



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Ανάπτυξη Ανεξάρτητου Συστήματος Αναγνώρισης Πτώσεων με χρήση Βαθιών Νευρωνικών Δικτύων και Edge Computing

Διπλωματική Εργασία

Αλέξανδρος Πάχος

Επιβλέπων: **Παναγιώτης Τσανάκας**

Καθηγητής ΕΜΠ

Αθήνα, Μάρτιος 2021



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
 ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
 ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
 ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Ανάπτυξη Ανεξάρτητου Συστήματος Αναγνώρισης Πτώσεων με χρήση Βαθιών Νευρωνικών Δικτύων και Edge Computing

Διπλωματική Εργασία

Αλέξανδρος Πάχος

Επιβλέπων: **Παναγιώτης Τσανάκας**

Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 22η Μαρτίου 2021

(Υπογραφή)

(Υπογραφή)

(Υπογραφή)

.....

.....

.....

Παναγιώτης Τσανάκας

Κωνσταντίνα Νικήτα

Δημήτριος-Διονύσιος
Κουτσούρης

Καθηγητής Ε.Μ.Π.

Καθηγήτρια Ε.Μ.Π.

Καθηγητής Ε.Μ.Π.

Αθήνα, Μάρτιος 2021

(Υπογραφή)

.....

Αλέξανδρος Πάχος

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

Copyright © Αλέξανδρος Πάχος, 2021

Με επιφύλαξη παντός δικαιώματος. All rights reserved.

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

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

Περίληψη

Μια από τις βασικές προκλήσεις της σύγχρονης κοινωνίας, είναι η εξασφάλιση της ομαλής και υγιούς γήρανσης. Εκτός από την προφανή συμβολή της ιατρικής επιστήμης σε αυτό, καθοριστικό ρόλο κατέχει επίσης η επιστήμη των υπολογιστών και οι εφαρμογές της.

Οι πτώσεις αποτελούν ένα από τα συχνότερα αίτια τραυματισμών, αλλά ακόμα και θανάτων, στους ανθρώπους από εξήντα πέντε ετών και πάνω. Προκαλούν σοβαρά προβλήματα υγείας, ψυχολογική πίεση, με την έννοια του φόβου μελλοντικής πτώσης αλλά και περιορίζουν τη γενικότερη ανεξαρτησία του ηλικιωμένου στην καθημερινή του ζωή. Ο συνδυασμός των παραπάνω με το φαινόμενο της δημογραφικής γήρανσης καθιστά αναγκαία την ανάπτυξη ενός συστήματος αναγνώρισης πτώσεων. Ένα τέτοιο σύστημα θα παρέχει τη δυνατότητα της άμεσης ανταπόκρισης σε περίπτωση πτώσης, πράγμα που δύναται ακόμα και να σώσει τη ζωή του υποκειμένου.

Στην παρούσα εργασία μελετάται η ανάπτυξη ενός συστήματος ανίχνευσης πτώσεων με χρήση νευρωνικών δικτύων βαθιάς μηχανική μάθησης. Δεδομένου ότι το σύστημα αυτό θα χρησιμοποιηθεί για τη βοήθεια ατόμων της τρίτης ηλικίας, καθίσταται απαραίτητη η εξασφάλιση της αυτονομίας και της ανεξαρτησίας της λειτουργίας του. Δηλαδή, επιλέχθηκε η υλοποίηση του σε ενσωματωμένο κύκλωμα (πλακέτα), το οποίο θ' αναλαμβάνει εξ' ολοκλήρου το υπολογιστικό φορτίο.

Η λειτουργία του συστήματος ξεκινά με την τροφοδοσία εικόνων από την κάμερα, η οποία είναι συνδεδεμένη στην πλακέτα, σε ένα νευρωνικό δίκτυο το οποίο υπολογίζει τα βασικά σημεία του ανθρώπινου σκελετού. Έπειτα αυτά προωθούνται σε ένα δεύτερο νευρωνικό δίκτυο το οποίο και αποφασίζει για το ενδεχόμενο ύπαρξης πτώσης.

Λέξεις κλειδιά: Τεχνητή νοημοσύνη, Μηχανική μάθηση, Νευρωνικά δίκτυα, Edge Computing, TPU, Πτώση.

Abstract

One of the main challenges of modern society is to ensure smooth and healthy aging. In addition to the obvious contribution of medical science to this objective, computer science and its applications can also play a significant role.

Falls are one of the most common causes of injuries and even death for people older than sixty-five. They cause serious health problems, psychological stress, in the sense of fear of future fall, but also limit the general independence of the elderly in their daily life. The combination of the above with the phenomenon of demographic aging makes it necessary to develop a system of fall detection. Such a system will provide the ability to respond immediately in the event of a fall, which may even save the subject's life.

In the present work, the development of a fall detection system using deep machine learning neural networks is studied. As this system will be used to help the elderly, it becomes necessary to ensure the autonomy and independence of its operation. For that reason, the use of an integrated circuit (board) was chosen, which will shoulder the entire computational load.

The operation of the system begins with the supply of images from the camera, which is connected to the board, to a neural network that estimates the location of the basic key points of the human skeleton. Those are forwarded to a second neural network which decides on the possibility of a fall.

Keywords: Artificial Intelligence, Machine Learning, Neural Networks, Edge Computing, TPU, Fall.

Ευχαριστίες

Η διπλωματική εργασία αυτή σηματοδοτεί το πέρας των σπουδών μου στο Εθνικό Μετσόβιο Πολυτεχνείο. Θα ήθελα να ευχαριστήσω λοιπόν τα πρόσωπα τα οποία συνέβαλαν στην προσπάθεια αυτή.

Αρχικά θα ήθελα να ευχαριστήσω τον επιβλέποντα της εργασίας μου, καθηγητή της σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών Ε.Μ.Π.

κο. Παναγιώτη Τσανάκα, για την ανάθεση ενός εξαιρετικά ενδιαφέροντος θέματος, αλλά και για την πολύτιμη και συνεχή βοήθεια και καθοδήγηση που μου προσέφερε καθ' όλη τη διάρκεια εκπόνησής του.

Ευχαριστίες θα ήθελα να εκφράσω στον κο. Ηλία Μαγκλογιάννη, καθηγητή του Πανεπιστημίου Πειραιά, ο οποίος παρείχε ανιδιοτελώς χρήσιμες συμβουλές πάνω στο τεχνικό κομμάτι της εργασίας.

Τέλος θα ήθελα να ευχαριστήσω την οικογένειά μου, τους φίλους μου και ιδιαίτερα τους γονείς μου για την αστείρευτη υποστήριξη και συμπαράσταση που μου χάρισαν όλα αυτά τα χρόνια της ακαδημαϊκής μου πορείας.

Περιεχόμενα

Κεφάλαιο 1 Μηχανική Μάθηση

1.1 Ορισμός	14
1.2 Ιστορική αναδρομή	16
1.3 Μηχανική μάθηση σήμερα	20

Κεφάλαιο 2 Θεωρία Νευρωνικών Δικτύων

2.1 Ορισμός	25
2.2 Μοντέλο τεχνητού νευρώνα	26
2.3 Συνάρτηση ενεργοποίησης (Activation function)	27
2.3.1 Συνάρτηση κατωφλίου (Threshold function)	28
2.3.2 Σιγμοειδής συνάρτηση (Sigmoid function)	29
2.3.3 Συνάρτηση υπερβολικής εφαπτομένης (Tanh Function)	29
2.3.3 Ανορθωμένη Γραμμική Συνάρτηση Ράμπας (ReLU)	30
2.4 Αρχιτεκτονικές νευρωνικών δικτύων	31
2.4.1 Ενός επιπέδου δίκτυα πρόσθιας τροφοδότησης	31
2.4.2 Πολυεπίπεδα δίκτυα πρόσθιας τροφοδότησης	32
2.4.3 Αναδρομικά δίκτυα	33
2.5 Μέθοδοι μάθησης	34
2.6 Συνάρτηση κόστους (Loss function)	37
2.6.1 Συνάρτηση μέσου τετραγωνικού σφάλματος (MSE)	37
2.6.2 Δυαδική συνάρτηση εντροπίας (Binary cross entropy)	38
2.6.3 Κατηγορική συνάρτηση εντροπίας (Categorical cross entropy)	38
2.7 Αλγόριθμοι βελτιστοποίησης	39
2.7.1 Αλγόριθμος κλίσης κατάβασης (Gradient Descent)	40
2.7.2 Αλγόριθμος εκτίμησης προσαρμοστικής ροπής (Adam)	42
2.8 Αλγόριθμος Οπισθοδιάδοσης (Backpropagation)	44

Κεφάλαιο 3 Συνελικτικά Νευρωνικά Δίκτυα

3.1 Ορισμός	47
3.2 Επίπεδα επεξεργασίας	48
3.2.1 Επίπεδο εισόδου (Input layer)	49
3.2.2 Συνελικτικό επίπεδο (Convolution layer)	50
3.2.3 Επίπεδο συγκέντρωσης (Pooling layer)	55
3.2.3 Πλήρως συνδεδεμένο επίπεδο (Fully connected layer)	56
3.3 Συνελικτικά δίκτυα μιας διάστασης	57

Κεφάλαιο 4 Edge Computing

4.1 Ορισμός	58
4.2 Πλεονεκτήματα	59
4.3 Προκλήσεις	61
4.4 Edge μονάδα επεξεργασίας τανυστή (Edge TPU)	62
4.4.1 Τανυστής (Tensor)	62
4.4.2 Μονάδα επεξεργασίας τανυστή	63
4.4.3 Coral development board	66

Κεφάλαιο 5 Σχεδιασμός και Υλοποίηση

5.1 Γενική περιγραφή	67
5.2 Συλλογή δεδομένων	69
5.3 Δίκτυο PoseNet	71
5.4 Προ-επεξεργασία δεδομένων	73
5.4.1 Φιλτράρισμα	74
5.4.2 Διάσπαση και ομαδοποίηση	74
5.4.3 Κανονικοποίηση	75
5.4.4 Παράθυρο ολίσθησης	75
5.4.5 Χαρακτηρισμός δειγμάτων	76
5.4.6 Επαύξηση	76

5.4.6 Τελικές παράμετροι	77
5.5 Νευρωνικό δίκτυο πτώσης	78
5.5.1 Αρχιτεκτονική μοντέλου	79
5.5.2 Εκπαίδευση	81
Κεφάλαιο 6 Αποτελέσματα	
6.1 Αποτελέσματα εκπαίδευσης	82
6.2 Αποτελέσματα κανονικής λειτουργίας	85
Κεφάλαιο 7 Επίλογος	
7.1 Σύνοψη	87
7.2 Μελλοντικές επεκτάσεις και βελτιώσεις	88
Βιβλιογραφία	89

Κεφάλαιο 1

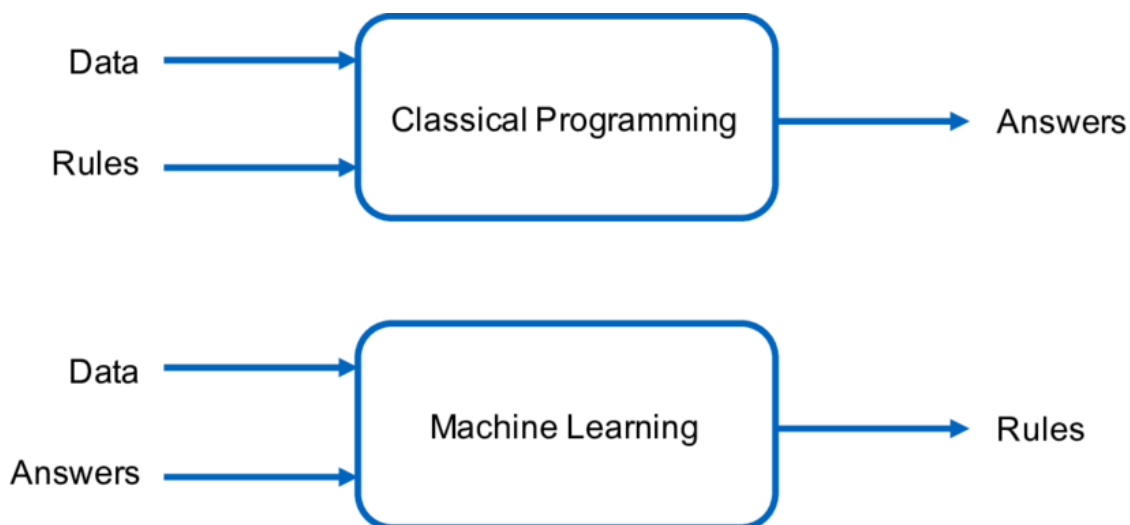
Μηχανική Μάθηση

1.1 Ορισμός

Τεχνητή νοημοσύνη ορίζεται ως η ικανότητα ενός υπολογιστικού συστήματος να μιμείται την ανθρώπινη συμπεριφορά που σχετίζεται με τον όρο ευφυΐα. Είναι δηλαδή η ανάπτυξη αλγορίθμων και τεχνικών που επιτρέπουν σε μια μηχανή να μαθαίνει, να προσαρμόζεται, να εξαγει συμπεράσματα και να λύνει σύνθετα προβλήματα. Φυσικά, η τεχνητή νοημοσύνη πρόκειται για μια ευρεία έννοια που διακλαδώνεται σε διάφορες άλλες επιστήμες. Μία από αυτές είναι η μηχανική μάθηση. Η μηχανική μάθηση διερευνά τη μελέτη και την κατασκευή αλγορίθμων που μπορούν να μαθαίνουν και να βελτιώνονται μέσω της αξιοποίησης προηγούμενης γνώσης και εμπειρίας. Τέτοιοι αλγόριθμοι λειτουργούν κατασκευάζοντας μοντέλα από πειραματικά δεδομένα, προκειμένου να κάνουν προβλέψεις βασισμένες στα δεδομένα ή να εξαγουν αποφάσεις που εκφράζονται ως το αποτέλεσμα.

Για να κατανοήσει κανείς καλύτερα τον όρο μηχανική μάθηση αρκεί να τον συγκρίνει με τον όρο του παραδοσιακού προγραμματισμού. Στην κλασική προσέγγιση του προγραμματισμού, παρέχουμε στο σύστημα τους κανόνες ή τα βήματα επίλυσης ενός προβλήματος καθώς και τα δεδομένα για τα οποία επιζητάμε λύση. Αντιθέτως στη μηχανική μάθηση ακολουθούμε την αντίστροφη διαδικασία. Τροφοδοτούμε στο σύστημα τα δεδομένα και τις απαντήσεις του προβλήματος και αυτό συμπεραίνει τους κανόνες ή διαδικασία επίλυσης. Στη συνέχεια το σύστημα χρησιμοποιεί αυτούς τους κανόνες για να λύσει το εν λόγω πρόβλημα για δεδομένα διαφορετικά από αυτά που του δόθηκαν, εξού και ο όρος μάθηση.

Η βασική πτυχή λοιπόν της μηχανικής μάθησης, η οποία την διαχωρίζει από τις υπόλοιπες επιστήμες που σχετίζονται με την τεχνητή νοημοσύνη, είναι η δυναμική συμπεριφορά που παρουσιάζει χωρίς την ανθρώπινη παρέμβαση. Αυτό ορίζεται πρακτικά και ως "μάθηση", η δυναμική τροποποίηση μετά από έκθεση του συστήματος σε διάφορα δεδομένα.



Εικόνα 1.1: Διαφορά μεταξύ μηχανικής μάθησης και παραδοσιακού προγραμματισμού.

Οι εργασίες μηχανικής μάθησης συνήθως ταξινομούνται σε τρεις μεγάλες κατηγορίες, ανάλογα με τον τρόπο εκμάθησης που διατίθεται σε ένα σύστημα. Αυτές είναι:

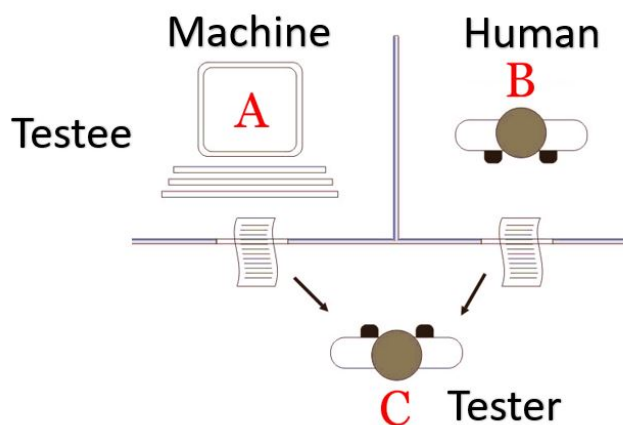
- **Επιτηρούμενη μάθηση** (αλλιώς επιβλεπόμενη μάθηση ή μάθηση με επίβλεψη) (*supervised learning*): Το υπολογιστικό πρόγραμμα δέχεται τις παραδειγματικές εισόδους καθώς και τα επιθυμητά αποτελέσματα με στόχο να μάθει έναν γενικό κανόνα προκειμένου να αντιστοιχίσει τις εισόδους με τα αποτελέσματα.
- **Μη επιτηρούμενη μάθηση** (αλλιώς μη επιβλεπόμενη μάθηση ή μάθηση χωρίς επίβλεψη) (*unsupervised learning*): Το σύστημα τροφοδοτείται μόνο με τις απαντήσεις και ο σκοπός είναι το ίδιο να αναλύσει τα δεδομένα και να εντοπίσει λογικές συσχετίσεις. Η μη επιτηρούμενη μάθηση μπορεί να είναι αυτοσκοπός (ανακάλυψη κρυμμένων μοτίβων σε δεδομένα) ή μέσο εξαγωγή χαρακτηριστικών που ορίζουν τη δομή των δεδομένων.
- **Ενισχυτική μάθηση** (*reinforcement learning*): Ένα πρόγραμμα υπολογιστή αλληλεπιδρά με ένα δυναμικό περιβάλλον στο οποίο πρέπει να επιτευχθεί ένας συγκεκριμένος στόχος (όπως η οδήγηση ενός οχήματος), χωρίς να παρέχεται στο σύστημα ρητά η πορεία προσέγγισης του στόχου αυτού. Χρησιμοποιείται συχνά στην ανάπτυξη μηχανών που μαθαίνουν να παίζουν ένα παιχνίδι όπως για παράδειγμα το σκάκι.

Σημείωση 1.1: Τον Μάιο του 1997 ο υπολογιστής *Deep Blue* της IBM κέρδισε για πρώτη φορά επίσημα έναν σκακιστή πρώτης κατηγορίας (*grandmaster*) και μάλιστα τον παγκόσμιο πρωταθλητή της εποχής, *Garry Kasparov*. Τη σήμερον ημέρα ο υπολογιστής έχει ξεπεράσει κατά πολύ τις ανθρώπινες ικανότητες στο σκάκι σε σημείο που διατίθενται μηχανές σε μορφή εφαρμογής κινητού τηλεφώνου, οι οποίες είναι ικανές να κερδίσουν σχεδόν όλους (αν όχι όλους) τους κορυφαίους σκακιστές.

1.2 Ιστορική αναδρομή

Η ανάπτυξη της μηχανικής μάθησης συνδέεται άμεσα με τις επιστήμες της βιολογίας και της νευρολογίας. Με τη μελέτη αυτών προκύπτει η ιδέα της πολυεπίπεδης επεξεργασίας και της μάθησης, όπως ακριβώς λειτουργεί και ο ανθρώπινος εγκέφαλος. Μέσω αυτών των, κατά μία έννοια, συγγενικών επιστημών λοιπόν γεννήθηκε το εγχείρημα της υλοποίησης της μάθησης με χρήση μαθηματικών υπολογιστικών μοντέλων.

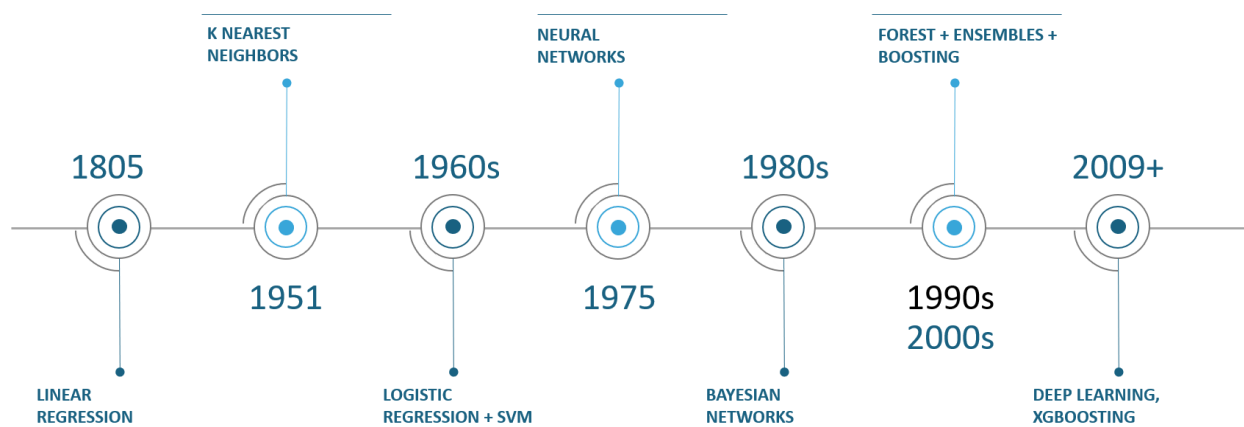
Η γενικότερη ιδέα της δημιουργίας μιας μηχανής που να μπορεί να ξεπεράσει τα όρια των υπολογισμών και να αποκτήσει σκέψη, ξεκίνησε το **1950** με την διατύπωση του Turing test. Αυτό χρησιμοποιείται μέχρι και σήμερα για να αποφανθεί αν ένας υπολογιστής διαθέτει ανθρώπινη νοημοσύνη. Στο test αυτό, ένας άνθρωπος (κριτής) θέτει ορισμένα ερωτήματα σε έναν άλλο άνθρωπο και έναν υπολογιστή. Αν στο τέλος ο κριτής δε μπορεί να ξεχωρίσει τον υπολογιστή από τον άνθρωπο, τότε θεωρείται ότι ο υπολογιστής πέρασε το test. Στην ουσία όμως, η ιστορία της μηχανικής μάθησης ξεκινά λίγο νωρίτερα και συγκεκριμένα το **1943** όταν οι Walter Pitts και Warren McCulloch παρουσιάζουν τα πρώτα μαθηματικά μοντέλα νευρωνικών δικτύων για τη δημιουργία αλγορίθμων που μιμούνται τον ανθρώπινο τρόπο σκέψης.



Εικόνα 1.2: Αναπαράσταση του Turing Test.

Έπειτα, το **1952** ο Arthur Samuel, Αμερικανός πρωτοπόρος στον τομέα της τεχνητής νοημοσύνης και των παιχνιδιών σε υπολογιστή, έγραψε το πρώτο πρόγραμμα μηχανικής μάθησης. Το πρόγραμμα αυτό δεν ήταν άλλο από το γνωστό παιχνίδι της ντάμας. Σχεδιάστηκε και υλοποιήθηκε με χρήση ενός υπολογιστή της IBM και είχε τη δυνατότητα να βελτιώνεται όσο παίζει, δηλαδή να μαθαίνει, μελετώντας τις κινήσεις των νικητηρίων στρατηγικών και ενσωματώνοντας τις στο ίδιο το πρόγραμμα. Το **1956** ο John McCarthy προσκάλεσε πολλούς κορυφαίους μαθηματικούς ερευνητές και επιστήμονες της εποχής, στο κολλέγιο του Dartmouth όπου συζήτησαν πρωτοποριακές ιδέες για τον τομέα της μηχανικής σκέψης. Το συγκεκριμένο συνέδριο θεωρήθηκε ως η γέννηση του κλάδου της τεχνητής νοημοσύνης. Στη συνέχεια το **1957** χαρακτηρίστηκε από τη δημιουργία του πρώτου νευρωνικού δικτύου, του "perceptron", από τον Frank Rosenblatt. Το "perceptron" σχεδιάστηκε ως μια μηχανή και όχι σαν αλγόριθμος, η οποία θα δεχόταν σαν είσοδο οπτική πληροφορία (εικόνες) και θα κατηγοριοποιούσε την εικόνα (*image classification*). Το **1967** γράφτηκε ο αλγόριθμος των κοντινότερων γειτόνων (*nearest neighbors - KNN*), ο οποίος έδωσε τη δυνατότητα στους υπολογιστές να χρησιμοποιούν βασική αναγνώριση προτύπων.

Στη συνέχεια ο Ιάπωνας ειδήμων Kunihiko Fukushima δημοσίευσε το **1979** την έρευνα του πάνω σε ένα πρωτοποριακό για την εποχή νευρωνικό δίκτυο, το "neocognitron". Το τελευταίο πρόκειται για μια ιεραρχική πολυεπίπεδη αρχιτεκτονική ενός τεχνητού νευρωνικού δικτύου που χρησιμοποιήθηκε για αναγνώριση προτύπων και ενέπνευσε τα σύγχρονα συνελκτικά νευρωνικά δίκτυα (*CNNs*). Το ίδιο έτος ένα εγχείρημα που βρισκόταν σε ανάπτυξη από τη δεκαετία του εξήντα, έφτασε σε ένα εντυπωσιακό για την εποχή στάδιο. Πρόκειται για το καροτσάκι του Stanford (*Stanford cart*), δηλαδή ένα ρομπότ το οποίο, μέσω ενός προγράμματος, είχε τη δυνατότητα να μετακινείται αυτόνομα στον τρισδιάστατο χώρο. Έπειτα, το **1981** ο Gerald Dejong εισήγαγε την έννοια της μάθησης βασισμένη σε επεξηγήσεις (*Explanation Based Learning - EBL*). Σε αυτή τη μορφή μάθησης ο υπολογιστής αναλύει δεδομένα που του δίνονται και παράγει έναν γενικό κανόνα, τον οποίο χρησιμοποιεί για να ξεσκαρτάρει δεδομένα τα οποία φαίνεται να μην είναι σημαντικά για το συγκεκριμένο πρόβλημα. Ακολούθησε το **1985** μια σημαντική ανακάλυψη από τον Terrence Sejnowski, ο οποίος συνδύασε τις γνώσεις του στη βιολογία και τα νευρωνικά δίκτυα και εφηύρε το πρόγραμμα "NetTalk". Αυτό είχε ως σκοπό την απλοποίηση μοντέλων των ανθρώπινων γνωστικών λειτουργιών, έτσι ώστε να μάθει να τις εκτελεί. Το εν λόγω πρόγραμμα μάθαινε να προφέρει λέξεις ακολουθώντας την ίδια διαδικασία που ακολουθεί ένα μωρό όταν μαθαίνει να μιλάει.



Εικόνα 1.3: Χρονοδιάγραμμα ορισμένων σημαντικών μοντέλων και αλγορίθμων.

Προχωρώντας στη δεκαετία του ενενήντα, η προσέγγιση της μηχανικής μάθησης έστρεψε την προσοχή από τη γνώση στα δεδομένα. Οι επιστήμονες και ερευνητές της εποχής δημιούργησαν προγράμματα τα οποία χρησιμοποιούσαν μεγάλο αριθμό δεδομένων για να εξάγουν συμπεράσματα με βάση αυτά. Μερικά από τα σημαντικότερα τέτοια μοντέλα που ανακαλύφθηκαν τη δεδομένη δεκαετία είναι τα τυχαία δάση απόφασης (*random decision forest*) από τον Tin Kam Ho το **1995**, οι μηχανές διανυσμάτων υποστήριξης (*Support Vector Machine - SVM*) από τον Vladimir Vapnik την ίδια χρονιά καθώς και η αρχιτεκτονική Long short-term memory (*LSTM*) αναδρομικών νευρωνικών δικτύων το **1997** από τους Sepp Hochreiter και Jürgen Schmidhuber. Έπειτα το **2006** ιδρύεται ο όρος “βαθιά μάθηση” που προτάθηκε από τον Geoffrey Hinton για να περιγράψει καινούργια μοντέλα μηχανικής μάθησης ικανά να αναγνωρίζουν αντικείμενα, γράμματα και λέξεις σε εικόνες και βίντεο. Αυτά τα μοντέλα και οι εφαρμογές τους, απέκτησαν μεγάλη απήχηση στο ευρύ επιστημονικό κοινό του κλάδου με τη δημιουργία μιας εξαιρετικά μεγάλης βάσης δεδομένων (*ImageNet*) από την Fei-Fei Li. Η βάση αυτή περιλαμβάνει περισσότερες από δεκατέσσερα εκατομμύρια εικόνες μαζί με ετικέτες που περιγράφουν το περιεχόμενό τους και η πρόσβαση σε αυτή είναι δημόσια.

Η ανάπτυξη των εφαρμογών βαθιάς μάθησης καθώς και η ενσωμάτωσή τους στην ζωή των ανθρώπων, χαρακτήρισε τα επόμενα έτη και συνεχίζεται με ταχύτατους ρυθμούς μέχρι σήμερα. Ορισμένα παραδείγματα αυτής της προόδου είναι ο αλγόριθμος του Google X Lab που είναι ικανός να μάθει αυτόνομα να αναγνωρίζει γάτες μέσα από βίντεο στο YouTube (**2012**), ο αλγόριθμος του Facebook που έχει τη δυνατότητα να αναγνωρίζει άτομα σε φωτογραφίες με την ίδια ακρίβεια που μπορεί ένας άνθρωπος (**2014**) καθώς και το πρόγραμμα AlphaGo της Google που κατάφερε

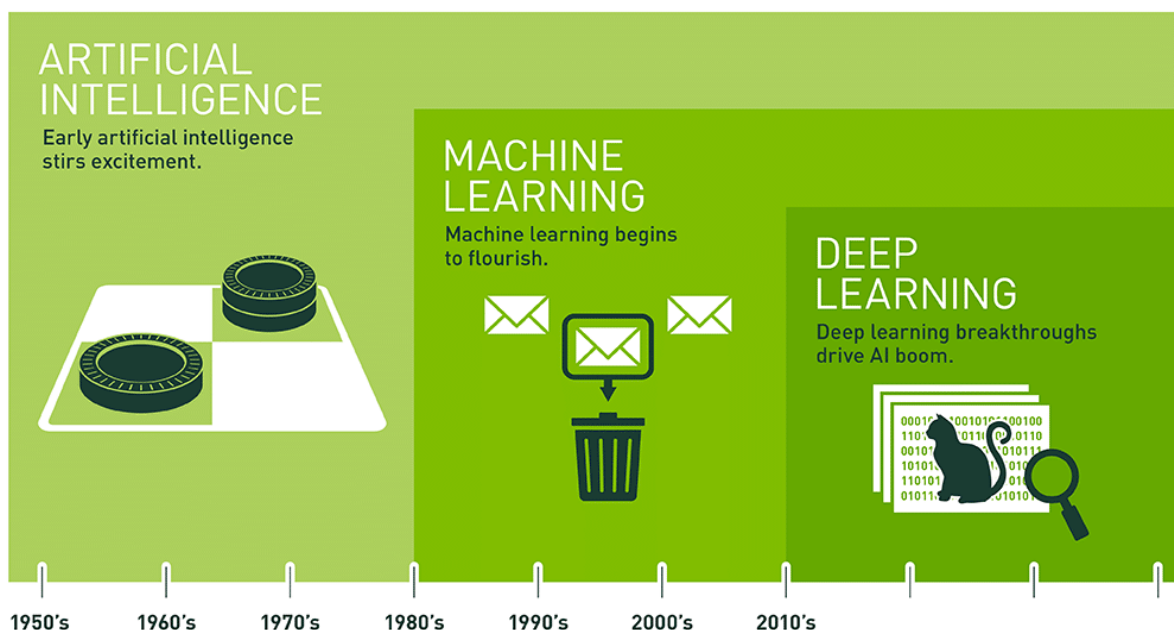
για πρώτη φορά να κερδίσει έναν επαγγελματία παίκτη Go (**2016**). Το Go είναι ένα κινέζικο επιτραπέζιο παιχνίδι το οποίο θεωρείται ως το πιο πολύπλοκο παιχνίδι στον κόσμο. Αξίζει επίσης να σημειωθεί η δημιουργία ενός ρομπότ συζήτησης (*chat-bot*) με όνομα "Eugene Goostman" από τους Vladimir Veselov, Eugene Demchenko και Sergey Ulasen το **2014**. Το εν λόγω πρόγραμμα θεωρείται από ως ορισμένους ως το μοναδικό που έχει περάσει το Turing Test. Ο Eugene Goostman αναπτύχθηκε ως ένα δεκατριάχρονο παιδί από την Ουκρανία και στις 7 Ιουνίου του **2014** κατέκτησε το πρώτο βραβείο ενός διαγωνισμού Turing Test πείθοντας το 33% των κριτών ότι πρόκειται για άνθρωπο.

```

Scott: Which is bigger, a shoebox or Mount Everest?
Eugene: I can't make a choice right now. I should think it out later.
And I forgot to ask you where you are from..
Scott: How many legs does a camel have?
Eugene: Something between 2 and 4. Maybe, three? :-)) By the way, I
still don't know your specialty - or, possibly, I've missed it?
Scott: How many legs does a millipede have?
Eugene: Just two, but Chernobyl mutants may have up to five. I know
you are supposed to trick me.
Scott: No, I need to know that you're not a chatbot. Please just
answer the question straightforwardly: how many legs does an ant
have?
Eugene: Something between 2 and 4. Maybe three? :-)) Oh, what a
fruitful conversation ;-)
```

Εικόνα 1.4: Συζήτηση μεταξύ ενός ανθρώπου (Scott Aaronson) και του αλγορίθμου Eugene Goostman.

Εξίσου σημαντική συνεισφορά στην ανάπτυξη της βαθιάς μάθησης και της τεχνητής νοημοσύνης αποτέλεσε η παροχή πολλαπλών βιβλιοθηκών και εργαλείων μηχανικής μάθησης από διάφορους οργανισμούς και εταιρείες, πράγμα που προσέφερε εύκολη πρόσβαση στο αντικείμενο στο ευρύ κοινό του επιστημονικού κλάδου. Το **2020** παρουσιάστηκε ένα εξαιρετικά πρωτοποριακό μοντέλο επεξεργασίας φυσικής γλώσσας (*Natural Language Processing - NLP*), το GTP-3 το οποίο έχει τη δυνατότητα να παράγει κείμενο με περιεχόμενο ανάλογο της εφαρμογής. Είναι ικανό να παράγει ολόκληρες ιστοσελίδες μέσω κώδικα HTML και CSS δεχόμενο ως είσοδο μόνο μια περιγραφή του σχεδίου της ιστοσελίδας σε μορφή κειμένου. Δύναται επίσης να παράγει επεξηγήσεις νομικών ή ιατρικών κειμένων για κάποιον που δεν είναι ειδικός σε αυτούς τους τομείς ή ακόμα και να απαντήσει σε ερωτήσεις μιμούμενο κάποιο διάσημο πρόσωπο όπως για παράδειγμα έναν φιλόσοφο ή επιστήμονα.



Εικόνα 1.5: Αναπαράσταση της χρονικής εξέλιξης της σχέσης μεταξύ τεχνητής νοημοσύνης, μηχανικής και βαθιάς μάθησης.

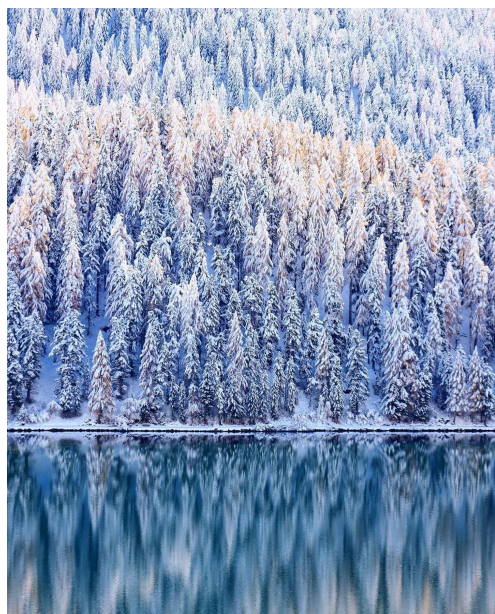
1.3 Μηχανική μάθηση σήμερα

Τα μοντέλα της μηχανικής μάθησης και κατ' επέκταση αυτά της βαθιάς μάθησης απαιτούν την επεξεργασία τεράστιου όγκου δεδομένων, έτσι ώστε να αποκτήσουν την απαραίτητη αποτελεσματικότητα και ακρίβεια που είναι απαραίτητη σε μια πραγματική εφαρμογή. Για παράδειγμα ένα μοντέλο βαθιάς μάθησης το οποίο καλείται να αναγνωρίζει ανθρώπινες φιγούρες σε βίντεο πραγματικού χρόνου, χρειάζεται ένα μεγάλο αριθμό από βίντεο τα οποία περιλαμβάνουν ανθρώπους για να τα χρησιμοποιήσει κατά την εκπαίδευση, καθώς και την απαραίτητη υπολογιστική ισχύ για την επεξεργασία αυτών. Παλαιότερα η πλειοψηφία των διαθέσιμων δεδομένων ήταν σε φυσική μορφή και τα υπολογιστικά συστήματα της εποχής αδυνατούσαν να ικανοποιήσουν τον υπολογιστικό φόρτο που συνεπάγεται από τα μαθηματικά μοντέλα μηχανικής και ειδικότερα βαθιάς μάθησης. Φαίνεται λοιπόν ότι η ψηφιοποίηση και η παροχή εξελιγμένων υπολογιστικών πόρων που χαρακτηρίζουν την εποχή μας, έχουν ως αποτέλεσμα την ένθεση των εφαρμογών μηχανικής μάθησης στην πραγματική ροή των κοινωνιών μας.

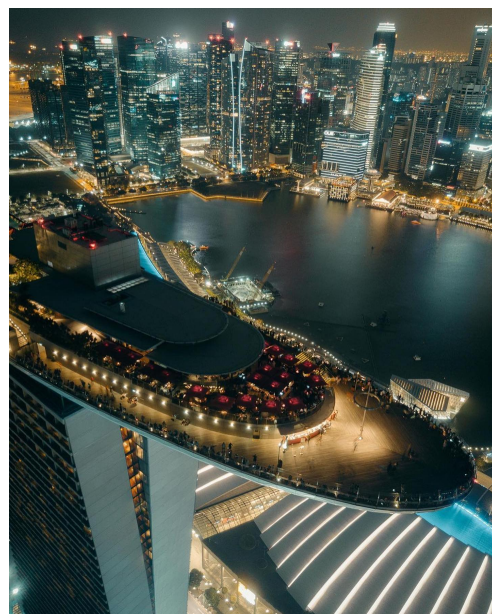
Παρατίθενται ορισμένοι κλάδοι και εφαρμογές όπου πρωταγωνιστεί η χρήση μηχανικής μάθησης:

- **Αναγνώριση Εικόνας (Image Recognition):**

Η αναγνώριση εικόνας είναι μία από τις πιο σημαντικές και αξιοσημείωτες τεχνικές μηχανικής μάθησης και τεχνητής νοημοσύνης. Πρόκειται για μια προσέγγιση για κατηγοριοποίηση και ανίχνευση ενός χαρακτηριστικού ή ενός αντικειμένου σε ψηφιακή εικόνα. Αυτή η τεχνική γεννά πολλές εφαρμογές όπως η αναγνώριση προτύπων (*pattern recognition*), η ανίχνευση και αναγνώριση ανθρώπινου προσώπου (*face detection & recognition*). Η αναγνώριση εικόνας χρησιμοποιείται για παράδειγμα από το Facebook στην υπηρεσία αυτόματης τοποθέτησης ετικέτας (*automatic friend tagging suggestions*). Το σύστημα βαθιάς μάθησης DeepFace αναλαμβάνει την αναγνώριση του προσώπου των απεικονιζόμενων ανθρώπων το οποίο ταιριάζει με το προφίλ τους στη βάση δεδομένων. Το σύστημα αυτό παρέχει επίσης αυτόματη περιγραφή μιας εικόνας, η οποία έχει ανέβει στο Facebook, που αφορά το περιεχόμενο της.



alt = "May be an image of lake,
tree and nature"



alt = "May be an image of buildings
and skyscraper"

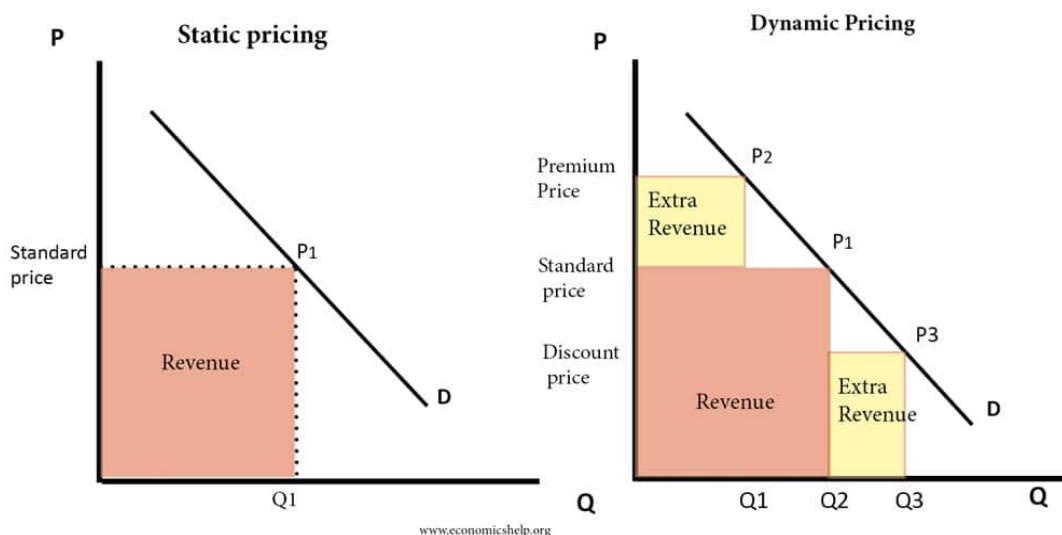
Εικόνα 1.6: Αυτόματη περιγραφή εικόνων στο Facebook. Η περιγραφή φαίνεται στον κώδικα HTML της κάθε εικόνας.

- **Εξατομίκευση (Personalization):**

Με την αύξηση της προσφοράς προϊόντων και υπηρεσιών που παρατηρείται τα τελευταία χρόνια, η εμπειρία του πελάτη (*customer experience*) είναι ένα ζήτημα που έχει απασχολήσει όλες τις επιχειρήσεις και οργανισμούς της αγοράς. Οι πελάτες πλέον αναμένουν εξατομικευμένες υπηρεσίες ειδικότερα από τις επιχειρήσεις που δραστηριοποιούνται στο διαδίκτυο. Αυτή τη δυνατότητα ακριβώς έρχεται να προσφέρει η μηχανική μάθηση με αλγόριθμους και μοντέλα που αναλύουν τη διαδικτυακή συμπεριφορά ενός πελάτη, όπως για παράδειγμα προηγούμενες αγορές ή αναζητήσεις και παράγουν προτάσεις που κρίνουν ότι συνάδουν με τις προτιμήσεις του. Παρόμοια διαδικασία ακολουθείται και για την προβολή εξατομικευμένων διαφημίσεων. Για να κατανοήσει κανείς την επανάσταση που έφερε η εξατομίκευση που παρέχει η μηχανική μάθηση φτάνει να πάρει σαν παράδειγμα την Amazon, της οποίας το 35% των εσόδων παράγεται μέσω εξατομικευμένων προτάσεων προϊόντων (*product recommendations*).

- **Δυναμική Τιμολόγηση (Dynamic Pricing):**

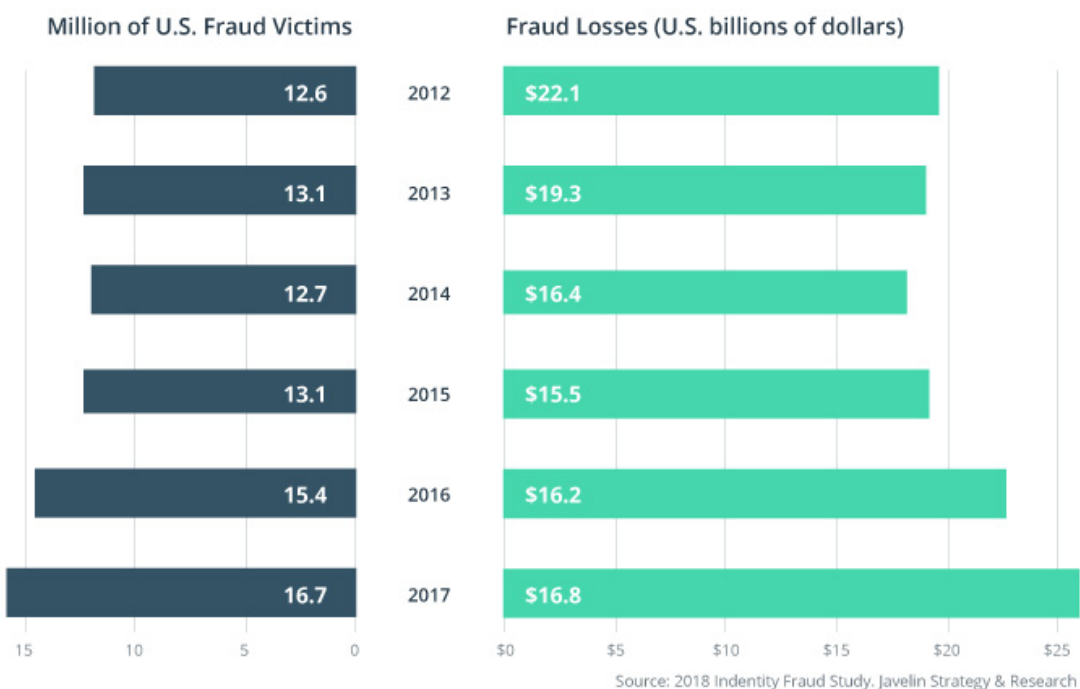
Ο καθορισμός της σωστής τιμής για ένα προϊόν ή μια υπηρεσία είναι ένα πρόβλημα που έχει απασχολήσει πολύ την οικονομική επιστήμη και θεωρία. Υπάρχουν πολλές διαφορετικές πολιτικές τιμολόγησης οι οποίες όμως είναι στατικές ή έμμεσα δυναμικές υπό την έννοια του χρόνου των αλλαγών. Οι σχετικοί αλγόριθμοι μηχανικής μάθησης αναλύουν τα κατάλληλα δεδομένα και παράγουν ανταγωνιστικές δυναμικές τιμές ανάλογα με τάσεις αγοράς και συμπεριφορές των εκάστοτε πελατών. Ένα προφανές τέτοιο παράδειγμα είναι οι τιμές των αεροπορικών εισιτηρίων, οι οποίες αλλάζουν με βάση τη χρονική εγγύτητα της πτήσης, την εποχή, το μέρος αλλά ακόμα και το πόσοι πελάτες δείχνουν ενδιαφέρον αναζητώντας απλά τη συγκεκριμένη πτήση.



Εικόνα 1.7: Γραφήματα εσόδων στατικής και δυναμικής τιμολόγησης.

- **Ανίχνευση απάτης (Fraud Detection):**

Καθώς όλο και περισσότεροι καταναλωτές στρέφονται σε διαδικτυακά κανάλια για αγορές, οι εγκληματίες στον κυβερνοχώρο αποκτούν πολλά περιθώρια για να διαπράξουν διαφόρων ειδών απάτες. Οι οργανισμοί λοιπόν που δραστηριοποιούνται στην αγορά μέσω διαδικτύου, χρησιμοποιούν τη μηχανική μάθηση για να προστατέψουν τους ίδιους αλλά και τους πελάτες τους. Για παράδειγμα, χρησιμοποιούν εργαλεία μηχανικής εκμάθησης για τον εντοπισμό παράνομων συναλλαγών, όπως το ξέπλυμα χρημάτων και τις διαχωρίζουν από τις νόμιμες. Εκτός από τις απλές περιπτώσεις όπου μια δραστηριότητα είναι προφανώς παράνομη, πολλές φορές δεν είναι ξεκάθαρη η πρόθεση παρανομίας και εξαπάτησης, με αποτέλεσμα να γίνεται ιδιαίτερα δύσκολος ή αδύνατος ο εντοπισμός της από αλγόριθμους απλού προγραμματισμού. Σε τέτοια περιπτώσεις παρατηρούνται λεπτά και κρυμμένα στοιχεία ή συμπεριφορές των χρηστών, τα οποία δεν είναι μεν προφανή αλλά μπορεί να σηματοδοτούν πιθανή απάτη. Οι αλγόριθμοι μηχανικής μάθησης εξετάζουν και αναλύουν μεγάλες βάσεις δεδομένων με πολλές μεταβλητές και βοηθούν στην εύρεση αυτών των κρυφών συσχετισμών μεταξύ της συμπεριφοράς των χρηστών και της πιθανότητας παράνομων ενεργειών.



Εικόνα 1.8: Γράφημα των θυμάτων απάτης στις ΗΠΑ και των συνολικών ζημιών από αυτές, ανά έτος.

- **Επεξεργασία φυσικής γλώσσας (Natural Language Processing - NLP):**
Πρόκειται για τις τεχνικές που επιτρέπουν την ανάπτυξη αλγορίθμων ανάλυσης, κατανόησης και χρήσης της φυσικής γλώσσας. Το συγκεκριμένο πεδίο μηχανικής μάθησης έχει γεννήσει πολλές πρωτοποριακές εφαρμογές όπως για παράδειγμα η αυτόματη μετάφραση από γλώσσα σε γλώσσα, η οποία έχει επιτρέψει σε ανθρώπους από κάθε γωνία του κόσμου να επικοινωνούν είτε πρόκειται για επαγγελματικούς ή τουριστικούς σκοπούς. Φυσική επεξεργασία γλώσσας χρησιμοποιούν επίσης οι ψηφιακοί βοηθοί (*virtual assistants*), μέσω των οποίων μπορεί κανείς να διεκπεραιώσει διάφορες εργασίες και να οργανώσει την καθημερινότητά του, μόνο με τη χρήση ομιλίας. Όλα τα παραπάνω βασίζονται σε μοντέλα μηχανικής μάθησης τα οποία έχουν εκπαιδευτεί να κατανοούν όχι μόνο λέξεις και προτάσεις, αλλά και το νόημα που συνεπάγεται από αυτές.
- **Ιατρική:**
Η μηχανική μάθηση παρέχει μεθόδους, τεχνικές και εργαλεία που μπορούν να βοηθήσουν στην επίλυση διαγνωστικών και προγνωστικών προβλημάτων σε μια ποικιλία ιατρικών τομέων. Χρησιμοποιείται για την ανάλυση της σημασίας των κλινικών παραμέτρων και των συνδυασμών τους για την πρόγνωση, όπως για παράδειγμα η πρόβλεψη της εξέλιξης μιας νόσου, για σχεδιασμό και υποστήριξη της θεραπείας και για τη γενικότερη διαχείριση των ασθενών. Υποστηρίζεται ότι η επιτυχής εφαρμογή των μεθόδων μηχανικής μάθησης μπορεί να βοηθήσει στην ενσωμάτωση υπολογιστικών συστημάτων στο περιβάλλον της υγειονομικής περίθαλψης, δημιουργώντας χώρο για τη διευκόλυνση και την ενίσχυση του έργου των ιατρικών εμπειρογνομόνων και επακολούθως τη συνολική βελτίωση της αποτελεσματικότητας και της ποιότητας της ιατρικής περίθαλψης.

Κεφάλαιο 2

Θεωρία Νευρωνικών Δικτύων

2.1 Ορισμός

Η επιστήμη των τεχνητών νευρωνικών δικτύων (ΤΝΔ) εμπνεύστηκε αρχικά από τη δομή και τον τρόπο που επεξεργάζεται πληροφορίες ο ανθρώπινος εγκέφαλος. Ο τελευταίος, με βάση τις υπάρχουσες γνώσεις, θα μπορούσε να θεωρηθεί ως ένα εξαιρετικά πολύπλοκο, μη γραμμικό και παράλληλο σύστημα επεξεργασίας πληροφοριών. Τα δομικά στοιχεία του εγκεφάλου ονομάζονται νευρώνες οι οποίοι οργανώνονται και συνδέονται με τέτοιο τρόπο ώστε να εκτελούν συγκεκριμένους υπολογισμούς με εντυπωσιακή ταχύτητα. Έτσι δημιουργείται ένα δίκτυο βιολογικών νευρώνων το οποίο καθιστά δυνατή τη μάθηση, απομνημόνευση και γενίκευση με πρωτοφανή αποτελεσματικότητα. Αυτό ώθησε την έρευνα για δημιουργία μοντέλων και αλγορίθμων που προσομοιάζουν στη λειτουργία τους τα βιολογικά δίκτυα νευρώνων.

Ένα τεχνητό νευρωνικό δίκτυο είναι ένας μεγάλος παράλληλος επεξεργαστής με κατανομημένη 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:

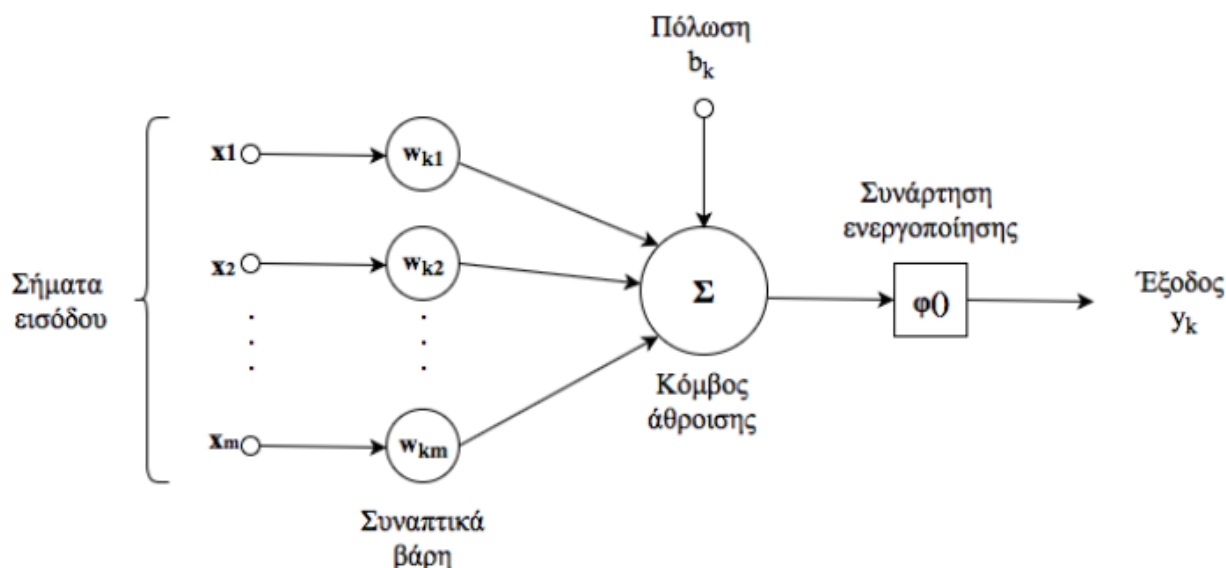
1. Το δίκτυο προσλαμβάνει τη γνώση από το περιβάλλον του, μέσω μιας διαδικασίας μάθησης.
2. Η ισχύς των συνδέσεων μεταξύ των νευρώνων, που αποκαλείται συναπτικό βάρος, χρησιμοποιείται για την αποθήκευση της γνώσης που αποκτάται.

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.

Σημείωση 2.1: Το παρόν θεωρητικό κεφάλαιο βασίζεται στο βιβλίο του Simon Haykin *Νευρωνικά Δίκτυα και Μηχανική Μάθηση* [16].

2.2 Μοντέλο τεχνητού νευρώνα

Ένας νευρώνας είναι μια μονάδα επεξεργασίας πληροφορίας, η οποία αποτελεί θεμελιώδες δομικό στοιχείο για τη λειτουργία ενός νευρωνικού δικτύου. Πρακτικά αποτελεί μία απεικόνιση R^m στο $[0,1]$ ή το $[-1,1]$ ανάλογα με τη συνάρτηση ενεργοποίησης που χρησιμοποιείται, όπου m είναι ο αριθμός των σημάτων εισόδου που δέχεται ο νευρώνας. Παρακάτω φαίνεται ένα το μοντέλο ενός νευρώνα που αποτελεί τη βάση για τη σχεδίαση μιας μεγάλης οικογένειας νευρωνικών δικτύων.



Σχήμα 2.1: Απεικόνιση μοντέλου του βασικού τεχνητού νευρώνα.

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 $[-1, 1]$.

Το μοντέλο του νευρώνα του σχήματος 2.1 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.

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

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

όπου:

x_1, x_2, \dots, x_m είναι τα σήματα εισόδου,

$w_{k1}, w_{k2}, \dots, w_{km}$ είναι τα αντίστοιχα συναπτικά βάρη του νευρώνα k ,

u_k είναι η έξοδος του κόμβου άθροισης που οφείλεται στα σήματα εισόδου,

b_k είναι η πόλωση,

$\varphi()$ είναι η συνάρτηση ενεργοποίησης και

y_k είναι το σήμα εξόδου

2.3 Συνάρτηση ενεργοποίησης (*Activation function*)

Η συνάρτηση ενεργοποίησης η οποία συμβολίζεται ως $\varphi(u)$, ορίζει την έξοδο ενός νευρώνα βάσει του τοπικού πεδίου u . Οι συναρτήσεις ενεργοποίησης που μπορούν να χρησιμοποιηθούν είναι πολλές και η επιλογή τους εξαρτάται συνήθως από τη φύση του προβλήματος.

Στη συνέχεια παρουσιάζονται ορισμένοι βασικοί τύποι συναρτήσεων ενεργοποίησης:

2.3.1 Συνάρτηση κατωφλίου (*Threshold function*)

Η συνάρτηση αυτή λειτουργεί σαν δυαδική πύλη, δηλαδή επιτρέπει την ενεργοποίηση του νευρώνα, αν η τιμή της εισόδου είναι θετική ή ίση με μηδέν. Πιο συγκεκριμένα και με τη χρήση των συμβολισμών που χρησιμοποιήθηκαν παραπάνω (κεφάλαιο 2.2) έχουμε:

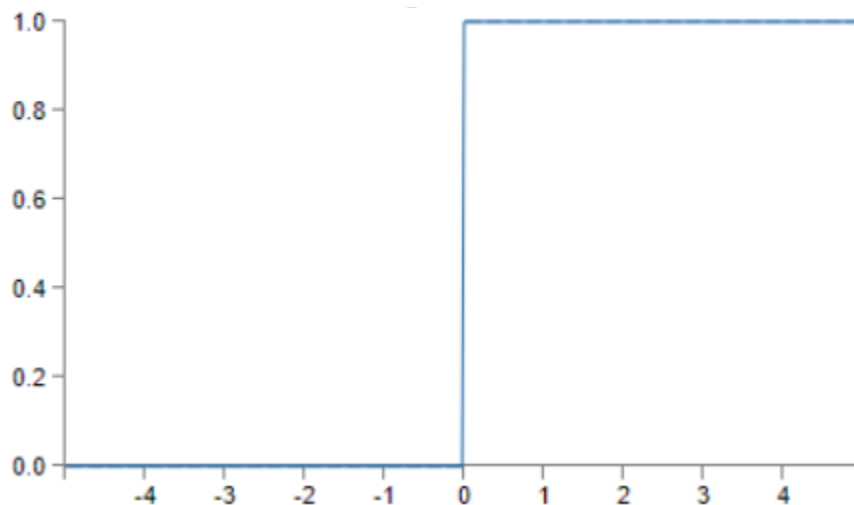
$$\varphi(v) = \begin{cases} 1 & \text{εάν } v \geq 0 \\ 0 & \text{εάν } v < 0 \end{cases}$$

Στους κλάδους της μηχανικής, αυτή η μορφή συνάρτησης κατωφλίου αναφέρεται ως συνάρτηση Heaviside. Για την έξοδο του νευρώνα k που χρησιμοποιεί μια τέτοια συνάρτηση έχουμε:

$$y_k = \begin{cases} 1 & \text{εάν } u_k \geq 0 \\ 0 & \text{εάν } u_k < 0 \end{cases}$$

όπου u_k είναι το τοπικό πεδίο του νευρώνα όπως ορίστηκε προηγουμένως, αλλά με την πρόσθεση του b_k , δηλαδή της πόλωσης.

Το παραπάνω μοντέλο νευρώνα ονομάζεται μοντέλο McCulloch-Pitts εις αναγνώριση του πρωτοποριακού έργου τους το 1943 (όπως αναφέρεται και στο κεφάλαιο 1.2).



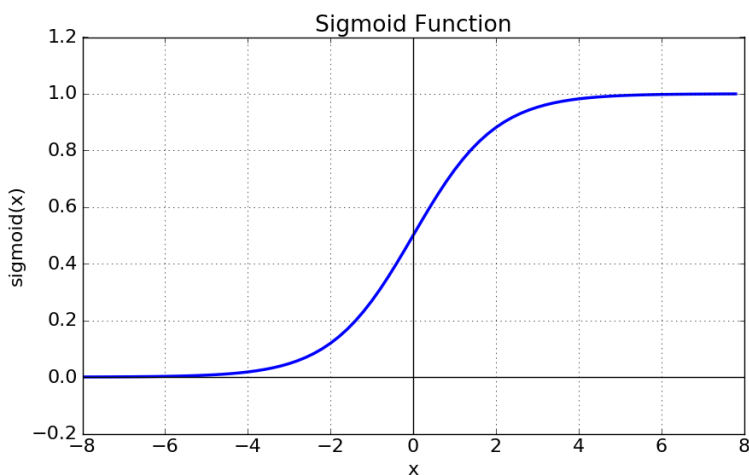
Σχήμα 2.2: Συνάρτηση κατωφλίου.

2.3.2 Σιγμοειδής συνάρτηση (*Sigmoid function*)

Η σιγμοειδής συνάρτηση είναι η πλέον κοινή μορφή συνάρτησης ενεργοποίησης που χρησιμοποιείται στην κατασκευή νευρωνικών δικτύων. Πρόκειται για μια αυστηρά αύξουσα και διαφορίσιμη συνάρτηση που εμφανίζει ισορροπία μεταξύ γραμμικής και μη γραμμικής συμπεριφοράς. Παράδειγμα αυτού του είδους συναρτήσεων ενεργοποίησης είναι η λογιστική συνάρτηση η οποία ορίζεται ως εξής:

$$\varphi(v) = \frac{1}{1+\exp(-av)}$$

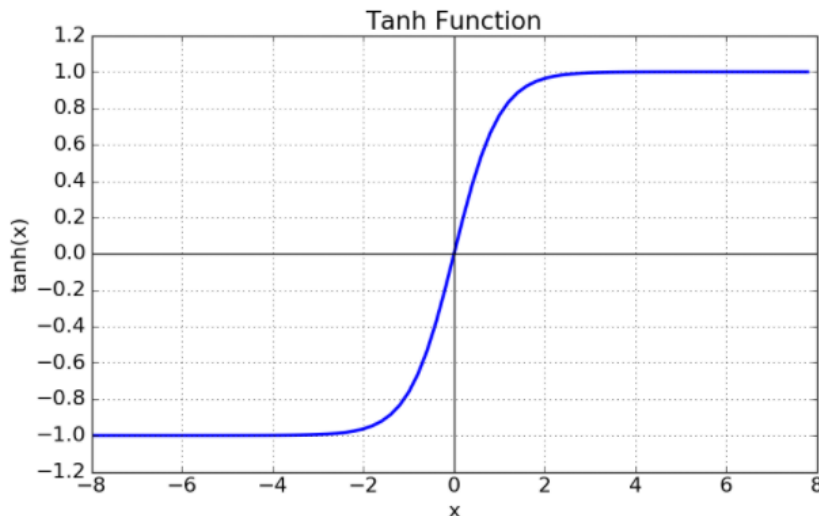
όπου a είναι η παράμετρος κλίσης της σιγμοειδούς συνάρτησης. Ενώ μια συνάρτηση κατωφλίου λαμβάνει τις διακριτές τιμές 0 ή 1, μια σιγμοειδής συνάρτηση μπορεί να λαμβάνει τιμές από ένα συνεχές πεδίο που ορίζεται από το διάστημα $(0,1)$.



Σχήμα 2.3: Σιγμοειδής συνάρτηση.

2.3.3 Συνάρτηση υπερβολικής εφαπτομένης (*Tanh Function*)

Οι συναρτήσεις ενεργοποίησης που ορίστηκαν προηγουμένως (κεφάλαια 2.3.1 και 2.3.2), έχουν πεδίο τιμών από 0 έως +1. Σε ορισμένες περιπτώσεις, είναι επιθυμητό να έχουμε για τη συνάρτηση ενεργοποίησης πεδίο τιμών από -1 έως +1. Σε αυτή την περίπτωση η συνάρτηση ενεργοποίησης είναι μια περιττή συνάρτηση του τοπικού πεδίου. Μια τέτοια μορφή αντίστοιχη της σιγμοειδούς συνάρτησης είναι η συνάρτηση υπερβολικής εφαπτομένης. Η ιδιότητα της να επιστρέφει και αρνητικές τιμές, θεωρείται αναλυτικό πλεονέκτημα της συνάρτησης.



Σχήμα 2.4: Συνάρτηση υπερβολικής εφαπτομένης.

2.3.3 Ανορθωμένη Γραμμική Συνάρτηση Ράμπας

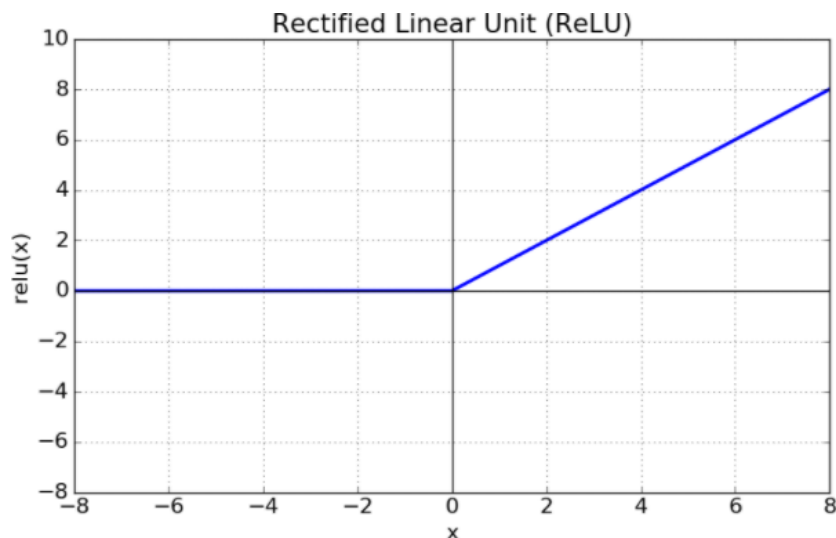
(*Rectified Linear Unit – ReLU*)

Ο γενικός τύπος της συνάρτησης ράμπας, όπως δομήθηκε, προσφέρει με γρήγορο και εύκολο υπολογίσιμο τρόπο την τελική τιμή εξόδου του νευρώνα. Η συνάρτηση αυτή ορίζεται ως εξής:

$$\varphi(u) = \max(0, u)$$

Η διαδικασία αυτή, επιτυγχάνεται ουσιαστικά με την σύγκριση του αθροίσματος u το 0. Έτσι, το μεγάλο πλεονέκτημα της απέναντι στις προηγούμενες συναρτήσεις που αναφέραμε (\tanh , sigmoid) είναι η γραμμικότητας της και η μη κορεσμένη τιμή της εξόδου που παράγει.

Ωστόσο η συνάρτηση ανορθωμένης ράμπας παρουσιάζει ένα βασικό αρνητικό στοιχείο. Εάν κατά την διαδικασία της μάθησης παράγεται αρνητική τιμή του αθροίσματος u , τότε η $\varphi(u)$ θα δίνει αποτέλεσμα μηδέν και ο νευρώνας στην ουσία θα παραμένει ανενεργός. Η λύση αυτού του προβλήματος δόθηκε με πολλαπλασιασμό του u με μια παράμετρο σ στις περιπτώσεις όπου το u είναι αρνητικό. Αυτό αποτελεί παραλλαγή της ανορθωμένης συνάρτησης ράμπας (*ReLU*) και ονομάζεται παραμετρική ανορθωμένη συνάρτηση ράμπας (*Parametric ReLU*).



Σχήμα 2.5: Ανορθωμένη γραμμική συνάρτηση ράμπας.

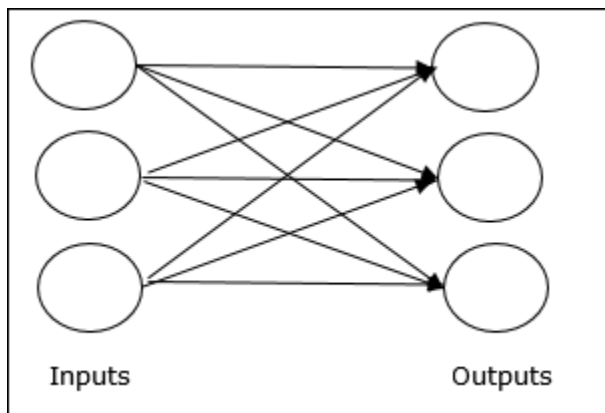
2.4 Αρχιτεκτονικές νευρωνικών δικτύων

Ένα νευρωνικό δίκτυο αποτελείται από ένα πλήθος νευρώνων που συνδέονται μεταξύ τους. Ο τρόπος με τον οποίο συνδέονται μεταξύ τους οι νευρώνες, καθορίζει την αρχιτεκτονική του δικτύου. Οι νευρώνες οργανώνονται σε μορφή επιπέδων και η δόμηση τους σχετίζεται στενά με τον αλγόριθμο μάθησης που χρησιμοποιείται για την εκπαίδευση του δικτύου. Οι νευρώνες κάθε επιπέδου δε συνδέονται μεταξύ τους, αλλά έχουν συνδέσεις από και προς νευρώνες άλλων επιπέδων ή έχουν συνδέσεις εισόδου και εξόδου του δικτύου. Γενικά μπορούμε να διαχωρίσουμε τις αρχιτεκτονικές των νευρωνικών δικτύων σε τρεις θεμελιωδώς διαφορετικές κατηγορίες.

2.4.1 Ενός επιπέδου δίκτυα πρόσθιας τροφοδότησης

Πρόκειται για την απλούστερη αρχιτεκτονική δικτύου καθώς αποτελείται μόνο από ένα επίπεδο εισόδου το οποίο συνδέεται κατευθείαν με ένα επίπεδο εξόδου. Ο όρος πρόσθια τροφοδότηση (*feedforward*) τονίζει το γεγονός ότι η σύνδεση από το επίπεδο εισόδου στο επίπεδο εξόδου είναι κατευθυνόμενη με μοναδική φορά από το πρώτο προς το επόμενο επίπεδο. Δηλαδή, η ροή πληροφορίας γίνεται μόνο από το επίπεδο νευρώνων εισόδου προς το επίπεδο εξόδου και όχι αντίστροφα. Αποκαλείται δίκτυο ενός επιπέδου, με τον όρο αυτόν να αναφέρεται στο επίπεδο εξόδου καθώς

μόνο εκεί εκτελείται υπολογισμός. Το επίπεδο εισόδου χρησιμεύει μόνο στο να παρέχει το διάνυσμα εισόδου στο δίκτυο και δεν προσμετράται στα επίπεδα του, λόγω της έλλειψης υπολογιστικής λειτουργίας.



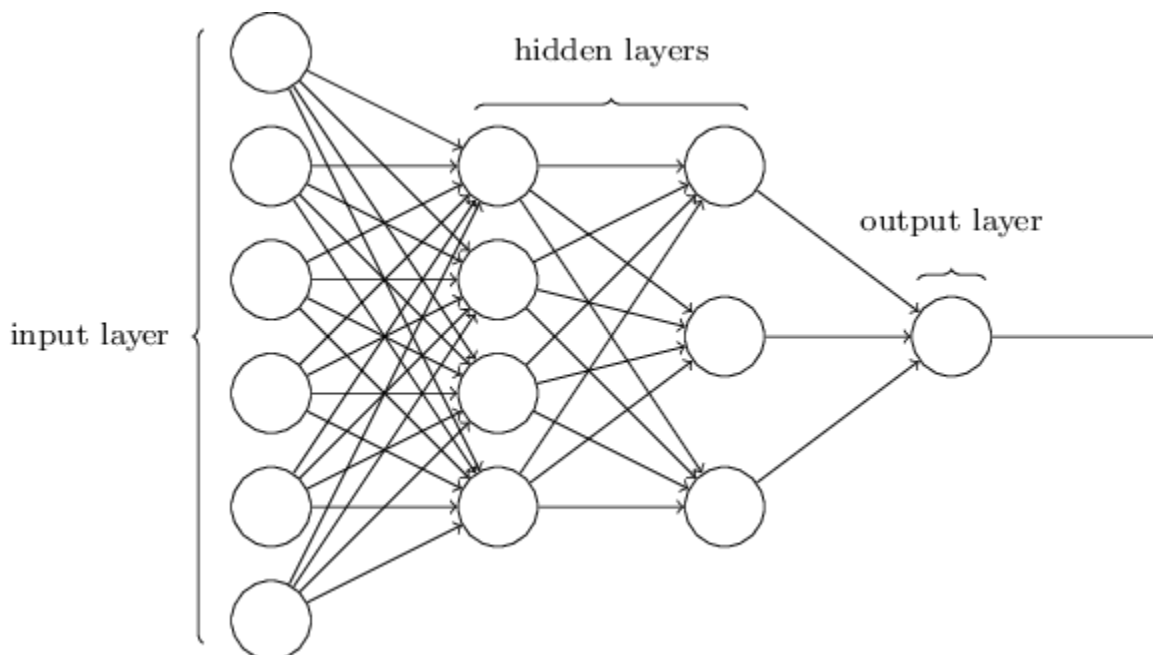
Σχήμα 2.6: Αναπαράσταση δικτύου πρόσθιας τροφοδότησης ενός επιπέδου.

2.4.2 Πολυεπίπεδα δίκτυα πρόσθιας τροφοδότησης

Ακολουθώντας τη λογική των δικτύων πρόσθιας τροφοδότησης ενός επιπέδου, παρατηρείται πάλι η παρουσία των επιπέδων νευρώνων εισόδου και εξόδου καθώς και η ίδια λογική πρόσθιας κατευθυνόμενης σύνδεσης. Η διαφορά στα πολυεπίπεδα δίκτυα είναι η παρουσία, ανάμεσα στην εξωτερικά προερχόμενη είσοδο και την έξοδο, ενός ή περισσότερων κρυφών επιπέδων (*hidden layers*), των οποίων οι υπολογιστικοί κόμβοι αποκαλούνται κρυφοί νευρώνες. Τα επίπεδα αυτά ονομάζονται κρυφά γιατί δεν είναι άμεσα ορατά από την είσοδο ή την έξοδο του δικτύου. Προσθέτοντας περισσότερα επίπεδα κρυφών νευρώνων σε ένα νευρωνικό δίκτυο, το δίκτυο αποκτά τη δυνατότητα να εξάγει περισσότερες πληροφορίες για τα δεδομένα εισόδου, μέσω των περισσότερων συνάψεων που έχει στη διάθεσή του και των μεγαλύτερης τάξης αλληλεπιδράσεων, μεταξύ των νευρώνων, που δημιουργούνται.

Η διαδικασία υπολογισμού της εξόδου του δικτύου είναι παρόμοια με αυτή της κατηγορίας των μονοεπίπεδων δικτύων που αναφέρθηκαν προηγουμένως. Οι πηγαίοι κόμβοι του επιπέδου εισόδου του δικτύου παρέχουν το διάνυσμα εισόδου του δικτύου στους κόμβους του πρώτου κρυφού επιπέδου. Αφού εκτελεστεί ο υπολογισμός σε αυτό το επίπεδο, προωθείται η έξοδος στο επόμενο κρυφό επίπεδο, το οποίο τη χρησιμοποιεί σαν είσοδο. Έτσι, η διαδικασία αυτή συνεχίζεται, με το ένα επίπεδο να χρησιμοποιεί σαν είσοδο την έξοδο του προηγούμενου, μέχρι να γίνει ο τελικός υπολογισμός στο επίπεδο εξόδου. Το σύνολο των σημάτων εξόδου των

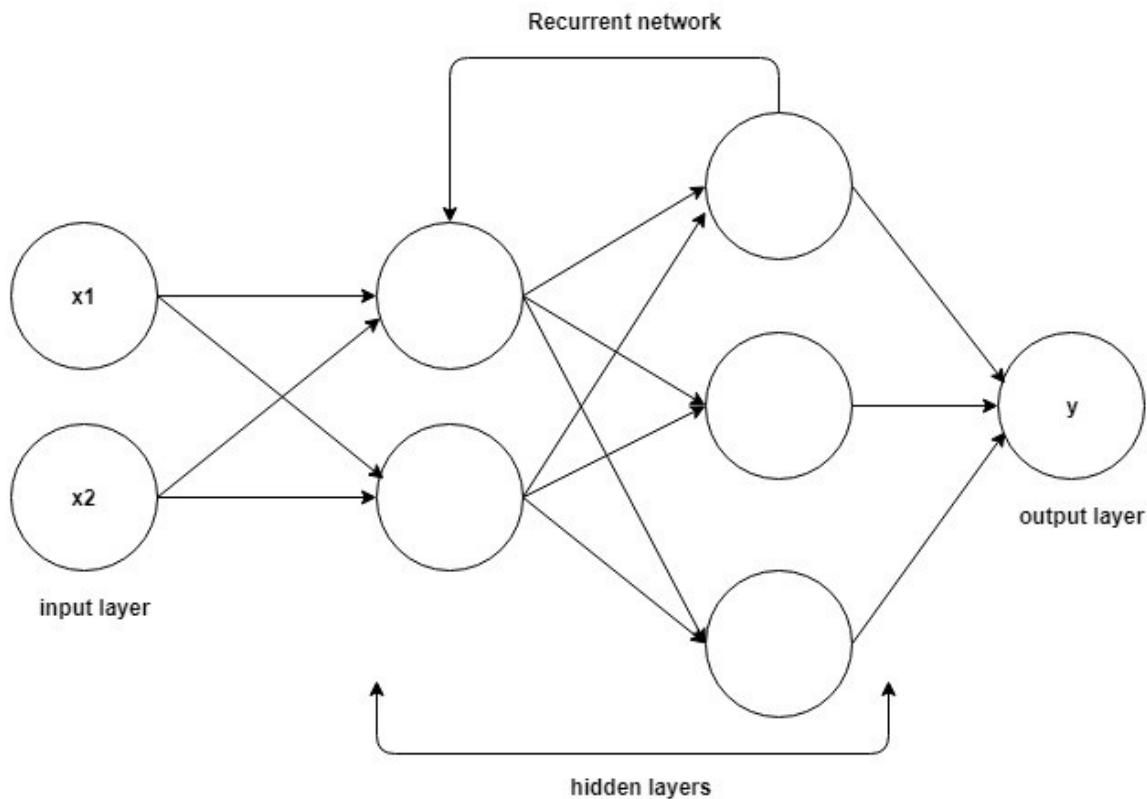
νευρώνων στο επίπεδο εξόδου του δικτύου, συνιστά τη συνολική απόκριση του δικτύου στο πρότυπο ενεργοποίησης που παρέχεται από τους κόμβους στο επίπεδο εισόδου.



Σχήμα 2.7: Αναπαράσταση δικτύου πρόσθιας τροφοδότησης τριών επιπέδων, δύο κρυφά επίπεδα και το επίπεδο εξόδου.

2.4.3 Αναδρομικά δίκτυα

Στα δίκτυα πρόσθιας τροφοδότησης ισχύει η παραδοχή ότι οι εισοδοί και οι έξοδοι των νευρώνων είναι ανεξάρτητες μεταξύ τους. Στην περίπτωση όμως των αναδρομικών δικτύων αυτό δεν ισχύει, καθώς περιέχουν τουλάχιστον έναν βρόχο ανάδρασης. Αυτό σημαίνει η έξοδος ενός νευρώνα τροφοδεύεται στην είσοδο ενός άλλου νευρώνα του ίδιου ή προηγούμενου επιπέδου. Ορίζεται επίσης η περίπτωση της αυτο-ανάδρασης όπου η έξοδος ενός νευρώνα ανατροφοδοτείται στην είσοδο του ίδιου νευρώνα. Η παρουσία βρόχων, απλής ή αυτό-ανάδρασης, έχει βαθιά επίδραση στη δυνατότητα μάθησης του δικτύου και στην απόδοσή του, καθώς προσφέρει στο δίκτυο κληρονομικότητα. Στην αναδρομική αρχιτεκτονική υπάρχει η δυνατότητα εκτέλεσης της ίδιας εργασίας, για κάθε στοιχείο μιας αλληλουχίας, με το αποτέλεσμα να εξαρτάται από τους προηγούμενους υπολογισμούς. Δημιουργείται δηλαδή, η έννοια της ύπαρξης μνήμης πληροφοριών σε αυτού του τύπου τα δίκτυα.



Σχήμα 2.8: Αναπαράσταση αναδρομικού δικτύου με έναν βρόχο ανάδρασης.

2.5 Μέθοδοι μάθησης

Ένα νευρωνικό δίκτυο προκειμένου να μπορέσει να λειτουργήσει αποτελεσματικά και να λαμβάνει σωστές αποφάσεις, πρέπει να υποβληθεί σε μια διαδικασία εκπαίδευσης. Αρχικά το δίκτυο μπορούμε να θεωρήσουμε ότι λαμβάνει τυχαίες αποφάσεις, δηλαδή δεν έχει καμία γνώση και συνεπώς κανέναν λογικό τρόπο να λάβει μια απόφαση. Με τη διαδικασία της εκπαίδευσης, αποκτά την απαραίτητη εμπειρία (μαθαίνει), η οποία αποθηκεύεται μέσω των τιμών των συναπτικών βαρών, ώστε να μπορέσει να αποφασίσει με ακρίβεια για το πρόβλημα. Με βάση τον ορισμό των Mendel και McClaren (1970):

Η μάθηση ορίζεται ως η διαδικασία με την οποία οι ελεύθερες παράμετροι ενός νευρωνικού δικτύου προσαρμόζονται μέσω μιας συνεχούς διαδικασίας διέγερσης από το περιβάλλον μέσα στο οποίο είναι ενσωματωμένο το νευρωνικό δίκτυο. Ο τύπος της μάθησης καθορίζεται από τον τρόπο με τον οποίο γίνεται η μεταβολή των παραμέτρων.

Όπως αναφέρθηκε περιληπτικά και στο κεφάλαιο 1.1 μπορούμε να κατηγοριοποιήσουμε τις διαδικασίες μάθησης σε τρεις βασικές κατηγορίες:

1. Μάθηση με επίβλεψη (Supervised learning):

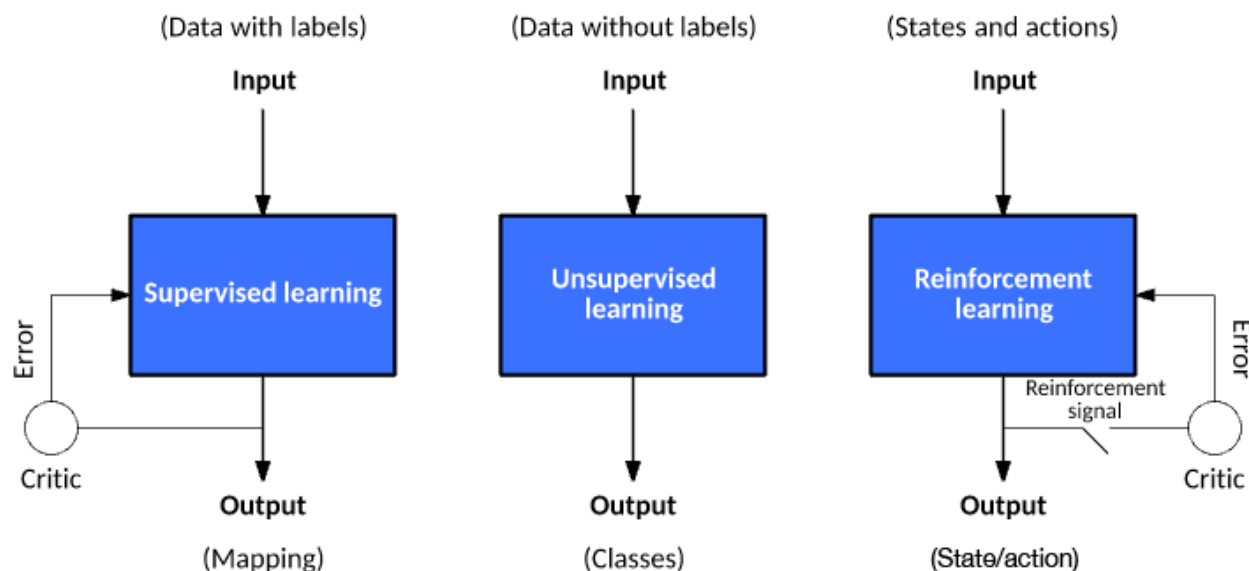
Στην εκπαίδευση με επίβλεψη ή διαφορετικά με εκπαιδευτή, το περιβάλλον είναι άγνωστο στο νευρωνικό δίκτυο. Έτσι κάθε διάνυσμα εισόδου που εισάγεται στο δίκτυο συνοδεύεται και από ένα διάνυσμα εξόδου, που είναι και η επιθυμητή απάντηση που πρέπει τελικά να δώσει το δίκτυο, για το διάνυσμα αυτό. Πιο συγκεκριμένα μπορούμε να θεωρήσουμε ένα εκπαιδευτή που κατέχει γνώση του περιβάλλοντος η οποία αναπαρίσταται από το σύνολο διανυσμάτων εισόδου-εξόδου. Ο εκπαιδευτής εφοδιάζει το δίκτυο με την επιθυμητή απόκριση για την κάθε είσοδο, η οποία αντιπροσωπεύει τη βέλτιστη ενέργεια που πρέπει να εκτελεστεί από το νευρωνικό δίκτυο. Οι παράμετροι του δικτύου προσαρμόζονται υπό τη συνδυασμένη επιρροή του διανύσματος εκπαίδευσης και του σήματος σφάλματος. Το τελευταίο ορίζεται ως η διαφορά μεταξύ της επιθυμητής απόκρισης και της πραγματικής απόκρισης του δικτύου. Έτσι, σταδιακά και με επαναληπτικό τρόπο το δίκτυο προσαρμόζει τις παραμέτρους του, με σκοπό να καταφέρει να προσομοιώσει τη συμπεριφορά του εκπαιδευτή, δηλαδή να δίνει τις σωστές απαντήσεις για κάθε είσοδο. Η διαδικασία αυτή επαναλαμβάνεται μέχρις ότου, το σφάλμα, ή αλλιώς κόστος, να μηδενιστεί ή να θεωρηθεί αποδεκτό. Υπό αυτή την έννοια λοιπόν, ο εκπαιδευτής μεταφέρει τη γνώση του περιβάλλοντος που διαθέτει στο νευρωνικό δίκτυο, μέσω της εκπαίδευσης και αυτή αποθηκεύεται στο δίκτυο με τη μορφή των συναπτικών βαρών. Στη συνέχεια, θα επικεντρωθούμε στα θεωρητικά σημεία που αφορούν τον παρόντα τύπο μάθησης, καθώς αυτός χρησιμοποιήθηκε και στο πρακτικό κομμάτι της διπλωματικής εργασίας.

2. Μάθηση χωρίς επίβλεψη (Unsupervised learning):

Στη μη επιβλεπόμενη μάθηση, αναφερόμενη και ως αυτο-οργανούμενη, δεν υπάρχει εξωτερικός εκπαιδευτής που επιβλέπει τη διαδικασία της μάθησης. Ο στόχος εδώ είναι να βελτιστοποιηθούν οι ελεύθερες παράμετροι του δικτύου με βάση μια αντικειμενική συνάρτηση, ανεξάρτητη της εργασίας που θα εκτελέσει το δίκτυο, η οποία και αποτελεί μέτρο της ποιότητας της απεικόνισης της εξωτερικής πληροφορίας που καλείται να αφομοιώσει το δίκτυο. Επομένως μέσω αυτής της διαδικασίας το δίκτυο, αφού πρώτα συντονιστεί στις στατιστικές κανονικότητες των δεδομένων εισόδου, αποκτά τη δυνατότητα να σχηματίζει εσωτερικές αναπαραστάσεις για την κωδικοποίηση διαφόρων χαρακτηριστικών της εισόδου. Συχνά χρησιμοποιείται σε εφαρμογές που αναζητούνται ομαδοποιήσεις των δεδομένων σε συστάδες (*clusters*), με βάση τα πιθανά κοινά χαρακτηριστικά που μπορεί να εμφανίζουν αυτά τα δεδομένα.

3. Ενισχυτική μάθηση (Reinforcement learning):

Σε αυτό τον τύπο μάθησης η παρατηρείται σημαντική διαφοροποίηση από τις τεχνικές εκπαίδευσης που προαναφέρθηκαν. Σε σύγκριση με την επιβλεπόμενη μάθηση, η αντιστοίχιση μιας εισόδου με την κατάλληλη έξοδο δε γίνεται μέσω κάποιου εκπαιδευτή, δηλαδή δεν παρέχεται στο δίκτυο η γνώση του περιβάλλοντος. Αντιθέτως, η εκμάθηση εκτελείται μέσω συνεχούς αλληλεπίδρασης με το περιβάλλον, με στόχο την ελαχιστοποίηση ενός βαθμωτού δείκτη απόδοσης. Ο σκοπός της ενισχυτικής μάθησης είναι η ελαχιστοποίηση μια συνάρτησης τρέχοντος κόστους, η οποία ορίζεται ως πρόβλεψη του συνολικού ή αθροιστικού κόστους ενεργειών που εκτελούνται σε αλληλουχία βημάτων και όχι του άμεσου κόστους μιας μόνο ενέργειας. Η λειτουργία του συστήματος είναι να ανακαλύψει ποιες ενέργειες συνεισφέρουν περισσότερο στην καλύτερη συμπεριφορά του συστήματος και να τις τροφοδοτήσει πίσω στο περιβάλλον. Η συγκεκριμένη τεχνική εκπαίδευσης παρουσιάζει ιδιαίτερο ενδιαφέρον καθώς παρέχει στο σύστημα μάθησης μια βάση επικοινωνίας με το περιβάλλον του. Έτσι, το σύστημα δύναται να μάθει να εκτελεί με ακρίβεια και αποτελεσματικότητα μια ενέργεια, βασιζόμενο αποκλειστικά στην εμπειρία που αποκτά μέσω της αλληλεπίδρασης με το περιβάλλον του.



Σχήμα 2.9: Σχηματικό διάγραμμα των τριών βασικών μεθόδων μάθησης.

Πηγή: IBM

2.6 Συνάρτηση κόστους (*Loss function*)

Οι συναρτήσεις κόστους εκφράζουν μια μέτρηση της αποδοτικότητας του νευρωνικού δικτύου, υπολογίζοντας το σφάλμα μεταξύ των προβλέψεων του δικτύου και των επιθυμητών εξόδων. Στην ουσία ποσοτικοποιούν την αποδοτικότητα της εκπαίδευσης του δικτύου. Ο μέσος όρος των σφαλμάτων σε ολόκληρο το σύνολο δεδομένων εκπαίδευσης εκφράζει τη διαφορά ακρίβειας του δεδομένου μοντέλου με το ιδανικό μοντέλο, δηλαδή αυτό που έχει απόλυτη ακρίβεια προβλέψεων. Η διαδικασία προσέγγισης αυτής της ιδανικής κατάστασης, ισοδυναμεί με την εύρεση των ελεύθερων παραμέτρων δικτύου που θα ελαχιστοποιούν τη συνάρτηση κόστους. Με βάση αυτή τη λογική, οι συναρτήσεις κόστους δίνουν τη δυνατότητα αναγωγής της αποδοτικής εκπαίδευσης του νευρωνικού δικτύου, σε ένα πρόβλημα βελτιστοποίησης και συγκεκριμένα ελαχιστοποίησης της συνάρτησης κόστους.

2.6.1 Συνάρτηση μέσου τετραγωνικού σφάλματος

(*Mean Squared Error - MSE*)

Πρόκειται για μια από τις συναρτήσεις που χρησιμοποιούνται ευρέως σε εφαρμογές παλινδρόμησης (*regression*) όπου η έξοδος του δικτύου είναι η πρόβλεψη της τιμής μιας συνεχής μεταβλητής. Η συνάρτηση μέσου τετραγωνικού σφάλματος υπολογίζει το τετράγωνο της διαφοράς μεταξύ των προβλέψεων του δικτύου και των επιθυμητών αποτελεσμάτων. Παράτιθεται η μαθηματική έκφραση:

$$\text{MSE} = \frac{1}{n} \sum_{i=1}^n (y_i - \tilde{y}_i)^2$$

Όπου:

y_i είναι η επιθυμητή έξοδος για το i -οστό δείγμα δεδομένων

\tilde{y}_i είναι η πρόβλεψη του δικτύου για το i -οστό δείγμα δεδομένων

n είναι το πλήθος των δειγμάτων δεδομένων

2.6.2 Δυαδική συνάρτηση εντροπίας (*Binary cross entropy*)

Χρησιμοποιείται σε προβλήματα δυαδικής κατηγοριοποίησης (*binary classification*) όπου το δίκτυο καλείται να αποφανθεί ανάμεσα σε δύο διακριτές κλάσεις, συμβολιζόμενες ως κλάση μηδέν και κλάση ένα. Σε αυτές τις εφαρμογές το δίκτυο επιστρέφει μια τιμή στο στρώμα εξόδου, η οποία κυμαίνεται από το μηδέν έως το ένα. Αυτή η τιμή περιγράφει την πιθανότητα η πρόβλεψη να αντιστοιχεί σε μία εκ των δύο κλάσεων, την κλάση ένα. Όπως φαίνεται και παρακάτω αν η πρόβλεψη του δικτύου είναι κοντά στο ένα, τότε επικρατεί ο πρώτος όρος του αθροίσματος, ενώ αν η πρόβλεψη είναι κοντά στο μηδέν επικρατεί ο δεύτερος. Παρατίθεται ο μαθηματικός τύπος:

$$H_p(q) = -\frac{1}{N} \sum_{i=1}^N y_i \cdot \log(p(y_i)) + (1 - y_i) \cdot \log(1 - p(y_i))$$

Όπου:

- y_i είναι η επιθυμητή έξοδος για το i -οστό δείγμα δεδομένων
- $p(y_i)$ είναι η πρόβλεψη της πιθανότητας το i -οστό δείγμα δεδομένων να ανήκει στην κλάση ένα
- N είναι το πλήθος των δειγμάτων δεδομένων

2.6.3 Κατηγορική συνάρτηση εντροπίας (*Categorical cross entropy*)

Η συγκεκριμένη συνάρτηση βρίσκει χρήση σε προβλήματα κατηγοριοποίησης, όπου κάθε δείγμα δεδομένων αντιστοιχεί μόνο σε μία κλάση. Συνηθίζεται ο αριθμός των κλάσεων να είναι μεγαλύτερος του δύο, αλλιώς πρόκειται για δυαδική κατηγοριοποίηση, όπως προαναφέρθηκε (κεφάλαιο 2.6.2). Η κατηγορική συνάρτηση εντροπίας συγκρίνει την κατανομή των προβλέψεων του δικτύου με την πραγματική κατανομή των δεδομένων, με την πιθανότητα της σωστής κλάσης να είναι ίση με ένα, ενώ για τις υπόλοιπες κλάσεις μηδέν. Παρατίθεται η μαθηματική έκφραση:

$$L(y, \hat{y}) = - \sum_{j=0}^M \sum_{i=0}^N (y_{ij} * \log(\hat{y}_{ij}))$$

Όπου:

- y_{ij} είναι η επιθυμητή έξοδος για το i -οστό δείγμα δεδομένων
 \tilde{y}_{ij} είναι η πρόβλεψη του δικτύου για το i -οστό δείγμα δεδομένων
 N είναι το πλήθος των δειγμάτων δεδομένων
 M είναι το πλήθος των κλάσεων

2.7 Αλγόριθμοι βελτιστοποίησης

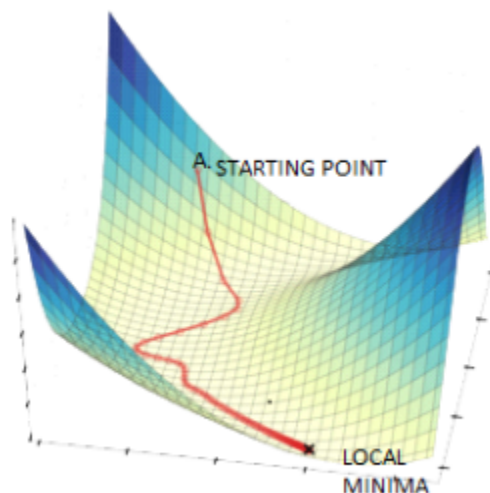
Όπως αναφέρθηκε προηγουμένως η διαδικασία της μάθησης ανάγεται σε ένα πρόβλημα βελτιστοποίησης μιας συνάρτησης κόστους. Κάθε σύνολο ελεύθερων παραμέτρων του νευρωνικού δικτύου αντιπροσωπεύει μια πρόβλεψη του δικτύου για τα δεδομένα εισόδου. Επομένως η βελτιστοποίηση της συνάρτησης κόστους ταυτίζεται με την κατάλληλη επιλογή των βαρών του δικτύου. Μπορούμε να θεωρήσουμε όλους τους δυνατούς συνδυασμούς των βαρών του δικτύου σαν έναν χώρο παραμέτρων, στον οποίο αναζητούμε τον συνδυασμό που δίνει τη δυνατότητα στις προβλέψεις του δικτύου να προσεγγίζει τα πραγματικά δεδομένα. Για την εύρεση αυτού του συνδυασμού ελεύθερων παραμέτρων χρησιμοποιούνται οι αλγόριθμοι βελτιστοποίησης. Ο λόγος που η χρήση τέτοιων αλγορίθμων κρίνεται απαραίτητη είναι ότι οι μεταβλητές των συναρτήσεων κόστους είναι πολλές και δεν είναι εύκολα υπολογίσιμο το ολικό τους ελάχιστο με αναλυτικό μαθηματικό τρόπο.

Οι αλγόριθμοι κόστους είναι κατά το πλείστον επαναληπτικές διαδικασίες. Σε κάθε επανάληψη καθορίζονται και έπειτα εφαρμόζονται οι αλλαγές των μεταβλητών της συνάρτησης κόστους, δηλαδή οι ελεύθερες παράμετροι του νευρωνικού δικτύου (βάρη και πολώσεις). Έτσι, στην επόμενη επανάληψη υπολογίζεται εκ νέου η συνάρτηση κόστους και επαναλαμβάνεται η ίδια διαδικασία μέχρι να βρεθεί το επιθυμητό ακρότατο της συνάρτησης.

Παρακάτω θα αναφερθούν δύο βασικοί και ευρέως χρησιμοποιούμενοι αλγόριθμοι βελτιστοποίησης.

2.7.1 Αλγόριθμος κλίσης κατάβασης (*Gradient Descent - GD*)

Ο αλγόριθμος αυτός εκτελεί μια επαναληπτική μέθοδο με στόχο να εντοπίσει ένα τοπικό (ή ιδανικά ολικό) ελάχιστο μιας παραγωγίσιμης συνάρτησης, που στην περίπτωση μας πρόκειται για τη συνάρτηση κόστους του δικτύου. Μπορούμε να σκεφτούμε κάθε συνδυασμό των παραμέτρων της συνάρτησης κόστους σαν ένα σημείο σε μια πολυδιάστατη επιφάνεια, όπου ο αριθμός των διαστάσεων αντιστοιχεί στο πλήθος των παραμέτρων. Επομένως, εφόσον αλλάζουμε τις μεταβλητές της συνάρτησης κόστους σε κάθε επανάληψη, στην ουσία μετακινούμαστε από ένα σημείο σε ένα άλλο, πάνω σε αυτή την επιφάνεια. Συνοπτικά, η ιδέα είναι ο υπολογισμός της κλίσης της συνάρτησης κόστους στο παρόν σημείο της επανάληψης και η επιλογή του επόμενου σημείου προς την αντίθετη κατεύθυνση του διανύσματος κλίσης. Με άλλα λόγια η επιλογή των βαρών γίνεται προς την κατεύθυνση της πλέον απότομης κατάβασης (*steepest descent*).



Σχήμα 2.10: Η μέθοδος κλίσης κατάβασης για μια συνάρτηση κόστους δύο μεταβλητών.

Συνεπώς ο αλγόριθμος κλίσης κατάβασης περιγράφεται φορμαλιστικά από τον παρακάτω τύπο:

$$w_{t+1} = w_t - \eta \cdot g_t$$

Όπου:

w_t το διάνυσμα των βαρών στην επανάληψη t

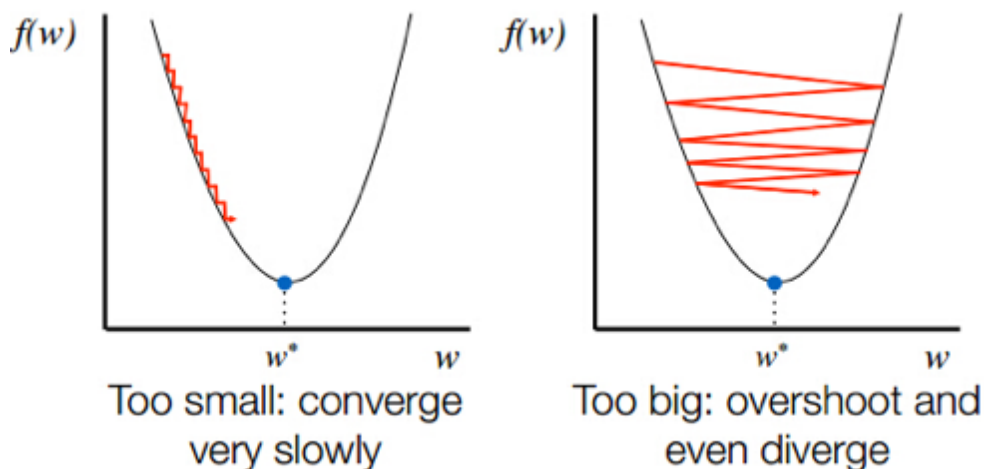
$g_t = \nabla\Phi(w)$ η κλίση της συνάρτησης κόστους $\Phi(w)$ στην επανάληψη t

η θετική σταθερά που αποκαλείται παράμετρος ρυθμού μάθησης

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

$$\Delta w_t = w_{t+1} - w_t = -\eta \cdot g_t$$

Είναι σημαντικό να σημειωθεί ότι η παράμετρος ρυθμού μάθησης έχει βαθιά επίδραση στη συμπεριφορά σύγκλισης του αλγορίθμου στη βέλτιστη αναζητούμενη λύση. Όταν η παράμετρος είναι πολύ μικρή η τροχιά του διανύσματος βαρών ακολουθεί ομαλή διαδρομή, ο αλγόριθμος αργεί να συγκλίνει και το δίκτυο χρειάζεται πολλές επαναλήψεις και χρόνο για να μάθει. Όταν αντιθέτως η παράμετρος παίρνει μεγάλες τιμές, τα βήματα που γίνονται είναι τόσο μεγάλα που παρατηρείται ταλαντωνόμενη διαδρομή.



Σχήμα 2.11: Παράδειγμα της επίδρασης του ρυθμού μάθησης για μια συνάρτηση κόστους μιας μεταβλητής $f(w)$. Αριστερά έχουμε μικρή τιμή της σταθεράς, ενώ δεξιά μεγάλη. Η βέλτιστη λύση συμβολίζεται ως w^* .

Η συνάρτηση κόστους χρησιμοποιεί στον υπολογισμό της όλα τα δείγματα εισόδου. Αυτό επιβαρύνει χρονικά τη διαδικασία μάθησης, καθώς συνήθως το πλήθος των δεδομένων που τροφοδοτούνται σε ένα νευρωνικό δίκτυο είναι συχνά πολύ μεγάλο. Μία μέθοδος, ή οποία αποτελεί υποπερίπτωση του αλγορίθμου κλίσης κατάβασης είναι η μέθοδος της στοχαστικής κλίσης κατάβασης (*Stochastic Gradient Descent - SGD*). Η μέθοδος αυτή χρησιμοποιεί ένα υποσύνολο του συνόλου δεδομένων εκπαίδευσης, προκειμένου να υπολογίσει την κλίση της συνάρτησης κόστους. Τα δεδομένα αυτά χωρίζονται σε παρτίδες (*batch*), οι οποίες χρησιμοποιούνται ξεχωριστά για τον υπολογισμό της κλίσης και την ενημέρωση των βαρών. Ονομάζεται στοχαστικός αλγόριθμος διότι η κλίση που υπολογίζεται για κάθε

παρτίδα δεδομένων αποτελεί στοχαστική προσέγγιση της πραγματικής. Ο στοχαστικός αυτός χαρακτήρας της μεθόδου, προσφέρει το πλεονέκτημα ότι μειώνει την πιθανότητα η διαδικασία μάθησης να παγιδευτεί σε τοπικό ελάχιστο.

2.7.2 Αλγόριθμος εκτίμησης προσαρμοστικής ροπής

(*Adaptive moment estimation - Adam*)

Παρουσιάστηκε από τους Diederik P. Kingma και Jimmy Lei Ba το 2015 και συγκαταλέγεται πλέον στις πιο αποδοτικές επιλογές αλγορίθμων βελτιστοποίησης. Αυτό εξηγείται από την ικανότητα του αλγορίθμου να λύνει γρήγορα και με ακρίβεια προβλήματα βαθιάς μάθησης. Η μεγαλύτερη διαφορά του με τον αλγόριθμο στοχαστικής κλήσης κατάβασης (*SGD*) που προαναφέρθηκε, είναι η μη σταθερή τιμή της παραμέτρου ρυθμού μάθησης. Ο αλγόριθμος Adam διατηρεί διαφορετικές παραμέτρους ρυθμού μάθησης για κάθε βάρος του δικτύου και τις προσαρμόζει ανεξάρτητα κατά τη διαδικασία της μάθησης.

Η λειτουργία του βασίζεται στον υπολογισμό δύο μεγεθών, της πρώτης και της δεύτερης ροπής της συνάρτησης κλίσης. Στα μαθηματικά και τη στατιστική οι ροπές είναι μεγέθη του σχήματος μια συνάρτησης, όπου η πρώτη ροπή αναφέρεται στη μέση τιμή και η δεύτερη ροπή στη διακύμανση. Συγκεκριμένα ο αλγόριθμος Adam υπολογίζει τους εκθετικούς μέσους όρους των κλίσεων και του τετραγώνου των κλίσεων, οι οποίοι φθίνουν με βάση τις παραμέτρους β_1 και β_2 . Οι ροπές της συνάρτησης κλίσης υπολογίζονται ως εξής:

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

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

Όπου:

m_t η πρώτη ροπή της κλίσης στην επανάληψη t

v_t η δεύτερη ροπή της κλίσης στην επανάληψη t

$g_t = \nabla\Phi(w)$ η κλίση της συνάρτησης κόστους $\Phi(w)$ στην επανάληψη t

β_1, β_2 εκθετικά ποσοστά μείωσης (δηλαδή σταθερές από 0 έως 1)

Από τις παραπάνω σχέσεις φαίνεται η αναδρομική φύση των όρων m_t και v_t και συνεπώς είναι προφανές ότι απαιτείται η αρχικοποίηση αυτών τιμών. Δεδομένου ότι οι ροπές m_t και v_t αρχικοποιούνται με μηδενική τιμή, οι δημιουργοί του αλγορίθμου παρατήρησαν ότι είναι προκατειλημμένα (*biased*) προς το μηδέν κατά τη διάρκεια των αρχικών επαναλήψεων, πράγμα που συμβαίνει ακόμα πιο έντονα όταν τα

ποσοστά μείωσης β_1 και β_2 έχουν τιμές κοντά στο ένα. Για να αντιμετωπιστεί αυτό το πρόβλημα, αφού προσδιοριστούν οι (*biased*) ροπές με τους μαθηματικούς τύπους που προαναφέρθηκαν, υπολογίζονται νέες τιμές για τις ροπές που διορθώνουν αυτή την προκατάληψη (*bias-corrected*).

$$\widehat{m}_t = \frac{m_t}{1-\beta_1}$$

$$\widehat{v}_t = \frac{v_t}{1-\beta_2}$$

Όπου:

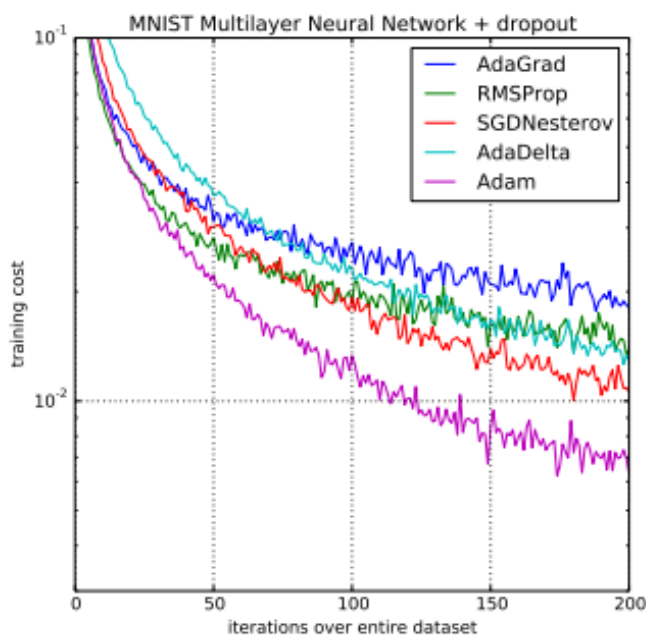
\widehat{m}_t η *bias-corrected* τιμή της πρώτης ροπής της κλίσης στην επανάληψη t

\widehat{v}_t η *bias-corrected* τιμή της δεύτερης ροπής της κλίσης στην επανάληψη t

Έτσι, με χρήση αυτών των νέων τιμών προκύπτει ο παρακάτω κανόνας ενημέρωσης του αλγορίθμου:

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{\widehat{v}_t + \epsilon}} \cdot \widehat{m}_t$$

Όπου: ϵ μια πολύ μικρή σταθερά για την αποφυγή διαίρεσης με το μηδέν



Σχήμα 2.12: Συγκριτικό γράφημα μεταξύ διαφόρων αλγορίθμων βελτιστοποίησης.

2.8 Αλγόριθμος Οπισθοδιάδοσης (*Backpropagation - BK*)

Ο αλγόριθμος οπισθοδιάδοσης είναι ένας ευρέως χρησιμοποιούμενος αλγόριθμος για την εκπαίδευση νευρικών δικτύων πρόσθιας τροφοδότησης. Κατά την εκπαίδευση ενός νευρωνικού δικτύου, ο αλγόριθμος αυτός υπολογίζει την κλίση της συνάρτησης κόστους σε σχέση με κάθε βάρος χρησιμοποιώντας τον κανόνα αλυσίδας. Η διαδικασία αυτή εκτελείται ξεχωριστά για κάθε στρώμα του δικτύου ξεκινώντας από το τελευταίο στρώμα με κατεύθυνση προς τα πίσω, δηλαδή προς το στρώμα εισόδου, με σκοπό να αποφευχθούν περιττοί υπολογισμοί των ενδιάμεσων όρων στον κανόνα της αλυσίδας. Αυτός ο αποτελεσματικός υπολογισμός προσφέρει τη δυνατότητα χρήσης αλγορίθμων βελτιστοποίησης που χρησιμοποιούν την κλίση της συνάρτησης κόστους, όπως οι δύο που αναφέρθηκαν παραπάνω (κεφάλαια 2.7.1 και 2.7.2).

Επειδή ο αλγόριθμος οπισθοδιάδοσης εξετάζει ξεχωριστά πως επιδρά το κάθε βάρος στο εξαγόμενο σφάλμα, αποτελεί προσέγγιση της διαίρεσης της συμβολής του κάθε βάρους στην ακρίβεια πρόβλεψης του δικτύου. Το κλειδί είναι να βρεθούν τα βάρη τα οποία είναι περισσότερο υπεύθυνα για το σφάλμα και στη συνέχεια να δοθεί μεγαλύτερη βαρύτητα στην προσαρμογή των τιμών τους.

Έστω δείγμα εκπαίδευσης $\{(x(n), d(n))\}$ με $n=1$ έως N , όπου $x(n)$ το διάνυσμα εισόδων και $d(n)$ οι αντίστοιχες ετικέτες, δηλαδή οι επιθυμητές αποκρίσεις. Δεδομένου ότι οι προσαρμογές των συναπτικών βαρών εκτελούνται σε βάση παράδειγμα προς παράδειγμα, το n αντιστοιχεί και στο μετρητή των επαναλήψεων του αλγορίθμου. Τα βήματα εκτέλεσης είναι τα εξής:

1. Αρχικοποίηση.

Υποθέτοντας ότι δεν παρέχεται πρότερη πληροφορία, επιλέγονται τα συναπτικά βάρη και τα κατώφλια από μια ομοιόμορφη κατανομή, της οποίας ο μέσος είναι μηδέν. Η διακύμανση επιλέγεται έτσι ώστε η τυπική απόκλιση των τοπικών πεδίων των νευρώνων να βρίσκεται στο όριο μεταξύ του γραμμικού και του στάνταρ μέρους της σιγμοειδούς συνάρτησης ενεργοποίησης.

2. Παρουσιάσεις των παραδειγμάτων εκπαίδευσης.

Τροφοδότηση μιας εποχής (*epoch*) παραδειγμάτων εκπαίδευσης στο δίκτυο. Έπειτα για κάθε παράδειγμα του δείγματος δεδομένων εκτελείται η ακολουθία των προς τα εμπρός και προς τα πίσω υπολογισμών που περιγράφεται στα βήματα 3 και 4, αντίστοιχα.

Σημείωση 2.1: Ο όρος εποχή ορίζεται ως η προσπέλαση, από το δίκτυο, ολόκληρου του συνόλου δεδομένων εκπαίδευσης μια φορά (προς τα εμπρός και προς τα πίσω).

3. Υπολογισμός προς τα εμπρός.

Έστω ένα παράδειγμα εισόδου $x(n)$ που εφαρμόζεται στο επίπεδο εισόδου και ένα δάνυσμα επιθυμητών αποκρίσεων $d(n)$ που παρουσιάζεται στο επίπεδο εξόδου. Υπολογίζονται τα τοπικά πεδία και τα λειτουργικά σήματα του δικτύου προχωρώντας προς τα εμπρός, επίπεδο προς επίπεδο, με τον εξής τρόπο:

$$v_j^{(l)} = \sum_i w_{ji}^{(l)}(n) \cdot y_i^{(l-1)}(n)$$

Όπου:

$v_j^{(l)}$ το τοπικό πεδίο για το νευρώνα j στο επίπεδο l

$y_i^{(l-1)}$ το σήμα εξόδου (λειτουργικό) του νευρώνα i του προηγούμενου επιπέδου $l - 1$

$w_{ji}^{(l)}$ το συναπτικό βάρος του νευρώνα j του επιπέδου l που τροφοδοτείται από τον νευρώνα i στο επίπεδο $l - 1$

Για $i=0$ έχουμε:

$$y_0^{(l-1)}(n) = 1 \quad \text{και} \quad w_{j0}^{(l)}(n) = b_j^{(l)}(n)$$

Όπου:

$b_j^{(l)}$ η πόλωση που εφαρμόζεται στο νευρώνα j στο επίπεδο l

Για το σήμα εξόδου ενός νευρώνα έχουμε τρεις περιπτώσεις. Η πρώτη αφορά τους νευρώνες στο στρώμα εισόδου, η δεύτερη τους νευρώνες των κρυφών στρωμάτων και η τρίτη τους νευρώνες του στρώματος εξόδου. Αυτές οι περιπτώσεις περιγράφονται μαθηματικά ως εξής:

$$y_j^{(l)} \begin{cases} x_j(n) & \text{για } l = 1 \\ \varphi_j(v_j(n)) & \text{για } 1 < l < L \\ o_j(n) & \text{για } l = L \end{cases}$$

Όπου:

$y_j^{(l)}$	το σήμα εξόδου του νευρώνα j στο επίπεδο l
$x_j^{(l)}$	το j -οστό στοιχείο του διανύσματος εισόδου
φ_j	η συνάρτηση ενεργοποίησης του νευρώνα j
o_j	ο j -οστό στοιχείο του διανύσματος εξόδου του δικτύου
L	το βάθος του δικτύου (σε στρώματα)

Έπειτα υπολογίζεται το σήμα σφάλματος ως εξής:

$$e_j(n) = d_j(n) - o_j(n)$$

Όπου:

d_j το j -οστό στοιχείο του διανύσματος επιθυμητών αποκρίσεων

4. Υπολογισμός προς τα πίσω.

Υπολογίζονται οι τοπικές κλήσεις του δικτύου με τη χρήση του παρακάτω μαθηματικού τύπου:

$$\delta_j^{(l)} = \begin{cases} e_j^{(L)}(n) \cdot \varphi_j'(v_j^{(L)}(n)) & \text{νευρώνας } j \text{ στο επίπεδο εξόδου } L \\ \varphi_j'(v_j^{(l)}(n)) \cdot \sum_k \delta_k^{(l+1)}(n) \cdot w_{kj}^{(l+1)}(n) & \text{νευρώνας } j \text{ στο κρυφό επίπεδο } l \end{cases}$$

Στη συνέχεια προσαρμόζονται τα συναπτικά βάρη του δικτύου στο επίπεδο l σύμφωνα με το γενικευμένο κανόνα Δέλτα:

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + a \cdot [w_{ji}^{(l)}(n-1)] + \eta \cdot \delta_j^{(l)}(n) \cdot y_i^{(l-1)}(n)$$

Όπου η είναι η παράμετρος μάθησης και a η σταθερά ορμής.

5. Επανάληψη.

Επαναλαμβάνονται οι υπολογισμοί που περιγράφονται στα βήματα 3 και 4, παρουσιάζοντας νέες εποχές παραδειγμάτων εκπαίδευσης στο δίκτυο, μέχρι να ικανοποιηθεί το κριτήριο τερματισμού.

Κεφάλαιο 3

Συνελικτικά Νευρωνικά Δίκτυα

3.1 Ορισμός

Τα συνελικτικά δίκτυα αποτελούν υποκατηγορία των πολυεπίπεδων δικτύων πρόσθιας τροφοδότησης (κεφάλαιο 2.4.2) αλλά διαφέρουν από τις αρχιτεκτονικές των παραδοσιακών πλήρως συνδεδεμένων νευρωνικών δικτύων. Η βασική ιδιαιτερότητα που παρουσιάζουν είναι η καταλληλότητά τους στην υποδοχή εικόνων ως μεταβλητή εισόδου. Τα συνηθισμένα πλήρως συνδεδεμένα δίκτυα αποσυνθέτουν την τρισδιάστατη μορφή των εικόνων αυτών και παράγουν ένα διάνυσμα μιας διάστασης, παραθέτοντας γραμμικά τα στοιχεία από όλες τις διαστάσεις. Αυτό φυσικά δεν είναι υπολογιστικά αποδοτικό καθώς για μια εικόνα διάστασης 320×240 pixels χρειαζόμαστε $320 \times 240 = 76800$ νευρώνες εισόδου και με την εκμάθηση χαρακτηριστικών για ένα τέτοιο διάνυσμα, η πολυπλοκότητα αυξάνεται σημαντικά. Η μεγάλη εξέλιξη των συνελικτικών δικτύων είναι ότι σαν είσοδο δέχονται ένα πολυδιάστατο πίνακα, πράγμα που επιτρέπει την διατήρηση της τρισδιάστατης μορφής των έγχρωμων φωτογραφιών και συνεπώς την αποδοτικότερη ανάλυσή τους. Συνοπτικά, τα συνελικτικά δίκτυα εκμεταλλεύονται την ιεραρχική δομή των δεδομένων μοντελοποιώντας μικρά τμήματα πληροφορίας, τα οποία στη συνέχεια ενώνονται για να δημιουργήσουν υψηλότερου επιπέδου πληροφορία. Η μελέτη της αρχιτεκτονικής και λειτουργίας των συνελικτικών δικτύων θα διεξαχθεί στα επόμενα υποκεφάλαια, αφού πρώτα όμως δοθεί ένας τυπικός ορισμός τους.

Ένα συνελικτικό δίκτυο (*Convolutional Neural Network - CNN*) 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 περίπλοκη εργασία διδάσκεται με διαδικασία επιβλεπόμενης μάθησης (κεφάλαιο 2.5), μέσω ενός δικτύου του οποίου η δομή περιλαμβάνει τις ακόλουθες μορφές περιορισμών:

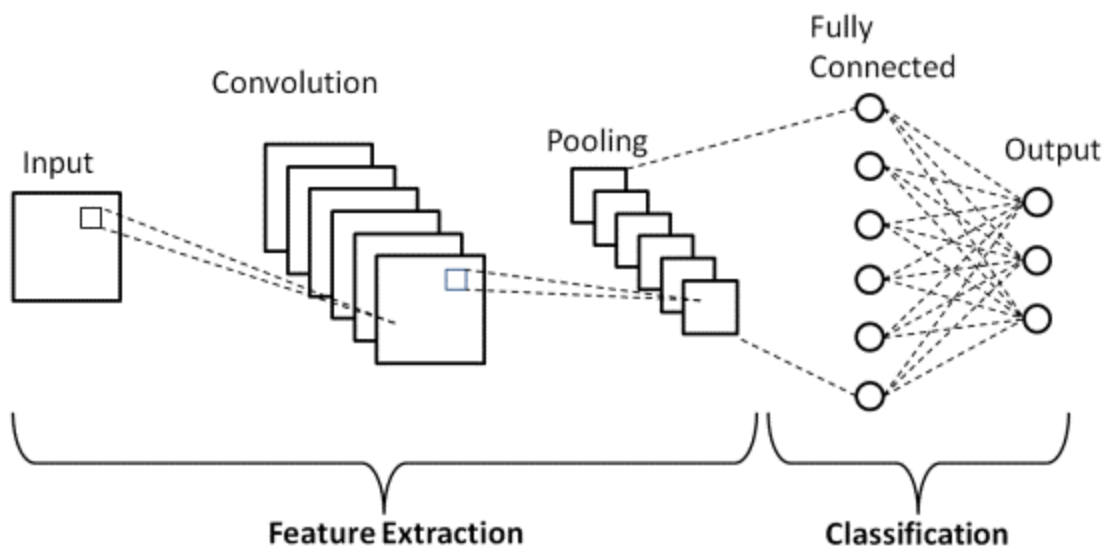
- 1. Εξαγωγή χαρακτηριστικών.** Κάθε νευρώνας λαμβάνει τις συναπτικές εισόδους του από ένα τοπικό δεκτικό πεδίο του προηγούμενου επιπέδου, υποχρεώνοντάς το να εξάγει τοπικά χαρακτηριστικά. Αφού εξαχθεί ένα χαρακτηριστικό, η ακριβής θέση του γίνεται λιγότερο σημαντική, εφόσον διατηρείται προσεγγιστικά η σχετική του θέση ως προς άλλα χαρακτηριστικά.

- 2. Αντιστοίχιση χαρακτηριστικών.** Κάθε υπολογιστικό επίπεδο του δικτύου αποτελείται από πολλαπλούς χάρτες χαρακτηριστικών (*features map*), με κάθε έναν από αυτούς να παίρνει τη μορφή ενός επιπέδου μέσα στο οποίο οι μεμονωμένοι νευρώνες μοιράζονται το ίδιο σύνολο συναπτικών βαρών. Με αυτό τον τρόπο επιτυγχάνεται η μη ευαισθησία του δικτύου ως προς τη μετατόπιση, η οποία επιβάλλεται στη λειτουργία ενός χάρτη χαρακτηριστικών μέσω της χρήσης της συνέλιξης με έναν πυρήνα (*kernel*) μικρού μεγέθους. Επιπροσθέτως μειώνεται ο αριθμός των ελεύθερων παραμέτρων, η οποία γίνεται δυνατή μέσω του διαμοιρασμού βαρών.
- 3. Υποδειγματοληψία.** Κάθε συνελικτικό επίπεδο ακολουθείται από ένα υπολογιστικό επίπεδο το οποίο εκτελεί υποδειγματοληψία με αποτέλεσμα τη μείωση της ανάλυσης του χάρτη χαρακτηριστικών. Αυτή η λειτουργία έχει ως αποτέλεσμα τη μείωση της ευαισθησίας της εξόδου του χάρτη χαρακτηριστικών στις μετατοπίσεις και άλλες μορφές παραμόρφωσης.

3.2 Επίπεδα επεξεργασίας

Στο *σχήμα 3.1*, που ακολουθεί, φαίνεται η γενική αρχιτεκτονική ενός συνελικτικού νευρωνικού δικτύου καθώς και τα δομικά επίπεδα που την απαρτίζουν. Το πρώτο επίπεδο είναι φυσικά αυτό της εισόδου (*input*), μέσω του οποίου τροφοδοτούνται τα δεδομένα στο δίκτυο. Ακολουθούν τα επίπεδα συνέλιξης (*convolution*) και συγκέντρωσης (*pooling*) τα οποία παράγουν τους πίνακες των χαρακτηριστικών και τους υποδειγματολειπτούς, αντίστοιχα. Ένα συνελικτικό δίκτυο μπορεί να αποτελείται από πολλαπλά ζεύγη αυτών των επιπέδων, ανάλογα με την πολυπλοκότητα του προβλήματος. Είναι σημαντικό να σημειωθεί ότι τα επίπεδα συγκέντρωσης δεν περιλαμβάνουν ελεύθερες παραμέτρους, δηλαδή βάρη και πολώσεις, καθώς καθορίζεται εξ αρχής η ακριβής λειτουργία τους και παραμένει σταθερή κατά τη διάρκεια της εκπαίδευσης αλλά και της λειτουργίας του δικτύου.

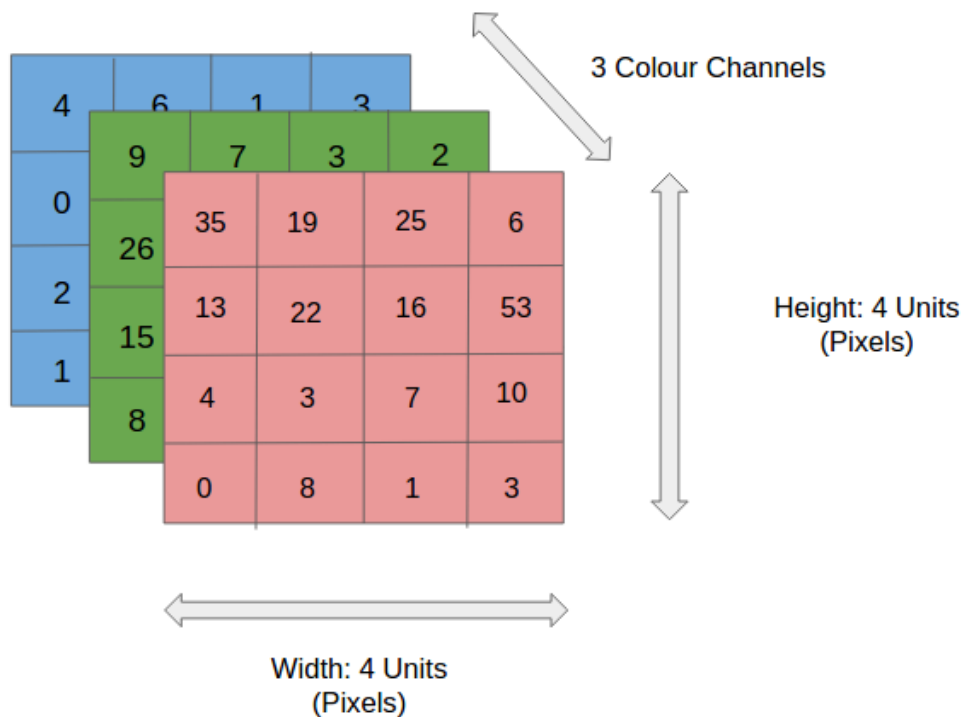
Στη συνέχεια, η πληροφορία συμπυκνώνεται σε μια διάσταση και τροφοδοτείται σε ένα ή περισσότερα πλήρως συνδεδεμένα στρώματα τα οποία είναι όμοια με τα παραδοσιακά πολυεπίπεδα δίκτυα πρόσθιας τροφοδότησης (κεφάλαιο 2.4.2). Τέλος, ανάλογα με τη φύση του προβλήματος επιλέγεται συγκεκριμένος αριθμός νευρώνων και συναρτήσεις ενεργοποίησης που σχηματίζουν το στρώμα εξόδου. Για παράδειγμα σε ένα πρόβλημα κατηγοριοποίησης (*classification*) αμοιβαία αποκλειόμενων κατηγοριών (δηλαδή κάθε δείγμα εισόδου αντιστοιχεί σε μόνο μία κατηγορία), ο αριθμός των νευρώνων στρώματος εξόδου είναι ίσος με το πλήθος των κατηγοριών αυτών.



Σχήμα 3.1: Αναπαράσταση της αρχιτεκτονικής ενός συνελικτικού δικτύου.

3.2.1 Επίπεδο εισόδου (*Input layer*)

Το επίπεδο εισόδου είναι το αρχικό επίπεδο του δικτύου μέσω του οποίου εισέρχονται τα ακατέργαστα δεδομένα στο δίκτυο. Στην πιο συνηθισμένη περίπτωση τα δεδομένα εισόδου είναι εικόνες και στην είσοδο καθορίζεται ένα διάνυσμα τεσσάρων διαστάσεων. Η πρώτη διάσταση σχετίζεται με το μέγεθος παρτίδας (*batch size*), οι επόμενες δύο διαστάσεις αφορούν το μήκος και το πλάτος των εικόνων που θα τροφοδοτηθούν στο δίκτυο και η τελευταία διάσταση αφορά τον αριθμό των καναλιών, μέσω των οποίων περιγράφεται η πληροφορία κάθε pixel. Τυπικά, για έγχρωμες εικόνες τα κανάλια είναι τρία, για τις τιμές RGB (*Red Blue Green*) για κάθε εικονοστοιχείο, ενώ για ασπρόμαυρες εικόνες το κανάλι είναι μόνο ένα. Ο λόγος που καθορίζονται εξ αρχής οι διαστάσεις των δεδομένων είναι ότι τα συνελικτικά δίκτυα δεν είναι σχεδιασμένα για να λειτουργούν με δεδομένα μεταβαλλόμενων διαστάσεων. Επομένως δεν είναι δυνατή η τροφοδότηση στο δίκτυο εικόνων διαφορετικών διαστάσεων, πράγμα που καθιστά συνηθισμένη πρακτική την αλλαγή μεγέθους των εικόνων ώστε να συμφωνούν στις διαστάσεις. Φυσικά για την αντιμετώπιση προβλημάτων όπου κρίνεται απαραίτητη η χρήση εικόνων μεταβλητών διαστάσεων, υπάρχουν παρόμοιες αρχιτεκτονικές όπως για παράδειγμα τα πλήρως συνελικτικά δίκτυα (*Fully Convolutional Networks - FCN*).



Σχήμα 3.2: Αναπαράσταση RGB μιας εικόνας εισόδου.

3.2.2 Συνελικτικό επίπεδο (*Convolution layer*)

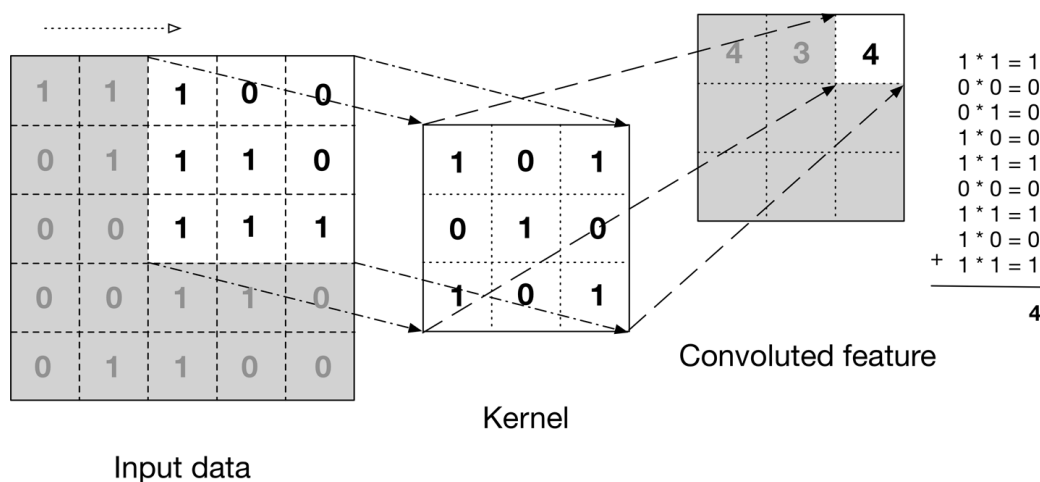
Το επίπεδο συνέλιξης είναι το βασικότερο δομικό στοιχείο της αρχιτεκτονικής ενός συνελικτικού δικτύου. Ο ρόλος αυτού του επιπέδου είναι η εξαγωγή χαρακτηριστικών (*feature extraction*) και γίνεται με τρόπο που εκμεταλλεύεται την τοπικότητα των χαρακτηριστικών αυτών.

Αρχικά συνέλιξη ονομάζεται μια μαθηματική πράξη που εκτελείται μεταξύ δύο συναρτήσεων και παράγει μια τρίτη συνάρτηση, η οποία εκφράζει τον τρόπο που επηρεάζει η μια συνάρτηση την άλλη. Στην περίπτωση των συνελικτικών δικτύων τον ρόλο της πρώτης συνάρτησης παίρνουν τα δεδομένα εισόδου, ενώ το ρόλο της δεύτερης παίρνουν τα φίλτρα. Η συνέλιξη αυτών των δύο παράγει τα χαρακτηριστικά, τα οποία απαρτίζουν τον χάρτη χαρακτηριστικών.

Για να γίνει σωστή κατανόηση του συνελικτικού επιπέδου, ακολουθεί η εξέταση των βασικών λειτουργικών στοιχείων του.

- **Πυρήνας (kernel):**

Πρόκειται για έναν δισδιάστατο τετραγωνικό πίνακα τιμών ο οποίος συνηθίζεται να έχει μικρό μήκος και πλάτος, συγκριτικά με το μέγεθος των εικόνων του δικτύου. Ο πυρήνας ολισθαίνει πάνω στα δεδομένα εισόδου και εφαρμόζεται η πράξη του πολλαπλασιασμού στοιχείο ανά στοιχείο μεταξύ του πυρήνα και των αντίστοιχων στοιχείων της εισόδου. Έπειτα προστίθενται τα αποτελέσματα αυτά και παράγουν μια τιμή. Με την ολοκλήρωση της διαδικασίας αυτής για όλες τις δυνατές ολισθήσεις, καταλήγουμε με έναν πίνακα που περιλαμβάνει τα αποτελέσματα κάθε συνέλιξης. Αυτός ο πίνακας ονομάζεται χάρτης χαρακτηριστικών (*feature map*) ή χάρτης ενεργοποίησης (*activation map*). Το πλήθος, οι διαστάσεις και το βήμα ολισθήσεως των πυρήνων αποτελούν υπερ-παραμέτρους (*hyperparameters*) του δικτύου και καθορίζονται στη σχεδίαση του εκάστοτε δικτύου. Οι τιμές όμως του πυρήνα, δηλαδή οι αριθμοί του πίνακα, αποτελούν ελεύθερες παραμέτρους μάθησης, δηλαδή μεταβάλλονται κατά τη διάρκεια της εκπαίδευσης του δικτύου.

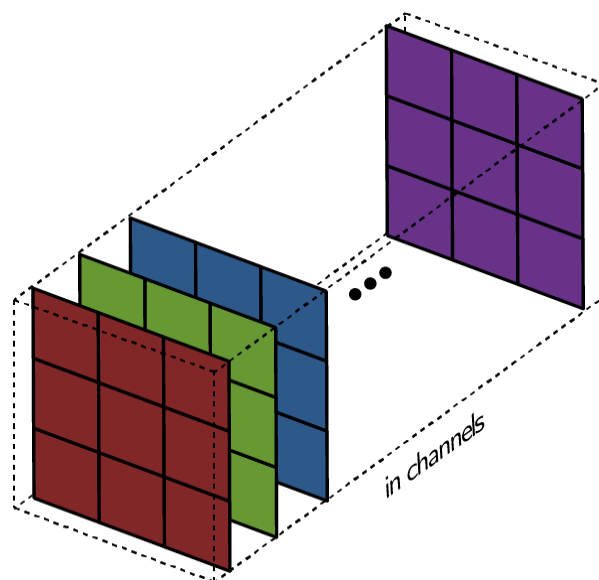


Σχήμα 3.3: Η πράξη της συνέλιξης για μια από τις ολισθήσεις ενός πυρήνα (3x3) πάνω στο δείγμα εισόδου (5x5).

- **Φίλτρο (filter):**

Το φίλτρο πολλές αναφέρεται σαν ταυτόσημη έννοια με τον πυρήνα. Αν θέλει όμως κανείς να είναι απόλυτα ακριβής θα πρέπει να διαχωρίσει τις δύο έννοιες. Όπως προαναφέρθηκε, στην πιο συνηθισμένη περίπτωση τα συνελικτικά δίκτυα καλούνται να επεξεργαστούν έγχρωμες εικόνες, οι οποίες εκτός από μήκος και ύψος, έχουν και βάθος που αντιστοιχεί στα κανάλια. Επομένως για κάθε κανάλι ορίζεται ξεχωριστός πυρήνας και ο συνδυασμός

αυτών ορίζει ένα φίλτρο. Με άλλα λόγια, ένα φίλτρο απαρτίζεται από τόσους πυρήνες, όσο και το βάθος της εισόδου, δηλαδή όσα και τα κανάλια χρώματος. Επομένως τα φίλτρα είναι αυτά που ολισθαίνουν πάνω στο δείγμα εισόδου (και κατά συνέπεια και οι πυρήνες τους) και μέσω της συνέλιξης παράγουν το χάρτη χαρακτηριστικών. Λόγω της ολίσθησης του φίλτρου πάνω σε ολόκληρο το δείγμα εισόδου, το δίκτυο δεν επηρεάζεται αν ένα χαρακτηριστικό δε βρίσκεται στην ίδια θέση σε κάθε δείγμα.

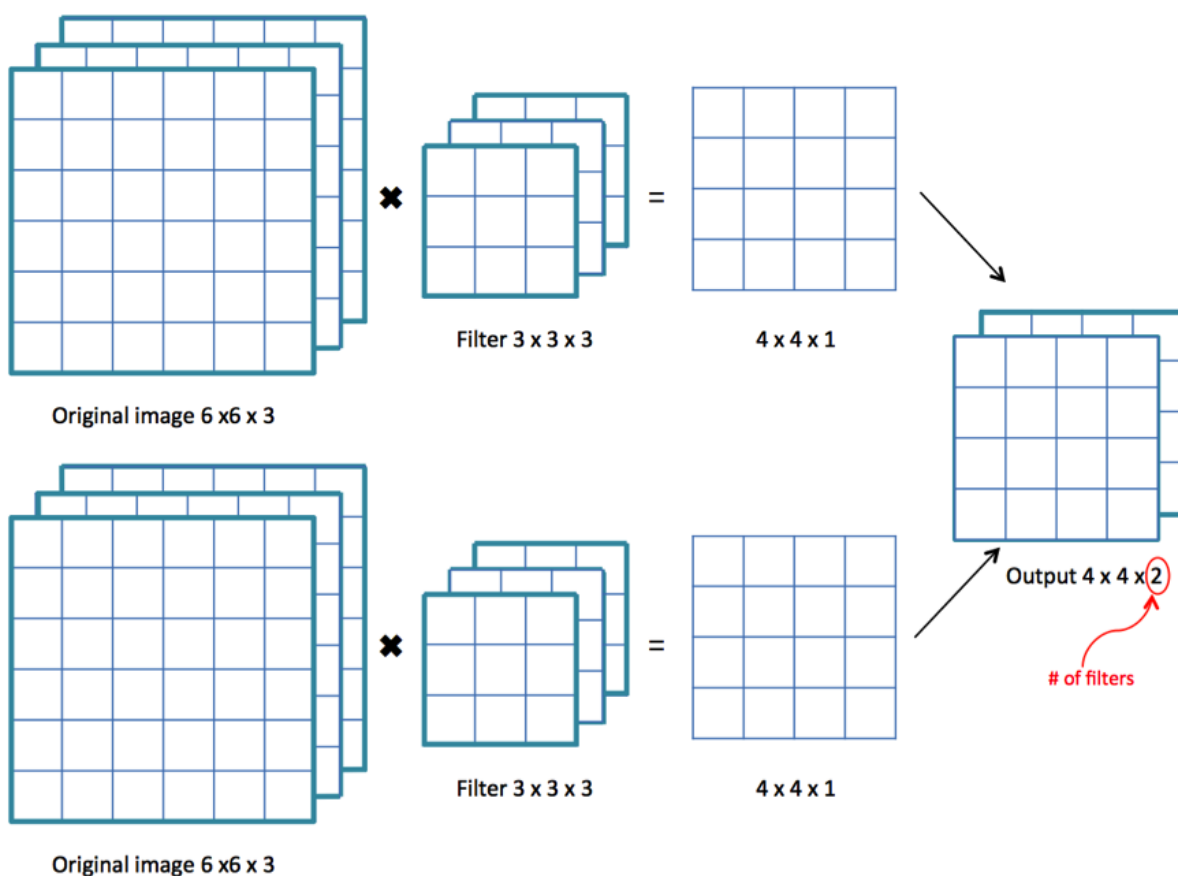


Σχήμα 3.4: Αναπαράσταση ενός φίλτρου που αποτελείται από C πυρήνες (3×3), όπου το C αντιστοιχεί στον αριθμό των καναλιών της εισόδου.

- **Χάρτης χαρακτηριστικών (feature map):**

Όπως αναφέρθηκε κάθε φίλτρο, όταν εφαρμόζεται στο δείγμα εισόδου, παράγει μια τιμή για κάθε ολίσθηση, η οποία αποθηκεύεται στον πίνακα χαρακτηριστικών. Έτσι, ο αριθμός στοιχείων του πίνακα χαρακτηριστικών καθορίζεται από τον αριθμό των δυνατών ολισθήσεων του φίλτρου πάνω στο δείγμα εισόδου, το οποίο με τη σειρά του ορίζεται μέσω των υπερ-παραμέτρων που επιλέγονται. Είναι φανερό επίσης ότι το πλήθος των φίλτρων ταυτίζεται με το πλήθος των πινάκων χαρακτηριστικών. Όλοι οι πίνακες χαρακτηριστικών που εξάγονται στοιβάζονται και δημιουργούν μία τρισδιάστατη αναπαράσταση, δηλαδή τον χάρτη χαρακτηριστικών. Στις περιπτώσεις των στοιβαγμένων επιπέδων συνέλιξης, ο χάρτης χαρακτηριστικών μπορεί να αποτελεί και είσοδο στο συνελκτικό επίπεδο, πράγμα που προσφέρει τη δυνατότητα σύνδεσης πιο απλών χαρακτηριστικών για την ανακάλυψη νέων περισσότερο περίπλοκων.

Στο παρακάτω παράδειγμα φαίνεται ένα δείγμα εισόδου διαστάσεων $6 \times 6 \times 3$ στο οποίο εκτελείται συνέλιξη με δύο φίλτρα διαστάσεων $3 \times 3 \times 3$. Επειδή το βήμα ολίσθησης έχει οριστεί ίσο με ένα, για κάθε μία από τις τέσσερις δυνατές ολισθήσεις στις γραμμές, έχουμε τέσσερις δυνατές ολισθήσεις στις στήλες. Συνεπώς οι διαστάσεις του χάρτη χαρακτηριστικών είναι $4 \times 4 \times 2$.

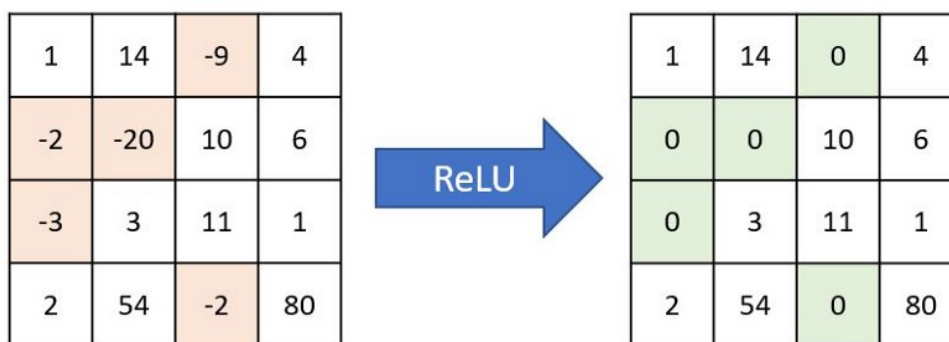


Σχήμα 3.5: Παράδειγμα εξαγωγής του χάρτη χαρακτηριστικών.

- **Συνάρτηση ενεργοποίησης ReLU.**

Στην κάθε τιμή του χάρτη χαρακτηριστικών εφαρμόζεται η συνάρτησης ενεργοποίησης της ανορθωμένης γραμμικής συνάρτησης ράμπας ή αλλιώς γνωστή ως ReLU. Η συνάρτηση αυτή περιγράφηκε στο κεφάλαιο (2.3.3) και πρόκειται στην ουσία για μια αποκοπή των αρνητικών τιμών. Επιλέγεται αυτή η συνάρτηση καθώς αυξάνει τη μη γραμμικότητα του δικτύου.

Για παράδειγμα αν το δείγμα εισόδου παρουσιάζει ένα τοπικό χαρακτηριστικό, τότε στις αντίστοιχες θέσεις του χάρτη χαρακτηριστικών θα υπάρχουν θετικές τιμές ενώ στις υπόλοιπες μπορεί οι τιμές να παίρνουν μηδενικές ή αρνητικές τιμές. Με το μηδενισμό των αρνητικών τιμών παραμένει ενεργοποιημένη μόνο η περιοχή του χαρακτηριστικού.



Σχήμα 3.6: Χάρτης χαρακτηριστικών 4x4x1 πριν και μετά την εφαρμογή της ReLU.

- **Υπερ-παράμετροι:**

Οι παράμετροι που καθορίζονται στο σχεδιασμό του συνελικτικού επιπέδου και επιλέγονται ανάλογα με τη φύση και πολυπλοκότητα του προβλήματος είναι οι εξής:

- Διαστάσεις πυρήνων (*kernel size*)
- Πλήθος φίλτρων (*filters*)
- Βήμα ολίσθησης (*stride*)
- Γέμισμα (*padding*)

Το γέμισμα αφορά τη συμπλήρωση τιμών συμμετρικά στα άκρα του δείγματος εισόδου. Αυτό αποσκοπεί αρχικά στη διατήρηση της διάστασης της εισόδου και επίσης στη αποφυγή της απώλειας σημαντικής πληροφορίας που μπορεί να βρίσκεται στα άκρα του δείγματος εισόδου. Συνήθως επιλέγεται το γέμισμα με μηδενικά (*zero-padding*).

Ένα σημαντικό χαρακτηριστικό του συνελικτικού επιπέδου είναι η ιδιαιτερότητα του δεκτικού πεδίου (*receptive field*) των νευρώνων. Το δεκτικό πεδίο ενός νευρώνα είναι το σύνολο των νευρώνων του προηγούμενου επιπέδου που συνδέονται με αυτόν. Στα παραδοσιακά πλήρως συνδεδεμένα δίκτυα, το δεκτικό πεδίο κάθε

νευρώνα περιλαμβάνει όλους τους νευρώνες του προηγούμενου στρώματος. Στα συνελκτικά επίπεδα, όπως φαίνεται και από την αρχιτεκτονική των φίλτρων, κάθε νευρώνας λαμβάνει είσοδο μόνο από μια περιορισμένη τοπικά περιοχή του προηγούμενου στρώματος. Αυτό μειώνει σημαντικά τις παραμέτρους του δικτύου και κατα συνέπεια τον χρόνο εκπαίδευσης. Όπως προαναφέρθηκε κάθε φίλτρο χρησιμοποιείται για να ανιχνεύσει ένα συγκεκριμένο χαρακτηριστικό, για παράδειγμα έναν κύκλο, όπου και αν βρίσκεται στην εικόνα. Αν λοιπόν υπάρχουν πολλαπλοί κύκλοι σε μια εικόνα, εντοπίζονται με χρήση του ίδιου φίλτρου. Εφόσον λοιπόν οι τιμές των φίλτρων αποτελούν στην ουσία τα βάρη του δικτύου, φαίνεται ότι πολλοί νευρώνες μοιράζονται τα ίδια βάρη. Αυτό ονομάζεται διαμοιρασμός παραμέτρων (*parameter sharing*) και αποτελεί μεγάλο πλεονέκτημα των συνελκτικών επιπέδων.

3.2.3 Επίπεδο συγκέντρωσης (*Pooling layer*)

Τα επίπεδα συγκέντρωσης εισάγονται συνήθως μετά το συνελκτικό επίπεδο και τη συνάρτηση ενεργοποίησης ReLU, ειδικότερα όταν πρόκειται ν' ακολουθήσει κι άλλο συνελκτικό επίπεδο.

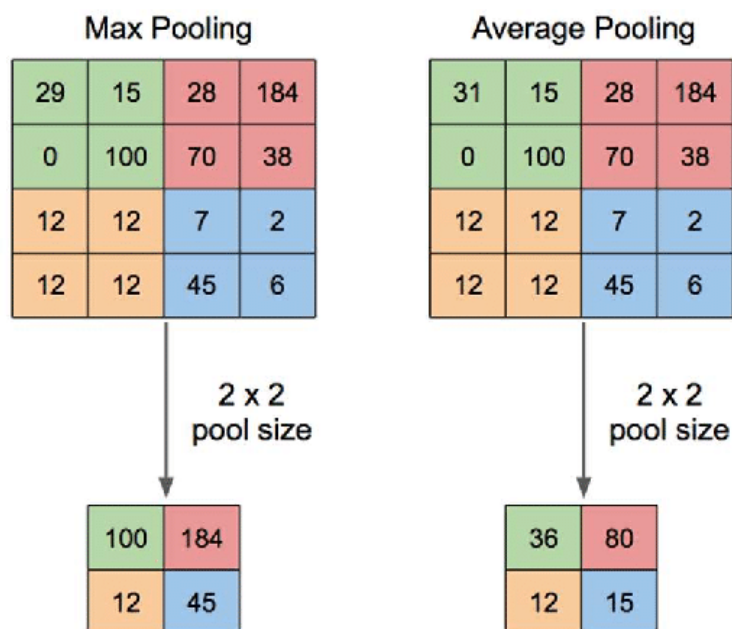
Ένα πρόβλημα που συναντάται σε πολλές εφαρμογές νευρωνικών δικτύων είναι η υπερπροσαρμογή (*overfitting*) του δικτύου. Αυτό συμβαίνει όταν το δίκτυο στην ουσία αντί να μαθαίνει, απομνημονεύει τη μορφή των δεδομένων εισόδου. Μπορεί να θεωρηθεί ως το αντίθετο της περίπτωσης όπου το δίκτυο δεν έχει εκπαιδευτεί αρκετά και γνωρίζει πολύ μικρό μέρος της πληροφορίας των δεδομένων. Πρόκειται λοιπόν για το φαινόμενο όπου το δίκτυο μαθαίνει λεπτομερής πληροφορίες ή χαρακτηριστικά των δεδομένων εκπαίδευσης, τα οποία όμως δεν συνεισφέρουν στη γενίκευση.

Το προαναφερθέν πρόβλημα είναι δυνατό να προκύψει μετά από ένα συνελκτικό επίπεδο, καθώς οι χάρτες χαρακτηριστικών είναι πιθανό να περιλαμβάνουν λεπτομερή πληροφορία και χαρακτηριστικά. Τα επίπεδα συγκέντρωσης υποδειγματολειπτούν τους χάρτες χαρακτηριστικών, διατηρώντας μόνο τη σημαντικότερη πληροφορία. Ο τρόπος που γίνεται αυτό είναι παρόμοιος με τον τρόπο λειτουργίας των φίλτρων του επιπέδου συνέλιξης. Για κάθε ολίσθηση ενός παραθύρου πάνω στους χάρτες χαρακτηριστικών, εφαρμόζεται μια συνάρτηση η οποία υπολογίζει μια μοναδική τιμή. Οι ευρέως χρησιμοποιούμενες τέτοια συναρτήσεις είναι ο υπολογισμός μεγίστου (*max pooling*) ή μέσου όρου (*average pooling*).

- **Υπερ-παράμετροι:**

Οι παράμετροι που καθορίζονται στο σχεδιασμό του επιπέδου συγκέντρωσης και επιλέγονται ανάλογα με τη φύση και πολυπλοκότητα του προβλήματος είναι οι εξής:

- Διαστάσεις παραθύρου (*pool size*)
- Βήμα ολίσθησης (*stride*)



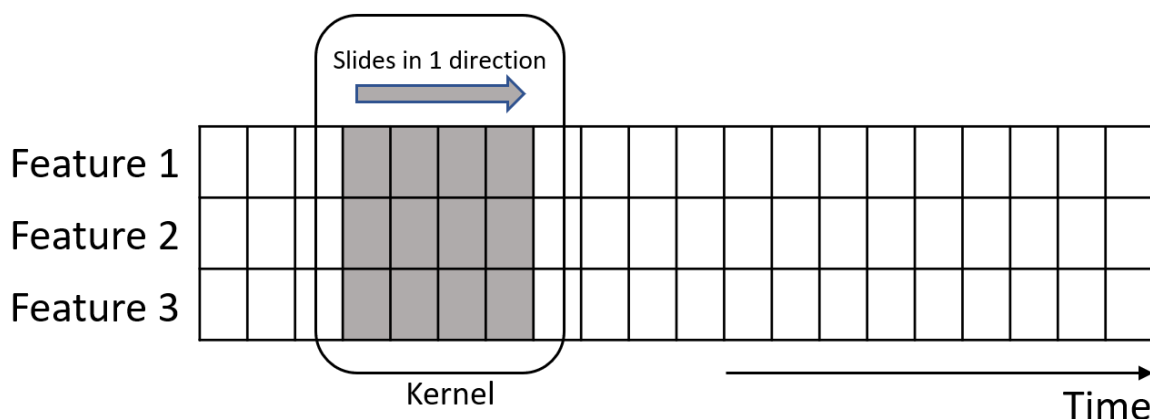
Σχήμα 3.7: Παράδειγμα συγκέντρωσης μεγίστου (αριστερά) και μέσου όρου (δεξιά), με μέγεθος παραθύρου 2x2 και βήμα ολίσθησης ίσο με 2.

3.2.3 Πλήρως συνδεδεμένο επίπεδο (*Fully connected layer*)

Το επίπεδο αυτό περιγράφεται στο κεφάλαιο (κεφάλαιο 2.4.2) και πρόκειται για την παραδοσιακή αρχιτεκτονική νευρωνικών δικτύων. Ο λόγος που το καθιστά απαραίτητο στα συνελκτικά δίκτυα είναι η ανάγκη της τελικής πρόβλεψης. Με το συνδυασμό των επιπέδων που εξετάστηκαν παραπάνω επιτυγχάνεται μόνο η εξαγωγή χαρακτηριστικών της εισόδου. Τη χρήση των χαρακτηριστικών αυτών για την δημιουργία της τελικής πρόβλεψης αναλαμβάνει το πλήρως συνδεδεμένο επίπεδο. Αν είναι επιθυμητό δύναται να προστεθούν περισσότερα από ένα πλήρως συνδεδεμένα στρώματα με σκοπό την εκμάθηση μη γραμμικών συνδυασμών των χαρακτηριστικών που έχουν εξαχθεί από τα συνελκτικά επίπεδα. Το τελευταίο πλήρες συνδεδεμένο στρώμα, είναι και το τελικό στρώμα εξόδου του δικτύου.

3.3 Συνελικτικά δίκτυα μιας διάστασης

Ο όρος συνελικτικά δίκτυα συνεπάγεται την ευρέως χρησιμοποιούμενη μορφή τους, δηλαδή τα δισδιάστατα συνελικτικά δίκτυα. Η ονομασία αυτή πηγάζει από τις διαστάσεις της εισόδου που λαμβάνουν. Όπως έχει ήδη τονιστεί τα συνελικτικά δίκτυα καλούνται κατά το πλείστον να αντιμετωπίσουν προβλήματα που πραγματεύονται οπτική πληροφορία, δηλαδή εικόνες. Τα συνελικτικά δίκτυα μιας διάστασης όμως, αποτελούν εξαίρεση σε αυτό τον κανόνα καθώς έχουν τη δυνατότητα να διαχειριστούν γενικότερες μορφές πληροφορίας, με χαρακτηριστικό παράδειγμα τα δεδομένα χρονοσειρών. Οι χρονοσειρές περιγράφουν οποιασδήποτε μορφής δεδομένα τα οποία συλλέγονται διαδοχικά στο χρόνο. Σε αυτή την περίπτωση, τα συνελικτικά δίκτυα μιας διάστασης αντί να εξάγουν τοπικά χαρακτηριστικά της εισόδου, εντοπίζουν και εξάγουν χαρακτηριστικά σε σχέση με το χρόνο. Η αρχιτεκτονική και η λειτουργία των συνελικτικών δικτύων μιας διάστασης είναι όμοια με αυτή των δύο διαστάσεων, η οποία αναφέρθηκε παραπάνω (κεφάλαιο 3.2). Η μοναδική διαφορά που διακρίνεται, είναι στις διαστάσεις της εισόδου και τον λειτουργικών στοιχείων του δικτύου. Η είσοδος πλέον αφορά ένα μέτρο ή μια τιμή μιας μεταβλητής σε κάθε διακριτή μονάδα του χρόνου, όπως για παράδειγμα η ταχύτητα ενός αυτοκινήτου ανά κάποιο χρονικό βήμα. Στα προβλήματα που περιλαμβάνουν περισσότερες από μια τέτοιες μεταβλητές, αυτές στοιβάζονται σε μια δεύτερη διάσταση. Στην ουσία αν μια εικόνα περιγράφεται από τρεις δισδιάστατους πίνακες n γραμμών και n στηλών (έναν για κάθε κανάλι χρώματος), τότε η αντιστοιχία εδώ είναι k πίνακες μιας γραμμής και n στηλών, όπου k το πλήθος των διαφορετικών μεταβλητών. Οι πυρήνες δεν ολισθαίνουν πλέον σε δύο διαστάσεις, αλλά μόνο σε μία, στη διάσταση του χρόνου. Κατά αντιστοιχία η μοναδική διαφορά στα υπόλοιπα δομικά στοιχεία (φίλτρα, χάρτης χαρακτηριστικών) εντοπίζεται στις διαστάσεις. Η λειτουργία, η αρχιτεκτονική ακόμα και οι υπερ-παραμέτροι ακολουθούν όμοια λογική με τα συνελικτικά δίκτυα δύο διαστάσεων.



Σχήμα 3.8: Αναπαράσταση ολίσθησης του πυρήνα σε είσοδο τριών μεταβλητών.

Κεφάλαιο 4

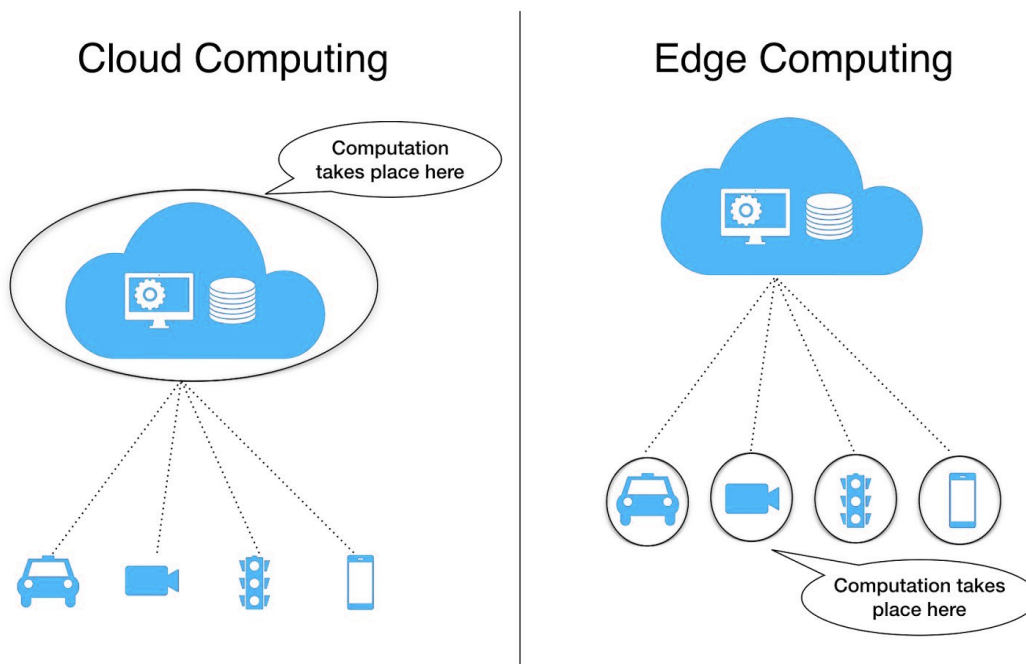
Edge Computing

4.1 Ορισμός

Το edge computing περιγράφει ένα πρότυπο καταναμημένων συστημάτων που φέρνει τον υπολογισμό και την αποθήκευση δεδομένων πιο κοντά στην τοποθεσία όπου απαιτείται, για τη βελτίωση των χρόνων απόκρισης και την εξοικονόμηση εύρους ζώνης.

Ως αντίθετος όρος θα μπορούσε να θεωρηθεί υπό μία έννοια το cloud computing, το οποίο αντιπροσωπεύει τη παροχή υπολογιστικών υπηρεσιών και αποθήκευσης δεδομένων απομακρυσμένα, μέσω του διαδικτύου. Για παράδειγμα, ένας χρήστης που επιθυμεί να εκτελέσει μια υπολογιστική διεργασία, επικοινωνεί με το απομακρυσμένο σύστημα και αυτό αφού εκτελέσει τον υπολογισμό, στέλνει τα αποτελέσματα πίσω στον χρήστη. Επομένως η επεξεργασία και αποθήκευση της πληροφορίας γίνεται απομακρυσμένα και όχι κοντά στον χρήστη. Ο χρόνος που απαιτείται για να ταξιδέψει η πληροφορία από τον χρήστη στο απομακρυσμένο σύστημα και πίσω, ονομάζεται καθυστέρηση (*latency*) και αποτελεί μία από τις καθοριστικές διαφορές των cloud και edge computing.

Συγκριτικά με το cloud, το edge computing επιλέγει τη μεταφορά του υπολογιστικού φόρτου εκτός του νέφους (*cloud*) και εντός του άκρου (*edge*) του δικτύου, εκεί δηλαδή που πρόκειται να χρησιμοποιηθεί η πληροφορία. Δεδομένου λοιπόν ότι δε χρειάζεται η μεταφορά πληροφορίας μέσα στο δίκτυο, θεωρείται ιδανική επιλογή για εφαρμογές που πραγματεύονται στιγμιαία δεδομένα, δηλαδή δεδομένα που συλλέγονται σε πραγματικό χρόνο. Φυσικά το edge υστερεί σε υπολογιστική και αποθηκευτική δύναμη σε σχέση με το cloud. Οι εξυπηρετητές και βάσεις δεδομένων στο cloud απαρτίζονται από ισχυρά υπολογιστικά συστήματα τα οποία έχουν σχεδιαστεί με γνώμονα την παροχή των δυνατοτήτων τους σε μεγάλο αριθμό χρηστών. Κάτι τέτοιο θα ήταν αδύνατο οικονομικά και πρακτικά να υλοποιηθεί στο edge για κάθε χρήστη. Συνεπώς φαίνεται ότι το edge computing δεν καταργεί το ρόλο του cloud computing, αλλά αντιθέτως τον συμπληρώνει. Οι κόμβοι χρησιμοποιούν edge computing για να εξυπηρετούν άμεσα τις εφαρμογές για τις οποίες διαθέτουν τους απαραίτητους πόρους, ενώ αναθέτουν σε κάποιον εξυπηρετητή cloud την εξυπηρέτηση αυτών που δεν είναι κρίσιμα ευάλωτες σε καθυστέρηση.



Σχήμα 4.1: Απλοποιημένη αναπαράσταση cloud (αριστερά) και edge (δεξιά) computing. Κυκλώνεται η τοποθεσία όπου γίνεται η υπολογιστική εργασία.

4.2 Πλεονεκτήματα

Η χρήση του προτύπου edge computing προσφέρει ορισμένα πλεονεκτήματα και δυνατότητες τα οποία μπορεί να φανούν καθοριστικά για ορισμένες εφαρμογές. Αυτά μπορούν να ομαδοποιηθούν στις παρακάτω κατηγορίες.

- **Εύρος ζώνης (bandwidth):**

Ο πολλαπλασιασμός των έξυπνων συσκευών στην καθημερινότητα των ανθρώπων σημαίνει ότι παράγεται μια εξαιρετική ποσότητα δεδομένων, χωρίς όμως όλα τα δεδομένα αυτά να είναι χρήσιμα. Για παράδειγμα σε ένα cloud σύστημα ασφαλείας με κάμερες, κάθε κάμερα μεταδίδει δεδομένα στο cloud με αποτέλεσμα τη χρήση μεγάλου εύρους ζώνης στο δίκτυο. Η πληροφορία που έχει αξία σε αυτό το παράδειγμα είναι τα στιγμιότυπα παραβίασης της ασφάλειας, τα οποία και θα αποτελούν μειοψηφία. Το edge παρέχει τη δυνατότητα στις κάμερες να επεξεργαστούν τοπικά αυτά τα δεδομένα και να μεταδώσουν μόνο το κρίσιμο κομμάτι τους στο cloud και άρα συμβάλλει σημαντικά στην εξοικονόμηση εύρους ζώνης του δικτύου.

- **Καθυστέρηση (*latency*):**

Ένα άλλο πλεονέκτημα των συσκευών edge είναι η μεγάλη μείωση της καθυστέρησης, δηλαδή του χρόνου που απαιτείται για την αποστολή δεδομένων και τη λήψη απάντησης. Με το πρότυπο cloud computing, μια συσκευή ενδεχεται να στέλνει πληροφορίες για επεξεργασία σε ένα κέντρο δεδομένων στην άλλη πλευρά του κόσμου και αυτό συχνά οδηγεί σε μια χρονική καθυστέρηση. Αυτό βέβαια δεν αποτελεί πρόβλημα για όλες τις εφαρμογές. Όταν όμως απαιτείται να ληφθεί απόφαση από το σύστημα σε πραγματικό χρόνο, τότε η καθυστέρηση δεν είναι αποδεκτή. Ένα τέτοιο παράδειγμα είναι ένα αυτοκινούμενο όχημα στο δρόμο, όπου μια καθυστέρηση στην απόφαση που θα λάβει το σύστημα μπορεί να καθορίσει την ύπαρξη ατυχήματος ή αποφυγής του.

- **Απόρρητο (*privacy*):**

Το edge computing προσφέρει στο χρήστη ιδιοκτησία και μεγαλύτερο έλεγχο των δεδομένων που παράγει. Θεωρητικά, με λιγότερα δεδομένα να μεταφορτώνονται στο cloud και περισσότερα δεδομένα να υποβάλλονται σε επεξεργασία τοπικά στη συσκευή του χρήστη, περιορίζεται η πρόσβαση εταιρειών και οργανισμών σε αυτά. Οι εταιρείες χρησιμοποιούν αυτά τα δεδομένα για να βελτιώσουν τα προϊόντα τους, για διαφημιστικούς αλλά και άλλους σκοπούς, πράγμα που τα καθιστά εξαιρετικά πολύτιμα και κερδοφόρα για αυτές. Επομένως φαντάζει δύσκολη η καθιέρωση του προτύπου edge computing σε πολλά προϊόντα όπως οι έξυπνοι οικιακοί βοηθοί και άλλα.

Requirement	Cloud	Edge
Distribution	Centralized	Decentralized
Distance to client/server	Off-Shore	On-Premises
Latency	High	Low
Location Awareness	No	Yes
Support Mobility	Limited	Full
Location of Service	Within Internet	Device
Data Storage	High	Low

Σχήμα 4.2: Συγκριτικός πίνακας ορισμένων χαρακτηριστικών μεταξύ cloud και edge.

4.3 Προκλήσεις

Όπως αναφέρθηκε το edge computing κατευθύνει το υπολογιστικό φορτίο από τα κέντρα δεδομένων στην άκρη του δικτύου, δηλαδή στους κόμβους των χρηστών. Αυτή η κατανομή των υπηρεσιών σε διαφορετικούς κόμβους του δικτύου, εκτός από θετικά στοιχεία, παρουσιάζει επίσης διάφορες προκλήσεις και προβλήματα τα οποία συνοψίζονται στις κατηγορίες που ακολουθούν.

- **Ασφάλεια** (*security*):

Ο καταναμημένος χαρακτήρας του προτύπου edge computing εισάγει μια μετατόπιση των συστημάτων ασφαλείας που χρησιμοποιούνται στο cloud computing. Στο edge computing, τα δεδομένα ενδέχεται να μετακινούνται μεταξύ διαφορετικών καταναμημένων κόμβων που συνδέονται μέσω του διαδικτύου και συνεπώς απαιτούνται ειδικοί μηχανισμούς κρυπτογράφησης ανεξάρτητοι από το cloud. Οι κόμβοι επίσης ενδέχεται να είναι περιορισμένοι σε υπολογιστικούς πόρους με αποτέλεσμα τον περιορισμό της επιλογής αποτελεσματικών μεθόδων ασφαλείας.

- **Επεκτασιμότητα** (*scalability*):

Ο όρος αυτός περιγράφει τη δυνατότητα της επέκτασης του συστήματος, δηλαδή για παράδειγμα της προσθήκης νέων κόμβων. Για την αποδοτική επεκτασιμότητα ενός δικτύου edge computing πρέπει να ληφθεί υπόψη η ετερογένεια των συσκευών, έχοντας διαφορετικούς περιορισμούς απόδοσης και ενέργειας, την εξαιρετικά δυναμική κατάσταση καθώς και την αξιοπιστία των συνδέσεων του δικτύου, σε σύγκριση με τις ισχυρές υποδομές των κέντρων δεδομένων cloud.

- **Αξιοπιστία** (*reliability*):

Η σωστή διαχείριση των σφαλμάτων είναι πρωταρχικής σημασίας παράγοντας για τη διατηρηθεί μία υπηρεσία ζωντανή. Εάν για παράδειγμα ένας κόμβος σταματήσει να λειτουργεί και δεν είναι προσβάσιμος, οι χρήστες θα πρέπει να έχουν τη δυνατότητα πρόσβασης στην υπηρεσία χωρίς διακοπές. Άρα τα συστήματα edge computing είναι απαραίτητο να διαθέτουν λειτουργίες αντιμετώπισης των σφαλμάτων και ενημέρωσης του χρήστη για το συμβάν. Για το σκοπό αυτό, κάθε συσκευή πρέπει να διατηρεί την τοπολογία δικτύου ολόκληρου του καταναμημένου συστήματος, έτσι ώστε η ανίχνευση σφαλμάτων και η ανάκτηση από αυτά να γίνεται εύκολα εφαρμόσιμη.

- **Υπολογιστική ισχύς** (computer power):
Οι συσκευές στα άκρα του δικτύου είναι τις περισσότερες φορές περιορισμένες από την άποψη του τεχνικού υπολογιστικού αποτυπώματος. Είναι προφανές ότι σε μια φορητή ή ενσωματωμένη συσκευή για παράδειγμα, δεν είναι δυνατή η παροχή υπολογιστικής ισχύος της τάξης μεγέθους ενός κέντρου δεδομένων.

4.4 Edge μονάδα επεξεργασίας τανυστή

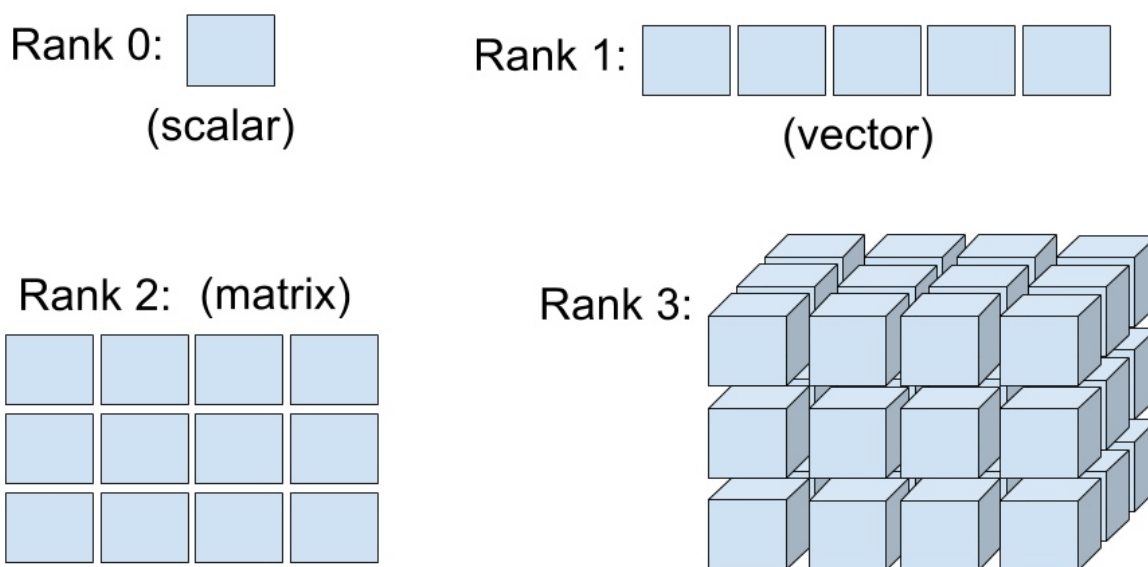
(*Edge Tensor Processing Unit - Edge TPU*)

Για την μελέτη της μονάδας επεξεργασίας τανυστή είναι προφανές ότι πρέπει να προηγηθεί η επεξήγηση της έννοιας του τανυστή.

4.4.1 Τανυστής (*Tensor*)

Στα μαθηματικά οι τανυστές είναι γεωμετρικά αντικείμενα που μπορούν να θεωρηθούν ως γενικευμένα διανύσματα. Περιγράφουν γραμμικές σχέσεις ανάμεσα σε διανύσματα, βαθμωτά μεγέθη και άλλους τανυστές. Βασικά παραδείγματα τέτοιων σχέσεων περιλαμβάνουν το εσωτερικό γινόμενο, το εξωτερικό γινόμενο και γραμμικούς μετασχηματισμούς. Τα διανύσματα και τα βαθμωτά μεγέθη είναι επίσης τανυστές. Οι τανυστές χρησιμοποιούνται για να αναπαραστήσουν αντιστοιχίες ανάμεσα σε σύνολα γεωμετρικών διανυσμάτων. Ένας τανυστής μπορεί να απεικονιστεί σαν μία πολυδιάστατη διάταξη αριθμητικών τιμών. Η τάξη ή βαθμός ενός τανυστή είναι η διαστατικότητα της διάταξης που χρειάζεται για να τον απεικονίσει ή ισοδύναμα, ο αριθμός των δεικτών που χρειάζονται για να γίνει αναφορά σε ένα στοιχείο αυτής της διάταξης. Για παράδειγμα, ένας γραμμικός μετασχηματισμός μπορεί να απεικονιστεί από έναν πίνακα, μία δισδιάστατη διάταξη και επομένως είναι τανυστής δεύτερης τάξης. Ένα διάνυσμα μπορεί να απεικονιστεί σαν μία μονοδιάστατη διάταξη και άρα είναι τανυστής πρώτης τάξης. Τα βαθμωτά μεγέθη είναι απλοί αριθμοί και συνεπώς τανυστές μηδενικής τάξης.

Στις πρακτικές εφαρμογές βαθιάς μάθησης ένα από ευρέως χρησιμοποιούμενα frameworks (βιβλιοθήκες) είναι το tensorflow, το οποίο αναπτύσσεται από την Google. Λόγω του όγκου και της πολλαπλότητας των διαστάσεων των δεδομένων που απαιτούνται στις εφαρμογές βαθιάς μάθησης, το tensorflow έχει καθιερώσει σαν βασική δομή τους τένσορες, τους οποίους και υλοποιεί σαν πολυδιάστατους πίνακες. Αν και θα ήταν λανθασμένο να θεωρηθεί ταύτιση του μαθηματικού όρου με αυτόν που χρησιμοποιείται στον προγραμματισμό, ο τελευταίος έχει κληρονομήσει πολλά από τα βασικά στοιχεία του πρώτου.



Σχήμα 4.3: Αναπαράσταση τανυστών διαφόρων βαθμών (rank).

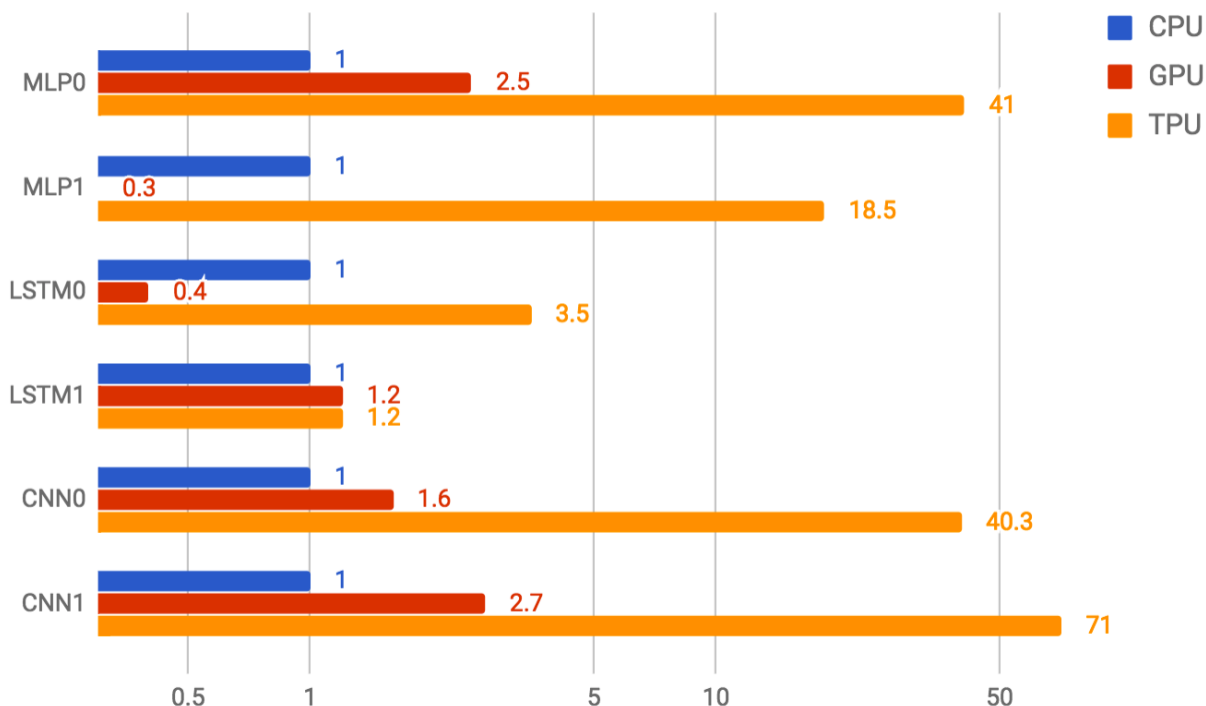
4.4.2 Μονάδα επεξεργασίας τανυστή

Η μονάδα επεξεργασίας τανυστή (*Tensor Processing Unit - TPU*) είναι ένα ολοκληρωμένο κύκλωμα που υπάγεται στην κατηγορία των ASIC (*application-specific integrated circuit*), το οποίο προορίζεται για επιτάχυνση εφαρμογών νευρωνικών δικτύων. Σχεδιάστηκε για αποκλειστική χρήση σε συνδυασμό με τη βιβλιοθήκη tensorflow και παρέχεται από την Google σαν υπηρεσία cloud ή σαν ανεξάρτητη συσκευή στο edge.

Πριν την καινοτομία των TPUs οι μονάδες επεξεργασίας που ήταν διαθέσιμες για εκπαίδευση νευρωνικών δικτύων ήταν οι κεντρικές μονάδες επεξεργασίας (*Central Processing Unit - CPU*) και οι μονάδες επεξεργασίας γραφικών (*Graphics Processing Unit - GPU*). Η πρώτη επιλογή δεν αποτελεί αποδοτική λύση καθώς πρόκειται για επεξεργαστές γενικής χρήσης με μικρό αριθμό πυρήνων και άρα περιορισμένη ικανότητα παράλληλων υπολογισμών. Κάτι τέτοιο είναι λογικό καθώς οι συγκεκριμένες μονάδες σχεδιάστηκαν με γνώμονα τη πολυχρηστικότητα, δηλαδή με απλά λόγια οι CPUs μπορούν να εκτελέσουν πολλές διαφορετικές εργασίες, αλλά αδυνατούν να διαχειριστούν τις υπολογιστικά βαρύτερες αποδοτικά. Οι GPUs από την άλλη εμφάνισαν μεγαλύτερη προοπτική. Η αρχιτεκτονική τους βασίζεται στη δυνατότητα τους να παράγουν και να επεξεργάζονται γραφικά, πράγμα που απαιτεί την παράλληλη επεξεργασία μεγάλων πακέτων δεδομένων.

Οι CPUs γενικής χρήσης αυτή τη στιγμή δομούνται με τέσσερις έως οκτώ πυρήνες ενώ οι αντίστοιχες GPUs περιλαμβάνουν εκατοντάδες πυρήνες. Αν και ένας μεμονωμένος πυρήνας CPU υπερέχει του αντίστοιχου πυρήνα GPU, ο μεγάλος αριθμός τους και ο παραλληλισμός που προσφέρει πετυχαίνει σημαντικά καλύτερους χρόνους εκπαίδευσης στα νευρωνικά δίκτυα.

Η ραγδαία ανάπτυξη της μηχανικής μάθησης και η κατασκευή όλο και πολυπλοκότερων μοντέλων δημιούργησε την ανάγκη ύπαρξης μιας μονάδας επεξεργασίας ειδικά σχεδιασμένη για αυτό τον σκοπό. Την απάντηση έφερε η Google με τη δημιουργία των TPUs, δηλαδή ειδικά διαμορφωμένων μονάδων επεξεργασίας για τη βελτιστοποίηση εφαρμογών μηχανικής μάθησης. Οι δοκιμές που έγιναν με χρήση της TPU πρώτης γενιάς δείχνουν απόδοση 83x ανά watt κέρδος έναντι της CPU και 29x κέρδος έναντι της GPU.



Σχήμα 4.4: Συγκριτικό διάγραμμα προβλέψεων ανά δευτερόλεπτο μεταξύ CPU, GPU και TPU για διάφορα μοντέλα νευρωνικών δικτύων.

Ο τρόπος που η TPU πετυχαίνει τόσο υψηλή απόδοση για εφαρμογές νευρωνικών δικτύων εντοπίζεται στην ιδιαίτερη αρχιτεκτονική της και συνοψίζεται στα παρακάτω στοιχεία.

- **CISC και μονάδα πολλαπλασιασμού πινάκων (MXU):**

Αρχικά η TPU ακολουθεί αρχιτεκτονική εντολών CISC αντί για RISC, δηλαδή χρησιμοποιούνται κομμάτια υλικού που καταλαβαίνουν και εκτελούν μια σειρά από εντολές assembly. Για παράδειγμα η πράξη του πολλαπλασιασμού δύο αριθμών σε αρχιτεκτονική CISC θα εκτελούνταν κατευθείαν από μια μονάδα υλικού, ενώ σε αρχιτεκτονική RISC η ίδια πράξη θα εκτελούνταν ως ξεχωριστές εντολές. Σε αυτή τη λογική βασίζεται η μονάδα πολλαπλασιασμού πινάκων (*Matrix Multiplier Unit - MXU*) που αποτελεί καθοριστικό παράγοντα της απόδοσης της TPU σε εφαρμογές νευρωνικών δικτύων, καθώς αυτές περιλαμβάνουν κατά το πλείστον παράλληλους πολλαπλασιασμούς πινάκων.

- **Συστολικός πίνακας (systolic array):**

Οι συστολικοί πίνακες είναι δομές υλικού κατασκευασμένες για γρήγορη και αποτελεσματική λειτουργία αλγορίθμων που εκτελούν την ίδια εργασία με διαφορετικά δεδομένα σε διαφορετικές χρονικές στιγμές. Στην περίπτωση μας η εργασία αυτή είναι ο πολλαπλασιασμός πινάκων που εκτελείται στις MXUs. Στην ουσία οι αριθμητικές πράξεις που γίνονται στη αριθμητική και λογική μονάδα (*Arithmetic and Logical Unit - ALU*), συνδέονται μεταξύ τους, μειώνοντας έτσι τις προσβάσεις στους καταχωρητές.

- **Περιορισμένη λογική ελέγχου:**

Μεγάλο μέρος της αρχιτεκτονικής κυρίως των CPUs, αλλά και των GPUs, έχει δεσμευτεί για τη λογική ελέγχου, η οποία διαχειρίζεται τις μνήμες cache, τις προβλέψεις των διακλαδώσεων (*branch prediction*) και πολλά ακόμα. Η δομή της TPU αντιθέτως περιλαμβάνει μόνο κατά 2% λογική ελέγχου, πράγμα που δίνει και τη δυνατότητα πρόβλεψης της καθυστέρησης εκτέλεσης με ακρίβεια.

Chip	TPUv1	TPUv2	TPUv3
Announced	2016	May-17	May-18
Access	Internal-Only	Service Beta	Undisclosed
Introduction	2015	Feb 2018	Undisclosed
Process	28nm	20nm est.	16/12nm est.
Die Size	~300mm ²	Undisclosed	Undisclosed
TOPS	92 / 23	45	90
Matrix Input	INT8 / INT16	bfloat16	bfloat16
Memory	8GB DDR3	16GB HBM	32GB HBM
CPU Interface	PCIe 3.0 x16	PCIe 3.0 x8	PCIe 3.0 x8 est.
Power Consumption	40W	200-250W est.	200W est.

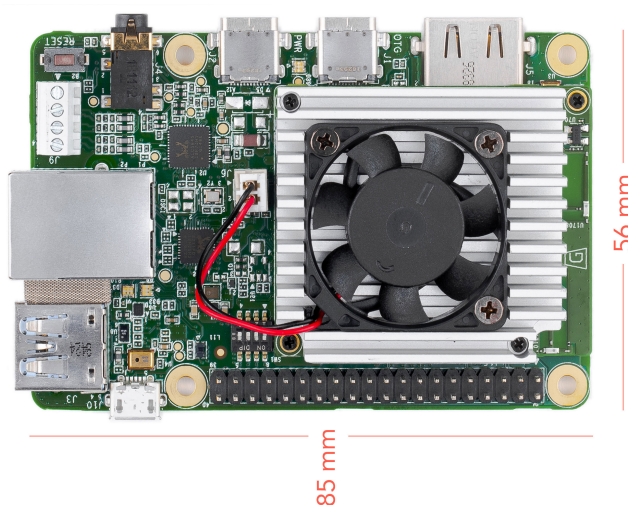
Σχήμα 4.5: Πίνακας σύγκρισης τεχνικών χαρακτηριστικών TPUs διαφορετικών εκδόσεων.

4.4.3 Coral development board

Αρχικά η χρήση TPU ήταν διαθέσιμη σαν υπηρεσία cloud, αλλά πλέον παρέχεται και σαν ανεξάρτητο board από την πλατφόρμα Coral της Google. Το board αυτό, το οποίο χρησιμοποιείται και στην παρούσα εργασία, πρόκειται στην ουσία για έναν υπολογιστή χαμηλής κατανάλωσης ο οποίος είναι εξοπλισμένος με CPU αλλά και TPU για αποδοτικότερο inference νευρωνικών δικτύων. Περιλαμβάνει λειτουργικό Linux και συγκεκριμένα ένα παράγωγο της διανομής Debian, εν ονόματι Mendel.

Το inference περιγράφει τη διαδικασία κατά την οποία ένα νευρωνικό δίκτυο χρησιμοποιεί τη γνώση που έχει λάβει από την εκπαίδευση του, για να παράξει ένα συμπέρασμα. Δηλαδή εισάγεται ένα άγνωστο σύνολο δεδομένων στο στρώμα εισόδου του δικτύου και μεταδίδεται μέχρι το στρώμα εξόδου εκτελώντας όλες τις πράξεις που καθορίζονται από την αρχιτεκτονική του δικτύου και τις τιμές των ελεύθερων παραμέτρων που έχουν προσδιοριστεί στην εκπαίδευση. Με αυτό τον τρόπο παράγεται μια πρόβλεψη για το άγνωστο δείγμα δεδομένων.

Το Edge TPU board της Coral υποστηρίζει λειτουργίες παραδοσιακών πλήρως συνδεδεμένων νευρωνικών δικτύων αλλά και συνελκτικών δικτύων. Είναι σημαντικό να σημειωθεί ότι οι μη υποστηριζόμενες λειτουργίες, όπως για παράδειγμα τα αναδρομικά δίκτυα, έχουν τη δυνατότητα να εκτελεστούν αλλά μόνο στην CPU. Επομένως το ζήτημα δεν είναι η περιορισμός επιλογών αρχιτεκτονικών δικτύων εις βάρος της δυνατότητας εκτέλεσης, αλλά εις βάρος της ταχύτητας του inference. Βέβαια για το inferencing ενός βαθιού δικτύου με μεγάλο αριθμό παραμέτρων η διαφορά χρονικής απόδοσης είναι τόσο καθοριστική που πρακτικά είναι αδύνατη η χρήση CPU.



Σχήμα 4.6: Coral edge TPU development board

Κεφάλαιο 5

Σχεδιασμός και Υλοποίηση

5.1 Γενική περιγραφή

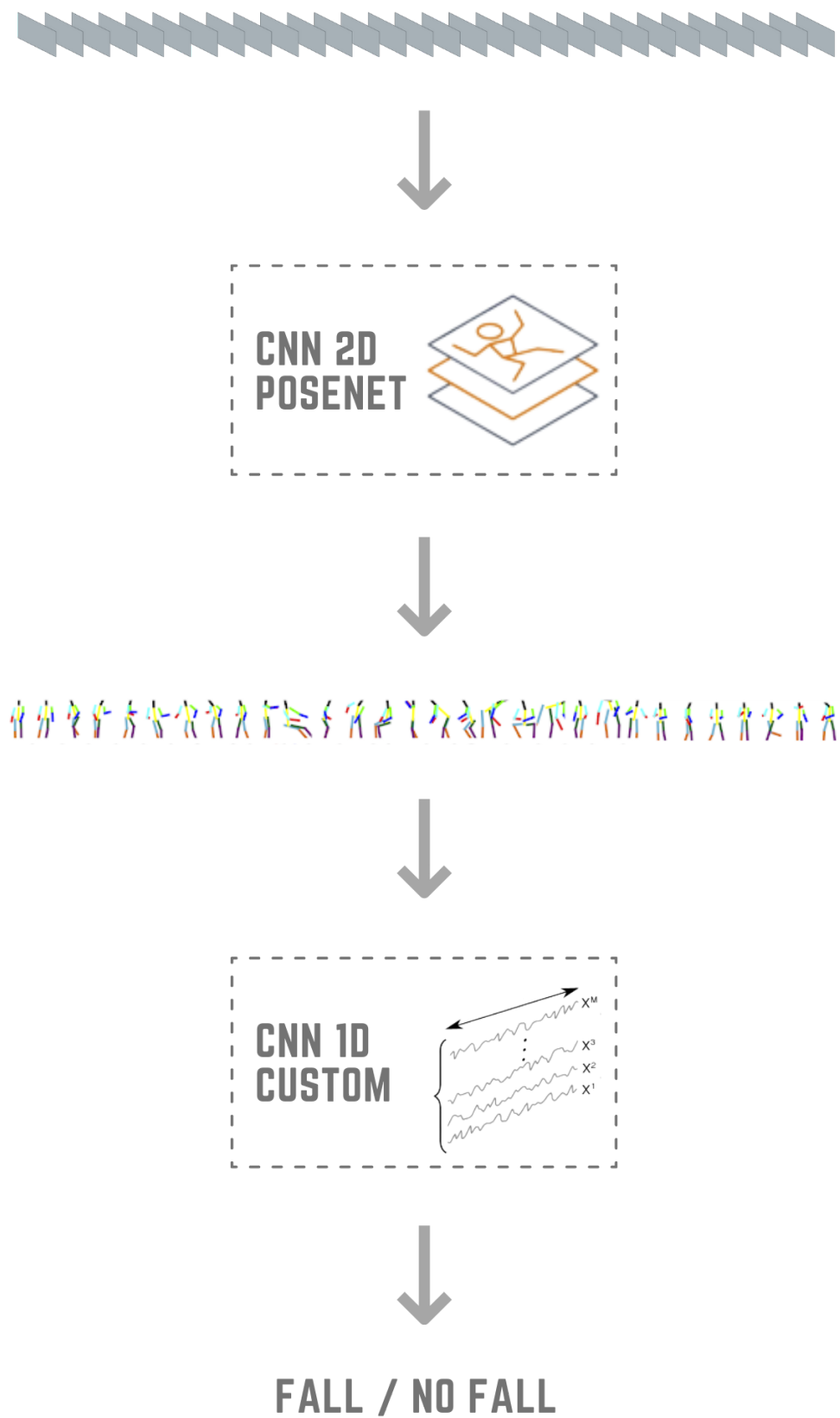
Σκοπός είναι η υλοποίηση ενός συστήματος ανίχνευσης πτώσεων μέσω οπτικής πληροφορίας πραγματικού χρόνου, η οποία συλλέγεται από μια κάμερα στο χώρο.

- **Γιατί νευρωνικά δίκτυα;**

Λόγω της φύσης του προβλήματος είναι εξαιρετικά δύσκολο να λυθεί με παραδοσιακό προγραμματισμό. Είναι αδύνατη η περιγραφή του προβλήματος από απλές συνθήκες και αλγορίθμους, καθώς η πτώση ενός ανθρώπου δεν ακολουθεί συγκεκριμένα πρότυπα. Με την προσθήκη της πρόκλησης που εμφανίζει η οπτική πληροφορία ο το πρόβλημα αυτό γίνεται ακόμα πιο απαιτητικό. Η οπτική γωνία της κάμερας, η φυσική μορφή κάθε ανθρώπου, η θέση του στο χώρο, ο φωτισμός, ο κάθε πιθανός τρόπος πτώσης και πολλά ακόμα, είναι μεταβλητές για τις οποίες θα πρέπει κανείς να ορίσει ξεχωριστές συνθήκες. Η διακλάδωση των περιπτώσεων ξεφεύγει από τα όρια ενός πρακτικά υλοποιήσιμου προγράμματος. Τα παραπάνω αιτιολογούν την επιλογή της τεχνολογίας των βαθιών νευρωνικών δικτύων και συγκεκριμένα των συνελκτικών δικτύων (κεφάλαιο 3).

- **Γιατί edge TPU;**

Το συγκεκριμένο πρόβλημα καλείται να επεξεργαστεί και να λάβει αποφάσεις πάνω σε δεδομένα πραγματικού χρόνου. Η κάμερα τροφοδοτεί συνεχώς και σε πραγματικό χρόνο το σύστημα, το οποίο πρέπει να λαμβάνει αμέσως απόφαση σχετικά με την ύπαρξη πτώσης. Συνεπώς το πρότυπο του edge computing (κεφάλαιο 4) δείχνει να είναι απαραίτητο καθώς παρέχει τη δυνατότητα της εξ' ολοκλήρου λειτουργίας του συστήματος τοπικά, δηλαδή στο χώρο στον οποίο συλλέγονται τα δεδομένα. Όπως έχει ήδη αναφερθεί, μια τέτοια λύση προσφέρει το Coral development board (κεφάλαιο 4.4.3) που σε συνδυασμό με τον βοηθητικό επεξεργαστή TPU είναι ικανό να διαχειριστεί βαθιά νευρωνικά δίκτυα χωρίς χρονικές καθυστερήσεις. Επιπροσθέτως η ανεξαρτησία που χαρακτηρίζει το edge βοηθά στη συγκεκριμένη περίπτωση αφού ένα τέτοιο σύστημα προορίζεται κυρίως για ανθρώπους της τρίτης ηλικίας, οι οποίοι είναι πιθανό να μη διαθέτουν ενεργή σύνδεση στο διαδίκτυο.



Σχήμα 4.7: Αναπαράσταση της λειτουργικής ροής του συστήματος.

Η λειτουργία και η δομή του συστήματος συνοψίζεται στο παραπάνω σχήμα. Τα παραλληλόγραμμα με διακεκομμένες γραμμές συμβολίζουν ένα νευρωνικό δίκτυο. Αρχικά χωρίζεται το βίντεο σε σειρές από εικόνες (*frames*) τα οποία τροφοδοτούνται σε ένα δίκτυο το οποίο εξάγει τις συντεταγμένες των βασικών σημείων του ανθρώπινου σκελετού για κάθε *frame*. Στη συνέχεια τα δεδομένα αυτά παρουσιάζονται σε ένα άλλο νευρωνικό δίκτυο το οποίο τελικά λαμβάνει την απόφαση για το ενδεχόμενο ύπαρξης πτώσης. Επομένως το σύστημα πραγματεύεται ένα πρόβλημα δυαδικής κατηγοριοποίησης, με την πρώτη κατηγορία να περιγράφει τη μη ύπαρξη πτώσης, ενώ τη δεύτερη την ύπαρξη αυτής.

5.2 Συλλογή δεδομένων

Για την εκπαίδευση ενός νευρωνικού δικτύου είναι απαραίτητη η συλλογή δεδομένων άμεσα σχετικών με το εν λόγω πρόβλημα που καλείται να λυθεί. Για τις ανάγκες του συγκεκριμένου προβλήματος συλλέχθηκε ένα πλήθος από βίντεο, στα οποία απεικονίζονται ανθρώπινες πτώσεις ή δραστηριότητες της καθημερινότητας. Τα σύνολα δεδομένων που χρησιμοποιήθηκαν είναι τα εξής:

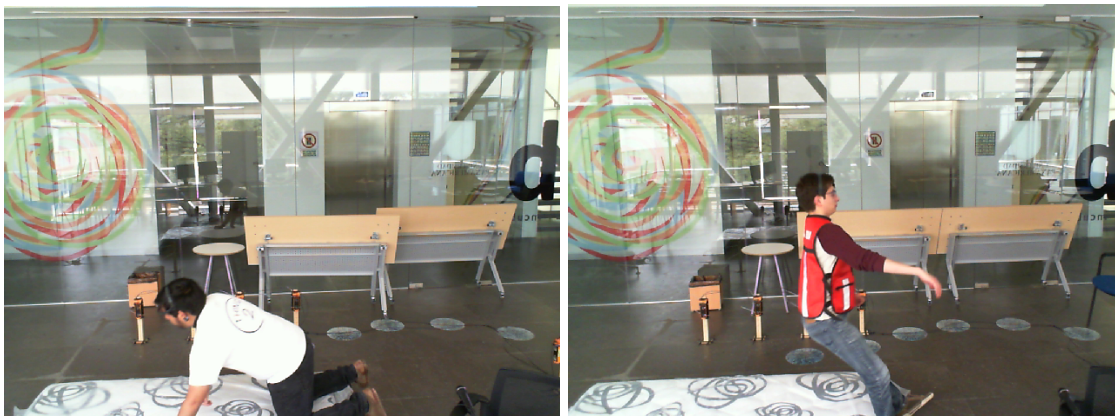
- **UP-Fall Dataset [1] :**

Περιλαμβάνει δεκαεπτά διαφορετικούς ανθρώπους οι οποίοι εκτελούν έντεκα κατηγορίες δραστηριοτήτων. Οι κατηγορίες αυτές φαίνονται παρακάτω.

Δείκτης	Κατηγορία
1-5	Πτώση
6	Περπάτημα σε κύκλους
7	Όρθια στάση
8	Κάθισμα σε καρέκλα
9	Όρθια στάση
10	Επιτόπιο άλμα
11	Ξαπλωμένη στάση

Από όλους τους διαθέσιμους τύπους δεδομένων που παράγουν διάφοροι αισθητήρες (επιταχυνσιόμετρο, υπέρυθρες κ.ά.) χρησιμοποιήθηκαν μόνο τα δισδιάστατα βίντεο RGB.

Παρατίθενται ορισμένες εικόνες σαν παράδειγμα.



Εικόνες 5.1 και 5.2: Εικόνες του συνόλου δεδομένων UP-Fall.

- **FDD - Fall Detection Dataset [2] :**

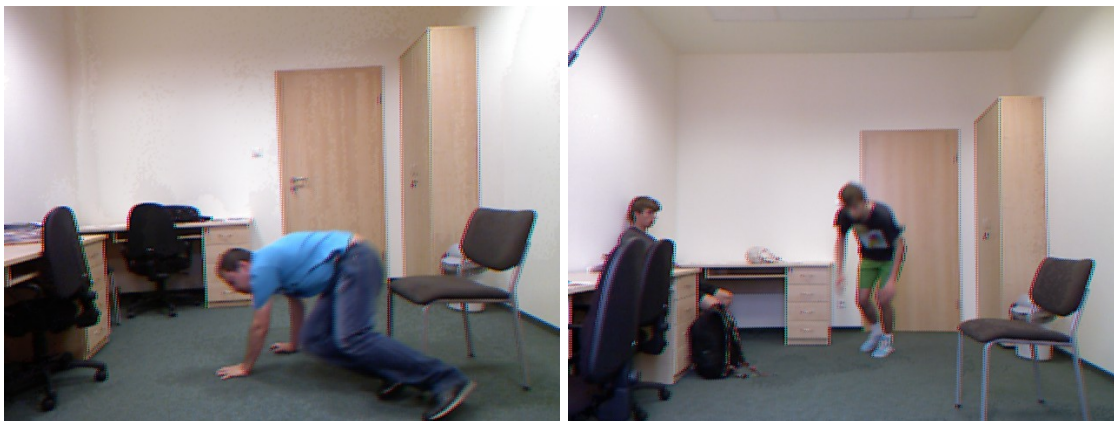
Αποτελείται συνολικά από 250 βίντεο εκ των οποίων τα 192 απεικονίζουν πτώσεις ενώ τα υπόλοιπα 57 δραστηριότητες της καθημερινότητας. Οι λήψεις έχουν γίνει σε τέσσερις διαφορετικούς εσωτερικούς χώρους. Οι ακολουθίες βίντεο περιέχουν μεταβλητό φωτισμό καθώς και σκιές και αντανακλάσεις, πράγμα που καλύπτει περισσότερες περιπτώσεις οπτικής πληροφορίας και βοηθά στην καλύτερη γενίκευση που πετυχαίνει το δίκτυο. Παρατίθεται ορισμένες εικόνες σαν παράδειγμα.



Εικόνες 5.3 και 5.4: Εικόνες του συνόλου δεδομένων FDD.

- **UR Fall Detection Dataset [3] :**

Περιλαμβάνει 40 βίντεο πτώσεων και 60 βίντεο δραστηριοτήτων της καθημερινότητας. Ομοίως με τα υπόλοιπα σύνολα δεδομένων, χρησιμοποιήθηκαν μόνο τα βίντεο RGB και όχι τα δεδομένα αισθητήρων, καθώς το σύστημα που μελετάται πραγματεύεται μόνο οπτική πληροφορία από κάμερα RGB. Τα περισσότερα βίντεο διαδραματίζονται στον ίδιο χώρο αλλά οι πτώσεις και οι δράστες μεταβάλλονται. Παρατίθεται κάποιες εικόνες σαν παράδειγμα.



Εικόνες 5.5 και 5.6: Εικόνες του συνόλου δεδομένων UR Fall.

5.3 Δίκτυο PoseNet

Το PoseNet είναι ένα συνελικτικό νευρωνικό δίκτυο δύο διαστάσεων (κεφάλαιο 3) που έχει αναπτυχθεί και εκπαιδευτεί από την Google, το οποίο αναλαμβάνει την εκτίμηση πόζας (*pose estimation*) μιας ανθρώπινης φιγούρας σε μια εικόνα. Ο όρος αυτός αναφέρεται στον προσδιορισμό της θέσης των βασικών σημείων του σκελετού, δηλαδή για παράδειγμα μιας άρθρωσης, όπως ο αγκώνας. Ο ρόλος του δικτύου στην υλοποίηση αυτή είναι η μετατροπή του βίντεο, δηλαδή της χρονοσειράς από εικόνες, σε μια αντίστοιχη χρονοσειρά από φιγούρες του σώματος. Υπό μία έννοια δηλαδή αναλαμβάνει το κομμάτι της εξαγωγής χαρακτηριστικών.

Ακολουθούν κάποιες σημαντικές έννοιες άμεσα σχετιζόμενες με το δίκτυο PoseNet.

- **Φιγούρα (*pose*):**

Το PoseNet δέχεται στην είσοδο του μια εικόνα RGB και επιστρέφει μια φιγούρα η οποία ορίζεται μέσω των σημείων-κλειδιών (*keypoints*) που την απαρτίζουν. Η φιγούρα αυτή αντιστοιχεί στον σκελετό του ανθρώπου που απεικονίζεται στην εικόνα εισόδου.

- **Σημείο-κλειδί** (*keypoint*):

Ένα μέρος της φιγούρας ενός απεικονιζόμενου ατόμου, όπως η μύτη, το δεξί αυτί, το αριστερό γόνατο, το δεξί πόδι, κ.λπ. Το PoseNet ανιχνεύει αυτήν τη στιγμή 17 σημεία-κλειδιά που απεικονίζονται στο παρακάτω σχήμα. Είναι σημαντικό να σημειωθεί ότι στη συγκεκριμένη υλοποίηση, χρησιμοποιήθηκαν μόνο τα σημεία-κλειδιά του κορμού του σώματος και όχι των άκρων. Συγκεκριμένα επιλέχθηκαν η μύτη, οι δύο ώμοι, οι δύο αγκώνες, το δεξί και αριστερό ισχίο καθώς και τα δύο γόνατα. Οι δοκιμές έδειξαν ότι τα υπόλοιπα σημεία είτε δε συνεισφέρουν σε βελτίωση της ακρίβειας, είτε εμφανίζουν ελαφρώς χειρότερο αποτελέσματα, όσον αφορά τον εντοπισμό πτώσης.



Σχήμα 5.7: Αναπαράσταση των σημείων-κλειδιών μιας φιγούρας του PoseNet

- **Θέση σημείου-κλειδιού** (*keypoint position*): Ένα διάνυσμα θέσης δύο διαστάσεων με τις συντεταγμένες (x, y) που εκτιμήθηκαν από το μοντέλο, οι οποίες αντιστοιχούν στη θέση του σημείου κλειδιού πάνω στην αρχική εικόνα εισόδου. Η αριστερά κάτω γωνία της εικόνας αντιστοιχεί στο σημείο με συντεταγμένες $(0, 0)$ και η δεξιά πάνω γωνία στο σημείο με συντεταγμένες (W, H) , όπου W είναι το πλάτος της εικόνας και H το ύψος.
- **Βαθμός εμπιστοσύνης σημείου-κλειδιού** (*keypoint confidence score*): Αυτό ορίζει ένα μέτρο βεβαιότητας για την ακρίβεια της θέσης του σημείου-κλειδιού που εκτιμήθηκε. Παίρνει τιμές στο διάστημα $[0, 1]$ με το μηδέν και το ένα να σηματοδοτούν μικρή και μεγάλη βεβαιότητα αντίστοιχα.

Το δίκτυο PoseNet έχει δύο εκδόσεις, η μία έχει ως βάση το συνελκτικό δίκτυο MobileNet ενώ η δεύτερη το ResNet. Τα μοντέλα αυτά είναι μοντέλα γενικής χρήσης για εφαρμογές υπολογιστικής όρασης (*computer vision*) καθώς έχουν εκπαιδευτεί σε τεράστιο πλήθος από εικόνες. Χρησιμοποιούνται για την εξαγωγή απλών χαρακτηριστικών και προστίθενται νέα στρώματα τα οποία αναλαμβάνουν τον εντοπισμό χαρακτηριστικών υψηλότερης τάξης και άμεσα σχετικών με το πρόβλημα. Το MobileNet έχει σχεδιαστεί με στόχο το inference σε φορητές συσκευές που διαθέτουν περιορισμένη υπολογιστική ισχύ και άρα υπερέρχει εναντίον του ResNet όσον αφορά το χρόνο πρόβλεψης, αλλά υστερεί στην ακρίβεια της πρόβλεψης.

5.4 Προ-επεξεργασία δεδομένων

Στις περισσότερες εφαρμογές νευρωνικών δικτύων η χρήση των ακατέργαστων δεδομένων (*raw data*), δηλαδή η αρχική μορφή τους όπως αυτά έχουν συλλεχθεί, δεν οδηγεί σε ικανοποιητικά αποτελέσματα και δυσχεραίνει τη διαδικασία μάθησης. Επομένως εκτελείται μια διαδικασία προ-επεξεργασίας των δεδομένων η οποία είναι όμοια για τη μέθοδο της εκπαίδευσης αλλά και για την πραγματική λειτουργία πρόβλεψης (*inferencing*). Εξαιρεση είναι μόνο τα βήματα 5.4.5 και 5.4.6 τα οποία εφαρμόζονται μόνο κατά τη διαδικασία της εκπαίδευσης του δικτύου. Το πρώτο βήμα της λειτουργίας του δικτύου είναι η μετατροπή της σειράς εικόνων (*frames*), σε σειρά από φιγούρες (*poses*). Αυτό επιτυγχάνεται με inference του δικτύου PoseNet (κεφάλαιο 5.3). Επομένως τα δεδομένα του προβλήματος είναι πλέον οι φιγούρες που επιστρέφει το PoseNet και οι θέσεις των σημείων που περιλαμβάνουν. Πάνω σε αυτό τον τύπο δεδομένων εφαρμόζεται η διαδικασία της προ-επεξεργασίας, η οποία περιγράφεται από τα βήματα που ακολουθούν.

5.4.1 Φιλτράρισμα

Η εξαγωγή της φιγούρας σε μια εικόνα δεν είναι πάντα ακριβής, καθώς μπορεί κάποιο αντικείμενο να κρύβει μέρος του σώματος του ανθρώπου ή ακόμα και να μην υπάρχει καθόλου άνθρωπος στην εικόνα. Η αναζήτηση πτώσης σε περίπτωση λανθασμένης εκτίμησης της φιγούρας ή μη ύπαρξης αυτής, είναι περιττή και επιβαρύνει το σύστημα σε όρους χρονικούς και ενεργειακούς. Επομένως εφαρμόζονται δύο στάδια φιλτραρίσματος.

- **Φιλτράρισμα των σημείων-κλειδιών.**

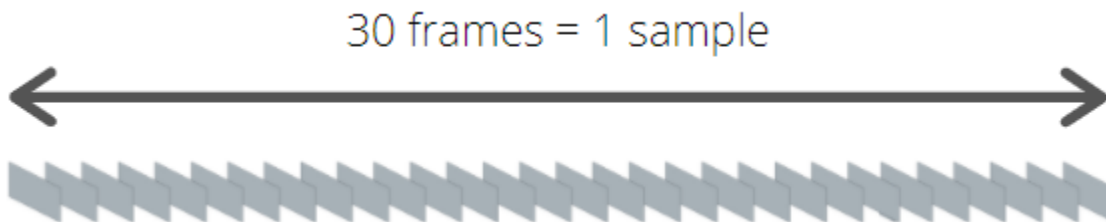
Αν σε μια εικόνα (*frame*) υπάρχουν σημεία-κλειδιά με βαθμό εμπιστοσύνης (*confidence score*) μικρότερο του 0.5, τότε θέτουμε τις συντεταγμένες αυτών των σημείων σε $(0, 0)$. Κατά την εκπαίδευση, το δίκτυο μας μαθαίνει ότι η τιμή μηδέν δεν επηρεάζει το αποτέλεσμα της απόφασης. Η παράμετρος που προκύπτει από αυτό το βήμα είναι το *confidence threshold*.

- **Φιλτράρισμα των frames.**

Αν ένα frame δεν έχει παρουσία ανθρώπου, άρα δεν υπάρχει φιγούρα, τότε δε χρησιμοποιείται καθόλου. Αν επίσης σε κάποια φιγούρα προκύψουν, μέσω του φιλτραρίσματος σημείων-κλειδιών, λιγότερα από δύο μη μηδενικά σημεία-κλειδιά, τότε δε χρησιμοποιείται το frame. Η παράμετρος που προκύπτει είναι το *points threshold*.

5.4.2 Διάσπαση και ομαδοποίηση

Κάθε βίντεο διασπάται σε εικόνες (*frames*), οι οποίες μετά ομαδοποιούνται ανά τριάντα. Επειδή μια πτώση διαδραματίζεται σε σχέση με το χρόνο, η μεμονωμένη εξέταση κάθε εικόνας είναι λιγότερο βοηθητική από την εξέταση μιας σειράς από εικόνες. Επομένως στο παρόν πρόβλημα το κάθε δείγμα (*sample*) συνιστά μια σειρά από διαδοχικές εικόνες, το μέγεθος της οποίας (δηλαδή το πλήθος των frames) ορίστηκε, μέσω δοκιμών, ίσο με 30. Η παράμετρος αυτή ονομάζεται *timesteps*. Πρακτικά το δίκτυο εξετάζει ξεχωριστά κάθε δείγμα και εκτιμά αν περιέχει πτώση.



Σχήμα 5.8: Μέγεθος ενός δείγματος σε frames.

5.4.3 Κανονικοποίηση

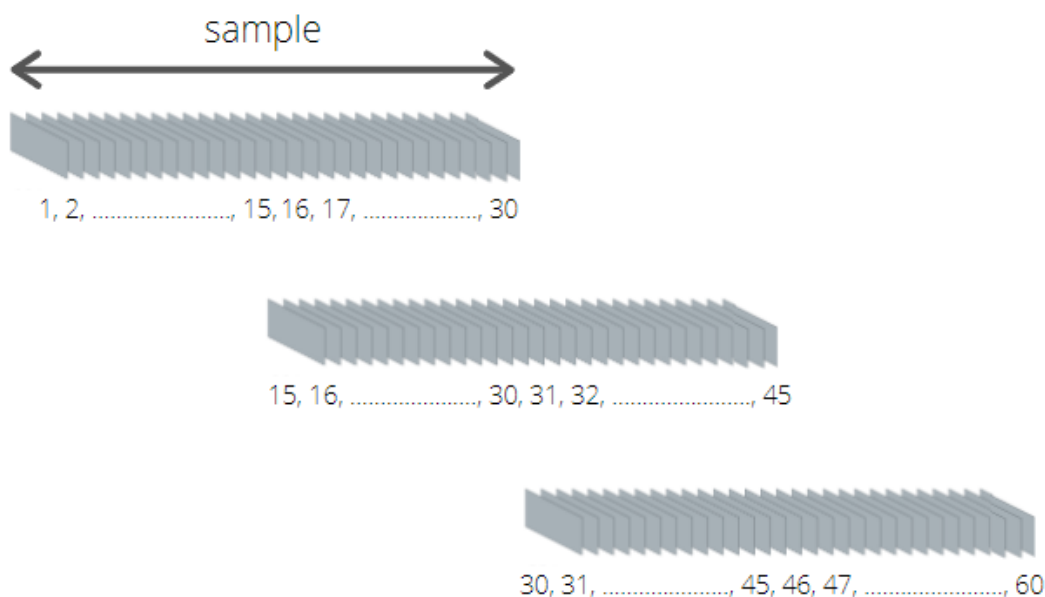
Επειδή είναι απαραίτητο το σύστημα να λειτουργεί για διαφορετικές διαστάσεις εικόνας εισόδου, γίνεται κανονικοποίηση των συντεταγμένων θέσης των κλειδιών-σημείων (*keypoints*). Συγκεκριμένα γίνεται κανονικοποίηση στο διάστημα $[0, 1]$ διαιρώντας τη συντεταγμένη του άξονα x με το πλάτος της εικόνας και τη συντεταγμένη του άξονα y με το ύψος της. Αυτό διευκολύνει την εκπαίδευση του δικτύου αλλά δίνει και τη δυνατότητα χρήσης διαφορετικών διαστάσεων για κάθε είσοδο.

5.4.4 Παράθυρο ολίσθησης

Λόγω της διάσπασης της χρονοσειράς των εικόνων σε ξεχωριστές ομάδες, εισάγεται ο κίνδυνος να διαχωριστεί κάποια δραστηριότητα σε διαφορετικά δείγματα. Για παράδειγμα θα μπορούσε μια πτώση που καταλαμβάνει 15 frames, να διασπαστεί στη μέση και τελικά να καταστεί αδύνατος ο εντοπισμός της από το δίκτυο. Για το λόγο αυτό ο διαχωρισμός των frames σε δείγματα (*samples*) γίνεται με τεχνική παραθύρου ολίσθησης με αποτέλεσμα τα δείγματα να εμφανίζουν επικάλυψη μεταξύ τους. Μετά από δοκιμές η επικάλυψη επιλέχθηκε ως το μισό του μήκους δείγματος. Άρα η παράμετρος που προκύπτει εδώ ονομάζεται *overlap*.

Αν έχουμε για παράδειγμα 60 frames, έστω fr_k με $k \in [1, 60]$, τότε θα έχουμε 3 διαφορετικά samples S_1 , S_2 και S_3 με :

$$S_1 = fr_1, fr_2, \dots, fr_{30}, \quad S_2 = fr_{15}, fr_{16}, \dots, fr_{45}, \quad S_3 = fr_{30}, fr_{31}, \dots, fr_{60}.$$



Σχήμα 5.9: Παράδειγμα διαχωρισμού 60 frames σε 3 δείγματα.

5.4.5 Χαρακτηρισμός δειγμάτων

Η μέθοδος μάθησης που αρμόζει στο παρόν πρόβλημα είναι η μάθηση με επίβλεψη (κεφάλαιο 2.5). Επομένως θα παρουσιαστεί στο δίκτυο μια σειρά από δείγματα μαζί με την επιθυμητή πρόβλεψη που θα πρέπει να κάνει το δίκτυο. Με απλά λόγια κάθε δείγμα εκπαίδευσης θα πρέπει να συνοδεύεται από μια ετικέτα (*label*), η οποία ορίζει τη σωστή κατηγορία στην οποία ανήκει. Δηλαδή, σε αυτό το βήμα κάθε δείγμα αντιστοιχίζεται είτε στην κλάση μη πτώσης, είτε στην κλάση πτώσης. Αυτές οι δύο κλάσεις συμβολίζονται με το δυαδικό 0 και 1, αντίστοιχα. Πρακτικά προστίθεται μια ετικέτα σε κάθε εικόνα (*frame*) η οποία συμβολίζει την κλάση στην οποία ανήκει. Η αντιστοίχιση όμως των δειγμάτων γίνεται σύμφωνα με το πλήθος των εικόνων που φέρουν την ετικέτα της κατηγορίας πτώσης. Επομένως εδώ εισάγεται η παράμετρος *label threshold*. Μια μικρή τιμή αυτής της παραμέτρου αυξάνει τον κίνδυνο να κατηγοριοποιηθούν δείγματα μη πτώσεων ως πτώσεις (*false positives*), ενώ αντιθέτως μια μεγάλη τιμή αυξάνει την πιθανότητα να μην εντοπιστεί κάποιο δείγμα πτώσης (*false negative*).

5.4.6 Επαύξηση

Ένα νευρωνικό δίκτυο χρειάζεται έναν τεράστιο αριθμό από παραδείγματα κατά την εκπαίδευση. Στις περισσότερες περιπτώσεις όσο περισσότερα παραδείγματα τροφοδοτηθούν στο δίκτυο τόσο αυξάνεται η ικανότητα γενίκευσης σε άγνωστα δεδομένα. Το συγκεκριμένο πρόβλημα χαρακτηρίζεται από μεγάλη ανισορροπία δεδομένων (*unbalanced data*). Πιο συγκεκριμένα τα δείγματα πτώσεων είναι πολύ λιγότερα από τα δείγματα μη πτώσεων, πράγμα λογικό καθώς για παράδειγμα σε ένα βίντεο μήκους 500 frames, η πτώση καταλαμβάνει συνήθως 20 έως 40 frames μόνο. Αν προσθέσουμε σε αυτό το γεγονός ότι είναι επίσης χρήσιμα βίντεο τα οποία δεν περιέχουν καθόλου πτώσεις (κάθισμα σε καρέκλα, επίκληση σε κρεβάτι κ.λπ.) αυτή η ανισορροπία των δεδομένων αυξάνεται.

Αυτό προκαλεί δυσκολία στην εκπαίδευση του δικτύου καθώς αυτή έχει σχεδιαστεί με στόχο την επίτευξη συνολικής ακρίβειας πρόβλεψης. Αν για παράδειγμα μόνο το 10% του συνόλου δεδομένων εκπαίδευσης αποτελείται από δείγματα πτώσεων, τότε το νευρωνικό δίκτυο κάνοντας τυχαίες προβλέψεις πετυχαίνει ήδη 90% ακρίβεια. Άρα τα περιθώρια μάθησης είναι περιορισμένα και το δίκτυο δεν θα καταφέρει να εντοπίσει το μεγαλύτερο μέρος των πτώσεων σε άγνωστα δεδομένα.

Για τον παραπάνω λόγο χρησιμοποιείται τεχνική επαύξησης δεδομένων (*data augmentation*) με σκοπό την διάθεση μεγαλύτερου πλήθους δειγμάτων πτώσεων για την εκπαίδευση του δικτύου. Η τεχνική αυτή εφαρμόζεται για κάθε δείγμα πτώσης ξεχωριστά και περιγράφεται από τα παρακάτω βήματα.

1. Υπολογισμός της ελάχιστης και μέγιστης τιμής της συντεταγμένης του x για ολόκληρο το δείγμα. Τοποθετούνται σε μια λίστα οι συντεταγμένες του άξονα x όλως των σημείων-κλειδιών για όλα τις φιγούρες που ορίζουν τα frames ενός δείγματος. Έπειτα υπολογίζεται η ελάχιστη και μέγιστη τιμή αυτών, έστω min_x και max_x .
2. Παραγωγή τυχαίων τιμών στα διανύσματα που ορίζουν οι τιμές που υπολογίστηκαν στο προηγούμενο βήμα με τα όρια του εύρους. Τα όριο του εύρους τιμών των συντεταγμένων που ορίζει η κανονικοποίηση (κεφάλαιο 5.4.3) είναι το 0 και το 1. Άρα τα δύο διαστήματα που προκύπτουν είναι τα: $(0, min_x)$ και $(max_x, 1)$. Παράγονται λοιπόν δύο τυχαίες τιμές που ανήκουν σε αυτά τα διαστήματα, έστω R_{1x} και R_{2x} .
3. Η τιμή R_{1x} αφαιρείται από τις συντεταγμένες x κάθε φιγούρας του δείγματος, πράγματα που δημιουργεί ένα καινούργιο δείγμα πτώσης μετατοπισμένο προς τα αριστερά στον άξονα των x . Αντίστοιχα η τιμή R_{2x} προστίθεται στις συντεταγμένες x του δείγματος και παράγεται ένα νέο δείγμα πτώσης μετατοπισμένο προς τα δεξιά.

Πρακτικά η θέση στο χώρο που γίνεται η πτώση δε συνιστά χρήσιμη πληροφορία για τον εντοπισμό πτώσης. Η μεταβολές της θέσης των σημείων-κλειδιών στον κάθετο άξονα είναι που έχουν μεγαλύτερη βαρύτητα στο συγκεκριμένο πρόβλημα. Αυτό είναι που μας επιτρέπει την εφαρμογή της παραπάνω μέθοδο επαύξησης δεδομένων.

5.4.6 Τελικές παράμετροι

Η διαδικασία της προ-επεξεργασίας δεδομένων που περιγράφηκε εισάγει κάποιες παραμέτρους στο πρόβλημα. Η αλλαγή της τιμής αυτών των παραμέτρων επηρεάζουν σημαντικά την ικανότητα μάθησης του δικτύου αλλά και την τελική απόδοση του.

Μετά από σειρά δοκιμών για διάφορους συνδυασμούς τιμών, επιλέχθηκαν αυτές που φαίνονται στον παρακάτω πίνακα.

ΠΑΡΑΜΕΤΡΟΣ	ΤΙΜΗ
Confidence Threshold Κατώτατο όριο βαθμού εμπιστοσύνης θέσης σημείου-κλειδιού.	0.5 /1
Points Threshold Κατώτατο όριο πλήθους μη μηδενικών σημείων-κλειδιών.	2 /9
Timesteps Πλήθος εικόνων (frames) που συνιστούν ένα δείγμα (sample).	30
Overlap Πλήθος εικόνων (frames) που παρουσιάζονται ως επικάλυψη σε δύο διαδοχικά δείγματα (samples).	15 /30
Label Threshold Κατώτατο όριο πλήθους εικόνων (frames) με ετικέτα πτώσης για να χαρακτηριστεί ένα δείγμα (sample) ως πτώση.	9 /30

Σχήμα 5.10: Πίνακας παραμέτρων προ-επεξεργασίας δεδομένων.

5.5 Νευρωνικό δίκτυο πτώσης

Μετά την προεπεξεργασία των δεδομένων, αυτά τροφοδοτούνται σε ένα νευρωνικό δίκτυο το οποίο και αναλαμβάνει την τελική εκτίμηση για το ενδεχόμενο ύπαρξης πτώσης. Δεδομένου ότι το δίκτυο καλείται να διαχειριστεί ένα πρόβλημα χρονοσειράς συνηθίζεται η χρήση αναδρομικού δικτύου (κεφάλαιο 2.4.3). Επειδή όμως το inference αναδρομικών δικτύων δεν υποστηρίζεται από το Edge TPU της Coral (κεφάλαιο 4.4.3), επιλέγεται η χρήση συνελκτικού δικτύου μιας διάστασης (κεφάλαιο 3.3). Η χρήση αναδρομικού δικτύου θα ισοδυναμούσε με inference στην CPU του board, με αποτέλεσμα την εισαγωγή χρονικής καθυστέρησης στο σύστημα.

Αρχικά θα γίνει αναφορά στην αρχιτεκτονική και την επιλογή των παραμέτρων του δικτύου, δηλαδή το πλήθος και τον τύπο των επιπέδων και τις παραμέτρους που σχετίζονται με αυτά. Έπειτα θα παρουσιαστεί η διαδικασία που ακολουθήθηκε για την εκπαίδευση του δικτύου.

5.5.1 Αρχιτεκτονική μοντέλου

Το μοντέλου που επιλέχθηκε για την κατηγοριοποίηση των δειγμάτων, είναι ένα συνελικτικό δίκτυο μιας διάστασης (κεφάλαιο 3.3). Για την επιλογή του πλήθους των επιπέδων αλλά και των παραμέτρων αυτών δεν υπάρχει κάποιο ντετερμινιστικό σύνολο κανόνων επιλογής. Πρακτικά με βάση τη φύση του προβλήματος προσδιορίζεται μια αρχική εκδοχή του μοντέλου η οποία εκπαιδεύεται και έπειτα επαληθεύεται η απόδοση της. Η επαλήθευση γίνεται με βάση μέτρα απόδοσης όπως η ακρίβεια πρόβλεψης και ο ρόλος της είναι η σωστή ρύθμιση των παραμέτρων του δικτύου. Επομένως μετά από πλήθος δοκιμών η τελική μορφή του δικτύου περιλαμβάνει τα επίπεδα που ακολουθούν.

1. Επίπεδο εισόδου.

Πρόκειται για το επίπεδο μέσω του οποίου τροφοδοτείται το δείγμα εισόδου στο σύστημα. Οι παράμετροι που δέχεται είναι οι παρακάτω.

- **Timesteps:** Όπως εξηγήθηκε και στο κεφάλαιο 5.4.6 πρόκειται για το μήκος του δείγματος, δηλαδή το πλήθος των χρονικών βημάτων που συνιστούν ένα δείγμα εισόδου.
- **Features:** Πλήθος των χαρακτηριστικών των δειγμάτων εισόδου. Είναι ίσο με το διπλάσιο του πλήθους των σημείων-κλειδιών της φιγούρας.

ΠΑΡΑΜΕΤΡΟΣ	ΤΙΜΗ
Timesteps	30
Features	18

2. Συνελικτικό επίπεδο.

Η λειτουργία καθώς και οι παράμετροί του έχουν μελετηθεί στο κεφάλαιο 3.2.2 και συνεπώς παρατίθεται απλώς ο πίνακας με τις τιμές που επιλέχθηκαν.

ΠΑΡΑΜΕΤΡΟΣ	ΤΙΜΗ
Filters	128
Kernel Size	6
Stride	1
Activation	ReLU

3. Επίπεδο συγκέντρωσης.

Το επίπεδο αυτό περιγράφεται στο κεφάλαιο 3.2.3 και επομένως δίνεται μόνο ο πίνακας των τιμών των παραμέτρων του.

ΠΑΡΑΜΕΤΡΟΣ	ΤΙΜΗ
Type	Max Pooling
Pool Size	2
Stride	1

4. Πλήρως συνδεδεμένο επίπεδο.

Οι παράμετροι που εισάγει το πλήρως συνδεδεμένο επίπεδο είναι ο αριθμός των νευρώνων καθώς και η συνάρτηση ενεργοποίησης. Είναι σημαντικό να σημειωθεί ότι στη συγκεκριμένη υλοποίηση επιλέχθηκε η συνοδεία αυτού του επιπέδου με έναν επίπεδο διακοπής (*dropout layer*). Αυτό μηδενίζει, επιλέγοντας τυχαία, κάποια βάρη του πλήρους συνδεδεμένου επιπέδου και βοηθά στην αποφυγή της υπερ-εκπαίδευσης (*overfitting*), το οποίο εξηγείται στην δεύτερη παράγραφο του κεφαλαίου 3.2.3. Η παράμετρος που ορίζει το επίπεδο διακοπής είναι το ποσοστό των βαρών που θα μηδενιστούν.

ΠΑΡΑΜΕΤΡΟΣ	ΤΙΜΗ
Units	128
Activation	ReLU
Dropout Rate	0.5

Όσον αφορά το πλήθος των επιπέδων, το δίκτυο πέτυχε ακριβής προβλέψεις με χρήση δύο διαδοχικών ζευγών συνελκτικών και συγκεντρωτικών επιπέδων και ένα μόνο πλήρως συνδεδεμένο επίπεδο συνοδευόμενο από το επίπεδο διακοπής.

Το συνολικό πλήθος ελεύθερων παραμέτρων εκπαίδευσης του νευρωνικού δικτύου είναι τελικά ίσο με 227841.

5.5.2 Εκπαίδευση

Οι παράμετροι που σχετίζονται με τη διαδικασία της εκπαίδευσης είναι το μέγεθος παρτίδας (*batch size*), το πλήθος των εποχών (*epochs*), η συνάρτησης κόστους (*loss function*) και ο αλγόριθμος βελτιστοποίησης (*optimizer*). Οι έννοιες αυτές έχουν αναλυθεί στο κεφάλαιο 2.6 και 2.7. Από διαθέσιμο το σύνολο δεδομένων το 80% διατέθηκε για την εκπαίδευση και το 20% για την επαλήθευση. Η επαλήθευση (*validation*) περιγράφει τη διαδικασία όπου το δίκτυο εξάγει προβλέψεις σε άγνωστα δεδομένα, δηλαδή δεδομένα με τα οποία δεν έχει εκπαιδευτεί και με βάση τα αποτελέσματα των προβλέψεων αυτών γίνεται ρύθμιση των παραμέτρων του δικτύου.

Οι επιλογές που έγιναν σχετικά με τις παραμέτρους της εκπαίδευσης συνοψίζονται στον παρακάτω πίνακα. Κατά την εκπαίδευση ορίστηκαν διαφορετικά βάρη για τις δύο κλάσεις ή κατηγορίες του προβλήματος. Συγκεκριμένα δόθηκε διπλάσιο βάρος στην κλάση της πτώσης, δηλαδή το δίκτυο, κατά την εκπαίδευση, αντιμετωπίζει κάθε δείγμα πτώσης ως δύο δείγματα μη πτώσης. Ο λόγος μιας τέτοιας επιλογής αφορά την ανισορροπία των δεδομένων όσον αφορά τις δύο αυτές κατηγορίες.

ΠΑΡΑΜΕΤΡΟΣ	ΤΙΜΗ
Batch Size	128
Epochs	100
Loss	Binary Cross-entropy
Optimizer	Adam

Είναι σημαντικό να σημειωθεί ότι η εκπαίδευση δεν γίνεται στο Edge, αλλά είτε τοπικά σε υπολογιστή με χρήση GPU είτε στο cloud. Η εκπαίδευση είναι μια διαδικασία που επιφέρει μεγάλο υπολογιστικό βάρος, επομένως δεν είναι πρακτική η εκτέλεση της σε φορητή συσκευή όπως το Edge TPU της Coral (κεφάλαιο 4.4.3). Εξάλλου η εκπαίδευση αποτελεί μέρος της προετοιμασίας του δικτύου και όχι της λειτουργίας του σε πραγματικό χρόνο. Επομένως η μεταφορά της στο Edge δεν θα προσέφερε κάποιο σημαντικό πλεονέκτημα στο συγκεκριμένο πρόβλημα.

Κεφάλαιο 6

Αποτελέσματα

6.1 Αποτελέσματα εκπαίδευσης

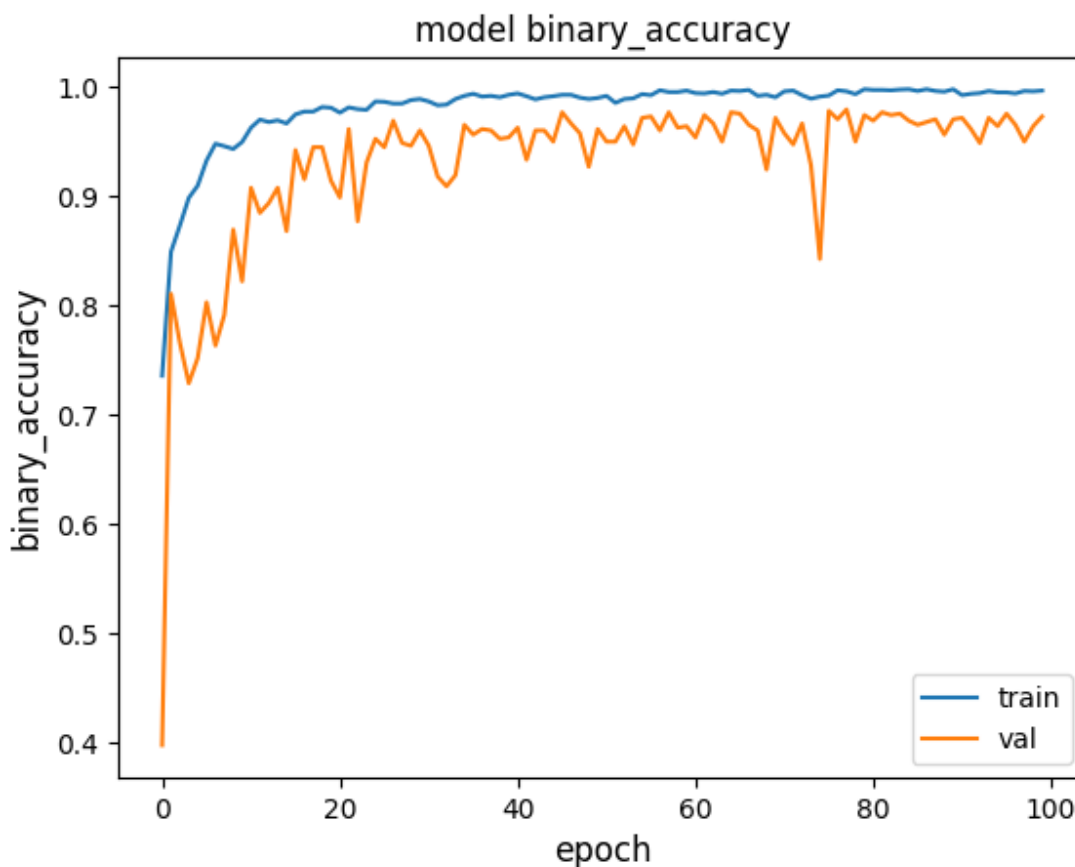
Για τις παραμέτρους και την αρχιτεκτονική του δικτύου που περιγράφηκε στο κεφάλαιο 5.5, λήφθηκαν κάποια στατιστικά αποτελέσματα. Αυτά αφορούν στην ουσία το κομμάτι της επαλήθευσης (*validation*).

ΜΕΤΑΒΛΗΤΗ	ΤΙΜΗ
# of training samples Πλήθος δειγμάτων που χρησιμοποιήθηκαν για την εκπαίδευση.	3909
# of fall samples Πλήθος δειγμάτων πτώσης που χρησιμοποιήθηκαν κατά την εκπαίδευση.	1368 /3909
# of not fall samples Πλήθος δειγμάτων μη πτώσης που χρησιμοποιήθηκαν κατά την εκπαίδευση.	2541 /3909
Validation Binary Accuracy Η ακρίβεια πρόβλεψης, σε ποσοστό, που πέτυχε το δίκτυο κατά το στάδιο της επαλήθευσης.	0,97 /1
Validation Loss Η τιμή της συνάρτησης κόστους, σε ποσοστό, κατά το στάδιο της επαλήθευσης.	0,21 /1

Σχήμα 6.1: Πίνακας αποτελεσμάτων εκπαίδευσης.

Ο αριθμός των δειγμάτων πτώσης πριν την επαύξηση των δεδομένων (κεφάλαιο 5.4.6) είναι 456. Με τη χρήση της επαύξησης παράγονται 2 δείγματα νέα δείγματα πτώσης για κάθε αρχικό, πράγμα που φαίνεται εκ του αποτελέσματος καθώς $456 \times 3 = 1368$.

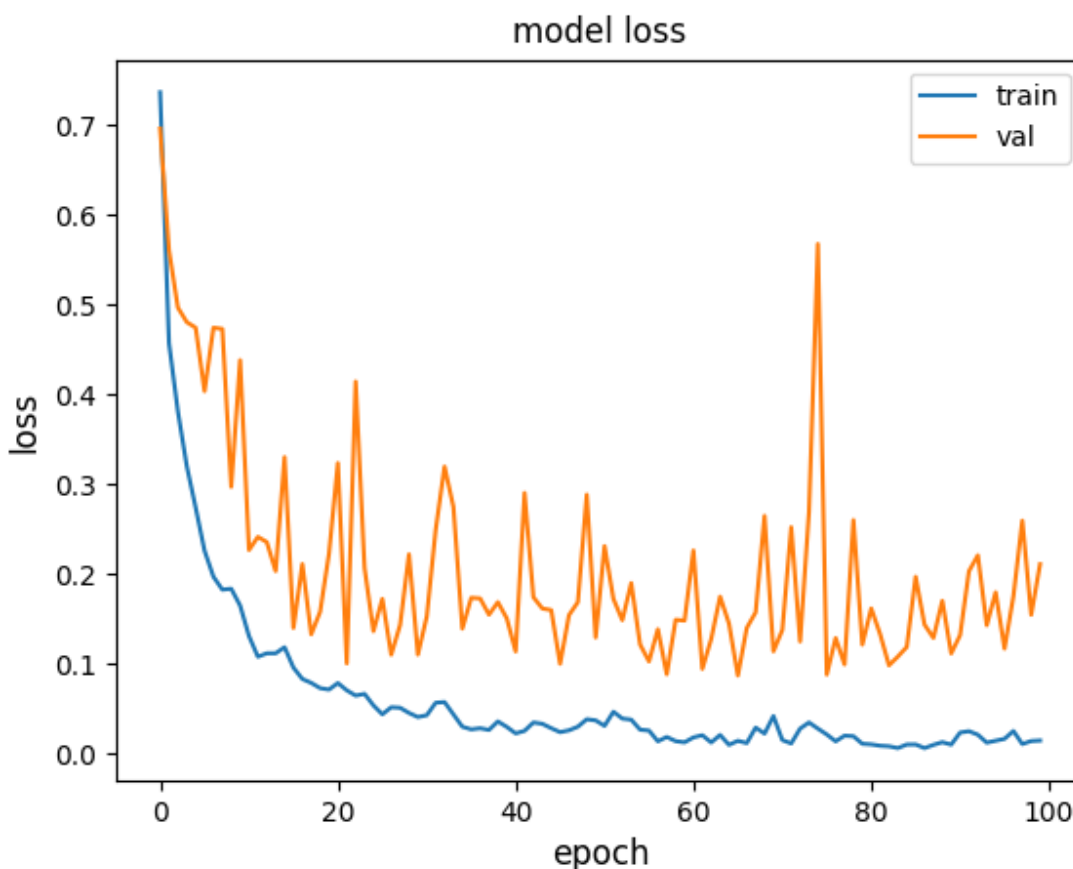
Παρακάτω παρατίθενται τα γραφήματα που παρουσιάζουν τις τιμές της ακρίβειας και του κόστους σε σχέση με τις εποχές, για το στάδιο της εκπαίδευσης αλλά και της επαλήθευσης.



Σχήμα 6.2: Γράφημα ακρίβειας πρόβλεψης σε σχέση με τις εποχές. Η πορτοκαλί γραφική παράσταση αναφέρεται στο στάδιο επαλήθευσης, ενώ η μπλε στη διαδικασία της εκπαίδευσης.

Παρατηρήσεις:

Φαίνεται ότι το δίκτυο συγκλίνει σχετικά γρήγορα, δηλαδή μέσα στις πρώτες 20 με 30 εποχές εκπαίδευσης, ενώ στις επόμενες εποχές σταθεροποιείται περισσότερο. Η απότομη πτώση στην ακρίβεια επαλήθευσης που παρατηρείται ανάμεσα στις εποχές 70 και 80 πιθανώς οφείλεται σε τοπικό ελάχιστο των βαρών, το οποίο δεν αποτελεί καλή επιλογή. Βλέπουμε όμως ότι το δίκτυο καταφέρνει να το ξεπεράσει και να σταθεροποιηθεί σε κάποιο καλύτερο σημείο ελαχίστου.



Σχήμα 6.3: Γράφημα κόστους σε σχέση με τις εποχές.
 Η πορτοκαλί γραφική παράσταση αναφέρεται στο στάδιο επαλήθευσης,
 ενώ η μπλε στη διαδικασία της εκπαίδευσης.

Παρατηρήσεις:

Η γραφική παράσταση του κόστους παρουσιάζει όμοια συμπεριφορά, αλλά αντίστροφη, καθώς περιγράφουν αντίστροφα μεγέθη. Στο ίδιο σημείο που είχαμε απότομη μείωση της ακρίβειας, παρατηρείται απότομη αύξηση του κόστους. Επίσης οι απότομες μεταβολές που φαίνεται να υπάρχουν στη γραφική παράσταση της επαλήθευσης οφείλονται στην χαμηλή σχετικά τιμή του μεγέθους παρτίδας (*batch size*), καθώς τα δεδομένα που συνιστούν κάθε παρτίδα μπορεί να μην αντιπροσωπεύουν με ακρίβεια το σύνολο των δεδομένων. Σε αυτό επιδρά αρνητικά επίσης και η ανισορροπία των κλάσεων των δεδομένων.

6.2 Αποτελέσματα κανονικής λειτουργίας

Τα αποτελέσματα της εκπαίδευσης και επαλήθευσης δεν δίνουν την ακριβή ικανότητα γενίκευσης του μοντέλου σε άγνωστα δεδομένα. Επειδή οι παράμετροι του δικτύου έχουν ρυθμιστεί με βάση αυτά τα στατιστικά στοιχεία, εισάγεται μια μορφή προκατάληψης (*bias*). Για αυτό τον λόγο χρησιμοποιείται ένα ξεχωριστό σύνολο δεδομένων, με τα οποία δεν έχει επιδράσει το δίκτυο σε κανένα στάδιο λειτουργίας του. Αυτό το σύνολο ονομάζεται *test set* και η δοκιμή πάνω σε αυτό γίνεται στο Edge TPU, καθώς εκεί καλείται να λειτουργήσει το δίκτυο στην τελική του μορφή.

ΜΕΤΑΒΛΗΤΗ	ΤΙΜΗ
# of test samples Πλήθος δειγμάτων που χρησιμοποιήθηκαν για τη δοκιμή κανονικής λειτουργίας (<i>test set</i>).	998
# of fall samples Πλήθος δειγμάτων πτώσης που χρησιμοποιήθηκαν κατά τη δοκιμή κανονικής λειτουργίας (<i>test set</i>).	140 /998
# of not fall samples Πλήθος δειγμάτων πτώσης που χρησιμοποιήθηκαν κατά τη δοκιμή κανονικής λειτουργίας (<i>test set</i>).	858 /998
Test Binary Accuracy Η ακρίβεια πρόβλεψης, σε ποσοστό, που πέτυχε το δίκτυο κατά τη δοκιμή κανονικής λειτουργίας.	0,95 /1
Test Loss Η τιμή της συνάρτησης κόστους, σε ποσοστό, κατά τη δοκιμή κανονικής λειτουργίας.	0,27 /1
False Positives Πλήθος δειγμάτων που κατηγοριοποιήθηκαν σαν πτώσεις, ενώ δεν είναι πραγματικά πτώσεις.	24 /858
False Negatives Πλήθος δειγμάτων πτώσεων που δεν εντοπίστηκαν, δηλαδή κατηγοριοποιήθηκαν σαν μη πτώσεις.	21 /140
Precision Ποσοστό των προβλέψεων πτώσης που ήταν σωστές.	0.83
Recall Ποσοστό των πραγματικών πτώσεων που εντοπιστήκαν.	0.85

Σχήμα 6.1: Πίνακας αποτελεσμάτων δοκιμής κανονικής λειτουργίας.

Παρατηρήσεις:

Παρατηρούμε μια πτώση στην ακρίβεια του μοντέλου, πράγμα λογικό καθώς αρχικά τα δεδομένα πλέον είναι εντελώς άγνωστα για το δίκτυο. Ένας δεύτερος λόγος είναι ότι η υλοποίηση στο Edge TPU χρησιμοποιεί μια ελαφρύτερη έκδοση του δικτύου PoseNet. Με τον όρο ελαφρύτερη εννοείται το μειωμένο πλήθος ελεύθερων παραμέτρων του δικτύου. Από τα μέτρα των False Positives, False Negatives, Precision και Recall συμπεραίνουμε ότι το δίκτυο σπανίως εκτιμά ως πτώση κάποιο δείγμα μη πτώσης, ενώ δυσκολεύεται περισσότερο στον εντοπισμό μιας πραγματικής πτώσης. Αυτό βέβαια είναι λογικό και πιθανώς οφείλεται στη μεγάλη ανισορροπία των δεδομένων. Η επαύξηση δεδομένων (κεφάλαιο 5.4.6) που χρησιμοποιήθηκε και μεν βοήθησε στην εξισορρόπηση των δεδομένων, αλλά στην ουσία δεν παρουσίασε στο δίκτυο νέου διαφορετικούς τύπους πτώσεων.

Με χρήση θύρας USB 2.0 για την τροφοδότηση της εισόδου στο δίκτυο, το συνολικό σύστημα λειτούργησε σε συχνότητα 21 frames ανά δευτερόλεπτο. Η τιμή αυτής της συχνότητας είναι αποδεκτή καθώς συμβαδίζει με τα σύνολα δεδομένων που χρησιμοποιήθηκαν κατά την εκπαίδευση. Εξάλλου μεγαλύτερη συχνότητα θα απαιτούσε νέα ρύθμιση των παραμέτρων του δικτύου, όπως για παράδειγμα το μέγεθος του δείγματος (*timesteps*).

Κεφάλαιο 7

Επίλογος

7.1 Σύνοψη

Για την ικανοποίηση της ανάγκης εντοπισμού πτώσης που εμφανίζεται κατά κύριο λόγο σε περιβάλλοντα με ηλικιωμένους ανθρώπους, αναπτύχθηκε το σύστημα που περιγράφηκε στα προηγούμενα κεφάλαια. Η επιλογή τεχνικών μηχανικής μάθησης και συγκεκριμένα βαθιών νευρωνικών δικτύων κρίθηκε απαραίτητη για την αντιμετώπιση του συγκεκριμένου προβλήματος. Αφού μελετήθηκε και αναλύθηκε η βασική θεωρία νευρωνικών δικτύων (κεφάλαιο 2), έγινε θεωρητική αναφορά στην αρχιτεκτονική των συνελκτικών δικτύων (κεφάλαιο 3), πράγμα που αποτελεί προϋπόθεση για την κατανόηση της λειτουργίας των μοντέλων επιλέχθηκαν για το τελικό σύστημα. Τα μοντέλα αυτά είναι το δίκτυο PoseNet (κεφάλαιο 5.3), το οποίο αναλαμβάνει την εξαγωγή της θέσης των βασικών σημείων του σκελετού ενός ανθρώπου και ένα συνελκτικό δίκτυο μιας διάστασης, το οποίο καλείται να διεξάγει την τελική εκτίμηση για τον εντοπισμό μιας πτώσης (κεφάλαιο 5.5). Πρώτα έλαβε μέρος η διαδικασία της εκπαίδευσης (*training*) του δικτύου με χρήση συνόλων δεδομένων που υπάρχουν διαθέσιμα στο διαδίκτυο. Αφού προσδιορίστηκαν και ρυθμίστηκαν οι παράμετροι του δικτύου μέσω της επαλήθευσης (*validation*), παρουσιάστηκαν και σχολιάστηκαν τα στατιστικά αποτελέσματα που προέκυψαν (κεφάλαιο 6.1). Λόγω της φύσης του προβλήματος και της πληροφορίας πραγματικού χρόνου που πραγματεύεται, επιλέχθηκε η υλοποίηση και λειτουργία του μοντέλου στο Edge (κεφάλαιο 4) και συγκεκριμένα του Development Board της Coral (κεφάλαιο 4.4.3). Τέλος εκτελέστηκε δοκιμή κανονικής λειτουργίας πάνω σε άγνωστα δεδομένα και συλλέχθηκαν και σχολιάστηκαν τα αποτελέσματα αυτής (κεφάλαιο 6.2).

7.2 Μελλοντικές επεκτάσεις και βελτιώσεις

Παρατίθενται ορισμένες προτάσεις ή επεκτάσεις που αποσκοπούν στην πιθανή ενίσχυση της απόδοσης του συνολικού συστήματος.

- **Συνολική κανονικοποίηση φιγούρας.**

Κανονικοποίηση της θέσης όλων των σημείων κλειδιών σε σχέση με το κέντρο βάρους τους. Αυτό θα εξαλείψει εντελώς τον πληροφορία της θέσης στο χώρο και η εξέταση πλέον θα γίνεται μόνο σε επίπεδο φιγούρας. Παρουσιάζονται ορισμένες δυσκολίες υπολογισμού του κέντρου βάρους των σημείων-κλειδιών, καθώς σε πολλά frames δεν παρέχεται πληροφορία για όλα τα σημεία αυτά.

- **Δημιουργία νέου συνόλου δεδομένων.**

Βιντεοσκόπηση νέων πτώσεων με στόχο τον εμπλουτισμό του συνόλου δεδομένων εκπαίδευσης και ειδικότερα της κατηγορίας των πτώσεων. Προβλέπεται ότι αυτό θα ενισχύσει σημαντικά την ικανότητα του δικτύου να εντοπίζει πτώσεις.

- **Επανεκπαίδευση του δικτύου PoseNet.**

Στην συγκεκριμένη υλοποίηση δε χρησιμοποιούνται όλα τα σημεία-κλειδιά που εξαγονται από το δίκτυο PoseNet. Συνεπώς, το σύστημα επιβαρύνεται τη χρονική καθυστέρηση που προκαλεί ο υπολογισμός της θέσης αυτών των σημείων χωρίς όμως τελικά η πληροφορία αυτή να χρησιμοποιείται. Προτείνεται η χρήση μεταφορική μάθησης (*transfer learning*), δηλαδή χρήση μόνο κάποιων στρωμάτων του δικτύου και επανεκπαίδευση των υψηλότερων στρωμάτων για την εξαγωγή μόνο των επιθυμητών σημείων-κλειδιών.

- **Χρήση tracker για την εκτίμηση φιγούρας.**

Μια ακόμα ιδέα είναι η χρήση προγράμματος tracking για την εκτίμηση της θέσης των σημείων κλειδιών. Πρακτικά το δίκτυο PoseNet δε θα χρειάζεται να καλείται για κάθε εικόνα (*frame*), αλλά μόνο για μερικά από αυτά. Στα ενδιάμεσα frames μπορεί να χρησιμοποιείται tracker, δηλαδή να εκτιμάται η επόμενη θέση ενός σημείου-κλειδιού με βάση τη θέση του στο παρόν frame. Αυτό θα βοηθήσει σημαντικά στην χρονική απόδοση του συνολικού συστήματος.

Βιβλιογραφία

- [1] L. Martínez-Villaseñor, H. Ponce, J. Brieva, E. Moya-Albor, J. Núñez-Martínez, and C. Peñafort-Asturiano, "UP-Fall Detection Dataset: A Multimodal Approach," *Sensors (Basel)*, vol. 19, no. 9, p. 1988, Apr. 2019, doi: 10.3390/s19091988.
- [2] I. Charfi, J. Miteran, J. Dubois, M. Atri, and R. Tourki, *Definition And Performance Evaluation Of A Robust SVM Based Fall Detection Solution*. 2012.
- [3] B. Kwolek and M. Kepski, "Human fall detection on embedded platform using depth maps and wireless accelerometer," *Comput. Methods Programs Biomed.*, vol. 117, no. 3, pp. 489–501, 2014, doi: <https://doi.org/10.1016/j.cmpb.2014.09.005>.
- [4] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, 1943.
- [5] A. M. TURING, "I.—COMPUTING MACHINERY AND INTELLIGENCE," *Mind*, vol. LIX, no. 236, pp. 433–460, Oct. 1950, doi: 10.1093/mind/LIX.236.433.
- [6] F. Rosenblatt, *The Perceptron, a Perceiving and Recognizing Automaton Project Para. Cornell Aeronautical Laboratory*, 1957.
- [7] A. L. Samuel, "Some Studies in Machine Learning Using the Game of Checkers.," *IBM J. Res. Dev.*, vol. 3, no. 3, pp. 210–229, 1959, doi: 10.1147/rd.33.0210.
- [8] T. Cover and P. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, pp. 21–27, 1967.
- [9] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, no. 4, pp. 193–202, 1980, doi: 10.1007/BF00344251.
- [10] H. P. Moravec, "The Stanford Cart and the CMU Rover," *Proc. IEEE*, vol. 71, no. 7, pp. 872–884, 1983, doi: 10.1109/PROC.1983.12684.
- [11] G. DeJong and R. Mooney, "Explanation-based learning: An alternative view," *Mach. Learn.*, vol. 1, no. 2, pp. 145–176, 1986.

- [12] T. Sejnowski and C. Rosenberg, "Parallel Networks that Learn to Pronounce English Text," *Complex Syst.*, vol. 1, 1987.
- [13] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.
- [14] T. K. Ho, "Random decision forests," in *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 1995, vol. 1, pp. 278–282 vol.1, doi: 10.1109/ICDAR.1995.598994.
- [15] S. Hochreiter and J. Schmidhuber, "Long Short-term Memory," *Neural Comput.*, vol. 9, pp. 1735–1780, Dec. 1997, doi: 10.1162/neco.1997.9.8.1735.
- [16] S. S. Haykin, "Neural networks and learning machines/Simon Haykin." New York: Prentice Hall, 2009.
- [17] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li, *ImageNet: a Large-Scale Hierarchical Image Database*. 2009.
- [18] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," *J. Mach. Learn. Res. - Proc. Track*, vol. 9, pp. 249–256, Jan. 2010.
- [19] Q. V. Le et al., "Building high-level features using large scale unsupervised learning," Dec. 2011, [Online]. Available: <http://arxiv.org/abs/1112.6209>.
- [20] M. D. Zeiler and R. Fergus, "Visualizing and Understanding Convolutional Networks," Nov. 2013, [Online]. Available: <http://arxiv.org/abs/1311.2901>.
- [21] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," Dec. 2014, [Online]. Available: <http://arxiv.org/abs/1412.6980>.
- [22] O. Russakovsky et al., "ImageNet Large Scale Visual Recognition Challenge," Sep. 2014, [Online]. Available: <http://arxiv.org/abs/1409.0575>.
- [23] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708, doi: 10.1109/CVPR.2014.220.
- [24] M. A. Nielsen, *Neural Networks and Deep Learning*. Determination Press, 2015.

- [25] A. Kendall, M. Grimes, and R. Cipolla, "PoseNet: A Convolutional Network for Real-Time 6-DOF Camera Relocalization," May 2015, [Online]. Available: <http://arxiv.org/abs/1505.07427>.
- [26] Y. LeCun, Y. Bengio, and G. Hinton, "Deep Learning," Nature, vol. 521, pp. 436–444, May 2015, doi: 10.1038/nature14539.
- [27] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," IEEE internet things J., vol. 3, no. 5, pp. 637–646, 2016.
- [28] J. Koushik, "Understanding Convolutional Neural Networks," May 2016, [Online]. Available: <http://arxiv.org/abs/1605.09081>.
- [29] M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems," Mar. 2016, [Online]. Available: <http://arxiv.org/abs/1603.04467>.
- [30] D. Silver et al., "Mastering the game of Go with deep neural networks and tree search," Nature, vol. 529, pp. 484–489, Jan. 2016, doi: 10.1038/nature16961.
- [31] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," Dec. 2018, [Online]. Available: <http://arxiv.org/abs/1812.08008>.
- [32] T. B. Brown et al., "Language Models are Few-Shot Learners," May 2020, [Online]. Available: <http://arxiv.org/abs/2005.14165>.
- [33] A. Yazdanbakhsh, K. Seshadri, B. Akin, J. Laudon, and R. Narayanaswami, "An Evaluation of Edge TPU Accelerators for Convolutional Neural Networks," Feb. 2021, [Online]. Available: <http://arxiv.org/abs/2102.10423>.