ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Τομέας Σημάτων, Ελέγχου και Ρομποτικής
Εργαστήριο Όρασης Υπολογιστών, Επικοινωνίας Λόγου και Επεξεργασίας Σημάτων

Αναγνώριση Ανθρώπινης Δράσης και Χειρονομιών χρησιμοποιώντας Συνελικτικά και Αναδρομικά Νευρωνικά Δίκτυα

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΘΕΟΔΩΡΟΣ Μ. ΠΙΣΣΑΣ
Ηλεκτρολόγος Μηχανικός και Μηχανικός Ηλεκτρονικών Υπολογιστών

Επιβλέπων: Πέτρος Μαραγκός
Καθηγητής ΕΜΠ

Αθήνα, Ιούλιος 2017

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
Τομέας Σημάτων, Ελέγχου και Ρομποτικής
Εργαστήριο Όρασης Υπολογιστών, Επικοινωνίας Λόγου και Επεξεργασίας Σημάτων

Αναγνώριση Ανθρώπινης Δράσης και Χειρονομιών χρησιμοποιώντας Συνελικτικά και Αναδρομικά Νευρωνικά Δίκτυα

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΘΕΟΔΩΡΟΣ Μ. ΠΙΣΣΑΣ

Επιβλέπων: Πέτρος Μαραγκός
Καθηγητής ΕΜΠ

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την

.....	.....	.....
Πέτρος Μαραγκός Καθηγητής ΕΜΠ	Γεράσιμος Ποταμιάνος Αναπληρωτής Καθηγητής Πανεπιστημίου Θεσσαλίας	Κωνσταντίνος Τζαφέστας Επίκουρος Καθηγητής ΕΜΠ

Αθήνα, Ιούλιος 2017

.....

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

Περίληψη

Στην παρούσα διπλωματική επιδιώκεται να επιλυθεί το πρόβλημα της αναγνώρισης δράσεων και χειρονομιών χρησιμοποιώντας μοντέλα Τεχνητών Νευρωνικών Δικτύων. Συγκεκριμένα εξετάζονται δύο κατηγορίες εξειδικευμένων νευρωνικών μοντέλων τα Συνελικτικά Νευρωνικά Δίκτυα και τα Αναδρομικά Νευρωνικά Δίκτυα. Τα πρώτα έχουν τη δυνατότητα να εντοπίζουν και να εξάγουν τοπικά χωρικά ή χωρικά-χρονικά χαρακτηριστικά από βίντεο, ενώ τα δεύτερα είναι κατάλληλα για τη συνολική χρονική μοντελοποίηση μίας δράσης. Προκειμένου να εξετασθεί η συνεισφορά των δύο κατηγοριών μοντέλων διεξήχθησαν πειράματα για τρία διαφορετικά μοντέλα: Ένα Νευρωνικό Δίκτυο Τρισδιάστατης Συνέλιξης (3D-CNN) το οποίο εξάγει μόνο τοπικά χωροχρονικά χαρακτηριστικά από τμήματα (δηλαδή έναν αριθμό συνεχόμενων καρέ) ενός βίντεο και δύο Αναδρομικά Νευρωνικά Δίκτυα που αποτελούνται από στρώματα νευρώνων Μακράς και Βραχείας Μνήμης (Long and Short Term Memory ή LSTM), εκ των οποίων, το πρώτο (3D-CNN-LSTM) χρησιμοποιεί τα τοπικά χωροχρονικά χαρακτηριστικά που εξάγει ένα Νευρωνικό Δίκτυο Τρισδιάστατης Συνέλιξης από τμήματα ενός βίντεο και το δεύτερο (CNN-LSTM) χρησιμοποιεί τα χωρικά χαρακτηριστικά που εξάγει ένα Νευρωνικό Δίκτυο Δισδιάστατης Συνέλιξης (2D-CNN ή απλά CNN) από κάθε καρέ ενός βίντεο. Τα παραπάνω μοντέλα εκπαιδεύτηκαν και αξιολογήθηκαν επί τριών βάσεων δεδομένων μεσαίας κλίμακας (KTH, SKIG). Στη βάση SKIG, που περιέχει βίντεο με δυναμικές χειρονομίες, εκπαιδεύονται ξεχωριστά μοντέλα για δύο διαφορετικές τροπικότητες το RGB βίντεο και το βίντεο βάθους (Depth). Κατά συνέπεια, εκτιμάται η επίδραση του είδους της οπτικής πληροφορίας στην απόδοση των μοντέλων και αξιολογείται η βελτίωση που προσφέρει η σύμμειξη τους. Επιπλέον, στα πλαίσια των πειραμάτων γίνεται πειραματική αξιολόγηση της επίδρασης κάποιων εμπειρικά επιβεβαιωμένων μεθοδολογιών (Προγραμματισμός ρυθμού μάθησης και Επαύξηση Δεδομένων) και τεχνικών κανονικοποίησης (Dropout και Batch Normalization) που αποσκοπούν στην βελτίωση της ικανότητας γενίκευσης των μοντέλων καθώς και στην επιτάχυνση της διαδικασίας εκπαίδευσης. Τέλος, επιδιώχθηκε να ενσωματωθούν τα εκπαιδευμένα μοντέλα σε ένα σύστημα on-line αναγνώρισης χειρονομιών, το οποίο αναπτύχθηκε εντός του περιβάλλοντος του R.O.S (Robotics Operating System). Το σύστημα αυτό επιτρέπει την ταχύτατη επεξεργασία και αναγνώριση χειρονομιών (της τάξης μεγέθους των εκατοντάδων mseconds ανά χειρονομία) σε σχέση με άλλες κλασσικές μεθόδους αναγνώρισης που βασίζονται στην εξαγωγή κατασκευα-

σμένων χαρακτηριστικών, όπως οι πυκνές τροχιές, που απαιτούν σημαντικά περισσότερο χρόνο για τον υπολογισμό τους.

Λέξεις Κλειδιά:

Συνελικτικά Νευρωνικά Δίκτυα, Δισδιάστατη και Τρισδιάστατη Συνέλιξη, Νευρώνες Μακράς και Βραχείας Μνήμης, Αναδρομικά Νευρωνικά Δίκτυα, Αναγνώριση Ανθρώπινης Δράσης, Αναγνώριση Ανθρωπίνων Χειρονομιών, Επαύξηση Δεδομένων, On-line αναγνώριση χειρονομιών

Summary

In the context of this diploma thesis we approach problems pertaining to the field of human action and gesture recognition using Artificial Neural Networks. More specifically, we focus on the two dominant categories of such models, the convolutional and the recurrent neural networks. The first are able to extract local spatial or spatiotemporal features from videos while the latter are suitable for conducting the global temporal modelling of an action. Aiming to evaluate the contribution of these two types of neural models we experiment with three different model architectures: The 3D Convolutional Neural Network (3D-CNN) which extracts local spatiotemporal features from short clips of a video and two models that utilize a layer of L.S.T.M neurons (Long-Short Term Memory) that receives its input from either a 3D-CNN or a 2D-CNN. These three models were trained and evaluated on two medium size datasets (SKIG and KTH) that comprise of dynamic gestures and temporally limited human actions, respectively. Regarding the SKIG dataset, which entails both Depth and RGB video modalities, we trained one separate model for each modality, compared the results and also performed modality fusion which led to improvements in the models' ability to generalize. Furthermore, while training the models we experimented with various empirically validated methodologies such as Dropout, Batch Normalization, Learning Rate scheduling and Data Augmentation, which eased training and improved the accuracy of the final models. Finally, we implemented an online gesture recognition system using the models that were trained in the context of the off-line experiments. We observed that our system predicts remarkably faster in comparison two systems that require extracting hand-crafted features.

Keywords

Convolutional Neural Networks, 2D and 3D convolution, Neurons with Long-Short Term Memory, Recurrent Neural Networks, Human action recognition, Gesture Recognition, Data Augmentation, On-line gesture recognition

Ευχαριστίες

Αρχικά θα ήθελα να ευχαριστήσω θερμά τον Καθ. Πέτρο Μαραγκό, για την ευκαιρία που μου έδωσε να εκπονήσω την παρούσα διπλωματική. Οι διαλέξεις του αποτέλεσαν το έναυσμα για την ενασχόληση μου με τους τομείς της Όρασης Υπολογιστών και της Μηχανικής Μάθησης. Ακολούθως, θα ήθελα να ευχαριστήσω τους Βασίλη Πιτσικάλη, Πέτρο Κούτρα και Ισίδωρο Ροδομαγουλάκη για τις χρήσιμες συμβουλές τους. Επίσης ευχαριστώ όλα τα υπόλοιπα μέλη του εργαστηρίου Όρασης Υπολογιστών, Επικοινωνίας Λόγου και Επεξεργασίας Σημάτων για τη βοήθειά τους σε ότι τους ζητήθηκε. Δεδομένου πως η παρούσα εργασία σηματοδοτεί και το τέλος των φοιτητικών μου χρόνων, θα ήθελα να εκφράσω τις ευχαριστίες μου σε όλους τους φίλους και συμφοιτητές μου με τους οποίους περάσαμε μαζί τα χρόνια αυτά, μοιραστήκαμε εμπειρίες και ανταλλάξαμε ιδέες. Τέλος, θα ήθελα να εκφράσω την ευγνωμοσύνη μου στους γονείς μου και τον αδερφό μου για την έμπρακτη υποστήριξή τους όλα αυτά τα χρόνια.

Περιεχόμενα

Κατάλογος σχημάτων

Κεφάλαιο 1

Εισαγωγή

1.1 Όραση Υπολογιστών και Μηχανική Μάθηση

Ο θεμελιώδης στόχος της Όρασης Υπολογιστών, ως διεπιστημονικής περιοχής, είναι η εύρεση μιας εύρωστης και πλήρους συμβολικής περιγραφής των αντικειμένων που βρίσκονται εντός μίας εικόνας ή μιας αλληλουχίας εικόνων στο χρόνο (βίντεο). Ειδικότερα, η Όραση Υπολογιστών αποτελεί το σύνολο των μεθόδων και αλγορίθμων που αποσκοπούν στην εξαγωγή περιγραφητών από οπτική πληροφορία ώστε στη συνέχεια χρησιμοποιώντας τεχνικές μηχανικής μάθησης αλλά και την προϋπάρχουσα γνώση για το φυσικό κόσμο να ελεγχθούν και να προσδιοριστούν οι ιδιότητες μίας φυσικής σκηνής. Οι ιδιότητες αυτές μπορεί να αφορούν σε χαρακτηριστικά υφής ή κίνησης μέχρι στη σημασιολογική ερμηνεία μιας εν εξελίξει δράσης ή ακόμα και στην πρόβλεψη της μελλοντικής της έκβασης. Η Μηχανική Μάθηση αποτελείται από ένα ευρύ φάσμα αλγορίθμων οι οποίοι ποικίλουν από έναν απλό ταξινομητή Bayes μέχρι σύνθετα βαθιά νευρωνικά δίκτυα. Σκοπός της μεγάλης πλειοψηφίας των αλγορίθμων αυτών, είναι η εξαγωγή ουσιώδους και χρήσιμης πληροφορίας από μικρά έως πολύ μεγάλα σύνολα δεδομένων για να μπορέσουν αυτόματα να εκτελέσουν ταξινόμηση της πληροφορίας σε κλάσεις. Οι αλγόριθμοι Μηχανικής Μάθησης αποτελούν συχνά το συνδετικό κρίκο ανάμεσα στα χαρακτηριστικά που εξάγονται με τη βοήθεια της Όρασης Υπολογιστών και στον προσδιορισμό της υπό προσδιορισμό της ιδιότητας της φυσικής σκηνής που εξετάζεται. Η συνέργεια Όρασης Υπολογιστών και Μηχανικής Μάθησης αποτελεί μία διεπιστημονική περιοχή η οποία αλληλεπιδρά ενεργά με άλλους επιστημονικούς κλάδους όπως η Τεχνητή Νοημοσύνη, η Ρομποτική, τα Εφαρμοσμένα Μαθηματικά και η Νευροβιολογία. Επιπρόσθετα, αξίζει να σημειωθεί η ισχυρή εξάρτηση των συστημάτων Όρασης Υπολογιστών και Μηχανικής Μάθησης από την ύπαρξη αισθητήρων και καταγραφικών συσκευών που επιτρέπουν τη ζεύξη του αναλογικού με τον ψηφιακό κόσμο και πολλές φορές προσφέρουν πλούσιες μορφές οπτικής πληροφορίας. Ένα παράδειγμα αισθητήρα που έχει διευρύνει σημαντικά τις δυνατότητες των συστημάτων αναγνώρισης ανθρωπίνων δράσεων και χειρονομιών είναι ο αισθητήρας βάθους (π.χ Microsoft Kinect). Αντίστοιχα, ένα παράδειγμα καταγραφικής συσκευής

που προσφέρει πλούσια οπτική πληροφορία σχετικά με το εσωτερικό του ανθρώπινου σώματος είναι η μαγνητική τομογραφία (MRI), η οποία μπορεί να αξιοποιηθεί από ”έξυπνα” συστήματα βιοιατρικών εφαρμογών που συνδυάζουν τεχνικές τόσο της Όρασης Υπολογιστών όσο και της Μηχανικής Μάθησης. Άλλες ενδιαφέρουσες περιπτώσεις συνέργειας μεταξύ των δύο επιστημονικών περιοχών είναι:

- Ανάκτηση πληροφορίας (content extraction), ανάλυση του περιεχομένου και κατηγοριοποίηση εικόνων και βίντεο του διαδικτύου με σκοπό τη δημιουργία βάσεων δεδομένων και την εύκολη αναζήτηση και ανάκλησή τους βάσει του περιεχομένου.

- Υλοποίηση συστημάτων υποβοηθούμενης διαβίωσης (assisted leaving technology) με σκοπό τη διευκόλυνση ατόμων που αντιμετωπίζουν κινητικές δυσκολίες ή κάποια ασθένεια [59, 58, 57].

- Παρακολούθηση συναισθηματικής κατάστασης μέσω της καταγραφής και ανάλυσης της έκφρασης του προσώπου ή/και της στάσης του σώματος ενός ατόμου.

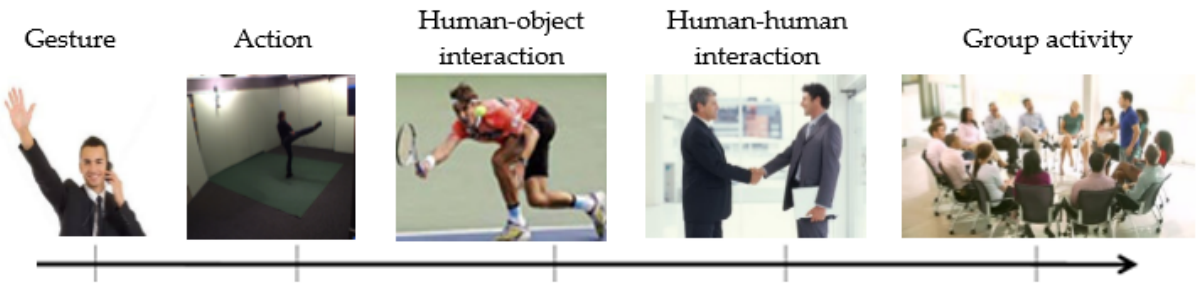
- Γρήγορη αλληλεπίδραση οχήματος-οδηγού μέσω χειρονομιών με σκοπό να μην αποσπάται η προσοχή του οδηγού [60].

1.2 Το Πρόβλημα της Αναγνώρισης Δράσης και Χειρονομιών

1.2.1 Ορισμοί προβλημάτων

Αρχικά με τον όρο ανθρώπινη δράση μπορεί να περιγραφεί ένα ευρύ φάσμα ενεργειών (Σχήμα 1.1), από απλές ή *πρωταρχικές* έως πολύ σύνθετες, που περιλαμβάνουν πολλαπλούς δράστες που αλληλεπιδρούν. Συνήθως μία δράση παρουσιάζει κάποια δομή υπό την έννοια ότι αποτελείται από την εκτέλεση κάποιων χρονικά διαδοχικών ή επικαλυπτόμενων απλούστερων δράσεων ή κινήσεων. Ένα χαρακτηριστικό παράδειγμα δράσης που εμφανίζει τέτοιου είδους στάδια είναι οι δυναμικές χειρονομίες, που, σε αντίθεση με τις στατικές, αποτελούνται από τρεις χρονικές φάσεις: της προετοιμασίας (preparation), του πυρήνα (nucleus) και της ανάκλησης (retractions). Με βάση αυτή την περιγραφή μπορούμε να θεωρούμε τις χειρονομίες σαν μία κατηγορία απλών δράσεων που αποτελούνται από κινήσεις σημείων ή ολόκληρου του σώματος. Συνήθως μία χειρονομία έχει σύντομη χρονική διάρκεια. Σε επίπεδο πολυπλοκότητας εμφάνισης και χρονικής εξέλιξης, οι χειρονομίες είναι συγκρίσιμες με απλές δράσεις όπως το περπάτημα, το τρέξιμο κ.λ.π όπου αρκεί συνήθως ένα σύντομο στιγμιότυπο για να διασαφηνιστεί η εκτελούμενη δράση.

Η *οπτική αναγνώριση δράσεων (visual action recognition)* είναι η προσπάθεια αυτόματης εξαγωγής μίας συμβολικής, στατιστικής περιγραφής μίας εξελισσόμενης



Σχήμα 1.1: Κατηγοριοποίηση ανθρώπινων δράσεων. Λήφθηκε από [38].

ενέργειας μέσα από τη ροή κάποιας μορφής οπτικής πληροφορίας και λήψης μίας απόφασης σχετικά με την κλάση ή κατηγορία στην οποία ανήκει αυτή η δράση. Για την αναγνώριση ανθρώπινης δράσης ή χειρονομιών μέσα από βίντεο, συνήθως ακολουθείται μία εκ των δύο βασικών προσεγγίσεων για την επίτευξη του παραπάνω στόχου. Οι δύο αυτές προσεγγίσεις περιγράφονται παρακάτω στις υποενότητες 1.2.3 και 1.2.4.

1.2.2 Προκλήσεις

Κατά την υλοποίηση συστημάτων αναγνώρισης ανθρωπίνων δράσεων και χειρονομιών, εμφανίζονται μία σειρά προκλήσεων. Κάποιες από αυτές αποτελούν μέχρι και σήμερα άλυτα προβλήματα. Στη συνέχεια παρουσιάζονται συνοπτικά, οι κυριότερες από αυτές:

- **Μεταβολές της γωνίας λήψης, της κλίμακας και των συνθηκών φωτισμού.** Η γωνία λήψης διαδραματίζει ίσως το σημαντικότερο ρόλο στη μορφή που παίρνει μια δράση και στη δυνατότητα αναγνώρισης τής, και μπορεί να είναι αιτία σφαλμάτων ακόμα κι από τον άνθρωπο. Χαρακτηριστικό παράδειγμα είναι ο διαχωρισμός των δράσεων "περπατώ" και "τρέχω" όταν έχουν ληφθεί *en face* (με την κάμερα μπροστά από τον άνθρωπο). Επιπρόσθετα, είναι σημαντικός ο ρόλος της απόστασης από την κάμερα και η ανάλυση του βίντεο. Διακυμάνσεις σε αυτές τις συνθήκες μπορούν να οδηγήσουν σε σημαντική μεταβολή της εμφάνισης μίας κίνησης και κατ'επέκταση μίας δράσης. Είναι, ακόμη, αρκετά συνηθισμένο να μεταβάλλονται τόσο η γωνία λήψης όσο και η απόσταση της κάμερας κατά τη διάρκεια του βίντεο, π.χ. στην περίπτωση που αυτός που κρατά την κάμερα μετακινείται ή "εστιάζει" σε συγκεκριμένα σημεία. Τέτοιες άσχετες με τη δράση κινήσεις οδηγούν σε μια περίπλοκη συνισταμένη κίνηση, από την οποία είναι δύσκολο να απομονωθεί αυτή που αντιστοιχεί στον άνθρωπο και αυτή που αντιστοιχεί στην δράση στην οποία αυτός επικεντρώνεται. Οι συνθήκες λήψης δεδομένων πολλές φορές περιορίζουν τη δυνατότητα χρήσης μοντέλων που στηρίζονται σε αυτά (μέσω εκπαίδευσης),

υπό μεταβαλλόμενες συνθήκες λειτουργίας. Ένα παράδειγμα, τέτοιου προβλήματος εντοπίστηκε για το on-line σύστημα του 6ου Κεφαλαίου.

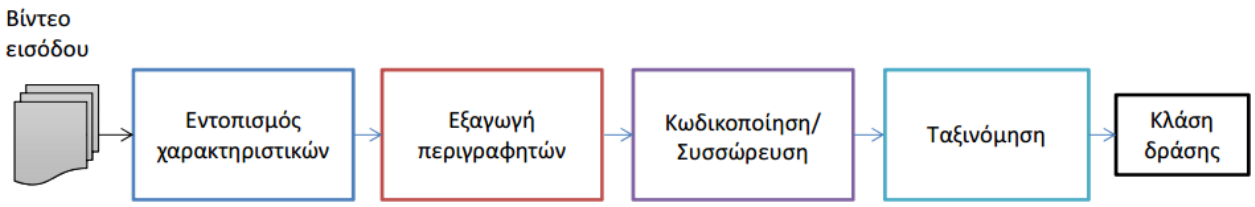
- **Μεταβλητότητα μεταξύ των κλάσεων και εντός της ίδιας κλάσης και μεταξύ των διαφορικών κλάσεων.** Η εκτέλεση της ίδιας δράσης μπορεί να διαφέρει σημαντικά από άνθρωπο σε άνθρωπο ως προς την ταχύτητα εκτέλεσης ή την πόζα του χεριού ή του σώματος. Επίσης, πολλές κατηγορίες δράσεων παρουσιάζουν μεγάλες ομοιότητες, όπως για παράδειγμα οι κλάσεις "running" και "jogging". Αυτό αναφέρεται συνήθως ως inter-class και intra-class variation.
- **Εκτέλεση των αλγορίθμων σε πραγματικό χρόνο ώστε να εξασφαλίζεται η φυσική επικοινωνία χρήστη και μηχανής**. Μεθοδολογίες που στηρίζονται σε χαρακτηριστικά όπως οι πυκνές τροχιές [52] απαιτούν σημαντικό χρόνο για την εκτέλεση των υπολογισμών. Επίσης η λειτουργία συστημάτων που επεξεργάζονται και αναγνωρίζουν χειρονομίες ή δράσεις που εμφανίζονται σε συνεχή ροή οπτικής πληροφορίας απαιτούν την εύρεση μίας εύρωστης και αποτελεσματικής μεθόδου κατάτμησης της συνεχούς ροής σε χρονικά τμήματα (clips ή temporal segments) τα οποία περιέχουν μόνο μία εκτέλεση κάποιας χειρονομίας.
- **Δυσκολία ακριβούς επισημείωσης των χρονικά διαδοχικών φάσεων των δράσεων για βάσεις μεγάλης κλίμακας.** Παρότι ένα από τα βασικά ζητήματα στα πλαίσια της Μηχανικής Μάθησης είναι η δημιουργία μεγάλων συνόλων δεδομένων ώστε η διαδικασία μάθησης να οδηγεί σε εύρωστα και αποδοτικά συστήματα, αυτό φαίνεται τα τελευταία χρόνια, σταδιακά να υπερβαίνεται. Για να μπορέσουν τα συστήματα να οδηγήσουν σε ακόμα καλύτερα αποτελέσματα είναι σημαντικό να εμπλουτισθεί ο τρόπος με τον οποίο γίνεται η επισημείωση των δεδομένων ώστε στο πλαίσιο της μάθησης με επίβλεψη, να επιτυγχάνεται πιο λεπτομερής αναγνώριση των επιμέρους φάσεων μία απλής ή σύνθετης δράσης. Αυτό δυστυχώς έχει υψηλό κόστος σε επίπεδο χρόνου και ανθρώπινης εργασίας και συνεπώς οι επισημειώσεις των συνόλων δεδομένων περιορίζονται, στις περισσότερες, περιπτώσεις στην καταγραφή της συνολικής δράσης.

1.2.3 Κατασκευασμένα ή "Hand-Crafted" χαρακτηριστικά

Μέχρι την επανάσταση που έφεραν τα βαθιά νευρωνικά δίκτυα στην ταξινόμηση εικόνας και βίντεο, κυριαρχούσαν τα *"hand crafted"* χαρακτηριστικά. Η γενική μορφή της διαδικασίας που ακολουθείται για την εξαγωγή τέτοιων χαρακτηριστικών συνοψίζεται στο Σχήμα 1.2.

Αρχικά απαιτείται ένας *ανιχνευτής χωροχρονικών σημείων ενδιαφέροντος (space-time interest points)*. Σκοπός του βήματος αυτού είναι ο προσδιορισμός των χωροχρονικών σημείων ενδιαφέροντος γύρω από τα οποία υπάρχει σημαντική πληρο-

φορία για την υπό εξέλιξη δράση. Οι μεθοδολογίες για αυτό το βήμα χωρίζονται σε *πυκνές (dense)* και *αραιές (sparse)*. Στην πρώτη κατηγορία εμπίπτει η πυκνή δειγματοληψία μέσω της οποίας επιλέγονται τα pixel γύρω από ένα σταθερό χωροχρονικό πλέγμα. Μία προέκταση της απλής πυκνής δειγματοληψίας είναι η παρακολούθηση των *τροχιών* των σημείων ενδιαφέροντος τα οποία έχουν επιλεχθεί αρχικά μέσω πυκνής δειγματοληψίας. Στην δεύτερη κατηγορία εντάσσονται μεθοδολογίες που χρησιμοποιούν σύνθετα κριτήρια οπτικής σημαντικότητας (visual saliency) όπως ο Harris 3D ανιχνευτής[9], ο Hessian ανιχνευτής [22], ο κυβικός ανιχνευτής [10]. Στη συνέχεια, πρέπει, γύρω από τα σημεία που έχουν επιλεχθεί, στο χώρο και στο χρόνο, να υπολογιστούν περιγραφητές που να ενσωματώνουν την τοπική πληροφορία μέσα σε ένα διάνυσμα χαρακτηριστικών. Πολύ διάσημοι και ευρέως χρησιμοποιούμενοι είναι οι περιγραφητές HOG και HOF και οι διάφορες παραλλαγές τους που κωδικοποιούν την πληροφορία σχετικά με την 2D ή 3D κλίση (gradient) και την οπτική ροή, αντίστοιχα, η οποία περιέχεται εντός κάθε χωροχρονικού όγκου γύρω από ένα σημείο ενδιαφέροντος. Συνηθίζεται τα χαρακτηριστικά που έχουν εξαχθεί να *κωδικοποιούνται* (encoding) και να *συσσωρεύονται* (aggregation). Το πρώτο στάδιο παράγει έναν κώδικα για κάθε περιγραφητή με βάση κάποιον κανόνα κωδικοποίησης και στη συνέχεια η συσσώρευση προκύπτει εφαρμόζοντας κάποιον τελεστή συσσώρευσης (pooling operator) για κάθε στοιχείο κάθε χαρακτηριστικού. Τέλος, το προκύπτον διάνυσμα χαρακτηριστικών αποτελεί την είσοδο κάποιου αλγορίθμου ταξινόμησης ο οποίος και δίνει την κλάση της δράσης του υπό επεξεργασία βίντεο.

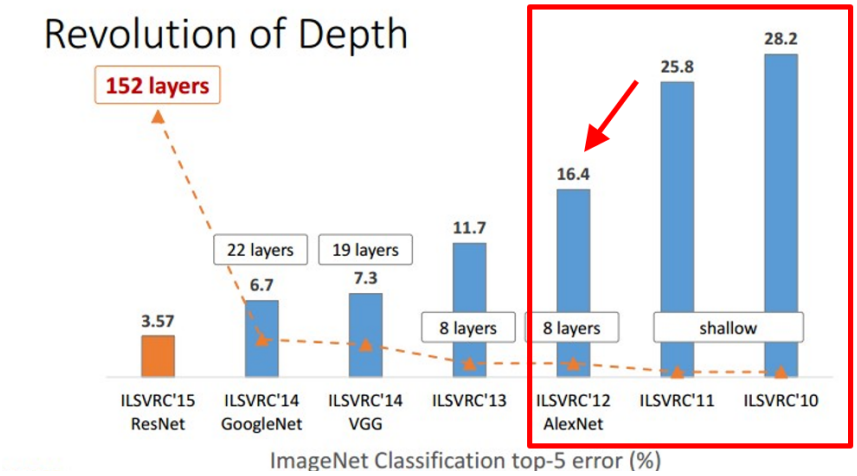


Σχήμα 1.2: Επιμέρους βήματα της εξαγωγής hand crafted χαρακτηριστικών και ταξινόμησης δράσης που πραγματοποιείται σε ένα βίντεο.

1.2.4 Νευρωνικά Δίκτυα

Έχει αποδειχθεί στην πράξη ό,τι τα Τεχνητά Νευρωνικά Δίκτυα, παρά τη σχετικά απλή δομή τους, είναι πολύ αποτελεσματικά στην επίλυση ποικίλων προβλημάτων της Όρασης Υπολογιστών. Γενικά, με τον όρο Τεχνητό Νευρωνικό Δίκτυο αναφερόμαστε σε ένα ευρύ σύνολο μοντέλων και αρχιτεκτονικών (MLPs, CNNs, RNNs) που αποτελούνται από συνδεδεμένα στρώματα υπολογιστικών μονάδων, τα οποία έχουν σαν βασική δομική μονάδα τους *νευρώνες*. Τα στρώματα των υπολογιστικών αυτών μονάδων είναι δυνατόν να τροφοδοτούν τις εξόδους τους είτε μόνο προς τα

στρώματα που έπονται, (οπότε και καλούνται *εμπρόσθιας διάδοσης ή τροφοδότησης*) είτε τόσο σε στρώματα που έπονται όσο και σε στρώματα που προηγούνται, (οπότε και ονομάζονται αναδρομικά *νευρωνικά δίκτυα*). Αξίζει να αναφερθεί πως παρά την πρόσφατη σημαντική αύξηση του ενδιαφέροντος της επιστημονικής κοινότητας σχετικά με τα βαθιά νευρωνικά δίκτυα, το υπολογιστικό πρότυπο των νευρωνικών δικτύων αποτελούσε αντικείμενο έρευνας από τη δεκαετία του 1970. Οι πρώτες προσπάθειες εκπαίδευσης απλών νευρωνικών μοντέλων όπως το Perceptron έγινε από τον Rosenblatt, τον Widrow, τον Hoff και άλλους [65, 15, 16]. Μέχρι και τις αρχές της προηγούμενης δεκαετίας, τα νευρωνικά δίκτυα δεν είχαν πείσει πως μπορούν να αποτελέσουν μία εύχρηστη και εύρωστη λύση για την απόδοση των συστημάτων Όρασης υπολογιστών [19, 24]. Στη συνέχεια έγινε η διάσημη δημοσίευση των A. Krizhevsky, I. Sutskever, και G. E. Hinton [50], όπου με χρήση βαθέων συνελικτικών δικτύων κέρδισαν το διαγωνισμό αναγνώρισης εικόνας μεγάλης κλίμακας του Imagenet. Σταδιακά παρατηρήθηκε πως εκτός από τα εκπληκτικά αποτελέσματα στην αναγνώριση εικόνας και αντικειμένων, τα συνελικτικά δίκτυα μπορούν να επεκταθούν και στο πρόβλημα της αναγνώρισης και ταξινόμησης βίντεο και συνεπώς και ανθρώπινης δράσης.

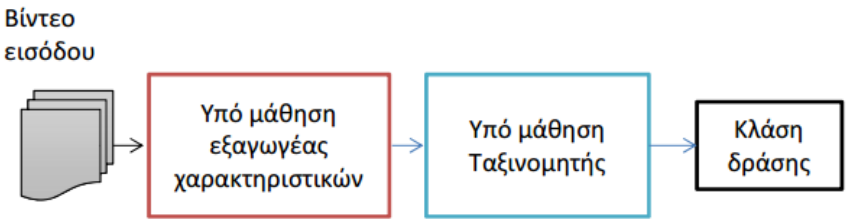


Σχήμα 1.3: Η βελτίωση των επιδόσεων στον διαγωνισμό αναγνώρισης εικόνας Imagenet. Όπως φαίνεται τα βαθιά νευρωνικά μοντέλα κυριαρχούν τα τελευταία χρόνια και η αύξηση του βάθους των μοντέλων οδήγησε σε άλματα τα τελευταία 5 χρόνια.

Στην αναφορά [20], εμφανίζεται ένα από τα πρώτα παραδείγματα συνδυασμού συνελικτικών και αναδρομικών δικτύων με σκοπό την αναγνώριση ανθρώπινης δράσης, που έδωσε ανταγωνιστικά αποτελέσματα. Επίσης, στις εργασίες [17, 3, 23, 56], εφαρμόσθηκαν συνελικτικά δίκτυα τρισδιάστατης συνέλιξης σε προβλήματα αναγνώρισης ανθρώπινων δράσεων.

Όλες οι προσεγγίσεις που χρησιμοποιούν νευρωνικά δίκτυα στηρίζονται στη γενική σχηματική αναπαράσταση που φαίνεται στο Σχήμα 1.4. Συγκεκριμένα, αντί να ακολουθείται η ακολουθία βημάτων που περιγράφεται στο Σχήμα 1.2, υπολογίζονται οι παράμετροι των μοντέλων μέσω end-to-end εκπαίδευσης επί κάποιου

συνόλου δεδομένων. Η βασική διαφορά ανάμεσα στις δύο προσεγγίσεις είναι πως στην περίπτωση των νευρωνικών δικτύων, η διαδικασία μάθησης του δικτύου οδηγεί στην εύρεση ενός βέλτιστου, για το κάθε πρόβλημα, εξαγωγέα χαρακτηριστικών ενώ η κλασσική προσέγγιση απαιτούσε την επιλογή κάποιου κατασκευασμένου ή "Hand-Crafted" εξαγωγέα χαρακτηριστικών. Παράλληλα με τη μάθηση του εξαγωγέα χαρακτηριστικών ένα νευρωνικό δίκτυο έχει τη δυνατότητα να μάθει και τις παραμέτρους κάποιο ταξινομητή, που στην πράξη είναι απλά το τελικό στρώμα νευρώνων ενός νευρωνικού δικτύου, με κατάλληλη έξοδο ώστε να δίνει την posterior πιθανότητα κάθε κλάσης.



Σχήμα 1.4: Υπό μάθηση εξαγωγέας χαρακτηριστικών και ταξινομητής βίντεο ανθρώπινης δράσης.

1.3 Στόχοι της διπλωματικής εργασίας

Ο βασικός σκοπός της παρούσας Διπλωματικής είναι ο πειραματισμός με μοντέλα νευρωνικών δικτύων που εξάγουν και συνδυάζουν χωρικά και χρονικά χαρακτηριστικά για την εφαρμογή και αξιολόγησή τους σε προβλήματα αναγνώρισης δράσης και χειρονομιών. Για τη σχεδίαση των μοντέλων επιλέχθηκε να χρησιμοποιηθούν δύο είδη νευρωνικών δικτύων, τα Συνελικτικά και τα Αναδρομικά Τεχνητά Νευρωνικά Δίκτυα. Η εκπαίδευση νευρωνικών δικτύων όπως και οποιουδήποτε μοντέλου Όρασης Υπολογιστών και Μηχανικής Μάθησης είναι άρρηκτα συνδεδεμένη με τη χρήση συνόλων ή βάσεων δεδομένων ώστε να είναι εφικτή η στατιστική εκτίμηση των παραμέτρων τους. Η επιλογή των βάσεων δεδομένων που χρησιμοποιήθηκαν και περιγράφονται στην ενότητα 1.5, στηρίχθηκε αρχικά στην αναγκαιότητα να μην είναι μεγάλης κλίμακας, ώστε, λαμβάνοντας υπόψη τους περιορισμούς υλικού, η εκπαίδευση των μοντέλων να ολοκληρώνεται εντός λογικών χρονικών πλαισίων. Στόχος της παρούσας Διπλωματικής επίσης είναι να εξετασθεί αν χρησιμοποιώντας σχετικά λίγα δεδομένα, της τάξης των $1000 - 1500$ παραδειγμάτων, είναι εφικτό να εκπαιδευτούν μοντέλα σημαντικού βάθους της τάξης των $5 - 7$ στρωμάτων (Συνελικτικών ή Πλήρως συνδεδεμένων στρωμάτων ή στρωμάτων LSTM νευρώνων) των οποίων η λειτουργία στηρίζεται στην εύρεση μέσω μάθησης, $10 - 20$ εκατομμυρίων παραμέτρων. Επιπρόσθετα, στην παραπάνω διερεύνηση επιχειρήθηκε να επιστρατευτούν ένα σύνολο μεθοδολογιών εκπαίδευσης και τεχνικών κανονικοποίησης ώστε να βελτιωθεί η ικανότητα γενίκευσης των μοντέλων και να επιταχυνθεί η

σύγκλιση της διαδικασίας εκπαίδευσης. Τέλος, ένα εξίσου σημαντικό μέρος της παρούσας Διπλωματικής είναι η υλοποίηση και η αξιολόγηση ενός συστήματος on-line αναγνώρισης χειρονομιών που χρησιμοποιεί ένα από τα μοντέλα που εκπαιδεύτηκε με δεδομένα της βάσης SKIG.

1.4 Σύνοψη Συνεισφοράς

Οι συνεισφορές της διπλωματικής εργασίας συνοψίζονται στα εξής σημεία:

Στο **5ο Κεφάλαιο** παρουσιάζονται τα αποτελέσματα πειραμάτων off-line ταξινόμησης τριών τύπων μοντέλων : Νευρωνικά δίκτυα Τρισδιάστατης συνέλιξης (3DCNN,C3D), Νευρωνικο δίκτυο Τρισδιάστατης συνέλιξης συνδεδεμένο με στρώμα LSTM νευρώνων (3DCNN-LSTM) και Νευρωνικό δίκτυο Δισδιάστατης συνέλιξης συνδεδεμένο με στρώμα LSTM νευρώνων (CNN-LSTM). Βασικός στόχος του κεφαλαίου είναι η αξιολόγηση της επίδοσης των παραπάνω μοντέλων επί βάσεων δεδομένων μικρής κλίμακας (SKIG, KTH). Επίσης, γίνεται σύγκριση της αποτελεσματικότητας της χρήσης χωροχρονικών ή μόνο χωρικών χαρακτηριστικών συνελικτικών χαρακτηριστικών από βίντεο και της βελτίωσης που επιφέρει η ενσωμάτωση καθολικής (και όχι μόνο τοπικής) χρονικής μοντελοποίησης που επιτυγχάνεται με την εισαγωγή ενός στρώματος LSTM νευρώνων στα μοντέλα. Τέλος, γίνεται προσπάθεια να χρησιμοποιηθούν και να αξιολογηθούν διάφορες τεχνικές και μεθοδολογίες που συναντώνται στη σύγχρονη βιβλιογραφία, σχετικά με την εκπαίδευση βαθιών νευρωνικών δικτύων και αποσκοπούν στη βελτίωσης της απόδοσης τους. Ιδιαίτερη αναφορά γίνεται στην επίδραση της επαύξησης δεδομένων η οποία φάνηκε να οδηγεί σε σημαντικές βελτιώσεις της ακρίβειας των μοντέλων.

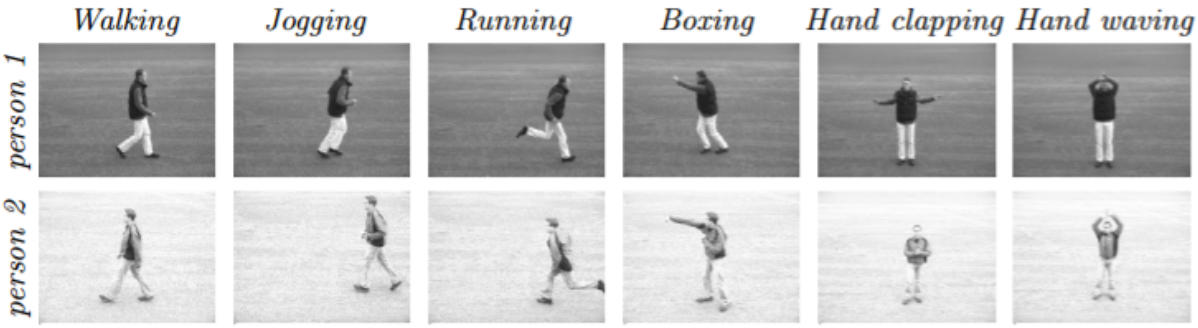
Στο **6ο Κεφάλαιο** επιδιώκεται να χρησιμοποιηθούν κάποια από τα μοντέλα που εκπαιδεύτηκαν στο 5ο Κεφάλαιο για να σχεδιαστεί ένα σύστημα αναγνώρισης χειρονομιών σε πραγματικό χρόνο (on-line). Συγκεκριμένα παρουσιάζονται οι σχεδιαστικές επιλογές που έγιναν για να μπορέσει το τελικό σύστημα να αναγνωρίζει-/προβλέπει σε πραγματικό χρόνο και εξηγείται η επιμέρους λειτουργία των δομικών στοιχείων που το αποτελούν. Τέλος παρουσιάζεται μία σύντομη αξιολόγηση των επιδόσεων του συστήματος και περιγράφονται τα πλεονεκτήματα και οι περιορισμοί του.

1.5 Βάσεις Δεδομένων

Η εξέλιξη και κατανόηση των νευρωνικών δικτύων είναι άρρηκτα συνδεδεμένη με τη διεύρυνση των διαθέσιμων βάσεων δεδομένων. Για τη διεξαγωγή των πειραμάτων του 5ου Κεφαλαίου επιλέχθηκαν οι βάσεις KTH και SKIG που παρουσιάζονται παρακάτω.

1.5.1 Η βάση ανθρώπινων δράσεων KTH

Η KTH [7] αποτελεί μια από τις πρώτες βάσεις δεδομένων ανθρώπινων δράσεων, και παραμένει μέχρι σήμερα αρκετά δημοφιλής. Τα βίντεο που περιλαμβάνει έχουν μαγνητοσκοπηθεί χειροκίνητα και σε ελεγχόμενο περιβάλλον, οπότε η εκτέλεση των δράσεων είναι "καθαρή", χωρίς επικαλύψεις ή ιδιαίτερες αποκλίσεις μεταξύ των διαφορετικών εκτελέσεων. Για τον λόγο αυτό καθιστά ευκολότερη την ανάλυση και την εις βάθος επισκόπηση των διαφόρων μεθόδων. Συνολικά περιλαμβάνει 2391 βίντεο χρωματικής κλίμακας Grayscale, καθένα από τα οποία ανήκει σε μια από τις εξής κατηγορίες: *walking, jogging, running, boxing, hand waving* και *hand clapping*. Ενδεικτικά καρέ από κάθε κατηγορία φαίνονται στο Σχήμα 1.5. Κάθε δράση εκτελείται περίπου 4 φορές από 25 διαφορετικά άτομα και υπό 4 διαφορετικές συνθήκες: σε εξωτερικό χώρο, σε εξωτερικό χώρο με μεταβλητή κλίμακα (ζουμ κατά τη διάρκεια του βίντεο), σε εξωτερικό χώρο με διαφορετικά ρούχα και σε εσωτερικό χώρο. Η λήψη όλων των βίντεο έγινε με στατική κάμερα και με ρυθμό 25 καρέ ανά δευτερόλεπτο (fps). Στη συνέχεια, τα βίντεο υποδειγματοληπτήθηκαν ώστε να έχουν σταθερή ανάλυση 160×120. Τα δεδομένα χωρίζονται σε σύνολο εκπαίδευσης και σύνολο αξιολόγησης, όπως προτάθηκε από τους δημιουργούς της βάσης. Το δεύτερο σύνολο περιλαμβάνει τα βίντεο των δειγμάτων (ατόμων) $2, 3, 5, 6, 7, 8, 9, 10$ και 22, ενώ το πρώτο αποτελείται από τα βίντεο όλων των υπολοίπων δειγμάτων.

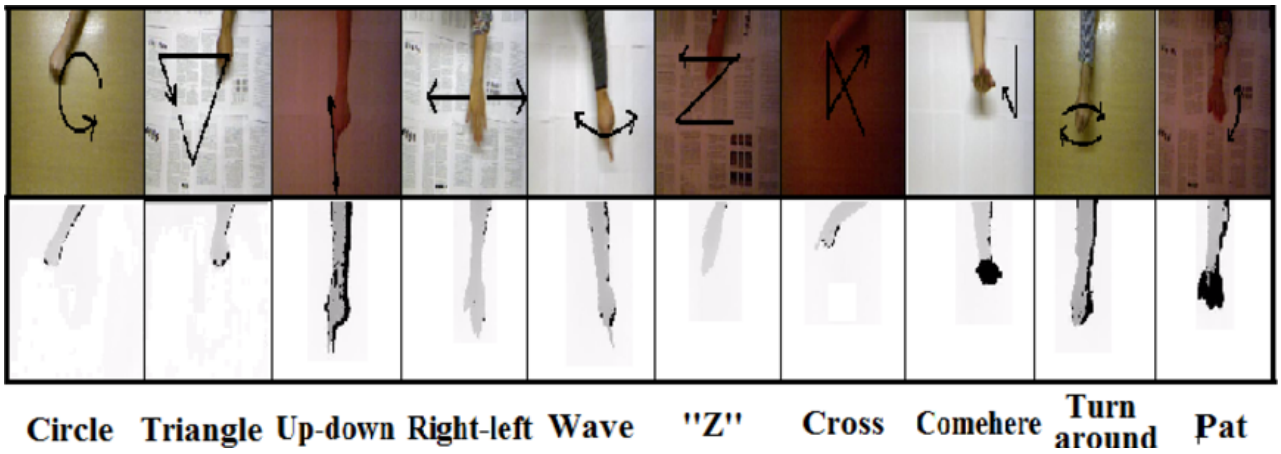


Σχήμα 1.5: Η κλάσεις ανθρωπίνων δράσεων της βάσης KTH. Λήφθηκε από [8]

1.5.2 Η βάση ανθρώπινων δυναμικών χειρονομιών SKIG (Sheffield KInect Gesture dataest)

Η SKIG [11] αποτελείται από 1080 έγχρωμα βίντεο (RGB) και 1080 βίντεο βάθους (Depth) που λήφθηκαν συγχρονισμένα από κάμερα Kinect τοποθετημένη σε σταθερή θέση πάνω από κάποια επιφάνεια μεταβλητού χρώματος και υλικού και υπό μεταβλητές συνθήκες φωτισμού. Το πλήθος των διαφορετικών επιφανειών και συνθηκών φωτισμού είναι 3 (ξύλινο, λευκό χαρτί, χαρτί με κείμενο) και 2 (έντονος και ασθενής φωτισμός), αντίστοιχα. Τα βίντεο που περιέχονται στη βάση, περι-

λαμβάνουν 10 διαφορετικές κλάσεις χειρονομιών που εκτελούνται από 6 διαφορετικά άτομα. Όπως φαίνεται στο Σχήμα 1.6 τα ονόματα των κλάσεων είναι *Circle, Triangle, Up-Down, Right-Left, Wave, "Z", Cross, Come here, Turn Around, Pat*. Για κάθε κλάση χειρονομίας χρησιμοποιούνται τρία είδη πόζας του χεριού (προτεταμένος δείκτης, επίπεδη παλάμη, γροθιά). Στο πειραματικό μέρος της παρούσας εργασίας χρησιμοποιήθηκαν τα βίντεο 4 ατόμων για την εκπαίδευση και 2 ατόμων για την αξιολόγηση των μοντέλων. Η συγκεκριμένη βάση επιλέχθηκε διότι και σε αυτή την περίπτωση δεν υπάρχουν επικαλύψεις και οι συνθήκες λήψης των δεδομένων είναι εύκολο να προσομοιωθούν ώστε να ενσωματωθούν τα νευρωνικά μοντέλα που εκπαιδεύτηκαν σε ένα σύστημα on-line αναγνώρισης χειρονομιών. Επιπλέον η συγκεκριμένη βάση δίνει τη δυνατότητα να συγκριθούν δύο διαφορετικές τροπικότητες ροής βίντεο. Τέλος, οι κλάσεις χειρονομιών της συγκεκριμένης βάσης δεδομένων δεν διαφέρουν έντονα μεταξύ τους και έτσι δίνουν τη δυνατότητα να εξετασθεί η ικανότητα των μοντέλων να διακρίνουν λεπτές διαφορές (fine grained discrimination) ανάμεσα στη χωρική και χρονική εξέλιξη κάποιων χειρονομιών.



Σχήμα 1.6: Οι κλάσεις χειρονομιών της βάσης SKIG

Κεφάλαιο 2

Νευρωνικά Δίκτυα Εμπρόσθιας Τροφοδότησης

Σε αυτό το κεφάλαιο παρουσιάζονται δύο βασικά μοντέλα νευρωνικών δικτύων, τα *Πολυστρωματικά Perceptron (Multilayer Percpetrons)* και τα *Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks)*. Τα μοντέλα αυτά αποτελούν βασικά δομικά στοιχεία κάθε σύγχρονου νευρωνικού δικτύου που επεξεργάζεται, εξάγει χαρακτηριστικά και αναγνωρίζει εικόνα και βίντεο. Επίσης, τα δύο αυτά είδη μοντέλων εντάσσονται στην κατηγορία των νευρωνικών μοντέλων εμπρόσθιας τροφοδότησης, των οποίων ο ορισμός ακολουθεί.

2.1 Δίκτυα Εμπρόσθιας Τροφοδότησης (Feedforward Networks)

Σε ένα *δίκτυο εμπρόσθιας τροφοδότησης*, η πληροφορία ρέει μόνο από το προηγούμενο προς το επόμενο στρώμα (δηλαδή δεν υπάρχει ανάδραση όπως στα Αναδρομικά Νευρωνικά Δίκτυα). Ο σκοπός ενός τέτοιου δικτύου είναι η προσέγγιση μίας συνάρτησης f^*. Στις περισσότερες περιπτώσεις η συνάρτηση αυτή περιγράφει τη λειτουργία ενός ταξινομητή και συνεπώς η γενική μορφή της συνάρτησης είναι $y = f^*(\mathbf{x})$, δηλαδή αντιστοιχίζει μία είσοδο $\mathbf{x}$ σε μία κατηγορία y. Η δράση ενός feedforward Νευρωνικού μοντέλου συνίσταται στην υλοποίηση της συνάρτησης $f(\mathbf{x}) = f(\mathbf{x}; \theta)$ οποία καθορίζεται από ένα σύνολο παραμέτρων (βαρών) θ, οι οποίες επιδιώκεται να υπολογιστούν μέσω της διαδικασίας μάθησης, ώστε να πετύχουν την καλύτερη δυνατή προσέγγιση της συνάρτησης f^*. Η τελική μορφή της υλοποιούμενης συνάρτησης f προκύπτει από τη σειριακή σύνθεση ενός αριθμού συναρτήσεων που υλοποιεί το κάθε στρώμα του δικτύου. Συμβολικά έχουμε:

$$f(\mathbf{x}) = f^{(D)}(f^{(D-1)}(f^{(D-2)}(...f^{(1)}(\mathbf{x})...))) \tag{2.1}$$

όπου D το βάθος του δικτύου. Οι συναρτήσεις $f^{(i)}$ παράγουν ενδιάμεσες αναπαραστάσεις της εισόδου, και ο σκοπός της μάθησης (εκπαίδευσης) είναι να εξάγει το ποιες θα είναι αυτές οι αναπαραστάσεις και πώς θα συνδυαστούν μεταξύ τούς.

2.1.1 Πολυστρωματικά Perceptrons

Ο Τεχνητός Νευρώνας Perceptron

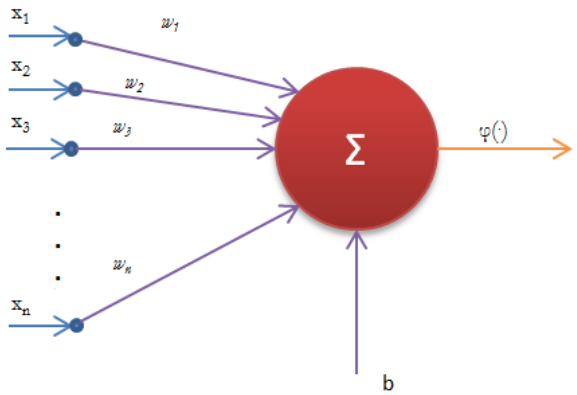
Η πιο απλή και θεμελιώδης μορφή ενός νευρικού δικτύου είναι τα *Πολυστρωματικά Perceptrons* (Multilayer Perceptrons ή MLPs) των οποίων το βασικό δομικό στοιχείο είναι ο νευρώνας Perceptron. Το μοντέλο του Perceptron, περιγράφηκε από τον Rosenblatt [65]. Το μοντέλο που περιέγραψε ο Rosenblatt αποτελείται από έναν νευρώνα, ο οποίος βασίζεται στο μοντέλο νευρώνα των McCulloh-Pitts [66], δέχεται σαν είσοδο ένα διάνυσμα, $\mathbf{x} \in \mathbb{R}^n$ και παράγει μία έξοδο $y \in \mathbb{R}$. Μία σχηματική αναπαράσταση του Perceptron παρουσιάζεται στο Σχήμα 2.1. Για να προκύψει η έξοδος του νευρώνα πραγματοποιούνται τα εξής υπολογιστικά βήματα:

- Κάθε στοιχείο της εισόδου x_i πολλαπλασιάζεται με ένα *βάρος σύνδεσης* ή *βάρος σύναψης* w_i και τα αποτελέσματα αθροίζονται. Η διαδικασία αυτή είναι ισοδύναμη με τον υπολογισμό του εσωτερικού γινομένου $\mathbf{x}^T\mathbf{w}$. Στο αποτέλεσμα προστίθεται ένας όρος πόλωσης $b \in \mathbb{R}$.

$$v(\boldsymbol{x}) = \sum_{i=1}^{n} w_i x_i + b \tag{2.2}$$

- Η τιμή $v(\mathbf{x})$ τροφοδοτείται σε μία *μη γραμμική συνάρτηση ενεργοποίησης* $\phi(\cdot)$ και έτσι προκύπτει η έξοδος του Perceptron, δηλαδή έχουμε:

$$y = f(\boldsymbol{x}) = \phi(\sum_{i=1}^{n} w_i x_i + b) \tag{2.3}$$



Σχήμα 2.1: Μοντέλο Perceptron

Συναρτήσεις Ενεργοποίησης

Σαν συνάρτηση ενεργοποίησης συνηθέστερα επιλέγεται μία από τις παρακάτω διαφορετικές συναρτήσεις:

- Η συνάρτηση ReLU (Rectifier Linear Unit) η οποία ορίζεται ως:

$$\phi(x) = \max\{0, x\} \tag{2.4}$$

- Η σιγμοειδής συνάρτηση (sigmoid), η οποία περιγράφεται από την παρακάτω σχέση:

$$\phi(x) = \frac{1}{1 + \exp(-x/T)} \tag{2.5}$$

- Η συνάρτηση υπερβολικής εφαπτόμενης, η οποία χρησιμοποιεί και τις παραμέτρους k και g και περιγράφεται από την παρακάτω σχέση:

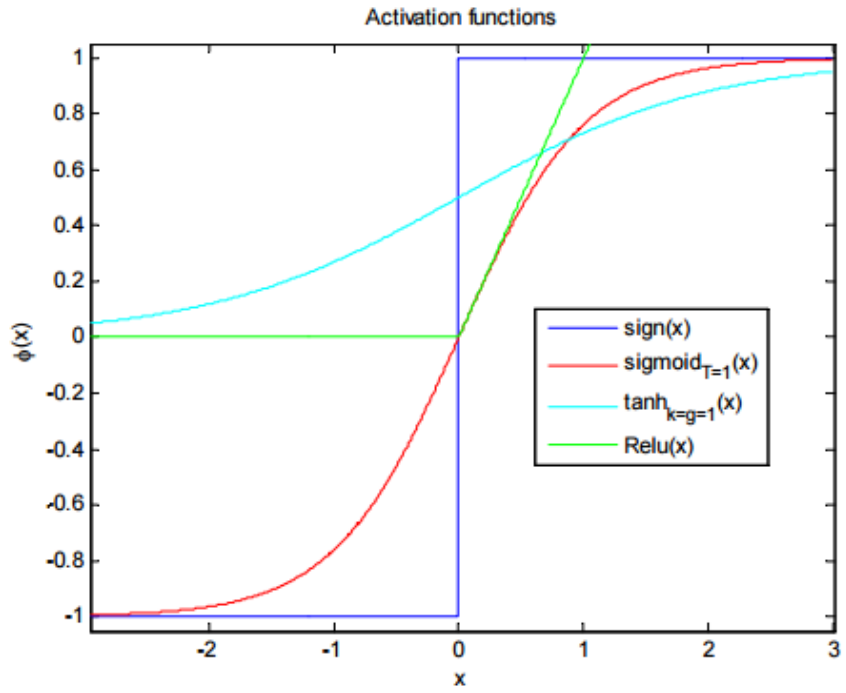
$$\phi(x) = k \tanh(gx) \tag{2.6}$$

- Η συνάρτηση προσήμου

$$\phi(x) = \text{sign}(x) = \begin{cases} 1 & \text{αν } x > 0 \\ 0 & \text{αν } x = 0 \\ -1 & \text{αν } x < 0 \end{cases} \tag{2.7}$$

Η χρήση των μη γραμμικών συναρτήσεων ενεργοποίησης διαφοροποιεί τα νευρωνικά μοντέλα από τα γραμμικά μοντέλα. Το Σχήμα 2.2 παρουσιάζει τις πιο συχνά χρησιμοποιούμενες συναρτήσεις ενεργοποίησης. Στην πράξη η συχνότερα εμφανιζόμενη συνάρτηση ενεργοποίησης είναι η ReLU που χρησιμοποιείται τόσο σε απλά MLPs όσο και στα Συνελικτικά Νευρωνικά Δίκτυα που περιγράφονται στη συνέχεια του κεφαλαίου. Η χρήση της ReLU έχει επικρατήσει διότι έχει χαρακτηριστικά που διευκολύνουν τη μάθηση μέσω μεθόδων gradient descent. Συγκεκριμένα, το γεγονός ότι στο ήμισυ του πεδίου ορισμού της, η τιμή της ReLU είναι μηδέν, διευκολύνει την ύπαρξη μεγαλύτερων τιμών κλίσης (gradient) της εξόδου ενός νευρώνα ως προς τα βάρη των συνάψεών του. Αντίθετα, οι σιγμοειδείς συναρτήσεις ενεργοποίησης παρουσιάζουν κορεσμό είτε σε μία θετική είτε σε μία αρνητική τιμή για το μεγαλύτερο μέρος του πεδίου ορισμού τους, γεγονός που οδηγεί σε πολύ μικρές τιμές κλίσης, καθιστώντας τη μάθηση αργή ή ακόμα και αδύνατη. Η συνάρτηση υπερβολικής εφαπτομένης πολλές φορές εμφανίζεται σαν συνάρτηση ενεργοποίησης σε νευρώνες αναδρομικών νευρωνικών μοντέλων όπως Long-Term Short-Term νευρώνες που εξετάζονται στο Κεφάλαιο 3.

Όπως ειπώθηκε και νωρίτερα στην παρούσα ενότητα, συνδυάζοντας νευρώνες Perceptron δομείται ένα *Πολυστρωματικό Perceptron* ή *MLP* (Multilayer Perceptron). Στο Σχήμα 2.1 φαίνεται, ένα MLP που αποτελείται από *στρώματα* (Layers) νευρώνων Perceptron, εκ των οποίων κάθε επόμενο είναι *πλήρως συνδεδεμένο* με το προηγούμενο (Fully Connected Layer). Οι νευρώνες ενός στρώματος δεν συνδέονται μεταξύ τους και συνεπώς η ενεργοποίηση ενός νευρώνα εξαρτάται μόνο από τις ενεργοποιήσεις των νευρώνων του προηγούμενου στρώματος και τις συνάψεις (ή



Σχήμα 2.2: Συναρτήσεις Ενεργοποίησης

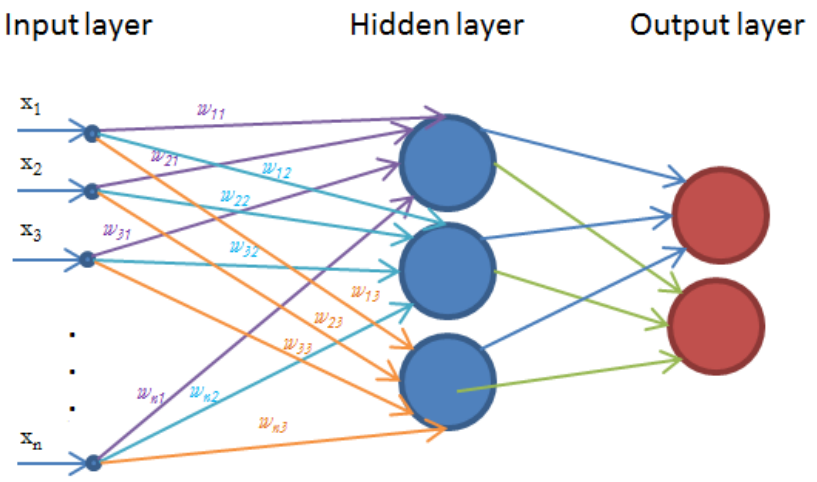
τα βάρη) ανάμεσα σ' αυτόν και τους προηγούμενους νευρώνες. Είναι φανερό, πως οι επιρροή της υπάρχουσας γνώσης σχετικά με τους βιολογικούς νευρώνες, στον τρόπο λειτουργίας των Τεχνητών Νευρωνικών Δικτύων, είναι έντονη. Τόσο η μη γραμμική συνάρτηση ενεργοποίησης των νευρώνων όσο και η πολυστρωματική αρχιτεκτονική ενός MLP που παράγει ενδιάμεσες αναπαραστάσεις της πληροφορίας αποτελούν αρχές στις οποίες έχει φανεί ότι στηρίζεται η λειτουργία συγκεκριμένων περιοχών του εγκεφάλου, όπως ο οπτικός φλοιός των θηλαστικών [67]. Τα στρώματα νευρώνων ενός MLP ονομάζονται *Κρυφά Στρώματα (Hidden Layers)* με εξαίρεση το τελευταίο στρώμα που ονομάζεται στρώμα εξόδου. Σε κάθε Κρυφό Στρώμα νευρώνων μπορούμε να εκφράσουμε με συμπαγή τρόπο το σύνολο των εξισώσεων που περιγράφουν τη γραμμική έξοδο των νευρώνων με την παρακάτω σχέση:

$$\boldsymbol{\upsilon}_l = W\mathbf{y}_{l-1} + \mathbf{b} \tag{2.8}$$

όπου W είναι ο πίνακας των βαρών που συνδέουν τους νευρώνες του στρώματος l με τις εξόδους (αφού έχει εφαρμοστεί η μη γραμμική συνάρτηση ενεργοποίησης) του στρώματος $l-1$, που προηγείται. Συγκεκριμένα:

$$W = \begin{bmatrix} \mathbf{w}_1^T \\ \mathbf{w}_2^T \\ \vdots \\ \mathbf{w}_M^T \end{bmatrix}$$

όπου $\mathbf{w}_j = [w_{1j}, \ldots, w_{Nj}]$ είναι το διάνυσμα βαρών του νευρώνα j του στρώματοςl.



Σχήμα 2.3: Παράδειγμα ενός Πολυστρωματικού Perceptron δύο στρωμάτων (το στρώμα εισόδου δεν προσμετράται) ένα κρυφό (Hidden Layer) και ένα στρώμα εξόδου (Output Layer). Τα βάρη του κάθε νευρώνα των εισόδων κάθε νευρώνα στο πρώτο στρώμα έχουν κωδικοποιηθεί χρωματικά και συμοβλίζονται ως $w_{i,j}$ που αντιστοιχεί στο βάρος της i-οστής εισόδου στον j-οστό νευρώνα.

Η διάσταση του πίνακα W είναι $M \times N$ όπου M το πλήθος των νευρώνων του στρώματος l και N το πλήθος των νευρώνων του στρώματος $l-1$.

Ένα στρώμα εξόδου ενός MLP, προκειμένου να χρησιμοποιηθεί για την επίλυση προβλημάτων ταξινόμησης, απαιτείται να δίνει στην έξοδο του την κατανομή πιθανότητας στις n διακριτές κλάσεις στις οποίες είναι πιθανόν να ανήκει το παράδειγμα που ταξινομείται. Σε ένα στρώμα εξόδου το πλήθος των νευρώνων είναι ίσο με το πλήθος των κλάσεων του προβλήματος ταξινόμησης. Επειδή οι γραμμικές έξοδοι (δηλαδή πριν εφαρμοστεί η συνάρτηση ενεργοποίησης) ενός στρώματος νευρώνων θεωρητικά λαμβάνουν οποιεσδήποτε συνεχείς τιμές, απαιτείται αυτές να απεικονιστούν στο διάστημα $[0, 1]$ ώστε να είναι σε πιθανοτική μορφή. Ακόμα πρέπει να έχουν άθροισμα 1. Συγκεκριμένα, η έξοδος του δικτύου πρέπει να είναι ένα διάνυσμα $\hat{\mathbf{y}}$, n στοιχείων όπου κάθε στοιχείο είναι η πιθανότητα $\hat{y}_i = p(y = i|\mathbf{x})$ και ισχύει $\sum_{i=1}^{n} p(y = i|\mathbf{x}) = 1$. Για να επιτευχθεί αυτό χρησιμοποιείται, στις περισσότερες περιπτώσεις, η συνάρτηση Softfmax η οποία περιγράφεται παρακάτω. Η γραμμική έξοδος ενός στρώματος εξόδου δίνεται από τη σχέση 2.8. Λόγω των ευνοϊκών ιδιοτήτων του λογαρίθμου στα προβλήματα βελτιστοποίησης μέσω μεθόδων της οικογένειας καθόδου κλίσεων και μεγιστοποίησης της πιθανοφάνειας, χρησιμοποιείται ο λογάριθμος της γραμμικής εξόδου του στρώματος $z_j = \log(p(y = j|\mathbf{x}))$. Με βάση τα παραπάνω, η συνάρτηση Softmax δίνεται από τη σχέση:

$$\text{Softmax}(z_j) = \frac{\exp(z_j)}{\sum_{j=1}^{n} \exp(z_j)} \tag{2.9}$$

Εφόσον χρησιμοποιείται η συνάρτηση κόστους αρνητικής λογαριθμικής αλληλοεντροπίας ή πιθανοφάνειας (categorical crossentropy) κατά την εκπαίδευση, χρησιμοποιείται η λογαριθμημένη εκδοχή της συνάρτησης Softmax στο στάδιο εξόδου

του MLP. Συγκεκριμένα:

$$\ln\{\text{Softmax}(z_j)\} = \ z_j - \ln \sum_{j=1}^{n} \exp(z_j) \tag{2.10}$$

Σε επόμενες ενότητες θα αναλυθεί πιο διεξοδικά η σημασία και οι ευνοϊκές ιδιότητες της συνάρτησης Softmax για την εκπαίδευση μέσω του αλγορίθμου Stochastic Gradient Descent εφαρμόζοντας το κριτήριο Μέγιστης Πιθανοφάνειας.

Αριθμός Παραμέτρων ενός MLP

Έστω ένα διάνυσμα εισόδου $\mathbf{x} \in \mathbb{R}^n$, τότε το πρώτο στρώμα νευρώνων, με πλήθος νευρώνων ίσο με N, θα έχει $n \times N_1$ ελεύθερες παραμέτρους (δηλαδή βάρη συνάψεων). Κάθε μετέπειτα στρώμα νευρώνων θα δέχεται είσοδο διάστασης ίση με τον αριθμό των νευρώνων του στρώματος που ακολουθεί. Συνεπώς, συνεχίζοντας, στο δεύτερο στρώμα νευρώνων με πλήθος νευρώνων N_2 θα έχουμε $N_1 \times N_2$ ελεύθερες παραμέτρους. Άρα αν D ο αριθμός στρωμάτων, N_i τα αυθαίρετα πλήθη νευρώνων σε κάθε ένα από τα στρώματα, θεωρώντας P το σύνολο των παραμέτρων του μοντέλου και $N_D = k$ τη διάσταση της εξόδου του MLP, θα ισχύει:

$$card(P) = n \times N_1 + N_1 \times N_2 + \cdots + N_{D-1} \times k \tag{2.11}$$

Εφαρμογή των MLP στην αναγνώριση εικόνας και βίντεο

Ολοκληρώνοντας τη συνοπτική παρουσίαση του θεμελιώδους νευρωνικού μοντέλου MLP, πρέπει να σημειώσουμε πως πολλές φορές αυτό αποτελεί τμήμα πιο σύνθετων αρχιτεκτονικών νευρωνικών δικτύων. Πιο συγκεκριμένα, τα τελικά στάδια Συνελικτικών Νευρωνικών Δικτύων συνηθίζεται να είναι ένα ή περισσότερα στρώματα πλήρως συνδεδεμένων νευρώνων που καταλήγουν σε Softmax έξοδο ώστε να λύνουν κάποιο πρόβλημα ταξινόμησης και να εκπαιδεύονται με εκτίμηση παραμέτρων μέγιστης πιθανοφάνειας (maximum loglikelihood method) και gradient descent βελτιστοποίηση της συνάρτησης κόστους. Σε αρκετές περιπτώσεις, οι έξοδοι των πλήρως συνδεδεμένων στρωμάτων ενός Συνελικτικού Δικτύου -πριν την εφαρμογή της συνάρτησης softmax- είναι ένα διάνυσμα χαρακτηριστικών το οποίο χρησιμοποιείται από κάποιο κλασσικό ταξινομητή όπως οι SVM. Προφανώς αν ακολουθηθεί η μεθοδολογία που παρουσιάστηκε στην Ενότητα 1.2.3 τότε είναι δυνατόν τα MLP να αποτελέσουν τον ταξινομητή που χρησιμοποιείται στο τελικό στάδιο της διαδικασίας όπως φαίνεται και στο Σχήμα 1.2. Αν ακολουθηθεί η μεθοδολογία του Σχήματος 1.4, είναι δυνατό ένα MLP να αποτελέσει τον υπό μάθηση εξαγωγέα χαρακτηριστικών και ταξινομητή ταυτόχρονα. Επειδή τα MLP δέχονται είσοδο σε μορφή διανύσματος είναι δυνατόν να δεχθούν σαν είσοδο εικόνα ή βίντεο μόνο αν η πληροφορία μετασχηματιστεί από τη μορφή δισδιάστατου και τρισδιάστατου

πλέγματος (grid) σε διάνυσμα [19]. Όμως η μεθοδολογία αυτή φάνηκε πως δεν είναι η αποτελεσματικότερη για την εξαγωγή χαρακτηριστικών από εικόνες και βίντεο όταν οι χωρικές διαστάσεις τους αρχίζουν να μεγαλώνουν, κυρίως λόγω του γεγονότος ότι το μοντέλο είναι πλήρως συνδεδεμένο με κάθε στοιχείο της εισόδου. Με βάση την εξίσωση 2.11, αν έχουμε μία RGB εικόνα ανάλυσης $200 \times 200 \times 3$, τότε η είσοδος του MLP θα είναι $\mathbf{I}$ (που θα μπορούσε να είναι και ένα καρέ ενός βίντεο) $\in R^{120.000}$, δηλαδή απαιτούνται 120.000 παράμετροι ανά νευρώνα του στρώματος εισόδου! Είναι αρκετά δύσκολο να αποφευχθεί το φαινόμενο του overfitting για ένα τόσο μεγάλο μοντέλο. Παρόλα αυτά, για μικρότερες εικόνες όπως των βάσεων δεδομένων CIFAR και MNIST με διάσταση $32 \times 32 \times 3$ τα MLPs έχουν δώσει ανταγωνιστικά αποτελέσματα. Τα Συνελικτικά Νευρωνικά Δίκτυα που παρουσιάζονται στη συνέχεια, έχουν κάποιες ιδιότητες που τα καθιστούν καταλληλότερα για την εξαγωγή χαρακτηριστικών σε προβλήματα Όρασης Υπολογιστών.

2.1.2 Συνελικτικά Νευρωνικά Δίκτυα

Τα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks ή CNN), περιγράφηκαν για πρώτη από τον LeCun [19]. Είναι μία ειδική κατηγορία νευρωνικών δικτύων τα οποία είναι κατάλληλα για την επεξεργασία πληροφορίας που μπορεί να αναπαρασταθεί σε μορφή πλέγματος. Στην περίπτωση των εικόνων, το πλέγμα είναι δισδιάστατο και κάθε στοιχείο του είναι η ένταση ενός pixel. Γενικεύοντας την ιδέα αυτή στην περίπτωση της ακολουθίας εικόνων (ή καρέ,frame), δηλαδή ενός βίντεο το πλέγμα είναι τρισδιάστατο και κάθε στοιχείο του είναι η ένταση ενός pixel σε μία συγκεκριμένη χρονική στιγμή. Παρότι τα Συνελικτικά Νευρωνικά Δίκτυα έχουν εφαρμοστεί με επιτυχία -όπως η αναγνώριση ομιλίας- σε προβλήματα όπου η είσοδος αναπαρίσταται από ένα μονοδιάστατο πλέγμα, στην παρούσα ενότητα θα περιοριστούμε στην περιγραφή τους στο πλαίσιο προβλημάτων αναγνώρισης εικόνας και βίντεο. Όπως δηλώνει και η ονομασία τους τα Συνελικτικά Νευρωνικά Δίκτυα, βασίζονται στη μαθηματική διαδικασία της *συνέλιξης* (convolution).

Η γενική δομή ενός Συνελικτικού Δικτύου είναι παρόμοια με αυτή ενός MLP, υπό την έννοια ότι αποτελείται από σειριακά συνδεδεμένα επίπεδα και συνεπώς η εξίσωση 2.1 περιγράφει και σε αυτή την περίπτωση τη συνολική λειτουργία του μοντέλου. Σε αντίθεση με ένα MLP ένα CNN, αντί για πλήρως συνδεδεμένους νευρώνες χρησιμοποιεί σε κάθε *Συνελικτικό Στρώμα* έναν αριθμό διαφορετικών φίλτρων με τα οποία συνελίσσεται η είσοδος και παράγεται ένα χάρτης ενεργοποίησης (activation map). Στην περίπτωση που το δίκτυο χρησιμοποιεί δισδιάστατη συνέλιξη και η είσοδος είναι εικόνα, τα φίλτρα είναι ένας τρισδιάστατος τανυστής με διαστάσεις ίσες με: (Χωρικό πλάτος, Χωρικό μήκος, Αριθμός Καναλιών). Όταν χρησιμοποιείται τρισδιάστατη συνέλιξη και η είσοδος είναι ένα βίντεο τα φίλτρα είναι τετραδιάστατα με διαστάσεις ίσες με: (Χωρικό πλάτος, Χωρικό μήκος, Χρονική Διάρκεια ή Αριθμός Καρέ, Αριθμός Καναλιών). Αντίστοιχα, ο χάρτης ενεργοποίη-

σης που προκύπτει από την συνέλιξη με κάθε φίλτρο είναι δισδιάστατος στην περίπτωση των εικόνων και τρισδιάστατος στην περίπτωση των βίντεο. Τα βάρη του κάθε φίλτρου αποτελούν τις υπό μάθηση παραμέτρους του μοντέλου. Μέχρι αυτό το σημείο, η δράση του συνελικτικού στρώματος στην είσοδο είναι μόνο γραμμική καθώς η συνέλιξη αποτελεί γραμμικό τελεστή. Στη συνέχεια, όπως και στην περίπτωση των MLP, σε κάθε τιμή του χάρτη ενεργοποίησης εφαρμόζεται μία μη γραμμική συνάρτηση ενεργοποίησης και έτσι προκύπτει η τελική έξοδος του στρώματος. Εκτός των στρωμάτων συνέλιξης, είναι κοινή πρακτική, να χρησιμοποιούνται μέσα σε ένα CNN και *στρώματα συσσώρευσης* (max pooling layers) με τελεστή συσσώρευσης τη συνάρτηση τοπικού μεγίστου που εφαρμόζεται σε μη επικαλυπτόμενες περιοχές του κάθε χάρτη ενεργοποίησης.

Η χρήση της συνέλιξης στα Συνελικτικά Δίκτυα προσφέρει κάποιες χρήσιμες ιδιότητες οι οποίες έχουν περιγραφεί στις αναφορές [19, 69]. Πριν περιγράψουμε τον φορμαλισμό και τις μεθόδους εκπαίδευσης, κρίθηκε χρήσιμο να περιγραφούν οι ιδιότητες των Συνελικτικών μοντέλων που τα καθιστούν ελκυστικά στις εφαρμογές Όρασης Υπολογιστών και Μηχανικής Μάθησης και οι οποίες είναι:

- Αραιή ή Τοπική Συνδεσιμότητα (Sparse or Local Connectivity)
- Κοινή Χρήση Παραμέτρων (Parameter Sharing)
- Συμεταβλητότητα στις μετατοπίσεις (Equivariance to translation)

Η **Αραιή ή Τοπική Συνδεσιμότητα** επιτυγχάνεται μέσω της χρήσης φίλτρων των οποίων οι χωρικές (ή και χρονικές) διαστάσεις είναι σημαντικά μικρότερες από αυτές της εισόδου. Η σχεδιαστική αυτή επιλογή αποτελεί άλλη μία ιδέα που συχνά δανείζονται τα συστήματα Όρασης Υπολογιστών από τον εγκέφαλο MENTION HUBEl. Τυπικά για μία εικόνα χωρικών διαστάσεων 200×200 οι χωρικές διαστάσεις ενός φίλτρου θα μπορούσαν να είναι της τάξης του 10×10. Αυτή η διαφορά έχει σαν αποτέλεσμα για μία εικόνα 40000 pixel, ένα φίλτρο να συνδέεται τοπικά με μία υποπεριοχή της, μεγέθους μόλις 100 pixel κατά τον υπολογισμό ενός βήματος της συνέλιξης. Το γινόμενο των χωρικών διαστάσεων του φίλτρου ονομάζεται και *δεκτικό πεδίο* (receptive field) του κάθε νευρώνα. Η αξία της αραιής και τοπικής συνδεσιμότητας εντοπίζεται στη δυνατότητα χρήσης λιγότερων παραμέτρων και στην ανάγκη εκτέλεσης λιγότερων υπολογισμών για να υπολογιστεί η έξοδος ενός συνελικτικού στρώματος χάρτες. Επίσης, ενώ στα πρώτα στάδια του δικτύου οι έξοδοι συνελικτικών στρωμάτων περιγράφουν τοπικά χαρακτηριστικά, καθώς κινούμαστε προς τα βαθύτερα στάδια του δικτύου. Οι είσοδοι ενός συνελικτικού στρώματος αρχίζουν να εξαρτώνται από όλο και μεγαλύτερες περιοχές της αρχικής εισόδου, αφού είναι ταυτόχρονα συνάρτηση πολλών τοπικών χαρακτηριστικών, χαμηλότερου επιπέδου, που εντοπίζονται από τα προηγούμενα στάδια του δικτύου. Ουσιαστικά, τα βαθύτερα στρώματα ”εκπαιδεύονται” να συνδυάζουν την

πληροφορία που εξάγουν τα ρηχότερα στρώματα ώστε να παράξουν υψηλότερου επιπέδου αναπαραστάσεις της εισόδου.

Η **Κοινή Χρήση Παραμέτρων** βασίζεται αποκλειστικά στον τρόπο υπολογισμού της συνέλιξης έτσι ώστε οι παράμετροι ενός φίλτρου να χρησιμοποιούνται (σχεδόν) σε κάθε σημείο της εισόδου χωρίς να αλλάζουν. Διαισθητικά μπορούμε να περιγράψουμε την ιδιότητα αυτή αν παρομοιάσουμε τη συνέλιξη με την μεταφορά και εφαρμογή του φίλτρου πάνω σε διαφορετικές γειτονίες pixel της εικόνας. Σε αυτή την περίπτωση οι παράμετροι που χρησιμοποιούνται για την εκτέλεση υπολογισμών χρησιμοποιώντας ένα εκ των φίλτρων σε κάθε γειτονιά σημείων είναι οι ίδιες. Ως εκ τούτου, ένα συνελικτικό δίκτυο έχει τη δυνατότητα να εντοπίζει και να εξάγει τα ίδια ακριβώς οπτικά χαρακτηριστικά από διαφορετικές περιοχές της εισόδου. Ένα πλήρως συνδεδεμένο μοντέλο δεν θα είχε τη δυνατότητα να έχει αυτή την αμετάβλητη ως προς περιοχή της εισόδου συμπεριφορά, καθώς θα χρησιμοποιούσε μία διαφορετική παράμετρο για κάθε διαφορετική θέση της εισόδου.

Από την σκοπιά της υπολογιστικής πολυπλοκότητας, σε ένα στρώμα νευρώνων ενός MLP με m εισόδους και n εξόδους, απαιτούνται $\mathscr{O}(m \times n)$ υπολογισμοί ενώ όταν το δεκτικό πεδίο ενός συνελικτικού στρώματος ισούται με r απαιτούνται $\mathscr{O}(r \times n)$ υπολογισμοί. Στην πράξη το r μπορεί να είναι αρκετές τάξεις μεγέθους μικρότερο από το m γεγονός που επιταχύνει την διαδικασία σε σχέση με ένα πλήρως συνδεδεμένο στρώμα νευρώνων. Συμπερασματικά, μπορούμε να πούμε πως οι δύο πρώτες ιδιότητες των συνελικτικών δικτύων αυξάνουν δραματικά την αποδοτικότητα και περιορίζουν την διάσταση του μοντέλου σε λογικά πλαίσια παρά την υψηλή διάσταση της εισόδου.

Η ιδιότητα της συμεταβλητότητας σε μετατόπιση (equivariance to translation) οφείλεται τόσο στην κοινή χρήση παραμέτρων όσο και στη χρήση της συνέλιξης. Μία συνάρτηση f παρουσιάζει συμεταβλητότητα σε μία συνάρτηση g αν ισχύει $f(g(x)) = g(f(x))$. Διαισθητικά, λόγω της ιδιότητας αυτής, η μετατόπιση ενός αντικειμένου οδηγεί σε ίση μετατόπιση της αναπαράστασης του στον χάρτη ενεργοποίησης ενός συνελικτικού στρώματος. Αξίζει σε αυτό το σημείο να παρουσιάσουμε και μία σύντομη μαθηματική απόδειξη αυτής της ιδιότητας για τη δισδιάστατη συνέλιξη. Λόγω της προσεταιριστικής ιδιότητας της συνέλιξης, αν η g ανήκει στην οικογένεια των συναρτήσεων που μετατοπίζουν την είσοδο τους ώστε $g(I[i,j]) = I[i+a, j+b]$ και $f(I[i,j]) = (I * W)[i,j]$ τότε ισχύει:

$$f(g(I[i,j])) = f(I[i+a, j+b]) = \sum_l \sum_k I[i+a-l, j+b-k] W[l,k] \tag{2.12}$$

Εφόσον ισχύει $g(cI[i,j]) = cg(I[i,j])$ προκύπτει επίσης:

$$\begin{aligned} g(f(I[i,j])) &= g(\sum_l \sum_k I[i-l, j-k]W[l,k]) \\ &= \sum_l \sum_k g(I[i-l, j-k]W[l,k]) \\ &= \sum_l \sum_k I[i+a-l, j+b-k]W[l,k] \end{aligned} \tag{2.13}$$

Και συνεπώς ισχύει:

$$g(f(I[i,j])) = f(g(I[i,j])) \tag{2.14}$$

Αποδεικνύεται λοιπόν πως η συνέλιξη παρουσιάζει συμεταβλητότητα ως προς τις μετατοπίσεις. Σε αυτό το σημείο, πρέπει να επισημανθεί πως οι παραπάνω ευνοϊκές ιδιότητες οφείλονται αποκλειστικά στη χρήση της συνέλιξης. Παρόλα αυτά, τα συνελικτικά δίκτυα παρουσιάζουν και άλλα χρήσιμα χαρακτηριστικά που οφείλονται στη χρήση στρωμάτων συσσώρευσης τα οποία περιγράφονται στη συνέχεια.

2.1.3 Το Συνελικτικό Στρώμα (Convolutional Layer)

Ένα συνελικτικό στρώμα, όπως φανερώνει και η ονομασία του, συνελίσσει την είσοδο του με μία σειρά από φίλτρα ή πυρήνες (kernels), παράγοντας έτσι για κάθε φίλτρο έναν χάρτη ενεργοποίησης. Στην περίπτωση της δισδιάστατης συνέλιξης, αν η είσοδος είναι μία εικόνα ή ένας τρισδιάστατος τανυστής, η μαθηματική περιγραφή της λειτουργίας του συνελικτικού στρώματός για κάθε φίλτρο είναι:

$$S_l[i,j] = \text{ReLU}(b_l + \sum_{m=0}^{M_{l-1}} \sum_{n=0}^{N_{l-1}} W_l[m,n,:] \circ I[i-m, j-n, :]) \tag{2.15}$$

όπου l ο δείκτης που αντιστοιχεί στο φίλτρο με τρισδιάστατο τανυστή βαρών W_l, M_{l-1} το ύψος και N_{l-1}, b_l ένα διάνυσμα πόλωσης διάστασης ίσης με την τρίτη διάσταση του φίλτρου. Στη θέση της μη γραμμικής συνάρτησης ReLU θα μπορούσε να τοποθετηθεί οποιαδήποτε άλλη μη γραμμική συνάρτηση ενεργοποίησης. Στην πράξη, μέσω πειραματισμού έχει φανεί πως η ReLU δίνει τα καλύτερα αποτελέσματα σε πολλά προβλήματα [50, 3]. Μία σημαντική λεπτομέρεια που αποτελεί σχεδιαστική επιλογή, στην συντριπτική πλειοψηφία τέτοιων μοντέλων, είναι ότι ο αριθμός καναλιών της εισόδου ισούται με την τρίτη διάσταση του τανυστή των φίλτρων. Αυτό υποδηλώνεται από την εξίσωση 2.15, μέσω του συμβόλου " : " που υπονοεί πως όλα τα στοιχεία του τανυστή κατά μήκος αυτής της τρίτης διάστασης, για σταθερή γραμμή m και στήλη n, χρησιμοποιούνται κατά τον υπολογισμό του κάθε γινομένου. Συνεπώς, τα γινόμενα της εξίσωσης εκτελούνται στοιχείο προς στοιχείο ανάμεσα στα βάρη του φίλτρου και το δεκτικό πεδίο του επί της εισόδου.

Στην περίπτωση της τρισδιάστατης συνέλιξης, αν η είσοδος είναι ένα βίντεο ή ένας τετραδιάστατος τανυστής, η μαθηματική περιγραφή της λειτουργίας του συνελικτικού στρώματος για κάθε φίλτρο είναι:

$$S_l[i,j,k] = \ ReLU(b_l + \sum_{m=0}^{M_{l-1}} \sum_{n=0}^{N_{l-1}} \sum_{\tau=0}^{T_{l-1}} W_l[m,n,\tau,:] \circ I[i-m, j-n, k-\tau, :]) \tag{2.16}$$

όπου l ο δείκτης που αντιστοιχεί στο φίλτρο με τετραδιάστατο τανυστή βαρών W_l, M_{l-1} το ύψος, N_{l-1} το πλάτος και T_{l-1} η χρονική διάσταση του φίλτρου, b_l ένα διάνυσμα πόλωσης διάστασης ίσης με την τέταρτη διάσταση του φίλτρου. Και σε αυτή την περίπτωση ο αριθμός των καναλιών της εισόδου είναι ίσος με την τέταρτη διάσταση του φίλτρου. Και στις δύο περιπτώσεις που περιγράφηκαν παραπάνω, απαιτείται συχνά να χρησιμοποιηθεί padding στα όρια της εισόδου ώστε να καθίσταται εφικτός ο υπολογισμός της συνέλιξης χωρίς να μειωθεί σημαντικά η διάσταση της εισόδου στην έξοδο. Επίσης είναι δυνατόν να καθορίσουμε αν το φίλτρο θα εφαρμόζεται σε κάθε δυνατή θέση της εισόδου ή αν θα παραλείπονται κάποιες θέσεις μέσω μίας παραμέτρου που συνήθως ονομάζεται *βήμα (stride)*. Αφού καθοριστούν όλες αυτές οι υπερπαράμετροι του κάθε στρώματος μπορούμε να υπολογίσουμε τον αριθμό των υπό μάθηση παραμέτρων του δικτύου.

Αριθμός Παραμέτρων ενός Συνελικτικού Στρώματος

Έστω συνελικτικό στρώμα δισδιάστατης συνέλιξης που δέχεται σαν είσοδο έναν τρισδιάστατο τανυστή διαστάσεων $W_1 \times H_1 \times C_1$ (πλάτος, ύψος, κανάλια). Αν αυτό αποτελείται από φίλτρα διαστάσεων $M \times N \times C_1$ (πλάτος, ύψος, κανάλια) το καθένα, $Stride$ είναι το βήμα που χρησιμοποιείται κατά τον υπολογισμό της συνέλιξης, και Pad το εύρος του zero padding στα άκρα της εισόδου, τότε η έξοδος του στρώματος είναι διαστάσεων $W_2 \times H_2 \times C_2$ όπου:

$$W_2 = \frac{W_1 - M + 2Pad}{Stride} + 1, \ H_2 = \frac{H_1 - N + 2Pad}{Stride} + 1, \ C_2 = K \tag{2.17}$$

Με βάση τα παραπάνω και λαμβάνοντας υπόψη τη διάσταση του διανύσματος πόλωσης, το πλήθος των υπό μάθηση παραμέτρων του συνελικτικού στρώματος είναι:

$$card(P) = K \cdot (M \cdot N \cdot C_1) + K \tag{2.18}$$

2.1.4 Το Στρώμα Συσσώρευσης (Pooling Layer)

Στην συντριπτική πλειοψηφία των μοντέλων, λόγω της ανάγκης μείωσης της διάστασης του μοντέλου αλλά και τη βελτίωση της απόδοσης του μοντέλου, χρησιμοποιείται μετά από κάποια εκ των στρωμάτων συνέλιξης ένα *Στρώμα Συσσώρευσης*. Σε ένα τέτοιο στρώμα ενός συνελικτικού μοντέλου, η έξοδος S ενός προηγούμενου

στρώματος συνέλιξης υφίσταται τη δράση ενός τελεστή συσσώρευσης. Συνηθίζεται ο τελεστής αυτός να είναι η συνάρτηση τοπικού μεγίστου που εφαρμόζεται σε μη επικαλυπτόμενες υποπεριοχές του τανυστή εισόδου με αποτέλεσμα αυτός να υποδειγματοληπτείται. Σημειώνεται πως το στρώμα συσσώρευσης δεν προσθέτει υπό μάθηση παραμέτρους στο μοντέλο. Η μαθηματική περιγραφή της λειτουργίας του στρώματος συσσώρευσης μεγίστου (Max Pooling Layer) σε ένα δίκτυο δισδιάστατης ή τρισδιάστατης συνέλιξης περιγράφεται παρακάτω: Σε κάθε μη επικαλυπτόμενη υποπεριοχή διάστασης $M \times N$ ή $M \times N \times T$ (για δίκτυα δισδιάστατης και τρισδιάστατης συνέλιξης, αντίστοιχα) κάθε χάρτη χαρακτηριστικών (δηλαδή για κάθε κανάλι) S_l του προηγούμενου στρώματος του δικτύου εφαρμόζεται η εξής σχέση:

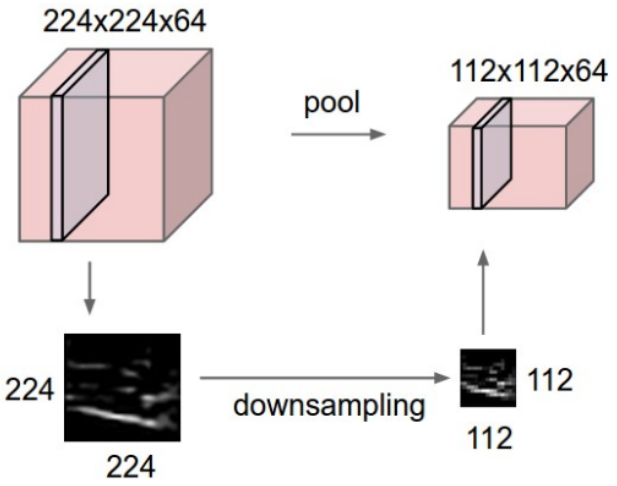
$$S_l^{Region} = \max_{\forall \mathbf{x} \in Region} \{S_l[\mathbf{x}]\} \tag{2.19}$$

Ουσιαστικά, με βάση την παραπάνω σχέση διατηρείται μόνο η μέγιστη ενεργοποίηση εντός μίας υποπεριοχής του χάρτη χαρακτηριστικών, ανεξαρτήτως των διαστάσεων της περιοχής (δισδιάστατης ή τρισδιάστατης). Το Σχήμα 2.4, απεικονίζει τη λειτουργία του στρώματος συσσώρευσης σε ένα δίκτυο δισδιάστατης συνέλιξης. Όπως και στην περίπτωση του στρώματος συνέλιξης συνηθίζεται να χρησιμοποιείται μία παράμετρος βήματος που καθορίζει αν η παραπάνω σχέση θα εφαρμοστεί σε κάθε υποπεριοχή της εισόδου ή αν αυτό θα συμβεί για λιγότερες από όλες τις δυνατές υποπεριοχές. Αν επιλεχθεί το δεύτερο, η είσοδος υποδειγματοληπτείται περαιτέρω. Συνεπώς, για ένα δίκτυο δισδιάστατης συνέλιξης, αν η είσοδος είναι ένας τανυστής διαστάσεων $W_1 \times H_1 \times C_1$ (πλάτος,ύψος,κανάλια), $Stride$ το βήμα και $M \times N \times$ η διάσταση των υποπεριοχών, τότε η έξοδος του στρώματος συσσώρευσης είναι ένας τανυστής διαστάσεων $W_2 \times H_2 \times C_2$ όπου:

$$W_2 = \frac{W_1 - M}{Stride} + 1,\ H_2 = \frac{H_1 - N}{Stride} + 1,\ C_2 = C_1 \tag{2.20}$$

2.1.5 Συδυασμός Στρωμάτων

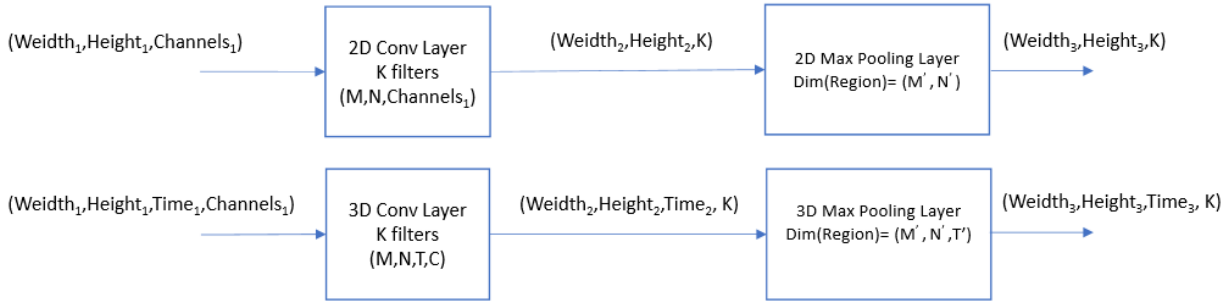
Η πιο συνηθισμένη δομή που συναντάται σε ένα συνελικτικό δίκτυο είναι η αλληλουχία ενός συνελικτικού στρώματος και ενός στρώματος συσσώρευσης. Μία τέτοια συνδεσμολογία φαίνεται στο Σχήμα 2.5. Εκτός της μείωσης της διάστασης ενός δικτύου, ο παραπάνω συνδυασμός συνέλιξης-συσσώρευσης δίνει τη δυνατότητα στο δίκτυο να έχει αμετάβλητη απόκριση σε μικρές μετατοπίσεις αντικειμένων ή χαρακτηριστικών στην είσοδο. Αυτή η ιδιότητα, οφείλεται στο γεγονός ότι αν μετατοπισθεί, ελαφρώς, ένα χαρακτηριστικό (π.χ μία ακμή ή γωνία) στο χώρο των pixel της εισόδου, τότε ο χάρτης ενεργοποίησης του φίλτρου που εντοπίζει το χαρακτηριστικό αυτό θα παρουσιάσει μεγαλύτερες τιμές σε θέσεις ελαφρώς μετατοπισμένες σε σχέση με πριν την μετατόπιση. Συνεπώς, εφόσον η μετατόπιση του χαρακτηριστικού είναι μικρότερη από το μέγεθος της περιοχής υποδειγματοληψίας



Σχήμα 2.4: Συσσώρευση Μεγίστου (Max Pooling). Σημειώνεται πως ο τελεστής μεγίστου εφαρμόζεται ξεχωριστά σε κάθε κανάλι (ή χάρτη ενεργοποίησης) της εισόδου με αποτέλεσμα ο αριθμός καναλιών εξόδου να μένει αμετάβλητος (στην περίπτωση αυτή 64). Λήφθηκε από την αναφορά [68].

επί της οποίας δρα το στρώμα συσσώρευση, η τελική απόκριση της αλληλουχίας των δύο στρωμάτων θα είναι σχεδόν η ίδια ανεξαρτήτως της μετατόπισης, καθώς μόνο η μέγιστη τιμή του χάρτη ενεργοποίησης εντός της κάθε υποπεριοχής του θα τροφοδοτηθεί στα επόμενα στρώματα του δικτύου.

Μία άλλη πολύ σημαντική ιδιότητα που προσφέρει ο συνδυασμός συνέλιξης-συσσώρευσης είναι η δυνατότητα ανάπτυξης αμετάβλητης απόκρισης του δικτύου σε διάφορα είδη μετασχηματισμών της εισόδου όπως η περιστροφή ή η κλιμάκωση, υπό κάποιες προϋποθέσεις. Η πρώτη προϋπόθεση είναι η ύπαρξη επαρκώς μεγάλου αριθμού φίλτρων (άρα και παραμέτρων) ώστε κατόπιν εκπαίδευσης αυτά να μάθουν να αναγνωρίζουν μετασχηματισμένες εκδοχές χαρακτηριστικών της εισόδου.



Σχήμα 2.5: Συνδυασμός Στρωμάτων Συνέλιξης και Συσσώρευσης σε Δίκτυο Τρισδιάστατης και Δισδιάστατης Συνέλιξης. Το σχήμα παρουσιάζει την διάσταση των τανυστών εισόδου και εξόδου κάθε στρώματος.

Συνηθίζεται σε αρχιτεκτονικές συνελικτικών δικτύων που αποσκοπούν στην επίλυση κάποιου προβλήματος ταξινόμησης είτε end-to-end είτε μέσω τροφοδότησης ενός διανύσματος χαρακτηριστικών σε κάποιον αλγόριθμο ταξινόμησης (π.χ

support vector machines, bayes ταξινομητής κλπ), να χρησιμοποιούνται στρώματα perceptron ή πλήρως συνδεδεμένα στρώματα νευρώνων, όπως συνηθέστερα αναφέρονται. Πιο συγκεκριμένα, κάθε στοιχείο του τανυστή εξόδου ενός συνελικτικού στρώματος ή ενός στρώματος συσσώρευσης αποτελεί στοιχείο ενός διανύσματος το οποίο αποτελεί την είσοδο ενός πλήρως συνδεδεμένου στρώματος. Η διαδικασία μετασχηματισμού των χαρτών χαρακτηριστικών σε διάνυσμα ονομάζεται *πλάτυνση (flattening)*. Τα χαρακτηριστικά και η παραμετροποίηση των πλήρως συνδεδεμένων στρωμάτων έχουν περιγραφεί στην υποενότητα 2.1.1.

2.1.6 Επιλογή Αρχιτεκτονικής Νευρωνικού Δικτύου

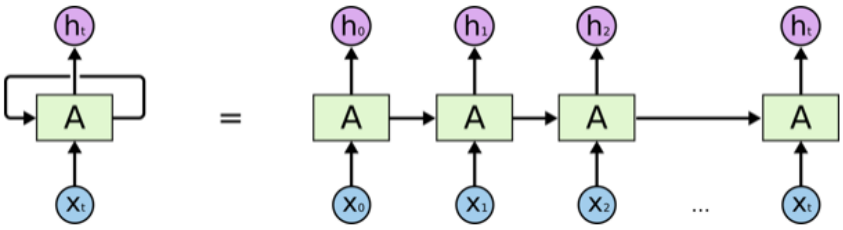
Τα συνελικτικά στρώματα, τα στρώματα συνέλιξης και τα πλήρως συνδεδεμένα στρώματα νευρώνων αποτελούν τα δομικά στοιχεία για να σχεδιαστεί ένα συνελικτικό δίκτυο. Παρότι η επιμέρους λειτουργία κάθε επιμέρους υπολογιστικής μονάδας είναι εύκολο να περιγραφεί και να αναλυθεί, η επιλογή του αριθμού στρωμάτων, ο συνδυασμός τους και ο καθορισμός των παραμέτρων τους (π.χ διαστάσεις φίλτρων, αριθμός φίλτρων, βήμα, υποπεριοχές συσσώρευσης) αποτελούν ένα ανοιχτό πρόβλημα της επιστημονικής περιοχής της μηχανικής μάθησης. Μέσω του εκτενούς πειραματισμού της επιστημονικής κοινότητας με τα βαθιά νευρωνικά μοντέλα, έχουν προκύψει κάποιοι εμπειρικοί κανόνες που διευκολύνουν τη σχεδίαση ενός μοντέλου. Αρχικά, η ανάπτυξη μίας αρχιτεκτονικής από την αρχή δεν είναι πάντα απαραίτητη. Πιο συγκεκριμένα, πολλά πρότυπα αρχιτεκτονικών που έχουν δοκιμαστεί σε βάσεις δεδομένων μεγάλης κλίμακας όπως το Imagenet [50], μπορούν να αποτελέσουν ένα αρχικό σημείο αναφοράς για την επιλογή των διαστάσεων των φίλτρων. Με βάση τα παραδείγματα των Alex Net [1], VGG Net [4] και Google LeNet που έχουν καταλάβει τις πρώτες θέσεις την τελευταία τετραετία στον διαγωνισμό ταξινόμησης εικόνας του Imagenet, έχει προκύψει πως τα συνελικτικά δίκτυα αποδίδουν καλύτερα όταν τα συνελικτικά στρώματα χρησιμοποιούν μικρών χωρικών διαστάσεων φίλτρα και κατά συνέπεια φίλτρα με μικρό δεκτικό πεδίο. Επίσης, μία μεθοδολογία που στην πράξη οδηγεί ευκολότερα στην εύρεση ενός μοντέλου που να μπορεί να εκπαιδευθεί και να δώσει ανταγωνιστικά αποτελέσματα ακρίβειας ταξινόμησης είναι η σταδιακή αύξηση του βάθους και του αριθμού των παραμέτρων ενός μοντέλου. Συγκεκριμένα, λαμβάνοντας υπόψη τους περιορισμούς μνήμης, υπολογιστικής ισχύος και δεδομένων, δοκιμάζονται όλο και βαθύτερες αρχιτεκτονικές μέχρι κάποιος από τους περιορισμούς να αποτρέπει την περαιτέρω επέκταση του μοντέλου. Παρόλα αυτά, πολλές φορές η σταδιακή ανάπτυξη ενός μοντέλου και η χρήση παραδειγμάτων αρχιτεκτονικών που έχουν αποδειχθεί αποτελεσματικές δεν επαρκεί. Για τον λόγο αυτό πολλές φορές η επιλογή της κατάλληλης αρχιτεκτονικής για την επίλυση ενός προβλήματος χρησιμοποιώντας βαθιά συνελικτικά (και όχι μόνο) δίκτυα απαιτεί εμπειρία, δοκιμές και χωρίς αμφιβολία και τύχη.

Κεφάλαιο 3

Αναδρομικά Νευρωνικά Δίκτυα (Recurrent Neural Networks)

3.1 Γενικά χαρακτηριστικά ενός αναδρομικού δικτύου

Τα αναδρομικά νευρωνικά δίκτυα είναι μία ειδική κατηγορία νευρωνικών δικτύων που επεξεργάζονται αποτελεσματικά κάθε είδους ακολουθιακά δεδομένα. Παραδείγματα, αποτελούν η φωνή, η γραφή, η οπτική πληροφορία που προκύπτει από μία κίνηση ή μία δράση ή ακόμα και τα pixel μίας εικόνας αν τα διατρέξουμε με κάποιο δομημένο τρόπο.



An unrolled recurrent neural network.

Σχήμα 3.1: ”Ξεδίπλωμα” ενός αναδρομικού δικτύου. Λήφθηκε από την αναφορά [37]

Ένα αναδρομικό δίκτυο μετασχηματίζει κάθε νέα είσοδο με τρόπο που εξαρτάται τόσο από την ίδια την είσοδο όσο και από τις προηγούμενες εισόδους που έχει δεχτεί. Φορμαλιστικά αυτή η βασική αρχή περιγράφεται ως εξής: Αν x_t είναι η είσοδος την χρονική στιγμή t, $f(\cdot)$ η συνάρτηση που περιγράφει την επίδραση του αναδρομικού δικτύου πάνω στην είσοδο και h_t η έξοδος του, τότε:

$$h_t = f(x_t, h_{t-1}) = f(x_t, f(x_{t-1}, h_{t-2})) = \cdots = f(x_t, f(x_{t-1}, \ldots, (f(x_1, h_0)) \ldots)) \tag{3.1}$$

Είναι φανερό πως υπάρχει στενή σχέση ανάμεσα στα γραφικά μοντέλα (graphical models) και τα αναδρομικά δίκτυα. Αυτή η σύνδεση ανάμεσα στις δύο κατηγορίες μοντέλων μπορεί να φανεί καλύτερα αν, όπως στο Σχήμα 3.1, το αναδρο-

μικό δίκτυο ”ξεδιπλωθεί”, σχεδιάζοντας ”εικονικές” υπολογιστικές μονάδες για να αναπαρασταθούν οι αναδρομικές εκτελέσεις της συνάρτησης f πάνω στις εισόδους $x_t, x_t+1, \ldots, x_T$. Σημειώνεται πως η έξοδος h_t συχνά αναφέρεται και ως κατάσταση του δικτύου. Στις παραπάνω εξισώσεις δεν εισήχθη για λόγους απλότητας η ύπαρξη ενός συνόλου παραμέτρων θ ή βαρών που όπως και στις άλλες περιπτώσεις νευρωνικών μοντέλων καθορίζουν την επίδραση του μοντέλου πάνω στην είσοδο τους. Οι παράμετροι αυτοί, και για τα αναδρομικά νευρωνικά δίκτυα είναι υπό μάθηση. Τα βάρη σε ένα αναδρομικό μοντέλο μπορούν να διαχωριστούν σε δύο κατηγορίες: αυτά που επιδρούν πάνω στην είσοδο (W) και αυτά που καθορίζουν την σημασία που δίνεται στην προηγούμενη κατάσταση του μοντέλου για τον υπολογισμό της επόμενης κατάστασης (U). Η γενική μορφή των εξισώσεων ενός αναδρομικού δικτύου είναι η εξής:

$$h_t = \phi(Wx_t + Uh_{t-1} + b) \tag{3.2}$$

όπου $\phi(\cdot)$ μία μη γραμμική συνάρτηση ενεργοποίησης, W ο πίνακας παραμέτρων που επιδρούν πάνω στην είσοδο x_t, U ο πίνακας παραμέτρων που επιδρούν πάνω στην έξοδο του δικτύου την προηγούμενη χρονική στιγμή και b ένα διάνυσμα πόλωσης. Λόγω αυτού του μηχανισμού τα αναδρομικά δίκτυα έχουν τη δυνατότητα να μοντελοποιούν χρονικές εξαρτήσεις ακόμα και ανάμεσα σε μη συνεχόμενες παρατηρήσεις, αφού μέσω της αναδρομικότητας υλοποιούν έναν μηχανισμό μνήμης. Όμως, στη γενική μορφή τους τα νευρωνικά αναδρομικά δίκτυα δεν καταφέρνουν να μοντελοποιήσουν αποτελεσματικά εξαρτήσεις μακράς διάρκειας (όπου η χρήση του όρου μακράς είναι σχετική και εξαρτάται από το πρόβλημα). Η δυσκολία αυτή πηγάζει από το γεγονός ότι ο αλγορίθμος Back Propagation στο χρόνο, που περιγράφεται στην ενότητα 4.2, δεν διατηρεί μεγάλες τιμές κλίσεων όσο εξετάζουμε όλο και παλαιότερα τμήματα της ακολουθίας. Όπως φαίνεται και από την εξίσωση 3.1, ο υπολογισμός της κλίσης του κόστους ως προς κάποια παρελθοντική είσοδο περιλαμβάνει την παραγώγιση μίας σύνθεσης συναρτήσεων που οδηγεί σε ένα όλο και αυξανόμενου αριθμού παραγόντων γινόμενου, καθώς κινούμαστε προς το παρελθόν. Κάποιο από τους παράγοντες αυτού του γινομένου είναι πιθανό να οδηγήσουν σε αστάθεια το συνολικό αποτέλεσμα αυξάνοντας ή μειώνοντας υπερβολικά την κλίση του κόστους ως προς τις παραμέτρους. Οι ανανεώσεις των παραμέτρων αυτών που περιγράφει ο αλγόριθμος S.G.D βασίζονται στον υπολογισμό της κλίσης και συνεπώς αν σε αυτόν παρουσιαστούν υπερβολικά μεγάλες ή μικρές τιμές η μάθηση θα αποτύχει. Το πρόβλημα αυτό συνήθως αναφέρεται *ως εξαφάνιση ή εκτόξευση της κλίσης (the vanishing or exploding gradient problem)* και έχει σαν αποτέλεσμα η διαδικασία εκπαίδευσης να μην εντοπίζει ικανοποιητικά καλές τιμές για τα βάρη του μοντέλου. Η παραπάνω δυσλειτουργία των αναδρομικών δικτύων, αντιμετωπίστηκε μέσω της εισαγωγής των νευρώνων Μακράς-Βραχείας Μνήμης (Long Term Short Term Memory ή L.S.T.M) που προτάθηκαν πρώτη φορά από τους Hochreiter

και Schmidhuber [62]. Αυτό το πιο εύρωστο είδος αναδρομικών δικτύων έχει φανεί πολύ αποτελεσματικό σε εφαρμογές όπως η επεξεργασία και μετάφραση φυσικής γλώσσας [39] ή η αναγνώριση συνεχούς χειρόγραφου [35, 36]. Επίσης έχει χρησιμοποιηθεί, σε συνδυασμό με συνελικτικά δίκτυα, για την αναγνώριση ανθρώπινης δράσης στα [56, 2, 26, 20] όπως γίνεται και στην παρούσα εργασία και παρουσιάζεται στο 5ο Κεφάλαιο.

3.2 Οι νευρώνες Μακράς-Βραχείας Μνήμης (Long Term Short Term Memory)

Για την παρουσίαση των νευρώνων LSTM, αρχικά θα περιγραφεί ο φορμαλισμός που τους διέπει και στη συνέχει θα δοθεί μία σύντομη ποιοτική ερμηνεία της λειτουργίας τους. Θεωρούμε ότι η κατάσταση ενός νευρώνα LSTM τη χρονική στιγμή t είναι μια συλλογή από διανύσματα του χώρου $\mathbb{R}^d$. Συγκεκριμένα έχουμε:

- Τη θύρα εισόδου (input gate) i_t
- Τη θύρα λησμόνησης (forget gate) f_t
- Ένα κύτταρο μνήμης (memory cell) c_t
- Τη θύρα εξόδου (output gate) o_t
- Τη κατάσταση ή κρυφή κατάσταση (hidden state) h_t

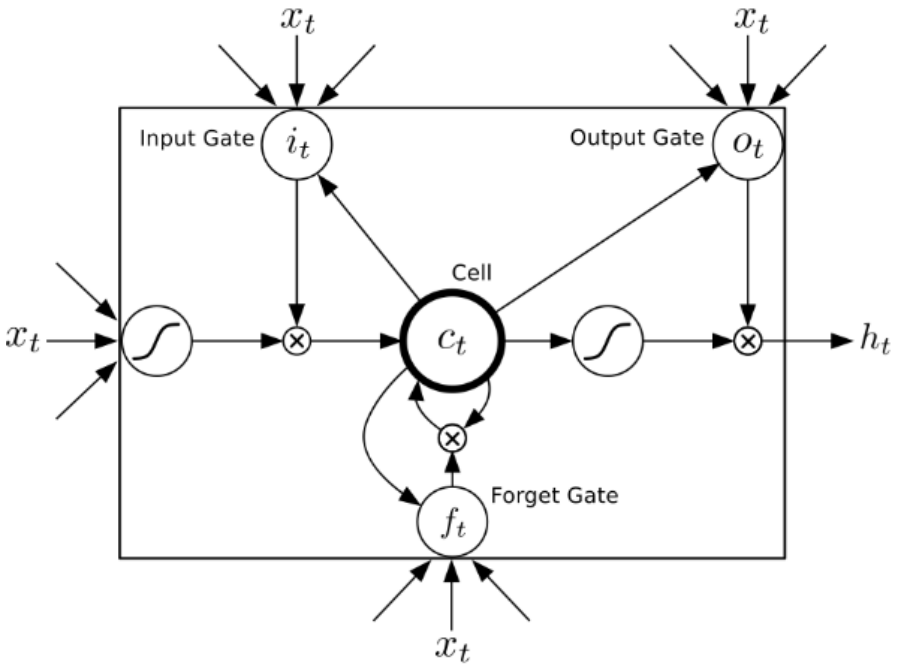
Δεδομένου ότι οι μη γραμμικές συναρτήσεις ενεργοποίησης που χρησιμοποιούνται είναι η σιγμοειδής συνάρτηση $\sigma(\cdot)$ και η υπερβολική εφαπτομένη $\tanh(\cdot)$, οι εξισώσεις κατάστασης που χαρακτηρίζουν έναν νευρώνα LSTM είναι οι εξής:

$$i_t = \sigma(W^i x_t + U^i h_{t-1} + b^i) \tag{3.3}$$
$$f_t = \sigma(W^f x_t + U^f h_{t-1} + b^f) \tag{3.4}$$
$$o_t = \sigma(W^o x_t + U^o h_{t-1} + b^o) \tag{3.5}$$
$$u_t = \tanh(W^u x_t + U^u h_{t-1} + b^u) \tag{3.6}$$
$$c_t = i_t \odot u_t + f_t \odot c_{t-1} \tag{3.7}$$
$$h_t = o_t \odot \tanh(c_t) \tag{3.8}$$

όπου x_t η είσοδος τη χρονική στιγμή t, b ένα διάνυσμα πόλωσης και $\odot$ το γινόμενο *Hadamard*. Διαισθητικά μπορούμε να ερμηνεύσουμε τη λειτουργία του μοντέλου ως εξής: πως η θύρα λησμόνησης ελέγχει το κατά πόσο οι παρελθούσες καταστάσεις πρέπει να ληφθούν υπόψη κατά το επόμενο βήμα, η θύρα εισόδου ελέγχει το ποια διανύσματα του νευρώνα θα μεταβληθούν και σε ποιο βαθμό,

- Η θύρα λησμόνησης ελέγχει το κατά πόσο πρέπει το περιεχόμενο του κυττάρου μνήμης c_t να ξεχαστεί κατά τη μετάβαση στο επόμενο βήμα λειτουργίας.

- Η θύρα εξόδου ελέγχει την έκθεση της εσωτερικής μνήμης στα υπόλοιπα μέρη του νευρώνα.
- Η θύρα εισόδου ελέγχει το ποια διανύσματα του νευρώνα θα μεταβληθούν και σε ποιο βαθμό.
- Η κρυφή κατάσταση αναπαριστά μία περιορισμένη και ελεγχόμενη εξωτερίκευση της αποθηκευμένης στο κύτταρο πληροφορίας η οποία εξαρτάται και από την νέα είσοδο x_t και όλες τις προηγούμενες λειτουργίες.



Σχήμα 3.2: Παράδειγμα LSTM νευρώνα. Λήφθηκε από [54]

Στα πειράματα του 5ου Κεφαλαίου, όσον αφορά τα αναδρομικά δίκτυα χρησιμοποιούνται στρώματα νευρώνων L.S.T.M. Με τον όρο στρώμα εννοούμε την συστοιχία αναδρομικών υπολογιστικών μονάδων που κάθε μία ξεχωριστά παράγει ένα ξεχωριστό διάνυσμα εξόδου με βάση την ακολουθία εισόδου και εφαρμογή των εξισώσεων 3.3.

Κεφάλαιο 4

Εκπαίδευση Ενός Νευρωνικού Δικτύου

Εκτός από τη διαθεσιμότητα βάσεων δεδομένων μεγάλης κλίμακας και πολλαπλάσιας, σε σχέση με πριν 30 χρόνια, υπολογιστικής ισχύος, ένας άλλος ουσιαστικός λόγος που βοήθησε εξίσου στην αλματώδη εξέλιξη και διάδοση των νευρωνικών μοντέλων, είναι η ύπαρξη αποδοτικών αλγορίθμων εκπαίδευσης των μοντέλων. Παρότι υπάρχουν αρκετές διαφορετικές μέθοδοι εκπαίδευσης, η συντριπτική πλειοψηφία αυτών, στηρίζονται σε δύο αλγορίθμους: τον αλγόριθμο *Back Propagation* και τον αλγόριθμο *Stochastic Gradient Descent* ή *SGD*. Ο αλγόριθμος *SGD* είναι μία σχετικά απλή εκδοχή της οικογένειας μεθόδων κατάβασης κλίσης (*Gradient Descent*) κατά την οποία κάθε υπό μάθηση παράμετρος μεταβάλλεται με βάση την *κλίση (gradient)* του κόστους ως προς την παράμετρο αυτή. Ο αλγόριθμος *Back Propagation* ή αλγόριθμος *οπίσθιας τροφοδότησης σφάλματος*, παρότι είχε περιγραφεί νωρίτερα από διάφορες εργασίες [5, 6], περιγράφηκε για πρώτη φορά στο πλαίσιο των τεχνητών νευρωνικών δικτύων από τον Werbos το 1982, ενώ επιβεβαιώθηκε πειραματικά η σημασία του για τα νευρωνικά μοντέλα από τον Rumelhart το 1986 [32]. Ο αλγόριθμος αυτός χρησιμοποιείται για την εξαγωγή μίας σχέσης μέσω της οποία υπολογίζεται αποδοτικά η αριθμητική τιμή της κλίσης του κόστους ως προς κάθε παράμετρο του μοντέλου. Ο αλγόριθμος Back Propagation όπως περιέγραψε ο Werbos [49], είναι εφικτό να γενικευθεί και στην περίπτωση των αναδρομικών μοντέλων. Στην περίπτωση αυτή, ονομάζεται *Back Propagation Through Time* ο οποίος επίσης περιγράφεται στο παρόν κεφάλαιο.

4.1 Stochastic Gradient Descent

4.1.1 Θεωρητική Θεμελίωση

Η εκπαίδευση ενός νευρωνικού δικτύου -όπως και οποιουδήποτε άλλου μοντέλου στα πλαίσια της μηχανικής μάθησης- είναι η εύρεση των τιμών των παραμέτρων που περιγράφουν μία συνάρτηση πρόβλεψης, μέσω μίας διαδικασίας βελτιστοποίησης ενός κριτηρίου ή κόστους επί του συνόλου δεδομένων εκπαίδευσης (training dataset). Σε αντίθεση με ένα πρόβλημα -αποκλειστικά- βελτιστοποίησης, η εκπαί-

δευση ενός νευρωνικού δικτύου επιδιώκει ταυτόχρονα τη βελτίωση της δυνατότητας που έχει το μοντέλο να γενικεύει τη γνώση που απέκτησε για το πρόβλημα μέσω των δεδομένων εκπαίδευσης, σε δεδομένα επί των οποίων δεν έχει εκπαιδευθεί (testing dataset). Ουσιαστικά, επιδιώκεται να ελαχιστοποιηθεί μια συνάρτηση κόστους με σκοπό έμμεσα να βελτιστοποιηθεί κάποια μετρική μέσω της οποίας αξιολογείται η απόδοση ενός μοντέλου. Με βάση το [69], η γενική μορφή μίας συνάρτησης κόστους περιγράφεται από τη σχέση:

$$J(\boldsymbol{\theta}) = \mathbf{E}_{(\mathbf{x},y)\ \hat{p}_{data}} L(f(\mathbf{x};\boldsymbol{\theta}), y) \tag{4.1}$$

όπου $\hat{p}_{data}$ είναι η εμπειρικά εκτιμώμενη κατανομή από την οποία θεωρείται πως παράγονται τα ζεύγη $(\mathbf{x}^{(i)}, y^i)$ του συνόλου δεδομένων εκπαίδευσης, L κάποια συνάρτηση κόστους, f η συνάρτηση που υλοποιεί το μοντέλο η οποία βασίζεται στις παραμέτρους $\boldsymbol{\theta}$. Ιδανικά, θα θέλαμε να γνωρίζουμε την πραγματική κατανομή p_{data}, προκειμένου να μπορούμε να ελαχιστοποιήσουμε άμεσα το πραγματικό σφάλμα γενίκευσης που δίνεται από την εξής σχέσης:

$$J^*(\boldsymbol{\theta}) = \mathbf{E}_{(\mathbf{x},y)\ p_{data}} L(f(\mathbf{x};\boldsymbol{\theta}), y) \tag{4.2}$$

Στην πράξη μπορούμε μόνο να εκτιμήσουμε την κατανομή $p_{data}(\mathbf{x}, y)$ μέσω των δειγμάτων που έχουμε στην διάθεση μας. Ουσιαστικά, η ελαχιστοποίηση του κόστους της εξίσωσης 4.1, ισοδυναμεί με την ελαχιστοποίηση του μέσου σφάλματος επί των δεδομένων εκπαίδευσης, δηλαδή:

$$J(\boldsymbol{\theta}) = \frac{1}{D} \sum_{i=1}^{D} L(f(\mathbf{x}^{(i)};\boldsymbol{\theta}), y^{(i)}) \tag{4.3}$$

Συνεπώς, η διαδικασία εκπαίδευσης ενός νευρωνικού δικτύου στηρίζεται στην *ελαχιστοποίηση του εμπειρικού ρίσκου (Empirical Risk Minimization)* διότι καλούμαστε να ελαχιστοποιήσουμε τη συνάρτηση κόστους 4.1, η οποία καλείται και *εμπειρικό ρίσκο* αντί της συνάρτησης κόστους 4.2, η οποία καλείται *ρίσκο*. Η έμμεση και προσεγγιστική ελαχιστοποίηση του κόστους της εξίσωσης 4.2 μέσω του κόστους της εξίσωσης 4.3 είναι μία από τις αιτίες που σχεδόν ποτέ η εκπαίδευση ενός βαθιού νευρωνικού δικτύου δεν καταλήγει σε κάποιο ολικό μέγιστο της συνάρτησης κόστους. Ειδικότερα, οι Bottu και Bousquet [46] κατέγραψαν πως για μία υποβέλτιστη λύση της παραμετροποιημένης συνάρτηση πρόβλεψης $\tilde{f}(\mathbf{x};\boldsymbol{\theta})$, δηλαδή για την οποία ισχύει $L(\tilde{f}) > L(f^*)$ υπάρχουν τρεις πηγές επιπρόσθετου σφάλματος:

- Το σφάλμα προσέγγισης που οφείλεται στο βαθμό στον οποίο η επιλεγμένη οικογένεια συναρτήσεων στην οποί ανήκει η $\tilde{f}$ μπορεί να προσεγγίσει την f^*. Η μείωση αυτού μπορεί να επιτευχθεί αν επιλεγεί μία μεγαλύτερη οικογένεια συναρτήσεων μέσα στην οποία θα αναζητηθεί η λύση.
- Το σφάλμα εκτίμησης που οφείλεται στο γεγονός ότι ελαχιστοποιείται το εμπειρικό ρίσκο αντί του πραγματικού ρίσκου. Η μείωση αυτού μπορεί να επιτευχθεί

αν επιλεγεί μικρότερη οικογένεια συναρτήσεων και υπάρχουν περισσότερα δεδομένα εκπαίδευσης.

- Το σφάλμα βελτιστοποίησης που συνδέεται με τον αριθμό επαναλήψεων του αλγορίθμου εκπαίδευσης. Όπως είναι αναμενόμενο, όσο περισσότερος χρόνος διατεθεί στη διαδικασία βελτιστοποίησης, τόσο περισσότερο θα μειωθεί το σφάλμα αυτό.

Όπως γίνεται εύκολα αντιληπτό, από τον παραπάνω διαχωρισμό, δεν υπάρχει ξεκάθαρη στρατηγική που αν ακολουθηθεί εξασφαλίζει καλύτερη σύγκλιση στην βέλτιστη τιμή των παραμέτρων $\boldsymbol{\theta}$. Για παράδειγμα η διεύρυνση της οικογένειας συναρτήσεων μπορεί να οδηγήσει σε αύξηση του σφάλματος εκτίμησης. Επίσης το πόσος επιπλέον χρόνος απαιτείται να διατεθεί στη διαδικασία εκπαίδευσης ώστε να μειωθεί σημαντικά η τιμή της συνάρτησης κόστους εξαρτάται τόσο από τον αριθμό των παραδειγμάτων εκπαίδευσης όσο και από την οικογένεια συναρτήσεων που χρησιμοποιείται.

Ένα άλλο σημαντικό στοιχείο που χαρακτηρίζει την εκπαίδευση ενός βαθιού νευρωνικού δικτύου είναι ότι οι επιφάνειες κόστους εντός του πολυδιάστατου χώρου των παραμέτρων είναι σύνθετες και κυριότερα *μη κυρτές (non-convex)*. Κατά συνέπεια οι διαδικασίες βελτιστοποίησης συχνά υποφέρουν από την ύπαρξη *επίπεδων περιοχών (plateau)* που οδηγούν σε πολύ χαμηλές κλίσεις του κόστους και επιβραδύνουν σημαντικά τη μάθηση. Ακόμα, συχνά εντοπίζονται επίπεδες περιοχές που περιβάλλονται από τιμές υψηλού κόστους και σε περίπτωση που η εκπαίδευση καταλήξει σε τέτοιες περιοχές "παγιδεύουν" τη διαδικασία εκπαίδευσης [43]. Παρόλα αυτά η εργασία [31] υποστηρίζει, μέσω θεωρητικής ανάλυσης και κάποιων μη ρεαλιστικών στην πράξη υποθέσεων, ότι η επιφάνεια κόστους πολυστρωματικών νευρωνικών μοντέλων παρουσιάζει μεγάλο αριθμό τοπικών ελαχίστων εκ των οποίων τα περισσότερα τείνουν να οδηγούν σε υψηλή ικανότητα γενίκευσης. Επίσης στην ίδια εργασία υποστηρίζεται πως ο αριθμός των τοπικών ελαχίστων που οδηγούν σε χαμηλή ικανότητα γενίκευσης τείνει να φθίνει καθώς αυξάνεται το μέγεθος του μοντέλου.

4.1.2 Αλγόριθμος Stochastic Gradient Descent και οι παραλλάγες του

Η βασική εκδοχή του αλγορίθμου κατάβασης κλίσης που συνήθως χρησιμοποιείται για την εκπαίδευση ενός νευρωνικού δικτύου ονομάζεται Stochastic Gradient Descent και εφαρμόζεται επαναληπτικά σε κάθε παράδειγμα εκπαίδευσης. Πιο συγκεκριμένα με βάση κάθε παράδειγμα εκπαίδευσης εκτιμάται η κλίση(ή παράγωγος ή gradient) του κόστους ως προς τις παραμέτρους και στην συνέχεια οι παράμετροι του μοντέλου μεταβάλλονται κατά έναν όρο ανάλογο της κλίσης που υπολογίστηκε. Η σχέση που περιγράφει την επαναληπτική διαδικασία εύρεσης των παραμέτρων

του μοντέλου είναι:

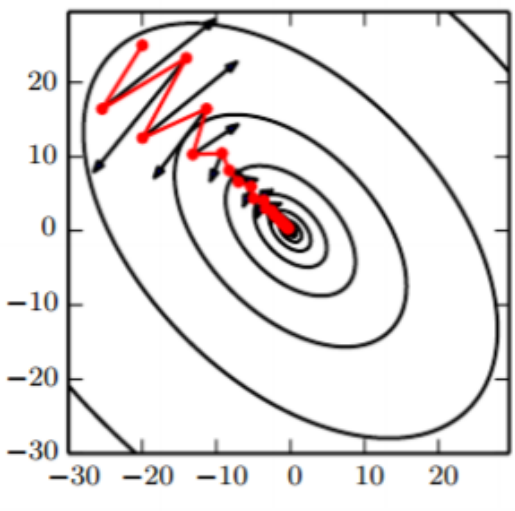
$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \lambda \nabla_{\boldsymbol{\theta}} L(f(\mathbf{x}^{(i)}; \boldsymbol{\theta}_t), y^{(i)}) \text{ για } i = 1...D \tag{4.4}$$

όπου λ ο *ρυθμός μάθησης (learning rate)*, που είναι μία υπερπαράμετρος η οποία ελέγχει το πόσο μεταβάλλονται οι παράμετροι του μοντέλου, με βάση την κλίση του κόστους. Η σημασία του ρυθμού μάθησης και του προγραμματισμού του κατά την εκπαίδευση είναι ένα ανοιχτό πρόβλημα και οι οποίες μεθοδολογίες χρησιμοποιούνται για την ρύθμιση του για μη-κυρτές επιφάνειες κόστους, στηρίζονται σε εμπειρικές ή πειραματικές παρατηρήσεις. Εμπειρικά έχει φανεί πως για βαθιά νευρωνικά μοντέλα και μέτριας έως μεγάλης κλίμακας σύνολα δεδομένων, ο παραπάνω αλγόριθμος συγκλίνει αργά καθώς η εκτίμηση της κλίσης ανά δείγμα μπορεί να εμφανίζει μεγάλη διασπορά με αποτέλεσμα να εμφανίζονται ταλαντώσεις της τιμή της συνάρτησης κόστους που επιβραδύνουν την εκπαίδευση. Η εμφάνιση των ταλαντώσεων μπορεί να περιοριστεί, σε κάποιο βαθμό, αν επιλεχθούν μικρότερες τιμές ρυθμού μάθησης, κάτι που όμως θα επιβραδύνει ακόμα περισσότερο τη διαδικασία εκπαίδευσης.

Η πιο ευρέως χρησιμοποιούμενη εκδοχή του αλγορίθμου SGD που έχει φανεί πως οδηγεί σε σταθερότερη σύγκλιση της διαδικασίας είναι ο αλγόριθμος *Stochastic Gradient Descent με Minibatches και χρήση όρου ορμής (Momentum term)*. Ο αλγόριθμος αυτός περιλαμβάνει δύο ουσιαστικές τροποποιήσεις σε σχέση με το βασικό αλγόριθμο της σχέσης 4.4. Αρχικά σε κάθε επανάληψη, η εκτίμηση της κλίσης του κόστους ως προς τις παραμέτρους του μοντέλου γίνεται με βάση ένα τυχαία δειγματοληπτημένο, προκαθορισμένου μεγέθους, υποσύνολο των δεδομένων εκπαίδευσης που καλείται συνήθως *Minibatch*.

Η τροποποίηση αυτή οδηγεί σε σημαντικά χαμηλότερη διακύμανση των ανανεώσεων των παραμέτρων και συνεπώς ομαλότερη σύγκλιση καθώς προσεγγίζεται ακριβέστερα η κλίση της συνάρτησης κόστους. Προκύπτει έτσι μία δεύτερη σημαντική υπερπαράμετρος που πρέπει να επιλεχθεί, η οποία είναι το μέγεθος του υποσυνόλου των δεδομένων που χρησιμοποιούνται σε κάθε επανάληψη του αλγορίθμου. Διαισθητικά μπορούμε να πούμε πως όταν αυξηθεί το μέγεθος του υποσυνόλου (Minibatch) λαμβάνουμε υψηλότερης ακρίβειας εκτίμηση της κλίσης. Επίσης η χρήση υπολογιστικών βελτιστοποιήσεων που χρησιμοποιούν οι κάρτες γραφικών αξιοποιείται στο μέγιστο όταν το μέγεθος του Minibatch τίθεται στη μέγιστη δυνατή τιμή που μπορεί να υποστηρίξει το υλικό με βάση τη μνήμη που παρέχεται. Από την άλλη η διατήρηση μικρότερου μεγέθους Minibatch επιτρέπει την ύπαρξη μίας ελεγχόμενης τιμής θορύβου λόγω της διακύμανσης στην εκτίμηση της παραγώγου που σε κάποιο βαθμό λειτουργεί σαν παράγοντας κανονικοποίησης των ανανεώσεων των βαρών.

Η δεύτερη τροποποίηση προτάθηκε αρχικά από τον Polyak [51] και αποσκοπεί στην επιτάχυνση της διαδικασίας εκπαίδευσης. Πιο συγκεκριμένα σε περιοχές γύρω από ένα σημείο τοπικού ελαχίστου, όπου οι κλίσεις είναι σχετικά υψηλές, η



Σχήμα 4.1: Σχηματική απεικόνιση της πορείας της τιμής κόστους προς κάποια περιοχή τοπικού ελαχίστου μίας τετραγωνικής συνάρτησης κόστους στην περίπτωση του αλγορίθμου SGD με Minibatches με (κόκκινη γραμμή) και χωρίς όρο ορμής (μαύρη γραμμή). Οι ελλείψεις του σχήματος αποτελούνται από σημεία ίσου κόστους το οποίο φθίνει καθώς κινούμαστε από την εξωτερική έλλειψη προς τις εσωτερικές. Όπως φανερώνει η σύγκριση ανάμεσα στις δύο πορείες ο όρος ορμής περιορίζει τις ταλαντώσεις και σταδιακά οι συσσωρευμένες μεγάλες κλίσεις ωθούν το κόστος ταχύτερα προς το τοπικό ελάχιστο. Λήφθηκε από την αναφορά [69]

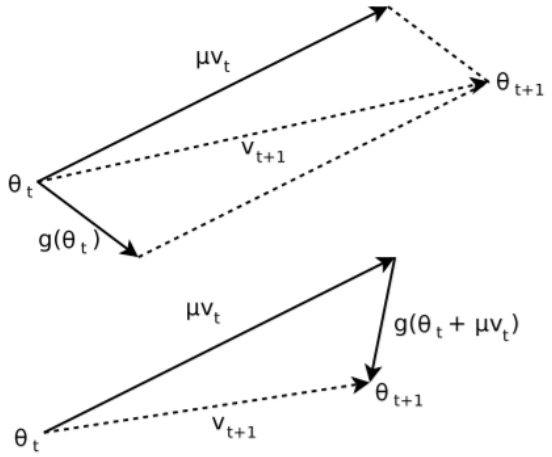
σύγκλιση επιταχύνεται αφού για κάθε ανανέωση λαμβάνονται υπόψη και οι κλίσεις στα προηγούμενα βήματα. Διαισθητικά μπορούμε να περιγράψουμε τον μηχανισμό της ορμής ως εξής: Καθώς οι κλίσεις που υπολογίζονται σε έναν αριθμό συνεχόμενων βημάτων γίνονται μεγαλύτερες, δρουν συσσωρευτικά και ωθούν γρηγορότερα το κόστος προς το σημείο που υποδεικνύεται από τις κλίσεις που υπολογίζονται. Αντίθετα όταν μία σειρά από προηγούμενες κλίσεις δεν είναι μεγάλες η επίδραση του όρου ορμής είναι περιορισμένη. Ο όρος ορμής προκύπτει ως το γινόμενο ενός σταθερού συντελεστή $m \in [0, 1)$ και ενός μεταβλητού παράγοντα u_t που εκφράζει την ταχύτητα μεταβολής ή "κίνησης" των παραμέτρων του μοντέλου. Όσο μεγαλύτερη τιμή λαμβάνει ο συντελεστής μ σε σχέση με τον ρυθμό μάθησης λ τόσο πιο ισχυρή είναι η επίδραση των προηγούμενων κλίσεων στην επόμενη ανανέωση των βαρών. Με βάση τα παραπάνω προκύπτει η παρακάτω εξίσωση ανανέωσης των παραμέτρων του μοντέλου:

$$\mathbf{u}_{t+1} = \mu \mathbf{u}_t - \lambda \frac{1}{B} \sum_{i}^{B} \nabla_{\boldsymbol{\theta}} L(f(\mathbf{x}^{(i)}; \boldsymbol{\theta}_t), y^{(i)}) \tag{4.5}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \mathbf{u}_{t+1} \tag{4.6}$$

όπου B το μέγεθος του Minibatch και $i : i + B$ οι δείκτες των παραδειγμάτων που λαμβάνονται με τυχαία δειγματοληψία από το σύνολο δεδομένων εκπαίδευσης.

Μία επιπλέον βελτίωση της παραπάνω μεθόδου είναι η χρήση *επιτάχυνσης Nesterov* που πρόσφατα προτάθηκε [29], και βασίζεται σε μία παλαιότερη εργασία του Nesterov [28]. Η διαφοροποίηση σε σχέση με τη χρήση απλού όρου ορμής είναι ότι η κλίση



Σχήμα 4.2: Σχηματική αναπαράσταση της επίδρασης του όρου ορμής στην ανανέωση των βαρών για την απλή εκδοχή του αλγορίθμου SGD με όρο ορμής (πάνω) και την εκδοχή που χρησιμοποιεί επιτάχυνση Nesterov (κάτω). Με $g()$ συμβολίζεται η κλίση. Η διαφορά ανάμεσα στις δύο μεθόδους είναι οι τιμές των παραμέτρων $\boldsymbol{\theta}$ για τις οποίες υπολογίζεται η κλίση. Στην πρώτη περίπτωση (πάνω) ο υπολογισμός της κλίσης γίνεται για τις τιμές $\boldsymbol{\theta}_t$ που προέκυψαν από την προηγούμενη επανάληψη, ενώ στην δεύτερη περίπτωση (κάτω) ο υπολογισμός της κλίσης γίνεται για τις τιμές $\boldsymbol{\theta}_t$ μετατοπισμένες κατά $\mu\mathbf{u}_t$ Λήφθηκε από την αναφορά [29]

υπολογίζεται για τιμές των παραμέτρων που έχουν μετατοπισθεί πρώτα κατά τον όρο ορμής της προηγούμενης επανάληψης. Στη συνέχεια υπολογίζεται ο νέος όρος ορμής και ανανεώνονται οι παράμετροι ακριβώς όπως και στην προηγούμενη περίπτωση. Το Σχήμα 4.2 παρουσιάζει την διαφορά ανάμεσα στις δύο μεθόδους. Οι τροποποιημένες εξισώσεις είναι οι εξής:

$$\mathbf{u}_{t+1} = \mu\mathbf{u}_t - \lambda\frac{1}{B}\sum_{i}^{B}\nabla_{\boldsymbol{\theta}}L(f(\mathbf{x}^{(i)};\boldsymbol{\theta}_t + \mu\mathbf{u}_t), y^{(i)}) \tag{4.7}$$

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \mu\mathbf{u}_{t+1} \tag{4.8}$$

Έχει φανεί πως αυτή η σχετικά απλή τροποποίηση στον υπολογισμό της κλίσης δίνει τη δυνατότητα στον όρο ορμής να προσαρμόζεται ταχύτερα σε σχέση με την απλούστερη περίπτωση γιατί λειτουργεί σαν πρόβλεψη πριν την οριστική μεταβολή των παραμέτρων. Για παράδειγμα, αν η μετατόπιση των παραμέτρων $\boldsymbol{\theta}_t$ κατά $\mu\mathbf{u}_t$ προκαλέσει μία ανεπιθύμητη αύξηση του κόστους τότε η κλίση $\nabla_{\boldsymbol{\theta}}L(f(\boldsymbol{\theta}_t + \mu\mathbf{u}_t)$ θα υποδεικνύει την απαραίτητη διόρθωση.

Ολοκληρώνοντας την περιγραφή του αλγορίθμου SGD και των βελτιώσεων του, πρέπει να τονισθεί η σημασία της τυχαίας δειγματοληψίας παραδειγμάτων από το σύνολο δεδομένων για το σχηματισμό κάθε Minibatch. Για να επιτευχθεί η τυχαία σειρά παρουσίασης των παραδειγμάτων στο δίκτυο, συνηθίζεται να ανακατεύονται τα παραδείγματα πριν από την έναρξη κάθε εποχής εκπαίδευσης. Παρότι αυτή η πρακτική αρχικά στηριζόταν κυρίως από εμπειρικές και πειραματικές ενδείξεις [21], πρόσφατα υπήρξε και ένα θεωρητικό αποτέλεσμα που για την περίπτωση της κυρτής επιφάνειας κόστους αποδεικνύει ότι η ανάδευση των παραδειγμάτων διευκολύνει τη σύγκλιση της διαδικασίας εκπαίδευσης [70].

Μία άλλη σημαντική λεπτομέρεια του αλγορίθμου είναι το *κριτήριο τερματισμού (Termination Criterion)*. Συνήθως σαν κριτήριο τερματισμού τίθεται η ολοκλήρωση ενός μέγιστου αριθμού επαναλήψεων ή υπέρβαση ενός αριθμού συνεχόμενων εποχών για τις οποίες η εκπαίδευση δεν βελτίωσε την ικανότητα γενίκευσης του μοντέλου επί του συνόλου παραδειγμάτων αξιολόγησης. Τέλος, εξίσου σημαντική είναι η επιλογή του προγραμματισμού μεταβολής του ρυθμού μάθησης με την πάροδο των εποχών ή επαναλήψεων εξέλιξης. Η γενική αρχή σχετικά με το ρυθμό μάθησης είναι ότι πρέπει σταδιακά η τιμή του να ελαττώνεται ώστε να εξασφαλιστεί η σταθερή μείωση του κόστους και να αποφευχθούν αστάθειες κατά την εκπαίδευση. Οι βασικές προσεγγίσεις που συνηθίζεται να ακολουθούνται είναι είτε η μείωση του ρυθμού μάθησης ανά προκαθορισμένο αριθμό εποχών ή επαναλήψεων είτε η μείωση του με βάση κάποια μαθηματική σχέση της μορφής $\lambda_t = \lambda_0(1 + \lambda_0 \epsilon t)^{-1}$ [48]. Στα πειράματα του Κεφαλαίου 5, χρησιμοποιείται ο αλγόριθμος SGD με Minibatches και επιτάχυνση Nesterov και η σύνοψη των βημάτων του, παρουσιάζεται στον πίνακα που ακολουθεί:

Algorithm 1 Minibatch Stochastic Gradient Descent with Nesterov's acceleration

Require: Maximum epochs T
Require: Learning rate schedule λ_k for $k = 1...T$
Require: Parameter vector initialization $\boldsymbol{\theta}_0$, Velocity vector initialization $\mathbf{u}_0$
Input: Training Dataset $X = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, ..., \mathbf{x}^{(D)}\}$, Labels $Y = \{y^{(1)}, y^{(2)}, ..., y^{(D)}\}$
Initialization $\boldsymbol{\theta}_1 = \boldsymbol{\theta}_0$, $\mathbf{u}_1 = \mathbf{u}_0$
while Stopping Criterion not met **do**
  Shuffle X, Y (keeping them aligned)
  while not all elements of X have been used **do**
    Sample a Minibatch of D samples $\{\mathbf{x}^{(i)}, ..., \mathbf{x}^{(i+B)}\}$ and their labels $\{\mathbf{y}^{(i)}, ..., \mathbf{y}^{(i+D)}\}$
    Apply intermediate update $\tilde{\boldsymbol{\theta}}_t = \boldsymbol{\theta}_t + \mu \mathbf{u}_t$
    Compute gradient $\mathbf{g} = \frac{1}{B} \sum_i^B \nabla_{\boldsymbol{\theta}} L(f(\mathbf{x}^{(i)}; \tilde{\boldsymbol{\theta}}_t), y^i)$
    Compute Velocity update $\mathbf{u}_{t+1} = \mu \mathbf{u}_t - \lambda_k \mathbf{g}$
    Apply parameter update $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t + \mathbf{u}_{t+1}$
  end while
  $k = k + 1$
end while

4.1.3 Η Συνάρτηση Κόστους

Μέχρι αυτό το σημείο έγινε η περιγραφή του αλγορίθμου S.G.D χωρίς να προσδιοριστεί η ακριβής μορφή της *ανά παράδειγμα* συνάρτησης κόστους $L(\cdot)$. Όπως φάνηκε και στην εξίσωση 4.3, η συνάρτηση κόστους $J(\boldsymbol{\theta})$ μπορεί να εκφραστεί ως ένα άθροισμα των επιμέρους τιμών κόστους που οφείλονται σε κάθε παράδειγμα. Στην παρούσα εργασία επικεντρωνόμαστε στην εκπαίδευση των νευρωνικών μοντέλων για την επίλυση προβλημάτων ταξινόμησης και συνεπώς θα περιγράψουμε τη συνάρτηση κόστους για τέτοιου είδους προβλήματα. Σε προβλήματα ταξινόμησης πολλαπλών κλάσεων (multiclass classification) συνηθίζεται να χρησιμοποιείται

η *συνάρτηση λογαριθμικής κατηγορικής αλληλοεντροπίας (categorical crossentropy)* που αναφέρεται και ως *αρνητική λογαριθμική πιθανοφάνεια (negative logarithmic likelihood)*. Αν το πρόβλημα ταξινόμησης για την επίλυση του οποίου εκπαιδεύεται το μοντέλο, έχει n κλάσεις τότε η αρνητική λογαριθμική πιθανοφάνεια ανά παράδειγμα $(\mathbf{x}^{(i)}, y^{(i)})$ είναι:

$$L(f(\mathbf{x}^{(i)}; \boldsymbol{\theta}), y^{(i)}) = \sum_{j}^{n} \mathbb{1}\{y^{(i)} = j\} \log(p(y^{(i)} = j|\mathbf{x}^{(i)}; \boldsymbol{\theta})) \tag{4.9}$$

όπου $f(\mathbf{x}^{(i)}; \boldsymbol{\theta})$ είναι η έξοδος του νευρωνικού μοντέλου για την είσοδο $\mathbf{x}^{(i)}$. Όπως έχει αναφερθεί στην ενότητα 2.1.1 και συγκεκριμένα στις εξισώσεις 2.9 και 2.10, η έξοδος ενός νευρωνικού μοντέλου, το οποίο επιλύει ένα πρόβλημα ταξινόμησης, προκύπτει μέσω ενός σταδίου *Softmax* μη γραμμικότητας που χρησιμοποιεί την έξοδο του προηγούμενου στρώματος νευρώνων και τη μετατρέπει σε ένα διάνυσμα n στοιχείων, εκ των οποίων κάθε ένα αντιστοιχεί στην *posterior* πιθανότητα κάθε κλάσης δεδομένης της εισόδου $\mathbf{x}^{(i)}$. Συνεπώς, συνδυάζοντας τις εξισώσεις 4.9 και 4.3, προκύπτει η ολοκληρωμένη μορφή της συνάρτησης κόστους που βασίζεται στην κατηγορική αλληλοεντροπία:

$$J(\boldsymbol{\theta}) = -\frac{1}{D} \sum_{i=1}^{D} \sum_{j}^{n} \mathbb{1}\{y^{(i)} = j\} \log(p(y^{(i)} = j|\mathbf{x}^{(i)}; \boldsymbol{\theta})) \tag{4.10}$$

Στην παραπάνω εξίσωση η μεταβλητή D είναι ο αριθμός των παραδειγμάτων εκπαίδευσης. Στην περίπτωση που χρησιμοποιείται ο αλγόριθμος SGD με minibatches που περιγράφηκε στον πίνακα 1, τότε το κόστος υπολογίζεται ανά minibatch. Αν το κάθε minibatch περιλαμβάνει B παραδείγματα και συνολικά μπορούν να προκύψουν $M = \lfloor \frac{D}{B} \rfloor$ minibatches τότε η εξίσωση 4.10 ισοδυναμεί με:

$$J(\boldsymbol{\theta}) = -\frac{1}{M} \sum_{m=1}^{M} \sum_{i=1}^{B} \sum_{j}^{n} \mathbb{1}\{y^{(i,m)} = j\} \log(p(y^{(i,m)} = j|\mathbf{x}^{(i,m)}; \boldsymbol{\theta})) \tag{4.11}$$

αν η μέση τιμή του κόστους των δεδομένων εκπαίδευσης υπολογίζεται ως η μέση τιμή του κόστους των minibatches και $\mathbf{x}^{(i,m)}$ είναι το i-οστό παράδειγμα του m-οστού minibatch.

4.1.4 Τεχνικές Κανονικοποίησης

Στη μηχανική μάθηση και ειδικότερα κατά την εκπαίδευση των μοντέλων, δεν επιδιώκουμε απλά να ελαχιστοποιήσουμε ένα κριτήριο κόστους επί ενός συνόλου δεδομένων, αλλά επιδιώκουμε η μείωση του κόστους αυτού να αυξήσει την ακρίβεια με την οποία το μοντέλο ταξινομεί δεδομένα που δεν υπεισέρχονται στην εκπαίδευση. Ένας τρόπος να βελτιωθεί σημαντικά η απόδοση των μοντέλων είναι να

χρησιμοποιηθεί ένα σύνολο *τεχνικών κανονικοποίησης (regularization techniques)*, θεωρητικά και πειραματικά επιβεβαιωμένης αποτελεσματικότητας. Παρότι υπάρχει μεγάλος αριθμός τεχνικών που μπορούν να εφαρμοστούν, στην παρούσα υποενότητα επικεντρωνόμαστε στις τεχνικές που χρησιμοποιούνται στο πειραματικό μέρος του Κεφαλαίου 5.

Κανονικοποίηση ως προς την L_2 νόρμα των παραμέτρων

Η πιο συχνά χρησιμοποιούμενη τεχνική κανονικοποίησης είναι η κανονικοποίηση των παραμέτρων του μοντέλου ως προς την L_2 νόρμα τους. Ο σκοπός αυτού του είδους κανονικοποίησης είναι η διατήρηση των παραμέτρων ενός μοντέλου κοντά στο 0, αποτρέποντας τη σύγκλιση της διαδικασίας εκπαίδευσης σε μεγάλες τιμές παραμέτρων. Επίσης όπως περιγράφεται αναλυτικά στην αναφορά [69], η L_2 κανονικοποίηση επιτρέπει την κίνηση των παραμέτρων προς τιμές που μειώνουν σημαντικά το κόστος, ενώ αποτρέπει την κίνηση τους προς κατευθύνσεις που δεν μειώνουν σημαντικά το κόστος. Εμμέσως, αυτή η μέθοδος κανονικοποίησης ευνοεί σε κάποιο βαθμό την κίνηση προς περιοχές σχετικά μεγάλης κλίσης του κόστους, όπου εκτελώντας κατάβαση κλίσεων το μοντέλο μαθαίνει γρηγορότερα. Η εφαρμογή της L_2 κανονικοποίησης γίνεται με απλό τρόπο, προσθέτοντας έναν τετραγωνικό όρο στην συνάρτηση κόστους. Συγκεκριμένα:

$$J(\mathbf{\theta})_{L_2} = J(\mathbf{\theta}) + \gamma \sum_{i=1}^{W} \theta_i^2 \tag{4.12}$$

όπου W ο αριθμός παραμέτρων του μοντέλου, δηλαδή ο αριθμός στοιχείων του διανύσματος $\mathbf{\theta}$ και γ μία σταθερά που καθορίζει την ισχύ της επίδρασης του όρου κανονικοποίησης στις τιμές των παραμέτρων.

Batch-normalization

Η τεχνική κανονικοποίησης Batch-Normalization προτάθηκε από τους Ioffe και Szegedy [27]. Το πρόβλημα που επιδιώκει να αντιμετωπίσει είναι η αλλαγή της κατανομής των ενεργοποιήσεων σε ένα νευρωνικό δίκτυο, που προκύπτει κάθε φορά που μεταβάλλονται οι παράμετροι ενός μοντέλου κατά την εκπαίδευση. Έχει φανεί πως αυτή η μεταβολή ενδέχεται να συγκρατεί τις τιμές των ενεργοποιήσεων στην περιοχή κορεσμού με αποτέλεσμα οι κλίσεις να είναι μικρές. Αυτό έχει σαν αποτέλεσμα η εκπαίδευση να καθυστερεί σημαντικά. Για να περιοριστεί το παραπάνω φαινόμενο επιδιώκεται να σταθεροποιηθεί η κατανομή των εισόδων x κάθε στρώματος, δηλαδή πριν εφαρμοστεί η μη γραμμική συνάρτηση ενεργοποίησης. Η εφαρμογή της τεχνικής Batch-Normalization σε ένα στρώμα συνοψίζεται στην πα-

ρακάτω εξίσωση:

$$\hat{x} = \frac{x - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta \tag{4.13}$$

όπου ϵ μία σταθερά που αποτρέπει προβλήματα που ενδέχεται να προκύψουν κατά τον αριθμητικό υπολογισμό της τετραγωνικής ρίζας και

$$\mu = E_{\text{minibatch}}[x] \tag{4.14}$$

$$\sigma = Var_{\text{minibatch}}[x] \tag{4.15}$$

Σημειώνεται πως η μέση τιμή και η διακύμανση υπολογίζονται επί όλων των παραδειγμάτων κάθε minibatch και όχι ανά παράδειγμα. Επίσης δεν υπάρχει περιορισμός ως προς το είδος του στρώματος στο οποίο θα εφαρμοστεί η παραπάνω μεθοδολογία. Στην περίπτωση που έχει ολοκληρωθεί η εκπαίδευση του δικτύου και η είσοδοι του μοντέλου είναι τα παράδειγμα αξιολόγησης οι τιμές των μ και σ υπολογίζονται επί του σύνολου των παραδειγμάτων εκπαίδευσης και διατηρούνται σταθερές σε κάθε υπολογισμό επί του συνόλου αξιολόγησης. Στην πράξη έχει φανεί πως η μέθοδος batch-normalization επιταχύνει την εκπαίδευση διότι επιτρέπει τη χρήση υψηλότερων ρυθμών μάθησης, καθώς οι ενεργοποιήσεις είναι πλέον λιγότερο ”ευαίσθητες” στις μεταβολές των παραμέτρων.

Στοχαστικός μηδενισμός βαρών (Dropout)

Στην αναφορά [30], προτάθηκε σε κάθε επανάληψη της διαδικασίας εκπαίδευσης να εφαρμόζεται ένας στοχαστικός (ή τυχαίος) μηδενισμός ενός ποσοστού των συνδέσεων μεταξύ νευρώνων (δηλαδή θέτοντας την έξοδο του νευρώνα που έπεται στον 0) δύο πλήρως συνδεδεμένων στρωμάτων νευρώνων και ονομάστηκε *dropout*. Η περιγραφή της ακριβούς θεμελίωσης του dropout ξεφεύγει από τους στόχους αυτής της ενότητας και συνεπώς θα αρκεστούμε σε μία ποιοτική ερμηνεία της επίδρασης αυτής της μεθόδου κανονικοποίησης. Όπως έχει φανεί στην πράξη, η μέθοδος αυτή βελτιώνει την ικανότητα γενίκευσης του μοντέλου. Αυτό που επιτυγχάνει το dropout είναι σε κάθε επανάληψη να εκπαιδεύεται ένα τροποποιημένο μοντέλο που αγνοεί την ύπαρξη κάποιων εκ των νευρώνων προηγούμενων ή και επόμενων στρωμάτων. Με αυτό τον τρόπο, σε κάθε επανάληψη διαφορετικές ομάδες των παραμέτρων του δικτύου εκπαιδεύονται απομονωμένες από την επίδραση άλλων παραμέτρων που έχουν μηδενιστεί. Η απομόνωση αυτή έχει σαν αποτέλεσμα οι νευρώνες του δικτύου να ”μαθαίνουν” να επιτελούν μία συγκεκριμένη λειτουργία (δηλαδή να εξάγουν κάποιο χαρακτηριστικό από την είσοδο τους) την οποία δεν υιοθετούν αυτούσια οι νευρώνες τους οποίους αυτοί τροφοδοτούν.

4.2 Ο αλγόριθμος οπίσθιας διάδοσης σφάλματος (Back-Propagation)

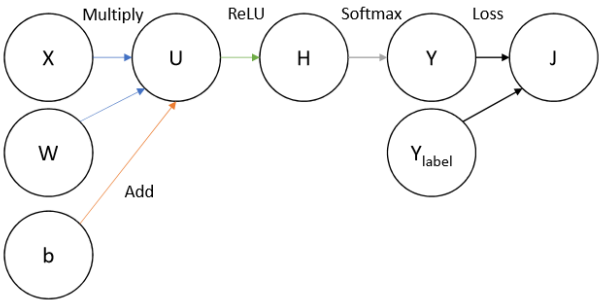
Όταν ένα νευρωνικό μοντέλο δέχεται μία είσοδο $\mathbf{x}$ και παράγει μια πρόβλεψη $\hat{\mathbf{y}}$, τότε η υπολογιστική διαδικασία που επιτελεί ονομάζεται και *εμπρόσθια διάδοση (Forward-Propagation*. Ο αλγόριθμος οπίσθιας διάδοσης σφάλματος ή Back-Propagation χρησιμοποιεί το αποτέλεσμα της εμπρόσθιας διάδοσης ώστε να υπολογιστεί το σφάλμα ή κόστος που οφείλεται στη διαφορά ανάμεσα στην πρόβλεψη $\hat{\mathbf{y}}$ και την πραγματική ή ορθή τιμή που αντιστοιχεί στην είσοδο $\mathbf{y}$. Στη συνέχεια διαδίδει την πληροφορία αυτή σε προηγούμενα στάδια του δικτύου ώστε να υπολογιστεί η κλίση του κόστους ως προς όλες τις παραμέτρους του. Στον υπολογισμό αυτό κεντρικός είναι ο ρόλος του *κανόνα της αλυσίδας (Chain Rule)* της μαθηματική ανάλυσης που θεωρείται πως πρώτος διατύπωσε ο Leibniz το 1676. Συγκεκριμένα ο κανόνας της αλυσίδας δηλώνει πως αν $\mathbf{x} \in \mathbb{R}^m$, $\mathbf{y} \in \mathbb{R}^n$, $z \in \mathbb{R}$, $\mathbf{y} = g(\mathbf{x})$ και $z = f(\mathbf{y})$ τότε ισχύει:

$$\frac{\partial z}{\partial x_i} = \sum_j \frac{\partial z}{\partial y_j} \frac{\partial y_j}{\partial x_i} \tag{4.16}$$

Στα πλαίσια των νευρωνικών δικτύων, η παραπάνω αλληλουχία απεικονίσεων $f(\cdot)$ και $g(\cdot)$ θα μπορούσε να εκφράζει τον υπολογισμό της εξόδου ενός στρώματος νευρώνων perceptron $\mathbf{y}$ (που ονομάζεται και πλήρως συνδεδεμένο στρώμα) και στην συνέχεια τον υπολογισμό του βαθμωτού κόστους z. Η ίδια λογική μπορεί να εφαρμοστεί και για ένα συνελικτικό στρώμα το οποίο έχει δρα πάνω σε τανυστές αντί για διανύσματα. Ο κανόνας της αλυσίδας για την περίπτωση δύο απεικονίσεων $f(\cdot)$ και $g(\cdot)$ που πλέον δρουν σε τανυστές X και Y και παράγουν τη βαθμωτή ποσότητα z είναι:

$$\nabla_X = \sum_j (\nabla_{Xj}) \frac{\partial z}{\partial Y_j} \tag{4.17}$$

Μία ακόμα έννοια, που απαιτείται να εισαχθεί για να κατανοηθεί η εφαρμογή του αλγορίθμου, είναι ο *υπολογιστικός γράφος (computational graph)*. Κάθε νευρωνικό δίκτυο, ανεξαρτήτως του είδους του μπορεί να αναπαρασταθεί ως ένας *κατευθυνόμενος ακυκλικός γράφος (directed acyclic graph)*. Ο γράφος αυτός αποτελείται από κόμβους και ακμές. Κάθε κόμβος, αντιστοιχίζεται σε μία μεταβλητή που υπολογίζεται μέσω κάποιου υπολογισμού (π.χ η έξοδος ή η είσοδος ενός στρώματος νευρώνων). Κάθε ακμή αντιστοιχίζεται στην υπολογιστική διαδικασία που συνδέει τον κόμβο από τον οποίο ξεκινά με τον κόμβο στον οποίο καταλήγει (π.χ μία συνάρτηση ενεργοποίησης ή ένας πολλαπλασιασμός ενός διανύσματος με έναν πίνακα βαρών). Παράδειγμα ενός υπολογιστικού γράφου παρουσιάζεται στο Σχήμα 4.3. Θα βασιστούμε σε αυτό το παράδειγμα ώστε να εξηγήσουμε την διαδικασία εφαρμογής του αλγορίθμου.



Σχήμα 4.3: Αναπαράσταση ενός στρώματος νευρώνων perceptron χρησιμοποιώντας έναν υπολογιστικό γράφο. Οι υπολογιστικές διαδικασίες που χρησιμοποιούνται είναι ο πολλαπλασιασμός πίνακα διανύσματος (multiply), η πρόσθεση διανυσμάτων (add), η συνάρτηση ReLU και ο υπολογισμός του κόστους L. Οι κόμβοι αναπαριστούν την είσοδο X, το διάνυσμα πόλωσης b, ο πίνακας βαρών W, η γραμμική έξοδος H, η έξοδος ενεργοποίησης Y, η επιθυμητή τιμή Y_{label} και η τιμή του κόστους J για την είσοδο .

Το πρώτο βήμα για την εφαρμογή του αλγορίθμου είναι πάντα η κατασκευή του υπολογιστικού γράφου που αντιστοιχεί σε αυτό. Πριν την εφαρμογή του αλγορίθμου Back Propagation, απαιτείται πρώτα ο υπολογισμός της εξόδου του δικτύου, αποθηκεύοντας τις ενδιάμεσες εξόδους των στρωμάτων του δικτύου. Αυτό βήμα ονομάζεται *εμπρόσθια προώθηση ή διάδοση (Forward Propagation)*, διότι υπολογίζουμε τις αποκρίσεις των στρωμάτων του δικτύου, για κάποια δεδομένη είσοδο (ένα παράδειγμα του συνόλου εκπαίδευσης) κινούμενοι από το ρηχότερο προς το βαθύτερο στρώμα. Συγκεκριμένα αν έχουμε ένα γράφο με τελικό κόμβο εξόδου u_n, η τιμή του οποίου είναι εξαρτάται από την τιμή n_i κόμβων εισόδου $u_1, \ldots, u_{n_i}$ του γράφου. Στο παράδειγμα του Σχήματος 4.3, οι κόμβοι εισόδου του γράφου είναι η τιμή του παραδείγματος εκπαίδευσης X, η επιθυμητή τιμή του Y_{label} και οι τιμές των παραμέτρων W και b, ενώ ο τελικός κόμβος είναι το κόστος J. Όπως είναι φανερό, μας ενδιαφέρει να υπολογίσουμε την κλίση της τιμής του κόμβου u_n ως προς τις τιμές των κόμβων $u_1, \ldots, u_{n_i}$, δηλαδή τις τιμές $\frac{\partial u_n}{\partial u_i}$ για $i \in \{1, \ldots, n_i\}$. Στο παράδειγμα του Σχήματος 4.3, μας ενδιαφέρει μόνο να βρούμε την κλίση του κόστους L ως προς τις τιμές των παραμέτρων W και b δηλαδή $\frac{\partial J}{\partial W}$ και $\frac{\partial J}{\partial b}$. Η τιμή κάθε κόμβου u_j του γράφου, υπολογίζεται μεσώ μίας διαδικασίας f_i η οποία λαμβάνει σαν ορίσματα τις τιμές όλων των προγόνων του u_i. Η εφαρμογή του βήματος εμπρόσθιας διάδοσης συνίσταται στην εφαρμογή όλων των διαδικασιών f_i ξεκινώντας από τους κόμβους εισόδου και κινούμενοι προς τον κόμβο εξόδου. Έτσι όλοι οι κόμβοι του γράφου έχουν λάβει και αποθηκεύσει κάποια τιμή για τις δεδομένες τιμές των κόμβων εισόδου. Έχοντας εκτελέσει αυτό το βήμα, είμαστε έτοιμοι να εφαρμόσουμε τον αλγόριθμο Back Propagation. Αν στον πίνακα $g[j]$ αποθηκεύονται οι κλίσεις $\frac{\partial u_n}{\partial u_j}$, τα βήματα του αλγορίθμου Back Propagation είναι τα εξής:

1. Υπολογίζουμε την κλίση του κόμβου u_n ως προς τον εαυτό του που πάντα δίνει μονάδα και την αποθηκεύουμε στον πίνακα g

2. Για κάθε κόμβο u_j ξεκινώντας από τον u_{n-1} και κινούμενοι προς τον u_1 υπολογίζουμε τις τιμές $\frac{\partial u_n}{\partial u_j} = \sum_{i:j \in ancestors(u_i)} \frac{\partial u_n}{\partial u_i} \frac{\partial u_i}{\partial u_j}$ και τις αποθηκεύουμε στον πίνακα g

3. Επιστρέφουμε τις τιμές του πίνακα $g[u_i]$ για $i \in \{1, \ldots, n_i\}$, δηλαδή τις κλίσεις του τελικού κόμβου ως προς τους κόμβους εισόδου.

Επιστρέφοντας στο παράδειγμα του Σχήματος 4.3, η εφαρμογή των παραπάνω βημάτων θα ήταν η εξής:
Για το βήμα Forward Propagation:

$$U^{(f)} = W^{(c)} X^{(c)} + b^{(c)} \tag{4.18}$$

$$H^{(f)} = max(0, U^{(f)}) \tag{4.19}$$

$$Y^{(f)} = Softmax(H^{(f)}) \tag{4.20}$$

$$J = L(Y^{(f)}, Y^{(c)}_{label}) \tag{4.21}$$

όπου με εκθέτη (f) συμβολίζονται η αριθμητικές τιμές που υπολογίστηκαν για το παράδειγμα $^{(c)}$, $label^{(c)}$ δεδομένων των τιμών $W^{(c)}$ και $b^{(c)}$ των παραμέτρων W και b. Με τον εκθέτη (c) υπονοείται το *τρέχων (current)* παράδειγμα και η τρέχουσα τιμή των παραμέτρων. Ο προσδιορισμός αυτός απαιτείται όταν χρησιμοποιείται ο επαναληπτικός αλγόριθμος S.G.D, κάθε επανάληψη του οποίου αλλάζει τις τιμές των παραμέτρων αλλά και του παραδείγματος εισόδου. Για την εφαρμογή του αλγορίθμου Back Propagation:

$$\frac{\partial J}{\partial W} = \frac{\partial J}{\partial Y} \Big|_{Y^{(f)}} \frac{\partial Y}{\partial H} \Big|_{H^{(f)}} \frac{\partial H}{\partial U} \Big|_{U^{(f)}} \frac{\partial U}{\partial W} \Big|_{W^{(c)}} \tag{4.22}$$

$$\frac{\partial J}{\partial b} = \frac{\partial J}{\partial Y} \Big|_{Y^{(f)}} \frac{\partial Y}{\partial H} \Big|_{H^{(f)}} \frac{\partial H}{\partial U} \Big|_{U^{(f)}} \frac{\partial U}{\partial b} \Big|_{b^{(c)}} \tag{4.23}$$

Εφόσον οι συναρτήσεις που χρησιμοποιούνται στις εξισώσεις 4.18 είναι παραγωγίσιμες τότε η παραπάνω εκφράσεις δίνουν την αριθμητική τιμή της κλίσης του κόστους ως προς τις παραμέτρους. Αν η είσοδος προέκυπτε από κάποιον άλλο υπολογιστικό γράφο και δεν αποτελούσε απλά το παράδειγμα εκπαίδευσης τότε η παραπάνω διαδικασία μπορούσε να συνεχιστεί και για τον γράφο αυτό, ώστε να υπολογιστούν οι κλίσεις του κόστους ως προς όσες επιπλέον παραμέτρους έχει ο γράφος αυτός. Για παράδειγμα για την περίπτωση ενός M.L.P η εφαρμογή του αλγορίθμου Back Propagation παρουσιάζεται στον πίνακα αλγορίθμου 2. Σημειώνεται πως εφόσον οι υπολογισμοί κάθε συνελικτικού στρώματος ενός συνελικτικού δικτύου μπορούν να εκτελεσθούν ως πολλαπλασιασμοί πινάκων, μπορεί να εξαχθεί μία ισοδύναμη περιγραφή με αυτή του πίνακα αλγορίθμου 2 και για τα συνελικτικά δίκτυα.

Algorithm 2 Feed Forward Network Back Propagation

Require: predicted output $\hat{y}$, ground truth y

1: $\mathbf{g} \leftarrow \nabla_{\hat{y}} J = \nabla_{\hat{y}} L(\hat{y}, y)$
2: **for** $k = l, l-1, \ldots, 1$ **do**
3: Compute the gradient w.r.t the layer's pre-nonlinear activation output with $a^{(k)}$ as the activations and ϕ as the nonlinearity of the layer which is applied element wise to its input $\mathbf{g} \leftarrow \nabla_{\mathbf{a}^{(k)}} J = \mathbf{g} \odot \phi'(\mathbf{a}^{(k)})$
4: Compute the gradient w.r.t the layer's parameters:
5: $\nabla_{\mathbf{b}^{(k)}} J = \mathbf{g}$
6: $\nabla_{\mathbf{W}^{(k)}} J = \mathbf{g}\mathbf{h}^{(k-1)T}$
7: Propagate the gradients w.r.t the previous layer's activations:
8: $\mathbf{g} \leftarrow \nabla_{\mathbf{h}^{(k-1)}} J = \mathbf{W}^{(k)T}\mathbf{g}$
9: **end for**

4.3 Ο αλγόριθμος οπίσθιας διάδοσης σφάλματος στο πεδίο του χρόνου (Back-Propagation through time)

Ο αλγόριθμος Back Propagation μπορεί να γενικευτεί και για την περίπτωση των αναδρομικών μοντέλων των οποίων η είσοδος είναι κάθε φορά ένα μέρος μίας χρονικά εξελισσόμενης ακολουθίας διανυσμάτων. Μέσω του αλγορίθμου *οπίσθιας διάδοσης σφάλματος στο πεδίο του χρόνου (Back Propagation through time)* υπολογίζεται η κλίση του κόστους μίας ακολουθίας εισόδου $\{X_1, X_2, \ldots, X_t, \ldots, X_T\}$ ως προς τις παραμέτρους W, U και b. Ένα βασικό πρώτο βήμα για να διευκολυνθεί η εφαρμογή του αλγορίθμου είναι να "ξεδιπλωθεί" στον άξονα του χρόνου το δίκτυο. Ένα παράδειγμα αυτής της διαδικασίας φαίνεται στο Σχήμα 4.4. Υπενθυμίζεται πως ένα αναδρομικό δίκτυο περιγράφεται από τις παρακάτω εξισώσεις:

$$h_t = \phi(WX_t + Uh_{t-1}) \tag{4.24}$$

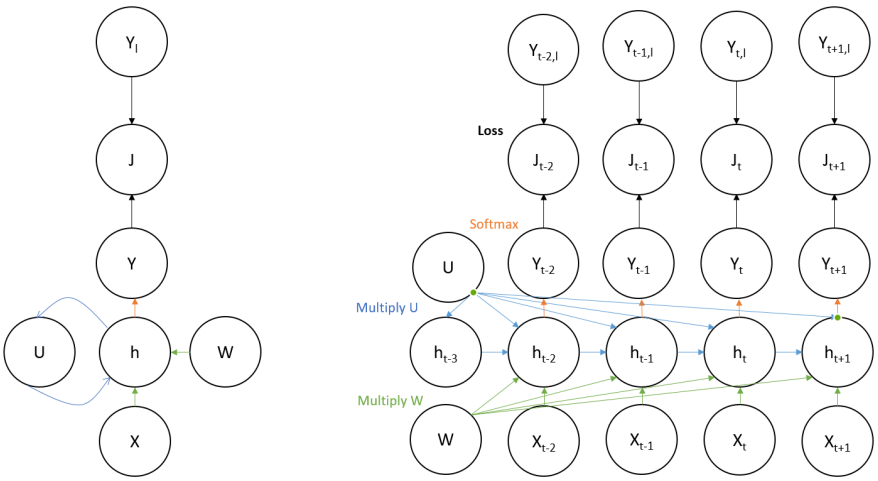
Η έξοδος του αναδρομικού δικτύου h_t τροφοδοτείται σε μία Softmax μη γραμμική συνάρτηση και στη συνέχεια υπολογίζεται το κόστος σε σχέση με την επιθυμητή τιμή εξόδου. Η λειτουργία της συνάρτησης Softmax είναι η εξής:

$$\hat{y} = \text{Softmax}(h_t) \tag{4.25}$$

Σημειώνεται, πως όπως έχει προαναφερθεί όλοι οι επιπλέον κόμβοι του γράφου, που προέκυψαν κατά το "ξεδίπλωμα" είναι βοηθητικοί, και στην πράξη στο τέλος οι συνεισφορά κάθε χρονικού βήματος στο κόστος και στην κλίση αθροίζεται χρονικής στιγμής. Συνεπώς το συνολικό κόστος για την ακολουθία εισόδου είναι:

$$J(y, \hat{y}) = \sum_{t=1}^{T} J_t(y_t, \hat{y}_t) \tag{4.26}$$

Εφόσον, το κόστος υπολογίζεται για όλη την ακολουθία εισόδου, κλίση του κόστους ως προς τις παραμέτρους του μοντέλου θα είναι το άθροισμα των επιμέρους κλίσεων



Σχήμα 4.4: "Ξεδίπλωμα" και υπολογιστικός γράφος ενός αναδρομικού δικτύου (Αριστερά). Υπολογιστικός γράφος ενός μη ξεδιπλωμένου υπολογιστικού γράφου (Δεξιά). Σημειώνεται πως οι διαδικασίες στις οποίες αντιστοιχεί κάθε ακμή είναι κωδικοποιημένες χρωματικά και είναι οι εξής: πολλαπλασιασμός της εισόδου με τον πίνακα W, πολλαπλασιασμός της εισόδου με τον πίνακα U, η Softmax συνάρτηση και η συνάρτηση υπολογισμού του κόστους. Στο "ξεδιπλωμένο" δίκτυο δεν εμφανίζονται (για λόγους απλότητας) όλα τα χρονικά βήματα της ακολουθίας αλλά ένα στιγμιότυπο της. Επίσης η επιθυμητή τιμή εξόδου στο σχήμα συμβολίζεται με $Y_{t,l}$.

κάθε χρονικής στιγμής t της ακολουθίας εισόδου:

$$\frac{\partial J}{\partial W} = \sum_{t=1}^{T} \frac{\partial J_t}{\partial W} \tag{4.27}$$

$$\frac{\partial J}{\partial U} = \sum_{t=1}^{T} \frac{\partial J_t}{\partial U} \tag{4.28}$$

Άρα πρέπει να βρεθεί το κόστος για κάθε χρονικό στιγμή. Αντιμετωπίζοντας τον "ξεδιπλωμένο" υπολογιστικό γράφο σαν έναν οποιοδήποτε γράφο :

$$\frac{\partial J_t}{\partial W} = \frac{\partial J_t}{\partial \hat{y}_t} \frac{\partial \hat{y}_t}{\partial h_t} \frac{\partial h_t}{\partial W} \tag{4.29}$$

$$\frac{\partial J_t}{\partial U} = \sum_{k=1}^{t} \frac{\partial J_t}{\partial \hat{y}_t} \frac{\partial \hat{y}_t}{\partial h_t} \frac{\partial h_t}{\partial h_k} \frac{\partial h_k}{\partial U} \tag{4.30}$$

Όπως και για την απλή εκδοχή του αλγορίθμου Back Propagation αν όλες οι παράγωγοι της σχέσης 4.29 μπορούν να υπολογιστούν τότε δεδομένων των τιμών των κόμβων που έχουν υπολογιστεί κατά την εμπρόσθιας διάδοση, υπολογίζεται η αριθμητική τιμή της κλίσης του κόστους ως προς τις παραμέτρους του μοντέλου. Μία προέκταση που πρέπει να αναφερθεί, είναι πως στην περίπτωση που η είσοδος $_t$ παράγεται από κάποιον άλλο υπολογιστικό γράφο, όπως ένα MLP ή ένα συνελικτικό δίκτυο ή ένα άλλο αναδρομικό δίκτυο, μπορούμε να συνεχίσουμε την οπίσθια διάδοση του σφάλματος μέχρι τους κόμβους των παραμέτρων και αυτού του γράφου. Για παράδειγμα όπως θα δούμε στο Κεφάλαιο που ακολουθεί, ένα σύνθετο

νευρωνικό μοντέλο όπως το CNN-LSTM λειτουργεί παράγει εξάγει ένα διάνυσμα χαρακτηριστικών$_t$ από το καρέ ενός βίντεο την χρονική στιγμή t το οποίο τροφοδοτεί στο αναδρομικό δίκτυο LSTM. Όταν εξαχθούν διανύσματα χαρακτηριστικών για όλα τα καρέ ενός βίντεο παράγεται μία ακολουθία εισόδου για το αναδρομικό μέρος του μοντέλου το οποίο και παράγει μία έξοδο για κάθε στοιχείο της ακολουθίας. Στη συνέχεια αυτές οι έξοδοι μπορούν να χρησιμοποιηθούν για να γίνει πρόβλεψη σχετικά με την κλάση στην οποία ανήκει το βίντεο. Το σύνθετο αυτό μοντέλο εκπαιδεύεται μέσω του αλγορίθμου Back Propagation στον χρόνο και οι παράμετροι τόσο του αναδρομικού όσο και του συνελικτικού μέρους του, ανανεώνονται εφαρμόζοντας τον αλγόριθμο για κάθε ακολουθία εισόδου.

Κεφάλαιο 5

Εκπαίδευση μοντέλων και πειραματικά αποτελέσματα offline αναγνώρισης

Στο αυτό το κεφάλαιο παρουσιάζονται πειραματικά αποτελέσματα ταξινόμησης ανθρωπίνων δράσεων και χειρονομιών. Ο σκοπός των πειραμάτων είναι η διερεύνηση της αποτελεσματικότητας των συνελικτικών και αναδρομικών δικτύων σε τέτοια προβλήματα. Παράλληλα, παρουσιάζεται λεπτομερώς ο τρόπος με τον οποίο εκπαιδεύονται end-to-end τα μοντέλα. Στα πλαίσια των πειραμάτων επιδιώκεται να συγκριθεί η βελτίωση που επιφέρει η ενσωμάτωση συνολικής χρονικής μοντελοποίησης (global temporal modelling) μέσω της προσθήκης του στρώματος LSTM νευρώνων σαν προέκταση του συνελικτικού δικτύου. Επίσης συγκρίνεται η αποδοτικότητα των χωροχρονικών χαρακτηριστικών που εξάγει ένα 3D-CNN με αυτή των χωρικών χαρακτηριστικών που εξάγονται από ένα 2D-CNN. Τέλος στις περιπτώσεις που είναι διαθέσιμες διαφορετικές τροπικότητες (modalities), παρουσιάζεται σύμμειξη αυτών (multimodal fusion) στο τελικό στάδιο της πρόβλεψης. Συνολικά εκπαιδεύονται και αξιολογούνται τρία είδη μοντέλων:

- Νευρωνικά δίκτυα τρισδιάστατης συνέλιξης για εξαγωγή τοπικών χωροχρονικών χαρακτηριστικών που τροφοδοτούνται σε έναν απλό softmax ταξινομητή (3DCNN και C3D).
- Αναδρομικά νευρωνικά δίκτυα αποτελούμενα από ένα στρώμα νευρώνων LSTM που δέχονται είσοδο από ένα νευρωνικό δίκτυο τρισδιάστατης συνέλιξης (3DCNN-LSTM)
- Αναδρομικά νευρωνικά δίκτυα αποτελούμενα από ένα στρώμα νευρώνων LSTM που δέχονται είσοδο από ένα νευρωνικό δίκτυο δισδιάστατης συνέλιξης (3DCNN-LSTM)

5.1 Πειραματικό πλαίσιο

Στη ενότητα αυτή περιγράφεται ο τρόπος επιλογής των υπερπαραμέτρων εκπαίδευσης των μοντέλων και η μεθοδολογία βελτίωσης της απόδοσης. Οι παρακάτω

υποενότητες περιγράφουν σημαντικά σημεία της εκπαίδευσης των μοντέλων και μεθοδολογίες που εφαρμόζονται σε όλα τα πειράματα που ακολουθούν. Ο ορισμός των μοντέλων, η εκπαίδευση και η αξιολόγησή τους υλοποιήθηκαν χρησιμοποιώντας τη βιβλιοθήκη *Lasagne* [13] που στηρίζεται στο προγραμματιστικό περιβάλλον *Theano* [12]. Επίσης χρησιμοποιήθηκε η βιβλιοθήκη της nvidia *cuDnn* που επιταχύνει τους υπολογισμούς που αφορούν στα νευρωνικά μοντέλα, χρησιμοποιώντας την GPU του υπολογιστή.

5.1.1 Υπερπαράμετροι της διαδικασίας εκπαίδευσης

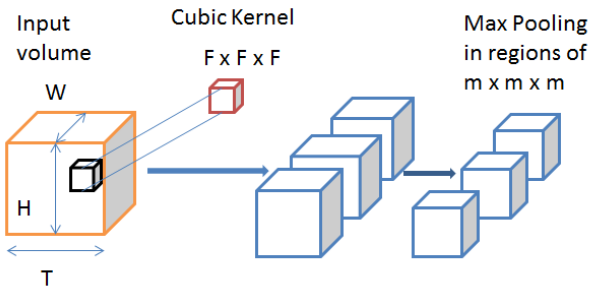
Η εκπαίδευση όλων των μοντέλων που περιγράφονται στο παρόν κεφαλαίο έγινε με χρήση του αλγορίθμου stochastic gradient descent με επιτάχυνση Nesterov. Η επιλογή αυτή βασίστηκε στο γεγονός πως η συντριπτική πλειοψηφία των εργασιών της σύγχρονης βιβλιογραφίας αναφέρει πως ο συγκεκριμένος αλγόριθμος δείχνει να είναι ο πιο αποτελεσματικός για την εκπαίδευση συνεκτικών δικτύων και αναδρομικών δικτύων [27, 3, 36, 35]. Με βάση αυτή την επιλογή, οι υπερπαράμετροι που πρέπει να καθοριστούν για την εκπαίδευση ενός μοντέλου είναι ο ρυθμός μάθησης (learning rate), το μέγεθος του υποσυνόλου των δεδομένων εκπαίδευσης που χρησιμοποιείται για τον υπολογισμό της κλίσης του κόστους ως προς τα βάρη του μοντέλου (Minibatch size) και η σταθερά κανονικοποίησης των βαρών. Τέλος, ιδιαίτερα σημαντικός παράγοντας, που επιδρά στην ταχύτητα σύγκλισης της εκπαίδευσης, είναι ο προγραμματισμός αλλαγής του ρυθμού μάθησης με την πάροδο των εποχών εκπαίδευσης. Τα εναλλακτικά είδη προγραμματισμού που δοκιμάστηκαν ήταν η προσχεδιασμένη βηματική μείωση του ρυθμού μάθησης ανά προκαθορισμένο αριθμό εποχών και η μείωση ανά εποχή με βάση κάποια μαθηματική σχέση. Κατά περίπτωση αναφέρεται ποιο είδος προγραμματισμού ακολουθείται. Στην πράξη φάνηκε πως για ταχύτερη σύγκλιση της διαδικασίας η παρακάτω μαθηματική σχέση έδωσε τα καλύτερα αποτελέσματα:

$$\lambda_i = \frac{\lambda_{i-1}}{\alpha + \beta \times i} \tag{5.1}$$

όπου *i* ο αριθμός της εποχής εκπαίδευσης και α, β δύο συντελεστές που ελέγχουν την ταχύτητα μείωσης του ρυθμού μάθησης. Αντίθετα η ”Hard-Coded” μείωση του ρυθμού ανά κάποιο αριθμό εποχών δεν οδήγησε σε καλά αποτελέσματα στην εκπαίδευση των αναδρομικών μοντέλων. Η ανάγκη για σταδιακή μείωση του ρυθμού μάθησης οφείλεται στο γεγονός ότι όσο εξελίσσεται η εκπαίδευση και οι αλλαγές των βαρών του μοντέλου πρέπει να γίνονται όλο και πιο μικρές για να αποφευχθούν ταλαντώσεις και αστάθειες της διαδικασίας εκπαίδευσης γύρω από τοπικά ελάχιστα της συνάρτησης κόστους. Επίσης πολλές φορές, όπως διαπιστώθηκε ακόμα και στα αρχικά στάδια της εκπαίδευσης, σταθερές τιμές ρυθμού μάθησης μπορεί να ”εγκλωβίσουν” το κόστος και (συνεπώς τις παραμέτρους) σε τοπικά ελάχιστα που οδηγούν σε χαμηλή ικανότητα γενίκευσης.

5.1.2 Γενικά χαρακτηριστικά Συνελικτικού Νευρωνικού Δικτύου Τρισδιάστατης Συνέλιξης

Είναι χρήσιμο να περιγράψουμε τα γενικά χαρακτηριστικά των δικτύων τρισδιάστατης συνέλιξης. Ένα Συνελικτικό Δίκτυο Τρισδιάστατης Συνέλιξης δέχεται σαν είσοδο έναν τανυστή 3 διαστάσεων $T \times h \times w$, που στην περίπτωση της χρήσης βίντεο, αποτελείται από καρέ και κάθε καρέ έχει χωρικό πλάτος w και χωρικό ύψος h. Οι πυρήνες ή φίλτρα (kernels ή filters) που χρησιμοποιούνται στα στρώματα συνέλιξης είναι κυβικοί διάστασης $f \times f \times f$. Η συνέλιξη του τανυστή εισόδου με κάθε έναν πυρήνα δίνει έναν νέο τανυστή ίδιας διάστασης. Κάθε επίπεδο συνέλιξης ακολουθείται από μία μη γραμμική συνάρτηση ενεργοποίησης ReLU (Rectified Linear Unit) που εφαρμόζεται σε κάθε τιμή του τανυστή που προκύπτει. Στη συνέχεια υποδειγματοληπτείται ο τανυστής εξόδου μέσω max pooling σε υποπεριοχές αυτού είτε κυβικής μορφής $m \times m \times m$ είτε με διαφορετική διάσταση στον χρονικό άξονα και τετραγωνικής μορφής ως προς τις χωρικές διαστάσεις, $k \times m \times$ m. Οι παραπάνω διαδικασίες φαίνονται και στο Σχήμα 5.1. Στο εξής όταν θα αναφέρεται η διάσταση τανυστή ή φίλτρου θα τηρείται ο συμβολισμός χρονική διάσταση $\times$ ύψος$\times$ πλάτος. Η χρήση μόνο κυβικού πυρήνα επιλέχθηκε για να απλοποιηθεί η διαδικασία εξεύρεσης του μοντέλου με την υψηλότερη επίδοση.



Σχήμα 5.1: Η διαδικασία 3Δ συνέλιξης και max pooling. Για λόγους απλοποίησης της παρουσίασης υποθέτουμε πως έχουμε 3 διαφορετικούς πυρήνες και κατά συνέπεια μετά την συνέλιξη προκύπτουν 3 διαφορετικοί τρισδιάστατοι χάρτες χαρακτηριστικών οι οποίοι υποδειγματολειπτούνται μέσω max pooling.

5.1.3 Γενικά χαρακτηριστικά Νευρώνων Μακράς-Βραχείας Μνήμης (LSTM)

Ένα στρώμα νευρώνων LSTM ορίζεται μέσω των εξισώσεων κατάστασης που έχουν περιγραφεί στο Κεφάλαιο 3. Οι παράμετροι που απαιτούνται για να οριστεί ένα στρώμα LSTM νευρώνων είναι ο αριθμός των νευρώνων και το είδος των μη γραμμικών συναρτήσεων ενεργοποίησης των διανυσμάτων κατάστασης του νευρώνα. Οι μη γραμμικές συναρτήσεις που επιλέχθηκαν ήταν η σιγμοειδής συνάρτησης και η υπερβολικής εφαπτομένη όπως ακριβώς φαίνεται στις εξισώσεις 3.3. Επίσης κατά την εκπαίδευση όλων των μοντέλων, επιλέχθηκε να περιορίζεται η τιμή της παραγώγου ως προς τις εισόδους του στρώματος LSTM x_t όπως αυτή εμφανίζε-

ται στο σχήμα 3.2 (gradient clipping), στο διάστημα $[-100, 100]$ όπως προτείνεται στα [35, 54, 56, 64]. Με τον τρόπο αυτό αποφεύγονται αστάθειες κατά τον αριθμητικό υπολογισμό της τιμής κλίσης.

5.1.4 Τεχνικές κανονικοποίησης κατά την εκπαίδευση

Κατά την εκπαίδευση όλων των μοντέλων χρησιμοποιήθηκαν τρεις τεχνικές κανονικοποίησης:

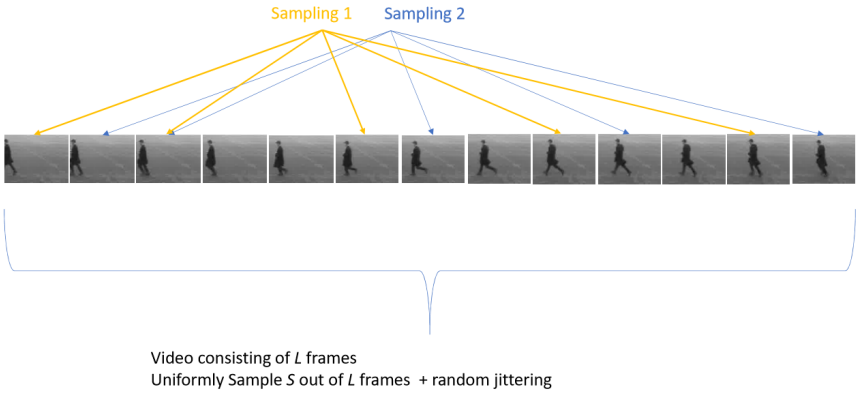
- Κανονικοποίηση (ή απόσβεση) των βαρών ως προς την L_2 νόρμα τους (weight decay), που επιτυγχάνεται μέσω της προσθήκης κατάλληλου όρου στη συνάρτηση κόστους της εξίσωσης 5.3. Σκοπός της κανονικοποίησης αυτής είναι να αποφευχθεί το φαινόμενο του overfitting στα δεδομένα εκπαίδευσης.
- Batch normalization, Τεχνική που εφαρμόζεται κατά την εκπαίδευση μετά από κάθε συνελικτικό στρώμα νευρώνων (convolutional layer) με βάση την οποία η κάθε είσοδος σε αυτό διαιρείται με τη διακύμανση και μειώνεται κατά τη μέση τιμή, του υποσυνόλου (batch) των παραδειγμάτων που χρησιμοποιούν για την κάθε επανάληψη του αλγορίθμου εκπαίδευσης. Η συγκεκριμένη τεχνική εφαρμόζεται σε όλα τα μοντέλα *μετά από όλα τα συνελικτικά στρώματα* εκτός αν αναφέρεται διαφορετικά
- Dropout, Τεχνική που εφαρμόζεται μεταξύ δύο πλήρως συνδεδεμένων στρωμάτων, με βάση την οποία για κάθε επανάληψη του αλγορίθμου εκπαίδευσης επιλέγονται τυχαία κάποιες συνδέσεις νευρώνων οι οποίες μηδενίζονται. Η τεχνική dropout εφαρμόζεται σε όλα τα πλήρως συνδεδεμένα στρώματα νευρώνων, όλων των μοντέλων. Για τον προσδιορισμό του ποσοστού των συνδέσεων που μηδενίζονται χρησιμοποιείται μία υπερπαράμετρος p που εκφράζει το ποσοστό αυτό. Στα πειράματα που ακολουθούν χρησιμοποιήθηκε $p \in [0.5, 0.75]$.

5.1.5 Προ-επεξεργασία δεδομένων

Λόγω της διαφοράς στη διάρκεια των βίντεο κρίθηκε αναγκαίο να χρησιμοποιείται σταθερός αριθμός καρέ ανά βίντεο των δεδομένων εκπαίδευσης. Αυτή η επιλογή ισοσταθμίζει τον αριθμό clip και καρέ ανά κλάση, με αποτέλεσμα να αποφεύγεται η "πολωμένη" εκπαίδευση του μοντέλου λόγω της ανισσοροπίας των δεδομένων ανά κλάση. Για αυτό το λόγο, επιλέχθηκε τα βίντεο να υποδειγματολειπτούνται ή να προεκτείνονται ώστε να έχουν προκαθορισμένη σταθερή διάρκεια L καρέ. Πιο συγκεκριμένα, η ακολουθία των καρέ ενός βίντεο δειγματοληπτήθηκε ομοιόμορφα. Αν υποθέσουμε πως ένα βίντεο αποτελείται από S καρέ, η παραπάνω διαδικασία περιγράφεται από την παρακάτω σχέση:

$$index_i = \lfloor (\frac{S}{L} \times i + jitter) \rfloor \text{ για } i = 1...L \tag{5.2}$$

όπου $index_i$ είναι το i-οστό καρέ που δειγματοληπτείται από το βίντεο και $jitter$ μία τυχαία παραγόμενη τιμή στο διάστημα $[-3, 3]$ που αποσκοπεί στην επαύξηση του συνόλου δεδομένων αφού με τον τρόπο αυτό δεν θα χρησιμοποιούνται σε κάθε εποχή εκπαίδευσης ακριβώς τα ίδια καρέ από κάθε βίντεο.



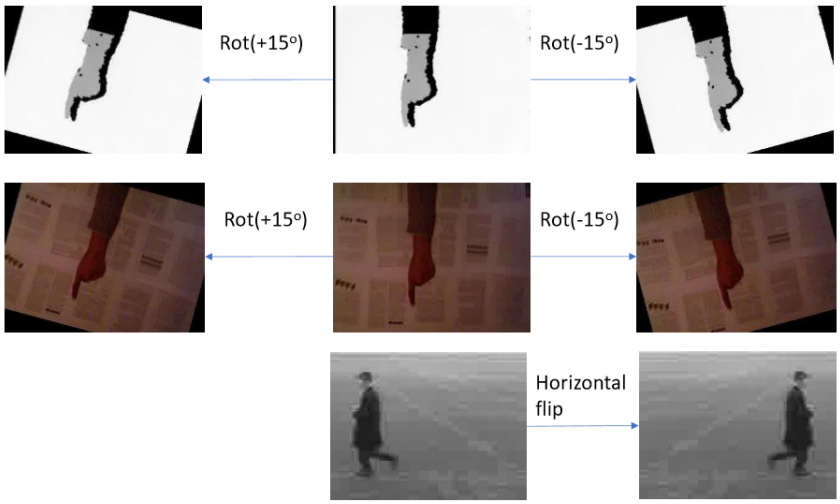
Σχήμα 5.2: Η διαδικασία ομοιόμορφης δειγματοληψίας καρέ από ένα βίντεο. Κατά την εκπαίδευση, επαναδειγματοληπτείται κάθε βίντεο κάθε φορά που χρησιμοποιείται σαν παράδειγμα εκπαίδευσης. Η παράμετρος jitter της εξίσωσης 5.2 μεταβάλλει με τυχαίο τρόπο τα καρέ που επιλέγονται από το κάθε παράδειγμα, σε κάθε εποχή εκπαίδευσης. Συνεπώς για το ίδιο παράδειγμα εκπαίδευσης σε διαφορετικές εποχές εκπαίδευσης το δίκτυο δεν θα δεχθεί σαν είσοδο τα ίδια ακριβώς καρέ. Παρατηρήστε πως τα καρέ που επιλέγονται στην επιλογή Sampling1 και Sampling2 δεν είναι τα ίδια.

Ο διαχωρισμός του συνόλου δεδομένων έγινε σε δεδομένα εκπαίδευσης και αξιολόγησης. Στη βάση KTH χρησιμοποιήθηκαν 1507 παραδείγματα δράσεων για εκπαίδευση και 884 για αξιολόγηση. Αντίστοιχα, στη βάση SKIG χρησιμοποιήθηκαν 720 παραδείγματα χειρονομιών για εκπαίδευση και 360 για αξιολόγηση. Αξίζει να σημειωθεί πως τα άτομα που συμμετέχουν στην εκτέλεση δράσεων και χειρονομιών στα σύνολα δεδομένων εκπαίδευσης δεν είναι τα ίδια με αυτά των συνόλων αξιολόγησης. Συνεπώς, τα μοντέλα -όπως θα φανεί και κατά την παρουσίαση των αποτελεσμάτων μαθαίνουν- (σε κάποιο βαθμό) να γενικεύουν τη γνώση που αποκτούν σε σχέση με τη δράση ή τη χειρονομία, ανεξάρτητα με το άτομο που τις εκτελεί. Τέλος, όσων αφορά στην ανάλυση των καρέ, για τα βίντεο και των δύο βάσεων υποδειγματοληπτούμε χωρικά κάθε καρέ από την αρχική του ανάλυση σε ανάλυση (64×64) pixel.

5.1.6 Επαύξηση δεδομένων (Data Augmentation)

Έχει φανεί εμπειρικά [1, 3, 4] πως τα αποτελέσματα εκπαίδευσης Συνελικτικών Νευρωνικών Δικτύων βελτιώνονται αν αυξηθεί ”τεχνητά” ο αριθμός των παραδειγμάτων, εφαρμόζοντας χωρικούς ή χρονικούς μετασχηματισμούς στα υπάρχοντα βίντεο για να παραχθούν νέα δεδομένα εκπαίδευσης. Η διαδικασία αυτή ονομάζεται *επαύξηση δεδομένων (data augmentation)*. Στα πλαίσια των πειραμάτων που εκτελέσθηκαν, οι μέθοδοι επαύξησης δεδομένων που χρησιμοποιήθηκαν ήταν offline

(δηλαδή πριν την αρχή της εκπαίδευσης) χωρικοί μετασχηματισμοί και online (δηλαδή πριν από την χρήση κάθε παραδείγματος στην εκπαίδευση σε κάθε εποχή) χρονική διαταραχή (temporal jittering). Συγκεκριμένα, για τα δεδομένα της βάσης KTH χρησιμοποιήθηκαν οι χωρικοί μετασχηματισμοί οριζόντιου αντικατοπτρισμού (*horizontal flip*) και περιστροφής ± 15 ή ± 35 (rotation) για τα καρέ κάθε βίντεο των δεδομένων εκπαίδευσης. Η χρονική επαύξηση επιτυγχάνεται με την εισαγωγή μίας τυχαία παραγόμενης παραμέτρου jitter κατά τη δειγματοληψία καρέ από ένα βίντεο όπως αυτή περιγράφεται από την εξίσωση 5.2. Για τα δεδομένα της βάσης Skig, χρησιμοποιήθηκαν offline περιστροφές ± 15 και online temporal jittering. Για την βάση KTH χρησιμοποιήθηκαν offline περιστροφές ± 35, οριζόντιος κατοπτρισμός και online temporal jittering. Μετά την offline επαύξηση δεδομένων τα παραδείγματα εκπαίδευσης για τη βάση KTH ήταν 3724 βίντεο και για τη βάση SKIG ήταν 2160 βίντεο για κάθε τροπικότητα (RGB και Depth). Αντίστοιχα τα σύνολα αξιολόγησης (test sets) αποτελούνταν από 365 και 881 βίντεο αντίστοιχα.
Ακολουθεί η παρουσίαση των μοντέλων και του τρόπου εκπαίδευσης και αξιολόγησής τους.



Σχήμα 5.3: Επαύξηση δεδομένων μέσω χωρικών μετασχηματισμών.

5.2 Τα μοντέλα 3DCNN και C3D

Τα δύο παρακάτω μοντέλα εξάγουν χωρο-χρονικά χαρακτηριστικά από κάθε απομονωμένο clip ενός τα οποία τροφοδοτούνται σε έναν απλό Softmax ταξινομητή. Κατά συνέπεια, μέσω αυτών των μοντέλων δεν μοντελοποιούνται χρονικές εξαρτήσεις μακράς διάρκειας αλλά μόνο βραχείας διάρκειας. Παρότι ο τελικός σκοπός είναι να χρησιμοποιηθούν μοντέλα που ενσωματώνουν χωροχρονική πληροφορία τόσο μακράς όσο και βραχείας διάρκειας, τα αποτελέσματα για αυτού του είδους τα

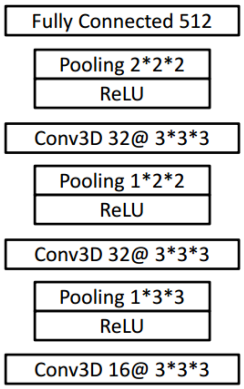
μοντέλα θα μας επιτρέψουν να αξιολογήσουμε την επίδραση της προέκτασης των μοντέλων στη συνέχεια. Ειδικότερα, μέσω των πειραμάτων για τα μοντέλα *3D-CNN* και *C3D*, συγκρίνεται η εκπαίδευση και απόδοση ενός ρηχού συνελικτικού δικτύου τρισδιάστατης συνέλιξης (3D-CNN) με τυχαία αρχικοποιημένες παραμέτρους (αναφέρεται και ως *"trained from scratch"*) και ενός πολύ βαθύτερου και μεγαλύτερου προεκπαιδευμένου δικτύου τρισδιάστατης συνέλιξης (C3D).

Το μοντέλο C3D, είναι ένα βαθύ νευρωνικό δίκτυο με 8 συνελικτικά (convolutional layers), 5 συσσωρευτικά (max pooling) και 2 πλήρως συνδεδεμένα στρώματα νευρώνων (με 4096 νευρώνες το καθένα), του οποίου η αρχιτεκτονική (Σχήμα 5.4) προτάθηκε από τους Tran et al [3] οι οποίοι έδειξαν πως το συγκεκριμένο μοντέλο έχει επιτύχει υψηλά ποσοστά μέσης ακρίβειας (mAP) στην αναγνώριση ανθρώπινης δράσης σε βάσεις δεδομένων μεγάλης κλίμακας (Sports1M και UCF101).Επιπλέον, οι συγγραφείς παρείχαν τις τιμές των τελικών βαρών του μοντέλου όταν αυτό εκπαιδεύτηκε στη βάση Sports1M που περιλαμβάνει $1.133.158$ βίντεο από 487 κλάσεις αθλητικών δραστηριοτήτων. Στα πλαίσια των πειραμάτων, χρησιμοποιούνται τα βάρη του μοντέλου για να αρχικοποιηθεί το υπό εκπαίδευση μοντέλο C3D. Μέσω της εκπαίδευσης επί των δεδομένων της βάσης KTH και SKIG, τα βάρη του C3D μοντέλου προσαρμόζονται ώστε να ταξινομούνται clip από τα βίντεο της κάθε βάσης, μήκους 16 καρέ, σε 6 και 10 κλάσεις ανθρώπινων δράσεων και χειρονομιών αντίστοιχα. Επίσης, για να επιτελείται η παραπάνω διαδικασία αντικαταστάθηκε το στάδιο εξόδου του μοντέλου με Softmax μη γραμμικότητα 6 εξόδων που καθιστά το τελικό στάδιο ένα Single Layer Perceptron ταξινομητή που προβλέπει χρησιμοποιώντας τα τοπικά χωροχρονικά χαρακτηριστικά που εξάγονται από τα προηγούμενα στρώματα. Ο αριθμός των παραμέτρων του μοντέλου είναι 80 εκατομμύρια και οι πυρήνες συνέλιξης έχουν επιλεχθεί να είναι σε όλο το μήκος του δικτύου κυβικοί διάστασης $3 \times 3 \times 3$. Κατά τα πειράματα με το C3D λόγω της χρήσης των προεκπαιδευμένων παραμέτρων δεν γίνεται να τροποποιηθεί η αρχιτεκτονική του μοντέλου.

Σχήμα 5.4: Το μοντέλο C3D : η είσοδος του μοντέλου είναι 16 καρέ ενός βίντεο, όλοι οι πυρήνες συνέλιξης είναι κυβικοί διάστασης $3 \times 3 \times 3$ και περιλαμβάνει 80 εκατομμύρια παραμέτρους-βάρη. Λήφθηκε από την αναφορά[3].

Το μοντέλο 3D-CNN, είναι ένα σημαντικά μικρότερο και ρηχότερο, σε σχέση με το C3D, συνελικτικό δίκτυο τρισδιάστατης συνέλιξης. Επιλέχθηκε να αποτελείται από 3 συνελικτικά επίπεδα με κυβικούς πυρήνες ακολουθούμενα από ReLU μη γραμμικότητα και συσσωρευτικά στρώματα μεγίστου. Τέλος η ταξινόμηση γίνεται, μέσω ενός πλήρως συνδεδεμένου στρώματος νευρώνων που ακολουθείται από Softmax μη γραμμικότητα όπως και προηγουμένως. Η αρχικοποίηση των βαρών

του δικτύου γίνεται με βάση τη μέθοδο των Glorot και Bengio [63]. Η αρχικοποίηση αυτή έχει φανεί πειραματικά πως αποτρέπει τον άμεσο (δηλαδή από τις πρώτες εποχές εκπαίδευσης) κορεσμό των μη γραμμικών συναρτήσεων, γεγονός που μειώνει σημαντικά τις τιμές της κλίσης και επιβραδύνει σημαντικά ή ακόμα και καθιστά αδύνατη τη συνέχιση της εκπαίδευσης. Ο αριθμός των παραμέτρων (βαρών) του μοντέλου είναι περίπου $3.141.100$. Η αρχιτεκτονική που τελικά έδωσε τα καλύτερα αποτελέσματα παρουσιάζεται στο Σχήμα 5.5. Για να καταλήξουμε σε αυτή την αρχιτεκτονική, δοκιμάστηκαν αρκετές τροποποιημένες εκδοχές της με τον περιορισμό να κρατηθεί χαμηλά ο αριθμός παραμέτρων.



Σχήμα 5.5: Το μοντέλο 3D-CNN: η είσοδος του μοντέλου είναι 8 καρέ ενός βίντεο η οποία μετασχηματίζεται σε ένα διάνυσμα 512 στοιχείων μέσω σταδίων Conv3D-ReLU-Max Pooling που επαναλαμβάνονται 3 φορές ακολουθιακά.

5.2.1 Συνάρτηση Κόστους

Για την εκπαίδευση των μοντέλων κάθε επανάληψη του αλγορίθμου S.G.D με Minibatches και χρήση επιτάχυνσης Nesterov που περιγράφεται στον πίνακα αλγορίθμου 1. Ο αλγόριθμος αυτός εκτελείται επί ενός υποσυνόλου των δεδομένων εκπαίδευσης (Minibatch). Η συνάρτηση κόστους που χρησιμοποιήθηκε για την εκπαίδευση των μοντέλων ήταν η κατηγορική αλληλοεντροπία (categorical crossentropy) που περιγράφεται αναλυτικά στην ενότητα 4.1.3. Η προσαρμογή της στο παρόν πρόβλημα γίνεται ως εξής: Αν κάθε βίντεο σταθερής διάρκειας L καρέ αποτελείται από C clips (χωρίς επικάλυψη) των καρέ το καθένα, τότε για κάθε υποσύνολο των βίντεο εκπαίδευσης $Minibatch = \{V_1, V_2, ..., V_B\}$ που περιέχει βίντεο προκύπτουν $N = B \times C$ clips. Επίσης, αν έχουμε D βίντεο εκπαίδευσης τότε προκύπτουν $M = \lfloor \frac{D}{B} \rfloor$ minibatches. Για το κάθε clip $c^{(i,m)}$ η κατηγορική μεταβλητή $y^{(i,m)}$ περιέχει την κλάση της δράσης η οποία εμφανίζεται στο i-οστό clip του m-οστού minibatch. Στην παραπάνω ποσότητα προστίθεται ένας όρος κανονικοποίησης ως προς την L_2 νόρμα των παραμέτρων του μοντέλου για να αποφευχθεί το φαινόμενο

του *ovefitting*. Η μαθηματική έκφραση για τη συνάρτηση κόστους τελικά είναι:

$$J(\mathbf{w}) = -\frac{1}{M}\sum_{m=1}^{M}\sum_{i=1}^{N}\sum_{j=1}^{n}\mathbb{1}\{y^{(i,m)}=j\}\log(p(y^{(i,m)}=j|c^{(i,m)};\mathbf{w})) + \gamma\sum_{i=1}^{W}w_i^2 \tag{5.3}$$

όπου γ είναι η υπερπαράμετρος που ρυθμίζει την επίδραση της κανονικοποίησης στην εκπαίδευση του μοντέλου, W η διάσταση του μοντέλου στον χώρο των παραμέτρων, N ο αριθμός clip του κάθε minibatch δεδομένων εκπαίδευσης, και n ο αριθμός των κλάσεων που είναι 6 και 10 για τις βάσεις KTH και SKIG, αντίστοιχα. Οι κανόνες ανανέωσης των παραμέτρων που αντιστοιχούν στην παραπάνω συνάρτηση κόστους είναι οι εξής:

$$\mathbf{w}^{(i+1)} = \mathbf{w}^{(i)} - \lambda\langle\nabla J(\mathbf{w})\rangle_{minibatch} - 2\gamma\mathbf{w}^{(i)} \tag{5.4}$$

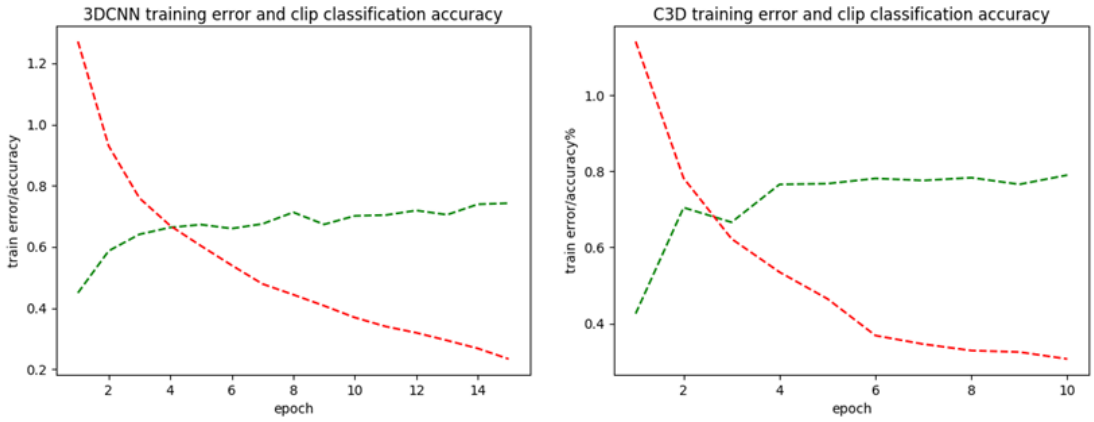
όπου ο εκθέτης αναφέρεται στον αριθμό της επανάληψης του αλγορίθμου S.G.D και η κλίση υπολογίζεται σαν ο μέσος όρος των κλίσεων που υπολογίζονται για τα παραδείγματα ενός minibatch.

5.2.2 Επιλογή υπερπαραμέτρων και αξιολόγηση

Για την επιλογή κατάλληλης τιμής υπερπαραμέτρων εκπαίδευσης, δηλαδή του *αρχικού* ρυθμού μάθησης, του μεγέθους κάθε minibatch και του συντελεστή κανονικοποίησης (λ, B, γ) εκτελέσθηκαν πολλές δοκιμές καθώς δεν υπάρχει τρόπος οι βέλτιστες τιμές τους να υπολογιστούν πριν την εκπαίδευση. Εδώ παρουσιάζονται για κάθε μοντέλο οι τιμές που έδωσαν τα καλύτερα αποτελέσματα. Για το μοντέλο 3D-CNN χρησιμοποιήθηκε $\lambda = 0.001$, $B = 4$ βίντεο και $\gamma = 0.00003$ και η εκπαίδευση διήρκεσε 50 εποχές και κάθε εποχή διήρκεσε περίπου 20 λεπτά. Για το μοντέλο C3D χρησιμοποιήθηκε $\lambda = 0.00001$, $B = 4$ βίντεο και $\gamma = 0.000003$ η εκπαίδευση διήρκεσε 15 εποχές με κάθε εποχή να διαρκεί περίπου 3 ώρες. Κατά την εκπαίδευση χρησιμοποιείται batch normalization,dropout και "hard-coded" μεταβολή του ρυθμού μάθησης. Και στις δυο περιπτώσεις παρατηρήσαμε πως η διαδικασία βελτιστοποίησης συνέκλινε πριν την ολοκλήρωση του προκαθορισμένου αριθμού εποχών. Κατά συνέπεια, το τελικό μοντέλο που παρουσιάζεται και αξιολογείται για κάθε περίπτωση δεν είναι αυτό που προκύπτει από την ολοκλήρωση του προκαθορισμένου αριθμού εποχών, αλλά αυτό που επιτυγχάνει τη μέγιστη ακρίβεια στον μικρότερο δυνατό αριθμό εποχών. Στη δεύτερη περίπτωση χρησιμοποιήθηκε μικρότερη τιμή ρυθμού μάθησης λόγω του γεγονότος ότι το μοντέλο που χρησιμοποιείται είναι προεκπαιδευμένο και δεν ήταν επιθυμητό να αλλάξουν σημαντικά τα βάρη του μοντέλου.

Για την αξιολόγηση των μοντέλων 3DCNN και C3D υπολογίστηκαν η μέση ακρίβεια πρόβλεψης (mAP) σε επίπεδο clip και η μέση ακρίβεια πρόβλεψης σε επίπεδο βίντεο. Στην πρώτη περίπτωση η πρόβλεψη του μοντέλου είναι η κλάση για

την οποία η posterior πιθανότητα $p_k(class = k|c)$, όπως δίνεται από την έξοδο του softmax σταδίου εξόδου, είναι μέγιστη. Στη δεύτερη περίπτωση οι έξοδοι για κάθε clip ενός βίντεο αθροίζονται και η πρόβλεψη είναι η κλάση για την οποία το άθροισμα των posterior πιθανοτήτων είναι μέγιστο, ακολουθώντας την προσέγγιση *πλειοψηφικής απόφασης με βάρη* (weighted majority voting scheme). Τα αποτελέσματα



Σχήμα 5.6: Η μέση ακρίβεια ταξινόμησης (πράσινη καμπύλη) σε επίπεδο clip και το κόστος εκπαίδευσης(κόκκινη καμπύλη) του μοντέλου 3D-CNN και C3D. Κατά την εκπαίδευσης χρησιμοποιούνται batch normalization, dropout και "hard-coded" μεταβολή του ρυθμού μάθησης. Στην περίπτωση του 3D-CNN το τελικό μοντέλο που χρησιμοποιήθηκε στην αξιολόγηση είναι αυτό που προέκυψε στην 15ή εποχή εκπαίδευσης ενώ για το C3D αυτό που προέκυψε στην 10ή εποχή εκπαίδευσης.

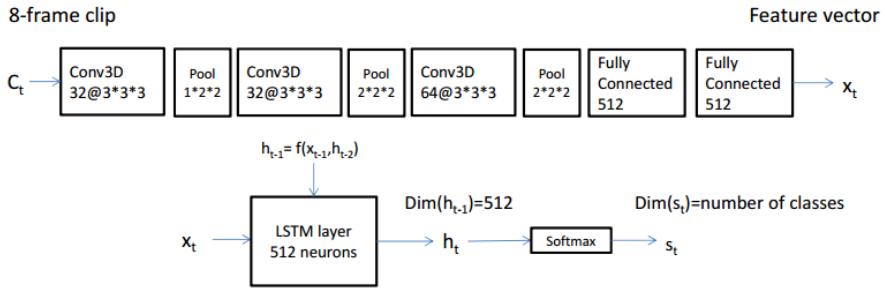
παρουσιάζονται στον πίνακα 5.1. Παρατηρούμε πως στις περισσότερες περιπτώσεις τα μοντέλα πετυχαίνουν ικανοποιητικά ποσοστά επιτυχίας αλλά σημαντικά χαμηλότερα συγκρίσιμα με αποτελέσματα της βιβλιογραφίας στις ίδιες βάσεις δεδομένων. Οι τιμές του πίνακα 5.1 θεωρούνται μία βασική μέτρια απόδοση (baseline) και θα συγκριθούν με τα αποτελέσματα των πιο σύνθετων μοντέλων που ακολουθούν. Επίσης, ένα ουσιαστικό συμπέρασμα που εξήχθη από αυτά τα πειράματα είναι ότι για τον αριθμό δεδομένων που είναι διαθέσιμα δεν είναι δυνατόν να εκπαιδευτούν τόσο βαθιά μοντέλα όσο το C3D για λόγους στατιστικούς αλλά και χρονικούς αφού μία εκπαίδευση 10 εποχών του C3D διαρκεί περίπου 1 μέρα. Με βάση τα στοιχεία αυτά αποφασίστηκε στην συνέχεια να περιοριστούμε σε αρκετά μικρότερα μοντέλα.

Dataset-Model	C3D	3DCNN
KTH	79.8	74.9
SKIG Rgb	75.6	69.3
SKIG Depth	76.1	73.8
SKIG Rgb-D	75.2	71.3

Πίνακας 5.1: mAP για τα μοντέλα 3DCNN και C3D στις βάσεις δεδομένων KTH και SKIG.

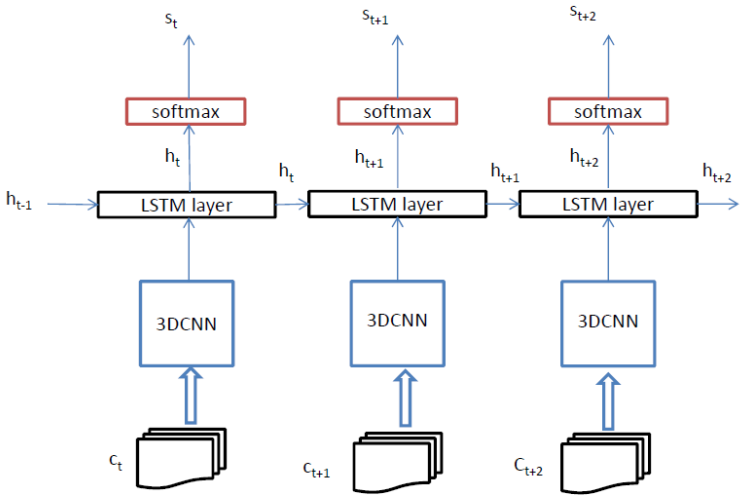
5.3 Το μοντέλο 3DCNN-LSTM

Στη συνέχεια παρουσιάζεται το μοντέλο 3D-CNN-LSTM το οποίο αποτελεί συνδυασμό συνελικτικού και αναδρομικού δικτύου. Η επέκταση του σε σχέση με τα προηγούμενα δύο μοντέλα είναι πως εκτός της επεξεργασίας της *τοπικής χωροχρονικής (local spatiotemporal)* πληροφορίας μέσω του συνελικτικού δικτύου περιλαμβάνει και την ενσωμάτωσή της σε ένα αναδρομικό μοντέλο που λαμβάνει υπόψη την *συνολική χρονική εξέλιξη (global temporal modeling)* της δράσης. Για τα πειράματα που εκτελέσθηκαν, επιλέχθηκε να σταθεροποιηθεί ο αριθμός των LSTM νευρώνων στους 512 και η μη γραμμικότητα που χρησιμοποιήθηκε για την έξοδο κάθε νευρώνα ήταν $\sigma_h = \tanh(\cdot)$. Για να συνδυαστούν τα χαρακτηριστικά που εξάγονται για ένα clip c_t από το 3D-CNN, με το στρώμα νευρώνων LSTM, αφαιρούμε το στάδιο softmax του πρώτου και η έξοδος του τελευταίου πλήρως συνδεδεμένου επιπέδου χρησιμοποιείται σαν είσοδος του αναδρομικού δικτύου. Για την εκδοχή του μοντέλου 3DCNN-LSTM,που έδωσε τα καλύτερα αποτελέσματα, η είσοδος του στρώματος LSTM είναι το διάνυσμα χωροχρονικών χαρακτηριστικών 512 στοιχείων που εξάγεται για κάθε χρονικά μη επικαλυπτόμενο clip διάρκειας ενός βίντεο 8 καρέ. Συνεπώς, για κάθε clip το στρώμα νευρώνων παράγει μία έξοδο h_t. Στη συνέχεια, η έξοδος αυτή τροφοδοτείται σε ένα πλήρως συνδεδεμένο επίπεδο softmax μη γραμμικότητας 512 εισόδων και n εξόδων (δηλαδή όσο και ο αριθμός κλάσεων). Η έξοδος της Softmax μη γραμμικότητας για κάθε clip c_t είναι posterior πιθανότητα $s_t = p_j(class = j|c_t, c_{t-1}..., c_0)$ για $j = 1...n$. Η παραπάνω περιγραφή της αρχιτεκτονικής του μοντέλου φαίνεται και στο Σχήμα 5.7. Οι υπό εκπαίδευση παράμετροι (βάρη) του μοντέλου είναι 17.201.450. Επιλέξαμε να μεγαλώσουμε το μέγεθος των μοντέλων αλλά όχι να πλησιάσουμε τα μεγέθη πολύ βαθιών αρχιτεκτονικών σαν το C3D, που περιγράφηκε στην προηγούμενη ενότητα. Η πρόβλεψη του μοντέλου για



Σχήμα 5.7: Η αρχιτεκτονική του μοντέλου 3D-CNN-LSTM. Στο πάνω μέρος του σχήματος περιγράφεται η λειτουργία του συνελικτικού μέρους του μοντέλου ενώ στο κάτω μέρος περιγράφεται η λειτουργία του αναδρομικού μέρους του μοντέλου.

κάθε clip λαμβάνεται, όπως και στις προηγούμενες περιπτώσεις, ως η κλάση της οποίας η posterior πιθανότητα είναι μέγιστη. Στη συνέχεια η πρόβλεψη για κάθε βίντεο προκύπτει συνδυάζοντας τις προβλέψεις των επιμέρους clip. Ειδικότερα, η πρόβλεψη για κάθε βίντεο είναι η κλάση για την οποία το άθροισμα των posterior



Σχήμα 5.8: Χρονικό ξεδίπλωμα του μοντέλου 3D-CNN-LSTM. Η είσοδος του μοντέλου σε κάθε χρονική στιγμή t είναι ένα χρονικά μη επικαλυπτόμενο clip του συνολικού video, για το οποίο υπολογίζεται η πιθανότητα να ανήκει σε κάθε από τις κλάσεις κάθε βάσης δεδομένων. Το παραπάνω σχήμα αποτελεί και ένα παράδειγμα του ”ξετυλίγματος” του αναδρομικού δικτύου (rnn unrolling) που ”εικονικά” συμβαίνει κατά την εκπαίδευση με τον αλγόριθμο BPTT.

πιθανοτήτων όλων των clip του βίντεο είναι μέγιστο. Ο μαθηματικός συμβολισμός της πρόβλεψης είναι ο εξής:

$$\hat{y}_{video} = \arg\max_{i} \sum_{t=1}^{C} p(class = i | c_t, c_{t-1}, \ldots, c_0) \tag{5.5}$$

όπου $C = L/T$ ο αριθμός των clip κάθε βίντεο, T τα καρέ εντός κάθε clip, c_t ένα clip του βίντεο. Και για τις δύο βάσεις δεδομένων χρησιμοποιήθηκε $L = 64$ καρέ, $= 8$ καρέ και συνεπώς $C = 8$ clips.

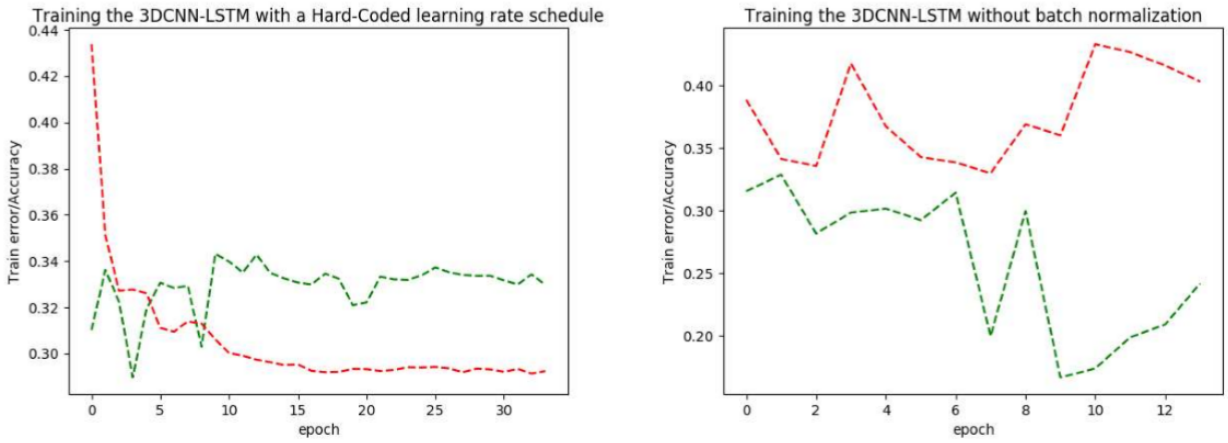
5.3.1 Εκπαίδευση με τον αλγόριθμο Back Propagation στο χρόνο και αποτελέσματα

Το μοντέλο 3DCNN-LSTM εκπαιδεύτηκε χρησιμοποιώντας τους αλγόριθμους *οπίσθιας μετάδοσης στον χρόνο (Back Propagation Through Time)* [49] (που περιγράφηκε στην ενότητα 4.3) για τον υπολογισμό της κλίσης και S.G.D (Stochastic Gradient Descent) με Minibatches και με χρήση επιτάχυνσης Nesterov (που συνοψίζεται στον πίνακα 1) για τη βελτιστοποίηση του κόστους. Η συνάρτηση κόστους που χρησιμοποιήθηκε ήταν η κατηγορική αλληλοεντροπία όπως αυτή περιγράφηκε από την εξίσωση 5.3. Ο αλγόριθμος B.P.T.T ουσιαστικά αποτελεί μία εκδοχή του αλγορίθμου Back Propagation με τη διαφορά ότι χρησιμοποιείται ένα εικονικό ”ξεδίπλωμα” του αναδρομικού νευρωνικού δικτύου στο πεδίο του χρόνου, εν προκειμένω του στρώματος LSTM νευρώνων. Με αυτό τον τρόπο υπολογίζεται η κλίση του σφάλματος ως προς τις παραμέτρους του δικτύου και ένα minibatch αποτελού-

μενο από ακολουθίες κάθε μια από τις οποίες αντιστοιχεί σε ένα βίντεο. Πιο συγκεκριμένα, η είσοδος του αλγορίθμου εκπαίδευσης σε κάθε επανάληψη, είναι πλέον μία ακολουθία clip που αποτελούν τμήματα ενός παραδείγματος βίντεο το οποίο έχει υποδειγματοληπτηθεί ώστε να έχει προκαθορισμένο αριθμό καρέ (Σχέση 5.2). Οι υπερπαράμετροι, ο προσδιορισμός των οποίων απαιτείται για την εκτέλεση του αλγορίθμου, ήταν το όριο στην τιμή της κλίσης (Gradient Clipping limit) που τέθηκε στις τιμές $\pm$ 100, καθώς και ο μέγιστος αριθμός χρονικών βημάτων κάθε ακολουθίας, τα οποία θα λαμβάνονται υπόψη κατά την ανανέωση των παραμέτρων του μοντέλου για κάθε νέο χρονικό βήμα, ο οποίος τέθηκε ίσος με το μήκος της ακολουθίας (δηλαδή κανένα χρονικό δεν αγνοήθηκε). Για $L = 64$ καρέ, $T = 8$ καρέ και συνεπώς $C = 8$ clips τα χρονικά βήματα είναι 8. Τονίζεται πως κατά την ανανέωση των βαρών σε επανάληψη του αλγορίθμου εκπαίδευσης, τροποποιούνται και τα βάρη του 3D-CNN δικτύου μέσω οπίσθιας τροφοδότησης του σφάλματος από την έξοδο του στρώματος LSTM μέχρι και την είσοδο του συνελικτικού μέρους του δικτύου.

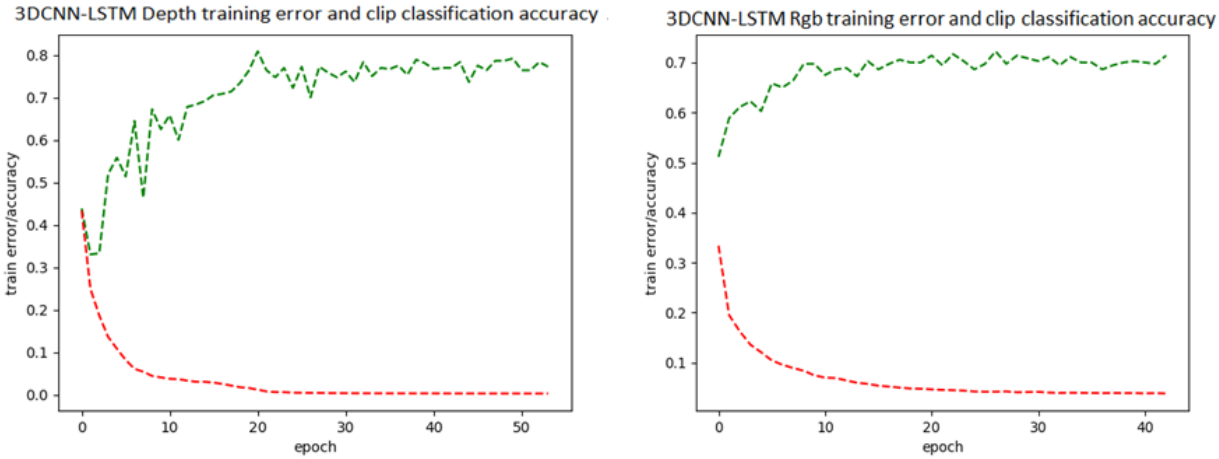
Για την επιλογή κατάλληλης τιμής υπερπαραμέτρων εκπαίδευσης, δηλαδή του *αρχικού* ρυθμού μάθησης, του μεγέθους κάθε batch και του συντελεστή κανονικοποίησης (λ, B, γ) εκτελέσθηκαν πολλές δοκιμές καθώς δεν υπάρχει τρόπος οι βέλτιστες τιμές τους να υπολογιστούν πριν την εκπαίδευση. Εδώ παρουσιάζονται για κάθε μοντέλο οι τιμές που έδωσαν τα καλύτερα αποτελέσματα. Στη βάση KTH χρησιμοποιήθηκε $\lambda = 0.02$, $B = 4$ βίντεο και $\gamma = 0.00003$, η εκπαίδευση διήρκεσε 60 εποχές και κάθε εποχή διήρκεσε περίπου 60 λεπτά. Στη βάση SKIG και για τις δύο τροπικότητες χρησιμοποιήθηκε $\lambda = 0.01$, $B = 4$ βίντεο και $\gamma = 0.00003$, η εκπαίδευση διήρκεσε 50 εποχές και κάθε εποχή διήρκεσε περίπου 40 λεπτά. Όπως έχει προαναφερθεί στην ενότητα 5.1.5 επειδή η χρονική διάρκεια των βίντεο δεν είναι σταθερή χρησιμοποιείται ομοιόμορφη δειγματοληψία για να επιλεχθούν τα καρέ που θα επιλεχθούν στην εκπαίδευση ανά βίντεο. Όσον αφορά στον προγραμματισμό του ρυθμού μάθησης, η αρχική επιλογή που έγινε ήταν να τίθεται σε κάποια προκαθορισμένη μικρότερη τιμή μετά από την ολοκλήρωση ενός προκαθορισμένου αριθμού εποχών. Γεγονός που σημαίνει πως για κάποιες εποχές ο ρυθμός ήταν σταθερός. Όπως φαίνεται στο αριστερό μέρος του Σχήματος 5.9 η εκπαίδευση με αυτό τον τρόπο παγιδεύτηκε σε κάποιο τοπικό ελάχιστο που δεν δίνει υψηλά ποσοστά ακρίβειας. Δοκιμάστηκε επίσης όταν βρεθεί σε τέτοια κατάσταση το δίκτυο, δηλαδή για συνεχόμενες εποχές να μην μειώνεται σημαντικά το κόστος, να αυξάνεται ο ρυθμός μάθησης ώστε να μπορέσει η πορεία της εκπαίδευσης να διαφύγει από την επίπεδη αυτή περιοχή. Αυτή η προσέγγιση δεν έδωσε κάποια βελτίωση.

Η επιπρόσθετη συνθετότητα που οφείλεται στην προσθήκη του αναδρομικού δικτύου φάνηκε πως απαιτεί πιο συντηρητική και ομαλή διαχείριση του ρυθμού μάθησης. Για το λόγο αυτό μεταβάλλαμε το ρυθμό μάθησης ανά εποχή, με βάση τη σχέση 5.1 και οι συντελεστές α και β αυξάνονται κάθε φορά που ολοκληρώνο-



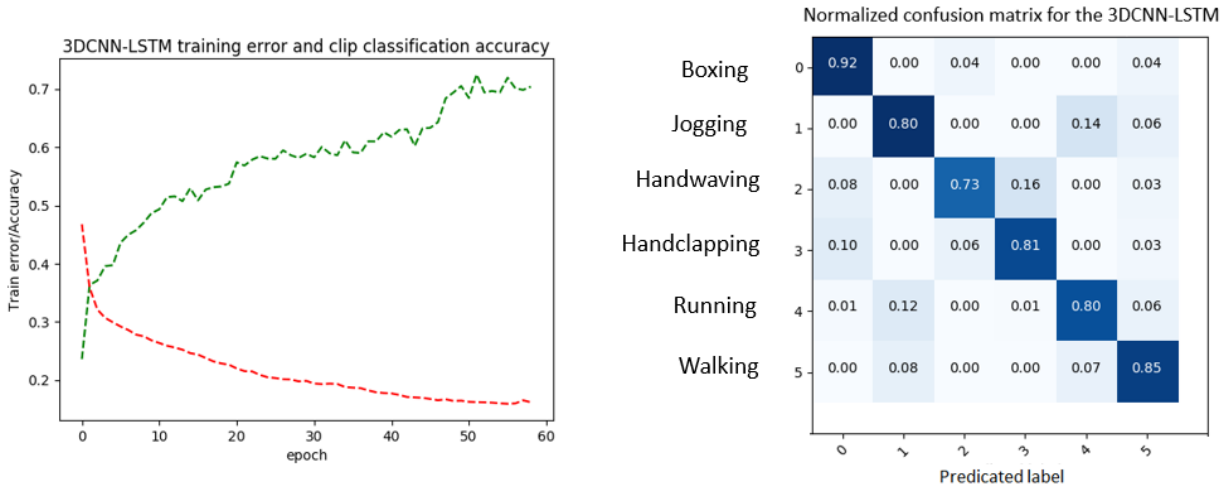
Σχήμα 5.9: Δύο περιπτώσεις αποτυχίας (εγκλωβισμού) της εκπαίδευσης του μοντέλου 3D-CNN-LSTM. Στο αριστερό διάγραμμα η εκπαίδευση (επί της βάσης KTH) χρησιμοποιεί "Hard-Coded" σχήμα μεταβολής του ρυθμού μάθησης της εξίσωσης 5.1 με αυτό τον τρόπο παγιδεύτηκε σε κάποιο τοπικό ελάχιστο που δεν δίνει υψηλά ποσοστά ακρίβειας. Στο δεξί διάγραμμα η εκπαίδευση δεν χρησιμοποιεί batch-normalization και παρατηρείται πως για έναν σημαντικό αριθμό εποχών έχουμε ταλάντωση τόσο του κόστους (κόκκινη καμπύλη) όσο και της ακρίβειας (πράσινη καμπύλη).

νται 10 εποχές, ώστε σταδιακά να επιτρέπονται όλο και μικρότερες μεταβολές των παραμέτρων. Αυτός ο προγραμματισμός του ρυθμού μάθησης οδήγησε σε καλύτερα αποτελέσματα. Η πορεία της εκπαίδευσης που χρησιμοποιεί τον παραπάνω προγραμματισμό του ρυθμού μάθησης απεικονίζεται στο Σχήμα 5.10 για τη βάση SKIG και στο Σχήμα 5.11 για τη βάση KTH.



Σχήμα 5.10: Η ακρίβεια ταξινόμησης σε επίπεδο clip και το σφάλμα εκπαίδευσης του μοντέλου 3DCNN-LSTM με χρήση των βίντεο Depth και RGB. Η μεταβολή του ρυθμού μάθησης ανά εποχή έγινε με βάση την σχέση 5.1. Η εκπαίδευση διεκόπη όταν για επαναλαμβανόμενο αριθμό εποχών η ακρίβεια ταξινόμησης δεν βελτιώθηκε. Παρατηρείται πως η σύγκλιση της εκπαίδευσης είναι πολύ ομαλότερη σε σχέση με την "Hard-Coded" μεταβολή του ρυθμού μάθησης.

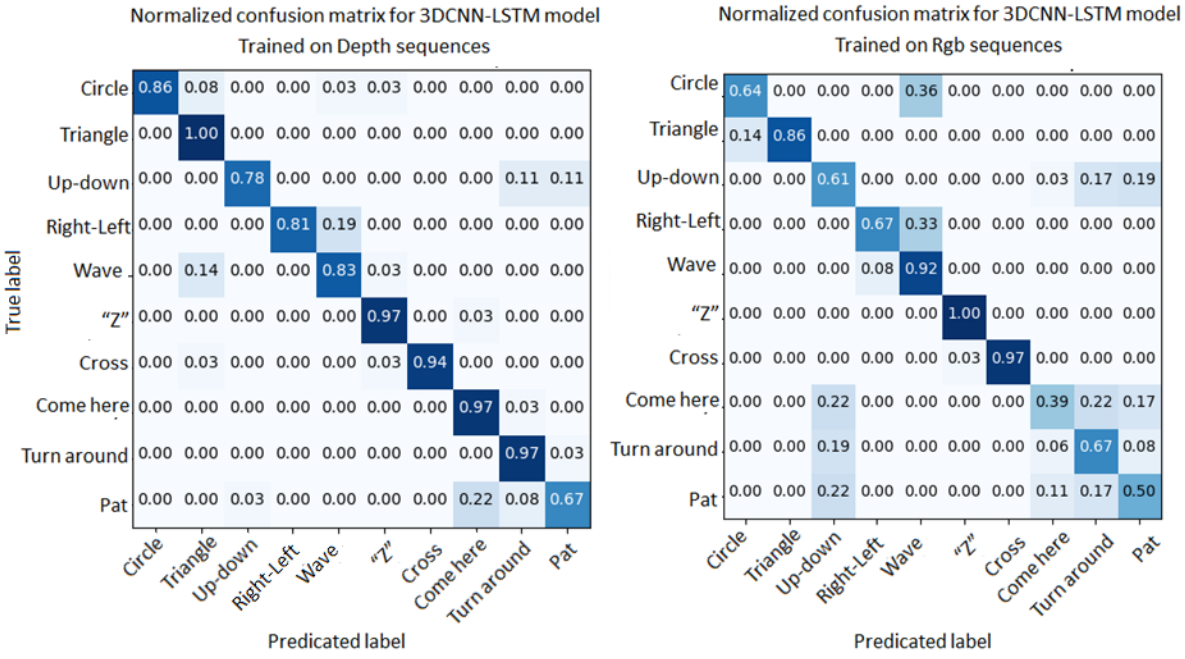
Η μέση ακρίβεια ταξινόμησης (mean Average Precision) και οι πίνακες σύγχυσης για τις βάσεις KTH και SKIG παρουσιάζονται στον πίνακα 5.2 και στο Σχήμα



Σχήμα 5.11: (Αριστερά)Η ακρίβεια ταξινόμησης σε επίπεδο clip και το σφάλμα εκπαίδευσης του μοντέλου 3DCNN-LSTM επί των δεδομένων της βάσης KTH. (Δεξιά) Ο πίνακας σύγχυσης του μοντέλου 3DCNN-LSTM εκπαιδευμένου χρησιμοποιώντας τα δεδομένα της KTH.

Dataset	3DCNN-LSTM
KTH	81.8
SKIG RGB	72.2
SKIG Depth	90.1

Πίνακας 5.2: mAP (επί τοις εκατό) για το μοντέλο 3DCNN-LSTM στις βάσεις δεδομένων KTH και SKIG. Η ίδια ακριβώς αρχιτεκτονική και του ίδιου μεγέθους μοντέλο χρησιμοποιήθηκε τόσο για τα πειράματα επί της βάσης SKIG (Depth και RGB) όσο και επί της βάσης KTH.

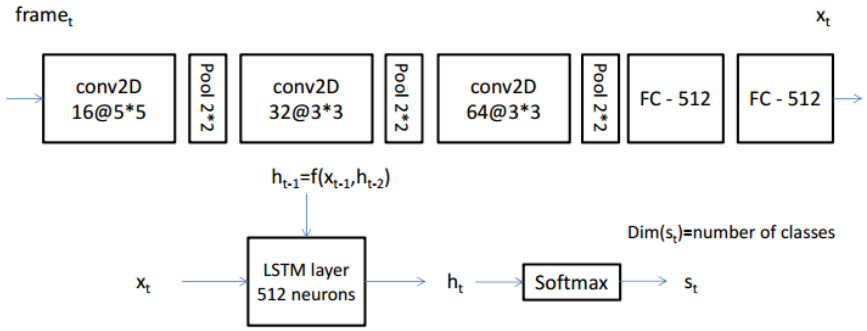


Σχήμα 5.12: Ο πίνακας σύγχυσης για το μοντέλο 3DCNN-LSTM εκπαιδευμένο με δεδομένα με βίντεο Depth και RGB από τη βάση Skig.

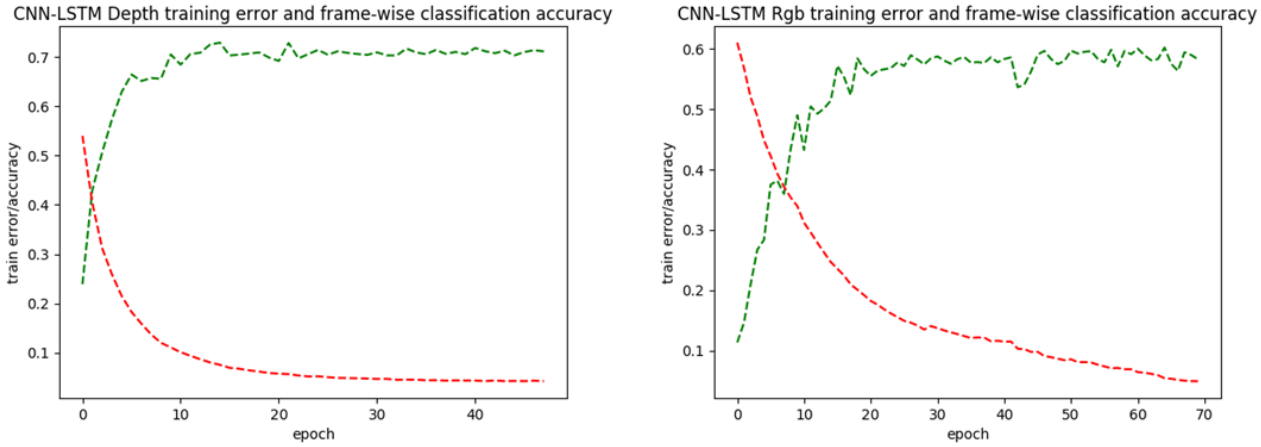
5.12, αντίστοιχα. Παρατηρούμε πως η τροπικότητα βάθους δίνει πολύ καλύτερα αποτελέσματα σε σχέση με την RGB. Σημαντική διαφορά ανάμεσα στα μοντέλα των δύο τροπικοτήτων παρουσιάζεται στις κλάσεις ”Come here”, ”Turn around”, ”Up down” και ”Pat”. Οι εκτελέσεις αυτών των χειρονομιών περιορίζονται σε μικρή κίνηση στο επίπεδο και περισσότερο περιέχουν κίνηση από και προς την κάμερα δημιουργώντας έτσι σημαντικές διακυμάνσεις στις τιμές βάθους. Ενδεχομένως αυτός είναι ένας από τους λόγους που η αναγνώριση αυτών των χειρονομιών είναι σχεδόν αλάνθαστη μέσω της τροπικότητας βάθους. Ένας άλλος παράγοντας που επηρεάζει σημαντικά την απόδοση του μοντέλου RGB και καθόλου την απόδοση του μοντέλου Depth είναι οι (ηθελημένα) μεγάλες διακυμάνσεις φωτεινότητας που εμφανίζουν τα δεδομένα της βάσης Skig. Πιο συγκεκριμένα ο αισθητήρας βάθους της κάμερας δεν επηρεάζεται καθόλου από τις διακυμάνσεις φωτεινότητας ενώ η μάθηση αναλλοίωτων ως προς την φωτεινότητα αναπαραστάσεων μέσω RGB δεδομένων από ένα σχετικά βαθύ νευρωνικό μοντέλο ενδέχεται να απαιτεί σημαντικά μεγαλύτερο αριθμό παραδειγμάτων από όσα είναι διαθέσιμα.

5.4 Το μοντέλο CNN-LSTM

Με σκοπό να συγκριθεί η εξαγωγή τοπικών χωροχρονικών χαρακτηριστικών από το μοντέλο 3D-CNN και η εξαγωγή τοπικών *μόνο* χωρικών χαρακτηριστικών από ένα δίκτυο δισδιάστατης συνέλιξης (2D Convolutional Neural Network), εκπαιδεύτηκε και αξιολογήθηκε το μοντέλο CNN-LSTM. Το μοντέλο αυτό δέχεται σαν είσοδο ένα καρέ μίας ακολουθίας και εξάγει ένα διάνυσμα χωρικών χαρακτηριστικών τα οποία όπως περιγράφηκε και στην προηγούμενη παράγραφο τροφοδοτούνται σε ένα στρώμα LSTM νευρώνων στων οποίων την έξοδο εφαρμόζεται Softmax μη γραμμικότητα και έτσι προκύπτει η πρόβλεψη του μοντέλου. Η αρχιτεκτονική του φαίνεται στο Σχήμα 5.13. Η εκπαίδευση του μοντέλου γίνεται με τον ίδιο τρόπο που παρουσιάστηκε για το μοντέλο 3D-CNN-LSTM της ενότητας 5.3 με τη μόνη διαφορά ότι η είσοδος ανά παράδειγμα βίντεο είναι μία ακολουθία από καρέ αντί για clip που περιέχουν πολλαπλά καρέ. Τα Σχήματα 5.14 και 5.15 παρουσιάζουν την ταυτόχρονη εξέλιξη του κόστους και της ακρίβειας ταξινόμησης κατά την εκπαίδευση, και τους πίνακες σύγχυσης για το μοντέλο εκπαιδευμένο με Depth και Rgb δεδομένα, αντιστοίχως. Η αρχιτεκτονική που έδωσε τα καλύτερα αποτελέσματα φαίνεται στο Σχήμα 5.13. Το μοντέλο έχει 21.885.722 υπό μάθηση παραμέτρους. Το μέγεθος του μοντέλου είναι συγκρίσιμο του μεγέθους των μοντέλων 3DCNN-LSTM ώστε να μπορούν να γίνουν συγκρίσεις ανάμεσα στα δύο μοντέλα.



Σχήμα 5.13: Η αρχιτεκτονική του μοντέλου CNN-LSTM.



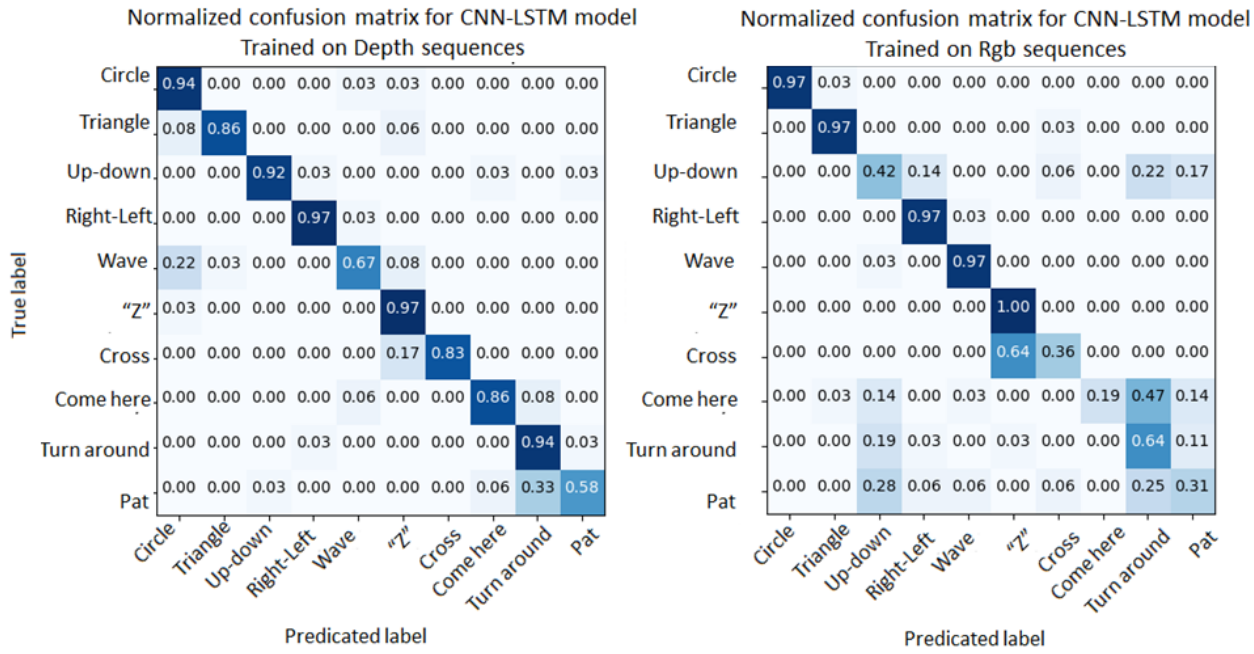
Σχήμα 5.14: Η ακρίβεια ταξινόμησης σε επίπεδο clip και το κόστος εκπαίδευσης του μοντέλου CNN-LSTM με χρήση δεδομένων Depth και Rgb βίντεο. Η εκπαίδευση διεκόπη όταν για επαναλαμβανόμενο αριθμό εποχών η ακρίβεια ταξινόμησης δεν βελτιώθηκε.

Dataset-Model	CNN-LSTM
SKIG Rgb	70.4
SKIG Depth	87.2
SKIG Rgb-D	87.7

Πίνακας 5.3: mAP (επί τοις εκατό) για το μοντέλο CNN-LSTM στην βάση δεδομένων SKIG. Η ίδια ακριβώς αρχιτεκτονική χρησιμοποιήθηκε και του ίδιου μεγέθους μοντέλο χρησιμοποιήθηκε για τα πειράματα με τις Depth και Rgb τροπικότητες της βάσης δεδομένων Skig.

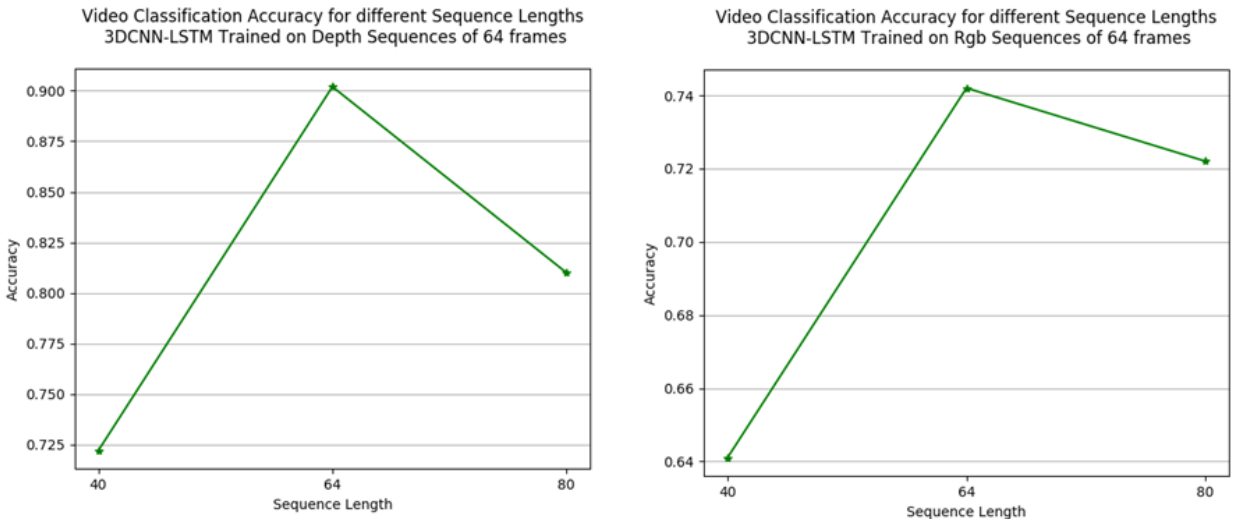
5.5 Ικανότητα γενίκευσης του μοντέλου 3DCNN-LSTM και CNN-LSTM για ακολουθίες διαφορετικής διάρκειας

Για τους σκοπούς της end-to-end εκπαίδευσης του μοντέλου 3DCNN-LSTM επιλέξαμε όλες οι ακολουθίες χειρονομιών της βάσης Skig να υποδειγματολειπτηθούν ή να προεκταθούν ώστε να έχουν σταθερή διάρκεια. Σε όλα τα παραπάνω πειράματα η διάρκεια αυτή ήταν 64 καρέ. Στην παρούσα ενότητα εξετάζεται κατά πόσο τα μοντέλα 3DCNN-LSTM και CNN-LSTM μπορούν να διατηρήσουν τα υψηλά ποσο-

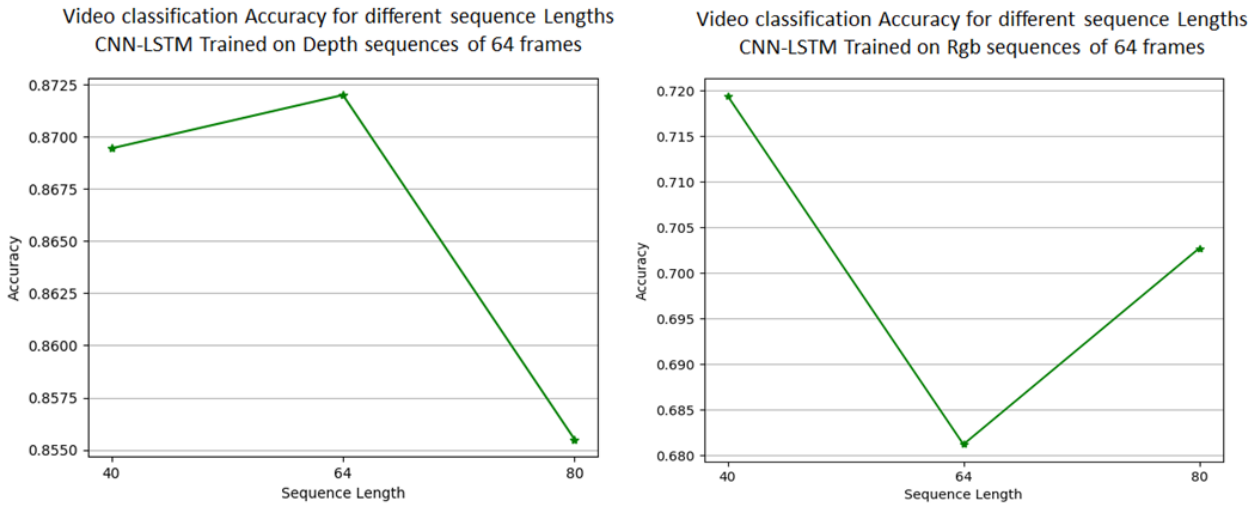


Σχήμα 5.15: Ο πίνακας σύγχυσης για το μοντέλο CNN-LSTM εκπαιδευμένο με δεδομένα με βίντεο Depth και Rgb από την βάση Skig.

στά ακρίβειας αν η διάρκεια της υπό ταξινόμηση χειρονομίας είναι διαφορετική από αυτή που χρησιμοποιήθηκε κατά την εκπαίδευση του μοντέλου. Πιο συγκεκριμένα παρουσιάζεται η μέση ακρίβεια των μοντέλων για κάθε τροπικότητα όταν τα βίντεο χειρονομιών έχουν διάρκεια μεγαλύτερη (80 καρέ) ή μικρότερη (40 καρέ) από αυτήν που χρησιμοποιήθηκε κατά την εκπαίδευση. Το πείραμα αυτό διεξήχθη μόνο για την βάση δεδομένων SKIG.



Σχήμα 5.16: Η μέση ακρίβεια ταξινόμησης του μοντέλου 3DCNN-LSTM για διαφορετικής διάρκειας ακολουθίες Depth και Rgb.



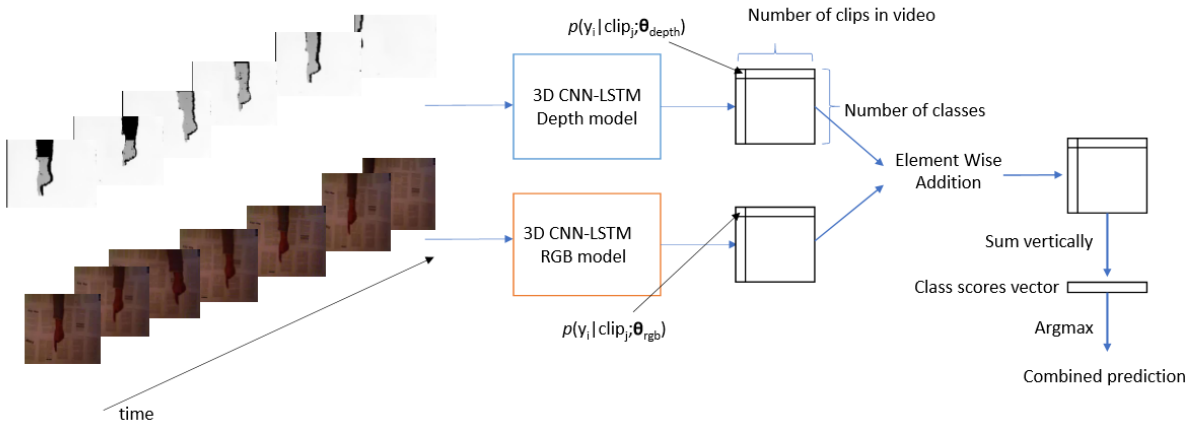
Σχήμα 5.17: Η μέση ακρίβεια ταξινόμησης του μοντέλου CNN-LSTM για διαφορετικής διάρκειας ακολουθίες Depth και Rgb.

5.6 Συνδυασμός Τροπικοτήτων στο τελικό στάδιο (Late Modality Fusion

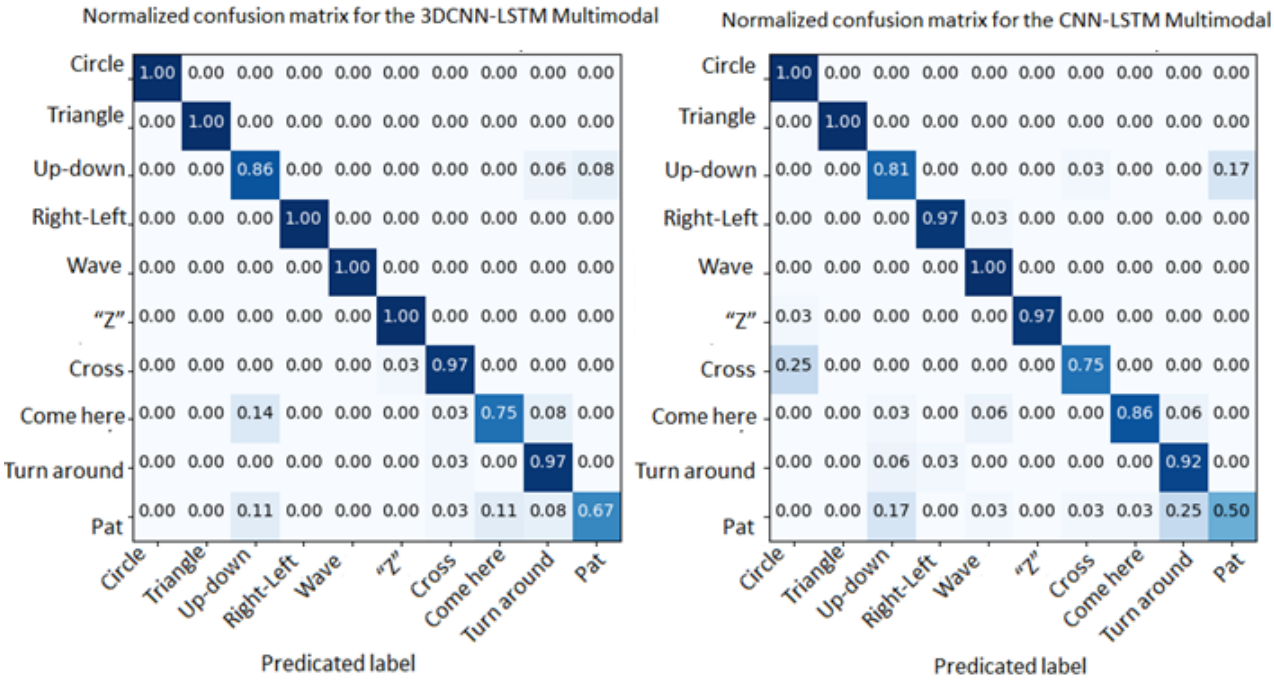
Για τα δεδομένα της βάσης SKIG που περιέχει συγχρονισμένα RGB και Depth βίντεο χειρονομιών υπολογίστηκε η μέση ακρίβεια πρόβλεψης όταν χρησιμοποιηθεί σύμμειξη των δύο τροπικοτήτων, για τα μοντέλα 3DCNN-LSTM και CNN-LSTM. Συγκεκριμένα για κάθε κλάση οι *posterior* πιθανότητες για κάθε clip ή frame της εισόδου, που υπολογίζεται από τα μοντέλα RGB και Depth προστίθενται. Με αυτό τον τρόπο προκύπτει ένα σύνολο από συνδυασμένα score εκ των οποίων το κάθε ένα εκφράζει την πιθανότητα κάθε clip να ανήκει σε μία εκ των κλάσεων. Στη συνέχεια, για κάθε κλάση αθροίζονται τα score που αντιστοιχούν σε αυτή ανά clip και η κλάση με το μεγαλύτερο άθροισμα αποτελεί την τελική πρόβλεψη. Το είδος αυτής της σύμμειξης που εφαρμόστηκε εμπίπτει στην προσέγγιση της *σύμμειξης στα τελικά στάδια επεξεργασίας (late modality fusion)*, που ορίζει πως η επεξεργασία της κάθε τροπικότητας γίνεται ξεχωριστά και οι προβλέψεις συνδυάζονται για να προκύψει η τελική πρόβλεψη. Το Σχήμα 5.18, παρέχει μία σχηματική απεικόνιση της παρακάτω διαδικασίας. Επίσης το Σχήμα 5.19 παρουσιάζει τους πίνακες σύγχυσης που προκύπτουν από την παραπάνω διαδικασία. Τέλος, η μέση ακρίβεια ταξινόμησης είναι 92.2% και 87.7% για τα μοντέλα 3DCNN-LSTM και CNN-LSTM, αντίστοιχα.

5.7 Η επίδραση της επαύξησης δεδομένων

Όπως είναι αναμενόμενο η αύξηση του μεγέθους του συνόλου δεδομένων εκπαίδευσης βελτιώνει τα αποτελέσματα. Για να ποσοτικοποιήθει πειραματικά η βελτίωση που επιφέρει η επαύξηση δεδομένων καταγράφηκε η μέση ακρίβεια με και χωρίς επαύξηση δεδομένων για το μοντέλο 3D-CNN-LSTM.



Σχήμα 5.18: Σύμμειξη τροπικοτήτων συνδυάζοντας το μοντέλο 3D CNN-LSTM που έχει εκπαιδευτεί με τα δεδομένα Depth και το μοντέλο 3D CNN-LSTM που έχει εκπαιδευτεί με τα δεδομένα RGB. Αντίστοιχη διαδικασία ακολουθείται και για το μοντέλο CNN-LSTM με τη διαφορά πως οι posterior πιθανότητες υπολογίζονται ανά καρέ και όχι ανά clip.



Σχήμα 5.19: Οι πίνακες σύγχυσης της ταξινόμησης επί της βάσης Skig, με χρήση συνδυασμού τροπικοτήτων στα μοντέλων 3DCNN-LSTM και CNN-LSTM της βάσης δεδομένων SKIG.

Dataset	Without Augmentation	With Augmentation
KTH	70.4	81.8
SKIG Depth	81.9	90.1
SKIG RGB	64.2	72.2

Πίνακας 5.4: Η επίδραση της επαύξησης δεδομένων για το μοντέλο 3DCNN-LSTM στις βάσεις δεδομένων KTH και SKIG. Η ίδια ακριβώς αρχιτεκτονική και του ίδιου μεγέθους μοντέλο χρησιμοποιήθηκε τόσο για τα πειράματα επί της βάσης SKIG (Depth και Rgb) όσο και επί της βάσης KTH.

5.8 Συμπεράσματα

Συγκεντρωτικά τα αποτελέσματα ακρίβειας ταξινόμησης για όλες τις βάσεις δεδομένων και όλα τα μοντέλα παρουσιάζονται στον πίνακα 5.5. Συνοπτικά τα αποτελέσματα είναι ως ακολούθως:

- **Επίδραση του βάθους και του μεγέθους του μοντέλου:** Παρατηρήθηκε πως όσο αυξάνει το μέγεθος του μοντέλου τόσο αποτελεσματικότερο γίνεται αυτό στην επίλυση του προβλήματος ταξινόμησης. Παρόλα αυτά στην περίπτωση που εφαρμόστηκε *finetuning* του προεκπαιδευμένου μοντέλου, επί ενός πολύ μικρότερου από το αρχικό συνόλου δεδομένων, (τα παραδείγματα της βάσης SKIG και KTH είναι 100 φορές λιγότερα από αυτά της 1 MSports), η διαφορά ανάμεσα στα δύο σύνολα δεδομένων μπορεί να ευθύνεται για την περιορισμένη βελτίωση που παρουσιάζει το βαθύ μοντέλο σε σχέση με το ρηχό. Στην περίπτωση του C3D και 3DCNN φάνηκε πως ούτε η βάση Skig ούτε η βάση KTH είχαν επαρκή αριθμό δεδομένων ώστε το επιπρόσθετο υπολογιστικό κόστος για να επανεκπαιδευτεί το μοντέλο C3D να αντισταθμίζεται από τη βελτίωση στην απόδοση του σε σχέση με ένα πολύ ρηχότερο, τυχαία αρχικοποιημένο μοντέλο (3D-CNN).

- **Δυνατότητα εκτίμησης υψηλής "εμπιστοσύνης" χωρίς να έχει ληφθεί υπόψη ολόκληρο το βίντεο της χειρονομίας:** Όπως φαίνεται στο Σχήμα 5.20 το μοντέλο 3DCNN στις περισσότερες κλάσεις καταφέρνει να προβλέψει ορθά και με υψηλή εμπιστοσύνη (πάνω από 80%) την κλάση στην οποία ανήκει η χειρονομία έχοντας λάβει υπόψη μόνο τα πρώτα 16 καρέ (δηλαδή 2 clip εισόδου).

- **Γενίκευση για ακολουθίες διαφορετικής χρονικής διάρκειας:** Με βάση τα Σχήματα 5.17 και 5.16 παρατηρούμε πως το μοντέλο είναι πιο εύρωστο στην αναγνώριση depth βίντεο, αφού η ακρίβεια ταξινόμησης μεταβάλλεται ελάχιστα αν αυξηθεί ή μειωθεί η διάρκεια της ακολουθίας εισόδου, σε σχέση με την διάρκεια των ακολουθιών που χρησιμοποιήθηκαν κατά την εκπαίδευση. Επίσης στην περίπτωση των Rgb βίντεο η ακρίβεια αυξάνεται τόσο όταν μειώνεται όσο και όταν αυξάνεται η διάρκεια της χειρονομίας.

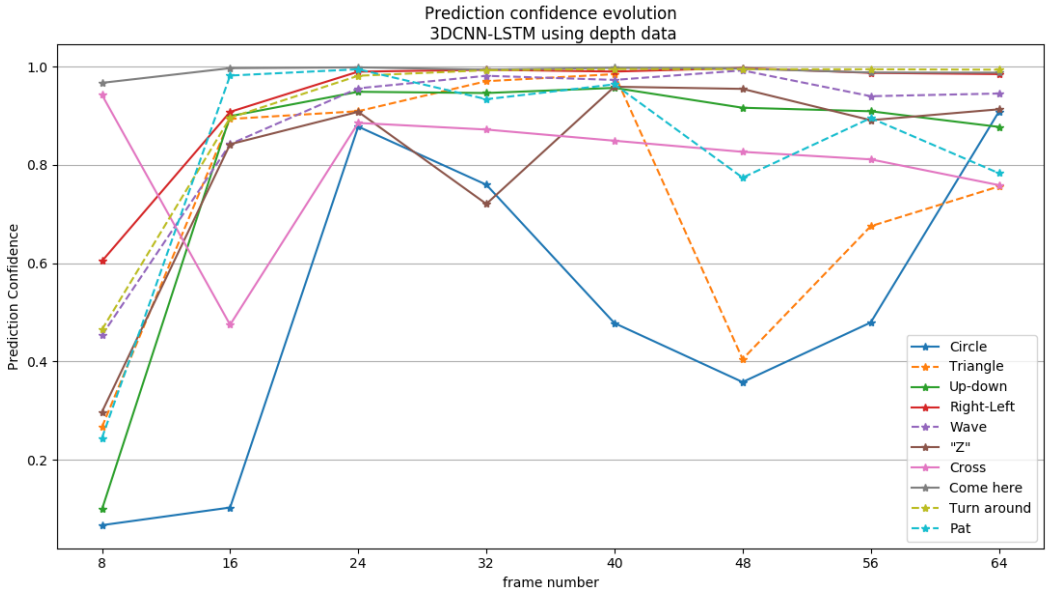
- **Συνδυασμός αναδρομικών και συνελικτικών δικτύων :** Ο συνδυασμός των αναδρομικού και συνελικτικού δικτύου βελτίωσε σημαντικά τη μέση ακρίβεια ταξινόμησης σε όλες τις περιπτώσεις. Αυτό επιβεβαιώνει ότι η χρονική μοντελοποίηση ανθρώπινης δράσης είναι εφικτή μέσω των μοντέλων 3DCNN-LSTM ή CNN-LSTM. Επιπλέον η ακρίβεια του μοντέλου 3DCNN-LSTM είναι ανώτερη αυτής του μοντέλου CNN-LSTM υποδεικνύοντας πως η εξαγωγή χωροχρονικών χαρακτηριστικών από το συνελικτικό δίκτυο οδηγεί σε καλύτερη πρόβλεψη των κλάσεων.

- **Σημασία της επαύξησης δεδομένων:** Η επαύξηση δεδομένων οδηγεί σε σημαντικές βελτιώσεις στην ακρίβεια των μοντέλων. Στην περίπτωση των βάσεων δεδομένων που χρησιμοποιήθηκαν η βελτίωση σε ορισμένες περιπτώσεις που διερευνήθηκαν άγγιξε περίπου το $10 - 13\%$ όπως δείχνουν οι τιμές του πίνακα 5.4. Προκύπτει έτσι το συμπέρασμα πως για μεσαία προς μικρά σύνολα δεδομένων με 6-10 κλάσεις και δράσεις μικρής διάρκειας, όπως αυτές που περιέχονται στις βάσεις που εξετάσθηκαν, είναι εφικτό να εκπαιδευτούν με επιτυχία μέτριου μεγέθους συνελικτικά και αναδρομικά νευρωνικά δίκτυα.

- **Σύγκριση της απόδοσης των μοντέλων για τις διαφορετικές τροπικότητες οπτικής πληροφορίας:** Όσον αφορά στην αναγνώριση χειρονομιών η τροπικότητα βάθους δίνει καλύτερα αποτελέσματα ενδεχομένως γιατί δεν επηρεάζεται από τις διακυμάνσεις φωτεινότητας και υποβάθρου που εντοπίζονται στα δεδομένα αξιολόγησης και εκπαίδευσης. Παρατηρούμε πως ακόμα και κλάσεις που απαιτούν fine-grained αναγνώριση διαχωρίζονται αρκετά καλά από το 3DCNN-LSTM μοντέλο με βάση τον πίνακα σύγχυσης 5.12, όπως οι κλάσεις Right-left και Wave που με δυσκολία διακρίνονται, εκ πρώτης όψεως, ακόμα και από άνθρωπο. Ακόμα, οι κλάσεις χειρονομιών με κίνηση κυρίως από και προς την κάμερα έχουν σχεδόν απόλυτη ακρίβεια αναγνώρισης από τα μοντέλα 3D-CNN-LSTM και CNN-LSTM. Η σύμμειξη τροπικοτήτων βελτιώνει την ακρίβεια πρόβλεψης των μοντέλων 3D-CNN-LSTM και CNN-LSTM.

- **Χωρική και χρονική υποδειγματοληψία και ταχύτητα υπολογισμού:** Όλα τα μοντέλα τα οποία μελετήθηκαν απαιτούν σχεδόν μηδενική προεπεξεργασία στην είσοδο. Ο χρόνος πρόβλεψης του μοντέλου 3D-CNN-LSTM με είσοδο μία ακολουθία 64 καρέ ήταν κατά μέσο όρο 0.12256 sec ενώ του μοντέλου CNN-LSTM ήταν 0.114912 sec. Εφόσον ο χρόνος αυτός είναι σημαντικά μικρότερος του 1 sec, που είναι ένα τυπικό όριο χρόνου ώστε ο υπολογισμός να θεωρείται ότι εκτελείται σε *πραγματικό χρόνο*, συμπεραίνεται πως τα μοντέλα που εκπαιδεύτηκαν θα μπορούσαν να ενσωματωθούν σε ένα σύστημα αναγνώρισης σε πραγματικό χρόνο (real time) όπως αυτό που εξετάζεται στο 6ο Κεφάλαιο. Πολύ σημαντικό ρόλο για την επίτευξη αυτής της ταχύτητας επεξεργασίας ήταν η χωρική υποδειγματοληψία των καρέ των βίντεο (64×64). Είναι σημαντικό πως παρότι σε όλες τις περιπτώσεις μειώθηκε ο όγκος της πληροφορίας εισόδου των μοντέλων επιτεύχθηκαν αρκετά υψηλά ποσοστά ακρίβειας. Επίσης τονίζεται πως δεν χρησιμοποιείται κανενός είδους εντοπισμός του χεριού ή του ατόμου αντίστοιχα στα βίντεο των βάσεων SKIG και KTH, αντίστοιχα. Συνεπώς, επιβεβαιώθηκε, η ικανότητα των μοντέλων να παρουσιάζουν αρκετά υψηλή ακρίβεια χρησιμοποιώντας ανεπεξέργαστα δεδομένα που έχουν υποστεί χωρική και χρονική υποδειγματοληψία.

Κατά την εκτέλεση των πειραμάτων οι βασικοί περιορισμοί που υπήρχαν ήταν



Σχήμα 5.20: Η μεταβολή της ”εμπιστοσύνης” πρόβλεψης με την πάροδο των καρέ εισόδου για παραδείγματα βίντεο χειρονομιών της βάσης δεδομένων SKIG που ταξινομήθηκαν στην σωστή κλάση από το μοντέλο 3DCNN-LSTM εκπαιδευμένο με Depth βίντεο. Με τον όρο ”εμπιστοσύνη” πρόβλεψης εννοείται η τιμή της posterior πιθανότητας της κλάσης στην οποία πραγματικά ανήκει το βίντεο, όπως δίνεται από το softmax στάδιο εξόδου του μοντέλου.

οι εξής:

- Σε αντίθεση με τα αναδρομικά νευρωνικά δίκτυα που επιτρέπουν επεξεργασία αυθαίρετης χρονικής διάρκειας βίντεο, τα συνελικτικά νευρωνικά δίκτυα επιβάλλουν σταθερές χωρικές διαστάσεις εισόδου και συνεπώς τα μοντέλα που εκπαιδεύτηκαν δεν μπορούν να αξιολογηθούν σε δεδομένα με διαφορετικές διαστάσεις.

- Ό χρόνος ”end-to-end” εκπαίδευσης του μοντέλου είναι το βασικό ”bottleneck” καθώς δεν υπάρχει αποδοτικός τρόπος να προαποφασιστούν οι διάφορες παράμετροι που την διέπουν. Κάποιες εμπειρικές πρακτικές που προτείνονται από τη βιβλιογραφία [48, 46] δοκιμάστηκαν και διευκόλυναν την επιλογή παραμέτρων ήταν η ρύθμιση του ρυθμού μάθησης επί ενός πολύ μικρού υποσυνόλου των δεδομένων εκπαίδευσης.

- Η εκπαίδευση ενός μοντέλου σε λογικά χρονικά πλαίσια επιτυγχάνεται με τη χρήση της GPU της nvidia GT 780 που προσφέρει 2GB RAM. Συνεπώς τα μεγέθη των μοντέλων που χρησιμοποιήθηκαν για τα μοντέλα 3D-CNN-LSTM και CNN-LSTM ενώ θεωρητικά θα μπορούσαν να είναι αρκετά μεγαλύτερα, περιορίστηκαν σε μεγέθη που να είναι εφικτό να φορτωθούν στην κάρτα γραφικών. Ταυτόχρονα το μέγεθος του batch των δεδομένων εκπαίδευσης που χρησιμοποιείται ανά επανάληψη του αλγορίθμου εκπαίδευσης περιορίζεται

επίσης λόγω του ορίου μνήμης της GPU. Η χρήση τόσο μεγαλύτερων μοντέλων τόσο και μεγαλύτερου μεγέθους batch θα μπορούσαν να δώσουν βελτιωμένα αποτελέσματα ταξινόμησης. Φυσικά, πρέπει να σημειωθεί πως η αυθαίρετη αύξηση του μεγέθους του κάθε μοντέλου θα προαπαιτούσε και σημαντικά περισσότερα δεδομένα για την εκπαίδευση του. Συνυπολογίζοντας τους παραπάνω περιορισμούς τα μοντέλα που εκπαιδεύτηκαν και τα αποτελέσματα που έδωσαν κρίνονται αρκετά ικανοποιητικά παρότι απέχουν από τα state of the art για τις χρησιμοποιούμενες βάσεις (KTH και SKIG).

Dataset	C3D	3DCNN	3DCNN-LSTM	CNN-LSTM
KTH Gray	79.8	74.9	81.8	-
SKIG Rgb	75.6	69.3	72.2	70.4
SKIG Depth	76.1	73.8	90.1	87.2
SKIG Rgb-D (late fusion)	-	-	92.2	87. 7

Πίνακας 5.5: Συγκεντρωτικά αποτελέσματα.

Κεφάλαιο 6

On-line Σύστημα Αναγνώρισης Χειρονομιών

6.1 On-line λειτουργία

Όπως εύκολα διαπιστώνεται, οποιοδήποτε μοντέλο από αυτά που έχουν περιγραφεί έως τώρα δεν μπορεί να αξιοποιηθεί άμεσα για τη δημιουργία ενός διαδραστικού περιβάλλοντος επικοινωνίας ανθρώπου-υπολογιστή μέσω χειρονομιών. Στα πειράματα του 5ου Κεφαλαίου, βασική προϋπόθεση ώστε να καθίσταται δυνατή η end-to-end εκπαίδευση των μοντέλων, είναι τα δεδομένα εκπαίδευσης να έχουν κατατμηθεί σε βίντεο που περιέχουν μόνο μία εκτέλεση της δράσης (όπως και συμβαίνει). Για να σχεδιαστεί ένα σύστημα που δέχεται συνεχή ροή βίντεο και αναγνωρίζει τις χειρονομίες που εκτελούνται από τον χρήστη, μία απλή λύση που υιοθετήθηκε είναι η χρήση ενός *ανιχνευτή δραστηριότητας (Activity Detector)* που περιγράφεται παρακάτω στην ενότητα 6.4. Πρόκειται για έναν ευριστικό τρόπο λήψης απόφασης σχετικά με την ύπαρξη δραστηριότητας εντός ενός χρονικού παραθύρου της συνεχούς ροής βίντεο. Μία δεύτερη σημαντική παράμετρος σχεδίασης του συστήματος είναι ο χρόνος εκτέλεσης των υπολογισμών που παράγουν την πρόβλεψη του συστήματος. Όπως έχει φανεί και στις μετρήσεις που παρουσιάζονται στον Πίνακα 6.1 τα μοντέλα CNN-LSTM και 3DCNN-LSTM επιτυγχάνουν ικανοποιητικά γρήγορο χρόνο υπολογισμού για εισόδους χρονικής διάρκειας 64 καρέ ανάλυσης 64×64. Ο τελικός σκοπός του συστήματος είναι η αντιστοίχηση μίας κλάσης χειρονομίας σε κάθε τμήμα της συνεχούς ροής που έχει επισημανθεί ως "έχων δραστηριότητα" από τον ανιχνευτή. Συνεπώς η συνολική λειτουργία του συστήματος μπορεί να συνοψιστεί στα εξής βήματα:

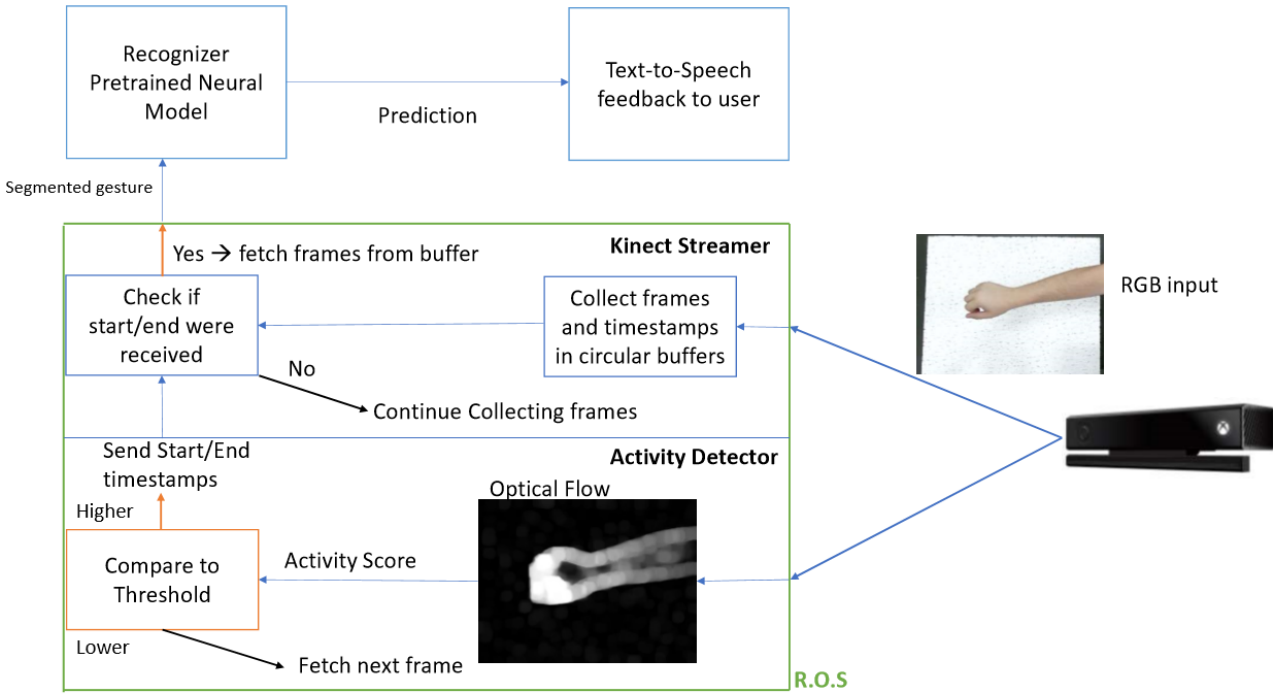
- Λήψη RGB και Depth video από τον αισθητήρα Kinect.
- Λήψη απόφασης για την ύπαρξη ή μη ύπαρξη, δραστηριότητας εντός των καρέ εισόδου
- Αν εντοπισθεί δραστηριότητα σώζονται η χρονική επισημείωση (time stamp) έναρξης και ολοκλήρωσης της δραστηριότητας και τα καρέ εντός αυτού του χρονικού διαστήματος τροφοδοτούνται στο μοντέλο αναγνώρισης, αφού λάβει την κατάλληλη χρονική και χωρική διάσταση (συμβατή με την είσοδο που

δέχεται το μοντέλο). Αν δεν εντοπισθεί δραστηριότητα το σύστημα εκτελεί το 1ο βήμα.

- Αν η εμπιστοσύνη πρόβλεψης του μοντέλου είναι μεγαλύτερη ενός κατωφλίου τότε ”δημοσιοποιείται” η κλάση χειρονομίας που αναγνωρίσθηκε, αλλιώς η πρόβλεψη αγνοείται και το σύστημα επιστρέφει στο 1ο βήμα.

Dataset|Model	3DCNN-LSTM	CNN-LSTM
SKIG Rgb	0.12256	0.114912
SKIG Depth	0.12134	0.124176

Πίνακας 6.1: Μέσοι Χρόνοι υπολογισμού σε sec για είσοδο 64 καρέ, ανάλυσης 64×64. Οι μέσοι χρόνοι μετρήθηκαν επί του συνόλου αξιολόγησης που χρησιμοποιήθηκε για τα πειράματα της βάσης SKIG. Οι υπολογισμοί εκτελέσθηκαν με χρήση της κάρτας γραφικών nvidia GT 780



Σχήμα 6.1: Λειτουργία του on-line συστήματος αναγνώρισης χειρονομιών. Στο σχήμα διαχωρίζεται εντός του πράσινου πλαισίου όποια διεργασία εκτελείται εντός του περιβάλλοντος του R.O.S.

6.2 Επιλογή Μοντέλου

Τα μοντέλα που δοκιμάστηκαν στο σχεδιασμό του συστήματος ήταν τα CNN-LSTM και 3DCNN-LSTM που στο 5ο Κεφάλαιο πέτυχαν την υψηλότερη ακρίβεια ταξινόμησης στη βάση δεδομένων SKIG. Παρότι τα αποτελέσματα της αξιολόγησης του μοντέλου 3D-CNN-LSTM και CNN-LSTM ήταν ανώτερα στην περίπτωση

του βάθους, η ενσωμάτωση του μοντέλου βάθους οδήγησε σε πολύ χαμηλή ακρίβεια ταξινόμησης της τάξης του 10%. Πιθανές αιτίες για την πολύ χαμήλη απόδοση του μοντέλου βάθους είναι οι εξής:

- Το μοντέλο έχει προσαρμόσει τις παραμέτρους του ώστε να επιλύει με υψηλή ακρίβεια το πρόβλημα ταξινόμησης των δεδομένων της βάσης SKIG. Αυτό έχει σαν αποτέλεσμα η σημαντική αλλαγή των συνθηκών λήψης βίντεο βάθους να μειώνει πολύ την ικανότητα του μοντέλου να διακρίνει τις χειρονομίες σωστά.

- Στο παραπάνω πρόβλημα έρχεται να προστεθεί το γεγονός πως ο αισθητήρας KINECT για την καταγραφή των δεδομένων της βάσης SKIG ήταν παλαιότερης έκδοσης από αυτόν που χρησιμοποιήθηκε για την σχεδίαση του συστήματος. Επίσης το ύψος στο οποίο τοποθετήθηκε ο αισθητήρας ρυθμίστηκε κατά προσέγγιση ώστε να βρίσκεται σε παρόμοιο ύψος μ' αυτό που βρισκόταν η κάμερα κατά την καταγραφή των δεδομένων της βάσης SKIG.

6.3 Ανιχνευτής δραστηριότητας

Ο παρακάτω αλγόριθμος βασίζεται σε ιδέες που περιγράφονται στα [58, 59]. Η λειτουργία του ανιχνευτή δραστηριότητας βασίζεται στον υπολογισμό οπτικής ροής. Η διαίσθηση στην οποία βασίζεται είναι ότι οι κινήσεις του χρήστη την ώρα που εκτελεί χειρονομίες θα οδηγήσουν σε μεγαλύτερες τιμές οπτικής ροής και κατά συνέπεια μεγαλύτερο *activity score* σε σχέση με τις υπόλοιπες στιγμές. Συνεπώς τα καρέ που περιέχουν δραστηριότητα μπορούν να διαχωριστούν από αυτά που δεν περιέχουν μέσω του ιστογράμματος των activity scores. Για κάποια πρώτα frames (τυπικά περίπου 100) ο activity detector δεν παίρνει "αποφάσεις", προκειμένου να μαζευτούν αρκετά scores εντός του buffer (calibration). Μόλις τελειώσει η φάση του calibration το κατώφλι που προκύπτει θεωρείται "κατώφλι αναφοράς". Στη συνέχεια τα νέα κατώφλια που υπολογίζονται περιορίζονται στο διάστημα μεταξύ 80 και 120 τοις εκατό του κατωφλίου αναφοράς. Για τον υπολογισμό του activity score σε κάθε καρέ i πραγματοποιούνται τα παρακάτω βήματα:

1. Υπολογίζεται το μέτρο της οπτικής ροής μεταξύ των καρέ i και $i-1$.

2. Όλες οι τιμές της οπτικής ροής (δισδιάστατος πίνακας της οπτικής ροής) αθροίζονται, κανονικοποιούνται και προκύπτει ένα "raw" score για το καρέ i. Το score αυτό αποθηκεύεται σε ένα buffer/queue (π.χ. μεγέθους 800 καρέ).

3. Ο buffer φιλτράρεται με median και γκαουσιανό φίλτρο (έστω μεγέθη M και N) για να εξομαλυνθούν τυχόν απότομες, "θορυβώδεις" διακυμάνσεις στα score.

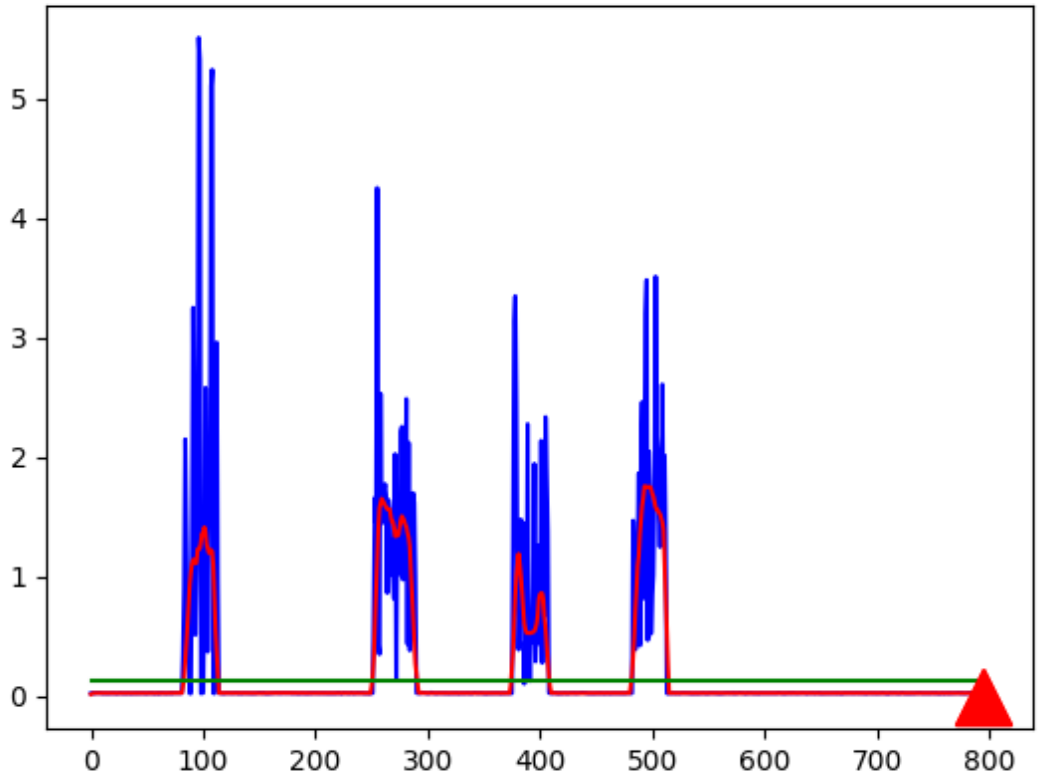
4. Τη χρονική στιγμή i, προκύπτει το τελικό score για το καρέ $i - \max(M, N)/2$ λόγω της συνέλιξης. Αν ξεπερνά το κατώφλι τότε έχουμε activity.

Το κατώφλι ανίχνευσης δραστηριότητας υπολογίζεται (ή ανανεώνεται) σε κάθε frame ως εξής:

1. Υπολογίζεται το ιστόγραμμα των (φιλτραρισμένων) scores του buffer.
2. Εντοπίζεται το score στο οποίο το ιστόγραμμα παρουσιάζει μέγιστο.
3. Το threshold ισούται με το πρώτο τοπικό ελάχιστο μετά το μέγιστο.

Ο ανιχνευτής μπορεί εύκολα να ρυθμιστεί ώστε να παρουσιάζει μικρή ή μεγάλη ευαισθησία σε κινήσεις του χρήση, προσθέτοντας κάποιον όρο στο κατώφλι που θέτει η παραπάνω διαδικασία. Επίσης μπορεί να απορρίπτει κινήσεις μικρής διάρκειας (π.χ λιγότερα από 10 καρέ). Όμως ένα σημαντικό μειονέκτημα του είναι πως δεν μπορεί να διαχωρίσει ενέργειες του χρήστη που εκτελούνται με πολύ μικρή χρονική διαφορά. Εξαιτίας της απλότητας του αντιμετωπίζει 2 χειρονομίες που εκτελούνται η μία αμέσως μετά την άλλη σαν μία. Για αυτό το λόγο, για να εξασφαλισθεί η ομαλή λειτουργία του συστήματος πρέπει ο χρήστης να αναμένει την αναγνώριση μίας χειρονομίας για να συνεχίσει στην επόμενη.

Σε επίπεδο προγραμματιστικής υλοποίησης, ο ανιχνευτής δραστηριότητας είναι ένας κόμβος (node) του περιβάλλοντος του R.O.S. (Robotics Operating System) [25], ο οποίος παρακολουθεί την ροή RGB πληροφορίας που λαμβάνεται από τον αισθητήρα Kinect και ”δημοσιοποιεί” τις χρονικές επισημειώσεις αρχής και τέλους κάθε τμήματος της συνεχούς ροής βίντεο που περιλαμβάνει δραστηριότητα, υπό τη μορφή μηνύματος. Τα μηνύματα αυτά διαβάζονται από έναν άλλο κόμβο, που ονομάζεται *kinect streamer*. Ο kinect streamer κρατά ένα ιστορικό των καρέ που λαμβάνει από το Kinect, μαζί με τις αντίστοιχες χρονικές επισημειώσεις μέσω μίας δομής κυκλικού buffer μίας χωρητικότητας της τάξης των 100-200 καρέ. Όταν αυτός συμπληρώσει την χωρητικότητα του, τα παλαιότερα καρέ διαγράφονται ώστε να μπορούν να εισέλθουν τα πιο πρόσφατα. Όταν ο streamer λάβει πρώτα ένα μήνυμα αρχής (start message) μαζί με την επισημείωση αρχής και στη συνέχεια ένα μήνυμα τέλους (end) μαζί με την επισημείωση τέλους της δραστηριότητας, καλεί τον *recognizer* μία συνάρτηση που λαμβάνει σαν είσοδο μία ακολουθία από καρέ εντός των οποίων υποθέτει πως υπάρχει κάποια εκ των προκαθορισμένων χειρονομιών την οποία και αναγνωρίζει χρησιμοποιώντας κάποιο προεκπαιδευμένο μοντέλο. Τέλος δημοσιεύεται η κλάση που προβλέπει το μοντέλο, αν η εμπιστοσύνη πρόβλεψης είναι πάνω από ένα κατώφλι (τυπικά περίπου 0.80). Για το μοντέλο 3D-CNN-LSTM η σχέση που δίνει την πρόβλεψη καθώς και οι διάφορες λεπτομέρειες που αφορούν την λειτουργία του έχουν παρουσιαστεί στην ενότητα 5.3. Σημειώνεται πάντως πως ο τύπος του μοντέλου που θα χρησιμοποιηθεί δεν είναι δεσμευτικός,



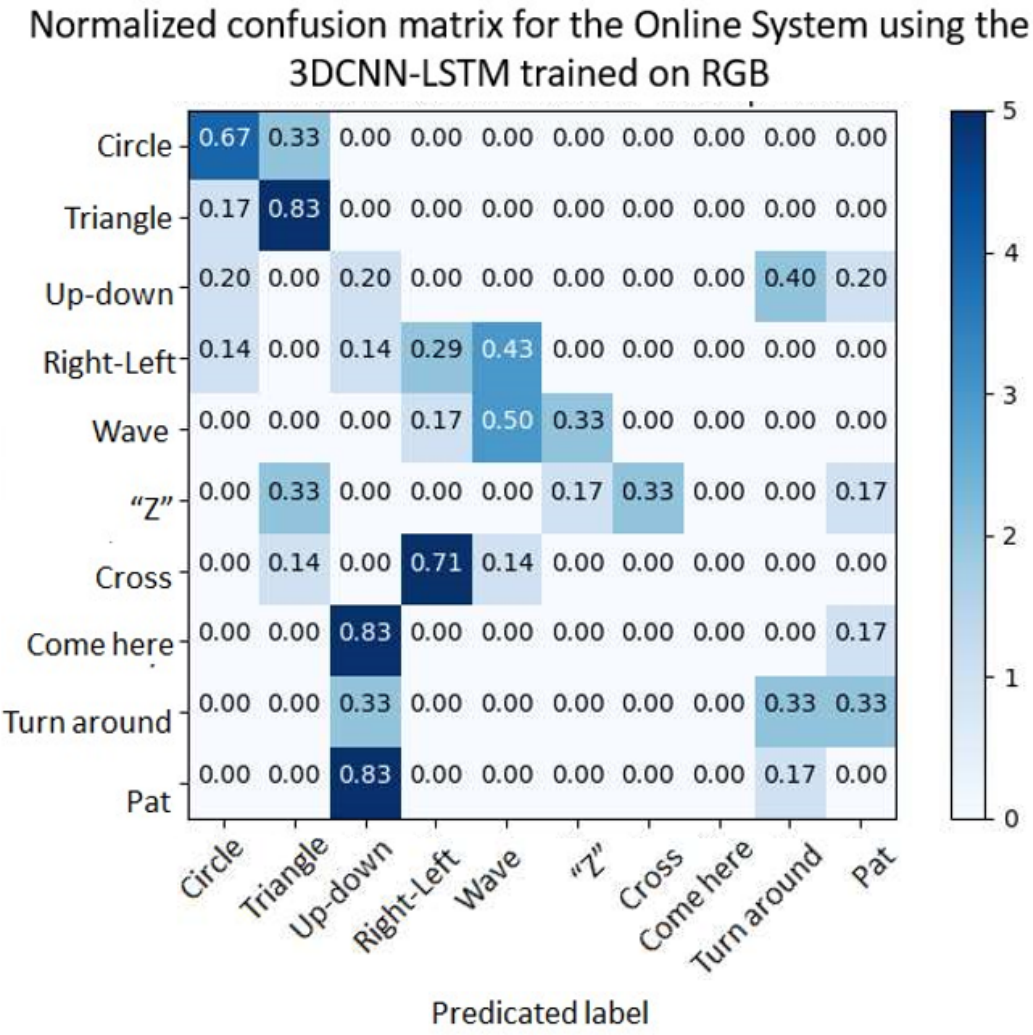
Σχήμα 6.2: Λειτουργία ανιχνευτή δραστηριότητας. Με μπλε γραμμή απεικονίζεται το score δραστηριότητας, με κόκκινη γραμμή απεικονίζεται η φιλτραρισμένη εκδοχή του score δραστηριότητας και με πράσινη απεικονίζεται το κατώφλι δραστηριότητας. Στο συγκεκριμένο σχήμα παρατηρούνται τέσσερις χρονικά εντοπισμένες και διαχωρισμένες χειρονομίες του χρήστη.

γεγονός που επιτρέπει την εύκολη αναπροσαρμογή του συστήματος σε περίπτωση αλλαγής του συνόλου δεδομένων που χρησιμοποιείται ή την ύπαρξη πιο αποτελεσματικού μοντέλου.

6.4 Αξιολόγηση της απόδοσης του συστήματος

Με σκοπό να αξιολογηθεί το σύστημα έγινε προσπάθεια προσομοίωσης, όσο το δυνατόν καλύτερα, των συνθηκών υπό της οποίες λήφθηκαν τα δεδομένα της βάσης SKIG. Συνεπώς η κάμερα Kinect τοποθετήθηκε σε θέση κάθετη ως προς μία επίπεδη λευκή επιφάνεια. Στη συνέχεια εκτελέσθηκαν σε συνεχή ροή 6 επαναλήψεις της κάθε χειρονομίας και αποθηκεύτηκαν οι προβλέψεις του μοντέλου. Στη συνέχεια υπολογίστηκαν τα στοιχεία του πίνακα σύγχυσης που φαίνεται στο Σχήμα 6.3. Παρατηρούμε πως η απόδοση του μοντέλου είναι σημαντικά χαμηλότερη σε σχέση με τα offline πειράματα που εκτελέσθηκαν στο προηγούμενο κεφάλαιο. Παρόλα αυτά το μοντέλο φαίνεται να έχει τη δυνατότητα να αναγνωρίσει αρκετά από

τα παραδείγματα παρά τη σημαντική απόκλιση ανάμεσα στις συνθήκες λήψης των δεδομένων στα οποία εκπαιδεύτηκε και των δεδομένων αξιολόγησης του on-line συστήματος. Σημειώνεται επίσης πως ο ανιχνευτής δραστηριότητας, μετά από ρύθμιση των παραμέτρων ευαισθησίας και προσδιορισμού του κατωφλίου, εντόπιζε πάντα σωστά την ύπαρξη δραστηριότητας στην συνεχή ροή βίντεο. Αυτό είναι σημαντικό διότι στα διαστήματα ανάμεσα στις εκτελέσεις των χειρονομιών το χέρι του χρήστη παραμένει εντός του πλάνου της κάμερας και συνεπώς μικρές κινήσεις του δεν γίνεται να αποφευχθούν.



Σχήμα 6.3: Ο πίνακας σύγχυσης που προέκυψε κατά την αξιολόγηση του on-line συστήματος. Τα παραπάνω αποτελέσματα προέκυψαν για ένα συνεχές βίντεο το οποίο περιείχε 6 εκτελέσεις κάθε χειρονομίας από έναν χρήστη.

Συνεπώς καταλήγουμε στα παρακάτω βασικά συμπεράσματα σχετικά με το σύ-

στημα που σχεδιάστηκε:

- **Δυνατότητα επαναχρησιμοποίησης τμημάτων του συστήματος (modularity):** Το σύστημα που σχεδιάστηκε αποτελείται από τρία ανεξάρτητα τμήματα: τον ανιχνευτή δραστηριότητας, τον streamer και τον recognizer. Κατά την αξιολόγηση του συστήματος φάνηκε πως τα πρώτα δύο τμήματα λειτουργούν αποτελεσματικά. Το τμήμα της αναγνώρισης στηρίζεται αποκλειστικά στην ύπαρξη ενός εύρωστου και κατάλληλα προσαρμοσμένου μοντέλου. Συνεπώς είναι εφικτό πολύ εύκολα να βελτιωθεί το σύστημα με τη δημιουργία και εκπαίδευση ενός καλύτερου ισχυρότερου μοντέλου
- **Υψηλή ταχύτητα υπολογισμού χρησιμοποιώντας ένα βαθύ νευρωνικό δίκτυο:** Σε αντίθεση με συστήματα που στηρίζονται σε ”Hand-Crafted” χαρακτηριστικά, βασιζόμενο στην ύπαρξη μίας φτηνής και μέτριων δυνατοτήτων κάρτας γραφικών, το σύστημα εκτελεί προβλέψεις για βίντεο 64 καρέ σε κλάσματα του δευτερολέπτου. Συστήματα που βασίζονται στην πολύ αποτελεσματική μέθοδο των πυκνών τροχιών και κάποιου Support Vector Machines ταξινομητή, χρειάζονται πολλαπλάσιο του δευτερολέπτου χρόνο.

Κεφάλαιο 7

Συμπεράσματα και Μελλοντικές Κατευθύνσεις

7.1 Συνεισφορά της διπλωματικής εργασίας

Στα πλαίσια αυτής της διπλωματικής, επικεντρωθήκαμε στην εφαρμογή των νευρωνικών δικτύων στην αναγνώριση ανθρωπίνων δράσεων και χειρονομιών. Εστιάσαμε σε βάσεις δεδομένων μεσαίας προς μικρής κλίμακας ($1000 - 1500$ παραδείγματα για εκπαίδευση και $700 - 1000$ παραδείγματα αξιολόγησης) που περιείχαν απλές και σύντομες στην εκτέλεση τους δράσεις και χειρονομίες όπως αυτές των βάσεων KTH και SKIG, αντίστοιχα. Ξεκινώντας από ένα απλό Νευρωνικό Δίκτυο Τρισδιάστατης Συνέλιξης (3D-CNN) εξετάσαμε τη δυνατότητα εκπαίδευσης ρηχών και σχετικά μικρών μοντέλων (4 στρωμάτων και περίπου $3.000.000$ παραμέτρων). Σταδιακά αποκτώντας εξοικείωση με τη διαδικασία εκπαίδευσης, προεκταθήκαμε σε ένα σημαντικά βαθύτερο και μεγαλύτερο προεκπαιδευμένο μοντέλο, το C3D, του οποίου τις παραμέτρους προσαρμόσαμε στα προβλήματα των βάσεων KTH και SKIG μέσω της επανεκπαίδευσης του μοντέλου (Σχήμα 5.6). Μέχρι αυτό το σημείο πετύχαμε ικανοποιητικά αποτελέσματα (Σχήμα 5.1) αλλά και αναδείχθηκε η αδυναμία πλήρους προσαρμογής (finetuning) πολύ μεγάλων μοντέλων σε νέα προβλήματα, απουσία μεγάλου πλήθους δεδομένων. Επίσης φάνηκε πως τα τοπικά χωροχρονικά χαρακτηριστικά δεν επαρκούν για την αποτελεσματική και εύρωστη μοντελοποίηση σύντομων δράσεων και χειρονομιών.

Στη συνέχεια, πειραματιστήκαμε με δύο Αναδρομικά Νευρωνικά Δίκτυα που αποτελούνται από στρώματα νευρώνων Μακράς και Βραχείας Μνήμης, εκ των οποίων το πρώτο (3D-CNN-LSTM) χρησιμοποιεί τα τοπικά χωροχρονικά χαρακτηριστικά που εξάγει ένα Νευρωνικό Δίκτυο Τρισδιάστατης Συνέλιξης από τμήματα ενός βίντεο και το δεύτερο (CNN-LSTM) χρησιμοποιεί τα χωρικά χαρακτηριστικά που εξάγει ένα Νευρωνικό Δίκτυο Δισδιάστατης Συνέλιξης (2D-CNN ή απλά CNN) από κάθε καρέ ενός βίντεο. Η λειτουργία αυτών των μοντέλων επεκτείνει την ικανότητα των συνελικτικών δικτύων να εξάγουν τοπικά χαρακτηριστικά (χωρικά ή χωροχρονικά) από τα βίντεο, προσθέτοντας συνολική χρονική μοντελοποίηση μέσω του

στρώματος LSTM νευρώνων. Φάνηκε πως τα πιο σύνθετα αυτά μοντέλα είναι αποτελεσματικότερα στην επίλυση των προβλημάτων των δύο βάσεων (5.2), δίνοντας μέχρι και 90% μέση ακρίβεια στη βάση SKIG. Επιπλέον, ο πειραματισμός με τη βάση δεδομένων SKIG μας έδωσε τη δυνατότητα να εξετάσουμε μία απλή μέθοδο σύμμειξης τροπικοτήτων, όπου οι προβλέψεις των μοντέλων της κάθε τροπικότητας συνδυάζονται. Με αυτόν τον τρόπο η ακρίβεια, του μοντέλου 3D-CNN-LSTM, επί του συνόλου αξιολόγησης, βελτιώθηκε. Όπως είναι αναμενόμενο, λόγω της μη κυρτής επιφάνειας κόστους που εμφανίζεται στα βαθιά νευρωνικά δίκτυα, παρουσιάστηκαν συχνά δυσκολίες στην εκπαίδευση των μοντέλων. Επιβεβαιώθηκε η σημασία τεχνικών κανονικοποίησης όπως η μέθοδος Dropout, Batch Normalization. Μέσω των δύο αυτών τεχνικών βελτιώνεται σημαντικά το αποτέλεσμα της διαδικασίας εκπαίδευσης όσον αφορά τον αριθμό εποχών που απαιτούνται μέχρι αυτή να συγκλίνει για όλα τα μοντέλα. Επιβεβαιώθηκε η δυνατότητα της επαύξησης δεδομένων να αντισταθμίζει την έλλειψη μεγάλου αριθμού παραδειγμάτων. Όπως περιγράφεται πιο αναλυτικά στο Κεφάλαιο 5, η χρήση off-line (δηλαδή πριν την έναρξη της εκπαίδευσης) και on-line (δηλαδή κατά τη διάρκεια της εκπαίδευσης) επαύξησης δεδομένων αυξάνει σημαντικά τη μέση ακρίβεια των μοντέλων επί των παραδειγμάτων αξιολόγησης.

Ολοκληρώνοντας, το πειραματικό μέρος της εργασίας, υλοποιήθηκε ένα σύστημα on-line αναγνώρισης χειρονομιών στο οποίο ενσωματώθηκε το εκπαιδευμένο επί της βάσης SKIG, μοντέλο 3D CNN-LSTM. Το σύστημα αναπτύχθηκε εντός του περιβάλλοντος του R.O.S (Robotics Operating System) και επιτρέπει ταχύτερη αναγνώριση (της τάξης μεγέθους των εκατοντάδων milisecond ανά χειρονομία) σε σχέση με άλλες κλασσικές μεθόδους αναγνώρισης που βασίζονται στην εξαγωγή κατασκευασμένων χαρακτηριστικών. Δεδομένου του πεπερασμένου κάθε διπλωματικής εργασίας, δεν υπήρξε ο χρόνος να μελετήσουμε και να ακολουθήσουμε πολλές από τις ερευνητικές κατευθύνσεις που θα επιθυμούσαμε, ορισμένες από τις οποίες καταγράφουμε στην επόμενη ενότητα.

7.2 Μελλοντικές Κατευθύνσεις

- **Δυνατότητα προσαρμογής του προεκπαιδευμένου μοντέλου σε νέες συνθήκες:** Με βάση την τριβή που αναπτύχθηκε με την αναγνώριση χειρονομιών εκτιμάται πως η απόδοση του on-line συστήματος θα βελτιωθεί αν συλλεχθεί ένα μέτριου μεγέθους σύνολο παραδειγμάτων (π.χ 400 παραδείγματα από 4 άτομα) υπό τις πραγματικές συνθήκες λειτουργίας (δηλαδή στο χώρο που θα λειτουργεί το σύστημα on-line αναγνώρισης) και είτε προσαρμοστεί το προεκπαιδευμένο 3D-CNN-LSTM μοντέλο μέσω finetuning είτε εκπαιδευτεί από την αρχή το μοντέλο 3D-CNN-LSTM χρησιμοποιώντας τα ήδη υπάρχοντα και τα νέα δεδομένα. Επίσης θα μπορούσε να επιδιωχθεί να δοθεί επιπλέον βα-

ρύτητα στα παραδείγματα που αντιπροσωπεύουν καλύτερα τις πραγματικές συνθήκες λειτουργίας σε σχέση με αυτά της βάσης SKIG. Αυτό μπορεί να επιτευχθεί μέσω της προσθήκης κάποιου όρου βαρύτητας στη συνάρτηση κόστους που να επιτρέπει σημαντικότερες αλλαγές των παραμέτρων κατά την εκπαίδευση επί των νέων παραδειγμάτων.

- **Πειραματισμός με διαφορετικούς τρόπου σύμμειξης τροπικοτήτων οπτικής πληροφορίας:** Θεωρούμε πως ο τρόπος σύμμειξης της πληροφορίας χρήζει, περαιτέρω διερεύνησης τόσο στο συνελικτικό όσο και το αναδρομικό τμήμα των μοντέλων που υλοποιήθηκαν. Μία σχετική εργασία που κινείται σε αυτή την κατεύθυνση παρουσιάζεται στην αναφορά [26]. Στην εργασία αυτή συνενώνονται τα συνελικτικά χαρακτηριστικά που έχουν εξαχθεί από το βίντεο βάθους με αυτά του βίντεο RGB και στη συνέχεια τροφοδοτούνται σε ένα αναδρομικό δίκτυο.

- **Εφαρμογή των μοντέλων που εκπαιδεύτηκαν σε πιο απαιτητικές βάσεις:** Θεωρούμε πως θα ήταν χρήσιμο να δοκιμαστούν οι αρχιτεκτονικές των μοντέλων που μελετήθηκαν σε βάσεις πιο μεγάλης κλίμακας ή βάσεις που παρουσιάζουν μεγαλύτερες διακύμανσης γωνίας λήψης, οπτικής κλίμακας, φωτισμού και χρονισμού εκτέλεσης των δράσεων. Ειδικότερα, αν τέτοιες βάσεις δεδομένων έχουν μεσαίο προς μικρό μέγεθος, ενδέχεται να απαιτηθεί μεγαλύτερο εύρος χωρικών και χρονικών μετασχηματισμών για την επαύξηση του συνόλου δεδομένων.

- **Χρήση μηχανισμού προσοχής:** Η χρήση ενός μηχανισμού προσοχής (attention mechanism) θα μπορούσε να προστεθεί στην αρχιτεκτονική του δικτύου θα μπορούσε προσθέσει στο μοντέλο την ικανότητα να μάθει, να δίνει έμφαση σε περιοχές της εισόδου που περιέχουν σημαντική πληροφορία για την δράση που εκτελείται. Μία τέτοια προσέγγιση έχει φανεί πως λειτουργεί αποτελεσματικά και βελτιώνει τα αποτελέσματα αναγνώρισης δράσης στην αναφορά [47]. Ένας μηχανισμός προσοχής μπορεί να είναι είτε ντετερμινιστικός είτε στοχαστικός όπως περιγράφεται στην αναφορά [44]. Στην πρώτη περίπτωση είναι δυνατόν ο μηχανισμός προσοχής να εκπαιδευτεί παράλληλα με το μοντέλο μέσω του αλγορίθμου Back Propagation, προσθέτοντας ουσιαστικά κάποιους κόμβους στον υπολογιστικό γράφο του δικτύου. Στην δεύτερη περίπτωση ο μηχανισμός μπορεί να εκπαιδευτεί μέσω του αλγορίθμου REINFORCE ([33]). Εκτός της βελτίωσης της επίδοσης, ένας μηχανισμός προσοχής μπορεί να χρησιμοποιηθεί ώστε να κατανοηθεί καλύτερα σε ποια κομμάτια της σκηνής ή σε ποιες κινήσεις, το μοντέλο εστιάζει, καθώς και γιατί οδηγείται σε λάθος πρόβλεψες.

- **Μάθηση με χρήση προγράμματος (curriculum learning):** Θα μπορούσε να εξετασθεί αν υπάρχει κάποια μέθοδος αξιολόγησης των παραδειγμάτων (ranking) εκπαίδευσης ώστε να ταξινομηθούν ανάλογα με την δυσκολία του συστήμα-

τος να μάθει από αυτά να αναγνωρίζει παρόμοιο περιεχόμενο. Αν υπάρχει ένας τρόπος να συμβεί αυτό, θα μπορούσαμε να εισάγουμε ένα πρόγραμμα αυξανόμενης δυσκολίας στη διαδικασία εκπαίδευσης ή συνδυασμού σε κάθε minibatch τόσο δύσκολων όσο και εύκολων παραδειγμάτων. Ο προγραμματισμός αυτός μπορεί να βελτιώσει και να επιταχύνει την διαδικασία εκπαίδευσης [40, 42].

Κλείνοντας αυτήν την ενότητα, μπορούμε να πούμε πως οι εντυπωσιακοί ρυθμοί της εξέλιξης της Όρασης Υπολογιστών και της Βαθιάς Μάθησης, μας επιτρέπει να αισιοδοξούμε πως πράγματα που μέχρι πριν μερικά χρόνια φάνταζαν δυσεπίλυτα, πλέον δεν απέχουν πολύ από το να είναι εφικτά. Επίσης, ευχόμαστε αυτή η διπλωματική εργασία, να βοηθήσει, έστω και στο ελάχιστο, κάποιον αναγνώστη να κατανοήσει τις θεματικές περιοχές τις οποίες πραγματεύεται.

Βιβλιογραφία

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, ”Imagenet classification with deep convolutional neural networks,” in *Proc. Advances in Neural Information Processes Systems, (NIPS)*, 2012, pp. 1097-1105.

[2] J. Donahue, et al., ”Long-term recurrent convolutional networks for visual recognition and description,” in *Proc. of the IEEE Conf. on Comput. Vision and Pattern Recognition, (CVPR)*, 2015, pp. 2625-2634.

[3] D. Tran et al, ”Learning spatiotemporal features with 3d convolutional networks,” in *Proc IEEE Int. Conf. on Comput. Vision, (ICCV)*, 2015, pp. 4489-4497.

[4] K. Simonyan and A. Zisserman, ”Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[5] S. Linnainmaa, ”The representation of the cumulative rounding error of an algorithm as a Taylor expansion of the local rounding errors,” M.S. thesis, Helsinki Univ., Helsinki, Finland, 1970.

[6] S. Linnainmaa, ”Taylor expansion of the accumulated rounding error,” *BIT Numerical Mathematics*, vol. 16(2), 1976, pp. 146-160,.

[7] I. Laptev et al, “Learning Realistic Human Actions from Movies,” *in Proc. IEEE Conf. on Comput. Vision and Pattern Recognition, (CVPR)*, Jun. 2008, pp. 1–8.

[8] I. Laptev. ”Local spatio-temporal image features for motion interpretation,” Ph.D. dissertation, Dept. Numerical Analysis and Comput. Science (NADA), KTH, Stockholm, 2004.

[9] C. Harris and M. Stephens, “A combined corner and edge detector,” in *Proc. Alvey Vision Conference*, Manchester, 1988, pp. 147–152.

[10] P. Dollar et al, ”Behavior recognition via sparse spatio-temporal features,” in *Proc. Workshop on Performance Evaluation of Tracking and Surveillance (VS-PETS)*, 2005.

[11] L. Liu and L. Shao, ”Learning discriminative representations from RGB-D video data,” in *Proc. 23rd Int. Joint Conf. on Artificial Intell.*, 2013, pp. 1493-1500.

[12] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley and Y. Bengio, “Theano: A CPU and GPU Math Expression Compiler,” in *Proc. Python for Scientific Computing Conf. (SciPy)*, June 30 - July 3, Austin, TX, 2010.

[13] S. Dieleman, J. Schlüter, C. Raffel, E Olson, S. K. Sønderby, D. Nouri, D. Maturana, M. Thoma, E. Battenberg, J. Kelly, J. De Fauw, M. Heilman, diogo149, B. McFee, H. Weideman, takacsg84, peterderivaz, Jon, instagibbs, K. Rasul, CongLiu, Britefury, J. Degrave, ”Lasagne: First release,” August 2015, 0.5281/zenodo.27878 http://dx.doi.org/10.5281/zenodo.27878

[14] S. Chetlur, C. Woolley, P. Vandermersch, J. Cohen and J. Tran ”cuDNN: Efficient Primitives for Deep Learning”, arXiv:1410.0759v3, 18 Dec, 2014.

[15] B.Widrow, and M. Hoff, ”Associative storage and retrieval of digital information in networks of adaptive neurons,” *Biological Prototypes and Synthetic Systems*, vol. 1, 1962, pp. 160.

[16] T. Kohonen, ”Correlation matrix memories,” *IEEE Transactions on Comput.*, vol. 100(4), 1972, pp. 353–359.

[17] S. Ji, W. Xu, M. Yang and K. Yu, ”3D Convolutional Neural Networks for Human Action Recognition ,” in *IEEE Transactions Pattern Anal. Mach. Intell., (PAMI)*, vol. 35(1), 2013, pp. 221–231 .

[18]] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. ”Large-scale video classification with convolutional neural networks.” in *Proc. of the IEEE Conf. on Comput. Vision and Pattern Recognition, (CVPR)*, 2014.

[19] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, ”Gradient-based learning applied to document recognition,” in *Proc. of the IEEE,* vol. 86(11), 1998, pp. 2278–2324.

[20] M. Baccouche, F. Mamalet, C. Wolf, C. Garcia, and A. Baskurt, ”Action classification in soccer videos with long short-term memory recurrent neural networks”, in *Proc. Int. Conf. on Artificial Neural Networks (ICANN)*, 2010, pp. 154-159.

[21] L. Bottou, ”Curiously Fast Convergence of some Stochastic Gradient Descent Algorithms,” in *Proc. of the Symp. on Learning and Data Science*, 2009.

[22] K. Mikolajczyk, C. Schmid A. Heyden and G. Sparr and M. Nielsen and P. Johansen, ”An affine invariant interest point detector”, in *Proc. European Conf. on Comput. Vision (ECCV)*, Copenhagen, Denmark, May 2002.

[23] G. W. Taylor, R. Fergus, Y. LeCun, and C. Bregler, “Convolutional learning of spatio-temporal features,” in*European Conf. on Comput. Vision (ECCV)*, 2010, Springer, pp. 140–153.

[24] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, “Backpropagation applied to handwritten zip code recognition,” in *Neural Computation*, vol. 1, no. 4, 1989, pp. 541–551.

[25] M. Quigley and B. Gerkey and K. Conley and J. Faust and T. Foote and J. Leibs and E. Berger and R. Wheeler and A. Ng, ”ROS: an open-source Robot Operating System”, in *Proc. of the IEEE Intl. Conf. on Robotics and Automation (ICRA) Workshop on Open Source Robotics*, Kobe, Japan, 2009.

[26] N. Nishida and H. Nakayama, ”Multimodal Gesture Recognition using Multi-stream Recurrent Neural Network,” in *7th Pacific Rim Symposium on Image and Video Technology (PSIVT)*, 2015, pp. 1–14.

[27] S. Ioffe and C. Szegedy, ”Batch normalization: Accelerating deep network training by reducing internal covariate shift”, in *Int. Conf. on Machin. Learning (ICML)*, 2015.

[28] Y. Nesterov, ”A method for unconstrained convex minimization problem with the rate of convergence o $(1/k^2)$” in *Doklady an SSSR*, vol 269, 1983, pp. 543–547.

[29] I. Sutskever, J. Martens, G. Dahl and G. Hinton, ”On the importance of initialization and momentum in deep learning,” in in *Int. Conf. on Machin. Learning (ICML)*, 2013.

[30] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever and R. Salakhutdinov, ”Dropout: A Simple Way to Prevent Neural Networks from Overfitting” , in *Journal of Machine Learning Research (JMLR)*, vol. 15, 2014, pp. 1929-1958.

[31] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, Y. LeCun, ”The Loss Surfaces of Multilayer Networks,” arXiv:1412.0233, 2015.

[32] D. E. Rumelhart, G. E. Hinton, R. J.Williams, ”Learning representations by back-propagating errors”, in *Nature*, vol. 323, 1986, pp. 533-536.

[33] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, ”Recurrent models of visual attention,” in *Proc. Advances in Neural Information Processes Systems, (NIPS)*, 2014, pp. 2204–2212.

[34] Y. LeCun, L. Bottou, G. Orr, K. Müller, ”Neural networks: Tricks of the trade”, 2012, pp. 9-48, Springer,.

[35] A. Graves, ”Supervised Sequence Labelling with Recurrent Neural Networks,” Ph.D thesis, Dept. of Informatics, Technical Univ. Munich, Munich, 2014.

[36] A. Graves, S. Fernández, F. Gomez, J. Schmidhuber, ”Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks”, in *Proc. 23rd Int. Conf. on Machine learning (ICML)*, Pittsburgh, Pennsylvania, USA, June 25 - 29, 2006.

[37] C. Olah. (2015, Aug. 27). *Understanding-LSTMs*. [Online]. Available:http://colah.github.io/posts/2015-08-Understanding-LSTMs/.

[38] A. B. Sargano, P. Angelov and Z. Habib, ”A Comprehensive Review on Handcrafted and Learning-Based Action Representation Approaches for Human Activity Recognition”, in *Appled Sciences*, 23 Jan, 2017.

[39] I. Sutskever, O. Vinyals and Q. V. Le, ”Sequence to Sequence Learning with Neural Networks,” arXiv:1409.3215, 2014.

[40] Y. Bengio, J.Louradour, R. Collobert and J. Weston, ”Curriculum learning.” in *Int. Conf. on Machine learning (ICML)*, 2009.

[41] Wilson D., Martinez T., ”The general inefficiency of batch training for gradient descent learning”, Neural Networks vol. 16(10), 2003, pp. 1429-51.

[42] A. Graves, M. G. Bellemare, J. Menick, R. Munos and K. Kavukcuoglu, ”Automated Curriculum Learning for Neural Networks”, arXiv:1704.03003v1, 2017.

[43] Y. Dauphin, R. Pascanu, C. Gulcehre, K. Cho, S. Ganguli and Y. Bengio ”Identifying and attacking the saddle point problem in high-dimensional non-convex optimization”, in *Advances in Neural Information Processing Systems,* 2014, pp. 2933-2941.

[44] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, ”Show and tell: A neural image caption generator.” in *Conf. on Comput. Vision and Pattern Recognition,*, 2015.

[45] L. Bottou, ”Large-Scale Machine Learning with Stochastic Gradient Descent” in *Proc. of Int. Conf. on Computational Statistics (COMPSTAT)*, 2010, pp. 177-186.

[46] L. Bottou, O. Bousquet ”The Tradeoffs of Large Scale Learning”, in *Proc. of the 20th Int. Conf. on Neural Information Processing Systems (NIPS)*, 2007, pp. 161-168.

[47] S. Sharma, R. Kiros, and R. Salakhutdinov, ”Action recognition using visual attention,” arXiv:1511.04119, 2015.

[48] L. Bottou, ”Stochastic Gradient Descent Tricks”, In K.-R. M¨uller, G. Montavon, and G. B. Orr, editors, *Neural Networks: Tricks of the Trade, Reloaded*. Springer , 2013.

[49] P. J. Werbos, ”Backpropagation Through Time: What It Does and How to Do It”, in *Proc. of the IEEE*, vol. 78(10), 1988, pp. 1550 – 1560.

[50] A. Krizhevsky, I. Sutskever and G. E. Hinton, ”Imagenet classification with deep convolutional neural networks”, in *Advances in Neural Information Processing systems*, 2012, pp. 1097-1105.

[51] B.T. Polyak, ”Some methods of speeding up the convergence of iteration methods,” in *Computational Mathematics and Mathematical Physics*, vol. 4, 1962, pp. 1–17.

[52] H. Wang, A. Klaser, C. Schmid, L. C. Lin., ”Action Recognition by Dense Trajectories.”, in *IEEE Conference on Computer Vision Pattern Recognition*, pp.3169-3176., Colorado Springs, United States, Jun 2011.

[53] K. Fukushima, “Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position,” in *Biological Cybernetics*, vol. 36, 1980, pp. 193–202.

[54] A. Graves, ”Generating Sequences With Recurrent Neural Networks,” arXiv preprint arXiv:1308.0850, 2013.

[55] K. Greff , R. K Srivastava, J. K. , B. R. Steunebrink , J. Schmidhuber, ”LSTM: A Search Space Odyssey, ” arXiv:1503.04069v1, 2015.

[56] P. Molchanov, X. Yang, S. Gupta, K. Kim, S. Tyree an J. Kautz, ”Online Detection and Classification of Dynamic Hand Gestures with recurrent 3D Convolutional Neural Networks”, in *IEEE Conf. on Comput. Vision and Pattern Recognition, (CVPR)*, 2016.

[57] G. Canal, S. Escalera, C. Angulo, ”A Real-time Human-Robot Interaction system based on gestures for assistive scenarios,” in *Comput. Vision Image Understand.*, vol. 149, 2016, pp 391-406.

[58] N. Kardaris, I. Rodomagoulakis, V. Pitsikalis, A. Arvanitakis and P. Maragos , ”A Platform for Building New Human-Computer Interface Systems that Support Online Automatic Recognition of Audio-Gestural Commands”, in *Proc. of the ACM on Multimedia Conference,* 2016, pp. 1169-1173.

[59] I. Rodomagoulakis , N. Kardaris , V. Pitsikalis , E. Mavroudi , A. Katsamanis , A. Tsiami and P. Maragos, ”Multimodal Human Action Recognition in Assistive Human-Robot Interaction”, in *(in Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 2702–2706.

[60] E. Ohn-Bar and M. Trivedi, ”Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations” in *IEEE Trans. on Intelligent Transportation Systems*, vol. 15(6), 2014, pp. 1–10.

[61] N. Neverova, C. Wolf, G. W. Taylor, and F. Nebout. ”Multiscale deep learning for gesture detection and localization.” in in *Proc. European Conf. on Comput. Vision (ECCV)*, 2014.

[62] S. Hochreiter and S.Schmidhuber, ”Long short-term memory,” in *Neural Computation*, vol. 9(8), pp 1735–1780, 1997.

[63] X. Glorot, and Y. Bengio, ”Understanding the difficulty of training deep feedforward neural networks,” in *Proc. of Artificial Intelligence and Statistics (AISTATS)*, vol. 9, 2010, pp. 249–256.

[64] R. Pascanu, T. Mikolov, and Y. Bengio. ”On the difficulty of training recurrent neural networks.” in *Int. Conf. on Machin. Learning (ICML)*, 2013.

[65] F. Rosenblatt, ”The perceptron: a probabilistic model for information storage and organization in the brain.”, in *Psychological review* vol. 65, 1958, pp. 6.

[66] W. S. MuCulloch and W. Pitts, ”A logical calculus of the ideas immanent in nervous activity,” in *The bulletin of math. biophysics*, vol. 5(4), 1943, pp. 115–133.

[67] D. C V. Essen and J. L.Gallant, ”Neural Mechanism of Form and Motion Processing in the Primate Visual System”, in *Neuron*, vol. 13, 1994, pp. 1-10.

[68] A. Karpathy, ”Stanford CS class CS231n: Convolutional Neural Networks for Visual Recognition, class notes 2017”, [Online]. Available:http://cs231n.github.io/

[69] I. Goodfellow, Y. Bengio and A. Courville, ”Deep Learning”, MIT Press, 2016.

[70] M. Gürbüzbalaban, A. Ozdaglar, P. Parrilo, ”Why Random Reshuffling Beats Stochastic Gradient Descent,” arXiv preprint arXiv:1510.08560, 2015.