ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΕΥΦΥΩΝ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Χρήση τεχνικών βαθιάς μάθησης για την αναγνώριση συναισθημάτων μέσα από εκφράσεις του προσώπου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΧΡΗΣΤΟΥ ΘΕΟΔΩΡΟΠΟΥΛΟΥ

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

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

Αθήνα, Οκτώβρης 2018

Η σελίδα αυτή είναι σκόπιμα λευκή.

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΕΥΦΥΩΝ ΥΠΟΛΟΓΙΣΤΙΚΩΝ
ΣΥΣΤΗΜΑΤΩΝ

Χρήση τεχνικών βαθιάς μάθησης για την αναγνώριση συναισθημάτων μέσα από εκφράσεις του προσώπου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΧΡΗΣΤΟΥ ΘΕΟΔΩΡΟΠΟΥΛΟΥ

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

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

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 3η Οκτώβριου 2018.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
……	……	……
Ανδρέας-Γεώργιος Σταφυλοπάτης Καθηγητής Ε.Μ.Π.	Γεώργιος Στάμου Αν. Καθηγητής Ε.Μ.Π.	Κωνσταντίνα Νικήτα Καθηγήτρια Ε.Μ.Π.

Αθήνα, Οκτώβρης 2018

(Υπογραφή)

……
ΧΡΗΣΤΟΣ ΘΕΟΔΩΡΟΠΟΥΛΟΣ

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

Περίληψη

Τα τελευταία χρόνια έχει παρατηρηθεί μία ραγδαία ανάπτυξη στον τομέα της τεχνητής νοημοσύνης και πιο συγκεκριμένα στις τεχνικές της όρασης των υπολογιστών (computer vision). Η βαθιά μάθηση (deep learning) έχει χρησιμοποιηθεί για να λυθούν αποδοτικά πληθώρα προβλημάτων που σχετίζονται με αναγνώριση προτύπων (pattern) σε εικόνες.

Σκοπός της παρούσας εργασίας είναι η αναγνώριση της συναισθηματικής κατάστασης του ανθρώπου μέσα από εκφράσεις του προσώπου. Το πρόβλημα αυτό είναι ιδιαίτερα σημαντικό στον ευρύ τομέα της αλληλεπίδρασης του ανθρώπου και του υπολογιστή. Ο επιστημονικός κλάδος της επικοινωνίας ανθρώπου-υπολογιστή ασχολείται τόσο με την κατανόηση του πως οι άνθρωποι χρησιμοποιούν τους υπολογιστές, όσο και με τον σχεδιασμό νέων συστημάτων που ενισχύουν την απόδοση και την εμπειρία του ανθρώπου. Η συναισθηματική κατάσταση ενός ανθρώπου, επηρεάζει σημαντικά τη συμπεριφορά και τις αποφάσεις του. Η ανάπτυξη συναισθηματικής νοημοσύνης (emotional intelligence) σε μηχανές είναι ένας κλάδος της τεχνητής νοημοσύνης γεμάτος προκλήσεις.

Τα μοντέλα που δημιουργήθηκαν σε αυτή τη διπλωματική βασίστηκαν σε πολύ αποδοτικά βαθιά συνελικτικά νευρωνικά δίκτυα (CNN) (ResNet-50 και Inception-ResNet-V2) και εκπαιδεύτηκαν έτσι ώστε να μεγιστοποιείται η πιθανότητα εξαγωγής σωστής πρόβλεψης του συναισθήματος. Η αναγνώριση της συναισθηματικής κατάστασης του ανθρώπου μέσα από εκφράσεις του προσώπου προσεγγίζεται ως πρόβλημα παλινδρόμησης (regression) και κατηγοριοποίησης (classification). Πειράματα που εκτελέστηκαν με χρήση της βάσης δεδομένων Semaine, αποδεικνύουν ότι τα βέλτιστα μοντέλα έχουν υψηλή απόδοση.

Λέξεις κλειδιά: Τεχνητή Νοημοσύνη, αναγνώριση συναισθημάτων, εκφράσεις προσώπου, βαθιά μάθηση, συνελικτικά νευρωνικά δίκτυα, ResNet-50, Inception-ResNet-V2, παλινδρόμηση, κατηγοριοποίηση

Η σελίδα αυτή είναι σκόπιμα λευκή.

Abstract

In recent years, a rapid growth has been observed in the field of artificial intelligence and more specifically in computer vision techniques. Deep learning has been used to efficiently solve many problems associated with pattern recognition in images.

The purpose of this work is to recognize the emotional state of human being through facial expressions. This problem is particularly important in the broad field of human and computer interaction. The human-computer communication scientific field deals with the understanding of how people use computers as well as with the design of new systems that enhance human performance and experience. A person's emotional state significantly affects his behavior and decisions. The development of emotional intelligence in machines is a branch of artificial intelligence full of challenges.

The models which were created in this diploma were based on highly efficient deep convolutional neural networks (CNN) (ResNet-50 and Inception-ResNet-V2) and were trained so as to maximize the possibility of extracting proper prediction of emotion. The recognition of the emotional state of the person through facial expressions is approached as a regression and classification problem. Experiments performed using the Semaine database demonstrate that the best models have high performance.

Keywords: Artificial intelligence, emotional recognition, facial expressions, deep learning, convolutional neural networks, ResNet-50, Inception-ResNet-V2, regression, classification

Η σελίδα αυτή είναι σκόπιμα λευκή.

Ευχαριστίες

Σε αυτό το σημείο θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή κ. Ανδρέα-Γεώργιο Σταφυλοπάτη, που με εμπιστεύτηκε και μου έδωσε την ευκαιρία να εκπονήσω τη διπλωματική εργασία σε ένα τομέα, ο οποίος με ενδιαφέρει ιδιαίτερα. Επιπροσθέτως, οφείλω να ευχαριστήσω τον υποψήφιο διδάκτορα Θάνο Τάγαρη για τη συνεχή καθοδήγηση και τη θεμελιώδη συνεισφορά του στην εκπόνηση της παρούσας εργασίας. Η άμεση ανταπόκριση και το ενδιαφέρον του έπαιξαν σημαντικό ρόλο έτσι ώστε να εξαχθεί το βέλτιστο δυνατό αποτέλεσμα. Τέλος θα ήθελα να ευχαριστήσω μέσα από την καρδιά μου την οικογένειά μου και τους φίλους μου για την αγάπη, την αστείρευτη υποστήριξη και την κατανόηση που έδειξαν όλα τα χρόνια της ακαδημαϊκής πορείας μου. Δίχως αυτούς δε θα είχα φτάσει μέχρι εδώ.

Χρήστος Θεοδωρόπουλος
Οκτώβρης 2018

‘Ever tried.
Ever failed.
No matter.
Try again.
Fail again.
Fail better.’

Samuel Beckett

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

1

Εισαγωγή

1.1 Τεχνητή νοημοσύνη

Η τεχνητή νοημοσύνη συνδυάζει μία τεράστια ποικιλία επιμέρους πεδίων, τα οποία καλύπτουν ένα φάσμα που ξεκινά από γενικούς τομείς, όπως η μάθηση και η αντίληψη και φτάνει σε συγκεκριμένες εργασίες όπως το σκάκι, η απόδειξη μαθηματικών θεωρημάτων, η συγγραφή ποίησης, η διάγνωση ασθενειών και η ανίχνευση της συναισθηματικής κατάστασης του ανθρώπου. Η τεχνητή νοημοσύνη συστηματοποιεί και αυτοματοποιεί τις διανοητικές εργασίες, για αυτό και μπορεί να έχει εφαρμογή σε οποιαδήποτε σφαίρα της ανθρώπινης διανοητικής δραστηριότητας. Με αυτή την έννοια, είναι πραγματικά ένα οικουμενικό πεδίο.

Δεν υπάρχει ένας καθολικός ορισμός που να περιγράφει τον όρο «τεχνητή νοημοσύνη». Στον ακόλουθο πίνακα παρατίθενται οχτώ ορισμοί. Οι ορισμοί αυτοί έχουν διαφορές σε δύο κύριες διαστάσεις. Οι επάνω ορισμοί εστιάζουν περισσότερο στις διαδικασίες σκέψης και στη συλλογιστική, ενώ οι κάτω ασχολούνται με τη συμπεριφορά. Οι ορισμοί στα αριστερά μετρούν την επιτυχία με βάση την εγγύτητα προς τις ανθρώπινες επιδόσεις, ενώ οι ορισμοί στα δεξιά τη μετρούν σε σχέση με μία ιδανική έννοια νοημοσύνης, η οποία ονομάζεται ορθολογικότητα. Ένα σύστημα είναι ορθολογικό αν κάνει «το σωστό», με δεδομένα όσα γνωρίζει.

Συστήματα που σκέπτονται σαν τον άνθρωπο.	**Συστήματα που σκέπτομαι ορθολογικά.**
« Η συναρπαστική νέα προσπάθεια για να κάνουμε τους υπολογιστές να σκέπτονται.» (Haugeland, 1985) « Η αυτοματοποίηση των δραστηριοτήτων που σχετίζονται με την ανθρώπινη σκέψη, όπως η λήψη αποφάσεων, η επίλυση προβλημάτων και η μάθηση. » (Bellman, 1978)	« Η μελέτη των νοητικών ικανοτήτων με τη χρήση υπολογιστικών μοντέλων. » (Charniak και McDermott, 1985) « Η μελέτη των υπολογιστικών εργασιών που μας δίνουν τη δυνατότητα να αντιλαμβανόμαστε και να ενεργούμε. » (Winston, 1992)
Συστήματα που ενεργούν σαν τον άνθρωπο.	**Συστήματα που ενεργούν ορθολογικά**
« Η τέχνη της δημιουργίας μηχανών που πραγματοποιούν λειτουργίες που απαιτούν νοημοσύνη όταν πραγματοποιούνται από ανθρώπους. » (Kurzweil, 1990) « Η μελέτη του πως μπορούμε να κάνουμε τους υπολογιστές να κάνουν πράγματα, στα οποία, προς το παρόν, οι άνθρωποι είναι καλύτεροι. » (Rick και Knight, 1991)	« Υπολογιστική νοημοσύνη είναι η μελέτη της σχεδίασης ευφυών πρακτόρων. » (Poole, 1998) « Η τεχνητή νοημοσύνη ασχολείται με την ευφυή συμπεριφορά των τεχνουργημάτων. » (Nilsson, 1998)
Πίνακας 1.1 Ορισμοί της τεχνητής νοημοσύνης ([2])	

Ένας τυπικός ορισμός του όρου «τεχνητή νοημοσύνη», που χρησιμοποιείται συχνά, δόθηκε από τον T. M. Mitchell και είναι ο εξής: «Ένα υπολογιστικό πρόγραμμα λέγεται ότι μαθαίνει από την εμπειρία E σε σχέση με κάποια τάξη εργασιών T και μέτρηση απόδοσης P εάν η απόδοσή του σε εργασίες στο T, όπως υπολογίζεται με βάση την P, βελτιώνεται με την εμπειρία E.» ('A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T, as measured by P, improves with experience E.') ([35]).

1.1.1 Μηχανική μάθηση

Ως μηχανική μάθηση ορίζεται το ευρύ επιστημονικό πεδίο της τεχνητής νοημοσύνης, το οποίο δίνει τη δυνατότητα στους υπολογιστές να «μαθαίνουν» με δεδομένα και να βελτιώνονται με το πέρασμα του χρόνου με αυτόματο τρόπο, ενώ τροφοδοτούνται με πληροφορία. Η μηχανική μάθηση διακρίνεται σε επιβλεπόμενη, μη επιβλεπόμενη και ενισχυτική μάθηση.

1.2 Νευρωνικά δίκτυα

Το έργο στο επιστημονικό πεδίο των νευρωνικών δικτύων βασίστηκε, από τις απαρχές, στο γεγονός ότι ο ανθρώπινος εγκέφαλος εκτελεί τους υπολογισμούς με εντελώς διαφορετικό τρόπο από το συμβατικό ψηφιακό υπολογιστή. Ο εγκέφαλος είναι ένας εξαιρετικά πολύπλοκος, μη γραμμικός, παράλληλος επεξεργαστής. Έχει τη δυνατότητα να οργανώνει τα δομικά του στοιχεία, γνωστά ως νευρώνες, με τρόπο ώστε να εκτελούν συγκεκριμένους υπολογισμούς (π.χ. αναγνώριση προτύπων, αντίληψη και έλεγχο κίνησης) με ταχύτητα πολλαπλάσια από αυτή του γρηγορότερου ψηφιακού υπολογιστή που υπάρχει σήμερα.

Ο χαρακτηρισμός ενός νευρωνικού συστήματος ως «εξελισσόμενο» είναι συνώνυμος με την έννοια της πλαστικότητας: αυτή δίνει στο νευρικό σύστημα τη δυνατότητα να προσαρμόζεται ανάλογα με το περιβάλλον του. Και, ακριβώς όπως είναι ζωτική για τη λειτουργία των νευρώνων ως μονάδες επεξεργασίας πληροφοριών στον ανθρώπινο εγκέφαλο, είναι εξίσου σημαντική για τα νευρωνικά δίκτυα που αποτελούνται από τεχνητούς νευρώνες. Στην πλέον γενική μορφή του, ένα νευρωνικό δίκτυο είναι μία μηχανή σχεδιασμένη να προσομοιώνει τον τρόπο με τον οποίο ο εγκέφαλος εκτελεί μία συγκεκριμένη εργασία ή λειτουργία.

Ένα τεχνητό νευρωνικό δίκτυο είναι ένας μεγάλος παράλληλος επεξεργαστής με κατανεμημένη αρχιτεκτονική, ο οποίος αποτελείται από απλές μονάδες επεξεργασίας και έχει από τη φύση του τη δυνατότητα να αποθηκεύει εμπειρική γνώση και να την καθιστά διαθέσιμη για χρήση. Μοιάζει με τον ανθρώπινο εγκέφαλο σε δύο σημεία:

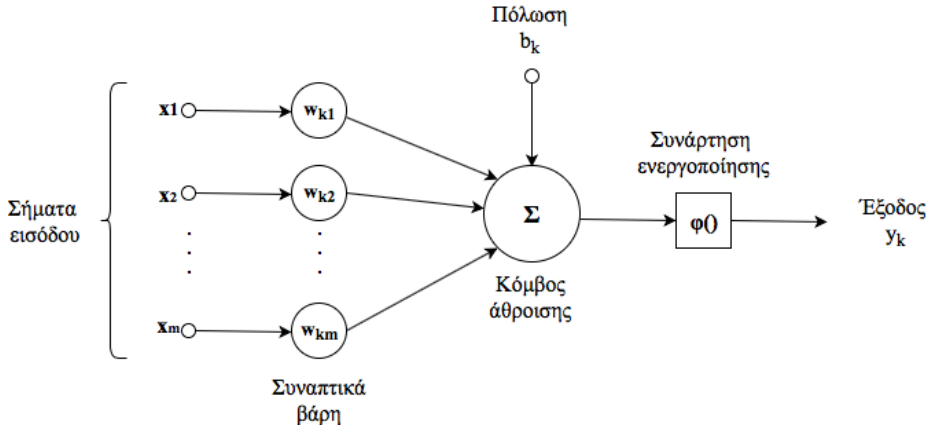
1. Το δίκτυο προσλαμβάνει τη γνώση από το περιβάλλον του, μέσω μίας διαδικασίας μάθησης.
2. Η ισχύς των συνδέσεων μεταξύ των νευρώνων, που αποκαλείται συναπτικό βάρος, χρησιμοποιείται για την αποθήκευση της γνώσης που αποκτιέται.

Η διαδικασία μέσω της οποίας επιτυγχάνεται η μάθηση αποκαλείται αλγόριθμος μάθησης και η λειτουργία του είναι να τροποποιεί τα συναπτικά βάρη του δικτύου με τον κατάλληλο τρόπο για την επίτευξη του επιθυμητού στόχου.

Το νευρωνικό δίκτυο οφείλει την υπολογιστική του ισχύ κατά πρώτον στην παράλληλη, κατανεμημένη δομή του και κατά δεύτερον στην ικανότητά του να μαθαίνει και, ως εκ τούτου, να γενικεύει. Ο όρος γενίκευση αναφέρεται στην παραγωγή, από το νευρωνικό δίκτυο, λογικών εξόδων για εισόδους τις οποίες δεν έχει συναντήσει κατά τη διάρκεια της εκπαίδευσής του. Αυτές οι δύο δυνατότητες δίνουν στα νευρωνικά δίκτυα την ικανότητα να βρίσκουν καλές προσεγγιστικές λύσεις σε πολύπλοκα προβλήματα, τα οποία είναι μη επιδεκτικά σε λύσεις.

1.2.1 Μοντέλα νευρώνων

Ένας νευρώνας είναι μία μονάδα επεξεργασίας πληροφορίας, η οποία είναι θεμελιώδης για τη λειτουργία ενός νευρωνικού δικτύου. Το ακόλουθο διάγραμμα παρουσιάζει το μοντέλο ενός νευρώνα που αποτελεί τη βάση για τη σχεδίαση μίας μεγάλης οικογένειας νευρωνικών δικτύων.



Σχήμα 1.1: Μοντέλο νευρώνα ([1])

Τα τρία βασικά στοιχεία αυτού του νευρώνα είναι:

1. Ένα σύνολο συνάψεων (ή διασυνδέσεων), κάθε μία εκ των οποίων χαρακτηρίζεται από το δικό της βάρος ή δύναμη. Συγκεκριμένα, ένα σήμα x_j στην είσοδο της σύναψης j που συνδέεται με το νευρώνα k πολλαπλασιάζεται επί το συναπτικό βάρος w_{kj}. Ο πρώτος δείκτης στο w_{kj} αναφέρεται στον εν λόγω νευρώνα και ο δεύτερος δείκτης αναφέρεται στο άκρο εισόδου της σύναψης στην οποία αναφέρεται το βάρος. Ανόμοια με το βάρος μία σύναψης στον ανθρώπινο εγκέφαλο, το συναπτικό βάρος ενός τεχνητού νευρώνα μπορεί να λαμβάνει και αρνητικές και θετικές τιμές.
2. Έναν αθροιστή για την άθροιση των σημάτων εισόδου, σταθμισμένων από τα αντίστοιχα συναπτικά βάρη του νευρώνα.
3. Μία συνάρτηση ενεργοποίησης για τον περιορισμό του πλάτους τους σήματος εξόδου ενός νευρώνα. Η συνάρτηση ενεργοποίησης αναφέρεται επίσης ως συνάρτηση περιορισμού, επειδή περιορίζει το επιτρεπτό εύρος πλάτους του σήματος εξόδου σε κάποια πεπερασμένη τιμή. Τυπικά, το κανονικοποιημένο εύρος τιμών πλάτους της εξόδου ενός νευρώνα γράφεται ως κλειστό διάστημα, με τη μορφή [0, 1] ή [-1, 1]. Οι πιο συνηθισμένες συναρτήσεις ενεργοποίησης είναι η βηματική, η συνάρτηση προσήμου, η ταυτοτική συνάρτηση (στην περίπτωση γραμμικών νευρώνων) και η σιγμοειδής συνάρτηση.

Το μοντέλο του νευρώνα του διαγράμματος 1.1 περιλαμβάνει επίσης μία εξωτερικά εφαρμοζόμενη πόλωση, η οποία συμβολίζεται ως b_k. Η πόλωση έχει ως αποτέλεσμα την αύξηση ή τη μείωση της δικτυακής διέγερσης της συνάρτησης ενεργοποίησης, ανάλογα με το εάν είναι θετική ή αρνητική, αντίστοιχα.

Με μαθηματικούς όρους ο νευρώνας k που απεικονίζεται στο διάγραμμα μπορεί να περιγραφεί με το ακόλουθο ζεύγος εξισώσεων.

$$u_k = \sum_{j=1}^{m} w_{kj}\, x_j \quad (1)$$

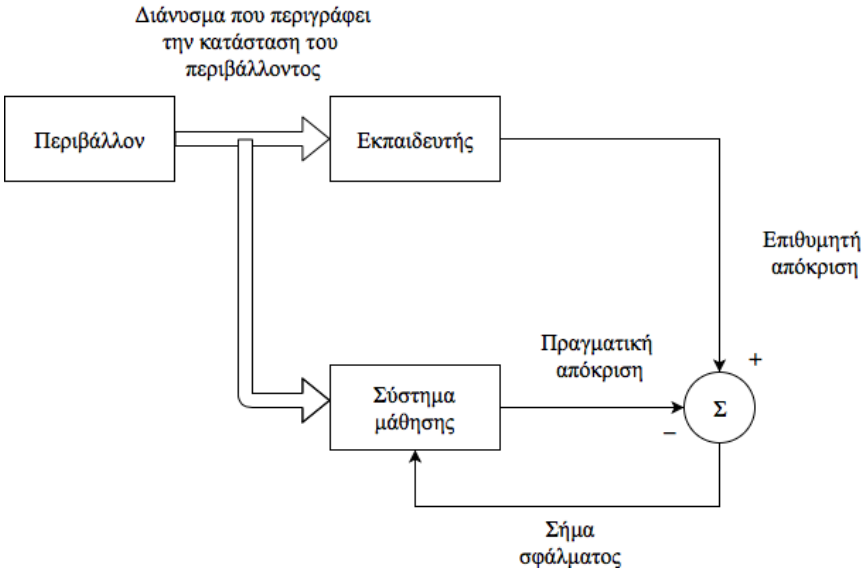
$$y_k = \varphi(u_k + b_k) \quad (2)$$

1.2.2 Βαθιά μηχανική μάθηση

Εφόσον έχει οριστεί η έννοια του τεχνητού νευρωνικού δικτύου, καθίσταται δυνατό να αποτυπωθεί η έννοια της βαθιάς μηχανικής μάθησης. Η βαθιά μηχανική μάθηση είναι η διαδικασία εφαρμογής «μαθησιακών διεργασιών» σε νευρωνικά δίκτυα πολλών επιπέδων. Αποτελεί κομμάτι μίας ευρύτερης οικογένειας μεθόδων μηχανικής μάθησης βασιζόμενων στην αναπαράσταση δεδομένων, σε αντίθεση με αλγορίθμους επικεντρωμένους σε υπολογιστικές εργασίες. Οι αρχιτεκτονικές βαθιάς μηχανικής μάθησης έχουν εφαρμογή σε πολλά πεδία της τεχνητής νοημοσύνης, όπως η όραση των υπολογιστών, η αναγνώριση της φωνής και η βιοιατρική.

1.2.3 Επιβλεπόμενη μάθηση τεχνητών νευρωνικών δικτύων

Η επιβλεπόμενη μάθηση βασίζεται στην υπόθεση ότι ένα σύνολο δεδομένων με ετικέτες (labels) είναι διαθέσιμο κατά την εκπαίδευση του νευρωνικού δικτύου. Το διάγραμμα 1.2 περιγράφει αυτή τη μορφή μάθησης.



Σχήμα 1.2: Σχηματικό διάγραμμα της μάθησης με εκπαίδευση ([1])

Με εννοιολογικούς όρους, ο εκπαιδευτής έχει γνώση του περιβάλλοντος και αυτή η γνώση αντιπροσωπεύεται από ένα σύνολο παραδειγμάτων εισόδου-εξόδου. Ωστόσο, το περιβάλλον είναι άγνωστο στο νευρωνικό δίκτυο. Χάρη στην εγγενή του γνώση, ο εκπαιδευτής είναι σε θέση να παρέχει στο νευρωνικό δίκτυο μία επιθυμητή απόκριση για το κάθε διάνυσμα εκπαίδευσης. Η επιθυμητή απόκριση αντιπροσωπεύει τη «βέλτιστη» ενέργεια που πρέπει να εκτελείται από το νευρωνικό δίκτυο. Οι παράμετροι του δικτύου προσαρμόζονται υπό τη συνδυασμένη επιρροή του διανύσματος εκπαίδευσης και του σήματος σφάλματος. Το σήμα σφάλματος ορίζεται ως η διαφορά μεταξύ της επιθυμητής απόκρισης και της πραγματικής απόκρισης του δικτύου. Αυτή η προσαρμογή εκτελείται με επαναληπτικό τρόπο, βήμα προς βήμα, με στόχο να φέρει τελικά το νευρωνικό δίκτυο σε μία κατάσταση όπου θα προσομοιώνει τη συμπεριφορά του εκπαιδευτή· η προσομοίωση κρίνεται ως «βέλτιστη» με κάποια στατιστική έννοια. Κατ' αυτό τον τρόπο, η γνώση του περιβάλλοντος που είναι διαθέσιμη στον εκπαιδευτή μεταφέρεται στο νευρωνικό δίκτυο μέσω εκπαίδευσης και αποθηκεύεται με τη μορφή «σταθερών» συναπτικών βαρών, τα οποία αντιπροσωπεύουν μακροπρόθεσμη μνήμη.

Η διαδικασία επιβλεπόμενης μάθησης συνιστά ένα σύστημα ανάδρασης κλειστού βρόχου, αλλά το άγνωστο περιβάλλον βρίσκεται εκτός του βρόχου. Ως μέτρο απόδοσης για το σύστημα μπορεί να χρησιμοποιηθεί το μέσο τετραγωνικό σφάλμα ή το άθροισμα των τετραγώνων των σφαλμάτων επί του δείγματος εκπαίδευσης, ορισμένο ως συνάρτηση των ελεύθερων παραμέτρων (δηλαδή των συναπτικών βαρών) του συστήματος. Αυτή η λειτουργία μπορεί να σχηματοποιηθεί ως μία πολυδιάστατη επιφάνεια σφάλματος-απόδοσης, ή απλώς επιφάνεια σφάλματος, με τις ελεύθερες παραμέτρους σα συντεταγμένες. Η πραγματική επιφάνεια σφάλματος υπολογίζεται ως μέσος όρος επί όλων των πιθανών παραδειγμάτων εισόδου-εξόδου. Οποιαδήποτε συγκεκριμένη λειτουργία του συστήματος υπό την επίβλεψη του εκπαιδευτή αναπαρίσταται ως ένα σημείο στην επιφάνεια σφάλματος.

Για να μπορεί το σύστημα να βελτιώνει την απόδοσή του με την πάροδο του χρόνου και, κατά συνέπεια, να μαθαίνει από τον εκπαιδευτή, το σημείο λειτουργίας του πρέπει να μετακινείται διαδοχικά προς τα κάτω, προς ένα ελάχιστο σημείο της επιφάνειας σφάλματος· το ελάχιστο σημείο μπορεί να είναι ένα τοπικό ελάχιστο ή ένα γενικό ελάχιστο. Ένα σύστημα επιβλεπόμενης μάθησης έχει τη δυνατότητα να το κάνει αυτό με τη χρήσιμη πληροφορία που διαθέτει σχετικά με την κλίση της επιφάνειας σφάλματος που αντιστοιχεί στην τρέχουσα συμπεριφορά του συστήματος. Η κλίση υπολογίζεται από την παράγωγο του σφάλματος.

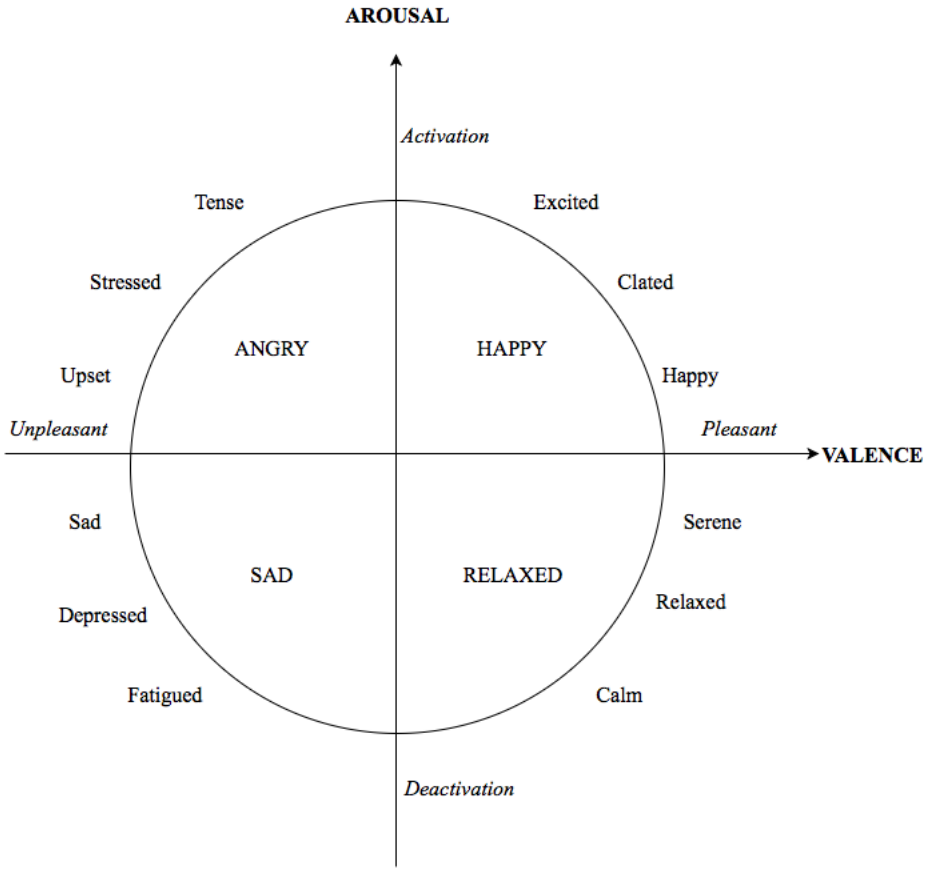
1.3 Αναγνώριση εκφράσεων προσώπου

Ο επιστημονικός κλάδος της επικοινωνίας ανθρώπου-υπολογιστή ασχολείται τόσο με την κατανόηση του πως οι άνθρωποι χρησιμοποιούν τους υπολογιστές, όσο και με τον σχεδιασμό νέων συστημάτων που ενισχύουν την απόδοση και την εμπειρία του ανθρώπου. Η συναισθηματική κατάσταση ενός ανθρώπου, επηρεάζει σημαντικά τη λειτουργία και τις αποφάσεις του. Θα ήταν σημαντικό, λοιπόν, στον κλάδο της επικοινωνίας ανθρώπου-υπολογιστή, να γνώριζε κανείς την συναισθηματική κατάσταση του χρήστη.

Ο άνθρωπος είναι εκπαιδευμένος να αναγνωρίζει τις καταστάσεις αυτές με έμμεσο τρόπο, από την έκφραση του προσώπου του, τον τόνο της φωνής του και από τις κινήσεις του. Πιο συγκεκριμένα, το πρόσωπο είναι ένας από τους πιο πολύπλοκους τρόπους που έχει ο άνθρωπος για να στέλνει σήματα στους γύρω του. Περιλαμβάνει

πάνω από 40 ξεχωριστούς μύες που λειτουργούν αυτόνομα και ανεξάρτητα ο ένας απ' τον άλλον και είναι ο κυρίαρχος τρόπος από τον οποίον οι άνθρωποι αναγνωρίζουν συναισθήματα. Για τον λόγο αυτό, η ερευνητική κοινότητα έχει ασχοληθεί εκτενώς με την αναγνώριση της έκφρασης του προσώπου.

Υπάρχουν δύο προσεγγίσεις για την κατηγοριοποίηση του συναισθήματος. Η μία ορίζει ένα σύνολο «βασικών» συναισθημάτων και θεωρεί πως τα υπόλοιπα συναισθήματα είναι όλα διακριτά μεταξύ τους και πηγάζουν από τα βασικά. Η δεύτερη ορίζει ένα σύστημα δύο συντεταγμένων αρεσκείας (valence) – διέγερσης (arousal) και τοποθετεί τα συναισθήματα στο χώρο αυτό. Στο ακόλουθο διάγραμμα απεικονίζεται η δεύτερη προσέγγιση. Σε κάθε τεταρτημόριο καταγράφεται το βασικό συναίσθημα ([39]).



Σχήμα 1.3: Σύστημα συντεταγμένων Valence – Arousal

Ο οριζόντιος άξονας είναι αυτός της αρέσκειας. Δεξιά βρίσκονται τα ευχάριστα συναισθήματα, ενώ προς τα αριστερά το συναίσθημα γίνεται δυσάρεστο για τον άνθρωπο. Ο κάθετος άξονας της διέγερσης δείχνει πόσο ενεργό-ισχυρό είναι το συναίσθημα. Μεγαλύτερες τιμές στον άξονα της διέγερσης χαρακτηρίζουν πιο έντονα συναισθήματα, ενώ χαμηλότερες πιο ήπια.

2

Θεωρητικό Υπόβαθρο

2.1 Συνελικτικά νευρωνικά δίκτυα (CNN)

2.1.1 Ορισμός

Ένα συνελικτικό δίκτυο είναι μία νευρωνική αρχιτεκτονική πολλών επιπέδων ειδικά σχεδιασμένη ώστε να αναγνωρίζει δισδιάστατα σχήματα με υψηλό βαθμό μη ευαισθησίας στη μετατόπιση, την κλιμάκωση, την στρέβλωση και άλλες μορφές παραμόρφωσης. Αυτή η δύσκολη εργασία διδάσκεται με επιβλεπόμενο τρόπο, μέσω ενός δικτύου του οποίου η δομή περιλαμβάνει τις ακόλουθες μορφές περιορισμών:

1. Εξαγωγή χαρακτηριστικών: Κάθε νευρώνας λαμβάνει τις συναπτικές εισόδους του από ένα τοπικό δεκτικό πεδίο του προηγούμενου επιπέδου, υποχρεώνοντάς το να εξάγει τοπικά χαρακτηριστικά. Αφού εξαχθεί ένα χαρακτηριστικό, η ακριβής θέση του γίνεται λιγότερο σημαντική, εφόσον διατηρείται προσεγγιστικά η σχετική του θέση ως προς άλλα χαρακτηριστικά.
2. Αντιστοίχιση χαρακτηριστικών: Κάθε υπολογιστικό επίπεδο του δικτύου απαρτίζεται από πολλαπλούς χάρτες χαρακτηριστικών, με κάθε χάρτη χαρακτηριστικών να είναι στη μορφή ενός επιπέδου μέσα στο οποίο οι μεμονωμένοι νευρώνες ελέγχονται ώστε να μοιράζονται το ίδιο σύνολο συναπτικών βαρών. Αυτή η δεύτερη μορφή δομικού περιορισμού έχει τα ακόλουθα επακόλουθα:
    i. μη ευαισθησία ως προς τη μετατόπιση, η οποία επιβάλλεται στη λειτουργία ενός χάρτη χαρακτηριστικών μέσω της χρήσης μίας συνέλιξης με έναν πυρήνα μικρού μεγέθους, η οποία ακολουθείται από την εφαρμογή μίας συνάρτησης ενεργοποίησης (πχ. ReLU).
    ii. μείωση του αριθμού των ελεύθερων παραμέτρων, η οποία επιτυγχάνεται μέσω διαμοιρασμού βαρών.
3. Υποδειγματοληψία: Κάθε συνελικτικό επίπεδο ακολουθείται από ένα υπολογιστικό επίπεδο το οποίο εκτελεί υποδειγματοληψία (πχ. average pooling). Έτσι, η ανάλυση του χάρτη χαρακτηριστικών μειώνεται. Αυτή η λειτουργία έχει ως αποτέλεσμα τη μείωση της ευαισθησίας της εξόδου του χάρτη χαρακτηριστικών στις μετατοπίσεις και άλλες μορφές παραμόρφωσης. ([1])
4. Εξαγωγή προβλέψεων: Συνήθως στο τέλος της αρχιτεκτονικής προστίθεται ένα ή περισσότερα πλήρως συνδεδεμένα επίπεδα έτσι ώστε να εξαχθεί η τελική πρόβλεψη.

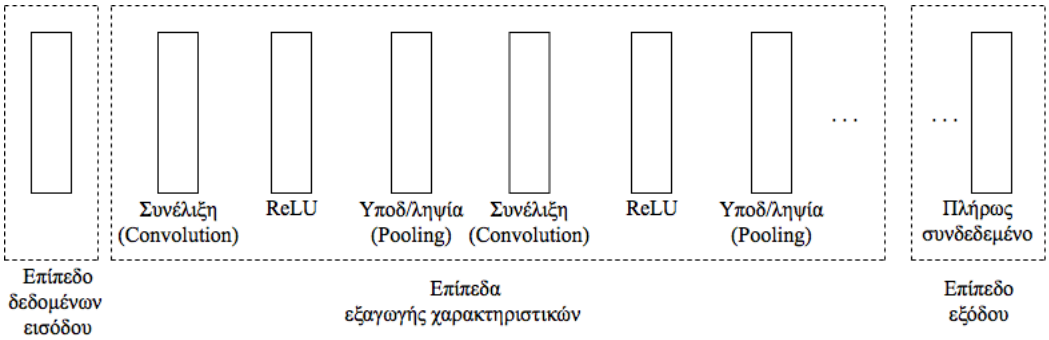
2.1.2 Επισκόπηση αρχιτεκτονικής

Μία αφηρημένη περιγραφή της δομής ενός συνελικτικού δικτύου είναι η ακόλουθη:

$$x^1 \rightarrow [w^1] \rightarrow x^2 \rightarrow [w^2] \rightarrow \cdots \rightarrow x^L \rightarrow [w^L] \rightarrow z \quad (1)$$

Η παραπάνω έκφραση (1) εξηγεί πως λειτουργεί ένα συνελικτικό δίκτυο ανά επίπεδο για προωθητικό πέρασμα (forward pass). Η είσοδος είναι το x^1, συνήθως μία εικόνα, η οποία περνά από διαδοχικά επίπεδα επεξεργασίας μέχρι να εξαχθεί η τελική έξοδος z. Ως επίπεδο επεξεργασίας ορίζεται το $[w^i]$, όπου w είναι το διάνυσμα παραμέτρων του επιπέδου. Κάθε επίπεδο επεξεργασίας δέχεται μία είσοδο, τη μετασχηματίζει και εξάγει μία έξοδο, η οποία αποτελεί είσοδο του επόμενου επιπέδου. Η διαδικασία είναι σειριακή. Όταν ολοκληρωθεί το προωθητικό πέρασμα, ξεκινά μία επιπλέον διαδικασία, η οπισθοδιάδοση σφάλματος (backward error propagation), η οποία είναι απαραίτητη για την αποδοτική εκπαίδευση του συνελικτικού νευρωνικού δικτύου.

Υπάρχει μεγάλη ποικιλία στις αρχιτεκτονικές συνελικτικών νευρωνικών δικτύων, όμως ακολουθείται συνήθως μία συγκεκριμένη γενική μορφή.



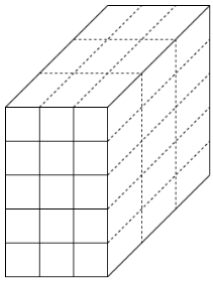
Σχήμα 2.1: Γενική αρχιτεκτονική συνελικτικού δικτύου ([4])

2.1.3 Επίπεδα επεξεργασίας

Στο σχήμα 2.1 παρουσιάζονται όλα τα βασικά δομικά επίπεδα ενός συνελικτικού δικτύου. Η κατανόηση του τρόπου λειτουργίας κάθε επιπέδου είναι σημαντική για την ορθή και αποδοτική δημιουργία ενός συνελικτικού δικτύου.

2.1.3.1 Επίπεδο εισόδου (Input layer)

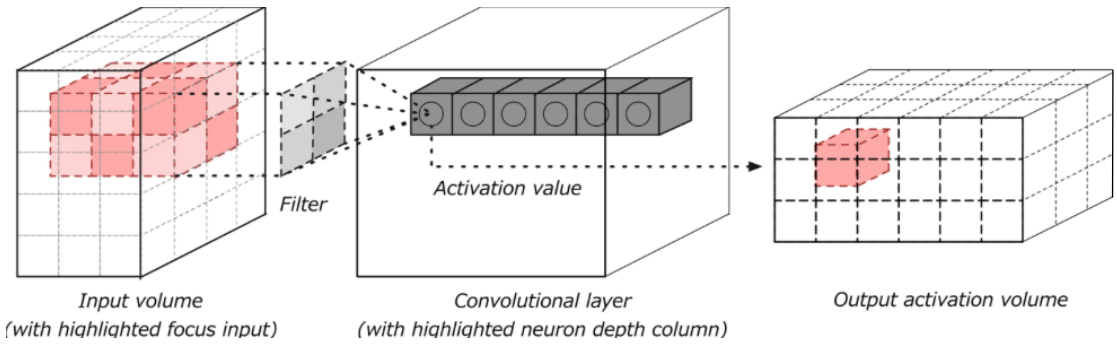
Το επίπεδο εισόδου είναι το αρχικό επίπεδο του δικτύου. Φορτώνει και αποθηκεύει τα ακατέργαστα δεδομένα. Στη πιο συνηθισμένη περίπτωση που τα δεδομένα εισόδου είναι εικόνες, τότε αυτά καθορίζουν το πλάτος, το ύψος και τον αριθμό των καναλιών. Τυπικά, τα κανάλια είναι τρία, για τις τιμές RGB για κάθε εικονοστοιχείο.



Σχήμα 2.2: Επίπεδο εισόδου, 3D αναπαράσταση ([4])

2.1.3.2 Συνελικτικό επίπεδο (convolution layer)

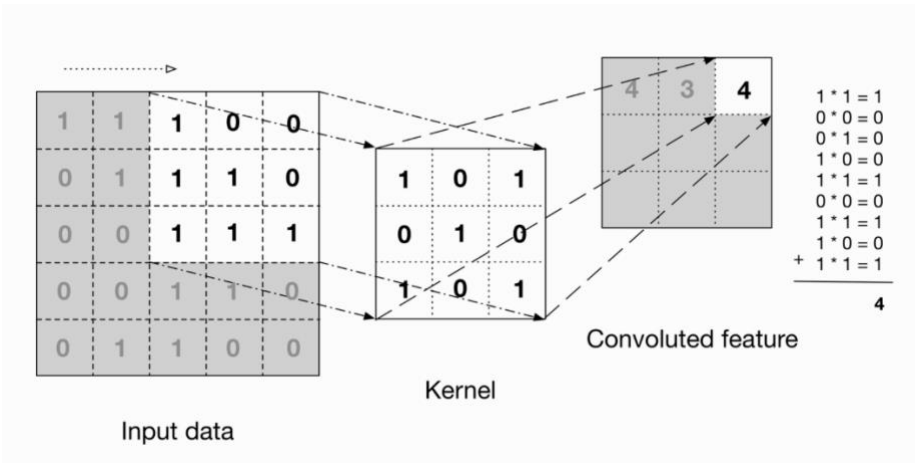
Το συνελικτικό επίπεδο είναι το πιο βασικό δομικό μέρος στην αρχιτεκτονική του συνελικτικού νευρωνικού δικτύου. Δέχεται τα δεδομένα και τα μετασχηματίζει, χρησιμοποιώντας ένα σύνολο συνδεδεμένων νευρώνων του προηγούμενου επιπέδου. Ο κύριος στόχος αυτού του επιπέδου είναι η εξαγωγή χαρακτηριστικών από την εικόνα εισόδου.



Σχήμα 2.3: Συνελικτικό επίπεδο με είσοδο και έξοδο ([4])

2.1.3.2.1 Συνέλιξη (convolution)

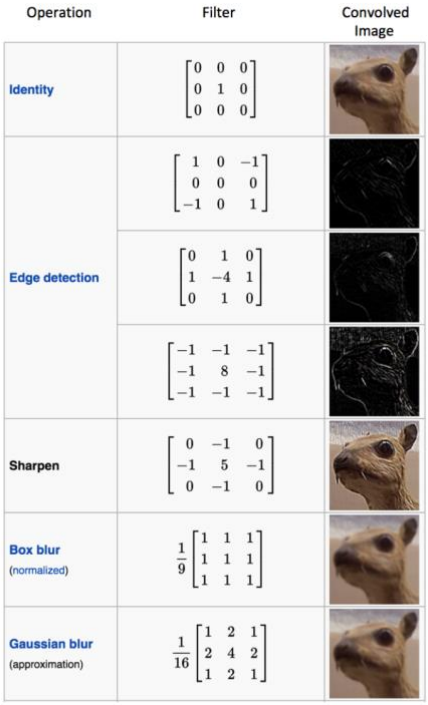
Συνέλιξη ορίζεται η μαθηματική διαδικασία που περιγράφει έναν κανόνα σύμφωνα με τον οποία ενώνουμε την πληροφορία από δύο διαφορετικά μέρη. Στα συνελικτικά δίκτυα η συνέλιξη είναι γνωστή και ως ανιχνευτής χαρακτηριστικών (feature detector). Η είσοδος της συνέλιξης μπορεί να είναι ακατέργαστα δεδομένα (raw data) ή ένας χάρτης χαρακτηριστικών (feature map) που προήλθε από προηγούμενου επίπεδο του δικτύου.



Σχήμα 2.4: Σχηματική περιγραφή της συνέλιξης ([4])

Ο πυρήνας (kernel) ή αλλιώς φίλτρο ολισθαίνει στα δεδομένα εισόδου, έτσι ώστε να παραχθεί το συνελικτικό χαρακτηριστικό (convoluted feature). Σε κάθε βήμα, τα στοιχεία του φίλτρου πολλαπλασιάζονται ένα προς ένα (element wise) με τα

αντίστοιχα στοιχεία του πίνακα εισόδου. Στην πράξη, η έξοδος της διαδικασίας έχει μεγαλύτερη τιμή αν το χαρακτηριστικό που αναζητείται ανιχνεύεται στην είσοδο. Ο πίνακας εξόδου που προκύπτει ονομάζεται χάρτης χαρακτηριστικών (feature map) ή χάρτης ενεργοποίησης (activation map). Στη λειτουργία της συνέλιξης εφαρμόζονται πολλά διαφορετικά φίλτρα, έτσι ώστε να εξαχθεί πολλή πληροφορία από τα δεδομένα εισόδου. Οι πίνακες χαρακτηριστικών που δημιουργούνται συνενώνονται έτσι ώστε να εξαχθεί μία τρισδιάστατη έξοδος.



Σχήμα 2.5: Πίνακες χαρακτηριστικών με χρήση διαφορετικών φίλτρων ([20])

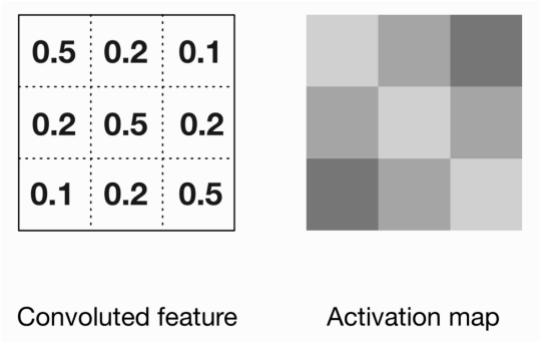
2.1.3.2.2 Φίλτρα (filters)

Τα φίλτρα εφαρμόζονται στα δεδομένα εισόδου με τη λογική ενός κινούμενου παραθύρου. Η έξοδος του φίλτρου υπολογίζεται παράγοντας το άθροισμα του ένα-προς-ένα πολλαπλασιασμού (element-wise product) των στοιχείων του φίλτρου και της περιοχής του πίνακα εισόδου. Οι παράμετροι του συνελικτικού επιπέδου του δικτύου καθορίζουν το σύνολο των φίλτρων που θα χρησιμοποιηθούν.

Η αρχιτεκτονική του συνελικτικού δικτύου ορίζεται με τέτοιο τρόπο ούτως ώστε τα παραγόμενα φίλτρα να εξάγουν την ισχυρότερη ενεργοποίηση σε χωρικά τοπικά πρότυπα εισόδου. Αυτό σημαίνει ότι τα φίλτρα έχουν μάθει ότι θα ενεργοποιηθούν σε μοτίβα (ή χαρακτηριστικά) μόνο όταν τα μοτίβα εμφανίζονται στα δεδομένα εκπαίδευσης στο αντίστοιχο πεδίο. Σε πιο βαθιά επίπεδου του δικτύου τα φίλτρα μπορούν να αναγνωρίσουν μη γραμμικούς συνδυασμούς χαρακτηριστικών και είναι ολοένα και πιο αφηρημένα για το πώς μπορούν να ανιχνεύσουν πρότυπα. Για το λόγο αυτό παρατηρείται ότι τα συνελικτικά δίκτυα με πολύ υψηλή απόδοση έχουν μεγάλο βάθος.

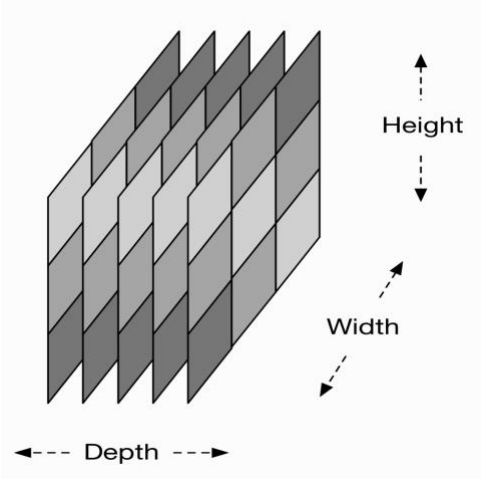
2.1.3.2.3 Πίνακες χαρακτηριστικών – ενεργοποίησης (feature – activation maps)

Κατά τη διάρκεια του εμπρόσθιου περάσματος της πληροφορίας από το συνελικτικό δίκτυο (forward pass) εφαρμόζονται τα φίλτρα στα δεδομένα εισόδου του κάθε επιπέδου της συνέλιξης. Αυτή η διαδικασία παράγει μία δυσδιάστατη έξοδο για κάθε φίλτρο που ονομάζεται πίνακας ενεργοποίησης. Το σχήμα 2.6 απεικονίζει τον τρόπο με τον οποίο αυτός ο χάρτης ενεργοποίησης σχετίζεται με το εξαγόμενο συνελικτικό χαρακτηριστικό.



Σχήμα 2.6: Συνέλιξη και πίνακας ενεργοποίησης ([4])

Όλοι οι πίνακες ενεργοποίησης που εξάγονται στοιβάζονται και δημιουργούν μία τρισδιάστατη αναπαράσταση της εξόδου. Στο σχήμα 2.7 απεικονίζεται μία τέτοια αναπαράσταση.



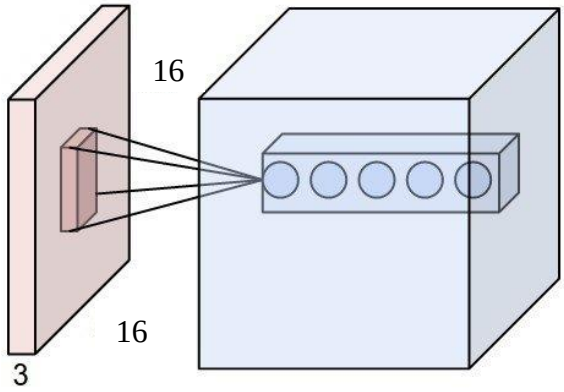
Σχήμα 2.7: Χάρτες ενεργοποίησης

2.1.3.2.4 Τοπική σύνδεση μεταξύ των νευρώνων

Η σύνδεση όλων των νευρώνων με αυτούς του προηγούμενου επίπεδου δεν είναι αποδοτική πρακτική γιατί συχνά η είσοδος είναι πολλών διαστάσεων. Για το λόγο αυτό, συνίσταται η σύνδεση του κάθε νευρώνα με ένα υποσύνολο των νευρώνων του

προηγούμενου επιπέδου. Η χωρική επέκταση της διασύνδεσης αποτελεί μία υπερπαράμετρο του δικτύου και ονομάζεται δεκτικό πεδίο ή μέγεθος φίλτρου.

Οι συνδέσεις είναι τοπικές στο χώρο (κατά ύψος και πλάτος), αλλά πάντα λαμβάνεται υπόψη το συνολικό βάθος του όγκου εισόδου. Για παράδειγμα, αν ο όγκος εισόδου έχει διαστάσεις 16x16x3 και δεκτικό επίπεδο του νευρώνα είναι 5x5, τότε κάθε νευρώνας στο συνελικτικό επίπεδο θα έχει βάρη σε μία περιοχή του όγκου εισόδου διαστάσεων 5x5x3. Επομένως, οι συνολικές τιμές των βαρών του νευρώνα θα είναι 5x5x3 = 75.



Σχήμα 2.8: Ένα παράδειγμα εισόδου διαστάσεων 16x16x3 και ένα παράδειγμα ενός συνόλου νευρώνων στο πρώτο συνελικτό επίπεδο. Κάθε νευρώνας στο συνελικτικό επίπεδο συνδέεται μόνο με μία τοπική περιοχή του όγκου εισόδου χωρικά, αλλά σε όλο το βάθος. Υπάρχουν 5 νευρώνες κατά βάθος. ([4])

2.1.3.2.5 Υπερπαράμετροι συνελικτικού επιπέδου (hyperparameters)

Οι υπερπαράμετροι που καθορίζουν τη χωρική διάταξη και το μέγεθος του όγκου εξόδου από το όγκου εξόδου από ένα συνελικτικό επίπεδο είναι:

1. Μέγεθος φίλτρου: Κάθε φίλτρο είναι μικρό χωροταξικά σε σχέση με το πλάτος και το ύψος του μεγέθους της εισόδου. Για παράδειγμα, το μέγεθος του φίλτρου μπορεί να είναι 5x5x3, ενώ οι διαστάσεις της εισόδου είναι 32x32x3.
2. Βάθος εξόδου: Αντιστοιχεί στο πλήθος των φίλτρων που θα χρησιμοποιηθούν, για τον εντοπισμό διαφορετικών χαρακτηριστικών της εισόδου.
3. Βήμα (stride): Ρυθμίζει πόσο μακριά θα μετακινηθεί το παράθυρο συρόμενου φίλτρου ανά εφαρμογή της λειτουργίας φίλτρου. Όταν το βήμα έχει την τιμή 1, τότε το φίλτρο μετακινείται κατά ένα pixel της φορά (εικόνα). Αύξηση του βήματος επιφέρει χωρική μείωση του όγκου εξόδου.
4. Γέμισμα ακραίων τιμών: Το μέγεθος του γεμίσματος γύρω από τα σύνορα εισόδου. Συνήθης πρακτική είναι η χρήση μηδενικών (zero-padding).

Το χωρικό μέγεθος του όγκου εξόδου είναι μία συνάρτηση του μεγέθους του όγκου εισόδου (I), του μεγέθους του δεκτικού πεδίου του συνελικτικού επιπέδου (F), του βήματος που εφαρμόζεται (S) και του πλήθους των μηδενικών τα οποία χρησιμοποιήθηκαν στο γέμισμα με μηδενικά (Z).

$$\pi\lambda\acute{\eta}\theta os\ \nu\epsilon\upsilon\rho\acute{\omega}\nu\omega\nu = \frac{I - F + 2Z}{S} + 1 \quad (2)$$

2.1.3.2.6 Διαμοιρασμός παραμέτρων (parameter sharing)

Τα συνελικτικά δίκτυα χρησιμοποιούν ένα σχήμα διαμοιρασμού παραμέτρων έτσι ώστε να μειώσουν το πλήθος των παραμέτρων. Με αυτό τον τρόπο η εκπαίδευση του νευρωνικού δικτύου γίνεται πιο γρήγορα, καθώς χρησιμοποιούνται λιγότερες πηγές δεδομένων εκπαίδευσης, χωρίς αυτό να επηρεάζει σημαντικά την τελική επίδοση του μοντέλου.

2.1.3.2.7 Σύνοψη συνελικτικού επιπέδου

Το συνελικτικό επίπεδο έχει τις εξής προδιαγραφές:

- Είσοδος διαστάσεων: $\mathbf{I_1}$ x $\mathbf{J_1}$ x $\mathbf{W_1}$
- Υπερπαράμετροι:
    - Πλήθος φίλτρων **K**
    - Χωρικό μέγεθος φίλτρου **F**
    - Βήμα **S**
    - Αριθμός του γεμίσματος με μηδενικά **Z**
- Έξοδος διαστάσεων: $\mathbf{I_2}$ x $\mathbf{J_2}$ x $\mathbf{W_2}$, όπου:
    - $I_2 = \frac{I_1 - F + 2Z}{S} + 1$
    - $J_2 = \frac{J_1 - F + 2Z}{S} + 1$
    - $W_2 = K$

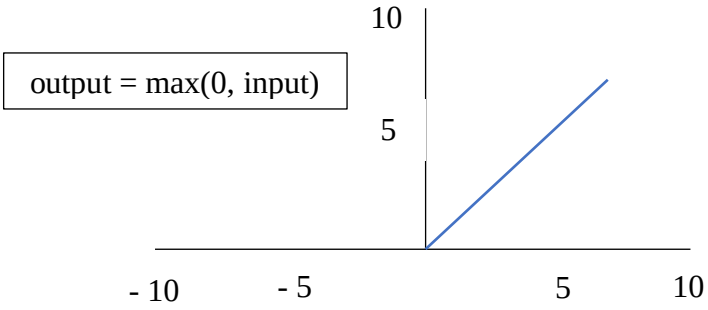
2.1.3.3 Επίπεδο ReLU

Για να οριστεί με σαφήνεια το επίπεδο ReLU είναι σημαντικό να γίνει μία αναφορά στη σημασιολογία ορισμένων όρων. Στο l-οστό επίπεδο του δικτύου η είσοδος συμβολίζεται ως x^l. Η είσοδος είναι τριών διαστάσεων επομένως απαιτείται μία τριάδα (i^l , j^l, k^l) τιμών έτσι ώστε να καθοριστεί ένα συγκεκριμένο σημείο της.

Το επίπεδο επεξεργασίας ReLU (Rectified Linear Unit) δεν επηρεάζει το μέγεθος της εισόδου, πράγμα που σημαίνει ότι το x^l (είσοδος) και το y (έξοδος) έχουν το ίδιο μέγεθος. Στην πράξη, η ReLU είναι μία αποκοπή που εφαρμόζεται σε κάθε στοιχείο της εισόδου ως εξής:

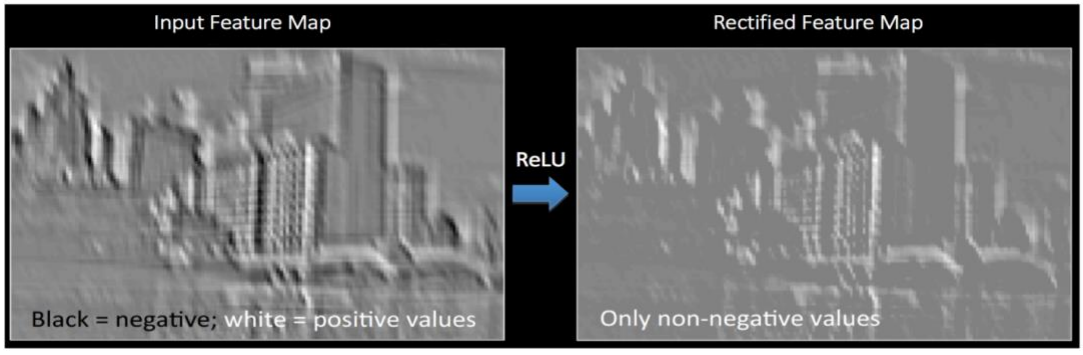
$$y_{i,j,k} = \max(0, x^l_{i,j,k}) \qquad (3)$$

Στο επίπεδο αυτό δεν υπάρχουν παράμετροι και συνεπώς δεν υφίσταται κάποια διαδικασία εκπαίδευσης. Σκοπός του επιπέδου ReLU είναι η αύξηση της μη-γραμμικότητας του CNN, καθώς τα περισσότερα πραγματικά δεδομένα με τα οποία εκπαιδεύονται τα νευρωνικά δίκτυα είναι μη-γραμμικά.



Σχήμα 2.9: Συνάρτηση ReLU

Η είσοδος του επιπέδου ReLU μπορεί να έχει θετική, μηδενική ή αρνητική τιμή. Η συνάρτηση ReLU μετατρέπει όλες τις μη-θετικές τιμές σε μηδενικές τιμές για να αναδείξει μη-γραμμικές συσχετίσεις στα δεδομένα εισόδου. Για παράδειγμα, αν η είσοδος έχει θετικές τιμές σε μία περιοχή της εικόνας που παρουσιάζει ένα συγκεκριμένο μοτίβο (πχ. ανθρώπινο πρόσωπο), τότε μπορεί να λαμβάνει αρνητικές ή μηδενικές τιμές σε άλλες περιοχές που δεν έχουν αυτό το μοτίβο. Σε αυτή την περίπτωση, το επίπεδο ReLU θα θέσει όλες τις αρνητικές τιμές ίσες με το μηδέν, με αποτέλεσμα να ενεργοποιείται μόνο η έξοδος σε περιοχές που παρουσιάζουν το επιθυμητό μοτίβο. Η λειτουργία της συνάρτησης ReLU γίνεται πιο κατανοητή στο σχήμα 2.10.



Σχήμα 2.10: Λειτουργία της συνάρτησης ReLU ([20])

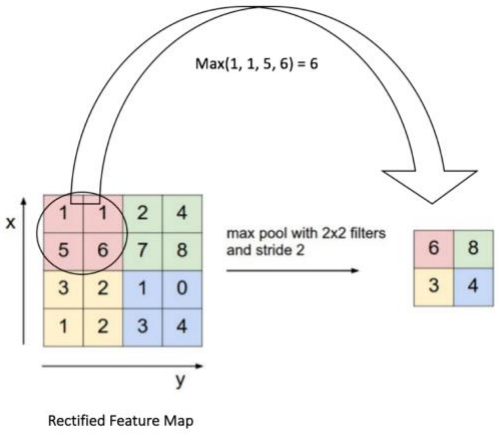
Η ReLU θεωρείται η καλύτερη συνάρτηση ενεργοποίησης επειδή έχει αποδειχθεί ότι λειτουργεί σε πολλές διαφορετικές καταστάσεις. Επειδή η κλίση της είναι είτε μηδενική είτε σταθερή, η συνάρτηση δεν παρουσιάζει συχνά το πρόβλημα της υπερβολικής αύξησης της κλίσης. Ποικίλες εφαρμογές έχουν δείξει ότι το νευρωνικό δίκτυο εκπαιδεύεται καλύτερα όταν χρησιμοποιείται η ReLU και όχι κάποια άλλη συνάρτηση ενεργοποίησης όπως η σιγμοειδής (sigmoid).

2.1.3.4 Συγκεντρωτικό επίπεδο (Pooling layer)

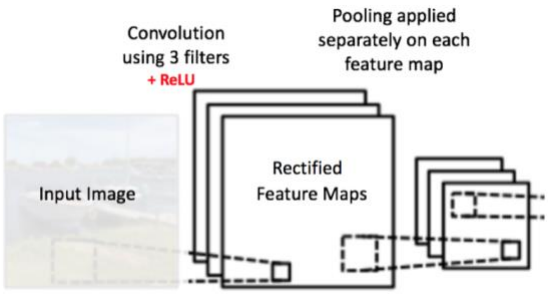
Τα συγκεντρωτικά επίπεδα συνήθως εισάγονται μεταξύ διαδοχικών συνελικτικών επιπέδων. Στόχος είναι η προοδευτική μείωση του χωρικού μεγέθους (πλάτος και ύψος) της αναπαράστασης δεδομένων και των παραμέτρων του δικτύου, έτσι ώστε να ελεγχθεί η υπερπροσαρμογή (overfitting) του μοντέλου. Επομένως τα συγκεντρωτικά επίπεδα μειώνουν τις διαστάσεις κάθε χάρτη χαρακτηριστικών αλλά διατηρεί τις πιο σημαντικές πληροφορίες. Είναι σημαντικό να σημειωθεί ότι αυτή η λειτουργία δεν επηρεάζει τη διάσταση του βάθους.

Σε αυτό το επίπεδο, ορίζεται ένας πίνακας-παράθυρο (πχ. διαστάσεων 2x2) που μετακινείται πάνω στο χάρτη χαρακτηριστικών και διατηρεί μόνο το μεγαλύτερο στοιχείο (max pooling) από την εκάστοτε περιοχή. Αντί να επιλέγεται το μεγαλύτερο στοιχείο θα μπορούσε να υπολογιζόταν ο μέσος όρος (average pooling) ή το άθροισμα (sum pooling) των στοιχείων της περιοχής. Αξίζει να σημειωθεί ότι στην πράξη έχει αποδειχθεί πως η επιλογή του μέγιστου στοιχείου (max pooling) είναι πιο αποδοτική τεχνική.

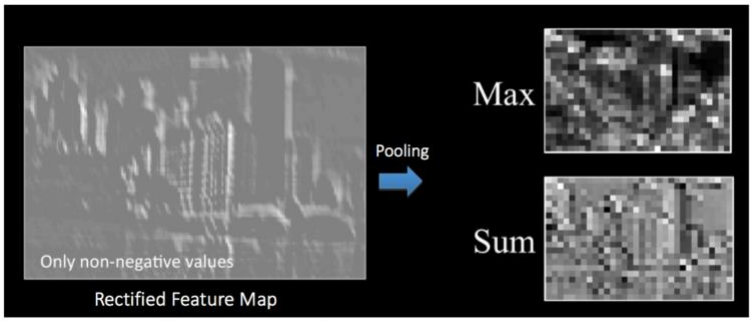
Αυτά τα στρώματα εκτελούν λειτουργίες υποδειγματοληψίας κατά μήκος της χωρικής διάστασης των δεδομένων εισόδου. Αυτό σημαίνει ότι εάν η εικόνα εισόδου είχε διαστάσεις 32x32, τότε η εικόνα εξόδου θα ήταν μικρότερη σε πλάτος και ύψος (πχ. 16x16). Η πιο κοινή ρύθμιση για ένα επίπεδο συγκέντρωσης είναι η εφαρμογή φίλτρων 2x2 με βήμα (stride) 2. Αυτή η πρακτική θα μειώσει τις χωρικές διαστάσεις (πλάτος ύψος) κατά ένα συντελεστή δύο. Η λειτουργία της δειγματοληψίας θα οδηγήσει στην απόρριψη του 75% των ενεργοποιήσεων (στην περίπτωση του max pooling).



Σχήμα 2.11: Παράδειγμα εφαρμογής της λειτουργίας max pooling σε χάρτη χαρακτηριστικών που προέκυψε μετά από συνέλιξη και εφαρμογή της συνάρτησης ReLU. Το φίλτρο έχει διαστάσεις 2x2 και το βήμα (stride) έχει τιμή 2. Η είσοδος έχει διαστάσεις 4x4 και μετά από την εφαρμογή της συγκεντρωτικής λειτουργίας η έξοδος έχει διαστάσεις 2x2. ([20])



Σχήμα 2.12: Αφαιρετική αναπαράσταση της συγκεντρωτικής λειτουργίας. Αξίζει να σημειωθεί πως εφαρμόζεται σε κάθε χάρτη χαρακτηριστικών ξεχωριστά. ([20])



Σχήμα 2.13: Διαφορετικοί τύποι συγκεντρωτικής λειτουργίας (max and sum pooling). Στο max pooling παρατηρείται ότι εξάγεται πιο έντονη πληροφορία συγκριτικά με το sum pooling. ([20])

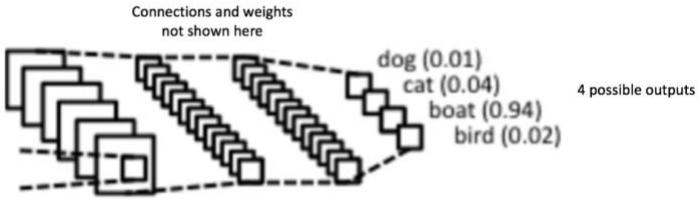
Το συγκεντρωτικό επίπεδο έχει τις εξής προδιαγραφές:

- Είσοδος διαστάσεων: $\mathbf{I_1}$ x $\mathbf{J_1}$ x $\mathbf{W_1}$
- Υπερπαράμετροι:
  - Χωρικό μέγεθος φίλτρου **F**
  - Βήμα **S**
- Έξοδος διαστάσεων: $\mathbf{I_2}$ x $\mathbf{J_2}$ x $\mathbf{W_2}$, όπου:
  - $I_2 = \frac{I_1 - F}{S} + 1$
  - $J_2 = \frac{J_1 - F}{S} + 1$
  - $W_2 = W_1$

2.1.3.5 Πλήρως συνδεδεμένο επίπεδο (fully connected layer)

Το πλήρως συνδεδεμένο επίπεδο είναι μία παραδοσιακή αρχιτεκτονική πολλών επιπέδων με νευρώνες, η οποία χρησιμοποιεί μία συνάρτηση ενεργοποίησης (συνήθως τη softmax) στην έξοδό της. Κάθε επίπεδο που ανήκει σε αυτή την αρχιτεκτονική έχει την ιδιότητα πως κάθε νευρώνας που περιλαμβάνει συνδέεται με όλους τους νευρώνες του προηγούμενου επιπέδου.

Η έξοδος των συνελικτικών και συγκεντρωτικών επιπέδων αναπαριστά χαρακτηριστικά υψηλών στρωμάτων. Στη βασική περίπτωση, που το πρόβλημα ανήκει στην κατηγορία της ταξινόμησης (classification), ο σκοπός του πλήρως συνδεδεμένου επιπέδου είναι να χρησιμοποιήσει αυτά τα χαρακτηριστικά έτσι ώστε να ταξινομήσει την εικόνα εισόδου σε διάφορες κλάσεις, βασιζόμενο στο σύνολο δεδομένων που χρησιμοποιήθηκαν για την εκπαίδευση του μοντέλου.



Σχήμα 2.14: Παράδειγμα ενός πλήρως συνδεδεμένου δικτύου. Η έξοδος του δικτύου είναι 4 τιμές που δηλώνουν την πιθανότητα να ανήκει η εικόνα εισόδου σε κάθε μία από τις 4 κλάσεις (σκύλος, γάτα, βάρκα, πουλί). Το μοντέλο εξάγει το συμπέρασμα ότι με μεγάλη πιθανότητα (0.94) στην εικόνα υπάρχει μία βάρκα.

Εκτός από την ταξινόμηση, η προσθήκη ενός πλήρως συνδεδεμένου στρώματος είναι επίσης ένας (συνήθως) «φτηνός» τρόπος εκμάθησης μη γραμμικών συνδυασμών αυτών των χαρακτηριστικών. Τα περισσότερα χαρακτηριστικά από τα στρώματα συνένωσης και συγκέντρωσης μπορεί να είναι καλά για την εργασία ταξινόμησης, αλλά οι συνδυασμοί αυτών των χαρακτηριστικών μπορεί να είναι ακόμη καλύτεροι.

Το άθροισμα των πιθανών εξόδων από το πλήρως συνδεδεμένο επίπεδο είναι 1, καθώς αποτελούν το άθροισμα όλων των πιθανοτήτων. Αυτό επιτυγχάνεται με τη χρήση της softmax ως της συνάρτησης ενεργοποίησης στο επίπεδο εξόδου του πλήρως συνδεδεμένου στρώματος. Η συνάρτηση softmax δέχεται σαν είσοδο ένα διάνυσμα από τυχαίες πραγματικές τιμές και τις αντιστοιχίζει σε ένα διάνυσμα τιμών από 0 έως 1. Το άθροισμα των στοιχείων του διανύσματος εξόδου ισούται με 1.

Το πλήρως συνδεδεμένο επίπεδο μπορεί να χρησιμοποιηθεί και σε προβλήματα παλινδρόμησης (regression), αρκεί στο τελευταίο επίπεδο να υπάρχει ένας μόνο νευρώνας του οποίου η έξοδος να αποτελεί την τελική έξοδο του μοντέλου.

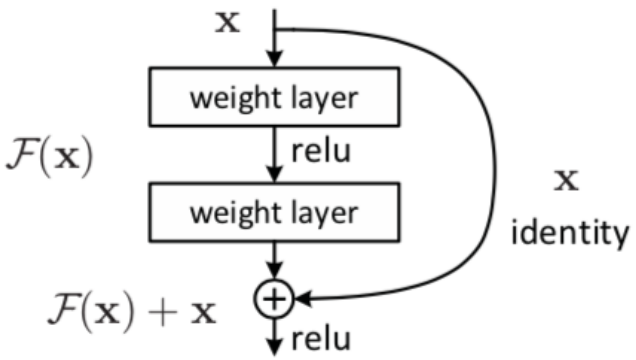
2.1.4 Μοντέλο ResNet

Το ‘ResNet’ ([6]) είναι ένα σύντομο όνομα για το ‘Residual Network’. Όπως υποδηλώνει το όνομα του δικτύου, η νέα ορολογία που εισάγει είναι η «υπολειμματική» μάθηση (residual learning). Τα βαθιά συνελικτικά δίκτυα έχουν οδηγήσει σε μία σειρά ανακαλύψεων για την ταξινόμηση των εικόνων. Έτσι, με την πάροδο του χρόνου υπάρχει μία τάση να αυξάνεται το βάθος των δικτύων έτσι ώστε να επιλύουν πιο σύνθετα προβλήματα. Όμως, όσο αυξάνεται το βάθος του δικτύου η εκπαίδευση καθίσταται πιο δύσκολη. Η «υπολειμματική» μάθηση προσπαθεί να λύσει αυτό το πρόβλημα.

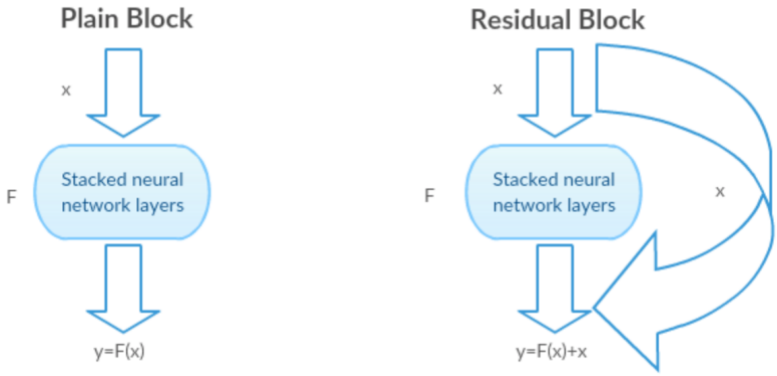
Σε γενικές γραμμές, σε ένα βαθύ συνελικτικό νευρωνικό δίκτυο, πολλά στρώματα (layers) στοιβάζονται και εκπαιδεύονται. Το δίκτυο μαθαίνει χαμηλού, μεσαίου και υψηλού επιπέδου χαρακτηριστικά στο τέλος των επιπέδων του. Στην «υπολειμματική» μάθηση πέρα από την εκμάθηση των τυπικών χαρακτηριστικών, το

δίκτυο προσπαθεί να μάθει και κάποια υπολείμματα (residuals). Το υπόλειμμα μπορεί να γίνει εύκολα κατανοητό ως η αφαίρεση του χαρακτηριστικού που εξάχθηκε από την είσοδο ενός στρώματος.

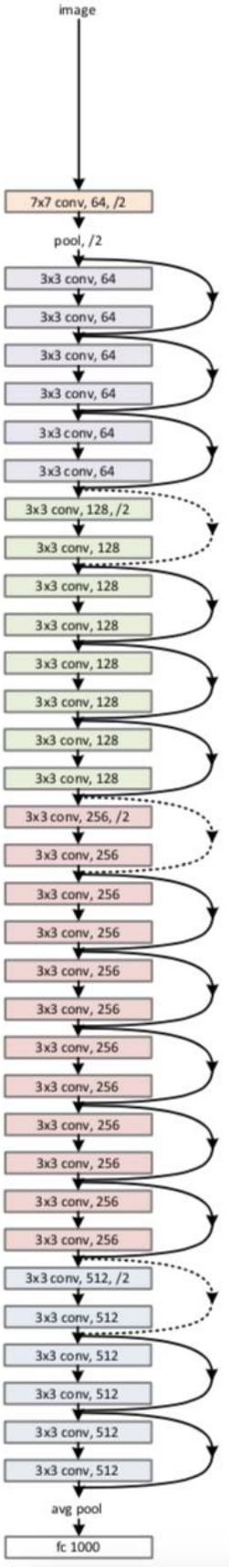
Το 'ResNet' εφαρμόζει αυτή τη μορφή μάθησης χρησιμοποιώντας συνδέσεις συντόμευσης, συνδέοντας άμεσα την είσοδο του ν-οστού επιπέδου με κάποιο επόμενο επίπεδο. Έχει αποδειχθεί ότι η εφαρμογή αυτής της μορφής μάθησης στα δίκτυα καθιστά ευκολότερη την εκπαίδευσή τους και επιλύει το πρόβλημα της περιορισμένης ακρίβειας.



Σχήμα 2.15: Περιγραφή της υπολειμματικής μάθησης (residual learning). ([6])



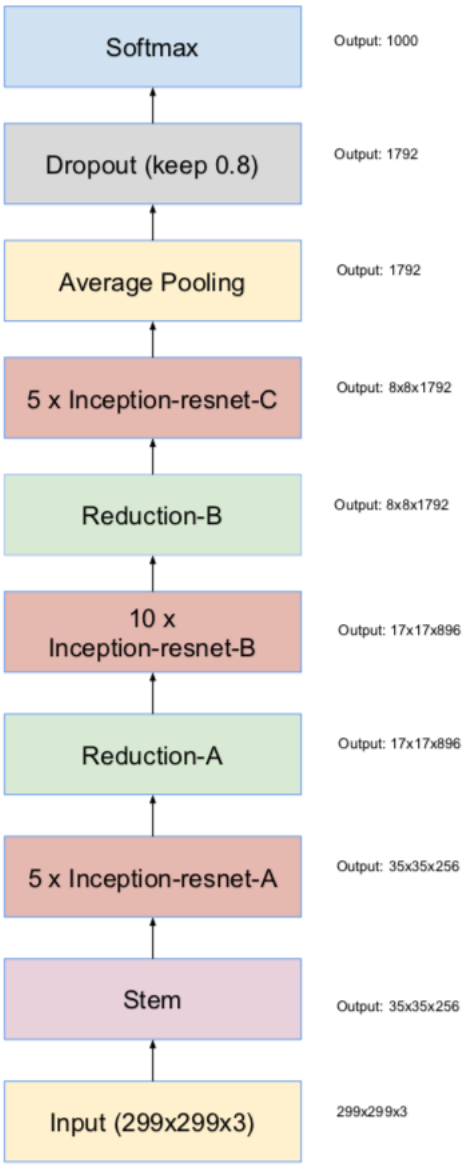
Σχήμα 2.16: Σύγκριση μεταξύ απλού και «υπολειμματικού» επιπέδου



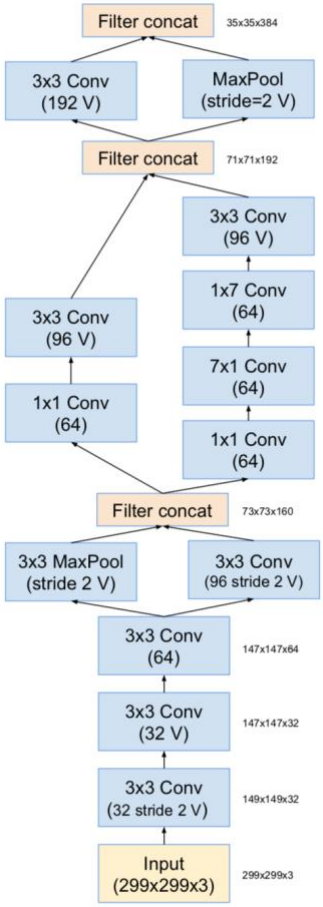
Σχήμα 2.17: Γενική αναπαράσταση του δικτύου ResNet 34 επιπέδων. Αν αντικατασταθεί κάθε διεπίπεδο μπλοκ που εφαρμόζεται η «παράκαμψη» (residual learning) με ένα τριεπίπεδο μπλοκ τότε προκύπτει το ResNet 50 επιπέδων. ([6])

2.1.5 Μοντέλο Inception-ResNet-V2

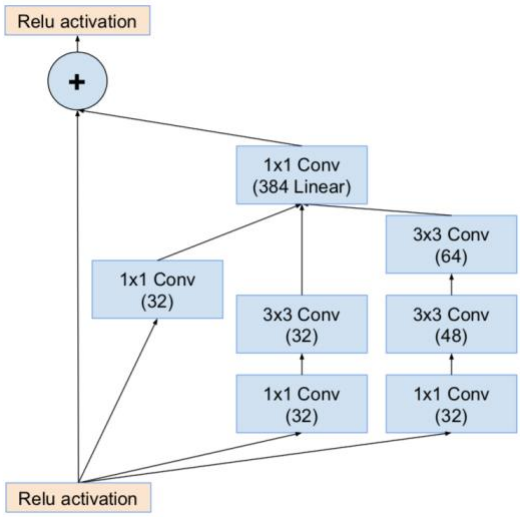
Το μοντέλο 'Inception-ResNet-V2' ([7]) είναι υβριδικό καθώς συνδυάζει στοιχεία της αρχιτεκτονικής 'Inception' ([9]) που εισήγαγε η Google αλλά και στοιχεία υπολειμματικής μάθησης (residual learning) ([6]), η οποία εφαρμόστηκε για πρώτη φορά στο δίκτυο 'ResNet'. Σε ένα τυπικό στρώμα συνελικτικού νευρωνικού δικτύου επιλέγεται αν θα χρησιμοποιηθούν φίλτρα διαστάσεων 3x3 ή 5x5 ή αν θα προστεθεί ένα μέγιστο συγκεντρωτικό επίπεδο (max pooling layer). Η κεντρική ιδέα της αρχιτεκτονικής 'Inception' είναι να μη γίνεται αυτή η επιλογή φίλτρων ή συγκεντρωτικού επιπέδου εκ των προτέρων και να δοκιμάζονται όλα. Στην πορεία το ίδιο το δίκτυο θα εκπαιδευτεί και θα μάθει τις βέλτιστες παραμέτρους για τη μείωση της συνάρτησης κόστους. Ακολουθούν σχήματα που περιγράφουν όλη την αρχιτεκτονική του μοντέλου.



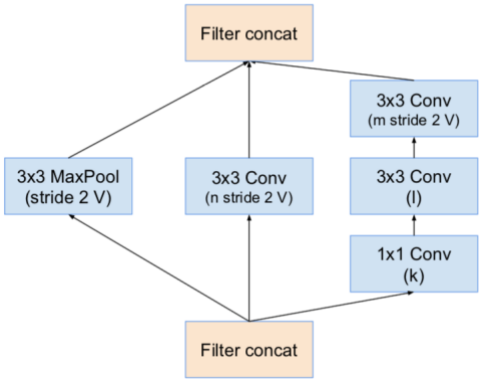
Σχήμα 2.18: Γενικό σχήμα αρχιτεκτονικής του μοντέλου Inception-ResNet-V2. ([7])



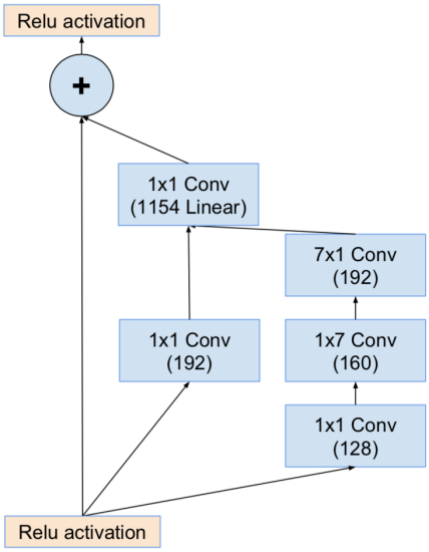
Σχήμα 2.19: Περιγραφή του επιπέδου 'Stem' που παρουσιάζεται στο σχήμα 2.16. ([7])



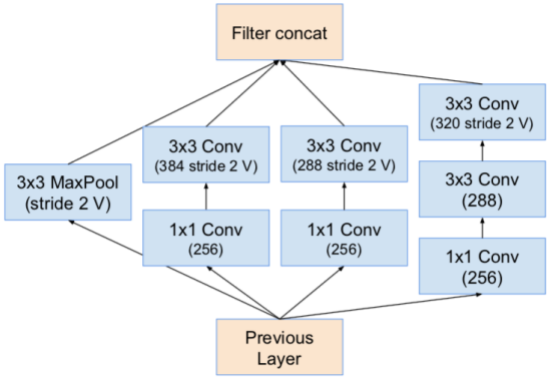
Σχήμα 2.20: Περιγραφή του επιπέδου 'Inception-ResNet-A' που παρουσιάζεται στο σχήμα 2.16. ([7])



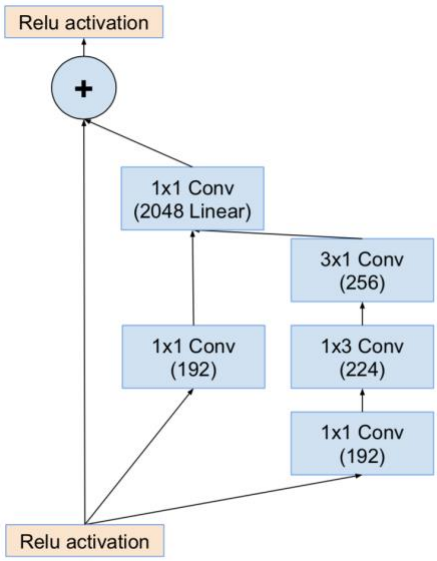
Σχήμα 2.21: Περιγραφή του επιπέδου 'Reduction-A' που παρουσιάζεται στο σχήμα 2.16, όπου k = 256, l = 256, m = 384 και n = 384. ([7])



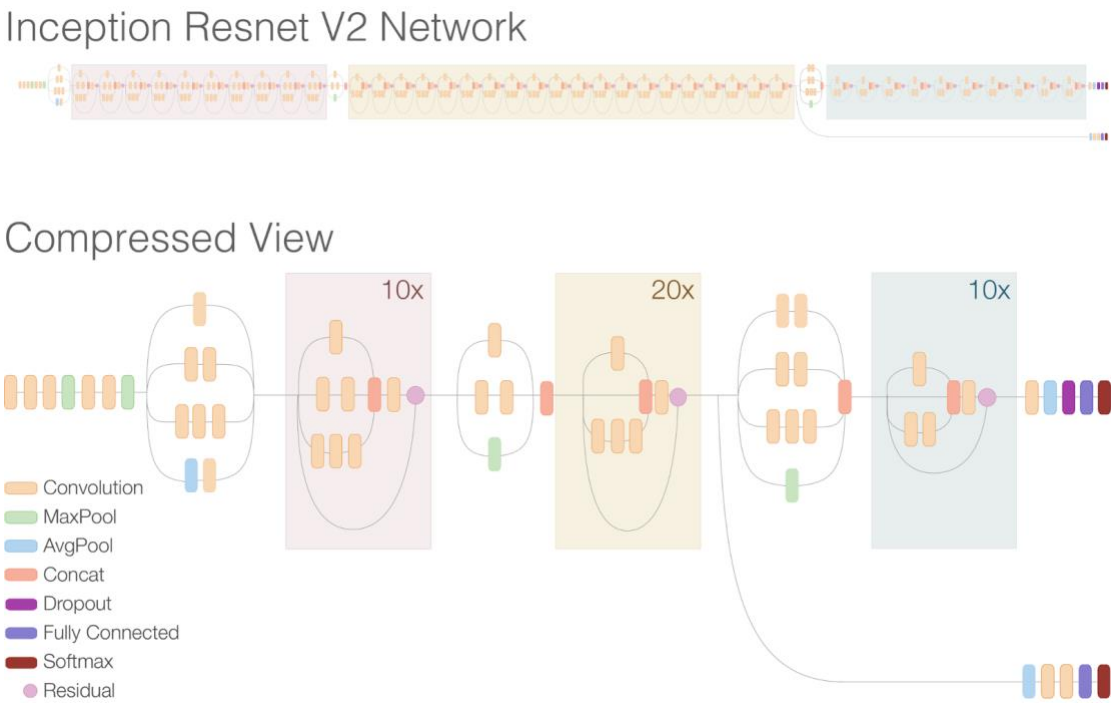
Σχήμα 2.22: Περιγραφή του επιπέδου 'Inception-ResNet-B' που παρουσιάζεται στο σχήμα 2.16. ([7])



Σχήμα 2.23: Περιγραφή του επιπέδου 'Reduction-B' που παρουσιάζεται στο σχήμα 2.16. ([7])



Σχήμα 2.24: Περιγραφή του επιπέδου 'Inception-ResNet-C' που παρουσιάζεται στο σχήμα 2.16. ([7])



Σχήμα 2.25: Παρουσίαση αρχιτεκτονικής μοντέλου Inception-ResNet-V2 με διαφορετικό τρόπο, έτσι ώστε να φαίνονται και άλλα ενδιάμεσα επίπεδα. ([25])

2.1.6 Άλλα γνωστά συνελικτικά δίκτυα

- **LeNet (**[20]) (1990s): Αποτελεί την πρώτη επιτυχημένη εφαρμογή συνελικτικών δικτύων που αναπτύχθηκε από τον Yann LeCun την δεκαετία του 1990. Η συγκεκριμένη αρχιτεκτονική χρησιμοποιήθηκε κυρίως για αναγνώριση κωδικών, ψηφίων κ.τ.λ.
- **AlexNet (**[20]) (2012): Το συγκεκριμένο δίκτυο ήταν το πρώτο το οποίο έκανε τα συνελικτικά δίκτυα διάσημα στο χώρο της όρασης υπολογιστών (computer vision). Το δίκτυο αυτό είχε μία πολύ παρόμοια αρχιτεκτονική με αυτή του LeNet, ωστόσο, ήταν βαθύτερο, μεγαλύτερο και είχε πολλά συνελικτικά επίπεδα, στοιβαγμένα το ένα πάνω στο άλλο, γεγονός που είχε αποτελέσει μια πρωτοποριακή τεχνική.
- **ZF Net (**[20]) (2013): Το δίκτυο αυτό αποτελεί μια βελτίωση του AlexNet, διορθώνοντας κάποιες υπερπαραμέτρους της αρχιτεκτονικής. Πιο συγκεκριμένα, επεκτείνανε το μέγεθος του μεσαίου συνελικτικού επιπέδου, ενώ παράλληλα έκαναν το βήμα και το μέγεθος φίλτρου του πρώτου επιπέδου μικρότερα. Το συγκεκριμένο δίκτυο απέσπασε την πρώτη θέση στον διαγωνισμό ILSVRC 2013.
- **VGGNet (**[20]) (2014): Το συγκεκριμένο δίκτυο, αν και απέσπασε την δεύτερη θέση στον διαγωνισμό ILSVRC 2014, ήταν αντίστοιχα αποτελεσματικό με το GoogleNet. Η βασική συνεισφορά του ήταν στο γεγονός ότι το βάθος ενός δικτύου είναι ένα σημαντικό συστατικό για την καλή απόδοση.
- **DenseNet (**[20]) (2016): Δημοσιευμένο από τον Gao Huang, το DenseNet έχει κάθε στρώμα απευθείας συνδεδεμένο με κάθε άλλο στρώμα που έπεται. Το DenseNet έχει αποδειχθεί ότι επιτυγχάνει σημαντικές βελτιώσεις σε σχέση με προηγούμενες υπερσύγχρονες αρχιτεκτονικές σε πέντε εξαιρετικά ανταγωνιστικές εργασίες συγκριτικής αξιολόγησης αντικειμένων.

2.2 Συνάρτηση κόστους (loss function)

Οι συναρτήσεις κόστους ποσοτικοποιούν πόσο αποδοτικά έχει εκπαιδευτεί το νευρωνικό δίκτυο, χρησιμοποιώντας τα δεδομένα εκπαίδευσης. Οι συναρτήσεις κόστους εκφράζουν μία μέτρηση με βάση το σφάλμα που παρατηρείται στις προβλέψεις του δικτύου. Ο μέσος όρος των σφαλμάτων σε ολόκληρο το σύνολο δεδομένων εκφράζει πόσο κοντά είναι το μοντέλο σε ένα ιδανικό μοντέλο που δεν κάνει ποτέ λάθος. Η αναζήτηση αυτής της ιδανικής κατάστασης ισοδυναμεί με την εύρεση των παραμέτρων που θα ελαχιστοποιήσουν τη συνάρτηση κόστους. Με βάση αυτή τη λογική, οι συναρτήσεις κόστους χρησιμοποιούνται έτσι ώστε η αποδοτική εκπαίδευση του νευρωνικού δικτύου να ανάγεται σε ένα πρόβλημα βελτιστοποίησης.

Υπάρχουν πολλές συναρτήσεις κόστους που χρησιμοποιούνται στα προβλήματα βελτιστοποίησης. Για προβλήματα παλινδρόμησης (regression) η πιο κοινή συνάρτηση κόστους είναι αυτή του μέσου τετραγωνικού σφάλματος (mean squared error – mse), ενώ σε προβλήματα κατηγοριοποίησης (classification) συνήθως χρησιμοποιείται η συνάρτηση εντροπίας (cross entropy).

$$J(W,b)_{MSE} = \frac{1}{N} * \sum_{i=1}^{N} \frac{1}{M} * \sum_{j=1}^{M} \left(y'_{i,j} - y_{i,j}\right)^2 \qquad (4)$$

$$J(W,b)_{cross-entropy} = -\sum_{i=1}^{N}\sum_{j=1}^{M} (y_{i,j} * \log(y'_{i,j})) \qquad (5)$$

Όπου:

- J(W, b): συνάρτηση κόστους
- W: βάρη του νευρωνικού δικτύου
- b: σταθεροί όροι του νευρωνικού δικτύου (biases)
- N: πλήθος δεδομένων
- M: πλήθος εξόδων του δικτύου
- y': έξοδος του νευρωνικού δικτύου
- y: πραγματική έξοδος - στόχος (ground truth)
- i: δείκτης, δηλώνει το i-οστο δείγμα (sample)
- j: δείκτης, δηλώνει το j-οστο χαρακτηριστικό (feature)

2.3 Πρόβλημα βελτιστοποίησης (optimization)

Η διαδικασία προσαρμογής των βαρών για την εξαγωγή ακριβέστερων προβλέψεων είναι γνωστή ως βελτιστοποίηση παραμέτρων. Αφαιρετικά, η διαδικασία αυτή είναι μία μέθοδο κατά την οποία δημιουργείται μία υπόθεση, η υπόθεση συγκρίνεται με την πραγματικότητα και με βάση τη σύγκριση αυτή η υπόθεση βελτιώνεται ή αντικαθίσταται με σκοπό να προσεγγίσει περισσότερο την πραγματικότητα.

Κάθε σύνολο βαρών του νευρωνικού δικτύου αντιπροσωπεύει μία συγκεκριμένη υπόθεση για το τι σημαίνουν τα δεδομένα εισόδου. Τα βάρη αντιπροσωπεύουν εικασίες σχετικά με τις συσχετίσεις της εισόδου και της εξόδου που επιδιώκουν να υποθέσουν.

Όλα τα πιθανά βάρη και οι συνδυασμοί τους μπορούν να περιγραφούν ως ο υποθετικός χώρος αυτού του προβλήματος. Η προσπάθειά να διαμορφωθεί η καλύτερη υπόθεση είναι θέμα αναζήτησης μέσα από αυτόν τον χώρο και γίνεται με χρήση συναρτήσεων κόστους και αλγορίθμων βελτιστοποίησης. Όσο περισσότερες είναι οι παράμετροι εισόδου, τόσο μεγαλύτερος είναι ο χώρος αναζήτησης του προβλήματος. Μεγάλο μέρος της διαδικασίας μάθησης αποτελεί η απόφαση για το ποιες παράμετροι είναι σημαντικές και ποιες όχι.

2.3.1 Αλγόριθμος μείωσης κλίσης (gradient descent)

Στη μέθοδο της πλέον απότομης κατάβασης (steepest descent) ([1]), οι διαδοχικές προσαρμογές που εφαρμόζονται στο διάνυσμα βαρών **w** είναι προς την κατεύθυνση της πλέον απότομης κατάβασης – δηλαδή, σε κατεύθυνση αντίθετη από

το διάνυσμα κλίσης $\nabla J(w)$, όπου J το διάνυσμα κόστους που είναι συνάρτηση των βαρών. Για ευκολία παρουσίασης:

$$g = \nabla J(w) \qquad (6)$$

Κατά συνέπεια ο αλγόριθμος της πλέον απότομης κατάβασης περιγράφεται φορμαλιστικά από τη σχέση:

$$w(n+1) = w(n) - \eta * g(n) \qquad (7)$$

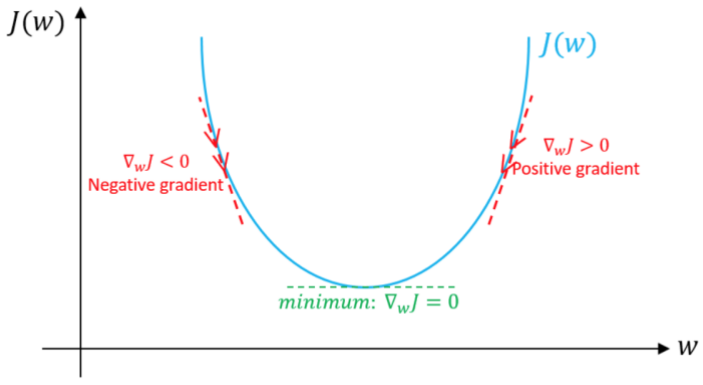
Όπου:

- η: θετική σταθερά που αποκαλείται ρυθμός μάθησης (learning rate)
- n: αριθμός που δηλώνει τον αύξοντα αριθμό επαναλήψεων που τρέχει ο αλγόριθμος
- g(n): το διάνυσμα κλίσης που υπολογίζεται στο σημείο w(n)
- w(n): διάνυσμα βαρών στην n-οστη επανάληψη εκτέλεσης του αλγορίθμου

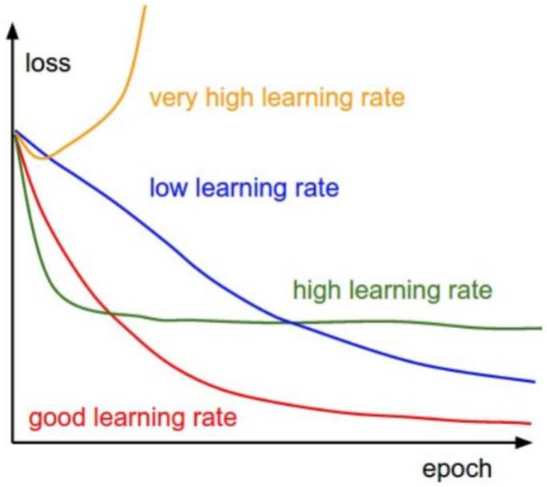
Κατά τη μετάβαση από την επανάληψη n στην n + 1, ο αλγόριθμος εφαρμόζει τη διόρθωση:

$$\Delta w(n) = w(n+1) - w(n) = -\eta * g(n) \qquad (8)$$

Ο αλγόριθμος μείωσης κλίσης συγκλίνει στη βέλτιστη λύση **w*** αργά. Επιπλέον η παράμετρος του ρυθμού μάθησης η έχει βαθιά επίδραση στη συμπεριφορά σύγκλισης του αλγορίθμου. Μεγάλες τιμές του ρυθμού μάθησης μπορεί να οδηγήσουν σε ταλαντωτική συμπεριφορά, ενώ πολύ μικρές τιμές έχουν σαν αποτέλεσμα ο αλγόριθμος να αργεί πολύ να συγκλίνει. Επομένως, υπάρχει ένα εύρος τιμών που αυξάνει την πιθανότητα για γρήγορη και επιτυχής σύγκλιση του αλγορίθμου.



Σχήμα 2.26: Γραφική παράσταση που δείχνει τη λειτουργία του αλγόριθμου μείωσης κλίσης (gradient descent).



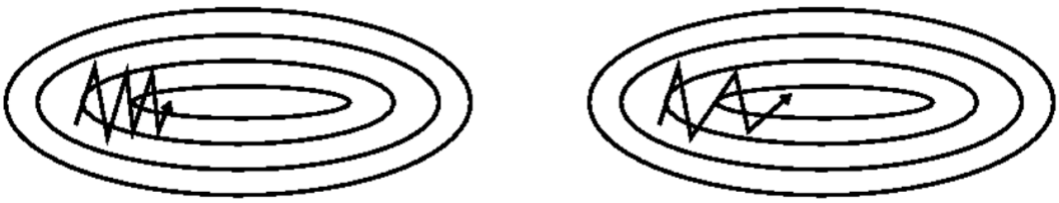
Σχήμα 2.27: Γραφική παράσταση που δείχνει τη συμπεριφορά του αλγορίθμου μείωσης κλίσης για διαφορετικές τιμές της παραμέτρου ρυθμού μάθησης.

2.3.2 Αλγόριθμος στοχαστικής μείωσης κλίσης (stochastic gradient descent - SGD)

Στον αλγόριθμο μείωσης κλίσης η συνολική απώλεια υπολογιζόταν με βάση όλα τα παραδείγματα του συνόλου δεδομένων εκπαίδευσης. Ο αλγόριθμος στοχαστικής μείωσης κλίσης αποτελεί μία υποπερίπτωση του γενικού αλγορίθμου, όπου ο υπολογισμός της κλίσης και η ενημέρωση των βαρών γίνεται με χρήση ενός ή λίγων (mini batch SGD) δειγμάτων εκπαίδευσης. Έχει αποδειχθεί ότι αυτή η παραλλαγή, στην πράξη, επιταχύνει την εκπαίδευση του νευρωνικού δικτύου.

2.3.3 Ορμή (momentum)

Ο αλγόριθμος στοχαστικής μείωσης κλήσης (SGD) έχει πρόβλημα πλοήγησης στις «χαράδρες», δηλαδή σε περιοχές όπου η επιφάνεια καμπυλώνεται πολύ πιο απότομα σε μία διάσταση από ότι σε μία άλλη, οι οποίες είναι κοινές γύρω από το τοπικό βέλτιστο (local optima). Σε αυτές τις περιπτώσεις, ο 'SGD' ταλαντώνεται στις πλαγιές της χαράδρας ενώ παράλληλα κάνει αργή πρόοδο στην προσπάθεια προσέγγισης του τοπικού βέλτιστου.



Σχήμα 2.28: Αριστερά φαίνεται η πρόοδος του 'SGD' χωρίς ορμή (momentum), ενώ δεξιά με ορμή. ([19])

Η ορμή (momentum) ([12]) είναι μια μέθοδος που βοηθά στην επιτάχυνση του 'SGD' στη σχετική κατεύθυνση και μειώνει τις ταλαντώσεις, όπως φαίνεται στο σχήμα 2.26. Αυτό επιτυγχάνεται προσθέτοντας ένα κλάσμα γ του διανύσματος ενημέρωσης του προηγούμενου βήματος χρόνου στο τρέχον διάνυσμα ενημέρωσης:

$$v_t = \gamma * v_{t-1} + \eta * \nabla J(w) \qquad (9)$$

$$w = w - v_t \qquad (10)$$

Ο όρος γ συνήθως ορίζεται στην τιμή 0.9. Ουσιαστικά όταν χρησιμοποιείται η ορμή, είναι σα να πιέζεται μια σφαίρα κάτω σε ένα λόφο. Η σφαίρα συσσωρεύει την ορμή, καθώς κυλάει προς τα κάτω, γρηγορότερα και ταχύτερα στο δρόμο (μέχρι να φτάσει στην τελική της ταχύτητας αν υπάρχει αντίσταση αέρα, δηλαδή $\gamma < 1$). Το ίδιο συμβαίνει με τις ενημερώσεις των παραμέτρων. Η ορμή αυξάνεται για τις διαστάσεις των οποίων οι κλίσεις δείχνουν προς τις ίδιες κατευθύνσεις και μειώνουν τις ενημερώσεις για τις διαστάσεις των οποίων οι κλίσεις αλλάζουν κατευθύνσεις. Ως αποτέλεσμα, επιτυγχάνεται ταχύτερη σύγκλιση και μειωμένη ταλάντωση.

2.3.4 Adam

Ο αλγόριθμος βελτιστοποίησης 'Adam' ([11]) είναι μια επέκταση του 'SGD' και χρησιμοποιείται για την ενημέρωση των βαρών του δικτύου με βάση τα δεδομένα εκπαίδευσης και παρουσιάστηκε από τους Diederik Kingma και Jimmy Ba. Υπολογίζει τα ποσοστά προσαρμοστικής εκμάθησης για κάθε παράμετρο. Εκτός από την αποθήκευση του εκθετικά φθίνοντος μέσου όρου προηγούμενων τετραγωνικών κλίσεων, όπως κάνουν οι αλγόριθμοι 'Adadelta' και 'RMSprop', ο 'Adam' κρατά τον εκθετικά φθίνοντα μέσο όρο προηγούμενων κλίσεων, όπως ο αλγόριθμος 'Momentum'. Ο 'Adam' είναι ευρέως διαδεδομένος στον τομέα της βαθιάς μηχανικής μάθησης γιατί επιτυγχάνει καλά αποτελέσματα γρήγορα. Ενώ η ορμή (momentum) μπορεί να θεωρηθεί σαν μια μπάλα που τρέχει σε μια πλαγιά, ο 'Adam' συμπεριφέρεται σαν μια βαριά μπάλα με τριβή, η οποία προτιμά τα επίπεδα ελάχιστα στην επιφάνεια σφάλματος. Ο υπολογισμός των βαθμών απόσβεσης του παρελθόντος m_t και των τετραγωνικών κλίσεων v_t γίνεται ως εξής:

$$m_t = \beta_1 * m_{t-1} + (1 - \beta_1) * \nabla J(w_t) \qquad (11)$$

$$v_t = \beta_2 * v_{t-1} + (1 - \beta_2) * \nabla J(w_t)^2 \qquad (12)$$

Όπου:

- m_t : εκτιμήσεις πρώτης στιγμής (μέσος όρος)
- v_t: εκτιμήσεις δεύτερης στιγμής (μη συγκεντρωμένη απόκλιση)
- β_1 και β_2 : ρυθμοί αποσύνθεσης (decay rates)

Δεδομένου ότι τα διανύσματα m_t και v_t αρχικοποιούνται ως μηδενικά, οι δημιουργοί του 'Adam' παρατήρησαν ότι τα διανύσματα είναι προκατειλημμένα (biased) προς το μηδέν κατά τη διάρκεια των αρχικών βημάτων και ειδικά όταν οι ρυθμοί αποσύνθεσης β_1 και β_2 είναι μικροί (με τιμή κοντά στο 1). Αυτές οι μεροληψίες

αντισταθμίζονται με υπολογισμό των διορθωμένων προκαταρκτικών εκτιμήσεων πρώτης και δεύτερης στιγμής.

$$\widehat{m}_t = \frac{m_t}{1-\beta_1^t} \qquad (13)$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \qquad (14)$$

Προκύπτει λοιπόν ο κανόνας ενημέρωσης του 'Adam':

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\hat{v}_t} + \varepsilon} * \widehat{m}_t \qquad (15)$$

Οι δημιουργοί του 'Adam' προτείνουν προκαθορισμένες τιμές 0.9, 0.999 και 10^{-8} για το β_1, το β_2 και το ε αντίστοιχα.

2.4 Εκπαίδευση (training)

Ένα καλά εκπαιδευμένο τεχνητό νευρωνικό δίκτυο έχει βάρη που ενισχύουν το σήμα εισόδου και περιορίζουν το θόρυβο. Ένα μεγαλύτερο βάρος επιφέρει μια στενότερη συσχέτιση μεταξύ ενός σήματος και του αποτελέσματος του δικτύου. Οι είσοδοι που έχουν συνδυαστεί με μεγάλα βάρη θα επηρεάσουν την ερμηνεία των δεδομένων από το δίκτυο περισσότερο από τις εισόδους που αντιστοιχούν σε μικρότερα βάρη.

Η διαδικασία εκπαίδευσης για οποιονδήποτε αλγόριθμο μάθησης με χρήση βαρών είναι η διαδικασία επαναπροσαρμογής των βαρών και των προκαταλήψεων (biases). Έτσι, το μοντέλο μαθαίνει ποιοι προγνωστικοί δείκτες ή χαρακτηριστικά συνδέονται με τις επιθυμητές προβλέψεις και προσαρμόζει ανάλογα τα βάρη και τις προκαταλήψεις που περιλαμβάνει.

Στα περισσότερα σύνολα δεδομένων, ορισμένα χαρακτηριστικά συσχετίζονται έντονα με ορισμένες ετικέτες. Τα νευρωνικά δίκτυα μαθαίνουν αυτές τις σχέσεις τυφλά κάνοντας μια εικασία με βάση τις εισόδους και τα βάρη και στη συνέχεια μετρώντας πόσο ακριβή είναι τα αποτελέσματα. Οι συναρτήσεις κόστους σε αλγόριθμους βελτιστοποίησης, όπως η στοχαστική μείωση της κλίσης (SGD), «επιβραβεύουν» το δίκτυο για καλές εικασίες και τον «τιμωρούν» για κακές. Ο αλγόριθμος βελτιστοποίησης ενημερώνει τις παραμέτρους έτσι ώστε το δίκτυο να εξάγει καλύτερες προβλέψεις.

2.4.1 Αλγόριθμος οπισθοδιάδοσης (backpropagation algorithm)

Σημαντικό μέρος της διαδικασίας εκπαίδευσης αποτελεί ο αλγόριθμος οπισθοδιάδοσης ([4]), ο οποίος χρησιμοποιείται για τη μείωση του σφάλματος του νευρωνικού δικτύου.

Το κλειδί είναι να βρεθούν τα βάρη που είναι περισσότερο υπεύθυνα για το εξαγόμενο σφάλμα και στη συνέχεια να αλλάξουν οι τιμές αυτών αναλόγως. Δηλαδή, αν ένα βάρος επηρεάζει περισσότερο την τιμή του σφάλματος τότε πρέπει να δεχθεί μεγαλύτερη προσαρμογή. Ο αλγόριθμος οπισθοδιάδοσης είναι μία ρεαλιστική προσέγγιση για τη διαίρεση της συμβολής σφάλματος για κάθε βάρος του νευρωνικού δικτύου.

Αλγόριθμος 1: Οπισθοδιάδοση για ενημέρωση των βαρών

1. Αρχικοποίησε τα βάρη με τυχαίες τιμές.
2. Για κάθε παράδειγμα στα δεδομένα εκπαίδευσης κάνε τα ακόλουθα:
    a. Υπολόγισε την έξοδο του δικτύου.
    b. Υπολόγισε το σφάλμα και τις παραγώγους του για κάθε νευρώνα του επιπέδου εξόδου.
    c. Ενημέρωσε τα βάρη που οδηγούν στο επίπεδο εξόδου.
    d. Για κάθε κρυφό επίπεδο ξεκινώντας από το επίπεδο εξόδου κάνε τα εξής:
        i. Υπολόγισε το σφάλμα για κάθε κόμβο.
        ii. Ενημέρωσε τα βάρη του επιπέδου.
3. Επανέλαβε το βήμα 2 μέχρι να συγκλίνει το δίκτυο.

2.4.1.1 Ψευδοκώδικας

function backpropagation_algo (network, training_set, learning_rate) returns network
    $network \leftarrow initialize\_weights(randomly)$
    start loop:
        for each example in training_set do:
            // Compute the output for this input example.
            $network\_output \leftarrow nn\_output(network, example)$

            // Compute the error and the [delta]
            // for neurons in the output layer.
            $example\_error \leftarrow target\_output - network\_output$

            // Update the weights leading to the output layer.
            $W_{j,i} \leftarrow W_{j,i} + a * a_j * Err_i * g'(input\_sum_i)$

            for each subsequent-layer in network do:
                // Compute the error at each node.
                $\Delta_j \leftarrow g'(input\_sum_j) * \sum_i (W_{j,i} * \Delta_i)$

                // Update the weights leading into the layer.
                $W_{k,j} \leftarrow W_{k,j} + a * a_k * \Delta_j$
            end for
        end for
    end loop when network has converted
    return network

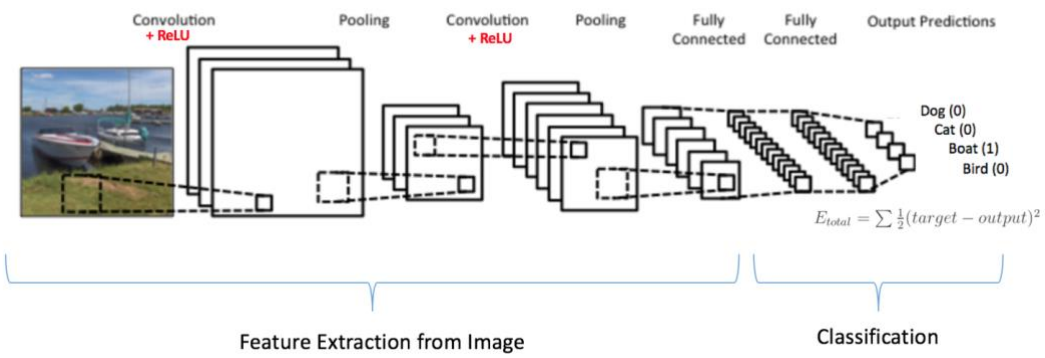
Πίνακας 2.1: Σημασιολογία συμβόλων

Σύμβολο	Σημασία
i	Δείκτης του νευρώνα
j	Δείκτης του νευρώνα σε προηγούμενο επίπεδο, συνδεδεμένου με το νευρώνα i
a_i	Τιμή ενεργοποίησης του νευρώνα i (έξοδος του νευρώνα i)
g	Συνάρτηση ενεργοποίησης
g’	Παράγωγος της συνάρτησης ενεργοποίησης
Err_i	Διαφορά μεταξύ της εξόδου του δικτύου και της πραγματικής τιμής εξόδου για το παράδειγμα εισόδου
W_i	Διάνυσμα βαρών των συνδέσεων του νευρώνα i
$W_{j,i}$	Βάρος της σύνδεσης μεταξύ του νευρώνα j του προηγούμενου επιπέδου και το νευρώνα i του επόμενου
$input\_sum_i$	Άθροισμα εισόδων με βάση τα βάρη στο νευρώνα i
$input\_sum_j$	Άθροισμα εισόδων με βάση τα βάρη στο νευρώνα j στο προηγούμενο επίπεδο
a	Ρυθμός μάθησης
Δ_j	Όρος σφάλματος για συνδεδεμένο νευρώνα j σε προηγούμενο επίπεδο
Δ_i	Όρος σφάλματος για το νευρώνα i

Η συνάρτηση που περιγράφει ο ψευδοκώδικας έχει τις ακόλουθες εισόδους:

- network: ένα πολυεπίπεδο νευρωνικό δίκτυο
- training_set: το σύνολο των δεδομένων εκπαίδευσης
- learning_rate: ο ρυθμός μάθησης

2.4.2 Συνολική διαδικασία εκπαίδευσης συνελικτικού δικτύου



Σχήμα 2.29: Συνολική διαδικασία εκπαίδευσης συνελικτικού νευρωνικού δικτύου ([20])

Αλγόριθμος 2: Συνολική διαδικασία εκπαίδευσης ([20])

1. Αρχικοποίησε όλα τα φίλτρα και τις παραμέτρους/βάρη του δικτύου με τυχαίες τιμές.
2. Το δίκτυο δέχεται μία είσοδο, έστω εικόνα, η οποία προωθείται στο δίκτυο περνώντας από όλα τα επίπεδα (συνελικτικά, συγκεντρωτικά κ.τ.λ.) και εξάγει τις πιθανότητες εξόδου για κάθε μία από τις υπάρχουσες κλάσεις.
    a. Έστω ότι οι κλάσεις είναι 4, όπως φαίνεται στο σχήμα 2.16 και οι πιθανότητες που εξάγονται είναι [0.2, 0.4, 0.1, 0.3].
    b. Εφόσον τα βάρη του δικτύου είναι αρχικοποιημένα με τυχαίες τιμές, οι εξαγόμενες πιθανότητες παίρνουν τυχαίες τιμές.
3. Υπολόγισε το συνολικό σφάλμα στο επίπεδο εξόδου (άθροισμα των 4 κλάσεων)
$$Συνολικό\ Σφάλμα = \sum \frac{1}{2}(πιθανότητα\ στόχου - πιθανότητα\ εξόδου)^2$$
4. Χρησιμοποίησε τη μέθοδο της οπισθοδιάδοσης (backpropagation) για να υπολογίσεις τις κλίσεις (gradients) του σφάλματος αναφορικά με όλα τα βάρη του δικτύου και στη συνέχεια εφάρμοσε τη μέθοδο κατηφορικής κλίσης (gradient descent) προκειμένου να ενημερώσεις όλες τις τιμές των φίλτρων και των βαρών, έτσι ώστε να ελαχιστοποιηθεί το σφάλμα εξόδου.
    a. Τα βάρη προσαρμόζονται με βάση τη συνεισφορά τους στο συνολικό σφάλμα.
    b. Όταν η ίδια εικόνα εισαχθεί στο δίκτυο σαν είσοδος ξανά, τότε οι πιθανότητες μπορεί να είναι [0.1, 0.1, 0.7, 0.1]. Οι τιμές αυτές είναι πιο κοντά στις τιμές του διανύσματος-στόχου [0, 0, 1, 0].
    c. Αυτό σημαίνει ότι το δίκτυο έχει μάθει να κατηγοριοποιεί τη συγκεκριμένη εικόνα σωστά με το να προσαρμόζει τα βάρη/φίλτρα έτσι ώστε να μειώνεται το σφάλμα εξόδου.
    d. Παράμετροι όπως ο αριθμός των φίλτρων, τα μεγέθη των φίλτρων, αρχιτεκτονική του δικτύου, είναι προκαθορισμένα πριν από το βήμα 1 και δεν μεταβάλλονται κατά την διάρκεια της εκπαίδευσης. Μόνο οι τιμές των φίλτρων και τα βάρη των συνδέσεων ενημερώνονται.
5. Επανέλαβε τα βήματα 2-4 με όλες τις διαθέσιμες εικόνες του συνόλου δεδομένων εκπαίδευσης.

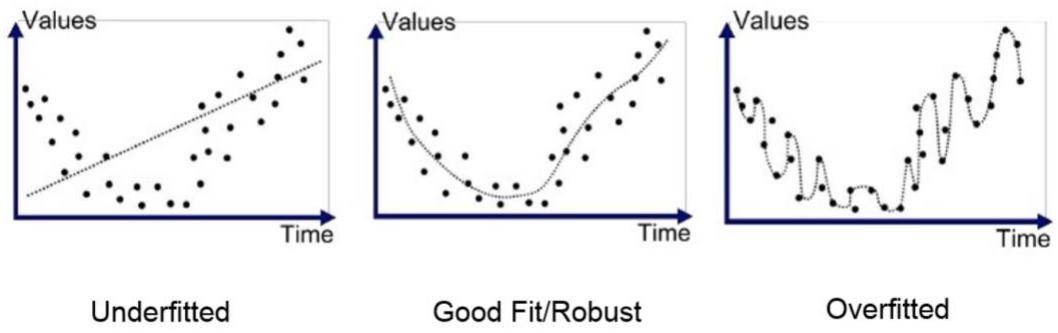
Τα παραπάνω βήματα εκπαιδεύουν ένα συνελικτικό δίκτυο. Αυτό ουσιαστικά σημαίνει ότι όλα τα βάρη και οι παράμετροι του δικτύου έχουν βελτιστοποιηθεί για να ταξινομήσουν σωστά τις εικόνες από το σύνολο δεδομένων εκπαίδευσης.

Όταν μια νέα εικόνα εισάγεται στο μοντέλο, το δίκτυο θα προωθήσει την εικόνα σε όλα τα επίπεδα και θα εξάγει μια πιθανότητα για κάθε κατηγορία (για μια νέα εικόνα, οι πιθανότητες εξόδου υπολογίζονται χρησιμοποιώντας τα βάρη που έχουν βελτιστοποιηθεί για να ταξινομηθούν σωστά όλα τα προηγούμενα παραδείγματα εκπαίδευσης). Εάν το σύνολο εκπαίδευσης είναι αρκετά μεγάλο και το δίκτυο είναι ορθά εκπαιδευμένο τότε το μοντέλο θα γενικεύσει καλά στις νέες εικόνες και θα τις ταξινομήσει σε σωστές κατηγορίες.

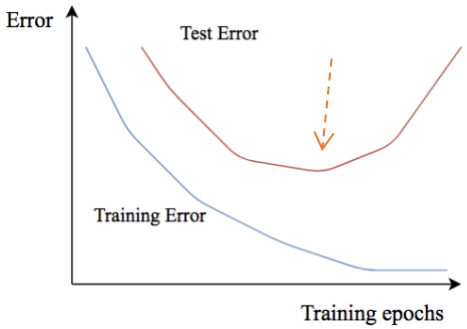
2.5 Πρόβλημα υπερπροσαρμογής (overfitting)

Οι αλγόριθμοι βελτιστοποίησης επιχειρούν πρώτα να λύσουν το πρόβλημα της υποπροσαρμογής (underfitting). Όταν το μοντέλο δεν έχει εκπαιδευτεί αποδοτικά τότε δεν προσεγγίζει καλά τα δεδομένα. Αντιθέτως, σε περίπτωση που το μοντέλο έχει εκπαιδευτεί πολύ καλά στις λεπτομέρειες και στο θόρυβο του εκπαιδευτικού συνόλου δεδομένων και αδυνατεί να γενικεύσει αποδοτικά σε δεδομένα που δεν έχουν χρησιμοποιηθεί στο στάδιο της εκπαίδευσής του, τότε παρουσιάζεται το πρόβλημα της υπερπροσαρμογής (overfitting).

Στόχος των μοντέλων της μηχανικής μάθησης είναι η αποδοτική γενίκευση σε νέα δεδομένα, έτσι ώστε να εξάγονται σωστές προβλέψεις. Επομένως, ιδανικά, το μοντέλο πρέπει να εκπαιδευτεί πολύ καλά με χρήση των δεδομένων εκπαίδευσης, έτσι ώστε να λάβει τη διαθέσιμη πληροφορία και στη συνέχεια πρέπει να εξάγει πολύ καλές προβλέψεις. Υπάρχει λοιπόν μία ισορροπία ανάμεσα στην υποπροσαρμογή και την υπερπροσαρμογή του μοντέλου.



Σχήμα 2.30: Γραφικές παραστάσεις που δείχνουν τον πρόβλημα της υποπροσαρμογής και της υπερπροσαρμογής. Στην πρώτη γραφική το μοντέλο δεν έχει εκπαιδευτεί αρκετά, ενώ στην τρίτη έχει εκπαιδευτεί πάρα πολύ με αποτέλεσμα να μη γενικεύει καλά. Η ιδανική περίπτωση παρουσιάζεται στη δεύτερη γραφική παράσταση. ([4])



Σχήμα 2.31: Πρόβλημα υπερπροσαρμογής. Παρατηρείται ότι μετά από ορισμένο αριθμό εποχών εκπαίδευσης το σφάλμα στο σύνολο δεδομένων για έλεγχο (testing error) αρχίζει να αυξάνεται (το μοντέλο δε γενικεύει καλά), ενώ το σφάλμα στο σύνολο δεδομένων για εκπαίδευση (training error) συνεχίζει και μειώνεται.

2.5.1 Τρόποι αντιμετώπισης

Ο εντοπισμός του προβλήματος υπερπροσαρμογής είναι σημαντικό βήμα, πρέπει όμως να αντιμετωπιστεί. Υπάρχουν αρκετοί τρόποι αντιμετώπισης, έτσι ώστε το δίκτυο να εκπαιδευτεί αποδοτικά έχοντας τη δυνατότητα να γενικεύει με μεγάλη ακρίβεια.

2.5.1.1 Αύξηση συνόλου δεδομένων εκπαίδευσης

Ένα μοντέλο που έχει υπερπροσαρμοστεί μπορεί να αποδώσει καλύτερα εάν ο αλγόριθμος μάθησης επεξεργάζεται περισσότερα δεδομένα εκπαίδευσης. Ενώ ένα υπάρχον σύνολο δεδομένων μπορεί να είναι περιορισμένο, για ορισμένα προβλήματα μηχανικής μάθησης υπάρχουν σχετικά εύκολοι τρόποι δημιουργίας συνθετικών δεδομένων. Για τις εικόνες ορισμένες συνήθεις τεχνικές είναι η περιστροφή, η ανάκλαση και η κλιμάκωση.

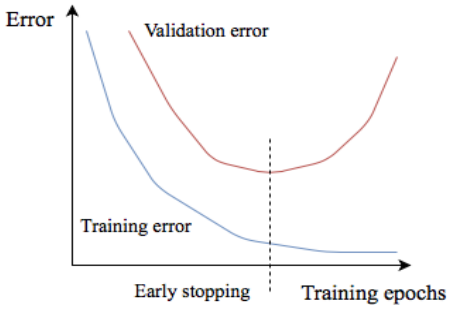
Δεν υπάρχει γενική συνταγή σχετικά με το πώς πρέπει να δημιουργούνται νέα δεδομένα και ποικίλει πολύ από πρόβλημα σε πρόβλημα ανάλογα με τη μορφή των δεδομένων που επεξεργάζεται το μοντέλο. Η γενική αρχή είναι η επέκταση του συνόλου δεδομένων εφαρμόζοντας λειτουργίες που αντικατοπτρίζουν τις πραγματικές μεταβολές του κόσμου όσο το δυνατόν πλησιέστερα. Η ύπαρξη καλύτερου συνόλου δεδομένων στην πράξη συμβάλλει σημαντικά στην ποιότητα των μοντέλων, ανεξάρτητα από την αρχιτεκτονική.

Γενικά η αύξηση του συνόλου δεδομένων εκπαίδευσης βοηθά το μοντέλο να εκπαιδεύεται πιο αποδοτικά και να μην υπερπροσαρμόζεται σε ένα μικρό σύνολο. Έτσι, το μοντέλο εν δυνάμει γενικεύει πιο καλά.

2.5.1.2 Πρόωρη διακοπή εκπαίδευσης (early stopping)

Η πρόωρη διακοπή καταπολεμά την υπερπροσαρμογή διακόπτοντας τη διαδικασία εκπαίδευσης, όταν η απόδοση του μοντέλου σε ένα σύνολο δεδομένων επικύρωσης (validation set) μειωθεί. Ένα σύνολο δεδομένων επικύρωσης είναι ένα σύνολο παραδειγμάτων που δεν χρησιμοποιείται στην εκπαίδευση του μοντέλου, αλλά δεν είναι επίσης μέρος του συνόλου ελέγχου (test set). Τα παραδείγματα επικύρωσης θεωρούνται αντιπροσωπευτικά των μελλοντικών παραδειγμάτων ελέγχου. Η πρόωρη διακοπή ρυθμίζει αποτελεσματικά την υπερπαράμετρο εποχές εκπαίδευσης (epochs).

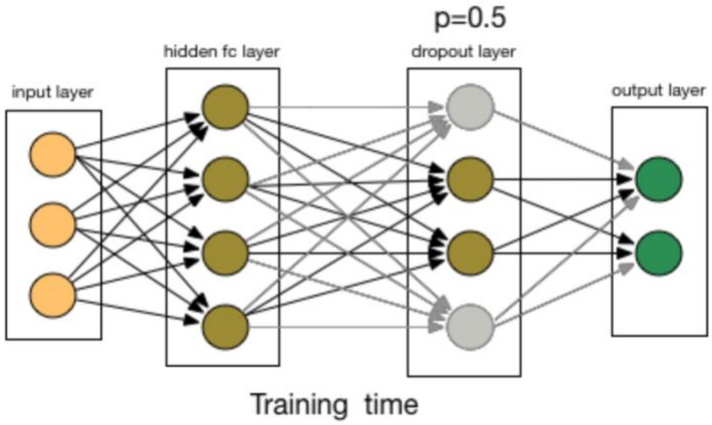
Διαισθητικά, καθώς το μοντέλο βλέπει περισσότερα δεδομένα και μαθαίνει πρότυπα και συσχετίσεις, το σφάλμα εκπαίδευσης (training error) και επικύρωσης (validation error) μειώνονται. Μετά από αρκετά περάσματα πάνω από τα δεδομένα εκπαίδευσης, το μοντέλο μπορεί να ξεκινήσει να υπερπροσαρμόζεται. Σε αυτή την περίπτωση, το σφάλμα της εκπαίδευσης θα συνεχίσει να μειώνεται ενώ το σφάλμα επικύρωσης, που δηλώνει πόσο καλά γενικεύει το μοντέλο, θα αρχίσει να αυξάνεται. Η πρόωρη διακοπή στοχεύει να σταματήσει τη διαδικασία εκπαίδευσης όταν το σφάλμα στα δεδομένα επικύρωσης αρχίσει να αυξάνεται.



Σχήμα 2.32: Εύρεση σημείου με ελάχιστο σφάλμα επικύρωσης (validation error).

2.5.1.3 Περιορισμός ενεργοποίησης (dropout)

Μία πιο πρόσφατη προσέγγιση στην αντιμετώπιση του προβλήματος της υπερπροσαρμογής περιλαμβάνει τον περιορισμό ενεργοποίησης. Ο περιορισμός ενεργοποίησης ([10]) είναι ένας υπολογιστικός φθηνός τρόπος ομαλοποίησης κατά τη διάρκεια της εκπαίδευσης του μοντέλου. Στο επίπεδο αυτό, σε κάθε επανάληψη της εκπαίδευσης, απενεργοποιούνται τυχαία κάποιοι νευρώνες του δικτύου μαζί με όλες τις εισερχόμενες και εξερχόμενες συνδέσεις. Εισάγεται μία πιθανότητα ενεργοποίησης ή όχι του εκάστοτε νευρώνα, μία συνήθης τιμή πιθανότητας είναι το 0.5 (50%).



Σχήμα 2.33: Επίπεδο περιορισμού ενεργοποίησης όπου ο κάθε νευρώνας έχει πιθανότητα 0.5 να απενεργοποιηθεί. ([4])

2.5.1.4 Ομαλοποίηση L1 και L2 (regularization)

Η ποινή βάρους είναι συνήθης τρόπος ομαλοποίησης (regularization) , που χρησιμοποιείται ευρέως στην εκπαίδευση μοντέλων. Βασίζεται έντονα στην υπόθεση ότι ένα μοντέλο με μικρά βάρη είναι κατά κάποιο τρόπο απλούστερο από ένα δίκτυο με μεγάλα βάρη. Οι ποινές χρησιμοποιούνται για να κρατηθούν τα βάρη μικρά ή ανύπαρκτα (μηδέν). Μια εναλλακτική ονομασία στη βιβλιογραφία για τις ποινές βάρους είναι η «αποσύνθεση του βάρους» (weight decay), καθώς αναγκάζει τα βάρη να φθίνουν προς το μηδέν. Οι ομαλοποιήσεις που χρησιμοποιούνται κυρίως για τον περιορισμό των βαρών είναι η L1 και η L2.

Ομαλοποίηση L1:

- Για κάθε βάρος w προστίθεται ο όρος $\lambda|w|$ στη συνάρτηση κόστους.
- Κάποια βάρη τείνουν κατ' ευθείαν στο μηδέν, ενώ ορισμένα εξακολουθούν να έχουν μεγάλες τιμές.

Ομαλοποίηση L2:

- Για κάθε βάρος w προστίθεται ο όρος $\frac{1}{2}\lambda w^2$ στη συνάρτηση κόστους. Έτσι η συνάρτηση κόστους λαμβάνει τη μορφή: $J(x,y)_{new} = J(x,y)_{old} + 0.5\lambda w^2$
- Τείνει να οδηγεί όλα τα βάρη σε μικρότερες τιμές.

2.6 Μεταφορά μάθησης (transfer learning)

Είναι σπάνιο ένα μοντέλο συνελικτικού δικτύου να εκπαιδεύεται από την αρχή με όλες τις παραμέτρους του αρχικοποιημένες με τυχαίες τιμές, λόγω του μεγάλου υπολογιστικού φόρτου που απαιτείται για την ολοκλήρωση της εκπαίδευσης. Επίσης συχνά δεν είναι διαθέσιμος τόσο μεγάλος όγκος δεδομένων για να καθίσταται δυνατή η πολύ αποδοτική εκπαίδευση του μοντέλου. Ένας τρόπος για να αντιμετωπιστούν αυτά τα προβλήματα είναι η χρήση μία αρχιτεκτονικής που είναι ήδη πολύ καλά εκπαιδευμένη. Η αρχιτεκτονική αυτή προστίθεται στο τελικό μοντέλο και μετέπειτα γίνεται εκπαίδευση όλου του μοντέλου σε ένα συγκεκριμένο διαθέσιμο σύνολο δεδομένων. Η διαδικασία αυτή ονομάζεται μεταφορά μάθησης (transfer learning).

Τα συνελικτικά δίκτυα έχει αποδειχθεί ότι μαθαίνουν γενικά οπτικά χαρακτηριστικά στα πρώτα επίπεδα και στη συνέχεια αναπτύσσουν σταδιακά πιο σύνθετα χαρακτηριστικά που αφορούν συγκεκριμένα στοιχεία σε μεταγενέστερα επίπεδα.

Στην περίπτωση της μεταφοράς μάθησης, ένα συνελικτικό μοντέλο εκπαιδευμένο σε ένα μεγάλο σύνολο δεδομένων, όπως το ImageNet ([37]), χρησιμοποιείται χωρίς το τελευταίο επίπεδο εξόδου, όπου συνήθως γίνεται η κατηγοριοποίηση. Το τελευταίο επίπεδο αντικαθίσταται με ένα άλλο που επιλύει ένα συγκεκριμένο πρόβλημα.

Για παράδειγμα το προεκπαιδευμένο μοντέλο μπορεί να έχει βελτιστοποιηθεί ώστε να κατηγοριοποιεί 1000 διαφορετικά αντικείμενα, ενώ το πρόβλημα που πρέπει να επιλυθεί είναι η κατηγοριοποίηση ανθρώπινων εικόνων με βάση τη συναισθηματική κατάσταση του εικονιζόμενου ατόμου. Έτσι, το τελευταίο επίπεδο θα αντικατασταθεί από ένα επίπεδο με 4 εξόδους, μία για κάθε συναισθηματική κατάσταση (θυμός, λύπη, χαλάρωση, χαρά).

Εφόσον γίνει η αντικατάσταση του τελευταίου επιπέδου, το μοντέλο επανεκπαιδεύεται με χρήση συγκεκριμένων δεδομένων σχετικών με το πρόβλημα (πχ. εικόνες ανθρώπων με ετικέτα που δηλώνει τη συναισθηματική τους κατάσταση). Ορισμένα δεδομένα θα συνεχίσουν να προκαλούν μικρές τροποποιήσεις των βαρών σε όλα τα επίπεδα του μοντέλου με χρήση του αλγόριθμου οπισθοδιάδοσης και άλλα θα ενημερώσουν μόνο τα μεταγενέστερα επίπεδα. Αυτό οφείλεται στο γεγονός ότι πολλά από τα χαρακτηριστικά, που εξάγονται από τα αρχικά επίπεδα, είναι γενικά για όλους τους τύπους προβλημάτων όρασης υπολογιστών και υπάρχει λιγότερη ανάγκη να ενημερωθούν. Τα μεταγενέστερα επίπεδα επικεντρώνονται στο συνδυασμό αυτών των

χαρακτηριστικών για να εξάγουν πιο σύνθετα χαρακτηριστικά που είναι πιο συγκεκριμένα στο πρόβλημα που καλείται να επιλύσει το μοντέλο.

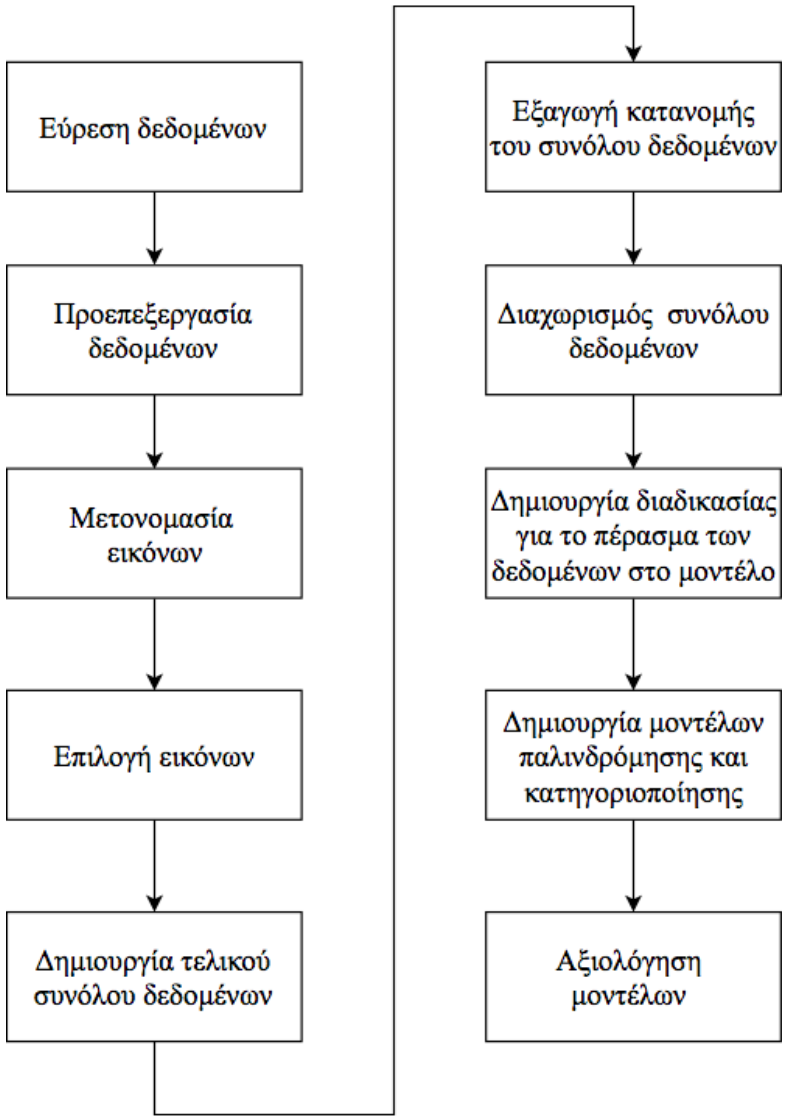
Αξίζει να σημειωθεί ότι αν και συνήθως αντικαθίσταται μόνο το τελευταίο επίπεδο του προ-εκπαιδευμένου δικτύου, υπάρχουν περιπτώσεις που μπορεί να αφαιρούνται περισσότερα επίπεδα και στη συνέχεια να προστίθεται μία διαφορετική αρχιτεκτονική.

3

Υλοποίηση και Σχεδιασμός μοντέλων

3.1 Γενική περιγραφή

Σε αυτό το κεφάλαιο θα γίνει η περιγραφή της διαδικασίας που ακολουθήθηκε για την υλοποίηση και το σχεδιασμό των πιθανοτικών μοντέλων. Η ανίχνευση της συναισθηματικής κατάσταση του ατόμου με βάση τις εκφράσεις του προσώπου προσεγγίστηκε ως πρόβλημα παλινδρόμησης (regression) και κατηγοριοποίησης (classification). Στο διάγραμμα που ακολουθεί φαίνονται σε υψηλό επίπεδο όλα τα στάδια της διαδικασίας. Στη συνέχεια γίνεται λεπτομερής ανάλυση κάθε σταδίου.



Σχήμα 3.1: Στάδια υλοποίησης και σχεδιασμού των μοντέλων

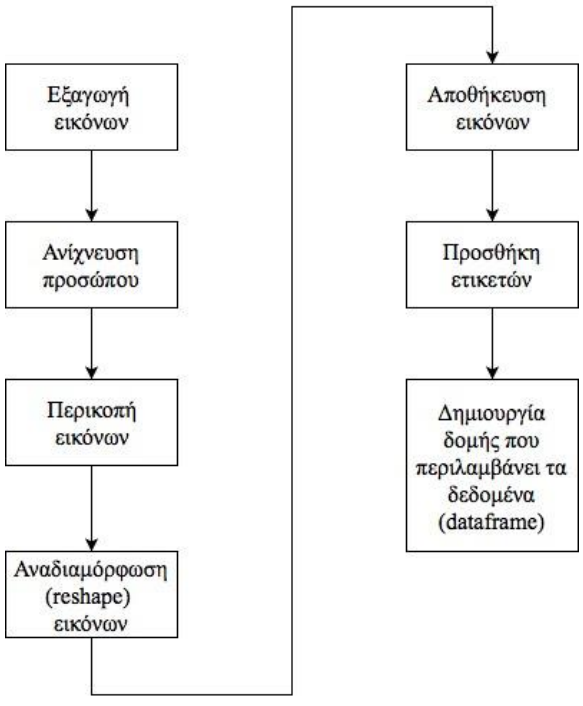
3.2 Εύρεση δεδομένων

Για τη υλοποίηση των πειραμάτων χρησιμοποιήθηκε η βάση δεδομένων ‘Semaine’ ([15]). Το πανεπιστήμιο Queen’s Belfast σε συνεργασία με το Imperial δημιούργησε μια μεγάλη οπτικοακουστική βάση δεδομένων ως μέρος μιας επαναληπτικής προσέγγισης για την κατασκευή πρακτόρων (agents) “Sensitive Artificial Listener” (SAL) που μπορούν να εμπλέξουν ένα άτομο σε μια συνεχή, συναισθηματικά φορτισμένη συνομιλία. Έγιναν εγγραφές σε συνολικά 150 συμμετέχοντες (959 συνομιλίες) με ξεχωριστούς χαρακτήρες SAL. Στις συνεδρίες προστέθηκαν ετικέτες σε διάφορες «συναισθηματικές διαστάσεις», όπως η διέγερση (arousal), το σθένος-ένταση (valence), φόβος και ο θυμός.

Στα πλαίσια της εργασίας χρησιμοποιήθηκαν αρχικά 94 συνεδρίες με ετικέτες που αφορούσαν τη διέγερση (arousal) και το σθένος (valence). Οι δύο αυτές ετικέτες είναι αρκετές για να καθορίσουν ένα συναίσθημα όπως περιγράφεται στο διάγραμμα 1.3. Σε κάθε συνεδρία συνήθως υπήρχαν παραπάνω από έναν εκτιμητές (raters) που καθόριζαν τις τιμές των ετικετών. Οι αποκλίσεις μεταξύ τους δεν ήταν μεγάλες. Κυρίως χρησιμοποιήθηκε ο εκτιμητής υπ’ αριθμόν 5 και λιγότερο οι 3, 4 και 6.

3.3 Προεπεξεργασία δεδομένων

Η προεπεξεργασία δεδομένων αποτέλεσε ένα μεγάλο μέρος της όλης διαδικασίας καθώς τα δεδομένα ήταν σε μορφή βίντεο. Το στάδιο αυτό ήταν πολύ σημαντικό για την απόδοση του μοντέλου καθώς η χρήση υψηλής ποιότητας δεδομένων έχει σαν αποτέλεσμα να εκπαιδεύεται καλύτερα το νευρωνικό δίκτυο. Στο σχήμα που ακολουθεί παρουσιάζονται όλα τα επιμέρους στάδια της προεπεξεργασίας δεδομένων.



Σχήμα 3.2: Επιμέρους στάδια της προεπεξεργασίας δεδομένων

3.3.1 Εξαγωγή εικόνων

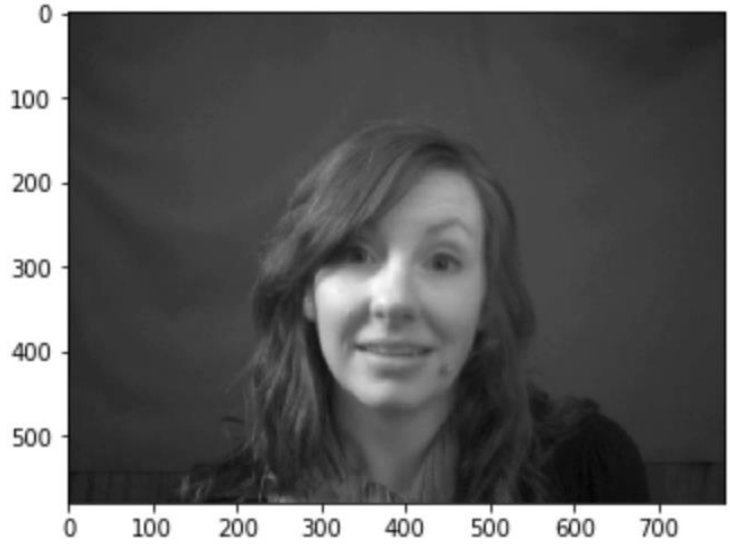
Στο στάδιο αυτό εξάχθηκαν εικόνες από τα βίντεο όλων των συνεδριών. Τα βίντεο είχαν τραβηχτεί με 5 καρέ ανά δευτερόλεπτο. Επιλέχθηκε δειγματοληψία με 1 εικόνα ανά 4 δευτερόλεπτα. Μικρότερη δειγματοληψία θα οδηγούσε σε εξαγωγή πολλών παρόμοιων εικόνων. Ο επιλεγόμενος ρυθμός εξαγωγής ήταν κατάλληλος για να δημιουργεί ένα μεγάλο σύνολο δεδομένων. Σε ορισμένες συνεδρίες επειδή δεν υπήρχε ιδιαίτερη συναισθηματική αλλαγή χρησιμοποιήθηκε δειγματοληψία με 1 εικόνα ανά 5 δευτερόλεπτα (συνεδρίες 10, 18, 27, 48, 56, 77).

Πίνακας 3.1: Πλήθος εξαγόμενων εικόνων ανά συνεδρία

Συνεδρία	Πλήθος εικόνων		Συνεδρία	Πλήθος εικόνων		Συνεδρία	Πλήθος εικόνων
1	768		33	1046		65	1093
2	311		34	1165		66	985
3	852		35	933		67	395
4	853		36	740		68	658
5	370		37	866		69	476
6	453		38	411		70	588
7	724		39	1003		71	1006
8	876		40	1298		72	993
9	259		41	765		73	561
10	412		42	1011		74	565
11	612		43	433		75	459
12	501		44	680		76	503
13	356		45	602		77	224
14	462		46	816		78	349
15	723		47	698		79	571
16	1007		48	701		80	555
17	373		49	549		81	644
18	606		50	887		82	587
19	1276		51	834		83	313
20	504		52	1507		84	738
21	577		53	1089		85	763
22	678		54	1151		86	526
23	842		55	623		87	438
24	716		56	668		88	526
25	588		57	281		89	526
26	785		58	1226		90	1113
27	606		59	455		91	688
28	639		60	793		92	788
29	796		61	686		93	738
30	728		62	889		94	1038
31	343		63	628			
32	930		64	926			

Συνολικά, από αυτή τη διαδικασία εξάχθηκαν 66.292 εικόνες.

Παράδειγμα εξαγόμενης εικόνας από τη συνεδρία 51:



3.3.2 Ανίχνευση προσώπου

Οι εικόνες που είχαν εξαχθεί από το προηγούμενο στάδιο δεν περιλάμβαναν μόνο το πρόσωπο του ανθρώπου. Προκειμένου να εκπαιδευθούν αποδοτικά τα μοντέλα έπρεπε να γίνει ανίχνευση του προσώπου σε κάθε εικόνα. Για την επίλυση αυτού του προβλήματος χρησιμοποιήθηκε ένα έτοιμος ταξινομητής με πολύ καλή απόδοση (Haar cascade classifier for frontal face detection) ([13]).

Το "cascading" είναι μια ιδιαίτερη περίπτωση μάθησης που βασίζεται στη συνένωση αρκετών ταξινομητών, χρησιμοποιώντας όλες τις πληροφορίες που συλλέγονται από την έξοδο από έναν δεδομένο ταξινομητή ως πρόσθετες πληροφορίες για τον επόμενο ταξινομητή. Οι διαβαθμισμένοι ταξινομητές εκπαιδεύονται με πολλές «θετικές» δειγματοληπτικές προβολές ενός συγκεκριμένου αντικειμένου, όπως ανθρώπινα πρόσωπα, και αυθαίρετες «αρνητικές» εικόνες ίδιου μεγέθους. Αφού εκπαιδευτεί ο ταξινομητής, μπορεί να εφαρμοστεί σε μια περιοχή μιας εικόνας και να ανιχνεύσει το εν λόγω αντικείμενο. Αυτή η διαδικασία χρησιμοποιείται πιο συχνά στην επεξεργασία εικόνας για ανίχνευση και παρακολούθηση αντικειμένων, κυρίως ανίχνευση και αναγνώριση προσώπου. Ο πρώτος ταξινομητής είναι ο ανιχνευτής προσώπου των Viola και Jones ([13]).

Ο ταξινομητής που αναφέρθηκε χρησιμοποιήθηκε σε όλες τις εικόνες έτσι ώστε να γίνει η ανίχνευση του ανθρώπινου προσώπου. Στις περισσότερες εικόνες η ανίχνευση έγινε με επιτυχία ενώ σε κάποιες δε βρέθηκε το πρόσωπο, οι εικόνες αυτές αφαιρέθηκαν από το σύνολο δεδομένων. Υπήρχε επίσης μία ιδιαίτερη περίπτωση όπου ο ταξινομητής ανίχνευε περισσότερα από ένα πρόσωπα, συνήθως όταν υπήρχε στιγμιαία κίνηση του ατόμου. Σε αυτή την περίπτωση έγινε έλεγχος δεύτερου επιπέδου έτσι ώστε να βρεθεί ποια ανίχνευση προσώπου ήταν η σωστή.

Πίνακας 3.2 Αποτελέσματα ταξινομητή για ανίχνευση προσώπου

Συνεδρία	Εικόνες με πρόσωπο	Εικόνες με πολλά πρόσωπα	Εικόνες με κανένα πρόσωπο		Συνεδρία	Εικόνες με ένα πρόσωπο	Εικόνες με πολλά πρόσωπα	Εικόνες με κανένα πρόσωπο
1	768	0	0		48	701	0	0
2	310	1	1		49	548	0	1
3	852	0	0		50	878	0	9
4	845	0	7		51	817	1	17
5	370	0	0		52	1492	0	15
6	453	0	0		53	1067	0	22
7	722	0	2		54	1110	0	41
8	871	0	5		55	622	4	1
9	258	0	1		56	667	0	1
10	410	0	2		57	281	1	0
11	596	3	16		58	1225	0	1
12	454	9	47		59	430	6	25
13	356	0	0		60	464	16	329
14	462	0	0		61	277	4	409
15	719	0	4		62	163	7	726
16	1007	0	0		63	628	2	0
17	373	0	0		64	918	5	8
18	600	0	6		65	1089	1	4
19	1269	9	7		66	983	2	2
20	503	0	1		67	390	0	5
21	573	3	4		68	625	5	33
22	671	27	7		69	465	0	11
23	841	0	1		70	555	0	33
24	704	0	12		71	1006	41	0
25	575	0	13		72	993	24	0
26	785	0	0		73	558	6	3
27	603	0	3		74	565	13	0
28	639	0	0		75	459	1	0
29	796	0	0		76	503	0	0
30	724	0	4		77	224	0	0
31	331	2	12		78	348	0	1
32	862	6	68		79	566	0	5
33	1034	6	12		80	553	0	2
34	1161	4	4		81	642	0	2
35	932	2	1		82	584	0	3
36	740	2	0		83	313	0	0
37	865	1	1		84	732	38	6
38	411	2	0		85	754	44	9
39	1003	0	0		86	518	11	8
40	1298	0	0		87	438	0	0
41	765	0	0		88	526	0	0
42	1011	0	0		89	525	0	1
43	433	122	0		90	1112	2	1

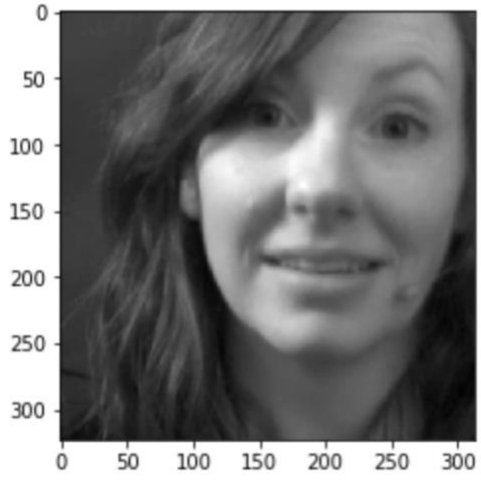
44	680	426	0		91	675	16	13
45	602	500	0		92	764	13	24
46	816	315	0		93	639	10	99
47	696	0	2		94	992	13	46

Συνολικά, οι εικόνες με ανιχνεύσιμο πρόσωπα ήταν 64.133. Οι συνεδρίες 61 και 62 αφαιρέθηκαν εξ ολοκλήρου από το σύνολο δεδομένων καθώς περιείχαν πολλές εικόνες που δεν ανιχνεύθηκε κάποιο πρόσωπο από τον ταξινομητή. Επομένως, μετά το πέρας αυτού του σταδίου, το σύνολο δεδομένων περιλάμβανε 63.693 εικόνες.

3.3.3 Περικοπή εικόνων

Ο ταξινομητής που χρησιμοποιήθηκε για την ανίχνευση των προσώπων, όταν έβρισκε κάποιο πρόσωπο επέστρεφε 4 τιμές που όριζαν ένα παραλληλόγραμμο εντός του οποίου βρισκόταν το πρόσωπο. Με βάση αυτή την πληροφορία έγινε η περικοπή των εικόνων έτσι ώστε να περιλαμβάνουν μόνο τα πρόσωπα των ανθρώπων. Το δύσκολο κομμάτι της διαδικασίας ήταν ότι οι τιμές που επιστρέφονταν δεν όριζαν πάντα ένα καλό παραλληλόγραμμο που περιλάμβανε όλο το πρόσωπο. Έτσι, για κάθε συνεδρία ορίστηκε ένα πιο κατάλληλο πλαίσιο εντός του οποίου βρισκόταν το πρόσωπο. Η αναπροσαρμογή του πλαισίου βασιζόταν στις τιμές που επέστρεφε ο ταξινομητής και συνήθως υπήρχε μία απόκλιση έως 30 pixels σε κάθε διάσταση.

Παράδειγμα εικόνας μετά την περικοπή:

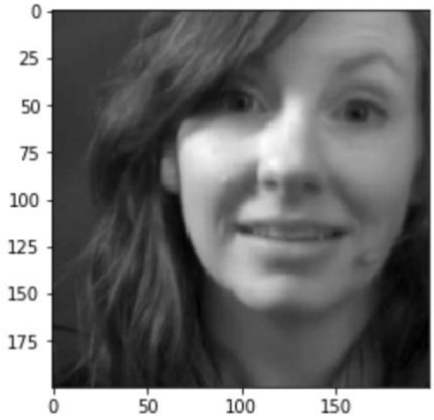


3.3.4. Αναδιαμόρφωση (reshape) εικόνων

Το νευρωνικό δίκτυο πρέπει να δέχεται εικόνες ίδιου μεγέθους έτσι ώστε να εκπαιδεύεται και να λειτουργεί ορθά. Επειδή στη διαδικασία της περικοπής των εικόνων δε εξασφαλιζόταν ότι οι εξαγόμενες εικόνες θα έχουν το ίδιο μέγεθος,

ακολούθησε το στάδιο της αναδιαμόρφωσης. Στο στάδιο αυτό ήταν σημαντικό να οριστεί κατάλληλα το μήκος και το πλάτος των εικόνων καθώς δεν έπρεπε να προστεθεί πολύς θόρυβος αλλά ταυτόχρονα δεν έπρεπε να είναι μεγάλες οι εικόνες που θα προέκυπταν, γιατί κάτι τέτοιο θα καθυστερούσε την εκπαίδευση των μοντέλων. Εν τέλει, επιλέχθηκε να γίνει αναδιαμόρφωση σε διαστάσεις 200 x 200.

Παράδειγμα εικόνας μετά την αναδιαμόρφωση:



3.3.5 Αποθήκευση εικόνων

Μετά το στάδιο της αναδιαμόρφωσης οι εικόνες αποθηκεύτηκαν ανά συνεδρία σε κατάλληλου υποφακέλους της μορφής ‘Dataset/Session1/S1_13.png’. Όλη η διαχείριση των δεδομένων έγινε σε επίπεδο συνεδρίας.

3.3.6 Προσθήκη ετικετών

Σε κάθε συνεδρία υπήρχαν διάφορες ετικέτες που χαρακτήριζαν τη συναισθηματική κατάσταση του ατόμου. Στην παρούσα εργασία χρησιμοποιήθηκαν σαν ετικέτες αυτές του σθένους (valence) και της διέγερσης (arousal). Αυτές οι δύο τιμές, με βάση τη θεωρία, αρκούν για να καθορίσουν τη συναισθηματική κατάσταση. Έγινε η προσθήκη αυτών των ετικετών, εφόσον προηγήθηκε ο αντίστοιχος έλεγχος και διαπιστώθηκε δεν έλειπαν τιμές από τα αρχεία. Υπήρχε πρόβλημα με τις ετικέτες της συνεδρίας 38, για αυτό αφαιρέθηκε από το σύνολο δεδομένων.

3.3.7 Δημιουργία δομής (dataframe) που περιλαμβάνει τα δεδομένα

Μία πολύ εύχρηστη δομή για τη διαχείριση των δεδομένων είναι η ‘dataframe’ ([28]). Για κάθε συνεδρία δημιουργήθηκε μία τέτοια δομή που περιλάμβανε τις εικόνες (paths) και τις ετικέτες του σθένους (valence) και της διέγερσης (arousal). Η δομή εξάχθηκε σε ένα αρχείο .csv για να αποθηκευτεί. Ακολουθεί παράδειγμα της δομής.

Πίνακας 3.3 Παράδειγμα δομής dataframe

Arousal	Paths	Valence
-0.0363	Dataset/Session2/S2_179.png	-0.4218
0.0332	Dataset/Session2/S2_180.png	-0.4269
0.121	Dataset/Session2/S2_181.png	-0.4576
0.2088	Dataset/Session2/S2_182.png	-0.4934
0.2991	Dataset/Session2/S2_183.png	-0.5291

3.4 Μετονομασία εικόνων

Για καλύτερη διαχείριση των δεδομένων έγινε μετονομασία των εικόνων. Για κάθε συνεδρία φορτώθηκαν τα αντίστοιχα .csv αρχεία. Τα νέα ονόματα των εικόνων ήταν ο αύξων αριθμός (πχ. 34.png). Οι εικόνες αποθηκεύτηκαν στον υποφάκελο 'Images' και δημιουργήθηκαν νέα .csv αρχεία (πχ. S2_data_updated.csv) με ενημερωμένα τα «μονοπάτια» (paths) των εικόνων.

3.5 Επιλογή εικόνων

Παρόλα τα στάδια επεξεργασίας στο σύνολο δεδομένων υπήρχαν εικόνες που δεν περιλάμβαναν ολόκληρο το πρόσωπο του ατόμου. Για να αυξηθεί η ποιότητα του συνόλου δεδομένων έγινε έλεγχος των εικόνων και κρατήθηκαν μόνο αυτές όπου φαίνεται ευκρινώς όλο το πρόσωπο. Μετά το πέρας της διαδικασίας δημιουργήθηκαν τα τελικά .csv αρχεία (πχ. S2_final_data.csv).

Πίνακας 3.4: Επιλογή εικόνων ανά συνεδρία

Συνεδρία	Διαγραμμένες Εικόνες	Εικόνες προς χρήση		Συνεδρία	Διαγραμμένες Εικόνες	Εικόνες προς χρήση
1	21	747		48	79	622
2	6	304		49	34	513
3	10	842		50	226	651
4	32	813		51	105	712
5	1	369		52	43	1449
6	10	443		53	109	958
7	31	691		54	226	884
8	32	839		55	35	587
9	32	226		56	47	620
10	34	376		57	12	268
11	87	506		58	186	1039
12	123	331		59	21	397
13	0	356		60	51	413
14	0	460		61	*	*
15	18	701		62	*	*

16	18	989		63	46	582
17	4	369		64	96	822
18	49	551		65	138	951
19	50	1219		66	124	859
20	13	490		67	18	372
21	37	536		68	94	531
22	70	601		69	22	443
23	17	824		70	20	535
24	38	666		71	30	976
25	70	505		72	47	946
26	114	671		73	24	533
27	54	549		74	3	560
28	115	524		75	2	457
29	64	732		76	2	501
30	61	663		77	2	222
31	29	302		78	6	340
32	129	723		79	26	540
33	124	906		80	28	525
34	65	1096		81	9	633
35	54	878		82	11	573
36	27	713		83	3	310
37	67	798		84	58	674
38	*	*		85	52	699
39	3	1000		86	89	429
40	17	1281		87	1	437
41	20	745		88	6	520
42	41	970		89	0	525
43	14	429		90	9	1103
44	4	676		91	76	594
45	0	602		92	100	664
46	7	809		93	135	503
47	66	630		94	211	781

** Οι συγκεκριμένες συνεδρίες δε χρησιμοποιήθηκαν (38, 61, 62).*

3.6 Δημιουργία τελικού συνόλου δεδομένων

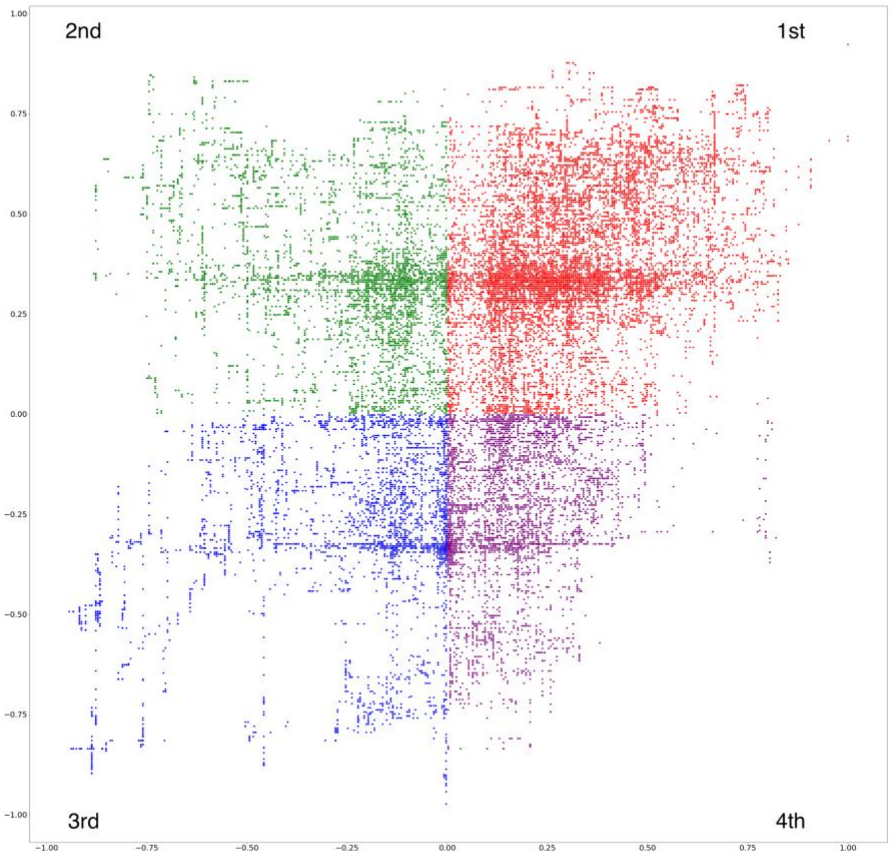
Το τελικό σύνολο δεδομένων που προέκυψε, περιλάμβανε 58.704 εικόνες. Το πλήθος των εικόνων είναι ικανοποιητικό προκειμένου να εκπαιδευτεί ένα βαθύ νευρωνικό δίκτυο και να ελεγχθεί η απόδοσή του. Οι συνεδρίες που κρατήθηκαν ήταν 91. Δημιουργήθηκε ένα τελικό .csv αρχείο που περιέχει όλα τα δεδομένα (overall.csv).

3.7 Εξαγωγή κατανομής του συνόλου δεδομένων

Προκειμένου να εξεταστεί ολοκληρωμένα η ανίχνευση συναισθημάτων μέσα από εκφράσεις του ανθρώπινου προσώπου, έπρεπε να υπάρχει επαρκής αναπαράσταση σε όλο το «συναισθηματικό» χώρο που ορίζουν η διέγερση (arousal) και το σθένος (valence). Για το λόγο αυτό ελέγχθηκε η κατανομή του συνόλου δεδομένων.

Πίνακας 3.5: Κατανομή συνόλου δεδομένων

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	27.252
2ο	11.479
3ο	9.872
4ο	10.101



Σχήμα 3.3: Κατανομή συνόλου δεδομένων στο χώρο που ορίζουν η διέγερση (arousal) και το σθένος (valence).

Στο σχήμα 3.3 φαίνεται ποιοτικά η κατανομή των δεδομένων στο χώρο. Το πλήθος δειγμάτων στα τεταρτημόρια 2, 3 και 4 είναι περίπου ίδιο. Υπάρχει μεγαλύτερη αναπαράσταση στο πρώτο τεταρτημόριο καθώς στις συνεδρίες που χρησιμοποιήθηκαν τα εύθυμα συναισθήματα ήταν πιο συνηθισμένα. Σε κάθε περίπτωση, η αναπαράσταση ήταν ικανοποιητική έτσι ώστε να γίνει προσέγγιση του θέματος τόσο ως πρόβλημα παλινδρόμησης (regression), όσο ως πρόβλημα κατηγοριοποίησης (classification). Δε χρειάστηκε να εφαρμοστεί κάποια τεχνική υποδειγματοληψίας (undersampling) ή υπερδειγματοληψίας (oversampling).

3.8 Διαχωρισμός συνόλου δεδομένων

Για την ορθή εκτέλεση των πειραμάτων έπρεπε να γίνει διαχωρισμός των δεδομένων σε τρία σύνολα. Το σύνολο εκπαίδευσης (training set) περιλάμβανε το 80% των συνολικών δεδομένων και χρησιμοποιήθηκε για την εκπαίδευση των μοντέλων. Το σύνολο επικύρωσης (validation set) είχε το 10% και χρησιμοποιήθηκε για τον έλεγχο το στάδιο της εκπαίδευσης έτσι ώστε να αποφευχθεί ο κίνδυνος της υπερπροσαρμογής (overfitting) του μοντέλου. Τέλος, το υπόλοιπο 10% των δεδομένων (test set) χρησιμοποιήθηκε για την αξιολόγηση των μοντέλων.

Ο διαχωρισμός που περιγράφηκε, έγινε με τυχαίο τρόπο. Τα σύνολα δεδομένων που δημιουργήθηκαν εξάχθηκαν και παρέμειναν σταθερά σε όλα τα πειράματα που εκτελέστηκαν, έτσι ώστε να είναι αντικειμενικό το μέτρο σύγκρισης των διαφορετικών μοντέλων.

Πίνακας 3.6: Σύνολα δεδομένων

Όνομα συνόλου	Πλήθος δειγμάτων
Εκπαίδευσης (training)	46.963
Επικύρωσης (validation)	5.870
Ελέγχου (test)	5.871

3.9 Δημιουργία διαδικασίας για το πέρασμα των δεδομένων στο μοντέλο

Έπρεπε να δημιουργηθεί μία διαδικασία για να τροφοδοτείται το μοντέλο αποδοτικά με δεδομένα κατά τη διάρκεια της εκπαίδευσης αλλά και στο στάδιο της αξιολόγησης. Η προγραμματιστική πλατφόρμα (framework) που χρησιμοποιήθηκε για τη δημιουργία των μοντέλων είναι το keras ([29]), το οποίο βασίζεται στο tensorflow ([30]). Δεν υπήρχε κάποια έτοιμη υλοποίηση που μπορούσε να χρησιμοποιηθεί αυτούσια στη διαδικασία. Για αυτό το λόγο έγινε τροποποίηση του κώδικα ανοιχτού λογισμικού του keras (κλάση ImageDataGenerator).

Η επεξεργασία του κώδικα ήταν δύσκολη εξαιτίας της αυξημένης πολυπλοκότητας. Δημιουργήθηκαν δύο γεννήτριες δεδομένων, μία για το πρόβλημα παλινδρόμησης (regression) και μία για το πρόβλημα ταξινόμησης (classification). Οι

γεννήτριες έπρεπε σε κάθε εποχή εκπαίδευσης να επιστρέφουν όλα τα δεδομένα εκπαίδευσης σε υποσύνολα συγκεκριμένου μεγέθους (batch size). Σε κάθε βήμα εκπαίδευσης τα υποσύνολα έπρεπε να δημιουργούνται με τυχαίο τρόπο. Αντίστοιχα στο στάδιο της επικύρωσης (validation) και της αξιολόγησης (test) του μοντέλου έπρεπε να επιστρέφονται όλα τα δεδομένα των αντίστοιχων συνόλων σε υποσύνολα συγκεκριμένου μεγέθους (batch size).

Η γεννήτρια που δημιουργήθηκε για το πρόβλημα παλινδρόμησης κάθε φορά επέστρεφε υποσύνολα δεδομένων που περιλάμβαναν τα μονοπάτια των εικόνων (paths) και τις αντίστοιχες ετικέτες της διέγερσης (arousal) και του σθένους (valence). Η γεννήτρια που χρησιμοποιήθηκε στο πρόβλημα ταξινόμησης (classification) έκανε ένα επιπλέον βήμα επεξεργασίας καθώς αντί για τις τιμές των ετικετών επέστρεφε τις κλάσεις (συναισθηματικές κατηγορίες). Με βάση τις τιμές των ετικετών (σύστημα συντεταγμένων σχήματος 1.3) ορίζονταν οι αντίστοιχες τέσσερις κλάσεις. Η κωδικοποίηση των κλάσεων έγινε με τη τεχνική ‘one hot encoding’ ([24]), έτσι ώστε να τροφοδοτηθεί ορθά το μοντέλο με δεδομένα.

Πίνακας 3.7: Παραδείγματα μορφής κλάσεων

Δείγμα	**Συναίσθημα**	**Τιμή Διέγερσης (arousal)**	**Τιμή Σθένους (valence)**	**Κλάση**
1	Χαρά	Θετική	Θετική	1
2	Θυμός	Θετική	Αρνητική	2
3	Λύπη	Αρνητική	Αρνητική	3
4	Χαλάρωση	Αρνητική	Θετική	4

Πίνακας 3.8: Παραδείγματα κωδικοποίησης κλάσεων ‘one hot encoding’

Δείγμα	**Κλάση 1 (1ο τεταρτημόριο)**	**Κλάση 2 (2ο τεταρτημόριο)**	**Κλάση 3 (3ο τεταρτημόριο)**	**Κλάση 4 (4ο τεταρτημόριο)**
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

3.10 Δημιουργία μοντέλων κατηγοριοποίησης και παλινδρόμησης – Επισκόπηση αρχιτεκτονικής

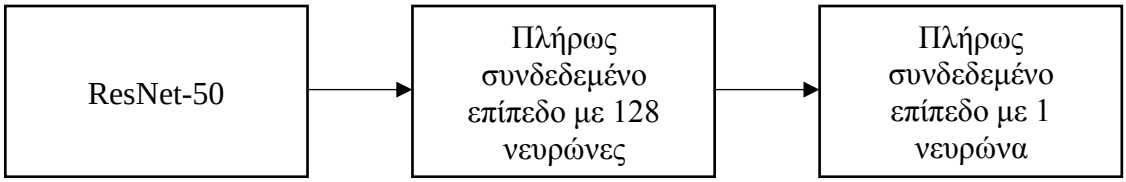
Για τη δημιουργία των μοντέλων χρησιμοποιήθηκε η τεχνική της μεταφοράς μάθησης (transfer learning). Εξετάστηκαν δύο προεκπαιδευμένα (pretrained) βαθιά νευρωνικά δίκτυα, το ‘ResNet-50’ και το ‘Inception Resnet V2’. Τα αρχικά βάρη των δύο αυτών δικτύων είχαν καθοριστεί μετά από εκπαίδευση με χρήση των δεδομένων του ‘ImageNet’ ([37]).

Το ‘ImageNet’ είναι μια μεγάλη οπτική βάση δεδομένων σχεδιασμένη για χρήση στην έρευνα αναγνώρισης οπτικών αντικειμένων. Πάνω από 14 εκατομμύρια εικόνες έχουν επισημανθεί για να υποδείξουν τα αντικείμενα που απεικονίζονται. Το ‘ImageNet’ περιέχει πάνω από 20 χιλιάδες κατηγορίες, μία τυπική κατηγορία, όπως «μπαλόνι» ή «φράουλα», περιέχει αρκετές εκατοντάδες εικόνες.

Από το 2010, διοργανώνεται ετήσιος διαγωνισμός αναγνώρισης αντικειμένων που βρίσκονται στο ‘ImageNet’, το ‘ImageNet Large Scale Visual Recognition Challenge’ (ILSVRC). Τα μοντέλα ‘ResNet-50’ και ‘Inception Resnet V2’ έχουν κερδίσει στο παρελθόν το διαγωνισμό.

3.10.1 Μοντέλα παλινδρόμησης

Εξετάστηκαν δύο διαφορετικές αρχιτεκτονικές. Η διαφορά τους ήταν στο προεκπαιδευμένο βαθύ νευρωνικό δίκτυο που χρησιμοποιήθηκε (ResNet-50, Inception-ResNet-V2). Οι δύο αρχιτεκτονικές παρουσιάζονται στα σχήματα που ακολουθούν.



Σχήμα 3.4: Πρώτο μοντέλο παλινδρόμησης

Στοιχεία του επιπέδου ResNet-50:

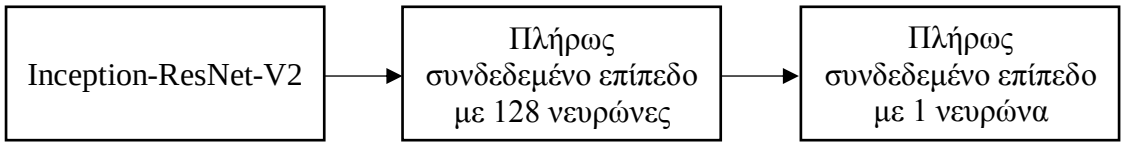
- Το τελευταίο επίπεδο του δικτύου αφαιρέθηκε. Όλη η υπόλοιπη λειτουργικότητά του χρησιμοποιήθηκε.
- Τα βάρη του δικτύου προέκυψαν μετά από εκπαίδευση του ‘ResNet’ με χρήση των δεδομένων του ‘ImageNet’.
- Οι διαστάσεις δεδομένων εισόδου ήταν 200 x 200 x 3. Οι εικόνες ήταν σε ασπρόμαυρη μορφή, αλλά το δίκτυο είχε ως περιορισμό να δέχεται εικόνες με τρία κανάλια. Για το λόγο αυτό το ασπρόμαυρο κανάλι αντιγράφηκε δύο φορές.
- Στα συγκεντρωτικά επίπεδα του δικτύου επιλέχθηκε να υπολογίζεται ο μέσος όρος των στοιχείων που περιλαμβάνονται στο κινούμενο παράθυρο.

Στοιχεία του πρώτου πλήρως συνδεδεμένου επιπέδου:

- Πλήθος νευρώνων: 128
- Συνάρτηση ενεργοποίησης: ReLU
- Το επίπεδο αυτό χρησιμοποιήθηκε έτσι ώστε να είναι πιο ομαλή η μετάβαση από την έξοδο του προτελευταίου επιπέδου του ‘ResNet’ στο τελευταίο επίπεδο της ολικής αρχιτεκτονικής όπου εξάγεται η τελική πρόβλεψη.

Στοιχεία του δεύτερου πλήρως συνδεδεμένου επιπέδου:

- Πλήθος νευρώνων: 1
- Συνάρτηση ενεργοποίησης: γραμμική (linear)
- Το επίπεδο αυτό χρησιμοποιήθηκε για να εξάγεται η τελική πρόβλεψη του μοντέλου.



Σχήμα 3.5: Δεύτερο μοντέλο παλινδρόμησης

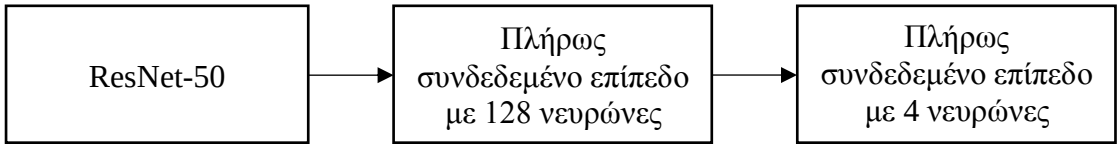
Η δεύτερη αρχιτεκτονική ήταν πανομοιότυπη με την πρώτη. Το μόνο που άλλαξε ήταν ότι χρησιμοποιήθηκε το 'Inception-ResNet-V2' με τις ίδιες προδιαγραφές με αυτές του ResNet-50. Τα υπόλοιπα στοιχεία ήταν ίδια, όπως περιγράφηκαν προηγουμένως.

Πίνακας 3.9: Παράμετροι για κάθε μοντέλο παλινδρόμησης

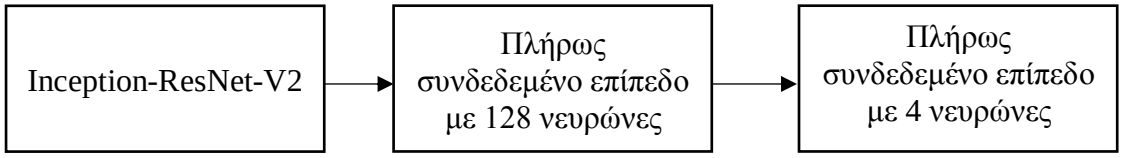
Μοντέλο	**Προεκπαιδευμένο δίκτυο**	**Συνολικές παράμετροι**	**Εκπαιδεύσιμες παράμετροι**	**Μη εκπαιδεύσιμες παράμετροι**
1ο	ResNet-50	23.850.113	23.796.993	53.120
2ο	Inception-ResNet-V2	54.533.601	54.473.057	60.544

3.10.2 Μοντέλα κατηγοριοποίησης

Αντίστοιχα με τα μοντέλα παλινδρόμησης, εξετάστηκαν δύο διαφορετικές αρχιτεκτονικές. Η διαφορά τους ήταν στο προεκπαιδευμένο βαθύ νευρωνικό δίκτυο που χρησιμοποιήθηκε (ResNet-50, Inception-ResNet-V2). Οι δύο αρχιτεκτονικές παρουσιάζονται στα σχήματα που ακολουθούν.



Σχήμα 3.6: Πρώτο μοντέλο κατηγοριοποίησης



Σχήμα 3.7: Δεύτερο μοντέλο κατηγοριοποίησης

Τα επίπεδα των προεκπαιδευμένων δικτύων που παρουσιάζονται στα δύο σχήματα ήταν ίδια με αυτά των μοντέλων παλινδρόμησης. Τα εναπομείναντα δύο επίπεδα ήταν ίδια και για τις δύο αρχιτεκτονικές, τα στοιχεία τους παρουσιάζονται στη συνέχεια.

Στοιχεία πρώτου πλήρως συνδεδεμένου επιπέδου:

- Πλήθος νευρώνων: 128
- Συνάρτηση ενεργοποίησης: ReLU
- Το επίπεδο αυτό χρησιμοποιήθηκε, έτσι ώστε να είναι πιο ομαλή η μετάβαση από την έξοδο του προτελευταίου επιπέδου του προεκπαιδευμένου δικτύου στο τελευταίο επίπεδο της ολικής αρχιτεκτονικής, όπου εξάγεται η τελική πρόβλεψη.

Στοιχεία δεύτερου πλήρως συνδεδεμένου επιπέδου:

- Πλήθος νευρώνων: 4
- Συνάρτηση ενεργοποίησης: Softmax
- Το επίπεδο αυτό χρησιμοποιήθηκε για να εξάγεται η τελική πρόβλεψη του μοντέλου (πιθανότητα για κάθε μία από τις τέσσερις κλάσεις).

Πίνακας 3.10: Παράμετροι για κάθε μοντέλο κατηγοριοποίησης

Μοντέλο	**Προεκπαιδευμένο δίκτυο**	**Συνολικές παράμετροι**	**Εκπαιδεύσιμες παράμετροι**	**Μη εκπαιδεύσιμες παράμετροι**
1ο	ResNet-50	23.850.500	23.797.380	53.120
2ο	Inception-ResNet-V2	54.533.988	54.473.444	60.544

3.10.3 Προσθήκη ομαλοποίησης (regularization)

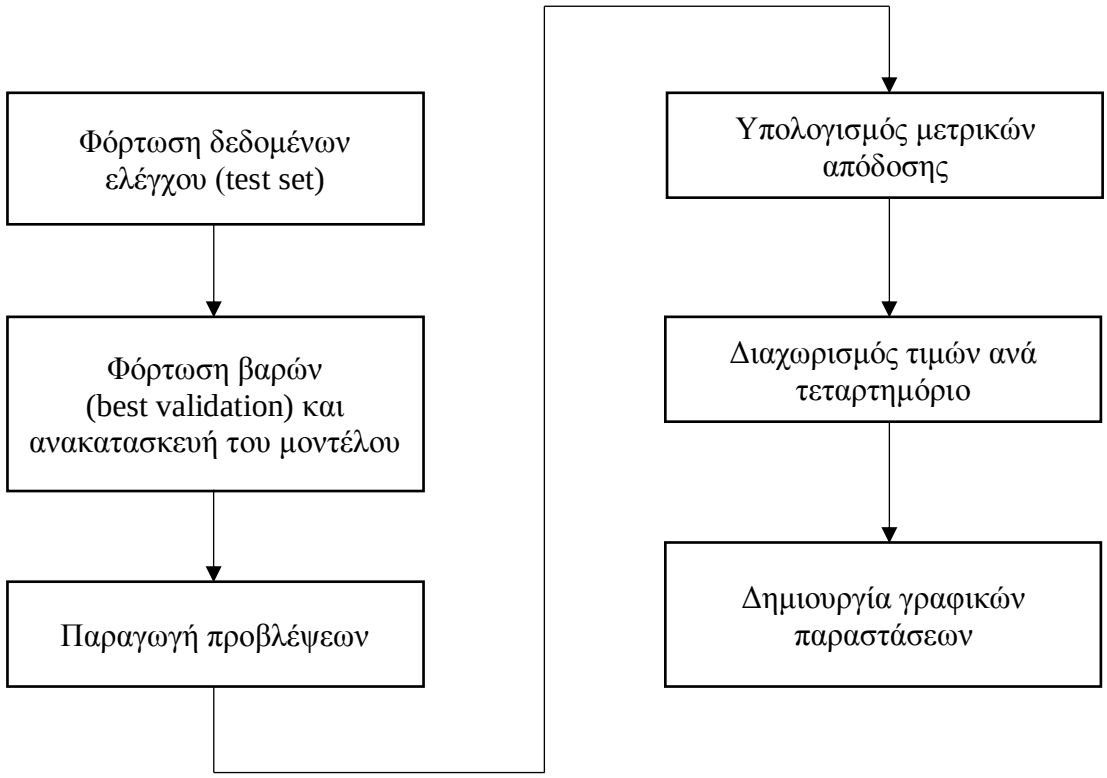
Σε μία δεύτερη σειρά πειραμάτων προστέθηκε στο δεύτερο επίπεδο των τεσσάρων αρχιτεκτονικών που περιγράφηκαν ομαλοποίηση L2. Η τιμή της παραμέτρου ποινής λ ήταν 0.1. Η προσθήκη αυτή έγινε προκειμένου να εξεταστεί αν η ομαλοποίηση στο συγκεκριμένο σημείο της αρχιτεκτονικής επηρεάζει θετικά τη συνολική επίδοση του μοντέλου.

3.10.4 Εκπαίδευση μοντέλων

Η διαδικασία εκπαίδευσης των μοντέλων διέφερε ανάλογα με τον τύπο του προβλήματος. Η εκπαίδευση των νευρωνικών δικτύων περιγράφεται αναλυτικά στο τέταρτο κεφάλαιο της εργασίας.

3.11 Αξιολόγηση μοντέλων

Το τελευταίο στάδιο των πειραμάτων αφορούσε την αξιολόγηση των μοντέλων. Αυτή η διαδικασία ήταν πολύ σημαντική καθώς μέσω αυτής έγινε η σύγκριση μεταξύ των μοντέλων. Επίσης στην αξιολόγηση των μοντέλων δίνονται διάφορα στοιχεία που μπορούν να οδηγήσουν σε περαιτέρω βελτιώσεις της αρχιτεκτονικής και της εκπαίδευσης των δικτύων. Σε γενικές γραμμές, αρχικά εξάγονταν οι προβλέψεις των μοντέλων. Έπειτα, υπολογίζονταν διάφορες μετρικές που υποδείκνυαν πόσο αποδοτικό ήταν το μοντέλο. Τέλος, δημιουργούνταν γραφικές παραστάσεις για την οπτική απεικόνιση της απόδοσης των μοντέλων. Το σχήμα, που ακολουθεί, περιγράφει την όλη διαδικασία που εκτελέστηκε στα μοντέλα παλινδρόμησης και κατηγοριοποίησης. Ορισμένα επιμέρους στάδια διαφοροποιούνταν ανάλογα με το τύπο προβλήματος (παλινδρόμηση ή κατηγοριοποίηση).



Σχήμα 3.8: Διαδικασία αξιολόγησης

3.11.1 Φόρτωση δεδομένων ελέγχου (test set)

Η αξιολόγηση όλων των μοντέλων έγινε με χρήση δεδομένων που δεν είχαν χρησιμοποιηθεί στην εκπαίδευση. Το σύνολο δεδομένων ελέγχου (test set) ήταν το ίδιο για όλα τα μοντέλα. Η φόρτωση του αποτέλεσε το πρώτο στάδιο της διαδικασίας αξιολόγησης.

3.11.2 Φόρτωση βαρών και ανακατασκευή του μοντέλου

Κατά τη διάρκεια της εκπαίδευσης του κάθε μοντέλου αποθηκεύτηκαν τα βάρη που έδιναν τη βέλτιστη απόδοση στο σύνολο δεδομένων επικύρωσης (validation set). Αυτά τα βάρη φορτώθηκαν ώστε να γίνει η ανακατασκευή του κάθε μοντέλου.

3.11.3 Παραγωγή προβλέψεων

Εξαιτίας των περιορισμένων υπολογιστικών πόρων έπρεπε να δημιουργηθεί μία διαδικασία έτσι ώστε να παράγονται οι προβλέψεις κατά ομάδες (batches). Σε κάθε εκτέλεση της διαδικασίας εξάγονταν 32 (batch size) προβλέψεις μέχρι να χρησιμοποιηθεί όλο το σύνολο ελέγχου (5871 δείγματα).

Στα μοντέλα παλινδρόμησης για κάθε εικόνα του συνόλου ελέγχου παράγονταν δύο τιμές, μία για το σθένος (valence) και μία για τη διέγερση (arousal). Επειδή όλες οι ετικέτες των εικόνων είχαν ακρίβεια 4 δεκαδικά ψηφία, έγινε στρογγυλοποίηση των προβλέψεων. Εν τέλει οι προβλέψεις κάθε μοντέλου αποθηκεύτηκαν σε .txt αρχεία.

Στα μοντέλα κατηγοριοποίησης για κάθε εικόνα του συνόλου ελέγχου παράγονταν 4 τιμές, που υποδήλωναν την πιθανότητα να βρίσκεται η εικόνα στη κλάση που αντιστοιχεί σε κάθε τεταρτημόριο. Προκειμένου να υπολογιστούν οι διάφορες μετρικές απόδοσης έπρεπε να γίνει μετατροπή των εξαγόμενων πιθανοτήτων κάθε κλάσης σε κωδικοποίηση ‘one hot encoding’. Έγινε λοιπόν αυτή η μετατροπή και οι προβλέψεις αποθηκεύτηκαν σε .txt αρχεία.

3.11.4 Υπολογισμός μετρικών απόδοσης

Το στάδιο αυτό διέφερε ανάλογα με τον τύπο του προβλήματος (παλινδρόμηση ή κατηγοριοποίηση). Σε κάθε περίπτωση όμως υπολογίστηκαν μετρικές με χρήση των πραγματικών ετικετών των εικόνων και των εξαγόμενων προβλέψεων, έτσι ώστε να παρουσιαστεί η αποδοτικότητα των μοντέλων.

3.11.4.1 Μετρικές μοντέλων παλινδρόμησης

Για τα μοντέλα παλινδρόμησης υπολογίστηκαν δύο μετρικές, το μέσο τετραγωνικό σφάλμα και το μέσο απόλυτο σφάλμα. Αξίζει να σημειωθεί ότι οι μετρικές υπολογίστηκαν ξεχωριστά για τη πρόβλεψη της διέγερσης (arousal) και αυτή του σθένους (valence). Οι τύποι των δύο μετρικών είναι οι ακόλουθοι:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y'_i)^2 \qquad (9)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - Y'_i| \qquad (10)$$

Όπου:

- MSE: μέσο τετραγωνικό σφάλμα
- MAE: μέσο απόλυτο σφάλμα
- n: πλήθος δειγμάτων συνόλου ελέγχου
- i: δείκτης
- Y_i: πραγματική τιμή – ετικέτα
- Y'_i: πρόβλεψη

3.11.4.2 Μετρικές μοντέλων κατηγοριοποίησης

Για τα μοντέλα κατηγοριοποίησης υπολογίστηκαν περισσότερες μετρικές. Οι μετρικές ήταν δύο τύποι ακρίβειας (precision και accuracy), η ανάκληση (recall), το 'f-1' αποτέλεσμα (f-1 score) και ο πίνακας σύγχυσης (confusion matrix).

Η ακρίβεια (precision) είναι διαισθητικά η ικανότητα του ταξινομητή να μην χαρακτηρίζει ως θετικό ένα δείγμα που είναι αρνητικό, δηλαδή να μην κατηγοριοποιεί το δείγμα σε μία κλάση ενώ δεν ανήκει σε αυτή. Ο άλλος τύπος ορθότητας (accuracy) υπολογίζει πόσα δείγματα κατηγοριοποιήθηκαν σωστά. Η ανάκληση (recall) είναι διαισθητικά η ικανότητα του ταξινομητή να βρει όλα τα θετικά δείγματα, δηλαδή κάθε δείγμα να κατηγοριοποιείται στη σωστή κλάση. Το 'f-1' αποτέλεσμα (f-1 score) είναι μία μετρική που λαμβάνει υπόψη και την ακρίβεια (precision) και την ανάκληση (recall). Τέλος ο πίνακας σύγχυσης (confusion matrix) δίνει μία συνολική λεπτομερή εικόνα της απόδοσης του ταξινομητή καθώς παρουσιάζει για κάθε κλάση πόσα δείγματα κατηγοριοποιήθηκαν σωστά και πόσα λάθος.

Για να οριστούν όλες αυτές οι μετρικές είναι σημαντικό να εξηγηθεί τι σημαίνουν οι όροι «αληθώς θετικό» (true positive – TP), «ψευδώς θετικό» (false positive - FP), «αληθώς αρνητικό» (true negative - TN) και «ψευδώς αρνητικό» (false negative - FN). Ο όρος «αληθώς θετικό» δηλώνει ότι το δείγμα κατηγοριοποιήθηκε ορθά στην κλάση ενώ «ψευδώς θετικό» σημαίνει ότι τοποθετήθηκε λανθασμένα. Ο όρος «αληθώς αρνητικό» δηλώνει ότι ορθά το δείγμα δεν κατηγοριοποιήθηκε στην κλάση, ενώ «ψευδώς αρνητικό» σημαίνει ότι έπρεπε να κατηγοριοποιηθεί στη συγκεκριμένη κλάση. Πλέον, μετά από τις παραπάνω επεξηγήσεις, μπορούν να οριστούν οι τύποι όλων των μετρικών.

$$Precision = \frac{TP}{TP + FP} \qquad (11)$$

$$Recall = \frac{TP}{TP + FN} \qquad (12)$$

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \qquad (13)$$

$$F1_{SCORE} = \frac{2 * Precision * Recall}{Precision + Recall} \qquad (14)$$

Πίνακας 3.11: Πίνακας σύγχυσης (confusion matrix)

		Πρόβλεψη (κλάση)	
		Θετική (P)	Αρνητική (N)
Πραγματική κλάση	Θετική (P)	TP	FN
	Αρνητική (N)	FP	TN

Όπου:

- TP: πλήθος «αληθώς θετικών» δειγμάτων
- FP: πλήθος «ψευδώς θετικών» δειγμάτων
- TN: πλήθος «αληθώς αρνητικών» δειγμάτων
- FP: πλήθος «ψευδώς αρνητικών» δειγμάτων

3.11.5 Διαχωρισμός τιμών ανά τεταρτημόριο

Ο διαχωρισμός των πραγματικών τιμών και των προβλέψεων ανά τεταρτημόριο ήταν ένα σημαντικό επιμέρους στάδιο καθώς καταδείκνυε πόσο αποδοτικό ήταν το μοντέλο. Ήταν καίριο να βρεθεί πόσα δείγματα είχαν τοποθετηθεί σε λανθασμένο τεταρτημόριο γιατί κάθε τεταρτημόριο εκφράζει μία διαφορετική κατηγορία συναισθήματος (χαρά, θυμός, λύπη και χαλάρωση). Αυτή η μελέτη έγινε και στους δύο τύπος προβλημάτων (παλινδρόμηση και κατηγοριοποίηση).

3.11.6 Δημιουργία γραφικών παραστάσεων

Η οπτικοποίηση των αποτελεσμάτων βοήθησε στην γενική κατανόηση της απόδοσης των διαφόρων μοντέλων. Δημιουργήθηκαν γραφικές παραστάσεις που παρουσιάζουν την κατανομή των πραγματικών τιμών και των προβλέψεων συνολικά αλλά και ανά τεταρτημόριο. Μόνο στα μοντέλα παλινδρόμησης παρουσιάστηκαν γραφικές παραστάσεις, καθώς σε αυτά τα μοντέλα γίνονταν ακριβείς προβλέψεις των τιμών της διέγερσης (arousal) και του σθένους (valence). Στα μοντέλα κατηγοριοποίησης η πρόβλεψη ήταν μία από τις τέσσερις κλάσεις συναισθημάτων.

3.12 Προγραμματιστικές πλατφόρμες και εργαλεία

Ο κώδικας για την υλοποίηση όλης της εργασίας γράφτηκε στην γλώσσα προγραμματισμού Python στο προγραμματιστικό περιβάλλον Jupyter Notebook, το οποίο είναι πολύ εύχρηστο για την υλοποίηση των πειραμάτων. Χρησιμοποιήθηκαν οι ακόλουθες βιβλιοθήκες:

- Pandas ([28]) : διαχείριση και επεξεργασία δεδομένων
- Numpy ([27]) : υπολογιστικές δυνατότητες και δομές
- Matplotlib ([31]) : δημιουργία γραφικών παραστάσεων
- Skimage ([32]) : επεξεργασία και διαχείριση εικόνων
- Sklearn ([32]) : μετρικές απόδοσης και εργαλεία μηχανικής μάθησης
- OpenCV ([33]) : επεξεργασία βίντεο

- Tensorflow ([30]) : ανοιχτού κώδικα (*open source*) προγραμματιστική βιβλιοθήκη για αριθμητικούς υπολογισμούς και ανάπτυξη εφαρμογών μηχανικής μάθησης με χρήση γράφων ροής δεδομένων *(data flow graphs)*
- Keras ([29]) : Υψηλού επιπέδου διεπαφή προγραμμισμού εφαρμογών (API) για ανάπτυξη νευρωνικών δικτύων που χρησιμοποιεί τη λειτουργικότητα του Tensorflow.

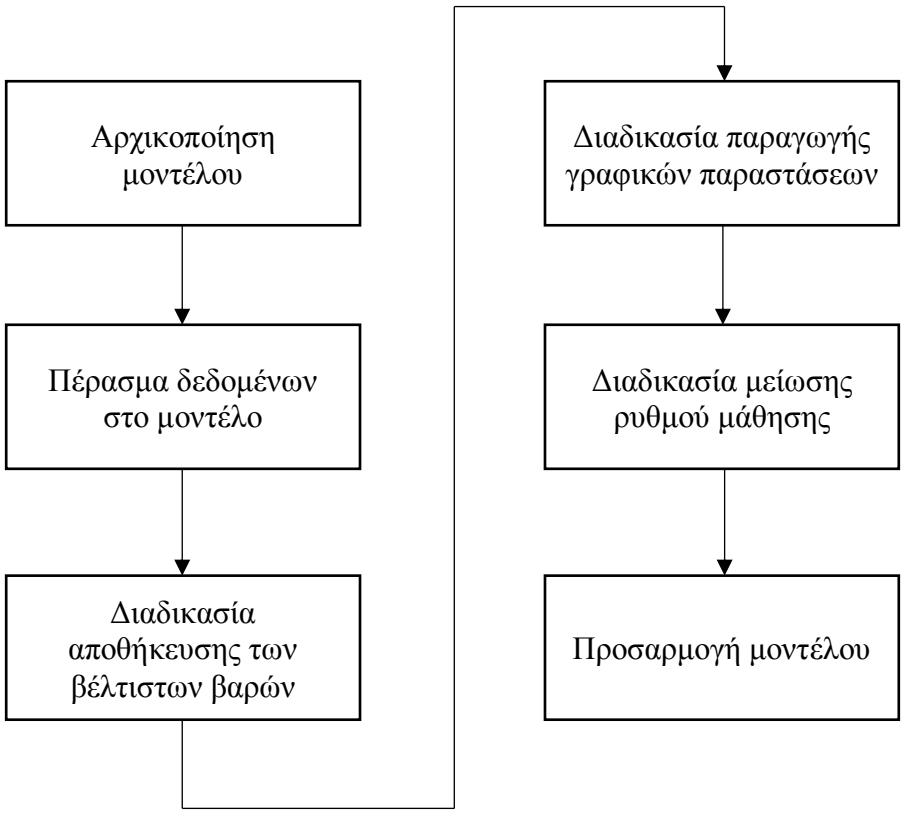
Η επεξεργασία των δεδομένων, η υλοποίηση των μοντέλων και κυρίως η εκπαίδευσή τους αποτελούν εργασίες που απαιτούν πολλούς υπολογιστικούς πόρους. Για το λόγο αυτό χρησιμοποιήθηκε μία κάρτα γραφικών (GPU) Nvidia GeForce GTX 1070 με 8GB μνήμη σε συνδυασμό με το λογισμικό 'CUDA' της Nvidia. Το 'NVIDIA CUDA Toolkit' παρέχει ένα αναπτυξιακό περιβάλλον για τη δημιουργία υψηλής απόδοσης GPU εφαρμογών. Οι βιβλιοθήκες CUDA επιτρέπουν την υπολογιστική επιτάχυνση σε πολλαπλούς τομείς, όπως η επεξεργασία εικόνων και βίντεο και η βαθιά μηχανική μάθησης (deep learning).

4

Εκπαίδευση μοντέλων και Αποτελέσματα

4.1 Προετοιμασία και διαδικασία εκπαίδευσης μοντέλων

Η γενική διαδικασία εκπαίδευσης ήταν η ίδια για όλα τα μοντέλα. Οι υπερπαράμετροι άλλαζαν ανάλογα με τον τύπο του προβλήματος (παλινδρόμηση και κατηγοριοποίηση). Το διάγραμμα που ακολουθεί περιγράφει σε υψηλό επίπεδο την πορεία εκπαίδευσης των δικτύων. Τα επιμέρους στάδια αναλύονται αναλυτικά στη συνέχεια.



Σχήμα 4.1: Περιγραφή γενικής διαδικασίας εκπαίδευσης

4.1.1 Αρχικοποίηση μοντέλου

Στο στάδιο αυτό αρχικά ορίστηκε η αρχιτεκτονική του νευρωνικού δικτύου. Όλες οι αρχιτεκτονικές που εξετάστηκαν περιγράφηκαν στο υποκεφάλαιο 3.10. Στα μοντέλα παλινδρόμησης χρησιμοποιήθηκε ως συνάρτηση σφάλματος (loss function) αυτή του μέσου τετραγωνικού σφάλματος (Mean Squared Error) ενώ σε κάθε βήμα εκπαίδευσης υπολογίζονταν επιπροσθέτως το μέσο απόλυτο σφάλμα (Mean Absolute Error) και το μέσο ποσοστιαίο απόλυτο σφάλμα (Mean Absolute Percentage Error). Στα μοντέλα κατηγοριοποίησης χρησιμοποιήθηκε η κατηγορική εντροπία (categorical

cross entropy) ως συνάρτηση σφάλματος ενώ επίσης υπολογίζονταν η κατηγορική ακρίβεια (categorical accuracy), ένας άλλος τύπος ακρίβειας (precision), η ανάκληση (recall) και το αποτέλεσμα 'f-1' (f-1 score). Ως βελτιστοποιητής (optimizer) της εκπαίδευσης όλων των μοντέλα χρησιμοποιήθηκε ο 'Adam'.

4.1.2 Πέρασμα δεδομένων στο μοντέλο

Όπως περιγράφηκε στο υποκεφάλαιο 3.9, δημιουργήθηκαν γεννήτριες έτσι ώστε να επιτελείται το πέρασμα των δεδομένων στα μοντέλα. Στα πλαίσια της διαδικασίας εκπαίδευσης έπρεπε να κατασκευαστούν δύο γεννήτριες δεδομένων, μία για τα δεδομένα εκπαίδευσης (training set) και μία για τα δεδομένα επικύρωσης (validation set). Εξαιτίας των περιορισμένων υπολογιστικών πόρων το πέρασμα των δεδομένων έπρεπε να γίνεται κατά ομάδες (batches). Το μέγεθος αυτών τον ομάδων δεδομένων ήταν 32 δείγματα (batch size).

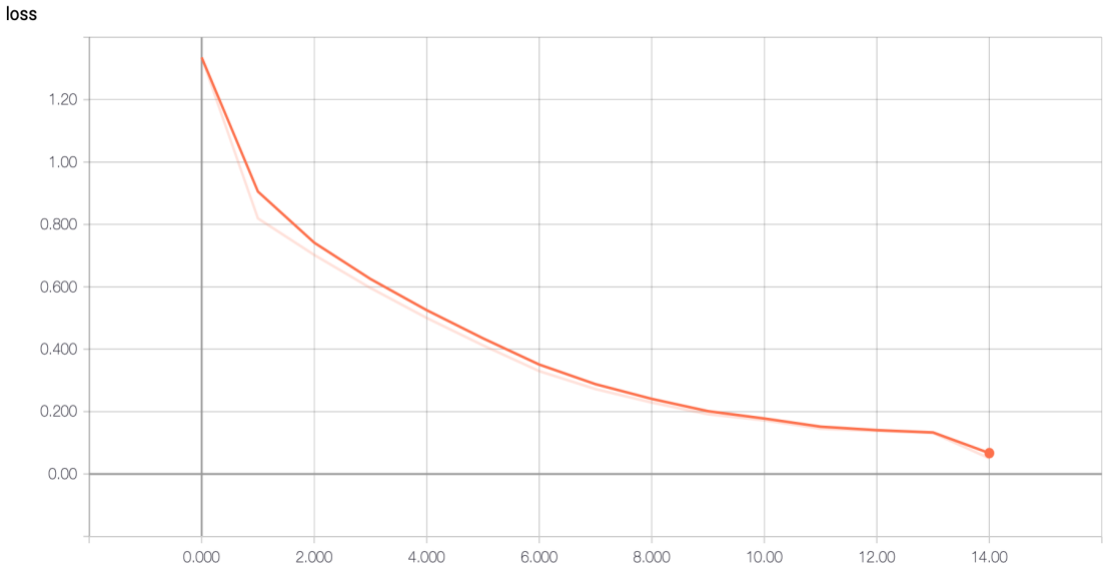
4.1.3 Διαδικασία αποθήκευσης των βέλτιστων βαρών

Κατά την εκτέλεση της εκπαίδευσης των μοντέλων, στο τέλος κάθε εποχής, γινόταν έλεγχος της επίδοσης στα δεδομένα επικύρωσης. Αυτό το σύνολο δεδομένων δε χρησιμοποιούταν κατά τη διάρκεια της εκπαίδευσης. Μετά από κάθε βήμα εκπαίδευσης γινόταν λοιπόν έλεγχος αν βελτιώνεται η απόδοση του μοντέλου και σε περίπτωση που αυτό γινόταν αποθηκεύονταν τα βάρη του μοντέλου. Έτσι, στο τέλος της διαδικασίας είχαν αποθηκευτεί τα βάρη που έδιναν το βέλτιστο αποτέλεσμα στο σύνολο επικύρωσης (validation set). Αυτή η προσέγγιση ακολουθήθηκε ώστε να αντιμετωπιστεί το πιθανό πρόβλημα της υπερπροσαρμογής του μοντέλου (overfitting). Για την υλοποίηση αυτής της προσέγγισης χρησιμοποιήθηκε η κλάση 'ModelCheckpoint' του 'Keras' ([29]).

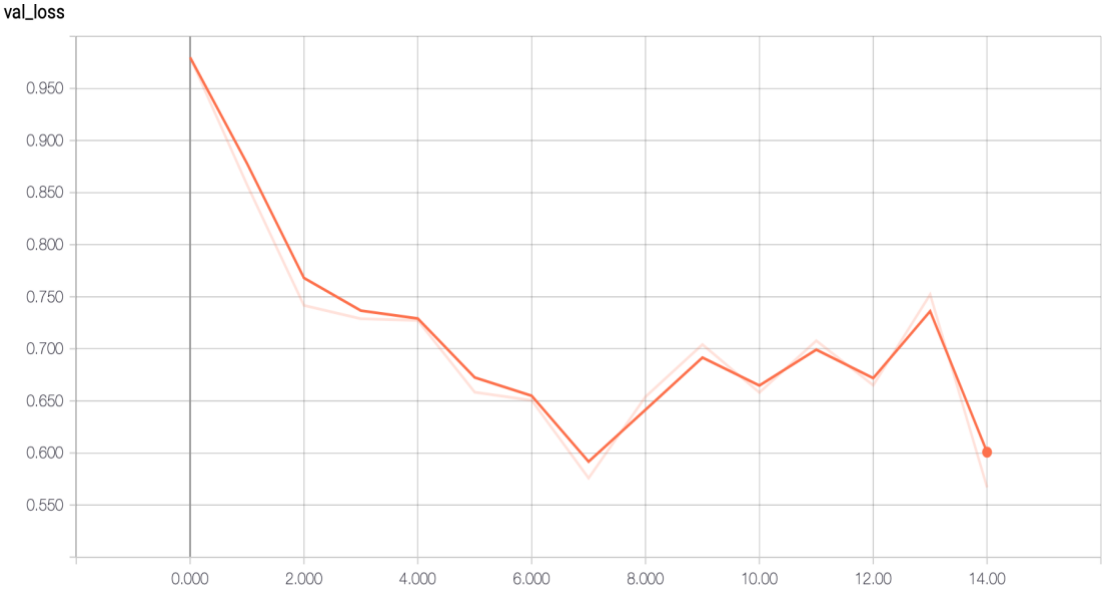
4.1.4 Διαδικασία παραγωγής γραφικών παραστάσεων

Για την αποδοτικότερη εκπαίδευση των μοντέλων ήταν σημαντική η δυναμική απεικόνιση διαφόρων μετρικών. Η συνεχής εποπτεία του συστήματος βοήθησε στην ορθή επιλογή υπερπαραμέτρων της εκπαίδευσης. Χρησιμοποιήθηκε το εύχρηστο εργαλείο του 'Tensorflow' [30]), το 'TensorBoard' ([26]).

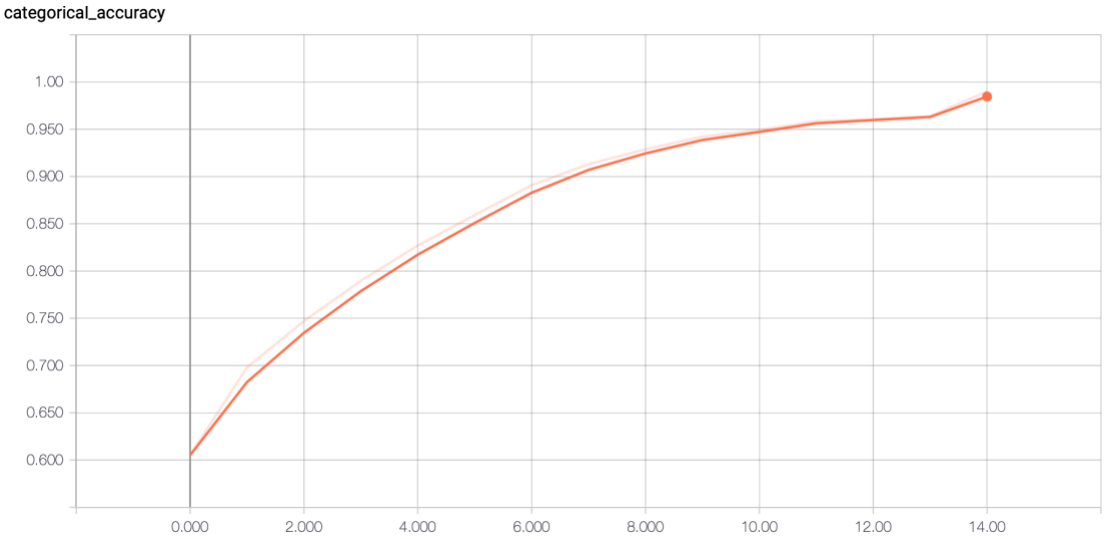
Ακολουθούν ορισμένα παραδείγματα. Η πρώτη ομάδα γραφικών παραστάσεων αφορά την εκπαίδευση του μοντέλου κατηγοριοποίησης που περιλάμβανε το InceptionResNet-V2 δίκτυο και είχε ομαλοποίηση (regularization). Η δεύτερη ομάδα γραφικών αφορά την αντίστοιχη αρχιτεκτονική που χρησιμοποιήθηκε για το πρόβλημα παλινδρόμησης. Με μπλε χρώμα παρουσιάζονται οι γραφικές που αναφέρονται στο σθένος (valence) ενώ με πορτοκαλί αυτές που αναφέρονται στη διέγερση (arousal). Σε όλες τις γραφικές παραστάσεις έχει εφαρμοστεί εξομάλυνση (smoothing) με τιμή 0.2. Ο x άξονας αναφέρεται στις εποχές εκπαίδευσης.



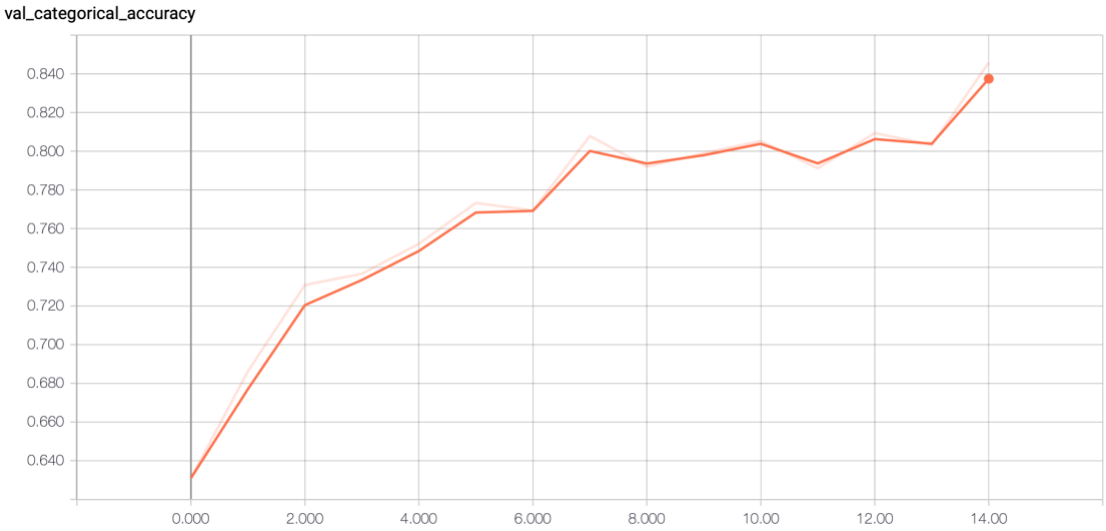
Σχήμα 4.2: Κατηγορική εντροπία στο σύνολο εκπαίδευσης



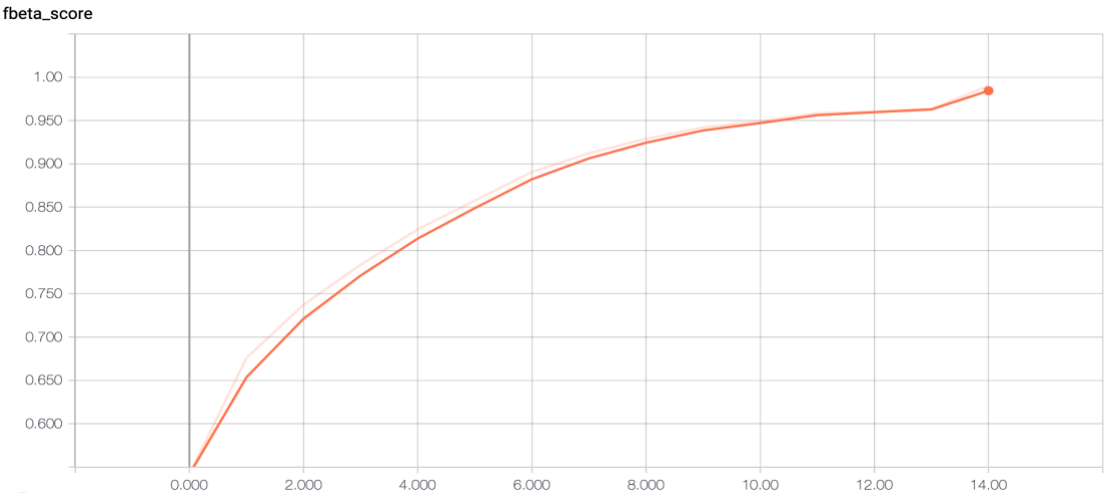
Σχήμα 4.3: Κατηγορική εντροπία στο σύνολο επικύρωσης



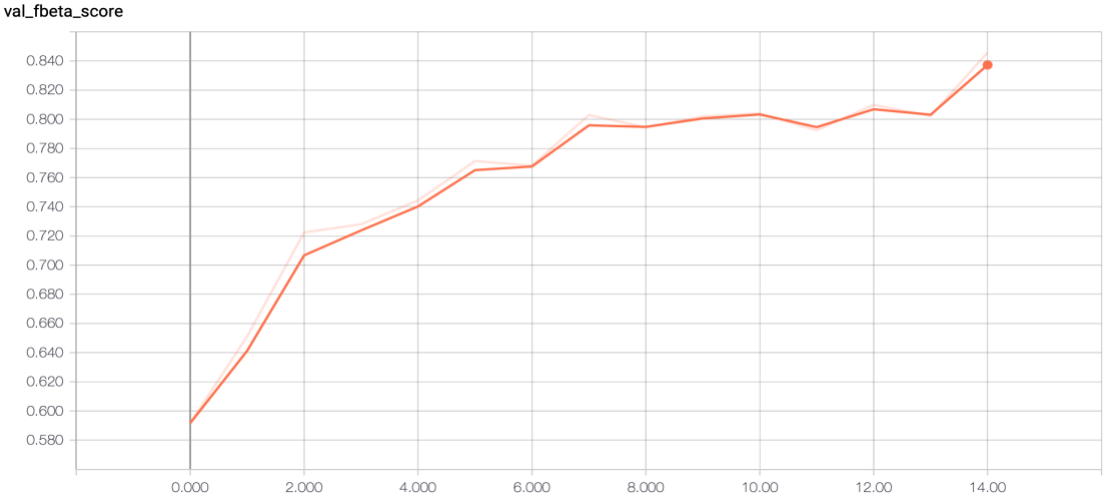
Σχήμα 4.4: Κατηγορική ακρίβεια στο σύνολο εκπαίδευσης



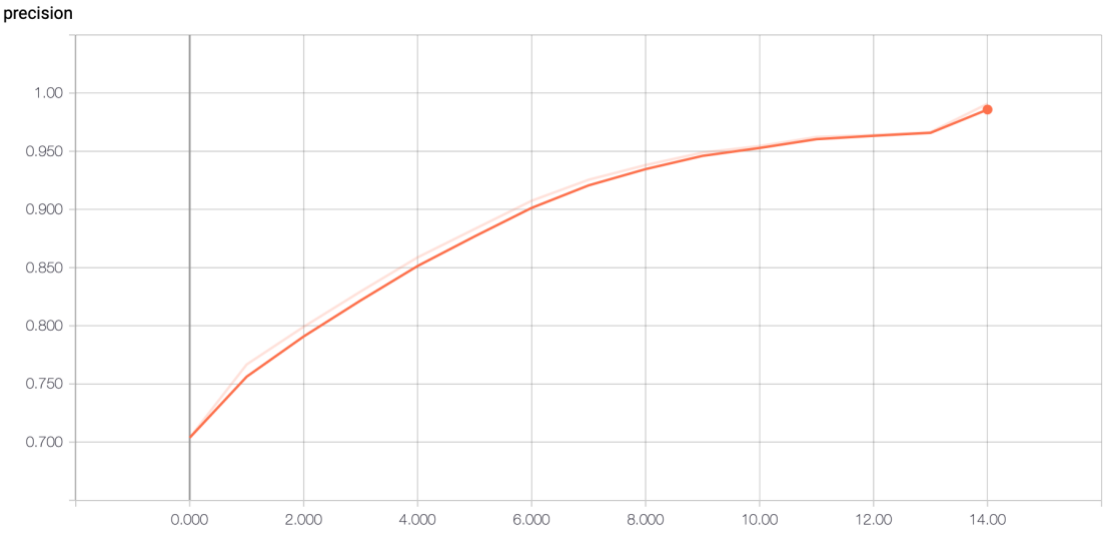
Σχήμα 4.5: Κατηγορική ακρίβεια στο σύνολο επικύρωσης



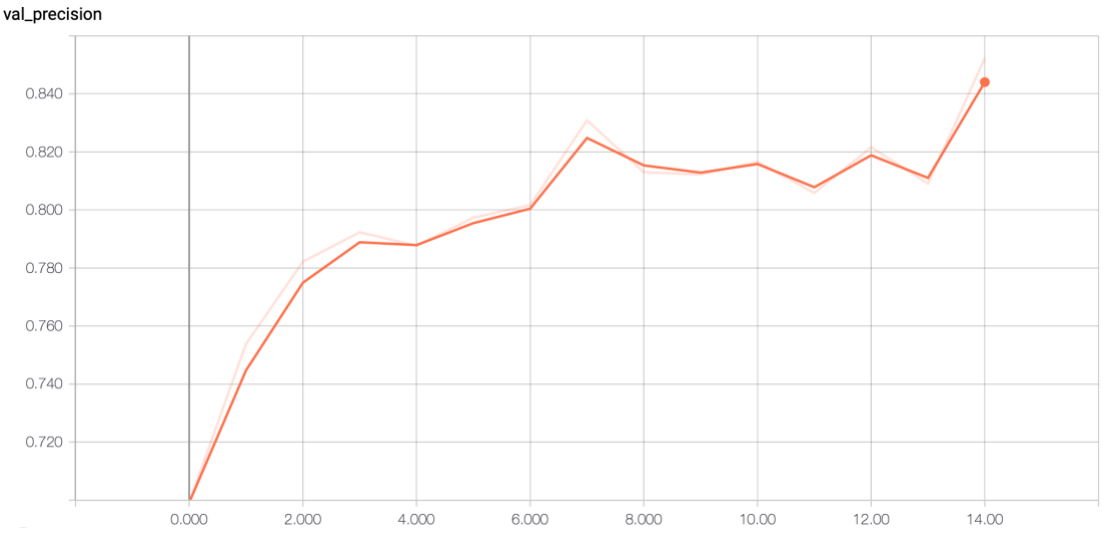
Σχήμα 4.6: 'f-1' αποτέλεσμα στο σύνολο εκπαίδευσης



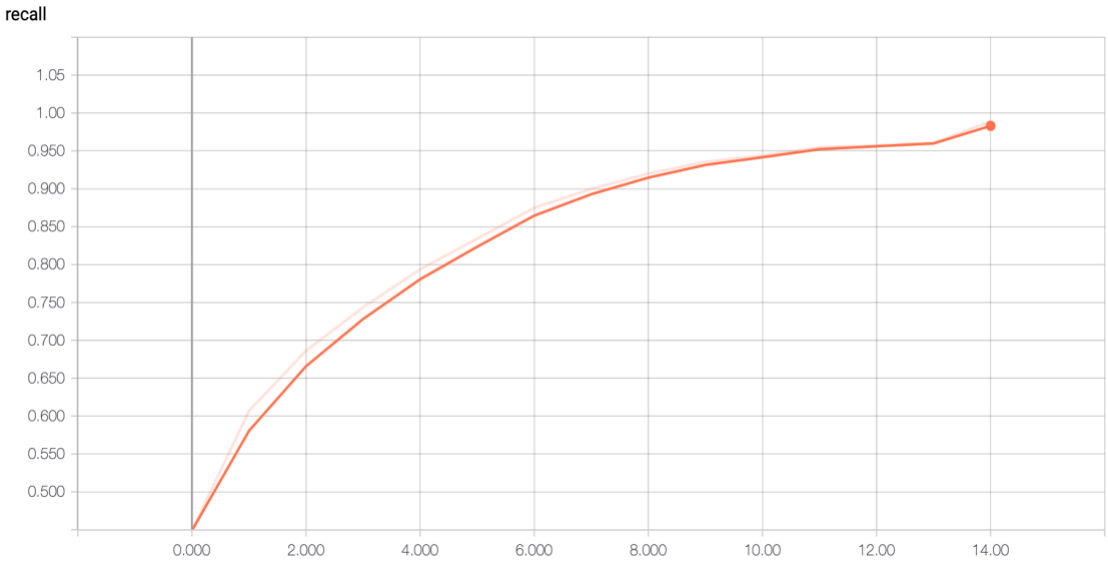
Σχήμα 4.7: 'f-1' αποτέλεσμα στο σύνολο επικύρωσης



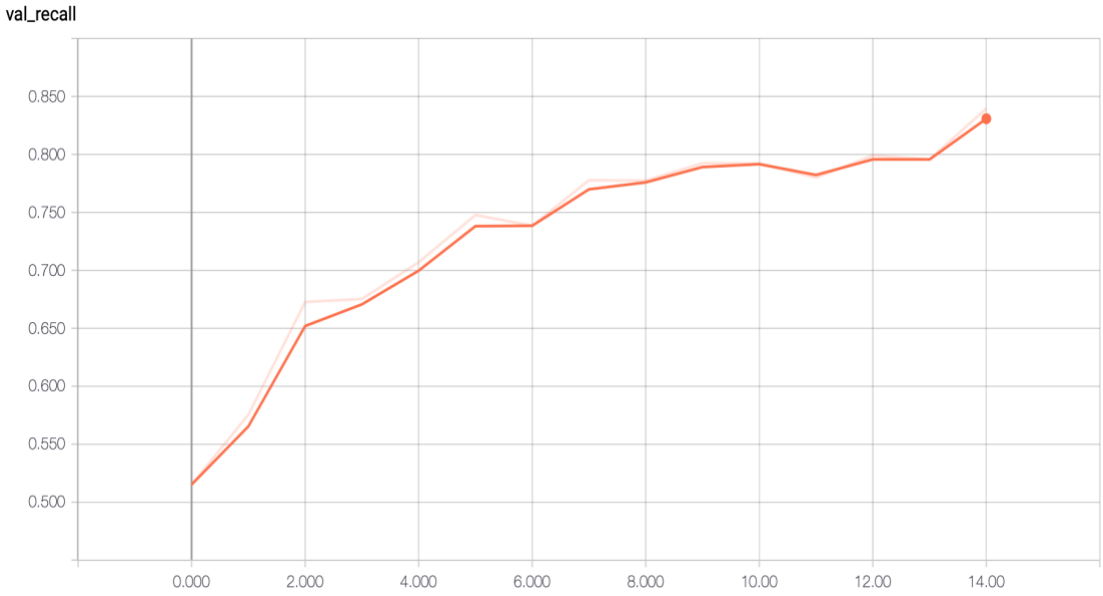
Σχήμα 4.8: Ακρίβεια στο σύνολο εκπαίδευσης



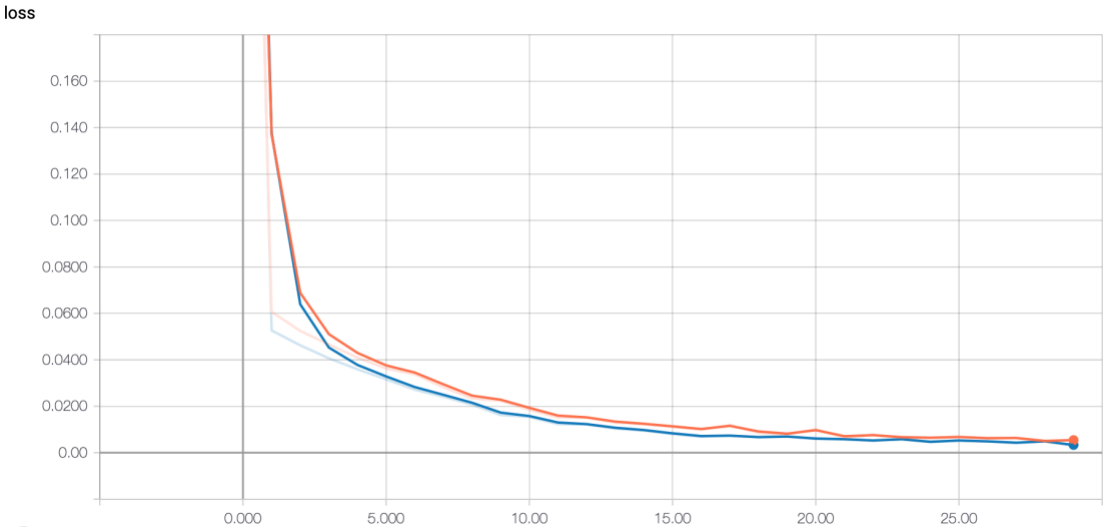
Σχήμα 4.9: Ακρίβεια (precision) στο σύνολο επικύρωσης



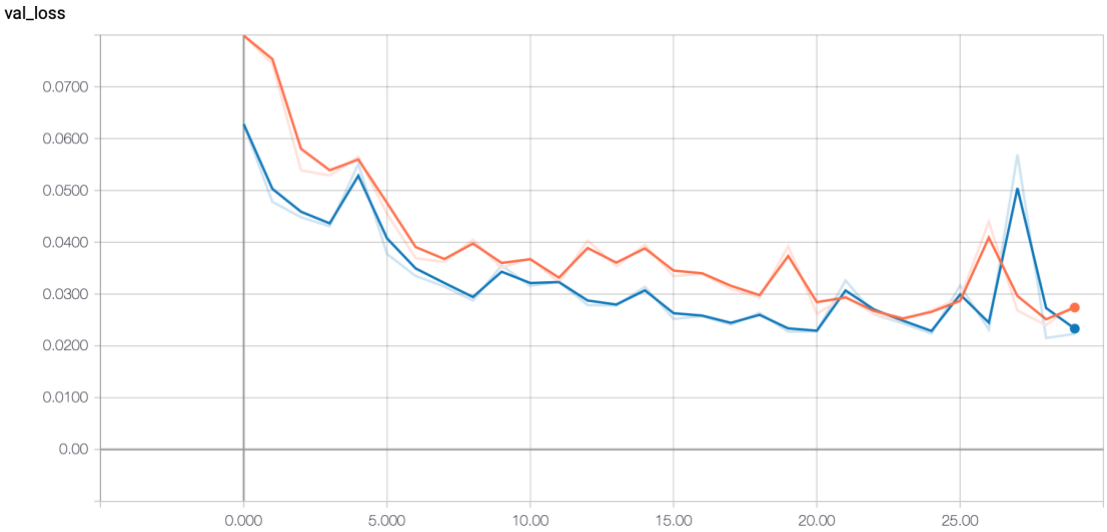
Σχήμα 4.10: Ανάκληση (recall) στο σύνολο εκπαίδευσης



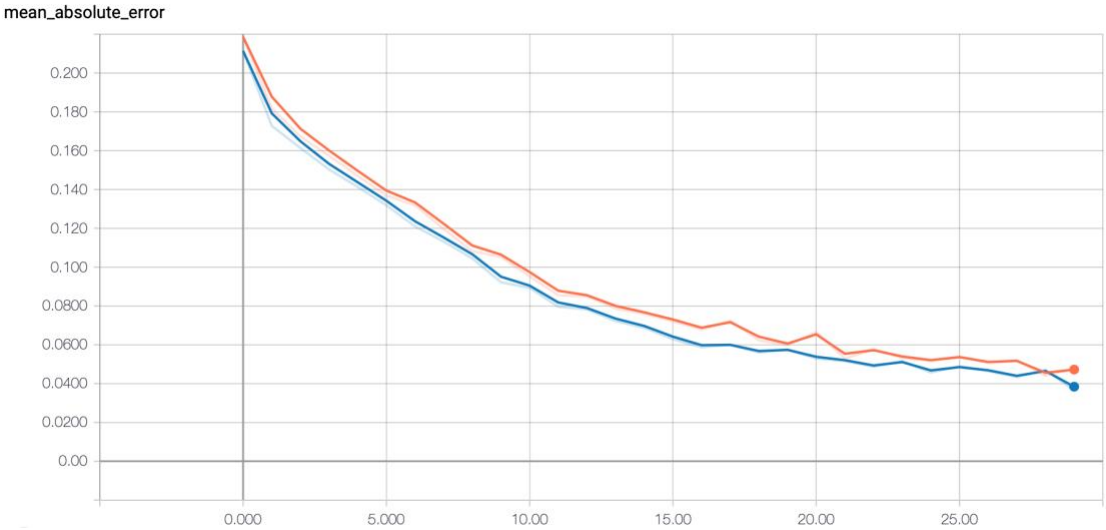
Σχήμα 4.11: Ανάκληση (recall) στο σύνολο επικύρωσης



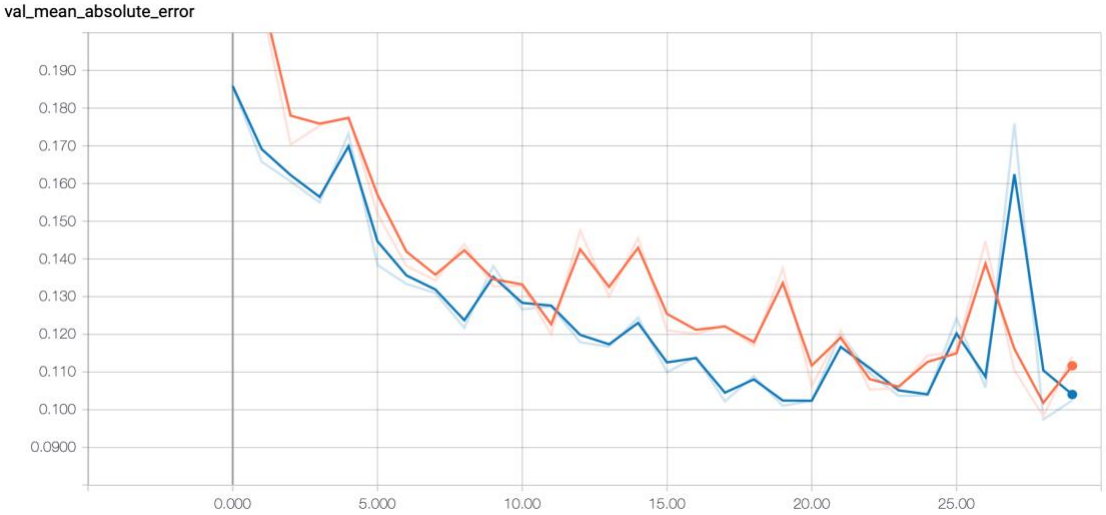
Σχήμα 4.12: Μέσο τετραγωνικό σφάλμα (mse) στα δεδομένα εκπαίδευσης



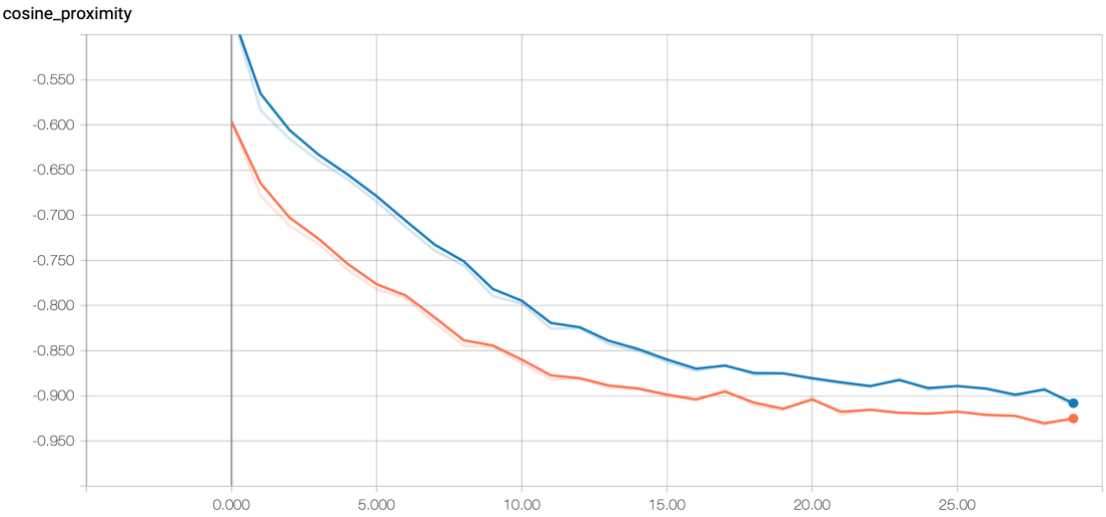
Σχήμα 4.13: Μέσο τετραγωνικό σφάλμα (mse) στα δεδομένα επικύρωσης



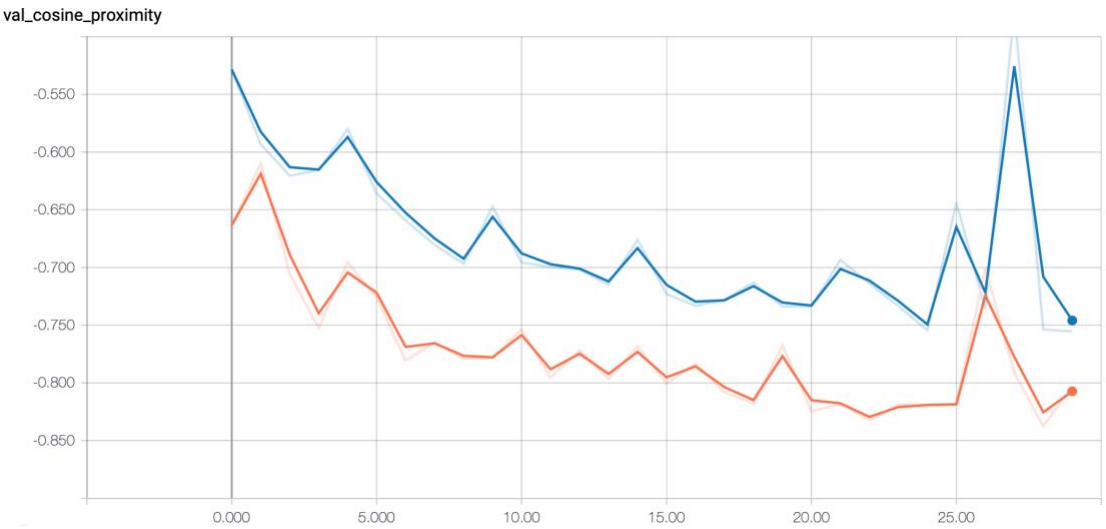
Σχήμα 4.14: Μέσο απόλυτο σφάλμα (mae) στα δεδομένα εκπαίδευσης



Σχήμα 4.15: Μέσο απόλυτο σφάλμα (mae) στα δεδομένα επικύρωσης



Σχήμα 4.16: Cosine proximity μετρική στα δεδομένα εκπαίδευσης



Σχήμα 4.17: Cosine proximity μετρική στα δεδομένα επικύρωσης

4.1.5 Διαδικασία μείωσης ρυθμού μάθησης

Είναι συνηθισμένη πρακτική να μειώνεται ο ρυθμός μάθησης του μοντέλου σε περίπτωση που δε βελτιώνεται κάποια μετρική απόδοσης. Στην παρούσα εργασία χρησιμοποιήθηκε αυτή η προσέγγιση στην εκπαίδευση όλων των μοντέλων. Αν μετά από 5 εποχές δεν είχε βελτιωθεί το μέσο τετραγωνικό σφάλμα στο σύνολο επικύρωσης για τα μοντέλα παλινδρόμησης και η κατηγορική εντροπία για τα μοντέλα κατηγοριοποίησης τότε μειωνόταν ο ρυθμός μάθησης πολλαπλασιαζόμενος με το παράγοντα 0.1. Είναι σημαντικό να σημειωθεί ότι η αρχική τιμή του ρυθμού μάθησης ήταν 0.001, καθώς αυτή ήταν η προτεινόμενη τιμή από τη δημοσίευση (paper) ([11]) που παρουσίασε τον αλγόριθμο 'Adam'. Χρησιμοποιήθηκε η κλάση 'ReduceLROnPlateau' του 'Keras' ([29]).

4.1.6 Προσαρμογή του μοντέλου

Στο τελευταίο στάδιο της διαδικασίας συνδυάζονταν όλα τα προηγούμενα στάδια που περιγράφηκαν για να γίνει η τελική προσαρμογή του μοντέλου. Το πλήθος των εποχών εκπαίδευσης διέφερε σε κάθε τύπο προβλήματος. Τα μοντέλα παλινδρόμησης εκπαιδεύτηκαν για 30 εποχές, ενώ τα μοντέλα κατηγοριοποίησης για 15. Η επιλογή του αριθμού των βημάτων εκπαίδευσης έγινε με βάση τις δυναμικές απεικονίσεις των μετρικών των μοντέλων. Η εκπαίδευση σταμάταγε όταν η γενική απόδοση σταματούσε να βελτιώνεται. Χρησιμοποιήθηκε η μέθοδος 'fit_generator' του 'Keras' ([29]).

4.2 Τελική αξιολόγηση και αποτελέσματα

4.2.1 Μοντέλα παλινδρόμησης

1ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: ResNet-50
- Χωρίς χρήση ομαλοποίησης (regularization)

Πίνακας 4.1: Μετρικές 1ου μοντέλου παλινδρόμησης

Ετικέτα	Μέσο τετραγωνικό σφάλμα (MSE)	Μέσο απόλυτο σφάλμα (MAE)
Διέγερση (arousal)	0.0252	0.1037
Σθένος (valence)	0.0228	0.1011

Πίνακας 4.2: Κατανομή δειγμάτων 1ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2932
2ο	1204	1186
3ο	914	840
4ο	1004	913

Αντιστοίχιση τεταρτημορίων:

- 1ο : σθένος > 0, διέγερση > 0
- 2ο : σθένος < 0, διέγερση > 0
- 3ο : σθένος < 0, διέγερση < 0
- 4ο : σθένος > 0, διέγερση < 0

Πίνακας 4.3: Λανθασμένα τοποθετημένα δείγματα 1ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	480
2ο	281
3ο	201
4ο	231
	Σύνολο: **1193**

2ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: ResNet-50
- Με χρήση ομαλοποίησης (regularization)

Πίνακας 4.4: Μετρικές 2ου μοντέλου παλινδρόμησης

Ετικέτα	Μέσο τετραγωνικό σφάλμα (MSE)	Μέσο απόλυτο σφάλμα (MAE)
Διέγερση (arousal)	0.0255	0.1044
Σθένος (valence)	0.0228	0.1004

Πίνακας 4.5: Κατανομή δειγμάτων 2ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2927
2ο	1204	1165
3ο	914	836
4ο	1004	943

Πίνακας 4.6: Λανθασμένα τοποθετημένα δείγματα 2ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	468
2ο	268
3ο	186
4ο	234
	Σύνολο: **1156**

3ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: InceptionResNet-v2
- Χωρίς χρήση ομαλοποίησης (regularization)

Πίνακας 4.7: Μετρικές 3ου μοντέλου παλινδρόμησης

Ετικέτα	Μέσο τετραγωνικό σφάλμα (MSE)	Μέσο απόλυτο σφάλμα (MAE)
Διέγερση (arousal)	0.0244	0.0997
Σθένος (valence)	0.0224	0.1010

Πίνακας 4.8: Κατανομή δειγμάτων 3ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2900
2ο	1204	1133
3ο	914	846
4ο	1004	992

Πίνακας 4.9: Λανθασμένα τοποθετημένα δείγματα 3ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	436
2ο	244
3ο	214
4ο	277
	Σύνολο: **1171**

4ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: InceptionResNet-v2
- Με χρήση ομαλοποίησης (regularization)

Πίνακας 4.10: Μετρικές 4ου μοντέλου παλινδρόμησης

Ετικέτα	Μέσο τετραγωνικό σφάλμα (MSE)	Μέσο απόλυτο σφάλμα (MAE)
Διέγερση (arousal)	0.0244	0.0974
Σθένος (valence)	0.022	0.099

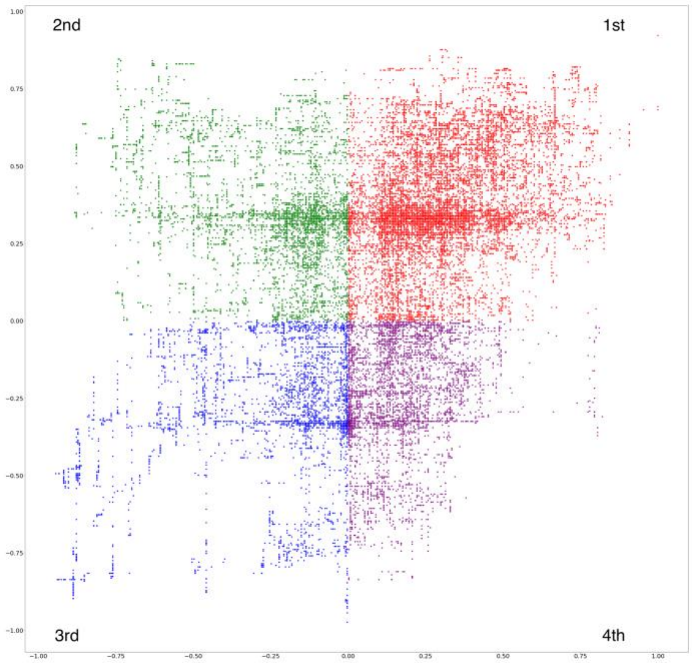
Πίνακας 4.11: Κατανομή δειγμάτων 4ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2903
2ο	1204	1246
3ο	914	797
4ο	1004	925

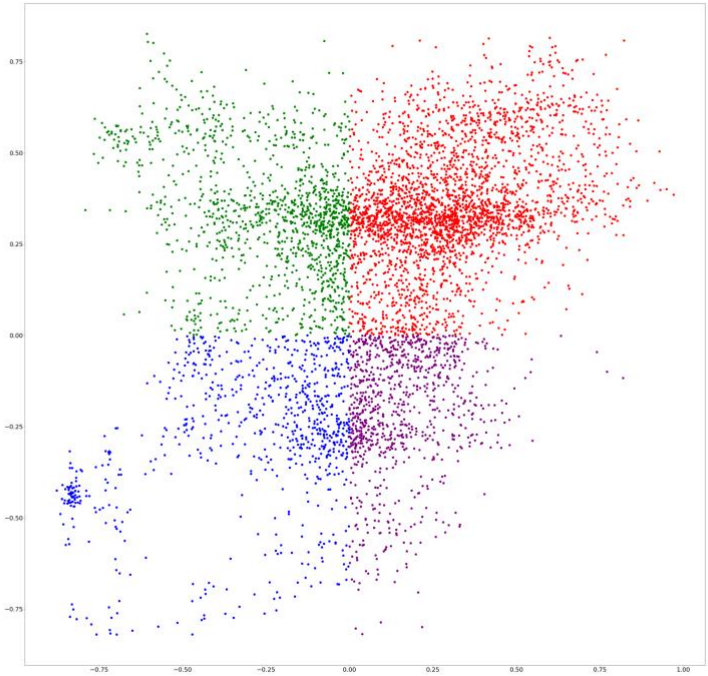
Πίνακας 4.12: Λανθασμένα τοποθετημένα δείγματα 4ου μοντέλου παλινδρόμησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	425
2ο	297
3ο	151
4ο	217
	Σύνολο: **1090**

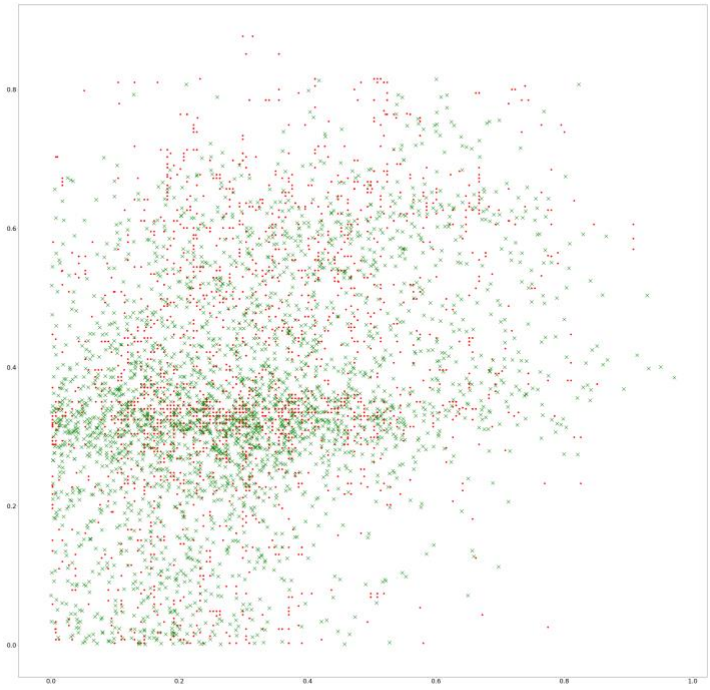
Στη συνέχεια παρουσιάζονται ορισμένες γραφικές παραστάσεις που δίνουν μία διαισθητική εικόνα για την επίδοση του 4ου μοντέλου. Με κόκκινο χρώμα φαίνονται οι πραγματικές τιμές ενώ με πράσινο οι προβλέψεις.



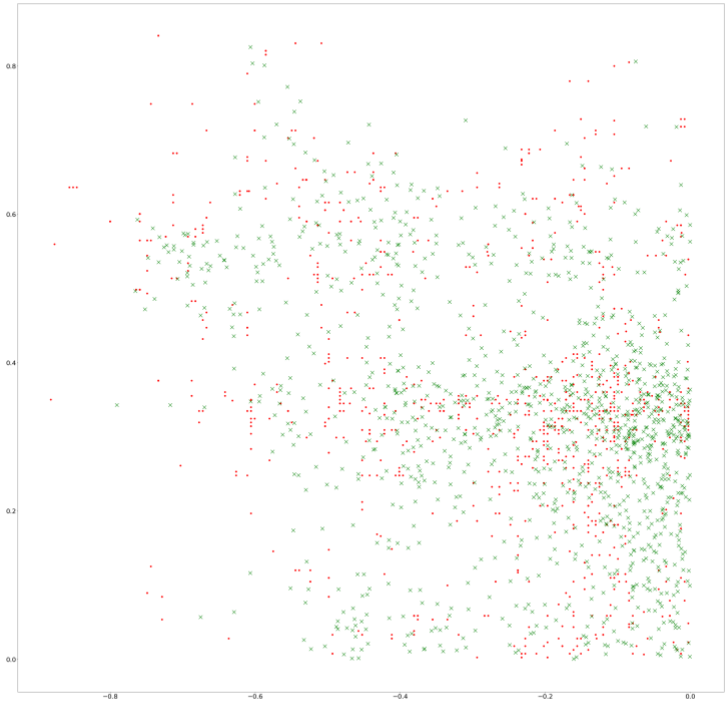
Σχήμα 4.18: Κατανομή δεδομένων ελέγχου (test set)



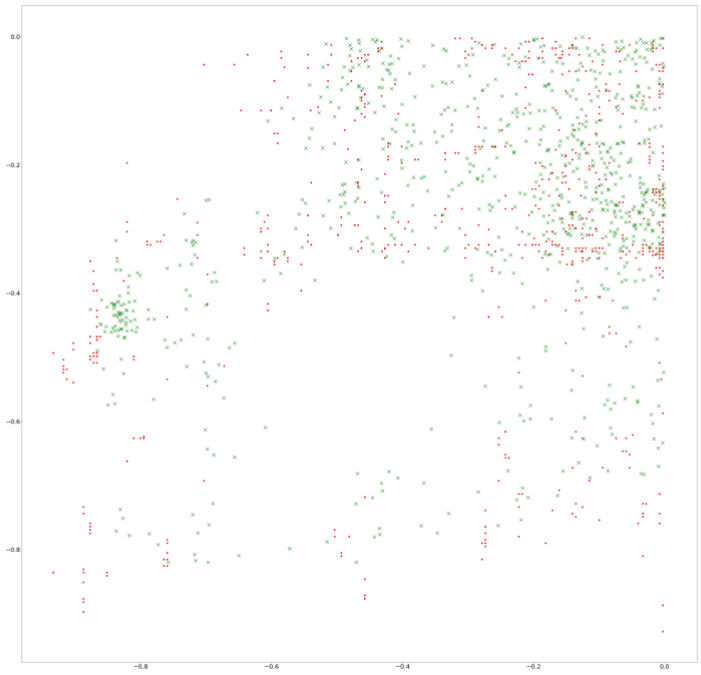
Σχήμα 4.19: Κατανομή προβλέψεων



Σχήμα 4.20: Κατανομή πραγματικών δεδομένων και προβλέψεων στο 1ο τεταρτημόριο

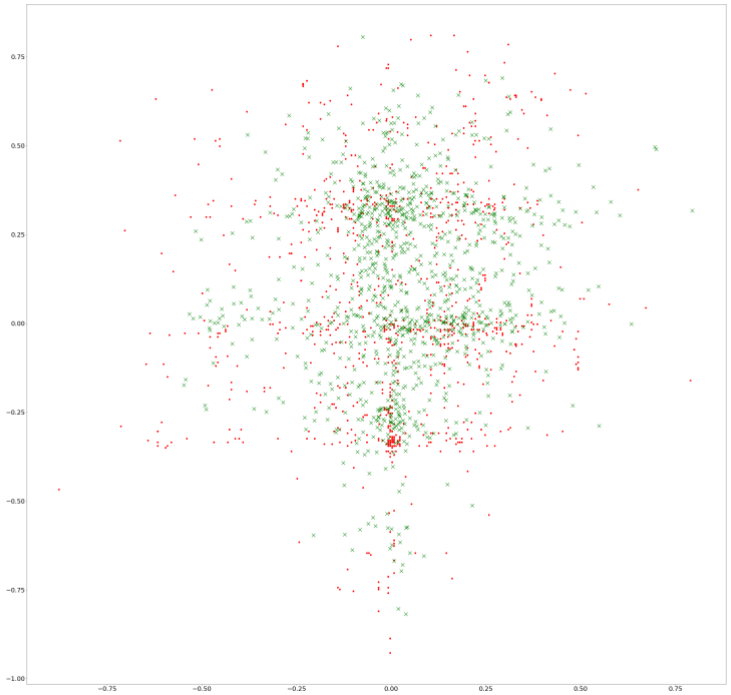


Σχήμα 4.21: Κατανομή πραγματικών δεδομένων και προβλέψεων στο 2ο τεταρτημόριο



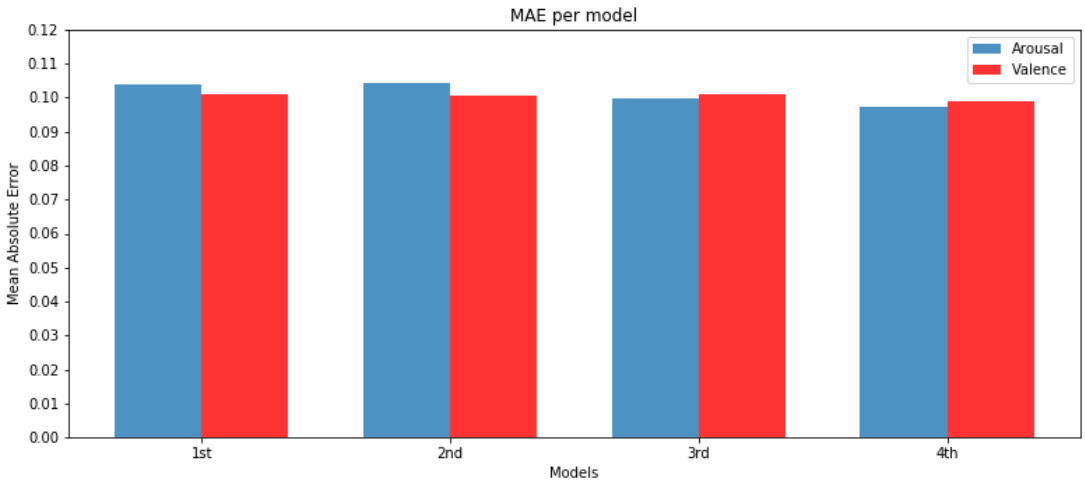
Σχήμα 4.22: Κατανομή πραγματικών δεδομένων και προβλέψεων στο 3ο τεταρτημόριο

Σχήμα 4.23: Κατανομή πραγματικών δεδομένων και προβλέψεων στο 4ο τεταρτημόριο

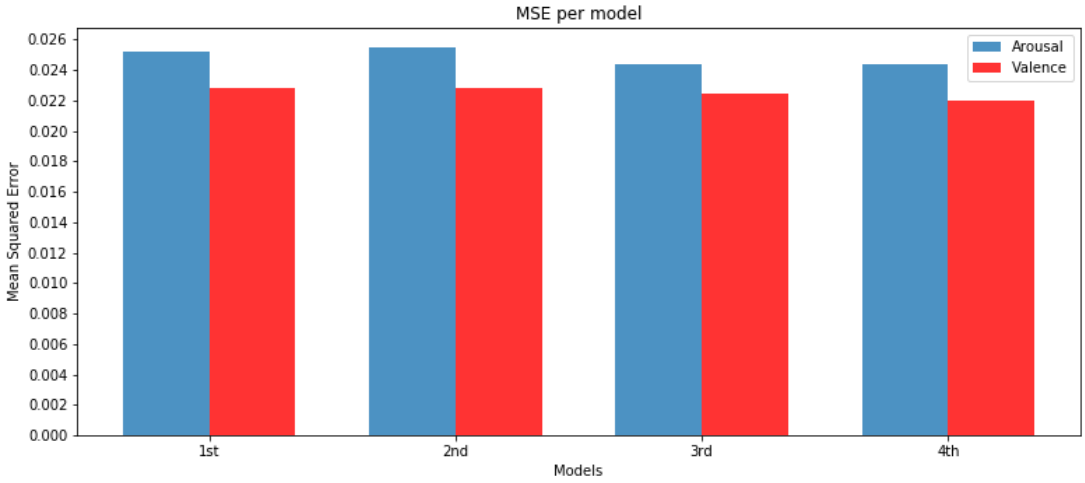


Σχήμα 4.24: Κατανομή προβλέψεων που έχουν τοποθετηθεί λανθασμένα.

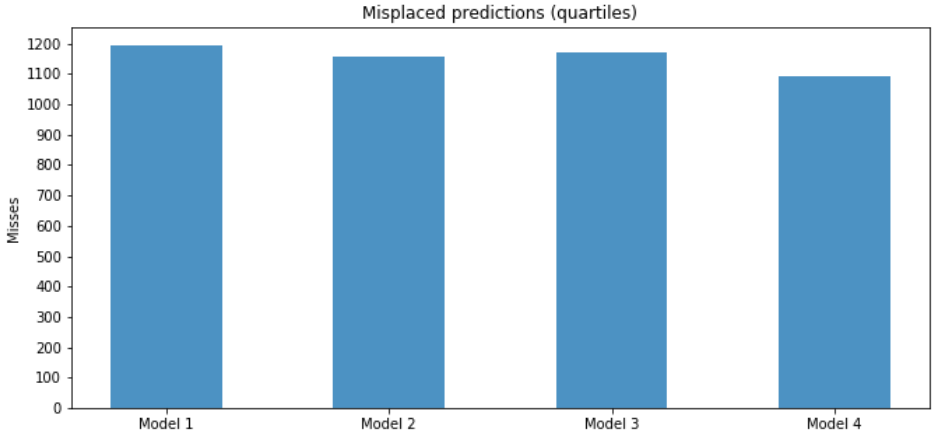
Ακολουθούν διαγράμματα που παρουσιάζουν οπτικά τις διαφορές των μετρικών μεταξύ των μοντέλων παλινδρόμησης.



Σχήμα 4.25: Μέσο απόλυτο σφάλμα για κάθε μοντέλο παλινδρόμησης, συγκριτικό διάγραμμα.



Σχήμα 4.26: Μέσο τετραγωνικό σφάλμα για κάθε μοντέλο παλινδρόμησης, συγκριτικό διάγραμμα.



Σχήμα 4.27: Λανθασμένα τοποθετημένα δείγματα ανά μοντέλο παλινδρόμησης, συγκριτικό διάγραμμα.

Στα σχήμα 4.25 και 4.26 φαίνεται ότι δεν υπάρχουν σημαντικές διαφορές μεταξύ των μετρικών των τεσσάρων μοντέλων παλινδρόμησης. Το σχήμα 4.27, όμως, δείχνει ότι το τέταρτο μοντέλο έχει λιγότερα λανθασμένα τοποθετημένα δείγματα.

4.2.2 Μοντέλα κατηγοριοποίησης

1ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: ResNet-50
- Χωρίς χρήση ομαλοποίησης (regularization)

Πίνακας 4.13: Μετρικές 1ου μοντέλου κατηγοριοποίησης

	Ακρίβεια (Precision)	**Ανάκληση (Recall)**	**'f-1' αποτέλεσμα (f-1 score)**		**Ορθότητα (Accuracy)**
Κλάση 0	0.78	0.81	0.8		
Κλάση 1	0.6	0.72	0.65		
Κλάση 2	0.68	0.61	0.64		**0.716**
Κλάση 3	0.73	0.54	0.62		
Συνολικά	**0.72**	**0.72**	**0.71**		

Αντιστοιχία κλάσεων:

- Κλάση 0: 1ο τεταρτημόριο
- Κλάση 1: 2ο τεταρτημόριο
- Κλάση 2: 3ο τεταρτημόριο
- Κλάση 3: 4ο τεταρτημόριο

Πίνακας 4.14: Πίνακας συγκρούσεων (confusion matrix) 1ου μοντέλου κατηγοριοποίησης

		Προβλέψεις			
		Κλάση 0	**Κλάση 1**	**Κλάση 2**	**Κλάση 3**
Πραγματικές κλάσεις	**Κλάση 0**	**2239**	355	67	88
	Κλάση 1	273	**865**	48	18
	Κλάση 2	126	139	**557**	92
	Κλάση 3	233	81	145	**545**

Πίνακας 4.15: Κατανομή δειγμάτων 1ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	**Πραγματικά δεδομένα**	**Προβλέψεις**
1ο	2749	2871
2ο	1204	1440
3ο	914	817
4ο	1004	743

Πίνακας 4.16: Λανθασμένα ταξινομημένα δείγματα 1ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	632
2ο	575
3ο	260
4ο	198
	Σύνολο: **1665**

2ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: ResNet-50
- Με χρήση ομαλοποίησης (regularization)

Πίνακας 4.17: Μετρικές 2ου μοντέλου κατηγοριοποίησης

	Ακρίβεια (Precision)	Ανάκληση (Recall)	'f-1' αποτέλεσμα (f-1 score)		Ορθότητα (Accuracy)
Κλάση 0	0.83	0.89	0.86		
Κλάση 1	0.79	0.68	0.73		
Κλάση 2	0.76	0.74	0.75		**0.801**
Κλάση 3	0.77	0.75	0.76		
Συνολικά	**0.8**	**0.8**	**0.8**		

Πίνακας 4.18: Πίνακας συγκρούσεων (confusion matrix) 2ου μοντέλου κατηγοριοποίησης

		Προβλέψεις			
		Κλάση 0	**Κλάση 1**	**Κλάση 2**	**Κλάση 3**
Πραγματικές κλάσεις	**Κλάση 0**	**2452**	149	57	91
	Κλάση 1	288	**820**	72	24
	Κλάση 2	77	53	**678**	106
	Κλάση 3	153	19	80	**752**

Πίνακας 4.19: Κατανομή δειγμάτων 2ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2970
2ο	1204	1041
3ο	914	887
4ο	1004	973

Πίνακας 4.20: Λανθασμένα ταξινομημένα δείγματα 2ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	518
2ο	221
3ο	209
4ο	221
	Σύνολο: **1169**

3ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: InceptionResNet-V2
- Χωρίς χρήση ομαλοποίησης (regularization)

Πίνακας 4.21: Μετρικές 3ου μοντέλου κατηγοριοποίησης

	Ακρίβεια (Precision)	Ανάκληση (Recall)	'f-1' αποτέλεσμα (f-1 score)		Ορθότητα (Accuracy)
Κλάση 0	0.83	0.88	0.86		
Κλάση 1	0.83	0.70	0.76		
Κλάση 2	0.80	0.72	0.76		**0.805**
Κλάση 3	0.71	0.79	0.75		
Συνολικά	**0.81**	**0.8**	**0.8**		

Πίνακας 4.22: Πίνακας συγκρούσεων (confusion matrix) 3ου μοντέλου κατηγοριοποίησης

		Προβλέψεις			
		Κλάση 0	**Κλάση 1**	**Κλάση 2**	**Κλάση 3**
Πραγματικές κλάσεις	**Κλάση 0**	**2430**	130	31	158
	Κλάση 1	258	**842**	61	43
	Κλάση 2	109	32	**656**	117
	Κλάση 3	128	11	68	**797**

Πίνακας 4.23: Κατανομή δειγμάτων 3ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2925
2ο	1204	1015
3ο	914	816
4ο	1004	115

Πίνακας 4.24: Λανθασμένα ταξινομημένα δείγματα 3ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	495
2ο	173
3ο	160
4ο	318
	Σύνολο: **1146**

4ο Μοντέλο:

- Προεκπαιδευμένο νευρωνικό δίκτυο: InceptionResNet-V2
- Με χρήση ομαλοποίησης (regularization)

Πίνακας 4.25: Μετρικές 4ου μοντέλου κατηγοριοποίησης

	Ακρίβεια (Precision)	Ανάκληση (Recall)	'f-1' αποτέλεσμα (f-1 score)		Ορθότητα (Accuracy)
Κλάση 0	0.87	0.9	0.88		
Κλάση 1	0.8	0.79	0.79		
Κλάση 2	0.8	0.82	0.81		**0.841**
Κλάση 3	0.84	0.76	0.8		
Συνολικά	**0.84**	**0.84**	**0.84**		

Πίνακας 4.26: Πίνακας συγκρούσεων (confusion matrix) 4ου μοντέλου κατηγοριοποίησης

		Προβλέψεις			
		Κλάση 0	**Κλάση 1**	**Κλάση 2**	**Κλάση 3**
Πραγματικές κλάσεις	**Κλάση 0**	**2468**	185	39	57
	Κλάση 1	182	**955**	53	14
	Κλάση 2	52	42	**749**	71
	Κλάση 3	128	18	92	**766**

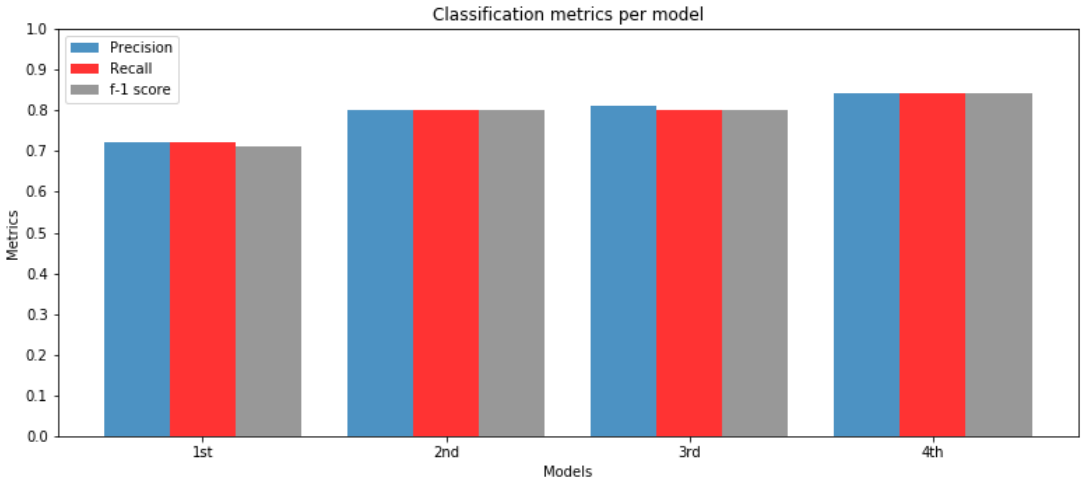
Πίνακας 4.27: Κατανομή δειγμάτων 4ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πραγματικά δεδομένα	Προβλέψεις
1ο	2749	2830
2ο	1204	1200
3ο	914	933
4ο	1004	908

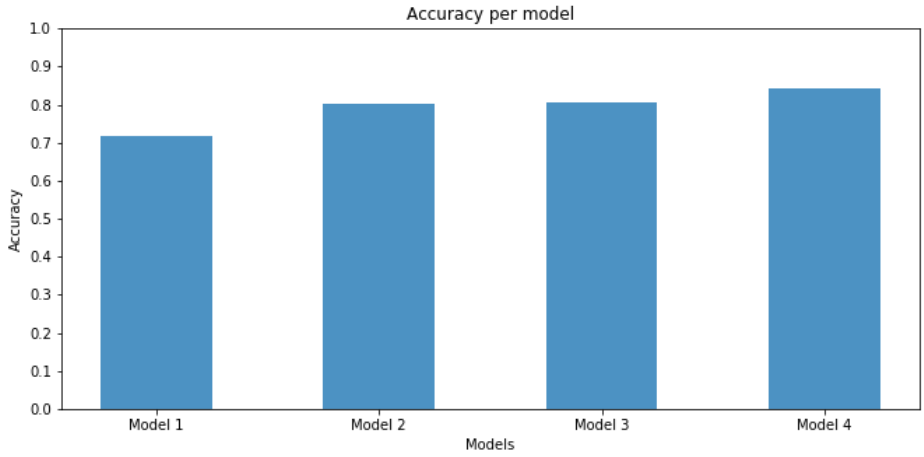
Πίνακας 4.28: Λανθασμένα ταξινομημένα δείγματα 4ου μοντέλου κατηγοριοποίησης

Τεταρτημόριο	Πλήθος δειγμάτων
1ο	362
2ο	245
3ο	184
4ο	142
	Σύνολο: **933**

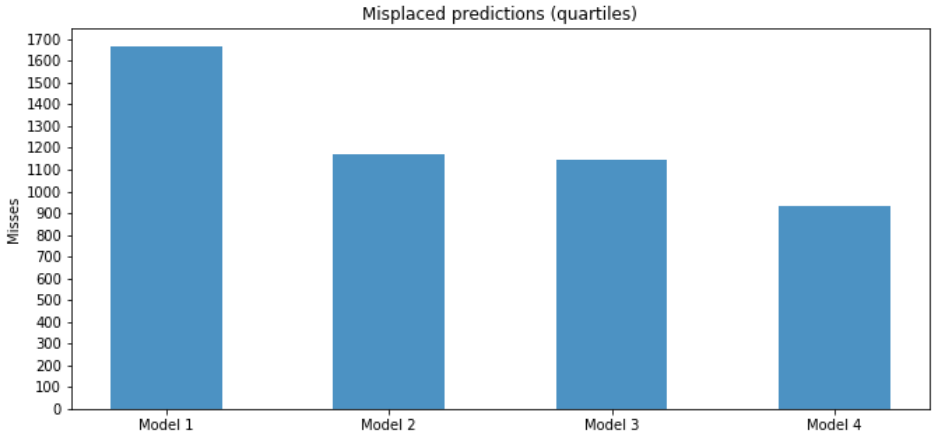
Ακολουθούν διαγράμματα που παρουσιάζουν οπτικά τις διαφορές των μετρικών μεταξύ των μοντέλων παλινδρόμησης.



Σχήμα 4.28: Ακρίβεια (precision), ανάκληση (recall) και 'f-1' αποτέλεσμα (f-1 score) για κάθε μοντέλο κατηγοριοποίησης, συγκριτικό διάγραμμα.



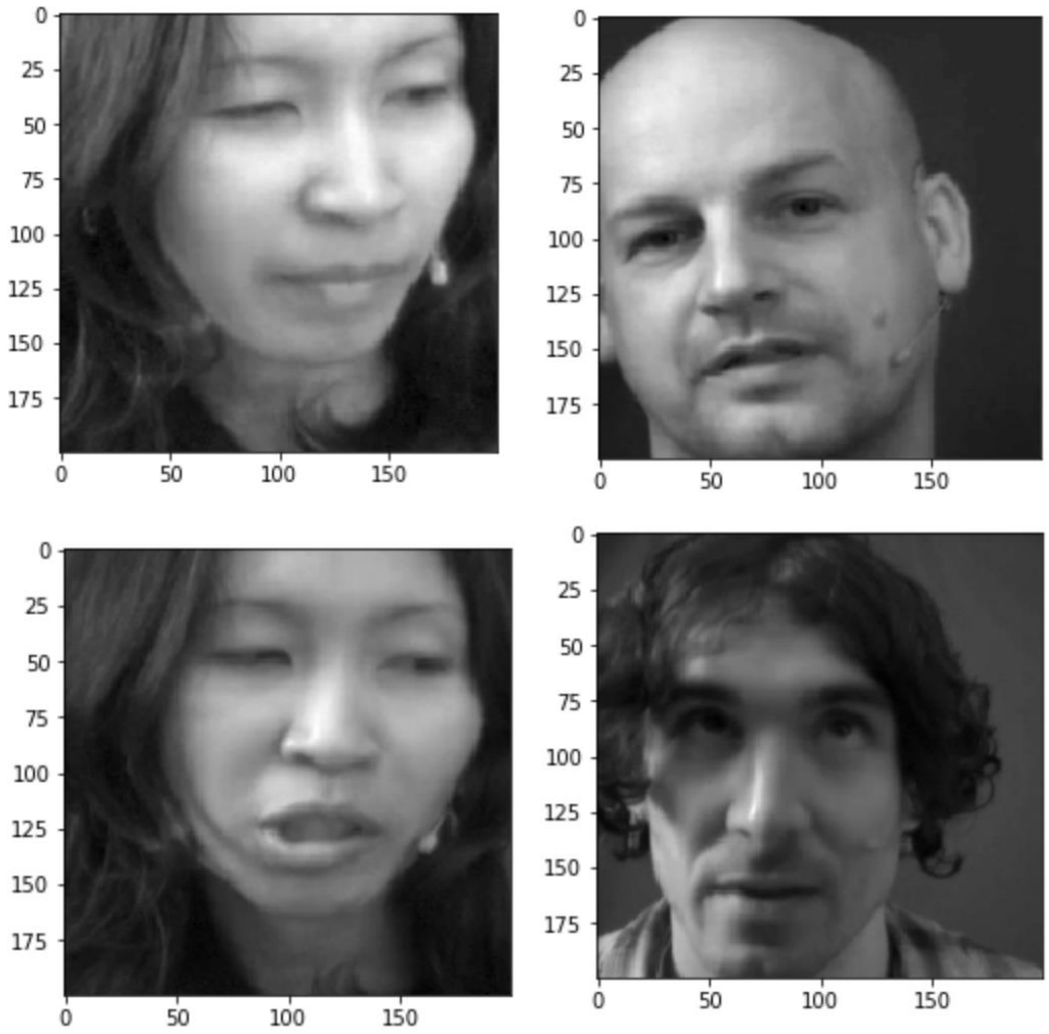
Σχήμα 4.29: Ορθότητα (accuracy) για κάθε μοντέλο κατηγοριοποίησης, συγκριτικό διάγραμμα.



Σχήμα 4.30: Λανθασμένα ταξινομημένα δείγματα ανά μοντέλο παλινδρόμησης, συγκριτικό διάγραμμα.

Η προοδευτική βελτίωση της απόδοσης είναι εμφανής. Το τέταρτο μοντέλο έχει την υψηλότερη επίδοση.

Ορισμένες εικόνες που το 4ο μοντέλο απέτυχε να κατηγοριοποιήσει σωστά είναι οι ακόλουθες:



Γενικά το μοντέλο αποτύγχανε σε εικόνες που ήταν δύσκολο και για τον ίδιο τον άνθρωπο να κατανοήσει τη συναισθηματική κατάσταση του εικονιζόμενου ατόμου. Σε ορισμένες αστοχίες το πρόσωπο δεν ήταν ευδιάκριτο ή το άτομο είχε στρίψει το κεφάλι του.

5

Επίλογος

5.1 Σύνοψη και Συμπεράσματα

Σε αυτή τη διπλωματική εργασία παρουσιάστηκαν μοντέλα νευρωνικών δικτύων με στόχο την αναγνώριση της συναισθηματικής κατάστασης του ανθρώπου μέσα από εκφράσεις του προσώπου. Το θέμα αυτό προσεγγίστηκε ως πρόβλημα παλινδρόμησης (regression) και κατηγοριοποίησης (classification). Σε όλα τα μοντέλα χρησιμοποιήθηκε η τεχνική της μεταφοράς γνώσης (transfer learning), όπου επιλέχθηκαν πολύ αποδοτικά προεκπαιδευμένα βαθιά νευρωνικά δίκτυα (ResNet50 και InceptionResNet-V2).

Τα πειράματα που εκτελέστηκαν σε μεγάλα σύνολα δεδομένων απέδειξαν τη σταθερότητα των μοντέλων τόσο ποιοτικά όσο και ποσοτικά. Το πρόβλημα της αναγνώρισης των συναισθημάτων του ανθρώπου μέσα από εκφράσεις του προσώπου επιλύεται αποδοτικά. Όλα τα μοντέλα εκπαιδεύτηκαν έτσι ώστε να μεγιστοποιείται η πιθανότητα να εξάγουν ορθές προβλέψεις.

Το καλύτερο μοντέλο για το πρόβλημα παλινδρόμησης ήταν αυτό που περιλάμβανε το προεκπαιδευμένο νευρωνικό δίκτυο ‘InceptionResNet-V2’ και στο οποίο εφαρμόστηκε ομαλοποίηση (regularization) L2. Το μέσο τετραγωνικό σφάλμα (mse) ήταν 0.0244 και 0.022 για την ετικέτα της διέγερσης (arousal) και του σθένους (valence) αντίστοιχα. Το σφάλμα, με βάση το οποίο έγινε η βελτιστοποίηση, ήταν πολύ χαμηλό. Το γεγονός αυτό αποδεικνύει ότι υπάρχει η δυνατότητα να προσεγγιστεί η ακριβής τιμή των ετικετών που καθορίζουν την τοποθέτηση των δειγμάτων στο χώρο συναισθημάτων (σχήμα 1.3) με μεγάλη ακρίβεια. Τα λανθασμένα τοποθετημένα δείγματα στα τεταρτημόρια ήταν 1090 από το σύνολο των 5871 δειγμάτων. Στο σχήμα 4.24 φαίνεται ότι η πλειονότητα των λανθασμένα τοποθετημένων δειγμάτων ήταν κοντά στους άξονες. Στα σχήματα 4.18, 4.19, 4.20, 4.21, 4.22 και 4.23 φαίνεται ότι η κατανομή των προβλέψεων προσεγγίζει σημαντικά την κατανομή των πραγματικών τιμών.

Στο πρόβλημα κατηγοριοποίησης η αρχιτεκτονική με το προεκπαιδευμένο νευρωνικό δίκτυο ‘InceptionResNet-V2’ και την ομαλοποίηση L2 έδωσε βέλτιστα αποτελέσματα. Όλες οι μετρικές επίδοσης (ορθότητα-accuracy, ακρίβεια-precision, ανάκληση-recall και f-1 αποτέλεσμα) έλαβαν τιμή 0.84. Μάλιστα στο πρώτο τεταρτημόριο που περιλάμβανε τα περισσότερα δείγματα η ορθότητα προσέγγισε το 90%. Η απόδοση του μοντέλου ήταν πολύ καλή καθώς τα λανθασμένα ταξινομημένα δείγματα ήταν 933 από το σύνολο των 5871 δειγμάτων.

Ενδιαφέρον παρουσιάζει το γεγονός ότι το πλήθος των λανθασμένα τοποθετημένων δειγμάτων στα τεταρτημόρια του χώρου συναισθημάτων (σχήμα 1.3) ήταν αρκετά μικρότερο στο βέλτιστο μοντέλο κατηγοριοποίησης συγκριτικά με το καλύτερο δίκτυο παλινδρόμησης. Επιπροσθέτως και στους δύο τύπους προβλημάτων το βαθύ νευρωνικό δίκτυο ‘InceptionResNet-V2’ απέδιδε καλύτερα από ότι το ‘ResNet50’, γεγονός που δείχνει ότι υπάρχει συνεχής βελτίωση των αρχιτεκτονικών καθώς το ‘InceptionResNet-V2’ παρουσιάστηκε πιο πρόσφατα. Η χρήση της ομαλοποίησης (regularization) επίσης βοήθησε στην αύξηση της επίδοσης των μοντέλων.

5.2 Μελλοντικές επεκτάσεις

Παρά το γεγονός ότι τα μοντέλα που δημιουργήθηκαν είχαν υψηλή απόδοση, υπάρχουν ακόμα αρκετές κατευθύνσεις κατά τις οποίες το σύστημα θα μπορούσε να βελτιωθεί περαιτέρω.

- Προσθήκη αρχιτεκτονικής με ανατροφοδοτούμενα νευρωνικά δίκτυα (RNN): Τα αρχικά δεδομένα που χρησιμοποιήθηκαν ήταν σε μορφή βίντεο, επομένως τα επί μέρους στιγμιότυπα που εξάχθηκαν ήταν μία ακολουθία (sequence) ανά συνεδρία. Το γεγονός αυτό ενισχύει τις πιθανότητες να πετύχουν ακόμα καλύτερα αποτελέσματα συστήματα υβριδικά που περιλαμβάνουν αρχιτεκτονικές CNN και RNN νευρωνικών δικτύων. Τα ανατροφοδοτούμενα νευρωνικά δίκτυα (RNN) διαθέτουν μνήμη. Η προσθήκη μνήμης, ωστόσο, σε ένα δίκτυο, έχει κάποιο σκοπό : Υπάρχουν πληροφορίες μέσα στις ακολουθίες εισόδου, τις οποίες χρησιμοποιούν τα ανατροφοδοτούμενα νευρωνικά δίκτυα προκειμένου να εκτελούν εργασίες που τα απλά προωθητικά δίκτυα δεν μπορούν. Τα δίκτυα αυτά έχουν την ικανότητα να εξάγουν συσχετίσεις μέσα από ακολουθίες δεδομένων, επομένως η χρήση τους μπορεί να οδηγήσει σε εξαγωγή σημαντικών χαρακτηριστικών που εκφράζουν την αλλαγή τη συναισθηματικής κατάστασης του χρήστη κατά τη διάρκεια της εκάστοτε συνεδρίας.

- Προσθήκη εικόνων από πλαϊνές γωνίες λήψης: Τα δεδομένα που χρησιμοποιήθηκαν στην παρούσα εργασίας εξάχθηκαν από βίντεο που φαινόταν όλο το πρόσωπο του ατόμου από μπροστά. Μία πιθανή επέκταση του συστήματος θα ήταν η προσπάθεια αναγνώρισης της συναισθηματικής κατάστασης του ανθρώπου με χρήση εικόνων από πλαϊνές γωνίες λήψης.

- Δημιουργία μοντέλου για την ανίχνευση του προσώπου: Αν και το πρόβλημα της ανίχνευσης τους προσώπου θεωρείται λυμένο από τον τομέα της όρασης υπολογιστικών, στην παρούσα εργασία ο έτοιμος ανιχνευτής προσώπου που χρησιμοποιήθηκε είχε ορισμένες αστοχίες. Μία πιθανή επέκταση θα ήταν η στοχευμένη δημιουργία ενός μοντέλου για την ανίχνευση του προσώπου με χρήση των δεδομένων που χρησιμοποιήθηκαν. Ένα πρόβλημα που ίσως προέκυπτε θα ήταν η εύρεση εικόνων με ετικέτες ώστε να γίνει η εκπαίδευση του μοντέλου, καθώς η διαδικασία χειροκίνητης δημιουργίας ετικετών είναι χρονοβόρα.

- Επέκταση του συνόλου δεδομένων: Στην παρούσα εργασία το μέγεθος του εξαγόμενου συνόλου δεδομένων ήταν ικανοποιητικό έτσι ώστε να εκτελεστούν επιτυχώς τα πειράματα και να δημιουργηθούν τα μοντέλα. Σε περίπτωση όμως που προστεθούν σε μελλοντική δουλειά ανατροφοδοτούμενα νευρωνικά δίκτυα το ολικό σύστημα θα γίνει πιο σύνθετο με αποτέλεσμα να απαιτούνται περισσότερα δεδομένα. Σε αυτή την περίπτωση θα μπορούσαν να χρησιμοποιηθούν τεχνικές αύξησης του συνόλου με χρήση των υπαρχόντων δεδομένων (data augmentation). Αξίζει να σημειωθεί ότι υπάρχει άλλη μία μεγάλη βάση δεδομένων, η ‘BP4D’ ([16], [17]) από τα πανεπιστήμιο Binghamton και Pittsburgh, η οποία μπορεί να χρησιμοποιηθεί για εύρεση δεδομένων.

Βιβλιογραφία

[1] Neural Networks and Learning Machines by S. O. Haykin

[2] Artificial Intelligence: A Modern Approach by S. Russell and P. Norvig

[3] Pattern Recognition by S. Theodoridis and K. Koutroumbas

[4] Deep Learning: A practitioner's approach by J. Patterson and A. Gibson

[5] S. Jaiswal, M. Valstar: 'Deep learning the dynamic appearance and shape of facial action units', 2016 (link)

[6] K. He, X. Zhang, S. Ren, J. Sun: 'Deep residual learning for image recognition', 2015 (link)

[7] C. Szegedy, S. Ioffe, V. Vanhoucke: 'Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning', 2016 (link)

[8] H. Jung, S. Lee, J. Yim, S. Park, J. Kim: 'Joint Fine-Tuning in Deep Neural Networks for Facial Expression Recognition', 2015 (link)

[9] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, 'Rethinking the Inception Architecture for Computer Vision', 2015 (link)

[10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov: 'Dropout: A Simple Way to Prevent Neural Networks from Overfitting', 2014 (link)

[11] D. P. Kingma, J. Ba: 'Adam a method for stochastic optimization', 2014 (link)

[12] N. Qian: 'The momentum term in gradient descent', 2011 (link)

[13] P. Viola, M. Jones: 'Rapid object detection using a boosted cascade of simple features', 2001 (link)

[14] C. Darken, J. Moody: 'Note on learning rate schedules for stochastic optimization', 1991 (link)

[15] G. McKeown, M. Valstar, R. Cowie, M. Pantic, M. Schroder: 'The SEMAINE Database: Annotated Multimodal Records of Emotionally Colored Conversations between a Person and a Limited Agent', 2011 (link)

[16] X. Zhang, L. Yin, J. Cohn, S. Canavan, M. Reale, A. Horowitz, P. Liu: 'A high resolution spontaneous 3D dynamic facial expression database', 2013 (link)

[17] 'Analyzing facial expressions in three dimensional space, BP4D-Spontanous: Binghamton-Pittsburgh 3D Dynamic Spontaneous Facial Expression Database', (link)

[18] 'Convolution Neural network for visual recognition' (link)

[19] 'An overview of gradient descent optimization algorithms' by S. Ruder (link)

[20] 'An Intuitive Explanation of Convolutional Neural Networks' (link)

[21] 'A Quick Introduction to Neural Networks' (link)

[22] 'Regularization in deep learning' by C. Scheau (link)

[23] 'A friendly introduction to cross-entropy loss' by R. DiPietro (link)

[24] 'What is One Hot Encoding? Why And When do you have to use it?' by R. Vasudev (link)

[25] 'Improving Inception and Image Classification in TensorFlow' by A. Alemi (link)

[26] 'TensorBoard: Visualizing Learning (link)

[27] NumPy Documentation (link)

[28] Pandas Documentation (link)

[29] Keras Documentation (link)

[30] Tensorflow Documentation (link)

[31] Matplotlib Documentation (link)

[32] Scikit-learn Documentation (link)

[33] OpenCV Documentation (link)

[34] Wikipedia: ‘Artificial Intelligence (link)

[35] Wikipedia: ‘Machine Learning’ (link)

[36] Wikipedia: ‘Deep Learning’ (link)

[37] Wikipedia: ‘ImageNet’ (link)

[38] Wikipedia: ‘Digital Image Processing’ (link)

[39] «Χρήση μεταφοράς γνώσης για την εκπαίδευση ενός βαθιού νευρωνικού δικτύου στην αναγνώριση εκφράσεων» από το Θ. Ταγάρη (link)