Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων
τομεας σηματων, ελεγχου και ρομποτικης

Visual Engagement Estimation and Gaze Tracking on Child Interactions

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικόλαου Μπενέτου

Επιβλέπων: Πέτρος Μαραγκός
Καθηγητής Ε.Μ.Π.

Συνεπιβλέπουσα: Νίκη Ευθυμίου
Μεταδιδακτορική Ερευνήτρια Ε.Μ.Π.

ΕΡΓΑΣΤΗΡΙΟ ΟΡΑΣΗΣ ΥΠΟΛΟΓΙΣΤΩΝ, ΕΠΙΚΟΙΝΩΝΙΑΣ ΛΟΓΟΥ ΚΑΙ ΕΠΕΞΕΡΓΑΣΙΑΣ ΣΗΜΑΤΩΝ

Αθήνα, Οκτώβριος 2024

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Σημάτων, Ελέγχου και Ρομποτικής
Εργαστήριο Όρασης Υπολογιστών, Επικοινωνίας Λόγου και Επεξεργασίας Σημάτων

Visual Engagement Estimation and Gaze Tracking on Child Interactions

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικόλαου Μπενέτου

Επιβλέπων: Πέτρος Μαραγκός
Καθηγητής Ε.Μ.Π.

Συνεπιβλέπουσα: Νίκη Ευθυμίου
Μεταδιδακτορική Ερευνήτρια Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 24η Οκτωβρίου, 2024.

.....	.....	.....
Πέτρος Μαραγκός	Γεράσιμος Ποταμιάνος	Ιωάννης Κορδώνης
Καθηγητής Ε.Μ.Π.	Αναπληρωτής Καθηγητής Π.Θ.	Επίκουρος Καθηγητής Ε.Μ.Π.

Αθήνα, Οκτώβριος 2024

.....

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

Στην οικογένειά μου

Περίληψη

Η Όραση Υπολογιστών είναι ένας τομέας της Τεχνητής Νοημοσύνης που στοχεύει στο να δώσει τη δυνατότητα στα υπολογιστικά συστήματα να αντλούν σημαντικές πληροφορίες από οπτικές εισόδους, όπως εικόνες και βίντεο, προκειμένου να λαμβάνουν αποφάσεις ή να κάνουν συστάσεις βάση αυτών. Έχει εφαρμογές σε πολλούς τομείς και βιομηχανίες, όπως οι μεταφορές, η εκπαίδευση, η υγειονομική περίθαλψη, τα συστήματα ασφαλείας και τα κινητά τηλέφωνα, για αυτόν τον λόγο η εξέλιξή της θεωρείται απαραίτητη. Ένας άλλος τομέας στον οποίο έχει διεισδύσει η υπολογιστική όραση είναι η μελέτη των αλληλεπιδράσεων των παιδιών μέσω της ανάλυσης βιντεοδεδομένων, που είναι κρίσιμη για την κατανόηση του πώς τα παιδιά αναπτύσσουν κοινωνικές, συναισθηματικές και γνωστικές δεξιότητες, ιδιαίτερα στην πρώιμη παιδική ηλικία.

Κλειδί σε αυτές τις αλληλεπιδράσεις είναι η στοχοπροσήλωση του παιδιού, η οποία αποτελεί έγκυρο δείκτη του πόσο εστιασμένο είναι το παιδί κατά τη διάρκεια της αλληλεπίδρασης και πώς διαφορετικές προσεγγίσεις αλλάζουν τη συμπεριφορά του. Η στοχοπροσήλωση από οπτική σκοπιά συχνά μετράται μέσω συμπεριφορών όπως το βλέμμα, το οποίο υποδεικνύει προσοχή και συγκέντρωση. Η κατανόηση της στοχοπροσήλωσης του παιδιού είναι κρίσιμη για την αξιολόγηση της μάθησής του, των κοινωνικών αλληλεπιδράσεων και της γνωστικής του ανάπτυξης. Η εκτίμηση του βλέμματος, ως βασική πτυχή της μη λεκτικής επικοινωνίας, παίζει καθοριστικό ρόλο στην ερμηνεία των επιπέδων προσοχής και στην αναγνώριση άτυπων συμπεριφορών, όπως αυτές που παρατηρούνται σε παιδιά με Διαταραχή Αυτιστικού Φάσματος (ΔΑΦ).

Αυτή η διπλωματική εργασία αξιοποιεί προηγμένες τεχνικές όρασης υπολογιστών για την αυτόματη παρακολούθηση και αξιολόγηση των επιπέδων βλέμματος και στοχοπροσήλωσης των παιδιών. Επιπλέον, μελετά την αποτελεσματικότητα των Δικτύων Τμηματικής Χρονικότητας (Temporal Segment Networks - TSN) στην βελτίωση των δυνατοτήτων των προτεινόμενων μοντέλων για αυτά τα καθήκοντα. Δημιουργούμε και εφαρμόζουμε πλαίσια τόσο για την εκτίμηση στοχοπροσήλωσης όσο και για την παρακολούθηση του βλέμματος και τα δοκιμάζουμε με μόνο με τη χρήση δεδομένων RGB, από πολύ απαιτητικά σύνολα δεδομένων που καταγράφουν παιδιά να αλληλεπιδρούν ελεύθερα σε διαφορετικά περιβάλλοντα. Τέλος, δημιουργούμε ένα διαφορετικό είδος δικτύου που συνδυάζει τα δίκτυα εκτίμησης στοχοπροσήλωσης και βλέμματος, ενώ χρησιμοποιεί τη μέθοδο TSN, σε μια νέα αρχιτεκτονική για περαιτέρω βελτίωση της εκτίμησης της στοχοπροσήλωσης.

Χρησιμοποιώντας μοντέλα μηχανικής μάθησης, η διπλωματική εργασία στοχεύει στην παροχή ακριβούς και κλιμακούμενης λύσης για την ανάλυση αυτών των συμπεριφορών σε πραγματικό χρόνο. Τα ευρήματα συμβάλλουν στη διευρυνόμενη έρευνα για την ανάπτυξη στην πρώιμη παιδική ηλικία, προσφέροντας ενδεχομένως εργαλεία για επαγγελματίες που επιθυμούν να κατανοήσουν καλύτερα και να υποστηρίξουν τις αναπτυξιακές ανάγκες των παιδιών, ιδιαίτερα στην αναγνώριση πρώιμων ενδείξεων διαταραχών όπως η ΔΑΦ.

Λέξεις-κλειδιά — Όραση Υπολογιστών, Μηχανική Μάθηση, Στοχοπροσήλωση, Παρακολούθηση Βλέμματος, TSN, Συνελικτικά Δίκτυα

Abstract

Computer Vision is a field of Artificial Intelligence that aims in enabling computer systems to derive important information from visual inputs, such as images and videos, in order to take actions or make recommendations based on them. It has had applications on a variety of areas and industries, like transportation, education, healthcare, security systems and smartphones, for that reason it's advancement is deemed necessary. Another area where computer vision has submerged is in studying children interactions through the analysis of video data, which is crucial for understanding how children develop social, emotional, and cognitive skills, particularly in early childhood. Key to these interactions is the child's engagement, which is a valid indicator of how focused the child is during its interaction and how different approaches change its behaviour. Engagement from a visual perspective is often measured through behaviors like gaze, which indicates attention and focus. Understanding child engagement is crucial for assessing their learning, social interactions, and cognitive development. Gaze estimation, as a key aspect of non-verbal communication, plays a pivotal role in interpreting attention levels and identifying atypical behavioral patterns, such as those seen in children with Autism Spectrum Disorder (ASD).

This thesis leverages state-of-the-art computer vision techniques to automatically track and assess children's gaze and engagement levels. In addition to that, it studies the effectiveness of Temporal Segment Networks (TSN), in improving the capabilities of the proposed models in those tasks. We create and implement frameworks for both the engagement estimation and gaze tracking and test them with visual data using only RGB data as input, from really challenging datasets that capture children interacting freely in different environments. Finally, we create a different kind of network that combines the engagement and gaze estimation networks, while using the TSN method, into a new architecture to further improve the engagement estimation.

By utilizing machine learning models, the thesis aims to provide an accurate and scalable solution for real-time analysis of these behaviors. The findings contribute to the growing body of research in early childhood development, offering potential tools for professionals to better understand and support children's developmental needs, particularly in identifying early signs of disorders such as ASD.

Keywords — Computer Vision, Machine Learning, Engagement, Gaze tracking, TSN, Convolutional Networks

Ευχαριστίες

Η παρούσα διπλωματική έρευνα μου έδωσε την ευκαιρία να ασχοληθώ με μια ευρεία θεματολογία του τομέα της Μηχανικής Μάθησης, και συγκεκριμένα της Όρασης Υπολογιστών. Μέσα από αυτήν, κατάφερα να εντρυφήσω περαιτέρω με την ερευνητική διαδικασία, γνωρίζοντας καλύτερα τόσο τις θετικές όσο και τις αρνητικές στιγμές της. Θα ήθελα να εκφράσω τις πιο θερμές μου ευχαριστίες στην συνεπιβλέπουσά μου Νίκη Ευθυμίου, μεταδιδακτορική ερευνήτρια του ΕΜΠ, για τη χρήσιμη καθοδήγηση και στήριξη της καθόλη τη διάρκεια της διπλωματικής μου εργασίας. Οι στοχευμένες συμβουλές και προτάσεις της ήταν πολύτιμες για την επιτυχημένη ολοκλήρωση της έρευνας μου, αλλα και την επιστημονική μου εξέλιξη. Επίσης, θα επιθυμούσα να ευχαριστήσω τον επιβλέποντα καθηγητή μου κ.Πέτρο Μαραγκό, καθηγητή Ε.Μ.Π., που το έργο του και η διδασκαλία τους ήταν καίριος παράγοντας ώστε να μου καλλιεργήσει τόσο το ενδιαφέρον όσο και την επιθυμία για να ασχοληθώ με τον τομέα της Όραση Υπολογιστών, καθώς και την δυνατότητα να εκπονήσω αυτήν την διπλωματική υπό την επίβλεψή του.

Τέλος, χωρίς την βοήθεια της οικογένειας και των φίλων μου η πορεία μου σε αυτό το πενταετές φοιτητικό ταξίδι δεν θα ήταν η ίδια, αφού στα εύκολα και στα δύσκολα ήταν πάντα δίπλα μου παρέχοντας μου στήριξη και καθοδήγηση όποτε το χρειαζόμουν.

Μπενέτος Νικόλαος
Οκτώβριος 2024

Περιεχόμενα

Λίστα Σχημάτων

Κατάλογος Πινάκων

Εκτεταμένη περίληψη στα Ελληνικά

Εισαγωγή

Η μελέτη των αλληλεπιδράσεων των παιδιών είναι κρίσιμη για την κατανόηση του τρόπου με τον οποίο αναπτύσσουν κοινωνικές, συναισθηματικές και γνωστικές δεξιότητες, ιδιαίτερα στην πρώιμη παιδική ηλικία. Οι αλληλεπιδράσεις με φροντιστές, συνομήλικους και δασκάλους βοηθούν στη διαμόρφωση της ικανότητας του παιδιού να επικοινωνεί, να επιλύει προβλήματα και να χτίζει σχέσεις. Τα τελευταία χρόνια, η Αλληλεπίδραση Παιδιού-Ρομπότ (Child-Robot Interaction, CRI) έχει αναδειχθεί ως σημαντικός τομέας έρευνας, προσφέροντας νέες γνώσεις σχετικά με το πώς τα παιδιά αλληλεπιδρούν με την τεχνολογία. Τα ρομπότ που σχεδιάζονται για εκπαιδευτικούς ή θεραπευτικούς σκοπούς μπορούν να λειτουργήσουν ως διαδραστικοί συνεργάτες, βοηθώντας τα παιδιά να μαθαίνουν μέσω παιχνιδιού και κοινωνικών ανταλλαγών. Κεντρικό σημείο αυτών των αλληλεπιδράσεων είναι η στοχοπροσήλωση (engagement) του παιδιού, η οποία αποτελεί έναν έγκυρο δείκτη για το πόσο συγκεντρωμένο είναι το παιδί κατά την αλληλεπίδραση του και πώς διαφορετικές προσεγγίσεις αλλάζουν τη συμπεριφορά του. Η στοχοπροσήλωση οπτικά μπορεί πολλές φορές να γίνει αντιληπτή μέσω ενδείξεων όπως το βλέμμα, που υποδεικνύει την προσοχή και τη συγκέντρωση. Τα μοτίβα του βλέμματος αποκαλύπτουν πώς τα παιδιά επεξεργάζονται τις πληροφορίες και το επίπεδο συμμετοχής τους στην αλληλεπίδραση. Η κατανόηση της στοχοπροσήλωσης και του βλέμματος τόσο στις αλληλεπιδράσεις ανθρώπων όσο και ρομπότ μπορεί να ενισχύσει τον σχεδιασμό παρεμβάσεων που προάγουν τη μάθηση, την κοινωνικοποίηση και την συναισθηματική ανάπτυξη. Αυτή η γνώση μπορεί να οδηγήσει σε βελτιωμένα αποτελέσματα στην εκπαίδευση, τη θεραπεία και τις συνεργασίες παιδιού-ρομπότ, αλλά και στην παρατήρηση παιδιών με διαταραχή του αυτιστικού φάσματος (ASD) και στο πώς αλληλεπιδρούν διαφορετικά σε σύγκριση με τα κανονικά αναπτυσσόμενα παιδιά. Τα παιδιά με αυτισμό συχνά αντιμετωπίζουν προκλήσεις στην κοινωνική στοχοπροσήλωση, την επικοινωνία και την προσοχή, και τα ρομπότ που έχουν σχεδιαστεί για εκπαιδευτικούς ή θεραπευτικούς σκοπούς προσφέρουν μια μοναδική, χωρίς κριτική, πλατφόρμα αλληλεπίδρασης. Αυτά τα ρομπότ μπορούν να προγραμματιστούν για να προσαρμόζονται στις ατομικές ανάγκες, βοηθώντας τα παιδιά με αυτισμό να εξασκούνται σε κοινωνικές συμπεριφορές σε ένα δομημένο και ελεγχόμενο περιβάλλον.

Στοχοπροσήλωση στις αλληλεπιδράσεις παιδιών και ρομπότ

Η στοχοπροσήλωση αναφέρεται στον βαθμό προσοχής, περιέργειας, ενδιαφέροντος και συμμετοχής που επιδεικνύει ένα άτομο σε μια συγκεκριμένη δραστηριότητα ή αλληλεπίδραση. Είναι μια πολυδιάστατη έννοια που μπορεί να περιλαμβάνει συναισθηματικά, γνωστικά και συμπεριφορικά στοιχεία, όπου το άτομο συμμετέχει ενεργά ή επενδύεται νοητικά στην παρούσα εργασία. Η στοχοπροσήλωση είναι κρίσιμη για την παραγωγικότητα και την επιτυχία σε διάφορα πλαίσια, από την εκπαίδευση και την εργασία έως τις προσωπικές σχέσεις. Στη μάθηση, για παράδειγμα, η υψηλή στοχοπροσήλωση συνήθως οδηγεί σε καλύτερη απορρόφηση γνώσεων και απόκτηση δεξιοτήτων, ενώ η έλλειψη στοχοπροσήλωσης μπορεί να εμποδίσει την πρόοδο και την ανάπτυξη [62].

Η στοχοπροσήλωση στην Αλληλεπίδραση Παιδιού-Ρομπότ (CRI) είναι μια κατάσταση στην οποία ένα παιδί επιδεικνύει ενδιαφέρον, περιέργεια και ενθουσιασμό κατά την αλληλεπίδρασή του με ένα ρομπότ [6]. Αυτό περιλαμβάνει το παιδί να εστιάζει την προσοχή του στις ενέργειες και τις αλληλεπιδράσεις του ρομπότ, να συμμετέχει ενεργά, είτε ξεκινώντας είτε ανταποκρινόμενο στα ερεθίσματα του ρομπότ. Επιπλέον, η στοχοπροσήλωση χαρακτηρίζεται επίσης από μια συναισθηματική απόκριση, όπως γέλιο, χαμόγελα ή εκφράσεις ενσυναίσθησης, που υποδηλώνουν μια βαθύτερη σύνδεση με την εμπειρία. Ένα σημαντικό χαρακτηριστικό της στοχοπροσήλωσης είναι η επιμονή, όπου το παιδί συνεχίζει να αλληλεπιδρά για παρατεταμένο χρονικό διάστημα, δείχνοντας διαρκές ενδιαφέρον και συμμετοχή. Υπάρχουν πολλά θετικά αποτελέσματα από τη στοχοπροσήλωση του παιδιού κατά

την αλληλεπίδραση με άλλους, συμπεριλαμβανομένης της διευκόλυνσης της μάθησης και της εξερεύνησης νέων ιδεών, εννοιών ή δεξιοτήτων, που τελικά συμβάλλουν στην παραγωγικότητα, την ικανοποίηση και την ευημερία του. Ένας σημαντικός στόχος στην έρευνα σχετικά με τη στοχοπροσήλωση είναι να καταφέρουμε να φτάσουμε σε ένα σημείο όπου ένα ρομπότ να μπορεί να αναγνωρίσει σωστά το επίπεδο στοχοπροσήλωσης, κάτι που όμως μπορεί να είναι εξαιρετικά δύσκολο, καθώς είναι μια εσωτερική νοητική κατάσταση και δεν μπορεί να παρατηρηθεί άμεσα [46], αφού το ρομπότ περιορίζεται στην εκμετάλλευση εξωτερικών οπτικών ή ακουστικών ενδείξεων για να εκτιμήσει το επίπεδό της [18].

Παρακολούθηση του βλέμματος

Η παρακολούθηση του βλέμματος είναι ένα ισχυρό εργαλείο για την κατανόηση και ανάλυση των μοτίβων οπτικής προσοχής. Παρακολουθώντας και καταγράφοντας την κατεύθυνση και την εστίαση του βλέμματος ενός παιδιού, οι ερευνητές μπορούν να αποκομίσουν σημαντικές πληροφορίες για διάφορες γνωστικές και κοινωνικές διαδικασίες, όπως το πώς τα παιδιά αντιλαμβάνονται και αλληλεπιδρούν με το περιβάλλον τους. Στο πλαίσιο των αναπτυξιακών μελετών, η παρακολούθηση του βλέμματος μπορεί να αποκαλύψει σημαντικές πτυχές των κοινωνικών δεξιοτήτων επικοινωνίας του παιδιού, της διάρκειας προσοχής και των μαθησιακών συμπεριφορών [55]. Συνήθως, για να αντιμετωπιστεί το πρόβλημα της παρακολούθησης του βλέμματος με επιτυχία και υψηλή ακρίβεια, απαιτούνται μεγάλα συστήματα παρακολούθησης και πειραματικές ρυθμίσεις [41]. Για αυτούς τους λόγους, στη μελέτη μας θα επικεντρωθούμε σε προσεγγίσεις που βασίζονται σε βίντεο καταγραφές κατά τη διάρκεια ζωντανών αλληλεπιδράσεων. Αυτές οι μέθοδοι για την εκτίμηση του βλέμματος δεν έχουν φτάσει ακόμη στην κορυφαία τους απόδοση, παρόλο που τα τελευταία χρόνια οι αλγόριθμοι για συναφή προβλήματα μοντελοποίησης ανθρώπων, όπως η παρακολούθηση της στάσης του σώματος και του προσώπου σε 2D, έχουν σημειώσει εντυπωσιακή επιτυχία αξιοποιώντας τη δυναμική των βαθιών συνελικτικών νευρωνικών δικτύων μαζί με πολύ μεγάλα σχολιασμένα σύνολα δεδομένων [11, 27, 34]. Αυτό οφείλεται κυρίως στη φύση της παρακολούθησης του βλέμματος σε φυσικές συνθήκες, όπου μπορεί να υπάρχει μερική ή πλήρης απόκρυψη των ματιών ή ακόμη και του προσώπου του ατόμου, καθιστώντας εξαιρετικά δύσκολο για τους μηχανισμούς μοντελοποίησης να μάθουν και να προβλέψουν πάνω σε αυτά. Επιπλέον, για να αντιμετωπιστεί με επιτυχία αυτό το συγκεκριμένο έργο, χρειάζεται επίσης ακριβής και πολύ ποικίλη συλλογή δεδομένων βλέμματος με απόλυτη αλήθεια, ιδίως έξω από το εργαστήριο, κάτι που συνήθως αποτελεί μια δύσκολη πρόκληση.

Διαταραχή του Αυτιστικού Φάσματος (ΔΑΦ)

Όπως αναφέρθηκε προηγουμένως, μπορεί να προκύψουν πολλά θετικά αποτελέσματα από την επιτυχή μελέτη των αλληλεπιδράσεων των παιδιών. Ένα από αυτά μπορεί να είναι η δημιουργία αρχιτεκτονικών που εντοπίζουν με επιτυχία παιδιά με διαταραχή αυτιστικού φάσματος. Η Διαταραχή Αυτιστικού Φάσματος (ΔΑΦ) είναι μια πολύπλοκη νευρολογική και αναπτυξιακή διαταραχή που χαρακτηρίζεται από προκλήσεις στην κοινωνική επικοινωνία, την αλληλεπίδραση, καθώς και από μια τάση για περιορισμένες και επαναλαμβανόμενες συμπεριφορές. Ο αυτισμός εκδηλώνεται διαφορετικά σε κάθε άτομο, με ευρύ φάσμα συμπτωμάτων και επιπέδων σοβαρότητας, καθιστώντας τον μια διαταραχή φάσματος. Κοινά σημάδια περιλαμβάνουν μειωμένη οπτική επαφή, περιορισμένες εκφράσεις του προσώπου, δυσκολία στην κατανόηση κοινωνικών σημάτων και άτυπες αντιδράσεις σε αισθητηριακά ερεθίσματα. Αυτά τα ελλείμματα στην κοινωνική επικοινωνία μπορούν να οδηγήσουν σε σημαντικές δυσκολίες στη δημιουργία σχέσεων, στη συμμετοχή σε τυπικές καθημερινές δραστηριότητες και στην επιτυχία σε παραδοσιακά εκπαιδευτικά περιβάλλοντα.

Η έγκαιρη ανίχνευση του αυτισμού είναι κρίσιμη για διάφορους λόγους. Η πρώιμη παρέμβαση έχει αποδειχθεί σταθερά ότι βελτιώνει τα αναπτυξιακά αποτελέσματα σε παιδιά με ΔΑΦ [77, 86]. Μέσω της αναγνώρισης της διαταραχής κατά τα κρίσιμα πρώτα χρόνια ανάπτυξης του εγκεφάλου, μπορούν να εφαρμοστούν στοχευμένες θεραπείες και εκπαιδευτικά προγράμματα για να αντιμετωπιστούν συγκεκριμένες προκλήσεις. Αυτές οι παρεμβάσεις μπορούν να ενισχύσουν σημαντικά τις δεξιότητες επικοινωνίας του παιδιού, τις κοινωνικές αλληλεπιδράσεις και τη συνολική ποιότητα ζωής, μειώνοντας τις μακροπρόθεσμες επιπτώσεις της διαταραχής.

Επιπλέον, η έγκαιρη ανίχνευση επιτρέπει στις οικογένειες να κατανοήσουν καλύτερα τις ανάγκες του παιδιού τους, να έχουν πρόσβαση στις κατάλληλες υπηρεσίες υποστήριξης και να λαμβάνουν ενημερωμένες αποφάσεις για την εκπαίδευση και τη φροντίδα του. Βοηθά επίσης στην ελάφρυνση του γονικού άγχους, παρέχοντας σαφήνεια και κατεύθυνση, επιτρέποντας στους γονείς να συμμετέχουν ενεργά στο αναπτυξιακό ταξίδι του παιδιού τους.

Δεδομένων των βαθιών επιπτώσεων του αυτισμού στη ζωή του ατόμου, αλλά και σε κοινωνικό και οικογενειακό

επίπεδο, η πρώιμη διάγνωση δεν είναι μόνο επωφελής αλλά απαραίτητη. Δημιουργεί τη βάση για παρεμβάσεις που μπορούν να αλλάξουν την αναπτυξιακή πορεία του παιδιού, ενισχύοντας την αυτονομία και την ενσωμάτωση στην κοινωνία. Τα κοινωνικά ρομπότ έχουν ήδη εισαχθεί στη διαδικασία αναγνώρισης παιδιών με Διαταραχή Αυτιστικού Φάσματος (ΔΑΦ) ή μαθησιακές δυσκολίες, για την αντιμετώπιση των συνεπειών και των προκλήσεων αυτών των διαταραχών [74, 40], και υπάρχουν επίσης ελπιδοφόρα αποτελέσματα από τη χρήση ρομπότ για την υποστήριξη της κοινωνικής και συναισθηματικής ανάπτυξης παιδιών με ΔΑΦ [70, 69, 2]. Καθώς η έρευνα συνεχίζει να εξελίσσεται, η ανάπτυξη ακριβέστερων και πιο προσιτών μεθόδων έγκαιρης ανίχνευσης παραμένει κορυφαία προτεραιότητα στον τομέα της έρευνας και φροντίδας του αυτισμού.

Στοχοπροσήλωση στην Ανίχνευση Αυτισμού

Η στοχοπροσήλωση παίζει καθοριστικό ρόλο στην ανίχνευση του αυτισμού, καθώς τα παιδιά με αυτισμό συχνά εμφανίζουν δυσκολίες στην κοινωνική επικοινωνία και αλληλεπίδραση, όπως μειωμένη οπτική επαφή, περιορισμένες εκφράσεις προσώπου και προκλήσεις στην απόκριση σε κοινωνικά σήματα. Επίσης, είναι σύνηθες να παρουσιάζουν άτυπα πρότυπα βλέμματος, όπως μειωμένη προσοχή σε κοινωνικά ερεθίσματα, όπως πρόσωπα ή μια τάση να επικεντρώνονται σε συγκεκριμένα αντικείμενα αντί για ανθρώπους. Οι τεχνολογίες αναγνώρισης της στοχοπροσήλωσης προσφέρουν μια υποσχόμενη προσέγγιση για την ανίχνευση αυτών των συμπεριφορών, παρέχοντας αντικειμενικά και ποσοτικά μέτρα των κοινωνικών και επικοινωνιακών συμπεριφορών των παιδιών. Αυτές οι τεχνολογίες προσφέρουν μια οικονομικά αποδοτική και κλιμακούμενη μέθοδο διαλογής μεγάλων πληθυσμών παιδιών, καθιστώντας την έγκαιρη ανίχνευση του αυτισμού πιο προσιτή και διαδεδομένη [17, 53].

Παρακολούθηση Βλέμματος στην Ανίχνευση Αυτισμού

Η τεχνολογία παρακολούθησης βλέμματος μπορεί να είναι εξαιρετικά πολύτιμη στην ταυτοποίηση άτυπων μοτίβων βλέμματος που μπορεί να υποδηλώνουν αναπτυξιακές διαταραχές, όπως η Διαταραχή Αυτιστικού Φάσματος (ΔΑΦ). Για παράδειγμα, τα παιδιά με ΔΑΦ συχνά εμφανίζουν ασυνήθιστα μοτίβα βλέμματος, όπως μειωμένη οπτική επαφή ή μεγαλύτερη εστίαση σε αντικείμενα παρά σε ανθρώπους. Η παρακολούθηση βλέμματος μέσω ανάλυσης βίντεο επιτρέπει την αντικειμενική μέτρηση αυτών των συμπεριφορών, παρέχοντας μια μη παρεμβατική και ακριβή μέθοδο για την αξιολόγηση της στοχοπροσήλωσης και της κοινωνικής αλληλεπίδρασης των παιδιών σε φυσικές συνθήκες [35, 50]. Ως εκ τούτου, έχει σημαντική προοπτική για την έγκαιρη ανίχνευση, την παρέμβαση και τη συνολική κατανόηση των αναπτυξιακών δυσκολιών των παιδιών.

Στόχοι και δομή της μελέτης

Όλα τα παραπάνω καταδεικνύουν τη σημασία τόσο της εκτίμησης της στοχοπροσήλωσης όσο και της παρακολούθησης του βλέμματος σε βίντεο αλληλεπιδράσεων παιδιών. Αυτές οι τεχνολογίες μπορούν να αποδειχθούν εξαιρετικά πολύτιμες, τόσο ως αυτόνομες εφαρμογές, όσο και ως μέρος μιας ευρύτερης εικόνας, όταν χρησιμοποιούνται ως εργαλεία για την ταυτοποίηση αναπτυξιακών διαταραχών, όπως η διαταραχή αυτιστικού φάσματος (ΔΑΦ). Για τον λόγο αυτό, η παρούσα μελέτη αποσκοπεί να ερευνήσει πώς τα προβλήματα αναγνώρισης στοχοπροσήλωσης και παρακολούθησης βλέμματος μπορούν να χρησιμοποιηθούν σε οπτικά σύνολα δεδομένων, με ποιους τρόπους η απόδοσή τους θα μπορούσε να βελτιωθεί περαιτέρω, και πώς η παρακολούθηση βλέμματος μπορεί να συμπληρώσει την εκτίμηση δέσμευσης, ώστε να παραχθούν καλύτερα αποτελέσματα.

Προσπαθούμε να επιτύχουμε αυτόν τον κύριο στόχο μας, την ενίσχυση της εκτίμησης της δέσμευσης αλλά και της παρακολούθησης του βλέμματος των παιδιών με απώτερο σκοπό να χρησιμοποιηθούν σε αρχιτεκτονικής αναγνώρισης παιδιών με αναπτυξιακές δυσκολίες, ακολουθώντας τα εξής βήματα:

1. Συγκεντρώνουμε πολλές πληροφορίες σχετικά με κάθε πρόβλημα ξεχωριστά, συμπεριλαμβανομένων προηγούμενων μεθόδων και αλγορίθμων που χρησιμοποιήθηκαν για την επίλυσή τους, όπως υβριδικά δίκτυα χωροχρονικής ανάλυσης και υλοποιήσεις πολυδιάστατης σύντηξης χαρακτηριστικών.

2. Δημιουργούμε ένα ανθεκτικό πλαίσιο εκτίμησης της στοχοπροσήλωσης, αποτελούμενο από ένα 3D ConvNet, υλοποιημένο χρησιμοποιώντας την αρχιτεκτονική του Δικτύου Χρονικών Τμημάτων (TSN).

3. Χρησιμοποιούμε ένα ήδη εκπαιδευμένο δίκτυο παρακολούθησης βλέμματος, το οποίο το μετατρέπουμε ώστε να μπορεί να ανταπεξέλθει στο πρόβλημα της παρακολούθησης βλέμματος σε ελεύθερα περιβάλλοντα, σε αλληλεπιδράσεις των παιδιών με τις μητέρες τους, όπου υπάρχουν στόχοι ενδιαφέροντος ως ετικέτες για την

εκτίμηση του βλέμματος αντί των συντεταγμένων της κατεύθυνσης του βλέμματος, που χρησιμοποιούνται συνήθως.

4. Αφού αντιμετωπίσουμε επιτυχώς τις εργασίες αναγνώρισης στοχοπροσήλωσης και παρακολούθησης βλέμματος ξεχωριστά, αποφασίζουμε να χρησιμοποιήσουμε το δίκτυο παρακολούθησης βλέμματος για να βελτιώσουμε περαιτέρω τα αποτελέσματα στο πρόβλημα της εκτίμησης της στοχοπροσήλωσης.

Αυτή η διατριβή χωρίζεται σε 5 κεφάλαια. Ακολουθεί μια σύντομη περίληψη του περιεχομένου τους:

- Στο κεφάλαιο 1, παρουσιάζεται το αντικείμενο αυτής της μελέτης, που επικεντρώνεται στην Εκτίμηση της Στοχοπροσήλωσης και την Παρακολούθηση Βλέμματος κατά τη διάρκεια των αλληλεπιδράσεων παιδιών, γιατί είναι σημαντικό να βελτιωθούν και πώς μπορούν να βοηθήσουν σε διάφορους τομείς.

- Στο κεφάλαιο 2, παρουσιάζεται το θεωρητικό υπόβαθρο αυτής της μελέτης, εξηγώντας σημαντικές δομές, μεθοδολογίες και μοντέλα που χρησιμοποιούνται σε παρόμοια προβλήματα Όρασης Υπολογιστών, καθώς και αλγόριθμους και πρακτικές που χρησιμοποιούνται στην εκπαίδευση μοντέλων Μηχανικής Μάθησης.

- Στο κεφάλαιο 3, παρουσιάζεται μια εκτενής μελέτη της σχετικής βιβλιογραφίας στους τομείς της αναγνώρισης της στοχοπροσήλωσης και της παρακολούθησης βλέμματος, με έμφαση σε διαφορετικές αρχιτεκτονικές μοντέλων και υλοποιήσεις που έχουν αναπτυχθεί και πώς αυτές προσπαθούν να αντιμετωπίσουν τα αντίστοιχα προβλήματα.

- Στο κεφάλαιο 4, πρώτα παρουσιάζονται οι μέθοδοι που προτείνουμε για να αντιμετωπιστούν τα προβλήματα της Εκτίμησης της Στοχοπροσήλωσης και της Παρακολούθησης Βλέμματος. Αυτές περιλαμβάνουν μια αρχιτεκτονική εκτίμησης της στοχοπροσήλωσης, που αποτελείται από ένα μοντέλο backbone και τη μέθοδο TSN, μια υλοποίηση εκτίμησης βλέμματος και τέλος μια συνδυασμένη υλοποίηση εκτίμησης της στοχοπροσήλωσης και βλέμματος που συνδυάζει δεδομένα παρακολούθησης βλέμματος και στοχοπροσήλωσης. Στη συνέχεια, αναλύουμε τα πειράματα που διεξήχθησαν σε αυτή τη διατριβή και παρουσιάζουμε τα αποτελέσματά τους. Αυτά περιλαμβάνουν υψηλή απόδοση στο πρόβλημα εκτίμησης της στοχοπροσήλωσης, ακριβή εκτίμηση βλέμματος σε παιδιά σε φυσικά περιβάλλοντα, καθώς και περαιτέρω βελτίωση της απόδοσης εκτίμησης της στοχοπροσήλωσης μέσω της σύντηξης των δικτύων της στοχοπροσήλωσης και παρακολούθησης βλέμματος.

- Στο κεφάλαιο 5, πραγματοποιείται μια σύνοψη της μελέτης αυτής, παρουσιάζοντας τα συμπεράσματα από τα πειράματα στους δύο αυτούς τομείς και τις αρχιτεκτονικές μοντέλων που αναπτύχθηκαν, και επιπλέον διατυπώνονται προτάσεις για μελλοντική έρευνα που μπορεί να βασιστεί στο παρουσιαζόμενο έργο.

Θεωρητικό Υπόβαθρο

Συνελικτικά Νευρωνικά Δίκτυα

Ο νευρώνας

Η θεμελιώδης μονάδα επεξεργασίας ενός Νευρωνικού Δικτύου, ο νευρώνας, είναι εμπνευσμένη από την παρατήρηση της δομής του εγκεφάλου. Όπως ακριβώς και στον εγκέφαλο, όπου οι νευρώνες σχηματίζουν ένα πολύπλοκο, άκρως διασυνδεδεμένο δίκτυο και στέλνουν ηλεκτρικά σήματα μεταξύ τους για να βοηθήσουν τους ανθρώπους να επεξεργαστούν πληροφορίες, έτσι και οι νευρώνες των νευρωνικών δικτύων συνεργάζονται για την επίλυση ενός κοινού προβλήματος, χρησιμοποιώντας υπολογιστικά συστήματα για να λύσουν μαθηματικούς υπολογισμούς. Συγκεκριμένα, ο νευρώνας ενός νευρωνικού δικτύου επεξεργάζεται μια τοπική περιοχή των εισερχόμενων δεδομένων, χρησιμοποιεί εκπαιδεύσιμα βάρη για να υπολογίσει ένα σταθμισμένο άθροισμα, προσθέτοντας επίσης μια εκπαιδεύσιμη μεροληψία, εφαρμόζει μια συνάρτηση ενεργοποίησης και συμβάλλει στο σχηματισμό χαρτών χαρακτηριστικών. Η συνάρτηση ενεργοποίησης που εφαρμόζεται ερμηνεύει το αποτέλεσμα και το παρουσιάζει με ουσιαστικό τρόπο, εισάγοντας συχνά μη-γραμμικότητα στο δίκτυο, επιτρέποντάς του να μοντελοποιεί πιο σύνθετα πρότυπα.

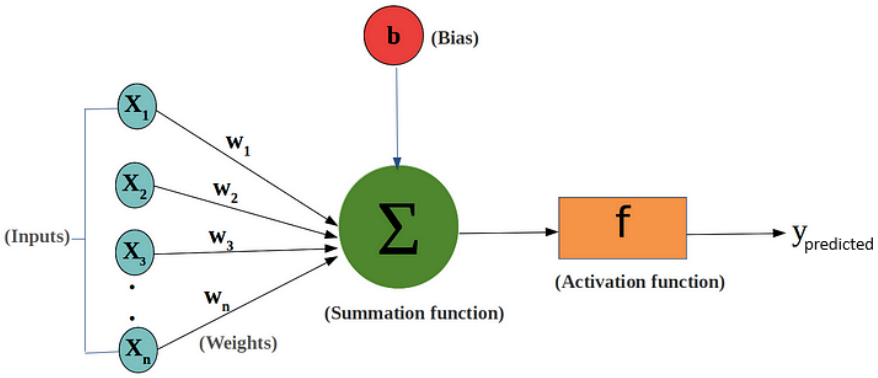


Figure 0.0.1: Η αρχιτεκτονική του νευρώνα, ο νευρώνας δέχεται ως είσοδο το σταθμοισμένο άθροισμα των δεδομένων εισόδου και των εκπαιδεύσιμων βαρών, προσθέτει μία εκπαιδεύσιμη μεροληψία, εφαρμόζει μία συνάρτηση ενεργοποίησης και παράγει την τελική έξοδο από [67].

Συναρτήσεις Ενεργοποίησης

Όπως αναφέρθηκε παραπάνω, οι συναρτήσεις ενεργοποίησης είναι συναρτήσεις που εφαρμόζονται στο σταθμισμένο άθροισμα των εισερχόμενων σημάτων με τα βάρη του νευρώνα και συχνά εισάγουν μη-γραμμικότητα στον υπολογισμό του αποτελέσματος. Μερικές από τις συναρτήσεις ενεργοποίησης που χρησιμοποιούνται συνήθως είναι:

ReLU Αυτή τη στιγμή η πιο χρησιμοποιούμενη στον κόσμο, η ReLU έχει κατώφλι στο μηδέν με την έξοδο:

$$f(x) = \max(0, x) \tag{0.0.1}$$

Αυτή η συνάρτηση υλοποιείται εύκολα και εξαλείφει το πρόβλημα του saturating gradient. Ένα ζήτημα με αυτή τη συνάρτηση είναι ότι όλες οι αρνητικές τιμές που δίνονται ως είσοδος γίνονται αμέσως μηδέν, γεγονός που με τη σειρά του επηρεάζει τα αποτελέσματα καθώς δεν χαρτογραφεί σωστά τις αρνητικές τιμές.

Leaky-ReLU Η Leaky-ReLU προσπαθεί να αντιμετωπίσει το πρόβλημα της ReLU πολλαπλασιάζοντας τις αρνητικές εισόδους με μια μικρή σταθερά, αντί να μηδενίζει πλήρως τις αρνητικές τιμές:

$$f(x) = \mathbb{1}(x < 0)(\alpha x) + \mathbb{1}(x \geq 0)(x) \tag{0.0.2}$$

Sigmoid Αυτή η συνάρτηση ενεργοποίησης χρησιμοποιείται κυρίως λόγω του ότι βρίσκεται μεταξύ 0 και 1. Επομένως, χρησιμοποιείται ιδιαίτερα σε μοντέλα όπου πρέπει να προβλέψουμε την πιθανότητα ως έξοδο. Δεδομένου ότι η πιθανότητα οτιδήποτε υπάρχει μόνο στο εύρος 0 έως 1, η sigmoid είναι η κατάλληλη επιλογή. Ο τύπος της είναι ο εξής:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{0.0.3}$$

An issue with this function is that it can cause a neural network to get stuck at the training time. Ένα πρόβλημα με αυτή τη συνάρτηση είναι ότι μπορεί να κολλήσει το νευρωνικό δίκτυο κατά τη διάρκεια της εκπαίδευσης.

Tanh Η συνάρτηση tanh είναι μια βελτιωμένη έκδοση της sigmoid, αλλά είναι μηδενικά κεντραρισμένη όπως φαίνεται στο Σχήμα 0.0.2. Η tanh χαρτογραφεί την είσοδο στο εύρος $[-1, 1]$, αντί για $[0, 1]$, με τον ακόλουθο τύπο:

$$\tanh(x) = s\sigma(2x) - 1 \tag{0.0.4}$$

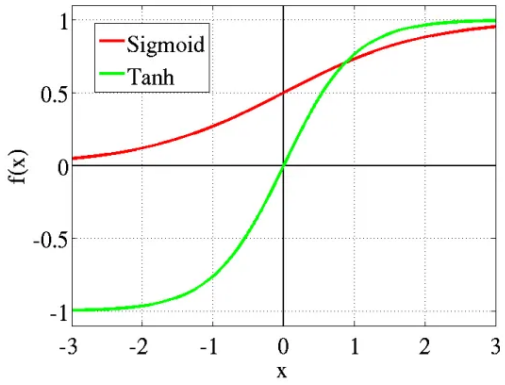


Figure 0.0.2: Sigmoid vs Tanh από [24].

Ωστόσο, μπορεί και πάλι να προκαλέσει την κορεσμό του gradient.

Νευρωνικά Δίκτυα

Τα Νευρωνικά Δίκτυα [43] αποτελούνται από στρώματα διασυνδεδεμένων νευρώνων, που συνεργάζονται για να επεξεργαστούν δεδομένα εισόδου, να αναγνωρίσουν μοτίβα και να λάβουν αποφάσεις ή προβλέψεις. Το πρώτο από τα στρώματα, το οποίο λαμβάνει τα ακατέργαστα δεδομένα, είναι το επίπεδο εισόδου (input layer), ακολουθούμενο από πολλά ενδιάμεσα στρώματα, που ονομάζονται Κρυφά Στρώματα (Hidden Layers), όπου οι νευρώνες επεξεργάζονται ανεξάρτητα τις εισόδους και ανιχνεύουν μοτίβα. Αυτά τα στρώματα εκτελούν μετασχηματισμούς στα δεδομένα και δεν είναι απευθείας παρατηρήσιμα από το εξωτερικό. Τέλος, το τελικό επίπεδο, που ονομάζεται επίπεδο εξόδου (output layer), παράγει τις προβλέψεις ή τις αποφάσεις του δικτύου και η δομή του εξαρτάται από την εργασία (π.χ., ταξινόμηση, παλινδρόμηση). Για να ενημερωθούν τα βάρη των νευρωνικών δικτύων σύμφωνα με τον αλγόριθμο backpropagation, είναι απαραίτητο το gradient της απώλειας. Η απώλεια (loss) υπολογίζεται χρησιμοποιώντας μια συνάρτηση απώλειας, η οποία συγκρίνει την έξοδο του τελευταίου στρώματος με την πραγματική τιμή (ground truth).

Συναρτήσεις απώλειας

Η γενικευμένη εξίσωση για τη συνάρτηση απώλειας είναι:

$$J(w^T, b) = \frac{1}{m} \sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)}) \tag{0.0.5}$$

όπου w είναι τα βάρη, b η μεροληψία (bias), m ο συνολικός αριθμός σημείων δεδομένων του συνόλου εκπαίδευσης, $\hat{y}$ είναι η πρόβλεψη και y η πραγματική τιμή.

Μέσο Τετραγωνικό Λάθος (Mean Squared Error) Η πιο δημοφιλής συνάρτηση απώλειας, που υπολογίζει τη διαφορά μεταξύ της πραγματικής τιμής και της προβλεπόμενης τιμής, την υψώνει στο τετράγωνο και στο τέλος υπολογίζει τον μέσο όρο:

$$MSE = \frac{1}{m}\sum_{i=1}^{m}(y_{actual} - y_{predicted})^2 \tag{0.0.6}$$

Τα μειονεκτήματά του είναι ότι επηρεάζεται από ακραίες τιμές (outliers) και ότι δεν μπορεί να χρησιμοποιηθεί για την άμεση ερμηνεία ή σύγκριση με την πραγματική τιμή.

Cross Entropy Loss Η Συνάρτηση Απώλειας Cross Entropy είναι μια συνάρτηση απώλειας που χρησιμοποιείται κυρίως για προβλήματα ταξινόμησης, μετρώντας τη διαφορά μεταξύ δύο κατανομών πιθανοτήτων: της προβλεπόμενης κατανομής και της πραγματικής κατανομής των ετικετών (συχνά αναπαριστάται ως one-hot encoded vectors). Για κάθε κατηγορία, υπολογίζει το αρνητικό λογάριθμο της προβλεπόμενης πιθανότητας που αντιστοιχεί στην πραγματική κατηγορία και προσθέτει αυτά τα αποτελέσματα για όλες τις κατηγορίες:

$$L = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \tag{0.0.7}$$

Το γεγονός ότι είναι μια ομαλή και παραγώγιμη συνάρτηση, που παρέχει χρήσιμες κλίσεις για μάθηση και είναι πιο αποτελεσματική στον χειρισμό πιθανοτήτων κατηγοριών, την καθιστά προτιμητέα σε προβλήματα ταξινόμησης, σε σύγκριση με άλλες συναρτήσεις απώλειας, όπως το mean squared error (MSE).

Η βελτιστοποίηση (optimization) - Gradient Descent

Η βελτιστοποίηση (optimization) είναι η διαδικασία προσαρμογής των παραμέτρων ενός μοντέλου (βαρών) για την ελαχιστοποίηση ή μεγιστοποίηση μιας συνάρτησης στόχου, συνήθως της συνάρτησης απώλειας. Αυτό μπορεί να επιτευχθεί με χρήση του Gradient Descent, ενός αλγορίθμου βελτιστοποίησης που προσπαθεί να ελαχιστοποιήσει μια συνάρτηση πηγαίνοντας επαναληπτικά προς την πιο απότομη κατεύθυνση καθόδου, όπως ορίζεται από το αρνητικό του gradient.

$$\theta = \theta - \eta\nabla_\theta J(\theta) \tag{0.0.8}$$

όπου η είναι ο ρυθμός μάθησης (learning rate). Το Gradient Descent είναι η διαδικασία επαναλαμβανόμενης αξιολόγησης του gradient και στη συνέχεια η εκτέλεση μιας ενημέρωσης των παραμέτρων.

Αλγόριθμος backpropagation

Ο αλγόριθμος backpropagation είναι μια ευρέως χρησιμοποιούμενη μέθοδος για την εκπαίδευση τεχνητών νευρωνικών δικτύων, ελαχιστοποιώντας το σφάλμα μεταξύ της προβλεπόμενης εξόδου και της πραγματικής τιμής στόχου. Περιλαμβάνει τον υπολογισμό του gradient της συνάρτησης απώλειας ως προς κάθε βάρος χρησιμοποιώντας τον κανόνα της αλυσίδας, προωθώντας αποδοτικά τα σφάλματα προς τα πίσω μέσα από τα στρώματα του δικτύου. Αυτή η διαδικασία επιτρέπει στο μοντέλο να προσαρμόζει τα βάρη του για να μειώνει το σφάλμα πρόβλεψης, βελτιώνοντας τελικά την απόδοσή του, καθώς στα μεγάλα νευρωνικά δίκτυα η σχέση κάποιων βαρών

με τη συνάρτηση απώλειας είναι πολύ δύσκολο να βρεθεί. Ο αλγόριθμος Backpropagation λειτουργεί μέσω δύο διαφορετικών διελεύσεων, που επαναλαμβάνονται για κάποιες εποχές, την προώθηση προς τα εμπρός και την αντίστροφη προώθηση.

Στην προώθηση προς τα εμπρός, ξεκινάμε προωθώντας τα δεδομένα εισόδου στο επίπεδο εισόδου, περνώντας από τα κρυφά στρώματα, μετρώντας τις προβλέψεις του δικτύου από το επίπεδο εξόδου και τέλος υπολογίζοντας το σφάλμα του δικτύου με βάση τις προβλέψεις που έκανε. Μόλις υπολογιστεί το σφάλμα του δικτύου, η φάση προώθησης προς τα εμπρός έχει τελειώσει και ξεκινά η αντίστροφη προώθηση.

Στην αντίστροφη προώθηση, η ροή αντιστρέφεται, ξεκινώντας από τον υπολογισμό του gradient της συνάρτησης απώλειας ως προς την έξοδο και στη συνέχεια εφαρμόζεται ο κανόνας της αλυσίδας για τον υπολογισμό του gradient της συνάρτησης απώλειας ως προς τις παραμέτρους και τις εισόδους κάθε στρώματος, με τα gradients να προωθούνται προς τα πίσω στο δίκτυο για την ενημέρωση των βαρών.

Πρώτα, υπολογίζουμε το Gradient της Απώλειας ως προς τα Βάρη του Επίπεδου Εξόδου:

$$\frac{\partial L_i}{\partial W_{output}} = \frac{\partial L_i}{\partial O} \frac{\partial O}{\partial W_{output}} \tag{0.0.9}$$

όπου L είναι η απώλεια, W τα βάρη και O η έξοδος.

Στη συνέχεια, προωθούμε το σφάλμα προς τα πίσω χρησιμοποιώντας τον κανόνα της αλυσίδας, οπότε για ένα κρυφό στρώμα h έχουμε:

$$\frac{\partial L_i}{\partial W_H} = \frac{\partial L_i}{\partial O} \frac{\partial O}{\partial h} \frac{\partial h}{\partial W_h} \tag{0.0.10}$$

Κανονικοποίηση (Regularization) Ο στόχος ενός νευρωνικού δικτύου είναι να μάθει μια αντιστοιχία από την είσοδο στην έξοδο από τα δεδομένα εκπαίδευσης και να την εφαρμόσει στα δεδομένα δοκιμών. Επομένως, είναι σημαντικό να μπορεί να γενικεύσει τα βάρη του και να μην μάθει συγκεκριμένα παραδείγματα από τα δεδομένα εκπαίδευσης. Όταν ένα μοντέλο μαθαίνει τον θόρυβο και τις λεπτομέρειες των δεδομένων εκπαίδευσης σε τέτοιο βαθμό που αποδίδει άσχημα σε νέα, άγνωστα δεδομένα, αυτό ονομάζεται υπερπροσαρμογή (overfitting). Η κανονικοποίηση είναι μια τεχνική που χρησιμοποιείται στη βαθιά μάθηση για την αποφυγή της υπερπροσαρμογής, προσθέτοντας μια ποινή στη συνάρτηση απώλειας κατά τη διάρκεια της εκπαίδευσης του μοντέλου, αποθαρρύνοντας υπερβολικά περίπλοκα μοντέλα και ενθαρρύνοντας απλούστερα, πιο γενικεύσιμα μοντέλα. Συγκεκριμένα, προσθέτει ένα επιπλέον συστατικό στη συνάρτηση απώλειας που εμποδίζει τα βάρη να αυξάνουν υπερβολικά το μέγεθός τους και έτσι να ενημερώνονται με έναν λιγότερο ευέλικτο τρόπο.

Κανονικοποίηση L1 (L1 Regularization) Στην κανονικοποίηση L1, για κάθε βάρος w προσθέτουμε τον όρο $\lambda_1|w|$ στη συνάρτηση απώλειας. Έχει την ενδιαφέρουσα ιδιότητα ότι οδηγεί τα διανύσματα βαρών να γίνονται αραιά κατά τη βελτιστοποίηση (δηλαδή, πολύ κοντά στο μηδέν). Με άλλα λόγια, οι νευρώνες με κανονικοποίηση L1 καταλήγουν να χρησιμοποιούν μόνο ένα αραιό υποσύνολο των πιο σημαντικών εισόδων τους και γίνονται σχεδόν αναλλοίωτοι στον "θορυβώδη" εισερχόμενο όγκο δεδομένων. Στην πράξη, αν δεν μας απασχολεί η ρητή επιλογή χαρακτηριστικών, η κανονικοποίηση L2 μπορεί να αναμένεται να δώσει ανώτερη απόδοση σε σχέση με την L1.

Κανονικοποίηση L2 (L2 Regularization) Η κανονικοποίηση L2 είναι ίσως η πιο κοινή μορφή κανονικοποίησης. Μπορεί να εφαρμοστεί με την ποινικοποίηση του τετραγωνικού μεγέθους όλων των παραμέτρων απευθείας στη συνάρτηση απώλειας. Δηλαδή, για κάθε βάρος w στο δίκτυο, προστίθεται ο όρος $\frac{1}{2}\lambda w^2$ στη συνάρτηση απώλειας, όπου λ είναι η ισχύς της κανονικοποίησης. Ο παράγοντας $\frac{1}{2}$ χρησιμοποιείται έτσι ώστε το gradient του όρου να είναι απλά λw. Η κανονικοποίηση L2 έχει την διαισθητική ερμηνεία της βαριάς ποινικοποίησης των "αιχμηρών" διανυσμάτων βαρών και της προτίμησης διάχυτων διανυσμάτων βαρών. Αυτό

έχει την ελκυστική ιδιότητα να ενθαρρύνει το δίκτυο να χρησιμοποιεί από λίγο όλες τις εισόδους του, αντί για μερικές εισόδους σε μεγάλο βαθμό. Επιπλέον, κατά την ενημέρωση των παραμέτρων στη gradient descent, η χρήση της κανονικοποίησης L2 τελικά σημαίνει ότι κάθε βάρος αποσυντίθεται γραμμικά: $w+ = -\lambda \times w$ προς το μηδέν. Είναι δυνατόν να συνδυάσετε την κανονικοποίηση L1 με την κανονικοποίηση L2: $\lambda_1|w| + \lambda_2 w^2$.

Dropout Η τεχνική του Dropout είναι μια δημοφιλής μέθοδος για την κανονικοποίηση των νευρωνικών δικτύων. Κατά τη διάρκεια κάθε επανάληψης εκπαίδευσης, το δίκτυο "αφαιρεί" τυχαία ένα ποσοστό νευρώνων (μαζί με τις συνδέσεις τους) από το δίκτυο. Αυτό αναγκάζει το δίκτυο να μαθαίνει πιο ανθεκτικά χαρακτηριστικά και μειώνει την πιθανότητα υπερβολικής εξάρτησης από συγκεκριμένους νευρώνες ή διαδρομές μέσω του δικτύου. Μπορεί να θεωρηθεί ως "δειγματοληψία" ενός νευρωνικού δικτύου μέσα στο πλήρες νευρωνικό δίκτυο, και ενημερώνονται μόνο οι παράμετροι του δειγματοληπτημένου δικτύου βάσει των δεδομένων εισόδου.

Επαύξηση Δεδομένων (Data Augmentation) Μια τεχνική κατά την οποία δημιουργούνται επιπλέον παραδείγματα εκπαίδευσης τροποποιώντας τα αρχικά δεδομένα εκπαίδευσης (π.χ., περιστροφή, αναστροφή, κοπή εικόνων). Βοηθά το μοντέλο να γενικεύει καλύτερα μαθαίνοντας από μια μεγαλύτερη ποικιλία παραδειγμάτων, μειώνοντας έτσι την υπερπροσαρμογή. Συχνά εφαρμόζεται σε λιγότερο αντιπροσωπευόμενες κατηγορίες του συνόλου δεδομένων εκπαίδευσης ώστε να μειωθεί η διαφορά με τις κατηγορίες πλειοψηφίας.

Μέγεθος παρτίδας (Batch size)

Το μέγεθος παρτίδας είναι μια κρίσιμη υπερπαράμετρος που αναφέρεται στον αριθμό των παραδειγμάτων εκπαίδευσης που χρησιμοποιούνται σε κάθε μία προώθηση/αντίστροφη προώθηση μέσω του μοντέλου. Έτσι, το μοντέλο πρώτα κάνει τις προβλέψεις του σε όλα τα δείγματα κάθε παρτίδας και αφού τελειώσει με αυτό, υπολογίζει το σφάλμα συγκρίνοντας τις αναμενόμενες μεταβλητές εξόδου με τις προβλέψεις. Η τιμή του μεγέθους παρτίδας μπορεί να ποικίλλει και η επιλογή της κατάλληλης επηρεάζει την απόδοση του μοντέλου, τον χρόνο εκπαίδευσης και το πόσο καλά γενικεύει σε άγνωστα δεδομένα. Ένα μικρό μέγεθος παρτίδας παρέχει μια πιο ακριβή εκτίμηση του gradient και βοηθά στην αποφυγή τοπικών ελαχίστων, εισάγοντας περισσότερο θόρυβο στη διαδικασία gradient descent, αλλά επίσης οδηγεί σε πιο αργή εκπαίδευση λόγω λιγότερο αποδοτικών υπολογισμών και απαιτεί πιο συχνές ενημερώσεις των παραμέτρων του μοντέλου. Από την άλλη, ένα μεγάλο μέγεθος παρτίδας αξιοποιεί πλήρως τις παράλληλες αρχιτεκτονικές υλικού, οδηγώντας σε ταχύτερους υπολογισμούς, και παρέχει πιο ομαλές και σταθερές ενημερώσεις των gradients, αλλά μπορεί να οδηγήσει σε φτωχότερη γενίκευση λόγω πιο ομαλών καμπυλών απωλειών και ενδεχομένως εγκλωβισμού σε απότομα τοπικά ελάχιστα.

Συνέλιξη (Convolution)

Η συνέλιξη [25] είναι μια μαθηματική πράξη που εφαρμόζεται σε δύο συναρτήσεις (όπως ένα φίλτρο ή πυρήνα και μια είσοδο) για να παραχθεί μια τρίτη συνάρτηση που εκφράζει πώς το σχήμα της μιας τροποποιείται από την άλλη. Σε απλούστερους όρους, είναι ένας τρόπος εφαρμογής ενός φίλτρου (ή πυρήνα) σε μια είσοδο για να δημιουργηθεί ένας χάρτης χαρακτηριστικών που αναδεικνύει ορισμένες πτυχές ή χαρακτηριστικά της εισόδου.

$$v_{ij}^{xy} = \tanh(b_{ij} + \sum_{m} \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} w_{ij}^{pq} v_{(i-1)m}^{(x+p)(y+q)}) \tag{0.0.11}$$

όπου το tanh είναι η υπερβολική εφαπτομένη, το b_{ij} είναι η μετατόπιση (bias) για αυτόν τον χάρτη χαρακτηριστικών, το m είναι το δείκτης πάνω στο σύνολο των χαρτών χαρακτηριστικών στο (i - 1)ο στρώμα που συνδέεται με τον τρέχοντα χάρτη χαρακτηριστικών, το w_{ijk}^{pq} είναι η τιμή στη θέση (p, q) του πυρήνα που συνδέεται με τον kο χάρτη χαρακτηριστικών, και τα P_i και Q_i είναι το ύψος και το πλάτος του πυρήνα, αντίστοιχα.

Ο πυρήνας είναι χωρικά μικρότερος από την εικόνα, αλλά έχει μεγαλύτερο βάθος. Αυτό σημαίνει ότι, αν η εικόνα αποτελείται από τρία κανάλια (RGB), το ύψος και το πλάτος του πυρήνα θα είναι χωρικά μικρά, αλλά το βάθος εκτείνεται σε όλα τα κανάλια.

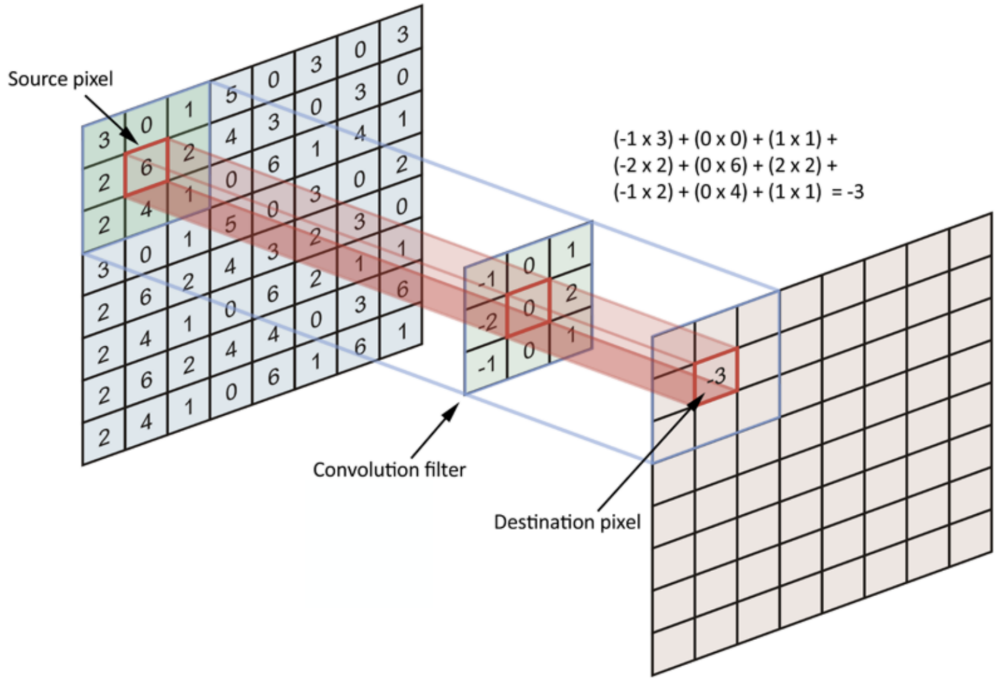


Figure 0.0.3: Οπτικοποίηση της συνέλιξης από [9].

Η συνέλιξη χρησιμοποιείται στα CNN για την επεξεργασία μιας εικόνας και την αναγνώριση συγκεκριμένων χαρακτηριστικών σε αυτή. Σε αντίθεση με ένα κανονικό νευρωνικό δίκτυο, τα στρώματα ενός CNN έχουν νευρώνες διατεταγμένους σε τρεις διαστάσεις: πλάτος, ύψος, βάθος. Οι νευρώνες σε ένα στρώμα συνδέονται μόνο με μια μικρή περιοχή του προηγούμενου στρώματος και όχι με όλους τους νευρώνες με πλήρως συνδεδεμένο τρόπο. Ένα απλό CNN είναι μια ακολουθία στρωμάτων, και κάθε στρώμα μετατρέπει έναν όγκο ενεργοποιήσεων σε έναν άλλο μέσω μιας διαφορίσιμης συνάρτησης. Οι τύποι στρωμάτων που χρησιμοποιούνται για τη δημιουργία της αρχιτεκτονικής ενός CNN περιλαμβάνουν: το Συνελικτικό Στρώμα (Convolutional Layer), το Στρώμα Υποδειγματοληψίας (Pooling Layer) και το Πλήρως Συνδεδεμένο Στρώμα (Fully-Connected Layer).

Λόγω του μεγάλου αριθμού pixel σε μια εικόνα, κάθε νευρώνας έχει ένα πεδίο υποδοχής που εφαρμόζεται μόνο στα αντίστοιχα στρώματα. Επομένως, έχει $w \times h \times c$ βάρη, όπου $w \times h$ είναι το πεδίο υποδοχής και το c είναι ο αριθμός των καναλιών της εικόνας εισόδου. Αυτός είναι και ο λόγος που τα συνελικτικά νευρωνικά δίκτυα δεν ανιχνεύουν τόσο επαρκώς την περιεχομενική πληροφορία όσο οι μηχανισμοί προσοχής (attention mechanisms). Το βάθος του φίλτρου σε κάθε πυρήνα, υποδεικνύει ότι υπάρχουν πολλοί νευρώνες που εφαρμόζονται στο ίδιο κομμάτι της εικόνας εισόδου. Επομένως, η εικόνα εξόδου είναι επίσης τρισδιάστατη, με βάθος ίσο με το βάθος του φίλτρου. Ο σκοπός των πολλαπλών διαδοχικών φίλτρων είναι να ανιχνεύουν πολλαπλά χαρακτηριστικά με ένα συνελικτικό στρώμα. Παρά το γεγονός ότι κάθε νευρώνας έχει συγκεκριμένο πεδίο υποδοχής, με την ολίσθηση του φίλτρου και τον υπολογισμό της εξόδου από κάθε τμήμα της εικόνας εισόδου, δεν υπάρχει ανάγκη για πολλαπλούς νευρώνες για κάθε τμήμα. Η ολίσθηση καθορίζεται από το stride του φίλτρου, που υποδεικνύει πόσα pixel μετακινείται το φίλτρο κατά μήκος της εικόνας πριν επαναληφθεί η εφαρμογή του. Επιπλέον, ακόμη μία σημαντική παράμετρος είναι το padding, το οποίο μπορεί να διαδραματίσει σημαντικό ρόλο στον έλεγχο των διαστάσεων της εξόδου. Το padding αναφέρεται στην προσθήκη επιπλέον pixel γύρω από την εικόνα εισόδου, συνήθως μηδενικά, το οποίο έχει ως αποτέλεσμα την αύξηση του τελικού χάρτη χαρακτηριστικών σε σχέση με τη χρήση χωρίς padding. Τέλος, εάν έχουμε μια είσοδο $w \times w \times D$ και D_{out} αριθμό πυρήνων με χωρικό μέγεθος F, stride S και ποσότητα padding P, τότε το μέγεθος του όγκου εξόδου μπορεί να προσδιοριστεί με τον ακόλουθο τύπο: $\frac{W-F+2P}{S} + 1$.

Στρώμα Υποδειγματοληψίας (Pooling Layer)

Το στρώμα υποδειγματοληψίας χρησιμοποιείται για να αντικαταστήσει την έξοδο του δικτύου σε συγκεκριμένες θέσεις, εξάγοντας μια στατιστική σύνοψη από τις κοντινές εξόδους. Αυτό βοηθά στη μείωση του χωρικού

μεγέθους της αναπαράστασης, το οποίο μειώνει την απαιτούμενη ποσότητα υπολογισμών και βαρών. Η λειτουργία υποδειγματοληψίας εκτελείται σε κάθε φέτα της αναπαράστασης ξεχωριστά.

Υπάρχουν πολλές συναρτήσεις υποδειγματοληψίας, όπως ο μέσος όρος της ορθογώνιας περιοχής, η νόρμα L2 της ορθογώνιας περιοχής και ο σταθμισμένος μέσος όρος βάσει της απόστασης από το κεντρικό pixel. Ωστόσο, η πιο δημοφιλής διαδικασία είναι το max pooling, που συλλέγει τη μέγιστη έξοδο από τη γειτονιά ενός πυρήνα.

Κανονικοποίηση Κατά Παρτίδες (Batch Normalization)

Η κανονικοποίηση κατά παρτίδες χρησιμοποιείται για να τυποποιήσει τις εισόδους σε οποιοδήποτε συγκεκριμένο στρώμα. Αυτό σημαίνει ότι η είσοδος έχει μηδενική μέση τιμή και μοναδιαία διακύμανση. Το στρώμα BN μετασχηματίζει κάθε είσοδο στο τρέχον mini-batch αφαιρώντας τη μέση τιμή εισόδου στο τρέχον mini-batch και διαιρώντας τη με την τυπική απόκλιση. Ωστόσο, δεν έχει αποδειχθεί ότι τα μοντέλα αποδίδουν καλύτερα με μηδενική μέση τιμή και μοναδιαία διακύμανση. Μπορεί να αποδίδουν καλύτερα με κάποια άλλη μέση τιμή και διακύμανση. Επομένως, το στρώμα BN εισάγει επίσης δύο εκπαιδεύσιμες παραμέτρους τις γ και β, οι οποίες προσαρμόζουν τη μέση τιμή και τη διακύμανση.

Πλήρως Συνδεδεμένο Στρώμα (Fully Connected Layer)

Στο πλήρως συνδεδεμένο στρώμα, οι νευρώνες έχουν πλήρη συνδεσιμότητα με όλους τους νευρώνες στο προηγούμενο και το επόμενο στρώμα, όπως φαίνεται στα κανονικά FCNN. Αυτός είναι ο λόγος που μπορεί να υπολογιστεί με τον συνηθισμένο τρόπο μέσω ενός πολλαπλασιασμού μητρών, ακολουθούμενου από μια επίδραση μετατόπισης (bias). Το στρώμα FC βοηθά στην αντιστοίχιση της αναπαράστασης μεταξύ της εισόδου και της εξόδου.

Δίκτυο χρονικών τμημάτων (Temporal Segment Network)

Το δίκτυο χρονικών τμημάτων, το οποίο παρουσιάστηκε για πρώτη φορά στο [75], είναι μια αρχιτεκτονική η οποία προσπαθεί να επιλύσει το πρόβλημα των σύνθετων δράσεων, όπως οι αθλητικές δράσεις, που λαμβάνουν χώρα σε πολλαπλά στάδια και εκτείνονται σε σχετικά μεγάλο χρονικό διάστημα, χρησιμοποιώντας την οπτική πληροφορία ολόκληρων των βίντεο για να πραγματοποιήσει προβλέψεις σε επίπεδο βίντεο.

Η Αρχιτεκτονική

Η αρχιτεκτονική χρονικών τμημάτων, όπως και άλλες χωροχρονικές μέθοδοι, αποτελείται από ConvNets χωρικής ροής και ConvNets χρονικής ροής. Ωστόσο, αντί να λειτουργεί σε μεμονωμένα καρέ ή ομάδες καρέ, το δίκτυο χρονικών τμημάτων λειτουργεί σε μια ακολουθία σύντομων αποσπασμάτων που λαμβάνονται αραιά από ολόκληρο το βίντεο. Κάθε απόσπασμα σε αυτή την ακολουθία θα παράγει τη δική του προκαταρκτική πρόβλεψη των κατηγοριών δράσης. Στη συνέχεια, θα εξαχθεί μια συναίνεση μεταξύ των αποσπασμάτων ως πρόβλεψη σε επίπεδο βίντεο, όπως φαίνεται στο Σχήμα 0.0.4. Στη διαδικασία εκμάθησης, οι τιμές απώλειας των προβλέψεων σε επίπεδο βίντεο, και όχι αυτές των προβλέψεων σε επίπεδο αποσπασμάτων που χρησιμοποιούνταν στα ConvNets δύο ροών, βελτιστοποιούνται με τη σταδιακή ενημέρωση των παραμέτρων του μοντέλου.

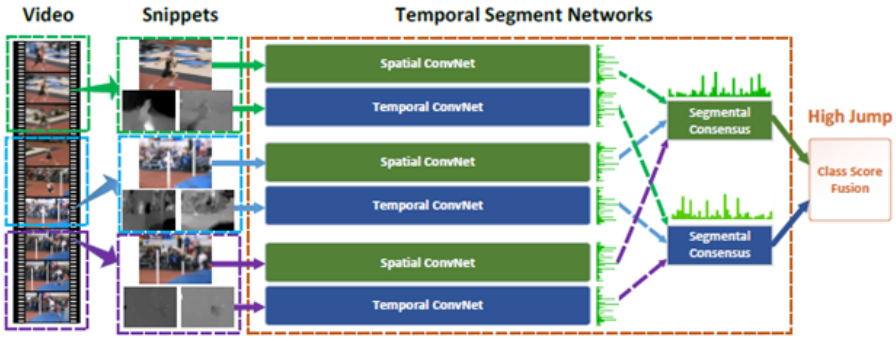


Figure 0.0.4: Αρχιτεκτονική Δικτύων Χρονικών Τμημάτων από [75].

Υπολειμματική μάθηση (Residual Learning)

Υπολειμματικό Μπλοκ (Residual Block)

Η υπολειμματική μάθηση (Residual Learning) παρουσιάστηκε για πρώτη φορά στο Σχήμα [30], προκειμένου να αντιμετωπίσει το πρόβλημα υποβάθμισης (degradation problem). Αν θεωρήσουμε το $H(x)$ ως μια υποκείμενη συνάρτηση που προσαρμόζεται από μερικά διαδοχικά στρώματα (όχι απαραίτητα ολόκληρο το δίκτυο), με το x να αναφέρεται στις εισόδους του πρώτου από αυτά τα στρώματα, τότε η υπολειμματική μάθηση υποθέτει ότι τα πολλαπλά μη γραμμικά στρώματα μπορούν ασυμπτωτικά να προσεγγίσουν τις υπολειμματικές συναρτήσεις, δηλαδή $H(x) - x$ (υποθέτοντας ότι οι είσοδοι και οι έξοδοι έχουν τις ίδιες διαστάσεις). Έτσι, αντί τα διαδοχικά στρώματα να προσεγγίζουν το $H(x)$, τους επιτρέπεται να προσεγγίζουν άμεσα μια υπολειμματική συνάρτηση $F(x) := H(x) - x$. Η αρχική συνάρτηση γίνεται επομένως $F(x) + x$, υποθέτοντας επίσης ότι είναι ευκολότερο να βελτιστοποιηθεί η υπολειμματική αντιστοίχιση από ό,τι η αρχική, μη αναφερόμενη αντιστοίχιση.

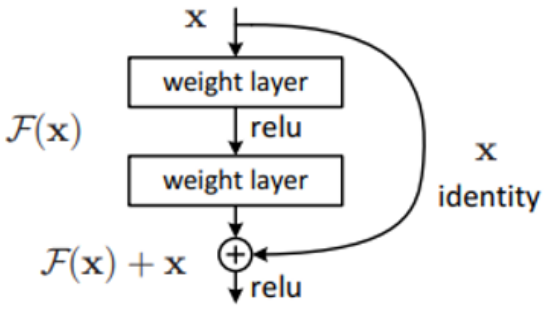


Figure 0.0.5: Υπολειμματικό Μπλοκ από [30].

Η διατύπωση του $F(x) + x$ μπορεί να επιτευχθεί μέσω τροφοδοτικών νευρωνικών δικτύων (feedforward neural networks) με «συνδέσεις συντόμευσης» (shortcut connections) που δεν προσθέτουν ούτε επιπλέον παραμέτρους ούτε υπολογιστική πολυπλοκότητα. Επιπλέον, το δομικό στοιχείο μπορεί να αντιπροσωπεύει πολλαπλά συνελικτικά στρώματα και μπορεί να χρησιμοποιηθεί άμεσα αν οι διαστάσεις εισόδου και εξόδου είναι ίδιες. Από την άλλη, αν οι διαστάσεις του x και του F δεν είναι ίσες, είτε μπορεί να προστεθεί padding είτε να πραγματοποιηθεί μια γραμμική προβολή W_s πάνω στο x.

$$y = F(x, W_i) + W_s \cdot x \tag{0.0.12}$$

3D ConvNets

3D Συνελίξεις

Στα δισδιάστατα CNNs (2D CNNs), οι συνελίξεις εφαρμόζονται στους 2D χάρτες χαρακτηριστικών για να υπολογιστούν χαρακτηριστικά μόνο από τις χωρικές διαστάσεις. Για αυτόν τον λόγο, όταν χρησιμοποιούνται σε προβλήματα ανάλυσης βίντεο, δεν καταφέρνουν να συλλάβουν τις πληροφορίες κίνησης που κωδικοποιούνται σε πολλαπλά συνεχόμενα καρέ, κάτι που είναι επιθυμητό. Ως αποτέλεσμα, οι 3D συνελίξεις προτάθηκαν [36] για να εκτελούνται στα στάδια συνελίξεων των CNNs, ώστε να υπολογίζονται χαρακτηριστικά τόσο από τις χωρικές όσο και από τις χρονικές διαστάσεις. Η 3D συνέλιξη επιτυγχάνεται με τη συνέλιξη ενός 3D πυρήνα στον κύβο που σχηματίζεται με τη στοίχιση πολλαπλών συνεχόμενων καρέ. Με αυτόν τον τρόπο, οι χάρτες χαρακτηριστικών συνελίξεων στο συνελικτικό στρώμα συνδέονται με πολλαπλά συνεχόμενα καρέ στο προηγούμενο στρώμα, καταγράφοντας πληροφορίες κίνησης. Η συνάρτηση που δίνει την τιμή στη θέση (x, y, z) στον j-οστό χάρτη χαρακτηριστικών στο i-οστό στρώμα είναι:

$$v_{ij}^{xyz} = \tanh(b_{ij} + \sum_{m} \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} \sum_{r=0}^{R_i-1} w_{ij}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)}) \tag{0.0.13}$$

όπου το $\tanh(\cdot)$ είναι η υπερβολική εφαπτομένη, το b_{ij} είναι η μετατόπιση για αυτόν τον χάρτη χαρακτηριστικών, το m είναι δείκτης πάνω στο σύνολο των χαρτών χαρακτηριστικών στο (i-1)ο στρώμα που συνδέεται με τον τρέχοντα χάρτη χαρακτηριστικών, το w_{pqr}^{ij} είναι η τιμή στη θέση (p, q, r) του πυρήνα που συνδέεται με τον m_{th} χάρτη χαρακτηριστικών στο προηγούμενο στρώμα, και τα R_i, P_i και Q_i είναι το μέγεθος του 3D πυρήνα κατά μήκος της χρονικής διάστασης, το ύψος και το πλάτος του πυρήνα, αντίστοιχα.

C3D

Για να βρεθεί μια καλή αρχιτεκτονική 3D ConvNet στο [72], ποίκιλλαν μόνο το χρονικό βάθος του πυρήνα των συνελικτικών στρωμάτων, ενώ κρατούσαν σταθερές όλες τις άλλες κοινές ρυθμίσεις, και πειραματίστηκαν με δύο τύπους αρχιτεκτονικών: 1) ομοιογενές χρονικό βάθος: όλα τα συνελικτικά στρώματα έχουν το ίδιο χρονικό βάθος πυρήνα. 2) μεταβλητό χρονικό βάθος: το χρονικό βάθος του πυρήνα αλλάζει σε κάθε στρώμα. Μετά την εκπαίδευση αυτών των δικτύων κατέληξαν στο συμπέρασμα ότι η ομοιογενής ρύθμιση με συνελικτικούς πυρήνες $3 \times 3 \times 3$ είναι η καλύτερη επιλογή για 3D ConvNets, κάτι που είναι σύμφωνο με παρόμοια ευρήματα στα 2D ConvNets [65]. Έτσι, σχεδιάστηκε ένα 3D ConvNet με όλους τους 3D συνελικτικούς πυρήνες του να έχουν μέγεθος $3 \times 3 \times 3$ με βήμα (stride) $1 \times 1 \times 1$. Επιπλέον, η αρχιτεκτονική του δικτύου αποτελούνταν από 8 συνελικτικά στρώματα, 5 στρώματα υποδειγματοληψίας (pooling layers), ακολουθούμενα από δύο πλήρως συνδεδεμένα στρώματα και ένα στρώμα εξόδου softmax. Η αρχιτεκτονική του δικτύου παρουσιάζεται στο Σχήμα 0.0.6 και ονομάζεται C3D. Επίσης, όλα τα 3D στρώματα υποδειγματοληψίας του είναι μεγέθους $2 \times 2 \times 2$ με βήμα $2 \times 2 \times 2$, εκτός από το pool1 που έχει μέγεθος πυρήνα $1 \times 2 \times 2$ και βήμα $1 \times 2 \times 2$, με σκοπό τη διατήρηση των χρονικών πληροφοριών στην αρχική φάση. Τέλος, κάθε πλήρως συνδεδεμένο στρώμα έχει 4096 μονάδες εξόδου.

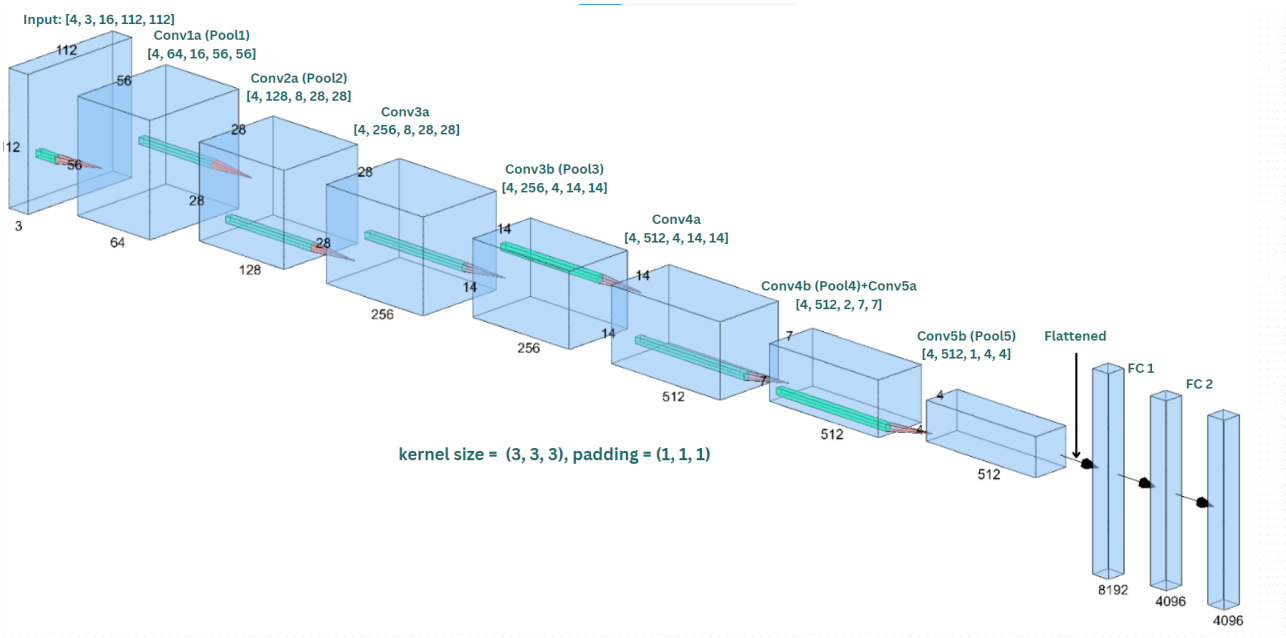


Figure 0.0.6: C3D αρχιτεκτονική από την 2D οπτική (spatial).

Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs)

Αρχιτεκτονική

Τα Επαναλαμβανόμενα Νευρωνικά Δίκτυα (Recurrent Neural Networks - RNNs) είναι ένας τύπος νευρωνικών δικτύων βαθιάς μάθησης που έχουν σχεδιαστεί για να αναγνωρίζουν μοτίβα σε ακολουθίες δεδομένων, όπως χρονοσειρές, ομιλία, κείμενο, βίντεο και άλλα ακολουθιακά δεδομένα. Σε αντίθεση με τα παραδοσιακά νευρωνικά δίκτυα τροφοδότησης (feedforward neural networks), τα RNNs έχουν συνδέσεις που γυρίζουν πίσω στον εαυτό τους, επιτρέποντάς τους να διατηρούν μια "μνήμη" των προηγούμενων εισόδων, κάτι που είναι κρίσιμο για εργασίες όπου η σειρά ή το πλαίσιο έχει σημασία.

Το κύριο χαρακτηριστικό της αρχιτεκτονικής των RNNs είναι η κρυφή κατάσταση, η οποία λειτουργεί ως μορφή μνήμης που αποθηκεύει πληροφορίες για ό,τι έχει επεξεργαστεί μέχρι στιγμής (τις προηγούμενες εισόδους). Σε κάθε βήμα μιας ακολουθίας, η κρυφή κατάσταση ενημερώνεται με βάση τόσο την τρέχουσα είσοδο όσο και την προηγούμενη κρυφή κατάσταση. Έτσι, όταν τα RNNs κάνουν την πρόβλεψή τους, χρησιμοποιούν τόσο την τρέχουσα είσοδο όσο και την αποθηκευμένη μνήμη για να προβλέψουν την επόμενη ακολουθία. Αυτό καθιστά τα RNNs ικανά να διατηρούν πληροφορίες σε διαδοχικά χρονικά βήματα.

Περιορισμοί

Παρόλο που τα RNNs έχουν αποδειχθεί πολύτιμα εργαλεία για ακολουθιακά προβλήματα, όπως οι εφαρμογές επεξεργασίας φυσικής γλώσσας (NLP), έχουν ακόμα αρκετούς περιορισμούς:

1. Εξαφανιζόμενη κλίση (Vanishing Gradient) Η κλίση (gradient) περιγράφει την ευαισθησία του ρυθμού σφάλματος του μοντέλου σε σχέση με τις παραμέτρους του. Η κλίση μπορεί να αναπαρασταθεί από την κλίση σε έναν λόφο· όσο μεγαλύτερη η κλίση, τόσο πιο απότομη η κάθοδος, και τόσο πιο γρήγορα ολοκληρώνεται η εκπαίδευση. Το πρόβλημα της εξασθενημένης κλίσης προκύπτει όταν οι τιμές της κλίσης γίνονται πολύ μικρές, και οι αντίστοιχες κλίσεις τόσο επίπεδες, κατά την εκπαίδευση, κάνοντας τα RNNs να αποτυγχάνουν να μάθουν αποτελεσματικά από τα δεδομένα εκπαίδευσης, οδηγώντας σε υποπροσαρμογή.

2. Εκρηκτική κλίση (Exploding Gradient)

   Από την άλλη, η εκρηκτική κλίση προκύπτει όταν η κλίση αυξάνεται εκθετικά μέχρι το RNN να γίνει ασταθές. Όταν οι κλίσεις γίνονται απεριόριστα μεγάλες, το RNN συμπεριφέρεται ακανόνιστα, με αποτέλεσμα προβλήματα όπως η υπερπροσαρμογή. Η υπερπροσαρμογή είναι ένα φαινόμενο όπου το μοντέλο μπορεί να προβλέπει με ακρίβεια στα δεδομένα εκπαίδευσης, αλλά δεν μπορεί να το κάνει με δεδομένα πραγματικού κόσμου.

3. Αργός χρόνος εκπαίδευσης

   Τα RNNs απαιτούν μεγάλη υπολογιστική ισχύ, χώρο μνήμης και χρόνο για να εκπαιδευτούν, καθώς χρειάζονται να αποθηκεύουν τις προηγούμενες εισόδους και να τις επεξεργάζονται κατά τη διάρκεια του χρόνου εκπαίδευσης.

Δίκτυα Μακράς Βραχυπρόθεσμης Μνήμης (Long Short-term Memory - LSTMs)

Τα LSTM είναι μια παραλλαγή των RNN, που επιτρέπει στο μοντέλο να επεκτείνει τη μνήμη του ώστε να μπορεί να διαχειρίζεται μεγαλύτερο χρονικό ορίζοντα. Ένα RNN μπορεί να θυμάται μόνο την αμμέσως προηγούμενη είσοδο του, επομένως δεν μπορεί να χρησιμοποιήσει εισόδους από προηγούμενες ακολουθίες για να βελτιώσει την πρόβλεψή του, σε αντίθεση με το LSTM.

Η αρχιτεκτονική του LSTM περιλαμβάνει το κελί μνήμης, το οποίο ελέγχεται από τρεις πύλες: την πύλη εισόδου, την πύλη λήθης και την πύλη εξόδου. Αυτές οι πύλες αποφασίζουν ποιες πληροφορίες θα προστεθούν, θα διαγραφούν ή θα εξάγονται από το κελί μνήμης.

Πύλη Λήθης (Forget Gate)

Η πύλη λήθης αποφασίζει ποιες πληροφορίες θα διαγραφούν από την κατάσταση του κελιού. Λαμβάνει ως είσοδο την προηγούμενη κρυφή κατάσταση (h_{t-1}) και την τρέχουσα είσοδο (x_t), και τις πολλαπλασιάζει με πίνακες βαρών, ακολουθούμενες από την πρόσθεση της προκατάληψης (bias). Το αποτέλεσμα περνά στη συνέχεια από μια συνάρτηση ενεργοποίησης, η οποία παράγει μια έξοδο είτε 0 είτε 1 για κάθε αριθμό στην κατάσταση του

κελιού C_{t-1}. Μια τιμή 1 σημαίνει "διατήρησε πλήρως αυτό", ενώ μια τιμή 0 σημαίνει "ξέχασε πλήρως αυτό". Η πύλη λήθης ορίζεται από την εξής εξίσωση:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{0.0.14}$$

όπου W_f είναι ο πίνακας βαρών για την πύλη λήθης, το b_f είναι ο όρος της προκατάληψης, το σ είναι η συνάρτηση ενεργοποίησης sigmoid και το $[h_{t-1},\ x_t]$ δηλώνει τη συνένωση της τρέχουσας εισόδου και της προηγούμενης κρυφής κατάστασης.

Πύλη Εισόδου (Input Gate)

Η πύλη εισόδου καθορίζει ποιες τιμές θα ενημερωθούν στην κατάσταση του κελιού. Αποτελείται από δύο μέρη: το επίπεδο πύλης εισόδου και ένα επίπεδο που δημιουργεί ένα διάνυσμα νέων υποψήφιων τιμών που θα μπορούσαν να προστεθούν στην κατάσταση. Το επίπεδο πύλης εισόδου χρησιμοποιεί τη συνάρτηση sigmoid για να φιλτράρει τις τιμές που θα διατηρηθούν, παρόμοια με την πύλη λήθης, χρησιμοποιώντας τις εισόδους h_{t-1} και x_t. Το επίπεδο νέων υποψήφιων τιμών δημιουργεί ένα διάνυσμα χρησιμοποιώντας τη συνάρτηση tanh, που δίνει έξοδο από -1 έως +1, περιέχοντας όλες τις πιθανές τιμές από h_{t-1} και x_t. Τελικά, οι τιμές του διανύσματος και οι ρυθμισμένες τιμές πολλαπλασιάζονται για να προκύψει η χρήσιμη πληροφορία. Η εξίσωση για την πύλη εισόδου είναι:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{0.0.15}$$

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{0.0.16}$$

Πολλαπλασιάζουμε την προηγούμενη κατάσταση με f_t, αγνοώντας τις πληροφορίες που είχαμε προηγουμένως επιλέξει να αγνοήσουμε. Στη συνέχεια, προσθέτουμε $i_t \odot \hat{C}_t$. Αυτό αντιπροσωπεύει τις ενημερωμένες υποψήφιες τιμές, προσαρμοσμένες στην ποσότητα που επιλέξαμε να ενημερώσουμε κάθε τιμή κατάστασης.

$$C_t = f_t \odot C_{t-1} + i_t \odot \hat{C}_t \tag{0.0.17}$$

Πύλη Εξόδου (Output Gate)

Η πύλη εξόδου αποφασίζει ποια θα είναι η επόμενη κρυφή κατάσταση h_t, η οποία θα χρησιμοποιηθεί τόσο για το επόμενο χρονικό βήμα όσο και για την έξοδο του τρέχοντος κελιού. Πρώτα, δημιουργείται ένα διάνυσμα εφαρμόζοντας τη συνάρτηση tanh στο κελί. Στη συνέχεια, οι πληροφορίες ρυθμίζονται χρησιμοποιώντας τη συνάρτηση sigmoid και φιλτράρονται από τις τιμές που θα θυμόμαστε, χρησιμοποιώντας τις εισόδους h_{t-1} και x_t. Τέλος, οι τιμές του διανύσματος και οι ρυθμισμένες τιμές πολλαπλασιάζονται για να χρησιμοποιηθούν ως έξοδο και ως είσοδος για το επόμενο κελί. Η εξίσωση που περιγράφει την πύλη εξόδου είναι:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{0.0.18}$$

Σχετική βιβλιογραφία

Ανίχνευση Στοχοπροσήλωσης(engagement)

Το πρόβλημα της στοχοπροσήλωσης των παιδιών έχει προσεγγιστεί με διάφορες μεθόδους και τεχνικές για την αξιοποίηση των διαφορετικών πτυχών του. Λόγω της δυναμικής φύσης του προβλήματος, που περιλαμβάνει χωρικές και χρονικές διαστάσεις, αυτές οι μέθοδοι χρησιμοποιούν κυρίως χωροχρονικά νευρωνικά δίκτυα. Αυτά αποτελούνται από δύο κύρια μέρη: το πρώτο εξάγει χωρικά χαρακτηριστικά από δεδομένα όπως εικόνες ή καρέ βίντεο, ενώ το δεύτερο εξετάζει την αλλαγή των πληροφοριών στον χρόνο. Χρησιμοποιούνται διάφορα δίκτυα όπως RNNs [63], LSTMs [32], GRUs [14] και TCNs [5], που είναι σχεδιασμένα για τη διαχείριση ακολουθιακών δεδομένων και χρονικών εξαρτήσεων. Τέλος, τα χωρικά και χρονικά χαρακτηριστικά συνδυάζονται με διάφορους τρόπους, όπως τα 3D Convolutional Networks και ConvLSTM Networks.

Χωροχρονικές οπτικές ενδείξεις

Όσον αφορά την ανίχνευση της στοχοπροσήλωσης, στο [3], προτείνεται μια νέα αρχιτεκτονική που βασίζεται στο χωροχρονικό συνελικτικό μπλοκ R(2+1)D για την αναγνώριση δράσεων. Αυτό το μπλοκ αποτελείται από χωροχρονικές συνελίξεις που εκτελούν δισδιάστατη συνέλιξη στο επίπεδο του χώρου ακολουθούμενη από μονοδιάστατη συνέλιξη κατά μήκος του άξονα του χρόνου. Αυτές οι χωροχρονικές συνελίξεις επαναλαμβάνονται δύο φορές και συνδυάζονται σε ένα υπολειμματικό (residual) μπλοκ. Το δίκτυο αποτελείται από πέντε διαδοχικά επίπεδα που δημιουργούνται από αυτά τα χωροχρονικά μπλοκ. Οι χωροχρονικές συνελικτικές στρώσεις ακολουθούνται από ένα επίπεδο προσαρμοστικής συγκέντρωσης (adaptive pooling layer) και τέλος ένα πλήρως συνδεδεμένο επίπεδο. Η σταθμισμένη συνάρτηση απώλειας (weighted CE loss function) χρησιμοποιείται για τον υπολογισμό της απώλειας του δικτύου. Διάφορα πειράματα διεξήχθησαν χρησιμοποιώντας ακατέργαστα δεδομένα RGB, μια σύμπτηξη δεδομένων RGB και βάθους, καθώς και μια σύμπτηξη ακατέργαστων δεδομένων RGB και δεδομένων οπτικής ροής. Εκτός από αυτά, μελετήθηκε και η σύμπτηξη των δικτύων RGB και οπτικής ροής τόσο σε πρώιμο όσο και σε μεταγενέστερο στάδιο του δικτύου, με το μεταγενέστερο να είναι πριν από το τελευταίο πλήρως συνδεδεμένο επίπεδο και το πρώιμο μετά το πρώτο από τα πέντε χωροχρονικά συνελικτικά επίπεδα. Τέλος, προέκυψαν ενθαρρυντικά αποτελέσματα σε όλα τα σύνολα δεδομένων στα οποία δοκιμάστηκε η αρχιτεκτονική, με το δίκτυο που συνέδεσε τα ακατέργαστα RGB και την οπτική ροή σε πρώιμο στάδιο να έχει τον καλύτερο συνδυασμό ακρίβειας (accuracy, f1-score και σταθμισμένης ακρίβειας (weighted precision) σε σύγκριση με τα άλλα.

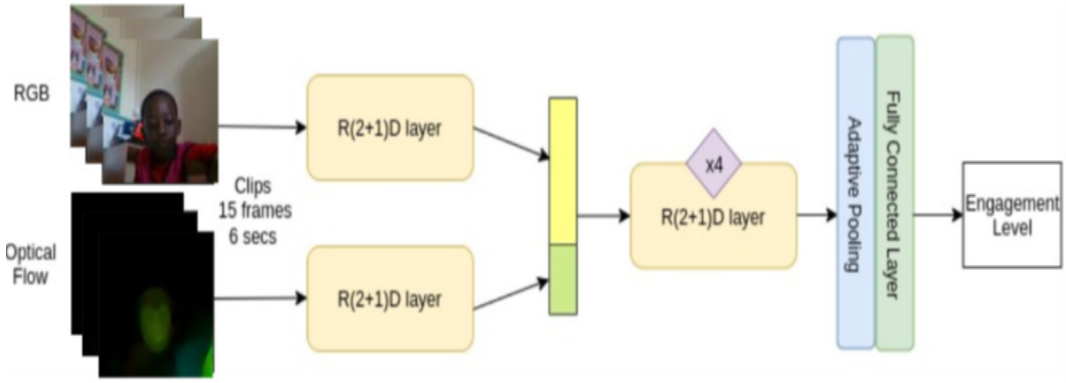


Figure 0.0.7: Δίκτυο για την εκτίμηση του engagement χρησιμοποιώντας χωροχρονικές συνελίξεις και συμπτύσσοντας τα δεδομένα RGB και οπτικής ροής σε πρώιμο στάδιο.

Resnet και TCN Υβριδικό Δίκτυο

Στο [1], υλοποιείται μια νέα αρχιτεκτονική νευρωνικού δικτύου Residual Network (ResNet) και Temporal Convolutional Network (TCN) για την ανίχνευση του επιπέδου εμπλοκής (engagement) των μαθητών σε βίντεο. Σε αυτή την αρχιτεκτονική, το 2D ResNet εξάγει τα χωρικά χαρακτηριστικά από μια ακολουθία ακατέργαστων

καρέ του βίντεο (frames) και στη συνέχεια το TCN αναλύει τις χρονικές αλλαγές σε αυτά για να ανιχνευθεί το engagement. Το TCN επιλέγεται καθώς υπερέχει στη μοντελοποίηση ακολουθιών μεγαλύτερου μήκους και στη διατήρηση μνήμης του ιστορικού σε σύγκριση με γενικές επαναλαμβανόμενες αρχιτεκτονικές όπως τα LSTM και GRU. Επίσης, όπως αναφέρθηκε και προηγουμένως, επιλέγεται και στη συγκεκριμένη μελέτη ένας συνδυασμός ενός χωρικού και ενός χρονικού μοντέλου, καθώς το engagement είναι μια χωροχρονική συναισθηματική κατάσταση που λαμβάνει χώρα σε διαδοχικά καρέ βίντεο με την πάροδο του χρόνου. Η αρχιτεκτονική του υβριδικού δικτύου παρουσιάζεται στο Σχήμα 0.0.8.

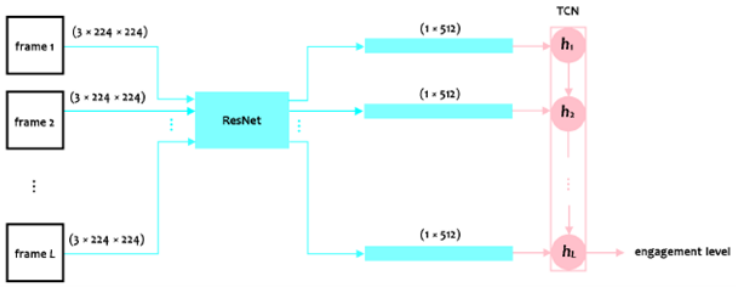


Figure 0.0.8: ResNet+TCN network architecture.

CNN και Αναδρομικό Δίκτυο (Recurrent Network)

Στο [19] προτείνεται μια προσέγγιση βαθιάς μάθησης για την εκτίμηση του engagement από ακολουθίες βίντεο, η οποία αποτελείται από δύο κύριες ενότητες: μια συνελικτική ενότητα που εξάγει χαρακτηριστικά εικόνας ανά καρέ (frame) και μια αναδρομική ενότητα που συναθροίζει τα χαρακτηριστικά των frames με την πάροδο του χρόνου για να παράγει έναν χρονικό διανύσμα χαρακτηριστικών της δράσης στο βίντεο.

Το συνελικτικό τμήμα του μοντέλου είναι ένα ResNetXt-50 Συνελικτικό Νευρωνικό Δίκτυο (CNN) [81] που έχει προ-εκπαιδευτεί στο σύνολο δεδομένων ImageNet [20]. Τα χαρακτηριστικά του κάθε frame λαμβάνονται από την ενεργοποίηση του τελευταίου πλήρως συνδεδεμένου επιπέδου του CNN, με διάσταση 2048, πριν από το softmax επίπεδο. Από την άλλη, η αναδρομική (recurrent) ενότητα είναι ένα Αναδρομικό Δίκτυο Μνήμης Μακράς Βραχυπρόθεσμης Διάρκειας (LSTM) [31] με 2048 μονάδες ακολουθούμενη από ένα Πλήρως Συνδεδεμένο (FC) επίπεδο μεγέθους 2048×1. Το LSTM λαμβάνει ως είσοδο μια ακολουθία χαρακτηριστικών από w frames που προέρχονται από τη συνελικτική ενότητα και παράγει, με τη σειρά του, ένα διάνυσμα χαρακτηριστικών που αντιπροσωπεύει ολόκληρη την ακολουθία των frames, ώστε να αποτυπώσει τη χρονική συμπεριφορά των ανθρώπων εντός του χρονικού παραθύρου w. Τα χρονικά χαρακτηριστικά περνούν από το FC επίπεδο με συνάρτηση ενεργοποίησης sigmoid στο τέλος για την παραγωγή των τελικών τιμών εκτίμησης βλεμμάτων. Κατά τη διάρκεια των πειραμάτων, το στρώμα του CNN είναι σταθερό, ενώ το αναδρομική τμήμσ εκπαιδεύεται για να προβλέπει τις τιμές του engagement από τις παρεχόμενες ετικέτες των δειγμάτων.

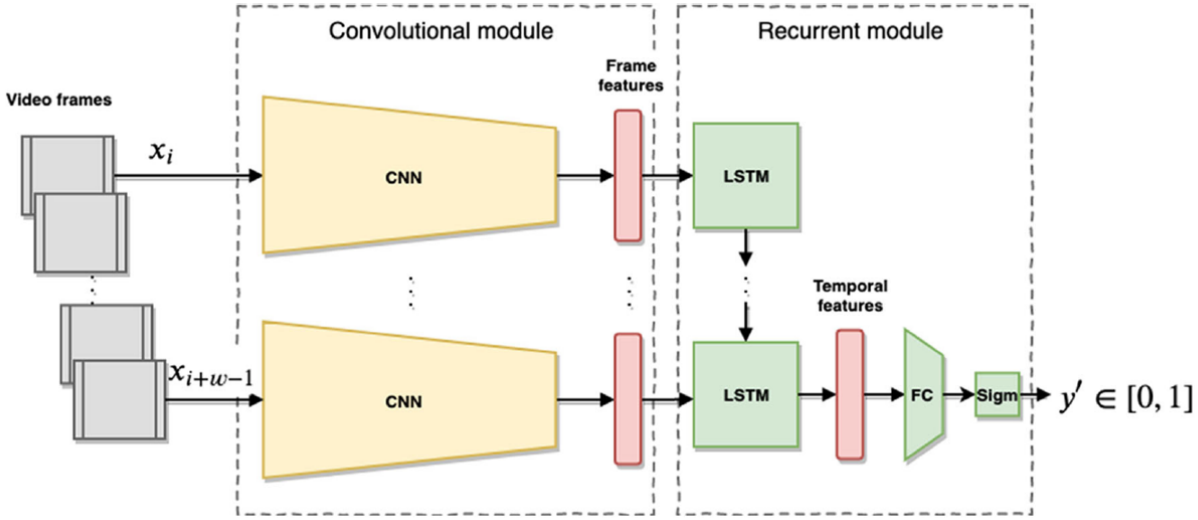


Figure 0.0.9: Επισκόπηση της προτεινόμενης αρχιτεκτονικής.

Τα αποτελέσματα αυτής της μελέτης έδειξαν ότι η συγκεκριμένη αρχιτεκτονική βαθιάς μάθησης δύο σταδίων, εκπαιδευμένη στο σύνολο δεδομένων TOGURO, είναι ένα κατάλληλο υπολογιστικό μοντέλο για την αποτύπωση της εγγενούς ανθρώπινης ερμηνείας της εμπλοκής (engagement). Όχι μόνο αυτό, αλλά μπορεί επίσης να εφαρμοστεί με επιτυχία σε ένα εντελώς διαφορετικό σενάριο, καθώς είναι αρκετά γενικό, όπως στο σύνολο δεδομένων UE-HRI [8], παρουσιάζοντας την εφαρμοσιμότητα του μοντέλου σε διαφορετικά περιβάλλοντα, με ένα διαφορετικό ρομπότ με διαφορετική κάμερα και με διαφορετικές εργασίες και ανθρώπους.

Εκτίμηση του βλέμματος

Η ικανότητα κατανόησης του σημείου στο οποίο κοιτάζει ένα άτομο, γνωστή ως ανίχνευση βλέμματος, παρέχει ανεκτίμητες πληροφορίες σε διάφορους τομείς έρευνας, όπως η αλληλεπίδραση ανθρώπου-υπολογιστή, οι υποστηρικτικές τεχνολογίες, η ψυχολογία και το μάρκετινγκ. Οι αρχιτεκτονικές που σχεδιάζονται για την εκτίμηση της κατεύθυνσης του βλέμματος συχνά χωρίζονται σε δύο κατηγορίες: τις μεθόδους που βασίζονται στην εμφάνιση (appearance based methods) και τις μεθόδους που βασίζονται σε μοντέλα (model-based). Οι μέθοδοι που βασίζονται στην εμφάνιση χρησιμοποιούν χαρακτηριστικά εικόνας και μοντέλα μηχανικής μάθησης για να εκτιμήσουν την κατεύθυνση του βλέμματος, ενώ οι μέθοδοι που βασίζονται σε μοντέλα χρησιμοποιούν γεωμετρικά μοντέλα του ματιού για να υπολογίσουν την κατεύθυνση του βλέμματος χρησιμοποιώντας σημεία αναφοράς του ματιού. Στην παρούσα μελέτη, θα επικεντρωθούμε στις μεθόδους που βασίζονται στην εμφάνιση, οι οποίες πρόσφατα έχουν βελτιώσει σημαντικά τις επιδόσεις των συστημάτων ανίχνευσης, λόγω των εξελίξεων στη βαθιά μάθηση και στις αρχιτεκτονικές της υπολογιστικής όρασης.

Μοντέλο Gaze360

Στην έρευνα [39] προτείνεται ένα μοντέλο που χρησιμοποιεί αμφίδρομες κάψουλες Long Short-Term Memory (LSTM) για να αντιμετωπίσει το πρόβλημα της εκτίμησης του βλέμματος. Σε αυτό το μοντέλο, η έξοδος που παράγεται για ένα στοιχείο εξαρτάται τόσο από τις προηγούμενες όσο και από τις μελλοντικές εισόδους, καθώς μια ακολουθία 7 καρέ χρησιμοποιείται για να προβλέψει το βλέμμα του κεντρικού καρέ. Καθένα από τα 7 καρέ που αποτελούν την είσοδο του μοντέλου είναι μια αποκοπή του κεφαλιού του ατόμου, του οποίου το βλέμμα πρόκειται να ανιχνευθεί, και στο πρώτο στάδιο του μοντέλου, κάθε ένα από αυτά επεξεργάζεται μεμονωμένα από ένα συνελικτικό νευρωνικό δίκτυο (backbone). Η έξοδος του backbone είναι τα χαρακτηριστικά υψηλού επιπέδου 256 διαστάσεων για κάθε καρέ, που στη συνέχεια χρησιμοποιούνται ως είσοδος για το αμφίδρομο LSTM, το οποίο αποτελείται από δύο στρώματα που επεξεργάζονται την αλληλουχία με προς τα εμπρός και προς τα πίσω διανύσματα. Τέλος, αυτά τα διανύσματα ενώνονται και περνούν από ένα πλήρως συνδεδεμένο επίπεδο

για να παραχθούν δύο έξοδοι: η πρόβλεψη του βλέμματος και η εκτίμηση ποσοστιαίου σφάλματος. Η πρόβλεψη του βλέμματος παράγεται σε σφαιρικές συντεταγμένες και η εκτίμηση ποσοστιαίου σφάλματος (σ) είναι ένα μέτρο της διαφοράς μεταξύ των 10 και 90 ποσοστών και της αναμενόμενης τιμής.

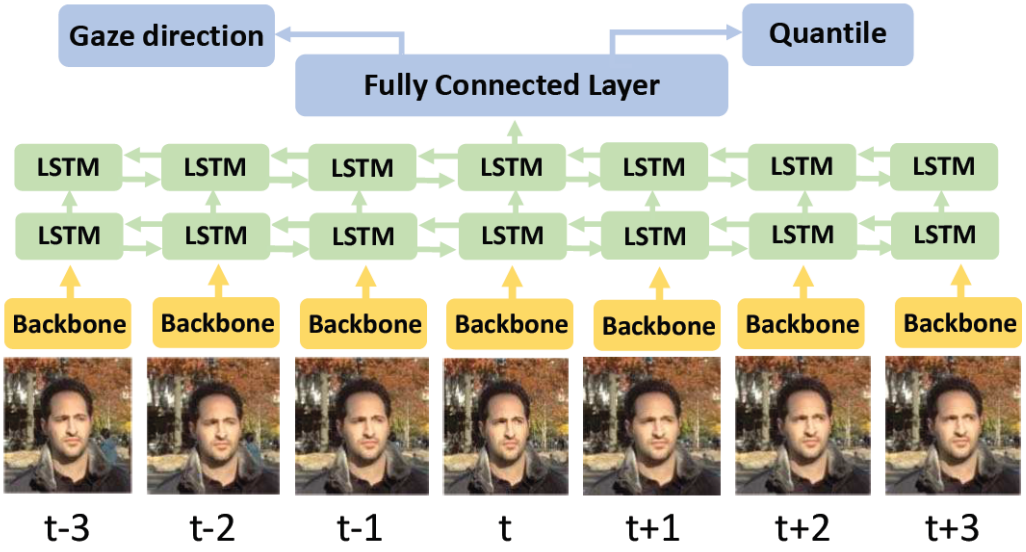


Figure 0.0.10: Η αρχιτεκτονική του μοντέλου Gaze360.

VGG-16 και RNN

Το προτεινόμενο μοντέλο στην [57] χωρίζεται σε 3 στάδια: (1) Ατομικό, (2) Συγχώνευση και (3) Χρονικό. Πρώτα, η Ατομική υπομονάδα μαθαίνει χαρακτηριστικά από κάθε οπτική ένδειξη ξεχωριστά (πρόσωπο και μάτια). Αποτελείται από ένα CNN δύο ροών, όπου η μία ροή είναι αφιερωμένη στην κανονικοποιημένη ροή εικόνας προσώπου και η άλλη στην κοινή κανονικοποιημένη εικόνα ματιών. Κάθε ροή της Ατομικής υπομονάδας βασίζεται στο βαθύ δίκτυο VGG-16 [60], που αποτελείται από 13 συνελικτικά επίπεδα, 5 επίπεδα max pooling και 1 πλήρως συνδεδεμένο (FC) επίπεδο με ενεργοποιήσεις Rectified Linear Unit (ReLU). Στη συνέχεια, η υπομονάδα Συγχώνευσης συνδυάζει τα εξαγόμενα χαρακτηριστικά κάθε ροής σε ένα μόνο διάνυσμα μαζί με τις κανονικοποιημένες συντεταγμένες των σημείων αναφοράς (facial landmarks). Το Ατομικό και το στάδιο Συγχώνευσης, καθορίζουν το Στατικό μοντέλο, και εφαρμόζονται σε κάθε καρέ (frame) της ακολουθίας. Τέλος, τα παραγόμενα διανύσματα χαρακτηριστικών κάθε καρέ δίνονται ως είσοδος στη Χρονική υπομονάδα που βασίζεται σε ένα many-to-one επαναλαμβανόμενο δίκτυο. Αυτή η υπομονάδα εκμεταλλεύεται τη διαδοχική πληροφορία για να προβλέψει τις κανονικοποιημένες 2D γωνίες βλέμματος του τελευταίου καρέ της ακολουθίας, χρησιμοποιώντας ένα επίπεδο γραμμικής παλινδρόμησης που προστίθεται πάνω της.

Κατά την διάρκεια της εκπαίδευσης, η είσοδος του μοντέλου αποτελείται από $s = 4$ διαδοχικά καρέ, με το μοντέλο να προσπαθεί να εκτιμήσει την κατεύθυνση του βλέμματος του τελευταίου καρέ. Επιπλέον, το δίκτυο εκπαιδεύεται σε σταδιακά, ξεκινώντας με την εκπαίδευση του Στατικού μοντέλου και του επιπέδου παλινδρόμησης (regression layer) από άκρη σε άκρη σε κάθε μεμονωμένο καρέ των δεδομένων εκπαίδευσης πρώτα. Επίσης, τα συνελικτικά μπλοκ είναι προεκπαιδευμένα με το σύνολο δεδομένων VGG-Face [60], ενώ τα FC εκπαιδεύονται από την αρχή. Έπειτα, κατά τη διάρκεια του τελικού σταδίου της εκπαίδευσης, τα χαρακτηριστικά κάθε καρέ της ακολουθίας εξάγονται από την "παγωμένη" τώρα Ατομική υπομονάδα, και τροφοδοτούνται στις στρώσεις Συγχώνευσης όπου κάνουν fine-tuning, ενώ παράλληλα εκπαιδεύουν τη Χρονική υπομονάδα, αλλά και το νέο τελικό επίπεδο παλινδρόμησης από την αρχή.

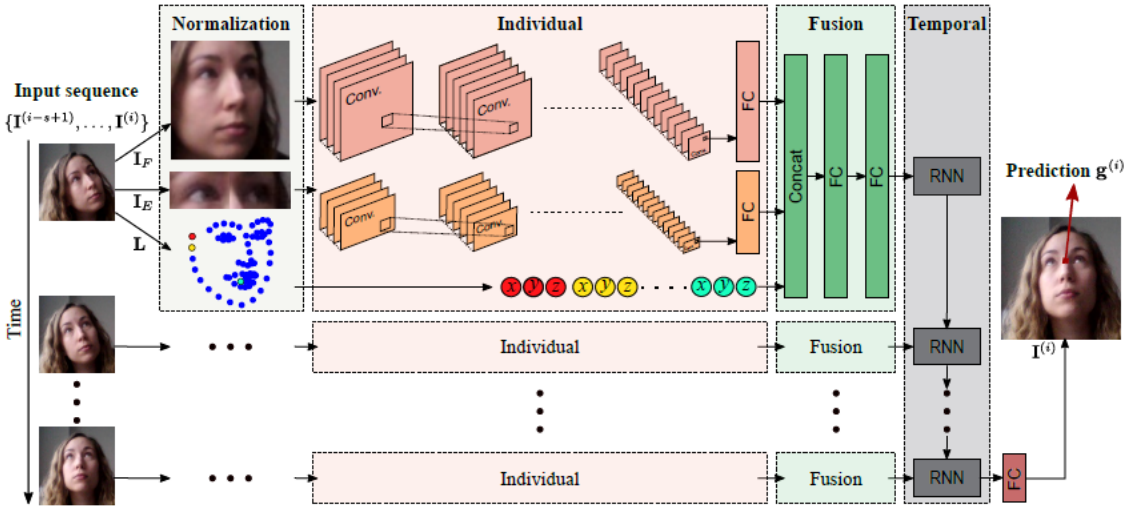


Figure 0.0.11: Η αρχιτεκτονική του προτεινόμενου Recurrent CNN δικτύου.

Multi-Clue Gaze (MCGaze)

Η μέθοδος Multi-Clue Gaze [26] είναι μια διαφορετική προσέγγιση που προτείνεται για την εκτίμηση της κατεύθυνσης του βλέμματος σε βίντεο, μέσω της παρατήρησης του χωροχρονικού πλαισίου αλληλεπίδρασης μεταξύ κεφαλιού, προσώπου και ματιών χρησιμοποιώντας μια query-based μέθοδο. Η βασική ιδέα της αρχιτεκτονικής του μοντέλου φαίνεται στο Σχήμα 0.0.12. Για κάθε εισαγόμενο κλιπ βίντεο ($\mathrm{I} \in R^{Tx3xHxW}$), το πρώτο βήμα είναι η εξαγωγή των χαρακτηριστικών από κάθε καρέ του, τροφοδοτώντας τα σε ένα backbone, ώστε να σχηματιστεί ένας τανυστής χαρακτηριστικών βίντεο ($\mathrm{F} \in R^{TxCxH'xW'}$). Το backbone που χρησιμοποιείται είναι το ResNet-50-FPN [30], προεκπαιδευμένο στο ImageNet-1K [20] και με την τιμή της παραμέτρου N ορισμένη σε 4. Στη συνέχεια, τα εξαγόμενα χαρακτηριστικά τροφοδοτούνται στην query-based αρχιτεκτονική, η οποία εκτελεί N επαναλήψεις και αποτελείται από δύο βασικά τμήματα: την αλληλεπίδραση χωροχρονικών queries και τα εξειδικευμένα κεφάλια (heads) για το εκάστοτε έργο (π.χ., clue localization head και gaze fusion head). Σε κάθε επανάληψη, τα queries για τα clues κεφαλιού, προσώπου και ματιού ενημερώνονται και το clue localization head προβλέπει την περιοχή του κεφαλιού, προσώπου και ματιού. Ταυτόχρονα, το gaze fusion head καθορίζει την κατεύθυνση του ανθρώπινου βλέμματος από τα clues κεφαλιού, προσώπου και ματιού. Το βλέμμα που προβλέπεται από την τελευταία επανάληψη χρησιμοποιείται ως η έξοδος του μοντέλου.

Για να συλλάβει τα χαρακτηριστικά του υποκειμένου, η συγκεκριμένη προσέγγιση εφαρμόζει queries πολλαπλών clues $q_{clue} \in R^{TC}$, clue ∈ κεφάλι, πρόσωπο, μάτι. Κάθε query αποτελείται από T ενσωματώσεις (embeddings) με διάσταση χαρακτηριστικών C, με κάθε embedding να επικεντρώνεται γενικά στην αναπαράσταση χαρακτηριστικών του αντίστοιχου καρέ. Επιπλέον, υπάρχουν προτεινόμενα κουτιά $p_{clue} \in R^{T4}$ που αντιστοιχούν σε κάθε query, τα οποία υποδεικνύουν τις τοποθεσίες του κεφαλιού, του προσώπου και του ματιού του υποκειμένου στον χάρτη χαρακτηριστικών. Οι παράμετροι των q_{clue} και p_{clue} είναι εκπαιδευόμενες (learnable). Για κάθε πλήρη προώθηση προς τα εμπρός, ενημερώνονται με επαναληπτικό τρόπο για να επιτύχουν αποτελεσματική εξαγωγή των clues-στόχων και των εκτιμήσεων της κατεύθυνσης του βλέμματος.

Επιπλέον, η τοπικο-καθολική (local-global) χωροχρονική μοντελοποίηση είναι μεγάλης σημασίας για την μελέτη προβλημάτων σε βίντεο [79], [33], έτσι σε αυτήν τη μελέτη σχεδιάστηκαν επίσης συγκεκριμένα queries για τα τρία βασικά clues του αντικειμένου. Συγκεκριμένα, χρησιμοποίησαν μια μονάδα αλληλεπίδρασης χωροχρονικών queries [82] για καλύτερο εντοπισμό των ιεραρχικών clues και τη δημιουργία μιας αποτελεσματικής ανταλλαγής πληροφοριών για ανθεκτικές αναπαραστάσεις βλέμματος. Σε αυτήν τη μονάδα, η χωρική αλληλεπίδραση μεταξύ των queries κεφαλιού, προσώπου και ματιού στο ίδιο καρέ επιτυγχάνεται μέσω της χρήσης ενός χωρικού πολυκεφαλικού στρώματος αυτοπροσοχής (spatial multi-head self-attention layer) [73], καθιστώντας τα queries να διαθέτουν τόσο καθολικές (global) όσο και τοπικές χωρικές προοπτικές για τον χαρακτηρισμό του βλέμματος. Επιπλέον, εφαρμόζεται ένα στρώμα αυτοπροσοχής (self-attention layer) που επιτρέπει την αλληλεπίδραση των queries κατά μήκος της χρονικής διάστασης, η οποία βοηθά στη μοντελοποίηση ακολουθίας διακριτών

χαρακτηριστικών, όπως η μεταβολή της στάσης και η κίνηση των ματιών, και διευκολύνει τη χρονική συνέπεια, οδηγώντας σε ανθεκτικό εντοπισμό clues και εκτίμηση βλέμματος.

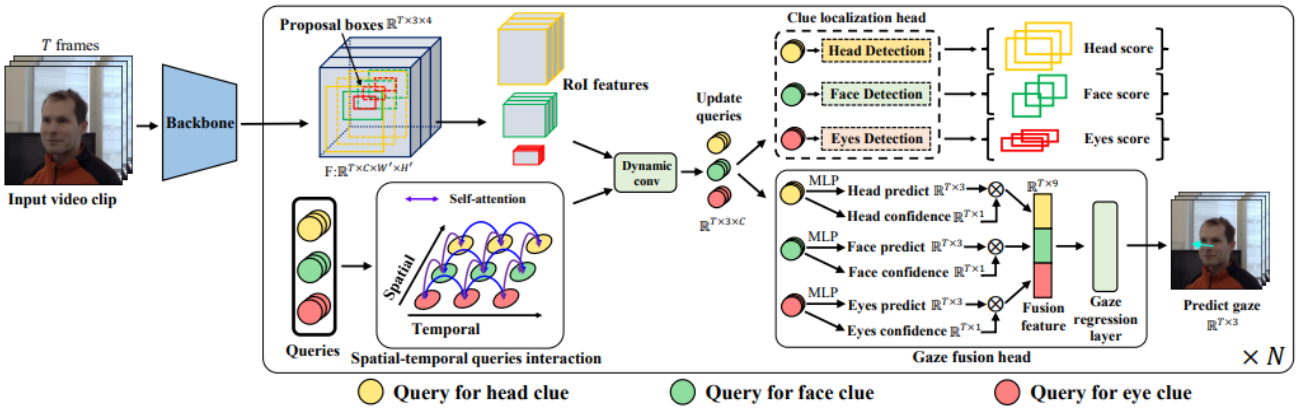


Figure 0.0.12: The proposed MCGaze architecture.

CrossGaze

Σε αντίθεση με προηγούμενες προσεγγίσεις, το CrossGaze [12] δεν απαιτεί εξειδικευμένη αρχιτεκτονική, χρησιμοποιώντας ήδη καθιερωμένα μοντέλα που ενσωματώνονται σε μία αρχιτεκτονική που προσαρμόζεται στο έργο της εκτίμησης 3D βλέμματος. Η είσοδος στο μοντέλο είναι μια εικόνα "in-the-wild", από την οποία εξάγονται οι περιοχές του προσώπου και των ματιών χρησιμοποιώντας ένα μοντέλο ανίχνευσης προσώπου. Στη συνέχεια, η εικόνα του προσώπου τροφοδοτείται σε έναν encoder που αποτυπώνει τα καθολικά (global) χαρακτηριστικά του βλέμματος, ενώ οι εικόνες των ματιών τροφοδοτούνται ως είσοδος στον encoder ματιών, ο οποίος εξάγει τοπικά χαρακτηριστικά. Για να ληφθεί μια πρόβλεψη βλέμματος που λαμβάνει υπόψη και τα μάτια, τα εξαγόμενα χαρακτηριστικά των 3 εικόνων συνδυάζονται μέσω μιας μονάδας cross-attention. Ο προκύπτων συνδυασμός χαρακτηριστικών περνά από επεξεργασία από ένα γραμμικό στρώμα που εξάγει την κατεύθυνση του 3D βλέμματος. Σε σύγκριση με άλλες μελέτες που χρησιμοποιούν πολικές συντεταγμένες ως εξόδους για την εκπαίδευση, σε αυτήν την εργασία διαπιστώθηκε ότι δεν υπάρχει βελτίωση σε σύγκριση με τη χρήση κανονικοποιημένων (x, y, z) τόσο ως εξόδους όσο και ως ετικέτες (labels). Για τη διαδικασία εκπαίδευσης χρησιμοποιήθηκε η απώλεια συνημιτόνου (cosine loss) που αντιπροσωπεύει το συμπλήρωμα της ομοιότητας συνημιτόνου (cosine similarity) και επίσης προσαρμόστηκε ο αλγόριθμος RandAugment [16] ως μέθοδος επαύξησης δεδομένων κατά την διάρκεια της εκπαίδευσης για την αύξηση της ανθεκτικότητας του μοντέλου εκτίμησης βλέμματος.

Για να αποφασιστεί ποιο backbone να χρησιμοποιηθεί, έγιναν πειράματα με διαφορετικά CNN backbones, όπως το EfficientNet [52], το ConvNeXt [49], το Inception ResNet [68] και ένα backbone βασισμένο στην προσοχή (attention-based) (Swin Transformer [48]). Από τα αποτελέσματά τους κατέληξαν στο συμπέρασμα ότι η αρχιτεκτονική Inception ResNet επιτυγχάνει το χαμηλότερο μέσο γωνιακό σφάλμα με τιμή 10.91 για τυχαία αρχικοποίηση και 10.82 για προεκπαιδευμένα βάρη στο ImageNet. Αυτά τα αποτελέσματα δείχνουν την καταλληλότητα της αρχιτεκτονικής Inception ResNet για την επεξεργασία του προσώπου: η ικανότητα να επεξεργάζεται την είσοδο σε πολλαπλές κλίμακες σε κάθε στρώμα βοηθά το μοντέλο να επικεντρώνεται τόσο στην περιοχή των ματιών όσο και στην περι-οφθαλμική περιοχή για την πρόβλεψη του βλέμματος. Επιπλέον, πειραματίστηκαν με διαφορετικά προεκπαιδευμένα βάρη για την αρχιτεκτονική Inception ResNet, δοκιμάζοντάς το στο ImageNet [20], το Casia-WebFace [83], και το VGGFace2 [10]. Το μοντέλο είχε καλύτερη απόδοση στο μεγαλύτερο σύνολο δεδομένων (VGGFace2) από το μοντέλο που είχε προεκπαιδευτεί στο Casia-WebFace λόγω της αυξημένης ποσότητας δεδομένων εκπαίδευσης. Ενώ το Casia-WebFace περιέχει περίπου 500K εικόνες, το VGGFace2 περιέχει 3.3M εικόνες, με αποτέλεσμα ένα 10.02 μέσο γωνιακό σφάλμα σε σύγκριση με το 10.26.

Πειράματα

Σύνολα Δεδομένων στο πρόβλημα της στοχοπροσήλωσης (engagement)

- Pinsoro σύνολο δεδομένων

  Το σύνολο δεδομένων Pinsoro [44] είναι ένα σύνολο δεδομένων ειδικά σχεδιασμένο για να καταγράψει την κοινωνική αλληλεπίδραση των παιδιών κατά την εκτέλεση εργασιών. Η κατανόηση αυτών των κοινωνικών αλληλεπιδράσεων είναι εξαιρετικά σημαντική στον τομέα της Αλληλεπίδρασης Ανθρώπου-Ρομπότ (Human-Robot Interaction, HRI), ειδικά σε περιοχές όπου λαμβάνουν χώρα μακροχρόνιες αλληλεπιδράσεις, όπως η υγειονομική περίθαλψη και η εκπαίδευση. Σε αυτούς τους τομείς, είναι σημαντικό το ρομπότ να μπορεί να αναγνωρίσει και να ερμηνεύσει τις κοινωνικές συμπεριφορές των παιδιών μέσω οπτικών ή ακουστικών ενδείξεων. Για τα πειράματα στη μελέτη μας χρησιμοποιούμε βίντεο από αυτό το σύνολο δεδομένων για να βρούμε έναν ακριβή και αξιόπιστο τρόπο αναγνώρισης της στοχοπροσήλωσης των παιδιών κατά την εκτέλεση μικρών εργασιών γύρω από ένα μεγάλο διαδραστικό τραπέζι. Συγκεκριμένα, επιλέγουμε 25 βίντεο παιδιών που αλληλεπιδρούν με ρομπότ από την άλλη πλευρά του τραπεζιού, καθώς το σύνολο δεδομένων Pinsoro περιλαμβάνει επίσης και αλληλεπιδράσεις παιδί-παιδί. Οι αλληλεπιδράσεις στα βίντεο έχουν επισημανθεί σε τρεις διαφορετικούς άξονες: στοχοπροσήλωσης στην εργασία, κοινωνική στοχοπροσήλωσης και κοινωνική στάση. Στη μελέτη αυτή πειραματιζόμαστε μόνο στον άξονα της στοχοπροσήλωσης στην εργασία, ο οποίος είναι χωρισμένος σε τέσσερα επίπεδα στοχοπροσήλωσης: μη-παιχνίδι, αναζήτηση ενήλικα, άσκοπη αλληλεπίδραση και στοχευμένη εργασία. Λόγω του γεγονότος ότι το επίπεδο της στοχοπροσήλωσης για αναζήτηση ενήλικα είναι εξαιρετικά σπάνιο, το αποκλείουμε από τη μελέτη μας.

- ASD Games σύνολο δεδομένων Το σύνολο δεδομένων ASD Games [4] είναι ένα σύνολο δεδομένων που συλλέχθηκε κατά τη διάρκεια του έργου BabyRobot [21], και αποτελείται από βίντεο στα οποία τα παιδιά που συμμετέχουν αντιμετωπίζουν διαταραχή του αυτιστικού φάσματος (μέση ηλικία 10.6 ετών). Σε αυτά τα βίντεο, κάθε παιδί συμμετέχει σε τέσσερα διαφορετικά παιχνίδια με δύο ρομπότ: Δείξε μου την Κίνηση, Εκφράσου, Παντομίμα και Μάντεψε το Αντικείμενο. Κάθε παιδί στέκεται μπροστά στα ρομπότ και αλληλεπιδρά ελεύθερα μαζί τους, ενώ κινείται στο δωμάτιο όπως επιθυμεί, και ταξινομείται σε τρία επίπεδα στοχοπροσήλωσης (engagement): αμέτοχος, παθητική προσοχή (ή ενέργεια χωρίς προσοχή) και engaged.

Σύνολα δεδομένων που χρησιμοποιούνται στην παρακολούθηση βλέμματος

- WiDo σύνολο δεδομένων Για τη μελέτη αυτή αναφερόμαστε σε αυτό το σύνολο δεδομένων ως WiDo [59, 58], καθώς είναι ένα πλούσια εμπλουτισμένο σύνολο δεδομένων, που αποτελείται από 17 βίντεο παιδιών με σύνδρομο Williams και Down (WiDo), αλλά και κανονικά αναπτυσσόμενα παιδιά, που αλληλεπιδρούν με τις μητέρες τους. Επιλέγουμε αυτό το σύνολο δεδομένων παρόλο που περιέχει παιδιά με σύνδρομα Williams και Down, ενώ έχουμε αναφερθεί εκτενώς σε παιδιά με αυτισμό, καθώς αυτές οι διαταραχές οδηγούν σε παρόμοιες συμπεριφορές στις οπτικές τους αποκρίσεις με εκείνες των παιδιών με αυτισμό, κατά τις αλληλεπιδράσεις τους. Αυτό το είδος δεδομένων είναι πολύ σημαντικό για την ανάπτυξη των αλγορίθμων μας. Το σύνολο δεδομένων WiDo περιέχει πολλές πληροφορίες για διάφορες εργασίες όρασης υπολογιστών, όπως αναγνώριση συναισθημάτων, παρακολούθηση βλέμματος, αναγνώριση ομιλητή, τύπο παιχνιδιού, σκοπό ενέργειας, μεταξύ άλλων. Σε κάθε βίντεο υπάρχει ένα παιδί με τη μητέρα του που αλληλεπιδρούν ελεύθερα μεταξύ τους και με διάφορα αντικείμενα και παιχνίδια στο σπίτι τους, έναν χώρο όπου αισθάνονται πολύ εξοικειωμένοι, καθιστώντας δύσκολη την ανάπτυξη αλγορίθμων όρασης υπολογιστών που επεξεργάζονται και μοντελοποιούν τις αλληλεπιδράσεις τους, καθώς υπάρχουν πολλές περιπτώσεις όπου τα πρόσωπα και το σώμα τους είναι μερικώς ή πλήρως κρυμμένα. Στη μελέτη μας, θα εστιάσουμε στην παρακολούθηση βλέμματος, για την οποία το σύνολο δεδομένων περιέχει 6 διαφορετικές κατηγορίες: (i) Δεν κοιτάζει πουθενά συγκεκριμένα ή την κάμερα, (ii) Κοιτάζει ένα αντικείμενο που χρησιμοποιεί μόνο το παιδί αλλά όχι η μητέρα, (iii) Κοιτάζει ένα αντικείμενο που χρησιμοποιείται στην αλληλεπίδραση με τη μητέρα, (iv) Κοιτάζει κάποιο μέρος του σώματος της μητέρας, (v) Κοιτάζει το πρόσωπο της μητέρας αλλά όχι τα μάτια και (vi) Κοιτάζει τη μητέρα στα μάτια.

Προτεινόμενες Μέθοδοι

Εκτίμηση της Στοχοπροσήλωσης

Αρχιτεκτονική TSN

Η αρχιτεκτονική των Temporal Segment Networks (TSN) αλλάζει τον τρόπο με τον οποίο τα δείγματα βίντεο επεξεργάζονται από τα μοντέλα. Αντί να παρέχεται ως είσοδος στο μοντέλο μια ακολουθία συνεχόμενων καρέ από ένα δείγμα βίντεο και το μοντέλο να προβλέπει την κατηγορία του, στην αρχιτεκτονική TSN τα δίκτυα λειτουργούν σε ακολουθίες σύντομων αποσπασμάτων που δειγματοληπτούνται αραιά από ολόκληρο το βίντεο. Στη συνέχεια, τα αποτελέσματα για κάθε απόσπασμα παράγονται από τα backbone μοντέλα και στο τέλος συνδυάζονται όλα μαζί για να προκύψει η τελική κοινή εκτίμηση της κατηγορίας.

Αρχιτεκτονική Μοντέλου χρησιμοποιώντας την TSN μέθοδο

Η προτεινόμενη αρχιτεκτονική μοντέλου που χρησιμοποιεί το πλαίσιο του Temporal Segment Network παρουσιάζεται στο Σχήμα 0.0.13. Ο αριθμός των τμημάτων επηρεάζει τον αριθμό των συνεχόμενων ομάδων καρέ που θα επιλεγούν από κάθε δείγμα βίντεο διάρκειας 10 δευτερολέπτων, ενώ τα καρέ ανά τμήμα επηρεάζουν τη διάρκεια, σε καρέ, που θα έχει κάθε μία από αυτές τις ομάδες. Μπορούμε να δούμε ότι σύμφωνα με την αρχιτεκτονική TSN, ο dataloader που δημιουργήσαμε θα επιλέξει τον καθορισμένο αριθμό τμημάτων καθ' όλη τη διάρκεια του δείγματος βίντεο, το καθένα αποτελούμενο από τον καθορισμένο αριθμό καρέ, και θα τροφοδοτήσει αυτά τα καρέ ως είσοδο στο backbone μοντέλο. Από εκεί, το backbone θα παράγει τα χαρακτηριστικά εξόδου (features) για κάθε καρέ, τα οποία θα κατευθυνθούν στο TSN head που θα παράγει την τελική έξοδο ταξινόμησης μέσω ενός μηχανισμού συναίνεσης που εφαρμόζεται στα χαρακτηριστικά των καρέ.

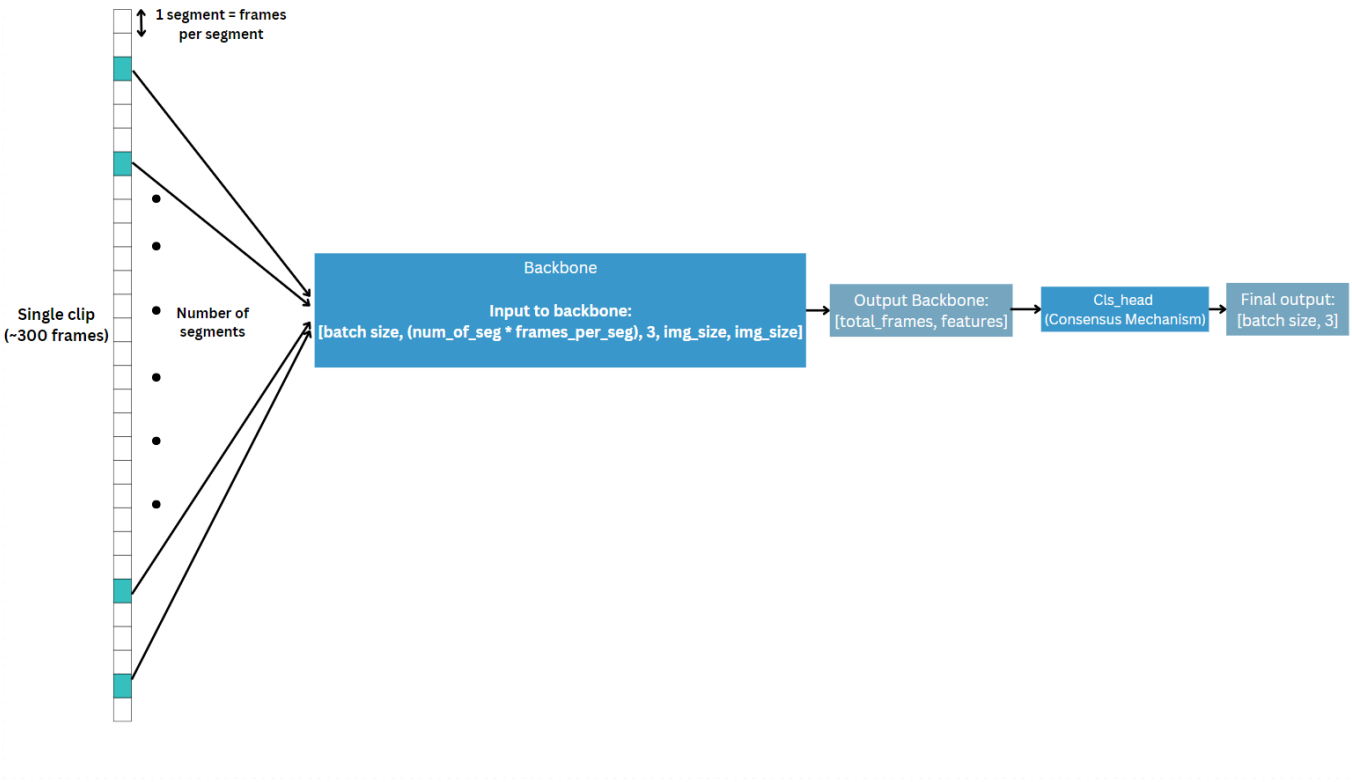


Figure 0.0.13: Προτεινόμενη αρχιτεκτονική μοντέλου για το πρόβλημα της εκτίμησης της στοχοπροσήλωσης, χρησιμοποιώντας την μέθοδο TSN.

Εκτίμηση του βλέμματος

Όπως είδαμε σε πολλές μελέτες που προσεγγίζουν το πρόβλημα της ανίχνευσης βλέμματος, προκειμένου τα δίκτυα να αντιμετωπίσουν καλύτερα το πρόβλημα και να παράγουν πιο στοχευμένες εκτιμήσεις, πρέπει να τροφοδοτούνται με συνεχόμενες περικοπές προσώπου των υποκειμένων ως εισόδους, και όχι με ολόκληρα καρέ [39, 26, 12]. Έτσι, οι είσοδοι στο μοντέλο μας θα είναι κάθε φορά 7 συνεχόμενες περικοπές προσώπου του παιδιού σε κάθε βίντεο, όπως φαίνεται στο Σχήμα 0.0.14. Το μοντέλο μας θα βασίζεται σε αυτό που προτείνεται στο [39], το οποίο λαμβάνει κάθε καρέ εισόδου, το οποίο το προεπεξεργάζεται πρώτα ένα backbone Resnet. Στη συνέχεια, τα εξαγόμενα χαρακτηριστικά τροφοδοτούνται σε ένα μοντέλο LSTM το οποίο τα επεξεργάζεται στη χρονική διάσταση και παράγει τα τελικά χαρακτηριστικά για όλα τα καρέ εισόδου, από τα οποία κρατάμε μόνο αυτά που αντιστοιχούν στο μεσαίο καρέ (το 4ο). Στη συνέχεια, αυτά τροφοδοτούνται μέσω του τελικού πλήρως συνδεδεμένου επιπέδου και παράγουν τους διανυσματικούς προσανατολισμούς βλέμματος. Χρησιμοποιούμε αυτά τα διανύσματα προσανατολισμού βλέμματος για να ελέγξουμε αν η κατεύθυνση του βλέμματος του παιδιού διασταυρώνεται με κάποια από τα κουτιά ενδιαφέροντος (bounding boxes) στο μεταλλαγμένο πρόβλημά μας. Αυτό το επιτυγχάνουμε δημιουργώντας μια γραμμή με την κλίση της να προκύπτει από τις συντεταγμένες dx και dy και την αρχή (0, 0) να είναι το σημείο εκκίνησης του βλέμματος του παιδιού. Αν διασταυρώνεται με κάποιο από τα κουτιά, τότε η πρόβλεψη αποδίδεται στην αντίστοιχη κατηγορία κουτιού, αλλιώς αποδίδεται στην κατηγορία "αλλού".

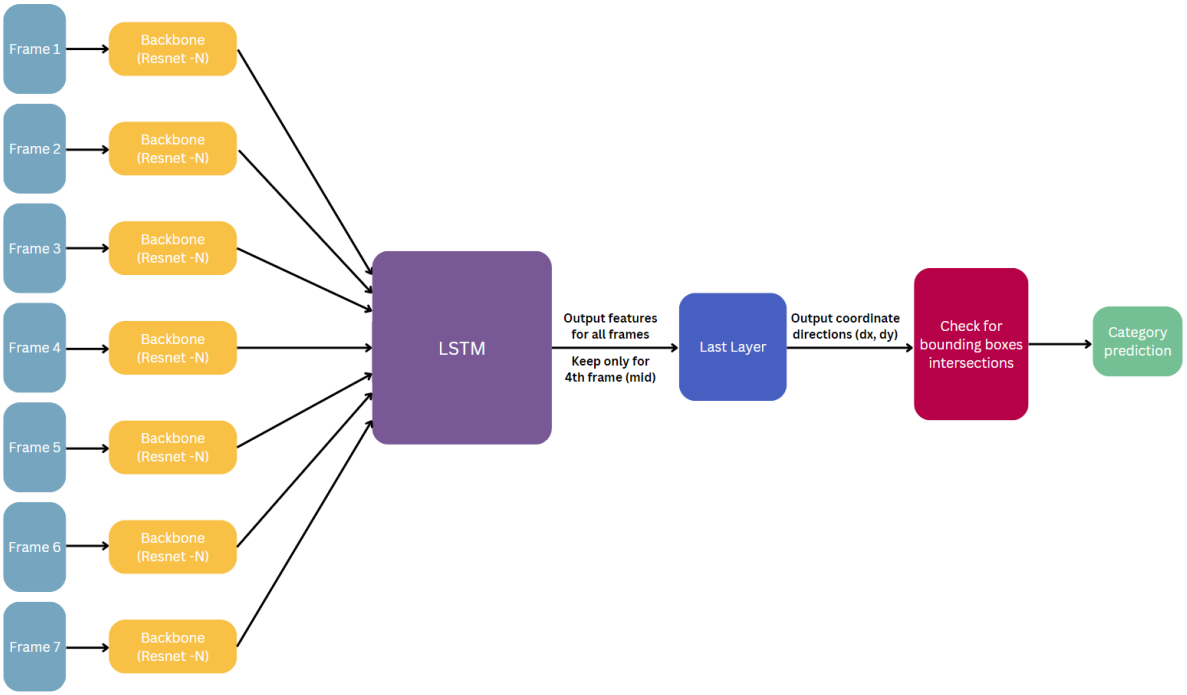


Figure 0.0.14: Εφαρμοσμένη αρχιτεκτονική για εκτίμηση βλέμματος όπου χρησιμοποιείται το προεκπαιδευμένο μοντέλο από [39] προσθέτοντας έναν έλεγχο κατηγορίας στο τέλος.

Ενσωμάτωση δεδομένων ανίχνευσης βλέμματος στην Εκτίμηση Εμπλοκής

Έχοντας μελετήσει και αντιμετωπίσει ξεχωριστά τα προβλήματα της αναγνώρισης της στοχοπροσήλωσης και της ανίχνευσης βλέμματος, αναρωτιόμαστε αν θα μπορούσε να επηρεαστεί θετικά η απόδοση στο πρόβλημα της στοχοπροσήλωσης όταν ενσωματώσουμε την υλοποίηση της ανίχνευσης βλέμματος. Για το λόγο αυτό, δημιουργούμε μια νέα αρχιτεκτονική δικτύου που συνδυάζει τα δύο μοντέλα που χρησιμοποιήθηκαν στις φάσεις αναγνώρισης εμπλοκής και ανίχνευσης βλέμματος, χρησιμοποιώντας τη μέθοδο TSN.

Το προτεινόμενο συνδυασμένο μοντέλο στοχοπροσήλωσης και βλέμματος φαίνεται στο Σχήμα 0.0.15. Όπως βλέπουμε, παρόμοια και με το πρόβλημα της στοχοπροσήλωσης, πρώτα σύμφωνα με την υλοποίηση TSN, ο νέος dataloader που δημιουργούμε επιλέγει έναν συγκεκριμένο αριθμό τμημάτων καθ' όλη τη διάρκεια του

κλιπ (δείγματος) και για κάθε τμήμα ένα ποσό συνεχόμενων καρέ, με τα οποία τροφοδοτεί τα επίπεδα εισόδου του Backbone. Στη συγκεκριμένη αρχιτεκτονική όμως, το μοντέλο βλέμματος θα τροφοδοτείται επίσης με τις περικοπές προσώπου των πρώτων 7 καρέ κάθε τμήματος (segment). Στο πείραμά μας χρησιμοποιούμε 4 τμήματα, έτσι το μοντέλο βλέμματος τώρα θα τροφοδοτείται με 4 εισόδους, παράγοντας 4 σύνολα χαρακτηριστικών (features) για την μεσαία εικόνα από το τελευταίο LSTM επίπεδό του. Κάθε σύνολο χαρακτηριστικών αποτελείται από 512 χαρακτηριστικά, που είναι αυτά που παράγονται για το μεσαίο καρέ κάθε ακολουθίας, έτσι συνολικά για τα 4 σύνολα παράγουμε 2048 χαρακτηριστικά. Αυτά τα σύνολα χαρακτηριστικών στη συνέχεια τροφοδοτούνται στο backbone, μαζί με τις εικόνες εισόδου. Στη συνέχεια, συγχωνεύονται με τα επίπεδα χαρακτηριστικών που παράγονται μετά το τελευταίο συνελικτικό επίπεδο του backbone και τροφοδοτούνται στα ακόλουθα τρία επίπεδα fc όλα μαζί. Στο τέλος, το μοντέλο θα παράγει την έξοδο του με βάση τα συγχωνευμένα χαρακτηριστικά, τα οποία θα τροφοδοτηθούν στο cls head και η ταξινόμηση θα πραγματοποιηθεί.

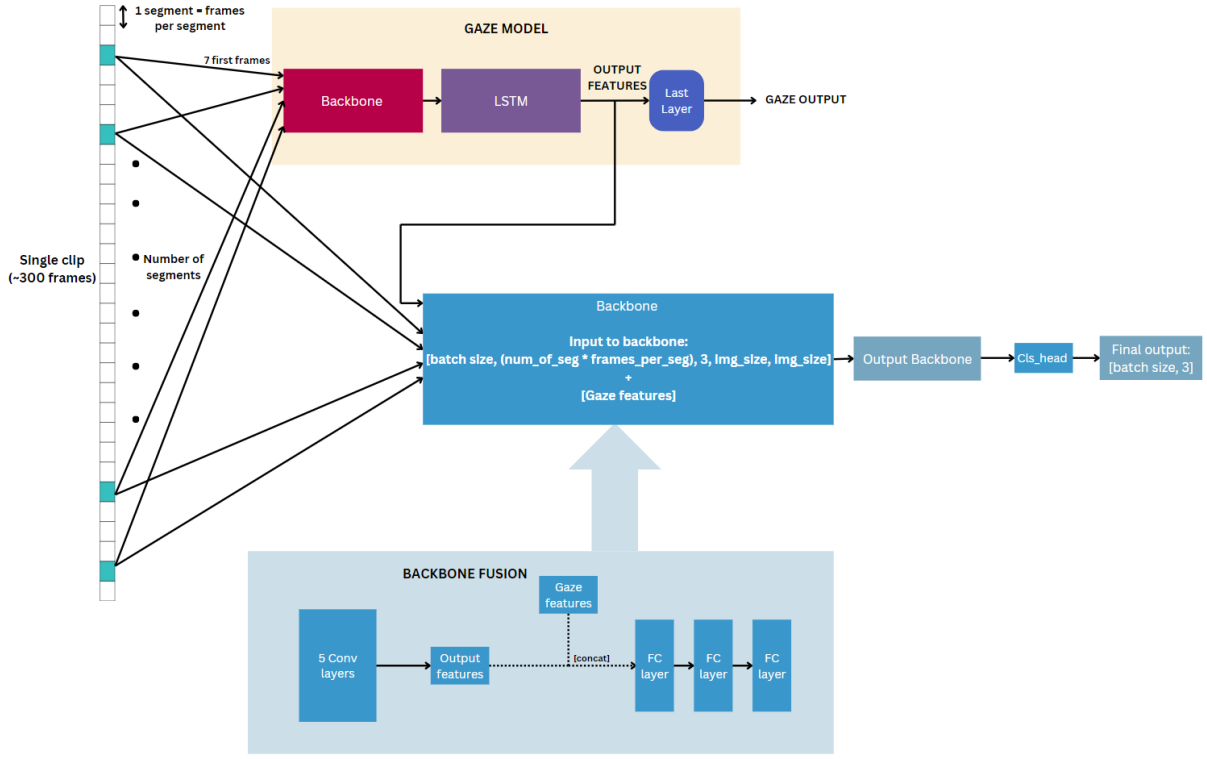


Figure 0.0.15: Η αρχιτεκτονική του δικτύου EngagementPlusGaze, με ενσωμάτωση χαρακτηριστικών παραγώμενων από το μοντέλο εκτίμησης βλέμματος [39] με τα χαρακτηριστικά παραγόμενα από το τελευταίο συνελικτικό στρώμα του backbone του μοντέλου της εκτίμησης της στοχοπροσήλωσης.

Διάφορες υλοποιήσεις συγχώνευσης των χαρακτηριστικών δοκιμάστηκαν, συμπεριλαμβανομένης της συγχώνευσης των χαρακτηριστικών με τις εξόδους που παράγονται μετά το backbone, πριν τροφοδοτηθούν στο cls head. Τα καλύτερα αποτελέσματα όμως επιτεύχθηκαν με την παραπάνω υλοποίηση.

Πειραματικά Αποτελέσματα στην Εκτίμηση Στοχοπροσήλωσης

Προεπεξεργασία

Για να μπορέσουμε να εκπαιδεύσουμε τα επιλεγμένα μοντέλα με την μέθοδο TSN, χρησιμοποιώντας την αρχιτεκτονική που παρουσιάζεται στο σχήμα 0.0.13, πρέπει πρώτα να προεπεξεργαστούμε τα βίντεο μας. Σε αυτού του είδους τα σύνολα δεδομένων, η διαδικασία προεπεξεργασίας είναι αρκετά δύσκολη, καθώς τα σύνολα δεδομένων αποτελούνται από βίντεο και αρχεία επισημειώσεων για κάθε παιδί. Σε κάθε αρχείο επισημειώσεων, το οποίο είναι σε μορφή csv στο Pinsoro σύνολο δεδομένων, οι ετικέτες δίνονται για κάθε καρέ σε κάθε γραμμή, και

δημιουργούμε κώδικα που διαβάζει και δημιουργεί τα συνεχόμενα τμήματα για κάθε κατηγορία (π.χ. από το 803 έως το 2300 οι ετικέτες είναι goaloriented). Στη συνέχεια, λαμβάνουμε κάθε συνεχόμενη ομάδα καρέ που ανήκουν σε μία μόνο κατηγορία και την χωρίζουμε σε διαστήματα των 10 δευτερολέπτων (περίπου 300 καρέ). Το διάστημα των 10 δευτερολέπτων επιλέχθηκε καθώς έχει βρεθεί ότι είναι μια μέση περίοδος κατά την οποία το επίπεδο στοχοπροσήλωσης του παιδιού μπορεί να ανιχνευθεί.

Έπειτα, κάθε κλιπ διάρκειας 10 δευτερολέπτων θεωρείται ως ξεχωριστό δείγμα, και δημιουργούμε ένα σύνολο δεδομένων σε νέα μορφή που θα είναι πιο εύχρηστη για την εκπαιδευτική διαδικασία των μοντέλων και του dataloader. Κάθε κατηγορία έχει έναν φάκελο ο οποίος περιέχει μέσα του ένα φάκελο για κάθε κλιπ διάρκειας 10 δευτερολέπτων που ανήκει σε αυτόν, και κάθε φάκελος κλιπ περιέχει όλα τα καρέ του.

Κατά την εκπαίδευση, ο καθορισμένος αριθμός συνολικών καρέ, που εξαρτάται από τις τιμές που επιλέγονται για τον αριθμό των τμημάτων (segments) και τα καρέ ανά τμήμα (frames per segment) (συνολικά καρέ = αριθμός τμημάτων * καρέ ανά τμήμα), θα επιλέγεται από κάθε έναν από τους φακέλους (συνήθως όχι περισσότερα από 40 καρέ συνολικά) και θα χρησιμοποιείται ως είσοδος στο μοντέλο για το συγκεκριμένο δείγμα.

Επιπλέον, πραγματοποιήσαμε επαύξηση δεδομένων στις λιγότερο πληθυσμιακά κατηγορίες "aimless" και "no play", καθώς η διαφορά δειγμάτων σε σύγκριση με την "goaloriented" ήταν αρκετά μεγάλη. Έτσι, δημιουργήθηκαν νέα κλιπ διάρκειας 10 δευτερολέπτων για κάθε ένα από τα υπάρχοντα και προστέθηκαν στους αντίστοιχους δύο φακέλους κατηγοριών, αντιπροσωπεύοντας τα νέα δείγματα για την εκπαιδευτική διαδικασία. Οι τεχνικές επαύξησης που χρησιμοποιήθηκαν ήταν HorizontalFlip, GaussianBlur, ElasticTransformation, και Salt and Pepper.

Πειράματα με Backbone Resnet-50 - Αρχιτεκτονική TSN

Ξεκινάμε τα πειράματά μας χρησιμοποιώντας το Resnet-50 ως backbone για την εξαγωγή χαρακτηριστικών. Για την υλοποίηση του κώδικά μας, χρησιμοποιούμε τη βιβλιοθήκη mmaction και τα πακέτα της. Για το backbone Resnet-50, χρησιμοποιούμε προεκπαιδευμένα βάρη που προέρχονται από το torchvision. Δοκιμάζουμε αυτή την υλοποίηση χρησιμοποιώντας διάφορους συνδυασμούς των παραμέτρων TSN και της εκπαιδευτικής διαδικασίας, και τα αποτελέσματα που παράγονται φαίνονται στον Πίνακα 1.

Resnet-50									
Number of segs	Frames per seg	Image size	Optimizer	Learning rate	Gamma	Step size	Accuracy	w. F1	w. Precision
4	7	224	Adam	0.01	0.2	8	49.48	48.71	-
4	7	224	Adam	0.005	0.2	8	47.06	45.11	48.14
4	7	224	SGD	0.01	0.2	8	52.20	50.88	52.8
3	10	224	SGD	0.01	0.2	8	52.62	51.98	52.51
4	10	190	SGD	0.01	0.2	8	49.48	46.59	49.69
4	10	224	SGD	0.01	0.2	8	46.12	45.83	46.13
5	8	224	SGD	0.01	0.2	8	51.57	50.75	52.24
4	10	224	SGD	0.01	0.22	6	53.46	51.18	53.54

Table 1: Επίδοση του Resnet-50 στο Pinsoro σύνολο δεδομένων [44] χρησιμοποιώντας την αρχιτεκτονική TSN, με διαφορετικές τιμές παραμέτρων.

Τα αποτελέσματα που παρήγαγε το backbone Resnet-50 χρησιμοποιώντας την αρχιτεκτονική TSN ήταν ελπιδοφόρα, αλλά όχι τα καλύτερα. Ένας βασικός λόγος για αυτό είναι ότι το μοντέλο δυσκολεύτηκε να προβλέψει τις κατηγορίες μειοψηφίας. Το υψηλότερο F1-score που επιτεύχθηκε ήταν 51.98%, με ακρίβεια 52.62% και ακρίβεια 52.51%. Για αυτό μπορεί να ευθύνεται το γεγονός ότι το Resnet-50 δεν είναι αρχιτεκτονική μοντέλου που επεξεργάζεται τα συνεχόμενα καρέ χρονικά, επομένως μπορεί να μην καταγράφει καλά τη χρονική δυναμική των διαφόρων κατηγοριών.

Ο πίνακας σύγχυσης (confusion matrix) του καλύτερου αποτελέσματος για αυτήν την αρχιτεκτονική παρουσιάζεται στο Σχήμα 0.0.16. Όπως μπορούμε να δούμε, το μοντέλο είναι σε θέση να προβλέψει το 46.6% των συνολικών τμημάτων "no play" και το 35.8% των τμημάτων "aimless", σε σύγκριση με το 69.6% που επιτεύχθηκε στα τμήματα "goaloriented" (κατηγορία πλειοψηφίας).

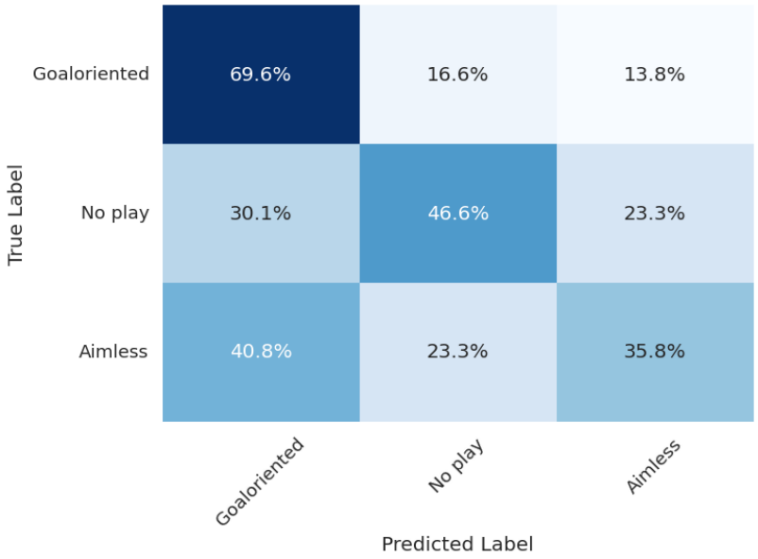


Figure 0.0.16: Ο πίνακας σύγχησης για τα καλύτερα αποτελέσματα του ResNet-50 στην εκτίμηση της στοχοπροσήλωσης στο σύνολο δεδομένων Pinsoro. [44]

Πειράματα με Backbone C3D - TSN Αρχιτεκτονική

Για να προσπαθήσουμε να παράγουμε καλύτερα αποτελέσματα, αποφασίσαμε να χρησιμοποιήσουμε ένα backbone δίκτυο που αποτελείται από 3D συνελικτικές μονάδες, το μοντέλο C3D [71]. Το C3D μπορεί να μοντελοποιήσει ταυτόχρονα τοπικές και χρονικές πληροφορίες, γεγονός που το βοηθά συνήθως να υπερέχει σε σχέση με άλλες αρχιτεκτονικές 2D ConvNet σε διάφορα καθήκοντα ανάλυσης βίντεο. Για τα πειράματά μας, ξεκινήσαμε τροποποιώντας τις ίδιες υπερπαραμέτρους που χρησιμοποιήθηκαν για το Resnet-50, και αφού βρήκαμε έναν συνδυασμό που παρήγαγε τα καλύτερα αποτελέσματα, τις κρατήσαμε σταθερές και προσπαθήσαμε να δούμε αν τα αποτελέσματα θα βελτιώνονταν αλλάζοντας λίγο την αρχιτεκτονική του μοντέλου. Τα αποτελέσματα που εμφανίζονται στον Πίνακα 2 παράγονται χρησιμοποιώντας μέγεθος εικόνας 112 (το αρχικό μέγεθος εισόδου του μοντέλου), τον βελτιστοποιητή SGD, αρχικό ρυθμό μάθησης 0.01 με gamma 0.22 και βήμα μεγέθους 6.

Το backbone C3D, αρχικά τροφοδοτείται με 16 καρέ συνολικά, το οποίο δοκιμάσαμε αρχικά χρησιμοποιώντας την αρχιτεκτονική δειγματοληψίας TSN, επιλέγοντας 2 τμήματα των 8 καρέ από κάθε κλιπ (δείγμα). Μετά από αυτή την αρχική εκτέλεση, αποφασίσαμε να τροποποιήσουμε λίγο το μοντέλο ώστε να μπορεί να τροφοδοτηθεί με μεγαλύτερες εισόδους, και αντί για 16 καρέ συνολικά, χρησιμοποιήσαμε 32, χωρίζοντάς τα σε 4 τμήματα των 8 καρέ το καθένα. Στην πρώτη υλοποίηση δεν αλλάξαμε τον αριθμό των συνελικτικών μπλοκ ή των πλήρως συνδεδεμένων επιπέδων, αλλά απλά αλλάξαμε την είσοδο του πρώτου πλήρως συνδεδεμένου επιπέδου από 8192 σε 16384, ωστόσο αυτό επιδείνωσε τα αποτελέσματα. Στη δεύτερη υλοποίηση αλλάξαμε ακόμα περισσότερο τις εισόδους και τις εξόδους των δύο πλήρως συνδεδεμένων επιπέδων από 8192 –> 4096 –> 4096 σε 16384 –> 8192 –> 4096.

Στην τελική μας υλοποίηση, που φαίνεται στο Σχήμα 0.0.17, αποφασίσαμε να προσθέσουμε ένα ακόμα πλήρως συνδεδεμένο επίπεδο, προκειμένου να επεξεργαστούμε περαιτέρω τα χαρακτηριστικά. Έτσι, οι νέες είσοδοι και έξοδοι των τριών πλήρως συνδεδεμένων επιπέδων τώρα είναι 16384 –> 8192 –> 4096 –> 4096.

Όπως μπορούμε να δούμε από τα αποτελέσματα στον Πίνακα 2, οι αρχιτεκτονικές του δικτύου C3D υπερέχουν σημαντικά σε σχέση με τα καλύτερα αποτελέσματα του Resnet-50, αναδεικνύοντας τη δύναμη των 3D συνελικτικών νευρωνικών δικτύων. Επιπλέον, παρατηρούμε ότι παρόλο που οι τιμές των μετρικών μειώθηκαν αρχικά όταν αυξήσαμε τον αριθμό των εισερχόμενων καρέ, στο τέλος καταφέραμε να τις βελτιώσουμε τροποποιώντας τα πλήρως συνδεδεμένα επίπεδα. Τα καλύτερα αποτελέσματα προέκυψαν από την τελική τροποποιημένη αρχιτεκτονική του C3D, η οποία λαμβάνει συνολικά 32 εισερχόμενα καρέ, χωρισμένα σε 4 τμήματα, παράγοντας έξοδο υψηλότερης διάστασης μετά το τελευταίο συνελικτικό επίπεδο, αλλά στη συνέχεια χρησιμοποιεί 3 πλήρως συνδεδεμένα επίπεδα για να τη μειώσει σταδιακά στα τελικά 4096 χαρακτηριστικά. Όλα αυτά οδηγούν σε ένα μοντέλο με μεγαλύτερο αριθμό παραμέτρων που μπορούν να εκπαιδευτούν, βοηθώντας το να μάθει το έργο σε

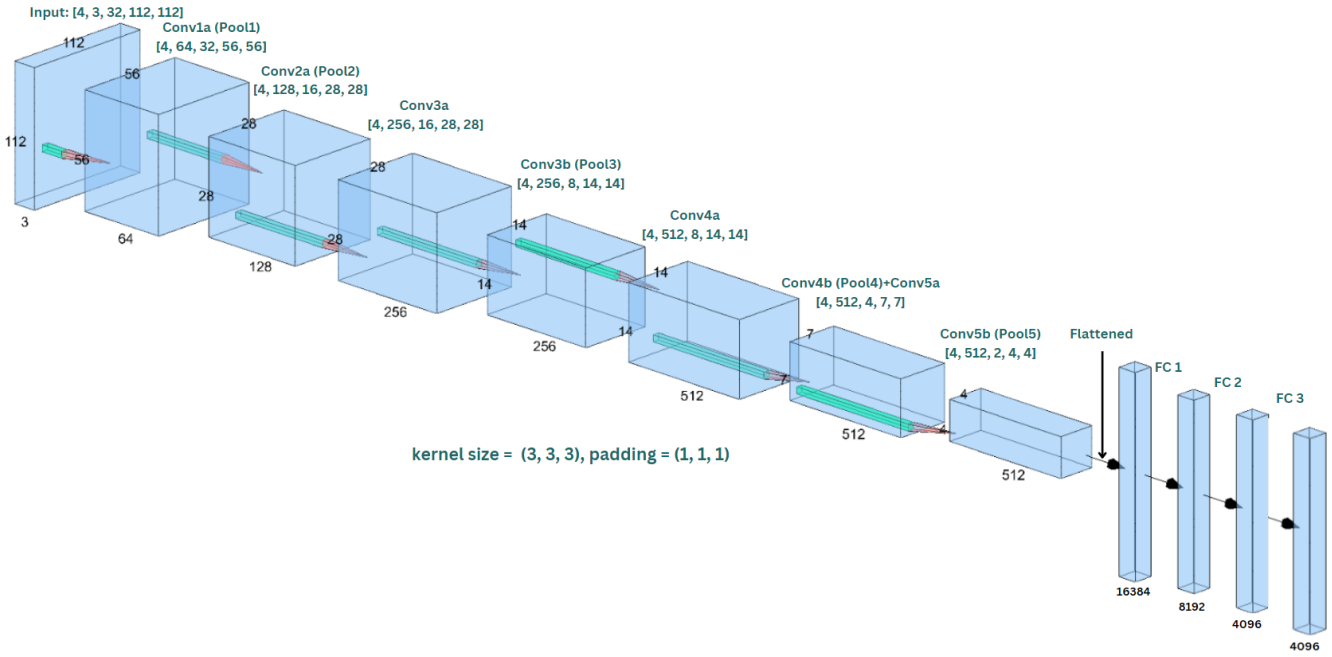


Figure 0.0.17: Αρχιτεκτονική μοντέλου C3D με 32 καρέ ως είσοδο και 3 fully connected επίπεδα.

C3D						
Number of segs	Frames per seg	FC layers	Feature Dimensions	Accuracy	w. F1	w. Precision
2	8	2	8192, 4096, 4096	61.85	61.24	62.72
4	8	2	16384, 4096, 4096	59.75	57.2	64.4
4	8	2	16384, 8192, 4096	64.15	62.81	65.37
4	8	3	16384, 8192, 4096, 4096	68.52	65.28	72.82
Resnet-50						
3	10			52.62	51.98	52.51

Table 2: Επίδοση διαφορετικών τροποποιήσεων του μοντέλου C3D χρησιμοποιώντας την TSN αρχιτεκτονική στο Pinsoro σύνολο δεδομένων

υψηλότερο επίπεδο.

Στο Σχήμα 0.0.19 μπορούμε να δούμε τον πίνακα σύγχυσης για τα καλύτερα αποτελέσματα που παρήχθησαν χρησιμοποιώντας την τελική τροποποιημένη αρχιτεκτονική C3D. Όπως βλέπουμε, το μοντέλο πέτυχε εξαιρετική απόδοση στην αναγνώριση της πλειοψηφικής κατηγορίας (στοχευμένο παιχνίδι) και ταυτόχρονα υπερδιπλασίασε την ικανότητά του να αναγνωρίζει την κατηγορία "aimless" σε σύγκριση με τα αποτελέσματα που χρησιμοποιούσαν το μοντέλο Resnet-50. Παρατηρούμε επίσης μια μικρή πτώση στην εκτίμηση της κατηγορίας "no play", που είναι αποτέλεσμα της σημαντικής αύξησης στις άλλες δύο κατηγορίες.

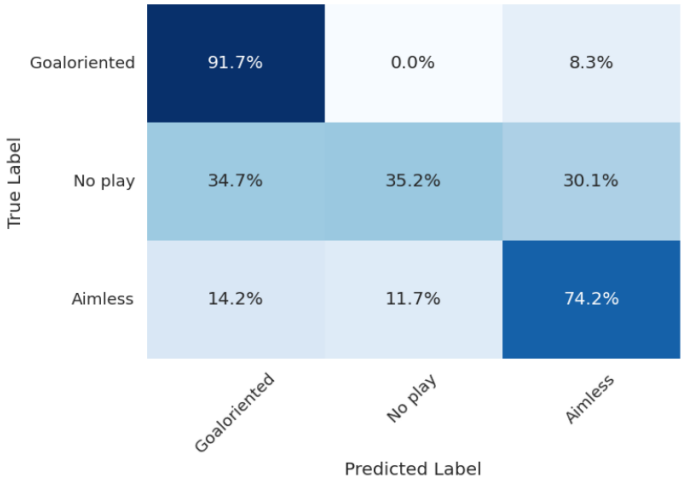


Figure 0.0.18: Πίνακας σύγχησης για τα αποτελέσματα του C3D στην εκτίμηση της στοχοπροσήλωσης στο σύνολο δεδομένων Pinsoro, χρημοποιώντας συνολικά 32 καρέ ως είσοδο και 3 fully connected επίπεδα.

Σύγκριση Αποτελεσμάτων

Για να αξιολογήσουμε καλύτερα την απόδοση των αρχιτεκτονικών μας, τις συγκρίνουμε με τα αποτελέσματα που παρήχθησαν στις μελέτες [3] και [7], δύο μελέτες που αντιμετώπισαν το πρόβλημα της εκτίμησης της στοχοπροσήλωσης στο σύνολο δεδομένων Pinsoro.

Model Comparison			
Model	Accuracy	w. F1	w. Precision
majority class	38	20.55	14.85
ResNet-50	40.52	34.32	35.05
R(2+1)D RGB + Depth [3]	62.49	59.87	59.36
R(2+1)D RGB + Flow (Late) [3]	66.78	65.30	65.32
R(2+1)D RGB + Flow (Early) [3]	68.40	**67.11**	68.50
Resnet50-TSN (ours)	52.62	51.98	52.51
C3DI3D-TSN-original (ours)	61.85	61.24	62.72
C3DI3D-TSN (ours)	**68.52**	65.28	**72.82**

Table 3: Σύγκριση επίδοσης διαφορετικών αρχιτεκτονικών στο πρόβλημα της εκτίμησης της στοχοπροσήλωσης στο Pinsoro σύνολο δεδομένων.

Από τη σύγκριση απόδοσης στον Πίνακα 3, βλέπουμε ότι μόνο η αρχιτεκτονική R(2+1)D με πρώιμη σύντηξη των δεδομένων RGB και οπτικής ροής υπερτερεί του πλαισίου C3D-TSN μας. Ωστόσο, είναι σημαντικό να αναφέρουμε ότι η υλοποίησή μας δεν απαιτεί τόσο ακατέργαστα δεδομένα όσο και δεδομένα οπτικής ροής, πράγμα που σημαίνει ότι μπορεί να εξοικονομήσει χρόνο τόσο στη φάση προεπεξεργασίας όσο και κατά την εκπαίδευση. Επιπλέον, είναι σημαντικό να σημειώσουμε τη σημαντική βελτίωση στην απόδοση του μοντέλου Resnet-50 όταν χρησιμοποιείται με την αρχιτεκτονική TSN σε σύγκριση με το αυτόνομο μοντέλο, καθώς η σταθμισμένη βαθμολογία F1 αυξήθηκε από 34.32% σε 51.98%. Αυτά τα αποτελέσματα δείχνουν αρχικά πώς η αρχιτεκτονική TSN μπορεί να βελτιώσει αυτόνομες αρχιτεκτονικές μοντέλων και επιπλέον πώς το μοντέλο C3D με την αρχιτεκτονική TSN μπορεί να γεφυρώσει το κενό μεταξύ της χρήσης μόνο ακατέργαστων καρέ RGB και της χρήσης τόσο καρέ RGB όσο και καρέ οπτικής ροής.

Πειραματικά Αποτελέσματα στην Εκτίμηση Βλέμματος

Εκπαίδευση του πλαισίου εκτίμησης βλέμματος στο Σύνολο Δεδομένων WiDo

Προεπεξεργασία

Όπως και τα σύνολα δεδομένων που εστιάζουν στο έργο της στοχοπροσήλωσης, το σύνολο δεδομένων WiDo είναι ένα ιδιαίτερα απαιτητικό σύνολο για επεξεργασία και αποτελεσματική εκπαίδευση στο έργο παρακολούθησης βλέμματος.

Στην πρώτη φάση, όπως και στα προηγούμενα δύο σύνολα δεδομένων που χρησιμοποιήθηκαν στο έργο εκτίμησης της στοχοπροσήλωσης, πρέπει να δημιουργήσουμε τις επισημειώσεις που θα χρησιμοποιηθούν. Σε αυτό το σύνολο δεδομένων, τα αρχεία επισημειώσεων είναι σε παρόμοια μορφή με το σύνολο δεδομένων ASD Games, όπου κάθε βίντεο χωρίζεται σε χρονικά τμήματα με ακρίβεια έως και χιλιοστών του δευτερολέπτου. Ωστόσο, στα προβλήματα παρακολούθησης βλέμματος, οι είσοδοι του μοντέλου αποτελούνται από συνεχή καρέ του βίντεο, ακόμα και αν δεν ανήκουν στην ίδια κατηγορία. Για αυτό το λόγο, το σύστημα αρχείων μας δεν είναι το ίδιο με αυτό που είδαμε στο πρόβλημα εκτίμησης εμπλοκής· εδώ απλά εξάγουμε όλα τα καρέ κάθε βίντεο σε έναν φάκελο με το όνομα του βίντεο.

Επιπλέον, η αρχιτεκτονική του μοντέλου που θα χρησιμοποιήσουμε σε αυτή τη μελέτη είναι παρόμοια με αυτή που προτείνεται στο 0.0.14, το οποίο δέχεται ως είσοδο 7 συνεχόμενα καρέ και προβλέπει το βλέμμα του μεσαίου καρέ. Για αυτό το λόγο, για να δημιουργήσουμε τις επισημειώσεις, περνάμε από τα καρέ κάθε βίντεο με βήμα των πέντε καρέ, και δημιουργούμε κάθε δείγμα επισημείωσης να αποτελείται από το όνομα του βίντεο στο οποίο ανήκει, το αρχικό καρέ από τα επτά και την κατηγορία στην οποία ανήκει η κατεύθυνση του βλέμματος, δηλαδή την κατηγορία που ανήκει το μεσαίο καρέ, η οποία βρίσκεται από τα τμήματα χρονικών στιγμών.

Όπως έχουμε δει σε πολλές μελέτες που προσεγγίζουν το πρόβλημα παρακολούθησης βλέμματος, για να αντιμετωπίσουν καλύτερα το πρόβλημα και να παράγουν καλύτερες εκτιμήσεις, τα δίκτυα πρέπει να τροφοδοτούνται με συνεχόμενες περικοπές προσώπων των υποκειμένων ως εισερχόμενα και όχι με ολόκληρα καρέ [39, 26, 12]. Για αυτό το λόγο, πρέπει να συλλέξουμε τις περικοπές προσώπων από το συγκεκριμένο σύνολο δεδομένων. Ωστόσο, προκύπτει μια πρόκληση λόγω της φύσης των επισημειώσεων του έργου παρακολούθησης βλέμματος στο σύνολο δεδομένων WiDo σε σύγκριση με άλλα σύνολα δεδομένων που εστιάζουν συγκεκριμένα στο πρόβλημα παρακολούθησης βλέμματος. Στο WiDo, οι ετικέτες αποτελούνται από κατηγορίες κατεύθυνσης του βλέμματος, ενώ σε άλλα σύνολα δεδομένων αποτελούνται από διανύσματα κατεύθυνσης του βλέμματος. Έτσι, στο δικό μας σύνολο δεδομένων δεν αρκεί απλώς να προβλέψουμε τα διανύσματα κατεύθυνσης του βλέμματος, αλλά θα πρέπει επίσης να ελέγξουμε σε ποια κατηγορία από τις έξι ανήκει αυτή η κατεύθυνση. Για να διευκολύνουμε αυτή τη συσχέτιση μεταξύ του διανύσματος βλέμματος και των κατηγοριών, αποφασίσαμε πρώτα να συγχωνεύσουμε κάποιες από τις κατηγορίες. Έτσι, αντί να έχουμε έξι διαφορετικές κατηγορίες, τώρα το σύνολο δεδομένων μας αποτελείται από τρεις: (i) Κοιτάζει κάποιο μέρος του σώματος της μητέρας, (ii) Κοιτάζει το πρόσωπο της μητέρας και (iii) Κοιτάζει κάπου αλλού. Ωστόσο, εξακολουθούμε να χρειαζόμαστε τις συντεταγμένες του σημείου εκκίνησης του βλέμματος του παιδιού, του κεφαλιού της μητέρας και του σώματος της μητέρας.

Στην πρώτη φάση, για να ανιχνεύσουμε το περίγραμμα της μητέρας και του παιδιού σε κάθε καρέ, χρησιμοποιούμε το μοντέλο YOLO [38]. Για να γίνει η διάκριση του σε ποιον ανήκει κάθε πλαίσιο (μητέρα ή παιδί) βάσει των συντεταγμένων x_min κάθε περιγράμματος και ποιο άτομο είναι στα αριστερά. Για αυτό το λόγο, πρέπει να είμαστε ιδιαίτερα προσεκτικοί και να παρακολουθούμε κάθε λίγες εκατοντάδες καρέ κάθε βίντεο τη σχετική θέση μεταξύ της μητέρας και του παιδιού σε περίπτωση που αλλάξει, καθώς στα βίντεο δεν υπάρχουν σταθερές θέσεις μητέρας-παιδιού. Αφού συλλέξουμε τα δύο περιγράμματα, χρησιμοποιούμε το πλαίσιο MediaPipe [51] για να ανιχνεύσουμε τα σημεία ορόσημα του προσώπου και του σώματος κάθε ατόμου και να εξαγάγουμε τα περιγράμματα του προσώπου και του σώματος της μητέρας και το αρχικό σημείο βλέμματος του παιδιού. Το αρχικό σημείο βλέμματος του παιδιού υπολογίζεται ως ο μέσος όρος των συντεταγμένων του αριστερού και δεξιού ματιού που βρίσκονται στα σημεία ορόσημα του προσώπου του παιδιού. Αποθηκεύουμε αυτές τις συντεταγμένες για κάθε καρέ σε ένα αρχείο αντιστοίχισης που αντιστοιχεί το βίντεο/καρέ στο σύνολο των συντεταγμένων του, το οποίο θα περάσει στο μοντέλο κατά τη διάρκεια της εκπαίδευσης για να το βοηθήσει να κάνει τις προβλέψεις του.

Τέλος, δημιουργούμε ένα αρχείο αντιστοίχισης που συσχετίζει κάθε καρέ με τις συντεταγμένες του κεφαλιού

και του σώματος της μητέρας και το σημείο εκκίνησης του βλέμματος του παιδιού. Αυτό το αρχείο παρέχεται στον dataloader μαζί με τα αρχεία επισημειώσεων, έτσι ώστε κατά τη διάρκεια της εκπαίδευσης να παρέχει στο μοντέλο τα εισερχόμενα καρέ μαζί με τις αντίστοιχες συντεταγμένες των μεσαίων καρέ.

Υλοποίηση προεκπαιδευμένου μοντέλου Gaze360 για την εκτίμηση βλέμματος

Όπως αναφέραμε στην προηγούμενη ενότητα, το έργο της περαιτέρω εκπαίδευσης του συνόλου δεδομένων WiDo για την παρακολούθηση βλέμματος μπορεί να είναι πολύ απαιτητικό λόγω της φύσης του προβλήματος, καθώς και του γεγονότος ότι οι ετικέτες δεν είναι συντεταγμένες κατεύθυνσης. Για αυτό το λόγο, ως επόμενο βήμα της μελέτης μας δημιουργούμε ένα παρόμοιο πλαίσιο, λαμβάνοντας ολόκληρο το προεκπαιδευμένο μοντέλο στο σύνολο δεδομένων Gaze360, προσθέτοντας τη μονάδα ελέγχου διασταύρωσης περιγραμμάτων και δοκιμάζοντας πώς θα αποδώσει, χωρίς περαιτέρω εκπαίδευση στο σύνολο δεδομένων WiDo στο πρόβλημα παρακολούθησης βλέμματος με τα περιγράμματα κεφαλιού και σώματος της μητέρας.

Τα αποτελέσματα που προέκυψαν ήταν σημαντικά βελτιωμένα και εμφανίζονται παρακάτω:

Accuracy	w. F1	w. Precision
68.33	69.36	72.08

Table 4: Επίδοση στο πλήρως προεκπαιδευμένο μοντέλο εκτίμησης του βλέμματος στο WiDo σύνολο δεδομένων για 3 κατηγορίες.

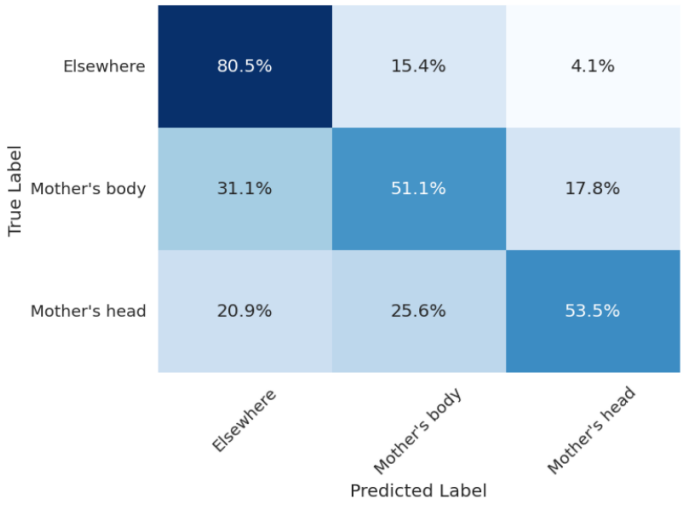


Figure 0.0.19: Πίνακας σύγχισης για την εκτίμηση βλέμματος στο WiDo σύνολο δεδομένων χρησιμοποιώντας 3 κατηγορίες και το πλήρως προεκπαιδευμένο μοντέλο Gaze360.

Όπως βλέπουμε, η ακρίβεια που επιτεύχθηκε και στις τρεις κατηγορίες έχει βελτιωθεί σημαντικά, υποδεικνύοντας πως η υλοποίηση του δικτύου μας μπορεί να χρησιμοποιηθεί για ένα τόσο απαιτητικό έργο, όπως αυτό.

Εκτίμηση Βλέμματος χρησιμοποιώντας 4 κατηγορίες

Ωστόσο, για να διερευνήσουμε περαιτέρω τα μοτίβα βλέμματος των παιδιών που δεν αναπτύσσονται τυπικά, θα χρειαστεί επίσης να εξετάσουμε στις μελέτες μας πόσο εστιασμένα είναι στο έργο/αντικείμενο με το οποίο ασχολούνται. Αυτό είναι σημαντικό, καθώς τα παιδιά με αυτισμό (ASD) τείνουν να χάνουν την εστίαση από την αλληλεπίδραση και να αποσπώνται πιο εύκολα από άλλα αντικείμενα σε σχέση με τα παιδιά που αναπτύσσονται κανονικά. Για αυτόν τον λόγο, τώρα διαιρούμε περαιτέρω την κατηγορία «Κάπου αλλού» που χρησιμοποιήσαμε στην προηγούμενη ενότητα, και το πρόβλημα μας αποτελείται από τις εξής τέσσερις κατηγορίες:

1. Δεν κοιτάζει κάπου συγκεκριμένα ή την κάμερα / Κοιτάζει ένα αντικείμενο που χρησιμοποιεί μόνο το παιδί και όχι η μητέρα (Νέα κατηγορία "Κάπου αλλού")
2. Κοιτάζει ένα αντικείμενο που χρησιμοποιείται στην αλληλεπίδραση με τη μητέρα
3. Κοιτάζει κάποιο μέρος του σώματος της μητέρας
4. Κοιτάζει το κεφάλι της μητέρας

Εκπαίδευση YOLO για ανίχνευση αντικειμένων

Για να μπορέσουμε να συνεχίσουμε με τον στόχο μας να προβλέψουμε το βλέμμα των παιδιών με βάση τις 4 προηγούμενες κατηγορίες, προκύπτει ένα πρόβλημα. Το πρόβλημα είναι ότι το συγκεκριμένο σύνολο δεδομένων δεν περιέχει τις συγκεκριμένες συντεταγμένες των αντικειμένων που χρησιμοποιούνται στην εικόνα. Έτσι, για να μπορέσουμε να διερευνήσουμε το πρόβλημα μας χρησιμοποιώντας τη μέθοδό μας που περιγράφεται στο Σχήμα 4.2.2, θα χρειαστεί να συλλέξουμε τις συντεταγμένες των παιχνιδιών. Για να το πετύχουμε αυτό, ολοκληρώνουμε επιπλέον εκπαίδευση στο μοντέλο yolo, δημιουργώντας τα δικά μας μικρά σύνολα δεδομένων των αντικειμένων που θέλουμε να ανιχνεύσουμε, χρησιμοποιώντας εικόνες που επισημαίνουμε οι ίδιοι. Πραγματοποιούμε αυτήν την ανίχνευση αντικειμένων σε διάφορα τμήματα βίντεο 5 παιδιών από το σύνολο δεδομένων WiDo.

Αποτελέσματα Εκτίμησης Βλέμματος σε 4 κατηγορίες

Αφού έχουμε προεπεξεργαστεί τα δείγματα και συλλέξει τις απαιτούμενες συντεταγμένες, δοκιμάζουμε την ίδια αρχιτεκτονική δικτύου όπως στην προηγούμενη ενότητα. Τα αποτελέσματα που παράγονται εμφανίζονται στο Σχήμα 0.0.20.

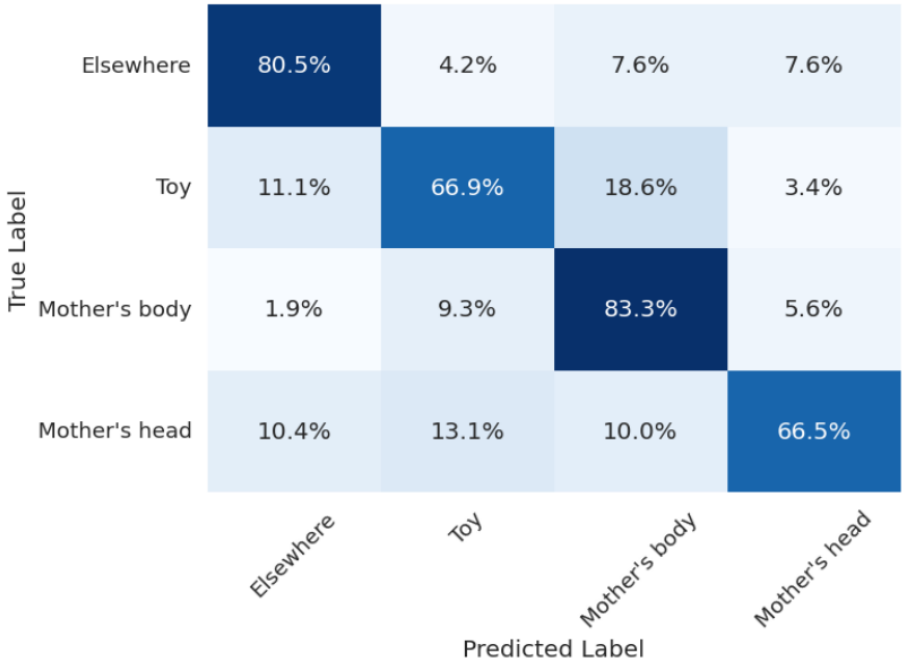


Figure 0.0.20: Αποτελέσματα που παράχθηκαν για ανίχνευση βλέμματος σε τέσσερις κατηγορίες στο σύνολο δεδομένων WiDo.

Όπως βλέπουμε, το δίκτυο αποδίδει πολύ καλά, καταφέρνοντας να διακρίνει τα δείγματα μεταξύ των τεσσάρων διαφορετικών κατηγοριών με υψηλό ποσοστό. Οι τιμές των μετρικών που επιτεύχθηκαν με αυτή τη μέθοδο εμφανίζονται στον Πίνακα 5, με πολύ υψηλή ακρίβεια **69.24%** και ένα ζυγισμένο F1 score **72.34%**. Είναι κρίσιμο να γνωρίζουμε αν το παιδί κοιτάζει το παιχνίδι (αντικείμενο αλληλεπίδρασης) ή κάποιο άλλο αντικείμενο γύρω, και όχι απλώς αν κοιτάζει κάποιο μέρος του σώματος της μητέρας του, καθώς αυτός είναι ένας τρόπος να διακρίνουμε

τα παιδιά με ASD από τα κανονικά αναπτυσσόμενα, καθώς τείνουν να αποσπώνται πολύ εύκολα από άλλα αντικείμενα ή ανθρώπους στο περιβάλλον τους. Έτσι, όπως φαίνεται από τον πίνακα σύγχυσης, επιτυγχάνουμε ένα πολύ υψηλό ποσοστό 80.5% στην ανίχνευση της κατηγορίας "κάπου αλλού", που υποδεικνύει τις στιγμές που το παιδί αποσπάται. Παρόλα αυτά, το μοντέλο δεν είναι ακόμα τέλειο, καθώς το πρόβλημα παρακολούθησης βλέμματος, ειδικά σε βίντεο όπου υπάρχει μεγάλη κίνηση από τους ανθρώπους που συμμετέχουν, μπορεί να είναι ιδιαίτερα απαιτητικό. Για παράδειγμα, μπορεί να υπάρχουν στιγμές που το αντικείμενο που αποσπά την προσοχή να βρίσκεται πίσω από το παιχνίδι ή τη μητέρα, οπότε, παρόλο που το βλέμμα παράγεται σωστά, το δείγμα μπορεί να ταξινομηθεί λανθασμένα σε άλλη κατηγορία.

Accuracy	w. F1	w. Precision
69.24	72.34	80.50

Table 5: Επίδοση της πλήρως προεκπαιδευμένης αρχιτεκτονικής εκτίμησης του βλέμματος στο WiDo σύνολο δεδομένων χωρισμένο σε 4 κατηγορίες.

Στο Σχήμα 0.0.21 παρουσιάζουμε μερικές από τις προβλέψεις βλέμματος που παράγονται από αυτήν την αρχιτεκτονική στο πρόβλημα που αποτελείται από τέσσερις κατηγορίες, χρησιμοποιώντας καρέ από διαφορετικά παιδιά. Όπως βλέπουμε, οι προβλέψεις είναι αρκετά ακριβείς, κάτι που είναι απαραίτητο, ειδικά για περιπτώσεις όπου το πλαίσιο είναι μικρότερο, όπως για παράδειγμα το κεφάλι της μητέρας.

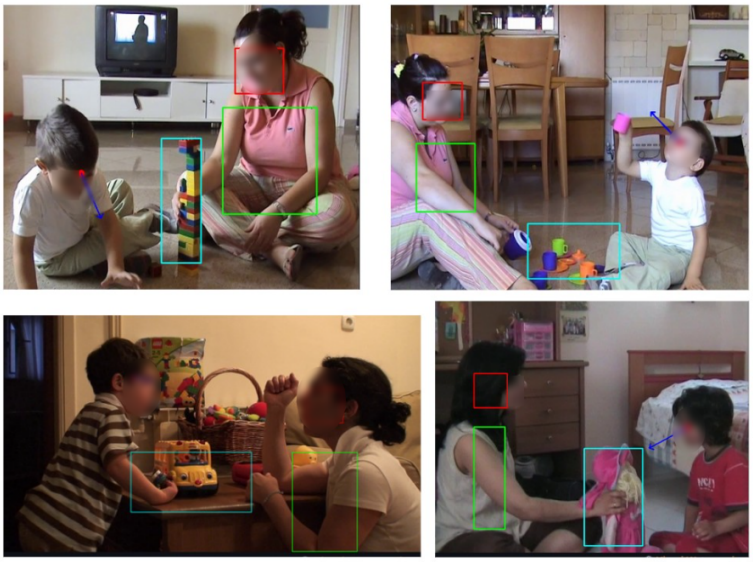


Figure 0.0.21: Αποτελέσματα που παράχθηκαν για την ανίχνευση βλέμματος σε τέσσερις κατηγορίες στο σύνολο δεδομένων WiDo.

Πειραματικά Αποτελέσματα για τη Συγχώνευση Εκτίμησης Στοχοπροσήλωσης και Βλέμματος

Αφού υλοποιήσαμε το δίκτυο Engagement Plus Gaze που προτείναμε, το οποίο συνδυάζει τα χαρακτηριστικά ανίχνευσης βλέμματος από το μοντέλο βλέμματος, μαζί με τα χαρακτηριστικά στοχοπροσήλωσης που παράγονται από την αρχιτεκτονική του μοντέλου στοχοπροσήλωσης, συνεχίζουμε με τις δοκιμές. Όπως αναφέραμε προηγουμένως, το δοκιμάσαμε χρησιμοποιώντας διαφορετικές τεχνικές συγχώνευσης, συμπεριλαμβανομένης της συγχώνευσης των χαρακτηριστικών με τα αποτελέσματα που παράγονται μετά την αρχιτεκτονική, πριν τροφοδοτηθούν στο tsn head, αλλά τα καλύτερα αποτελέσματα παράχθηκαν με την υλοποίηση που φαίνεται στο Σχήμα 0.0.15. Ο πίνακας σύγχυσης για αυτά τα καλύτερα αποτελέσματα εμφανίζεται στο Σχήμα 0.0.22.

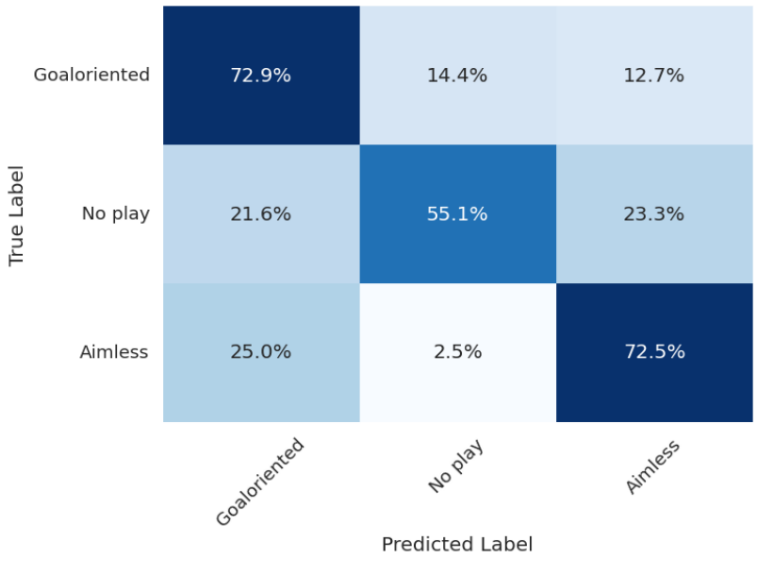


Figure 0.0.22: Ο πίνακας σύγχυσης για τα καλύτερα αποτελέσματα που παράχθηκαν χρησιμοποιώντας το πλαίσιο EngagementPlusGaze στο σύνολο δεδομένων Pinsoro [44].

Όπως βλέπουμε, η κατανομή των προβλέψεων έχει βελτιωθεί σημαντικά. Παρόλο που παρατηρούμε μείωση στην ακρίβεια της κατηγορίας "προσανατολισμένης στο στόχο", αυτό εξισορροπείται από την αύξηση της κατανομής για την κατηγορία "χωρίς παιχνίδι" (no-play), η οποία ήταν πολύ χαμηλή στα προηγούμενα αποτελέσματα. Αυτή η αύξηση είναι πολύ σημαντική, καθώς το δίκτυο μπορεί να διακρίνει τις τρεις κατηγορίες με μεγάλη βεβαιότητα.

Στον Πίνακα 6, μπορούμε να δούμε τα αποτελέσματα των μετρικών του μοντέλου Engagement Plus Gaze σε σύγκριση με το μοντέλο της Ενότητας 4.4, όπου το μοντέλο ανίχνευσης βλέμματος δεν περιλαμβανόταν. Μπορούμε να παρατηρήσουμε μια μείωση στην ακρίβεια και την precision, κυρίως λόγω της μείωσης στην κατηγορία προσανατολισμένη στο στόχο, η οποία έχει τα περισσότερα δείγματα. Ωστόσο, το πιο σημαντικό αποτέλεσμα είναι η αύξηση της τιμής του ζυγισμένου F1 score, που είναι ο κύριος δείκτης μετρικής μας.

Model Comparison			
Model	Accuracy	w. F1	w. Precision
C3DI3D-TSN-original (ours)	61.85	61.24	62.72
C3DI3D-TSN (ours)	**68.52**	65.28	**72.82**
C3DI3D-PlusGaze-TSN (ours)	66.25	**66.15**	67.95

Table 6: Σύγκριση επιδόσεων των τριών διαφορετικών μας αρχιτεκτονικών στο πρόβλημα της εκτίμησης της στοχοπροσήλωσης στο Pinsoro σύνολο δεδομένων.

Chapter 1

Introduction

Studying child interactions is crucial for understanding how children develop social, emotional, and cognitive skills, particularly in early childhood. Interactions with caregivers, peers, and teachers help shape a child's ability to communicate, solve problems, and build relationships. In recent years, Child-Robot Interaction (CRI) has emerged as an important area of research, offering new insights into how children engage with technology. Robots designed for educational or therapeutic purposes can serve as interactive partners, helping children learn through play and social exchanges. Key to these interactions is the child's engagement, which is a valid indicator of how focused the child is during its interaction and how different approaches change its behaviour. Engagement from a visual perspective is often measured through behaviors like gaze, which indicates attention and focus. Gaze patterns reveal how children process information and their level of involvement in the interaction. Understanding engagement and gaze in both human and robot interactions can enhance the design of interventions that foster learning, socialization, and emotional development. This knowledge can lead to improved outcomes in education, therapy, and child-robot collaborations, but also in observing children with autism spectrum disorder (ASD) and how they interact differently compared to normally developing ones. Children with autism often face challenges in social engagement, communication, and attention, and robots designed for educational or therapeutic purposes offer a unique, non-judgmental platform for interaction. These robots can be programmed to adapt to individual needs, helping children with autism practice social behaviors in a structured, controlled environment, as well as possibly identifying them.

1.1 Engagement in Child-Robot Interaction

Engagement, in a general sense, refers to the degree of attention, curiosity, interest, and involvement that an individual exhibits in a particular activity or interaction. It's a multidimensional concept that can involve emotional, cognitive, and behavioral components, where the individual is actively participating or mentally invested in the task at hand. Engagement is crucial for productivity and success in various contexts, from education and work to personal relationships. In learning, for instance, high engagement typically leads to better knowledge retention and skill acquisition, while disengagement can hinder progress and development [62].

Engagement in CRI, is a state in which a child exhibits interest, curiosity, and enthusiasm during its interactions with a human or even a robot [6]. This involves the child focusing their attention on the robot's actions and interactions, actively participating by initiating or responding to the robot's prompts. In addition to that, engagement is also characterized by an emotional response, such as laughter, smiles, or expressions of empathy, which indicate a deeper connection with the experience. Moreover, a key aspect of engagement is persistence, where the child continues to interact over an extended period, demonstrating sustained interest and involvement. There can be lots of positive outcomes from the child being engaged when interacting with others including facilitating the child's learning and exploration of new ideas, concepts, or skills, ultimately contributing to their productivity, satisfaction, and well-being. An important goal in the engagement task is to reach a point where a robot can correctly identify the level of engagement, just like it has been achieved in other tasks, like the creation of a CRI system with the joint capability of both action and emotion recognition [22]. That goal can be challenging as engagement is an internal mental state and cannot be easily directly observed [46], since the observing robot stays confined to the exploitation of external vision or audio cues to estimate its level [29, 18].

1.2 Gaze tracking

Gaze tracking is a powerful tool in understanding and analyzing visual attention patterns. By monitoring and recording the direction and focus of a child's gaze, researchers can gain insights into various cognitive and social processes, such as how children perceive and interact with their environment. In the context of developmental studies, gaze tracking can reveal important aspects of a child's social communication skills, attention span, and learning behaviors [55]. Usually, in order to tackle the gaze tracking problem successfully and with high precision, large tracking devices and experimental set-ups are required [41]. For these reasons, in our study we are going to focus on approaches that are based on video recordings during live interactions.

These kind of methods for gaze estimation have not yet reached peak performance, even though in recent years, methods for related human modeling problems, like 2D body pose and face tracking have achieved impressive success by leveraging the representational power of deep convolutional neural networks along with very large annotated datasets [11, 27, 34]. This is primarily due to the nature of the gaze tracking in the wild task, where there might be partial or total occlusion of the person's eyes or even the face, making it extremely hard for the modeling mechanisms to learn and predict on them. In addition, for the specific task to be tackled successfully there also needs to be precise and highly varied collection of gaze data with ground truth, particularly outside of the lab, which usually is a challenging deed.

1.3 Autism Spectrum Disorder (ASD)

Like mentioned before, there might be lots of positive outcomes by being able to study successfully child interactions. One of them might be to be able to build frameworks that identify successfully children with Autism Spectrum Disorder. Autism Spectrum Disorder (ASD) is a complex neurological and developmental disorder characterized by challenges in social communication, interaction, and a tendency toward restricted and repetitive behaviors. Autism manifests differently in each individual, with a wide range of symptoms and severity levels, making it a spectrum disorder. Common signs include reduced eye contact, limited facial expressions, difficulty understanding social cues, and atypical responses to sensory stimuli. These social communication deficits can lead to significant difficulties in forming relationships, engaging in typical daily activities, and succeeding in traditional educational settings.

The early detection of autism is crucial for several reasons. Early intervention has been consistently shown to improve developmental outcomes in children with ASD [77, 86]. By identifying the disorder during the critical early years of brain development, targeted therapies and educational programs can be implemented to address specific challenges. These interventions can significantly enhance a child's communication skills, social interactions, and overall quality of life, reducing the long-term impact of the disorder.

Moreover, early detection allows families to better understand their child's needs, access appropriate support services, and make informed decisions about their education and care. It also helps in alleviating parental stress by providing clarity and direction, enabling parents to actively participate in their child's developmental journey.

Given the profound implications of autism on an individual's life, as well as the societal and familial levels, early diagnosis is not just beneficial but essential. It lays the foundation for interventions that can alter the trajectory of a child's development, fostering greater independence and integration into society. Social robots have already been introduced in the process of identifying children with Autism Spectrum Disorder (ASD) or learning difficulties to tackle consequences and challenges of such disorders [74, 40], and not only that, but also there are promising results in the use of robots in supporting the social and emotional development of children with ASD [70, 69, 2]. As research continues to evolve, developing more accurate and accessible early detection methods remains a top priority in the field of autism research and care.

1.3.1 Engagement in Detecting Autism

Engagement plays a crucial role in detecting autism, as children with autism often exhibit difficulties in social communication and interaction, such as reduced eye contact, limited facial expressions, and challenges in responding to social cues. Additionally, it is common for them to demonstrate atypical gaze patterns, such as reduced attention to social stimuli like faces or a tendency to fixate on specific objects rather than people. Engagement recognition technologies offer a promising approach to detecting these behaviors, providing objective and quantitative measures of children's social and communicative behaviors. These technologies offer a resource-efficient screening method, allowing for a cost-effective and scalable approach to screening large populations of children, making early detection of autism more accessible and widespread [17, 53].

1.3.2 Gaze tracking in Detecting Autism

The technology of gaze tracking can be potentially extremely valuable in identifying atypical gaze patterns that may be indicative of developmental disorders, such as autism spectrum disorder (ASD). For instance, children with ASD often exhibit unusual gaze patterns, such as reduced eye contact or more focus on objects rather than people. Gaze tracking in video analysis allows for the objective measurement of these behaviors, providing a non-invasive and precise method for assessing a child's engagement and social interaction in naturalistic settings [35, 50]. As such, it holds significant potential for early detection, intervention, and the overall understanding of developmental trajectories in children.

1.4 Goals and structure of this study

All the above demonstrate the importance of both the engagement estimation and gaze tracking in video interactions of children. They are technologies that can be proved extremely valuable both as standalone implementations, but also as part of a bigger picture, when used as tools in identifying developmental disorders, such as autism spectrum disorder (ASD). For this reason, this study aims in investigating how the engagement recognition and gaze tracking tasks can be used on visual datasets, in what ways their performance could possibly be further improved, and how could the gaze tracking help complement the engagement implementation in order to produce better results.

We try to achieve our main goal of enhancing the children engagement estimation using gaze tracking data by taking the following steps:

1. A lot of information is gathered about each problem separately, including previous methods and algorithms used to tackle them, like spatiotemporal hybrid networks and multi-feature fusion implementations.
2. We build a robust engagement framework consisting of a 3D ConvNet implemented using the Temporal Segment Network (TSN) architecture.
3. We use an already pretrained gaze tracking network, integrating it on a more general and into the wild gaze tracking problem focusing on child interactions with their mothers, which consist of having bounding boxes as labels and not the coordinates of the gaze direction, that are formally used.
4. After tackling successfully both the engagement recognition and gaze tracking tasks separately, we decide to use the gaze tracking network to further improve the results on the engagement estimation problem.

This thesis is separated in 5 chapters. Below, follows a brief summary of their contents.

- In chapter 1, the focus of this study is introduced, which centers on Engagement Estimation and Gaze Tracking during children's interactions, why they should be improved and how they can help in different areas.
- In chapter 2, the theoretical background of this study is presented, explaining important structures, methodologies and models used in similar Computer Vision problems, as well as algorithms and practices used in the training of Machine Learning models.
- In chapter 3, an extensive study of related work on the areas of engagement recognition and gaze tracking is presented, featuring different model architectures and implementations built and how they try to tackle each one of their respective tasks.
- In chapter 4, first we present the methods that we propose to tackle the Engagement Estimation and Gaze Tracking problems. Those include an engagement estimation architecture consisting of a backbone and the TSN method, a gaze estimation implementation and finally a combined engagement and gaze

implementation that fuses gaze tracking and engagement data. Then we analyze the experiments that were conducted in this thesis and the results that they produced are presented. Those include a high level performance on the engagement estimation problem that is up there with other recent architectures, an accurate gaze estimation on children into the wild and a further improvement of the engagement estimation performance by fusing the engagement and gaze tracking networks.

- In chapter 5, a synopsis of this study is conducted, presenting the conclusions of the experimentation with these two fields and the model architectures that were constructing, and additionally suggestions are made about possible future research work that can be conducted based on the presented work.

Chapter 2

Theoretical Background

2.1 Convolutional Neural Networks

2.1.1 The neuron

The fundamental processing unit of a Neural network, the neuron, is inspired from the observation of the brain's formation. So just like in the brain where the neurons form a complex, highly interconnected network and send electrical signals to each other to help humans process information, the neurons of the neural networks work together on a common problem, using computing systems to solve mathematical calculations. Specifically, a neural network's neuron processes a local patch of input data, uses learnable weights to compute a weighted sum also adding a learnable bias, applies an activation function, and contributes to the formation of feature maps. The activation function that is being applied interprets the result and presents it in a meaningful way, often introducing non-linearity into the network, allowing it to model more complex patterns.

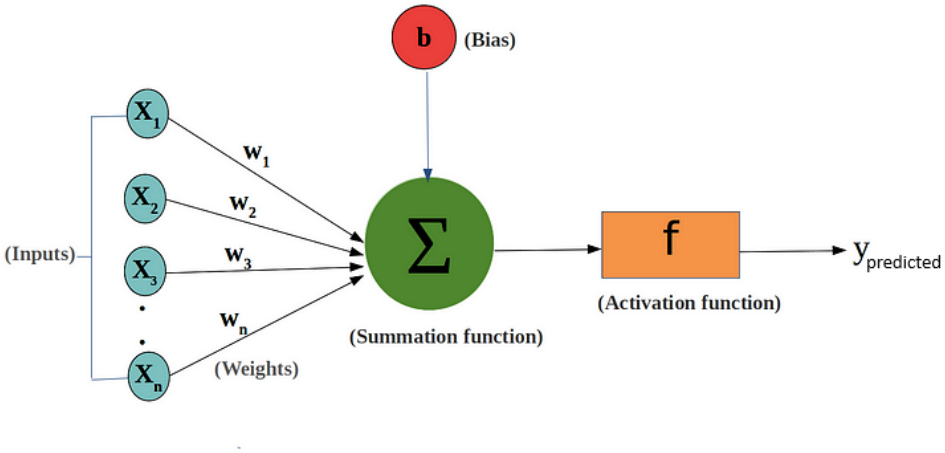


Figure 2.1.1: The neuron architecture, the neuron takes the weighted sum of the input data and the learnable weights, adds a learnable bias, applies an activation function and produces the final output from [67].

Activation functions

Like mentioned above, activation functions are functions that are applied to the weighted sum of the input signals with the weights of the neuron and often they introduce a non-linearity to the computation of the result. Some activation functions that are usually used are:

ReLU Currently the most used in the world, ReLu thresholds at zero by outputting:

$$f(x) = \max(0, x) \tag{2.1.1}$$

This function is easily implemented and eliminates the problem of the saturating gradient. An issue with this function is that all the negative values given as input become zero immediately, which in turns affects the results by not mapping the negative values appropriately.

Leaky-ReLU Leaky-ReLU tries to encounter the issue with ReLu by multiplying the negative inputs with a small constant, instead of completely zeroing negative inputs:

$$f(x) = \mathbb{1}(x < 0)(\alpha x) + \mathbb{1}(x \geq 0)(x) \tag{2.1.2}$$

Sigmoid This activation function is mainly used, due to the fact, that it exists between 0 to 1. Therefore, it is especially used for models where we have to predict the probability as an output. Since probability of anything exists only between the range of 0 and 1, sigmoid is the right choice. Its formula is the following:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.1.3}$$

An issue with this function is that it can cause a neural network to get stuck at the training time.

Tanh The tanh function is an improved version of the sigmoid one, but being zero-centered as shown in 2.1.2. Tanh maps input to the range $[-1, 1]$, instead of $[0, 1]$, with the following formula:

$$\tanh(x) = s\sigma(2x) - 1 \tag{2.1.4}$$

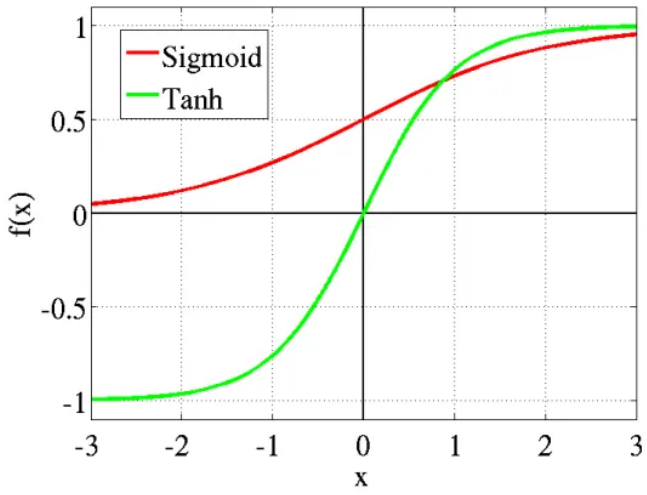


Figure 2.1.2: Sigmoid vs Tanh activation functions from [24].

However, it still may cause the gradient to saturate.

2.1.2 Neural Networks

Neural Networks [43] are consisted of layers of interconnected neurons, which work together to process input data, recognize patterns, and make decisions or predictions. The first of the layers, which receives the raw data is the input layer, followed by multiple intermediate layers, called Hidden Layers, where neurons process inputs independently and detect patterns. These layers perform transformations on the data and are not directly observable from the outside. Finally, the final layer, called the output layer, produces the network's predictions or decisions and its structure depends on the task (e.g., classification, regression). In order for

the weights of the neural networks to be updated according to the backpropagation algorithm, the gradient of the loss is needed. The loss is calculated using a loss function, which compares the output of the last layer to the ground truth.

Loss functions

The generalized equation for the loss function is:

$$J(w^T, b) = \frac{1}{m} \sum_{i=1}^{m} L(\hat{y}^{(i)}, y^{(i)}) \tag{2.1.5}$$

where w are the weights, b the bias, m is the total number of training set data points, $\hat{y}$ is the prediction and y is the ground truth.

Mean Squared Error The most popular loss function calculating the difference between the actual value and the predicted value, squaring it and in the end taking the mean of it:

$$MSE = \frac{1}{m} \sum_{i=1}^{m} (y_{actual} - y_{predicted})^2 \tag{2.1.6}$$

Its disadvantages are that it is affected by outliers and that it can't be used to interpret or compare directly with actual value.

Cross Entropy Loss The Cross Entropy Loss is a loss function especially used for classification tasks, measuring the difference between two probability distributions: the predicted probability distribution and the actual distribution of the labels (often represented as one-hot encoded vectors). For each class, it calculates the negative log of the predicted probability corresponding to the actual class and sums these across all classes:

$$L = -\sum_{i=1}^{n} y_i \log(\hat{y}_i) \tag{2.1.7}$$

The fact that it is a smooth and differentiable function which provides useful gradients for learning and is more effective in handling class probabilities makes it preferable in classification tasks compared to other loss functions, like mean squared error (MSE).

Optimization - Gradient Descent

Optimization is the process of adjusting the parameters of a model (weights) in order to minimize or maximize some objective function, typically the loss function. This can be achieved using Gradient Descent, an optimization algorithm which tries to minimize a function by iteratively moving towards the steepest descent, as defined by the negative of the gradient.

$$\theta = \theta - \eta \nabla_\theta J(\theta) \tag{2.1.8}$$

where η is the learning rate. Gradient Descent is the procedure of repeatedly evaluating the gradient and then performing a parameter update.

Backpropagation Algorithm

The backpropagation algorithm is a widely used method for training artificial neural networks by minimizing the error between the predicted output and the actual target output. It involves computing the gradient of the loss function with respect to each weight by the chain rule, efficiently propagating errors backward through the network layers. This process allows the model to adjust its weights to reduce prediction error, ultimately improving its performance, as in large neural networks the relationship of some weights and the loss function is very hard to find. The Backpropagation algorithm works by two different passes, that are repeated for some epochs, the forward pass and the backward pass.

In the forward pass, we start by propagating the data inputs to the input layer, go through the hidden layer(s), measure the network's predictions from the output layer, and finally calculate the network error based on the predictions the network made. Once the network error is calculated, then the forward propagation phase has ended, and backward pass starts.

In the backward pass, the flow is reversed so that we start by computing the gradient of the loss function with respect to the output, then applying the chain rule is iteratively to compute the gradient of the loss function with respect to each layer's parameters and inputs and in the end the gradients are propagated backward through the network to update the weights.

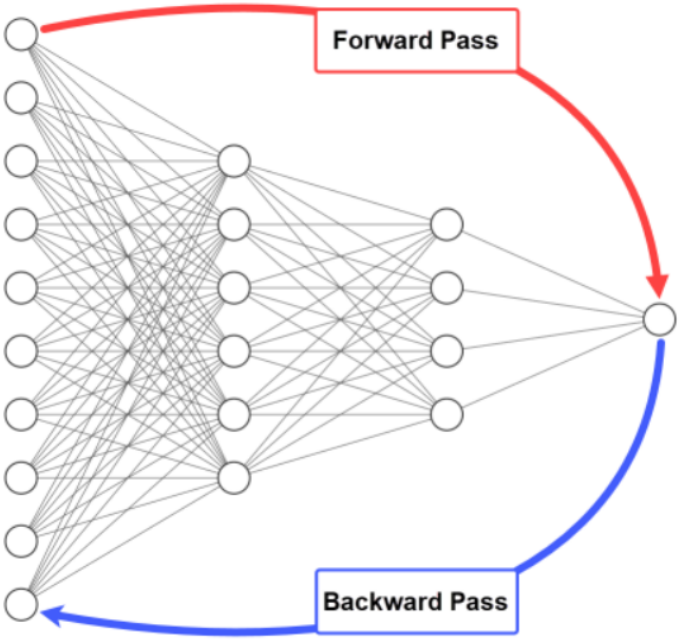


Figure 2.1.3: The forward and the backward pass from [64].

So, firstly we compute the Gradient of the Loss with Respect to Output Layer Weights:

$$\frac{\partial L_i}{\partial W_{output}} = \frac{\partial L_i}{\partial O} \frac{\partial O}{\partial W_{output}} \tag{2.1.9}$$

where L the loss, W the weights and O the output.

And then we propagate the error backwards using the chain rule, so for a hidden layer h we have:

$$\frac{\partial L_i}{\partial W_H} = \frac{\partial L_i}{\partial O} \frac{\partial O}{\partial h} \frac{\partial h}{\partial W_h} \tag{2.1.10}$$

Regularization

The goal of a neural network is to learn a correspondence of input to output from training data and apply it on test data. Thus, it is important to be able to generalize its weights and not learn specifically the examples from the training data. When a model learns the noise and details of the training data to such an extent that it performs poorly on new, unseen data it is called overfitting. Regularization is a technique used in deep learning to prevent overfitting, by adding a penalty to the loss function during model training, discouraging overly complex models and encouraging simpler, more generalizable models. Specifically, it adds an extra component to the loss function which prevents the weights from increasing excessively their magnitude and thus update in a less flexible way.

L1 regularization In L1 regularization, for each weight w we add the term $\lambda_1|w|$ to the loss function. It has the intriguing property that it leads the weight vectors to become sparse during optimization (i.e. very close to exactly zero). In other words, neurons with L1 regularization end up using only a sparse subset of their most important inputs and become nearly invariant to the "noisy" inputs. In practice, if you are not concerned with explicit feature selection, L2 regularization can be expected to give superior performance over L1.

L2 regularization L2 regularization is perhaps the most common form of regularization. It can be implemented by penalizing the squared magnitude of all parameters directly in the loss function. That is, for every weight w in the network, the term $\frac{1}{2}\lambda w^2$ is added to the loss function where λ is the regularization strength. The factor of $\frac{1}{2}$ is used so that the gradient of the term is simple λw. The L2 regularization has the intuitive interpretation of heavily penalizing peaky weight vectors and preferring diffuse weight vectors. This has the appealing property of encouraging the network to use all of its inputs a little rather than some of its inputs a lot. Additionally, during gradient descent parameter update, using the L2 regularization ultimately means that every weight is decayed linearly:$w+ = -\lambda \times w$ towards zero. It is possible to combine the L1 regularization with the L2 regularization: $\lambda_1|w| + \lambda_2 w^2$.

Dropout Dropout [**Sriv+14**] is a popular technique for regularizing neural networks. During each training iteration, the network randomly "drops out" a fraction (dropout probability) of neurons (along with their connections) from the network. This forces the network to learn more robust features and reduces the likelihood of over-reliance on specific neurons or paths through the network. It can be thought as sampling a Neural Network within the full Neural Network, and only updating the parameters of the sampled network based on the input data.

Data Augmentation A technique where additional training examples are created by modifying the original training data (e.g., rotating, flipping, or cropping images). It helps the model generalize better by learning from a wider variety of examples, thereby reducing overfitting. Often it is performed in less populated categories of the training dataset in order to close the gap on the difference with the majority categories.

Batch size

The batch size is a critical hyperparameter that refers to the number of training examples utilized in one forward/backward pass through the model. So, the model first makes its predictions to all the samples of each batch and after it is finished with that calculates the error by comparing the expected output variables with the predictions. The value of the batch size can vary and choosing the appropriate one affects the model's performance, training time, and how well it generalizes to unseen data. A small batch size provides a more accurate estimate of the gradient and helps avoid local minima by introducing more noise in the gradient descent process, but also results in slower training due to less efficient computation and requires more frequent updates to model parameters. On the other hand, a large batch size makes full use of parallel

hardware architectures, leading to faster computation, and also provides smoother and more stable gradient updates, but can lead to poorer generalization due to smoother loss landscapes and potentially getting stuck in sharp local minima.

2.1.3 Convolution

Convolution [25] is a mathematical operation applied on two functions (such as a filter or kernel and an input) to produce a third function that expresses how the shape of one is modified by the other. In simpler terms, it is a way of applying a filter (or kernel) to an input to create a feature map that highlights certain aspects or features of the input. Formally, in discrete 2-D convolutions the value of an unit at position (x, y) in the jth feature map in the ith layer, denoted as v_{ij}^{xy}, is given by the following formula:

$$v_{ij}^{xy} = \tanh(b_{ij} + \sum_{m} \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} w_{ij}^{pq} v_{(i-1)m}^{(x+p)(y+q)}) \tag{2.1.11}$$

where tanh is the hyperbolic tangent function, b_ij is the bias for this feature map, m indexes over the set of feature maps in the (i - 1)th layer connected to the current feature map, w_{ijk}^{pq} is the value at the position (p. q) of the kernel connected to the k_th feature map, and P_i and Q_i are the height and width of the kernel, respectively.

The kernel is spatially smaller than an image but is more in-depth. This means that, if the image is composed of three (RGB) channels, the kernel height and width will be spatially small, but the depth extends up to all three channels.

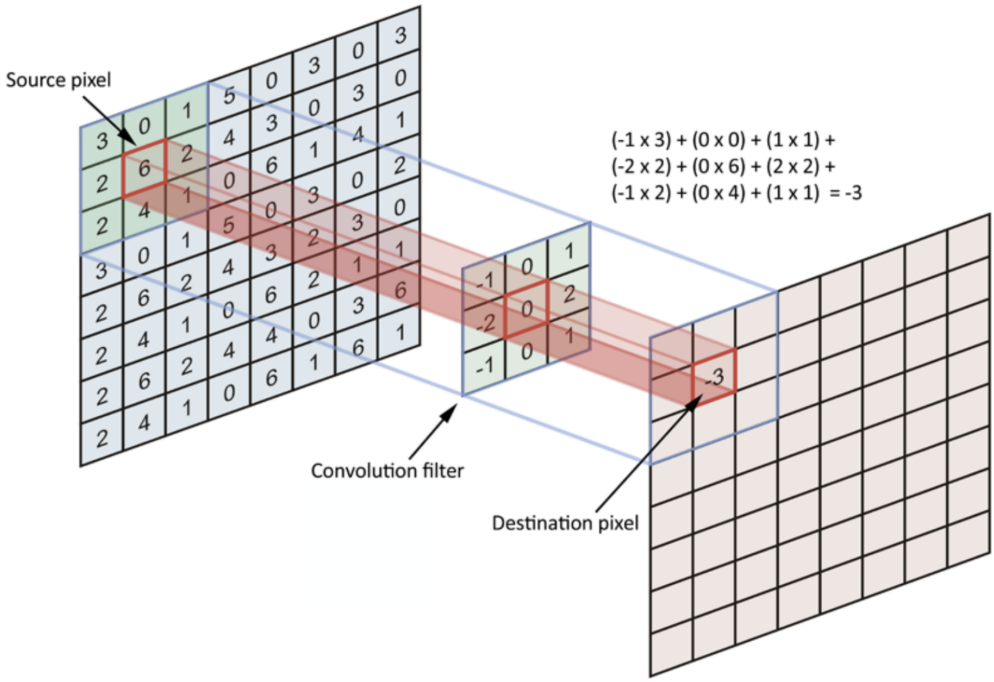


Figure 2.1.4: Convolution Visualization. The filter (kernel) is applied on an area of the input every time calculating a single point of the feature map from [9].

The convolution operation is used in CNNs to process an image and recognize on it specific characteristics. Unlike a regular Neural Network, the layers of a CNN have neurons arranged in 3 dimensions: width, height, depth. The neurons in a layer will only be connected to a small region of the layer before it, instead of all of

the neurons in a fully-connected manner. A simple CNN is a sequence of layers, and every layer of a CNN transforms one volume of activations to another through a differentiable function. The types of layers used to build a CNN architecture include: Convolutional Layer, Pooling Layer, and Fully-Connected Layer.

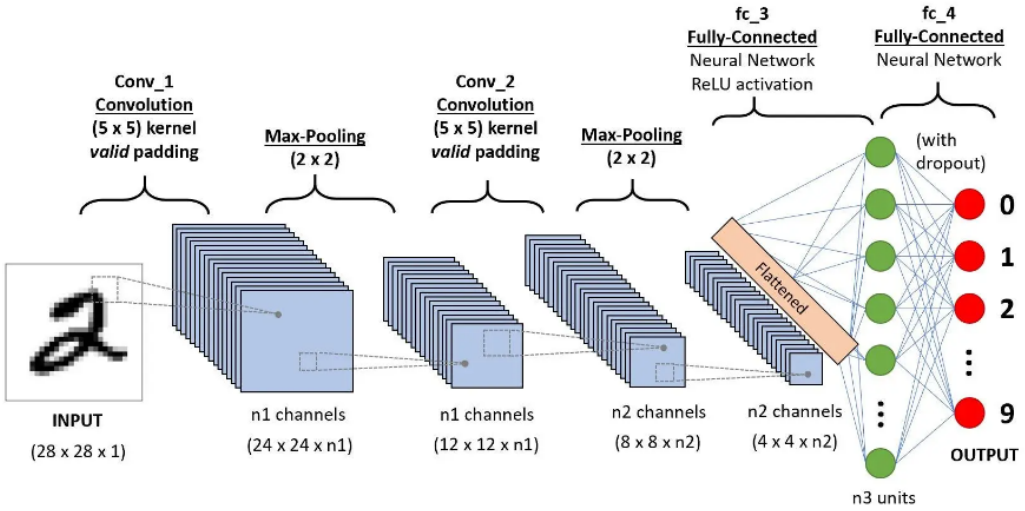


Figure 2.1.5: Convolution Network architecture from [64].

Due to the large number of pixels in an image, every neuron has a receptive field and is applied only to the corresponding layers. Therefore, it has $w \times h \times c$ number of weights, where $w \times h$ is the receptive field and c is the number of channels of the input image. That's why convolutional neural networks do not detect as sufficiently as attention mechanisms contextual information. The depth of the filter in each kernel, for example n1 and n2 in Figure 2.1.5 indicates that there are multiple neurons applied to the same patch of the input image. Therefore, the output image is also 3-dimensional with a depth equal to the depth of the filter. The purpose of multiple stacked filters is to detect multiple characteristics with one convolutional layer. Even though every neuron has a specific receptive field, by shifting the filter and computing the output of every patch there is not need for multiple neurons for every patch of the input image. The shifting is specified by the stride of the filter which indicates how many pixels the filter is slided across the image before it is reapplied. Additionally, another important hyperparameter is the padding, which can play an importat role in controling the output dimensions even more. It refers to adding extra pixels around the input image, usually zeros, which results in the increase of the final feature map compared to using no padding. Finally, assuming if we hace an input of w x w x D and Dout number of kernels with a spatial size of F with stride S and amount of padding P, then the size of output volume can be determined by the following formula: $\frac{W-F+2P}{S} + 1$.

2.1.4 Pooling layer

The pooling layer is used to replace the output of the network at certain locations by deriving a summary statistic of the nearby outputs. This helps in reducing the spatial size of the representation, which decreases the required amount of computation and weights. The pooling operation is processed on every slice of the representation individually.

There are several pooling functions such as the average of the rectangular neighborhood, L2 norm of the rectangular neighborhood, and a weighted average based on the distance from the central pixel. However, the most popular process is max pooling, which reports the maximum output from the neighborhood.

2.1.5 Batch Normalization

Batch normalization is used to standardize the inputs to any particular layer. That entails the input to have zero mean and unit variance. BN layer transforms each input in the current mini-batch by subtracting the input mean in the current mini-batch and dividing it by the standard deviation. However, it is not proven that the models performs better with zero mean and unit variance. It might perform better with some other mean and variance. Hence the BN layer also introduces two learnable parameters γ and β which adjust those parameters.

2.1.6 Fully connected layer

In the fully connected layer the neurons have full connectivity with all neurons in the preceding and succeeding layer as seen in regular FCNN. This is why it can be computed as usual by a matrix multiplication followed by a bias effect. The FC layer helps to map the representation between the input and the output.

2.2 Temporal Segment Network

Temporal segment network, first introduced in [75], is a video-level framework that tries to resolve the problem of complex actions, such as sports actions, taking place on multiple stages spanning over a relatively long time, by utilizing the visual information of entire videos to perform video-level prediction.

2.2.1 The Framework

The temporal segment framework like other spatiotemporal methods is also composed of spatial stream ConvNets and temporal stream ConvNets. But instead of working on single frames or frame stacks, temporal segment networks operate on a sequence of short snippets sparsely sampled from the entire video. Each snippet in this sequence will produce its own preliminary prediction of the action classes. Then a consensus among the snippets will be derived as the video-level prediction as shown in Figure 2.2.1. In the learning process, the loss values of video-level predictions, other than those of snippet-level predictions which were used in two-stream ConvNets, are optimized by iteratively updating the model parameters.

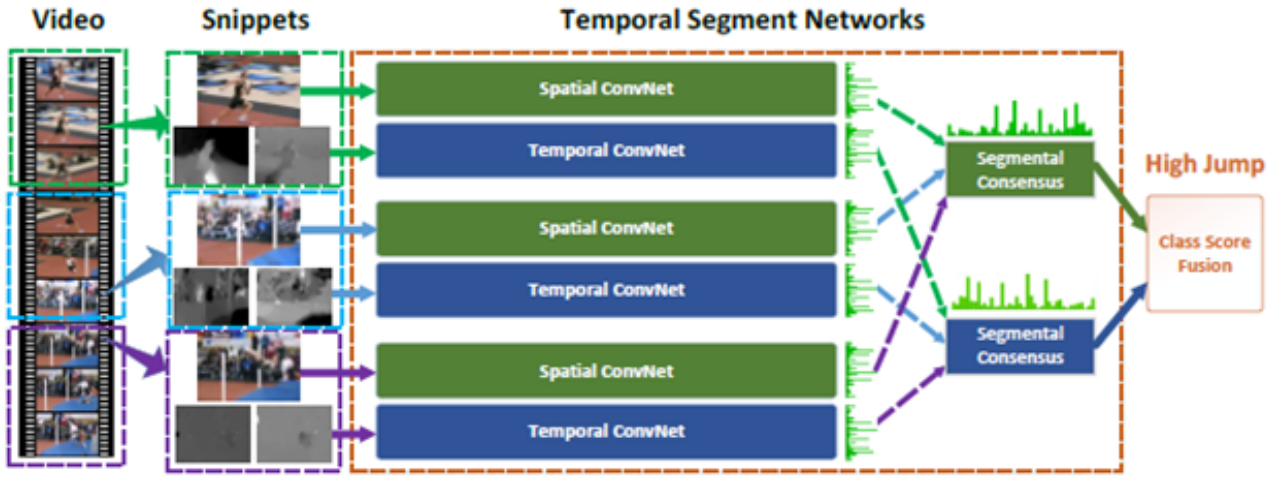


Figure 2.2.1: Temporal Segment Network Architecture from [75]

2.3 Residual Learning

2.3.1 Residual Block

Residual learning was first introduced in [30], in order to address the degradation problem. If we consider $H(x)$ as an underlying mapping to be fit by a few stacked layers (not necessarily the entire net), with x denoting the inputs to the first of these layers, then residual learning hypothesizes that the multiple nonlinear layers can asymptotically approximate the residual functions, i.e. $H(x) - x$ (assuming that the input and output are of the same dimensions). So instead of the stacked layers approximating $H(x)$, it explicitly lets these layers approximate a residual function $F(x) := H(x) - x$. The original function thus becomes $F(x) + x$, also assuming that it is easier to optimize the residual mapping than to optimize the original, unreferenced mapping.

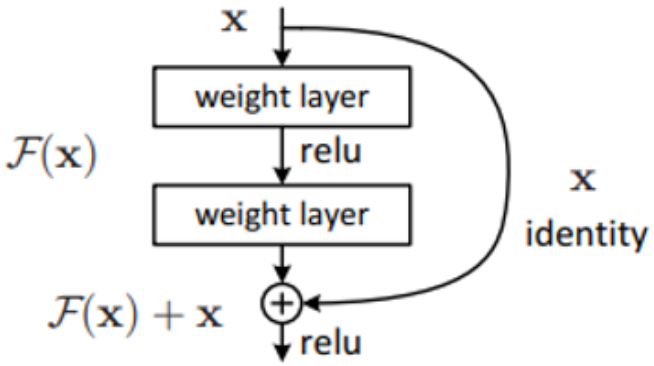


Figure 2.3.1: Residual learning block from [30].

The formulation of $F(x) + x$ can be achieved by feedforward neural networks with "shortcut connections" that add neither extra parameter nor computational complexity. In addition to that, the building block can represent multiple convolution layers and can be used right away if the input and output dimensions are the same. On the other hand, if the dimensions of x and F are not equal, either padding can be added or a linear projection W_s can be performed on x.

$$y = F(x, W_i) + W_s \cdot x \tag{2.3.1}$$

2.3.2 Residual Networks

Resnet [30] is a deep residual learning framework that accomplishes its goal of addressing the degradation problem ('Vanishing Gradients'), by adopting residual learning to every few stacked layers. As we can see in the Resnet-50 architecture in Figure 2.3.2 shortcut connections have been added which turn the plain network to its counterpart residual version. As mentioned above the identity shortcuts can be directly used when the input and output are of the same dimensions (solid line shortcuts in 2.3.2). When the dimensions increase (dotted line shortcuts in 2.3.2), two options are considered: (A) The shortcut still performs identity mapping, with extra zero entries padded for increasing dimensions. This option introduces no extra parameter; (B) The projection shortcut in 2.3.1 is used to match dimensions (done by 1×1 convolutions).

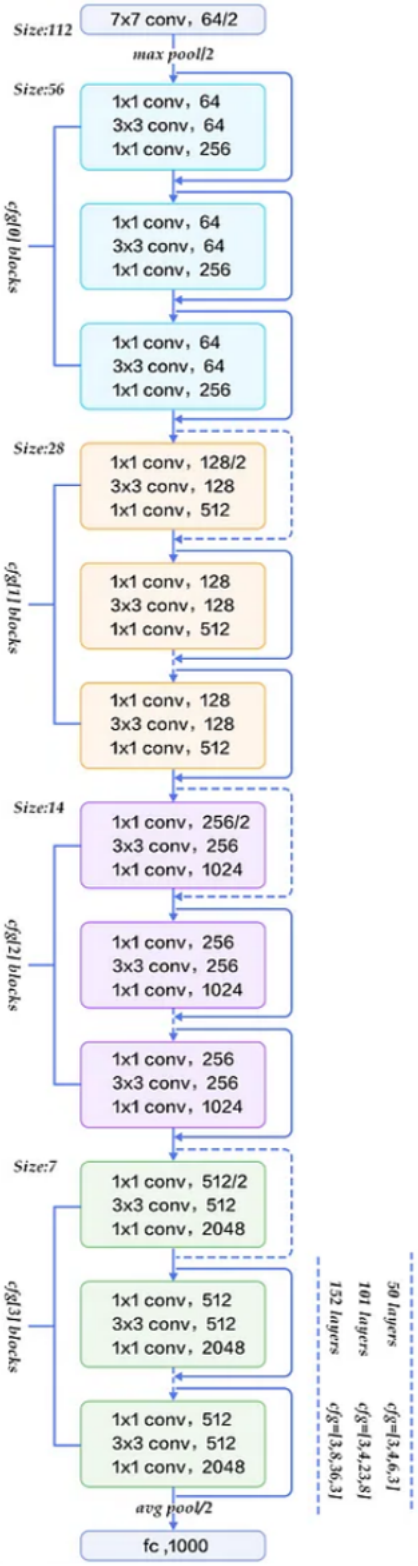


Figure 2.3.2: Resnet-50 architecture from [30]. Solid lines represent shortcuts with the same input and output dimensions, while the dotted lines, shortcuts with dimensions increase.

2.4 3D ConvNets

2.4.1 3D Convolutions

In 2D CNNs, convolutions are applied on the 2D feature maps to compute features from the spatial dimensions only. For this reason, when they are used in video analysis problems, they don't really capture the motion information encoded in multiple contiguous frames, which is what is desired. As a result, the 3D convolutions where proposed [36] to be performed in the convolution stages of CNNs to compute features from both spatial and temporal dimensions. The 3D convolution is achieved by convolving a 3D kernel to the cube formed by stacking multiple contiguous frames together. In this way, the convolution feature maps in the convolution layer are connected to multiple continuous frames in the previous layer, capturing motion information. The function that gives the value at position (x, y, z) on the jth feature map in the ith layer is:

$$v_{ij}^{xyz} = \tanh(b_{ij} + \sum_{m} \sum_{p=0}^{P_i-1} \sum_{q=0}^{Q_i-1} \sum_{r=0}^{R_i-1} w_{ij}^{pqr} v_{(i-1)m}^{(x+p)(y+q)(z+r)}) \tag{2.4.1}$$

where $\tanh(\cdot)$ is the hyperbolic tangent function, b_{ij} is the bias for this feature map, m indexes over the set of feature maps in the (i-1)th layer connected to the current feature map, w_{pqr}^{ij} is the value at the position (p, q, r) of the kernel connected to the m_{th} feature map in the previous layer, and R_i, P_i and Q_i are the size of the 3D kernel along the temporal dimension, the height and the width of the kernel, respectively.

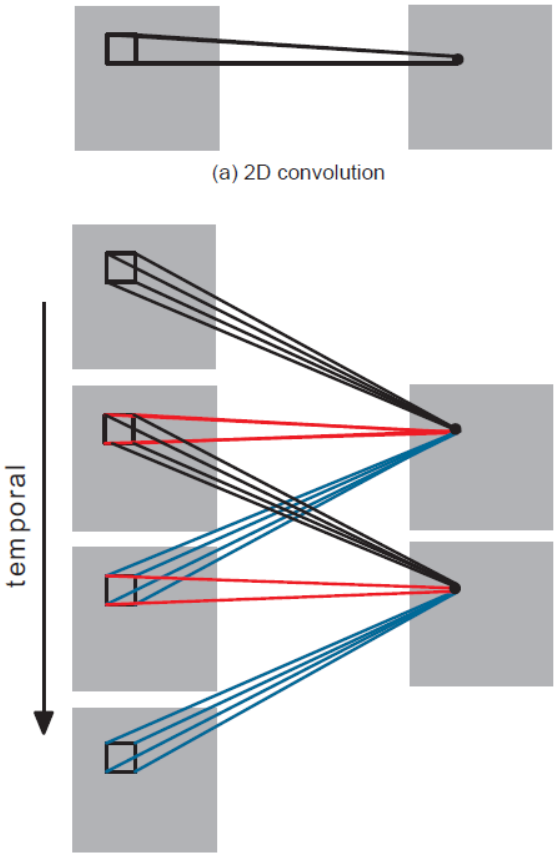


Figure 2.4.1: Comparison of 2D and 3D convolutions from [36]

2.4.2 C3D

In order to find a good 3D ConvNet architecture in [72], they only varied kernel temporal depth of the convolution layers while keeping all other common settings fixed, and experimented with two types of architectures: 1) homogeneous temporal depth: all convolution layers have the same kernel temporal depth; and 2) varying temporal depth: kernel temporal depth is changing across the layers. After training those networks they concluded that the homogeneous setting with convolution kernels of $3 \times 3 \times 3$ is the best option for 3D ConvNets, which is consistent with a similar finding in 2D ConvNets [65]. So, a 3D ConvNet was designed that has all its 3D convolution filters sized $3 \times 3 \times 3$ with stride $1 \times 1 \times 1$. In addition, the architecture of the network consisted of 8 convolution layers, 5 pooling layers, followed by two fully connected layers, and a softmax output layer. The network architecture is presented in Figure 2.4.2 and is called C3D. Also, all its 3D pooling layers are 2×2×2 with stride 2×2×2 except for pool1 which has kernel size of $1 \times 2 \times 2$ and stride $1 \times 2 \times 2$ with the intention of preserving the temporal information in the early phase. Finally, each fully connected layer has 4096 output units.

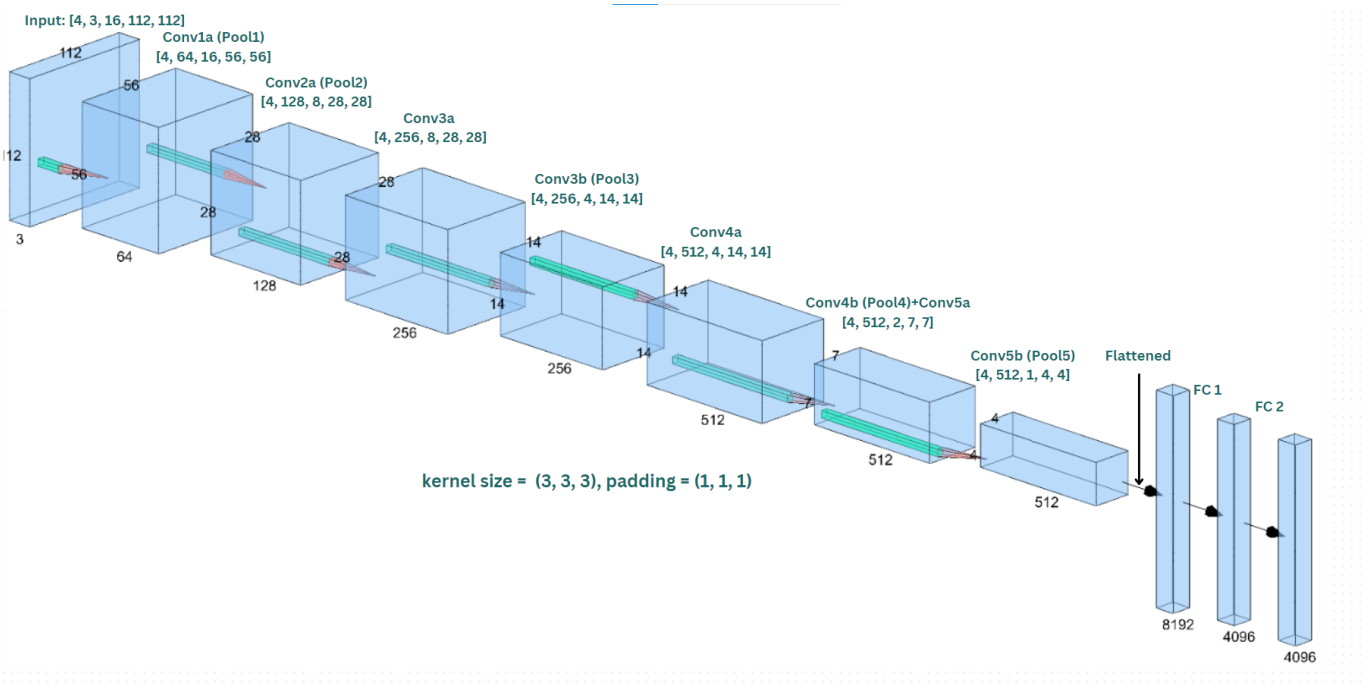


Figure 2.4.2: C3D architecture from the 2D point of view (spatial).

2.5 Recurrent Neural Networks

2.5.1 Architecture

Recurrent Neural Networks (RNNs) are a type of deep learning neural network designed to recognize patterns in sequences of data, such as time series, speech, text, video, and other sequential data. Unlike traditional feedforward neural networks, RNNs have connections that loop back on themselves, allowing them to maintain a "memory" of previous inputs, which is crucial for tasks where context or sequence matters.

The arcitecture of the RNNs is shown in Figure 2.5.1. Their main feature is their hidden state, which acts as a form of memory that captures information about what has been processed so far (the previous inputs). At each step in a sequence, the hidden state is updated based on both the current input and the previous hidden state. So when the RNNs make their prediction, they use both the current input and the stored memory to predict the next sequence. This makes RNNs capable of retaining information across time steps.

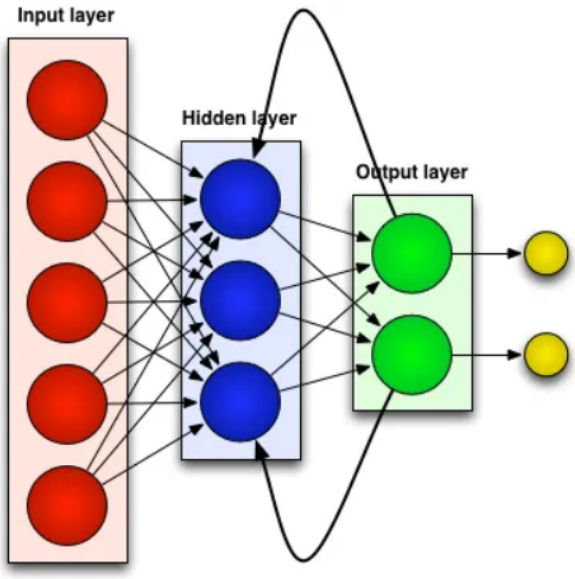


Figure 2.5.1: RNNs architecture, consisting of an additional hidden state which acts as a form of memory from [54].

2.5.2 Limitations

Even though the RNNs have been proven valuable tools for sequencial problems, such as natural language processing (NLP) applications, they still have plenty limitations:

1. Vanishing gradient

   The gradient describes the sensitivity of the error rate of the model that corresponds to the model's parameters. The gradient can be imagined as a slope, the larger the gradient, the steeper the slope, the more quickly the system can roll downhill to the finish line and complete its training. The vanishing gradient is a problem where the gradient values get too small, and their corresponding slopes so flat, during training, making the RNNs fail to learn effectively from the training data, resulting in underfitting.

2. Exploding gradient

   Exploding gradient on the other hand, occurs, when the gradient increases exponentially until the RNN becomes unstable. When gradients become infinitely large, the RNN behaves erratically, resulting in

performance issues such as overfitting. Overfitting is a phenomenon where the model can predict accurately with training data but can't do the same with real-world data.

3. Slow training time

   RNNs require massive computing power, memory space, and time to train as they need to store previous inputs and process them too during training time.

2.5.3 Long Short-term memory Networks (LSTMs)

Long short-term memory (LSTM) is an RNN variant, that enables the model to expand its memory capacity to accommodate a longer timeline. An RNN can only remember the immediate past input, so It can't use inputs from several previous sequences to improve its prediction, compared to the LSTM.

The LSTM architecture involves the memory cell, which is controlled by three gates: the input gate, the forget gate, and the output gate. These gates decide what information to add to, remove from, and output from the memory cell.

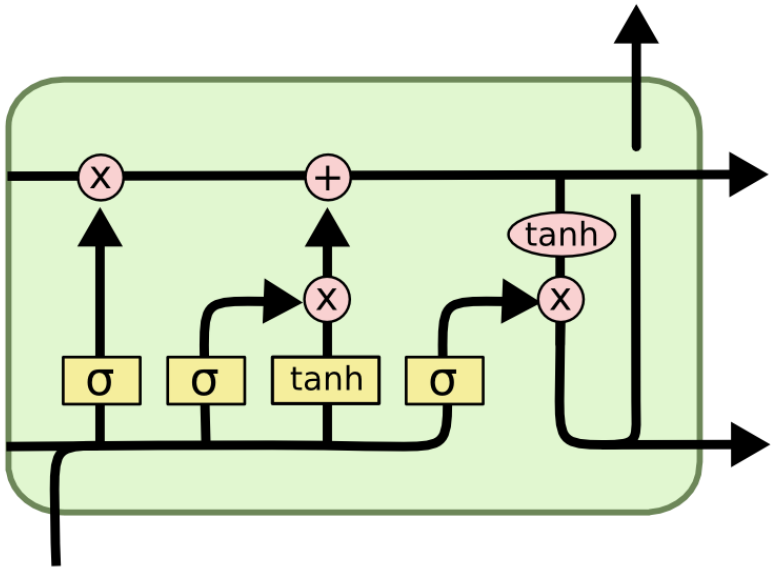


Figure 2.5.2: LSTM network architecture. To the left the forget gate, in the middle the input gate and to the right the output gate from [56].

The Forget Gate

The forget gate decides what information to discard from the cell state. It takes as input the previous hidden state $(h_t - 1)$ and the current input (x_t), and multiplies them with weight matrices followed by the addition of bias. Their results is then passed through an activation function, which produces an output of either 0 or 1 for each number in the cell state C_{t-1}. A value of 1 means "completely keep this" while a value of 0 means "completely forget this." The forget gate is defined by the following equation:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2.5.1}$$

where W_f is the weight matrix for the forget gate, b_f is the bias term, σ is the sigmoid activation function and $[h_{t-1}, x_t]$ denotes the concatenation of the current input and the previous hidden state.

The Input Gate

The input gate determines which values will be updated in the cell state. It consists of two parts: the input gate layer and a layer that creates a vector of new candidate values that could be added to the state. The input gate layer uses the sigmoid function to filter the values to be remembered similar to the forget gate using inputs h_{t-1} and x_t. The new candidates layer creates a vector using the tanh function, which gives an output from -1 to +1, that contains all the possible values from h_{t-1} and x_t. At last, the values of the vector and the regulated values are multiplied to obtain the useful information. The equation for the input gate is:

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{2.5.2}$$

$$\hat{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \tag{2.5.3}$$

Where the $\hat{C}_t$ represents the candidate vector amd tanh is the tanh activation function.

We multiply the previous state by f_t, disregarding the information we had previously chosen to ignore. Next, we include $i_t * C_t$. This represents the updated candidate values, adjusted for the amount that we chose to update each state value.

$$C_t = f_t \odot C_{t-1} + i_t \odot \hat{C}_t \tag{2.5.4}$$

Where $\odot$ denotes element-wise multiplication.0

The Output gate

The output gate decides what the next hidden state h_t should be, which will be used for both the next time step and the output of the current cell. First, a vector is generated by applying tanh function on the cell. Then, the information is regulated using the sigmoid function and filtered by the values to be remembered using inputs h_{t-1} and x_t. Finally, the values of the vector and the regulated values are multiplied to be used as output and as input to the next cell. The equation that describes the output gate is:

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{2.5.5}$$

Chapter 3

Related Work

3.1 Engagement detection

The children engagement problem has been approached with a wide variety of methods and techniques in order to benefit from its different aspects. Because of the problem's dynamic nature having both spatial and temporal aspects, these methods mostly use spatiotemporal neural networks, which mostly consist of two components, the first a network that is used for extracting spatial features from data, such as images or frames in a video, capturing local patterns and textures. The second one, deals with the temporal aspect of the data, capturing how the information changes over time. Some of the different networks used for this component might be Recurrent neural networks (RNNs) '[63], long short-term memory networks (LSTMs) [32], gated recurrent units (GRUs)[14], or more recently, temporal convolutional networks (TCNs) [5]. These models are designed to handle sequential data and maintain temporal dependencies. Finally, the spatial and temporal features produced by the two components are combined or fused effectively in various ways (3D Convolutional Networks, ConvLSTM Networks etc.)

3.1.1 Spatiotemporal Visual Cues

Regarding engagement detection, in [3], they propose a novel architecture built using the spatiotemporal convolutional block R(2+1)D for action recognition, which is made of spatiotemporal convolutions that perform a two dimensional convolution over the spatial plane followed by a one dimensional convolution along the time axis. These spatiotemporal convolutions are repeated twice and combined in a residual block. The network is consisted of five consecutive layers created from these spatiotemporal blocks. The spatiotemporal convolutional layers are followed by an adaptive pooling layer and finally fully connected layer. The weighted CE loss function is used to compute the network loss. Different experiments were conducted using raw RGB data, a fusion of RGB and depth data and also a fusion of raw RGB and optical flow data. In addition to those, the fusion of rgb and optical flow networks during both at a late and an early network stage was investigated, with late being before the last fully connected layer and at an early after the first of the five spatiotemporal convolutional layers. Finally, promising results were produced in all the datasets that the architecture was tested on, with the network that fused the raw Rgb and the optical flow at an early stage having the best combination of accuracy, f1-score and weighted precision compared to the others.

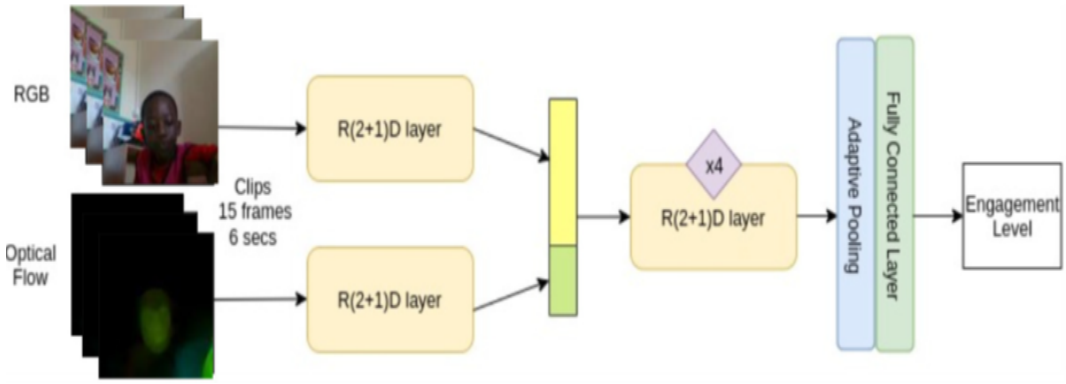


Figure 3.1.1: Network for engagement estimation using spatiotemporal visual cues, and fusing the raw RGB and optical flow data at an early stage. [3]

3.1.2 Resnet and TCN Hybrid Network

In [1], a novel end-to-end Residual Network (ResNet) and Temporal Convolutional Network (TCN) hybrid neural network architecture is created for students' engagement level detection in videos. In this architecture the 2D Resnet extracts the spatial features from a sequence of raw frames and then a TCN network analyzes the temporal changes in them to detect the output level of engagement. The TCN is chosen as it is superior in modeling sequences of larger length and retaining memory of history in comparison to generic recurrent architectures such as LSTMs and GRUs. Also, like it was mentioned before a combination of a spatial and a temporal model is chosen in this architecture too, as the engagement is a spatio-temporal affective state that takes place in consecutive frames of video over time. The architecture of the hybrid network is shown in Figure 3.1.2.

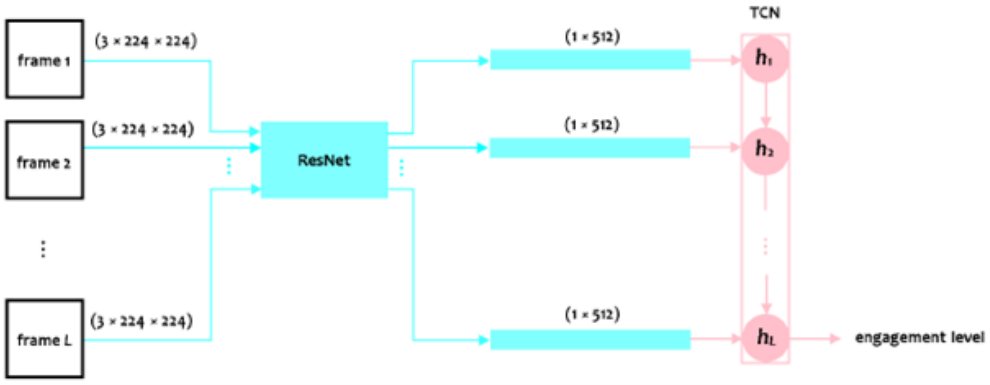
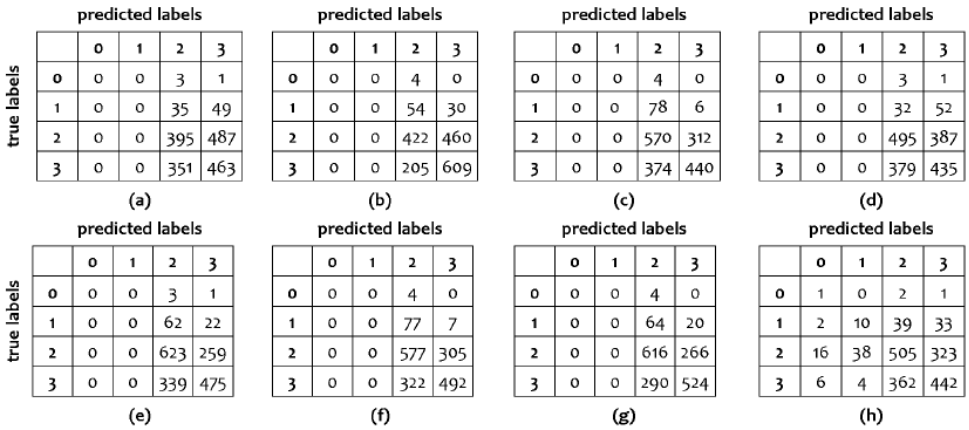


Figure 3.1.2: ResNet+TCN network architecture. First each frame is processed individually by the backbone and then all of their features temporally by the TCN module. [1]

In Figure 3.1.3 we can see the results produced in this study. A really important observation to be made here, is that a common problem in the existing student engagement detection datasets is the highly imbalanced distribution of engagement levels. That results in a challenge to detect the minority engagement level, which can be seen from the fact that even though the model (n) has the highest accuracy, in its confusion matrix (g) it can be seen that it makes no correct prediction in the 0 and 1 classes.

To tackle this problem a weight term is added to the loss function, and indeed more minority level samples are detected but at the cost of more false alarms (reducing the overall accuracy). Additionally, the prediction distribution improves a bit after a weighted sampling and cross entropy loss is used, but with the same cost of worsening the accuracy.



(a)	0	1	2	3
0	0	0	3	1
1	0	0	35	49
2	0	0	395	487
3	0	0	351	463

(b)	0	1	2	3
0	0	0	4	0
1	0	0	54	30
2	0	0	422	460
3	0	0	205	609

(c)	0	1	2	3
0	0	0	4	0
1	0	0	78	6
2	0	0	570	312
3	0	0	374	440

(d)	0	1	2	3
0	0	0	3	1
1	0	0	32	52
2	0	0	495	387
3	0	0	379	435

(e)	0	1	2	3
0	0	0	3	1
1	0	0	62	22
2	0	0	623	259
3	0	0	339	475

(f)	0	1	2	3
0	0	0	4	0
1	0	0	77	7
2	0	0	577	305
3	0	0	322	492

(g)	0	1	2	3
0	0	0	4	0
1	0	0	64	20
2	0	0	616	266
3	0	0	290	524

(h)	0	1	2	3
0	1	0	2	1
1	2	10	39	33
2	16	38	505	323
3	6	4	362	442

Figure 3.1.3: Engagement-level confusion matrices of different methods on the DAiSEE dataset [28], (a) C3D feature extraction [72], (b) C3D fine tuning [72], (c) C3D + LSTM [61], (d) C3D averaging + LSTM [61], (e) ResNet + LSTM (proposed), (f) C3D + TCN (proposed), (g) ResNet + TCN (proposed), (h) ResNet + TCN with weighted sampling and weighted loss (proposed).

3.1.3 Multi-dimensional feature fusion (Mediapipe, VGG16 and ResNet101)

In [80], an Online Facial-Based Engagement Recognition System framework is created, which integrates three major indicators -human eye movement, human posture, and student expression— at the feature level to accurately assess student engagement in the classroom.

This framework is consisted of three parts. The image preprocessing module, which transforms video sequences to a series of consecutive frame images highlighting the most representative facial features. Secondly, the feature extraction module, which adopts the Mediapipe, VGG16 and ResNet101 structs extracting eye-mouth behaviour, facial expressions and head pose estimation respectively. The third part is the feature fusion module, which leverages multiple dimensions of data by using a Backpropagation Neural Network (BPNN) having an input layer of 12 nodes (7 emotions, roll, pitch, yaw, mouth aspect ratio and ear aspect ratio) and an output of three engagement levels: Not Engaged, Nominally Engaged, and Very Engaged.

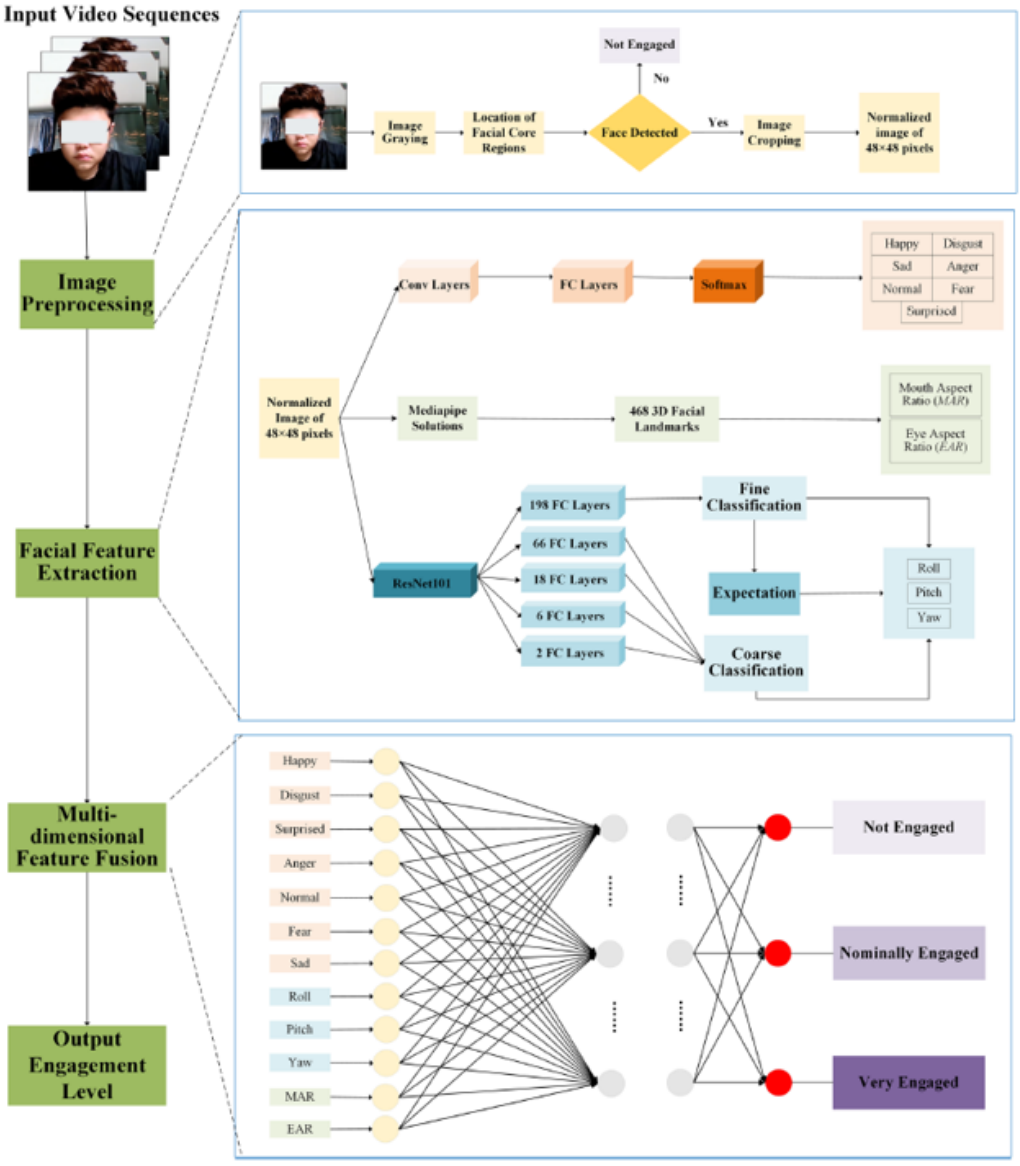


Figure 3.1.4: Overview of framework architecture to classify ASD and TD based on learned features. [80]

3.1.4 CNN and Recurrent Network

In [19] a deep learning approach for the estimation of human engagement from video sequences, is proposed that is consisted of two main modules: a convolutional module which extracts frame-wise image features and a recurrent module that aggregates the frame features over a time to produce a temporal feature vector of the scene.

The convolutional module is a ResNetXt-50 Convolutional Neural Network (CNN) [81] that is pre-trained on the ImageNet dataset [20]. The frame features are obtained from the activation of the last fully connected layer of the CNN, with dimension 2048, before the softmax layer. The recurrent module on the other hand, is a single layer Long-Short Term Memory (LSTM) [31] with 2048 units followed by a Fully Connected (FC) layer of size 2048×1. The LSTM takes in input a sequence of w frame features coming from the convolutional module and produces, in turn, a feature vector that represents the entire frame sequence, to capture the temporal behavior of humans within the time window w. The temporal features are passed through the FC layer with a sigmoid activation function at the end to produce the final gaze estimation values. During the experiments the CNN layer is fixed, while the recurrent module is trained to predict engagement values from the provided annotations.

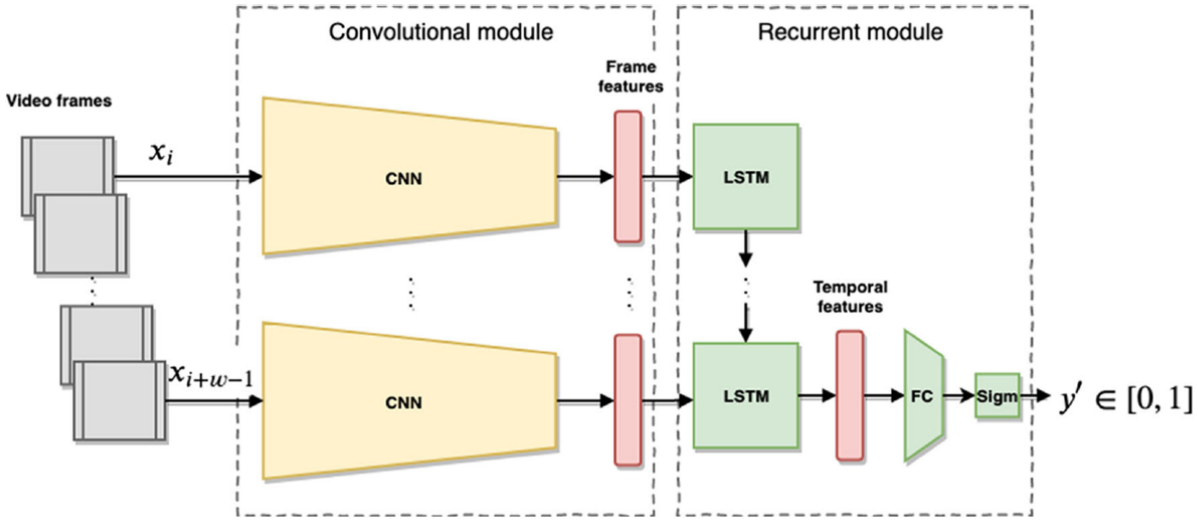


Figure 3.1.5: Overview of the proposed architecture from [19]. First each frame is processed spatially by the CNN and then their features are processed temporally by the LSTM.

The results in this study showed that this two-stage deep-learning architecture, trained to the TOGURO dataset is a suitable computational regression model to capture the inherent human interpretation of engagement. Not only that, but it can also be successfully applied in a completely different scenario, as it is generic enough, the UE-HRI dataset [8], showing the applicability of the model also in different environments, on a different robot with a different camera, and with different tasks and people.

3.1.5 VGG Face CNN

The Engagement recognition framework constructed in [84], tries to overcome the problem of the small and not really diverse children engagement datasets, by utilizing a pretrained model, VGG Face CNN [60], which will help the network learn general features related to face identification problem.

Firstly, for the training procedure they use only facial images in order to prevent the model from focusing on the background information and as result lower its accuracy on test datasets with different backgrounds. Then, T facial images are fed into the VGG Face network one after another, and their low level features are extracted at the last convolutional layer. After that, these low-level features are passed into a temporal dynamics module comprising of a mixture of convolutional layers and pooling layers to produce a high-level feature. Finally, this high-level feature is passed through a fully connected layer, a ReLU layer, a dropout layer, an another fully connected layer and a softmax layer with K classes to get the final output.

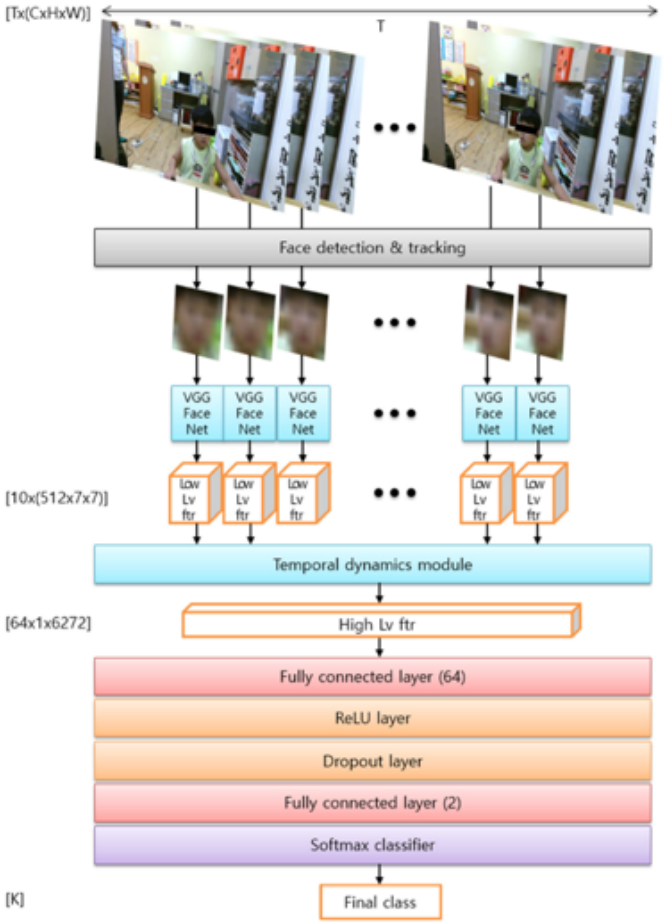


Figure 3.1.6: Overview of the engagement recognition framework using VGG Face.[60]

3.2 Gaze tracking

The ability of understanding of where a person is looking, known as gaze tracking, provides invaluable insights across various domains of research such as human-computer interaction, assistive technologies, psychology and marketing. The architectures designed to estimate the gaze direction are often divided into two categories, the appearance based methods and the model based methoda. The appearance-based ones are using image features and machine learning models to estimate gaze direction, while the model-based ones built on geometric models of the eye to calculate gaze direction using eye landmarks. In this study, we are going to focus on the appearance based ones, which recently have greatly improved the performance tracking systems, due to the advancements of deep learning and computer vision architectures.

3.2.1 Gaze360 model

In [39] a model that uses bidirectional Long Short-Term Memory capsules (LSTM) is proposed in order to tackle the problem of gaze estimation. In this model, the output that is produced for one element is dependent on both past and future inputs, as a sequence of 7 frames is used to predict the gaze of the central frame. Each one of the 7 frames which is the input of the model is a head crop of the human, whose gaze is going to be tracked, and at the first part of the model, each one is individually processed by a convolutional neural network (backbone). The output of the backbone are the high-level features with dimensionality 256, that are then used as input to the bidirectional LSTMs, that consist of two layers which digest the sequence within forward and backward vectors. Finally, these vectors are concatenated and passed through a fully connected layer to produce two outputs: the gaze prediction and an error quantile estimation. The gaze prediction is produced in spherical coordinates and the error quantile estimation (σ) is a measure of the difference between the 10 and 90 quantiles and the expected value.

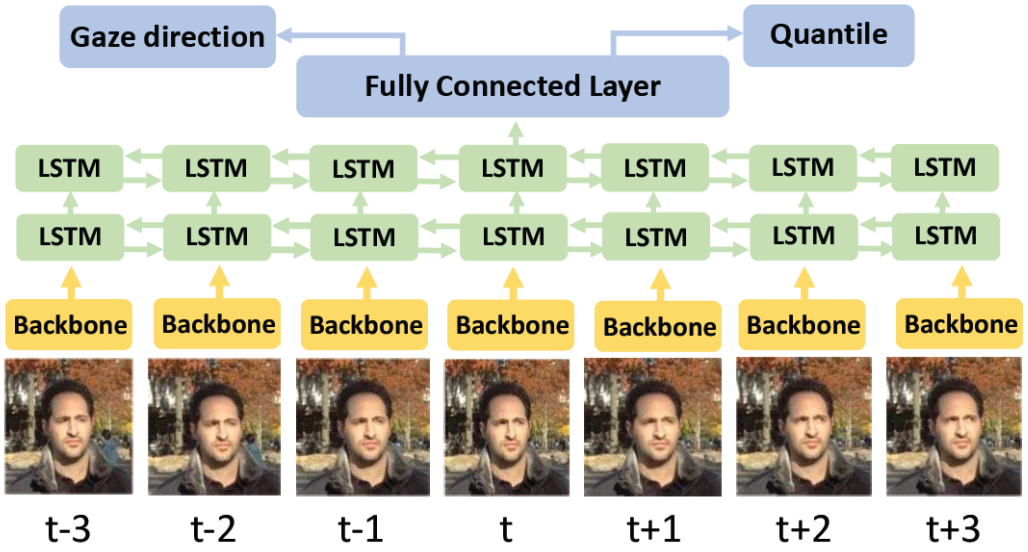


Figure 3.2.1: The Gaze360 model architecture using a backbone to process the frames spatially and a LSTM model for their features' temporal processing.[39]

3.2.2 Recurrent CNN architecture

The proposed model in [57], is divided in 3 modules: (1) Individual, (2) Fusion, and (3) Temporal. Firstly, the Individual module learns features from each appearance cue separately. It consists of a two-stream CNN, one devoted to the normalized face image stream and the other to the joint normalized eyes image. Each stream of the Individual module is based on the VGG-16 deep network [60], consisting of 13 convolutional layers, 5 max pooling layers, and 1 fully connected (FC) layer with Rectified Linear Unit (ReLU) activations. Next, the Fusion module combines the extracted features of each appearance stream in a single vector along with the normalized landmark coordinates. Then, it learns a joint representation between modalities in a late-fusion fashion. Both Individual and Fusion modules, further referred to as Static model, are applied to each frame of the sequence. Finally, the resulting feature vectors of each frame are input to the Temporal module based on a many-to-one recurrent network. This module leverages sequential information to predict the normalized 2D gaze angles of the last frame of the sequence using a linear regression layer added on top of it.

For the training, the input sequence is composed of s = 4 consecutive frames, whose gaze direction target is the gaze direction of the last frame. In addition, the network is trained in a stage-wise fashion, starting with training the Static model and the final regression layer end-to-end on each individual frame of the training data first. Also, the convolutional blocks are pre-trained with the VGG-Face dataset [60], whereas the FCs are trained from scratch. Then, during the final training procedure the features of each frame of the sequence are extracted from a frozen Individual module, then they fine-tune the Fusion layers, and train both, the Temporal module and a new final regression layer from scratch.

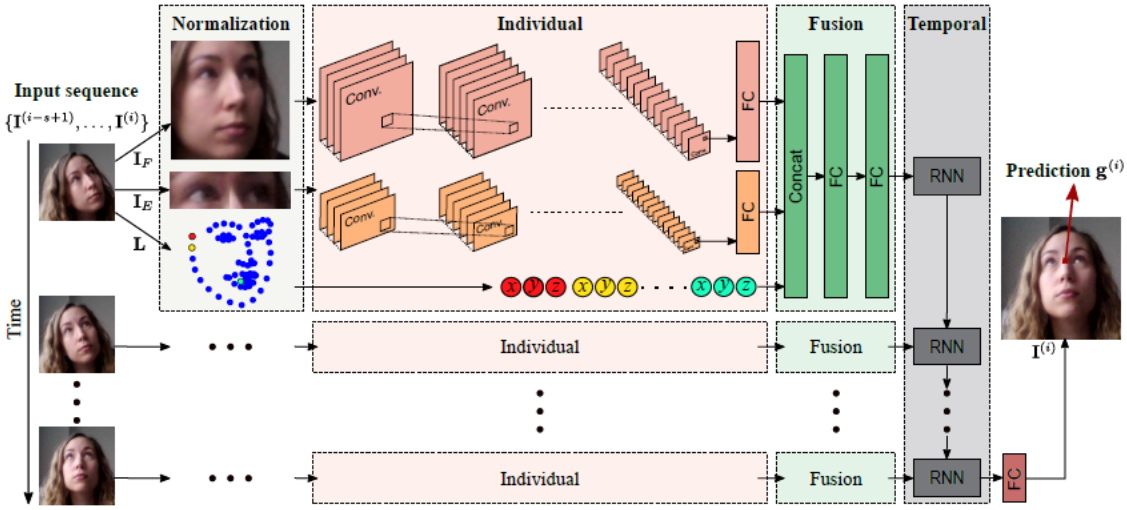


Figure 3.2.2: The Recurrent CNN proposed network.[57]

3.2.3 Multi-Clue Gaze (MCGaze)

Multi-Clue Gaze [26] is a new method proposed to facilitate video gaze estimation by capturing spatial-temporal interaction context among head, face, and eye in an end-to-end query-based learning way. The main idea of the method is shown in 3.2.3. For each input video clip ($I \in R^{Tx3xHxW}$), the first step is to extract its per-frame features, by applying a backbone, in order to form a video feature tensor ($F \in R^{TxCxH'xW'}$. The backbone used is the ResNet-50-FPN [30], pre-trained on ImageNet-1K [20] and with the iteration time N set to 4. After that, the extracted features are fed into our query-based architecture, which iterates N times and consists of two main components: the spatial-temporal query interaction and the task-specific heads (i.e., clue localization head and gaze fusion head). In each iteration, the queries for the head, face, and eye clue are updated, and the clue localization head predicts the clue region of the head, face, and eye. At the same time, the gaze fusion head determines the direction of the human gaze from the head, face, and eye clue. The gaze predicted by the last iteration is used as the output of the model.

In order to capture the subject's corresponding clue regions and gaze representations this approach applies multi-clue queries $q_{clue} \in R^{TC}$, clue $\in$ head, face, eye. Each query comprises T embeddings with a feature dimension of C, with each embedding generally focusing on the feature representation of the corresponding frame. Additionally, there are proposal boxes $p_{clue} \in R^{T4}$ that correspond to each query, which indicate the locations of the subject's head, face, and eye in the feature map. The parameters of both q_{clue} and p_{clue} are learnable. For each complete forward propagation, they will be updated in an iterative way to achieve effective extraction of target clues and gaze representations from it.

Additionally, local-global spatial-temporal modeling is of great importance for the video task [79], [33], so in this studied the also designed specific queries for the three key clues of their task. Specifically, they used a spatial-temporal queries interaction module [82] to better localize the hierarchical clues and build effective information exchange for robust gaze representations. In this module, the spatial interaction among head, face, and eye query within the same frame is enabled by the usage of a spatial multi-head self-attention layer [73], leading the queries to be of both global and local spatial perspectives for gaze characterization. In addition, they apply a self-attention layer to enable temporal interaction for each query along the temporal dimension, which promotes sequential modeling of distinctive features, such as pose variation and eye movement, and facilitates temporal consistency, leading to robust clue localization and gaze estimation.

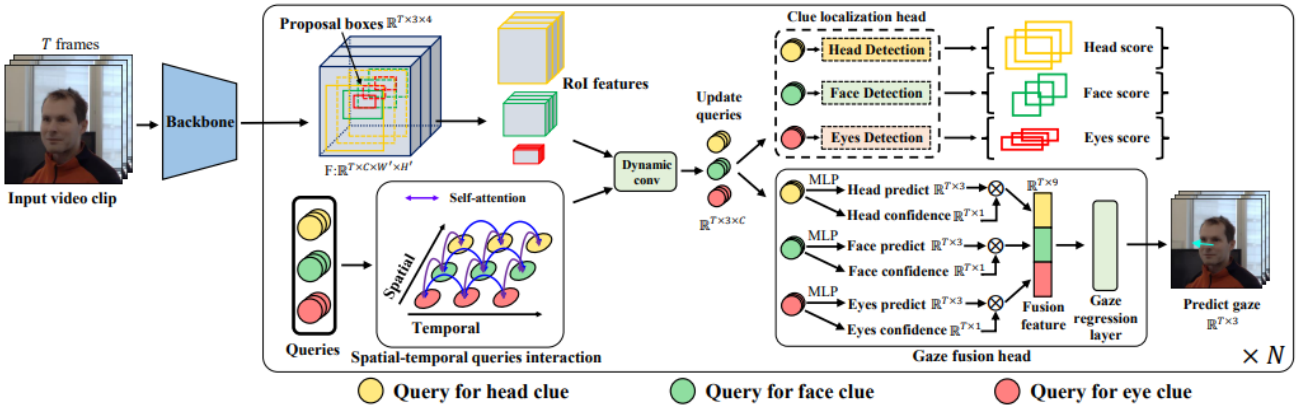


Figure 3.2.3: The proposed MCGaze architecture.[26]

3.2.4 CrossGaze

Unlike previous approaches, CrossGaze [12] does not require a specialized architecture, utilizing already established models that are integrated in one architecture that adapts to the task of 3D gaze estimation. An overview of the proposed architecture is shown in 3.2.3. The input to the model is an in-the-wild image, from which bounding boxes of the face and eye regions are derived using a face detection model. Then the face image is fed into an encoder that captures global gaze features, while the eye images are given as input to the eye encoder which extracts local features. To obtain an eye-informed gaze prediction, the extracted features of the 3 images are combined through a cross-attention module. The resulting combination of features is averaged and processed by a linear layer that outputs the 3D gaze direction. Compared to other studies that use polar coordinates as outputs and labels for training, in this one they found out that there is no improvement in comparison to the use of normalized (x, y, z) as both outputs and labels. For their training procedure they utilized the cosine loss which represents the complement of the cosine similarity and also adapted the RandAugment algorithm [16] as training time augmentation to increase the robustness of the gaze estimation models.

In order to decide which backbone to use they experimented with different CNN backbones such as EfficientNet [52], ConvNeXt [49], Inception ResNet [68] and an attention-based backbone (i.e. Swin Transformer [48]). From their results they concluded that the Inception ResNet architecture obtains the lowest mean angular error with a value of 10.91 for random initialization and 10.82 for ImageNet pretrained weights. These results demonstrate the suitability of the Inception ResNet architecture for processing the face: the capability to process the input at multiple scales in every layer helps the model focus both on the eyes region and on the periocular area for the gaze prediction. In addition to that, they experimented with different pretrained weights for the Inception ResNet architecture testing it on ImageNet [20], Casia-WebFace [83], and VGGFace2 [10]. The model performed better on the larger dataset (VGGFace2) than the model pretrained on Casia-WebFace due to the increased amount of training data. While Casia-WebFace contains approximately 500K images, VGGFace2 contains 3.3M images, resulting in a 10.02 in mean angular error compared to 10.26.

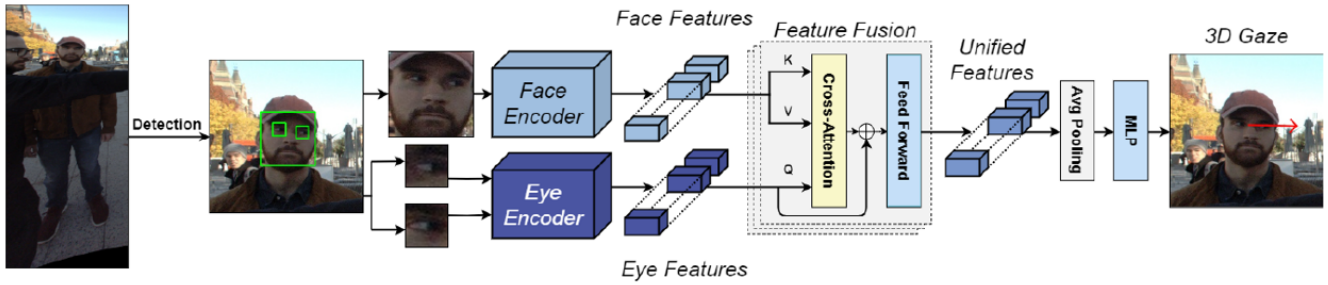


Figure 3.2.4: Overview of the CrossGaze architecture. Two encoder a face and an eye are used to produce the face and eye features respectively, which are then fused to be processed for the final gaze estimations. [12]

3.2.5 Dilated-Convolutions

In [13] a new architecture is proposed, called Multi-region Dilated-Net, which is inspired by iTracker [42], but takes advantage of the Dilated convolutions in order ti improve previous results. This architecture takes an image of the face and images of both eyes as input and feeds them to a face network and two eye networks, respectively.

The face network is a VGG-like network [65], consisting of four blocks of stacked convolutional layers (Conv) followed by a max-pooling layer, as well as two fully connected layers (FC). The weights of the first seven convolutional layers are transferred from the first seven layers of VGG-16 pre-trained on the ImageNet dataset. Additionally, a 1×1 convolutional network (Network in Network [47]) is inserted after the last transferred layer to reduce the number of channels.

The two eye networks have identical architecture and share the same parameters in all convolutional and dilated-convolutional layers. The network starts with four convolutional layers with a max-pooling layer in the middle, followed by a 1 × 1 convolutional layer, four dilated-convolutional layers (dilated-Conv) and one fully connected layer. The weights of the first four convolutional layers are also transferred from the first four layers of VGG-16 pre-trained on the ImageNet dataset. Also, the dilation rates of the layers are (2, 2), (3, 3), (4, 5) and (5, 11), respectively. These dilation rates are designed according to the hybrid dilated convolution (HDC) in [76] so that the RF of each layer covers a square region without any holes in it.

In order to combine the features of the three different networks, the output of their final fully connected layers are concatenated and fed to FC-2 and in the end the output layer. In this study, the also tried to show the improvement achieved by dilated convolutions, by using a deep CNN that has similar architecture but no dilated-convolutions. It replaces the four dilated-convolutional layers by four convolutional layers and three max pooling layers located at the beginning, in the middle and at the end of the four convolutional layers, respectively. The size of the final feature maps is the same as the one in Dilated-Nets, and this CNN has the same number of parameters as the corresponding Dilated-Net. The results showed as that indeed the Dilated-Net can estimate the gaze better in both the Columbia Gaze [66] and the MPII Gaze [85] datasets, as the use of dilated-convolutions allows the network to extract high level features at high resolution from eye images so that the network can capture small variability.

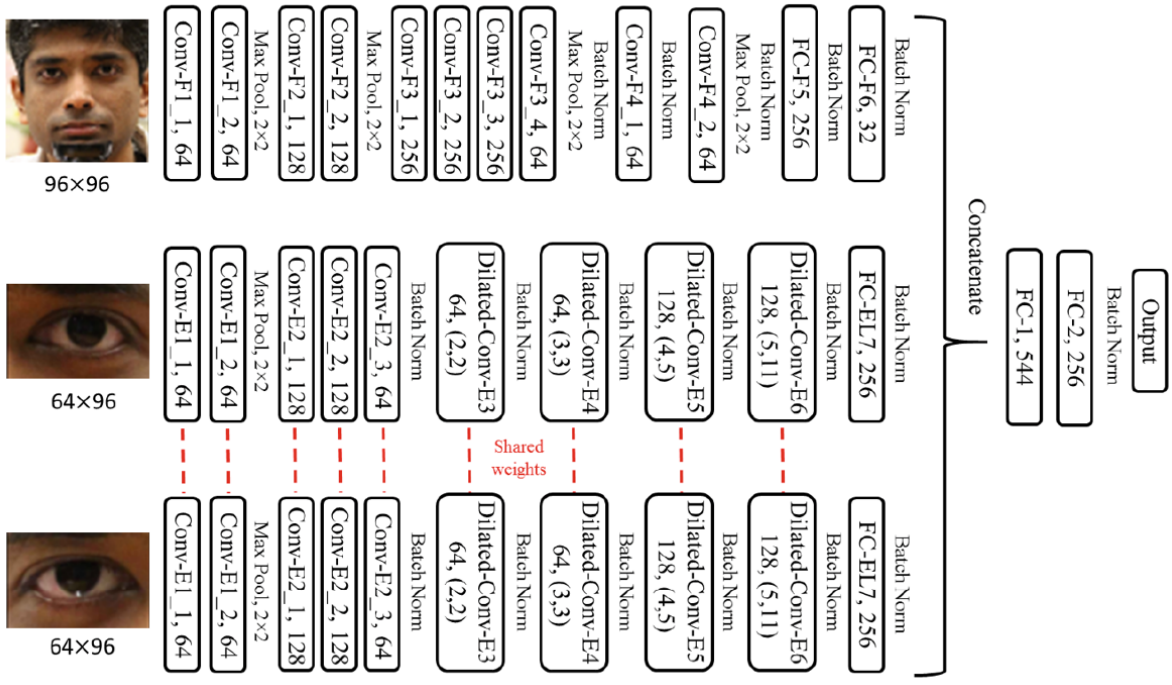


Figure 3.2.5: Overview of the multi-region Dilated-Net architecture, consisting of 2 networks one for the face images and one for they eye images. [13]

3.3 Autism Detection

Autism detection refers to the process of identifying signs and symptoms indicative of Autism Spectrum Disorder (ASD) in individuals, particularly in young children. This process involves a combination of behavioral assessments, developmental screenings, and clinical evaluations designed to observe social communication skills, repetitive behaviors, and sensory sensitivities. Traditional methods of autism detection often rely on standardized questionnaires and observational tools administered by healthcare professionals, such as pediatricians, psychologists, and developmental specialists. In recent years, advancements in technology have introduced new ways to detect autism by analyzing atypical patterns in eye movement, facial expressions, and interaction behaviors. In this way, methods such as gaze tracking and engagement recognition can be valuable in detecting autism.

3.3.1 Based on Gaze-Following

A novel deep neural network (DNN) model, which extracts discriminative features and classifies ASD and TD children on single images, is proposed in [23]. The data used in this study are produced by an eye tracking experiment where images from the GazeFollow dataset were showed to the children and their visual attention maps were collected.

In order to achieve their final goal of classifying ASD children, firstly the DoF density maps are predicted, using a SAM-ResNet [15], which is pretrained on SALICON [37] dataset, to extract coarse feature maps and then a LSTM network to process saliency features recurrently, to enhance the prediction. In addition, the locations of eyes and gaze points in images are highlighted and smoothened with a series of Gaussian kernels. Then, these Gaze-following prior maps and feature maps learned by the LSTM are combined in a fusion layer to produce the features (512×1×1 dimensional vectors extracted from each fixation location). The first 12 fixation points for each subject in each image are preserved ([12, 512, 1, 1] per person per image). After that, they are flattened to (6144,1) and fed to two fully connected (FC) layers with units of 1024 and 128 respectively (dropout ratio of 30% for the first) to finally produce the desired prediction.

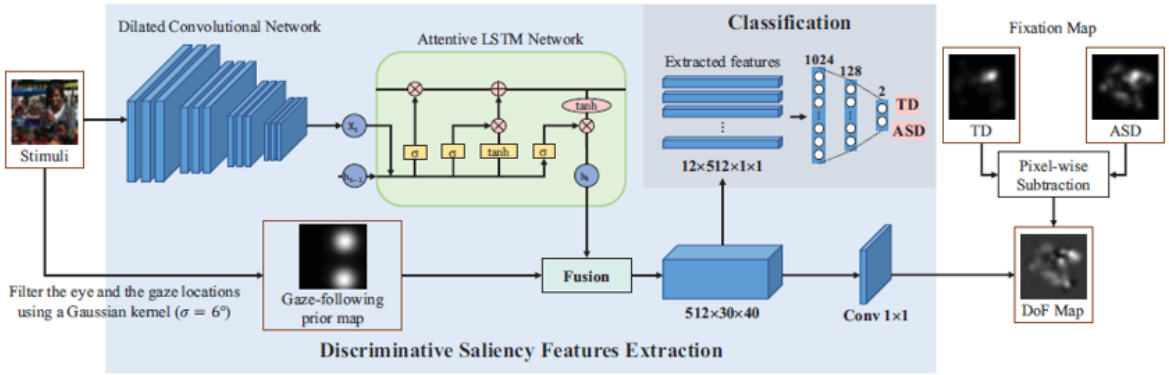


Figure 3.3.1: Overview of framework architecture to classify ASD and TD based on learned features from [23].

3.3.2 Machine learning approach to ASD diagnosis based on identifying specific behaviors from videos

In [78], they first introduce a large-scale video dataset of children's social interaction with manual labeling of behaviors that are clinically significant for ASD diagnosis. Then they propose a machine learning framework which consists of two stages: behavior detection and ASD prediction based on statistical features of behaviors. For behavior detection which consists of four different categories, look face, smile, look object and vocal, they propose two baseline deep-learning techniques, one based on end-to-end training on raw video frames, while the other relies on facial features of the child subject. They achieve significant performance, even though the specific dataset is challenging due to the fact that the child's face is often in a non-frontal position due to the camera position. Moreover, face occlusions by toy objects or head movement frequently occur and lead to failed face detections. In addition, they achieve low sensitivity for the vocal task, which indicates that this model is not well suited for detection of vocalization task since a frame based approach cannot make use of the motion of lips. For ASD prediction, they investigated feature selection, class rebalancing, and neural network classifiers to link the statistics of behaviors to ASD diagnosis. The combined rebalancing scheme produces the overall best results based on the F1 scores and accuracy.

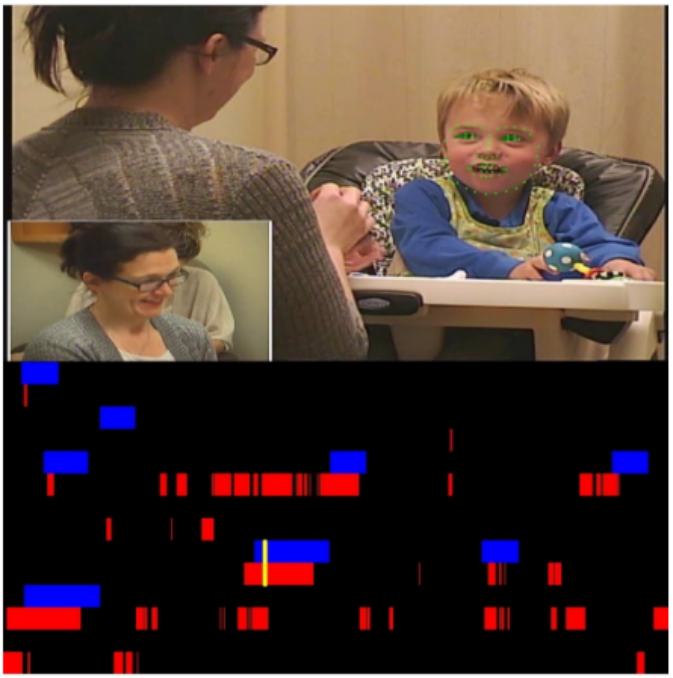


Figure 3.3.2: A video frame along with the facial keypoints and gaze (above) and the snapshot of ground truth (blue) vs predicted labels (red) for look face on a test video (below), from [78].

Chapter 4

Experiments

4.1 Datasets

4.1.1 Datasets used in engagement

- Pinsoro Dataset

  The Pinsoro Dataset [44] is a dataset specifically designed to capture the social interaction of children, while executing tasks. The understanding of these social interactions is really important in the field of Human-Robot Interaction (HRI), especially in areas where longer-term interactions are taking place, such as healthcare and education. In these areas, it is important for the robot to be able to recognise and interpret the social behaviours of children given visual or audio cues. For the experiments in our study, we use video recordings of this dataset in order to find an accurate and robust way of recognising the engagement of the children, while executing small tasks taking place around a large interactive table. Specifically, we select 25 videos of children interacting with robots on the other side of the table, as the pinsoro dataset also contains and child-child interactions. The interactions on the videos are annotated on three different axes: task engagement, social engagement and social attitude. For this study, we experiment only on the task engagement axis, which is split into four different engagement levels: no-play, adult-seeking, aimless and goal-oriented. Due to the fact that the adult-seeking engagement level is extremely rare we exclude it from our study.

Figure 4.1.1: Pinsoro dataset [45] frame examples.

- ASD Games Dataset

  The ASD Games Dataset [4] is a dataset collected during BabyRobot project [21], that consists of videos in which the children that participate face autism spectrum disorder (mean age 10.6 years old). In these videos, each child participating, plays four different games with two robots: Show me the Gesture, Express the Feeling, Pantomime and Guess the Object. Each child stood in front of the robots and interacted freely with them, while moving in the room as they wanted and it is classified on three levels of engagement: disengaged, passive attention (or act with no attention) and engaged.

4.1.2 Datasets used in gaze tracking

- WiDo Dataset

  For this study we will refer to this dataset as WiDo [59, 58], as it is a richly populated dataset, consisting of 17 videos of children with Williams and Down disorders (WiDo), but also normally developing ones, while interacting with their mothers. We select this dataset containing children with Williams and Down disorders, even though we have talked extensively about children with ASD, as these disorders lead to children having similar behaviours in their visual responses with the ones with ASD, during their interactions with their mothers. So, this kind of data are really important for the development of our algorithms. The WiDo dataset contains plenty of information for different computer vision tasks, such as Emotion, Gaze tracking, Speaker recognition, Play type, Action purpose among others. In each video, there is a child with its mother, freely interacting among each other and with different objects and toys in their house, a place where they feel really familiar, making it challenging in developing computer vision algorithms that process and model their interactions, as there are plenty of cases where their faces and bodies are partially occluded. In our study, we are going to focus on the task of gaze tracking, for which the dataset contains 6 different categories: (i) Not looking somewhere exactly or the camera, (ii) Looking at an item that only the child is using but not the mother, (iii) Looking at an item that is used in the interaction with the mother, (iv) Looking at some part of the mother's body, (v) Looking at the mother's face but not on the eyes and (vi) Looking at the mother in the eyes.

- Gaze360 Dataset

  The Gaze360 Dataset [39] consists of 238 subjects in indoor and outdoor environments with labelled 3D gaze across a wide range of head poses and distances. It is the largest publicly available dataset of its kind by both subject and variety, made possible by a simple and efficient collection method. In Figure 4.1.2 the diversity of the dataset in environment, illumination, age, sex, ethnicity, head pose and gaze direction is shown.

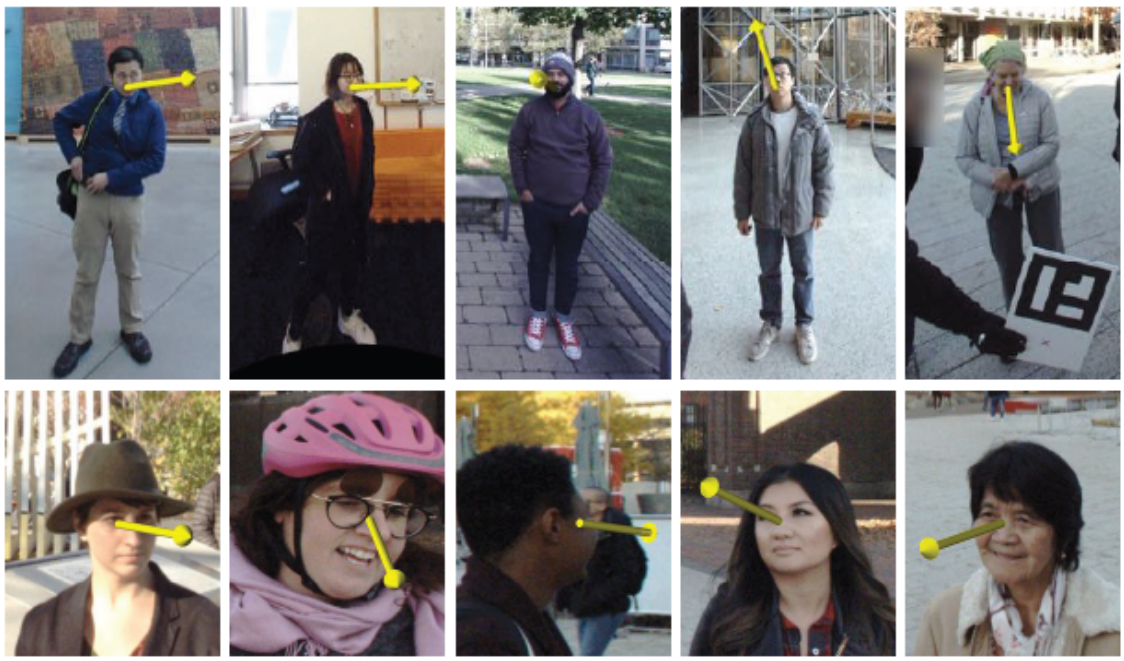


Figure 4.1.2: Gaze360 dataset samples from [39].

4.2 Proposed Methods

4.2.1 Engagement Estimation

TSN Architecture

The Temporal Segment Network architecture alters the way that video samples are being processed by the models. Instead of just providing as input to the model a sequence of continuous frames of a video sample or a number of frames sampled throughout the whole video and the model predicting its class, in the TSN architecture the networks operate on sequences of short snippets that are sparsely sampled from the entire video. Then the outputs for each short snippet are produced by the models, which are in the end combined using a segmental consensus function to obtain a consensus of class hypothesis among them. Based on this consensus, the prediction function predicts the probability of each class for the whole video sample.

Model Architecture using TSN

The proposed model architecture that utilizes the Temporal Segment Network framework is shown in Figure 4.2.1. The number of segments value affects the amount of continuous groups of frames that are going to be selected from each 10 sec video clip sample, while the frames per segment value, affects the length, in frames, that each one of those groups is going to be. We can see that according to the TSN architecture the dataloader will select the specified number of segments, throughout the video sample, each consisting of the specified number of frames, and feed the total frames as the backbone model's input. From there the backbone, for which different kinds of models are tested, will produce its output features for the frames of each segment, and those will be directed to the TSN head. There the TSN head will produce the final classification output, through a consensus mechanism being applied on the frame features.

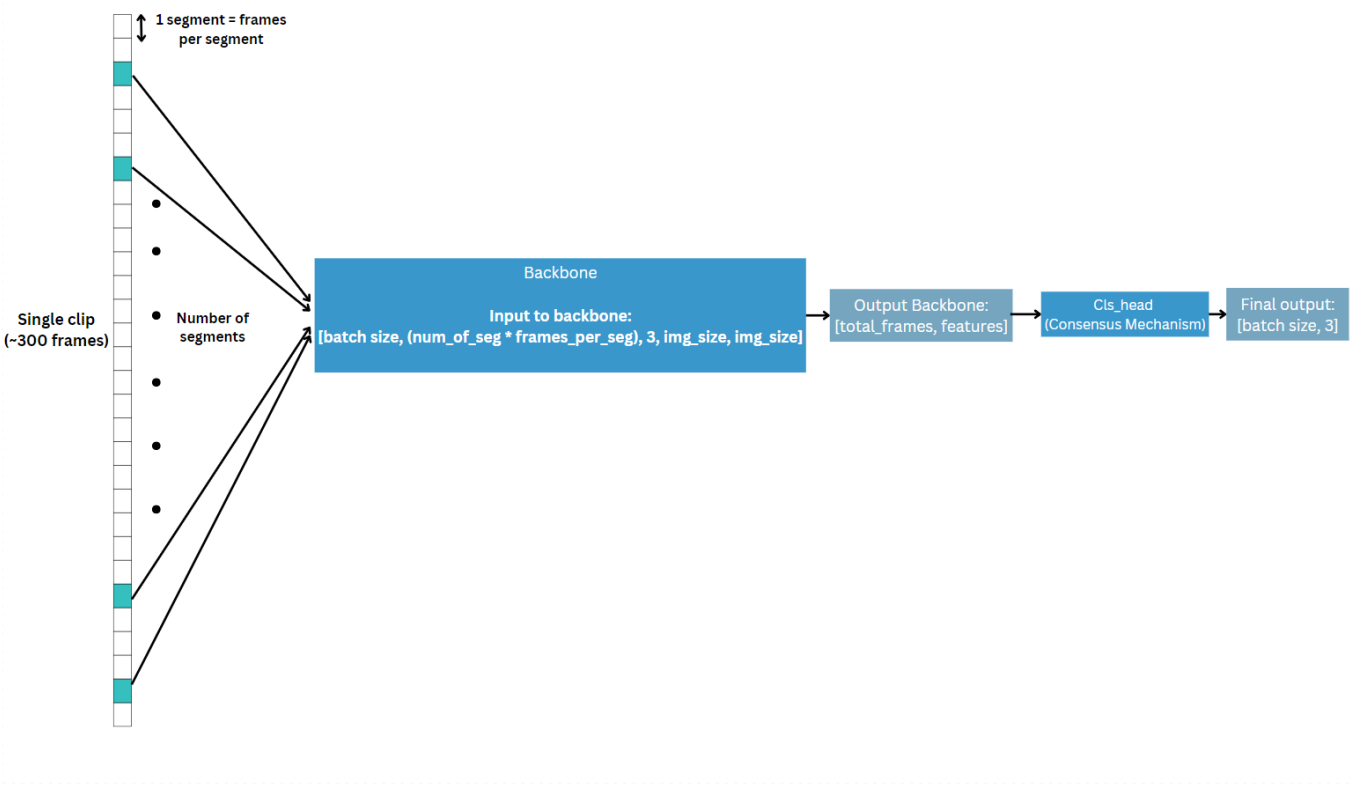


Figure 4.2.1: Model architecture that is proposed for the engagement estimation problem, using the TSN method.

4.2.2 Gaze Estimation

Like we saw in numerous studies that approach the gaze tracking problem, in order for the networks to better tackle the problem and produce better estimates, they need to be fed with continuous face crops of the subjects as the inputs, and not the whole frames [39, 26, 12]. So, our inputs to the model are going to be every time 7 continuous face crops of the child in each video, shown in Figure 4.2.2. Our model architecture is based mainly on the Gaze360 model proposed in [39], which is consisted of three main parts. Fristly, it takes each input sample consisted of 7 frames and preprocesses each one of the frames using a Resnet backbone model. There the local features for each frame are going to be extracted, and then are going to fed into the second part a LSTM model, which processes the derived features in the temporal dimension. From there, the final features are going to be produced for all the input frames. From the total features only the ones corresponding to the mid frame (4th one) are being kept. Those are fed into the third part a final fully connected layer which produces the gaze direction vectors. Then in order to be able to use this model in our problem, where we don't just need the gaze direction, but we also want to make the gaze estimation of the category to which the child is looking we had to create a check module. This check module takes these gaze direction vectors produced by the previous step and uses them to check if the gaze direction of the child crosses any of the bounding boxes of interest in our altered problem, by creating a line with its gradient being derived from the dx and dy coordinates (of the gaze direction) and the origin (0, 0) being the starting point of the child's gaze. If the gaze direction crosses any of the bounding boxes, then the prediction is assigned to the corresponding box class, else it is assigned to the "elsewhere" class.

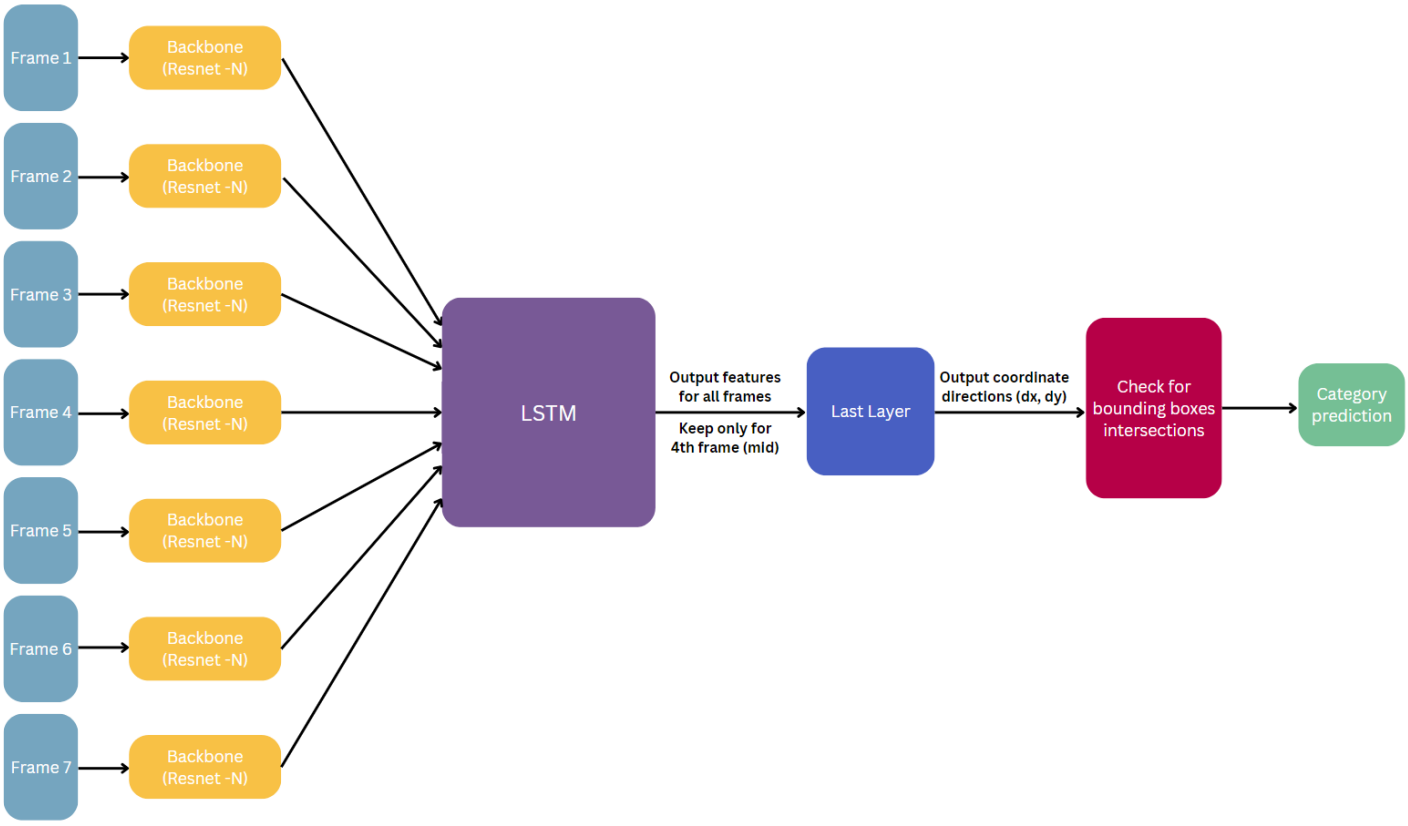


Figure 4.2.2: Implemented architecture for Gaze prediction classification where the pretrained model proposed in [39] is used with a check module added in the end.

4.2.3 Integration of Gaze tracking data into Engagement Estimation

Having studied and tackled both the engagement recognition and gaze tracking tasks separately, we can't help but wonder how would the performance on the engagement problem alter, if we integrate the gaze tracking implementation into it. For this reason, we create a new network architecture that combines the previous two models used at the engagement and gaze tracking phases, while using the TSN method.

The combined engagement plus gaze model is shown in Figure 4.2.3. As we can see, just like in the engagement problem, first according to the TSN implementation, the new dataloader that we create, selects a specific number of segments throughout the clip (sample) and for each segment an amount of continuous frames, with which it feeds the input layers of the Backbone. In order to be able to combine the TSN method with the gaze model too, we decide to create our architecture so, the gaze model is also fed with the face crops of the first 7 frames of each segment. In our experiment, we use 4 segments so the gaze model now will be fed with 4 inputs of 7 frames each, producing 4 sets of features from the last LSTM layer. Each set is consisted of 512 features, being the ones that are produced for the mid frame of each sequence, so in total for the 4 different segments 2048 features are produced. These total features are then fed into the backbone of the engagement estimation problem, along with the input images. We finally fuse them with the flattened features produced after the last convolutional layer of the C3D backbone (used in our best engagement model) and fed to the following three fc layers all together. At the end the backbone model will produce its output feature based on the fused features, which will be fed into the TSN head and the classification will be made.

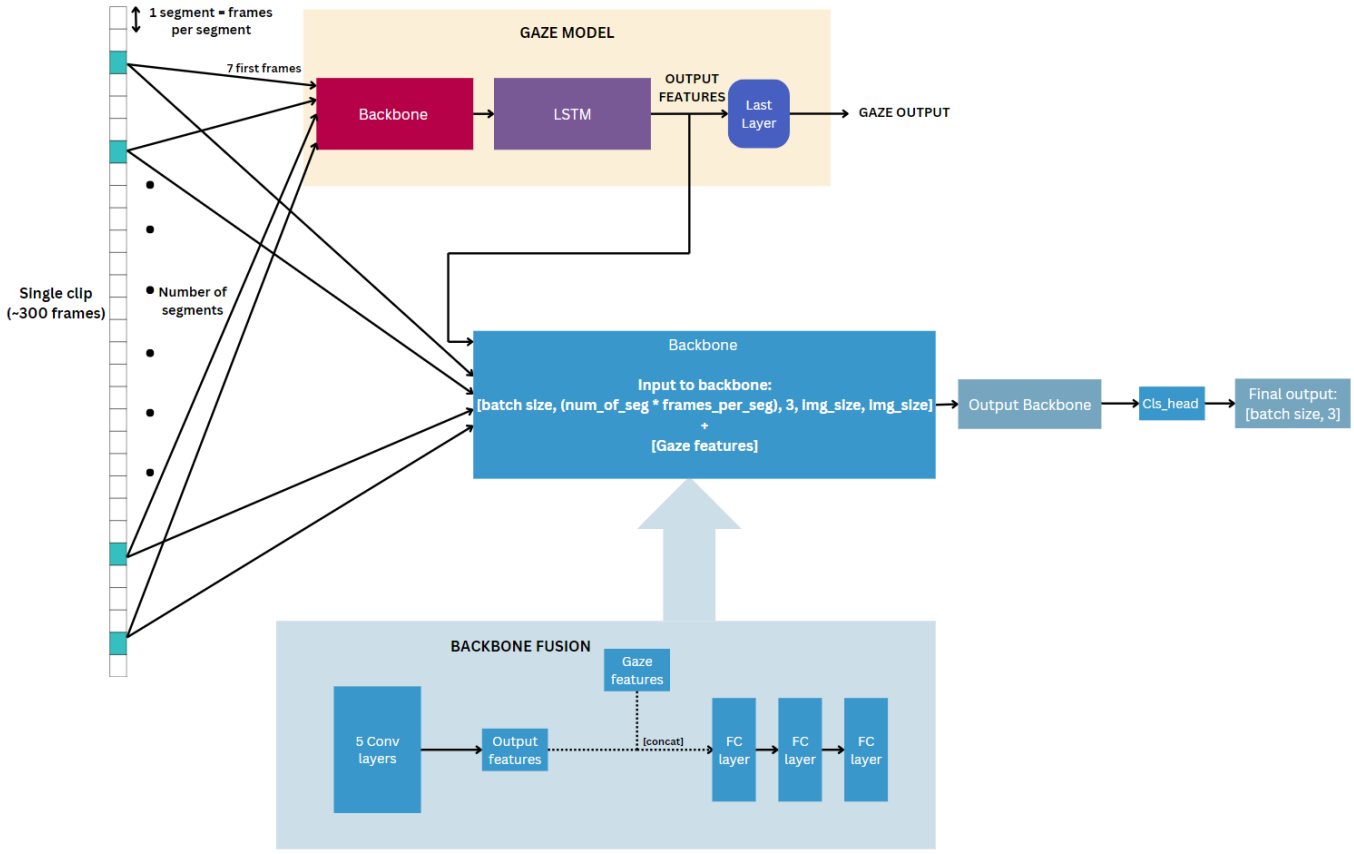


Figure 4.2.3: The EngagementPlusGaze network architecture, fusing Gaze tracking features produced by the frozen Gaze360 model [39] with the features produced by the last convolutional layer of the backbone of the engagement model.

Different fusion implementations were tested, including fusing the features with the outputs produced after the backbone, before being fed to the TSN head. The best results though, were achieved using the implementation

above.

4.3 Evaluation Metrics

To assess the performance of our engagement recognition framework, we are going to use the weighted F1-score, the accuracy and the weighted precision metrics. As our main indicator though we will consider the F1-score as it takes in consideration both precision and recall using their harmonic mean, and maximizing the F1 score implies simultaneously maximizing both precision and recall. The F1-score is given by the following formula:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{4.3.1}$$

The weighted F1-score is the weighted average of the F1-score of the different classes and respectively, the weighted precision is the weighted average of precision of different classes. Additionally, we don't choose the accuracy as our main metric, because it can often be misleading, for example in cases where there is a majority class and if the model just assigns all the samples to that class it achieves a really high accuracy, which might get it stuck in local minima and prevent it from further exploring the training space.

4.4 Experimental Results on Engagement Estimation

4.4.1 Preprocessing

In order to be able to train the selected models with the TSN architecture, using the framework shown in Figure 4.2.1, we firstly need to preprocess our videos. In these kind of datasets the preprocessing procedure is a really challenging one, as the existing parts of each dataset are the pairs of a video and an annotation file for each child. In each annotation file, which is in csv format, the labels are given for each frame in every row, and we create code that reads and creates the continuous segments for each category (eg. from 803 to 2300 the labels is goaloriented). After that, we take each continuous group of frames that belong to a single class and we split it in 10 second intervals (about 300 frames). The 10 second interval was chosen as this was found out to be the period during which the child's expressed engagement level can be detected.

Now, each 10 second clip is considered as a separate sample, and we create a dataset in a new format that will be easier for the models' training procedure and the dataloader. Each category folder contains a folder for every 10sec clip belonging to it, and each 10 sec clip folder contains all its frames in it.

During training the specified amount of total frames, which depends on the values selected for the number of segments and frames per segment (total frames = num of seg * frames per seg) will be selected from each one of the folders (usually no more than 40 frames in total) and will be used as input to the model for the specified sample.

Additionally, we performed data augmentation on the less populated categories of "aimless" and "no play", as the sample difference compared to the "goaloriented" one was really large. So, new 10 second clips were created for each one of the existing ones and were added in their corresponding two category folders, representing the new samples for the training procedure. The augmentation techniques that were used were HorizontalFlip, GaussianBlur, ElasticTransformation, and Salt and Pepper.

4.4.2 Experimenting with Resnet-50 backbone - TSN Architecture

We start our experiments by using the Resnet-50 as our backbone for the feature extraction. To implement our code we use the mmaction library and its packages. For the Resnet-50 backbone, we use pretrained weights derived from torchvision. The implementated model using the Resnet backbone is shown in Figure 4.4.1. We test this implementation using different combinations of the TSN and the training procedure parameters and the results produced are shown in Table 4.1.

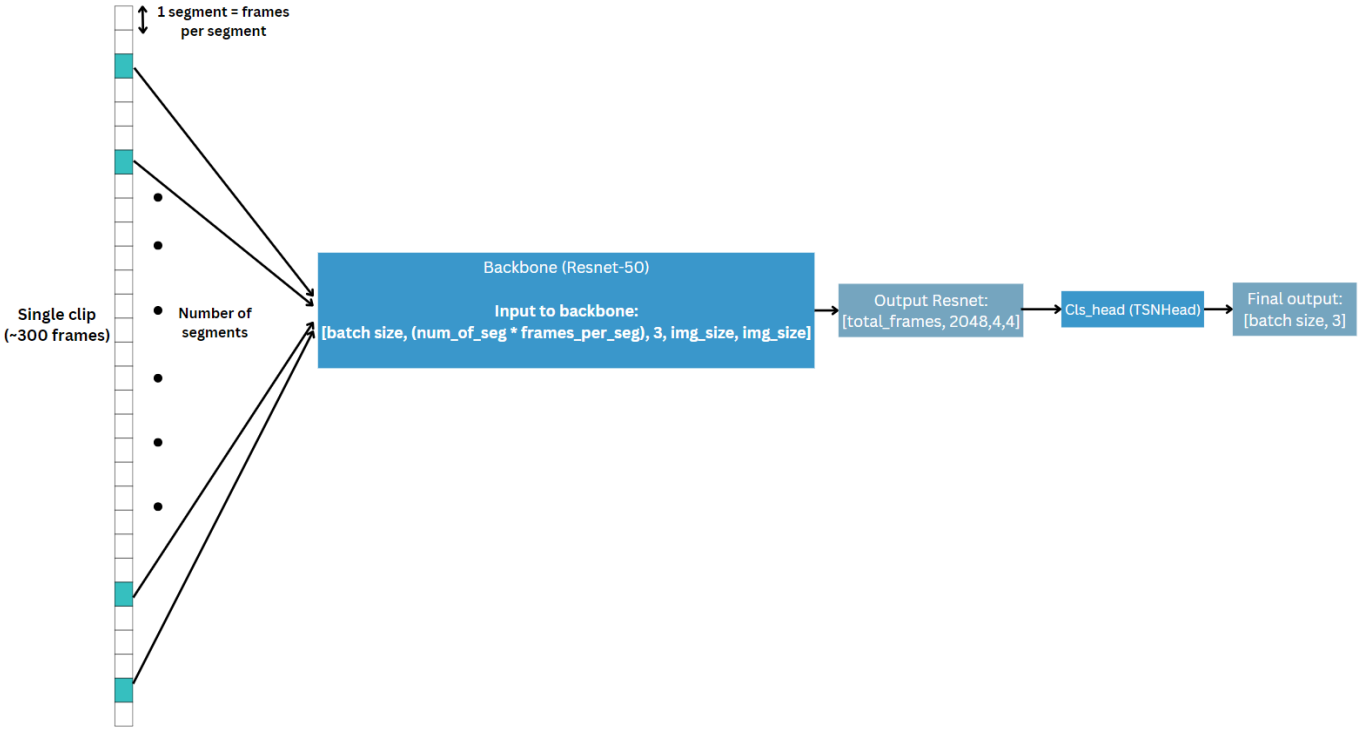


Figure 4.4.1: ResNet-50-TSN architecture used for the engagement estimation problem on the Pinsoro dataset [44].

Resnet-50									
Number of segs	Frames per seg	Image size	Optimizer	Learning rate	Gamma	Step size	Accuracy	w. F1	w. Precision
4	7	224	Adam	0.01	0.2	8	49.48	48.71	-
4	7	224	Adam	0.005	0.2	8	47.06	45.11	48.14
4	7	224	SGD	0.01	0.2	8	52.20	50.88	52.8
3	10	224	SGD	0.01	0.2	8	52.62	51.98	52.51
4	10	190	SGD	0.01	0.2	8	49.48	46.59	49.69
4	10	224	SGD	0.01	0.2	8	46.12	45.83	46.13
5	8	224	SGD	0.01	0.2	8	51.57	50.75	52.24
4	10	224	SGD	0.01	0.22	6	53.46	51.18	53.54

Table 4.1: Performance of Resnet-50 on Pinsoro dataset [44] using TSN architecture, for different hyperparameter values.

The results produced by the Resnet-50 backbone using the TSN architecture were promising, but not the best. A main reason for that, is because it was hard for the model to predict the minority classes. The highest F1-score achieved was 51.98%, having accuracy of 52.62% and precision of 52.51%. This outcome could be due to the fact that the Resnet-50 isn't a model architecture that processes the continuous frames temporally, so it might not capture well the temporal dynamics of the different categories.

The confusion matrix of the best result for this architecture is provided in the Figure 4.4.2. As we can see,

the model is able to predict 46.6% of the total no play segments and 35.8% of the aimless ones, in comparison to the 69.6% that it reached in the goaloriented segments (majority class).

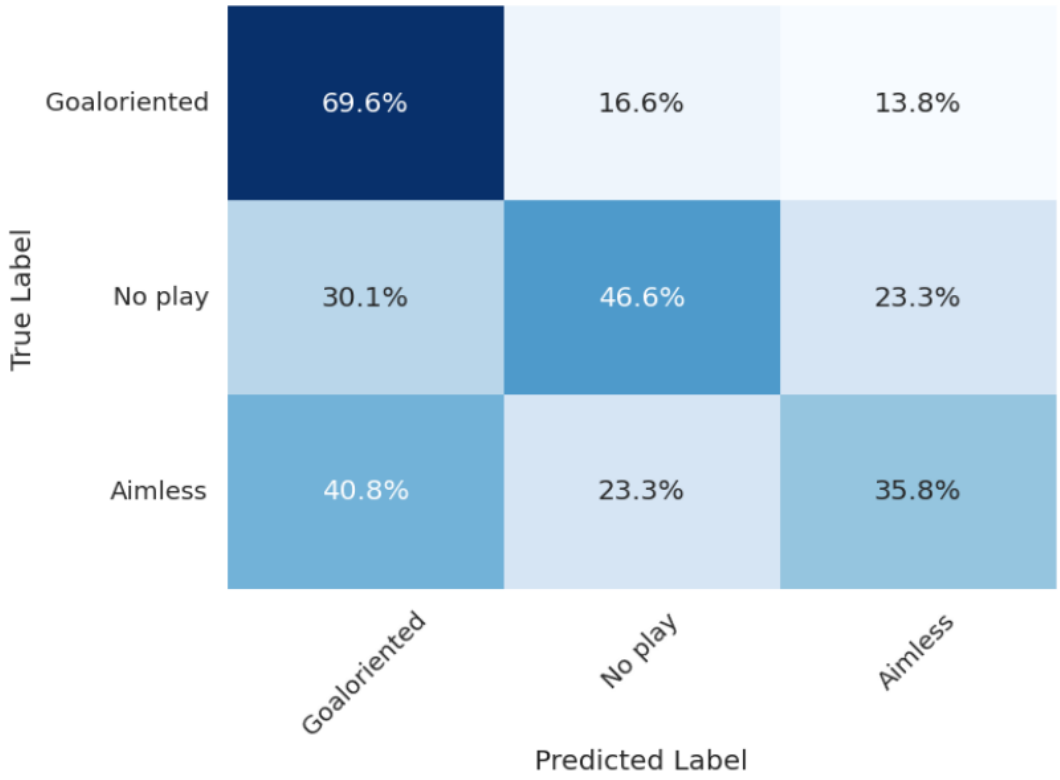


Figure 4.4.2: ResNet-50 best results for engagement estimation on the Pinsoro dataset. [44]

4.4.3 Experimenting with C3D backbone - TSN Architecture

In order to try to produce better results, we decided to utilize a backbone network that consists of 3D Convolutions, the C3D model [71]. The C3D can model appearance and motion information simultaneously, which helps it usually in outperforming other 2D ConvNet model architectures on various video analysis tasks. For our experiments, we started by modifying the same hyperparameters used for the Resnet-50 and after we found a combination that produced the best results, we kept them constant and tried to see if the results would improve by changing a bit the architecture of the model. So, the results shown in the Table 4.2 are produced using image size 112 (the size that the model originally has as input), the SGD optimizer, starting learning rate 0.01 with gamma 0.22 and step size 6.

The C3D backbone, originally is fed with 16 frames in total, which we tried testing first using the TSN sampling architecture, by selecting 2 segments of 8 frames from each clip (sample). After this initial run, we decided to modify the model a bit so that it can be fed with larger inputs, and instead of 16 frames in total we used 32, splitting them in 4 segments of 8 frames each. In our first implementation of this model with increased input, we didn't change the number of convolutional blocks or fully connected layers, we just changed the input of the first fully connected layer from 8192 to 16384. This as we can see in 4.2, led to worse results, probably because of the sudden drop to the fully connected features. So, in our second implementation, shown in Figure 4.4.3, we changed even more the input and output features of the two fully connected layers from 16384 –> 4096 –> 4096 to 16384 –> 8192 –> 4096.

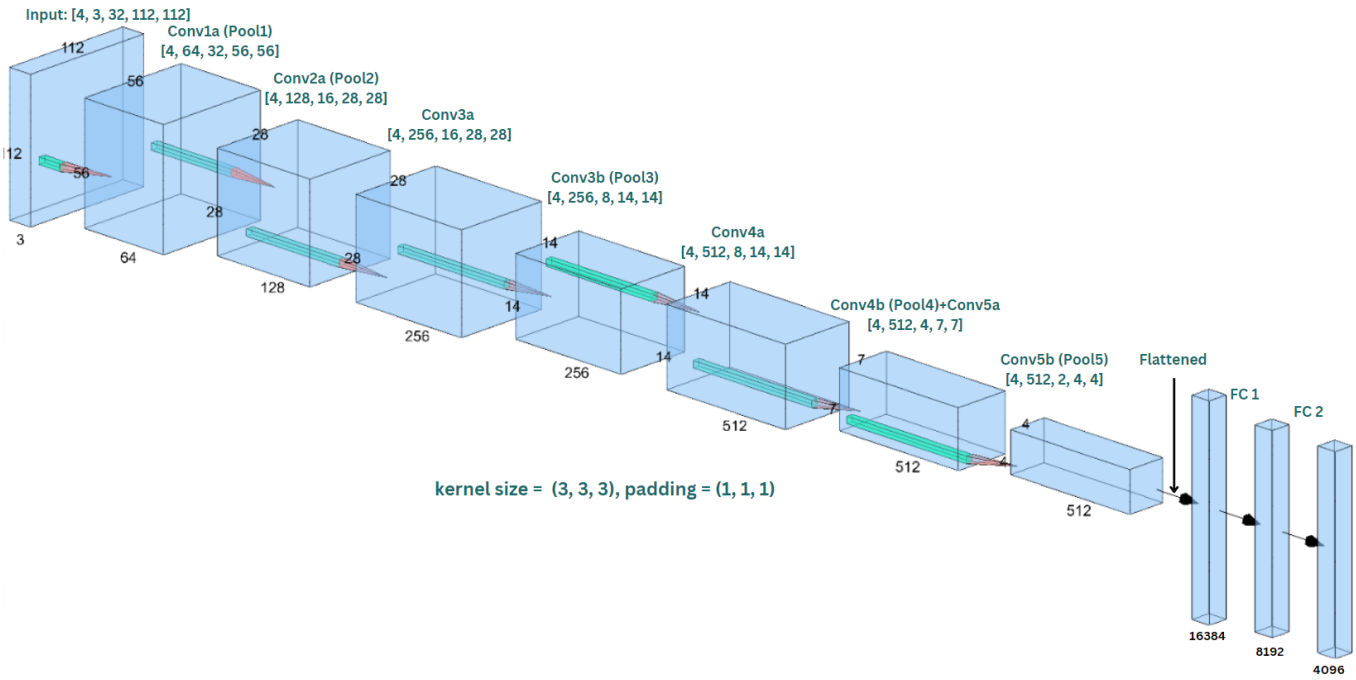


Figure 4.4.3: C3D model architecture with 32 frames as input and 2 fully connected layers.

In our final implementation, shown in Figure 4.4.4, we decided to add another fully connected layer in between the other two in order to process the features even more. So the new inputs and outputs of the three fully connected layers now are 16384 –> 8192 –> 4096 –> 4096.

C3D						
Number of segs	Frames per seg	FC layers	Feature Dimensions	Accuracy	w. F1	w. Precision
2	8	2	8192, 4096, 4096	61.85	61.24	62.72
4	8	2	16384, 4096, 4096	59.75	57.2	64.4
4	8	2	16384, 8192, 4096	64.15	62.81	65.37
4	8	3	16384, 8192, 4096, 4096	68.52	65.28	72.82
Resnet-50						
3	10			52.62	51.98	52.51

Table 4.2: Performance of different implentation of the C3D model using TSN architecture on the Pinsoro dataset

As we can see from the results in Table 4.2, the C3D network architectures significantly outperform the best Resnet-50 results, indicating the power of the 3d convolutions. In addition, we can see that even though the metric values dropped when we first increased the number of input frames, in the end we were able to improve them by modifying the fully connected layers. The best results were produced by our final modified C3D network architecture, taking in a total of 32 frames input, divided in 4 segments, producing a higher dimension output after the last convolutional layer, but then using 3 fully connected layers to steadily reduce it to the final output of 4096 features. All these result in the model having a higher number of trainable parameters, helping it to be able to learn the task at a higher level.

In Figure 4.5.3 we can view the confusion matrix for the best results produced using the final C3D modified network architecture. As we can see, the model has achieved great performance in its ability to recognize the majority class (goaloriented), and at the same time, more than doubling its ability to recognize the aimless class, compared to the results using the Resnet-50 model. We also notice a small drop in the estimation of

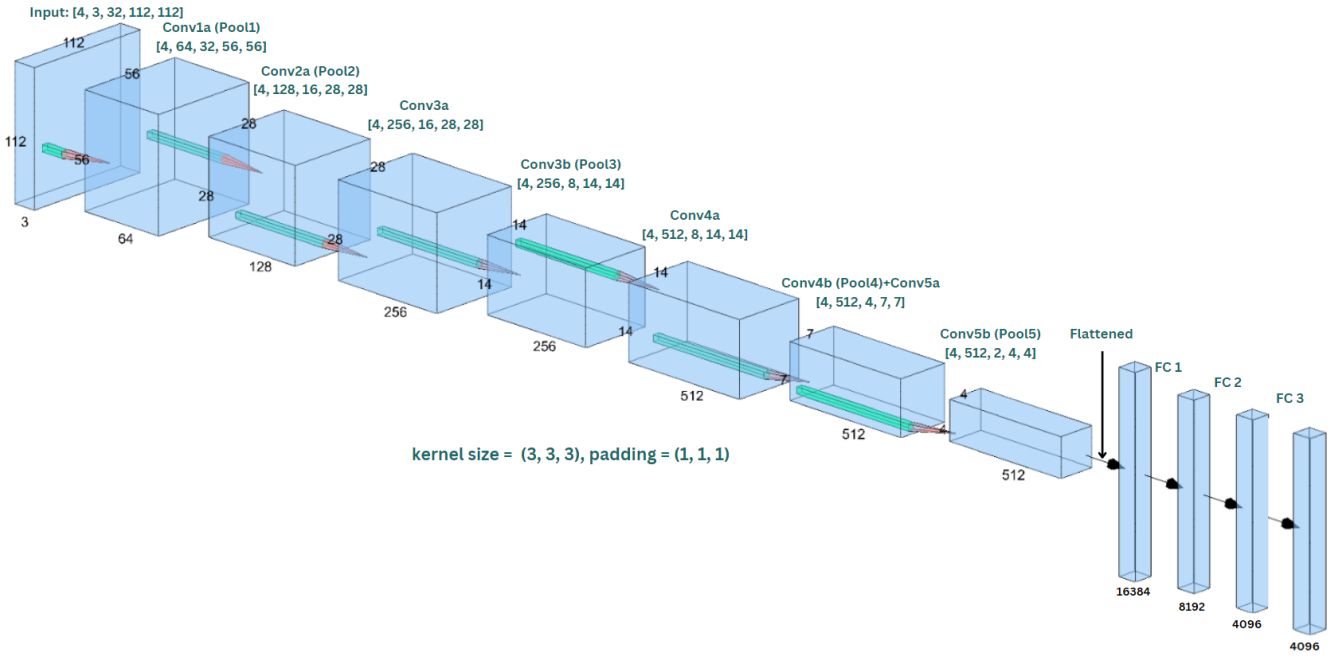


Figure 4.4.4: C3D model architecture with 32 frames as input and 3 fully connected layers.

the no play category, that comes as an outcome of the significant increase on the other two categories.

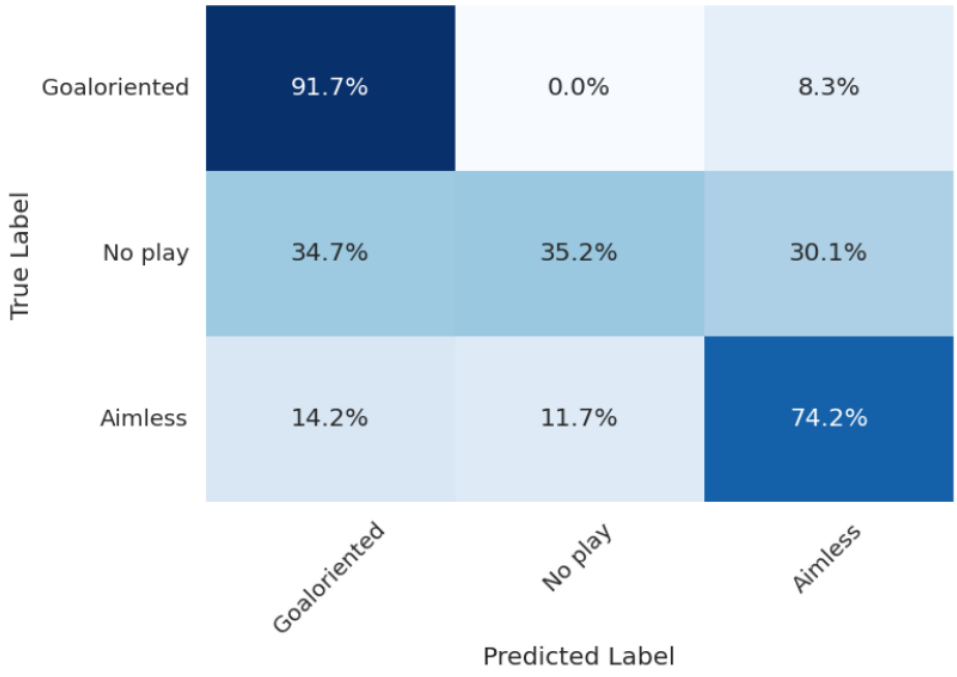


Figure 4.4.5: C3D results using on engagement estimation on the Pinsoro dataset, using a total of 32 frames as input and 3 fully connected layers .

4.4.4 Results Comparison

In order to better evaluate how well our architectures perform, we compare them to the results produced in [3] and [7], two studies that tackled the engagement estimation problem on the Pinsoro dataset.

Model Comparison			
Model	Accuracy	w. F1	w. Precision
majority class	38	20.55	14.85
ResNet-50	40.52	34.32	35.05
R(2+1)D RGB + Depth [3]	62.49	59.87	59.36
R(2+1)D RGB + Flow (Late) [3]	66.78	65.30	65.32
R(2+1)D RGB + Flow (Early) [3]	68.40	**67.11**	68.50
Resnet50-TSN (ours)	52.62	51.98	52.51
C3D-TSN-original (ours)	61.85	61.24	62.72
C3D-TSN (ours)	**68.52**	65.28	**72.82**

Table 4.3: Performance comparison of different architectures on engagement estimation on the Pinsoro dataset.

From the performance comparison in Table 4.3 we can see that only the R(2+1)D architecture with an early fusion of the RGB and the optical flow data outperforms our C3D-TSN framework. It is significant to mention though that our implementation doesn't need both raw and optical flow data, which means that it can save time at both the preprocessing phase and during training. Additionally, it is really important to mention the great improvement of the performance of the Resnet-50 model when used with TSN architecture compared to the standalone model, as the weighted F1-score was increased from 34.32% to 51.98%. These results firsty show how the the TSN architecture can improve standalone model architectures and additionally, how the C3DI3D model using TSN architecture can overcome the gap betwen using only Raw RGB frames and using both RGB frames and optical flow frames.

4.4.5 Training time Comparison

In Figure 4.4.6 we compare the train and validation losses, the accuracy and the F1-score, that were produced during the training of the two different architectures. We can notice that the architecture using the C3D model, achieves a high performance from an early stage during training and even though the train loss continuous to decrease after 10-12 epochs the validation loss stays pretty level meaning that the model starts to overfit. When we compare it to using the Resnet-50 backbone we can see that in that case the train and validation losses stay pretty close during the whole training. So, we can conclude that the C3D backbone is a really strong model and can achieve high results in the engagement detection task without needing a long time of training (20-30) epochs.

Figure 4.4.6: Comparison of the losses, accuracy and F1 score of the two different models during training.

4.4.6 Experimenting on ASD Games Dataset

In order to test even further the performance and robustness of our proposed architecture, we decided to evaluate it on one more dataset the ASD Games Dataset.

Preprocessing and training settings

In the ASD Games Dataset the annotations are in a different format, so we have to create new code for preprocessing to adapt to it. The annotations split each video into timesteps, with time values in milliseconds, and then they assign a category from one timestep to another. So, for every video we first extract every one of its frames to a folder, then we collect all the videos' annotations and after that we combine the consecutive timestep segments that belong to a single category to one segment. Finally, we iterate over each annotation and after we convert them to frame numbers, we then split each continuous segment into clips of a chosen 6 second interval this time, as there weren't a lot of large segments consisting of continuous frames of the same class in this dataset. This means that each 6 second clip, having a frame rate of 30fps, will consist of 180 frames. During training the specified amount of total frames, depending on the values selected for the number of segments and frames per segment, is selected from each one of those clips (usually no more than 40 frames in total) and is used as input to the model for the specified sample.

Additionally, in this dataset we perform data augmentation on the less populated categories in order to tackle the problem of the big sample difference. The augmentation techniques that were used this time were RandomRotate90, Transpose, ShiftScaleRotate, Blur, OpticalDistortion, GridDistortion, HueSaturationValue from the albumentations library.

Results

The performance of C3DI3D model with TSN architecture in this dataset, reached similar results with the PInSoRo dataset. Those results, include a 69.8% accuracy, a 70.67% weighted precision and a 68.02% f1 score. These results indicate how robust the proposed network is, as it can be used for engagement detection, between different datasets, that include different settings.

Accuracy	w. F1	w. Precision
69.8	68.02	70.67

Table 4.4: Performance of C3D-TSN model on the ASD games dataset.

4.5 Experimental Results on Gaze Estimation

4.5.1 Training of Gaze Estimation framework on WiDo Dataset

Preprocessing

Just like the datasets that are focusing on the engagement task, the WiDo dataset is a really demanding dataset to process and efficiently train on it for the gaze tracking task.

In the first phase, just like in the previous two datasets used in the engagement estimation task we have to create the annotations that will be used. In this dataset, the annotation files are in a similar format with the ASD Games Dataset, where each video is split in time segments with up to milliseconds accuracy. In the gaze tracking problems though, the inputs to the model are consisted of continuous frames of the video even if they don't belong to the same category. For this reason, our file system isn't the same as we saw on the engagement problem, instead here we just extract all the frames of each video in a folder that is named as the video.

Additionally, the model architecture that we will be using in this study is similar to the one proposed in 4.2.2, which takes as input 7 consecutive frames and predicts the gaze of the central one. For this reason, in order to create the annotations we iterate over the frames of each video, with a step of five frames, and we create every annotation sample to consist of the video name that it belongs to, its starting frame of the seven and the category that the gaze direction belongs to, that is the category that the mid frame belongs to, which is found from the timestep segments. An example of the annotations format, that we create is shown below:

```
WiDo_MiniDV-01,8990,2
WiDo_MiniDV-01,8995,2
WiDo_MiniDV-01,9000,2
```

Like we saw in numerous studies that approach the gaze tracking problem, in order for the networks to better tackle the problem and produce better estimates, they need to be fed with continuous face crops of the subjects as the inputs, and not the whole frames [39, 26, 12]. For this reason, we have to collect the face crops of the specific dataset. However, a challenge arises due to the nature of the annotations of the gaze tracking task of the WiDo dataset compared to other datasets specifically focusing on the gaze tracking problem. In the WiDo dataset the labels are consisted of categories of the gaze direction, while in other ones they are consisted of the gaze direction vectors. So, in our dataset it won't be enough to just predict the gaze direction vectors, but we will also need to check to which category of the six that gaze direction belongs to. In order to be able to make that association between the gaze vector and the categories easier, we decided first to converge some of the categories together. So, instead of having six different categories now our dataset consists of three: (i) Looking at some part of the mother's body, (ii) Looking at the mother's face and (iii) looking somewhere else. But we still will need to know the coordinates of the child's gaze starting point, the mother's head and the mother's body.

In the first phase, to be able to detect the bounding box of the mother and the child in each frame, we utilize the YOLO (You Only Look Once) model [38], which is a real-time object detection model known for its speed and accuracy. Unlike traditional models, which use sliding windows or region proposals, YOLO treats object detection as a single regression problem, predicting both class probabilities and bounding boxes simultaneously. This allows it to process an entire image in one go, making it very fast. YOLO divides the image into a grid and predicts bounding boxes and class probabilities for each grid cell. It's highly efficient for tasks requiring real-time detection, such as video analysis. After, we have extracted the human bounding boxes each frame we make the distinction of which bounding box belongs to whom, based on the x_min coordinates of each bounding box and which person is to the left. For this reason, we have to be really careful and look at every few thousand frames of each video the relative position between the mother and the child in case they change. After we collect the two bounding boxes we use the mediapipe framework [51] to detect the facial and body landmarks of every person and extract the mother's face and body bounding boxes and the child's starting gaze point. The child's starting gaze point is calculated as the average of the left and right eye coordinates found in the child's facial landmarks. We store these coordinates for each frame in a mapping file that corresponds video/frame to its coordinates set, which will be passed to the model during

training time to help it make its predictions. A few examples of frames from the datasets and the coordinates sets found are shown in the Figure 4.5.1.

Figure 4.5.1: Example images from the WiDo dataset with detected bounding boxes and starting gaze point.

Finally, we create a mapping file that makes an association between each frame and the coordinates of the mother's head and body and the starting point of the child's gaze. This file is provided to the dataloader along with the annotations files, so during training, the dataloader selects from it the respective coordinates corresponding to the mid frame of each input to give to the model along with the input frames.

Experiments and Results

As a first step we tried training the framework shown in Figure 4.2.2 even further, by using the pretrained weights only for the backbone model. The best results that were produced using this implementation are shown in 4.5.2. We can see that it reaches a high accuracy on predicting the elsewhere and the mother's body categories. That being said though we can also notice that its predictions are mostly towards the "elsewhere" category, which isn't desirable. Sadly the results aren't as good as we would like, as the nature of this modified problem in this kind of dataset where there is a high amount of movements by the children is challenging. In addition to that, the fact that some samples may be classified correctly in the class "elsewhere", but the gaze produced by the model still may not be correct can result in messing up the training. For this reason, we aren't going to continue our study by trying to train a gaze model even further on this dataset, but instead we will be using a pretrained model as a whole. Finally the values of the metrics achieved by this method are the following:

Accuracy	w. F1	w. Precision
54.94	50.82	62.34

Table 4.5: Performance of gaze tracking architecture after training in gaze estimation on the WiDo dataset.

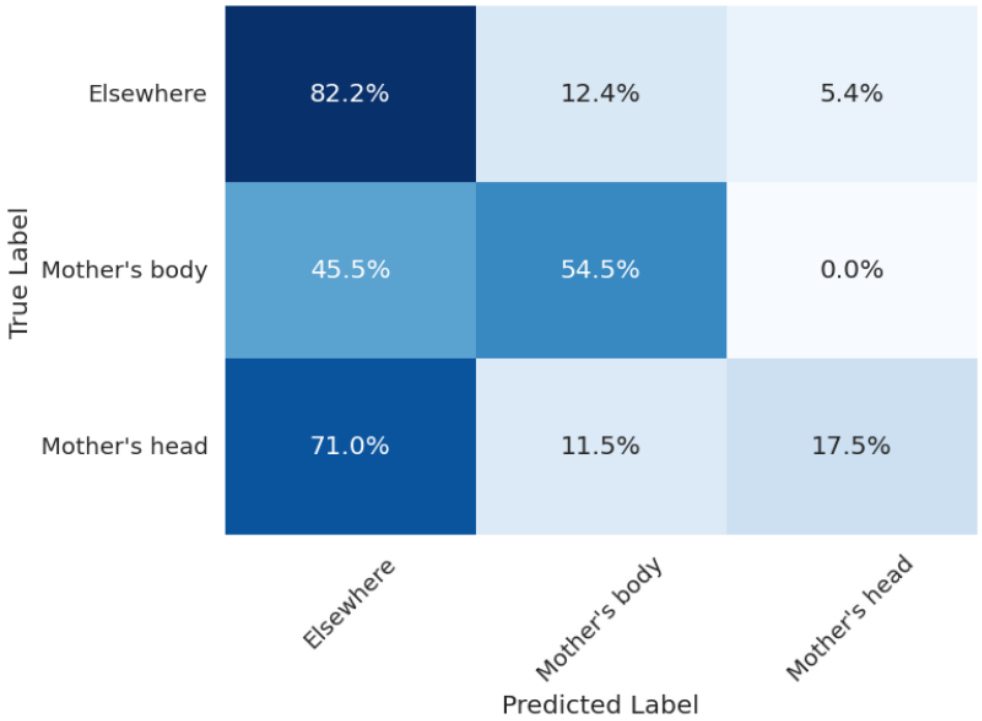


Figure 4.5.2: Confusion matrix of the results produced for Gaze detection classification of 3 categories on the WiDo dataset.

4.5.2 Pretrained Gaze360 model implementation for Gaze estimation

As we mentioned in the previous section, the task of training even further the WiDo dataset on gaze tracking can be really challenging due to the nature of the problem and also the fact that the labels aren't coordinate directions. For this reason, as a next step of our study we create a similar framework, by taking the whole pretrained model on the Gaze360 dataset adding the module of the check of the bounding boxes intersection and testing how it will perform, with no further training on the WiDo dataset on the problem of gaze tracking with the mother's head and body bounding boxes.

The results produced were really improved and are shown below:

Accuracy	w. F1	w. Precision
68.33	69.36	72.08

Table 4.6: Performance of fully pretrained gaze tracking architecture in gaze estimation on the WiDo dataset for 3 categories.

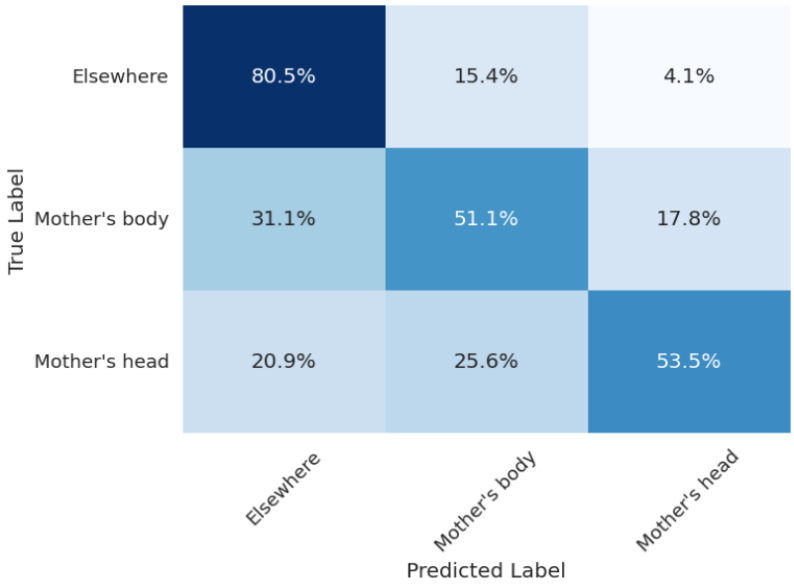


Figure 4.5.3: Gaze tracking results on WiDo dataset for 3 categories, using the fully pretrained model from Gaze360.

As we can see the accuracy achieved in all 3 categories has improved significantly, indicating at how our network implementation can be used for such a challenging task, as this one. In addition to that, it can be really useful in implementations in which we want to be able to model the different gaze patterns between children with autism (similar social interactions with children with Williams and Down disorders) and typically developing ones, as they don't look as much at the other people participating in their interactions.

In Figure 4.5.4, we can see some examples of the output produced by the model on the WiDo dataset. As we can see the model's predictions are really close to the real gaze directions of the children and by using the bounding box check module to assign them to each category, we achieve our goal at a higher performance.

Figure 4.5.4: Examples of Gaze tracking predictions on the WiDo dataset.

4.5.3 Gaze Estimation using 4 categories

In order to further investigate the gaze patterns of children not typically developing, we will need to also investigate in our studies how focused they are on the task/object that they are occupied with. This is important as children with ASD tend to lose focus from the interaction and get distracted by other objects easier than normally developing ones. For this reason, now we further divide the "Elsewhere" category that we used in the previous section and our problem consists of the following four categories:

1. Not looking somewhere exactly or the camera / Looking at an item that only the child is using but not the mother (New Elsewhere category)
2. Looking at an item that is used in the interaction with the mother
3. Looking at some part of the mother's body
4. Looking at the mother's head

Yolo training for object detection

In order though to be able to continue with our goal to predict the gaze of the children based on the 4 previous categories a challenge arises. This challenge consists of the fact that the specific dataset doesn't contain a single object that is fixed in all frames so we have its specific coordinates, but instead there are multiple different toys that are used in each interaction and their positions are evolving during the development of the interaction. So, in order to be able to investigate our problem using our method described in 4.2.2 we will need to collect the toy coordinates. To achieve that, we use the YOLO model, to which we perform additional training for 200 epochs, by creating our own small datasets of the objects we want to detect, using images that we annotate. A few examples of the object detections we performed are shown in Figure 4.5.5. We carry out this object detection in different segments of 5 of the children of the WiDo dataset.

Figure 4.5.5: Examples of pink stuffed toy detection on images of the WiDo dataset.

Gaze Estimation results on 4 categories

After we have preprocessed the samples and collected the coordinates needed, we test the same network architecture as in the previous section. The results produced are shown in Figure 4.5.6

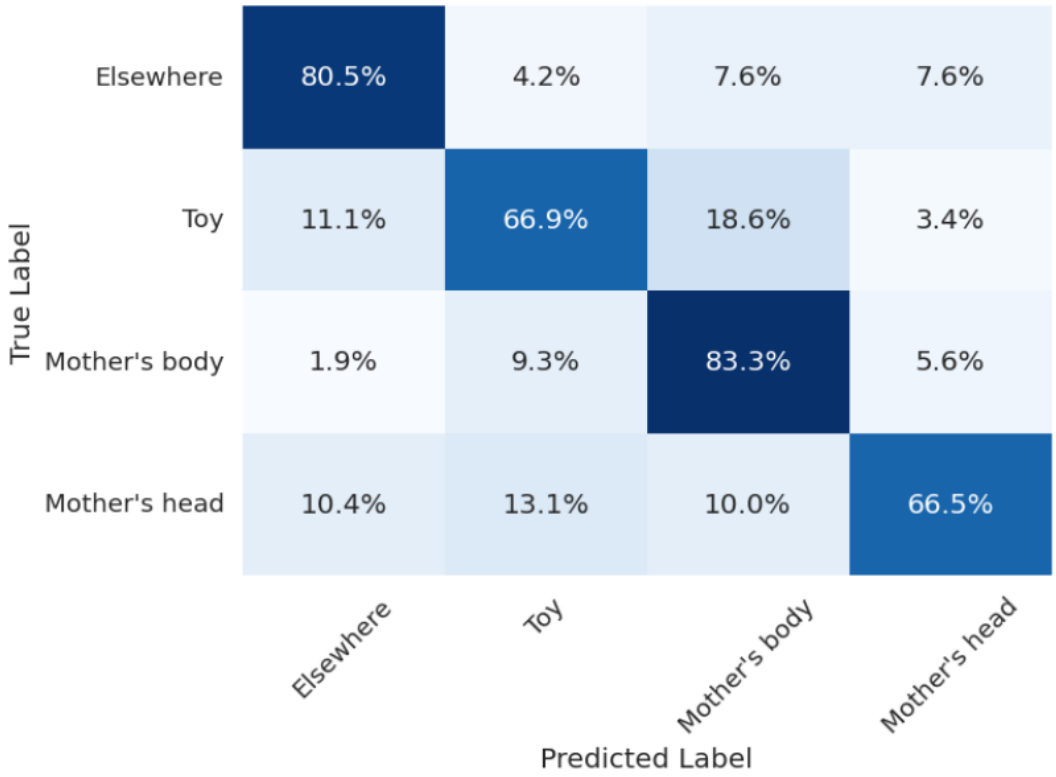


Figure 4.5.6: Results produced for four category gaze detection on the WiDo dataset.

As we can see the network performs really well, being able to distinguish the samples between the four different categories at a high rate. The metric values achieved by this method are shown in Table 4.7, with a really high accuracy of **69.24%** and a weighted F1 score of **72.34%**. It is crucial to know whether the child is looking on the toy (object of interaction) or at some other object around, and not just if it's looking at some part of the body of its mother or not, as it is a way to distinguish children with ASD with the normally developing ones, as they tend to get distracted really easily, by other objects or people in their environment. So, as it can be seen from the confution matrix, we achieve a really high percentage of 80.5% on the detection of the elsewhere class, which indicates the moments that the child gets "distracted". That being said, the model is still not perfect as the gaze tracking problem especially in videos where there is a lot of movement by the people participating in them can be really challenging, for example there might be times when the object of distraction could possibly be somewhere behind the toy or the mother, so even though the gaze is correctly produced, the sample can be wrongly classified to another category.

Accuracy	w. F1	w. Precision
69.24	72.34	80.50

Table 4.7: Performance of fully pretrained gaze tracking architecture in gaze estimation on the WiDo dataset for 4 categories.

In Figure 4.5.7 we present some of the gaze predictions produced by this architectures in the problem consisting of four categories using frames from different children. As we can see, the predictions are pretty accurate,

which is needed especially for situations where the bounding box is smaller, for example the head of the mother.

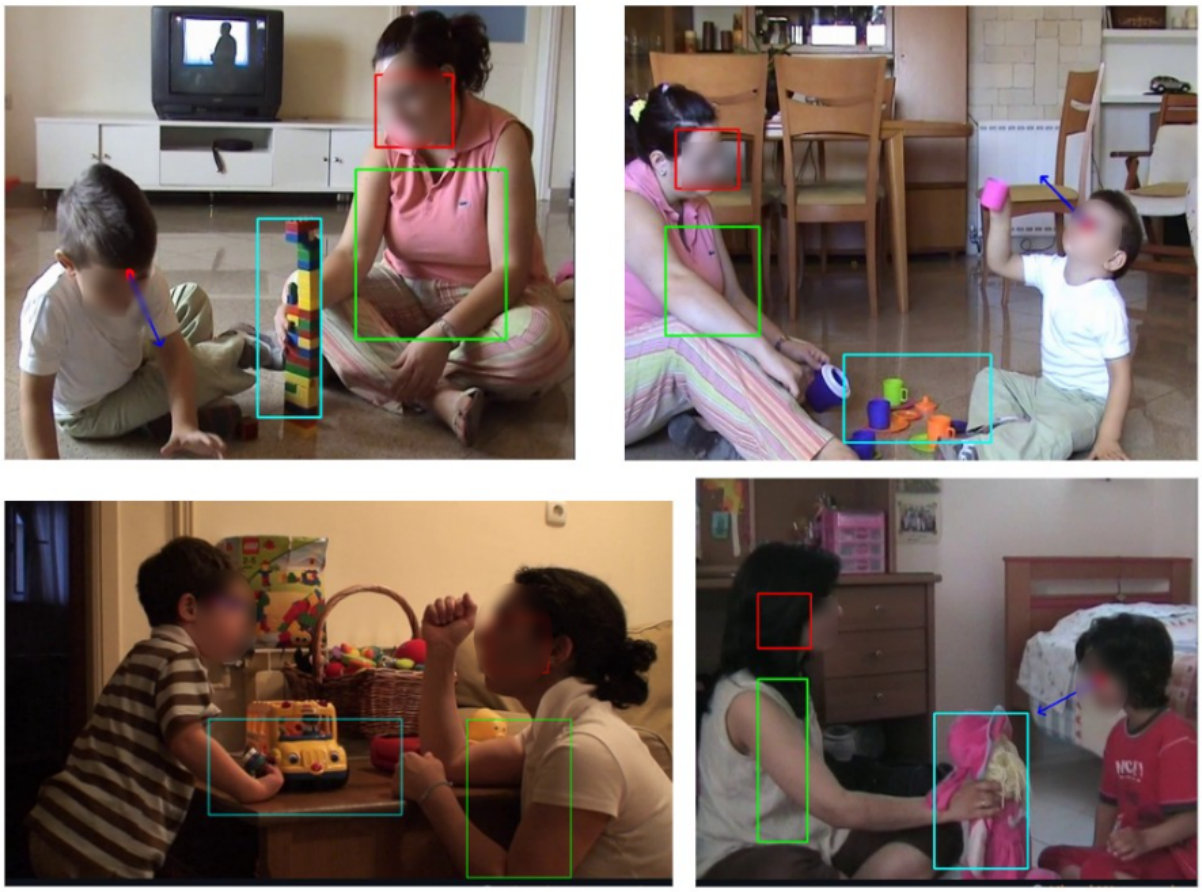


Figure 4.5.7: Results produced for four category gaze detection on the WiDo dataset.

4.6 Experimental Results on Fusion of Engagement and Gaze Estimation

After creating the Engagement Plus Gaze network proposed in Section 4.2.3, that combines the gaze tracking features from the gaze model, together with the engagement features produced by the backbone of the engagement model, we continue on testing it. Like mentioned before, we tested it using different fusion techniques, including fusing the features with the outputs produced after the backbone, before being fed to the cls head, but the best results were produced with the implementation shown in Figure 4.2.3. The confusion matrix, for these best results is shown in Figure 4.6.1.

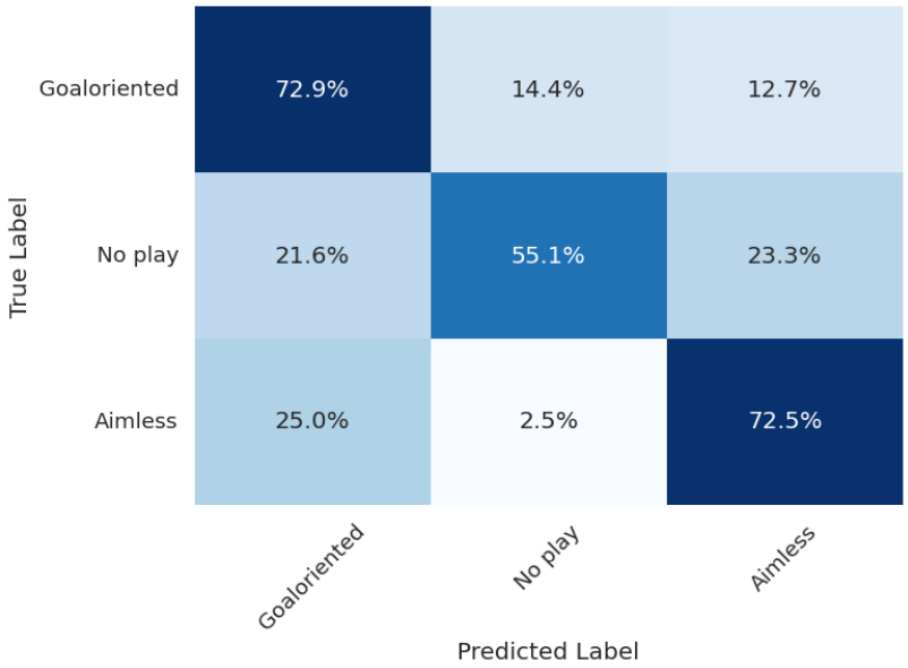


Figure 4.6.1: The confusion matrix for the best results produced using the EngagementPlusGaze framework on the Pinsoro dataset [44].

As we can see the distribution of the predictions has improved greatly. Even though we can notice a decrease in the accuracy of the goaloriented category, that is balanced by the increase of the distribution for the no play one, which was really low in the previous results. This increase is really important as the network can distinguish the three categories with high certainty.

Figure 4.6.2: Examples of the gaze estimations produced on the Pinsoro dataset [44].

In the Table 4.8, we can see the metric results of the engagement plus gaze model compared to the model from the Section 4.4, where the gaze tracking model wasn't included. We can notice that we have a decrease in the accuracy and the precision, mainly because of the drop in the goaloriented category which has the most samples. That being said though, the most important outcome is the increase of the weighted F1 score value, which is our main metric indicator.

Model Comparison			
Model	Accuracy	w. F1	w. Precision
C3D-TSN-original (ours)	61.85	61.24	62.72
C3D-TSN (ours)	**68.52**	65.28	**72.82**
C3D-PlusGaze-TSN (ours)	66.25	**66.15**	67.95

Table 4.8: Performance comparison of our three different architectures on engagement recognition.

4.7 Discussion

With our work in this thesis, firstly we demonstrate how the Temporal Segment Networks (TSN) method can be implemented in model architectures for challenging tasks, like the engagement estimation, as it has mostly been used for action recognition problems in past research. Here, we show that its addition can be useful, as combined with a 3D convolution model, it can achieve significant results, just by using RGB data for training. In addition to that, we show how gaze tracking model architectures can be implemented effectively into more challenging into the wild gaze estimation problems, like gaze classification ones, where we have to decide whether a child is looking at an object or person of interest and not just to provide the gaze vectors. Finally, we propose a new way that the gaze tracking architectures can be used, this time with them being

a helping tool for further improve the results produced on the engagement estimation task, like we show on Table 4.8, by implementing a fusion between the features produced by the gaze model and the ones produced by the engagement backbone.

Chapter 5

Conclusion

5.1 Brief Summary and Conclusion

Studying child interactions is crucial for understanding how children develop social, emotional, and cognitive skills, particularly in early childhood. Being able to estimate the engagement of children especially is crucial, because it provides valuable insights into how individuals interact with tasks, environments, or social settings. In fields like child development or therapy, especially for children with developmental disorders like autism, engagement estimation offers important clues about social and cognitive development. By tracking attention and emotional responses, caregivers and professionals can better support a child's growth and adapt interventions accordingly.

For the reasons above, in this study we first tackled the task of the engagement estimation, building and experimenting with different architectures, implemented using the Temporal Segment Networks (TSN) method, which splits each video clip into segments and processes a specific number of segments individually first and then combines their results, to also learn from the temporal dynamics of each clip. For backbones we experimented with different ones, including the Resnet-50 and C3D models, and we concluded to the importance of 3D convolutions for video based problems, as the networks that included the C3D architecture produced the best results, almost reaching the performance of networks that combined RGB and optical flow data, by just using RGB data and the TSN method.

Additionally, as a next step in our goal to better understand children interactions, we decided that it would be crucial to experiment with the Gaze tracking task, as it could be benefitial for understanding where each child's focus is. Firstly, we were successful in creating a network architecture, which distinguished between 3 categories of the WiDo dataset (Elsewhere, Mother's body, Mother's head). This architecture consisted of an already pretrained gaze estimation model combined with an implemented "check module", which received the gaze vectors and decided whether or not the gaze direction crossed one of the bounding boxes that represented the mother's head and body. After this first implementation, we concluded that it would be important to also know when the children are looking at the objects used in the interactions or are distracted, so we performed the same procedure for the WiDo dataset but using 4 categories, splitting the elsewhere category into "Toy" and "Elsewhere".

Finally, after seeing how successfully gaze tracking methods can be implemented into visual children interaction problems, we decided to try integrating gaze tracking data with our engagement estimation model and test out how it would perform on the Pinsoro dataset. The results produced where significant as this fusion of gaze tracking and engagement features, further improved the performance on the engagement estimation task.

In short, the contributions of our work in the research community are the following:

1. Development of engagement estimation framework and demonstration of how the Temporal Segment Networks (TSN) method can be implemented along with it.
2. Studying of how gaze architecture can be implemented effectively into more challenging into the wild gaze estimation problems, during free interactions of young children with developmental disorders and their mothers
3. Proposal and experimentation of how the engagement estimation and gaze tracking problems can be fused to further improve the engagement estimation task.

5.2 Future Work

In future work, there is a number of possible directions that research could move towards to firstly further improve the current network architectures on the engagement and gaze estimation tasks, but also try and use them focusing on higher goals:

- Use the proposed network of the C3D model combined with the TSN architecture, but train it with both optical flow and Raw RGB data, which can prove benefitial in videos where the environment

stays constant especially, which is the case on the Pinsoro dataset, so crucial information can also be extracted from the movements of the children.

- For the EngagementPlusGaze network, experiment with different more complex fusion techniques, for example multi-head attention mechanisms. In this way, the fused features could possibly provide more meaningful information and further improve the engagement estimation.
- Use the proposed engagement estimation architecture to identify children with Autism Spectrum Disorders, by extracting their different engagement and gaze patterns produced, compared to the typically developing children.

Appendix A

Bibliography

[1] Abedi, A. and Khan, S. S. *Improving state-of-the-art in Detecting Student Engagement with Resnet and TCN Hybrid Network*. 2021.

[2] Ali, S., Mehmood, F., Dancey, D., Ayaz, Y., Khan, M. J., et al. "An adaptive multi-robot therapy for improving joint attention and imitation of ASD children". In: *IEEE Access* 7 (2019), pp. 81808–81825.

[3] Anagnostopoulou, D., Efthymiou, N., Papailiou, C., and Maragos, P. "Child Engagement Estimation in Heterogeneous Child-Robot Interactions Using Spatiotemporal Visual Cues". In: *Proc. IROS*. 2022.

[4] Anagnostopoulou, D., Efthymiou, N., Papailiou, C., and Maragos, P. "Engagement Estimation During Child Robot Interaction Using Deep Convolutional Networks Focusing on ASD Children". In: *Proc. ICRA*. 2021.

[5] Bai, S., Kolter, J. Z., and Koltun, V. *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. 2018.

[6] Bakeman, R. and Adamson, L. B. "Coordinating attention to people and objects in mother-infant and peer-infant interaction". In: *Child development* (1984), pp. 1278–1289.

[7] Bartlett, M. E., Stewart, T. C., and Thill, S. "Estimating levels of engagement for social human-robot interaction using legendre memory units". In: *Proc. HRI*. 2021.

[8] Ben-Youssef, A., Clavel, C., Essid, S., Bilac, M., Chamoux, M., et al. "UE-HRI: a new dataset for the study of user engagement in spontaneous human-robot interactions". In: *Proceedings of the 19th ACM international conference on multimodal interaction*. 2017.

[9] Biswas, A. and Islam, M. S. "An efficient CNN model for automated digital handwritten digit classification". In: *Journal of Information Systems Engineering and Business Intelligence* (2021), pp. 42–55.

[10] Cao, Q., Shen, L., Xie, W., Parkhi, O. M., and Zisserman, A. "Vggface2: A dataset for recognising faces across pose and age". In: *IEEE international conference on automatic face & gesture recognition*. 2018.

[11] Cao, Z., Simon, T., Wei, S.-E., and Sheikh, Y. "Realtime multi-person 2d pose estimation using part affinity fields". In: *Proc. CVPR*. 2017.

[12] Cătrună, A., Cosma, A., and Rădoi, E. "CrossGaze: A Strong Method for 3D Gaze Estimation in the Wild". In: *arXiv preprint arXiv:2402.08316* (2024).

[13] Chen, Z. and Shi, B. E. "Appearance-based gaze estimation using dilated-convolutions". In: *Asian Conference on Computer Vision*. 2018.

[14] Cho, K., Merrienboer, B. van, Bahdanau, D., and Bengio, Y. *On the Properties of Neural Machine Translation: Encoder-Decoder Approaches*. 2014.

[15] Cornia, M., Baraldi, L., Serra, G., and Cucchiara, R. "Predicting Human Eye Fixations via an LSTM-Based Saliency Attentive Model". In: *IEEE Transactions on Image Processing* 27 (2018), pp. 5142–5154.

[16] Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q. V. "Randaugment: Practical Automated Data Augmentation With a Reduced Search Space". In: *Proc. CVPR*. 2020.

[17] Dawson, G., Toth, K., Abbott, R., Osterling, J., Munson, J., et al. "Early social attention impairments in autism: social orienting, joint attention, and attention to distress." In: *Developmental psychology* 40 (2004), p. 271.

[18] Del Duchetto, F., Baxter, P., and Hanheide, M. "Are You Still With Me? Continuous Engagement Assessment From a Robot's Point of View". In: *Frontiers in Robotics and AI* 7 (2020).
[19] Del Duchetto, F., Baxter, P., and Hanheide, M. "Are you still with me? continuous engagement assessment from a robot's point of view". In: *Frontiers in Robotics and AI* 7 (2020), p. 116.
[20] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., et al. "Imagenet: A large-scale hierarchical image database". In: *Proc. CVPR*. 2009.
[21] Efthymiou, N., Filntisis, P. P., Koutras, P., Tsiami, A., Hadfield, J., et al. "Childbot: Multi-robot perception and interaction with children". In: *Robotics and Autonomous Systems* 150 (2022), p. 103975.
[22] Efthymiou, N., Filntisis, P. P., Potamianos, G., and Maragos, P. "Visual robotic perception system with incremental learning for child–robot interaction scenarios". In: *Technologies* 9.4 (2021), p. 86.
[23] Fang, Y., Duan, H., Shi, F., Min, X., and Zhai, G. "Identifying Children with Autism Spectrum Disorder Based on Gaze-Following". In: *Proc. ICIP*. 2020.
[24] Glorot, X., Bordes, A., and Bengio, Y. "Deep sparse rectifier neural networks". In: *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. 2011, pp. 315–323.
[25] Goodfellow, I. *Deep learning*. 2016.
[26] Guan, Y., Chen, Z., Zeng, W., Cao, Z., and Xiao, Y. "End-to-end video gaze estimation via capturing head-face-eye spatial-temporal interaction context". In: *IEEE Signal Processing Letters* 30 (2023), pp. 1687–1691.
[27] Güler, R. A., Neverova, N., and Kokkinos, I. "Densepose: Dense human pose estimation in the wild". In: *Proc. CVPR*. 2018.
[28] Gupta, A., D'Cunha, A., Awasthi, K., and Balasubramanian, V. "Daisee: Towards user engagement recognition in the wild". In: *arXiv preprint arXiv:1609.01885* (2016).
[29] Hadfield, J., Chalvatzaki, G., Koutras, P., Khamassi, M., Tzafestas, C. S., et al. "A deep learning approach for multi-view engagement estimation of children in a child-robot joint attention task". In: *IROS*. 2019.
[30] He, K., Zhang, X., Ren, S., and Sun, J. *Deep Residual Learning for Image Recognition*. 2015.
[31] Hochreiter, S. "Long Short-term Memory". In: *Neural Computation MIT-Press* (1997).
[32] Hochreiter, S. and Schmidhuber, J. "Long Short-term Memory". In: *Neural computation* 9 (1997), pp. 1735–80.
[33] Hu, M., Jiang, K., Wang, Z., Bai, X., and Hu, R. "Cycmunet+: Cycle-projected mutual learning for spatial-temporal video super-resolution". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2023).
[34] Hu, P. and Ramanan, D. "Finding tiny faces". In: *Proc. CVPR*. 2017.
[35] Jenner, L., Farran, E., Welham, A., Jones, C., and Moss, J. "The use of eye-tracking technology as a tool to evaluate social cognition in people with an intellectual disability: a systematic review and meta-analysis". In: *Journal of Neurodevelopmental Disorders* 15 (2023), p. 42.
[36] Ji, S., Xu, W., Yang, M., and Yu, K. "3D convolutional neural networks for human action recognition". In: *IEEE transactions on pattern analysis and machine intelligence* 35 (2012), pp. 221–231.
[37] Jiang, M., Huang, S., Duan, J., and Zhao, Q. "SALICON: Saliency in Context". In: *Proc. CVPR*. 2015.
[38] Jocher, G. *Ultralytics YOLOv5*. 2020.
[39] Kellnhofer, P., Recasens, A., Stent, S., Matusik, W., and Torralba, A. *Gaze360: Physically Unconstrained Gaze Estimation in the Wild*. 2019.
[40] Kim, E. S., Berkovits, L. D., Bernier, E. P., Leyzberg, D., Shic, F., et al. "Social robots as embedded reinforcers of social behavior in children with autism". In: *Journal of autism and developmental disorders* 43 (2013), pp. 1038–1049.
[41] Koutras, P. and Maragos, P. "Estimation of eye gaze direction angles based on active appearance models". In: *Proc. ICIP*. 2015.
[42] Krafka, K., Khosla, A., Kellnhofer, P., Kannan, H., Bhandarkar, S., et al. "Eye tracking for everyone". In: *Proc. CVPR*. 2016.
[43] Lecun, Y., Bottou, L., Bengio, Y., and Haffner, P. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86 (1998), pp. 2278–2324.
[44] Lemaignan, S., Edmunds, C. E. R., Senft, E., and Belpaeme, T. "The PInSoRo dataset: Supporting the data-driven study of child-child and child-robot social dynamics". In: *PLOS ONE* 13 (2018), pp. 1–19.
[45] Lemaignan, S., Edmunds, C. E., Senft, E., and Belpaeme, T. "The PInSoRo dataset: Supporting the data-driven study of child-child and child-robot social dynamics". In: *PloS one* 13 (2018), e0205999.

[46] Lemaignan, S., Garcia, F., Jacq, A., and Dillenbourg, P. "From real-time attention assessment to "with-me-ness" in human-robot interaction". In: *Proc. HRI*. 2016.
[47] Lin, M. "Network in network". In: *arXiv preprint arXiv:1312.4400* (2013).
[48] Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., et al. "Swin transformer: Hierarchical vision transformer using shifted windows". In: *Proc. ICCV*. 2021.
[49] Liu, Z., Mao, H., Wu, C.-Y., Feichtenhofer, C., Darrell, T., et al. "A convnet for the 2020s". In: *Proc. CVPR*. 2022, pp. 11976–11986.
[50] Lord, C., Rutter, M., Dilavore, P. C., Risi, S., Gotham, K., et al. *ADOS-2: Autism diagnostic observation schedule*. Hogrefe., 2008.
[51] Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., et al. "Mediapipe: A framework for building perception pipelines". In: *arXiv preprint arXiv:1906.08172* (2019).
[52] Mingxing, T. and Quoc, V. L. "Efficientnet: Rethinking model scaling for convolutional neural networks". In: *arXiv preprint arXiv:1905.11946* 1 (2019).
[53] Mundy, P. C. and Acra, C. F. "Joint attention, social engagement, and the development of social competence". In: *The Development of Social EngagementNeurobiological Perspectives* (2006), pp. 81–117.
[54] Nobakht, A. and Gholamhosseinzadeh, M. "Enhancing Service Quality in Healthcare Systems Through Deep Learning". In: *AISP*. IEEE. 2024.
[55] Non-Verbal Communication, R. S. (B. S. G. on et al. *Non-verbal communication*. Cambridge University Press, 1972.
[56] O'Shea, T. J., Hitefield, S., and Corgan, J. "End-to-end radio traffic sequence recognition with deep recurrent neural networks". In: *arXiv preprint arXiv:1610.00564* (2016).
[57] Palmero, C., Selva, J., Bagheri, M. A., and Escalera, S. *Recurrent CNN for 3D Gaze Estimation using Appearance and Shape Cues*. 2018.
[58] Papaeliou, C. F., Fryssira, H., Kodakos, A., Kaila, M., Benaveli, E., et al. "Nonverbal communication, play, and language in Greek young children with Williams syndrome". In: *Child Neuropsychology* 17 (2011), pp. 225–241.
[59] Papailiou, C., Polemikos, N., Fryssira, H., Kontakos, A., Kaila, M., et al. "Joint attention and language development in toddlers with Down syndrome." In: *Psychology: The Journal of the Hellenic Psychological Society* (2011).
[60] Parkhi, O., Vedaldi, A., and Zisserman, A. "Deep face recognition". In: 2015, pp. 1–12.
[61] Parmar, P. and Tran Morris, B. "Learning to Score Olympic Events". In: *Proc. CVPR*. 2017.
[62] Resnick, E. "Defining engagement". In: *Journal of International Affairs* (2001), pp. 551–566.
[63] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. "Learning internal representations by error propagation". In: 1986.
[64] Saha, S. *A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way*. 2020.
[65] Simonyan, K. and Zisserman, A. "Very deep convolutional networks for large-scale image recognition". In: *arXiv preprint arXiv:1409.1556* (2014).
[66] Smith, B. A., Yin, Q., Feiner, S. K., and Nayar, S. K. "Gaze locking: passive eye contact detection for human-object interaction". In: *Proceedings of the 26th annual ACM symposium on User interface software and technology*. 2013.
[67] Stewart, M. *What's the Role of Weights and Bias in a Neural Network?* 2021.
[68] Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. "Inception-v4, inception-resnet and the impact of residual connections on learning". In: *Proc. AAAI*. 2017.
[69] Taheri, A., Alemi, M., Meghdari, A., Pouretemad, H., and Holderread, S. "Clinical application of humanoid robots in playing imitation games for autistic children in Iran". In: *Procedia-Social and Behavioral Sciences* 176 (2015), pp. 898–906.
[70] Tariq, S., Baber, S., Ashfaq, A., Ayaz, Y., Naveed, M., et al. "Interactive therapy approach through collaborative physical play between a socially assistive humanoid robot and children with autism spectrum disorder". In: *Proc. ICSR*. 2016.
[71] Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. "Learning Spatiotemporal Features With 3D Convolutional Networks". In: *Proc. ICCV*. 2015.
[72] Tran, D., Bourdev, L., Fergus, R., Torresani, L., and Paluri, M. "Learning spatiotemporal features with 3d convolutional networks". In: *Proc. ICCV*. 2015.
[73] Vaswani, A. "Attention is all you need". In: *Advances in Neural Information Processing Systems* (2017).

[74] Wainer, J., Robins, B., Amirabdollahian, F., and Dautenhahn, K. "Using the humanoid robot KASPAR to autonomously play triadic games and facilitate collaborative play among children with autism". In: *IEEE Transactions on Autonomous Mental Development* 6 (2014), pp. 183–199.

[75] Wang, L., Xiong, Y., Wang, Z., Qiao, Y., Lin, D., et al. *Temporal Segment Networks: Towards Good Practices for Deep Action Recognition*. 2016.

[76] Wang, P., Chen, P., Yuan, Y., Liu, D., Huang, Z., et al. "Understanding convolution for semantic segmentation". In: *Proc. WACV*. 2018.

[77] Whitehouse, A. J., Varcin, K. J., Pillar, S., Billingham, W., Alvares, G. A., et al. "Effect of preemptive intervention on developmental outcomes among infants showing early signs of autism: A randomized clinical trial of outcomes to diagnosis". In: *JAMA pediatrics* 175 (2021), e213298–e213298.

[78] Wu, C., Liaqat, S., Helvaci, H., Chcung, S.-c. S., Chuah, C.-N., et al. "Machine learning based autism spectrum disorder detection from videos". In: *Proc. HEALTHCOM*. 2021.

[79] Xiao, Y., Yuan, Q., Jiang, K., Jin, X., He, J., et al. "Local-global temporal difference learning for satellite video super-resolution". In: *IEEE Transactions on Circuits and Systems for Video Technology* (2023).

[80] Xie, N., Liu, Z., Li, Z., Pang, W., and Lu, B. "Student engagement detection in online environment using computer vision and multi-dimensional feature fusion". In: *Multimedia Systems* (2023), pp. 1–19.

[81] Xie, S., Girshick, R., Dollár, P., Tu, Z., and He, K. "Aggregated residual transformations for deep neural networks". In: *Proc. CVPR*. 2017.

[82] Yang, S., Wang, X., Li, Y., Fang, Y., Fang, J., et al. "Temporally efficient vision transformer for video instance segmentation". In: *Proc. CVPR*. 2022.

[83] Yi, D., Lei, Z., Liao, S., and Li, S. Z. "Learning face representation from scratch". In: *arXiv preprint arXiv:1411.7923* (2014).

[84] Yun, W.-H., Lee, D., Park, C., Kim, J., and Kim, J. "Automatic Recognition of Children Engagement from Facial Video Using Convolutional Neural Networks". In: *IEEE Transactions on Affective Computing* 11 (2020), pp. 696–707.

[85] Zhang, X., Sugano, Y., Fritz, M., and Bulling, A. "Appearance-based gaze estimation in the wild". In: *Proc. CVPR*. 2015.

[86] Zwaigenbaum, L., Bauman, M. L., Stone, W. L., Yirmiya, N., Estes, A., et al. "Early identification of autism spectrum disorder: recommendations for practice and research". In: *Pediatrics* 136 (2015), S10–S40.