

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ - ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ  
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΠΟΛΟΓΙΣΤΩΝ



ΠΑΝΕΠΙΣΤΗΜΙΟ  
**ΠΑΤΡΩΝ**  
UNIVERSITY OF PATRAS

Τομέας: Τομέας Συστημάτων και Αυτομάτου Ελέγχου  
Εργαστήριο: Εργαστήριο Ψηφιακής Επεξεργασίας Σημάτων και  
Εικόνων

### Διπλωματική Εργασία

του φοιτητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας  
Υπολογιστών της Πολυτεχνικής Σχολής του Πανεπιστημίου Πατρών

Τσουδερού Γεώργιου του Ιωάννη

αριθμός μπτρώου: 1059528

### Θέμα

Συμπίεση Εικόνας με Τεχνικές Βαθιάς Μάθησης

### Επιβλέπων

Καθηγητής Αθανάσιος Σκόδρας, Πανεπιστήμιο Πατρών

Αριθμός Διπλωματικής Εργασίας:

Πάτρα, Ιούνιος 2023



Πανεπιστήμιο Πατρών, Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών.

Τσουδερός Γεώργιος

© 2023 – Με την επιφύλαξη παντός δικαιώματος

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

# ΠΙΣΤΟΠΟΙΗΣΗ

Πιστοποιείται ότι η διπλωματική εργασία με θέμα

## Συμπίεση Εικόνας με Τεχνικές Βαθιάς Μάθησης

του φοιτητή του Τμήματος Ηλεκτρολόγων Μηχανικών και Τεχνολογίας  
Υπολογιστών

Τσουδερού Γεώργιου του Ιωάννη

(Α.Μ.: 1059528)

παρουσιάτηκε δημόσια και εξετάστηκε στο τμήμα Ηλεκτρολόγων Μηχανικών και  
Τεχνολογίας Υπολογιστών στις

06/07/2023

Ο Επιβλέπων

Ο Διευθυντής του Τομέα

Αθανάσιος Σκόδρας  
Καθηγητής

Χαράλαμπος Μπεχλιούλης  
Αναπληρωτής Καθηγητής

# Στοιχεία διπλωματικής εργασίας

Θέμα: Συμπίεση Εικόνας με Τεχνικές Βαθιάς Μάθησης

Φοιτητής: Τσουδερός Γεώργιος του Ιωάννη

Εξεταστική Επιτροπή

Καθηγητής Αθανάσιος Σκόδρας, Πανεπιστήμιο Πατρών, Τμήμα ΗΜΤΥ

Αναπληρωτής Καθηγητής Ψαράκης Εμμανουήλ, Πανεπιστήμιο Πατρών,  
Τμήμα ΜΗΥ&Π

Καθηγητής Μουστάκας Κωνσταντίνος, Πανεπιστήμιο Πατρών, Τμήμα  
ΗΜΤΥ

Συνεπιβλέπων

Επίκουρος Καθηγητής Αμπελιώτης Δημήτριος, Πανεπιστήμιο Ιονίου, Τμήμα  
Ψηφιακών Μέσων και Επικοινωνίας

Εργαστήριο Εργαστήριο Ψηφιακής Επεξεργασίας Σημάτων και Εικόνων

Περίοδος εκπόνησης της εργασίας:

Μάιος 2021 - Ιούλιος 2023

Η εργασία αυτή γράφτηκε σε L<sup>A</sup>T<sub>E</sub>X.



# Περίληψη

Έχουμε πλέον εισέλθει στην ψηφιακή εποχή. Η νέα έκρηξη περιεχομένου από τα μέσα ενημέρωσης, των κοινωνικών πλατφόρμων και των πλατφόρμων ζωντανής συνεχούς μετάδοσης, έχουν καταστήσει τον τομέα της συμπίεσης εικόνας κρίσιμο προς μελέτη. Η πρόκληση έγκειται στην προσπάθεια εξισορρόπησης του μεγέθους των αρχείων και της ποιότητας της μεταδιδόμενης εικόνας. Ενώ οι παραδοσιακές τεχνικές, όπως το JPEG, έχουν αποτελέσει τα θεμέλια στην συμπίεση της εικόνας, η εμφάνιση της Τεχνητής Νοημοσύνης (TN), και συγκεκριμένα της Βαθιάς Μάθησης, έχει εισάγει μια νέα διάσταση στον συγκεκριμένο τομέα.

Στην παρούσα διπλωματική παρουσιάζεται το ευρύ φάσμα εφαρμογών της Τεχνητής Νοημοσύνης και κατόπιν ακολουθεί ανασκόπηση των πρόσφατων εξελίξεων στη συμπίεση εικόνας με βάση τη Βαθιά Μάθηση. Παρουσιάζονται ωστόσο πρώτα οι κλασικές μέθοδοι συμπίεσης, ενώ τονίζεται η σημασία των πλέον καθιερωμένων κωδικοποιητών όπως το πρότυπο JPEG, όπου παρά την εμφάνιση νεότερων μεθόδων, συνεχίζουν να χρησιμοποιούνται ευρέως και να χρησιμεύουν ως σημεία αναφοράς βάσει των οποίων αξιολογούνται όλοι οι νέοι αλγόριθμοι συμπίεσης.

Παρουσιάζεται μια λεπτομερής εξέταση των διαφόρων τύπων νευρωνικών δικτύων, με έμφαση στην αρχιτεκτονική, τη λειτουργικότητα και τις δυνατότητές τους στην επεξεργασία εικόνας και τη συμπίεση. Τέλος, κατασκευάζεται το προτεινόμενο μοντέλο συμπίεσης εικόνας και συγκρίνεται με καθιερωμένες λύσεις. Ενώ τα αποτελέσματα αναδεικνύουν τις δυνατότητες της βαθιάς μάθησης στη συμπίεση εικόνας, υπογραμμίζουν επίσης την λειτουργικότητα των παραδοσιακών τεχνικών.

Η διπλωματική ανοίγει το δρόμο για μελλοντική διερεύνηση, η οποία θα περιλαμβάνει μια εις βάθος μελέτη διαφορετικών μοντέλων νευρωνικών δικτύων και μια προσπάθεια τελειοποίησης της παρουσιαζόμενης μεθόδου. Τα ευρήματα επιβεβαιώνουν τις δυνατότητες της τεχνητής νοημοσύνης στη διαμόρφωση του μέλλοντος της συμπίεσης εικόνας και την ανάγκη περαιτέρω βελτίωσης για την πλήρη αξιοποίηση των δυνατοτήτων της.



# Abstract

We are currently living in the digital age. The new explosion of content from social media platforms and live streaming platforms have made the image compression field critical to study. The challenge lies in trying to balance the size and the quality of the transmitted image. While traditional techniques, such as JPEG, have been the foundation in image compression, the emergence of Artificial Intelligence (AI), specifically Deep Learning, has introduced a new dimension in this area.

In this thesis, the wide range of applications of AI is presented, followed by a review of the recent developments in Deep Learning based image compression. However, classical compression methods are presented first, while the importance of the well established encoders such as the JPEG standard is emphasized, where despite the emergence of newer methods, they continue to be widely used and serve as benchmarks against which all new compression algorithms are evaluated.

A detailed examination of the different types of neural networks is presented, focusing on their architecture, functionality and potential in image processing and compression. Finally, the proposed image compression model is constructed and compared with the established compression models. While the results highlight the potential of deep learning in image compression, they also underline the robustness of traditional techniques.

The thesis paves the way for future exploration, which will include an in-depth study of different neural network models and an attempt to refine the presented method. The findings confirm the potential of artificial intelligence in shaping the future of image compression and the need for further improvement to fully exploit its potential.



## Ευχαριστίες

Θα ήθελα να ευχαριστήσω θερμά τα μέλη της ομάδας επίβλεψης και ιδιαίτερα τον καθηγητή και επιβλέποντα αυτής της διπλωματικής εργασίας κ. Αθανάσιο Σκόδρα για την συνεχή καθοδήγηση όπου μου προσέφερε αλλά και για την άμεση ανταπόκριση του όταν χρειαζόμουν βοήθεια. Επιπλέον, οφείλω να δώσω ένα μεγάλο ευχαριστώ στον διδάκτορα Παναγιώτη Τσίγγανο και στην Σοφία Ηλιοπούλου για την προθυμία τους να με βοηθήσουν σε οποιοδήποτε θέμα προέκυπτε κατά την διεξαγωγή της διπλωματικής μου εργασίας καθώς και για την άμεση ανταπόκριση τους στις ερωτήσεις όπου τους έθετα.



# Περιεχόμενα

<b>Κατάλογος σχημάτων</b>	<b>xv</b>
0.1 Acronyms . . . . .	xvii
<b>1 Κλασικές Μέθοδοι Συμπίεσης Εικόνας</b>	<b>1</b>
1.1 Εισαγωγή . . . . .	1
1.2 Θεωρητικό Υπόβαθρο . . . . .	1
1.2.1 Συμπίεση δίχως Απώλειες . . . . .	1
1.2.1.1 Αριθμητική Κωδικοποίηση . . . . .	2
1.2.1.2 Κωδικοποίηση Huffman . . . . .	3
1.2.1.3 Κωδικοποίηση Lempel-Ziv-Welch (LZW) . . . . .	4
1.2.1.4 Κωδικοποίηση Μίκους Διαδρομής . . . . .	6
1.2.1.5 Προβλεπτική Κωδικοποίηση . . . . .	7
1.2.1.6 Κωδικοποίηση Περιοχής . . . . .	8
1.2.2 Συμπίεση με Απώλειες . . . . .	8
1.2.2.1 Κωδικοποίηση Μετασχηματισμού . . . . .	9
1.2.2.2 Κωδικοποίηση Αποκοπής Μπλοκ . . . . .	10
1.2.2.3 Μορφοκλασματική Κωδικοποίηση . . . . .	11
1.3 Το Πρότυπο JPEG . . . . .	11
1.3.1 Τα Μοντέλα JPEG Κωδικοποίησης . . . . .	12
1.3.2 Ο Αλγόριθμος Κωδικοποίησης JPEG . . . . .	12
1.3.2.1 Μετατροπή Χρωματικού Χώρου . . . . .	13
1.3.2.2 Διακριτός Μετασχηματισμός Συνημιτόνου . . . . .	14
1.3.2.3 Κβάντιση Συντελεστών . . . . .	14
1.3.2.4 Κωδικοποίηση Huffman . . . . .	14
1.3.3 Κριτήρια πιστότητας . . . . .	15
1.3.3.1 Μέσο Τετραγωνικό Σφάλμα (MSE) . . . . .	15
1.3.3.2 Peak Signal-To-Noise Ratio (PSNR) . . . . .	16
1.3.3.3 Δομική Ομοιότητα (SSIM) . . . . .	16
<b>2 Μηχανική Μάθηση</b>	<b>17</b>
2.1 Εισαγωγή . . . . .	17
2.2 Μηχανική Μάθηση . . . . .	17
2.2.1 Επιβλεπόμενη Μάθηση (Supervised Learning) . . . . .	18
2.2.2 Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning) . . . . .	20
2.2.3 Ήμι-Επιβλεπόμενη Μάθηση (Semi-Supervised Learning) . . . . .	20
2.2.4 Ενισχυτική Μάθηση (Reinforcement Learning) . . . . .	21
2.3 Βαθιά Μάθηση . . . . .	21
2.3.1 Εισαγωγή . . . . .	21
2.3.2 Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks - ANN) . . . . .	21
2.3.3 Αρχιτεκτονικές των Τεχνητών Νευρωνικών Δικτύων . . . . .	22
2.3.4 Εφαρμογές . . . . .	23

2.3.5	Συνάρτηση Ενεργοποίησης . . . . .	24
2.3.6	Εκπαίδευση . . . . .	26
2.3.7	Οπισθοδιάδοση . . . . .	27
2.3.8	Υπερ-παραμετροί Νευρωνικών Δικτύων . . . . .	27
2.4	Συνελικτικά Δίκτυα (Convolutional Neural Networks - CNN) . . . . .	32
2.4.1	Αρχιτεκτονική Συνελικτικών Δικτύων . . . . .	33
2.4.2	Εφαρμογές Συνελικτικών Δικτύων . . . . .	36
2.5	Autoencoders . . . . .	36
2.5.1	Αρχιτεκτονική των Autoencoders . . . . .	36
2.5.2	Κατηγορίες Autoencoders . . . . .	38
2.5.2.1	Undercomplete autoencoder . . . . .	38
2.5.2.2	Denoising autoencoders (DAE) . . . . .	39
2.5.2.3	Sparse autoencoders (SAE) . . . . .	40
2.5.2.4	Contractive autoencoders (CAE) . . . . .	41
2.6	Αναγεννητικά Αντιπαραθετικά Δίκτυα (GANs) . . . . .	41
2.6.1	Αρχιτεκτονική των Αναγεννητικών Αντιπαραθετικών Δικτύων . . . . .	42
2.6.2	Εφαρμογές των Αναγεννητικών Αντιπαραθετικών Δικτύων . . . . .	43
<b>3</b>	<b>Συμπίεση Εικόνας Με Χρήση Τεχνικών Βαθιάς Μάθησης</b>	<b>45</b>
3.1	Βιβλιογραφική Ανασκόπηση . . . . .	45
<b>4</b>	<b>Επιλεχθείσα Μέθοδος</b>	<b>51</b>
4.0.1	Βάση Δεδομένων . . . . .	51
4.0.2	Επιλεχθείσα Μέθοδος . . . . .	51
4.0.2.1	Προεπεξεργασία Δεδομένων . . . . .	51
4.0.2.2	Αρχιτεκτονική του μοντέλου . . . . .	52
4.0.2.3	Αρχιτεκτονική ComCNN . . . . .	52
4.0.2.4	Αρχιτεκτονική RecCNN . . . . .	53
4.0.2.5	JPEG . . . . .	54
4.0.2.6	Εκπαίδευση . . . . .	54
<b>5</b>	<b>Πειραματική Διαδικασία</b>	<b>57</b>
5.0.1	Πειραματική Διαδικασία . . . . .	57
5.0.2	Συμπεράσματα και μελλοντικοί στόχοι . . . . .	67
<b>Βιβλιογραφία</b>		<b>69</b>

# Κατάλογος σχημάτων

1.1 Αριστερά: Συμπίεση δύχως απώλειες. Δεξιά: Συμπίεση με απώλειες. [1]	1
1.2 Ένα διάγραμμα που δείχνει την αποκωδικοποίηση του 0,538 (η στρογγυλή τελεία) στο μοντέλο του παραδείγματος. Η περιοχή διαιρείται σε υποπεριοχές ανάλογες με τις συχνότητες των συμβόλων, και στη συνέχεια η υποπεριοχή που περιέχει το σημείο υποδιαιρείται διαδοχικά με τον ίδιο τρόπο. [2]	3
1.3 Δέντρο Huffman που παράγεται από τις ακριβείς συχνότητες του κειμένου "αυτό είναι ένα παράδειγμα ενός δέντρου Huffman". [3]	4
1.4 Παράδειγμα πίνακα κωδικοποίησης με μέθοδο LZW. [4]	5
1.5 Παράδειγμα πίνακα αποκωδικοποίησης με μέθοδο LZW. [4]	6
1.6 Παράδειγμα εφαρμογής του αλγορίθμου Κωδικοποίησης Μήκους Διαδρομής σε δυαδική εικόνα.	7
1.7 Διάγραμμα ροής της συμπίεσης με απώλειες.	9
1.8 Παράδειγμα κωδικοποίησης μετασχηματισμού σε εικόνα. [5]	10
1.9 Διαδικασία συμπίεσης και αποσυμπίεσης εικόνας με JPEG.	13
2.1 Διάγραμμα Βεν τεχνητής νοημοσύνης, μηχανικής μάθησης, βαθιάς μάθησης. [6]	17
2.2 Ο κύκλος της τεχνητής νοημοσύνης. [7]	18
2.3 Παράδειγμα επιβλεπόμενης μάθησης. [8]	18
2.4 Παράδειγμα μη επιβλεπόμενης μάθησης. [8]	20
2.5 Παράδειγμα ήμι-επιβλεπόμενης μάθησης. [8]	21
2.6 Παράδειγμα νευρώνα στα νευρωνικά δίκτυα. [9]	22
2.7 Παράδειγμα πλήρους συνδεδεμένου νευρωνικού δικτύου. [10]	23
2.8 Η σιγμοειδής συνάρτηση ενεργοποίησης. [11]	24
2.9 Η υπερβολική εφαπτομένη συνάρτηση ενεργοποίησης. [12]	25
2.10 Η ανυψωμένη γραμμική μονάδα συνάρτηση ενεργοποίησης. [13]	25
2.11 Η Leaky ReLu συνάρτηση ενεργοποίησης. [14]	26
2.12 Η softmax συνάρτηση ενεργοποίησης. [15]	26
2.13 Παράδειγμα υποπροσαρμογής και υπερπροσαρμογής. [16]	32
2.14 Συνελικτικό δίκτυο για την αναγνώριση οχημάτων.	33
2.15 Αρχιτεκτονική ενός συνελικτικού δικτύου. [17]	33
2.16 Πράξη συνέλιξης με μηδενική συμπλήρωση (zero padding) ώστε να διατηρούνται οι διαστάσεις. Σε αυτό το παράδειγμα, το μέγεθος του πυρίνα και το βήμα ορίζονται ως $3 \times 3$ και 1, αντίστοιχα. [18]	34
2.17 Πράξη συνέλιξης σε εικόνα RGB. [19]	35
2.18 Average pooling και max pooling σε μια εικόνα. [20]	36
2.19 Αρχιτεκτονική Autoencoder[21].	37
2.20 Στάδια συνελικτικού autoencoder.[22]	38
2.21 Αρχιτεκτονική undercomplete autoencoder.	39
2.22 Αρχιτεκτονική DAE [23].	40
2.23 Αρχιτεκτονική SAE [24].	40

2.24 Αρχιτεκτονική CAE [25]. . . . .	41
2.25 Αρχιτεκτονική των αναγεννητικών αντιπαραθετικών δικτύων [26]. . . . .	42
2.26 Τρείς εικόνες που έχουν δημουργηθεί από το Midjourney[27]. . . . .	43
2.27 Εφαρμογή του image inpainting σε εικόνα. . . . .	43
3.1 Αποτελέσματα με χρήση CAE.[28] . . . . .	46
3.2 Αποτελέσματα των Zhengxue Cheng et al.[29] . . . . .	47
3.3 Αποτελέσματα των Yoojin Choi et al.[30] . . . . .	48
3.4 Μοντέλο των Mu Li et al.[31] . . . . .	48
3.5 Αποτελέσματα των Mu Li et al.[31] . . . . .	49
4.1 Υλοποίηση του μοντέλου[32]. . . . .	52
4.2 Αρχιτεκτονική του ComCNN[32]. . . . .	53
4.3 Αρχιτεκτονική του RecCNN[32]. . . . .	54
5.1 PSNR γραφική ρυθμού παραμόρφωσης για το validation test. . . . .	58
5.2 PSNR γραφική ρυθμού παραμόρφωσης για το BSD300 dataset. . . . .	58
5.3 SSIM γραφική ρυθμού παραμόρφωσης για το validation set. . . . .	59
5.4 SSIM γραφική ρυθμού παραμόρφωσης για το BSD300 dataset. . . . .	59
5.5 Εικόνα συμπιεσμένη από το μοντέλο, με bpp=1.079 (αριστερά) και εικόνα συμπιεσμένη με JPEG, με bpp=1.075 (δεξιά). . . . .	60
5.6 Εικόνα συμπιεσμένη από το μοντέλο, με bpp=1.1 (αριστερά) και εικόνα συμπιεσμένη με JPEG, με bpp=1 (δεξιά). . . . .	60
5.7 Εικόνα συμπιεσμένη από το μοντέλο, με bpp=1.366 (αριστερά) και εικόνα συμπιεσμένη με JPEG, με bpp=1.424 (δεξιά). . . . .	61
5.8 Εικόνα συμπιεσμένη από το μοντέλο, με bpp=1.00 (αριστερά) και εικόνα συμπιεσμένη με JPEG, με bpp=1.01 (δεξιά). . . . .	61
5.9 PSNR ιστόγραμμα με QF=15 στο validation set. . . . .	62
5.10 SSIM ιστόγραμμα με QF=15 στο validation set. . . . .	62
5.11 Έξοδος ComCNN και συμπίεση JPEG. . . . .	63
5.12 Upscaled εικόνα. . . . .	63
5.13 Η πραγματική υπολειψματική εικόνα. . . . .	64
5.14 Υπολειψματική εικόνα. . . . .	64
5.15 Ανακατασκευασμένη εικόνα από το δίκτυο RecCNN. . . . .	65
5.16 Πραγματική υπολειψματική εικόνα + upscaled. . . . .	65
5.17 Έξοδος έπειτα από JPEG κωδικοποίηση. . . . .	66
5.18 Κατάτυπηση υπολειψματικής εικόνας. . . . .	66
5.19 Πραγματική υπολειψματική εικόνα. . . . .	67
5.20 Τελική ανακατασκευασμένη εικόνα. . . . .	67

## 0.1 Acronyms

**AI** ARTIFICIAL INTELLIGENCE

**SVM** SUPPORT VECTOR MACHINE

**CNN** CONVOLUTIONAL NEURAL NETWORK

**RNN** RECURRENT NEURAL NETWORK

**LSTM** LONG SHORT TERM MEMORY

**LZW** LEMPEL ZIV WELCH

**DPCM** DIFFERENTIAL PULSE MODULATION

**IFS** ITERATED FUNCTION SYSTEM

**JPEG** JOINT PHOTOGRAPHIC EXPERTS GROUP

**RGB** RED-GREEN-BLUE

**MSE** MEAN SQUARED ERROR

**DCT** DISCRETE COSINE TRANSFORM

**PSNR** PEAK SIGNAL TO NOISE RATIO

**SSIM** STRUCTURAL SIMILARITY

**GAN** GENERATIVE ADVERSARIAL NEURAL NETWORK

**DAE** DENOISING AUTOENCODER

**SAE** SPARSE AUTOENCODER

**CAE** CONVOLUTIONAL AUTOENCODER

**CAE** CONTRACTIVE AUTOENCODER

**ReLU** RECTIFIED LINEAR UNIT

**RLE** RUN LENGTH ENCODING

**TC** TRANSFORM CODING

**BTC** BLOCK TRUNCATION CODING



# 1. Κλασικές Μέθοδοι Συμπίεσης Εικόνας

## 1.1 Εισαγωγή

Η συμπίεση εικόνας, δηλαδή η μείωση του όγκου των απαραίτητων δεδομένων για την αναπαράσταση μιας εικόνας, αποτελεί μια από τις σπουδαιότερες και πιο χρήσιμες τεχνολογίες στο πεδίο της ψηφιακής επεξεργασίας εικόνας. Καθώς τα συστήματα και οι εφαρμογές σε αυτόν τον κλάδο αναπτύσσονται, η συμπίεση εικόνας και τα πρότυπά που χρησιμοποιούνται έχουν αποκτήσει ολοένα και μεγαλύτερη σημασία. Οι συνεχείς μειώσεις του κόστους στην υπολογιστική ισχύ, την αποδύναμευση και τις επικοινωνίες καθιστούν τα συστήματα αυτά πιο πρακτικά, με τη συμπίεση να περιλαμβάνεται σχεδόν πάντα για την επίτευξη οικονομικά αποδοτικότερων λύσεων. Συνεπώς, πριν προχωρήσουμε στους τρεις προσαρμογές στης μηχανικής μάθησης για την αύξηση απόδοσης στην συμπίεση εικόνας, είναι σημαντικό να δοθεί αρχικά το απαραίτητο θεωρητικό υπόβαθρο.

## 1.2 Θεωρητικό Υπόβαθρο

### 1.2.1 Συμπίεση δίχως Απώλειες

Υπάρχουν δύο κύριοι τρόποι για την εφαρμογή συμπίεσης σε δεδομένα, η συμπίεση δίχως απώλειες και η συμπίεση με απώλειες.



Σχήμα 1.1: Αριστερά: Συμπίεση δίχως απώλειες. Δεξιά: Συμπίεση με απώλειες. [1]

Ας ξεκινήσουμε από την πρώτη κατηγορία: Η βασική αρχή της συμπίεσης εικόνας δίχως απώλειες είναι η μείωση του μεγέθους της εικόνας χωρίς να χάνεται καμία από τις πληροφορίες της αρχικής κατά την ανακατασκευή της [33]. Αυτό επιτυγχάνεται με τη χρήση αλγορίθμων που εκμεταλλεύονται τα επαναλαμβανόμενα μοτίβα στα δεδομένα της εικόνας, όπως οι πλεονασμοί και οι επαναλήψεις, για να κωδικοποιήσουν τα δεδομένα σε πιο συμπαγή μορφή. Αυτό είναι σημαντικό

και χρησιμοποιείται σε περιπτώσεις που η ποιότητα της εικόνας είναι κρίσιμη, όπως στην ιατρική κατά την απεικόνιση ακτινογραφιών ή σε άλλες εφαρμογές που ακόμη και μικρές αλλαγές στα αρχικά δεδομένα θα μπορούσαν να έχουν σημαντικές συνέπειες. Οι πιο γνωστές μέθοδοι για την συμπίεση χωρίς απώλειες είναι:

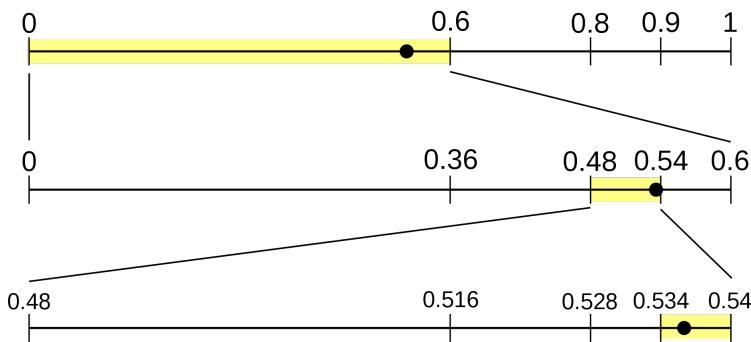
- Αριθμητική Κωδικοποίηση
- Κωδικοποίηση Huffman
- Κωδικοποίηση Lempel-Ziv-Welch (LZW)
- Κωδικοποίηση Μίκους Διαδρομής
- Κωδικοποίηση Επιπέδου bits
- Προβλεπτική Κωδικοποίηση
- Κωδικοποίηση Βασισμένη σε Σύμβολα

#### 1.2.1.1 Αριθμητική Κωδικοποίηση

Η αριθμητική κωδικοποίηση πρόκειται για μία τεχνική κωδικοποίησης η οποία διαφέρει από τους κλασσικούς κώδικες μεταβλητού μήκους. Στον συγκεκριμένο τρόπο κωδικοποίησης, δεν αντιστοιχίζονται ένα προς ένα τα σύμβολα της πηγής (μήνυμα) με τις κωδικολέξεις που τα τα περιγράφουν. Αντίθετα, μια αριθμητική κωδικολέξη αντιστοιχίζεται σε μία ολόκληρη ακολουθία συμβόλων πηγής. Ο τρόπος με τον οποίο πραγματοποιείται η παραπάνω διαδικασία είναι η εξής: Η κάθε κωδικολέξη ορίζει ένα μοναδιαίο διάστημα  $[0,1]$ . Όσο μεγαλύτερο είναι το πλήθος των συμβόλων του μηνύματος, το διάστημα το οποίο θα χρησιμοποιηθεί για την αναπαράσταση του μικραίνει. Αντίστοιχα, οι μονάδες πληροφορίας (bits) που απαιτούνται για την αναπαράσταση του διαστήματος αυτού αυξάνονται. Είναι αξιοσημείωτο ότι επειδή η συγκεκριμένη τεχνική δεν απαιτεί την μετατροπή του κάθε συμβόλου σε ένα ακέραιο πλήθος κωδικοσυμβόλων (όπως συμβαίνει στην κωδικοποίηση Huffman), μπορεί σε θεωρητικό επίπεδο να καταφέρει να προσεγγίζει το όριο που υπαγορεύει το πρώτο θεώρημα του Shannon. [34] Για την καλύτερη κατανόηση της αριθμητικής κωδικοποίησης, ακολουθεί αντίστοιχο παράδειγμα.

Έστω ότι επιθυμούμε την κωδικοποίηση του μηνύματος  $a_1a_2a_3a_4$ , το οποίο έχει προέλθει από μία πηγή τεσσάρων διαφορετικών συμβόλων. Αρχικά, το μήνυμα καταλαμβάνει το διάστημα  $[0,1]$ . Στην συνέχεια, διαχωρίζουμε το διάστημα αυτό σε υποδιαστήματα, το κάθε ένα ίσο με την πιθανότητα εμφάνισης του αντίστοιχου συμβόλου. Έστω ότι η πιθανότητα εμφάνισης του συμβόλου  $a_1$  ισούται με  $P(a_1) = 0.2$ . Έτσι, επειδή τυχαίνει να είναι και το πρώτο σύμβολο που εμφανίζεται στο μήνυμα που κωδικοποιούμε, το διάστημα του μηνύματος περιορίζεται στο  $[0,0.2]$ . Στην συνέχεια, διαχωρίζουμε ξανά το νέο διάστημα έτσι ώστε το κάθε υποδιάστημα να αντιστοιχεί με την πιθανότητα εμφάνισης του αντίστοιχου συμβόλου, προσαρμοσμένη όμως στα νέα άκρα. Κατόπιν, εντοπίζουμε το επόμενο σύμβολο που εμφανίζεται στο μήνυμα, το  $a_2$ , το οποίο περιορίζει το διάστημα στο  $[0.04,0.08]$ . Η διαδικασία συνεχίζεται έως ότου τελειώσουν τα σύμβολα του μηνύματος. Όταν καταλήξουμε στο τελευταίο υποδιάστημα, στην συγκεκριμένη περίπτωση το  $[0.06752,0.0688]$ , μπορούμε να επιλέξουμε οποιοδήποτε αριθμό σε αυτό το διάστημα για την αναπαράσταση του μηνύματος, πχ το 0.0688. Προσοχή στο γεγονός ότι τα άνω άκρα του διαστήματος είναι πάντα ανοικτά και συνεπώς δεν μπορεί να επιλεχτεί η ανώτατη τιμή του διαστήματος. Από το παραπάνω παράδειγμα παρατηρούμε ότι για ένα μήνυμα 5 συμβόλων χρειάστηκαν μόνο 3 δεκαδικά ψηφία για την κωδικοποίηση του. Αυτό αντιστοιχεί σε 0.6 δεκαδικά ψηφία ανά σύμβολο πηγής, το οποίο είναι αρκετά κοντά με την εντροπία της πηγής, η οποία είναι 0.58. Όσο το μήκος της ακολουθίας αυξάνεται, και όπως περιγράφηκε προηγουμένως, το υποδιάστημα ελαττώνεται, η τεχνική της αριθμητικής κωδικοποίησης προσεγγίζει την τιμή της εντροπίας. Ωστόσο, πρακτικά αυτό δεν

είναι εφικτό καθώς υπάρχει η προσθήκη συμβόλου που σηματοδοτεί το τέλος του μηνύματος και η κωδικοποίηση χαρακτηρίζεται από πεπερασμένη αριθμητική ακρίβεια.



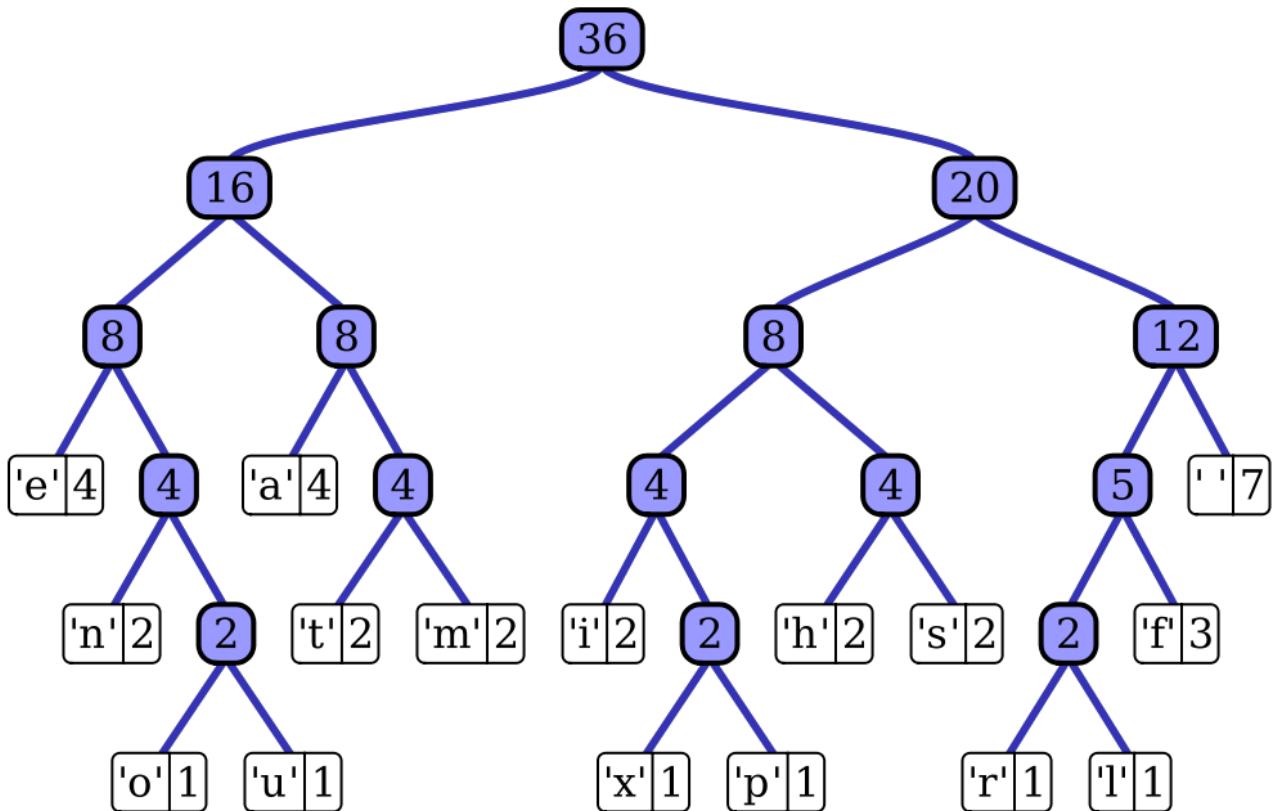
Σχήμα 1.2: Ένα διάγραμμα που δείχνει την αποκωδικοποίηση του 0,538 (η στρογγυλή τελεία) στο μοντέλο του παραδείγματος. Η περιοχή διαιρείται σε υποπεριοχές ανάλογες με τις συχνότητες των συμβόλων, και στη συνέχεια η υποπεριοχή που περιέχει το σημείο υποδιαιρείται διαδοχικά με τον ίδιο τρόπο. [2]

### 1.2.1.2 Κωδικοποίηση Huffman

Η κωδικοποίηση Huffman είναι μια από τις πιο δημοφιλείς τεχνικές για την αφαίρεση πλεονασμού κατά την κωδικοποίηση και αναπτύχθηκε από τον David A. Huffman το 1952, ενώ ήταν φοιτητής στο MIT [35]. Είναι σημαντικό να αναφερθεί ότι στις περιπτώσεις που επιθυμούμε κωδικοποίηση των συμβόλων πηγής ένα προς ένα, σύμφωνα με το πρώτο θεώρημα του Shannon, η κωδικοποίηση Huffman είναι η βέλτιστη και οδηγεί στον ελάχιστο αριθμό κωδικοσυμβόλων ανά σύμβολο πηγής. Στις περιπτώσεις της συμπίεσης εικόνας, τα σύμβολα της πηγής συνήθως εκφράζουν τιμές των εντάσεων της εικόνας.

Ο αλγόριθμος με τον οποίο εφαρμόζεται η κωδικοποίηση Huffman είναι ο παρακάτω:

- Καταμέτρηση της συχνότητας εμφάνισης κάθε χαρακτήρα στο σύνολο δεδομένων.
- Δημιουργία ενός δυαδικού δέντρου με βάση τη συχνότητα κάθε χαρακτήρα, με τους λιγότερο συχνούς χαρακτήρες στο κάτω μέρος του δέντρου.
- Ανάθεση ενός δυαδικού κώδικα σε κάθε χαρακτήρα με βάση τη θέση του στο δυαδικό δέντρο, με το αριστερό παιδί να λαμβάνει ένα "0" και το δεξί παιδί ένα "1".
- Κωδικοποίηση του αρχικού μηνύματος χρησιμοποιώντας τους αποδιδόμενους δυαδικούς κώδικες για κάθε χαρακτήρα.
- Αποκωδικοποίηση του κωδικοποιημένου μηνύματος διατρέχοντας το δυαδικό δέντρο με βάση τους δυαδικούς κώδικες, ανακατασκευάζοντας το αρχικό μήνυμα.



Σχήμα 1.3: Δέντρο Huffman που παράγεται από τις ακριβείς συχνότητες του κειμένου "αυτό είναι ένα παράδειγμα ενός δέντρου Huffman". [3]

Η κωδικοποίηση Huffman χρησιμοποιείται ευρέως σε εφαρμογές που απαιτείται συμπίεση δεδομένων, όπως η συμπίεση εικόνας, πίχου και βίντεο, καθώς και σε εφαρμογές που βασίζονται σε κείμενο, όπως η συμπίεση αρχείων και η μετάδοση δεδομένων μέσω δικτύων. Επιπλέον, τα περισσότερα πρότυπα κωδικοποίησης εικόνας χρησιμοποιούν τεχνικές συμπίεσης με απώλειες στα αρχικά στάδια της συμπίεσης και χρησιμοποιούν την κωδικοποίηση Huffman ως τελικό στάδιο.

### 1.2.1.3 Κωδικοποίηση Lempel-Ziv-Welch (LZW)

Η κωδικοποίηση Lempel-Ziv-Welch (LZW) είναι ένας αλγόριθμος συμπίεσης δεδομένων ο οποίος εστιάζει στην αντιμετώπιση των χωρικών πλεονασμών που εμφανίζονται σε μία εικόνα. Περιγράφηκε για πρώτη φορά το 1984 από τους Abraham Lempel, Jacob Ziv και Terry Welch και πρόκειται για μια τεχνική κωδικοποίησης βασισμένη σε λεξικά που λειτουργεί αντικαθιστώντας ακολουθίες χαρακτήρων με κώδικες.

Συγκεκριμένα, ο αλγόριθμος LZW δημιουργεί ένα λεξικό ακολουθιών χαρακτήρων που συναντώνται στα δεδομένα εισόδου και αναθέτει έναν κωδικό σε κάθε ακολουθία. Κατά τη διάρκεια της κωδικοποίησης, τα δεδομένα εισόδου σαρώνονται, αναζητώντας ακολουθίες δεδομένων οι οποίες ταιριάζουν με αυτές του λεξικού. Όταν βρεθεί αντιστοιχία, εξάγεται ο αντίστοιχος κωδικός. Όταν συναντάται μια νέα ακολουθία, προστίθεται στο λεξικό και αποδίδεται ένας νέος κωδικός. Συνεπώς, από την παραπάνω διαδικασία προκύπτει και ένα από τα βασικότερα χαρακτηριστικά του αλγορίθμου, το οποίο είναι ότι δεν απαιτείται εκ των προτέρων γνώση των πιθανοτήτων εμφάνισης των συμβόλων προς κωδικοποίηση.

Current Sequence	Next Char	Output		Extended Dictionary	Comments	
		Code	Bits			
NULL	T					
T	O	20	10100	27: TO	27 = first available code after 0 through 26	
O	B	15	01111	28: OB		
B	E	2	00010	29: BE		
E	O	5	00101	30: EO		
O	R	15	01111	31: OR		
R	N	18	10010	32: RN	32 requires 6 bits, so for next output use 6 bits	
N	O	14	001110	33: NO		
O	T	15	001111	34: OT		
T	T	20	010100	35: TT		
TO	B	27	011011	36: TOB		
BE	O	29	011101	37: BEO		
OR	T	31	011111	38: ORT		
TOB	E	36	100100	39: TOBE		
EO	R	30	011110	40: EOR		
RN	O	32	100000	41: RNO		
OT	#	34	100010		# stops the algorithm; send the cur seq	
		0	000000		and the stop code	

Σχήμα 1.4: Παράδειγμα πίνακα κωδικοποίησης με μέθοδο LZW. [4]

Input		Output Sequence	New Dictionary Entry		Comments
Bits	Code		Full	Conjecture	
10100	20	T		27: T?	
01111	15	O	27: TO	28: O?	
00010	2	B	28: OB	29: B?	
00101	5	E	29: BE	30: E?	
01111	15	O	30: EO	31: O?	
10010	18	R	31: OR	32: R?	created code 31 (last to fit in 5 bits)
001110	14	N	32: RN	33: N?	so start reading input at 6 bits
001111	15	O	33: NO	34: O?	
010100	20	T	34: OT	35: T?	
011011	27	TO	35: TT	36: TO?	
011101	29	BE	36: TOB	37: BE?	36 = TO + 1st symbol (B) of
011111	31	OR	37: BEO	38: OR?	next coded sequence received (BE)
100100	36	TOB	38: ORT	39: TOB?	
011110	30	EO	39: TOBE	40: EO?	
100000	32	RN	40: EOR	41: RN?	
100010	34	OT	41: RNO	42: OT?	
000000	0	#			

Σχήμα 1.5: Παράδειγμα πίνακα αποκωδικοποίησης με μέθοδο LZW. [4]

Αξίζει να σημειωθεί ότι το λεξικό της κωδικοποίησης κατασκευάζεται παράλληλα με την διαδικασία κωδικοποίησης των δεδομένων, ενώ επιπλέον, ο κωδικοποιητής LZW κατασκευάζει και ακόμη ένα λεξικό για την αποσυμπίεση, αποκωδικοποιώντας ταυτόχρονα την συμπιεσμένη ροή δεδομένων της εισόδου. Τέλος, χρησιμοποιείται ευρέως σε μια ποικιλία τύπων αρχείων όπως είναι το GIF, PDF και TIFF.

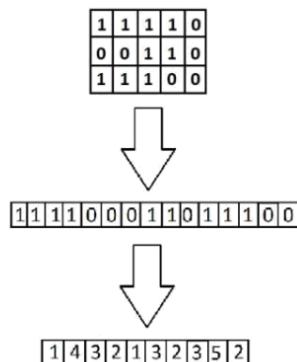
#### 1.2.1.4 Κωδικοποίηση Μήκους Διαδρομής

Η κωδικοποίηση μήκους Διαδρομής (Run Length Encoding, RLC) είναι μια απλή μέθοδος συμπιεσης δεδομένων που είναι ιδιαίτερα χρήσιμη για επαναλαμβανόμενα δεδομένα [36], όπως σε δυαδικές εικόνες ή εικόνες σε κλίμακα του γκρι. Η τεχνική λειτουργεί με την αντικατάσταση ακολουθιών παρόμοιων συμβόλων, επίσης γνωστών ως "διαδρομές", με ένα συντομότερο σύμβολο. Για εικόνες κλίμακας του γκρι, ο κώδικας μήκους διαδρομής αποτελείται από δύο τιμές: την ένταση του εικονοστοιχείου και τον αριθμό των διαδοχικών εικονοστοιχείων που έχουν την ίδια ένταση.

Το πλεονέκτημα του RLC είναι ότι μπορεί να μειώσει σημαντικά το μέγεθος εικόνων με μεγαλύτερες περιοχές και λιγότερες ακμές. Με την αναπαράσταση της έντασης των εικονοστοιχείων με ένα μόνο byte, το συλλογικό μέγεθος της κωδικοποιημένης εικόνας εξαρτάται σε μεγάλο βαθμό

από το περιεχόμενο της αρχικής εικόνας. Ωστόσο, είναι σημαντικό να σημειωθεί ότι η μέθοδος αυτή μπορεί να εφαρμοστεί μόνο σε δυαδικές εικόνες ή εικόνες κλίμακας του γκρι, ενώ για εικόνες RGB, κάθε χρωματικό κανάλι πρέπει να συμπιεστεί ξεχωριστά.

Για να εξασφαλιστεί η αποτελεσματική συμπίεση, πρέπει επίσης να ληφθεί προσεκτικά υπόψη το μοτίβο σάρωσης της εικόνας. Ανάλογα με τη φύση της εικόνας, μπορεί να είναι πιο αποτελεσματική η σάρωση της κάθετα, οριζόντια ή σε μοτίβο ζιγκ-ζαγκ. Εάν το μοτίβο σάρωσης δεν είναι βελτιστοποιημένο, το αποτέλεσμα της κωδικοποίησης μπορεί να καταλήξει να είναι μεγαλύτερο από την αρχική εικόνα.



Σχήμα 1.6: Παράδειγμα εφαρμογής του αλγορίθμου Κωδικοποίησης Μήκους Διαδρομής σε δυαδική εικόνα.

Θα πρέπει επίσης να σημειωθεί ότι η RLC μπορεί να μετατραπεί σε μια αποτελεσματικότερη τεχνική συμπίεσης με απώλειες, εάν η απαίτηση για παρόμοια ένταση είναι πιο χαλαρή. Στην περίπτωση αυτή, η συμπιεσμένη εικόνα μπορεί να μην είναι ακριβές αντίγραφο της αρχικής, αλλά το αντιστάθμισμα είναι ένα σημαντικά μικρότερο μέγεθος αρχείου.

Συνολικά, η κωδικοποίηση μήκους εκτέλεσης είναι μια χρήσιμη και αποτελεσματική μέθοδος συμπίεσης για δυαδικές εικόνες ή εικόνες κλίμακας του γκρι με επαναλαμβανόμενα δεδομένα. Επιλέγοντας προσεκτικά το μοτίβο σάρωσης και λαμβάνοντας υπόψη το επίπεδο ομοιότητας που απαιτείται για τη συμπίεση, η RLC μπορεί να επιτύχει σημαντικές μειώσεις στο μέγεθος του αρχείου.

### 1.2.1.5 Προβλεπτική Κωδικοποίηση

Η προβλεπτική κωδικοποίηση που προτάθηκε πρώτη φορά από τον Cutler το 1952 [37], είναι μια στατιστική τεχνική εκτίμησης που χρησιμοποιεί παρελθοντικές και παρούσες παρατηρήσιμες τυχαίες μεταβλητές για να προβλέψει μελλοντικές τυχαίες μεταβλητές. Ο στόχος είναι να εκτιμηθούν οι μελλοντικές τιμές με όσο το δυνατόν μεγαλύτερη ακρίβεια χρησιμοποιώντας τις πληροφορίες που είναι διαθέσιμες στο παρόν και στο παρελθόν.

Η μέθοδος διαφορικής διαμόρφωσης παλμικού κώδικα χωρίς απώλειες (differential pulse code modulation- DPCM) είναι μια συχνή μέθοδος προβλεπτικής κωδικοποίησης που χρησιμοποιείται για την αποτελεσματική κωδικοποίηση ψηφιακών εικόνων. Σε αυτή την τεχνική, κάθε τιμή εικονοστοιχείου σε μια εικόνα, εκτός από τις τιμές στα όρια των εικόνων, προβλέπεται με βάση τις γειτονικές της τιμές για τη δημιουργία μιας προβλεπόμενης εικόνας. Η διαφορά μεταξύ της πραγματικής και της προβλεπόμενης τιμής του εικονοστοιχείου χρησιμοποιείται για την δημιουργία μιας υπολειμματικής εικόνας.

Η υπολειμματική εικόνα έχει συνήθως πολύ μικρότερο δυναμικό εύρος τιμών εικονοστοιχείων από την αρχική εικόνα. Αυτό σημαίνει ότι μπορεί να συμπιεστεί αποτελεσματικά χρησιμοποιώντας

τεχνικές όπως η κωδικοποίηση Huffman, η οποία μειώνει τον αποθηκευτικό χώρο που απαιτείται για την αποθήκευση της.

### 1.2.1.6 Κωδικοποίηση Περιοχής

Η κωδικοποίηση περιοχής είναι μια προηγμένη μορφή της τεχνικής κωδικοποίησης μήκους διαδρομής (RLE) για τη συμπίεση δεδομένων. Ενώ η συμπίεση με RLE μπορεί να έχει απώλειες, η κωδικοποίηση περιοχής την βελτιώνει παραμένοντας μια μέθοδος συμπίεσης χωρίς απώλειες. Βασίζεται στην ιδέα ότι οι εικόνες είναι δισδιάστατα σήματα και ο αλγόριθμος κωδικοποίησης περιοχής προσπαθεί να εντοπίσει ορθογώνιες περιοχές με ίδια χαρακτηριστικά. Αυτές οι περιοχές στη συνέχεια κωδικοποιούνται ως στοιχεία με δύο σημεία και μια ορισμένη δομή.

Η κωδικοποίηση περιοχής είναι μια αποτελεσματική τεχνική για τη συμπίεση δεδομένων, αλλά έχει το μειονέκτημα ότι χρησιμοποιεί μια μη γραμμική μέθοδο που δεν μπορεί να υλοποιηθεί σε hardware. Ως εκ τούτου, ο χρόνος συμπίεσης μπορεί να μην είναι ανταγωνιστικός σε σύγκριση με άλλες τεχνικές συμπίεσης, αν και ο λόγος συμπίεσης είναι καλός. Με τον εντοπισμό ορθογώνιων περιοχών με παρόμοια χαρακτηριστικά, η κωδικοποίηση περιοχής μπορεί να μειώσει τον όγκο των δεδομένων που απαιτούνται για την αναπαράσταση μιας εικόνας, με αποτέλεσμα μικρότερο μέγεθος αρχείου.

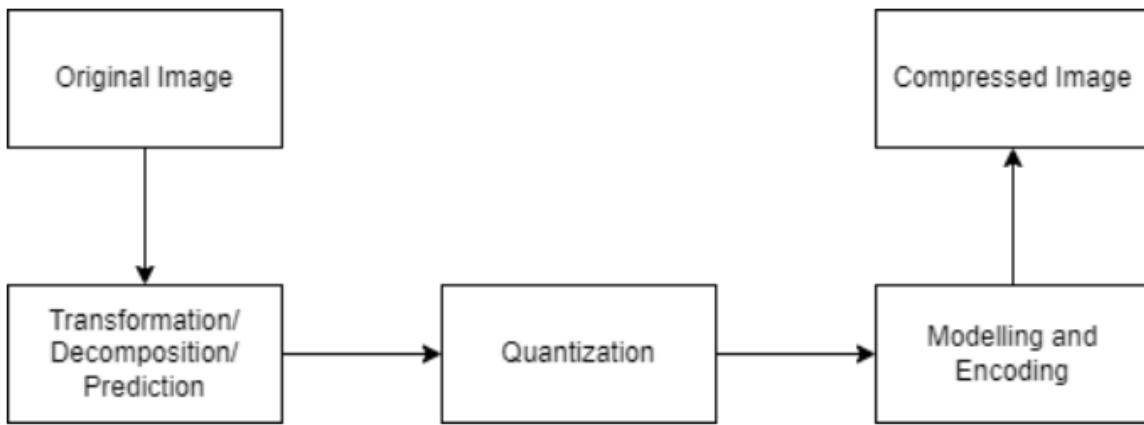
### 1.2.2 Συμπίεση με Απώλειες

Η συμπίεση με απώλειες είναι ένας τύπος συμπίεσης δεδομένων που περιλαμβάνει την μείωση του μεγέθους του αρχείου με κόστος κάποια απώλεια πληροφοριών. Όταν τα δεδομένα συμπιέζονται χρησιμοποιώντας τεχνικές με απώλειες, κάποια από τα αρχικά δεδομένα αφαιρούνται ή αλλοιώνονται μόνιμα και τα συμπιεσμένα δεδομένα δεν μπορούν να ανακατασκευαστούν ακριβώς όπως ήταν πριν από τη συμπίεση. Ο βαθμός απώλειας πληροφοριών εξαρτάται από τον αλγόριθμο συμπίεσης που χρησιμοποιείται και το επίπεδο συμπίεσης που εφαρμόζεται.

Το πλεονέκτημα που φέρουν οι τεχνικές συμπίεσης με απώλειες είναι η επίτευξη υψηλότερων λόγων συμπίεσης σε σχέση με τις τεχνικές που αναφέρθηκαν στον προηγούμενο υποκεφάλαιο. Ως αποτέλεσμα, το συμπιεσμένο αρχείο καταλαμβάνει λιγότερο χώρο αποθήκευσης και μπορεί να μεταδοθεί ταχύτερα μέσω ενός δικτύου.

Οι τεχνικές συμπίεσης με απώλειες χρησιμοποιούν συνήθως έναν συνδυασμό πρόβλεψης, μετασχηματισμού και αποσύνθεσης για την αφαίρεση περιττών ή περιττών πληροφοριών από τα δεδομένα. Η κράντιση είναι μια διαδικασία που χρησιμοποιείται στη συμπίεση με απώλειες, η οποία μειώνει τον όγκο των δεδομένων στρογγυλοποιώντας τις τιμές στο πλησιέστερο επίπεδο που μπορεί να αναπαρασταθεί χρησιμοποιώντας λιγότερα bits. Αυτή η διαδικασία εισάγει πρόσθετη παραμόρφωση ή σφάλμα στα δεδομένα.

Μετά την κράντιση, τα υπόλοιπα δεδομένα συμπιέζονται στη συνέχεια με τη χρήση κωδικοποίησης εντροπίας, η οποία είναι μια τεχνική συμπίεσης χωρίς απώλειες που μπορεί να αντιστραφεί για την ανάκτηση των συμπιεσμένων δεδομένων. Κατά την διαδικασία της αποκωδικοποίησης, τιμές του συμπιεσμένου σήματος αποκωδικοποιούνται για την ανάκτηση των κραντισμένων τιμών του σήματος. Στη συνέχεια, χρησιμοποιείται η διαδικασία απο-κραντισμού για την ανάκτηση των αρχικών τιμών σήματος από τις κραντισμένες, οι οποίες μπορεί να εξακολουθούν να περιέχουν κάποια παραμόρφωση ή απώλεια πληροφοριών. [38]



Σχήμα 1.7: Διάγραμμα ροής της συμπίεσης με απώλειες.

Κατά την συμπίεση με απώλειες, υπάρχουν οι παρακάτω σημαντικές μετρητικές που πρέπει να έχουμε υπ' όψην:

- Λόγος συμπίεσης
- Λόγος σίματος προς θόρυβο
- Ταχύτητα κωδικοποίησης και αποκωδικοποίησης

Μερικές από τις μεθόδους που χρησιμοποιούνται συχνότερα είναι:

- Κωδικοποίηση Μετασχηματισμού
- Κωδικοποίηση Αποκοπής Μπλοκ (Block Truncation Coding)
- Μορφοκλασματική Κωδικοποίηση (Fractal Coding)

### 1.2.2.1 Κωδικοποίηση Μετασχηματισμού

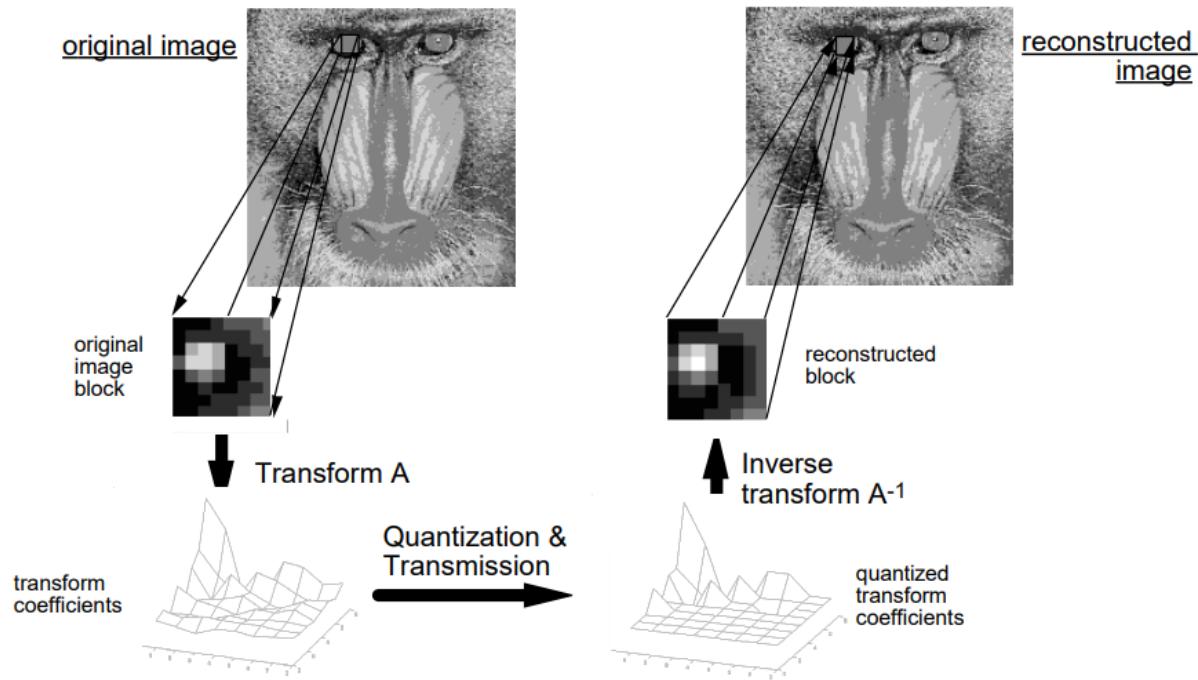
Στην κωδικοποίηση μετασχηματισμού, μια εικόνα υφίσταται μετασχηματισμό από ένα πεδίο, συνήθως χωρικό ή χρονικό, σε μια διαφορετική αναπαράσταση με τη χρήση ενός γνωστού μετασχηματισμού. Οι προκύπτουσες μετασχηματισμένες τιμές στη συνέχεια κωδικοποιούνται, με αποτέλεσμα υψηλή συμπίεση δεδομένων. Η κωδικοποίηση μετασχηματισμού (TC) είναι μια εξαιρετικά αποτελεσματική τεχνική κωδικοποίησης που αξιοποιεί τη συσχέτιση μεταξύ των εικονοστοιχείων. Χρησιμοποιεί το πεδίο συχνότητας στο οποίο το σύστημα κωδικοποίησης, μέσω μιας συνάρτησης μετασχηματισμού, μετατρέπει αρχικά τα εικονοστοιχεία από το πεδίο του χώρου, παράγοντας ένα σύνολο φασματικών συντελεστών που κωδικοποιούνται κατάλληλα και μεταδίδονται.

Ωστόσο, η ίδια η λειτουργία μετασχηματισμού δεν επιτυγχάνει καμία συμπίεση. Ο πρωταρχικός της στόχος είναι να αποσυσχετίσει τα αρχικά δεδομένα και να συμπιέσει ένα μεγάλο μέρος της ενέργειας του σίματος σε ένα πολύ μικρότερο σύνολο συντελεστών του μετασχηματισμού. Αυτό επιτρέπει την απόρριψη πολλών συντελεστών μετά τον κβαντισμό και πριν από την κωδικοποίηση.

Οι αλγόριθμοι κωδικοποίησης μετασχηματισμού ξεκινούν συνήθως με την κατάτμηση της αρχικής εικόνας σε μικρότερα μπλοκ, συνήθως  $8 \times 8$ . Στη συνέχεια υπολογίζονται οι συντελεστές μετασχηματισμού για κάθε μπλοκ, μετατρέποντας ουσιαστικά τον αρχικό πίνακα τιμών εικονοστοιχείων σε πίνακα συντελεστών. Οι συντελεστές που βρίσκονται πιο κοντά στην πάνω αριστερή γωνία περιέχουν συνήθως τις περισσότερες πληροφορίες που απαιτούνται για την κβάντιση και την

κωδικοποίηση της εικόνας με μικρή αντιληπτική παραμόρφωση. Οι συντελεστές που προκύπτουν στη συνέχεια κβαντίζονται και η έξοδος του κβαντιστή χρησιμοποιείται από μια τεχνική κωδικοποίησης συμβόλων για την παραγωγή της ροής bit εξόδου που αντιπροσωπεύει την κωδικοποιημένη εικόνα. [39]

## Principle of block-wise transform coding



Σχήμα 1.8: Παράδειγμα κωδικοποίησης μετασχηματισμού σε εικόνα. [5]

Η μέθοδος αυτή περιλαμβάνει τη μέση τιμή των χρωμάτων που χρησιμοποιούνται σε μια εικόνα, έτσι ώστε να χρησιμοποιούνται συνολικά λιγότερα χρώματα. Δεδομένου ότι το ανθρώπινο μάτι αντιλαμβάνεται ορισμένα χρώματα πιο έντονα από άλλα, η ροή στης της μεθόδου δεν είναι συνήθως αντιληπτή από τους περισσότερους ανθρώπους. Η κωδικοποίηση μετασχηματισμού χρησιμοποιείται τόσο στην συμπίεση JPEG όσο και στη JPEG2000, χρησιμοποιώντας τον διακριτό μετασχηματισμό συνημιτόνου και τον διακριτό μετασχηματισμό κυματιδίου, αντίστοιχα.

### 1.2.2.2 Κωδικοποίηση Αποκοπής Μπλοκ

Η κωδικοποίηση αποκοπής μπλοκ (BTC) είναι μια τεχνική συμπίεσης εικόνων σε κλίμακα του γκρι, η οποία έχει επίσης επεκταθεί για συμπίεση έγχρωμων εικόνων και βίντεο. Η μέθοδος περιλαμβάνει το σπάσιμο των φωτογραφιών εισόδου σε μπλοκ και στη συνέχεια τη ροή στην ενός κβαντιστή για την ελαχιστοποίηση του αριθμού των επιπέδων γκρι σε κάθε μπλοκ, διατηρώντας σταθερές τη μέση τιμή και την τυπική απόκλιση. [40] Η διαδικασία έχει ως εξής:

Μια εικόνα χωρίζεται σε μη επικαλυπτόμενα μπλοκ, τα οποία έχουν συνήθως μέγεθος  $4 \times 4$  pixel. Για κάθε μπλοκ, υπολογίζεται ο μέσος όρος και η τυπική απόκλιση των τιμών των εικονοστοιχείων του. Όπως είναι λογικό, οι τιμές αυτές μπορεί να διαφέρουν από μπλοκ σε μπλοκ.

Στη συνέχεια, οι τιμές των εικονοστοιχείων κάθε νέου μπλοκ καθορίζονται έτσι ώστε κάθε μπλοκ της συμπιεσμένης εικόνας BTC να έχει περίπου την ίδια μέση τιμή και τυπική απόκλιση με το αντίστοιχο μπλοκ της αρχικής εικόνας. Εάν μια τιμή εικονοστοιχείου είναι μεγαλύτερη από τη

μέση τιμή, αποδίδεται η τιμή "1"- διαφορετικά, αποδίδεται η τιμή "0". Τιμές ίσες με τη μέση τιμή μπορεί να έχουν "1" ή "0" ανάλογα με την επιλογή του χρήστη.

Τέλος, αυτό το μπλοκ των 16 bit αποθηκεύεται ή μεταδίδεται μαζί με τις τιμές του μέσου όρου και της τυπικής απόκλισης. Η μέση τιμή και η τυπική απόκλιση διατηρούνται με τη χρήση των τιμών "a" και "b" στην ανακατασκευή. Οι τιμές "a" και "b" υπολογίζονται με τη χρήση συγκεκριμένων τύπων. Για την ανακατασκευή της εικόνας, τα στοιχεία με 0 αντικαθίστανται με την τιμή "a", ενώ εκείνα με 1 αντικαθίστανται με την τιμή "b". Προκύπτει ότι ο αλγόριθμος είναι ασύμμετρος, με τον κωδικοποιητή να αναλαμβάνει μεγαλύτερο φόρτο εργασίας από τον αποκωδικοποιητή. Ο αποκωδικοποιητής απλώς αντικαθιστά τα 1 και τα 0 με την εκτιμώμενη τιμή, ωστόσο ο κωδικοποιητής πρέπει επίσης να υπολογίζει τη μέση τιμή, την τυπική απόκλιση και δύο τιμές "a" και "b".

### 1.2.2.3 Μορφοκλασματική Κωδικοποίηση

Η μορφοκλασματική κωδικοποίηση (Fractal Coding) είναι μια μέθοδος επεξεργασίας εικόνας που βασίζεται στην μορφοκλασματική γεωμετρία, έναν μοναδικό τύπο γραφικής αναπαράστασης μαθηματικών συναρτήσεων ή συνόλων αριθμών που παράγονται από μαθηματικές συναρτήσεις. Η βασική ιδέα που χρησιμοποιεί η μορφοκλασματική κωδικοποίηση για τη συμπίεση εικόνας είναι ότι τιμήματα μιας εικόνας μοιάζουν με άλλα τιμήματα της ίδιας εικόνας.

Το Σύστημα Επαναληπτικών Συναρτήσεων (IFS) και το Θεώρημα Κολάζ (Collage Theorem) είναι τα δύο κύρια κοινάτια της μορφοκλασματικής κωδικοποίησης.

**Επαναληπτικό σύστημα συναρτήσεων (IFS):** Πρόκειται για έναν τρόπο δημιουργίας κλασμάτων (fractals) τα οποία συχνά είναι παρόμοια μεταξύ τους. Τα fractals IFS, όπως είναι συνήθως γνωστά, μπορούν να έχουν οποιονδήποτε αριθμό διαστάσεων, αλλά συνήθως υπολογίζονται και απεικονίζονται σε δύο διαστάσεις.

**Θεώρημα κολάζ:** Αυτό το θεώρημα είναι το θεμέλιο για τη συμπίεση στην μορφοκλασματική κωδικοποίηση. Ουσιαστικά δηλώνει ότι αν μπορεί να βρεθεί ένας συσταλτικός μετασχηματισμός (contractive transform) με έξοδο κοντά στην είσοδο, τότε το σταθερό σημείο αυτού του μετασχηματισμού θα είναι επίσης κοντά στην αρχική είσοδο.

Η κωδικοποίηση μορφοκλασματικής εικόνας αρχίζει με την κατάτμηση της εικόνας σε υπομπλοκ γνωστά ως μπλοκ εύρους και μπλοκ πεδίου. Για κάθε μπλοκ εύρους, η μέθοδος αναζητά ένα μπλοκ πεδίου που είναι παρόμοιο αλλά μεγαλύτερο. Στη συνέχεια αναζητά έναν μαθηματικό μετασχηματισμό που θα μειώσει το μπλοκ πεδίου ώστε να μοιάζει με το μπλοκ εύρους. Οι παραμέτροι αυτών των μετασχηματισμών αποθηκεύονται και αναπαριστούν την συμπιεσμένη αναπαράσταση της εικόνας. [41]

## 1.3 Το Πρότυπο JPEG

Το πρότυπο JPEG, είναι μια μέθοδος συμπίεσης με απώλειες που χρησιμοποιείται ευρέως για ψηφιακές εικόνες. Δημιουργήθηκε το 1992 και παραμένει το πιο συχνά χρησιμοποιούμενο πρότυπο συμπίεσης εικόνων στον κόσμο [42]. Ο βαθμός συμπίεσης είναι ρυθμιζόμενος, παρέχοντας επιλέξιμο βαθμό του λόγου μεταξύ του μεγέθους αποθηκεύσης και της ποιότητας της εικόνας. Σε γενικές γραμμές, το JPEG επιτυγχάνει συμπίεση 10:1 με μικρή απώλεια στην ποιότητα της εικόνας.

Το πρότυπο JPEG χρησιμοποιεί τον διακριτό μετασχηματισμό συνημιτόνου ως τον τρόπο συμπίεσης των δεδομένων. Η αποτελεσματικότητα της συμπίεσης με αυτή την μέθοδο, βασίζεται σε τρεις σημαντικές παρατηρήσεις. Πρώτον, τα χρήσιμα περιεχόμενα της εικόνας τείνουν να μεταβάλλονται σχετικά αργά κατά την εικόνα, καθιστώντας ασυνήθιστο οι τιμές έντασης να μεταβάλλονται σε μεγάλο βαθμό πολλές φορές μέσα σε ένα μπλοκ εικόνας  $8 \times 8$ . Δεύτερον, πειράματα έχουν δείξει ότι οι άνθρωποι είναι πολύ λιγότερο πιθανό να αντιληφθούν την απώλεια συνιστωσών πολύ

υψηλής χωρικής συχνότητας από ότι την απώλεια συνιστωσών χαμηλότερης συχνότητας. Τέλος, η οπτική οξύτητα είναι πολύ μεγαλύτερη για το γκρι ("ασπρόμαυρο") από ότι για το χρώμα.

### 1.3.1 Τα Μοντέλα JPEG Κωδικοποίησης

Το πρότυπο JPEG είναι μια ευέλικτη τεχνική συμπίεσης εικόνας που προσφέρει τέσσερις διαφορετικούς τρόπους λειτουργίας. Οι τρόποι αυτοί είναι η διαδοχική κωδικοποίηση, η προοδευτική κωδικοποίηση, η κωδικοποίηση χωρίς απώλειες και η ιεραρχική κωδικοποίηση, ο καθένας με το δικό του σύνολο ζευγών κωδικοποιητή/αποκωδικοποιητή, γνωστών και ως codecs.

Ο πρώτος τρόπος λειτουργίας, η διαδοχική κωδικοποίηση, είναι μια απλή μέθοδος που κωδικοποιεί τις εικόνες από πάνω προς τα κάτω. Αυτός ο τρόπος υποστηρίζει δεδομένα δείγματος 8 και 12 bit και κάθε χρωματικό στοιχείο κωδικοποιείται εξ ολοκλήρου από μία μόνο σάρωση. Στο πλαίσιο της διαδοχικής λειτουργίας, το πρότυπο JPEG ορίζει δύο εναλλακτικές διαδικασίες κωδικοποίησης εντοπίσιας: Η κωδικοποίηση Huffman και η αριθμητική κωδικοποίηση. Αυτές οι διαδικασίες καθορίζουν τον τρόπο συμπίεσης και αποσυμπίεσης των δεδομένων της εικόνας.

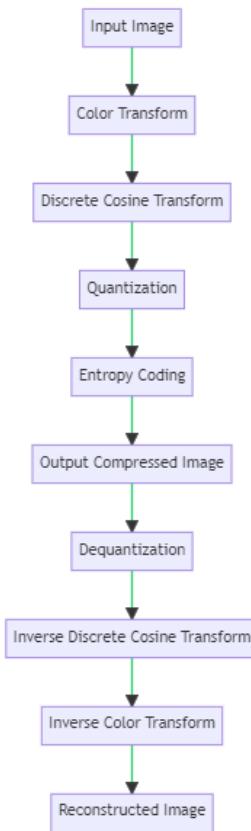
Η δεύτερη λειτουργία, η προοδευτική κωδικοποίηση, είναι πιο πολύπλοκη από τη διαδοχική κωδικοποίηση. Στις προοδευτικές εικόνες JPEG, τα στοιχεία κωδικοποιούνται σε πολλαπλές σαρώσεις, με ελάχιστο αριθμό το 2 και μέγιστο αριθμό 896 σαρώσεων. Οι αρχικές σαρώσεις δημιουργούν μια πρόχειρη έκδοση της εικόνας, ενώ οι επόμενες σαρώσεις την βελτιώνουν.

Η τρίτη λειτουργία, η κωδικοποίηση χωρίς απώλειες, διατηρεί ακριβή την αρχική εικόνα. Ωστόσο, δεν χρησιμοποιείται συχνά καθώς χαρακτηρίζεται από μικρό λόγο συμπίεσης.

Ο τέταρτος τρόπος, η ιεραρχική κωδικοποίηση, είναι ένας υπερπροοδευτικός τρόπος στον οποίο η εικόνα αναλύεται σε διάφορες υποεικόνες που ονομάζονται πλαίσια. Ένα καρέ είναι μια συλλογή από μία ή περισσότερες σαρώσεις. Στην ιεραρχική λειτουργία, το πρώτο καρέ δημιουργεί μια έκδοση της εικόνας με χαμηλή ανάλυση και τα υπόλοιπα καρέ βελτιώνουν την εικόνα αυξάνοντας την ανάλυση.

### 1.3.2 Ο Αλγόριθμος Κωδικοποίησης JPEG

Το πρότυπο JPEG διαθέτει ένα "βασικό" σχήμα συμπίεσης γνωστό ως βασική διαδοχική κωδικοποίηση, το οποίο είναι κατάλληλο για χρήση σε απλές εφαρμογές. Ως εκ τούτου, αυτή η μέθοδος θα αποτελέσει και το επίκεντρο για την ανάλυση του αλγορίθμου καθώς εξηγεί τη βασική ιδέα πίσω από τους περισσότερους τρόπους λειτουργίας του JPEG. Η διαδικασία συμπίεσης και αποσυμπίεσης απεικονίζεται στο παρακάτω σχήμα: *insertimage*



Σχήμα 1.9: Διαδικασία συμπίεσης και αποσυμπίεσης εικόνας με JPEG.

### 1.3.2.1 Μετατροπή Χρωματικού Χώρου

Ο αλγόριθμος συμπίεσης JPEG ξεκινά με τη μετατροπή του χρωματικού χώρου της εικόνας, η οποία περιλαμβάνει τη μετάβαση από τον χρωματικό χώρο RGB στον χώρο YUV (YCbCr). Η φωτεινότητα αντιπροσωπεύεται από το Y, ενώ τα Cb και Cr αντιπροσωπεύουν τη χρωματική ένταση. Ο χρωματικός χώρος RGB χρησιμοποιεί τη χρωματική απεικόνιση για την ανάμειξη των χαρακτηριστικών της φωτεινότητας και της χρωμικότητας, αλλά ο διαχωρισμός του χρώματος στις συνιστώσες φωτεινότητας και χρωμικότητας επιτρέπει την αποτελεσματικότερη επεξεργασία και μετάδοση των έγχρωμων σημάτων σε διάφορες εφαρμογές. Για την επίτευξη αυτού του στόχου, αναπτύχθηκαν χρωματικές συντεταγμένες τριών συνιστώσων, που η μία συνιστώσα (Y) αντιπροσωπεύει τη φωτεινότητα, ενώ οι άλλες δύο (Cb, Cr) αντικατοπτρίζουν την απόχρωση και τον κορεσμό (χρωματική ένταση) συλλογικά.

Οι τιμές [Y Cb Cr] στη συντεταγμένη YUV συνδέονται μαθηματικά με τις τιμές [R G B] στη συντεταγμένη RGB μέσω των παρακάτω εξισώσεων [43].

$$Y = 0.299R + 0.587G + 0.114B$$

$$Cb = -0.147R - 0.289G + 0.436B$$

$$Cr = 0.615R - 0.515G - 0.100B$$

$$R = Y + 1.140Cr$$

$$G = Y - 0.395Cb - 0.581Cr$$

$$B = Y + 2.032Cb$$

### 1.3.2.2 Διακριτός Μετασχηματισμός Συνημιτόνου

Το επόμενο βήμα του αλγορίθμου είναι η κωδικοποίηση μετασχηματισμού. Συγκεκριμένα, μετά την μετατροπή του χρωματικού χώρου, χρησιμοποιούμε τον διακριτό μετασχηματισμό συνημιτόνου (Discrete cosign transform) ως την μέθοδο κωδικοποίησης [44]. Ο αλγόριθμος που ακολουθείται είναι ο παρακάτω:

1. Διαχωρισμός της εικόνας εισόδου σε μπλοκ συγκεκριμένου μεγέθους. Το μέγεθος αυτών μπορεί να ποικίλλει αλλά συνηθίζεται η επιλογή μπλοκ NxN.
2. Για κάθε μπλοκ, αφαιρείται η μέση τιμή των εικονοστοιχείων από κάθε ένα από αυτά. Η διαδικασία αυτή αφαιρεί το DC στοιχείο το οποίο αντιπροσωπεύει την μέση τιμή της φωτεινότητας του έκαστου μπλοκ.
3. Εφαρμογή του τύπου του διακριτού μετασχηματισμού συνημιτόνου στο κάθε μπλοκ. Κατά αυτό τον τρόπο δημιουργούμε ένα σετ από συντελεστές. Η φόρμουλα του DCT υπολογίζει το συνημίτονο διαφορετικών συχνοτήτων σε διαφορετικές θέσεις του μπλοκ. Ο τύπος δίνεται από τις παρακάτω εξισώσεις:

$$F(u, v) = \frac{2}{N} + C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos \left[ \frac{(2x+1)u\pi}{2N} \right] \cos \left[ \frac{(2y+1)v\pi}{2N} \right]$$

για  $u=0,1\dots N-1$  και  $v=0,1\dots N-1$ , που  $N=8$  και

$$C(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \\ 1, & \text{αλλού} \end{cases}$$

$$f(x, y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cos \left[ \frac{(2x+1)u\pi}{2N} \right] \cos \left[ \frac{(2y+1)v\pi}{2N} \right]$$

για  $u=0,1\dots N-1$  και  $v=0,1\dots N-1$ , που  $N=8$  και

$$C(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \\ 1, & \text{αλλού} \end{cases}$$

### 1.3.2.3 Κβάντιση Συντελεστών

Στην συνέχεια οι συντελεστές διαιρούνται με ένα σύνολο προκαθορισμένων τιμών ο οποίος ονομάζεται πίνακας κβαντισμού. Αυτή η διαδικασία απαλείφει τις συνιστώσες υψηλής συχνότητας με μικρά πλάτη, οι οποίες είναι λιγότερο σημαντικές για την οπτική ποιότητα της εικόνας. Η παραπάνω διαδικασία ονομάζεται κβαντισμός.

### 1.3.2.4 Κωδικοποίηση Huffman

Η κωδικοποίηση εντροπίας είναι το τελευταίο βήμα της συμπίεσης JPEG. Με την χρήση της, καταφέρνουμε υψηλότερη συμπίεση δίχως απώλειες συμπιέζοντας τους κβαντισμένους συντελεστές του DCT. Το πρότυπο JPEG επιτρέπει τόσο την κωδικοποίηση Huffman όσο και την αριθμητική κωδικοποίηση για αυτό το βήμα. Ο βασικός διαδοχικός κωδικοποιητής χρησιμοποιεί μόνο την κωδικοποίηση Huffman, αλλά οι άλλοι τρόποι λειτουργίας μπορούν να χρησιμοποιήσουν και τα δύο. Η κωδικοποίηση Huffman λειτουργεί καλά για σύμβολα πηγής που έχουν διαφορετικές πιθανότητες. Αναθέτει μικρότερες δυαδικές συμβολοσειρές σε πιο συχνά σύμβολα με βάση τον πίνακα των συχνοτήτων τους. Αυτή η μέθοδος έχει ήδη περιγραφεί νωρίτερα σε αυτό το κεφάλαιο, οπότε δεν θα την επαναλάβουμε εδώ. Η τελική συμπιεσμένη εικόνα είναι η έξοδος αυτού του βήματος. [44]

### 1.3.3 Κριτήρια πιστότητας

Όπως προαναφέρθηκε, κατά την συμπίεση μιας εικόνας μειώνεται το πλήθος των δεδομένων που είναι απαραίτητα για την αναπαράσταση της. Συνεπώς, κατά αυτή την διαδικασία γίνεται απομάκρυνση "άσχετης" οπτικής πληροφορίας από την εικόνα η οποία θα αξιολογηθεί με βάση τα κριτήρια πιστότητας. Τα κριτήρια πιστότητας εικόνας είναι μέθοδοι που χρησιμοποιούνται για την αξιολόγηση της ποιότητας των ψηφιακών εικόνων. Η ποιότητα της εικόνας μπορεί να υποβαθμιστεί λόγω παραμορφώσεων κατά την παραλαβή και την επεξεργασία της ενώ μερικά παραδείγματα παραμόρφωσης είναι ο θόρυβος, το θόλωμα και ο δακτύλιος. [45] Συγκεκριμένα, τα κριτήρια αυτά μπορούν να παρακολουθούν τα μη αντιληπτά σφάλματα καθώς διαδίδονται κατά την επεξεργασία της εικόνας και μπορούν να χρησιμοποιηθούν για τη σύγκριση αλγορίθμων επεξεργασίας εικόνας [45]. Υπάρχουν δύο τύποι κριτηρίων για αξιολόγηση ψηφιακών εικόνων:

- Αντικειμενικά κριτήρια πιστότητας

Όταν η απώλεια της πληροφορίας μπορεί να εκφραστεί με μία συνάρτηση της εισόδου και της εξόδου, χρησιμοποιούνται τα αντικειμενικά κριτήρια πιστότητας

- Υποκειμενικά κριτήρια πιστότητας

Τα αντικειμενικά κριτήρια πιστότητας αποτελούν ένα εύκολο και ακριβή τρόπο αξιολόγησης της συμπιεσμένης εικόνας. Ωστόσο, σε τελική ανάλυση, αυτές οι εικόνες εξετάζονται από ανθρώπους. Για αυτό τον λόγο υπάρχουν τα υποκειμενικά κριτήρια πιστότητας στην οποία παραπορτές καταμετρούν την τελική ποιότητα της εικόνας.

Μερικά από τα πιο δημοφιλή κριτήρια πιστότητας που χρησιμοποιούνται στην συμπίεση εικόνας είναι τα παρακάτω:

#### 1.3.3.1 Μέσο Τετραγωνικό Σφάλμα (MSE)

Ας υποθέσουμε ότι έχουμε μια εικόνα  $f(x, y)$  και την συμπιεσμένη της μορφή  $g(x, y)$ . Για κάθε τιμή των παραμέτρων  $x$  και  $y$ , το σφάλμα μεταξύ των δύο εικόνων δίνεται ως:

$$e(x, y) = g(x, y) - f(x, y)$$

Συνεπώς, το συνολικό σφάλμα μεταξύ των δύο εικόνων είναι:

$$\sum_{x=1}^M \sum_{y=1}^N g(x, y) - f(x, y)$$

Τέλος, το συνολικό μέσο τετραγωνικό σφάλμα δίνεται:

$$MSE = \frac{1}{MN} \sum_{x=1}^M \sum_{y=1}^N (g(x, y) - f(x, y))^2$$

που  $f$  είναι η αρχική εικόνα,  $g$  είναι η επεξεργασμένη εικόνα,  $M$  είναι το ύψος της εικόνας και  $N$  είναι το πλάτος της εικόνας. Συνοψίζοντας, το MSE υπολογίζεται λαμβάνοντας το άθροισμα των τετραγωνικών διαφορών μεταξύ των εικονοστοιχείων της αρχικής και της επεξεργασμένης εικόνας και διαιρώντας το με το συνολικό αριθμό των εικονοστοιχείων της εικόνας. Η τιμή αυτή υπολογίζεται στη συνέχεια ως μέσος όρος ώστε να προκύψει το μέσο τετραγωνικό σφάλμα.

### 1.3.3.2 Peak Signal-To-Noise Ratio (PSNR)

Το PSNR είναι ακόμη μια τεχνική για τον υπολογισμό της ποιότητας των ψηφιακών εικόνων και πρόκειται για τον λόγο της μέγιστης ισχύος προς την ισχύ του θορύβου της εικόνας. Χρησιμοποιείται συνήθως για την αξιολόγηση της ποιότητας της συμπίεσης εικόνας με απώλειες, που οι ανώτερες τιμές υποδηλώνουν υψηλότερη ποιότητα στην εξεταζόμενη εικόνα. Ο τρόπος υπολογισμού του είναι:

$$PSNR = 10 \log_{10} \left( \frac{MAX_I^2}{MSE} \right)$$

που  $MAX_I$  είναι η μέγιστη δυνατή τιμή εικονοστοιχείου της εικόνας και  $MSE$  είναι το μέσο τετραγωνικό σφάλμα που περιγράφεται παραπάνω. Η τιμή του  $MAX_I$  είναι συνήθως 255 για εικόνες κλίμακας του γκρι 8-bit και 65535 για εικόνες κλίμακας του γκρι 16-bit.

### 1.3.3.3 Δομική Ομοιότητα (SSIM)

Η δομική ομοιότητα ή SSIM είναι ένα κριτήριο βασιζόμενο στην αντίληψη, οπού θεωρεί την υποβάθμιση της εικόνας ως αντιληπτή αλλαγή στη δομική πληροφορία, ενώ ενσωματώνει επίσης σημαντικά φαινόμενα μεταξύ των οποίων είναι η μάσκα φωτεινότητας και αντίθεσης.

Η διαφορά με άλλες τεχνικές όπως το MSE ή το PSNR που εκτιμούν τα απόλυτα σφάλματα είναι η ιδέα ότι τα εικονοστοιχεία έχουν ισχυρές αλληλεξαρτήσεις, ιδίως όταν βρίσκονται σε χωρική εγγύτητα και μεταφέρουν σημαντικές οπτικές πληροφορίες σχετικά με τη δομή των αντικειμένων. Η μάσκα φωτεινότητας είναι ένα φαινόμενο σύμφωνα με το οποίο οι παραμορφώσεις της εικόνας τείνουν να είναι λιγότερο ορατές σε φωτεινές περιοχές, ενώ η μάσκα αντίθεσης είναι ένα φαινόμενο σύμφωνα με το οποίο οι παραμορφώσεις γίνονται λιγότερο ορατές εκεί που υπάρχει σημαντική δραστηριότητα ή "υφή" στην εικόνα.

Η δομική ομοιότητα χρησιμοποιείται συχνά για την αξιολόγηση της ποιότητας της συμπίεσης εικόνας με απώλειες, που ο στόχος είναι η μείωση του μεγέθους της εικόνας με ταυτόχρονη ελαχιστοποίηση της απώλειας ποιότητας. Ο τύπος για τον υπολογισμό του SSIM μιας εικόνας δίνεται από:

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

που  $x$  και  $y$  είναι η αρχική και η επεξεργασμένη εικόνα, αντίστοιχα,  $\mu$  είναι ο μέσος όρος της εικόνας,  $\sigma$  είναι η τυπική απόκλιση της εικόνας και  $c_1$  και  $c_2$  είναι σταθερές. Οι σταθερές  $c_1$  και  $c_2$  ορίζονται ως εξής:

$$c_1 = (k_1 L)^2$$

$$c_2 = (k_2 L)^2$$

που  $k_1$  και  $k_2$  είναι σταθερές που εξαρτώνται από το δυναμικό εύρος της εικόνας και  $L$  είναι ο αριθμός των πιθανών τιμών εικονοστοιχείου στην εικόνα.

Οι τιμές των  $c_1$  και  $c_2$  χρησιμοποιούνται για να μην γίνει ο παρονομαστής του τύπου SSIM πολύ μικρός, γεγονός που μπορεί να προκαλέσει μεγάλα σφάλματα στα αποτελέσματα. Με την προσθήκη αυτών των σταθερών, ο παρονομαστής είναι πάντα τουλάχιστον τόσο μεγάλος όσο οι ίδιες οι σταθερές, εξασφαλίζοντας ότι η διαίρεση είναι πάντα καλά ορισμένη. Τέλος, με βάση τον τύπο:

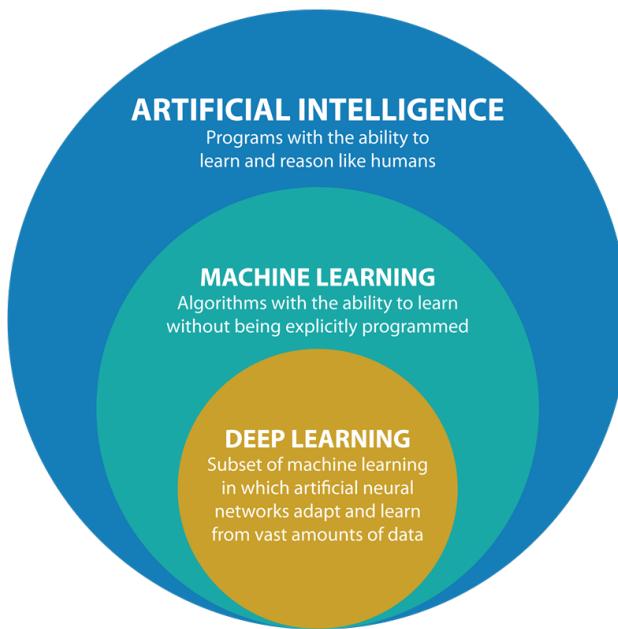
$$MSSIM(x, y) = \frac{1}{N} \sum_{i=1}^N SSIM(x_i, y_i)$$

που  $x$  και  $y$  είναι η αρχική και η επεξεργασμένη εικόνα αντίστοιχα,  $N$  είναι ο αριθμός των παραμέτρων στην εικόνα και  $SSIM(x_i, y_i)$  είναι η τιμή SSIM για το iοστό παράμυθο, υπολογίζεται η συνολική δομική ομοιότητα των δύο εικόνων.

## 2. Μηχανική Μάθηση

### 2.1 Εισαγωγή

Η μηχανική μάθηση (Machine Learning) αποτελεί έναν από τους πλέον ταχύτατα αναπτυσσόμενους τομείς της Τεχνητής Νοημοσύνης. Πρόκειται για το πεδίο έρευνας αφοσιωμένο στην κατανόηση και κατασκευή μεθόδων οι οποίες εκμεταλλευόμενες δεδομένα καταφέρνουν να μαθαίνουν, να πραγματοποιούν προβλέψεις και να λαμβάνουν αποφάσεις χωρίς να έχουν προγραμματιστεί ρητά στο να το κάνουν. Για αυτό τον λόγο είναι και συχνή η συσχέτιση τέτοιων αλγορίθμων με τους νευρώνες του εγκεφάλου καθώς επιχειρείται η προσομοίωση των ενεργειών που εκτελεί ο ανθρώπινος νους όταν δέχεται κάποιο εξωτερικό ερέθισμα.

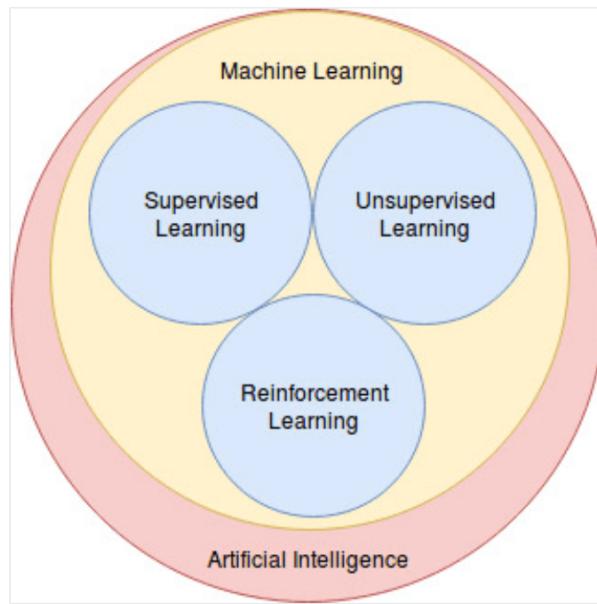


Σχήμα 2.1: Διάγραμμα Βεν τεχνητής νοημοσύνης, μηχανικής μάθησης, βαθιάς μάθησης. [6]

### 2.2 Μηχανική Μάθηση

Τα προβλήματα στην μηχανική μάθηση χωρίζονται σε 4 κατηγορίες, ανάλογα με τον τρόπο που παρέχονται τα δεδομένα προς εκμάθηση και θα αναλυθούν παρακάτω [8].

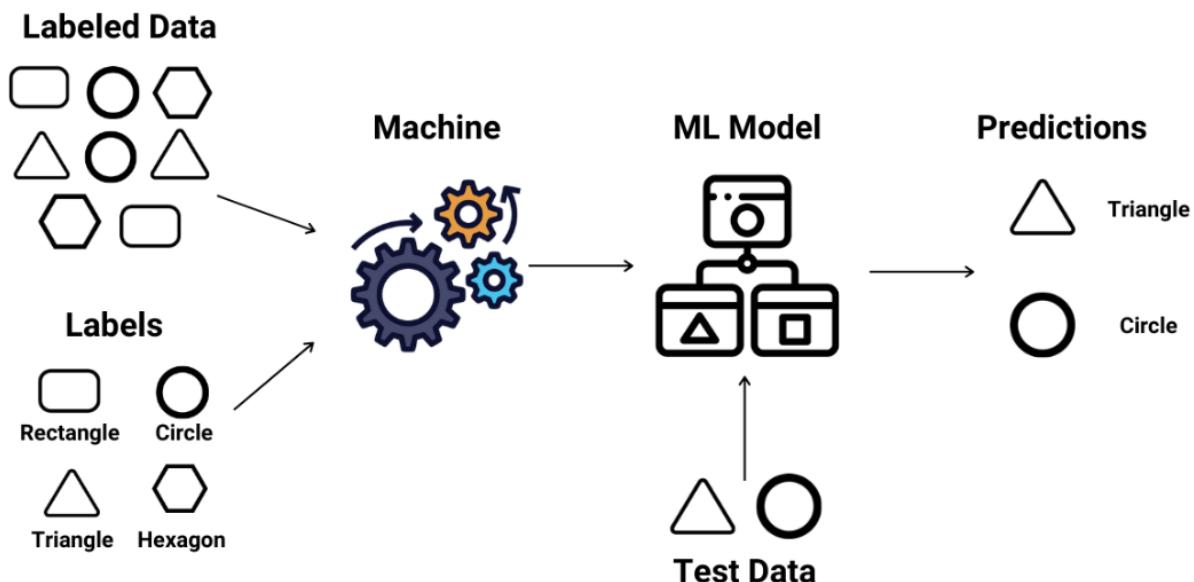
1. Επιβλεπόμενη Μάθηση (Supervised Learning)
2. Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)
3. Ημι-Επιβλεπόμενη Μάθηση (Semi-Supervised Learning)
4. Ενισχυτική Μάθηση (Reinforcement Learning)



Σχήμα 2.2: Ο κύκλος της τεχνητής νοημοσύνης. [7]

### 2.2.1 Επιβλεπόμενη Μάθηση (Supervised Learning)

Η επιβλεπόμενη μάθηση αποτελεί μια από τις υποκατηγορίες της μηχανικής μάθησης. Σε αυτήν την κατηγορία εντάσσονται τα προβλήματα στα οποία τα δεδομένα αποτελούνται από ζεύγη  $(x,y)$  που  $x$  είναι η είσοδος του μοντέλου και  $y$  είναι η ετικέτα (label) που περιγράφει την επιθυμητή έξοδο όταν τροφοδοτείται στο σύστημα το  $x$ . Κατά την εκπαίδευση του δικτύου, εισάγεται η πληροφορία  $x$  (εικόνα, συμβολοσειρά, πρόταση κλπ.) και κατόπιν συγκρίνομε την προβλεπόμενη από το σύστημα έξοδο με την ετικέτα  $y$  της εισόδου. Η εκπαίδευση συνεχίζεται έως ότου αποκτηθεί η ακρίβεια που επιθυμούμε.



Σχήμα 2.3: Παράδειγμα επιβλεπόμενης μάθησης. [8]

Υπάρχουν δύο βασικές υποκατηγορίες της επιβλεπόμενης μάθησης:

- Παλινδρόμηση (Regression): Σε αυτό το πρόβλημα προσπαθούμε να μοντελοποιήσουμε τη σχέση μεταξύ των εξαρτημένων και των ανεξάρτητων μεταβλητών. Η έξοδος είναι μια συνεχής μεταβλητή για την οποία το μοντέλο μαθαίνει μια συνάρτηση που την αντιστοιχεί στην είσοδο. Συνεπώς η παραπάνω μέθοδος είναι αποτελεσματική και χρησιμοποιείται στην πρόβλεψη συνεχών μεταβλητών.

Τα πιο γνωστά μοντέλα παλινδρόμησης είναι:

- Γραμμική Παλινδρόμηση (Linear Regression):  
Το μοντέλο προσδιορίζει μια γραμμή που χωρίζει τα κατάλληλα τα δεδομένα. Προέκταση της προηγούμενης μεθόδου είναι και το μοντέλο Πολλαπλής Γραμμικής Παλινδρόμησης (Multiple Linear Regression), το οποίο κατασκευάζει ένα σύνολο γραμμών, δηλαδή ένα επίπεδο, το οποίο χωρίζει τα δεδομένα.
- Δέντρα Αποφάσεων (Decision Trees):  
Ένα δέντρο αποφάσεων είναι ένα εργαλείο υποστήριξης αποφάσεων που χρησιμοποιεί ένα μοντέλο αποφάσεων που μοιάζει με δέντρο και τις πιθανές συνέπειές τους, συμπεριλαμβανομένων των αποτελεσμάτων τυχαίων συμβάντων, του κόστους πόρων και της χρησιμότητας.
- Τυχαία Δάση (Random Forests):  
Τα τυχαία δάση είναι μια μέθοδος μάθησης παραπλήσια με την παραπάνω. Περιλαμβάνει την κατασκευή πολλών δέντρων αποφάσεων κατά το χρόνο εκπαίδευσης και ως τελική απόφαση επιλέγεται το αποτέλεσμα που δίνει η πλειοψηφία των δένδρων.
- Νευρωνικά Δίκτυα (Neural Networks)  
Μοντέλα που προσομοιώνουν την λειτουργία του ανθώπινου εγκεφάλου. Περιλαμβάνουν νευρώνες (neurons) οργανωμένους σε επίπεδα (layers), με ενώσεις (edges) μεταξύ τους σε μορφή συνάψεων.

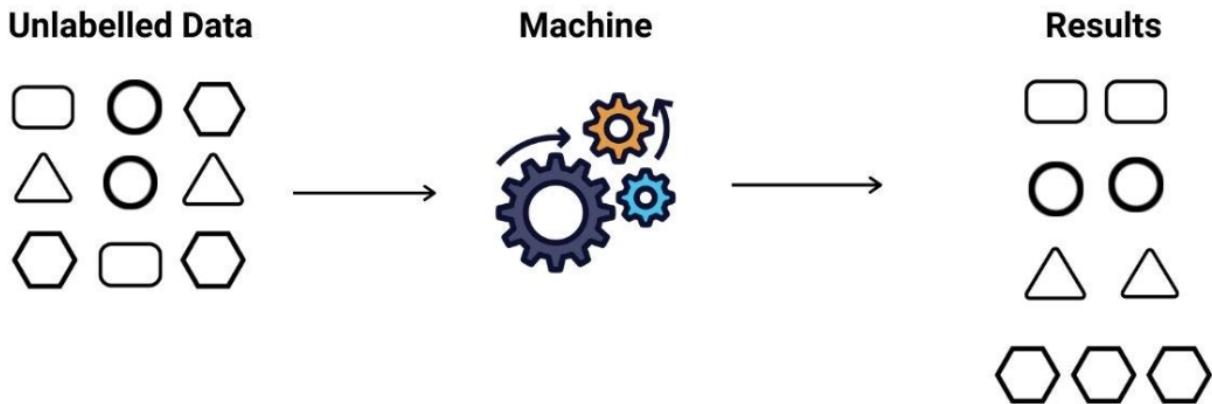
## 2. Ταξινόμηση (Classification)

Στο πρόβλημα της ταξινόμησης τα δεδομένα διαχωρίζονται σε  $N$  τάξεις. Σκοπός είναι ο προσδιορισμός μιας συνάρτησης  $f$  ικανής να αναθέτει τα δεδομένα  $x$  στην σωστή κατηγορία  $y$ . Οι ετικέτες (labels) στην προκειμένη περίπτωση είναι διακριτές. Όταν οι αριθμός των τάξεων είναι  $N = 2$  πρόκειται για πρόβλημα δυαδικής ταξινόμησης (binary classification) ενώ για  $N > 2$  ονομάζονται προβλήματα ταξινόμησης πολλαπλών τάξεων (multi-class classification). Τα πιο γνωστά μοντέλα ταξινόμησης είναι τα παρακάτω:

- Λογιστική Παλινδρόμηση (Logistic Regression)  
Παραπλήσιο της γραμμικής παλινδρόμησης αλλά χρησιμοποιείται στην μοντελοποίηση πιθανοτήτων ενός πεπερασμένου πλήθους αποτελεσμάτων.
- Μοντέλο Naïve Bayes:  
Πιθανοτικό μοντέλο που χρησιμοποιείται στην κατηγοριοποίηση δεδομένων.
- Μηχανές Διανυσμάτων Υποστήριξης (Support Vector Machines):  
Οι μηχανές διανυσμάτων υποστήριξης (SVMs) είναι μοντέλα μάθησης με επίβλεψη σχετιζόμενα με αλγορίθμους μάθησης που αναλύουν δεδομένα για ταξινόμηση και παλινδρόμηση.
- Δέντρα Αποφάσεων, Τυχαία Δάση και Νευρωνικά Δίκτυα:  
Παρόμοια μοντέλα με αυτά που αναφέρθηκαν στην κατηγορία της παλινδρόμησης.

## 2.2.2 Μη Επιβλεπόμενη Μάθηση (Unsupervised Learning)

Στην μη επιβλεπόμενη μάθηση, τα δεδομένα που παρέχονται στο σύστημα δεν αποτελούνται πλέον από ετικέτες (labels). Συνεπώς το μοντέλο καλείται να κατανοήσει μόνο του τα δεδομένα και να εντοπίσει μέσω κατάλληλης συνάρτησης τα μοτίβα και τις κατηγορίες που περιέχονται στα δεδομένα. Καθώς η ανθρώπινη συμβολή και εμπειρία απέχει από το μοντέλο (παροχή της επιθυμητής εξόδου στο σύστημα), ο τύπος μάθησης καλείται μη επιβλεπόμενος.



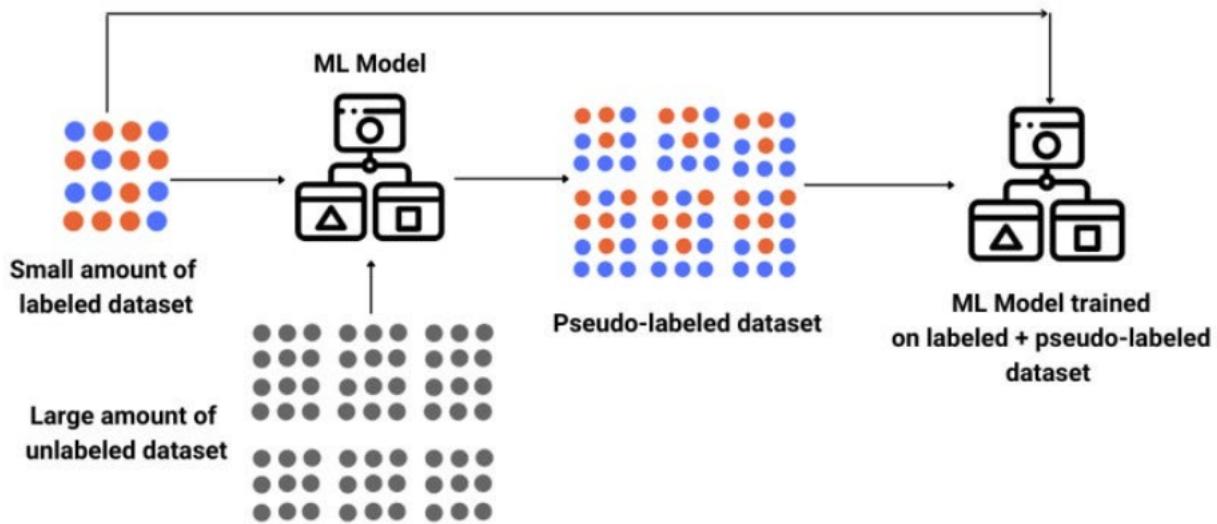
Σχήμα 2.4: Παράδειγμα μη επιβλεπόμενης μάθησης. [8]

Οι κύριες μέθοδοι που χρησιμοποιούνται σε αυτή την κατηγορία είναι:

1. Ομαδοποίηση (Clustering): Το σύστημα χρησιμοποιώντας μεθόδους όπως K-means, Hierarchical, Mean Shift και Destiny Based Clustering εντοπίζει δομές στα δεδομένα με βάση τις οποίες τα ταξινομεί σε διαφορετικές κατηγορίες.
2. Μείωση Διαστασημότητας (Dimensionality Reduction): Ελαττώνονται τα βήματα της διαδικασίας, ελαττώνοντας την διάσταση του συνόλου χαρακτηριστικών. Η μείωση διαστασημότητας διακρίνεται σε:
  - Αποκλεισμό Χαρακτηριστικών
  - Εξαγωγή Χαρακτηριστικών: Η μέθοδος Ανάλυσης Κύριων Συνιστωσών (Principal Component Analysis - PCA) είναι χαρακτηριστικό παράδειγμα στην εξαγωγή χαρακτηριστικών. Με την χρήση της επιτυγχάνεται η μείωση της διάστασης δεδομένων, διατηρώντας όμως την αρχική δομή και τις σχέσεις που χαρακτηρίζουν τις αρχικές δομές του συνόλου δεδομένων.

## 2.2.3 Ήμι-Επιβλεπόμενη Μάθηση (Semi-Supervised Learning)

Η ήμι-επιβλεπόμενη μάθηση αποτελεί μια ειδική περίπτωση την αδύναμης επίβλεψης (weak supervision) και εντάσσεται κάπου ανάμεσα στην προηγούμενη δύο μεθόδους που αναφέρθηκαν. Πιο συγκεκριμένα, στην φάση της εκπαίδευσης παρέχεται στο δίκτυο ένα μικρό κομμάτι δεδομένων με ετικέτες (labels) και ένα αρκετά μεγαλύτερο κομμάτι με δεδομένα δίχως κάποια ετικέτα (unlabeled data)[46]. Συνεπώς, το σύστημα καλείται με μια μικρή ανθρώπινη συμβολή να πραγματοποιήσει ότι και στην μη επιβλεπόμενη μάθηση. Ωστόσο, η μικρή ποσότητα επισημειώμένων δεδομένων που παρέχονται στο δίκτυο σε συνδυασμό με τα δεδομένα χωρίς, είναι αρκετή για την παρατήρηση αξιοσημείωτων αυξήσεων στην ακρίβεια.



Σχήμα 2.5: Παράδειγμα ήμι-επιβλεπόμενης μάθησης. [8]

## 2.2.4 Ενισχυτική Μάθηση (Reinforcement Learning)

Αρχικά, βασικοί όροι στην ενισχυτική μάθηση αποτελούν ο πράκτορας (agent), το περιβάλλον (environment), η κατάσταση (state), η δράση (action) και η ανταμοιβή (reward) [47]. Στην ενισχυτική μάθηση, κατασκευάζεται μια μέθοδος η οποία ανταμείβει την επιθυμητή συμπεριφορά του συστήματος και αντίστοιχα τιμωρεί την αρνητική. Αναλυτικότερα, αυτή η μέθοδος αναθέτει θετικές τιμές στις σωστές πράξεις που πραγματοποιεί το σύστημα και ομοίως, αναθέτει αρνητικές στις μη επιθυμητές. Συγκεκριμένα, υπάρχει ένας agent ικανός να αντιλαμβάνεται το περιβάλλον του, να λαμβάνει αποφάσεις και να πραγματοποιεί ενέργειες. Όταν μια απόφαση οδηγεί σε επιθυμητή κατάσταση λαμβάνει ανταμοιβή. Κατά αυτόν τον τρόπο, οδηγείται στην εύρεση της μεγαλύτερης δυνατής ανταμοιβής που θα οδηγήσει και στην βέλτιστη λύση του προβλήματος.

## 2.3 Βαθιά Μάθηση

### 2.3.1 Εισαγωγή

Η βαθιά μάθηση αποτελεί μια υποκατηγορία της Μηχανικής Μάθησης που έχει φέρει επανάσταση στον τομέα της ανάλυσης δεδομένων. Ωστόσο, είναι λογικό να διερωτηθεί κανείς γιατί ενώ οι βασικές τεχνικές ιδέες για την βαθιά μάθηση και τα νευρωνικά δίκτυα υπάρχουν εδώ και αρκετά χρόνια, μόλις πρόσφατα έχει αυτή η εκθετική αύξηση στην χρήση τους. Για κύριο αίτιο αυτού μπορούμε να προσδιορίσουμε την ψηφιοποίηση του κόσμου τριγύρω μας. Τα τελευταία χρόνια έχει υπάρξει μια έκρηξη ψηφιακών δεδομένων καθώς τα πάντα αρχίζουν να αποκτούν μια ψηφιακή παρουσία. Συνεπώς η κινητήρια δύναμη της βαθιάς μάθησης, τα δεδομένα, έχει πολλαπλασιαστεί και σαν απόρροια, τα αποτελέσματα και η ακρίβεια που παίρνουμε σε αυτά έχει αυξηθεί σημαντικά. Αυτό ξεκινάει έναν φαύλο κύκλο από σύλληψη νέων ιδεών, τον προγραμματισμό τους και τον πειραματισμό, απαραίτητο για την ανάπτυξη οποιουδήποτε τομέα.

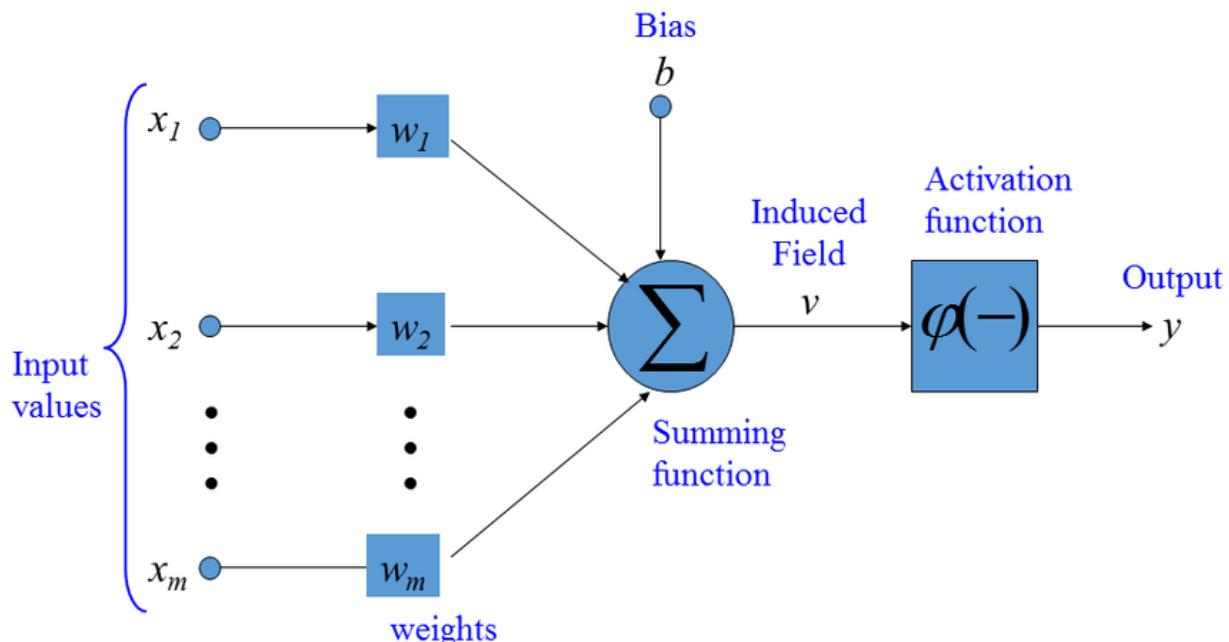
### 2.3.2 Τεχνητά Νευρωνικά Δίκτυα (Artificial Neural Networks - ANN)

Τα τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks - ANN) αποτελούν την βάση της βαθιάς μάθησης. Πρόκειται για υπολογιστικά συστήματα που οφείλουν την ονομασία τους στον βιολογικό εγκέφαλο καθώς προσπαθούν να πετύχουν την προσομοίωση του, να μιμηθούν δηλαδή τον τρόπο που επικοινωνούν οι νευρώνες μεταξύ τους για την εξαγωγή κάποιου συμπεράσματος.

Τα απλά ANN είναι γνωστά και ως Feed-Forward Neural Networks καθώς η πληροφορία διαδίδεται μόνο προς τα μπροστά. Φυσικά υπάρχουν και αρχιτεκτονικές δικτύων που η πληροφορία μπορεί να διαδοθεί και προς τα πίσω, με χαρακτηριστικό παράδειγμα τα επαναλαμβανόμενα νευρωνικά δίκτυα (Recurrent Neural Networks) που θα αναλυθούν και στα παρακάτω κεφάλαια.

### 2.3.3 Αρχιτεκτονικές των Τεχνητών Νευρωνικών Δικτύων

Όπως προαναφέρθηκε, τα τεχνητά νευρωνικά δίκτυα προσπαθούν να προσομοιώσουν τις λειτουργίες του ανθρώπινου εγκεφάλου. Δομικό τους στοιχείο αποτελούν οι νευρώνες (neurons). Ο νευρώνας δέχεται μία οι περισσότερες εισόδους (inputs) και έπειτα από κατάλληλη επεξεργασία των εισερχομένων δεδομένων παράγει μία έξοδο (output). Η μαθηματική μοντελοποίηση του νευρώνα αναφαίνεται στο σχήμα παρακάτω.



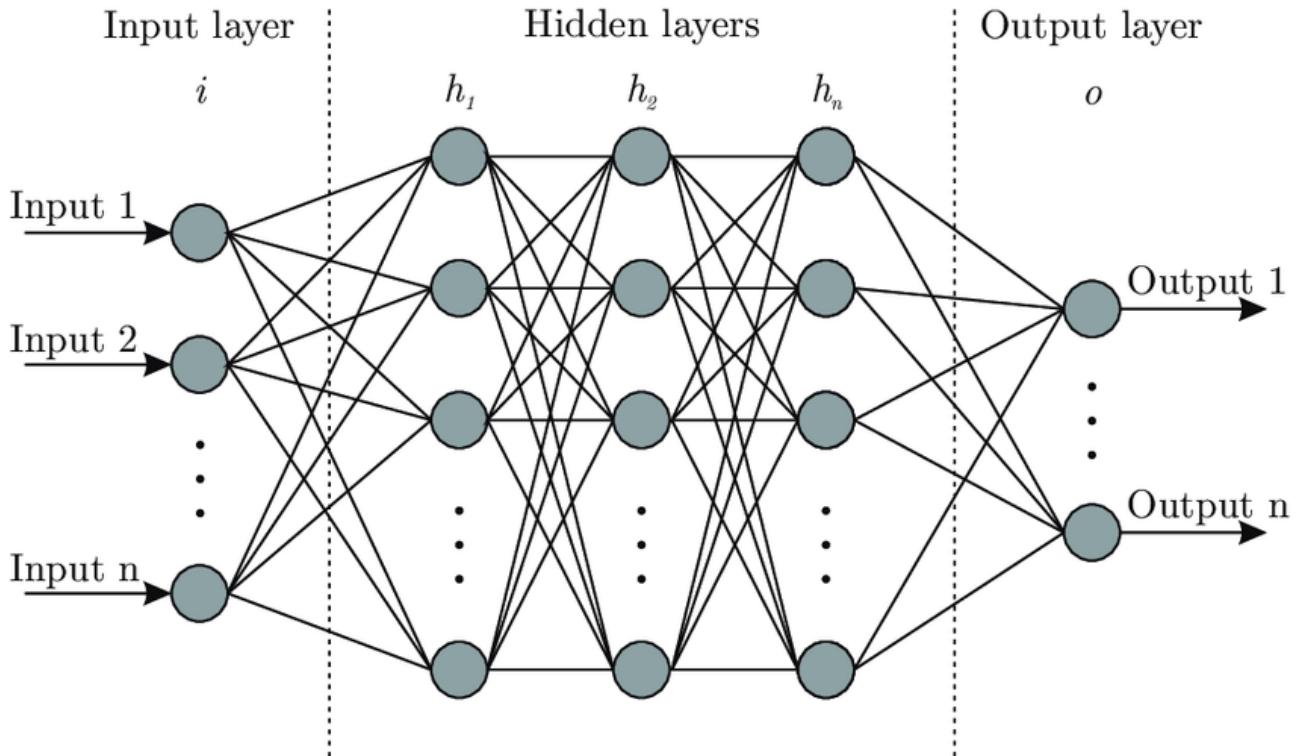
Σχήμα 2.6: Παράδειγμα νευρώνα στα νευρωνικά δίκτυα. [9]

Οι είσοδοι του κάθε νευρώνα εισέρχονται σε αυτόν μέσω ακμών. Κάθε ακμή φέρνει και ένα βάρος (weight) που δείχνει το πόσο σημαντική είναι η εκάστοτε είσοδος στην εξαγωγή του συμπεράσματος. Με βάση λοιπόν τα βάρον, υπολογίζεται το σταθμισμένο άρθροισμα των εισόδων, στο οποίο προστίθεται μια σταθερά (bias) και έτσι κατασκευάζεται η εσωτερική κατάσταση του νευρώνα. Στην συνέχεια, εφαρμόζεται η συνάρτηση ενεργοποίησης (activation function) και κατά αυτόν τον τρόπο προκύπτει η έξοδος του νευρώνα. Η τιμή της έξόδου που προκύπτει καθορίζει αν ο νευρώνας θα ενεργοποιηθεί. Η διαδικασία της διάδοσης της πληροφορίας μέσα από το δίκτυο ονομάζεται forward propagation. Μερικές συναρτήσεις ενεργοποίησης παρουσιάζονται παρακάτω αλλά θα αναλυθούν στην συνέχεια περαιτέρω.

Σε ένα δίκτυο βαθιάς μάθησης, οι νευρώνες δομούνται σε επίπεδα (layers) τα οποία συνδέονται με τις συνάψεις (σταθμισμένες ακμές) που φέρουν οι νευρώνες. Κάθε επίπεδο μπορεί να επικοινωνίσει μόνο με το προηγούμενο και το επόμενο layer. Συνεπώς ο κάθε νευρώνας γνωρίζει μόνο την πληροφορία που εισέρχεται σε αυτό, αυτή που μεταβιβάζει στους νευρώνες του επόμενου επιπέδου και όχι την κατάσταση ολόκληρου του δικτύου. Όταν κάθε νευρώνας ενός επιπέδου συνδέεται με κάθε νευρώνα του επόμενου (όπως το δίκτυο της εικόνας), το δίκτυο ονομάζεται πλήρως συνδεδεμένο δίκτυο (fully connected network).

Στο πρώτο layer του νευρωνικού δικτύου εισέρχονται τα δεδομένα που διαθέτουμε για εκπαίδευση και για αυτό τον λόγο ονομάζεται επίπεδο εισόδου (input layer). Αντίστοιχα, για προφανής

λόγους το τελευταίο επίπεδο είναι το επίπεδο εξόδου (output layer). Τα ενδιάμεσα επίπεδα, καθώς δεν μας παρέχουν άμεσα κάποια χρήσιμη πληροφορία, καλούνται κρυφά επίπεδα (hidden layers).



Σχήμα 2.7: Παράδειγμα πλήρους συνδεδεμένου νευρωνικού δικτύου. [10]

### 2.3.4 Εφαρμογές

Σύμφωνα με την εργασία του Cybenko το 1989 [48], τα νευρωνικά δίκτυα είναι ικανά να προσεγγίσουν οποιαδήποτε τυπωματικά συνεχή συνάρτηση οσοδήποτε κοντά επιθυμούμε. Συνεπώς δεν θα έπρεπε να είναι δύσκολο να φανταστούμε τις αμέτρητες εφαρμογές που θα μπορούσαν να συμβάλλουν σημαντικά τα νευρωνικά δίκτυα καθώς κατά βάση οι περισσότερες ανθρώπινες εργασίες μπορούν να αναπαρασταθούν από ένα σύνολο δεδομένων τα οποία οι άνθρωποι επεξεργάζονται και εξάγουν τα αποτελέσματα που επιθυμούν. Καθώς η υπολογιστική ισχύ των μηχανών ξεπερνούν εκείνη του ανθρώπινου εγκεφάλου, είναι ακολούθως λογική και η ταχύτερη και ακριβέστερη εύρεση λύσεων στα προβλήματα του. Οι κύριες εφαρμογές της βαθιάς μάθησης σήμερα είναι:

- Αυτόνομη Οδήγηση (Self-Driving Cars)
- Επεξεργασία Φυσικής Γλώσσας (Natural Language Processing - NLP)
- Εικονική Βοηθοί (Virtual Assistants)
- Εικονική Αναγνώριση (Virtual Recognition)
- Healthcare
- Ανάκτηση εικονοστοιχείων εικόνας (Pixel Restoration)
- Συμπίεση Εικόνας (Image Compression)

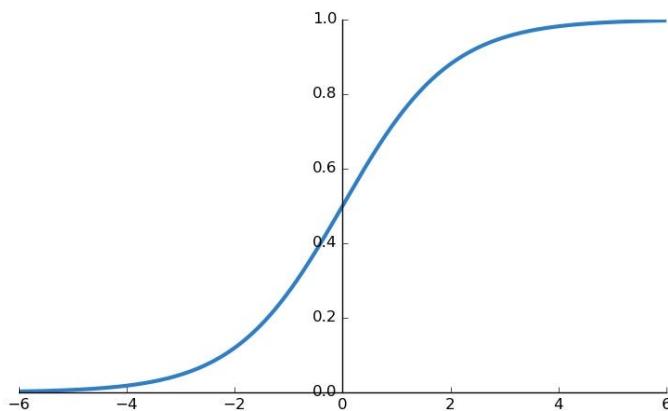
και πολλά άλλα.

### 2.3.5 Συνάρτηση Ενεργοποίησης

Όπως προαναφέρθηκε, στην εσωτερική κατάσταση κάθε νευρώνα εφαρμόζεται μια συνάρτηση ενεργοποίησης (activation function). Αυτή είναι μια πολύ σημαντική ενέργεια στα νευρωνικά δίκτυα καθώς όχι μόνο αποφασίζει αν ο νευρώνας θα ενεργοποιηθεί και θα μεταφέρει την πληροφορία στους παρακάτω νευρώνες αλλά βασικά εισάγει και το στοιχείο της μη γραμμικότητας (non-linearity). Δίχως την συνάρτηση ενεργοποίησης, κάθε νευρωνικό δίκτυο θα ήταν απλά ένα σύστημα γραμμικής παλινδρόμησης, περιορισμένο στην επίλυση προβλημάτων με γραμμικά διαχωρίσιμα δεδομένα. Συμπεριλαμβάνοντας την συνάρτηση ενεργοποίησης μας δίνεται συνεπώς η δυνατότητα αντιμετώπισης αρκετά πιο περίπλοκων εργασιών [49]. Οι πιο δημοφιλής συναρτήσεις ενεργοποίησης είναι:

1. Σιγμοειδής (Sigmoid): Ο κύριος λόγος χρήσης της σιγμοειδούς συνάρτησης είναι ότι το αποτέλεσμα περιορίζεται σε τιμές από 0 έως 1. Έτσι, είναι ιδιαίτερα χρήσιμη σε προβλήματα που η έξοδος είναι μια πιθανότητα και συνεπώς περιορίζεται σε αυτό το διάστημα. Πρόκειται για παραγωγήσιμη συνάρτηση, δηλαδή είναι δυνατή η εύρεση της κλίσης της ανάμεσα σε δύο οποιαδήποτε σημεία. Η συνάρτηση είναι μονότονη αλλά όχι και η παράγωγός της. Ο τύπος της είναι ο παρακάτω:

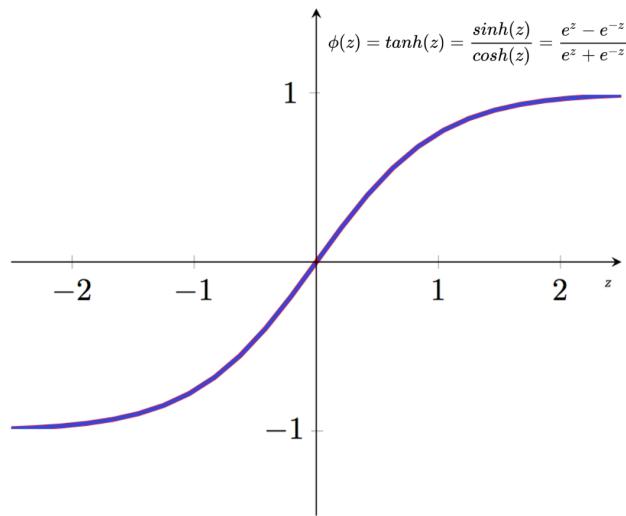
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$



Σχήμα 2.8: Η σιγμοειδής συνάρτηση ενεργοποίησης. [11]

2. Υπερβολική Εφαπτομένη (Hyperbolic Tangent): Η συνάρτηση της υπερβολικής εφαπτομένης είναι μια καλύτερη εκδοχή της σιγμοειδούς. Οι έξοδοι των νευρώνων ανίκουν στο σύνολο (-1,1). Έτσι, οι αρνητικές τιμές αντιστοιχίζονται ξανά σε αρνητικές και οι μηδενικές αντιστοιχά σε 0. Ο τύπος της είναι ο παρακάτω:

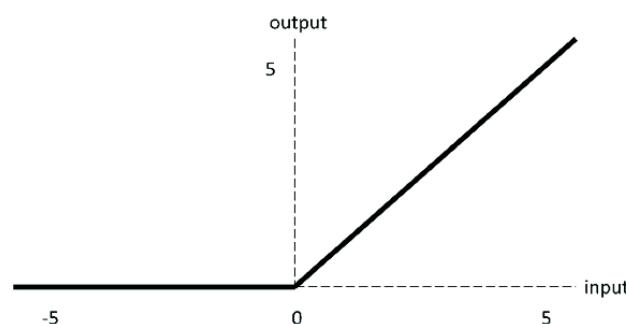
$$a(x) = \tanh(x)$$



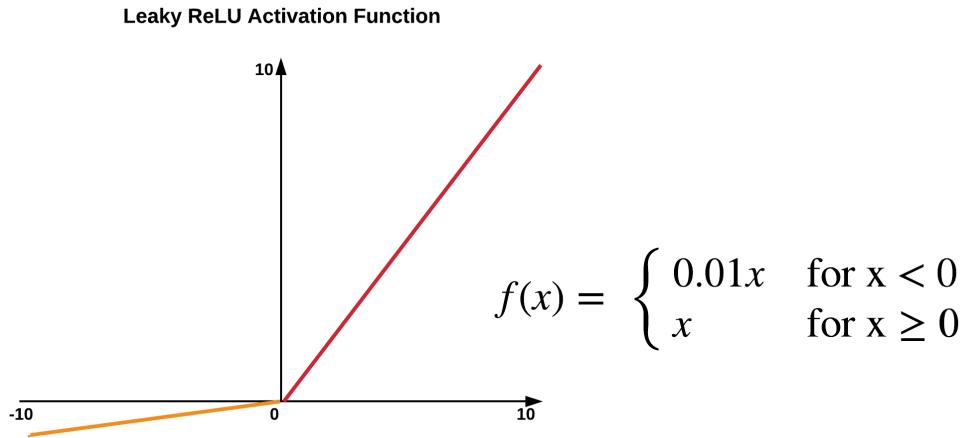
Σχήμα 2.9: Η υπερβολική εφαπτομένη συνάρτηση ενεργοποίησης. [12]

3. Ανυψωμένη γραμμική μονάδα (Rectified Linear Unit - ReLu): Η συνάρτηση ανυψωμένης γραμμικής μονάδας είναι η πιο ευρέως χρησιμοποιούμενη συνάρτηση ενεργοποίησης στα νευρωνικά δίκτυα. Όπως φαίνεται και στο σχήμα παρακάτω, όλες οι αρνητικές τιμές αντιστοιχίζονται στο 0. Υστερα, η συνάρτηση είναι γραμμική και συνεπώς το εύρος της είναι (0, άπειρο). Η συνάρτηση είναι παραγωγίσιμη και όπως και η παραγάγωγός της είναι μονότονη. Το πρόβλημα της ReLu είναι ότι όλες οι αρνητικές τιμές αντιστοιχίζονται στο 0, κάτι που μπορεί να εμποδίσει το δίκτυο να μάθει σωστά. Για αυτό τον λόγο κατασκευάστηκε η leaky ReLu (βλέπε σχήμα) που σύμφωνα με την παράμετρο α υπάρχει μία ανοχή στις αρνητικές τιμές των εξόδων των εσωτερικών καταστάσεων των νευρώνων. Ο τύπος είναι ο παρακάτω:

$$a(x) = \max(0, x)$$



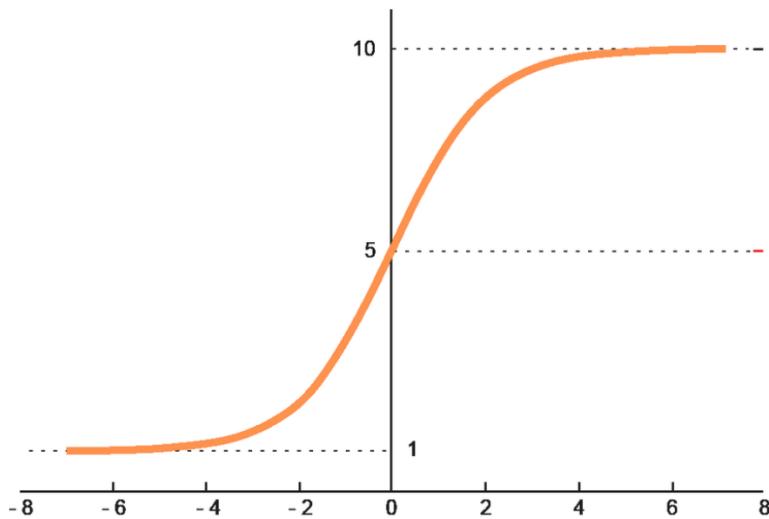
Σχήμα 2.10: Η ανυψωμένη γραμμική μονάδα συνάρτηση ενεργοποίησης. [13]



Σχήμα 2.11: Η Leaky ReLU συνάρτηση ενεργοποίησης. [14]

4. Softmax: Η Softmax αποτελεί μια πιο γενική μορφή της λογιστικής συνάρτησης ενεργοποίησης. Χρησιμοποιείται κυρίως για προβλήματα που απαιτείται κατηγοριοποίηση πολλαπλών κλάσεων. Είναι παραγωγίσιμη και μονότονη ενώ η παράγωγος είναι μη μονότονη. Ο τύπος είναι ο παρακάτω:

$$S(x_i) = \frac{e^{x_i}}{\sum_{j=1}^n e^{x_j}}$$



Σχήμα 2.12: Η softmax συνάρτηση ενεργοποίησης. [15]

Οι συναρτήσεις ενεργοποίησης αποτελούν πολύ σημαντικό σημείο στην σωστή λειτουργία ενός νευρωνικού δικτύου. Έπειτα από κατανόηση της φύσης του προβλήματος που προσπαθούμε να επιλύσουμε, επιλέγουμε και τις κατάλληλες συναρτήσεις ενεργοποίησης. Οφείλουμε να σημειώσουμε ότι δεν είναι η απαραίτητη η χρήση ενός τύπου activation function σε όλο το δίκτυο, αντίθετα συνηθίζεται η χρήση πολλαπλών, ανάλογα με την εργασία που υλοποιούμε.

### 2.3.6 Εκπαίδευση

#### Συνάρτηση Σφάλματος

Όπως είδαμε στα προηγούμενα κεφάλαια, ο τρόπος λειτουργίας των νευρωνικών δικτύων ακολουθεί την παρακάτω λογική. Τα δεδομένα εισάγονται στο πρώτο επίπεδο (input layer) του δικτύου. Αφού εκτελεστούν οι απαιτούμενες πράξεις από τους νευρώνες, η πληροφορία μεταδίδεται βαθύτερα στο δίκτυο έως και το επίπεδο εξόδου (output layer) που λαμβάνουμε το αποτέλεσμα που παρήγαγε το μοντέλο μας. Ωστόσο απαραίτητο για την σωστή εκπαίδευσή του δικτύου είναι η σύγκριση του output (y\_pred) με το ιδανικό (y\_true). Σε αυτό το κομμάτι εισάγεται η έννοια της συνάρτησης σφάλματος που καλείται να δώσει μια μετρητική για την αναπαράσταση της απόστασης της εξόδου και του y\_true.

Δημοφιλέστερες συναρτήσεις σφάλματος είναι το μέσο τετραγωνικό σφάλμα (Mean Squared Error - MSE), η συνάρτηση δυαδικής εντροπίας (binary cross-entropy) για προβλήματα δυαδικής κατηγοριοποίησης και τέλος η κατηγορική εντροπία (categorical cross-entropy) σε περιπτώσεις που έχουμε παραπάνω από δύο κατηγορίες. Γενικά, το συνολικό σφάλμα είναι το σταθμισμένο άθροισμα των σφαλμάτων του κάθε στοιχείου του δικτύου και εκφράζεται με τον παρακάτω τύπο:

$$L = \frac{1}{N} \sum_{i=1}^N L_i$$

### Βελτιστοποίηση

#### 2.3.7 Οπισθοδιάδοση

Η οπισθοδιάδοση (backpropagation) στον τομέα της μηχανικής μάθησης χρησιμοποιείται για την εκπαίδευση των νευρωνικών δικτύων. Συγκεκριμένα, ο αλγόριθμος μεταδίδει το σφάλμα της εξόδου του δικτύου προς τα πίσω, περνώντας από όλα τα προηγούμενα επίπεδα του νευρωνικού. Κατά αυτό τον τρόπο, καθορίζει ποια μονοπάτια έφεραν μεγαλύτερο συντελεστή στην δημιουργία του σφάλματος. Ο αλγόριθμος της οπισθοδιάδοσης χρησιμοποιεί τον κανόνα της αλυσίδας για τον υπολογισμό της κλίσης της συνάρτησης σφάλματος σε σχέση με τα βάρον. Στην συνέχεια, τα βάρον ανανεώνονται με την χρήση της κλίσης καθόδου (gradient descent) ή διαφορετικών μεθόδων βελτιστοποίησης και έτσι επιτυγχάνεται ελαχιστοποίηση του σφάλματος.

#### 2.3.8 Υπερ-παραμέτροι Νευρωνικών Δικτύων

Κάθε νευρωνικό δίκτυο χαρακτηρίζεται από ένα σύνολο υπερ-παραμέτρων οι οποίες δεν μαθαίνονται από το δίκτυο αλλά πρέπει να οριστούν εκ των προτέρων. Σκοπός αυτών είναι η βελτιστοποίηση του δικτύου και η μεγιστοποίηση της ακρίβειας του μοντέλου. Οι σημαντικότερες από αυτές είναι οι παρακάτω:

##### 1. Μέγεθος νευρωνικού δικτύου

Ένα από τα σημαντικότερα μεγέθη μοντέλοποίησης των νευρωνικών δικτύων είναι η επιλογή του αριθμού των επιπέδων και των νευρώνων που θα τα απαρτίζουν. Το μέγεθος του δικτύου επηρεάζει άμεσα την ακρίβεια, την αποτελεσματικότητα και την πολυπλοκότητα του μοντέλου. Αξιοσημείωτο είναι ότι με ένα επαρκώς μεγάλο δίκτυο, μπορεί να προσεγγίσει οποιαδήποτε συνάρτηση. Μερικοί από τους παραγοντες που μπορεί να επηρεάσουν την επιλογή του μεγέθους του νευρωνικού είναι:

- Η πολυπλοκότητα του προβλήματος
- Η ποσότητα και ποιότητα των διαθέσιμων δεδομένων
- Η διαθέσιμη υπολογιστική ισχύς

##### 2. Είδος συνάρτησης ενεργοποίησης

Δίνεται επιλογή της συνάρτησης ενεργοποίησης του κάθε επιπέδου, όπως και η επιλογή των υπερπαραμέτρων της κάθε συνάρτησης.

### 3. Ρυθμός εκμάθησης

Με την επιλογή του ρυθμού εκμάθησης (learning rate) του μοντέλου, επηρεάζεται το πόσο θα μεταβάλλονται τα βάροντα του νευρωνικού δικτύου. Με την σωστή προσαρμογή του μπορούμε να προσεγγίσουμε ταχύτερα και ακριβέστερα την επιθυμητή έξοδο του μοντέλου.

### 4. Βελτιστοποιητής (Optimizer)

Οι διαφορετικοί βελτιστοποιητές όπου μπορούμε να επιλέξουμε είναι:

- κλίσης καθόδου (Gradient Descent)

Πρόκειται για τον πιο απλό και ευρέως χρησιμοποιούμενο βελτιστοποιητή. Ενημερώνει τις παραμέτρους κατά την αντίθετη κατεύθυνση της κλίσης της συνάρτησης σφάλματος. Έχει σταθερό ρυθμό εκμάθησης ο οποίος και καθορίζει το μέγεθος της αλλαγής του επόμενου βήματος. Πρόκειται για αλγόριθμο ο οποίος μπορεί εύκολα να κατανοθεί και να υλοποιηθεί. Το αρνητικό σημείο του αλγορίθμου αυτού είναι ότι μπορεί να υπάρξει δυσκολία στην σύγκλιση στην βέλτιστη λύση και να κολλήσει σε τοπικά ελάχιστα. Ο τύπος δίνεται παρακάτω:

$$\theta_{t+1} = \theta_t - \eta \nabla J(\theta_t)$$

- Στοχαστικής κλίσης καθόδου (Stochastic gradient descent)

Ο βελτιστοποιητής κλίσης καθόδου, όπως φαίνεται και από τον παρακάτω τύπο, αποτελεί μια παραλλαγή της κλίσης καθόδου. Ο συγκεκριμένος αλγόριθμος ενημερώνει τις παραμέτρους του μοντέλου κάθε φορά που υπολογίζεται το κόστος σε κάθε δεδομένο εκπαίδευσης. Αυτό έχει ως αποτέλεσμα την ταχύτερη σύγκλιση. Ωστόσο, αξίζει να σημειωθεί ότι υπάρχει πιθανότητα αποσταθεροποίησης του μοντέλου, ακόμα και αν αυτό έχει συγκλίνει στην βέλτιστη λύση. Συνεπώς, όταν γίνεται χρήση του συγκεκριμένου βελτιστοποιητή, συνδυάζεται με την σταδιακή μείωση του συντελεστή εκμάθησης. Ο τύπος του είναι ο παρακάτω:

$$\theta_{t+1} = \theta_t - \eta \nabla J(\theta_t; x_i, y_i)$$

- Ορμής (Momentum)

Η ορμή είναι μια επέκταση του αλγορίθμου βελτιστοποίησης με κλίση καθόδου που επιτρέπει στην αναζήτηση να αναπτύξει αδράνεια προς μια κατεύθυνση στο χώρο αναζήτησης και να ξεπεράσει τις ταλαντώσεις των θορυβωδών κλίσεων και να διασχίσει τα επίπεδα σημεία του χώρου αναζήτησης. Βασίζεται στην ίδια έννοια της ορμής που ορίζεται και στην φυσική, που ένα κινούμενο αντικείμενο τείνει να συνεχίσει την κίνηση του εκτός και αν επιδράσει πάνω του μια εξωτερική δύναμη. Η ορμή έχει δύο υπερπαραγμέτρους: Τον ρυθμό μάθησης και τη σταθερά ορμής. Ο ρυθμός μάθησης ελέγχει πόσο ενημερώνονται τα βάροντα σε κάθε βήμα, ενώ η σταθερά ορμής ελέγχει πόσο διατηρείται η προηγούμενη ενημέρωση. Το διάνυσμα ορμής στο βήμα  $i$  δίνεται από τον τύπο:

$$b_i = \mu b_{i-1} + g_i$$

που  $\mu$  είναι η σταθερά ορμής,  $b_{i-1}$  είναι το προηγούμενο διάνυσμα ορμής και  $g_i$  είναι η τρέχουσα κλίση. Ενώ η ενημέρωση του βάρους στο βήμα  $i$  δίνεται από:

$$\theta_i = \theta_{i-1} - \gamma b_i$$

που  $\theta_i$  είναι το τρέχον βάρος,  $\theta_{i-1}$  είναι το προηγούμενο βάρος,  $\gamma$  είναι ο ρυθμός μάθησης και  $b_i$  είναι το τρέχον διάνυσμα ορμής.

- Adam

Ο βελτιστοποιητής Adam μπορεί να χρησιμοποιηθεί αντί της κλασικής διαδικασίας στοχαστικής κλίσης καθόδου. Ο Adam, επιτρέπει την επαναληπτική ενημέρωση των βαρών του δικτύου με βάση τα δεδομένα εκπαίδευσης ενώ χρησιμοποιεί τις ροπές πρώτης και δεύτερης τάξης των κλίσεων. Ο παραπάνω βελτιστοποιητής είναι υπολογιστικά αποδοτικός, συγκλίνει ταχύτατα και είναι κατάλληλος για προβλήματα τα οποία περιέχουν μεγάλο αριθμό δεδομένων ή παραμέτρων. Οι εξισώσεις του είναι οι παρακάτω:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \\ \hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1 - \beta_2^t} \\ \theta_{t+1} &= \theta_t - \alpha \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \end{aligned}$$

που

- $m_t$  είναι η πρώτη ροπή, ο μέσος όρος των κλίσεων
- $v_t$  είναι η δεύτερη ροπή, η μη επικεντρωμένη διακύμανση των κλίσεων
- $\hat{m}_t$  and  $\hat{v}_t$  είναι οι διορθωμένες με bias(?) εκτιμήσεις της πρώτης και δεύτερης ροπής
- $\alpha$  είναι το μέγεθος του βήματος
- $\beta_1$  and  $\beta_2$  είναι οι ρυθμοί μείωσης των εκτιμήσεων των ροπών
- $\epsilon$  είναι μικρή σταθερά για την αποφυγή διαίρεσης με το 0
- $g_t$  είναι η κλίση στο βήμα  $t$
- $\theta_t$  είναι το διάνυσμα παραμέτρων το βήμα  $t$

## 5. Συνάρτηση Απώλειας

Η συνάρτηση απώλειας πρόκειται για μια μέθοδο που χρησιμοποιείται για τον υπολογισμό της απόστασης των εξόδων του μοντέλου και των πραγματικών στόχων. Συγκεκριμένα, η συνάρτηση λαμβάνει μικρότερες τιμές όσο λιγότερο απέχει η έξοδος από την πραγματική τιμή, υπολογίζοντας το μέγεθος του σφάλματος που πραγματοποιεί το μοντέλο στα δεδομένα εκπαίδευσης και κατευθύνοντας τον βελτιστοποιητή προς την ελαχιστοποίηση του. Η επιλογή της κατάλληλης συνάρτησης απώλειας εξαρτάται κυρίως από τον τύπο του προβλήματος και την κατανομή της εξόδου. Μια από τις πιο ευρέως χρησιμοποιούμενες συναρτήσεις απώλειας, η οποία αποτελεί και την συνάρτηση που χρησιμοποιείται και στην συγκεκριμένη υλοποίηση είναι η συνάρτηση:

- **Μέσου Τετραγωνικού Σφάλματος**

Η συνάρτηση MSE, ή συνάρτηση μέσου τετραγωνικού σφάλματος μετρά τον μέσο όρο των τετραγωνικών διαφορών μεταξύ των προβλέψεων του μοντέλου και των τιμών-στόχων. Η εξίσωση είναι η παρακάτω:

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

που  $N$  είναι ο αριθμός των δειγμάτων,  $y_i$  είναι η τιμή-στόχος και  $\hat{y}_i$  είναι η έξοδος που προβλέπει το μοντέλο.

Η συνάρτηση MSE έχει ορισμένα πλεονεκτήματα και μειονεκτήματα. Αρχικά, τιμωρεί τα μεγάλα σφάλματα περισσότερο από τα μικρά, γεγονός που μπορεί να βοηθήσει στην αποφυγή ακραίων τιμών και στη βελτίωση της ακρίβειας του μοντέλου. Ένα ακόμη πλεονέκτημα είναι ότι είναι διαφοροποιήσιμη και συνεπώς καθιστά εύκολη τη χρήση της με αλγορίθμους βελτιστοποίησης βασισμένους στην κλίση. Ωστόσο, μπορεί να είναι ευαίσθητη στο θόρυβο και στις ακραίες τιμές, οι οποίες μπορεί να παραμορφώσουν το σήμα σφάλματος και να οδηγήσουν το μοντέλο στην υπερ-εκπαίδευση.

## 6. Μέγεθος Παρτίδας

Η έννοια του μεγέθους παρτίδας είναι κομβικής σημασίας για την εκπαίδευση των νευρωνικών δικτύων και καθορίζει τον αριθμό των δειγμάτων που διαβίδονται μέσω του νευρωνικού δικτύου. Πιο συγκεκριμένα, σε μια περίπτωση που τα συνολικά δεδομένα εισόδου είναι 100 δείγματα και έχουμε ορίσει ως μέγεθος παρτίδας 20, κάθε φορά η εκπαίδευση του δικτύου θα γίνεται στα δείγματα που έχει ορίσει το μέγεθος παρτίδας, δηλαδή θα εκπαιδευτεί αρχικά στα πρώτα 20, έπειτα στα επόμενα 20 κ.ο.κ., μέχρι να χρησιμοποιηθούν όλα τα δεδομένα.

Ένα σημαντικό πλεονέκτημα που φέρει η χρήση μεγέθους παρτίδας, είναι ότι το δίκτυο εκπαιδεύεται σε υποσύνολα του αρχικού συνόλου δεδομένων, που περιέχουν λιγότερα δείγματα, γεγονός που μειώνει τη χρήση μνήμης. Επιπλέον, η εκπαίδευση είναι ταχύτερη όταν το νευρωνικό δίκτυο εκπαιδεύεται με μίνι παρτίδες, καθώς το μοντέλο ενημερώνει τις παραμέτρους του συχνότερα για κάθε πέρασμα των δεδομένων. Στο παραπάνω παράδειγμα, το σύστημα εκπαιδεύτηκε σε πέντε διαφορετικές παρτίδες, με αποτέλεσμα τα βάρον να ενημερώνονται πέντε φορές. Εάν το σύνολο δεν χωριζόταν σε μίνι παρτίδες, τα βάρον θα ενημερώνονταν μόνο μία.

Ωστόσο, ένα σημαντικό μειονέκτημα των μικρότερων μεγεθών παρτίδων είναι ότι ο υπολογισμός της κλίσης γίνεται λιγότερο ακριβής, γεγονός που μπορεί να προκαλέσει αστάθεια κατά τη διάρκεια της εκπαίδευσης. Αυτή η ανακρίβεια συμβαίνει επειδή η κλίση υπολογίζεται με βάση μικρότερο αριθμό δειγμάτων, τα οποία ενδέχεται να μην αντιπροσωπεύουν με ακρίβεια τη συνολική κατανομή των δεδομένων. Ως αποτέλεσμα, η εκπαίδευση με μικρότερα μεγέθη παρτίδων απαιτεί προσεκτική εξέταση για να διασφαλιστεί ότι το μοντέλο δεν προσαρμόζεται υπερβολικά στα δεδομένα εκπαίδευσης.

## 7. Αριθμός Εποχών

Οι εποχές είναι μια ακόμη κρίσιμη πτυχή της εκπαίδευσης ενός νευρωνικού δικτύου. Αντιπροσωπεύουν το πέρασμα όλων των δεδομένων του συνόλου εκπαίδευσης μέσω του δικτύου. Στις περισσότερες περιπτώσεις όμως, μια εποχή δεν είναι αρκετή για την επίτευξη του ολικού ελάχιστου και, ως εκ τούτου, τα δεδομένα τροφοδοτούνται επανειλημμένα στο σύστημα μέχρι να επιτευχθεί ένα αποδεκτό ποσοστό ακρίβειας.

Ο καθορισμός του αριθμού των εποχών πρέπει να είναι κατάλληλος, καθώς οι πολλές εποχές μπορεί να οδηγήσουν σε υπερπροσαρμογή, ενώ οι πολύ λίγες μπορεί να οδηγήσουν σε υποπροσαρμογή. Για να μετριαστεί το φαινόμενο αυτό, τίθεται σε εφαρμογή ένας μηχανισμός παρακολούθησης της απώλειας του συστήματος στο σύνολο επικύρωσης. Με βάση τις αλλαγές, το σύστημα καθορίζει την κατάλληλη εποχή για τον τερματισμό της εκπαίδευσης.

## 8. Υπερπροσαρμογή και υποπροσαρμογή

Όταν θέλουμε να αναλύσουμε πόσο καλά αποδίδει ένα μοντέλο μηχανικής μάθησης, αναφερόμαστε συνήθως στην ακρίβεια και το ποσοστό λάθους που πραγματοποιεί το εκάστοτε μοντέλο. Χαρακτηρίζουμε λοιπόν ένα μοντέλο ως "καλό" όταν καταφέρνει να γενικεύει με κατάλληλο τρόπο καινούργια δεδομένα εισόδου και δημιουργεί σωστές προβλέψεις πάνω σε αυτά. Για να ελέγχουμε κατά πόσο λοιπόν αυτό το μοντέλο έχει καλή απόδοση,

εισάγονται οι όροι της υπερπροσαρμογής και υποπροσαρμογής. Προτού αναλύσουμε τους δύο αυτούς όρους, είναι σημαντικό να εξηγηθούν οι δύο παρακάτω έννοιες:

- Προτίμηση (Bias)

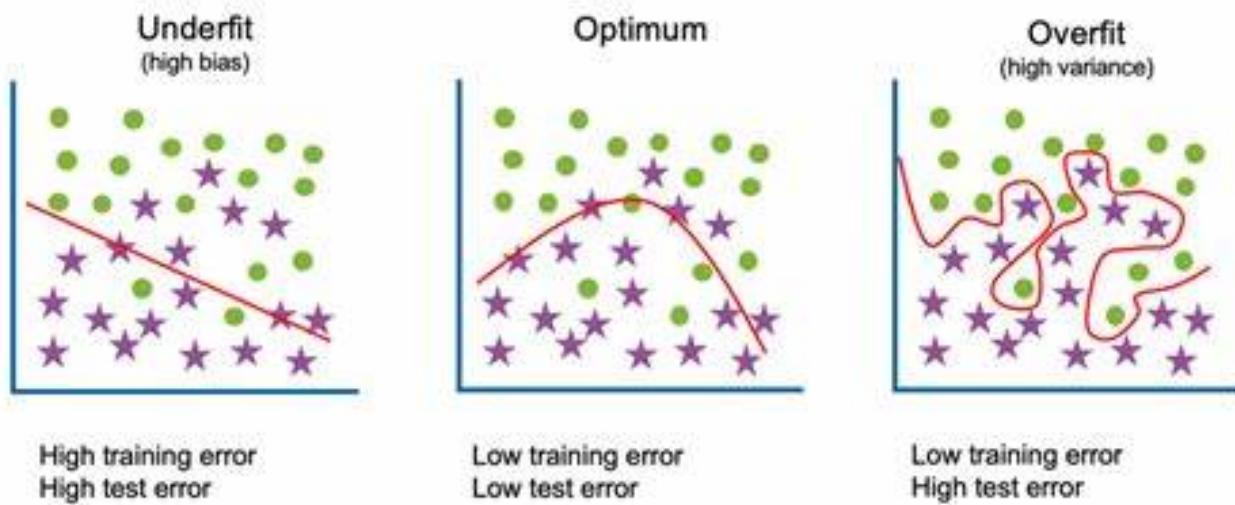
Το bias πρόκειται για υποθέσεις που γίνονται από ένα μοντέλο για να διευκολυνθεί η εκμάθηση μιας συνάρτησης. Στην πραγματικότητα είναι το ποσοστό σφάλματος των δεδομένων εκπαίδευσης. Όταν το ποσοστό σφάλματος έχει υψηλή τιμή, το ονομάζουμε υψηλό Bias και όταν το ποσοστό σφάλματος έχει χαμηλή τιμή, το ονομάζουμε χαμηλό Bias.

- Διακύμανση (Variance)

Η διαφορά μεταξύ του ποσοστού σφάλματος των δεδομένων εκπαίδευσης και των δεδομένων δοκιμής ονομάζεται διακύμανση. Εάν η διαφορά είναι υψηλή τότε ονομάζεται υψηλή διακύμανση και όταν η διαφορά των σφαλμάτων είναι χαμηλή τότε ονομάζεται χαμηλή διακύμανση. Συνήθως, θέλουμε να έχουμε χαμηλή διακύμανση για την επίτευξη σωστής γενίκευσης του μοντέλου μας.

Με βάση τους παραπάνω όρους, η υποπροσαρμογή ορίζεται το παρακάτω φαινόμενο: Ένα στατιστικό μοντέλο ή ένας αλγόριθμος μηχανικής μάθησης λέγεται ότι έχει υποπροσαρμογή όταν δεν μπορεί να συλλάβει την υποκείμενη τάση (trend) των δεδομένων, δηλαδή αποδίδει καλά μόνο σε δεδομένα εκπαίδευσης αλλά ελάχιστα σε δεδομένα δοκιμής. Η εμφάνισή της σημαίνει απλώς ότι το μοντέλο μας ή ο αλγόριθμος δεν προσαρμόζεται αρκετά καλά στα δεδομένα, ενώ συνήθως συμβαίνει όταν υπάρχουν λίγα δεδομένα για να φτιάξουμε ένα ακριβές μοντέλο ή όταν προσπαθούμε να φτιάξουμε ένα γραμμικό μοντέλο με λιγότερα μη γραμμικά δεδομένα. Σε τέτοιες περιπτώσεις, οι κανόνες του μοντέλου μηχανικής μάθησης είναι πολύ εύκολοι και ευέλικτοι για να εφαρμοστούν σε τόσο ελάχιστα δεδομένα και επομένως το μοντέλο πιθανόν να κάνει πολλές λανθασμένες προβλέψεις. Το φαινόμενο της υποπροσαρμογής μπορεί να αποφευχθεί με τη χρήση περισσότερων δεδομένων και με τη μείωση των χαρακτηριστικών, με την επιλογή συγκεκριμένων χαρακτηριστικών.

Αντιθέτως, το μοντέλο έχει υπερπροσαρμογή όταν δεν κάνει ακριβείς προβλέψεις στα δεδομένα δοκιμών. Συγκεκριμένα, όταν εκπαιδεύεται με μεγάλο πλήθος δεδομένων, το μοντέλο αρχίζει να μαθαίνει από το θόρυβο και τις ανακριβείς καταχωρίσεις στο σύνολο δεδομένων μας. Στις περιπτώσεις που η δοκιμή με δεδομένα δοκιμής οδηγεί σε υψηλή διακύμανση, τότε το μοντέλο δεν κατηγοριοποιεί σωστά τα δεδομένα, λόγω των πολλών λεπτομερειών και του θιρύβου. Οι αιτίες της υπερπροσαρμογής είναι οι μη παραμετρικές και μη γραμμικές μέθοδοι, επειδή αυτοί οι τύποι αλγορίθμων μηχανικής μάθησης έχουν μεγαλύτερη ελευθερία στην κατασκευή του μοντέλου με βάση το σύνολο δεδομένων και επομένως μπορούν πραγματικά να κατασκευάσουν μη ρεαλιστικά μοντέλα. Μια λύση για την αποφυγή της υπερπροσαρμογής είναι η χρήση ενός γραμμικού αλγορίθμου, αν έχουμε γραμμικά δεδομένα ή η χρήση των παραμέτρων όπως το μέγιστο βάθος αν χρησιμοποιούμε δέντρα αποφάσεων.



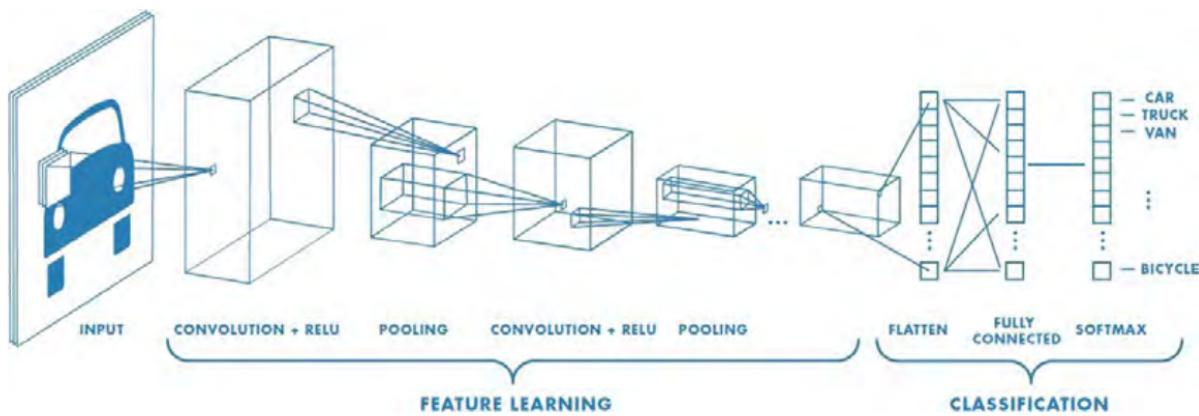
Σχήμα 2.13: Παραδειγματικό υποπροσαρμογής και υπερυποσαρμογής. [16]

## 2.4 Συνελικτικά Δίκτυα (Convolutional Neural Networks - CNN)

Τα συνελικτικά νευρωνικά δίκτυα (CNN) είναι μια κατηγορία νευρωνικών δικτύων με κύριο χαρακτηριστικό ότι δέχονται δισδιάστατη είσοδο ενώ χρησιμοποιούνται κυρίως στην ανάλυση φωτογραφιών. Ο λόγος για την ονομασία τους είναι το γεγονός ότι στη θέση του απλού μητρικού πολλαπλασιασμού, χρησιμοποιείται η πράξη της συνέλιξης, σε τουλάχιστον ένα επίπεδο τους[50]. Στην περίπτωση των CNN, δύο αναπαραστάσεις πινάκων εικόνων πολλαπλασιάζονται για την εξαγωγή χαρακτηριστικών όπως ακμές, όρια, σχήματα, υφές, αντικείμενα, κύκλοι κλπ. Τα αρχικά στρώματα συνεπώς ανιχνεύουν βασικά σχήματα και ακμές, ενώ τα επόμενα στρώματα γίνονται πιο εξελιγμένα, εντοπίζοντας τελικά συγκεκριμένα αντικείμενα όπως μάτια, αυτιά, σκύλους, γάτες και άλλα στοιχεία εντός της εικόνας. Τα CNN προτιμώνται έναντι των νευρωνικών δικτύων τροφοδότησης προς τα εμπρός (feed-forward neural networks) για την ανάλυση εικόνων λόγω της φύσης της εισόδου. Η ισοπέδωση (flatten) της εικόνας σε διάνυσμα και η τροφοδότηση της σε ένα perceptron πολλαπλών επιπέδων, δεν παράγει ακριβή αποτελέσματα. Επιπλέον, την τροφοδότηση μιας εικόνας του πραγματικού κόσμου μεγέθους 200x200x3 σε ένα feed-forward neural network, απαιτεί έναν τεράστιο αριθμό παραμέτρων και συνεπώς πολυάριθμους νευρώνες. Ως αποτέλεσμα, είναι εύκολο το δίκτυο να καταλήξει σε υπερυποσαρμογή.

Τα CNN, από την άλλη πλευρά, αναλύουν ένα τμήμα της εικόνας κάθε φορά και προχωρούν προς τα εμπρός μέχρι να αναλυθεί όλη η εικόνα. Ό αριθμός των νευρώνων και των παραμέτρων που απαιτούνται για την επεξεργασία της εικόνας μειώνεται σημαντικά. Εφαρμόζοντας σχετικά φίλτρα, το CNN καταγράφει τις χωρικές εξαρτήσεις εντός της εικόνας, μειώνοντας την εικόνα σε μια πιο συμπιεσμένη μορφή που είναι ευκολότερο να επεξεργαστεί χωρίς να θυσιάζονται κρίσιμα χαρακτηριστικά.

Συνοψίζοντας, τα CNN αποτελούν πολύτιμο εργαλείο στην ανάλυση εικόνων λόγω της ικανότητάς τους να εξάγουν χαρακτηριστικά και να ανιχνεύουν μοτίβα, οδηγώντας σε πιο ακριβή αποτελέσματα σε μικρότερο χρονικό διάστημα.

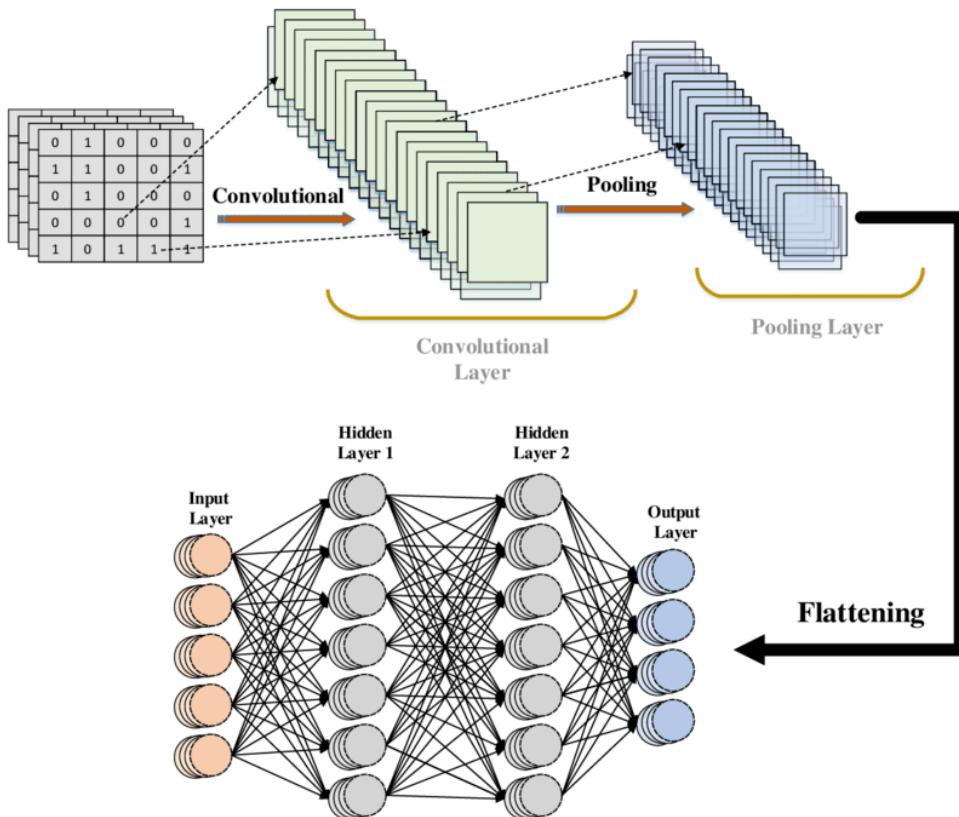


Σχήμα 2.14: Συνελικτικό δίκτυο για την αναγνώριση οχημάτων.

#### 2.4.1 Αρχιτεκτονική Συνελικτικών Δικτύων

Τα συνελικτικά νευρωνικά δίκτυα ακολουθούν την παρακάτω αρχιτεκτονική: Αποτελούνται από τρία διαφορετικά επίπεδα τα οποία προσδίδουν στο μοντέλο τα χαρακτηριστικά που προαναφέρθηκαν. Συγκεκριμένα, τα επίπεδα αυτά είναι:

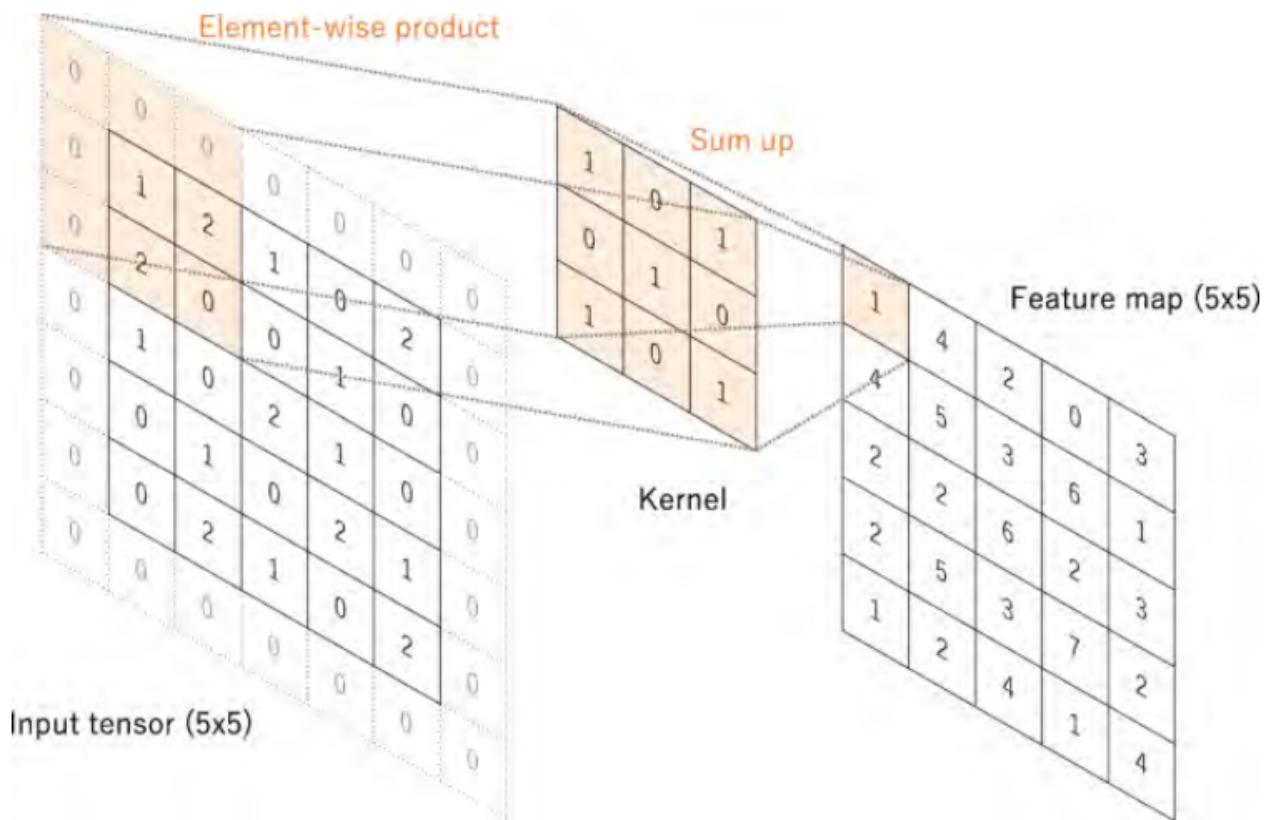
- Συνελικτικό επίπεδο (convolutional layer)
- Επίπεδο υποδειγματοληψίας (pooling layer)
- Πλήρως συδεδεμένο επίπεδο (fully connected layer)



Σχήμα 2.15: Αρχιτεκτονική ενός συνελικτικού δικτύου. [17]

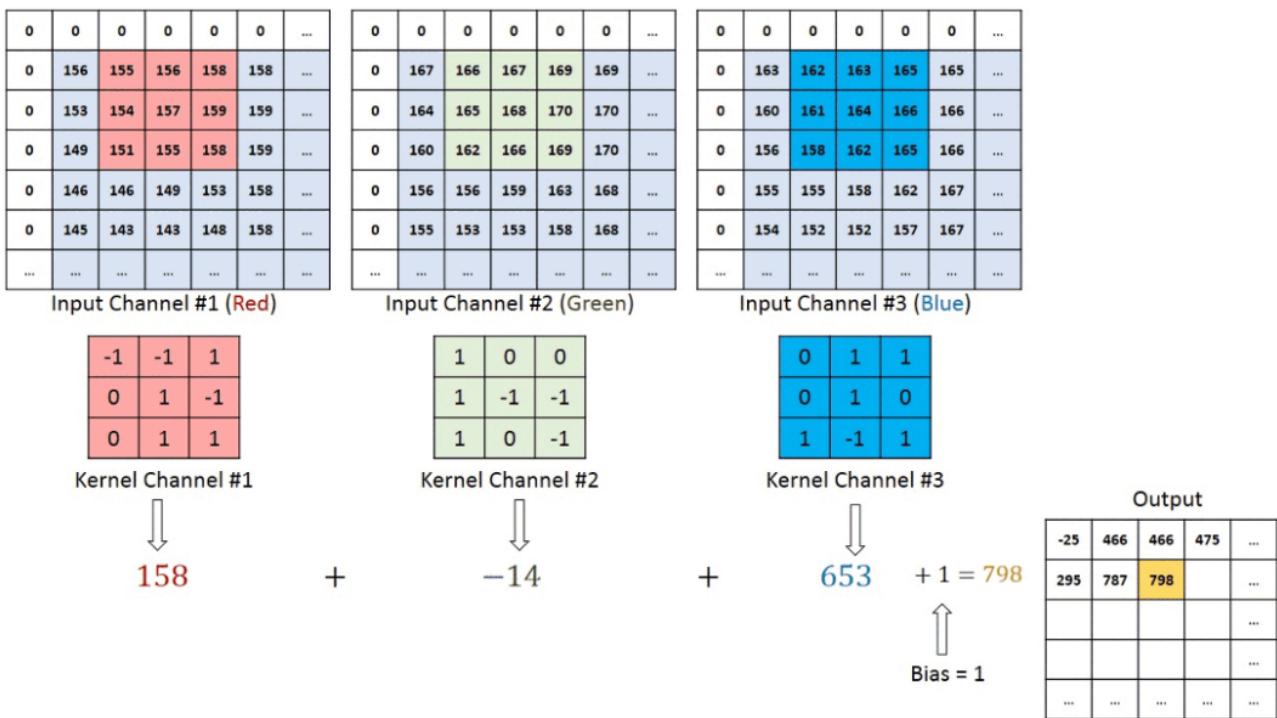
## Συνελικτικό επίπεδο

Το συνελικτικό επίπεδο χρησιμοποιείται για την εξαγωγή των διαφόρων χαρακτηριστικών από τις εικόνες εισόδου. Σε αυτό το επίπεδο εκτελείται η μαθηματική πράξη της συνέλιξης μεταξύ της εικόνας εισόδου και ενός φίλτρου (kernel) συγκεκριμένου μεγέθους  $M \times M$ . Ολισθαίνοντας το φίλτρο πάνω στην εικόνα εισόδου, λαμβάνεται το γινόμενο μεταξύ του φίλτρου και των τιμιάτων της εικόνας εισόδου σε σχέση με το μέγεθος του φίλτρου ( $M \times M$ ). Η έξοδος ονομάζεται Χάρτης χαρακτηριστικών (feature map), ο οποίος μας δίνει πληροφορίες σχετικά με την εικόνα, όπως τις γωνίες και τις ακμές. Το μέγεθος του έχει προφανώς μικρότερο μέγεθος από εκείνο της εικόνας εισόδου. Ωστόσο, αν θέλουμε το feature map να έχει τις ίδιες διαστάσεις με την είσοδο, καλούμαστε να την προσανέξουμε με μηδενικά (zero padding). Στο παρακάτω σχήμα αναφαίνεται η παραπάνω διαδικασία.



Σχήμα 2.16: Πρόξει συνέλιξης με υπδενική συμπλήρωση (zero padding) ώστε να διατηρούνται οι διαστάσεις. Σε αυτό το παράδειγμα, το μέγεθος του πυρήνα και το βήμα ορίζονται ως  $3 \times 3$  και 1, αντίστοιχα. [18]

Μια ακόμη σημαντική παράμετρος κατά την συνέλιξη είναι το βήμα (stride) με το οποίο ολισθαίνει το φίλτρο πάνω στην εικόνα. Θεωρώντας stride = 1, το kernel μετακινείται κάθε φορά κατά μια θέση. Ωστόσο, αποτελεί συχνό φαινόμενο ο ορισμός του βήματος σε υψηλότερους αριθμούς. Με αυτό τον τρόπο, εκτελούνται λιγότερες συνελίξεις, μειώνοντας έτσι το συνολικό υπολογιστικό φόρτο. Μια ακόμη σημαντική λεπτομέρεια αποτελούν οι εικόνες RGB. Όπως γνωρίζουμε, αυτού του τύπου εικόνες αποτελούνται από τρία κανάλια (channels), ένα για το κόκκινο, ένα για το πράσινο και ένα για το μπλε. Έτσι, το kernel φέρει επίσης τρία κανάλια και η παραπάνω διαδικασία εφαρμόζεται σε κάθε κανάλι ξεχωριστά. Στο τέλος, τα εικονοστοιχεία εξόδου από κάθε χρωματική συνιστώσα προστίθεται και στην συνέχεια προστίθεται ακόμη μια τιμή bias η οποία είναι η ίδια για όλα τα αθροίσματα. Στο παρακάτω σχήμα αναφαίνεται η συνέλιξη γιας RGB εικόνας MxNx3 και ενός φίλτρου 3x3x3.

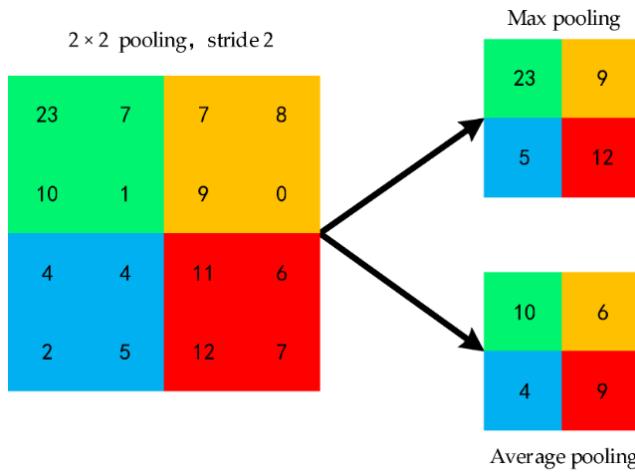


Σχήμα 2.17: Πράξη συνέλιξης σε εικόνα RGB. [19]

Η έξοδος του συνελικτικού επιπέδου συνήθως τροφοδοτείται σε μια συνάρτηση ενεργοποίησης. Αυτό πραγματοποιείται για την εισαγωγή μη γραμμικότητας στο σύστημα, καθώς οι πράξεις που εκτελεί το μοντέλο ως αυτό το σημείο αποτελούν απλές γραμμικές πράξεις (πολλαπλασιασμός πινάκων και πρόσθιμης bias). Η κύρια συνάρτηση ενεργοποίησης που χρησιμοποιείται είναι η ανυψωμένη γραμμική μονάδα καθώς ερευνητές έχουν αναδείξει σημαντικά ταχύτερη εκπαίδευση του συστήματος. Επιπλέον, επιλύει το πρόβλημα των "εξαφανιζόμενων κλίσεων" (vanishing gradients) το οποίο προκύπτει όταν οι τιμές της κλίσεις μειώνονται εκθετικά και η εκπαίδευση συνεχίζεται με αρκετά αργότερο χρονικό. Αργότερα, ο χάρτης χαρακτηριστικών τροφοδοτείται σε άλλα επίπεδα για να μάθει το μοντέλο άλλα χαρακτηριστικά της εικόνας εισόδου.

Επίπεδο υποδειγματοληψίας (pooling layer)

Το επίπεδο υποδειγματοληψίας ή αλλιώς pooling layer αποτελεί ένα ακόμη σημαντικό επίπεδο των συνελικτικών νευρωνικών δικτύων. Η κύρια λειτουργία του, πέρα από την εξαγωγή σημαντικών χαρακτηριστικών της εικόνας, είναι η περαιτέρω μείωση της εξόδου του δικτύου, δηλαδή του feature map, και συνεπώς η ελάττωση της απαιτούμενης υπολογιστικής ισχύς [51]. Στην βιβλιογραφία συναντάμε δύο είδη υποδειγματοληψίας, το max pooling και το average pooling τα οποία αναφαίνονται και στην παρακάτω εικόνα. Το max pooling πρόκειται για ένα είδος μη γραμμικής υποδειγματοληψίας. Πιο συγκεκριμένα, κατά την σάρωση κάθε υποπεριοχής της εικόνας, επιλέγουμε το στοιχείο με την μέγιστη τιμή που βρίσκεται εντός της εκάστοτε υποπεριοχής. Στη συνέχεια, αντικαθιστούμε όλα τα εικονοστοιχεία της υποπεριοχής αυτής με την επιλεγμένη τιμή. Με τον αντίστοιχο τρόπο εφαρμόζεται και το average pooling, με την βασική διαφορά ότι η αντικατάσταση όλων των εικονοστοιχείων της περιοχής γίνεται με την τιμή του μέσου όρου των τιμών της. Στο παρακάτω σχήμα αναφαίνεται η διαδικασία που περιγράφηκε.



Σχήμα 2.18: Average pooling και max pooling σε μια εικόνα. [20]

**Πλήρως συνδεδεμένο επίπεδο (fully connected layer)** Το τρίτο βασικό επίπεδο που ακολουθεί τα συνελικτικά και τα επίπεδα υποδειγματοληψίας, είναι το πλήρως συνδεδεμένο επίπεδο (fully connected layer). Η έξοδος του επιπέδου υποδειγματοληψίας γίνεται flattened και τροφοδοτείται στην συνέχεια σε ένα ή παραπάνω fully connected layers. Αξίζει να σημειωθεί ότι στην προκειμένη περίπτωση δεν υφίσταται πρόβλημα υπολογιστικού κόστους κατά την διαδικασία του flattening καθώς η εικόνα έχει πλέον πολύ μικρότερες διαστάσεις από την αρχική. Τα πλήρως συνδεδεμένα επίπεδα συνδέουν κάθε νευρώνα σε ένα επίπεδο με κάθε νευρώνα σε ένα άλλο επίπεδο. Είναι το ίδιο με ένα παραδοσιακό πολυστρωματικό νευρωνικό δίκτυο perceptron (MLP). Ο πεπλατυσμένος πίνακας περνά από ένα πλήρως συνδεδεμένο στρώμα για την ταξινόμηση των εικόνων. [52]

## 2.4.2 Εφαρμογές Συνελικτικών Δικτύων

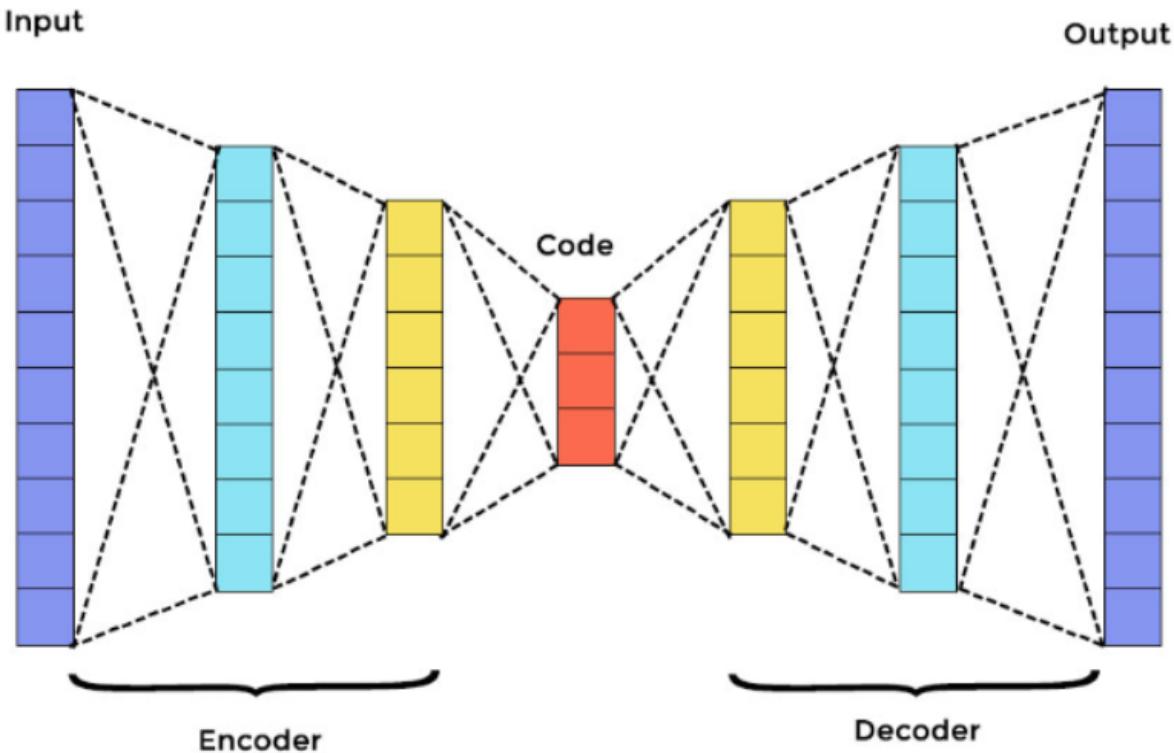
## 2.5 Autoencoders

Όπως έχουμε αναλύσει παραπάνω, υπάρχουν πολλοί τρόποι για τη συμπίεση δεδομένων και καθαρά στατιστικές μέθοδοι όπως η ανάλυση κύριων συνιστωσών (PCA), μπορούν να βοηθήσουν στον εντοπισμό των βασικών χαρακτηριστικών που ευθύνονται για τη μεταβλητότητα των δεδομένων, και να τα χρησιμοποιήσουν για την αναπαράσταση των πληροφοριών με λιγότερα bits. Αυτός ο τρόπος συμπίεσης ονομάζεται "Μείωση διαστάσεων". Ωστόσο, αυτή η λύση PCA προσφέρει μόνο κωδικοποίηση με γραμμικά ασυγχέτιστα χαρακτηριστικά.

Ο δημοφιλής τύπος νευρωνικού δικτύου που μπορεί να μάθει κωδικοποίηση με γραμμικών χαρακτηριστικών είναι ο αυτοκωδικοποιητής (autoencoder), ο οποίος χρησιμοποιείται συνήθως για μάθηση χωρίς επίβλεψη. Πιο συγκεκριμένα, στόχος του autoencoder είναι η μείωση των διαστάσεων των δεδομένων εισόδου με σκοπό τον εντοπισμό και την καταγραφή των πιο σημαντικών στοιχείων τους. Συνεπώς η χρήση των autoencoders είναι κυρίως αποτελεσματική σε περιπτώσεις που ο επιφανειακός εντοπισμός κάποιας σχέσης μεταξύ των δεδομένων δεν είναι εμφανής αλλά παρουσιάζεται έπειτα από την μείωση των διαστάσεων.

### 2.5.1 Αρχιτεκτονική των Autoencoders

Οι autoencoders αποτελούνται από τρία κύρια μέλη: Τον κωδικοποιητή (encoder), το σημείο συμφόρησης/κώδικας (bottleneck/code) και τον αποκωδικοποιητή (decoder).



Σχήμα 2.19: Αρχιτεκτονική Autoencoder[21].

Η επεξήγηση αυτών των σταδίων θα γίνει με την πορεία που ακολουθούν τα δεδομένα όταν εισέρχονται στο δίκτυο.

- **Κωδικοποιητής/Encoder**

Ο ρόλος του κωδικοποιητή είναι η συμπίεση των δεδομένων σε μια μιορφή κάθικα. Ο τρόπος υλοποίησης του παραπάνω γίνεται με την εισαγωγή πολλαπλών συνελικτικών επιπέδων και επιπέδων υποδειγματοληψίας (pooling layers) το κάθε ένα από τα οποία βοηθούν στην μείωση των διαστάσεων των δεδομένων εισόδου. Κατά αυτόν τον τρόπο αυτό, επιτυγχάνεται αποθορυβούστηκη των δεδομένων, διατηρώντας μόνο τα μέρη που είναι αρκετά σημαντικά για να κωδικοποιηθούν. Σημειώνεται ότι ο βαθμός της μείωσης των διαστάσεων εξαρτάται από υποπαραμέτρους που έχουν αναφερθεί στο προηγούμενο υποκεφάλαιο.

- **Σημείο Συμφόρησης/Κάθικας (Bottleneck/Code)**

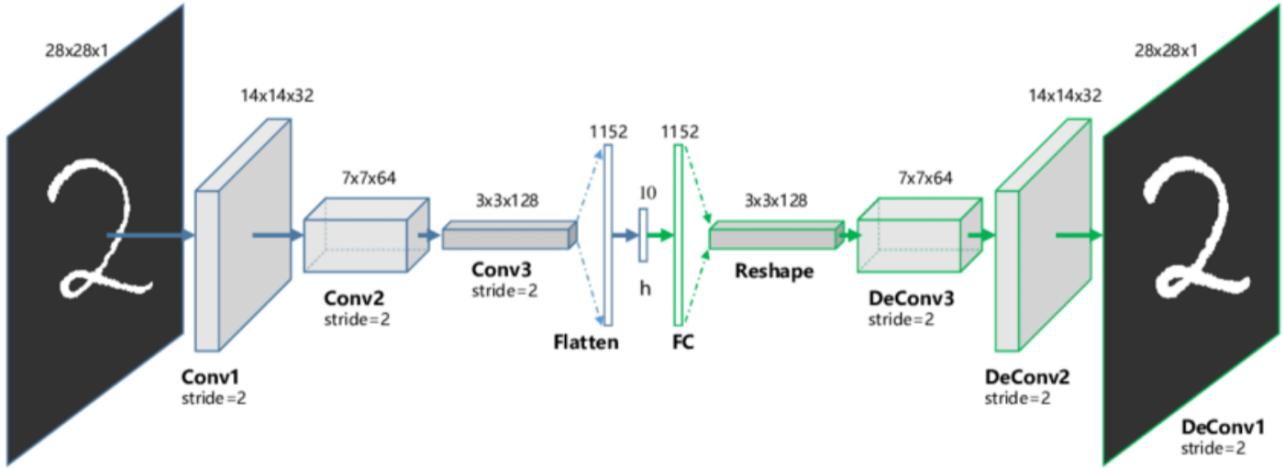
Το στρώμα του νευρωνικού δικτύου που έχει τις λιγότερες διαστάσεις και βρίσκεται συνήθως στο μέσο όλων των στρωμάτων, καλείται το σημείο συμφόρησης (bottleneck). Το bottleneck λοιπόν, ρυθμίζει την ύψη των δεδομένων από τον κωδικοποιητή προς τον αποκωδικοποιητή, επιτρέποντας την διέλευση μόνο των δεδομένων που επιθυμούμε. Αν μας ενδιαφέρει να κρατήσουμε περισσότερες πληροφορίες για μια συγκεκριμένη περίπτωση, θα χρειαστούμε μεγαλύτερο μέγεθος κάθικα. Αυτό πραγματοποιείται με τον σωστό ορισμό της υπερ-παραμέτρου "μέγεθος κάθικα", καθώς καθορίζει πόσα δεδομένα θα κωδικοποιηθούν. Παραδείγματος χάρη, ένας πολύ μεγάλος κάθικας θα αναπαριστά περισσότερο θόρυβο, αλλά ένας πολύ μικρός κάθικας μπορεί να μην αναπαριστά με ακρίβεια τα εκάστοτε δεδομένα.

- **Αποκωδικοποιητής (Decoder)**

Ο αποκωδικοποιητής είναι το τελικό στάδιο του μοντέλου. Η λειτουργία του είναι η ανακατασκευή της εισόδου με βάση τον κάθικα που έχει λάβει από το bottleneck με αύξηση των διαστάσεων των δεδομένων. Πιο συγκεκριμένα, καλείται σε ιδανικές περιπτώσεις, να

επιτύχει τέλεια ανακατασκευή της εισόδου από τα συμπτεισμένα και χαμηλά σε διαστάσεις δεδομένα που έχει λάβει. Ωστόσο, αυτό είναι αδύνατο και για αυτό, στόχος του δικτύου είναι η ελαχιστοποίηση του σφάλματος ανακατασκευής, δηλαδή της διαφοράς μεταξύ των δεδομένων εισόδου και εξόδου.

Παρακάτω μπορούμε να παρατηρήσουμε αναλυτικότερα τα στάδια που προαναφέρθηκαν, σε μια εικόνα ενός συνελικτικού autoencoder.



Σχήμα 2.20: Στάδια συνελικτικού autoencoder.[22]

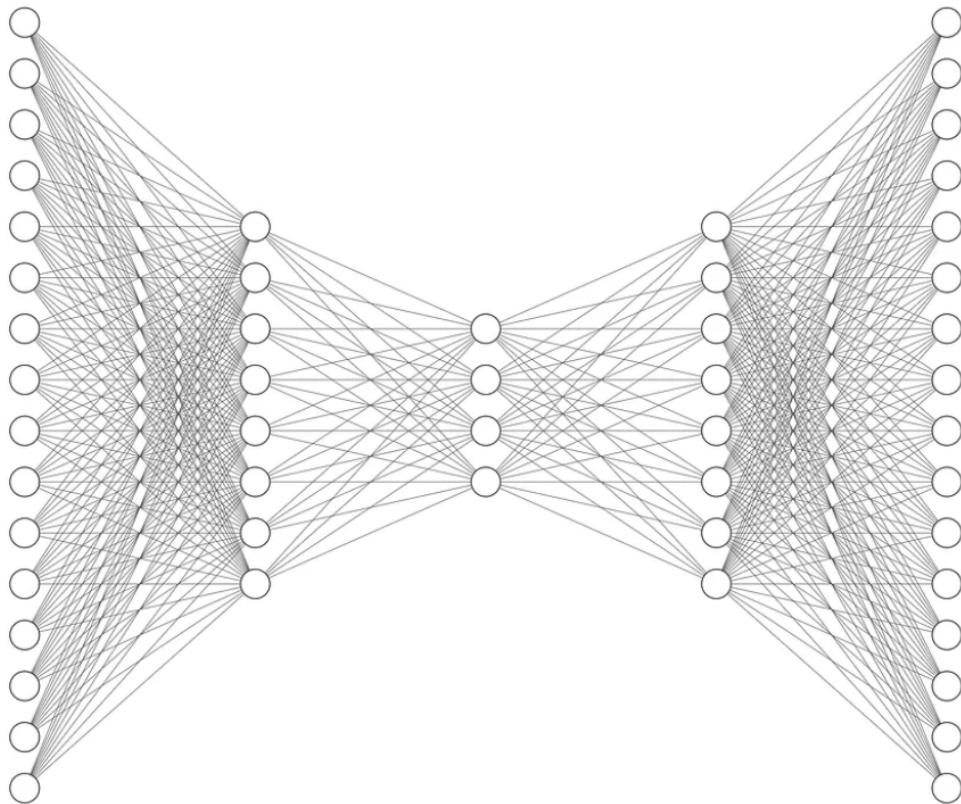
## 2.5.2 Κατηγορίες Autoencoders

Υπάρχει μεγάλο πλήθως τύπων autoencoders αλλά οι πιο ευρέως χρησιμοποιούμενοι είναι:

- Undercomplete autoencoder
- Denoising autoencoder (DAE)
- Sparse autoencoder
- Contractive autoencoder (CAE)

### 2.5.2.1 Undercomplete autoencoder

Ο undercomplete autoencoder πρόκειται για μια από τις πιο απλές μορφές των AE. Στόχος του είναι η λήψη των πιο σημαντικών χαρακτηριστικών στα δεδομένα μέσω της μείωσης των διαστάσεων. Συνίθως ως είσοδο έχουμε μια εικόνα την οποία προσπαθεί να ανακατασκευάσει ο αποκωδικοποιητής μέσω του κώδικα που έχει λάβει από το bottleneck. Πρόκειται φυσικά για μη επιβλεπόμενη μάθηση καθώς ο στόχος του δικτύου είναι η ανακατασκευή της ίδια εικόνας και συνεπώς δεν χρειάζεται κάποια ετικέτα (label).

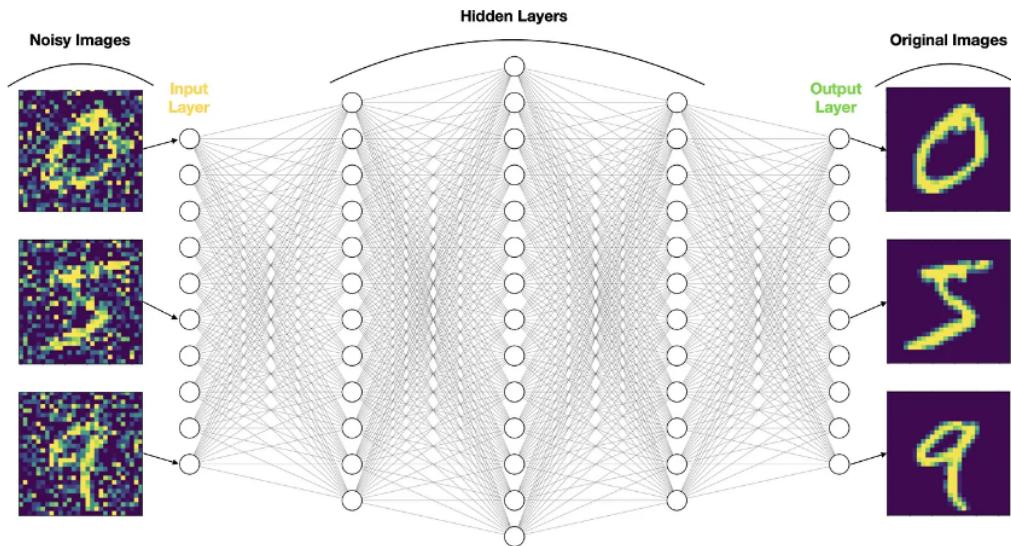


Σχήμα 2.21: Αρχιτεκτονική undercomplete autoencoder.

Η συνάρτηση σφάλματος με την οποία εκπαιδεύεται ο undercomplete autoencoder είναι το σφάλμα ανακατασκευής (reconstruction error) το οποίο δίνει μια μέτρηση για το "πόσο καλά" έχει καταφέρει το δίκτυο να ανακατασκευάσει την αρχική εικόνα.

### 2.5.2.2 Denoising autoencoders (DAE)

Η λειτουργία το συγκεκριμένου autoencoder φαίνεται και από την ονομασία του. Σκοπός του είναι η αφαίρεση του θορύβου από την εικόνα εισόδου. Αξίζει να σημειωθεί ότι στην συγκεκριμένη περίπτωση δεν είναι δυνατή η κατασκευή μιας απλής αντιστοίχησης για την απομνημόνευση των δεδομένων εκπαίδευσης καθώς δεν έχουμε πλέον ίδια έξιδο με είσοδο εφόσον στις εικόνες εισόδου έχει προστεθεί θόρυβος [53]. Αυτοί οι αυτόματοι κωδικοποιητές όπως αναφέρομε λαμβάνουν μια μερικώς αλλοιωμένη είσοδο κατά την εκπαίδευση για να ανακτήσουν την αρχική μη αλλοιωμένη είσοδο. Το μοντέλο μαθαίνει ένα διανυσματικό πεδίο για την απεικόνιση των δεδομένων εισόδου σε μια χαμηλότερη διάσταση που περιγράφει τα φυσικά δεδομένα για να εξαλείψει τον προστιθέμενο θόρυβο.

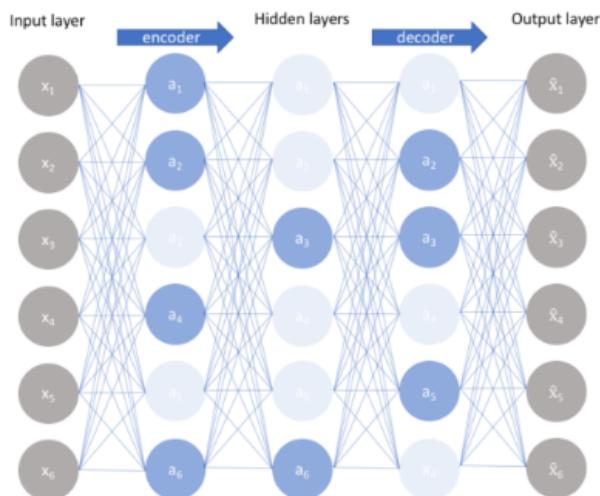


Σχήμα 2.22: Αρχιτεκτονική DAE [23].

### 2.5.2.3 Sparse autoencoders (SAE)

Ο sparse autoencoder αποτελεί ένα είδος αυτοκωδικοποιητή που δημιουργεί το σημείο συμφώρουσης πληροφοριών χρησιμοποιώντας την έννοια της αραιότητας. Η αραιότητα αναφέρεται σε έναν περιορισμό που προωθεί τον autoencoder να χρησιμοποιεί μόνο ένα μικρό αριθμό των διαθέσιμων νευρώνων του, ενθαρρύνοντας έτσι την ανάπτυξη πιο πολύτιμων και ισχυρών χαρακτηριστικών.

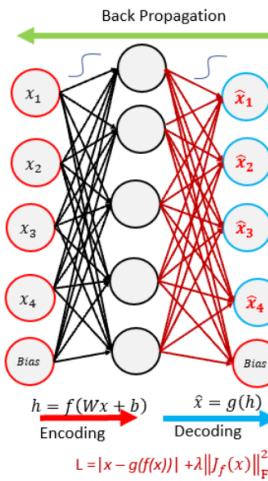
Η αραιότητα επιτυγχάνεται με το σχεδιασμό της συνάρτησης σφάλματος έτσι ώστε οι ενεργοποιήσεις εντός ενός στρώματος να τιμωρούνται. Αυτό ουσιαστικά σημαίνει ότι δεν μπορούν να πυροδοτηθούν ταυτόχρονα όλοι οι νευρώνες στο κρυφό στρώμα. Αυτός ο περιορισμός μπορεί να εφαρμοστεί με τη χρήση τεχνικών όπως η κανονικοποίηση L1 ή η απόκλιση Kullback-Leibler (KL) μεταξύ της αναμενόμενης μέσης ενεργοποίησης των νευρώνων και μιας προκαθορισμένης ιδανικής κατανομής.



Σχήμα 2.23: Αρχιτεκτονική SAE [24].

### 2.5.2.4 Contractive autoencoders (CAE)

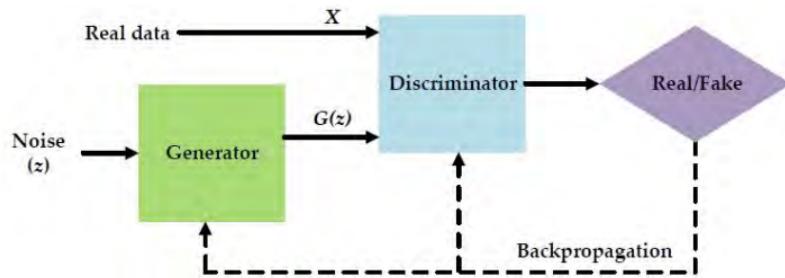
Ο στόχος ενός contractive autoencoder είναι η κατασκευή μιας ισχυρής αναπαράστασης, λιγότερο ευαίσθητη σε μικρές διακυμάνσεις των δεδομένων[54]. Η ανθεκτικότητα της επιτυγχάνεται με την εφαρμογή ενός όρου ποινής στη συνάρτηση σφάλματος. Ο CAE πρόκειται για ακόμη μια άλλη τεχνική ρύθμισης όπως ο SAE και DAE. Ωστόσο, αυτός ο ρυθμιστής αντιστοιχεί στη νόρμα Frobenius του Ιακωβιανού πίνακα των ενεργοποιήσεων του κωδικοποιητή σε σχέση με την είσοδο. Η νόρμα Frobenius του Ιακωβιανού πίνακα για το κρυφό στρώμα υπολογίζεται σε σχέση με την είσοδο και είναι ουσιαστικά το άθροισμα των τετραγώνων όλων των στοιχείων. Ένα πλεονέκτημα αυτής της μεθόδου είναι ότι το μοντέλο μαθαίνει μια κωδικοποίηση στην οποία παρόμοιες είσοδοι έχουν παρόμοιες κωδικοποίησεις. Ως εκ τούτου, αναγκάζουμε το μοντέλο να μάθει πώς να συστέλλει μια γειτονιά εισόδων σε μια μικρότερη γειτονιά εξόδων.



Σχήμα 2.24: Αρχιτεκτονική CAE [25].

## 2.6 Αναγεννητικά Αντιπαραθετικά Δίκτυα (GANs)

Τα αναγεννητικά αντιπαραθετικά δίκτυα είναι μια σχετικά νέα και ισχυρή κατηγορία της μηχανικής μάθησης. Η ιδέα αναπτύχθηκε αρχικά από τον Ian Goodfellow και τους συναδέλφους του τον Ιούνιο του 2014 [55], ενώ χρησιμοποιούνται στην μη επιβλεπόμενη μηχανική μάθηση. Τα GANs αποτελούνται από δύο μέρη, το κάθε ένα από τα οποία είναι νευρωνικό δίκτυο. Συγκεκριμένα, τα GANs εμπεριέχουν το αναγεννητικό (generator) και το διαχωριστικό δίκτυο (discriminator). Ο ρόλος του generator είναι η παραγωγή δεδομένων (όπως εικόνες, μουσική, ομιλία κλπ.), ενώ το διαχωριστικό δίκτυο αναλαμβάνει την αξιολόγηση της γνησιότητας των παραγόμενων αυτών δεδομένων. Η συνάρτηση απώλειας του μοντέλου ακολουθεί την λογική ενός min-max παιγνίου, οδηγώντας το δίκτυο σε ανταγωνισμό. Ειδικότερα, όσο το διαχωριστικό δίκτυο καταφέρνει να ταξινομεί τα δεδομένα με υψηλότερη ακρίβεια, το αναγεννητικό, προσπαθεί να κατασκευάζει καλύτερα συνθετικά δεδομένα με σκοπό να το "ξεγελάσει". Τα μοντέλα GAN ακολουθούν την παρακάτω δομή:



Σχήμα 2.25: Αρχιτεκτονική των αναγεννητικών αντιπαραθετικών δικτύων [26].

### 2.6.1 Αρχιτεκτονική των Αναγεννητικών Αντιπαραθετικών Δικτύων

Το διαχωριστικό δίκτυο είναι ένας απλός ταξινομητής που αναλαμβάνει τον ρόλο της διάκρισης των πραγματικών δεδομένων από τα δεδομένα που συνθέτει το αναγεννητικό δίκτυο. Η αρχιτεκτονική του μπορεί να διαφέρει ανάλογα με τον τύπο των δεδομένων που καλείται να ταξινομίσει, ωστόσο όταν πρόκειται για εικόνες, σχεδόν πάντοτε χρησιμοποιούνται τα συνελικτικά δίκτυα. Τα δεδομένα εκπαίδευσης για τον διαχωριστή προέρχονται από δύο πηγές: πραγματικές περιπτώσεις δεδομένων, οι οποίες χρησιμεύουν ως θετικά παραδείγματα κατά την εκπαίδευση, και ψεύτικες περιπτώσεις δεδομένων που δημιουργούνται από τον generator, οι οποίες αντίστοιχα χρησιμεύουν ως αρνητικά. Κατά τη διάρκεια της εκπαίδευσης, ο διαχωριστής ταξινομεί τόσο τα πραγματικά όσο και τα συνθετικά δεδομένα, ενώ η συνάρτηση απώλειας των τιμώρει για την εσφαλμένη ταξινόμηση ενός πραγματικού παραδείγματος ως ψεύτικου ή ενός ψεύτικου ως πραγματικού. Τέλος, τα βάρη του ενημερώνονται μέσω της οπισθοδιάδοσης.

Το αναγεννητικό δίκτυο προσπαθεί να συνθέσει δεδομένα τα οποία φαίνονται ότι προέρχονται από την ίδια κατανομή με τα πραγματικά. Η είσοδος αυτού του δικτύου είναι τυχαίος θόρυβος, του οποίου η κατανομή είναι συνήθως η ομοιόμορφη. Από αυτήν την είσοδο, ο generator προσπαθεί να κατασκευάσει δεδομένα τα οποία θα καταφέρουν να "ξεγελάσουν" τον discriminator. Πρέπει να σημειωθεί ότι το αναγεννητικό δίκτυο δεν συνδέεται άμεσα με την συνάρτηση απώλειας του καθώς η έξοδος του τροφοδοτείται στο διαχωριστικό δίκτυου του οποίου την έξοδο προσπαθούμε να επηρεάσουμε. Πιο αναλυτικά, κατά την εκπαίδευση, παγώνουν τα βάρη του διαχωριστικού δικτύου, το οποίο καλείται να κατατάξει την έξοδο του generator ως πραγματική ή συνθετική. Άν η κατάταξη του είναι ορθή (το αναγεννητικό δίκτυο δεν κατάφερε να παράξει δεδομένα ικανά να "ξεγελάσουν" το διαχωριστικό δίκτυο), η συνάρτηση απώλειας του generator το τιμώρει και ανανεώνει τα βάρη του. Η διαδικασία της εκπαίδευσης είναι η παρακάτω:

- 1. Το αναγεννητικό δίκτυο εκπαιδεύεται για μια ή περισσότερες εποχές
- 2. Το διαχωριστικό δίκτυο εκπαιδεύεται για μια ή περισσότερες εποχές
- 3. Η διαδικασία επαναλαμβάνεται έως ότου το διαχωριστικό δίκτυο να επιτύχει ακρίβεια 50%. Πλέον, ο discriminator δεν θεωρείται χρήσιμος καθώς η ακρίβειά του ισοδυναμεί με τυχαία επιλογή.

Η συνάρτηση απώλειας του αναγεννητικού αντιπαραθετικού δικτύου είναι:

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log(1 - D(G(z)))]$$

- Το  $x$  αντιπροσωπεύει τα πραγματικά δεδομένα.
- Το  $z$  αντιπροσωπεύει τα σημεία θορύβου.
- Το  $G(z)$  αντιπροσωπεύει τα δεδομένα που παράγει το παραγωγικό δίκτυο.

- Το  $D(x)$  και  $D(G(z))$  αντιπροσωπεύουν τις εκτιμήσεις του διαχωριστικού δικτύου για τις πιθανότητες που  $x$  και  $G(z)$  είναι πραγματικά αντίστοιχα.

## 2.6.2 Εφαρμογές των Αναγεννητικών Αντιπαραθετικών Δικτύων

Μια από τις πιο ενδιαφέρουσες και εντυπωσιακές εφαρμογές των GANs βρίσκεται στην σύνθεση εικόνων. Λόγω του μοντέλου που περιγράφει παραπάνω, τα GANs επιτυγχάνουν σύνθεση ρεαλιστικών εικόνων, οι οποίες σχεδόν δεν διακρίνονται από τις πραγματικές. Πρόσφατα έχουν υπάρξει αρκετά προγράμματα τα οποία μέσα από μια λεκτική περιγραφή της εικόνας από τον χρήστη, καταφέρνουν να παράγουν εκπληκτικές εικόνες. Παρακάτω ακολουθούν μερικές εικόνες οι οποίες έχουν δημιουργηθεί από το Midjourney:



(α) Λεκτική περιγραφή: portrait. A single tear down her cheek...



(β) Λεκτική περιγραφή: Imagine an opulent Louis Vuitton Alma bag, crafted from luxurious Monogram Vernis leather ...



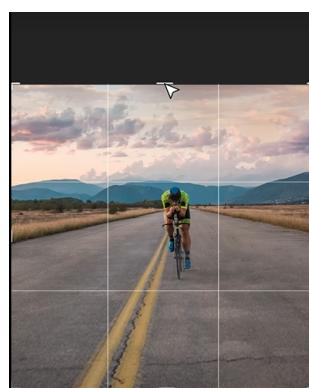
(γ) Λεκτική περιγραφή: National Geographic Photography, Close up on an Exotic Animal in the Congo, stunning photography, large format film, intense...

Σχήμα 2.26: Τρείς εικόνες που έχουν δημιουργηθεί από το Midjourney[27].

Μια ακόμη εφαρμογή των GANs η οποία βρίσκεται υπό ραγδαία ανάπτυξη είναι το image inpainting το οποίο κατατάσσεται στην κατηγορία της αποκατάστασης εικόνας. Το image inpainting προσπαθεί να γεμίζει χαμένες περιοχές της εικόνας ή να αντικαταστήσει υπάρχοντες. Η γνωστή εφαρμογή επεξεργασίας εικόνας "Photoshop" έχει ενσωματώσει στο περιβάλλον της την λειτουργία "generative fill" η οποία δίνει την δυνατότητα στον χρήστη του image inpainting.



(α) Αρχική εικόνα.



(β) Επιλογή του δρόμου και εισαγωγή κίτρινων γραμμών στο σημείο επιλογής.



(γ) Επιλογή του άνω μέρους της εικόνας και επέκταση της.

Σχήμα 2.27: Εφαρμογή του image inpainting σε εικόνα.



### 3. Συμπίεση Εικόνας Με Χρήση Τεχνικών Βαθιάς Μάθησης

#### 3.1 Βιβλιογραφική Ανασκόπηση

Οι George Toderici et al. στην εργασία τους [56] χρησιμοποίησαν τον τύπο νευρωνικών δικτύων LSTMs για την συμπίεση ψηφιακών εικόνων. Συγκεκριμένα, κατασκεύασαν τρεις διαφορετικές αρχιτεκτονικές, όπου η κάθε μια "χτίζει" πάνω στην προηγούμενη και βελτιώνει το ποσοστό συμπίεσης. Ωστόσο, όλες οι εκδοχές ακολουθούν την παρακάτω λογική:

- **Συνάρτηση E:** Δέχεται ένα κομμάτι εικόνας ως είσοδο και παράγει την κωδικοποιημένη μορφή του.
- **Συνάρτηση Διαδικοποίησης (binarization) B:** Οι τιμές της εισόδου αυτής της συνάρτησης αντιστοιχίζονται σε τιμές στο διάστημα (-1,1).
- **Συνάρτηση αποκωδικοποίησης D:** Λαμβάνει την δυαδική αναπαράσταση όπου έχει κατασκευάσει η συνάρτηση B και παράγει το ανακατασκευασμένο κομμάτι εικόνας εξόδου.

Στη πρώτη αρχιτεκτονική όπου εφαρμόστηκε, οι συναρτήσεις E και D τέθηκαν ως στοιβαγμένα πλήρως συνδεδεμένα επίπεδα των οποίων ο αριθμός εξόδων ορίστηκε ο σταθερός αριθμός 512, και έγινε χρήση της συνάρτησης υπερβολικής εφαπτομένης ( $\tanh$ ) ως εισαγωγή μη γραμμικότητας. Σε αυτή την αρχιτεκτονική, ακολούθησαν δύο προσεγγίσεις:

- Στην πρώτη προσέγγιση, τα βάρον μοιράζονται σε όλα τα στάδια του μοντέλου. Σε ένα κομμάτι 8x8 μιας εικόνας 32x32 με μέγεθος στόχου 128B, η συμπιεσμένη εικόνα είχε SSIM = 0.48.
- Στην δεύτερη προσέγγιση, το μοντέλο μάθαινε ξεχωριστά βάρον σε κάθε στάδιο. Σε ένα κομμάτι 8x8 μιας εικόνας 32x32 με μέγεθος στόχου 128B, η συμπιεσμένη εικόνα είχε SSIM = 0.75.

Στη συνέχεια, στην δεύτερη αρχιτεκτονική έγινε χρήση μοντέλων LSTM στις συναρτήσεις E και D. Στην συνάρτηση E υπάρχει ένα πλήρως συνδεδεμένο επίπεδο, το οποίο ακολουθείται από δύο στοιβαγμένα LSTM επίπεδα. Αντίθετα, η συνάρτηση D έχει την ανάποδη δομή, δύο στοιβαγμένα επίπεδα LSTM ακολουθούνται από ένα πλήρως συνδεδεμένο επίπεδο με την συνάρτηση υπερβολικής εφαπτομένης για την εισαγωγή μη γραμμικότητας. Τα LSTM μπλοκ αποτελούνταν και σε αυτή την υλοποίηση από 512 επίπεδα. Σε ένα κομμάτι 8x8 μιας εικόνας 32x32 με μέγεθος στόχου 128B, η συμπιεσμένη εικόνα είχε SSIM = 0.81. Η τρίτη υλοποίηση ονομάζεται συνελικτική/αποσυνελικτική LSTM συμπίεση όπου συμπεριλαμβάνονται οι πράξεις της συνέλιξης και αποσυνέλιξης μαζί με τα LSTM μοντέλα. Σε ένα κομμάτι 8x8 μιας εικόνας 32x32 με μέγεθος στόχου 128B, η συμπιεσμένη εικόνα είχε SSIM = 0.87.

Σε μία άλλη εργασία, οι George Toderici et al.[57] χρησιμοποίησαν νευρωνικά δίκτυα τύπου LSTM για την κατασκευή ενός μοντέλου συμπίεσης εικόνας, ανταγωνιστικό σε εικόνες κάθε μεγέθους. Επιπλέον, το μοντέλο αυτό παρέχει την δυνατότητα μεταβλητού λόγου συμπίεσης δίχως την

επανεκπαίδευση του. Συγκεκριμένα, προτείνουν ένα νέο συνδυασμό GRU και ResNet, σημειώνοντας βελτίωση 4.3%-8.8% AUC (περιοχή κάτω από την καμπύλη). Η αρχιτεκτονική όπου ακολουθεί το δίκτυο συμπίεσης είναι η παρακάτω:

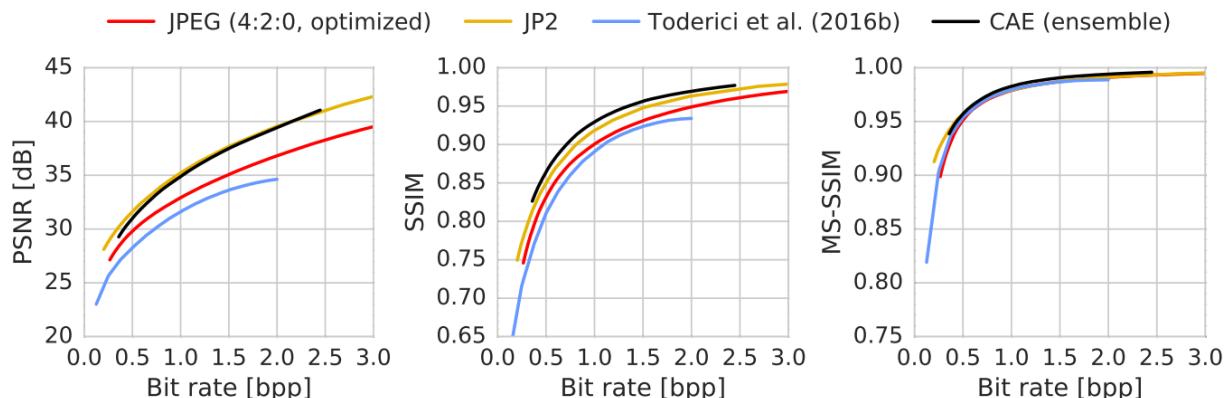
- Δίκτυο κωδικοποίησης E
- Συνάρτηση Διαδικοποίησης (binarization) B
- Δίκτυο αποκωδικοποίησης D

Τα δίκτυα E, D αποτελούνται από RNNs και η ροή του συστήματος είναι η ακόλουθη. Πρώτα, οι εικόνες εισόδου τροφοδοτούνται στο δίκτυο κωδικοποίησης E και με το binarizer μετατρέπονται σε δυαδικό κώδικα. Υστερα, το δίκτυο αποκωδικοποίησης δημιουργεί μια εκτίμηση της αρχικής εικόνας εισόδου, βασιζόμενο στον δυαδικό κώδικα όπου έχει λάβει ως είσοδο. Η παραπάνω διαδικασία επαναλαμβάνεται με συνάρτηση σφάλματος την διαφορά μεταξύ της αρχικής και της ανακατασκευασμένης εικόνας. Τα αποτελέσματα όπου προκύπτουν από το παραπάνω μοντέλο όταν η εκπαίδευση γίνεται σε εικόνες μεγέθους 32x32 είναι MS-SSIM AUC=1.8098 και PSNR-HVS AUC=53.15.

Οι Lucas Theis et al.[28] στην εργασία τους "Lossy image compression with compressive autoencoders" έκαναν χρήση των "compressive autoencoders" για την επίτευξη συμπίεσης. Για την αντιμετώπιση του προβλήματος της διαφοροποίησης της συνάρτησης κβαντισμού, προτείνουν την αντικατάσταση της παραγώγου της συνάρτησης μόνο κατά την οπισθοδόμηση με:

$$\frac{d}{dy}[y] := \frac{d}{dy}r(y)$$

Εμπειρικά, διαπίστωσαν ότι η ταυτότητα  $r(y) = y$  λειτουργεί καλά και σε πιο περίπλοκες επιλογές εικόνων. Τα αποτελέσματα της παραπάνω αρχιτεκτονικής είναι:



Σχήμα 3.1: Αποτελέσματα με χρήση CAE.[28]

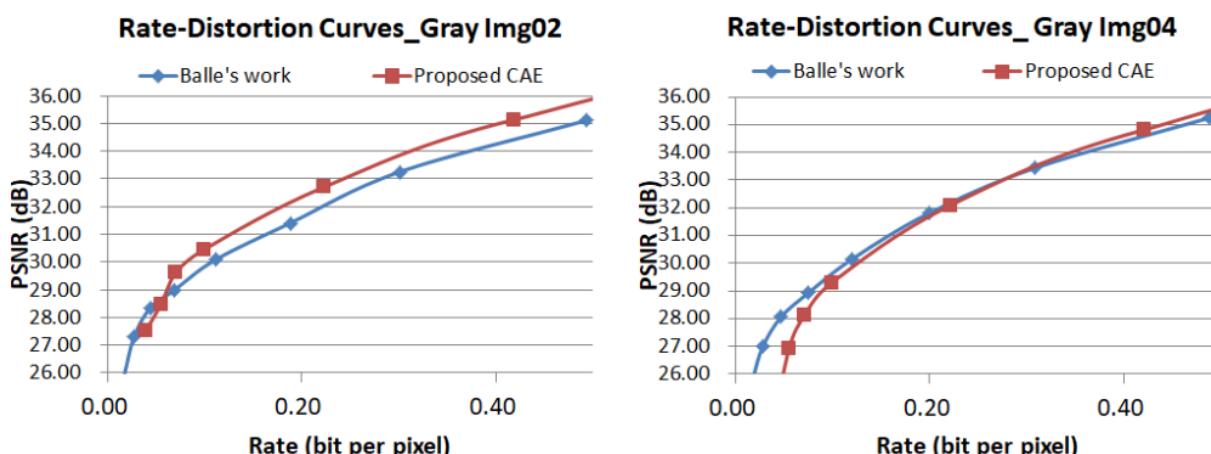
Οι Zhengxue Cheng et al. στην εργασία τους "Deep Convolutional AutoEncoder-based Lossy Image Compression"[29] παρουσιάζουν μια αρχιτεκτονική συμπίεσης εικόνας με απώλειες η οποία εκμεταλλεύεται τα πλεονεκτήματα του συνελικτικού αυτοκωδικοποιητή (Autoencoder) για την επίτευξη υψηλής απόδοσης κωδικοποίησης. Αρχικά, για την αντικατάσταση των κλασσικών μεθόδων κωδικοποίησης με μετασχηματισμό (πχ. Διακριτός μετασχηματισμός συνημιτόνου), κατασκευάζεται ένας συνελικτικός autoencoder (CAE), ο οποίος εκπαιδεύεται με την χρήση της συνάρτησης σφάλματος ρυθμού-παραμόρφωσης (rate-distortion loss function). Στην συνέχεια, εφαρμόζεται ανάλυση κύριων συνιστωσών (PCA) και τέλος ο κβαντισμός και η κωδικοποίηση εντροπίας για την παραγωγή του κώδικα. Το μοντέλο μπορεί να περιγραφεί ως εξής:

- Κωδικοποίηση: Το στάδιο της κωδικοποίησης περιλαμβάνει την προ-επεξεργασία (Μετατροπή στον χρωματικό χώρο YCbCr και κανονικοποίηση), τον CAE, το PCA, την κβάντιση και την κωδικοποίηση εντροπίας.
- Αποκωδικοποίηση: Το στάδιο της αποκωδικοποίησης είναι ο καθρέπτης της αρχιτεκτονικής της κωδικοποίησης.

Στους κωδικοποιητές, η συνηθέστερη συνάρτηση όπου χρησιμοποιείται για τον κβαντισμό είναι η συνάρτηση στρογγυλοποίησης. Ωστόσο, όπως είναι γνωστό η συνάρτηση αυτή δεν είναι διαφοροποιήσιμη και συνεπώς δεν μπορεί να ενσωματωθεί άμεσα στη διαδικασία βελτιστοποίησης των συνελικτικών αυτοκωδικοποιητών (CAE). Για την αντιμετώπιση του παραπάνω προβλήματος, έχουν προταθεί διάφορες προσεγγίσεις.<sup>[29]</sup> Συγκεκριμένα:

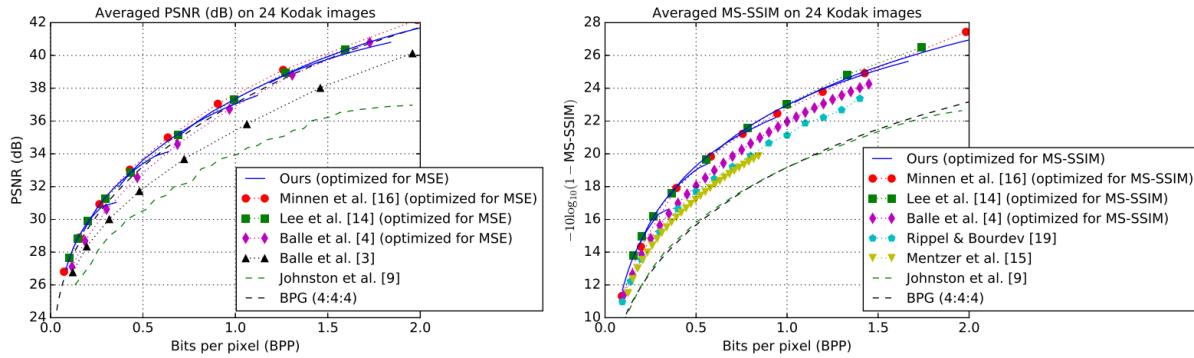
- Οι Theis et al. πρότειναν την αντικατάσταση της παραγώγου κατά το backpropagation με τιμές σχεδόν ίσες με 1.
- Οι Balle et al. αντικατέστησαν την κβάντιση με πρόσθιτο ομοιόμορφο θόρυβο.
- Οι Toderici et al. χρησιμοποίησαν μια στοχαστική συνάρτηση δυαδικοποίησης όπου λαμβάνει την τιμή -1 όταν η είσοδος του έχει τιμή μικρότερη του 0 και την τιμή 1 σε διαφορετικές περιπτώσεις.

Στην συγκεκριμένη εργασία, χρησιμοποιείται απλός ομοιόμορφος θόρυβος για την προσομοίωση του θορύβου του κβαντιστή κατά την εκπαίδευση του CAE. Τα αποτελέσματα του παραπάνω μοντέλου αναφαίνονται στις εικόνες παρακάτω:



Σχήμα 3.2: Αποτελέσματα των Zhengxue Cheng et al.<sup>[29]</sup>

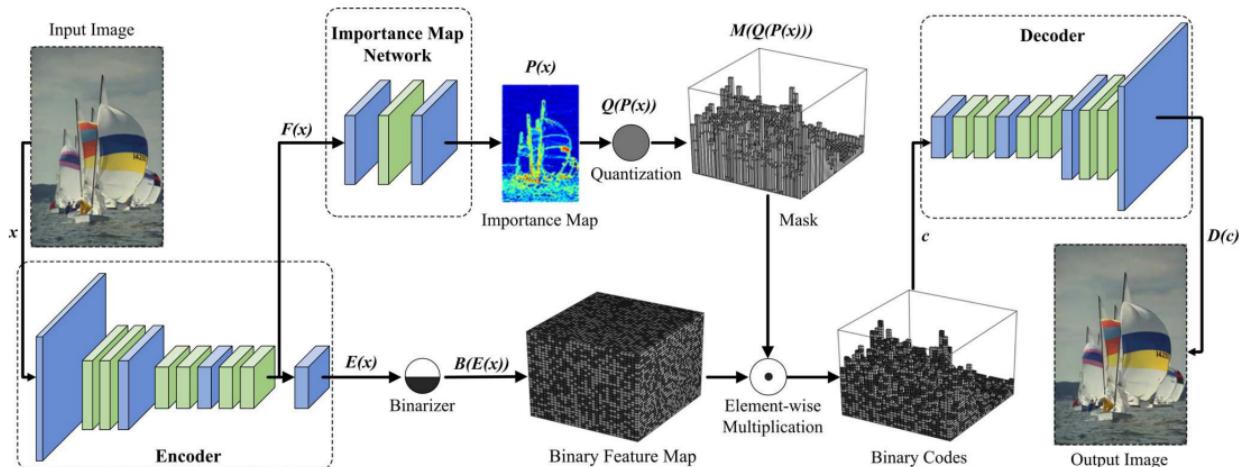
Οι Yoojin Choi et al στην εργασία τους "Variable Rate Deep Image Compression With a Conditional Autoencoder"<sup>[30]</sup> ανέπτυξαν ένα δίκτυο συμπίεσης εικόνας με μεταβλητό ρυθμό bit, ο οποίος υλοποιείται με έναν conditional autoencoder. Παρέχονται δύο παράμετροι για τον έλεγχο του ρυθμού, ο πολλαπλασιαστής Lagrange και το μέγεθος του κβαντισμού. Οι παράμετροι αυτοί δίνονται ως μεταβλητές στο δίκτυο. Η προσαρμογή του ρυθμού σε μια συγκεκριμένη τιμή πραγματοποιείται μεταβάλλοντας τον πολλαπλασιαστή Lagrange, ενώ μπορεί να ρυθμιστεί περαιτέρω με τη προσαρμογή του μεγέθους κβαντισμού της κωδικοποιημένης μορφής. Τα αποτελέσματα όπου φέρει το παραπάνω μοντέλο είναι:



Σχήμα 3.3: Αποτελέσματα των Yoojin Choi et al.[30]

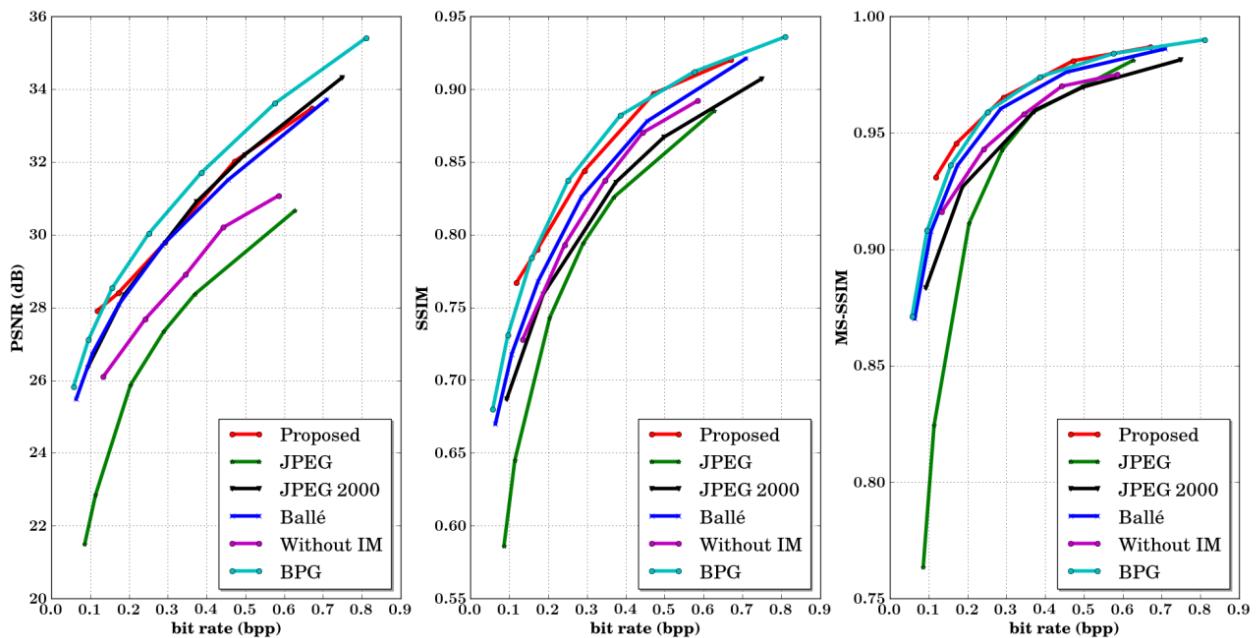
Οι Mu Li et al. στην εργασία τους "Learning Convolutional Networks for Content-weighted Image Compression"[31], σπουδάζουν την δυσκολία στην κατασκευή ενός CNN δικτύου με σκοπό την συμπίεση, λόγω της αδυναμίας διαφοροποίησης της συνάρτησης κβαντισμού. Ωστόσο, βασιζόμενοι στο γεγονός ότι η τοπική πληροφορία είναι μια χωρικά εξαρτώμενη μεταβλητή, προτείνουν ότι το bit rate μιας περιοχής της εικόνας πρέπει να προσαρμόζεται ανάλογα με το τοπικό περιεχόμενο εκείνης της περιοχής. Το bit rate όπου θα αποδίδεται εκεί, θα τοποθετείται με την βοήθεια ενός πίνακα όπου περιέχει τα βάρον τοπικής σημασίας. Κατά αυτό τον τρόπο, το άθροισμα του παραπάνω πίνακα θα αποτελεί την εναλλακτική λύση ως εκτιμητή εντροπίας.

Για το πρόβλημα της διαδικοποίησης, οι Mu Li et al. χρησιμοποιούν τον binarizer για τον κβαντισμό της εξόδου του κωδικοποιητή και εισάγουν μια συνάρτηση προσέγγισης της διαδικοποίησης έτσι ώστε να είναι διαφοροποιημένη κατά την οπισθοδρόμηση. Η αρχιτεκτονική του μοντέλου τους είναι η παρακάτω:



Σχήμα 3.4: Μοντέλο των Mu Li et al.[31]

Αντίστοιχα, τα αποτελέσματα όπου προκύπτουν από την εφαρμογή αυτού του μοντέλου είναι:



Σχήμα 3.5: Αποτελέσματα των Mu Li et al.[31]



## 4. Επιλεχθείσα Μέθοδος

### 4.0.1 Βάση Δεδομένων

Ως βάση δεδομένων χρησιμοποιήθηκε το RAISE\_127[58]. Η συγκεκριμένη βάση αποτελείται από 8156 RAW φωτογραφίες οι οποίες χωρίζονται στις παρακάτω υποκατηγορίες:

- Φωτογραφίες εξωτερικού χώρου.
- Φωτογραφίες εσωτερικού χώρου.
- Φωτογραφίες τοπίου.
- Φωτογραφίες φύσης.
- Φωτογραφίες ανθρώπων.
- Φωτογραφίες αντικειμένων.
- Φωτογραφίες κτηρίων.

Η επιλογή της συγκεκριμένης βάσης δεδομένων έγινε λόγω των RAW φωτογραφιών, καθώς σε περιπτώσεις συμπίεσης εικόνας είναι επιθυμητό οι αρχικές εικόνες να μην έχουν επεξεργαστεί και να αποτελούνται από 8 bpp σε περιπτώσεις ασπρόμαυρων (grayscale), δηλαδή ύπαρξη  $2^8 = 256$  εντάσεων του γκρι, και 24 bpp ( $8\text{bpp} * 3\text{color channels}$ ) σε περιπτώσεις έγχρωμων (RGB). Το RAISE\_127 αποτελεί ένα excel αρχείο το οποίο εμπεριέχει τους συνδέσμους για την αποθήκευση κάθε μιας από τις φωτογραφίες, από το οποίο επιλέχθηκαν 500 τυχαίες για την χρήση τους στην υλοποίηση της εργασίας.

### 4.0.2 Επιλεχθείσα Μέθοδος

Η συγκεκριμένη εργασία στοχεύει στην ανάπτυξη ενός αλγορίθμου βαθιάς μάθησης για την συμπίεση εικόνων με χρήση των συνελικτικών νευρωνικών δικτύων (CNNs). Αν και πλέον τα συνηθέστερα νευρωνικά δίκτυα όπου εφαρμόζονται στον συγκεκριμένο τομέα είναι οι συνελικτικοί αυτοκωδικοποιητές (convolutional autoencoders) και τα παραγωγικά αντιπαραθετικά δίκτυα (GANs), έχει ενδιαφέρον η μελέτη και η κατασκευή μοντέλων τα οποία θα τα συναγωνίζονται. Σε αυτό το κεφάλαιο, θα παρουσιαστεί η επιλεχθείσα μέθοδος και θα αναλυθούν τα επιμέρους στοιχεία που την απαρτίζουν.

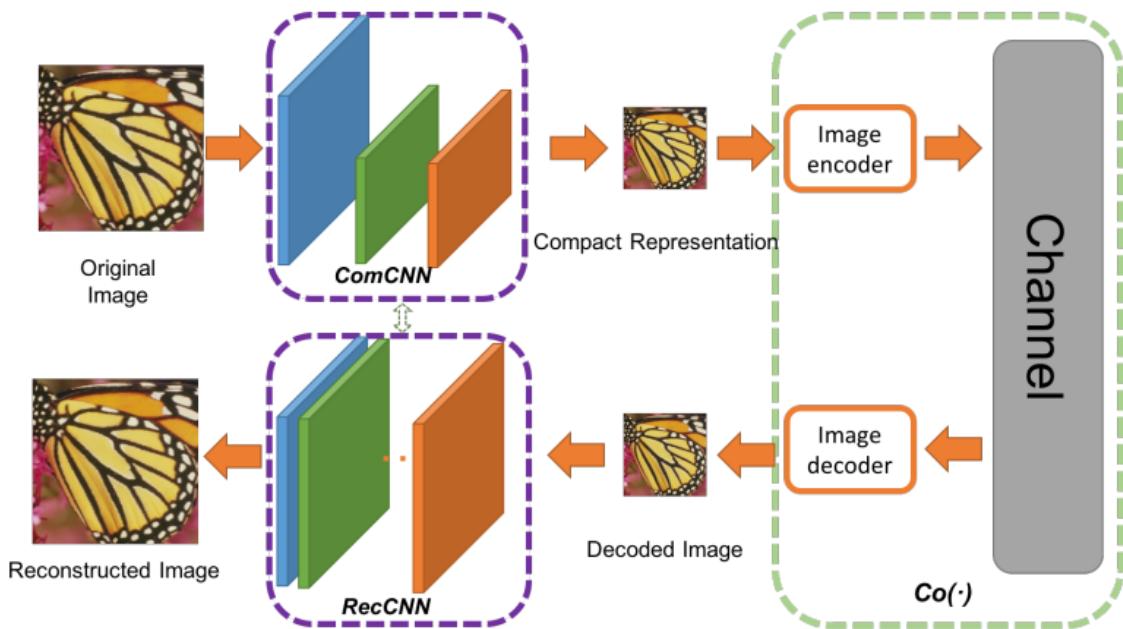
#### 4.0.2.1 Προεπεξεργασία Δεδομένων

Ας ξεκινήσουμε από την προεπεξεργασία των δεδομένων. Για την προετοιμασία των δεδομένων για την χρήση τους από το μοντέλο, ακολουθείται μια συνήθης διαδικασία όπου συναντάται συχνά στα προβλήματα βαθιάς μάθησης. Συγκεκριμένα, όπως προαναφέραμε, αρχικά κατασκευάζεται ένα dataset το οποίο αποτελείται από 500 RGB RAW φωτογραφίες.

Στην συνέχεια αυτές με την βοήθεια των βιβλιοθηκών της tensorflow, μετατρέπονται σε εικόνες grayscale και συνεπώς έχουν τιμή bpp ίση με 8. Στην συνέχεια, οι διαστάσεις των δεδομένων τίθενται σε (180x180) και κανονικοποιούνται.

#### 4.0.2.2 Αρχιτεκτονική του μοντέλου

Το μοντέλο που θα υλοποιηθεί σε αυτή την εργασία, αποτελείται από δύο συνελικτικά δίκτυα και έναν κωδικοποιητή. Πιο συγκεκριμένα, το πρώτο δίκτυο ονομάζεται συμπαγές συνελικτικό δίκτυο (compact convolutional neural network - ComCNN) και μαθαίνει την βέλτιστη συμπαγή αναπαράσταση από μια εικόνα εισόδου, έτσι ώστε να διατηρεί την δομική της πληροφορία. Στην συνέχεια, η έξοδος του ComCNN, κωδικοποιείται από έναν κωδικοποιητή της επιλογής μας (JPEG στην δική μας περίπτωση) και εισέρχεται στο δεύτερο νευρωνικό δίκτυο του συστήματος, όπου ονομάζεται ανακατασκευαστικό συνελικτικό δίκτυο (reconstruction convolutional neural network - RecCNN) το οποίο χρησιμοποιείται για την ανακατασκευή της κωδικοποιημένης εικόνας. Για να επιτύχουμε αποτελεσματική συνεργασία ανάμεσα στα δύο νευρωνικά δίκτυα, αναπτύσσεται ένας end to end αλγόριθμος εκμάθησης με βάση τον οποίο το ComCNN και το RecCNN μαθαίνουν ταυτόχρονα. Ο παραπάνω αλγόριθμος θα αναλυθεί περαιτέρω στο κεφάλαιο της εκπαίδευσης.



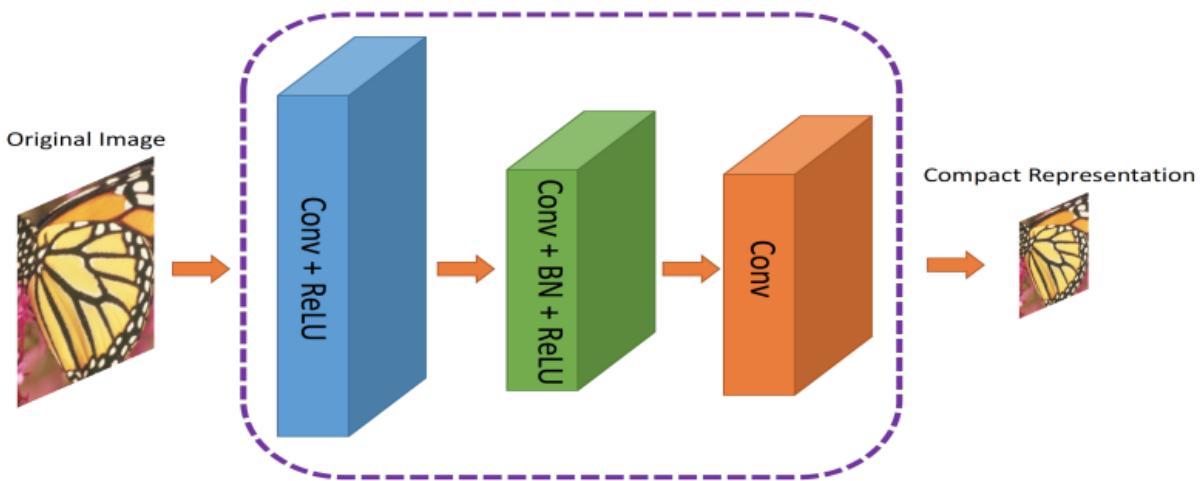
Σχήμα 4.1: Υλοποίηση του μοντέλου[32].

Όπως έχει αναφερθεί και στο προηγούμενο κεφάλαιο, όταν επιθυμούμε να χρησιμοποιήσουμε νευρωνικά δίκτυα σε εργασίες συμπίεσης εικόνας, υπάρχει το πρόβλημα του κβαντισμού. Πιο συγκεκριμένα, στο στάδιο της συμπίεσης με JPEG, γνωρίζουμε ότι κατά τον κβαντισμό, η συνάρτηση στρογγυλοποίησης (rounding function) δεν έχει παράγωγο και κατά συνέπεια συναντάται πρόβλημα κατά την οπισθοδρόμηση. Για την επίλυση αυτού το προβλήματος, κατασκευάζεται ένας σταδιακός αλγόριθμος μάθησης, με βάση τον οποίο, θέτοντας σταθερές τις παραμέτρους του ενός δικτύου, παγώνοντας δηλαδή τα βάρον του, εκπαιδεύεται το δεύτερο.

#### 4.0.2.3 Αρχιτεκτονική ComCNN

Το συμπαγές συνελικτικό νευρωνικό δίκτυο αποτελείται από τρία στρώματα τα οποία διαπλούν την δομική πληροφορία όπου υπάρχει στην εικόνα. Το πρώτο στρώμα χρησιμοποιείται για την εξαγωγή των επικαλυπτόμενων patches από την εικόνα εισόδου. Αποτελείται από ένα συνελικτικό επίπεδο (convolutional layer) το οποίο ακολουθεί η συνάρτηση ενεργοποίησης ReLU. 64 φίλτρα διαστάσεων 3x3x $c$ , όπου  $c$  ο αριθμός των καναλιών της εικόνας (1 για ασπρόμαυρες εικόνες, 3 για έγχρωμες) χρησιμοποιούνται για την δημιουργία 64 χαρτών χαρακτηριστικών (feature maps). Το δεύτερο κύριο επίπεδο του μοντέλου έχει σκοπό την περαιτέρω μείωση των διαστάσεων της

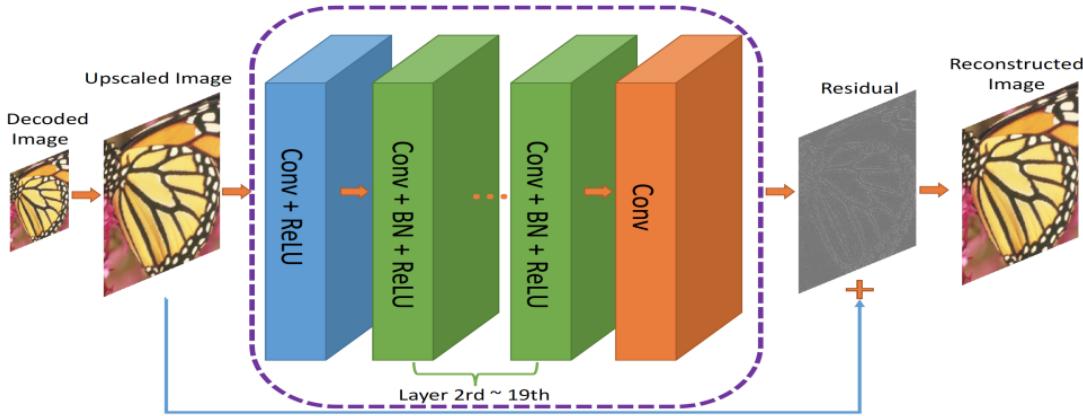
εικόνας και την ενίσχυση των χαμηλών χαρακτηριστικών. Για αυτό τον λόγο, χρησιμοποιείται ένα συνελικτικό επίπεδο με stride=2 όπου δημιουργεί 64 φίλτρα διαστάσεων 3x3x64. Όπως και προηγουμένως, ακολουθεί η συνάρτηση ενεργοποίησης ReLU. Στο τελευταίο στρώμα, γίνεται χρήση c φίλτρων διαστάσεων 3x3x64 με σκοπό την κατασκευή της συμπαγούς αναπαράστασης της εικόνας.



Σχήμα 4.2: Αρχιτεκτονική του ComCNN[32].

#### 4.0.2.4 Αρχιτεκτονική RecCNN

Το ανακατασκευαστικό συνελικτικό δίκτυο αναλαμβάνει την εργασία της ανακατασκευής της εικόνας. Αποτελείται από 20 στρώματα τα οποία ακολουθούν τρεις τύπους συνδυασμών: Συνέλιξη + ReLU, Συνέλιξη + Κανονικοποίηση παρτίδας (Batch Normalization) + ReLU, Συνέλιξη. Για το πρώτο στρώμα, 64 φίλτρα διαστάσεων 3x3x64 χρησιμοποιούνται για την δημιουργία 64 χαρτών χαρακτηριστικών και έπειτα εφαρμόζεται η ReLU. Το δεύτερο έως και το 19ο επίπεδο αποτελούνται από ένα συνελικτικό επίπεδο 64 φίλτρων διαστάσεων 3x3x64, ακολουθούνται από επίπεδο batch normalization και τέλος έπειτα η συνάρτηση ReLU. Στο τελευταίο στρώμα, c φίλτρα διαστάσεων 3x3x64 χρησιμοποιούνται για την ανακατασκευή της εξόδου. Όπως παρατηρείται και στην εικόνα παρακάτω, στο συγκεκριμένο δίκτυο υλοποιείται υπολειμματική εκπαίδευση. Συγκεκριμένα, η είσοδος του δικτύου RecCNN αποτελεί μια συμπιεσμένη εικόνα η οποία έχει υποστεί αυξητική δειγματοληψία με την χρήση του bicubic interpolation, ενώ αποτελείται από τα χαρακτηριστικά χαμηλής συχνότητας της αρχικής εικόνας. Το δίκτυο RecCNN προσπαθεί να κατασκευάσει μια υπολειμματική εικόνα, στην οποία όταν προστεθεί η είσοδος του, η προκύπτουσα εικόνα θα προσεγγίζει την αρχική.



Σχήμα 4.3: Αρχιτεκτονική του RecCNN[32].

#### 4.0.2.5 JPEG

Στην συγκεκριμένη υλοποίηση, συμπεριλαμβάνεται και ο κλασσικός τρόπος συμπίεσης εικόνας με JPEG. Συγκεκριμένα, τον εφαρμόζουμε στην έξοδο του ComCNN για την περαιτέρω μείωση των διαστάσεων της συμπαγούς εικόνας αλλά κυρίως, μέσω της εναλλαγής του quality factor του κωδικοποιητή, επιτυγχάνεται η δημιουργία εικόνων με διαφορετικά bits per pixel.

#### 4.0.2.6 Εκπαίδευση

Όπως προαναφέρθηκε, λόγω της συνάρτησης στρογγυλοποίησης του κωδικοποιητή JPEG, όπου δεν παραγωντοποιείται, για την εκπαίδευση του μοντέλου κατασκευάζεται ένας end to end αλγόριθμος εκμάθησης. Προτού αναλυθεί ο αλγόριθμος, δίνονται οι συναρτήσεις σφάλματος των δύο δικτύων:

- Για την εκπαίδευση του ComCNN δικτύου:

$$L_1(\theta_1) = \frac{1}{2N} \sum_{k=1}^N \|Re(\hat{\theta}_2, Cr(\theta_1, x_k)) - x_k\|^2$$

όπου  $Re(\cdot)$  είναι το RecCNN,  $\hat{\theta}_2$  οι παράμετροι του RecCNN προς εκπαίδευση,  $Cr(\cdot)$  το ComCNN δίκτυο και  $x_k$  η αρχική εικόνα. Παρατηρούμε ότι η συνάρτηση σφάλματος είναι η MSE διαφορά μεταξύ της αρχικής και της ανακατασκευασμένης εικόνας. Επιπλέον, όταν εκπαιδεύεται το ComCNN, αφού η συνάρτηση σφάλματος του περιλαμβάνει και το δίκτυο RecCNN, η εικόνα δεν περνάει από τον κωδικοποιητή JPEG για να μην υπάρχει πρόβλημα κατά την οπισθοδόμηση.

- Για την εκπαίδευση του RecCNN δικτύου:

$$L_2(\theta_2) = \frac{1}{2N} \sum_{k=1}^N \|res(Co(\hat{x}_{mk}, \theta_2) - (Co(\hat{x}_{mk}) - x_k)\|^2$$

όπου  $\theta_2$  είναι οι παράμετροι του RecCNN όπου εκπαιδεύονται,  $res(\cdot)$  το υπόλειμμα όπου έχει μάθει το RecCNN,  $Co(\cdot)$  ο κωδικοποιητής JPEG και  $\hat{x}_{mk}$  η εικόνα αφού περάσει από το δίκτυο ComppCNN και τον κωδικοποιητή. Η έκφραση  $(Co(\hat{x}_{mk}) - x_k)$  είναι ουσιαστικά η αφαίρεση των "χαμηλών" χαρακτηριστικών ( $Co(\hat{x}_{mk})$ ) από την αρχική εικόνα. Συνεπώς, αναπαριστά τις λεπτομέρειες της αρχικής εικόνας. Έτσι, το μοντέλο RecCNN καθώς προσπαθεί να ελαχιστοποιήσει την διαφορά μεταξύ της residual εξόδου του και της παραπάνω διαφοράς, μαθαίνει την παραγωγή residuals τα οποία περιέχουν τα "υψηλά" χαρακτηριστικά

---

της αρχικής εικόνας. Παρατηρώντας την εικόνα αρχιτεκτονικής του μοντέλου RecCNN, στην υπολειμματική εικόνα προσθέτουμε την upscaled για την παραγωγή της ανακατασκευασμένης. Από την παραπάνω ανάλυση, γίνεται εμφανές ότι η εκπαίδευση του RecCNN γίνεται έτσι ώστε να εκλείπουν από την residual εικόνα οι πληροφορίες όπου της προσθέτουμε έπειτα.

Ο αλγόριθμος εκπαίδευσης είναι ο παρακάτω:

**Βήμα 1ο:**

Εισαγωγή της αρχικής εικόνας στο δίκτυο ComCNN.

**Βήμα 2ο:**

Κωδικοποίηση της εξόδου του δίκτυο με το πρότυπο JPEG

**Βήμα 3ο:**

Ενημέρωση των  $\hat{\theta}_2$  παραμέτρων του RecCNN με βάση την συνάρτηση απώλειας του.

**Βήμα 4ο:**

Για  $\theta_2 = \hat{\theta}_2$  ενημέρωση των  $\hat{\theta}_1$  παραμέτρων του ComCNN με βάση την συνάρτηση απώλειας του.



## 5. Πειραματική Διαδικασία

### 5.0.1 Πειραματική Διαδικασία

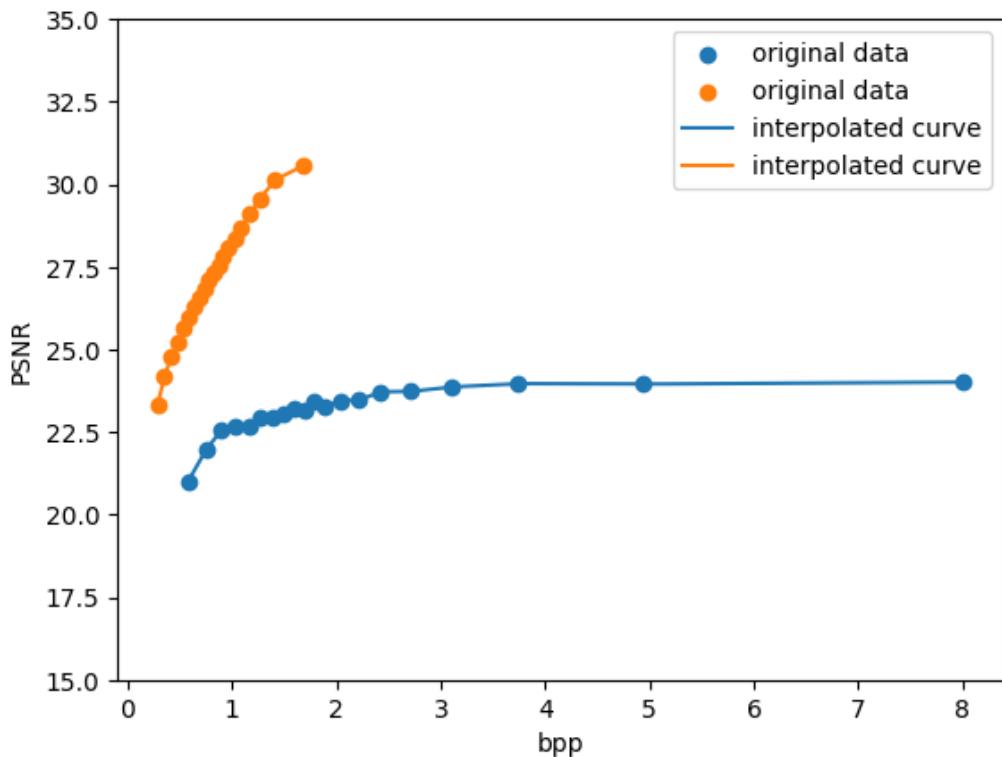
Η μέθοδος όπου επιλέχθηκε για την συμπίεση εικόνας με συνελικτικά δίκτυα αποτελεί υλοποίηση της εργασίας των Feng Jiang et al[32]. Όπως προαναφέρθηκε στο προηγούμενο κεφάλαιο, για την εκπαίδευση του δικτύου χρησιμοποιήθηκαν 500 RAW RGB φωτογραφίες, οι οποίες κατά το στάδιο της προεπεξεργασίας έχουν μετατραπεί σε grayscale, οι διαστάσεις τους έχουν οριστεί σε 180x180 και έχουν κανονικοποιηθεί σε εύρος τιμών [0,1]. Το 0.1 του παραπάνω dataset δεσμεύεται ως validation set, ενώ το υπόλοιπο χρησιμοποιείται στην εκπαίδευση. Τέλος, ως test set χρησιμοποιήθηκε το Berkeley Segmentation Dataset (BSD300) όπου αποτελείται από 100 JPEG εικόνες.

Στην διεξαγωγή του πειράματος, δεν χρησιμοποιήθηκαν data augmentations και patch extractions. Στην είσοδο του μοντέλου εισέρχεται λοιπόν ολόκληρη η εικόνα (180x180), την οποία το μοντέλο προσπαθεί να ανακατασκευάσει. Το μέγεθος παρτίδας ορίστηκε 8, ο αριθμός των εποχών ως 20, αν και η σύγκλιση παρατηρείται νωρίτερα ενώ ως βελτιστοποιητής επιλέχθηκε ο Adam.

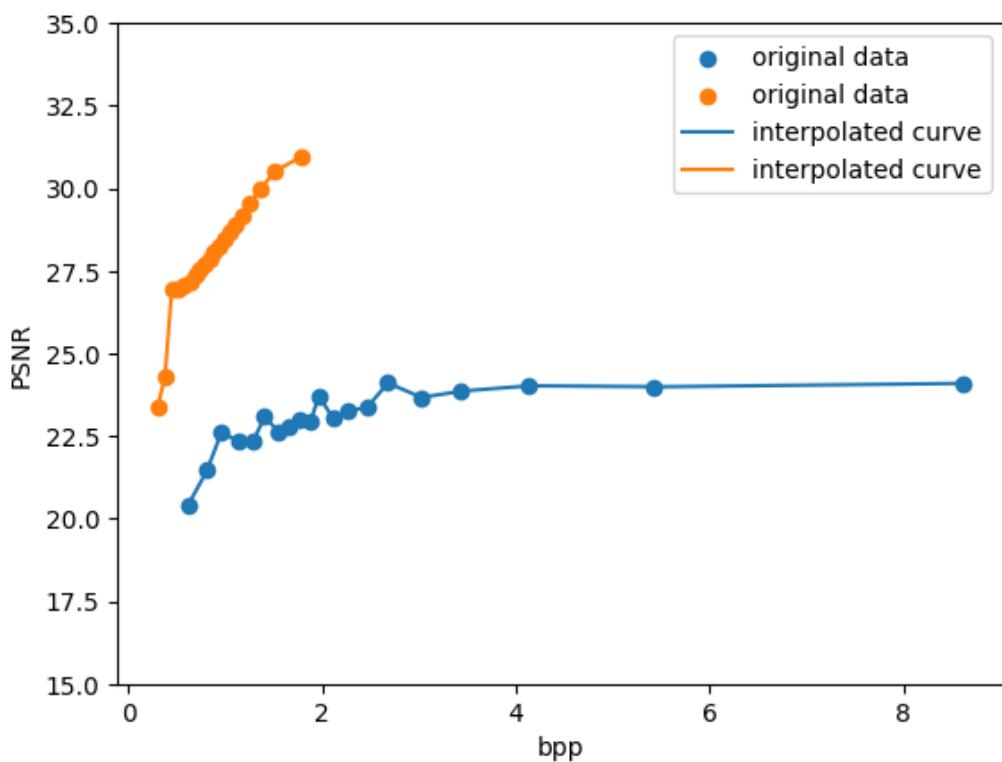
Όπως αναφέρθηκε προηγουμένως, το μοντέλο αποτελείται από δύο συνελικτικά δίκτυα. Για κάθε ένα από αυτά επιλέγεται και διαφορετικό learning rate καθώς οι αρχιτεκτονικές τους διαφέρουν σημαντικά μεταξύ τους. Για το ComCNN (συμπαγές συνελικτικό δίκτυο), θέτουμε learning rate στην τιμή 0.001 την οποία μειώνουμε εκθετικά έως ότου φτάσει την τιμή 0.0001. Αντίστοιχα, για το RecCNN η τιμή του ρυθμού εκμάθησης τίθεται στο 0.01, η οποία και αυτή μειώνεται εκθετικά έως την τιμή 0.0001. Καθώς η αρχιτεκτονική του RecCNN είναι σημαντικά πιο βαθιά από ότι εκείνη του ComCNN, αποτελεί συνετή επιλογή ο ορισμός του ρυθμού εκμάθησης του σε υψηλότερη τιμή.

Τέλος, οι μετρικές όπου χρησιμοποιήθηκαν για την αξιολόγηση των αποτελεσμάτων είναι το PSNR και το SSIM. Η επιλεχθείσα μέθοδος συγκρίθηκε με την κωδικοποίηση JPEG η οποία επεξηγήθηκε σε προηγούμενο κεφάλαιο. Για την ορθή σύγκριση των αποτελεσμάτων, θέτουμε στον κωδικοποιητή του μοντέλου μας τιμές στο quality factor τέτοιες ώστε οι προκύπτουσες ανακατασκευασμένες εικόνες να έχουν τιμές bpp (bits per pixel) παραπλήσιες με εκείνες όπου συμπιέζονται με το πρότυπο JPEG. Όταν λοιπόν επιτύχουμε κοντινές τιμές σε bpp σε μια εικόνα όπου συμπιέστηκε και με τις δύο μεθόδους, προχωράμε στην σύγκριση της με βάση το PSNR και το SSIM της.

Παρατίθενται οι PSNR γραφικές ρυθμού παραμόρφωσης (rate distortion curves) για το validation set και το BSD300 dataset:

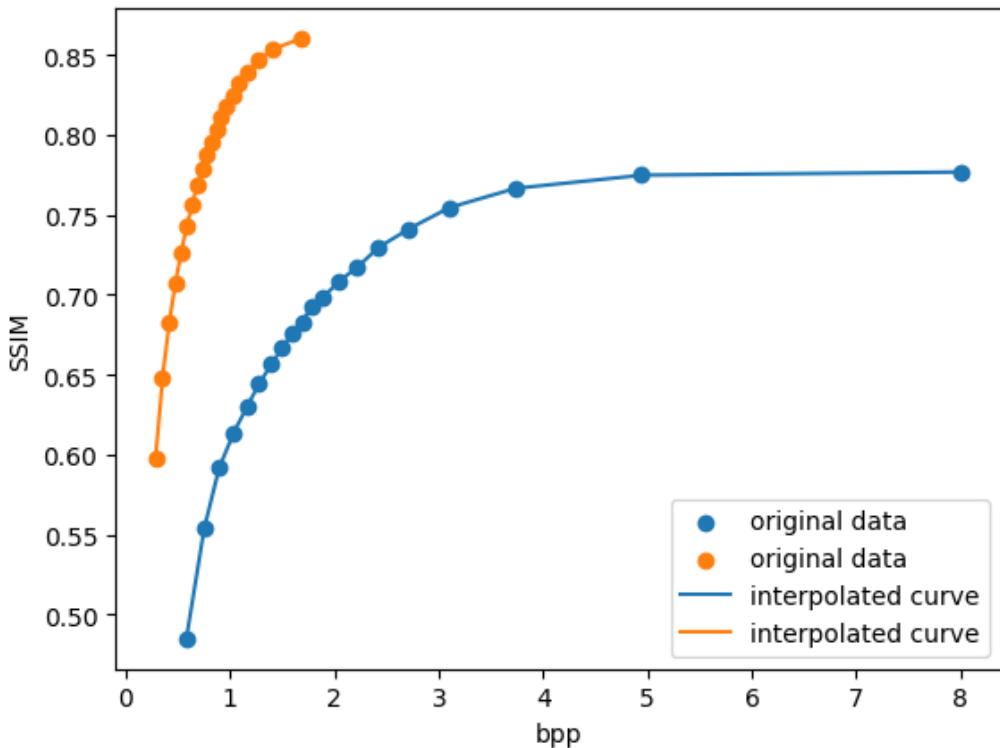


Σχήμα 5.1: PSNR γραφική ρυθμού παραμόρφωσης για το validation test.

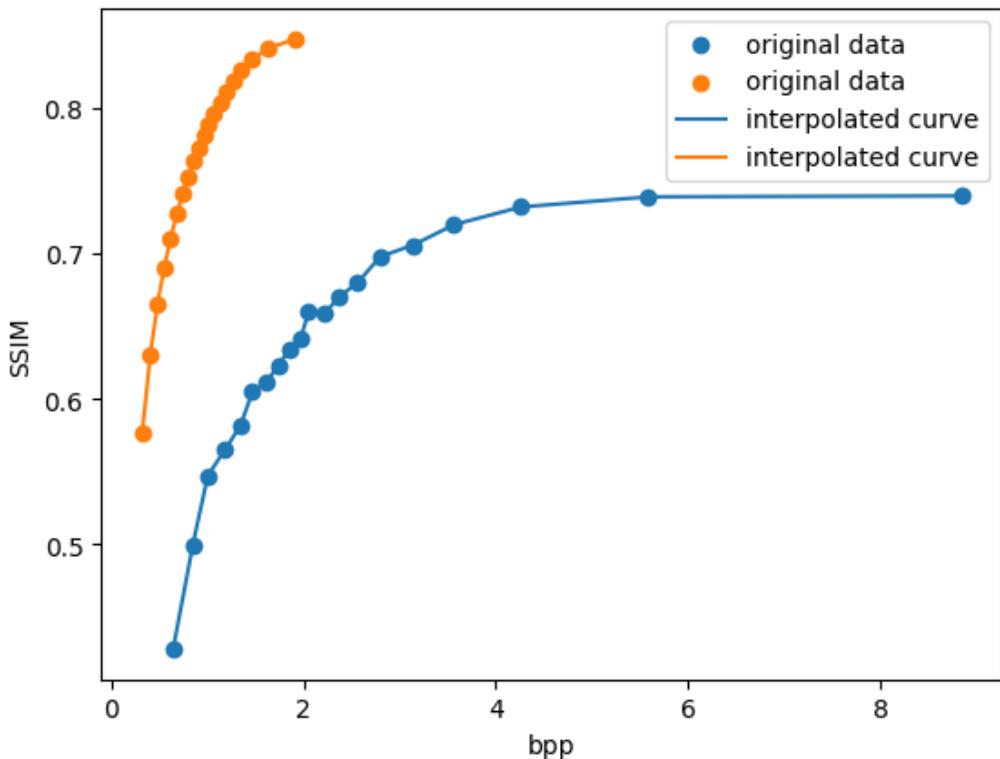


Σχήμα 5.2: PSNR γραφική ρυθμού παραμόρφωσης για το BSD300 dataset.

Αντίστοιχα, παρατίθενται οι SSIM γραφικές ρυθμού παραμόρφωσης για τα δύο datasets:



Σχήμα 5.3: SSIM γραφική ρυθμού παραμόρφωσης για το validation set.

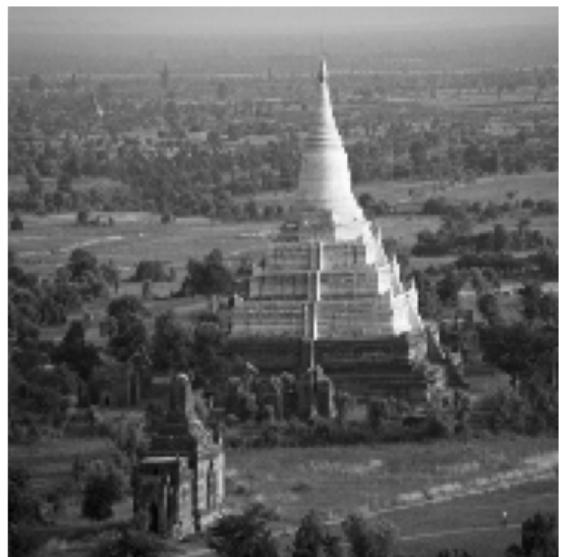
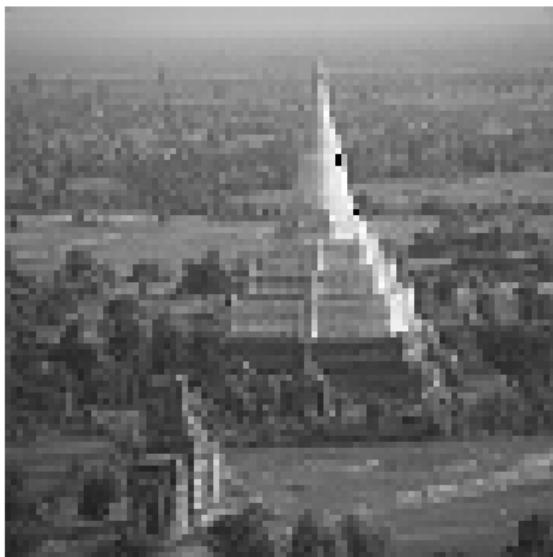


Σχήμα 5.4: SSIM γραφική ρυθμού παραμόρφωσης για το BSD300 dataset.

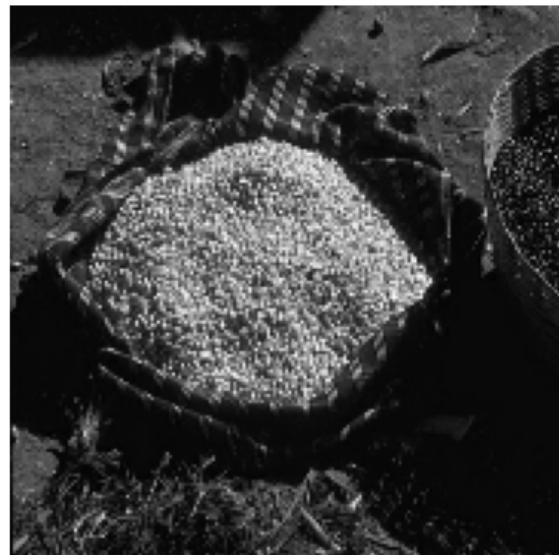
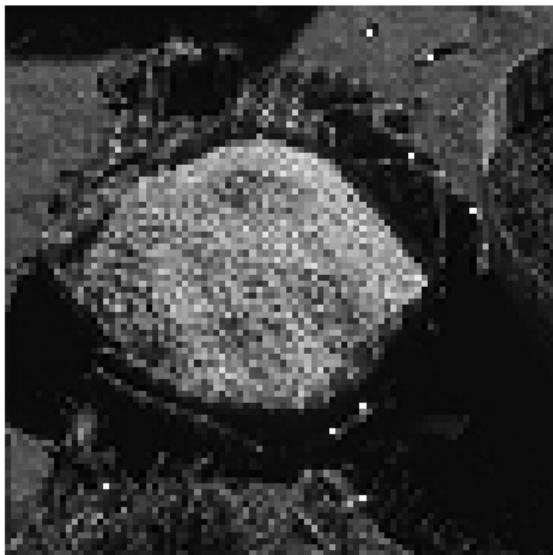
Παρατηρούμε από τις παραπάνω γραφικές ότι το μοντέλο μας δεν καταφέρνει να ξεπεράσει τον κωδικοποιητή JPEG όσο αναφορά στις μετρικές PSNR και SSIM για οποιαδήποτε τιμή bpp. Προτού αναδείξουμε που οφείλεται το παραπάνω, παραθέτονται εικόνες συμπιεσμένες από το μοντέλο μας και εικόνες συμπιεσμένες με το πρότυπο JPEG με παρόμοιο δείκτη bpp:



Σχήμα 5.5: Εικόνα συμπιεσμένη από το μοντέλο, με  $\text{bpp}=1.079$  (αριστερά) και εικόνα συμπιεσμένη με JPEG, με  $\text{bpp}=1.075$  (δεξιά).



Σχήμα 5.6: Εικόνα συμπιεσμένη από το μοντέλο, με  $\text{bpp}=1.1$  (αριστερά) και εικόνα συμπιεσμένη με JPEG, με  $\text{bpp}=1$  (δεξιά).

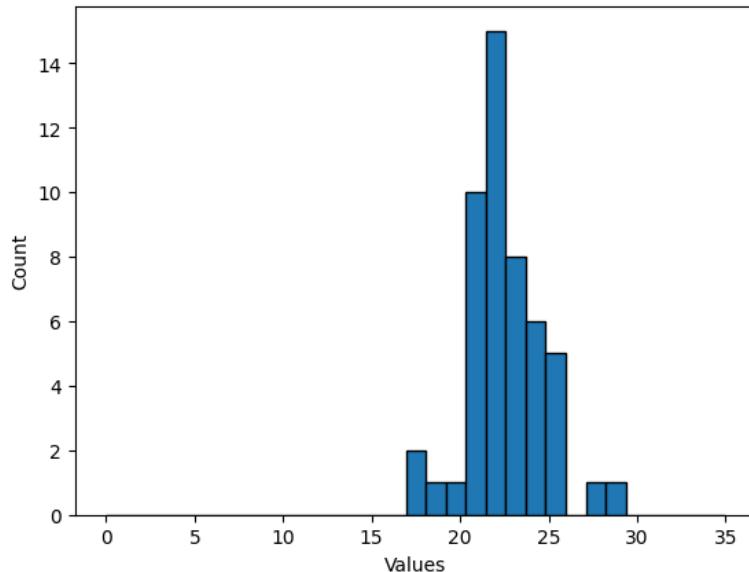


Σχήμα 5.7: Εικόνα συμπιεσμένη από το μοντέλο, με bpp=1.366 (αριστερά) και εικόνα συμπιεσμένη με JPEG, με bpp=1.424 (δεξιά).

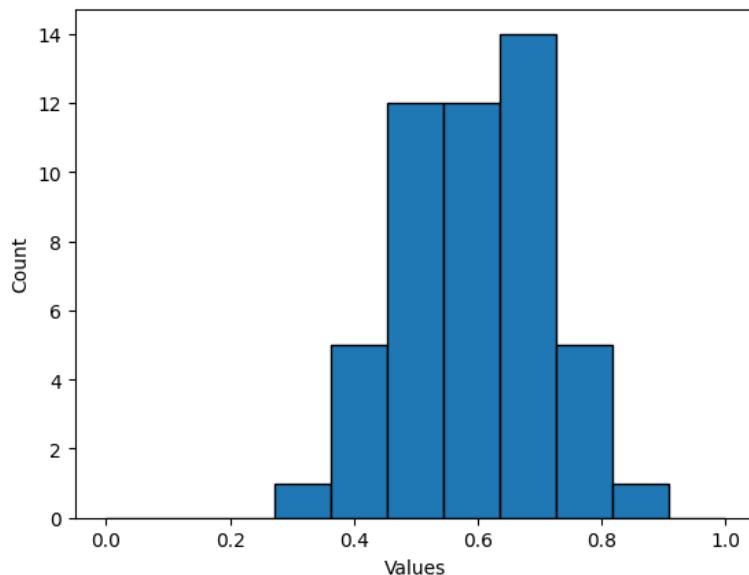


Σχήμα 5.8: Εικόνα συμπιεσμένη από το μοντέλο, με bpp=1.00 (αριστερά) και εικόνα συμπιεσμένη με JPEG, με bpp=1.01 (δεξιά).

Στην συνέχεια, αναφαίνονται ιστογράμματα στα οποία μπορούμε να δούμε την κατανομή των εικόνων με βάση τις τιμές PSNR και SSIM τους.



Σχήμα 5.9: PSNR ιστόγραμμα με QF=15 στο validation set.



Σχήμα 5.10: SSIM ιστόγραμμα με QF=15 στο validation set.

Όπως συμπτεραίνουμε, το μοντέλο μας αποδίδει ανεπαρκώς και δεν καταφέρνει να προσομοιάσει τα αποτελέσματα του JPEG κωδικοποιητή. Ωστόσο, είναι αρκετά ενδιαφέροντα η κατανόηση του λόγου που συμβαίνει αυτό. Για αυτό τον λόγο, θα ακολουθήσουμε την πορεία της εικόνας στο μοντέλο. Όπως εξηγήθηκε στο προηγούμενο κεφάλαιο, η εικόνα εισέρχεται στο συμπαγές συνελικτικό δίκτυο (ComCNN) και στην συνέχεια κωδικοποιείται με βάση το πρότυπο JPEG. Πριν τροφοδοτηθεί στο RecCNN, κατασκευάζεται μια upsampled εικόνα η οποία είναι μια θολή αναπαράσταση της συμπιεσμένης μορφής αλλά με ίδιες διαστάσεις με την αρχική. Υπενθυμίζεται ότι το παραπάνω επιτυγχάνεται με την μέθοδο του bicubic interpolation.

Όπως παραπρούμε, η εικόνα που προκύπτει από την έξοδο του ComCNN και έπειτα από την κωδικοποίηση JPEG, είναι συμπιεσμένη σε μεγάλο βαθμό και εκλείπουν οι λεπτομέρειες της αρχικής εικόνας.



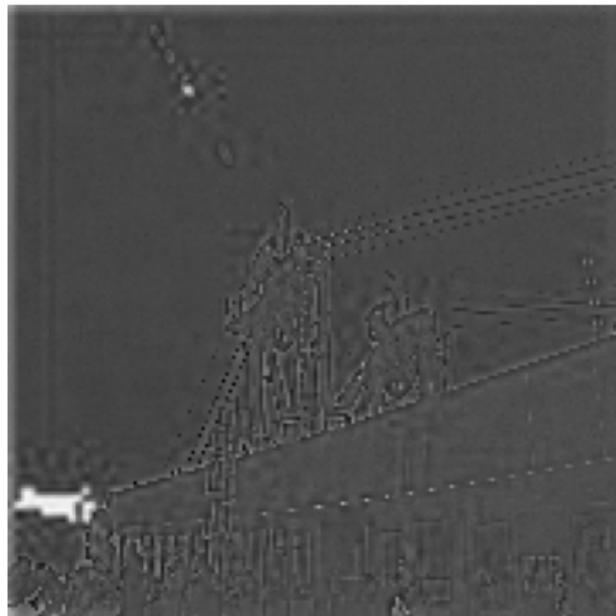
Σχήμα 5.11: Έξοδος ComCNN και συμπίεση JPEG.

Στη συνέχεια κατασκευάζεται η upscaled εικόνα:



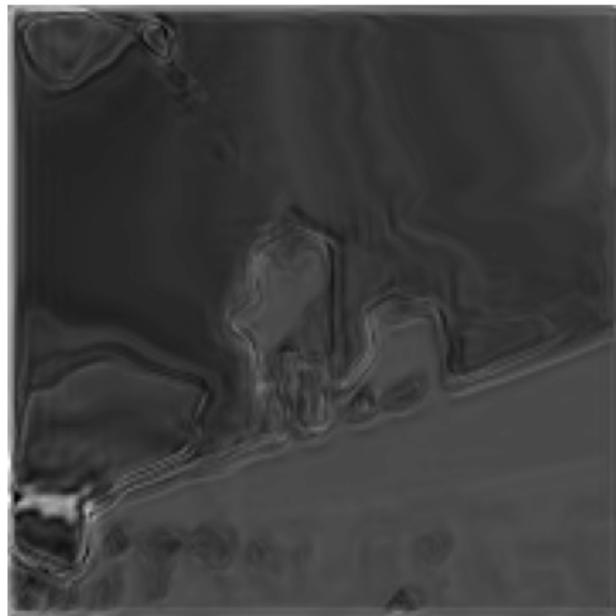
Σχήμα 5.12: Upscaled εικόνα.

Για την κατασκευή της πραγματικής υπολειμματικής εικόνας, αφαιρούμε από την αρχική εικόνα την upscaled. Έτσι δημιουργείται μια εικόνα η οποία εμπεριέχει τις λεπτομέρειες όπου χρειάζονται στην ανακατασκευή της αρχικής εικόνας.



Σχήμα 5.13: Η πραγματική υπολειμματική εικόνα.

Η παραπάνω εικόνα αποτελεί και την εικόνα όπου προσπαθεί να προσεγγίσει το RecCNN δίκτυο. Η υπολειμματική εικόνα όπου δημιουργεί το RecCNN είναι:



Σχήμα 5.14: Υπολειμματική εικόνα.

Όπως παραπούμε, το RecCNN δίκτυο δεν καταφέρνει να προσεγγίσει με επαρκή ακρίβεια της επιθυμητή έξοδο. Συγκεκριμένα, είναι αναποτελεσματικό στην δημιουργία ακμών με ακρίβεια. Ως αποτέλεσμα, η ανακατασκευασμένη εικόνα παραμένει θολή.



Σχήμα 5.15: Ανακατασκευασμένη εικόνα από το δίκτυο RecCNN.

Για καλύτερη κατανόηση της διαδικασίας, προσθέτοντας την πραγματική υπολειμματική εικόνα στην upscaled, λαμβάνουμε το παρακάτω αποτέλεσμα:



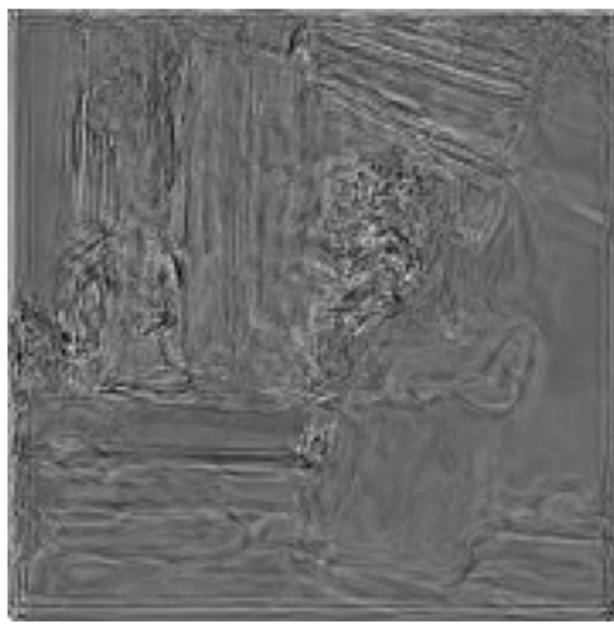
Σχήμα 5.16: Πραγματική υπολειμματική εικόνα + upscaled.

Για την αντιμετώπιση του παραπάνω προβλήματος πραγματοποιήθηκαν αλλαγές στον ρυθμό εκμάθησης των μοντέλων και του βαθμού εκθετικής τους μείωσης ανά εποχή. Επιπλέον, έγινε δοκιμή προσθήκης επιπέδου υποδειγματοληψίας στην αρχιτεκτονική του μοντέλου RecCNN και συγκεκριμένα max pooling. Ωστόσο, κανένα από τα παραπάνω δεν έφερε υψηλότερη απόδοση στο σύστημα.

Τέλος, προστέθηκε ένα επίπεδο στην αρχιτεκτονική του δικτύου για κατάτμηση εικόνας. Έπειτα από την κατασκευή της υπολειμματικής εικόνας από το μοντέλο, η έξοδος εισάγεται στο παραπάνω επίπεδο. Σκοπός είναι η τεχνητή δημιουργία ακμών στην υπολειμματική εικόνα για την πιο ακριβή ανακατασκευή της αρχικής εικόνας.

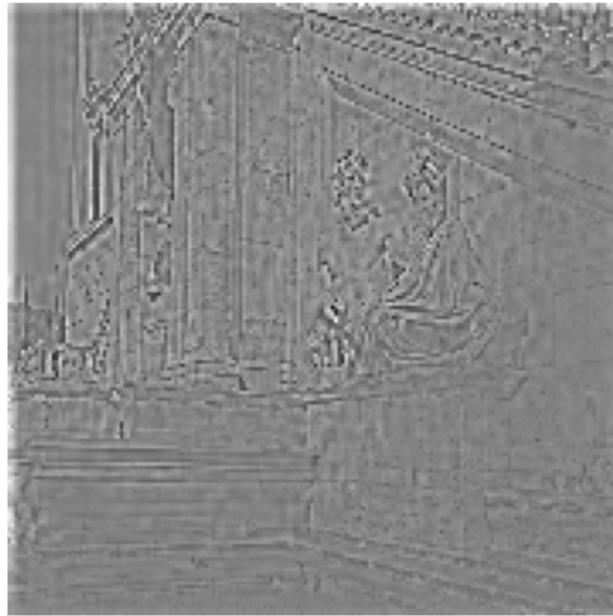


Σχήμα 5.17: Έξοδός έπειτα από JPEG κωδικοποίηση.



Σχήμα 5.18: Κατάτυπη υπολειμματικής εικόνας.

Η κατάτυπη της υπολειμματικής εικόνας φαίνεται οπτικά να περιέχει πιο ισχυρές ακμές.



Σχήμα 5.19: Πραγματική υπολειμματική εικόνα.

Η ανακατασκευασμένη εικόνα όπου προκύπτει είναι η παρακάτω:



Σχήμα 5.20: Τελική ανακατασκευασμένη εικόνα.

### 5.0.2 Συμπεράσματα και μελλοντικοί στόχοι

Σε αυτή την εργασία επιχειρήσαμε την προσέγγιση της διαδικασίας της συμπίεσης εικόνας με τεχνικές βαθιάς μάθησης. Κατασκευάστηκαν δύο συνελικτικά νευρωνικά δίκτυα καθένα από τα οποία ανέλαβε τον ρόλο της συμπίεσης και της ανακατασκευής της εικόνας. Τα δύο δίκτυα εκπαιδεύτηκαν ταυτόχρονα δημιουργώντας κατά αυτόν τον τρόπο ένα end-to-end σύστημα, αντιμετωπίζοντας έτσι παράλληλα το πρόβλημα της οπισθοδρόμησης όπου έγκειται στην συμπίεση εικόνας με τεχνικές βαθιάς μάθησης. Η προσθήκη του κωδικοποιητή JPEG, καθιστά επίσης το σύστημα συμβατό με τα υπάρχοντα πρότυπα κωδικοποίησης εικόνας.

Με βάση τα αποτελέσματα του μοντέλου μπορούμε να συμπεράνουμε ότι αν και δεν επιτεύχθηκε προσέγγιση των αποτελεσμάτων όπου φέρουν καθιερωμένα πρότυπα όπως το JPEG, η βασική δομή του μοντέλου είναι άξια για περισσότερη έρευνα. Συγκεκριμένα, παρατηρείται ότι το ανακατασκευαστικό συνελικτικό δίκτυο δεν επιτυγχάνει ακριβή αναπαράσταση της υπολειμματικής εικόνας, γεγονός όπου οδηγεί το σύστημα σε ανακριβή αποτελέσματα. Η διαδικασία παραγωγής μιας εικόνας υψηλής ανάλυσης από μια εικόνα χαμηλής, πρόκειται για πρόβλημα Single-Image Super Resolution (SISR) το οποίο μπορεί να υλοποιηθεί με χρήση συνελικτικών δικτύων σε συνδυασμό με μεγάλα datasets ζευγών εικόνων.

Μελλοντικοί στόχοι αποτελούν η μελέτη των παραπάνω τεχνικών για την κατασκευή εικόνων υψηλής ανάλυσης αλλά και η αλλαγή της αρχιτεκτονικής του ανακατασκευαστικού συνελικτικού δικτύου του μοντέλου μας. Καθώς ο εντοπισμός των λεπτομερειών μιας εικόνας από την εκδοχή χαμηλής ανάλυσης της αποτελεί ένα δύσκολο και πολύπλοκο πρόβλημα, πιστεύουμε ότι με την χρήση ενός βαθύτερου δικτύου, σε συνδυασμό με ένα μεγαλύτερο δείγμα εικόνων μπορεί να φέρει πιο ικανοποιητικά αποτελέσματα.

# Βιβλιογραφία

- [1] Lithmee. Difference Between Lossy and Lossless Compression. <https://pediaa.com/difference-between-lossy-and-lossless-compression/>, August 2018.
- [2] Wikipedia contributors. Arithmetic coding --- Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Arithmetic\\_coding](https://en.wikipedia.org/w/index.php?title=Arithmetic_coding), 2023. [Online; accessed 3-July-2023].
- [3] Wikipedia contributors. Huffman coding --- Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Huffman\\_coding](https://en.wikipedia.org/w/index.php?title=Huffman_coding), 2023. [Online; accessed 3-July-2023].
- [4] Wikipedia contributors. Lempel-ziv-welch --- Wikipedia, the free encyclopedia. <https://en.wikipedia.org/wiki/Lempel-Ziv-Welch>, 2023. [Online; accessed 3-July-2023].
- [5] Bernd Girod. Stanford university, ee398a image and video compression transform coding no. 1 lecture notes.
- [6] Harikrishna B. Deep Learning. <https://medium.datadriveninvestor.com/deep-learning-2025e8c4a50>, December 2018.
- [7] Machine Learning & Deep Learning. <https://www.datacamp.com/tutorial/machine-deep-learning>.
- [8] Supervised, Unsupervised and Semi-supervised Learning. <https://www.enjoyalgorithms.com/blogs/supervised-unsupervised-and-semisupervised-learning>.
- [9] Artificial Neuron - GM-RKB. [https://www.gabormelli.com/RKB/Artificial\\_Neuron](https://www.gabormelli.com/RKB/Artificial_Neuron).
- [10] Artificial neural network. <https://samansiadati.blogspot.com/2019/03/artificial-neural-network.html>.
- [11] Mohamed Shawky. Activation functions – why is there more than 1? <https://medium.com/udacity-pytorch-challengers/activation-functions-why-is-there-more-than-1-5464e6754122>, January 2019.
- [12] Deep Learning Activation Functions. <https://www.nickmccullum.com/python-deep-learning/deep-learning-activation-functions/>.
- [13] ReLU activation function. [https://www.researchgate.net/figure/ReLU-activation-function\\_fig7\\_333411007](https://www.researchgate.net/figure/ReLU-activation-function_fig7_333411007).
- [14] Ekaba Bisong. Deep Learning. [https://ekababisong.org//ieee-ompi-workshop/deep\\_learning/](https://ekababisong.org//ieee-ompi-workshop/deep_learning/).

- [15] Graphic representation of the softmax activation function. [https://www.researchgate.net/figure/Graphic-representation-of-the-softmax-activation-function\\_fig5\\_348703101](https://www.researchgate.net/figure/Graphic-representation-of-the-softmax-activation-function_fig5_348703101).
- [16] What is Overfitting? | IBM. <https://www.ibm.com/topics/overfitting>.
- [17] Convolutional Neural Network architecture. [https://www.researchgate.net/figure/Convolutional-Neural-Network-architecture\\_fig1\\_347776755](https://www.researchgate.net/figure/Convolutional-Neural-Network-architecture_fig1_347776755).
- [18] Rikiya Yamashita, Mizuho Nishio, Richard Kinh Gian Do, and Kaori Togashi. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, 9(4):611--629, August 2018.
- [19] How Convolutional layer work exactly in RGB image processing. <https://stats.stackexchange.com/q/534080>, July 2021.
- [20] Xiang Li, Yabo Dan, Rongzhi Dong, Zhuo Cao, Chengcheng Niu, Yuqi Song, Shaobo Li, and Jianjun Hu. Computational Screening of New Perovskite Materials Using Transfer Learning and Deep Learning. *Applied Sciences*, 9(24):5510, January 2019. Number: 24 Publisher: Multidisciplinary Digital Publishing Institute.
- [21] Autoencoders | Main Components and Architecture of Autoencoder. <https://www.educba.com/autoencoders/>, October 2019.
- [22] Xifeng Guo, Xinwang Liu, En Zhu, and Jianping Yin. Deep clustering with convolutional autoencoders. pages 373--382, 10 2017.
- [23] Saul Dobillas. Denoising Autoencoders (DAE) — How To Use Neural Networks to Clean Up Your Data. <https://towardsdatascience.com/denoising-autoencoders-dae-how-to-use-neural-networks-to-clean-up-your-data-cd9c19bc6915>, April 2022.
- [24] Syoya Zhou. What happens in sparse autencoder. <https://medium.com/@syoya/what-happens-in-sparse-autencoder-b9a5a69da5c6>, December 2018.
- [25] Different types of Autoencoders. <https://iq.opengenus.org/types-of-autoencoder/>, July 2019.
- [26] Jie Feng, Xueliang Feng, Jiantong Chen, Xianghai Cao, Xiangrong Zhang, Licheng Jiao, and Tao Yu. Generative adversarial networks based on collaborative learning and attention mechanism for hyperspectral image classification. *Remote Sensing*, 12:1149, 04 2020.
- [27] Midjourney Showcase. <https://www.midjourney.com/showcase/recent/>.
- [28] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders, 2017.
- [29] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Deep convolutional autoencoder-based lossy image compression, 2018.
- [30] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Variable rate deep image compression with a conditional autoencoder, 2019.
- [31] Mu Li, Wangmeng Zuo, Shuhang Gu, Debin Zhao, and David Zhang. Learning convolutional networks for content-weighted image compression, 2017.

- [32] Feng Jiang, Wen Tao, Shaohui Liu, Jie Ren, Xun Guo, and Debin Zhao. An end-to-end compression framework based on convolutional neural networks, 2017.
- [33] Asha Lata and Permender Singh. Review of Image Compression Techniques. 3(7), 2013.
- [34] C. E. Shannon. A mathematical theory of communication. The Bell System Technical Journal, 27(3):379--423, 1948.
- [35] David A. Huffman. A method for the construction of minimum-redundancy codes. Proceedings of the IRE, 40(9):1098--1101, 1952.
- [36] Hardeep Samra. Image compression techniques. INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY, 2:49--52, 04 2003.
- [37] C. Chapin Cutler. [https://en.wikipedia.org/w/index.php?title=C\\_\char"0307\relaxChapin\\_Cutler](https://en.wikipedia.org/w/index.php?title=C_\char), March 2023. Page Version ID: 1142438342.
- [38] Athira B. Kaimal and S. Manimurugan. Image compression techniques: A survey. 2013.
- [39] V.K. Goyal. Theoretical foundations of transform coding. IEEE Signal Processing Magazine, 18(5):9--21, 2001.
- [40] Wikipedia contributors. Block truncation coding --- Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Block\\_Truncation\\_Coding&oldid=1023847614](https://en.wikipedia.org/w/index.php?title=Block_Truncation_Coding&oldid=1023847614), 2021.
- [41] Guojun Lu. Fractal image compression. Signal Processing: Image Communication, 5(4):327--343, 1993.
- [42] Graham Hudson, Alain Léger, Birger Niss, Istvan Sebestyén, and Jørgen Vaaben. JPEG-1 standard 25 years: past, present, and future reasons for a success. J. Electronic Imaging, 27(04):040901, 2018.
- [43] A. M. Raid, W. M. Khedr, Mohamed A. El-Dosuky, and Wesam Ahmed. Jpeg image compression using discrete cosine transform - A survey. CoRR, abs/1405.6147, 2014.
- [44] Gregory K. Wallace. The jpeg still picture compression standard. Commun. ACM, 34(4):30–44, apr 1991.
- [45] Image Quality Metrics - MATLAB & Simulink. <https://www.mathworks.com/help/images/image-quality-metrics.html>.
- [46] Wikipedia contributors. Weak supervision --- Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Weak\\_supervision&oldid=1162532084](https://en.wikipedia.org/w/index.php?title=Weak_supervision&oldid=1162532084), 2023. [Online; accessed 3-July-2023].
- [47] What is Reinforcement Learning? | Definition from TechTarget. <https://www.techtarget.com/searchenterpriseai/definition/reinforcement-learning>.
- [48] George V. Cybenko. Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals and Systems, 2:303--314, 1989.
- [49] SAGAR SHARMA. Activation Functions in Neural Networks. <https://towardsdatascience.com/activation-functions-neural-networks-1cbd9f8d91d6>, November 2022.

- [50] Jeff Heaton. Ian Goodfellow, Yoshua Bengio, and Aaron Courville: Deep learning. Genetic Programming and Evolvable Machines, 19(1):305--307, June 2018.
- [51] Radhey Shyam. Convolutional neural network and its architectures. 12:2021, 10 2021.
- [52] Wikipedia contributors. Convolutional neural network Wikipedia, the free encyclopedia. [https://en.wikipedia.org/w/index.php?title=Convolutional\\_neural\\_network](https://en.wikipedia.org/w/index.php?title=Convolutional_neural_network), 2023.
- [53] Bartomeu Coll and Jean-Michel Morel. A review of image denoising algorithms, with a new one. SIAM Journal on Multiscale Modeling and Simulation, 4, 01 2005.
- [54] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. J. Mach. Learn. Res., 11:3371–3408, dec 2010.
- [55] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.
- [56] George Toderici, Sean M. O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable Rate Image Compression with Recurrent Neural Networks, March 2016. arXiv:1511.06085 [cs].
- [57] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. Full resolution image compression with recurrent neural networks, 2017.
- [58] RAISE - Dataset. <http://loki.disi.unitn.it/RAISE/download.html>.

---

Πανεπιστήμιο Πατρών, Πολυτεχνική Σχολή  
Τμήμα Ηλεκτρολόγων Μηχανικών και Τεχνολογίας Υπολογιστών  
Τσουδερός Γεώργιος του Ιωάννη  
© Ιούνιος 2023 -- Με την επιφύλαξη παντός δικαιώματος.

---