} + (1 - \gamma)g_t^2$$

$$\theta_{t+1} = \theta_t - \eta \frac{g_t}{\sqrt{E_t + \epsilon}}$$

όπου  $\eta$  είναι ο ρυθμός μάθησης χωρίς προσαρμογή,  $\theta_t$  είναι το διάνυσμα των παραμέτρων την χρονικών στιγμή  $t$ ,  $g_t$  το διάνυσμα της κλίσης,  $E[g^2]_t$  είναι ο κινούμενος μέσος όρος των τετραγώνων των κλίσεων κάθε στοιχείου,  $\gamma$  είναι η υπέρ-παραμέτρος που καθορίζει το βάρος των παλαιότερων τιμών και το  $\epsilon$  χρησιμοποιείται για την αποφυγή διαίρεσης με το μηδέν. Οι προτεινόμενες τιμές των  $\gamma, \eta$ , και  $\epsilon$  είναι 0.9, 0.001 και  $10^{-8}$  αντίστοιχα.

- **Adam (Adaptive Moment Estimation):** Ο αλγόριθμος Adam συνδυάζει την προσαρμογή του ρυθμού μάθησης του RMSProp με έναν όρο ορμής (momentum). Συγκεκριμένα, διατηρεί έναν κινούμενο μέσο όρο  $m_t$  των κλίσεων  $g_t$ , καθώς και έναν κινούμενο μέσο όρο  $v_t$  των τετραγώνων  $g_t^2$ . Οι εξισώσεις που περιγράφουν τη μεταβολή των παραμέτρων στο βήμα  $t$  είναι:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

$$\hat{m}_t = \frac{m_t}{1 - (\beta_1)^t}$$

$$\hat{v}_t = \frac{v_t}{1 - (\beta_2)^t}$$

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

Οι τιμές των  $\eta, \beta_1, \beta_2, \epsilon$  που συνιστώνται από τους συγγραφείς του αλγορίθμου είναι 0.001, 0.9, 0.999,  $10^{-8}$  αντίστοιχα.

#### 3.4.2.5. Επιλογή Αρχιτεκτονικής

Πριν ξεκινήσει η συλλογή δεδομένων και η εκπαίδευση ενός νευρωνικού δικτύου, λαμβάνονται σημαντικές αποφάσεις σχετικά με την αρχιτεκτονική του μοντέλου. Καταρχάς, προσδιορίζεται ο αριθμός των νευρώνων του επιπέδου εισόδου (input

layer), ο οποίος πρέπει να είναι ίσος με τον αριθμό των χαρακτηριστικών των δεδομένων εισόδου που έχουν επιλεγεί. Αυτό εξασφαλίζει ότι τα δεδομένα εισόδου ταιριάζουν με τη δομή του δικτύου και μπορούν να προωθηθούν μέσα σε αυτό.

Ο αριθμός των νευρώνων του επιπέδου εξόδου (output layer) πρέπει να είναι ίσος με τον αριθμό των κατηγοριών που θα χρησιμοποιηθούν για την ταξινόμηση, για προβλήματα ταξινόμησης, ή σε περιπτώσεις προβλημάτων παλινδρόμησης, μπορεί να υπάρχει ένας μοναδικός νευρώνας εξόδου. Για παράδειγμα, ένα τέτοιου τύπου δίκτυο χρησιμοποιείται για να προβλέψει την αξία μιας κατοικίας.

Στη συνέχεια επιλέγεται ο αριθμός των ενδιάμεσων (κρυφών) επιπέδων και του αριθμού των νευρώνων σε κάθε ένα από αυτά. Αυτό θα καθορίσει την πολυπλοκότητα του μοντέλου και τη δυνατότητά του να αντιμετωπίζει διάφορα επίπεδα προσέγγισης των δεδομένων. Επιπλέον, γίνεται επιλογή της συνάρτησης σφάλματος, που χρησιμοποιείται για την εκπαίδευση του μοντέλου, της συνάρτησης ενεργοποίησης των ενδιάμεσων επιπέδων και του επιπέδου εξόδου. Οι επιλογές αυτές επηρεάζουν την επίδοση και την ικανότητα του μοντέλου να μάθει από τα δεδομένα εκπαίδευσης και να γενικεύει ως προς άγνωστα δεδομένα.

Τέλος, πολύ καλά αποτελέσματα είναι πιθανό να επιτευχθούν και με τη χρήση έτοιμων αρχιτεκτονικών, προ-εκπαιδευμένα δηλαδή μοντέλων που έχουν εκπαιδευτεί σε πολύ μεγάλα σύνολα δεδομένων και είναι αποδοτικά σε συγκεκριμένα είδη εργασιών. Παραδείγματα αποτελούν τα μοντέλα Resnet, Inception για εικόνα και βίντεο, τα μοντέλα BERT, GPT (Generative Pre-Trained Transformer) για επεξεργασία φυσικής γλώσσας κ.α. Αυτά τα μοντέλα μπορούν να χρησιμοποιηθούν ως η βάση για την προσαρμογή σε κάποιο δικό μας πρόβλημα με λίγη επιπρόσθετη ή και καθόλου εκπαίδευση στα νέα δεδομένα.

#### *3.4.2.6. Δημιουργία του συνόλου δεδομένων εκπαίδευσης*

Η δημιουργία του συνόλου δεδομένων εκπαίδευσης (training set) είναι ένα αναπόσπαστο κομμάτι της διαδικασίας ανάπτυξης του τεχνητού νευρωνικού δικτύου. Αφού έχουν αποφασιστεί τα χαρακτηριστικά των δεδομένων εισόδου και η αρχιτεκτονική του δικτύου, πρέπει να επιλεγούν τα δεδομένα που θα χρησιμοποιηθούν για την εκπαίδευση του μοντέλου.

Υπάρχουν διάφορες πηγές από τις οποίες μπορούν να προέλθουν τα δεδομένα εκπαίδευσης. Τα δεδομένα σπανίως δημιουργούνται εξαιτίας της ανάγκης εκπαίδευσης του νευρωνικού δικτύου, αλλά συχνότερα είναι ήδη διαθέσιμα εντός της εταιρείας ή του οργανισμού που θέλει να κατασκευάσει το μοντέλο. Επιπλέον, μπορεί να υπάρχουν διαθέσιμα δεδομένα σε εναλλακτικές πηγές, όπως στο διαδίκτυο, που μπορούν να χρησιμοποιηθούν.



Σε κάθε περίπτωση, τα δεδομένα απαιτούν επεξεργασία πριν χρησιμοποιηθούν για την εκπαίδευση. Για παράδειγμα, είναι συνηθισμένο να γίνεται αναστροφή των αριθμών που αντιπροσωπεύουν τα pixel μιας φωτογραφίας, ώστε να μετατραπούν σε ένα μονοδιάστατο διάνυσμα για κάθε εικονοστοιχείο. Επίσης, μη αριθμητικές τιμές μπορεί να μετατραπούν σε αριθμητικές τιμές.

Το σύνολο δεδομένων χωρίζεται συνήθως σε δύο κατηγορίες: το σετ εκπαίδευσης (training set) και το σετ ελέγχου (test set). Η αναλογία μεταξύ τους είναι συνήθως 70-80% για το σετ εκπαίδευσης και 20-30% για το σετ ελέγχου. Σε ορισμένες περιπτώσεις, ένα τμήμα του σετ ελέγχου μπορεί να χωριστεί επίσης κατά ένα ποσοστό (συνήθως 50%) το οποίο θα οριστεί ως σετ επαλήθευσης (validation set). Το σετ επαλήθευσης χρησιμοποιείται για την αξιολόγηση της απόδοσης του μοντέλου κατά τη διάρκεια της εκπαίδευσης, για λόγους παρακολούθησης της προόδου, επιλογής υπερπαραμέτρων ή έγκαιρου σταματήματος της εκπαίδευσης.

#### *3.4.2.7. Διαδικασία Εκπαίδευσης*

Η εκπαίδευση ενός νευρωνικού δικτύου (στο πλαίσιο επιβλεπόμενης μάθησης) γίνεται με μια σειρά διαπεράσεων του σετ εκπαίδευσης και ανάλογη ρύθμιση των παραμέτρων του, δηλαδή των συναπτικών βαρών κάθε επιπέδου. Στόχος της είναι η προοδευτική αύξηση της ακρίβειας του δικτύου τόσο αναφορικά με το σετ εκπαίδευσης όσο και με το σύνολο δοκιμής. Πιο συγκεκριμένα, κάθε είσοδος που ανήκει στο σύνολο δεδομένων εκπαίδευσης εισάγεται στο νευρωνικό δίκτυο και στη συνέχεια πραγματοποιούνται οι προσθήσεις, οι πολλαπλασιασμοί και εφαρμόζονται οι συναρτήσεις ενεργοποίησης σε κάθε ένα από τα επίπεδα του δικτύου μέχρι και το επίπεδο εξόδου, όπου δημιουργείται το διάνυσμα της πρόβλεψης του δικτύου.

Ακολουθεί η σύγκριση με το σωστό διάνυσμα εξόδου και μετά τον υπολογισμό του σφάλματος με χρήση της συνάρτησης σφάλματος γίνεται εφαρμογή της μεθόδου της οπισθοδιάδοσης του λάθους (back propagation). Η παραπάνω διαδικασία ακολουθείται για όλες τις εισόδους του σετ εκπαίδευσης.

Στη συνέχεια υπολογίζεται η ακρίβεια (accuracy) των προβλέψεων του νευρωνικού δικτύου πάνω στο σετ εκπαίδευσης. Πρόκειται για το ποσοστό συμφωνίας με τις επιθυμητές προβλέψεις. Μετρώνται δηλαδή οι σωστές προβλέψεις του δικτύου και δίνεται ένα ποσοστό της επί τοις εκατό ευστοχίας. Επίσης υπολογίζεται η ακρίβειά του και πάνω στα σετ επαλήθευσης και ελέγχου, τα οποία περιέχουν άγνωστα στο δίκτυο δεδομένα. Η ακρίβεια αυτή αποτελεί ένα δείγμα του πόσο καλά μπορεί να γενικεύσει τις προβλέψεις του το νευρωνικό δίκτυο πέραν των γνωστών στις οποίες προσπαθεί να προσαρμοστεί.

Κατά τη διάρκεια της εκπαίδευσης και μέσω της εκτίμησης της ακρίβειας του νευρωνικού δικτύου στα δεδομένα του σετ επαλήθευσης, δίνονται επίσης ενδείξεις

για το αν το νευρωνικό δίκτυο έχει αρχίσει να εμφανίζει το φαινόμενο της **υπερ-προσαρμογής (over-fitting)** ή της **υπο-προσαρμογής (under-fitting)**.

**Υπερ-προσαρμογή** υφίσταται όταν τα συναπτικά βάρη του νευρωνικού δικτύου έχουν τιμές τέτοιες ώστε οι προβλέψεις να επαληθεύουν τέλεια τις σωστές για το σετ εκπαίδευσης αλλά να μην προσεγγίζουν επαρκώς αυτές για τα νέα, όχι γνωστά δεδομένα. Από την άλλη, η **υπο-προσαρμογή** αφορά στην αδυναμία προσέγγισης των σωστών τιμών ακόμα και για τα δεδομένα εκπαίδευσης, κάτι που επίσης οδηγεί και σε αντίστοιχη αδυναμία προσέγγισης των σωστών τιμών για άγνωστα δεδομένα.

Έπειτα από τον υπολογισμό της ακρίβειας των προβλέψεων για τα σετ εκπαίδευσης, επαλήθευσης και ελέγχου, προσαρμόζονται αναλόγως τα βάρη σε κάθε επίπεδο του δικτύου και ξεκινά η επόμενη διαπέραση. Μετά από έναν συνήθως μεγάλο αριθμό διαπεράσεων, αν η ακρίβεια των προβλέψεων του νευρωνικού δικτύου επί του σετ εκπαίδευσης και με εξέταση πάνω στο σετ επαλήθευσης κριθεί ικανοποιητική, η διαδικασία της εκπαίδευσης θα πρέπει να σταματήσει. Αν αυτό δεν συμβεί, παρόλο που η ακρίβεια στο σετ εκπαίδευσης θα συνεχίσει να αυξάνεται περίπου με κάθε ολοκληρωμένο πέρασμα του συνόλου των δεδομένων (εποχή – epoch), η ακρίβεια στα σετ επαλήθευσης και ελέγχου θα αρχίσει να μειώνεται, κάτι που αποτελεί σαφή ένδειξη υπερ-προσαρμογής στα δεδομένα της εκπαίδευσης.

Γενικότερα, κατά την εκπαίδευση κάθε νευρωνικού δικτύου αναζητείται η χρυσή τομή μεταξύ των φαινομένων της υπερ-προσαρμογής και της υπο-προσαρμογής. Χρειάζεται δηλαδή το δίκτυο να έχει αρκετά μεγάλη ακρίβεια πρόβλεψης επί των δεδομένων του σετ εκπαίδευσης αλλά και αρκετά μεγάλη επί αυτών του σετ επαλήθευσης και ελέγχου. Πρόκειται για μια επίπονη διαδικασία που απαιτεί μεθόδευση και μια σειρά δοκιμών μέχρι να επιτευχθεί το επιθυμητό αποτέλεσμα.

#### *3.4.2.8. Χρήση του εκπαιδευμένου μοντέλου*

Κατά τη διάρκεια αυτού του σταδίου, το εκπαιδευμένο μοντέλο χρησιμοποιείται για να προβλέψει αποτελέσματα ή εκτιμήσεις για νέα δεδομένα που δεν έχει ξαναδεί, κάτι για το οποίο έχουμε ήδη μία καλή εικόνα παρακολουθώντας κατά την εκπαίδευση την ακρίβεια που εμφάνισε στα σετ επαλήθευσης και ελέγχου. Αυτό μπορεί να περιλαμβάνει την ταξινόμηση αντικειμένων, την αναγνώριση προτύπων, τον προσδιορισμό τιμών ή προβλέψεις για μελλοντικά γεγονότα.

Κατά την πρόβλεψη, το μοντέλο εφαρμόζει τις «γνώσεις» που έχει αποκτήσει κατά τη διάρκεια της εκπαίδευσης του. Το αποτέλεσμα αυτής της διαδικασίας μπορεί να είναι, για παράδειγμα, η ταξινόμηση μιας εικόνας ως "σκύλος" ή "γάτα," η πρόβλεψη της τιμής ενός ακινήτου, ή η ανίχνευση ανωμαλιών (anomaly detection) σε δεδομένα.

Το στάδιο αυτό καθορίζει την γενικότερη χρησιμότητα και την αξιοπιστία του νευρωνικού δικτύου για συγκεκριμένες εφαρμογές. Αν το μοντέλο δεν παρουσιάζει

ικανοποιητική απόδοση, πρέπει να ακολουθήσουν συγκεκριμένα βήματα για να αντιμετωπιστεί το πρόβλημα. Αυτά μπορεί να είναι:

- **Βελτιστοποίηση των υπερπαραμέτρων:** Οι υπερπαραμέτροι του μοντέλου, όπως ο ρυθμός μάθησης, το μέγεθος του δικτύου και ο αριθμός των εποχών εκπαίδευσης, μπορούν να προσαρμοστούν για να βελτιωθεί η απόδοση του μοντέλου. Μικρές αλλαγές σε αυτές τις παραμέτρους μπορεί να έχουν σημαντικές επιπτώσεις.
- **Συλλογή περισσότερων δεδομένων:** Ενίοτε, τα μοντέλα χρειάζονται μεγαλύτερο όγκο δεδομένων για να βελτιώσουν την απόδοσή τους. Η συλλογή περισσότερων δεδομένων εκπαίδευσης μπορεί να βοηθήσει στην αντιμετώπιση της υπο-εκπαίδευσης (under-fitting).
- **Βελτίωση των δεδομένων:** Αυτό αφορά στην προ-επεξεργασία των δεδομένων και συγκεκριμένες τεχνικές προσαρμογής των δεδομένων στο σύστημα (data augmentation) που θα αναλυθούν παρακάτω.
- **Αλλαγή της αρχιτεκτονικής:** Ενδεχομένως, η αρχιτεκτονική του νευρωνικού δικτύου να μην είναι η καλύτερη για την συγκεκριμένη εργασία. Σε αυτήν την περίπτωση, μπορεί να χρειαστεί να δοκιμαστεί κάποια άλλη αρχιτεκτονική για το δίκτυο.

## 4. Συνελικτικά Νευρωνικά Δίκτυα (CNNs)

### 4.1. Εισαγωγή

Τα **Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs)** αποτελούν μια εξειδικευμένη κατηγορία νευρωνικών δικτύων που έχουν ευρεία εφαρμογή σε προβλήματα επεξεργασίας εικόνων, ήχων και φυσικής γλώσσας. Το όνομά τους προέρχεται από τη μαθηματική διαδικασία της **συνέλιξης (convolution)**, την οποία εφαρμόζουν σε τουλάχιστον ένα από τα επίπεδά τους, αντί να βασίζονται σε πολλαπλασιασμό πινάκων.

Τα ΣΝΔ έχουν ειδική αρχιτεκτονική, σχεδιασμένη έτσι ώστε να αναγνωρίζει δισδιάστατα σχήματα με υψηλή ακρίβεια. Αυτή η αρχιτεκτονική εκμεταλλεύεται τη δομή των εικόνων και άλλων δισδιάστατων σημάτων, κάτι που τα καθιστά αποτελεσματικά στην επεξεργασία τέτοιων δεδομένων.

Σήμερα, τα ΣΝΔ αποτελούν την κορυφαία τεχνολογία στον τομέα της αναγνώρισης προτύπων, ιδίως στην κατηγοριοποίηση και ταξινόμηση εικόνων. Επίσης, έχουν εφαρμογές στην ταξινόμηση ιατρικών δεδομένων. Ένα από τα κύρια πλεονεκτήματα των CNN είναι η ευκολία στην εκπαίδευσή τους και ο αριθμός των παραμέτρων που

χρειάζονται, ο οποίος είναι πολύ μικρότερος σε σχέση με τα πλήρως συνδεδεμένα νευρωνικά δίκτυα με ίδιο αριθμό κρυφών επιπέδων.

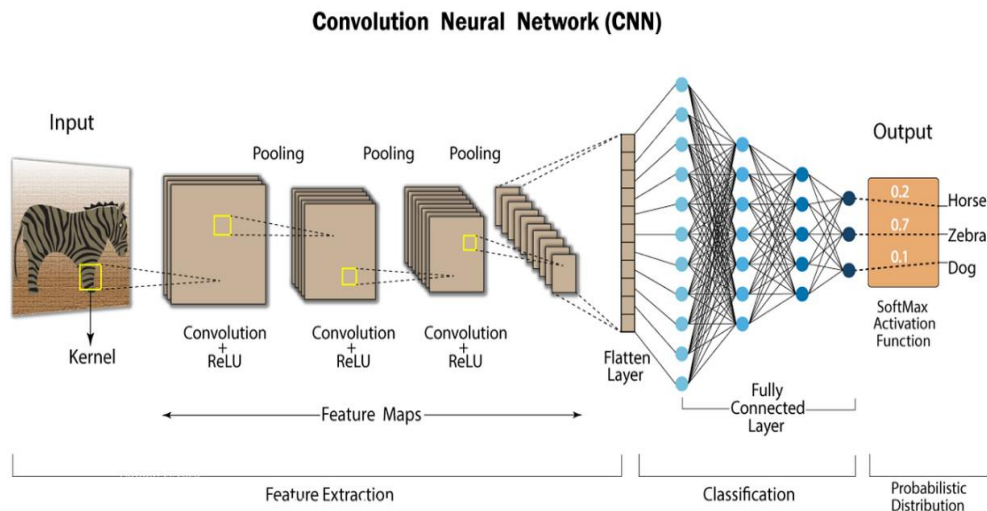
Στη συνέχεια του κεφαλαίου, γίνεται μια λεπτομερής ανάλυση της δομής των CNNs και των τεχνικών που χρησιμοποιούνται για τη βελτίωση της απόδοσής τους. Επίσης, παρουσιάζονται διάφορες αρχιτεκτονικές που έχουν προταθεί τα τελευταία χρόνια, οι οποίες μπορούν να εφαρμοστούν σε ποικίλες εφαρμογές.

## 4.2. Δομή των Συνελικτικών Νευρωνικών Δικτύων

Τα συνελικτικά νευρωνικά δίκτυα είναι ουσιαστικά δομημένα ως μια διαδοχική σειρά επιπέδων, καθένα από τα οποία πραγματοποιεί έναν ξεχωριστό μετασχηματισμό του όγκου δεδομένων εισόδου. Είναι σχεδιασμένα έτσι ώστε να επεξεργάζονται διαδοχικά δομές, ξεκινώντας συχνά με εικόνες και στη συνέχεια υποβάλλοντάς τις σε μία συστηματική αλληλουχία εξειδικευμένων επιπέδων. Τα επίπεδα αυτά είναι προσαρμοσμένα ώστε να βελτιώνουν και να αφαιρούν σταδιακά ορισμένα χαρακτηριστικά από τα δεδομένα εισόδου, καθιστώντας τα πιο διαχειρίσιμα.

Τα βασικά επίπεδα που αποτελούν ένα ΣΝΔ είναι τα εξής:

- **Συνελικτικά Επίπεδα (Convolutional Layers):** Τα συνελικτικά επίπεδα εφαρμόζουν φίλτρα στην εικόνα εισόδου για την εξαγωγή σχετικών χαρακτηριστικών. Αυτά τα φίλτρα «γλιστρούν» πάνω πάνω στα δεδομένα εισόδου, εκτελώντας συνελίξεις για να αναγνωρίσουν μοτίβα, άκρες, υφές και άλλα.
- **Μη-Γραμμικά Επίπεδα (Non-Linear Layers):** Αυτά τα επίπεδα εισάγουν μη γραμμικότητα στα δίκτυα, επιτρέποντας τους να μοντελοποιούν περίπλοκες, μη γραμμικές σχέσεις εντός των δεδομένων. Κοινές συναρτήσεις ενεργοποίησης αποτελούν οι συναρτήσεις ReLU, σιγμοειδής και υπερβολική εφάπτομένη.
- **Επίπεδα Συγκέντρωσης (Pooling Layers):** Γνωστά και ως επίπεδα υπο-δειγματοληψίας (sub-sampling), αυτά τα επίπεδα μειώνουν τις διαστάσεις των χαρτών χαρακτηριστικών (feature maps) που δημιουργούνται μετά από τα συνελικτικά επίπεδα. Κοινές τεχνικές που χρησιμοποιούνται είναι το max-pooling και το average-pooling για τη μείωση των διαστάσεων δείγματος και την εξαγωγή των πιο σημαντικών χαρακτηριστικών, μειώνοντας την υπολογιστική πολυπλοκότητα του δικτύου.
- **Πλήρως Συνδεδεμένα Επίπεδα (Fully Connected Layers):** Τα συναντάμε κατά κανόνα στο τέλος του δικτύου καθώς είναι υπεύθυνα για την ανάλυση των χαρακτηριστικών που έχουν εξάγει τα προηγούμενα συνελικτικά και συγκεντρωτικά επίπεδα από τα δεδομένα εισόδου αλλά και για την εξαγωγή του τελικού αποτελέσματος.

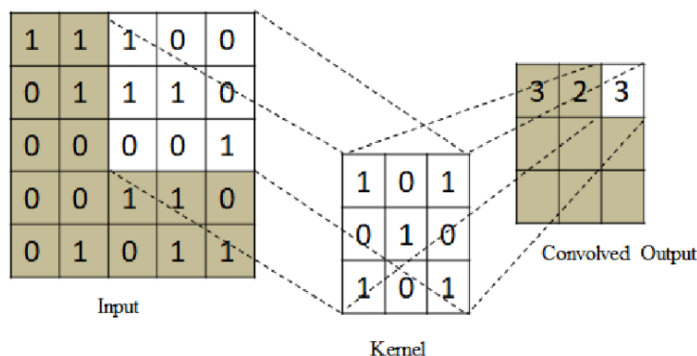


**Εικόνα 4.1.** Αναπαράσταση δομής συνελικτικού νευρωνικού δικτύου

#### 4.2.1. Συνελικτικά Επίπεδα (Convolutional Layers)

Τα συνελικτικά επίπεδα αποτελούν θεμελιώδες στοιχείο των ΣΝΔ, καθώς διαθέτουν προσδίδουν την ικανότητα εξαγωγής χαρακτηριστικών από δεδομένα εικόνων. Λειτουργούν χρησιμοποιώντας μικρά, φίλτρα (filters ή kernels) που μπορούν να εκπαιδευτούν από το δίκτυο και που ολισθαίνουν στα δεδομένα εισόδου (συνέλιξη), καταγράφοντας τοπικά πρότυπα και δομές. Είναι συχνά μικροί πίνακες που βοηθούν στην ανίχνευση χαρακτηριστικών όπως ακμές, υφές ή σχήματα. Η συνέλιξη μεταξύ της εικόνας και του φίλτρου οδηγεί στη δημιουργία του λεγόμενου χάρτη χαρακτηριστικών (feature map). Σε κάθε συνελικτικό επίπεδο μπορεί να υπάρχουν πολλαπλά φίλτρα ή ομάδες φίλτρων διαφορετικών μεγεθών, για να γίνεται δυνατή ή επεξεργασία εισόδων διαφορετικών μοτίβων.

Κατά την πρόσθια διάδοση, όλα τα φίλτρα σαρώνουν την εικόνα σε όλο το πλάτος και το ύψος της εισόδου και υπολογίζουν το γινόμενο μεταξύ των αντίστοιχων θέσεων του φίλτρου και της εισόδου σε κάθε θέση. Έτσι το δίκτυο αντιλαμβάνεται σε ποιο σημείο της εικόνας έχει ενεργοποιηθεί το αντίστοιχο φίλτρο, δηλαδή αντιλαμβάνεται το πότε θα ενεργοποιηθεί όταν έρχεται σε επαφή με ακμές ή γωνίες ή σε επίπεδα υφής (texture). Η έξοδος του επιπέδου προκύπτει από τη «στοίβα» όλων των χαρτών ενεργοποίησης.



**Εικόνα 4.2.** Ένα στιγμιότυπο εφαρμογής φίλτρου 3x3 σε εικόνα 5x5 (no padding)

Υπάρχουν τρεις υπερπαράμετροι που αφορούν τα φίλτρα και τον καθορισμό της λειτουργίας τους:

- **Μέγεθος Φίλτρου (Kernel Size):** Το μέγεθος του συνελκτικού φίλτρου πρέπει να ταιριάζει με τα δεδομένα εισόδου και είναι συνήθως ένας τετράγωνος πίνακας με διαστάσεις όπως 3x3, 5x5 ή 7x7 (για 2D δεδομένα εισόδου). Σε περίπτωση που η είσοδος είναι εικόνα RGB, άρα τριών διαστάσεων, τότε και τα συνελκτικά φίλτρα θα πρέπει να είναι τρισδιάστατα. Η επιλογή του μεγέθους καθορίζει το πεδίο υποδοχής (receptive field) του φίλτρου.
- **Βήμα (Stride):** Καθορίζει το πως ολισθαίνουν πάνω στην εικόνα τα φίλτρα σε κάθε βήμα. Όσο μεγαλύτερο το άλμα τόσο μικρότερες οι διαστάσεις της εξόδου που προκύπτει από το συνελκτικό φίλτρο.
- **Γέμισμα Περιθωρίου (Padding):** Πρόκειται για γραμμές και στήλες με μηδενικά που τοποθετούνται γύρω από τον πίνακα εισόδου. Χρησιμοποιείται για να ελέγχει το μέγεθος της εξόδου αλλά και για να μην χάνεται πληροφορία από τα περιθώρια της εικόνας. Η πληροφορία αυτή χάνεται γιατί σε περίπτωση που δεν υπάρχει γέμισμα περιθωρίου, το συνελκτικό φίλτρο θα ολισθήσει μία μόνο φορά πάνω στην πρώτη και τελευταία γραμμή του πίνακα εισόδου αλλά δύο στις ενδιάμεσες. Το αντίστοιχο συμβαίνει και κατά μήκος των στηλών. Με την προσθήκη μηδενικών γύρω από την είσοδο αυτό διορθώνεται, χωρίς να επηρεάζεται το αποτέλεσμα της συνέλιξης. Διακρίνουμε δυο περιπτώσεις:

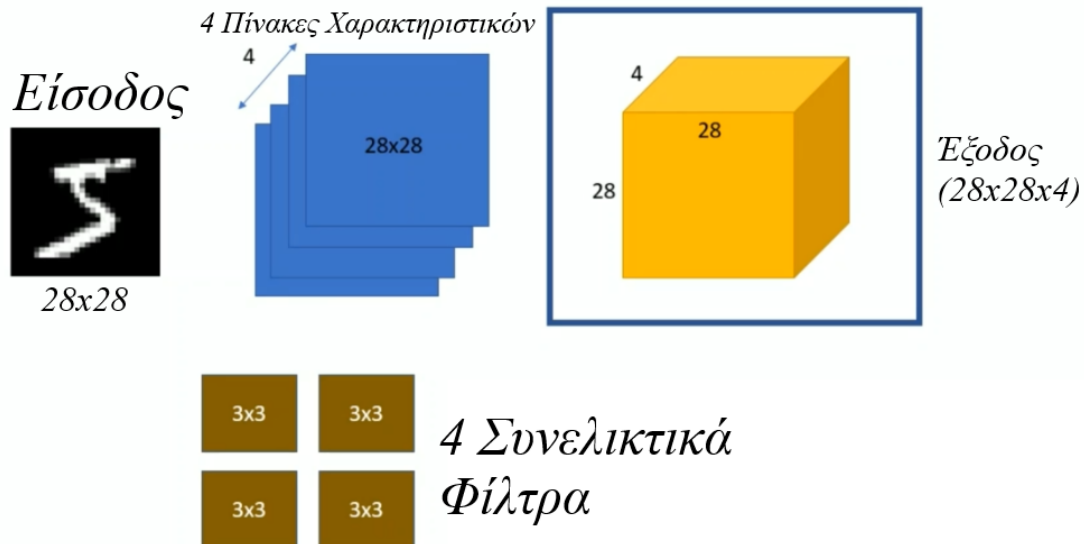
1. **Valid ή No Padding:** Δεν εφαρμόζεται γέμισμα περιθωρίου άρα το μέγεθος του πίνακα εξόδου μειώνεται.

**Same ή Zero Padding:** Εφαρμόζεται γέμισμα περιθωρίου ώστε να εξασφαλιστεί ότι ο πίνακας εξόδου θα είναι ίδιων διαστάσεων με τον πίνακα εισόδου. Ο μαθηματικός τύπος είναι:

$$\text{Same Padding} = \frac{\text{Filter Size} - 1}{2}$$

Ακόμη, η σχέση που καθορίζει τις διαστάσεις των χαρτών χαρακτηριστικών είναι:

$$\text{Feature Map Size} = \frac{\text{Input Size} - \text{Filter Size} + 2 * \text{Padding}}{\text{Stride}} + 1$$



**Εικόνα 4.3.** Αναπαράσταση συνέλιξης με 4 φίλτρα 3x3 (2D φίλτρα καθώς η είσοδος είναι εικόνα αποχρώσεων του γκρι)

#### 4.2.2 Μη-Γραμμικά Επίπεδα (Non-Linear Layers)

Τα μη γραμμικά επίπεδα, γνωστά και ως επίπεδα ενεργοποίησης (activation layers), εισάγουν τη μη γραμμικότητα στην αρχιτεκτονική των συνελικτικών νευρωνικών δικτύων. Αυτά τα επίπεδα προσδίδουν την ικανότητα στο δίκτυο να μαθαίνει περίπλοκες, μη γραμμικές σχέσεις και μοτίβα στα δεδομένα καθώς αυτά «ρέουν» εντός του δικτύου. Τοποθετούνται συνήθως μετά τις γραμμικές λειτουργίες όπως η συνέλιξη και τα πλήρως συνδεδεμένα επίπεδα. Οι πιο κοινές μη γραμμικές συναρτήσεις ενεργοποίησης περιλαμβάνουν τη ReLU, τη σιγμοειδή και την υπερβολική εφαιπτομένη συνάρτηση (βλ. Κεφ. 3.4.2.1.)



**Εικόνα 4.4.** Παράδειγμα χρήσης της συνάρτησης ενεργοποίησης ReLU



#### 4.2.3. Συγκεντρωτικά Επίπεδα (Pooling Layers)

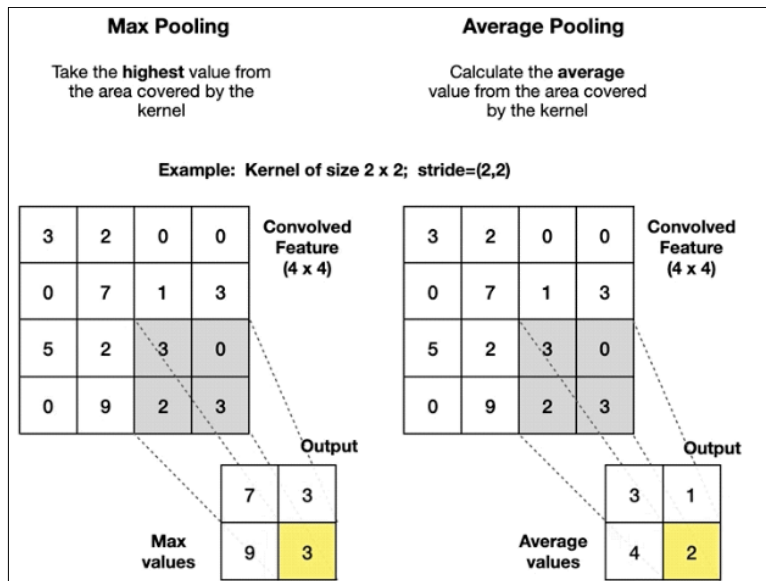
Η κύρια λειτουργία των συγκεντρωτικών επιπέδων είναι η υπο-δειγματοληψία (sub sampling) των δεδομένων και επομένως ο περιορισμός του αριθμού των απαιτούμενων πράξεων. Τοποθετούνται συνήθως μετά από ένα ή περισσότερα συνελκτικά επίπεδα και σκοπός τους είναι να συνοψίζουν τις εξόδους γειτονικών στοιχείων του πίνακα χαρακτηριστικών εντός ενός παραθύρου (patch) με μια αντιπροσωπευτική τιμή, ενώ συνήθως τα γειτονικά παράθυρα δεν επικαλύπτονται. Επιπλέον βοηθούν στη βελτίωση της γενίκευσης του μοντέλου και της πιθανότητας υπερ-προσαρμογής (over-fitting) μειώνοντας τον αριθμό των παραμέτρων. Κοινές κατηγορίες τεχνικών pooling είναι:

- **Συγκέντρωση Μεγίστου Στοιχείου (Max Pooling):** Διατηρείται σε κάθε περιοχή του παραθύρου πάνω στην εικόνα (ή τον χάρτη χαρακτηριστικών) μόνο η μέγιστη τιμή. Έτσι εντοπίζονται τα πιο σημαντικά χαρακτηριστικά σε κάθε περιοχή συγκέντρωσης.
- **Συγκέντρωση Μέσου Όρου (Average Pooling):** Η τεχνική αυτή εξυπηρετεί διαφορετικό σκοπό, παρέχοντας μία ομαλή αναπαράσταση του χάρτη χαρακτηριστικών, οδηγώντας στη μείωση του θορύβου και σε μια πιο γενικευμένη προβολή των δεδομένων. Υπολογίζεται για κάθε περιοχή συγκέντρωσης ο μέσος όρος των τιμών.

Ο μαθηματικός τύπος είναι για την υπολογισμό των διαστάσεων της εξόδου μετά από ένα συγκεντρωτικό επίπεδο είναι:

$$Output\ Size = \frac{Input\ Size}{Pooling\ Size}$$



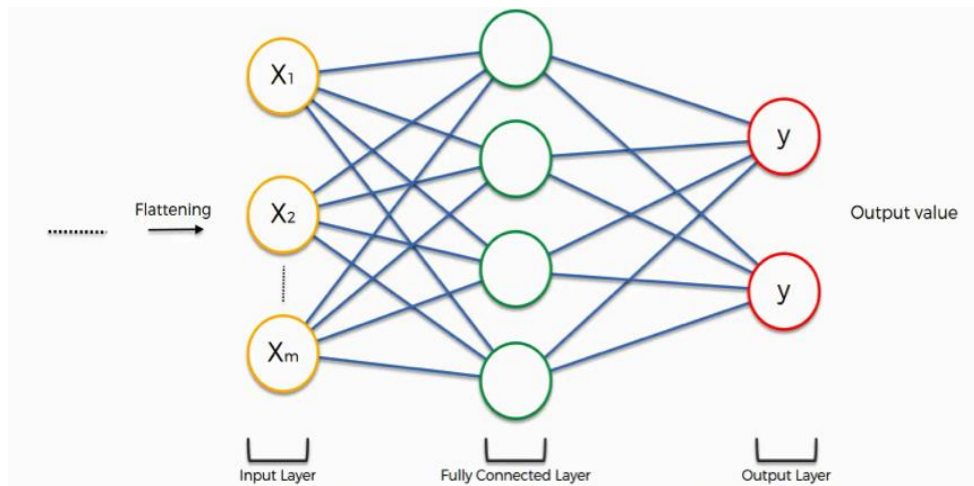


**Εικόνα 4.4.** Πραγματοποίηση Pooling σε 4x4 χάρτη χαρακτηριστικών με 2x2 patch

#### 4.2.4. Πλήρως Συνδεδεμένα Επίπεδα (Fully Connected Layers)

Στα πλήρως συνδεδεμένα επίπεδα, κάθε νευρώνας των κρυφών επιπέδων συνδέεται με την έξοδο όλων των νευρώνων του προηγούμενου επιπέδου. Τα επίπεδα αυτά αποτελούν σημαντικό μέρος της αρχιτεκτονικής, ακολουθούν τα συνελκτικά και συγκεντρωτικά επίπεδα, και είναι υπεύθυνα για την επεξεργασία και συγκέντρωση των χαρακτηριστικών που εξάγονται από τα προηγούμενα επίπεδα. Πριν τα δεδομένα εισέλθουν σε αυτά τα επίπεδα, συνήθως πρέπει να υποστούν ένα ενδιάμεσο βήμα που ονομάζεται "flattening" (επίπεδη μετατροπή). Αυτή η διαδικασία μετατρέπει τα πολυδιάστατα χαρακτηριστικά σε έναν μεγάλο πίνακα ή διάνυσμα, προετοιμάζοντάς τα για επεξεργασία από τα πλήρως συνδεδεμένα επίπεδα.

Τα πλήρως συνδεδεμένα επίπεδα χρησιμοποιούν συναρτήσεις ενεργοποίησης για την εκτέλεση μη γραμμικών μετασχηματισμών στα χαρακτηριστικά, επιτρέποντας στο δίκτυο να αντιλαμβάνεται πολύπλοκα μοτίβα και να μοντελοποιεί σύνθετες αντιστοιχίες μεταξύ των εισόδων και των εξόδων.



**Εικόνα 4.5.** Πλήρως συνδεδεμένα επίπεδα

### 4.3. Τεχνικές Βελτίωσης της απόδοσης των ΤΝΔ

Για να αξιοποιηθεί πλήρως το δυναμικό των ΤΝΔ είναι απαραίτητο να αντιμετωπιστούν προκλήσεις που σχετίζονται με τη σταθερότητα της εκπαίδευσης, την υπερ-προσαρμογή/υπο-προσαρμογή, την ταχύτητα σύγκλισης κ.α. Στην αντιμετώπιση τέτοιων προβλημάτων μπορούν να δώσουν λύση ορισμένες τεχνικές. Οι τεχνικές αυτές χρησιμοποιούνται για την πρόληψη της υπερ-προσαρμογής και εισάγουν τροποποιήσεις κατά τη διάρκεια της εκπαίδευσης για να διασφαλίσουν ότι το δίκτυο δεν θα γίνει πολύ περίπλοκο, με σκοπό την βελτίωση της ικανότητάς του να κάνει προβλέψεις σε άγνωστα δεδομένα (generalization). Διασφαλίζουν ακόμη ότι τα δεδομένα εισόδου ή οι ενεργοποιήσεις των νευρώνων κλιμακώνονται ή κεντράρονται με τρόπο που σταθεροποιεί την εκπαίδευση και διευκολύνει τη ταχύτερη σύγκλιση. Παρακάτω αναφέρονται ορισμένες γνωστές τεχνικές:

#### 4.3.1. L1 & L2 Κανονικοποίηση (L1 & L2 Regularization)

Η **L1 κανονικοποίηση**, επίσης γνωστή ως κανονικοποίηση Lasso (Last Absolute Shrinkage and Selection Operator), προσθέτει έναν όρο «ποινής» στην συνάρτηση σφάλματος που είναι ανάλογος με τις απόλυτες τιμές των συναπτικών βαρών του δικτύου. Ο στόχος είναι ορισμένα βάρη να γίνουν ακριβώς μηδέν, ώστε να επιλεγεί στην ουσία ένα υποσύνολο των σημαντικότερων χαρακτηριστικών (ή νευρώνων). Πιο συγκεκριμένα έχουμε:

$$\text{Κόστος ή Απώλεια} = \text{Αρχικό κόστος} + \lambda \sum_{i=1}^n |w_i|$$

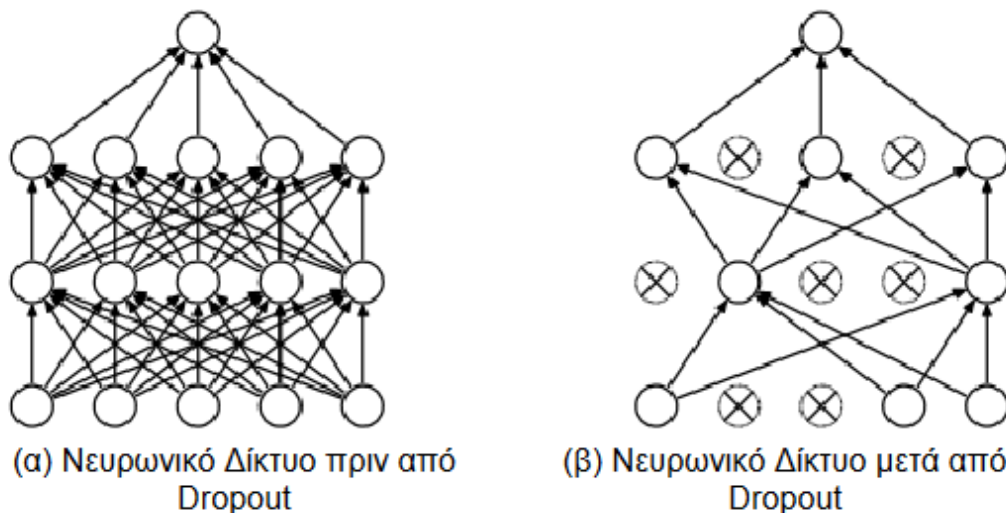
όπου  $n$  ο συνολικός αριθμός βαρών και  $\lambda$  η υπερπαράμετρος κανονικοποίησης

Η **L2 κανονικοποίηση**, επίσης γνωστή ως κανονικοποίηση Ridge, έχει παρόμοια λειτουργία με την L1 κανονικοποίηση με τη διαφορά ότι στον όρο «ποινής» περιλαμβάνεται το άθροισμα του τετραγώνου των βαρών. Σκοπός είναι μερικά από τα βάρη να γίνουν μικρότερα αλλά όχι ακριβώς μηδέν. Πιο συγκεκριμένα:

$$\text{Κόστος ή Απώλεια} = \text{Αρχικό κόστος} + \lambda \sum_{i=1}^n w_i^2$$

#### 4.3.2. Απενεργοποίηση νευρώνων (Dropout)

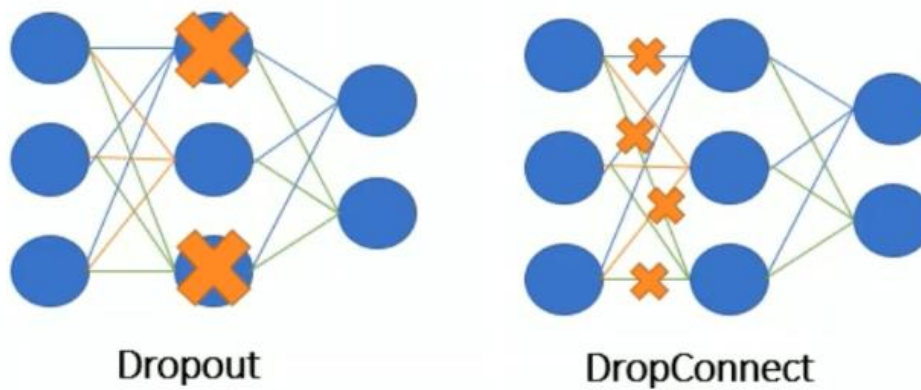
Η τεχνική αυτή λειτουργεί απενεργοποιώντας τυχαία ένα ποσοστό (dropout rate) των νευρώνων σε κάθε επίπεδο σε κάθε επανάληψη κατά την εκπαίδευση. Αυτή η τυχαία απενεργοποίηση εισάγει μεταβλητότητα, εμποδίζοντας το δίκτυο να βασίζεται πολύ σε συγκεκριμένους νευρώνες ή χαρακτηριστικά. Το dropout συμβάλλει στη βελτίωση της ικανότητας του δικτύου να γενικεύει και να αποτρέπει την υπερ-προσαρμογή. Κατά τον έλεγχο (testing), το dropout απενεργοποιείται και οι προβλέψεις γίνονται χρησιμοποιώντας το πλήρες δίκτυο, δίνοντας ένα αποτέλεσμα αντιπροσωπευτικό του μέσου όρου των προβλέψεων που θα έδιναν τα υποδίκτυα που δημιουργήθηκαν κατά τη διάρκεια της εκπαίδευσης.



**Εικόνα 4.6.** Εφαρμογή dropout και αναπαράσταση ενός υποδικτύου

#### 4.3.3. Απενεργοποίηση συνδέσεων νευρώνων (DropConnect)

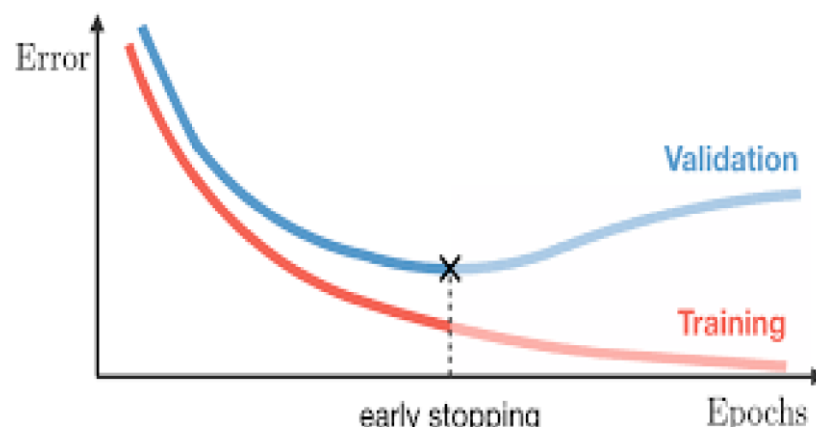
Λειτουργεί με λογική παρόμοια με της τεχνικής Dropout, αλλά αντί να απενεργοποιεί τυχαία ολόκληρους νευρώνες, απενεργοποιεί τυχαία μεμονωμένες συνδέσεις μεταξύ των νευρώνων. Δηλαδή ένα κλάσμα των συνδέσεων (συναπτικά βάρη) μεταξύ των νευρώνων μηδενίζεται τυχαία σε κάθε επανάληψη της εκπαίδευσης.



**Εικόνα 4.7.** Αναπαράσταση διαφοράς τεχνικών Dropout και DropConnect σε ΤΝΔ

#### 4.3.4. Πρόωρη Διακοπή (Early Stopping)

Αποτελεί μία ακόμη τεχνική που εφαρμόζεται για να αποφευχθεί η υπερ-προσαρμογή του δικτύου κατά τη διαδικασία της εκπαίδευσης. Αυτό γίνεται με την παρακολούθηση των παραμέτρων του δικτύου (validation, training, testing loss) ώστε να ελέγχεται σε κάθε εποχή η πρόοδός τους. Το σημαντικό είναι να βρεθεί το σημείο στο οποίο αρχίζουν ορισμένες παράμετροι του δικτύου να χειροτερεύουν συνεχώς με το πέρασμα κάποιου επιλεγμένου αριθμού εποχών (early stopping patience), οπότε και θα πρέπει να σταματήσει η εκπαίδευση. Επίσης μπορεί να αποθηκεύεται η τρέχουσα κατάσταση των βαρών του δικτύου σε κάθε εποχή ώστε να κρατηθούν οι καλύτερες τιμές με το πέραν των εποχών.

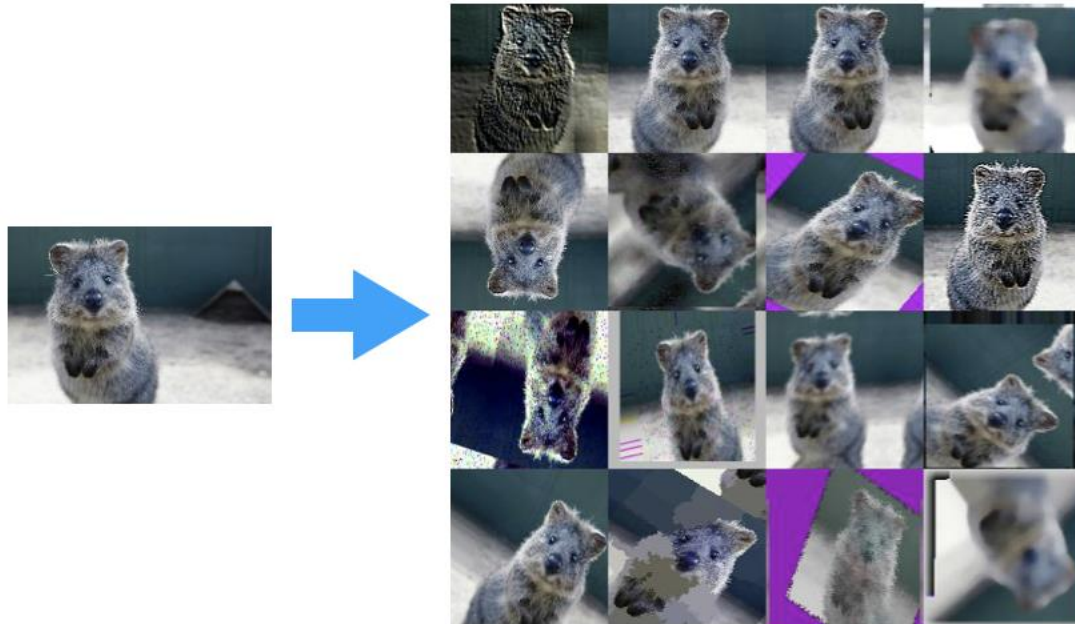


**Εικόνα 4.8.** Πρόωρη διακοπή εκπαίδευσης λόγω χειροτέρευσης του Validation Loss

#### 4.3.5. Επαύξηση Δεδομένων (Data Augmentation)

Η επαύξηση των δεδομένων εισόδου αποτελεί ισχυρή τεχνική βελτίωσης της γενίκευσης των μοντέλων αναγνώρισης εικόνας και μετριάζει την υπερ-προσαρμογή. Εφαρμόζοντας διάφορους μετασχηματισμούς σε υπάρχουσες εικόνες, όπως περιστροφή, κλιμάκωση, αναστροφή ή αλλαγή φωτεινότητας και αντίθεσης, η αύξηση αυτή των δεδομένων διαφοροποιεί το σύνολο των δεδομένων εκπαίδευσης.

Για παράδειγμα, στις εργασίες ανίχνευσης αντικειμένων, η αύξηση των εικόνων εισάγοντας μικρές παραλλαγές στις συνθήκες φωτισμού μπορεί να βοηθήσει το μοντέλο να προσαρμοστεί καλύτερα σε διαφορετικές συνθήκες φωτισμού για νέα άγνωστα δεδομένα.



**Εικόνα 4.9.** Επαύξηση δεδομένων με εφαρμογή διάφορων μετασχηματισμών

#### 4.3.6. Κανονικοποίηση Δεσμίδας (Batch Normalization)

Ένα από τα προβλήματα που αντιμετωπίζονται κατά την εκπαίδευση των ΤΝΔ, είναι η συνεχής αλλαγή της εισόδου των νευρώνων στα κρυφά επίπεδα και κατ' επέκταση των κατανομών αυτών των εισόδων. Αυτή η αλλαγή ονομάζεται **εσωτερική συνδιακυμαίνουσα μετατόπιση (internal covariate shift)** και είναι ένας από τους λόγους που κατά την εκπαίδευση ενός δικτύου είναι απαραίτητη η μείωση του ρυθμού μάθησης ώστε να συνεχίσει να μειώνεται το σφάλμα. Ο προτεινόμενος μηχανισμός για την αντιμετώπιση αυτού του προβλήματος είναι η κανονικοποίηση δεσμίδας και υλοποιείται υπολογίζοντας συνεχώς τις κατανομές όλων των ενεργοποιήσεων για κάθε επιλεγμένη δεσμίδα δειγμάτων εκπαίδευσης. Καθώς υπολογίζονται οι ενεργοποιήσεις και κατά την εμπροσθοδιάδοση των ενεργοποιήσεων, γίνεται κανονικοποίηση βάσει των στατιστικών της δεσμίδας για το κάθε επίπεδο. Οι εξισώσεις που αφορούν αυτή την τεχνική είναι:

$$\text{Μέσος όρος παρτίδας : } \mu_B = \frac{1}{m} \sum_{i=1}^m x_i$$

$$\text{Διακύμανση παρτίδας } \sigma_B^2 = \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$$

$$\text{Κανονικοποιημένη ενεργοποίηση } \hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

$$\text{Αλλαγή κλίμακας και μετατόπιση ενεργοποίησης } y_i = \gamma \hat{x}_i + \beta$$

Όπου  $\gamma$  η παράμετρος κλίμακας (scaling) και  $\beta$  η παράμετρος μετατόπισης (shifting) τις οποίες καλείται να «μάθει» το δίκτυο (**learnable parameters**) και να προσαρμόσει τις κανονικοποιημένες ενεργοποιήσεις στην επιθυμητή κλίμακα και περιοχή.

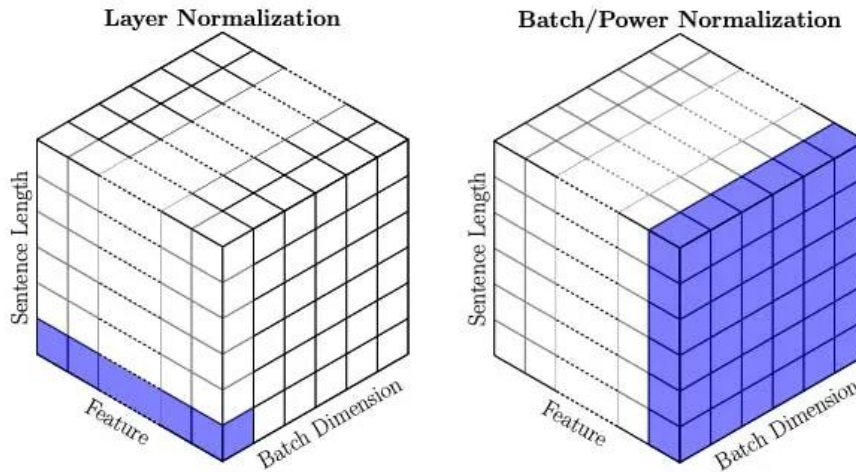
Η κανονικοποίηση δεσμίδας εφαρμόζεται συνήθως στις ενεργοποιήσεις ενός επιπέδου ακριβώς πριν από την εφαρμογή της συνάρτησης ενεργοποίησης (π.χ., ReLU). Βοηθά στο μετριασμό ζητημάτων όπως η εξαφάνιση των κλίσεων, βοηθάει στην ταχύτερη σύγκλιση (παρόλο που το υπολογιστικό κόστος κάθε εποχής αυξάνεται) και μειώνει την ευαισθησία του δικτύου στην αρχικοποίηση των συναπτικών βαρών. Χρησιμοποιείται συνήθως σε εφαρμογές με συνελκτικά νευρωνικά δίκτυα και σε βαθιά δίκτυα πρόσθιας τροφοδότησης.

#### 4.3.7. Κανονικοποίηση Επιπέδου (Layer Normalization)

Η κανονικοποίηση επιπέδου είναι μια τεχνική που χρησιμοποιείται στα ΤΝΔ για την κανονικοποίηση των ενεργοποιήσεων ενός επιπέδου **ανά δείγμα**, διασφαλίζοντας ότι οι ενεργοποιήσεις έχουν σταθερές στατιστικές ιδιότητες σε κάθε μεμονωμένο δείγμα εκπαίδευσης. Τα στατιστικά στοιχεία (μέσος όρος και διακύμανση) υπολογίζονται σε όλους τους νευρώνες εντός του επιπέδου για το κάθε δείγμα ανεξάρτητα από τα υπόλοιπα δείγματα.

Σε περιπτώσεις όπου έχουμε διαδοχικά δεδομένα (sequential data) τα οποία εξαρτώνται το ένα από το άλλο, ή όταν έχουμε μικρό όγκο δεδομένων, η χρήση του batch normalization ίσως δημιουργήσει ακόμη παραπάνω θόρυβο και τα αποτελέσματα να μην είναι ικανοποιητικά. Προτείνεται σε αυτές τις περιπτώσεις η χρήση της κανονικοποίησης επιπέδου, η οποία θεωρείται η πλέον κατάλληλη μέθοδος για εφαρμογές με αναδρομικά νευρωνικά δίκτυα (recurrent neural networks) και νευρωνικά δίκτυα μετασχηματισμών (transformers).





**Εικόνα 4.10.** Κανονικοποίηση επιπέδου και δεσμίδας. Τα μπλε στοιχεία σε κάθε τεχνική χρησιμοποιούνται για τον υπολογισμό των στατιστικών στοιχείων

#### 4.4. Μετρικές αξιολόγησης της απόδοσης των ΣΝΔ

Οι μέθοδοι μέτρησης αξιολόγησης (evaluation metrics) είναι απαραίτητα εργαλεία στον τομέα της μηχανικής μάθησης και της βαθιάς μάθησης, που μας επιτρέπουν να αξιολογούμε ποσοτικά την απόδοση των μοντέλων μας σε διάφορες εργασίες, από την ταξινόμηση και την παλινδρόμηση έως την επεξεργασία φυσικής γλώσσας και την όραση υπολογιστών (ανάλυση των οπτικών δεδομένων). Αυτά χρησιμεύουν ως αντικειμενική μέθοδος μέτρησης για το πόσο καλά ένα μοντέλο γενικεύει τη μάθησή του από τα δεδομένα της εκπαίδευσης σε άγνωστα δεδομένα του πραγματικού κόσμου. Η επιλογή του σωστού συνόλου μεθόδων μέτρησης αξιολόγησης είναι πρωταρχικής σημασίας, καθώς εξαρτάται από το εκάστοτε πρόβλημα, τη φύση των δεδομένων και τους στόχους της εργασίας μας. Στο σύνολό τους και με τον κατάλληλο συνδυασμό τους μας παρέχουν πολύτιμες πληροφορίες για τα δυνατά και αδύναμα σημεία του μοντέλου μας, καθοδηγώντας τελικά τις βελτιώσεις και τη λήψη κατάλληλων αποφάσεων. Μερικές μέθοδοι μέτρησης της αξιολόγησης αναλύονται παρακάτω:

- **Ορθότητα (Accuracy):** Είναι μια από τις πιο συχνά χρησιμοποιούμενες μέθοδοι μέτρησης αξιολόγησης για εργασίες ταξινόμησης πολλών τάξεων. Μετρά την αναλογία των σωστά ταξινομημένων δειγμάτων (instances) μεταξύ όλων των δειγμάτων στο σύνολο δεδομένων. Σε μια εφαρμογή πολλών κλάσεων όπως η δική μας, η ορθότητα παρέχει μια γενική επισκόπηση του πόσο καλά αποδίδει το μοντέλο σε όλες τις κατηγορίες.

$$Accuracy = \frac{\text{Number of Correctly Classified Instances}}{\text{Total Number of Instances}}$$

Τα θετικό της ορθότητας (accuracy) είναι ότι αποτελεί μια απλή μέθοδο μέτρησης για αξιολόγηση της γενικής (overall) απόδοσης του μοντέλου, και είναι εύκολη στην ερμηνεία, καθώς μια αυξημένη τιμή της ορθότητας μεταφράζεται και σε καλύτερη απόδοση. Τα αρνητικά της είναι ότι δεν θεωρείται κατάλληλη σε περίπτωση που υπάρχουν διαφορές την ποσότητα των δειγμάτων των κλάσεων (class imbalances) στο σύνολο των δεδομένων, καθώς και ότι δεν αναγνωρίζει τα διαφορετικά είδη λαθών, δηλαδή η λανθασμένη ταξινόμηση μίας εικόνας σε κάποια κλάση ίσως πολύ κοντινή με τη σωστή, αναγνωρίζεται απλά ως λανθασμένη. Για μία πιο λεπτομερή αξιολόγηση ενός μοντέλου, η ορθότητα από μόνη της δεν επαρκεί.

- **Ακρίβεια (Precision):** Είναι ακόμα μία βασική μέτρηση στην ταξινόμηση πολλαπλών κατηγοριών που μετρά την ικανότητα του μοντέλου να κάνει ακριβείς θετικές προβλέψεις για μια συγκεκριμένη κατηγορία. Με άλλα λόγια, απαντά στην ερώτηση: "Από όλες τις περιπτώσεις που το μοντέλο προέβλεψε ότι ανήκαν σε μια συγκεκριμένη κατηγορία, πόσες ήταν σωστές;" Για μια δεδομένη κατηγορία, η ακρίβεια υπολογίζεται χρησιμοποιώντας τον τύπο:

$$Precision = \frac{True\ Positives\ for\ the\ Class}{True\ Positives\ for\ the\ Class + False\ Positives\ for\ the\ Class}$$

Βασικό πλεονέκτημα της ακρίβειας είναι ότι εστιάζει στις θετικές (positive) προβλέψεις του μοντέλου, κάνοντας το χρήσιμο όταν η ελαχιστοποίηση των false positives προβλέψεων (το να κατατάξει ένα δείγμα σε μία κλάση, ενώ αυτό δεν ισχύει) είναι σημαντική. Παραδείγματα τέτοια αποτελούν ο εντοπισμός απάτης (fraud detection) και η ιατρική διάγνωση από εικόνες (medical diagnosis). Επίσης παρέχει πολύτιμες πληροφορίες για την απόδοση του μοντέλου για μεμονωμένες κατηγορίες εντοπίζοντας κατηγορίες με υψηλή ακρίβεια πρόβλεψης και κατηγορίες που χρειάζονται βελτίωση.

Ωστόσο, για να αξιολογηθεί η συνολική απόδοση του μοντέλου, μπορούν να χρησιμοποιηθούν μετρικά όπως το micro-averaged, macro-averaged και weighted precision. Αναλυτικότερα,

1. **Micro-Averaged Precision:** Υπολογίζει την ακρίβεια αθροίζοντας τα αληθινώς θετικά (true positives) και τα ψευδώς θετικά (false positives) για όλες τις κατηγορίες. Παρέχει ίση βαρύτητα σε κάθε δείγμα, όχι σε κάθε κατηγορία.
2. **Macro-Averaged Precision:** Υπολογίζει την ακρίβεια για κάθε κατηγορία ανεξάρτητα και στη συνέχεια παίρνει τον μέσο όρο, δίνοντας ίση βαρύτητα σε κάθε κατηγορία.
3. **Weighted Precision:** Παρόμοια με το macro-averaged precision, αλλά λαμβάνει υπόψη τις ανισορροπίες μεταξύ κατηγοριών. Υπολογίζει την ακρίβεια για κάθε



κλάση και υπολογίζει τον σταθμισμένο μέσο όρο, δίνοντας μεγαλύτερη βαρύτητα σε κατηγορίες με περισσότερα δείγματα.

- **Ανάκληση (Recall):** Μετράει την ικανότητα του μοντέλου να προσδιορίζει σωστά όλες τις πραγματικά θετικές περιπτώσεις για μια συγκεκριμένη κατηγορία. Απαντά στην ερώτηση: "Από όλες τις περιπτώσεις που ανήκουν πραγματικά σε μια συγκεκριμένη κατηγορία, πόσες ταξινομήσε σωστά ως τέτοιες το μοντέλο;" Για μια δεδομένη κατηγορία, η ανάκληση υπολογίζεται χρησιμοποιώντας τον τύπο:

$$Recall = \frac{True\ Positives\ for\ the\ Class}{True\ Positives\ for\ the\ Class + False\ Negatives\ for\ the\ Class}$$

Η ανάκληση είναι χρήσιμο μετρικό όταν θέλουμε να ελαχιστοποιήσουμε τα λανθασμένα αρνητικά (false negatives), δηλαδή όταν το μοντέλο κατηγοριοποιεί ως «όχι» ένα δείγμα ενώ θα έπρεπε να είναι «ναι», ή σε περίπτωση πολλών κλάσεων όταν προβλέπει ότι ένα δείγμα δεν ανήκει σε μία συγκεκριμένη κλάση, ενώ στην πραγματικότητα ανήκει. Παράδειγμα αποτελεί ο μη εντοπισμός μίας αρνητικής κατάστασης από την αξονική τομογραφία ενός ασθενή, ενώ αυτή υπάρχει. Αυτό το σφάλμα μπορεί να επιφέρει τρομερά αρνητικές συνέπειες καθώς απέτυχε να εντοπίσει σωστά μία θετική περίπτωση.

Ο υπολογισμός της «συνολικής» ανάκλησης/ικανότητας του μοντέλου να εντοπίζει επιτυχώς τις θετικές περιπτώσεις, μπορεί αντιστοίχως με το precision να γίνει με χρήση των micro-averaged, macro-averaged, weighted recall.

- **F1 – Score:** Το F1-Score συνδυάζει την ακρίβεια (precision) και την ανάκληση για να παρέχει μια ισορροπημένη μέτρηση της απόδοσης ενός μοντέλου για την ταξινόμηση μεταξύ κατηγοριών. Η ακρίβεια μετρά την ικανότητα του μοντέλου να κάνει ακριβείς θετικές προβλέψεις, ενώ η ανάκληση μετρά την ικανότητα του μοντέλου να προσδιορίζει όλες τις πραγματικές θετικές περιπτώσεις. Το F1-Score λαμβάνει υπόψη και τις δύο αυτές τιμές υπολογίζοντας τον αρμονικό μέσο όρο τους (harmonic mean), ενθαρρύνοντας το μοντέλο να εξισορροπήσει την αντιστάθμιση μεταξύ της πραγματοποίησης ακριβών θετικών προβλέψεων και του εντοπισμού όλων των πραγματικών θετικών περιπτώσεων. Υπολογίζεται από τον τύπο:

$$F1\ Score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = 2 \frac{Precision * Recall}{Precision + Recall}$$

Ουσιαστικά, το F1-Score είναι χρήσιμο όταν θέλουμε να διασφαλίσουμε ότι το μοντέλο δεν εστιάζει πολύ σε ένα από τις δύο αυτές μεθόδους μέτρησης, αλλά βρίσκει μια μέση λύση όπου και οι δύο πτυχές είναι ισορροπημένες. Αυτό μπορεί

να είναι σημαντικό σε σενάρια όπου θέλουμε να ελαχιστοποιήσετε τόσο τα ψευδώς θετικά όσο και τα ψευδώς αρνητικά χωρίς να προτιμάται το ένα έναντι του άλλου. Το F1-Score συνολικά για ένα μοντέλο υπολογίζεται ομοίως ως micro-averaged, macro-averaged, weighted F1-Score, όπως στην περίπτωση ταξινόμησης μεταξύ πολλαπλών κλάσεων.

## 5. Πειραματική διαδικασία

### 5.1. Εισαγωγή

Το κεφάλαιο αυτό αναφέρεται στην πειραματική διαδικασία που ακολουθήθηκε στα πλαίσια αυτής της διπλωματικής εργασίας. Αυτό περιλαμβάνει την παρουσίαση των ταξινομητών που δοκιμάστηκαν, το σύνολο δεδομένων που χρησιμοποιήθηκε και τους πίνακες και τα διαγράμματα που προέκυψαν.

Οι διαδικασίες ταξινόμησης που δοκιμάστηκαν αναφέρονται σε τεχνικές βαθιών νευρωνικών δικτύων και συγκεκριμένα στα Συνελικτικά Νευρωνικά Δίκτυα (ΣΝΔ). Σκοπός είναι η δημιουργία, εκπαίδευση, αξιολόγηση και αποθήκευση του καλύτερα εκπαιδευμένου μοντέλου, το οποίο θα μπορεί να αναγνωρίζει το είδος του ορυκτού από νέες, άγνωστες σε αυτό εικόνες. Ο ταξινομητής θα πρέπει να ανταποκρίνεται μετά την εκπαίδευση του στην κατηγοριοποίηση πολλαπλών κλάσεων, και κάθε μία κλάση αποτελεί και ένα διαφορετικό ορυκτό.

Πιο συγκεκριμένα, δημιουργήθηκε και εκπαιδεύτηκε ένα ΣΝΔ από το μηδέν με επιλεγμένα χαρακτηριστικά, και ένα ακόμη προ-εκπαιδευμένο μοντέλο VGG16 (Visual Geometry Group 16) ώστε να συγκριθούν τα αποτελέσματά τους.

Για την δημιουργία του απαραίτητου λογισμικού χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python (έκδοση 3.11.5), η πλατφόρμα Anaconda για τη διευκόλυνση της οργανωμένης εκτέλεσης μερών κώδικα και της δημιουργίας απεικονίσεων με χρήση των Jupyter Notebooks, η ανοιχτού κώδικα βιβλιοθήκης PyTorch (έκδοση 2.1.0) και ένα ακόμη απαραίτητο εργαλείο, το CUDA (Compute Unified Device Architecture, έκδοση 12.1), το οποίο σε συνεργασία με το PyTorch επιτρέπει την επιτάχυνση των υπολογισμών, μέσω παραλληλισμού με χρήση της κάρτας γραφικών (GPU), για την εκπαίδευση των μοντέλων.

### 5.2. Σύνολο Δεδομένων

Στην παρούσα εργασία χρησιμοποιήθηκαν εικόνες που συλλέχθηκαν από ιστοσελίδες του διαδικτύου ανοιχτές στο κοινό, και πιο συγκεκριμένα από τις [www.kaggle.com](http://www.kaggle.com) και [www.mindat.org](http://www.mindat.org). Οι εικόνες αυτές απεικονίζουν 10 διαφορετικά ορυκτά με αρκετά δείγματα για το κάθε ένα.

Ορυκτό	Αριθμός εικόνων
<i>Azurite – Αζουρίτης</i>	776

<i>Biotite – Βιοτίτης</i>	656
<i>Calcite – Ασβεστίτης</i>	514
<i>Cinnabar – Κινναβαρίτης</i>	403
<i>Hematite – Αιματίτης</i>	546
<i>Malachite – Μαλαχίτης</i>	422
<i>Orpiment – Κίτρινη Σανδαράχη</i>	489
<i>Quartz - Χαλαζίας</i>	430
<i>Sulphur - Θειάφι</i>	529
<i>Turquoise – Τουρκουάζ</i>	505

**Πίνακας 5.1.** Αριθμός εικόνων που συλλέχθηκαν για κάθε ορυκτό

### 5.3. Δημιουργία Απαραίτητων Υπο-συνόλων

Αρχικά πρέπει να γίνει εισαγωγή όλων των βιβλιοθηκών, οι οποίες θα μας παρέχουν τις απαραίτητες συναρτήσεις.

```

import numpy as np

import os
import cv2

import torch
import torchvision
import torch.nn as nn
import torch.optim as optim
import torchvision.transforms as transforms
from torchvision.datasets import ImageFolder
from torch.utils.data import DataLoader

```

Έπειτα θέτουμε τον μετατροπέα (transformer), από τον οποίο θα περάσει κάθε εικόνα πριν εισέλθει στο δίκτυο (εντός κάποιου υπο-συνόλου). Στην ουσία γίνεται επαύξηση των δεδομένων (data augmentation).

```

transformer = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.RandomHorizontalFlip(), # flip horizontally
    transforms.RandomRotation(degrees=10), # rotate the image by up to 10 degrees
    transforms.ColorJitter(brightness=0.2, contrast=0.2), # adjust brightness, contrast
    transforms.ToTensor(), # change data type from numpy to tensor
    transforms.Normalize(mean=[0.485, 0.456, 0.406], std=[0.229, 0.224, 0.225])
])

```

Ο μετασχηματιστής όπως βλέπουμε θα πάρει κάθε εικόνα και θα εκτελέσει ορισμένες μετατροπές ή αλλαγές πριν αυτές τροφοδοτηθούν στο δίκτυο. Όλες θα αποκτήσουν το ίδιο μέγεθος σε pixels (224x224), κάποιες θα αναστραφούν οριζόντια, θα περιστραφούν τυχαία μέχρι 10 μοίρες, θα γίνει αλλαγή της φωτεινότητας και της αντίθεσης. Επίσης γίνεται κανονικοποίηση (normalization) ώστε τα τρία χρώματα κάθε εικόνας να ακολουθούν μία συγκεκριμένη κατανομή με μέση τιμή και διασπορά. Αυτό μπορεί να οδηγήσει σε ταχύτερη σύγκλιση και καλύτερη γενίκευση του μοντέλου.

Με το παρακάτω τμήμα κώδικα επιβεβαιώνουμε ότι η κάρτα γραφικών μας είναι συμβατή με το CUDA και θέτουμε τη μεταβλητή device.

```

# Check if a CUDA-compatible GPU is available
device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
device

device(type='cuda')

```

Ορίζουμε το path για το φάκελο όπου περιέχονται οι 10 υπο-φάκελοι, με τις εικόνες του κάθε ορυκτού στον φάκελο με το όνομά του.

```
# Define dataset path
dataset_path = "images"

dataset = ImageFolder(root=dataset_path, transform=transformer)
dataset.classes

['Azurite',
 'Biotite',
 'Calcite',
 'Cinnabar',
 'Hematite',
 'Malachite',
 'Orpiment',
 'Quartz',
 'Sulphur',
 'Turquoise']
```

Επιβεβαιώνουμε ότι έχουν διαβαστεί όλες οι εικόνες των ορυκτών **(συνολικά 5270)**:

```
from collections import Counter
class_counts = Counter(dataset.targets)
class_counts
```

```
Counter({0: 776,
         1: 656,
         4: 546,
         8: 529,
         2: 514,
         9: 505,
         6: 489,
         7: 430,
         5: 422,
         3: 403})
```

Με χρήση του πακέτου scikit-learn για το διαχωρισμό και της κλάσης DataLoader της PyTorch δημιουργούμε τρία υπο-σύνολα με τυχαία δείγματα από το σύνολο των δεδομένων μας. Το training set θα έχει πλήθος δειγμάτων το 60% του αρχικού συνόλου, το validation set 20% του συνόλου και το testing set 20% του συνόλου.

## Split dataset to training, validation, testing sets

```
from sklearn.model_selection import train_test_split

train_size = 0.6 # 60% for training
combined_valid_test_size = 0.4 # 40% for validation + testing

train_data, valid_test_data = train_test_split(dataset, test_size=combined_valid_test_size, random_state=42)

# Split the validation-testing dataset into validation and testing sets
valid_size = 0.5 # 50% of the validation-testing data for validation
test_size = 0.5 # 50% of the validation-testing data for testing

valid_data, test_data = train_test_split(valid_test_data, test_size=test_size, random_state=42)

# Create DataLoaders for training and test sets
batch_size = 80
train_dataloader = DataLoader(train_data, batch_size=batch_size, shuffle=True)
valid_dataloader = DataLoader(valid_data, batch_size=batch_size, shuffle=True)
test_dataloader = DataLoader(test_data, batch_size=batch_size, shuffle=True)
```

Κάθε dataloader που δημιουργείται αποτελείται από δεσμίδες (batches) με τυχαία επιλεγμένα δείγματα από το κάθε υπο-σύνολο. Ο αριθμός του batch\_size είναι ενδεικτικός και θα μεταβάλλεται στη συνέχεια κατά τις δοκιμές.

### 5.4. Μοντέλο 1<sup>ο</sup> : Προσαρμοσμένη αρχιτεκτονική ΣΝΔ

Η αρχιτεκτονική που αρχικά χρησιμοποιήθηκε δεν αποτελεί κάποιο έτοιμο ή γνωστό μοντέλο συνελκτικού νευρωνικού δικτύου, αλλά ένα τυπικό (custom) CNN δίκτυο. Κάθε επίπεδο και τα χαρακτηριστικά αυτών στο δίκτυο επιλέχθηκαν χάριν πειραματισμού και εξερεύνησης της απόδοσης του στα δεδομένα των εικόνων που συλλέχθηκαν.

## Custom Model

```
class ConvNet(nn.Module):
    def __init__(self, num_classes=len(dataset.classes)):
        super(ConvNet, self).__init__()
        self.conv1 = nn.Conv2d(in_channels=3, out_channels=32, kernel_size=3, padding=1)
        self.bn1 = nn.BatchNorm2d(32)

        self.conv2 = nn.Conv2d(in_channels=32, out_channels=64, kernel_size=3, padding=1)
        self.bn2 = nn.BatchNorm2d(64)

        self.fc1 = nn.Linear(64 * 56 * 56, 128)
        self.dropout1 = nn.Dropout(0.5) # dropout rate 50%

        self.fc2 = nn.Linear(128, 64)
        self.dropout2 = nn.Dropout(0.3) # dropout rate 30%

        self.fc3 = nn.Linear(64, num_classes)

    def forward(self, x):
        x = self.conv1(x)
        x = self.bn1(x)
        x = nn.functional.relu(x)
        x = nn.functional.max_pool2d(x, 2)

        x = self.conv2(x)
        x = self.bn2(x)
        x = nn.functional.relu(x)
        x = nn.functional.max_pool2d(x, 2)

        x = x.view(x.size(0), -1)

        x = self.fc1(x)
        x = nn.functional.relu(x)
        x = self.dropout1(x)

        x = self.fc2(x)
        x = nn.functional.relu(x)
        x = self.dropout2(x)

        x = self.fc3(x)
        x = nn.functional.softmax(x, dim=1)

        return x

model = ConvNet(num_classes=len(dataset.classes)).to(device)
```

Πρώτα πραγματοποιείται αρχική απόδοση της κλάσης ConvNet ώστε να γίνει ανάθεση αργότερα στη μεταβλητή model. Το δίκτυο μας αποτελείται από **2 συνελικτικά επίπεδα (self.conv1 και self.conv2)**. Μετά από κάθε συνελικτικό επίπεδο παρεμβάλλεται και ένα **επίπεδο κανονικοποίησης δεσμίδας (batch normalization: self.bn1 και self.bn2)**, ένα **μη-γραμμικό επίπεδο με συνάρτηση ενεργοποίησης την ReLU (2 \* nn.functional.relu(x))** και ένα **συγκεντρωτικό επίπεδο (max-pooling)** με μέγεθος patch 2x2. Στην συνέχεια ακολουθεί η **επίπεδη μετατροπή (flattening)** του x (x.view(x.size(0),-1)) και έπονται **πλήρως συνδεδεμένα επίπεδα (self.fc1 και self.fc2)**, τα οποία ακολουθούνται επίσης από τη συνάρτηση ενεργοποίησης ReLU και από την τεχνική dropout με πιθανότητα 50% και 30% αντίστοιχα για κάθε μία. Τέλος, ακολουθεί το τρίτο και τελευταίο **πλήρως συνδεδεμένο επίπεδο (self.fc3)** με συνάρτηση ενεργοποίησης την softmax.

Αναλυτικότερα, το πρώτο συνελικτικό δίκτυο δέχεται εικόνες με 3 channels, δηλαδή εικόνες RGB. Εφαρμόζει 32 διαφορετικά φίλτρα (kernels) και εξάγει 32 feature maps. Κάθε φίλτρο από αυτά έχει μέγεθος 3x3 (kernel\_size = 3). Σε κάθε συνέλιξη φίλτρου και input-εικόνας εφαρμόζεται επίσης padding. Στη συνέχεια γίνεται η πρώτη κανονικοποίηση δεσμίδας.



Το δεύτερο συνελκτικό επίπεδο δέχεται 32 feature maps (τα outputs του πρώτου συνελκτικού επιπέδου) και εφαρμόζει 64 φίλτρα με μέγεθος 3x3 και padding ξανά. Ακολουθεί η δεύτερη κανονικοποίηση δεσμίδας.

Κατά την επίπεδη μετατροπή φαίνεται ότι το πρώτο πλήρως συνδεδεμένο επίπεδο έχει **64x56x56=200.704 νευρώνες**. Αυτό συμβαίνει διότι το δεύτερο συνελκτικό επίπεδο παράγει 64 feature maps, ενώ τα inputs-εικόνες από πίνακες 224x224 έχουν πλέον μετατραπεί σε 56x56 inputs. Αυτό συνέβη διότι περνώντας δύο φορές από το επίπεδο συγκέντρωσης (pooling) μειώθηκε το μέγεθος δύο φορές στο μισό, αφού εφαρμόστηκε δύο φορές patch 2x2. Δηλαδή 224x224, 112x112, 56x56.

Οι 200.704 νευρώνες του πρώτου πλήρως συνδεδεμένου επιπέδου συνδέονται με το επόμενο επίπεδο (self.fc2) που αποτελείται από 128 νευρώνες. Αυτό συνδέεται με το τελευταίο πλήρως συνδεδεμένο (self.fc3) που αποτελείται από 64 νευρώνες, το οποίο και εξάγει αποτέλεσμα ταξινόμησης για num\_classes = 10 κλάσεις (output neurons).

Ακολουθεί ο ορισμός της συνάρτησης σφάλματος και της τεχνικής βελτιστοποίησης. Χρησιμοποιήθηκε η **κατηγορική διασταυρούμενη εντροπία (Categorical Cross Entropy Loss)** και η **τεχνική στοχαστικής καθόδου κλίσης (Stochastic Gradient Descent - SGD)**.

```
loss_function = nn.CrossEntropyLoss(weight = class_weights.to(device))
optimizer = optim.SGD(model.parameters(), lr=0.002, momentum=0.9, weight_decay=1e-3)
```

Ένα σημαντικό στοιχείο είναι το βάρος (weight) κατά τον ορισμό της συνάρτησης σφάλματος. Καθώς στο σύνολο δεδομένων μας υπάρχουν ορυκτά των οποίων τα δείγματα είναι περισσότερα από άλλα ορυκτά, πρέπει να δοθούν μεγαλύτερα βάρη στα ορυκτά με λιγότερα δείγματα. Η αντιμετώπιση της **ανισορροπίας κλάσεων (class imbalance)** είναι απαραίτητη ώστε να μην οδηγηθεί το δίκτυο σε υπο-βέλτιστα αποτελέσματα δίνοντας μεγαλύτερη έμφαση σε ορυκτά με περισσότερα δείγματα, γεγονός που θα επηρέαζε και την γενικότερη απόδοσή του. Αυτό γίνεται με τις παρακάτω εντολές:

```
import torch.nn.functional as F
total_samples = len(dataset)
class_weights = [total_samples / (len(dataset.classes) * class_counts[class_idx]) for class_idx in range(len(dataset.classes))]
class_weights = F.normalize(torch.tensor(class_weights), p=1, dim=0)
class_weights
tensor([0.0655, 0.0774, 0.0988, 0.1261, 0.0931, 0.1204, 0.1039, 0.1182, 0.0960,
        0.1006])
```

Παρατηρούμε ότι στο σύνολο τους τα 10 βάρη πρέπει να αθροίζουν στο 1, ενώ οι κατηγορίες με περισσότερα δείγματα έχουν μικρότερα βάρη. Για παράδειγμα η πρώτη κλάση που αφορά τον Αζουρίτη (776 δείγματα-εικόνες) έχει βάρος 6.5% ενώ η τελευταία κλάση που αφορά το Τουρκουάζ (505 δείγματα-εικόνες) έχει βάρος 10%.

Όσον αφορά την βελτιστοποίηση, στην πραγματικότητα η συνάρτηση optim.SGD παρόλο που αναφέρεται στην stochastic gradient descent (SGD), λειτουργεί ως **μέθοδος καθόδου κλίσης με μίνι-δεσμίδες (Mini-Batch Gradient Descent)**. Αυτό αφορά τον τρόπο που λειτουργεί η βιβλιοθήκη torch.optim, αφού αντιλαμβάνεται ότι τα inputs είναι δεσμίδες (batches) εισόδων, όπως θέσαμε κατά την δημιουργία των απαραίτητων υπο-συνόλων παραπάνω.

Η μεταβλητή lr αναφέρεται στον **ρυθμό μάθησης (learning rate)**, η μεταβλητή momentum στην **ορμή** και η μεταβλητή weight\_decay στην **L2 κανονικοποίηση (L2 Regularization)**, έννοιες που καλύφθηκαν νωρίτερα στην παρούσα εργασία. Οι τιμές που έχουν δωθεί είναι ενδεικτικές και θα μεταβάλλονται στη συνέχεια κατά τις δοκιμές.

Κατά την εκτέλεση κάθε εκπαίδευσης χρησιμοποιούμε επίσης την τεχνική της **πρόωρης διακοπής (early stopping)**, με αναμονή έως και 5 εποχές όπου το σφάλμα επαλήθευσης (validation loss) χειροτερεύει.

```
# Early stopping
if average_validation_loss < best_validation_loss:
    best_validation_loss = average_validation_loss
    wait_epochs = 0
    # save the best model
    best_model_state_dict = model.state_dict()
else:
    wait_epochs += 1
    if wait_epochs >= early_stopping_patience: # early_stopping_patience = 5
        print(f"Validation loss has increased for {early_stopping_patience} consecutive epochs. Training stopped.")
        break
```

Αρχικά ορίζουμε κάθε δίκτυο να εκπαιδευτεί για 80 εποχές, και κάθε φορά αποθηκεύεται το δίκτυο με το καλύτερο έως τώρα validation loss, ώστε να το φορτώσουμε (load) αργότερα σε περίπτωση που συνεχιστεί η εκπαίδευση και σταματήσει λόγω early stopping, οπότε θα έχει ήδη ξεκινήσει να αυξάνεται το overfitting, το οποίο δεν επιθυμούμε.

Για κάθε δίκτυο παρακολουθούμε σε ζώντανο χρόνο την εκπαίδευση του όπως φαίνεται παρακάτω ενδεικτικά, έως ώτου σταματήσει λόγω early stopping:

Epoch 4/10 - Average Batch Training Loss: 1.6252503834272687  
 Epoch 4/10 - Average Validation Loss: 1.7333474159240723  
 Epoch 4/10 - Average Testing Loss: 1.735192867425772

Epoch 5/10 - Average Batch Training Loss: 1.610308970275678  
 Epoch 5/10 - Average Validation Loss: 1.7417632066286528  
 Epoch 5/10 - Average Testing Loss: 1.7420898859317486

Epoch 6/10 - Average Batch Training Loss: 1.6136866274632906  
 Epoch 6/10 - Average Validation Loss: 1.7365082594064565  
 Epoch 6/10 - Average Testing Loss: 1.7479649782180786

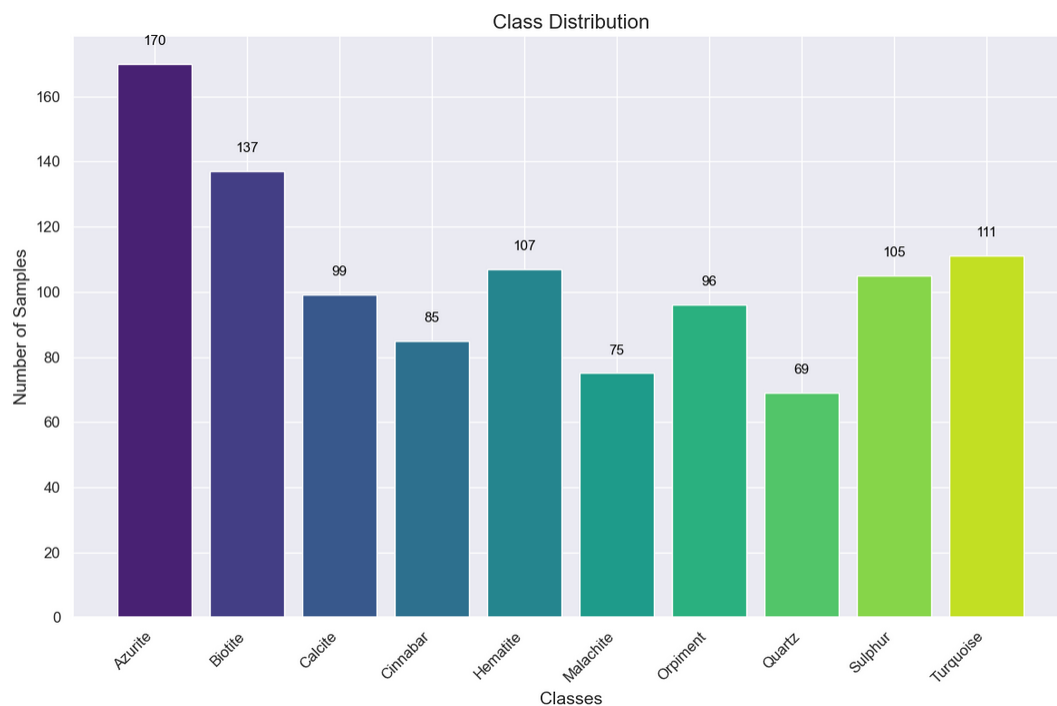
Epoch 7/10 - Average Batch Training Loss: 1.6066889794249284  
 Epoch 7/10 - Average Validation Loss: 1.7350892195334802  
 Epoch 7/10 - Average Testing Loss: 1.7396032443413367

Epoch 8/10 - Average Batch Training Loss: 1.6003124713897705  
 Epoch 8/10 - Average Validation Loss: 1.7396948887751653  
 Epoch 8/10 - Average Testing Loss: 1.7411248408831084

Epoch 9/10 - Average Batch Training Loss: 1.5995892945088839  
 Epoch 9/10 - Average Validation Loss: 1.745826849570641  
 Validation loss has increased for 5 consecutive epochs. Stopping training.

**Εικόνα 5.1.** Παρακολούθηση σε ζωντανό χρόνο της εκπαίδευσης του κάθε ΣΝΔ

Όλα τα πειράματα έγιναν πάνω στα ίδια υπο-σύνολα (dataloaders) τα οποία δημιουργήθηκαν μία φορά, με shuffle όπως δείξαμε παραπάνω. Το σύνολο των δειγμάτων στο test\_dataloader είναι 1054, το 20% δηλαδή του συνόλου των δεδομένων μας. Ενδεικτικά η κατανομή των δειγμάτων στο test set πάνω στο οποίο αξιολογούνται τα εκπαιδευμένα μοντέλα φαίνεται παρακάτω:



**Εικόνα 5.2.** Κατανομή των δειγμάτων στο test\_dataloader

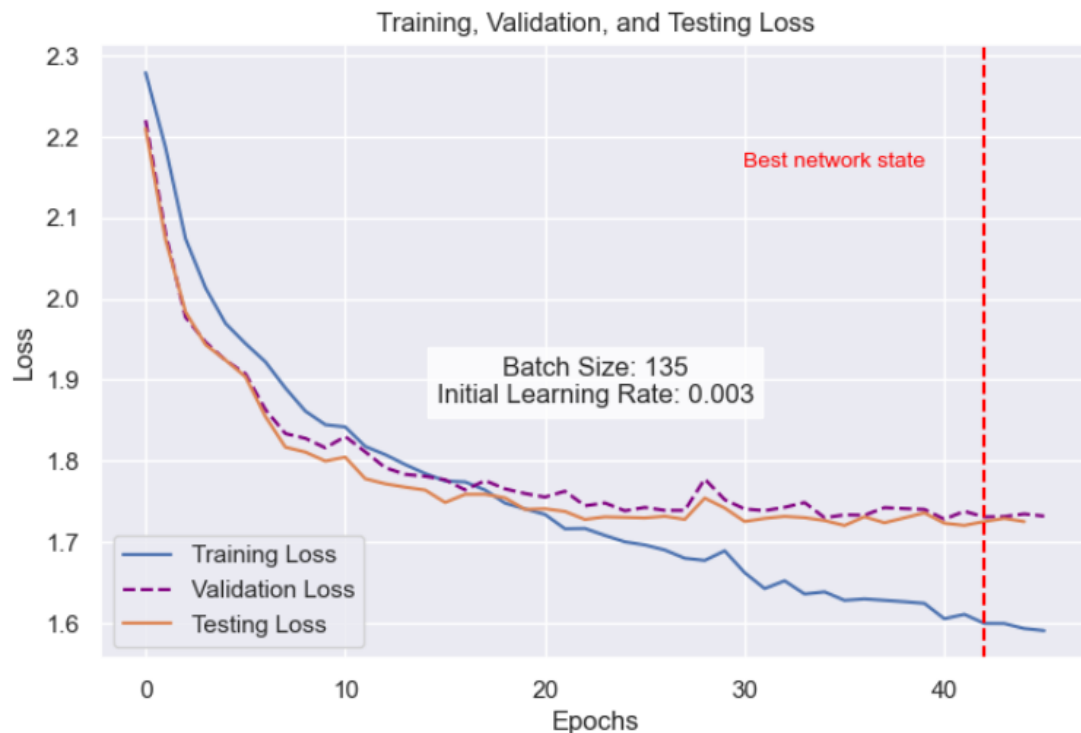
Παρακάτω παρουσιάζεται η ορθότητα (accuracy) που επετεύχθη για διάφορους συνδυασμούς batch size και αρχικών learning rate, με χρήση του **optimizer mini-batch gradient descent** και **συνάρτησης σφάλματος categorical cross entropy (weighted)**.

Η επιλογή μεγάλης τιμής **αρχικού learning rate** μπορεί να οδηγήσει σε ταχύτερη σύγκλιση, αλλά όταν το δίκτυο πλησιάζει προς το βέλτιστο που μπορεί να επιτύχει, υπάρχει κίνδυνος να παραλείψει το βέλτιστο καθώς οι μεγάλες αλλαγές βαρών «πηδούν» πάνω από το βέλτιστο. Η τεχνική που χρησιμοποιήσαμε είναι κάθε φορά που το δίκτυο δεν βελτιώνεται για 5 συνεχόμενες εποχές, το αποθηκεύουμε, μειώνουμε το αρχικό learning rate ( $\text{new\_lr} = \text{lr}/2$ ) και συνεχίζουμε την εκπαίδευση, με τελικό learning rate  $\text{lr} < 0.001$  όπου σταματάει λόγω early stopping και κρατάμε το τελευταίο καλύτερο δίκτυο βάσει του μέσου batch validation loss. Ο αριθμός που αναγράφεται στην παρένθεση είναι ο συνολικός αριθμός εποχών που χρειάστηκαν.

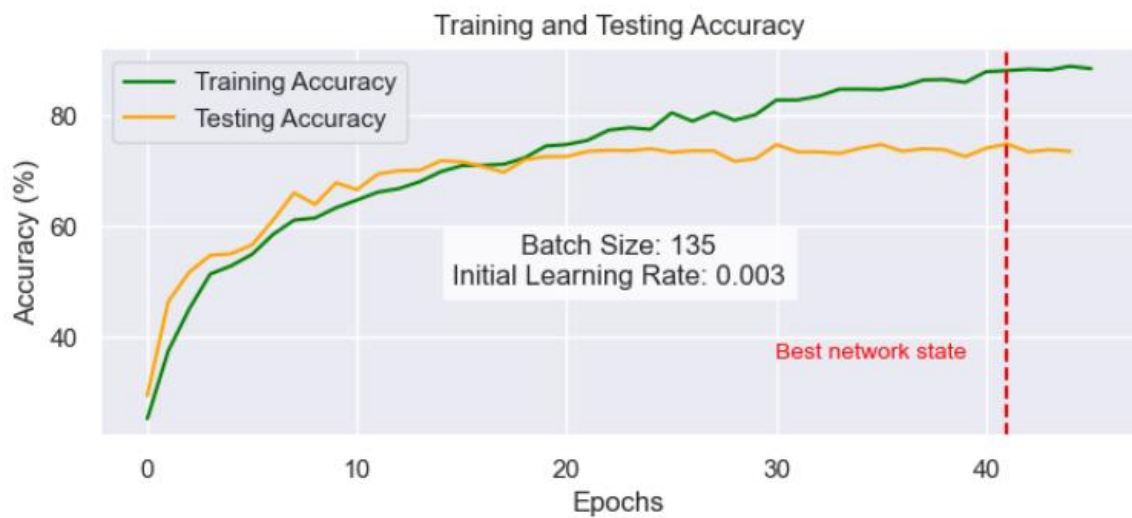
<u><i>Batch Size</i></u> <u><i>Learning rate</i></u>	0.003	0.002
60	72,86% (35)	68,24% (52)
85	71,82% (46)	70,77% (31)
110	68,97% (44)	70,49% (48)
135	<b>74,76% (41)</b>	73,34% (58)

**Πίνακας 5.2.** Testing Accuracy για διάφορους συνδυασμούς batch size/learning rate  
Optimizer : **Mini-Batch Gradient Descent**

Αποθηκεύοντας το **καλύτερο δίκτυο** σύμφωνα με το testing accuracy, προκύπτουν τα παρακάτω διαγράμματα που απεικονίζουν την διαδικασία εκπαίδευσης ανά εποχή.



**Εικόνα 5.3.** Μέσα σφάλματα εκπαίδευσης, επαλήθευσης, ελέγχου ανά εποχή



**Εικόνα 5.4.** Ορθότητα στο training και test set ανά εποχή

Η ορθότητα που πετυχαίνει το δίκτυο στο άγνωστο υπο-σύνολο δεδομένων ελέγχου είναι αρκετά ικανοποιητική. Για να έχουμε πλήρη εικόνα όμως απαιτούνται κι άλλες μέθοδοι μέτρησης αξιολόγησης της απόδοσης του, όπως η ακρίβεια (precision), το recall και το f1-Score. Ακολουθεί επίσης ο πίνακας σύγχυσης (confusion matrix).

Accuracy: 74.76%  
Precision: 75.32%  
Recall: 74.76%  
F1-score: 74.64%

Classification Report:

	precision	recall	f1-score	support
Azurite	0.93	0.88	0.90	170
Biotite	0.56	0.42	0.48	137
Calcite	0.70	0.64	0.67	99
Cinnabar	0.86	0.87	0.87	85
Hematite	0.51	0.75	0.61	107
Malachite	0.86	0.87	0.86	75
Orpiment	0.78	0.82	0.80	96
Quartz	0.65	0.59	0.62	69
Sulphur	0.83	0.87	0.85	105
Turquoise	0.83	0.79	0.81	111
accuracy			0.75	1054
macro avg	0.75	0.75	0.75	1054
weighted avg	0.75	0.75	0.75	1054

Confusion Matrix:

```
[[149  3  0  0 10  0  0  1  1  6]
 [ 6 58  8  4 48  0  4  7  1  1]
 [ 0  6 63  1  1  0 12  9  5  2]
 [ 1  3  0 74  3  0  1  3  0  0]
 [ 3 16  1  4 80  0  0  0  1  2]
 [ 0  0  0  0  6 65  0  0  0  4]
 [ 0  4  1  3  0  0 79  1  8  0]
 [ 1 10 10  0  3  0  1 41  0  3]
 [ 0  4  4  0  2  0  4  0 91  0]
 [ 1  0  3  0  4 11  0  1  3 88]]
```

Παρατηρούμε υψηλές τιμές ακρίβειας (precision) στις κλάσεις Αζουρίτης, Κιναβαρίτης, Μαλαχίτης και Θειάφι ενώ χαμηλότερες εντοπίζονται στις κλάσεις Βιοτίτης και Αιματίτης. Αυτό ήταν γενικώς αναμενόμενο καθώς τα δύο αυτά ορυκτά έχουν αρκετά παρόμοια χαρακτηριστικά μεταξύ τους, και το δίκτυο μπερδεύεται ελαφρώς κατατάσσοντας τον Βιοτίτη ως Αιματίτη και το αντίστροφο κατά τον έλεγχο του. Χαμηλή τιμή recall παρατηρείται επίσης στην κλάση *Ασβεστίτης*.

Για να αποφασίσουμε ποιο δίκτυο είναι το καταλληλότερο είναι χρήσιμο να δοκιμάσουμε επίσης το Adam optimizer αντί του Mini-Batch Gradient. Θα κρατήσουμε τελικά το καλύτερο δίκτυο που προέκυψε από όλους τους

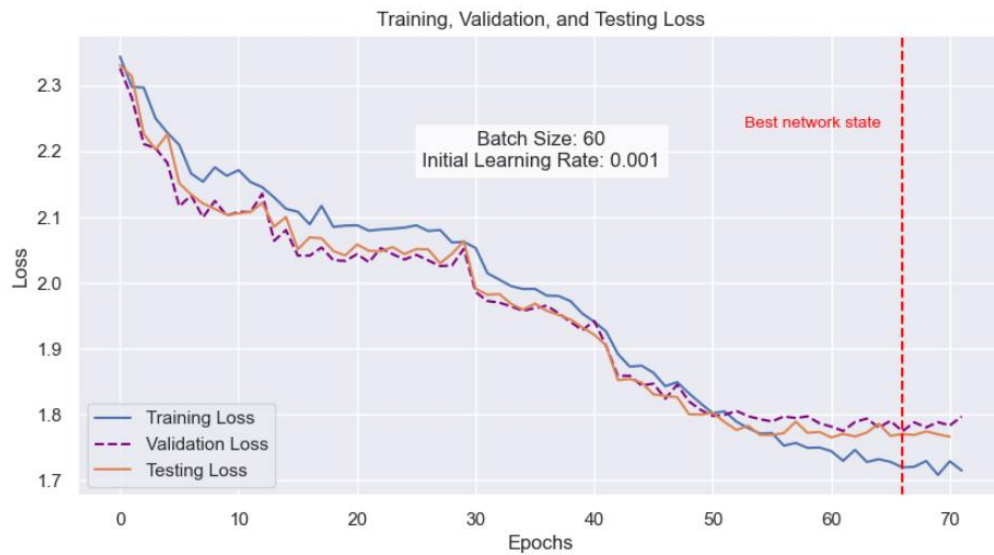
πειραματισμούς. Το ιδανικό αρχικό learning rate για το Adam ενδέχεται να είναι διαφορετικό για τα βέλτιστα αποτελέσματα γρήγορης και σταθερής σύγκλισης.

Παρακάτω παρουσιάζεται η ορθότητα (accuracy) που επετεύχθη για διάφορους συνδυασμούς batch size και αρχικών learning rate, με χρήση του **optimizer Adam** και της **συνάρτησης σφάλματος categorical cross entropy (weighted)**.

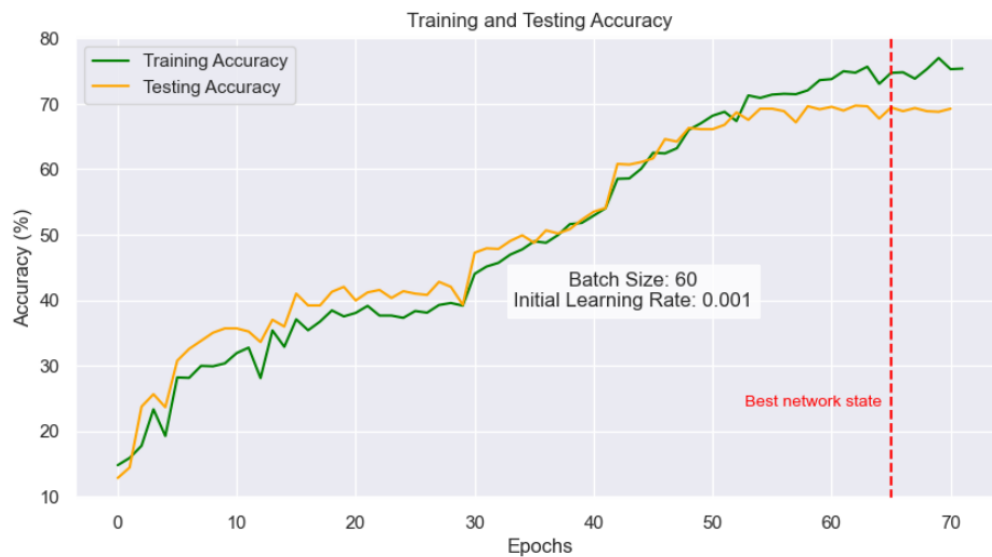
<u><i>Batch Size</i></u> <u><i>Learning rate</i></u>	<i>0,001</i>	<i>0,0005</i>
60	<b>69,07% (72)</b>	62,74% (76)
85	64,04% (47)	59,77% (58)
110	57,74% (48)	67,74% (68)
135	59,80% (84)	61,23% (90)

**Πίνακας 5.3.** *Testing Accuracy για διάφορους συνδυασμούς batch size/learning rate*  
*Optimizer : Adam*

Αποθηκεύοντας το **καλύτερο δίκτυο** σύμφωνα με το testing accuracy, προκύπτουν τα παρακάτω διαγράμματα που απεικονίζουν την διαδικασία εκπαίδευσης ανά εποχή, και το classification report.



**Εικόνα 5.5.** Μέσα σφάλματα εκπαίδευσης, επαλήθευσης, ελέγχου ανά εποχή



**Εικόνα 5.6.** Ορθότητα στο training και test set ανά εποχή



Accuracy: 69.07%  
Precision: 65.82%  
Recall: 69.07%  
F1-score: 67.11%

Classification Report:

	precision	recall	f1-score	support
Azurite	0.90	0.88	0.89	170
Biotite	0.49	0.50	0.50	137
Calcite	0.57	0.47	0.52	99
Cinnabar	0.83	0.87	0.85	85
Hematite	0.42	0.65	0.51	107
Malachite	0.82	0.84	0.83	75
Orpiment	0.69	0.78	0.74	96
Quartz	0.00	0.00	0.00	69
Sulphur	0.82	0.85	0.83	105
Turquoise	0.78	0.82	0.80	111
accuracy			0.69	1054
macro avg	0.63	0.67	0.65	1054
weighted avg	0.66	0.69	0.67	1054

Confusion Matrix:

```
[[150  2  0  1  9  1  0  0  1  6]
 [ 5 69  5  5 49  0  3  0  0  1]
 [ 2 14 47  1  8  0 19  0  5  3]
 [ 0  6  1 74  3  0  1  0  0  0]
 [ 4 24  2  4 70  0  1  0  0  2]
 [ 0  0  0  0  2 63  0  0  0 10]
 [ 0  3  1  2  2  1 75  0 12  0]
 [ 4 19 22  2 17  0  2  0  0  3]
 [ 0  3  4  0  1  0  7  0 89  1]
 [ 1  0  0  0  5 12  0  0  2 91]]
```

Κατεβάζοντας δύο άγνωστες στο δίκτυο μας εικόνες από το διαδίκτυο, ενός Αζουρίτη και ενός Βιοτίτη, θα αξιολογηθούν οι προβλέψεις του δικτύου. Χρησιμοποιούμε επίσης **temperature = 0,3** ώστε να δώσει προβλέψεις με μεγαλύτερη σιγουριά (confidence).



**Εικόνα 5.7.** Εικόνα Αζουρίτη για έλεγχο της σωστής κατηγοριοποίησης

```
# Get the class probabilities
temperature = 0.3 # enhance prediction confidence
probabilities = torch.nn.functional.softmax(output[0] / temperature, dim=0)

# Predicted class = index with the highest probability
predicted_class = torch.argmax(probabilities).item()

class_labels = dataset.classes
predicted_label = class_labels[predicted_class]

print(f"Predicted class: {predicted_label}")
print(f"Class probabilities: {probabilities}")

Predicted class: Azurite
Class probabilities: tensor([0.7570, 0.0270, 0.0270, 0.0270, 0.0270, 0.0270, 0.0270, 0.0270, 0.0270,
0.0270])
```

**Εικόνα 5.8.** Η πρόβλεψη του δικτύου για την άγνωστη εικόνα-input του Αζουρίτη

Το δίκτυο βλέπουμε ότι προέβλεψε σωστά το είδος του ορυκτού, δίνοντας πιθανότητα 75,7% η εικόνα να απεικονίζει ορυκτό Αζουρίτη, και πολύ μικρότερες πιθανότητες να απεικονίζει οτιδήποτε άλλο.



**Εικόνα 5.9.** Εικόνα Βιοτίτη για έλεγχο της σωστής κατηγοριοποίησης

Η πρόβλεψη του δικτύου για την παραπάνω εικόνα Βιοτίτη:

```
Predicted class: Biotite
Class probabilities: tensor([0.0287, 0.7400, 0.0286, 0.0286, 0.0309, 0.0286, 0.0286, 0.0286, 0.0286,
0.0286])
```

**Εικόνα 5.10.** Η πρόβλεψη του δικτύου για την άγνωστη εικόνα-input του Βιοτίτη

Η πρόβλεψη του ορκυτού είναι ξανά σωστή παρά τους ενδοιασμούς λόγω ομοιότητας με τον Αιματίτη, δίνοντας πιθανότητα 74% η εικόνα να απεικονίζει ορुकτό Βιοτίτη, και πολύ μικρότερες πιθανότητες να απεικονίζει οτιδήποτε άλλο.

Τα αποτελέσματα θα ήταν ακόμη καλύτερα αν είχαμε στη διάθεση μας περισσότερες εικόνες, ή ίσως και αν δοκιμάζαμε ένα πιο μεγάλο ΣΝΔ. Η χρήση ενός πιο πολύπλοκου ΣΝΔ ίσως οδηγήσει σε υπερ-προσαρμογή, οπότε απαιτείται εκπαίδευση και πειραματισμός. Θα ακολουθήσει η εκπαίδευση και αξιολόγηση μίας γνωστής αρχιτεκτονικής ΣΝΔ, του VGG16, ώστε να γίνει σύγκριση των αποτελεσμάτων.

### 5.5. Μοντέλο 2<sup>ο</sup> : Συνελικτικό Νευρωνικό Δίκτυο VGG16

Το VGG16 (Visual Geometry Group 16) είναι μια αρχιτεκτονική συνελικτικού νευρωνικού δικτύου (CNN) που έχει σχεδιαστεί για ταξινόμηση και αναγνώριση εικόνων. Εισήχθη από την ομάδα Visual Geometry στο Πανεπιστήμιο της Οξφόρδης το 2014, με τη δημοσίευση "VERY DEEP CONVOLUTIONAL NETWORKS FOR LARGE-SCALE IMAGE RECOGNITION", κερδίζοντας το 1<sup>ο</sup> και 2<sup>ο</sup> βραβείο στα δύο μέρη του διαγωνισμού ILSVRC challenge, που αφορούσε τον εντοπισμό αντικειμένων σε εικόνες και την κατηγοριοποίηση εικόνων αντίστοιχα.

Το "16" στο VGG16 αναφέρεται στο βάθος του δικτύου, υποδεικνύοντας ότι έχει 16 επίπεδα, συμπεριλαμβανομένων 13 συνελικτικών και 3 πλήρως συνδεδεμένων επιπέδων.

Τα βασικά χαρακτηριστικά του VGG16 περιλαμβάνουν:

- **Ομοιόμορφη αρχιτεκτονική:** Το VGG16 έχει απλή και ομοιόμορφη αρχιτεκτονική, με μικρά συνελικτικά φίλτρα 3x3 που χρησιμοποιούνται σε όλο το δίκτυο. Αυτή η απλότητα καθιστά εύκολη την κατανόηση και την εφαρμογή το.
- **Βάθος:** Το βάθος του δικτύου, που επιτυγχάνεται με τη στοίβαξη πολλαπλών συνελικτικών επιπέδων, του επιτρέπει να μαθαίνει ιεραρχικά χαρακτηριστικά αυξανόμενης πολυπλοκότητας.
- **Συγκεντρωτικά επίπεδα:** Τα συγκεντρωτικά επίπεδα (max-pooling) που χρησιμοποιούνται για τη μείωση του χωρικών διαστάσεων της εισόδου, μειώνοντας την υπολογιστική πολυπλοκότητα
- **Πλήρως συνδεδεμένα επίπεδα:** Το δίκτυο ολοκληρώνεται με τρία πλήρως συνδεδεμένα επίπεδα, ακολουθούμενα από μια συνάρτηση ενεργοποίησης Softmax για ταξινόμηση.



μεγάλο dataset (ImageNet). Αυτό συνήθως εφαρμόζεται ώστε να διατηρείται η «γνώση» αν τα σύνολα δεδομένων έχουν αρκετά κοινά μεταξύ τις ή και για τη μείωση του υπολογιστικού κόστους, αφού λιγότερες παράμετροι χρειάζεται να αλλάζουν σε κάθε εποχή.

Στα πειράματα που θα ακολουθήσουν, καθώς η κατηγοριοποίηση ορυκτών θεωρούμε ότι είναι πολύ συγκεκριμένη εργασία και έχουμε στη διάθεση μας σχετικά μικρό σύνολο εικόνων, θα αφήσουμε τις παραμέτρους ελεύθερες να προσαρμοστούν κατά την εκπαίδευση.

<u><i>Batch Size</i></u> <u><i>Learning rate</i></u>	0,001
60	87,57% (5 epochs)
85	86,24% (5 epochs)
110	<b>87,95% (5 epochs)</b>
135	85,86% (5 epochs)

**Πίνακας 5.3.** *Testing Accuracy* για διάφορες επιλογές *batch size* για το VGG16 Optimizer : **Mini – Batch Gradient Descent**

Γενικότερα παρατηρούμε ότι για οποιοδήποτε μέγεθος batch size το προ-εκπαιδευμένο ΣΝΔ VGG16 τα πηγαίνει καλύτερα από το προσαρμοσμένο μας δίκτυο, επιτυγχάνοντας εντυπωσιακά αποτελέσματα ακρίβειας. Αυτό ήταν εμφανώς κάτι αναμενόμενο.

Αποθηκεύοντας το **καλύτερο δίκτυο** σύμφωνα με το testing accuracy, προκύπτουν τα παρακάτω διαγράμματα που απεικονίζουν την διαδικασία εκπαίδευσης ανά εποχή, και την έκθεση ταξινόμησης (classification report).



**Εικόνα 5.12.** Μέσα σφάλματα εκπαίδευσης, επαλήθευσης, ελέγχου ανά εποχή (VGG)

Accuracy: 87.95%  
Precision: 88.15%  
Recall: 87.95%  
F1-score: 87.94%

Classification Report:

	precision	recall	f1-score	support
Azurite	0.97	0.98	0.97	170
Biotite	0.87	0.81	0.84	137
Calcite	0.81	0.71	0.76	99
Cinnabar	0.96	0.95	0.96	85
Hematite	0.76	0.86	0.81	107
Malachite	0.93	0.89	0.91	75
Orpiment	0.91	0.93	0.92	96
Quartz	0.73	0.83	0.78	69
Sulphur	0.91	0.88	0.89	105
Turquoise	0.89	0.91	0.90	111
accuracy			0.88	1054
macro avg	0.87	0.87	0.87	1054
weighted avg	0.88	0.88	0.88	1054

Confusion Matrix:

```
[[167  0  0  0  3  0  0  0  0  0]
 [ 2 111  0  0 14  0  2  7  0  1]
 [ 0  3 70  0  3  0  3 14  5  1]
 [ 0  1  1 81  1  0  1  0  0  0]
 [ 0  7  2  3 92  0  0  0  0  3]
 [ 2  0  0  0  1 67  0  0  0  5]
 [ 0  2  1  0  1  0 89  0  2  1]
 [ 1  2  4  0  3  0  0 57  1  1]
 [ 0  1  7  0  1  0  3  0 92  1]
 [ 1  0  1  0  2  5  0  0  1 101]]
```

Οι τιμές όλων των μεθόδων μέτρησης, για το δίκτυο αλλά και για κάθε ορυκτό ξεχωριστά είναι πολύ υψηλές, με εξαίρεση ίσως τον Αιματίτη και τον Χαλαζία που εμφανίζουν σχετικά χαμηλή ακρίβεια (precision), γεγονός που επηρεάζει και το f1-score για τις συγκεκριμένες κλάσεις.

Δοκιμάζοντας δύο άγνωστες στο δίκτυο εικόνες από το διαδίκτυο, ενός Αζουρίτη και ενός Βιοτίτη, θα αξιολογήσουμε τις προβλέψεις του δικτύου VGG16. Χρησιμοποιούμε επίσης **temperature = 0,3** ώστε να δώσει προβλέψεις με μεγαλύτερη σιγουριά (confidence).



**Εικόνα 5.14.** Εικόνα Αιματίτη για έλεγχο της σωστής κατηγοριοποίησης

```
Predicted class: Hematite  
Class probabilities: tensor([1.1269e-13, 1.0688e-02, 6.5214e-14, 2.8735e-14, 9.8931e-01, 1.5719e-08,  
5.1443e-17, 6.6921e-12, 9.2711e-16, 2.8797e-17], device='cuda:0')
```

**Εικόνα 5.15.** Η πρόβλεψη του VGG16 για την άγνωστη εικόνα-input του Αιματίτη

Το δίκτυο βλέπουμε ότι προέβλεψε σωστά το είδος του ορυκτού, δίνοντας πιθανότητα 98,9% η εικόνα που απεικονίζει το ορυκτό Αζουρίτη, και πολύ μικρότερες πιθανότητες να απεικονίζει οτιδήποτε άλλο. Επίσης, για την εικόνα του Βιοτίτη με την οποία αξιολογήθηκε και το προσαρμοσμένο δίκτυο παραπάνω, το VGG16 κάνει την παρακάτω πρόβλεψη.

```
Predicted class: Biotite  
Class probabilities: tensor([1.0170e-16, 9.9717e-01, 9.2027e-14, 3.2410e-16, 2.8348e-03, 4.3565e-16,  
2.2201e-14, 1.9022e-13, 2.8874e-14, 4.0629e-16], device='cuda:0')
```

**Εικόνα 5.16.** Η πρόβλεψη του VGG16 για την άγνωστη εικόνα-input του Βιοτίτη

Το δίκτυο προέβλεψε σωστά ξανά το είδος του ορυκτού, δίνοντας πιθανότητα 99,7% η εικόνα να απεικονίζει ορυκτό Βιοτίτη.



## 6. Ερμηνεία αποτελεσμάτων

Συγκρίνοντας τα αποτελέσματα με τις αρχικές μας υποθέσεις, γίνεται φανερό πως το νευρωνικό μας δίκτυο μπορεί να ξεχωρίσει τις κλάσεις των ορυκτών. Δυστυχώς η χρήση του φίλτρου Laplace για την αναγνώριση ακμών στην εικόνα δεν ήταν επιτυχής, αφού δεν βοήθησε το νευρωνικό μας δίκτυο να βελτιώσει την απόδοση του και να επαληθεύσει τον Νόμο της συμμετρίας των κρυσταλλικών σχημάτων ως τρόπο σύγκρισης των κλάσεων. Πιθανά τα αποτελέσματα και η διαδικασία να είχαν μεγαλύτερη επιτυχία αν το πλήθος πληροφοριών ανά κλάση ήταν μεγαλύτερο. Πιο συγκεκριμένα το παραπάνω φίλτρο αφαιρέθηκε κατά τη διαδικασία της εκπαίδευσης του νευρωνικού δικτύου ώστε να βελτιωθούν τα κριτήρια απόδοσής του. Επιπλέον παρατηρούνται υψηλές τιμές ακρίβειας (precision) στις κλάσεις Αζουρίτης, Κιναβαρίτης, Μαλαχίτης και Θειάφι ενώ χαμηλότερες εντοπίζονται στις κλάσεις Βιοτίτης, Ασβεστίτης και Αιματίτης. Χαμηλή τιμή recall παρατηρείται επίσης στις κλάσεις Ασβεστίτης και Βιοτίτης. Κάτι πολύ σημαντικό είναι η πλήρης «αποτυχία» του δικτύου να αναγνωρίσει τον Χαλαζία (μηδενικά μετρικά precision, recall, f1-score). Τα αποτελέσματα του βέλτιστου δικτύου με χρήση του Adam είναι εμφανώς χειρότερα από τον Mini-Batch Gradient Descent, και γενικότερα σύμφωνα με τις τιμές των τεσσάρων γενικών μεθόδων μέτρησης αλλά και ειδικότερα παρατηρώντας τις τιμές των μεθόδων μέτρησης και για κάθε κλάση ορυκτού μεμονωμένα. Το καλύτερο δίκτυο προέκυψε με χρήση του Mini-Batch Gradient Descent optimizer, το οποίο και θα κρατήσουμε από όλα τα παραπάνω πειράματα. Συνολικά, τα αποτελέσματα είναι αρκετά ικανοποιητικά, αφού το **καλύτερο μοντέλο (optimizer: Mini-Batch Gradient Descent, Batch Size:135, Initial Learning Rate: 0,003)** παρουσιάζει υψηλές τιμές και στα 4 μετρικά και τα πηγαίνει πολύ καλά στην σωστή κατηγοριοποίηση άγνωστων εικόνων στην σωστή ετικέτα ορυκτού.

Από την άλλη πλευρά για το έτοιμο και προ εκπαιδευμένο δίκτυο τα αποτελέσματα είναι ξεκάθαρα. Αρχικά η εκπαίδευσή του ήταν ευκολότερη αφού για οποιοδήποτε μέγεθος παρτίδας σε συνδυασμό με πλησίον αριθμό στην ταχύτητα εκμάθησης επιτυγχάνει ακρίβεια ανώτερη του 85%. Τα πειράματα δεν άργησαν να εμφανίσουν το βέλτιστο αποτέλεσμα του δικτύου VGG16. Το **καλύτερο μοντέλο (Mini – Batch Gradient Descent, Batch Size 110, Initial Learning Rate 0,001)**. Συμφωνα με τα παραπάνω αποτελέσματα είναι ορθό να συμπεράνουμε πως η εκπαίδευση του νευρωνικού δικτύου σε άλλα αντικείμενα ενισχύει την αντίληψη του ως προς τις διαφορές των ορυκτών. Τα δύο δίκτυα δεν έχουν μεγάλη διαφορά αρχιτεκτονικά. Το κατασκευασμένο μας δίκτυο είναι μικρότερο αφού αποτελείται από μόνο 2 συνελκτικά επίπεδα σε σύγκριση με το μεγαλύτερο VGG16. Προφανώς τα συστήματα εξελίσσονται, αλλά η διαφορά τους ανάμεσα στην ακρίβεια καλύπτεται ως προς τα μικρότερα απαιτούμενα χαρακτηριστικά που χρειάζεται το κάθε σύστημα για να εξοπλιστεί με το κάθε εργαλείο.

Επιπλέον μπορεί η αυτοπεποίθηση του προ εκπαιδευμένου VGG16 να είναι μεγαλύτερη, αλλά και τα δύο νευρωνικά καταλήγουν σε ορθό αποτέλεσμα. Άρα ο σκοπός της χρήσης της τεχνητής νοημοσύνης σαν συνοδευτικό εργαλείο του ανθρώπου, ορυκτολόγου ή μηχανικού είναι εφικτή.

Συνοψίζοντας τα αποτελέσματα είναι ικανοποιητικά ως προς τους αρχικούς σκοπούς. Ενδεχομένως με την ανάπτυξη των βάσεων δεδομένων στον κυβερνοχώρο να υπάρχει και βελτίωση των αποτελεσμάτων. Δυνητικά η τεχνητή νοημοσύνη είναι ικανή να ξεχωρίσει διαφορετικά ορυκτά από ψηφιακές εικόνες που βρίσκονται στο ίδιο πέτρωμα. Σε ακόμα πιο γενικό επίπεδο θα μπορούσε να αναγνωρίσει διαφορές ανάμεσα γεωλογικά στρώματα και να συμβάλλει στην στρωματογραφία. Όμως αυτές οι προσαρμογές απαιτούν χρόνο μεγαλύτερη κατανόηση και βάθος στο γνωστικό επίπεδο που χρησιμοποιήθηκε για την σύνταξη αυτής της εργασίας. Επιπλέον οι πληροφορίες που είναι ελεύθερες για χρήση είναι περιορισμένες, οπότε η κατασκευή αντίστοιχων συνόλων για εκπαίδευση θεωρείται αδύνατη τη συγκεκριμένη χρονική στιγμή. Όσον αφορά την επιλογή διαφόρων παραμέτρων όπως επαύξηση των δεδομένων πρόωρη διακοπή , μετρικές αξιολόγησης κ.α. έγιναν από υποκειμενική εμπειρία και είναι εφικτές διαφορετικές επιλογές να καταλήξουν σε καλύτερο αποτέλεσμα.

## 7. Βιβλιογραφία - Αναφορές

### **Βιβλία - Άρθρα:**

1. Krizhevsky A., Sutskever I., Hinton G.E., "ImageNet Classification with Deep Convolutional Neural Networks", University of Toronto, 2012
2. Βλαχάβας Ι., Κεφαλάς Π., Βασιλειάδης Ν., Κόκκορας Φ., Σακελλαρίου Η., «Τεχνητή Νοημοσύνη, Δ' έκδοση, Εκδόσεις Πανεπιστημίου Μακεδονίας 2020»
3. Wei L, Dexian Z, Review of mineral recognition and its future ,Applied Geochemistry, Volume 122, 2020,
4. Zhang J, Gao Q, Luo H, Long T. Mineral Identification Based on Deep Learning Using Image Luminance Equalization. Applied Sciences. 2022; 12(14):7055.
5. Διαμαντάρας Κ. , Μπότσης Δ., «Μηχανική Μάθηση, Εκδ. Κλειδάριθμος, 2019»
6. Weiwen L, Shan Q, Three-dimensional quantitative mineral prediction from convolutional neural network model in developing intelligent cleaning technology, Resources Policy, Volume 88, 2024,
7. C. Muller, Sarah Guido, "Introduction to Machine Learning with Python"
8. Russ, J.C., The image-processing handbook. 5th ed. 2007, Boca Raton: CRC/Taylor and Francis.
9. Rich E. & K. Knight "Artificial Intelligence", 2nd Edition, 1991
10. Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. Understanding the exploding gradient problem. In: CoRR abs/1211.5063 (2012).
11. Ali Sharif Razavian, Hossein Azizpour, Josephine Sullivan, Stefan Carlsson, "CNN Features off-the-shelf: an Astounding Baseline for Recognition" CVAP, KTH (Royal Institute of Technology), Stockholm, Sweden, 2014.
12. J. Schmidhuber, "Deep Learning in neural networks: An overview," Neural Networks
13. S. Ioffe and C. Szegedy, "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift," 2015.
14. C. Szegedy et al., "Going Deeper with Convolutions, 2014.

15. Κορνίτσας, Κ. (2017). Εμπλουτισμός Μεταλλευμάτων , Σημειώσεις του μαθήματος, Πολυτεχνείο Κρήτης, Χανιά.

### Ιστοσελίδες:

1. PyTorch Documentation (<https://pytorch.org/docs/stable/index.html>)
2. Scikit-learn Documentation (<https://scikit-learn.org/stable/index.html>)
3. CUDA Documentation (<https://docs.nvidia.com/cuda/>)
4. Anaconda Documentation (<https://docs.anaconda.com/free/anaconda/>)
5. <https://www.kaggle.com/>
6. <https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>
7. <https://www.sciencedirect.com/science/article/abs/pii/S0883292720302195>
8. <https://towardsdatascience.com/>
9. <https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/>
10. <https://neurohive.io/en/popular-networks/vgg16/>
11. <https://datagen.tech/guides/computer-vision/vgg16/>
12. <https://www.image-net.org/challenges/LSVRC/2014/>
13. <https://www.sciencedirect.com/science/article/abs/pii/S0301420723011297>
14. <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/>
15. <https://www.kaggle.com/datasets/asiedubrempong/minerals-identification-dataset>
16. <https://www.mindat.org/>
17. <https://www.mdpi.com/2076-3417/12/14/7055>

## Βιβλιογραφία εικόνων

- Εικόνα 2.3: <https://stackoverflow.com/questions/22255992/conversion-form-rgb-to-hdr-image-format-exr-format-in-c>
- Εικόνες 2.1, 2.2: Τροποποίηση της Εικόνας 2.3. με Image Channel Splitter (RGB) <https://www.aatbio.com/tools/online-image-channel-splitter-rgb>
- Εικόνα 2.4
- [https://commons.wikimedia.org/wiki/File:Image\\_processing\\_pre\\_otsus\\_algorithm.jpg](https://commons.wikimedia.org/wiki/File:Image_processing_pre_otsus_algorithm.jpg) και τροποποίηση με image thresholder <https://threshold.imageonline.co/>
- Εικόνα 2.5. <http://wallpapersforus.blogspot.com/2010/03/ponds-rivers-lakes.html>, μετατροπή σε gray-scale και τροποποίηση με image segmentation με χρήση k-means με χρήση Python
- Εικόνα 2.6. [www.mindat.org](http://www.mindat.org) και εφαρμογή φίλτρου Laplace με χρήση Python
- Εικόνα 2.7. <https://www.mathworks.com/discovery/support-vector-machine.html>
- Εικόνα 2.8. <https://medium.com/analytics-vidhya/random-forest-classifier-and-its-hyperparameters-8467bec755f6>
- Εικόνα 2.9. Πλατφόρμα Udemy
- Εικόνα 3.1. Πλατφόρμα Udemy
- Εικόνα 3.2. <https://habr.com/en/sandbox/83605/>
- Εικόνα 3.3. Πλατφόρμα Udemy
- Εικόνα 3.4. <https://www.spiceworks.com/tech/artificial-intelligence/articles/what-is-a-neural-network/>
- Εικόνα 3.5. Πλατφόρμα Udemy
- Εικόνα 3.6. <https://www.ibm.com/topics/gradient-descent>
- Εικόνα 3.7. <https://stackoverflow.com/questions/51109921/why-is-gradient-descent-with-momentum-an-exponentially-weighted-average>
- Εικόνα 4.1. <https://medium.com/dataroots/a-gentle-introduction-to-geometric-deep-learning-by-vitale-sparacello-91e529cf439b>
- Εικόνα 4.2. Πλατφόρμα Udemy
- Εικόνα 4.3. Αυτοδημιούργητη
- Εικόνα 4.4. Αυτοδημιούργητη
- Εικόνα 4.5. <https://towardsdatascience.com/convolutional-neural-networks-explained-how-to-successfully-classify-images-in-python-df829d4ba761>
- Εικόνα 4.6. <https://it.emcelettronica.com/object-detection-con-yolo-v3>
- Εικόνα 4.7. <https://medium.com/mllearning-ai/convolutional-networks-recurrent-neural-networks-and-transformers-ee6a63ddb57f>
- Εικόνα 4.8. Πλατφόρμα Udemy
- Εικόνα 4.9. Πλατφόρμα Udemy
- Εικόνα 4.10. <https://towardsdatascience.com/mlp-mixer-an-all-mlp-architecture-for-vision-7438fac99a06>
- Εικόνα 5.7, 5.8, 5.14. [www.mindat.org](http://www.mindat.org)
- Εικόνα 5.11. <https://www.geeksforgeeks.org/vgg-16-cnn-model/>