Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και
Υπολογιστών

Νευρωνικά Δίκτυα με Κάψουλες: Συμφωνία μεταξύ καψουλών μέσω Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΠΙΤΟΣΚΑΣ ΙΩΑΝΝΗΣ

Επιβλέπων : Στέφανος Κόλλιας
Καθηγητής Ε.Μ.Π.

Συνεπιβλέπων : Εμμανουήλ Σεφέρης
Υποφήφιος Διδάκτωρ

Αθήνα, Ιούλιος 2022

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και
Υπολογιστών

Νευρωνικά Δίκτυα με Κάψουλες: Συμφωνία μεταξύ καψουλών μέσω Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΠΙΤΟΣΚΑΣ ΙΩΑΝΝΗΣ

Επιβλέπων : Στέφανος Κόλλιας
Καθηγητής Ε.Μ.Π.

Συνεπιβλέπων : Εμμανουήλ Σεφέρης
Υποφήφιος Διδάκτωρ

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 15η Ιουλίου 2022.

Στέφανος Κόλλιας Καθηγητής Ε.Μ.Π.	Ανδρέας Γεώργιος Σταφυλοπάτης Καθηγητής Ε.Μ.Π.	Γιώργος Στάμου Καθηγητής Ε.Μ.Π.

Αθήνα, Ιούλιος 2022

Πιτόσκας Ιωάννης

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

Περίληψη

Η παρούσα διπλωματική εργασία μελετά ένα είδος τεχνητών νευρωνικών δικτύων που προτάθηκε πριν λίγα μόλις χρόνια στον χώρο της όρασης υπολογιστών και της μηχανικής μάθησης, τα Νευρωνικά Δίκτυα με Κάψουλες. Πρόκειται για ένα είδος δίκτυων στα οποία η κλασσική ιδέα των βαθμωτών νευρώνων αντικαθίσταται από κάψουλες, οι οποίες είναι συνήθως μαθηματικές μονάδες, όπως διανύσματα ή μήτρες, που ενθυλακώνουν την έννοια του προσανατολισμού με τρόπο που εξασφαλίζει την έννοια του ισομεταβλητού της οπτικής γωνίας. Αυτό έχει ως αποτέλεσμα την μοντελοποίηση των ιεραρχικών σχέσεων των οντοτήτων που βρίσκονται σε μια εικόνα, ως απόπειρα πιο εύστοχης μίμησης των βιολογικών νευρωνικών αποκρίσεων. Στην ερευνητική αυτή εργασία, μελετόνται και προτείνονται ορισμένες νέες μέθοδοι δρομολόγησης καψουλών, που χρησιμοποιούν ιδέες από τον ευρύ τομέα της μηχανικής μάθησης προς όφελος της θεωρίας των καψουλών. Οι αλγόριθμοι έχουν ως βασική ιδέα, μια πιθανοτική προσέγγιση που υπολογίζει τις πιθανότητες ύπαρξης των υπο-οντοτήτων μιας εικόνας (ή μιας περιοχής αυτής), και σταθμίζει με αυτές τις αντίστοιχες μονάδες προσανατολισμού τους, για την σύνθεση μιας υψηλότερου επιπέδου οντότητας. Αυτή είναι η ιδέα της πιθανοτικής στάθμισης μέσου, και την εμπλουτίσαμε με επιπλέον μηχανισμούς προς σχεδιασμό αλγορίθμων που θα εισάγουν στο μοντέλο τυχαιότητα, θα αγνοούν ακραίες τιμές ή θα δίνουν βάση στην μεγιστοποίηση της "συμφωνίας" μεταξύ καψουλών, με μέτρο αυτής να είναι διάφορες γνωστές μετρικές ομοιότητας. Στην εργασία αυτή διαπιστώνεται πως αυτές οι μεταξύ τους διαφορετικές προσεγγίσεις της ίδιας βασικής μεθόδου παρουσιάζουν διαφορετικές επιδόσεις οι οποίες είναι άμεσα εξαρτώμενες από τον τύπο του συνόλου δεδομένων στο οποίο εκπαιδεύεται το εκάστοτε δίκτυο καψουλών με τα σύνολα δεδομένων που χρησιμοποιήθηκαν είναι τα MNIST, smallNORB, Fashion-MNIST, SVHN και CIFAR-10.

Λέξεις κλειδιά

Μηχανική Μάθηση, Βαθιά Νευρωνικά Δίκτυα, Όραση Υπολογιστών, Συνελικτικά Νευρωνικά Δίκτυα, Μηχανισμός Pooling, Αμετάβλητο, Διανυσματικές Κάψουλες, Κάψουλες Μήτρες, Νευρωνικά Δίκτυα με Κάψουλες, Ισομεταβλητό, Δυναμική Δρομολόγηση-με-Συμφωνία, EM Δρομολόγηση, Ανακατασκευή, Προσθήκη Συντεταγμένων, Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου, Dropout Δρομολόγηση, Δρομολόγηση Υποσυνόλου, RANSAC Δρομολόγηση, Πλήρως Συνδεδεμένο Επίπεδο Καψουλών, Συνελικτικό Επίπεδο Καψουλών.

Abstract

This thesis studies a type of artificial neural network proposed a few years ago in the field of computer vision and machine learning, the Capsule Neural Networks. It is a type of network in which, the traditional neurons are replaced by capsules, which are most commonly mathematical units, such as vectors or matrices, that encapsulate the concept of orientation, in a way that ensures viewpoint equivariance. This results in the better modeling of hierarchical relationships of entities located in an image, as an attempt to more closely mimic biological neural response. In this research paper, several new capsule routing methods are studied and proposed, which use ideas from the wide field of machine learning for the benefit of capsule theory. The algorithms' main idea is a probabilistic approach which calculates the activation probabilities of the sub-entities of an image (or a part of it) to exist, and weights the respective units of orientation with them, for the composition of a higher-level entity. This is the idea of probabilistic weighted averaging, and we enriched it with additional mechanisms in order to design algorithms that would add randomness to the model, would ignore outlying capsules or even route capsules by their "agreement", which is measured using commonly known similarity metrics. In this thesis it is found that these different approaches of the same main method, show different performances which are dependent on the type of dataset on which the respective capsule network is trained with the datasets used being MNIST, smallNORB, Fashion-MNIST, SVHN, CIFAR-10.

Key words

Machine Learning, Deep Neural Networks, Computer Vision, Convolutional Neural Networks, Pooling Mechanism, Invariant, Vector Capsules, Matrix Capsules, Capsule Neural Networks, Equivariant, Dynamic Routing-by-Agreement, EM Routing, Reconstruction, Coordinate Addition, Probabilistic Weighted Average Routing, Dropout Routing, Subset Routing, RANSAC Routing, Fully Connected Capsule Layer, Convolutional Capsule Layer.

Ευχαριστίες

Πρώτα και σημαντικότερα, θα ήθελα να ευχαριστήσω θερμά τον επιβλέποντα καθηγητή και καθοδηγητή της παρούσας διπλωματικής εργασίας, κ. Σ. Κόλλια, Καθηγητή Ε.Μ.Π., που κατά πρώτον είχα την τιμή να είναι καθηγητής μου σε πολλαπλά μαθήματα του ακαδημαϊκού προγράμματος, και που κατ' επέκταση μου έδωσε τη δυνατότητα και το έναυσμα να ερευνήσω και να μελετήσω βαθύτερα έναν πολλά υποσχόμενο κλάδο του ευρύτερου τομέα της Μηχανικής Μάθησης. Θέλω να τον ευχαριστήσω για την συμβολή του, συνεργασιμότητα, την αμεσότητα και την κατανόηση που έδειξε σε όλο αυτό το διάστημα της εκπόνησης της εργασίας. Παράλληλα θα ήθελα να ευχαριστήσω τα μέλη της επιτροπής κ. Α. Γ. Σταφυλοπάτη, Καθηγητή Ε.Μ.Π., και κ. Γ. Στάμου, Καθηγητή Ε.Μ.Π., για τον χρόνο που αφιέρωσαν, γεγονός που με τιμάει ιδιαιτέρως.

Οφείλω ακόμη ένα τεράστιο ευχαριστώ στον υποψήφιο διδάκτορα του Ε.Μ.Π. και συνεργάτη, Μάνο Σεφέρη του οποίου η συμβολή ήταν ζωτικής σημασίας για τόσο για την εξέλιξη όσο και για την επιτυχή ολοκλήρωση της ερευνητικής αυτής εργασίας. Τον ευχαριστώ βαθύτατα για το μεράκι, την υπομονή και την υποστήριξη, που είχαν ως αποτέλεσμα μια τόσο καρποφόρα συνεργασία.

Με την ολοκλήρωση της διπλωματικής αυτής εργασίας, το ταξίδι στη Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Ε.Μ.Π. έρχεται στο τέλος του. Ένα ταξίδι γεμάτο γνώσεις, εμπειρίες, προκλήσεις και όχι μόνο. Αλλά πάνω απ' όλα πρόκειται για ένα τάξιδι στο οποίο συνοδοιπόροι μου ήταν άνθρωποι αξιόλογοι και όλοι με τον δικό τους τρόπο ξεχωριστοί. Θέλω λοιπόν να πω ένα μεγάλο ευχαριστώ από καρδιάς στους συμφοιτητές πλην φίλους μου Αντώνη Π., Γιάννη Σ., Γιώργο Χ., Αλίκη Μ., Θοδωρή Κ., Μαρίνο Χ. και Φοίβο Ο. που είχα την τύχη να μοιράστω αυτό το ταξίδι μαζί τους.

Δε θα μπορούσα σε καμία περίπτωση βέβαια να παραλείψω να ευχαριστήσω την οικογένεια μου και ιδιαίτερα τον αδερφό μου Άρη και τους γονείς μου Στέργιο και Πολέτα για την υποστήριξη τους όλα αυτά τα χρόνια. Δίχως την ανιδιοτελή αγάπη αυτών των αθρώπων και των όσων μου έχουν προσφέρει απλόχερα στο πέρας της μέχρι σήμερα ζωής μου, οι προοπτικές για την εξέλιξη μου ως άνθρωπος, με την ευρύτερη έννοια, δε θα ήταν οι ίδιες.

Κλείνοντας θα ήθελα να κάνω ιδιαίτερη αναφορά στους φίλους μου Νίκο Κ., Στράτο Μ., Μάριο Μ., Ευθύμη Ξ. και Χριστίνα Κ.,των οποίων η κατανόηση, υποστήριξη και πίστη καθ' όλη τη διάρκεια των σπουδών μου αποτέλουσε και συνεχίζει να αποτελεί σταθερή κινητήριο δύναμη για την επίτευξη των στόχων μου.

Πιτόσκας Ιωάννης,

Αθήνα, 15η Ιουλίου 2022

Περιεχόμενα

Κατάλογος πινάκων

Κατάλογος σχημάτων

Κατάλογος αλγορίθμων

Κεφάλαιο 1

Εισαγωγή

Η ανθρώπινη όραση αγνοεί ασήμαντες λεπτομέρειες χρησιμοποιώντας μια προσεκτικά καθορισμένη σειρά σημείων σταθεροποίησης προκειμένου να διασφαλίσει ότι μόνο ένα μικρό κλάσμα της οπτικής συστοιχίας υποβάλλεται σε επεξεργασία έτσι, ώστε να κατανοήσουμε τι μέρος της γνώσης για μια σκηνή προέρχεται από την αλληλουχία αυτών των σημείων σταθεροποίησης και πόση πληροφορία μπορούμε να αντλήσουμε από μία σταθεροποίηση. Στα πλαίσια αυτής της διπλωματικής εργασίας θα υποθέσουμε ότι τα σημεία σταθεροποίησης του οπτικού συστήματος δίνουν πολύ περισσότερα πέρα από την απλή αναγνώριση ενός αντικειμένου και των ιδιοτήτων του. Υποθέτουμε ότι το πολυεπίπεδο οπτικό μας σύστημα δημιουργεί μια δομή που μοιάζει με δέντρο ανάλυσης σε κάθε τέτοιο σημείο σταθεροποίησης και αγνοούμε το ζήτημα του τρόπου με τον οποίο αυτά τα δέντρα ανάλυσης μίας σταθεροποίησης συνδυάζονται για τον σχηματισμό πολλαπλών πιο σύνθετων σταθεροποιήσεων.

Τα δέντρα ανάλυσης γενικά κατασκευάζονται με δυναμική κατανομή μνήμης. Θα πρέπει [1] να υποθέσουμε ότι για ένα σημείο σταθεροποίησης, κατασκευάζεται ένα δέντρο ανάλυσης από ένα πολυεπίπεδο νευρωνικό δίκτυο. Κάθε επίπεδο θα χωριστεί σε πολλές μικρές ομάδες νευρώνων που ονομάζονται "κάψουλες" [2] και κάθε κόμβος στο δέντρο ανάλυσης θα αντιστοιχεί σε μια ενεργή κάψουλα. Χρησιμοποιώντας μια επαναληπτική ή μη διαδικασία δρομολόγησης, κάθε ενεργή κάψουλα θα επιλέξει μια κάψουλα στο παραπάνω επίπεδο ως γονέα του στο δέντρο. Για τα υψηλότερα επίπεδα ενός οπτικού συστήματος, αυτή η διαδικασία είναι μια προσέγγιση για την επίλυση του προβλήματος της ανάθεσης μικρότερων μερών για τη σύνθεση μεγαλύτερων.

Οι δραστηριότητες των νευρώνων μέσα σε μια ενεργή κάψουλα αντιπροσωπεύουν τις διάφορες ιδιότητες μιας συγκεκριμένης οντότητας που υπάρχει στην εικόνα. Αυτές οι ιδιότητες μπορεί να περιλαμβάνουν πολλούς διαφορετικούς τύπους παραμέτρων στιγμιοτύπου όπως είναι η πόζα (θέση, μέγεθος, προσανατολισμός), η παραμόρφωση, η ταχύτητα, η απόχρωση, η υφή κ.λ.π. Μια πολύ ιδιαίτερη βέβαια ιδιότητα είναι η ύπαρξη μιας οντότητας στην εικόνα. Ένας προφανής τρόπος κωδικοποίησης της ύπαρξης είναι χρησιμοποιώντας μια ξεχωριστή λογιστική μονάδα της οποίας η έξοδος είναι η πιθανότητα ύπαρξης της οντότητας.

Τα συνελικτικά νευρωνικά δίκτυα (CNN) χρησιμοποιούν μετατοπισμένες εκδοχές των ανιχνευτών χαρακτηριστικών που έχουν μάθει. Αυτό τους επιτρέπει να αξιοποιήσουν τα βάρη που έμαθαν ότι λειτουργούν καλά σε μια θέση της εικόνας, και για άλλες θέσεις της ίδιας εικόνας. Αυτό έχει αποδειχθεί εξαιρετικά χρήσιμο αναφορικά με την ερμηνεία των εικόνων. Αντικαθιστούμε λοιπόν τους ανιχνευτές χαρακτηριστικών βαθμωτής εξόδου των CNN με κάψουλες διανυσματικής εξόδου και τον μηχανισμό υποδειγματοληψίας μεγίστου (max-pooling) με μια διαδικασία δρομολόγησης. Όπως και στα CNN, οι κάψουλες υψηλότερου επιπέδου καλύπτουν μεγαλύτερες περιοχές της εικόνας. Μάλιστα, σε αντίθεση με το max-pooling, κρατάμε πληροφορίες σχετικά με την ακριβή θέση της οντότητας. Για κάψουλες χαμηλότερου επιπέδου, οι πληροφορίες θέσης κωδικοποιούνται με βάση το ποια κάψουλα είναι ενεργή. Καθώς ανεβαίνουμε στην ιεραρχία των υπο-οντοτήτων του υπό εξέταση αντικειμένου, όλο και περισσότερες από τις πληροφορίες θέσης κωδικοποιούνται σε στοιχεία του διανύσματος εξόδου μιας κάψουλας. Το γεγονός ότι οι κάψουλες υψηλότερου επιπέδου αντιπροσωπεύουν πιο σύνθετες οντότητες με περισσότερους βαθμούς ελευθερίας υποδηλώνει ότι η διάσταση των καψουλών θα πρέπει να αυξάνεται καθώς ανεβαίνουμε στην ιεραρχία.

1.1 Σκοπός της εργασίας

Σκοπός της παρούσας διπλωματικής εργασίας είναι η βαθύτερη μελέτη των Νευρωνικών Δικτύων με Κάψουλες, εστιάζοντας στον μηχανισμό με τον οποίο οι ψήφοι των χαμηλότερου επιπέδου καψουλών δρομολογούνται για τον υπολογισμό των υψηλότερου επιπέδου καψουλών.

Πιο συγκεκριμένα θα μελετήσουμε και θα προτείνουμε ορισμένες νέες μεθόδους δρομολόγησης καψουλών, που χρησιμοποιούν ιδέες από τον ευρύ τομέα της μηχανικής μάθησης προς όφελος της θεωρίας των καψουλών. Οι αλγόριθμοι αυτοί έχουν ως βασική ιδέα, μια πιθανοτική προσέγγιση που υπολογίζει τις πιθανότητες ύπαρξης των υπο-οντοτήτων μιας εικόνας (ή μιας περιοχής αυτής), και σταθμίζει με αυτές τις αντίστοιχες μονάδες προσανατολισμού τους, για την σύνθεση μιας υψηλότερου επιπέδου οντότητας. Αυτή είναι η ιδέα της πιθανοτικής στάθμισης μέσου, και την εμπλουτίσαμε με επιπλέον μηχανισμούς προς σχεδιασμό αλγορίθμων που θα εισάγουν στο μοντέλο τυχαιότητα, θα αγνοούν ακραίες τιμές ή θα δίνουν βάση στην μεγιστοποίηση της "συμφωνίας" μεταξύ καψουλών, με μέτρο αυτής να είναι διάφορες γνωστές μετρικές ομοιότητας. Στην εργασία αυτή διαπιστώνεται πως αυτές οι μεταξύ τους διαφορετικές προσεγγίσεις της ίδιας βασικής μεθόδου παρουσιάζουν διαφορετικές επιδόσεις. Επιγραμματικά, στην εργασία αυτή θα προσπαθήσουμε να δώσουμε απάντηση στα εξής ερωτήματα:

1. Ποια η σημασία της πιθανότητας ύπαρξης της εκάστοτε υπο-οντότητας για τη σύνθεση μιας υψηλότερου επιπέδου οντότητας;
2. Ποια η σημασία του προσανατολισμού της εκάστοτε υπο-οντότητας για τη σύνθεση μιας υψηλότερου επιπέδου οντότητας;
3. Ποια η σημασία της επιλογής καψουλών με μέγιστη "συμφωνία";
4. Πόσο και πώς επηρεάζει ο τύπος των δεδομένων την απάντηση στα παραπάνω;
5. Κατά πόσο κρίνεται απαραίτητη η επαναληψιμότητα ενός αλγορίθμου δρομολόγησης;

1.2 Προηγούμενες Μελέτες

Οι πρώτες μελέτες, που έδωσαν το έναυσμα καθιστώντας τα Νευρωνικά Δίκτυα με Κάψουλες ένα πολλά υποσχόμενο εργαλείο για τον τομέα της βαθιάς μάθησης, έγιναν από τον Geoffrey Hinton και την ομάδα του εισάγοντας πρώτα τις διανυσματικές κάψουλες με χρήση ενός αλγορίθμου Δυναμικής Δρομολόγησης-με-Συμφωνία (Dynamic Routing-by-Agreement), και στη συνέχεια μια ιδέα καψουλών [3], που συνίστανται από την πόζα σε μορφή μητρώου και μιας λογιστικής μονάδας που αντιπροσωπεύει πιθανότητα, η δρομολόγηση των οποίων είναι εμπευσμένη από την γνωστό αλγόριθμο Προσδοκίας-Μεγιστοποίησης ή αλλιώς EM (Expectation-Maximization) [4, 5] και ονομάστηκε EM Δρομολόγηση [6].

Σε συνέχεια αυτών έγιναν περαιτέρω μελέτες πάνω στα νευρωνικά δίκτυα με κάψουλες που βελτίωσαν τις επιδόσεις σε μια σειρά από προβλήματα κατηγοριοποίησης. Πολύ σημαντική ήταν η συμβολή των Σ. Κόλλια, F. De Sousa Ribeiro και της ομάδας τους, που το 2019 με αφορμή την ερευνητική εργασία "Deep Bayesian Self-Training" [7], μπορέσαν εν συνεχεία να ερευνήσουν βαθύτερα την θεωρία των καψουλών προτείνοντας έναν καινοτόμο αλγόριθμο δρομολόγησης VB-Routing που χρησιμοποιεί την Μπεϋζιανή θεωρία συμπερασματολογίας (Variational Bayes Inference) [8, 9], επιτυγχάνοντας state-of-the-art επιδόσεις.

Τέλος, το 2020 οι Vittorio Mazzia, Francesco Salvetti και Marcello Chiaberge, κινούμενοι σε μια διαφορετική κατεύθυνση σχεδίασαν δίκτυα καψουλών στην δρομολόγηση των οποίων εισήχθη η σύγχρονη ιδέα του μηχανισμού προσοχής (Attention mechanism), τα οποία και ονόμασαν Efficient-CapsNet [10].

Βέβαια δε θα μπορούσαμε να παραλείψουμε την αναφορά της συμβολής, στη συγκεκριμένη διπλωματική εργασία, περαιτέρω μελετών που έχουν γίνει, από τον Σ. Κόλλια και την ομάδα

του, και εντός και εκτός του "Εργαστηρίου Τεχνητής Νοημοσύνης και Συστημάτων Μάθησης" (AILS Lab). Κάποιες από αυτές αναφέρθηκαν ήδη, όπως οι "Deep Bayesian Self-Training" [7], "Capsule Routing via Variational Bayes" [8] και "Introducing Routing Uncertainty in Capsule Networks" [9], και αφορούν καθαρά τη θεωρία των νευρωνικών δικτύων με κάψουλες. Μεγάλη όμως σημασία δόθηκε και στις ερευνητικές εργασίες "Adaptation and contextualization of deep neural network models" [11], "Deep Transparent Prediction through Latent Representation Analysis" [12] και "Transparent adaptation in deep medical image diagnosis" [13]. Από τις ερευνητικές αυτές εργασίες, που αφορούσαν διαφορετικά αλλά κοντινά πεδία της μηχανικής μάθησης, αντλήθηκε γνώση και μαζί διάφορες ιδέες που θα μπορούσαν να τροποποιηθούν κατάλληλα ώστε να χρησιμοποιηθούν προς όφελος της θεωρίας των καψουλών.

Μέρος Ι

Θεωρητικό Μέρος

Κεφάλαιο 2

Θεωρία Διανυσμάτων και Διανυσματικοί Χώροι

Σε αυτήν την ενότητα θα γίνει αναφορά σε κάποιες βασικές μαθηματικές έννοιες από το πεδίο της γραμμικής άλγεβρας που θα φανούν χρήσιμες στην συνέχεια.

2.1 Διανυσματικοί Χώροι στο $\mathbb{R}$

Ονομάζουμε διανυσματικό χώρο V στο σύνολο των πραγματικών αριθμών $\mathbb{R}$ ένα σύνολο V με διανύσματα για τα οποία ισχύουν οι εξής ιδιότητες (έστω $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in V$ και $c, c_1, c_2 \in \mathbb{R}$):

1. $\boldsymbol{u} + \boldsymbol{v} = \boldsymbol{v} + \boldsymbol{u}$
   (Αντιμεταθετική ιδιότητα της πρόσθεσης)
2. $\boldsymbol{u} + (\boldsymbol{v} + \boldsymbol{w}) = (\boldsymbol{u} + \boldsymbol{v}) + \boldsymbol{w}$
   (Προσεταιριστική ιδιότητα της πρόσθεσης)
3. $\exists\ \mathbf{0} \in V : \mathbf{0} + \boldsymbol{u} = \boldsymbol{u}$
   (Ουδέτερο στοιχείο της πρόσθεσης)
4. $\boldsymbol{u} \in V \Rightarrow (-\boldsymbol{u}) \in V : \boldsymbol{u} + (-\boldsymbol{u}) = \mathbf{0}$
   (Αντίθετο στοιχείο της πρόσθεσης)
5. $1 \cdot \boldsymbol{u} = \boldsymbol{u}$
   (Ουδέτερο στοιχείο του βαθμωτού πολλαπλασιασμού)
6. $c_1 \cdot (c_2 \cdot \boldsymbol{u}) = (c_1 \cdot c_2) \cdot \boldsymbol{u}$
   (Συμβατότητα του βαθμωτού πολλαπλασιασμού με τον πολλαπλασιασμό στοιχείων του χώρου)
7. $c \cdot (\boldsymbol{u} + \boldsymbol{v}) = c \cdot \boldsymbol{u} + c \cdot \boldsymbol{v}$
   (Επιμεριστική ιδιότητα σε σχέση με την προσθήκη διανύσματος)
8. $(c_1 + c_2) \cdot \boldsymbol{u} = c_1 \cdot \boldsymbol{u} + c_2 \cdot \boldsymbol{u}$
   (Επιμεριστική ιδιότητα σε σχέση με την προσθήκη στοιχείων του χώρου)

2.2 Διανυσματικοί Χώροι με Νόρμα στο $\mathbb{R}$

Έστω V ένας διανυσματικός χώρος. Ονομάζουμε νόρμα ενός διανύσματος $\boldsymbol{u} \in V$ και συμβολίζουμε με $\|\boldsymbol{u}\|$ μια συνάρτηση $\|\cdot\| : V \to [0, +\infty)$ για την οποία ισχύουν οι παρακάτω ιδιότητες (έστω $\boldsymbol{u}, \boldsymbol{v} \in V$ και $c \in \mathbb{R}$) :

1. $\|\boldsymbol{u}\| \geq 0$
2. $\|\boldsymbol{u}\| = 0 \Leftrightarrow \boldsymbol{u} = \mathbf{0}$

3. $\|c \cdot \boldsymbol{u}\| = |c| \cdot \|\boldsymbol{u}\|$

4. $\|\boldsymbol{u} + \boldsymbol{v}\| \leq \|\boldsymbol{u}\| + \|\boldsymbol{v}\|$ (τριγωνική ανισότητα)

Ένας διανυσματικός χώρος εφοδιασμένος με μία νόρμα ονομάζεται Διανυσματικός Χώρος με Νόρμα ή Νορμικός Διανσυσματικός Χώρος.

Η νόρμα μπορεί να χρησιμοποιηθεί για να επάγει μία μετρική ορίζοντας, $d(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|$. Ένας πλήρης νορμικός διανυσματικός χώρος (υπό τη συνήθη μετρική η οποία επάγεται από τη νόρμα) καλείται χώρος Banach.

Η νόρμα ορίζει τη γνωστή έννοια του μεγέθους ενός στοιχείου (ο συμβολισμός με δύο κάθετες γραμμές χρησιμοποιείται για να τονίσει το γεγονός ότι η νόρμα είναι γενίκευση της έννοιας της απόλυτης τιμής). Η νόρμα της διαφοράς δύο στοιχείων είναι ένα μέτρο της εγγύτητας ή ομοιότητας των στοιχείων και επίσης καθορίζουν τη μορφή της γειτονιάς ενός στοιχείου. Ο τρόπος με τον οποίο ορίζεται η νόρμα αφήνει περιθώρια για πολλές συναρτήσεις οι οποίες μπορεί να είναι υποψήφιες για τον σκοπό αυτό.

Χαρακτηριστική περίπτωση νόρμας ενός διανύσματος $\boldsymbol{u}$ αποτελεί η L^p νόρμα, για $p \in \mathbb{R}$, $p \geq 1$, η οποία ορίζεται ως εξής:

$$\|\boldsymbol{u}\|_p = \left(\sum_i |u_i|^p \right)^{\frac{1}{p}} \tag{2.1}$$

Η επιλογή του p εξαρτάται από τη συγκεκριμένη εφαρμογή και έχει διαφορετική φυσική σημασία. Για παράδειγμα, η L^1 νόρμα υπολογίζει απλά το άθροισμα των απόλυτων τιμών των στοιχείων του διανύσματος, η L^2 νόρμα είναι η γνωστή Ευκλείδεια νόρμα, δηλαδή η απόσταση της αρχής των αξόνων από το σημείο που προσδιορίζεται από το εκάστοτε διάνυσμα κ.ο.κ. Συνηθίζουμε ωστόσο να θεωρούμε ως μέγεθος ενός διανύσματος την L^2 νόρμα του. Κατά συνέπεια, η χρήση της Ευκλείδειας νόρμας της διαφοράς $\|\boldsymbol{u} - \boldsymbol{v}\|$ αποτελεί μια από τις πιο συνήθεις τεχνικές για τη μέτρηση της εγγύτητας δύο διανυσμάτων $\boldsymbol{u}, \boldsymbol{v}$, καθώς αντιπροσωπεύει την μεταξύ τους Ευκλείδεια απόσταση. Ακόμη, λόγω της ευρείας χρήσης της L^2 νόρμας συνηθίζουμε να συμβολίζουμε την L^2 νόρμα ένος διανύσματος $\boldsymbol{u}$ απλά ως $\|\boldsymbol{u}\|$. Αξίζει βέβαια να σημειωθεί ότι σε αρκετές εφαρμογές κρίνεται σημαντική και η χρήση της ειδικής περίπτωσης της L^∞ νόρμας, η αλλιώς της νόρμας μεγίστου. Η L^∞ ενός διανύσματος $\boldsymbol{u}$ ορίζεται απλά ως το μέγιστο κατ' απόλυτη τιμή στοιχείο του $\boldsymbol{u}$:

$$\|\boldsymbol{u}\|_\infty = \max_i |u_i| \tag{2.2}$$

2.3 Διανυσματικοί Χώροι με Εσωτερικό Γινόμενο στο $\mathbb{R}$

Έστω ένας νορμικός διανυσματικός χώρος $(V, \|\cdot\|)$. Ονομάζουμε εσωτερικό γινόμενο δύο διανυσμάτων $\boldsymbol{u}, \boldsymbol{v} \in (V, \|\cdot\|)$ και συμβολίζουμε με $\langle \boldsymbol{u}, \boldsymbol{v} \rangle$ μια συνάρτηση $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ για την οποία ισχύουν οι παρακάτω ιδιότητες (έστω $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w} \in (V, \|\cdot\|)$ και $c_1, c_2 \in \mathbb{R}$):

1. $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \langle \boldsymbol{v}, \boldsymbol{u} \rangle$ (συμμετρία)

2. $\langle c_1 \cdot \boldsymbol{u} + c_2 \cdot \boldsymbol{v}, \boldsymbol{w} \rangle = c_1 \cdot \langle \boldsymbol{u}, \boldsymbol{w} \rangle + c_2 \cdot \langle \boldsymbol{v}, \boldsymbol{w} \rangle$ (γραμμικότητα)

3. $\langle \boldsymbol{u}, \boldsymbol{u} \rangle = \|\boldsymbol{u}\|^2 \geq 0$

Ένας νορμικός διανυσματικός χώρος εφοδιασμένος με μια συνάρτηση εσωτερικού γινομένου ονομάζεται Διανυσματικός Χώρος με Εσωτερικό Γινόμενο. Αν αυτός ο νορμικός διανυσματικός χώρος είναι και πλήρης (δηλαδή είναι χώρος Banach) τότε ο διανυσματικός χώρος εσωτερικού γινομένου καλείται χώρος Hilbert.

Χαρακτηριστική περίπτωση διανυσματικού χώρου με εσωτερικό γινόμενο είναι ο πραγματικός n-διάστατος χώρος $\mathbb{R}^n$ ενσωματωμένος με την πράξη του απλού γινομένου ως εσωτερικό γινομένο. Ένας τέτοιος χώρος ονομάζεται Ευκλίδειος Χώρος (μιας και η χρησιμοποιούμενη νόρμα είναι η L^2 νόρμα ή αλλιώς Ευκλείδεια Νόρμα). Πιο αναλυτικά:

Έστω $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{R}^n$ με $\boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix}$ και $\boldsymbol{v} = \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix}$.

Τότε:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \left\langle \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_n \end{pmatrix}, \begin{pmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{pmatrix} \right\rangle = \boldsymbol{u}^{\mathsf{T}} \boldsymbol{v} = \sum_{i=1}^{n} u_i v_i \tag{2.3}$$

Μπορεί ακόμη να οριστεί και η γωνία θ μεταξύ των διανυσμάτων $\boldsymbol{u}, \boldsymbol{v}$ ως εξής:

$$\theta = \angle(\boldsymbol{u}, \boldsymbol{v}) = \cos^{-1}\left(\frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\| \cdot \|\boldsymbol{v}\|} \right) \tag{2.4}$$

Χρησιμοποιώντας το συνημίτονο της γωνίας μεταξύ δύο διανυσμάτων $\boldsymbol{u}, \boldsymbol{v}$ μπορούμε να ορίσουμε ένα ακόμη μέτρο για την ομοιότητα των $\boldsymbol{u}$ και $\boldsymbol{v}$, την Ομοιότητα Συνημιτόνου (2.5). Η ομοιότητα συνημιτόνου αποτελεί (μαζί με την Ευκλείδεια απόσταση που αναφέραμε σε προηγούμενη ενότητα) μια από τις πιο συνήθεις τεχνικές για τη μέτρηση της εγγύτητας δύο διανυσμάτων.

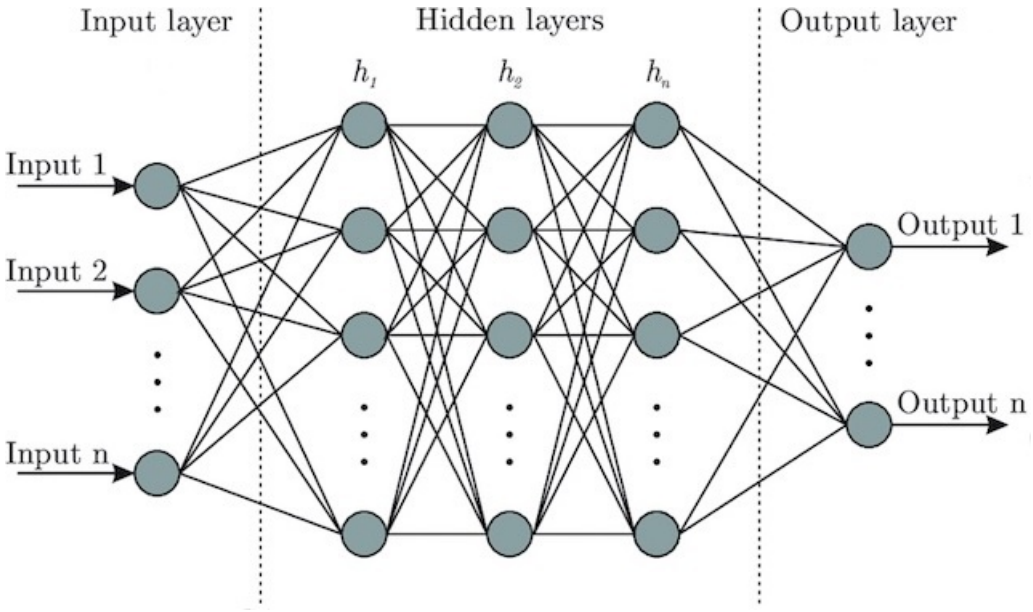
$$\text{cosine similarity} := \cos\theta = \frac{\langle \boldsymbol{u}, \boldsymbol{v} \rangle}{\|\boldsymbol{u}\| \cdot \|\boldsymbol{v}\|} \tag{2.5}$$

Κεφάλαιο 3

Βαθιά Νευρωνικά Δίκτυα

3.1 Νευρωνικά Δίκτυα με Εμπρόσθια Τροφοδότηση (FFNNs)

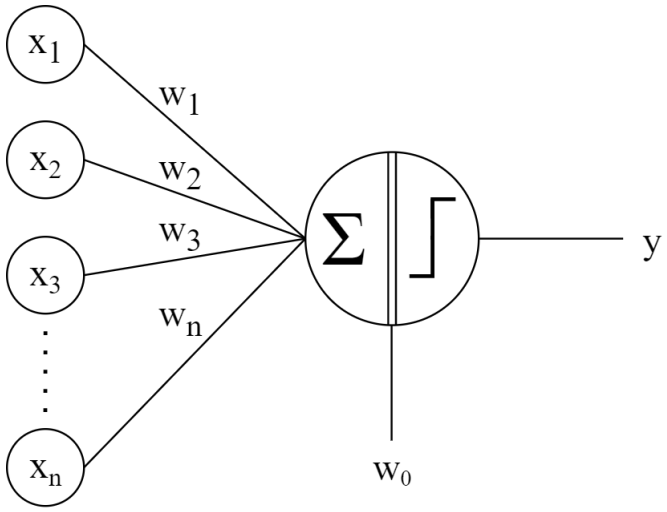
Τα νευρωνικά δίκτυα με εμπρόσθια τροφοδότηση είναι τεχνητά νευρωνικά δίκτυα στα οποία η πληροφορία ρέει αποκλειστικά από το επίπεδο εισόδου (Input Layer), μέσω των κρυφών επιπέδων (Hidden Layers), καταλήγοντας στο επίπεδο εξόδου (Output Layer), όπως φαίνεται στο Σχήμα 3.1.



Πηγή: Medium
Σχήμα 3.1: Νευρωνικό Δίκτυο με Εμπρόσθια Τροφοδότηση

Στόχος ενός νευρωνικού δικτύου με εμπρόσθια τροφοδότηση είναι να προσεγγίσει τη συνάρτηση f^* η οποία αντιστοιχίζει μια είσοδο $\boldsymbol{x}$ στην κατάλληλη έξοδο $y = f^*(\boldsymbol{x})$. Πιο συγκεκριμένα, σκοπός ενός τέτοιου δικτύου είναι να μάθει τις τιμές των παραμέτρων $\boldsymbol{\theta}$ ώστε η συνάρτηση $y = f(\boldsymbol{x}; \boldsymbol{\theta})$ να προσεγγίζει όσο το δυνατόν καλύτερα την f^* [14]. Η πιο απλή περίπτωση νευρωνικού δικτύου με εμπρόσθια τροφοδότηση είναι το perceptron [15], ένα πλήρως συνδεδεμένο δίκτυο ενός επιπέδου. Το perceptron είναι ένας αλγόριθμος επιβλεπόμενης μάθησης και πιο συγκεκριμένα ένας δυαδικός ταξινομητής που εφευρέθηκε το 1957 και βασίζεται σε μια γραμμική συνάρτηση απόφασης συνδυάζοντας το διάνυσμα εισόδου με ένα διάνυσμα βαρών ως εξής:

$$f(\boldsymbol{x}) = \begin{cases} 1 & \text{αν } \boldsymbol{w} \cdot \boldsymbol{x} > 0 \\ 0 & \text{αλλιώς} \end{cases} \tag{3.1}$$

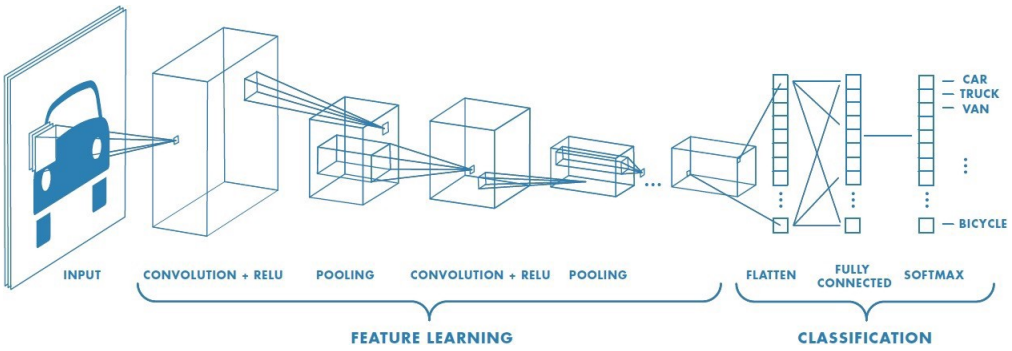


Σχήμα 3.2: Αρχιτεκτονική απλού perceptron ενός επιπέδου

Παρ' όλο που το perceptron έμοιαζε πολλά υποσχόμενο στην αρχή, εν τέλει αποδείχθηκε ότι τα perceptrons είναι ικάνα μόνο για την εκμάθηση γραμμικών προτύπων [16]. Προκειμένου να καταπολεμηθεί αυτός ο περιορισμός αναπτύχθηκαν πιο ισχυρά πλήρως συνδεδεμένα νευρωνικά δίκτυα με εμπρόσθια τροφοδότηση, τα πολυεπίπεδα perceptrons. Τα πολυεπίπεδα perceptrons υλοποιούνται στοιβάζοντας απλά perceptron μεταξύ τους, και χρησιμοποιώντας μη γραμμικές συναρτήσεις ενεργοποίησης (π.χ. ReLU, sigmoid) μεταξύ των επιπέδων, παράγοντας έτσι μια μη γραμμική αντιστοίχιση της εισόδου στην έξοδο.

3.2 Συνελικτικά Νευρωνικά Δίκτυα (CNNs)

Το πολυεπίπεδο perceptron αν και ανταποκρίνεται πολύ αποτελεσματικά σε σύνθετα προβλήματα φαίνεται να έχει κάποιους περιορισμούς. Όντας ένα πλήρως συνδεδεμένο δίκτυο, κάθε νευρώνας ενός επιπέδου είναι συνδεδεμένος με όλους τους νευρώνες του επόμενου. Αυτό καθιστά την λειτουργία του πολύ υπολογιστικά κοστοβόρα [17], έχοντας μάλιστα πλεονασμό από συνδέσεις μεταξύ νευρώνων αφού δεν έχουν πάντα όλα τα χαρακτηριστικά εισόδου υψήλη συσχέτηση μεταξύ τους. Επίσης, αυτή η πληθώρα συνδέσεων συνεπάγεται αυξήμενη πολυπλοκότητα του μοντέλου και κατ' επέκταση αυξημένη δυσκολία γενίκευσης [18].



Πηγή: TowardsDataScience
Σχήμα 3.3: Αρχιτεκτονική Συνελικτικού Νευρωνικού Δικτύου (CNN)

Τα Συνελικτικά Νευρωνικά Δίκτυα (CNNs) [19, 20, 21, 22] είναι μια κατηγορία νευρωνικών δικτύων με εμπρόσθια τροφοδότηση που ξεπερνούν αυτούς τους δύο περιορισμούς. Τα CNNs είναι δικτύα των οποίων τουλάχιστον ένα επίπεδο είναι συνελικτικό, δηλαδή εφαρμόζει την πράξη της συνέλιξης μεταξύ της εισόδου και ενός φίλτρου βαρών κοινό για όλα τα χαρακτηριστικά της

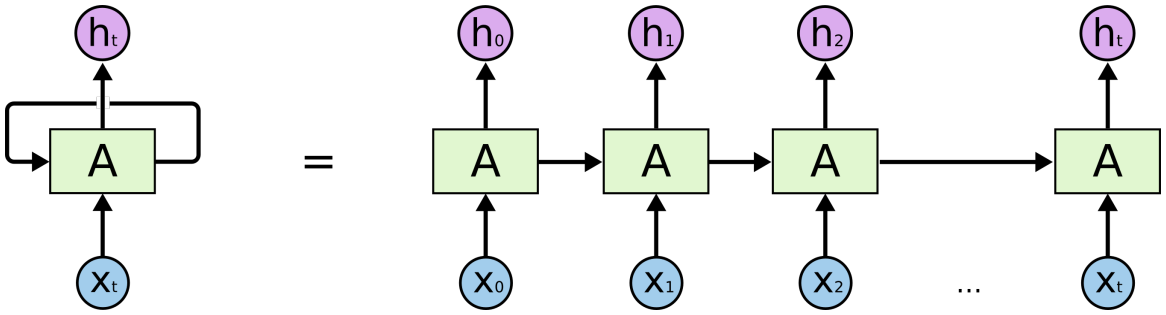
εισόδου (αναφορικά με το συγκεκριμένο επίπεδο). Συνήθως, στην έξοδο του συνελικτικού επιπέδου εφαρμόζεται μια μη γραμμική συνάρτηση ενεργοποίησης (στα CNNs χρησιμοποιείται κυρίως η ReLU), ακολουθούμενη μερικές φορές από ένα επίπεδο υποδειγματοληψίας στο οποίο θα αναφερόμαστε ως μηχανισμός pooling. Έτσι, ξεπερνάται το πρόβλημα της πλήρους συνδεσιμότητας με μια απόπειρα δημιουργίας δικτύων με αραιή και πιο «εύστοχη» συνδεσιμότητα. Τα CNNs συνήθως χρησιμοποιούνται σε προβλήματα επεξεργασίας εικόνων και όρασης υπολογιστών. Εκμεταλλευόμενα την μεγάλη πιθανότητα υψηλής συσχέτησης που έχουν τα pixels μιας περιοχής της εικόνας μεταξύ τους, πετυχαίνουμε την μείωση των παραμέτρων και της πολυπλοκότητας του μοντέλου, και κατ' επέκταση δίνεται η δυνατότητα για την κατασκευή βαθύτερων δικτύων. Μπορούμε λοιπόν να σκεφτούμε τα CNNs ως δίκτυα στα οποία εμπεριέχεται τουλάχιστον ένα συνελικτικό επίπεδο. Ένα συνελικτικό επίπεδο αποτελείται από έναν πλήθος φίλτρων βαρών k (συνήθως μικρών διαστάσεων π.χ. 3x3, 5x5) που καθένα απ' αυτά διασχίζει με κάποιο βήμα (stride) την εικόνα I σαν κυλιόμενο παράθυρο εφαρμόζωντας έτσι την πράξη της συνέλιξης[1] [14]:

$$(I * k)(i, j) = \sum_m \sum_n I(i + m, j + n) \cdot k(m, n) \tag{3.2}$$

Λόγω των κοινών βαρών (weight sharing), το φίλτρο αναζητά στην εικόνα ένα συγκεκριμένο μοτίβο το οποίο μπορεί να εντοπίσει σε πολλά διαφορετικά μέρη της εικόνας κατά το πέρασμα του. Αυτό σε συνδυασμό με τον μηχανισμό pooling, καθιστούν τα CNNs αμετάβλητα κατά τη μετατόπιση (translation-invariant). Γι' αυτό το λόγο τα CNNs είναι πολύ αποτελεσματικά στην ανίχνευση και αναγνώριση αντικειμένων, καθώς εκμεταλλεύονται την μεγάλη πιθανότητα υψηλής συσχέτησης που έχουν τα pixels μιας περιοχής της εικόνας μεταξύ τους.

3.3 Αναδρομικά Νευρωνικά Δίκτυα (RNNs)

Αν και τα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης είναι πολύ αποτελεσματικά, μπορούμε εύκολα να παρατηρήσουμε ότι επεξεργάζονται κάθε μέρος της ακολουθίας εισόδου ξεχωριστά. Αυτό αποτελεί πρόβλημα για την επίλυση συγκεκριμένων κατηγοριών προβλημάτων όπως είναι τα προβλήματα χρονοσειρών, επεξεργασίας φυσικής γλώσσας κλπ. Σε τέτοιες κατηγορίες προβλημάτων τα «συμφραζόμενα» διαδραματίζουν πολύ σημαντικό ρόλο. Έτσι, σχεδιάστηκαν αρχιτεκτονικές όπως αυτή που φαίνεται στο Σχήμα 3.4, που λαμβάνουν υπόψη τα «συμφραζόμενα», τα Αναδρομικά Νευρωνικά Δίκτυα (RNNs). Τα RNNs [23] είναι μια επέκταση της ευρύτερης κατηγορίας των νευρωνικών δικτύων με εμπρόσθια τροφοδότηση, προσθέτοντας μηχανισμούς ανατροφοδότησης [14].



Πηγή: Artificial Intelligence in Plain English
Σχήμα 3.4: Αρχιτεκτονική Αναδρομικού Νευρωνικού Δικτύου (RNN)

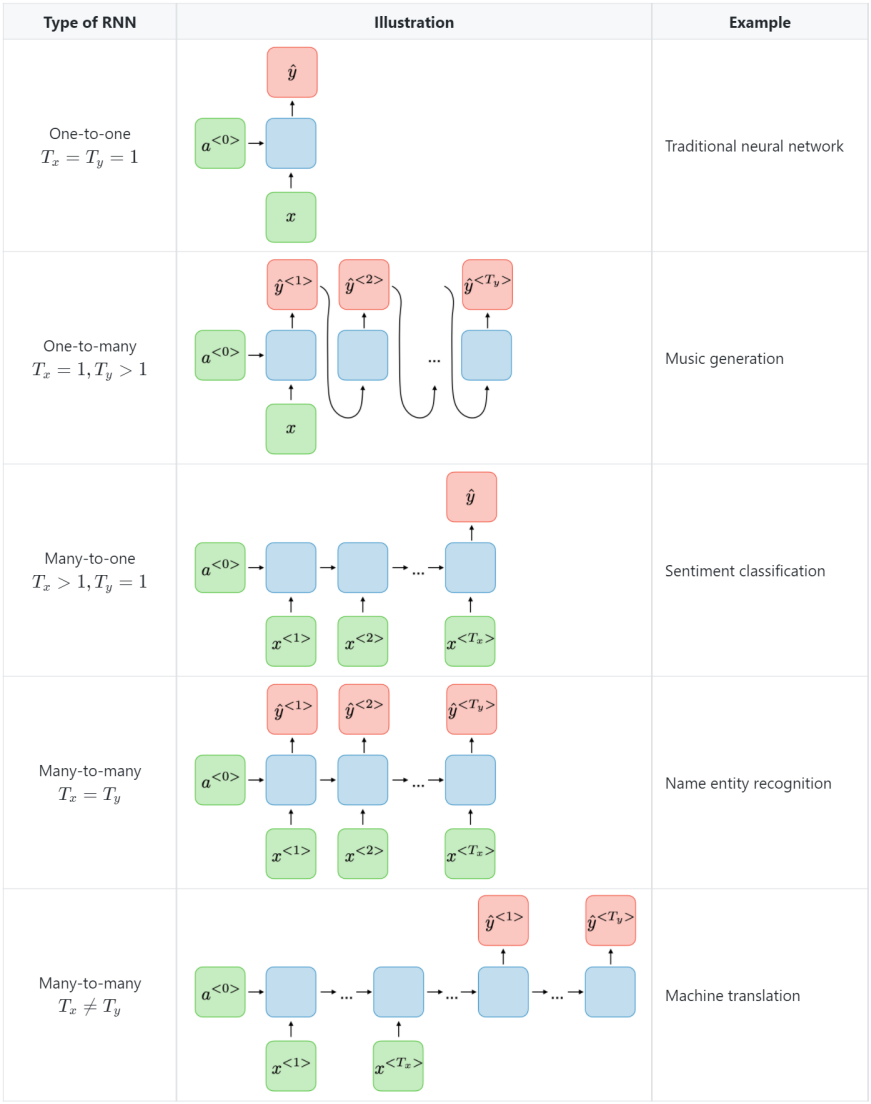
[1] Η σχέση (3.2) δεν αποτελεί τον πραγματικό ορισμό της πράξης της συνέλιξης, πρόκειται για μια παρόμοια συνάρτηση που ονομάζεται **cross-correlation**. Όπως συνηθίζεται στον χώρο της Μηχανικής Μάθησης, θα κάνουμε την σύμβαση με τον όρο "συνέλιξη" να αναφερόμαστε στο cross-correlation.

Τα RNNs, όπως και τα νευρωνικά δίκτυα με εμπρόσθια τροφοδότηση, έχουν κρυφά επίπεδα. Ωστόσο, τα κρυφά επίπεδα των RNNs έχουν συνδέσεις πίσω στον εαυτό τους, επιτρέποντας στις καταστάσεις των κρυφών επιπέδων τη στιγμή t να χρησιμοποιηθούν ως είσοδος στα κρυφά επίπεδα τη στιγμή $t+1$, γεγονός που τα καθιστά δίκτυα με «μνήμη», επιτρέποντας στις κρυφές καταστάσεις να καταγράφουν πληροφορίες σχετικά με την χρονική συσχέτιση μεταξύ των ακολουθιών εισόδου και εξόδου. Ένα σύνολο εξισώσεων (3.3) που περιγράφει τους υπολογισμούς που πραγματοποιεί ένα απλό RNN φαίνεται παρακάτω:

$$\begin{aligned} h^t &= g(\boldsymbol{h^{t-1}}, \boldsymbol{x^t}; \boldsymbol{\theta}) \\ o^t &= f(\boldsymbol{h^t}; \boldsymbol{\theta}) \end{aligned} \tag{3.3}$$

όπου o^t είναι η έξοδος του RNN τη στιγμή t, x^t είναι η t-οστό στοιχείο της ακολουθίας εισόδου x και h^t είναι η κατάσταση του κρυφού επιπέδου τη στιγμή t, και η εξαρτημένη μεταβλήτη θ αποτελεί τις παραμέτρους του μοντέλου.

Αξίζει τέλος να σημειωθεί πως, ενώ τα νευρωνικά δίκτυα εμπρόσθιας τροφοδότησης αντιστοιχίζουν μια είσοδο σε μια έξοδο, τα RNNs, ανάλογα με το τι πρόβλημα καλούνται να επιλύσουν, μπορούν να αντιστοιχίσουν ένα-σε-πολλά, πολλά-σε-ένα και πολλά-σε-πολλά όπως παρουσιάζεται στο Σχήμα 3.5.



Πηγή: Stanford
Σχήμα 3.5: Τύποι RNN

3.4 Νευρωνικά Δίκτυα Αυτοκωδικοποίησης (Autoencoders)

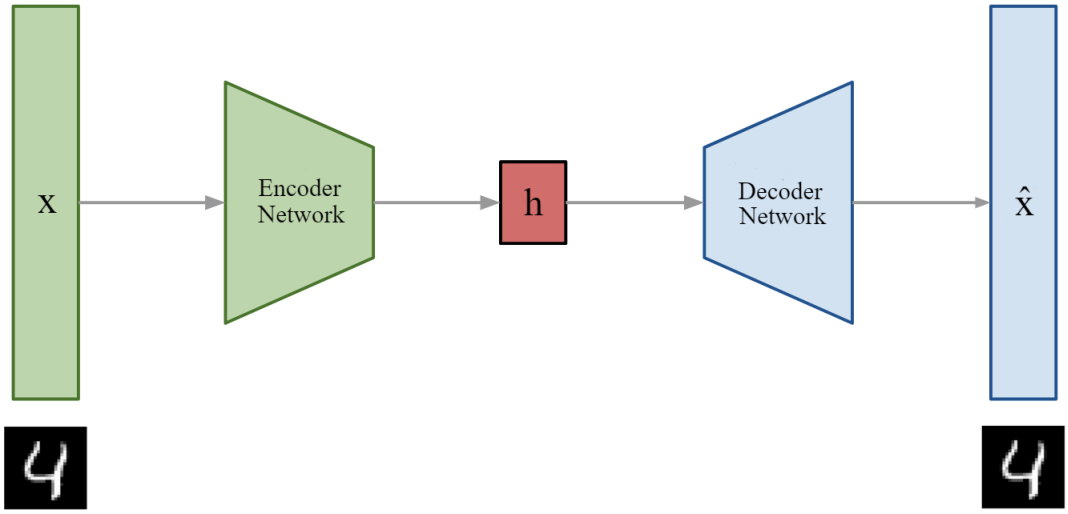
Χρησιμοποιώντας τα παραπάνω είδη δικτύων ή μέρη τους ως δομικά στοιχεία, μπορούμε να κατασκευάσουμε μια νέα υποκατηγορία νευρωνικών δικτών, τα δίκτυα αυτοκωδικοποίησης ή αλλιώς autoencoders. Οι autoencoders [23, 24] είναι νευρωνικά δίκτυα μη-επιβλεπόμενης μάθησης, που έχουν ως στόχο την ανακασκευή της εισόδου x στην έξοδο $\hat{x}$ ελαχιστοποιώντας κάποιο σφάλμα ανακατασκευής $L(x, \hat{x})$ που αντιπροσωπεύει το κατά πόσο διαφέρει η πραγματική είσοδος από την ανακατασκευασμένη. Ένας autoencoder αποτελείται από δύο βασικά μέρη [25]:

1. **Κωδικοποιητής (Encoder)**: Αποτελεί έναν μηχανισμό κωδικοποίησης-συμπίεσης της πραγματικής είσοδου σε μια κρυφή αναπαράσταση $h = enc(x)$, μικρότερης συνήθως διαστατικότητας.

$$h = enc(x) \tag{3.4}$$

2. **Αποκωδικοποιητής (Decoder)**: Αποτελεί έναν μηχανισμό αποκωδικοποίησης που έχει ως στόχο την ανακατασκευή $\hat{x}$ της εισόδου από την κρυφή αναπαράσταση h.

$$\hat{x} = dec(h) \tag{3.5}$$



Σχήμα 3.6: Autoencoder για ανακατασκευή εικόνας από το MNIST Dataset

Κεφάλαιο 4

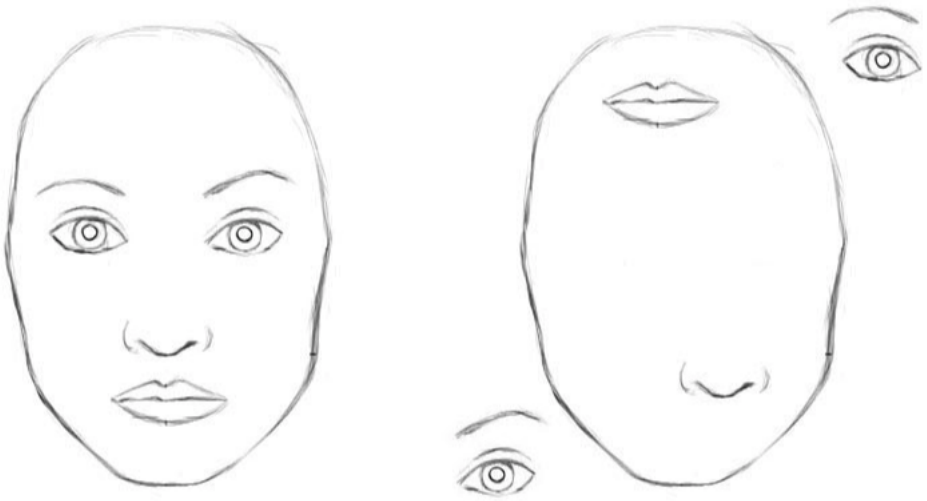
Νευρωνικά Δίκτυα με Κάψουλες

4.1 Αδυναμίες των Συνελικτικών Νευρωνικών Δικτύων

Τα CNNs είναι ένας από τους λόγους που η βαθιά μηχανική μάθηση είναι τόσο δημοφιλής σήμερα. Βέβαια, αν και επιτυγχάνουν εκπληκτικές επιδόσεις (ειδικά σε προβλήματα όρασης υπολογιστών), έχουν ορισμένες θεμελιώδεις αδυναμίες και μειονεκτήματα.

Το κύριο στοιχείο των CNNs είναι το συνελικτικό επίπεδο, το επίπεδο δηλαδή που εφαρμόζει την πράξη της συνέλιξης. Σε αυτό το επίπεδο ανιχνεύονται σημαντικά χαρακτηριστικά στα pixel μιας εικόνας. Τα πρώτα συνελικτικά επίπεδα που βρίσκονται κοντά στην είσοδο, μαθαίνουν να ανιχνεύουν απλά χαρακτηριστικά της εικόνας, όπως αμκές και χρωματικές διαβαθμίσεις. Τα συνελικτικά επίπεδα που ακολουθούν συνδυάζουν αυτά τα απλά χαρακτηριστικά, που έχουν ανιχνευθεί σε προηγούμενα επίπεδα, για τον εντοπισμό πιο συγκεκριμένων και σύνθετων χαρακτηριστικών της εικόνας. Τέλος, τα επίπεδα που βρίσκονται κοντά στην έξοδο του δικτύου, ανιχνεύουν πολύ υψηλού επιπέδου χαρακτηριστικά τέτοια ώστε να μπορούν να πραγματοποιήσουν προβλέψεις για την έξοδο. Τα υψηλότερου επιπέδου χαρακτηριστικά συνδυάζουν αυτά των προηγούμενων επιπέδων ως ένα σταθμισμένο άθροισμα (weighted sum). Τα αποτελέσματα της συνάρτησης ενεργοποίησης των προηγούμενων επιπέδων, τροφοδοτούνται ως είσοδοι στους νευρώνες των επόμενων συνελικτικών επιπέδων, όπου πολλαπλασιάζονται με τα αντίστοιχα βάρη των φίλτρων, και αθροίζονται πριν περάσουν στο στάδιο της επόμενης μη γραμμικής ενεργοποίησης.

Ωστόσο σε αυτή τη διάταξη εντοπίζεται ένα σημαντικό πρόβλημα. Δεν υφίσταται πουθενά η θέσπιση περιστροφικής ή μετατοπιστικής σχέσης (rotational or translational) μεταξύ του συνόλου των απλούστερων χαρακτηριστικών τα οποία στην συνέχεια συνθέτουν χαρακτηριστικά υψηλότερου επιπέδου. Η προσέγγιση των CNN για την επίλυση αυτού του προβλήματος είναι η χρήση του μηχανισμού max-pooling ή διαδοχικών συνελικτικών επιπέδων έτσι ώστε να μειωθεί το χωρικό μέγεθος προς επεξεργασία των δεδομένων και κατ' επέκταση να αυξηθεί το "πεδίο όρασης" των νευρώνων των υψηλότερων επιπέδων, επιτρέποντας τους έτσι να ανιχνεύουν χαρακτηριστικά υψηλότερης τάξης σε μια μεγαλύτερη περιοχή της εικόνας εισόδου. Είναι γεγονός ότι το max-pooling είναι ένας μηχανισμός που έκανε τα συνελικτικά νευρωνικά δίκτυα να λειτουργούν εκπληκτικά καλά, επιτυγχάνοντας υπεράνθρωπες επιδόσεις σε πολλούς τομείς. Πρέπει ωστόσο να σημειωθεί ότι ο μηχανισμός αυτός ευθύνεται για την απώλεια πολύτιμης πληροφορίας. Σύμφωνα με τον Geoffrey Hinton "Ο μηχανισμός pooling που χρησιμοποιείται στα συνελικτικά νευρωνικά δίκτυα είναι ένα μεγάλο λάθος και το γεγονός ότι λειτουργεί τόσο καλά είναι καταστροφή". Ασφαλώς και μπορούμε να καταργήσουμε τη χρήση του μηχανισμού pooling και να έχουμε και πάλι ικανοποιητικές επιδόσεις με τα παραδοσιακά συνελικτικά νευρωνικά δίκτυα, αλλά και πάλι δε θα λυθεί το κύριο πρόβλημα "Ο τρόπος με τον οποίο αναπαριστά εσωτερικά τα δεδομένα ένα συνελικτικό νευρωνικό δίκτυο, δε λαμβάνει υπόψη σημαντικές χωρικές ιεραρχίες-σχέσεις μεταξύ απλών και σύνθετων οντοτήτων."



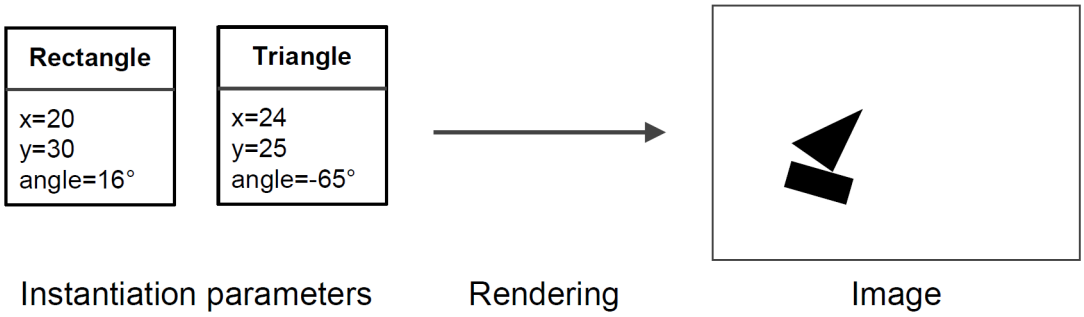
Πηγή: pechyonkin.me
Σχήμα 4.1: Για ένα CNN, και οι δύο εικόνες είναι παρόμοιες, αφού και οι δύο περιέχουν παρόμοια στοιχεία.

Ας εξετάσουμε το πολύ απλό και μη τεχνικό παράδειγμα ενός ανθρώπινου προσώπου. Ένα πρόσωπο είναι μια οντότητα σχήματος οβάλ η οποία κατά κύριο λόγο απαρτίζεται από δύο μάτια, δύο φρύδια, μια μύτη και ένα στόμα. Για ένα CNN, μια απλή παρουσία αυτών των αντικειμένων μπορεί να είναι ένας πολύ ισχυρός δείκτης για να θεωρήσει ότι υπάρχει ένα πρόσωπο στην εικόνα. Δεν δίνει δηλαδή ιδιαίτερη σημασία στον προσανατολισμό και στις σχετικές χωρικές σχέσεις των ανωτέρω χαρακτηριστικών, από τις οποίες προσδιορίζεται ορθά η ύπαρξη ή όχι ενός προσώπου.

Μπορεί να γίνει άμεσα αντιληπτό από το Σχήμα 4.1, πως μια απλή παρουσία δύο ματιών, δύο φρυδιών, ενός στόματος και μιας μύτης σε μια εικόνα δεν σημαίνει ότι υπάρχει πρόσωπο, καθώς πρέπει επίσης να γνωρίζουμε πως αυτά τα αντικείμενα είναι προσανατολισμένα μεταξύ τους.

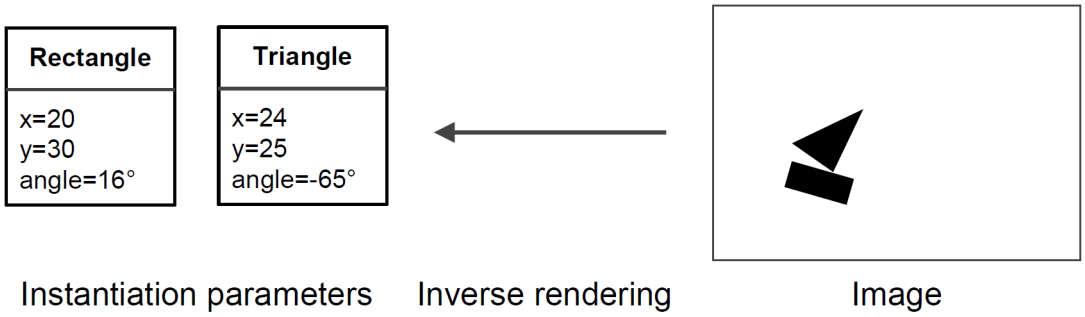
4.2 Επιστήμη των Γραφικών Υπολογιστή και Νευρωνικά Δίκτυα με Κάψουλες

Η επιστήμη των Γραφικών Υπολογιστή (Computer Graphics) αφορά κατά κύριο λόγο τη σύνθεση μιας εικόνας από κάποια εσωτερική ιεραρχική αναπαράσταση γεωμετρικών δεδομένων. Η δομή αυτής της αναπαράστασης λαμβάνει υπ' όψιν τις σχετικές θέσεις των αντικειμένων. Αυτές οι εσωτερικές αναπαραστάσεις αποθηκεύονται στην μνήμη των υπολογιστών ως συστοιχίες των γεωμετρικών αντικειμένων και ως μήτρες που αναπαριστούν τις σχετικές θέσεις και τον προσανατολισμό των αντικειμένων αυτών. Στην συνέχεια ειδικό λογισμικό μετατρέπει αυτήν την αναπαράσταση σε εικόνα στον υπολογιστή. Αυτή η διαδικασία ονομάζεται απόδοση εικόνας (rendering).



Πηγή: Capsule Networks, Aurélien Géron, 2017
Σχήμα 4.2: Διαδικασία απόδοσης εικόνας από κάποιες αρχικές παραμέτρους αναπαράστασης.

Ο ανθρώπινος εγκέφαλός ωστόσο λειτουργεί με την αντίστροφη διαδικασία σύμφωνα με την θεωρία των Νευρωνικών Δικτύων με Κάψουλες, την οποία ο Geoffrey Hinton ονομάζει αντεστραμμένα γραφικά (inverse graphics).



Πηγή: Capsule Networks, Aurélien Géron, 2017
Σχήμα 4.3: Αντεστραμμένη διαδικασία απόδοσης εικόνας. Από την αρχική εικόνα, γίνεται η αποσύνθεση των αρχικών παραμέτρων αναπαράστασης.

Από την αρχική οπτική πληροφορία που λαμβάνει ο εγκέφαλος από τα μάτια, αποσυνθέτει μια ιεραρχική αναπαράσταση του κόσμου τριγύρω και προσπαθεί να την συνδυάσει με ήδη γνωστά μοτίβα και σχέσεις αποθηκευμένες σε αυτόν. Με αυτόν τον τρόπο επιτυγχάνεται η αναγνώριση, και το κλειδί είναι πως η αναπαράσταση των αντικειμένων δεν εξαρτάται από την οπτική γωνία.

Στην ουσία τα Νευρωνικά Δίκτυα με Κάψουλες είναι νευρωνικά δίκτυα που προσπαθούν να εκτελέσουν την αντίστροφη διαδικασία της απόδοσης εικόνας. Επομένως για την υλοποίηση των Νευρωνικών Δικτύων με Κάψουλες, απαιτείται η μοντελοποίηση αυτών των ιεραρχικών σχέσεων, μέσα σε ένα νευρωνικό δίκτυο. Η επιστήμη των γραφικών υπολογιστή, δίνει τη λύση στο συγκεκριμένο πρόβλημα. Σε αυτήν, οι σχέσεις μεταξύ τρισδιάστατων αντικειμένων μπορούν να αναπαρασταθούν από την πόζα (pose), που στην ουσία είναι ο συνδυασμός της μετατόπισης (translation) με την περιστροφή (rotation).

Ένα κλασσικό πρόβλημα της όρασης υπολογιστών (computer vision), και της ρομποτικής (robotics), είναι η αναγνώριση συγκεκριμένων αντικειμένων σε μία εικόνα και ο καθορισμός της θέσης και του προσανατολισμού καθενός απ' αυτά, σε σχέση με κάποιο σύστημα συντεταγμένων. Η θέση (position) και ο προσανατολισμός (orientation) ενός αντικειμένου στον τρισδιάστατο χώρο, αναφέρονται μαζί με τον όρο πόζα (pose) ή γενικευμένη θέση του αντικειμένου.

Η θεωρία των Νευρωνικών δικτύων με Κάψουλες, υποστηρίζει πως προκειμένου να υπάρξει ορθή κατηγοριοποίηση (classification), και αναγνώριση αντικειμένων (object recognition), είναι σημαντικό να διατηρηθούν οι ιεραρχικές σχέσεις της πόζας, δηλαδή της θέσης και του προσανατολισμού ενός αντικειμένου στον τρισδιάστατο χώρο, μεταξύ τμημάτων των αντικειμένων.

Σε αυτό το σημείο έγκειται και η σημαντικότητα της συγκεκριμένης θεωρίας, καθώς ενσωματώνει σχετικές χωρικές θέσεις μεταξύ αντικειμένων οι οποίες αναπαρίστανται αριθμητικά ως ένας τετραδιάστατος πίνακας πόζας (4D pose matrix). Όταν οι παραπάνω προαναφερθείσες σχέσεις, δομηθούν σε μια εσωτερική αναπαράσταση δεδομένων, γίνεται πολύ εύκολο για ένα μοντέλο να κατανοήσει πως το αντικείμενο που βλέπει προβάλλεται απλώς από μια άλλη οπτική γωνία σε σχέση με κάποια που ενδεχομένως να έχει ξαναδεί στο παρελθόν.

Το ακόλουθο παράδειγμα ενισχύει την παραπάνω παραδοχή. Παρατηρώντας εικόνες του "Αγάλματος την Ελευθερίας" από διαφορετικές οπτικές γωνίες όπως φαίνεται στο Σχήμα 4.4, ο ανθρώπινος εγκέφαλος μπορεί εύκολα να αναγνωρίσει ότι πρόκειται για το ίδιο αντικείμενο. Αυτό συμβαίνει καθώς η εσωτερική αναπαράσταση του "Αγάλματος την Ελευθερίας" στον εγκέφαλο δεν εξαρτάται από την γωνία θέασής του. Πιθανόν να μην έχει ξαναδεί τις συγκεκριμένες εικόνες ποτέ, ωστόσο αμέσως αναγνωρίζει τι είναι αυτό που βλέπει.

Πηγή: pechyonkin.me
Σχήμα 4.4: Το Άγαλμα της Ελευθερίας από διαφορετικές οπτικές γωνίες.

Για ένα CNN, η αναγνώριση ενός αντικειμένου σε διαφορετικές οπτικές γωνίες από αυτές στις οποίες το έχει ήδη δει είναι πολύ δύσκολο, καθώς δεν έχει ενσωματωμένη την κατανόηση του τρισδιάστατου χώρου. Αντιθέτως, για ένα Νευρωνικό Δίκτυο με Κάψουλες είναι πολύ πιο εύκολο επειδή αυτές οι σχέσεις μοντελοποιούνται εκτενώς. Γι' αυτό το λόγο, τα Νευρωνικά Δικτύα με Κάψουλες χρειάζεται να χρησιμοποιήσουν μόνο ένα πολύ μικρό κομμάτι των δεδομένων που θα χρησιμοποιούσε ένα Συνελικτικό Νευρωνικό Δίκτυο. Η θεωρία των Νευρωνικών Δικτύων με Κάψουλες προσομοιάζει πολύ περισσότερο την λειτουργία του ανθρώπινου εγκεφάλου στην πράξη. Ο ανθρώπινος εγκέφαλος προκειμένου να μάθει να ξεχωρίζει να αναγνωρίζει κάποιες κατηγορίες αντικειμένων χρειάζεται να δει μόνον μερικές δεκάδες το πολύ εκατοντάδες παραδείγματα. Τα Συνελικτικά Νευρωνικά Δίκτυα από την άλλη χρειάζονται δεκάδες χιλιάδες παραδείγματα προκειμένου να επιτύχουν πολύ καλή επίδοση, γεγονός που αποτελεί μια αρκετά χρονοβόρα και μη βέλτιστη προσέγγιση, η οποία είναι πολύ κατώτερη ποιοτικά από αυτό που κάνει ο ανθρώπινος εγκέφαλος.

4.3 Διανυσματικές Κάψουλες και Δυναμική Δρομολόγηση με Συμφωνία

4.3.1 Εισαγωγή στις Διανυσματικές Κάψουλες και στην έννοια του "Ισομεταβλητού"

Προκειμένου να γίνει κατανοητό τι εννοούμε με τον όρο "κάψουλα" κρίνεται απαραίτητο να ανατρέξουμε στη δημοσίευση [2] — "Transforming Auto-encoders" — στην οποία και έγινε η πρώτη τους αναφορά από τον Geoffrey Hinton, της οποίας ορισμένες χρήσιμες προτάσεις παρατίθενται παρακάτω:

> "Αντί να στοχεύουμε στο αμετάβλητο της οπτικής γωνίας (viewpoint invariance) των νευρώνων, οι οποίοι χρησιμοποιούν μια βαθμωτή έξοδο για να περιγράψουν τη δραστηριότητα μιας γειτονιάς αναπαραγμένων ανιχνεύτων χαρακτηριστικών (feature detectors), τα τεχνητά νευρωνικά δίκτυα θα πρέπει να χρησιμοποιούν τοπικές "κάψουλες" οι οποίες να εκτελούν κάποιους περίπλοκους εσωτερικούς υπολογισμούς πάνω στις εισόδους τους και ύστερα να ενθυλακώνουν τα αποτελέσματα αυτών των υπολογισμών σε ένα μικρό διάνυσμα εξόδων πολύτιμης πληροφορίας. Κάθε κάψουλα μαθαίνει να αναγνωρίζει μια σιωπηρά καθορισμένη οπτική οντότητα σε ένα περιορισμένο εύρος οπτικών συνθηκών και παραμορφώσεων και εξάγει τόσο την πιθανότητα εύρεσης της οντότητας εντός του περιορισμένου πεδίου της, όσο και ένα σύνολο παραμέτρων αναπαράστασης που μπορεί να περιλαμβάνουν την ακριβή πόζα, τον φωτισμό και την παραμόρφωση της οντότητας σε σχέση με μια σιωπηρά καθορισμένη κανονική εκδοχή αυτής. Όταν η κάψουλα λειτουργεί σωστά, η πιθανότητα ύπαρξης της οπτικής οντότητας είναι τοπικά αμετάβλητη — δεν αλλάζει όσο η οντότητα κινείται πάνω στο πολύπτυχο μόρφωμα (manifold) των πιθανών μορφών εμφάνισης της εντός του περιορισμένου πεδίου που καλύπτεται από την κάψουλα. Αντιθέτως, οι παράμετροι αναπαράστασης, είναι "ισομεταβλητές" (equivariant) — όσο οι οπτικές συνθήκες αλλάζουν και η οντότητα κινείται πάνω στο πολύπτυχο μόρφωμα των μορφών εμφάνισης της, οι παράμετροι αναπαράστασης αλλάζουν αντιστοίχως καθώς αντιπροσωπεύουν τις εσωτερικές συντεταγμένες της οντότητας στο πολύπτυχο μόρφωμα των μορφών εμφάνισης της."

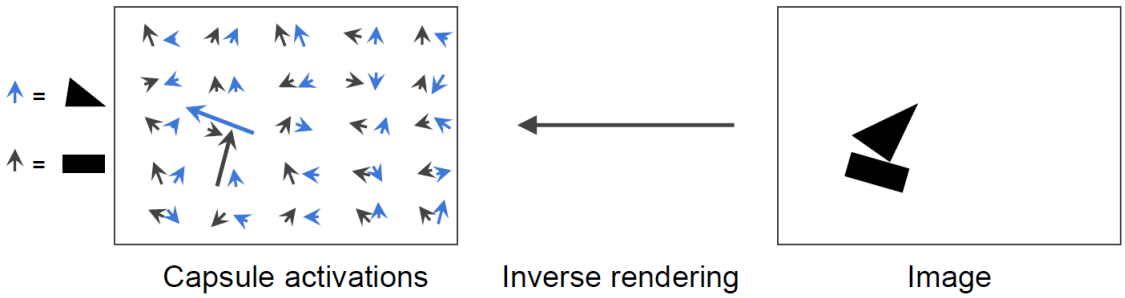
Προκειμένου να γίνει πιο κατανοητή η ουσία της παραπάνω παραγράφου, θα προσπαθήσουμε να την αναπαράγουμε και να αναλύσουμε με πιο απλά λόγια: Κάθε τεχνητός νευρώνας έχει βαθμωτή έξοδο. Επιπλέον, τα CNNs χρησιμοποιούν συνελικτικά επίπεδα τα οποία, για κάθε πυρήνα, αναπαράγουν τα ίδια βάρη αυτού πυρήνα σε όλο τον όγκο εισόδου και εκ των υστέρων δίνουν ως έξοδο μια δισδιάστατη μήτρα, της οποίας κάθε στοιχείο είναι η έξοδος της συνέλιξης αυτού του πυρήνα με το εκάστοτε μέρος του όγκου εισόδου. Μπορούμε, λοιπόν, να δούμε αυτό τη δισδιάστατη μήτρα ως έξοδο αναπαραγομένου ανιχνευτή χαρακτηριστικών (feature detector). Ακολούθως, οι δισδιάστατες μήτρες όλων των πυρήνων στοιβάζονται η μία πάνω στην άλλη παράγοντας την συνολική έξοδο ενός συνελικτικού επιπέδου. Έπειτα, προσπαθούμε να επιτύχουμε το αμετάβλητο της οπτικής γωνίας (viewpoint invariance) για τις δραστηριότητες των νευρώνων. Αυτό επιτυγχάνεται, με χρήση του μηχανισμού max-pooling που εξετάζει διαδοχικά τις υποπεριοχές της προαναφερθείσας δισδιάστατης μήτρας και επιλέγει το μεγαλύτερο στοιχείο της εκάστοτε υποπεριοχής. Με τον όρο "αμετάβλητο" ή αλλιώς "αναλλοίωτο" εννοούμε ότι με μικρή αλλαγή της εισόδου, η έξοδος θα παραμείνει η ίδια. Με άλλα λόγια, άμα στην εικόνα μετατοπιστεί λίγο το αντικείμενο που επιθυμούμε να εντοπίσουμε, οι δραστηριότητες των νευρώνων δε θα αλλάξουν, γεγονός που οφείλεται στον μηχανισμό max-pooling. Έτσι το δίκτυο θα είναι και πάλι σε θέση να ανιχνεύσει το αντικείμενο. Ωστόσο, ο παραπάνω μηχανισμός, όπως έχουμε ξαναναφέρει, δεν είναι πολύ καλός καθώς το max-pooling έχει ως επίπτωση την απώλεια πολύτιμης πληροφορίας και επιπλέον δεν κωδικοποιεί τις σχετικές χωρικές σχέσεις μεταξύ χαρακτηριστικών. Πρέπει, λοιπόν, να χρησιμοποιηθούν κάψουλες καθώς αυτές θα ενθυλακώνουν,

σε διανυσματική μορφή (σε αντίθεση με τη βαθμώτη έξοδο των νευρώνων), όλη τη σημαντική πληροφορία σχετικά με την κατάσταση των χαρακτηριστικών που εντοπίζουν.

Στην πράξη, κάθε πραγματικό αντικείμενο αποτελείται από διάφορα μικρότερα μέρη ή αλλιώς οντότητες. Αυτές οι οντότητες του αντικειμένου σχηματίζουν μια συγκεκριμένη ιεραρχία. Παρατηρώντας ένα αντικείμενο, τα μάτια δημιουργούν κάποια σημεία σταθεροποίησης (fixation points), πάνω του. Οι σχετικές χωρικές θέσεις των σημείων αυτών και οι ιδιότητές τους, βοηθούν τον ανθρώπινο εγκέφαλο να αναγνωρίσει το αντικείμενο αυτό, χωρίς να χρειάζεται να επεξεργαστεί κάθε μικρή λεπτομέρεια.

4.3.2 Η Μη-Γραμμική Διανυσματική Ενεργοποίηση squash

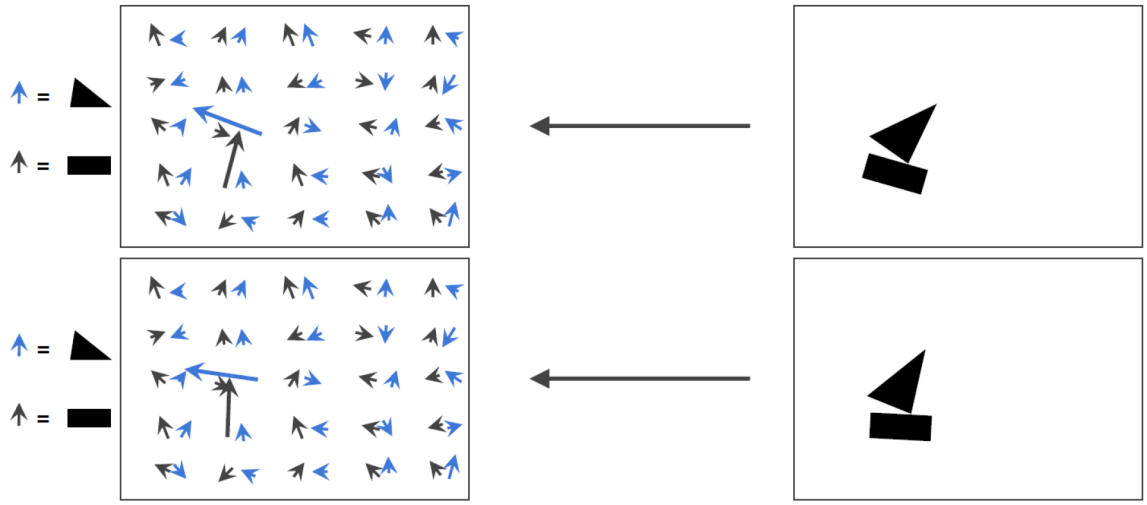
Στην πρώτη προσέγγιση [3] που θα αναλύσουμε, ο Geoffrey Hinton αναφέρεται στον όρο "κάψουλα", ως μία ομάδα από νευρώνες, της οποίας το διάνυσμα δραστηριότητας (activity vector) αναπαριστά τις παραμέτρους αναπαράστασης (instantiation parameters) μιας οντότητας συγκεκριμένου τύπου, όπως ένα μέρος ενός αντικειμένου ή και ένα ολόκληρο αντικείμενο. Ως δραστηριότητα ορίζεται απλώς το σήμα εξόδου της κάψουλας. Οι κάψουλες ανιχνεύουν την ύπαρξη συγκεκριμένων οντοτήτων σε μία εικόνα. Το μήκος του διανύσματος δραστηριότητας της κάψουλας αντιπροσωπεύει την πιθανότητα ύπαρξης της οντότητας και ο προσανατολισμός του τις παραμέτρους αναπαράστασης όπως φαίνεται και στο Σχήμα 4.5.



Πηγή: Capsule Networks, Aurélien Géron, 2017

Σχήμα 4.5: Η εικόνα στα δεξιά μιας βάρκας που αποτελείται από δύο οντότητες, τρίγωνο (μπλε) και ορθογώνιο (μαύρο), οι οποίες αναπαρίστανται στα αριστερά από τα αντίστοιχα διανύσματα ενεργοποίησης.

Όταν το ανιχνευόμενο χαρακτηριστικό κινείται μέσα στην εικόνα, η με κάποιον τρόπο αλλάζει η κατάσταση του, η πιθανότητα ύπαρξης του παραμένει σταθερή (το μήκος του διανύσματος δεν αλλάζει). Ωστόσο ο προσανατολισμός του αλλάζει, καθώς οι παράμετροι αναπαράστασής του μεταβάλλονται κατά ένα αντίστοιχο ποσό (Σχήμα 4.6). Με άλλα λόγια, η παρούσα προσέγγιση κωδικοποιεί το "ισομεταβλητό" της οπτικής γωνίας (viewpoint equivariance).



Πηγή: Capsule Networks, Aurélien Géron, 2017

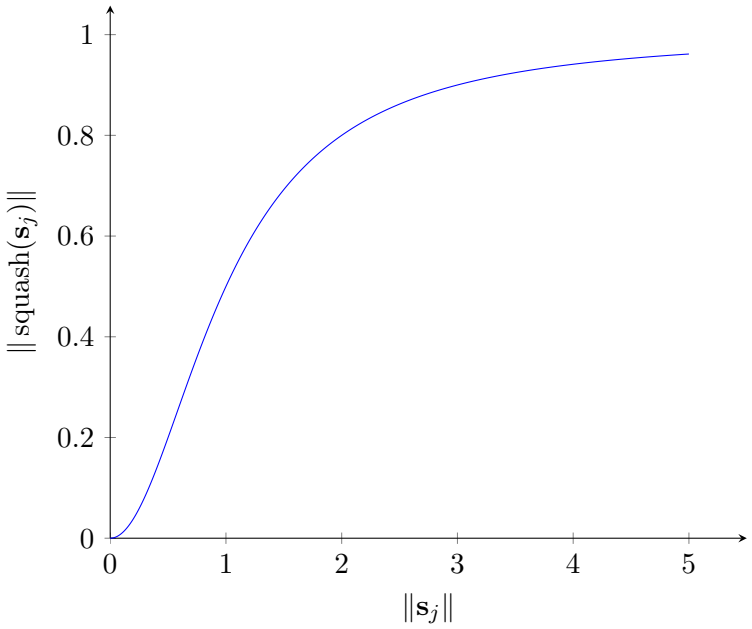
Σχήμα 4.6: Οι πιθανότητες ύπαρξης των οντοτήτων του τριγώνου και του ορθογωνίου, δηλαδή τα μήκη των διανυσμάτων δραστηριοτήτων τους (μπλε και μαύρο αντίστοιχα), παραμένουν σταθερά (invariance). Ωστόσο, καθώς το αντικείμενο στις εικόνες δεξιά κινείται, ο προσανατολισμός των διανυσμάτων αυτών, δηλαδή οι παράμετροι αναπαράστασής τους, μεταβάλλονται αντίστοιχα (equivariance).

Κάθε κάψουλα εκπροσωπεί έναν τύπο οντότητας, και το μέτρο του διανύσματος εξόδου της επιθυμούμε να αντιπροσωπεύει την πιθανότητα ύπαρξης αυτής της οντότητας στην παρούσα είσοδο. Έστω $\mathbb{R}^n$ ο διανυσματικός χώρος που αποτελείται από όλα τα n-διάστατα πραγματικά διανύσματα, και έστω $\mathrm{V}_1^n \subset \mathbb{R}^n$ ο διανυσματικός χώρος με όλα τα n-διάστατα πραγματικά διανύσματα $\mathbf{v}$ για τα οποία ισχύει $\|\mathbf{v}\| < 1$ ορίζεται μια νέα μη-γραμμική συνάρτηση squash : $\mathbb{R}^n \to \mathrm{V}_1^n$ η οποία συμπιέζει τα μικρού μήκους διανύσματα σε περίπου μηδενικό μήκος, ενώ συμπιέζει τα μεγάλου μήκους διανύσματα σε μήκος κατά κάτι λιγότερο του μοναδιαίου. Ο τύπος αυτής της μη γραμμικής συνάρτησης συμπίεσης (squashing function) ορίζεται ως:

$$\mathbf{v}_j = \mathrm{squash}(\mathbf{s}_j) = \frac{\|\mathbf{s}_j\|^2}{1 + \|\mathbf{s}_j\|^2} \cdot \frac{\mathbf{s}_j}{\|\mathbf{s}_j\|} \tag{4.1}$$

Στην εξίσωση (4.1), $\mathbf{v}_j$ είναι η έξοδος του διανύσματος της κάψουλας j και $\mathbf{s}_j$ είναι η συνολική είσοδός της. Αυτός ο μη γραμμικός μετασχηματισμός, η συνάρτηση συμπίεσης, λειτουργεί ως η αντίστοιχη συνάρτηση ενεργοποίησης στα Νευρωνικά Δίκτυα με Κάψουλες. Το δεξί μέρος της εξίσωσης κανονικοποιεί (unit scaling) το διάνυσμα εισόδου ώστε να έχει μοναδιαίο μήκος, και το αριστερό μέρος εφαρμόζει επιπρόσθετη συμπίεση (additional squashing). Το μήκος του διανύσματος εξόδου $\mathbf{v}_j$ μπορεί να ερμηνευθεί ως η πιθανότητα ενός δοθέντος χαρακτηριστικού να ανιχνευθεί από την κάψουλα.

$$\mathbf{v}_j = \underbrace{\frac{\|\mathbf{s}_j\|^2}{1+\|\mathbf{s}_j\|^2}}_{\text{additional "squashing"}} \underbrace{\frac{\mathbf{s}_j}{\|\mathbf{s}_j\|}}_{\text{unit scaling}}$$



Σχήμα 4.7: Γράφημα της νέας μη γραμμικότητας squash(·) στη βαθμωτή μορφή της. Αναπαρίσταται το μήκος του διανύσματος εξόδου σε σχέση με το μήκος του διανύσματος εισόδου.

4.3.3 Το Επίπεδο Καψουλών και ο αλγόριθμος Δυναμικής Δρομολόγησης με Συμφωνία

Για όλες τις κάψουλες, εκτός από αυτές στο πρώτο επίπεδο, η συνολική είσοδος σε μια κάψουλα $\mathbf{s}_j$, είναι το σταθμισμένο άθροισμα, όλων των διανυσμάτων πρόβλεψης (prediction vectors) $\hat{\mathbf{u}}_{j|i}$ από τις κάψουλες του προηγούμενου κατώτερου επιπέδου. Το διάνυσμα πρόβλεψης $\hat{\mathbf{u}}_{j|i}$ μιας κάψουλας κατώτερου επιπέδου υπολογίζεται πολλαπλασιάζοντας την έξοδό της $\mathbf{u}_i$, με έναν πίνακα αφινικού μετασχηματισμού (affine transformation matrix) $\mathbf{W}_{ij}$.

$$\hat{\mathbf{u}}_{j|i} = \mathbf{W}_{ij}\mathbf{u}_i \tag{4.2}$$

$$\mathbf{s}_j = \sum_i c_{ij}\hat{\mathbf{u}}_{j|i} \tag{4.3}$$

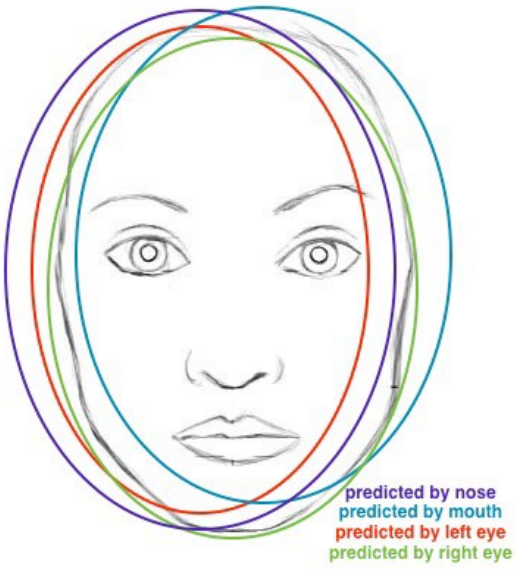
Ως c_{ij} ορίζονται οι συντελεστές σύζευξης (coupling coefficients), οι οποίοι υπολογίζονται από την επαναληπτική διαδικασία της Δυναμικής Δρομολόγησης (dynamic routing). Το άθροισμα των συντελεστών σύζευξης μιας κάψουλας i με όλες τις κάψουλες του επόμενου ανώτερου επιπέδου είναι ίσο με 1.

Οι συντελεστές σύζευξης c_{ij} υπολογίζονται από μία δρομολογημένη συνάρτηση softmax (routing softmax), της οποίας οι συντελεστές b_{ij} αντιπροσωπεύουν την λογαριθμική αρχική πιθανότητα η κάψουλα i (του επιπέδου l) να πρέπει να συνδυαστεί με την κάψουλα j (του επιπέδου $l+1$). Είναι ένας δείκτης του κατά πόσο η ύπαρξη της κάψουλας j οφείλεται στην κάψουλα i. Εξαρτώνται από τη θέση και το είδος των δύο καψουλών i και j, αλλά όχι και από την τρέχουσα εικόνα εισόδου.

$$c_{ij} = \frac{e^{b_{ij}}}{\sum_k e^{b_{ik}}} \tag{4.4}$$

Οι συντελεστές b_{ij}, όπως και όλα τα υπόλοιπα βάρη $\mathbf{W}_{ij}$, υπολογίζονται και ανανεώνονται κατά την εκπαίδευση του δικτύου. Συγκεκριμένα τα b_{ij} υπολογίζονται κατά τη διάρκεια της επαναληπτικής διαδικασίας της Δυναμικής Δρομολόγησης (Dynamic Routing). Οι αρχικές τιμές

των συντελεστών σύζευξης καθορίζονται επαναληπτικά μετρώντας την "συμφωνία" μεταξύ της παρούσας εξόδου $\mathbf{v}_j$ κάθε κάψουλας j, και της πρόβλεψης $\hat{\mathbf{u}}_{j|i}$ που πραγματοποιείται από την κάψουλα i. Με τον όρο "συμφωνία" εννοούμε απλά το βαθμωτό γινόμενο $\mathbf{a}_{ij} = \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$. Αυτή τη "συμφωνία" τη μεταχειριζόμαστε ως λογαριθμική πιθανοφάνεια (log-likelihood) και την προσθέτουμε στα αρχικό b_{ij}, προτού υπολογιστούν οι καινούριες τιμές για όλους τους συντελεστές σύζευξης που συνδέουν την κάψουλα i με αυτές των υψηλότερων επιπέδων.



Πηγή: pechyonkin.me

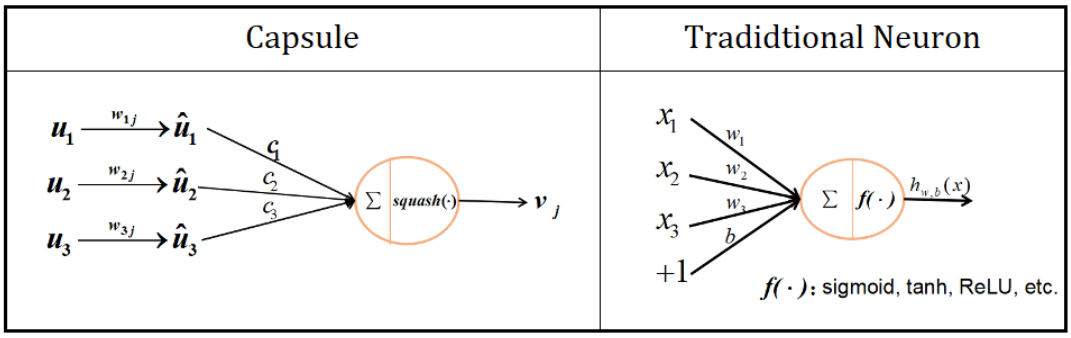
Σχήμα 4.8: Οι προβλέψεις της μύτης, του στόματος και των ματιών για την θέση του προσώπου συμφωνούν σε μεγάλο βαθμό. Είναι, λοιπόν, πολύ πιθανό εκεί να υπάρχει πρόσωπο.

Οι ενεργές κάψουλες ενός χαμηλότερου επιπέδου, παράγουν προβλέψεις $\hat{\mathbf{u}}_{j|i}$ μέσω πινάκων μετασχηματισμού $\mathbf{W}_{ij}$ (transformation matrices), για τις παραμέτρους αναπαράστασης από τις κάψουλες των υψηλότερων επιπέδων. Όταν πολλαπλές προβλέψεις συμφωνούν, μία κάψουλα υψηλότερου επιπέδου ενεργοποιείται. Μία κάψουλα χαμηλότερου επιπέδου προτιμά να στέλνει την έξοδό της σε εκείνες τις κάψουλες υψηλότερου επιπέδου των οποίων τα διανύσματα εξόδου $\mathbf{v}_j$ έχουν μεγάλο εσωτερικό γινόμενο με την πρόβλεψη προερχόμενη από την κάψουλα χαμηλότερου επιπέδου, δηλαδή το γινόμενο $\mathbf{a}_{ij} = \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$ είναι μεγάλο. Οι κάψουλες j του επιπέδου $l+1$, στέλνουν σήματα ανατροφοδότησης στις κάψουλες i του επιπέδου l. Εάν τα διανύσματα πρόβλεψης από τις κάψουλες i βρίσκονται σε συμφωνία με το διάνυσμα εξόδου από τις κάψουλες j, τότε όπως προαναφέρθηκε το γινόμενό τους θα πρέπει να είναι υψηλό. Επομένως και η βαρύτητα των προβλέψεων από τις κάψουλες i αυξάνεται αντίστοιχα και στα διανύσματα εξόδου από τις κάψουλες j. Αυτή η διαδικασία επαναλαμβάνεται. Η τελική επαναληπτική διαδικασία (Alg. 1) ονομάζεται Δυναμική Δρομολόγηση με Συμφωνία (Dynamic Routing by Agreement).

Αλγόριθμος 1 Δυναμική Δρομολόγηση με Συμφωνία

procedure DynamicRouting($\hat{\mathbf{u}}_{j|i}, r, l$):
  for all capsule i in layer l and capsule j in layer $(l+1)$: $b_{ij} \leftarrow 0$
  for r iterations **do**
    for all capsule i in layer l: $c_{ij} \leftarrow \text{softmax}(\mathbf{b}_i)$ ▷ softmax Equation (4.4)
    for all capsule j in layer $(l+1)$: $\mathbf{s}_j \leftarrow \sum_i c_{ij} \hat{\mathbf{u}}_{j|i}$
    for all capsule j in layer $(l+1)$: $\mathbf{v}_j \leftarrow \text{squash}(\mathbf{s}_j)$ ▷ squash Equation (4.1)
    for all capsule i in layer l and capsule j in layer $(l+1)$: $b_{ij} \leftarrow b_{ij} + \hat{\mathbf{u}}_{j|i} \cdot \mathbf{v}_j$
  return $\mathbf{v}_j$

Σε αυτό το σημείο είναι ενδιαφέρον, να συνοψίσουμε τις διαφορές (Σχήμα 4.9 και 4.10) μεταξύ μιας διανυσματικής κάψουλας και ενός απλού νευρώνα, τόσο ως προς τα χαρακτηριστικά τους, όσο και περί του τρόπου με τον οποίο χρησιμοποιμε τα δύο αυτά εργαλεία.



Πηγή: Medium
Σχήμα 4.9: Γραφική Σύγκριση Κάψουλας - Νευρώνα

Capsule vs. Traditional Neuron			
Input from low-level capsule/neuron		$\text{vector}(\mathbf{u}_i)$	$\text{scalar}(x_i)$
Operation	Affine Transform	$\hat{\mathbf{u}}_{j\|i} = \mathbf{W}_{ij}\mathbf{u}_i$	–
	Weighting	$\mathbf{s}_j = \sum_i c_{ij}\hat{\mathbf{u}}_{j\|i}$	$a_j = \sum_i w_i x_i + b$
	Sum		
	Nonlinear Activation	$\mathbf{v}_j = \frac{\|\mathbf{s}_j\|^2}{1+\|\mathbf{s}_j\|^2}\frac{\mathbf{s}_j}{\|\mathbf{s}_j\|}$	$h_j = f(a_j)$
Output		$\text{vector}(\mathbf{v}_j)$	$\text{scalar}(h_j)$

Πηγή: ResearchGate
Σχήμα 4.10: Σύγκριση Κάψουλας - Νευρώνα

4.3.4 Η συνάρτηση απωλειών MarginLoss

Όπως αναφέρθηκε και παραπάνω, το μήκος του διανύσματος αναπαράστασης αντιπροσωπεύει την πιθανότητα ύπαρξης της οντότητας μιας κάψουλας. Στην περίπτωση που αναφερόμαστε στο υψηλότερο/τελικό επίπεδο, το διάνυσμα αναπαράστασης της κάψουλας k αντιστοιχεί στην πιθανότητα ύπαρξης της υπ' αριθμόν k κατηγορίας. Η κάψουλα k θα πρέπει να έχει μεγάλου μέτρου διάνυσμα αναπαράστασης αν και μόνο αν στην εικόνα εισόδου είναι παρούσα οντότητα της κατηγορίας k. Προκειμένου να μοντελοποιηθεί αυτή η ιδέα και μάλιστα σε πολλαπλές κατηγορίες, εισήχθη μια νέα συνάρτηση απωλειών MarginLoss (Σχέση 4.5), η οποία εφαρμόζεται ξεχωριστά

σε κάθε κάψουλα κατηγοριών k.

$$L_k = \underbrace{T_k \max(0, m^+ - \|\mathbf{v}_k\|)^2}_{\text{class } k \text{ present}} + \underbrace{\lambda(1 - T_k) \max(0, \|\mathbf{v}_k\| - m^-)^2}_{\text{class } k \text{ not present}} \tag{4.5}$$

όπου $T_k = 1$ όταν πρόκειται για εικόνα που είναι γνωστό ότι περιέχει στιγμιότυπο της κατηγορίας k, αλλιώς $T_k = 0$. Η υπερπαράμετρος λ συνεισφέρει στη μείωση του βάρους των απωλειών που οφείλονται στις κατηγορίες που δεν είναι παρούσες στην τρέχουσα είσοδο, έτσι ώστε η αρχική εκπαίδευση να μην επιφέρει συρρίκνωση του μήκους των διανυσμάτων δραστηριότητας για όλες τις κάψουλες κατηγοριών. Τελικά, η συνολική απώλεια είναι απλά το άθροισμα των απωλειών όλων των καψουλών κατηγοριών.

$$L = \sum_k L_k \tag{4.6}$$

4.4 Κάψουλες Μήτρες και EM Δρομολόγηση με Συμφωνία

4.4.1 Εισαγωγή στις Κάψουλες Μήτρες

Μια κάψουλα είναι μια ομάδα νευρώνων των οποίων οι έξοδοι αντιπροσωπεύουν διαφορετικές ιδιότητες της ίδιας οντότητας. Κάθε επίπεδο σε ένα δίκτυο καψουλών περιέχει πολλές κάψουλες. Περιγράφουμε μια έκδοση καψουλών στην οποία κάθε κάψουλα έχει μια λογιστική μονάδα που αντιπροσωπεύει την πιθανότητα ύπαρξης μιας οντότητας και μια μήτρα 4x4 που θα μπορούσε να μάθει να αναπαριστά τη σχέση μεταξύ αυτής της οντότητας και του θεατή (την πόζα). Μια κάψουλα σε ένα επίπεδο "ψηφίζει" για τη μήτρα πόζας πολλών διαφορετικών καψουλών του αμέσως επόμενου επιπέδου πολλαπλασιάζοντας τη δική της μήτρα πόζας με εκπαιδεύσιμους πίνακες μετασχηματισμού αμετάβλητης οπτικής γωνίας με σκοπό να μάθουν να αναπαριστούν σχέσεις μέρους-όλου (part-whole relationships).

Πηγή: "Matrix Capsules with EM Routing" [6]
Σχήμα 4.11: Κάψουλα Μήτρα

Ένα Νευρωνικό Δίκτυο με Κάψουλες αποτελείται από έναν αριθμό επιπέδων με κάψουλες. Θα συμβολίζουμε το σύνολο των καψουλών στο επίπεδο L ως Ω_L. Σύμφωνα με την προσέγγιση του Geoffrey Hinton, όπως αυτή αναλύεται στην ερευνητική εργασία "Matrix Capsules with EM Routing" [6], κάθε κάθε κάψουλα αποτελείται από μια 4×4 μήτρα πόζας M και από μια πιθανότητα a. Αυτές οι κάψουλες είναι οι αντίστοιχες δραστηριότητες που υπάρχουν σε ένα κλασσικό νευρωνικό δίκτυο (εξαρτώνται από την είσοδο και δεν αποθηκεύονται). Μεταξύ κάθε κάψουλας i του επιπέδου L και j του επιπέδου $L + 1$ υπάρχει ένας 4×4 εκπαιδεύσιμος πίνακας μετασχηματισμού W_{ij}. Αυτοί οι πίνακες μετασχηματισμού είναι οι μόνες αποθηκευμένες παράμετροι και εκπαιδεύονται ξεχωριστά. Η μήτρα πόζας της κάψουλας i μετασχηματίζεται μέσω του W_{ij} έτσι ώστε να υποβάλει μια ψήφο $V_{ij} = M_i W_{ij}$ για την μήτρα πόζας της κάψουλας j. Οι

πόζες και οι πιθανότητες ενεργοποίησης όλων των καψουλών στο επίπεδο $L+1$ υπολογίζονται χρησιμοποιώντας μία μη γραμμική διαδικασία δρομολόγησης που δέχεται ως είσοδο τις ψήφους V_{ij} και τις πιθανότητες ενεργοποίησης a_i για κάθε $i \in \Omega_L$, $j \in \Omega_{L+1}$.

Αυτή η μη γραμμική διαδικασία δρομολόγησης αποτελεί μια τροποποιημένη έκδοση του αλγορίθμου Προσδοκίας-Μεγιστοποίησης ή αλλιώς EM (Expectation-Maximization algorithm). Προσαρμόζει επαναληπτικά τις μέσες τιμές, τις διασπορές και τις πιθανότητες ενεργοποίησης των καψουλών του $L+1$ επιπέδου και τις πιθανότητες ανάθεσης μεταξύ όλων των $i \in \Omega_L$, $j \in \Omega_{L+1}$.

4.4.2 Ο αλγόριθμος της EM Δρομολόγησης με Συμφωνία

Ας υποθέσουμε ότι έχει ήδη αποφασιστεί η πόζα και η πιθανότητα ενεργοποίησης για καθεμία από τις κάψουλες σε ένα επίπεδο L, και ότι τώρα πρέπει να αποφασιστεί το ποιες κάψουλες αυτού του επιπέδου θα ενεργοποιηθούν και πως θα αναθέσουμε κάθε ενεργοποιημένη κάψουλα του χαμηλότερου επιπέδου L σε κάποια κάψουλα του υψηλότερου επιπέδου $L+1$. Κάθε κάψουλα του υψηλότερου επιπέδου $L+1$ αντιστοιχεί σε μια Γκαουσιανή και η πόζα κάθε ενεργής κάψουλας του χαμηλότερου επιπέδου L (σε διανυσματική μορφή) αντιστοιχεί σε ένα σημείο δεδομένων (είναι ένα κλάσμα σημείου δεδομένων εάν είναι κάψουλα είναι μερικώς ενεργή). Χρησιμοποιώντας την αρχή του ελάχιστου μήκους περιγραφής έχουμε την επιλογή όταν αποφασίζουμε αν θα ενεργοποιηθεί ή όχι μια κάψουλα του υψηλότερου επιπέδου [6].

Επιλογή 0: Εάν δεν ενεργοποιηθεί, θα πρέπει να πληρώσουμε ένα σταθερό κόστος $-\beta_u$ ανά σημείο δεδομένων για την περιγραφή της πόζας όλων των καψουλών του χαμηλότερου επιπέδου που έχουν ανατεθεί στις κάψουλες του υψηλότερου επιπέδου. Αυτό το κόστος είναι η αρνητική λογαριθμική πυκνότητα πιθανότητας (negative log probability density) του σημείου δεδομένων κάτω από μια όχι κατάλληλη ομοιόμορφη αρχική κατανομή.

Επιλογή 1: Εάν ενεργοποιηθεί, θα πρέπει να πληρώσουμε ένα σταθερό κόστος $-\beta_a$ για την κωδικοποίηση τη μέση τιμή και τη διακύμανσή του και το γεγονός ότι είναι ενεργή, και στη συνέχεια πληρώνουμε επιπλέον κόστη, αναλογικά με τις πιθανότητες ανάθεσης, για την περιγραφή των αποκλίσεων μεταξύ των μέσων τιμών του χαμηλότερου επιπέδου και των τιμών που προβλέπονται για αυτές όταν η μέση τιμή της κάψουλας του υψηλότερου επιπέδου χρησιμοποιείται για την πρόβλεψή τους μέσω της αντίστροφης μήτρας μετασχηματισμού.

Ένας πολύ απλούστερος τρόπος για να υπολογιστεί το κόστος της περιγραφής ενός σημείου δεδομένων είναι να χρησιμοποιηθεί η αρνητική λογαριθμική πυκνότητα πιθανότητα της ψήφου αυτού του σημείου δεδομένων κάτω από την κατανομή Gauss προσαρμοσμένη σε οποιαδήποτε υψηλότερου επιπέδου κάψουλα ανατίθεται. Παρότι αυτή η μέθοδος έχει σφάλμα, χρησιμοποιείται γιατί απαιτεί πολύ μικρότερη υπολογιστική πολυπλοκότητα. Στην διαφορά κόστους μεταξύ της Επιλογής 0 και της Επιλογής 1, εφαρμόζεται στη συνέχεια σε κάθε επανάληψη η λογιστική συνάρτηση προκειμένου να προσδιοριστεί η πιθανότητα ενεργοποίησης της κάψουλας του υψηλότερου επιπέδου.

Χρησιμοποιώντας την αποδοτική προσέγγιση για την Επιλογή 1 παραπάνω, το αυξητικό κόστος για να εξηγήσουμε εξ ολοκλήρου ένα σημείο δεδομένων i χρησιμοποιώντας μια ενεργή κάψουλα j, που έχει έναν πίνακα συνδιακύμανσης ευθυγραμμισμένο ως προς τους άξονες, είναι απλώς το άθροισμα, πάνω σε όλες τις διαστάσεις, του κόστους εξήγησης για κάθε διάσταση h της ψήφου V_{ij}. Αυτό είναι απλά $-\ln P^h_{i|j}$, όπου $P^h_{i|j}$ είναι η πυκνότητα πιθανότητας της h συνιστώσας της διανυσματικής ψήφου V_{ij} στη j-οστή Γκαουσιανή για τη διάσταση h που έχει μέση τιμή μ^h_j (όπου μ_j η διανυσματική μορφή του j-οστού μητρώου πόζας M_j) και διακύμανση $(\sigma^h_j)^2$.

$$P^h_{i|j} = \frac{1}{\sqrt{2\pi(\sigma^h_j)^2}} \exp\left(- \frac{(V^h_{ij} - \mu^h_j)^2}{2(\sigma^h_j)^2} \right) \tag{4.7}$$

$$\ln\left(P_{i|j}^{h}\right) = -\frac{(V_{ij}^{h} - \mu_{j}^{h})^{2}}{2(\sigma_{j}^{h})^{2}} - \ln\left(\sigma_{j}^{h}\right) - \frac{\ln\left(2\pi\right)}{2} \tag{4.8}$$

Αθροίζοντας πάνω σε όλες τις κάψουλες του χαμηλότερου επιπέδου για τη διάσταση h του j έχουμε:

$$\begin{aligned} cost_{j}^{h} &= \sum_{i} -r_{ij} \ln\left(P_{i|j}^{h}\right) \\ &= \frac{\sum_{i} r_{ij}(V_{ij}^{h} - \mu_{j}^{h})^{2}}{2(\sigma_{j}^{h})^{2}} + \left(\ln\left(\sigma_{j}^{h}\right) + \frac{\ln\left(2\pi\right)}{2}\right)\sum_{i} r_{ij} \\ &= \left(\ln\left(\sigma_{j}^{h}\right) + \frac{1}{2} + \frac{\ln\left(2\pi\right)}{2}\right)\sum_{i} r_{ij} \end{aligned} \tag{4.9}$$

όπου το $\sum_{i} r_{ij}$ είναι η ποσότητα των δεδομένων που ανατέθηκαν στο j, και V_{ij}^{h} η τιμή του V_{ij} στη διάσταση h. Ενεργοποιώντας την κάψουλα j αυξάνεται το μήκος περιγραφής για τις μέσες τιμές των καψουλών του χαμηλότερου επιπέδου που έχουν ανατεθεί στην j, από $-\beta_u$ για κάθε κάψουλα του χαμηλότερου επιπέδου, μέχρι $-\beta_a$ συν το κόστος πάνω σε όλες τις διαστάσεις. Έτσι ορίζουμε την συνάρτηση ενεργοποίησης της κάψουλας j ως εξής:

$$a_{j} = logistic\left(\lambda\left(\beta_{a} - \beta_{u}\sum_{i} r_{ij} - \sum_{h} cost_{j}^{h}\right)\right) \tag{4.10}$$

Τα β_a και β_u μαθαίνονται ξεχωριστά και τίθεται ένα σταθερό πλάνο δρομολόγησης για τις τιμές της υπερπαραμέτρου λ.

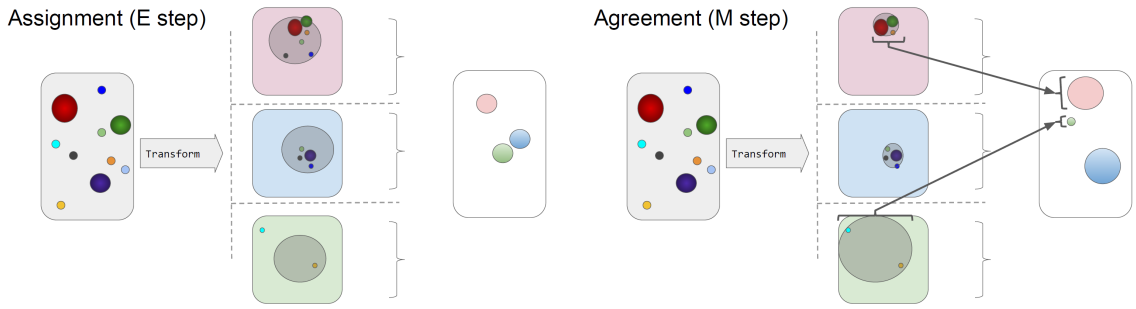
Για την οριστικοποίηση της πόζας και της πιθανότητας ενεργοποίησης κάθε κάψουλας στο επίπεδο $L+1$, εκτελείται αλγόριθμος EM για έναν αριθμό επαναλήψεων, αφότου έχουν ήδη οριστικοποιηθεί οι πόζες και οι πιθανότητες ενεργοποίησης των καψουλών στο επίπεδο L. Η μη γραμμικότητα που υλοποιείται από ένα επίπεδο καψουλών έχει τη μορφή μιας διαδικασίας συσταδοποίησης με χρήση του αλγορίθμου EM, και έτσι ονομάστηκε EM Δρομολόγηση (Alg. 2).

Αλγόριθμος 2 EM Δρομολόγηση

procedure EM ROUTING(V, a):
  $\forall i \in \Omega_L,\ j \in \Omega_{L+1} : R_{ij} \leftarrow 1/|\Omega_{L+1}|$
  for t iterations **do**
    $\forall j \in \Omega_{L+1}$: M-STEP($a, R, V, j$)
    $\forall i \in \Omega_L$: E-STEP(μ, σ, a, V, i)
  return a, M

procedure M-STEP(a, R, V, j): ▷ for one higher-level capsule, j
  $\forall i \in \Omega_L : R_{ij} \leftarrow R_{ij} \cdot a_i$
  $\forall h : \mu_{j}^{h} \leftarrow \frac{\sum_{i} R_{ij} V_{ij}^{h}}{\sum_{i} R_{ij}}$
  $\forall h : (\sigma_{j}^{h})^{2} \leftarrow \frac{\sum_{i} R_{ij}(V_{ij}^{h} - \mu_{j}^{h})^{2}}{\sum_{i} R_{ij}}$
  $cost^{h} \leftarrow (\beta_{u} + \log\left(\sigma_{j}^{h}\right)) \sum_{i} R_{ij}$
  $a_{j} \leftarrow logistic(\lambda(\beta_{a} - \sum_{h} cost^{h}))$

procedure E-STEP(μ, σ, a, V, i): ▷ for one lower-level capsule, i
  $\forall j \in \Omega_{L+1} : p_{j} \leftarrow \frac{1}{\sqrt{\prod_{h}^{H} 2\pi(\sigma_{j}^{h})^{2}}} \exp\left(-\sum_{h}^{H} \frac{(V_{ij}^{h} - \mu_{j}^{h})^{2}}{2(\sigma_{j}^{h})^{2}}\right)$
  $\forall j \in \Omega_{L+1} : R_{ij} \leftarrow \frac{a_{j} p_{j}}{\sum_{k \in \Omega_{L+1}} a_{k} p_{k}}$



(a) E-step (b) M-step

Πηγή: Introduction to Capsules, Sara Sabour

Σχήμα 4.12: Γραφική αναπαράσταση των βημάτων E-step και M-step της EM Δρομολόγησης

4.4.3 Η τεχνική της Προσθήκης Συντεταγμένων

Όπως προαναφέθηκε, το δίκτυο μπορεί να μάθει όποια χαρακτηριστικά φαίνεται ότι είναι πιο ταιριαστά. Μπορούμε, λοιπόν, να το προσαρμόσουμε ώστε να αναγνωρίζει συγκεκριμένα χαρακτηριστικά μέσω μιας τεχνικής που ονομάζεται "Προσθήκη Συντεταγμένων" (Coordinate Addition) αντί μιας διαδικασίας ανακατεσκευής.

Για να εξετάσουμε αν ένα δίκτυο έχει σωστή γνώση για τις συντεταγμένες (x, y) ενός αντικειμένου σε περίπτωση ορθής ανακατασκευής μιας εικόνας εισόδου, θα απαιτούσαμε να σχεδιαστεί το ίδιο αντικείμενο στις συντεταγμένες $(x+dx, y+dy)$. Αν η έξοδος ταιριάζει με την ίδια εικόνα εισόδου με το αντικείμενο μετατοπισμένο κατά τις ποσότητες (dx, dy), τότε θα ξέραμε ότι το δίκτυο έμαθε να αποθηκεύει την θέση του αντικειμένου ως ένα ζεύγος (x, y).

Η τεχνική, λοιπόν, της προσθήκης συντεταγμένων μπορεί να χρησιμοποιηθεί σε όλα τα επίπεδα καψουλών έτσι ώστε να συντονίσει όλες κάψουλες να παρακολουθούν χωρικές πληροφορίες. Στην ερευνητική εργασία του Geoffrey Hinton "Matrix Capsules with EM Routing", η προαναφερθείσα τεχνική υλοποιήθηκε ως η προσθήκη της θέσης του κέντρου του "πεδίου όρασης" του πυρήνα, στα δύο πρώτα στοιχείο της μήτρας πόζας της εξόδου [6].

4.4.4 Η συνάρτηση απωλειών SpreadLoss

Προκειμένου να μειωθεί η ευαισθησία της εκπαίδευσης τόσο ως προς την αρχικοποίηση όσο και τις υπερπαραμέτρους του μοντέλου, εισήχθη η συνάρτηση απωλειών SpreadLoss (Σχέση 4.11), έτσι ώστε να μεγιστοποιείται άμεσα το κενό μεταξύ της ενεργοποίησης της κατηγορίας που αναπαριστά η τρέχουσα εικόνα εισόδου και των ενεργοποιήσεων που αφορούν στις υπόλοιπες κατηγορίες [6]. Αν η ενεργοποίηση μιας λανθασμένης κατηγορίας a_i, είναι πιο κοντά στο a_t, από 'τι είναι το περιθώριο m, τότε τιμωρείται με το τετράγωνο της απόστασής της από το περιθώριο.

$$L_i = \max(0, m - (a_t - a_i))^2 \tag{4.11}$$

Τελικά, η συνολική απώλεια είναι απλά το άθροισμα των απωλειών όλων των καψουλών κατηγοριών εξαιρώντας την απώλεια που αφορά την κατηγορία που αναπαρίσταται στην τρέχουσα εικόνα εισόδου.

$$L = \sum_{i \neq t} L_i \tag{4.12}$$

Ξεκινώντας από μια χαμηλή τιμή για το περιθώριο m (π.χ. $m = 0.2$) και γραμμικά αυξάνοντας το κατά τη διάρκεια της εκπαίδευσης (π.χ. έως και $m = 0.9$), αποφεύγεται το πρόβλημα των "νεκρών" ή αλλιώς "ανενεργών" καψουλών στα αρχικά επίπεδα. Αξίζει να σημειωθεί ότι η συνάρτηση απωλειών SpreadLoss για $m = 1$ είναι ισοδύναμη με την τετραγωνική συνάρτηση απωλειών Hinge (Squared Hinge Loss).

Μέρος ΙΙ

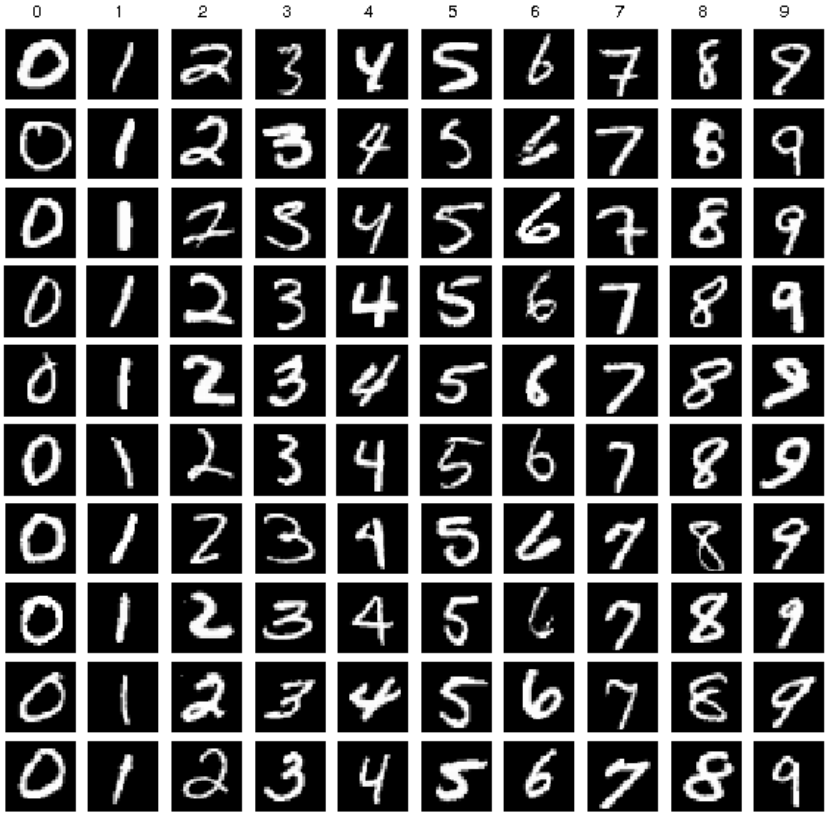
Πρακτικό Μέρος

Κεφάλαιο 5

Σύνολα Δεδομένων

5.1 Σύνολο MNIST

Το MNIST [26] είναι ένα σύνολο δεδομένων αποτελούμενο από χειρόγραφα ψηφία. Αποτελείται από ένα σύνολο εκπαίδευσης 60.000 παραδειγμάτων και ένα σύνολο ελέγχου 10.000 παραδειγμάτων. Είναι ένα υποσύνολο ενός μεγαλύτερου συνόλου που διατίθεται από το NIST [27]. Τα ψηφία έχουν κανονικοποιηθεί ως προς το μέγεθος και έχουν κεντραριστεί σε μια εικόνα σταθερού μεγέθους. Είναι ένα καλό σύνολο δεδομένων για τη δοκιμή τεχνικών εκμάθησης και μεθόδων αναγνώρισης προτύπων σε δεδομένα του πραγματικού κόσμου, δίχως να απαιτείται ιδιαίτερη προεπεξεργασία και μορφοποίηση. Οι αρχικές ασπρόμαυρες εικόνες από το NIST είχαν κανονικοποιηθεί σε μέγεθος ώστε να χωρούν σε πλαίσιο 20x20 pixel διατηρώντας παράλληλα την αναλογία διαστάσεων. Οι εικόνες που προκύπτουν περιέχουν γκρι pixel ως αποτέλεσμα της τεχνικής anti-aliasing που χρησιμοποιείται από τον αλγόριθμο κανονικοποίησης. Οι εικόνες κεντραρίστηκαν σε ένα πλαίσιο 28x28 υπολογίζοντας το κέντρο μάζας των pixel και μετατοπίζοντας την εικόνα έτσι ώστε να τοποθετηθεί αυτό το σημείο στο κέντρο του 28x28 πλέγματος.

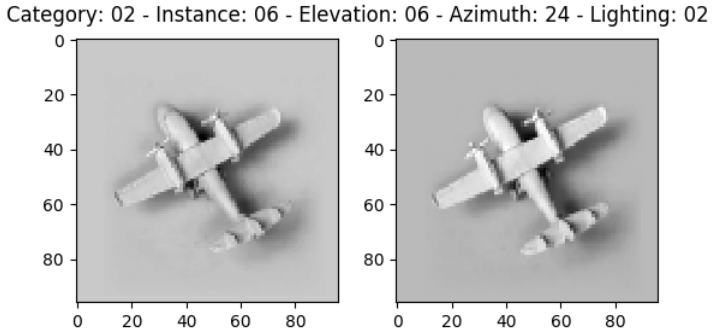


Πηγή: ResearchGate
Σχήμα 5.1: Σύνολο Δεδομένων MNIST

Το MNIST κατασκευάστηκε από το Special Database 3 και το Special Database 1 του NIST που περιέχουν δυαδικές εικόνες χειρόγραφων ψηφίων. Το NIST αρχικά όρισε το SD-3 ως σύνολο εκπαίδευσης και το SD-1 ως σύνολο ελέγχου. Ωστόσο, το SD-3 είναι πολύ πιο καθαρό και πιο εύκολο να αναγνωριστεί συγκριτικά με το SD-1. Αυτό πιθανότατα οφείλεται στο γεγονός ότι το SD-3 συγκεντρώθηκε από υπαλλήλους του Census Bureau, ενώ το SD-1 συγκεντρώθηκε από μαθητές γυμνασίου. Προκειμένου να εξαχθούν λογικά συμπεράσματα από πειράματα μάθησης, πρέπει το αποτέλεσμα να είναι ανεξάρτητο από την επιλογή του συνόλου εκπαίδευσης και του συνόλου ελέγχου από το πλήρες σύνολο των δειγμάτων. Ως εκ τούτου, ήταν απαραίτητο να δημιουργηθεί ένα νέο σύνολο δεδομένων με την ανάμειξη των διαφορετικών συνόλων που περιέχει το NIST. Το σύνολο εκπαίδευσης του MNIST αποτελείται από 30.000 δείγματα από το SD-3 και 30.000 δείγματα από το SD-1. Το σύνολο ελέγχου του MNIST αποτελείται από 5.000 δείγματα από το SD-3 και 5.000 δείγματα από το SD-1. Πολλές μέθοδοι έχουν δοκιμαστεί με αυτό το σύνολο δεδομένων. Εδώ είναι μερικά παραδείγματα. Μερικά από αυτά τα πειράματα χρησιμοποίησαν μια έκδοση της συνόλου δεδομένων όπου οι εικόνες εισόδου έγιναν diskewed (υπολογίζοντας τον κύριο άξονα του σχήματος που είναι πιο κοντά στην κατακόρυφο και μετατοπίζοντας τις γραμμές έτσι ώστε να γίνει κατακόρυφο). Σε ορισμένα άλλα πειράματα, το σετ εκπαίδευσης επαυξήθηκε με τεχνητά παραμορφωμένες εκδοχές των αρχικών δειγμάτων εκπαίδευσης. Οι παραμορφώσεις είναι τυχαίοι συνδυασμοί από shifts, scaling, skewing, and compression.

5.2 Σύνολο smallNORB

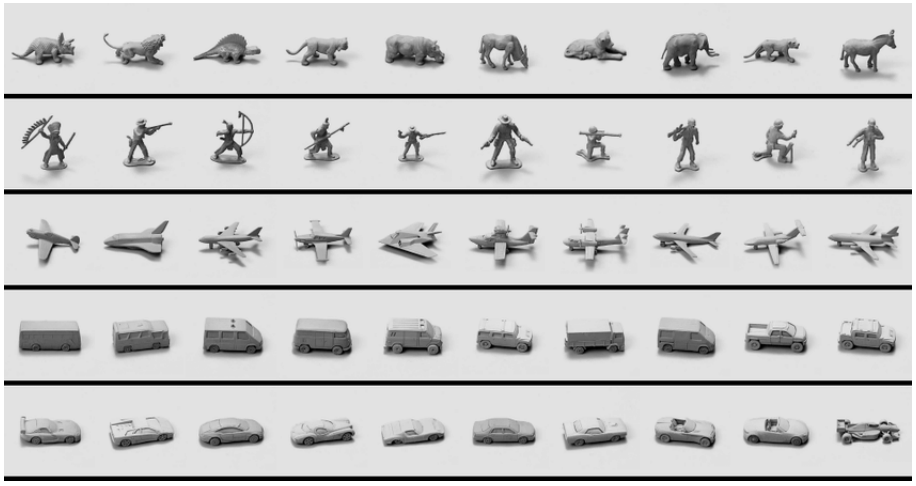
Το smallNORB [28] προορίζεται για πειράματα σε τρισδιάστατη (3D) αναγνώριση αντικειμένων με βάση το σχήμα και περιέχει δικάναλες (stereo) γκρι εικόνες μεγέθους 96x96 pixel.



Πηγή: Stack Overflow

Σχήμα 5.2: Ζευγάρι εικόνων που ανήκει στην κατηγορία "αεροπλάνο" με χρήση των δύο καμερών

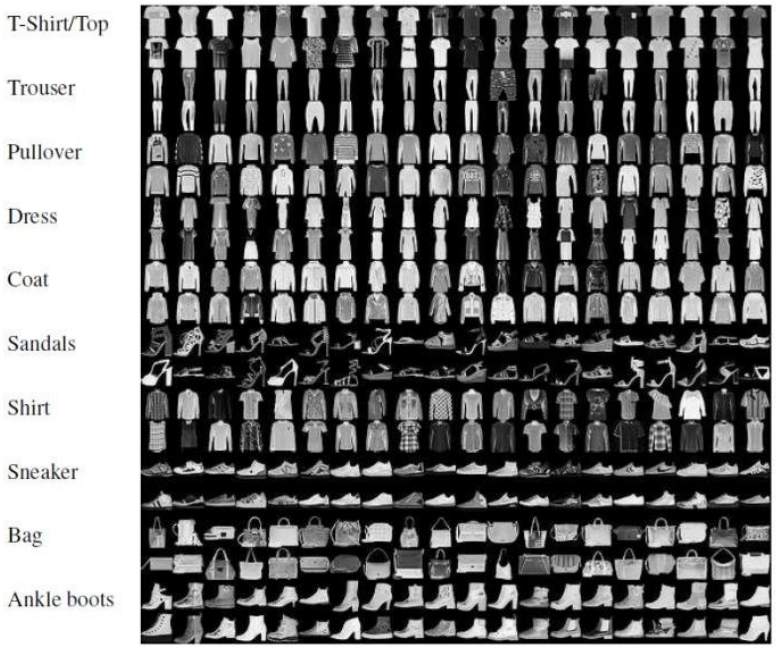
Οι εικόνες του smallNORB απεικονίζουν 50 παιχνίδια-αντικείμενα που ανήκουν σε 5 γενικές κατηγορίες: τετράποδα ζώα, ανθρώπινες φιγούρες, αεροπλάνα, φορτηγά και αυτοκίνητα. Υπάρχουν 10 διαφορετικά φυσικά αντικείμενα σε κάθε κατηγορία (Σχήμα 5.3), 5 εκ των οποίων έχουν επιλεχθεί για το σύνολο εκπαίδευσης, ενώ τα άλλα 5 έχουν επιλεχθεί για το σύνολο ελέγχου. Για την απεικόνιση αυτών των αντικειμένων χρησιμοποιήθηκαν δύο κάμερες υπό 6 διαφορετικές συνθήκες φωτισμού, 9 διαφορετικά υψόμετρα προβολής (γωνίες 30° έως 70° με βήμα 5°) και 18 διαφορετικά αζιμούθια (γωνίες 0° έως 340° με βήμα 20°). Έτσι, τόσο το σύνολο εκπαίδευσης όσο και το σύνολο ελέγχου αποτέλουνται από 24.300 ζευγάρια (στέρεο) εικόνων έκαστο.



Πηγή: ResearchGate
Σχήμα 5.3: Σύνολο Δεδομένων smallNORB, με τα αντικείμενα κάθε κατηγορίας ευθυγραμισμένα σε μηδενικό αζιμούθιο

5.3 Σύνολο FashionMNIST

Το Fashion-MNIST [29] είναι ένα σύνολο δεδομένων που απαρτίζεται από 70.000 γκρι εικόνες μεγέθους 28x28 pixel, οι οποίες απεικονίζουν προϊόντα μόδας που ανήκουν σε 10 διαφορετικές κατηγορίες, με 7.000 δείγματα να αντιστοιχούν σε κάθε μια από αυτές. Το σύνολο δεδομένων Fashion-MNIST χωρίζεται σε ένα σύνολο εκπαίδευσης. που περιέχει 60.000 εικόνες, και σε ένα σύνολο ελέγχου, που περιέχει τις υπόλοιπες 10.000 εικόνες. Το Fashion-MNIST δημιουργήθηκε έτσι ώστε να αποτελέσει μια άμεση αντικατάσταση του συνόλου δεδομένων MNIST, προκειμένου να γίνεται συγκριτική αξιολόγηση μεταξύ αλγορίθμων μηχανικής μάθησης, μιας και σε σχέση τα δύο σύνολα μοιράζονται κοινά στοιχεία όπως ίδιες διαστάσεις εικόνων, ίδια μορφή δεδομένων και ίδια δομή ως προς το χωρισμό σε σύνολο εκπαίδευσης και σύνολο ελέγχου.



Πηγή: ResearchGate
Σχήμα 5.4: Σύνολο Δεδομένων Fashion-MNIST

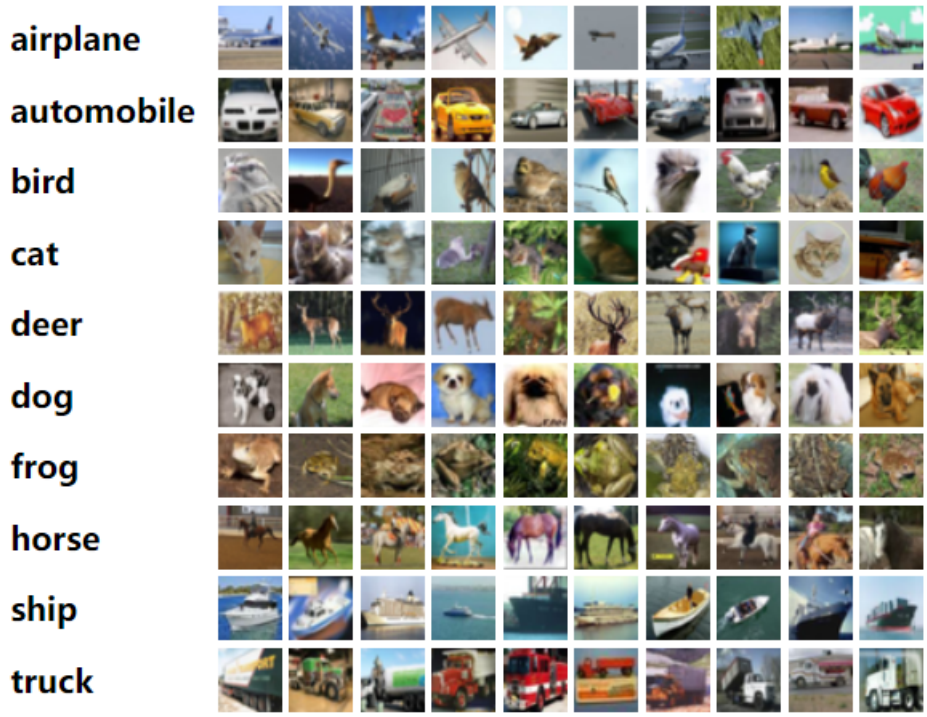
5.4 Σύνολο SVHN

Το Street View House Numbers (SVHN) είναι ένα σύνολο δεδομένων εικόνων πραγματικού κόσμου που έχει συλλεχθεί από τους αριθμούς σπιτιών στις εικόνες του Google Street View και χρησιμοποιείται για την ανάπτυξη αλγορίθμων μηχανικής εκμάθησης και αναγνώρισης αντικειμένων. Είναι ένα από τα κοινά χρησιμοποιούμενα σύνολα δεδομένων αναφοράς, καθώς απαιτεί ελάχιστη προεπεξεργασία και μορφοποίηση δεδομένων. Αν και μοιράζεται κάποιες ομοιότητες με το MNIST όπου οι εικόνες είναι με μικρά περικομμένα ψηφία, το SVHN αλλά απαρτίζεται από μιας τάξη μεγέθους περισσότερα επισημασμένα δεδομένα (πάνω από 600.000 εικόνες ψηφίων). Προέρχεται επίσης από ένα σημαντικά δυσκολότερο πραγματικό πρόβλημα αναγνώρισης ψηφίων και αριθμών, αυτό της αναγνώρισης σε εικόνες φυσικών σκηνών. Οι εικόνες δεν έχουν καμία ομαλοποίηση αντίθεσης, περιέχουν επικαλυπτόμενα ψηφία και αποσπούν την προσοχή, γεγονός που το καθιστά πολύ πιο δύσκολο πρόβλημα σε σύγκριση με το MNIST.

Πηγή: http://ufldl.stanford.edu/housenumbers
Σχήμα 5.5: Σύνολο Δεδομένων SVHN

5.5 Σύνολο CIFAR-10

Το CIFAR-10 [30] (όπως και το CIFAR-100) είναι επισημασμένο υποσύνολο του συνόλου δεδομένων "80 million tiny images" [31], και συλλέχθηκε από τους Alex Krizhevsky, Vinod Nair, και Geoffrey Hinton. Το σύνολο δεδομένων CIFAR-10 αποτελείται από 60.000 έγχρωμες εικόνες μεγέθους 32x32 pixel, που απεικονίζουν 10 διαφορετικές κατηγορίες, με 6.000 δείγματα να αντιστοιχούν σε κάθε μια από αυτές. Το CIFAR-10 χωρίζεται σε 50.000 εικόνες εκπαίδευσης και 10.000 εικόνες ελέγχου. Στο Σχήμα 5.6 φαίνονται οι κατηγορίες του συνόλου δεδομένων, κάθε μια συνοδευόμενη από 10 τυχαία επιλεγμένα παραδείγματα εικόνων που της αντιστοιχούν.



Πηγή: http://www.cs.toronto.edu/~kriz/cifar.html
Σχήμα 5.6: Σύνολο Δεδομένων CIFAR-10

Σημειώνεται ότι το σύνολο δεδομένων είναι χωρισμένο σε 5 δέσμες εκπαίδευσης και 1 δέσμη ελέγχου, με 10.000 εικόνες. Η δέσμη ελέγχου περιέχει ακριβώς 1.000 τυχαία επιλεγμένες εικόνες από κάθε κατηγορία. Οι δέσμες εκπαίδευσης περιέχουν τις υπολειπόμενες εικόνες σε τυχαία διάταξη. Ωστόσο, κάποιες δέσμες εκπαίδευσης μπορεί να περιέχουν περισσότερες εικόνες από μια κατηγορία σε σχέση με μια άλλη. Στο συνόλο τους βέβαια, οι δέσμες εκπαίδευσης περιέχουν ακριβώς 5.000 εικόνες από κάθε κατηγορία.

Κεφάλαιο 6

Κάψουλες και Αλγόριθμοι Δρομολόγησης

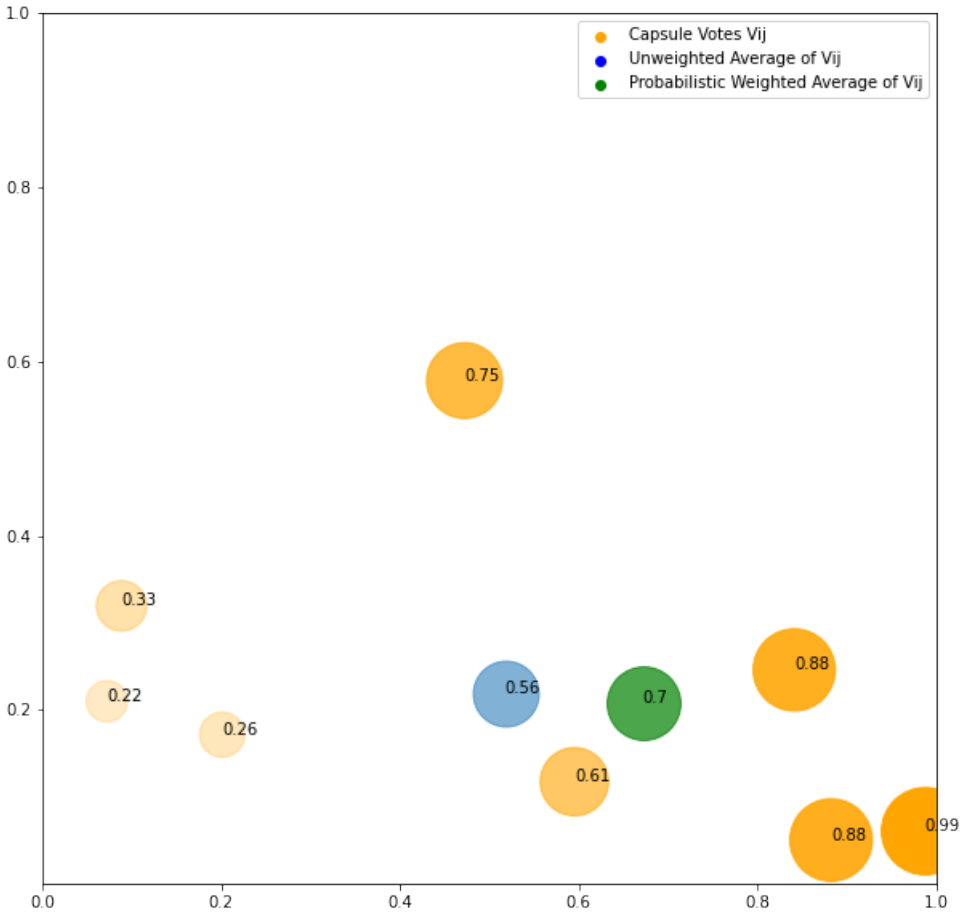
Στο κεφάλαιο αυτό θα γίνει περιγραφή μιας σειράς αλγορίθμων δρομολόγησης για κάψουλες, οι οποίοι σχεδιάστηκαν και υλοποιήθηκαν στην παρούσα ερευνητική εργασία, καθώς και αναλυτική παρουσίαση της επίδοσης τους, στα γνωστά σύνολα δεδομένων που αναφέρθηκαν σε προηγούμενη ενότητα.

Για τη συνέχεια, θα υποθέσουμε ότι έχουμε ένα σύνολο από διανύσματα-ψήφους V των καψουλών παιδιών (επίπεδο l) για τις πόζες των καψουλών γονέων (επίπεδο $l+1$). Πιο συγκεκριμένα το σύνολο V αποτλείται από στοιχεία V_{ij} που αντιπροσωπεύουν την ψήφο της κάψουλας-παιδί i για την πόζα της κάψουλας-γονέα j.

6.1 Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου

Προκειμένου να δρομολογήσουμε τις ψήφους των καψουλών i προς κάποια κάψουλα-γονέα j ορίστηκε ως βάση ένας αλγόριθμος πιθανοτικής δρομολόγησης σταθμισμένου μέσου. Η βασική επιδίωξη μέσω αυτού του αλγορίθμου είναι, δίνοντας πιο πολύ βάρος στις κάψουλες-παιδιά που έχουν μεγάλη πιθανότητα ενεργοποίησης παρά σε εκείνες που έχουν μικρή πιθανότητα ενεργοποίησης, να υπολογίσουμε μια πόζα για την κάψουλα-γονέα j. Όπως έχει ξαναναφερθεί στη θεωρία των διανυσματικών καψουλών, κάθε κάψουλα αντιπροσωπεύει μια οντότητα της εικόνας εισόδου, και το μήκος του διανύσματος αυτού (ή αλλιώς η ευκλείδεια νόρμα αυτού του διανύσματος) αποτελεί την πιθανότητα ύπαρξης της οντότητας αυτής στην εικόνα. Το ίδιο ισχύει και για το διάνυσμα ψήφο V_{ij} μιας κάψουλας-παιδί i με διάνυσμα u_i προς την κάψουλα γονέα j, δηλαδή η νόρμα $\|V_{ij}\|$ αποτελεί την πιθανότητα ύπαρξης της οντότητας της κάψουλας i με την πόζα της να έχει διαμορφωθεί από τη χρήση του ενδιάμεσου αφινικού μετασχηματισμού W_{ij} τέτοιου ώστε $V_{ij} = W_{ij} \cdot u_i$. Ορίζουμε λοιπόν ως πόζα για την κάψουλα-γονέα j τον σταθμισμένο μέσο όλων των ψήφων V_{ij} με βάρη της αντίστοιχες πιθανότητες ενεργοποίησης $\|V_{ij}\|$.

$$\mu_j = \frac{\sum_i \|V_{ij}\| \cdot V_{ij}}{\sum_i \|V_{ij}\|} \tag{6.1}$$



Σχήμα 6.1: Πόζα κάψουλας γονέα (πράσινο) χρησιμοποιώντας πιθανοτική δρομολόγηση σταθμισμένου μέσου στις ψήφους των καψουλών παιδιών (κίτρινα) συγκριτικά με το να χρησιμοποιηθεί απλός μέσος (μπλε)

Αλγόριθμος 3 Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου

procedure ProbabilisticAverageRouting(V_j):
    $\mu_j \leftarrow \frac{\sum_i \|V_{ij}\| \cdot V_{ij}}{\sum_i \|V_{ij}\|}$
    return μ_j

6.2 Dropout Δρομολόγηση

Στην βασική ιδέα που παρουσιάσαμε προηγουμένως παρατηρήθηκε ότι εκπαιδεύοντας περισσότερο ένα δίκτυο χρησιμοποιώντας την προηγούμενη τεχνική, μπορούμε εύκολα να οδηγηθούμε σε υπερταίριασμα (overfitting). Υποτέθηκε ότι ενδεχομένως να μη πρέπει σε κάθε πέρασμα του δικτύου να χρησιμοποιούνται όλες οι κάψουλες παιδιά i για τον προσδιορισμό της πόζας μιας κάψουλας γονέα j, έτσι ώστε το μοντέλο να καταφέρει να γενικεύσει καλύτερα. Μια κλασ-

σική τεχνική για την καταπολέμηση του overfitting στα νευρωνικά δίκτυα είναι η μέθοδος του dropout [32]. Αντί βέβαια να εισαχθεί ένα κλασσικό dropout επίπεδο, σχεδιάστηκε μια παραλλαγμένη μορφή αυτού, προσαρμοσμένη σε κάψουλες και ενθυλακωμένη μέσα σε έναν αλγόριθμο δρομολόγησης αυτών.

Ας υποθέσουμε ότι έχουμε μια dropout πιθανότητα p για κάψουλες, η οποία αντιπροσωπεύει την πιθανότητα να απενεργοποιηθεί εξ ολοκλήρου το διάνυσμα ψήφου μιας κάψουλας παιδιού i προς την κάψουλα γονέα j. Θεωρούμε ακόμη (όπως και στην κλασσική μέθοδο dropout) ότι τα ενδεχόμενα ω_i να απενεργοποιηθεί η ψήφος μιας κάψουλας i προς την κάψουλα γονέα j είναι μεταξύ τους ανεξάρτητα. Έτσι λοιπόν για κάθε ζεύγος καψουλών παιδιού και γονέα, i και j αντίστοιχα, ορίζεται μια τυχαία μεταβλητή $r_{ij} \sim \text{Bernoulli}(1-p)$, η οποία θα είναι ίση με 0 ή 1 αν η ψήφος της κάψουλας παιδιού i προς την κάψουλα γονέα j απενεργοποιείται ή ενεργοποιείται αντίστοιχα:

$$P(r_{ij}) = \begin{cases} 1-p & \text{αν } r_{ij} = 1 \\ p & \text{αν } r_{ij} = 0 \end{cases} \tag{6.2}$$

Επομένως, ορίζεται ένα νέο σύνολο από διανύσματα-ψήφους U καψουλών παιδιών για τις πόζες των καψουλών γονέων, όπου η ψήφος μιας κάψουλας παιδί i για την πόζα μιας κάψουλας γονέα j είναι:

$$U_{ij} = \begin{cases} V_{ij} & \text{αν } r_{ij} = 1 \\ \mathbf{0} & \text{αν } r_{ij} = 0 \end{cases} \tag{6.3}$$

Στη συνέχεια εφαρμόζεται η ιδέα του αλγορίθμου της πιθανοτικής δρομολόγησης σταθμισμένου μέσου στο νέο σύνολο ψήφων U.

$$\mu_j = \frac{\sum_i \|U_{ij}\| \cdot U_{ij}}{\sum_i \|U_{ij}\|} \tag{6.4}$$

Αλγόριθμος 4 Dropout Δρομολόγηση

1: **procedure** DropoutRouting(V_j, p, l):
2: **for** all capsule i in lower layer l **do**: $r_{ij} \sim \text{Bernoulli}(1-p)$
3: **for** all capsule i in lower layer l **do**: $U_{ij} \leftarrow r_{ij} \cdot V_{ij}$
4: $\mu_j \leftarrow \frac{\sum_i \|U_{ij}\| \cdot U_{ij}}{\sum_i \|U_{ij}\|}$
5: **return** μ_j

6.3 Δρομολόγηση Υποσυνόλου

Έως τώρα παρουσιάστηκαν ιδέες οι οποίες λαμβάνουν υπόψη μόνο την πιθανότητα ύπαρξης των καψουλών παιδιών, και όχι τη συμφωνία μεταξύ τους. Μπορούμε να πάμε την ιδέα της dropout δρομολόγησης ένα βήμα παραπέρα εισάγοντας μέσα και την έννοια της συμφωνίας.

Σχεδιάστηκε λοιπόν ένας αλγόριθμος δρομολόγησης, με βάση τον οποίο η πόζα μιας κάψουλας γονέα j θα υπολογίζεται από ένα υποσύνολο S ψήφων V_{ij} των καψουλών παιδιών i που έχουν μεγαλύτερο βαθμό συμφωνίας βάσει κάποιου κριτηρίου, και γι' αυτό ο αλγόριθμος αυτός ονομάστηκε δρομολόγηση υποσυνόλου.

Η διαδικασία της δρομολόγησης υποσυνόλου ξεκινάει υπολογίζοντας μια εκτίμηση για την πόζα μιας κάψουλας γονέα j με βάση τον αλγόριθμο πιθανοτικής δρομολόγησης σταθμισμένου μέσου:

$$\mu_j = \frac{\sum_i \|V_{ij}\| \cdot V_{ij}}{\sum_i \|V_{ij}\|} \tag{6.5}$$

Για να οριστεί ο βαθμός συμφωνίας μεταξύ μιας ψήφου V_{ij} και της εκτίμησης μ_j, χρησιμοποιήσαμε δύο διαφορετικές μετρικές, την L^2 νόρμα της διαφοράς τους, και το εσωτερικό τους γινόμενο. Όσο μικρότερη είναι η L^2 νόρμα της διαφοράς τους (ή όσο μεγαλύτερο είναι το εσωτερικό τους γινόμενο), τόσο μεγαλύτερο βαθμό συμφωνίας θα λέμε ότι παρουσίαζουν.

Χρησιμοποιώντας μια διαδικασία ανατροφοδότησης, ενεργοποιούνται οι S ψήφοι που εμφανίζουν μεγαλύτερο βαθμό συμφωνίας με την εκτίμηση μ_j, και απενεργοποιούνται οι υπόλοιπες. Έτσι, επιτυγχάνεται να έχουμε μια καλή προσέγγιση των S ψήφων που συμφωνούν και περισσότερο μεταξύ τους. Δρομολογούνται λοιπόν μόνο αυτές οι S ψήφοι, δηλαδή ορίζεται ένα νέο σύνολο από διανύσματα-ψήφους U καψουλών παιδιών για τις πόζες των καψουλών γονέων, και πόζα της κάψουλας γονέα j υπολογίζεται ως:

$$\mu_j^* = \frac{\sum_i \|U_{ij}\| \cdot U_{ij}}{\sum_i \|U_{ij}\|} \tag{6.6}$$

Αλγόριθμος 5 Δρομολόγησης Υποσυνόλου

procedure SUBSETROUTING(V_j, S, l):
  $\mu_j \leftarrow \frac{\sum_i \|V_{ij}\| \cdot V_{ij}}{\sum_i \|V_{ij}\|}$
  for all capsule i in lower layer l **do**: $L_{ij} \leftarrow \|\mu_j - V_{ij}\|$
  $threshold_j \leftarrow s_{th}$ smallest loss in L_j
  for all capsule i in lower layer l **do**:
    if $L_{ij} \leq threshold_j$ **then** $r_{ij} \leftarrow 1$ **else** $r_{ij} \leftarrow 0$
  for all capsule i in lower layer l **do**: $U_{ij} \leftarrow r_{ij} \cdot V_{ij}$
  $\mu_j^* \leftarrow \frac{\sum_i \|U_{ij}\| \cdot U_{ij}}{\sum_i \|U_{ij}\|}$
  return μ_j^*

6.4 Δρομολόγηση Τυχαίου Δείγματος Συναίνεσης (RANSAC)

Στη συνέχεια, σχεδιάστηκε ένας μηχανισμός δρομολόγησης που χρησιμοποιεί τις βασικές ιδέες του γνωστού αλγορίθμου RANSAC. Με βάση αυτήν την RANSAC δρομολόγηση, για τον υπολογισμό της πόζας μιας κάψουλας γονέα j θα λαμβάνονται H υποθέσεις. Κάθε υπόθεση h θα χαρακτηρίζεται από ένα τυχαία επιλεγμένο υποσύνολο U^h, αποτελούμενο από S ψήφους V_{ij} των καψουλών παιδιών i, με βάση το οποίο στη συνέχεια θα υπολογίζεται μια εκτίμηση μ_j^h για την πόζα της κάψουλας γονέα j, ως ο πιθανοτικός σταθμισμένος μέσος του τρέχοντος υποσυνόλου.

$$\mu_j^h = \frac{\sum_i \|U_{ij}^h\| \cdot U_{ij}^h}{\sum_i \|U_{ij}^h\|} \tag{6.7}$$

Σκοπός της RANSAC δρομολόγησης είναι να επιλέξει την βέλτιστη υπόθεση h^* από το σύνολο των H υποθέσεων, δηλαδή να επιλέξει, από τα δοθέντα υποσύνολα, το υποσύνολο ψήφων U^{h^*} που παράγει εκτίμηση $\mu_j^{h^*}$ για την πόζα της κάψουλας γονέα j η οποία μεγιστοποιεί την συνολική συμφωνία, δηλαδή το άθροισμα των επιμέρους συμφωνιών αυτής με κάθε ψήφο V_{ij} του αρχικού συνόλου ψήφων V.

Υποθέτουμε ότι ως κριτήριο μεγιστοποίησης του αθροίσματος των επιμέρους συμφωνιών, χρησιμοποιείται η ελαχιστοποίηση του αθροίσματος των L^2 νορμών των επιμέρους διαφορών.

$$L_j^h = \sum_i \|\mu_j^h - V_{ij}\| \tag{6.8}$$

$$h^* = \underset{1 \leq h \leq H}{\operatorname{argmin}}(L_j^h) \tag{6.9}$$

Επομένως, όπως γίνεται αντιληπτό η πόζα της της κάψουλας γονέα j δίνεται ως εξής:

$$\mu_j^{h^*} = \frac{\sum_i \|U_{ij}^{h^*}\| \cdot U_{ij}^{h^*}}{\sum_i \|U_{ij}^{h^*}\|} \tag{6.10}$$

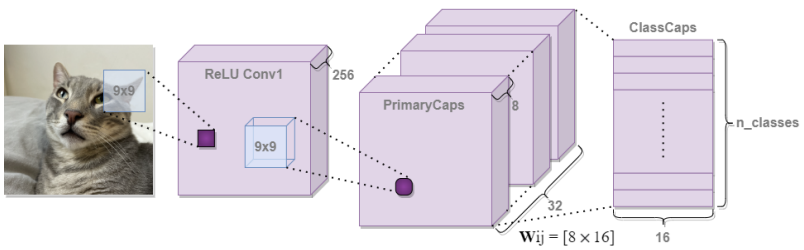
Αλγόριθμος 6 RANSAC Δρομολόγηση

procedure RansacRouting(V_j, s, H, l):
  for H hypotheses **do**:
    $r_j^h \leftarrow$ random vector with s 1_s in total and the rest being 0_s
    for all capsule i in lower layer l **do**: $U_{ij}^h \leftarrow r_{ij}^h \cdot V_{ij}$
    $\mu_j^h \leftarrow \frac{\sum_i \|U_{ij}^h\| \cdot U_{ij}^h}{\sum_i \|U_{ij}^h\|}$
    $L_j^h \leftarrow \sum_i \|\mu_j^h - V_{ij}\|$
  $h^* \leftarrow \underset{1 \leq h \leq H}{\operatorname{argmin}}(L_j^h)$
  return $\mu_j^{h^*}$

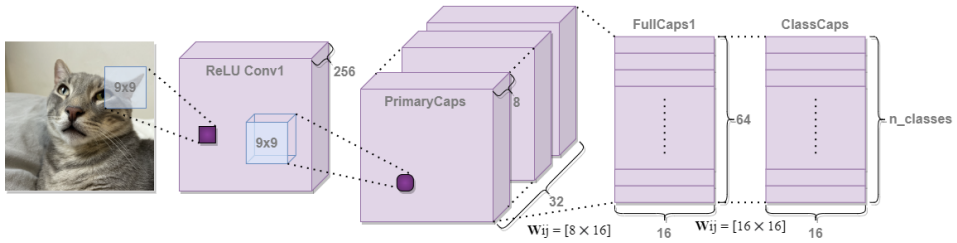
6.5 Πειραματικά αποτελέσματα αλγορίθμων δρομολόγησης

Στη συνέχεια θα γίνει μια εκτενής πειραματική ανάλυση προκειμένου να εκτιμηθεί η επίδοση καθενός εκ των προαναφερθέντων προτεινόμενων αλγορίθμων δρομολόγησης. Πιο συγκεκριμένα, οι αλγόριθμοι διαφοροποιούνται ως προς τον τρόπο με τον οποίο προσεγγίζουν μια είσοδο. Θα χρησιμοποιήσουμε 5 σύνολα δεδομένων, τα οποία αποτελούν ευρέως γνωστά σημεία αναφορας (benchmarks) για την εκτίμηση της επίδοσης, τόσο κλασσικών συνελικτικών δικτύων, όσο και νευρωνικών δικτύων με κάψουλες. Προκείται για τα σύνολα MNIST, Fashion-MNIST, smallNORB, SVHN και CIFAR-10, που περιγράψαμε σε προήγουμενη ενότητα, και όπως είδαμε διαφέρουν σημαντικά μεταξύ τους. Στόχος λοιπόν αυτής της πειραματικής ανάλυσης είναι να δαπιστωθεί η κατεύθυνση στην οποία πρέπει να κίνηθει η υλοποίηση ενός νευρωνικού δικτύου με κάψουλες, καθώς και του αλγορίθμου με τον οποίο αυτές δρομολογούνται από ένα επίπεδο σε ένα άλλο, έτσι ώστε να προσεγγιστεί καλύτερα καθένα από αυτά τα σύνολα δεδομένων.

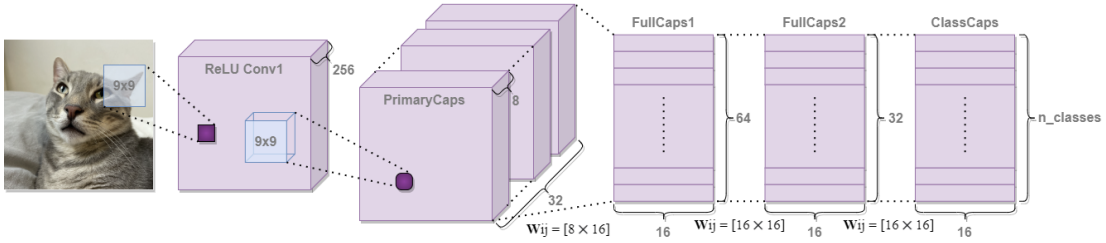
Κατασκευάστηκαν 5 διαφορετικές αρχιτεκτονικές νευρωνικών δικτύων με κάψουλες οι οποίες και φαίνονται παρακάτω:



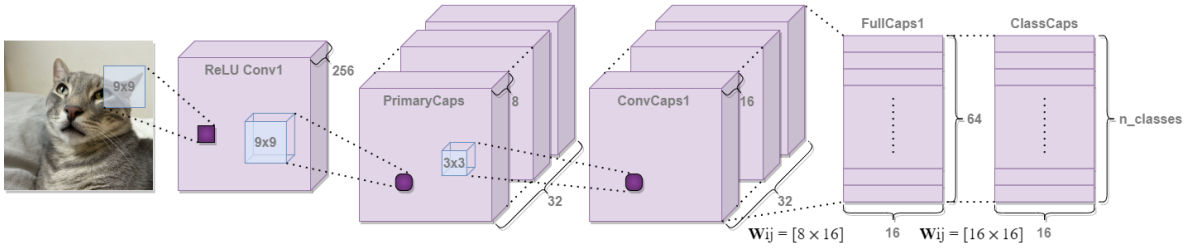
Σχήμα 6.2: CapsNet-1: Νευρωνικό Δίκτυο Καψουλών αποτελούμενο από ένα συνελικτικό επίπεδο, το βασικό επίπεδο καψουλών και το επίπεδο καψουλών κατηγοριών



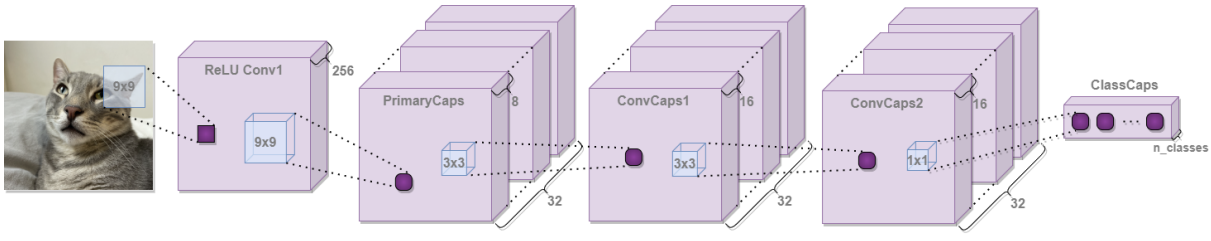
Σχήμα 6.3: CapsNet-2: Νευρωνικό Δίκτυο Καψουλών αποτελούμενο από ένα συνελικτικό επίπεδο, το βασικό επίπεδο καψουλών, ένα πλήρως συνδεδεμένο επίπεδο καψουλών και το επίπεδο καψουλών κατηγοριών



Σχήμα 6.4: CapsNet-3: Νευρωνικό Δίκτυο Καψουλών αποτελούμενο από ένα συνελικτικό επίπεδο, το βασικό επίπεδο καψουλών, δύο πλήρως συνδεδεμένα επίπεδα καψουλών και το επίπεδο καψουλών κατηγοριών



Σχήμα 6.5: CapsNet-4: Νευρωνικό Δίκτυο Καψουλών αποτελούμενο από ένα συνελικτικό επίπεδο, το βασικό επίπεδο καψουλών, ένα συνελικτικό επίπεδο καψουλών, ένα πλήρως συνδεδεμένο επίπεδο καψουλών και το επίπεδο καψουλών κατηγοριών



Σχήμα 6.6: CapsNet-5: Νευρωνικό Δίκτυο Καψουλών αποτελούμενο από ένα συνελικτικό επίπεδο, το βασικό επίπεδο καψουλών, δύο συνελικτικά επίπεδα καψουλών και ένα συνελικτικό επίπεδο καψουλών κατηγοριών

Η συνάρτηση απωλειών που χρησιμοποιήθηκε για την εκπαίδευση του εκάστοτε δικτύου καψουλών είναι η συνάρτηση MarginLoss (Σχέση 6.11) της οποίας η λειτουργία περιγράφηκε

στην §4.3.4.

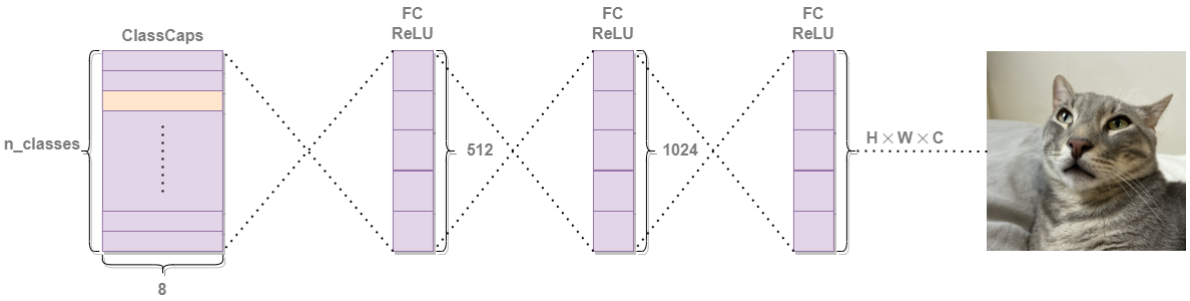
$$L_k = \underbrace{T_k \max(0, m^+ - \|\mathbf{v}_k\|)^2}_{\text{class } k \text{ present}} + \underbrace{\lambda(1 - T_k) \max(0, \|\mathbf{v}_k\| - m^-)^2}_{\text{class } k \text{ not present}} \tag{6.11}$$

Επιθυμούμε μια υψηλή τιμή για το κατώφλι πιθανότητας m^+ ύπαρξης μιας κατηγορίας, ενώ επιθυμούμε μια χαμήλη τιμή για το κατώφλι πιθανότητας m^- μη ύπαρξης μιας κατηγορίας. Επομένως, για τις υπερπαραμέτρους που αντιπροσωπεύουν τα δύο κατώφλια επιλέχθηκαν οι τιμές $m^+ = 0.9$ και $m^- = 0.1$. Αναφορικά με την υπερπαράμετρο λ που συνεισφέρει στη μείωση του βάρους των απωλειών που οφείλονται στις κατηγορίες που δεν είναι παρούσες στην τρέχουσα είσοδο, επιλέχθηκε η τιμή $\lambda = 0.5$ έτσι ώστε η αρχική εκπαίδευση να μην επιφέρει συρρίκνωση του μήκους των διανυσμάτων δραστηριότητας για όλες τις κάψουλες κατηγοριών.

Τελικά, το συνολικό MarginLoss είναι απλά το άθροισμα των απωλειών όλων των καψουλών κατηγοριών.

$$\mathcal{L}_{margin} = \sum_k L_k \tag{6.12}$$

Κρίθηκε χρήσιμο επίσης σε μερικές από αυτές τις αρχιτεκτονικές να φανεί κατά πόσο θα ήταν χρήσιμη η προσθήκη ενός δικτύου αποκωδικοποίησης, το οποίο θα ανακατασκευάζει την αρχική εικόνα εισόδου από την κάψουλα της εξόδου του ClassCaps επιπέδου που περιέχει την αντίστοιχη αναπαράσταση ως προς την πραγματική κατηγορία στην οποία ανήκει η αρχική είσοδος. Πιο συγκεκριμένα, αυτό το δίκτυο αποκωδικοποίησης λαμβάνει ως είσοδο ένα διάνυσμα αναπαράστασης, αυτό της προαναφερθείσας κάψουλας, αγνοώντας (masking) τις κάψουλες που αντιστοιχούν στις λοιπές κατηγορίες. Αυτό το διάνυσμα χαρακτηριστικών, τροφοδοτείται μέσα σε 3 πλήρως συνδεδεμένα επίπεδα (συνοδευόμενα από τις αντίστοιχες συναρτήσεις ενεργοποίησης), όπως φαίνεται στο Σχήμα 6.7 με σκοπό την ανακατασκευή της αρχικής εικόνας εισόδου. Κατά την εκπαίδευση λοιπόν του δικτύου, αυτό το δίκτυο στοχεύει στην ελαχιστοποίηση της ευκλείδειας απόστασης της ανακατασκευασμένης εικόνας σε σχέση με την αρχική, δηλαδή στην ελαχιστοποίηση του αθροίσματος των τετραγωνικών διαφορών μεταξύ των εντάσεων των pixel της αρχική εικόνας και των λογιστικών μονάδων της εξόδου αποκωδικοποίησης.



Σχήμα 6.7: Δίκτυο αποκωδικοποίησης για την ανακατασκευή μιας εικόνας εισόδου από τις αναπαραστάσεις του ClassCaps επιπέδου

Στην περίπτωση χρήσης του δικτύου αποκωδικοποίησης λοιπόν εισάγεται μια επιπρόσθετη απώλεια που οφείλεται στην ανακατασκευή. Αν υποθέσουμε ότι έχουμε μια αρχική εικόνα εισόδου $\boldsymbol{x}$, και $\hat{\boldsymbol{x}}$ μια ανακατασκευασμένη εκδοχή της $\boldsymbol{x}$ όπως παράγεται από το δίκτυο αποκωδικοποίησης, τότε ορίζουμε μια συνάρτηση απωλειών ανακατασκευής ReconstructionLoss όπως φαίνεται στην Σχέση 6.13.

$$\mathcal{L}_{rec} = \|\boldsymbol{x} - \hat{\boldsymbol{x}}\| \tag{6.13}$$

Σε αυτήν την περίπτωση λοιπόν η συνολική απώλεια L_{total} θα είναι ένα σταθμισμένο άθροισμα του συνολικού MarginLoss και του ReconstructionLoss:

$$\mathcal{L}_{total} = \alpha_{margin} \cdot \mathcal{L}_{margin} + \alpha_{rec} \cdot \mathcal{L}_{rec} \tag{6.14}$$

Για τις υπερπαραμέτρους στάθμισης των δύο απωλειών τέθηκαν οι τιμές $\alpha_{margin} = 1.0$ και $\alpha_{rec} = 0.0005$, έτσι ώστε το MarginLoss να μείνει αναλοίωτο, και η απώλεια ανακατασκευής να μην κυριαρχεί επί του MarginLoss κατά την εκπαίδευση, παραμόνο να ωθεί σε επαρκή βαθμό τις κάψουλες κατηγοριών να κωδικοποιήσουν τις παραμέτρους στιγμιοτύπου της εικόνας εισόδου.

Ακόμη, στα δίκτυα στα οποία το επίπεδο ClassCaps αποτελεί συνελικτικό επίπεδο καψουλών, δοκιμάσαμε, αντί της εισαγωγής δικτύου ανακατασκευής, να εφαρμόσουμε την τεχνική της Προσθήκης Συντεταγμένων (Coordinate Addition) πάνω στις εξόδους του συνελικτικού ClassCaps επιπέδου [6], όπως αυτή περιγράφηκε στην §4.4.3.

Κατά τα λοιπά, σε όλα τα πειράματα που πραγματοποιήθηκαν, συμβουλεύομενοι τις μεθόδους που έχουν ακολουθηθεί σε προηγούμενες ερευνητικές εργασίες πάνω σε νευρωνικά δίκτυα με κάψουλες [3, 6, 8], χρησιμοποιήθηκε αποκλειστικά ο βελτιστοποιητής Adam. Ωστόσο, πειραματιστήκαμε με διάφορες τιμές για την υπερπαράμετρο του ρυθμού εκπαίδευσης του βελτιστοποιητή, καθώς και με διαφοερτικές μεθόδους με τις οποίες αυτό θα μεταβάλλεται, δυναμικά ή μη, κατά την εκπαίδευση.

Ακόμη, πειραματιστήκαμε και με διαφορετικές τιμές αναφορικά με την υπερπαράμετρο του μέγεθους δέσμης. Προσπαθήσαμε να τροφοδοτούμε πάντοτε το εκάστοτε δίκτυο με τη μέγιστη δυνατή σε μέγεθος δέσμη. Μιας και οι πόροι του χρησιμοποιούμενου υπολογιστικού συστήματος που μας παρασχέθηκε, και πιο συγκεκριμένα η μνήμη RAM και η μνήμη της/των GPU(s), ήταν περιορισμένης χωριτικότητας, ενώ τα μοντέλα αρκετά σύνθετα και υψηλής χωρικής πολυπλοκότητας ως επί το πλείστον, προσπαθήσαμε να προσαρμόζουμε το τροφοδοτούμενο μέγεθος δέσμης στην εκάστοτε αρχιτεκτονική, γνωρίζοντας βέβαια ότι όσο μεγαλύτερο μέγεθος δέσμης χρησιμοποιήθει τόσο μεγαλύτερα και πιο εύρωστα βήματα εκπαίδευσης θα πρέπει να αναμένουμε, ενώ όσο μικρότερο μέγεθος δέσμης χρησιμοποιηθεί τόσο μικρότερα θα είναι τα αντίστοιχα βήματα εκπαίδευσης και τόσο πιο πιθανό θα είναι να είσαγουν θόρυβο στο μοντέλο.

Τέλος, χάρη στην υψηλή ταχύτητα των GPU(s) του υπολογιστικού συστήματος που μας παρασχέθηκε είχαμε τη δυνατότητα να πειραματιστούμε και με διαφορετικές τιμές αναφορικά με την υπερπαράμετρο των εποχών εκπαίδευσης, καθώς επιθυμούσαμε να διαπιστώσουμε κατά πόσο η βέλτιστη επιλογή αυτής εξαρτάται τόσο από το συνόλο δεδομένων όσο και από την επιλογή της αρχιτεκτονικής και του αλγορίθμου δρομολόγησης.

6.5.1 Πειραματικά αποτελέσματα αλγορίθμων δρομολόγησης στο MNIST

Κατά την εκπαίδευση ενός νευρωνικού δικτύου καψουλών στο MNIST, κάθε 28x28 γκρι εικόνα του συνόλου εκπαίδευσης κανονικοποιείται και μετατοπίζεται κατά 2 pixel σε κάθε κατεύθυνση με χρήση μηδενικής συμπλήρωσης (zero padding), δημιουργώντας κανονικοποιημένα μονοκάναλα δείγματα εκπαίδευσης μεγέθους 32x32 pixel. Η ίδια διαδικασία ακολουθείται και για κάθε 28x28 γκρι εικόνα του συνόλου ελέγχου του MNIST, δημιουργώντας κανονικοποιημένα μονοκάναλα δείγματα ελέγχου μεγέθους 32x32 pixel.

Εκπαιδεύσαμε το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο MNIST προκειμένου να δούμε πως καθένας εκ των αλγορίθμων δρομολόγησης προσεγγίζει το συγκεκριμένο σύνολο δεδομένων. Τα αποτελέσματα των πειραμάτων με χρήση διαφορετικών υπερπαραμέτρων φαίνονται παρακάτω:

Πίνακας 6.1: Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου στο MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Error(%)
CapsNet-1	512	100	0.001	yes	**0.33**
CapsNet-1	256	100	0.003	yes	0.37

Όπως φαίνεται στον Πίνακα 6.1, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο MNIST, με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 0.33% στο σύνολο ελέγχου του MNIST.

Πίνακας 6.2: Dropout Δρομολόγηση στο MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	p	Error(%)
CapsNet-1	512	100	0.001	yes	0.2	0.37
CapsNet-1	256	100	0.001	yes	0.4	**0.36**
CapsNet-1	512	100	0.001	yes	0.6	0.39
CapsNet-1	512	100	0.001	yes	0.8	0.38

Όπως φαίνεται στον Πίνακα 6.2, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο MNIST, με χρήση της Dropout Δρομολόγησης με dropout πιθανότητα $p = 0.4$ και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 0.36% στο σύνολο ελέγχου του MNIST.

Πίνακας 6.3: Δρομολόγηση Υποσυνόλου στο MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	Similarity	Error(%)
CapsNet-1	512	100	0.001	yes	0.3	MSE	**0.34**
CapsNet-1	256	100	0.001	yes	0.5	MSE	0.4
CapsNet-1	512	100	0.001	yes	0.7	MSE	0.4

Όπως φαίνεται στον Πίνακα 6.3, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο MNIST, με χρήση της Δρομολόγησης Υποσυνόλου με κλάσμα υποσυνόλου $S = 0.3$ και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 0.34% στο σύνολο ελέγχου του MNIST.

Πίνακας 6.4: RANSAC Δρομολόγηση στο MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	H	Similarity	Error(%)
CapsNet-1	256	50	0.001	yes	0.5	10	MSE	**0.42**

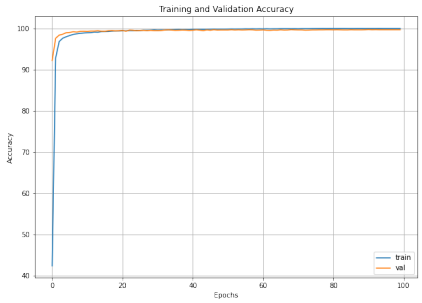
Όπως φαίνεται στον Πίνακα 6.4, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο MNIST, με χρήση της RANSAC Δρομολόγησης με κλάσμα υποσυνόλου $S = 0.5$ και $H = 10$ υποθέσεις, καθώς και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 0.42% στο σύνολο ελέγχου του MNIST.

Αν και όλοι οι αλγόριθμοι δρομολόγησης φαίνεται να επιτυγχάνουν εξίσου εξαιρετικές επιδόσεις στο MNIST εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1, παρατηρείται κατά τι υψηλότερη επίδοση από πλευράς της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου (Πίνακας 6.5).

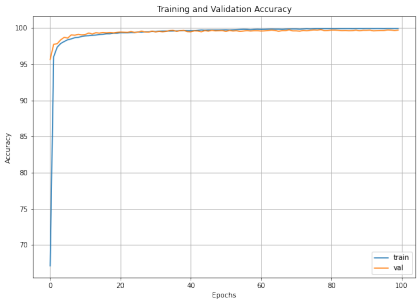
Πίνακας 6.5: Σύγκριση Αλγορίθμων Δρομολόγησης στο MNIST με την απλή αρχιτεκτονική αναφοράς CapsNet-1

Routing Method	Network	Error(%)
Probabilistic Weighted Average	CapsNet-1	**0.33**
Dropout	CapsNet-1	0.36
Subset	CapsNet-1	0.34
RANSAC	CapsNet-1	0.42

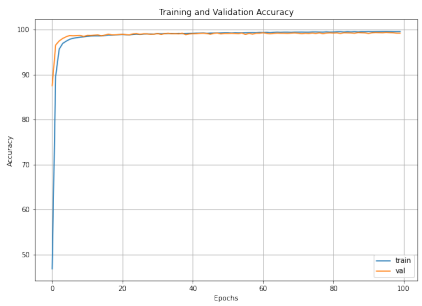
Στα γραφήματα των εικόνων 6.8 και 6.9 φαίνονται, για κάθε άλγόριθμο δρομολόγησης στην αρχιτεκτονική CapsNet-1, οι καμπύλες εκπαίδευσης, ακρίβειας (Accuracy) και απώλειας (Loss) αντίστοιχα, των μοντέλων που επιτυγχάνουν το μικρότερο ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του MNIST.



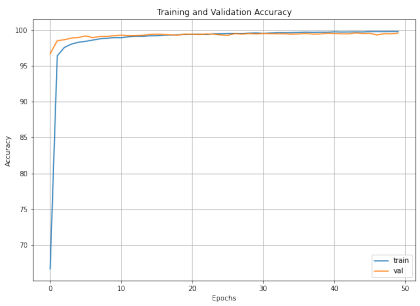
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση

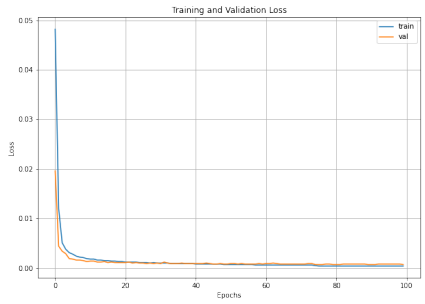


(c) Δρομολόγηση Υποσυνόλου

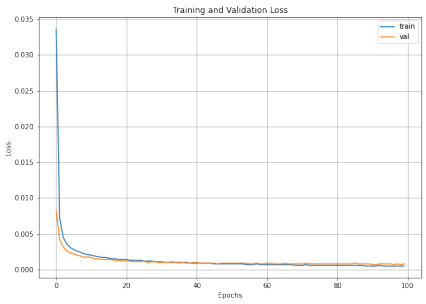


(d) RANSAC Δρομολόγηση

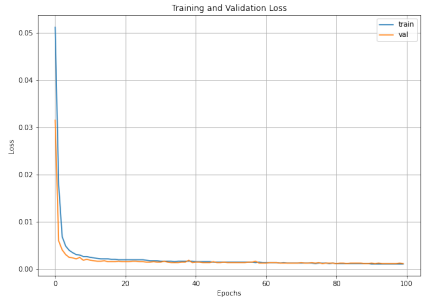
Σχήμα 6.8: Καμπύλες Εκπαίδευσης (Accuracy) ανά αλγόριθμο Δρομολόγησης στο MNIST



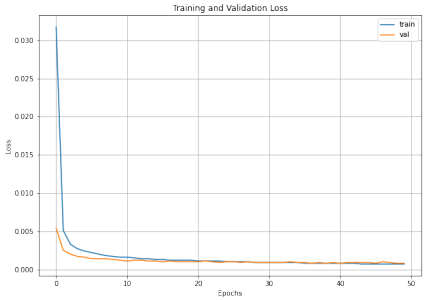
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση



(c) Δρομολόγηση Υποσυνόλου



(d) RANSAC Δρομολόγηση

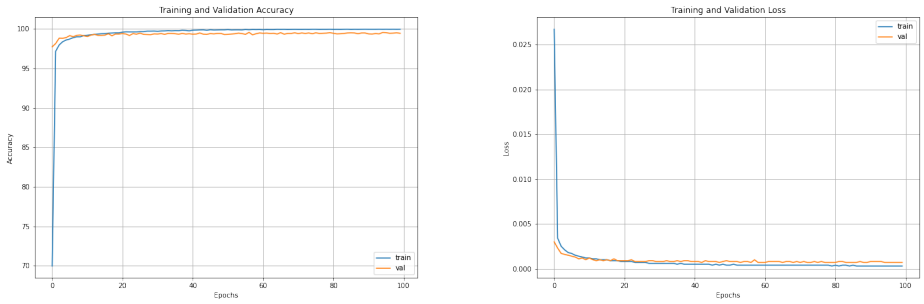
Σχήμα 6.9: Καμπύλες Εκπαίδευσης (Loss) ανά αλγόριθμο Δρομολόγησης στο MNIST

Επομένως, πραγματοποιήθηκαν ορισμένα επιπλέον πειράματα (Πίνακας 6.6) σε βαθύτερα δίκτυα καψουλών με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου, ώστε να φανεί πως μπορεί η προσθήκη επιπλέον επιπέδων καψουλών να επηρεάσει την εκπαίδευση και κατ' επέκταση το ποσοστό ακρίβειας αναφορικά με το σύνολο ελέγχου του MNIST.

Πίνακας 6.6: Περαιτέρω μελέτη της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου στο MNIST σε βαθύτερα δίκτυα καψουλών

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Error(%)
CapsNet-1	512	150	0.001	yes	0.33
CapsNet-1	256	150	0.003	yes	0.37
CapsNet-2	32	200	0.001	no	0.37
CapsNet-2	32	200	0.001	yes	**0.31**

Παρατηρούμε ότι με χρήση ενός επιπλέον επιπέδου πλήρως συνδεδεμένων καψουλών επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 0.31% στο σύνολο επαλήθευσης του MNIST (Σχήμα 6.10), ενώ ταυτόχρονα φαίνεται και ο ρόλος του δικτύου ανακατασκευής, μιας και χωρίς την ύπαρξη αυτού, το ίδιο δίκτυο επιτυγχάνει ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 0.37% στο σύνολο επαλήθευσης του MNIST.



Σχήμα 6.10: Καμπύλες Εκπαίδευσης (ακρίβειας και απώλειας αντίστοιχα) καλύτερου μοντέλου στο MNIST με χρήση Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου

6.5.2 Πειραματικά αποτελέσματα αλγορίθμων δρομολόγησης στο smallNORB

Το smallNORB, όπως προαναφέρθηκε, αποτελείται από δικάναλες (stereo) γκρι εικόνες μεγέθους 96x96 pixel. Προτού γίνει χρήση τους, είτε σε επίπεδο εκπαίδευσης είτε σε επίπεδο ελέγχου, πραγματοποιείται κατάλληλη υποδειγματοληψία έτσι ώστε να προκύψουν δικάναλες γκρι εικόνες μεγέθους 48x48 pixel.

Κατά την εκπαίδευση ενός νευρωνικού δικτύου καψουλών στο smallNORB, από κάθε 48x48 δικάναλη γκρι εικόνα του συνόλου εκπαίδευσης περικόπτεται ένα τυχαίο 32x32 τμήμα και κατόπιν υφίσταται κανονικοποίηση, δημιουργώντας κανονικοποιημένα δικάναλα δείγματα εκπαίδευσης μεγέθους 32x32 pixel. Στην περίπτωση του ελέγχου, η διαδικασία που ακολουθείται είναι κατά τι διαφορετική. Σε κάθε 48x48 δικάναλη γκρι εικόνα του συνόλου ελέγχου του smallNORB, περικόπτεται το κεντρικό της 32x32 τμήμα και κατόπιν κανονικοποιείται, δημιουργώντας, αυτή τη φορά μη τυχαία κανονικοποιημένα δικάναλα δείγματα ελέγχου μεγέθους 32x32 pixel.

Εκπαιδεύσαμε το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο smallNORB προκειμένου να δούμε πως καθένας εκ των αλγορίθμων δρομολόγησης προσεγγίζει το συγκεκριμένο σύνολο δεδομένων. Τα αποτελέσματα των πειραμάτων με χρήση διαφορετικών υπερπαραμέτρων φαίνονται παρακάτω:

Πίνακας 6.7: Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου στο smallNORB

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Error(%)
CapsNet-1	512	50	0.001	yes	**4.68**
CapsNet-1	256	50	0.003	yes	4.91

Όπως φαίνεται στον Πίνακα 6.7, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο smallNORB, με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 4.68% στο σύνολο ελέγχου του.

Πίνακας 6.8: Dropout Δρομολόγηση στο smallNORB

Network	Batch Size	Epochs	Learning Rate	Reconstruction	p	Error(%)
CapsNet-1	512	50	0.001	yes	0.2	4.49
CapsNet-1	256	50	0.001	yes	0.4	**4.60**
CapsNet-1	512	50	0.001	yes	0.6	4.68
CapsNet-1	512	50	0.001	yes	0.8	4.75

Όπως φαίνεται στον Πίνακα 6.8, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο smallNORB, με χρήση της Dropout Δρομολόγησης με dropout πιθανότητα $p = 0.4$ και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 4.60% στο σύνολο ελέγχου του smallNORB.

Πίνακας 6.9: Δρομολόγηση Υποσυνόλου στο smallNORB

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	Similarity	Error(%)
CapsNet-1	512	50	0.001	yes	0.3	MSE	5.14
CapsNet-1	256	50	0.001	yes	0.5	MSE	5.11
CapsNet-1	512	50	0.001	yes	0.7	MSE	**3.96**
CapsNet-1	512	50	0.001	yes	0.7	Cosine	4.01

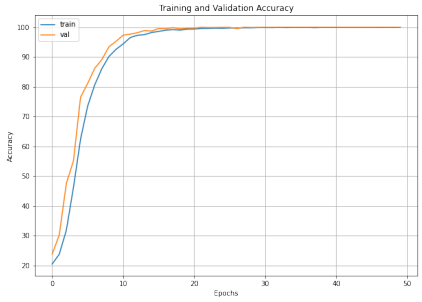
Όπως φαίνεται στον Πίνακα 6.9, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο smallNORB, με χρήση της Δρομολόγησης Υποσυνόλου με κλάσμα υποσυνόλου $S = 0.7$ και την ευκλείδεια απόσταση ως κριτήριο συμφωνίας και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 3.96% στο σύνολο ελέγχου του smallNORB.

Πίνακας 6.10: RANSAC Δρομολόγηση στο smallNORB

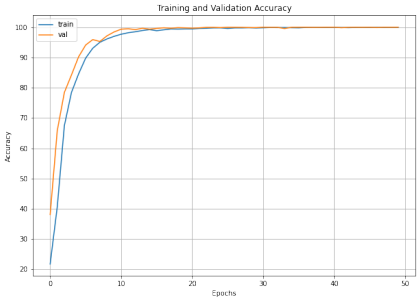
Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	H	Similarity	Error(%)
CapsNet-1	256	50	0.001	yes	0.5	10	MSE	**4.19**
CapsNet-1	256	50	0.001	yes	0.7	10	MSE	4.88

Όπως φαίνεται στον Πίνακα 6.10, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο smallNORB, με χρήση της RANSAC Δρομολόγησης με κλάσμα υποσυνόλου $S = 0.5$ και $H = 10$ υποθέσεις, καθώς και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 4.19% στο σύνολο ελέγχου του smallNORB.

Στα γραφήματα των εικόνων 6.11 και 6.12 φαίνονται, για κάθε άλγόριθμο δρομολόγησης στην αρχιτεκτονική CapsNet-1, οι καμπύλες εκπαίδευσης, ακρίβειας (Accuracy) και απώλειας (Loss) αντίστοιχα, των μοντέλων που επιτυγχάνουν το μικρότερο ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του smallNORB.

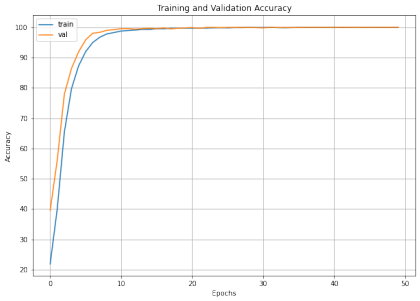


(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



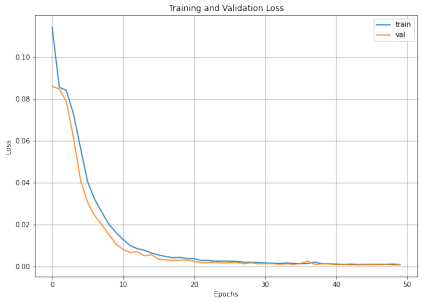
(b) Dropout Δρομολόγηση

(c) Δρομολόγηση Υποσυνόλου

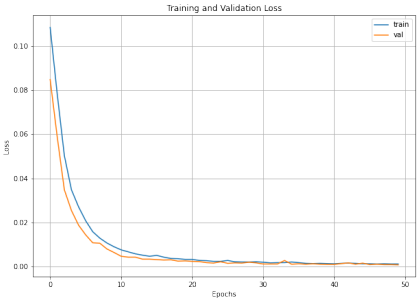


(d) RANSAC Δρομολόγηση

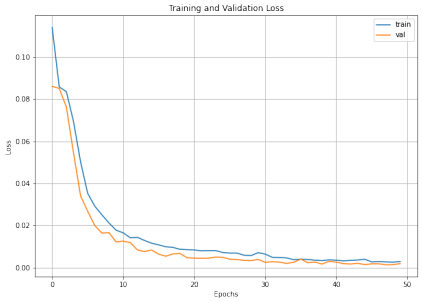
Σχήμα 6.11: Καμπύλες Εκπαίδευσης (Accuracy) ανά αλγόριθμο Δρομολόγησης στο smallNORB



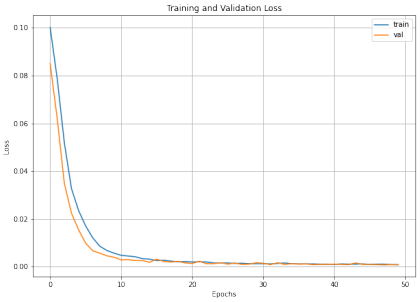
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση



(c) Δρομολόγηση Υποσυνόλου



(d) RANSAC Δρομολόγηση

Σχήμα 6.12: Καμπύλες Εκπαίδευσης (Loss) ανά αλγόριθμο Δρομολόγησης στο smallNORB

Πίνακας 6.11: Σύγκριση Αλγορίθμων Δρομολόγησης στο smallNORB με την απλή αρχιτεκτονική αναφοράς CapsNet-1

Routing Method	Network	Error(%)
Probabilistic Weighted Average	CapsNet-1	4.68
Dropout	CapsNet-1	4.60
Subset	CapsNet-1	**3.96**
RANSAC	CapsNet-1	4.19

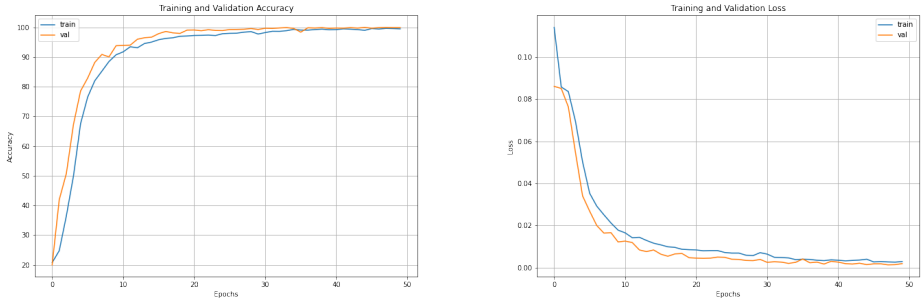
Στην περίπτωση του smallNORB, η Δρομολόγηση Υποσύνολου επιτυγχάνει ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 3.96% στο σύνολο ελέγχου το οποίο είναι το μικρότερο συγκριτικά με τους υπόλοιπους αλγορίθμους δρομολόγησης. Δεύτερη σε σειρά επιτυχίας έρχεται η RANSAC Δρομολόγηση. Το αποτέλεσμα αυτό εκτιμάται ότι οφείλεται στο γεγονός ότι οι δύο αυτοί αλγόριθμοι δρομολόγησης (Δρομολόγηση Υποσυνόλου, RANSAC Δρομολόγηση) δίνουν μεγάλη έμφαση στον καθορισμό των καψουλών που "συμφωνούν" περισσότερο μεταξύ τους, και μόνο τότε λαμβάνουν υπόψην την πιθανότητα ύπαρξης της οντότητας που καθεμιά από αυτές τις κάψουλες αντιπροσωπεύει. Αντιθέτως, οι δύο άλλοι αλγόριθμοι δρομολόγησης (Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου, Dropout Δρομολόγηση) δίνουν απόλυτη έμφαση στην πιθανότητα ύπαρξης των οντοτήτων των καψουλών και αδιαφορούν για την μεταξύ τους "συμφωνία" ή μη. Αντιθέτως, το smallNORB είναι ένα σύνολο δεδομένων, του οποίου τα δείγματα είναι εικόνες αντικειμένων από διαφορετικές οπτικές γωνίες και συνθήκες φωτισμού. Οι οποίες του smallNORB δεν είναι καθόλου πολύπλοκες, αποτελούν πολύ ξεκάθαρες απεικονίσεις των αντικειμένων, με το φόντο της εκάστοτε εικόνας να έχει αμελητέες χρωματικές διαβαθμίσεις ή διαβαθμίσεις φωτεινότητας, τόσο που το smallNORB ως πρόβλημα κατηγοριοποίησης θα μπορούσε να εντάσσεται στην κατηγορία αναγνώρισης σχήματος (shape recognition). Σε ένα τέτοιο σύνολο δεδομένων κρίνεται αρκετά χρήσιμη όχι άπλα η αναγνώριση ύπαρξης των μικρότερων οντοτήτων που σχηματίζουν το τελικό αντικείμενο, αλλά και ο τρόπος με τον οποίο αυτά τα μικρότερα μέρη συνδυάζονται μεταξύ τους (π.χ. προσανατολισμός) για την σύνθεση όλο και μεγαλύτερων οντοτήτων με τελικό στόχο το απεικονιζούμενο αντικείμενο της εκάστοτε κατηγορίας. Ουσιαστικά ο τρόπος που συνδυάζονται οι μικρότερες οντότητες για να σχηματίσουν μια πιο σύνθετη είναι αυτό που προσπαθούμε να προσομοιώσουμε με την συμφωνία των χαμηλότερου επιπέδου καψουλών προς σχηματισμό μιας κάψουλας υψηλότερου επιπέδου.

Επομένως, πραγματοποιήθηκαν ορισμένα επιπλέον πειράματα (Πίνακας 6.12) σε βαθύτερα δίκτυα καψουλών με χρήση της Δρομολόγησης Υποσυνόλου, ώστε να φανεί πως μπορεί η προσθήκη επιπλέον επιπέδων καψουλών να επηρεάσει την εκπαίδευση και κατ' επέκταση το ποσοστό ακρίβειας αναφορικά με το σύνολο ελέγχου του smallNORB.

Πίνακας 6.12: Περαιτέρω μελέτη της Δρομολόγησης Υποσυνόλου στο smallNORB σε βαθύτερα δίκτυα καψουλών

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Coordinate Addition	S	Similarity	Error(%)
CapsNet-1	512	250	0.001	yes	–	0.3	MSE	5.14
CapsNet-1	256	250	0.001	yes	–	0.5	MSE	5.11
CapsNet-1	512	250	0.001	yes	–	0.7	MSE	**3.96**
CapsNet-1	512	250	0.001	yes	–	0.7	Cosine	4.01
CapsNet-2	32	300	0.001	yes	–	0.7	MSE	4.76
CapsNet-2	32	300	0.001	yes	–	0.7	Cosine	4.28
CapsNet-3	32	250	0.001	no	–	0.7	MSE	4.66
CapsNet-3	32	250	0.001	no	–	0.7	Cosine	3.97
CapsNet-4	32	115	0.001	no	–	0.7	MSE	5.70
CapsNet-4	32	350	0.001	no	–	0.7	Cosine	5.05
CapsNet-5	16	250	0.001	no	yes	0.7	MSE	4.51

Παρατηρούμε ότι εκπαιδεύοντας τις λοιπές αρχιτεκτονικές που αποτελούνται από επιπλέον επίπεδα καψουλών (άλλωτε πλήρως συνδεδεμένα επίπεδα καψουλών και άλλωτε συνελικτικά επίπεδα καψουλών) το ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης δεν μειώνεται. Αντιθέτως το σφάλμα αυτό αυξάνεται, καθώς τα δείγματα του smallNORB δεν είναι πολύπλοκες εικόνες. Στην πραγματικότητα παρατηρήθηκε ότι με αύξηση των επιπέδων το υπό εκπαίδευση δίκτυο έχει την τάση να προσπαθεί να υπερταιριάξει (overfitting) τα βάρη του στα δεδομένα με αποτέλεσμα να δυσκολεύεται περισσότερο να προβλέψει κατηγορίες για δεδομένα που δεν έχει "ξαναδεί".



Σχήμα 6.13: Καμπύλες Εκπαίδευσης (ακρίβειας και απώλειας αντίστοιχα) καλύτερου μοντέλου στο smallNORB με χρήση Δρομολόγησης Υποσυνόλου

6.5.3 Πειραματικά αποτελέσματα αλγορίθμων δρομολόγησης στο Fashion-MNIST

Κατά εκπαίδευση ενός νευρωνικού δικτύου καψουλών στο Fashion-MNIST, κάθε 28x28 γκρι εικόνα του συνόλου εκπαίδευσης κανονικοποιείται και μετατοπίζεται κατά 2 pixel σε κάθε κατεύθυνση με χρήση μηδενικής συμπλήρωσης (zero padding), δημιουργώντας κανονικοποιημένα μονοκάναλα δείγματα εκπαίδευσης μεγέθους 32x32 pixel. Η ίδια διαδικασία ακολουθείται και για κάθε 28x28 γκρι εικόνα του συνόλου ελέγχου του Fashion-MNIST, δημιουργώντας κανονικοποιημένα μονοκάναλα δείγματα ελέγχου μεγέθους 32x32 pixel.

Εκπαιδεύσαμε το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο Fashion-MNIST προκειμένου να δούμε πως καθένας εκ των αλγορίθμων δρομολόγησης προσεγγίζει το συγκεκριμένο σύνολο δεδομένων. Τα αποτελέσματα των πειραμάτων με χρήση διαφορετικών υπερπαραμέτρων φαίνονται παρακάτω:

Πίνακας 6.13: Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου στο Fashion-MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Error(%)
CapsNet-1	512	150	0.001	yes	**6.60**
CapsNet-1	512	150	0.001	no	6.71
CapsNet-1	256	150	0.003	yes	6.62

Όπως φαίνεται στον Πίνακα 6.13, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο Fashion-MNIST, με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 6.60% στο σύνολο ελέγχου του.

Πίνακας 6.14: Dropout Δρομολόγηση στο Fashion-MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	p	Error(%)
CapsNet-1	512	150	0.001	yes	0.2	**6.70**
CapsNet-1	256	150	0.001	yes	0.4	6.77
CapsNet-1	512	200	0.001	yes	0.6	6.82
CapsNet-1	512	200	0.001	yes	0.8	7.15

Όπως φαίνεται στον Πίνακα 6.14, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο Fashion-MNIST, με χρήση της Dropout Δρομολόγησης με dropout πιθανότητα $p = 0.2$ και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 6.70% στο σύνολο ελέγχου του Fashion-MNIST.

Πίνακας 6.15: Δρομολόγηση Υποσυνόλου στο Fashion-MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	Similarity	Error(%)
CapsNet-1	512	200	0.001	yes	0.3	MSE	7.71
CapsNet-1	512	200	0.001	yes	0.5	MSE	7.52
CapsNet-1	512	300	0.001	yes	0.7	MSE	**7.25**
CapsNet-1	512	200	0.001	yes	0.7	Cosine	7.44

Όπως φαίνεται στον Πίνακα 6.15, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο Fashion-MNIST, με χρήση της Δρομολόγησης Υποσυνόλου με κλάσμα υποσυνόλου $S = 0.7$ και την ευκλείδεια απόσταση ως κριτήριο συμφωνίας και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 7.25% στο σύνολο ελέγχου του Fashion-MNIST.

Πίνακας 6.16: RANSAC Δρομολόγηση στο Fashion-MNIST

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	H	Similarity	Error(%)
CapsNet-1	256	150	0.001	yes	0.5	10	MSE	**7.62**
CapsNet-1	256	150	0.001	yes	0.7	10	MSE	7.83

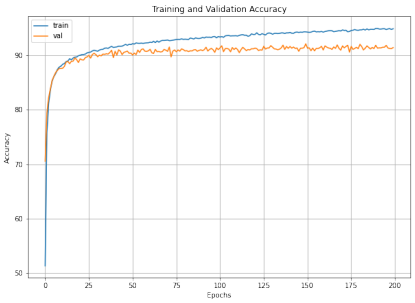
Όπως φαίνεται στον Πίνακα 6.16, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο Fashion-MNIST, με χρήση της RANSAC Δρομο-

λόγησης με κλάσμα υποσυνόλου $S = 0.5$ και $H = 10$ υποθέσεις, καθώς και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 7.62% στο σύνολο ελέγχου του Fashion-MNIST.

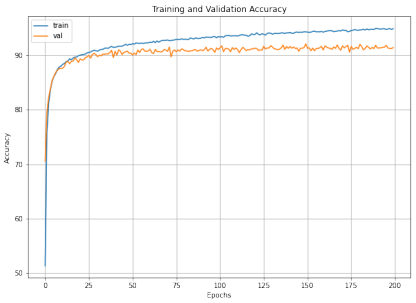
Στα γραφήματα των εικόνων 6.14 και 6.15 φαίνονται, για κάθε άλγόριθμο δρομολόγησης στην αρχιτεκτονική CapsNet-1, οι καμπύλες εκπαίδευσης, ακρίβειας (Accuracy) και απώλειας (Loss) αντίστοιχα, των μοντέλων που επιτυγχάνουν το μικρότερο ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του Fashion-MNIST.

(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου

(b) Dropout Δρομολόγηση

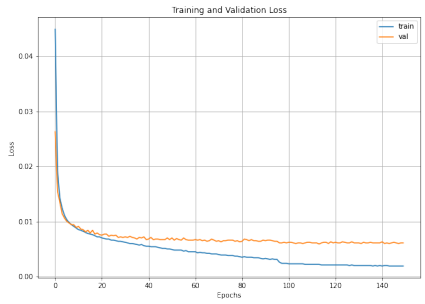


(c) Δρομολόγηση Υποσυνόλου

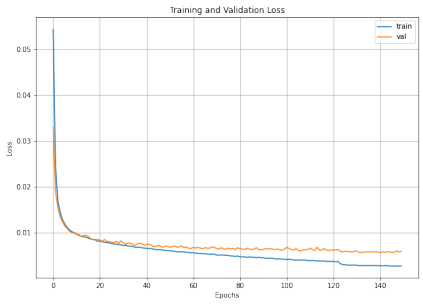


(d) RANSAC Δρομολόγηση

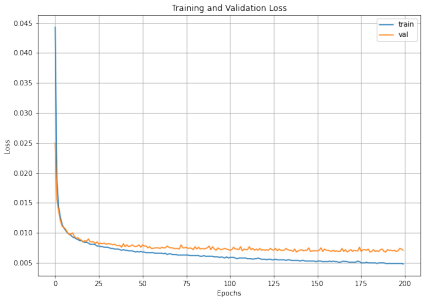
Σχήμα 6.14: Καμπύλες Εκπαίδευσης (Accuracy) ανά αλγόριθμο Δρομολόγησης στο Fashion-MNIST



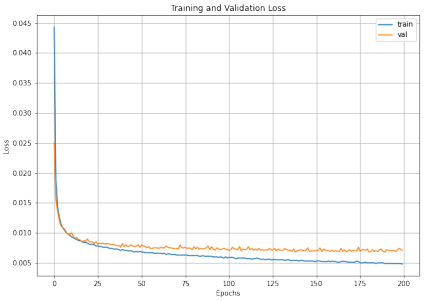
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση



(c) Δρομολόγηση Υποσυνόλου



(d) RANSAC Δρομολόγηση

Σχήμα 6.15: Καμπύλες Εκπαίδευσης (Loss) ανά αλγόριθμο Δρομολόγησης στο Fashion-MNIST

Πίνακας 6.17: Σύγκριση Αλγορίθμων Δρομολόγησης στο Fashion-MNIST με την απλή αρχιτεκτονική αναφοράς CapsNet-1

Routing Method	Network	Error(%)
Probabilistic Weighted Average	CapsNet-1	**6.60**
Dropout	CapsNet-1	6.70
Subset	CapsNet-1	7.25
RANSAC	CapsNet-1	7.62

Στην περίπτωση του Fashion-MNIST, η Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου επιτυγχάνει ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 6.16% στο σύνολο ελέγχου το οποίο είναι το μικρότερο συγκριτικά με τους υπόλοιπους αλγορίθμους δρομολόγησης. Δεύτερη σε σειρά επιτυχίας έρχεται η Dropout Δρομολόγηση. Το αποτέλεσμα αυτό εκτιμάται ότι οφείλεται στο γεγονός ότι οι δύο αυτοί αλγόριθμοι δρομολόγησης (Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου, Dropout Δρομολόγηση) δίνουν απόλυτη έμφαση στην πιθανότητα ύπαρξης της οντότητας που αντιπροσωπεύει κάθε κάψουλα. Αντιθέτως, οι δύο άλλοι αλγόριθμοι δρομολόγησης (Δρομολόγηση Υποσυνόλου, RANSAC Δρομολόγηση) δίνουν μεγάλη έμφαση στον καθορισμό των καψουλών που "συμφωνούν" περισσότερο μεταξύ τους, και μόνο τότε λαμβάνουν υπόψην την πιθανότητα ύπαρξης της οντότητας. Αντιθέτως, το Fashion-MNIST είναι ένα σύνολο δεδομένων, του οποίου τα δείγματα είναι εικόνες προϊόντων μόδας που έχουν ληφθεί από μία και μόνο οπτική γωνία για κάθε κατηγορία. Αναμένουμε ότι χαρακτηριστικά όπως ο προσανατολισμός μιας μεσαίου προς υψηλού επιπέδου οντότητας θα έχει χαμηλή διακύμανση στο Fashion-MNIST. Επομένως, όλες οι επιμέρους οντότητες που απαρτίζουν ένα αντικείμενο έχουν συνδυαστεί με παρόμοιο τρόπο προκειμένου να συνθέσουν το τελικό απεικονιζούμενο προϊόν.

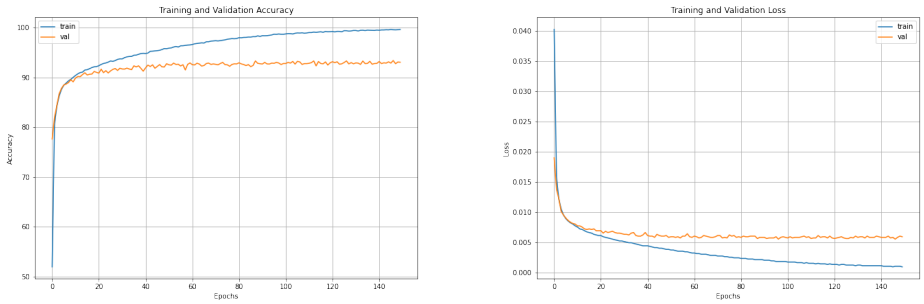
Διαισθητικά μάλιστα οι πιο υψηλού επιπέδου οντότητες φαίνονται να είναι αρκετά ενδεικτικές της κατηγορίας στην οποία ανήκει το εκάστοτε αντικείμενο, και έτσι η ύπαρξη μιας τέτοιας οντότητας φαίνεται να είναι αρκετή προκειμένου να συναχθεί στο συμπέρασμα της κατηγορίας στην οποία αυτό ανήκει. Από την αλλή δίνοντας έμφαση στη συμφωνία μεταξύ των καψουλών και απορρίπτοντας καθολικά έναν αριθμό αυτών, μπορεί να χαθεί σημαντική πληροφορία (όπως και συμβαίνει) καθώς ενδέχεται να αγνοηθούν κάψουλες με μεγάλη πιθανότητα ύπαρξης της αντίστοιχης οντότητας.

Επομένως, πραγματοποιήθηκαν ορισμένα επιπλέον πειράματα (Πίνακας 6.18) σε βαθύτερα δίκτυα καψουλών με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου, ώστε να φανεί πως μπορεί η προσθήκη επιπλέον επιπέδων καψουλών να επηρεάσει την εκπαίδευση και κατ' επέκταση το ποσοστό ακρίβειας αναφορικά με το σύνολο ελέγχου του Fashion-MNIST.

Πίνακας 6.18: Περαιτέρω μελέτη της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου στο Fashion-MNIST σε βαθύτερα δίκτυα καψουλών

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Coordinate Addition	Error(%)
CapsNet-1	512	150	0.001	yes	–	6.60
CapsNet-1	512	150	0.001	no	–	6.71
CapsNet-1	256	150	0.003	yes	–	6.62
CapsNet-2	128	150	0.001	yes	–	**6.16**
CapsNet-3	32	200	0.001	no	–	6.43
CapsNet-4	32	200	0.001	no	–	7.41
CapsNet-4	32	200	0.001	yes	–	7.24
CapsNet-5	8	200	0.001	yes	no	7.12
CapsNet-5	16	200	0.005	no	yes	8.00

Παρατηρούμε ότι προσθέτοντας ένα ακόμη πλήρως συνδεδεμένο επίπεδο καψουλών και πάλι τη χρήση δικτύου ανακατασκευής, μπορεί να επιτευχθεί μείωση του ποσοστού σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του Fashion-MNIST κατά $\approx 0.5\%$, καθώς το δίκτυο αναγκάζεται να μάθει στις κάψουλες πιο πολύπλοκα χαρακτηριστικά των εικόνων εισόδου με αποτέλεσμα τελικά να επιτυγχάνεται ποσοστό σφάλματος 6.16%. Οι καμπύλες εκπαίδευσης του αντίστοιχου μοντέλου φαίνονται στο Σχήμα 6.16.



Σχήμα 6.16: Καμπύλες Εκπαίδευσης (ακρίβειας και απώλειας αντίστοιχα) καλύτερου μοντέλου στο Fashion-MNIST με χρήση Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου

6.5.4 Πειραματικά αποτελέσματα αλγορίθμων δρομολόγησης στο SVHN

Κατά την εκπαίδευση ενός νευρωνικού δικτύου καψουλών στο SVHN, κάθε 32x32 έγχρωμη εικόνα του συνόλου εκπαίδευσης κανονικοποιείται και μετατοπίζεται κατά 2 pixel σε κάθε κατεύθυνση με χρήση μηδενικής συμπλήρωσης (zero padding), δημιουργώντας εικόνες μεγέθους 36x36 pixel. Έπειτα, από κάθε 36x36 έγχρωμη εικόνα του συνόλου εκπαίδευσης περικόπτεται ένα τυχαίο 32x32 τμήμα, δημιουργώντας τελικά κανονικοποιημένα δείγματα έγχρωμων εικόνων εκπαίδευσης μεγέθους 32x32 pixel. Αντιθέτως, στην περίπτωση του ελέγχου η διαδικασία που ακολουθείται είναι αρκέτα απλούστερη, καθώς κάθε έγχρωμη εικόνα του συνόλου ελέγχου του SVHN απλώς κανονικοποιείται, δημιουργώντας απλά κανονικοποιημένες εκδοχές των αρχικών έγχρωμων 32x32 εικόνων ελέγχου.

Εκπαιδεύσαμε το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο SVHN προκειμένου να δούμε πως καθένας εκ των αλγορίθμων δρομολόγησης προσεγγίζει το συγκεκριμένο σύνολο δεδομένων. Τα αποτελέσματα των πειραμάτων με χρήση διαφορετικών υπερπαραμέτρων φαίνονται παρακάτω:

Πίνακας 6.19: Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου στο SVHN

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Error(%)
CapsNet-1	512	150	0.001	yes	**5.08**
CapsNet-1	256	150	0.003	yes	5.11

Όπως φαίνεται στον Πίνακα 6.19, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο SVHN, με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 5.08% στο σύνολο ελέγχου του.

Πίνακας 6.20: Dropout Δρομολόγηση στο SVHN

Network	Batch Size	Epochs	Learning Rate	Reconstruction	p	Error(%)
CapsNet-1	512	150	0.001	yes	0.2	**5.32**
CapsNet-1	256	150	0.001	yes	0.4	5.46
CapsNet-1	512	150	0.001	yes	0.6	5.63
CapsNet-1	512	150	0.001	yes	0.8	5.90

Όπως φαίνεται στον Πίνακα 6.20, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο SVHN, με χρήση της Dropout Δρομολόγησης με dropout πιθανότητα $p = 0.2$ και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 5.32% στο σύνολο ελέγχου του SVHN.

Πίνακας 6.21: Δρομολόγηση Υποσυνόλου στο SVHN

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	Similarity	Error(%)
CapsNet-1	512	300	0.001	yes	0.3	MSE	7.28
CapsNet-1	512	300	0.001	yes	0.5	MSE	7.88
CapsNet-1	512	300	0.001	yes	0.7	MSE	**5.81**
CapsNet-1	512	300	0.001	yes	0.7	Cosine	6.72

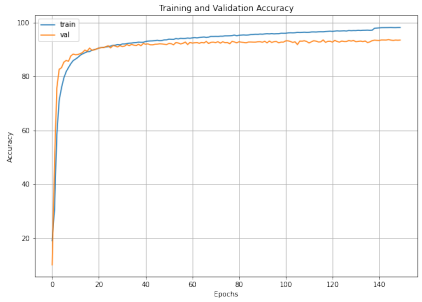
Όπως φαίνεται στον Πίνακα 6.21, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο SVHN, με χρήση της Δρομολόγησης Υποσυνόλου με κλάσμα υποσυνόλου $S = 0.7$ και την ευκλείδεια απόσταση ως κριτήριο συμφωνίας και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 5.81% στο σύνολο ελέγχου του SVHN.

Πίνακας 6.22: RANSAC Δρομολόγηση στο SVHN

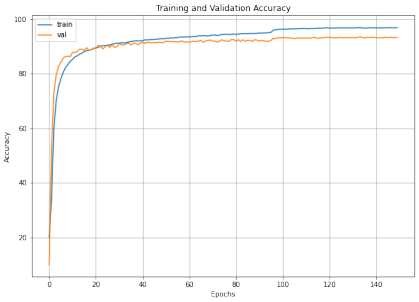
Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	H	Similarity	Error(%)
CapsNet-1	256	140	0.001	yes	0.5	10	MSE	**5.87**
CapsNet-1	256	140	0.001	yes	0.7	10	MSE	6.32

Όπως φαίνεται στον Πίνακα 6.22, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο SVHN, με χρήση της RANSAC Δρομολόγησης με κλάσμα υποσυνόλου $S = 0.5$ και $H = 10$ υποθέσεις, καθώς και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 5.87% στο σύνολο ελέγχου του SVHN.

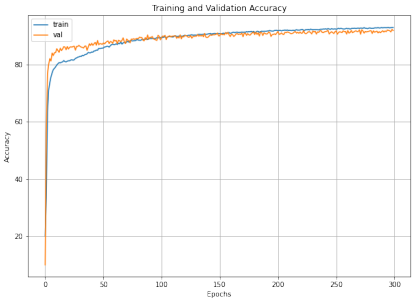
Στα γραφήματα των εικόνων 6.17 και 6.18 φαίνονται, για κάθε άλγόριθμο δρομολόγησης στην αρχιτεκτονική CapsNet-1, οι καμπύλες εκπαίδευσης, ακρίβειας (Accuracy) και απώλειας (Loss) αντίστοιχα, των μοντέλων που επιτυγχάνουν το μικρότερο ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του SVHN.



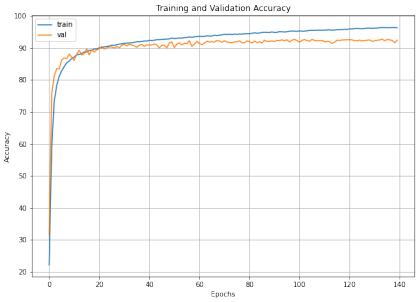
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση

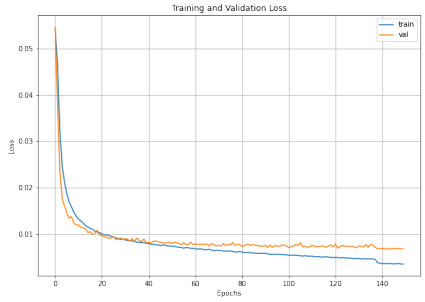


(c) Δρομολόγηση Υποσυνόλου

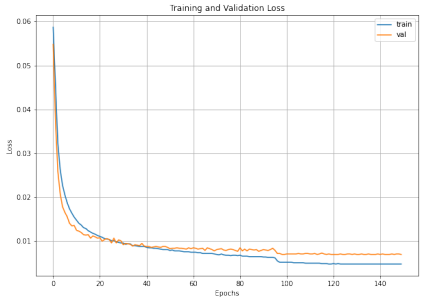


(d) RANSAC Δρομολόγηση

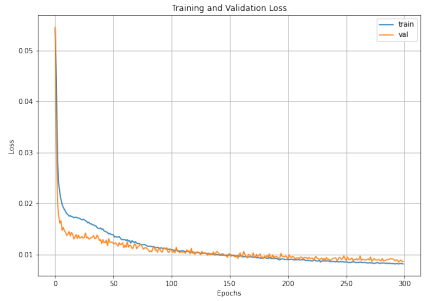
Σχήμα 6.17: Καμπύλες Εκπαίδευσης (Accuracy) ανά αλγόριθμο Δρομολόγησης στο SVHN



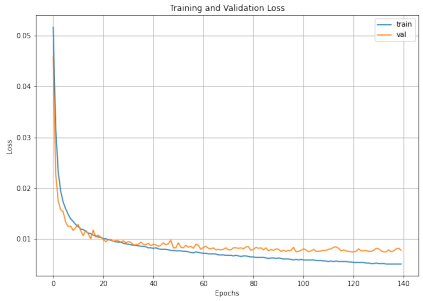
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση



(c) Δρομολόγηση Υποσυνόλου



(d) RANSAC Δρομολόγηση

Σχήμα 6.18: Καμπύλες Εκπαίδευσης (Loss) ανά αλγόριθμο Δρομολόγησης στο SVHN

Πίνακας 6.23: Σύγκριση Αλγορίθμων Δρομολόγησης στο SVHN με την απλή αρχιτεκτονική αναφοράς CapsNet-1

Routing Method	Network	Error(%)
Probabilistic Weighted Average	CapsNet-1	**5.08**
Dropout	CapsNet-1	5.32
Subset	CapsNet-1	5.81
RANSAC	CapsNet-1	5.87

Στην περίπτωση του SVHN, η Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου επιτυγχάνει ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 5.08% στο σύνολο ελέγχου το οποίο είναι το μικρότερο συγκριτικά με τους υπόλοιπους αλγορίθμους δρομολόγησης, με μικρή βέβαια απόκλιση. Δεύτερη σε σειρά επιτυχίας έρχεται η Dropout Δρομολόγηση. Το αποτέλεσμα αυτό εκτιμάται, όπως και προηγουμένως, ότι οφείλεται στο γεγονός ότι οι δύο αυτοί αλγόριθμοι δρομολόγησης (Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου, Dropout Δρομολόγηση) δίνουν απόλυτη έμφαση στην πιθανότητα ύπαρξης της οντότητας που αντιπροσωπεύει κάθε κάψουλα. Αντιθέτως, οι δύο άλλοι αλγόριθμοι δρομολόγησης (Δρομολόγηση Υποσυνόλου, RANSAC Δρομολόγηση) δίνουν μεγάλη έμφαση στον καθορισμό των καψουλών που "συμφωνούν" περισσότερο μεταξύ τους, και μόνο τότε λαμβάνουν υπόψην την πιθανότητα ύπαρξης της οντότητας. Αντιθέτως, το SVHN είναι ένα σύνολο δεδομένων, που μπορεί να θεωρηθεί επέκταση του MNIST, με τα δείγματα να είναι έγχρωμες εικόνες ψηφίων από φυσικές σκηνές (Google Street View). Αυτό και μόνο τις καθιστά αυτομάτως πιο σύνθετες από αυτές του MNIST. Επομένως, αρκετές κάψουλες μπορεί να αντιστοιχούν σε οντότητες των οποίων η ύπαρξη σε μια εικόνα ενδεχομένως να μην παίζει σημαντικό ρόλο για τον καθορισμό του αποτελέσματος της κατηγοριοποίησης. Για παράδειγμα μια κάψουλα μπορεί να ευθύνεται για την αναγνώριση μιας οντότητας η οποία βρίσκε-

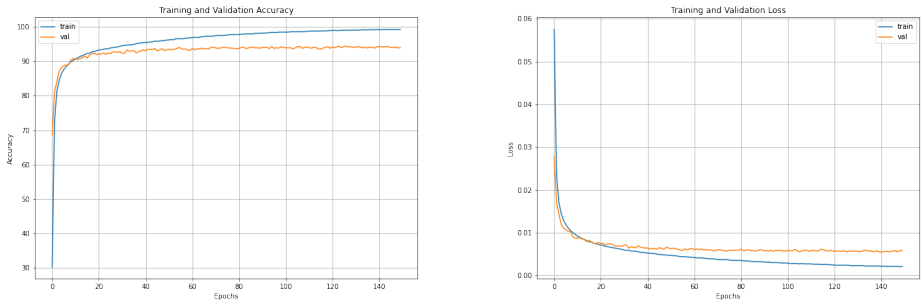
ται στο φόντο πολλών εικόνων σε διαφορετικούς προσανατολισμούς και να θεωρήθει ταιριαστός ο συνδυασμός αυτής με μια άλλη σε κατάλληλους προσανατολισμούς. Αυτό είναι ένα σφάλμα που μπορεί να συμβεί με αλγορίθμους δρομολόγησης που επιλέγουν καθολικά "σύμφωνες" μεταξύ τους κάψουλες οντότηων και αγνοούν τις υπόλοιπες, καθώς ενδέχεται να απορριφθούν κάψουλες με μεγάλη πιθανότητα ύπαρξης της οντότητας που αντιπροσωπεύουν. Από την άλλη η Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου θα λάβει όλες τις κάψουλες υπόψη με βάση το πόσο σημαντική και πιθανή έχει θεωρηθεί η ενεργοποίηση καθεμιάς.

Επομένως, πραγματοποιήθηκαν ορισμένα επιπλέον πειράματα (Πίνακας 6.24) σε βαθύτερα δίκτυα καψουλών με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου, ώστε να φανεί πως μπορεί η προσθήκη επιπλέον επιπέδων καψουλών να επηρεάσει την εκπαίδευση και κατ' επέκταση το ποσοστό ακρίβειας αναφορικά με το σύνολο ελέγχου του SVHN.

Πίνακας 6.24: Περαιτέρω μελέτη της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου στο SVHN σε βαθύτερα δίκτυα καψουλών

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Coordinate Addition	Error(%)
CapsNet-1	512	150	0.001	yes	–	5.08
CapsNet-1	256	150	0.003	yes	–	5.11
CapsNet-2	64	150	0.001	yes	–	**4.38**
CapsNet-3	32	200	0.001	no	–	4.53
CapsNet-4	32	200	0.001	no	–	5.34
CapsNet-4	32	200	0.001	yes	–	5.20
CapsNet-5	8	200	0.001	yes	no	5.25
CapsNet-5	8	200	0.0002	yes	no	5.37
CapsNet-5	8	200	0.005	yes	no	4.83
CapsNet-5	16	200	0.001	no	yes	6.17

Αν και τα MNIST και SVHN είναι σύνολα δεδομένων για αναγνώριση ψηφίων, το γεγονός ότι το απλό μοντέλο πετυχαίνει εξαιρετικά αποτελέσματα στο MNIST, δε σημαίνει ότι το ίδιο δίκτυο αρκεί για να επιτευχθεί καλή επίδοση και στο SVHN. Δεδομένου μάλιστα ότι το SVHN περιέχει πολύ πιο πολύπλοκες εικόνες ψηφίων όπως αναφέραμε και παραπάνω, αναμενόταν ότι μπορεί να είναι απαραίτητη η χρήση ενός βαθύτερου δικτύου έτσι ώστε οι κάψουλες να αναγκάστουν να μάθουν πιο πολύπλοκα χαρακτηριστικά των ψηφίων εισόδου. Παρατηρήθηκε λοιπόν, ότι προσθέτοντας ένα ακόμη πλήρως συνδεδεμένο επίπεδο καψουλών και πάλι τη χρήση δικτύου ανακατασκευής, μπορεί να επιτευχθεί μείωση του ποσοστού σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου στο 4.38%. Οι καμπύλες εκπαίδευσης του αντίστοιχου μοντέλου φαίνονται στο Σχήμα 6.19.



Σχήμα 6.19: Καμπύλες Εκπαίδευσης (ακρίβειας και απώλειας αντίστοιχα) καλύτερου μοντέλου στο SVHN με χρήση Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου

6.5.5 Πειραματικά αποτελέσματα αλγορίθμων δρομολόγησης στο CIFAR-10

Κατά την εκπαίδευση ενός νευρωνικού δικτύου καψουλών στο CIFAR-10, κάθε 32x32 έγχρωμη εικόνα του συνόλου εκπαίδευσης κανονικοποιείται και μετατοπίζεται κατά 2 pixel σε κάθε κατεύθυνση με χρήση μηδενικής συμπλήρωσης (zero padding), δημιουργώντας εικόνες μεγέθους 36x36 pixel. Έπειτα, από κάθε 36x36 έγχρωμη εικόνα του συνόλου εκπαίδευσης περικόπτεται ένα τυχαίο 32x32 τμήμα στο οποίο εφαρμόζεται τυχαίος οριζόντιος καθρέπτισμος με πιθανότητα 0.5, δημιουργώντας τελικά κανονικοποιημένα δείγματα έγχρωμων εικόνων εκπαίδευσης μεγέθους 32x32 pixel. Αντιθέτως, στην περίπτωση του ελέγχου η διαδικασία που ακολουθείται είναι αρκέτα απλούστερη, καθώς κάθε έγχρωμη εικόνα του συνόλου ελέγχου του CIFAR-10 απλώς κανονικοποιείται, δημιουργώντας απλά κανονικοποιημένες εκδοχές των αρχικών έγχρωμων 32x32 εικόνων ελέγχου.

Εκπαιδεύσαμε το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο CIFAR-10 προκειμένου να δούμε πως καθένας εκ των αλγορίθμων δρομολόγησης προσεγγίζει το συγκεκριμένο σύνολο δεδομένων. Τα αποτελέσματα των πειραμάτων με χρήση διαφορετικών υπερπαραμέτρων φαίνονται παρακάτω:

Πίνακας 6.25: Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου στο CIFAR-10

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Error(%)
CapsNet-1	512	250	0.001	yes	**17.44**
CapsNet-1	256	250	0.003	yes	17.84

Όπως φαίνεται στον Πίνακα 6.25, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο CIFAR-10, με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 17.44% στο σύνολο ελέγχου του.

Πίνακας 6.26: Dropout Δρομολόγηση στο CIFAR-10

Network	Batch Size	Epochs	Learning Rate	Reconstruction	p	Error(%)
CapsNet-1	512	250	0.001	yes	0.2	**17.93**
CapsNet-1	256	250	0.001	yes	0.4	18.10
CapsNet-1	512	250	0.001	yes	0.6	18.50
CapsNet-1	512	250	0.001	yes	0.8	19.42

Όπως φαίνεται στον Πίνακα 6.26, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο CIFAR-10, με χρήση της Dropout Δρομολόγησης με dropout πιθανότητα $p = 0.2$ και δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 17.93% στο σύνολο ελέγχου του CIFAR-10.

Πίνακας 6.27: Δρομολόγηση Υποσυνόλου στο CIFAR-10

Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	Similarity	Error(%)
CapsNet-1	512	700	0.001	yes	0.3	MSE	23.35
CapsNet-1	512	700	0.001	yes	0.5	MSE	22.98
CapsNet-1	512	700	0.001	yes	0.7	MSE	**20.70**
CapsNet-1	512	700	0.001	yes	0.7	Cosine	22.63

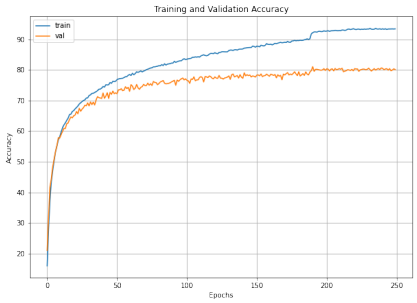
Όπως φαίνεται στον Πίνακα 6.27, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο CIFAR-10, με χρήση της Δρομολόγησης Υποσυνόλου με κλάσμα υποσυνόλου $S = 0.7$ και την ευκλείδεια απόσταση ως κριτήριο συμφωνίας και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 20.70% στο σύνολο ελέγχου του CIFAR-10.

Πίνακας 6.28: RANSAC Δρομολόγηση στο CIFAR-10

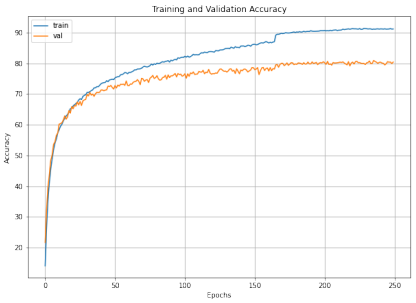
Network	Batch Size	Epochs	Learning Rate	Reconstruction	S	H	Similarity	Error(%)
CapsNet-1	256	150	0.001	yes	0.5	10	MSE	**19.33**
CapsNet-1	256	150	0.001	yes	0.7	10	MSE	19.92

Όπως φαίνεται στον Πίνακα 6.28, εκπαιδεύοντας το δίκτυο καψουλών της απλής αρχιτεκτονικής αναφοράς CapsNet-1 πάνω στο CIFAR-10, με χρήση της RANSAC Δρομολόγησης με κλάσμα υποσυνόλου $S = 0.5$ και $H = 10$ υποθέσεις, καθώς και χρήση δικτύου αποκωδοικοποίησης Σχήμα 6.7 για ανακατασκευή, επιτυγχάνεται ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 19.33% στο σύνολο ελέγχου του CIFAR-10.

Στα γραφήματα των εικόνων 6.20 και 6.21 φαίνονται, για κάθε άλγόριθμο δρομολόγησης στην αρχιτεκτονική CapsNet-1, οι καμπύλες εκπαίδευσης, ακρίβειας (Accuracy) και απώλειας (Loss) αντίστοιχα, των μοντέλων που επιτυγχάνουν το μικρότερο ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του CIFAR-10.

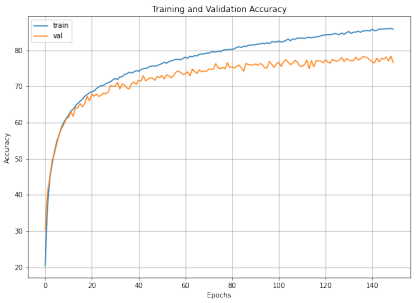


(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



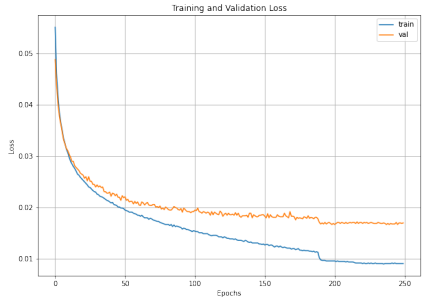
(b) Dropout Δρομολόγηση

(c) Δρομολόγηση Υποσυνόλου

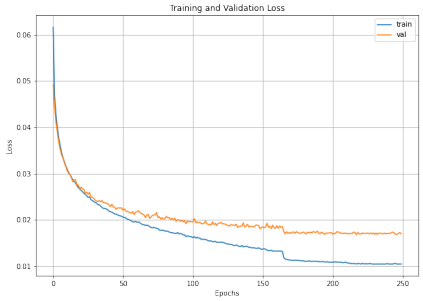


(d) RANSAC Δρομολόγηση

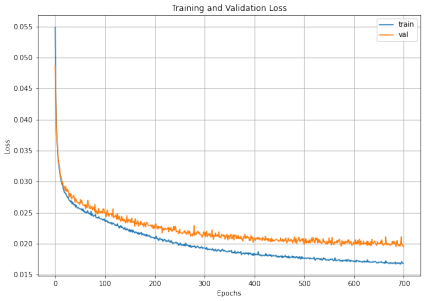
Σχήμα 6.20: Καμπύλες Εκπαίδευσης (Accuracy) ανά αλγόριθμο Δρομολόγησης στο CIFAR-10



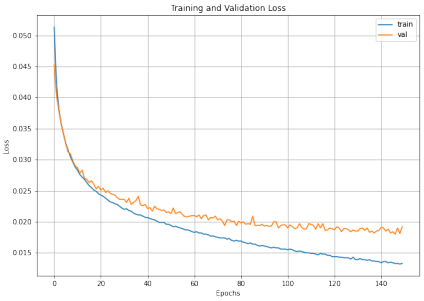
(a) Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου



(b) Dropout Δρομολόγηση



(c) Δρομολόγηση Υποσυνόλου



(d) RANSAC Δρομολόγηση

Σχήμα 6.21: Καμπύλες Εκπαίδευσης (Loss) ανά αλγόριθμο Δρομολόγησης στο CIFAR-10

Πίνακας 6.29: Σύγκριση Αλγορίθμων Δρομολόγησης στο CIFAR-10 με την απλή αρχιτεκτονική αναφοράς CapsNet-1

Routing Method	Network	Error(%)
Probabilistic Weighted Average	CapsNet-1	**17.44**
Dropout	CapsNet-1	17.93
Subset	CapsNet-1	20.70
RANSAC	CapsNet-1	19.33

Στην περίπτωση του CIFAR-10, η Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου επιτυγχάνει ποσοστό σφάλματος ακρίβειας κατηγοριοποίησης 17.44% στο σύνολο ελέγχου το οποίο είναι το μικρότερο συγκριτικά με τους υπόλοιπους αλγορίθμους δρομολόγησης. Δεύτερη σε σειρά επιτυχίας έρχεται η Dropout Δρομολόγηση. Όπως περιγράφηκε και στις προηγούμενες ενότητες, το αποτέλεσμα αυτό εκτιμάται, όπως και προηγουμένως, ότι οφείλεται στο γεγονός ότι οι δύο αυτοί αλγόριθμοι δρομολόγησης (Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου, Dropout Δρομολόγηση) δίνουν απόλυτη έμφαση στην πιθανότητα ύπαρξης της οντότητας που αντιπροσωπεύει κάθε κάψουλα. Αντιθέτως, οι δύο άλλοι αλγόριθμοι δρομολόγησης (Δρομολόγηση Υποσυνόλου, RANSAC Δρομολόγηση) δίνουν μεγάλη έμφαση στον καθορισμό των καψουλών που "συμφωνούν" περισσότερο μεταξύ τους, και μόνο τότε λαμβάνουν υπόψην την πιθανότητα ύπαρξης της οντότητας. Αντιθέτως, το CIFAR-10 είναι ένα σύνολο δεδομένων, με τα δείγματα να είναι έγχρωμες εικόνες που είναι αρκετά σύνθετες. Επομένως προκύπτει παρόμοιο πρόβλημα με αυτό που περιγράψαμε για το SVHN αλλά σε πολύ μεγαλύτερο βαθμό. Η Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου, σε σχέση με τους αλγορίθμους δρομολόγησης που δίνουν βάση στη "συμφωνία", είναι πιο κοντά στη λογική ένος κλασσικού συνελικτικού δικτύου, και γι' αυτό και όπως ήταν αναμενόμενο, επιτυγχάνει καλύτερα αποτελέσματα σε σύν-

θετα σύνολα δεδομένων όπως το CIFAR-10. Ωστόσο, και εδώ φαίνεται όπως είναι ήδη γνωστό ότι τα νευρωνικά δίκτυα με κάψουλες δυσκολεύονται να εκπαιδευτούν και να επιτύχουν ορθή κατηγοριοποίηση στο CIFAR-10.

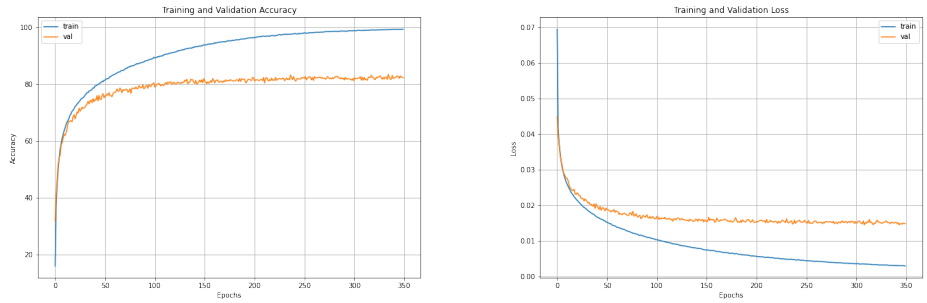
Αρκετές κάψουλες μπορεί να αντιστοιχούν σε οντότητες των οποίων η ύπαρξη σε μια εικόνα ενδεχομένως να μην παίζει σημαντικό ρόλο για τον καθορισμό του αποτελέσματος της κατηγοριοποίησης. Για παράδειγμα μια κάψουλα μπορεί να ευθύνεται για την αναγνώριση μιας οντότητας η οποία βρίσκεται στο φόντο πολλών εικόνων σε διαφορετικούς προσανατολισμούς και να θεωρήθει ταιριαστός ο συνδυασμός αυτής με μια άλλη σε κατάλληλους προσανατολισμούς. Αυτό είναι ένα σφάλμα που μπορεί να συμβεί με αλγορίθμους δρομολόγησης που επιλέγουν καθολικά "σύμφωνες" μεταξύ τους κάψουλες οντότηων και αγνοούν τις υπόλοιπες, καθώς ενδέχεται να απορριφθούν κάψουλες με μεγάλη πιθανότητα ύπαρξης της οντότητας που αντιπροσωπεύουν. Από την άλλη η Πιθανοτική Δρομολόγηση Σταθμισμένου Μέσου θα λάβει όλες τις κάψουλες υπόψη με βάση το πόσο σημαντική και πιθανή έχει θεωρηθεί η ενεργοποίηση καθεμιάς.

Επομένως, πραγματοποιήθηκαν ορισμένα επιπλέον πειράματα (Πίνακας 6.30) σε βαθύτερα δίκτυα καψουλών με χρήση της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου, ώστε να φανεί πως μπορεί η προσθήκη επιπλέον επιπέδων καψουλών να επηρεάσει την εκπαίδευση και κατ' επέκταση το ποσοστό ακρίβειας αναφορικά με το σύνολο ελέγχου του CIFAR-10.

Πίνακας 6.30: Περαιτέρω μελέτη της Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου στο CIFAR-10 σε βαθύτερα δίκτυα καψουλών

Network	Batch Size	Epochs	Learning Rate	Reconstruction	Coordinate Addition	Error(%)
CapsNet-1	512	250	0.001	yes	–	17.44
CapsNet-1	256	250	0.003	yes	–	17.84
CapsNet-2	64	350	0.001	yes	–	**15.35**
CapsNet-3	32	400	0.001	no	–	15.94
CapsNet-4	32	400	0.001	no	–	19.41
CapsNet-4	32	400	0.001	yes	–	18.84
CapsNet-5	16	400	0.005	yes	no	23.14
CapsNet-5	16	400	0.005	no	yes	21.52

Παρατηρούμε (όπως και στα Fashion-MNIST και SVHN) ότι προσθέτοντας ένα ακόμη πλήρως συνδεδεμένο επίπεδο καψουλών και πάλι τη χρήση δικτύου ανακατασκευής, μπορεί να επιτευχθεί μείωση του ποσοστού σφάλματος ακρίβειας κατηγοριοποίησης στο σύνολο ελέγχου του CIFAR-10, κατά $\approx 2.1\%$. Το CIFAR-10, όπως προαναφέραμε αποτελείται από αρκετά σύνθετες εικόνες αποτελώντας ένα δύσκολο πρόβλημα αναγνώρισης. Αυτό λοιπόν το βαθύτερο δίκτυο βοηθάει τις κάψουλες να μάθουν πιο πολύπλοκα χαρακτηριστικά των εικόνων εισόδου με αποτέλεσμα τελικά να επιτυγχάνεται ποσοστό σφάλματος 15.35%. Οι καμπύλες εκπαίδευσης του αντίστοιχου μοντέλου φαίνονται στο Σχήμα 6.22. Ωστόσο, φάνηκε ότι αν και υπάρχει βελτίωση, το CIFAR-10 συνεχίζει να αποτελεί ένα σύνολο δεδομένων στο οποίο τα νευρωνικά δίκτυα με κάψουλες, με χρήση των προτεινόμενων μεθόδων (αλλά και με χρήση άλλων μεθόδων τι βιβλιογραφίας των καψουλών) , έχουν εμφάνη αδυναμία ως προς την εκπαίδευση και κατ'επέκταση την κατηγοριοποίηση.



Σχήμα 6.22: Καμπύλες Εκπαίδευσης (ακρίβειας και απώλειας αντίστοιχα) καλύτερου μοντέλου στο CIFAR-10 με χρήση Πιθανοτικής Δρομολόγησης Σταθμισμένου Μέσου

6.5.6 Σύγκριση με υπάρχουσες μεθόδους Νευρωνικών Δικτύων με Κάψουλες

Στον συγκεντρωτικό Πίνακα 6.31 παρουσιάζονται ανά σύνολο δεδομένων τα αποτελέσματα σε ποσοστά σφάλματος μεθόδων που έχουν χρησιμοποιηθεί στο παρελθόν σε νευρωνικά δίκτυα καψουλών σε σύγκριση με τους αλγορίθμους δρομολόγησης που σχεδιάστηκαν και προτείνονται στην παρούσα ερευνητική εργασία.

Πίνακας 6.31: Σύγκριση σφάλματος ελέγχου κατηγοριοποίησης των προτεινόμενων μεθόδων με την ήδη υπάρχουσα βιβλιογραφία στα Νευρωνικά Δίκτυα με Κάψουλες

Previous Methods	**smallNORB** Error(%)	**Fashion-MNIST** Error(%)	**SVHN** Error(%)	**CIFAR-10** Error(%)
HitNet	–	7.7	5.5	26.7
DCNet	5.57	5.36	4.42	17.37
MS-Caps	–	7.3	–	24.3
Dynamic	2.7	–	4.3	10.6
Nair	–	10.2	8.94	32.47
FRMS	2.6	6.0	–	15.6
MaxMin	–	7.93	–	24.08
KernelCaps	–	–	8.6	22.3
FREM	2.2	6.2	–	14.3
EM-Routing	1.8	–	–	11.9
VB-Routing	1.6	5.2	3.9	11.2
Proposed Methods				
Probabilistic Weighted Average	4.68	**6.16**	**4.38**	**15.35**
Dropout	4.6	6.70	5.32	17.93
Subset	**3.96**	7.25	5.81	20.7
RANSAC	4.19	7.62	5.87	19.33

Παρατηρούμε ότι οι επιδόσεις των προτεινόμενων μας μεθόδων στα σύνολα δεδομένων smallNORB, Fashion-MNIST, SVHN και CIFAR-10 είναι αρκετά ικανοποιητικές, καθώς ξεπερνάνε τις επιδόσεις πολλών μεθόδων που έχουν σχεδιαστεί στο παρελθόν σε δίκτυα με κάψουλες. Μάλιστα σε πολλές περιπτώσεις, οι μέθοδοι που υλοποιήθηκαν στην παρούσα ερευνητική εργασία πλησιάζουν αρκετά τα ποσοστά των μεθόδων που δείχνουν να πετυχαίνουν τα καλύτερα αποτελέσματα ανά σύνολο δεδομένων.

Κεφάλαιο 7

Συμπεράσματα και Μελλοντικές Κατευθύνσεις

7.1 Συμπεράσματα

Στην παρούσα εργασία εξετάσαμε την επίδοση μιας νέας κατηγορίας αλγορίθμων δρομολόγησης που χρησιμοποιούν ως βάση μια προσέγγιση με βάση την πιθανοτική στάθμιση μέσου, κάθε φορά εμπλουτισμένη με κλασικές ιδέες από τον χώρο της μηχανικής μάθησης και της όρασης υπολογιστών, ειδικά τροποποιημένες ώστε να προσαρμόζονται στη θεωρία των δικτύων καψουλών.

Αρχικά, είναι πολύ σημαντικό να αναφέρουμε ότι μέσω των πειραμάτων αυτών διαπιστώθηκε η καθοριστικής σημασίας χρήση δικτύου αποκωδικοποίησης για ανακατσκευή της εικόνας εισόδου. Η ένταξη της ιδέας της ανακατασκευής στην συνάρτηση απωλειών έδειξε να βοηθάει το δίκτυο να μάθει καλύτερα τις θέσεις στις οποίες βρίσκονται οι οντότητες που εντοπίζει, μειώνοντας έτσι σημαντικά το σφάλμα κατηγοριοποίησης σε κάθε περίπτωση που εξετάστηκε.

Ακόμη, μέσω της μελέτης αυτών των αλγορίθμων διαπιστώθηκε ότι κρίνεται πολύ σημαντικό το είδος των υπό εξέταση δεδομένων. Ανάλογα με τα δεδομένα, φαίνεται κατά πόσο κρίνεται σημαντική η χρήση ή μη ενός μηχανισμού που θα δρομολογεί τις κάψουλες ενός χαμηλότερου επιπέδου στο επόμενο, με κύριο κριτήριο την μετάξυ τους "συμφωνία", ορίζοντας ως μετρική για τον υπολογισμό της "συμφωνίας", είτε την ευκλείδεια απόσταση είτε την ομοιότητα συνημιτόνου. Παρατηρήθηκε ότι ένα δίκτυο μπορεί να μάθει καλύτερα τα μοτίβα που βρίσκονται σε σύνολα δεδομένων όπως τα Fashion-MNIST, SVHN και το CIFAR-10 όταν δίνεται σημασία μόνον στην πιθανότητα ύπαρξης των υπο-οντοτήτων που συνθέτουν μια υψηλότερου επιπέδου οντότητα. Αντιθέτως, σε σύνολα δεδομένων όπως το smallNORB, που περιέχει πολλαπλές οπτικές γωνίες των ίδιων αντικειμένων και το δίκτυο καλείται ουσιαστικά να λύσει ένα πρόβλημα που θυμίζει αναγνώριση σχήματος, καλύτερες επιδόσεις δείχνουν αλγόριθμοι όπως αυτός της Δρομολόγησης Υποσυνόλου, που δίνουν έμφαση πρώτα στη "συμφωνία", προκειμένου να καθορίσουν το τελικό υποσύνολο καψουλών που τελικά θα δρομολογηθεί με βάση τις αντίστοιχες πιθανότητες ύπαρξης.

Επιβεβαιώθηκε μάλιστα ότι τα νευρωνικά δίκτυα καψουλών δυσκολεύονται σε σύνολα δεδομένων που αποτελούνται από πολύπλοκες εικόνες (π.χ. CIFAR-10). Συνήθως αυτή η δυσκολία συναντάται όταν πρόκειται για σύνολα των οποίων οι εικόνες, περιέχουν επίπλεον ασήμαντες οντότητες πέραν του βασικού αντικειμένου που καλείται ένα δίκτυο να αναγνωρίσει.

Επιπλέον, να σημειωθεί ότι μέχρι τώρα πολλοί από τους αλγορίθμους δρομολόγησης που έχουν προταθεί αποτελούν επαναληπτικές διαδικασίες, δίχως τη σύγκλιση των οποίων το σφάλμα κατηγοριοποίησης είναι αρκετά αυξημένο. Με τον κατάλληλο αριθμό επαναλήψεων, ναι μεν επιτυγχάνεται σημάντικη μείωση του σφάλματος αυτού, αλλά αυξάνεται η υπολογιστική πολυπλοκότητα της εκπαίδευσης. Αντιθέτως, οι αλγόριθμοι που προτείνονται στην παρούσα εργασία (εξαιρουμένης της RANSAC Δρομολόγησης) αποτελούν μη επαναλαμβανόμενες διαδικάσιες, που δρομολογούν άμεσα τις κάψουλες ενός χαμηλότερου επιπέδου υπολογίζοντας απευθείας τις προβλέψεις αναφορικά με τις κάψουλες του ανωτέρου επιπέδου. Οι προτεινόμενοι αλγόριθμοι, μιας και δεν εισάγουν νέες εκπαιδευόμενες παραμέτρους, συνεισφέρουν στη μείωση της υπολογιστικής πολυπλοκότητας τόσο της εκπαίδευσης όσο και της απλής διάσχισης του δικτύου, και επιτυγχάνουν επιδόσεις που, πρώτον, ξεπερνάνε κατά πολύ τις επιδόσεις των επαναληπτι-

κών αλγορίθμων δρομολόγησης αν αυτοί εφαρμοστούν για μια μόνον επανάληψη, και δεύτερον πλησιάζουν σε πολύ μεγάλο βαθμό τις επιδόσεις των επαναληπτικών αυτών αλγορίθμων όταν χρησιμοποιηθεί βέλτιστος ως προς τη σύγκλιση αριθμός επαναλήψεων. Επομένως, αυτό μας δείχνει ότι ο αλγόριθμος δρομολόγησης μπορεί να λειτουργήσει επιτυχώς και δίχως να αποτελεί μια επαναληπτική διαδικασία.

7.2 Μελλοντικές Κατευθύνσεις

Όπως ήδη αναφέρθηκε, οι μη επαναληπτικοί αλγόριθμοι δρομολόγησης που προτάθηκαν πλησιάζουν σε πολύ μεγαλό βαθμό τις επιδόσεις που επιτυγχάνονται από τους επαναληπτικούς αλγορίθμους δρομολόγησης που βρίσκονται στη βιβλιογραφία των νευρωνικών δικτύων με κάψουλες. Αυτό είναι ένας πολύ ενθαρρυντικός παραγόντας για τον σχεδιασμό αλγορίθμων που δρομολογούν απευθείας της κάψουλες ενός επιπέδου στο επόμενο, δίχως να απαιτούν κάποιο πλήθος επαναληψέων.

Τώρα, αναφορικά με τη δυσκολία που εμφανίζουν τα νευρωνικά δίκτυα με κάψουλες σε πολύπλοκα σύνολα δεδομένων όπως το CIFAR-10, θα ήταν χρήσιμη η τροποποίηση των κλασικών αρχιτεκτονικών. Πιο συγκεκριμένα, προτείνουμε την εισαγωγή επιπλέον συνελικτικών επιπέδων, προκειμένου να εξαχθούν υψηλότερου επίπεδου χαρακτηριστικά που θα χρησιμοποιηθούν ως είσοδοι για τα επίπεδα καψουλών.

Επιπλεόν, να αναφέρουμε ότι η RANSAC Δρομολόγηση είναι μια πολύ κοστοβόρη υπολογιστικά διαδικασία. Με αύξηση του αριθμού των υποθέσεων, αναμένουμε ότι θα επιλέγονται πιο εύστοχοι συνδυασμοί καψουλών αλλά παράλληλα θα παρουσιάζεται εκτόξευση της υπολογιστικής πολυπλοκότητας, γεγονός που απαιτεί επιπλέον υπολογιστικούς πόρους. Θεωρούμε λοιπόν πως οι επίδοσεις που έχουμε παρουσιάσει αναφορικά με αυτόν τον αλγόριθμο δρομολόγησης δεν είναι και τόσο ενδεικτικές μιας και χρησιμοποιήθηκαν τιμές για το πλήθος των υποθέσεων, τέτοιες ώστε να συμβαδίζουν με τις δυνατότητες του χρησιμοποιούμενου σύστηματος. Θα φαίνοταν χρήσιμο λοιπόν στο μέλλον, με απαραίτητη προϋπόθεση την χρήση ενός συστήματος μεγαλύτερης υπολογιστικής ισχύος, να πραγματοποιηθεί πειραματισμός με αύξηση του πλήθους των υποθέσεων της RANSAC Δρομολόγησης, έτσι ώστε να διαπιστωθεί πόσο καλές επιδόσεις μπορεί να επιτύχει σε βάρος της πολυπλοκότητας. Βέβαια, αντ' αυτού φαίνεται πιο ρεαλιστικό να τροποποιηθεί η παρούσα RANSAC Δρομολόγηση εισάγοντας κάποιο μη τυχαίο κριτήριο για την επιλογή των συνδυασμών των καψουλών με τρόπο τέτοιο ώστε να αρκεί ένα μικρό πλήθος υποθέσεων.

Τέλος, εκτιμάμαι σε μεγάλο βαθμό ότι μπορεί να γίνει μια πιο εξαντλητική αναζήτηση στον χώρο των υπερπαραμέτρων. Ναι μεν χρησιμοποιήθηκε μεγάλο πλήθος συνδυασμών για τις τιμές τόσο των υπερπαραμέτρων της εκπαίδευσης και της αρχιτεκτονικής γενικότερα όσο και αυτών που αφορούν τους προτεινόμενους αλγορίθμους δρομολόγησης, αλλά πάντα υπάρχει η περίπτωση να βρεθεί κάποιος συνδυασμός που βελτιστοποιεί τις επιδόσεις αλλά έχει παραληφθεί.

Βιβλιογραφία

[1] Geoffrey E Hinton, Zoubin Ghahramani, and Yee Whye Teh. Learning to parse images. In S. Solla, T. Leen, and K. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12. MIT Press, 1999.

[2] Geoffrey E. Hinton, Alex Krizhevsky, and Sida D. Wang. Transforming auto-encoders. In *ICANN*.

[3] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. Dynamic routing between capsules, 2017.

[4] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society: Series B*, 39:1–38, 1977.

[5] Radford M. Neal and Geoffrey E. Hinton. *A View of the Em Algorithm that Justifies Incremental, Sparse, and other Variants*, pages 355–368. Springer Netherlands, Dordrecht, 1998.

[6] Geoffrey Hinton, Sara Sabour, and Nicholas Frosst. Matrix capsules with em routing. 2018.

[7] Fabio De Sousa Ribeiro, Francesco Calivá, Mark Swainson, Kjartan Gudmundsson, Georgios Leontidis, and Stefanos Kollias. Deep bayesian self-training. *Neural Computing and Applications*, 32(9):4275–4291, 2020.

[8] Fabio De Sousa Ribeiro, Georgios Leontidis, and Stefanos D Kollias. Capsule routing via variational bayes. In *AAAI*, pages 3749–3756, 2020.

[9] Fabio De Sousa Ribeiro, Georgios Leontidis, and Stefanos Kollias. Introducing routing uncertainty in capsule networks. *Advances in Neural Information Processing Systems*, 33:6490–6502, 2020.

[10] Vittorio Mazzia, Francesco Salvetti, and Marcello Chiaberge. Efficient-capsnet: capsule network with self-attention routing. *Scientific reports*, 11, 2021.

[11] Dimitrios Kollias, Miao Yu, Athanasios Tagaris, Georgios Leontidis, Andreas Stafylopatis, and Stefanos Kollias. Adaptation and contextualization of deep neural network models. In *2017 IEEE symposium series on computational intelligence (SSCI)*, pages 1–8. IEEE.

[12] D Kollias, N Bouas, Y Vlaxos, V Brillakis, M Seferis, I Kollia, L Sukissian, J Wingate, and S Kollias. Deep transparent prediction through latent representation analysis. *arXiv preprint arXiv:2009.07044*, 2020.

[13] Dimitris Kollias, Y Vlaxos, M Seferis, Ilianna Kollia, Levon Sukissian, James Wingate, and S Kollias. Transparent adaptation in deep medical image diagnosis. In *International Workshop on the Foundations of Trustworthy AI Integrating Learning, Optimization and Reasoning*, pages 251–267. Springer, 2020.

[14] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. http://www.deeplearningbook.org.

[15] Frank Rosenblatt. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65 6:386–408, 1958.

[16] Wikipedia contributors. Perceptron — Wikipedia, the free encyclopedia. https://en.wikipedia.org/w/index.php?title=Perceptron&oldid=1037601316, 2021. [Online; accessed 16-December-2021].

[17] Berke Akkaya and Nurdan Çolakoğlu. Comparison of multi-class classification algorithms on early diagnosis of heart diseases. 09 2019.

[18] Pierre Comon and Christian Jutten. *Handbook of Blind Source Separation: Independent Component Analysis and Applications*. Academic Press, Inc., USA, 1st edition, 2010.

[19] Kunihiko Fukushima. Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1(2):119–130, 1988.

[20] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.

[21] Yann LeCun and Yoshua Bengio. *Convolutional Networks for Images, Speech, and Time Series*, page 255–258. MIT Press, Cambridge, MA, USA, 1998.

[22] Y. LeCun, P. Haffner, L. Bottou, and Y. Bengio. Object recognition with gradient-based learning. In D. Forsyth, editor, *Feature Grouping*. Springer, 1999. (original is ps).

[23] David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams. Learning Representations by Back-propagating Errors. *Nature*, 323(6088):533–536, 1986.

[24] Y. Bengio. Learning deep architectures for ai. *Foundations*, 2:1–55, 01 2009.

[25] Nathan Hubens. Deep inside: Autoencoders. 2018.

[26] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[27] JAmes Matey. The nist irisdaily dataset: Description and initial analysis, 2021-05-04 04:05:00 2021.

[28] Yann LeCun, Fu Jie Huang, and Léon Bottou. Learning methods for generic object recognition with invariance to pose and lighting. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2:II97–II104, 2004. Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004 ; Conference date: 27-06-2004 Through 02-07-2004.

[29] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017.

[30] Alex Krizhevsky. Learning multiple layers of features from tiny images. pages 32–33, 2009.

[31] Antonio Torralba, Rob Fergus, and William T. Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.

[32] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929–1958, 2014.

[33] Tijmen Tieleman. The affnist dataset. `https://www.cs.toronto.edu/~tijmen/affNIST/`, 2013.

[34] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.

Παράρτημα Α

Ευρετήριο συμβολισμών

$A \to B$: συνάρτηση από το πεδίο A στο πεδίο B.

$\|\cdot\|$: νόρμα διανύσματος.

$\mathbb{R}$: το σύνολο των πραγματικών αριθμών