



Πανεπιστήμιο Δυτικής Μακεδονίας  
Πολυτεχνική Σχολή  
Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών  
Υπολογιστών

---

**Διάγνωση Διαταραχής Αυτιστικού Φάσματος με  
την χρήση τεχνολογίας Eye-tracking και  
μηχανικής μάθησης**

---

*Διπλωματική Εργασία*

Καλοφορίδης Νικόλαος

Επιβλέπων Καθηγητής  
Φραγκούλης Γεώργιος

Κοζάνη 2024





University of Western Macedonia  
School of Engineering  
Department of Electrical and Computer Engineering

---

**Diagnosis of Autism Spectrum Disorder using  
Eye-tracking Technology and Machine  
Learning**

---

*Diploma Thesis*

Kaloforidis Nikolaos

Supervising Professor

Fragulis Georgios

Kozani 2024

## Δήλωση Πνευματικών Δικαιωμάτων

Δηλώνω ρητά ότι, σύμφωνα με το άρθρο 8 του Ν. 1599/1986 και τα άρθρα 2,4,6 παρ. 3 του Ν. 1256/1982, η παρούσα Διπλωματική Εργασία με τίτλο

“ Διάγνωση Διαταραχής Αυτιστικού Φάσματος με την χρήση τεχνολογίας Eye-tracking και μηχανικής μάθησης

*Diagnosis of Autism Spectrum Disorder using Eye-tracking Technology and Machine Learning*

”

καθώς και τα ηλεκτρονικά αρχεία και πηγαίοι κώδικες που αναπτύχθηκαν ή τροποποιήθηκαν στα πλαίσια αυτής της εργασίας και αναφέρονται ρητώς μέσα στο κείμενο που συνοδεύουν, και η οποία έχει εκπονηθεί στο Τμήμα Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Πανεπιστημίου Δυτικής Μακεδονίας, υπό την επίβλεψη του μέλους του Τμήματος κ. Γεωργίου Φραγκούλη

αποτελεί αποκλειστικά προϊόν προσωπικής εργασίας και δεν προσβάλλει κάθε μορφής πνευματικά δικαιώματα τρίτων και δεν είναι προϊόν μερικής ή ολικής αντιγραφής, οι πηγές δε που χρησιμοποιήθηκαν περιορίζονται στις βιβλιογραφικές αναφορές και μόνον. Τα σημεία όπου έχω χρησιμοποιήσει ιδέες, κείμενο, αρχεία ή / και πηγές άλλων συγγραφέων, αναφέρονται ευδιάκριτα στο κείμενο με την κατάλληλη παραπομπή και η σχετική αναφορά περιλαμβάνεται στο τμήμα των βιβλιογραφικών αναφορών με πλήρη περιγραφή.

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

Copyright (C) Ονοματεπώνυμο Φοιτητή & Επιβλέποντα/ες, Έτος, Πόλη

Copyright (C) Καλοφορίδης Νίκος, Φραγκούλης Γεώργιος , 2024 , Κοζάνη



## ***Ευχαριστίες***

Θα ήθελα να εκφράσω την ειλικρινή μου ευγνωμοσύνη σε όλους όσοι συνέβαλαν στην ολοκλήρωση αυτής της διπλωματικής εργασίας.

Πρώτα και πάνω απ' όλα, ευχαριστώ την οικογένειά μου για την αδιάκοπη στήριξή τους σε όλη τη διάρκεια των σπουδών μου. Η υπομονή, η αγάπη και η εμπύχωση που μου προσέφεραν ήταν αναντικατάστατες και αποτελούν τον λόγο που κατάφερα να φτάσω μέχρι εδώ.

Θα ήθελα επίσης να εκφράσω την ευγνωμοσύνη μου στον καθηγητή μου, Γεώργιο Φραγκούλη, ο οποίος με κατεύθυνε και με ενέπνευσε καθ' όλη τη διάρκεια της εργασίας. Η πολύτιμη καθοδήγησή του, οι γνώσεις του και η υποστήριξή του υπήρξαν καθοριστικές για την ανάπτυξη και την ολοκλήρωση της έρευνάς μου.

Τέλος, θα ήθελα να ευχαριστήσω ιδιαίτερα τον Κωνσταντίνο-Φίλιππο Κόλλια, ο οποίος με καθοδήγησε με επιμονή και αφοσίωση σε κάθε στάδιο αυτής της διαδικασίας. Η εμπειρία του και οι συμβουλές του ήταν ανεκτίμητες, βοηθώντας με να βελτιώσω την εργασία μου και να αναπτύξω την ακαδημαϊκή μου σκέψη.

Σε όλους όσοι με στήριξαν με οποιονδήποτε τρόπο σε αυτό το ταξίδι, τους ευχαριστώ από καρδιάς.

## ***Περίληψη***

Η πρόωμη διάγνωση της Διαταραχής Αυτιστικού Φάσματος (ΔΑΦ) είναι πολύ σημαντική για την κατανόηση των αναγκών αλλά και την καλύτερη αντιμετώπιση των προβλημάτων των ατόμων με ΔΑΦ. Σε αυτή τη μελέτη χρησιμοποιώντας δεδομένα που συλλέχθηκαν από eye tracker εκπαιδεύτηκαν 5 μοντέλα μηχανικής μάθησης για την κατηγοριοποίηση ατόμων ως άτομα με ΔΑΦ ή νευροτυπικά. Η εκπαίδευση των μοντέλων έδειξε υψηλά ποσοστά σε recall και AUC με τα καλύτερα αποτελέσματα να παρουσιάζονται από τον Random Forest σε συνδυασμό με Convolutional filters (recall=90%, AUC=88%). Η δοκιμή του αλγορίθμου σε πραγματικά δεδομένα που συλλέχθηκαν μέσω ενός eye-tracker, ειδικά σχεδιασμένου για την έρευνα, έδωσε εξίσου θετικά αποτελέσματα, ενισχύοντας την αξιοπιστία της προτεινόμενης μεθόδου. Περιλαμβάνονται επίσης περιορισμοί και συστάσεις για μελλοντική έρευνα.

## ***Abstract***

The early diagnosis of Autism Spectrum Disorder (ASD) is very important for understanding the needs and addressing the challenges faced by individuals with ASD. In this study, data collected from an eye tracker were used to train five machine learning models for the categorization of individuals as either having ASD or being neurotypical. The training of the models demonstrated high recall and AUC rates, with the best results achieved by the Random Forest algorithm in combination with Convolutional Filters (recall = 90%, AUC = 88%). The testing of the algorithm on real data collected through an eye tracker specifically designed for this research also yielded equally positive results, reinforcing the reliability of the proposed method. Limitations and recommendations for future research are also included.



## Πίνακας Περιεχομένων

1	Εισαγωγή.....	12
1.1	Διαταραχή Αυτιστικού Φάσματος .....	12
1.2	Σημασία της πρόωρης διάγνωσης της ΔΑΦ .....	13
1.3	Τεχνολογία ανίχνευσης ματιών και εφαρμογές της .....	13
1.4	Μετρικές στην ανίχνευση ματιών για την διάγνωση ΔΑΦ .....	14
1.5	Βασικές τεχνικές και μηχανισμοί .....	14
1.6	Αλγόριθμοι μηχανικής μάθησης στην έρευνα της ΔΑΦ .....	16
2	Βιβλιογραφική Ανασκόπηση .....	19
3	Σκοπός της Έρευνας.....	21
4	Μεθοδολογία .....	22
4.1	Περιγραφή Dataset.....	22
4.2	Προ-επεξεργασία δεδομένων .....	25
4.3	Μετρικά αξιολόγησης μοντέλων .....	26
4.4	Αλγόριθμοι .....	28
4.5	Δημιουργία Eye-tracker.....	30
4.5.1	Εξαγωγή χαρακτηριστικών προσώπου.....	30
4.5.2	Calibration του Eye-tracker .....	31
4.5.3	Διαδικασία εντοπισμού κίνησης των ματιών .....	32
4.5.4	Υπολογισμός μαθηματικών σχέσεων .....	32
4.5.5	Οπτικοποίηση αποτελεσμάτων.....	33
5	Ανάλυση Κώδικα.....	34
5.1	Αρχικό παράθυρο.....	34
5.2	Calibration του Eye-tracker .....	35
5.3	Επιλογή κατάλληλων σημείων .....	37
5.4	Υπολογισμός ταχύτητας, επιτάχυνσης, jerk.....	38
5.5	Κυρίως κώδικας .....	40
6	Αποτελέσματα .....	44
6.1	Επιλογή μοντέλου μηχανικής μάθησης .....	44
6.2	Παραμετροποίηση μοντέλου.....	47
7	Συμπεράσματα .....	53
8	Επίλογος .....	55
9	Αναφορές.....	55

## Πίνακας Εικόνων

ΕΙΚΟΝΑ 1 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ RANDOM FOREST .....	17
ΕΙΚΟΝΑ 2 ΠΑΡΑΔΕΙΓΜΑ ΑΛΓΟΡΙΘΜΟΥ SUPORT VECTOR MACHINE .....	17
ΕΙΚΟΝΑ 3 ΠΑΡΑΔΕΙΓΜΑ ΤΑΞΙΝΟΜΙΤΗ BAYER .....	18
ΕΙΚΟΝΑ 4 ΠΑΡΑΔΕΙΓΜΑ MLP ΔΙΚΤΥΟΥ .....	18
ΕΙΚΟΝΑ 5 ΠΑΡΑΔΕΙΓΜΑ CNN ΔΙΚΤΥΟΥ .....	19
ΕΙΚΟΝΑ 6 ΠΑΡΑΔΕΙΓΜΑ RNN ΔΙΚΤΥΟΥ .....	19
ΕΙΚΟΝΑ 7 ΚΑΤΑΝΟΜΗ ΦΥΛΟΥ ΣΤΟ DATASET .....	22
ΕΙΚΟΝΑ 8 ΚΑΤΑΝΟΜΗ ΝΕΥΡΟΤΥΠΙΚΩΝ / ΑΤΟΜΩΝ ΜΕ ΔΑΦ ΣΤΟ DATASET .....	23
ΕΙΚΟΝΑ 9 ΗΛΙΚΙΑΚΗ ΚΑΤΑΝΟΜΗ ΤΩΝ ΣΥΜΜΕΤΕΧΟΝΤΩΝ .....	23
ΕΙΚΟΝΑ 10 ΔΕΙΚΤΗΣ CARS ΤΩΝ ΣΥΜΜΕΤΕΧΟΝΤΩΝ.....	24
ΕΙΚΟΝΑ 11 ΕΥΕ-TRACKING PATTERN ΑΤΟΜΟΥ ΜΕ ΔΑΦ (ΑΡΙΣΤΕΡΑ) ΚΑΙ ΝΕΥΡΟΤΥΠΙΚΟΥ ΑΤΟΜΟΥ (ΔΕΞΙΑ).....	25
ΕΙΚΟΝΑ 12 ΚΑΤΑΝΟΜΗ ΕΤΙΚΕΤΩΝ ΣΤΟ DATASET.....	25
ΕΙΚΟΝΑ 13 ΠΑΡΑΔΕΙΓΜΑ ΠΙΝΑΚΑ ΣΥΓΧΗΣΗΣ ΔΥΑΔΙΚΟΥ ΜΟΝΤΕΛΟΥ .....	27
ΕΙΚΟΝΑ 14 ΚΑΜΠΥΑΗ ROC ΙΔΑΝΙΚΟΥ ΜΟΝΤΕΛΟΥ .....	27
ΕΙΚΟΝΑ 15 ΚΑΜΠΥΑΗ ROC ΠΡΑΓΜΑΤΙΚΟΥ ΜΟΝΤΕΛΟΥ.....	27
ΕΙΚΟΝΑ 16 ΔΙΑΦΟΡΕΤΙΚΕΣ ΣΥΝΑΡΤΗΣΕΙΣ ΕΝΕΡΓΟΠΟΙΗΣΗΣ .....	28
ΕΙΚΟΝΑ 17 ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΚΠΑΙΔΕΥΣΗΣ ANN .....	29
ΕΙΚΟΝΑ 18 ΑΠΟΤΕΛΕΣΜΑΤΑ ΕΚΠΑΙΔΕΥΣΗΣ CNN .....	30
ΕΙΚΟΝΑ 19 ΠΑΡΑΔΕΙΓΜΑ ΠΛΕΓΜΑΤΟΣ ΠΡΟΣΩΠΟΥ .....	31
ΕΙΚΟΝΑ 20 ΠΑΡΑΔΕΙΓΜΑ ΔΗΜΙΟΥΡΓΗΜΕΝΗΣ ΕΙΚΟΝΑΣ .....	34
ΕΙΚΟΝΑ 21 ΠΑΡΑΔΕΙΓΜΑ GUI ΤΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ .....	34
ΕΙΚΟΝΑ 22 SNAPSHOT ΑΠΟ ΤΗΝ ΔΗΜΙΟΥΡΓΙΑ ΤΟΥ GUI.....	35
ΕΙΚΟΝΑ 23 SNAPSHOT ΣΥΝΑΡΤΗΣΗΣ SHOW CALIBRATION.....	36
ΕΙΚΟΝΑ 24 SNAPSHOT ΣΥΝΑΡΤΗΣΗΣ CALIBRATION ΤΟΥ ΕΥΕ-TRACKER.....	37
ΕΙΚΟΝΑ 25 SNAPSHOT ΣΥΝΑΡΤΗΣΗΣ ΕΠΙΛΟΓΗΣ ΣΗΜΕΙΩΝ.....	38
ΕΙΚΟΝΑ 26 SNAPSHOT ΣΥΝΑΡΤΗΣΗΣ ΥΠΟΛΟΓΙΣΜΟΥ ΤΑΧΥΤΗΤΑΣ .....	39
ΕΙΚΟΝΑ 27 SNAPSHOT ΣΥΝΑΡΤΗΣΗΣ ΥΠΟΛΟΓΙΣΜΟΥ ΕΠΙΤΑΧΥΝΣΗΣ .....	39
ΕΙΚΟΝΑ 28 SNAPSHOT ΣΥΝΑΡΤΗΣΗΣ ΥΠΟΛΟΓΙΣΜΟΥ JERK .....	39
ΕΙΚΟΝΑ 29 SNAPSHOT 1 MAIN CODE .....	40
ΕΙΚΟΝΑ 30 SNAPSHOT 2 MAIN CODE .....	41
ΕΙΚΟΝΑ 31 SNAPSHOT 3 MAIN CODE .....	42
ΕΙΚΟΝΑ 32 SNAPSHOT 4 MAIN CODE .....	42
ΕΙΚΟΝΑ 33 ΠΑΡΑΔΕΙΓΜΑ ΑΠΟΤΕΛΕΣΜΑΤΟΣ.....	43
ΕΙΚΟΝΑ 34 ΜΕΤΡΙΚΑ ΤΩΝ ΔΙΑΦΟΡΕΤΙΚΩΝ ΑΛΓΟΡΙΘΜΩΝ.....	44
ΕΙΚΟΝΑ 35 ΚΑΜΠΥΑΗ ROC CNN .....	45
ΕΙΚΟΝΑ 37 ΚΑΜΠΥΑΗ ROC RANDOM FOREST .....	45
ΕΙΚΟΝΑ 36 ΚΑΜΠΥΑΗ ROC ANN .....	45
ΕΙΚΟΝΑ 38 ΚΑΜΠΥΑΗ ROC CONV RANDOM FOREST .....	45
ΕΙΚΟΝΑ 39 ΚΑΜΠΥΑΗ ROC SVM.....	45
ΕΙΚΟΝΑ 40 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ ANN RANDOM FOREST .....	ΕΙΚΟΝΑ 41 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ CONV RANDOM FOREST .....
ΕΙΚΟΝΑ 42 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ RANDOM FOREST	ΕΙΚΟΝΑ 43 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ SVM
	46
ΕΙΚΟΝΑ 44 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ CNN.....	46
ΕΙΚΟΝΑ 45 ΕΚΠΑΙΔΕΥΣΗ PRE-TRAINED ΜΟΝΤΕΛΟΥ ΣΤΑ ΔΕΛΟΜΕΝΑ .....	48
ΕΙΚΟΝΑ 46 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ FINE TUNED MODEL.....	49

<b>ΕΙΚΟΝΑ 47 ΚΑΜΠΥΑΗ ROC FINE TUNED MODEL.....</b>	<b>50</b>
<b>ΕΙΚΟΝΑ 48 ΠΙΝΑΚΑΣ ΣΥΓΧΥΣΗΣ HYPER PARAMETER TUNED MODEL .....</b>	<b>52</b>
<b>ΕΙΚΟΝΑ 49 ΚΑΜΠΥΑΗ ROC HYPER PARAMETER TUNED MODEL .....</b>	<b>52</b>

## **Πίνακας Πινάκων**

<b>ΠΙΝΑΚΑΣ 1 ΣΥΝΟΛΙΚΑ ΣΤΑΤΙΣΤΙΚΑ ΤΩΝ ΣΥΜΜΕΤΕΧΟΝΤΩΝ .....</b>	<b>24</b>
<b>ΠΙΝΑΚΑΣ 3 ΣΗΜΕΙΑ CALIBRATION.....</b>	<b>31</b>
<b>ΠΙΝΑΚΑΣ 4 ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ .....</b>	<b>44</b>
<b>ΠΙΝΑΚΑΣ 5 ΑΠΟΤΕΛΕΣΜΑΤΑ FINE TUNING.....</b>	<b>48</b>
<b>ΠΙΝΑΚΑΣ 6 ΠΛΕΓΜΑ ΑΝΑΖΗΤΗΣΗΣ ΠΑΡΑΜΕΤΡΩΝ .....</b>	<b>50</b>
<b>ΠΙΝΑΚΑΣ 7 ΒΕΛΤΙΣΤΕΣ ΠΑΡΑΜΕΤΡΟΙ.....</b>	<b>51</b>
<b>ΠΙΝΑΚΑΣ 8 ΑΠΟΤΕΛΕΣΜΑΤΑ HYPER PARAMETER TUNING .....</b>	<b>51</b>

# 1 Εισαγωγή

## 1.1 Διαταραχή Αυτιστικού Φάσματος

Η Διαταραχή Αυτιστικού Φάσματος (ΔΑΦ) είναι μια νευροαναπτυξιακή διαταραχή που επηρεάζει τον τρόπο με τον οποίο ένα άτομο αλληλοεπιδρά, μαθαίνει και συμπεριφέρεται. Χαρακτηρίζεται ως αναπτυξιακή διαταραχή, καθώς τα συμπτώματα συνήθως εμφανίζονται τα πρώτα τρία χρόνια της ζωής ενός ατόμου. Πρόκειται για μια διαταραχή φάσματος, καθώς εμφανίζεται διαφορετικά σε κάθε άτομο, κυμαίνοντας από ελαφριές έως σοβαρές περιπτώσεις.

Η ΔΑΦ τείνει να επηρεάζει τους άντρες περισσότερο από τις γυναίκες, με μια αναλογία περίπου 4 προς 1. Σε περιπτώσεις υψηλής λειτουργικότητας, η αναλογία αυξάνεται σε περίπου 10 προς 1, ενώ σε περιπτώσεις χαμηλότερης λειτουργικότητας, η αναλογία γίνεται περίπου 2 προς 1 [1].

Οι αιτίες που προκαλούν τον αυτισμό δεν είναι ακόμα πλήρως γνωστές. Επιστήμονες έχουν συσχετίσει τον αυτισμό με γενετικούς, αλλά και περιβαλλοντικούς παράγοντες. Ωστόσο, η εξέλιξη των ερευνών συνεχίζεται για την κατανόηση των βαθύτερων αιτιών και την ανάπτυξη πιο αποτελεσματικών προσεγγίσεων θεραπείας.

Σύμφωνα με το κλινικό εγχειρίδιο DSM-5 [2], τα άτομα με ΔΑΦ χαρακτηρίζονται από:

- **Δυσκολίες στην αναγνώριση και στην επίδειξη συναισθημάτων:** Τα άτομα με ΔΑΦ μπορεί να έχουν προβλήματα στο να αναγνωρίσουν και να εκφράσουν συναισθήματα.
- **Δυσκολίες στην ανάπτυξη και συντήρηση των ανθρώπινων σχέσεων:** Η δημιουργία και διατήρηση σχέσεων με άλλους ανθρώπους μπορεί να αποτελεί πρόκληση για τα άτομα με ΔΑΦ.
- **Δυσκολίες στην λεκτική και μη λεκτική επικοινωνία:** Συμπεριλαμβάνουν έλλειψη οπτικής επαφής, χρήση χειρονομιών και γλωσσικές "ανωμαλίες".
- **Δυσκολίες στην κατανόηση κοινωνικών κανόνων:** Οι άνθρωποι με ΔΑΦ μπορεί να μην κατανοούν πλήρως τους κοινωνικούς κανόνες.
- **Επαναλαμβανόμενα μοτίβα στην συμπεριφορά αλλά και στην καθημερινότητά τους:** Συχνά εμφανίζουν επαναλαμβανόμενα μοτίβα συμπεριφοράς και προτίμηση για σταθερότητα στην καθημερινή τους ζωή.
- **Στερεοτυπική μηχανοποιημένες κινήσεις εκφράσεις:** Μπορεί να εκφράζουν στερεοτυπικές ή μηχανοποιημένες κινήσεις και εκφράσεις.
- **Υπέρ ή υπό ευαισθησία στις αισθητηριακές εισροές και στις αισθητηριακές πτυχές του περιβάλλοντος:** Μπορεί να αντιδρούν υπερβολικά ή με ανεπαρκή αντίδραση στις αισθητηριακές εισροές και στα αισθητηριακά χαρακτηριστικά του περιβάλλοντος.

## 1.2 Σημασία της πρόωρης διάγνωσης της ΔΑΦ

Ο αυτισμός ως νευροαναπτυξιακή διαταραχή που επηρεάζει την συμπεριφορά αλλά και την κοινωνική αλληλεπίδραση του ατόμου απαιτεί έγκυρη διάγνωση έτσι ώστε να μεγιστοποιηθούν τα οφέλη της παρέμβασης. Ιστορικά, τα παιδιά με ΔΑΦ ιδρύματοποιούνταν σε κακές συνθήκες. Όμως με την πρόοδο στις στρατηγικές παρεμβάσεις των τελευταίων δεκαετιών έχει παρατηρηθεί σημαντική βελτίωση στα αποτελέσματα με πολλά παιδιά να έχουν την δυνατότητα να ενταχθούν σε κανονικά εκπαιδευτικά πλαίσια και περίπου το 25 % χάνει τελείως την διάγνωση [3],[4].

Στο παρελθόν οι Lovaas [5] και Prizant [6] δείξαν ότι ακόμα και με τις καλύτερες στρατηγικές παρεμβάσεις περίπου το 50% των παιδιών με ΔΑΦ παρέμεναν μη-λεκτικά σε όλη την διάρκεια της ζωής τους. Ενώ αντίθετα πιο σύγχρονες μελέτες [7] έδειξαν ότι περίπου το 10% των ατόμων με ΔΑΦ παραμένει μη λεκτικό σε όλη την διάρκεια της ζωής του. Επιπλέον τα παιδιά που δέχονται βοήθεια σε προσχολικές ηλικίες είναι πιο πιθανό να αναπτύξουν λεκτική επικοινωνία από τα άτομα που δέχονται βοήθεια σε μεγαλύτερη ηλικία [8],[9],[10].

Οι πρόοδοι στις παρεμβάσεις έχουν επίσης οδηγήσει σε μια καλύτερη κατανόηση των αναγκών των παιδιών με αυτισμό, επιτρέποντας την ανάπτυξη πιο εξατομικευμένων προγραμμάτων που προάγουν την κοινωνική και γνωστική ανάπτυξη. Η σημασία της έγκαιρης διάγνωσης και παρέμβασης αλλά και της συνεχιζόμενης υποστήριξης δεν μπορεί να υποτιμηθεί, καθώς αυτά τα μέτρα όχι μόνο βελτιώνουν την ποιότητα ζωής των ατόμων με αυτισμό αλλά και μειώνουν το άγχος και την αβεβαιότητα των οικογενειών τους.

## 1.3 Τεχνολογία ανίχνευσης ματιών και εφαρμογές της

Η τεχνολογία eye-tracking (ET) είναι μια μέθοδος παρακολούθησης των κινήσεων των ματιών, καταγράφοντας πού κοιτάζει ένα άτομο, για πόσο χρόνο και πώς τα μάτια του κινούνται. Μετράει επιπλέον τα σημεία του βλέμματος του ματιού σε σχέση με το πρόσωπο αλλά και το πώς αντιδράει η κόρη του ματιού σε διαφορετικά ερεθίσματα. Για να υλοποιηθεί η τεχνολογία ET αρχικά χρησιμοποιούνται οι συσκευές καταγραφής (κάμερες και αισθητήρες) έτσι ώστε να καταγράψουν τις κινήσεις των ματιών. Στην συνέχεια χρησιμοποιούνται υπέρυθρες ακτίνες οι οποίες αντανακλώνται στα μάτια έτσι ώστε να καταγραφούν οι αντανάκλασεις στον κερατοειδή και να επιτευχθεί πιο ακριβής μέτρηση της κίνησης των ματιών. Τέλος τα δεδομένα συλλέγονται και επεξεργάζονται από ειδικούς αλγόριθμους έτσι ώστε να υπολογιστεί η κατεύθυνση του βλέμματος και οι περιοχές ενδιαφέροντος (ROI). Η τεχνολογία ET αξιοποιείται στον τομέα της ψυχολογίας και της νευροεπιστήμης, για την διάγνωση νευρολογικών διαταραχών όπως η ΔΑΦ [11] και η νόσος του Πάρκινσον [12]. Επιπλέον μπορεί να χρησιμοποιηθεί στον τομέα του μάρκετινγκ για να βοηθήσει τους ειδικούς να κατανοήσουν ποια στοιχεία των διαφημίσεων τραβούν παραπάνω την προσοχή και ποιες συσκευασίες ελκύουν παραπάνω το ενδιαφέρον των πελατών [13]. Μια ακόμα εφαρμογή είναι στον τομέα της διδασκαλίας ως μέσο αξιολόγησης της επίδρασης των μαθητών με το διδακτικό υλικό [14]. Τέλος είναι συνηθισμένη η χρήση της στον τομέα της

ψυχαγωγίας (ταινίες, βιντεοπαιχνίδια κλπ.) παρέχοντας στους καταναλωτές πιο αλληλεπιδραστικές εμπειρίες [15].

## 1.4 Μετρικές στην ανίχνευση ματιών για την διάγνωση ΔΑΦ

- **Στερεώση (Fixation):** Ο χρόνος εστίασης σε μια συγκεκριμένη περιοχή (π.χ. πρόσωπο ή αντικείμενο).
- **Λανθάνουσα περίοδος πρώτης στερεώσης (First Fixation Latency):** Ο χρόνος που χρειάζεται για να πραγματοποιηθεί η πρώτη στερεώση μετά την εμφάνιση του ερεθίσματος.
- **Εύρος Σακκάδων (Saccades Amplitude):** Η απόσταση που καλύπτουν τα μάτια μεταξύ διαδοχικών στερεώσεων.
- **Ταχύτητα Σακκάδων (Saccades Velocity):** Η ταχύτητα κίνησης των ματιών μεταξύ των στερεώσεων.
- **Διαστολή κόρης (Pupil dilation):** Μεταβολές στο μέγεθος της κόρης του ματιού.
- **Πρότυπα βλέμματος (Gaze Patterns):** Ανάλυση της τοποθεσίας και της χρονικής διάρκειας του βλέμματος σε συγκεκριμένες περιοχές ενδιαφέροντος.
- **Εντροπία Μετάβασης Βλέμματος (Gaze Transition Entropy):** Η τυχαιότητα των μεταβάσεων του βλέμματος μεταξύ διαφορετικών περιοχών ενδιαφέροντος.
- **Διαδρομή σάρωσης (Scan Path):** Η αλληλουχία στερεώσεων και σακκάδων σε μια οπτική σκηνή.
- **Διάρκεια βλέμματος στα μάτια (On-Eye Gaze Duration):** Ειδική προσοχή στην περιοχή των ματιών όταν κοιτάζουν πρόσωπα. Τα άτομα με ΔΑΦ συνήθως αφιερώνουν λιγότερο χρόνο κοιτώντας στα μάτια.
- **Μετατόπιση Βλέμματος (Gaze Shifting):** Η ικανότητα μετατόπισης του βλέμματος μεταξύ αντικειμένων ή προσώπων σε κοινωνικές αλληλεπιδράσεις. Τα άτομα με ΔΑΦ συχνά αντιμετωπίζουν δυσκολίες στη μετατόπιση του βλέμματος ως απόκριση σε κοινωνικά σήματα.
- **Μετρήσεις κοινωνικής προσοχής (Social attention Metrics):** Προσοχή σε κοινωνικά ερεθίσματα έναντι μη κοινωνικών ερεθισμάτων. Τα άτομα με ΔΑΦ μπορεί να παρουσιάζουν μειωμένη προσοχή σε κοινωνικά ερεθίσματα, όπως άνθρωποι ή ζώα, σε σύγκριση με αντικείμενα ή στοιχεία του φόντου.

## 1.5 Βασικές τεχνικές και μηχανισμοί

Η τεχνολογία ανίχνευσης κινήσεων των ματιών αποτελεί μια από τις κυρίαρχες τεχνολογίες παρακολούθησης και ανάλυσης της ανθρώπινης συμπεριφοράς. Η καταγραφή και η ανάλυση των κινήσεων μπορεί να δώσει πληροφορίες για τον τρόπο που ένα άτομο επεξεργάζεται τα οπτικά ερεθίσματα, αλληλοεπιδρά με το περιβάλλον και λαμβάνει αποφάσεις σε πραγματικό χρόνο.

Η ανίχνευση κινήσεων ματιών βασίζεται σε συγκεκριμένες τεχνικές οι οποίες βοηθούν στην λεπτομερή καταγραφή και ανάλυση των κινήσεων. Οι τρεις κυρίαρχες τεχνικές

που χρησιμοποιούνται είναι οι βιντεο-οφθαλμολογία (**VOG**), βιντεο-βασισμένη υπέρυθρη αντανάκλαση κόρης-κερατοειδούς (**PCR**) και ηλεκτρο-οφθαλμογραφία (**EOG**) [16].

Η βιντεο-οφθαλμολογία χρησιμοποιεί ορατό φως και βιντεοκάμερες για την καταγραφή των ματιών. Για να πραγματοποιηθεί η καταγραφή μπορεί να γίνει απομακρυσμένα ή κεφαλοδεσμικά (head mounted). Στα απομακρυσμένα συστήματα, η κάμερα συνήθως τοποθετείται κάτω από την οθόνη. Αντίθετα στα κεφαλοδεσμικά συστήματα η κάμερα είναι ενσωματωμένη είτε στα γυαλιά καταγραφής είτε σε κάποιο κράνος που ο χρήστης καλείται να φορέσει. Εκτός από την κάμερα για την καταγραφή των ματιών είναι συνηθισμένο να χρησιμοποιείται και μια κάμερα καταγραφής της σκηνής δηλαδή μια κάμερα που παρακολουθεί το οπτικό περιβάλλον του χρήστη έτσι ώστε να γίνει πιο εύκολα η συσχέτιση του βλέμματος με την περιοχή ενδιαφέροντος του περιβάλλοντος. Σημαντικό ρόλο στην ακρίβεια της καταγραφής κατέχουν οι προδιαγραφές της κάμερας. Για την βέλτιστη ακρίβεια χρησιμοποιούνται κάμερες με μεγάλη ανάλυση και framerate.

Παρόλο που η VOG αποτελεί μια μη παρεμβατική τεχνική για τον χρήστη καθώς του παρέχει μερική ελευθερία κινήσεων, επηρεάζεται από τις συνθήκες φωτισμού, τυχόν αντανάκλασεις από τα γυαλιά του χρήστη ενώ η ακρίβεια μπορεί να μειωθεί και από τις κινήσεις του κεφαλιού.

Η βιντεο-βασισμένη υπέρυθρη αντανάκλαση κόρης-κερατοειδούς για να μειώσει το πρόβλημα της ακρίβειας που δημιουργείται από το ορατό φως και τις κινήσεις του κεφαλιού, χρησιμοποιεί ένα σημείο αναφοράς το οποίο λέγεται αντανάκλαση κερατοειδή. Για να δημιουργηθεί αυτό το σημείο χρησιμοποιείται μια πηγή τεχνητού υπέρυθρου φωτός (IR) η οποία κατευθύνεται στην κόρη του ματιού με δύο διαφορετικές διατάξεις (on-axis ή off-axis). Στην on-axis διάταξη η υπέρυθρη πηγή βρίσκεται ευθυγραμμισμένη με την οπτική γωνία της κάμερας φωτίζοντας έτσι την κόρη του ματιού και δημιουργώντας έτσι το φαινόμενο bright-pupil το οποίο καθιστά πιο εύκολη την αναγνώριση της κόρης από το λογισμικό του υπολογιστή. Στην off-axis διάταξη η υπέρυθρη πηγή τοποθετείται έξω από το οπτικό πεδίο της κάμερας δημιουργώντας εικόνες dark pupil. Καθώς το υπέρυθρο φως δεν είναι ορατό στο ανθρώπινο μάτι και οι δύο διατάξεις επιτυγχάνουν καλό φωτισμό του ματιού χωρίς να επηρεάζεται η διάμετρος της κόρης του ματιού και χωρίς να διαταράσσεται η οπτική του χρήστη [17]. Για να υπολογιστεί η αντανάκλαση κερατοειδούς συγκρίνεται η αντανάκλαση από την υπέρυθρη πηγή με το κέντρο της κόρης. Καθώς η θέση της αντανάκλασης του κερατοειδούς παραμένει σχετικά σταθερή κατά την διάρκεια της κίνησης του ματιού μπορεί να υπολογιστεί ο άξονας του βλέμματος συγκρίνοντας την κίνηση της κόρης του ματιού με την αντανάκλαση του κερατοειδούς.

Παρόλη την ακρίβεια που επιφέρει το υπέρυθρο φως στην ανίχνευση του βλέμματος αυτή η μέθοδος αντιμετωπίζει δυσκολίες σε περιβάλλοντα με μεταβλητό φωτισμό όπως οι εξωτερικοί χώροι.

Η ηλεκτροοφθαλμολογία βασίζεται στο γεγονός ότι το ανθρώπινο μάτι μπορεί να μοντελοποιηθεί ως ένα ηλεκτρικό δίπολο με θετικό πόλο τον κερατοειδή και αρνητικό των αμφιβληστροειδή. Αυτό το ηλεκτρικό δίπολο μας δίνει μια διαφορά δυναμικού η οποία χρησιμοποιείται για την καταγραφή κινήσεων των ματιών. Για να μετρηθεί η διαφορά δυναμικού χρησιμοποιούνται ηλεκτρόδια τοποθετημένα γύρω από κάθε μάτι και ένα ηλεκτρόδιο ως σημείο αναφοράς σε κάποιο σταθερό σημείο πχ στο μέτωπο. Με τις κινήσεις του αμφιβληστροειδή και του κερατοειδή συγκριτικά με το σημείο αναφοράς αλλάζει ο προσανατολισμός του δίπολου επιτρέποντας έτσι την καταγραφή των κινήσεων. Καθώς δεν είναι απαραίτητη η εικόνα του ματιού στην EOG αλλά μόνο το ηλεκτρικό σήμα που παράγει είναι ιδιαίτερα χρήσιμη εκεί που η video-based τεχνικές υστερούν δηλαδή σε περιβάλλοντα με μεταβλητό η και καθόλου φωτισμό. Επιπλέον καθώς η καταγραφή μπορεί να γίνει ακόμα και αν τα μάτια είναι κλειστά η EOG διαδεδομένη για μετρήσεις που παίρνονται κατά την διάρκεια του ύπνου πχ για την αναγνώριση των φάσεων REM όσο ένα άτομο κοιμάται [18]. Τέλος καθώς το μόνο που εξάγεται είναι ηλεκτρικό σήμα και όχι διάφορες βιντεοσκοπήσεις των ματιών η επεξεργασία αλλά και η αποθήκευση των αποτελεσμάτων είναι αρκετά πιο εύκολη καθιστώντας έτσι την EOG κατάλληλη για την διεξαγωγή πολύωρων πειραμάτων.

Παρόλα αυτά η EOG αποτελεί αρκετά παρεμβατική μέθοδο λόγω των ηλεκτροδίων που τοποθετούνται γύρω από το δέρμα του ματιού το οποίο μπορεί να καθιστά την διαδικασία του εντοπισμού άβολη για τον χρήστη. Επιπλέον τα σήματα που παράγονται αν και πιο εύκολα στην επεξεργασία τους από αυτά που παράγονται από video-based τεχνικές είναι και πιο ευαίσθητα σε θόρυβο και παρεμβολές ειδικά όταν το περιβάλλον καταγραφής δεν είναι σταθερό.

## 1.6 Αλγόριθμοι μηχανικής μάθησης στην έρευνα της ΔΑΦ

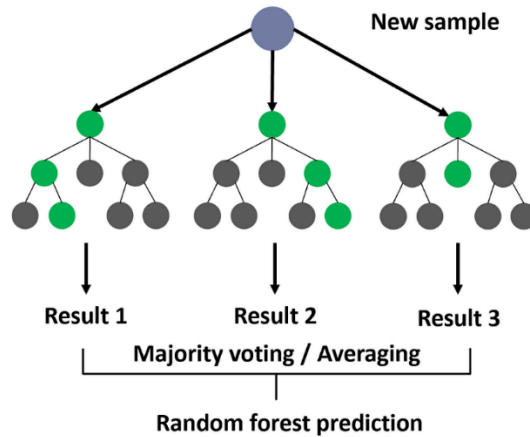
Η έρευνα για τη διαταραχή αυτιστικού φάσματος (ΔΑΦ) είναι ζωτικής σημασίας για την κατανόηση των πολλών πτυχών του ανθρώπινου εγκεφάλου και της συμπεριφοράς. Η εφαρμογή των αλγορίθμων μηχανικής μάθησης έχει δημιουργήσει νέους δρόμους για τη μελέτη, τη διάγνωση και τη διαχείριση του αυτισμού.

Οι αλγόριθμοι μηχανικής μάθησης επιτρέπουν την ανάλυση μεγάλου όγκου δεδομένων από διάφορες πηγές συμπεριλαμβανομένων των γενετικών, νευροφυσιολογικών και συμπεριφορικών δεδομένων. Με αυτό τον τρόπο, οι ερευνητές μπορούν να αναπτύξουν πιο ακριβείς διαγνωστικές και θεραπευτικές μεθόδους για την ΔΑΦ.

Οι αλγόριθμοι που χρησιμοποιούνται περισσότερο για την έρευνα αυτή είναι:

**Random forest (Εικόνα 1):** Είναι ένας από τους δημοφιλέστερους αλγορίθμους μηχανικής μάθησης για επίλυση προβλημάτων ταξινόμησης και παλινδρόμησης. Λειτουργεί συνδυάζοντας πολλά ασυσχέτιστα μεταξύ τους δέντρα απόφασης ( Decision Tree ) για πραγματοποιήσει μια εκτίμηση. Βασικός του στόχος είναι η μείωση της συσχέτισης μεταξύ των ταξινομητών που αποτελείται και την μείωση του φαινομένου overfitting.

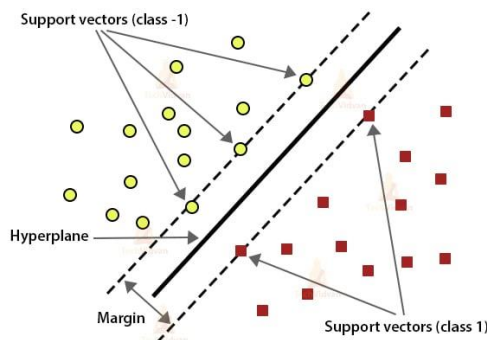




Εικόνα 1 Παράδειγμα αλγορίθμου Random forest

**Support Vector Machine (SVM) (Εικόνα 2):** Είναι ένας αλγόριθμος επιβλεπόμενης μάθησης ο οποίος ταξινομεί τα δεδομένα μας σε 2 κατηγορίες τοποθετώντας τα στον  $n$ -διάστατο χώρο και δημιουργώντας μια επιφάνεια η οποία χωρίζει τις δύο κατηγορίες.

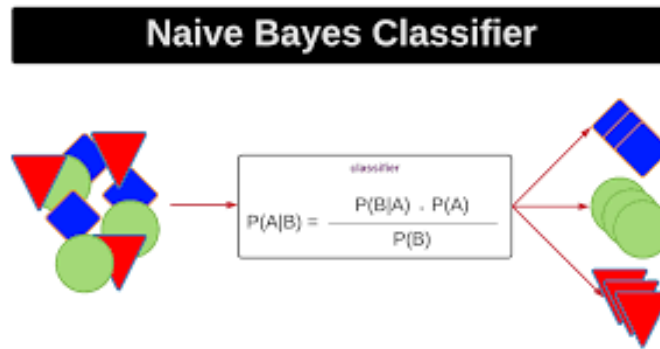
### Support Vector Machines



Εικόνα 2 Παράδειγμα αλγορίθμου Support Vector Machine

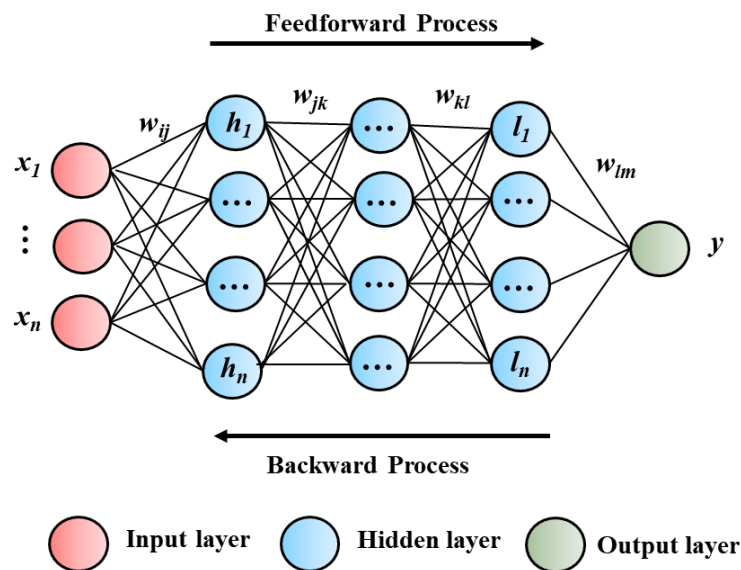
**Naïve Bayes (Εικόνα 3):** Αποτελεί έναν αλγόριθμο επιβλεπόμενης μάθησης που χρησιμοποιείται για ταξινόμηση με βάση το θεώρημα Bayes το οποίο κάνει προβλέψεις με βάση την πιθανότητα εμφάνισης του αντικειμένου.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$



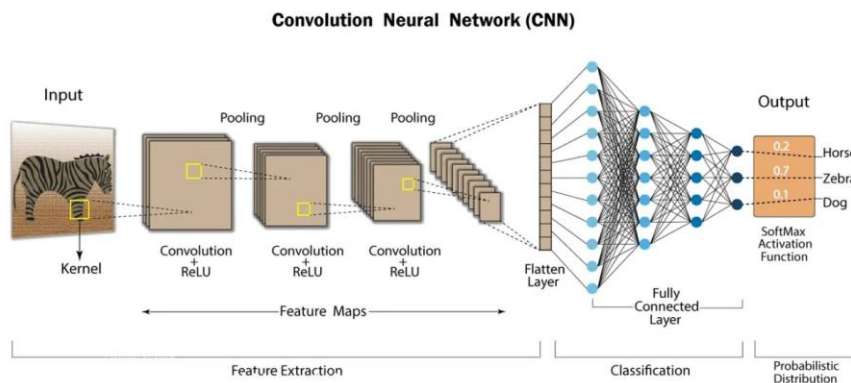
Εικόνα 3 Παράδειγμα ταξινόμητη Bayes

**Multi Layer Perceptron (Εικόνα 4):** Αποτελεί ένα είδος τεχνητού νευρωνικού δικτύου με πολλαπλά επίπεδα νευρώνων οι οποίοι είναι συνδεδεμένοι μεταξύ τους. Ένα MLP έχει το λιγότερο 3 διαδοχικά επίπεδα (είσοδος - κρυφά επίπεδο – έξοδος) όπου ο κάθε νευρώνας του επιπέδου συνδέεται με κάθε νευρώνα στο επόμενο επίπεδο με ένα βάρος το οποίο ρυθμίζει την σημασία της συνεισφοράς του νευρώνα στο επόμενο επίπεδο.



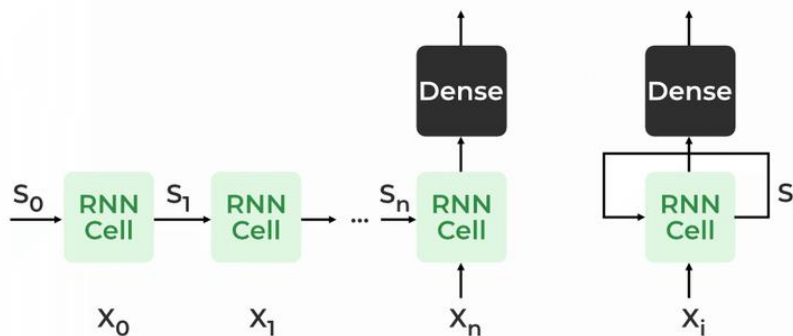
Εικόνα 4 Παράδειγμα MLP δικτύου

**Deep Learning Models:** Αποτελούν διάφορα μοντέλα με πολλά επίπεδα νευρώνων που χρησιμοποιούνται για την εξαγωγή πληροφορίας από τα δεδομένα μας. Τα πιο γνωστά είδη deep learning μοντέλα είναι τα **Convolutional Neural networks (CNN)** (Εικόνα 5) τα οποία χρησιμοποιούνται για επεξεργασία και ταξινόμηση σε εικόνες και βίντεο και τα **Recurrent Neural Network (RNN)** (Εικόνα 6) τα οποία χρησιμοποιούνται σε ακολουθιακά δεδομένα όπως πχ η φωνή καθώς μαθαίνουν να αναγνωρίζουν τα δεδομένα με σκοπό να προβλέψουν το επόμενο.



Εικόνα 5 Παράδειγμα CNN δικτύου

## RECURRENT NEURAL NETWORKS



Εικόνα 6 Παράδειγμα RNN δικτύου

## 2 Βιβλιογραφική Ανασκόπηση

Διάφορες μελέτες έχουν εφαρμόσει αλγορίθμους μηχανικής μάθησης για την μελέτη ατόμων με ΔΑΦ. Αυτές οι μελέτες έχουν εστιάσει σε διαφορετικές μεθοδολογίες και χρησιμοποιούν διαφορετικούς αλγορίθμους μηχανικής μάθησης.

Αρκετοί ερευνητές κάνουν προγνώσεις με βάση εικόνες από εγκεφαλικές απεικονίσεις χρησιμοποιώντας συστήματα MRI (Magnetic Resonance Imaging). Οι Krishna Kumar et al. [19] αναφέρουν πως σε μελέτη τους, η οποία χρησιμοποίησε μοντέλα μηχανικής μάθησης για την εύρεση μοτίβων σε μαγνητικές απεικονίσεις κατάφεραν να δημιουργήσουν ένα μοντέλο το οποίο προβλέπει με 92% ακρίβεια περιπτώσεις ΔΑΦ δίχως να αναφέρουν συγκεκριμένα ποια μοντέλα χρησιμοποίησαν. Οι Song et al.[20] χρησιμοποίησαν την τεχνική radiomics (μια μέθοδο ανάλυσης ιατρικών εικόνων για την εξαγωγή χαρακτηριστικών και μοτίβων που σχετίζονται με την ασθένεια) για την εξαγωγή χαρακτηριστικών από εικόνες MRI εστιάζοντας στην εμφάνιση λευκής ουσίας

(ένα τύπο νευρικού ιστού ο οποίος βοηθάει στην σύνδεση και επικοινωνία διαφόρων περιοχών του εγκεφάλου). Αυτά τα χαρακτηριστικά στην συνέχεια τροφοδότησαν διάφορα μοντέλα (Logistic Regression, K-Nearest-Neighbors, Support Vector Machines, Convolutional Neural Networks) με τα SVM και CNN να ξεχωρίζουν πετυχαίνοντας 89.47% και 86.48% ακρίβεια εκάστοτε. Αυτή η έρευνα βοήθησε στην δημιουργία του MultiUNet ενός μοντέλου τμηματοποίησης (διαίρεση της εικόνας σε περιοχές με κοινά χαρακτηριστικά για την ευκολότερη επεξεργασία της) το οποίο πέτυχε IoU (Intersection over Union μετρικό που χρησιμοποιείται στην αξιολόγηση της ακρίβειας τμηματοποίησης) 0.92 καθιστώντας το ένα δυνατό εργαλείο στην πρόιμη διάγνωση της ΔΑΦ.

Οι Thabtah et al. [21], αναλύοντας τα dataset AQ-Adults, AQ-Child, AQ-Adolescent (dataset για διαφορετικές ηλικιακές μονάδες που περιέχουν διάφορα χαρακτηριστικά της ΔΑΦ) κατάφεραν να δημιουργήσουν δύο μοντέλα μηχανικής μάθησης κατάλληλα για ταξινόμηση ατόμων με ΔΑΦ. Πιο συγκεκριμένα χρησιμοποίησαν την μέθοδο Na για να μειώσουν την διάσταση των χαρακτηριστικών και χρησιμοποιώντας τους αλγορίθμους RIPPER και C4.5 κατάφεραν recall = 87.3% στο AQ-Adolescent dataset με το νούμερο αυτό να μειώνεται στο 80.95% (C4.5) και 82.54% (RIPPER) για το AQ-Adult. Οι Raj et al. [22] ακολουθώντας μια παρόμοια προσέγγιση χρησιμοποίησαν ένα dataset το οποίο περιείχε 20 συνηθισμένα χαρακτηριστικά που συσχετίζονται με την ΔΑΦ για να εκπαιδεύσουν διάφορα μοντέλα μηχανικής μάθησης για την ταξινόμηση ενός ατόμου ως νευροτυπικό ή άτομο με ΔΑΦ. Πιο συγκεκριμένα χρησιμοποιώντας τους αλγορίθμους Support Vector Machine, Naïve Bayes, Logistic Regression, K-Nearest Neighbor, ANN και CNN πέτυχαν συνολική ακρίβεια στην ταξινόμηση σε συνολικό ποσοστό 96.88% με τα καλύτερα αποτελέσματα να τα επιτυγχάνουν οι CNN και SVM σε ποσοστό 99.53%.

Αρκετά συνηθισμένη είναι και η χρήση ηλεκτροεγκεφαλογραφήματος (EEG) στην μελέτη της ΔΑΦ. Σύμφωνα με μελέτη των Jayawardana et al. [23], οι οποίοι χρησιμοποιώντας EEG συλλεγμένα από 8 παιδιά με ΔΑΦ και 9 περιπτώσεις χαμηλού κινδύνου κατέληξαν σε αποτελέσματα ακρίβειας 56% στην ταξινόμηση με την χρήση γραμμικών μοντέλων με την ακρίβεια αυτή να βελτιώνεται σημαντικά με την χρήση μη γραμμικών μοντέλων ξεπερνώντας το 90% με την χρήση του CNN. Οι Bhaskarachary et al. [24] χρησιμοποιώντας δεδομένα που συλλέχθηκαν από EEG δημιούργησαν ένα μοντέλο transfer learning το οποίο προεπεξεργάζεται τα ηλεκτρικά σήματα από το εγκεφαλογράφημα εξάγει την ισχύ των διαφορετικών συχνοτήτων του εγκεφάλου (δέλτα, θήτα, άλφα, βήτα) ως χαρακτηριστικά και τα χρησιμοποιεί για την ταξινόμηση των ατόμων με τους αλγορίθμους Extra Trees και XGBoost, φτάνοντας έτσι σε αποτελέσματα ακρίβειας 67.7% και recall 83.3%. Ακόμα μια πρωτοποριακή έρευνα ήταν και των Mohi et al. [25] οι οποίοι μετασχημάτισαν δεδομένα από EEG χρησιμοποιώντας διάφορους μαθηματικούς μετασχηματισμούς με τους πιο σημαντικούς από αυτούς να είναι οι διακριτός μετασχηματισμός Φουριερ (DFT) και Autoregressive (AR) coefficients καταφέρνοντας έτσι ακρίβεια 90% στον K-Nearest Neighbors χρησιμοποιώντας μετασχηματισμό AR με την ακρίβεια για το ίδιο μοντέλο

να αυξάνεται στο 93% όταν τα χαρακτηριστικά εξαγόταν με DFT. Οι Liao κ.α [26] δημιούργησαν ένα framework χρησιμοποιώντας δεδομένα που συλλέχθηκαν από στερεώσεις ματιών, EEG και εκφράσεις προσώπου των συμμετεχόντων στην συνέχεια συνδυάζοντας τα 3 αυτά είδη δεδομένων εκπαίδευσαν 3 διαφορετικούς αλγόριθμους (Random Forest, Support Vector Machine, K-Nearest Neighbors, weighted Naïve bayes) καταφέρνοντας έτσι ακρίβεια στην ταξινόμηση ίση με 87.5% με τον αλγόριθμο weighted Naïve bayes.

Αρκετές έρευνες έχουν εστιάσει στην χρήση τεχνολογίας eye-tracking για ταξινομήσεις που σχετίζονται με την ΔΑΦ. Πιο συγκεκριμένα οι Carette et al. [27] προτείνει την χρήση ενός LSTM (Long Term Short Memory) μοντέλου για την ταξινόμηση παιδιών ηλικίας 8-10 ετών ως άτομα με ΔΑΦ. Οι ερευνητές χρησιμοποίησαν δεδομένα που συλλέχθηκαν από ένα σύστημα ανίχνευσης ματιών και με την χρήση του LSTM το οποίο ανέλυσε τις κινήσεις των ματιών με την πάροδο του χρόνου κατέληξαν σε σωστή διάκριση μεταξύ των δύο κατηγοριών στο 83% των περιπτώσεων με εμπιστοσύνη στα αποτελέσματα σε επίπεδο 95%. Σε επόμενη μελέτη τους [28] δοκίμασαν την χρήση διάφορων μοντέλων μηχανικής μάθησης (Naïve Bayes, Logistic Regression, Support Vector Machine, Random Forest και Artificial Neural Networks) για να ταξινομήσουν εικόνες οι οποίες δημιουργήθηκαν από eye-tracking patterns παιδιών μέσης ηλικίας 8 ετών σε άτομα με ΔΑΦ ή νευροτυπικά και παράλληλα δημιουργώντας ένα dataset το οποίο χρησιμοποιήθηκε σε διάφορες μελέτες παρόμοιου θέματος. Μια διαφορετική προσέγγιση εφάρμοσαν οι Elbattah et al. [29] χρησιμοποιώντας το dataset που προαναφέρθηκε εφάρμοσαν τον k-mean αλγόριθμο ο οποίος τους οδήγησε στο συμπέρασμα πως υπάρχει σύνδεση μεταξύ της γρηγορότερης κίνησης των ματιών και τις σοβαρότητας των συμπτωμάτων.

Οι Zhao et al. [30] θεώρησαν πως υπάρχει διαφορά στην συμπεριφορά του βλέμματος μεταξύ πραγματικής αλληλεπίδρασης με άλλα άτομα και παρακολούθησης οπτικών μέσων. Χρησιμοποιώντας τον αλγόριθμο SVM κατάφεραν να ταξινομήσουν παιδιά ως τυπικά αναπτυσσόμενα ή με ΔΑΦ με ακρίβεια 92.31% αποδεικνύοντας έτσι πως η μηχανική μάθηση μπορεί να χρησιμοποιηθεί για την διάγνωση ΔΑΦ σε παιδιά με αντικειμενικά κριτήρια σε καθημερινές κοινωνικές αλληλεπιδράσεις.

### **3 Σκοπός της Έρευνας**

Αυτή η έρευνα χρησιμοποιώντας το dataset που ανέπτυξαν οι Carette et al. στο άρθρο [28], πραγματοποιεί classification των συμμετεχόντων σε νευροτυπικούς και άτομα με ΔΑΦ χρησιμοποιώντας 5 διαφορετικούς αλγόριθμους μηχανικής μάθησης (SVM, Random Forest, CNN, ANN, Random Forest with Convolutional Filters). Η ειδοποιός διαφορά αυτής της έρευνας με παρόμοιες που χρησιμοποιούν το ίδιο dataset είναι η χρήση του αλγόριθμου Random Forest with Convolutional Filters, που συνδυάζει την αναγνώριση μοτίβων στις εικόνες, που παρέχουν τα convolutional filter των CNN, με την ταχύτητα στην εκπαίδευση του Random Forest. Μια ακόμα πρωτοτυπία είναι και η δημιουργία ενός eye-tracker με βασικές λειτουργίες καθιστώντας έτσι την δοκιμή του

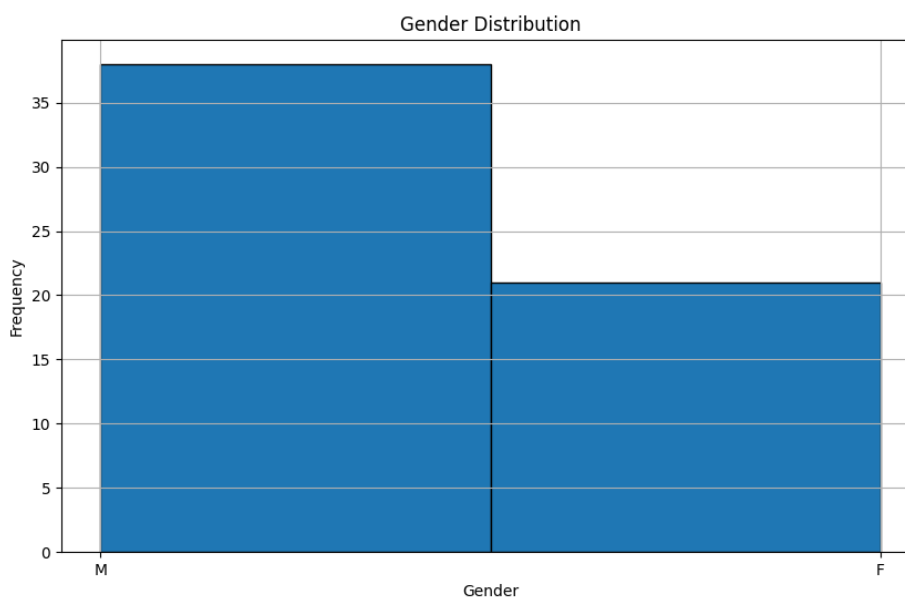
μοντέλου σε πραγματικά δεδομένα πιο προσιτή με την χρήση απλά μιας κάμερας υπολογιστή αντί για την αγορά εξειδικευμένων eye-trackers.

## 4 Μεθοδολογία

### 4.1 Περιγραφή Dataset

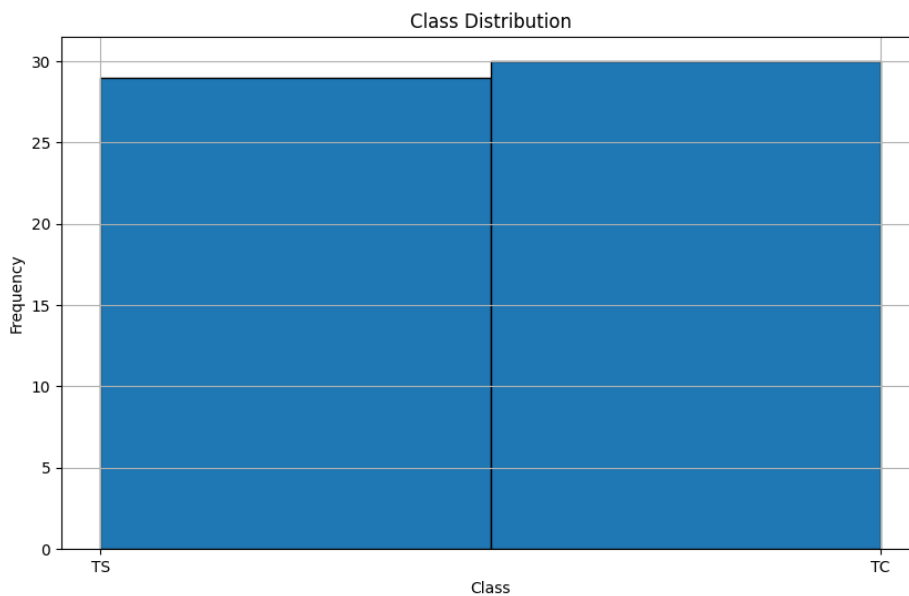
Για την μελέτη μας χρησιμοποιήθηκε το dataset από το άρθρο «Visualization of Eye-Tracking Patterns in Autism Spectrum Disorder: Method and Dataset» [28] για την ανάλυση και οπτικοποίηση των προτύπων eye-tracking σε άτομα με ΔΑΦ. Το dataset περιλαμβάνει οπτικοποιημένα δεδομένα eye-tracking τα οποία καταγράφουν τις κινήσεις ματιών και την κατανομή της προσοχής μεταξύ ατόμων με ΔΑΦ και νευροτυπικών ατόμων. Οι συμμετέχοντες παρακολούθησαν ποικιλία οπτικών ερεθισμάτων και τα δεδομένα που συλλέχθηκαν περιλαμβάνουν συντεταγμένες σημείων βλέμματος, διάρκεια ακινητοποιήσεων και ταχείες κινήσεις των ματιών (saccades).

Όπως φαίνεται στην εικόνα 7, για τη δημιουργία αυτού του dataset συλλέχθηκαν δείγματα από 59 διαφορετικούς συμμετέχοντες (38 Άντρες και 21 Γυναίκες).



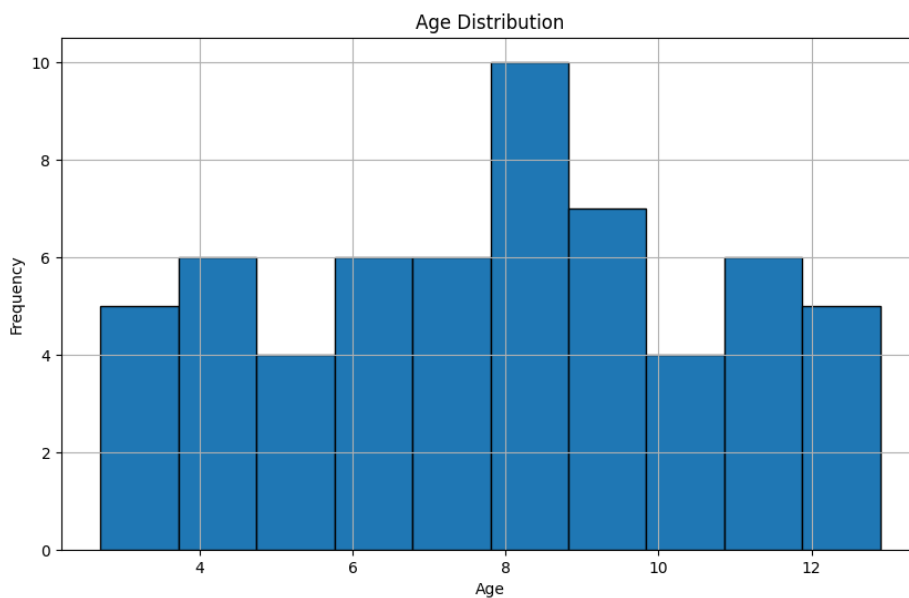
*Εικόνα 7 Κατανομή φύλου στο dataset*

Με την βοήθεια ενός ψυχολόγου, οι συμμετέχοντες ταξινομήθηκαν με βάση τα αποτελέσματά τους σε δύο κατηγορίες 1) Νευροτυπικά (TC) ή 2) Άτομα με ΔΑΦ (TS). (Εικόνα 8)



*Εικόνα 8 Κατανομή νευροτυπικών / ατόμων με ΔΑΦ στο dataset*

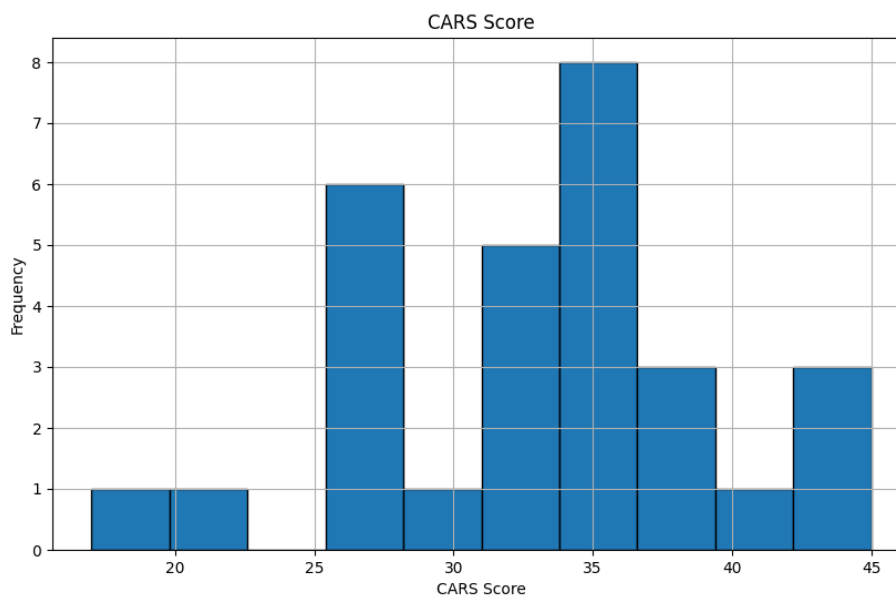
Στην εικόνα 9 φαίνεται το η ηλικιακή κατανομή των συμμετεχόντων. Το dataset αποτελείται από άτομα νεαρής ηλικίας (3-13 χρονών) με περίπου το 47% των συμμετεχόντων να έχει ηλικία μεταξύ 5 και 9 ετών.



*Εικόνα 9 Ηλικιακή κατανομή των συμμετεχόντων*

Ως μέρος των μεταδεδομένων του dataset οι συμμετέχοντες αξιολογήθηκαν και με βάση την κλίμακα Childhood Autism Rating Scale. Η Childhood Autism Rating Scale (CARS) είναι μια κλινική κλίμακα αξιολόγησης η οποία χρησιμοποιείται από ειδικούς για την διάγνωση ΔΑΦ αλλά και την ταξινόμηση μέσα στο αυτιστικό φάσμα. Αποτελείται από 15 διαφορετικούς τομείς συμπεριφοράς συσχετισμένους με το φάσμα

του αυτισμού και ο κάθε τομέας μπορεί να πάρει τιμές από 1 έως 4. Τα συνολικά αποτελέσματα κυμαίνονται από 15 έως 60 με αποτελέσματα κάτω από 30 να υποδεικνύουν ότι το άτομο δεν είναι στο φάσμα. Αποτελέσματα 30 έως 36.5 υποδεικνύουν ήπια έως μέτρια μορφή αυτισμού ενώ αποτελέσματα από 37 και πάνω συσχετίζονται με πιο σοβαρές περιπτώσεις [31]. Όπως φαίνεται και στην εικόνα 10 ο μεγαλύτερος αριθμός των συμμετεχόντων είχε αξιολογηθεί με CARS από 25 έως 36 υποδηλώνοντας έτσι πως είτε δεν ανήκαν στο φάσμα είτε οι περισσότερες περιπτώσεις ήταν ελαφριές μορφές ΔΑΦ.



Εικόνα 10 Δείκτης CARS των συμμετεχόντων

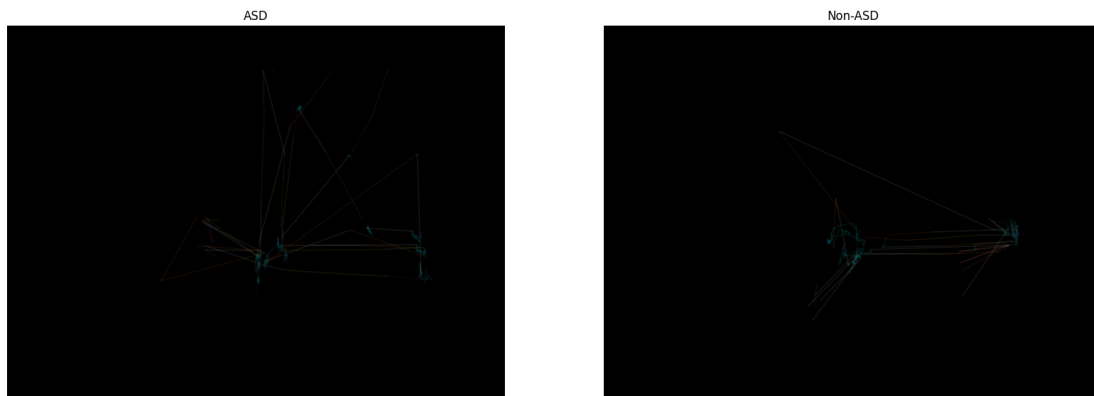
Στον πίνακα 1 φαίνονται τα συνολικά στατιστικά των συμμετεχόντων.

Αριθμός συμμετεχόντων	59
Κατανομή φύλου(Αρσενικό / Θηλυκό)	38 (64.4%) /21 (35.6%)
Νευροτυπικά άτομα	30
Άτομα με ΔΑΦ	29
Ηλικία(Μέσος όρος /Διάμεσος)	7.88 / 8.1
CARS(Μέσος όρος /Διάμεσος)	32.97 / 34.5

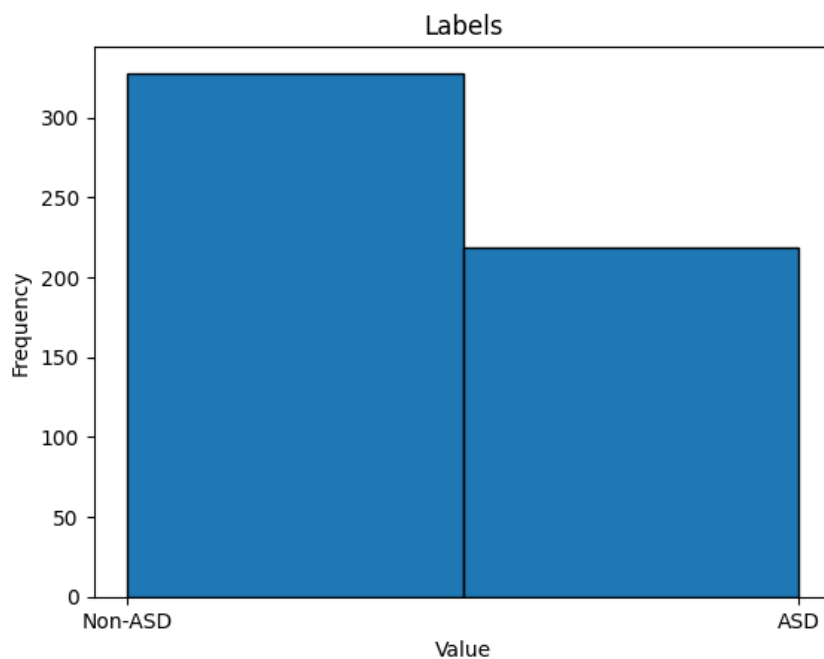
Πίνακας 1 Συνολικά στατιστικά των συμμετεχόντων

Στη συνέχεια, οι συμμετέχοντες παρακολούθησαν διάφορα οπτικοακουστικά ερεθίσματα και η παρακολούθηση των ματιών τους καταγράφηκε και τα δεδομένα οπτικοποιήθηκαν, παράγοντας έτσι 547 διαφορετικές εικόνες (328 νευροτυπικών ετικετών / 219 ετικετών που υποδηλώνουν ΔΑΦ) όπως φαίνεται στην εικόνα 12, οι οποίες αναπαριστούν τις κινήσεις των ματιών και τις περιοχές ενδιαφέροντος νευροτυπικών ατόμων και ατόμων με ΔΑΦ. Στην εικόνα 11 φαίνονται οπτικοποιημένα παραδείγματα από μοτίβα κινήσεων ματιών του dataset ενώ η κατανομή των ετικετών του dataset φαίνεται στην εικόνα 12.





Εικόνα 11 Eye-Tracking Pattern ατόμου με ΔΑΦ (Αριστερά) και νευροτυπικού ατόμου (Δεξιά)



Εικόνα 12 Κατανομή ετικετών στο dataset

## 4.2 Προ-επεξεργασία δεδομένων

Τα δεδομένα μας αρχικά αποτελούνται από εικόνες μεγέθους 480 x 640 pixel 3 καναλιών χρώματος.

Το πρώτο βήμα στην προ-επεξεργασία των εικόνων μας είναι η κανονικοποίησή τους στο διάστημα 0-1. Για να το επιτύχουμε αυτό διαιρούμε όλα τα pixel της εικόνας με 255.0 που είναι η μέγιστη τιμή ενός pixel. Με αυτό τον τρόπο επιτυγχάνουμε πιο σταθερή διαδικασία μάθησης και μείωση της υπερεκπαίδευσης καθώς τα μοντέλα που χρησιμοποιούν δεδομένα εισόδου με μεγάλες διακυμάνσεις τείνουν να μαθαίνουν εξειδικευμένα χαρακτηριστικά αντί να γενικεύουν σε νέα δεδομένα. Τέλος, με την κανονικοποίηση επιταχύνεται η σύγκλιση των αλγορίθμων βελτιστοποίησης.

Στη συνέχεια, οι εικόνες μετατράπηκαν σε γκρι κλίμακα (grayscale) έτσι ώστε να μειωθούν τα κανάλια χρώματος από τρία (RGB) σε ένα, να μειωθεί η πολυπλοκότητα των δεδομένων, και να αυξηθεί η ταχύτητα επεξεργασίας των δεδομένων. Επιπλέον, με αυτή την τεχνική μειώνεται τυχόν θόρυβος σε εικόνες, που δημιουργείται από την πληροφορία του χρώματος, κάνοντας έτσι πιο εύκολη την αναγνώριση προτύπων.

Τέλος, καθώς οι εικόνες μας έχουν ένα μαύρο background και αρκετό κενό χώρο έγινε αλλαγή μεγέθους των εικόνων σε 225 x 225 pixel. Με αυτό τον τρόπο διατηρείται η ομοιομορφία των δεδομένων καθώς όλες οι εικόνες έχουν το ίδιο μέγεθος ενώ ταυτόχρονα μειώνεται και η ποσότητα των δεδομένων που πρέπει να επεξεργαστεί το μοντέλο μας ελαττώνοντας έτσι τον χρόνο εκπαίδευσης και πρόβλεψης χωρίς όμως να χάνονται σημαντικές λεπτομέρειες στις εικόνες.

### 4.3 Μετρικά αξιολόγησης μοντέλων

Στη δυαδική ταξινόμηση υπάρχουν δύο διαφορετικές κατηγορίες οι οποίες πρέπει να ταξινομηθούν και κατηγοριοποιούνται ως θετικές και αρνητικές ( 1 ή 0).

- **Accuracy (Ακρίβεια):** Η ακρίβεια αναπαριστά πόσο συχνά ένα μοντέλο προβλέπει σωστά το αποτέλεσμα. Υπολογίζεται ως το άθροισμα των σωστών προβλέψεων προς τις συνολικές προβλέψεις.

$$acc = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Recall (Ανάκληση):** Η ανάκληση ή ‘ευαισθησία’ αντιπροσωπεύει το ποσοστό των αληθινά θετικών παραδειγμάτων που το μοντέλο κατάφερε να ταξινομήσει ένα μοντέλο μηχανικής μάθησης. Υπολογίζεται ως ο αριθμός των πραγματικά θετικών προβλέψεων διαιρούμενο με το άθροισμα των ψευδός αρνητικών αποτελεσμάτων και των αληθινά θετικών αποτελεσμάτων.

$$rec = \frac{TP}{TP + FN}$$

- **Precision (Ακρίβεια):** Η ακρίβεια σε αντίθεση με την ανάκληση αφορά στο πλήθος των παραδειγμάτων όπου το μοντέλο ταξινόμησε ως σωστά και ήταν πραγματικά σωστά. Υπολογίζεται ως ο αριθμός των πραγματικά θετικών τιμών διαιρούμενος με το άθροισμα των αληθινά θετικών αποτελεσμάτων με τον ψευδός θετικών αποτελεσμάτων.

$$prec = \frac{TP}{TP + FP}$$

- **F1-Scores:** Καθώς η ακρίβεια (precision) και η ανάκληση είναι και τα δύο χρήσιμα μετρικά αλλά εν μέρη αντίθετα το f1 – score είναι ένα μετρικό το οποίο υπολογίζει το αρμονικό μέσο της ακρίβειας και της ανάκλησης.

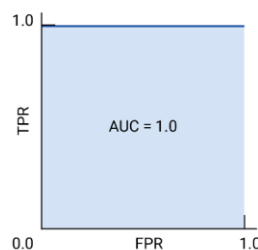
$$2 * \frac{precision * recall}{precision + recall} \text{ ή } \frac{2TP}{2TP + FP + FN}$$

- **Confusion Matrix (Πίνακας σύγχυσης):** Όπως παρουσιάζεται στην εικόνα 13, ο πίνακας ή μήτρα σύγχυσης είναι ένας πίνακας ο οποίος εμφανίζει και συγκρίνει τις πραγματικές με τις προβλεπόμενες τιμές του ενός μοντέλου. Είναι ιδιαίτερα χρήσιμος καθώς σε αντίθεση με άλλα μετρικά παρέχει μια πιο ολοκληρωμένη εικόνα του τρόπου απόδοσης ενός μοντέλου.

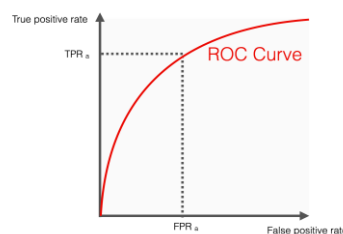
		Actual Values	
		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

*Εικόνα 13 Παράδειγμα πίνακα σύγχυσης δυαδικού μοντέλου*

- **Receiver Operating Characteristic Curve (ROC Curve)(Εικόνα 15):** Είναι μια καμπύλη, η οποία σχεδιάζεται υπολογίζοντας το True Positive Rate (TPR) και False Positive Rate (FPR).
- **Area Under the Curve (AUC)(Εικόνα 14):** Είναι το εμβαδόν της περιοχής που καλύπτεται από την καμπύλη ROC. Όσο πλησιάζει το 1 τόσο καλύτερη ταξινόμηση κάνει το μοντέλο.



*Εικόνα 14 Καμπύλη ROC ιδανικού μοντέλου*



*Εικόνα 15 Καμπύλη ROC πραγματικού μοντέλου*

## 4.4 Αλγόριθμοι

Για την επιλογή του μοντέλου που θα ταξινομεί τα αποτελέσματα δοκιμάστηκαν οι παρακάτω αλγόριθμοι με διαφορετικές τιμές έτσι ώστε να βρεθεί η βέλτιστη υλοποίησή τους.

### Random forest

- **N\_estimators:** Ο αριθμός των δέντρων απόφασης που θα χρησιμοποιηθούν στον αλγόριθμο.
- **Max\_depth:** το μέγιστο βάθος που θα φτάσει ο αλγόριθμος
- **Min\_samples\_split:** Ο ελάχιστος αριθμός δειγμάτων που χρειάζεται για να χωριστεί ένας κόμβος.
- **Min\_sample\_leaf:** Ο ελάχιστος αριθμός δειγμάτων που χρειάζεται για να χαρακτηριστεί ένας κόμβος φύλλο.

### Support Vector Machines

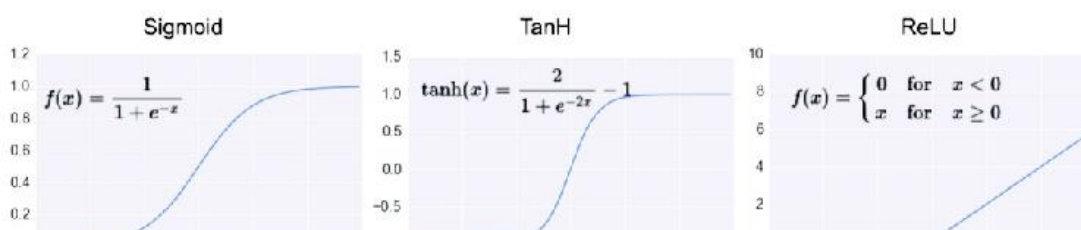
#### Artificial Neural Network for binary Classification

Στον πίνακα 2 φαίνονται οι τιμές των υπερπαραμέτρων που χρησιμοποιήθηκαν για την εκπαίδευση του ANN

Παράμετρος	Τιμή
Number of hidden layers	2
Number of neurons	128, 64
Activation function	sigmoid
Dropout Rate	0.5

Πίνακας 2 Παράμετροι ANN

- **Number of hidden layers:** Ο αριθμός κρυφών επιπέδων του μοντέλου μας
- **Number of neurons:** Ο αριθμός των νευρώνων για κάθε κρυφό επίπεδο του μοντέλου μας
- **Activation function(Εικόνα 16):** Η συνάρτηση ενεργοποίησης του κάθε νευρώνα στο στρώμα μας

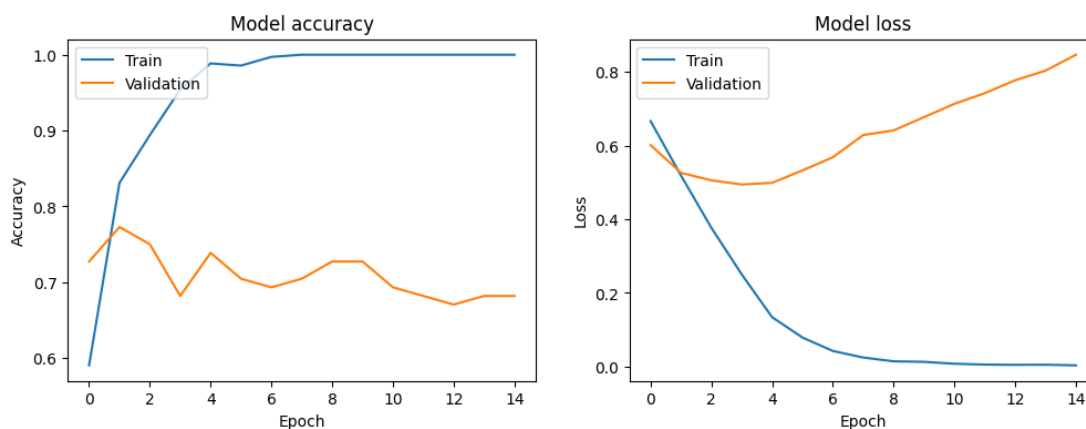


Εικόνα 16 Διαφορετικές συναρτήσεις ενεργοποίησης

- **Dropout rate:** Το ποσοστό των νευρώνων που απορρίπτονται τυχαία κατά την επεξεργασία κάθε παρτίδας δεδομένων.

Το μοντέλο εκπαιδεύτηκε για 15 epochs με το 20% των δεδομένων εκπαίδευσης να χρησιμοποιείται για validation.

Η εικόνα 17 δείχνει την ακρίβεια και την απώλεια εκπαίδευσης / validation ανά εποχή.



Εικόνα 17 Αποτελέσματα εκπαίδευσης ANN

## Convolutional Neural Network

Στον πίνακα 3 φαίνονται οι τιμές των υπερπαραμέτρων που χρησιμοποιήθηκαν για την εκπαίδευση του CNN

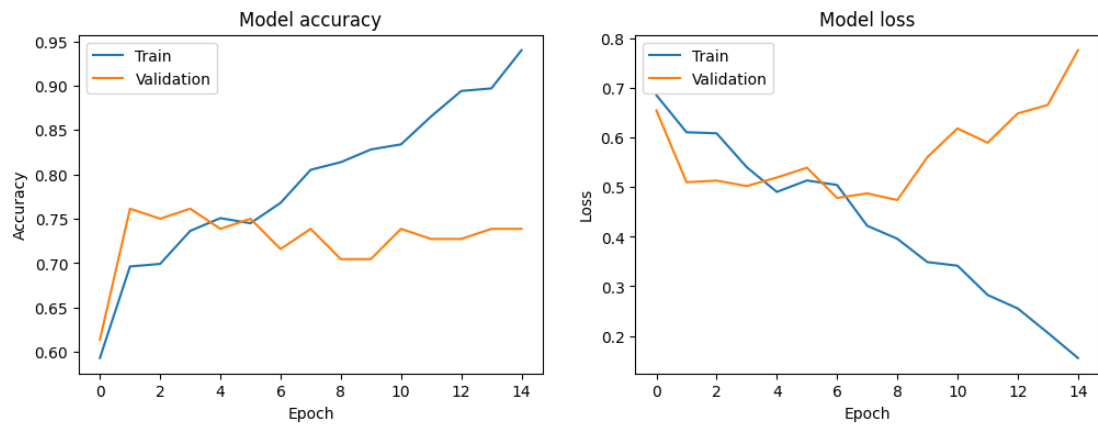
Παράμετρος	Τιμή
Convolutional filters	32, 64, 128
Kernel Size	(3,3)
Pool Size	(2,2)
Hidden Layers	2
Number of neurons	128, 64
Dropout Rate	0.5

Πίνακας 3 Παράμετροι CNN

- **Convolutional filters:** Ο αριθμός των φίλτρων που εφαρμόζονται σε κάθε στάδιο συνέλιξης. Κάθε φίλτρο χρησιμοποιείται για την εξαγωγή συγκεκριμένων χαρακτηριστικών από την εικόνα.
- **Kernel Size:** Είναι ο διδιάστατος πίνακας που χρησιμοποιείται ως φίλτρο στην εφαρμογή της συνέλιξης. Οι διαστάσεις του πυρήνα καθορίζουν το πλήθος των σημείων της εικόνας που ελέγχονται κατά την εφαρμογή του φίλτρου σε κάθε βήμα της συνέλιξης.
- **Pool Size:** Αφορά το μέγεθος της περιοχής από την οποία επιλέγεται το μέγιστο ή το μέσο όρο των τιμών, κατά τη διαδικασία της συγκέντρωσης (pooling). Η συγκέντρωση χρησιμοποιείται για τη μείωση της χωρικής διάστασης των χαρακτηριστικών, διατηρώντας παράλληλα τα σημαντικά χαρακτηριστικά

Το μοντέλο εκπαιδεύτηκε για 15 epochs με το 20% των δεδομένων εκπαίδευσης να χρησιμοποιείται για validation.

Η εικόνα 18 δείχνει την ακρίβεια και την απώλεια εκπαίδευσης / validation ανά εποχή



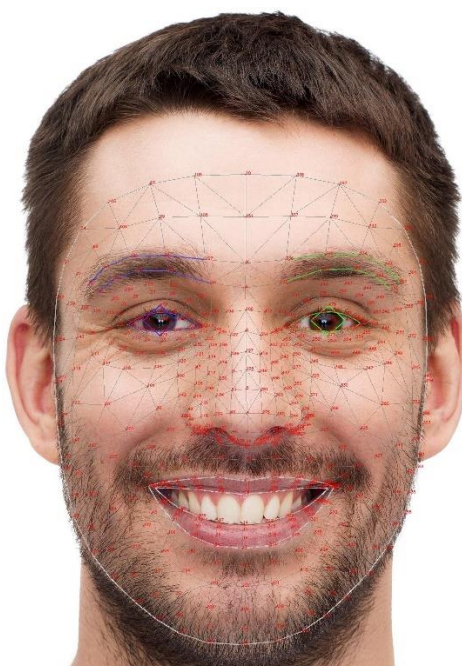
• *Εικόνα 18 Αποτελέσματα εκπαίδευσης CNN*

## 4.5 Δημιουργία Eye-tracker

Για την αξιολόγηση του μοντέλου σε πραγματικά δεδομένα δημιουργήθηκε ένα πρόγραμμα το οποίο λειτουργεί ως ένας απλοϊκός VOG eye-tracker κάνοντας ανίχνευση της θέσης του βλέμματος αλλά και υπολογισμούς σχετικά με την ταχύτητα μεταβολής του βλέμματος. Το πρόγραμμα δημιουργήθηκε χρησιμοποιώντας την γλώσσα προγραμματισμού rython μαζί με διάφορες βιβλιοθήκες της και την κάμερα ενός υπολογιστή για την παρακολούθηση των ματιών.

### 4.5.1 Εξαγωγή χαρακτηριστικών προσώπου

Για την εξαγωγή των χαρακτηριστικών του προσώπου (Εικόνα 19) δημιουργείται ένα πλέγμα από σημεία, το οποίο αναπαριστά το πρόσωπο του ατόμου. Στη συνέχεια, επιλέγονται 5 σημεία για κάθε μάτι και πιο συγκεκριμένα, 4 περιμετρικά του ματιού και 1 στην κόρη.



Εικόνα 19 Παράδειγμα πλέγματος προσώπου

Με την καταγραφή εικόνας μέσα από την κάμερα του υπολογιστή η θέση των ματιών ‘αιχμαλωτίζεται’ 30 φορές το δευτερόλεπτο.

#### 4.5.2 Calibration του Eye-tracker

Καθώς το βλέμμα σε κάθε άτομο είναι διαφορετικό, για να λειτουργήσει σωστά ένας eye-tracker χρειάζεται κάποια αρχική παραμετροποίηση. Για να επιτευχθεί αυτό εμφανίζονται διαδοχικά πέντε σημεία στην οθόνη, ένα σε κάθε γωνία και ένα στο κέντρο και ο χρήστης ενθαρρύνεται να εστιάσει σε αυτά συγκεκριμένο χρονικό διάστημα. Η τοποθέτηση των σημείων προσαρμόζεται δυναμικά ανάλογα με το μέγεθος και την ανάλυση της οθόνης.

Στον πίνακα 4 φαίνονται τα σημεία που χρησιμοποιούνται για το calibration.

Θέση σημείου στην οθόνη	Συντεταγμένες
Κέντρο	[πλάτος οθόνης / 2, ύψος οθόνης / 2]
Πάνω αριστερά	[πλάτος οθόνης – 100, 100]
Κάτω αριστερά	[100, ύψος οθόνης – 100]
Πάνω δεξιά	[100, ύψος οθόνης – 100]
Κάτω δεξιά	[πλάτος οθόνης -100, ύψος οθόνης – 100]

Πίνακας 4 Σημεία Calibration

Οι συντεταγμένες x και y καταγράφονται ξεχωριστά για κάθε μάτι. Στη συνέχεια, υπολογίζεται η μέση τιμή του x και του y και των δύο ματιών και μέσω της παρακάτω σχέσης μετατρέπονται οι συντεταγμένες των ματιών σε συντεταγμένες οθόνης.

$$screenX = avgX * screenWidth \mid screenY = avgY * screenHeight$$

Καθώς αυτό συμβαίνει σε κάθε frame, συλλέγονται τόσα σημεία όσα και το framerate της κάμερας \* 2 που είναι τα δευτερόλεπτα που μένει ορατό ένα σημείο. Για κάθε

σημείο calibration υπολογίζεται το μέσο  $x$  και  $y$  όλων των δεδομένων που συλλέχθηκαν. Μόλις αυτή η διαδικασία ολοκληρωθεί, υπολογίζεται ο πίνακας ομοιογραφίας των σημείων που υπολογίσαμε και των σημείων calibration. Ο πίνακας ομοιογραφίας χρησιμοποιείται για το μετασχηματισμό των σημείων δύο εικόνων έτσι ώστε τα σημεία στις δύο εικόνες να ευθυγραμμιστούν μεταξύ τους

Υπολογίζεται από τον παρακάτω μαθηματικό τύπο:

$$\begin{bmatrix} x' & y' & 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Τέλος, για να υπολογιστεί η πραγματική θέση των ματιών στην οθόνη εφαρμόζεται στις συντεταγμένες  $x$  και  $y$  προοπτικός μετασχηματισμός χρησιμοποιώντας τον πίνακα ομοιογραφίας που υπολογίσαμε. Ο προοπτικός μετασχηματισμός χρησιμοποιεί έναν πίνακα  $3 \times 3$  για να μετασχηματίσει τις συντεταγμένες ενός επιπέδου (π.χ. επιφάνεια ματιού) σε ένα άλλο επίπεδο (π.χ. οθόνη).

Υπολογίζεται από τον μαθηματικό τύπο:

$$(x', y', 1) = (x, y, 1) \cdot \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$$

#### 4.5.3 Διαδικασία εντοπισμού κίνησης των ματιών

Αρχικά λαμβάνονται καρέ από την κάμερα του υπολογιστή τα οποία αναστρέφονται και μετατρέπονται από BGR (Blue – Green – Red) το οποίο είναι ο προεπιλεγμένος χρωματισμός της βιβλιοθήκης `opencv` σε RGB (Red – Green – Blue) έτσι ώστε να είναι έτοιμα για επεξεργασία. Στη συνέχεια, γίνεται η καταγραφή ματιών όπως περιγράφεται στην ενότητα 3.4.1. Για κάθε καρέ υπολογίζονται οι συντεταγμένες του κάθε ματιού ενώ για να υπολογιστεί η θέση του βλέμματος σχετικά με το καρέ υπολογίζεται ο μέσος όρος των συντεταγμένων του κάθε ματιού. Για να μεταφραστούν οι συντεταγμένες πάνω στην οθόνη, θα χρησιμοποιηθεί ο πίνακας ομοιογραφίας που υπολογίστηκε κατά την διαδικασία του calibration έτσι ώστε να πραγματοποιηθεί προοπτικός μετασχηματισμός όπως περιγράφεται στην ενότητα 3.4.3. Τέλος, οι συντεταγμένες ελέγχονται για το αν ξεπερνούν τα όρια της οθόνης ( σε περίπτωση που τα ξεπερνούν η τιμή τους θέτεται ως η τελευταία αποδεκτή τιμή )

#### 4.5.4 Υπολογισμός μαθηματικών σχέσεων

Αρχικά, θα επιλεγθούν 200 σημεία από τα συλλεγμένα δεδομένα για να οπτικοποιηθούν. Το μήκος του βήματος υπολογίζεται ως το μέγεθος του πίνακα των συντεταγμένων  $x$  προς 200 (ο αριθμός των ζητούμενων σημείων). Στη συνέχεια, ξεκινώντας από το πρώτο σημείο, στις συντεταγμένες  $x$  και  $y$  αποθηκεύονται τα σημεία προχωρώντας με το υπολογισμένο βήμα.



Στη συνέχεια, υπολογίζεται ο αριθμός των καρέ ως το μήκος του βίντεο προς τον αριθμό των έγκυρων σημείων:

$$frame = \frac{(video\ length)}{x\_valid}$$

Χρησιμοποιώντας τον αριθμό των καρέ θα υπολογιστεί ο χρόνος ανάμεσα σε δύο διαδοχικές συντεταγμένες:

$$time\_frame = \frac{1}{frame}$$

Για τον υπολογισμό της ταχύτητας κίνησης του βλέμματος θα χρησιμοποιηθεί ο μαθηματικός τύπος:

$$speed = \frac{(distance)}{time_{frame}}$$

Ο υπολογισμός της απόστασης θα πραγματοποιηθεί με τον τύπο:

$$distance = \sqrt{((x_i - x_{i-1})^2 + (y_i - y_{i-1})^2)}$$

Η επιτάχυνση του βλέμματος θα υπολογιστεί υπολογίζοντας την παράγωγο της ταχύτητας:

$$acceleration = abs\left(\frac{speed_i - speed_{(i-1)}}{time_{frame}}\right)$$

Ομοίως, το jerk θα υπολογιστεί ως η παράγωγος της επιτάχυνσης:

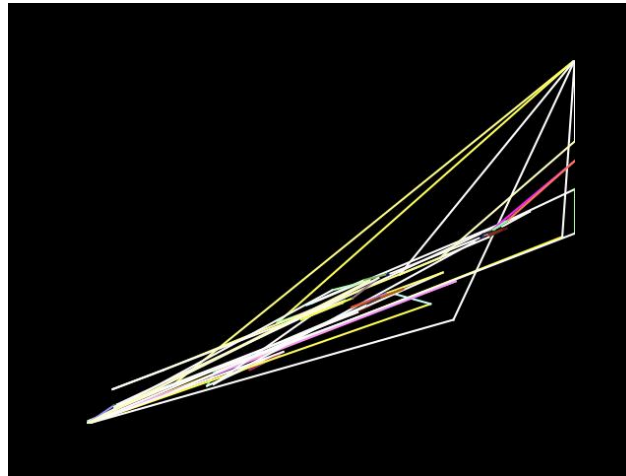
$$jerk = abs\left(\frac{acceleration_i - acceleration_{i-1}}{time_{frame}}\right)$$

#### 4.5.5 Οπτικοποίηση αποτελεσμάτων

Εφόσον έχουν συλλεχθεί τα δεδομένα από τον eye-tracker, θα χρησιμοποιηθούν για τη δημιουργία μιας εικόνας παρόμοιας με αυτές του dataset έτσι ώστε το μοντέλο να μπορεί να τις ταξινομήσει στις δύο κατηγορίες.

Αρχικά, οι κινήσεις στους άξονες x και y διαιρούνται με το πλάτος και το ύψος της οθόνης αντίστοιχα και στη συνέχεια πολλαπλασιάζονται με το θεμιτό πλάτος και ύψος έτσι ώστε να κανονικοποιηθούν στο πραγματικό μέγεθος της εικόνας. Στη συνέχεια, με μια επαναληπτική δομή σχεδιάζεται μια ευθεία μεταξύ κάθε σημείου (  $x_i, y_i$  ) με το επόμενο του (  $x_{i+1}, y_{i+1}$  ), η οποία χρωματίζεται με τιμές rgb (red - green - blue),

αντίστοιχη με τις τιμές της ταχύτητας, της επιτάχυνσης και του jerk. Στην εικόνα 20 φαίνεται παράδειγμα της εικόνας που δημιουργείται.

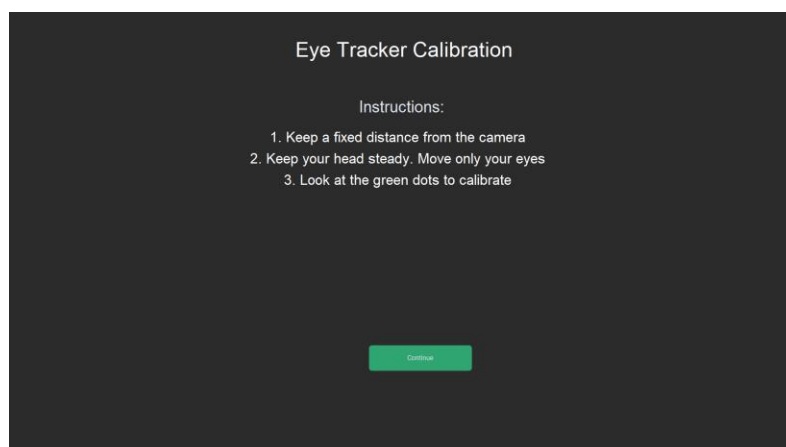


*Εικόνα 20 Παράδειγμα δημιουργημένης εικόνας*

## 5 Ανάλυση Κώδικα

### 5.1 Αρχικό παράθυρο

Για να δημιουργηθεί ένα αρχικό παράθυρο γραφικών (εικόνα 21) που θα παρέχει στο χρήστη οδηγίες για την διαδικασία calibration, χρησιμοποιήθηκε η βιβλιοθήκη customtkinter. Αρχικά ορίστηκε το theme ως σκούρο και πράσινο και το παράθυρο δημιουργήθηκε σε πλήρη οθόνη. Snapshot του κώδικα δημιουργίας του παραθύρου φαίνεται στην εικόνα 22.



*Εικόνα 21 παράδειγμα GUI του προγράμματος*

```
1 def start_window():
2     tk.set_appearance_mode("dark")
3     tk.set_default_color_theme("green")
4     _, screen_height = getScreenDim()
5     root = tk.Tk()
6     root.attributes("-fullscreen", True)
7     frame = tk.TkFrame(master=root)
8     frame.pack(fill="both", expand=True)
9
10    title = tk.TkLabel(
11        master=frame,
12        text="Eye Tracker Calibration",
13        font=("Arial", 40),
14        text_color="white",
15    )
16    title.pack(pady=50, padx=100)
17
18    label = tk.TkLabel(master=frame, text="Instructions: ", font=("Arial", 32))
19    label.pack(pady=20, padx=0)
20
21    instruct_1 = tk.TkLabel(
22        master=frame,
23        text="1. Keep a fixed distance from the camera ",
24        font=("Arial", 28),
25        text_color="white",
26    )
27    instruct_1.pack(pady=5, padx=0)
28
29    instruct_2 = tk.TkLabel(
30        master=frame,
31        text="2. Keep your head steady. Move only your eyes ",
32        font=("Arial", 28),
33        text_color="white",
34    )
35    instruct_2.pack(pady=5, padx=0)
36
37    instruct_3 = tk.TkLabel(
38        master=frame,
39        text="3. Look at the green dots to calibrate ",
40        font=("Arial", 28),
41        text_color="white",
42    )
43    instruct_3.pack(pady=5, padx=0)
44
45    exit_button = tk.TkButton(
46        master=frame,
47        text="Continue",
48        width=200,
49        height=50,
50        command=root.destroy,
51    )
52    exit_button.place(x=800, y=screen_height - (0.4 * screen_height), anchor="n")
53    root.mainloop()
```

Εικόνα 22 Snapshot από την δημιουργία του GUI

## 5.2 Calibration του Eye-tracker

Αρχικά αποθηκεύονται σε μεταβλητές τα σημεία που θα πραγματοποιηθεί το calibration, το πλάτος και το ύψος της οθόνης.

Η συνάρτηση `show_calibration_points` (εικόνα 23) δέχεται σαν ορίσματα το `frame` που θα εκτυπωθεί το σημείο ( `image` ), τις συντεταγμένες του σημείου που θα εκτυπωθεί (`calib_point`), τις συντεταγμένες του βλέμματος ( σε περίπτωση που δεν αρχικοποιηθούν ορίζονται ως `none` ) και την ακτίνα του κύκλου που θα λειτουργεί σαν ένδειξη για την θέση του βλέμματος. Η συνάρτηση δημιουργεί και επιστρέφει μια εικόνα η οποία έχει το μέγεθος της οθόνης και περιέχει το σημείο calibration ως έναν πράσινο κύκλο με ακτίνα 30 pixel, ενώ σε περίπτωση που οι συντεταγμένες του ματιού δεν είναι κενές έναν μπλε ημιδιαφανή κύκλο ακτίνας 80 pixel, ο οποίος λειτουργεί ως ένδειξη της θέσης του βλέμματος σχετικά με το σημείο.

```
1 screen_width, screen_height = Visuals.getScreenDim()
2
3
4 Calibration_Points = [
5     (int(screen_width / 2), int(screen_height / 2)), # Center point
6     (100, 100), # Top Left point
7     ((screen_width - 100), 100), # Top right Point
8     (100, (screen_height - 100)), # Bottom Left Point
9     ((screen_width - 100), (screen_height - 100)), # Bottom Right Points
10 ]
11
12
13 def show_calibration_points(image, calib_point, gaze_coords=None, blur_radius=80):
14     cv.namedWindow("Calibrating", cv.WND_PROP_FULLSCREEN)
15     cv.setWindowProperty("Calibrating", cv.WND_PROP_FULLSCREEN, cv.WINDOW_FULLSCREEN)
16     cv.circle(image, calib_point, 30, (0, 255, 0), -1)
17
18     if gaze_coords:
19
20         for x, y in gaze_coords:
21             mask = np.zeros_like(image)
22             cv.circle(mask, (int(x), int(y)), blur_radius, (255, 0, 0), -1)
23             blurred_mask = cv.GaussianBlur(
24                 mask, (2 * blur_radius + 1, 2 * blur_radius + 1), blur_radius
25             )
26             image = cv.addWeighted(image, 1.0, blurred_mask, 0.5, 0)
27
28     return image
29
```

Εικόνα 23 Snapshot συνάρτησης Show Calibration

Η συνάρτηση `calibrate_eye_tracker` (εικόνα 24) δέχεται σαν ορίσματα το μέσο που θα κάνει την καταγραφή (cam) και το πλέγμα προσώπου που θα περιέχει τις συντεταγμένες των σημείων του προσώπου και επιστρέφει τον πίνακα ομοιογένειας (3.4.2) του συστήματος. Για κάθε σημείο στα σημεία calibration αρχικοποιεί έναν πίνακα ο οποίος είναι υπεύθυνος για την συλλογή των δειγμάτων (30 δείγματα). Κάθε δείγμα που συλλέγεται αντιστοιχεί σε ένα frame το οποίο αντιστρέφεται (flip), μεταφράζεται από BGR σε RGB και στη συνέχεια επεξεργάζεται από το πλέγμα προσώπου. Στην συνέχεια συλλέγονται τα σημεία και υπολογίζονται οι συντεταγμένες του βλέμματος με την διαδικασία που περιγράφεται στο 3.4.3 και εκτυπώνεται το σημείο calibration και η θέση των ματιών στην οθόνη με την χρήση της συνάρτησης `show_calibration_points`. Για να προστεθούν τα σημεία στον τελικό πίνακα `eyes_coordinates` υπολογίζεται ο μέσος όρος των δειγμάτων που συλλεχθήκαν. Αν δεν συλλεχθήκαν δείγματα εκτυπώνεται σχετικό μήνυμα. Μόλις συλλεχθούν δείγματα για όλα τα σημεία calibration γίνεται έλεγχος αν υπάρχουν τουλάχιστον 4 έγκυρα σημεία και στην συνέχεια υπολογίζεται ο πίνακας ομοιογραφίας των σημείων που συλλεχθήκαν σε σχέση με τα σημεία calibration.

```

1 def calibrate_eye_tracker(cam, faceMesh):
2     eyes_coordinates = []
3
4     cv.namedWindow("Calibrating", cv.WND_PROP_FULLSCREEN)
5     cv.setWindowProperty("Calibrating", cv.WND_PROP_FULLSCREEN, cv.WINDOW_FULLSCREEN)
6
7     for point in Calibration_Points:
8         data_samples = []
9
10        for _ in range(30):
11            ret, frame = cam.read()
12
13            if not ret:
14                print("Didn't capture a frame!")
15                continue
16
17            flip_frame = cv.flip(frame, 1)
18            rgb_frame = cv.cvtColor(flip_frame, cv.COLOR_BGR2RGB)
19            output = faceMesh.process(rgb_frame)
20            landmarks_points = output.multi_face_landmarks
21
22            if landmarks_points:
23                landmarks = Landmarks_points[0].landmark
24
25                eye_1 = [
26                    (landmark_eye.x, landmark_eye.y)
27                    for landmark_eye in Landmarks[473:479]
28                ]
29                eye_2 = [
30                    (landmark_eye.x, landmark_eye.y)
31                    for landmark_eye in Landmarks[468:473]
32                ]
33
34                avg_x = (sum(x for x, _ in eye_1) + sum(x for x, _ in eye_2)) / (
35                    Len(eye_1) + Len(eye_2)
36                )
37                avg_y = (sum(y for _, y in eye_1) + sum(y for _, y in eye_2)) / (
38                    Len(eye_1) + Len(eye_2)
39                )
40
41                data_samples.append((avg_x, avg_y))
42
43                screen_x = int(avg_x * screen_width)
44                screen_y = int(avg_y * screen_height)
45
46                calib_image = np.zeros((screen_height, screen_width, 3), dtype=np.uint8
47            )
48            calib_image = show_calibration_points(
49                calib_image, point, [(screen_x, screen_y)]
50            )
51            cv.imshow("Calibrating", calib_image)
52            cv.waitKey(1)
53
54            if data_samples:
55                avg_coords = np.mean(data_samples, axis=0)
56                eyes_coordinates.append(avg_coords)
57            else:
58                print(f"No valid eye coordinates for point {point}.")
59
60            if Len(eyes_coordinates) < 4:
61                print("Not enough calibration data. Calibration failed.")
62                cv.destroyWindow("Calibrating")
63                return None
64
65            source_points = np.array(eyes_coordinates, dtype=np.float32)
66            destination_points = np.array(Calibration_Points, dtype=np.float32)
67
68            transform_matrix, _ = cv.findHomography(source_points, destination_points)
69
70            print("Calibration finished!")
71
72            cv.destroyWindow("Calibrating")
73
74            return transform_matrix
75

```

Εικόνα 24 Snapshot συνάρτησης calibration του Eye-tracker

### 5.3 Επιλογή κατάλληλων σημείων

Σύμφωνα με το [28] για να αποφευχθεί ο μεγάλος όγκος πληροφοριών στην εικόνα χρησιμοποιήθηκαν συντεταγμένες από 200 σημεία. Η συνάρτηση calcValidPoints (εικόνα 25) δέχεται σαν ορίσματα δύο πίνακες με τις κινήσεις των ματιών στους άξονες

x και y και μια μεταβλητή που αναπαριστά τον αριθμό των σημείων που θα επιλεγεί από τα δεδομένα που συλλέχθηκαν. Αρχικά ελέγχει αν ο αριθμός των ζητούμενων σημείων είναι μεγαλύτερος από τον αριθμό των σημείων που συλλέχθηκαν. Στην συνέχεια υπολογίζει το μέγεθος του βήματος διαιρώντας το μήκος ενός από τους δύο πίνακες με τον αριθμό των ζητούμενων σημείων. Τέλος προχωρώντας με το βήμα που υπολογίσθηκε προσθέτει τα αντίστοιχα σημεία σε 2 καινούριους πίνακες και τους επιστρέφει ( x\_valid, y\_valid ).

```
1 def calcValidPoints(x_movement, y_movement, num_points):
2
3     # If collected data < number of points return collected data
4     if num_points >= len(x_movement):
5         print("Not enough points")
6         return x_movement, y_movement
7
8     x_valid = []
9     y_valid = []
10
11     # Calculate the size of the step
12     step = len(x_movement) / num_points
13
14     for i in range(num_points):
15         # Calculate the int step of the index
16         index = int(i * step)
17
18         x_valid.append(x_movement[index])
19         y_valid.append(y_movement[index])
20
21     return x_valid, y_valid
```

*Εικόνα 25 Snapshot συνάρτησης επιλογής σημείων*

## 5.4 Υπολογισμός ταχύτητας, επιτάχυνσης, jerk

Οι 3 αυτές συναρτήσεις(εικόνα 26, εικόνα 27, εικόνα 28) εφαρμόζουν τους μαθηματικούς τύπους που αναλύθηκαν στο 3.4.4

```
1 def calculate_speed(x_movement, y_movement, time_frame):
2     speed_of_gaze = []
3     for i in range(1, Len(x_movement)):
4
5         cur_x = x_movement[i]
6         cur_y = y_movement[i]
7
8         prev_x = x_movement[i - 1]
9         prev_y = y_movement[i - 1]
10
11         # calculate euclidian distance
12         dist = np.sqrt((cur_x - prev_x) ** 2 + (cur_y - prev_y) ** 2)
13         cur_speed = dist / time_frame
14         speed_of_gaze.append(cur_speed)
15
16     return speed_of_gaze
```

*Εικόνα 26 Snapshot συνάρτησης υπολογισμού ταχύτητας*

```
1 def calculate_acceleration(speed_of_gaze, time_frame):
2     acceleration_of_gaze = []
3     for i in range(1, Len(speed_of_gaze)):
4
5         cur_acceleration = abs(speed_of_gaze[i] - speed_of_gaze[i - 1]) / time_frame
6         acceleration_of_gaze.append(cur_acceleration)
7
8     return acceleration_of_gaze
```

*Εικόνα 27 Snapshot συνάρτησης υπολογισμού επιτάχυνσης*

```
1 def calculate_jerk(acceleration_of_gaze, time_frame):
2     jerk_of_gaze = []
3
4     for i in range(1, Len(acceleration_of_gaze)):
5
6         cur_jerk = (
7             abs(acceleration_of_gaze[i] - acceleration_of_gaze[i - 1]) / time_frame
8         )
9         jerk_of_gaze.append(cur_jerk)
10
11     return jerk_of_gaze
```

*Εικόνα 28 Snapshot συνάρτησης υπολογισμού jerk*

## 5.5 Κυρίως κώδικας

Για λόγους ασφαλείας αλλά και για να είναι πιο εύκολη η χρήση διαφορετικού βίντεο η τοποθεσία που είναι αποθηκευμένο το βίντεο αποθηκεύεται σαν μια μεταβλητή περιβάλλοντος.

Αρχικά φορτώνονται οι μεταβλητές περιβάλλοντος και δημιουργείται το αντικείμενο της κάμερας που θα πραγματοποιήσει την καταγραφή αν αυτές οι δύο ενέργειες ολοκληρωθούν με επιτυχία εκτυπώνονται σχετικά μηνύματα.

Στην συνέχεια δημιουργείται το πλέγμα προσώπου και ανοίγει το παράθυρο που ενημερώνει τον χρήστη για της οδηγίες χρήσης του προγράμματος. Μόλις ο χρήστης πατήσει το σχετικό κουμπί που κλείνει το παράθυρο ξεκινάει η διαδικασία calibration του eye-tracker και ο πίνακας ομοιογραφίας που υπολογίζεται εκτυπώνεται στην οθόνη (εικόνα 29).

```
1 load_dotenv()
2 video_path = os.getenv("VIDEO_PATH")
3 feature_extract_path = os.getenv("FEATURE_EXTRACT")
4 classifier_path = os.getenv("CLASSIFIER")
5 print("Enviroment variables loaded")
6
7 print("Opening camera...")
8
9 cam = cv.VideoCapture(0)
10
11 if not cam.isOpened():
12     print("Error opening camera!")
13     exit()
14 else:
15     print("Camera opened successfully!")
16
17
18 faceMesh = mp.solutions.face_mesh.FaceMesh(refine_Landmarks=True)
19
20 print("Calibrating Eye Tracker...")
21 Visuals.start_window()
22 transform_matrix = Calibration.calibrate_eye_tracker(cam, faceMesh)
23 print("Generated Transform Matrix: ", transform_matrix)
24
25 cv.waitKey(2000)
26
```

*Εικόνα 29 Snapshot 1 main code*

Αφού περιμένει 2 δευτερόλεπτα φορτώνει το βίντεο εφόσον αυτό ολοκληρωθεί με επιτυχία ξεκινάει την καταγραφή των ματιών για την συλλογή των δεδομένων.



Καθώς η συνάρτηση GatherData επιστρέφει έναν πίνακα με tuples συντεταγμένων x και y χωρίζονται τα tuples σε δύο ξεχωριστούς πίνακες στην συνέχεια επιλέγονται 200 σημεία για την δημιουργία της εικόνας.(εικόνα 30)



```
1 print("Opening Video...")
2 video = cv.VideoCapture("Misc/test_video.mp4")
3
4 if not video.isOpened():
5     print("Error opening video!")
6     exit()
7 else:
8     print("Video opened successfully!")
9
10 gaze_coordinates, windowHeight, windowWidth, video_length = dc.GatherData(
11     cam, faceMesh, video, transform_matrix
12 )
13
14 x_movement = []
15 y_movement = []
16
17 cap = Formulas.calculateCap(windowWidth, windowHeight)
18
19 for x, y in gaze_coordinates:
20     x_movement.append(x)
21     y_movement.append(y)
22
23 x_valid, y_valid = Formulas.calcValidPoints(x_movement, y_movement, 200)
24
```

*Εικόνα 30 Snapshot 2 main code*

Υπολογίζεται ο χρόνος μεταξύ 2 διαδοχικών σημείων αλλά και η ταχύτητα, επιτάχυνση και jerk. Για να κανονικοποιηθούν σε ένα συγκεκριμένο threshold που σύμφωνα με το [28] έχει οριστεί ως το  $\frac{1}{4}$  της διαγωνίου της οθόνης, χρησιμοποιείται η συνάρτηση normalizeToCap, η οποία διαιρεί την αριθμητική τιμή με το threshold.

Για να έχουν όλοι οι πίνακες τις ίδιες διαστάσεις, ο πίνακας της επιτάχυνσης μετακινείται μια θέση δεξιά ενώ ο πίνακας jerk μετακινείται δύο θέσεις δεξιά.(εικόνα 31)

```
1 frames = video_length / len(x_valid)
2 time_frame = 1 / frames
3
4 speed_of_gaze = Formulas.calculate_speed(x_valid, y_valid, time_frame)
5
6 acceleration_of_gaze = Formulas.calculate_acceleration(speed_of_gaze, time_frame)
7
8 jerk_of_gaze = Formulas.calculate_jerk(acceleration_of_gaze, time_frame)
9
10 speed_capped = []
11 acceleration_capped = []
12 jerk_capped = []
13
14 for s in speed_of_gaze:
15     speed_capped.append(Formulas.normalizeToCap(s, cap))
16
17 for a in acceleration_of_gaze:
18     acceleration_capped.append(Formulas.normalizeToCap(a, cap))
19
20 for j in jerk_of_gaze:
21     jerk_capped.append(Formulas.normalizeToCap(j, cap))
22
23 acceleration_capped = [0] + acceleration_capped
24 jerk_capped = [0, 0] + jerk_capped
25
```

*Εικόνα 31 Snapshot 3 main code*

Μόλις έχει πραγματοποιηθεί ο υπολογισμός όλων των απαραίτητων μεταβλητών δημιουργείται η εικόνα και έπειτα προεπεξεργάζεται ( normalize, resize ).

Σε περίπτωση που η εικόνα είναι 4 καναλιών ( RGB – A ) δηλαδή έχει και σαν κανάλι χρώματος την διαφάνεια μετατρέπεται σε 3 κανάλια χρώματος.(εικόνα 32)

```
1 print("\nCreating image from data...")
2 Visuals.createImage(x_valid, y_valid, speed_capped, acceleration_capped, jerk_capped)
3
4 img = Image.open("Misc/test_image.png")
5 print("Image loaded!")
6 img = np.array(img)
7
8 image = Preprocess.normalize(img)
9 image = Preprocess.resize(image)
10
11 print("Image preprocessed!")
12
13 if image.shape[-1] == 4: # RGBA to RGB
14     image = image[..., :3]
15
16
17 image = np.expand_dims(image, axis=0)
18
19 feature_extractor = load_model("Model/feature_extractor.h5")
20 model = load("Model/rf_classifier.pkl")
21 print("Model loaded!")
22
23 image_features = feature_extractor.predict(image)
24 image_features = image_features.reshape(image_features.shape[0], -1)
25 print("Features extracted!")
26
27 predict = model.predict(image_features)
28 proba = model.predict_proba(image_features)
29 proba = np.array(proba)
30
31 print("=====")
32 print("\t\tPrediction: ", predict, "\n\t\tProbability: ", np.max(proba))
33 print("=====")
```

*Εικόνα 32 Snapshot 4 main code*

Τέλος φορτώνονται τα δύο μοντέλα ( για την εξαγωγή των χαρακτηριστικών και για την ταξινόμηση ) και με δεδομένο εισόδου την εικόνα που έχει δημιουργηθεί από τα δεδομένα που συλλέχθηκαν πραγματοποιείται ταξινόμηση του χρήστη ως άτομο με ΔΑΦ ή νευροτυπικό και εκτυπώνεται και η εμπιστοσύνη στο αποτέλεσμα.(εικόνα 33)

```
Features extracted!  
=====  
Prediction: ['Non-ASD']  
Probability: 0.7647558158346971  
=====  
PS: GET ASD
```

Εικόνα 33 Παράδειγμα αποτελέσματος

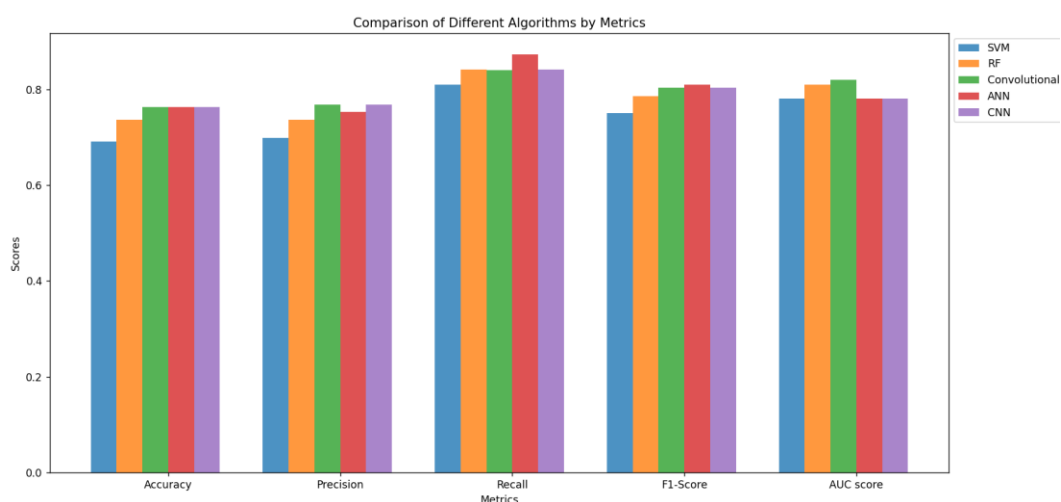
## 6 Αποτελέσματα

### 6.1 Επιλογή μοντέλου μηχανικής μάθησης

Στον πίνακα 5 φαίνονται οι διαφορετικοί αλγόριθμοι που χρησιμοποιήθηκαν και οι αποδόσεις τους στα διάφορα μετρικά που επιλέχθηκαν και στην εικόνα 34 φαίνονται οπτικοποιημένα αυτά τα αποτελέσματα.

Metrics	Random Forest	SVM	ANN	CNN	Conv Random Forest
Accuracy	0.736	0.691	0.764	0.764	0.764
Precision	0.736	0.698	0.753	0.768	0.768
Recall	0.841	0.809	0.873	0.841	0.841
F1-Score	0.785	0.75	0.808	0.803	0.803
AUC Score	0.81	0.78	0.78	0.78	0.82
Prediction time	0.0002	0.035	0.002	0.013	0.0006

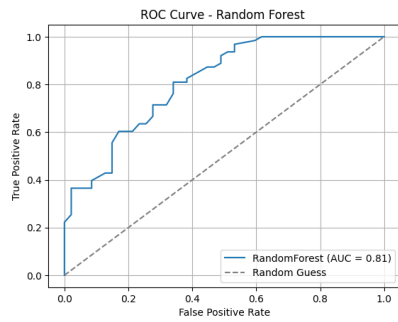
Πίνακας 5 Αποτελέσματα των αλγορίθμων



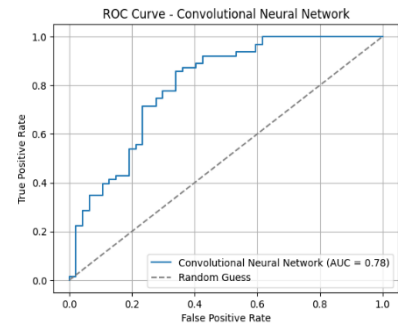
Εικόνα 34 Μετρικά των διαφορετικών αλγορίθμων

Από την παραπάνω εικόνα παρατηρούμε πως όλοι οι αλγόριθμοι βρίσκονται σχετικά κοντά στα μετρικά με τους αλγορίθμους Random forest και Convolutional Random Forest να παρουσιάζουν καλύτερο AUC. Επιπλέον ο Convolutional Random Forest μαζί με τους CNN και ANN παρουσιάζουν την καλύτερη ακρίβεια και precision. Ενώ ο ANN παρουσιάζει καλύτερα F1-score και recall.

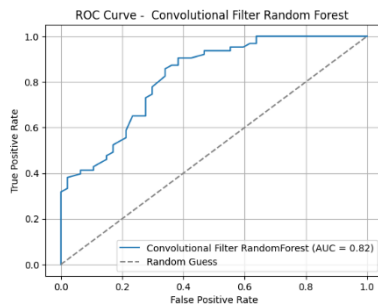
Στις εικόνες 35-39 φαίνονται οι καμπύλες ROC των διαφορετικών μοντέλων μας.



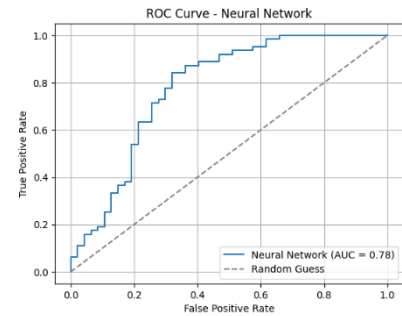
**Εικόνα 37 Καμπύλη ROC Random Forest**



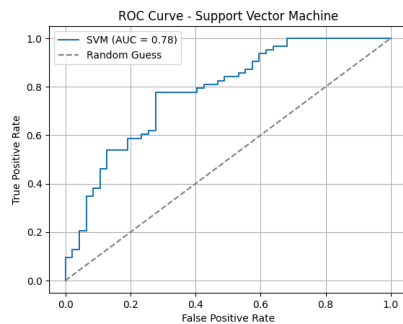
**Εικόνα 35 Καμπύλη ROC CNN**



**Εικόνα 38 Καμπύλη ROC conv Random Forest**



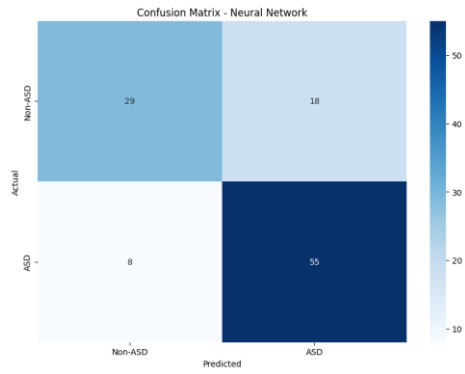
**Εικόνα 36 Καμπύλη ROC ANN**



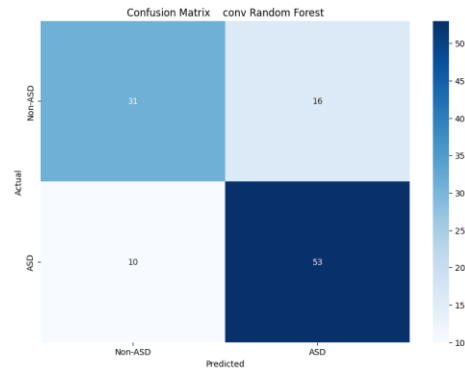
**Εικόνα 39 Καμπύλη ROC SVM**

Από αυτές τις καμπύλες σε συνδυασμό με το AUC SCORE μας παρατηρούμε ότι τα μοντέλα με την μέθοδο ταξινόμησης Random Forest κάνουν καλύτερη διάκριση μεταξύ των κλάσεων καθώς η καμπύλη τους είναι πιο κοντά στην μονάδα συγκριτικά με τα υπόλοιπα μοντέλα που η καμπύλη τους είναι ανάμεσα στην μονάδα και στην ευθεία 0.5 που δηλώνει μεγαλύτερη τυχαιότητα.

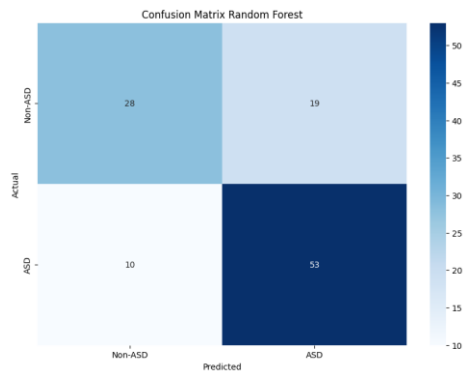
Στις εικόνες 40-44 φαίνονται συγκριτικά οι πίνακες σύγχυσης των διαφόρων μοντέλων μας



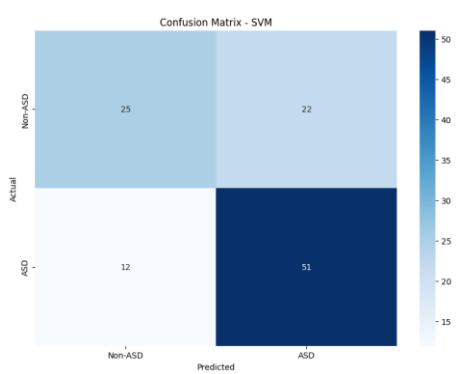
*Εικόνα 40 Πίνακας σύγκρισης ANN*



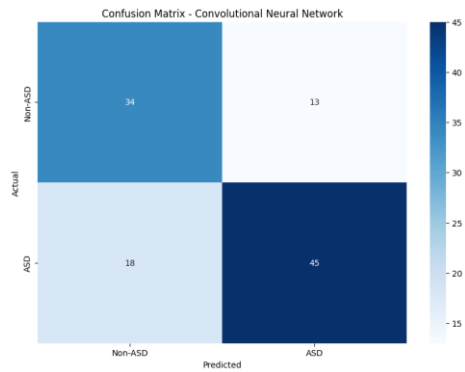
*Εικόνα 41 Πίνακας σύγκρισης conv Random Forest*



*Εικόνα 42 Πίνακας σύγκρισης Random Forest*



*Εικόνα 43 Πίνακας σύγκρισης SVM*



*Εικόνα 44 Πίνακας σύγκρισης CNN*

Μέσω των πινάκων σύγκρισης γίνεται αντιληπτό ότι ενώ όλα τα μοντέλα μπορούν να ταξινομήσουν σωστά όταν ένα άτομο έχει στην πραγματικότητα ΔΑΦ το μοντέλο convolutional Random Forest διατηρεί σχετικά μεγαλύτερη ακρίβεια όταν το άτομο είναι νευροτυπικό.

Παρατηρούμε πως όλοι οι αλγόριθμοι αποδίδουν παρόμοια στα παραπάνω μετρικά όμως ο Convolutional Random Forest είναι ελαφρώς καλύτερος συνολικά αυτός είναι

και ο λόγος που θα είναι ο αλγόριθμος που θα χρησιμοποιηθεί για την ταξινόμηση στο πείραμά μας.

## 6.2 Παραμετροποίηση μοντέλου

Το μοντέλο Convolutional Random Forest αποτελείται από δύο διαφορετικά μοντέλα. Το πρώτο είναι ένα CNN το οποίο κάνει την εξαγωγή των χαρακτηριστικών και έναν random forest classifier ο οποίος κάνει το classification.

Για την εξαγωγή των χαρακτηριστικών από τις εικόνες θα χρησιμοποιήσουμε το προ-εκπαιδευμένο μοντέλο VGG16 ( Visual Geometry Group ) [19] ένα μοντέλο βαθιάς μάθησης το οποίο αναπτύχθηκε στο πανεπιστήμιο της Οξφόρδης και αποτελείται από 16 επίπεδα. Τα πρώτα 13 επίπεδα είναι συνελκτικά ενώ τα τελευταία 3 είναι πλήρως συνδεδεμένα. Το μοντέλο έχει συνολικά 138 εκατομμύρια παραμέτρους αλλά συνήθως χρησιμοποιείται για εφαρμογές μεταφοράς μάθησης ( transfer learning ) δηλαδή είναι ήδη προ εκπαιδευμένο και τα τελευταία στρώματα ξανά εκπαιδεύονται έτσι ώστε να προσαρμοστούν στο εκάστοτε πρόβλημα.

Αρχικά για να γίνει fine-tune το μοντέλο για το dataset που θα χρησιμοποιηθεί προστίθεται ένα GlobalAveragePooling2D στρώμα το οποίο χρησιμοποιείται για την μείωση των διαστάσεων των χαρακτηριστικών και την δημιουργία ενός απλούστερου διανύσματος για κάθε χαρακτηριστικό και ένα Dense ( πλήρως συνδεδεμένο ) στρώμα ενός νευρώνα για να πραγματοποιήσει την ταξινόμηση.

Στην συνέχεια τα βάρη του κάθε νευρώνα για τα πρώτα 12 στρώματα ορίζονται ως μη εκπαιδευσιμα έτσι ώστε τα μόνα στρώματα που θα εκπαιδευτούν να είναι αυτά τα οποία εξάγουν τα πιο σύνθετα χαρακτηριστικά από τις εικόνες. Το μοντέλο εκπαιδεύεται για 15 epochs με τον adam optimizer και ένα μικρό learning rate (0.00001) έτσι ώστε να αποφευχθεί η υπερκπαίδευση.

Η εικόνα 45 δείχνει την ακρίβεια και την απώλεια κατά την διάρκεια της εκπαίδευσης



Εικόνα 45 Εκπαίδευση pre-trained μοντέλου στα δεδομένα

Όταν πραγματοποιηθεί η εκπαίδευση του μοντέλου πάνω στο dataset αφαιρούνται τα τελευταία δύο στρώματα τα οποία είναι υπεύθυνα για την ταξινόμηση και στη θέση τους μπαίνει ένας random forest classifier, ο οποίος αντί να εκπαιδευτεί στις εικόνες του dataset εκπαιδεύεται πάνω στα δεδομένα που εξάγονται από το VGG16.

Ο πίνακας 6 περιέχει την απόδοση του fine tuned μοντέλου( πάνω στο test-dataset ) στα μετρικά που χρησιμοποιήθηκαν

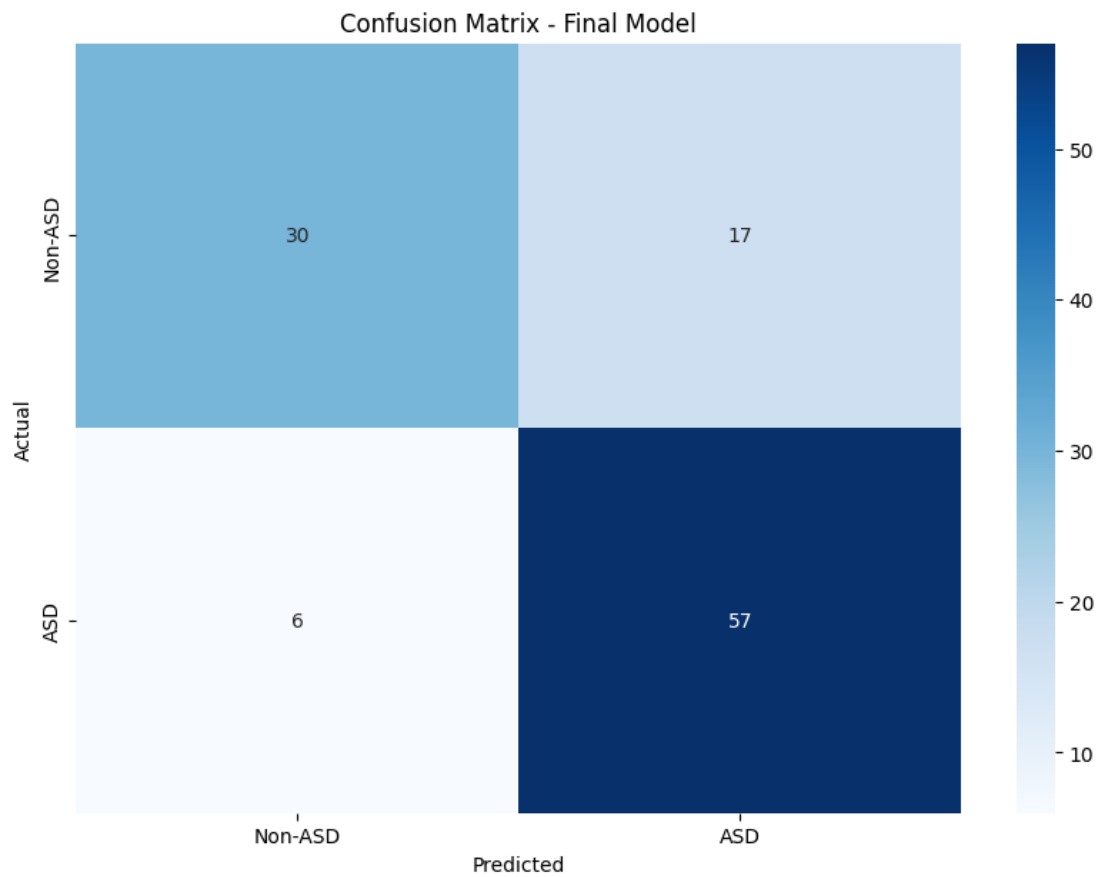
Metrics	Score
Accuracy	79.09
Precision	77.03
Recall	90.47
F1-Score	83.21
AUC-Score	0.88

Πίνακας 6 Αποτελέσματα Fine tuning

Παρατηρείται μια γενική βελτίωση σε όλα τα μετρικά με την πιο σημαντική να φαίνεται στο AUC Score.

Στην εικόνα 46 φαίνεται ο πίνακας σύγκρισης του μοντέλου.

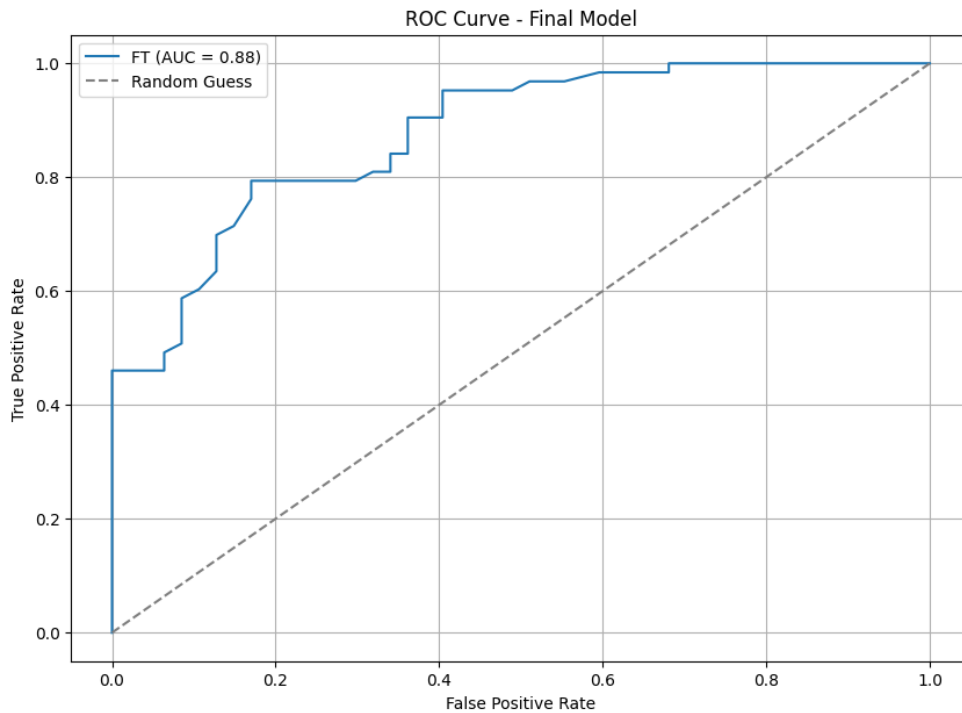




*Εικόνα 46 Πίνακας σύγκρισης Fine Tuned Model*

Από τον πίνακα σύγκρισης βγαίνει το συμπέρασμα πως και αυτό το μοντέλο έχει την τάση να διακρίνει πιο εύκολα την ΔΑΦ ταξινομώντας, ενώ δυσκολεύεται όταν ένα άτομο είναι νευροτυπικό ταξινομώντας περίπου 1 στις 3 περιπτώσεις λάθος.

Στην εικόνα 47 φαίνεται η καμπύλη ROC αλλά και το AUC Score.



Εικόνα 47 Καμπύλη ROC Fine Tuned Model

Φαίνεται πως εκτός από σημαντική βελτίωση στο AUC η καμπύλη ROC εμφανίζεται και πιο ομαλή υποδηλώνοντας έτσι πιο σταθερή απόδοση.

Στη συνέχεια, αφού έχει πραγματοποιηθεί η ρύθμιση του μοντέλου εξαγωγής χαρακτηριστικών, θα πραγματοποιηθεί ρύθμιση υπερπαραμέτρων στον ταξινομητή έτσι ώστε να αξιολογηθεί αν υπάρχουν περιθώρια βελτίωσης στο συνολικό μοντέλο.

Για να γίνει ο η ρύθμιση υπερπαραμέτρων θα χρησιμοποιηθεί η τεχνική Grid Search η οποία βρίσκει τις βέλτιστες τιμές υπερπαραμέτρων πραγματοποιώντας αναζήτηση μέσα σε ένα πλέγμα δοκιμάζοντας όλους τους πιθανούς συνδυασμούς αξιολογώντας τους με το επιλεγμένο μετρικό.

Στον πίνακα 7 φαίνεται το πλέγμα που δημιουργήθηκε για την αναζήτηση.

Υπερπαραμέτρος	Τιμή
<b>N_estimators</b>	<b>100, 200, 300</b>
<b>Max_depth</b>	<b>None, 10, 20</b>
<b>Min_samples_split</b>	<b>2, 5, 7</b>
<b>Min_sample_leaf</b>	<b>1, 2, 5</b>

Πίνακας 7 Πλέγμα αναζήτησης παραμέτρων

- 

Η τεχνική Grid Search θα δοκιμάσει  $3 * 3 * 3 * 3 = 81$  συνδυασμούς κάνοντας τους 5 φορές διασταυρωμένη επικύρωση ( μια τεχνική αξιολόγησης της εκπαίδευσης του μοντέλου η οποία χωρίζει τα δεδομένα σε 5 υποσύνολο από τα οποία τα 4 χρησιμοποιούνται για εκπαίδευση και το ένα για επικύρωση η διαδικασία

επαναλαμβάνεται 5 φορές έτσι ώστε το κάθε υποσύνολο να χρησιμοποιηθεί για επικύρωση) οπότε θα τρέξει 405 φορές.

Όπως παρουσιάζεται στον πίνακα 8, η τεχνική Grid Search έδειξε πως οι βέλτιστες τιμές για αυτό το πρόβλημα είναι:

Υπερπαράμετρος	Τιμή
<b>N_estimators</b>	<b>300</b>
<b>Max_depth</b>	<b>10</b>
<b>Min_samples_split</b>	<b>5</b>
<b>Min_sample_leaf</b>	<b>2</b>

*Πίνακας 8 Βέλτιστες παράμετροι*

Στην συνέχεια θα χρησιμοποιηθεί ο VGG16 για την εξαγωγή των χαρακτηριστικών από τις εικόνες και ο random forest με τις βέλτιστες υπερπαραμέτρους θα εκπαιδευτεί πάνω σε αυτά τα χαρακτηριστικά.

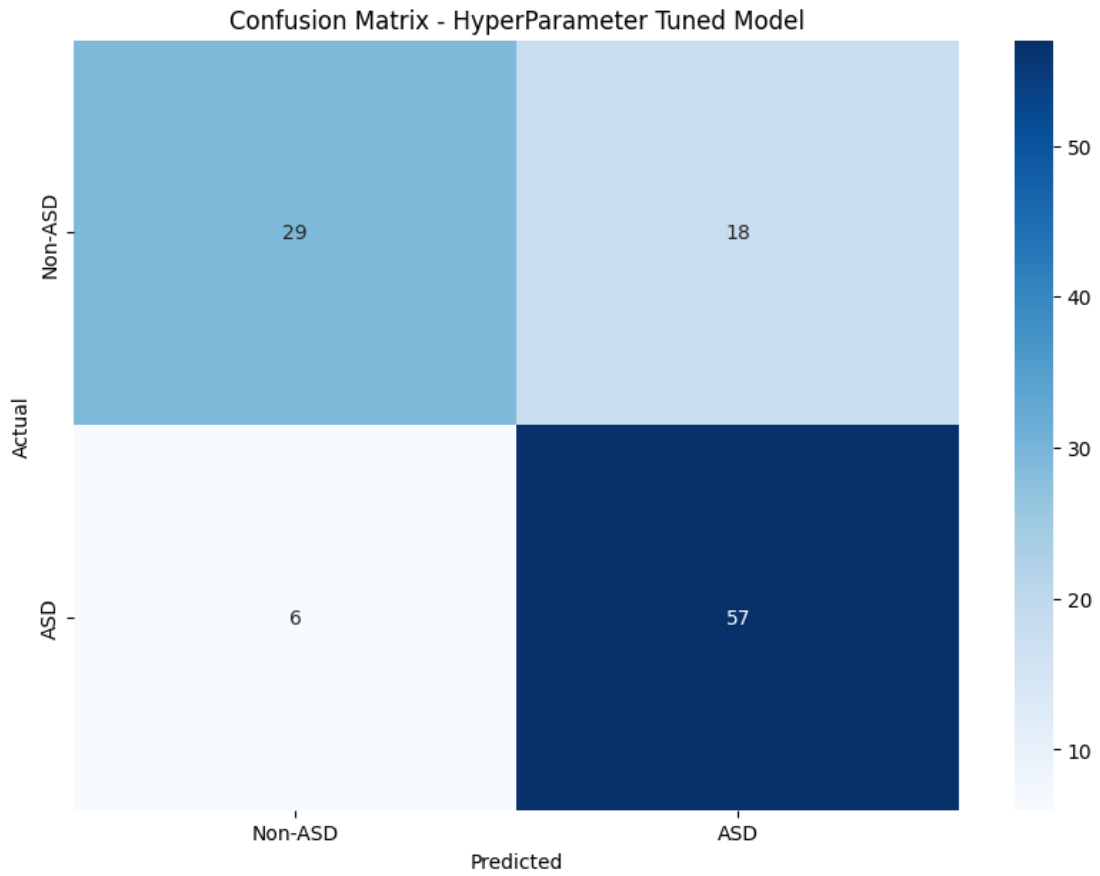
Ο πίνακας 9 περιέχει την απόδοση του fine tuned μοντέλου(πάνω στο test-dataset) στα μετρικά που χρησιμοποιήθηκαν

Metrics	Score
<b>Accuracy</b>	78.18
<b>Precision</b>	76.03
<b>Recall</b>	90.48
<b>F1-Score</b>	82.61
<b>AUC-Score</b>	0.86

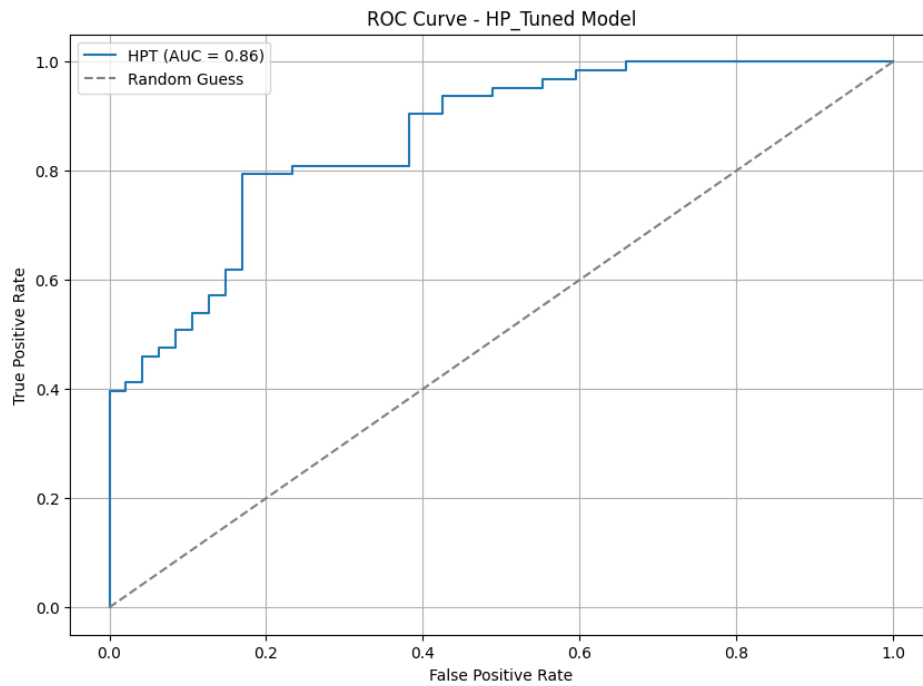
*Πίνακας 9 Αποτελέσματα Hyper Parameter Tuning*

Παρατηρείται μια βελτίωση των αποτελεσμάτων συγκριτικά με τα αρχικά μοντέλα αλλά τα αποτελέσματα συγκριτικά με το ρυθμισμένο VGG16 είναι ελαφρώς κατώτερα.

Στην εικόνα 48 φαίνεται ο πίνακας σύγκυσης του μοντέλου ενώ στην εικόνα 49 φαίνεται η καμπύλη ROC αλλά και το AUC Score.



*Εικόνα 48 Πίνακας σύγκρισης Hyper Parameter Tuned Model*



*Εικόνα 49 Καμπύλη ROC Hyper Parameter Tuned Model*

Όπως και πριν παρατηρείται σημαντική βελτίωση στο AUC συγκριτικά με τα βασικά μοντέλα αλλά το ρυθμισμένο VGG16 μοντέλο υπερισχύει.

## 7 Συμπεράσματα

Η έρευνα επικεντρώθηκε στην εφαρμογή αλγορίθμων μηχανικής μάθησης σε συνδυασμό με την τεχνολογία eye-tracking για την βοήθεια της έγκαιρης διάγνωσης της ΔΑΦ σε παιδιά. Στην έρευνα αυτή δοκιμάστηκαν 5 αλγόριθμοι ( Support Vector machine, Random Forest, Convolutional Neural Network, Artificial Neural Network, Random Forest with convolutional filter ), χρησιμοποιώντας ως μετρικά για την αξιολόγηση τους τα accuracy, precision, recall, f1-scores, roc curve, auc score, confusion matrix. Μετά από εκτενή ανάλυση επιλέχθηκε ο Random Forest with convolutional filter ως ο πιο αποδοτικός καθώς παρόλο που οι περισσότεροι αλγόριθμοι είχαν παρόμοια αποτελέσματα σε κάποια μετρικά, αυτός ήταν ο πιο συνεπής στα αποτελέσματα. Πιο συγκεκριμένα συγκρίνοντας όλα τα μετρικά μεταξύ τους βλέπουμε πως ο Random Forest with convolutional filter ήταν συνολικά πιο αποτελεσματικός κατά 18.95% από τον SVM, 10.27% από τον Random Forest, 8.51% από τον ANN και 4.62% από τον CNN.

Με τη χρήση ενός pretrained deep-learning μοντέλου για την εξαγωγή χαρακτηριστικών από τα δεδομένα της εικόνας διασφαλίστηκε η ακρίβεια στην εξαγωγή των χαρακτηριστικών. Για τη βελτίωση αυτών των αποτελεσμάτων κάποια στρώματα του pretrained μοντέλου εκπαιδεύτηκαν πάνω στα δεδομένα του προβλήματος έτσι ώστε να κατανοηθούν καλύτερα οι διαφορές στα μοτίβα κινήσεων των ματιών μεταξύ παιδιών με ΔΑΦ και νευροτυπικών παιδιών. Με πρόθεση να βρεθούν οι βέλτιστες υπερπαραμέτροι, ώστε να προσαρμοστεί ο αλγόριθμος ταξινόμησης στα δεδομένα του προβλήματος έγινε ρύθμιση υπερπαραμέτρων χωρίς να επιφέρει κάποια δραστική αλλαγή στα αποτελέσματα. Μετά την βελτιστοποίηση του αλγορίθμου βρέθηκε πως το τελικό μας μοντέλο είναι κατά 2.4% πιο αποδοτικό από τον ήδη καλύτερο αλγόριθμο μεταξύ των 5 έχοντας AUC = 88% και recall = 90%.

Σε αυτή την μελέτη, χρησιμοποιήθηκε το dataset των Carette et al. [28] με στόχο την ανάπτυξη ενός μοντέλου που συνδυάζει τον αλγόριθμο Random Forest με Convolutional Filters, επιδιώκοντας την ακριβή κατηγοριοποίηση των συμμετεχόντων σε νευροτυπικούς και άτομα με ΔΑΦ. Το μοντέλο κατάφερε να πετύχει AUC=88%, f1 score=83.12% και recall=90%. Παρόλο που η ακρίβεια του μοντέλου είναι ελαφρώς χαμηλότερη από το 95% που επιτεύχθηκε από τους Zhang et al. [32], οι οποίοι συνδύασαν τα δεδομένα του ίδιου dataset με δεδομένα EEG, τα αποτελέσματα του παρόντος μοντέλου είναι ιδιαίτερα αξιολόγη, ειδικά ως προς το recall (90%), το οποίο διασφαλίζει υψηλή ευαισθησία στην ανίχνευση ατόμων με ΔΑΦ. Επιπλέον, σε σύγκριση με την έρευνα των Alcaniz και συνεργατών [32], οι οποίοι επίσης χρησιμοποίησαν το ίδιο δεδομένα καθώς και εικονική πραγματικότητα σημειώνοντας ακρίβεια 86% με recall 91%, η μέθοδός μας προσφέρει πλεονεκτήματα όσον αφορά στην ταχύτητα εκπαίδευσης και στην αποδοτικότητα. Τέλος η ανάπτυξη και χρήση του

απλού εργαλείου eye-tracking κάνει την έρευνα πιο προσιτή καθώς δεν απαιτείται εξειδικευμένος εξοπλισμός, δίνοντας έτσι κίνητρο και σε άλλους ερευνητές να ασχοληθούν με τη μελέτη της συμπεριφοράς των ατόμων με ΔΑΦ και την εκτέλεση περισσότερων πειραμάτων με σκοπό την διεύρυνση της βάσης γνώσης γύρω από τις διαταραχές του αυτιστικού φάσματος.

Το εργαλείο που χρησιμοποιήθηκε για τη συλλογή των δεδομένων ήταν ένας video based eye-tracker, ο οποίος δημιουργήθηκε στα πλαίσια αυτής της μελέτης, έτσι ώστε να δοκιμαστεί ο αλγόριθμος σε δεδομένα εκτός του πλαισίου εκπαίδευσης. Ο eye-tracker παρόλο που κατείχε μόνο βασικές λειτουργίες δηλαδή ανίχνευση της θέσης των ματιών στην οθόνη και υπολογισμός της ταχύτητας και των παραγώγων της έδειξε θετικά αποτελέσματα αυτής της μεθόδου για την διάγνωση της ΔΑΦ σε παιδιά.

Παρά τα θετικά αποτελέσματα η συγκεκριμένη έρευνα έχει κάποιους περιορισμούς. Αρχικά η τεχνολογία eye-tracking και ειδικά ή απλουστευμένη έκδοση που χρησιμοποιήθηκε στη μελέτη έχει περιθώρια βελτίωσης. Αρχικά, αυτά θα μπορούσαν να επικεντρωθούν στην ακριβή ανίχνευση των κινήσεων των ματιών σε διαφορετικά περιβάλλοντα και συνθήκες φωτισμού και συγχρόνως στην ελαχιστοποίηση της παρεμβατικότητας στο χρήστη, καθώς οι υπάρχουσες τεχνικές υστερούν σε τουλάχιστον ένα από τα δύο. Ένας ακόμα περιορισμός είναι η περιορισμένη ποσότητα δεδομένων που χρησιμοποιήθηκαν για την εκπαίδευση των μοντέλων, γεγονός που συνδέεται με πιθανά προβλήματα στην γενίκευση των αποτελεσμάτων σε διαφορετικά δεδομένα π.χ. μεγαλύτεροι ηλικιακοί πληθυσμοί. Καθώς η εκπαίδευση μοντέλων μηχανικής μάθησης γίνεται με τη χρήση πολυάριθμων και πολύπλοκων μαθηματικών πράξεων και την επεξεργασία τεράστιου όγκου δεδομένων, ένας σημαντικός περιορισμός αποτελεί η υπολογιστική δύναμη του συστήματος που χρησιμοποιείται για την εκπαίδευση

Μελλοντικά οι ερευνητές θα μπορούσαν να εστιάσουν στη συλλογή περισσότερων δεδομένων, κατάλληλων για την εκπαίδευση αλγορίθμων καθώς και στη χρήση πιο εξελιγμένων αλγορίθμων όπως π.χ. βαθιά νευρωνικά δίκτυα ικανά για την επεξεργασία πολυδιάστατων δεδομένων για τη βελτίωση της ακρίβειας των αποτελεσμάτων. Επιπλέον, είναι σημαντική η επένδυση σε πιο δυνατές υπολογιστικές μονάδες ή και η χρήση υπολογιστικής νέφους (cloud computing) για τη βελτίωση των χρόνων εκπαίδευσης των μοντέλων. Τέλος, η παραπάνω έρευνα θα μπορούσε να πραγματοποιηθεί για την καλύτερη υλοποίηση του eye-tracker και την προσθήκη περισσότερων και πιο σύνθετων χαρακτηριστικών (π.χ. αναγνώριση βλεφαρίσματος).

Με βάση τα ευρήματα της παρούσας μελέτης αλλά και αντίστοιχων ερευνών η τεχνολογία eye - tracking μπορεί να αποτελέσει ένα ισχυρό συμπληρωματικό εργαλείο για την κλινική διάγνωση της ΔΑΦ ειδικά σε νεαρές ηλικίες παρέχοντας μέχρι στιγμής μια από τις πιο ταχείες, αντικειμενικές αλλά και λιγότερο επεμβατικές μεθόδους για τον εντοπισμό συμπεριφορών συσχετιζόμενες με το φάσμα του αυτισμού.

## 8 Επίλογος

Οι προοπτικές της χρήσης αυτής της τεχνολογίας σε συνεργασία με τον ταχέως ανερχόμενο τομέα της μηχανικής μάθησης ανοίγουν νέους δρόμους για την κατανόηση και διάγνωση της διαταραχής, προσφέροντας δυνατότητες για μια πιο έγκαιρη αλλά και εξατομικευμένη παρέμβαση έτσι ώστε να μειθούν όσο το δυνατόν περισσότερο τα συμπτώματα που σχετίζονται με αυτή τη διαταραχή.

## 9 Αναφορές

- [1] E. Fombonne, “Epidemiology of Pervasive Developmental Disorders,” *Pediatr. Res.*, vol. 65, no. 6, pp. 591–598, Jun. 2009, doi: 10.1203/PDR.0b013e31819e7203.
- [2] O. Whooley, “Diagnostic and Statistical Manual of Mental Disorders ( DSM ),” in *The Wiley Blackwell Encyclopedia of Health, Illness, Behavior, and Society*, 1st ed., W. C. Cockerham, R. Dingwall, and S. Quah, Eds., Wiley, 2014, pp. 381–384. doi: 10.1002/9781118410868.wbehibs011.
- [3] H. Cohen, M. Amerine-Dickens, and T. Smith, “Early Intensive Behavioral Treatment: Replication of the UCLA Model in a Community Setting,” *J. Dev. Behav. Pediatr.*, vol. 27, no. Supplement 2, pp. S145–S155, Apr. 2006, doi: 10.1097/00004703-200604002-00013.
- [4] M. Helt *et al.*, “Can Children with Autism Recover? If So, How?,” *Neuropsychol. Rev.*, vol. 18, no. 4, pp. 339–366, Dec. 2008, doi: 10.1007/s11065-008-9075-9.
- [5] O. I. Lovaas, “Behavioral treatment and normal educational and intellectual functioning in young autistic children,” *J. Consult. Clin. Psychol.*, vol. 55, no. 1, pp. 3–9, 1987, doi: 10.1037/0022-006X.55.1.3.
- [6] B. M. Prizant, “Language Acquisition and Communicative Behavior in Autism: Toward an Understanding of the ‘Whole’ of It,” *J. Speech Hear. Disord.*, vol. 48, no. 3, pp. 296–307, Aug. 1983, doi: 10.1044/jshd.4803.296.
- [7] L. K. Koegel, “Interventions to facilitate communication in autism,” *J. Autism Dev. Disord.*, vol. 30, no. 5, pp. 383–391, Oct. 2000, doi: 10.1023/a:1005539220932.
- [8] R. Landa, “Early communication development and intervention for children with autism,” *Ment. Retard. Dev. Disabil. Res. Rev.*, vol. 13, no. 1, pp. 16–25, 2007, doi: 10.1002/mrdd.20134.
- [9] “Educating Children with Autism | The National Academies Press.” Accessed: Sep. 09, 2024. [Online]. Available: <https://nap.nationalacademies.org/catalog/10017/educating-children-with-autism>
- [10] L. K. Koegel, R. L. Koegel, K. Ashbaugh, and J. Bradshaw, “The importance of early identification and intervention for children with or at risk for autism spectrum disorders,” *Int. J. Speech Lang. Pathol.*, vol. 16, no. 1, pp. 50–56, Feb. 2014, doi: 10.3109/17549507.2013.861511.
- [11] K.-F. Kollias, C. K. Syriopoulou-Delli, P. Sarigiannidis, and G. F. Fragulis, “The Contribution of Machine Learning and Eye-Tracking Technology in Autism Spectrum Disorder Research: A Systematic Review,” *Electronics*, vol. 10, no. 23, p. 2982, Nov. 2021, doi: 10.3390/electronics10232982.

- [12] D. C. Brien *et al.*, “Classification and staging of Parkinson’s disease using video-based eye tracking,” *Parkinsonism Relat. Disord.*, vol. 110, p. 105316, Dec. 2023, doi: 10.1016/j.parkreldis.2023.105316.
- [13] M. Wedel and R. Pieters, “Eye Tracking for Visual Marketing,” *Found. Trends® Mark.*, vol. 1, no. 4, pp. 231–320, 2006, doi: 10.1561/17000000011.
- [14] M.-L. Lai *et al.*, “A review of using eye-tracking technology in exploring learning from 2000 to 2012,” *Educ. Res. Rev.*, vol. 10, pp. 90–115, Dec. 2013, doi: 10.1016/j.edurev.2013.10.001.
- [15] P. Corcoran, F. Nanu, S. Petrescu, and P. Bigioi, “Real-time eye gaze tracking for gaming design and consumer electronics systems,” *IEEE Trans. Consum. Electron.*, vol. 58, no. 2, pp. 347–355, May 2012, doi: 10.1109/TCE.2012.6227433.
- [16] P. Majaranta and A. Bulling, “Eye Tracking and Eye-Based Human–Computer Interaction,” in *Advances in Physiological Computing*, S. H. Fairclough and K. Gilleade, Eds., in Human–Computer Interaction Series. , London: Springer London, 2014, pp. 39–65. doi: 10.1007/978-1-4471-6392-3\_3.
- [17] A. Duchowski, *Eye Tracking Methodology: Theory and Practice*. 2007. doi: 10.1007/978-1-84628-609-4.
- [18] J. R. Smith, M. J. Cronin, and I. Karacan, “A multichannel hybrid system for rapid eye movement detection (REM detection),” *Comput. Biomed. Res.*, vol. 4, no. 3, pp. 275–290, Jun. 1971, doi: 10.1016/0010-4809(71)90032-2.
- [19] K. K. R *et al.*, “Revolutionizing Autism Spectrum Disorder Diagnosis: Harnessing the Power of Machine Learning and MRI for Enhanced Neurodevelopmental Care,” in *2024 2nd International Conference on Artificial Intelligence and Machine Learning Applications Theme: Healthcare and Internet of Things (AIMLA)*, Namakkal, India: IEEE, Mar. 2024, pp. 1–6. doi: 10.1109/AIMLA59606.2024.10531585.
- [20] J. Song *et al.*, “Combining Radiomics and Machine Learning Approaches for Objective ASD Diagnosis: Verifying White Matter Associations with ASD,” 2024, *arXiv*. doi: 10.48550/ARXIV.2405.16248.
- [21] F. Thabtah, F. Kamalov, and K. Rajab, “A new computational intelligence approach to detect autistic features for autism screening,” *Int. J. Med. Inf.*, vol. 117, pp. 112–124, Sep. 2018, doi: 10.1016/j.ijmedinf.2018.06.009.
- [22] S. Raj and S. Masood, “Analysis and Detection of Autism Spectrum Disorder Using Machine Learning Techniques,” *Procedia Comput. Sci.*, vol. 167, pp. 994–1004, Jan. 2020, doi: 10.1016/j.procs.2020.03.399.
- [23] Y. Jayawardana, M. Jaime, and S. Jayarathna, “Analysis of Temporal Relationships between ASD and Brain Activity through EEG and Machine Learning,” in *2019 IEEE 20th International Conference on Information Reuse and Integration for Data Science (IRI)*, Jul. 2019, pp. 151–158. doi: 10.1109/IRI.2019.00035.
- [24] C. Bhaskarachary, A. J. Najafabadi, and B. Godde, “Machine Learning Supervised Classification Methodology for Autism Spectrum Disorder Based on Resting-State Electroencephalography (EEG) Signals,” in *2020 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, Sep. 2020, pp. 1–4. doi: 10.1109/SPMB50085.2020.9353626.
- [25] Q. Mohi ud Din and A. K. Jayanthy, “Detection of autism spectrum disorder by feature extraction of eeg signals and machine learning classifiers,” *Biomed. Eng. Appl. Basis Commun.*, vol. 35, no. 01, p. 2250046, Feb. 2023, doi: 10.4015/S1016237222500466.



- [26] M. Liao, H. Duan, and G. Wang, “Application of Machine Learning Techniques to Detect the Children with Autism Spectrum Disorder,” *J. Healthc. Eng.*, vol. 2022, no. 1, p. 9340027, Jan. 2022, doi: 10.1155/2022/9340027.
- [27] R. Carette, F. Cilia, G. Dequen, J. Bosche, J.-L. Guerin, and L. Vandromme, “Automatic Autism Spectrum Disorder Detection Thanks to Eye-Tracking and Neural Network-Based Approach,” in *Internet of Things (IoT) Technologies for HealthCare*, vol. 225, M. U. Ahmed, S. Begum, and J.-B. Fasquel, Eds., in Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 225. , Cham: Springer International Publishing, 2018, pp. 75–81. doi: 10.1007/978-3-319-76213-5\_11.
- [28] R. Carette, M. Elbattah, F. Cilia, G. Dequen, J.-L. Guérin, and J. Bosche, “Learning to Predict Autism Spectrum Disorder based on the Visual Patterns of Eye-tracking Scanpaths:,” in *Proceedings of the 12th International Joint Conference on Biomedical Engineering Systems and Technologies*, Prague, Czech Republic: SCITEPRESS - Science and Technology Publications, 2019, pp. 103–112. doi: 10.5220/0007402601030112.
- [29] M. Elbattah, R. Carette, G. Dequen, J.-L. Guerin, and F. Cilia, “Learning Clusters in Autism Spectrum Disorder: Image-Based Clustering of Eye-Tracking Scanpaths with Deep Autoencoder,” in *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Berlin, Germany: IEEE, Jul. 2019, pp. 1417–1420. doi: 10.1109/EMBC.2019.8856904.
- [30] Z. Zhao, H. Tang, X. Zhang, X. Qu, X. Hu, and J. Lu, “Classification of Children With Autism and Typical Development Using Eye-Tracking Data From Face-to-Face Conversations: Machine Learning Model Development and Performance Evaluation,” *J. Med. Internet Res.*, vol. 23, no. 8, p. e29328, Aug. 2021, doi: 10.2196/29328.
- [31] C. Chlebowski, J. A. Green, M. L. Barton, and D. Fein, “Using the Childhood Autism Rating Scale to Diagnose Autism Spectrum Disorders,” *J. Autism Dev. Disord.*, vol. 40, no. 7, pp. 787–799, Jul. 2010, doi: 10.1007/s10803-009-0926-x.
- [32] S. Zhang, D. Chen, Y. Tang, and L. Zhang, “Children ASD Evaluation Through Joint Analysis of EEG and Eye-Tracking Recordings With Graph Convolution Network,” *Front. Hum. Neurosci.*, vol. 15, May 2021, doi: 10.3389/fnhum.2021.651349.