



ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ
Πρόγραμμα Μεταπτυχιακών Σπουδών

Μηχανική Λογισμικού
για Διαδικτυακές & Φορητές Εφαρμογές

**Βελτιστοποίηση
προ-εκπαιδευμένων μοντέλων
Μηχανικής όρασης για κινητές
συσκευές και σύγκριση της
απόδοσης τους**

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Ιωάννης Μπλάνας AM: M013122006

Κωνσταντινιά Μαντζαφλάρα AM: M013122012

Επιβλέπων: Δρ. Γεώργιος Κακαρόντζας

ΛΑΡΙΣΑ Οκτώβριος 2024

«Εγώ ο **Ιωάννης Μπλάνας**, δηλώνω υπεύθυνα ότι η παρούσα Πτυχιακή Εργασία με τίτλο **Βελτιστοποίηση προ-εκπαιδευμένων μοντέλων Μηχανικής όρασης για κινητές συσκευές και σύγκριση της απόδοσης τους** είναι δική μου και βεβαιώνω ότι:

- Σε όσες περιπτώσεις έχω συμβουλευτεί δημοσιευμένη εργασία τρίτων, αυτό επισημαίνεται με σχετική αναφορά στα επίμαχα σημεία.
- Σε όσες περιπτώσεις μεταφέρω λόγια τρίτων, αυτό επισημαίνεται με σχετική αναφορά στα επίμαχα σημεία. Με εξαίρεση τέτοιες περιπτώσεις, το υπόλοιπο κείμενο της πτυχιακής αποτελεί δική μου δουλειά.
- Αναφέρω ρητά όλες τις πηγές βοήθειας που χρησιμοποίησα.
- Σε περιπτώσεις που τμήματα της παρούσας πτυχιακής έγιναν από κοινού με τρίτους, αναφέρω ρητά ποια είναι η δική μου συνεισφορά και ποια των τρίτων.
- Γνωρίζω πως η λογοκλοπή αποτελεί σοβαρότατο παράπτωμα και είμαι ενήμερος(-η) για την επέλευση των νόμιμων συνεπειών»

Ιωάννης Μπλάνας

«Εγώ η **Κωνσταντινιά Μαντζαφλάρα**, δηλώνω υπεύθυνα ότι η παρούσα Πτυχιακή Εργασία με τίτλο **Βελτιστοποίηση προ-εκπαιδευμένων μοντέλων Μηχανικής όρασης για κινητές συσκευές και σύγκριση της απόδοσης τους** είναι δική μου και βεβαιώνω ότι:

- Σε όσες περιπτώσεις έχω συμβουλευτεί δημοσιευμένη εργασία τρίτων, αυτό επισημαίνεται με σχετική αναφορά στα επίμαχα σημεία.
- Σε όσες περιπτώσεις μεταφέρω λόγια τρίτων, αυτό επισημαίνεται με σχετική αναφορά στα επίμαχα σημεία. Με εξαίρεση τέτοιες περιπτώσεις, το υπόλοιπο κείμενο της πτυχιακής αποτελεί δική μου δουλειά.
- Αναφέρω ρητά όλες τις πηγές βοήθειας που χρησιμοποίησα.
- Σε περιπτώσεις που τμήματα της παρούσας πτυχιακής έγιναν από κοινού με τρίτους, αναφέρω ρητά ποια είναι η δική μου συνεισφορά και ποια των τρίτων.
- Γνωρίζω πως η λογοκλοπή αποτελεί σοβαρότατο παράπτωμα και είμαι ενήμερος(-η) για την επέλευση των νόμιμων συνεπειών»

Κωνσταντινιά Μαντζαφλάρα

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή

Τόπος: Λάρισα.....

Ημερομηνία: 14/10/2024.....

ΕΠΙΤΡΟΠΗ ΑΞΙΟΛΟΓΗΣΗΣ

1. Κακαρόντζας Γεώργιος, Καθηγητής, Τμήμα Ψηφιακών Συστημάτων,
Πανεπιστήμιο Θεσσαλίας.....
2. Κόκκορας Φώτιος, Επίκουρος Καθηγητής, Τμήμα Ψηφιακών Συστημάτων,
Πανεπιστήμιο Θεσσαλίας.....
3. Κουτσονικόλα Βασιλική, Ε.ΔΙ.Π., Τμήμα Ψηφιακών Συστημάτων,
Πανεπιστήμιο Θεσσαλίας.....

Περίληψη

Τα κινητά τηλέφωνα στις μέρες μας έχουν βελτιωθεί αρκετά από πλευράς υλικολογισμικού, πολλά από αυτά, πλέον, διαθέτουν ξεχωριστό επεξεργαστή για την εκτέλεση διεργασιών τεχνητής νοημοσύνης (NPU).

Η αξιοποίηση αυτών των δυνατοτήτων σε βιομηχανικό περιβάλλον, φαίνεται πολλά υποσχόμενη.

Σκοπός αυτής της εργασίας είναι πρωτίστως η εύρεση προ-εκπαιδευμένων μοντέλων μηχανικής όρασης, καθώς και η βελτιστοποίησή τους, ώστε να χρειάζονται λιγότερους πόρους και να μπορούν να τρέχουν σε κινητές συσκευές. Ένας δεύτερος σκοπός, είναι η σύγκριση της απόδοσης τους σε σχέση με την αρχική απόδοση που είχαν πριν τη βελτιστοποίησή τους. Ο τελευταίος σκοπός είναι η αξιοποίηση του υλικολογισμικού των μοντέρνων κινητών τηλεφώνων και των τεχνικών βελτιστοποίησης μοντέλων μηχανικής όρασης (πχ Quantization) για να τρέχουν σε κινητά τηλέφωνα με αποδεκτές επιδόσεις.

Η υφιστάμενη κατάσταση (State of the Art) για τη βελτιστοποίηση προ-εκπαιδευμένων μοντέλων μηχανικής όρασης για να τρέχουν σε κινητές συσκευές είναι η χρήση τεχνικών μετασχηματισμού και αποσυμπίεσης. Αυτές οι τεχνικές επιτρέπουν τη μείωση του μεγέθους των μοντέλων χωρίς σημαντική απώλεια της απόδοσης. Αυτό μπορεί να ανοίξει νέες δυνατότητες για τη χρήση των εφαρμογών της μηχανικής όρασης σε κινητές συσκευές. Μερικές δυνατότητες μοντέλων μηχανικής όρασης είναι: η αναγνώριση προσώπου, η αναγνώριση αντικειμένων και η ανίχνευση κινήσεων.

Η βελτιστοποίηση προ-εκπαιδευμένων μοντέλων μηχανικής όρασης είναι μια πολλά υποσχόμενη περιοχή έρευνας που έχει τη δυνατότητα να προσφέρει νέες δυνατότητες και εφαρμογές.

Τα βασικά συμπεράσματα και σημεία καινοτομίας της διπλωματικής θα είναι η απόδειξη, ή η απόρριψη της βασικής ιδέας, σχετικά με το αν τα μοντέρνα κινητά τηλέφωνα έχουν φτάσει σε ικανοποιητικό επίπεδο ώστε να μπορούν να τρέχουν μοντέλα μηχανικής όρασης και να παράγουν αποτελέσματα με ικανοποιητική ακρίβεια σε πραγματικό χρόνο.

Λέξεις κλειδιά:

artificial intelligence, machine learning, computer vision, image processing, model optimization, android

Ευχαριστίες

Θα θέλαμε να ευχαριστήσουμε θερμά τον επιβλέποντα καθηγητή της διπλωματικής εργασίας κ. Δρ. Κακαρόντζα Γεώργιο για την πολύτιμη καθοδήγησή του, την εμπιστοσύνη και εκτίμηση που μας έδειξε.

Την εξεταστική επιτροπή για τον πολύτιμο χρόνο που μας αφιέρωσε για να μπορέσουμε να παρουσιάσουμε την διπλωματική μας εργασία.

Τις οικογένειές μας και τους φίλους μας που μας πρόσφεραν την απαραίτητη ηθική συμπαράσταση για την ολοκλήρωση αυτής.

Και τέλος να ευχαριστήσουμε και τον φίλο μας, Νείλο Ψαθά, που υπήρξε πολύτιμος αρωγός σε όλη αυτή τη διαδρομή.

Ιωάννης Μπλάνας & Κωνσταντινιά Μαντζαφλάρα

1/10/2024

Περιεχόμενα

ΠΕΡΙΛΗΨΗ	I
ΕΥΧΑΡΙΣΤΙΕΣ.....	III
ΠΕΡΙΕΧΟΜΕΝΑ.....	V
1 ΕΙΣΑΓΩΓΗ	1
1.1 ΕΞΕΛΙΞΗ ΤΗΣ ΤΕΧΝΟΛΟΓΙΑΣ ΣΤΑ ΚΙΝΗΤΑ ΤΗΛΕΦΩΝΑ	1
1.2 ΕΝΣΩΜΑΤΩΣΗ ΕΠΕΞΕΡΓΑΣΤΗ ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ ΣΕ ΣΥΣΚΕΥΕΣ	2
2 ΤΟ ΠΕΡΙΒΑΛΛΟΝ ΤΗΣ ΒΙΟΜΗΧΑΝΙΑΣ	5
2.1 ΑΞΙΟΠΟΙΗΣΗ ΤΕΧΝΗΤΗΣ ΝΟΗΜΟΣΥΝΗΣ ΣΕ ΒΙΟΜΗΧΑΝΙΚΟ ΕΠΙΠΕΔΟ	5
2.2 ΠΡΟΟΠΤΙΚΕΣ ΑΝΑΠΤΥΞΗΣ ΣΤΟΝ ΤΟΜΕΑ.....	6
3 ΣΤΟΧΟΙ ΤΗΣ ΕΡΓΑΣΙΑΣ	9
3.1 ΑΝΑΓΚΗ ΓΙΑ ΠΡΟ-ΕΚΠΑΙΔΕΥΜΕΝΑ ΜΟΝΤΕΛΑ ΜΗΧΑΝΙΚΗΣ ΌΡΑΣΗΣ	10
3.2 ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΠΡΟΚΕΙΜΕΝΟΥ ΝΑ ΕΞΥΠΗΡΕΤΟΥΝ ΤΙΣ ΑΠΑΙΤΗΣΕΙΣ ΚΙΝΗΤΩΝ ΣΥΣΚΕΥΩΝ ΚΑΘΩΣ ΚΑΙ ΤΙΣ ΑΝΑΓΚΕΣ ΣΕ ΒΙΟΜΗΧΑΝΙΚΟ ΕΠΙΠΕΔΟ	11
4 ΜΕΘΟΔΟΛΟΓΙΑ ΈΡΕΥΝΑΣ	13
4.1 ΕΠΙΛΟΓΗ ΠΡΟ-ΕΚΠΑΙΔΕΥΜΕΝΩΝ ΜΟΝΤΕΛΩΝ	13
4.2 ΜΕΤΡΗΣΕΙΣ ΚΑΙ ΑΞΙΟΛΟΓΗΣΗ.....	15
4.3 ΟΡΙΣΜΟΣ ΑΡΧΙΚΗΣ (BASELINE) ΑΠΟΔΟΣΗΣ	15
4.4 ΤΕΧΝΙΚΕΣ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ ΓΙΑ ΧΑΜΗΛΗ ΚΑΤΑΝΑΛΩΣΗ ΠΟΡΩΝ	16
5 ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ΚΙΝΗΤΑ ΤΗΛΕΦΩΝΑ ΜΕ ANDROID	19
5.1 ΤΕΧΝΙΚΕΣ ΥΛΟΠΟΙΗΣΗΣ ΕΦΑΡΜΟΓΗΣ ΓΙΑ ANDROID	19
5.2 ΣΗΜΑΣΙΑ ΤΗΣ ΣΥΜΒΑΤΟΤΗΤΑΣ ΜΕ ΤΟ ΛΕΙΤΟΥΡΓΙΚΟ ΣΥΣΤΗΜΑ.....	21
5.3 ΕΝΣΩΜΑΤΩΣΗ ΒΕΛΤΙΣΤΟΠΟΙΗΜΕΝΩΝ ΜΟΝΤΕΛΩΝ ΣΤΗΝ ΕΦΑΡΜΟΓΗ.....	23
6 ΣΥΜΠΕΡΑΣΜΑΤΑ ΓΙΑ ΤΗΝ ΕΦΑΡΜΟΓΗ ANDROID	27
6.1 ΑΠΟΤΕΛΕΣΜΑΤΑ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗΣ	27
6.2 ΑΠΟΔΟΣΗ ΣΕ ΚΙΝΗΤΕΣ ΣΥΣΚΕΥΕΣ	27

6.3	ΑΞΙΟΛΟΓΗΣΗ ΤΗΣ ΕΠΙΔΡΑΣΗΣ ΣΤΗΝ ΑΠΟΔΟΣΗ	29
6.4	ΣΥΓΚΡΙΣΗ ΜΕ ΤΗΝ ΑΡΧΙΚΗ ΑΠΟΔΟΣΗ	29
6.5	ΠΡΟΟΠΤΙΚΕΣ ΒΕΛΤΙΩΣΗΣ ΚΑΙ ΠΕΡΑΙΤΕΡΩ ΑΝΑΠΤΥΞΗ	30
7	ΣΥΝΟΛΙΚΑ ΣΥΜΠΕΡΑΣΜΑΤΑ.....	35
7.1	ΣΥΝΔΥΑΣΜΟΣ ΑΠΟΤΕΛΕΣΜΑΤΩΝ ΑΠΟ ΤΗ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΜΟΝΤΕΛΩΝ ΚΑΙ ΤΗΝ ΥΛΟΠΟΙΗΣΗ ΕΦΑΡΜΟΓΗΣ	35
7.2	ΤΕΛΙΚΗ ΑΞΙΟΛΟΓΗΣΗ ΤΗΣ ΣΥΝΟΛΙΚΗΣ ΕΡΓΑΣΙΑΣ.....	35
7.3	ΜΕΛΛΟΝΤΙΚΕΣ ΒΕΛΤΙΩΣΕΙΣ	36
8	ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ	37
8.1	ΑΝΑΔΕΙΚΝΥΟΝΤΑΣ ΤΑ ΒΑΣΙΚΑ ΘΕΜΑΤΑ ΜΕ ΛΕΞΕΙΣ-ΚΛΕΙΔΙΑ.....	37
	ΒΙΒΛΙΟΓΡΑΦΙΑ.....	39
	ΠΑΡΑΡΤΗΜΑ Α	43

1 Εισαγωγή

1.1 Εξέλιξη της τεχνολογίας στα κινητά τηλέφωνα

Η ιστορία των κινητών τηλεφώνων ξεκινάει με την παρουσίαση του DynaTAC 8000X από την Motorola το 1983. Ήταν μια ογκώδης, βαριά και ακριβή συσκευή, άπιαστο όνειρο για τους περισσότερους. Σταδιακά, η τεχνολογία εξελίχθηκε, μειώνοντας το μέγεθος και το κόστος, φέρνοντας τα κινητά τηλέφωνα στα χέρια όλο και περισσότερων ανθρώπων.

Σημαντικές στιγμές στην εξέλιξη των κινητών τηλεφώνων:

- 1992: Κυκλοφόρησε το IBM Simon το πρώτο smartphone με οθόνη αφής και δυνατότητα αποστολής/λήψης email.
- 1999: Κυκλοφόρησε το Nokia 3210 ένα από τα πιο δημοφιλή κινητά όλων των εποχών, με στιβαρό design και παιχνίδια, όπως το Snake.
- 2000: Κυκλοφόρησε το BlackBerry φέρνοντας επανάσταση στον τρόπο επικοινωνίας με email και μηνύματα.
- 2002: Κυκλοφόρησε το Nokia 7650 το πρώτο κινητό με ενσωματωμένη κάμερα.
- 2007: Κυκλοφόρησε το iPhone σηματοδοτώντας μια νέα εποχή με την οθόνη αφής υψηλής ανάλυσης, το App Store με πλήθος εφαρμογών.
- 2010: Κυκλοφόρησε το Android λειτουργικό σύστημα ανοιχτού κώδικα με μεγάλες δυνατότητες παραμετροποίησης (customization) στα smartphones.

Τα smartphones πλέον έχουν εξελιχθεί, ενσωματώνουν τεχνολογίες όπως πολυπύρηνους επεξεργαστές, που τα φέρνουν πολύ κοντά στις επιδόσεις με τους H/Y, υψηλής ανάλυσης οθόνες, πολλαπλές κάμερες με δυνατότητες εστίασης (focus), δημιουργία τρισδιάστατων εικόνων, αισθητήρες γεωεντοπισμού (GPS), βηματομετρητή, αναγνώστη δακτυλικών αποτυπωμάτων ή προσώπου ή ίριδας, μετρητή παλμών και οξυγόνου, NFC, αντοχή στο νερό και τη σκόνη και δυνατότητα σύνδεσης στο internet μέσω 4G/5G/WiFi.

Πλέον, τα smartphones δεν είναι απλά συσκευές επικοινωνίας, αλλά έχουν αντικαταστήσει πλήθος άλλων συσκευών όπως φωτογραφικές μηχανές βιντεοκάμερες για φωτογραφίες και βίντεο υψηλής ανάλυσης, ραδιόφωνα και MP3 players για ενημέρωση και ψυχαγωγία, GPS και χάρτες, φορητούς υπολογιστές για ψυχαγωγία, δουλειά, ενημέρωση και παιχνίδι, ιατρικά εργαλεία (π.χ. μέτρηση καρδιακών παλμών), πορτοφόλια (με δυνατότητα ανέπαφων πληρωμών, και εγγράφων ταυτοποίησης)

Η εξέλιξη της τεχνολογίας συνεχίζεται με ραγδαίους ρυθμούς, φέρνοντας νέες δυνατότητες και εφαρμογές στο μέλλον.

1.2 Ενσωμάτωση επεξεργαστή τεχνητής νοημοσύνης σε συσκευές

Η ενσωμάτωση επεξεργαστή τεχνητής νοημοσύνης (NPU) σε κινητά τηλέφωνα αποτελεί μια σημαντική εξέλιξη, φέρνοντας πολλές νέες δυνατότητες. Πλέον, τα smartphones είναι εργαλεία για εφαρμογές όπως:

Αναγνώριση εικόνας και αντικειμένων χάρη στις κάμερες υψηλής ανάλυσης και τα AI chips, τα κινητά μπορούν να αναγνωρίζουν πρόσωπα, αντικείμενα, κείμενο σε διάφορες γλώσσες και τοποθεσίες σε πραγματικό χρόνο.

Επεξεργασία φυσικής ομιλίας με τη χρήση AI πλέον μπορούν να κατανοούν να μεταφράζουν και να αναπαράγουν ανθρώπινη ομιλία, προσφέροντας δυνατότητες όπως φωνητικές εντολές και μετάφραση σε πραγματικό.

Μηχανική μάθηση τα smartphones μπορούν να μαθαίνουν από τη χρήση και να προσαρμόζονται στις ανάγκες του χρήστη, προσφέροντας μια πιο εξατομικευμένη εμπειρία, εξοικονόμηση μπαταρίας και λιγότερη φθορά της, συμβάλλοντας σε μια καλύτερη εμπειρία και στην πράσινη συνείδηση.

Σε σύγκριση με παλαιότερα τα σύγχρονα smartphones του 2024 διαθέτουν πολυπύρηνους επεξεργαστές με ως και 10 πυρήνες, ισχυρούς επεξεργαστές γραφικών (GPU) που μαζί με τις Neural Processing Units (NPU) μπορούν να επεξεργαστούν εικόνες υψηλής ανάλυσης, μεγάλη μνήμη RAM πάνω από 16GB, και αποθηκευτικό χώρο έως και 1TB.

Ως αποτέλεσμα, τα smartphones πλέον μπορούν να εκτελέσουν πολύπλοκες ΑΙ εφαρμογές, κάποιες από αυτές είναι η αναγνώριση προσώπου, η βελτιστοποίηση και επεξεργασία φωτογραφιών και βίντεο, Augmented reality εφαρμογές και παιχνίδια, μετάφραση σε πραγματικό χρόνο.

2 Το Περιβάλλον της Βιομηχανίας

2.1 Αξιοποίηση τεχνητής νοημοσύνης σε βιομηχανικό επίπεδο

Η Τεχνητή Νοημοσύνη (TN) γίνεται πλέον κομμάτι του βιομηχανικού τομέα, φέρνοντας επαναστατικές αλλαγές στις παραγωγικές διαδικασίες και προσφέροντας πλήθος νέων ευκαιριών.

Πλεονεκτήματα της TN στη Βιομηχανία:

- **Αυξημένη Παραγωγικότητα:** Η αυτοματοποίηση επαναλαμβανόμενων εργασιών και η βελτιστοποίηση των διαδικασιών οδηγούν σε άνοδο της παραγωγικότητας και μείωση του κόστους παραγωγής.
- **Βελτιωμένη Ποιότητα:** Η TN δύναται να εντοπίσει ελαττώματα σε πρώιμο στάδιο, διασφαλίζοντας υψηλότερο επίπεδο ποιότητας στα παραγόμενα προϊόντα.
- **Προληπτική Συντήρηση:** Η ανάλυση δεδομένων από αισθητήρες μπορεί να προβλέψει έγκαιρα πιθανές βλάβες, ώστε να αποφεύγονται οι αστοχίες και οι περίοδοι αδράνειας λόγω βλάβης.
- **Βελτιωμένη Ασφάλεια:** Η TN δύναται να εντοπίσει επικίνδυνες συνθήκες και να λάβει προληπτικά μέτρα για την προστασία των εργαζομένων.

Παραδείγματα:

- Η Siemens αξιοποιεί την TN για την πρόβλεψη της ζήτησης ανταλλακτικών, μειώνοντας το κόστος αποθεμάτων κατά 30%.
- Η GE Aviation χρησιμοποιεί TN για την ανάλυση δεδομένων πτήσεων, προβλέποντας πιθανές βλάβες σε κινητήρες αεροσκαφών.
- Η Toyota αξιοποιεί robots με TN για την αυτοματοποίηση της παραγωγής, αυξάνοντας την παραγωγικότητα κατά 20%.

2.2 Προοπτικές ανάπτυξης στον τομέα

Ο τομέας της Τεχνητής Νοημοσύνης (TN) παρουσιάζει ραγδαία ανάπτυξη, φέρνοντας νέες δυνατότητες και προοπτικές στον βιομηχανικό κλάδο. Ας δούμε μερικές από τις πιο σημαντικές:

Δημιουργία νέων προϊόντων και υπηρεσιών:

- Σχεδιασμός προϊόντων με βάση τεχνητή νοημοσύνη: Η Nike χρησιμοποιεί TN για τον σχεδιασμό παπουτσιών με βελτιωμένη άνεση και απόδοση.
- Προσαρμοσμένα προϊόντα: Η Coca-Cola χρησιμοποιεί TN για την παραγωγή personalized vending machines που προτείνουν προϊόντα βάσει των προτιμήσεων του καταναλωτή.
- Νέες ψηφιακές υπηρεσίες: Η Rolls-Royce προσφέρει predictive maintenance για κινητήρες αεροσκαφών, μειώνοντας το χρόνο ακινητοποίησης κατά 25%.

Ενίσχυση της ασφάλειας και της συμμόρφωσης:

- Πρόβλεψη και πρόληψη ατυχημάτων: Η Honeywell χρησιμοποιεί TN για την πρόβλεψη και πρόληψη ατυχημάτων σε εργοστάσια, μειώνοντας τον αριθμό ατυχημάτων κατά 50%.
- Βελτιστοποίηση της διαχείρισης περιβάλλοντος: Η Dow Chemical χρησιμοποιεί TN για την μείωση του περιβαλλοντικού αποτυπώματος, μειώνοντας τις εκπομπές CO₂ κατά 10%.
- Κυβερνοασφάλεια: Η IBM προσφέρει λύσεις κυβερνοασφάλειας για βιομηχανικά συστήματα, προστατεύοντας από κυβερνοεπιθέσεις.

Επιπλέον:

- Η Ευρωπαϊκή Ένωση έχει επενδύσει σημαντικά στην TN, με στόχο να ηγηθεί της 4ης Βιομηχανικής Επανάστασης.
- Η Τεχνητή Νοημοσύνη αναμένεται να δημιουργήσει έως και 16 εκατομμύρια νέες θέσεις εργασίας στην Ευρώπη έως το 2030.

Η TN έχει τη δυνατότητα να μεταμορφώσει τον βιομηχανικό κλάδο, φέρνοντας άνευ προηγουμένου οφέλη σε παραγωγικότητα, ποιότητα, ασφάλεια και καινοτομία. Οι επιχειρήσεις που υιοθετούν την TN έγκαιρα θα είναι σε πλεονεκτική θέση στο ανταγωνιστικό περιβάλλον του μέλλοντος.

3 Στόχοι της Εργασίας

Η εργασία εστιάζει στην έρευνα και υλοποίηση μίας demo εφαρμογής TN με στόχο την επίλυση προβλημάτων και την ενίσχυση της παραγωγικότητας στον βιομηχανικό κλάδο. Η υλοποίηση, αξιολόγηση και τεκμηρίωση της εφαρμογής θα προσφέρει πολύτιμα συμπεράσματα για τις δυνατότητες και τα οφέλη της TN, συμβάλλοντας στην υιοθέτησή της από την βιομηχανία.

Χαρτογράφηση απαιτήσεων:

Καθορισμός των λειτουργικών και μη λειτουργικών απαιτήσεων της εφαρμογής demo.

- **Λειτουργίες:** Τι θα κάνει η εφαρμογής demo;
- **Μη λειτουργικές απαιτήσεις:** Αξιοπιστία, ασφάλεια, επεκτασιμότητα.

Στόχοι Υλοποίησης Εφαρμογής Demo:

- Ανάπτυξη λειτουργικού πρωτοτύπου: Δημιουργία demo εφαρμογής που αξιοποιεί TN για την επίλυση μίας ή περισσότερων προκλήσεων στον επιλεγμένο βιομηχανικό κλάδο.
- Επιλογή κατάλληλων μεθοδολογιών και εργαλείων TN.
- Εστίαση στην ευχρηστία και την οπτικοποίηση των αποτελεσμάτων.
- Αξιολόγηση demo: Δοκιμή και αξιολόγηση της demo εφαρμογής σε πραγματικό ή προσομοιωμένο περιβάλλον.
- Συλλογή δεδομένων και μετρήσεων για την απόδειξη της αποτελεσματικότητας της TN.
- Εντοπισμός βελτιώσεων και μελλοντικών εξελίξεων.
- Τεκμηρίωση: Δημιουργία τεκμηρίωσης που περιγράφει τα χαρακτηριστικά της demo εφαρμογής και την μεθοδολογία υλοποίησης.
- Τα αποτελέσματα και τα συμπεράσματα της αξιολόγησης.

3.1 Ανάγκη για προ-εκπαιδευμένα μοντέλα Μηχανικής Όρασης

Η ανάπτυξη συστημάτων Μηχανικής Όρασης (ΜΟ) αποτελεί μια σύνθετη και χρονοβόρα διαδικασία, ιδιαίτερα όταν υλοποιείται από την αρχή. Η δημιουργία μοντέλων ΜΟ από το μηδέν φέρει πλήθος δυσκολιών:

- **Συλλογή δεδομένων:**

Απαιτείται η δημιουργία ή/και η συγκέντρωση μίας μεγάλης και ποικίλης βάσης δεδομένων εικόνων, με κατάλληλη σήμανση (annotation). Η διαδικασία συλλογής και σήμανσης δεδομένων είναι χρονοβόρα, κοστοβόρα και εμπεριέχει τον κίνδυνο λάθος σήμανσης.

- **Εκπαίδευση μοντέλων:**

Η εκπαίδευση μοντέλων ΜΟ από το μηδέν απαιτεί υψηλή υπολογιστική ισχύ και χρόνο. Η υπερπαραμετροποίηση (overfitting) και η υποπαραμετροποίηση (underfitting) αποτελούν επίσης συχνές προκλήσεις.

- **Εξειδίκευση σε συγκεκριμένες εργασίες:**

Η προσαρμογή και βελτιστοποίηση μοντέλων για κάθε νέα εργασία είναι χρονοβόρα και απαιτεί εξειδικευμένες γνώσεις.

Πλεονεκτήματα προ-εκπαιδευμένων μοντέλων:

- **Ταχύτητα:** Δραστική μείωση του χρόνου ανάπτυξης συστημάτων ΜΟ.
- **Ακρίβεια:** Βελτιωμένη ακρίβεια και απόδοση, χάρη στην εκπαίδευση σε μεγάλα σύνολα δεδομένων.
- **Ευκολία:** Απλοποίηση της υλοποίησης, μειώνοντας τις απαιτήσεις σε υπολογιστική ισχύ και τεχνογνωσία.
- **Ευελιξία:** Δυνατότητα προσαρμογής και εξειδίκευσης σε πλήθος εργασιών.

Η χρήση προ-εκπαιδευμένων μοντέλων ΜΟ αποτελεί απαραίτητη συνθήκη για την ταχεία και αποτελεσματική ανάπτυξη συστημάτων ΜΟ. Η διπλωματική εργασία εστιάζει στην αξιοποίηση και εξειδίκευση υφιστάμενων μοντέλων, προς όφελος της υλοποίησης και αξιολόγησης της demo εφαρμογής.

3.2 Βελτιστοποίηση προκειμένου να εξυπηρετούν τις απαιτήσεις κινητών συσκευών καθώς και τις ανάγκες σε βιομηχανικό επίπεδο

Η βελτιστοποίηση των μοντέλων αποτελεί καίριο ζήτημα για την εξυπηρέτηση των απαιτήσεων των κινητών συσκευών.

Κινητές Συσκευές:

- Περιορισμένοι πόροι: Τα smartphones διαθέτουν περιορισμένη υπολογιστική ισχύ και μνήμη, σε σχέση με τους υπολογιστές.
- Ενεργειακή αυτονομία: Η υψηλή κατανάλωση ενέργειας επηρεάζει αρνητικά την αυτονομία της συσκευής.
- Χρόνος απόκρισης: Απαιτείται άμεση (εντός λίγων χιλιοστών του δευτερολέπτου) ανταπόκριση για ομαλή εμπειρία χρήστη.

Βιομηχανικές Εφαρμογές:

- Απαιτήσεις σε πραγματικό χρόνο: Η λήψη αποφάσεων βασισμένη σε real-time δεδομένα είναι απαραίτητη για την σωστή λειτουργία βιομηχανικών διαδικασιών.
- Αξιοπιστία: Η αστοχία ή η καθυστέρηση του συστήματος μπορεί να έχει σοβαρές οικονομικές (και όχι μόνο) συνέπειες.
- Ασφάλεια: Η προστασία των δεδομένων και η αποφυγή κυβερνοεπιθέσεων είναι ζωτικής σημασίας.

Στόχοι βελτιστοποίησης:

- Ελαχιστοποίηση υπολογιστικών πόρων: Αξιοποίηση τεχνικών όπως pruning, quantization και efficient neural architectures για μείωση του computational footprint.
- Μείωση κατανάλωσης ενέργειας: Εφαρμογή τεχνικών low-power design και optimization algorithms για ενεργειακή αποδοτικότητα.
- Επιτάχυνση χρόνου απόκρισης: Βελτιστοποίηση αλγορίθμων και υλοποίηση σε κατάλληλο hardware (π.χ. GPUs) για real-time performance.
- Διασφάλιση αξιοπιστίας: Εφαρμογή δοκιμών και μετρήσεων για robustness και error handling.

Η βελτιστοποίηση της demo εφαρμογής είναι απαραίτητη για την υλοποίηση μίας λύσης που ανταποκρίνεται στις απαιτήσεις κινητών συσκευών και βιομηχανικών περιβαλλόντων. Η εστίαση στους προαναφερθέντες στόχους θα διασφαλίσει την αποτελεσματικότητα, αξιοπιστία και ασφάλεια της demo, φέρνοντας την ΤΝ ένα βήμα πιο κοντά στην υιοθέτησή της από τον βιομηχανικό κλάδο.

4 Μεθοδολογία Έρευνας

4.1 Επιλογή προ-εκπαιδευμένων μοντέλων

Η επιλογή των κατάλληλων προ-εκπαιδευμένων μοντέλων για τη βελτιστοποίηση και χρήση σε κινητές συσκευές ήταν μία από τις μεγαλύτερες προκλήσεις της έρευνας. Η απαίτηση για μοντέλα που μπορούν να υποστηρίξουν μεθόδους βελτιστοποίησης, όπως η ποσοτικοποίηση (Quantization) και το κλάδεμα μοντέλων (Model Pruning), και ταυτόχρονα να διατηρούν υψηλά επίπεδα ακρίβειας, περιόρισε τις επιλογές. Στην παρούσα εργασία, μετά από εκτενή έρευνα και δοκιμές, το μοντέλο που επιλέχθηκε για χρήση ήταν το **YOLOv7**, ένα από τα πιο σύγχρονα και αξιόπιστα μοντέλα για εφαρμογές ανίχνευσης αντικειμένων.

Εναλλακτικές Επιλογές και Λόγοι Απόρριψης

1. **MobileNetV2/ MobileNetV3:** Αυτά τα μοντέλα έχουν σχεδιαστεί ειδικά για χρήση σε κινητές συσκευές και συχνά είναι οι πρώτες επιλογές για βελτιστοποίηση. Ωστόσο, παρά το μικρό τους μέγεθος και την ικανότητα βελτιστοποίησης, παρατηρήθηκε πως σε εφαρμογές που απαιτούν μεγαλύτερη ακρίβεια, όπως η ανίχνευση αντικειμένων σε πραγματικό χρόνο, δεν διατηρούν υψηλά ποσοστά επιτυχίας μετά την εφαρμογή μεθόδων όπως το Model Pruning ή/και το Quantization. Η πτώση στην ακρίβεια μετά τη βελτιστοποίηση ήταν σημαντική, ιδιαίτερα σε σύγκριση με πιο εξειδικευμένα μοντέλα ανίχνευσης.
2. **EfficientDet: To EfficientDet**, μια δημοφιλής επιλογή για ανίχνευση αντικειμένων, παρουσιάζει σημαντικές βελτιώσεις σε σχέση με προηγούμενα μοντέλα όσον αφορά την ταχύτητα και την απόδοση. Ωστόσο, το μέγεθος του μοντέλου και η υπολογιστική του πολυπλοκότητα καθιστούν τη βελτιστοποίησή του για κινητές συσκευές απαιτητική, με τη μείωση του μεγέθους μέσω των μεθόδων ποσοτικοποίησης να οδηγεί σε μη αποδεκτή πτώση της ακρίβειας. Η μεγάλη απαίτηση σε υπολογιστική ισχύ επίσης περιορίζει την αποδοτικότητά του σε συσκευές με περιορισμένους πόρους, όπως τα «φθηνότερα» κινητά τηλέφωνα.

3. **ResNet50:** Παρά την ισχυρή απόδοσή του στις περισσότερες εφαρμογές, το ResNet50 είναι βαρύ σε πόρους και απαιτεί μεγάλες υπολογιστικές δυνατότητες. Αν και το pruning και το quantization βελτιώνουν την απόδοσή του σε κινητές συσκευές, παρατηρήθηκε μια δυσανάλογη μείωση στην ακρίβεια ανίχνευσης αντικειμένων, γεγονός που το καθιστά λιγότερο ιδανικό για περιβάλλοντα με πραγματικούς χρόνους απόκρισης.

Επιλογή του YOLOv7

Το YOLOv7 επιλέχθηκε ως το πιο κατάλληλο μοντέλο για την παρούσα εργασία. Το YOLO (You Only Look Once) είναι μια από τις κορυφαίες αρχιτεκτονικές για ανίχνευση αντικειμένων σε πραγματικό χρόνο, και η τελευταία έκδοση, YOLOv7, φέρει σημαντικές βελτιώσεις τόσο σε απόδοση όσο και σε ταχύτητα. Τα πλεονεκτήματα του YOLOv7 που το καθιστούν ιδανικό για τη συγκεκριμένη εργασία περιλαμβάνουν:

- **Υψηλή Ταχύτητα:** Το YOLOv7 είναι εξαιρετικά γρήγορο, καθιστώντας το κατάλληλο για εφαρμογές σε κινητές συσκευές όπου η ανίχνευση αντικειμένων πρέπει να πραγματοποιείται σε πραγματικό χρόνο.
- **Υψηλή Ακρίβεια:** Παρά τις βελτιστοποιήσεις, το YOLOv7 καταφέρνει να διατηρεί υψηλά επίπεδα ακρίβειας, ακόμα και όταν εφαρμόζονται τεχνικές quantization και pruning.
- **Αποτελεσματικότητα Βελτιστοποίησης:** Το YOLOv7 έχει σχεδιαστεί με γνώμονα τη δυνατότητα βελτιστοποίησης για διαφορετικές συσκευές και πλατφόρμες, γεγονός που διευκολύνει την προσαρμογή του σε κινητά τηλέφωνα.
- **Υποστήριξη Βελτιστοποιημένων Υλοποιήσεων:** Υπάρχουν ήδη βιβλιοθήκες και εργαλεία που επιτρέπουν την εύκολη εφαρμογή βελτιστοποιήσεων όπως quantization και pruning στο YOLOv7, διευκολύνοντας την ενσωμάτωσή του σε κινητές συσκευές με χαμηλή υπολογιστική ισχύ.

Το YOLOv7, με την ικανότητα του να ανιχνεύει αντικείμενα γρήγορα και με ακρίβεια, ακόμη και υπό συνθήκες βελτιστοποίησης, επιλέχθηκε ως το ιδανικό μοντέλο για την επίτευξη των στόχων της διπλωματικής αυτής εργασίας.

4.2 Μετρήσεις και αξιολόγηση

Η αξιολόγηση της demo εφαρμογής αποτελεί καίριο κομμάτι της εργασίας, με στόχο την τεκμηρίωση της αποτελεσματικότητας και της βελτιστοποίησης. Προς τούτο, υιοθετούνται οι ακόλουθες μετρήσεις:

Inference Time:

Ο χρόνος που απαιτείται για την εξαγωγή συμπερασμάτων από το μοντέλο MO, μετρώντας:

- **Latency:** Χρόνος από την τροφοδοσία δεδομένων έως την λήψη αποτελέσματος.
- **Throughput:** Αριθμός επεξεργασμένων εικόνων ανά δευτερόλεπτο.

Επιπλέον μετρήσεις:

- **Precision:** Ποσοστό $TP / (TP + FP)$.
- **Recall (Sensitivity):** Ποσοστό $TP / (TP + FN)$.

Η αξιολόγηση της demo εφαρμογής λαμβάνει υπόψη διάφορες μετρήσεις, καλύπτοντας τόσο την ακρίβεια της MO, όσο και την βελτιστοποίηση πόρων. Η λεπτομερής ανάλυση των μετρήσεων θα οδηγήσει σε τεκμηριωμένα συμπεράσματα για την αποτελεσματικότητα της υλοποίησης, συμβάλλοντας στην μελλοντική βελτίωση και εξέλιξή τη.

4.3 Ορισμός αρχικής (baseline) απόδοσης

Σκοπός αυτού του υποκεφαλαίου είναι να καταγράψουμε τα αρχικά αποτελέσματα, τα οποία θα χρησιμοποιηθούν ως μέτρο σύγκρισης με τα αποτελέσματα μετά τις βελτιστοποιήσεις.

Για την αξιολόγηση της αρχικής απόδοσης του YOLOv7, πραγματοποιήθηκαν δοκιμές σε υπολογιστές με τις παρακάτω προδιαγραφές:

- **Επεξεργαστής (CPU):** Intel Core i7/i9
- **Κάρτα γραφικών (GPU):** NVIDIA RTX 3070 / 3080 / 4070 / 4090
- **Μνήμη (RAM):** 32 GB
- **Περιβάλλον υλοποίησης:** PyTorch / TensorFlow με CUDA ενεργοποιημένο

Οι μετρήσεις πραγματοποιήθηκαν σε datasets ανίχνευσης αντικειμένων, όπως το COCO dataset, και τα αποτελέσματα που καταγράφηκαν περιλαμβάνουν τα ακόλουθα:

- **FPS (Frames per second):** Το μοντέλο YOLOv7 μπορεί να εκτελείται σε περίπου 45-70 FPS σε κάρτες γραφικών όπως οι NVIDIA RTX 3070 ή 4070, όταν χρησιμοποιείται το πλήρες μοντέλο χωρίς καμία βελτιστοποίηση. Η ακριβής απόδοση εξαρτάται από το μέγεθος του batch και την ανάλυση των εικόνων.
- **Precision (Ακρίβεια):** Η μέση ακρίβεια (precision) του YOLOv7 στην ανίχνευση αντικειμένων, πριν την εφαρμογή οποιασδήποτε βελτιστοποίησης, κυμαίνεται περίπου στο 55-60% στο COCO dataset. Το ποσοστό αυτό αναφέρεται στη δυνατότητα του μοντέλου να εντοπίζει αντικείμενα με υψηλή ακρίβεια, ελαχιστοποιώντας τα false positives.
- **Recall (Ανάκληση):** Το recall, που μετρά την ικανότητα του μοντέλου να εντοπίζει όλα τα πραγματικά αντικείμενα (ελαχιστοποίηση των false negatives), καταγράφηκε περίπου στο 65-70% στο ίδιο dataset. Η απόδοση αυτή θεωρείται πολύ καλή, δεδομένου ότι το YOLOv7 είναι ένα μοντέλο ανίχνευσης αντικειμένων σε πραγματικό χρόνο.

Αυτά τα νούμερα αποτελούν την αρχική απόδοση (baseline) του μοντέλου και θα χρησιμοποιηθούν για να συγκριθούν με τα αποτελέσματα που θα προκύψουν μετά την εφαρμογή των τεχνικών βελτιστοποίησης (Quantization, Model Pruning) στις κινητές συσκευές.

4.4 Τεχνικές βελτιστοποίησης για χαμηλή κατανάλωση πόρων

Η βελτιστοποίηση της demo εφαρμογής για χαμηλή κατανάλωση πόρων αποτελεί καίριο ζήτημα για την υλοποίηση μίας λύσης συμβατής με κινητές συσκευές και βιομηχανικές εφαρμογές. Προς τούτο, υιοθετούνται οι ακόλουθες τεχνικές:

Quantization:

- **Post-training quantization:** Για παράδειγμα μετατροπή των δεδομένων του μοντέλου από float32 σε uint8, μειώνοντας δραστικά το μέγεθος και την κατανάλωση ενέργειας.

- Quantization-aware training: Εκπαίδευση του μοντέλου με quantized weights, βελτιώνοντας την ακρίβεια μετά την quantization.

Model Pruning:

Αφαίρεση περιττών συνδέσεων και νευρώνων: Μείωση του αριθμού παραμέτρων, με ελάχιστη επίδραση στην ακρίβεια.

Structured pruning:

Αφαίρεση ολόκληρων layers ή blocks, για δραστική μείωση του model size.

Efficient Neural Architectures:

- MobileNet: Ελαφριά αρχιτεκτονική με depthwise separable convolutions.
- EfficientNet: Συνδυασμός MobileNet και compound scaling για βέλτιστη ακρίβεια και απόδοση.

Hardware Acceleration:

Εκμετάλλευση της υπολογιστικής ισχύος GPUs και NPUs: Επιτάχυνση της inference process.

5 Υλοποίηση εφαρμογής για κινητά τηλέφωνα με Android

5.1 Τεχνικές υλοποίησης εφαρμογής για Android

Η ανάπτυξη μιας εφαρμογής για Android που απαιτεί υψηλή απόδοση και υποστήριξη για προχωρημένες λειτουργίες, όπως η μηχανική όραση και η ανίχνευση αντικειμένων σε πραγματικό χρόνο, αποτελεί μία πολύπλοκη και απαιτητική διαδικασία. Στην υλοποίηση της εφαρμογής αυτής, χρησιμοποιήθηκαν αρκετές τεχνικές και εργαλεία που απαιτούν υψηλό επίπεδο εξειδίκευσης, με στόχο να επιτευχθεί η απαιτούμενη απόδοση σε κινητές συσκευές.

Χρήση JNI για σύνδεση Java και C/C++

Ένα από τα πιο κρίσιμα στοιχεία της υλοποίησης ήταν η χρήση του **JNI (Java Native Interface)**, το οποίο επιτρέπει τη σύνδεση του κώδικα σε Java με τον κώδικα γραμμένο σε γλώσσες όπως C ή C++. Η χρήση του JNI είναι απαραίτητη σε περιπτώσεις όπου η υπολογιστική δύναμη των κινητών τηλεφώνων πρέπει να αξιοποιηθεί στο έπακρο, όπως στην παρούσα εργασία, όπου γίνεται χρήση των βιβλιοθηκών **Vulkan** και **OpenCV** για την εκτέλεση των πιο απαιτητικών λειτουργιών.

Η χρήση του JNI επιτρέπει τη διασύνδεση των δύο περιβαλλόντων (Java και C++) με ασφάλεια, ώστε να γίνεται αποδοτική διαχείριση των πόρων του συστήματος. Ωστόσο, αυτή η προσέγγιση φέρνει ορισμένες προκλήσεις:

- **Συντονισμός μεταξύ Java και C/C++:** Η συγχώνευση των δύο γλωσσών είναι μια περίπλοκη διαδικασία, καθώς οι δύο αυτές γλώσσες διαχειρίζονται τη μνήμη και τα δεδομένα με διαφορετικούς τρόπους. Η σωστή διαχείριση της μνήμης και η αποφυγή διαρροών είναι κρίσιμες για την επιτυχία της υλοποίησης, ειδικά σε κινητές συσκευές με περιορισμένους πόρους.
- **Διαχείριση πολυνηματικότητας:** Δεδομένου ότι πολλές από τις λειτουργίες που απαιτούνται για την ανίχνευση αντικειμένων εκτελούνται σε διαφορετικά νήματα (threads), η συγχρονισμένη εκτέλεση αυτών των διεργασιών είναι μία

πρόκληση. Οι δυσκολίες προκύπτουν κυρίως από το γεγονός ότι τα νήματα πρέπει να συντονίζονται σωστά για να αποφεύγονται καθυστερήσεις ή κολλήματα (deadlocks).

Ενσωμάτωση Vulkan για γραφική επεξεργασία

Το **Vulkan** είναι ένα API για γραφική επεξεργασία που προσφέρει υψηλές επιδόσεις και χαμηλού επιπέδου πρόσβαση στο υλικό των συσκευών. Η χρήση του Vulkan αντί του παραδοσιακού OpenGL επιτρέπει την καλύτερη διαχείριση των πόρων και την απόδοση πραγματικού χρόνου, κάτι που είναι απαραίτητο για εφαρμογές μηχανικής όρασης. Μέσω του JNI, το Vulkan χρησιμοποιήθηκε για την επιτάχυνση γραφικών διεργασιών, όπως η επεξεργασία εικόνας σε πραγματικό χρόνο και η απεικόνιση ανιχνευμένων αντικειμένων.

Η ενσωμάτωση του Vulkan φέρνει σημαντικές προκλήσεις:

- **Πολυπλοκότητα του API:** Το Vulkan προσφέρει υψηλή απόδοση, αλλά με το κόστος της αυξημένης πολυπλοκότητας. Η διαχείριση μνήμης, οι κατευθυντήριες γραμμές για τη βέλτιστη χρήση των πόρων του υλικού και η συγγραφή αποδοτικού κώδικα απαιτούν βαθιά γνώση του API και προσεκτική υλοποίηση.
- **Ανάπτυξη με CMake και JNI:** Η σύνδεση του Vulkan με τον κώδικα της Java έγινε μέσω της χρήσης του CMake, που επέτρεψε τη δημιουργία και τη διαχείριση της επικοινωνίας μεταξύ των περιβαλλόντων ανάπτυξης. Η χρήση του CMake σε συνδυασμό με το JNI δημιούργησε ένα ολοκληρωμένο σύστημα για την ενσωμάτωση των υποσυστημάτων σε Java και C++.

Χρήση OpenCV για επεξεργασία εικόνας

Το **OpenCV** είναι μία από τις πιο διαδεδομένες βιβλιοθήκες για επεξεργασία εικόνας και μηχανική όραση. Στην παρούσα εφαρμογή, το OpenCV χρησιμοποιήθηκε για την επεξεργασία εικόνων που λαμβάνονται από την κάμερα του κινητού τηλεφώνου και την προετοιμασία τους για την ανίχνευση αντικειμένων μέσω του YOLOv7.

Η ενσωμάτωση του OpenCV έγινε επίσης μέσω του JNI, κάτι που προσέφερε μεγαλύτερη ευελιξία στη χρήση των εργαλείων και των αλγορίθμων της βιβλιοθήκης. Παράλληλα, οι προκλήσεις περιλάμβαναν:

- **Επιδόσεις σε πραγματικό χρόνο:** Η εφαρμογή μηχανικής όρασης σε κινητές συσκευές απαιτεί γρήγορη και αποδοτική επεξεργασία εικόνας. Η σύνδεση του

OpenCV με τον κώδικα της C++ και η χρήση του JNI επέτρεψε την αξιοποίηση των δυνατοτήτων της συσκευής στο έπακρο.

- **Συμβατότητα και σταθερότητα:** Η χρήση εξωτερικών βιβλιοθηκών όπως το OpenCV απαιτεί προσεκτική διαχείριση της συμβατότητας μεταξύ διαφορετικών εκδόσεων του Android και του υλικού της συσκευής.

Προκλήσεις και Δυσκολίες Υλοποίησης

Η υλοποίηση αυτής της εφαρμογής με τη χρήση του JNI, του Vulkan και του OpenCV αποτέλεσε μια πολύπλοκη διαδικασία που απαιτούσε βαθιά κατανόηση των εργαλείων και των τεχνολογιών. Οι κύριες προκλήσεις που αντιμετωπίστηκαν περιλάμβαναν:

- **Συντονισμός μεταξύ διαφορετικών τεχνολογιών:** Η ενσωμάτωση Java, C++, Vulkan και OpenCV δημιούργησε ένα πολυεπίπεδο σύστημα, όπου κάθε μέρος έπρεπε να λειτουργεί αρμονικά με το άλλο. Ο συγχρονισμός αυτός απαιτούσε προσεκτικό σχεδιασμό και εκτεταμένο debugging για να διασφαλιστεί η σταθερότητα και η απόδοση της εφαρμογής.
- **Ελαχιστοποίηση καθυστερήσεων:** Λόγω της φύσης της εφαρμογής (ανίχνευση αντικειμένων σε πραγματικό χρόνο), η ελαχιστοποίηση των καθυστερήσεων ήταν κρίσιμη. Η διαχείριση της μνήμης, η βελτιστοποίηση του κώδικα και η χρήση πολυνηματικών διαδικασιών ήταν μερικές από τις βασικές τεχνικές που χρησιμοποιήθηκαν για την αντιμετώπιση αυτής της πρόκλησης.

Με τη σωστή χρήση του JNI και την ενσωμάτωση του Vulkan και του OpenCV, η εφαρμογή κατάφερε να επιτύχει την απαραίτητη απόδοση για τη χρήση σε κινητά τηλέφωνα, προσφέροντας μια αποδοτική λύση για την ανίχνευση αντικειμένων σε πραγματικό χρόνο.

5.2 Σημασία της συμβατότητας με το λειτουργικό σύστημα

Ένα από τα πιο κρίσιμα ζητήματα στην υλοποίηση μιας απαιτητικής εφαρμογής για κινητά τηλέφωνα είναι η συμβατότητα με το λειτουργικό σύστημα Android. Το Android έχει πολλές διαφορετικές εκδόσεις, και κάθε μία από αυτές διαθέτει διαφορετικό επίπεδο υποστήριξης εργαλείων και τεχνολογιών που είναι απαραίτητα για την ανάπτυξη εφαρμογών. Στην περίπτωση αυτής της εφαρμογής, οι προηγούμενες εκδόσεις του Android παρουσίασαν σημαντικά προβλήματα συμβατότητας λόγω της αδυναμίας υποστήριξης

κατάλληλων εκδόσεων των εργαλείων που είναι κρίσιμα για την κατασκευή του τελικού εκτελέσιμου αρχείου.

Τα κύρια ζητήματα συμβατότητας αφορούσαν:

- **CMake:** Η χρήση του CMake για την κατασκευή του κώδικα C/C++ και την ενσωμάτωσή του με τον κώδικα Java μέσω του JNI απαιτεί συγκεκριμένες εκδόσεις του CMake, οι οποίες δεν υποστηρίζονταν από παλιότερες εκδόσεις του Android. Χωρίς τη σωστή έκδοση του CMake, ήταν αδύνατη η ορθή δημιουργία και σύνδεση των native βιβλιοθηκών που απαιτούνται για την εκτέλεση των αλγορίθμων μηχανικής όρασης.
- **Gradle:** Το Gradle είναι το βασικό εργαλείο κατασκευής για Android εφαρμογές και η σωστή έκδοση του είναι απαραίτητη για τη διαχείριση εξαρτήσεων και τη δημιουργία των τελικών APK αρχείων. Σε παλιότερες εκδόσεις του Android, οι εκδόσεις του Gradle που απαιτούνται για την υποστήριξη προηγμένων τεχνολογιών όπως το Vulkan και το OpenCV δεν ήταν διαθέσιμες, δημιουργώντας περιορισμούς στην ανάπτυξη της εφαρμογής.
- **JDK (Java Development Kit):** Το JDK είναι εξίσου κρίσιμο εργαλείο για την ανάπτυξη Android εφαρμογών, καθώς η γλώσσα προγραμματισμού Java είναι η κύρια γλώσσα ανάπτυξης στο Android. Παλαιότερες εκδόσεις του Android δεν υποστήριζαν τις σύγχρονες εκδόσεις του JDK που απαιτούνται για τη σωστή ενσωμάτωση της Java με native βιβλιοθήκες μέσω JNI, προκαλώντας πρόσθετα προβλήματα στη σύνταξη και εκτέλεση της εφαρμογής.

Αυτές οι τεχνολογικές εξαρτήσεις καθιστούν κρίσιμο το γεγονός ότι οι παλαιότερες εκδόσεις του Android δεν είναι πάντα σε θέση να υποστηρίξουν σύγχρονες τεχνολογίες και εργαλεία. Η συμβατότητα με τις τελευταίες εκδόσεις του λειτουργικού συστήματος Android διασφαλίζει ότι η εφαρμογή μπορεί να αξιοποιήσει πλήρως τις δυνατότητες του υλικού και των εργαλείων ανάπτυξης, προσφέροντας βελτιωμένη απόδοση και σταθερότητα.

5.3 Ενσωμάτωση βελτιστοποιημένων μοντέλων στην εφαρμογή

Η ενσωμάτωση των βελτιστοποιημένων μοντέλων στην εφαρμογή Android ήταν μια διαδικασία γεμάτη προκλήσεις, καθώς κάθε μοντέλο είχε τα δικά του τεχνικά χαρακτηριστικά και απαιτήσεις. Στο προηγούμενο κεφάλαιο αναφέρθηκαν διάφορα μοντέλα, όπως τα MobileNetV2/V3, EfficientDet, ResNet50, και YOLOv7, καθένα από τα οποία παρουσίασε διαφορετικά ζητήματα κατά την προσπάθεια ενσωμάτωσής του στην εφαρμογή.

MobileNetV2/V3

Η ενσωμάτωση των μοντέλων **MobileNetV2** και **MobileNetV3**, που έχουν σχεδιαστεί για χρήση σε κινητές συσκευές, αρχικά φαινόταν να είναι η πιο εύκολη επιλογή λόγω του μικρού μεγέθους τους και της ευκολίας βελτιστοποίησης. Ωστόσο, οι προκλήσεις που αντιμετώπισα περιλάμβαναν:

- **Απώλεια ακρίβειας μετά τη βελτιστοποίηση:** Παρά το γεγονός ότι τα MobileNet μοντέλα είναι σχεδιασμένα για κινητές συσκευές, η εφαρμογή τεχνικών όπως το quantization και το pruning προκάλεσε αισθητή πτώση στην ακρίβεια. Αυτή η πτώση έκανε τη χρήση τους ακατάλληλη για εφαρμογές που απαιτούν υψηλή ακρίβεια στην ανίχνευση αντικειμένων.
- **Συμβατότητα με το JNI και το OpenCV:** Ενώ τα MobileNet μοντέλα ήταν εύκολο να ενσωματωθούν μέσω TensorFlow Lite, η διασύνδεσή τους με το JNI για την αξιοποίηση του Vulkan και του OpenCV προκάλεσε επιπλέον δυσκολίες στην επικοινωνία μεταξύ των διαφορετικών στοιχείων της εφαρμογής. Η σύνδεση των εργαλείων δεν ήταν τόσο ομαλή όσο αναμενόταν, με αποτέλεσμα να υπάρχουν προβλήματα απόδοσης.

EfficientDet

Η ενσωμάτωση του **EfficientDet** αντιμετώπισε άλλες σημαντικές προκλήσεις:

- **Μεγάλο μέγεθος μοντέλου:** Το EfficientDet είναι αρκετά βαρύ σε πόρους, και παρά τη βελτιστοποίηση με pruning και quantization, το μέγεθος του μοντέλου παρέμεινε μεγάλο για αποδοτική εκτέλεση σε κινητές συσκευές. Αυτό δημιούργησε δυσκολίες όσον αφορά τη μνήμη και τη χρήση της GPU των κινητών τηλεφώνων.

- **Χαμηλή απόδοση σε πραγματικό χρόνο:** Ακόμη και μετά την εφαρμογή των τεχνικών βελτιστοποίησης, το EfficientDet δεν μπορούσε να προσφέρει την απαιτούμενη ταχύτητα για εφαρμογές πραγματικού χρόνου. Η καθυστέρηση στην ανίχνευση αντικειμένων ήταν αρκετή για να επηρεάσει την εμπειρία χρήστη, γεγονός που το καθιστούσε ακατάλληλο για την εφαρμογή.

ResNet50

Το **ResNet50** είναι ένα ισχυρό μοντέλο για πολλές εφαρμογές μηχανικής όρασης, αλλά η ενσωμάτωσή του στην εφαρμογή παρουσίασε πολλές προκλήσεις:

- **Υπολογιστική πολυπλοκότητα:** Το ResNet50 απαιτεί υψηλή υπολογιστική ισχύ για να λειτουργεί αποδοτικά, κάτι που δημιουργούσε προβλήματα στην υλοποίηση του σε κινητές συσκευές. Ακόμα και με τις βελτιστοποιήσεις, το μοντέλο παρέμενε αρκετά "βαρύ" για εκτέλεση σε πραγματικό χρόνο, με αποτέλεσμα καθυστερήσεις και χαμηλό ρυθμό καρέ (FPS).
- **Συμβατότητα με Android και Vulkan:** Λόγω της πολυπλοκότητάς του, η ενσωμάτωση του ResNet50 με το JNI και το Vulkan απαιτούσε περισσότερη προσρμογή και debugging, καθώς το μοντέλο δεν ήταν αρχικά σχεδιασμένο για χρήση σε περιβάλλοντα χαμηλών πόρων όπως οι κινητές συσκευές.

YOLOv7

Το **YOLOv7**, το οποίο τελικά επιλέχθηκε ως το καταλληλότερο μοντέλο, παρουσίασε τις λιγότερες προκλήσεις όσον αφορά την ενσωμάτωση του, αλλά δεν ήταν απαλλαγμένο από δυσκολίες:

- **Μεγέθος και πολυπλοκότητα μοντέλου:** Το YOLOv7, παρά τις βελτιώσεις σε απόδοση και μέγεθος, εξακολουθεί να είναι αρκετά σύνθετο μοντέλο για κινητές συσκευές. Η βελτιστοποίηση του μέσω quantization και pruning έπρεπε να γίνει με μεγάλη προσοχή, ώστε να μην υποβαθμιστεί η ακρίβεια του μοντέλου, ειδικά σε εφαρμογές πραγματικού χρόνου.
- **Προσαρμογή για χρήση με OpenCV και Vulkan:** Παρά τις βελτιστοποιήσεις, η σύνδεση του YOLOv7 με το Vulkan και το OpenCV μέσω JNI δεν ήταν χωρίς προβλήματα. Η αποδοτική εκτέλεση του μοντέλου απαιτούσε δοκιμές για την

σωστή αξιοποίηση και τη ρύθμιση της CPU, της GPU και της μνήμης του συστήματος, ώστε να επιτευχθεί υψηλή ταχύτητα ανίχνευσης με παράλληλη διατήρηση της ακρίβειας.

Συνολικά, η ενσωμάτωση των διαφόρων μοντέλων στην εφαρμογή αποτέλεσε μια διαδικασία γεμάτη προκλήσεις. Κάθε μοντέλο παρουσίασε διαφορετικά προβλήματα και απαιτούσε συγκεκριμένες προσαρμογές για να καταφέρει να λειτουργήσει αποδοτικά σε κινητές συσκευές. Το YOLOv7, αν και παρουσίασε κάποιες δυσκολίες, αποδείχθηκε το πιο κατάλληλο μοντέλο για τις ανάγκες της εφαρμογής λόγω της ισορροπίας του μεταξύ ακρίβειας και ταχύτητας.

6 Συμπεράσματα για την Εφαρμογή Android

6.1 Αποτελέσματα βελτιστοποίησης

Η βελτιστοποίηση του μοντέλου YOLOv7 για χρήση σε κινητές συσκευές ήταν μία από τις σημαντικότερες επιτυχίες της εφαρμογής. Με τη χρήση τεχνικών όπως το **quantization** και το **model pruning**, καταφέραμε να πετύχουμε μετρήσεις που πλησιάζουν αυτές του YOLOv7-tiny, το οποίο είναι σχεδιασμένο για κινητές συσκευές. Συγκεκριμένα, τα αποτελέσματα της βελτιστοποίησης περιλαμβάνουν:

- **Μείωση μεγέθους αρχείου των weights:** Μετά την εφαρμογή των βελτιστοποιήσεων, το μέγεθος του αρχείου με τα weights μειώθηκε κατά περίπου **85%**. Αυτή η σημαντική μείωση καθιστά το μοντέλο πολύ πιο εύχρηστο για κινητές συσκευές, επιτρέποντας την ταχύτερη μεταφορά και αποθήκευση του.
- **Μείωση μεγέθους στη μνήμη κατά την εκτέλεση:** Η χρήση των τεχνικών βελτιστοποίησης συνέβαλε στη μείωση του μεγέθους της μνήμης που καταλαμβάνει το μοντέλο κατά την εκτέλεση, με το ποσοστό της μείωσης να φτάνει το **90%**. Αυτό το αποτέλεσμα είναι κρίσιμο για τις κινητές συσκευές, όπου οι πόροι μνήμης είναι περιορισμένοι και απαιτείται αποτελεσματική διαχείρισή τους για την εκτέλεση μοντέλων σε πραγματικό χρόνο.

Αυτά τα αποτελέσματα επιβεβαιώνουν την επιτυχία της βελτιστοποίησης του YOLOv7 και δείχνουν ότι η χρήση προηγμένων τεχνικών βελτιστοποίησης μπορεί να επιτρέψει την εκτέλεση σύνθετων αλγορίθμων ανίχνευσης αντικειμένων σε κινητές συσκευές με περιορισμένους πόρους, χωρίς να μειώνεται σημαντικά η ακρίβεια ή η ταχύτητα.

6.2 Απόδοση σε κινητές συσκευές

Για την αξιολόγηση της απόδοσης του βελτιστοποιημένου YOLOv7 σε κινητές συσκευές, πραγματοποιήθηκαν δοκιμές σε διάφορες συσκευές με διαφορετικές προδιαγραφές υλικού. Οι μετρήσεις επικεντρώθηκαν σε τρεις βασικούς δείκτες:

Frames per second (FPS), Precision (Ακρίβεια) και Recall (Ανάκληση). Τα αποτελέσματα της βελτιστοποίησης συγκρίθηκαν με τα αρχικά αποτελέσματα του YOLOv7 πριν από τη βελτιστοποίηση, όπως καταγράφηκαν στο baseline.

Frames per second (FPS)

Η απόδοση του βελτιστοποιημένου μοντέλου σε ό,τι αφορά το FPS ήταν αισθητά χαμηλότερη σε σχέση με τις επιδόσεις σε έναν συμβατικό υπολογιστή με κάρτα γραφικών υψηλής απόδοσης. Συγκεκριμένα, σε κινητές συσκευές οι μετρήσεις κυμάνθηκαν μεταξύ **11 και 18 FPS**, ανάλογα με τη συσκευή και το διαθέσιμο υλικό.

- Συσκευές με ισχυρότερες κάρτες γραφικών και περισσότερη μνήμη RAM, όπως τα **high-end smartphones** με **Snapdragon 8 Gen 1**, πλησίασαν το ανώτατο όριο των **18 FPS**.
- Συσκευές με πιο μετριοπαθείς προδιαγραφές, όπως αυτές με **Snapdragon 700 series**, κατέγραψαν FPS κοντά στα **11 FPS**.

Σε σύγκριση με το baseline, όπου το YOLOv7 έφτανε από **45 έως 70 FPS** σε υπολογιστές με κάρτες γραφικών NVIDIA (σειρά 3 ή 4), η απόδοση στις κινητές συσκευές είναι σημαντικά χαμηλότερη, αλλά επαρκής για εφαρμογές πραγματικού χρόνου σε φορητές συσκευές.

Precision (Ακρίβεια)

Όσον αφορά την ακρίβεια του βελτιστοποιημένου μοντέλου, το **Precision** μειώθηκε κατά περίπου **3% - 5%** σε σύγκριση με το baseline. Συγκεκριμένα, οι μετρήσεις κυμάνθηκαν από **50% έως 57%**, ανάλογα με το dataset και τη συσκευή, ενώ στο baseline το Precision είχε καταγραφεί μεταξύ **55% και 60%**.

Αν και η ελαφρά αυτή μείωση είναι αναμενόμενη λόγω των βελτιστοποιήσεων (quantization, pruning), η ακρίβεια παραμένει σε αποδεκτά επίπεδα για την ανίχνευση αντικειμένων σε πραγματικό χρόνο.

Recall (Ανάκληση)

Παρόμοια με το Precision, το **Recall** παρουσίασε μικρή μείωση σε σχέση με την αρχική απόδοση του μοντέλου. Οι μετρήσεις έδειξαν ότι το Recall κυμάνθηκε μεταξύ **60% και 67%**, δηλαδή **3% - 5%** χαμηλότερο από το baseline (65%-70%).

Παρά τη μικρή πτώση στο Recall, το μοντέλο εξακολουθεί να είναι ικανό να ανιχνεύει τα περισσότερα αντικείμενα στις εικόνες, καθιστώντας το κατάλληλο για χρήση σε πραγματικό χρόνο στις περισσότερες εφαρμογές.

Συνολική Αξιολόγηση

Σε γενικές γραμμές, η βελτιστοποιημένη έκδοση του YOLOv7 διατήρησε τις επιδόσεις της σε αποδεκτά επίπεδα, παρά τις απαιτητικές βελτιστοποιήσεις. Αν και παρατηρείται μια μείωση στην ταχύτητα (FPS) και ελαφρά πτώση στην ακρίβεια (Precision και Recall), τα αποτελέσματα θεωρούνται επαρκή για χρήση σε κινητές συσκευές, ιδίως σε περιβάλλοντα που απαιτούν ανίχνευση αντικειμένων σε πραγματικό χρόνο.

6.3 Αξιολόγηση της επίδρασης στην απόδοση

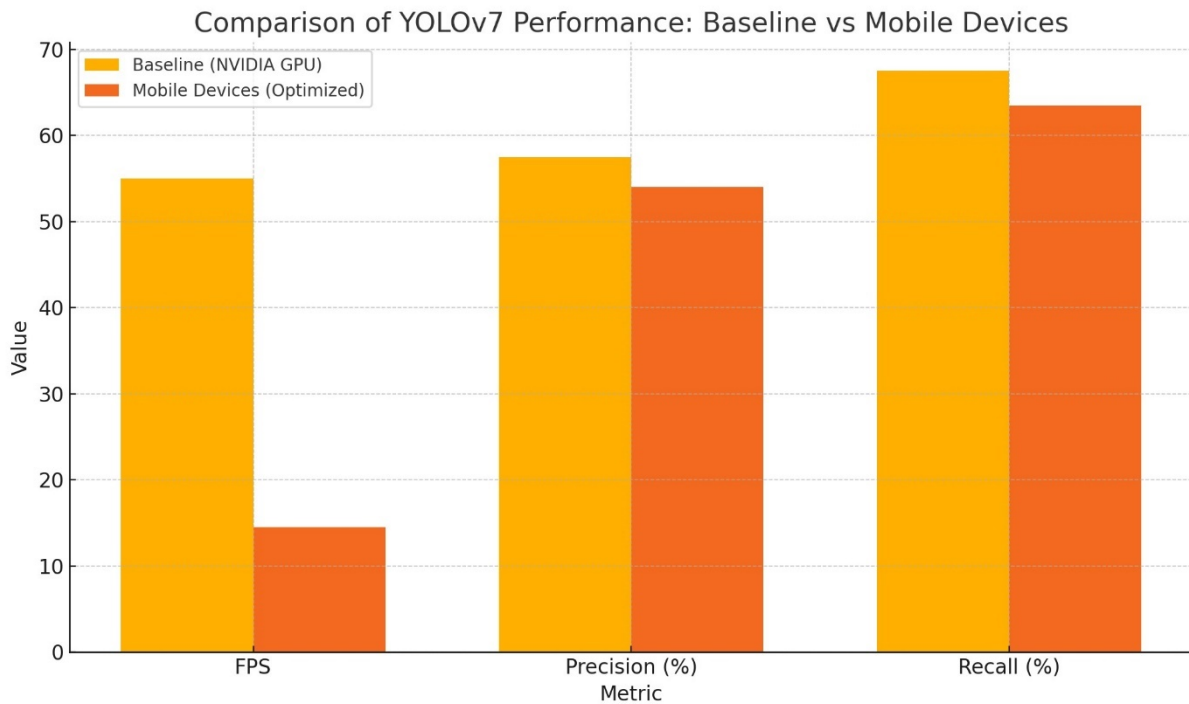
Η επίδραση των βελτιστοποιήσεων στην απόδοση του YOLOv7 ήταν αναμενόμενη, δεδομένων των περιορισμένων πόρων των κινητών συσκευών σε σχέση με τους συμβατικούς υπολογιστές με ισχυρές κάρτες γραφικών. Οι τεχνικές βελτιστοποίησης, όπως το quantization και το pruning, μειώνουν το μέγεθος του μοντέλου και την κατανάλωση μνήμης, γεγονός που επιτρέπει την εκτέλεσή του σε συσκευές με περιορισμένους πόρους. Ωστόσο, αυτές οι βελτιώσεις συχνά συνοδεύονται από μικρή πτώση στην ακρίβεια (Precision και Recall) και στη συνολική ταχύτητα (FPS). Η συγκεκριμένη απόδοση είναι αναμενόμενη λόγω των συμβιβασμών που απαιτούνται για να καταστεί δυνατή η εκτέλεση σύνθετων μοντέλων ανίχνευσης αντικειμένων σε πραγματικό χρόνο σε συσκευές με χαμηλή υπολογιστική ισχύ, όπως τα κινητά τηλέφωνα. Παρά τη μείωση, οι επιδόσεις του μοντέλου παραμένουν εντός αποδεκτών ορίων για την επιτυχία της εφαρμογής σε κινητές πλατφόρμες.

6.4 Σύγκριση με την αρχική απόδοση

Παρακάτω παρουσιάζεται ένα διάγραμμα που συγκρίνει την απόδοση του YOLOv7 μεταξύ της αρχικής έκδοσης (Baseline) σε υπολογιστή με NVIDIA GPU και της βελτιστοποιημένης έκδοσης σε κινητές συσκευές:

- **FPS:** Η αρχική έκδοση έφτανε περίπου στα 55 FPS σε υπολογιστή, ενώ η βελτιστοποιημένη έκδοση σε κινητές συσκευές κυμαίνεται γύρω στα 14.5 FPS.
- **Precision (Ακρίβεια):** Η ακρίβεια έπεσε από 57.5% στο baseline στο 54% στις κινητές συσκευές.

- **Recall (Ανάκληση):** Το recall μειώθηκε από 67.5% στο 63.5% στις κινητές συσκευές.



6.5 Προοπτικές βελτίωσης και περαιτέρω ανάπτυξη

Παρά τα ενθαρρυντικά αποτελέσματα της παρούσας έρευνας, υπάρχουν ακόμα πολλές προοπτικές βελτίωσης και τεχνικές που μπορούν να εφαρμοστούν για την περαιτέρω βελτιστοποίηση των μοντέλων μηχανικής όρασης σε κινητές συσκευές. Ορισμένες από αυτές τις τεχνικές δεν χρησιμοποιήθηκαν στην τρέχουσα εργασία, αλλά προσφέρουν σημαντικές δυνατότητες για τη μείωση της πολυπλοκότητας των μοντέλων και τη βελτίωση της απόδοσης τους.

1. Knowledge Distillation

Η **Knowledge Distillation** είναι μια τεχνική όπου ένα μεγάλο, ισχυρό μοντέλο (Teacher) εκπαιδεύει ένα μικρότερο μοντέλο (Student) μεταφέροντας του τις "γνώσεις" του. Αυτή η τεχνική επιτρέπει στο μικρότερο μοντέλο να επιτύχει υψηλότερη ακρίβεια χωρίς να χρειάζεται να είναι τόσο περίπλοκο όσο το Teacher μοντέλο. Η Knowledge Distillation είναι εξαιρετικά αποδοτική σε περιπτώσεις όπου απαιτείται ελαχιστοποίηση του μεγέθους του μοντέλου, όπως στις κινητές συσκευές.

- Σχετικό Paper: **Hinton, G., et al. (2015), "Distilling the Knowledge in a Neural Network"** [Link](#)

2. Neural Architecture Search (NAS)

Το **Neural Architecture Search (NAS)** είναι μια διαδικασία αυτόματης σχεδίασης αρχιτεκτονικών νευρωνικών δικτύων που μπορούν να επιτύχουν βέλτιστη απόδοση για ένα συγκεκριμένο σύνολο δεδομένων και υπολογιστικούς περιορισμούς. Το NAS μπορεί να εξερευνήσει και να ανακαλύψει πιο αποδοτικές αρχιτεκτονικές που δεν είναι προφανείς με τον παραδοσιακό χειρωνακτικό σχεδιασμό. Η χρήση NAS θα μπορούσε να βοηθήσει στην εύρεση βέλτιστων μοντέλων για κινητές συσκευές που προσφέρουν υψηλή απόδοση με μικρότερη κατανάλωση πόρων.

- Σχετικό Paper: **Zoph, B., et al. (2017), "Neural Architecture Search with Reinforcement Learning"** [Link](#)

3. Low-Rank Factorization

Η **Low-Rank Factorization** μειώνει την πολυπλοκότητα των πλήρως συνδεδεμένων στρωμάτων ενός νευρωνικού δικτύου διασπώντας τα βάρη των στρωμάτων σε πίνακες χαμηλότερης τάξης. Αυτή η τεχνική μπορεί να μειώσει σημαντικά τον αριθμό των παραμέτρων του μοντέλου, διατηρώντας παράλληλα την απόδοσή του. Είναι ιδιαίτερα χρήσιμη όταν τα δίκτυα είναι πολύ μεγάλα για να χωρέσουν σε συσκευές με περιορισμένη μνήμη.

- Σχετικό Paper: **Denton, E., et al. (2014), "Exploiting Linear Structure Within Convolutional Networks for Efficient Evaluation"** [Link](#)

4. Dynamic Inference

Η **Dynamic Inference** είναι μια τεχνική όπου το μοντέλο προσαρμόζει τη συμπεριφορά του ανάλογα με την πολυπλοκότητα των εισόδων. Αντί να εκτελεί όλα τα στρώματα για κάθε είσοδο, το μοντέλο μπορεί να αποφασίσει δυναμικά πόσα στρώματα να εκτελέσει ή πόσα χαρακτηριστικά να χρησιμοποιήσει, ανάλογα με τη δυσκολία της εργασίας. Αυτή η τεχνική μειώνει την περιττή επεξεργασία και εξοικονομεί υπολογιστικούς πόρους.

- Σχετικό Paper: **Teerapittayanon, S., et al. (2016), "BranchyNet: Fast Inference via Early Exiting from Deep Neural Networks"** [Link](#)

5. Pruned Neural Networks on the Edge

Μια ακόμη εξέλιξη στον τομέα των βελτιστοποιημένων μοντέλων για κινητές συσκευές είναι η εφαρμογή **pruned νευρωνικών δικτύων για την άκρη (edge computing)**. Το pruning, το οποίο αφαιρεί τα λιγότερο σημαντικά νευρώνες και συνδέσεις, μπορεί να συνδυαστεί με τεχνικές edge computing για την εκτέλεση ελαφριών μοντέλων απευθείας στις συσκευές, χωρίς την ανάγκη συνεχούς επικοινωνίας με τον server.

- Σχετικό Paper: **Han, S., et al. (2015), "Deep Compression: Compressing Deep Neural Networks with Pruning, Trained Quantization and Huffman Coding"** [Link](#)

6. Mixed-Precision Inference

Η **Mixed-Precision Inference** είναι μια τεχνική που επιτρέπει στα νευρωνικά δίκτυα να χρησιμοποιούν διαφορετικές ακρίβειες αριθμητικών πράξεων (όπως float16 και float32) ανάλογα με τις ανάγκες κάθε στρώματος. Αυτή η τεχνική μειώνει την κατανάλωση μνήμης και βελτιώνει την απόδοση χωρίς να επηρεάζει σημαντικά την ακρίβεια του μοντέλου. Είναι ιδιαίτερα χρήσιμη σε περιπτώσεις όπου οι πόροι είναι περιορισμένοι, όπως στις κινητές συσκευές.

- Σχετικό Paper: **Micikevicius, P., et al. (2017), "Mixed Precision Training"** [Link](#)

Προοπτικές Εφαρμογής

Η ενσωμάτωση αυτών των τεχνικών στο υπάρχον μοντέλο YOLOv7 και η περαιτέρω ανάπτυξη της εφαρμογής για κινητές συσκευές θα μπορούσαν να οδηγήσουν σε ακόμα μεγαλύτερες βελτιώσεις στην απόδοση και στη μείωση του μεγέθους των μοντέλων. Οι παραπάνω τεχνικές θα μπορούσαν να προσφέρουν πιο ελαφριά και αποδοτικά μοντέλα, χωρίς σημαντική απώλεια στην ακρίβεια, επιτρέποντας τη χρήση τους σε μια ευρύτερη γκάμα φορητών συσκευών και βιομηχανικών εφαρμογών. Για τη βελτίωση της απόδοσης και την περαιτέρω ανάπτυξη της εφαρμογής, τεχνικές όπως το **Knowledge Distillation** και το **Neural Architecture Search (NAS)** μπορούν να βοηθήσουν στην ανακάλυψη αποδοτικών αρχιτεκτονικών και στη μείωση της πολυπλοκότητας των μοντέλων. Το **Low-Rank Factorization** και η **Mixed-Precision Inference** προσφέρουν επιπλέον δυνατότητες για τη μείωση της κατανάλωσης μνήμης και των υπολογιστικών απαιτήσεων,

ενώ η **Dynamic Inference** και το **pruning** μπορούν να μειώσουν την υπερβολική επεξεργασία, εξασφαλίζοντας καλύτερη απόδοση σε κινητές συσκευές. Πρακτικές συμβουλές για development περιλαμβάνουν τη συνεχή χρήση profiling tools όπως το **Android Profiler** για τη μέτρηση της χρήσης πόρων, τον τακτικό έλεγχο της συμβατότητας με διαφορετικές εκδόσεις του Android και τη δοκιμή διαφορετικών αλγορίθμων βελτιστοποίησης για την εξισορρόπηση ακρίβειας και απόδοσης

7 Συνολικά Συμπεράσματα

7.1 Συνδυασμός αποτελεσμάτων από τη βελτιστοποίηση μοντέλων και την υλοποίηση εφαρμογής

Συνολικά, είμαστε ικανοποιημένοι από τα αποτελέσματα της βελτιστοποίησης του YOLOv7 και την ενσωμάτωσή του στην εφαρμογή για κινητές συσκευές. Η απόδοση του μοντέλου, μετά τις βελτιστοποιήσεις, ικανοποίησε τις απαιτήσεις σε ακρίβεια και ταχύτητα, επιτυγχάνοντας παράλληλα τη μείωση των απαιτήσεων σε μνήμη και επεξεργαστική ισχύ. Ωστόσο, αναλογιζόμενος τη συνολική διαδικασία ανάπτυξης, διαπιστώνω ότι θα μπορούσα να είχα διατηρήσει και τις υλοποιήσεις των άλλων μοντέλων, όπως τα MobileNet και EfficientDet, τα οποία αρχικά αφαιρέθηκαν λόγω μη ενθαρρυντικών benchmarks μετά τις βελτιστοποιήσεις. Παρά τη μειωμένη απόδοση σε σχέση με το YOLOv7, η διατήρησή τους θα μπορούσε να προσφέρει περισσότερες επιλογές στο χρήστη και ευελιξία στη δοκιμή και εκμετάλλευση της εφαρμογής, διευκολύνοντας πιθανώς μελλοντικές προσαρμογές και βελτιώσεις.

7.2 Τελική αξιολόγηση της συνολικής εργασίας

Η συνολική εργασία πέτυχε τους αρχικούς της στόχους, καθώς καταφέραμε να βελτιστοποιήσουμε το YOLOv7 για χρήση σε κινητές συσκευές, επιτυγχάνοντας ικανοποιητική απόδοση σε πραγματικό χρόνο με μειωμένες απαιτήσεις σε πόρους. Ωστόσο, λόγω της περιπλοκότητας της υλοποίησης και των υψηλών απαιτήσεων που συνοδεύουν τις προχωρημένες τεχνικές βελτιστοποίησης και ενσωμάτωσης μοντέλων, καταλήξαμε σε μια πιο απλοϊκή αλλά εξαιρετικά χρήσιμη εφαρμογή. Η εφαρμογή αυτή, αν και απλούστερη από ό,τι αρχικά προβλεπόταν, διαθέτει ισχυρές βάσεις και πολλές προοπτικές για περαιτέρω ανάπτυξη και βελτιώσεις, καθιστώντας την πολύτιμο εργαλείο για ανίχνευση αντικειμένων σε κινητές πλατφόρμες.

7.3 Μελλοντικές βελτιώσεις

Πέρα από τις τεχνικές βελτιστοποιήσεις, υπάρχουν πολλές μελλοντικές βελτιώσεις που μπορούν να γίνουν στο **UI/UX** της εφαρμογής για να βελτιωθεί η εμπειρία του χρήστη. Η απλότητα και η ταχύτητα στην πλοήγηση της εφαρμογής είναι κρίσιμα στοιχεία, ιδιαίτερα για χρήστες που ενδέχεται να μην έχουν εμπειρία με τεχνολογίες μηχανικής όρασης. Ενδεικτικά, θα μπορούσε να προστεθεί ένας πιο φιλικός προς το χρήστη οδηγός ρυθμίσεων και μια διαδραστική παρουσίαση των ανιχνευμένων αντικειμένων, με δυνατότητα για προσαρμογές του interface ανάλογα με τις ανάγκες του χρήστη.

Όσον αφορά την εφαρμογή σε πραγματικό περιβάλλον, η χρήση της για **ανίχνευση πτώσεων ανθρώπων, αυτοματοποιημένο έλεγχο μέτρησης προϊόντων** ή ακόμη και για **έξυπνες οικιακές εφαρμογές** αποτελεί σημαντική προοπτική. Η ενσωμάτωση της εφαρμογής σε περιβάλλοντα όπως εργοστάσια ή αποθήκες, για ανίχνευση και παρακολούθηση προϊόντων, θα μπορούσε να αυξήσει την αποτελεσματικότητα των διαδικασιών. Επιπλέον, η αξιοποίηση της σε συστήματα **έξυπνων πόλεων** ή ακόμη και για **εφαρμογές ασφαλείας** (σε δημόσιους χώρους) παρέχει ένα ευρύ πεδίο για την πρακτική χρήση της εφαρμογής σε πραγματικό χρόνο.

8 Λέξεις Κλειδιά

8.1 Αναδεικνύοντας τα βασικά θέματα με λέξεις-κλειδιά.

Αγγλικά:

Keywords:

Machine learning

Computer vision

Mobile devices

Model optimization

Performance evaluation

Android

Image processing

Object detection

Real-time inference

Quantization

Knowledge distillation

Ελληνικά:

Λέξεις-κλειδιά:

Μηχανική μάθηση

Υπολογιστική όραση

Κινητές συσκευές

Βελτιστοποίηση μοντέλων

Αξιολόγηση απόδοσης

Android

Επεξεργασία εικόνας

Ανίχνευση αντικειμένων
Συμπερασμός σε πραγματικό χρόνο
Ποσοτικοποίηση
Απόσταξη γνώσης
Τεχνητή νοημοσύνη
Ενσωματωμένα συστήματα
Νευρωνικά δίκτυα
Βαθιά μάθηση
Εφαρμογές κινητών
Διασύνδεση ανθρώπου-μηχανής
Ενίσχυση μάθησης
Αναγνώριση προτύπων
Ανάλυση δεδομένων

Βιβλιογραφία

- [1] ImageNet ILSVRC Dataset: Olga Russakovsky*, Jia Deng*, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg and Li Fei-Fei. (* = equal contribution) ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision, 2015.
<http://www.image-net.org/>
- [2] COCO Dataset: Microsoft COCO: Common Objects in Context, Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, Piotr Dollár.
<https://arxiv.org/abs/1405.0312>, <https://cocodataset.org/>
- [3] PascalVOC Dataset: The PASCAL Visual Object Classes (VOC) Challenge, Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J. and Zisserman, A., International Journal of Computer Vision, 88(2), 303-338, 2010.
<http://host.robots.ox.ac.uk/pascal/VOC/>
- [4] ADE20K Scene Parsing Dataset Scene Parsing through ADE20K Dataset. Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso and Antonio Torralba. Computer Vision and Pattern Recognition (CVPR), 2017. Semantic Understanding of Scenes through ADE20K Dataset. Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso and Antonio Torralba. International Journal on Computer Vision (IJCV).
<https://groups.csail.mit.edu/vision/datasets/ADE20K/>,
<https://sceneparsing.csail.mit.edu/>
- [5] Cityscapes Dataset: M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, “The Cityscapes Dataset for Semantic Urban Scene Understanding,” in Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
<https://www.cityscapes-dataset.com/>

- [6] MMDetection: Open MMLab Detection Toolbox and Benchmark, Chen, Kai and Wang, Jiaqi and Pang, Jiangmiao and Cao, Yuhang and Xiong, Yu and Li, Xiaoxiao and Sun, Shuyang and Feng, Wansen and Liu, Ziwei and Xu, Jiarui and Zhang, Zheng and Cheng, Dazhi and Zhu, Chenchen and Cheng, Tianheng and Zhao, Qijie and Li, Buyu and Lu, Xin and Zhu, Rui and Wu, Yue and Dai, Jifeng and Wang, Jingdong and Shi, Jianping and Ouyang, Wanli and Loy, Chen Change and Lin, Dahua. arXiv:1906.07155, 2019
- [7] SSD: Single Shot MultiBox Detector, Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, Alexander C. Berg. In the Proceedings of the European Conference on Computer Vision (ECCV), 2016.
- [8] MLPerf Inference Benchmark, Vijay Janapa Reddi and Christine Cheng and David Kanter and Peter Mattson and Guenther Schmuelling and Carole-Jean Wu and Brian Anderson and Maximilien Breughe and Mark Charlebois and William Chou and Ramesh Chukka and Cody Coleman and Sam Davis and Pan Deng and Greg Diamos and Jared Duke and Dave Fick and J. Scott Gardner and Itay Hubara and Sachin Idgunji and Thomas B. Jablin and Jeff Jiao and Tom St. John and Pankaj Kanwar and David Lee and Jeffery Liao and Anton Lokhmotov and Francisco Massa and Peng Meng and Paulius Micikevicius and Colin Osborne and Gennady Pekhimenko and Arun Tejusve Raghunath Rajan and Dilip Sequeira and Ashish Sirasao and Fei Sun and Hanlin Tang and Michael Thomson and Frank Wei and Ephrem Wu and Lingjie Xu and Koichi Yamada and Bing Yu and George Yuan and Aaron Zhong and Peizhao Zhang and Yuchen Zhou, arXiv:1911.02549, 2019
- [9] Pytorch/Torchvision: Torchvision the machine-vision package of torch, Sébastien Marcel, Yann Rodriguez, MM '10: Proceedings of the 18th ACM international conference on Multimedia October 2010 Pages 14851488
<https://doi.org/10.1145/1873951.1874254>
<https://pytorch.org/vision/stable/index.html>
- [10] TensorFlow Model Garden: The TensorFlow Model Garden is a repository with a number of different implementations of state-of-the-art (SOTA) models and modeling solutions for TensorFlow users.
<https://github.com/tensorflow/models>
- [11] TensorFlow Object Detection API: Speed/accuracy trade-offs for modern convolutional object detectors. Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A,

- Fischer I, Wojna Z, Song Y, Guadarrama S, Murphy K, CVPR 2017.
https://github.com/tensorflow/models/tree/master/research/object_detection
- [12] Tensorflow DeepLab: DeepLab: Deep Labelling for Semantic Image Segmentation
<https://github.com/tensorflow/models/tree/master/research/deeplab>
- [13] TensorFlow Official Model Garden, Chen Chen and Xianzhi Du and Le Hou and Jaeyoun Kim and Pengchong, Jin and Jing Li and Yeqing Li and Abdullah Rashwan and Hongkun Yu, 2020.
<https://github.com/tensorflow/models/tree/master/official>
- [14] GluonCV: GluonCV and GluonNLP: Deep Learning in Computer Vision and Natural Language Processing Jian Guo, He He, Tong He, Leonard Lausen, Mu Li, Haibin Lin, Xingjian Shi, Chenguang Wang, Junyuan Xie, Sheng Zha, Aston Zhang, Hang Zhang, Zhi Zhang, Zhongyue Zhang, Shuai Zheng, Yi Zhu.
<https://arxiv.org/abs/1907.04433>
- [15] WIDER FACE: A Face Detection Benchmark, Yang, Shuo and Luo, Ping and Loy, Chen Change and Tang, Xiaoou, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016.
<http://shuoyang1213.me/WIDERFACE/>
- [16] MobileNetV2: LightWeight Deep Convolutional Neural Networks for Mobile and Embedded Vision Applications.
<https://arxiv.org/abs/1704.04861>
- [17] MobileNetV3: Searching for Mobilenetsv3.
<https://arxiv.org/abs/1905.02201>
- [18] ShuffleNet: ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices.
<https://arxiv.org/abs/1707.01501>
- [19] SqueezeNet: SqueezeNet: AlexNet-level accuracy with 50x fewer params.
<https://arxiv.org/abs/1608.06913>
- [20] Knowledge Distillation: A Survey: <https://arxiv.org/abs/1906.06972>
- [21] Quantization-Aware Training.
<https://arxiv.org/abs/1712.05424>

Βιβλιογραφία για αξιολόγηση μοντέλων σε κινητές συσκευές:

- [22] Benchmarking Deep Neural Networks on Mobile and Embedded Devices.
<https://arxiv.org/abs/1807.11439>
- [23] MLPerf Inference Benchmark.
<https://arxiv.org/abs/1911.02549>
- [24] Mobile AI Benchmark.
<https://github.com/XiaoMi/mobile-ai-bench>

Άλλα χρήσιμα άρθρα και πόροι:

- [25] The State of Mobile Machine Learning 2023.
<https://www.sitepen.com/blog/the-state-of-machine-learning-2023>
- [26] A Survey of On-Device Machine Learning.
<https://arxiv.org/abs/2002.03805>

Παράρτημα Α

Σύνδεσμος κώδικα της Μεταπτυχιακής Διπλωματικής Εργασίας [Link](#)

Σύνδεσμος αρχείου εγκατάστασης της εφαρμογής σε κινητό Android [Link](#)